5-2015

Statistical Dependence in Imputed High-Dimensional Data for a Colorectal Cancer Study

Anvar Suyundikov
Utah State University

Follow this and additional works at: https://digitalcommons.usu.edu/etd

Part of the Mathematics Commons

Recommended Citation
Suyundikov, Anvar, "Statistical Dependence in Imputed High-Dimensional Data for a Colorectal Cancer Study" (2015). All Graduate Theses and Dissertations. 4371.
https://digitalcommons.usu.edu/etd/4371

This Dissertation is brought to you for free and open access by the Graduate Studies at DigitalCommons@USU. It has been accepted for inclusion in All Graduate Theses and Dissertations by an authorized administrator of DigitalCommons@USU. For more information, please contact rebecca.nelson@usu.edu.
STATISTICAL DEPENDENCE IN IMPUTED HIGH-DIMENSIONAL DATA FOR
A COLORECTAL CANCER STUDY

by

Anvar Suyundikov

A dissertation submitted in partial fulfillment
of the requirements for the degree
of
DOCTOR OF PHILOSOPHY
in
Mathematical Sciences
(Statistics)

Approved:

Dr. John R. Stevens
Major Professor

Dr. Chris Corcoran
Committee Member

Dr. Daniel Coster
Committee Member

Dr. Adele Cutler
Committee Member

Dr. Abby D. Benninghoff
Committee Member

Dr. Mark R. McLellan
Vice President for Research and
Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2015
ABSTRACT

Statistical Dependence in Imputed High-dimensional Data for a Colorectal Cancer Study

by

Anvar Suyundikov, Doctor of Philosophy
Utah State University, 2015

Major Professor: Dr. John R. Stevens
Department: Mathematics and Statistics

The main purpose of this dissertation was to examine the statistical dependence of imputed microRNA (miRNA) data in a colorectal cancer study. The dissertation addressed three related statistical issues that were raised by this study. The first statistical issue was motivated by the fact that miRNA expression was measured in paired tumor-normal samples of hundreds of patients, but data for many normal samples were missing due to lack of tissue availability. We compared the precision and power performance of several imputation methods, and drew attention to the statistical dependence induced by K-Nearest Neighbors (KNN) imputation. The second statistical issue was raised by the necessity to address the bimodality of distributions of miRNA data along with the imputation-induced dependency among subjects. We proposed and compared the performance of three nonparametric methods to identify the differentially expressed miRNAs in the paired tumor-normal data while accounting for the imputation-induced dependence. The third statistical issue was related to the development of a normalization method for miRNA data that would reduce not only technical variation but also the variation caused by the characteristics of subjects, while maintaining the true biological differences between arrays.

(132 pages)
PUBLIC ABSTRACT

Statistical Dependence in Imputed High-dimensional Data for a Colorectal Cancer Study

by

Anvar Suyundikov, Doctor of Philosophy
Utah State University, 2015

Major Professor: Dr. John R. Stevens
Department: Mathematics and Statistics

The research objective of this dissertation was to provide novel statistical methods to fill potential gaps in the analyses of micro-ribonucleic acid (miRNA) data, and consequently to identify the miRNAs that contribute to cancer development. Mainly, this dissertation addressed the statistical issues raised by the statistical dependence of imputed (i.e., the missing data were replaced with substituted values) miRNA data in the colorectal cancer study. This dissertation presented a modified imputation method, the weighted KNN imputation accounting for dependence, that predicted the expression levels of missing normal samples with greater imputation accuracy than other imputation methods, and had moderate power to identify the differentially expressed miRNAs. The dissertation also improved the differential expression tests that do not assume a specific distribution of the miRNAs and account for the dependence structure of the data. Particularly, it provided an effective computational solution for the nonparametric permutation t-test by increasing its computational efficiency more than 100 times. Moreover, this dissertation contributed to the normalization literature, which includes a critical data analysis step in detecting differentially expressed miRNA features, by developing a new normalization method that removes not only technical variation but also the variation caused by the characteristics of subjects, and maintains true biological differences between arrays.
I would like to dedicate my dissertation to my beloved parents
ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. John Stevens, for spending countless hours in assistance and guidance over the completion of this dissertation. His wisdom, knowledge, and commitment to the highest standards inspired and motivated me. I would also like to thank Drs. Daniel Coster, Adele Cutler, Chris Corcoran, and Abby Benninghoff for serving on the committee and for their support and assistance throughout the entire process.

I would like to give credit to Dr. Stevens’ STAT 7570 Statistical Bioinformatics course and Dr. Cutler’s STAT 7550 Advanced Statistical Computing course where I acquired indispensable knowledge and skills to finish my dissertation.

I also thank the Division of Research Computing at Utah State University for providing technical resources to perform numerous study simulations. This research was supported by both an NIH grant, award number 1R01CA163683-01A1, and a School of Graduate Studies Dissertation Fellowship.

Anvar Suyundikov
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>iii</td>
</tr>
<tr>
<td>PUBLIC ABSTRACT</td>
<td>iv</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>vi</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>x</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xi</td>
</tr>
<tr>
<td>ABBREVIATIONS</td>
<td>xiii</td>
</tr>
<tr>
<td>ACRONYMYS</td>
<td>xiv</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2 ACCOUNTING FOR DEPENDENCE INDUCED BY WEIGHTED KNN IMPUTATION IN PAIRED SAMPLES, MOTIVATED BY A COLORECTAL CANCER STUDY</td>
<td></td>
</tr>
<tr>
<td>2.1 Introduction</td>
<td>4</td>
</tr>
<tr>
<td>2.2 Methods</td>
<td>6</td>
</tr>
<tr>
<td>2.2.1 Multiple imputation</td>
<td>7</td>
</tr>
<tr>
<td>2.2.1.1 MI using Markov chain Monte Carlo (MCMC)</td>
<td>8</td>
</tr>
<tr>
<td>2.2.1.2 MI using Expectation-Maximization (EM) with bootstrap-algorithms</td>
<td>8</td>
</tr>
<tr>
<td>2.2.2 K-Nearest Neighbors (KNN): modified and accounting for dependence</td>
<td>8</td>
</tr>
<tr>
<td>2.2.2.1 KNN in general</td>
<td>9</td>
</tr>
<tr>
<td>2.2.2.2 Choice of optimal ( k )</td>
<td>11</td>
</tr>
<tr>
<td>2.2.2.3 Accounting for dependence of KNN-imputed data</td>
<td>12</td>
</tr>
<tr>
<td>2.2.2.4 Testing for differential expression of miRNA while accounting for dependence</td>
<td>13</td>
</tr>
<tr>
<td>2.2.3 Measuring performance</td>
<td>15</td>
</tr>
<tr>
<td>2.3 Results</td>
<td>15</td>
</tr>
<tr>
<td>2.3.1 Optimal number of nearest neighbor subjects ( (k) )</td>
<td>16</td>
</tr>
<tr>
<td>2.3.2 Simulation data sets</td>
<td>16</td>
</tr>
<tr>
<td>2.3.3 Performance of imputation techniques</td>
<td>19</td>
</tr>
<tr>
<td>2.3.4 Differential expression testing</td>
<td>20</td>
</tr>
<tr>
<td>2.4 Discussion</td>
<td>22</td>
</tr>
</tbody>
</table>
# 3 Nonparametric Tests for Differential Expression in Paired Tumor-Normal Samples with Imputation-Induced Dependence

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Introduction</td>
<td>24</td>
</tr>
<tr>
<td>3.2 Methods</td>
<td>25</td>
</tr>
<tr>
<td>3.2.1 Nonparametric permutation t-test</td>
<td>26</td>
</tr>
<tr>
<td>3.2.1.1 Permutation t-test analysis</td>
<td>26</td>
</tr>
<tr>
<td>3.2.1.2 Optimization of permutation t-test analysis</td>
<td>26</td>
</tr>
<tr>
<td>3.2.2 SRT for paired tumor-normal samples of correlated miRNA data</td>
<td>29</td>
</tr>
<tr>
<td>3.2.2.1 SRTs for clustered data</td>
<td>29</td>
</tr>
<tr>
<td>3.2.2.2 SRT accounting for dependence</td>
<td>31</td>
</tr>
<tr>
<td>3.2.3 Wilcoxon SRT after whitening transformation of data</td>
<td>36</td>
</tr>
<tr>
<td>3.3 Results</td>
<td>40</td>
</tr>
<tr>
<td>3.3.1 Motivational example</td>
<td>40</td>
</tr>
<tr>
<td>3.3.2 Optimizing the performance of nonparametric permutation t-test</td>
<td>40</td>
</tr>
<tr>
<td>3.3.3 Simulation data sets</td>
<td>41</td>
</tr>
<tr>
<td>3.3.4 Differential expression testing</td>
<td>42</td>
</tr>
<tr>
<td>3.3.5 Real data application</td>
<td>47</td>
</tr>
<tr>
<td>3.4 Discussion</td>
<td>54</td>
</tr>
</tbody>
</table>

# 4 Incorporation of Demographic and Lifestyle Variables in Quantile Normalization of miRNA Data

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Introduction</td>
<td>56</td>
</tr>
<tr>
<td>4.2 Methods</td>
<td>58</td>
</tr>
<tr>
<td>4.2.1 Normalization methods for miRNA data</td>
<td>58</td>
</tr>
<tr>
<td>4.2.2 Quantile normalization: modified and incorporating demographic and lifestyle variables</td>
<td>60</td>
</tr>
<tr>
<td>4.2.2.1 Quantile normalization</td>
<td>60</td>
</tr>
<tr>
<td>4.2.2.2 Quantile normalization incorporating demographic and lifestyle variables</td>
<td>61</td>
</tr>
<tr>
<td>4.3 Results</td>
<td>63</td>
</tr>
<tr>
<td>4.3.1 Motivational example</td>
<td>64</td>
</tr>
<tr>
<td>4.3.2 Simulation data sets</td>
<td>65</td>
</tr>
<tr>
<td>4.3.3 Differential expression testing</td>
<td>66</td>
</tr>
<tr>
<td>4.3.4 Normalization accounting for unrelated covariates</td>
<td>68</td>
</tr>
<tr>
<td>4.4 Real data application</td>
<td>69</td>
</tr>
<tr>
<td>4.5 Discussion</td>
<td>71</td>
</tr>
</tbody>
</table>

# 5 Discussion

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Summary</td>
<td>73</td>
</tr>
<tr>
<td>5.2 Future work</td>
<td>75</td>
</tr>
</tbody>
</table>

# References

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>References</td>
<td>77</td>
</tr>
</tbody>
</table>
APPENDICES

A Additional plots ........................................... 87
A.1 Imputation methods ....................................... 87
A.2 Nonparametric tests ....................................... 89
A.3 Normalization methods ................................... 93
B R code ......................................................... 95
B.1 R code for imputation methods ......................... 95
B.2 R code for nonparametric methods ..................... 101
B.3 R code for normalization methods ....................... 109
C Journal copyright release form ............................ 114
D Non-committee co-author release forms ................... 115

CURRICULUM VITAE .............................................. 118
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Performances of matrix inversion operation by different methods</td>
</tr>
</tbody>
</table>

x
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 The RMSE values for different number of neighbor subjects ($k$)</td>
<td>16</td>
</tr>
<tr>
<td>2.2 The RMSE values for different imputation techniques</td>
<td>19</td>
</tr>
<tr>
<td>2.3 TPR and FDR for sample sizes of 50, 100, 200, and 400 with missingness of 10%-50%</td>
<td>21</td>
</tr>
<tr>
<td>3.1 Example of clustering subjects</td>
<td>31</td>
</tr>
<tr>
<td>3.2 Distributions of quantile normalized expressions of $hsa$-$miR$-99b-$5p$ miRNA from normal and tumor samples</td>
<td>40</td>
</tr>
<tr>
<td>3.3 TPR and FDR of all nonparametric methods for sample sizes of 200 and 400 and missing percentages of 10, 30, and 50</td>
<td>43</td>
</tr>
<tr>
<td>3.4 TPR and FDR of some of SRT tests and permutation methods for sample sizes of 200 and 400 and missing percentages of 10, 30, and 50</td>
<td>45</td>
</tr>
<tr>
<td>3.5 TPR and FDR of SRT dependent for sample size of 400 and missing percentages of 10–90</td>
<td>46</td>
</tr>
<tr>
<td>3.6 Scatterplot matrix of adjusted p-values of Permutation full, Permutation independent, Permutation case deletion, and Permutation dependent (in double negative log-scale)</td>
<td>49</td>
</tr>
<tr>
<td>3.7 Scatterplot matrix of adjusted p-values of SRT full, SRT independent, SRT case deletion, and SRT dependent (in double negative log-scale)</td>
<td>50</td>
</tr>
<tr>
<td>3.8 Scatter plot of adjusted p-values of SRT full and SRT dependent (in double negative log-scale)</td>
<td>51</td>
</tr>
<tr>
<td>3.9 Distributions of fully-observed and imputed expressions of “$hsa$-$miR$-4440” (a “false positive” in Figure 3.8) from normal and tumor samples</td>
<td>51</td>
</tr>
<tr>
<td>3.10 Distributions of fully-observed and imputed expressions of “$hsa$-$miR$-1470” (a “false negative” in Figure 3.8) from normal and tumor samples</td>
<td>52</td>
</tr>
<tr>
<td>3.11 Scatter plots of the partially expressed and fully-expressed original and imputed miRNAs of normal samples from randomly selected subjects</td>
<td>53</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4.1 Non-normalized and quantile normalized miRNA expressions of tumor samples from non-risk and risk group subjects</td>
<td>64</td>
</tr>
<tr>
<td>4.2 TPR and FDR for sample sizes of 200 and 400</td>
<td>67</td>
</tr>
<tr>
<td>4.3 TPR and FDR for sample sizes of 200 and 400 (dropping no normalization)</td>
<td>67</td>
</tr>
<tr>
<td>4.4 TPR and FDR for sample sizes of 200 and 400 for the quantile normalization and the weighted quantile normalization while accounting for unrelated covariates</td>
<td>69</td>
</tr>
<tr>
<td>4.5 Scatter plot of adjusted p-values of the CRC miRNA data, normalized by the quantile and the weighted quantile normalization methods (in log-scale). The green and red circles represent the miRNAs that are found significant only in the horizontal and vertical analyses, respectively.</td>
<td>70</td>
</tr>
<tr>
<td>A.1 TPR and FPR for sample sizes of 50, 100, 200, and 400 with missingness of 10%-50%</td>
<td>88</td>
</tr>
<tr>
<td>A.2 TPR and FPR of all nonparametric methods for sample sizes of 200 and 400 and missing percentages of 10, 30, and 50</td>
<td>89</td>
</tr>
<tr>
<td>A.3 TPR and FPR of some of SRT tests and permutation methods for sample sizes of 200 and 400 and missing percentages of 10, 30, and 50</td>
<td>90</td>
</tr>
<tr>
<td>A.4 TPR and FPR of SRT dependent for sample size of 400 and missing percentages of 10–90</td>
<td>91</td>
</tr>
<tr>
<td>A.5 Scatterplot matrix of adjusted p-values of SRT full, SRT independent, SRT case deletion, and SRT dependent based on fully-expressed miRNAs (in double negative log-scale)</td>
<td>92</td>
</tr>
<tr>
<td>A.6 TPR and FPR for sample sizes of 200 and 400</td>
<td>93</td>
</tr>
<tr>
<td>A.7 TPR and FPR for sample sizes of 200 and 400 only for conventional and weighted quantile normalizations</td>
<td>94</td>
</tr>
<tr>
<td>A.8 TPR and FPR for sample sizes of 200 and 400 for the quantile normalization and the weighted quantile normalization while accounting for unrelated covariates</td>
<td>94</td>
</tr>
<tr>
<td>D.1 Permission letter from Jennifer Herrick</td>
<td>115</td>
</tr>
<tr>
<td>D.2 Permission letter from Roger K. Wolff</td>
<td>116</td>
</tr>
<tr>
<td>D.3 Permission letter from Martha L. Slattery</td>
<td>117</td>
</tr>
</tbody>
</table>
# ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AJCC</td>
<td>American Joint Committee on Cancer</td>
</tr>
<tr>
<td>BMI</td>
<td>Body Mass Index</td>
</tr>
<tr>
<td>CRC</td>
<td>Colorectal Cancer</td>
</tr>
<tr>
<td>EM</td>
<td>Expectation-Maximization</td>
</tr>
<tr>
<td>FDR</td>
<td>False Discovery Rate</td>
</tr>
<tr>
<td>FPR</td>
<td>False Positive Rate</td>
</tr>
<tr>
<td>HRT</td>
<td>Hormone Replacement Therapy</td>
</tr>
<tr>
<td>KNN</td>
<td>K-Nearest Neighbors</td>
</tr>
<tr>
<td>MCMC</td>
<td>Markov chain Monte Carlo</td>
</tr>
<tr>
<td>MI</td>
<td>Multiple Imputation</td>
</tr>
<tr>
<td>miRNA</td>
<td>Micro-Ribonucleic Acid</td>
</tr>
<tr>
<td>mRNA</td>
<td>Messenger RNA</td>
</tr>
<tr>
<td>NIH</td>
<td>National Health Institute</td>
</tr>
<tr>
<td>NSAID</td>
<td>Non-Steroidal Anti-Inflammatory Drug</td>
</tr>
<tr>
<td>RMSE</td>
<td>Root Mean Squared Error</td>
</tr>
<tr>
<td>RNA</td>
<td>Ribonucleic Acids</td>
</tr>
<tr>
<td>SEER</td>
<td>Surveillance, Epidemiology, and End Results</td>
</tr>
<tr>
<td>SRT</td>
<td>Signed-Rank Test</td>
</tr>
<tr>
<td>TPR</td>
<td>True Positive Rate</td>
</tr>
</tbody>
</table>
## ACRONYMS

<table>
<thead>
<tr>
<th>Full</th>
<th>No missingness imposed technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case deletion</td>
<td>No imputation technique</td>
</tr>
<tr>
<td>KNN dependent</td>
<td>KNN imputation accounting for dependence</td>
</tr>
<tr>
<td>KNN independent</td>
<td>KNN imputation ignoring dependence</td>
</tr>
<tr>
<td>Permutation case deletion</td>
<td>Permutation test on the data set with no imputation</td>
</tr>
<tr>
<td>Permutation dependent</td>
<td>Permutation test on the imputed data while accounting for dependence</td>
</tr>
<tr>
<td>Permutation full</td>
<td>Permutation test on the full (no missingness imposed) data set</td>
</tr>
<tr>
<td>Permutation independent</td>
<td>Permutation test on the imputed data, but assuming independence</td>
</tr>
<tr>
<td>Datta-Satten</td>
<td>SRT based on the method of Datta and Satten [1]</td>
</tr>
<tr>
<td>SRT average</td>
<td>SRT based on the averages of ((tumor - normal)) values in each cluster</td>
</tr>
<tr>
<td>SRT case deletion</td>
<td>SRT on the data set with no imputation</td>
</tr>
<tr>
<td>SRT dependent</td>
<td>SRT on the imputed data while accounting for dependence</td>
</tr>
<tr>
<td>SRT full</td>
<td>SRT on the full (no missingness imposed) data set</td>
</tr>
<tr>
<td>SRT independent</td>
<td>SRT on the imputed data, but assuming independence</td>
</tr>
<tr>
<td>Whitening</td>
<td>SRT after whitening transformation of data</td>
</tr>
</tbody>
</table>
MicroRNAs (miRNAs) are small non-coding ribonucleic acids (RNA) molecules of about 21-23 nucleotides in length. They are considered “master regulators” of the gene expression as they have major influences in cell biology. A single miRNA has the ability to orchestrate the expression of hundreds of genes. They are highly involved in biological networks and pathways as they are often deregulated in many diseases, such as cancer. The targeting of a particular miRNA can result in normalization of genetic pathways and halting of a disease process. More information about the biology of miRNAs and their role in cancer development is given in Section 2.1.

This dissertation was motivated by statistical dependence of miRNA data in a colorectal cancer (CRC) case study that was supported by a National Institutes of Health (NIH) grant to determine the association of miRNAs with CRC in paired tumor-normal samples. Three related statistical issues from this study led to the three-paper format of the dissertation. The CRC miRNA data have the expression profiles of normal and tumor samples from each of more than 400 first available subjects with 2006 miRNA on each sample. However, in the final analysis using all available subjects, 10% to 50% of the subjects will have missing normal samples due to lack of tissue availability. This problem motivated us to concentrate Chapter 2 of this dissertation (i.e., the first paper) on the development of an imputation method that estimates the expression levels of miRNAs in missing normal samples with the smallest prediction error (defined by root mean squared error (RMSE)) while considering the dependence on non-missing samples. This chapter describes the modified version (additional proofreading was done and additional comments regarding the convergence of Expectation-Maximization (EM) algorithm were added in Section 2.2.1.2) of “Accounting for dependence induced by weighted K-Nearest Neighbors (KNN) imputation in paired samples, motivated
by this colorectal cancer study” paper, and has been published as Suyundikov et al. [6]. In this paper, we developed the KNN imputation method accounting for dependence (KNN dependent), and examined its performance compared to the multiple imputation techniques using Markov chain Monte Carlo (MCMC) and EM, as well as the case deletion technique. We also performed the differential expression testing on data sets simulated based on the characteristics of the CRC case study, imputed by various imputation methods to see how well we could identify the differentially expressed miRNAs.

Chapter 3 of this dissertation (i.e., the second paper) addresses the bimodality of distributions of miRNA data along with the imputation-induced dependency among subjects. The currently developed nonparametric methods either do not consider the correlation structure of data or are not computationally feasible for a large data set. We propose three nonparametric methods (the computationally efficient nonparametric permutation t-test accounting for dependence (Permutation dependent), the signed-rank test accounting for dependence (SRT dependent), and the Wilcoxon signed-rank test after whitening transformation of data (Whitening)) to identify the differentially expressed miRNAs in the paired tumor-normal data. The performance of the proposed methods compared to other nonparametric methods were evaluated using simulated and real data sets.

