Genome-wide association analysis within The Manitoba Personalized Lifestyle Research study

By

Badurally Adam Bibi Taskeen

A thesis submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements of the degree of

Master of Science

Department of Food and Nutritional Sciences

University of Manitoba

Winnipeg

Copyright @ 2024 Badurally Adam Bibi Taskeen

Overall Abstract

Background: Obesity is a major public health challenge worldwide and in Manitoba. An estimated 40-70% of obesity is heritable; however, these genetic associations are poorly investigated in the Manitoban adult population using the "gold standard" Dual Energy X-Ray Absorptiometry (DXA) to assess obesity phenotypes. The Manitoba Personalized Lifestyle Research (TMPLR) project enabled a genetic association study in middle-aged Manitobans. Obesity phenotypes considered are total, gynoid, android, arms, legs, and trunk fat mass.

Objectives:

1. Perform a systematic review to determine genes associated with obesity phenotypes assessed by DXA in middle-aged cohorts.
2. Develop a "pipeline" to conduct Genome-Wide Association Studies (GWAS) within the TMPLR.
3. Conduct a GWAS on obesity phenotypes within TMPLR.
4. Compare the genes from the systematic review to those from TMPLR data analysis.

Methods: The Covidence platform was used for the systematic review. Publications up to July 2023 sourced from Embase and Medline. The methods for the TMPLR project have been published. A GWAS pipeline was established using the Biodata Catalyst bioinformatics platform and the Seven Bridges R studio version 4.1.

Results: Out of 94 and 25 studies obtained from Medline and Embase respectively, 14 studies met the eligibility criteria and 13 genes were identified that are associated with obesity-related phenotypes. No significant genome-wide association with obesity was established in the TMPLR cohort. However, 23 loci have had suggestive associations with the obesity phenotypes.

Conclusion: There is a lack of high-quality genetic studies that use DXA data and adult populations. No genome-wide associations were reported in TMPLR, likely due to the limited sample size of the cohort. However, a bioinformatics pipeline to analyze genome-wide associations in TMPLR cohort is established and can be used for larger cohorts, such as UK biobanks.

Acknowledgments

First and foremost, I would like to express my deepest gratitude and appreciation to God for granting me the unbelievable wisdom, strength, and perseverance to complete this thesis.

Words cannot express my gratitude to my advisor**, Dr. Peter Eck** for his invaluable kindness, respect, patience, and, feedback during this journey. **Dr Peter** has always guided me with his immense knowledge and expertise in the field of genetics and nutrition. He supported me despite starting my course during the Covid period and having to take maternity leave after a year. His belief in my skills and abilities throughout this journey has been noticed and appreciated.

I am also very grateful to my co-advisor **Dr. Pingzhao Hu** who has extensive skills and knowledge in computational approaches to health research. I was fortunate to have him among my lecturers for the course IMED 7280 Medical Computational Biology, his dedication was amazing and he would always quickly reply to my messages about troubleshooting and coding, something I will never forget and thanks to him and **Dr. Pete**r, my motivation stayed high and I scored A+ in the course. I am also thankful to the other members of my committee, **Dr Dylan Mackay, Dr Semone Myrie, and Dr. James G. Taylor.**

Dr. Dylan is the author of the Manitoba Personalized Lifestyle Research Project and thanks to him and his hard work, I was able to contribute to a truly amazing project and vision. I am also grateful for his support during my data extraction. **Dr Semone** kindly lent me a laptop to use while I was finishing the setup of our computer with the required software, I truly appreciated the support. I am also really grateful to **Dr James** who proposed the use of the Biodata Catalyst in my data analysis which was great. A special thanks to my friend **Rutong** who gave me support and motivation during this journey as well as help with sending my samples to Genome Quebec during my maternity leave and returning the blood samples to **Dr. Semone**.

I am grateful for funding from Research Manitoba and the Biodata Catalyst. I would like to extend my sincere thanks to the Velsera team which is the technical support of the Biodata catalyst particularly to **Dr. Cera R. Fisher**, **Dave Roberson,** and **Ryan Poterfield** who helped me with troubleshooting. I also want to acknowledge **Mr. Seth Berke** who provided me with the application “Split by Chr” on the Biodata Catalyst which helped me early on when I was working with the raw data provided by Genome Quebec.

It was truly a blessing to have known all these people and it was a pleasure to interact with them. Lastly, I would give many thanks to my amazing parents, husband, and son for their continuous support system that allowed me to excel in everything I was doing.

Dedications

I dedicate this thesis to my parents, **Nasrullah Badurally Adam** and **Nasseem BeeBee Banon Badurally Adam**, for their unconditional support and encouragement throughout my academic career.

To my amazing husband, **Mohammad Nadiim Molin**, who constantly believed in me when I doubted myself and whose support was just phenomenal and gave me a lot of strength.

To my loving 2-year-old son, **Mohammad Ibraheem Molin**, who is simply amazing and gives me a lot of strength with just a hug!

To my parents-in-law, **Shaynaaz Molin** and **Noorani Molin** for their constant support and encouragement.

To my advisor, **Dr Peter Eck**, for his exceptional respect, and kindness and whose guidance and insights were invaluable.

To the loving memory of my maternal grandmother, **Bibi Fatmah Mosaheb**, both paternal grandparents, **Bibi Mahani Badurally Adam, Hassam Badurally Adam**, and my husband's paternal grandparent, **Haleema Molin,** all of who passed away in Mauritius while I was studying for this degree in Canada.

To the loving memory of my paternal grandfather **Yacoob Mosaheb** and my husband's grandparents, **Ismael Molin, Ansoo Rumjan,** and **Abdool Gafour Rumjan.**

Tables of Contents

Overall abstract …….....2
Acknowledgements ……...3
Dedications……4
Table of Contents……5
List of tables……...…7
List of figures……8
Abbreviations……9
Chapter 1: General Introduction……...…11
1.1 Introduction ……11
1.2 Objectives…….. 13
Chapter 2: Systematic Literature Review……14
2.1 Abstract…… 14
2.2 Introduction……. 15
2.3 Materials and Methods……16
2.4 Results……19
2.5 Discussion…… 29
2.6 Strength and Limitations……30
2.7 Conclusion…… 31
2.8 Conflict of interest ……31
Chapter 3: Genome-wide Association Study of obesity-related phenotypes from the Manitoba personalized lifestyle research…….32
3.1 Abstract……...…32
3.2 Introduction……33
3.3 Materials and Methods……34
3.4 Results……35
3.5 Discussion……..47
3.6 Conclusion……..50
3.7 Acknowledgements ……50
3.8 Conflict of interest ……51
3.9 Funding ……51
Chapter 4: General Discussion…….……... 52
4.1 Discussion ……...……...52
4.2 Strengths and Limitations …… 53

4.3 Future Directions ……...……53

Chapter 5: Overall Conclusion…… 56

Chapter 6: References ……..……..57

Appendix A: Gel image for DNA quality control ..……69

Appendix B: Applications used on Biodata catalyst……79

Appendix C: R codes for all the phenotypes……89

Appendix D: R codes for annotation……99

Appendix E: Descriptive Statistics ……... 128

Appendix F: Kindship coefficient and principal component analysis results…….132

List of Tables

Table Name	Page
Table 1: Medline search strategy	17
Table 2: Embase search strategy	18
Table 3: Characteristics of included studies	22
Table 4: Genetic Loci and the variations described to be associated with obesity phenotypes.	24
Table 5: Assessment of the strength of the Publications	28
Table 6: Basic descriptive parameters for TMPLR participants in the GWAS	35
Table 7: Suggestive genetic associations for obesity phenotypes in the TMPLR cohort	38
Table 8: SNPs common between systematic review and TMPLR data analysis	55

List of Figures

Figure Name	Page
Figure 1: Systematic Review Workflow Diagram	20
Figure 2: Genome-wide associations with Total Fat Mass	41
Figure 2a: Suggestive genetic associations with Total fat mass	41
Figure 3: Genome-wide associations with Android Fat Mass	42
Figure 3a: Suggestive genetic associations with Android fat mass	42
Figure 4: Genome-wide associations with Gynoid Fat Mass	43
Figure 4a: Suggestive genetic associations with Gynoid fat mass	43
Figure 5: Genome-wide associations with Arms Fat Mass	44
Figure 5a: Suggestive genetic associations with Arms fat mass	44
Figure 6: Genome-wide associations with Legs Fat Mass	45
Figure 6a: Suggestive genetic associations with Legs fat mass	45
Figure 7: Genome-wide associations with Trunk Fat Mass	46
Figure 7a: Suggestive genetic associations with Trunk fat mass	46

Abbreviations

Abbreviation	Definition	Page
GWAS	Genome-wide association studies	2
DXA	Dual X-Ray Absorptiometry	2
TMPLR	The Manitoba Personalized Lifestyle Research	2
DNA	Deoxyribonucleic acid	11
SNPs	Single nucleotide polymorphisms	12
BMI	Body mass index	12
LEP	Obesity including leptin	17
LEPR	The leptin receptor	17
POMC	Proopiomelanocortin	17
PCSK1	Prohormone convertase 1	17
MC4R	Melanocortin 4 receptor	17
SIM1	The single-minded homolog 1	17
BDNF	Brain-derived neurotrophic factor	17
NTRK2	Neurotrophic tyrosine kinase receptor type 2 gene	17
B.T.B.A	Bibi Taskeen Badurally Adam	18
P.E	Peter Eck	18
mp.	Title, abstract, heading word, drug trade name, original title, device manufacturer, drug manufacturer, device trade name, keyword heading word, floating subheading word, candidate term word	19
PRISMA	Preferred Reporting Items for Systematic reviews and Meta-analyses	20
FTO	Fat mass and obesity-associated	23
LINCO1122	Long Intergenic Non-Protein Coding RNA 1122	23
FADS1	Fatty acid desaturase 1	23
BRINP2	Bone Morphogenetic Protein/Retinoic Acid Inducible Neural-Specific 2	23
SEC16B	SEC16 Homolog B, Endoplasmic Reticulum Export Factor	23
LGR4	Leucine-rich repeat-containing G protein-coupled receptor 4	23
MSTN	Myostatin	23
ALOX12	Arachidonate 12-lipoxygenase	23
PRDM16	PRD1-BF-1-RIZ1 homologous domain containing protein-16	23

Abbreviations (continued)

Abbreviation	Definition	Page
CAP2	Cyclase Associated Actin Cytoskeleton Protein 2	23
NHGRI-EBI	The National Human Genome Research Institute-European Bioinformatics Institute	31
CT	computer tomography	31
MRI	Magnetic Resonance Imaging	31
RAPGEF4	Rap Guanine Nucleotide Exchange Factor 4	37
EPAC	Exchange Protein directly Activated by cAMP	37
PARK2	Parkin RBR E3 Ubiquitin Protein Ligase	37
FBLN7	Fibulin 7	37
PRIMA1	Proline-Rich Membrane Anchor 1	37
CTNND2	Catenin Delta 2	37
SIPA1L1	Signal-Induced Proliferation Associated 1 Like 1	37
ATP8A1	ATPase Phospholipid Transporting 8A1	37
RHNO1	RAD9-HUS1-RAD1 Interacting Nuclear Orphan 1	37
STAU2	Staufen Double-Stranded RNA Binding Protein 2	37
KIF26A	Kinesin Family Member 26A	37
LD	linkage disequilibrium	39
MAC	Minor Allele Count	39
PCA	Principle component analysis	39
RNU6-939P	RNA, U6 Small Nuclear 939 Pseudogene	40
SPAG16	Sperm Associated Antigen 16	40
TPST2	Tyrosylprotein Sulfotransferase 2	40
HRASLS5	HRAS-Like Suppressor Family Member 5	40
USP13	Ubiquitin Specific Peptidase 13	40
DDR	DNA damage response	53

Chapter 1: General Introduction

1.1 Introduction

Obesity is an excessive fat accumulation that can harm individuals' health and is known to be caused by an imbalance between energy intake and energy expenditure [1,2]. In 2022, 2.5 billion adults (43%) worldwide were overweight including, 890 million (16%) living with obesity, and 37 million children under the age of 5 were overweight [1]. The World Obesity Atlas 2024 did a global, regional, and national estimate of obesity and declared "No area of the world is unaffected by the consequences of obesity" [3]. Furthermore, in 2018, data from Statistics Canada [4] shows that in Manitoba 30.8% of residents aged 18 and above were obese and Tran et al. [5] reported the annual costs of obesity in Canada ranged from 1.27 to 11.08 billion dollars.

Obesity is associated with several diseases and conditions leading to high morbidity and mortality rates [6]. Obesity causes complications through both anatomical and metabolic ways. Some of the anatomical effects of obesity are obstructive sleep apnoea, obesity hypoventilation syndrome, and osteoarthritis [7-9]. The major mechanism understood to explain obesity's metabolic effect is the release of proinflammatory cytokines from adipose tissue, particularly tumor necrosis factor alpha (TNF-α), interleukin (IL)-1 and IL-6 which were associated with cardiometabolic diseases, cancer, and infectious diseases [10].

To attempt to decrease the prevalence of obesity, we must first understand its causes. Obesity is a complex disease and there are many risk factors for obesity including the broad categories: environment, genetics, aging, gut microbiota, and diets. Obesity has a strong genetic component and estimated 40% to 70% of obesity is heritable [11]. There are two main genetic categories of obesity, namely monogenic and polygenic. Monogenic obesity is rare but severe and has an early onset, high genetic contribution, a single mutation in one gene with a large genetic effect, and no environmental influence [12]. On the other hand, polygenic obesity is common obesity with modest genetic contribution and hundreds of variants in or near many genes with each one having a small effect and the environment remains a key determinant [12].

To study the effect of genes on obesity, Genome-Wide Association Studies (GWAS) are conducted. GWAS has radically changed the study of complex disease genetics by associating genetic risk factors with diseases that are common in the population [13-14]. GWAS measures and analyzes hundreds of thousands to millions of Deoxyribonucleic Acid (DNA) sequence variations from the human genome of many individuals to determine their associations with diseases [13-14]. GWAS would determine an individual's common Single Nucleotide Polymorphisms (SNPs), variations at a single position along the DNA sequence. It is well-accepted that this will lead to the development of new prevention and treatment strategies [13,14]. The benefits [15] of GWAS include novel trait association, discovery of new biological mechanisms, understanding of the complexity of traits for different ethnicities, and identifying novel monogenic and oligogenic disease genes. Initially, GWAS discoveries included mostly easy-to-measure phenotypes in predominantly European populations [16]. However, studies now aim to expand GWAS to a wider range of phenotypes, larger sample sizes, more diverse populations, and ethnic groups, different study designs, and analyses of gene-gene and gene-environment interactions [16].

Numerous genetic studies analyzing obesity-related traits have been conducted with a large portion of studies using Body Mass Index (BMI) as the main obesity-related phenotype. Other studies would use waist circumference as well or bioelectrical impedance analysis. However, BMI does not distinguish between lean and fat mass [17]. Bioelectrical impedance analysis gets its estimates of body composition by measuring the resistance of body tissues to a small electrical current and is influenced by hydration status, temperature, body position, and device calibration. Waist circumference is an anthropometric measurement that gives no direct measurement of body fat mass or fat percentage, instead, it indicates the presence of abdominal fat but a larger waist circumference may be attributed to muscle mass.

Although BMI, bioelectrical impedance, and waist circumference are not costly and are relatively easy to obtain, the gold standard for body measurement that gives high precision and distinguishes between lean and fat mass is the dual-energy X-ray absorptiometry (DXA) measurements. DXA gives the distribution of fat in the body including total, android, gynoid, arms, legs, visceral, and trunk fat mass. In a recent study by Mohammad et al. [18], they found that visceral fat mass and android-to-gynoid fat ratio were strongly associated with pre-diabetes and males with higher gynoid fat had higher pre-diabetes prevalence highlighting the importance of understanding the effects of fat distribution on obesity.

We used The Manitoba Personalized Lifestyle Research (TMPLR) cross-section cohort, which was published in 2019 [19]. TMPLR was conducted in Manitoba which consists of a unique population characteristic including indigenous and immigrant groups leading to genetic diversity and the possibility of identifying unique sets of genetic variants. The study also used DXA and targets a specific age group of adults between 30 to 46 years old. Since there are not many studies that use DXA within this narrow age group, TMPLR provides a means by which we can perform genome-wide associations for genetic variants that are associated with obesity in Manitoba. Manitoba may have environmental, diet, and lifestyle factors that could interreact with the genetic makeup of individuals leading to some individuals being more prone to developing chronic diseases. A cohort of a total of 391 participants was used in this study to identify genes that are associated with obesity. More details about TMPLR including methods, study population, and measurement can be accessed from its original publication [19].

To establish background knowledge for our GWAS study within TMPLR, a systematic review of genes associated with obesity phenotypes from DXA in adults aged 30 to 46 was conducted. A genome-wide association analysis within the TMPLR was performed to interrogate if genes associated with obesity phenotypes from DXA in the age group 30 to 46 could be identified .

1.3 Objectives

1. Perform a systematic review to determine genes that are associated with obesity phenotypes including total, gynoid, android, arms, legs, and trunk fat mass measured using DXA in adults between the ages of 30 to 46, and assess the quality of the genetic studies.
2. Develop a “pipeline” in the Biodata Catalyst bioinformatics platform to conduct GWAS within the TMPLR which can later be used for other phenotypes and can be used for replicating the study in larger cohorts.
3. Determine what are the genes that are associated with obesity-related phenotypes including total, gynoid, android, arms, legs, and trunk fat mass measured using DXA in adults between the ages of 30 to 46 for participants of TMPLR.
4. Compare the genes that are related to obesity phenotypes including total, gynoid, android, arms, legs, and trunk fat mass obtained from the systematic review to those obtained from the genome-wide association analysis within TMPLR.

Chapter 2: Systematic Literature Review

2.1 Abstract

Background: Obesity is a major health concern worldwide. The excess amount of fat in tissue is responsible for approximately 30% to 40% of coronary heart disease, different types of cancers, diabetes, and osteoarthritis. While genetics is a major component that results in obesity, there are no systematic reviews that address Genome-Wide Association Studies (GWAS) in the adult population with obesity phenotypes assessed by Dual Energy X-ray Absorptiometry (DXA).

Objective: A systematic review of genetic associations with obesity phenotypes assessed by DXA in middle aged adults.

Methods: Embase and Medline were interrogated for publications prior to July 2023. Key criteria were genetic association studies on adult populations (30 to 46 years) where obesity was assessed using DXA. The online Covidence platform was used for data extraction, synthesis and quality assessment.

Results: Initially 94 and 25 studies were identified from Medline and Embase respectively. Fourteen studies met the eligibility criteria for data extraction and analysis. Thirteen genes were reported to be associated with obesity phenotypes assessed by DXA. Genes FTO, FADS1, LGR4, BRINP2|SEC16B PRDM16 ALOX12 LINC01122 CAP2 MSTN, and Single Nucleotide Polymorphisms (SNPs) 13:72584929_AAC_A, rs117686994, rs143707182 and rs138014219 are associated with one or more of the following phenotypes: total fat, android, gynoid, leg, arm and trunk fat mass.

Conclusion: This systematic review provides a baseline for future work in determining the association between genes and obesity phenotypes.

2.2 Introduction

Health Canada defines obesity as "the progressive chronic disease that is characterized by abnormal or excessive fat accumulation that may impair health [20]. In 2022, 2.5 billion adults (43%) worldwide were overweight including 890 million (16%) living with obesity [1]. The prevalence of obesity in Canada has increased from 21% in 2003 to 30% in 2022 for Canadians aged 18 and older[3]. The excess fat in tissue is associated with approximately 30% to 40% of coronary heart diseases, different types of cancers, diabetes, and osteoarthritis [21]. Obesity has an estimated 40% to 70% heritability [11].

According to the World Health Organisation (WHO) [21], adult obesity (18 years or older) is among the principal risk factors for numerous diseases including type 2 diabetes, cardiovascular disease, and certain cancers. Additionally, adults often face more severe health consequences from obesity compared to the younger population and many risk factors that are associated with obesity accumulate over time. However, the age group 30 to 46 years old is understudied.

Genome-wide association Studies (GWAS) are conducted to study disease genetics by associating genetic risk factors with diseases that are common in the population [13,14]. Using the GWAS over 500 genetic variations have been associated with obesity-related phenotypes [22]. However, only a few of these genes have been accepted as genes with high effect on obesity, including leptin (LEP), the leptin receptor (LEPR), proopiomelanocortin (POMC), prohormone convertase 1 (PCSK1), the melanocortin 4 receptor (MC4R), single-minded homolog 1 (SIM1), brain-derived neurotrophic factor (BDNF), and the neurotrophic tyrosine kinase receptor type 2 gene (NTRK2) [16].

A large portion of these GWAS uses Body Mass Index (BMI) as the main obesity-related phenotype since it is not costly and is relatively easy to obtain. However, the BMI does not distinguish between lean and fat mass and therefore does not allow a more specific distinction of obesity traits. Other types of measurements for fat mass include waist circumference and bioelectrical impedance analysis which are relatively low in precision and accuracy compared to Dual-energy X-ray absorptiometry (DXA) which can distinguish between compartments of fat and lean mass [23, 24]. Obesity-related phenotypes from DXA include total fat mass, arm fat mass, leg fat mass, gynoid fat mass, android fat mass, and trunk fat mass. Obesity-related phenotypes of regional distribution of fat have a high degree of heritability [25] with specific associations in the progression of many chronic diseases [26].

The objective of the current systematic review is to obtain a list of the associations between genes and obesity-related phenotypes assessed by DXA in a adult populations.

2.3 Materials and Methods

2.3.1 Eligibility criteria

Criteria to include studies into the systematic review were: (a) genetic studies excluding case reports, editorials or reviews, or studies solely including animals (b) obesity phenotypes derived from DXA data excluding any other type of measurements (c) adult population aged range 30 to 46 years, excluding older adults, teenagers and children (d) English language excluding other languages. During the search process, no publication year limit was applied to include all relevant publications paving the way for a comprehensive overview of the available evidence.

2.3.2 Information sources and search strategy

Two electronic databases were searched, Medline and Embase via Ovid on 4th July 2023 by B.T.B.A in the presence of P.E. The keywords search was conducted and search items were divided into three main categories: Obesity, GWAS and DXA. For the obesity category, the following keywords were used: “obesity”, “adiposity”, “fat mass” and “obesity indices”. To include the GWAS aspect, we applied the items: “Genome-wide association studies”, “genome-wide association study”, “Genome, Human or exp Genome-wide Association Study”, “Genetic Predisposition to disease”, “GWAS” or “Genomics” and “Polymorphism, Single Nucleotide”. The same search strategy was used for both Ovid Medline and Embase databases [27] through the University of Manitoba Library and the number of hits is shown in **Table 1** and **Table 2** and a total of 119 studies were identified. The second reviewer (P.E) independently reviewed the search strategy and methodology confirming it was appropriate, comprehensive, and well-executed. The second reviewer’s approval serves as an extra layer of quality control with no critical elements being overlooked.

2.3.3 Study Selection

The Covidence [28] management system for systematic reviews was used to process information from Medline and Embase for studies published up to July 2023. However, July 2023 is the cut-off date and the only temporal imitation applied since all included studies were based on their relevance to the review's scope. A two-stage screening process to assess the eligibility of the studies identified through the literature search was performed [29].

In the first stage, titles and abstracts were screened by B.T.B. A and P.E, blinded to each other's selection. Irrelevant studies were excluded and full-text articles were obtained and uploaded onto Covidence. In the second stage, the full-text articles were screened for the inclusion and exclusion criteria and disagreements were resolved by discussion between the two reviewers. A third reviewer was not needed because of the effectiveness of resolving all the conflicts that were resolved amicably ensuring that each step involved consistent decision-making. Reasons for excluding studies were recorded. Records for retrieval were independently assessed and any disagreements were resolved via consensus. The reasons for excluding studies are recorded in **Figure 1**.

Table 1: Medline search strategy

Search	Query	Number of hits
#1	(Obesity or Adiposity or Overweight or fat mass or Obesity indices).mp.	452101
#2	(dual-energy absorptiometry or dexa or dxa).mp.	31931
#3	Genome-wide association studies.mp. or exp genome-wide association study/	51201
#4	exp Genome, Human/ or exp Genome-wide Association Study/ or exp Genetic Predisposition to disease/or GWAS.mp. or exp Genomics/ or exp Polymorphism, Single Nucleotide	432219
#5	3 or 4	435626
#6	1 and 2 and 5	131
#7	limit 6 to "all adult (19 plus years)"	94

Table 2: Embase search strategy

Search	Query	Number of hits
#1	(Obesity or Adiposity or Overweight or fat mass or Obesity indices).mp.	734123
#2	(dual-energy absorptiometry or dexa or dxa).mp.	39757
#3	genome-wide association studies.mp. or exp genome-wide association study/ or exp Genetic Predisposition to disease/or GWAS.mp. or exp Genomics/ or exp Polymorphism, Single Nucleotide	67542
#4	1 and 2 and 3	25
#5	limit 4 to (adult <18 to 64 years> or aged <65+ years>)	25

2.3.4 Data Extraction

Data extraction was performed by B.T.B.A. in the presence of another reviewer (P.E) facilitating active discussion and evaluation of all the publications. A third reviewer was not needed as all conflicts were resolved after discussions. The following data were extracted if available: authors, publication year, country of origin, ethnicity, age, study design, description of the study population, sample size, gender, genotype variant, risk allele, risk genotype, mapped gene, reported trait(phenotype), statistical results including p-values, beta, confidence interval (CI) effect sizes, odd ratio, haplotype, impact size.

2.3.5 Risk of bias and methodological quality

Risk of bias was determined to the specific criteria adapted from the Quality of Genetic Association studies tool [30] with the following sections: selection bias (inclusion/exclusion criteria and population stratification), sample size calculations, genetic data (DNA sampling, genotyping method, quality control, blinding and Hardy-Weinberg equilibrium) and statistical analyses (measure of association and replication within the study), giving a maximum 11 points. The studies were classified according to their score whereby ≤6 points being low quality, 7-8 medium quality, and ≥ 9 high quality. The criteria were assessed independently.

2.4 Results

The results are presented according to the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines [31].

2.4.1 Literature Search Results and Determination of Eligibility

A total of 119 publications were obtained, 94 from Medline and 25 from Embase. Each publication was uploaded onto the Covidence platform, where the selection was performed. Four duplicates were removed (**Figure 1**). Of the 115 studies left, 81 were excluded because they did not meet all the selection criteria after title and abstract screening. 34 full texts were reviewed, and 20 studies were excluded due to one of the following reasons: Wrong setting, wrong outcomes, wrong comparator, wrong intervention, wrong study design, and wrong patient population. Through this process, 14 studies were eligible for data extraction and systematic analysis.

2.4.2 Characteristics of the Included Publications

The countries of origin were diverse, including Spain, the United Kingdom, China, the United States, the Netherlands, South Africa, and India. Therefore, ethnicities varied with Caucasian, east Asian (Chinese), Indo-Asian, as well as African origins (**Table 3**).

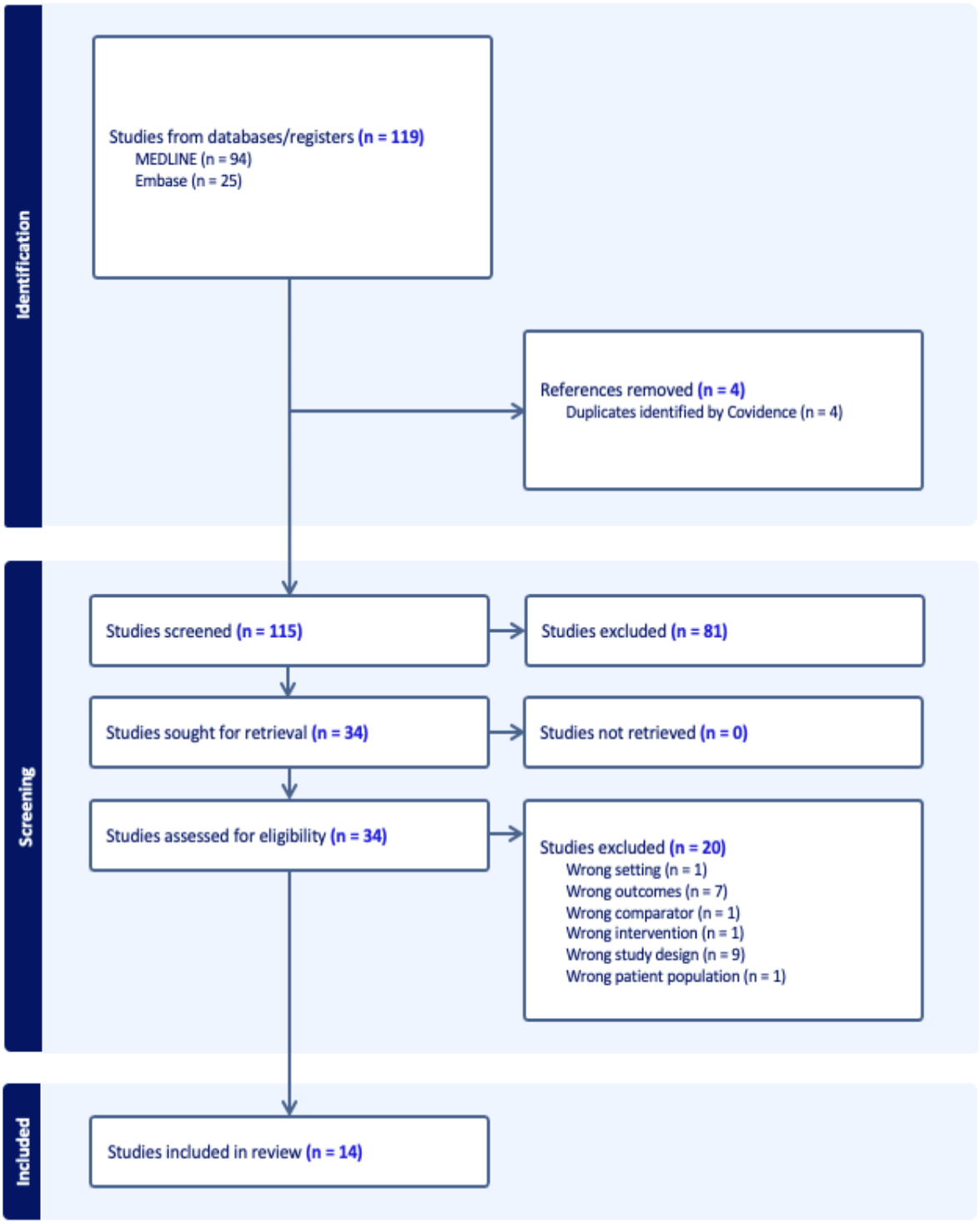
Diverse study designs were utilized. Nine studies are cross-sectional, two are case-control, two studies are longitudinal and one study is a discovery GWAS. The genetic analyses ranged from single variation association studies to GWAS.

All studies include both sexes, except for the study conducted by Arrizabalaga et al. [32]. Ages in the adult cohort ranged from 18 and 82, and sample sizes were between 67 and 9667.

Some publications did not describe details on the effect allele, beta value, or impact sizes, which therefore could not be included in further analysis. In addition, due to study design specifications, the statistical significance threshold differed greatly.

Five publications [35, 37, 39, 40, 42] marked by an Asterix (*) in **Table 3** did not present any statistically significant results.

Figure 1: Systematic Review Workflow Diagram



2.4.3 Critical Appraisal of the Publications Strength

Out of the 14 publications, 9 were assessed as medium quality, 3 as low quality, and 2 as high quality (**Table 5**). All the publications described their statistical methods and controls. All the publications stated the aim of the study and the inclusion or exclusion criteria. However, only 2 studies presented strong statistical designs ensuring associations' replicability. Furthermore, 8 studies had enough power based on sufficient sample size. Seven studies did not give any information on whether population stratification was done. Two studies [33, 38] failed to adequately describe the genotyping.

2.4.4 Genetic Loci described as associated with obesity phenotypes assessed by DXA

Genetic variations in 11 genes (FTO, MC4R, LINCO1122, CAP2, FADS1, BRINP2/SEC16B, LGR4, ALOX12, PRDM16, MSTN, and an unknown locus) and a locus not mapped to any gene were identified to be associated with obesity phenotypes assessed by DEXA (**Table 4**). The obesity phenotypes that were reported in the studies include fat mass, android fat mass, gynoid fat mass, trunk fat mass, arm fat mass, leg fat mass, total fat mass, percentage of total fat mass, and percentage fat mass.

In the 11 mapped genes identified, 17 SNPs (**Table 4**) show a statistically significant association with obesity-related phenotypes in these populations.

