Statistical Analysis Of Genetic Sequence Variants In Whole Exome Sequencing Data From Patients With Prostate Cancer

Kelvin Ofori-Minta

University of Texas at El Paso

Follow this and additional works at: https://scholarworks.utep.edu/open_etd

Part of the Bioinformatics Commons, and the Statistics and Probability Commons

Recommended Citation
https://scholarworks.utep.edu/open_etd/3309

This is brought to you for free and open access by ScholarWorks@UTEP. It has been accepted for inclusion in Open Access Theses & Dissertations by an authorized administrator of ScholarWorks@UTEP. For more information, please contact lweber@utep.edu.
STATISTICAL ANALYSIS OF GENETIC SEQUENCE VARIANTS IN WHOLE EXOME SEQUENCING DATA FROM PATIENTS WITH PROSTATE CANCER

KELVIN OFORI-MINTA

Master’s Program in Statistics

APPROVED:

________________________________________
Ming-Ying Leung, Ph.D., Chair

________________________________________
Amy Wagler, Ph.D.

________________________________________
Lela Vukovic, Ph.D.

________________________________________
Jonathon E. Mohl, Ph.D.

______________________________
Stephen L. Crites, Jr., Ph.D.
Dean of the Graduate School
To my advisor, Dr. Ming-Ying Leung. My parents Isaac and Margaret who taught me that even the largest and difficult tasks can be accomplished, if it is done one step at a time. My lovely siblings Isaac, Samuel, Basma, Dennise and to the Turkson family.
STATISTICAL ANALYSIS OF GENETIC SEQUENCE VARIANTS IN WHOLE EXOME SEQUENCING DATA FROM PATIENTS WITH PROSTATE CANCER

by

KELVIN OFORI-MINTA, BSc.

THESIS
Presented to the Faculty of the Graduate School of
The University of Texas at El Paso
in Partial Fulfillment
of the Requirements
for the Degree of

MASTER OF SCIENCE

Department of Mathematical Sciences
THE UNIVERSITY OF TEXAS AT EL PASO
August 2021
Acknowledgements

I am forever grateful to the creator of the universe and the giver of life for granting us the existence to conduct this study. I express my sincere gratitude to my supervisor, Dr. Ming-Ying Leung, Director of Bioinformatics and Computational Science Program at The University of Texas at El Paso, for her assistance, patience and support at every stage of this thesis dissertation, her unwavering support and guidance afforded me enough motivation to take this bold step in Statistical Bioinformatics. In no particular order, I would also like to thank my thesis committee panel members viz, Dr. Jonathon E. Mohl, Dr. Amy Wagler and Dr. Lela Vukovic for their invaluable co-operation and support towards the success of this thesis dissertation. Worth mentioning is the indispensable support of Bofei Wang for his comments and suggestions towards this work.

Finally, I appreciate my family and friends for their love and moral support through-out my educational journey.
Abstract

A single variation in the genetic sequence within the DNA of an organism could easily lead to beneficial, detrimental or neutral effects. Most often than not, these effects are detrimental than beneficial. While many biomedical and bioinformatics studies have been conducted to determine the genetic cause of prostate cancer (PrCa) which is still the second leading cause of cancer related death among men in the United States. An appreciable effort in statistical bioinformatics researches has been directed towards this aim. Through statistical analyses of a set of whole exome sequencing data from patients with PrCa obtained via The Cancer Genome Atlas (TCGA), this work seeks to augment current efforts by employing both partitional and hierarchical clustering methods to find groups of highly correlated genes associated with PrCa. The scan statistics were also used to identify possible mutational hotspots on those genes containing high numbers of genetic sequence variants. Our results indicated three pairs of variants that are constantly grouped together by multiple clustering methods. Furthermore, we found small regions on several genes containing unusually high concentration of sequence variants, which might suggest mutational hotspots that predispose individuals to PrCa. These results will be reported to biomedical scientists for further bioinformatics analyses and wet lab studies.

Key words: Statistical bioinformatics, Partitional clustering, Hierarchical clustering, Scan statistics, The Cancer Genome Atlas (TCGA), Prostate cancer, Whole exome sequencing (WES).
# Table of Contents

Acknowledgements ........................................................................................................ v

Abstract ......................................................................................................................... vi

Table of Contents .......................................................................................................... vii

List of Tables .................................................................................................................. ix

List of Figures ................................................................................................................ x

Chapter 1: Introduction .................................................................................................. 1
  1.1 Background and Significance ................................................................................. 2
  1.2 Motivation for the Study ....................................................................................... 3
  1.3 Research Objectives and Overview ...................................................................... 3

Chapter 2: Literature Review ......................................................................................... 4
  2.1 Whole Exome Sequencing on Prostate Cancer .................................................... 4
  2.2 Clustering Methods in Cancer Studies ................................................................ 5
  2.3 Scan Statistics in Cancer Studies ......................................................................... 6

Chapter 3: Materials and Statistical Methods ............................................................... 7
  3.1 Data Description .................................................................................................. 7
  3.2 Data Analysis Workflow ...................................................................................... 11
  3.3 Exploratory Data Analysis .................................................................................. 12
  3.4 Clustering Frequent Variants by Their Occurrence Patterns in Patients .......... 14
    3.4.1 Partitional Clustering ................................................................................... 14
    3.4.2 Hierarchical Clustering ............................................................................... 15
    3.4.3 Dissimilarity Measure ............................................................................... 16
    3.4.4 Assessing Clustering Tendency .................................................................. 18
    3.4.5 Choosing Optimal Number and Quality of Clusters .................................. 19
  3.5 Identifying Genomic Regions with high Variant Concentration ...................... 20
    3.5.1 Simulated Scan Statistics ........................................................................... 21
    3.5.2 Theoretical Scan Statistics ......................................................................... 22

Chapter 4: Results and Discussion .............................................................................. 23
  4.1 Exploratory Results ............................................................................................. 23
4.1.1 Statistical Summary of Tumor Variants .......................................................23
4.2 Correlations ...........................................................................................................26
4.3 Dissimilarity Measure and Cluster Tendency .......................................................30
4.4 Clustering Results .................................................................................................32
    4.4.1 Partition Around Medoids – PAM .................................................................32
    4.4.2 Hierarchical Agglomerative Clustering ...........................................................36
    4.4.3 Hierarchical Divisive Clustering ....................................................................40
4.5 Gene Regions with High Variant Concentration .................................................45

Chapter 5: Conclusion and Future Work ....................................................................48
    5.1 Conclusion ...........................................................................................................48
    5.2 Future Research ..................................................................................................49

References .....................................................................................................................50

Appendices .....................................................................................................................55

Curriculum Vita ..............................................................................................................93
List of Tables

Table 4.1: Distribution of GSV counts on SNPs ................................................................. 24
Table 4.2: Five-Point Summary with Mean and Standard Deviation of GSV counts per patient 26
Table 4.3: Summary Statistics and Five-Point Summary of Number of Patients per GSV........ 27
Table 4.4: Highly Correlated Frequent Variant Pairs ............................................................. 29
Table 4.5: Gower cluster membership with PAM ................................................................. 34
Table 4.6: Correlation cluster membership with PAM ........................................................... 35
Table 4.7: Agglomerative hierarchical cluster membership with Gower dissimilarity ............. 38
Table 4.8: Agglomerative hierarchical cluster membership with correlation dissimilarity........ 39
Table 4.9: Divisive hierarchical cluster membership with Gower dissimilarity ..................... 42
Table 4.10: Divisive hierarchical cluster membership with correlation dissimilarity ............. 43
Table 4.11: Summary of clustered correlated variant pairs by clustering methods............... 44
Table 4.12: Gene regions with high concentrations of GSVs identified by scan statistics using window length of 0.01. ........................................................................................................ 46
List of Figures

Figure 3.1: VCF file showing metadata section ................................................................. 8
Figure 3.2: VCF file showing data lines ........................................................................... 8
Figure 3.3: Processed data file ......................................................................................... 9
Figure 3.4: Workflow ......................................................................................................... 11
Figure 3.5: Scan Statistic ................................................................................................. 21
Figure 4.1: Distribution of GSVs on chromosomes including chromosomes X and Y ........ 23
Figure 4.2: Variant density (number of GSVs per megabase) on chromosomes including X and Y ......................................................................................................................... 24
Figure 4.3: Distribution of GSV counts for the 12 different types of SNPs. Each pair of bases on the x-axis stands for the mutation from the first base to the second ................................................. 25
Figure 4.4: Proportions of GSVs found in dbSNP ............................................................. 26
Figure 4.5: Five-number summaries for the GSV and patient distributions in the 91FV dataset. 27
Figure 4.6 Correlogram of variants with more than 1% relative frequency of occurrence (91 FV) ................................................................................................................................. 28
Figure 4.7: VAT diagram with Gower dissimilarity measure ........................................... 30
Figure 4.8: VAT diagram with correlation dissimilarity measure ........................................ 31
Figure 4.9: Cluster plot with Gower dissimilarity shown on first two PCC ......................... 32
Figure 4.10: Cluster plot with correlation dissimilarity shown on first two PCC ............... 33
Figure 4.11: Agglomerative hierarchical cluster dendrogram with Correlation dissimilarity ...... 36
Figure 4.12: Agglomerative hierarchical cluster dendrogram with Gower dissimilarity ............ 37

Figure 4.13: Divisive hierarchical cluster dendrogram with correlation dissimilarity ............. 40

Figure 4.14: Divisive hierarchical cluster dendrogram with Gower dissimilarity.................. 41

Figure 4.15: Sliding window plot for gene KIR2DS3 (length = 123645 bases) with a window size of 0.01 on the gene normalized to unit length. ................................................................. 45

Figure 4.17: Grided sliding window plots of genes with significant clusters using a window size of 0.01. The highest peak in each plot gives the scan statistic value. ............................................. 47
Chapter 1: Introduction

Deoxyribonucleic Acid (DNA) - is an organic chemical and complex molecular structure found in eukaryotic and prokaryotic organisms. The section of the DNA that codes for the synthesis of a specific protein is called a gene – the functional unit of hereditary, whereas the complete set of genetic material present in an organism is collectively known as a genome. DNA sequencing involves identifying the order of the four chemical building blocks (nucleotide bases), namely Adenine (A), Thymine (T), Cytosine (C) and Guanine (G), that make up the DNA molecule. This sequencing data can inform researchers which section of the DNA contain genes or genetic instructions likely to cause diseases. Within the double helical structure of the DNA molecule, one strand is held to another by hydrogen bonds in which A pairs with T and C pairs with G. This specific pairing is the stable underlying framework by which cells divide and experimental DNA sequencing is done. (Britannica, 2020). The variation that occurs when there is an alteration in a single nucleotide base at a single position in a DNA sequence in at least 1% of all subjects is thought of as a single nucleotide polymorphism (SNP).

We can think of genetic sequence variants (GSVs) as the mutation of the DNA bases in our genes, this can only be done by figuring out the DNA sequence with the use of gene sequencing technologies. These mutations sometimes change the gene’s DNA make-up, but sometimes maintains the function of the protein produced by that same gene. In other cases, gene mutations change the initial protein to a new protein made by the gene, where this new protein might have abnormal protein structures or be produced at reduced or elevated levels. When a mutation changes a protein that plays a critical function in the body, it can disrupt normal development or cause a medical condition (Medline Plus Genetics, 2021). Therefore, a mutated DNA provides enough evidence to suspect a biological alteration on the synthesis of protein responsible for carrying out biological and enzymatic functions such as cell growth, digestion, reproduction etc.
1.1 **Background and Significance**

Prostate cancer (PrCa) occurs when there is an uncontrolled outgrowth of cells within the prostate gland of the human reproductive system. Located in front of the rectum and just beneath the bladder, the prostate is a walnut shaped gland, enclosing the urethra that is responsible for the production of fluids that nourishes the semen.

Statistical updates of the Global Cancer Observatory (GCO), a sub-division of the World Health Organization (WHO), by Sung et al. (2021) attributes 7.3% (1,414,259) of all reported cancer cases to PrCa. Out of these reported cases, 3.8% (375,304) led to death, while 3.5% are recorded to survive across both sexes and ages. In the United States of America, 13.73% (34,130) deaths are recorded from 248,530 reported cases of PrCa, known to be second leading cause of cancer related deaths, with the exception of lung cancer, and it is highly likely to occur in the lifetime of 1 out of 8 men. (American Cancer Society, 2021)

According to current publication by the Cancer Prevention and Control, a sub-division of Centers for Disease Control and Prevention (CDC), treatments and diagnosis of PrCa will vary from one individual to the other, thus there exist no singled-out prescription. While terminal stages of PrCa are very lethal and difficult to treat with a 5-year survival rate of approximately 0.28, and 0.98 for early stages of PrCa, successful treatment depends on the individual’s immune system coupled with early diagnosis and administration of varied treatment methods by a prostate health expert (active surveillance) among other treatment methods.

With a pin-point focus on genetic risk factors of PrCa, such as acquired genetic alterations during an individual’s lifetime, it is imperative to examine the DNA of affected subjects to discover, understand and describe the impact of genetic variations to the evolvement of prostate tumors and obtain relevant statistical and genetic insights. These are essential for biomedical researchers to develop concrete healthcare routines capable of mitigating the burden of cancerous prostate cells.
1.2 Motivation for the Study

There have been several extensive and independent studies on the analysis of whole exome sequencing (WES) data to obtain meaningful scientific conclusions and biomedical recommendations on some characteristic genomic features susceptible to the dangers of PrCa. Summarized findings from a few independently published studies (Raspin et al., 2021; Schaid et al., 2021, Gupta et al., 2020; Rand et al., 2016) provided compelling results in each respective study that, there actually exist numerous and different genes on different chromosomal regions significant enough to predispose an individual or a selected cohort to the hazards of PrCa. Moreover, Xu et al. (2013) confirmed the findings of Ewing et al. (2012) that a rare and recurrent variant (G84E) in HOXB13 was indeed associated with PrCa.

1.3 Research Objectives and Overview

The goal of this thesis is to mine a WES dataset from 503 patients diagnosed with PrCa with two specific aims:

(i) Identify individual GSVs or groups of correlated GSVs likely to be associated with PrCa.
(ii) Identify genes containing high density of PrCa-related GSVs and search for mutational hotspots on these genes.

Our results will be reported for other researchers to conduct further experiments in the wet lab to better understand the biological implications of the identified GSVs on PrCa. The next chapter of this work reviews various statistical methods that have been used in the analysis of cancer studies. This is followed by a description of methodologies and results for discussions in Chapters 3 and 4 respectively. This work then finishes off with conclusions and recommendations for future studies in Chapter 5.
Chapter 2: Literature Review

This chapter reviews various literatures concerning the applications of WES data analysis, clustering methods and scan statistics in PrCa, or cancer studies at large.

2.1 Whole Exome Sequencing on Prostate Cancer

Proteins are produced according to the instructions provided by bits and pieces of the human DNA, these pieces are called exons which are collectively known as exome, WES uses next generation sequencing reads that can be mapped to a specific location on a reference genome to identify mutations within a sample. Since the majority of known genetic mutations that cause diseases occur in exons, whole exome sequencing is thought to be an efficient method to identify possible disease-causing mutations (Medline Plus Genetics, 2021). Essentially, WES had proven to be an effective process to dissect the coding sections to discover interesting variants connected to a particular disorder within a human genome (Biesecker., 2010).

A pilot study on PrCa data obtained from TCGA portal by Zeng et al. (2021) indicated that, an increased expression of the gene UGT2B4 was associated with localized PrCa. They suggested further studies on the functional effects on a novel gene pair (BTBD7 - SLC2AS) to help evolve medications for PrCa. Gupta et al. (2020) reported a number of variants in DNA repair genes such as TP53, helicas, BRCA to be associated with benign forms of PrCa. Schaid et al. (2021) discovered eleven (11) massive genes (ATM, BRCA, HOXB13, FAM111A, EMSY, HNF1B, KLK3, MSMB, PCAT1, PRSS3, TERT) previously published to be associated with PrCa, as well as ten (10) novel genes (PABPC1, QK1, FAM114A1, MUC6, MYCBP2, RAPGEF4, RNASEH2B, ULK4, XPO7, and THAP3) not reported before. Out of which PABPC1 and ULK4 were primarily associated with aggressive forms of PrCa when they sequenced exomes of a PrCa family. WES datasets can be publicly assessed via the Genomic Data Commons (GDC) of The Cancer Genomic Atlas (TCGA). This platform links-up and combine all respective normal and tumor samples of
several cancer types at a molecular level and specifies over tens of thousands of primary cancer samples readily available for use by researchers worldwide.

