Dynamic Bayesian Networks
Modeling and Analysis of Neural Signals

by

Junning Li

B.Sc., Tsinghua University, 2002
M.Sc., Tsinghua University, 2005

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

in
The Faculty of Graduate Studies
(Electrical and Computer Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA
(Vancouver)
August, 2009
© Junning Li 2009
Abstract

Studying interactions between different brain regions or neural components is crucial in understanding neurological disorders. Dynamic Bayesian networks, a type of statistical graphical model, have been suggested as a promising tool to model neural communication systems. This thesis investigates the employment of dynamic Bayesian networks for analyzing neural connectivity, especially with focus on three topics: structural feature extraction, group analysis, and error control in learning network structures.

Extracting interpretable features from experimental data is important for clinical diagnosis and improving experiment design. A framework is designed for discovering structural differences, such as the pattern of sub-networks, between two groups of Bayesian networks. The framework consists of three components: Bayesian network modeling, statistical structure-comparison, and structure-based classification. In a study on stroke using surface electromyography, this method detected several coordination patterns among muscles that could effectively differentiate patients from healthy people.

Group analyses are widely conducted in neurological research. However for dynamic Bayesian networks, the performances of different group-analysis methods had not been systematically investigated. To provide guidance on selecting group-analysis methods, three popular methods, i.e. the virtual-typical-subject, the common-structure and the individual-structure methods, were compared in a study on Parkinson’s disease, from the aspects of their statistical goodness-of-fit to the data, and more importantly, their sensitivity in detecting the effect of medication. The three methods led to considerably different group-level results, and the individual-structure approach was more sensitive to the normalizing effect of medication.
Abstract

Controlling errors is a fundamental problem in applying dynamic Bayesian networks to discovering neural connectivity. An algorithm is developed for this purpose, particularly for controlling the false discovery rate (FDR). It is proved that the algorithm is able to curb the FDR under user-specified levels (for example, conventionally 5%) at the limit of large sample size, and meanwhile recover all the true connections with probability one. Several extensions are also developed, including a heuristic modification for moderate sample sizes, an adaption to prior knowledge, and a combination with Bayesian inference.
Table of Contents

Abstract ................................................................. ii

Table of Contents ....................................................... iv

List of Tables ........................................................... viii

List of Figures ........................................................... ix

Acknowledgements ......................................................... xi

Statement of Co-Authorship ........................................ xii

1 Introduction and Overview ............................................. 1
  1.1 Discovering Neural Interactions ................................. 1
  1.2 Current Methods .................................................. 2
  1.3 Dynamic Bayesian Networks .................................... 5
  1.4 Applications of Bayesian Networks ............................. 7
  1.5 Thesis Outline ................................................... 10

Bibliography .................................................................. 14

2 Extracting Structural Features from Bayesian Networks ........ 22
  2.1 Introduction .......................................................... 23
  2.2 Methods ............................................................... 27
    2.2.1 Framework and Components ............................... 27
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.2.2 Bayesian Networks</td>
<td>28</td>
</tr>
<tr>
<td>2.2.3 Sub-network Patterns and Muscle Synergies</td>
<td>33</td>
</tr>
<tr>
<td>2.2.4 Classification</td>
<td>35</td>
</tr>
<tr>
<td>2.2.5 Modeling sEMG Signals</td>
<td>38</td>
</tr>
<tr>
<td>2.3 Results</td>
<td>40</td>
</tr>
<tr>
<td>2.3.1 Real sEMG Datasets</td>
<td>40</td>
</tr>
<tr>
<td>2.3.2 Learned Bayesian Networks</td>
<td>42</td>
</tr>
<tr>
<td>2.3.3 Triple Patterns</td>
<td>44</td>
</tr>
<tr>
<td>2.3.4 Classification Performance</td>
<td>47</td>
</tr>
<tr>
<td>2.4 Conclusions and Discussions</td>
<td>52</td>
</tr>
<tr>
<td>Bibliography</td>
<td>56</td>
</tr>
<tr>
<td>3 Comparing Group-Analysis Methods Based on Bayesian Networks</td>
<td>60</td>
</tr>
<tr>
<td>3.1 Introduction</td>
<td>61</td>
</tr>
<tr>
<td>3.2 Materials and Methods</td>
<td>64</td>
</tr>
<tr>
<td>3.2.1 fMRI Data</td>
<td>64</td>
</tr>
<tr>
<td>3.2.2 Dynamic Bayesian Networks</td>
<td>68</td>
</tr>
<tr>
<td>3.2.3 Learning DBNs</td>
<td>71</td>
</tr>
<tr>
<td>3.2.4 Comparison of Three Approaches</td>
<td>73</td>
</tr>
<tr>
<td>3.2.5 Group Comparison</td>
<td>77</td>
</tr>
<tr>
<td>3.3 Results</td>
<td>78</td>
</tr>
<tr>
<td>3.4 Discussion</td>
<td>86</td>
</tr>
<tr>
<td>Bibliography</td>
<td>90</td>
</tr>
<tr>
<td>4 Controlling the Error Rate in Learning Network Structures</td>
<td>95</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>96</td>
</tr>
<tr>
<td>4.1.1 More Recent Related Works</td>
<td>104</td>
</tr>
</tbody>
</table>
### 4.2 Controlling FDR with PC Algorithm
- **4.2.1 Notations and Preliminaries**
- **4.2.2 PC Algorithm**
- **4.2.3 False Discovery Rate**
- **4.2.4 PC Algorithm with FDR**
- **4.2.5 Asymptotic Performance**
- **4.2.6 Computational Complexity**
- **4.2.7 Miscellaneous Discussions**

### 4.3 Empirical Evaluation
- **4.3.1 Simulation Study**
- **4.3.2 Applications to Real fMRI Data**

### 4.4 Conclusions and Discussions

### 4.5 Proof of Theorems

### 4.6 Statistical Tests with Asymptotic Power Equal to One

### 4.7 Error Rates of Interest

### Bibliography

### 5 Extending Error-Rate Control to Dynamic Bayesian Networks
- **5.1 Integration of Prior Knowledge**
- **5.1.1 False Discovery Rate**
- **5.1.2 PC\textsubscript{fdr}\textsubscript{*} Algorithm with Prior Knowledge**
- **5.1.3 Conditional-Independence Test for fMRI Signals**
- **5.1.4 Test of Relative Consistence Among Graphs**
- **5.1.5 Application to Parkinson’s Disease**
- **5.2 Bayesian Inference with FDR-Controlled Prior**
- **5.2.1 Bayesian Inference with Uniform Prior Distribution**
<table>
<thead>
<tr>
<th>Chapter 5</th>
<th>Modelling with Prior Distribution Derived from FDR-Controlled PC Algorithm</th>
<th>170</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chapter 5</td>
<td>Simulation</td>
<td>172</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>Application to Parkinson’s Disease</td>
<td>176</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>Conclusions and Discussions</td>
<td>179</td>
</tr>
<tr>
<td></td>
<td>Bibliography</td>
<td>181</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Selecting Regions of Interest in fMRI Analysis</td>
<td>183</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Introduction</td>
<td>183</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Experimental Methods</td>
<td>191</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>FMRI Image Pre-processing</td>
<td>193</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>LLDA Algorithm</td>
<td>193</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Backward Step-wise Discrimination</td>
<td>197</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Contrasting Multiple Tasks</td>
<td>198</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Significance of Discrimination</td>
<td>199</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>SPM Methodology</td>
<td>200</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Results</td>
<td>201</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Discussion</td>
<td>205</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Technical Appendix</td>
<td>207</td>
</tr>
<tr>
<td></td>
<td>Bibliography</td>
<td>210</td>
</tr>
<tr>
<td>Chapter 7</td>
<td>Conclusions and Discussions</td>
<td>214</td>
</tr>
<tr>
<td>Chapter 7</td>
<td>Summary</td>
<td>214</td>
</tr>
<tr>
<td>Chapter 7</td>
<td>Contributions and Potential Applications</td>
<td>218</td>
</tr>
<tr>
<td>Chapter 7</td>
<td>Future Work</td>
<td>222</td>
</tr>
<tr>
<td></td>
<td>Bibliography</td>
<td>227</td>
</tr>
</tbody>
</table>
# List of Tables

1.1 Publications on discovering neural interactions with Bayesian networks .............................................. 8  
2.1 Muscle coordination patterns significantly different among subjects ...................................................... 46  
2.2 Classification error-rates using sEMG signals ......................................................................................... 48  
3.1 Group-level BIC scores of simulated data ................................................................................................. 79  
3.2 Group-level BIC scores of real data ........................................................................................................ 79  
3.3 Numbers of connections normalized by medication .................................................................................. 82  
3.4 Connections statistically significantly normalized by medication .......................................................... 84  
4.1 Possible outcomes of multiple-hypothesis testing ..................................................................................... 111  
4.2 Criteria for multiple-hypothesis testing .................................................................................................. 111  
4.3 Ratios of extra computational time spent on FDR-control by PC_{fdr} algorithm ................................. 133  
4.4 Ratios of extra computational time spent on FDR-control by PC_{fdr*} algorithm ................................ 133  
4.5 Brain regions involved in bulb-squeezing data set .................................................................................... 136  
4.6 Brain regions involved in sentence-picture data set ................................................................................ 137  
4.7 Realized error rates on bulb-squeezing and sentence-picture data sets ................................................. 138  
5.1 Possible outcomes of multiple-hypothesis testing ..................................................................................... 162
# List of Figures

1.1 Example of dynamic Bayesian networks ........................................... 6

2.1 Representing the interactions of muscle activities with a Bayesian network ....................................................... 28

2.2 Example of essential graphs .......................................................... 31

2.3 Coordination patterns among three muscles ........................................ 34

2.4 Cross-subject validation ................................................................. 37

2.5 Within-subject validation ............................................................... 37

2.6 Auto-correlation of sEMG “carrier” signals ......................................... 39

2.7 Histogram of the temporal distribution of sEMG “carrier” signals .......... 39

2.8 Muscle coordination networks learned from sEMG signals .................. 42

2.9 Structural differences among muscle coordination networks ................ 43

2.10 Boxplot of the frequency of muscle coordination patterns ................. 45

2.11 Classification trees using networks as the classification features .......... 49

2.12 Classification trees using sub-network patterns as the classification features ................................................. 50

2.13 Principal components of the log likelihood of Bayesian networks learned from sEMG signals ................................................................. 51

3.1 Experimental task design ............................................................... 66

3.2 Example of dynamic Bayesian networks ........................................... 69

3.3 Comparisons of three group-analysis methods: at structural level ........ 80

3.4 Comparisons of three group-analysis methods: at parametric level ........ 81
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5</td>
<td>Variances of network structures learned with the “individual-structure” approach</td>
<td>83</td>
</tr>
<tr>
<td>3.6</td>
<td>Connections statistically significantly normalized by medication</td>
<td>85</td>
</tr>
<tr>
<td>4.1</td>
<td>Simulation design: directed acyclic graphs</td>
<td>125</td>
</tr>
<tr>
<td>4.2</td>
<td>Simulation results: the false discovery rate</td>
<td>129</td>
</tr>
<tr>
<td>4.3</td>
<td>Simulation results: the detection power</td>
<td>130</td>
</tr>
<tr>
<td>4.4</td>
<td>Simulation results: the type I error rate</td>
<td>131</td>
</tr>
<tr>
<td>4.5</td>
<td>Simulation results: computation time</td>
<td>132</td>
</tr>
<tr>
<td>4.6</td>
<td>Networks learned from bulb-squeezing data</td>
<td>136</td>
</tr>
<tr>
<td>4.7</td>
<td>Networks learned from sentence-picture data</td>
<td>137</td>
</tr>
<tr>
<td>5.1</td>
<td>Possible connections of dynamic Bayesian networks</td>
<td>164</td>
</tr>
<tr>
<td>5.2</td>
<td>Consistent brain connectivity inferred from fMRI data</td>
<td>168</td>
</tr>
<tr>
<td>5.3</td>
<td>Simulation design: dynamic Bayesian network</td>
<td>173</td>
</tr>
<tr>
<td>5.4</td>
<td>Simulation results: the false discovery rate</td>
<td>174</td>
</tr>
<tr>
<td>5.5</td>
<td>Simulation results: the detection power</td>
<td>175</td>
</tr>
<tr>
<td>5.6</td>
<td>Simulation results: histogram of Bayesian-information-criterion scores</td>
<td>176</td>
</tr>
<tr>
<td>5.7</td>
<td>Brain connectivity inferred from fMRI data</td>
<td>178</td>
</tr>
<tr>
<td>6.1</td>
<td>Comparison of fMRI group-analysis methods</td>
<td>186</td>
</tr>
<tr>
<td>6.2</td>
<td>Activation paradigm for fMRI motor task</td>
<td>192</td>
</tr>
<tr>
<td>6.3</td>
<td>Comparison of group results: EG + IG</td>
<td>202</td>
</tr>
<tr>
<td>6.4</td>
<td>Comparison of group results: EG vs IG</td>
<td>203</td>
</tr>
<tr>
<td>6.5</td>
<td>Discriminability of overall discriminant function across subjects</td>
<td>204</td>
</tr>
</tbody>
</table>
Acknowledgements

This work would not have been possible without the support, enthusiasm and guidance of my supervisor, Dr. Z. Jane Wang, to whom I owe a debt of thanks. I would also like to thank Dr. Martin J. McKeown (Department of Medicine) for his constructive comments and valuable suggestions. Many thanks to Dr. Janice Eng (Department of Physical Therapy), and PhD candidate Samantha Palmer (Department of Medicine), with whom I collaborated to produce the manuscripts contained herein. I would also like to acknowledge the feedback and support from my fellow graduate students in our signal-processing laboratory.

I owe many people for their generosity and support during my four-year study at the University of British Columbia. Dr. Max Hui-Bon-Hoa (Hong Kong University), my spiritual mentor, gave me constant love, support and wise advice during my life challenges. My dear friends, Zhi-Ming Fan, Stephen Ney, Jia Li, Henry Ngan, Michelle Tan, Kevin Khang, Eric Vyskocil, and Xue-Gui Song, not only shared in my excitement, but also encouraged me when I was in frustration during this long PhD journey. Rod Foist, my fellow PhD student, helped me proofread some of my writing. Andrey Vlassov, our excellent software engineer, helped me recover my research data from crashed hard drives. I thank all my friends in Vancouver, Beijing, and California.

The work produced here was funded partially by the University of British Columbia Graduate Fellowship.
Statement of Co-Authorship

This thesis is a compilation of several manuscripts, resulting from collaboration between several researchers. Chapter 2 is based on a research article published in IEEE Transactions on Biomedical Engineering, co-authored with Dr. Z. Jane Wang, Dr. Janice J. Eng and Dr. Martin J. McKeown. Chapter 3 is based on a research article published in NeuroImage, co-authored with Dr. Z. Jane Wang, Samantha J. Palmer and Dr. Martin J. McKeown. Chapter 4 is based on a research article published in the Journal of Machine Learning Research, co-authored with Dr. Z. Jane Wang. Chapter 5 is based on an article published in the Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, and an article accepted by the 3rd International Conference on Bioinformatics and Biomedical Engineering. They were co-authored with Dr. Z. Jane Wang and Dr. Martin J. McKeown. Chapter 6 is based on a research article published in NeuroImage, co-authored with Dr. Martin J. McKeown, Dr. Xuemei Huang, Dr. Mechelle M. Lewis, Dr. Seungshin Rhee, Dr. K.N. Young Truong and Dr. Z. Jane Wang. The authors’ specific contributions to these collaborations are detailed below.

The research outline was designed jointly by the author, Dr. Z. Jane Wang and Dr. Martin J. McKeown. The majority of the research, including literature survey, model design, proofs of theorems, numerical simulation, statistical data analysis and compilation of results, was conducted by the author, with suggestions from Dr. Z. Jane Wang. The manuscripts were primarily drafted by the author, with helpful comments from Dr. Z. Jane Wang and Dr. Martin J. McKeown. The neurological part of Chapter 6 was written by Dr. McKeown. He also wrote the paragraph about the neurological background.
in the introduction of Chapters 2 and 3 and helped on the neurological interpretation of the analysis results in Chapters 2, 3, and 4. Part of the technical review in the introduction of Chapters 2 and 3 was written by Dr. Z. Jane Wang. The paragraph kinetically interpreting the analysis results in Chapter 2 was written by Dr. Janice J. Eng. The description of the data-acquisition process in Chapters 3 and 4 was originally written by Samantha J. Palmer. The statistical parametric mapping in Chapter 6 was conducted by Dr. Xuemei Huang, Dr. Mechelle M. Lewis, Dr. Seungshin Rhee and Dr. K.N. Young Truong.
Chapter 1

Introduction and Overview

1.1 Discovering Neural Interactions

Studying the interactions among neural components, e.g. different regions of the brain and nerve cells driving different muscles, is crucial in understanding neurological disorders. All of our sensations, thoughts and movements trigger neural signals transmitted between nerve cells. If nerve cells cannot communicate correctly, functions are not coordinated and the body does not respond properly. For example, Parkinson’s disease patients can no longer move voluntarily when dopamine, a neurotransmitter that carries signals between nerve cells, is depleted.

Researchers studying neural interactions want to find out which neural components interact with each other, when they interact, and how they interact, i.e. stimulatively or repressively. Further, researchers are also interested in whether certain interactions are associated with state of health, for instance, medication, years of onset and severity of disease. Very commonly, experiments are performed at group level to discover features that are associated with a population, for example, stroke patients. Researchers are usually not only interested in each individual, but also and even more in a group of people. They want to answer the questions, “what are the interactions commonly shared by a population?” and “what are the interactions possessed only by a specific individual?”

Modern neuro-signal recording techniques provide us with the ability to sense neural impulses in living humans, and also encourage development of suitable mathematical analysis tools. Functional magnetic resonance imaging (fMRI) non-invasively measures
increase in oxygen in venous blood. Electromyography (EMG) detects the electrical impulses directly driving muscles. Positron emission tomography (PET) and electroencephalography (EEG) can also record the brain’s activities.

These neural signals share certain common features. First, they all are stochastic processes accompanied by physiological noise. Repeated experimental trials yield similar but not the same signals. Noise induced by respiration, heart beating, etc. is not simply white noise, but actually has certain patterns and structures. Second, experiments can generate long, multi-channel time series, but the number of subjects is usually relatively small because of the high cost of experiments or the limited availability of patients. Analysis methods should be adaptable to these features and also meet the concerns of biomedical researchers.

1.2 Current Methods

A wide range of mathematical methods, such as correlation threshold \( [9] \), linear decomposition \( [8, 30] \), multivariate auto-regression, structural equation models (SEM) \( [5] \), dynamic causal models \( [15] \) and clustering, have been employed to discover neural interactions. The correlation threshold (CT) method \( [9, 18, 23, 32, 44, 46] \) estimates how strongly two neural components interact with each other by calculating the correlation coefficient between their activities. If the correlation coefficient is so high that it is almost impossible to be just a result of randomness, then the two components are considered associated with each other. Backed by random field theory \( [9] \), CT is statistically rigorous and is able to control the Type-I error rate, and to estimate the strength of interactions. However, it only analyzes two components at a time, so it cannot distinguish whether the two components interact directly or indirectly through a third component \( [21] \).

Linear decomposition methods such as principal component analysis (PCA) and independent component analysis (ICA) \( [8, 30] \) assume that observed neural signals are a
1.2. Current Methods

combination of many simultaneous functional activities. The goal of decomposition is to find the underlying functional activities that correlate closely with the input stimuli, and to detect the neural components executing the functions. Linear decomposition is computationally efficient because it does not require a model-selection procedure. Its applications to sEMG data [30] and fMRI group analysis [8] have produced meaningful results. However, simulation showed that it is not sensitive to focal interactions [46].

Multivariate auto-regression (MAR) models [6, 19, 45], structural equation models (SEM) [5, 22, 29, 41, 49] and dynamic causal models (DCM) [14, 15, 39–41] are the most popular multivariate linear regressions used in discovering neural interactions. The three models share a common format and can be unified, as shown in Eq. (1.1):

\[ X(t) = \sum_{i=0}^{m} A_i X(t - i) + e(t), \]

where \( X(t) \) is a multi-channel time series, \( A_i \)s \((i = 0, \ldots, m)\) are regression matrices and \( e(t) \) is a noise term. A non-zero coefficient, \( A_i(j, k) \), implies that the \( j \)-th channel depends on the \( k \)-th channel with a lag of \( i \) time points, which suggests that neural component \( j \) interacts with another component \( k \). MAR, SEM and DCM emphasize different research purposes by setting different constraints on Eq. (1.1). MAR focuses only on lagged interactions, by setting the constraint \( A_0 = 0 \). Traditional SEM focuses only on simultaneous interactions, by setting the constraint \( A_i = 0 \) \((i = 1, \ldots, m)\). DCM is distinct from the other two in its consideration of modulated interactions [41] and modelling neural activities as hidden variables. These features possibly make DCM more closer to the reality. Multivariate linear regression models are commonly statistically rigorous, and are supported by many well-developed algorithms for various kinds of purposes, such as spectrum analysis, model selection, and statistical inference. They are also very flexible. For instance, modulated interactions can be modelled by adding bilinear terms, as in DCM. Dependence relationships can be visualized as graphs, to which further network
1.2. Current Methods

analysis can be applied. Nevertheless, these models usually demand much computation, especially when features such as latent variables or bilinear terms are involved.

Clustering techniques, such as self-organizing map (SOM) [13, 38], k-means clustering [2, 3, 12], hierarchical clustering [10] and graph clustering [11, 20, 28], provide a data-driven approach to exploring unknown neural interactions without assuming a predefined interaction pattern. The underlying idea of the clustering approach is that if neural components cluster together, they behave similarly, which suggests interactions among them. Clustering is usually implemented with fast and heuristic algorithms and thus is suitable for large-scale problems where it is difficult to perform rigorous statistical analysis. New interactions that are discovered can be verified later with rigorous statistical models. However, the data-driven feature also brings disadvantages. Certain algorithms either fall in local optimal solutions or their convergence cannot be proved. Statistical criteria such as specificity and sensitivity cannot be theoretically analyzed for clustering methods.

Bayesian networks, a type of graphical model, are suitable for discovering neural interactions due to their graphical nature and rigorous underlying theory. First, the structural similarity between Bayesian networks and the nervous systems makes the former promising tools for modelling the latter. The nervous system is a network of connected neurons that transmit electrochemical signals between each other through nerve fibres. The topology of this complicated system can be naturally abstracted as a graph, that is, nodes connected with edges. Second, the edges of a Bayesian network are directional, which is suitable for modeling the transmission path of neural signals. Third, the node variables of a Bayesian network only locally depend on their parent nodes, which is similar to neurons’ local and direct interactions with their neighbor neurons through nerve fibres. Fourth, Bayesian networks are modular and flexible. Given a network structure, different statistical models can be used to describe the dependence relationships between nodes and their parent nodes. Fifth, plenty of model-learning and computation methods have
been developed for Bayesian networks by researchers in the field of artificial intelligence.

1.3 Dynamic Bayesian Networks

A dynamic Bayesian network (DBN) is an extension of a Bayesian network (also called a belief network) for stochastic processes. Actually, the term “dynamic” does not mean that the model itself changes temporally, but that the network models a dynamic system. Also, “Bayesian” does not mean that Bayesian statistics are employed, but that Bayes’ rule is employed in inference. Though the term “temporal belief network” seems to be a more suitable name, the term “dynamic Bayesian network” has been more popular.

A Bayesian network employs a directed and acyclic graph (DAG) to encode conditional independence among random variables. The essential concept in the encoding is d-separation, which we will introduce after defining related concepts in graph theory. A DAG $G$ is a pair $(V, E)$ where $V$ is a set of vertices and $E \subseteq V \times V$ is a set of arrows without cycles. A chain between two vertices $\alpha$ and $\beta$ is a sequence $\alpha = \alpha_0, \ldots, \alpha_n = \beta$ of distinct vertices such that $(\alpha_{i-1}, \alpha_i)$ or $(\alpha_i, \alpha_{i+1}) \in E$ for all $i=1, \ldots, n$. Vertex $\beta$ is a descendant of vertex $\alpha$ if and only if there is a sequence $\alpha = \alpha_0, \ldots, \alpha_n = \beta$ of distinct vertices such that $(\alpha_{i-1}, \alpha_i) \in E$ for all $i=1, \ldots, n$. If three disjoint subsets $A$, $B$ and $S \subseteq V$ satisfy the condition that any chain $\pi$ between $\forall \alpha \in A$ and $\forall \beta \in B$ contains a vertex $\gamma \in \pi$ such that either

- arrows of $\pi$ do not meet head-to-head at $\gamma$ and $\gamma \in S$,
- arrows of $\pi$ meet head-to-head at $\gamma$ and $\gamma$ is neither in $S$ nor has any descendants in $S$,

then $S$ d-separates $A$ and $B$. Let $P$ denote a probability distribution of random variables $x$, and $x_\alpha$ denote the random variable represented by vertex $\alpha$ and $x_A = \{x_\alpha | \alpha \in A\}$. $P$ is said to admit the global Markov property according to a DAG $G$ if
1.3. Dynamic Bayesian Networks

Figure 1.1: An example of dynamic Bayesian networks. \( Z_t = [U_t, X_t, Y_t]^T \) is a first-order Markov process with dependence relationships specified as in the figure. \( X_t \) is a Markov process whose transition distribution \( P(X_t|X_{t-1}, U_t) \) varies according to the input \( U_t \). Arrows from \( X_{t-1} \) and \( U_t \) to \( X_t \) are associated with the transition distribution. \( Y_t \) is the output observation at time \( t \). The arrow from \( X_t \) to \( Y_t \) is associated with the distribution \( P(Y_t|X_t) \). Such a process can be represented by the first two time-slices circled by dots.

\[ S \text{ d-separates } A \text{ and } B \Rightarrow x_A \perp x_B|x_S, \text{ i.e. } x_A \text{ and } x_B \text{ are independent given } x_S. \]

The same set of conditional independence can be encoded by different DAGs, and a DAG can be converted to an essential graph that uniquely encodes the set of conditional independence [1]. If \( P \) obeys the global Markov property, it can be factorized according to \( G \) as

\[
f(x) = \prod_{\alpha \in V} f(x_\alpha|x_{\text{pa}(\alpha)}),
\]

where \( f(x) \) is the joint probability density function (pdf) of \( P \) and \( \text{pa}(\alpha) = \{\beta | (\beta, \alpha) \in E\} \) are the parents of \( \alpha \) [23]. In summary, a Bayesian network is composed of a DAG that encodes the conditional-independence relationships and a set of conditional probability distributions that specifies the joint distribution.

A multi-channel stochastic process can be modelled with a Bayesian network of \( C \times T \) vertices, where \( C \) is the number of channels and \( T \) is the number of time points and each
1.4 Applications of Bayesian Networks

A vertex represents the signal of a channel at a time point. In this case, the DAG is subject to an additional constraint, that vertices at time $t$ cannot have vertices after $t$ as their parents because the future cannot influence either the present or the past. If the same dependence relationships repeat time after time and the signals at $t$ only depend on the signals from $t - N$ to $t$, then the whole network can be “rolled up” as its DBN representation, a DAG composed of only vertices from $t - N$ to $t$ (see Fig. 1.1). In certain contexts, we call normal Bayesian networks “static Bayesian networks” to distinguish them from DBNs. Though the directed arcs of static Bayesian networks cannot be simply explained as causality, the arcs of DBNs from the past to the present can be interpreted as Granger causality [17]. Many classical multivariate probabilistic models, such as hidden Markov models, Kalman filters, and multivariate auto-regression models, are actually special cases of DBNs [36].

1.4 Applications of Bayesian Networks

Only a limited number of papers have been published since (D)BNs were introduced into the field of discovering neural interactions in around 2003. At the current stage, researchers are still exploring suitable techniques for using DBNs in this field. In this section, we chronologically review the development of (D)BNs for discovering neural interactions.

In 2003, Michell et al. proposed Naive Bayesian Networks (NBNs), a special case of static Bayesian networks with strong independence assumptions, to identify instantaneous cognitive states from fMRI data [34]. Because the structure of NBNs is predefined, they can be applied to huge networks, as the authors did to networks of about 2000 nodes. Statistical tests showed that NBNs were able to utilize the discriminating information contained in the fMRI data, but the authors did not compare NBNs with other classifiers to investigate whether the former exploited the discriminating information better or not.
### Table 1.1: Publications on Applications of Bayesian networks in Discovering Neural Interactions

<table>
<thead>
<tr>
<th>Paper</th>
<th>Year</th>
<th>Bayesian Network</th>
<th>Static or Dynamic</th>
<th>CPD</th>
<th>Number of Nodes</th>
<th>Constraint</th>
<th>Structure Search</th>
<th>Score Function</th>
<th>Search Algorithm</th>
<th>Final Result</th>
<th>Data Set</th>
<th>Sample Size</th>
<th>Signal</th>
<th>Number of Components</th>
<th>Group Model</th>
<th>Classification</th>
<th>Structure Analysis</th>
<th>Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>[34]</td>
<td>2003</td>
<td>S</td>
<td>D</td>
<td>Gau.</td>
<td>7-20,000</td>
<td>No</td>
<td>No</td>
<td>BDE</td>
<td>Greedy</td>
<td>Best</td>
<td>fMRI</td>
<td>10-33 subj.</td>
<td>fMRI</td>
<td>7-20,000</td>
<td>Pool</td>
<td>Pool</td>
<td>Pool</td>
<td>Biological</td>
</tr>
<tr>
<td>[47]</td>
<td>2005</td>
<td>D</td>
<td>Bin. &amp; Gau.</td>
<td>Gau.</td>
<td>10(5 hidden)</td>
<td>Bi</td>
<td>No</td>
<td>BIC</td>
<td>MCMC</td>
<td>Best</td>
<td>sEMG</td>
<td>6-28 subj.</td>
<td>fMRI</td>
<td>5</td>
<td>Pool</td>
<td>Pool</td>
<td>No</td>
<td>Simulation</td>
</tr>
<tr>
<td>[25]</td>
<td>2006</td>
<td>S</td>
<td></td>
<td>Gau.</td>
<td>7</td>
<td></td>
<td>No</td>
<td></td>
<td></td>
<td>Best</td>
<td>fMRI</td>
<td>12 subj.</td>
<td>fMRI</td>
<td>7</td>
<td></td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[27]</td>
<td>2007</td>
<td>D</td>
<td></td>
<td>Gau.</td>
<td>7</td>
<td>Yes</td>
<td>No</td>
<td></td>
<td></td>
<td>Best</td>
<td>fMRI</td>
<td>7</td>
<td>fMRI</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Although the purpose of the study was classification rather than discovering neural interactions, the authors did suggest that Bayesian networks are an applicable and promising tool for studying neural signals.

One year later, Burge et al. proposed DBNs for both detecting altered neural connections and classification, with an application to fMRI data [7]. Compared with Michell et al.’s NBNs, Burge et al.’s DBNs were much more complex: the structures of the DBNs were not predefined, but were learned from the data. Since the authors studied networks of a considerable number of nodes (about 150), they excluded simultaneous interactions, and focused only on lagging interactions, decomposing the model-selection problem to many simple local variate-selection problems. Though Gaussian NBNs, special cases of static Bayesian networks, outperformed DBNs in classification, hypothesis testing strongly suggested that temporal information should be exploited, which was consistent with previous research [35]. The authors also compared the DBNs of the case group against those of the control group but did not perform rigorous hypothesis testing in this comparison.

To the best of our knowledge, the most complicated Bayesian network used in this field is a DBN with hidden nodes [47] (2006) where the true neural activities were modelled as hidden nodes, and the dynamic structure between the hidden nodes was learned from the data with the structure expectation maximization algorithm. Because of the high computational complexity of this algorithm, the authors investigated a network of only 5 hidden nodes. However, the results were very promising. The DBNs not only outperformed k-nearest neighbor (KNN) classifiers but also intuitively discovered consistent interaction patterns across subjects.

Zheng et al.’s paper published in *NeuroImage* [48] (2006) supported Bayesian networks as a promising approach to discovering neural interactions. Though the method used in this study was simple, just static Gaussian Bayesian networks, the interactions discovered were interpreted neurologically and were backed by more than 15 citations.
In a simulation, Bayesian networks yielded higher likelihood than structural equation models (SEM) did. However, the study also found that when there were more than 13 nodes, the learned structure of Bayesian networks did not match the true structure well.

Li et al. explored group analysis based on Bayesian networks in papers [26] [27]. In [7], group analysis was performed by pooling or averaging all the data of the same experimental group, and then learning one Bayesian network from the pooled or averaged data as if they were sampled from the same subject. An alternative approach was explored in [26]: a Bayesian network was learned for each subject individually, and group analysis was performed by comparing two groups of Bayesian networks with the concept of “network motif” [33]. In [27], a trade-off between the two approaches was proposed. The Bayesian networks of each individual subject were constrained to share the same structure but allowed to have different parameters to accommodate inter-subject variability.

1.5 Thesis Outline

The research presented in this thesis focuses on adapting classical methods, or creating novel methods for dynamic Bayesian networks, to meet the particular need of neurological research and the limitations of current neural-signal acquisition technology. Since biomedical research usually has high and special requirements of interpretability, generality, and reliability, we mainly investigated the following topics: feature extraction from Bayesian networks, group analysis, and error control in learning network structures.

Chapter 2 describes our research on extracting structural features from Bayesian networks. To provide guidance on further experiment design and clinical diagnosis, mathematical models in biomedical research should not only fit experimental data well, but also be able to extract interpretable features from data. A framework is designed to discover statistically significant structural differences, such as the pattern of sub-networks,
between two groups of Bayesian networks. The framework includes three components: Bayesian-network modeling, statistical structure-comparison, and structure-based classification. The framework is demonstrated in an application to discovering the coordination patterns of muscle activities using surface electromyography.

Chapter 3 presents our research on group-analysis methods for dynamic Bayesian networks. A common and fundamental challenge in biomedical studies is to meaningfully combine population-common features and individual-specific features for group inference. Dynamic Bayesian networks have recently been introduced for modelling neural connectivity, but their group-analysis methods have not been systematically investigated. In this chapter, three popular group-analysis methods, i.e. the “virtual-typical-subject” approach, the “individual-structure” approach [16] and the “common-structure” approach [22 31], are compared in a study on Parkinson’s disease using fMRI, from the aspects of their statistical goodness-of-fit to the data, and more importantly their sensitivity in detecting the effect of medication on the disease. Three fundamental questions are investigated: “which approach most accurately reflects the underlying biomedical behavior?”, “do the approaches lead to considerably different analysis results?”, and “how can the suitable approach be selected?” This chapter ends with a summary of our observations in the comparison, including the limitations of the three methods, and as well vision for future research on group analysis.

Chapter 4, a highlight of the thesis, elaborates our research on error control in learning network structures. In real world applications, graphical statistical models, such as dynamic Bayesian networks, are not only a tool for operations such as classification or prediction, but usually the network structures of the models themselves are also of great interest (e.g. in modeling brain connectivity). The false discovery rate (FDR) [4 43], the expected ratio of falsely claimed connections to all those claimed, is often a reasonable error-rate criterion in these applications. Controlling the FDR provides researchers statistical confidence of the learned networks. However, current structure-learning algorithms
have not been adequately adapted to the concerns of the FDR. The traditional practice of controlling the type I error rate and the type II error rate under a conventional level does not necessarily keep the FDR low, especially in the case of sparse networks. In this chapter, a novel algorithm is designed to control the FDR of the network connections inferred from experimental data under user-specified levels. It is proved that the new algorithm is able to curb the FDR under user-specified levels (for example, conventionally 5%) at the limit of large sample size, and meanwhile recover all the true connections with probability one. We named this algorithm the $\text{PC}_{\text{fdr}}$ algorithm. This chapter includes detailed explanation about the algorithm and its heuristic modification, theoretical proofs of its asymptotic performance, analysis on its computational complexity, extensive evaluation with simulated data and real data, and discussions on its alternatives.

Based on the original version of the $\text{PC}_{\text{fdr}}$ algorithm, several modified or extended versions are developed. Hereafter, we refer to this family of algorithms as the “$\text{PC}_{\text{fdr}}$ algorithms”, and refer to the original version as the “$\text{PC}_{\text{fdr}}$ algorithm”.

Chapter 5 extends the control over the false discovery rate from static Bayesian networks to dynamic Bayesian networks. The $\text{PC}_{\text{fdr}}$ algorithms in Chapter 4 are only basic implementations of the main idea. In this chapter, two extensions are designed for dynamic Bayesian networks. One is an adaptation to prior knowledge, allowing users to specify which edges must appear in the network, which cannot, and which are to be learned from data. This extension is naturally applicable to dynamic Bayesian networks, by simply regarding them as Bayesian networks that cannot have edges from time $t+1$ to time $t$. The other extension is using the $\text{PC}_{\text{fdr}}$ algorithms to improve Bayesian inference of dynamic Bayesian networks. The idea is to first learn a network with the $\text{PC}_{\text{fdr}}$ algorithms, and then make Bayesian inference based on a prior distribution derived from the learned network. It accelerates Bayesian inference and is relatively robust to perturbing noise.

Chapter 6, as a supplement to Chapters 2 to 5, provides a fast method to select
1.5. Thesis Outline

node variables (e.g. brain regions-of-interest) for Bayesian-network modelling. Methods in Chapters 2 to 5 assume that the node variables of interest have been pre-defined, which however is not always true in exploring research. The method presented in Chapter 6, designed for fMRI study, uses extended linear discriminant analysis to preliminarily select brain regions of interest from thousands of candidate voxels. The activities of those selected brain regions can be analyzed further with Bayesian networks.

Chapter 7 briefly summarizes the results of the thesis, and provides concluding remarks, and discusses potential future work.

The thesis itself is written in a manuscript style as permitted by the Faculty of Graduate Studies at the University of British Columbia, with each chapter representing an independent research effort. Each chapter has its own motivation and literature review and can be read independently. For convenience, the references of each chapter are placed in an individual bibliography at the end of the chapter.
Bibliography


Chapter 1. Bibliography


Chapter 2

Extracting Structural Features from Bayesian Networks

with Applications to Discovering “Dependent Synergies” among Muscles

To provide guidance on further research exploration and clinical diagnosis, mathematical models in biomedical research should not only fit experimental data well, but also be able to extract interpretable features from data. In this chapter, a structural-feature extraction framework is developed to discover statistically significant structural differences, such as the patterns of sub-networks, between two groups of Bayesian networks. The framework includes three components: Bayesian-network modeling, statistical structure-comparison, and structure-based classification. It is demonstrated in a study on the coordination pattern of muscle activities using surface electromyography. It discovered several groups of muscles whose coordinated activities in reaching movement significantly differ between healthy people and stroke patients, and it also achieved very low classification error-rates.

2.1 Introduction

An important goal of motor control studies is to understand how the central nervous system (CNS) selects and co-ordinates the muscle activity patterns necessary to achieve a variety of natural motor behaviors [6]. A key emerging concept in motor control is the importance of synergies [6], or groups of muscles that act together. Works on frogs have suggested that a complex repertoire of movements can emerge from the appropriate control and selection of only a few synergies which each represent a primitive movement [33]. However, identifying muscle synergies from all possible muscle patterns and efficient decomposing of complex and variable motor behaviors into meaningful synergies remain challenging problems. To address this goal, a necessary intermediate step is to determine how muscles efficiently collaborate together during movements. In this paper, we plan to infer muscle interaction patterns from surface Electromyogram (sEMG) recordings during reaching movements. Especially we are interested in investigating whether certain muscle interactions are selectively recruited across subjects based on hand dominance or affliction by stroke.

A sEMG is a semi-stochastic signal whose properties depend upon a number of factors including the anatomical and physiological properties of the contracting muscles, the amount of subcutaneous fat, and choice of electrodes [21]. Standard analytical methods, including frequency-based ones, may be particularly sensitive to parameters difficult to measure, such as capacitive effects of muscles and subcutaneous tissue [31]. Nevertheless, despite its limitations, the non-invasive nature of sEMG makes it practical to record several muscles simultaneously in humans, and hence allows the investigation of synergies.

A sEMG signal can be modelled as a zero-mean wide-sense stationary stochastic process (i.e. the so-called carrier signal) modulated by the sEMG amplitude [5]. A common practice in the sEMG literature is to focus only on the amplitude data, i.e. doing rectifying and then low-pass filtering the raw sEMG signal, while the carrier data is generally ignored. Here we use a fundamentally different approach and model sEMG
2.1. Introduction

carrier signal after the effects of amplitude have been minimized.

During the last years, partially linear decomposition methods such as Principal Component Analysis (PCA), linear Independent Component Analysis (ICA) [19] [20], and Nonnegative Matrix Factorization (NMF) [32] have been suggested to infer synergistic action between muscles. For instance, linear ICA was applied to noisy sEMG data and revealed meaningful interactions between muscles [19]. These methods are characterized by a number of latent variables and project multi-channel sEMG signals to a subspace. They share a common assumption that a small set of source signals are upstream of the muscles and produce the sEMG signals. For example, ICA assumes that the observations are combinations of statistically independent components, and the goal of ICA is to find the underlying sources.

Here we propose a different approach, the Bayesian network (BN) modeling approach, to directly represent interactions between muscles without using latent variables. Methods such as P/ICA do not explicitly reveal interactions but implicitly through underlying sources. Our BN framework provides an alternative which captures the interactions directly from the observed sEMG signals by detecting conditional dependence/independence between muscle activities, i.e. whether the activities of two muscles are associated given that of a third muscle.

There are a number of biological and statistical reasons that make the assumption of conditional dependence between muscle activities plausible. Depending upon the neural context, the same neurons participating in central pattern generators can demonstrate remarkably dissimilar behaviors [28]. Activity in muscles themselves may be modulated by Ia inhibitory interneurons from antagonists or Renshaw cell activity in the spinal cord that may project to other motor neurons in the spinal cord. [3]. Ordinary coherence, often used to infer connectivity coupling between muscles, cannot distinguish whether two channels are directly connected or indirectly connected via other channels suggesting that conditional dependence may need to be considered [14]. In fact, failure to identify
potential conditional dependence between muscles may lead to erroneous interpretations regarding the overall interactions between muscles, necessitating the use of partial coherence, often resulting in a distinctly different connection pattern. A BN represents the conditional dependence/independence through a graph of nodes and edges connected according to rigorous statistical rules (see Sec. 2.2.2), so it is suitable to discover conditional interactions between muscles.

There are a number of reasons why BNs, as opposed to other choices of graphical models (such as Boolean networks), may be particularly well-suited for modeling muscle interaction networks. First, BN models have a solid basis in statistics, enabling them to deal with the stochastic and nonlinear aspects of sEMG measurements in a natural way. As a rigorous probabilistic model, a BN allows incorporation of the stochastic nature of sEMG recordings that may be caused by any number of biological factors along the cortex → spinal cord → peripheral nerve → neuromuscular junction → muscle pathway. Second, BN's modular nature makes it easily extensible to the task of modeling sub-networks of sEMG signals. In BNs, conditional probability distributions (CPD) are specified locally at each node to encode dependence relationships between a node and its parents, so the whole network can be decomposed into many small sub-networks. Since a node is independent of its ancestors given its parents, a simple conditional independence relationship between muscles could be that the interaction between muscle-A and muscle-B does not depend on the activity in muscle-C. Further, the rich repertoire of techniques developed for network analysis in other areas can be used for inferring muscle networks using sEMG. Finally, BNs can be used when incomplete knowledge is available, and can also deal with dynamical aspects of muscle interactions through generalizations like dynamical BNs.