Table 3: Characteristics of included studies

Reference	Aim(s)	Country (Ethnicity)	Study design	Sample size	Gender	Age (Years)
Arrizabalaga 2014 [32]	FTO rs9939609 and MC4R rs17782313 polymorphisms: -Associations with Resting Energy Expenditure -Role in these key mechanisms of body weight and adiposity control.	Spain (Caucasian)	Case-control study	77	Female	19 to 49
Clifton 2017 [33]	Association of all 96 biallelic reported BMI-related variants with fat, lean, and bone mass across body regions	United Kingdom (92.3% White British)	Case-control study	9667	Male Female	29.4 to 63.7 30 to 64
Guo 2017 [34]	Examine the effect of FADS1 rs174547 on plasma Fatty acid composition, body composition, body fat indices, and plasma lipid profiles	China (Chinese)	Cross-sectional study	67	Female Male	18 to 82
*Lara-Castro 2005 [35]	Relationship between abdominal fat compartments and the Ala^{54}Thr FABP2 polymorphism	United States of America (African-American and Caucasian)	Longitudinal Study	223	Female	41 to 42 (±7)
Livingstone 2021 [36]	Understanding the genetics of the three-compartment model of body composition.	United Kingdom (White British)	Discovery GWAS	4 386	Male Females	40 to 69
*Pecioska 2010 [37]	To evaluate whether the recently identified Type 2 Diabetes risk alleles are associated with human measures of fatness as characterized by DEXA	Netherlands (Caucasian)	Cross-sectional study	3,000	Male Female	49.9 to 50.33 (±16)
Sahibdeen 2018 [38]	To assess whether European Body Composition-associated gene loci played a similar role in a South African black population	South Africa (South African)	Longitudinal Study	1926	Male Female	42±8.8 (Caregivers)

Table 3: Characteristics of included studies (continued)

Reference	Aim(s)	Country (Ethnicity)	Study design	Sample size	Gender	Age (Years)
*Sharma 2011 [39]	To determine the frequency distribution of LMNA 1908C>T SNP and its association with generalized obesity, abdominal obesity, and coexistent metabolic disorders	India (Asian Indians)	cross-sectional population-based	551	Male Female	40.2± 8.7 (Obese) 38.9 ± 8.7 (non-obese) 40.2± 8.5 (Insulin resistant)
*Sharma 2013 [40]	To determine the polymorphic status of 11-bHSD1 4478T>G and TNF-a-308G>A in Asian Indians in north India.	North India (Asian Indians)	Cross-sectional	498	Male Female	40.2± 8.7 (Obese) 38.9 ± 8.7 (non-obese) 40.2± 8.5 (Insulin resistant)
Xiao 2011 [41]	To determine whether polymorphisms in the ALOX12 gene were associated with variations in peak bone mineral density (BMD) and obesity phenotypes in young Chinese men	China (Chinese Han)	Cross-sectional	1215	Male Female	61.1 ± 7.1 (Father) 58.4 ± 6.3 (Mother) 30.4 ± 6.1 (son)
*Yu 2010 [42]	To investigate the effect of low-density lipoprotein receptor-related protein 5 (LRP5) gene polymorphisms on bone and obesity phenotypes	China (Chinese)	Cross-sectional	1244	Male Female	61.1 ± 7.0 (Father) 58.2 ± 6.3 (Mother) 30.6 ± 6.3 (son)
Yu 2020 [43]	To explore the relationship between genetic variations in the LGR4 gene and peak bone mineral density (peak BMD) and body composition phenotypes	China (Chinese)	Cross-sectional	1215	Male Female	61.2 ± 7.1 (Father) 58.5 ± 6.3 (Mother) 30.2 ± 6.2 (son)
Yue 2012 [44]	To investigate whether Myostatin played a role in the normal variation in peak BMD, lean mass (LM), and fat mass (FM)	China (Chinese Han)	Cross-sectional	1215	Male Female	61.1 ± 7.1 (Father) 58.4 ± 6.4 (Mother) 30.4 ± 6.1 (son
Yue 2013 [45]	To investigate the contribution of genetic variants of PRDM16 to obesity-related phenotype variations	China (Chinese Han)	Cross-sectional	3204	Male Female	20-40 (Young males) 50-80 (older men)

Table 4: Genetic Loci and the variations described to be associated with obesity phenotypes.

Mapped Gene	Genotype Variants	Reported trait (Phenotype)	Risk Genotype/Haplotype	Effect Allele	P value	Beta or Impact Size	Reference
FTO	rs9939609	Fat Mass	TA + AA		0.032		Arrizabalaga 2014 [32]
LINC01122	rs7592270	Android fat mass in both male and female		T	7.84 x 10-9	401.91	Livingstone 2021 [36]
		Trunk fat mass male and female		T	1.56 x 10-8	2011.4	
		VAT mass in male and female		T	4.49 x 10-8	251.04	
		VAT volume in male and female		T	4.49 x 10-8	266.09	
CAP2	rs113380185	Gynoid fat mass in male		T	1.58 x 10-8	959.03	
Unknown	13:72584929_AAC_A	Gynoid fat mass in male		A	1.89 x 10-8	759.52	
		Leg fat mass in male		A	1.59 x 10-8	1333.3	
	rs117686994	Arm fat mass in female		G	1.36 x 10-8	662.11	
	rs143707182	Leg fat mass in male		G	2.48 x 10-8	1782.6	
	rs138014219	Leg fat mass in male		T	2.96 x 10-8	2194.3	

Table 4: Genetic Loci and the variations described to be associated with Obesity Phenotypes (continued)

Mapped Gene	Genotype Variants	Reported trait (Phenotype)	Risk Genotype/ Haplotype	Effect Allele	P value	Beta or Impact Size	Reference
FADS1	rs174547	Gynoid fat mass percent in men (P^c Adjusted)	CC		0.040 (Dominant) 0.020 (Addictive)	-0.01	Guo 2017 [34]
		Limb fat mass percent in men (P^b Adjusted)	CC		0.033 (Dominant) 0.042 (Addictive)	-0.38	
		Android fat mass percent in women (P^c Adjusted)	CC		0.041 (Dominant)	-0.09	
		Android/gynoid fat ratio in women (P^a Adjusted)	CC		0.038 (Addictive)	-0.01	
		Total fat mass percent in women (P^a Adjusted)	CC		0.006 (Addictive) 0.003 (Recessive)	-0.32	
		Limb fat mass in women (P^a Adjusted)	CC		0.040 (Addictive) 0.037 (Recessive)	-0.10	
		Trunk fat mass in women (P^a Adjusted)	CC		Limb fat mass in women (P^a Adjusted)	-0.21	
BRINP2| SEC16B	rs6664268	Fat mass		C	$9.48 \times 10{-}7$	−1.80	Sahibdeen 2018 [38]
		Percentage Fat mass		C	$7.90 \times 10{-}7$	−1.35	
LGR4	rs11029986	Total fat mass (TFM) -within family associations			0.014		Yu 2020 [43]
		Percentage of Total fat mass (PFM) – within family associations			0.011		

Table 4: Genetic Loci and the variations described to be associated with Obesity Phenotypes (continued)

Mapped Gene	Genotype Variants	Reported trait (Phenotype)	Risk Genotype/Haplotype	Effect Allele	P value	Beta or Impact Size	Reference
MSTN	rs3791783	Trunk FM (For all associations)			9e-011	Not determined	Yue 2012 [44]
		Arms FM Tests of Population Stratification	AGG		0.0380		
		Total FM Tests of population stratification			0.0366		
		Trunk FM Tests of total association			1e-010		
		Trunk FM Tests of within-family association			1e-010		
		Trunk FM 1000 permutations of within-family association			1e-010		
		Trunk FM Tests of total association	AAA		1e-010		
		Trunk FM Tests of within-family association			8e-011		
		Trunk FM 1000 permutations of within-family association			8e-011		
		Trunk FM (Tests of total association within-family association and 1000 permutations of within-family association)	TGG		1e-010		

Table 4: Genetic Loci and the variations described to be associated with Obesity Phenotypes (continued)

Mapped Gene	Genotype Variants	Reported trait (Phenotype)	Risk Genotype/Haplotype	Effect Allele	P value	Beta or Impact Size	Reference
ALOX12	rs2073438	Total fat mass	AA		0.001	2.05 grams more than AG and 3.31grams more than GG	Xiao 2011 [41]
		Percentage fat mass	AA		0.015	0.41 grams less than AA and 0.19g less than GG	
	rs312462	Total fat mass	GG		0.003	2.67 grams less than GG and 0.72 grams more than AA	
		Percentage fat mass	GG		0.001	3.05 grams less than GG and 4.48 grams less than AA	
PRDM16	rs2236518	trunk Fat mass			0.023		Yue 2013 [45]
		Total Fat mass			0.004		

Table 5: Assessment of the strength of the Publications

Study	Aim of Study	Incl. / Excl.	Pop. Strat.	Power	Phenotype Definition	Genotyping	P value ≤ 5×10^{-8}	Statistical Methods and control	Hardy-Weinb.	Stren. Assoc.	Rep.	Sum (0–11)
Arrizabalaga 2014 [32]	1	1	0	0	1	1	1	1	1	0	0	7
Clifton 2017 [33]	1	1	0	0	1	1	0	1	0	0	0	5
Guo 2017 [34]	1	1	0	0	1	0	0	1	1	0	0	5
Lara-Castro 2005 [35]	1	1	0	0	1	1	0	1	1	0	0	6
Livingstone 2021 [36]	1	1	1	1	1	1	1	1	1	0	0	9
Pecioska 2010 [37]	1	1	0	1	1	1	0	1	1	0	0	7
Sahibdeen 2018 [38]	1	1	1	1	1	1	1	1	0	0	0	8
Sharma 2011 [39]	1	1	0	1	1	0	0	1	1	1	1	8
Sharma 2013 [40]	1	1	0	1	1	1	0	1	1	1	1	9
Xiao 2011 [41]	1	1	1	0	1	1	0	1	1	0	0	7
Yu 2010 [42]	1	1	1	0	1	1	0	1	1	0	0	7
Yu 2020 [43]	1	1	1	1	1	1	0	1	1	0	0	8
Yue 2012 [44]	1	1	1	1	1	1	0	1	1	0	0	8
Yue 2013 [45]	1	1	1	1	1	1	0	1	1	0	0	8

Study: Reference of study
Incl./ excl.: Inclusion and exclusion criteria mentioned
Pop. Strat.: Population stratification.
Power: Power of study according to sample size
Hardy-Weinb.: Hardy-Weinberg
Rep.: Replication
Stren. Assoc.: Strength of Association for Replication

2.5 Discussion

This systematic review identified 12 genetic loci associated with obesity phenotypes assessed by Dual X-Ray Absorptiometry (DEXA). This relatively small number identified reflects the focused nature of this review which targeted studies using the DXA technology to investigate middle-aged individuals.

Research involving the genetics of obesity has been growing steadily. The National Human Genome Research Institute-European Bioinformatics Institute NHGRI-EBI Catalog of Genome-wide Association Studies (GWAS) [46] database has documented 78 publications reporting 295 associations showcasing GWAS as the ideal model to identify genetic variants. However, this review highlights considerable methodological variability among the included studies, particularly in their designs and outcome measures. The majority of the included studies used the candidate gene approach, which indicates significant limitations to contribute to the knowledge in this field [47-49].

Body Mass Index (BMI) remains the most commonly used metric for obesity-related research since it is simple, easy and affordable. However, BMI is an unspecific anthropometric measurement. In contrast, the technically complex and costly DXA remains underutilized, demonstrated by a complete lack of records in the NHGRI-EBI Catalog of GWAS [46]. DXA offers precise measurements of anatomically distinct fat compartments, which is essential for a nuanced understanding of differential fat distribution and to associate these with health outcomes. In the future, wider use of DXA in genetics epidemiology will improve association studies and hopefully lead to precise diagnostics and interventions.

It has recently been shown that other imaging-based analyses utilizing computer tomography (CT) and Magnetic Resonance Imaging (MRI) allow us to determine the heritability of differential fat distribution [50-52]. However, evidence of BMI-independent measures of local adiposity remains limited and the inheritance of fat distribution in the general population and precise health implications requires further investigations.

This review supports the growing evidence that fat distribution is an important indicator when assessing the risk of obesity and its complications but highlights the methodological shortcomings in exiting publications.

2.6 Strengths and Limitations

This review demonstrates a substantial methodological heterogeneity in the studies published, specifically differences in study design and outcome measures. There was a lack of GWAS studies and many of the selected studies were using candidate gene designs. The candidate gene design examines the relationship between an already implicated gene and a trait or disease, introducing bias and due to the less stringent nature of the statistical analyses a high change of false positive error. GWAS enables unbiased discovery across the whole genome. The presence of only one GWAS therefore limits the ability to report novel genomic loci in the overall body of evidence. The lack of records in this specific field highlights the enormous research gap and should inspire future GWAS studies to clarify the validity of the reported genetic loci. It is crucial to interrogate larger cohorts, diverse populations, longitudinal outcomes, and the integration of other omics data.

The review acknowledges the methodological limitations of the observational nature of included studies, which always restricts the ability to establish causal relationships immediately. However, observational studies represent the most feasible approach to studying the obesity phenomenon and it is assumed that future studies will multiply and refine the body of scientific evidence. Observational studies are more feasible due to two main reasons: first, the ethical constraints of experimental designs that may involve manipulating genetic factors or deliberately causing obesity which raises ethical concerns; and second, the logistical constraints of conducting randomized controlled trials for complex traits like obesity because of practical limitation such as large sample sizes or the need for long follow-ups. With the emergence of large cohort studies, it is anticipated that the field of obesity research and intervention measures will evolve to reach a state where valid genetic associations can be utilised for precision health interventions.

Despite the challenges of study variability, this review provides valuable knowledge on genetic loci related to obesity. The strength of this systematic review lies in its use of a predefined protocol to reduce bias, its evidence-based approach, and its clear methodology ensuring transparency. In addition, Covidence is known as a valuable systematic review tool which streamlined the study selection and data extraction process. The narrow focus of the study on DXA technology with precise measurements of fat distribution leads to a more detailed understanding of obesity compared to studies relying solely on BMI, waist circumference, or bioelectrical impedance analysis.

2.7 Conclusion

A list of loci and genetic variants associated with obesity traits accessed by DXA has been gathered. This sets a frame of reference to be compared to future genome-wide associations for fat distribution assessed by DXA technology. It is predicted, that integrating genetic insights determined through advanced imaging methodologies will help to refine insights into the interplay of genetics, lifestyle, and environmental factors in addressing a growing public health crisis.

2.8 Conflict of interest

The authors report no conflict of interest.

Chapter 3: Genome-wide Association study of obesity phenotypes from the Manitoba personalized lifestyle research

3.1 Abstract

Background: Obesity rates are growing in most nations and are proportionally higher in Manitoba compared to other provinces in Canada. In 2018, 30.8% of Manitoban residents aged 18 and above were obese, predisposing them to associated adverse health consequences. The Manitoba Personalized Lifestyle Research (TMPLR) is a cross-sectional study in adults, aged 30 to 46 years where obesity phenotypes were measured using very precise Dual Energy X-ray Absorptiometry (DXA). Obesity phenotypes that were included are total fat mass, gynoid fat mass, android fat mass, legs fat mass, arms fat mass, and trunk fat mass.

Aim: To perform a genome-wide association study (GWAS) within the TMPLR cohort to identify the genes that are associated with obesity phenotypes in the Manitoban adult population.

Methods: The Illumina Infinium Global Diversity Array with Enhanced PGx Genome Build 37 was used to conduct a GWAS of obesity phenotypes assessed by DXA on 391 individuals previously enrolled in TMPLR. Bioinformatics analysis was performed in the Biodata catalyst platform and Seven Bridges R studio version 4.1 and Bioconductor version 3.14.

Results: No significant genome-wide association with obesity was established in the TMPLR cohort. However, suggestive associations were identified for several genes across the different fat mass phenotypes: 11 genes for total fat mass, 8 genes for android fat mass, 3 genes for gynoid fat mass, 6 genes for arms fat mass, 2 genes for leg fat mass, 4 genes for trunk fat mass. Recurrent genes across the various phenotypes are RHNO1, RAPGEF4, RP11 family, and PRIMA1. PARK2, RAPGEF4, PRIMA1, and RP11 family have been previously linked to obesity while SIPA1L1, ATP8A1, KIF26A, STAU2, FBLN7, and RHNO1 have been previously linked to cancer highlighting the possible link between metabolic processes and broader disease pathways. Additionally, TPST2 has been linked to cardiovascular diseases.

Conclusion: Although many suggestive associations were found with obesity, none met the stringent genome-wide significance threshold of $p < 5 \times 10^{-8}$ likely due to the small sample size. Nevertheless, the findings provide a baseline and valuable insights on which further studies with larger sample sizes can extend and validate.

3.2 Introduction

Incidences of obesity are growing worldwide leading to a significant increase in associated preventable morbidity and mortality [53]. Canada experiences the same phenomenon where its province Manitoba lies about 4% above the national average [4]. The burden of obesity lies in its three primary consequences: Type 2 diabetes, heart diseases, and certain types of cancers. Although diet and physical activity may account for approximately 30-60% of obesity risk, evidence suggests that genetics account for 40-70% [25]. Genome-wide Association Studies (GWAS) are the most powerful tool to associate genetic variants with diseases such as obesity. To date, no GWAS study interrogated genome wide associations with obesity in middle aged Manitobans.

In a report [54] on adult obesity in Manitoba published in 2011, the Body Mass Index (BMI) was utilised as the measure. Although commonly used, the BMI does not distinguish between fat and lean mass or the precise location and sizes of the fat deposits. Imaging technology such as Dual Energy X-ray Absorptiometry (DXA) provides data on the distribution of fat in the body, enabling to determine total fat mass, android fat mass, gynoid fat mass, arms fat mass, legs fat mass, trunk fat mass and visceral fat mass. Karastergiou et al. [55] emphasized the role of android and gynoid fat mass in disease risk reinforcing knowledge that fat distribution plays a role in disease occurrence. The report [54] considered a total of 23 variables and their relation to obesity levels, however, they did not elaborate on the genetics of the participants due to the unavailability of these data. The increasing prevalence of obesity in Manitoba and the lack of research using precision tools such as DXA imaging defines an understudied research area.

To address these research gaps a GWAS in the cross-sectional cohort of The Manitoba Personalized Lifestyle Research (TMPLR) project was conducted [19]. A cohort of 391 participants aged 30 to 46 was available, for which DXA as well as genotyping data existed. This enabled the genome wide interrogation of different obesity phenotypes in the Manitoban adult population.

3.3 Materials and Methods

3.3.1 TMPLR

The cross sectional TMPLR [19] design was published in 2019 and a cohort of 391 participants was available where DXA as well as genotyping data exist. Further details including methods, study population, and measurement can be accessed from its original publication [19].

3.3.2 DNA isolation

DNA was isolated from buffy coats of 426 individuals using the QIAGEN [56] (Protocol 1) for DNA extractions, and its quality was assessed via UV-spectrometry and DNA-Gel images (**Appendix A)**.

3.3.3 Dual Energy X-ray Absorptiometry (DXA)

The obesity phenotypes total fat mass, android fat mass, gynoid fat mass, arms fat mass, legs fat mass, and trunk fat mass were determined using a GE Lunar Prodigy [57] DXA machine. Data were curated by removing the duplicates and formatted to be used on the biodata catalyst platform.

3.3.4 Genotyping

The DNA was genotyped at the Genome Quebec [58] core facility using an Illumina Infinium Global Diversity Array with Enhanced PGx Genome Build 37 interrogating about 2 million variations. Seven categories of output files were created: metabolizer status report, CNV variant call file, SNV variant call file, genotyping call file, bed graph, genotype summary file, and copy number batch file.

3.3.5 Bioinformatics within the Biodata Catalyst

The Biodata Catalyst [59] platform was used for the data analysis based on the GENESIS tutorial 2023 [60]. The documentations for all application used in Biodata Catalyst [12] are detailed in **Appendix B**.

Genotyped data were pruned to a subset not in linkage disequilibrium (LD) with an LD threshold of 0.32. The kindship coefficients were estimated and were used as a measure of ancestry divergence. Principle component analysis (PCA) was performed to correct for population stratification. The null model workflow was used to fit the regression or mixed effect model under the null hypothesis of no genotype effect. The results were regressed and used in association tests. A Minor allele count **(**MAC) threshold was used according to the sample size check [61,62]**.** The threshold for genome-wide statistical significance was set to $p < 5 \times 10^{-8}$. In addition, suggestive statistical significance was explored with a significance level of $p < 1 \times 10^{-5}$ which is widely accepted [63,64] in GWAS, particularly in relatively small sample size studies like this. R studio [65] version 4.1 and Bioconductor version 3.14 in the Seven Bridges Bioinformatics platform of the Biodata Catalyst was utilised [59]. Some of the

R codes used were adapted from GENESIS [60] R scripts and are provided in **Appendix C**. Furthermore, R codes for the annotation using the VEP application are provided in **Appendix D**. The participants were mainly Caucasians.

3.4 Results

The total number of participants in the GWAS was 391 with 168 women and 223 men. The mean heights, weights, and ages are shown in **Table 6.** The minimum age of participants was 30 and the maximum age was 46 which is shown in **Table E1** in **Appendix E**. **Table E1** also shows the minimum and maximum height and weight. **Appendix E** shows the distribution of all the phenotypes namely: arms fat mass, legs fat mass, trunk fat mass, android fat mass, gynoid fat mass, and total fat mass.

Table 6: Basic descriptive parameters for TMPLR participants in the GWAS

Gender	Count	Height Mean (cm)	Height SD	Weight Mean (Kg)	Weight SD	Age Mean (Years)	Age SD
Female	168	165.2	6.8	75.7	19.2	38.7	4.7
Male	223	178.5	7.5	88.0	15.5	37.4	4.8

SD: Standard deviation, **Kg:** Kilograms, **cm:** centimeter

No variation was identified to be associated with any of the interrogated phenotypes when applying strict GWAS criteria with a significance level of $p < 5 \times 10^{-8}$ (**Figures 2-7**). Therefore, variations above the p-threshold of $p < 1 \times 10^{-5}$, which suggests a potential genetic association in a GWAS, were also determined and are described in the following paragraphs (**Figure 2a-7a**).

Suggestive genetic associations with total fat mass were observed for variations in the RHNO1, RNU6-939P, RAPGEF4, SPAG16, PARK2, PRIMA1, CTNND2, RP11-847H18.2, SIPA1L1, ATP8A1, and an unmapped locus (**Table 9, Figure 2a**). Suggestive genetic associations with gynoid fat mass were observed for variations in the RP11-103J8.1 and the TPST2 locus (**Table 9, Figure 4a**). Suggestive genetic associations with android fat mass were observed for

variations in the RHNO1, the SPAG16, RP11-406O16.1, KIF26A, PRIMA1, AC073257.2, and an unknown locus (**Table 9, Figure 3a**). Suggestive genetic associations with leg fat mass were observed for variations in the RAPGEF4 and the FBLN7 locus (**Table 9, Figure 6a**). Suggestive genetic associations with the arms fat mass were observed for variations in the RP11-472K22.1, STAU2, HRASLS5, AP002833.1, AC011288.2, and an unknown locus (**Table 9, Figure 5a**). Suggestive genetic associations with the trunk fat mass were observed for variations in the RHNO1, SPAG16, USP13, and an unknown locus (**Table 9, Figure 7a**). The Manhattan plots in **Figures 2** to **7 and Figures 2a** to **7a** help visualize the key associations. All the SNPs that are in Table 9 are represented by the figures. For instance, the x-axis shows the position of these variants across the chromosomes while the y-axis shows the significance of the association between each variant and the trait, represented by -log10 (p-value). **Figures 2** to **7** have a dotted line which represents the genome-wide significance threshold of $p < 5 \times 10^{-8}$, while **Figures 2a** to **7a** represent the threshold of suggestive statistical significance of $p < 1 \times 10^{-5}$.

Table 7 shows all the suggestive $p < 1 \times 10^{-5}$ obtained for each of the obesity phenotypes namely total, android, gynoid, arms, legs, and trunk fat mass. For total fat mass, the SPAG16-A allele is associated with an increase of 2140.78g in weight. The rest of the genes are associated with a decrease in weight for total fat mass. For gynoid fat mass, the G allele of RP11-103J8.1 (rs4813943) is associated with an increase of 314.68g. For android fat mass, the A allele of gene SPAG16 is associated with an increase of 310.99g, and the T allele of the SNP rs777625 is associated with an increase of 782.45g. No allele from any genes was associated with an increase in leg fat mass. Five genes that caused an increase in weight in arms fat mass are as follows: G allele of RP11-472K22.1 with an increase of 798.17g, A allele position 17722077 with an increase of 762.99g, A allele of HRASLS5 gene with an increase of 1020.11g, C allele for AP002833.1 gene with an increase of 389.45g and C allele for AC011288.2 gene with an increase of 166.17g. For trunk fat mass, the A allele of the SPAG16 gene with an increase of 1542.89g and the C allele of the USP13 gene with an increase of 1154.90g.

Figure 1F in **Appendix F** represents the kinship estimates. The dots on the kinship coefficient have dots that represent a pairwise comparison between two individuals. The density of pairwise comparison in the region of the plot is represented by different shades of blue with darker shades indicating more pairs of individuals with that combination [66,67]. The kindship

estimate is on the y-axis which represents the genetic similarity between two individuals whereas the x-axis represents the proportion of the genome where two individuals do not share any alleles identical by descent, labeled as k0, also known as the identical by descent value [66,67].

There are three sets of clusters in **Figure F1**. The first set completely at the top is relatively small and located above the kinship estimate of 0.4 which indicates the presence of some closely related individuals who have high relatedness. The second medium set of clusters is in the middle and is relatively larger than the first top clusters and is located between 0.2 to 0.3 kinship estimate on the y-axis demonstrating some distantly related individuals with some relatedness and showing similar ancestral backgrounds and ethnicities. The larger cluster at the bottom of the figure is close to the 0 kinship estimate but near 0.75 on the x-axis value indicating that a large portion of the participants were distantly related or unrelated. The results suggest that the majority of the participants had different ancestral backgrounds, especially in terms of ethnicities or population with a lack of significant overlapping in genetic variations.

Table 7: Suggestive genetic associations for obesity phenotypes in the TMPLR cohort

Phenotype	CHR	REF	ALT	Gene Symbols	SNP ID	Position	Score	Score. SE	Score. Stat	Score. pval	Est	Est.SE
Total Fat Mass	12	A	G	RHNO1	rs142328102	2994612	-0.0027	0.0005	-5.5627	2.66E-08	-11509.18	2069.00
	2	G	A	RNU6-939P	rs4569456	37554788	-0.0056	0.0011	-4.9366	7.95E-07	-4347.60	880.68
	2	C	T	RAPGEF4	rs7604579	173785000	-0.0052	0.0011	-4.8804	1.06E-06	-4572.20	936.85
	2	A	C	SPAG16	rs1510552	214917402	0.0111	0.0023	4.8700	1.12E-06	2140.78	439.58
	7	C	T	Unknown	rs117042944	121147147	-0.0042	0.0009	-4.6413	3.46E-06	-5081.59	1094.86
	6	G	A	PARK2	rs35621098	162162658	-0.0040	0.0009	-4.6304	3.65E-06	-5326.29	1150.28
	14	C	T	PRIMA1	rs78070585	94229429	-0.0029	0.0006	-4.5976	4.27E-06	-7411.02	1611.94
	5	G	A	CTNND2	rs62339961	11886744	-0.0059	0.0013	-4.5316	5.85E-06	-3455.13	762.45
	12	A	G	RP11-847H18.2	rs190651674	34197919	-0.0036	0.0008	-4.5184	6.23E-06	-5745.06	1271.49
	14	A	T	SIPA1L1	rs117440071	72122311	-0.0029	0.0007	-4.4417	8.92E-06	-6764.13	1522.86
	4	C	T	ATP8A1	rs41265689	42551362	-0.0049	0.0011	-4.4216	9.80E-06	-3994.97	903.52
Gynoid Fat Mass	20	G	A	RP11-103J8.1	rs4813943	10756166	0.0689	0.0148	4.6563	3.22E-06	314.68	67.58
	22	A	G	TPST2	rs13056218	26917626	-0.0552	0.0124	-4.4565	8.33E-06	-359.98	80.78
	20	A	G	RP11-103J8.1	rs17538569	10759245	0.0560	0.0127	4.4212	9.82E-06	348.89	78.91
Android Fat Mass	12	A	G	RHNO1	rs142328102	2994612	-0.0166	0.0032	-5.1183	3.08E-07	-1577.63	308.23
	2	A	C	SPAG16	rs1510552	214917402	0.0723	0.0153	4.7433	2.10E-06	310.99	65.56
	6	A	T	RP11-406O16.1	rs9454444	68938740	-0.0193	0.0042	-4.6435	3.43E-06	-1117.17	240.59

Table 7: Suggestive statistically significant SNPs in TMPLR data analysis (continued)

Phenotype	CHR	REF	ALT	Gene Symbols	SNP ID	Position	Score	Score. SE	Score. Stat	Score. pval	Est	Est.SE
Android Fat Mass	6	A	G	RP11-406O16.1	rs6900019	68938286	-0.0192	0.0042	-4.5942	4.34E-06	-1098.31	239.06
	6	T	C	Unknown	rs777625	68898262	0.0264	0.0058	4.5441	5.52E-06	782.45	172.19
	14	G	T	KIF26A	rs117573425	104642489	-0.0328	0.0073	-4.5108	6.46E-06	-619.95	137.44
	14	C	T	PRIMA1	rs78070585	94229429	-0.0187	0.0042	-4.4969	6.90E-06	-1081.10	240.41
	2	C	A	AC073257.2	rs73952626	121295944	-0.0141	0.0032	-4.4617	8.13E-06	-1409.27	315.86
Legs Fat Mass	2	C	T	RAPGEF4	rs7604579	173785000	-0.0108	0.0023	-4.6113	4.00E-06	-1972.29	427.70
	2	C	T	FBLN7	rs7349410	112913967	-0.0283	0.0063	-4.5120	6.42E-06	-720.56	159.70
Arms Fat Mass	4	G	A	RP11-472K22.1	rs76841669	29133262	0.0336	0.0065	5.1801	2.22E-07	798.17	154.08
	16	A	G	Unknown	Unknown	17722077	0.0296	0.0062	4.7505	2.03E-06	762.99	160.61
	8	C	A	STAU2	rs61414848	74548187	-0.0496	0.0105	-4.7000	2.60E-06	-445.66	94.82
	11	A	C	HRASLS5	rs115216338	63238576	0.0199	0.0044	4.5006	6.78E-06	1020.11	226.66
	11	C	T	AP002833.1	rs79366772	126869125	0.0512	0.0115	4.4647	8.02E-06	389.45	87.23
	7	C	T	AC011288.2	rs12154889	13333829	0.1182	0.0267	4.4322	9.33E-06	166.17	37.49
Trunk Fat Mass	12	A	G	RHNO1	rs142328102	2994612	-0.0037	0.0007	-5.4157	6.10E-08	-7981.1	1473.69
	2	A	C	SPAG16	rs1510552	214917402	0.0157	0.0032	4.9209	8.62E-07	1542.89	313.54
	7	C	T	Unknown	rs117042944	121147147	-0.0059	0.0013	-4.5471	5.44E-06	-3533.16	777.01
	3	C	T	USP13	rs12633396	179389656	0.0174	0.0039	4.4807	7.44E-06	1154.90	257.75

Legend for Tables 7 and 8

CHR: Chromosome (CHR) represents the specific chromosome where the variant or SNP is located.

REF: Reference allele (REF) is the nucleotide base (A, T, C, G) found at the genomic location in the reference genome and is typically the most common in the population.

ALT: Alternate allele (ALT) is the nucleotide base that differs from the reference allele at the specific genomic location and would represent the variant form (mutation or polymorphism) found in individuals.

Gene symbols: The gene symbols are the abbreviated names of the genes that are associated with the specific SNP or variant.

SNP ID: SNP ID is the unique identifier from the dbSNP database which helps in identifying the SNPs or variants.

Position: The position is the exact base pair location on the chromosome in the genome build GRCh37 that we used.

Score: The score represents the value indicating the strength of the SNP with obesity. It is obtained from linear regression used to analyze data.

Score.SE: The standard error of the score (Score.SE) represents the uncertainty of the score. Lower Score.SE would mean that the score estimate is very precise.

Score.Stat: The statistical value (Score.Stat) is the summary statistic that indicates how powerful the association is. The larger the number (negative or positive) would mean a stronger association.

Score.pval: The P-value (Score.pval) is the probability value showing statistical significance. All values in the table show suggestive statistical significance of $\mathbf{p < 1 \times 10^{-5}}$**.** The stronger the evidence then the smaller the p value would be.

Est: Effect size estimate (Est) measure the extent of the effect of the SNP. A positive estimate would mean an increase in weight in grams and a negative estimate a decrease in weight in grams.

Est.SE: The Standard error for the effect size estimate reflects the precision of the effect size estimate. A smaller value would mean a greater precision of the effect size estimate. On the other hand, a large standard error indicates greater variability.

Genotyped data was pruned with an LD threshold of 0.32. The kindship coefficients were estimated and was used as a measure of ancestry divergence. PCA was performed in order to correct for population stratification. The null model workflow was used to fit the regression or mixed effect model. The results were regressed and used in association tests.

Figure 2: Genome-wide associations with Total Fat Mass

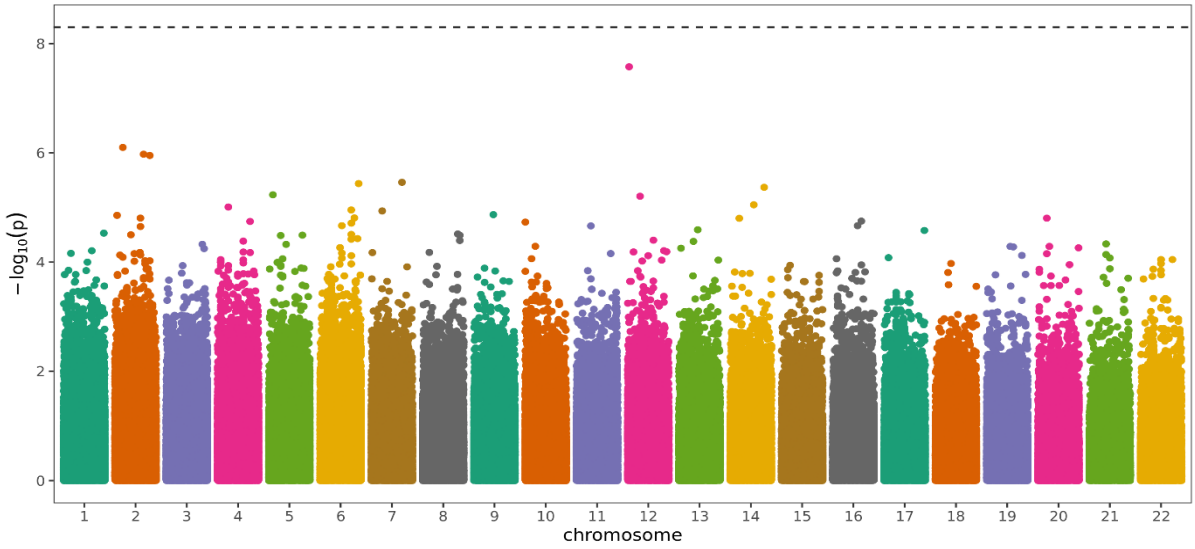


Figure 2 is a Manhattan plot for genome-wide association for total fat mass with a genome-wide significance threshold of $p < 5 \times 10^{-8}$. The graph demonstrates that there are no SNPs that reached genome-wide statistical significance for total fat mass.