2.2 Clustering Methods in Cancer Studies

Basically, clustering defines an unsupervised machine learning technique that is able to detect and group a bunch of observations or variables that possess similar or dissimilar characteristics. This is possible when respective distance measures are appropriately fed to the clustering algorithm, these distance measures serve as a yardstick to decide the proximity of pairwise distances between component data points. Partitional clustering and hierarchical clustering are the basic forms of segmentation methods used in clustering, the former comprises of k-means and partition around medoids (PAM) also called k–medoids algorithms (Macqueen, 1967; Kaufman & Rousseuw, 1990), while the latter is subdivided into agglomerative and divisive methods of hierarchical clustering algorithms (Murtagh & Contreras, 2011, 2017). Partitional clustering techniques usually groups observations into k–clusters, while hierarchical clustering usually produces a cluster-tree dendrogram (hierarchy) of clusters which does not require pre-specification clusters.

Akamine et al. (2020) demonstrated the use of the hierarchical clustering algorithm to diagnose PrCa in an easy and interpretable by clustering images which distinguishes between tumor and normal tissues of their subjects. For early detection of prostate cancer, Kavuri and Liu (2014) developed a hierarchical cluster method to improve upon imaging results, by reconstructing and dividing the human prostate region into geometric clusters to find regions of abnormalities. Kakushadze and Yu (2017) applied k-means clustering technique to extract signature cluster structures from genome data. They found the projection of three cancer types (liver, lung and cell carcinoma) cancers that do not possess well defined clusters, while two clusters exhibited great number of within-cluster correlations while eleven other cancers indicating a common defined
structure. This approach suggested a new procedure for clustering cancer structures from genomic data.

2.3 Scan Statistics in Cancer Studies

Customarily, scan statistics have been used to comb through time and space to obtain convincing evidence of significant clusters of observations. In recent times, they have gained much recognition as a method for genomic data analyses (Leung, 2017). The one-dimensional scan statistic is basically used to test whether a point process is purely random or there is an unusual aggregation of points.

With the variation in superfund sites, Amin et al. (2018) obtained convincing significant geographic locations with superfund sites of elevated levels of cancer incidence accompanied by minority populations, when they used the scan statistic feature from the disease surveillance software package SaTScan to identify spatial locations with relative risk of cancer clusters alongside superfund site count and density. They also completed and verified their findings using the surveillance software FlexScan, to determine significant clusters of the three most common types of pediatric cancer incidence. Amin et al. (2014) employed spatial scan statistics with a space-time analytical tool, SaTScan™ with combination of a non-parametric space-time interaction test and a multivariate spatial cluster analysis, they obtained a significant cluster for each cancer type, as well as two significant clusters for the combined pediatric cancer types in Florida. This study was a build-up on a previous study with the incorporation of extensional years of study.
Chapter 3: Materials and Statistical Methods

In this chapter, we will introduce the data repository and all materials needed for the analysis section, as well as a thorough description of all statistical techniques to be deployed for this analysis. The software used for majority of analyses in this work is the R Software version 4.0.2 embedded in R-Studio. However, for a clear and easy-to-read correlogram, I used the Scientific Python Development Environment (SPYDER) version 4.0.1, an open-source python integrated development environment (IDE) via anaconda package manager distribution.

3.1 Data Description

The main data repository for this work is The Cancer Genomic Atlas (TCGA) platform, released via the National Cancer Institute (NCI) Genomic Data Commons (GDC) which applies high technology throughput sequencing and bioinformatics methods to designate over tens of thousands of primary cancer and matched normal samples across several cancer types. Detailed description and information regarding this platform can be found directly via The Cancer Genome Atlas Program - National Cancer Institute. This PrCa data is publicly accessible via the variant call format (VCF), treated as protected data. Each of these VCF files contains a paired sample from the same patient with PrCa: one extracted from the primary tumor tissue and the other from the blood-derived normal sample. All 503 VCF files were downloaded and stored on the bioinformatics computer network for further processing. VCF is a commonly used format for storing and reporting identified genetic sequence variants (GSVs). In general, GSVs refer to changes or mutations on the DNA. The VCF file is composed of meta-information lines (Figure 3.1), a header line and data lines (Figure 3.2). Meta-information lines describe the date of creation and provide specific explanations about entries in data section. The header line displays eight (8) required fields and could also include additional columns. Each line in data section represents a variant, containing all information specified in the header line.
## VCF file showing metadata section

<table>
<thead>
<tr>
<th>CHROM</th>
<th>POS</th>
<th>ID</th>
<th>REF</th>
<th>ALT</th>
<th>QUAL</th>
<th>FILTER</th>
<th>INFO</th>
<th>FORMAT</th>
<th>TUMOR</th>
<th>NORMAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr1</td>
<td>3493967</td>
<td>rs7511631G</td>
<td>G</td>
<td>A</td>
<td>.</td>
<td>PASS</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 1/1:8:4:4, 0/0:10:10,0:29:0:4:0:29:29,0:32:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>6219437</td>
<td></td>
<td>G</td>
<td>C</td>
<td>.</td>
<td>PASS</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:31:21:0, 0/0:29:29,0:32:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>16632199</td>
<td>rs9662561G</td>
<td>A</td>
<td>T</td>
<td>.</td>
<td>Tier1</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:22:18,0, 0/1:18:18,0:35:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>24671670</td>
<td></td>
<td>G</td>
<td>A</td>
<td>.</td>
<td>Tier1</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:15:13,0, 0/1:24:24,0:34:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>37721709</td>
<td></td>
<td>G</td>
<td>T</td>
<td>.</td>
<td>Tier2</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:23:21,0, 0/1:19:19,0:31:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>47074876</td>
<td>rs1466546C</td>
<td>T</td>
<td>.</td>
<td>.</td>
<td>Tier1</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:12:8:4,0, 0/1:16:16,0:33:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>63436758</td>
<td></td>
<td>C</td>
<td>T</td>
<td>.</td>
<td>Tier1</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:13:11,0, 0/1:26:26,0:34:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>74369623</td>
<td>rs1998072C</td>
<td>A</td>
<td>.</td>
<td>.</td>
<td>PASS</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:28:17,0, 0/1:56:56,0:28:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>74600000</td>
<td></td>
<td>G</td>
<td>A</td>
<td>.</td>
<td>PASS</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:10:7:3,0, 0/1:12:12,0:34:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>77956753</td>
<td>rs7535327A</td>
<td>T</td>
<td>.</td>
<td>.</td>
<td>Tier1</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:47:38,0, 0/1:51:51,0:30:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>92245463</td>
<td></td>
<td>T</td>
<td>A</td>
<td>.</td>
<td>Tier5</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:21:19,0, 0/1:16:16,0:31:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>1.17E+08</td>
<td>G</td>
<td>T</td>
<td>.</td>
<td>.</td>
<td>Tier1</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:15:51,0, 0/1:42:42,0:34:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>1.18E+08</td>
<td>A</td>
<td>C</td>
<td>.</td>
<td>.</td>
<td>PASS</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:16:46,0, 0/1:32:32,0:33:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>1.21E+08</td>
<td>C</td>
<td>A</td>
<td>.</td>
<td>.</td>
<td>Tier1</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:15:13,0, 0/1:11:11,0:34:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>1.53E+08</td>
<td>C</td>
<td>A</td>
<td>.</td>
<td>.</td>
<td>Tier2</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:23:21,0, 0/1:33:33,0:31:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>1.77E+08</td>
<td>G</td>
<td>A</td>
<td>.</td>
<td>.</td>
<td>PASS</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:49:40,0, 0/1:72:72,0:28:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>1.82E+08</td>
<td>rs3773824C</td>
<td>T</td>
<td>.</td>
<td>.</td>
<td>PASS</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:110:71,0, 0/1:92:92,0:30:0:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chr1</td>
<td>1.86E+08</td>
<td>T</td>
<td>A</td>
<td>.</td>
<td>.</td>
<td>PASS</td>
<td>SOMATIC</td>
<td>GT:DP:AD: 0/1:17:10,0, 0/1:13:13,0:32:0:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Computational and statistical approaches were used to preprocess and analyze prostate cancer VCF files. (Wang et al., 2020) outlines all basic procedures and instructions necessary to obtain the processed data for analysis, which includes filtering and determination of variant type, gene name, genomic region and change type as well as merging all 503 files. Figure 3.3 displays a section of the final data file. The final file contains at least 16 columns with information on the chromosome number, GSV position, gene name, reference and mutated nucleotides, genomic region, change type, GSV identification number in the database dbSNP if known, and an extra 503 columns to record the occurrences of a GSV in the tumor tissues of these 503 PrCa patients. There are 93760 rows with each row being a data line which specifies the location of a single variant on a chromosome and other genomic information that specifies that particular variant from other variants.

![Figure 3.3: Processed data file](image)

Note: The image contains a table of data with columns for chromosome number, left position, right position, reference sequence, gene name, alternate allele, change type, sub11, sub10, sub20, sub00, dbSNP ID, and overlap.
Our data analyses were conducted on the GSVs from 503 PrCa patients’ tumor samples. First, we explored the entire dataset to provide an overview of the GSV statistics. Next, to address the two specific aims of this research, we prepared the data differently to suit each proposed analytical method. For specific aim (i), the GSVs of all chromosomes from each of the 503 subjects with less than 1% relative frequency were screened out and variants with strictly more than 1% relative frequency, i.e., only GSVs with 6 or more distinct count were retained. This data file will be labelled as 91FV since it contains 91 frequent variants. Using this dataset, we assessed the five-number summary of GSV counts per patient and the number of patients per GSV. We further performed correlation and various cluster analyses based on this same data file fixing the issue of data sparsity. The final part of our analysis for specific aim (ii), is to identify chromosomal or gene regions with high variant concentration, when variants with less than sixty (60) distinct GSVs are filtered out.
3.2 Data Analysis Workflow

The methods and statistical analysis conducted are outlined in the flowchart below.

![Data Analysis Workflow Diagram](image-url)

**Figure 3.4: Workflow**

Methods and Statistical Analysis

Data Structure, Description & Pre-processing

Preliminary Analysis

- Exploratory Statistics
- Correlation
- Clustering

Further Analysis

- Scan Statistics

Input Data:
- Transcript Variant Distribution
- 91 FV
- Chromosomes (> 60 Distinct Variants)

Underlying trend of GSVs in PrCa patients.

(Specific Aim 1)
Clusters of correlated GSVs likely to be associated with PrCa.

(Specific Aim 2)
Mutational hotspots on genes with high concentration of PrCa related GSVs.
3.3 Exploratory Data Analysis

We begin the preliminary exploratory analysis with summarized statistics of our subjects and variants using Tukey’s five-number summary (Tukey, 1977), to display the five standard summary statistics of GSV counts per patient and the number of patients per GSV with boxplots. A lollipop chart will be used to analyze the count of all twelve (12) different types of SNPs and GSV counts across all chromosomes of our subjects, including chromosomes X and Y. A hybrid between a bar chart and dot plot, the lollipop chart produces attractive and clear visualizations with the vertical bars indicating a specific category and the circle on top represents the count or value of that category. In displaying large number of similar values, lollipop charts are preferred to bar charts since they prevent the “moire pattern” - a visual discomfort when viewing superimposed stacked charts (Weitz, 2020). Nevertheless, the density of variants on each chromosome will be displayed with a simple density bar chart. The doughnut chart will be used to explore “part-to-whole” relationships in the categorical database SNP (Db-SNP) variable. Doughnut chart is an improved version of the traditional pie-chart, which is known to eliminate trickiness in reading traditional pie charts and also provides simple and compact representations as compared to traditional pie charts. Essentially, it will be used to obtain clear and concise representation of proportions of known and unknown database resources of SNPs.

Identifying highly correlated variant pairs is an important step towards obtaining groups of variants that tend to occur together in PrCa patients. Since it would not be very meaningful to assess correlation for those variants that are hardly found in patients, we screened out all the GSVs with less than 1% inclusive relative frequency and retained only 91 GSVs with 6 or more patient count to construct a correlogram. In this vein, we seek to obtain information about the magnitude and direction of the association (correlation) between these 91 frequent variants (FV) by calculating the Pearson product moment correlation (PPMC), Pearson correlation coefficient (r), mathematically expressed as:
\[ r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n \sum x^2 - (\sum x)^2] \ [n \sum y^2 - (\sum y)^2]}} \]

Correlation coefficient values can range from -1 to +1, where +1 indicates a perfect positive relationship, -1 indicates a perfect negative relationship, and a 0 indicates no relationship exists.

PPMC is incorporated in R via the “cor()” function which takes in any two numeric vectors (x, y) i.e., any two FVs with equal length and computes the correlation co-efficient only, while “cor.test()” function takes in same data entries as the “cor()” function and returns additional details such as the p-value, confidence interval etc., referring to appendix C, I specifically wrote a function to extract all correlations of FVs greater that 0.3 excluding all main diagonals, and used the “cor.test()” function on each respective correlated variant, which provided both correlation coefficient (r), and the significance level (p-value) of the correlation. These two functions return the PPMC by default without specifying the “Pearson” method.

In a similar manner, the correlogram can be obtained with R software, via the “corrplot()” function in the corrplot package but due to clarity and visibility issues using the entire 91 by 91 correlation matrix, I deployed python’s pandas dataframe.corr() function with the pearson method which produced a correlation matrix by computing the pairwise correlation of all columns in the dataframe and then visualized the correlation matrix using “sns.heatmap()” via the seaborn library. Correlation matrices and co-efficient obtained from both softwares were consistent but the python version produced a clearer and more visible display resolution of the visualized correlogram of the 91 variants.
3.4 Clustering Frequent Variants by Their Occurrence Patterns in Patients

To find a group or “a clique” of highly correlated variants, we will consider two main clustering approaches. Partitional clustering approach via partition around medoid (PAM) method and hierarchical clustering approach using the agglomerative and divisive methods.

3.4.1 Partitional Clustering

Commonly used partitional clustering techniques include but are not limited to k-means clustering and k-medoids (PAM) clustering whose algorithms have been duly explained (Macqueen, 1967; Kaufman & Rousseuw, 1990). For their efficiency’s sake, PAM will be preferred since it possesses numerous advantages over the traditional k-means clustering.

Before describing these clustering methods, we have to consider the issue of data sparsity among our 91FV dataset. We dealt with this issue by eliminating all subjects with zero variant count, in this case 148 out of 503 subjects recorded zero variants counts. These subjects will have no variance on our results thus making it difficult for machine learning algorithms to produce meaningful results. These kinds of predictors are not only non-informative, they could, in certain situations, break some models we may want to fit to our data (Thiagogm, 2014). The basic idea behind clustering is to define clusters such that the total intra-cluster variation (known as total within-cluster variation) is minimized (Kassambara, 2017). PAM being robust and a better alternative, outperforms all partition-based algorithms (Reynolds et al., 2006), will be used to classify the 91 FVs into a set of pre-specified k groups (i.e., k clusters) based on a dissimilarity measure. This clustering method is almost similar to k-means clustering with the exception that, cluster representatives chosen are real data points known as medoids, in this case a medoid will be a real data object from our 91 FVs which is a representative exemplar of members in that specific cluster, also PAM allows for the flexibility to use different dissimilarity measures.

To determine how well the partitional clusters are structured, we call out the average silhouette width value for the entire clusters and silhouette widths (SW) for individual clusters.
A good confirmatory resource to determine this can be found in (Kaufman & Rousseuw, 1990). The PAM method as explained by (Kaufman & Rousseuw, 1990) works in two main sections, namely the Build Phase and Swap Phase. PAM works with a matrix of dissimilarity, which were fed with the Gower distance and correlation distance for our analyses, via the pam function in the “cluster” package. The resulting clusters generated by the two different dissimilarity measures were compared. Since the 91FV data file is a huge, multi-dimensional data set, it is of statistical prudence to perform a principal component analysis (PCA) on the whole data matrix to obtain a reduced matrix of principal components, where clustered variants are plotted according to the first two principal components, the R function “fviz_cluster” from the “factoextra” package automatically visualize cluster results by returning an elegant “ggplot” of clusters based on first two principal components coordinates (PCC) labelled via the x and y axes (Kassambara, 2017).