Therefore, in this paper, we develop a BN modeling framework to statistically capture the interactions between muscle activity patterns directly. More specifically, we generalize the muscle synergy idea into the concept of a muscle network, defined as a set of muscle
activity patterns with *probabilistic* and *conditional* interactions between them that are coordinated to achieve specific motor behaviors. In our approach, we first model the overall muscle activity across several simultaneously recorded sEMG signals. From the learned muscle network we then define muscle synergies as statistically significant sub-networks or network-motifs \[23\] and use the term “dependent synergies” to refer to (conditionally) dependent muscle (not necessarily pair-wise) interactions.

We also plan to tackle the problem of investigating consistent muscle synergies across subjects within a certain group. While there are many potential factors that may affect muscle synergies and thus sEMG patterns during a reaching movement, here we focus on the effects of stroke and hand dominance. Hand dominance has been reported as an important factor in motor control \[26\], and different muscle activations between the dominant and non-dominant hands have recently been observed during reaching movements \[29\]. Though hand dominance as a factor in motor and functional performance has been studied in the literature, to the best of our knowledge, no studies have investigated the impact of hand dominance in healthy or stroke subjects in terms of muscle association/interaction patterns.

The main contributions of this paper are as follows:

- To present a framework for learning the *muscle interaction networks* during reaching movements based on the BN modeling of sEMG data.

- To demonstrate how the trained BNs can then be probed with network motif analysis to determine “dependent synergies”.

- To demonstrate that some network structure features are relatively robust across subjects, and thus can be used to distinguish factors such as handedness and stroke status.

- To demonstrate that specific three-muscle synergies may provide insights into the compensatory changes seen in reaching movements after stroke.
2.2 Methods

To indicate that the sEMG “carrier” signal (after the amplitude information is estimated and removed) can also be informative, even though it has been traditionally ignored by sEMG analysis methods.

The paper is organized as follows. In Section 2.2, we describe the proposed BN framework for learning muscle networks and analyzing the sub-network patterns. A real case study utilizing sEMG recordings from stroke and healthy subjects, including data from both dominant and non-dominant arms, is discussed in Section 2.3. Finally, we conclude our paper and suggest some directions for future research.

2.2 Methods

2.2.1 Framework and Components

Our Bayesian network (BN) framework includes three components: BN modelling, graph structure analysis and classification. First, BNs are applied to multi-muscle sEMG signals, with directed acyclic graphs (DAG) encoding the overall interactions between muscles. Researchers can choose different types of BNs according to their interest and prior knowledge about a specific application. For example, they can use a static BN to model the invariant interactions, a dynamic BN \[25\] to model the dynamics of muscles, or a BN with hidden nodes \[25\] to model the unobserved neural signals which drive the muscles.

Secondly, graph structure analysis is conducted on the learned BNs to extract structural features which characterize the interaction patterns among muscle activity patterns. Structural features can be the number of edges in or out from a node (which is called “degree” in graph theory), or the length of the shortest path from one node to another (which is called “distance” in graph theory). To go beyond the node and edge levels, but to a sub-network level, we employed the “network motif” concept \[23\].

Thirdly, classification is performed, based on the BNs. As statistical models, BNs can be naturally extended to statistical classifiers with the posterior probability criterion.
2.2. Methods

Figure 2.1: An example of representing the stochastic interactions of sEMG recordings of different muscles with a Bayesian Network. This particular graph is only for illustrative purposes and does not have intrinsic biological meaning. Each node represents a sEMG recording of a muscle and the whole graph represents the interaction among the muscles in a movement. This DAG encodes the independence relationships \( X_1 \perp X_2 \) (but \( X_1 \perp X_2 \mid X_3 \)) and \( (X_1, X_2) \perp X_4 \mid X_3 \) (but \( (X_1, X_2) \perp X_4 \)). Nodes \( X_1 \) and \( X_2 \) do not have any parent and they are associated with unconditional probability distributions. Nodes \( X_3 \) and \( X_4 \) have parent(s) and they are associated with conditional probability distributions. The joint probability distribution can be factorized according to the DAG as: 

\[
P(X) = P(X_1)P(X_2)P(X_3|X_1X_2)P(X_4|X_3).
\]

Alternatively, the DAGs of BNs can also be used as input features to other classifiers such as classification trees. In the following sub-sections, we will elaborate on each of the three components alluded to above.

2.2.2 Bayesian Networks

**Introduction** A Bayesian Network (BN) [16], also referred as a "Bayesian belief network" or simply "belief network", is a graphical model that consists of a directed acyclic graph (DAG) and a set of (conditional) probability distributions. The DAG encodes the (conditional) dependence/independence relationships among random variables, and the probability distributions constitute the joint probability distribution with Bayes’ rule. A BN, in short, is a representation of the joint distribution over random variables by indicating the conditional dependence/independence relationships with a DAG.
A DAG encodes a set of (conditional) independence relationships between node variables with the concept of d-separation [16]. For instance, let $X_1$, $X_2$ and $X_3$ denote three node variables. According to the global Markov property [16], if $X_1$ is d-separated from $X_2$ by $X_3$ in the DAG, then it is said that $X_1$ is conditionally independent of $X_2$ given $X_3$, i.e. $P(X_1X_2|X_3) = P(X_1|X_3)P(X_2|X_3)$, and we denote it as $X_1\perp X_2|X_3$. The definition of conditional independence is similar to that of unconditional independence $P(X_1X_2) = P(X_1)P(X_2)$ except that it is conditional on a third random variable $X_3$. Here we focus on two simplified but important corollaries within the broader concept of d-separation. (1) If $X_1$ and $X_2$ are connected (i.e. there is a path in the DAG from $X_1$ to $X_2$ or vice versa), then $X_1$ is not independent of $X_2$ which we denote as $X_1\nmid X_2$. (2) If $X_1$ precedes $X_2$ (i.e. there is at least a path from $X_1$ to $X_2$) and $X_3$ blocks all the paths from $X_1$ to $X_2$, then $X_1\perp X_2|X_3$. Fig. 2.1 shows an example of BN which represents a network of four muscles. (Note that this figure is only for illustrative purposes and does not necessarily represent any real muscle network.) According to the above corollaries, this DAG encodes a set of relationships: $X_1\perp X_2$ (but $X_1\nmid X_2|X_3$) and $(X_1, X_2)\perp X_4|X_3$ (but $(X_1, X_2)\nmid X_4$). This example also shows that conditional independence does not imply unconditional independence, and vice versa.

If the joint probability distribution of a set of random variables $X$ is subject to the Markov property, it can be factorized according to the DAG as

$$P(X) = \prod_{\text{pa}[X_i] \neq \emptyset} P(X_i|\text{pa}[X_i], \theta_i) \prod_{\text{pa}[X_i] = \emptyset} P(X_i|\theta_i), \quad (2.1)$$

where the set $\theta = \{\theta_1 \ldots \theta_n\}$ denotes the parameters used in the probability distributions and $\text{pa}[X_i]$ denotes parents of $X_i$, i.e. nodes with an edge to $X_i$. If a node $X_i$ has parent nodes, i.e. $\text{pa}[X_i] \neq \phi$, it is associated with a conditional probability distribution $P(X_i|\text{pa}[X_i])$. If a node $X_i$ does not have any parent, it is associated with an unconditional probability distribution $P(X_i)$. For example, the joint probability of
2.2. Methods

the four-muscle network in Fig. 2.1 generally can be decomposed as Eq. 2.2 according to the chain rule, and further as Eq. 2.3 according to the (conditional) independence relationships specified by the DAG.

\[
P(X) = \prod_{i=1}^{4} P(X_i|X_1, \ldots, X_{i-1})
\]

\[
= P(X_1)P(X_2)P(X_3|X_1X_2)P(X_4|X_3).
\]

Using BNs to represent muscle interactions can be summarized as follows. sEMG signals are regarded as a vector-valued stochastic process \(X(t)=[X_1(t), X_2(t), \ldots X_n(t)]^T\) where \(n\) denotes the number of muscles and \(X_i(t)\) the observed signal of the \(i\)th muscle at time \(t\). A DAG with nodes \(X = \{X_1, \ldots, X_n\}\) indicates the interactions between muscles. If a node \(X_i\) is connected to another node \(X_j\), then the corresponding muscles are considered to interact. If \(X_i\) is d-separated from \(X_j\) by another node \(X_k\), then the muscles represented by \(X_i\) and \(X_j\) do not interact conditionally on the activity of the muscle \(X_k\). Conditional and unconditional probability distributions are associated with the nodes, describing how the muscle activity patterns interact with each other. In this study, we employed Gaussian BNs to model the multi-muscle sEMG signals, i.e. we modelled each variable \(X_i\) as the sum of a Gaussian noise and a linear combination of its parents \(pa[X_i]\). Gaussian BNs are not only applicable but also one of the most popular BNs for modeling multi-channel continuous variables.

It should be pointed out that before DAGs are analyzed to reveal muscle interactions, it must be converted to an essential graph (EG) which is also referred as a completed acyclic partially directed graph (CPDAG) in the literature. This is because there are often several different DAGs representing the same set of conditional independence relationships, while there is only one EG uniquely encoding the set of conditional independence relationships, as shown in Fig. 2.2 An EG has the same edges as a DAG does except that some edges are not directed. Algorithms to convert a DAG to an EG have been
2.2. Methods

Figure 2.2: An example of essential graphs (EGs). Both the directed and acyclic graphs DAG 1 and DAG 2 represent the same set of conditional independence: $A \perp (C, D)$, $B \perp D | C$ and $B \perp D | (A, C)$. Since the edge between $C$ and $D$ is reversible, its direction is removed in the EG. Directions in either the DAGs or the EG do not necessarily imply causality.

A directed graph, such as those that contain directed edges of either a BN or an EG to encode conditional independence/dependence and independence/dependence, does not necessarily imply causality but rather association. Hence our model does not conflict with the fact that the activities of multiple muscles are often coupled by kinematics and dynamics of the bones and joints. For further details on BNs, EGs and d-separation, the reader is referred to Lauritzen’s book (1996) \cite{16} and Andersson’s paper (1997) \cite{1}.

**Learning Bayesian networks** Learning a BN includes two steps: (1) structure learning and (2) parameter learning. Structure learning is to select an appropriate DAG among many candidate DAGs. Parameter learning is to estimate the parameters of the conditional and unconditional distributions given the DAG. In structure learning, we attempted to select the most probable DAG based on the observations according to the maximum a posteriori (MAP) criterion. Let $X$ denote the observations and $S$ the DAG, the best structure from the view of Bayesian statistics is:

$$\hat{S} = \arg \max_S p(S | X),$$  \hspace{1cm} (2.4)
2.2. Methods

where according to Bayes’ rule, we have

\[
p(S|X) = \frac{p(X|S)p(S)}{p(X)}, \quad (2.5)
\]

\[
p(X|S) = \int p(X|\theta, S)p(\theta|S)d\theta, \quad (2.6)
\]

where \(p(S)\) is the prior probability of the structure \(S\) and \(p(\theta|S)\) is the probability of the parameter \(\theta\) given the structure \(S\). The MAP criterion has the advantage of allowing users to incorporate their knowledge in the prior probability. As the denominator in Eq. (2.5) does not depend on \(S\), only the numerator needs to be maximized. If \(p(S)\) (the prior probability) is uniform over all the possible structures, only \(p(X|S)\) in Eq. (2.6) (the conditional probability) needs to be maximized. The uniform assumption is reasonable in practice since we do not prefer any structure before we observe the data. However, alternates to a uniform prior distribution may be considered to enhance computationally efficiency [9].

The MAP approach can be implemented by selecting the structure with the largest Bayesian Information Criterion (BIC) score [30] which is defined as

\[
\text{BIC}(S) = \sup_{\theta} \log P(X|S, \theta) - 0.5K \log N, \quad (2.7)
\]

where \(N\) denotes the sample size of \(X\) and \(K\) denotes the number of free parameters in \(\theta\). In the comparison between two models \(S_1\) and \(S_2\), \(\exp[\text{BIC}(S_1) - \text{BIC}(S_2)]\) asymptotically approximates the ratio of their posterior probability \(P(S_1|X)/P(S_2|X)\) if the two models \(S_1\) and \(S_2\) have the same prior probability, i.e. \(P(S_1)=P(S_2)\) and \(p(\theta|S)\) is uniform [30]. (For rigorous proof, please refer to Schwarz’s paper in 1978.) The large sample size in our sEMG study, 1000 time points (see Sec. 2.3.1), should satisfy the condition of the asymptotical approximation. As shown in Eq. (2.7), the BIC consists of two terms: the maximum log likelihood term \(\sup \log P(X|S)\) and the penalty term \(-0.5K \log N.\)
The penalty term prevents “over fitting”, i.e. choosing a structure which has too many edges compared with the data size. Because a structure with more edges tends to have larger likelihood, we will inevitably choose a fully connected DAG if only the maximum likelihood criterion is used. Therefore, a penalty term is needed in model selection. The BIC penalty term is proportional to the number of free parameters \( (K) \), and “punishes” structures with redundant edges.

After structure learning, we estimated the parameters of the conditional and unconditional distributions via the maximum likelihood criterion. Since the number of all the possible DAGs is super-exponential to the number of nodes, it is impractical to exhaustively search for the best DAG. To avoid local maxima found by greedy algorithms, we employed the Markov chain Monte Carlo algorithm (MCMC) \[13\] to learn the structure.

Our implementation of learning BNs was developed based on the software Bayes Net Toolbox (BNT) \[24\] for Matlab.

### 2.2.3 Sub-network Patterns and Muscle Synergies

Milo and Shen-Orr \[23\] reported that in the networks of the real world (for instance, gene regulation networks), certain connection patterns of sub-networks appear more frequently than would be expected from chance, and these patterns (named “network motifs” by the authors) can be used to characterize the networks. Since in the current case, sub-networks represent the co-activation between muscles, we adopt a similar idea to discover the muscle synergies from the DAGs of BNs. However, in contrast to the original proposal by Milo and Shen-Orr’s, where the goal was to determine network motifs that appear more frequently in real graphs than in randomized graphs, our goal is to identify network motifs that distinguish a group of graphs from another group. Specifically, the question of interest is that “given two groups of muscle interaction graphs derived from sEMG recordings (e.g. recordings of stroke and normal subjects), is it possible to determine which network motif(s) distinguish one group from the other?”
2.2. Methods

Figure 2.3: Different ways that three muscles can interact within a BN framework. The directed triple graph is first converted to an undirected graph, and then it is classified as one of the patterns. The four patterns are abbreviated as B, L, V and T respectively.

We propose the following way to detect network motifs distinguishing between two groups of graphs. First, the occurrences of each possible connection pattern in each graph is counted. As a result, the count of a particular connection pattern in one group of graphs is a group of numbers. Then, the two groups of numbers are compared with a hypothesis test such as a t-test. Finally, patterns appearing significantly more frequently in one group than in the other are selected as the feature network motifs of that group. We note that patterns which are functionally important but not statistically significant could exist and could be missed by this approach.

In our sEMG study, we focus on triplet network motifs, i.e. sub-networks with three nodes. Though sub-networks with more nodes can be analyzed similarly without theoretic difficulty, we did not pursue more than three in this exploratory research due to limited computation power and the observation that triplets have demonstrated our framework adequately. The complete and detailed procedure of detecting and evaluating triplets from two groups of graphs is as follows. First, DAGs are converted to EGs because a EG uniquely determines the dependence relationships among nodes (see Sec. 2.2.2). Secondly, the appearing frequencies of triplet connection patterns are counted. The possible triplet patterns are show in Fig. 2.3. Thirdly, t-test is performed to evaluate whether a specific connection pattern appears significantly more often in one group of EGs than in another. As a result, each triplet pattern is associated with a level of significance, or p-value. Forth, the p-values are adjusted for the effect of multiple testings with Sidak correction.

34
2.2. Methods

as in Eq. (2.8),

\[ p_a = 1 - (1 - p)^h, \]  

(2.8)

where \( p \) is the original \( p \)-value, \( p_a \) is the adjusted one and \( h \) is the number of hypotheses tested simultaneously. In the context, \( h \) equals the number of the interested connection patterns. The effect of multiple testings can also be adjusted with the false discovery rate (FDR) [2] which controls \( q \)-values, i.e. the expected portion of falsely rejected hypotheses among those rejected. Finally, connection patterns with the adjusted \( p \)-values or \( q \)-values lower than 0.05 are selected as network motifs.

While the above general network motifs provide information on the overall connectivity patterns of the network, we are also interested in specific muscle triplets because it is possible that alterations between the interactions of a few particular muscles may significantly influence classification of reaching movements between groups. The identification procedure is similar to that of the general network motifs, except that the connection patterns are counted for each combination of three specific muscles individually. For \( n \) muscles, all the \( C_n^3 \) triplets are exhaustively examined. Instead of a \( t \)-test, Fisher’s exact test is employed to check whether a connection pattern of a specific muscle triple appears significantly more in a group of graphs than in another. The effect of multiple testing is also adjusted, but the number of simultaneous hypothesis tests \( h \) is much larger. If there are \( n \) muscles and \( m \) patterns of interest, the number \( h \) is \( mC_n^3 \).

2.2.4 Classification

BNs can be used for classification purpose in two ways: (1) as statistical models, they can be extended to be statistical classifiers; (2) as graphical models, their structures can be input as features to other classifiers.

BNs can be extended to be a statistical classifier naturally with the posterior probability criterion. Suppose \( M_1 \) and \( M_2 \) are the statistical models of the sEMG signals of two
2.2. Methods

groups of subjects respectively, e.g. a control group and a stroke group. Given sEMG signals $X$, its model index can be predicted by the posterior probability criterion as in Eq. (2.9). The more $p(M_1|X)$ is larger than $p(M_2|X)$, the more likely that $X$ belongs to group 1, and vice versa. If the prior probabilities $p(M_1)$ and $p(M_2)$ are equal, the ratio $p(M_1|X)/p(M_2|X)$ is the same as the ratio $p(X|M_1)/p(X|M_2)$ according to Bayes’ rule.

Suppose there are totally $N_i$ subjects of group $i$ and $M_{ij}$ represents the BN model of the $j$th subject in group $i$. With the assumption that each individual model of group $i$ is equally representative of the group, the group model $M_i$ ($i=1,2$) can be built by averaging the BN models trained from individual subjects within the same group, as expressed in Eq. (2.10).

$$\text{If } \frac{p(M_1|X)}{p(M_2|X)} \begin{cases} > 1, \text{ then } X \text{ belongs to group 1}, \\ < 1, \text{ then } X \text{ belongs to group 2}; \end{cases}$$

(2.9)

$$p(X|M_i) = \frac{\sum_{j=1}^{N_i} p(X|M_{ij})}{N_i}, i = 1, 2.$$  

(2.10)

Structure features of BNs can also be used as the input to other classifiers in various ways. As we mentioned in Sec. 2.2.2, DAGs should be converted to EGs before used to represent the interactions between muscles. An EG of $n$ nodes can be encoded as an $n$-by-$n$ binary adjacent matrix $A = \{a_{ij}\}$ where $a_{ij} = 1$ indicates an edge from node $i$ to node $j$. Because an EG cannot have any edge which circles from and to the same node, the diagonal elements are all zeros, and they are uninformative. The elements off the diagonal line can be lined up as a binary vector with $n(n-1)$ elements and then used as input to a classification tree. We choose a classification tree, but not other classifiers such as the support vector machine (SVM) [34] because of two reasons. First, classification tree is especially suitable for categorical data as an adjacent matrix is; secondly and also most importantly, it is easier to interpret since a classification tree explicitly gives the conditions of predicting the class. In contrast, despite its popularity, SVM results are hard to interpret, as the SVM algorithm implicitly maps features to a high-dimensional
imaginary space. Triple patterns derived from a BN’s structure can also be used as input features to a classification tree. A graph of \( n \) nodes contains \( C_n^3 \) triples which can then be converted to a categorical vector of \( C_n^3 \) elements. In this study, \( n \) equals 7, resulting in 42-dimensional classification features for an EG and 35-dimensional features for exploring triple patterns.

The performance of BN-based classifiers were evaluated with both cross-subject validation and within-subject validation on a real sEMG data set containing repeated trials of arm reaching movements. In cross-subject validation (Fig. 2.4), all the trials of one arm side of one subject were kept aside as the testing data, and all the other data were used to train a classifier which was then used to predict the stroke state and hand dominance of the testing arm side. The stroke state and hand dominance of each testing trial was predicted individually, and then all the predictions voted on the state of the testing arm side. In this way, all the trials of the arm being tested were used to predict its group membership. This procedure is repeated for each subject in a leave-one-out, cross-validation manner. Cross-subject validation evaluates whether data of the same group share common features while data of different groups have distinguishing features. The strategy of within-subject validation is shown in Fig. 2.5. One trial of a subject was kept aside as the testing data, and all the other trials of the same subject were used to train a classifier which was then used to predict the group membership of the testing
trial. This procedure was repeated and each time a different trial was selected as the testing trial. Within-subject validation was used to evaluate whether different trials of one arm from the same subject were consistent, yet trials of the other arm from the same subject were different.

The performance of the BN-based classification trees applied to sEMG carrier was compared with that of a PCA-based SVM applied to the sEMG amplitude. The PCA-based SVM approach includes two steps: 1) reduce the dimension of sEMG amplitude with PCA. 2) input the dimension-reduced data to SVMs for classification. In our study, 6 principal components (PC) were needed to explain 80% of the total variance. The PC coefficients of the seven muscles in our study were then concatenated as input feature vectors to SVMs whose length was $6 \times 7 = 42$. We constrained both classification trees and SVMs from using no more than three elements of the input features to avoid over-fitting. We tried to input all the 35 or 42 features to the classifiers, but the corresponding classification error rates were as high as 30% to 50%, so we set the constraint above to improve the performance. All the combinations of no more than three features were exhaustively searched, and finally the best performance was selected. The results showed that this modified implementation provided much better performance than using all the features together. We think that the comparisons between the BN-based classification trees and the PCA-based SVMs are fair because: they were subject to the same constraint; the best combination of features were enumeratively searched; and the classification trees were not provided with more features than the SVMs were.

### 2.2.5 Modeling sEMG Signals

A sEMG signal $y(t)$ is usually considered as a zero-mean, Gaussian, band-limited and wide-sense stationary stochastic process $x(t)$ modulated by the EMG amplitude $a(t)$, expressed as

$$y(t) = x(t)a(t),$$  \hspace{1cm} (2.11)
2.2. Methods

where $t$ indicates time and $x(t)$ is named by us as “carrier”, a term borrowed from the field of communication. Accepting these assumptions and being consistent with what we observed in this study, we assume

1. $x(t)$ is wide-sense stationary;

2. $x(t)$ follows a Gaussian process;

3. $x(t)$ is approximately white;

4. $x(t)$ is ergodic.

Fig. 2.6, a typical auto-correlation plot of $x(t)$, shows that $x(t)$ is approximately white, *i.e.* $x(t_1)$ and $x(t_2)$ are approximately independent if $t_1 \neq t_2$. Fig. 2.7, a typical histogram of the distribution of $x(t)$, shows that $x(t)$’s distribution is almost Gaussian in the time-domain. Since the distribution of $x(t)$ is suggested to be Gaussian both spatially and temporally, the ergodicity assumption at least is not severely violated yet not rigorously proved.

The four assumptions as a whole imply that a single-channel “carrier” signal $x(t)$ is independent identically-distributed (iid) at different time points and the distribution is
2.3. Results

well approximated as Gaussian. Therefore, we can model multi-channel “carrier” signals with static Gaussian BNs by adding one more assumption that the joint distribution of the multi-channel signals follows a multivariate Gaussian distribution. Since static models are used here, what we attempt to discover is not the dynamics of the muscles’ activities, but the invariant interaction patterns among the muscles during the reaching movements.

Most of the existing literature on sEMG describes rectification and low-pass filtering of the data and hence emphasizes the amplitude $a(t)$ while $x(t)$ the carrier is generally abandoned. However, in contrast here we focus on the “carrier” signals but not the amplitude, which will provide novel insights into the underlying system. As supported by our analysis results reported in Sec. 2.3, the “carrier” signal is also informative and provides a robust way to deal with the challenging issue of inter-subject variability in sEMG data.

2.3 Results

2.3.1 Real sEMG Datasets

All research was approved by the University of British Columbia Ethics Board. Thirteen stroke subjects and 9 healthy subjects were recruited. In the experiment, subjects sat in a chair with their hands on the thigh, and then reached to a shoulder-height target as fast as they could for five to ten trials with each arm. The sEMG of the following seven muscles were collected: the deltoids (anterior and lateral), the triceps (long and lateral heads), the biceps brachium, the latissimus dorsi, and the brachioradialis. A bipolar montage was used to minimize the effect of crosstalk. The seven-channel sEMG signals were amplified, high-pass filtered at 20 Hz to reduce movement artifact, and then sampled at 600 Hz. (Please refer to [18] for further details on the sEMG experiment procedure). The amplitude of the sEMG was estimated with root-mean-square (RMS),
with a moving window of 0.1 second. As EMG signal can be considered as a wide sense stationary stochastic process modulated by the EMG amplitude \[5\], the carrier stochastic process (see Sec. 2.2.5) was also calculated by dividing the sEMG signal by the estimated amplitude. Finally, the sEMG signals (both the amplitude and carrier) of different trials were resampled with cubic spline interpolation so that the overall movement duration from reaching-start to target-touching was exactly 1000 time points. This prevents the sub-network analysis (see Sec. 2.2.3) from being biased by the unequal data length since a DAG learned from more time points tend to include more connections than another learned from less time points. Preliminary studies included testing the effect of the moving window by stepwise increasing the width from 20 ms to 300 ms. It was determined visually that 100 ms gave the best estimation, as it yielded good amplitude estimate and produced an approximately wide-sense stationary carrier signal.

Since we were interested in the influence on sEMG patterns of two factors, stroke condition and hand dominance, sEMG recordings were grouped into four types of experimental groups: healthy dominant hand (HD), healthy non-dominant hand (HN), stroke more affected side involving the dominant hand (SD) and stroke more affected side involving the non-dominant hand (SN). To sharpen the contrast between the stroke and healthy states, the less affected side of stroke subjects was excluded because it may not be a valid comparison against the healthy state. Most individuals with stroke have subtle deficits on the non-paretic side due to a number of factors, including the contribution of the small portion of corticospinal tracts that do not decussate, and remain ipsilateral. To focus on the effect of one factor at a time, we fixed the state of one factor and compared the two states of the other factor in four group comparisons: HD v.s. HN, SD v.s. SN, HN v.s. SN and HD v.s. SD.
2.3. Results

A. Deltoi
Biceps
Tri. Long
Tri. Lat.
L. Deltoi
Brachior
Lat.

(a) Healthy, Dominant
(b) Stroke, Non-dominant

Figure 2.8: Examples of typical DAGs of subjects with different hand dominance and different stroke state.

2.3.2 Learned Bayesian Networks

Examples of the learned DAGs of the BNs are given in Fig. 2.8, where the left side illustrates a typical DAG of the dominant hand side of a healthy subject (i.e. a HD case) and the right is for a typical DAG of the non-dominant hand side of a stroke subject (i.e. a SN case). The two DAGs showed different connection features. For example, the lateral deltoid is connected to all the other six muscles in the DAG of the HD case while it is completely isolated in the SN case. The long head of the triceps is more connected with others in the HD case than in the SN case. The biceps is connected to the lateral, long heads of the triceps and the lateral deltoid in the HD case while to the anterior deltoid, latissimus dorsi, the brachioradialis and the lateral triceps in the SN case. These differences between the two typical subjects suggest that both stroke and hand dominance can affect the sEMG muscle association patterns during reaching movements.

Comparisons between the DAGs of different hand dominance are shown in Fig. 2.9. The width of the displayed edge is proportional to the log odds ratio (LOR) of their appearance rates in the two groups. First the DAGs were converted to EGs, then the
Figure 2.9: Mean differences of overall network structures as a function of experimental condition. The labels under the sub-figures are the experimental conditions in comparison. A solid edge means it appears more frequently in the first type than in the second, and a dashed edge does vice versa. The width of an edge is proportional to the contrast of the frequencies of its appearances. The contrast are measured with log odds ratio (LOR) which is defined as $\text{LOR} = \ln\left(\frac{f_1/(1 - f_1)}{f_2/(1 - f_2)}\right)$ where $f_1$ and $f_2$ are the appearance frequencies of the edge in two types of experimental situations under contrast. If the edge appears $k$ times in $n$ trials, the appearance frequency $f$ is estimated as $k + 1/(n + 2)$ with the Bayes estimator. The Bayes estimator is more robust than MLE when $k$ and $n$ are small, and converges to MLE when $k$ and $n$ are large. Only edges whose absolute value of the LOR is greater than $\ln(2)$ are shown in the figure.
2.3. Results

The appearance rate of each edge in each group of the EGs was calculated, and finally the rate was converted to LOR. Only the edges whose LOR’s absolute value exceeded \( \ln(2) \) were shown in the figure. A solid edge means that it appears more frequently in the first group than in the second, and a dashed edge means a higher appearing frequency in the second group. Sub-figures (a) and (b) compared the BNs learned from dominant hand and non-dominant hand groups. We note that networks from dominant hand groups have more connections between the muscle pairs (biceps, triceps long head), (lateral triceps, lateral deltoid) and (anterior deltoid, lateral deltoid). Sub-figures (c) and (d) compared the BNs between healthy subjects and stroke subjects. We note that healthy subject group has more connections between the muscle pairs (triceps long head, biceps) and (brachioradialis, lateral deltoid).

### 2.3.3 Triple Patterns

The learned BNs from different experimental groups demonstrated different triple connection patterns, as shown in Fig. 2.10. The V pattern (see Fig. 2.3) appears significantly more frequently in the HN group than in the SN group \( (p = 0.0002) \), and more often in the SD group than in the SN group. The Line pattern (see Fig. 2.3) appears significantly more frequently in the SN group than in the HN group \( (p = 0.0103) \), and also in the SN group than in the SD group \( (p = 0.0077) \). In addition to the general triple patterns, specific muscles triples also showed significantly different connection patterns across experiment groups. Muscles involved in these triplets are deltoid (both anterior and lateral), triceps and brachioradialis, as shown in Table 2.1. However, no significant results were discovered about the Blank pattern and the Triangle pattern (see Fig. 2.3).

Since the V pattern is the most efficient pattern to connect three muscles and a Line pattern only connects two muscles, muscles of the HN and SD groups seem cooperate more closely than those of the SN group. This observation coincides with clinic experience, where the SN group typically have the most difficulty in performing reaching
2.3. Results

Figure 2.10: Comparison of the count of appearances of connection patterns in different experimental groups. The distribution of the number of appearances is shown with box plot. The boxes have lines at the lower quantile, median and the higher quantile. The whiskers are the extent of the rest of the data and the plus symbols are the outliers. If the notches of two boxes overlap, their medians differ significantly with type I error rate less than 5%. The means of the distributions are also compared with t-test and significant results are labeled with arrows. *p*-values are adjusted for multiple comparisons with Sidak correction. Since the notches compare the medians and the t-test compares the means, their results may differ from each other when the distributions are skewed, for example in the comparison of pattern “Line” between HN and SN. There are no statistical differences in the number of edges between the above groups, and thus the statistically different results between groups are not based on the number of edges.
### Results

Comparison | Muscle Triplet | Pattern | Count | OR and 95% CI | $p$-value | $q$-value |
--- | --- | --- | --- | --- | --- | --- |
HD vs. SD | A. Deltoid, L. Deltoid and Tri. Lat. | L | 13/45 vs. 0/41 | inf, (3.5052, inf) | 0.0151 | 0.0076 |
| | L. Deltoid, Tri. Lat. and Tri. Long | L | 13/45 vs. 0/41 | inf, (3.5052, inf) | 0.0151 | 0.0076 |
HN vs. SN | Brachior, L. Deltoid and Tri. Lat. | L | 10/45 vs. 34/55 | 0.1765, (0.0650, 0.4638) | 0.0140 | 0.0141 |
SD vs. SN | A. Deltoid, Brachior and Tri. Long | B | 24/41 vs. 9/55 | 7.2157, (2.5614, 21.0100) | 0.0036 | 0.0036 |
| | A. Deltoid, L. Deltoid and Tri. Lat. | L | 0/41 vs. 14/55 | 0.0000, (0.0000, 0.3312) | 0.0390 | 0.0157 |

Table 2.1: Muscle coordination patterns significantly different among subjects. The connection patterns of the specific muscle triples appears significantly more/less frequently in a type of experimental group than in another type. Counts are in the form of (No. of appearance / No. of trials). OR and CI are short for “odds ratio” and “confidence interval”. $p$-values are originally calculated with Fisher’s exact test and are then adjusted for multiple comparisons with the Sidak correction or converted to $q$-values with the FDR [2].
movements. Lack of normal cooperation between the muscles may explain this empirical observation of the SN group’s demonstrating inferior performance \[12\]. The importance of the deltoid (both anterior and lateral) in these results is consistent with a previous traditional analysis of these data \[18\] where the deltoids’ activation was found significantly altered after stroke. The functional connectivity between the brachioradialis and the deltoid that we detected (Fig. 2.9 c-d) during reaching movements has been suggested by previous studies. Lemon et al. \[17\] used transcranial magnetic stimulation during reaching movements in human subjects and found evidence of a strong cortical drive to both the deltoid and brachioradialis throughout a reaching movement. The connectivity between the deltoid and triceps found more prominently in the stroke subjects (Fig. 2.9 c-d) may suggest a more traditional stroke synergy, where there is breakdown in the normal independent activation of muscles involving the shoulder girdle and those involved in movement of the elbow \[10\]. Probably because the reaching task is sufficiently simple that healthy subjects mastered it easily even with their non-dominant hands, no significant difference between HD and HN groups was found. An alternative explanation is that handedness may not be strongly contrasted in individuals that exhibit forms of ambidexterity.

### 2.3.4 Classification Performance

The across-subject classification performances were reported in Table 2.2. The proposed methods, which use the structure features of the BNs learned from the carrier signals, provide very high classification accuracy in the four classification tasks, and outperform the PCA-based SVM approach applied to the amplitude signals. This excellent classification performance is unlikely related to over-fitting because the error rate was estimated with cross-subject validation and the classifiers were provided with almost equal chances to achieve good performance. Although most of current studies on sEMG focus on the amplitude signals for classification, our classification trees are based on the “carrier”
### 2.3. Results

<table>
<thead>
<tr>
<th>Signal Feature Classifier</th>
<th>Car. EG</th>
<th>Car. Triple EG</th>
<th>Amp. PCs SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>HD vs. HN</td>
<td>0.0556</td>
<td>0.1111</td>
<td>0.1667</td>
</tr>
<tr>
<td>SD vs. SN</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0769</td>
</tr>
<tr>
<td>HN vs. SN</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.1250</td>
</tr>
<tr>
<td>HD vs. SD</td>
<td>0.1333</td>
<td>0.0000</td>
<td>0.0667</td>
</tr>
</tbody>
</table>

Table 2.2: The error rates of cross-subject classification. “Car.” and “Amp.” are short for the carrier and the amplitude respectively. “EG” and “Triple” are the essential graphs and the triple patterns of the carrier’s Bayesian networks. “PC” is short for principal component. “CT” and “SVM” are short for classification trees and support vector machines respectively. All the error rates were estimated with the cross-subject validation procedure in Fig. 2.4. Bold numbers are the best performance of the four classifiers.

The signal (see Sec. 2.2.5) which is usually lost in the traditional process of rectification, smoothing and other preprocessing steps. In addition to almost perfect classification involving stroke, the proposed methods also offer visible interpretations.

The BN-based classification trees generally provided better performance when the triple patterns were used as the input classification features than when the adjacency matrices of EGs were used. We think this is because an EG just encodes pair-wise interactions but a triple involves three muscles. Figs. 2.11 and 2.12 showed the best classification trees for the four classification tasks. These trees include interactions between agonist-antagonist pairs (e.g. long head of triceps, biceps), muscles with similar actions (e.g. biceps, brachioradialis), and muscles with no obvious similarity of function but might be part of larger synergies (e.g. brachioradialis and latissimus dorsi).

As previously mentioned in Sec. 2.2.4 in addition to using their structure features, BNs by themselves can be extended straightforwardly to a statistical classifier. The BN statistical classifier separated the trials of the same subject perfectly (i.e. the within-subject cross-validation error rate = 0%) but performed poorly in cross-subject validation (i.e. the error rate ≈ 50%). Fig. 2.13 demonstrated the usage of a BN classifier for classifying healthy subjects’ dominant and non-dominant hands. Although the BN classifier showed high trial-to-trial reliability, its across-subject classification performance
2.3. Results

(a) Healthy, Dominant vs. Non-dominant  
(b) Stroke, Dominant vs. Non-dominant

(c) Non-dominant side, Healthy vs. Stroke  
(d) Dominant side, Healthy vs. Stroke

Figure 2.11: The best classification trees by using EG structures as the classification features. If two trees have equal cross-subject classification error rates, the one using fewer input edges is chosen. For each node, a label specifies its group membership and its predicted value (in bold). For a branch, its label is the decision rule. For the performances of these trees, please refer to Table 2.2.
2.3. Results

(a) Healthy, Dominant vs. Non-dominant
(b) Stroke, Dominant vs. Non-dominant
(c) Non-dominant side, Healthy vs. Stroke
(d) Dominant side, Healthy vs. Stroke

Figure 2.12: The best classification trees by using triple patterns as the classification features. If two trees have equal cross-subject classification error rates, the one using fewer input edges is chosen. For each node, a label specifies its group membership and its predicted value (in bold). For a branch, its label is the decision rule. For the performances of these trees, please refer to Table 2.2.
2.3. Results

Figure 2.13: Principal components of the log likelihood of BNs trained from the sEMG data of the dominant (HD) and non-dominant (HN) arms of healthy subjects. As demonstrated in Eqs. (2.9) and (2.10), a statistical BN classifier is a function of $p(x|M)$ where $x$ is the data of a trial and $M$ is the BN model of another trial. Thus, a trial is represented as a log likelihood vector composed of $p(x|M_i)$ where $M_i$s are the models of many trials. To visualize the high dimension vectors, we plot their first two principal components. Note that trials of the same arm tend to cluster together, which suggests that the BN represents the reaching movement reliably and consistently. Trials of HD and HN are not separated, which suggests that arms of the same type do not share a common distinguishing pattern in their log likelihood, but each has a different pattern. Since a statistical BN classifier is based on log-likelihood, its cross-subject performance is poor while its within-subject performance is excellent.
2.4. Conclusions and Discussions

was poor. We believe that the poor across-subject performance of the BN statistical classifiers is not due to deficiencies in the method, but rather reflective of the underlying biology. We note that the likelihood function is very consistent across trials of the same subject, which suggests that it is robust to various artifacts that may corrupt sEMG signals. Nevertheless, there may be considerable variations between individuals due to factors such as variance in genetics, developmental environment, compensatory strategies, and ongoing plasticity in response to environmental stressors. Thus a key result of the present work is implication that most within-group, inter-subject variability is not in the network structure (which itself is sensitive to handedness and effects of stroke), but rather in the parameters that specify the interactions within the network structure.

2.4 Conclusions and Discussions

In this paper, we have developed a Bayesian network (BN) framework based for modeling muscle networks to represent muscle co-ordinations in motor control. To demonstrate the benefits of the proposed approach, we applied this method to multi-channel sEMG data simultaneously recorded during reaching movements in healthy and stroke subjects. We noted that dependent muscle synergies can be revealed by first using a BN to model the muscle interaction network and then analyzing subnets of the derived BN. Further examination of the muscle synergies (muscle association patterns) suggested that stroke may particularly affect the interaction between a few specific muscles, especially the deltoid, during reaching movements. Classification trees based on the BNs’ structure features can effectively classify the reaching performed by the healthy dominant, healthy non-dominant, stroke dominant and stroke non-dominant arms across subjects. Classification trees provide an additional benefit: their classification processes can be easily visualized.

A key result of this study was that statistically significant differences between the
sEMG recorded under different conditions were noticed from analyzing the “carrier” signal, which is obtained by estimating and removing the amplitude information from the raw sEMG data. This is noteworthy observation, since in many sEMG studies, only the amplitude is investigated with the carrier signal discarded. Although not specifically explored in this study, we believe that the widespread statistical dependencies between the “carrier” of the sEMG signals from different muscles may reflect widespread synchronization between different cortical areas and muscles known to exist during dynamic movements \[8\][20]. We are intrigued that the significant features of the networks (Fig. 2.9 c-d) suggest a significant statistical interaction between the deltoid and brachioradialis, consistent with previously-described cortical-muscle interactions during reaching movements in normal subjects \[17\].

Another promising result from this study is the finding that the structure of BNs and their subsets are quite robust across individuals within the same group, yet demonstrate enough sensitivity to detect handedness and the effects of stroke. One of the fundamental challenges in classification of movements after stroke is how to deal with the inherent subject-to-subject variability in sEMG recordings, yet still be sensitive enough to detect impairment. The overall BN models demonstrated robustness to trial-to-trial variability within subjects (Fig. 2.13), but were quite different across subjects within the same group. However, the success of the classification trees based on the structure features of the subject-specific BNs suggests that the BN structure is sensitive to the effects due to stroke or handedness factor, but robust to inter-subject variability.

Although we suggest that our results demonstrate strong evidence to support the use of BNs as a tool to study sEMG signals, there are nevertheless shortcomings of the proposed method. The BN model, as proposed here, assumes stationarity of the muscle interactions. Yet there is evidence that muscle interactions may be dynamically affected by a number of factors. For example, different heads of the gastrocnemius muscle may be activated during walking as a function of activity in sensory afferents \[7\].
2.4. Conclusions and Discussions

an observation with neuroanatomical basis \[15\]. Presumably integrating the time-varying amplitude information with a dynamic Bayesian network (DBN) in addition to the carrier information for classification of sEMG data may be a fruitful avenue to explore in the future.

Each sEMG channel is a measure of the depolarization of muscle fibers, which is the end result of motor cortex activity, conduction along peripheral nerves, propagation across the neuromuscular junction, and propagation within the muscles. At each stage of motor propagation, there is the possibility for temporal variability, for example, “neuromotor noise” in the central nervous system which may be particularly important in disease states, conduction velocity along the peripheral nerves, which is a strong function of temperature, jitter at the neuromuscular junction and possible disease states in the muscles themselves \[22, 27\]. However, successfully incorporating these features in the future will additionally involve explicitly modeling the dynamic temporal relationships between sEMG recordings.

We do not suggest that the model is a biologically-accurate description of the generative process of EMG activity. Nevertheless, we note that both Renshaw cells and Ia inhibitory interneurons have complex effects on the firing of \(\alpha\) motor neurons in the spinal cord. Since these modulatory cells are themselves activated by muscle activities \[28\], the “dependent synergies” proposed – where interactions between muscles are influenced by the activity of other muscle(s) – are physiological plausible. However, explicitly incorporating inhibitory interneurons would require the addition of hidden nodes, beyond the scope of the current proposed method.

Acknowledgment

This research was partially funded by a National Parkinsons Foundation (NPF) Center of Excellence Grant and a Natural Sciences and Engineering Research Council of Canada
(NSERC) Grant.
Bibliography


[12] JE Harris and JJ Eng. Individuals with the dominant hand affected following stroke demonstrate less impairment than those with the non-dominant hand affected. *Neurorehabilitation and Neural Repair*, in press.