Figure 2a: Suggestive genetic associations with Total fat mass

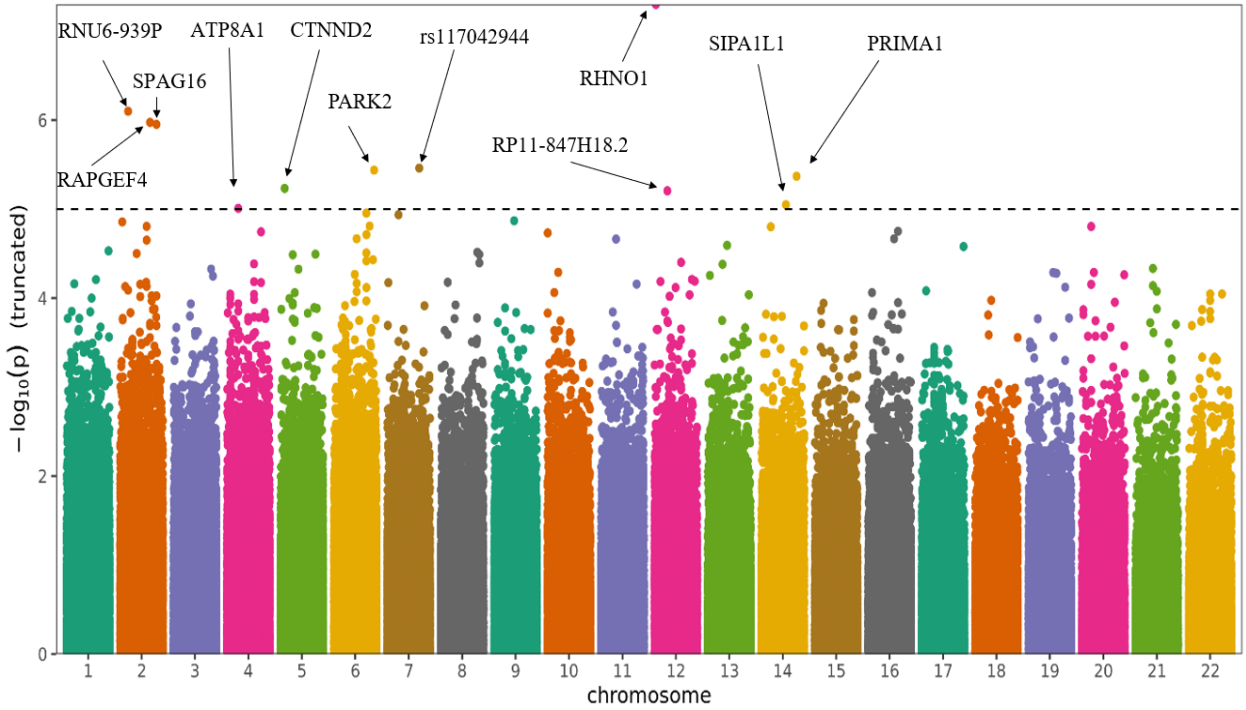


Figure 2a shows a Manhattan plot for suggestive genome-wide association for obesity phenotype total fat mass with a threshold of $p < 1 \times 10^{-5}$. The A allele of the SPAG16 gene is associated with an increase of 2140.78g in weight. The rest of the genes are associated with a decrease in weight.

Figure 3: Genome-wide associations with Android Fat Mass

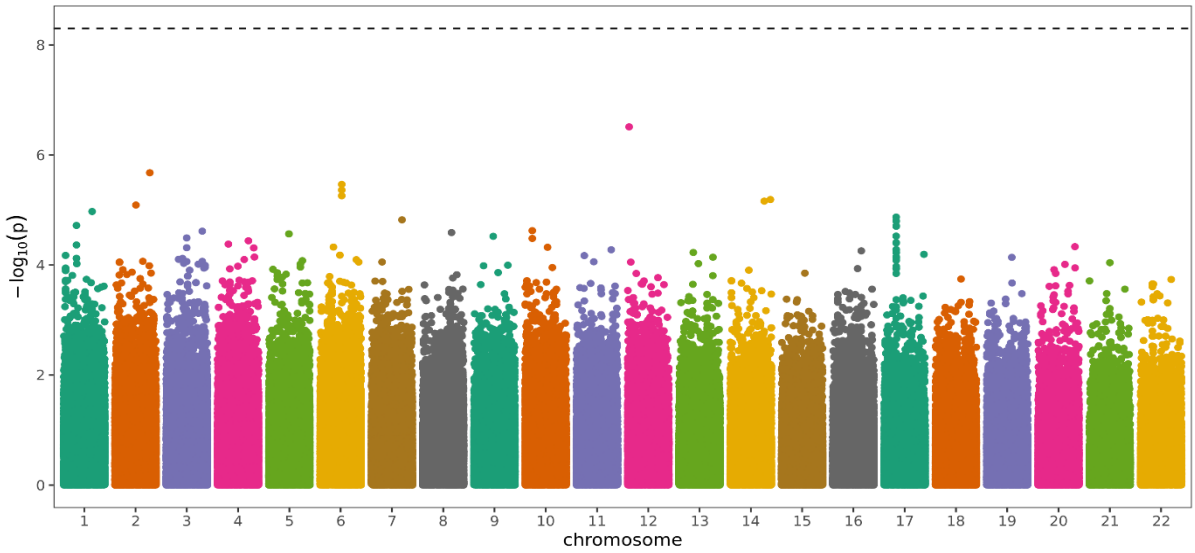


Figure 3 is a Manhattan plot for the genome-wide association for android fat mass with a genome-wide significance threshold of $p < 5 \times 10^{-8}$. The graph demonstrates that there are no SNPs that reached genome-wide statistical significance for android fat mass.

Figure 3a: Suggestive genetic associations with Android fat mass

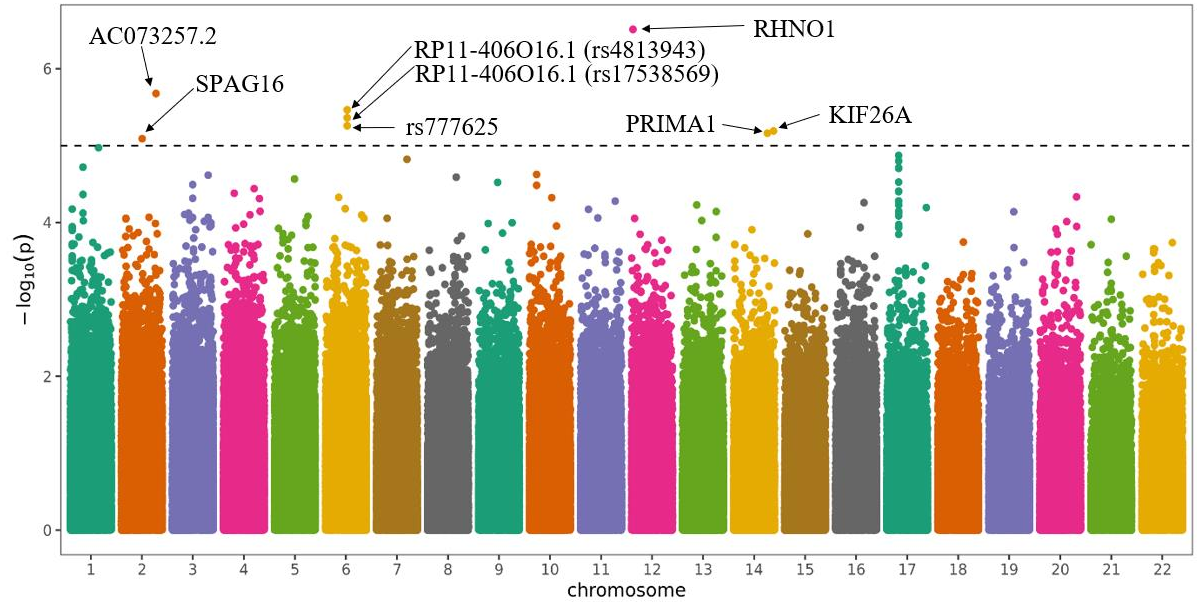


Figure 3a shows a Manhattan plot for suggestive genome-wide association for obesity phenotype android fat mass with a threshold of $p < 1 \times 10^{-5}$. The A allele of gene SPAG16 is associated with an increase of 310.99g and the T allele of the SNP rs777625 is associated with an increase of 782.45g.

Figure 4: Genome-wide associations with Gynoid Fat Mass

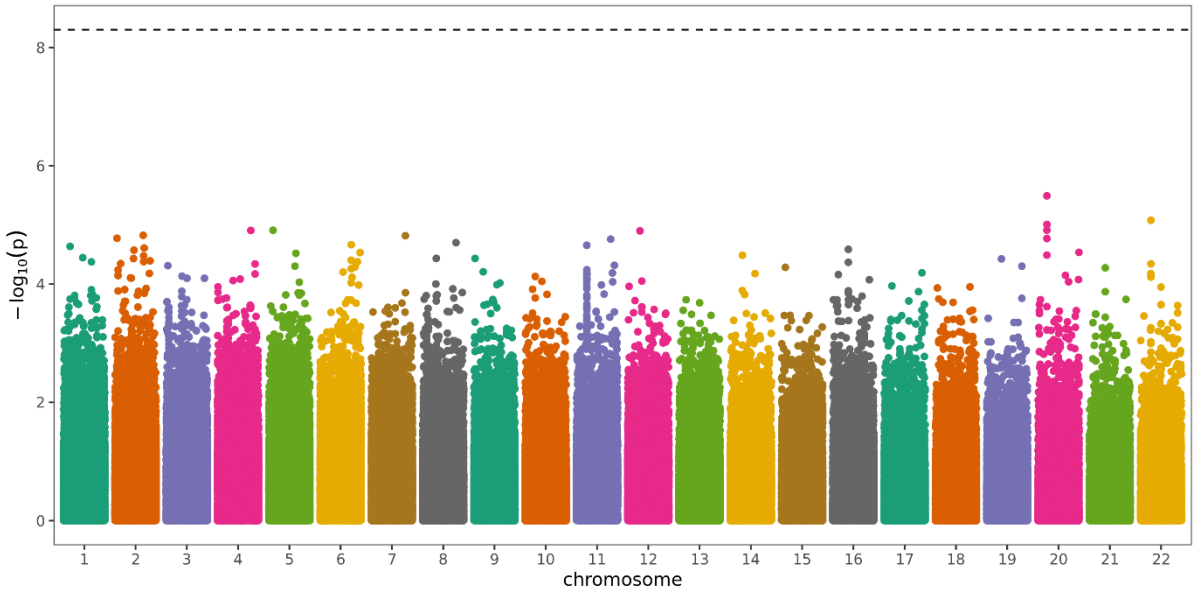


Figure 4 is a Manhattan plot for genome-wide association for gynoid fat mass with a genome-wide significance threshold of $p < 5 \times 10^{-8}$. The graph demonstrates that there are no SNPs that reached genome-wide statistical significance for gynoid fat mass.

Figure 4a: Suggestive genetic associations with Gynoid fat mass

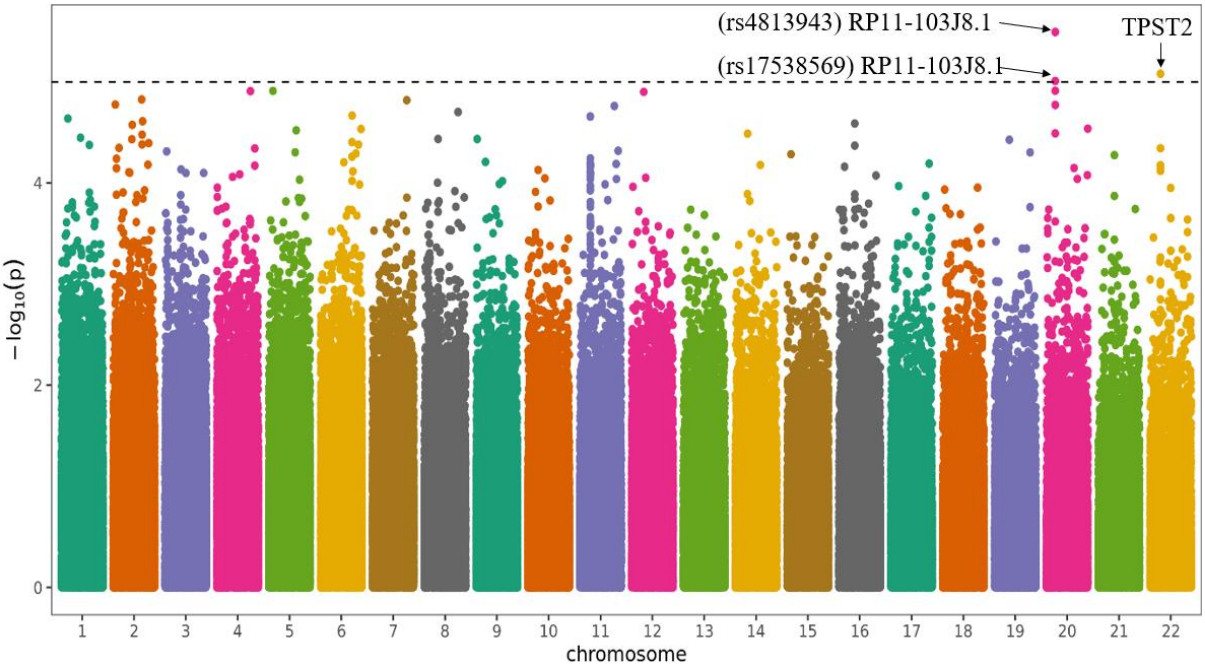


Figure 4a shows a Manhattan plot for suggestive genome-wide association for obesity phenotype gynoid fat mass with a threshold of $p < 1 \times 10^{-5}$. The G allele of RP11-103J8.1 (rs4813943) is associated with an increase of 314.68g.

Figure 5: Genome-wide associations with Arms Fat Mass

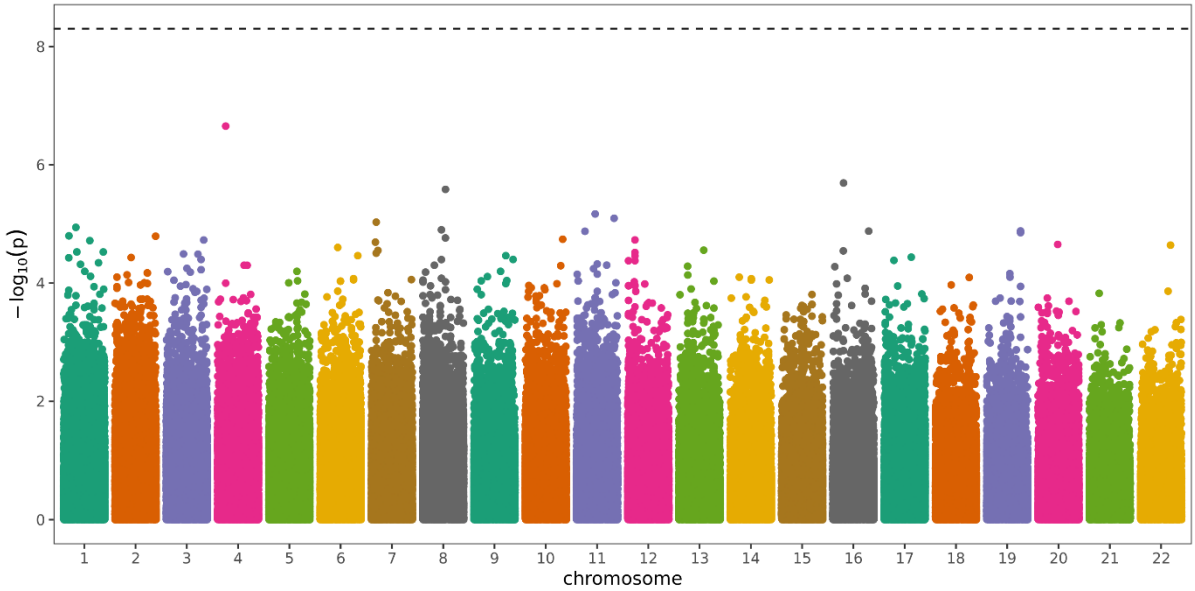


Figure 5 is a Manhattan plot for genome-wide association for arms fat mass with a genome-wide significance threshold of $p < 5 \times 10^{-8}$. The graph demonstrates that there are no SNPs that reached genome-wide statistical significance for arms fat mass.

Figure 5a: Suggestive genetic associations with Arms fat mass

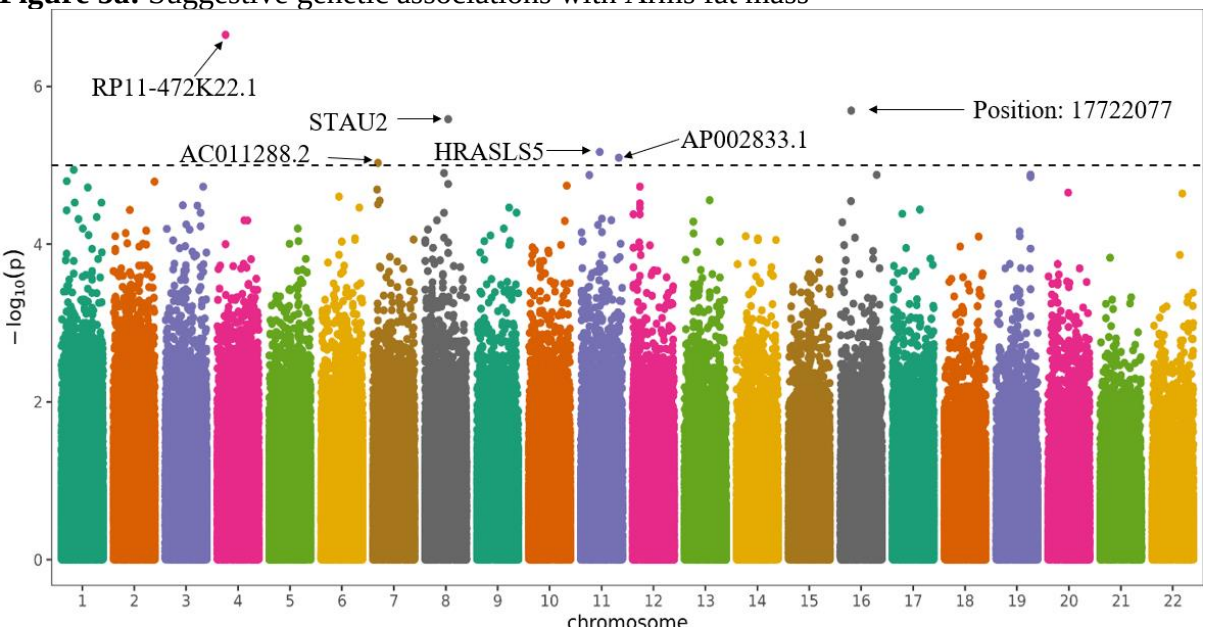


Figure 5a shows a Manhattan plot for suggestive genome-wide association for obesity phenotype arms fat mass with a threshold of $p < 1 \times 10^{-5}$. The G allele of RP11-472K22.1 with increase of 798.17g, the A allele position 17722077 with an increase of 762.99g, the A allele of HRASLS5 gene with an increase of 1020.11g, the C allele for AP002833.1 gene with an increase of 389.45g and the C allele for AC011288.2 gene with an increase of 166.17g.

Figure 6: Genome-wide associations with Legs Fat Mass

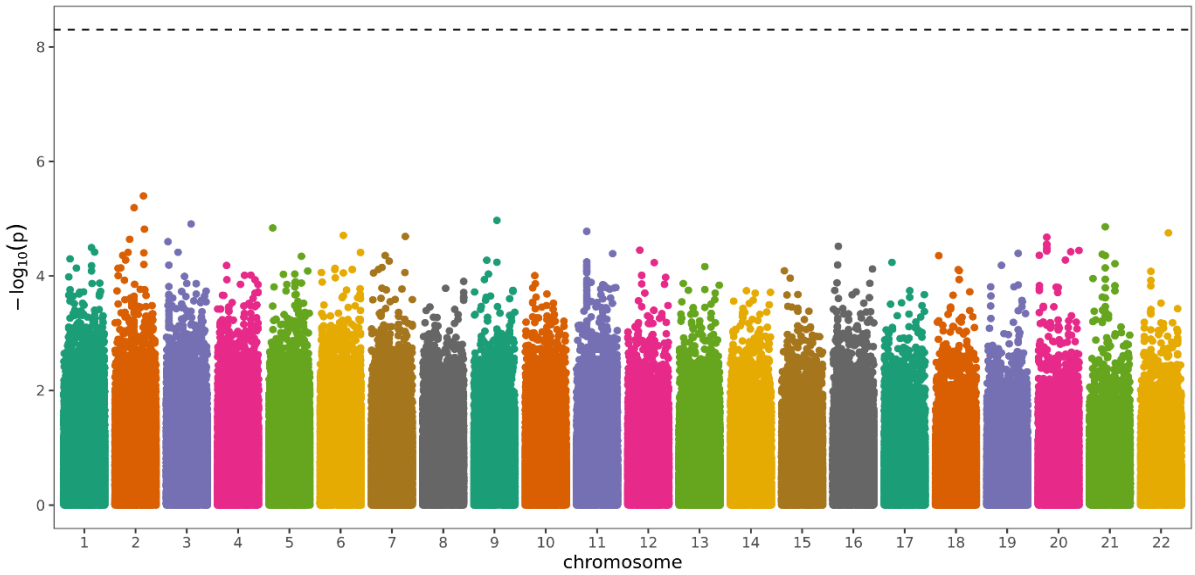


Figure 6 is a Manhattan plot for the genome-wide association for leg fat mass with a genome-wide significance threshold of $p < 5 \times 10^{-8}$. The graph demonstrates that there are no SNPs that reached genome-wide statistical significance for leg fat mass.

Figure 6a: Suggestive genetic associations with Legs fat mass

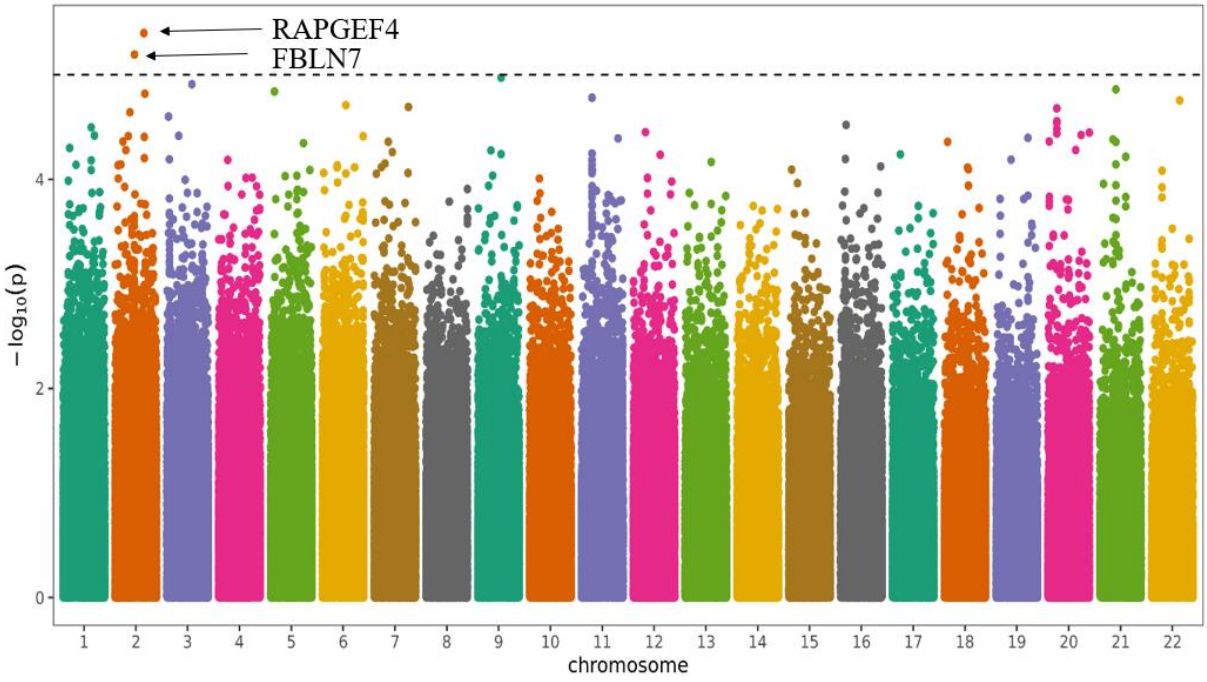


Figure 6a shows a Manhattan plot for suggestive genome-wide association for obesity phenotype legs fat mass with a threshold of $p < 1 \times 10^{-5}$. No allele of any of those genes was associated with an increase in weight.

Figure 7: Genome-wide associations with Trunk Fat Mass

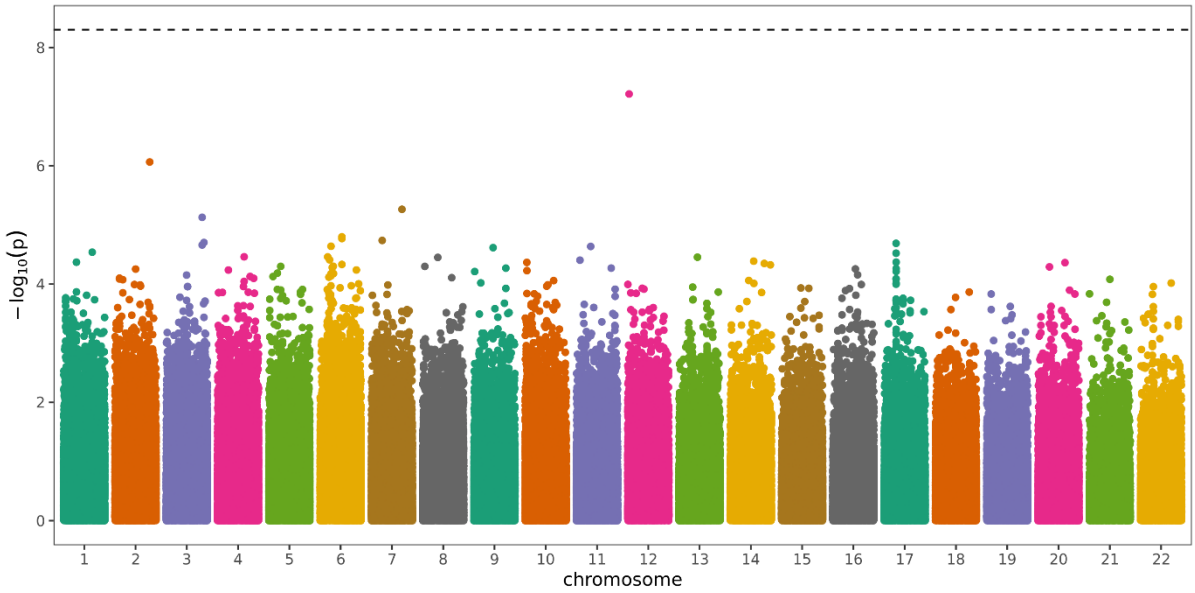


Figure 7 is a Manhattan plot for the genome-wide association for trunk fat mass with a genome-wide significance threshold of $p < 5 \times 10^{-8}$. The graph demonstrates that there are no SNPs that reached genome-wide statistical significance for trunk fat mass.

Figure 7a: Suggestive genetic associations with Trunk fat mass

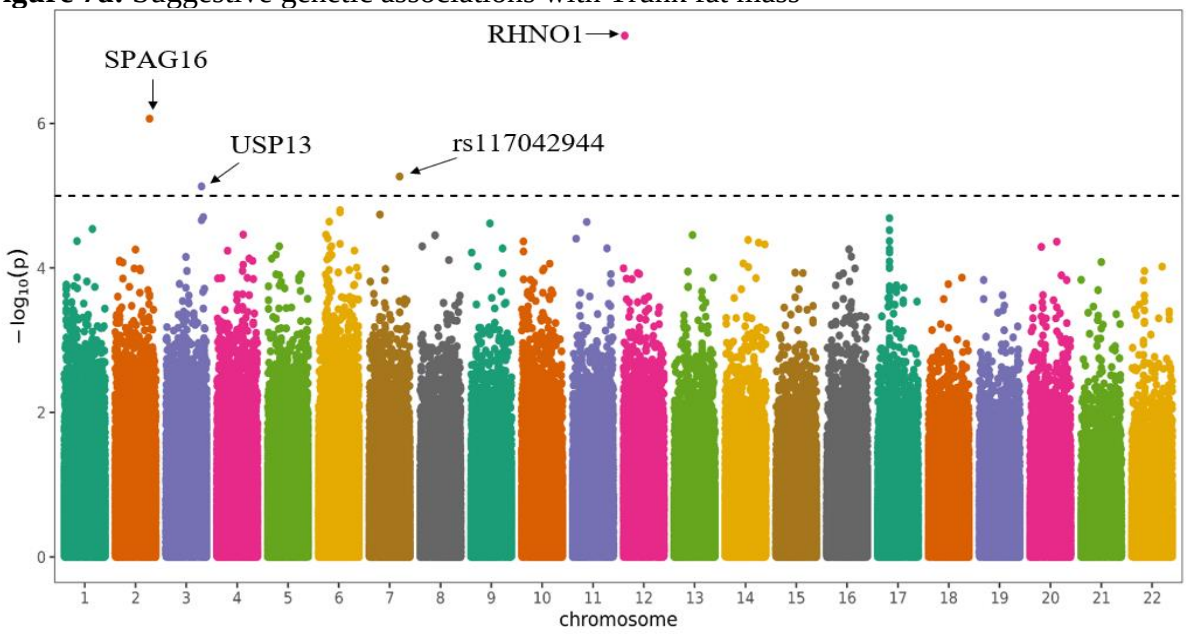


Figure 7a shows a Manhattan plot for suggestive genome-wide association for obesity phenotype trunk fat mass with a threshold of $p < 1 \times 10^{-5}$. The A allele of the SPAG16 gene with an increase of 1542.89g and the C allele of the USP13 gene with an increase of 1154.90g.

3.5 Discussion

No significant genome-wide association was discovered in this GWAS on differential fat distribution. Due to a lack of sample size in the cross-sectional cohort only the most penetrant variants would have been discoverable. The exploration of suggestive associations, just under the threshold of genome-wide significance, identified 23 genetic loci.

Unlike most studies that use the BMI as measure of obesity, our study specifically focused on total body fat mass and its distribution. Currently, 295 genetic associations to general obesity are listed in the NHGRI-EBI Catalog of GWAS [46]. In contrast, no genetic associations using DXA technology to differentiate the fat distribution are listed, signifying the underutilization of this technology and at the same time highlighting a substantial research gap. Evidence on the heritability of differential fat distribution has emerged, but it was created using computer tomography (CT) and MRI imaging [68-70]. However, evidence on the inheritance of fat distribution in the general population and its associations with health implications is extremely limited. The increased use of DXA and other imaging techniques in genetics epidemiology is expected to improve future association studies to inform on precise diagnostics and avenues of interventions.

The presented analysis could suggest an implication of 23 genes, some of which overlap across the different phenotypes namely RHNO1, RAPGEF4, PRIMA1 and RP11 family indicating possible common biological mechanism. PARK2, RAPGEF4, PRIMA1 and RP11 family have been previously associated with obesity in other studies while SIPA1L1, ATP8A1, KIF26A, STAU2, FBLN7 and RHNO1 have previously been linked to cancer highlighting the possible link between metabolic processes and broader disease pathways. TPST2 has been linked to cardiovascular diseases. Lastly, CTNND2, SPAG16, HRASLS5, AP002833.1, AC011288.2, RNU6-939P, rs777625, rs117042944, Position 17722077 showed relation to other diseases or have not been associated with any disease in previous publications.

The Rap Guanine Nucleotide Exchange Factor 4 (RAPGEF4) gene, also known as Exchange Protein Directly Activated by cAMP 2 (EPAC2) associated with total fat mass and leg fat mass. Gene cards [71] reveal that RAPGEF4 has many aliases for which it is used in the literature. It has been implicated in metabolic pathways related to obesity through its role in signal transduction [72]. In 2015, Hwang et al. [72], discovered that in East Asians, variants in RAPGEF4 encoding EPAC2 are associated with fasting blood glucose. RAPGEF4 is expressed mainly in the brain and adrenal gland but also in the heart, small intestine, and testis [73]. A notable role of RAPGEF4 is in the pancreatic islet for the secretion of insulin and glucagon

and in the brain for the release of neurotransmitters [74-77]. Nevertheless, the connection between RAPGEF4 and obesity is still under investigation and our study would contribute to the growing evidence that this gene could influence obesity-related phenotypes.

Parkin RBR E3 Ubiquitin Protein Ligase (PARK2), also known as Parkin is known to be involved in the development of Parkinson's disease [78]. However, similar to our study, recent research has shown its involvement in pathways related to mitochondrial and energy metabolism. PARK2 has been shown to take part in adipose tissue browning and beige adipocyte thermogenesis [79-81]. Obesity causes metabolic stress and PARK2 protects mitochondrial function [82]. Furthermore, according to a recent study by Moore et al. [83] it regulates mitochondrial homeostasis. The high suggestive value in this study reinforces the previous link found between PARK2 and obesity.

The genes RP11-847H18.2, RP11-103J8.1, RP11-406O16.1, and RP11-472K22.1 are long intergenic non-coding RNAs (lincRNAs). A GWAS study by Chiang et al. [84] reported a significant association between RP11-638 L3.1 and obesity. The AC073257.2 gene is also an IncRNA and is involved in the regulation of gene expression and metabolic processes and its role as a gene enhancer has been demonstrated [85].

USP13 (ubiquitin-specific peptidase 13) is known to be involved in deubiquitinate processes that may influence cellular pathways. It is involved in cell cycle regulation [86], myoblast differentiation [87], and autophagy [88]. It also plays a role in numerous pathological conditions such as infection [89], inflammation [90], fibrosis [91], neurodegenerative diseases [92] and cancers [93]. Hence, it might be indirectly linked to obesity but mechanistic research is missing.

The FBLN7 gene encodes fibulin-7 which is expressed in numerous tissues including the placenta, eye, blood vessels, and cartilage implying possible importance in pathological conditions. It could have a possible role in early development [94], overexpression in certain cancers [95], angiogenesis [96], inflammation [97], and ocular disorders [98]. A role in decreasing angiogenesis and inflammation is suggested [95, 98]. However, more studies are needed to understand its relationship to obesity.