Chapter 4 of this dissertation (i.e., the third paper) considers the development of a miRNA normalization method that incorporates subject-level demographic and lifestyle covariates. The normalization procedure is a critical data analysis step that reduces technical variation and maintains true biological changes between arrays. The currently used normalization methods in miRNA data analysis were mainly developed for messenger RNA arrays and do not consider the particular characteristics of miRNA arrays. The proposed normalization method is based on the quantile normalization from Bolstad et al. [7], but it incorporates the covariates by considering the similarities among subjects in the normalization procedure, which has not been done before. We evaluated the performance of the proposed weighted quantile normalization method over the conventional quantile and the no normalization methods by applying the differential expression testing on data sets normalized by the different methods.
The dissertation concludes with a summary and discussion of future work in Chapter 5. Appendix A contains the additionally produced plots. All analyses were carried out in R [8] and C++ [9], and the graphical plots in Chapters 2, 3, and 4 were produced by using the R packages lattice [10], latticeExtra [11], and ggplot2 [12], as well as the vector graphics language TikZ [13]. This dissertation was written in LaTeX [14].
CHAPTER 2
ACCOUNTING FOR DEPENDENCE INDUCED BY WEIGHTED KNN IMPUTATION IN PAIRED SAMPLES, MOTIVATED BY A COLORECTAL CANCER STUDY

2.1 Introduction

MicroRNAs (miRNAs) are small non-coding ribonucleic acids (RNA) molecules that regulate gene expression by targeting messenger RNAs. They were first discovered in 1993 during a study into development in the nematode Caenorhabditis elegans (C. elegans) regarding the protein gene lin-14 [3]. Lee et al. [3] found that the abundance of protein lin-14 was regulated by a small RNA encoded by the lin-4 locus. This regulation was transcribed into a 22-nucleotide RNA molecule that could repress the expression of the lin-14 messenger RNA (mRNA) by directly interacting with its 3’ untranslated region.

The scientific community is currently highly interested in the functional roles of miRNAs. MiRNA biogenesis that functions properly results in the normal rates of cellular growth, proliferation, differentiation, and cell death. But the reduction or deletion of miRNAs that is caused by defects at any stage of miRNA biogenesis leads to inappropriate expression of the miRNA-target oncoproteins that causes increasing proliferation, invasiveness or angiogenesis, or decreasing levels of apoptosis [4, 5].

The miRBase database, a searchable database of published miRNA sequences and annotation, had listed 2,588 unique mature human miRNAs for July 2014 (from http://www.mirbase.org). Since miRNAs can regulate more than one target, they may regulate

\footnote{This chapter is co-authored by Jennifer Herrick, Roger K. Wolff, and Martha L. Slattery.}

\footnote{Permission to include this manuscript in the dissertation has kindly been given by all authors.}

\footnote{This chapter has been published in the journal PLOS ONE, 10(4):e0119876, 2015.}
up to more than 30% of all protein-coding genes in the human genome (from [http://www.mirnarx.com](http://www.mirnarx.com)). This makes miRNAs one of the largest regulators of gene expression.

The association between miRNAs and colorectal cancer (CRC) was reported for the first time in 2003, when the miR-143 and miR-145 genes were observed to be downregulated in CRC tumor tissues compared with normal tissues [15]. Since then, several studies have shown that miRNAs are extensively deregulated in CRC [16–18].

The miRNA data, as most other expression data, can be considered in the form of large matrices of expression levels of features (rows) for different subjects (columns). The data sets may have either some features missing in some samples or all features missing in some samples. The former case often occurs due to insufficient resolution, image corruption, dust or scratches on the slide, and other various experimental and technical reasons, while the latter case may happen due to lack of collected tissue or limited funds. As an example of the latter case, we present a case study from research to determine the association of miRNAs with CRC in paired normal-tumor samples. As part of a preliminary analysis using the first available subjects, we wanted to compare miRNA expression profiles of normal and tumor samples from each of more than 400 subjects with 2006 miRNAs on each sample. We also collected extensive information about demographic and lifestyle variables of these CRC patients. Few CRC studies have collected such extensive data for such variables. However, in the final analysis using all available subjects, 10% to 50% of the subjects will have missing normal samples due to lack of tissue availability.

The immediate objective in this CRC case study is to understand the alternatives for imputation, along with their comparative strengths and weaknesses. Specifically, we wish to know for a given imputation method whether its application to missing miRNA data among normal samples will yield accurate predictions of their actual expression levels, and how such predictions are further affected by the percentage of subjects with missing values. We further wish to understand how these results affect statistical power to detect differentially expressed miRNA while controlling for Type I error.

With the proliferation of gene expression studies over the past decade, more attention
has been paid to imputation methods for miRNA data. Conventional approaches often involve simply excluding miRNAs with missing values, replacing missing values with zeros, or imputing using row or column averages. Such options ignore the correlation structure of the data and have limited power [19]. Moreover, they do not leverage potentially informative demographic or lifestyle variables. More sophisticated options use multiple imputation based on Markov Chain Monte Carlo (MCMC) and Expectation-Maximization (EM) algorithms, which allow the incorporation of additional covariates [20,22].

In this paper, we introduced and evaluated an imputation method that accounts for the dependence induced by a weighted K-Nearest Neighbor (KNN) technique and considers the covariates, over the multiple imputation techniques using MCMC and EM with bootstrapping algorithms, as well as the case deletion technique using characteristics of this large CRC data set.

This paper is arranged in the following manner: first, we provide an overview of imputation assumptions and methods, as well as the root mean squared error (RMSE) method to assess the performance of various imputation techniques. Then we demonstrate the application of imputation techniques using simulation data sets. Finally, we conclude with a discussion of the important issues presented in the paper, such as the performance of the new KNN imputation method over the other multiple imputation techniques.

2.2 Methods

Before performing an imputation of missing data, it is necessary to know whether the missing data occur randomly, as the result of unobserved factors, or are caused by the researcher. It needs to consider two assumptions: missing at random (MAR) and missing completely at random (MCAR) [23]. The missing data are MAR when missing values are not randomly distributed across all observations but are randomly distributed within one or more subsamples of data. A variable (miRNA or $x$) can be considered MAR if the probability of observing $x$ (conditional on other observed variables) does not depend on $x$. The MCAR assumption is a special case of MAR, when the missing data values are a simple random sample of all data values. One can define the missing data as a missing not at
random (MNAR) if neither MCAR nor MAR assumptions hold. In this case, missing data cannot be imputed based on the available data. Thus, imputation techniques can only be applied to the data which satisfy either MAR or MCAR assumptions. The characteristics of the CRC miRNA data satisfy MAR assumptions because the probability of subjects having missing normal samples does not depend on the miRNA expression values in those subjects.

We consider the following methods to estimate the miRNA expression levels for missing normal samples of patients.

2.2.1 Multiple imputation

Multiple imputation (MI) was originally designed to handle missingness in public-use large data sets [23]. The application of the MI process has been extended to various big data sets including microarrays [24]. The method replaces each missing value with multiple substitute values, say \( m \), that represent the probability distribution of the missing value. A completed data set is created by each set of \( m \) draws. So the \( m \) imputations for each missing value create \( m \) complete data sets. These complete data sets are stored in an auxiliary matrix, multiply-imputed data sets with one row for missing value and \( m \) columns. The first row of this matrix corresponds to the first set of imputed values of the missing values, and so on. As the complete-data analyses are applied to each multiply-imputed data set (treating imputed values as fully observed and independent), \( m \) different sets of the parameter estimates and their variance-covariance matrices are generated. To combine the inferences from them, Rubin [23] suggests to take an average of all results, except the standard error (SE) term. The SE is constructed by the within variance of each data set as well as the variance between imputed items on each data set. These two variances are added together and the square root of them determines the SE. The author recommends to use no more than 5 imputations and sometimes as few as 2 or 3 to generate useful statistical inferences. We used \( m = 5 \) for MI techniques in our analysis. It is important to note that the complete-data analyses in MI treat the imputed data as though they had been fully observed. This approach does not consider any dependence of the imputed data on the actual fully observed data.
2.2.1.1 MI using Markov chain Monte Carlo (MCMC)

Multiple imputed data sets can be generated by the MCMC method, which is applied to an arbitrary missing data pattern that assumes multivariate normality. MCMC has been used to explore posterior probability distributions to express unknown parameters in Bayesian inference. Using this method, the entire joint posterior distribution of the unknown quantities is simulated and the parameter estimates based on the simulation are generated [25].

This process can be described in two steps. The first step is the imputation I-step which randomly draws values for missing values from the assumed distribution of missing values given observed values using the estimated mean vector and variance-covariance matrix, i.e. it draws values for \( Y_{mis}^{(t+1)} \) from \( p(Y_{mis}|Y_{obs}, \theta^t) \), where \( Y_{mis} \) and \( Y_{obs} \) are variables with missing values and observed values, respectively, and \( \theta^t \) is a parameter estimate at the \( t^{th} \) iteration.

The posterior P-step randomly simulates the population mean vector and variance-covariance matrix from the complete sample estimates, i.e. it draws \( \theta^{(t+1)} \) from \( p(\theta|Y_{obs}, Y_{mis}^{(t+1)}) \). These new estimates are then used in the I-step. This creates a Markov chain \((Y_{mis}^{(1)}, \theta^{(1)}), (Y_{mis}^{(2)}, \theta^{(2)}), \ldots \) , which converges in distribution to \( p(Y_{mis}, \theta|Y_{obs}) \). Enough iterations are carried out to have reliable results for a multiply imputed data set and to converge to its stationary distribution from which we can simulate an approximately random draw of the missing values [26].

2.2.1.2 MI using Expectation-Maximization (EM) with bootstrapping algorithms

The EM algorithm is a very general iterative algorithm for maximum likelihood estimation of missing data [20]. One assumes a model for the data, maximizes the likelihood under the assumed model, obtains parameter estimates, and makes inferences based on the parameter estimates. The explicit form of parameter estimates does not usually exist for missing data. Here numerical methods like the Newton-Raphson algorithm are very complicated to use. Thus, one can apply the EM algorithm, which is an iterative method for
maximizing the likelihood in missing data [21]. Compared to the Newton-Raphson algorithm, the EM algorithm is slower, but it increases the likelihood with each iteration and may converge to a maximum for the distribution with one mode [21, 27]. Note that if the conditions of Theorem 3 of Wu [27] are not satisfied, the EM algorithm might converge to a stationary point that is neither a local maximum nor a local minimum (a saddle point). Wu [27] highlight that the convergence to stationary value or local maximum or global maximum depends on the choice of starting values. He recommends to try different starting values with the EM iterations.

The EM algorithm consists of two steps, the Expectation (E) and the Maximization (M) steps. The algorithm calculates the conditional expectation of missing values given non-missing values and current parameter estimates in the expectation step. In the maximization step, the calculated expected values are used to maximize the likelihood of the complete data. These steps are iterated until the maximum likelihood of the data converges.

The maximization step can be computationally expensive, which can make the EM algorithm unattractive. Fortunately, the EM with bootstrapping algorithm resolves this problem. It uses the conventional EM algorithm on multiple bootstrapped samples of the original missing data to draw values of the complete-data parameters. Then the EM with bootstrapping algorithm draws imputed values from each set of bootstrapped parameters, replacing the missing values with these draws. It can impute missing values in much less time than the EM algorithm itself [22].

2.2.2 K-Nearest Neighbors (KNN): modified and accounting for dependence

2.2.2.1 KNN in general

The conventional KNN method replaces missing values using the $k$ most similar non-missing subjects’ values [28, 29]. This method can impute both discrete attributes (using the most frequent value among the k-nearest neighbors) and continuous attributes (using the mean among the k-nearest neighbors).

Troyanskaya et al. [19] implemented the KNN method, which weights the contribution
of each nearest neighbor by its similarity to the subject with the missing value. In our CRC study, the weights of the nearest neighbors in the imputation of missing values are measured by the Euclidean distance metrics of demographic and lifestyle variables (continuous variables should be scaled by their standard deviations) such that the nearer neighbors to the subject contribute more to its imputation than the more distant ones. Based on the weighting method of Troyanskaya et al. [19], we briefly outline our weight calculations here. Let \( k \) be the chosen number of nearest neighbors, \( D_{i1} \leq \ldots \leq D_{ik} \) be the sorted distances of the \( k \) nearest neighbors from normal-missing subject \( i \), and \( D_i^{(\text{max})} \) be the maximum distance (among all fully-observed subjects) from subject \( i \). Then the weights \( a_{i1}, \ldots, a_{ik} \) among the \( k \) nearest neighbors for subject \( i \) are obtained as follows:

\[
 w_{il} = 1 - \frac{D_{il}}{D_i^{(\text{max})}} \quad \text{and} \quad a_{il} = \frac{w_{il}}{\sum_{t=1}^{k} w_{it}}. \tag{2.1}
\]

These weights are used by the weighted KNN method to impute missing expression values of a particular gene as in Equation (2.2) of Section 2.2.3.

Our proposed imputation method accounts for the dependence induced by weighted KNN and can use the additional covariates such as demographic, general health, genetic, and lifestyle variables, as well as other biologically related information. The proposed imputation method takes advantage of the conventional KNN [28, 29] and further developed weighted KNN [19] imputation methods’ robustness to missing data, non-parametric approach, and speed in estimating missing values for microarray data, while considering the correlation structure of the data. In order to impute missing samples in the above mentioned motivating CRC case study, the proposed method has been modified to impute expressions for all miRNA of missing normal samples based on multivariate covariates (demographic and lifestyle variables) and to account for the dependence of the imputed data in subsequent differential expression tests. The demographic and lifestyle variables considered in this paper are five continuous (age, number of cigarettes/day, calories, BMI (Body mass index), and lutein and zeaxanthin concentration) and five binary variables (gender, recent aspirin/NSAID (Non-steroidal anti-inflammatory drug) use, recent smoker, menopause, and
post menopause taking HRT (Hormone replacement therapy) within 2 years statuses).

This modified KNN technique imputes all miRNA expression levels of missing normal samples by finding the $k$ most similar subjects, based on the distance matrices of demographic and lifestyle covariates of patients (instead of gene expression levels as in conventional KNN methods) and produces variance-covariance matrices for each miRNA. For example, we can estimate the miRNA expression levels in missing normal tissues from a particular subject, based on the expression levels of scanned normal tissues from subjects who have similar demographic and lifestyle covariates.

Another advantage of this method is that it can simultaneously integrate multivariate covariates by aggregating and normalizing their distance matrices (Euclidean, Manhattan, Minkowski, etc.) to find the nearest neighbor subjects. Specifically, two between-subject distance matrices are constructed based on the fully observed continuous and discrete covariates separately, using Euclidean and Manhattan distances, respectively. These two distance matrices are normalized by scaling between 0 and 1 [30] and aggregated by taking the weighted average of each distance matrix to achieve a single between-subject distance matrix.

2.2.2.2 Choice of optimal $k$

There have been many studies carried out to determine the optimal choice of $k$ for the KNN algorithm. Duda and Hart [29] suggest to use the square root of the average number of complete cases after missing data removal, rounded to the nearest odd integer. The simulation studies of different $k$ on likert scale data [31] show the square root of the number of complete cases which is rounded to the nearest odd integer is a suitable choice for $k$. Moreover, Batista and Monard [32] report on $k = 10$ for large data like from microarrays. Troyanskaya et al. [19] argue that the imputation method is fairly insensitive to the choice of $k$ in the range 10-20. As $k$ gets larger, the average distance to the neighbors increases which implies that the imputed value could be less accurate and the imputation time will increase.
However, the choice of a small \( k \) diminishes the KNN performance because the imputation process overemphasizes a few dominant genes (or subjects in our modification) in estimating the missing values. On the other hand, a large \( k \) may include genes (or subjects) that are significantly different from the missing values, which may result in degrading the imputation performance.

### 2.2.2.3 Accounting for dependence of KNN-imputed data

Because the weighted KNN-imputed expression values are linear combinations of the fully observed subjects’ expression values, the imputed values are not, in general, independent of the fully observed values. The modified KNN-based imputation method has the advantage of considering this dependence induced by weighted KNN by providing variance-covariance matrices for each miRNA, which can be used when searching for differentially expressed miRNAs. We refer to this method as “KNN dependent,” while referring to the KNN imputation method that ignores the dependence as “KNN independent” in this paper. The algorithms of both methods work almost the same as the algorithms of the conventional KNN-based methods, except it treats the rows as subjects or samples, and the columns as miRNAs.

To see how the proposed imputation method estimates the miRNA expression levels in missing normal samples and accounts for the dependence induced by the weighted KNN, suppose that in the CRC study of \( N \) subjects, we want to estimate expression levels of \( G \) miRNAs for normal samples of missing \( S \) subjects using demographic and lifestyle covariate data. For each normal-missing subject \( i \), we find the \( k \) most similar subjects with non-missing normal samples (say subjects \( i_1, \ldots, i_k \)), and impute the missing miRNA expression values by multiplying the miRNA expressions from normal samples of the \( k \) subjects with their corresponding weights, which are generated from the between-subject distance matrix. The imputation of the expression level of miRNA \( j \) in missing normal sample \( i \) will be produced as in (2.2):

\[
\hat{x}_{ij} = a_{i1}x_{i1j} + a_{i2}x_{i2j} + \ldots + a_{ik}x_{ikj}.
\]  

(2.2)
Here, \( i = 1, \ldots, S \) and \( j = 1, \ldots, G \). \( x_{lj} \) is the observed expression value of miRNA \( j \) in the observed normal sample of subject \( l \), and \( a_{lj} \) is the weight of the subject in the imputation. The weights \( a_{i1}, \ldots, a_{ik} \) are obtained as outlined in Equation (2.1) above. Equation (2.2) can be generalized to Equation (2.3):

\[
\hat{X}_\sim = \sim A^T X.
\]  

Here, \( \hat{X}_\sim \) is an \( S \times G \) matrix of imputed normal tissue expression values, \( \sim A \) is a \((N-S) \times S\) matrix of weights \( a \), and \( \sim X \) is a \((N-S) \times G\) matrix of observed normal tissue expression values. In column \( i \) of \( \sim A \), the only non-zero elements are in rows \( i_1, i_2, \ldots, i_k \), and are the coefficients \( a_{i1}, a_{i2}, \ldots, a_{ik} \) in Equation (2.2).

The variance-covariance matrix of the normal tissue expression for miRNA \( j \) will be calculated as in (2.4), assuming the order in the data is the fully observed \( N - S \) subjects followed by the \( S \) normal-missing subjects:

\[
\Sigma_j \sim \sigma^2_j = \begin{pmatrix}
I_\sim & \sim A \\
\sim A^T & \sim \sim A
\end{pmatrix} \sigma^2_j.
\]  

Here, \( \sigma^2_j \) is the variance of miRNA \( j \) and \( I_\sim \) is the \((N-S) \times (N-S)\) identity matrix of non-missing subjects to represent the independence among non-missing subjects. The matrix part of the right-hand side of Equation (2.4) is denoted by \( \Sigma_j \).

2.2.2.4 Testing for differential expression of miRNA while accounting for dependence

The paired t-test may be used to check whether the miRNAs are differentially expressed in paired normal-tumor samples while accounting for the dependence induced by the imputation method. The paired t-test can be simplified to a one sample t-test of the difference of normal and tumor samples. The per-miRNA null hypothesis is that the difference of mean expression levels of miRNAs between normal and tumor samples is equal to zero. The test statistic for miRNA \( j \) can be found beginning with the following equation,
as discussed in chapter 3 of [34].

\[ D_{\sim j} \sim 1\mu_j + \epsilon. \]  

(2.5)

Here, \( D_{\sim j} \) is a \( N \times 1 \) vector of the difference of the \( j^{th} \) miRNA expressions for normal and tumor samples, \( \mu_j \) is a single parameter representing the difference of mean expression levels of miRNA \( j \) between normal and tumor samples, and \( 1 \sim \) is \( N \times 1 \) vector of 1’s.

\[ \text{Var}(\epsilon) = \sigma_j^2 V_{\sim j}; \]  

where \( V_{\sim j} \) is the variance-covariance matrix of the tumor-normal difference in miRNA expression values for miRNA \( j \), i.e., \( V_{\sim j} = I + \Sigma_{\sim j} \), and needs to be a positive definite matrix.

The mean tumor-normal difference for miRNA \( j \) can be estimated by (2.6):

\[ \hat{\mu}_j = (1^{T}V_{\sim j}^{-1}1)^{-1}1^{T}V_{\sim j}^{-1}D_{\sim j}. \]  

(2.6)

The \( \hat{\mu}_j \) in (2.7) can be substituted from (2.6):

\[ \hat{\sigma}_j^2 = \frac{(D_{\sim j} - 1\mu_j)^{-1}V_{\sim j}^{-1}(D_{\sim j} - 1\mu_j)}{N - 1}. \]  

(2.7)

Then, the estimated variance of \( \hat{\mu}_j \) would be calculated as in (2.8):

\[ \text{Var}(\hat{\mu}_j) = \hat{\sigma}_j^2(1^{T}V_{\sim j}^{-1}1)^{-1}. \]  

(2.8)

Finally, the test statistic will be found using (2.9) with a degree of freedom of \( N - 1 \).

\[ t = \frac{\hat{\mu}_j}{\sqrt{\text{Var}(\hat{\mu}_j)}}. \]  

(2.9)

This paired t-test can be used with the other imputation methods by replacing \( \Sigma_{\sim j} \) with the identity matrix, which represents the assumed independence of imputed miRNA values.
2.2.3 Measuring performance

The performance of the imputation methods on miRNA data was evaluated through RMSE. The RMSE-based evaluation technique is the most commonly used method to compare similarity between true expression values and imputed expression values. Various variants of RMSE measures are used in the literature: the non-normalized RMSE measure [35] and the normalized RMSE measure by different normalizing constants: average value over all observations in complete data [19], standard deviation of the values in complete data over missing entries [36, 37], and root mean square of the values in complete data over missing entries [38]. However, all above mentioned various RMSE measures provide highly similar results [39].