Chapter 3

Comparing Group-Analysis Methods Based on Bayesian Networks

with Applications in Modelling Brain Connectivity

To make analysis results generally applicable to a population, rather than just a specific individual, experimental data should be studied at group level. In a study on Parkinson’s disease using functional magnetic resonance imaging, three popular group-analysis methods, i.e. the “virtual-typical-subject” approach, the “individual-structure” approach, and the “common-structure” approach, were compared from the aspects of their statistical goodness-of-fit to the data, and more importantly their sensitivity in detecting the effect of medication on the disease. They led to considerably different group-level results, learning different network structures, and detecting different numbers of connections normalized by the medication. The “virtual-typical-subject” approach fitted the data of the healthy people best, while the “individual-structure” approach fitted the data of the patients best. The “individual-structure” approach was more sensitive in detecting the normalizing effect of the medication on brain connectivity, but it also tended to yield results supporting its assumption even when the assumption actually is incorrect.

3.1 Introduction

Effective brain connectivity, defined as the neural influence that one brain region exerts over another [4, 6], is important for the assessment of normal brain function, and its impairment is associated with neurodegenerative diseases such as Alzheimer’s or Parkinson’s disease (PD). Various mathematical methods, such as structural equation modelling (SEM) [18], multivariate auto-regressive modelling (MAR) [8], dynamic causal modelling (DCM) [5], bilinear dynamical system (BDS) [24], and Bayesian networks (BN) [27, 38], have been proposed for inferring effective connectivity from functional magnetic resonance imaging (fMRI). These models can all be visualized as a graph whose nodes denote brain regions and directed edges denote connections between brain regions. Connection parameters are associated with edges, indicating the strength of the connections. MAR, SEM and BNs are ordinary graphical models, but with different constraints on the network structures, regarding whether time lags are considered and whether cycles are allowed on the graph. DCM and BDS add a hidden layer to the graph for the unobserved underlying neural activities, and also allow the interaction between two regions to be modulated by the activities of a third region, at the cost of intensive computation.

FMRI experiments are usually performed to infer brain activations consistently shared by a population or to identify its differences between populations. Therefore, it is important to develop group-analysis methods for the aforementioned graphical models if they are to be more fully adopted in group analysis rather than just being used at the individual level [27]. Here we investigate potential group-analysis approaches, using BNs as a prototype of graphical models. We are especially interested in BN modelling of brain connectivity because it is flexible to handle both categorical and continuous data and as well both linear and nonlinear relationships [13], and also because plenty of methods about model learning and computation have been developed for BNs in the field of artificial intelligence.

Group analyses of any sort fundamentally involve two factors: common features
shared by group members, and the specific features of each individual. The common features seen within a group are typically of ultimate interest, although individually-specific features cannot simply be ignored, as they may indicate the presence of outliers, potentially biasing the overall group result. Many studies have found that fMRI activation patterns may considerably differ among subjects even within the same group. For example, Sugihara et al. [33] found three of the five regions related to writing were inconsistently activated among subjects; in Vandenbroucke et al. [36]'s study, eighteen of twenty nine subjects showed activation varied in both location and size. These results suggest that the similarity and diversity among group members should be balanced in group models, rather than either over-emphasized or neglected.

There have been several group-analysis techniques implicitly employed in graphical modelling of fMRI data, based on models such as SEM, MAR and BNs. A review of the literature reveals that these techniques can be divided into three broad categories, namely the “virtual-typical-subject” (VTS) approach, the “individual-structure” (IS) approach and the “common-structure” (CS) approach. The VTS approach assumes that every subject within a group performs the same function with exactly the same connectivity network, and it does not generally accommodate inter-subject variability. These methods reconstruct a virtual typical subject to represent the whole group, by pooling or averaging group data as if they were sampled from a single virtual subject [7, 27, 38], and then learn the connectivity network of the virtual typical subject. When subjects are homogeneous within a group or the inter-subject variability follows certain regular distributions such as the Gaussian distribution, the VTS approach could increase sensitivity, because pooling can yield a relatively large data set, and averaging can enhance the signal-to-noise ratio.

At the other extreme, the “individual-structure” (IS) approach learns a network for each subject separately, and then performs group analysis on the individually-learned networks [7, 15]. The IS approach is consistent with the concept of functional degeneracy, i.e. “the ability of elements that are structurally different to perform the same function
3.1. Introduction

... or yield the same output” [3], or more plainly, “there are multiple ways of completing the same task” [26]. The IS approach certainly considers inter-subject variability, but may not integrate group data tightly enough to enable correct inference about statistically significant differences between groups.

The “common-structure” (CS) approach is a trade-off between the two extremes, imposing the same network structure on the statistical graphical models of every subject, while allowing the parameters of the models to differ across subjects [12, 19]. The CS approach assumes that cognitive functions invoke a similar connectivity pattern for every subject, but the exact details of the connectivity patterns, in terms of connectivity strength (coefficients), differ across subjects. Thus, the CS approach addresses group similarity at the structural level and inter-subject variability at the parameter level.

We investigate the three approaches using data collected from subjects with Parkinson Disease (PD), as the consistent dramatic effects of L-dopa medication seen in this population provide an additional qualitative “ground truth” to the effects of group inference. One of the cardinal features of PD is bradykinesia, a slowness of performed voluntary movements, that represents a major source of disability in PD and is related to impairments in daily activities such as walking and writing. Impaired ramping of force may be fundamentally related to the clinical feature of bradykinesia [31]. Similarly, several studies have demonstrated disturbances of rhythmic movement in PD [25] that are likely related to bradykinesia.

However, the bradykinesia is thus far not fully explained in research. A recent study used PET imaging to infer brain areas that appear related to bradykinesia [35], such as under-activity in the sensorimotor cortex contralateral to the moving arm, bilateral dorsal premotor cortices, and ipsilateral cerebellum. However the PET modality did not enable inference about the interaction between these regions. A local problem in the basal ganglia circuit in Parkinson’s disease may cause disruption of downstream distributed motor control networks. A greater understanding of the relative contribution...
of neural regions to bradykinesia in healthy controls and patients may be of diagnostic and therapeutic significance for patients with Parkinson’s disease.

The main reason that we chose PD as an example for assessing group analyses, is that the standard medication used for this condition, L-dopa, has dramatic effects against bradykinesia and rigidity (although less effect against tremor, balance and gait). The effects of L-dopa on idiopathic disease are sufficiently dramatic that lack of response to this medication makes the diagnosis of Parkinson’s disease questionable \[10\]. Thus after introduction of the L-dopa medication, we would expect group features of analyses from PD subjects performing a paradigm assessing bradykinesia to approach that of normal subjects.

In this paper, we investigate the performances of the three group-analysis approaches (VTS, IS, CS) by applying them to an fMRI study on Parkinson’s disease. Broadly speaking, we attempted to answer three fundamental questions: “which approach most accurately reflects the underlying biomedical behavior?”, “do the approaches lead to considerably different analysis results?”, and “how can the suitable approach be selected?” The three approaches are compared from the aspects of their statistical goodness-of-fit to the data, and more importantly their sensitivity in detecting the effect of L-dopa medication on the disease. To the best of our knowledge, this is the first study specifically devoted to group-analysis on fMRI with BN modelling.

3.2 Materials and Methods

3.2.1 fMRI Data

The fMRI data were collected from ten healthy people and ten Parkinson’s disease (PD) patients, each of whom was asked to squeeze a rubber bulb at three different speeds or at a constant force, as cued by visual instruction. The patients performed the entire task twice, once before and another after the L-dopa medication which is most effective against
3.2. Materials and Methods

Slowness of movement and rigidity. Six regions were selected for the analysis according to the Talairach atlas: the left and right primary motor cortex (M1) (Brodman Area 4), supplementary motor cortex (SMA) (Brodman Area 6), and lateral cerebellar hemispheres (CER). FMRI time courses were collected at the sampling frequency of 0.503Hz for 4 minutes and 18 seconds, in total of 130 time points.

Subjects: The study was approved by the University of British Columbia Ethics Board. Subjects gave written informed consent prior to participating. Ten volunteers with clinically diagnosed PD participated in the study (4 men, 6 women, mean age 66±8 years, 8 right-handed, 2 left-handed). All patients had mild to moderate PD (Hoehn and Yahr stage 2-3) with mean symptom duration of 5.8±3 years. Exclusion criteria included atypical Parkinsonism, presence of other neurological or psychiatric conditions and use of antidepressants, sleeping tablets, or dopamine blocking agents. All patients were taking L-dopa with an average daily dose of 685±231 mg. We also recruited ten healthy, age-matched control subjects without active neurological disorders (3 men, 7 women, mean age 57.4±14 years, 9 right-handed, 1 left-handed).

All patients stopped their anti-Parkinson medications overnight for a minimum of 12 hours before the study. The mean Unified Parkinson Disease Rating Scale (UPDRS) motor score during this off-L-dopastate was 26±8. There were no significant correlations between UPDRS motor scores and age. All patients exhibited some aspects of bradykinesia on examination.

After completing the experiment in an off-medication state, patients were given the equivalent to their usual morning dose of L-dopa in immediate release form (mean 125±35.3 mg L-dopa). They then repeated the same tasks post-medication following an interval of approximately 1 hour to allow time for L-dopa to reach peak dose.

Experimental task: Subjects were instructed to lie on their back in the magnetic resonance scanner viewing a computer screen via a projection mirror system. All subjects
3.2. Materials and Methods

Figure 3.1: Graph of the first half of the computer task subjects had to perform. Subjects were either asked to statically squeeze at 10% maximum voluntary contraction (MVC), or squeeze sinusoidally (between 5 and 15% of MVC) at differing frequencies in 20s blocks. All frequencies were performed twice per session.

used an in-house designed response device in their left hand, which was a custom-built MR-compatible rubber squeeze-bulb connected to a pressure transducer outside the scanner room. They lay with their forearm resting down in a stable position, and were instructed to squeeze the bulb using an isometric hand grip and to keep their grip constant throughout the study. Each subject had their maximum voluntary contraction (MVC) measured at the start of the experiment and all subsequent movements were scaled to this, so that they had to squeeze at 5-15% of maximal force to accomplish the task. Using the squeeze bulb, subjects were required to control the width of an inflatable ring (shown as a black horizontal bar on the screen) in order to keep the ring within an undulating pathway without scraping the sides. Applying greater pressure to the bulb increased the width of the bar, and releasing pressure from the bulb decreased the width of the bar. To avoid scraping the sides of the tunnel, the required pressure was between 5 and 15% MVC. No additional visual feedback or error reporting was given when subjects went outside the white target lines so subjects had to monitor their own performance carefully. The pathway used a block design with sinusoidal sections in three different frequencies (0.25, 0.5 and 0.75 Hz) in a pseudo-random order, and straight parts in between where the subjects had to keep a constant force of 10% of MVC. The frequencies were chosen.
3.2. Materials and Methods

based on prior findings and pilot studies were used to determine that PD patients could comfortably perform the required task. Each block lasted 19.85 seconds (exactly 10 TR intervals), alternating a sinusoid, constant force, sinusoid and so on (Fig. 3.1) to a total of 4 minutes and 18 seconds.

PD patients performed the task both after an overnight withdrawal (minimum of 12 hours since last dose of L-dopa) of their anti-Parkinson drugs and repeated the same series one hour after admission of L-dopa. Before the first scanning session, subjects practiced the task at each frequency until errors stabilized and they were familiar with the task requirements. Custom Matlab software (Mathworks) and the Psychtoolbox was used to design and present stimuli, and to collect behavioral data from the response devices.

**FMRI acquisition:** Functional MRI was conducted on a Philips Achieva 3.0 T scanner (Philips, Best, the Netherlands) equipped with a head-coil. We collected echo-planar (EPI) T2*-weighted images with blood oxygenation level-dependent (BOLD) contrast. Scanning parameters were: repetition time 1985 ms, echo time 3.7, flip angle 90, field of view (FOV) [240.00 240.00 240.00] mm, matrix size = 128×128, voxel size 1.9 mm×1.9 mm× slice thickness. Each functional run lasted 4 minutes and 18 seconds. Thirty-six axial slices of 3 mm thickness were collected in each volume, with a gap thickness of 1 mm. We selected slices to cover the dorsal surface of the brain and include the cerebellum ventrally. A high resolution, 3-dimensional T1-weighted image consisting of 170 axial slices was acquired of the whole brain to facilitate anatomical localization of activation for each subject. Head motion was minimized by a foam pillow placed around the subjects head within the coil. Subjects also used ear plugs to minimize the noise of the scanner. The subjects constantly viewed visual stimuli on a screen through a mirror built into the head coil.
3.2. Materials and Methods

Preprocessing: The functional MRI data were preprocessed for each subject, using Brain Voyager trilinear interpolation for 3D motion correction and Sinc interpolation for slice timing correction. No temporal or spatial smoothing was performed on the data. The data were then further motion corrected with Motion Corrected Independent Component Analysis (MCICA), a computationally expensive but highly accurate method for motion correction [16, 17]. The Brain Extraction Tool in MRIcro [28] (http://www.sph.sc.edu/comd/rorden/mricro.html) was used to strip the skull off the anatomical and first functional image from each run, to enable a more accurate alignment of the functional and anatomical scans. Custom scripts in Amira software (Amira 3D Visualization and Volume modelling) were used to co-register the anatomical and functional images. The following ROIs were drawn separately in each hemisphere, based upon anatomical landmarks and guided by the Talairach atlas [34]: primary motor cortex (M1) (Brodman Area 4), supplementary motor cortex (SMA) (Brodman Area 6), and lateral cerebellar hemispheres (CER). The labels on the segmented anatomical scans were resliced at the fMRI resolution. The raw time courses of the voxels within each ROI were averaged as the overall activity of the ROI, and the averaged time course were then detrended and normalized to unit variance before input into the BN models.

3.2.2 Dynamic Bayesian Networks

A dynamic Bayesian network (DBN) [21] is a graphical model for stochastic processes. The term “dynamic” does not mean that the model itself changes over time but that the Bayesian network (BN) models a dynamic system. As an extension of BN, a DBN follows the same rules as a regular BN does, encoding the conditional-independence/dependence relationships among random variables with a directed acyclic graph (DAG). If a random variable $x_a$ directly depends on another random variable $x_b$, i.e. $x_a$ still depends on $x_b$ even given all the other random variables, then the dependence is encoded as an edge
3.2. Materials and Methods

Figure 3.2: An example of dynamic Bayesian networks. This DBN is a first-order Kalman filter process. Arrows from $X_{t-1}$ and $U_t$ to $X_t$ ($t=2, 3, \ldots$) mean that $X_t$ depends on $U_t$ and $X_{t-1}$, and are associated with the transition distribution $f(X_t|X_{t-1}, U_t)$ which varies according to the input $U_t$. Arrows from $X_t$ to $Y_t$ ($t=1, 2, \ldots$) mean that the output $Y_t$ depends on $X_t$, and is associated with $f(Y_t|X_t)$. As the same dependence relationships repeat, the process can be represented by just the first two time slices circled by dots. The joint probability density function of the process is $f(X, Y, U) = f(U_1)f(X_1|U_1)f(Y_1|X_1)\prod_{t=2}^{T} f(U_t)f(X_t|X_{t-1}U_t)f(Y_t|X_t)$ where $T$ is the number of time points.

between nodes $a$ and $b$ in the DAG (see Fig. 3.2 for an example). In this text, nodes $a$ and $b$ are associated with $x_a$ and $x_b$, respectively. When fMRI signals of ROIs are modeled with BNs, an edge in the DAG implies that two ROIs interact with each other even after removing the influence from all the other ROIs. Regarding the Bayes rule, the joint probability (density) function of all the random variables can be factorized according to the DAG as:

$$f(x) = \prod_{a \in W} f(x_a|x_{pa[a]}), \quad (3.1)$$

where $W$ denotes the set of all the nodes and $pa[a]$ denotes the set of node $a$’s parent node(s).

---

3BNs graphically encode conditional independence theoretically according to Markov properties: if two sets of nodes $A$ and $B$ are d-separated by a third set of nodes $C$ according to the DAG, and the three sets of nodes are disjoint, then $x_A$ and $x_B$ are conditionally independent given $x_C$, i.e.

$$P(x_A x_B|x_C) = P(x_A|x_C)P(x_B|x_C).$$

The d-separation is a complex concept. For its exact definition, please refer to [14].
3.2. Materials and Methods

A multi-channel stochastic process can be modelled with a BN of $C \times T$ nodes, where $C$ denotes the number of channels, $T$ denotes the number of time points, and each node represents the signal of a channel at one time point. Because the future can influence neither the presence nor the past, nodes at time $t + 1$ can have only nodes after time $t$ as their children. BNs that account for all time points will become intractably large, so first-order Markov and stationary assumptions are usually applied. In this case, the same dependence relationships repeat time after time and signals at $t$ only depend on signals from $t - 1$ to $t$, so the whole network can be “rolled up” as its DBN representation, a DAG composed of only nodes from $t - 1$ to $t$, as Fig. 3.2 illustrates.

We modelled fMRI signals with first-order Gaussian DBNs by regarding the signals of ROIs as a multi-channel stochastic Gaussian process. Gaussian DBNs whose conditional probability distributions are all Gaussian are limited to modelling linear relationships between random variables, while discrete DBNs are capable of modelling non-linear relationships, but at the cost of precision and accuracy. Gaussian DBNs whose parameters explicitly indicate the strength of interactions between ROIs also facilitate the cross-group quantitative comparison. To accommodate nonlinear responses of brain activities to the task frequencies, we represented the bulb-squeezing frequency level (see Section 3.2.1) as an extra categorical input node $L$ in the network. If node $L$ is a parent of node $a$, $x_a$ is regressed to other parent variables conditionally on the frequency level $x_L$, as in Eq. (3.2)

$$x_a = \sum_{b \in \text{pa}[a] \setminus \{L\}} \beta_{ba}|L x_b + \epsilon_{a|L},$$

(3.2)

where $x_a$ and $x_b$ are the signals of node $a$ and $b$, respectively, $\beta_{ba}|L$ is the connection coefficient from node $b$ to node $a$ conditional on the frequency level $x_L = l$, and $\epsilon_{a|L}$ which follows a zero-mean Gaussian distribution is the regression error conditional on $x_L = l$. If node $L$ is not a parent node of $a$, $\beta_{ba}|L$ takes the same value at all the four frequency levels. If there is no arrow from $b$ to $a$, the connection coefficients are set to zero, as $\beta_{ba}|L = 0$. 


3.2. Materials and Methods

We imposed on the DBN structures a constraint that there must be at least one path from the input node \( L \) to any ROI node. The constraint was imposed to ensure that the response of the ROIs to the bulb-squeezing task was accommodated. It is worth mentioning that in preliminary studies, we had relaxed the constraint to that there must be one path from the input node to at least one ROI, but all the structures learned under the relaxed constraint satisfied the more rigorous one.

3.2.3 Learning DBNs

Because many different DBNs may fit the same data almost equally well, we mixed them together according to their posterior probabilities, to provide a more stable approximation to the data than a single best DBN does. The mixture formula is 

\[
M^* = \sum_i P(M_i | X) M_i
\]

where \( M^* \) is the mixed model, \( M_i \) is a possible DBN and \( P(M_i | X) \) is its posterior probability given data \( X \). We used its mixed structure and its mixed coefficients to profile the mixture model which was composed of thousands of DBNs. The mixed structure is encoded as a matrix \( G^* = \{ g_{ab}^* \} \) whose element \( g_{ab}^* \) is the posterior probability of the existence of the connection from \( a \) to \( b \), defined as Eq. (3.3)

\[
g_{ab}^* = \sum_i P(M_i | X) g_{ab}(i), \tag{3.3}
\]

where the binary variable \( g_{ab}(i) \) indicates whether the connection from \( a \) to \( b \) appears in model \( M_i \) or not. The mixed coefficients \( B^* = \{ \beta_{ab|l}^* \} \) are the posterior expected values of the connection coefficients, defined as Eq. (3.4)

\[
\beta_{ab|l}^* = \sum_i P(M_i | X) \beta_{ab|l}(i), \tag{3.4}
\]

where \( \beta_{ab|l}(i) \) is the coefficient of model \( M_i \).

The mixed structure and the mixed coefficients were estimated with the Bayesian-information-criterion (BIC) scores [30] and Markov chain Monte Carlo (MCMC) [20].
The BIC score is defined as Eq. \((3.5)\) where \(N\) denotes the sample size of data \(X\) and \(K\) denotes the number of free parameters \(\theta\) of DBN model \(M\).

\[
\text{BIC}(M|X) = \sup_{\theta} \ln P(X|M, \theta) - 0.5K \ln N \tag{3.5}
\]

If two models \(M_1\) and \(M_2\) have the same prior probability, \(i.e. P(M_1) = P(M_2)\) and the prior distributions of their parameters \(\theta\) are uniform distributions, then the ratio of their posterior probabilities can be asymptotically approximated as \(\exp[\text{BIC}(M_1) - \text{BIC}(M_2)]\).  

\(4\) In our implementation, 1500 different DBN structures were sampled with MCMC, according to their relative posterior probabilities, \(i.e. \exp[\text{BIC}(M_i|X)]\) which was calculated when \(M_i\) was sampled, and then all the sampled DBNs were averaged according to their appearance frequencies in MCMC, as the estimation of the mixture model \(M^*\).

Based on the assumption that subjects perform the task with different patterns of brain connectivity, the IS approach learns a mixture DBN model individually for each subject. Thus each subject is associated with its own \(G^*\) and \(B^*\). The group mean structure (GMS) \(\bar{G}\) and the group mean coefficients (GMC) \(\bar{B}\) are the mean of \(G^*\) and the mean of \(B^*\) over all the subjects in the same group, respectively. The group-level BIC score of applying a combination of subject-specific DBNs \(\mathcal{M} = \{M_1, \cdots, M_S\}\) (where \(S\) is the number of subjects) respectively to the data of each subject is:

\[
\text{BIC}(\mathcal{M}|X_1, \cdots, X_S) = \sum_{s=1}^{S} \text{BIC}(M_s|X_s) = \sum_{s=1}^{S} \left[ \sup_{\theta_s} \ln P(X_s|M_s, \theta_s) - 0.5K_s \ln N_s \right], \tag{3.6}
\]

where all the notations are similar to those in Eq. \((3.5)\) besides the subscript \(s\) indicating that they are associated with subject \(s\).

Based on the assumption that subjects perform the task with the same pattern of

\(4\)For rigorous proof, please refer to Schwarz [30].
3.2. Materials and Methods

brain connectivity but differently in the details of the interaction between brain regions, the CS approach applies DBN models with the same network structure to the data of every subject in the same group, but optimizes the connection coefficients individually for each subject. Every subject is associated with the same $G^*$, and each subject is associated with its own $B^*$. The GMS $\bar{G}$ equals the $G^*$ and the GMC $\bar{B}$ is the mean of $B^*$ over all the subjects in the same group. The group-level BIC score of applying a DBN $M$ to the group data in the manner of the CS approach is the sum of its BIC scores over the data of each subject:

$$\text{BIC}(M|X_1, \cdots, X_S) = \sum_{s=1}^{S} \left[ \sup_{\theta_s} \ln P(X_s|M, \theta_s) - 0.5K \ln N_s \right], \quad (3.7)$$

where $K$ is without the subscript $s$ because DBNs with the same structure have the same number of parameters.

Based on the assumption that inter-subject variability is so small that they can be neglected, the VTS approach pools together all subject’s data, and learns a mixture model for the whole group. Every subject is associated with the same $G^*$ and $B^*$, and they are the GMS $\bar{G}$ and the GMC $\bar{B}$ as well, respectively. Because the data are pooled together, the group-level BIC score of a model $M$ is

$$\text{BIC}(M|X_1, \cdots, X_S) = \sup_{\theta} \left[ \sum_{s=1}^{S} \ln P(X_s|M, \theta) \right] - 0.5K \ln \left( \sum_{s=1}^{S} N_s \right) \quad (3.8)$$

3.2.4 Comparison of Three Approaches

The IS, VTS and CS approaches were compared from the aspects of the goodness of fit, the similarity between their models at the group level and the sensitivity to the effect of the L-dopa medication. First, their best group-level BIC scores found in the MCMC sampling were compared. As a criterion of model selection, the greater the BIC score is,
3.2. Materials and Methods

the better the model fits the data (see Section 3.2.3). More parameters do not guarantee a model a larger BIC score because the penalty term $0.5K \ln N$ in Eq. (3.5) punishes models with redundant parameters. Group-level BIC scores of the IS approach, the CS approach and the VTS approaches are defined in Eqs. (3.6), (3.7) and (3.8), respectively.

To examine the feasibility of using the best BIC scores for selecting the IS, VTS and CS approaches, we carried out simulation studies to investigate the following question: whether the true underlying generating models can be correctly identified with the highest BIC scores. In our simulation, similar to the setting of our real fMRI data set, fMRI time courses of the 6 ROIs at 130 time points were simulated for 10 subjects from mixture DBNs derived from the real fMRI data set. We leaned mixture DBN models from the real fMRI data of a group of 10 subjects (e.g. PD patients before medication) with the IS, VTS and CS approaches respectively. Then the learned mixture models were used as true models of the IS, VTS and CS approaches respectively to generate the simulation data.

Different approaches may fit the data with different BIC scores, but it is still possible for them to yield similar network structures and parameters at the group level. To inspect whether the choice of the analysis approaches impacts the group-level model, we compared the GMSs $\bar{G}$ or GMCs $\bar{B}$ across approaches. For each of the three comparisons (IS vs. CS, IS vs. VTS and CS vs VTS), we element-wise plotted the GMSs or GMCs across the two approaches under comparison. Because an element of $\bar{B}$ will become zero when its corresponding element of $\bar{G}$ is zero, an element of $\bar{B}$ is plotted only if its corresponding element of $\bar{G}$ is larger than 1%, to eliminate the information that has already been presented in the comparison of GMSs. If the dots on the plotted graphs are located roughly alone the diagonal line, the two approaches yield similar network structures or parameters at the group level.

As a statistical metric, a BIC score alone cannot indicate whether a model approximates the underlying biomedical truth well, so the sensitivity of the three approaches
3.2. Materials and Methods

to the possible effect of the L-dopa medication was also compared. Because it has been demonstrated that disturbances of rhythmic movement in PD is related to bradykinesia (slowness of movement) and L-dopa is most effective against bradykinesia, we assume that the L-dopa medication will generally normalize PD patients’ brain connectivity in this frequency-related bulb-squeezing task. The overall effect of the medication was verified by checking whether the mixture DBN model of each patient before medication (denoted as the P_pre group) changed toward those of the normal group (denoted as the N group) after the patient took the medication (denoted as the P_post group).

At the structure level, a connection is considered to have been changed toward the normal group, if elements of $G^s_{P_{s_{post}}} - G^s_{P_{s_{pre}}}$ and $G_N - G^s_{P_{s_{pre}}}$ (where subscripts “$P_{s_{pre}}$” and “$P_{s_{post}}$” indicate that the comparison is about patient $s$) have the same signs, or equally elements of

$$G'_s = (G^s_{P_{s_{post}}} - G^s_{P_{s_{pre}}}) * \text{sign}(G_N - G^s_{P_{s_{pre}}})$$ (3.9)

are positive where “$s$” denotes element-wise products. At the parameter level, a connection was considered to have been changed toward the normal group, if elements of $B^s_{P_{s_{post}}} - B^s_{P_{s_{pre}}}$ and $B_N - B^s_{P_{s_{pre}}}$ had the same sign, or equally elements of

$$B'_s = (B^s_{P_{s_{post}}} - B^s_{P_{s_{pre}}}) * \text{sign}(B_N - B^s_{P_{s_{pre}}})$$ (3.10)

are positive. To summarize this patient-matched comparison at the group level, we averaged $G'_s$ and $B'_s$ over all the patients as $G' = \sum_{s=1}^{S} G'_s / S$ and $B' = \sum_{s=1}^{S} B'_s / S$ respectively. If an element of $G'$ or $B'$ is positive, we call the corresponding connection a “normalized connection” at the structure or the parameter level respectively. The same patient performed the task twice, once before and the other after the medication, so the comparison is patient-matched. Because the normal group are healthy people which can not be individually matched with patients, its group means ($\bar{G}_N$ and $\bar{B}_N$) are used to
represent healthy people. The signs of $G'_s$ and $B'_s$ are standardized relatively to the signs of $\tilde{G}_N - G_{P_{s pre}}^*$ and $\tilde{B}_N - B_{P_{s pre}}^*$ respectively, so that positive values indicate that the change is toward that of healthy people.

Based on the “normalized connections”, we performed tests between the two hypotheses below:

- $H_0$: medication does not functionally change brain connectivity in PD overall.
- $H_1$: medication changes the PD subjects’ connections between brain regions functionally toward those of normal subjects.

The possibility of a connection appearing to be “normalized” by the medication is parametrized as $\gamma$. The value of $\gamma$ equals 0.5 under $H_0$, due to the effect of randomness; $\gamma$ is expected to be greater than 0.5 under $H_1$. If $m$ out of $n$ connections appear to be normalized by medication, the right-tailed $p$-value of this observation under $H_0$ is $\sum_{i=m}^{n} C_i^n / 2^n$.

To further assess the similarity/diversity of the DBN structures learned with the IS approach, the variance of a connection’s posterior probabilities i.e. $\text{Var}[g_{ab}^*]$ was calculated among subjects’ mixture DBNs. A small $\text{Var}[g_{ab}^*]$ suggests that the connection from ROI $a$ to ROI $b$ consistently appears (or does not appear) in subjects’ mixture DBNs; a large variance suggests that its appearance differs considerably among subjects’ mixture DBNs. If a big portion of connections have small $\text{Var}[g_{ab}^*]$, then the DBNs learned individually for each subject are structurally consistent; if a big portion of connections have large $\text{Var}[g_{ab}^*]$, then the DBNs structurally differs among subjects. Please note that the upper bound of the variance is 0.5, not 1, because posterior probabilities are in the range of [0, 1].
3.2.5 Group Comparison

The group analysis is to identify those connections that appear to be statistically significantly normalized by the medication. Though a connection appears to be normalized if its corresponding element in $B'$ is positive, the statistical significance of the normalization still need to be tested. In this section, we employed mixed-effect models to verify whether elements of $B'$ are significantly positive. Because the data of the normal group were best fitted with the VTS approach and the data of the patient groups were best fitted with the IS approach (see Table 3.2), we applied the $B_N$ of the VTS approach and the $B_{Ps^{pre}}$ and $B_{Ps^{post}}$ of the IS approach to Eq. (3.10) to calculate $B'$.

An element $\beta_{abl}$ of $B'$ is associated with a connection from $a$ to $b$ and a frequency level $l$. When the connection from $a$ to $b$ is under investigation, $\beta_{abl}$ from all the patients and at all the frequency levels can be regressed to the frequency level as in Eq. (3.11),

$$\beta_{ab} = c + \text{Freq} + e,$$  (3.11)

where $\beta_{ab}$ is a vector of $\beta_{abl}$, $c$ is a constant, and vector $e$ is the regression errors which follow a zero-mean multivariate Gaussian distribution. $\beta_{ab}$ can also be just regressed to a constant as in Eq. (3.12) without the term Freq if the effect of the medication does not differ at different task frequencies.

$$\beta'_{ab} = c + e.$$  (3.12)

To accommodate the variability among the randomly sampled subjects and as well the noise in the fMRI of each subject, the variance of $e$ is composed of two parts: $V = V_a + V_w$ which are the variance among and within subjects respectively, in the way of the summary statistics approach. Since the variance term includes both among-subject and within-subject terms, this is a mixed-effect model that can infer about the population,
rather than just the pool of the recruited subjects. $V_a$ has the pattern of $\sigma^2 I$, under the assumption that subjects are independently sampled from the population. $V_w$ is a blocked-diagonal matrix whose diagonal blocks are variances within subjects. The diagonal blocks of $V_w$ are estimated at the stage of calculating $\beta_{\text{adj}}$, as $V_{ws} = V(\beta_{\text{pre}}^s) + V(\beta_{\text{post}}^s)$ where the subscript $s$ indicates that the block is associated with patient $s$.

Eqs. (3.11) and (3.12) were solved with the restricted maximum likelihood method \cite{2} to ensure an unbiased estimation of $V$.

The analysis includes three steps. First, nested models Eq. (3.11) and Eq. (3.12) are compared with likelihood-ratio test, to test whether the medication altered the connections differently for tasks at different frequencies. If Eq. (3.11) fits the data significantly better than Eq. (3.12) does, then the medication has different effects for tasks at different frequencies. However, significant results were not discovered for any connection (see Section 3.3), so all the preceded analyses were performed with Eq. (3.12). Second, whether $c$ is equal to or significantly above zero is tested with right-tailed t-tests. A $p$-value was calculated for each possible connection in the DBN. Finally, the effect of testing multiple connections simultaneously was adjusted by Storey’s positive-false-discovery-rate (pFDR) procedure \cite{32} which controls the pFDR (characterized by a $q$-value), the expected ratio of falsely rejected hypotheses among all those being rejected. Connections whose $q$-values were smaller than 5% were selected, and regarded to have been statistically significantly normalized by the medication.

### 3.3 Results

Table 3.1 shows the group-level BIC scores of the simulated data set. The generating models were correctly identified with the highest BIC scores. Table 3.2 shows the group-level BIC scores of the real data set. For the normal group, the VTS approach yielded the highest group-level BIC score; for the pre-medication and post-medication patient
3.3. Results

Table 3.1: Group-level BIC Scores of Simulated Data

<table>
<thead>
<tr>
<th></th>
<th>IS</th>
<th>CS</th>
<th>VTS (Pool)</th>
</tr>
</thead>
<tbody>
<tr>
<td>True Models IS</td>
<td>-5.40e3</td>
<td>-5.85e3</td>
<td>-6.47e3</td>
</tr>
<tr>
<td>CS</td>
<td>-5.42e3</td>
<td>-5.35e3</td>
<td>-5.68e3</td>
</tr>
<tr>
<td>VTS</td>
<td>-6.27e3</td>
<td>-6.36e3</td>
<td>-5.63e3</td>
</tr>
</tbody>
</table>

Group-level BIC scores of the IS approach, the CS approach and the VTS approach are defined in Eqs. (3.6), (3.7) and (3.8) respectively. Bold digits are the highest BIC scores found by the three approaches.

Table 3.2: Group-level BIC Scores of Real Data

<table>
<thead>
<tr>
<th></th>
<th>IS</th>
<th>CS</th>
<th>VTS (Pool)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>-5.25e3</td>
<td>-5.38e3</td>
<td>-5.06e3</td>
</tr>
<tr>
<td>P_{pre}</td>
<td>-4.99e3</td>
<td>-5.21e3</td>
<td>-5.31e3</td>
</tr>
<tr>
<td>P_{post}</td>
<td>-6.11e3</td>
<td>-6.55e3</td>
<td>-6.39e3</td>
</tr>
</tbody>
</table>

Group-level BIC scores of the IS approach, the CS approach and the VTS approach are defined in Eqs. (3.6), (3.7) and (3.8) respectively. Bold digits are the highest BIC scores found by the three approaches. “N”, “P_{pre}” and “P_{post}” denote the normal group, the pre-medication group and the post-medication group respectively.

groups, the IS approach yielded the highest group-level BIC scores. The best fitting approaches outperformed the second best fitting approaches with at least 190 BIC scores, which implies that if the conditions of the BIC’s asymptomatic approximation to the log posterior probability are satisfied, then the model of the best fitting approach is $e^{190}$ times more likely than that of the second best fitting approach. Though the conditions of the BIC’s asymptomatic approximation may not be guaranteed in this study, the sizeable contrasts among the BIC scores still suggest that the goodness-of-fit of the three approaches is considerably different.

Fig. 3.3 shows the element-wise comparison of the group mean structures learned with the three approaches. If two approaches learn similar networks at the group level, dots on the figure should be located close to the (0, 0) to (1, 1) diagonal lines. However, this phenomenon is not observed. The connections’ posterior probabilities estimated by the CS approach tend to cluster at extreme values near 0 and 1, while those estimated by the IS approach cover the range from 0 to 1 more evenly.
3.3. Results

Figure 3.3: Comparisons of the group mean structures. The group-mean-structure matrices $\bar{G}$ of the IS, CS and VTS approaches are plotted element-wise. Plots of the three experimental groups are overlaid with different dot symbols. Legends “N”, “Ppre” and “Ppost” denote the normal group, the pre-medication group and the post-medication group respectively.
3.3. Results

Figure 3.4: Comparisons of the group mean coefficients. The group mean coefficients $\bar{B}$ of the IS, CS, and VTS approaches are plotted element-wise. An element of $\bar{B}$ is plotted only if its corresponding element of $\bar{G}$ is larger than 1%. Plots of the three experimental groups are overlaid with different dot symbols. Legends “N”, “Ppre” and “Ppost” denote the normal group, the pre-medication group and the post-medication group respectively.
3.3. Results

Table 3.3: The counts of normalized connections

<table>
<thead>
<tr>
<th></th>
<th>IS</th>
<th></th>
<th>CS</th>
<th></th>
<th>VTS (Pool)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Count</td>
<td>p</td>
<td>Count</td>
<td>p</td>
<td>Count</td>
<td>p</td>
</tr>
<tr>
<td>Struct.</td>
<td>59/66</td>
<td>0.0000</td>
<td>15/66</td>
<td>1.0000</td>
<td>29/66</td>
<td>0.8661</td>
</tr>
<tr>
<td>.00Hz</td>
<td>64/66</td>
<td>0.0000</td>
<td>27/66</td>
<td>0.9456</td>
<td>35/66</td>
<td>0.3561</td>
</tr>
<tr>
<td>.25Hz</td>
<td>62/66</td>
<td>0.0000</td>
<td>27/66</td>
<td>0.9456</td>
<td>34/66</td>
<td>0.4511</td>
</tr>
<tr>
<td>.50Hz</td>
<td>62/66</td>
<td>0.0000</td>
<td>27/66</td>
<td>0.9456</td>
<td>35/66</td>
<td>0.3561</td>
</tr>
<tr>
<td>.75Hz</td>
<td>64/66</td>
<td>0.0000</td>
<td>27/66</td>
<td>0.9456</td>
<td>34/66</td>
<td>0.4511</td>
</tr>
<tr>
<td>Mean</td>
<td>63/66</td>
<td>0.0000</td>
<td>27/66</td>
<td>0.9456</td>
<td>34/66</td>
<td>0.4511</td>
</tr>
</tbody>
</table>

The format of counts is “the number of normalized connections / the number of possible connections”. In row “Struct.” are the counts of positive elements in $G'$; in rows from “.00Hz” to “.75Hz” are the counts of positive elements in $B'$, given the task frequency level; in row “Mean” are the counts of positive elements in $B'$, with the elements at the four frequency levels averaged.

Fig. 3.4 shows the element-wise comparison of the group mean parameters learned with the three approaches. To remove the information that Fig. 3.3 has showed, an element of $\bar{B}$ is plotted only if its corresponding element of $\bar{G}$ are larger than 1%. Dots are located roughly along the (-1,-1) to (1,1) diagonal lines, which suggests that the three approaches tend to learn similar coefficients at the group level for those connections that always appear in the learned networks no matter which approach is applied.

Table 3.3 shows the number of normalized connections detected with the three approaches. The IS approach detected more normalized connections than the other two approaches. It found that at least 59 connections out of all the 66 possible ones were normalized after the medication. The $p$-value of this observation is smaller than $1.193 \times 10^{-11}$, under the hypothesis that the medication does not improve patients’ brain connectivity overall.

Fig. 3.5 shows the histogram of $Var[g_{ab}^*]$, i.e. the variances of connections’ posterior probabilities among the mixture DBN models learned with the IS approach. Given the fact that the variance of posterior probabilities cannot exceed 0.5, a large portion of connections vary considerably among the networks of different subjects, which suggests that different subjects’ networks learned with the IS approach are not consistent. Even though subjects of the normal group probably share the same brain connectivity pattern,
3.3. Results

Figure 3.5: The histogram of the variances of connections’ posterior probabilities estimated with the IS approach. The posterior probability of a connection’s existence was estimated individually for each subject, and the variance of a connection’s posterior probabilities among different subjects was then calculated. This figure shows the histogram of the posterior-probability variances of the connections. Legends “N”, “Ppre” and “Ppost” denote the normal group, the pre-medication group and the post-medication group respectively.
3.3. Results

Table 3.4: Connections whose coefficients were statistically significantly normalized toward those of the normal group.

<table>
<thead>
<tr>
<th>Connection</th>
<th>t-stat</th>
<th>d.f.</th>
<th>p</th>
<th>q</th>
</tr>
</thead>
<tbody>
<tr>
<td>R_CER → R_CER</td>
<td>2.5828</td>
<td>18</td>
<td>0.0094</td>
<td>0.0424</td>
</tr>
<tr>
<td>R_CER → L_CER</td>
<td>4.1458</td>
<td>15</td>
<td>0.0004</td>
<td>0.0065</td>
</tr>
<tr>
<td>R_M1 → R_M1</td>
<td>3.0563</td>
<td>18</td>
<td>0.0034</td>
<td>0.0275</td>
</tr>
<tr>
<td>R_SMA → R_CER</td>
<td>2.6912</td>
<td>27</td>
<td>0.0060</td>
<td>0.0362</td>
</tr>
<tr>
<td>L_CER → R_CER</td>
<td>6.6237</td>
<td>15</td>
<td>0.0000</td>
<td>0.0001</td>
</tr>
<tr>
<td>L_CER → L_CER</td>
<td>2.6565</td>
<td>9</td>
<td>0.0131</td>
<td>0.0424</td>
</tr>
<tr>
<td>R_CER → R_M1</td>
<td>2.4417</td>
<td>12</td>
<td>0.0155</td>
<td>0.0424</td>
</tr>
<tr>
<td>R_M1 → R_CER</td>
<td>3.0219</td>
<td>18</td>
<td>0.0037</td>
<td>0.0275</td>
</tr>
<tr>
<td>R_M1 → L_M1</td>
<td>2.4717</td>
<td>12</td>
<td>0.0147</td>
<td>0.0424</td>
</tr>
<tr>
<td>R_SMA → R_CER</td>
<td>2.3685</td>
<td>18</td>
<td>0.0146</td>
<td>0.0424</td>
</tr>
<tr>
<td>L_SMA → R_SMA</td>
<td>2.4601</td>
<td>12</td>
<td>0.0150</td>
<td>0.0424</td>
</tr>
</tbody>
</table>

Arrows → denote connections with a time lag; arrows ↔ denote connections without a time lag. “d.f.” is short for “degree of freedom”.

which is suggested by the highest BIC score learned with the VTS approach, networks still vary considerably from subject to subject when the IS approach was applied.