RAD9-HUS1-RAD1 Interacting Nuclear Orphan 1 (RHNO1) had the highest statistical significance in three phenotype categories: total fat mass, android fat mass, and trunk fat mass. In 2010, Kim et al. [99] were the first to publish about RHNO1 and its overexpression in breast cancer lines including tissues in both mRNA and protein. Although more studies are needed to confirm this association between total, android, and trunk fat mass and RHNO1, higher BMI lowers locoregional and distant recurrence-free survival and increases association with

mortality [100-101]. Its overexpression was associated with a decreased survival rate among hepatocellular carcinoma patients [102]. The DNA damage response (DDR) [103] detects, signals, and repairs damaged DNA without which mutations can occur leading to diseases such as cancer. A high-throughput screen [103] is a technique to determine which genes, proteins, or small molecules are involved in the response to DNA damage at the cellular level. Indeed, RHNO1 was reported in the high-throughput screen as a regulator for DDR and its interactions promote the activation of proteins integral to cellular DNA replication stress [104-105]. Similarly, in 2021, a study [106] was published about its overexpression in ovarian cancer and pan-cancer tissues. Both obesity and cancer are very complex diseases, and obesity impacts the tumor microenvironment as a modulating factor. This association of RHNO1 to obesity highlights the importance of having interventions that target obesity since they might be at risk of developing certain kinds of cancers.

Other genes involved in tumor progression are SIPA1L1, ATP8A1, KIF26A, STAU2, CTNND2, and PRIMA1, all show suggestive associations in our study. SIPA1L1 [107] activates GTPase-activating protein with tumor-suppressing effects. ATP8A1 has a role in intracellular transport and prevents apoptosis of cancer cells [108-109]. KIF26A is involved in intracellular transport [110] and is downregulated in breast cancer. STAU2 binds to mRNA [111-112] encoding proteins that take part in cell cycle regulation [113]. In a recent paper published in 2022, Wang et al. [114] demonstrated how STAU2 could serve as a drug target due to its relation to pancreatic adenocarcinoma, where it decreased cell growth, migration, and invasion and induced apoptosis. The involvement of above-described genes in both obesity and tumor development might open up new avenues of research.

Pleiotropic effects of genetic variations are common, and the CTNND2 gene could be discussed in that context. CTNND2 plays a role in brain functions [115]. Genetic alterations of CTNND2 can lead to neurocognitive disorders including cortical cataract-linked Alzheimer's [116] and Cri-du-chat syndrome [117], neurological disorder [118], including autism [119], schizophrenia [120], and intellectual disturbance [121]. It is also involved in cancer [112], and infectious diseases [123]. Obesity can lead to brain changes affecting cognitive areas and is associated with the risk of developing Alzheimer's disease [124]. Although there are inconsistencies in studies [125-126] about the impact of obesity on cognitive performance, we need more studies to assess the relationship between CTNND2, obesity, and neurological abnormalities. Mechanisms currently proposed for the link between obesity and central nervous system regulation implicate the leptin-melanocortin pathway.

The PRIMA1 gene is located on chromosome 11 and codes for a protein that is primarily involved in acetylcholinesterase [127] membrane targeting. While PRIMA1 has not been extensively researched concerning obesity, its role in neurobiology and metabolic processes suggests that it could be involved in appetite regulation and metabolic health. The human protein atlas [127] also suggests a potential association with cancer.

There is currently no published evidence of a direct association between tyrosyl protein sulfotransferase 2 (TPST2), HRAS-like suppressor 5 **(**HRASLS5)**,** AP002833.1**,** AC011288.2, RNU6-939P, rs777625, and rs117042944 and obesity although it was found above our suggestive threshold in our phenotypes. TPST2 is involved in tyrosine sulfation [128]. Chronic kidney patients showed decreased atherosclerosis and peritoneal macrophage adherence in TPST2 knockout [129]**.** Mammalian sperm-associated antigen 16 (SPAG16) encodes a protein essential for flagellar motility whereby mutation would lead to paralyzed flagella [130]. SPAG16 is essential for the survivability of the nucleus of germ cells during spermatogenesis. However, to our knowledge, it has not been directly associated with obesity, and further studies are needed to confirm its link to obesity.

These identified variants offer insights into how obesity is linked to different diseases and understanding how the distribution of fat affects the risk of obesity. However, further studies are needed to validate these associations in larger cohorts and different populations to establish the functional relevance of these genetic variants in relation to obesity pathophysiology.

3.6 Conclusion

In conclusion, this GWAS of body fat distribution determined by DXA does not identify a statistically significant genome-wide association. However, suggestive associations could contribute to the determination of the genetic architecture that influences the differential distribution of adipose tissue. Determining the genetics and mechanisms leading to differential fat distribution will lead to more precise risk assessment and interventions in obesity-related diseases.

3.7 Acknowledgements

We wish to thank the Velsera team, **Dr. Cera R. Fisher**, **Dave Roberson,** and **Ryan Poterfield** for their technical support and assistance on Biodata Catalyst. We thank **Mr. Seth Berke** for providing the application “Split by Chr” on the Bio Data Catalyst platform.

3.8 Conflict of interest

The authors report no conflicts of interest.

3.9 Funding

We received funding from Research Manitoba and the Biodata Catalyst.

Chapter 4: General Discussion

4.1 Discussion

This project used the systematic review design to compile a list of genetic variants reported in the literature to be associated with DXA-derived fat distribution phenotypes.
To uncover new genetic variants that might contribute to differential fat compartmentation in a Manitoban cross-sectional cohort a GWAS was conducted. Although the GWAS findings were suggestive rather than statically significant on a genome wide scale, they emphasise the potential of using methods to precisely determine fat distribution to gain insight into accurate diagnostics and related precision interventions. Future GWAS studies using DXA data are expected to enhance the understanding how genetic variants associate with differential fat distribution. Like other GWAS, many of the variants identified in this research are non-coding. Although understanding these non-coding regions is limited, they are valuable as they identify genes and pathways involved in disease mechanisms or intersect with regulatory elements involved in translational and post translational modifications. Non-coding regions can interact with epigenetic factors influencing the activity of a gene or bind to transcription factors without the DNA sequence being modified affecting the health of individuals.

When comparing the list of genes compiled in the systematic review with the suggestive loci determined in the TMPLR-GWAS three overlaps appear: FADS1, FTO, and SNP rs13014219 with no annotated locus nomination (**Table 8**). Despite the limited cohort size and statistic power two of the most penetrant genes associated with obesity (FADS1, FTO) top the list of suggestive loci, which adds an element of strength to the GWAS analysis. Therefore, it might be worth considering the other genes reported as legitimate candidate genes associated with the reported phenotypes. For now, they are a firs reference list for genes possibly associated to fat distribution phenotypes to wait replication in larger cohorts. However, the divergence observed may be dew to the differences in population-based genetics, for example, allele frequency variations and unique linkage disequilibrium patterns in the Manitoban cohort.

Of overall significance, a bioinformatics pipeline was developed using the Biodata Catalyst platform, standardising the analyses and adding a common tool to analyse additional datasets from the TMPLR cohort. The pipeline provides a foundation for future investigation into gene-environment interactions and their effects on disease occurrence.

4.2 Strengths and Limitations

We acknowledge the limitations of this study. The relatively small sample size of TMPLR dataset limited the statistical power to detect significant statistical associations. In this thesis project, no replication study to validate the genetic associations was included.

Despite the challenges, this research has unique strengths. The use of the Biodata Catalyst for analysis in this project enabled accurate and reliable results that can be replicated due to the availability of standardized workflows and efficient data storage and management. In addition, the Biodata Catalyst is an advanced tool and provides us with a platform that adheres to ethical considerations and regulatory standards for data use. I am proud to have contributed to its development. Furthermore, TMPLR dataset's inclusion of individuals with diverse ethnic and cultural backgrounds allowed for robust analysis, capturing unique genetic variations within the Manitoban population. Lastly, DXA data availability added significant strength to this study, as DXA is widely regarded as an imaging tool for accuracy, reliability, and precision to determine differential fat distribution.

4.2 Future Directions

The pipeline created in the Biodata Catalyst can be used to analyse more phenotypes in the TMPLR cohort and other larger cohorts. This will enable the validation of the genetic loci found in this research.

There is also the potential to investigate the mechanistic role of identified genetic variants which will deepen our understanding of how the genetic loci that we found affect fat distribution and metabolism which can lead to targeted interventions for obesity. Future research could focus on precision nutrition, using genetic insights to create personalized treatment plans to improve patient care of obesity and other metabolic disorders.

Overall, broadening the use of DXA in genetic studies will bridge the existing gaps in the literature and will offer more specific diagnostic and interventions.

For future projects, there is the possibility of integrating polygenic risk analysis to estimate participant’s genetic predisposition to obesity considering the combined effect of numerous genetic variants. This is a analysis which shows how the small amount of contribution of

individual genetic variants relates to an overall risk since obesity is influenced by many genetic variants.

Another area to explore is to perform a phenome-wide association (PheWAS) analysis based on the genetic variants that we identified from the systematic review and our TMPLR GWAS. A pheWAS would investigate the associations between the genetic variants identified in the papers and a wide range of phenotypic traits including diseases and clinical characteristics across a population. This study might help to identify new associations between our genetic variants and health outcomes including diseases other than obesity.

Table 8: SNPs common between systematic review and TMPLR data analysis

Phenotype	CHR	Gene Symbols	SNP ID	pos	Score	Score.SE	Score.Stat	Score.pval	Est	Est.SE
Legs Fat Mass	11	FADS1	rs174547	61570783	-0.00293	0.002783	-1.0524	0.292614	-378.173	359.3418
Android Fat Mass				61570783	-0.00154	0.018697	-0.08228	0.934427	-4.40052	53.48463
Gynoid Fat Mass				61570783	-0.01724	0.014371	-1.19968	0.230264	-83.4804	69.58568
Trunk Fat Mass				61570783	-0.00243	0.003909	-0.62147	0.53429	-158.966	255.7903
Arms Fat Mass				61570783	-0.0046	0.025288	-0.18205	0.855542	-7.19914	39.54437
Legs Fat Mass				61570783	-0.0076	0.006114	-1.24325	0.213774	-203.35	163.5627
Legs Fat Mass	13	Unknown	rs138014219	27409628	-0.00021	0.000493	-0.42379	0.671717	-860.117	2029.572
Android Fat Mass				27409628	-0.00419	0.003307	-1.26663	0.205289	-382.975	302.3584
Gynoid Fat Mass				27409628	0.001647	0.002545	0.647273	0.517455	254.3365	392.9353
Trunk Fat Mass				27409628	-0.0005	0.000692	-0.73026	0.465228	-1056.05	1446.113
Arms Fat Mass				27409628	-0.01242	0.004479	-2.77297	0.005555	-619.098	223.262
Legs Fat Mass				27409628	0.000867	0.001082	0.801179	0.423028	740.5093	924.2742
Legs Fat Mass	16	FTO	rs9939609	53820527	0.00409	0.002939	1.391582	0.164049	473.5274	340.28
Android Fat Mass				53820527	0.036871	0.019716	1.870081	0.061473	94.84885	50.7191
Gynoid Fat Mass				53820527	0.00419	0.015211	0.275476	0.782951	18.11063	65.74317
Trunk Fat Mass				53820527	0.007862	0.004126	1.905359	0.056733	461.7792	242.358
Arms Fat Mass				53820527	-0.00112	0.026715	-0.04193	0.966552	-1.56967	37.43225
Legs Fat Mass				53820527	-0.00019	0.006467	-0.02887	0.976971	-4.46366	154.63

Chapter 5: Overall Conclusion

There is a lack of high-quality genetic studies analyzing obesity phenotypes via DXA in adult populations. The TMPLR-GWAS did not identify loci reaching genome-wide associators, due to a limited sample size, but provided the foundation for future research. Although there is an overlap in some genetic loci, FADS1, FTO, and SNP rs13014219, in the systematic review and the GWAS TMPLR analysis, the lack of statistical significance suggests the need for replication studies across different populations and study designs. More refined GWAS and mechanistic studies should strengthen existing evidence and identify additional loci as targets for precision interventions. Although this study was designed for the Manitoba cohort/community, replications in larger international cohorts, such as the UK Biobank represent an immediate step towards these goals.

Chapter 6: References
All references and citations were made according to the American Journal of Clinical

1. S. Camacho, A. Ruppel, Is the calorie concept a real solution to the obesity epidemic?, Glob Health Action. 10 (1) (2017)1-12.
2. S. Ibrahim, Z. Akram, A. Noreen, M.T. Baig, S. Sheikh, A. Huma, et al., Overweight and obesity prevalence and predictors in people living in Karachi. J Pharmaceut Res Int. 25(7) (2021)631-640.
3. N. Chami, M. Preuss, R.W. Walker, Moscati A, R.J.F. Loos, The role of polygenic susceptibility to obesity among carriers of pathogenic mutations in MC4R in the UK Biobank population, PLoS Med 17 (7) (2020) e1003196
4. Statistics Canada. Overweight and obese adults, 2018 [Internet] 2017 [cited 2024 Oct 1]. Available from: https://www150.statcan.gc.ca/n1/pub/82-625-x/2019001/article/00005-eng.htm
5. B. X. Tran, A.V. Nair, S. Kuhle, A. Ohinmaa, P.J. Veugelers, Cost analyses of obesity in Canada: scope, quality, and implications. Cost Effectiveness Resource Alloc, 11 (3) (2013)
6. M. Abdelaal, C.W le Roux, N.G Docherty, Morbidity and mortality associated with obesity. Ann Transl Med 5 (7) (2017)161
7. C. Zammit, H. Liddicoat, I. Moonsie, H. Makker, Obesity and respiratory diseases. Int J Gen Med (2010)335–343.
8. B. Mokhlesi, Obesity hypoventilation syndrome: a state-of-the-art review. Respir Care 55 (2010)1347–1362.
9. H. Bliddal, A.R. Leeds, R. Christensen, Osteoarthritis, obesity and weight loss: evidence, hypotheses and horizons - a scoping review. Obes Rev 15 (2014)578–586.
10. S. De Ferranti, D. Mozaffarian, The perfect storm: obesity, adipocyte dysfunction, and metabolic consequences. Clin Chem 54 (2008)945–955.
11. P.M. Visscher, M.A. Brown, M.I. McCarthy, J. Yang. Five years of GWAS discovery. Am J Hum Genet 90 (2012)7–24.
12. P.M. Visscher, 10 years of GWAS discovery: biology, function, and translation. Am J Hum Genet 101 (2017)5–22.
13. J. MacArthur. The new NHGRI-EBI Catalog of published genome-wide association studies (GWAS Catalog). Nucleic Acids Res (2017)45–D901.

14. M. Rask-Andersen, T. Karlsson, W.E. Ek, Å. Johansson, Genome-wide association study of body fat distribution identifies adiposity loci and sex-specific genetic effects. Nat Commun 10 (2019)339
15. A. Scuteri, S. Sanna, W.M. Chen, M. Uda, G. Albai, J. Strait, et al. Genome-wide association scan shows genetic variants in the FTO gene are associated with obesity-related traits. PLoS Genet 3 (2007)
16. V. Tam, N. Patel, M. Turcotte, Y. Bossé, G. Paré, D. Meyre, Benefits and limitations of genome-wide association studies. Nat Rev Genet 20 (8) (2019)467-484.
17. A. Javed, M. Jumean, M.H. Murad, D. Okorodudu, S. Kumar, V.K. Somers, et al., Diagnostic performance of body mass index to identify obesity as defined by body adiposity in children and adolescents: a systematic review and meta-analysis. Pediatr. Obes. 10 (2015)234–244
18. A. Mohammad, A.H Ziyab, T. Mohammad., Anthropometric and DXA-derived measures of body composition in relation to pre-diabetes among adults. BMJ Open Diabetes Res Care. 11(5) (2023)e003412.
19. D. Mackay, R.C. Mollard, M. Granger, S. Bruce, H. Blewett, J. Carlberg, et al., The Manitoba Personalized Lifestyle Research (TMPLR) study protocol: a multicentre bidirectional observational cohort study with administrative health record linkage investigating the interactions between lifestyle and health in Manitoba, Canada. BMJ Publishing Group Ltd 9 (10) (2019)
20. Statistics Canada. Overview of weight and height measurements on World Obesity Day. StatCan Plus. Published March 4, 2024. Available from: https://www.statcan.gc.ca/o1/en/plus/5742-overview-weight-and-height-measurements-world-obesity-day
21. World Health Organization. Factsheet Obesity and Overweight. World Health Organization. Geneva, Switzerland; 311 (2012).
22. G. Thorleifsson, G.B. Walters, D.F. Gudbjartsson, et al. Genomewide association yields new sequence variants at seven loci that associate with measures of obesity. Nat Genet. 41 (2009)18–24.
23. J.A. Shepherd, B. Fan, Y. Lu, X.P. Wu, W.K. Wacker, D.L. Ergun, et al. A multinational study to develop universal standardization of whole-body bone density and composition using GE Healthcare Lunar and Hologic DXA systems. J Bone Miner Res. 27 (10) (2012) 2208–16.

24. E.G. Kamel, G. McNeill, T.S. Han, F.W. Smith, A. Avenell, L. Davidson, et al. Measurement of abdominal fat by magnetic resonance imaging, dual-energy X-ray absorptiometry and anthropometry in non-obese men and women. Int J Obes Relat Metab Disord. 23 (7) (1999) 686–92.
25. A.E. Locke, B. Kahali, S.I. Berndt, A.E. Justice, T.H. Pers, F.R. Day, et al. Genetic studies of body mass index yield new insights for obesity biology. Nature. 518 (7538) (2015) 197–206.
26. A.S. Al-Goblan, M.A. Al-Alfi, M.Z. Khan. Mechanism linking diabetes mellitus and obesity. Diabetes Metab Syndr Obes. 7 (2014) 587–91.
27. Ovid MEDLINE Database. Search conducted via Ovid platform, University of Manitoba library, (2024). Available at: https://libguides.lib.umanitoba.ca/databases.
28. Covidence systematic review software. Veritas Health Innovation, Melbourne, Australia. Available at: www.covidence.org.
29. W.M. Bramer, J. Milic, F. Mast. Reviewing retrieved references for inclusion in systematic reviews using EndNote. J Med Libr Assoc. 105 (1) (2017) 84–7.
30. Sohani, Z.N.; Meyre, D.; de Souza, R.J.; Joseph, P.G.; Gandhi, M.; Dennis, B.B.; Norman, G.; Anand, S.S. Assessing the quality of published genetic association studies in meta-analyses: The quality of genetic studies (Q-Genie) tool. BMC Genet. 2015, 16, 50.
31. A. Liberati, D.G. Altman, J. Tetzlaff, C. Mulrow, P.C. Gotzsche, J.P. Ioannidis, et al. The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate healthcare interventions: explanation and elaboration. BMJ. 33 (2009).
32. M. Arrizabalaga, E. Larrarte, J. Margareto, S. Maldonado-Martin, L. Barrenechea, I. Labayen. Preliminary findings on the influence of FTO rs9939609 and MC4R rs17782313 polymorphisms on resting energy expenditure, leptin and thyrotropin levels in obese non-morbid premenopausal women. J Physiol Biochem. 70 (2014) 255–62.
33. E.A.D. Clifton, F.R. Day, E. De Lucia Rolfe, N.G. Forouhi, S. Brage, S.J. Griffin, et al. Associations between body mass index-related genetic variants and adult body composition: The Fenland cohort study. Int J Obes. 41 (2017) 613–9.
34. H. Guo, L. Zhang, C. Zhu, F. Yan, S. Wang, S. Zhu, X. Ma. A single nucleotide polymorphism in the FADS1 gene is associated with plasma fatty acid and lipid profiles and might explain gender difference in body fat distribution. Lipids Health Dis. 16 (2017) 67.

35. C. Lara-Castro, G.R. Hunter, J.C. Lovejoy, B.A. Gower, J.R. Fernandez. Association of the intestinal fatty acid-binding protein Ala54Thr polymorphism and abdominal adipose tissue in African-American and Caucasian women. J Clin Endocrinol Metab. 90 (2005) 1196–201.
36. K.M. Livingstone, M.H. Tan, G. Abbott, R.L. Duckham, L. Croft, J. Ward, et al. Discovery genome-wide association study of body composition in 4,386 adults from the UK Biobank's pilot imaging enhancement study. Front Endocrinol. 12 (2021) 1–7.
37. S. Pecioska, M.C. Zillikens, P. Henneman, P.J. Snijders, B.A. Oostra, C.M. Duijn, Y.S. Aulchenko. Association between type 2 diabetes loci and measures of fatness. PLoS One. 5 (2010) 1–5.
38. V. Sahibdeen, N.J. Crowther, H. Soodyall, L.M. Hendry, R.J. Munthali, S. Hazelhurst, et al. Genetic variants in SEC16B are associated with body composition in black South Africans. Nutr Diabetes. 8 (2018) 43.
39. M. Sharma, A. Misra, N.K. Vikram, B. Suryaprakash, S. Chhabra, N. Garg, et al. Genotype of the LMNA 1908C>T variant is associated with generalized obesity in Asian Indians in North India. Clin Endocrinol. 75 (2011) 642–9.
40. M. Sharma, N.K. Vikram, A. Misra, S. Bhatt, M. Tarique, H.A. Parray, et al. Assessment of 11-b hydroxysteroid dehydrogenase (11-bHSD1) 4478T>G and tumor necrosis factor-a (TNF-a)-308G>A polymorphisms with obesity and insulin resistance in Asian Indians in North India. Mol Biol Rep. 40 (2013) 6261–70.
41. W.J. Xiao, J.W. He, H. Zhang, W.W. Hu, J.M. Gu, H. Yue, et al. ALOX12 polymorphisms are associated with fat mass but not peak bone mineral density in Chinese nuclear families. Int J Obes. 35 (2011) 378–86.
42. J. Yu, Y. Ke, J. He, H. Zhang, W. Hu, Y. Hu, et al. No association between LRP5 gene polymorphisms and bone and obesity phenotypes in Chinese male offspring nuclear families. Acta Pharmacol Sin. 31 (2010) 1464–9.
43. W. Yu, Z. Zhang, W. Fu, J. He, C. Wang, Z. Zhang. Association between LGR4 polymorphisms and peak bone mineral density and body composition. J Bone Miner Metab. 38 (2020) 658–69.
44. H. Yue, J. He, H. Zhang, C. Wang, W. Hu, J. Gu, et al. Contribution of myostatin gene polymorphisms to normal variation in lean mass, fat mass and peak BMD in Chinese male offspring. Acta Pharmacol Sin. 33 (2012) 660–7.

45. H. Yue, J. He, Y. Ke, H. Zhang, X. Wang, W. Hu, et al. Association of single nucleotide polymorphism Rs2236518 in PRDM16 gene with BMI in Chinese males. Acta Pharmacol Sin. 34 (2013) 710–6.
46. NHGRI-EBI GWAS Catalog [Internet]. European Bioinformatics Institute; 2024 [cited 2024 Nov]. Available from: https://www.ebi.ac.uk/gwas.
47. Hirschhorn J.N., Daly M.J. Genome-wide association studies for common diseases and complex traits. Nature Reviews Genetics. 6 (2005) 95-108. https://doi.org/10.1038/nrg1521
48. Nebert D.W. Extreme discordant phenotype methodology: An intuitive approach to genetic dissection of complex traits. Proceedings of the National Academy of Sciences. 97 (2000) 14584-14586. https://doi.org/10.1073/pnas.97.26.14584
49. Colhoun H.M., McKeigue P.M., Davey Smith G. Problems of reporting genetic associations with complex outcomes. The Lancet. 361 (2003) 865-872. https://doi.org/10.1016/S0140-6736(03)12715-8
50. F.C.S. Fox, et al. Genome-wide association for abdominal subcutaneous and visceral adipose reveals a novel locus for visceral fat in women. PLoS Genet. 8 (2012).
51. A.Y. Chu, et al. Multiethnic genome-wide meta-analysis of ectopic fat depots identifies loci associated with adipocyte development and differentiation. Nat Genet. 49 (2017) 125-130.
52. Y. Liu, et al. Genetic architecture of 11 organ traits derived from abdominal MRI using deep learning. eLife. 10 (2021).
53. M. Abdelaal, C.W le Roux, N.G. Docherty, Morbidity and mortality associated with obesity. Ann Transl Med 5 (7) (2017)161.
54. Fransoo, R., Martens, P., Prior, H., Chateau, D., McDougall, C., Schultz, J., McGowan, K.-L., Soodeen, R.-A., & Bailly, A. Adult Obesity in Manitoba: Prevalence, Associations, & Outcomes (3rd ed.). Manitoba Centre for Health Policy, Department of Community Health Sciences, Faculty of Medicine, University of Manitoba. (2011).
55. K. Karastergiou, S.R. Smith, A.S. Greenberg, S.K. Fried, Sex differences in human adipose tissues: the biology of pear shape. Biol Sex Differ 3(2012)13.
56. QIAGEN, QIAamp DNA Mini Kit and QIAamp DNA Blood Mini Kit Handbook. (2023) [cited October 1, 2024]. Available from: https://www.qiagen.com/.
57. D.L. Ergun, M.P. Rothney, M.K. Oates, Y. Xia, W.K. Wacker, N.C. Binkley, Visceral adipose tissue quantification using Lunar Prodigy. J Clin Densitom. 16 (2013) 75–8.

58. Genome Québec, Genome Québec Innovation Centre - Illumina Sequencing Services. Montréal, QC: Genome Québec (2023) [cited October 1, 2024]. Available from: https://www.genomequebec.com.
59. The NHLBI BioData Catalyst, National Heart, Lung, and Blood Institute, National Institutes of Health, U.S. Department of Health and Human Services (2020)
60. University of Washington Genetic Analysis Center, Genesis Tutorial 2023. Seattle, WA: University of Washington (2023) [cited October 1, 2024]. Available from: https://uw-gac.github.io/SISG_2023.
61. S. Purcell, S.S. Cherny, P.C. Sham, Genetic Power Calculator: design of linkage and association genetic mapping studies of complex traits. Bioinformatics. 19 (2003) 149–150.
62. E.P. Hong, J.W. Park, Sample size and statistical power calculation in genetic association studies. Genomics Inform. 10 (2) (2012) 117–122.
63. C.N. Camp, M.C. Spencer, Genome-wide association studies and their impact on the study of complex diseases. Hum Genet 107 (1) (2000) 50–3.
64. P. Deloukas, S. Kanoni, C. Willenborg, M. Farral, T.L. Assimes, J.R. Thompson et al., Large-scale association analysis identifies new risk loci for coronary artery disease. Nat Genet 45 (1) (2013) 25–33.
65. RStudio Team, RStudio: Integrated Development for R. RStudio, PBC (2023). Available from: https://posit.co.
66. A. Manichaikul, J.C. Mychaleckyj, S.S. Rich, K. Daly, M. Sale, Robust relationship inference in genome-wide association studies. Bioinformatics 26 (22) (2010) 2857–2863.
67. M. Lynch, B. Walsh, Genetics and analysis of quantitative traits. Sunderland, MA: Sinauer Associates (1998).
68. C.S. Fox, Y. Liu, C.C. White, M. Feitosa, A.V. Smith, N. Heard-Costa, K. Lohmane, et al., Genome-wide association for abdominal subcutaneous and visceral adipose reveals a novel locus for visceral fat in women. PLoS Genet, 8 (5) (2012) e1002695.
69. A.Y. Chu, X. Deng, V.A. Fisher, A. Drong, Y. Zhang, M.F. Feitosa, et al., Multiethnic genome-wide meta-analysis of ectopic fat depots identifies loci associated with adipocyte development and differentiation. Nat Genet 49 (1) (2017) 125–130.

70. Y. Liu, N. Basty, B. Whitcher, J.D. Bell, E.P. Sorokin, N.V Bruggen, et al., Genetic architecture of 11 organ traits derived from abdominal MRI using deep learning. eLife. 10 (2021).
71. GeneCards. RAPGEF4 (EPAC2). 2024. Retrieved 2024 Oct 10 from: https://www.genecards.org/cgi-bin/carddisp.pl?gene=RAPGEF4&keywords=RAPGEF4
72. J.Y. Hwang, X. Sim, Y. Wu, J. Liang, Y. Tabara, C. Hu, K. Hara, C.H.T. Tam, Q. Cai, Q. Zhao, et al., Genome-wide association meta-analysis identifies novel variants associated with fasting plasma glucose in East Asians, Diabetes 64 (5) (2015) 291–298.
73. H. Kawasaki, G.M. Springett, N. Mochizuki, S. Toki, M. Nakaya, M. Matsuda, D.E. Housman, A.M. Graybiel, A family of cAMP-binding proteins that directly activate Rap1, Science 282 (5396) (1998) 2275–2279.
74. K. Sugawara, T. Shibasaki, H. Takahashi, S. Seino, Structure and functional roles of Epac2 (Rapgef4), Gene 575 (2016) 577–583.
75. T. Shibasaki, H. Takahashi, T. Miki, Y. Sunaga, K. Matsumura, M. Yamanaka, et al., Essential role of Epac2/Rap1 signaling in regulation of insulin granule dynamics by cAMP, Proc. Natl. Acad. Sci. U.S.A. 104 (49) (2007) 19333–19338.
76. Y.Z. De Marinis, A. Salehi, C.E. Ward, Q. Zhang, F. Abdulkader, M. Bengtsson, et al., GLP-1 inhibits and adrenaline stimulates glucagon release by differential modulation of N- and L-type Ca^{2+} channel-dependent exocytosis, Cell Metab. 11 (6) (2010) 543–553.
77. N. Ozaki, T. Shibasaki, Y. Kashima, T. Miki, K. Takahashi, H. Ueno, et al., cAMP-GEFII is a direct target of cAMP in regulated exocytosis, Nat. Cell Biol. 2 (2000) 805–811.
78. GBD 2016 Parkinson's Disease Collaborators, Global, regional, and national burden of Parkinson's disease, 1990-2016: a systematic analysis for the Global Burden of Disease Study 2016, Lancet Neurol. 17 (11) (2018) 939–953. doi:10.1016/S1474-4422(18)30295-3.
79. D. Taylor, R.A. Gottlieb, Parkin-mediated mitophagy is downregulated in browning of white adipose tissue, Obes. 25 (2017) 704–712.
80. X. Lu, S. Altshuler-Keylin, Q. Wang, Y. Chen, C.H. Sponton, K. Ikeda, et al., Mitophagy controls beige adipocyte maintenance through a Parkin-dependent and UCP1-independent mechanism, Sci. Signal. 11 (527) (2018).

81. M. Cairó, L. Campderrós, A. Gavaldà-Navarro, R. Cereijo, A. Delgado-Anglés, T. Quesada-López, et al., Parkin controls brown adipose tissue plasticity in response to adaptive thermogenesis, EMBO Rep. 20 (5) (2019).
82. C. Cui, S. Chen, J. Qiao, L. Qing, L. Wang, T. He, et al., PINK1-Parkin alleviates metabolic stress induced by obesity in adipose tissue and in 3T3-L1 preadipocytes, Biochem. Biophys. Res. Commun. 498 (3) (2018) 445–452.
83. T.M. Moore, L. Cheng, D.M. Wolf, J. Ngo, M. Segawa, X. Zhu, et al., Parkin regulates adiposity by coordinating mitophagy with mitochondrial biogenesis in white adipocytes, Nat. Commun. 13 (2022) 6661.
84. K.-M. Chiang, H.-C. Chang, H.-C. Yang, C.-H. Chen, H.-H. Chen, W.-J. Lee, et al., Genome-wide association study of morbid obesity in Han Chinese, BMC Genet. 20 (1) (2019) 97.
85. H. Huang, S. Fu, D. Liu, Detection and analysis of the Hedgehog signaling pathway-related long non-coding RNA (lncRNA) expression profiles in keloid, Med. Sci. Monit. 24 (2018) 9032–9044.
86. E. Petsalaki, G. Zachos, DNA damage response proteins regulating mitotic cell division: Double agents preserving genome stability, FEBS J. 287 (9) (2020) 1700–1721.
87. Z. Zhang, K. Deng, Z. Kang, F. Wang, Y. Fan, MicroRNA profiling reveals miR-145-5p inhibits goat myoblast differentiation by targeting the coding domain sequence of USP13, FASEB J. 36 (7) (2022).
88. I. Dikic, Z. Elazar, Mechanism and medical implications of mammalian autophagy, Nat. Rev. Mol. Cell Biol. 19 (6) (2018) 349–364.
89. D.S. Aaronson, C.M. Horvath, A road map for those who don't know JAK-STAT, Science 296 (5573) (2002) 1653–1655.
90. F. Yu, Y. Li, Q. Ye, J. Miao, S.J. Taleb, Y. Zhao, et al., Lipopolysaccharide reduces USP13 stability through c-Jun N-terminal kinase activation in Kupffer cells, J. Cell Physiol. 236 (6) (2021) 4360–4368.
91. S.G.S. Suri, G. Kaur, C.K. Jha, M. Tiwari, Understanding idiopathic pulmonary fibrosis - clinical features, molecular mechanism and therapies, Exp. Gerontol. 153 (2021) 111473.
92. U. Sengupta, R. Kayed, Amyloid β, Tau, and α-Synuclein aggregates in the pathogenesis, prognosis, and therapeutics for neurodegenerative diseases, Prog. Neurobiol. 214 (2022) 102270.