In the motivating CRC case study, all miRNA expression levels of up to 50% missing normal samples, i.e. up to 50% missing rows (samples) of miRNA data must be imputed. Thus, the non-normalized RMSE that measures the difference between the imputed part of matrix and the original part of matrix, divided by the number of missing cells, can be used. RMSE is calculated as (2.10):

$$RMSE = \sqrt{\frac{1}{S \cdot G} \sum_{i=1}^{S} \sum_{j=1}^{G} (x_{ij} - \hat{x}_{ij})^2}$$

where \( i = 1, ..., S \) and \( j = 1, ..., G \). \( x_{ij} \) is the original value for missing sample \( i \) and miRNA \( j \), while \( \hat{x}_{ij} \) is the imputed value for missing sample \( i \) and miRNA \( j \).

2.3 Results

We evaluated the performance of the proposed imputation method, which accounts for the dependence induced by weighted KNN and considers the demographic and lifestyle covariates (KNN dependent), over the weighted KNN ignoring the dependence (KNN independent), MI techniques using MCMC and EM with bootstrapping algorithms, as well as the case deletion technique which only considers fully-observed subjects [20] using simulated data sets.
Fig. 2.1. The RMSE values for different number of neighbor subjects ($k$)

2.3.1 Optimal number of nearest neighbor subjects ($k$)

Figure 2.1 shows the effect of the number of neighbor subjects, $k$, used in the KNN imputation method on the RMSE values for simulated data sets with different number of subjects and percent of normal-missing subjects. The RMSE decreases, i.e. the performance of KNN imputation increases, while the value of $k$ increases. The falling of RMSE values slows down after $k$ value of 10, and becomes approximately the same for the rest of $k$ values. The imputation performance becomes approximately insensitive to the value of $k$ within the range of 10 - 25 neighbor subjects. Thus, we used 10 nearest neighbor subjects to estimate the miRNA expression levels of normal samples for missing subjects.

2.3.2 Simulation data sets

While we have complete normal and tumor sample data for more than 400 subjects in the CRC study, we compare imputation methods using simulated data to have clearly
defined power and Type I errors. The imputation analyses were performed on normally
distributed paired data matrices of $G = 2000$ miRNA features (columns) for each of the
normal and tumor samples with sample sizes of $N = 50, 100, 200, \text{ and } 400$ subjects (rows).
We simulated expression levels of miRNAs for normal and tumor samples by controlling true
differentially expressed miRNAs of tumor samples across all simulations. Particularly, all
miRNA features of normal samples and only non-differentially expressed miRNA features
of tumor samples were simulated based on $\mu = 2$ and $\sigma = 1.25$, while the differentially
expressed miRNA features of tumor samples, which consisted of 20% of all miRNA features
of tumor samples, were simulated based on $\mu = 2.5$ and $\sigma = 1.25$. This 20% differential
expression rate as well as this mean tumor-normal difference of 2.5 and standard deviation
of 1.25 were chosen based on characteristics of the motivating CRC study. We randomly
applied missingness from 10 to 50 percent of the normal data rows. We performed 25
simulations for each sample size with different percent missingness.

To ensure that the simulated data sets reflected the characteristics of the CRC study,
and that the demographic and lifestyle variables carried some useful information for im-
putation, the multivariate covariate data sets with demographic and lifestyle variables of
subjects were simulated based on $z$ randomly selected true differentially expressed miRNA
expression levels using the characteristics of the CRC case study covariate data. For exam-
ple, a continuous variable such as age of subjects was simulated as in (2.11):

$$\hat{C} = \beta_0 + \sum_{j=1}^{z} \beta_j x_j + \epsilon. \quad (2.11)$$

Here, $j = 1, \ldots, z$, $\hat{C}$ is a simulated value of age, $\beta_0$ is the mean age of the patients in
CRC case study, and $\beta_j$ is uniformly distributed with a minimum and a maximum of up to
5% of the minimum and the maximum of the CRC case study patients’ age, respectively. In
this paper, we used 2% of the minimum and the maximum of the continuous variables with
$z = 20$, which was selected for computational simplicity, to simulate variables with similar
characteristics of CRC case study covariates. $x_j$ is the expression of truly differentially
expressed miRNA $j$ in tumor, and the error term $\epsilon$ is normally distributed with zero mean
\( (\mu = 0) \) and variance of 10\% of variance of the patients’ age \( (\sigma^2 = 0.1 \times \sigma_{\text{age}}^2) \).

The binary variables such as gender of subjects was simulated using a logistic regression model in equations (2.12) and (2.13):

\[
\log \frac{p}{1-p} = \beta_0 + \sum_{j=1}^{z} \beta_j x_j. \tag{2.12}
\]

Here, \( p \) is the probability of gender=female, say.

Equation (2.12) can be rewritten as Equation (2.13):

\[
\hat{P} = \left[1 + \exp\left(-\left(\beta_0 + \sum_{j=1}^{z} \beta_j x_j\right)\right)\right]^{-1}. \tag{2.13}
\]

Here, \( \hat{P} \) is a simulated probability of gender=female, \( \beta_0 \) is the mode of the patients’ gender in the CRC case study, and \( \beta_j \) is uniformly distributed as \( U[-0.5, 0.5] \). To ensure variability in simulated binary variables, we calculate \( \hat{P}' \) as in Equation (2.14):

\[
\hat{P}' = \frac{\hat{P} - \min(\hat{P})}{\max(\hat{P}) - \min(\hat{P})}. \tag{2.14}
\]

In our simulated study, we had denoted as a male if the value of \( \hat{P}' \) was between 0 and 0.5, and as a female if the \( \hat{P}' \) was bigger than 0.5 but less than or equal to 1.

Demographic and lifestyle variables were thus simulated based on characteristics of five continuous (age, number of cigarettes/day, calories, BMI, and lutein and zeaxanthin concentration) and five binary variables (gender, recent aspirin/NSAID use, recent smoker, menopause, and post menopause taking HRT within 2 years statuses) from the CRC study.

We carried out the performance analyses as follows: first, we called arbitrarily the subjects with missing normal samples. Then, we imputed expression levels of the missing normal samples using the imputation methods mentioned in Section 2.2. We evaluated the performance of these imputation methods against the initial generated data matrices by calculating the RMSE for such simulated data set. Moreover, we carried out the differential expression analyses on the imputed data sets to check whether the KNN dependent method
has an equal statistical power in finding differentially expressed miRNA as other imputation techniques.

### 2.3.3 Performance of imputation techniques

The performance of the modified KNN method was assessed over MI techniques using MCMC and EM with bootstrapping algorithms for data matrices with different number of subjects and different percents of normal missing subjects. In Figure 2.2, the modified KNN method shows consistently better performance than other imputation techniques (systematically lower RMSE values) for sample sizes of 50, 100, 200, and 400 subjects, with missing percentages of 10 - 50.

The KNN imputation method also shows a robustness to increasing the percent of missing normal samples and the number of subjects in miRNA data sets. It keeps relatively the same performance for all levels of missing percents and number of subjects.
Moreover, the KNN imputation method required much less computational expense than the MI techniques using MCMC and EM with bootstrapping algorithms. For example, to impute the expressions of 50% missing normal samples in 400 subjects on a machine with CPU speed of 1.86 GHz and 2 GB RAM, the KNN method took approximately 35 minutes, whereas MCMC and EM with bootstrapping algorithms took approximately 10 and 5 hours, respectively.

### 2.3.4 Differential expression testing

We applied the paired t-test to the data sets, which were imputed by various imputation methods, to see how well we could identify differentially expressed miRNAs. First, we obtained a test statistic and a p-value for each miRNA feature in each imputed data set by controlling the false discovery rate (FDR) at 0.05 within each simulation. Then, we calculated the true positive rate (TPR), the false positive rate (FPR), and the false discovery rate (FDR) based on the miRNAs which were controlled as truly differentially expressed in the simulations. The TPR and FPR were defined and calculated as in Bolstad [40] and Stevens et al. [41], and the FDR was defined as in Benjamini and Hochberg [42].

Figure 2.3 shows the performance (including power and FDR control) of the paired t-tests on the data sets imputed by the KNN dependent, the KNN independent, the MI using MCMC and MI using EM algorithms, on the data set with no imputation (Case deletion), as well as on the full data set (Full) for the number of subjects of 50, 100, 200, and 400 with the missing percent of normal samples of 10%-50%. The scatter plots of TPR and FPR for the similar conditions are represented in Figure A.1 in Appendix A.1.

From Figure 2.3 we can see that the power (i.e., the TPR values) increases with larger sample sizes. For 400 subjects and 50% missing normal samples, which are the characteristics of the CRC case study, there are clear clusterings of TPR and FDR values, separately for Full, for KNN dependent and Case deletion, and for KNN independent, MCMC, and EM methods. Although the KNN dependent has slightly lower power than the other imputation methods (the TPR values are in the range of 0.93-0.98 for 400 subjects and 50% missing), it controls the FDR values below the threshold of 0.05, which is represented by
Fig. 2.3.  TPR and FDR for sample sizes of 50, 100, 200, and 400 with missingness of 10%-50%
red dotted lines in the figures. The KNN independent, the MCMC, and the EM with bootstrapping algorithms have the highest power (the TPR values are in the range of 0.985-1 for 400 subjects and 50% missing), but lack control of the FDR, i.e. the FDR values cross the threshold of 0.05 for all number of subjects and missing percentages. The case deletion method shows the lowest power, but maintains control of the FDR for all number of subjects and percentages of missing normal samples.

2.4 Discussion

The imputation accuracy of the proposed KNN imputation method, using the aggregated metric distance matrices of the demographic and lifestyle data, in the simulation data sets was higher than that of the MI methods using MCMC and EM with bootstrapping algorithms. Moreover, the proposed KNN method was robust and imputed the miRNA features of missing normal samples with less computational expense than the other imputation methods.

The differential expression tests of the KNN imputed data sets show that KNN dependent provided greater power than the case deletion approach and maintained control of the FDR. KNN independent, as well as MCMC and EM with bootstrapping algorithms had higher power than the power of KNN dependent, but failed to control the FDR. These effects are more clear for larger missing percents and number of subjects.

Depending on the study goals, researchers could select the KNN independent (achieving more power and higher proportion of false discoveries) or the KNN dependent methods (moderate loss of power but lower proportion of false discoveries). In the motivating CRC study, the chosen approach is KNN dependent, with moderate loss of power but maintaining control of the FDR.

The case deletion method showed the lowest power to identify differentially expressed miRNAs, though it had similar FDR control as the KNN dependent method.

In this paper, we applied the paired t-test to identify differentially expressed miRNAs from normally distributed simulated miRNA data while accounting for the dependence structure of the imputed data. However, miRNA data can be noisy and not normally
distributed. Currently available nonparametric tests may also not be directly applicable because they assume independence. In this respect, it is challenging to construct a statistical model which tests for significant miRNAs from paired samples while accounting for the dependence. Our future work is to develop a nonparametric t-test method which enables paired t-tests on a large number of miRNA data, using permutations with manageable computational expense, while accounting for the dependence induced by KNN imputation.

R code for the application of imputation methods and the differential expression tests can be found in Appendix B.1.
CHAPTER 3
NONPARAMETRIC TESTS FOR DIFFERENTIAL
EXPRESSION IN PAIRED TUMOR-NORMAL SAMPLES
WITH IMPUTATION-INDUCED DEPENDENCE

3.1 Introduction

The miRNA of the paired tumor-normal CRC data tend to have bimodal expression distributions, and as discussed in Section 2.2.2.3, imputation induces dependency among subjects. In this context, differential expression tests that do not assume a specific distribution but account for the dependence structure of the data are preferable.

The nonparametric permutation t-test can be applied to identify the differentially expressed miRNAs in the CRC case study. Although the t-test assumes a normal distribution of data, the differential expression testing procedure becomes nonparametric by estimating p-values from permuted data sets [43]. Thus, the differentially expressed miRNA features can be identified by permuting the tumor/normal labels of the paired tumor-normal samples within each subject. Dependence among subjects under KNN imputation can be accounted for by using the estimated covariance matrix in each permutation’s test statistic calculation. In this chapter, we improved the calculation of test statistics while accounting for the dependence structure by rewriting R code in C++ [9] using the R package Rcpp [44, 45] in order to speed up the computational process of the permutation algorithm.

Another nonparametric test that can be used in testing for differential expression in the paired tumor-normal miRNA data is the Wilcoxon signed-rank test [46]. This signed-rank test (SRT) is used to test the equality of means of two paired samples that are not normally distributed. However, this test cannot account for the dependence structure of data. Several SRTs have been proposed to incorporate correlation effects of observations
within clusters in the literature. But these tests do not directly reflect the characteristics of the CRC case study. The miRNA features of the paired tumor-normal CRC data have bimodal expression distributions and have imputation-induced dependence among subjects. In this chapter, we also modified the SRT for clustered data from Datta and Satten [1] to incorporate the imputation-induced correlation among subjects by replacing the estimated variances among the ranks with the elements of variance-covariance matrices of the miRNAs.

Another considered nonparametric approach to identify the differentially expressed miRNAs is to use a whitening transformation of the data first, and then apply the conventional Wilcoxon SRT. The whitening transformation is a decorrelation method that transforms a set of random variables having the covariance matrix $\Sigma$ into a set of new random variables whose covariance is an identity matrix [47]. This approach removes the correlation structure of data and then performs a Wilcoxon SRT to identify the differentially expressed miRNAs.

This chapter is organized as follows: we provide an overview of current and proposed nonparametric methods developed for the paired tumor-normal miRNA data in Section 3.2. In Section 3.3 we demonstrate the application and the performance of proposed nonparametric methods using simulation and real data sets. In Section 3.4 we draw the conclusions from our analysis.

### 3.2 Methods

We propose three modified nonparametric methods: (1) a computationally efficient nonparametric permutation t-test accounting for dependence (Permutation dependent), (2) a SRT accounting for dependence (SRT dependent), and (3) a Wilcoxon SRT after a whitening transformation of data (Whitening) to identify differentially expressed miRNA features from bimodal paired tumor-sample miRNA data.
3.2.1 Nonparametric permutation t-test

3.2.1.1 Permutation t-test analysis

One possible nonparametric test is the permutation t-test. Dudoit et al. [48] used a permutation based t-test to identify differentially expressed genes in cDNA microarray data. They performed the differential expression tests by computing first a test statistic for each gene by permuting the subjects’ treatment/control labels (thus preserving the dependence structure between the genes), and then obtaining adjusted p-values through a multiple testing procedure. The authors’ permutations assume that the membership of the subjects to the control or treatment groups is independent of gene expression. They suggest to consider all possible $B$ permutations of the subjects when computationally feasible, otherwise to use only a random subset of $B$ permutations (including the observed).

In testing multiple genes, Dudoit et al. [48] consider the family-wise Type I error rate of Westfall [49]. Particularly, they used a multiple testing procedure based on the maxT adjusted p-values to identify differentially expressed genes. This multiple testing procedure is discussed in more detail in Ge et al. [50]. The mt.maxT function from the R package multtest [51] computes permutation adjusted p-values for step-down maxT multiple testing procedure based on Algorithms 2.8 and 4.1 in Westfall [49].

Troyanskaya et al. [43] compared the permutation t-test approach with other nonparametric methods to identify differentially expressed genes in non-normally distributed microarray data. They performed 50,000 permutations for each data set. From each permuted data set, they calculated the t-statistic ($t_{\text{perm}}$) for each gene $j$ and counted how many times it exceeded the observed t-statistic ($t_{\text{obs}}$) for that gene. At the end, the p-values of multiple tests were corrected by using the Bonferroni correction.

3.2.1.2 Optimization of permutation t-test analysis

The nonparametric permutation t-test can be applied to identify the differentially expressed miRNAs in the CRC case study by permuting the tumor/normal labels of the paired tumor-normal samples within each subject. Since the number of all possible permutations
of the miRNA data with 400 subjects is huge \( (B = 2^{400}) \), a large random subset of \( B \) permutations can instead be performed. In our analysis, we used \( B = 50,000 \) as in Troyanskaya et al. [43].

For miRNA \( j \), the KNN imputation accounting for dependence (KNN dependent) (see Chapter 2) produces a covariance matrix \( \Sigma_{\sim j} \) – with rows and columns for each tumor and normal sample; it will be a diagonal matrix for the fully-observed samples. For permutation \( b \), the algorithm keeps track of the subjects in which the tumor/normal labels of the paired tumor-normal samples were permuted, switches the corresponding elements of covariance matrices in normal sample to the corresponding elements of covariance matrices in tumor sample, and creates a new covariance matrix \( \Sigma'_{\sim j,b} \). This permuted covariance matrix, which represents dependence among subjects under KNN imputation, can be used in the calculation of test statistic \( t_{j,b} \) (see Section 2.2.2.4).

The following algorithm performs the permutation t-test procedure to find the differentially expressed miRNA features in the CRC case study. For the \( b \)th permutation, \( b = 1, \ldots, B \), using \( N \times 2G \) miRNA data matrix \( X\sim \), where \( N \) is the total number of subjects and \( G \) is the total number of miRNAs of normal or tumor samples:

1. From the set of all possible permutations, choose one at random, and let \( n \) be the number of subjects selected to have their tumor/normal labels switched.
2. Permute the tumor/normal labels of the \( n \) selected rows (subjects) of the data matrix \( X\sim \).
3. Switch the corresponding elements of the \( n \) subjects in variance-covariance matrices \( \Sigma_{\sim 1}, \ldots, \Sigma_{\sim G} \) and create new covariance matrices \( \Sigma'_{\sim 1,b}, \ldots, \Sigma'_{\sim G,b} \).
4. Compute test statistics \( t_{1,b}, \ldots, t_{G,b} \) for each hypothesis. The test statistic for miRNA \( j \) is calculated as discussed in Section 2.2.2.4.
After the $B$ permutations are done, for two-sided alternative hypotheses, the permutation p-value for $H_j$: “miRNA $j$ is not DE” is

$$p^*_j = \frac{\sum_{b=1}^{B} I(|t_{j,b}| \geq |t_j|)}{B}$$

(3.1)

where $j = 1, \ldots, G$ and $I(\cdot)$ is the indicator function, equaling 1 if the condition in parentheses is true, and 0 otherwise [48].

As in Section 2.2.2.4, the per-miRNA null hypothesis $H_j$ is that the difference of mean expression levels of miRNAs between normal and tumor samples is equal to zero. The permutation p-values are adjusted by the false discovery rate of Benjamini and Hochberg [42].

However, running the above mentioned algorithm in R takes about 18 minutes on a machine with CPU speed of 1.86 GHz and 2 GB RAM to perform a single permutation and calculation of test statistics for miRNA data with 400 subjects and 2000 miRNAs in each tumor-normal sample (which are the characteristics of the CRC case study) while accounting for the dependence among subjects. The calculation of test statistics for a subset of 50,000 random permutations would take approximately 15,000 hours or 625 days. Thus, one needs to employ high-performance computing resources to decrease the computational time.

In this chapter, we applied the technique to rewrite permutation R code in a faster language, C++ [9]. Eddelbuettel and Sanderson [52] developed a new method of avoiding the speed penalty in R by using the Rcpp extension package [44] in conjunction with the Armadillo C++ linear algebra library [53]. The RcppArmadillo [52] package is more efficient than the Rcpp package because it is applied with the highly expressive Armadillo C++ library, which provides vector, matrix and cube types as well as a subset of trigonometric and statistics functions [52]. We also used the code optimization techniques from Wickham [54] to decrease computational time of our algorithm in R / C++. Particularly, we used the R packages lineprof [55] and shiny [56] to perform and visualize line profiling of our code to find bottlenecks (the slowest parts of code).
3.2.2 SRT for paired tumor-normal samples of correlated miRNA data

3.2.2.1 SRTs for clustered data

Another nonparametric test that can be used in testing for differential expression for paired tumor-normal miRNA data is the Wilcoxon SRT [46]. This SRT is used to test the equality of means of two paired samples that are not normally distributed. It is the nonparametric analogue of the t-test for paired samples. However, it can be applied only if the pairs are independent and come from the same population [46]. In this section, we consider SRTs that account for the dependence structure of data.

There are several rank-sum and signed-rank tests in the literature that have been proposed to consider correlation among observations within clusters. One of the first group of researchers who accounted for clustering effects with the Mann-Whitey U-test (another name of Wilcoxon rank-sum test) were Rosner and Grove [57]. Then, Rosner et al. [58] proposed the modified Wilcoxon rank-sum test that allowed incorporating clustering effects for ranked data where the cluster (subject) was the unit of randomization and the subunit (eye) was the unit of analysis. They calculated a rank-statistic for clustered ophthalmological data for the case when there was a correlation between ordinal responses for two observations (right and left eyes) in the same cluster. Rosner et al. [59] extended the rank-sum test from Rosner et al. [58] to account for group membership at the subunit level. They assumed the ocular itching scores of multiple subunits (e.g., several eyes) within a cluster (e.g., person) in an ocular allergy study would be dependent, while the change scores for subunits (e.g., eyes) from different clusters (e.g., persons) would be independent. Rosner et al. [60] also modified the Wilcoxon SRT for paired comparisons of clustered data. They were motivated by the case that the conventional Wilcoxon SRT for paired data (pre- and post-treatment responses) could not be used to compare two eyes of the same subject.

In contrast to previous rank sum tests for clustered data, Datta and Satten [61] modified the rank-sum test for the case when members of the same cluster belong to different groups (for example, treatment and control), or when the correlation between cluster members differs across groups, but the clusters are disjoint. They assumed that observations from
the two groups have the same distribution under $H_0$. They selected one observation at random from each cluster and used them to construct a Wilcoxon rank-sum statistic. This way, the test statistic is computed with independent observations. They calculated a final test statistic by averaging the Wilcoxon rank-sum statistics over all possible independent samples formed by selecting one observation per cluster. They used the estimated variances among the ranks to ensure the validity of the test when both groups have the same scale parameter under $H_0$. But the variance estimates may allow for the intra-cluster correlation to depend on group memberships.