The likelihood-rate tests show that Eq. (3.11) does not fit $\beta'_{ab}$ significantly better than Eq. (3.12) does for any connection, with the minimum $p$-value larger than 0.0769, before correction for the effect of multiple comparisons. This result does not support that the medication differently altered the connections for tasks at different frequencies. This does not mean that the medication does not altered the connections, but just that the alterations are probably the same at different frequency levels. The connections whose coefficients were significantly normalized according to the right-tailed t-test with Eq. (3.12) are listed in Table 3.4 and further graphically illustrated in Fig. 3.6. These results indicate that the right SMA, right M1 and left cerebellum, all part of the motor system expected to be activated with the left hand were modulated by L-dopa mediation.
3.3. Results

Figure 3.6: Connections whose coefficients were statistically significantly normalized toward those of the normal group. Solid arrows denote connections with a time lag; line arrows denote those without a time lag.
3.4 Discussion

The comparison of the three DBN-based group-analysis approaches (the “virtual-typical-subject” (VTS), “common-structure” (CS) and “individual-structure” (IS) approaches) suggests that no single approach is universally superior over the other two. For this fMRI study on Parkinson’s disease that included a motor task at different frequencies, the VTS approach fits the data of the normal group best, while the IS approach fits the data of the pre-medication and post-medication patient group best, from the view of statistics with the BIC score as the goodness-of-fit metric (see Table 3.2).

The three approaches led to considerably different group-level results, learning different network structures, and detecting different numbers of connections normalized by the medication, from the same data set. Fig. 3.3 shows that the probability of a connection’s existence may be estimated to be quite high with one approach, but quite low with another approach. The overall effect of the L-dopa medication on the brain connectivity of Parkinson’s disease patients are also estimated quite differently with the three approaches, with the proportion of normalized connections varies from 15/66 to 64/66, as shown in Table 3.3. The results of the IS approach support the assumption better that the L-dopa medication normalizes the brain connectivity for the patients than those of the other two approaches do. The regions showing normalized activity after L-dopa medication are consistent with motor tasks involving the left hand. The sizeable differences among the results of the three approaches suggest that choice over the three approaches influence the analysis considerably. Analysts should not arbitrarily choose an approach without justification. However, as Fig. 3.4 shows, the three approaches tend to learn similar coefficients at the group level for those connections that always appear in the learned networks no matter which approach is applied.

If the IS approach is applied, the networks tend to vary considerably from subject to subject, even when statistical metrics support that the underlying true models probably share the same structure. For the normal group in this study, its highest BIC score was
learned with the VTS approach, suggesting that the data of those normal people are more similar than dissimilar. However, the networks learned with the IS approach still vary considerably from subject to subject, as Fig. 3.5 shows sizeable variances of connections’ posterior probabilities among the individual DBN models. The reason behind this phenomenon probably is that the limited time points of each individual’s data do not support an accurate and robust estimation of the true model. The inconsistency among individual networks is not effective evidence of inter-subject variability, if only the IS approach is applied.

Each of the three approaches has its limitations in practice. The IS approach do not take advantage of the similarity among group members, so when each subject’s data is not long enough, it tend to learn non-robust networks for each subject as mentioned above with Fig. 3.5. The VTS approach does not accommodate differences among subjects, but inter-subject variability is believed to be a common factor in biomedical studies. As shown in Table 3.3, ignoring the differences among patients may limit the analysis’ sensitivity to the effect of the L-dopa medication. The CS approach balances the similarity and diversity among group members at the point that the connectivity structure is the same for all the subjects but details of the connectivity are different. However it is not sensitive to the effect of the L-dopa medication either, as shown in Table 3.3. It is possible that the subjects’ connectivity networks are the same partially at some connections but differs at other connections, in which case none of the three approaches is suitable. It is ultimately desirable to develop a method that allows researchers to adjust the degree of the balance between the similarity and diversity among group members. The method Niculescu-Mizil and Caruana [22] proposed is a possible solution. It allows network structures to be different while punishes excessive diversity among the structures and controls the degree of the punishment with an adjustable parameter. Detection of sub-groups and outliers can also improve group analysis when inter-subject variability cannot be ignored [11].
3.4. Discussion

Group-analysis approaches should be selected according to comprehensive criteria, such as both statistical and biomedical evidence. In exploratory researches where the ground truth is usually unknown, statistical evidence is helpful in the selection of group-analysis methods. Besides the BIC score used in the paper, the model evidence calculated with MCMC is a more accurate metric, at the cost of intensive computation, since the BIC score which asymptotically approximates the log posterior probability is inaccurate when the sample size is small. Other goodness-of-fit metrics, such as the Dirichlet Prior Score Metric (DPSM), and the Bayesian Dirichlet metrics (BDe) can also be used [37]. The confidence of the statistical evidence can be verified with data simulated from the learned models. Generating data from the models learned from the real data is to make the simulated data as similar to the real data as possible. However, statistical evidence alone is not sufficient in making decisions, especially in biomedical studies, if heterogeneity of group members are not considered from other aspects such as age, gender and etc. If possible, researchers can design biomedical markers, and select group-analysis approaches according to their ability to detect those markers, as we assumed the effect of the L-dopa medication in the study.

While the author of this thesis was finalizing the writing, he noticed a valuable reflection by Robert L. Savoy on current fMRI-based research that is quite related to group analysis. In [29], Savoy cautioned against the current torrent of using a large number of subjects to conduct group analysis, but recommended focusing on studying a small number of subjects. From the negative aspect, he argued, first, that due to possible misregistration it is problematic to conform different brains to a standard shape, and second, that group averaging can be misleading when outliers exist. From the positive aspect, he pointed out, first, that some interesting discoveries actually relied on the finding of exceptional subjects, and second, that some fMRI work, such as that in presurgical planning, is inherently just for a single subject. He recommended thoroughly studying a few subjects with various experimental factors, such as, gender, or education background.
3.4. Discussion

He concluded “it may someday turn out that the information from a few brains, thoroughly studied, will reveal more about the universal aspects of human brain function and organization than the current torrent of studies from large collections of brains.”

Acknowledgments

This work was supported by NSERC Grant CHRPJ 323602-06 (MJM) and CIHR grant CPN-80080 (MJM).
Bibliography


[18] A. R. McIntosh and F. Gonzalez-Lima. Structural equation modeling and its appli-


Chapter 4

Controlling the Error Rate in Learning Network Structures

In real world applications, graphical statistical models are not only a tool for operations such as classification or prediction, but usually the network structures of the models themselves are also of great interest (e.g. in modeling brain connectivity). The false discovery rate (FDR), the expected ratio of falsely claimed connections to all those claimed, is often a reasonable error-rate criterion in these applications. However, current learning algorithms for graphical models have not been adequately adapted to the concerns of the FDR. The traditional practice of controlling the type I error rate and the type II error rate under a conventional level does not necessarily keep the FDR low, especially in the case of sparse networks. We propose embedding an FDR-control procedure into the PC algorithm to curb the FDR of the skeleton of the learned graph. It is proved that the proposed method can control the FDR under user-specified levels at the limit of large sample sizes. In the cases of moderate sample size (about several hundred), empirical experiments show that the method is still able to control the FDR under the user-specified level, and a heuristic modification of the method is able to control the FDR accurately around the user-specified level. The proposed method is applicable to any models for which statistical tests of conditional independence are available, such as discrete models and Gaussian models.

4.1 Introduction

Graphical models have attracted increasing attention in the fields of data mining and machine learning in the last decade. These models, such as Bayesian networks (also called belief networks) and Markov random fields, generally represent events or random variables as vertices (also referred to as nodes), and encode conditional-independence relationships among the events or variables as directed or undirected edges (also referred to as arcs) according to the Markov properties [see \textsuperscript{18}, chapt. 3]. Of particular interest here are Bayesian networks [see \textsuperscript{25}, chapt. 3.3] that encode conditional-independence relationships according to the directed Markov property [see \textsuperscript{18} pages 46–53] with directed acyclic graphs (DAGs) (\textit{i.e.} graphs with only directed edges and with no directed cycles). The directed acyclic feature facilitates the computation of Bayesian networks because the joint probability can be factorized recursively into many local conditional probabilities.

As a fundamental and intuitive tool to analyze and visualize the association and/or causality relationships among multiple events, graphical models have become more and more explored in biomedical researches, such as discovering gene regulatory networks and modelling functional connectivity between brain regions. In these real world applications, graphical models are not only a tool for operations such as classification or prediction, but often the network structures of the models themselves are also output of great interest: a set of association and/or causality relationships discovered from experimental observations. For these applications, a desirable structure-learning method needs to account for the error rate of the graphical features of the discovered network. Thus, it is important for structure-learning algorithms to control the error rate of the association/causality relationships discovered from a limited number of observations closely below a user-specified level, in addition to finding a model that fits the data well. As edges are fundamental elements of a graph, error rates related to them are of natural concerns.

In a statistical decision process, there are basically two sources of errors: the type
I errors, \textit{i.e.} falsely rejecting negative hypotheses when they are actually true; and the type II errors, \textit{i.e.} falsely accepting negative hypotheses when their alternatives, the positive hypotheses are actually true. In the context of learning graph structures, a negative hypothesis could be that an edge does not exist in the graph while the positive hypothesis could be that the edge does exist. Because of the stochastic nature of random sampling, data of a limited sample size may appear to support a positive hypothesis more than a negative hypothesis even when actually the negative hypothesis is true, or vice versa. Thus it is generally impossible to absolutely prevent the two types of errors simultaneously, but has to set a threshold on a certain type of errors, or keep a balance between the them, for instance by minimizing a certain lost function associated with the errors according to the Bayesian decision theory. For example, when diagnosing cancer, to catch the potential chance of saving a patient’s life, doctors probably hope that the type II error rate, \textit{i.e.} the probability of falsely diagnosing a cancer patient as healthy, to be low, such as less than 5%. Meanwhile, when diagnosing a disease whose treatment is so risky that may cause the loss of eyesight, to avoid the unnecessary but great risk for healthy people, doctors probably hope that the type I error rate, \textit{i.e.} the probability of falsely diagnosing a healthy people as affected by the disease, to be extremely low, such as less than 0.1%. Learning network structures may face scenarios similar to the two cases above of diagnosing diseases. Given data of a limited sample size, there is not an algorithm guaranteeing a perfect recovery of the structure of the underlying graphical model, and any algorithm has to compromise on the two types of errors.

For problems involving simultaneously testing multiple hypotheses, such as verifying the existence of edges in a graph, there are several different criteria for their error-rate control (see Table 4.2), depending on researchers’ concerns or the scenario of the study. Generally there are not mathematically or technically superior relationships among different error-rate criteria if the research scenario is not specified. One error-rate criterion may be favoured in one scenario while another criterion may be right of interest in a
different scenario, just as the aforementioned examples of diagnosing diseases. In real world applications, selecting the error rate of interest is largely not an abstract question “which error rate is superior over others?” , but a practical question “which error rate is the researchers’ concern?” For extended discussions on why there are not general superior relationships among different error-rate criteria, please refer to Appendix 4.7, where examples of typical research scenarios, accompanied by theoretical discussions, illustrate that each of the four error-rate criteria in Table 4.2 may be favoured in a certain study.

The false discovery rate (FDR) [see 4, 34], defined as the expected ratio of falsely discovered positive hypotheses to all those discovered, has become an important and widely used criterion in many research fields, such as bioinformatics and neuroimaging. In many real world applications that involve multiple hypothesis testing, the FDR is more reasonable than the traditional type I error rate and type II error rate. Suppose that in a pilot study researchers are selecting candidate genes for a genetic research on schizophrenia. Due to the limited funding, only a limited number of genes can be studied thoroughly in the afterward genetic research. To use the funding efficiently, researchers would hope that 95% of the candidate genes selected in the pilot study are truly associated with the disease. In this case, the FDR is chosen as the error rate of interest and should be controlled under 5%. Simply controlling the type I error rate and the type II error rate under certain levels does not necessarily keep the FDR sufficiently low, especially in the case of sparse networks. For example, suppose a gene regulatory network involves 100 genes, where each gene interacts in average with 3 others, i.e. there are 150 edges in the network. Then an algorithm with the realized type I error rate = 5% and the realized power = 90% (i.e. the realized type II error rate = 10%) will recover a network with $150 \times 90\% = 135$ correct connections and $[100 \times (100 - 1)/2 - 150] \times 5\% = 240$ false connections. This means that $240/(240 + 135) = 64\%$ of the claimed connections actually do not exist in the true network. Due to the popularity of the FDR in research practices, it is highly desirable to develop structure-learning algorithms that allow the control over
4.1. Introduction

However, current structure-learning algorithms for Bayesian networks have not been adequately adapted to explicitly control the FDR of the claimed “discovered” networks. Score-based search methods [see 13] look for a suitable structure by optimizing a certain criterion of goodness-of-fit, such as the Akaike information criterion (AIC), the Bayesian information criterion (BIC), or the Bayesian Dirichlet likelihood equivalent metric (BDE), with a random walk (e.g. simulated annealing) or a greedy walk (e.g. hill-climbing), in the space of DAGs or their equivalence classes. It is worth noting that the restricted case of tree-structured Bayesian networks has been optimally solved, in the sense of Kullback-Leibler divergence, with Chow and Liu [6]'s method, and that Chickering [5] has proved that the greedy equivalence search can identify the true equivalence class in the limit of large sample sizes. Nevertheless, scores do not directly reflect the error rate of edges, and the sample sizes in real world applications are usually not large enough to guarantee the perfect asymptotic identification.

The Bayesian approach first assumes a certain prior probability distribution over the network structures, and then estimates the posterior probability distribution of the structures after data are observed. Theoretically, the posterior probability of any structure features, such as the existence of an edge, the existence of a path, or even the existence of a sub-graph, can be estimated with the Bayesian approach. This consequently allows the control of the posterior error rate of these structure features, i.e. the posterior probability of the non-existence of these features. It should be pointed out that the posterior error rate is conceptually different from those error rates such as the type I error rate, the type II error rate, and the FDR, basically because they are from different statistical perspectives. The posterior error rate is defined from the perspective of Bayesian statistics. From the Bayesian perspective, network structures are assumed to be random, according to a probability distribution, and the posterior error rate is the probability of

---

6 An equivalence class of DAGs is a set of DAGs that encode the same set of conditional-independence relationships according to the directed Markov property.
4.1. Introduction

the non-existence of certain features according to the posterior probability distribution over the network structures. Given the same data, different posterior distributions will be derived from different prior distributions. The type I error rate, the type II error rate, and the FDR are defined from the perspective of classical statistics. From the classical perspective, there is a true, yet unknown, model behind the data, and the error rates are defined by comparing with the true model. Nevertheless, a variant of the FDR, the positive false discovery rate (pFDR) proposed by Storey [34], can be interpreted from the Bayesian perspective [35].

The Bayesian approach for structure learning is usually conducted with the maximum-a posteriori-probability (MAP) method or the posterior-expectation method. The MAP method selects the network structure with the largest posterior probability. The optimal structure is usually searched for in a score-based manner, with the posterior probability or more often approximations to the relative posterior probability (for instance the BIC score) being the score to optimize. Cooper and Herskovits [7] developed a heuristic greedy search algorithm called K2 that can finish the search in a polynomial time with respect to the number of vertices, given the order of vertices. The MAP method provides us with a single network structure, the posteriorly most probable one, but does not address error rates in the Bayesian approach.

To control errors in the Bayesian approach, the network structure should be learned with the posterior-expectation method, i.e. calculating the posterior probabilities of network structures, and then deriving the posterior expectation of the existence of certain structure features. Though theoretically the posterior-expectation method can control the error rate of any structure features, in practice its capacity is largely limited for computational reasons. The number of DAGs increases super-exponentially as the number of vertices increases [29]. For 10 vertices, there are already about $4.2 \times 10^{18}$ DAGs. Though the number of equivalence classes of DAGs is much smaller than the number of DAGs, it is

7The algorithm is named K2 because it evolved from a system named Kutato [14]
4.1. Introduction

still forbiddingly large, empirically asymptotically decreasing to 1/13.652 of the number of DAGs, as the number of vertices increases \([33]\). Therefore, exact inference of posterior probabilities is only feasible for small scale problems, or under certain additional constraints. For certain prior distributions, and given the order of vertices, Friedman and Koller \([12]\) have derived a formula that can be used to calculate the exact posterior probability of a structure feature with the computational complexity bounded by \(O(N^{D_{\text{in}}+1})\), where \(N\) is the number of vertices and \(D_{\text{in}}\) is the upper bound of the in-degree for each vertex. Considering similar prior distributions, but without the restriction on the order of vertices, Koivisto and Sood \([17]\) have developed a fast exact-Bayesian-inference algorithm based on dynamic programming that is able to compute the exact posterior probability of a sub-network with the computational complexity bounded by \(O(N2^N + N^{D_{\text{in}}+1}L(m))\), where \(L(m)\) is the complexity of computing a marginal conditional likelihood from \(m\) samples. In practice, this algorithm runs fairly fast when the number of vertices is less than 25. For networks with more than 30 vertices, the authors suggested setting more restrictions or combining with inexact techniques. These two breakthroughs made exact Bayesian inference practical for certain prior distributions. However, as Friedman and Koller \([12]\) pointed out, the prior distributions which facilitate the exact inference are not hypothesis equivalent \([see\ [13]\), \(i.e.\) different network structures that are in the same equivalence class often have different priors. The simulation performed by Eaton and Murphy \([8]\) confirmed that these prior distributions deviate far from the uniform distributions. This implies that the methods cannot be applied to the widely accepted uninformative prior, \(i.e.\) the uniform prior distribution over DAGs. For general problems, the posterior probability of a structure feature can be approximated with Markov chain Monte Carlo (MCMC) methods \([21]\). As a versatile implementation of Bayesian inference, the MCMC method can estimate the posterior probability given any prior probability distribution. However, MCMC usually requires intensive computation and the results may depend on the initial state of randomization.
4.1. Introduction

Constraint-based approaches, such as the SGS\(^8\) algorithm [see 32, pages 82–83], inductive causation (IC)\(^9\) [see 26, pages 49–51], and the PC\(^{10}\) algorithm [see 32, pages 84–89], are rooted in the directed Markov property, the rule by which Bayesian networks encode conditional independence. These methods first test hypotheses of conditional independence among random variables, and then combine those accepted hypotheses of conditional independence to construct a partially directed acyclic graph (PDAG) according to the directed Markov property. The computational complexity of these algorithms is difficult to analyze exactly, though for the worst case, which rarely occurs in real world applications, is surely bounded by \(O(N^{2N})\) where \(N\) is the number of vertices. In practice, the PC algorithm and the fast-causal-inference (FCI) algorithm [see 32, pages 142–146] can achieve polynomial time if the maximum degree of a graph is fixed. It has been proved that if the true model satisfies the faithfulness constraints [see 32, pages 13 and 81] and all the conditional-independence/dependence relationships are correctly identified, then the PC algorithm and the IC algorithm can exactly recover the true equivalence class, and so do the FCI algorithm and the IC* algorithm\(^{11}\) [see 26, pages 51–54] for problems with latent variables. Kalisch and Bühlmann\(^{12}\) have proved that for Gaussian Bayesian networks, the PC algorithm can consistently estimate the equivalence class of an underlying sparse DAG as the sample size \(m\) approaches infinity, even if the number of vertices \(N\) grows as fast as \(O(m^\lambda)\) for any \(0 < \lambda < \infty\). Yet, as in practice hypotheses of conditional independence are tested with statistical inference from limited data, false decisions cannot be entirely avoided and thus the ideal recovery cannot be achieved. In current implementations of the constraint-based approaches, the error rate of testing conditional independence is usually controlled individually for each test, under

---

\(^8\)“SGS” stands for Spirtes, Glymour and Scheines who invented this algorithm.

\(^9\) An extension of the IC algorithm which was named as IC* [see 26, pages 51–54] was previously also named as IC by Pearl and Verma [27]. Here we follow Pearl [26].

\(^10\) “PC” stands for Peter Spirtes and Clark Glymour who invented this algorithm. A modified version of the PC algorithm which was named as PC* [see 32, pages 89–90] was previously also named as PC by Spirtes and Glymour [31]. Here we follow Spirtes et al. [32].

\(^11\) See footnote 9
a conventional level such as 5% or 1%, without correcting the effect of multiple hypothesis testing. Therefore these implementations may fail to curb the FDR, especially for sparse graphs.

In this paper, we propose embedding an FDR-control procedure into the PC algorithm to curb the error rate of the skeleton of the learned PDAGs. Instead of individually controlling the type I error rate of each hypothesis test, the FDR-control procedure considers the hypothesis tests together to correct the effect of simultaneously testing the existence of multiple edges. We prove that the proposed method, named as the PC\textsubscript{fdr}-skeleton algorithm, can control the FDR under a user-specified level at the limit of large sample sizes. In the case of moderate sample sizes (about several hundred), empirical experiments show that the method is able to control the FDR under the user-specified level, and a heuristic modification of the method is able to control the FDR more accurately around the user-specified level. Schäfer and Strimmer [30] have applied an FDR procedure to graphical Gaussian models to control the FDR of the non-zero entries of the partial correlation matrix. Different from Schäfer and Strimmer [30]'s work, our method, built within the framework of the PC algorithm, is not only applicable to the special case of Gaussian models, but also generally applicable to any models for which conditional-independence tests are available, such as discrete models.

We are particularly interested in the PC algorithm because it roots in conditional-independence relationships, the backbone of Bayesian networks, and \( p \)-values of hypothesis testing represent one type of error rates. We consider the skeleton of graphs because constraint-based algorithms usually first construct an undirected graph, and then annotate it into different types of graphs while keeping the skeleton as the same as that of the undirected one.

The PC\textsubscript{fdr}-skeleton algorithm is not designed to replace or claimed to be superior over the standard PC algorithm, but provide the PC algorithm with the ability to control the FDR over the skeleton of the recovered network. The PC\textsubscript{fdr}-skeleton algorithm controls
the FDR while the standard PC algorithm controls the type I error rate, as illustrated in Section 4.3.1. Since there are no general superior relationships between different error-rate criteria, as explained in Appendix 4.7, neither be there between the PC_{fdr}-skeleton algorithm and the standard PC algorithm. In research practices, researchers first decide which error rate is of interest, and then choose appropriate algorithms to control the error rate of interest. Generally they will not select an algorithm that sounds “superior” but controls the wrong error rate. Since the purpose of the paper is to provide the PC algorithm with the control over the FDR, we assume in this paper that the FDR has been selected as the error rate of interest, and selecting the error rate of interest is out of the scope of the paper.

### 4.1.1 More Recent Related Works

After this paper had been accepted for publication, we still kept tracking the advancement in the FDR control in network structure learning, and noticed that two recent conference papers also addressed the problem. To keep the review on related works up-to-date, we add this subsection to discuss the two conference papers.

Listgarten and Heckerman in [20] investigated the problem with both Bayesian and classical approaches. The Bayesian approach, as normally regarded, estimates the FDR through posterior probabilities. The classical approach, quite different from the method proposed in this paper, employs permutation to estimate the number of spurious edges. The advantage of this permutation approach is that it is generally applicable to any structure learning algorithms. On the other hand, it also needs repeating the structure learning at least dozens of times to achieve a robust estimation, which scales up the total computation time. Although its implementation in the paper is from intuition, lacking concrete theoretical justification, based on a fixed order of nodes, and leading to over conservative results, it is still very interesting and worth further development.

In a paper [37] published four months after the submission of this paper, Tsamardinos
4.2 Controlling FDR with PC Algorithm

and Brown proposed an estimation of the upper bound of FDR. The method in [37] and the PC\textsubscript{fdr} algorithm here share certain similarities, but yet technically different. Both are rooted in Proposition 1 and both use an FDR-control procedure to correct p-values. Unlike the PC\textsubscript{fdr} algorithm, the method in [37] does not embed the FDR-control procedure in the structure learning procedure, but separates it as an afterward error-rate estimation. Consequently, it does not allow users to specify the desired FDR in advance. Its asymptotic performance is not discussed in [37].

The remainder of the paper is organized as follows. In Section 4.2 we review the PC algorithm, present the FDR-embedded PC algorithm, prove its asymptotic performance, and analyze its computational complexity. In Section 4.3 we evaluate the proposed method with simulated data, and demonstrate its real world applications to learning functional connectivity networks between brain regions using functional-magnetic-resonance-imaging (fMRI). Finally, we discuss the advantages and limitations of the proposed method in Section 4.4.

4.2 Controlling FDR with PC Algorithm

In this section, we first briefly introduce Bayesian networks and review the PC algorithm. Then, we expatiate on the FDR-embedded PC algorithm and its heuristic modification, prove their asymptotic performances, and analyze their computational complexity. At the end, we discuss other possible ideas of embedding FDR control into the PC algorithm.
4.2. Controlling FDR with PC Algorithm

4.2.1 Notations and Preliminaries

To assist the reading, notations frequently used in the paper are listed as follows:

- \( a, b, \ldots \) : vertices
- \( X_a, X_b, \ldots \) : variables respectively represented by vertices \( a, b, \ldots \)
- \( A, B, \ldots \) : vertex sets
- \( X_A, X_B, \ldots \) : variable sets respectively represented by vertex sets \( A, B, \ldots \)
- \( V \) : the vertex set of a graph
- \( N = |V| \) : the number of vertices of a graph
- \( a \rightarrow b \) : a directed edge or an ordered pair of vertices
- \( a \sim b \) : an undirected edge, or an unordered pair of vertices
- \( E \) : a set of directed edges
- \( E^\sim \) : the undirected edges derived from \( E \), i.e. \( \{ a \sim b | a \rightarrow b \text{ or } b \rightarrow a \in E \} \)
- \( G = (V, E) \) : a directed graph composed of vertices in \( V \) and edges in \( E \)
- \( G^\sim = (V, E^\sim) \) : the skeleton of a directed graph \( G = (V, E) \)
- \( \text{adj}(a, G) \) : vertices adjacent to \( a \) in graph \( G \), i.e. \( \{ b | a \rightarrow b \text{ or } b \rightarrow a \in E \} \)
- \( \text{adj}(a, G^\sim) \) : vertices adjacent to \( a \) in graph \( G^\sim \), i.e. \( \{ b | a \sim b \in E^\sim \} \)
- \( a \perp b | C \) : vertices \( a \) and \( b \) are d-separated by vertex set \( C \)
- \( X_a \perp X_b | X_C \) : \( X_a \) and \( X_b \) are conditional independent given \( X_C \)
- \( p_{a \perp b | C} \) : the p-value of testing \( X_a \perp X_b | X_C \)

A Bayesian network encodes a set of conditional-independence relationships with a DAG \( G = (V, E) \) according to the directed Markov property defined as follows.

**Definition 1** the Directed Markov Property: if \( A \) and \( B \) are d-separated by \( C \) where \( A, B \) and \( C \) are three disjoint sets of vertices, then \( X_A \) and \( X_B \) are conditionally independent given \( X_C \), i.e. \( P(X_A, X_B | X_C) = P(X_A | X_C) P(X_B | X_C) \). [see [18], pages 46–53]

The concept of d-separation [see [18] page 48] is defined as follows. A chain between two vertices \( a \) and \( b \) is a sequence \( a = a_0, a_1, \ldots, a_n = b \) of distinct vertices such that \( a_{i-1} \sim a_i \in E^\sim \) for all \( i=1, \ldots, n \). Vertex \( b \) is a descendant of vertex \( a \) if and only if there is a sequence \( a = a_0, a_1, \ldots, a_n = b \) of distinct vertices such that \( a_{i-1} \rightarrow a_i \in E \) for
4.2. Controlling FDR with PC Algorithm

all \( i = 1, \ldots, n \). For three disjoint subsets \( A, B \) and \( C \subseteq V \), \( C \) d-separates \( A \) and \( B \) if and only if any chain \( \pi \) between \( \forall a \in A \) and \( \forall b \in B \) contains a vertex \( \gamma \in \pi \) such that either

- arrows of \( \pi \) do not meet head-to-head at \( \gamma \) and \( \gamma \in C \), or
- arrows of \( \pi \) meet head-to-head at \( \gamma \) and \( \gamma \) is neither in \( C \) nor has any descendants in \( C \).

Moreover, a probability distribution \( P \) is faithful to a DAG \( G \) [see 32, pages 13 and 81] if all and only the conditional-independence relationships derived from \( P \) are encoded by \( G \). In general, a probability distribution may possess other independence relationships besides those encoded by a DAG.

It should be pointed out that there are often several different DAGs encoding the same set of conditional-independence relationships and they are called an equivalence class of DAGs. An equivalence class can be uniquely represented by a completed acyclic partially directed graph (CPDAG) (also called the essential graph in the literature) that has the same skeleton as a DAG does except that some edges are not directed [see 3].

4.2.2 PC Algorithm

If a probability distribution \( P \) is faithful to a DAG \( G \), then the PC algorithm [see 32, pages 84–89] is able to recover the equivalence class of the DAG \( G \), given the set of the conditional-independence relationships. In general, a probability distribution may include other independence relationships besides those encoded by a DAG. The faithfulness assumption assures that the independence relationships can be perfectly encoded by a DAG. In practice, the information on conditional independence is usually unknown but extracted from data with statistical hypothesis testing. If the \( p \)-value of testing a hypothesis \( X_a \perp X_b | X_C \) is lower than a user-specified level \( \alpha \) (conventionally 5% or 1%), then
the hypothesis of conditional independence is rejected while the hypothesis of conditional dependence \(X_a \not\perp X_b|X_C\) is accepted.

The first step of the PC algorithm is to construct an undirected graph \(G^\sim\) whose edge directions will later be further determined with other steps, while the skeleton is kept the same as that of \(G^\sim\). Since we restrict ourselves to the error rate of the skeleton, here we only present in Algorithm 1 the first step of the PC algorithm, as implemented in software Tetrad version 4.3.8 (see http://www.phil.cmu.edu/projects/tetrad), and refer to it as the PC-skeleton algorithm.

**Algorithm 1** PC-skeleton

**Input:** the data \(X_V\) generated from a probability distribution faithful to a DAG \(G_{true}\), and the significance level \(\alpha\) for every statistical test of conditional independence

**Output:** the recovered skeleton \(G^\sim\)

1. Form the complete undirected graph \(G^\sim\) on the vertex set \(V\).
2. Let depth \(d = 0\).
3. repeat
   4. for each ordered pair of adjacent vertices \(a\) and \(b\) in \(G^\sim\), i.e. \(a \sim b \in E^\sim\) do
      5. if |\(\text{adj}(a, G^\sim) \setminus \{b\}\)| \(\geq d\), then
         6. for each subset \(C \subseteq \text{adj}(a, G^\sim) \setminus \{b\}\) and |\(C\)| = \(d\) do
            7. Test hypothesis \(X_a \perp\!\!\!\perp X_b|X_C\) and calculate the p-value \(p_{a\perp\!\!\!\perp b|C}\).
            8. if \(p_{a\perp\!\!\!\perp b|C} \leq \alpha\), then
              9. Remove edge \(a \sim b\) from \(G^\sim\).
              10. Update \(G^\sim\) and \(E^\sim\).
              11. break the for loop at line 6
         end if
      end for
   end if
end for
14. end if
15. end for
16. Let \(d = d + 1\).
17. until |\(\text{adj}(a, G^\sim) \setminus \{b\}\)| \(< d\) for every ordered pair of adjacent vertices \(a\) and \(b\) in \(G^\sim\).

The theoretical foundation of the PC-skeleton algorithm is **Proposition 1**: if two vertices \(a\) and \(b\) are not adjacent in a DAG \(G\), then there is a set of other vertices \(C\) that either all are neighbors of \(a\) or all are neighbors of \(b\) such that \(C\) d-separates \(a\) and \(b\), or equivalently, \(X_a \perp\!\!\!\perp X_b|X_C\), according to the directed Markov property. Since two adjacent
vertices are not d-separated by any set of other vertices (according to the directed Markov property), Proposition 1 implies that $a$ and $b$ are not adjacent if and only if there is a d-separating $C$ in either neighbors of $a$ or neighbors of $b$. Readers should notice that the proposition does not imply that $a$ and $b$ are d-separated only by such a $C$, but just guarantees that a d-separating $C$ can be found in either neighbors of $a$ or neighbors $b$.

**Proposition 1** *If vertices $a$ and $b$ are not adjacent in a DAG $G$, then there is a set of vertices $C$ which is either a subset of $\text{adj}(a,G)\backslash\{b\}$ or a subset of $\text{adj}(b,G)\backslash\{a\}$ such that $C$ d-separates $a$ and $b$ in $G$. This proposition is a corollary of Lemma 5.1.1 on page 411 of book Causation, Prediction, and Search [32].*

The logic of the PC-skeleton algorithm is as follows, under the assumption of perfect judgment on conditional independence. The most straightforward application of Proposition 1 to structure learning is to exhaustively search all the possible neighbors of $a$ and $b$ to verify whether there is such a d-separating $C$ to disconnect $a$ and $b$. Since the possible neighbors of $a$ and $b$ are unknown, to guarantee the detection of such a d-separating $C$, all the vertices other than $a$ and $b$ should be searched as possible neighbors of $a$ and $b$. However, such a straightforward application is very inefficient because it probably searches many unnecessary combinations of vertices by considering all the vertices other than $a$ and $b$ as their possible neighbors, especially when the true DAG is sparse. The PC-skeleton algorithm searches more efficiently, by keeping updating the possible neighbors of a vertex once some previously-considered possible neighbors have been found actually not connected with the vertex. Starting with a fully connected undirected graph $G$ (step 1), the algorithm searches for the d-separating $C$ progressively by increasing the size of $C$, i.e. the number of conditional variables, from zero (step 2) with the step size of one (step 16). Given the size of $C$, the search is performed for every vertex pair $a$ and $b$ (step 4). Once a $C$ d-separating $a$ and $b$ is found (step 8), $a$ and $b$ are disconnected (step 9), and the neighbors of $a$ and $b$ are updated (step 10). In the algorithm, $G$ is continually updated, so $\text{adj}(a,G)$ is also constantly updated as the
4.2. Controlling FDR with PC Algorithm

The algorithm progresses. The algorithm stops when all the subsets of the current neighbors of each vertex have been examined (step 17). The Tetrad implementation of the PC-skeleton algorithm examines an edge \( a \sim b \) as two ordered pairs \((a, b)\) and \((b, a)\) (step 4), each time searching for the d-separating \( C \) in the neighbors of the first element of the pair (step 6). In this way, both the neighbors of \( a \) and the neighbors of \( b \) are explored.

The accuracy of the PC-skeleton algorithm depends on the discriminability of the statistical test of conditional independence. If the test can perfectly distinguish dependence from independence, then the algorithm can correctly recover the true underlying skeleton, as proved by Spirtes et al. \[32\], pages 410–412. The outline of the proof is as follows. First, all the true edges will be recovered because an adjacent vertex pair \( a \sim b \) is not d-separated by any vertex set \( C \) that excludes \( a \) and \( b \). Second, if the edge between a non-adjacent vertex pair \( a \) and \( b \) has not been removed, subsets of either \( \text{adj}(a, G) \setminus \{b\} \) or \( \text{adj}(b, G) \setminus \{a\} \) will be searched until the \( C \) that d-separates \( a \) and \( b \) according to Proposition 1 is found, and consequently the edge between \( a \) and \( b \) will be removed. If the judgments on conditional independence and conditional dependence are imperfect, the PC-skeleton algorithm is unstable. If an edge is mistakenly removed from the graph in the early stage of the algorithm, then other edges which are not in the true graph may be included in the graph [see \[32\], page 87].

4.2.3 False Discovery Rate

In a statistical decision process, there are basically two sources of errors: the type I errors, \( i.e. \) falsely rejecting negative hypotheses when they are actually true; and the type II errors, \( i.e. \) falsely accepting negative hypotheses when their alternative, the positive hypotheses are actually true. The FDR [see \[4\]] is a criterion to assess the errors when multiple hypotheses are simultaneously tested. It is the expected ratio of the number of falsely claimed positive results to that of all those claimed to be positive, as defined in Table 4.2. A variant of the FDR, the positive false discovery rate (pFDR), defined as in
4.2. Controlling FDR with PC Algorithm

<table>
<thead>
<tr>
<th>Test Results</th>
<th>Truth</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Negative</td>
</tr>
<tr>
<td>Negative</td>
<td>TN (true negative)</td>
</tr>
<tr>
<td>Positive</td>
<td>FP (false positive)</td>
</tr>
<tr>
<td>Total</td>
<td>$T_1$</td>
</tr>
</tbody>
</table>

Table 4.1: Results of multiple hypothesis testing, categorized according to the claimed results and the truth.

<table>
<thead>
<tr>
<th>Full Name</th>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>False Discovery Rate</td>
<td>FDR</td>
<td>$E(FP/R_2)$ (See note *)</td>
</tr>
<tr>
<td>Positive False Discovery Rate</td>
<td>pFDR</td>
<td>$E(FP/R_2</td>
</tr>
<tr>
<td>Family-Wise Error Rate</td>
<td>FWER</td>
<td>$P(FP \geq 1)$</td>
</tr>
<tr>
<td>Type I Error Rate (False Positive Rate)</td>
<td>$\alpha$</td>
<td>$E(FP/T_1)$</td>
</tr>
<tr>
<td>Specificity (True Negative Rate)</td>
<td>$1 - \alpha$</td>
<td>$E(TN/T_1)$</td>
</tr>
<tr>
<td>Type II Error Rate (False Negative Rate)</td>
<td>$\beta$</td>
<td>$E(FN/T_2)$</td>
</tr>
<tr>
<td>Power (Sensitivity, True Positive Rate)</td>
<td>$1 - \beta$</td>
<td>$E(TP/T_2)$</td>
</tr>
</tbody>
</table>

Table 4.2: Criteria for multiple hypothesis testing. Here $E(x)$ means the expected value of $x$, and $P(A)$ means the probability of event $A$. Please refer to Table 4.1 for related notations. * If $R_2 = 0$, FP/$R_2$ is defined to be 0.

Table 4.2 was proposed by Storey [34]. Clearly, pFDR = FDR / $P(R_2 > 0)$, so the two measures will be similar if $P(R_2 > 0)$ is close to 1, and quite different otherwise.

The FDR is a reasonable criterion when researchers expect the “discovered” results are trustful and dependable in afterward studies. For example, in a pilot study, we are selecting candidate genes for a genetic research on Parkinson’s disease. Because of the limited funding, we can only study a limited number of genes in the afterward genetic research. Thus, when selecting candidate genes in the pilot study, we hope that 95% of the selected candidate genes are truly associated with the disease. In this case, the FDR is chosen as the error rate of interest and should be controlled under 5%. Since similar situations are quite common in research practices, the FDR has been widely adopted in many research fields such as bioinformatics and neuroimaging.

In the context of learning the skeleton of a DAG, a negative hypothesis could be that a connection does not exist in the DAG, and a positive hypothesis could be that the connection exists. The FDR is the expected proportion of the falsely discovered
connections to all those discovered. Learning network structures may face scenarios similar to the aforementioned pilot study, but the FDR control has not yet received adequate attention in structure learning.

Benjamini and Yekutieli [4] have proved that, when the test statistics have positive regression dependency on each of the test statistics corresponding to the true negative hypotheses, the FDR can be controlled under a user-specified level $q$ by Algorithm 2. In other cases of dependency, the FDR can be controlled with a simple conservative modification of the procedure by replacing $H^*$ in Eq. (4.1) with $H(1 + 1/2, \ldots, +1/H)$. Storey [34] has provided algorithms to control the pFDR for independent test statistics. For a review and comparison of more FDR methods, please refer to Qian and Huang [28]’s work. It should be noted that the FDR procedures do not control the realized FDR of a trial under $q$, but control the expected value of the error rate when the procedures are repetitively applied to randomly sampled data.

**Algorithm 2** FDR-stepup  
**Input:** a set of $p$-values $\{p_i\}_{i=1, \ldots, H}$, and the threshold of the FDR $q$  
**Output:** the set of rejected null hypotheses  

1: Sort the $p$-values of $H$ hypothesis tests in the ascendant order as $p(1) \leq \ldots \leq p(H)$.  
2: Let $i = H$, and $H^* = H$ (or $H^* = H(1 + 1/2, \ldots, +1/H)$, depending on the assumption of the dependency among the test statistics).  
3: while \[
\frac{H^*}{i} p(i) > q \quad \text{and} \quad i > 0, \tag{4.1}
\]  
   do  
   4: Let $i = i - 1$.  
   5: end while  
6: Reject the null hypotheses associated with $p(1), \ldots, p(i)$, and accept the null hypotheses associated with $p(i+1), \ldots, p(H)$.

Besides the FDR and the pFDR, other criteria, as listed in Table 4.2, can also be applied to assess the uncertainty of multiple hypothesis testing. The type I error rate is the expected ratio of the type I errors to all the negative hypotheses that are actually true. The type II error rate is the expected ratio of the type II errors to all the positive
hypotheses that are actually true. The family-wise error rate is the probability that at least one of the accepted positive hypotheses are actually wrong. Generally, there are not mathematically or technically superior relationships among these error-rate criteria. Please refer to Appendix 4.7 for examples of typical research scenarios where each particular error rate is favoured.

Controlling both the type I and the type II error rates under a conventional level (such as $\alpha < 5\%$ or $1\%$ and $\beta < 10\%$ or $5\%$) does not necessarily curb the FDR at a desired level. As shown in Eq. (4.2), if $FP/T_1$ and $FN/T_2$ are fixed and positive, $FP/R_2$ approaches 1 when $T_2/T_1$ is small enough. This is the case of sparse networks where the number of existing connections $T_2$ is much smaller than the number of non-existing connections $T_1$.

$$\frac{FP}{R_2} = \frac{FP}{T_1} + \frac{FN}{T_2} \cdot \frac{T_2}{T_1}.$$  (4.2)

4.2.4 PC Algorithm with FDR

Steps 8–12 of the PC-skeleton algorithm control the type I error rate of each statistical test of conditional independence individually below a pre-defined level $\alpha$, so the algorithm can not explicitly control the FDR. We propose embedding an FDR-control procedure into the algorithm to curb the error rate of the learned skeleton. The FDR-control procedure collectively considers the hypothesis tests related to the existence of multiple edges, correcting the effect of multiple hypothesis testing. The proposed method is described in Algorithm 3 and we name it as the PC\_fdr-skeleton algorithm. Similar to the PC-skeleton algorithm, $G^\sim$, $\text{adj}(a, G^\sim)$ and $E^\sim$ are constantly updated as the algorithm progresses.

The PC\_fdr-skeleton and the PC-skeleton algorithms share the same search strategy, but differ on the judgment of conditional independence. The same as the PC-skeleton algorithm, the PC\_fdr-skeleton algorithm increases $d$, the number of conditional variables,
4.2. Controlling FDR with PC Algorithm

Algorithm 3 PC$_{\text{fdr}}$-skeleton

**Input:** the data $X_V$ generated from a probability distribution faithful to a DAG $G_{\text{true}}$, and the FDR level $q$ for the discovered skeleton

**Output:** the recovered skeleton $G^\sim$

1: Form the complete undirected graph $G^\sim$ on the vertex set $V$
2: Initialize the maximum $p$-values associated with edges as $P_{\text{max}} = \{p_{a\sim b}^{\text{max}} = -1\}_{a \neq b}$.
3: Let depth $d = 0$.
4: repeat
5: for each ordered pair of adjacent vertices $a$ and $b$ in $G^\sim$, i.e. $a \sim b \in E^\sim$ do
6: if $|\text{adj}(a, G^\sim) \backslash \{b\}| \geq d$, then
7: for each subset $C \subseteq \text{adj}(a, G^\sim) \backslash \{b\}$ and $|C| = d$ do
8: Test hypothesis $X_a \perp X_b | X_C$ and calculate the $p$-value $p_{a \perp b | C}$.
9: if $p_{a \perp b | C} > p_{a \sim b}^{\text{max}}$, then
10: Let $p_{a \sim b}^{\text{max}} = p_{a \perp b | C}$.
11: if every element of $P_{\text{max}}$ has been assigned a valid $p$-value by step 10 then
12: Run the FDR procedure, Algorithm 2 with $P_{\text{max}}$ and $q$ as the input.
13: if the non-existence of certain edges are accepted, then
14: Remove these edges from $G^\sim$.
15: Update $G^\sim$ and $E^\sim$.
16: if $a \sim b$ is removed, then
17: break the for loop at line 7
18: end if
19: end if
20: end if
21: end if
22: end for
23: end if
24: end for
25: Let $d = d + 1$.
26: until $|\text{adj}(a, G^\sim) \backslash \{b\}| < d$ for every ordered pair of adjacent vertices $a$ and $b$ in $G^\sim$.