3.4.2 Hierarchical Clustering

In addition to the partition clustering methods mentioned above, the hierarchical cluster analysis (HCA) will be used to segment objects based on their similarities and or differences. The agglomerative subdivision of HCA initially considers each variant as a cluster of its own (leaf). Then, the most similar variants or clusters are successively and iteratively merged until there is just one huge (root) cluster (Murtagh & Contreras, 2011; Murtagh & Contreras, 2017). Divisive clustering of HCA, is an inverse of agglomerative clustering, which begins with the root, in which the entire variants are considered to be one cluster, then the most dissimilar variants are successively dissociated until all variants are rightly grouped and clustered. (Kaufman and Rousseeuw, 1990). A visualized display and thorough explanation of both algorithms can be accessed directly via Hierarchical Clustering Algorithms - The Essentials.

The agglomerative procedure was implemented via the “hclust” function in base R, by specifying “average” linkage method and divisive clustering via the R function “diana” from the “cluster” package on the 91FV data. The results of both HCA methods will be a cluster tree –
dendrogram, which represents a tree-based segmentation of our 91FV in a multi-level hierarchy of clusters, where a cluster at each level is joined to form a cluster at the subsequent level. To implement HCA via “hclust” and “diana”, our computed dissimilar matrices need to be converted into a “dist” object (Mardia et al., 1979). The linkage function created with the “dist” results groups similar variants into clusters which are then connected to form hierarchical tree diagram represented as dendrograms. While traditional and basic linkage methods such as “ward”, “single” and “complete”, “mcquitty” and “centroid” linkages exist to find dense, closely-packed, similar (“friends of friends”) clusters, our chosen linkage method will be the “average” linkage method, because of its tendency to find agglomerative clusters with a middle-ground between the qualities possessed by the listed traditional linkage methods of HCA. (Manning et al., 2009). Conveniently, deciding the number of clusters produced will be specified by the level at which we cut the cluster tree dendrogram and vice versa. A representative dendrogram can be accessed directly at Cluster Tree Dendrogram.

3.4.3 Dissimilarity Measure

All clustering algorithms described above required some methods for computing the dissimilarity or distance measure between each pair of GSVs. The computed distances are then organized into a dissimilarity or distance matrix. The choice of distance measure is a critical step in clustering because it defines and influences the shape of the clusters as well as the number of members in each cluster. Classical methods for distance measures are “Euclidean” and “Manhattan” distances. However, other dissimilarity measures, such as several correlation-based distances including the Pearson correlation, Eisen cosine correlation, Spearman correlation, and Kendall correlation, are also commonly used (Kassambara, 2017).

For the purpose of this work, we considered the Gower distance and the Pearson correlation distances as our two main dissimilarity measures. The correlation-based distance is defined by subtracting the Pearson correlation coefficient from one (1):
Correlation-based measure would judge any two variants to be identical if their features are highly correlated. Perfect correlation implies that the distance between any two variants is zero. Pearson’s correlation is sometimes known to be quite sensitive to outliers but would not hurt our cluster analysis because the correlation is over many (91) GSVs and also outliers might possess important characteristics needed for clustering.

The Gower distance is suitable for mixed data types including purely numeric and pure non-numeric data. (McCaffrey, 2020). This distance is an indicator of how two entries \((i, j)\) in a variable, \(k\) are dissimilar. Measured on a numbered scale between 0 (identical) and 1 (different), data entries might contain combinations of logical, numerical, categorical or text data. A vivid explanation of the different forms of the metric is given in the original paper (Gower, 1971), where any two observations \((i, j)\) may be compared with a variable \((k)\), and assigned a score \(S_{i,j,k}\), based on their similarity and dissimilarity. The similarity between \(i\) and \(j\) is defined as the mean score taken over all possible comparisons, \(v\). For an all-numeric data matrix, a simple and easier illustration of the Gower distance, \(d_{(a,b)}\) defined in (McCaffrey, 2020) is the averaged numerical feature, \(Nf_{(i)}\) obtained from all 91FVs, where a numerical feature between any two observations, \(Nf_{(a,b)}\) of a variant is the absolute value of their difference divided by the range. This is illustrated as:

\[
d_{(a,b)} = \frac{\sum_{i=1}^{n} Nf_{(a,b)}}{n}
\]

\[
Nf_{(a,b)} = \frac{|a - b|}{range}
\]
Therefore, the Gower distance metric will always be a value between 0 indicating the two items have identical features and 1 indicating that the two items are far apart in respect to the dataset.

3.4.4 Assessing Clustering Tendency

A clustering algorithm will always try to find clusters within any dataset, irrespective of its clustering feasibility. The process of assessing cluster tendency is to investigate if our data indeed contains non-random structures of meaningful clusters, and if that is the case, how many clusters are possible. Cluster tendency assessment was done on the results of our dissimilarity measure. In this section, we will describe two methods for determining partition clustering tendency namely, the Hopkins statistic and a visual method called visual assessment of cluster tendency (VAT). The Hopkins statistic is used to assess the clustering tendency of a dataset by measuring the probability that a given dataset is generated by a uniform data distribution (Lawson and Jurs, 1990). It tests the spatial randomness of the data with the formula defined as:

\[ H = \frac{\sum_{i=1}^{t} B_i}{\sum_{i=1}^{t} A_i + \sum_{i=1}^{t} B_i} \]

Let D be our data set, whose clustering tendency needs to be evaluated. 

**T** is a set of **t** data point from **D**

**A**\(_1, \ldots, \ A\_t\) are distances of points in **T** to their nearest neighbors in **D**

**R** is a set of **t** randomly generated data points

**B**\(_1, \ldots, \ B\_t\) are distances of points in **R** to their nearest neighbors in **D**

A value of about 0.5 for \( H \) is an indication that \( \sum_{i=1}^{t} B_i \) and \( \sum_{i=1}^{t} A_i \) are close to each other, thus the dataset is uniformly distributed. The Hopkins statistic is iteratively obtained with a 0.5 threshold. That is, \( H < 0.5 \) indicates that data is unlikely to have statistically significant clusters. (Kassambara, 2017). On the other hand, if the value of \( H \) is close to 1, then we can reject the null hypothesis and conclude that the dataset is significantly clusterable. VAT is a visual approach for assessing cluster tendency. It provides a square image which shows pair-wise dissimilarity.
information about the data points. Objects are reordered in such a way that highlights potential clustering in a form of visible blocks along the diagonal in a VAT image. The VAT algorithm computes the dissimilarity matrix (DM) for our 91FV dataset using our choice of distance measures, re-orders the DM such that similar variants are close to one another to form an ordered dissimilarity matrix (ODM), and outputs an ordered dissimilarity image (ODI) at the end. The clustering tendency is provided in a visual form by the number of square shaped dark blocks along the diagonal in a VAT image. A thorough explanation of cluster tendency assessment is also found in (Kassambara, 2017).

In the case of HCA, cluster validity will be examined by how the cluster tree diagram perfectly reflects and preserves the original pairwise characteristics of our distance measure, this will be done with the cophenetic correlation coefficient (CPCC), which measures the correlation between the cophenetic distance and the original distance measure, where strong correlation (higher CPCC value) is an indication of a valid cluster (Saraçli et al., 2013). The cophenetic distance matrix introduced by (Sokal & Rohlf, 1962), seeks to measure within-group dissimilarity(heights) at which two observations are initially merged into a single cluster. The divisive coefficient will be used to judge strength of group distinctions in the divisive analysis. (Greenwell, 2020).

3.4.5 Choosing Optimal Number and Quality of Clusters

There are many statistical metrics helpful for choosing the optimal number of clusters $k$, but in general there is no definitive method for determining exact value of $k$. In R, the package “NbClust” provides 30 indices for determining the optimal number of clusters. Two of them - Hubert index and D index are graphical methods. The optimal number of clusters is in their “elbow point” (where the rate of statistics decreases shifts sharply). For the purpose of this work, the number of clusters among the 91FV were judged to be twenty (20), since we desire to have
relatively small clusters with 2 and 6 members. It is worth mentioning that the silhouette metric would also be used to assess the quality of our clusters obtained.

This metric provides a visualization of how well each object lies within its cluster. The silhouette measure, $S$ is defined as

$$S = \frac{b_i - a_i}{\max\{a_i, b_i\}}$$

where $a_i$ is the average distance to all other data points in the clusters and $b_i$ is the minimum of average distance to other clusters, $S$ has a range of $[-1, 1]$, where positive silhouette values are preferable as they indicate that the sample is far away from the neighboring clusters.

### 3.5 Identifying Genomic Regions with High Variant Concentration

A general application of scan statistics is to scan through space or time to detect unusually high concentrations of points or events and determine if such phenomena could reasonably occur by chance or otherwise. This helped us identify gene regions with high variant concentration, treating the occurrence of every distinct variant as an event. Glaz et al. (2001) defined the scan statistics as follows. Given $n$ points distributed over the interval $(0, T)$, if $N_w$ is the largest number of events in a fixed-length of window $w$, then this maximum, $N_w$, is called the scan statistic. For computational purposes, it is more convenient to obtain the scan statistics by taking the maximum number of events in the windows with left-hand end points placed right at the occurrence of each event. Thus, for a set of independent and identically distributed random variables $X_1, ..., X_n$ that are uniformly distributed on the unit interval $(0,1)$, the scan statistic $N_w = \max_{i=1,...,n} N_w(i)$, where $N_w(i)$ is number of events in a window of length $w$, with left end placed at the value of the $i^{th}$ smallest point on the interval

I developed an R code for the sliding window analytical plot, which was used extensively to detect regions of high variant concentration represented as peaks on a graph. This technique
works on the principle that, for a given length of gene, with a fixed window size, we slide the window along the gene length by a step size, that takes its left end point to the next nearest variant continuously, keeping count of variants within the window at each. The maximum variant count at each snapshot of using the sliding window analysis becomes the observed scan statistic \((N_w)\) – “a cluster”, and the position of that “cluster” recorded. This process was repeated for various window sizes for GSVs of a gene on the 15 chromosomes with more than 60 distinct GSV counts.

![Figure 3.5: Scan Statistic](image)

### 3.5.1 Simulated Scan Statistics

To investigate the sensitivity, reactivity and the eventual effects of our scan statistic model to alternative date sets, a self-written R-code was written to intensively simulate the model over one thousand (1000) repeated trials denoted as \(N\), from randomly generated uniform distribution data, this data should be equivalent in dimension to the number of distinct variants on the unit gene length on each chromosome. The simulated scan statistic value, \((S_w)\) was recorded in each trial, when the user specifies a window size \((w)\) and the distinct GSV data to emulate. Again, using a self-written R-code, we count the number of times \(S_w\) was greater than the observed scan statistic \((N_w)\) and denote this number as \(K\). The value obtained when \(K\) is divided by \(N\) becomes the p-value estimate from the simulated scan statistic \((P_s)\) for a specific chromosome with a specified window size \((w)\).
3.5.2 Theoretical Scan Statistics

Theoretically, we will refer to (Glaz, 1989) to obtain the distribution of the scan statistic by the approximation of the \( m^{th} \) order Markov sequence. These approximations with \( m = 2 \), have been applied successfully by Leung et al. (1994, 1996, 2005) for analyzing genomic DNA sequences in herpesviruses.

For \( X_1, \ldots, X_n \sim \text{i.i.d. uniform} \ (0,1) \), the approximated value of \( P\{N_w > r\} \) is given as;

\[
P\{N_w > r\} = 1 - Q_m \left( \frac{q_{m+1}}{q_m} \right)^{n-r-1}
\]  

For the second order approximation \( (m = 2) \), the formulas below will suffice;

\[
Q_1 = \sum_{\frac{n-r}{r-1}} b(k; n, w)
\]
\[
Q_2 = Q_1 - \sum_{k=0}^{n-r} (-1)^k b(r+k; n, w)
\]
\[
Q_3 = Q_2 - \sum_{k=0}^{n-r} (-1)^k (2 + k - k^2) b(r+k; n, w) / 2
\]

for \( r > 2 \) and \( 0 < w < \frac{1}{2} \); where \( b(k; n, w) = \left( \binom{n}{k} w^k (1-w)^{n-k} \right) \).

The approximated theoretical P-value \( P\{N_w > r\} \), follows directly by equation (3.1) when we obtain estimates for \( Q_1, Q_2 \) and \( Q_3 \). This is also done for all variants on the 15 chromosomes with a specified window size of 0.01. At this time, we can observe significant and non-random cluster when compared to an \( \alpha \) level.

Since we now have 15 simulated p-values to compare with the 15 theoretical approximated p-values, we will check how both values closely approximate each other. Another relevant issue posed here is the multiple tests of significance obtained. Statistically, we could make a false-positive declaration (Type 1 error) if we use an error rate of 0.05 (\( \alpha \)) on all 15 tests of significance of the scan statistic, this is because assigning an \( \alpha = 0.05 \) to the first of fifteen tests might impact the results of the remaining fourteen tests, to correct this error, we applied the Bonferroni adjustment which takes the alpha value (\( \alpha = 0.05 \)) and divide it by the number of tests, the quotient obtained is the p-value via the Bonferroni adjustment factor. (Bonferroni, 1936; Haynes, 2013)
Chapter 4: Results and Discussion

In this chapter, we will present an in-depth result-analyses on our dataset, beginning with preliminary exploratory analysis which seeks to provide a general overview of the descriptive measures and trends in our data. A further analysis with scan statistics will be done to conclude findings on PrCa data analysis.

4.1 EXPLORATORY RESULTS

4.1.1 Statistical Summary of Tumor Variants.

The diagrams below display statistical summaries of variants in the tumor samples of our subjects by exploring the distribution of variants on each chromosome and on each of the 12 types of SNPs as shown in Figures 4.1 and 4.3 respectively.

Figure 4.1: Distribution of GSVs on chromosomes including chromosomes X and Y
Figure 4.1 reveals chromosomes 1, 2, 19, 17, 3, 4 and 11 to contain over 5000 counts of gene mutations, giving a clear picture of how rich these chromosomes are in terms of variant count. Out of these, chromosome 1 displays the highest variant count.

![Graph showing variant density on chromosomes including X and Y](source: Tumor Variants by Chromosomes)

While most chromosomes exhibit low density of tumor mutations, chromosomes 19 and 17 evidently displays much higher density of tumor mutations among our tumor samples.

Table 4.1: Distribution of GSV counts on SNPs

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1533</td>
<td>15468</td>
<td>6235</td>
<td>21847</td>
<td>2961</td>
<td>2897</td>
<td>2353</td>
<td>2275</td>
<td>11362</td>
<td>1651</td>
<td>18918</td>
<td>6275</td>
</tr>
</tbody>
</table>
Figure 4.3: Distribution of GSV counts for the 12 different types of SNPs. Each pair of bases on the x-axis stands for the mutation from the first base to the second.

We can infer from the above chart and table 4.1 that the top four most dominant nucleotide changes within the tumor patients are G-A, C-T, C-A and G-T respectively, this is an indication that tumor SNPs are likely to change from G or C towards A or T. Therefore, GSV mutations in prostate cancer patients are believed to change from G/C towards A/T.

Figure 4.4 displays a pictorial proportion of the GSVs in our dataset that have been published in a database of single nucleotide variants called dbSNP. About 35% of the GSVs in our dataset have been recorded in dbSNP and they each has a known “RS” identification number, while the remaining majority of the variants (labeled “None” in the figure) have not.
4.2 Correlations

Prior to obtaining our correlogram, we inspected the statistical summaries of 91FV in the tables 4.2 and 4.3 below. The first five columns in both table shows Tukey’s Five-Number summary while the last two columns display the mean and standard deviation to explain the general characteristics of the 91FV dataset.

Table 4.2: Five-Point Summary with Mean and Standard Deviation of GSV counts per patient

<table>
<thead>
<tr>
<th>Minimum</th>
<th>Lower Quartile</th>
<th>Median</th>
<th>Upper Quartile</th>
<th>Maximum</th>
<th>Mean</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>2.0</td>
<td>9.0</td>
<td>1.42</td>
<td>1.37</td>
</tr>
</tbody>
</table>

The number of GSV counts (mutations in each 91 genes) found in all 503 patients were found to be between 0 and 9. Thus each patient recorded at most 9 gene mutations. Nevertheless, our dimension reduction analysis revealed 148 (29.42%) patients who had no gene mutations but were found to be diagnosed with the cancer, this could possibly be an indirect indication of other possible causes of the tumor instead of strict gene mutations.
Table 4.3: Summary Statistics and Five-Point Summary of Number of Patients per GSV.