Datta and Satten [1] extended the Datta and Satten [61] rank-sum test to obtain a novel SRT to compare two samples when the pairs are clustered. They considered the null hypothesis that the distribution of pairwise differences for a randomly chosen pair in a randomly chosen cluster is symmetric (around 0) against the alternative that the distribution is not symmetric. They discussed two types of distribution functions that can be used depending on what hypothesis is tested. Those distribution functions are defined in equations (3.2) and (3.3):

$$F(x) = \mathbb{E}\left\{ \frac{1}{n_i} \sum_{j=1}^{n_i} I(X_{ij} \leq x) \right\}$$  \hspace{1cm} (3.2)

$$F'(x) = \mathbb{E}\left\{ \frac{1}{N} \sum_{i=1}^{M} \sum_{j=1}^{n_i} I(X_{ij} \leq x) \right\}.$$  \hspace{1cm} (3.3)

Here $X_{ij}$ is a pair difference for the $j$th pair in the $i$th cluster, $n_i$ is the $i$th cluster size, and $N = \sum_i n_i$ with $i = 1, ..., M$ (i.e., $N$ is a total number of cluster sizes and $M$ is the number of clusters).

The authors suggested that “the functions $F(x)$ and $F'(x)$ can differ when cluster size is informative because $F'(x)$ gives equal weight to each possible pair $X_{ij}$ whereas $F(x)$ gives equal weight to each cluster” [1]. Here cluster size can be informative when the response depends on the cluster size. In the context of the radiation toxicity data example in Section 4 of Datta and Satten [1], where the bladder toxicity levels of 230 study participants (data from each individual form a cluster) were measured over time during radiation therapy
for prostate cancer, the authors used the marginal distribution (3.2) to give equal weights of $1/M$ to each cluster-individuals. They concluded that the individuals should have the same weight despite how many radiation sessions they have. This implies each difference pair in a large cluster will receive a smaller weight than each pair in a smaller cluster. In comparison with several SRTs, Datta and Satten [1] showed that their method gave more adequate power in the situations in which the distribution of pairwise differences within a cluster depended on the cluster size.

### 3.2.2.2 SRT accounting for dependence

The conventional Wilcoxon SRT is a nonparametric test for paired data based on independent units of analysis. The miRNA of the paired tumor-normal CRC data tend to have bimodal expression distributions and imputation-induced dependence among subjects. In this section, we modified the SRT for clustered data from Datta and Satten [1] to incorporate the imputation-induced correlation among subjects by replacing the estimated variances among the ranks with the elements of the variance-covariance matrix for each miRNA.

In this respect, we constructed clusters on the paired differences of tumor-normal miRNA features using information of $K$ nearest neighbors, which was provided by the
KNN dependent method (see Chapter 2). Compared to the paired differences in toxicity levels of disjoint participant-clusters in the radiation toxicity data in Datta and Satten [1], the paired tumor-normal CRC data have joint clusters with non-informative (i.e., when the response does not depend on the cluster size) sizes, and each subject may be a member of more than one cluster. The clusters are constructed for each non-missing normal sample with their corresponding imputed samples. For example, suppose the miRNA expressions of the non-missing normal sample of subject 1 is used in the imputation of the miRNA expressions of the missing normal samples of subjects 7, 8, and 9, while the miRNA expressions of the non-missing normal sample of subject 2 is used to impute the miRNA expressions of the missing normal samples of subjects 6, 7, 8, and 10. Then cluster 1 has as sub-units subjects 1, 7, 8, and 9, while cluster 2 contains subjects 2, 6, 7, 8, and 10. Subjects 7 and 8 are members of two cluster groups (see Figure 3.1). The larger the number of nearest neighbors is used \(K\), the larger number of subjects will be joint members of multiple clusters. The cluster sizes can differ; they can include from one (i.e., subjects that are not among the \(K\) nearest neighbors for any other subject) up to all missing-normal subjects. Datta and Satten [1] suggest to consider two approaches, which are giving equal weight to each cluster (Equation (3.2)) and giving equal weight to each possible pair \(X_{ij}\) (Equation (3.3)), in the analysis of disjoint clusters. However, the application of these weight approaches is not practical to the joint clustered miRNA data because the same subject may be a member of more than one cluster and may be counted multiple times. This will increase the total number of cluster sizes \(N\) in Equation (3.3), and consequently, will decrease the absolute value of the test statistic (see Equations (3.4)–(3.7)) and lose power to detect the differentially expressed miRNAs. Therefore, we gave the same weight to each cluster despite the size of cluster, i.e., we divided the rank of each cluster by the total number of clusters.

Another major difference of the CRC imputation-based clusters from the radiation toxicity data clusters in Datta and Satten [1] is that the CRC clusters are not completely disjoint (some subjects may be in several clusters), and consequently there is some dependence across clusters. To address these issues, we included the weighted variances and
covariances of subjects in more than one cluster in the calculation of variances of cluster contributions to the SRT statistic (see Equation (3.9)).

For each miRNA, the distribution function \( F'(x) \), which is given by (3.3), gives equal weight to each possible pair \( X_{ij} \):

\[
F'(x) = \mathbb{E}\left\{ \frac{1}{N} \sum_{i=1}^{M} \sum_{j=1}^{n_i} I(X_{ij} \leq x) \right\}.
\] (3.4)

Here \( X_{ij} \) is a pair difference (tumor − normal) for the \( j \)th subject in the \( i \)th cluster, and \( n_i \) is the \( i \)th cluster size. While Datta and Satten [1] assume the clusters are disjoint and their \( N = \sum_i n_i \) with \( i = 1, ..., M \), we consider joint clusters where subjects can be members of one or more clusters, and our \( N \) is the total number of clusters (\( M \)).

We tested the following marginal hypotheses as in Datta and Satten [1]:

\( H_0: \ F' \) is symmetric (around 0)

\( H_1: \ F' \) is not symmetric.

The marginal SRT statistic is given as Equation 2 in Datta and Satten [1]:

\[
T_M = \mathbb{E}^* \left( \sum_{i=1}^{M} R_i^{*+} \text{sign}(X_i^*) \right)
\] (3.5)

where \( X_i^* \), conditionally given the clustered data \( X \), is generated from \( \hat{F}_i \), which is the empirical distribution of \( \{X_{i1}, \ldots, X_{in_i}\}; \ 1 \leq i \leq M \). More details on \( X_i^* \) are given in Datta and Satten [1]. They assume that \( \mathbb{E}^* \) is the conditional expectation on the \( X^* \) data given the original data \( X = \{X_{ij}: 1 \leq j \leq n_i^*; \ 1 \leq i \leq M\} \). “\( R_i^{*+} \) is the rank of \( |X_i^*| \) amongst \( |X_1^*|, \ldots, |X_M^*| \), and sign is the standard signum function defined as \( \text{sign}(x) = I(x > 0) - I(x < 0) \).” Compared to \( n_i \), which is the size of cluster \( i \), \( n_i^* \) represents the number of sub-unit subjects in the \( i \)th cluster.

We also modified the equation of rank of Datta and Satten [1], which handles ties, according to the marginal distribution (3.4) and the characteristics of CRC data clusters.
That is,
\[ \hat{R}_i(x) = \frac{1}{2N} \sum_{i' \neq i} M \left\{ \sum_{j' \neq j} n_{i'}^{+} I(|X_{i'j'}| \leq x) + \sum_{j' \neq j} n_{i'}^{-} I(|X_{i'j'}| < x) \right\}. \]  
(3.6)

Notation \( j' \neq j \) in (3.6) avoids the comparison of the same subject who is a member of more than one cluster in calculating a rank of a particular subject. For example, assume one wants to find a rank for the value of subject 7 in cluster 1 (say \( X_{1,7} \)) in the previously mentioned cluster example at the beginning of Section 3.2.2.2 (see Figure 3.1). One will only compare \( X_{1,7} \) to the values of \( X_{2,2}, X_{2,6}, X_{2,8}, \) and \( X_{2,10} \) in cluster 2, not to \( X_{2,7} \) itself.

After replacing \( R_i^{++} \) in (3.5) with \( \hat{R}_i \) from (3.6), we obtain:
\[ T_M = \sum_{i=1}^{M} \frac{n_i^+ - n_i^-}{n_i} + \sum_{i=1}^{M} \frac{1}{n_i} \sum_{j=1}^{n_i} \text{sign}(X_{ij}) \hat{R}_i(|X_{ij}|) \]  
(3.7)

where
\[ n_i^+ = \sum_{j=1}^{n_i} I(X_{ij} > 0) \quad \text{and} \quad n_i^- = \sum_{j=1}^{n_i} I(X_{ij} < 0). \]

Note that other than the \( j' \neq j \) notation in (3.6), this \( T_M \) is equivalent to the \( T_M \) in Datta and Satten [1].

Note also that \( T_M \) is of the form:
\[ T_M = \sum_{i=1}^{M} \lambda_i. \]  
(3.8)

Datta and Satten [1] assume the independence of clusters, and so can consider the variance of \( T_M \) as \( \sum_{i=1}^{M} \text{Var}(\lambda_i) \), or the sum (across all clusters \( i = 1, \ldots, M \)) of the variance of each cluster’s rank contribution (\( \lambda_i \)) to the score \( T_M \). However, the potential for overlapping (and thus non-independent) clusters requires a different approach. We consider replacing the estimated variances among the clusters’ rank contributions with the weighted elements of the miRNA’s variance/covariance matrix \( \Sigma \), with weights reflecting
cluster membership. That is, for each cluster $i$ we define:

$$
\hat{S}^2_i = \sum_{j', j \in C_i} \omega_{j', j}^{(i)} \text{Cov}(X_{j'}, X_j)
$$

(3.9)

where $X_j$ is the (tumor − normal) value for subject $j$, $C_i$ is the set of subjects in cluster $i$ (i.e., those subjects whose observed or imputed values involve subject $i$’s value), $n_i = |C_i|$, and $\omega_{j', j}^{(i)}$ is the weight of the covariance of subjects $j'$ and $j$ in cluster $i$. For $j' = j$, $\text{Cov}(X_{j'}, X_j) = \sigma_j^2$, then $\omega_{j', j}^{(i)} = \omega_{j}^{(i)}$ which is the weight of the variance of subject $j$ in cluster $i$.

To ensure the interpretation of $\omega_{j'}^{(i)}$, as reflecting the relative membership of subjects $j$ and $j'$ in cluster $i$, the weights of variance and covariance of subjects $j$ and $j'$ must satisfy the following:

$$
\sum_{i=1}^{M} \omega_{j', j}^{(i)} = 1 \quad (\text{for all pairs of } j, j').
$$

(3.10)

The $\omega_{j}^{(i)}$ is calculated using the weights of the subjects in the imputation of the expression levels of miRNA of missing normal samples. As Equation (2.2) of Section 2.2.2.3, we imputed the expression levels of miRNA features in missing normal sample of subject $j'$ as follows:

$$
\hat{x}_{j'} = a_{j'(1)} x_{j'(1)} + a_{j'(2)} x_{j'(2)} + \cdots + a_{j'(k)} x_{j'(k)} \quad \text{where } i = j'(i) \in K_{j'} = \{j'_1, \ldots, j'_k\}
$$

(3.11)

where $x_{j'(i)}$ ($i = 1, \ldots, k$) is the observed expression level of the miRNA in the observed normal sample of subject $j'(i)$, and $a_{j'(i)}$ is the weight of the subject in the imputation.

Using the weights of the subject in (3.11), we can calculate the weights of variances of imputed subject $j'$ in cluster $i$:

$$
\omega_{j'}^{(i)} = \frac{|a_{j'(i)}|}{\sum_{i' \in K_{j'}} |a_{j'(i')}|}
$$

(3.12)

where $K_{j'}$ is a set of nearest neighbor subjects of $\hat{x}_{j'}$, as in (3.11) above.
For fully-observed (non-imputed) subject \( j' \), the weights of variances are defined as \( \omega_{j'}^{(i)} \equiv 1 \).

The weights of variances \( \omega_{j'}^{(i)} \), defined in (3.12), are used in calculation of the weights of covariances of subjects \( j \) and \( j' \) in cluster \( i \) as follows:

\[
\omega_{jj'}^{(i)} = \frac{\omega_j^{(i)} + \omega_{j'}^{(i)}}{\sum_{i'=1}^{M} \left( \omega_j^{(i')} + \omega_{j'}^{(i')} \right)}.
\]  

(3.13)

As in Equation 3 in Datta and Satten [1], the standardized test statistic (with assumed \( \mathcal{N}(0, 1) \) sampling distribution for demonstration purposes) is computed as follows using \( T_M \) from (3.7) and \( \hat{S}_i^2 \) from (3.9):

\[
Z_M = \frac{T_M}{\sqrt{\sum_{i=1}^{M} \hat{S}_i^2}}.
\]  

(3.14)

### 3.2.3 Wilcoxon SRT after whitening transformation of data

In this section, we used a whitening transformation of the data first, and then the Wilcoxon SRT to identify the differentially expressed miRNA features in the paired tumor-normal data. Krizhevsky and Hinton [47] defined the whitening transformation as a decorrelation method that transforms a set of random variables having the covariance matrix \( \Sigma \) into a set of new random variables whose covariance is the identity matrix.

The whitening transformation is often used in neural networks and data preprocessing applications as it flattens noise and amplifies the signal [62]. There are a few works on applying this transformation to microarray data. Galbraith et al. [62] used the whitening transformation to increase the convergence of network component analysis (NCA) by preprocessing the microarray data. The authors defined the NCA as a method to deduce transcription factor (TF) activities and TF-gene regulation control strengths from gene expression data and a TF-gene binding connectivity network. Rueda and Qin [63] used the whitening transformation in a microarray image segmentation that identifies the spots and separates the foreground from the background.
The whitening transformation of data consists of two steps: (1) decorrelating step that transforms data to have a diagonal covariance matrix, and (2) whitening step that transforms data to have an identity covariance matrix. The following algorithm, which is modified for the paired expressions of the tumor-normal miRNA data, is based on the whitening algorithms discussed in [47, 64–66].

For each miRNA $j$:

1. Make a vector of the differences between each subjects’ tumor and normal samples’ expressions into a zero-mean vector $x$ (by subtracting its mean).

2. Obtain a covariance matrix $\sim$. This $N \times N$ covariance matrix, where $N$ is the total number of sample-subjects, is obtained by:

$$
\sim = \sim_{normal,j} + \sim_{tumor,j}
$$

(3.15)

where the covariance matrix of miRNA $j$ from normal samples, a $N \times N$ matrix $\sim_{normal,j}$, represents the imputation-induced dependence of missing normal samples. The $N \times N$ covariance matrix $\sim_{tumor,j}$ is a diagonal matrix of variance of miRNA $j$ from tumor samples. The $\sim$ matrix is a positive semi-definite matrix. See Section 2.2.2.4 for the construction of $\sim_{normal,j}$ and $\sim_{tumor,j}$.

3. Perform decorrelation of data:

First, one needs to find the eigenvalues and associated eigenvectors of $\sim$ by using the Eigenvalue Decomposition procedure in (3.16):

$$
\sim = \sim_{\Phi} = \sim \sim \sim = \sim \sim \sim = \sim
$$

(3.16)

where $\sim \sim = \sim$ is a diagonal $N \times N$ matrix with eigenvalues and $\sim \sim$ is a $N \times N$ matrix of eigenvectors.

From (3.16), one can obtain:

$$
\sim_{\sim} = \sim_{\sim} \sim_{\sim} \sim = \sim
$$

(3.17)
Because of orthonormality of columns of $\Phi$, $\Phi^{-1} = \Phi^T$. Thus (3.17) is written as:

$$\Phi^T \Sigma \Phi = \Lambda.$$  

(3.18)

Assume one needs to transform the vector $x$ into a new vector $y$ as:

$$y = Z_{\sim_D} x,$$  

(3.19)

where the vector $y$ is uncorrelated and has a diagonal covariance matrix $\Lambda$ so that:

$$\Lambda = \text{Cov}(y) = \mathbb{E}[yy^T].$$  

(3.20)

Substituting equations (3.18) and (3.19) into (3.20), and assuming $\Sigma = \mathbb{E}[xx^T]$:

$$\Lambda = Z_{\sim_D} x \left( Z_{\sim_D} x \right)^T$$

$$\Phi^T \Sigma \Phi = Z_{\sim_D} \Sigma Z_{\sim_D}^T$$

$$\Sigma^{-1} \Phi^T \Sigma \Phi = \Sigma^{-1} Z_{\sim_D} \Sigma Z_{\sim_D}^T$$

$$\Phi^T \Phi = Z_{\sim_D} Z_{\sim_D}^T.$$

(3.21)

Thus, $Z_{\sim_D} = \Phi^T$.

Then, the vector $y$ can be decorrelated as:

$$y = \Phi^T x.$$  

(3.22)

4. Perform whitening of data:

First, one needs to determine that

$$\Lambda^{-1} \Lambda = I.$$  

(3.23)
where $I$ is a $N \times N$ identity matrix.

Substituting (3.18) into (3.23), one can write:

$$\Lambda \sim^{-1/2} \Lambda \sim^{1/2} = I \sim$$  

(3.24)

$$\Lambda \sim^{-1/2} \Phi^T \Sigma \Phi \Lambda \sim^{-1/2} = I \sim.$$  

Assume one needs to have

$$w = Z \sim_W x$$  

(3.25)

where $Z \sim_W$ is the desired whitening transformation that allows the covariance of $w$ to be the identity matrix.

Similarly as before,

$$\mathbb{E}[ww^T] = I \sim$$  

(3.26)

Substituting (3.24) and (3.25) into (3.26):

$$Z \sim_W x \left( Z \sim_D x \right)^T = I \sim$$  

$$Z \sim_W \Sigma Z \sim_W^T = \Lambda \sim^{-1/2} \Phi^T \Sigma \Phi \Lambda \sim^{-1/2} \sim \sim \sim \sim$$  

(3.27)

$$\Sigma \sim^{-1} Z \sim_W \Sigma Z \sim_W^T = \Sigma \sim^{-1} \Lambda \sim^{-1/2} \Phi^T \Sigma \Phi \Lambda \sim^{-1/2} \sim \sim \sim \sim$$  

Thus, $Z \sim_W = \Lambda \sim^{-1/2} \Phi^T \sim \sim \sim \sim \sim$.

Then, the vector $w$ can be written as:

$$w = \Lambda \sim^{-1/2} \Phi^T x = \Lambda \sim^{-1/2} y.$$  

(3.28)

5. Add the mean of the initial differences of paired tumor-normal samples’ expressions to the whitened vector $w$.

We applied the above algorithm of whitening transformation to the imputed CRC
Fig. 3.2. Distributions of quantile normalized expressions of hsa-miR-99b-5p miRNA from normal and tumor samples

data to remove the correlation structure of data and then performed the Wilcoxon SRT to identify the differentially expressed miRNAs.

3.3 Results

In this section, we evaluated the performances of three proposed nonparametric methods (Permutation dependent, SRT dependent, and Whitening) compared to other nonparametric methods.

3.3.1 Motivational example

Figure 3.2 provides the distributions of quantile normalized expressions of a representative miRNA from the paired tumor-normal CRC data. This miRNA is randomly selected from the CRC case study. Figure 3.2 represents the non-normality of the distribution of the paired tumor-normal samples’ differences that motivated us to develop nonparametric methods to address the bimodality of data along with the imputation-induced dependence.

3.3.2 Optimizing the performance of nonparametric permutation t-test

We performed profiling of the R code, which calculates test statistics for paired tumor-normal samples of miRNA data while accounting for the imputation-induced dependence among subjects, by using the R packages lineprof [55] and shiny [56]. For example, the
Table 3.1. Performances of matrix inversion operation by different methods

<table>
<thead>
<tr>
<th>Methods</th>
<th>Replications</th>
<th>Elapsed time</th>
<th>Relative time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 solve()</td>
<td>1000</td>
<td>308.962</td>
<td>20.043</td>
</tr>
<tr>
<td>2 RcppArmadillo</td>
<td>1000</td>
<td>52.224</td>
<td>3.388</td>
</tr>
<tr>
<td>3 RcppArmadillo + optimization</td>
<td>1000</td>
<td>15.415</td>
<td>1.000</td>
</tr>
</tbody>
</table>

Profiling analysis showed that the most time was spent on the `solve()` function from the R package `base` [8]. The `solve()` performs inverse of $X$, where $X$ is a square matrix. Table 3.1 shows the performances of R code in calculating test statistics for paired tumor-normal samples using the `solve()` function, the written function from the `RcppArmadillo` package that returns the inverse of square matrix $X$, and the function from the `RcppArmadillo` package with optimization techniques (avoid copies, code organization, code vectorization, and “do as little as possible” techniques) from Wickham [54]. The simple replacement of `solve()` by the function from the `RcppArmadillo` package improved the performance by almost six times for 1,000 replications. If we used the optimization techniques along with `RcppArmadillo`, we would decrease computational time by about 20 times.

Table 3.1 demonstrates the advantage of using the `RcppArmadillo` package with the code optimization techniques based only on the matrix inversion operation. The optimized whole R code with `RcppArmadillo` decreases the computational time of nonparametric permutation t-test analysis by more than 100 times, i.e., it takes only 6.2 days to finish 50,000 permutations instead of 625 days. The permutation t-test analysis task can be performed even faster by using the parallelization technique from Matloff [67] and McCallum and Weston [68]. It uses multiple cores to work simultaneously on different parts of the permutation analysis. Though it doesn’t reduce the computational time, it saves the time of completing analysis.

3.3.3 Simulation data sets

We carried out the nonparametric methods for differential expression analyses on simulated bimodal paired tumor-normal data matrices. The matrices have $G = 2000$ miRNA expression features (rows) for each of the normal and tumor samples with sample sizes
of $N = 200$ and $400$ subjects (columns). We simulated expression levels of miRNAs for normal and tumor samples by controlling true differentially expressed miRNAs in tumor samples across all simulations. The simulated bimodal miRNA data sets were generated by the mixture of two normal distributions, based on characteristics of the CRC miRNA data. Particularly, all miRNA features of normal samples and only non-differentially expressed miRNA features of tumor samples were simulated based on $\mu = 0.75$ and $\sigma = 0.025$ for the first distribution and $\mu = 4.0$ and $\sigma = 0.5$ for the second distribution, while the differentially expressed miRNA features of tumor samples, which consisted of $20\%$ of all miRNA features of tumor samples, were simulated based on $\mu = 0.75$ and $\sigma = 0.025$ for the first distribution and $\mu = 3.25$ or $\mu = 4.75$ and $\sigma = 0.5$ for the second distribution.