* A heuristic modification at step 15 of the algorithm is to remove from $P_{\text{max}}$ the $p_{a \sim b}^{\text{max}}$'s whose associated edges have been deleted from $G^\sim$ at step 14, i.e. to update $P_{\text{max}}$ as $P_{\text{max}} = \{p_{a \sim b}^{\text{max}} \}_{a \sim b \in E^\sim}$ right after updating $E^\sim$ at step 15. This heuristic modification is named as the PC$_{\text{fdr}}$-skeleton algorithm.
4.2. Controlling FDR with PC Algorithm

from zero (step 3) with the step size of one (step 25), and also keeps updating the neighbors of vertices (steps 14 and 15) when some previously-considered possible neighbors have been considered not connected (step 13). The PC\textsubscript{fdr}-skeleton algorithm differs from the PC-skeleton algorithm on the inference of d-separation, with its steps 11–20 replacing steps 8–12 of the PC-skeleton algorithm. In the PC-skeleton algorithm, two vertices are regarded as d-separated once the conditional-independence test between them yields a \( p \)-value larger than the pre-defined significant level \( \alpha \). In this way, the type I error of each statistical test is controlled separately, without consideration of the effect of multiple hypothesis testing. The PC\textsubscript{fdr}-skeleton algorithm records in \( p_{\text{max}}^{a\sim b} \) the up-to-date maximum \( p \)-value associated with an edge \( a \sim b \) (steps 9 and 10), and progressively removes those edges whose non-existence is accepted by the FDR procedure (step 12), with \( P_{\text{max}}^{\sim} = \{p_{a\sim b}^{\max}\}_{a\neq b} \) and the pre-defined FDR level \( q \) being the input. The FDR procedure, Algorithm 2, is invoked at step 12 either immediately after every element of \( P_{\text{max}}^{\sim} \) has been assigned a valid \( p \)-value for the first time, or later once any element of \( P_{\text{max}}^{\sim} \) is updated.

The \( p_{a\sim b}^{\max} \) is the upper bound of the \( p \)-value of testing the hypothesis that \( a \) and \( b \) are d-separated by at least one of the vertex sets \( C \) searched in step 7. According to the directed Markov property, \( a \) and \( b \) are not adjacent if and only if there is a set of vertices \( C \subseteq V\setminus\{a,b\} \) d-separating \( a \) and \( b \). As the algorithm progresses, the d-separations between \( a \) and \( b \) by vertex sets \( C_1, \ldots, C_K \subseteq V\setminus\{a,b\} \) are tested respectively, and consequently a sequence of \( p \)-values \( p_1, \ldots, p_K \) are calculated. If we use \( p_{a\sim b}^{\max} = \max_{i=1}^{K} p_i \) as the statistic to test the negative hypothesis that there is, though unknown, a \( C_j \) among \( C_1, \ldots, C_K \) d-separating \( a \) and \( b \), then due to

\[
P(p_{a\sim b}^{\max} \leq p) = P(p_i \leq p \text{ for all } i = 1, \ldots, K) \leq P(p_j \leq p) = p,
\]

\( p_{a\sim b}^{\max} \) is the upper bound of the \( p \)-value of testing the negative hypothesis. Eq. (4.3) also clearly shows that the PC-skeleton algorithm controls the type I error rate of the negative
hypothesis, since its step is equivalent to “if $p_{a\sim b}^{max} < \alpha$, then . . . ” if $p_{a\sim b}^{max}$ is recorded in the PC-skeleton algorithm.

The statistical tests performed at step of the PC$_{fdr}$-skeleton algorithm generally are not independent with each other, since the variables involved in two hypotheses of conditional independence may overlap. For example, conditional-independence relationships $a \perp b_1|C$ and $a \perp b_2|C$ both involve $a$ and $C$. It is very difficult to prove whether elements of $P^{max}$ have positive regression dependency or not, so rigorously the conservative modification of Algorithm[2] should be applied at step. However, since $p_{a\sim b}^{max}$ is probably a loose upper bound of the $p$-value of testing $a \neq b$, in practice we simply apply the FDR procedure that is correct for positive regression dependency.

It should be noted that different from step of the PC-skeleton algorithm, step of the PC$_{fdr}$-skeleton algorithm may remove edges other than just $a \sim b$, because the decisions on other edges can be affected by the updating of $p_{a\sim b}^{max}$.

A heuristic modification of the PC$_{fdr}$-skeleton algorithm is to remove $p_{a\sim b}^{max}$ from $P^{max}$ once edge $a \sim b$ has been deleted from $G^\sim$ at step. We name this modified version as the PC$_{fdr*}$-skeleton algorithm. In the PC$_{fdr}$-skeleton algorithm, $p_{a\sim b}^{max}$ is still recorded in $P^{max}$ and input to the FDR procedure after the edge $a \sim b$ has been removed. This guarantees that the algorithm can asymptotically keep the FDR under the user-specified level $q$ (see Section[4.2.5]). The motivation of the heuristic modification is that if an edge has been eliminated, then it should not be considered in the FDR procedure any longer. Though we cannot theoretically prove the asymptotic performance of the heuristic modification in the sense of controlling the FDR, it is shown to control the FDR closely around the user-specified level in our empirical experiments and gain more detection power than that of the PC$_{fdr}$-skeleton algorithm (see Section[4.3]).
4.2. Controlling FDR with PC Algorithm

4.2.5 Asymptotic Performance

Here we prove that the PC-fdr-skeleton algorithm is able to control the FDR under a user-specified level \( q \) \((q > 0)\) at the limit of large sample sizes if the following assumptions are satisfied:

(A1) The probability distribution \( P \) is faithful to a DAG \( G_{\text{true}} \).

(A2) The number of vertices is fixed.

(A3) Given a fixed significant level of testing conditional-independence relationships, the power of detecting conditional-dependence relationships with statistical tests approaches 1 at the limit of large sample sizes. (For the definition of power in hypothesis testing, please refer to Table 4.2.)

Assumption (A1) is generally assumed when graphical models are applied, although it restricts the probability distribution \( P \) to a certain class. Assumption (A2) is usually implicitly stated, but here we explicitly emphasize it because it simplifies the proof. Assumption (A3) may seem demanding, but actually it can be easily satisfied by standard statistical tests, such as the likelihood-ratio test introduced by Neyman and Pearson \[24\], if the data are identically and independently sampled. Two statistical tests that satisfy Assumption (A3) are listed in Appendix 4.6.

The detection power and the FDR of the PC-fdr-skeleton algorithm and its heuristic modification at the limit of large sample sizes are elucidated in Theorems 1 and 2. The detailed proofs are provided in Appendix 4.5.

**Theorem 1** Assuming (A[1]), (A[2]) and (A[3]), both the PC-fdr-skeleton algorithm and its heuristic modification, the PC-fdr*-skeleton algorithm, are able to recover all the true connections with probability one as the sample size approaches infinity:

\[
\lim_{m \to \infty} P(E_{\text{true}} \subseteq E^\sim) = 1,
\]
where \( \tilde{E}_{true} \) denotes the set of the undirected edges derived from the true DAG \( G_{true} \), \( \tilde{E} \) denotes the set of the undirected edges recovered with the algorithms, and \( m \) denotes the sample size.

**Theorem 2** Assuming (A1), (A2) and (A3), the FDR of the undirected edges recovered with the PC\(_{fdr}\)-skeleton algorithm approaches a value not larger than the user-specified level \( q \) as the sample size \( m \) approaches infinity:

\[
\limsup_{m \to \infty} \text{FDR}(\tilde{E}, \tilde{E}_{true}) \leq q,
\]

where FDR\( (\tilde{E}, \tilde{E}_{true}) \) is defined as

\[
\text{FDR}(\tilde{E}, \tilde{E}_{true}) = \frac{\mathbb{E}[|\tilde{E}\setminus \tilde{E}_{true}|]}{|\tilde{E}|},
\]

Define \( \frac{|\tilde{E}\setminus \tilde{E}_{true}|}{|\tilde{E}|} = 0 \), if \( |\tilde{E}| = 0 \).

### 4.2.6 Computational Complexity

The PC\(_{fdr}\)-skeleton algorithm spends most of its computation on performing statistical tests of conditional independence at step 8 and controlling the FDR at step 12. Since steps 13 to 19 of the PC\(_{fdr}\)-skeleton algorithm play a role similar to steps 8 to 12 of the PC-skeleton algorithm do, and all the other parts of both algorithms employ the same search strategy, the computation spent by the PC\(_{fdr}\)-skeleton algorithm on statistical tests has the same complexity as that by the PC-skeleton algorithm. The only extra computational cost of the PC\(_{fdr}\)-skeleton algorithm is at step 12 for controlling the FDR.

The computational complexity of the search strategy employed by the PC algorithm has been studied by Kalisch and Bühlmann [15] and see Spirtes et al. [32], pages 85–87. Here to make the paper self-contained, we briefly summarize the results as follows. It is difficult to analyze the complexity exactly, but if the algorithm stops at the depth
4.2. Controlling FDR with PC Algorithm

d = d_{\text{max}}, then the number of conditional-independence tests required is bounded by

\[ T = 2C_N^2 \sum_{d=0}^{d_{\text{max}}} C_{N-2}^d, \]

where \( N \) is the number of vertices, \( C_N^2 \) is the number of combinations of choosing 2 un-ordered and distinct elements from \( N \) elements, and similarly \( C_{N-2}^2 \) is the number of combinations of choosing from \( N-2 \) elements. In the worst case that \( d_{\text{max}} = N-2 \), the complexity is bounded by \( 2C_N^2 2^{N-2} \). The bound usually is very loose, because it assumes that no edge has been removed until \( d = d_{\text{max}} \). In real world applications, the algorithm is very fast for sparse networks.

The computational complexity of the FDR procedure, Algorithm 2, generally is \( O(H \log(H) + H) = O(H \log(H)) \) where \( H = C_N^2 \) is the number of input \( p \)-values. The sorting at step 1 costs \( H \log(H) \) and the “while” loop from step 3 to step 5 repeats \( H \) times at most. However, if the sorted \( P^{\text{max}} \) is recorded during the computation, each time when an element of \( P^{\text{max}} \) is updated at step 10 of the PC_fdr-skeleton algorithm, the complexity of keeping the updated \( P^{\text{max}} \) sorted is only \( O(H) \). With this optimization, the complexity of the FDR-control procedure is \( O(H \log(H)) \) at its first operation, and is \( O(H) \) later. The FDR procedure is invoked only when \( p_{a \perp b | C} > p_{a \sim b}^{\text{max}} \). In the worst case that \( p_{a \perp b | C} \) is always larger than \( p_{a \sim b}^{\text{max}} \), the complexity of the computation spent on the FDR control in total is bounded by \( O(C_N^2 \log(C_N^2) + TC_N^2) = O(N^2 \log(N) + TN^2) \) where \( T \) is the number of performed conditional-independence tests. This is a very loose bound because it is rare that \( p_{a \perp b | C} \) is always larger than \( p_{a \sim b}^{\text{max}} \).

The computational complexity of the heuristic modification, the PC_fdr*-skeleton algorithm, is the same as that of the PC_fdr-skeleton algorithm, since they share the same search strategy and both employ the FDR procedure. In the PC_fdr*-skeleton algorithm, the size of \( P^{\text{max}} \) keeps decreasing as the algorithm progresses, so each operation of the FDR procedure is more efficient. However, since the PC_fdr*-skeleton algorithm adjusts the
effect of multiple hypothesis testing less conservatively, it may remove less edges than the PC\(_{\text{fdr}}\)-skeleton algorithm does, and invokes more conditional-independence tests. Nevertheless, their complexity is bounded by the same limit in the worst case.

It is unfair to directly compare the computational time of the PC\(_{\text{fdr}}\)-skeleton algorithm against that of the PC-skeleton algorithm, because if the \(q\) of the former is set at the same value as the \(\alpha\) of the latter, the former will remove more edges and perform much less statistical tests, due to its more stringent control over the type I error rate. A reasonable way is to compare the time spent on the FDR control at step 12 against that on conditional-independence tests at step 8 in each run of the PC\(_{\text{fdr}}\)-skeleton algorithm. If \(P^{\text{max}}\) is kept sorted during the learning process as aforementioned, then each time (except the first time) the FDR procedure just needs linear computation time (referring to the size of \(P^{\text{max}}\)) with simple operations such as division and comparing two numerical values. Thus, we suspect that the FDR procedure will not contribute much to the total computation time of the structure learning. In our simulation study in Section 4.3.1, the extra computation added by the FDR control was only a tiny portion, less than 0.5%, to that spent on testing conditional independence, performed with the Cochran-Mantel-Haenszel (CMH) test [see\(^1\) pages 231–232], as shown in Tables 4.3 and 4.4.

4.2.7 Miscellaneous Discussions

An intuitive and attracting idea of adapting the PC-skeleton algorithm to the FDR control is to “smartly” determine such an appropriate threshold of the type I error rate \(\alpha\) that will let the errors be controlled at the pre-defined FDR level \(q\). Given a particular problem, it is very likely that the FDR of the graphs learned by the PC-skeleton algorithm is an monotonically increasing function of the pre-defined threshold \(\alpha\) of the type I error rate. If this hypothesis is true, then there is a one-to-one mapping between \(\alpha\) and \(q\) for the particular problem. Though we cannot prove this hypothesis rigorously, the following argument may be enlightening. Instead of directly focusing on FDR =
4.2. Controlling FDR with PC Algorithm

$E(\frac{FP}{R_2})$ (see Table 4.2), the expected ratio of the number of false positives (FP) to the number of accepted positive hypotheses ($R_2$), we first focus on $\frac{E(FP)}{E(R_2)}$, the ratio of the expected number of false positives to the expected number of accepted positive hypotheses, since the latter is easier to link with the type I error rate according to Eq. (4.2), as shown in Eq. (4.4),

$$\frac{E(FP)}{E(R_2)} = \frac{E\left(\frac{FP}{T_1}\right)}{E\left(\frac{FP}{T_1} + (1 - E(\frac{FN}{T_2}))\frac{T_2}{T_1}\right)} = \frac{E\left(\frac{FP}{T_1}\right)}{E\left(\frac{FP}{T_1} + (1 - E(\frac{FN}{T_2}))\right)\frac{T_2}{T_1}} = \frac{\alpha}{\alpha + (1 - \beta)\frac{T_2}{T_1}}.$$  

(4.4)

where $\alpha$ and $\beta$ are the type I error rate and the type II error rate respectively. A sufficient condition for $\frac{E(FP)}{E(R_2)}$ being a monotonically increasing function of the type I error rate includes (I) $(1 - \beta)/\alpha > \partial(1 - \beta)/\partial\alpha$, (II) $T_1 > 0$ and (III) $T_2 > 0$, where $\partial(1 - \beta)/\partial\alpha$ is the derivative of $(1 - \beta)$ over $\alpha$. If $(1 - \beta)$, regarded as a function of $\alpha$, is a concave curve from $(0, 0)$ to $(1,1)$, then condition (I) is satisfied. Recall that $(1 - \beta)$ versus $\alpha$ actually is the receiver operating characteristic (ROC) curve, and that with an appropriate statistic the ROC curve of a hypothesis test is usually a concave curve from $(0, 0)$ to $(1,1)$, we speculate that condition (I) is not difficult to satisfy. With the other two mild conditions (II) $T_1 > 0$ and (III) $T_2 > 0$, we could expect that $\frac{E(FP)}{E(R_2)}$ is a monotonically increasing function of $\alpha$. $\frac{E(FP)}{E(R_2)}$ is the ratio of the expected values of two random variables, while $E(\frac{FP}{R_2})$ is the expected value of the ratio of two random variables. Generally, there is not a monotonic relationship between $\frac{E(FP)}{E(R_2)}$ and $E(\frac{FP}{R_2})$. Nevertheless, if the average number of false positives, $E(FP)$, increases proportionally faster than that of the accepted positives, $E(R_2)$, we speculate that under certain conditions, the FDR $= E(\frac{FP}{R_2})$ also increases accordingly. Thus the FDR may be a monotonically increasing function of the threshold $\alpha$ of the type I error rate for the PC-skeleton algorithm.

However, even though the FDR of the PC-skeleton algorithm may decrease as the pre-defined significant level $\alpha$ decreases, the FDR of the PC-skeleton algorithm still cannot
4.2. Controlling FDR with PC Algorithm

be controlled at the user-specified level for general problems by “smartly” choosing an
\( \alpha \) beforehand, but somehow has to be controlled in a slightly different way, such as the
PC\(_{\text{fdr}}\)-skeleton algorithm does. First, the value of such an \( \alpha \) for the FDR control depends
on the true graph, but unfortunately the graph is unknown in problems of structure
learning. According to Eq. (4.2), the realized FDR is a function of the realized type I
and type II error rates, as well as \( T_2/T_1 \), which in the context of structure learning is the
ratio of the number of true connections to the number of non-existing connections. Since
\( T_2/T_1 \) is unknown, such an \( \alpha \) cannot be determined completely in advance without any
information about the true graph, but has to be estimated practically from the observed
data. Secondly, the FDR method we employ is such a method that estimates the \( \alpha \) from
the data to control the FDR of multiple hypothesis testing. The output of the FDR
algorithm is the rejection of those null hypotheses associated with \( p \)-values \( p(1), \ldots, p(i) \)
(see Algorithm 2). Given \( p(1) \leq \ldots \leq p(H) \), the output is equivalent to the rejection of all
those hypotheses whose \( p \)-values are smaller than or equal to \( p(i) \). In other words, it is
equivalent to setting \( \alpha = p(i) \) in the particular multiple hypothesis testing. Thirdly, the
PC\(_{\text{fdr}}\)-skeleton algorithm is a valid solution to combining the FDR method with the PC-
skeleton algorithm. Because the estimation of the \( \alpha \) depends on \( p \)-values, and \( p \)-values
are calculated one by one as the PC-skeleton algorithm progresses with hypothesis tests,
the \( \alpha \) cannot be estimated separately before the PC-skeleton algorithm starts running,
but the estimation has to be embedded within the algorithm, like in the PC\(_{\text{fdr}}\)-skeleton
algorithm.

Another idea on the FDR control in structure learning is a two-stage algorithm. The
first stage is to draft a graph that correctly includes all the existing edges and their
orientations but may also include non-existing edges as well. The second stage is to
select the real parents for each vertex, with the FDR controlled, from the set of potential
parents determined in the first stage. The advantage of this algorithm is that the selection
of real parent vertices in the second stage is completely decoupled from the determination
of edge orientations, because all the parents of each vertex have been correctly connected with the particular vertex in the first stage. However, a few concerns about the algorithm should be noticed before researchers start developing this two-stage algorithm. First, to avoid missing many existing edges in the first stage, a considerable number of non-existing edges may have to be included. To guarantee a perfect protection of the existing edges given any randomly sampled data, the first stage must output a graph whose skeleton is a fully connected graph. The reason for this is that the type I error rate and the type II error rate contradict each other and the latter reaches zero generally when the former approaches one (see Appendix 4.7). Rather than protecting existing edges perfectly, the first stage should trade off between the type I and the type II errors, in favour of keeping the type II error rate low. Second, selecting parent vertices from a set of candidate vertices in the second stage, in certain sense, can be regarded as learning the structure of a sub-graph locally, in which error-rate control remains as a crucial problem. Thus error-rate control is still involved in both of the two stages. Though this two-stage idea may not essentially reduce the problem of the FDR control to an easier task, it may break the big task of simultaneously learning all edges to many local structure-learning tasks.

4.3 Empirical Evaluation

The PC$_{fdr}$-skeleton algorithm and its heuristic modification are evaluated with simulated data sets, in comparison with the PC-skeleton algorithm, in the sense of the FDR, the type I error rate and the power. The PC$_{fdr}$-skeleton and the PC-skeleton algorithms are also applied to two real functional-magnetic-resonance-imaging (fMRI) data sets, to check whether the two algorithms correctly curb the error rates that they are supposed to control in real world applications.
4.3. Empirical Evaluation

4.3.1 Simulation Study

The simulated data sets are generated from eight different DAGs, shown in Figure 4.1, with the number of vertices \( N = 15, 20, 25 \) or 30, and the average degree of vertices \( D = 2 \) or 3. The DAGs are generated as follows:

1. Sample \( \frac{N \times D}{2} \) undirected edges from \( \{a \sim b | a, b \in V \text{ and } a \neq b\} \) with equal probabilities and without replacement to compose an undirected graph \( G_{\text{true}}^{-} \).

2. Generate a random order \( > \) of vertices with permutation.

3. Orientate the edges of \( G^{-} \) according to the order \( > \). If \( a \) is before \( b \) in the order \( > \), then orientate the edge \( a \sim b \) as \( a \rightarrow b \). Denote the orientated graph as a DAG \( G_{\text{true}} \).

For each DAG, we associate its vertices with (conditional) binary probability distributions as follows, to extend it to a Bayesian network.

1. Specify the strength of (conditional) dependence as a parameter \( \delta > 0 \).

2. Randomly assign each vertex \( a \in V \) with a dependence strength \( \delta_a = 0.5\delta \) or \(-0.5\delta\), with equal possibilities.

3. Associate each vertex \( a \in V \) with a logistic regression model

\[
\Delta = \sum_{b \in \text{pa}[a]} X_b \delta_b, \\
P(X_a = 1 | X_{\text{pa}[a]}) = \frac{\exp(\Delta)}{1 + \exp(\Delta)}, \\
P(X_a = -1 | X_{\text{pa}[a]}) = \frac{1}{1 + \exp(\Delta)},
\]

where \( \text{pa}[a] \) denotes the parent vertices of \( a \).
Figure 4.1: DAGs used in the simulation study. $N$ denotes the number of vertices and $D$ denotes the average degree of the vertices. Unshaded vertices are associated with positive dependence strength $0.5\delta$, and shaded ones are associated with negative dependence strength $-0.5\delta$. 

4.3. Empirical Evaluation
4.3. Empirical Evaluation

The parameter $\delta$ reflects the strength of dependence because if the values of all the other parent variables are fixed, the difference between the conditional probabilities of a variable $X_a = 1$ given a parent variable $X_b = 1$ and -1 is

$$\left| \logit[P(X_a = 1|X_b = 1, X_{pa[a]\setminus\{b\}})] - \logit[P(X_a = 1|X_b = -1, X_{pa[a]\setminus\{b\}})] \right| = |2\delta_b| = \delta,$$

where the logit function is defined as $\logit(x) = \log\left(\frac{x}{1-x}\right)$.

Since the accuracy of the PC-skeleton algorithm and its FDR versions is related to the discriminability of the statistical tests, we generated data with different values of $\delta$ ($\delta = 0.5, 0.6, 0.7, 0.8, 0.9$ and $1.0$) to evaluate the algorithms’ performances with different power of detecting conditional dependence. The larger the absolute value of $\delta$ is, the easier the dependence can be detected with statistical tests. Because statistical tests are abstract queries yielding $p$-values about conditional independence for the structure-learning algorithms, the accuracy of the algorithms is not determined by the particular procedure of a statistical test, or a particular family of conditional probability distributions but by the discriminability of the statistical tests. Given a fixed sample size, the stronger the conditional-dependence relationships are, the higher discriminability the statistical tests have. By varying the dependence strength $\delta$ of the binary conditional probability distributions, we have varied the discriminability of the statistical tests, as if by varying the dependence strength of other probability distribution families. To let the readers intuitively understand the dependence strength of these $\delta$ values, we list as follows examples of probability pairs whose logit contrasts are equal to these $\delta$ values:

$$0.5 = \logit(0.5622) - \logit(0.4378),$$

$$0.6 = \logit(0.5744) - \logit(0.4256),$$

$$0.7 = \logit(0.5866) - \logit(0.4134),$$

$$0.8 = \logit(0.5987) - \logit(0.4013),$$

$$0.9 = \logit(0.6106) - \logit(0.3894),$$

$$1.0 = \logit(0.6225) - \logit(0.3775).$$
4.3. Empirical Evaluation

In total, we performed the simulation with 48 Bayesian networks generated with all the combinations of the following parameters:

\[
N = 15, 20, 25, 30;
\]
\[
D = 2, 3;
\]
\[
\delta = 0.5, 0.6, 0.7, 0.8, 0.9, 1.0.
\]

From each Bayesian network, we repetitively generated 50 data sets each of 500 samples to estimate the statistical performances of the algorithms. A non-model-based test, the Cochran-Mantel-Haenszel (CMH) test [see \[1\], pages 231–232], was employed to test conditional independence among random variables. Both the significant level \( \alpha \) of the PC-skeleton algorithm and the FDR level \( q \) of the PC_{fdr}-skeleton algorithm and its heuristic modification were set at 5%.

Figures 4.2, 4.3 and 4.4 respectively show the empirical FDR, power and type I error rate of the algorithms, estimated from the 50 data sets repetitively generated from each Bayesian network, with error bars indicating the 95% confidence intervals of these estimations. The PC_{fdr}-skeleton algorithm controls the FDR under the user-specified level 5% for all the 48 Bayesian networks, and the PC_{fdr*}-skeleton algorithm steadily controls the FDR closely around 5%, while the PC-skeleton algorithm yields the FDR ranging from about 5% to about 35%, and above 15% in many cases, especially for those sparser DAGs with the average degree of vertices \( D = 2 \). The PC_{fdr}-skeleton algorithm is conservative, with the FDR notably lower than the user-specified level, while its heuristic modification controls the FDR more accurately around the user-specified level, although the correctness of the heuristic modification has not been theoretically proved. As the discriminability of the statistical tests increases, the power of all the algorithms approaches 1. When their FDR level \( q \) is set at the same value as the \( \alpha \) of the PC-skeleton algorithm, the PC_{fdr}-skeleton algorithm and its heuristic modification control the type I error rate more stringently than the PC-skeleton algorithm does, so
their power generally is lower than that of the PC-skeleton algorithm. Figure 4.4 also clearly shows, as Eq. 4.3 implies, that it is the type I error rate, rather than the FDR, that the PC-skeleton algorithm controls under 5%.

Figure 4.5 shows the average computational time spent during each run of the PC$_{fdr}$-skeleton algorithm and its heuristic modification on the statistical tests of (conditional) independence at step 8 and the FDR control at step 12. The computational time was estimated on the platform of an Intel Xeon 1.86GHz CPU and 4G RAM, and with the code implemented in Matlab R14. Tables 4.3 and 4.4 show the average ratios of the computational time spent on the FDR control to that spent on the statistical tests. The average ratios are not more than 2.57\% for all the 48 Bayesian networks. The relatively small standard deviations, as shown in brackets in the tables, indicate that these estimated ratios are trustful. Because the PC$_{fdr}$-skeleton algorithm and its heuristic modification employ the same search strategy as the PC-skeleton algorithm does, this result evidences that the extra computation cost to achieve the control over the FDR is trivial in comparison with the computation already spent by the PC-skeleton algorithm on statistical tests.
4.3. Empirical Evaluation

Figure 4.2: The FDR (with 95% confidence intervals) of the PC-skeleton algorithm, the PC\textit{fdr}\textsuperscript{*}-skeleton algorithm and the PC\textit{fdr}-skeleton algorithm on the DAGs in Figure 4.1 as the dependence parameter $\delta$ shown on the x-axes increases from 0.5 to 1.0.
4.3. Empirical Evaluation

Figure 4.3: The power (with 95% confidence intervals) of the PC-skeleton algorithm, the PC\textsubscript{fdr}\textsuperscript{*}-skeleton algorithm and the PC\textsubscript{fdr}-skeleton algorithm on the DAGs in Figure 4.1 as the dependence parameter $\delta$ shown on the x-axes increases from 0.5 to 1.0.
Figure 4.4: The type I error rates (with 95% confidence intervals) of the PC-skeleton algorithm, the PC\_fdr-skeleton algorithm and the PC\_fdr*-skeleton algorithm on the DAGs in Figure 4.1 as the dependence parameter $\delta$ shown on the x-axes increases from 0.5 to 1.0.
4.3. Empirical Evaluation

Figure 4.5: The average computational time (in seconds, with 95% confidence intervals) spent on the FDR control and statistical tests during each run of the \( \text{PC}_{\text{fdr}} \)-skeleton algorithm and its heuristic modification.
4.3. Empirical Evaluation

<table>
<thead>
<tr>
<th></th>
<th>(N=15)</th>
<th>(N=20)</th>
<th>(N=25)</th>
<th>(N=30)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(D=2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\delta)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>1.11e-03 (3.64e-04)</td>
<td>7.19e-04 (2.32e-04)</td>
<td>5.44e-04 (1.79e-04)</td>
<td>4.81e-04 (1.37e-04)</td>
</tr>
<tr>
<td>0.6</td>
<td>1.48e-03 (2.03e-04)</td>
<td>1.24e-03 (2.15e-04)</td>
<td>1.21e-03 (3.32e-04)</td>
<td>1.09e-03 (1.44e-04)</td>
</tr>
<tr>
<td>0.7</td>
<td>1.58e-03 (1.68e-04)</td>
<td>1.61e-03 (2.01e-04)</td>
<td>1.59e-03 (3.13e-04)</td>
<td>1.64e-03 (1.09e-04)</td>
</tr>
<tr>
<td>0.8</td>
<td>1.63e-03 (1.64e-04)</td>
<td>1.81e-03 (1.61e-04)</td>
<td>1.93e-03 (1.20e-04)</td>
<td>1.89e-03 (1.04e-04)</td>
</tr>
<tr>
<td>0.9</td>
<td>1.59e-03 (1.50e-04)</td>
<td>1.83e-03 (1.50e-04)</td>
<td>2.06e-03 (1.19e-04)</td>
<td>1.95e-03 (9.63e-05)</td>
</tr>
<tr>
<td>1.0</td>
<td>1.64e-03 (1.51e-04)</td>
<td>1.88e-03 (1.59e-04)</td>
<td>2.15e-03 (1.12e-04)</td>
<td>2.01e-03 (9.01e-05)</td>
</tr>
<tr>
<td>(D=3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\delta)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>1.69e-03 (3.70e-04)</td>
<td>1.50e-03 (2.55e-04)</td>
<td>9.80e-04 (1.90e-04)</td>
<td>9.10e-04 (1.52e-04)</td>
</tr>
<tr>
<td>0.6</td>
<td>2.06e-03 (2.82e-04)</td>
<td>2.22e-03 (1.45e-04)</td>
<td>1.93e-03 (1.71e-04)</td>
<td>1.82e-03 (1.47e-04)</td>
</tr>
<tr>
<td>0.7</td>
<td>2.11e-03 (1.84e-04)</td>
<td>2.36e-03 (1.24e-04)</td>
<td>2.31e-03 (1.19e-04)</td>
<td>2.29e-03 (1.12e-04)</td>
</tr>
<tr>
<td>0.8</td>
<td>2.02e-03 (1.68e-04)</td>
<td>2.35e-03 (1.20e-04)</td>
<td>2.45e-03 (1.28e-04)</td>
<td>2.20e-03 (1.15e-04)</td>
</tr>
<tr>
<td>0.9</td>
<td>2.04e-03 (1.50e-04)</td>
<td>2.34e-03 (9.05e-05)</td>
<td>2.53e-03 (1.15e-04)</td>
<td>2.03e-03 (1.23e-04)</td>
</tr>
<tr>
<td>1.0</td>
<td>1.99e-03 (1.41e-04)</td>
<td>2.28e-03 (9.18e-05)</td>
<td>2.57e-03 (9.66e-05)</td>
<td>1.92e-03 (1.25e-04)</td>
</tr>
</tbody>
</table>

Table 4.3: The average ratios (with their standard deviations in brackets) of the computational time spent on the FDR control to that spent on the statistical tests during each run of the \(\text{PC}_{fdr}\)-skeleton algorithm.

<table>
<thead>
<tr>
<th></th>
<th>(N=15)</th>
<th>(N=20)</th>
<th>(N=25)</th>
<th>(N=30)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(D=2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\delta)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>8.88e-04 (2.57e-04)</td>
<td>5.93e-04 (2.25e-04)</td>
<td>3.94e-04 (1.60e-04)</td>
<td>3.24e-04 (1.15e-04)</td>
</tr>
<tr>
<td>0.6</td>
<td>1.12e-03 (2.99e-04)</td>
<td>8.82e-04 (3.19e-04)</td>
<td>7.08e-04 (2.58e-04)</td>
<td>5.86e-04 (1.81e-04)</td>
</tr>
<tr>
<td>0.7</td>
<td>1.17e-03 (2.76e-04)</td>
<td>1.04e-03 (2.92e-04)</td>
<td>9.02e-04 (1.51e-04)</td>
<td>7.45e-04 (1.37e-04)</td>
</tr>
<tr>
<td>0.8</td>
<td>1.16e-03 (2.66e-04)</td>
<td>1.04e-03 (1.83e-04)</td>
<td>1.03e-03 (1.38e-04)</td>
<td>8.23e-04 (1.24e-04)</td>
</tr>
<tr>
<td>0.9</td>
<td>1.22e-03 (2.73e-04)</td>
<td>1.08e-03 (1.91e-04)</td>
<td>1.06e-03 (8.76e-05)</td>
<td>8.37e-04 (1.35e-04)</td>
</tr>
<tr>
<td>1.0</td>
<td>1.21e-03 (2.68e-04)</td>
<td>1.11e-03 (2.09e-04)</td>
<td>1.12e-03 (1.04e-04)</td>
<td>8.31e-04 (1.05e-04)</td>
</tr>
<tr>
<td>(D=3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\delta)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>1.33e-03 (3.42e-04)</td>
<td>1.01e-03 (1.86e-04)</td>
<td>6.74e-04 (1.54e-04)</td>
<td>5.49e-04 (1.08e-04)</td>
</tr>
<tr>
<td>0.6</td>
<td>1.44e-03 (3.46e-04)</td>
<td>1.19e-03 (1.63e-04)</td>
<td>1.08e-03 (2.08e-04)</td>
<td>7.97e-04 (8.83e-05)</td>
</tr>
<tr>
<td>0.7</td>
<td>1.43e-03 (2.41e-04)</td>
<td>1.20e-03 (1.10e-04)</td>
<td>1.09e-03 (1.51e-04)</td>
<td>7.19e-04 (7.80e-05)</td>
</tr>
<tr>
<td>0.8</td>
<td>1.36e-03 (1.38e-04)</td>
<td>1.17e-03 (9.36e-05)</td>
<td>1.10e-03 (1.20e-04)</td>
<td>6.48e-04 (9.32e-05)</td>
</tr>
<tr>
<td>0.9</td>
<td>1.35e-03 (1.34e-04)</td>
<td>1.24e-03 (1.03e-04)</td>
<td>1.10e-03 (8.31e-05)</td>
<td>6.19e-04 (6.57e-05)</td>
</tr>
<tr>
<td>1.0</td>
<td>1.39e-03 (1.72e-04)</td>
<td>1.29e-03 (1.01e-04)</td>
<td>1.14e-03 (9.77e-05)</td>
<td>5.98e-04 (4.86e-05)</td>
</tr>
</tbody>
</table>

Table 4.4: The average ratios (with their standard deviations in brackets) of the computational time spent on the FDR control to that spent on the statistical tests during each run of the \(\text{PC}_{fdr*}\)-skeleton algorithm.

4.3.2 Applications to Real fMRI Data

We applied the \(\text{PC}_{fdr}\)-skeleton and the PC-skeleton algorithms to real-world research tasks, studying the connectivity network between brain regions using functional magnetic resonance imaging (fMRI). The purpose of the applications is to check whether the two algorithms correctly curb the error rates in real world applications. The purpose of
the applications is not, and also should not be, to answer the question “which algorithm, the PC_fdr-skeleton or the PC-skeleton, is superior?”, for the following reasons. Basically, the two algorithms control different error rates between which there is not a superior relationship (see Appendix 4.7). Secondly, the error rate of interest for a specific application is selected largely not by mathematical superiority, but by researchers’ interest and the scenario of research (see Appendix 4.7). Thirdly, the simulation study has clearly revealed the properties of and the differences (not superiority) between the two algorithms. Lastly, the approximating graphical models behind the real fMRI data are unknown, so the comparison on the real fMRI data is rough, rather than rigorous.

The two algorithms were applied to two real fMRI data sets, one including 11 discrete variables and 1300 observations, and the other including 25 continuous variables and 1098 observations. The first data set, denoted by “the bulb-squeezing data set”, was collected from 10 healthy subjects each of whom was asked to squeeze a rubber bulb with their left hand at three different speeds or at a constant force, as cued by visual instruction. The data involve eleven variables: the speed of squeezing and the activities of the ten brain regions listed in Table 4.5. The speed of squeezing is coded as a discrete variable with four possible values: the high speed, the medium speed, the low speed, and the constant force. The activities of the brain regions are coded as discrete variables with three possible values: high activation, medium activation and low activation. The data of each subject include 130 time points. The data of the ten subjects are pooled together, so in total there are 1300 time points. For details of the data set, please refer to Li et al. [19].

The second data set, denoted by “the sentence-picture data set”, was collected from a single subject performing a cognitive task. In each trial of the task, the subject was shown in sequence an affirmative sentence and a simple picture, and then answered whether the sentence correctly described the picture. In half of the trials, the picture was presented first, followed by the sentence. In the remaining trials, the sentence was presented first,
followed by the picture. The data involve the activities of 25 brain regions, as listed in Table 4.6, encoded as continuous variables, at 1098 time points. For details of the data set, please refer to Keller et al. [16] and Mitchell et al. [23].

The PC\textsubscript{fdr}-skeleton and the PC-skeleton algorithms were applied to both the bulb-squeezing and the sentence-picture data sets. Both the FDR level $q$ of the PC\textsubscript{fdr}-skeleton algorithm and the type-I-error-rate level $\alpha$ of the PC-skeleton algorithm were set at 5%. For the bulb-squeezing data set, all of whose variables are discrete, conditional independence was tested with Pearson’s Chi-square test; for the sentence-picture data set, all of whose variables are continuous, conditional independence was tested with the t-test for partial correlation coefficients [10].

The networks learned from the bulb-squeezing data set and the networks learned from the sentence-picture data set are shown in Figures 4.6 and 4.7 respectively. For ease of comparison, the networks learned by the two algorithms are overlaid. Thin solid black edges are those connections detected by both the two algorithms; thin dashed blue edges are those detected only by the PC\textsubscript{fdr}-skeleton algorithm; thick solid red edges are those detected only by the PC-skeleton algorithm. In Figure 4.6, there are 17 thin solid black edges, 0 thin dashed blue edge and 1 thick solid red edge; in Figure 4.7, there are 39 thin solid black edges, 3 thin dashed blue edges and 12 thick solid red edges.

The results intuitively, though not rigorously, support our expectation of the performances of the two algorithms in real world applications. First, since the data sets are relatively large, with the sample sizes more than 1000, it is expected that both algorithms will recover many of the existing connections, and consequently the networks recovered by the two algorithms may share many common connections. This is consistent with the fact that in Figures 4.6 and 4.7 there are many thin solid black edges, i.e. the connections recovered by both algorithms.

Second, since the PC\textsubscript{fdr}-skeleton algorithm is designed to control the FDR while the PC-skeleton algorithm to control the type I error rate, it is expected that the two algo-
4.3. Empirical Evaluation

<table>
<thead>
<tr>
<th>Full Name</th>
<th>Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left/Right anterior cingulate cortex</td>
<td>L_ACC, R_ACC</td>
</tr>
<tr>
<td>Left/Right lateral cerebellar hemispheres</td>
<td>L_CER, R_CER</td>
</tr>
<tr>
<td>Left/Right primary motor cortex</td>
<td>L_M1, R_M1</td>
</tr>
<tr>
<td>Left/Right pre-frontal cortex</td>
<td>L_PFC, R_PFC</td>
</tr>
<tr>
<td>Left/Right supplementary motor cortex</td>
<td>L_SMA, R_SMA</td>
</tr>
</tbody>
</table>

Table 4.5: Brain regions involved in the bulb-squeezing data set. The prefixes “L” or “R” in the abbreviations stand for “Left” or “Right”, respectively.

Figure 4.6: The networks learned from the bulb-squeezing data set, by the PC_{fdr}-skeleton and the PC-skeleton algorithms. For ease of comparison, the networks learned by the two algorithms are overlaid. Thin solid black edges are those connections detected by both the two algorithms; thick solid red edges are those connections detected only by the PC-skeleton algorithm. For the full names of the brain regions, please refer to Table 4.5.
### 4.3. Empirical Evaluation

<table>
<thead>
<tr>
<th>Full Name</th>
<th>Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calcarine fissure</td>
<td>CALC</td>
</tr>
<tr>
<td>Left/Right dorsolateral prefrontal cortex</td>
<td>L_DLPFC, R_DLPFC</td>
</tr>
<tr>
<td>Left/Right frontal eye field</td>
<td>L_FEF, R_FEF</td>
</tr>
<tr>
<td>Left inferior frontal gyrus</td>
<td>L_IFG</td>
</tr>
<tr>
<td>Left/Right inferior parietal lobe</td>
<td>L_IP, R_IP</td>
</tr>
<tr>
<td>Left/Right intraparietal sulcus</td>
<td>L_IPS, R_IPS</td>
</tr>
<tr>
<td>Left/Right inferior temporal lobule</td>
<td>L_IT, R_IT</td>
</tr>
<tr>
<td>Left/Right opercularis</td>
<td>L_OPER, R_OPER</td>
</tr>
<tr>
<td>Left/Right posterior precentral sulcus</td>
<td>L_PPREC, R_PPREC</td>
</tr>
<tr>
<td>Left/Right supramarginal gyrus</td>
<td>L_SGA, R_SGA</td>
</tr>
<tr>
<td>Supplementary motor cortex</td>
<td>SMA</td>
</tr>
<tr>
<td>Left/Right superior parietal lobule</td>
<td>L_SPL, R_SPL</td>
</tr>
<tr>
<td>Left/Right temporal lobe</td>
<td>L_T, R_T</td>
</tr>
<tr>
<td>Left/Right triangularis</td>
<td>L_TRIA, R_TRIA</td>
</tr>
</tbody>
</table>

Table 4.6: Brain regions involved in the sentence-picture data set. The prefixes “L” or “R” in the abbreviations stand for “Left” or “Right”, respectively.