<table>
<thead>
<tr>
<th>Minimum</th>
<th>Lower Quartile (Q1)</th>
<th>Median</th>
<th>Upper Quartile (Q3)</th>
<th>Maximum</th>
<th>Mean</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>20.0</td>
<td>7.84</td>
<td>2.57</td>
</tr>
</tbody>
</table>

Table 4.3 above explains the fact that, each of 91 gene mutations recorded at least 6 patients but does not exceed 20 patients. Thus, there was a particular mutation(s) found in as many as 20 patients.

We can infer from both tables that some patients were diagnosed with prostate tumor even though they had no record of GSV count or mutation. However, all gene mutations recorded at least one patient across all 503 patients.

![Box plot showing Five-Number summaries for the GSV and patient distributions in the 91FV dataset.](image)

Figure 4.5: Five-number summaries for the GSV and patient distributions in the 91FV dataset.

The box plots above show some outliers in the distribution of GSV counts on the Five-Number summary statistics. This is because, the maximum counts in both scenarios (20 and 9), are greater than $1.5(IQR) + Q3$, which are 11 and 5 respectively. Therefore, the maximum statistics in both cases are seen as outliers by the boxplot. Thus, the upper whiskers are indicated at the greatest value smaller than $1.5IQR$ above the third quartile as seen above.
The diagram below displays the correlations between the 91 variants that occurred in more than 1% of patients. These variants, with detailed information listed in supplemental file are designated fv01 through fv91 and we will refer to them as frequent variants. The brightly colored spots of the correlogram below is indicative of highly positive correlations between the occurrences of the pair of variants in the 503 PrCa patients (i.e., the pair tend to occur together in the same patients), while dark colored spots are representative of low or negative correlations.

Figure 4.6 Correlogram of variants with more than 1% relative frequency of occurrence (91 FV)
With a correlation coefficient threshold of 0.3, Table 4.2.1 below displays highly and significantly correlated genes with a coefficient above the threshold.

Table 4.4: Highly Correlated Frequent Variant Pairs

<table>
<thead>
<tr>
<th>Variant 1</th>
<th>Variant 2</th>
<th>Correlation</th>
<th>P - value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fv03</td>
<td>Fv24</td>
<td>0.3253</td>
<td>7.339e-14</td>
</tr>
<tr>
<td>Fv11</td>
<td>Fv76</td>
<td>0.3253</td>
<td>7.339e-14</td>
</tr>
<tr>
<td>Fv18</td>
<td>Fv19</td>
<td>0.7577</td>
<td>2.2e-16</td>
</tr>
<tr>
<td>Fv27</td>
<td>Fv28</td>
<td>0.8730</td>
<td>2.2e-16</td>
</tr>
<tr>
<td>Fv32</td>
<td>Fv33</td>
<td>0.4205</td>
<td>2.2e-16</td>
</tr>
<tr>
<td>Fv43</td>
<td>Fv44</td>
<td>0.6403</td>
<td>2.2e-16</td>
</tr>
<tr>
<td>Fv45</td>
<td>Fv46</td>
<td>0.3799</td>
<td>2.2e-16</td>
</tr>
<tr>
<td>Fv59</td>
<td>Fv82</td>
<td>0.3253</td>
<td>7.339e-14</td>
</tr>
</tbody>
</table>

We can observe very high correlations (above 0.5) among gene pairs found in close proximity to each other and located on the same chromosome. This is not surprising because since they are found to be nearest to each other on the same chromosome, they tend to exhibit a direct positive correlation among themselves.

Rather, we will be more interested in gene pairs, either located on the same chromosome but quite far from each other or gene pairs entirely located on different chromosomes. So, the pairs (fv03, fv24), (fv11, fv76) and (fv59, fv82) will be of more importance in the quest to attain the first specific aim of our research objective.
4.3 **Dissimilarity Measure and Cluster Tendency**

In this section, we will assess and discuss the cluster tendency of our dissimilarity measures using the Hopkins statistic and the VAT diagram. A measure of dissimilarity serves as a necessary tool, fed into the clustering algorithm for better partitions and segmentations. The graph below shows the pictorial view of Gower and Correlation dissimilarity measure assessment respectively.

![VAT diagram with Gower dissimilarity measure](image)

**Figure 4.7: VAT diagram with Gower dissimilarity measure**

The number of square shaped blocks along the diagonal in the Gower VAT image is the main clue to detect cluster tendency. As the colors explain, major values of dissimilarity seem to be low with few higher dissimilarity gradients and objects belonging to same cluster are placed in a consecutive manner as read from the legend. This is further confirmed by the additional Hopkins Statistic of 0.7889 obtained, which is close enough to 1 and indicates a good cluster tendency.
From the correlation VAT below, we can infer that similar groups of observations with low dissimilarities are placed in a consecutive manner along the diagonal block. The overall color gradient depicts instances of few observations with low dissimilarities and a good number of observations with high dissimilarities above 1.0.

The Hopkins statistic of 0.8155 is very well close to the rule of thumb, giving an indication of a good cluster tendency.

![Figure 4.8: VAT diagram with correlation dissimilarity measure.](image)

In summary, even though both dissimilarity measures produced promising results of cluster tendency given the VAT and Hopkins Statistic measures, the correlation dissimilarity measure performed better. Nevertheless, both dissimilarity measures will be applied for further clustering analysis.

From the nature of blocks along diagonals on both VAT graphs, the Gower distance measure gives early indication of huge blocks (clusters) while the correlation distance measure shows smaller cluster sizes.
4.4 CLUSTERING RESULTS

All individual subsections within this section specifically present the results of all clustering methods which have been deployed to cluster frequent variants by their occurrences.

4.4.1 Partition Around Medoids – PAM

Visual representation of Partition Clustering of variants into 20 distinct clusters using the both Gower and Correlation distance measures with the PAM method, are displayed in figures 4.8 and 4.9 below, 20 clusters are plotted a graph, where x and y axes are the first two principal component co-ordinates (PCC).

Figure 4.9: Cluster plot with Gower dissimilarity shown on first two PCC
Figure 4.10: Cluster plot with correlation dissimilarity shown on first two PCC

Figures 4.9 and 4.10 shown above, is a visualized two-dimensional plot of clusters with the Gower and correlation distances, plot in two-dimension with the first two principal component co-ordinates.

In each table of cluster memberships shown below, FVs of interest are highlighted in bold italics in their respective clusters ranging from table 4.5 to table 4.10.
Table 4.5: Gower cluster membership with PAM

<table>
<thead>
<tr>
<th>Clusters</th>
<th>Frequent Variants (FV)</th>
<th>S. W</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>01, 37, 56, 64, 71, 73, 78, 89</td>
<td>0.03946</td>
</tr>
<tr>
<td>2</td>
<td>02, 07, 17, 35, 58, 75, 80, 85, 91</td>
<td>0.00799</td>
</tr>
<tr>
<td>3</td>
<td>03, 04, 06, 09, 10, 13, 15, 21, 26, 03, 04, 06, 09, 10, 13, 15, 21, 26, 31, 34, 35, 38, 39, 40, 44, 50, 66, 69, 72, 74, 77, 78.</td>
<td>-0.03058</td>
</tr>
<tr>
<td>4</td>
<td>05, 32, 33, 54</td>
<td>0.1025</td>
</tr>
<tr>
<td>5</td>
<td>08, 25, 52, 59, 60, 65, 67, 70, 82</td>
<td>0.08766</td>
</tr>
<tr>
<td>6</td>
<td>11, 16, 20, 24, 55, 76, 83, 90</td>
<td>0.0474</td>
</tr>
<tr>
<td>7</td>
<td>12</td>
<td>0.0000</td>
</tr>
<tr>
<td>8</td>
<td>14</td>
<td>0.0000</td>
</tr>
<tr>
<td>9</td>
<td>18, 19</td>
<td>0.6500</td>
</tr>
<tr>
<td>10</td>
<td>22</td>
<td>0.0000</td>
</tr>
<tr>
<td>11</td>
<td>23, 30, 42, 46, 48, 49, 51, 57, 61, 68, 87</td>
<td>-0.00218</td>
</tr>
<tr>
<td>12</td>
<td>27, 28</td>
<td>0.8169</td>
</tr>
<tr>
<td>13</td>
<td>29</td>
<td>0.0000</td>
</tr>
<tr>
<td>14</td>
<td>36</td>
<td>0.0000</td>
</tr>
<tr>
<td>15</td>
<td>41, 43, 63, 79, 86</td>
<td>-0.0171</td>
</tr>
<tr>
<td>16</td>
<td>45</td>
<td>0.0000</td>
</tr>
<tr>
<td>17</td>
<td>47</td>
<td>0.0000</td>
</tr>
<tr>
<td>18</td>
<td>62</td>
<td>0.0000</td>
</tr>
<tr>
<td>19</td>
<td>84</td>
<td>0.0000</td>
</tr>
<tr>
<td>20</td>
<td>88</td>
<td>0.0000</td>
</tr>
<tr>
<td>Avg. S. W</td>
<td></td>
<td>0.045</td>
</tr>
</tbody>
</table>
Table 4.6: Correlation cluster membership with PAM

<table>
<thead>
<tr>
<th>Clusters</th>
<th>Frequent Variants (FV)</th>
<th>S.W (correlation)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>01, 07, 35, 42</td>
<td>0.05598</td>
</tr>
<tr>
<td>2</td>
<td>02, 12, 34, 68</td>
<td>0.04397</td>
</tr>
<tr>
<td>3</td>
<td>03, 24, 45, 46</td>
<td>0.1388</td>
</tr>
<tr>
<td>4</td>
<td>04, 05, 25, 56, 67, 84, 86</td>
<td>0.03557</td>
</tr>
<tr>
<td>5</td>
<td>06, 08, 74, 75</td>
<td>0.1215</td>
</tr>
<tr>
<td>6</td>
<td>09, 10, 21, 38, 79, 88</td>
<td>0.03334</td>
</tr>
<tr>
<td>7</td>
<td>11, 16, 20, 55, 76, 90</td>
<td>0.07882</td>
</tr>
<tr>
<td>8</td>
<td>13, 47, 52, 59, 60, 65, 82</td>
<td>0.05747</td>
</tr>
<tr>
<td>9</td>
<td>14, 22, 61, 81</td>
<td>0.04471</td>
</tr>
<tr>
<td>10</td>
<td>15, 36, 53, 63</td>
<td>0.02796</td>
</tr>
<tr>
<td>11</td>
<td>17, 29, 80, 85, 91</td>
<td>0.01655</td>
</tr>
<tr>
<td>12</td>
<td>18, 19</td>
<td>0.7434</td>
</tr>
<tr>
<td>13</td>
<td>23, 40, 50, 51</td>
<td>0.04351</td>
</tr>
<tr>
<td>14</td>
<td>26, 41, 58, 72, 77</td>
<td>0.009413</td>
</tr>
<tr>
<td>15</td>
<td>27, 28</td>
<td>0.8061</td>
</tr>
<tr>
<td>16</td>
<td>30, 31, 66, 70, 87</td>
<td>0.03989</td>
</tr>
<tr>
<td>17</td>
<td>32, 33, 48</td>
<td>0.1669</td>
</tr>
<tr>
<td>18</td>
<td>37, 57, 62, 64, 71, 78, 89</td>
<td>0.02773</td>
</tr>
<tr>
<td>19</td>
<td>39, 49, 54, 69, 73, 83</td>
<td>0.05916</td>
</tr>
<tr>
<td>20</td>
<td>43, 44</td>
<td>0.5728</td>
</tr>
<tr>
<td>Avg. S. W</td>
<td></td>
<td>0.100</td>
</tr>
</tbody>
</table>
Positive values of silhouette indices are preferred indicators of how well observations are clustered in respect to their dissimilarity measures. With quite lower values for both clusters, the correlation measure used for clustering these 91 genes seem to perform better than the Gower measure. The correlation cluster membership shows cluster sizes between 2 and 7. In contrast, the Gower clusters shows numerous singleton members and huge chunk of cluster with over 30 members. PAM clustering with Gower distance clustered only (FV11, FV76) and (FV 59, FV82) in clusters 5 and 6 respectively, while PAM clustering with correlation distance grouped (FV03, FV24), (FV11, FV76) and (FV 59, FV82) in clusters 3, 7 and 8.

The overall nature of cluster membership obtained by PAM Correlation Cluster is preferred since we sought to obtain small groups of correlated variants which exhibits or possesses similar characteristics to be reported to biological laboratories.

4.4.2 Hierarchical Agglomerative Clustering

Beneath are visual representations of 20 distinct clusters reported by the hierarchical agglomerative clusters via dendrograms with their respective distance measures.

![Dendrogram](image.png)

Figure 4.11: Agglomerative hierarchical cluster dendrogram with Correlation dissimilarity
The height of the Agglomerative Correlation Cluster Dendrogram is cut at 0.98, to obtain 20 clusters of even or almost - equal membership size, while the Agglomerative Gower Cluster Dendrogram is anticipated at a height of 0.04 to obtain 20 distinct cluster with a huge imbalance of cluster memberships. A closer look at the entities in each cluster for both dendrograms are spelled out in tables 4.7 and 4.8 below.
Table 4.7: Agglomerative hierarchical cluster membership with Gower dissimilarity

<table>
<thead>
<tr>
<th>Clusters</th>
<th>Frequent Variants (FV)</th>
<th>CPCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11, 13, 15, 16, 17, 20, 21, 24, 25, 26, 27, 28, 30, 31, 32, 33, 34, 35, 37, 38, 39, 40, 42, 43, 44, 46, 50, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 80, 81, 82, 83, 85, 89, 90, 91</td>
<td>0.9333</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>18, 19</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>41, 86</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>47</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>48</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>49</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>62</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>63</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>79</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>84</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>87</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>88</td>
<td></td>
</tr>
<tr>
<td>CPCC</td>
<td>0.9333</td>
<td></td>
</tr>
</tbody>
</table>
Table 4.8: Agglomerative hierarchical cluster membership with correlation dissimilarity

<table>
<thead>
<tr>
<th>Clusters</th>
<th>Frequent Variants (FV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>01, 35, 42</td>
</tr>
<tr>
<td>2</td>
<td>02, 07, 12, 17, 18, 19, 27, 28, 29, 79, 85, 91</td>
</tr>
<tr>
<td>3</td>
<td>03, 24, 45, 46</td>
</tr>
<tr>
<td>4</td>
<td>04, 15, 37</td>
</tr>
<tr>
<td>5</td>
<td>05, 41, 56, 63, 67, 84, 86</td>
</tr>
<tr>
<td>6</td>
<td>06, 25, 64, 74, 75</td>
</tr>
<tr>
<td>7</td>
<td>08, 52, 59, 60, 65, 70, 82</td>
</tr>
<tr>
<td>8</td>
<td>09, 68</td>
</tr>
<tr>
<td>9</td>
<td>10, 14, 22, 36, 43, 44, 53, 80, 81</td>
</tr>
<tr>
<td>10</td>
<td>11, 20, 55, 76, 90</td>
</tr>
<tr>
<td>11</td>
<td>13, 61</td>
</tr>
<tr>
<td>12</td>
<td>16, 40, 48, 66, 78, 87</td>
</tr>
<tr>
<td>13</td>
<td>21, 38, 88</td>
</tr>
<tr>
<td>14</td>
<td>23, 50, 51</td>
</tr>
<tr>
<td>15</td>
<td>26, 47, 58, 72</td>
</tr>
<tr>
<td>16</td>
<td>30, 31, 57</td>
</tr>
<tr>
<td>17</td>
<td>32, 33, 54</td>
</tr>
<tr>
<td>18</td>
<td>34, 77</td>
</tr>
<tr>
<td>19</td>
<td>39, 49, 73, 83</td>
</tr>
<tr>
<td>20</td>
<td>62, 69, 71, 89</td>
</tr>
<tr>
<td>CPCC</td>
<td>0.6614</td>
</tr>
</tbody>
</table>
Quite interesting, agglomerative hierarchical clustering with Gower distance grouped (FV03, FV24), (FV11, FV76) and (FV 59, FV82) together in cluster 1, while agglomerative hierarchical clustering with correlation distance grouped (FV03, FV24) into cluster 3, (FV11, FV76) into cluster 10 and (FV 59, FV82) into cluster 2. The Cophenetic Correlation Coefficient (CPCC) gives an indication of how pairwise effects in each distance measure is religiously preserved by the average linkage dendrogram (tree cluster). The average linkage provided the highest CPCC among all eight (8) indices tested. It is worth mentioning once again, that this same linkage was judged in our methodology section, as the best HCA method since it provides the balance between the basic linkage methods i.e., “single” – nearest neighbor, “complete” – nearest neighbor and “ward” – minimum variation with compact clusters. The Gower membership shows a massive chunk of variants(genes) in cluster 1, with singleton genes in 19 remaining clusters, while the Correlation membership once again shows desirable gene cliques (small groups) in all clusters.