We also generated the demographic and lifestyle data based on characteristics of the CRC case study. As in Section 2.3.2, the covariates were simulated based on characteristics of five continuous (age, number of cigarettes/day, calories, BMI, and lutein and zeaxanthin concentration) and five binary variables (gender, recent aspirin/NSAID use, recent smoker, menopause, and post menopause taking HRT within 2 years statuses) from the CRC study. These variables are simulated by the same approaches that are mentioned in Section 2.3.2. Then, we randomly applied missingness of 10, 30, and 50 percent to the normal data rows first, then imputed them with the KNN dependent method. Thus, the KNN imputation provided the covariance matrices for each miRNA feature. Finally, we applied the nonparametric methods to the simulated data sets to compare the methods’ statistical power in identifying differentially expressed miRNA.

We performed 10 simulations for each sample size with different percent missingness, and 50,000 permutations of the tumor/normal labels of the paired tumor-normal samples for the nonparametric permutation analyses.

### 3.3.4 Differential expression testing

We applied the nonparametric tests to the imputed data sets to check how well we could identify the differentially expressed miRNA features. We obtained a test statistic and a p-value for each miRNA in each paired tumor-normal sample data set while controlling
the false discovery rate (FDR) at 0.05 within each simulation. Then, we calculated the true positive rate (TPR), the false positive rate (FPR), and the false discovery rate (FDR) based on the miRNAs which were controlled as truly differentially expressed in the simulations. The TPR and FPR were defined and calculated as in Bolstad [40] and Stevens et al. [41], and the FDR was defined as in Benjamini and Hochberg [42].

Figure 3.3 demonstrates the performance (including power and FDR control) of the following nonparametric methods:

- Permutation full: permutation test on the full (no missingness imposed) data set
- Permutation independent: permutation test on the imputed data, but assuming independence
- Permutation case deletion: permutation test on the data set with no imputation
• Permutation dependent: permutation test on the imputed data while accounting for
dependence (Section 3.2.1.2)

• SRT full: SRT on the full (no missingness imposed) data set

• SRT independent: SRT on the imputed data, but assuming independence

• SRT case deletion: SRT on the data set with no imputation

• SRT dependent: SRT on the imputed data while accounting for dependence (Section
  3.2.2.2)

• Datta-Satten: SRT based on the method of Datta and Satten [1]

• SRT average: SRT based on the averages of (tumor − normal) values in each cluster.
The clusters are as constructed in the SRT dependent analysis.

• Whitening: SRT after whitening transformation of data (Section 3.2.3)

The scatter plots of TPR and FPR for the similar conditions are given in Figure A.2 in Appendix A.2.

The nonparametric methods are displayed by different symbols and colors in Figure 3.3. The permutation methods (Permutation full, Permutation independent, Permutation case
deletion, and Permutation dependent) are represented in red, while the SRT test methods
(SRT full, SRT independent, SRT case deletion, and SRT dependent) are in blue. As the
performances of the Datta-Satten, the SRT average, and the Whitening methods are similar,
the same gray color is used for them. Squares are used for methods on the full data set,
diamonds for methods assuming independence of imputed data, pluses for methods on the
data sets with no imputation, and circles for methods accounting for dependence of imputed
data. The red dotted lines in the scatter plots represent the FDR threshold of 0.05.

The power (i.e., the TPR values) for the Datta - Satten, the SRT average, and the
Whitening methods generally increases with larger sample sizes and missing percentages,
but they fail to control the FDR values anywhere near the threshold of 0.05. The other
methods generally control the FDR values at about 0.05 with moderate powers. Figure 3.4
Fig. 3.4. TPR and FDR of some of SRT tests and permutation methods for sample sizes of 200 and 400 and missing percentages of 10, 30, and 50

shows the performance of these other nonparametric methods (dropping Datta - Satten, SRT average, and Whitening) on the simulated data sets for the similar conditions as displayed in Figure 3.3. The scatter plots of TPR and FPR for the reduced list of methods can be found in Figure A.3 in Appendix A.2.

Figure 3.4 shows that the overall power increases with larger sample sizes. Particularly, Permutation independent and SRT independent have higher power than the Permutation dependent, the Permutation case deletion, the SRT dependent, and the SRT case deletion methods for all number of subjects and missing percentages, but they lack control of the FDR for 400 subjects and 50% missing normal samples, which are the characteristics of the CRC case study. Permutation dependent and SRT dependent have moderate power (the TPR values are in the range of 0.825-1.00) and control relatively well the FDR near 0.05 for 400 subjects and 50% missing. At 50% missing, Permutation case deletion and SRT case
Fig. 3.5. TPR and FDR of SRT dependent for sample size of 400 and missing percentages of 10–90

deletion have the lowest power to identify the differentially expressed miRNA features, but
they generally maintain control of the FDR, comparable to the full data analyses.

Although the Permutation dependent method keeps moderately high power for all
number of subjects and missing percentages, the power for SRT dependent increases with
larger missing percentages in Figure 3.4. This happens because higher missing percentages
result in larger clusters with higher numbers of joint members, i.e., the same subject with
non-missing normal sample can be used in the imputation of several closely located (by de-
mographic and lifestyle characteristics) subjects’ normal samples. Numerically, this makes
the marginal SRT statistic ($T_M$) increase faster than the denominator in Equation (3.14).
Figure 3.5 demonstrates how the performance of SRT dependent changes with increasing missing percentages for sample size of 400. Here we performed 20 simulations for sample size of 400 and missing percentages of 10–90 as described in Section 3.3.3. The TPR values increase with increasing missing percentages from 10% to 60% and start to decrease for missing percentages of 70, 80, and 90. FDR control is also worse for higher missing percentages, but for 400 subjects and 50% missing normal samples, SRT dependent has moderately high power. The power drops sharply for missing percentage of 90. Figure A.4 of TPR and FDR of the SRT dependent method for sample size of 400 and missing percentages of 10–90 can be found in Appendix A.2. In contrast to Permutation dependent, SRT dependent runs the differential expression testing with significantly less computational time. It takes about 1.5 hours on a machine with CPU speed of 1.86 GHz and 2 GB RAM to perform the differential expression testing on the data set with 400 subjects and 2000 miRNAs, compared to 6.2 days for Permutation dependent.

3.3.5 Real data application

For demonstration purposes, we applied the permutation tests (including Permutation full, Permutation independent, Permutation case deletion, and Permutation dependent) and the SRTs (including SRT full, SRT independent, SRT case deletion, and SRT dependent) to an imputed data set to identify the differentially expressed miRNAs in real miRNA data from the CRC case study.

The miRNA data consist of 2006 miRNAs from the paired tumor-normal samples of the first available 527 subjects. Since the distributions of many miRNAs are either unexpressed (933 miRNAs out of 2006 are unexpressed in either normal or tumor samples of subjects) or partially expressed (for example, only 703 miRNAs out of 2006 are expressed in at least 50% of either normal or tumor samples of subjects), we applied the filtering method to remove the unexpressed miRNAs in the differential expression testing. This approach helps to reduce the number of tests and to increase the power to detect true differences among subjects [69].

First, we randomly applied 50% missingness to the normal samples of the CRC study
subjects. Then, we kept the miRNAs that were expressed on at least 10% of arrays in either normal or tumor samples of fully-observed subjects after randomly applying 50% normal sample missingness (832 miRNAs were selected). We imputed the values of the missing normal samples using KNN dependent. Here we used all available 19 noncollinear demographic and lifestyle variables: 7 continuous (age, number of cigarettes/day, calories, BMI, lutein and zeaxanthin concentration, vitamin D (mcg), and lycopene (mcg)) and 12 binary variables (gender, recent aspirin/NSAID use, recent smoker, menopause, post menopause taking HRT within 2 years statuses, data collection centers, race, smoking statuses, long-term alcohol consumption, SEER summary staging, AJCC stage groupings, and study types) from the CRC case study.

Figure 3.6 displays a scatterplot matrix of the FDR adjusted p-values from the permutation tests in double negative log-scale. These results are based on 50,000 permutations of normal/tumor labels of the paired tumor-normal samples within subjects. As previously mentioned, the lower-left rectangles produced by the reference lines represent the significant miRNAs from both permutation analyses. From Figure 3.6 it is not obvious to tell which one of the permutation methods (Permutation independent, Permutation case deletion, and Permutation dependent) can better predict the miRNAs that are found significant in the Permutation full analysis. The pairwise scatter plots of all permutation methods have similar patterns in Figure 3.6. Figure 3.6 shows that Permutation dependent has better control of the Type I error rate (i.e., there are only a few observations in the upper-left rectangle of the upper triangle scatter plot between the Permutation full and the Permutation dependent methods) than the other permutation methods and it may have moderate power to detect the differentially expressed miRNAs (i.e., there are many observations in the lower-right rectangle). The computational time of SRT dependent analysis of the real data was about 2 hours, while the 50,000 permutations of normal/tumor labels of the paired tumor-normal samples within subjects for the Permutation dependent analysis took about 3.5 days by using the parallelization technique with 10 machines which worked simultaneously on different parts of the permutation analysis (see Section 3.3.2).
Figure 3.7 shows a scatterplot matrix of the FDR adjusted p-values of the SRTs in double negative log-scale ($-\log(-\log(\text{adj.p}))$). The red dashed vertical and horizontal lines in the scatter plots represent the threshold of 0.05 in double negative log-scale. The lower-left rectangles produced by the reference lines are the miRNAs that are found significant in the horizontal and vertical SRT analyses. For this particular set of missing normal samples, SRT case deletion gives the best concordance with the SRT full method. The next best agreement with the results of SRT full is the results from the SRT independent method.
Fig. 3.7. Scatterplot matrix of adjusted p-values of SRT full, SRT independent, SRT case deletion, and SRT dependent (in double negative log-scale)

The SRT dependent method gives the worst agreement with the SRT full analysis. SRT dependent has better agreement with the SRT full analysis for the fully-expressed miRNAs (see Figure A.5 in Appendix A.2) than for the partially expressed miRNAs (see Figure 3.7).

The advantages of Permutation dependent and SRT dependent over Permutation independent, Permutation case deletion, SRT independent, and SRT case deletion are not clear on the real data (see Figures 3.6 and 3.7). One of the reasons of poor performances of Permutation dependent and SRT dependent might be still the presence of many partially expressed miRNAs. Figure 3.8 shows two representative miRNAs: a red circle represents a
**Fig. 3.8.** Scatter plot of adjusted p-values of SRT full and SRT dependent (in double negative log-scale)

**Fig. 3.9.** Distributions of fully-observed and imputed expressions of “hsa-miR-4440” (a “false positive” in Figure 3.8) from normal and tumor samples
miRNA which is not significant with the SRT full test, but is found significant in the SRT dependent analysis; a blue rectangle represents a miRNA which is not significant with the SRT dependent test, but is called significant in the SRT full analysis. The distributions of the original and the imputed expressions of missing normal samples and the fully-observed expressions of tumor samples, as well as the paired tumor-normal differences of these representative miRNAs are given in Figures 3.9 and 3.10.

The peaks at negative one in the histograms of Figure 3.9 represent the unexpressed expressions of miRNAs. Since the distributions of the fully-observed (original) miRNA expressions from the paired tumor-normal samples are very similar, and consequently the distribution of their paired differences looks like a mixture of two normal distributions with a peak near zero in Figure 3.9, the SRT full analysis called this miRNA non-significant. But the distribution of the imputed expressions of missing normal sample differs from the distribution of expressions of tumor sample (the first two histograms from the left in Figure

Fig. 3.10. Distributions of fully-observed and imputed expressions of “hsa-miR-1470” (a “false negative” in Figure 3.8) from normal and tumor samples
Fig. 3.11. Scatter plots of the partially expressed and fully-expressed original and imputed miRNAs of normal samples from randomly selected subjects.

Consequently, SRT dependent called it a differentially expressed miRNA. The peak in the histogram of the fully-observed normal sample results in the right skewed distribution for the imputed normal sample (see Figure 3.9). The poor imputed values for unexpressed miRNAs can lead to very different tumor-normal values (as in Figure 3.9) that can result in “false positives” (as the circled point in Figure 3.8). The poor (higher) imputation of unexpressed miRNAs (such as in Figure 3.9) is shown as line-shaped clusters at ones of the horizontal axes in Figure 3.11.

In Figure 3.10 the tumor-normal differences in the original data tend to be slightly negative due to the higher expressions in normal samples. As a result, the SRT full test
called this miRNA significant. However, the imputed expressions in normal samples tend
to be lower than original, so the tumor-normal differences in the imputed data are more
symmetric about zero (even with fewer non-expressed normal samples), resulting in non-
significance for the imputed data (and resulting in “false negatives” as the rectangle point
in Figure 3.8). The poor (lower) imputation of moderately low expressed miRNAs is shown
(such as in Figure 3.10) as the spread of expression values near fives of the horizontal axes
in Figure 3.11.

Figure 3.11 shows the scatter plots of the partially expressed (expressed in at least 10% of
subjects) and fully-expressed (in at least 100% of subjects) original and imputed miRNAs
of normal samples from randomly selected subjects. These distributions are shifted by plus
2 to remove negative ones for logarithmic axes transformation. The scatter plots of the
partially expressed miRNAs in Figure 3.11 create line-shaped clusters of expressions at ones
(corresponding to unexpressed) of the horizontal axes and show partial agreement between
the partially expressed original and imputed normal samples. The scatter plots of the
fully-expressed miRNAs show general agreement between the fully-expressed original and
imputed normal samples. Figure 3.11, which is provided for only demonstration purposes,
shows that the KNN imputation method is fairly good for higher-expressed miRNAs, but
worse for lower-expressed miRNAs, and highly variable for unexpressed miRNAs.

3.4 Discussion

In this chapter, we proposed three nonparametric methods to identify the differen-
tially expressed miRNAs in the paired tumor-normal data while accounting for imputation-
induced dependence. These methods are Permutation dependent, SRT dependent, and
Whitening.

For 400 subjects and 50% missing normal samples, which are the characteristics of
the CRC case study, the Permutation independent and the SRT independent methods
showed higher power than the other nonparametric methods, but fail to control the FDR in
simulation. The Permutation dependent and the SRT dependent methods have moderate
power and generally maintain control of the FDR. Permutation case deletion and SRT case
deletion have the lowest power to identify the differentially expressed miRNA features, but they control relatively well the FDR.

The Datta - Satten, the SRT average, and the Whitening methods are generally powerful in identifying the differentially expressed miRNA but fail to control the FDR for all number of subjects and missing percentages.

For the differential expression analysis of the real paired tumor-normal miRNA data from the CRC case study, the SRT dependent and the Permutation dependent tests don’t provide clear advantages over Permutation independent, Permutation case deletion, SRT independent, and SRT case deletion in reproducing the results of the SRT full and the Permutation full analyses, respectively. One of the reasons why the proposed nonparametric methods didn’t perform as well as in the simulation data is still the presence of many miRNAs that are partially or rarely expressed in the real miRNA data. The unexpressed miRNAs caused the imputed expressions to differ significantly from the original expressions.

Depending on the research objectives, researchers could use the Datta - Satten, the SRT average, and the Whitening methods in testing for differential expression that give higher power for all sample sizes and missing percentages. But these methods fail to control the FDR. Permutation dependent and SRT dependent have moderate power, but control the FDR. Compared to SRT dependent, Permutation dependent method gives more power for missing percentages of 10 and 30, but it is more computationally expensive (6.2 days versus 1.5 hours for 400 subjects and 2000 miRNAs).

R code for the application of nonparametric methods and the differential expression tests can be found in Appendix B.2.
CHAPTER 4
INCORPORATION OF DEMOGRAPHIC AND LIFESTYLE VARIABLES IN QUANTILE NORMALIZATION OF MIRNA DATA

4.1 Introduction

A critical data analysis step in detecting differentially expressed microRNA (miRNA) features is normalization. The normalization procedure reduces technical variation and maintains true biological changes between arrays. Various normalization techniques exist, but until recently, all were developed for messenger RNA (mRNA) arrays. The miRNA data are very different from mRNA data due to the small total number of miRNAs (a few hundred versus 10,000 to 50,000 genes in mRNA data), and the majority of miRNAs are either not expressed or are expressed at very low levels [70]. Therefore, normalization methods used for mRNA expression array may not be appropriate for miRNA arrays.

Researchers modified the normalization methods of mRNA to adapt the characteristics of miRNA data in particular experimental settings [71–75]. However, a universal normalization method has not been developed for miRNA data yet. Meyer et al. [76] recommend to select the optimal normalization method based on the characteristics of the data set, and examine it carefully in specific biological contexts.

The importance of finding an appropriate normalization method for miRNA data in the colorectal cancer (CRC) study motivated us to develop a normalization method that accounts for the characteristics of data, removes any artificial variations, and keeps the crucial biological information. CRC is the third most common type of cancer and the second leading cause of cancer death in the United States [77]. Most colorectal cancers are due to demographic, lifestyle, and health-related factors, with only a small number of
cases due to underlying genetic disorders \cite{78,79}. Cunningham et al. \cite{80} and Watson and Collins \cite{81} listed older age, male gender, high intake of fat, alcohol or red meat, obesity, smoking, and a lack of physical exercise as risk factors of CRC. Taken together with known and hypothesized associations of miRNA with CRC, this suggests that the expression levels of miRNAs in tissues from risk group (for example, old and smoker) patients are likely to be differentially expressed than the expression levels from non-risk group (young and non-smoker) patients.

The CRC case study researchers have collected extensive information about demographic and lifestyle variables of CRC patients along with the miRNA features from normal and tumor samples. Few CRC studies have collected such extensive data for such variables. This data may be helpful to consider not only the artificial intra- and inter-array differences, but also the differences caused by the demographic and lifestyle characteristics of patients, and to maintain only biological differences during the normalization procedure.

In this chapter, we incorporated the demographic and lifestyle variables in miRNA normalization, which has not been done before. We modified the quantile normalization method from Bolstad et al. \cite{7} that is commonly used in miRNA data analysis and was found as an efficient method to remove the artificial differences across arrays \cite{82,83,84,85}. The quantile normalization equalizes the distributions of expression intensities across samples while ignoring any differences of characteristics of samples. In the quantile normalization method, each subject’s normalized distribution of expression values depends on all other subjects’ distributions equally. Such normalization of miRNA expressions while ignoring the characteristics of data results in loss of important biological information. In our modified normalization method, we assume that the normalized distribution of miRNAs from one subject should depend on the weighted distribution of miRNAs from other subjects. The weights of subjects are determined from the distance matrix generated from various distance metrics of multivariate covariates. The elements of the generated distance matrix represent the pairwise distances between two subjects based on demographic and lifestyle variables. The distances (weights) among subjects are considered in the quantile normalization of miRNA.
This chapter is arranged in the following manner: first, we provide an overview of normalization methods developed for miRNA data and explain our modified normalization method in detail. Then we show the application of normalization techniques using simulation and real data sets. Finally, we conclude with a discussion of the important findings presented in this chapter.

4.2 Methods

4.2.1 Normalization methods for miRNA data

The normalization procedure is one of the important steps of preprocessing of miRNA data\footnote{The other steps of preprocessing are background correction and summarization.}. The procedure removes the systematic differences between arrays that do not represent true biological variation. Meyer et al. [76] state that “normalization, often an underestimated aspect of data processing, can minimize systematic technical or experimental variation and thus has significant impact on the detection of differentially expressed miRNAs.” Bolstad et al. [7] highlights that the need of normalization arises naturally when multiple arrays are involved in experiments. There are two types of variations that can be expected: the first variation is an “interesting” variation, which represents biological differences between the expression levels of genes (or miRNAs) from tumor and normal tissues, and the other is an “obscuring” variation, which is not interesting for the researchers and needs to be removed by a normalization procedure. The currently used normalization methods in miRNA data analysis are developed for mRNA arrays, which have an exceedingly high density of probes with 10,000 to 50,000 genes. In comparison, miRNAs are lower density arrays with a few hundred to a couple of thousand genes. Wang and Xi [70] mentioned that the majority of miRNAs are either not expressed or are expressed at very low levels. Therefore, researchers concluded that normalization methods of mRNA arrays may not be appropriate for miRNA arrays [76] [83] [85].

There are several studies that compared the performances of normalization methods based on mRNA data to see how methods can reduce the experimentally induced varia-
tion and maintain true biological changes between arrays in miRNA data analysis. Rao et al. [83] applied the commonly used normalization methods, including cyclic loess, quantile, median or mean, and no normalization techniques to normalize miRNA expression arrays. Their analyses show that the quantile normalization method works better than other normalization techniques in removing differences across arrays in miRNA expression data. Pradervand et al. [85] also examined the impacts of mRNA array normalization procedures based on scaling, quantile, and variance stabilizing normalization (VSN) on miRNA data. They found that the quantile normalization was the most robust procedure and performed at least as well as the developed normalization method based on the set of invariants (invariants-based) among the mRNA normalization techniques that they considered (including quantile, invariants-based, scaling, VSN, and no normalization methods) over all experimental conditions tested. All normalization methods performed better than no normalization. Over the last decade, the quantile normalization method has been commonly used in miRNA data analysis compared to the other normalization techniques developed for mRNA data [82, 84]. In addition, the miRNA expressions of the CRC case study, which were used in the analyses of Chapters 2 and 3 of this dissertation, were normalized by the quantile normalization option from the R package AgiMicroRna [86]. We explain the algorithm of quantile normalization in Section 4.2.2.

A number of modifications of normalization methods of mRNA have been performed to adapt the characteristics of miRNA data that were profiled by real time quantitative PCR (RT-qPCR) and microarray hybridization approaches. Though each modified normalization method has been shown to perform well based on the characteristics of tested miRNA data [71, 75], a universal normalization method for miRNA data has not been developed yet. Meyer et al. [76] strongly suggest to select the optimal normalization method based on the characteristics of the data set, and examine it carefully in specific biological contexts. The choice of normalization method is one of the primary factors that affects the inference of differential expression [73].