93. V.V. Senichkin, A.Y. Streletskaia, B. Zhivotovsky, G.S. Kopeina, Molecular comprehension of mcl-1: From gene structure to cancer therapy, Trends Cell Biol. 29 (7) (2019) 549–562.
94. J. Tsunezumi, H. Sugiura, L. Oinam, A. Ali, B.Q. Thang, A. Sada, et al., Fibulin-7, a heparin binding matricellular protein, promotes renal tubular calcification in mice, Matrix Biol. 74 (2018) 5–20.
95. S. de Vega, A. Kondo, M. Suzuki, H. Arai, S. Jiapaer, H. Sabit, et al., Fibulin-7 is overexpressed in glioblastomas and modulates glioblastoma neovascularization through interaction with angiopoietin-1, Int. J. Cancer 145 (2019) 2157–2169.
96. T. Ikeuchi, S. de Vega, P. Forcinito, A.D. Doyle, J. Amaral, I.R. Rodriguez, et al., Extracellular protein Fibulin-7 and its C-terminal fragment have in vivo antiangiogenic activity, Sci. Rep. 8 (2018) 17654.
97. P.P. Sarangi, P. Chakraborty, S.P. Dash, T. Ikeuchi, S. de Vega, K. Ambatipudi, et al., Cell adhesion protein fibulin-7 and its C-terminal fragment negatively regulate monocyte and macrophage migration and functions in vitro and in vivo, FASEB J. 32 (2018) 4889–4898.
98. P. Chakraborty, S.P. Dash, N. Dalpati, P. Kumar, D. Jain, P.P. Sarangi, A C-terminal fragment of adhesion protein Fibulin-7 inhibits growth of murine breast tumor by regulating macrophage reprogramming, FEBS J. (2020).
99. J.W. Kim, C. Fukukawa, K. Ueda, T. Nishidate, T. Katagiri, Y. Nakamura, Involvement of C12orf32 overexpression in breast carcinogenesis, Int. J. Oncol. 37 (2010) 861–867.
100. C. Bergom, T. Kelly, M. Bedi, H. Saeed, P. Prior, L.E. Rein, et al., Association of locoregional control with high body mass index in women undergoing breast conservation therapy for early-stage breast cancer, Int. J. Radiat. Oncol. Biol. Phys. 96 (1) (2016) 65–71.
101. F. Petrelli, A. Cortellini, A. Indini, G. Tomasello, M. Ghidini, O. Nigro, et al., Association of obesity with survival outcomes in patients with cancer: A systematic review and meta-analysis, JAMA Netw. Open. 4 (3) (2021). doi:10.1001/jamanetworkopen.2021.3520.
102. D. Du, S. Wang, T. Li, Z. Liu, M. Yang, L. Sun, et al., RHNO1 disruption inhibits cell proliferation and induces mitochondrial apoptosis via the PI3K/Akt pathway in hepatocellular carcinoma, Biochem. Biophys. Res. Commun. 673 (2023) 96–105.
103. J. FitzGerald, L.S. Murillo, G. O'Brien, E. O'Connell, A. O'Connor, K. Wu, et al., A high-throughput screen for small molecule modulators of the DNA damage response, PLOS ONE 9 (6) (2014). doi:10.1371/journal.pone.0098891. Available from:

https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0098891&type=printable.

104. C. Cotta-Ramusino, E.R. McDonald, K. Hurov, M.E. Sowa, J.W. Harper, S.J. Elledge, A DNA damage response screen identifies RHINO, a 9-1-1 and TopBP1 interacting protein required for ATR signaling, Science 332 (2011) 1313–1317.

105. L.A. Lindsey-Boltz, M.G. Kemp, C. Capp, A. Sancar, RHINO forms a stoichiometric complex with the 9-1-1 checkpoint clamp and mediates ATR-Chk1 signaling, Cell Cycle 14 (2015) 99–108.

106. C.J. Barger, L. Chee, M. Albahrani, C. Munoz-Trujillo, L. Boghean, C. Branick, et al., Coregulation and function of FOXM1/RHNO1 bidirectional genes in cancer, Elife 10 (2021).

107. Q. Gao, S.N. Srinivasan, S.N. Boyer, D.E. Wazer, V. Band, The E6 oncoproteins of high-risk papillomaviruses bind to a novel putative GAP protein, E6TP1, and target it for degradation, Mol. Cell. Biol. 19 (1) (1999) 733–744.

108. P. Magnaghi, R. D'Alessio, B. Valsasina, N. Avanzi, S. Rizzi, D. Asa, et al., Covalent and allosteric inhibitors of the ATPase VCP/p97 induce cancer cell death, Nat. Chem. Biol. 9 (2013) 548–556.

109. J.A. Seo, B. Kim, D.N. Dhanasekaran, B.K. Tsang, Y.S. Song, Curcumin induces apoptosis by inhibiting sarco/endoplasmic reticulum Ca2+ ATPase activity in ovarian cancer cells, Cancer Lett. 371 (2016) 30–37.

110. H. Huang, S. Fu, D. Liu, Detection and analysis of the Hedgehog signaling pathway-related long non-coding RNA (lncRNA) expression profiles in keloid, Med. Sci. Monit. 24 (2018) 9032–9044.

111. H. Kawasaki, G.M. Springett, N. Mochizuki, S. Toki, M. Nakaya, M. Matsuda, et al., A family of cAMP-binding proteins that directly activate Rap1, Science 282 (1998) 2275–2279.

112. K. Sugawara, T. Shibasaki, H. Takahashi, S. Seino, Structure and functional roles of Epac2 (Rapgef4), Gene 575 (2016) 577–583.

113. T. Shibasaki, H. Takahashi, T. Miki, Y. Sunaga, K. Matsumura, M. Yamanaka, et al., Essential role of Epac2/Rap1 signaling in regulation of insulin granule dynamics by cAMP, Proc. Natl. Acad. Sci. U.S.A. 104 (2007) 19333–19338.

114. K. Basu, N. Maurya, J. Kaur, R. Saxena, V. Gupta, R. Sihota, et al., Possible role of differentially expressing novel protein markers (ligatin and fibulin-7) in human aqueous humor and trabecular meshwork tissue in glaucoma progression, Cell Biol. Int. 43 (2019) 820–834.

115. I. Israely, R.M. Costa, C.W. Xie, A.J. Silva, K.S. Kosik, X. Liu, Deletion of the neuron-specific protein delta-catenin leads to severe cognitive and synaptic dysfunction, Curr. Biol. 14 (2004) 1657–1663.

116. P.G. Ridge, M.T. Ebbert, J.S. Kauwe, Genetics of Alzheimer's disease, Biomed. Res. Int. 2013 (2013) 254954.

117. J.M. Sardina, A.R. Walters, K.E. Singh, R.X. Owen, V.E. Kimonis, Amelioration of the typical cognitive phenotype in a patient with the 5pter deletion associated with Cri-du-chat syndrome in addition to a partial duplication of CTNND2, Am. J. Med. Genet. A 164A (2014) 1761–1764.

118. T.N. Turner, K. Sharma, E.C. Oh, Y.P. Liu, R.L. Collins, M.X. Sosa, et al., Loss of δ-catenin function in severe autism, Nature 520 (2015) 51–56.

119. M.G. Nivard, H. Mbarek, J.J. Hottenga, J.H. Smit, R. Jansen, B.W. Penninx, et al., Further confirmation of the association between anxiety and CTNND2: replication in humans, Genes Brain Behav. 13 (2014) 195–201.

120. C. Belcaro, S. Dipresa, G. Morini, V. Pecile, A. Skabar, A. Fabretto, CTNND2 deletion and intellectual disability, Gene 565 (2015) 146–149.

121. H. Wang, M. Yang, S. Su, L. Kang, R. Zhu, J. Shi, et al., Association of ZNF644, GRM6, and CTNND2 genes polymorphisms with high myopia, Zhonghua Yi Xue Za Zhi 94 (2014) 1289–1293.

122. Q. Lu, δ-Catenin dysregulation in cancer: interactions with E-cadherin and beyond, J. Pathol. 222 (2010) 119–123.

123. M.J. Mackinnon, C. Ndila, S. Uyoga, A. Macharia, R.W. Snow, G. Band, et al., Environmental correlation analysis for genes associated with protection against malaria, Mol. Biol. Evol. (2016).

124. G. Livingston, J. Huntley, A. Sommerlad, D. Ames, C. Ballard, S. Banerjee, et al., Dementia prevention, intervention, and care: 2020 report of the Lancet Commission, Lancet 396 (2020) 413–446.

125. L. Dye, N.B. Boyle, C. Champ, C. Lawton, The relationship between obesity and cognitive health and decline, Proc. Nutr. Soc. 76 (2017) 443–454.

126. E. Smith, P. Hay, L. Campbell, J.N. Trollor, A review of the association between obesity and cognitive function across the lifespan: implications for novel approaches to prevention and treatment, Obes. Rev. 12 (2011) 740–755.

127. Human Protein Atlas, PRIMA1 (Proline-Rich Membrane Anchor 1). Available from: https://www.proteinatlas.org/ENSG00000175785-PRIMA1. Accessed October 11, 2024.

128. S. Zhang, Q. Wang, W. Li, J. Chen, MIR100HG regulates CALD1 gene expression by targeting miR-142-5p to affect the progression of bladder cancer cells in vitro, as revealed by transcriptome sequencing, Front. Mol. Biosci. 8 (2022) 793493.

129. T.F. Li, H.J. Zeng, Z. Shan, R.Y. Ye, T.Y. Cheang, Y.J. Zhang, et al., Overexpression of kinesin superfamily members as prognostic biomarkers of breast cancer, Cancer Cell Int. 20 (2020) 123.

130. D.R. Nagarkatti-Gude, R. Jaimez, S.C. Henderson, M.E. Teves, Z. Zhang, J.F.III Strauss, Spag16, an axonemal central apparatus gene, encodes a male germ cell nuclear speckle protein that regulates SPAG16 mRNA expression. PLoS One 6(5) (2011) e20625.

Appendix A: Gel images for DNA Quality control

The images below present the DNA gel electrophoresis results from blood samples of TMPLR participants. Each gel contains two rows of data, with each column representing the results from one well. In each column, the top part of the gel should be clear and white, indicating that enough high-quality DNA has been extracted. DNA extraction was performed at least twice on each sample to ensure reliable and consistent results.

Lane 1 (Top row): Molecular weight marker (used as a reference to estimate the size of DNA fragments).

Lanes 2 to 16 (Top and Bottom rows): Participant samples, where the visible bands indicate successful DNA extraction. The consistent appearance of bands across all lanes suggests the DNA extraction was of high quality and was successfully performed for all samples.