4.4.3 Hierarchical Divisive Clustering

![Figure 4.13: Divisive hierarchical cluster dendrogram with correlation dissimilarity](image-url)
Figure 4.14: Divisive hierarchical cluster dendrogram with Gower dissimilarity

At a height of 1.00, figure 4.13 displays twenty (20) distinct clusters of evenly sized members while the figure 4.14 displays 20 distinct clusters of huge imbalanced memberships at a height of 0.050. A closer look at the entities in each cluster for both dendrograms are spelled out in tables 4.9 and 4.10 below.
Table 4.9: Divisive hierarchical cluster membership with Gower dissimilarity

<table>
<thead>
<tr>
<th>Clusters</th>
<th>Frequent Variants (FV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>01, 02, 03, 04, 06, 08, 09, 10, 13, 15, 17, 21, 25, 26, 31, 34, 35, 37, 38, 39, 40, 42, 49, 50, 52, 54, 57, 58, 59, 60, 61, 64, 65, 66, 68, 69, 70, 71, 72, 73, 74, 75, 77, 78, 81, 82, 83, 89</td>
</tr>
<tr>
<td>2</td>
<td>05, 32, 33, 56, 67, 86</td>
</tr>
<tr>
<td>3</td>
<td>07, 41, 43, 44, 53, 80</td>
</tr>
<tr>
<td>4</td>
<td>11, 16, 20, 24, 27, 28, 48, 55, 76, 85, 87, 90, 91</td>
</tr>
<tr>
<td>5</td>
<td>12</td>
</tr>
<tr>
<td>6</td>
<td>14</td>
</tr>
<tr>
<td>7</td>
<td>18, 19</td>
</tr>
<tr>
<td>8</td>
<td>22</td>
</tr>
<tr>
<td>9</td>
<td>23</td>
</tr>
<tr>
<td>10</td>
<td>29</td>
</tr>
<tr>
<td>11</td>
<td>30</td>
</tr>
<tr>
<td>12</td>
<td>36</td>
</tr>
<tr>
<td>13</td>
<td>45, 46</td>
</tr>
<tr>
<td>14</td>
<td>47</td>
</tr>
<tr>
<td>15</td>
<td>51</td>
</tr>
<tr>
<td>16</td>
<td>62</td>
</tr>
<tr>
<td>17</td>
<td>63</td>
</tr>
<tr>
<td>18</td>
<td>79</td>
</tr>
<tr>
<td>19</td>
<td>84</td>
</tr>
<tr>
<td>20</td>
<td>88</td>
</tr>
<tr>
<td><strong>D.C</strong></td>
<td><strong>0.61</strong></td>
</tr>
</tbody>
</table>
Table 4.10: Divisive hierarchical cluster membership with correlation dissimilarity

<table>
<thead>
<tr>
<th>Clusters</th>
<th>Frequent Variants (FV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>01, <strong>11</strong>, 20, 35, 42, <strong>76</strong>, 83, 90</td>
</tr>
<tr>
<td>2</td>
<td>02, 12, 17</td>
</tr>
<tr>
<td>3</td>
<td><strong>03</strong>, <strong>24</strong>, 26, 45, 46, 88</td>
</tr>
<tr>
<td>4</td>
<td>04, 15, 30, 31, 36, 41, 53, 68</td>
</tr>
<tr>
<td>5</td>
<td>05, 25, 32, 33, 56, 67, 78, 86</td>
</tr>
<tr>
<td>6</td>
<td>06, 47, 80</td>
</tr>
<tr>
<td>7</td>
<td>07, 16, 40, 48, 55, 66, 87</td>
</tr>
<tr>
<td>8</td>
<td>08, 79</td>
</tr>
<tr>
<td>9</td>
<td>09, 21</td>
</tr>
<tr>
<td>10</td>
<td>10, 14</td>
</tr>
<tr>
<td>11</td>
<td>13, 61</td>
</tr>
<tr>
<td>12</td>
<td>18, 19, 27, 28, 29, 85</td>
</tr>
<tr>
<td>13</td>
<td>22, 43, 44, 63, 81</td>
</tr>
<tr>
<td>14</td>
<td>23, 50, 51, 70</td>
</tr>
<tr>
<td>15</td>
<td>34, 77</td>
</tr>
<tr>
<td>16</td>
<td>37, 38, 57, 64, 71, 89</td>
</tr>
<tr>
<td>17</td>
<td>39, 49, 52, 54, <strong>59</strong>, 60, 65, 69, 72, 73, <strong>82</strong></td>
</tr>
<tr>
<td>18</td>
<td>58, 84</td>
</tr>
<tr>
<td>19</td>
<td>fv62</td>
</tr>
<tr>
<td>20</td>
<td>74, 75, 91</td>
</tr>
<tr>
<td>D.C</td>
<td>0.23</td>
</tr>
</tbody>
</table>

Irrespective of the high strength of the cluster distinction recorded with Gower measure, the gene membership in its clusters were not balanced as there were two clusters of chunked membership, and majority being singleton clusters. Conversely, the divisive clusters with the
correlation measure produced a relatively lower divisive cluster distinction but, majority of its clusters contained nearly balanced variant-membership. Divisive hierarchical Gower cluster grouped (FV59, FV82) and (FV11, FV76) in clusters 1 and 4, while divisive hierarchical correlation cluster grouped and (FV11, FV76), (FV03, FV24) and (FV59, FV82) into clusters 1, 3 and 17 respectively.

Table 4.11: Summary of clustered correlated variant pairs by clustering methods

<table>
<thead>
<tr>
<th>Correlated Gene Pairs</th>
<th>PAM</th>
<th>Agglomerative H.C</th>
<th>Divisive H.C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Correlation</td>
<td>Gower</td>
<td>Correlation</td>
</tr>
<tr>
<td>(FV03, FV24)</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>(FV11, FV76)</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>(FV59, FV82)</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

The table above shows the clustering of highly correlated gene pairs in each clustering method. An asterisk in each cell is an indication of the presence of a cluster containing both pairs of variants, given their respective cluster methods. It is clearly evident that only (FV11, FV76) and (FV59 and FV82) were consistently grouped in the same by the all six (6) clustering methods.
4.5 Gene Regions with High Variant Concentration

The graph of scan statistics along the unit-gene length of a chromosome identified certain regions believed to amass prostate cancer GSVs. An exemplary look at figure 4.15 below shows a sliding window plot displaying how variants are clustered on the KIR2DS3 gene on chromosome 19 and the significance of that highest peak explained.

![Figure 4.15: Sliding window plot for gene KIR2DS3 (length = 123645 bases) with a window size of 0.01 on the gene normalized to unit length.](image)

A closer look at the sliding window plot above shows a peak value of 32 variants at the point 0.9905536 on the unit gene-length which translates to position 122477 on the actual gene length. This high variant of 32 is the scan statistic which illustrates a cluster of variants at that specified point. The second highest peaked value of 30 found at point 0.3656193 translating to position 45207 on the gene-length also shows another observation of variant cluster.
To determine whether this cluster happened at random or otherwise, the approximated theoretical p-value (Glaz, 1989) and simulated p-value obtained after 1000 repeated trials with randomly generated uniform data, when adjusted with the Bonferroni correction factor for multiple tests produced significant p-value, thus the observed cluster is significant and did not occur by chance. This treatment was applied on all fifteen (15) variants with more than 60 distinct GSVs.

Table 4.12: Gene regions with high concentrations of GSVs identified by scan statistics using window length of 0.01.

<table>
<thead>
<tr>
<th>Chromosome</th>
<th>Gene</th>
<th>Location</th>
<th>Scan Statistic</th>
<th>Simulated P-Value</th>
<th>Theoretical P-Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>KIR2DS3</td>
<td>122477</td>
<td>32</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>7</td>
<td>TCAF2</td>
<td>338169</td>
<td>69</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>6</td>
<td>SYNE1</td>
<td>243038</td>
<td>9</td>
<td>0.0010</td>
<td>0.0020</td>
</tr>
<tr>
<td>5</td>
<td>PCDAH1</td>
<td>21348</td>
<td>12</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>2</td>
<td>TTN</td>
<td>201123</td>
<td>9</td>
<td>0.0001</td>
<td>0.0010</td>
</tr>
<tr>
<td>2</td>
<td>LRP1B</td>
<td>469094</td>
<td>9</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>1</td>
<td>HMCN1</td>
<td>351940</td>
<td>8</td>
<td>0.0000</td>
<td>0.0010</td>
</tr>
<tr>
<td>3</td>
<td>MUC4</td>
<td>41717</td>
<td>17</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>6</td>
<td>TNXB</td>
<td>53114</td>
<td>8</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

When the Bonferroni adjusted p-value of 0.0033 was used as a filtering threshold, for all fifteen (15) variants, we obtained a shortlist of nine (9) remaining variants as shown in table 4.12 above indicating regions on each chromosome with an observed non-random cluster.

Figure 4.17 displays the remaining sliding window graphs which shows the scan statistic of each gene on its respective chromosome, where we can refer to table 4.12 above to fetch the exact position on a gene at which significant variant concentration or clusters are located.
Figure 4.16: Grided sliding window plots of genes with significant clusters using a window size of 0.01. The highest peak in each plot gives the scan statistic value.
Chapter 5: Conclusion and Future Work

5.1 Conclusion

The statistical analyses of this WES dataset have revealed several interesting properties of GSVs in PrCa tumor tissues. For example, the number and densities of GSVs are quite heterogeneous among different chromosomes, with the highest count in chromosome 1, and highest density in chromosome 19. Among the 12 (twelve) types of single nucleotide substitution variants in the cancer samples, the mutations G-A, C-T, C-A, and G-T that change the strong bases to weak bases occur more frequently than the remaining eight (8) mutations. Also, there are pairs of frequent variants on different chromosomes with significant statistical correlations. Cluster analysis further suggests that these correlated GSV pairs could be extended to a few small groups of frequent variants with 4-7 members that are close to one another in terms of their occurrence patterns among the patients. On examining the GSV locations on the top genes containing highest numbers of GSVs, the scan statistic has helped us identify certain gene regions harboring high concentration of GSVs.

These findings will be reported to biomedical scientists for further bioinformatics analysis and wet-lab experimental studies. With the rapidly increasing amounts of genomic sequencing data, statistical tools have proven to be indispensable for extracting useful information that can serve as a guide for more finely-tuned research leading to better methods of diagnosis and treatments of cancer.
5.2 Future Research

As stated in the first specific aim on section 1.3, we have been able to identify individual GSVs or groups of correlated GSVs from the sequenced variants obtained from our subjects diagnosed with prostate cancer by deploying basic partitional and hierarchical clustering methods which were able to finding spherical and compact clusters. However, future research can focus on improving the quality of variant clusters with the usage of other forms of clustering algorithms robust to noise and can handle sparse data matrices comfortably.

With respect to the second specific aim, our scan statistic technique was able to give an accurate count of variants to determine the locations of the regions with high GSV concentration on the gene as long as the code is rightly fed with the correct parameters. However, due to the time-frame and setting of this thesis, the code was only able to fish out the starting positions at which the variants are clustered on the gene, Future research can be directed at finding the exact boundary points of gene regions in which GSVs are significantly concentrated.
References


Appendices

Appendix A: Statistical Summary of Tumor Variants

# Lollipop Chart of variant count on each SNP

```r
require(ggplot2)

alltvd <- read.csv("all_tvd.csv")

snp <- table(alltvd$SNP)

snps <- data.frame(snp)

x = snps$Var1

y = snps$Freq

snps <- data.frame(x, y)

ggplot(snps, aes(x=x, y=y), stat="identity") +
  geom_segment(aes(x=x, xend=x, y=0, yend=y)) +
  geom_point(size=5, color="red", fill=alpha("orange", 0.3),
             alpha=0.7, shape=21, stroke=2) +
  labs(title="Distributions of Variant on SNP",
       caption="Source: Tumor Sample - Transcript Variant Distribution",
       x="Single Nucleotide Polymorphism (SNP)",
       y="Count") +
  theme(axis.text.x = element_text(angle=0, vjust=0.5)) +
  theme(plot.title = element_text(hjust = 0.5, face="bold"))
```

# Lollipop Chart of Variants on Each Chromosome

```r
var$Chromosome <- c(seq(1,22),"x","y")

var$Chromosome <- factor(var$Chromosome, levels=c(seq(1,22),"x","y"))

y <- var$var.by.chromosome

x <- var$Chromosome

dat <- data.frame(x, y)

ggplot(dat, aes(x=x, y=y), stat="identity") +
  geom_segment(aes(x=x, xend=x, y=0, yend=y)) +
  geom_point(size=5, color="red", fill=alpha("orange", 0.3),
             alpha=0.7, shape=21, stroke=2) +
  labs(title="Distribution of Variants on Each Chromosome",
       x="Chromosome",
       y="Count") +
  theme(axis.text.x = element_text(angle=0, vjust=0.5)) +
  theme(plot.title = element_text(hjust = 0.5, face="bold"))
```
library(ggplot2)

data <- data.frame(
  category=c("Known(RS)", "None"),
  count=c(30492, 63283))
data$fraction <- (data$count / sum(data$count)) * 100
data$ymax <- cumsum(data$fraction)
data$ymin <- c(0, head(data$ymax, n=-1))
data$labelPosition <- (data$ymax + data$ymin) / 2
data$label <- paste0(data$category)
ggplot(data, aes(ymax=ymax, ymin=ymin, xmax=4, xmin=3, fill=category)) +
  geom_rect() + geom_label( x=3.5, aes(y=labelPosition, label=label), size=6) +
  scale_fill_brewer(palette=11) + coord_polar(theta="y") +
  xlim(c(2, 4)) + theme_void() +
  theme(legend.position = "none") + labs(title="Distribution of DbSNP", 
    caption="Source: Tumor Sample - All Regions on Transcript Variant") +
  theme(plot.title = element_text(hjust = 0.5, face="bold")) +
  theme(plot.subtitle = element_text(hjust = 0.5, face="bold"))
Appendix B: Statistical Summaries of 91 FV.