In this chapter, we developed a novel miRNA normalization approach based on the
quantile normalization, incorporating additional covariates. Availability of extensive information about demographic and lifestyle variables of subjects in the CRC case study gives us an opportunity to consider the characteristics of array-samples in the normalization procedure, which has not been done before.

4.2.2 Quantile normalization: modified and incorporating demographic and lifestyle variables

4.2.2.1 Quantile normalization

Bolstad et al. [7] were early researchers who used the quantile normalization in microarray data analysis. They compared its performance with the cyclic loess and contrast based normalization methods that had already been successfully used in mRNA data analysis. The purpose of the quantile normalization is to force the distribution of probe intensities for each array in a set of arrays to have the same or at least similar distribution. Bolstad et al. [7] were motivated by the idea that a quantile-quantile plot demonstrates the same distribution for two data vectors if the plot is a straight diagonal line and not the same distribution if the plot is other than a diagonal line. They extended this concept to \( n \) dimensions of data vectors so that all data vectors have the same distribution. The quantiles of \( n \) data vectors are plotted in such a way that the plot gives a straight line along the line given by the unit vector \((\frac{1}{\sqrt{n}}, \ldots, \frac{1}{\sqrt{n}})\). From this, one can make the distribution of a set of data vectors the same if one projects the points of the \( n \) dimensional quantile onto the diagonal. More details about the projection of the quantiles onto the diagonal are provided in Bolstad et al. [7].

Bolstad et al. [7] provided the following algorithm to perform a quantile normalization:

1. Arrange the logarithmic transformed microarray data into a \( G \times N \) matrix \( X \), where \( G \) and \( N \) are total numbers of genes and arrays, respectively. Sort each column of \( X \) to give \( X_{\sim \text{sort}} \).
2. Then take the means across the rows of \( X_{\sim \text{sort}} \) and assign this mean to each element in the row to get \( X'_{\sim \text{sort}} \).
3. At the end, obtain the normalized version \( X_{\sim \text{norm}} \) of \( X_{\sim} \) by rearranging each column of \( X'_{\sim \text{sort}} \) to have the same ordering as the original \( X \).
Another algorithm to carry out the quantile normalization was introduced by Amaratunga and Cabrera [87]. They described the algorithm as follows: “calculate the percentiles \((Q_{i0},...,Q_{i100})\) of the \(i\)th array and the percentiles \((Q_{M0},...,Q_{M100})\) of the median mock array. For any value \(X_{gi}\), find the interval, \([Q_{ih},Q_{i(h+1)}]\), to which it belongs and obtain its normalized value, \(X'_{gi}\), by linearly interpolating between the pair points \((Q_{Mh},Q_{ih})\) and \((Q_{M(h+1)},Q_{i(h+1)})\)” [87]. In this algorithm, \(X_{gi}\) means the logarithmic transformed spot intensity measurement for gene \(g\) on array \(i\). They define the median mock array as the array fashioned out of the medians of the arrays being normalized.

To date, no research has been conducted to compare performances of the above mentioned quantile algorithms. The algorithm from Bolstad et al. [7] is more widely applied in practice than the algorithm of Amaratunga and Cabrera [87]. In our analysis, we used a `normalize.quantiles` function from the R package `preprocessCore` [88] that is based upon the concept of the quantile normalization from Bolstad et al. [7]. We further refer to the quantile normalization based on the algorithm from Bolstad et al. [7] as the conventional quantile normalization.

### 4.2.2.2 Quantile normalization incorporating demographic and lifestyle variables

The conventional quantile normalization does not account for additional characteristics of samples when it normalizes the miRNA arrays. In this respect, we propose a novel approach that removes the non-biological differences across samples while incorporating the demographic and lifestyle characteristics of sample-subjects in normalization. Here, we assume that the normalized distribution of miRNA expression levels from one sample should depend on the weighted distribution from other samples. The weights of subjects are determined from a distance matrix, which is aggregated from various normalized (values are between 0 and 1) distance matrices (based on Euclidean, Manhattan, Minkowski, and other methods) of multivariate covariates. The elements of the aggregated distance matrix \((\sim D)\) represent the pairwise distances between two subjects based on relevant demographic and lifestyle variables.
The algorithm of our proposed method is based on the quantile normalization algorithm from Bolstad et al. [7], but accounts for the weighted distance metrics of demographic and lifestyle variables. The algorithm is as follows:

1. Obtain the logarithmic (log2-based) transformed miRNA data as a $G \times N$ matrix $X$, where $G$ is the total number of miRNAs and $N$ is the total number of subjects. The log-transformation of miRNA data is performed to reduce the effect of outliers on the calculation of miRNA expression levels.

2. Sort each column-subject of $X$ to give $X_{\sim \text{sort}}$.

3. Obtain an aggregated distance matrix as a $N \times N$ matrix $D$.

4. Obtain the between-subject weight matrix as a $N \times N$ matrix $W$. For example, the weight of subject $i$ in the normalized expression distribution of subject $j$ is calculated as in (4.1):

$$w_{ij} = 1 - \frac{d_{ij}}{\max\{d_{i1}, \ldots, d_{iN}\}}$$  \hspace{1cm} (4.1)

where $d_{ij}$ is a distance between subjects $i$ and $j$ with $i, j = 1, \ldots, N$. If $i = j$, then $w_{ij} = 1$.

Note: the matrix $W$ is not symmetric and the elements of row $i$ correspond to the weights of the $N$ subjects in the normalized expression distribution of subject $i$.

5. Normalize the between-subject weight matrix $W$ by the sum of the weights of row-subjects and obtain a $N \times N$ matrix $W^*$. For example, the weighted means of weights of subjects in the normalized expression distribution of subject $i$ can be found as in (4.2):

$$w^*_{ij} = \frac{w_{ij}}{N \sum_{j=1}^{N} w_{ij}} \hspace{1cm} \text{s.t} \hspace{1cm} \sum_{j=1}^{N} w^*_{ij} = 1.$$  \hspace{1cm} (4.2)
6. Calculate the weighted means across the rows of $X_{\sim sort}^\sim$ and assign this weighted mean to each element in the row to get a $G \times N$ matrix $X'_{\sim sort}^\sim$.

The weighted means of $X_{\sim sort}^\sim$ can be computed as in (4.3):

$$X'_{\sim sort}^\sim = X_{\sim sort}^\sim [W^\star]^T. \quad (4.3)$$

For example, the weighted normalized expression levels for subject $i$ are computed as in (4.4):

$$x'_{sort,i} = \sum_{g=1}^{G} \sum_{j=1}^{N} x_{sort,gj} \cdot w_{ij}^\star. \quad (4.4)$$

7. Obtain the normalized version $X_{\sim norm}^\sim$ of $X$ by rearranging each column of $X'_{\sim sort}^\sim$ to have the same ordering as the original $X^\sim$.

The above mentioned algorithm generalizes the quantile algorithm of Bolstad et al. [7], in which all $w_{ij}^\star = \frac{1}{N}$.

In contrast to the normalization methods for miRNA data where disjoint clusters of miRNAs were considered (Mestdagh et al. [89], Bargaje et al. [90]), this normalization technique will consider both clustered and overlapped subjects by accounting for each subject’s weight in the average. The clustered subjects will have heavier weights in the average than the weights of subjects who are not clustered. This normalization technique will contribute to reduction of intra- and inter-array technical variability while maintaining biological differences. We subsequently refer to this proposed method as the weighted quantile normalization.

4.3 Results

In this section, we demonstrated the performance of the weighted quantile normalization method over the conventional quantile normalization method and the no normalization technique (only log-transformed expressions) using different simulation scenarios.
4.3.1 Motivational example

Figure 4.1 shows a motivational example of the distributions of non-normalized and quantile normalized miRNA expressions of tumor samples from non-risk and risk group patients. The plot compares the distributions of miRNA expressions of a 41-year-old, non-smoker, non-drinker, and normal weight woman (non-risk group patient) against the distributions of expressions of a 76-year-old, current smoker, heavy drinker, and overweight man (risk group patient) using the no normalization technique, which only considers the log-transformed expressions, and the conventional quantile normalization, which equates the quantiles of the log-transformed expressions. The non-normalized miRNA expression histograms, top sub-plots in Figure 4.1, are clearly different from each other: the distribution of the risk-group patient is flatter and more skewed right than the distribution of the non-risk group patient. After performing the quantile normalization of miRNA expressions, both subjects have very similar distributions. If one carried out a Wilcoxon signed-rank test (SRT) to find differentially expressed miRNAs, one would have less chance of finding any differences between these two distributions. This example motivated us to develop a
novel normalization method that removes any technical variations and keeps true biological information while accounting for additional covariates.

### 4.3.2 Simulation data sets

The normalization analyses were carried out on bimodally distributed paired data matrices of $G = 2000$ miRNA expression features (rows) for each of the normal and tumor samples with sample sizes of $N = 200$ and 400 subjects (columns). We simulated expression levels of miRNAs for normal and tumor samples by controlling true differentially expressed miRNAs of tumor samples across all simulations. The simulated bimodal miRNA data sets were generated by the mixture of two normal distributions. Particularly, all miRNA features of normal samples and only non-differentially expressed miRNA features of tumor samples were simulated based on $\mu = 0.75$ and $\sigma = 0.025$ for the first distribution and $\mu = 4.0$ and $\sigma = 0.5$ for the second distribution, while the differentially expressed miRNA features of tumor samples, which consisted of 20% of all miRNA features of tumor samples, were simulated based on $\mu = 0.75$ and $\sigma = 0.025$ for the first distribution and $\mu = 3.25$ or $\mu = 4.75$ and $\sigma = 0.5$ for the second distribution. These parameters were chosen based on characteristics of the bimodal CRC miRNA data. We performed 20 simulations for each sample size.

Moreover, we simulated demographic and lifestyle variables of subjects in such a way that they could reflect the characteristics of the CRC study and also carry some useful information for the normalization procedure. In our analysis, we simulated all available 19 non-collinear demographic and lifestyle variables: 7 continuous (age, number of cigarettes/day, calories, BMI, lutein and zeaxanthin concentration, vitamin D (mcg), and lycopene (mcg)) and 12 binary variables (gender, recent aspirin/NSAID use, recent smoker, menopause, and post menopause taking HRT within 2 years statuses, data collection centers, race, smoking statuses, long-term alcohol consumption, SEER summary staging, AJCC stage groupings, and study types) from the CRC case study. The simulations of multivariate covariates were performed as described in Section 2.3.2.

We started the simulation analyses by generating miRNA expressions of normal and
tumor samples and demographic and lifestyle variables based on the simulation parameters mentioned above. During simulation of miRNA expressions, we controlled arbitrarily 20% of miRNAs from tumor samples as differentially expressed features.

As an aside, the application of the weighted quantile normalization method is not computationally burdensome; it took less than one minute to normalize the expressions of 2000 miRNAs from 400 subjects on a machine with CPU speed of 1.86 GHz and 2 GB RAM.

### 4.3.3 Differential expression testing

We carried out the normalization methods mentioned in Section 4.2. We conducted the differential expression analyses between tumor-normal (using a per-miRNA SRT) on the normalized data sets to check whether the weighted quantile normalization method has an equal statistical power in finding differentially expressed miRNA as other normalization techniques. First, we obtained the Wilcoxon SRT statistic and p-value for each miRNA feature in each normalized data set and controlled the false discovery rate (FDR) at 0.05 within each simulation. Then, we calculated the true positive rate (TPR), the false positive rate (FPR), and the false discovery rate (FDR) based on the miRNAs which were controlled as truly differentially expressed in the simulations. The TPR and FPR were defined and calculated as in Bolstad [40] and Stevens et al. [41], and the FDR was defined as in Benjamini and Hochberg [42].

Figure 4.2 shows the performance (including power and FDR control) of the Wilcoxon SRTs on the data sets normalized by the conventional quantile and the weighted quantile methods, and on the data set without applying any normalization methods for the numbers of subjects of 200 and 400. As shown in this scatter plot, the power (i.e., the TPR values) increases for the conventional quantile (a blue open rectangular symbol) and the weighted quantile (a red solid triangular symbol) normalization methods with larger sample sizes. The TPR values are almost zero for the no normalization technique (a black open circular symbol) for both sample sizes. The no normalization technique also fails to control the FDR values below the threshold of 0.05, which is represented by red dotted lines in the figures.
**Fig. 4.2.** TPR and FDR for sample sizes of 200 and 400

**Fig. 4.3.** TPR and FDR for sample sizes of 200 and 400 (dropping no normalization)
Figure 4.3 presents the scatter plots of TPR and FDR only for the conventional quantile and the weighted quantile normalization approaches. The weighted quantile normalization has clearly higher power than the conventional quantile method. For 400 subjects, which have characteristics based on those of the CRC case study, the differences of TPR values between the two methods are up to 10%: the power for the weighted quantile normalization is in the range of 83.6% – 90.5%, while the power for conventional one is in 77.2% – 85.3%. That is, one can identify correctly more, up to 10% of a total number of features, the differentially expressed miRNA features from the data set normalized by the weighted quantile normalization than from the data set normalized by the conventional quantile method. Both normalization methods generally control the FDR near 0.05 for both sample sizes.

Figure A.6 in Appendix A.3 shows the scatter plots of TPR and FPR for the similar conditions that are represented in Figure 4.2. The FPR values, which refer to the probability of falsely rejecting the null hypothesis for a particular test, are small for all normalization techniques. The scatter plots of TPR and FPR for data sets normalized only by the conventional quantile and the weighted quantile methods are presented in Figure A.7 of Appendix A.3.

### 4.3.4 Normalization accounting for unrelated covariates

The application of the weighted quantile normalization method is only appropriate when demographic and lifestyle variables (on which between-subject distance is based) are relevant to the “treatment” group comparison of interest. Figure 4.4 shows the performance (the TPR versus the FDR) of the Wilcoxon SRT on the simulated data sets that are normalized by the conventional quantile and the weighted quantile methods while accounting for unrelated covariates. The power and the FDR control are approximately the same (overlap in most simulations) for both normalization methods. Thus, the weighted quantile normalization performs at least as well as the conventional quantile normalization when demographic and lifestyle variables are not associated with the treatment group. Figure A.8 in Appendix A.3 shows the scatter plots of TPR and FPR for sample sizes of 200 and
4.4 Real data application

We used the Wilcoxon SRT to identify the differentially expressed miRNAs in the paired tumor-normal miRNA data sets from the CRC case study. The miRNA data sets were normalized by the conventional quantile and by the weighted quantile methods while accounting for the demographic and lifestyle characteristics of CRC subjects. These data sets contain the first available 527 subjects with 2006 miRNA on each sample. In this analysis, we used all available 19 noncollinear demographic and lifestyle variables that are discussed in Section 4.3.2.

Figure 4.5 shows a scatter plot of the FDR adjusted p-values in logarithmic scale. The green circles (there are 121) in the plot represent the miRNAs that are found significant from the quantile normalized data, but found not significant from the weighted quantile
Fig. 4.5. Scatter plot of adjusted p-values of the CRC miRNA data, normalized by the quantile and the weighted quantile normalization methods (in log-scale). The green and red circles represent the miRNAs that are found significant only in the horizontal and vertical analyses, respectively.

normalized data. The red circles (there are 119) show the miRNAs that are found significant only in the weighted quantile normalized data. There is no information about the truly differentially expressed miRNAs that could be helpful to evaluate the performances of both normalization methods. However, we can see from Figure 4.5 that many miRNAs (in the lower right triangle of the plot) that are found significant in the quantile normalized data are found to be even more significant in the data set normalized by the weighted quantile method. The plot shows that the proposed weighted quantile normalization method has more power than the conventional quantile method.
4.5 Discussion

In this chapter, we modified the quantile normalization method from Bolstad et al. [7] to reduce not only the artificial variations across samples, but also the variations caused by the characteristics of data, as well as to maintain true biological differences across samples. The proposed normalization method incorporates demographic and lifestyle variables by considering the distances (weights) among subjects based on their characteristics, and considers these weights in equating the quantiles of distributions of miRNA expressions.

We performed the differential expression tests on the simulated bimodally distributed miRNA expressions and demographic and lifestyle variables that reflect the characteristics of the CRC case study. We applied the Wilcoxon SRT to identify differentially expressed miRNAs from the normalized simulated miRNA data. The scatter plots of TPR versus FDR and FPR help to evaluate the impact of normalization techniques on the results of the differential expression test (see Figures 4.2, 4.3, and A.6).

The differential expression tests on the data sets normalized by these various methods showed that both quantile based normalizations performed better than the no normalization technique. The no normalization technique was not able to identify almost any differentially expressed miRNAs and failed to control FDR at 0.05 for the sample sizes of 200 and 400.

For both sample sizes, the differential expression tests on the data sets normalized by the weighted quantile method gave more power (power difference is up to 10%) than the tests on the data sets normalized by the conventional quantile method. That is, one can identify more, up to 10% of total number of features, of the differentially expressed miRNA features if one uses the proposed weighted quantile normalization method instead of the conventional quantile method. The weighted quantile normalization is computationally feasible as well.

For the differential expression tests on the paired tumor-normal miRNA data from the CRC case study, the Wilcoxon SRT found many miRNAs which were called significant in the conventional quantile normalized data, even more significant in the data set normalized by the weighted quantile method.
R code for the application of normalization methods and the differential expression tests can be found in Appendix B.3.
CHAPTER 5
DISCUSSION

5.1 Summary

The motivating CRC case study raises challenging issues about statistical dependence in miRNA data with extensive information about the demographic and lifestyle characteristics of the CRC patients. These issues suggest potential gaps in the analyses of miRNA data that can be filled with novel statistical methods.

The first paper of this dissertation (Chapter 2) proposes the KNN imputation method while accounting for the dependence structure of the data. Based on the results obtained from the simulation data sets, the KNN method shows greater imputation accuracy than the multiple imputation techniques using MCMC and EM algorithms. Moreover, the KNN dependent method has moderate power and maintains control of the FDR, whereas the KNN independent, the MCMC, and the EM methods have higher power but fail to control the FDR. The case deletion method had similar FDR control as KNN dependent, but demonstrated the lowest power among the considered methods.

The contributions of the second paper (Chapter 3) are motivated by the characteristics of the miRNA expression levels of paired tumor-normal samples. Those characteristics are the non-normality of distributions and the imputation-induced dependence among subjects. We proposed three nonparametric methods to identify the differentially expressed miRNAs in the paired tumor-normal data while accounting for the imputation-induced dependence. These methods are Permutation dependent, SRT dependent, and Whitening. The differential expression tests on the simulation data sets show that the Permutation dependent and the SRT dependent methods have moderate power and generally control the FDR, while Whitening has better power but fails to control the FDR. The Permutation independent
and the SRT independent methods, which ignore the dependence structure of data, showed higher power than the other nonparametric methods, but do not maintain control of the FDR. Permutation case deletion and SRT case deletion have the lowest power to identify the differentially expressed miRNAs, but generally control the FDR.

The differential expression tests on the paired tumor-normal miRNA data from the CRC case study demonstrate that the SRT dependent method has worse agreement with the SRT full analysis than the SRT independent and the SRT case deletion methods. This might be due to the poor performance of KNN dependent in predicting the lower-expressed miRNA expressions of missing normal samples. The SRT dependent method showed better agreement with the SRT full analysis for the fully-expressed miRNAs than the partially expressed miRNAs. The results of the permutation methods (including Permutation independent, Permutation case deletion, and Permutation dependent) have similar agreement with the results of the Permutation full method. Compared to these other permutation methods, the Permutation dependent test showed better control of the FDR and moderate power to reproduce the results of the Permutation full analysis.

The novelty of the third paper (Chapter 4) is the incorporation of the characteristics of patients in the normalization of their miRNA expressions, which has not been applied before. The expression levels of miRNAs in risk group (old, heavy smoker, overweight, etc.) patients’ tissues are differentially expressed compared to the expressions in tissues from non-risk group (young, non-smoker, healthy weight, etc.) patients. Availability of extensive information about demographic and lifestyle variables of CRC patients allowed us to consider them with the widely-used quantile normalization method to remove not only the technical differences, but also the biological differences caused by the additional covariates, and to maintain true biological differences between tumor and normal tissues. The proposed weighted quantile normalization incorporates demographic and lifestyle variables by considering the distances (weights) among subjects based on their characteristics, and considers these weights in equating the quantiles of distributions of miRNA expressions. The differential expression testing analyses on simulated data sets that are normalized by the weighted
quantile method give more power than on data sets normalized by the conventional quantile method. Both quantile based normalizations perform better than the no normalization technique and generally maintain control of the FDR. For the given simulation scenarios, the no normalization technique is not able to identify almost any differentially expressed miRNAs and fails to control the FDR.

5.2 Future work

While current needs for the CRC case study have been met with the contributions of this work, future work remains to be done. In the differential expression testing of the paired tumor-normal miRNA data from the CRC case study, the SRT dependent method showed the worst performance in reproducing the results of the SRT full analysis using a filter based on the miRNAs which were expressed in at least 10% of subjects. The SRT dependent method has better agreement with the SRT full analysis based on the miRNAs which are fully-expressed. In this respect, we could explore different filter thresholds (1%, 5%, 20%, etc.) to see how the performance of the SRT dependent method changes.

Another approach to improve the performance of the SRT dependent method is revisiting the denominator of the $Z_M$ statistic of (3.14); the choice used here was mainly for exploratory and demonstration purposes. For example, it may be possible to use cluster overlap information to estimate $Cov(\lambda_i, \lambda_{i'})$, with $\lambda_i$ as in Equation (3.8), and then evaluate $Var(T_M)$ as

$$
\sum_{i=1}^{M} Var(\lambda_i) - 2 \sum_{i<i'} Cov(\lambda_i, \lambda_{i'}). 
$$

(5.1)

In the third paper, we have considered and modified the widely-used quantile normalization method to incorporate demographic and lifestyle variables. However, it would be highly desirable to consider other miRNA normalization methods to see whether they can incorporate the characteristics of subjects and perform at least as well as the weighted quantile normalization. One of the candidate normalization methods used in miRNA data analysis is the normalization method based on the set of invariants. This method performed as well as the quantile normalization method over all experimental conditions tested in Pradervand
et al. [85]. Though the authors did not account for the covariates when they developed this method, it would be worthwhile to investigate incorporating the demographic and lifestyle variables with this normalization method.
REFERENCES


clustered data: two-group comparisons with group membership defined at the subunit


In Image analysis and recognition, pp. 886–893. Springer.