Figure A1: Gel images Part 1

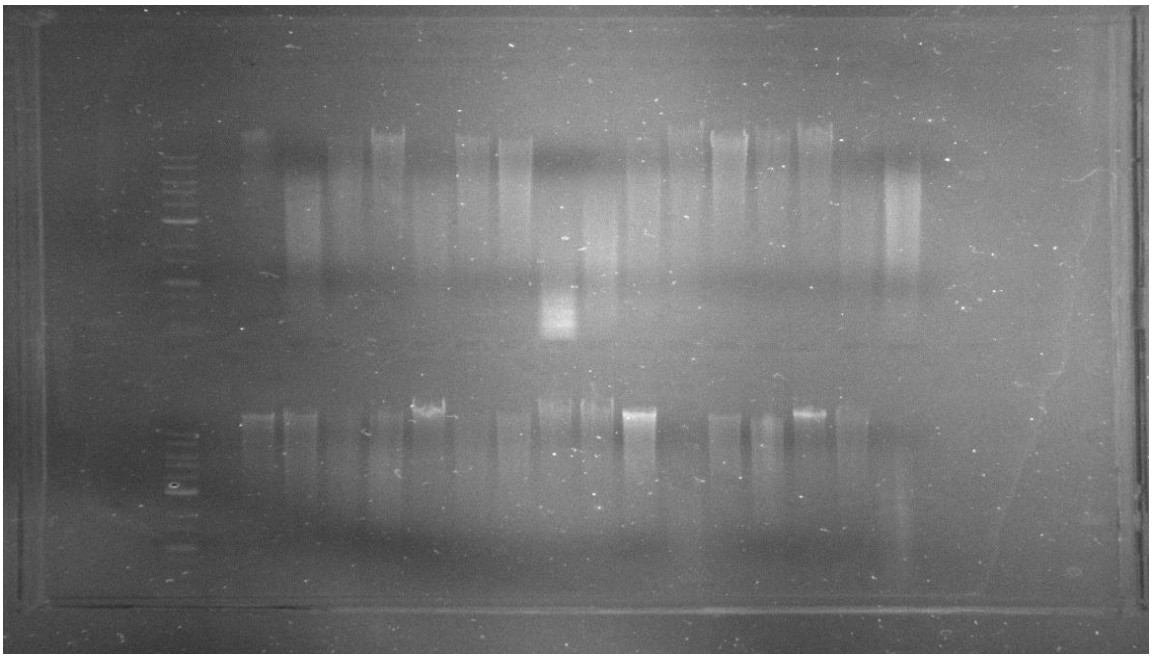


Figure A2: Gel images Part 2

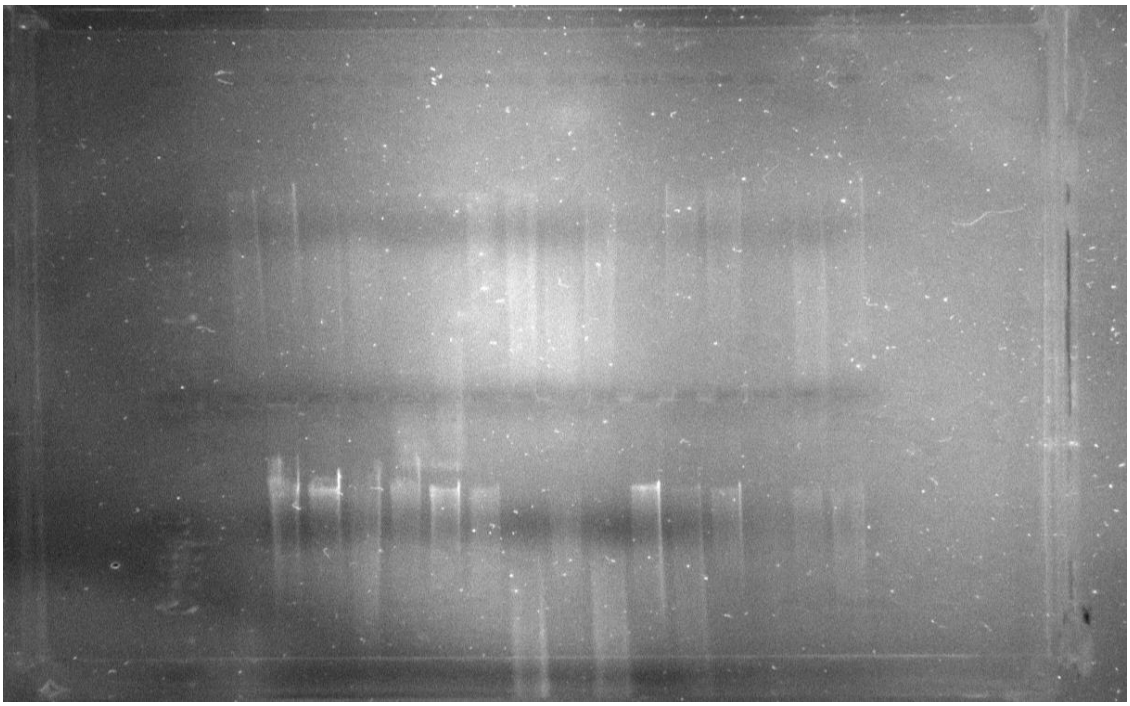


Figure A3: Gel images Part 3

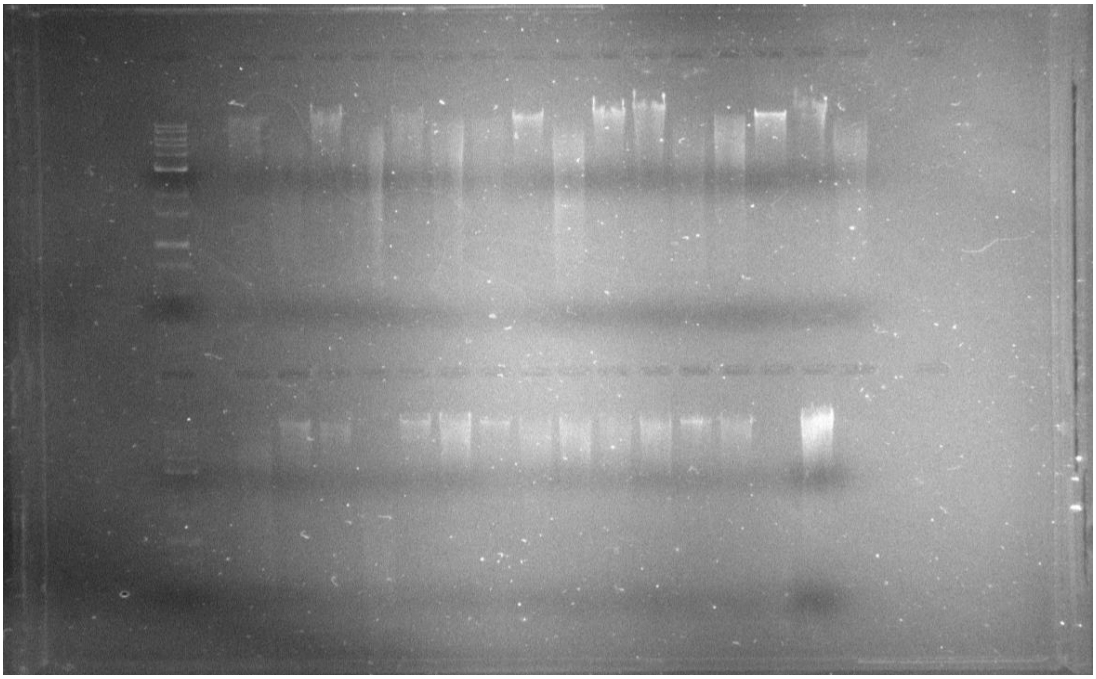


Figure A4: Gel images Part 4

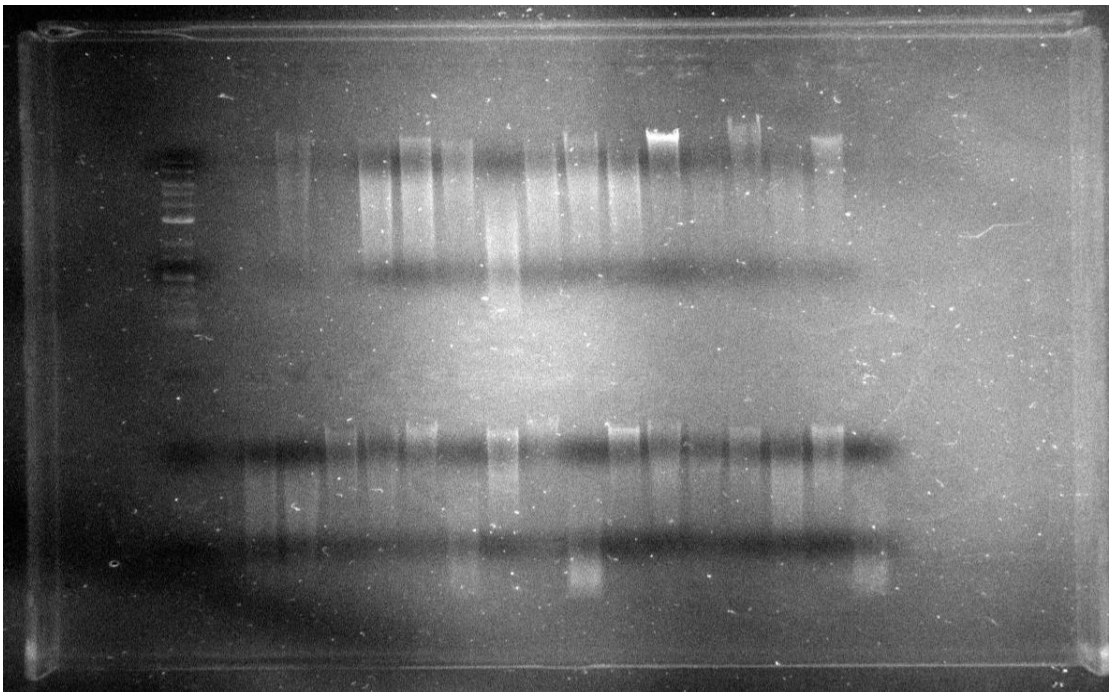


Figure A5: Gel images Part 5

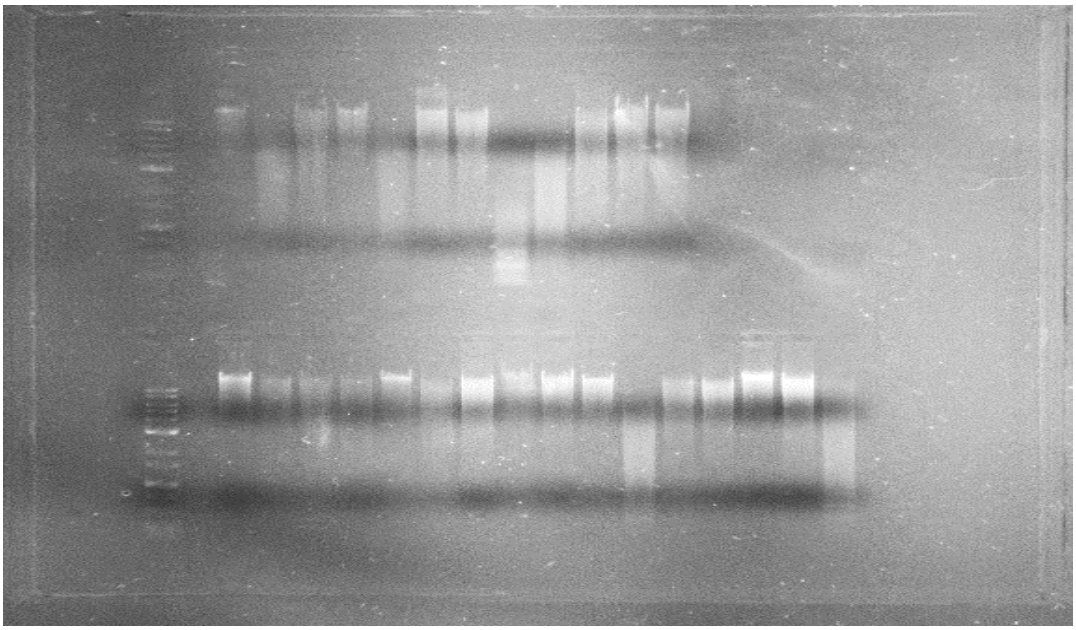


Figure A6: Gel images Part 6

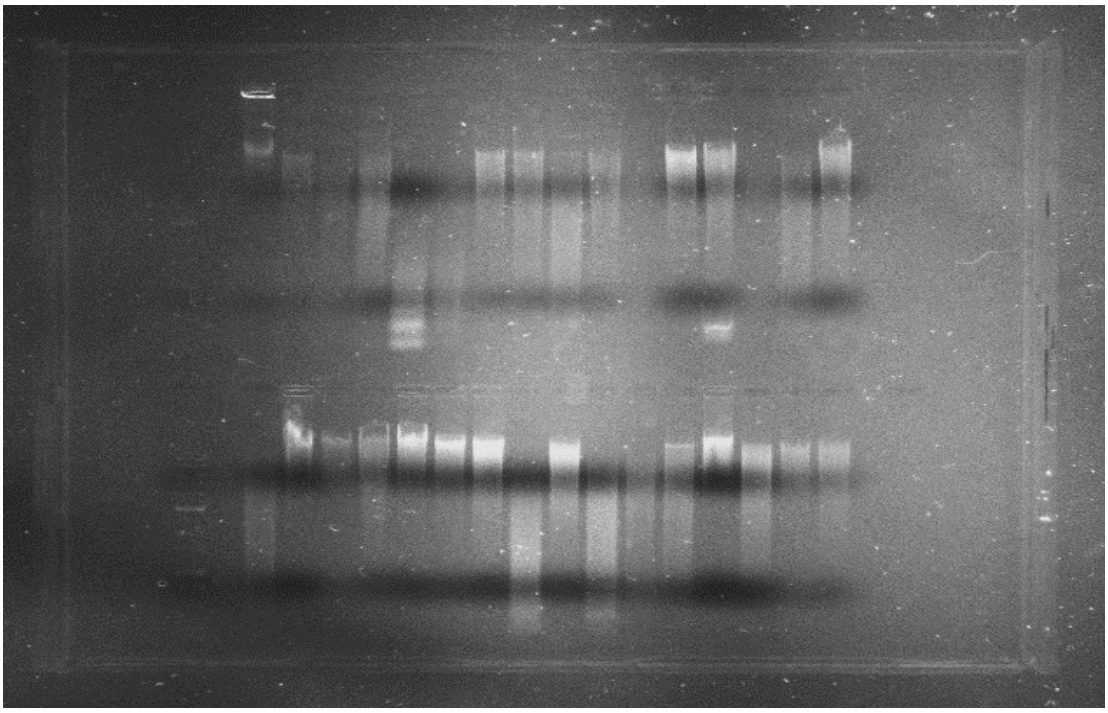


Figure A7: Gel images Part 7

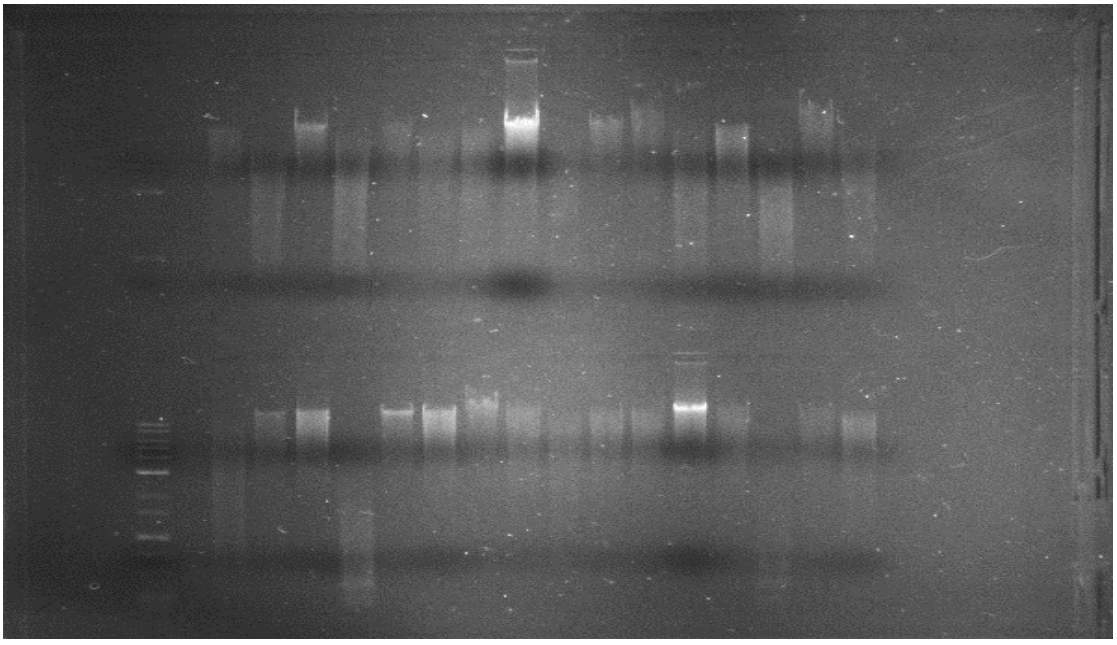


Figure A8: Gel images Part 8

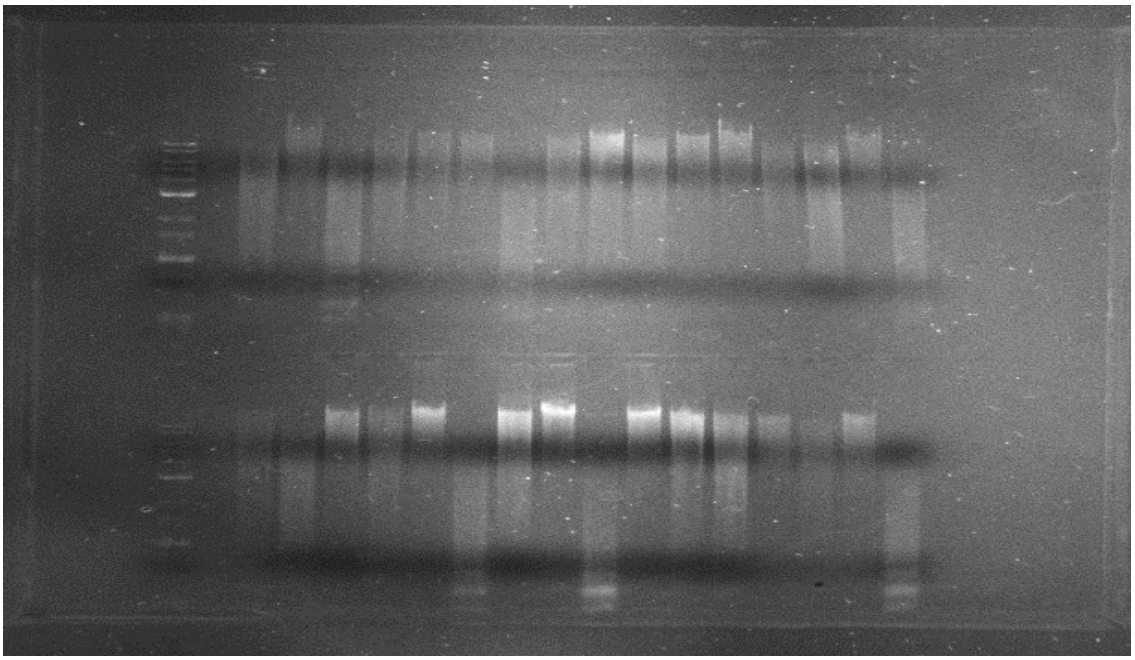


Figure A9: Gel images Part 9

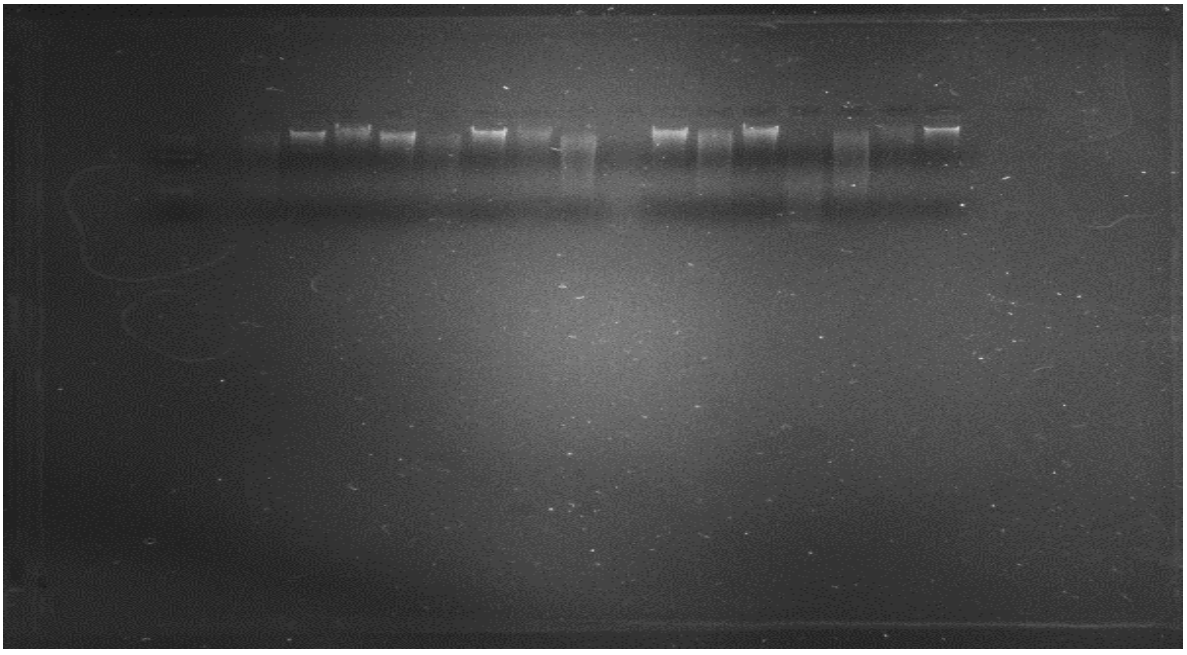


Figure A10: Gel images Part 10

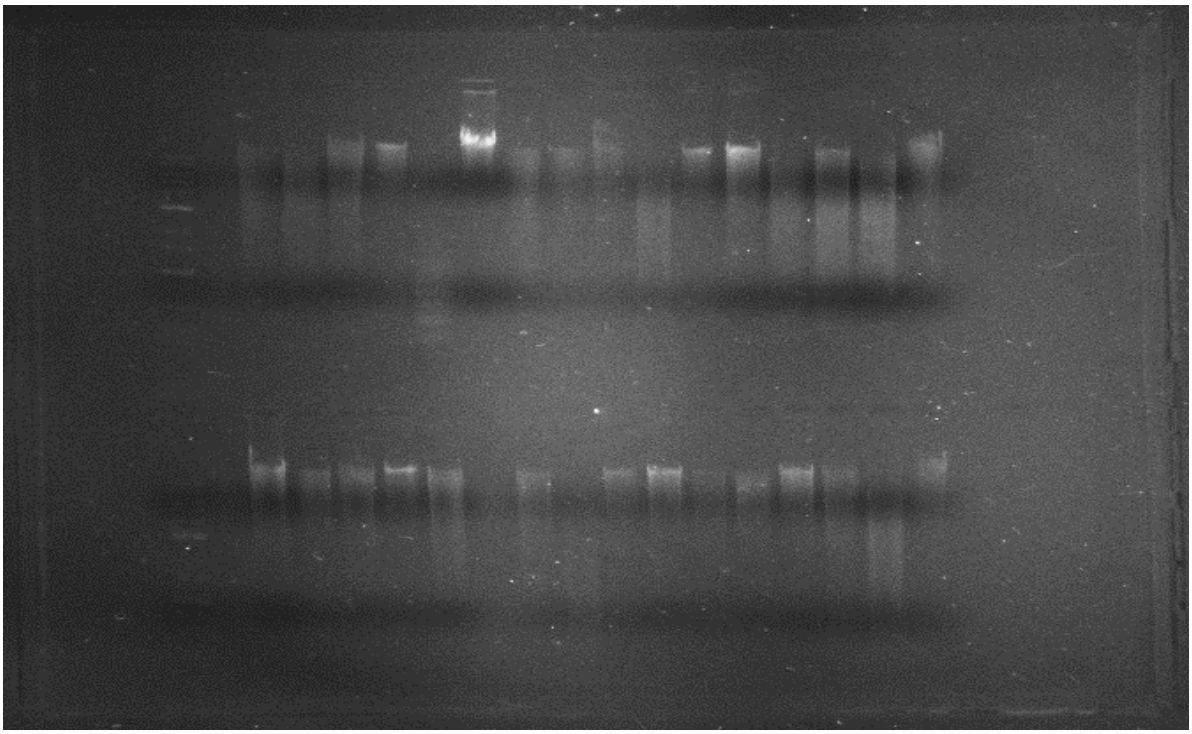


Figure A11: Gel images Part 11

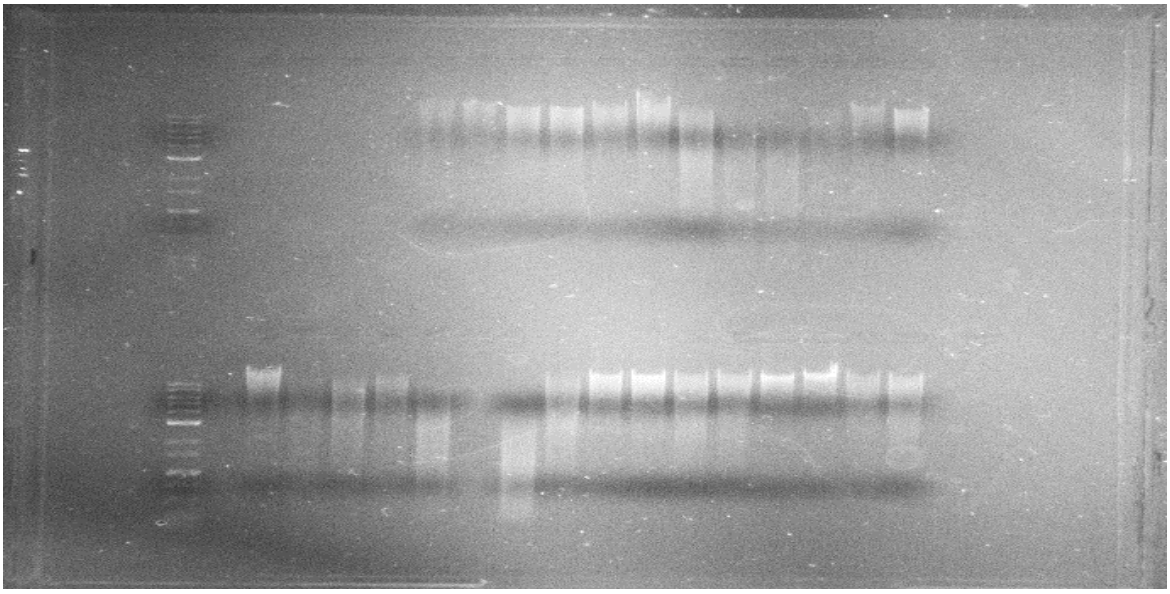


Figure A12: Gel images Part 12

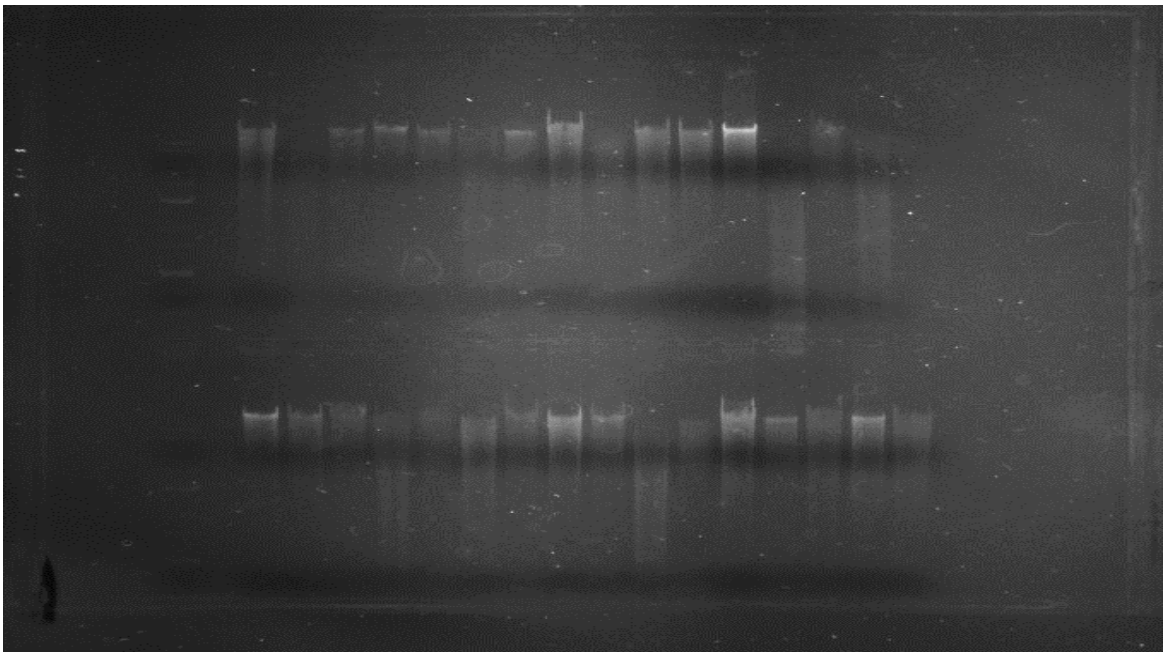


Figure A13: Gel images Part 13

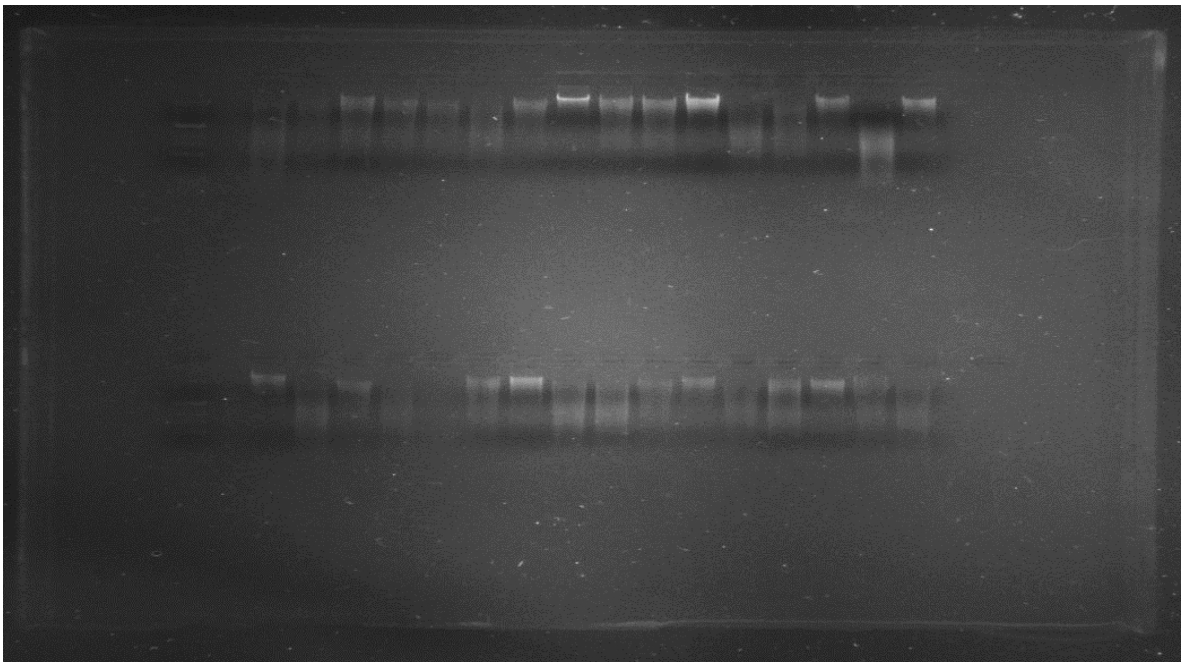


Figure A14: Gel images Part 14

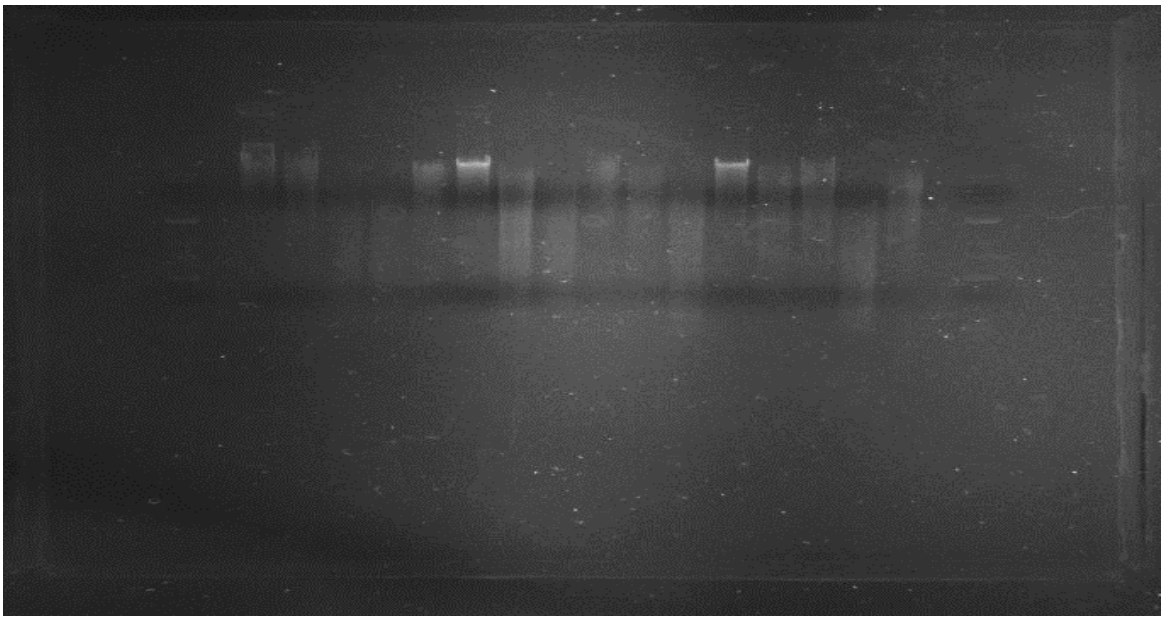


Figure A15: Gel images Part 15

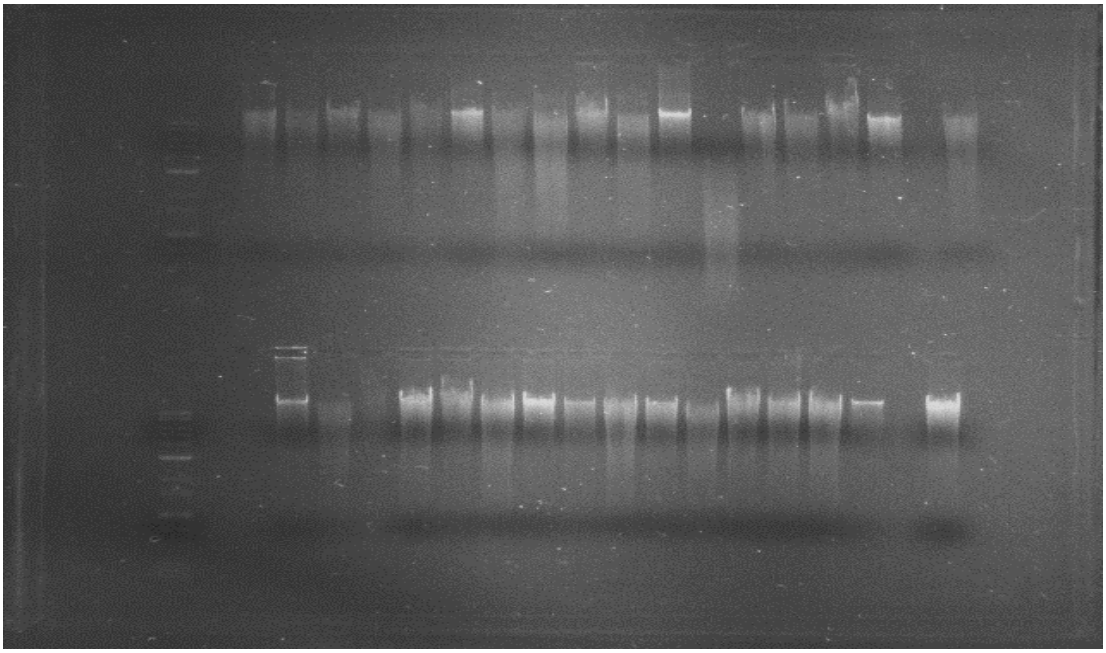


Figure A16: Gel images Part 16

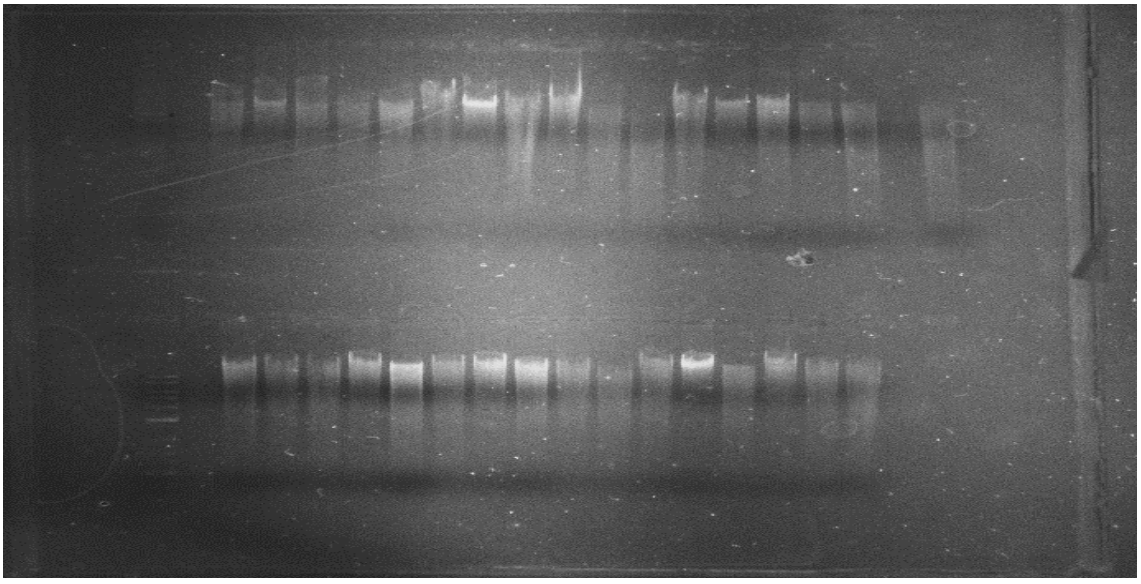


Figure A17: Gel images Part 17

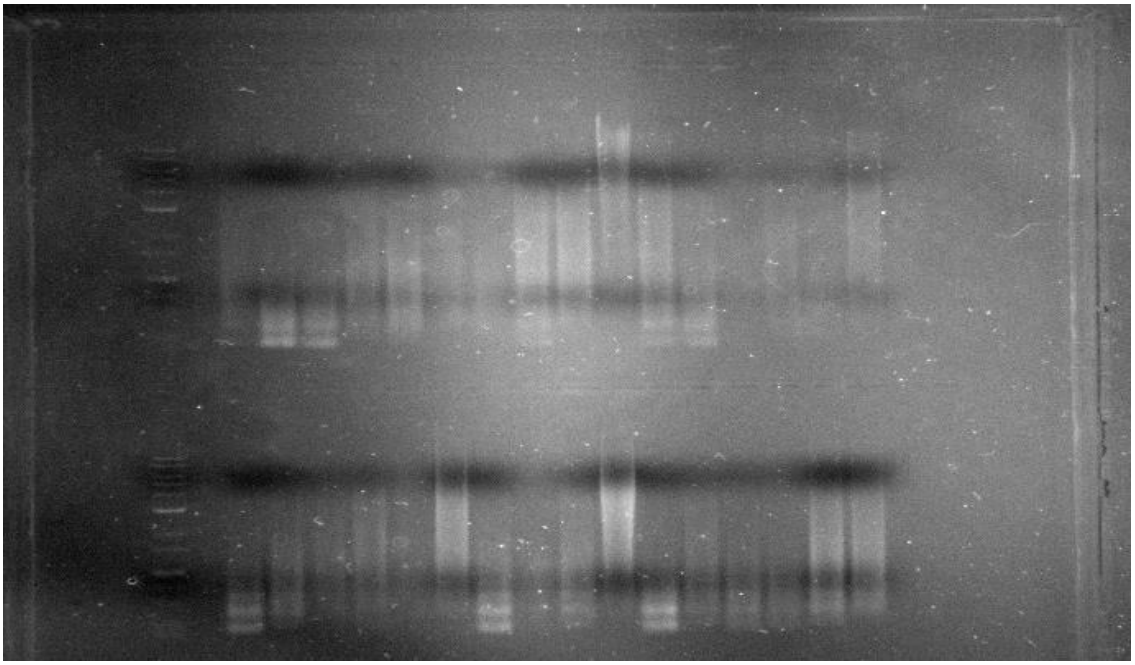
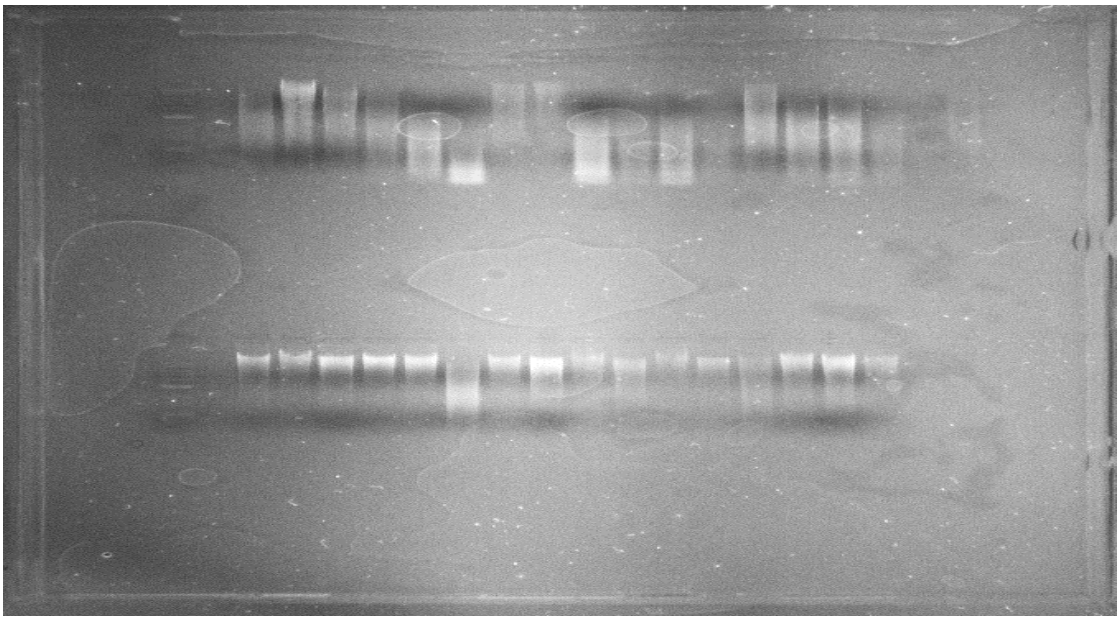


Figure A18: Gel images Part 18



Appendix B: Applications used on Biodatacatalyst

This appendix consists of 11 applications that were used in this data analysis. These packages and workflows were used in secure workspaces and all information saved on a cloud-based ecosystem. All these information can be found on the Biocatalyst and forms part of the documentation provided for each of the applications. This section gives the name of the application and a description of how it performs where needed.

Application 1: BCF tools merge and Filter

Application 2: Split by Chr | Numeral Chromosomes | BCFTools (Provided by Mr Seth Berke)

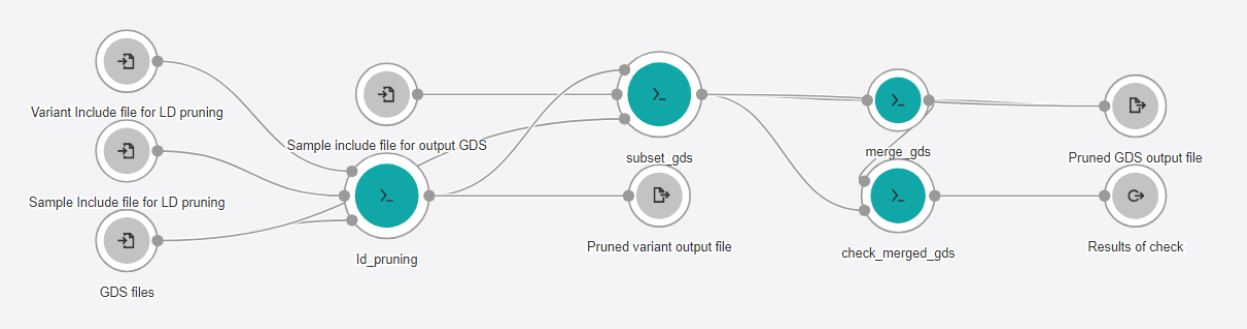
Split whole-genome vcf.gz files by chromosome. This version works when the chromosome column is bare numerals (e.g. 1, 2, 3; not preceded by "chr").

Application 3: VCF to GDS Converter

Merge multiple VCF/BCF files from non-overlapping sample sets to create one multi-sample file, and filter out any monomorphic variants.

Application 4: LD Pruning

Figure B1: LD Prunning



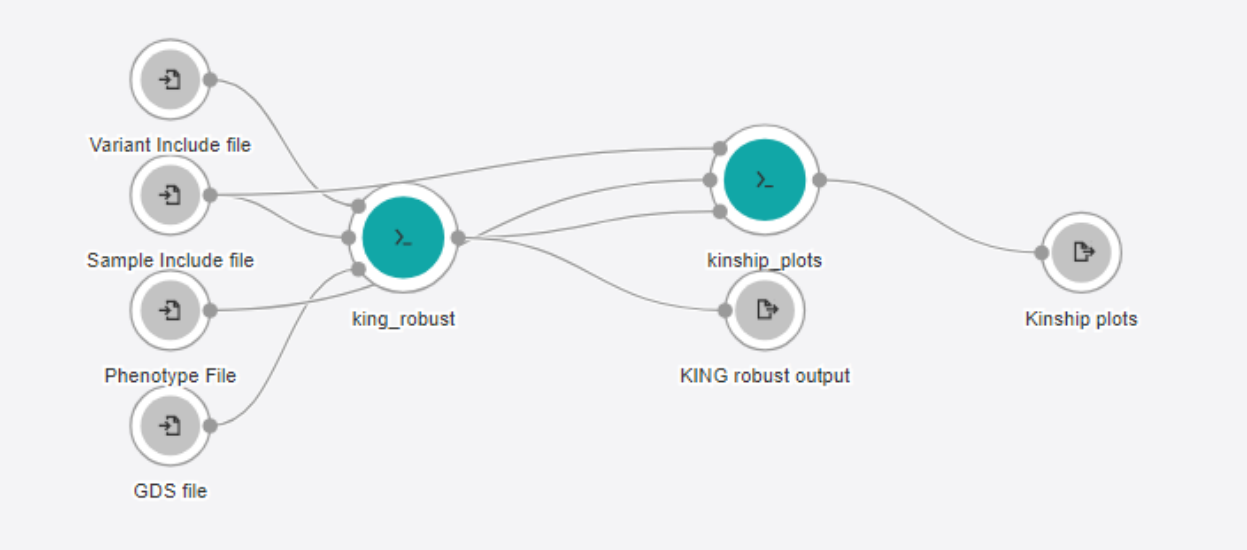
This workflow prunes variants to a subset not in linkage disequilibrium (LD) with each other and creates a new GDS file containing only the independent variants.

LD is a measure of correlation of genotypes between a pair of variants. The pruning process filters variants so that those that remain have LD measures below a given threshold. This procedure is typically used to identify a (nearly) independent subset of variants. This is often the first step in evaluating relatedness and population structure to avoid having results driven by clusters of variants in high LD regions of the genome.

Recommended usage is to provide separate GDS files for each chromosome, in which case the workflow will scatter over the GDS files. Each file will be subset to only the pruned variants, then the subset files will be merged into a single output file with pruned variants from all chromosomes. A subsequent step verifies that the merge was performed correctly. If a single file is provided, the merge step will be skipped. In this case, variants from only one chromosome may be selected using the 'chromosome' input parameter. 'Chromosome' should not be specified when running on multiple files.

Application 5: King Robust

Figure B2: King Robust

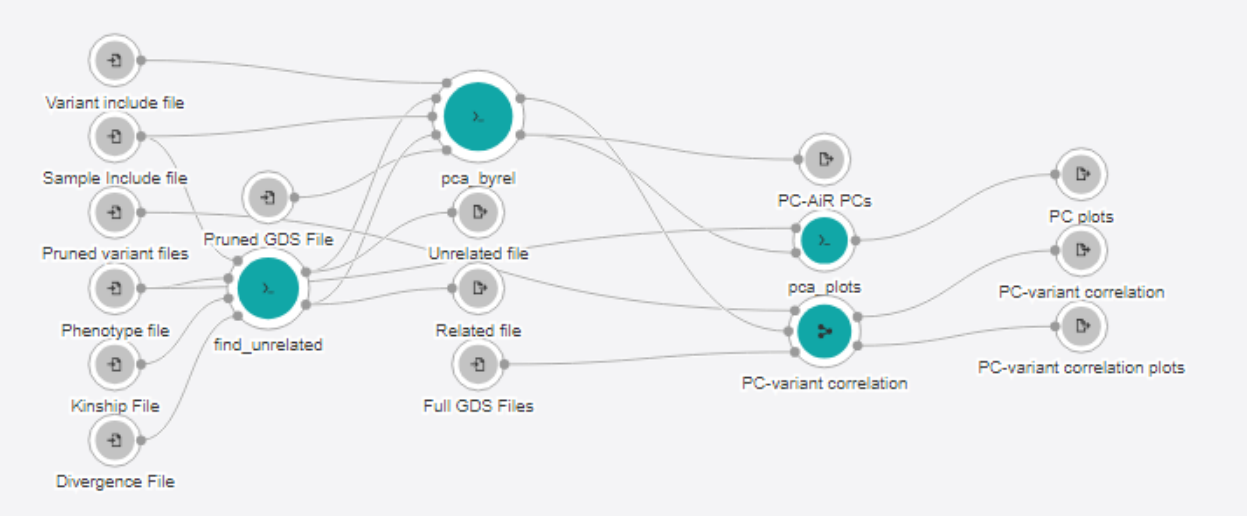


This workflow uses the KING-robust method to estimate kinship coefficients, and returns results for all pairs of samples. Due to the negative bias of these kinship estimates for samples of different ancestry, they can be used as a measure of ancestry divergence in PC-AiR.

The final output of the workflow is a plot of kinship estimates vs IBS0, which gives on overview of the amount of relatedness in the dataset. Only pairs above the specified kinship plot threshold are displayed. Since many analyses include multiple cohorts, a phenotype file may be provided to identify cohorts or groups for plotting separately. The phenotype file should be in RDATA format and contain a data.frame or AnnotatedDataFrame. Columns must include sample.id and a group variable, with the name of the group column specified as a separate argument. If these inputs are provided, additional plots will be created showing kinship separately within each group and across groups.

Application 6: PC-Air

Figure B3: PC-Air



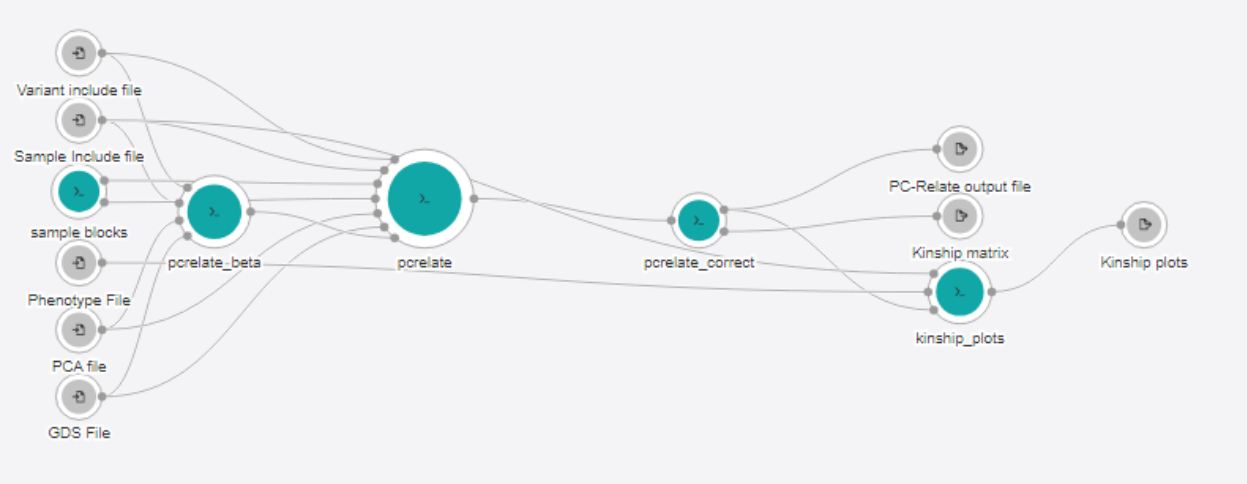
This workflow uses the PC-AiR algorithm to compute ancestry principal components (PCs) while accounting for kinship.

Step 1 uses pairwise kinship estimates to assign samples to an unrelated set that is representative of all ancestries in the sample. Step 2 performs Principal Component Analysis (PCA) on the unrelated set, then projects relatives onto the resulting set of PCs. Step 3 plots the PCs, optionally color-coding by a grouping variable. Step 4 (optional) calculates the correlation between each PC and variants in the dataset, then plots this correlation to allow screening for PCs that are driven by particular genomic regions.

The workflow has several outputs; 1) RData files with vectors of unrelated and related sample ids, 2) RData file with a 'pcair' object containing PCs for all samples, 3) plots showing PCs, color-coded by group if a phenotype file was provided, 4) (optionally) GDS files containing PC-variant correlation values, 5) (optionally) plots showing PC-variant correlation.

Application 7: PC-Relate

Figure B4: PC-Relate



This workflow estimates kinship and IBD sharing probabilities between all pairs of samples using the PC-Relate method, which accounts for population structure by conditioning on ancestry PCs.

Recommended usage is to provide the pruned GDS file from the LD pruning workflow as input. Additional variant filtering may be done by providing a file of variant.id to include. Note that the variant ids in the LD pruned GDS file may be different than the ids in the per-chromosome GDS files used as input to the LD pruning workflow.

The PCA input should be the output of the PC-AiR workflow, a 'pcair' object stored in an RData file. It is recommended to view PC plots prior to selecting the number of PCs to include in the analysis; as only the number of PCs which are informative about distant ancestry should be included.

For large sample sizes, the workflow may be parallelized by setting the number of sample blocks to be > 1. Pairwise kinship for each combination of sample blocks is then calculated in a separate job, and results are combined in the subsequent step. The variant block size controls how many variants are read from the GDS file at one time. Adjusting both sample and variant blocks controls the size of the matrix used for computation in each iteration of the algorithm, and may be tuned for maximum performance.

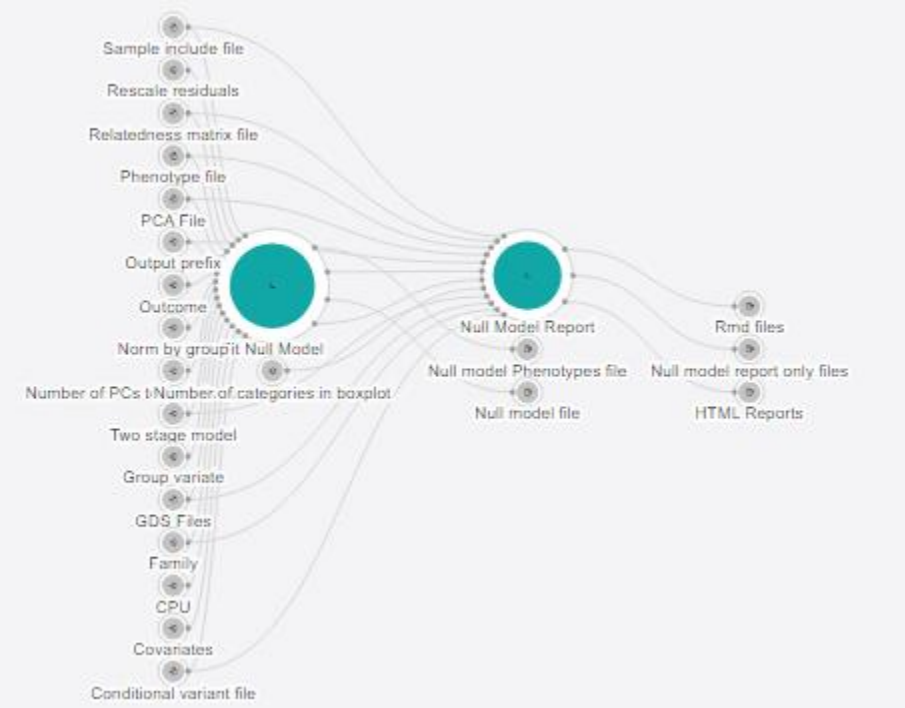
The output of the workflow is a 'pcrelate' object with data frames containing pairwise kinship estimates ('kinBtwn') as well as 'inbreeding' coefficients ('kinSelf'). Kinship estimates are also returned as a block-diagonal Matrix object in R, with values for pairs outside of family blocks set to zero. The threshold for sparsity may be set by the user. This format represents a huge savings in storage and computation time for subsequent analyses over a dense matrix.

The final output of the workflow is a plot of kinship estimates vs k0, which gives on overview of the amount of relatedness in the dataset. Only pairs above the specified kinship plot threshold are displayed. Since many analyses include multiple cohorts, a phenotype file may be provided to identify cohorts or groups for plotting separately. The phenotype file should be in RDATA format and contain a data.frame or AnnotatedDataFrame. Columns must include sample.id and a group variable, with the name of the group column specified as a separate argument. If these inputs are provided, additional plots will be created showing kinship separately within each group and across groups.

Setting 'ibd_probs' to False with skip computing the IBD probabilities k0 and k1. This saves some computation time, but the kinship plots are not generated.

Application 8: GENESIS Null Model

Figure B5: GENESIS Null Model



Null Model workflow fits the regression or mixed effects model under the null hypothesis of no genotype effects. i.e., The outcome variable is regressed on the specified fixed effect covariates and random effects. The output of this null model is then used in the association tests.

Quantitative and binary outcomes are both supported. Set parameter **family** to gaussian, binomial or poisson depending on the outcome type. Fixed effect covariates from the **Phenotype file** are specified using the **Covariates** parameter, and ancestry principal components can be included as fixed effects using the **PCA Files** and **Number of PCs to include as covariates** parameters. A kinship matrix (KM) or genetic relationship matrix (GRM) can be provided using the **Relatedness matrix file** parameter to account for genetic similarity among samples as a random effect.