# Five – Number Summary GSV Count Per Patient

```r
setwd("C:\Users\Kelvin\Documents\FINAL.THEESIS\data\EDA")
fv<-read.csv("matrix.csv")
gsv.pp<-rowSums(fv)
summary(gsv.pp)
sd(gsv.pp)
fivenum(gsv.pp)
```

# Number of patients per GSV in the tumor samples

```r
npp.gsv<-colSums(fv)
summary(npp.gsv)
fivenum(npp.gsv)
sd(npp.gsv)
```

# Boxplot of both Summaries

```r
colors=c("red", "blue")
patients_per_gsv <- npp.gsv
gsv_per_patient<-gsv.pp
boxplot(patients_per_gsv, gsv_per_patient, col=c("red", "blue"), xlab="Distribution of Gene Mutations")
legend('topright', fill=colors, legend=c("Patients per GSV", "GSVs per patient"))
```
Appendix C: Correlations among 91 FV

# Correlogram

setwd("C:/Users/Kelvin/Documents/FINAL.THESES/data/fv")
suppressPackageStartupMessages(require(corrplot))
matrixx <- read.csv("matrix.csv", header=T)
corrplot(cor(as.matrix(matrixx), method = "pearson", use = "complete.obs"), is.corr = T,
        type = "lower", tl.col = "darkred", tl.cex = 0.8, tl.srt = 50, main = "Correlogram of 91 FV")

# Write correlation matrix to excel workbook.

mat <- cor(matrixx, method = "pearson")
write.csv(mat, file="C:/Users/Kelvin/Documents/FINAL.THESES/data/fv/correlation_matrix.csv", row.names = T)

higher_correlations <- mat[mat > 0.3 & mat != 1]

# Enumerate Paired Correlations of Variants greater than 0.3 excluding all main diagonals.

for (i in 1:nrow(mat)){
    correlations <- which((mat[i,] > 0.3) & (mat[i,] != 1))
    if(length(correlations) > 0){
        print(colnames(matrixx)[i])
        print(correlations)
    }
}

# Significance of Highly Correlated Variants

s <- as.matrix(matrixx)
cor.test(s, "fv03", s, "fv24", method = "pearson")
cor.test(s, "fv11", s, "fv76", method = "pearson")
cor.test(s, "fv18", s, "fv19", method = "pearson")
cor.test(s, "fv27", s, "fv28", method = "pearson")
cor.test(s, "fv32", s, "fv33", method = "pearson")
cor.test(s, "fv43", s, "fv44", method = "pearson")
cor.test(s, "fv45", s, "fv46", method = "pearson")
cor.test(s, "fv59", s, "fv82", method = "pearson")

# Correlogram from Python
require(reticulate)
import pandas as pd
import numpy as np
import scipy as sp
import scipy.stats
import seaborn as sns
import matplotlib.pyplot as plt
mat = pd.read_csv(r'C:\Users\Kelvin\Documents\FINAL.THERESIS\data\fv\matrix.csv')
dat = pd.DataFrame(data=mat)
correlation = dat.corr(method="pearson")
#Write correlations to excel workbook
correlation.to_excel(r"C:\Users\Kelvin\Documents\FINAL.THERESIS\data\fv\python\supplementary.xlsx", index=True)
plt.figure(figsize=(20,20))
sns.heatmap(correlation)
cor03=correlation[correlation > 0.3]  #Correlations greater than 0.3

Appendix D: Partitional Clustering of 91 FV
#Eliminating subjects with Zero Variance
fv<-read.csv("matrix.csv")
fv1<-as.matrix(fv)
fv1<-t(fv1)
PCs <- prcomp(fv1)
which(apply(fv1, 2, var)==0) #rows/subjects with zero variance
k =as.data.frame(fv1[,which(apply(fv1, 2, var) != 0)]); k
#Correlation Dissimilarity Distance and Cluster Tendency
r<-as.data.frame(cor(t(k)))
corr<-1-r
corre <- get_clust_tendency(corr, n = ncol(t(k))-1, graph = T)
#Alternative option with inbuilt function
suppressPackageStartupMessages(require(DiffCorr))
cd<-cor.dist(k, methods = "pearson", absolute = FALSE)
#Gower Dissimilarity with its Cluster Tendency
suppressWarnings(gower_dist<-daisy(k, metric="gower", stand=FALSE))
gower_mat<-as.matrix(gower_dist)
gowerr <- get_clust_tendency(gower_mat, n = ncol(t(k))-1, graph = T)
round(gowerr$hopkins_stat, 4)
gowerr$plot
#Libraries Required for Clustering
suppressPackageStartupMessages(require(kableExtra))
suppressPackageStartupMessages(library(tidyverse))
suppressPackageStartupMessages(library(cluster))
suppressPackageStartupMessages(library(factoextra))

#Partitional Clustering with Gower Distance.
pamResult10 <-pam(gower_mat, k = 20)
fviz_cluster(pamResult10, main="Cluster with Gower Dissimilarity")
silhouette_width1<-pamResult10$silinfo
Medoid1<-pamResult10$id.med
table(pamResult10$clustering)
rownames(k)[pamResult10$clustering == 1]
rownames(k)[pamResult10$clustering == 2]
rownames(k)[pamResult10$clustering == 3]
rownames(k)[pamResult10$clustering == 4]
rownames(k)[pamResult10$clustering == 5]
rownames(k)[pamResult10$clustering == 6]
rownames(k)[pamResult10$clustering == 7]
# Partitional Clustering with Correlations Distance.

```r
pamResult11 <- pam(corr, k = 20)
fviz_cluster(pamResult11, main="Clustering with Correlation Dissimilarity")
silhouette_width1 <- pamResult11$silinfo
Medoids2 <- pamResult11$id.med

table(pamResult11$clustering)
```

rownames(k)[pamResult10$clustering == 8]
rownames(k)[pamResult10$clustering == 9]
rownames(k)[pamResult10$clustering == 10]
rownames(k)[pamResult10$clustering == 11]
rownames(k)[pamResult10$clustering == 12]
rownames(k)[pamResult10$clustering == 13]
rownames(k)[pamResult10$clustering == 14]
rownames(k)[pamResult10$clustering == 15]
rownames(k)[pamResult10$clustering == 16]
rownames(k)[pamResult10$clustering == 17]
rownames(k)[pamResult10$clustering == 18]
rownames(k)[pamResult10$clustering == 19]
rownames(k)[pamResult10$clustering == 20]
rownames(k)[pamResult11$clustering == 10]
rownames(k)[pamResult11$clustering == 11]
rownames(k)[pamResult11$clustering == 12]
rownames(k)[pamResult11$clustering == 13]
rownames(k)[pamResult11$clustering == 14]
rownames(k)[pamResult11$clustering == 15]
rownames(k)[pamResult11$clustering == 16]
rownames(k)[pamResult11$clustering == 17]
rownames(k)[pamResult11$clustering == 18]
rownames(k)[pamResult11$clustering == 19]
rownames(k)[pamResult11$clustering == 20]

Appendix E: Hierarchical Clustering of 91 FV
#Distance Measure: Agglomerative - Correlation cluster
suppressPackageStartupMessages(require(DiffCorr))
cd<-cor.dist(k, methods = "pearson", absolute = FALSE)
#converted to dist object for H.C
cd2<-as.dist(cd)

# Linkage Functions and Cophenetic distances - Correlations
WARD1<-hclust(d = cd2, method = "ward.D");  WARD2<-hclust(d = cd2, method = "ward.D2");
SINGLE<-hclust(d = cd2, method = "single");  COMPLETE<-hclust(d = cd2, method = "complete");
AVERAGE<-hclust(d=cd2,method="average");  MCQUITTY<-hclust(d = cd2, method="mcquitty");
MEDIAN<-hclust(d = cd2, method = "median");  CENTROID<-hclust(d = cd2, method="centroid");
# Compute cophenetic distance of link functions
coph.ward1 <- cophenetic(WARD1) ;  coph.ward2<-cophenetic(WARD2)
coph.single<-cophenetic(SINGLE);  coph.complete<-cophenetic(COMplete)
coph.average<-cophenetic(AVERAGE);  coph.mcquitty<-cophenetic(MCQUITTY)
coph.median<-cophenetic(MEDIAN);  coph.centroid<-cophenetic(CENTROID)

# Correlation between cophenetic distance and the original distance
# HC Agglomerative Dendrogram and Membership - Correlations

```r
#Correlation between CD2 and other similarity measures
cor(cd2, coph.single); cor(cd2, coph.complete)

# Agglomerative HC Dendrogram and Membership
fviz_dend(AVERAGE, cex = 0.45, main="Agglomerative Cluster Dendrogram with average",
k = 20, palette = "jco", xlab="Genes", ggtheme = theme_classic(), rect = T, rect_fill = T,
rect_border = 2:5, rect_lty = 1, lwd = 0.7, color_labels_by_k = TRUE, lower_rect = 0)

# Cluster Membership
partition1 <- cutree(AVERAGE, k = 20)
table(partition1) #Cluster Membership

# Colnames of CD2 based on partition1
colnames(cd)[partition1 == 1]; colnames(cd)[partition1 == 2]
colnames(cd)[partition1 == 3]; colnames(cd)[partition1 == 4]
colnames(cd)[partition1 == 5]; colnames(cd)[partition1 == 6]
colnames(cd)[partition1 == 7]; colnames(cd)[partition1 == 8]
colnames(cd)[partition1 == 9]; colnames(cd)[partition1 == 10]
colnames(cd)[partition1 == 11]; colnames(cd)[partition1 == 12]
colnames(cd)[partition1 == 13]; colnames(cd)[partition1 == 14]
colnames(cd)[partition1 == 15]; colnames(cd)[partition1 == 16]
colnames(cd)[partition1 == 17]; colnames(cd)[partition1 == 18]
colnames(cd)[partition1 == 19]; colnames(cd)[partition1 == 20]
```

# Linkage Functions - Gower Distance

```r
# Linkage Functions
WARD1.g <- hclust(d = gower.mat, method = "ward.D");
WARD2.g <- hclust(d = gower.mat, method = "ward.D2");
SINGLE.g <- hclust(d = gower.mat, method = "single");
COMPLETE.g <- hclust(d = gower.mat, method = "complete");
AVERAGE.g <- hclust(d = gower.mat, method = "average");
MCQUITTY.g <- hclust(d = gower.mat, method = "mcquitty");
MEDIAN.g <- hclust(d = gower.mat, method = "median");
```
CENTROID.g <- hclust(d = gower.mat, method = "centroid");

# Compute cophenetic distance of link functions
coph.ward1 <- cophenetic(WARD1.g)      coph.ward2 <- cophenetic(WARD2.g)
coph.single <- cophenetic(SINGLE.g)    coph.complete <- cophenetic(COMPLETE.g)
coph.average <- cophenetic(AVERAGE.g)  coph.mcquitty <- cophenetic(MCQUITTY.g)
coph.median <- cophenetic(MEDIAN.g)    coph.centroid <- cophenetic(CENTROID.g)

# Correlation between cophenetic distance and the original distance
cor(gower.mat, coph.single)           cor(gower.mat, coph.complete)
cor(gower.mat, coph.ward1)           cor(gower.mat, coph.ward2)
cor(gower.mat, coph.average)         cor(gower.mat, coph.mcquitty)
cor(gower.mat, coph.median)          cor(gower.mat, coph.centroid)

# HC Agglomerative Dendrogram and Membership - Gower
AVERAGE.g <- hclust(d = gower.mat, method = "average");

# Make presentable dendrograms
fviz_dend(AVERAGE.g, cex = 0.45,
          main = "Agglomerative Cluster Dendrogram with AVERAGE.G", k = 20, palette = "lancet",
          xlab = "genes", ggtheme = theme_classic(), rect = T, rect_fill = T, rect_border = 2:5,
          rect_lty = 1, lwd = 0.7, color_labels_by_k = TRUE, lower_rect = 0)
partition1 <- cutree(AVERAGE.g, k = 20)
table(partition1)

colnames(cd)[partition1 == 1];
colnames(cd)[partition1 == 2];
colnames(cd)[partition1 == 3];
colnames(cd)[partition1 == 4];
colnames(cd)[partition1 == 5];
colnames(cd)[partition1 == 6];
colnames(cd)[partition1 == 7];
colnames(cd)[partition1 == 8];
colnames(cd)[partition1 == 9];
colnames(cd)[partition1 == 10];
colnames(cd)[partition1 == 11];

64
# HC Divisive Dendrogram and Membership - Correlations

dv <- diana(cd2, diss = T, stand=F)
divisive.coefficient <- dv$dc
fviz_dend(dv, cex = 0.5, k = 20, palette = "jco", ggtheme = theme_classic(), rect = T,
rect_fill = T, rect_border = 2:5, rect_lty = 1, lwd = 0.7, color_labels_by_k = TRUE, lower_rect = 0)
dvgroups <- cutree(as.hclust(dv), k = 20)
table(dvgroups)
rownames(cd)[dvgroups == 1]; rownames(cd)[dvgroups == 2]
rownames(cd)[dvgroups == 3]; rownames(cd)[dvgroups == 4]
rownames(cd)[dvgroups == 5]; rownames(cd)[dvgroups == 6]
rownames(cd)[dvgroups == 7]; rownames(cd)[dvgroups == 8]
rownames(cd)[dvgroups == 9]; rownames(cd)[dvgroups == 10]
rownames(cd)[dvgroups == 11]; rownames(cd)[dvgroups == 12]
rownames(cd)[dvgroups == 13]; rownames(cd)[dvgroups == 14]
rownames(cd)[dvgroups == 15]; rownames(cd)[dvgroups == 16]
rownames(cd)[dvgroups == 17]; rownames(cd)[dvgroups == 18]
rownames(cd)[dvgroups == 19]; rownames(cd)[dvgroups == 20]

# HC Divisive Dendrogram and Membership - Gower
dv.g <- diana(gower_mat, diss = T, stand=F)
plot(dv.g)
divisive_coefficient_gower <- dv.g

fviz_dend(dv.g, cex = 0.5, k = 20, palette = "jco", ggtheme = theme_classic(), rect = T, rect_fill = T, rect_border = 2:5, rect_lty = 1, lwd = 0.7, color_labels_by_k = TRUE, lower_rect = 0)
dvg <- cutree(as.hclust(dv.g), k=20)
table(dvg) # 8 and 42 group members
rownames(cd)[dvg == 1]; rownames(cd)[dvg == 2]
rownames(cd)[dvg == 3]; rownames(cd)[dvg == 4]
rownames(cd)[dvg == 5]; rownames(cd)[dvg == 6]
rownames(cd)[dvg == 7]; rownames(cd)[dvg == 8]
rownames(cd)[dvg == 9]; rownames(cd)[dvg == 10]
rownames(cd)[dvg == 11]; rownames(cd)[dvg == 12]
rownames(cd)[dvg == 13]; rownames(cd)[dvg == 14]
rownames(cd)[dvg == 15]; rownames(cd)[dvg == 16]
rownames(cd)[dvg == 17]; rownames(cd)[dvg == 18]
rownames(cd)[dvg == 19]; rownames(cd)[dvg == 20]

Appendix F: Observed and Simulated Scan Statistics with P-values

# Directory and Libraries
setwd("C:/Users/Kelvin/Documents/FINAL.THESIS/scan statistics/SW-FINALS")
suppressPackageStartupMessages(require(kableExtra))
suppressPackageStartupMessages(require(tidyverse))
suppressPackageStartupMessages(require(ggplot2))

#Function to Locate and Compute Observed Scan Statistic
sw2 <- function(ws, data){
  chunks <- 1:nrow(data)
  for (i in 1:nrow(data)) {
    cursor <- location[i]
    selection <- (location >= cursor) & (location <= (cursor + ws))
    chunks[i] <- sum(data$var[selection])}


#Function to compute p-value of Simulation Scan Statistic

```r
def p_value = function(max_real)
{
  trials=1000
  va = as.matrix(v1)
  Sm_scan = length(va[va>=max_real])
  pv = round(Sm_scan/trials, digits = 4)
  return(pv)
}
```

#chrom19 - kir2ds3: Scan Statistic on Real Data

```r
setwd("C:/Users/Kelvin/Documents/FINAL.THESIS/scan statistics/SW-FINALS")
chr19 = read.csv("chr19.kir2ds3.csv")
colnames(chr19) = c("id", "gene-loc", "points", "var")
location = chr19$points
cursor = min(chr19$points)
count1 = sw2(ws=0.01, data=chr19) # specify window and data
window1 = chr19$points
dframe1 = data.frame(id=chr19$id, window1, count1, position=chr19$gene-loc)
max1 = max(count1)
```

#plot scan statistic

```r
theme_set(theme_classic())

n = ggplot(dframe1, aes(window1))
n = n + geom_line(aes(y=count1), colour="red") + labs(title="Sliding Window Plot", 
subtitle="Variant Peakedness on Chromosome 19", caption="Source: Gene GSV location", 
x="Window positions", y="Peaks") + theme(plot.title = element_text(hjust = 0.5, 
face="bold")) + theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n
```

```r

a = dframe1%>%slice_max(count1); a #subsetting the maximum value.
```

#Repeated Trials of Scan Statistic on Simulated Data

```r
start02 = Sys.time()
```

```r
```
ct <- 0
chr19 <- read.csv("chr19.kir2ds3.csv")
v1 <- NULL
repeat{
    points <- sort(runif(196, min = 0, max = 1))
    random <- data.frame(points, var = chr19$SUB.LOC)
    location <- random$points
    cursor <- min(random$points)
    count1 <- sw2(ws = 0.01, data = random)
    e <- max(count1)
    v1 <- append(v1, e, after = length(v1))
    ct <- ct + 1
    if (ct == 1000){
        break}
}
print(v1)
end02 <- Sys.time()
time02 <- end02 - start02; time02

#P-value of Simulated Scan Statistic on Chrom19
p_chr19 <- p_value(max1); p_chr19