![Network Diagram](image)

Figure 4.7: The networks learned from the sentence-picture data set, by the PC_{fdr}-skeleton and the PC-skeleton algorithms. For ease of comparison, the networks learned by the two algorithms are overlaid. Thin solid black edges are those connections detected by both the two algorithms; thin dashed blue edges are those connections detected only by the PC_{fdr}-skeleton algorithm; thick solid red edges are those connections detected only by the PC-skeleton algorithm. For the full names of the brain regions, please refer to Table 4.6.
4.3. Empirical Evaluation

<table>
<thead>
<tr>
<th>Bulb-Squeezing</th>
<th>Assumed Truth</th>
<th>Realized Detection</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Exist</td>
<td>Non-Exist</td>
</tr>
<tr>
<td>PC\textsubscript{fdr}</td>
<td>17</td>
<td>$\frac{11\times(11-1)}{2} - 17 = 38$</td>
</tr>
<tr>
<td>PC</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Correct</th>
<th>False</th>
<th>FDR</th>
<th>Type I Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC\textsubscript{fdr}</td>
<td>17</td>
<td>0</td>
<td>0.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>PC</td>
<td>17</td>
<td>1</td>
<td>5.56%</td>
<td>2.63%</td>
</tr>
</tbody>
</table>

Sentence-Picture

<table>
<thead>
<tr>
<th></th>
<th>Assumed Truth</th>
<th>Realized Detection</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Exist</td>
<td>Non-Exist</td>
</tr>
<tr>
<td>PC\textsubscript{fdr}</td>
<td>39</td>
<td>$\frac{25\times(25-1)}{2} - 39 = 261$</td>
</tr>
<tr>
<td>PC</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Correct</th>
<th>False</th>
<th>FDR</th>
<th>Type I Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC\textsubscript{fdr}</td>
<td>39</td>
<td>3</td>
<td>7.14%</td>
<td>1.14%</td>
</tr>
<tr>
<td>PC</td>
<td>39</td>
<td>12</td>
<td>23.5%</td>
<td>4.60%</td>
</tr>
</tbody>
</table>

Table 4.7: The realized error rates of the PC\textsubscript{fdr}-skeleton and the PC algorithms on the bulb-squeezing and sentence-picture data sets, under the TI assumption that all and only those connections detected by both of the two algorithms truly exist.

The algorithms will control the corresponding error rate under or around the pre-defined level, which is 5% in this study. To verify whether the error rates were controlled as expected, we need to know which connections really exist and which do not. Unfortunately, this is very difficult for real data sets, because unlike the simulated data, the true models behind the real data are unknown, and in the literature, researchers usually tend to report evidence supporting the existence of connections rather than supporting the non-existence. However, since the sample sizes of the two data sets are relatively large, more than 1000, we can speculate that both of the two algorithms have recovered most of the existing connections. Extrapolating this speculation a bit, we intuitively assume that those connections detected by both of the two algorithms truly exist while all the others do not. In other words, we assume that all and only the thin black edges in the figures truly exist.

We refer to this assumption as the “True Intersection” (TI) assumption. The statistics about Figures 4.6 and 4.7 under the TI assumption, are listed in Table 4.7. The realized FDR of the PC\textsubscript{fdr}-skeleton algorithm on the bulb-squeezing and sentence-picture data sets are 0.00% and 7.14%, respectively; the realized type I error rate of the PC-skeleton algorithm on the bulb-squeezing and sentence-picture data sets are 2.63% and 4.60%, respectively. Considering that the realized error rate, as a statistic extracted from just
4.4 Conclusions and Discussions

a trial, may slightly deviate from its expected value, these results, derived under the TI assumption, support that the two algorithms controlled the corresponding error rate under the pre-defined level 5%.

Third, according to Eq. (4.2), the sparser and the larger the true network is, the higher the FDR of the PC-skeleton algorithm will be. For the bulb-squeezing data set, there are 11 vertices, and under the TI assumption, 17 existing connections and 38 non-existing connections. In this case, the realized FDR of the PC-skeleton algorithm is only 5.56% (Table 4.7). For the sentence-picture data set, there are 25 vertices, and under the TI assumption, 39 existing connections and 261 non-existing connections. In this case, the realized FDR of the PC-skeleton algorithm rises to 23.5% (Table 4.7). This notable increase of the realized FDR is consistent with the prediction based on Eq. (4.2).

It should be noted that the preceding arguments are rough rather than rigorous, since they are based on the TI assumption rather than the true models behind the data. However, because the true models behind the real data are unknown, the TI assumption is a practical and intuitive approach to assess the performance of the two algorithms in the two real world applications.

4.4 Conclusions and Discussions

We have proposed a modification of the PC algorithm, the PC_{fdr}-skeleton algorithm, to curb the false discovery rate (FDR) of the skeleton of the learned Bayesian networks. The FDR-control procedure embedded into the PC algorithm collectively considers the hypothesis tests related to the existence of multiple edges, correcting the effect of multiple hypothesis testing. Under mild assumptions, it is proved that the PC_{fdr}-skeleton algorithm can control the FDR under a user-specified level $q$ ($q > 0$) at the limit of large sample sizes (see Theorem 2). In the cases of moderate sample size (about several hundred), empirical experiments have shown that the method is still able to control the FDR.
under the user-specified level. The PC\textsubscript{fdr}-skeleton algorithm, a heuristic modification of the proposed method, has shown better performance in the simulation study, steadily controlling the FDR closely around the user-specified level and gaining more detection power, although its asymptotic performance has not been theoretically proved. Both the PC\textsubscript{fdr}-skeleton algorithm and its heuristic modification can asymptotically recover all the edges of the true DAG (see Theorem 1). The idea of controlling the FDR can be extended to other constraint-based methods, such as the inductive causation (IC) algorithm [see 26, pages 49–51] and the fast-causal-inference (FCI) algorithm [see 32, pages 142–146].

The simulation study has also shown that the extra computation spent on achieving the FDR control is almost negligible when compared with that already spent by the PC algorithm on statistical tests of conditional independence. The computational complexity of the new algorithm is closely comparable with that of the PC algorithm.

As a modification based on the PC algorithm, the proposed method is modular, consisting of the PC search strategy, statistical tests of conditional independence and an FDR-control procedure. Different statistical tests and FDR-control procedures can be “plugged in”, depending on the data type and the statistical model. Thus, the method is applicable to any models for which statistical tests of conditional independence are available, such as discrete models and Gaussian models.

It should be noted that the PC\textsubscript{fdr}-skeleton algorithm is not proposed to replace the PC-skeleton algorithm. Instead, it provides an approach to controlling the FDR, a certain error-rate criterion for testing the existence of multiple edges. When multiple edges are involved in structure learning, there are different applicable error-rate criteria, such as those listed in Table 4.2. The selection of these criteria depends on researchers’ interest and the scenarios of studies, which is beyond the scope of this paper. When the FDR is applied, the PC\textsubscript{fdr}-skeleton algorithm is preferable; when the type I error rate is applied, the PC-skeleton algorithm is preferable. The technical difference between the two algorithms is that the PC\textsubscript{fdr}-skeleton algorithm adaptively adjusts the type I error
rate according to the sparseness of the network to achieve the FDR control, while the PC-skeleton algorithm fixes the type I error rate.

Currently the FDR control is applied only to the skeleton of the graph, but not to the directions of the edges yet. The final output of the PC algorithm is a partially directed acyclic graph that uniquely represents an equivalence class of DAGs, so a possible improvement for the PC$_{fdr}$-skeleton algorithm is to extend the FDR control to the directions of the recovered edges. Because both type I and type II errors may lead to wrong directions in the later steps of the PC algorithm, minimizing direction errors may lead to a related, yet different, error-control task.

The asymptotic performance of the PC$_{fdr}$-skeleton algorithm has only been proved under the assumption that the number of vertices is fixed. Its behavior when both the number of vertices and the sample size approach infinity has not been studied yet. Kalisch and Bühlmann [15] proved that for Gaussian Bayesian networks, the PC algorithm consistently recovers the equivalence class of the underlying sparse DAG, as the sample size $m$ approaches infinity, even if the number of vertices $N$ grows as quickly as $O(m^{\lambda})$ for any $0 < \lambda < \infty$. Their idea is to adaptively decrease the type I error rate $\alpha$ of the PC-skeleton algorithm as both the number of vertices and the sample size increase. It is desirable to study whether similar behavior can be achieved with the PC$_{fdr}$-skeleton algorithm if the FDR level $q$ is adjusted appropriately as the sample size increases.

It is also worth developing FDR control for structure-learning methods not based on hypothesis tests, such as the popular and fast “least absolute shrinkage and selection operator” (lasso) [36]. For Gaussian graphical models, the lasso is a robust and efficient method for recovering sparse networks. It uses the $L_1$-norm to curb the complexity of linear regression models when selecting predictors. Subject to an upper bound on the $L_1$-norm of the regression coefficients, it minimizes the residual squared errors. Graphical lasso [22] considers learning connections as selecting predictors locally for each node variable. In comparison with other methods, it is extremely fast, able to solve a problem with
4.5. Proof of Theorems

1000 nodes in about a minute [11]. Under and almost only under the “irrepresentable condition” [15, 39], graphical lasso is able to consistently recover the true network asymptotically, even when the number of nodes increases exponentially as the sample size does. A method for controlling the type I error rate with lasso has been presented in [22], but the problem of FDR control has not been addressed, still being an interesting problem for future research.

A Matlab® package of the PC_fdr-skeleton algorithm and its heuristic modification is downloadable at www.junningli.org/software.

Acknowledgment

The authors thank Dr. Martin J. McKeown for sharing the functional magnetic resonance imaging (fMRI) data [19] and helpful discussions.

4.5 Proof of Theorems

To assist the reading, we list below notations frequently used in the proof:

$G_{\text{true}}$: the skeleton of the true underlying Bayesian network.

$\mathcal{A}_{a \sim b}$: the event that edge $a \sim b$ is in the graph recovered by the PC_fdr-skeleton algorithm.

$\mathcal{A}_{E_{\text{true}}}$: the joint event that all the edges in $G_{\text{true}}$, the skeleton of the true DAG, are recovered by the PC_fdr-skeleton algorithm.

$E_{\text{true}}$: the set of the undirected edges that are not in $G_{\text{true}}$. 

$p_{a \sim b}$: the value of $p_{a \sim b}^{\text{max}}$ when the PC_fdr-skeleton algorithm stops.

$C_{a \sim b}^a$: a certain vertex set that d-separates $a$ and $b$ in $G_{\text{true}}$ and that is also a subset of either $\text{adj}(a, G_{\text{true}}) \backslash \{b\}$ or $\text{adj}(b, G_{\text{true}}) \backslash \{a\}$, according to Proposition [11] $C_{a \sim b}^a$ is defined only for vertex pairs that are not adjacent in the true DAG $G_{\text{true}}$. 
$p_{a\sim_b}^*: \text{the } p\text{-value of testing } X_a \perp X_b | X_{C_{a\sim_b}}$. The conditional-independence relationship may not be really tested during the process of the PC_{fdr}-skeleton algorithm, but $p_{a\sim_b}^*$ can still denote the value as if the conditional-independence relationship was tested.

$H^*$: the value in Eq. (4.1) that is either $H$ or $H(1 + 1/2, \ldots, +1/H)$, depending on the assumption of the dependency of the $p$-values.

**Lemma 1** If as $m$ approaches infinity, the probabilities of $K$ events $\mathcal{A}_i(m), \cdots, \mathcal{A}_K(m)$ approach 1 at speed

$$P(\mathcal{A}_i(m)) = 1 - o(\beta(m))$$

where $\lim_{m \to \infty} \beta(m) = 0$ and $K$ is a finite integer, then the probability of the joint of all these events approaches 1 at speed

$$P\left( \bigcap_{i=1}^{K} \mathcal{A}_i(m) \right) \geq 1 - Ko(\beta(m))$$

as $m$ approaches infinity.

**Proof**

\[
\begin{align*}
\therefore \bigcap_{i=1}^{K} \mathcal{A}_i(m) &= \bigcup_{i=1}^{K} \bar{\mathcal{A}}_i(m). \\
\therefore P\left( \bigcap_{i=1}^{K} \mathcal{A}_i(m) \right) &= 1 - P\left( \bigcup_{i=1}^{K} \bar{\mathcal{A}}_i(m) \right) \geq 1 - \sum_{i=1}^{K} P(\bar{\mathcal{A}}_i(m)) \\
&= 1 - \sum_{i=1}^{K} [1 - P(\mathcal{A}_i(m))] = 1 - \sum_{i=1}^{K} o(\beta(m)) = 1 - Ko(\beta(m)).
\end{align*}
\]

**Corollary 1** If $\mathcal{A}_i(m), \cdots, \mathcal{A}_K(m)$ are a finite number of events whose probabilities each approach 1 as $m$ approaches infinity:

$$\lim_{m \to \infty} P(\mathcal{A}_i(m)) = 1,$$
then the probability of the joint of all these events approaches 1 as \( m \) approaches infinity:

\[
\lim_{m \to \infty} P \left( \bigcap_{i=1}^{K} \mathcal{A}_i(m) \right) = 1.
\]

**Lemma 2** If there are \( F \) \((F \geq 1)\) false hypotheses among \( H \) tested hypotheses, and the \( p \)-values of all the false hypotheses are smaller than or equal to \( \frac{F}{H^*} q \), where \( H^* \) is either \( H \) or \( H(1+1/2, \ldots, +1/H) \), depending on the assumption of the dependency of the \( p \)-values, then all the \( F \) false hypotheses will be rejected by the FDR procedure, Algorithm \( \text{Algorithm 2} \).

**Proof**

Let \( p_i \) \((i = 1, \ldots, H)\) denote the \( p \)-value of the \( i \)th hypothesis, \( p_f \) denote the maximum of the \( p \)-values of the \( F \) false hypotheses, and \( r_f \) denote the rank of \( p_f \) in the ascending order of \( \{p_i\}_{i=1, \ldots, H} \).

\[
F \leq p_f.
\]

\[
r_f = |\{p_i | p_i \leq p_f\}| \geq F.
\]

\[
\rightarrow \frac{H^*}{r_f} p_f \leq \frac{H^*}{F} \cdot p_f.
\]

\[
\rightarrow \frac{H^*}{r_f} p_f \leq \frac{H^*}{F} \cdot p_f \leq q.
\]

Thus, hypotheses with \( p \)-values not greater than \( p_f \) will be rejected.

\[
\rightarrow \text{The } p \text{-values of the } F \text{ false hypotheses are not greater than } p_f.
\]

\[
\rightarrow \text{All the } F \text{ false hypotheses will be rejected by the FDR procedure, Algorithm 2.}
\]

**Proof of Theorem 1**

If there is not any edge in the true DAG \( G_{\text{true}} \), then the proof is trivially \( E_{\text{true}} \supseteq \emptyset \subseteq E^\sim \).

In the following part of the proof, we assume \( E_{\text{true}} \neq \emptyset \). For the PC_{\text{fdr}}-skeleton algorithm and its heuristic modification, whenever the FDR procedure, Algorithm 2 is invoked, \( p_{a \rightarrow b}^\max \) is always less than \( \max_{C \in V \setminus \{a, b\}} \{p_{a \leftarrow b | C}\} \), and the number of \( p \)-values input to the FDR
4.5. Proof of Theorems

Theorem 1

The algorithm is always not more than $C_N^2$. Thus, according to Lemma 2 if

$$\max_{a \sim \alpha \in E_{\text{true}}} \left\{ \max_{C \in V \setminus \{a, b\}} \{ p_{a \perp \lvert \lvert C} \} \right\} \leq \frac{|E_{\text{true}}|}{C_N^2} \frac{1}{q},$$

(4.5)

then all the true connections will be recovered by the PCfdr-skeleton algorithm and its heuristic modification. Let $\mathcal{A}_{a \perp \lvert \lvert C}$ denote the event

$$p_{a \perp \lvert \lvert C} \leq \frac{|E_{\text{true}}|}{C_N^2} \frac{1}{q},$$

$\mathcal{A}_{E_{\text{true}}}^IR$ denote the event of Eq. (4.5), and $\mathcal{A}_{E_{\text{true}}}^IR$ denote the event that all the true connections are recovered by the PCfdr-skeleton algorithm and its heuristic modification.

$\therefore \mathcal{A}_{E_{\text{true}}}^IR$ is a sufficient condition for $\mathcal{A}_{E_{\text{true}}}^IR$, according to Lemma 2.

$\therefore \mathcal{A}_{E_{\text{true}}} \subseteq \mathcal{A}_{E_{\text{true}}}^IR.$

$\therefore P(\mathcal{A}_{E_{\text{true}}}^IR) \geq P(\mathcal{A}_{E_{\text{true}}}^IR).$

$\therefore \mathcal{A}_{E_{\text{true}}}^IR$ is the joint of a limited number of events as

$$\mathcal{A}_{E_{\text{true}}}^IR = \bigcap_{a \sim \alpha \in E_{\text{true}}} \bigcap_{C \in V \setminus \{a, b\}} \mathcal{A}_{a \perp \lvert \lvert C},$$

and $\lim_{m \to \infty} P(\mathcal{A}_{a \perp \lvert \lvert C}) = 1$ according to Assumption (A3).

$\therefore$ According to Corollary 1 $\lim_{m \to \infty} P(\mathcal{A}_{E_{\text{true}}}^IR) = 1.$

$\therefore 1 \geq \lim_{m \to \infty} P(\mathcal{A}_{E_{\text{true}}}^IR) \geq \lim_{m \to \infty} P(\mathcal{A}_{E_{\text{true}}}^IR) = 1.$

$\therefore \lim_{m \to \infty} P(\mathcal{A}_{E_{\text{true}}}^IR) = 1.$

**Lemma 3** Given any FDR level $q > 0$, if the p-value vector $P = [p_1, \cdots, p_H]$ input to Algorithm 2 is replaced with $P' = [p_1', \cdots, p_H']$, such that (1) for those hypotheses that are rejected when $P$ is the input, $p_i'$ is equal to or less than $p_i$, and (2) for all the other hypotheses, $p_i'$ can be any value between 0 and 1, then the set of rejected hypotheses when...
4.5. Proof of Theorems

$P'$ is the input is a superset of those rejected when $P$ is the input.

Proof

Let $R$ and $R'$ denote the sets of the rejected hypotheses when $P$ and $P'$ are respectively input to the FDR procedure.

If $R = \emptyset$, then the proof is trivially $R' \supseteq \emptyset = R$.

If $R \neq \emptyset$, let us define $\alpha = \max_{i \in R} p_i$ and $\alpha' = \max_{i \in R} p'_i$. Let $r = |R|$ denote the rank of $\alpha$ in the ascending order of $P$ and $r'$ denote the rank of $\alpha'$ in the ascending order of $P'$.

\[
\therefore p'_i \leq p_i \text{ for all } i \in R.
\]

\[
\therefore \alpha' \leq \alpha.
\]

\[
\therefore \alpha' = \max_{i \in R} p'_i.
\]

\[
\therefore r' \geq |R| = r.
\]

\[
\therefore \frac{H^*}{r} \alpha \leq q.
\]

\[
\therefore \frac{H^*}{r} \alpha' \leq \frac{H^*}{r} \alpha \leq q.
\]

\[
\therefore \text{When $P'$ is the input, hypotheses with $p'_i$ smaller than or equal to $\alpha'$ will be rejected.}
\]

\[
\therefore p'_i \leq \alpha', \forall i \in R.
\]

\[
\therefore R \subseteq R', \text{ equivalently } R' \supseteq R.
\]

Corollary 2 Given any FDR level $q > 0$, if the p-value vector $P = [p_1, \cdots, p_H]$ input to Algorithm 2 is replaced with $P' = [p'_1, \cdots, p'_H]$ such that $p'_i \leq p_i$ for all $i = 1, \cdots, H$, then the set of rejected hypotheses when $P'$ is the input is a superset of those rejected when $P$ is the input.

Proof of Theorem 2

Let $E_{\text{stop}}$ and $E_{\text{stop}}^\times$ denote the undirected edges respectively recovered and removed by the PC$_{fdr}$-skeleton algorithm when the algorithm stops. Let sequence $P_{1}^{\text{max}}, \cdots, P_{K}^{\text{max}}$ denote the values of $P^{\text{max}}$ when the FDR procedure is invoked at step 12 as the algorithm progresses, in the order of the update process of $P^{\text{max}}$, and let $E_{k}^{\times}$ denote the set
of removable edges indicated by the FDR procedure, with $P_k^{max}$ as the input. $E_k^r$ may include edges that have already been removed.

**: The PC$_{fdr}$-skeleton algorithm accumulatively removes edges in $E_k^r$.
**: $E_{stop}^r = \bigcup_{k=1}^{K} E_k^r$.
**: $P^{max}$ is updated increasingly at step $10$ of the algorithm.
**: According to Corollary $2$ $E_1^r \subseteq \cdots \subseteq E_K^r$.
**: $E_{stop}^r = \bigcup_{k=1}^{K} E_k^r = E_K^r$.

Let $P = \{p_{a\sim b}\}$ denote the value of $P^{max}$ when the PC$_{fdr}$-skeleton algorithm stops.
**: The FDR procedure is invoked whenever $P^{max}$ is updated.
**: The value of $P^{max}$ does not change after the FDR procedure is invoked for the last time.
**: $P = P_K^{max}$.
**: $E_{\sim stop}^r$ is the same as the edges recovered by directly applying the FDR procedure to $P$.

The theorem is proved through comparing the result of the PC$_{fdr}$-skeleton algorithm with that of applying the FDR procedure to a virtual $p$-value set constructed from $P$.

The virtual $p$-value set $P^*$ is defined as follows.

For a vertex pair $a \sim b$ that is not adjacent in the true DAG $G_{true}$, let $C_{a\sim b}^*$ denote a certain vertex set that d-separates $a$ and $b$ in $G_{true}$ and that is also a subset of either $\text{adj}(a, G_{true}) \setminus \{b\}$ or $\text{adj}(b, G_{true}) \setminus \{a\}$. Let us define $P^* = \{p_{a\sim b}^*\}$ as:

$$p_{a\sim b}^* = \begin{cases} p_{a \perp \!\!\!\perp C_{a\sim b}^*} : a \sim b \in E_{true}^r, \\ p_{a \sim b} : a \sim b \in E_{true}^r. \end{cases}$$

Though $p_{a \perp \!\!\!\perp C_{a\sim b}^*}$ may not be actually calculated during the process of the algorithm, $p_{a \perp \!\!\!\perp C_{a\sim b}^*}$ still can denote the value as if it was calculated. Let us design a virtual algorithm, called $Algorithm^*$, that recovers edges by just applying the FDR procedure to
4.5. Proof of Theorems

Let \( P^* \), and let \( E_{\sim}^{*} \) denote the edges recovered by this virtual algorithm. This algorithm is virtual and impracticable because the calculation of \( P^* \) depends on the unknown \( E_{\text{true}}^{\sim} \), but this algorithm exists because \( E_{\text{true}}^{\sim} \) exists. For any vertex pair \( a \) and \( b \) that is not adjacent in \( G_{\text{true}} \):

\[
\vdash: X_a \text{ and } X_b \text{ are conditional independent given } X_{C_{a-b}}^*.
\]

\[
\vdash: p_{a \perp | C_{a-b}^*} \text{ follows the uniform distribution on } [0, 1].
\]

\[
\vdash: \text{The FDR of } \text{Algorithm}^* \text{ is under } q.
\]

When all the true edges are recovered by the PC\(_{\text{thr}}\)-skeleton algorithm, i.e. \( E_{\text{true}}^{\sim} \subseteq E_{\text{stop}} \), the conditional independence between \( X_a \) and \( X_b \) given \( X_{C_{a-b}}^* \) is tested for all the falsely recovered edges \( a \sim b \in E_{\text{true}}^{\sim} \cap E_{\text{stop}} \), because for these edges, subsets of \( \text{adj}(a, G_{\text{true}}) \setminus \{b\} \) and subsets of \( \text{adj}(a, G_{\text{true}}) \setminus \{b\} \) have been exhaustively searched and \( C_{a-b}^* \) is one of them. Therefore, \( p_{a \sim b} \geq p_{a \sim b}^* \) for all \( a \sim b \in E_{\text{stop}}^{\sim} \) when event \( \mathcal{A}_{E_{\text{true}}^{\sim}} \) happens. Consequently, according to Lemma 3 if event \( \mathcal{A}_{E_{\text{true}}^{\sim}} \) happens, \( E_{\text{stop}}^{\sim} \subseteq E_{\sim}^{*} \).

Let \( q(E_{\sim}) \) denote the realized FDR of reporting \( E_{\sim} \) as the recovered skeleton of the true DAG:

\[
q(E_{\sim}) = \begin{cases} \frac{|E_{\sim} \cap E_{\text{true}}^{\sim}|}{|E_{\sim}|} & : \text{ } E_{\sim} \neq \emptyset, \\ 0 & : \text{ } E_{\sim} = \emptyset. \end{cases}
\]

The FDRs of the PC\(_{\text{thr}}\)-skeleton algorithm and Algorithm\(^*\) are \( E[q(E_{\text{stop}}^{\sim})] \) and \( E[q(E_{\sim}^{*})] \) respectively. Here \( E[x] \) means the expected value of \( x \).

\[
\vdash: E[q(E_{\text{stop}}^{\sim})] = E[q(E_{\text{stop}}^{\sim}) | \mathcal{A}_{E_{\text{true}}^{\sim}}] P(\mathcal{A}_{E_{\text{true}}^{\sim}}) + E[q(E_{\text{stop}}^{\sim}) | \mathcal{A}_{E_{\text{true}}^{\sim}}^c] P(\mathcal{A}_{E_{\text{true}}^{\sim}}^c) \\
\leq Q + P(\mathcal{A}_{E_{\text{true}}^{\sim}}^c), \text{ where } Q = E[q(E_{\text{stop}}^{\sim}) | \mathcal{A}_{E_{\text{true}}^{\sim}}] P(\mathcal{A}_{E_{\text{true}}^{\sim}}).
\]

\[
\vdash: \limsup_{m \to \infty} E[q(E_{\text{stop}}^{\sim})] \leq \limsup_{m \to \infty} Q + \limsup_{m \to \infty} P(\mathcal{A}_{E_{\text{true}}^{\sim}}^c).
\]

\[
\vdash: \lim_{m \to \infty} P(\mathcal{A}_{E_{\text{true}}^{\sim}}) = 1, \text{ according to Theorem 1.}\n\]

\[
\vdash: \limsup_{m \to \infty} P(\mathcal{A}_{E_{\text{true}}^{\sim}}^c) = \lim_{m \to \infty} P(\mathcal{A}_{E_{\text{true}}^{\sim}}) = 0.
\]

\[
\vdash: \limsup_{m \to \infty} E[q(E_{\text{stop}}^{\sim})] \leq \limsup_{m \to \infty} Q.
\]

\[
\vdash: Q \leq E[q(E_{\text{stop}}^{\sim})].
\]

\[
\vdash: \limsup_{m \to \infty} Q \leq \limsup_{m \to \infty} E[q(E_{\text{stop}}^{\sim})].
\]
4.6 Statistical Tests with Asymptotic Power Equal to One

Assumption (A3) on the asymptotic power of detecting conditional dependence appears demanding, but actually the detection power of several standard statistical tests approaches one as the number of identically and independently sampled observations approaches infinity. Listed as follows are two statistical tests satisfying Assumption (A3) for Gaussian models or discrete models.

\[
\limsup_{m \to \infty} E[Q(\tilde{E}_{stop})] = \limsup_{m \to \infty} Q = \limsup_{m \to \infty} E[q(\tilde{E}_{stop}) | A_{E_{true}}] P(A_{E_{true}}).
\]

Similarly, \[\limsup_{m \to \infty} E[q(\tilde{E}^{*})] = \limsup_{m \to \infty} E[q(\tilde{E}^{*}) | A_{E_{true}}] P(A_{E_{true}}).\]

\[
\therefore \text{ Given event } A_{E_{true}}, \tilde{E}_{true} \subseteq E_{stop} \subseteq \tilde{E}^{*}.
\]

\[
\therefore \text{ Given event } A_{E_{true}}, \quad q(\tilde{E}_{stop}) = \frac{|E_{stop}| - |E_{true}|}{|E_{stop}|} = 1 - \frac{|E_{true}|}{|E_{stop}|} \leq 1 - \frac{|E_{true}|}{|\tilde{E}^{*}|} = \frac{|\tilde{E}^{*}| - |E_{true}|}{|\tilde{E}^{*}|} = q(\tilde{E}^{*}).
\]

\[
\therefore \limsup_{m \to \infty} E[q(\tilde{E}_{stop}) | A_{E_{true}}] P(A_{E_{true}}) \leq \limsup_{m \to \infty} E[q(\tilde{E}^{*}) | A_{E_{true}}] P(A_{E_{true}}).
\]

\[
\therefore \limsup_{m \to \infty} E[q(\tilde{E}_{stop})] \leq \limsup_{m \to \infty} E[q(\tilde{E}^{*})].
\]

\[
\therefore \text{ Algorithm}^{*} \text{ controls the FDR under } q.
\]

\[
\therefore E[q(\tilde{E}^{*})] \leq q.
\]

\[
\therefore \limsup_{m \to \infty} E[q(\tilde{E}^{*})] \leq q.
\]

\[
\therefore \limsup_{m \to \infty} E[q(\tilde{E}_{stop})] \leq q.
\]
4.6. Statistical Tests with Asymptotic Power Equal to One

Fisher’s z transformation on sample partial-correlation-coefficients for Gaussian models

In multivariate Gaussian models, $X_a$ and $X_b$ are conditional independent given $X_C$ if and only if the partial-correlation-coefficient of $X_a$ and $X_b$ given $X_C$ is zero [see IS pages 129–130]. The partial-correlation-coefficient $\rho$ is defined as:

$$
\rho = \frac{\text{Cov} [Y_a,Y_b]}{\sqrt{\text{Var}[Y_a]\text{Var}[Y_b]}},
$$

$$
Y_a = X_a - < W_a, X_C >,
$$

$$
Y_b = X_b - < W_b, X_C >,
$$

$$
W_a = \arg \min_w E[(X_a - < w, X_C >)^2],
$$

$$
W_b = \arg \min_w E[(X_b - < w, X_C >)^2].
$$

The sample partial-correlation-coefficient $\hat{\rho}$ can be calculated from $m$ i.i.d. samples $[x_{ai}, x_{bi}, x_{Ci}]$ ($i = 1, \cdots, m$) as:

$$
\hat{\rho} = \frac{\frac{1}{m} \sum_{i=1}^{m} [(\hat{y}_{ai} - \bar{y}_a)(\hat{y}_{bi} - \bar{y}_b)]}{\sqrt{\frac{1}{m} \sum_{i=1}^{m} (\hat{y}_{ai} - \bar{y}_a)^2 \frac{1}{m} \sum_{i=1}^{m} (\hat{y}_{bi} - \bar{y}_b)^2}},
$$

$$
\hat{y}_a = \frac{1}{m} \sum_{i=1}^{m} \hat{y}_{ai},
$$

$$
\hat{y}_b = \frac{1}{m} \sum_{i=1}^{m} \hat{y}_{bi},
$$

$$
\hat{y}_{ai} = x_{ai} - < \hat{W}_a, x_{Ci} >,
$$

$$
\hat{y}_{bi} = x_{bi} - < \hat{W}_b, x_{Ci} >,
$$

$$
\hat{W}_a = \arg \min_w \sum_{i=1}^{m} (x_{ai} - < w, x_{Ci} >)^2,
$$

$$
\hat{W}_b = \arg \min_w \sum_{i=1}^{m} (x_{bi} - < w, x_{Ci} >)^2.
$$

The asymptotic distribution of $z(\hat{\rho})$, where $z(x)$, the Fisher’s z transformation [see 9], is defined as

$$
z(x) = \frac{1}{2} \log \frac{1 + x}{1 - x},
$$
4.6. Statistical Tests with Asymptotic Power Equal to One

is the normal distribution with mean \( z(\rho) \) and variance \( \frac{1}{m - |C| - 3} \) [see [2] pages 120–134]. When the type I error rate is kept lower than \( \alpha \), the power of detecting \( \rho \neq 0 \) with Fisher’s \( z \) transformation is the probability that \( \sqrt{m - |C| - 3} z(\hat{\rho}) \) falls in the range \( (-\infty, \Phi^{-1}(\alpha/2)] \) or \([\Phi^{-1}(1 - \alpha/2), +\infty)\), where \( \Phi \) is the cumulative distribution function of the standard normal distribution and \( \Phi^{-1} \) is its inverse function. Without loss of generality, we assume the true partial-correlation-coefficient \( \rho \) is greater than zero, then the asymptotic power is

\[
\lim_{m \to \infty} \text{Power} = \lim_{m \to \infty} P \left( \sqrt{m - |C| - 3} z(\hat{\rho}) \geq \Phi^{-1}(1 - \alpha/2) \right) = \lim_{m \to \infty} \left( 1 - \Phi[\Phi^{-1}(1 - \alpha/2) - \sqrt{m - |C| - 3} z(\rho)] \right) = (1 - \Phi[-\infty]) = 1.
\]

The likelihood-ratio test generally applicable to nested models

The likelihood ratio is the ratio of the maximum likelihood of a restricted model to that of a saturated model [see [24]]. Let \( f(x, \theta) \) denote the probability density function of a random vector \( x \) parametrized with \( \theta = [\theta^1, \cdots, \theta^k] \). The null hypothesis restricts \( \theta \) to a set \( \Omega \) specified with \( r \) (\( r \leq k \)) constraints

\[
\xi_1(\theta) = \xi_2(\theta) = \cdots = \xi_r(\theta) = 0.
\]

Given i.i.d. observations \( x_1, \cdots, x_m \), let \( L(\theta) \) denote the likelihood function

\[
L(\theta) = \prod_{i=1}^{m} f(x_i, \theta).
\]

The likelihood ratio \( \Lambda \) given the observations is defined as

\[
\Lambda = \frac{\sup_{\theta \in \Omega} L(\theta)}{\sup_{\theta} L(\theta)}.
\]
4.6. Statistical Tests with Asymptotic Power Equal to One

Wald \[38\] has proved that under certain assumptions on \(f(x, \theta)\) and \(\xi_1(\theta), \cdots , \xi_r(\theta)\), the limit distribution of the statistic \(-2\log\Lambda\) is the \(\chi_r^2\) distribution with \(r\) degrees of freedom if the null hypothesis true. If the null hypothesis is not true, the distribution of \(-2\log\Lambda\) approaches the non-central \(\chi_r^2(\lambda)\) distribution with \(r\) degrees of freedom and the non-central parameter

\[
\lambda = mD(\theta) \geq 0,
\]

\[
D(\theta) = \sum_{i=1}^{k} \sum_{j=1}^{k} \frac{\xi_i(\theta)\xi_j(\theta)}{E \left[ -\frac{\partial^2 f(x, \theta)}{\partial \theta_i \partial \theta_j} \right]}.
\]

If \(D(\theta) > 0\), then \(\lim_{m \to \infty} \lambda = \infty\). Let \(t (t < \infty)\) denote the threshold of rejecting the null hypothesis with type I error rate under \(\alpha (\alpha > 0)\). The asymptotic power of detecting a \(\theta\) that is not in \(\Omega\) and whose \(D(\theta)\) is greater than 0 is \(\lim_{\lambda \to \infty} P(\chi_r^2(\lambda) > t)\). The mean and the variance of the \(\chi_r^2(\lambda)\) distribution is \(u = r + \lambda\) and \(\sigma^2 = 2(r + 2\lambda)\), respectively. When \(\lambda\) is large enough,

\[
P(\chi_r^2(\lambda) > t) \geq P(t < \chi_r^2(\lambda) < u + (u - t)) = 1 - P(|\chi_r^2(\lambda) - u| \geq u - t).
\]

According to Chebyshev’s inequality,

\[
P(|\chi_r^2(\lambda) - u| \geq u - t) \leq \frac{\sigma^2}{(u - t)^2} = \frac{2(r + 2\lambda)}{(r + \lambda - t)^2}.
\]

\(\because\) When \(\lambda\) is large enough, \(P(\chi_r^2(\lambda) > t) \geq 1 - \frac{2(r+2\lambda)}{(r+\lambda-t)^2}\).

\(\therefore\) \(\lim_{m \to \infty} \lambda = \infty\) and both \(r\) and \(t\) are fixed.

\(\therefore\) \(\lim_{m \to \infty} \frac{2(r+2\lambda)}{(r+\lambda-t)^2} = 0\).

\(\therefore\) \(\lim_{m \to \infty} P(\chi_r^2(\lambda) > t) = 1\).
4.7 Error Rates of Interest

Statistical decision processes usually involve choices between negative hypotheses and their alternatives, positive hypotheses. In the decision, there are basically two sources of errors: the type I errors, \( i.e. \) falsely rejecting negative hypotheses when they are actually true; and the type II errors, \( i.e. \) falsely accepting negative hypotheses when their alternatives, the positive hypotheses are actually true. In the context of learning graph structures, a negative hypothesis could be that an edge does not exist in the graph, while the positive hypothesis could be that the edge does exist. It is generally impossible to absolutely prevent the two types of errors simultaneously, because observations of a limited sample size may appear to support a positive hypothesis more than a negative hypothesis even when actually the negative hypothesis is true, or vice versa, due to the stochastic nature of random sampling. Moreover, the two types of errors generally contradict each other. Given a fixed sample size and a certain statistic extracted from the data, decreasing the type I errors will increase the type II errors, and vice versa. To guarantee the absolute prevention of the type I errors in any situations, one must accept all negative hypotheses, which will generally lead the type II error rate to be one, and vice versa. The contradiction between the two types of errors is clearly revealed by the monotone increase of receiver operating characteristic (ROC) curves. Thus the errors must be controlled by setting a threshold on a certain type of errors, or trading off between them, for instance, by minimizing a certain lost function associated with the errors according to the Bayesian decision theory.

Rooted in the two types of errors, there are several different error-rate criteria (as listed in Table 4.2) for problems involving simultaneously testing multiple hypotheses, such as verifying the existence of edges in a graph. The type I error rate is the expected ratio of the type I errors to all the negative hypotheses that are actually true; the type II error rate is the expected ratio of the type II errors to all the positive hypotheses that are actually true; the false discovery rate (FDR) \( \text{[see 43-44]} \), is the expected ratio of
falsely accepted positive hypotheses to all those accepted positive hypotheses; the family-wise error rate is the probability that at least one of the accepted positive hypotheses is actually wrong.

Generally, there are no mathematically or technically superior relationships among these error-rate criteria. Each of these error rates may be favoured in certain research scenarios. For example:

- We are diagnosing a dangerous disease whose treatment is so risky that may cause the loss of eyesight. Due to the great risk of the treatment, we hope that less than 0.1% of healthy people will be falsely diagnosed as patients of the disease. In this case, the type I error rate should be controlled under 0.1%.

- We are diagnosing cancer patients. Because failure in detecting the disease will miss the potential chance to save the patient’s life, we hope that 95% of the cancer patients will be correctly detected. In this case, the type II error rate should be controlled under 5%.

- In a pilot study, we are selecting candidate genes for a genetic research on Parkinson’s disease. Because of the limited funding, we can only study a limited number of genes in the afterward genetic research, so when selecting candidate genes in the pilot study, we hope that 95% of the selected candidate genes are truly associated with the disease. In this case, the FDR will be chosen as the error rate of interest and should be controlled under 5%.

- We are selecting electronic components to make a device. Any error in any component will cause the device to run out of order. To guarantee the device functions well with a probability higher than 99%, the family-wise error rate should be controlled under 1%.

In these examples, the particular error-rate criteria are selected by reasons beyond mathematical or technical superiority, but by the researchers’ interest, to minimize a certain
lost function associated with the errors according to the Bayesian decision theory. Learning network structures in real world applications may face scenarios similar to the above examples.

The excellent discrimination between negative hypotheses and positive hypotheses cannot be achieved by “smartly” setting a threshold on a “superior” error-rate criterion. Setting a threshold on a certain type of error rate is just choosing a cut-point on the ROC curve. If the ROC curve is not sharp enough, any cut-point on the curve away from the ends (0,0) and (1,1) still leads to considerable errors. To discriminate more accurately between a negative hypothesis and a positive hypothesis, one must design a better statistic or increase the sample size to achieve a sharper ROC curve.
Bibliography


Chapter 4. Bibliography


Chapter 5

Extending Error-Rate Control to Dynamic Bayesian Networks

The PC$_{fdr}$ algorithms presented in Chapter 4 have a significant contribution in the field of network learning, in that they introduce an adequate approach of handling the false discovery rate with theoretical justification. The algorithms in Chapter 4 are only basic implementations of the main idea. To extend the control over the false discovery rate to dynamic Bayesian networks, the PC$_{fdr}$ algorithms must be adapted and further developed. In this chapter, we present two extensions of the PC$_{fdr}$ algorithms designed for dynamic Bayesian networks. One is an adaptation to prior knowledge, allowing users to specify which edges must appear in the network, which cannot, and which are to be learned from data. This extension is naturally applicable to dynamic Bayesian networks, by simply regarding them as Bayesian networks that cannot have edges from time $t+1$ to time $t$. The other extension is using the PC$_{fdr}$ algorithms to improve Bayesian inference of dynamic Bayesian networks. The idea is to first learn a network with the PC$_{fdr}$ algorithms, and then make Bayesian inference based on a prior distribution derived from the learned network. It accelerates Bayesian inference and is relatively robust to perturbing noise.

---

12 A version of Section 5.1 of this chapter has been published. Junning Li, Z. Jane Wang and Martin J. McKeown (2008) Learning Brain Connectivity with the False-Discovery-Rate-Controlled PC-Algorithm. Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) 4617–4620. A version of Section 5.2 of this chapter has been accepted for publication. Junning Li, Z. Jane Wang and Martin J. McKeown (2009) Incorporating Error-Rate-Controlled Prior in Modelling Brain Functional Connectivity. The 3rd International Conference on Bioinformatics and Biomedical Engineering.
5.1 Integration of Prior Knowledge

The PC$_{fdr}$ algorithms in Chapter 4 do not consider prior knowledge and cannot be applied to models such as dynamic Bayesian networks. In this section, we present an extension of the PC$_{fdr}$ algorithms that is able to incorporate prior knowledge. We also demonstrate how to apply it to learning Brain connectivity from continuous fMRI data, with an application to a study on Parkinson’s disease.

5.1.1 False Discovery Rate

When the existence of many connections is investigated simultaneously, the error rate should be controlled appropriately. The false discovery rate (FDR) is an error-rate criterion of multiple testing, defined as the expected ratio of falsely rejected negative hypotheses to all those rejected. Referring to Table 5.1, the FDR is formally defined as

\[
FDR = E\left(\frac{FP}{R_2}\right),
\]

where $FP/R_2$ is defined to be zero when $R_2$ is zero. A variation of the FDR, the positive false discovery rate (pFDR), defined as

\[
pFDR = E\left(\frac{FP}{R_2} | R_2 > 0\right)
\]

was proposed in [11]. In the context of learning brain connectivity, a negative hypothesis is that a connection between two brain regions does not exist, and a positive hypothesis is that the connection exists. The FDR is the expected proportion of the falsely discovered connections to all those discovered.

Among many error rate criteria, such as the type I error rate, the type II error rate...
5.1. Integration of Prior Knowledge

Table 5.1: Results of multiple hypothesis testing

<table>
<thead>
<tr>
<th>Test Results</th>
<th>Truth</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Negative</td>
<td>TN (true negative)</td>
<td>FN (false negative)</td>
<td>( R_1 )</td>
</tr>
<tr>
<td>Positive</td>
<td>FP (false positive)</td>
<td>TP (true positive)</td>
<td>( R_2 )</td>
</tr>
<tr>
<td>Total</td>
<td>( T_1 )</td>
<td>( T_2 )</td>
<td>( R )</td>
</tr>
</tbody>
</table>

and etc., the FDR is a reasonable criterion for biomedical applications, because it directly reflects the uncertainty of reported positive results.