When no **Relatedness matrix file** is provided, standard linear regression is used if the parameter **family** is set to gaussian, logistic regression is used if the parameter **family** is set to binomial and poisson regression is used in case when **family** is set to poisson. When **Relatedness**

matrix file is provided, a linear mixed model (LMM) is fit if **family** is set to gaussian, or a generalized linear mixed model (GLMM) is fit using the GMMAT method if **family** is set to binomial or poisson. For either the LMM or GLMM, the AI-REML algorithm is used to estimate the variance components and fixed effects parameters.

When samples come from multiple groups (e.g., study or ancestry group), it is common to observe different variances by group for quantitative traits. It is recommended to allow the null model to fit heterogeneous residual variances by group using the parameter group_var. This often provides better control of false positives in subsequent association testing. Note that this only applies when **family** is set to gaussian.

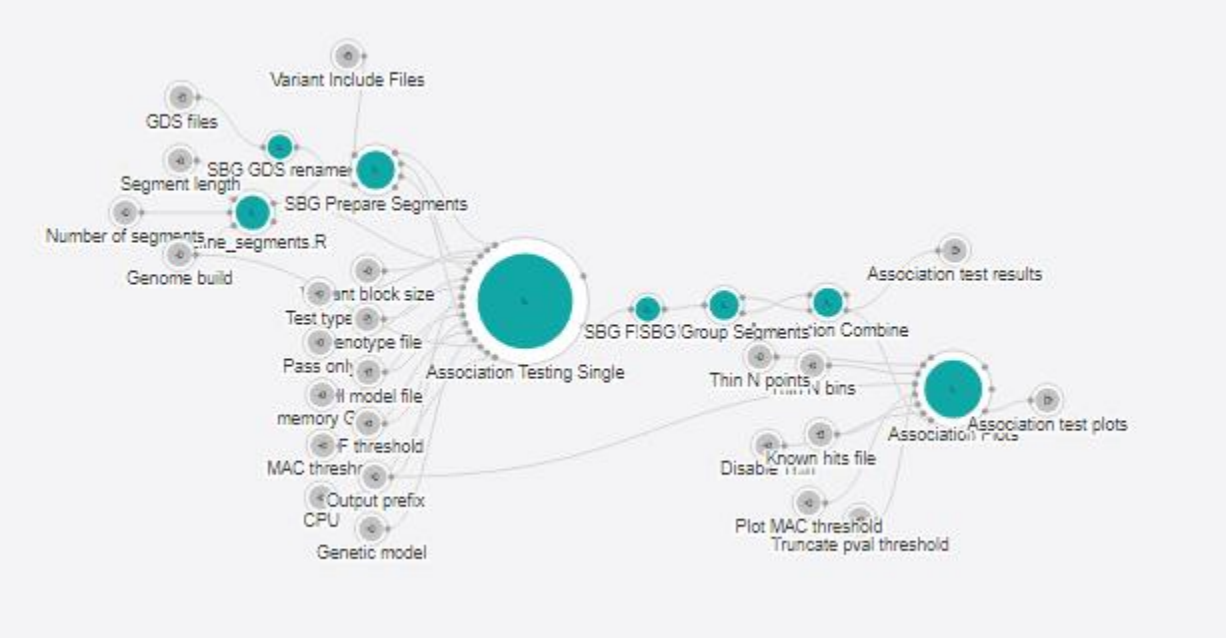
Rank-based inverse Normal transformation is supported for quantitative outcomes via the inverse_normal parameter. This parameter is TRUE by default. When **inverse normal** parameter is set to TRUE, (1) the null model is fit using the original outcome values, (2) the marginal residuals are rank-based inverse Normal transformed, and (3) the null model is fit again using the transformed residuals as the outcome; fixed effects and random effects are included both times the null model is fit. It has been shown that this fully adjusted two-stage procedure provides better false positive control and power in association tests than simply inverse Normalizing the outcome variable prior to analysis (**reference**).

This workflow utilizes the *fitNullModel* function from the GENESIS software.

Workflow consists of two steps. First step fits the null model, and the second one generates reports based on data. Reports are available both in RMD and HTML format. If **inverse normal** is TRUE, reports are generated for the model both before and after the transformation. Reports contain the following information: Config info, phenotype distributions, covariate effect size estimates, marginal residuals, adjusted phenotype values and session information.

Application 9: GENESIS Single Variant Association Test

Figure B6: GENESIS Single Variant Association Test



Single Variant workflow runs single-variant association tests. It consists of several steps. Define Segments divides genome into segments, either by a number of segments, or by a segment length. Note that number of segments refers to whole genome, not a number of segments per chromosome. Association test is then performed for each segment in parallel, before combining results on chromosome level. Final step produces QQ and Manhattan plots.

This workflow uses the output from a model fit using the null model workflow to perform tests for all variants individually. The reported effect estimate is for the alternate allele, and multiple alternate alleles for a single variant are tested separately.

When testing a binary outcome, the saddlepoint approximation (SPA) for p-values [1][2] can be used by specifying **Test type** = 'score.spa'; this is generally recommended. SPA will provide better calibrated p-values, particularly for rarer variants in samples with case-control imbalance.

When testing a binary outcome, the BinomiRare test is also available[3]. This is a "carriers only" exact test that compares the observed number of variant carriers who are cases to the expected

number based on the probability of being a case under the null hypothesis of no association between outcome and variant. This test may be useful when testing association of very rare variants with rare outcomes.

When using the test_type ‘score‘ or ‘score.spa‘, the fast approximation to the score standard error (SE) implemented by Zhou et al. (2018) in their SAIGE software [5] is available by using a null_model_file prepared with the Update Null Model for Fast Score Test workflow. This approximation may be much faster than computing the true score SE in large samples, as it replaces the full covariance matrix in the calculation with the product of a diagonal matrix and a scalar correction factor (se.correction) in the updated null model output.

If your genotype data has sporadic missing values, they are mean imputed using the allele frequency observed in the sample.

On the X chromosome, males have genotype values coded as 0/2 (females as 0/1/2).

This workflow utilizes the *assocTestSingle* function from the GENESIS software [4]

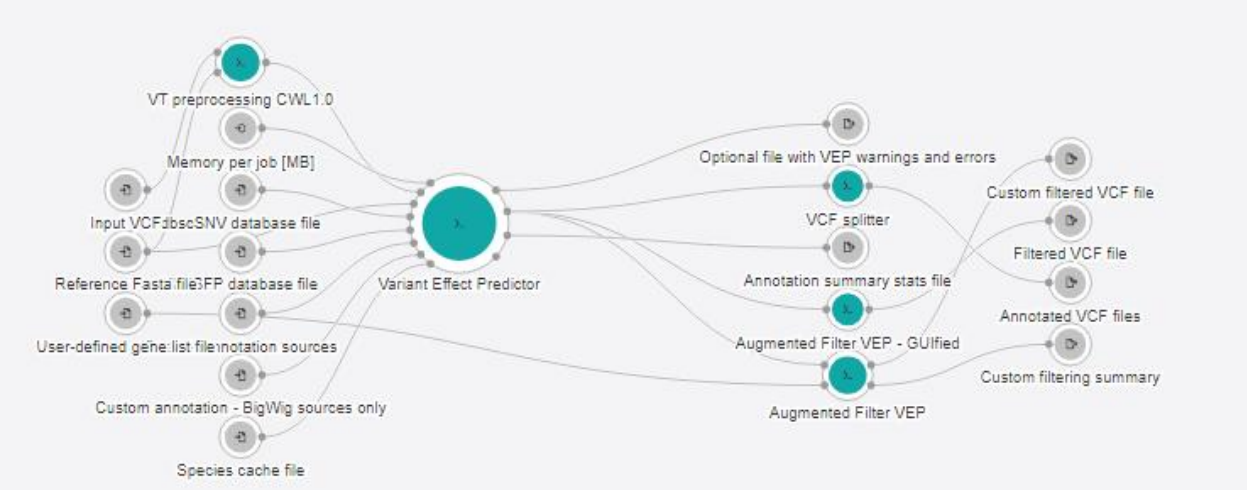
Application 10: GENESIS Association Single Plotting

The UW-GAC GENESIS Association Result Plotting standalone app creates Manhattan and QQ plots from GENESIS association test results with additional filtering and stratification options available. This app is run automatically with default options set by the GENESIS Association Testing Workflows. Users can fine-tune the Manhattan and QQ plots by running this app separately, after one of the association testing workflows. The available options are:

- Create QQ plots by chromosome.
- Include a user-specified subset of the results in the plots.
- Filter results to only those with MAC or MAF greater than a specified threshold.
- Calculate genomic inflation lambda at various quantiles.
- Specify the significance type and level.
- Create QQ plots stratified by MAC or MAF.
- Specify a maximum p-value to display on the plots.

Application 11: VEP annotation workflow 109.3

Figure B7: VEP annotation workflow 109.3



The **VEP annotation workflow 109.3** is used for preprocessing, annotating, and filtering VCF files. VCF or VCF.GZ files can be processed using this workflow. The first step in this pipeline is an optional VCF normalization and decomposition performed using the **vt toolkit**. To minimize processing time, the input file is streamed to **vt normalize** and **vt decompose**, which are invoked as piped processes. Preprocessing with **vt** requires a FASTA file to be provided with the input VCF. The final step of preprocessing is bgzip compression and tabix indexing, which is set to occur even if the **vt preprocessing** of the input file is skipped. Input files are sorted (**vt sort**, sorting mode local or full, depending on the absence/presence of contigs in the input VCF header) before preprocessing and **vt uniq** is used to drop any potentially duplicated variant entries after decomposition.

The input file is annotated using **Variant Effect Predictor**. Annotations are added from VEP cache, plugins, or custom annotation sources (VCF, BED, GFF, GTF, BW, etc.). If you are using custom annotation sources, all associated parameters should be provided separately in the order of the provided input files. The third step of the pipeline is the optional filtering of the annotated VCF file. This step uses the filter_vep script from the ensembl-vep toolkit (**Filter VEP**), which has been wrapped to allow pass-through (if no filters are selected) and with a number of convenient filters alongside the standard filter_vep syntax.

Appendix C: R codes for all the phenotypes

This appendix provides the R codes or scripts that were written and used in R studio within the Biodata Catalyst. Files that are used in the applications provided in **Appendix B** have to be processed and worked with before it can be used since it requires certain format and information to be present. Also, some files that results from the applications can be looked in details using R studio and used according to our needs.

```
#Load the file Phenotype_9th_May_df.tsv from project file

library(dplyr)
library(tidyverse)
library(Biobase)
library(readr)

pheno_9June_df <-data.frame(June_phenotype_file)

csv_file_path <- "/sbgenomics/output-files/pheno_9June_df.csv"
tsv_file_path <- "/sbgenomics/output-files/pheno_9June_df.tsv"

write.table(pheno_9June_df, file = csv_file_path, sep = "\t", quote = FALSE, row.names =
FALSE)
write.table(pheno_9June_df, file = tsv_file_path, sep = "\t", quote = FALSE, row.names =
FALSE)

pheno_9June_df <- "/sbgenomics/project-files/pheno_9June_df.tsv"

phen <- read.table(file =pheno_9June_df, header = TRUE, sep = "\t", as.is = TRUE)
head(phen)

# Define the metadata data frame with labelDescription column
# Define the metadata data frame
metadata <- data.frame(
  labelDescription = c(
    "sample.id from Illumina",
    "subject.id from DXA",
    "gender of subject",
    "height in cm",
    "weight in kg",
    "age in years",
    "arm fat mass in g",
    "legs fat mass in g",
    "trunk fat mass in g",
```

```
    "android fat mass in g",
    "gynoid fat mass in g",
    "total fat mass in g"),
  row.names = colnames(phen))

# Print the metadata data frame
print(metadata)

# Create AnnotatedDataFrame
annot <- AnnotatedDataFrame(data = phen, varMetadata = metadata)
annot

# Save AnnotatedDataFrame
save(annot, file = "/sbgenomics/output-files/pheno_9June_df.RData")


###### reading the pca file from pc air########
library(Biobase)
#the name of the file is pca
load("/sbgenomics/project-files/_2_tmplr_pca.RData")
# Accessing and viewing each component
vectors_matrix <- pca$vectors
# Convert vectors_matrix to a dataframe for better viewing
vectors_df <- as.data.frame(vectors_matrix)
# View the dataframes or vectors
View(vectors_df)
vectors_df <- data.frame(vectors_df)
# Assuming 'vectors_df' is your PCA dataframe
colnames(vectors_df) <- paste0("PC", 1:ncol(vectors_df))
# Store the row names
sample_ids <- rownames(vectors_df)
# Remove row names
rownames(vectors_df) <- NULL
# Add sample ids as the first column
vectors_df <- cbind(sample.id = sample_ids, vectors_df)
vectors_df2<-data.frame(vectors_df)
save(vectors_df2, file="/sbgenomics/output-files/pcs_9June.RData")


####Creating the phenotype file with all PCs for the Null Model package
library(Biobase)

###Load June_phenotype_file.csv from project-files
###Load pcs_9June.RData from project-files

###Note:the pcs_9June.RData is uploaded as vectors_df
```

```
##merge the two files by sample.id
library(dplyr)

# Read the CSV files
# Merge the data frames by 'sample.id' column
merged_data <- merge(June_phenotype_file, vectors_df2, by="sample.id")
merged_data_df<-data.frame(merged_data)
# Write the merged data to a new CSV file
save(merged_data_df, file="/sbgenomics/output-files/finalpcs_9June.RData")
save(merged_data_df, file="/sbgenomics/output-files/finalpcs_9June.csv")
```

Debugging: there is a duplicate row

```
# View rows with specific sample IDs
specific_rows <- merged_data_df[merged_data_df$sample.id %in% c('206397600035_R06C01',
'206439950008_R01C01'), ]

# Print the specific rows
print(specific_rows)
sample.id subject.id Gender   Height   Weight Age Arms_Fat_Mass Legs_Fat_Mass
Trunk_Fat_Mass Android_Fat_Mass
183 206397600035_R06C01        564 Female 164.4 cm  84.8 kg  33       3805.47       13103.01
25958.14          4608.32
184 206397600035_R06C01        564   Male 188.5 cm  86.5 kg  31       1986.47       8027.98
12522.29          2112.98
313 206439950008_R01C01         21   Male 184.7 cm  85.0 kg  41       1283.98       5811.86
6851.89          1147.28
314 206439950008_R01C01         21 Female 159.0 cm 103.1 kg  44       2854.88       17331.03
31240.27          5191.83
Gynoid_Fat_Mass Total_Fat_Mass         PC1          PC2          PC3         PC4          PC5
PC6         PC7
183         7095.52       43881.46 -0.01981181 -0.0003768725 -0.006557861 0.001945132
0.001026755 -0.012503169 -0.02561023
184         3740.93       23437.36 -0.01981181 -0.0003768725 -0.006557861 0.001945132
0.001026755 -0.012503169 -0.02561023
313         2529.02       15035.60 -0.01976405  0.0028068394 -0.002362126 0.001831231 -
0.005522877 -0.004270973 -0.01146022
314         8394.47       52371.85 -0.01976405  0.0028068394 -0.002362126 0.001831231 -
0.005522877 -0.004270973 -0.01146022
PC8         PC9         PC10
183 -0.001290657 0.009511884  0.007837340
184 -0.001290657 0.009511884  0.007837340
313 -0.009366425 0.004875478 -0.007738365
```

```
314 -0.009366425 0.004875478 -0.007738365

# View rows with specific sample IDs
specific_rows2 <- June_phenotype_file[June_phenotype_file$sample.id %in%
c('206397600035_R06C01', '206439950008_R01C01'), ]
print(specific_rows2)

ample.id subject.id Gender   Height   Weight Age Arms_Fat_Mass Legs_Fat_Mass
Trunk_Fat_Mass Android_Fat_Mass
8   206439950008_R01C01         21 Female 159.0 cm 103.1 kg  44       2854.88      17331.03
31240.27         5191.83
9   206439950008_R01C01         21   Male 184.7 cm  85.0 kg  41       1283.98       5811.86
6851.89         1147.28
240 206397600035_R06C01        564 Female 164.4 cm  84.8 kg  33       3805.47      13103.01
25958.14         4608.32
241 206397600035_R06C01        564   Male 188.5 cm  86.5 kg  31       1986.47       8027.98
12522.29         2112.98
Gynoid_Fat_Mass Total_Fat_Mass
8          8394.47       52371.85
9          2529.02       15035.60
240         7095.52       43881.46
241         3740.93       23437.36

##### removing all duplicated rows
#remove line 183, 184, 313, 314 from merged_data_df

merged_data_mod <- merged_data_df[-c(183, 184, 313, 314), ]
rownames(merged_data_mod) <- NULL

save(merged_data_mod, file="/sbgenomics/output-files/pcs_10June.RData")

## Debugging again: Error suggests suggests that the design matrix used in model is not full
rank,
#indicating multicollinearity among covariates. To resolve this, removing cm in height column
and kg in weight column
# Assuming merged_data_mod is your data frame
# Convert the columns to character type if they are not already
merged_data_mod$Height <- as.character(merged_data_mod$Height)
merged_data_mod$Weight <- as.character(merged_data_mod$Weight)

# Remove "cm" from the height column and "kg" from the weight column
merged_data_mod$Height <- gsub("cm", "", merged_data_mod$Height)
merged_data_mod$Weight <- gsub("kg", "", merged_data_mod$Weight)

# Convert the columns back to numeric if needed
merged_data_mod$Height <- as.numeric(merged_data_mod$Height)
```

```
merged_data_mod$Weight <- as.numeric(merged_data_mod$Weight)

# Check the result
head(merged_data_mod)
#merged_data_mod has 395 observation of 22 variables (12 variables plus 10 PCs)
save(merged_data_mod, file="/sbgenomics/output-files/pcs_11June.RData")
write.csv(merged_data_mod, file="/sbgenomics/output-files/pcs_11June.csv")


library(data.table)
library(dplyr)

# Define a function to load and filter each chromosome data
load_and_filter <- function(chr_num) {
  file_path <- paste0('/sbgenomics/project-files/tmplr_FAT_Mass_single_chr', chr_num, '.RData')
  loaded_object <- load(file_path)
  dt <- data.table(get(loaded_object))
  dt <- dt[MAC >= 100]
  dt$chr <- chr_num
  return(dt)
}

# Load, filter, and process all chromosomes into a list
assoc_list <- lapply(1:22, load_and_filter)

# Combine all data tables into one
all_data <- rbindlist(assoc_list)

# Get the lowest 6 p-values for each chromosome
lowest_pvals_per_chr <- all_data %>%
  group_by(chr) %>%
  arrange(Score.pval) %>%
  slice_head(n = 6) %>%
  ungroup()

# Get the lowest 30 p-values across all chromosomes
lowest_30_pvals <- all_data %>%
  arrange(Score.pval) %>%
  slice_head(n = 30)

# Print the resulting tables
print("Lowest 6 p-values per chromosome:")
print(lowest_pvals_per_chr)

print("\nLowest 30 p-values overall:")
print(lowest_30_pvals)
```

```
# Write the resulting tables to CSV files
write.csv(lowest_pvals_per_chr, file="/sbgenomics/output-files/lowest_pvals_per_chr_df.csv",
row.names = FALSE)
write.csv(lowest_30_pvals, file="/sbgenomics/output-files/lowest_30_pvals_df.csv", row.names
= FALSE
```

R codes - Android_Fat_Mass

```
#Start at PC Air application. Same file "pheno_9June_df.RData" is used but group is
Android_Fat_Mass

###### Creating the pca file for NUll Model########
library(Biobase)
#the name of the file is pca
load("/sbgenomics/project-files/_4_tmplr_pca.RData")

# Accessing and viewing each component
vectors_matrix4 <- pca$vectors

# Convert vectors_matrix to a dataframe for better viewing
vectors_df4 <- as.data.frame(vectors_matrix4)

# View the dataframes or vectors
View(vectors_df4)
# Assuming 'vectors_df' is your PCA dataframe
colnames(vectors_df4) <- paste0("PC", 1:ncol(vectors_df4))

# Store the row names
sample_ids <- rownames(vectors_df4)

# Remove row names
rownames(vectors_df4) <- NULL

# Add sample ids as the first column
vectors_df_Android <- cbind(sample.id = sample_ids, vectors_df4)
vectors_df_Android_2<-data.frame(vectors_df_Android)
save(vectors_df_Android_2, file="/sbgenomics/output-files/Android_Fat_Mass.RData")

library(Biobase)

###Load phenotypefile_nodup_12June.csv from project-files
###Load Gynoid_Fat_Mass.RData from project-files

##merge the two files by sample.id
```

```
library(dplyr)
load("/sbgenomics/project-files/phenotypefile_nodup_12June.RData")

# Read the CSV files
# Merge the data frames by 'sample.id' column
merged_data_Android <- merge(Final_pheno_nodup_12thJune, vectors_df_Android_2,
by="sample.id")
merged_data_Android_df<-data.frame(merged_data_Android)
# Write the merged data to a new CSV file
save(merged_data_Android_df, file="/sbgenomics/output-files/Androidpcs.RData")
write.csv(merged_data_Android_df, file="/sbgenomics/output-files/Androidpcs.csv")
```

R codes - Arms_Fat_Mass

```
#Start at PC Air application. Same file "pheno_9June_df.RData" is used but group is
Arms_Fat_Mass

###### Creating the pca file for NUll Model########
library(Biobase)
#the name of the file is pca
load("/sbgenomics/project-files/_7_tmplr_pca.RData")

# Accessing and viewing each component
vectors_matrix7 <- pca$vectors


# Convert vectors_matrix to a dataframe for better viewing
vectors_df7 <- as.data.frame(vectors_matrix7)

# View the dataframes or vectors
View(vectors_df7)
# Assuming 'vectors_df' is your PCA dataframe
colnames(vectors_df7) <- paste0("PC", 1:ncol(vectors_df7))

# Store the row names
sample_ids <- rownames(vectors_df7)

# Remove row names
rownames(vectors_df7) <- NULL

# Add sample ids as the first column
vectors_df_Arms <- cbind(sample.id = sample_ids, vectors_df7)
vectors_df_Arms_2<-data.frame(vectors_df_Arms)
save(vectors_df_Arms_2, file="/sbgenomics/output-files/Arms_Fat_Mass.RData")

library(Biobase)
```

```
###Load phenotypefile_nodup_12June.csv from project-files
###Load Gynoid_Fat_Mass.RData from project-files

##merge the two files by sample.id
library(dplyr)
load("/sbgenomics/project-files/phenotypefile_nodup_12June.RData")

# Read the CSV files
# Merge the data frames by 'sample.id' column
merged_data_Arms <- merge(Final_pheno_nodup_12thJune, vectors_df_Arms_2,
by="sample.id")
merged_data_Arms_df<-data.frame(merged_data_Arms)
# Write the merged data to a new CSV file
save(merged_data_Arms_df, file="/sbgenomics/output-files/Armspcs.RData")
write.csv(merged_data_Arms_df, file="/sbgenomics/output-files/Armspcs.csv")
```

R codes – Gynoid_Fat_Mass

```
#Start at PC Air application. Same file "pheno_9June_df.RData" is used but group is
Gynoid_Fat_Mass

###### Creating the pca file for NUll Model########
library(Biobase)
#the name of the file is pca
load("/sbgenomics/project-files/_3_tmplr_pca.RData")

# Accessing and viewing each component
vectors_matrix3 <- pca$vectors


# Convert vectors_matrix to a dataframe for better viewing
vectors_df3 <- as.data.frame(vectors_matrix3)

# View the dataframes or vectors
View(vectors_df3)
# Assuming 'vectors_df' is your PCA dataframe
colnames(vectors_df3) <- paste0("PC", 1:ncol(vectors_df3))

# Store the row names
sample_ids <- rownames(vectors_df3)

# Remove row names
rownames(vectors_df3) <- NULL

# Add sample ids as the first column
vectors_df_gynoid <- cbind(sample.id = sample_ids, vectors_df3)
vectors_df_gynoid_2<-data.frame(vectors_df_gynoid)
```

```
save(vectors_df_gynoid_2, file="/sbgenomics/output-files/Gynoid_Fat_Mass.RData")

library(Biobase)

###Load phenotypefile_nodup_12June.csv from project-files
###Load Gynoid_Fat_Mass.RData from project-files

##merge the two files by sample.id
library(dplyr)
load("/sbgenomics/project-files/phenotypefile_nodup_12June.RData")

# Read the CSV files
# Merge the data frames by 'sample.id' column
merged_data_Gynoid <- merge(Final_pheno_nodup_12thJune, vectors_df_gynoid_2,
by="sample.id")
merged_data_Gynoid_df<-data.frame(merged_data_Gynoid)
# Write the merged data to a new CSV file
save(merged_data_Gynoid_df, file="/sbgenomics/output-files/gynoidpcs.RData")
write.csv(merged_data_Gynoid_df, file="/sbgenomics/output-files/gynoidpcs.csv")

R codes - Trunk_Fat_Mass

#Start at PC Air application. Same file "pheno_9June_df.RData" is used but group is
Trunk_Fat_Mass

###### Creating the pca file for Null Model########
library(Biobase)
#the name of the file is pca
load("/sbgenomics/project-files/_5_tmplr_pca.RData")

# Accessing and viewing each component
vectors_matrix5 <- pca$vectors


# Convert vectors_matrix to a dataframe for better viewing
vectors_df5 <- as.data.frame(vectors_matrix5)

# View the dataframes or vectors
View(vectors_df5)
# Assuming 'vectors_df' is your PCA dataframe
colnames(vectors_df5) <- paste0("PC", 1:ncol(vectors_df5))

# Store the row names
sample_ids <- rownames(vectors_df5)

# Remove row names
```

```
rownames(vectors_df5) <- NULL

# Add sample ids as the first column
vectors_df_Trunk <- cbind(sample.id = sample_ids, vectors_df5)
vectors_df_Trunk_2<-data.frame(vectors_df_Trunk)
save(vectors_df_Trunk_2, file="/sbgenomics/output-files/Trunk_Fat_Mass.RData")

library(Biobase)

###Load phenotypefile_nodup_12June.csv from project-files
###Load Gynoid_Fat_Mass.RData from project-files

##merge the two files by sample.id
library(dplyr)
load("/sbgenomics/project-files/phenotypefile_nodup_12June.RData")

# Read the CSV files
# Merge the data frames by 'sample.id' column
merged_data_Trunk <- merge(Final_pheno_nodup_12thJune, vectors_df_Trunk_2,
by="sample.id")
merged_data_Trunk_df<-data.frame(merged_data_Trunk)
# Write the merged data to a new CSV file
save(merged_data_Trunk_df, file="/sbgenomics/output-files/Trunkpcs.RData")
write.csv(merged_data_Trunk_df, file="/sbgenomics/output-files/Trunkpcs.csv")
```