#chrom7 - TCAF2: Scan Statistic on Real Data
chr7 <- read.csv("chr7.TCAF2.csv")
colnames(chr7) <- c("id", "gene-loc", "points", "var")
location <- chr7$points
cursor <- min(chr7$points)
count2 <- sw2(ws = 0.01, data = chr7)
window2 <- chr7$points
dframe2 <- data.frame(id = chr7$id, window2, count2, position = chr7$`gene-loc`)
max2 <- max(count2); max2
#plot scan statistic

```r
tHEME_SET(THREE_CLASSIC())

n <- ggplot(dframe2, aes(window2))
n <- n + geom_line(aes(y=count2), colour="red") + labs(title="Sliding Window Plot", subtitle="Variant Peakedness on Chromosome 7", caption="Source: Gene GSV location", x="Window positions", y="Peaks") + theme(plot.title = element_text(hjust = 0.5, face="bold"), plot.subtitle = element_text(hjust = 0.5, face="bold")); n

b <- dframe2%dashslice_max(count2); b

#Repeated Trials of Scan Statistic on Simulated Data

start <- Sys.time()

ct <- 0

v1 <- NULL

repeat{
  points <- sort(runif(145, min = 0, max = 1))
  random2 <- data.frame(points = points, var = chr7$var)
  location <- random2$points
  cursor <- min(random2$points)
  count2 <- sw2(ws = 0.01, data = random2)
  e <- max(count2)
  v1 <- append(v1, e, after = length(v1))
  ct <- ct + 1
  if (ct == 1000)
    break
}

print(v1)

end <- Sys.time()

time <- end - start; time

#P-value of Simulated Scan Statistic on Chromosome 7

p_chr7 <- p_value(max2); p_chr7
```
#chrom6 - SYNE1: Scan Statistic on Real Data

chr6<-read.csv("chr6.syne1.csv")
colnames(chr6)<-c("id","gene-loc","points","var")
location<-chr6$points; cursor<-min(chr6$points)
count3<-sw2(ws=0.01, data=chr6)

window3<-chr6$points
dframe3<-data.frame(id=chr6$id, window3,count3, position=chr6$`gene-loc`)

max3<-max(count3)
dframe3%>%slice_max(count3)

#Plot
theme_set(theme_classic())
n <- ggplot(dframe3, aes(window3))
n <- n + geom_line(aes(y=count3), colour="red")+ labs(title="Sliding Window Plot",
subtitle="Variant Peakedness on Chromosome 6", caption="Source: Gene GSV location",
x="Window positions", y="Peaks"), theme(plot.title = element_text(hjust = 0.5, face="bold"))
+ theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n

#Repeated Trials of Scan Statistic on Simulated Data
start02<-Sys.time()
ct<-0
v1<-NULL
repeat{
    points<-sort(runif(115, min = 0, max = 1))
    random3<-data.frame(points, var=chr6$var)
    location<-random3$points; cursor<-min(random3$points)
    count3<-sw2(ws=0.01, data=random3)
    e<-max(count3)
    v1<-append(v1,e, after=length(v1))
ct <- ct+1  
if (ct=1000){
    break}}
print(v1)
end03<-Sys.time()

time03<-end02-start02; time03

# P-value of Simulated Scan Statistic on Chromosome 6
p_chr6<-round(p_value(max3), 4); p_chr6

#chrom5 - PCDAH1: Scan Statistic on Real Data
chr5<-read.csv("chr5.PCDAH1.csv")
colnames(chr5)<-c("id", "gene-loc", "points", "var")

location<-chr5$points
cursor<-min(chr5$points)
count4<-sw2(ws=0.01, data=chr5)
window4<-chr5$points
dframe4<-data.frame(id=chr5$id, window4, count4, sequence=chr5$`gene-loc`)  
dframe4%>%slice_max(count4)

theme_set(theme_classic())
n <- ggplot(dframe4, aes(window4))
n <- n + geom_line(aes(y=count4), colour="red")+ labs(title="Sliding Window Plot",  
    subtitle="Variant Peakedness on Chromosome 5", caption="Source: Gene GSV location",  
    x="Window positions", y="Peaks")+ theme(plot.title=element_text(hjust = 0.5, face="bold"))
+theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n

#Repeated Trials of Scan Statistic on Simulated Data
start04<-Sys.time()

ct<-0
v1<-NULL
repeat{
    points<-sort(runif(136, min = 0, max = 1))
random4<-data.frame(points, var=chr5$var)
location<-random4$points
cursor<-min(random4$points)
count4<-sw2(ws=0.01, data=random4)
e<-max(count4)
v1<-append(v1,e, after=length(v1))
ct <- ct+1
if (ct==1000){
  break}
print(v1)
end04<-Sys.time()
time04<-end04-start04
time04

# P-value of Simulated Scan Statistic on Chromosome 5
p_chr5<-p_value(max4); p_chr5

# chrom2 - TTN: Scan Statistic on Real Data
chr2<-read.csv("chr2.TTN.csv")
colnames(chr2)<-c("id","gene-loc","points","var")
location<-chr2$points
cursor<-min(chr2$points)
count5<-sw2(ws=0.01, data=chr2)
window5<-chr2$points
dframe5<-data.frame(id=chr2$id, window5,count5, sequence=chr2$`gene-loc`
)
dframe5%>%slice_max(count5)
theme_set(theme_classic())
n <- ggplot(dframe5, aes(window5))
n <- n + geom_line(aes(y=count5), colour="red")+ labs(title="Sliding Window Plot",
  subtitle="Variant Peakedness on Chromosome 2", caption="Source: Gene GSV location",
  }
# Repeated Trials of Scan Statistic on Simulated Data

```r
count5 <- sw2(ws = 0.01, data = random5)
e <- max(count5)
v1 <- append(v1, e, after = length(v1))
ct <- ct + 1
if (ct == 1000) {
  break
}
print(v1)
```

# P-value of Simulated Scan Statistic on Chromosome 2

```r
p_chr2 <- p_value(max5); p_chr2
```

# chrom1 - OBSCN: Scan Statistic on Real Data

```r
chr1 <- read.csv("chr1.OBSCN.csv")
colnames(chr1) <- c("id", "gene-loc", "points", "var")
```
location <- chr1$points; cursor <- min(chr1$points)
count6 <- sw2(ws=0.01, data=chr1); window6 <- chr1$points
dframe6 <- data.frame(id=chr1$id, window6, count6, sequence=chr1$`gene-loc`)
dframe6%>%slice_max(count6)
theme_set(theme_classic())
n <- ggplot(dframe6, aes(window6))
n <- n + geom_line(aes(y=count6), colour="red") +
 labs(title="Sliding Window Plot", subtitle="Variant Peakedness on Chr1-OBSCN1",
    caption="Source: Gene GSV location", x="Window positions", y="Peaks") +
theme(plot.title = element_text(hjust = 0.5, face="bold")) +
theme(plot.subtitle = element_text(hjust = 0.5, face="bold"); n

# Repeated Trials of Scan Statistic on Simulated Data
start06 <- Sys.time()
ct <- 0
v1 <- NULL
repeat{
    points <- sort(runif(101, min = 0, max = 1))
    random6 <- data.frame(points, var=chr1$var)
    location <- random6$points
    cursor <- min(random6$points)
    count6 <- sw2(ws=0.01, data=random6)
    e <- max(count6)
    v1 <- append(v1, e, after=length(v1))
    ct <- ct + 1
    if (ct==1000){
        break
    }
}
print(v1)
end06<-Sys.time()
time06<-end06-start06; time06
# P-value of Simulated Scan Statistic on Chromosome 1
p_chr1<-round(p_value(max6), 4); p_chr1

chrom19 - MUC16: Scan Statistic on Real Data
chr_19<-read.csv("chr19.MUC16.csv")
colnames(chr_19)<-c("id","gene-loc","points","var")
location<-chr_19$points; cursor<-min(chr_19$points)
count7<-sw2(ws=0.01, data=chr_19); window7<-chr_19$points
dframe7<-data.frame(id=chr_19$id,window7,count7, sequence=chr_19$`gene-loc`)
dframe7%>%slice_max(count7)
theme_set(theme_classic())
n <- ggplot(dframe7, aes(window7))
n <- n + geom_line(aes(y=count7), colour="red")+ labs(title="Sliding Window Plot", subtitle="Variant Peakedness on Chr19-MUC16", caption="Source: Gene GSV location", x="Window positions", y="Peaks") + theme(plot.title = element_text(hjust = 0.5, face="bold")) + theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n

Repeated Trials of Scan Statistic on Simulated Data
start07<-Sys.time()
ct<-0
v1<-NULL
repeat{
    points<-sort(runif(89, min = 0, max = 1))
    random7<-data.frame(points, var=chr_19$var)
    location<-random7$points; cursor<-min(random7$points)
count7<-sw2(ws=0.01, data=random7); e<-max(count7)
v1<-append(v1,e, after=length(v1))
ct <- ct+1
if (ct==1000){
  break}
}
print(v1)
end07<-Sys.time()
time07<-end07-start07; time07

# P-value of Simulated Scan Statistic on Chromosome 19
CHROM19<-p_value(max7); CHROM19

#chrom5 - PCDAGH2: Scan Statistic on Real Data
chr_5<-read.csv("chr5.PCDAGH2.csv")
colnames(chr_5)<-c("id","gene-loc","points","var")
location<-chr_5$points; cursor<-min(chr_5$points)
count8<-sw2(ws=0.01, data=chr_5); window8<-chr_5$points
dframe8<-data.frame(id=chr_5$id, window8,count8, sequence=chr_5$`gene-loc`)
dframe8%>%slice_max(count8)
theme_set(theme_classic())
n <- ggplot(dframe8, aes(window8))
  n <- n + geom_line(aes(y=count8), colour="red")+
    labs(title="Sliding Window Plot", subtitle="Variant Peakedness on Chr5-PCDAGH2",
         caption="Source: Gene GSV location", x="Window positions", y="Peaks") +
    theme(plot.title = element_text(hjust = 0.5, face="bold")) +
    theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n

#Repeated Trials of Scan Statistic on Simulated Data
start08<-Sys.time()
ct<0
v1<-NULL
repeat{
  points<-sort(runif(85, min = 0, max = 1))
  random8<-data.frame(points, var=chr_5$var); location<-random8$points
cursor<-min(random8$points); count8<-sw2(ws=0.01, data=random8)
e<-max(count8)
v1<-append(v1,e, after=length(v1))
ct <- ct+1
if (ct==1000){
  break}
print(v1)
end0<-Sys.time()
time0<-end0-start0; time0

# P-value of Simulated Scan Statistic on Chromosome 5
CHROM5<-p_value(max8); CHROM5

#chrom15 - RYR3: Scan Statistic on Real Data
chr15<-read.csv("chr15.RYR3.csv")
colnames(chr15)<-c("id","gene-loc","points","var")
location<-chr15$points; cursor<-min(chr15$points)
count9<-sw2(ws=0.01, data=chr15); window9<-chr15$points
dframe9<-data.frame(id=chr15$id, window9,count9, sequence=chr15$`gene-loc´)
dframe9%>%slice_max(count9)
theme_set(theme_classic())
n <- ggplot(dframe9, aes(window9))
n <- n + geom_line(aes(y=count9), colour="red") +
labs(title="Sliding Window Plot", subtitle="Variant Peakedness on Chr15-RYR3",
    #caption="Source: Gene GSV location", x="Window positions", y="Peaks") +
    theme(plot.title = element_text(hjust = 0.5, face="bold") +
    theme(plot.subtitle = element_text(hjust = 0.5, face="bold"))); n

#Repeated Trials of Scan Statistic on Simulated Data
start09<-Sys.time()
ct <- 0
v1 <- NULL
repeat{
    points <- sort(runif(75, min = 0, max = 1))
    random9 <- data.frame(points, var = chr15$var)
    location <- random9$points; cursor <- min(random9$points)
    count9 <- sw2(ws = 0.01, data = random9); e <- max(count9)
    v1 <- append(v1, e, after = length(v1))
    ct <- ct + 1
    if (ct == 1000) {
        break
    }
}
print(v1)
end09 <- Sys.time()
time09 <- end09 - start09; time09

# P-value of Simulated Scan Statistic on Chromosome 15
p_chr15 <- p_value(max9); p_chr15

#chrom2 - RYR3: Scan Statistic on Real Data
chr2 <- read.csv("chr2.LRPRB1.csv")
colnames(chr2) <- c("id", "gene-loc", "points", "var")
location <- chr2$points; cursor <- min(chr2$points)
count10 <- sw2(ws = 0.01, data = chr2); window10 <- chr2$points
dframe10 <- data.frame(id = chr2$id, window10, count10, sequence = chr2$`gene-loc`")
dframe10%>%slice_max(count10)
theme_set(theme_classic())
n <- ggplot(dframe10, aes(window10))
n <- n + geom_line(aes(y = count10), colour = "red") + labs(title = "Sliding Window Plot",
    subtitle = "Variant Peakedness on Chr2 -LRPRB1", caption = "Source: Gene GSV location",
x="Window positions", y="Peaks") + theme(plot.title = element_text(hjust = 0.5, face="bold")) + theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n

#Repeated Trials of Scan Statistic on Simulated Data
start02<-Sys.time()
c<-0
v1<-NULL
repeat{
  points<-sort(runif(74, min = 0, max = 1))
  random10<-data.frame(points, var=chr2$var);
  location<-random10$points; cursor<-min(random10$points);
  count10<-sw2(ws=0.01, data=random10); e<-max(count10)
  v1<-append(v1,e, after=length(v1))
  c<- c+1
  if (c==1000){
    break}}
print(v1)
end10<-Sys.time()
time10<-end10-start10; time10

# P-value of Simulated Scan Statistic on Chromosome 2
CHROM2<-p_value(max10); CHROM2

#chrom21 - RYR3: Scan Statistic on Real Data
chr21<-read.csv("chr21.TPTE.csv")
colnames(chr21)<-c("id","gene-loc","points","var")
location<-chr21$points; cursor<-min(chr21$points)
count11<-sw2(ws=0.01, data=chr21); window11<-chr21$points
dframe11<-data.frame(id=chr21$id, window11,count11, sequence=chr21$`gene-loc`)
dframe11%>%slice_max(count11)
theme_set(theme_classic())
n <- ggplot(dframe11, aes(window11))
n <- n + geom_line(aes(y=count11), colour="red")+
labs(title="Sliding Window Plot",
subtitle="Variant Peakedness on Chr21-TPTE", #caption="Source: Gene GSV location",
x="Window positions", y="Peaks") + theme(plot.title = element_text(hjust = 0.5,
face="bold")) + theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n

#Repeated Trials of Scan Statistic on Simulated Data
Start11<-Sys.time()
ct<-0
v1<-NULL
repeat{
  points<-sort(runif(72, min = 0, max = 1))
  random11<-data.frame(points, var=chr21$var)
  location<-random11$points
  cursor<-min(random11$points)
  count11<-sw2(ws=0.01, data=random11)
  e<-max(count11);
  v1<-append(v1,e, after=length(v1))
  ct <- ct+1
  if (ct==1000){
    break}
}
print(v1)
end11<-Sys.time()
time11<-(end11-start11);time11
# P-value of Simulated Scan Statistic on Chromosome 21
p_chr21<-p_value(max11); p_chr21
#chrom1 - HMCN1: Scan Statistic on Real Data

```r
chr_1<-read.csv("chr1.HMCN1.csv")
colnames(chr_1)<-c("id","gene-loc","points","var")
location<chr_1$points; cursor<-min(chr_1$points)
count12<-sw2(ws=0.01, data=chr_1)
window12<-chr_1$points
dframe12<data.frame(id=chr_1$id, window12,count12, sequence=chr_1$'gene-loc')
dframe12%>%slice_max(count12)
theme_set(theme_classic())
n <- ggplot(dframe12, aes(window12))
n <- n + geom_line(aes(y=count12), colour="red")+ labs(title="Sliding Window Plot",
    subtitle="Variant Peakedness onChr1-HMCN1", caption="Source: Gene GSV location",
    x="Window positions", y="Peaks") + theme(plot.title = element_text(hjust = 0.5,
    face="bold")) + theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n
```