Yoav Benjamini and Daniel Yekutieli have proved that, when the test statistics have positive regression dependency on each of the test statistics corresponding to the true negative hypotheses, the FDR can be controlled under the user-specified level \( q \) by the following procedure.

1. Sort the \( p \)-values of \( H \) hypothesis tests in ascendant order as \( p_1 \leq \ldots \leq p_H \).

2. Find the largest \( k \) such that \( p_k \leq \frac{k}{H} q \).

3. Reject hypotheses \( 1, \ldots, k \).

In other cases of dependency, the FDR can be controlled with a simple conservative modification of the procedure by replacing \( H \) in step 2 with \( 1 + 1/2 + \cdots + 1/H \).

5.1.2 PC\(_{fdr^*}\) Algorithm with Prior Knowledge

The extended PC\(_{fdr^*}\) algorithm is described in Algorithm 4. It incorporates prior knowledge with two inputs: \( E_{\text{must}} \) and \( E_{\text{test}} \). \( E_{\text{must}} \) is the set of edges that must appear in the true graph according to the prior knowledge and \( E_{\text{test}} \) is the set of edges that are to be tested from the observed data \( X \). Let \( E_{\text{full}} \) denote the edges of the fully connected graph. Because the union of \( E_{\text{must}} \) and \( E_{\text{test}} \) does not necessarily cover all the edges of the fully connected graph, the edge set \( E \) initialized in step 2 may exclude some edges in \( E_{\text{full}} \). In this way, the prior knowledge on certain impossible edges is incorporated.
5.1. Integration of Prior Knowledge

too. The original PC_{fdr*} algorithm can be regarded as a special case of the extended algorithm, by setting $E_{must} = \emptyset$ and $E_{test} = E_{full}$.

**Algorithm 4** The extended PC_{fdr*} algorithm

Notations: $a$, $b$, $\cdots$ and $A$, $B$, $\cdots$ denote vertices and vertex sets respectively; $X_a$, $X_b$, $\cdots$ and $X_A$, $X_B$, $\cdots$ denote variables and variable sets represented by their subscripts respectively. $a \sim b$ denotes an undirected edge. $G$ denotes an undirected graph and $E$ denotes the undirected-edge set of $G$. $\text{adj}(a, G)$ denotes vertices adjacent to $a$ in graph $G$. $X_a \perp X_b | X_C$ denotes the conditional independence between $X_a$ and $X_b$ given $X_C$.

**Input:** the data $X$, the undirected edges $E_{must}$ that must appear in the true undirected graph $G_{true}$ according to prior knowledge, the undirected edges $E_{test}$ ($E_{must} \cap E_{test} = \emptyset$) whose existences are to be tested from the data $X$, and the FDR level $q$ for testing $E_{test}$.

**Output:** an undirected graph $G$.

1: Initialize the undirected edge set $E$ as $E = E_{test} \cup E_{must}$, and form an undirected graph $G$ from $E$ accordingly.
2: Initialize the maximum $p$-values associated with the edges in $E_{test}$ as $P_{max} = \{p_{a \sim b} = -1\}_{a \sim b \in E_{test}}$.
3: Let depth $d = 0$.
4: repeat
5:  for each ordered pair of vertices $a$ and $b$ that $a \sim b \in E_{test}$ and $|\text{adj}(a, G) \setminus \{b\}| \geq d$ do
6:   for each subset $C \subseteq \text{adj}(a, G) \setminus \{b\}$ and $|C| = d$ do
7:    Test hypothesis $X_a \perp X_b | X_C$ and calculate the $p$-value $p_{a \perp b | C}$. If $p_{a \perp b | C} > p_{a \sim b}^{max}$, then let $p_{a \sim b}^{max} = p_{a \perp b | C}$.
8:   if every element of $P_{max}$ has been assigned a valid $p$-value and $p_{a \sim b}^{max}$ is just updated by step 7 then
9:    Apply the FDR procedure to $P_{max}$ to control the FDR under $q$. Let $E_{negative}$ denote the edges whose non-existences are accepted by the FDR procedure. Remove $E_{negative}$ from $E$ and also remove their corresponding elements from $P_{max}$. Update $G$ accordingly.
10: if $a \sim b$ is removed, then
11:   break the for loop at line 6
12: end if
13: end for
14: end for
15: end for
16: Let $d = d + 1$.
17: until $|\text{adj}(a, G) \setminus \{b\}| < d$ for every ordered pair of vertices $a$ and $b$ that $a \sim b$ is in $E \cap E_{test}$.

The extended PC_{fdr*} algorithm is applicable to dynamic Bayesian networks to which the PC_{fdr*} algorithm is not. A multi-channel stochastic process can be modelled with a dynamic Bayesian network of $M \times T$ nodes, where $M$ and $T$ denote the number of
5.1. Integration of Prior Knowledge

Figure 5.1: All the possible connections (with the directions ignored) of a first-order dynamic Bayesian network for a two-channel stochastic process channels and the number of time points respectively, and each node represents the signal of a channel at one time point. Because the future can influence neither the presence nor the past, nodes at time $t + 1$ can have only nodes after time $t$ as their children. To reduce the complexity of the model, the order of dynamics is usually restricted. For example, it may be assumed that nodes at time $t$ cannot have nodes at or after $t + 2$ as their children. In these cases, many connections must be excluded from the network. To reduce the complexity further, the same connection pattern is usually be assumed to repeat overtime. Figure 5.1 shows all the possible connections (with the directions ignored) of a first-order dynamic Bayesian network for a two-channel stochastic process. Because this is a first-order dynamic model, connections from $a_1$ and $b_1$ to $a_3$ and $b_3$ should be excluded. If the same connection pattern is assumed to repeat, then all the possible connections in the repeating pattern can be represented by the sub-network circled by dots. The repeating connection pattern can be learned with the extended PC$_{fdr^*}$ algorithm by excluding the connection $a_1 \sim b_1$ from both $E_{must}$ and $E_{test}$. The original PC$_{fdr^*}$ algorithm cannot incorporate the exclusion of $a_1 \sim b_1$.

5.1.3 Conditional-Independence Test for fMRI Signals

A basic modular of the extended PC$_{fdr^*}$ algorithm is to test whether two random variables $X_a$ and $X_b$ are conditionally independent or not upon a set of other random variables $X_C$. $X_a$ and $X_b$ are conditionally independent given $X_C$ if and only if $P(X_a, X_b|X_C) = P(X_a|X_C)P(X_b|X_C)$. 
5.1. Integration of Prior Knowledge

fMRI data are continuous, so we modelled the fMRI signals of multiple brain regions with multi-variate Gaussian stochastic processes. The t-test on the partial correlation coefficient is the most widely used test of conditional independence for multi-variate Gaussian distributions. If \([X_a, X_b, X_C]\) follows a multi-variate Gaussian distribution \(N(\mu, \Sigma)\), then the partial correlation coefficient between \(X_a\) and \(X_b\) given \(X_C\), denoted as \(\rho_{ab|C}\), is

\[
\rho_{ab|C} = -\frac{k_{ab}}{\sqrt{k_{aa}k_{bb}}},
\]

(5.3)

where \(k_{ab}\), \(k_{aa}\) and \(k_{bb}\) are the elements of \(\Sigma^{-1}\) corresponding to \(X_a\) and \(X_b\). \(\rho_{ab|C}\) is the correlation coefficient between \(X_a\) and \(X_b\) after the effect of \(X_C\) has been removed. If and only if \(X_a\) and \(X_b\) are conditionally independent given \(X_C\), then \(\rho_{ab|C}\) is zero \([6]\). The sample partial correlation coefficient \(r_{ab|C}\) can be calculated in a similar way by replacing \(\Sigma\) with the covariance matrix estimated from the data. Under the null hypothesis of conditional independence, if \(r_{ab|C}\) is estimated from \(S\) identical independent samples, then the statistic

\[
t = \sqrt{S-2-|C|} \frac{r_{ab|C}}{\sqrt{1-r_{ab|C}^2}}
\]

(5.4)

follows the t-distribution with \(S-2-|C|\) degrees of freedom, and the two-tailed t-test can be used to test the null hypothesis \([7]\).

5.1.4 Test of Relative Consistence Among Graphs

The group analysis was conducted with the individual-structure approach, i.e. to learn the connectivity networks individually for each subject, and then to extract those connections that are consistently recruited by subjects in the same experimental group.

If the appearance frequency of a certain connection among a group of graphs is statistically significantly higher than the result of randomness, the connection is considered to be relatively consistently recruited. Suppose there are \(N\) graphs each containing \(L_n\) \((n = 1, \ldots, N)\) connections respectively and the total number of possible connections of
a graph is $L$. If connections in the same graph are recruited with equal chances, then given $L_n(n=1,\ldots,N)$, the number of a connection’s appearances in the $N$ graphs is a random number $Y = \sum_{n=1}^{N} X_n$ where $X_n \sim \text{Bernoulli}(L_n/L)$. Under this hypothesis of randomness, the probability that a connection appears $y$ times or more is $P(Y \geq y)$. We applied the test on relative consistence to all the possible connections, adjusted the effect of multiple testing with the FDR procedure, and finally selected those whose $q$-values are lower than 5% as relatively consistent connections to compose a connectivity network at the group level.

5.1.5 Application to Parkinson’s Disease

**Data Collection:** The study was approved by the University of British Columbia ethics board and all subjects gave written informed consent prior to participating. Ten healthy people and ten Parkinson’s disease patients participated in the study. While in the MRI scanner, subjects were instructed to squeeze a rubber bulb with their left hand at four frequencies (0.00Hz, 0.25Hz, 0.5Hz and 0.75Hz) in 30s blocks, arranged in a pseudo-random order. The patients performed the experiment twice, once before L-dopa medication and the other after the medication. fMRI data of their brain activities was collected with a Philips Achieva 3.0 T scanner. The following twelve brain regions in the Talairach atlas were selected with the local linear discriminant analysis [9] as the regions of interest (ROI) in the study: the left and right (denoted as prefixes “L” or “R” in the abbreviated names of ROIs) – primary motor cortex (L_M1 and R_M1), supplementary motor cortex (L_SMA and R_SMA), lateral cerebellar hemisphere (L_CER and R_CER), putamen (L_PUT and R_PUT), caudate (L_CAU and R_CAU), and thalamus (L_THA and R_THA).

**Preprocessing:** After motion correction, the fMRI time courses of the voxels within each ROI were averaged as the summary activity of the ROI. Then, the averaged time courses were detrended and normalized to unit variance.
5.1. Integration of Prior Knowledge

Results:

The connectivity networks of the normal people, the patients before and after the medication are shown in Figure 5.2. The graphs are obviously not results of randomness. For example, the connections between the left ROIs and their right counterparts appear in all the three graphs. Since all the subjects performed the same motor task, it is reasonable for the networks to share certain similarity. This also suggests that controlling the error rate yields robust results, curbing the effect of randomness. We observe that after the L-dopa medication, the network of the patients changed toward that of the normal people. Connections L_SMA—L_M1, L_THA—L_THA and R_THA—R_THA appear in the network of the normal people, but do not in that of the patients before the medication. After the medication, these connections appear in the network (Figure 5.2c) again. Connection L_CAU—R_THA does not appear in the network of the normal people, but does in that of the patients before the medication. After the medication, this connection disappears in the network (Figure 5.2c). These observations are consistent with the fact that L-dopa has dramatic effects against bradykinesia and rigidity, cardinal features of Parkinson’s disease [5].
5.1. Integration of Prior Knowledge

![Diagram](image)

(a) Normal Subjects

(b) Patients Before Medication

(c) Patients After Medication

Figure 5.2: The relatively consistent connections at the group level. Edge labels are the $q$-values of the edges in the group analysis. Prefixes “L” and “R” before ROI names are short for “left” and “right” respectively.
5.2 Bayesian Inference with FDR-Controlled Prior

The PC_{fdr} and the PC_{fdr*} algorithms can make fast and reliable inference of the structure of a Bayesian networks, with the false discovery rate being controlled at user-specified levels. If the data were perturbed away from the generating model by additional noise, the algorithms would not to control the FDR precisely, but rather yield a fast and rough estimation of the generating model. On the other hand, Bayesian inference is relatively robust to perturbation, but needs cumbersome computation. For example, sometimes even the overhead “burn-in” process of its generic implementation, Markov chain Monte Carlo, needs thousands of iterations.

We propose a post-hoc method to combine the PC_{fdr} algorithms and Bayesian inference together, taking the advantages from both of them. The basic idea is to first learn a network with the PC_{fdr} algorithms, and then for Bayesian inference propose a prior distribution of network structures based on the learned network. This method may accelerate Bayesian inference and is more robust to perturbing noise.

5.2.1 Bayesian Inference with Uniform Prior Distribution

One intuitive approach to controlling error rates is Bayesian inference, i.e. inferring the posterior probability of network structures from the observed data based on a certain prior probability distribution. The uniform distribution is usually employed as a non-informative prior. As a powerful, generic tool for Bayesian inference, Markov chain Monte Carlo (MCMC) has been widely employed for structure learning of DBNs. MCMC sampling is usually implemented as follows: A neighborhood is first defined for each network structure, usually by adding, deleting or reversing an edge of the network.
Then, starting from a randomly selected network structure, the chain iteratively moves from one structure to one of its neighbors according to a certain probability derived from the Bayesian factor between them. Instead of sampling in the space of directed acyclic graphs, Chickering suggested sampling in the space of equivalence classes $[3]$. In our following simulations, we will use MCMC to sample network structures in the space of equivalence classes according to their posterior probabilities derived from the uniform prior.

### 5.2.2 Modelling with Prior Distribution Derived from FDR-Controlled PC Algorithm

This proposed method, named as the $PC_{fdr}$-Prior procedure, is summarized as follows:

1. Learn the network structure $S$ using the $PC_{fdr}$ or the $PC_{fdr*}$ algorithm.

2. Derive a prior probability distribution $D$ over the network structures, parameterizing our confidence of $S$.

3. Make Bayesian inference with the prior distribution $D$.

In Step 2, we use the realized FDR and the realized detection power to parameterize our confidence of $S$, the output of the $PC_{fdr}$ or the $PC_{fdr*}$ algorithms. Because both the realized FDR and the realized detection power are random variables within $[0, 1]$, it is natural to model them with beta distributions, as in Eqs. (5.5) and (5.6), where $G$ is the existing, yet unknown true network structure, and $|S|$ denotes the number of edges in $S$.

$$fdr = \frac{|S \cap G|}{|S|} \sim Beta(a_{fdr}, b_{fdr}). \tag{5.5}$$

$$\text{power} = \frac{|S \cap G|}{|G|} \sim Beta(a_{\text{power}}, b_{\text{power}}). \tag{5.6}$$
We employ the following process to generate network structures from the prior distribution $D$. First, generate random numbers $fdr$ and $power$ from Eqs. (5.5) and (5.6). Second, select $|S|\cdot fdr$ undirected edges from network $S$, and $|S|\cdot fdr/power - |S|\cdot fdr$ from those not in $S$. Finally, determine the directions of the edges randomly according a random order of the nodes.

If the PC$_{fdr\ast}$ algorithm is used in step 1 to point out this specification, we will call the method the PC$_{fdr\ast}$-Prior method, but the name “PC$_{fdr}$-Prior” is still more general than specific, still covering the choice between the PC$_{fdr}$ and PC$_{fdr\ast}$ algorithms.

We must point out that the PC$_{fdr}$-Prior procedure is not rigorously correct from the view of Bayesian inference, because the information in the data is used twice during the inference, once in step 1 and the other in step 3. However, we justify the procedure as, which we all, a “third-party” Bayesian inference. Traditional Bayesian inference is a process involving two parties: the observer and the data. The observer has prior knowledge directly about the model, and then updates his knowledge with the information in the data. In real world, inference processes may involve not just the observer and the data, but also, for example, an extra “counsellor”. Let us consider a consulting procedure as follows. The observer has certain prior knowledge about the problem, but not directly about the model, rather, indirectly about the counsellor’s ability to read information from the data. The observer first hands the data to the counsellor, and according to his trust in the counsellor, he accepts a prior based on the counsellor’s report, and later he studies the data by himself with the accepted prior. In the PC$_{fdr}$-Prior procedure, the PC$_{fdr}$ algorithms play the role of a counsellor in step 1. Equations (5.5) and (5.6) for step 2 describe the observer’s trust in the PC$_{fdr}$ algorithms, that is his indirect prior knowledge about the problem. In step 3, the observer makes Bayesian inference by himself.

Although the correctness of this “third-party” Bayesian inference is still arguable, we decided to keep it in the thesis for the following reasons. First, it provides an approach for
fusing different data analysis methods. Second, we do not want to be over conservative about new yet arguable technologies, but would rather like to protect the bud of potential innovations. When Bayesian inference was first introduced, statisticians argued about it, and some did not accept it. Nowadays, it has become an active and important branch of statistics. This “third-party” Bayesian idea, though is arguable, has a real-world consulting procedure to justify it, so we keep it and leave it for investigation in the future.

Alternatively, to be theoretically correct from the view of Bayesian inference, instead of using the outputs of the PC_{fdr} algorithms as the prior, we can use such outputs as the initialization of the standard MCMC sampling. This usage can accelerate the burn-in process of MCMC, and significantly reduce the computational time. This idea is strongly supported by the results in Section 5.2.3.

5.2.3 Simulation

Synthetic Data  Data were generated from a dynamic Bayesian network, with the network structure as shown in Fig. 5.3 and with the conditional-dependence relationships defined in Eqs. (5.7) and (5.8):

\[ x_a = \sum_{b \in \text{pa}[a]} \beta_{ba} x_b + e_a, \quad (5.7) \]
\[ y_a = x_a + \epsilon_a, \quad (5.8) \]

where \( a \) and \( b \) denote node indices, \( \text{pa}[a] \) denote the parent nodes of \( a \), \( \beta_{ba} \) is the connection coefficient from \( b \) to \( a \), and \( \epsilon_a \) and \( \epsilon_a \) are the additive Gaussian noise related to node \( a \). \( e_a \) represents the intrinsic randomness of the stochastic system, while \( \epsilon_a \) is the perturbation noise occurring during data acquisition. It should be noted that due to the \( \epsilon_a \) term, Fig. 5.3 represents only the conditional-independence relationships among \( \{x_a, x_b, \ldots\} \), but not necessarily those among the observations \( \{y_a, y_b, \ldots\} \). However,
5.2. Bayesian Inference with FDR-Controlled Prior

Figure 5.3: The simulation model. Solid blue arrows are edges from time $t$ to $t+1$; dashed red arrows are edges without time lags.

$\{y_a, y_b, \ldots\}$ but not $\{x_a, x_b, \ldots\}$ were available for structure learning. Here we included $\epsilon_a$ and analyzed the observations $\{y_a, y_b, \ldots\}$ deliberately to investigate the effect of noise perturbation in practice.

Connection coefficients $\beta_{ba}$ were randomly selected from the uniform distribution on $[0.2, 0.4]$, and the variances of $\{\epsilon_a\}$ were randomly selected from the uniform distribution on $[0.2, 1.2]$. From a single model for $\{x_a, x_b, \ldots\}$, data of 200 time points and 500 time points were sampled repetitively 50 times. Based on the signals $\{x_a, x_b, \ldots\}$, we added $\epsilon_a$ at the signal-to-noise ratio (SNR) levels 16, 9, 4 and 3 to synthesize the observations $\{y_a, y_b, \ldots\}$.

**Results** Figs. 5.4 and 5.5 respectively show the FDRs and the detection powers of the PC$_{fdr}$-Prior method, the PC$_{fdr^*}$ algorithm and the Bayesian inference based on the uniform prior distribution. We note that, due to the perturbation noise, none of the algorithms controlled the FDR closely around the expected level 5%. It should be pointed out that the performance of the PC$_{fdr^*}$ algorithm without perturbation noise
5.2. Bayesian Inference with FDR-Controlled Prior

Figure 5.4: The estimated false discovery rate (FDR). The x-axis is the level of the perturbation noise, in terms of $1/\sqrt{SNR}$; the y-axis is the FDR. The bars are the standard deviation of the estimations.

has been investigated in Chapter 4 and it controlled the FDR accurately around 5% without perturbation noise. The data used in the simulation here have been perturbated from the true generating model by the extra noise in Eq. (5.8). The empirical FDRs of the Bayesian inference based on the uniform prior distribution (implemented with MCMC of 5000 steps) are higher than those of the other two algorithms, especially in the case of 200 time points. Meanwhile, its detection power is lower than that of the PC\textit{fdr*} algorithm, and merely higher than that of the PC\textit{fdr}-Prior method. As the perturbation level increases, the FDR of the PC\textit{fdr}-Prior method increases slower than that of the PC\textit{fdr*} algorithm. This suggests that the post-hoc Bayesian average makes the structure-learning more robust to perturbation noise. The PC\textit{fdr*} algorithm and the Bayesian inference based on the uniform prior distribution steadily detected more true edges than the PC\textit{fdr*-Prior} method did, nevertheless at the cost of higher FDRs, as shown in Fig. 5.4. In the case of 500 time points, all the three methods detected more than 80% of the true connections. In the case of 200 time points, the detection power dropped sharply as the perturbation noise increased.

Though the theoretical foundation of this post-hoc Bayesian inference, the “third-party” Bayesian inference, is not well established, step 2 of the PC\textit{fdr}-Prior method still can be used to initialize standard MCMC sampling, accelerating the burn-in process,
5.2. Bayesian Inference with FDR-Controlled Prior

Figure 5.5: The estimated detection power. The x-axis is the level of the perturbation noise, in terms of $1/\sqrt{SNR}$; the y-axis is the detection power. The bars are the standard deviation of the estimations.

and significantly reducing the computational cost. This usage does not violate the rule of Bayesian statistics. For the case of SNR($\epsilon_a$)=16 and 200 time points, Fig. 5.6 shows the average trajectory of the Bayesian-information-criterion (BIC) scores of the MCMC sampling based on the posterior distribution derived from the uniform distribution. The chain became mature after about 2500 steps (BIC = $-6511.4$) which took more than an hour with Matlab on our workstation (Intel Xeon 1.86GHz CPU and 4G RAM). The horizontal histogram is the BIC scores sampled from the prior distribution built in Step 2 of the PC$_{fdr}$-Prior method. According to the histogram, the probability $Pr(\text{BIC} \geq -6511.4)$ is approximately 0.3791, implying that if we randomly sample 20 networks from the distribution, with probability $1 - (1 - 0.3791)^{20} = 99.99\%$ we can find a BIC score which takes the MCMC trajectory about 2500 steps to achieve. The overhead of initializing MCMC with the PC$_{fdr}$ algorithms is to first preliminarily learn a network structure with the PC$_{fdr}$ algorithms. For the simulation presented here, it took no more than 4 seconds for the case of 200 samples, and around 10 seconds for the case of 500 samples.
5.2. Bayesian Inference with FDR-Controlled Prior

![Graph showing BIC scores](image)

Figure 5.6: The histogram of the BIC scores sampled from the prior distribution built in Step 2 of the PC\textsubscript{fdr}-Prior method, and the average trajectory of the Bayesian-information-criterion (BIC) scores during the MCMC sampling for Bayesian inference based on the uniform prior distribution.

5.2.4 Application to Parkinson’s Disease

**Data collection** Ten normal people and ten Parkinson’s disease (PD) patients participated in the study. Subjects continually squeezed a bulb in their right hand to control an “inflatable” ring so that the ring moved through an undulating tunnel without touching the sides. A trial of the task was five minutes. Normal people performed only one trial; the patients performed twice, once before L-dopa medication, the other after the medication.

fMRI data were collected with a Philips Achieva 3.0 T scanner at 0.5038 Hz. The following eighteen brain regions were selected as regions of interest (ROIs) in the study: the left and right (denoted as prefixes “L” or “R” in the abbreviated names of ROIs) primary motor cortex (L\textsubscript{M1} and R\textsubscript{M1}), supplementary motor cortex (L\textsubscript{SMA} and R\textsubscript{SMA}), lateral cerebellar hemisphere (L\textsubscript{CER} and R\textsubscript{CER}), putamen (L\textsubscript{PUT} and R\textsubscript{PUT}), caudate (L\textsubscript{CAU} and R\textsubscript{CAU}), thalamus (L\textsubscript{THA} and R\textsubscript{THA}), anterior cingulate cortex (L\textsubscript{ACC} and R\textsubscript{ACC}), globus pallidus (L\textsubscript{GLP} and R\textsubscript{GLP}), and prefrontal cortex (L\textsubscript{PFC} and R\textsubscript{PFC}).

After motion correction, the fMRI time courses of the voxels within each ROI were
averaged as the summary activity of the ROI. The averaged time courses were then detrended and normalized to unit variance. Data were grouped into three categories: group N for the normal people, group $P_{\text{pre}}$ for the PD subjects before medication, and group $P_{\text{post}}$ for the PD subjects after taking L-dopa medication. Data in the same group were pooled together for group analysis.

**Results** The connectivity networks learned with the $PC_{fdr}$-Prior method are shown in Fig. 5.7. We argue that the results are not random, but informative. Fig. 5.7(d) shows the connections that are shared by all the three groups, N, $P_{\text{pre}}$ and $P_{\text{post}}$. Most of these connections are either auto-regression to the same ROIs, or between left and right counterparts. The pathway GLP – PUT – CAU is also symmetric.

Fig. 5.7(e) shows those connections that are shared by the N and $P_{\text{post}}$ groups, but not $P_{\text{pre}}$. In other words, we can regard these connections as “impaired” by the disease but later “recovered” by the L-dopa medication. The pathway $L_{\text{M1}} – L_{\text{SMA}} – L_{\text{PFC}}$ is compatible with recruitment of the left sensorimotor system. Interactions between the SMA and M1 have previously been demonstrated to be abnormal in PD, and later improved with medication [2, 4].

Fig. 5.7(f) shows those connections that are only in the $P_{\text{pre}}$ group, but in neither the N nor the $P_{\text{post}}$ group. A prominent feature is the importance of the $L_{\text{CER}}$, the contralateral cerebellum. The cerebellum has recently been shown to be hyperactive in PD [12], and may represent a compensation-mechanism of the disease Parkinson’s disease. Our results extend prior observations by demonstrating that not only does the amplitude of the fMRI signal increase in the cerebellum but also the connections to other brain regions.
5.2. Bayesian Inference with FDR-Controlled Prior

Figure 5.7: The functional connectivity networks learned from the bulb-squeezing data with the PC_fdr-Prior method. The prefixes “L” and “R” before ROI names are short for “left” and “right” respectively. Solid blue edges are those with a one-time-point lag; dashed red are those without time lags.
5.3 Conclusions and Discussions

Graphical models have been increasingly investigated as an exploratory tool for inferring brain connectivity in fMRI analyses. It is critical to control the error rate in the “discovered” connectivity networks in real fMRI studies. Two extensions of the PC$_{fdr}$ algorithms are designed to introduce the control over the false discovery rate to dynamic Bayesian networks.

The first one is an adaptation to prior knowledge. It allows users to specify which edges must appear in the network, which cannot, and which are to be learned from data. It is naturally applicable to dynamic Bayesian networks by simply regarding dynamic Bayesian networks as certain special prior constraints. In a study on Parkinson’s disease using functional magnetic resonance imaging, it showed promising performance. It learned connectivity networks that are consistent with the dramatic effects of L-dopa against bradykinesia and rigidity, cardinal features of the disease. It yielded robust results, curbing the effect of randomness.

The second one is using the PC$_{fdr}$ algorithms to improve the Bayesian inference of dynamic Bayesian networks. The basic idea is to make a post-hoc Bayesian inference based on the prior distribution derived from the network structure learned with the PC$_{fdr}$ algorithms. Simulations showed its superior performances in terms of both computational cost and robustness to perturbation noise. In a study on Parkinson’s disease using functional magnetic resonance imaging, it detected the normalizing effect of L-dopa medication in motor pathways, as well as regions involved in the compensatory mechanisms of the disease. This post-hoc Bayesian inference is not rigorously correct from the view of Bayesian statistics, because the information in the data is used twice during the inference. However, it can be justified with a new idea, which we call, “third-party” Bayesian inference. Though it is arguable, it describes real-world consulting procedures.

Though the theoretical foundation of this post-hoc Bayesian inference, the “third-party” Bayesian inference, is not well established, step 2 of the PC$_{fdr}$-Prior method still
can be used to initialize standard MCMC sampling, accelerating the burn-in process, and significantly reducing the computational cost. This usage does not violate the rule of Bayesian statistics.

The proposed structure-learning algorithms are based on the PC-algorithm [10] which does not consider latent variables. If certain brain regions in the functional system are not involved in the structure learning, then a connection between two brain regions may be a result of either their direct communication or the common exertion over them by a latent region. To distinguish whether two brain regions are directly connected or both driven by a third but uninvolved brain region, the Fast-Causal-Inference (FCI) algorithm [10] which considers latent variables should be used as the analysis framework.
Bibliography


Chapter 6

Selecting Regions of Interest in fMRI Analysis

Using Local Linear Discriminant Analysis

6.1 Introduction

Group analysis in fMRI is typically done in several consecutive steps. First, fMRI data are corrected for motion, despite the fact that most methods cannot easily distinguish changes in fMRI signal from that induced by motion \[11\] \[12\]. Data are then spatially transformed to a common space such as the atlas by Talaraich \[23\] or the probabilistic space suggested by the Montreal Neurological Institute \[3\] to minimize intersubject differences. However, because of the variability in human brain anatomy, the intersubject registration is typically imperfect, so spatial low-pass filtering ("smoothing") is performed to de-emphasize anatomical differences \[6\]. Once data have been motion corrected, warped to a common space, and spatially smoothed, the task-related activation of a voxel of a subject \(k\) is estimated with linear regression techniques:

\[
Y_k = X_k \beta_k + \varepsilon_k, \quad \text{and} \quad Cov(\varepsilon_k) = \sigma^2_k V_k
\]  

\[6.1\]

A version of this chapter has been accepted for publication. Martin J. McKeown, Junning Li, Xuemei Huang, Mechele M. Lewis, Seungshin Rhee, K.N. Young Truong and Z. Jane Wang (2007) Local Linear Discriminant Analysis (LLDA) for Group and Region of Interest (ROI)-Based fMRI Analysis. NeuroImage 37: 855–865.
where $Y_k$ is the $T_k \times 1$ time course of the voxel, $X_k$ is the $T_k \times D$ design matrix containing the hypothesized activation (often incorporating estimates of the hemodynamic response function) as well as other covariates, $\varepsilon_k$ is the $T_k \times 1$ vector of residuals, $\sigma_k^2$ is the homogeneous variance of the residuals and $V_k$ is the correlation matrix. The subscript $k$ indicates that all the variables are related to subject $k$.

As fMRI data are typically not temporally white, data are often pre-whitened using a whitening matrix $W_k$ such that:

$$W_k V_k W_k^T = I \quad (6.2)$$

(for an excellent summary the reader is referred to: [16]). If each term in $(6.1)$ is pre-multiplied by $W_k$, we have:

$$Y_k^* = X_k^* \beta_k + \varepsilon_k^* \quad (6.3)$$

where the superscript * denotes the whitened quantities. The whitening matrix $W_k$ is estimated by the residuals $\varepsilon_k$ and $V_k$ as:

$$W_k = V_k^{-1/2}. \quad (6.4)$$

The regression estimates of Eq. (6.3) can then be estimated by Ordinary Least Squares (OLS) to give the Generalized Least Squares estimate of Eq. (6.1):

$$\hat{\beta}_{GLS}^k = \left(X_k^{*T} X_k^*\right)^{-1} X_k^{*T} Y_k^* \quad (6.5)$$

$$\text{Cov} \left(\hat{\beta}_{GLS}^k\right) = \sigma_k^2 \left(X_k^{*T} X_k^*\right)^{-1} \quad (6.6)$$

Contrasts between conditions are of most interest in an experiment, e.g. contrasting BOLD signal during performance of a given task compared to rest. In the study on a
single subject, the null hypothesis is that the contrast between the least-squared estimates is zero:

\[ H_0 : c\beta_k = 0 \]

where \( c \) is the contrast row vector. For example, if we are interested in the comparison between task 1 and task 2, \( c \) is \([1, -1]\).

Group analyses are usually done using a Summary Statistics method, which is a two-staged approach; first individual models are fit to each subject as described above, and then a second level is applied to make group inference on the \( c\beta_k \) [16]. In the usual situation where one is contrasting activation across two groups, the second level is a multivariate regression equation with the design matrix encoded with group inclusion indicators (Figure 6.1 (a)):

\[
\beta_{cont} = X_g\beta_g + \varepsilon_g
\]  
(6.7)

where \( X_g = \begin{pmatrix} 10 \\ ... \\ 10 \\ 01 \\ ... \\ 01 \end{pmatrix} \) is a binary \( K \times 2 \) matrix coded to show group inclusion, \( K \) is the number of subjects from the two groups), \( \beta_{cont} \) is composed of the contrasts \( c\beta_k \) for each individual as defined in the first stage, \( \beta_g = [\beta_{g1}, \beta_{g2}]^T \) is mean activation of the two groups and \( \varepsilon_g \sim N(0, \delta^2_g V_g) \) is the residual with the variance \( \delta^2_g \) and the correlation matrix \( V_g \) being a diagonal matrix, typically just \( I \). Here the null hypothesis is that the group activations for a given voxel in the common spatially transformed space are not significantly different:
Figure 6.1: Comparison of fMRI group analysis methods. (a) With a standard GLM approach, a two step approach is typically employed. An individually-specific general linear model is first used to estimate task-related activity. In the second stage, the individual contrasts from each subject are collected with a dummy matrix encoding group membership. Note that this implicitly implies that each voxel across subjects is comparable, i.e. each subject’s data have been spatially warped to the same space. (b) With the LLDA approach, voxel-based, activation t-statistics from different ROIs are compared for each subject. The optimum combination of ROIs resulting in maximal discriminability (represented by the straight lines and the Ui’s) is calculated. Note that a joint optimization is performed (represented by the curved arrows), so that the each Ui is not calculated in isolation. (lower panel) The individual Ui’s can then be (weighted) averaged to get an overall linear combination of ROIs that maximally discriminate between tasks.
6.1. Introduction

\[ H_0 : \beta_{g1} - \beta_{g2} = 0. \]

A number of different implementations have been proposed to implement the above analysis in a practical way. The fMRIStat method uses Restricted Maximum Likelihood (ReML) to estimate \( \sigma^2_g \) \[^{[25]}\], then smoothes the data to increase its degree of freedom and accuracy and finally tests the hypothesis with \( t \)-statistics. The SPM2 package \[^{[7,8]}\] estimates the \( \delta^2V + \delta^2_g V_g \) term with ReML under a simplifying assumption that all the subjects share a common covariance matrix \( \delta^2_k V_k = \delta^2 V \), and then tests the hypotheses with \( F \) statistics. The FMRIB software library estimates \( \sigma^2_g \) with the maximum \textit{a posteriori} (MAP) criteria, then screens obviously insignificant voxels with \( Z \)-statistics and finally performs a Bayesian inference on the significance of the remaining voxels with a slower but more accurate Markov Chain Monte Carlo (MCMC) simulation \[^{[1]}\].

Nevertheless, there are a number of shortcomings with the previously described methods. The above methods work on the voxel level — this assumes that after suitable spatial transformation, there is a perfect correspondence between the same voxel across subjects. While this may be mitigated somewhat by spatial smoothing, such low-pass filtering degrades the spatial resolution of the data. Activation estimates in small, subcortical structures such as the basal ganglia or thalami, which abut functionally different tissues (e.g. the internal capsule), may be particularly affected by mis-registration errors.

One way to partially circumvent the difficulties associated with spatially transforming functional maps to a common space is to manually draw anatomical regions of interest (ROIs) for each subject, and performing analyses at the ROI level — as opposed to the individual voxel level. Using standard atlases, a particular brain region (e.g. the lateral cerebellar hemisphere) is manually circumscribed on the high-resolution structural MRI scans that have been co-registered with the functional data, and the voxels within this region are analyzed. The benefit of this method is that it does not require rigid spatial transformation, preventing possible gross distortion of a particular brain area, as
may occur if the anatomy of a given individual differs significantly in size and shape to the homologous area in the exemplar brain. However, drawing ROIs is labor-intensive, subject to human error, and requires the assumption that a functionally active region (the SMA for example) of a given brain will be within an anatomically standardized index (i.e. Broadman’s Area 9) which is used to draw the ROI.

In addition to the possibilities of mis-registration, the previously-described voxel-based methods do not explicitly model interactions between brain regions. Covarying regions are often of interest, but are not included in the group methods described above. Conceptually, group methods are done in two stages: in the first stage, individually-specific regression models are fit to the data; and then the results of these models are used in a group-level analysis. Because the goal of these methods is to test a specific hypothesis, these methods may be conducted sequentially. In contrast, if the goal is to find which combination of brain regions are maximally different between tasks, it is desirable to jointly optimize the individual statistical model and the overall models simultaneously.

In individual-subject fMRI analysis, in addition to hypothesis driven methods, there is a role for data driven methods, such as Independent Component Analysis (ICA) [2,14], which do not need rigorous a priori specification of activation patterns. In an analogous manner, there may be particular interest in discovering the combinations of brain regions (specified by ROIs) that are maximally contrasted during performance of certain tasks (Figure 6.1 (b)). There is therefore a need for a multivariate, discriminant analysis approach that works at the region of interest (ROI) level as opposed to the individual voxel level.

Previous work has taken individual activations (or the T-statistics associated with them) and used a multivariate discriminant approach [15]. In order to apply a discriminant approach, we first assume that some statistical analysis has been performed to assign a $t$-statistic, related to activation, to each voxel, such as simple $t$-test or regres-
sion (e.g. \( t \)-statistic of a voxel comparing mean BOLD signal for a given condition to BOLD signal for that voxel at rest). Since most multivariate discriminant approaches assumes Gaussian distributions, we extract features from the \( t \)-statistics with bootstrap methods to ensure the Gaussian assumption. Consider a subject \( k \), we randomly select a voxel from within each of its \( N \) ROIs, and assemble the result into a column \( t \)-statistic vector:

\[
t(k, V) = [t(k, 1, v_1), t(k, 2, v_2), \ldots, t(k, N, v_N)]^T, \tag{6.8}
\]

where \( t(k, r, v_r) \) is the \( t \)-statistic of the \( v_r \)th voxel in the \( r \)th ROI of subject \( k \) and the voxel index \( V = [v_1, v_2, \ldots, v_N]^T \). The random selection is repeated a number of times, say \( B_t \) times, and the \( i \)th draw is notated as \( t(k, V_i) \). After a reasonable number of draws, (e.g., \( B_t = 30 \)), the mean of the \( t \)-statistic vectors is taken as feature to ensure the data can be modeled as multivariate Gaussian:

\[
f_k = \frac{1}{\sqrt{B_t}} \sum_{i=1}^{B_t} t(k, V_i). \tag{6.9}
\]

The above process is repeated \( B_f \) times (\( B_f \) is several hundreds or thousands) and all feature vectors are then collected:

\[
F_k = [f_k(1), f_k(2), \ldots, f_k(B_f)], \tag{6.10}
\]

where \( f_k(i) \) is the \( i \)th random sample of a feature vector. This process is then repeated for all \( S \) subjects in the groups and all the \( F_k \)'s are concatenated into a big matrix

\[
X = [F_1, F_2, \ldots, F_S]. \tag{6.11}
\]

The benefit of formulating the problem in this way is two fold; the \( F_k \)'s will be asymptotically normally distributed (by the multivariate central limit theorem [5]) even if the
6.1. Introduction

Voxel-based statistics are not, and that \( p \)-variate linear analyses can now be performed on \( X \) \cite{5}.

The variability of the amplitude of activation across subjects is well known and typically this is dealt with by using a random effects analysis \cite{16}. In random-effects models, the data are assumed to be derived from a hierarchy of different populations whose differences are constrained by the hierarchy. Each subject from a different group (e.g. group of normal subjects) is considered to be representative of the entire population of normal subjects.

In some cases the magnitude of inter-subject differences in fMRI activation can exceed the task-specific differences within individuals. In order to deal with this situation, yet still maintaining the benefits of linear discriminant analysis, we propose using a recently developed solution, Local Linear Discriminant Analysis (LLDA), that was initially designed to solve a somewhat different, but still related, problem \cite{10}. In discriminant analysis trying to classify images of faces, often the difference in discriminant features of different poses of the same face can greatly exceed the difference in discriminant features between faces. Finding a classifier that is sensitive to images from different subjects, yet insensitive to different poses from the same subject, has been problematic. Kim and Kittler suggested LLDA as a solution to this problem \cite{10}. We therefore propose using LLDA to sensitively discriminate between task-dependent ROI-based patterns of activity, while being relatively robust to the differences between subjects (Figure 6.1 (b)).

We apply this method to data derived from a motor paradigm that would be expected to activate cortical and subcortical structures. We show that the proposed method, consistent with prior neuroscience knowledge derived from animal models, detects significant group activation in subcortical structures that was not present when the same group of data were analyzed using standard methods utilizing spatial normalization.
6.2 Experimental Methods

To demonstrate the proposed method, we utilized fMRI data that would be expected, based on prior knowledge, to activate subcortical structures. We enrolled 10 healthy volunteers, 5 males and 5 females, (range 27-45 years). All subjects were right hand dominant, and had normal neurological examinations, had no history of neurological disease, and were not currently using of any psychoactive prescription medications. Handedness was determined according to Edinburgh Handedness Inventory. The paradigm consisted of externally guided (EG) or internally guided (IG) movements based on three different finger sequencing movements (FSMs) performed alternatively by either the right or left hand. For FSM #1, subjects had to (a) make finger-to-thumb opposition movements in the specific order of the index, middle, ring and little finger; (b) open and clench the fist twice; (c) complete finger-to-thumb oppositions in the opposite order (i.e., little, ring, middle and index finger); (d) open and clench the fist twice again; and then (e) repeat the same series of movements. The FSM #2 was the same as above except the sequence for (a) changed to index, ring, middle and little fingers and (c) changed to the reversed order of the revised (a) (i.e., little finger, middle, ring, and index finger). The FSM #3 was the same as above except the sequence for (a) changed to middle, little, index, and ring fingers and (c) changed to the reverse of above the revised (a) (i.e., ring, index, little, middle fingers). The above three sequences (instead only one sequence) were chosen to insure the continuous engagements of the subjects’ attention.