[64] Picard RW (Fall 2010). Topic: decorrelating and then whitening data. MAS
media.mit.edu/2010fall/mas622j/whiten.pdf

Coloring_Transforms_S14_MH.pdf

Machine Blog. https://theclevermachine.wordpress.com/2013/03/30/
the-statistical-whitening-transform/

Starch Press.


APPENDICES
APPENDIX A

Additional plots

A.1 Imputation methods
Fig. A.1. TPR and FPR for sample sizes of 50, 100, 200, and 400 with missingness of 10%-50%
A.2 Nonparametric tests

![Graph showing TPR and FPR for various nonparametric methods and sample sizes](image)

**Fig. A.2.** TPR and FPR of all nonparametric methods for sample sizes of 200 and 400 and missing percentages of 10, 30, and 50
Fig. A.3. TPR and FPR of some of SRT tests and permutation methods for sample sizes of 200 and 400 and missing percentages of 10, 30, and 50
Fig. A.4. TPR and FPR of SRT dependent for sample size of 400 and missing percentages of 10–90
Fig. A.5. Scatterplot matrix of adjusted p-values of SRT full, SRT independent, SRT case deletion, and SRT dependent based on fully-expressed miRNAs (in double negative log-scale)
A.3 Normalization methods

Fig. A.6. TPR and FPR for sample sizes of 200 and 400
**Fig. A.7.** TPR and FPR for sample sizes of 200 and 400 only for conventional and weighted quantile normalizations

**Fig. A.8.** TPR and FPR for sample sizes of 200 and 400 for the quantile normalization and the weighted quantile normalization while accounting for unrelated covariates
APPENDIX B

R code

B.1 R code for imputation methods

setwd("~/DISSERTATION/R code")

## PACKAGES:
library("mi")
library("Amelia")
library("fmsb")

source("ImputationFunctions.R")

## CHARACTERISTICS OF DEMOGRAPHIC AND LIFE STYLE VARIABLES OF CRC:
CRC.cont = read.csv( file="ContVariables.csv", sep="", head=TRUE, as.is =TRUE )
CRC.bin = read.csv( file="BinVariables.csv", sep="", head=TRUE, as.is =TRUE )

generate.data = function(dem.disc, dem.cont, n, G, muN, muT, sigmaN, sigmaT,
missing ){  
NvsT.samples = generate.NandT.data(n, G, muN, muT, sigmaN, sigmaT )
normal = NvsT.samples$normal

tumor = NvsT.samples$tumor

demvar.disc = generate.demog.samples(dem.disc, n)
demvar.cont = generate.demog.samples(dem.cont, n)

missing.rows = sample(nrow(normal), round(nrow(normal)*missing))
missing.normal = normal

missing.normal[missing.rows, ] = NA

fullmissing.data = cbind(missing.normal, tumor, demvar.cont, demvar.disc)
initial.data = cbind(normal, tumor)
withoutNA.data = na.omit(missing.data)

## distance matrices:
aggregate.dist = aggregate.distance(tumor, disc.data=demvar.disc,
cont.data=demvar.cont)

return(list( x.initial=initial.data, x.withoutNA=withoutNA.data,
x.missing=missing.data, x.fullmissing=fullmissing.data,
x.dist=aggregate.dist ) )
}

imputation.kNNImpute = function(x, k, x.dist = NULL){
## re-order matrix:
missing.matrix = is.na(x)
onNA.rows = which( !apply(missing.matrix, 1, function(i){ any(i) }) )
NA.rows = which( apply(missing.matrix, 1, function(i){ any(i) }) )
NA.cols = which( apply(missing.matrix, 2, function(i){any(i)}))

## new matrix
xx = rbind(x[nonNA.rows,], x[NA.rows,])
x.x.nonNA.rows = which( !apply(is.na(xx), 1, function(i){ any(i) }) )
x.x.NA.rows = which( apply(is.na(xx), 1, function(i){ any(i) }) )

if (is.null(x.dist))
x.dist = dist(xx)
V = list()
x3 = matrix(nrow=nrow(x), ncol=length(NA.cols))

for( g in 1:length(NA.cols) ){
x1 = xx[x.x.nonNA.rows, g]
neighbor.indices = which(!is.na(xx)[, g])

A = matrix(nrow=length(xx.nonNA.rows), ncol=length(xx.NA.rows))
for( r in 1:length(xx.NA.rows) ){
  indices.id = .dist.2dto1d.kNNImpute(xx.NA.rows[r], neighbor.indices, nrow(x))
knn.dist = x.dist[indices.id]
ranks = order(knn.dist)
small.dist = knn.dist[ranks]
weights = 1 - (small.dist/max(knn.dist))[1:k]
knn.weights = weights/sum(weights)
knn_weights <- rep(0, length(xx.nonNA.rows))
knn_weights[ranks[1:k]] <- knn.weights
A[,r] <- knn_weights
}
\[ x_2 = t(A) x_1 \]
\[ x_3[,g] = c(x_1, x_2) \]

\[ I = \text{diag}(1, \text{ncol}=\text{length}(x_1)) \]
\[ VV = \text{rbind}( \text{cbind}(I, A), \text{cbind}(t(A), t(A) x_1) ) \cdot \text{var}(x_1) \]
\[ V[g] = VV \]

\[ xx = \text{cbind}(c(\text{nonNA.rows}, \text{NA.rows}), x_3[, (\text{length(NA.cols)}+1):\text{ncol}(xx)]) \]
\[ x.\text{imputed} = xx[\text{order}(xx[,i]),] \]
\[ x.\text{imputed} = x.\text{imputed}[, -1] \]

\[ \text{colnames}(x.\text{imputed}) = \text{colnames}(x) \]

\[ \text{return}(\ \text{list}(x.\text{imputed} = x.\text{imputed}, \ \text{VarCovMatrix} = V)) \]

} ## End of function

generate.results = function(imputed.data, G, true.DE, CovMatrix, dependence, correction, pmethod){

\[ \text{normal} = \text{imputed.data}[,1:G] \]
\[ \text{tumor} = \text{imputed.data}[, (G+1):(2*G)] \]

## p-values:
\[ \text{pvals} = \text{pairedttest.kNNImpute}(\text{normal, tumor, CovMatrix, dependence, correction, pmethod}) \]
\[ \text{pvals} = \text{cbind}(\text{pvals, true.DE}) \]
\[ \text{pvals} = \text{matrix}(\ \text{pvals}[\text{order}(\text{pvals}[,1], \text{decreasing=}\text{TRUE}),], \text{nrow}=2) \]

\[ \text{DE.genes} = \text{pvals}[,2] \]
\[ \text{true.DE} = \text{length}(\text{which}(\text{DE.genes==1})) \]
\[ \text{nottrue.DE} = \text{length}(\text{which}(\text{DE.genes==0})) \]

\[ \text{ROC.rates} = c() \]
\[ \text{for}(i \text{ in } 1: \text{length}((\text{DE.genes})))\{
\]
\[ \text{sig.DE} = \text{DE.genes}[i: \text{length}((\text{DE.genes}))] \]

## ROC rates:
\[ \text{TPR} = (\text{length}(\text{which}(\text{sig.DE==1}))/\text{true.DE}) \]
\[ \text{FPR} = (\text{length}(\text{which}(\text{sig.DE==0}))/\text{nottrue.DE}) \]

\[ \text{TPR}[\text{which}(\text{TPR=="NaN"}]] = 1 \]
\[ \text{FPR}[\text{which}(\text{FPR=="NaN"}]] = 1 \]

\[ \text{ROC.rates} <- \text{rbind}(\text{ROC.rates, c(FPR, TPR)}) \]
}
colnames(ROC.rates) = c("FPR", "TPR")
return(ROC.rates)
}

############################
## SIMULATION PARAMETERS: ##
############################
n = 50 # n = 50, 100, 200, and 400
missing = 0.1 # missing = 0.1, 0.2, 0.3, 0.4, and 0.5

G = 2000 # MIRNAS
k = 10 # K-NEAREST NEIGHBORS

meanN=2; meanT=2.5; sdN=1.25; sdT=1.25
muN = rep(meanN, G)
muT = muN
muT[which(true_DE==1)] = meanT
sigmaN = rep(sdN, G)
sigmaT = rep(sdT, G)

set.seed(1)
DE=0.2
true_DE = rbinom(G, 1, DE)
mirnaDE = which(true_DE==1)

pX = 0.05
nDE = round(G*pX*DE)

seed = 123; StartSim = 1; EndSim = 20

set.seed(seed)

RMSE.TOTAL = c()
for( sim in StartSim:EndSim){ # BEGIN SIMULATION PART

## GENERATE MIRNA AND DEMOGRAPHIC DATA SETS:
data = generate.data(CRC.cont, CRC.bin, n, G, muN, muT, sigmaN, sigmaT, missing)

x.initial = data$x.initial
x = data$x.missing
x.demog = data$x.fullmissing[,c(1:(2*G))]

x.CD = data$x.withoutNA
x.dist = data$x.dist
missing.rows = data$missing.rows

x.mi = cbind( x[,1:G], x.demog )
colnames(x.mi) = rep(1:ncol(x.mi))
selec.vars = (G+1):ncol(x.mi)
for(i in 1:ncol(x.mi))
    colnames(x.mi)[i] = paste("V", colnames(x.mi)[i], sep="")
x.mi = as.data.frame(x.mi)
x.em = x.mi

#################
## IMPUTATIONS ##
#################

## KNN (ALSO COVARIANCE MATRICES):
out = imputation.kNNImpute(x=x, k=k, x.dist=x.dist)

## MI (MCMC):
mi.imputed = mimi(x.mi, G, selec.vars)
mi.imputed = cbind(mi.imputed[,1:G], x.initial[, (G+1):(2*G)])

## MI (EM):
em.imputed = emem(x.em, G, selec.vars, n)
em.imputed = cbind(em.imputed[,1:G], x.initial[, (G+1):(2*G)])

#################
## RMSE VALUES ##
#################

## RMSE FOR KNN:
R = x.initial[missing.rows, 1:G]
I = out$x.imputed[missing.rows, 1:G]
N = ncol(I)*nrow(I)
RMSE.KNN = sqrt( sum((R-I)^2)/N )

## RMSE FOR MI:
R = x.initial[missing.rows, 1:G]
I = mi.imputed[missing.rows, 1:G]
N = ncol(I)*nrow(I)
RMSE.MI = sqrt( sum((R-I)^2)/N )

## RMSE FOR EM:
R = x.initial[missing.rows, 1:G]
I = em.imputed[missing.rows, 1:G]
N = ncol(I) * nrow(I)
RMSE.EM = sqrt( sum((R-I)^2)/N )

RMSE = c(sim, RMSE.KNN, RMSE.MI, RMSE.EM)
RMSE.TOTAL = rbind(RMSE.TOTAL, RMSE)
colnames(RMSE.TOTAL) = c("SimN", "RMSE.KNN", "RMSE.MI", "RMSE.EM")

## PRODUCE RMSE OUTPUTS:
write.table(RMSE.TOTAL, file=paste("RMSE_miss", missing*100, ",", n, ",", sim, StartSim, "-_", EndSim, ",", sep="", sep="", row.names = FALSE, col.names = TRUE)

### ANALYSIS I: FULL DATA
results.full = generate.results(imputed.data=x.initial, G, true_DE=true_DE, CovMatrix=FALSE, dependence=FALSE, correction=TRUE, pmethod="fdr")

## ANALYSIS II (ACCOUNTING FOR DEPENDENCE):
results.dep = generate.results(imputed.data=out$x.imputed, G, true_DE=true_DE, CovMatrix=out$VarCovMatrix, dependence=TRUE, correction=TRUE, pmethod="fdr")

## ANALYSIS III (IGNORING DEPENDENCE):
results.indep = generate.results(imputed.data=out$x.imputed, G, true_DE=true_DE, CovMatrix=FALSE, dependence=FALSE, correction=TRUE, pmethod="fdr")

## ANALYSIS IV (MCMC):
results.mi = generate.results(imputed.data=mi.imputed, G, true_DE=true_DE, CovMatrix=FALSE, dependence=FALSE, correction=TRUE, pmethod="fdr")

## ANALYSIS V (EM):
results.em = generate.results(imputed.data=em.imputed, G, true_DE=true_DE, CovMatrix=FALSE, dependence=FALSE, correction=TRUE, pmethod="fdr")

## ANALYSIS VI (CASE DELETION):
results.CD = generate.results(imputed.data=x.CD, G, true_DE=true_DE, CovMatrix=FALSE, dependence=FALSE, correction=TRUE, pmethod="fdr")

## PRODUCE ROC OUTPUTS:
outputs = cbind(results.full$pval.raw, results.dep$pval.raw, results.indep$pval.raw, results.mi$pval.raw, results.em$pval.raw, results.CD$pval.raw, results.full$pval.cor, results.dep$pval.cor, results.indep$pval.cor, results.mi$pval.cor, results.em$pval.cor, results.CD$pval.cor, results.full$ROC.rates[,1], results.full$ROC.rates[,2], results.dep$ROC.rates[,1], results.dep$ROC.rates[,2], results.CD$ROC.rates[,1], results.CD$ROC.rates[,2], results.indep$ROC.rates[,1], results.indep$ROC.rates[,2], results.mi$ROC.rates[,1], results.mi$ROC.rates[,2], results.em$ROC.rates[,1], results.em$ROC.rates[,2])
results.full$ROC.rates[,3], results.dep$ROC.rates[,1],
results.dep$ROC.rates[,2], results.dep$ROC.rates[,3],
results.indep$ROC.rates[,1], results.indep$ROC.rates[,2],
results.indep$ROC.rates[,3], results.mi$ROC.rates[,1],
results.mi$ROC.rates[,2], results.mi$ROC.rates[,3], results.em$ROC.rates[,1],
results.em$ROC.rates[,2], results.em$ROC.rates[,3], results.CD$ROC.rates[,1],
results.CD$ROC.rates[,2], results.CD$ROC.rates[,3])

colnames(outputs) = c("RawPval_full", "RawPval_dep", "RawPval_indep", "RawPval_mi",
"FPR_dep", "TPR_dep", "FDR_dep", "FPR_indep", "TPR_indep", "FDR_indep",
"FPR_mi", "TPR_mi", "FDR_mi", "FPR_em", "TPR_em", "FDR_em", "FPR_CD",
"TPR_CD", "FDR_CD")

write.table(outputs, file=paste("Outputs_miss", missing*100, ",_n", n, ",_sim",
                           sim, ",.csv", sep=""), sep="", row.names = FALSE, col.names = TRUE)

print(paste("End of Simulation: ", sim, sep=""))

} # END OF SIMULATION PART

B.2 R code for nonparametric methods

inpath = "/DISSERTATION/R code/
setwd("/DISSERTATION/R code/Nonparametric/")

library("mi")
library("Amelia")
library("fmsb")
library("gtools")
library("Rcpp")
library("RcppArmadillo")
library("inline")
library("MASS")
library("Matrix")
library("rbenchmark")
library("multtest")

source("Nonparametric_Functions.R")
sourceCpp("RcppArmadillo_functions.cpp")

# SOME OF REQUIRED FUNCTIONS: #
generate.nondata = function(CRC, n, G, mu1, muN, muT, sig1, sigmaN, sigmaT, missing, mirnaDE, nDE, cpct ){

NvsT.samples = generate.bimodaldata(n, G, mu1, muN, muT, sig1, sigmaN, sigmaT, cpct)
normal = NvsT.samples$normal
tumor = NvsT.samples$tumor

## CRC VARIABLES:
demvar.cont1 = predDemContVar(data=tumor, CRC=CRC, variable="SEL_AGE", rounding=TRUE)
demvar.cont2 = predDemContVar(data=tumor, CRC=CRC, variable="CIGAMT", rounding=TRUE)
demvar.cont3 = predDemContVar(data=tumor, CRC=CRC, variable="CALORIES", rounding=TRUE)
demvar.cont4 = predDemContVar(data=tumor, CRC=CRC, variable="BMIADJ", rounding=FALSE)
demvar.cont5 = predDemContVar(data=tumor, CRC=CRC, variable="LUTZEA", rounding=FALSE)

demvar.disc1 = predDemBinVar(data=tumor, CRC=CRC, variable="SEX")
demvar.disc2 = predDemBinVar(data=tumor, CRC=CRC, variable="RecentAspIbu")
demvar.disc3 = predDemBinVar(data=tumor, CRC=CRC, variable="RecentSmoke")
demvar.disc4 = predDemBinVar(data=tumor, CRC=CRC, variable="MENOPAU")
demvar.disc5 = predDemBinVar(data=tumor, CRC=CRC, variable="HRTPOST")

demvar.cont = cbind(demvar.cont1, demvar.cont2, demvar.cont3, demvar.cont4, demvar.cont5)
demvar.disc = cbind(demvar.disc1, demvar.disc2, demvar.disc3, demvar.disc4, demvar.disc5)

## MISSINGNESS:
missing.rows = sort( sample(nrow(normal), round(nrow(normal)*missing)) )
nonmiss.rows = which(!(1:n %in% missing.rows))
missing.normal = normal
missing.normal[missing.rows, ] = NA

## REORDER SAMPLES/ROWS OF MIRNAS AND CRC:
normal = rbind(normal[nonmiss.rows, ], normal[missing.rows, ])
tumor = rbind(tumor[nonmiss.rows, ], tumor[missing.rows, ])
missing.normal = rbind(missing.normal[nonmiss.rows, ], missing.normal[missing.rows, ])

initial.data = cbind(normal, tumor)
missing.data = cbind(missing.normal, tumor)
withoutNA.data = na.omit(missing.data)
fullmissing.data = cbind(missing.normal, tumor, demvar.cont, demvar.disc)

demvar.cont2 = rbind(demvar.cont[nonmiss.rows, ], demvar.cont[missing.rows, ])
demvar.disc2 = rbind(demvar.disc[nonmiss.rows, ], demvar.disc[missing.rows, ])

## AGGREGATE DISTANCE MATRICES:
aggregate.dist = aggregate.distance(disc.data=demvar.disc2, cont.data=demvar.cont2)
return(list( missing.rows=missing.rows, x.initial=initial.data, 
            x.withoutNA=withoutNA.data, x.missing=missing.data, 
            x.fullmissing=fullmissing.data, x.dist=aggregate.dist) )
}

#####################
## RCCP FUNCTIONS: ##
#####################

// [[Rcpp::export]]
double Z_cpp( arma::mat xdiff, arma::vec ni, int& M, int& N, 
              arma::mat omega_i, arma::mat V, List omega_ij, 
              Function f1, Function f2, Function f3){
  NumericMatrix x = wrap(xdiff);
  NumericVector n = wrap(ni);
  double A = 0.0, TM = 0.0, SM = 0.0;

  for(int i=0; i<M; i++){
    mat xdiff2 = xdiff.rows( find( xdiff(span(),1)==i+1 ) );
    NumericMatrix x2 = wrap(xdiff2);
    NumericVector x21 = x2(_, 0);
    NumericVector b(x21.size());
    NumericVector A1 = f1(x21 > 0), A2=f1(x21 < 0);
    A = ( A1[0]-A2[0] );
    NumericVector B1 = f2(x21);
    NumericMatrix B2 = f3(x, x2, i+1, N, b);
    NumericVector B = f1( B1 * B2 );
    TM += (A + B[0]) / ni[i];
  }

  mat omegai = omega_i(i, span() );
  NumericVector Omegi = wrap(omegai * V.diag());
  NumericMatrix Omegij1 = omega_ij[i];
  NumericVector Omegij = f1(Omegij1);
  NumericVector SM1 = Omegi + Omegij;
  SM += SM1[0];
}

double Z = TM / sqrt(SM);
return Z;
}

// [[Rcpp::export]]
cube VarCovMatrixPermuted_cpp( int& n, int& G, mat& tumor, List& VV,
                                NumericVector pindex, Function diag ){

  NumericVector t(n);
std::fill(t.begin(), t.end(), 1);
for(int i=0; i<pindex.size(); ++i){ // REPLACE TO t[pind] = -1
    t[pindex[i]-1] = -1;
}
NumericMatrix A2 = diag(t, n);
momat A = Rcpp::as<arma::mat>(A2); // CONVERT RCPP TO RCPPARMADILLO

mat X2 = VV[i];
cube V(X2.n_rows, X2.n_cols, G, fill::zeros);

for(uword i=0; i<G; ++i){
    mat Vn = VV[i];
    mat Vt = mat(Vn.n_rows, Vn.n_cols, fill::eye) * var(tumor(span(),i));
    mat Vd = Vn + Vt;
    V.slice(i) = A*Vd*trans(A);
}
return V;
}

// [[Rcpp::export]]
mat pairedTTest_perm_cpp( mat& pnormal, mat& ptumor, NumericVector V ) {
    mat D = ptumor - pnormal;
    mat Ones = ones( D.n_rows );