#Repeated Trials of Scan Statistic on Simulated Data

```r
Start12<-Sys.time()
ct<-0
v1<-NULL
repeat{
    points<-sort(runif(71, min = 0, max = 1))
    random12<-data.frame(points, var=chr_1$var)
    location<-random12$points; cursor<-min(random12$points)
    count12<-sw2(ws=0.01, data=random12); e<-max(count12)
    v1<-append(v1,e, after=length(v1))
    ct <- ct+1
    if (ct==1000){
        break}
}
print(v1);
end12<-Sys.time()
```
# P-value of Simulated Scan Statistic on Chromosome 1
CHROM1<-p_value(max12); CHROM1

#chrom3 - MUC4: Scan Statistic on Real Data
chr3<-read.csv("chr3.MUC4.csv")
colnames(chr3)<-c("id","gene-loc","points","var")
location<-chr3$points; cursor<-min(chr3$points)
count13<-sw2(ws=0.01, data=chr3); window13<-chr3$points
dframe13<-data.frame(id=chr3$id, window13,count13, sequence=chr3$`gene-loc`)  
dframe13%>%slice_max(count13)
theme_set(theme_classic())
n <- ggplot(dframe13, aes(window13))
n <- n + geom_line(aes(y=count13), colour="red")+ labs(title="Sliding Window Plot",
subtitle="Variant Peakedness on Chr3-MUC4", caption="Source: Gene GSV location",
x="Window positions", y="Peaks") + theme(plot.title = element_text(hjust = 0.5, face="bold")) + theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n

#Repeated Trials of Scan Statistic on Simulated Data
Start13<-Sys.time()
ct<-0
v1<-NULL
repeat{
  points<-sort(runif(68, min = 0, max = 1))
  random13<-data.frame(points, var=chr3$var)
  location<-random13$points; cursor<-min(random13$points)
  count13<-sw2(ws=0.01, data=random13); e<-max(count13)
  v1<-append(v1,e, after=length(v1))
  ct <- ct+1
if (ct==1000){
    break }}
print(v1)
end13<-Sys.time()
time13<-end13-start13; time13

# P-value of Simulated Scan Statistic on Chromosome 3
p_chr3<-p_value(max13); p_chr3

# Chromosome 1 - SPTA1: Scan Statistic on Real Data
chr01<-read.csv("chr1.SPTA1.csv")
colnames(chr01)<-c("id","gene-loc","points","var")
location<chr01$points; cursor<min(chr01$points)
count14<-sw2(ws=0.01, data=chr01); window14<chr01$points
dframe14<data.frame(id=chr01$id, window14,count14, sequence=chr01$`gene-loc`) 
dframe14%>%slice_max(count14)
theme_set(theme_classic())
n <- ggplot(dframe14, aes(window))
n <- n + geom_line(aes(y=count14, colour="red")+ labs(title="Sliding Window Plot", subtitle="Variant Peakedness on Chr1-SPTA1", 
    caption="Source: Gene GSV location", x="Window positions", y="Peaks") + theme(plot.title = element_text(hjust = 0.5, face="bold")) + theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n

# Repeated Trials of Scan Statistic on Simulated Data
start14<-Sys.time()
ct<0
v1<-NULL
repeat{
    points<-sort(runif(64, min = 0, max = 1))
    random14<data.frame(points, var=chr01$var)
location<-random14$points; cursor<-min(random14$points)
count14<-sw2(ws=0.01, data=random14); e<-max(count14)
v1<-append(v1,e, after=length(v1))
ct <- ct+1
if (ct==1000){
  break }
print(v1)
end14<-Sys.time()
time14<-end14-start14; time14
# P-value of Simulated Scan Statistic on Chromosome 1
CHROME1<-p_value(max14); CHROME1

#Chromosome 6 - TNXB: Scan Statistic on Real Data
chr6<-read.csv("CHR6.TNXB.csv")
colnames(chr6)<-c("id", "gene-loc", "points", "var")
location<-chr6$points; cursor<-min(chr6$points)
count15<-sw2(ws=0.01, data=chr6); window15<-chr6$points
dframe15<-data.frame(id=chr6$id, window15, count15, sequence=chr6$`gene-loc`) 
dframe15%>%slice_max(count15)
theme_set(theme_classic())
n <- ggplot(dframe15, aes(window15))
n <- n + geom_line(aes(y=count15), colour="red")+ labs(title="Sliding Window Plot",
        subtitle="Variant Peakedness on Chr6-TNXB", caption="Source: Gene GSV location",
        x="Window positions", y="Peaks") + theme(plot.title = element_text(hjust = 0.5, 
        face="bold")) + theme(plot.subtitle = element_text(hjust = 0.5, face="bold")); n

#Repeated Trials of Scan Statistic on Simulated Data
start15<-Sys.time()
ct<-0
v1<-NULL
repeat{
  points<-sort(runif(64, min = 0, max = 1))
  random14<-data.frame(points, var=chr01$var)
  location<-random14$points; cursor<-min(random14$points)
  count14<-sw2(ws=0.01, data=random14); e<-max(count14)
  v1<-append(v1,e, after=length(v1))
  ct <- ct+1
  if (ct==1000){
      break } ); print(v1)
end15<-Sys.time()
  time15<-end15-start15; time15
  
  #P-value of Simulated Scan Statistic on Chromosome 6
  CHROME6<-p_value(max15); CHROME6

Appendix G: Function to Compute Q1, Q2, Q3 and Theoretical P-values
# n is total GSVs on a gene, r is the scan statistic value and p is the probability (window size).
#Function to compute Q1
q1<-function(n,r,p){
  X<-0:r-1
  a<-sum(dbinom(x=X, size=n, prob=p))
  print(a)}
#Function to compute Q2
q2<-function(n,r,p){
  k<-seq(0, n-r, by=1)
  for(i in k){
      b<- Q1 - sum((-1)^k * dbinom(r+k,n,p)) 
      print(b) } }
print(b)
#Function to compute Q3
q3<-function(n,r,p){
k<-seq(0,n-r, by=1)
for (i in k){
  cc<- Q2 - (sum((-1)^k* (2+k^2) * dbinom(r+k,n,p))/2 )
print(cc)
}

#Function for Approximated Theoretical Probability
prob<-function(n,r){
  p<-round(1-Q2*(Q3/Q2)^(n-r-1),6)
  print(p)
}

# Chromosome 19 - KIR2DS3: Input Parameters: n=196, r=32, p=0.01
Q1<-q1(n=196, r=32, p=0.01)
Q2<-q2(n=196, r=32, p=0.01)
Q3<-q3(n=196, r=32, p=0.01)
p1<-prob(n=196, r=32)

# Chromosome 7 - TCAF2: Input Parameters: n=196, r=32, p=0.01]
Q1<-q1(n=145, r=69, p=0.01)
Q2<-q2(n=145, r=69, p=0.01)
Q3<-q3(n=145, r=69, p=0.01)
p2<-prob(n=145, r=69)

# Chromosome 6 - SYNE1: Input Parameters: n=136, r=9, p=0.01]
Q1<-q1(n=136, r=9, p=0.01)
Q2<-q2(n=136, r=9, p=0.01)
Q3<-q3(n=136, r=9, p=0.01)
p3<-prob(n=136, r=9)

# Chromosome 5 - PCDAH1: Input Parameters: n=115, r=12, p=0.01]
Q1<-q1(n=115, r=12, p=0.01)
Q2<-q2(n=115, r=12, p=0.01)
Q3<-q3(n=115, r=12, p=0.01)
p4<-prob(n=115, r=12)

# Chromosome 2 - TTN: Input Parameters: n=103, r=9, p=0.01]
Q1<-q1(n=103, r=9, p=0.01)
Q2<-q2(n=103, r=9, p=0.01)
Q3<-q3(n=103, r=9, p=0.01)
p5<-prob(n=103, r=9)

#Chromosome 1 - OBSCN: Input Parameters: n=101, r=7, p=0.01]
Q1<-q1(n=101, r=7, p=0.01);
Q2<-q2(n=101, r=7, p=0.01);
Q3<-q3(n=101, r=7, p=0.01);
p6<-prob(n=101, r=7)

# Chromosome 19 - MUC16: Input Parameters: n=89, r=5, p=0.01]
Q1<-q1(n=89, r=5, p=0.01);
Q2<-q2(n=89, r=5, p=0.01)
Q3<-q3(n=89, r=5, p=0.01)
p7<-prob(n=89, r=5)

# Chromosome 5 - PCDAGH2: Input Parameters: n=85, r=6, p=0.01]
Q1<-q1(n=85, r=6, p=0.01);
Q2<-q2(n=85, r=6, p=0.01)
Q3<-q3(n=85, r=6, p=0.01)
p8<-prob(n=85, r=6)

# Chromosome 15 - RYR3: Input Parameters: n=75, r=6, p=0.01]
Q1<-q1(n=75, r=6, p=0.01);
Q2<-q2(n=75, r=6, p=0.01)
Q3<-q3(n=75, r=6, p=0.01)
p9<-prob(n=75, r=6)

# Chromosome 2 - LRPRB1: Input Parameters: n=74, r=9, p=0.01]
Q1<-q1(n=74, r=9, p=0.01);
Q2<-q2(n=74, r=9, p=0.01)
Q3<-q3(n=74, r=9, p=0.01)
p10<-prob(n=74, r=9)

#Chromosome 21 - TPTE : Input Parameters: n=72, r=12, p=0.01]  
Q1<q1(n=72, r=12, p=0.01);  Q2<q2(n=72, r=12, p=0.01)  
Q3<q3(n=72, r=12, p=0.01)

p11<-prob(n=72, r=12)

#Chromosome 1 - HMCN1 : Input Parameters: n=71, r=8, p=0.01]
Q1<q1(n=71, r=8, p=0.01)  
Q2<q2(n=71, r=8, p=0.01)  
Q3<q3(n=71, r=8, p=0.01)

p12<-prob(n=71, r=8)

#Chromosome 3 - MUC4 : Input Parameters: n=69, r=17, p=0.01]  
Q1<q1(n=68, r=17, p=0.01)  
Q2<q2(n=68, r=17, p=0.01)  
Q3<q3(n=68, r=17, p=0.01)

p13<-prob(n=68, r=17)

#Chromosome 1 - SPTA1 : Input Parameters: n=64, r=5, p=0.01]  
Q1<q1(n=64, r=5, p=0.01)  
Q2<q2(n=64, r=5, p=0.01)  
Q3<q3(n=64, r=5, p=0.01)

p14<-prob(n=64, r=5)

#Chromosome 6 - TNXB : Input Parameters: n=63, r=8, p=0.01]  
Q1<q1(n=63, r=8, p=0.01)  
Q2<q2(n=63, r=8, p=0.01)  
Q3<q3(n=63, r=8, p=0.01)

p15<-prob(n=63, r=8)
Appendix H: Sliding window plot for gene KIR2DS3 (length = 123645 bases) with a window size of 0.01 on the gene normalized to unit length.

```r
library("gridExtra")
require(cowplot)
library(ggplot2)
theme_set(theme_classic())
n01 <- ggplot(dframe1, aes(window1))
n01 <- n01 + geom_line(aes(y=count1), colour="red")+
  labs(title="GSV Concentration on KIR2DS3",
       x="Variant locations on gene normalised to Unit Gene-Length",
       y="Variant counts") +
  theme(plot.title = element_text(hjust = 0.5, face="bold")) +
  theme(plot.subtitle = element_text(hjust = 0.5, face="bold")) +
  background_grid(major = "xy", minor = "none")
n01
```

Appendix I: Grided Sliding Window Plots

```r
theme_set(theme_classic())
n02 <- ggplot(dframe2, aes(window2))
n02 <- n02 + geom_line(aes(y=count2), colour="red") +
  labs(subtitle="GSV concentration on TCAF2",
       x="Positions on gene (TCAF2) normalised to unit length",
       y="Variant counts") +
  theme(plot.title = element_text(hjust = 0.5, face="bold")) +
  theme(plot.subtitle = element_text(hjust = 0.5, face="bold")) +
  background_grid(major = "xy", minor = "none")
n02
```

```r
theme_set(theme_classic())
n03 <- ggplot(dframe3, aes(window3))
n03 <- n03 + geom_line(aes(y=count3), colour="red")+
```

89
labs(subtitle="GSV concentration on SYNE1",
    x="Positions on gene (SYNE1) normalised to unit length",
    y="Variant counts") +
theme(plot.title = element_text(hjust = 0.5, face="bold")) +
theme(plot.subtitle = element_text(hjust = 0.5, face="bold")) +
background_grid(major = "xy", minor = "none")

n03

tHEME_SET(theme_classic())

n04 <- ggplot(dframe4, aes(window4))
n04 <- n04 + geom_line(aes(y=count4), colour="red") +
    labs(subtitle="GSV concentration on PCDAH1",
         x="Positions on gene (PCDAH1) normalised to unit length",
         y="Variant counts") +
theme(plot.title = element_text(hjust = 0.5, face="bold")) +
theme(plot.subtitle = element_text(hjust = 0.5, face="bold")) +
background_grid(major = "xy", minor = "none")
n04

n05 <- ggplot(dframe5, aes(window5))
n05 <- n05 + geom_line(aes(y=count5), colour="red") +
    labs(subtitle="GSV concentration on TTN",
         x="Positions on gene (TTN) normalised to unit length",
         y="Variant counts") +
theme(plot.title = element_text(hjust = 0.5, face="bold")) +
theme(plot.subtitle = element_text(hjust = 0.5, face="bold")) +
background_grid(major = "xy", minor = "none")
n05

theme_set(theme_classic())
n06 <- ggplot(dframe10, aes(window10))
n06 <- n06 + geom_line(aes(y=count10), colour="red") +

labs(subtitle="GSV concentration on LRP1B",
x="Positions on gene (LRP1B) normalised to unit length",
y="Variant counts") +
theme(plot.title = element_text(hjust = 0.5, face="bold")) +
theme(plot.subtitle = element_text(hjust = 0.5, face="bold")) +
background_grid(major = "xy", minor = "none")
n06

theme_set(theme_classic())
n07 <- ggplot(dframe12, aes(window12))
n07 <- n07 + geom_line(aes(y=count12), colour="red") +
labs(subtitle="GSV concentration on HMCN1",
x="Positions on gene (HMCN1) normalised to unit length",
y="Variant counts") +
theme(plot.title = element_text(hjust = 0.5, face="bold")) +
theme(plot.subtitle = element_text(hjust = 0.5, face="bold")) +
background_grid(major = "xy", minor = "none")
n07

theme_set(theme_classic())
n08 <- ggplot(dframe13, aes(window13))
n08 <- n08 + geom_line(aes(y=count13), colour="red") +
labs(subtitle="GSV concentration on MUC4",
x="Positions on gene (MUC4) normalised to unit length",
y="Variant counts") +
theme(plot.title = element_text(hjust = 0.5, face="bold")) +
theme(plot.subtitle = element_text(hjust = 0.5, face="bold")) +
background_grid(major = "xy", minor = "none")
n08
theme_set(theme_classic())

n09 <- ggplot(dframe15, aes(window15))

n09 <- n09 + geom_line(aes(y=count15), colour="red") +
    labs(subtitle="GSV concentration on TNXB",
         x="Positions on gene (TNXB) normalised to unit length",
         y="Variant counts") +
    theme(plot.title = element_text(hjust = 0.5, face="bold")) +
    theme(plot.subtitle = element_text(hjust = 0.5, face="bold")) +
    background_grid(major = "xy", minor = "none");

n09

grid.arrange(n02, n03, n04, n05, ncol=2, nrow =2)

grid.arrange(n06, n07, n08, n09, ncol=2, nrow =2)
Curriculum Vita

Kelvin Ofori-Minta, is the third of four lovely siblings, born in Ghana to Mr. Isaac Ofori-Dwumfuo and Madam Margaret Asantewaa Appah. He graduated from Abuakwa State College Senior High School in 2013 with a certificate in general and applied life sciences. In 2017, he obtained a bachelor’s degree in Actuarial Science with second class honors upper division from the CK-Tedam University of Technology and Applied Sciences (CKT-UTAS) formally known as University for Development Studies (UDS– Navrongo campus) where he won the Prudential award for excellence in actuarial studies in Ghana by emerging as a top student in his graduating class. Upon graduation, Kelvin briefly served as a reinsurance risk analyst with Quality Insurance Company (QIC). Until his eventual desire to pursue graduate studies in Statistics at the University of Texas at El Paso (UTEP) in the fall semester of 2019. Kelvin currently resides in El Paso, Texas and about to begin his doctoral degree program at UTEP.

Email address: koforimint@miners.utep.edu