The above FSM were performed in two test conditions: following (Externally guided movements — EG) and continuation (Internally guided movements — IG). In the EG condition (30 s), subjects followed the finger tapping sequence shown on the screen (tapping frequency of 1 Hz). In the IG condition (30 s), the visual cue discontinued and the subjects were instructed to continue to keep tapping the same sequence as shown on the earlier screen (Figure 6.2). The two consecutive conditions were preceded and followed by a rest (R) period (30 s). The EG, IG, and R periods were designated using the visual
6.2. Experimental Methods

Figure 6.2: Activation paradigm for fMRI motor task. A block design paradigm was used wherein subjects held their right hand at rest, followed the sequence on the screen, or generated the previous sequence internally. Each block was 30 seconds in duration and the rate of finger tapping was 1 Hz.

cues, “FOLLOW”, “CONTINUE”, and “REST”, respectively. The visual cues remained on the screen with a pair of hands labeled as “Right” and “Left” to mirror the subjects’ hands throughout the fMRI session. The FOLLOW-CONTINUE-REST cycle was repeated four times during each run (total duration of 6 minutes). There were total of 4-6 runs performed on each subject (depending upon tolerability of each subject). Subjects practiced the task for about 20 minutes prior to scanning session, with more than 80% correction rate for the sequence. Subjects were not able to see their own hands at anytime during the scans. All subjects were monitored for performance accuracy throughout the study with the use of a video camera mounted on a tripod. These videos were assessed for accuracy and tabulated for each subject.
6.2. Experimental Methods

6.2.1 FMRI Image Pre-processing

The fMRI data was preprocessed for each individual independently for motion correction, smoothing, and time realignment using standard parametric mapping software (SPM 5). The time series of functional images were aligned for each slice in order to minimize the signal changes related to small motion of the subject during the acquisition. Temporal filtering of functional time series included removal of the linear drifts of the signal with respect to time from each voxel’s time-course and low-pass filtering of each voxel’s time-course with a one-dimensional Gaussian filter with FWHM = 6 s. The data were not spatially smoothed and were not spatially transformed to a common space.

Eight regions of interests (ROIs) were defined bilaterally (total = 16 ROIs) and manually traced for each subject based on anatomical sulcal landmarks and with the guidance of a brain atlas \[4\]: anterior cingulate cortex (ACC), supplementary motor areas (SMA), primary motor cortex (PMC), dorsal lateral prefrontal cortex (DLPFC), caudate (Caud), globus pallidus/putamen (GP/Put), thalamus, and lateral cerebellum hemisphere.

The proposed method is a post-processing method that utilizes statistical parametric maps. To ensure that any benefits from the proposed method were not due to the methods of obtaining the statistical parametric maps themselves, we utilized simple t-tests based on the BOLD signal changes in all runs between a task and rest (e.g., right hand IG vs. rest). The voxels in the statistical maps were then labeled by the appropriate ROIs drawn on the anatomical image. The labeled statistic parametric maps for different conditions (e.g. Right Hand IG vs. rest contrasted to Right EG vs rest) were then contrasted with LLDA.

6.2.2 LLDA Algorithm

The underlying idea of LLDA \[10\] is that global nonlinear data structures in many cases are locally linear and local structures can be linearly aligned. The LLDA linearly transforms each local structure (called a “cluster”) to a common vector space with an indi-
vidual transformation matrix and optimizes the discriminant between different classes globally in the common space.

Starting with the resampled feature matrix $X$ in Eq. (6.11) in a study involving $K$ subjects and two tasks, we regard the data of each subject as a local linear structure/cluster and attempt to find an individual transformation matrix for it such that the transformed data of all the subjects are globally and optimally discriminated between the tasks/classes. Let $x$, a $N \times 1$ vector be a column of $X$ and it belongs to a subject $k \in \{1, 2, ..., K\}$ and a task $c \in \{1, 2\}$. (Notations $x \in k$ and $x \in c$ will respectively mean that $x$ belongs to subject $k$ and task $c$). Next, $x$ is transformed to $y$ in the common vector space with Eq. (6.12)

$$y = U_k^T \left( x - m_k \right)$$  \hspace{1cm} (6.12)$$

where 

$$m_k = \frac{1}{N_k} \sum_{x \in k} x$$  \hspace{1cm} (6.13)$$

$$U_k = \left[ u_{k1}, ..., u_{kn}, ..., u_{kN} \right]$$ is the $N \times N$ orthogonal transformation matrix of cluster $k$ with $u_{kn}$ being its $n$th base, $m_k$ is the mean vector of cluster $k$, and $N_k$ is number of $x$’s which belong to cluster $k$. The mean of each cluster is removed in the transformation.

The discriminant after the transformation is scored with Eq. (6.14):

$$J = \log \left( \frac{\tilde{B}}{\tilde{W}} \right)$$  \hspace{1cm} (6.14)$$

where $\tilde{B}$ and $\tilde{W}$ are the between-class and within-class scatter matrices in the common
6.2. Experimental Methods

The transformed scatter matrices are defined as:

$$\tilde{B} = \sum_{c=1}^{C} N_c \cdot (\tilde{m}_c - \tilde{m}) (\tilde{m}_c - \tilde{m})^T$$

$$= \sum_{c=1}^{C} N_c \cdot \tilde{m}_c \cdot \tilde{m}_c^T$$

(6.15)

$$\tilde{W} = \sum_{c=1}^{C} \sum_{x \in c} (y - \tilde{m}_c) (y - \tilde{m}_c)^T$$

(6.16)

where $\tilde{m} = \frac{1}{N} \sum_y y = 0$ and $\tilde{m}_c = \frac{1}{N_c} \sum_{x \in c} y$ are the global mean and the mean of class $c$ respectively after the transformation, and $N_c$ is the number of $x$’s which belong to class $c$. Because the mean of a cluster is removed in the transformation, in our case $\tilde{m}$ equals 0. $\tilde{B}$ and $\tilde{W}$ can also be written in their matrix form as:

$$\tilde{B} = \left[ \begin{array}{ccc} U_1^T & \cdots & U_K^T \end{array} \right] \cdot \left[ \begin{array}{ccc} B_{11} & \cdots & B_{1K} \\ \vdots & \ddots & \vdots \\ B_{K1} & \cdots & B_{KK} \end{array} \right] \cdot \left[ \begin{array}{c} U_1 \\ \vdots \\ U_K \end{array} \right] = U^T B U$$

(6.17)

where $B_{ij} = \sum_{c=1}^{C} N_c m_{ci} m_{cj}^T$ and $m_{ck} = \frac{1}{N_c} \sum_{x \in c, x \in k} (x - m_{rk})$

$$\tilde{W} = \left[ \begin{array}{ccc} U_1^T & \cdots & U_K^T \end{array} \right] \cdot \left[ \begin{array}{ccc} W_{11} & \cdots & W_{1K} \\ \vdots & \ddots & \vdots \\ W_{K1} & \cdots & W_{KK} \end{array} \right] \cdot \left[ \begin{array}{c} U_1 \\ \vdots \\ U_K \end{array} \right] = U^T W U$$

(6.18)

where

$$W_{ij} = \begin{cases} -B_{ij}, & \text{if } i \neq j \\ \sum_{x \in k} (x - m_{rk})(x - m_{rk})^T - B_{kk}, & \text{if } i = j = k \end{cases}$$

(6.19)

LLDA attempts to maximize $J = \log \left( \frac{|U^T B U|}{|U^T W U|} \right)$ under the orthogonality and normal constraint $U_k U_k^T = U_k^T U_k = I$. The constrained nonlinear programming is solved by successively calculating the bases of $U_k$ from the subspace orthogonal to the already calculated bases. Unlike the original LLDA description by Kim and Kittler, we
propose using an overall optimization procedure using two routines, a “subspace” routine, and a “one-base” LLDA routine, which we found more robust and reliable for fMRI data. The “subspace” routine creates subspaces orthogonal to the already calculated bases $u_{k1} \cdots u_{k(n-1)}$ and calculates $u_{kn}$ in the subspace by calling the “one-base” routine which solves a one-base LLDA problem. The “subspace” routine repeats iteratively until all the bases are calculated. The “one-base” routine solves a one-base LLDA problem that is similar to LLDA but with a different constraint $U_k^T U_k = 1$ where $U_k$ is just a column vector and only subject to the normal constraint. The pseudo-code of the two routines is given in the appendix.

The procedure proposed here has several advantages over the original one proposed by Kim and Kittler for LLDA [11]. First, the orthogonalization is performed only once for every $u_{kn}$ in our procedure (at the “subspace” Routine’s step 4) because the orthogonality constraint is implemented in advance of solving the one-base LLDA problem by projecting $u_{kn}$ (at the “subspace” Routine steps 3 and 5) to the subspaces bases $A_k$ which is orthogonal to the already calculated $u_{k1} \cdots u_{k(n-1)}$. In contrast, the orthogonality constraint in the original LLDA implementation [11] is carried out in solving the one-base LLDA problem and thus the orthogonalization must be performed at every iteration, which is less computationally efficient. Second, in the one-base LLDA problem, we do not randomize the start point as Kim and Kittler did but estimate the initial starting point by solving a generalized eigenvector problem, which results in much faster and more stable convergence.

As pointed out by Kherif et al [9], averaging of fMRI data across individuals is only prudent when the mean is a good representation of the group. They suggested a way to look for selecting subjects with “similar” activation patterns. In the current situation, since the activation statistics from each individual have been transformed to a common vector space, defined by the $y$’s, we can now selectively weight each subject so that the weighted $y$s are maximally discriminable in the transformed vector space. Specifically,
6.2. Experimental Methods

we can weight each subject, \( k \), by a small positive factor \( \alpha_k \),

\[
\psi^* = \alpha_k y = \alpha_k U_k^T (x - m_k),
\]

subject to:

\[
\sum_{k=1}^{K} \alpha_k = 1, \quad \alpha_k \in [0, 1]
\]

where \( k \) is the subject and there are \( K \) total subjects and the means of the \( y^*\)s are maximally discriminable by, e.g., a standard t-test. The \( \alpha_k \)'s can then be estimated by constrained non-linear optimization methods.

To estimate the overall transformation, i.e., the linear combination of ROIs that maximally discriminate between groups, we take advantage of the individual subject \( \alpha_k \)'s computed with (6.20) and summarize all the individual \( U_k \)s by:

\[
\bar{U} = \sum_{k=1}^{K} \alpha_k U_k
\]

In the appendix, we explicitly demonstrate how \( \bar{U} \), at least for the two-class problem, can be jointly estimated during the LLDA optimization. This joint optimization procedure gaver results virtually identical to that given by (6.22).

6.2.3 Backward Step-wise Discrimination

To determine which ROIs should actually be included in the discriminant analysis, a backward step-wise procedure can be employed. This method first assumes that all ROIs significantly contribute to the discrimination. The least insignificant element of \( \bar{U} \) is then removed from the analysis, and the procedure is repeated. The fewest number of overall included ROIs that result in the maximum number of significant ROIs is then retained.
6.2.4 Contrasting Multiple Tasks

In fMRI experiments there is frequently interest in contrasting multiple tasks. For example, given the model:

\[ Y_k = X_k\beta_k + \varepsilon_k \]

the contrast of choice for a single task and rest is: [1, -1, 0, 0, ...], where the first two columns of the design matrix represent the anticipated activation and rest respectively. In the current case, we are interested in contrasting activations resulting from right or left hand movement, independent of whether or not the task was externally-guided (EG) or internally-guided (IG), resulting in a contrast citation [1/2, 1/2, -1, 0, 0, ...] if the design matrix represented [EG, IG, rest, ...]. However, for the LLDA implementation we can proceed two ways:

1. we can pool the t-statistics from the EG and IG tasks — in effect placing the same ROI mask over both statistical maps from the EG vs. rest and IG vs. rest to create an overall group of voxel-based statistics representing that ROI. We note that this is equivalent to creating a contrast such as [1/2, 1/2, -1, 0, 0, ...]

2. we can treat each task as a separate cluster and calculate the \( U_k \) for each task of each subject.

Although standard approaches employ method (6.1), we have had much better empirical results with method (6.2). While it is difficult to analytically determine why this would be so, a possible and intuitive explanation is the “Simpson paradox” [21]. The paradox shows that the conditional association between two factors \( A \) and \( B \) given another factor \( C \) may be completely altered compared to their marginal association, without knowing \( C \). In our situation, right (or left) hand movement and ROI’s activities are the two factors \( A \) and \( B \), while the condition EG or IG is the third factor. Method (6.1) pools
6.2. Experimental Methods

the EG and IG data together and thus studies the marginal association, possibly missing meaningful interactions. Method (6.2) considers EG and IG separately and thus studies the conditional association.

6.2.5 Significance of Discrimination

A necessary but not sufficient condition of assessing the role of individual ROIs in the overall transformation, $\bar{U}$, results in separation that is statistically significant. Specifically, we determine the separation on a subject by subject basis by evaluating the differences with t-tests [20]:

$$t_i \sim \frac{\bar{y}_{A,i} - \bar{y}_{B,i}}{\sqrt{\frac{\sigma^2_{y_{A,i}}}{n_{y_{A,i}}} + \frac{\sigma^2_{y_{B,i}}}{n_{y_{B,i}}}}}$$  \hspace{1cm} (6.23)

where $y$ is the transformed data using the mean transformation, $\bar{U}$, $A$ and $B$ refer to different activation maps being compared, and $i$ is the subject index. We can test the null hypothesis that the different $t_i$'s from all subjects are not significantly different from 0.

In order to determine whether or not the overall discriminant, $\bar{U}$ (Eq. (6.22)) resulted in false positives, ROC curves were drawn to compare the distributions of $\bar{U}$ under two hypotheses:

H0: None of the ROIs are differently activated when the subjects perform the two tasks.

H1: Some of ROIs are differently activated when the subjects perform the two tasks.

The distribution of $\bar{U}$ are estimated from data with two resampling techniques, bootstrapping and permutation. Bootstrapping resamples S subjects from the real data with replacement and simulates the random recruitment of subjects from a population. If the data of the same subject is sampled more than once, they are considered as different subjects, for the purposes of LLDA, in the resampled data. Permutation randomly shuffles
6.2. Experimental Methods

the task labels of the data matrices of each subject.

The distribution of $\bar{U}$ under H0 is estimated with two steps: 1) generate many new data sets with bootstrapping followed by permutation. 2) apply LLDA to the new data sets and calculate a $\bar{U}$ for each data set. We performed 2000 bootstraps to estimate $\bar{U}$ under H0. The distribution of $\bar{U}$ under H0 was estimated similarly, but only bootstrapping was employed in the resampling. With the estimated distribution of $\bar{U}$ under H0 and H1, ROC curves were drawn.

6.2.6 SPM Methodology

We also performed SPM 5 analysis on our data. The FMRI data were pre-processed (motion corrected, spatially normalized to a common stereotactic space and smoothed with a Gaussian kernel with 6-mm full width at half maximum (FWHM)). The first level comparisons were made between right hand and left hand finger sequential movements, and the individual activation map (right > left; and left > right) for each subject was first generated by a fixed-effect model with the voxels that exceeded a probability threshold of $p = 0.05$ FDR (False Discovery Rate – corrected). In order to provide a reasonable comparison, the first level analysis was restricted to those voxels within the ROIs, using the PickAtlas software package [13]. The second level analysis was made based on the results of the individual activation maps generated in the first level comparison. That is, the contrast images, one from each of 10 subjects from the first level comparison (for example, right>left) were assessed using one sample t-test by a random-effect model. This can be expressed in the equation, $C\hat{\beta} = Z = X_g\beta_g + \varepsilon_g$, where $C\hat{\beta}$ is the contrast result from first level analysis, $X_g$ is a design matrix that is simply a single column of 1’s, $\beta_g$ is a second level parameter, and $\varepsilon_g \sim N(0, \sigma_g^2)$. The regions that have clusters with at least 5 contiguous voxels exceeding a probability threshold of $p = 0.001$ (uncorrected) were identified as activated regions.
6.3 Results

Using SPM 5, we found activation in the left primary motor cortex during right hand movement, and similarly left primary motor cortex activation during right hand movement. Activation in the left cerebellar hemisphere, right somatosensory cortex, and right thalamus was detected using left hand movement only (Figure 6.3a).

Similarly, during right hand movement, significant activation was detected with LLDA in the left primary cortex. With left hand movement LLDA detected significant activity in the right primary motor cortex, the right supplementary motor area, the right striatum, the right thalamus and the left cerebellar hemisphere (Figure 6.3b). The ROCs demonstrated large deviations from the diagonal, suggesting that the null hypothesis of a region not being active and the hypotheses that it is active have little overlap and a threshold can easily be set to classify them well. The ROC curves also demonstrated superior performance of LLDA (solid line) over LDA (dotted line).

In comparing the SPM and LLDA results contrasting EG vs. IG there were also distinct differences (Figure 6.4). SPM analysis did not identify any clusters as having significant activity for either the EG>IG comparison or the IG>EG comparison. In contrast, for the EG vs. IG comparison LLDA detected significant activity in the left primary motor cortex and the left SMA and when comparing IG>EG tasks, a structure previously found to display increased activity during IG tasks [22]. When comparing EG>IG, significant activity was found in the dorsolateral prefrontal cortex, bilaterally consistent with previous results [18].

The weighting across subjects for the joint optimization was relatively consistent (Figure 6.5). In all cases the separation across each subject (cf. Eq. (6.23)) was significantly different from zero (Figure 6.5).
Figure 6.3: Comparison of group results. Note that this is a combination of EG and IG movements (top panel) SPM results. (bottom panel) LLDA results. ROC curves are shown for significantly active regions. A comparison between LLDA (solid line) and regular LDA (dotted line) is shown.
6.3. Results

Figure 6.4: Comparison of group results, examining movement type (EG vs IG). (top panel) SPM results. (bottom panel) LLDA results.
Figure 6.5: Discriminability of overall discriminant function, $\bar{U}$, across subjects.
6.4 Discussion

We have utilized a modification of the local linear discriminant (LLDA) algorithm to perform group-wise analysis on activation maps that have been manually labelled with individually-drawn Regions of Interest (ROIs). In addition to finding the contralateral primary motor cortical activation and ipsilateral cerebellar activation similar to the SPM approach, we found a number of regions that were significantly active, including bilateral dorsolateral prefrontal cortex during externally guided movement. We also found differences between the use of the right and left hand, such that the right SMA was activated only during left hand performance. This result most likely reflects differences due to hand dominance.

In contrast to linear discriminant analysis (LDA), which treats each subject independently, LLDA tried to simultaneously discriminate combinations of ROIs differing between tasks within individuals, while still trying to determine a common transformation across individuals (Figure 6.1). We are particularly interested in the vector space of ROI-based activation statistics that is invariant across individuals, but differs by task activation. A permutation/bootstrapping approach has been employed to prevent false positives.

A useful feature of LLDA is that it transforms the activation statistics from each subject (or each task of each subject) into a common vector space with respect to relative activation of ROIs during task performance. This allows us to meaningfully compare how similar the combined activation, specified by $\bar{U}$, is across subjects (Figure 6.5) providing a check for subjects whose activations represent outliers.

The proposed method may be computationally expensive. To run 10 subjects with 16 ROIs, using full backward step-wise discrimination, with LLDA at each step, took approx 30 minutes total on a P4 PC. As this was implemented in Matlab for programming convenience, it is expected that compilation could improve the computational performance.

By repetitively resampling voxel-based statistics to ensure a multivariate Gaussian
distribution of the feature vectors (Eq. (6.9)) there is a risk that our bootstrap samples may no longer be i.i.d. Often researchers restrict the ROIs to the intersection between anatomically derived ROIs and voxels deemed "active" for a given task. However, by restricting subsequent analyses to these active voxels, the distribution of activation statistics will necessarily be highly non-Gaussian. A balance must therefore be obtained between making the data more Gaussian and the risk of using non-i.i.d. bootstrap samples. Since recent work has suggested that the i.i.d. assumption can be violated without disrupting the results excessively (e.g., [19]), we suggest the current approach is reasonable.

Since we are interested in subcortical structures, some consideration must be given to ROIs that have few voxels. We suggest that this will result in discriminant functions that are overly conservative. For example, consider just two ROIs, $ROI_{small}$ with relatively few voxels and $ROI_{large}$ with many voxels. The effective oversampling of $ROI_{small}$ to create a $2 \times n$ bootstrap matrix ($n$ being the number of bootstrap samples, cf. Eq. (6.11)) will result in conservative estimates of the discriminant function, since the heavily oversampled $ROI_{small}$ voxels will be distributed over the entire range of voxels in $ROI_{large}$.

There is growing recognition that warping of individual subjects’ brains to a common space may cause particular problems with registration, especially with small subcortical structures [17, 24]. Despite widespread evidence that subcortical structures such as the thalamus are an integral part of the network used for motor control, only the LLDA approach, when applied to unwarped data, was able to detect significant activation differences between right and left handed task performance. We suggest that experimenters employing tasks expected to activate subcortical structures should use caution when warping individual subjects’ brains to a common space.
6.5 Technical Appendix

“Subspace” routine: incrementally calculate $u_{kn}$

(1) Initialize: $n = 1$, $A_k = I_{N \times N}$.

$A_k$, a $N \times (N + 1 - n)$ matrix is the subspace which $u_{kn}$ is projected to.

(2) Call the “One-base” Routine to calculate $u'_{kn}$.

$u'_{kn}$, a column vector of $N + 1 - n$ elements, is the projection of $u_{kn}$ in $A_k$.

(3) Convert $u'_{kn}$ to $u_{kn}$: $u_{kn} = A_k u'_{kn}$.

$u_{kn}$ is a column vector of $N$ elements.

(4) Create a subspace $S_k$ which is orthogonal to $u'_{kn}$.

$S_k$ is a $(N + 1 - n) \times (N - n)$ matrix.

(5) Prepare for the next iteration:

$$A_k = A_k S_k,$$

$$S = \text{diag}(S_1 \cdots S_K),$$

$$B = S^T B S,$$

$$W = S^T W S,$$

$$n = n + 1.$$

(6) If $n > N$, stop; else go back to step 2.

“One-base” routine: solve the one-base LLDA problem

The non-linear programming with the quadratic constraint is solved with a sequential quadratic programming (SQP) method implemented in Matlab function $fmincon$. The start point $U_{k}^{\text{init}}$ is initialized by solving a generalized eigenvector problem as shown below.
Joint optimization of mean U for the two-class problem

The ultimate goal of the optimization is to maximize the discriminant function of \( y^{\text{mean}} \), i.e., the weighted y. As \( y^{\text{mean}} = \alpha_k y = \alpha_k U_k^T (x - m_k) \), the discriminant function of \( y^{\text{mean}} \) is

\[
J = \log \frac{|B^*|}{|W^*|},
\]

where

\[
B^* = \begin{bmatrix}
\alpha_1 U_1^T & \cdots & \alpha_K U_K^T
\end{bmatrix}^* \begin{bmatrix}
B_{11} & \cdots & B_{1K} \\
\vdots & \ddots & \vdots \\
B_{K1} & \cdots & B_{KK}
\end{bmatrix}^* \begin{bmatrix}
\alpha_1 U_1 \\
\vdots \\
\alpha_K U_K
\end{bmatrix} = U^T B U^*
\]

\[
W^* = \begin{bmatrix}
\alpha_1 U_1^T & \cdots & \alpha_K U_K^T
\end{bmatrix}^* \begin{bmatrix}
W_{11} & \cdots & W_{1K} \\
\vdots & \ddots & \vdots \\
W_{K1} & \cdots & W_{KK}
\end{bmatrix}^* \begin{bmatrix}
\alpha_1 U_1 \\
\vdots \\
\alpha_K U_K
\end{bmatrix} = U^T B U^*
\]

In the case of a two-class problems, \( U^* \) is a column vector composed of several column vectors \( \alpha_i U_i^T \). Though \( U_i \) and \( \alpha_i \) are subject to the constraints \( |U_i| = 1, \sum \alpha_i = 1 \) and \( \alpha_i \geq 0 \), the optimization can be solved without the constraints by the following. First, we solve

\[
\max_{U^*} J = \log \frac{|B^*|}{|W^*|} = \log \frac{|U^{T} B U^*|}{|U^{T} W U^*|}
\]
6.5. Technical Appendix

\[ U^* = \begin{bmatrix} U_1^* \\ \vdots \\ U_K^* \end{bmatrix} \]

without the constraints by calculating the generalized eigenvectors of \( B \) and \( W \). Then, we normalize each \( U_i^* \) to \( U_i \):

\[
\begin{align*}
    b_i &= |U_i^*| \\
    U_i &= U_i^*/b_i
\end{align*}
\]

Finally, we derive the weight \( \alpha_i \) from \( b_i \):

\[
\alpha_i = \frac{b_i}{\sum b_i}
\]

Because \( U_i \) and \( \alpha_i \) solved in this way satisfy the constraints and they optimize the unconstrained problem whose solution is not worse than that of the constrained problem, they are also the solution of the constrained problem.

Acknowledgments

This work was supported by a National Parkinson’s Foundation Center of Excellence grant (MJM), NSERC Grant CHRPJ 323602-06 (MJM), CIHR grant CPN-80080 (MJM) and NIH grants AG21491 (XH), and RR00046 (GCRC).
Bibliography


Chapter 7

Conclusions and Discussions

The previous chapters have, at length, discussed several approaches of applying dynamic Bayesian networks to analyzing neural signals. This chapter completes the study by reviewing them in light of current research in the field. The topics, methods and results of the conducted research are summarized in Section 7.1. The contributions and potential applications of the technology developed in the research are highlighted in Section 7.2. The limitations and possible future work are discussed in Section 7.3.

7.1 Summary

Graphical models are a suitable prototype for modeling the interactions among neurons. It is very natural to abstract the nervous system mathematically as a network whose neurons (or nodes) connect with each other through nerve fibres (or edges). Bayesian networks have been actively studied for decades in the field of artificial intelligence, but to meet the particular need of biomedical research and the limitations of current neural-signal acquisition technology, classical methods still need to be adapted and novel methods still need to be developed.

Since biomedical research usually has high requirements of interpretability, generality and reliability, we mainly investigated the following topics: structural-feature extraction, group analysis, and error control. To provide guidance on further research exploration and clinical diagnosis, mathematical models in biomedical research should not only fit experimental data well, but also be able to extract interpretable features from data. To make the discoveries generally applicable to a population, rather than just a specific
individual, experimental data should be studied at group level, with both population homogeneity and heterogeneity being considered. To produce reliable analysis results, the rate of errors must be curbed at a low level or balanced against the chance of missing true evidence.

**Structural-Feature Extraction**

Our research on extracting informative features from Bayesian networks was described in Chapter 2. We developed a structural-feature extraction framework for Bayesian networks that is able to discover structural features that are significantly different between two groups of Bayesian networks, such as the pattern of sub-networks. The framework includes three components: Bayesian-network modeling, statistical structure-comparison, and structure-based classification. It was demonstrated with a study on the coordination patterns of muscle activities using surface electromyography.

First, Bayesian networks were applied to modelling the sEMG signals of multiple muscles, with directed acyclic graphs encoding the interactions between muscle activities. Second, statistical comparison was conducted between the learned Bayesian networks, to extract structural features which characterize the interaction patterns of muscle activities. Structural features can be the number of edges in or out from a node (which is called degree in graph theory), or the length of the shortest path from one node to another (which is called distance in graph theory). To go beyond the level of individual nodes and individual edges, we worked on subnetworks, adapting the concept of network motifs. Third, interpretable classifiers, such as classification trees, were built on those structural features. We used classification trees because their classification processes can be easily visualized. In our study on thirteen stroke patients and nine healthy subjects, the classification trees could effectively (with error rates lower than 6%) classify the reaching movements performed by the healthy subjects and the stroke patients.
7.1. Summary

Comparison of Group-Analysis Methods

Our research on group analysis based on dynamic Bayesian networks was presented in Chapter 3. Group analyses of any sort fundamentally involve two factors: common features shared by group members, and specific features of each individual. A review of the literature reveals that group-analysis methods for graphical models can be divided into three broad categories, namely the “virtual-typical-subject” approach [5, 16, 23], the “individual-structure” approach [5, 11] and the “common-structure” approach [8, 13]. The three approaches handle the similarity and diversity of a group of graphical models, e.g. Bayesian networks, at different structural and parametric levels. We investigated the performances of the three group-analysis approaches in an fMRI study on Parkinson’s disease with L-dopa medication. Broadly speaking, we attempted to answer three fundamental questions: “which approach most accurately reflects the underlying biomedical behavior?”, “do the approaches lead to considerably different analysis results?”, and “how can the suitable approach be selected?” We compared the three approaches from the aspects of their statistical goodness-of-fit to the data, and more importantly their sensitivity in detecting the effect of the medication on the disease.

The three approaches led to considerably different group-level results, learning different network structures, and detecting different numbers of connections normalized by the medication, from the same data set. The “virtual-typical-subject” approach fitted the data of the healthy people best, while the “individual-structure” approach fitted the data of the patients best, from the view of the Bayesian-information-criterion (BIC). The “individual-structure” approach was more sensitive than the other two approaches in detecting the normalizing effect of the medication on brain connectivity, finding sixty-four out of sixty-six connections between brain regions being normalized by the medication.
Error-Rate Control in Structure Learning

Our research on error control was elaborated in Chapter 4 and then further developed in Chapter 5. We designed an algorithm for static Bayesian networks that is able to control the false discovery rate (FDR) \[1\] of the network connections inferred from experimental data under user-specified levels, for example, conventionally 5%. The FDR is the expected ratio of falsely claimed positive hypotheses to all those claimed. In the context of learning network connections, the FDR is the expected ratio of falsely “discovered” connections to all those “discovered”. In other words, not rigorously, the FDR can tell how many among the “discovered” connections are actually spurious. The algorithm is based on the popular PC algorithm invented by Peter Spirtes and Clark Glymour \[19\] which is fast but does not control the effect of multiple testing. We built an FDR-control procedure \[1\] into the PC algorithm to correct the effect of simultaneously testing the existence of multiple edges. We named the FDR-embedded PC algorithm the \(PC_{fdr}\) algorithm, and a heuristic modification the \(PC_{fdr}^*\) algorithm.

Theoretically, we proved that under mild conditions the \(PC_{fdr}\) algorithm is able to curb the FDR under user-specified levels at the limit of large sample size, and that both the \(PC_{fdr}\) algorithm and the \(PC_{fdr}^*\) algorithm are able to recover all the true connections with probability one as the sample size approaches infinity.

Empirically, we tested the two new algorithms on simulated data of moderate sample sizes, such as several hundred. In the test, the \(PC_{fdr}\) algorithm controlled the FDR under user-specified levels, and the \(PC_{fdr}^*\) algorithm controlled the FDR accurately around the user-specified levels.

The original version of the \(PC_{fdr}\) and \(PC_{fdr}^*\) algorithms was designed only for static Bayesian networks, taking only the classical approach. Two advanced versions were then developed. One is an adaptation to prior knowledge, allowing users to specify which edges must appear in the network, which cannot, and which are to be learned from data. This extension is naturally applicable to dynamic Bayesian networks, by simply regarding
them as Bayesian networks that cannot have edges from time $t + 1$ to time $t$. The other extension is using the $\text{PC}_{\text{fdr}}$ algorithms to improve Bayesian inference of (dynamic) Bayesian networks. The idea is to first learn a network with the $\text{PC}_{\text{fdr}}$ algorithms, and then make Bayesian inference based on a prior distribution derived from the learned network. It accelerates Bayesian inference and is relatively robust to perturbing noise.

These new FDR-controlled structure-learning algorithms were applied to studying the functional connectivity among brain regions using functional magnetic resonance imaging (fMRI). Evidence was found to support the normalizing effect of medication on Parkinson’s disease, and the compensation mechanism of the disease.

Selection of Regions of Interest

Chapter 6, as a supplement to Chapters 2 to 5, provides a fast method to select brain regions of interest for further modeling their connectivity with Bayesian networks.

7.2 Contributions and Potential Applications

The major contributions of this research to dynamic Bayesian-network modeling of neural signals are: algorithms that are able to learn reliable network structures from experimental data, a framework for extracting interpretable structural features, and guidance on selecting group-analysis strategies.

Structure-Learning Algorithms with Error Rates Controlled

The $\text{PC}_{\text{fdr}}$ and the $\text{PC}_{\text{fdr}*}$ algorithms, as the reviewers of the Journal of Machine Learning Research commented, “have a significant contribution in the field of network learning, in that it introduces an adequate method of handling errors with finite data and theoretical justification.” The algorithms provide explicit control over the false discovery rate of network structures learned from experimental data. They are modular, flexible
and computationally efficient. Their asymptotic performances have been theoretically proved, and in the case of moderate sample size they have been extensively evaluated with simulated data.

Current structure-learning algorithms for Bayesian networks have not been adequately adapted to explicitly control the FDR of the claimed “discovered” networks. Score-based search methods [6] look for a suitable network structure by optimizing a certain criterion of goodness-of-fit, such as the Akaike information criterion (AIC), the Bayesian information criterion (BIC), or the Bayesian Dirichlet likelihood equivalent metric (BDE), but scores of goodness-of-fit cannot not directly reflect the error rate. The Bayesian approach, theoretically, can estimate the exact posterior probability of any structure features, but in practice its capacity is largely limited for computational reasons, and only feasible for small scale problems, or under certain additional constraints. For certain special prior distributions, exact-inference algorithms [9] are available, and are efficient for problems with less than thirty nodes. However, they are not applicable to the widely accepted uninformative prior, i.e. the uniform prior distribution over DAGs [3,4]. Markov chain Monte Carlo (MCMC) [12], a generic implementation of Bayesian inference, usually requires intensive computation and the results may depend on the initial state of the randomization.

The $PC_{fdr}$ and $PC_{fdr^*}$ algorithms provide direct control over the FDR of the networks learned from experimental data, reporting reliable analysis results. We proved that under mild conditions the $PC_{fdr}$ algorithm is able to curb the FDR under user-specified levels at the limit of large sample size, and that both the $PC_{fdr}$ algorithm and the $PC_{fdr^*}$ algorithm are able to recover all the true connections with probability one as the sample size approaches infinity. In the cases of moderate sample size (about several hundred), simulation experiments have showed that the method is still able to control the FDR under the user-specified level, and its heuristic modification, the $PC_{fdr^*}$ algorithm is able to control the FDR accurately around the user-specified level.
The simulation study has also shown that the extra computational cost to achieve the FDR control is negligible when compared with that already spent by the PC algorithm on statistical tests of conditional independence. The computational complexity of the new algorithm is closely comparable with that of the PC algorithm.

The \texttt{PCfdr} and the \texttt{PCfdr*} algorithms are modular, consisting of the PC search strategy, statistical tests of conditional independence and FDR-control procedures. Different statistical tests and FDR-control procedures can be “plugged in” depending on the type of data and the statistical model. Thus, the method is applicable to any models for which statistical tests of conditional independence are available, such as discrete models and Gaussian models. A previous method \cite{17} is only applicable to graphical Gaussian models.

The \texttt{PCfdr} and the \texttt{PCfdr*} algorithms are flexible. Their extended versions allow users to specify which edges must appear in the network, which cannot, and which are to be learned from data. The extended versions are not only applicable to static Bayesian networks, but also to dynamic Bayesian networks. The \texttt{PCfdr} algorithms can also be used to accelerate Bayesian inference on dynamic Bayesian networks.

The algorithms of the \texttt{PCfdr} family are widely applicable to research where network-learning is involved and the uncertainty of the learned networks should be controlled or assessed, for example, discovering gene regulatory networks, or cellular metabolic networks. We have successfully applied the algorithms to discovering connectivity between brain regions using functional magnetic resonance imaging (fMRI). The discoveries were consistent with the normalizing effect of medication on Parkinson’s disease.

**Effective Framework for Structural-Feature Extraction**

The framework based on Bayesian networks for structural-feature extraction provides an effective and flexible method for discovering interaction patterns among neural activities, and its application to sEMG signals casts fundamentally new understanding on the nature
of sEMG signals.

The framework is effective and flexible in discovering interaction patterns among neural activities. As demonstrated in a study on stroke using sEMG, it discovered several groups of muscles whose coordinated activities in reaching movement significantly differs between healthy people and stroke patients, and in classification it also achieved error rates lower than 6%. The analysis framework, consisting of Bayesian-network modeling, statistical structure-comparison, and structure-based classification, is also flexible to different research specifications. Different types of Bayesian networks can be chosen according to prior knowledge and specific research scenarios. For example, dynamic Bayesian networks can be used to model temporal interactions, and Bayesian networks with hidden nodes to model unobservable neural signals driving muscles. Various structural features can be extracted with the analysis framework, for instance, the number of edges in or out from a node, or the length of the shortest path from one node to another, or the pattern of sub-networks.

Its application to modelling sEMG signals casts fundamentally new understanding on the nature of sEMG signals. Traditionally, sEMG signals are decomposed into two components: the amplitude which is extracted for further analysis, and the modulated carrier which is ignored as random noises [2]. However, as we still lack comprehensive understanding of the nature of the carrier signals, the common practice of disregarding them is questionable. With the framework for structural-feature extraction, the carrier signals could classify the reaching movements of stroke patients and healthy people with very low error rates, between 0.00% and 5.56%. The classification error-rates based on the amplitude with traditionally standard methods were considerably higher, between 7.69% and 16.67%. This implies that the carrier signals are not purely random, but also informative.
Guidance on Selection of Group-Analysis Methods

The comparison on group-analysis methods, to the best of our knowledge, was the first study specifically devoted to group analysis with graphical models, in real biomedical applications. It reveals the limitations of current methods, showing directions of future development. It also provides guidance on the selection of group-analysis methods in current research practice.

The three popular group-analysis methods, that is, the virtual-typical-subject approach, the common-structure approach, and the individual-structure approach, yielded considerably different group-level results, and tended to find results supporting their own assumptions even when the assumptions were actually wrong. This clearly revealed their limitations: none of them provides a tunable balance between group-common features and subject-specific features, or a means to estimate the degree of inter-subject variability, so that they are confined to their rigid assumptions. While the author was finalizing the thesis, Stephan, Penny, etc. in a recently accepted paper [21] proposed a group Bayesian model to address these limitations.

The comparison also suggests that biomedical evidence is an effective criterion for selecting group-analysis models. The sensitivity to the normalizing effect of medication on Parkinson’s disease was used as the criterion for selecting group-analysis methods, and it effectively differentiated the individual-structure approach in favor from the other two candidate approaches. This suggests that whenever possible researchers should design biomedical markers, and use them to select group-analysis strategies.

7.3 Future Work

Structure-learning and group analysis are important for applying dynamic Bayesian networks to neural signals, and much work is left to be done. Some of the methods we developed herein are promising prototypes but still need to be improved to better match
the challenges of real-world problems. Some are successful inventions, both theoretically and empirically, and they can be extended to fully exploit their potential. Future research can focus on enhancing their ability to handle large scale problems, and incorporating with methods in other broad research fields.

**High-Capacity Structure-Learning Algorithms**

Current network-learning methods for graphical models can only reliably recover networks of about twenty nodes from data of a practical sample size, providing results at very low spatial resolution. Random graph theory has revealed that many real-world networks, such as the cellular metabolic networks, actually possess surprising scale-free properties, such as the small-world property, or self-similarity \[18\]. For neural-interaction networks, such as brain functional connectivity, there are also sparse reports and conjectures on their structural properties, from the perspective of random graph theory. Since this kind of prior knowledge has not been exploited in current network-learning, it is interesting to embed these scalable properties into structure-learning methods to enhance their ability to handle networks at high spatial resolution. This development will not only potentially benefit the study on neural-interaction networks, but also many other studies that involve discovering networks from real-world data.

**Extensions of PC_{fdr} Algorithms**

The scheme used in the PC_{fdr} algorithms for controlling the false discovery rate (FDR) of network structures can be extended from many aspects to better meet the need of biomedical studies. It has been theoretically proved and empirically validated that the PC_{fdr} algorithms are able to robustly control the false discovery rate of networks learned from experimental data. Nevertheless, the algorithms are initial inventions, and the potential of the fundamental scheme used in them is far from being fully exploited.

First, it can be embedded into algorithms that are similar to but more advanced than
the PC algorithm. Besides the PC algorithm, many other algorithms, for instance, the Inductive Causation (IC) algorithm [15], the IC* algorithm [15], and the Fast-Causal-Inference (FCI) algorithm [19], are also rooted in the Markov properties [10] and based on the test of conditional-independence. Because these algorithms share the same theoretical foundation, the scheme we developed for the PC algorithm can be “transplanted” into them. The FCI algorithm and the IC* algorithm can solve problems with latent variables, so it is especially interesting to adapt the FDR-control scheme for them.

Second, the PC\textsubscript{fdr} algorithms can be extended for group analysis to control the FDR at group level. Current PC\textsubscript{fdr} algorithms are designed for data sampled from a single Bayesian network. With modification, they can be extended for data sampled from a group of Bayesian networks. A key inference procedure in the PC\textsubscript{fdr} algorithms is testing whether two random variables \(a\) and \(b\) are conditionally independent given a set of other random variables \(C\). If we extend the test of conditional independence to group level, \textit{i.e.} testing whether \(a\) and \(b\) are conditionally independent at the group level, then the FDR control can be applied to group analysis.

The asymptotic performance of the PC\textsubscript{fdr} algorithm has only been proved under the assumption that the number of vertices is fixed. Its behavior when both the number of vertices and the sample size approach infinity has not been studied yet. Kalisch and Bühlmann proved that for Gaussian Bayesian networks, the PC algorithm consistently recovers the equivalence class of an underlying sparse DAG, as the sample size \(m\) approaches infinity, even if the number of vertices \(N\) grows as quickly as \(O(m^{\lambda})\) for any \(0 < \lambda < \infty\) [7]. Their idea is to adaptively decrease the type I error rate \(\alpha\) of the PC algorithm as both the number of vertices and the sample size increase. It is desirable to study whether similar behavior can be achieved with the PC\textsubscript{fdr} algorithm if the FDR level \(q\) is adjusted appropriately as the sample size increases.
7.3. Future Work

Modular, Updatable and Scalable Group-Analysis Methods

Current group-analysis methods for Bayesian networks cannot effectively balance group-common features and subject-specific features, or meet the need of various experiment designs. It is necessary to develop more advanced group-analysis methods in brand new frameworks. As we discussed in Section 7.2, the three popular group-analysis approaches, i.e. the virtual-typical-subject approach, the common-structure approach, and the individual-structure approach, tend to found results supporting their own assumptions even when the assumptions are actually wrong. One of the underlying reasons is that none of the three methods provides a tunable balance between group-common features and subject-specific features, or a means to estimate the degree of cross-subject variability. They build an unnecessarily rigid boundary between the two different yet related concepts, structures and parameters. This sheds light on the development of new group-analysis methods: to transcend the boundary. In a recently accepted paper [21], Stephan, Penny, etc. have proposed a Bayesian group model to smooth the boundary between structures and parameters.

To set clear goals for the development of new group-analysis methods, we suggest three highly desirable features: being modular, being incrementally updatable, and being scalable. Being modular means that a group-analysis method is not only designed for a particular type of single-subject model, but versatile and applicable to different types of single-subject models. For example, both Bayesian networks and structural equation models are applicable to fMRI signals at the single-subject level, so the group-analysis method should not be restricted to only one of them, but should be able to work with both of them. If a group-analysis model can be a module of itself, then it will be able to handle hierarchical group structures.

Being incrementally updatable means that group-inference results can be summarized as summary statistics, and forwarded to analysis involving newly collected data. This feature is very useful in research practice because experimental data are usually collected
7.3. Future Work

incrementally. For example, after a study on eighty subjects half a year ago, twenty more subjects might be recruited. In this case, if the group analysis must start over, but not incrementally update, it may need cumbersome computation to analyze the data of the one hundred subjects as a whole. On the other hand, if the group inference is incrementally updatable, it may need much less computation to involve the twenty additional subjects.

Being scalable means that a group-analysis method can handle fast growing diversity among subjects. Because modern exploring research usually involves investigation on a large number of candidate models, scalability has become a highly desirable feature for group analysis. For example, if we assume that connectivity among brain regions is different from one subject to another, and the connectivity among ten brain regions is investigated, then a group-analysis method should be able to handle the diversity of about $3.1 \times 10^{17}$ different directed acyclic graphical models of brain connectivity.
Bibliography