    // ARRAY TO CUBE
    NumericVector vecArray( V );
    IntegerVector arrayDims = vecArray.attr("dim");
cube VV(vecArray.begin(), arrayDims[0], arrayDims[1], arrayDims[2], false);

    vec z = zeros( D.n_cols );
    for(int j=0; j<D.n_cols; ++j){
        mat U = VV.slice(j);
        vec Dj = D( span(), j );
        mat onesT = trans( Ones );
        mat Uinv = inv( U );
        mat U1inv = inv( onesT * Uinv * Ones );
        mat muhat = U1inv * onesT * Uinv * Dj;
        mat DD = Dj - Ones * muhat;
        mat Q = DD.t() * Uinv * DD;
        mat sigma2 = Q / (D.n_rows - 1);
        mat Cov_muhat = sigma2 * Uinv;
        mat z_test = muhat / sqrt(Cov_muhat);
        z[j] = z_test[0];
    }
}
return z;
StartPerm=1; EndPerm=50000; B=50000; sim=1; seed=1
missings=c(0.1,0.2,0.3,0.4,0.5); samples=c(200, 400)
G=2000; k=10; DE=0.2; alpha=0.05;
meanN=4; meanT1=3.25; meanT2=4.75; mu1=0.75;
sdN=0.5; sdT=0.5; sig1=0.025; cpct=0.5
set.seed(1)
true_DE = rbinom(G, 1, DE)
muN = rep(meanN, G)
muT <- muN
for(i in which(true_DE==1)){ muT[i] = sample(c(meanT1, meanT2),1) }
sigmaN = rep(sdN, G)
sigmaT = rep(sdT, G)
mirnaDE = which(true_DE==1)
pX = 0.50
nDE = round(length(mirnaDE)*pX)

CRC.distal = read.csv(file = paste(inpath, "CRC_distal_2.csv", sep=""), sep="",
                   head = TRUE, as.is=TRUE)
CRC.proximal = read.csv(file = paste(inpath, "CRC_proximal_2.csv", sep=""), sep="",
                      head = TRUE, as.is=TRUE)
CRC.rectal = read.csv(file = paste(inpath, "CRC_rectal_2.csv", sep=""), sep="",
                 head = TRUE, as.is=TRUE)
CRC = rbind(CRC.distal, CRC.proximal, CRC.rectal)

for( n in samples ){ for( missing in missings ){} }
set.seed(seed)

## GENERATE DATA FOR NONPARAMETRIC ANALYSIS:
data = generate.nondata(CRC, n, G, mu1, muN, muT, sig1, sigmaN, sigmaT, missing, 
mirnaDE, nDE, cpct=0.5 )
x.initial = data$x.initial
x = data$x.missing
x.CD = data$x.withoutNA
x.dist = data$x.dist

D.full = x.initial[,((G+1):(2*G))] - x.initial[,1:G]
D.cd = x.CD[,((G+1):(2*G))] - x.CD[,1:G]

## IMPUTATION AND VAR-COVARIANCE MATRICES:
out = imputation.kNNImpute(x=x, k=k, x.dist=x.dist)
nonNA.rows = which(!(1:n %in% out$imputed.samples))
NA.rows = out$imputed.samples

normal = out$x.imputed[,1:G]
tumor = out$x.imputed[,((G+1):(2*G))]
D.imp = tumor - normal
meanDiff = apply(tumor-normal, 2, mean)

# PERMUTATION ANALYSES 

m=1
V2 = out$VarCovMatrix[[m]]
V = V2 + diag(var(tumor[,m]), nrow=n)
x.diff = xclusters(normal[,m], tumor[,m], V, NA.rows, nonNA.rows)
omega_i = out$Coefficients # WEIGHTS OF VARIANCES OF CLUSTERS
omega_ij = WeightCovClus(x.diff, NA.rows, nonNA.rows, omega_i) # WEIGHTS OF COVARIANCES OF CLUSTERS
ni <- as.vector(table(x.diff[,2]))
M <- length(ni)
cni <- c(0, cumsum(ni))
N <- length(ni)

for(i in 1:G){
  ## 1. WILCOXON SRT ON FULL/NON-MISSING DATA:
  test.full = wilcox.test(D.full[,i], alternative="two.sided")
Pval.full = c(Pval.full, test.full$p.value)

  ## 2. WILCOXON SRT ON FULL/NON-MISSING DATA:
  test.imp = wilcox.test(D.imp[,i], alternative="two.sided")
Pval.imp = c(Pval.imp, test.imp$p.value)

  ## 3. WILCOXON SRT ON CASE DELETION DATA:
  test.cd = wilcox.test(D.cd[,i], alternative="two.sided")
Pval.cd = c(Pval.cd, test.cd$p.value)

  ## 4. WHITENING TRANSFORMATION:
  V2 = out$VarCovMatrix[i]
  V = V2 + diag(var(tumor[,i]), nrow=n)
  V = as.matrix(V)
  D = as.matrix(D.imp[,i], ncol=1)
  D2 = D - mean(D)

  if(all(eigen(V)$values<=0))
    print(paste(m, " IS NOT POSITIVE MATRIX", sep=""))
  Lam = diag(eigen(V)$values, nrow=length(eigen(V)$values))
  Phi = eigen(V)$vectors
  Y = Phi%*%D2
  W1 = sqrt(solve(Lam))%*%Y
  W = W1 + mean(D)
  test.white = wilcox.test(W, alternative="two.sided")
Pval.white = c(Pval.white, test.white$p.value)

  x.diff = xclusters(normal[,i], tumor[,i], V, NA.rows, nonNA.rows)

  ## 5. SRT_DS (DATTA & SATTEN (2008)):
  Xij=x.diff[,1]
  test.ds = DS_cpp( Xij, cni, ni, M, f1=sum, f2=sign )
Pval.ds = c(Pval.ds, 2*(1-pnorm(abs(test.ds))))

  ## 6a. SRT_DEP: ni=ni
  test.dep = Z_cpp( xdiff=as.matrix(x.diff), ni=ni, M=M, N=N, omega_i=omega_i, V=V,
                  omega_ij=omega_ij, f1=sum, f2=sign, f3=B_cpp)
Pval.dep = c(Pval.dep, 2*(1-pnorm(abs(test.dep))))

  ## 7. AVERAGE CLUSTERED SRT (FOR RHO != 0):
X = c()
for( i in 1:M ){ X = c(X, sum(x.diff$Diff[(cni[i]+1):cni[i+1]])/ni[i]) }
test.ave = wilcox.test( X, alternative="two.sided" )
Pval.ave = c(Pval.ave, test.ave$p.value)
}

## 8. maxT: FULL AND IMPUTED
normal.full = t(x.initial[,1:G])
tumor.full = t(x.initial[,,(G+1):(2*G)])
normal.imp = t(normal)
tumor.imp = t(tumor)

X.full = matrix(nrow=G, ncol=(2*n))
X.imp = matrix(nrow=G, ncol=(2*n))
nn = seq(1, 2*n, by=2)
for( i in 1:n ){
    X.imp[,nn[i]:(nn[i]+1)] = cbind(normal.imp[,i], tumor.imp[,i])
    X.full[,nn[i]:(nn[i]+1)] = cbind(normal.full[,i], tumor.full[,i])
}

classlabel<-rep(c(0,1), n)
resT.full = mt.maxT(X.full, classlabel, test="pairt", side="abs", B=B)
resT.full = resT.full[order(resT.full[,1]),]
resT.imp = mt.maxT(X.imp, classlabel, test="pairt", side="abs", B=B)
resT.imp = resT.imp[order(resT.imp[,1]),]

## 9. T-TEST OBSERVED:
tobs = pairedTTest_nonperm_cpp( pnormal=normal, ptumor=tumor, V=out$VarCovMatrix, permute=0 )
ttest.permuted = cbind(true_DE, meanDiff, Pval.full, Pval.imp, Pval.cd, Pval.white, Pval.ds, Pval.dep, Pval.ave, resT.full$rawp, resT.imp$rawp, tobs)

colnames(ttest.permuted) = c("True_DE","MeanDiff","WilcoxFull","WilcoxImp", "WilcoxCD","White","SRT_DS","SRT_dep","SRT_ave","maxTfull", "maxImp", "t_obs")

## 10. PERMUTATION TESTS:
set.seed(seed+StartPerm)
for( j in StartPerm:EndPerm ){ # BEGIN PERMUTATION

## PERMUTED SUBJECTS:
indexes = rep(1:n)
pindex = which( rbinom( n, 1, prob=0.5 ) == 1 )
notindex = which(indexes %in% pindex == FALSE)

## DATA AFTER PERMUTATION:
pnormal = normal
\begin{verbatim}

ptumor = tumor
pnorm[pindex,] = tumor[pindex,]
ptumor[pindex,] = normal[pindex,]

## COVARIANCE MATRICES:
VarCovMatrix.permuted = VarCovMatrixPermuted_cpp(n, G, tumor, VV=out$VarCovMatrix, pindex, diag)

## PAIRED T-TEST:
tstat = pairedTTest_perm_cpp( pnormal, ptumor, V=VarCovMatrix.permuted )
colnames(tstat) = paste("t_perm", j, sep=""

ttest.permuted = cbind(ttest.permuted, tstat)

   print(paste("End of Test Permutation: ", j, " -- ", date(), sep="")) }

} ## END OF PERMUTATIONS


B.3 R code for normalization methods

inpath = "~/DISSERTATION/R code/"
outpath = "~/DISSERTATION/R code/Normalization/"
library("preprocessCore")
source(paste(outpath, "Normalization_Functions.R", sep=""))

# SOME OF REQUIRED FUNCTIONS:

generate.data = function(CRC, n, G, mu1, muN, muT, sig1, sigmaN, sigmaT, subjDE){

NvsT.samples = generate.bimodaldata(n, G, mu1, muN, muT, sig1, sigmaN, sigmaT, subjDE)
normal = NvsT.samples$normal

tumor = NvsT.samples$tumor

# CRC VARIABLES:

\end{verbatim}
demvar.cont1 = predDemContVar(data=tumor, CRC=CRC, variable="SEL_AGE", rounding=TRUE)
demvar.cont2 = predDemContVar(data=tumor, CRC=CRC, variable="CIGAMT", rounding=TRUE)
demvar.cont3 = predDemContVar(data=tumor, CRC=CRC, variable="CALORIES", rounding=TRUE)
demvar.cont4 = predDemContVar(data=tumor, CRC=CRC, variable="BMIADJ", rounding=FALSE)
demvar.cont5 = predDemContVar(data=tumor, CRC=CRC, variable="LUTZEA", rounding=FALSE)
demvar.cont6 = predDemContVar(data=tumor, CRC=CRC, variable="VITD", rounding=FALSE)
demvar.cont7 = predDemContVar(data=tumor, CRC=CRC, variable="LYCOPENE", rounding=FALSE)
demvar.disc1 = predDemBinVar(data=tumor, CRC=CRC, variable="SEX")
demvar.disc2 = predDemBinVar(data=tumor, CRC=CRC, variable="RecentAspIbu")
demvar.disc3 = predDemBinVar(data=tumor, CRC=CRC, variable="RecentSmoke")
demvar.disc4 = predDemBinVar(data=tumor, CRC=CRC, variable="MENOPAU")
demvar.disc5 = predDemBinVar(data=tumor, CRC=CRC, variable="HRTPOST")
demvar.disc6 = predDemBinVar(data=tumor, CRC=CRC, variable="CENTER")
demvar.disc7 = predDemBinVar(data=tumor, CRC=CRC, variable="RACE")
demvar.disc8 = predDemBinVar(data=tumor, CRC=CRC, variable="CIGARETT")
demvar.disc9 = predDemBinVar(data=tumor, CRC=CRC, variable="ALC_LT")
demvar.disc10 = predDemBinVar(data=tumor, CRC=CRC, variable="STAGE")
demvar.disc11 = predDemBinVar(data=tumor, CRC=CRC, variable="AJCC")
demvar.disc12 = predDemBinVar(data=tumor, CRC=CRC, variable="STUDY")

demvar.cont = cbind(demvar.cont1, demvar.cont2, demvar.cont3, demvar.cont4,
                    demvar.cont5, demvar.cont6, demvar.cont7)
demvar.disc = cbind(demvar.disc1, demvar.disc2, demvar.disc3, demvar.disc4,
                    demvar.disc5, demvar.disc6, demvar.disc7, demvar.disc8,
                    demvar.disc9, demvar.disc10, demvar.disc11, demvar.disc12)
demdata = cbind(demvar.cont, demvar.disc)

## AGGREGATE DISTANCE MATRICES:
aggregate.dist = aggregate.distance(disc.data=demvar.disc, cont.data=demvar.cont)

return(list( nmiRNA=normal, tmiRNA=tumor, demdata=demdata, x.dist=aggregate.dist) )
}

weightedquantile = function(data, CRC, x.dist){
  data = t(data)

  ## 1. ORIGINAL MATRIX (COLUMNS ARE SAMPLES, ROWS ARE MIRNAS):
  x1 = as.matrix(data)
  x1.rank = apply(x1, 2, rank)

  ## 2. REORDER WITHIN COLUMN:
  x2 = apply(x1, 2, sort)

  ## DISTANCE / WEIGHT:
  aggreg.dist = x.dist
  D = as.matrix(aggregate.dist)
$W = \text{matrix}(\text{nrow}=\text{nrow}(D), \text{ncol}=\text{ncol}(D))$

for(i in 1:\text{nrow}(D)){
    for(j in 1:\text{ncol}(D)){
        \hspace{0.5em}W[i,j] = 1 - D[i,j] / \max(D[i,])
    }
}

$W^{\star} = \text{matrix}(\text{nrow}=\text{nrow}(W), \text{ncol}=\text{ncol}(W))$

for(i in 1:\text{nrow}(W)){
    for(j in 1:\text{ncol}(W)){
        \hspace{0.5em}W^{\star}[i,j] = W[i,j] / \sum(W[i,])
    }
}

## 3. CALCULATE WEIGHTED MEANS:
$x_3 = x_2 \times W^{\star}$

## 4. UNSORT
$x_4 = \text{matrix}(\text{nrow}=\text{nrow}(x_3), \text{ncol}=\text{ncol}(x_3))$

for(i in 1:\text{ncol}(x_4)){ x_4[,i] = x_3[,i][x_1.\text{rank}[,i]] }$

return(x_4)}$

getti

# SIMULATION PARAMETERS: #

set.seed(1)
true_DE = \text{rbinom}(G, 1, \alpha)\hspace{0.5em}muN = \text{rep}(meanN, G)
muT <- muN
for(i in which(true_DE==1)){ muT[i] = \text{sample}(c(meanT1, meanT2),1) }

sigmaN = \text{rep}(sdN, G)
sigmaT = \text{rep}(sdT, G)

mirnaDE = which(true_DE==1)
pX = 0.50
nDE = \text{round}(\text{length}(\text{mirnaDE})\times pX)
### UPLOAD DEMOGRAPHIC CRC DATA: ###

```r
CRC.distal = read.csv(file = paste(inpath, "CRC_distal_2.csv", sep=""), sep="",
    head = TRUE, as.is=TRUE)
CRC.proximal = read.csv(file = paste(inpath, "CRC_proximal_2.csv", sep=""), sep="",
    head = TRUE, as.is=TRUE)
CRC.rectal = read.csv(file = paste(inpath, "CRC_rectal_2.csv", sep=""), sep="",
    head = TRUE, as.is=TRUE)

CRC2 = rbind(CRC.distal, CRC.proximal, CRC.rectal)
CRC = CRC2[order(CRC$SID),]
rownames(CRC) = c(1:nrow(CRC))

CRC = CRC[-which(duplicated(CRC$SID)),]
```

### ANALYSES: ###

```r
for(n in samples){
    subjDE = sort(sample(1:n, n*.5)) # SUBJECTS IN RISK GROUP

    for( sim in StartSim:EndSim ){ # BEGIN OF SIMULATION
        set.seed(seed + sim)

        ## GENERATE BIMODAL MIRNA AND DEMOGRAPHIC DATA SETS:
        data = generate.data(CRC, n, G, mu1, muN, muT, sig1, sigmaN, sigmaT, subjDE)

        ## LOG TRANSFORMED MIRNA:
        nmiRNA = data$nmiRNA
        tmiRNA = data$tmiRNA

        ## RAW MIRNA:
        nmiRNAraw = nmiRNA
        tmiRNAraw = tmiRNA

        demdata = data$demdata
        x.dist = data$x.dist

        ## NO NORMALIZATION ON RAW DATA:
        pval = c()
```
for(i in 1:ncol(nmiRNA)){
    t = wilcox.test( tmiRNAraw[,i], nmiRNAraw[,i], paired=TRUE )
    pval = c(pval, t$p.value)
}
pval[which(pval=="NaN")]=1

## CONVENTIONAL QUANTILE NORMALIZATION:
nmiRNA.quant = normalize.quantiles(t(nmiRNA))
tmiRNA.quant = normalize.quantiles(t(tmiRNA))

quant.pval = c()
for(i in 1:nrow(nmiRNA.quant)){
    t = wilcox.test( tmiRNA.quant[i,], nmiRNA.quant[i,], paired=TRUE )
    quant.pval = c(quant.pval, t$p.value)
}
quant.pval[which(quant.pval=="NaN")]=1

## WEIGHTED QUANTILE NORMALIZATION:
nmiRNA.weight = weightedquantile(nmiRNA, demdata, x.dist)
tmiRNA.weight = weightedquantile(tmiRNA, demdata, x.dist)

weighted.pval = c()
for(i in 1:nrow(nmiRNA.weight)){
    t = wilcox.test( tmiRNA.weight[i,], nmiRNA.weight[i,], paired=TRUE )
    weighted.pval = c(weighted.pval, t$p.value)
}
weighted.pval[which(weighted.pval=="NaN")]=1

## PRODUCE RAW P-VALUES:
out = cbind(true_DE, pval, quant.pval, weighted.pval)
colnames(out) = c("True_DE", "RawPval_no", "RawPval_quant", "RawPval_weight")

write.table(out, file=paste(outpath, "Normalize_sim", sim, ",", n, ".csv", sep=""),
            row.names=FALSE, col.names=TRUE)

print(paste("End of Simulation: ", sim, " -- ", date(), sep=""))

} # END OF SIMULATION
}
APPENDIX C

Journal copyright release form

Rights and permissions of *PLOS ONE* journal

PLOS ONE is an international, peer-reviewed, open-access, online publication. The journal mentions that “PLOS applies the Creative Commons Attribution License (CCAL) to all works we publish. Under the CCAL, authors retain ownership of the copyright for their article, but authors allow anyone to download, reuse, reprint, modify, distribute, and/or copy articles in PLOS journals, so long as the original authors and source are cited. No permission is required from the authors or the publishers” [91].
APPENDIX D

Non-committee co-author release forms

May 4, 2015

Jennifer Herrick
Division of Epidemiology
Department of Internal Medicine
University of Utah School of Medicine
383 Colow Row, Salt Lake City, UT 84108

Dear Jennifer Herrick:

I am preparing my dissertation in the Mathematics and Statistics Department at Utah State University. I plan to complete my degree in the summer semester of 2015.

I am requesting your permission to include our co-authored paper “Accounting for Dependence Induced by Weighted KNN Imputation in Paired Samples, Motivated by a Colorectal Cancer Study” which appeared in PLoS ONE journal (April 7, 2015, DOI: 10.1371/journal.pone.0119876), as a chapter in my dissertation. I will include acknowledgments and appropriate citations to your role as co-author in the dissertation.

Please indicate your approval to release the paper to be published in my dissertation by signing in the space provided below. If you have any questions, please contact me.

Thank you for your assistance,

[Signature]

Anvar Suyundikov

I hereby give permission to Anvar Suyundikov to release the publication “Accounting for Dependence Induced by Weighted KNN Imputation in Paired Samples, Motivated by a Colorectal Cancer Study”, to be printed in his dissertation.

[Signature] Jennifer Herrick  Date 5/4/15

Fig. D.1. Permission letter from Jennifer Herrick
May 4, 2015

Dr. Roger K. Wolff
Division of Epidemiology
Department of Internal Medicine
University of Utah School of Medicine
383 Coloway Road, Salt Lake City, UT 84108

Dear Dr. Wolff:

I am preparing my dissertation in the Mathematics and Statistics Department at Utah State University. I plan to complete my degree in the summer semester of 2015.

I am requesting your permission to include our co-authored paper "Accounting for Dependence Induced by Weighted KNN Imputation in Paired Samples, Motivated by a Colorectal Cancer Study" which appeared in PloS ONE Journal (April 7, 2013, DOI: 10.1371/journal.pone.0119876), as a chapter in my dissertation. I will include acknowledgments and appropriate citations to your role as co-author in the dissertation.

Please indicate your approval to release the paper to be published in my dissertation by signing in the space provided below. If you have any questions, please contact me.

Thank you for your assistance,

Anvar Suyundikov

__________________________________________

I hereby give permission to Anvar Suyundikov to release the publication "Accounting for Dependence Induced by Weighted KNN Imputation in Paired Samples, Motivated by a Colorectal Cancer Study", to be printed in his dissertation.

Signed: [Signature] Date 5/4/15

Fig. D.2. Permission letter from Roger K. Wolff
May 4, 2015

Dr. Martha L. Slattery
Division of Epidemiology
Department of Internal Medicine
University of Utah School of Medicine
233 Colorow Road, Salt Lake City, UT 84108

Dear Dr. Slattery:

I am preparing my dissertation in the Mathematics and Statistics Department at Utah State University. I plan to complete my degree in the summer semester of 2015.

I am requesting your permission to include our co-authored paper "Accounting for Dependence Induced by Weighted KNN Imputation in Paired Samples, Motivated by a Colorectal Cancer Study" which appeared in PloS ONE journal (April 7, 2015, DOI: 10.1371/journal.pone.0119876), as a chapter in my dissertation. I will include acknowledgments and appropriate citations to your role as co-author in the dissertation.

Please indicate your approval to release the paper to be published in my dissertation by signing in the space provided below. If you have any questions, please contact me.

Thank you for your assistance,

\[Signature\]

\[Name\]

I hereby give permission to Arvar Sayndikov to release the publication “Accounting for Dependence Induced by Weighted KNN Imputation in Paired Samples, Motivated by a Colorectal Cancer Study”, to be printed in his dissertation.

Signed: \[Signature\]  Date: 5-04-2015

Fig. D.3.  Permission letter from Martha L. Slattery
CURRICULUM VITAE

Anvar Suyundikov

PhD in Mathematical Sciences, Statistics Specialization, Utah State University, Logan, Utah. MS in Statistics, Utah State University, Logan, Utah. MS in Applied Economics, Utah State University, Logan, Utah. BA in International Economic Relations, University of World Economy and Diplomacy, Tashkent, Uzbekistan. Research interests include imputation of missing values, nonparametric statistics, computational statistics, biostatistics, and statistical consulting.