Appendix D: R codes for annotation

After the VEP annotation package was used, we had to use R studio to analyze the files generated. There were many fields and we had to choose the ones relevant for this research, for example, gene symbol and SNP ID.

```
install.packages("dplyr")
install.packages("tidyr")
install.packages("data.table")
install.packages("stringr")
install.packages("janitor")

if (!requireNamespace("BiocManager", quietly = TRUE)) {
  install.packages("BiocManager")
}
BiocManager::install("VariantAnnotation")


library(dplyr)
library(tidyr)
library(data.table)
library(stringr)
library(VariantAnnotation)
library(janitor)
library(data.table)

###################  Chromosome 2 ################################

library(dplyr)

# Read in the VEP output file for chromosome 2
vep_data_chr2 <- read.table("/sbgenomics/project-files/tmplr_subset_chr2.processed.vep.vcf",
sep="\t", header=TRUE, comment.char="#")

# Display the names of the first 8 columns
first_8_colnames_chr2 <- colnames(vep_data_chr2)[1:8]
print(first_8_colnames_chr2)

# Automatically make column names unique by appending a suffix
vep_data_chr2 <- vep_data_chr2 %>%
  clean_names()  # Ensures no duplicate column names

# Create the subset with the relevant columns
```

```
selected_columns_chr2 <- vep_data_chr2[, c("x2", "x10587", "rs28804817", "c", "g",
"ac_30_an_868_csq_g_regulatory_region_variant_modifier_regulatory_feature_ensr000019140
36_ctcf_binding_site_rs28804817_snv_0_1328_0_3994_0_0648_0_0218_0_0278_0_0429_0_3
994_afr_rs28804817_g_intergenic_variant_modifier_rs28804817_snv_0_1328_0_3994_0_0648
_0_0218_0_0278_0_0429_0_3994_afr_rs28804817")]

# Rename the columns
colnames(selected_columns_chr2) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")


if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

  # Filter combined_data for chromosome 2
  chr2_data <- combined_data %>%
    filter(chr == 2)

  # Extract unique position numbers from chromosome 4
  chr2_positions <- unique(chr2_data$pos)

  # Display the first few position numbers for chromosome 4
  print(head(chr2_positions))

} else {
  stop("Error: 'combined_data' is either not loaded or does not contain the required columns ('chr'
and 'pos').")
}

# Define the positions to filter
chr2_positions <- c(37554788, 214917402, 121295944, 112913967, 173785000)

# Filter the data based on these positions
filtered_data_chr2 <- selected_columns_chr2[selected_columns_chr2$POS %in%
chr2_positions, ]

# Check the filtered data
head(filtered_data_chr2)

# Ensure unique column names
colnames(filtered_data_chr2) <- make.unique(colnames(filtered_data_chr2))

# Extract CSQ field and remove "CSQ=" prefix
filtered_data_chr2 <- filtered_data_chr2 %>%
  mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%  # Extract CSQ field
  mutate(CSQ = str_remove(CSQ, "CSQ="))            # Remove "CSQ=" prefix

# Verify the CSQ field extraction
```

```
print(head(filtered_data_chr2$CSQ, 10))  # Check the first 10 rows

# Check if any CSQ values are empty or missing
print(sum(is.na(filtered_data_chr2$CSQ)))  # Count NA values in CSQ
print(sum(filtered_data_chr2$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

# Filter out rows where CSQ is missing or empty
filtered_data_chr2 <- filtered_data_chr2 %>%
  filter(!is.na(CSQ) & CSQ != "")
# Recalculate max_fields based on the cleaned data
max_fields <- filtered_data_chr2 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%  # Count delimiters and add 1
  summarise(max_fields = max(field_count)) %>%         # Find the maximum
  pull(max_fields)

# Print max_fields to verify it's a reasonable value
print(max_fields)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr2 <- filtered_data_chr2 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields), sep = "\\|", fill = "right")


# Select only the desired columns
# Use dplyr::select explicitly to avoid conflicts
desired_columns_chr2 <- dplyr::select(filtered_data_chr2, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)


# Rename the selected columns
desired_columns_chr2 <- dplyr::rename(desired_columns_chr2,
                                      Protein_coding = Field8,
                                      Variant_type = Field2,
                                      Gene_symbols = Field4,
                                      Ensembl_Gene_IDs = Field5,
                                      Variant_feature_types = Field6,
                                      rs_Ids = Field18)


# Save the results

write.csv(desired_columns_chr2, file="/sbgenomics/output-files/desired_columns_chr2.csv")
```

```
################################ chromosome 3 ################################


# Read in the VEP output file for chromosome 3
vep_data_chr3 <- read.table("/sbgenomics/project-files/tmplr_subset_chr3.processed.vep.vcf",
sep="\t", header=TRUE, comment.char="#")

# Display the names of the first 8 columns
first_8_colnames_chr3 <- colnames(vep_data_chr3)[1:8]
print(first_8_colnames_chr3)

# Automatically make column names unique by appending a suffix
vep_data_chr3 <- vep_data_chr3 %>%
  clean_names()  # Ensures no duplicate column names

# Create the subset with the relevant columns
selected_columns_chr3 <- vep_data_chr3[, c("x3", "x60197", "rs115479960", "g", "a",
"ac_45_an_870_csq_a_intergenic_variant_modifier_rs115479960_snv_0_0523_0_1059_0_049_
0_001_0_0557_0_0317_0_1059_afr")]
# Rename the columns
colnames(selected_columns_chr3) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

# If 'combined_data' exists and contains the required columns
if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

  # Filter combined_data for chromosome 3
  chr3_data <- combined_data %>%
    filter(chr == 3)

  # Extract unique position numbers from chromosome 3
  chr3_positions <- unique(chr3_data$pos)

  # Display the first few position numbers for chromosome 3
  print(head(chr3_positions))

} else {
  stop("Error: 'combined_data' is either not loaded or does not contain the required columns ('chr'
and 'pos').")
}

# Define the positions to filter
chr3_positions <- c(179389656)

# Filter the data based on these positions
filtered_data_chr3 <- selected_columns_chr3[selected_columns_chr3$POS %in%
chr3_positions, ]
```

```
# Check the filtered data
head(filtered_data_chr3)

# Ensure unique column names
colnames(filtered_data_chr3) <- make.unique(colnames(filtered_data_chr3))

# Extract CSQ field and remove "CSQ=" prefix
filtered_data_chr3 <- filtered_data_chr3 %>%
  mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%  # Extract CSQ field
  mutate(CSQ = str_remove(CSQ, "CSQ="))            # Remove "CSQ=" prefix

# Verify the CSQ field extraction
print(head(filtered_data_chr3$CSQ, 10))  # Check the first 10 rows

# Check if any CSQ values are empty or missing
print(sum(is.na(filtered_data_chr3$CSQ)))  # Count NA values in CSQ
print(sum(filtered_data_chr3$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

# Filter out rows where CSQ is missing or empty
filtered_data_chr3 <- filtered_data_chr3 %>%
  filter(!is.na(CSQ) & CSQ != "")

# Recalculate max_fields based on the cleaned data
max_fields <- filtered_data_chr3 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%  # Count delimiters and add 1
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%         # Find the maximum
  pull(max_fields)

# Print max_fields to verify it's a reasonable value
print(max_fields)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr3 <- filtered_data_chr3 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields), sep = "\\|", fill = "right")

# Select only the desired columns
# Use dplyr::select explicitly to avoid conflicts
desired_columns_chr3 <- dplyr::select(filtered_data_chr3, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

# Rename the selected columns
desired_columns_chr3 <- dplyr::rename(desired_columns_chr3,
                                      Protein_coding = Field8,
                                      Variant_type = Field2,
                                      Gene_symbols = Field4,
```

```
                    Ensembl_Gene_IDs = Field5,
                    Variant_feature_types = Field6,
                    rs_Ids = Field18)

# Save the results
write.csv(desired_columns_chr3, file="/sbgenomics/output-files/desired_columns_chr3.csv")

#########################chromosomes 4 ############################

# Read in the VEP output file for chromosome 4
vep_data_chr4 <- read.table("/sbgenomics/project-files/tmplr_subset_chr4.processed.vep.vcf",
sep="\t", header=TRUE, comment.char="#")

# Display the names of the first 8 columns
first_8_colnames_chr4 <- colnames(vep_data_chr4)[1:8]
print(first_8_colnames_chr4)

# Automatically make column names unique by appending a suffix
vep_data_chr4 <- vep_data_chr4 %>%
  clean_names()  # Ensures no duplicate column names

# Create the subset with the relevant columns
selected_columns_chr4 <- vep_data_chr4[, c("x4", "x27851", "jhu_4_27850", "g", "a",
"ac_870_an_870_csq_a_regulatory_region_variant_modifier_regulatory_feature_ensr000016727
37_open_chromatin_region_rs1581282342_rs4690167_snv_rs4690167_a_intergenic_variant_m
odifier_rs1581282342_rs4690167_snv_rs4690167")]

# Rename the columns
colnames(selected_columns_chr4) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

# If 'combined_data' exists and contains the required columns
if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

  # Filter combined_data for chromosome 4
  chr4_data <- combined_data %>%
    filter(chr == 4)

  # Extract unique position numbers from chromosome 4
  chr4_positions <- unique(chr4_data$pos)

  # Display the first few position numbers for chromosome 4
  print(head(chr4_positions))

} else {
  stop("Error: 'combined_data' is either not loaded or does not contain the required columns ('chr'
and 'pos').")
```

```
}

# Define the positions to filter
chr4_positions <- c(42551362, 29133262)  # Update with relevant positions for chromosome 4

# Filter the data based on these positions
filtered_data_chr4 <- selected_columns_chr4[selected_columns_chr4$POS %in%
chr4_positions, ]

# Check the filtered data
head(filtered_data_chr4)

# Ensure unique column names
colnames(filtered_data_chr4) <- make.unique(colnames(filtered_data_chr4))

# Extract CSQ field and remove "CSQ=" prefix
filtered_data_chr4 <- filtered_data_chr4 %>%
  mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%  # Extract CSQ field
  mutate(CSQ = str_remove(CSQ, "CSQ="))            # Remove "CSQ=" prefix

# Verify the CSQ field extraction
print(head(filtered_data_chr4$CSQ, 10))  # Check the first 10 rows

# Check if any CSQ values are empty or missing
print(sum(is.na(filtered_data_chr4$CSQ)))  # Count NA values in CSQ
print(sum(filtered_data_chr4$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

# Filter out rows where CSQ is missing or empty
filtered_data_chr4 <- filtered_data_chr4 %>%
  filter(!is.na(CSQ) & CSQ != "")

# Recalculate max_fields based on the cleaned data
max_fields <- filtered_data_chr4 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%  # Count delimiters and add 1
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%         # Find the maximum
  pull(max_fields)

# Print max_fields to verify it's a reasonable value
print(max_fields)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr4 <- filtered_data_chr4 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields), sep = "\\|", fill = "right")

# Select only the desired columns
# Use dplyr::select explicitly to avoid conflicts
```

```
desired_columns_chr4 <- dplyr::select(filtered_data_chr4, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

# Rename the selected columns
desired_columns_chr4 <- dplyr::rename(desired_columns_chr4,
                                      Protein_coding = Field8,
                                      Variant_type = Field2,
                                      Gene_symbols = Field4,
                                      Ensembl_Gene_IDs = Field5,
                                      Variant_feature_types = Field6,
                                      rs_Ids = Field18)


# Save the results
write.csv(desired_columns_chr4, file="/sbgenomics/output-files/desired_columns_chr4.csv")

################################## chromosome 5 ##################################


# Read in the VEP output file for chromosome 5
vep_data_chr5 <- read.table("/sbgenomics/project-files/tmplr_subset_chr5.processed.vep.vcf",
sep="\t", header=TRUE, comment.char="#")

# Display the names of the first 8 columns
first_8_colnames_chr5 <- colnames(vep_data_chr5)[1:8]
print(first_8_colnames_chr5)

# Automatically make column names unique by appending a suffix
vep_data_chr5 <- vep_data_chr5 %>%
  clean_names()  # Ensures no duplicate column names

# Create the subset with the relevant columns for chromosome 5
selected_columns_chr5 <- vep_data_chr5[, c("x5", "x13301", "rs76192746", "g", "t",
"ac_3_an_870_csq_t_regulatory_region_variant_modifier_regulatory_feature_ensr00000402131
_ctcf_binding_site_rs76192746_snv_0_0389_0_1377_0_0144_0_001_0_002_0_0_1377_afr_rs7
6192746_t_intergenic_variant_modifier_rs76192746_snv_0_0389_0_1377_0_0144_0_001_0_0
02_0_0_1377_afr_rs76192746")]
# Rename the columns
colnames(selected_columns_chr5) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

# If 'combined_data' exists and contains the required columns
if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

  # Filter combined_data for chromosome 5
  chr5_data <- combined_data %>%
    filter(chr == 5)
```

```
  # Extract unique position numbers from chromosome 5
  chr5_positions <- unique(chr5_data$pos)

  # Display the first few position numbers for chromosome 5
  print(head(chr5_positions))

} else {
  stop("Error: 'combined_data' is either not loaded or does not contain the required columns ('chr'
and 'pos').")
}

# Define the positions to filter
chr5_positions <- c(11886744)  # Update with relevant positions for chromosome 5

# Filter the data based on these positions
filtered_data_chr5 <- selected_columns_chr5[selected_columns_chr5$POS %in%
chr5_positions, ]

# Check the filtered data
head(filtered_data_chr5)

# Ensure unique column names
colnames(filtered_data_chr5) <- make.unique(colnames(filtered_data_chr5))

# Extract CSQ field and remove "CSQ=" prefix
filtered_data_chr5 <- filtered_data_chr5 %>%
  mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%  # Extract CSQ field
  mutate(CSQ = str_remove(CSQ, "CSQ="))            # Remove "CSQ=" prefix

# Verify the CSQ field extraction
print(head(filtered_data_chr5$CSQ, 10))  # Check the first 10 rows

# Check if any CSQ values are empty or missing
print(sum(is.na(filtered_data_chr5$CSQ)))  # Count NA values in CSQ
print(sum(filtered_data_chr5$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

# Filter out rows where CSQ is missing or empty
filtered_data_chr5 <- filtered_data_chr5 %>%
  filter(!is.na(CSQ) & CSQ != "")

# Recalculate max_fields based on the cleaned data
max_fields <- filtered_data_chr5 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%  # Count delimiters and add 1
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%         # Find the maximum
  pull(max_fields)
```

```
# Print max_fields to verify it's a reasonable value
print(max_fields)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr5 <- filtered_data_chr5 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields), sep = "\\|", fill = "right")

# Select only the desired columns
# Use dplyr::select explicitly to avoid conflicts
desired_columns_chr5 <- dplyr::select(filtered_data_chr5, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)


# Rename the selected columns
desired_columns_chr5 <- dplyr::rename(desired_columns_chr5,
                                      Protein_coding = Field8,
                                      Variant_type = Field2,
                                      Gene_symbols = Field4,
                                      Ensembl_Gene_IDs = Field5,
                                      Variant_feature_types = Field6,
                                      rs_Ids = Field18)

# Save the results
write.csv(desired_columns_chr5, file="/sbgenomics/output-files/desired_columns_chr5.csv")

##############################Chromosome 6 ##########################

# Read in the VEP output file for chromosome 6
  vep_data_chr6 <- read.table("/sbgenomics/project-files/tmplr_subset_chr6.processed.vep.vcf",
sep="\t", header=TRUE, comment.char="#")

# Display the names of the first 8 columns
first_8_colnames_chr6 <- colnames(vep_data_chr6)[1:8]
print(first_8_colnames_chr6)

# Automatically make column names unique by appending a suffix
vep_data_chr6 <- vep_data_chr6 %>%
  clean_names()  # Ensures no duplicate column names

# Create the subset with the relevant columns for chromosome 6
selected_columns_chr6 <- vep_data_chr6[, c("x6", "x149609", "jhu_6_149608", "g", "a",
"ac_777_an_860_csq_a_upstream_gene_variant_modifier_cicp18_ensg00000217929_transcript
_enst00000406017_processed_pseudogene_rs7754266_4526_1_snv_hgnc_48828_yes_rs775426
6_a_upstream_gene_variant_modifier_linc00266_3_ensg00000170590_transcript_enst00000436
899_linc_rna_rs7754266_4724_1_snv_hgnc_21112_yes_rs7754266")]
```

```
# Rename the columns
colnames(selected_columns_chr6) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

# If 'combined_data' exists and contains the required columns
if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

  # Filter combined_data for chromosome 6
  chr6_data <- combined_data %>%
    filter(chr == 6)

  # Extract unique position numbers from chromosome 6
  chr6_positions <- unique(chr6_data$pos)

  # Display the first few position numbers for chromosome 6
  print(head(chr6_positions))

} else {
  stop("Error: 'combined_data' is either not loaded or does not contain the required columns ('chr'
and 'pos').")
}

# Define the positions to filter for chromosome 6
chr6_positions <- c(162162658,68898262,68938286,68938740)  # Update with relevant
positions for chromosome 6

# Filter the data based on these positions
filtered_data_chr6 <- selected_columns_chr6[selected_columns_chr6$POS %in%
chr6_positions, ]

# Check the filtered data
head(filtered_data_chr6)

# Ensure unique column names
colnames(filtered_data_chr6) <- make.unique(colnames(filtered_data_chr6))

# Extract CSQ field and remove "CSQ=" prefix
filtered_data_chr6 <- filtered_data_chr6 %>%
  mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%  # Extract CSQ field
  mutate(CSQ = str_remove(CSQ, "CSQ="))            # Remove "CSQ=" prefix

# Verify the CSQ field extraction
print(head(filtered_data_chr6$CSQ, 10))  # Check the first 10 rows

# Check if any CSQ values are empty or missing
print(sum(is.na(filtered_data_chr6$CSQ)))  # Count NA values in CSQ
print(sum(filtered_data_chr6$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ
```

```
# Filter out rows where CSQ is missing or empty
filtered_data_chr6 <- filtered_data_chr6 %>%
  filter(!is.na(CSQ) & CSQ != "")

# Recalculate max_fields based on the cleaned data
max_fields_chr6 <- filtered_data_chr6 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%  # Count delimiters and add 1
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%         # Find the maximum
  pull(max_fields)

# Print max_fields to verify it's a reasonable value
print(max_fields_chr6)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr6 <- filtered_data_chr6 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields_chr6), sep = "\\|", fill = "right")

# Select only the desired columns
# Use dplyr::select explicitly to avoid conflicts
desired_columns_chr6 <- dplyr::select(filtered_data_chr6, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

# Rename the selected columns
desired_columns_chr6 <- dplyr::rename(desired_columns_chr6,
                                      Protein_coding = Field8,
                                      Variant_type = Field2,
                                      Gene_symbols = Field4,
                                      Ensembl_Gene_IDs = Field5,
                                      Variant_feature_types = Field6,
                                      rs_Ids = Field18)

# Save the results
write.csv(desired_columns_chr6, file="/sbgenomics/output-files/desired_columns_chr6.csv")


######################Chromosome 7 ########################################

# Read in the VEP output file for chromosome 7
vep_data_chr7 <- read.table("/sbgenomics/project-files/tmplr_subset_chr7.processed.vep.vcf",
sep="\t", header=TRUE, comment.char="#")

# Display the names of the first 8 columns
first_8_colnames_chr7 <- colnames(vep_data_chr7)[1:8]
print(first_8_colnames_chr7)
```

```
# Automatically make column names unique by appending a suffix
vep_data_chr7 <- vep_data_chr7 %>%
  clean_names()  # Ensures no duplicate column names

# Create the subset with the relevant columns for chromosome 7
selected_columns_chr7 <- vep_data_chr7[, c("x7", "x16487", "jhu_7_16486", "g", "t",
"ac_868_an_868_csq_t_intergenic_variant_modifier_rs190578649_snv_rs190578649")]

# Rename the columns
colnames(selected_columns_chr7) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

# If 'combined_data' exists and contains the required columns
if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

  # Filter combined_data for chromosome 7
  chr7_data <- combined_data %>%
    filter(chr == 7)

  # Extract unique position numbers from chromosome 7
  chr7_positions <- unique(chr7_data$pos)

  # Display the first few position numbers for chromosome 7
  print(head(chr7_positions))

} else {
  stop("Error: 'combined_data' is either not loaded or does not contain the required columns ('chr'
and 'pos').")
}

# Define the positions to filter for chromosome 7
chr7_positions <- c(121147147,13333829)  # Update with relevant positions for chromosome 7

# Filter the data based on these positions
filtered_data_chr7 <- selected_columns_chr7[selected_columns_chr7$POS %in%
chr7_positions, ]

# Check the filtered data
head(filtered_data_chr7)

# Ensure unique column names
colnames(filtered_data_chr7) <- make.unique(colnames(filtered_data_chr7))

# Extract CSQ field and remove "CSQ=" prefix
filtered_data_chr7 <- filtered_data_chr7 %>%
  mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%  # Extract CSQ field
  mutate(CSQ = str_remove(CSQ, "CSQ="))            # Remove "CSQ=" prefix
```

```
# Verify the CSQ field extraction
print(head(filtered_data_chr7$CSQ, 10))  # Check the first 10 rows

# Check if any CSQ values are empty or missing
print(sum(is.na(filtered_data_chr7$CSQ)))  # Count NA values in CSQ
print(sum(filtered_data_chr7$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

# Filter out rows where CSQ is missing or empty
filtered_data_chr7 <- filtered_data_chr7 %>%
  filter(!is.na(CSQ) & CSQ != "")

# Recalculate max_fields based on the cleaned data
max_fields_chr7 <- filtered_data_chr7 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%  # Count delimiters and add 1
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%         # Find the maximum
  pull(max_fields)

# Print max_fields to verify it's a reasonable value
print(max_fields_chr7)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr7 <- filtered_data_chr7 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields_chr7), sep = "\\|", fill = "right")

# Select only the desired columns
# Use dplyr::select explicitly to avoid conflicts
desired_columns_chr7 <- dplyr::select(filtered_data_chr7, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

# Rename the selected columns
desired_columns_chr7 <- dplyr::rename(desired_columns_chr7,
                                      Protein_coding = Field8,
                                      Variant_type = Field2,
                                      Gene_symbols = Field4,
                                      Ensembl_Gene_IDs = Field5,
                                      Variant_feature_types = Field6,
                                      rs_Ids = Field18)
write.csv(desired_columns_chr7, file="/sbgenomics/output-files/desired_columns_chr7.csv")


  ##############################chromosome 8 ################################

  # Read in the VEP output file for chromosome 8
  vep_data_chr8 <- read.table("/sbgenomics/project-files/tmplr_subset_chr8.processed.vep.vcf",
sep="\t", header=TRUE, comment.char="#")
```

```
  # Display the names of the first 8 columns
  first_8_colnames_chr8 <- colnames(vep_data_chr8)[1:8]
  print(first_8_colnames_chr8)

  # Automatically make column names unique by appending a suffix
  vep_data_chr8 <- vep_data_chr8 %>%
    clean_names()  # Ensures no duplicate column names

  # Create the subset with the relevant columns for chromosome 8
  selected_columns_chr8 <- vep_data_chr8[, c("x8", "x161610", "jhu_8_161609", "t", "g",
"ac_3_an_870_csq_g_downstream_gene_variant_modifier_rpl23ap53_ensg00000223508_transc
ript_enst00000521854_transcribed_processed_pseudogene_rs866724732_1641_1_snv_hgnc_35
921_g_downstream_gene_variant_modifier_rpl23ap53_ensg00000223508_transcript_enst00000
606975_processed_transcript_rs866724732_1576_1_snv_hgnc_35921_yes_g_regulatory_region
_variant_modifier_regulatory_feature_ensr00001416842_promoter_flanking_region_rs8667247
32_snv")]
  # Rename the columns
  colnames(selected_columns_chr8) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

  # If 'combined_data' exists and contains the required columns
  if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

    # Filter combined_data for chromosome 8
    chr8_data <- combined_data %>%
      filter(chr == 8)

    # Extract unique position numbers from chromosome 8
    chr8_positions <- unique(chr8_data$pos)

    # Display the first few position numbers for chromosome 8
    print(head(chr8_positions))

  } else {
    stop("Error: 'combined_data' is either not loaded or does not contain the required columns
('chr' and 'pos').")
  }

  # Define the positions to filter for chromosome 8
  chr8_positions <- c(74548187)  # Update with relevant positions for chromosome 8

  # Filter the data based on these positions
  filtered_data_chr8 <- selected_columns_chr8[selected_columns_chr8$POS %in%
chr8_positions, ]

  # Check the filtered data
```

```
head(filtered_data_chr8)

# Ensure unique column names
colnames(filtered_data_chr8) <- make.unique(colnames(filtered_data_chr8))

# Extract CSQ field and remove "CSQ=" prefix
filtered_data_chr8 <- filtered_data_chr8 %>%
  mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%  # Extract CSQ field
  mutate(CSQ = str_remove(CSQ, "CSQ="))            # Remove "CSQ=" prefix

# Verify the CSQ field extraction
print(head(filtered_data_chr8$CSQ, 10))  # Check the first 10 rows

# Check if any CSQ values are empty or missing
print(sum(is.na(filtered_data_chr8$CSQ)))  # Count NA values in CSQ
print(sum(filtered_data_chr8$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

# Filter out rows where CSQ is missing or empty
filtered_data_chr8 <- filtered_data_chr8 %>%
  filter(!is.na(CSQ) & CSQ != "")

# Recalculate max_fields based on the cleaned data
max_fields_chr8 <- filtered_data_chr8 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%  # Count delimiters and add 1
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%        # Find the maximum
  pull(max_fields)

# Print max_fields to verify it's a reasonable value
print(max_fields_chr8)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr8 <- filtered_data_chr8 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields_chr8), sep = "\\|", fill = "right")

# Select only the desired columns
# Use dplyr::select explicitly to avoid conflicts
desired_columns_chr8 <- dplyr::select(filtered_data_chr8, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

# Rename the selected columns
desired_columns_chr8 <- dplyr::rename(desired_columns_chr8,
                                      Protein_coding = Field8,
                                      Variant_type = Field2,
                                      Gene_symbols = Field4,
                                      Ensembl_Gene_IDs = Field5,
                                      Variant_feature_types = Field6,
```

```
                        rs_Ids = Field18)

  # Save the results
  write.csv(desired_columns_chr8, file="/sbgenomics/output-files/desired_columns_chr8.csv")

###################chromosome 11 ######################################


  # Read in the VEP output file for chromosome 11
  vep_data_chr11 <- read.table("/sbgenomics/project-
files/tmplr_subset_chr11.processed.vep.vcf", sep="\t", header=TRUE, comment.char="#")

  # Display the names of the first 8 columns
  first_8_colnames_chr11 <- colnames(vep_data_chr11)[1:8]
  print(first_8_colnames_chr11)

  # Automatically make column names unique by appending a suffix
  vep_data_chr11 <- vep_data_chr11 %>%
    clean_names()  # Ensures no duplicate column names

  # Create the subset with the relevant columns for chromosome 11
  selected_columns_chr11 <- vep_data_chr11[, c(1, 2, 3, 4, 5, 8)]

  # Rename the columns
  colnames(selected_columns_chr11) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

  # If 'combined_data' exists and contains the required columns
  if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

    # Filter combined_data for chromosome 11
    chr11_data <- combined_data %>%
      filter(chr == 11)

    # Extract unique position numbers from chromosome 11
    chr11_positions <- unique(chr11_data$pos)

    # Display the first few position numbers for chromosome 11
    print(head(chr11_positions))

  } else {
    stop("Error: 'combined_data' is either not loaded or does not contain the required columns
('chr' and 'pos').")
  }

  # Define the positions to filter for chromosome 11
```

```
  chr11_positions <- c(63238576,126869125)  # Update with relevant positions for chromosome
11

  # Filter the data based on these positions
  filtered_data_chr11 <- selected_columns_chr11[selected_columns_chr11$POS %in%
chr11_positions, ]

  # Check the filtered data
  head(filtered_data_chr11)

  # Ensure unique column names
  colnames(filtered_data_chr11) <- make.unique(colnames(filtered_data_chr11))

  # Extract CSQ field and remove "CSQ=" prefix
  filtered_data_chr11 <- filtered_data_chr11 %>%
    mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%  # Extract CSQ field
    mutate(CSQ = str_remove(CSQ, "CSQ="))            # Remove "CSQ=" prefix

  # Verify the CSQ field extraction
  print(head(filtered_data_chr11$CSQ, 10))  # Check the first 10 rows

  # Check if any CSQ values are empty or missing
  print(sum(is.na(filtered_data_chr11$CSQ)))  # Count NA values in CSQ
  print(sum(filtered_data_chr11$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

  # Filter out rows where CSQ is missing or empty
  filtered_data_chr11 <- filtered_data_chr11 %>%
    filter(!is.na(CSQ) & CSQ != "")

  # Recalculate max_fields based on the cleaned data
  max_fields_chr11 <- filtered_data_chr11 %>%
    mutate(field_count = str_count(CSQ, "\\|") + 1) %>%  # Count delimiters and add 1
    summarise(max_fields = max(field_count, na.rm = TRUE)) %>%  # Find the maximum
    pull(max_fields)

  # Print max_fields to verify it's a reasonable value
  print(max_fields_chr11)

  # Separate the CSQ column into multiple fields based on the max_fields
  filtered_data_chr11 <- filtered_data_chr11 %>%
    separate(CSQ, into = paste0("Field", 1:max_fields_chr11), sep = "\\|", fill = "right")

  # Select only the desired columns
  # Use dplyr::select explicitly to avoid conflicts
  desired_columns_chr11 <- dplyr::select(filtered_data_chr11, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)
```

```
# Rename the selected columns
desired_columns_chr11 <- dplyr::rename(desired_columns_chr11,
                                       Protein_coding = Field8,
                                       Variant_type = Field2,
                                       Gene_symbols = Field4,
                                       Ensembl_Gene_IDs = Field5,
                                       Variant_feature_types = Field6,
                                       rs_Ids = Field18)

# Save the results
write.csv(desired_columns_chr11, file="/sbgenomics/output-files/desired_columns_chr11.csv")




########################## chromosome 12 ###################################

# Read in the VEP output file for chromosome 12
vep_data_chr12 <- read.table("/sbgenomics/project-
files/tmplr_subset_chr12.processed.vep.vcf", sep="\t", header=TRUE, comment.char="#")

# Display the names of the first 8 columns
first_8_colnames_chr12 <- colnames(vep_data_chr12)[1:8]
print(first_8_colnames_chr12)

# Automatically make column names unique by appending a suffix
vep_data_chr12 <- vep_data_chr12 %>%
  clean_names()  # Ensures no duplicate column names

# Create the subset with the relevant columns for chromosome 12
selected_columns_chr12 <- vep_data_chr12[, c(1, 2, 3, 4, 5, 8)]

# Rename the columns
colnames(selected_columns_chr12) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

# If 'combined_data' exists and contains the required columns
if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

  # Filter combined_data for chromosome 12
  chr12_data <- combined_data %>%
    filter(chr == 12)

  # Extract unique position numbers from chromosome 12
```

```
  chr12_positions <- unique(chr12_data$pos)

  # Display the first few position numbers for chromosome 12
  print(head(chr12_positions))

 } else {
  stop("Error: 'combined_data' is either not loaded or does not contain the required columns
('chr' and 'pos').")
 }

 # Define the positions to filter for chromosome 12
 chr12_positions <- c(2994612,34197919)  # Update with relevant positions for chromosome 12

 # Filter the data based on these positions
 filtered_data_chr12 <- selected_columns_chr12[selected_columns_chr12$POS %in%
chr12_positions, ]

 # Check the filtered data
 head(filtered_data_chr12)

 # Ensure unique column names
 colnames(filtered_data_chr12) <- make.unique(colnames(filtered_data_chr12))

 # Extract CSQ field and remove "CSQ=" prefix
 filtered_data_chr12 <- filtered_data_chr12 %>%
  mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%
  mutate(CSQ = str_remove(CSQ, "CSQ="))

 # Verify the CSQ field extraction
 print(head(filtered_data_chr12$CSQ, 10))  # Check the first 10 rows

 # Check if any CSQ values are empty or missing
 print(sum(is.na(filtered_data_chr12$CSQ)))  # Count NA values in CSQ
 print(sum(filtered_data_chr12$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

 # Filter out rows where CSQ is missing or empty
 filtered_data_chr12 <- filtered_data_chr12 %>%
  filter(!is.na(CSQ) & CSQ != "")

 # Recalculate max_fields based on the cleaned data
 max_fields_chr12 <- filtered_data_chr12 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%
  pull(max_fields)

 # Print max_fields to verify it's a reasonable value
```

```
  print(max_fields_chr12)

  # Separate the CSQ column into multiple fields based on the max_fields
  filtered_data_chr12 <- filtered_data_chr12 %>%
    separate(CSQ, into = paste0("Field", 1:max_fields_chr12), sep = "\\|", fill = "right")

  # Select only the desired columns
  desired_columns_chr12 <- dplyr::select(filtered_data_chr12, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

  # Rename the selected columns
  desired_columns_chr12 <- dplyr::rename(desired_columns_chr12,
                                Protein_coding = Field8,
                                Variant_type = Field2,
                                Gene_symbols = Field4,
                                Ensembl_Gene_IDs = Field5,
                                Variant_feature_types = Field6,
                                rs_Ids = Field18)

  # Save the results
  write.csv(desired_columns_chr12, file="/sbgenomics/output-files/desired_columns_chr12.csv")


  ########################chromosome 14 ############################

  # Read in the VEP output file for chromosome 14
  vep_data_chr14 <- read.table("/sbgenomics/project-
files/tmplr_subset_chr14.processed.vep.vcf", sep="\t", header=TRUE, comment.char="#")

  # Display the names of the first 8 columns
  first_8_colnames_chr14 <- colnames(vep_data_chr14)[1:8]
  print(first_8_colnames_chr14)

  # Automatically make column names unique by appending a suffix
  vep_data_chr14 <- vep_data_chr14 %>%
    clean_names()  # Ensures no duplicate column names

  # Create the subset with the relevant columns for chromosome 14
  selected_columns_chr14 <- vep_data_chr14[, c(1, 2, 3, 4, 5, 8)]

  # Rename the columns
  colnames(selected_columns_chr14) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

  # If 'combined_data' exists and contains the required columns
  if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {
```

```
    # Filter combined_data for chromosome 14
    chr14_data <- combined_data %>%
      filter(chr == 14)

    # Extract unique position numbers from chromosome 14
    chr14_positions <- unique(chr14_data$pos)

    # Display the first few position numbers for chromosome 14
    print(head(chr14_positions))

  } else {
    stop("Error: 'combined_data' is either not loaded or does not contain the required columns
('chr' and 'pos').")
  }

  # Define the positions to filter for chromosome 14
  chr14_positions <- c(72122311, 94229429, 104642489)  # Update with relevant positions for
chromosome 14

  # Filter the data based on these positions
  filtered_data_chr14 <- selected_columns_chr14[selected_columns_chr14$POS %in%
chr14_positions, ]

  # Check the filtered data
  head(filtered_data_chr14)

  # Ensure unique column names
  colnames(filtered_data_chr14) <- make.unique(colnames(filtered_data_chr14))

  # Extract CSQ field and remove "CSQ=" prefix
  filtered_data_chr14 <- filtered_data_chr14 %>%
    mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%
    mutate(CSQ = str_remove(CSQ, "CSQ="))

  # Verify the CSQ field extraction
  print(head(filtered_data_chr14$CSQ, 10))  # Check the first 10 rows

  # Check if any CSQ values are empty or missing
  print(sum(is.na(filtered_data_chr14$CSQ)))  # Count NA values in CSQ
  print(sum(filtered_data_chr14$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

  # Filter out rows where CSQ is missing or empty
  filtered_data_chr14 <- filtered_data_chr14 %>%
    filter(!is.na(CSQ) & CSQ != "")

  # Recalculate max_fields based on the cleaned data
```

```
  max_fields_chr14 <- filtered_data_chr14 %>%
    mutate(field_count = str_count(CSQ, "\\|") + 1) %>%
    summarise(max_fields = max(field_count, na.rm = TRUE)) %>%
    pull(max_fields)

  # Print max_fields to verify it's a reasonable value
  print(max_fields_chr14)

  # Separate the CSQ column into multiple fields based on the max_fields
  filtered_data_chr14 <- filtered_data_chr14 %>%
    separate(CSQ, into = paste0("Field", 1:max_fields_chr14), sep = "\\|", fill = "right")

  # Select only the desired columns
  desired_columns_chr14 <- dplyr::select(filtered_data_chr14, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

  # Rename the selected columns
  desired_columns_chr14 <- dplyr::rename(desired_columns_chr14,
                                         Protein_coding = Field8,
                                         Variant_type = Field2,
                                         Gene_symbols = Field4,
                                         Ensembl_Gene_IDs = Field5,
                                         Variant_feature_types = Field6,
                                         rs_Ids = Field18)

  # Save the results
  write.csv(desired_columns_chr14, file="/sbgenomics/output-files/desired_columns_chr14.csv")

  ########################chr 16 ##############################
  # Read in the VEP output file for chromosome 16
  vep_data_chr16 <- read.table("/sbgenomics/project-
files/tmplr_subset_chr16.processed.vep.vcf", sep="\t", header=TRUE, comment.char="#")

  # Display the names of the first 8 columns
  first_8_colnames_chr16 <- colnames(vep_data_chr16)[1:8]
  print(first_8_colnames_chr16)

  # Automatically make column names unique by appending a suffix
  vep_data_chr16 <- vep_data_chr16 %>%
    clean_names()  # Ensures no duplicate column names

  # Create the subset with the relevant columns for chromosome 16
  selected_columns_chr16 <- vep_data_chr16[, c(1, 2, 3, 4, 5, 8)]

  # Rename the columns
  colnames(selected_columns_chr16) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")
```

```
  # If 'combined_data' exists and contains the required columns
  if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

    # Filter combined_data for chromosome 16
    chr16_data <- combined_data %>%
      filter(chr == 16)

    # Extract unique position numbers from chromosome 16
    chr16_positions <- unique(chr16_data$pos)

    # Display the first few position numbers for chromosome 16
    print(head(chr16_positions))

  } else {
    stop("Error: 'combined_data' is either not loaded or does not contain the required columns
('chr' and 'pos').")
  }

  # Define the positions to filter for chromosome 16
  chr16_positions <- c(17722077)  # Update with relevant positions for chromosome 16

  # Filter the data based on these positions
  filtered_data_chr16 <- selected_columns_chr16[selected_columns_chr16$POS %in%
chr16_positions, ]

  # Check the filtered data
  head(filtered_data_chr16)

  # Ensure unique column names
  colnames(filtered_data_chr16) <- make.unique(colnames(filtered_data_chr16))

  # Extract CSQ field and remove "CSQ=" prefix
  filtered_data_chr16 <- filtered_data_chr16 %>%
    mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%
    mutate(CSQ = str_remove(CSQ, "CSQ="))

  # Verify the CSQ field extraction
  print(head(filtered_data_chr16$CSQ, 10))  # Check the first 10 rows

  # Check if any CSQ values are empty or missing
  print(sum(is.na(filtered_data_chr16$CSQ)))  # Count NA values in CSQ
  print(sum(filtered_data_chr16$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

  # Filter out rows where CSQ is missing or empty
  filtered_data_chr16 <- filtered_data_chr16 %>%
```

```
  filter(!is.na(CSQ) & CSQ != "")

# Recalculate max_fields based on the cleaned data
max_fields_chr16 <- filtered_data_chr16 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%
  pull(max_fields)

# Print max_fields to verify it's a reasonable value
print(max_fields_chr16)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr16 <- filtered_data_chr16 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields_chr16), sep = "\\|", fill = "right")

# Select only the desired columns
desired_columns_chr16 <- dplyr::select(filtered_data_chr16, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

# Rename the selected columns
desired_columns_chr16 <- dplyr::rename(desired_columns_chr16,
                                       Protein_coding = Field8,
                                       Variant_type = Field2,
                                       Gene_symbols = Field4,
                                       Ensembl_Gene_IDs = Field5,
                                       Variant_feature_types = Field6,
                                       rs_Ids = Field18)

# Save the results
write.csv(desired_columns_chr16, file="/sbgenomics/output-files/desired_columns_chr16.csv")

######################### chr20##############################
# Read in the VEP output file for chromosome 20
vep_data_chr20 <- read.table("/sbgenomics/project-
files/tmplr_subset_chr20.processed.vep.vcf", sep="\t", header=TRUE, comment.char="#")

# Display the names of the first 8 columns
first_8_colnames_chr20 <- colnames(vep_data_chr20)[1:8]
print(first_8_colnames_chr20)

# Automatically make column names unique by appending a suffix
vep_data_chr20 <- vep_data_chr20 %>%
  clean_names()  # Ensures no duplicate column names

# Create the subset with the relevant columns for chromosome 20
selected_columns_chr20 <- vep_data_chr20[, c(1, 2, 3, 4, 5, 8)]
```

```
  # Rename the columns
  colnames(selected_columns_chr20) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

  # If 'combined_data' exists and contains the required columns
  if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

    # Filter combined_data for chromosome 20
    chr20_data <- combined_data %>%
      filter(chr == 20)

    # Extract unique position numbers from chromosome 20
    chr20_positions <- unique(chr20_data$pos)

    # Display the first few position numbers for chromosome 20
    print(head(chr20_positions))

  } else {
    stop("Error: 'combined_data' is either not loaded or does not contain the required columns
('chr' and 'pos').")
  }

  # Define the positions to filter for chromosome 20
  chr20_positions <- c(10756166, 10759245)  # Update with relevant positions for chromosome
20

  # Filter the data based on these positions
  filtered_data_chr20 <- selected_columns_chr20[selected_columns_chr20$POS %in%
chr20_positions, ]

  # Check the filtered data
  head(filtered_data_chr20)

  # Ensure unique column names
  colnames(filtered_data_chr20) <- make.unique(colnames(filtered_data_chr20))

  # Extract CSQ field and remove "CSQ=" prefix
  filtered_data_chr20 <- filtered_data_chr20 %>%
    mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%
    mutate(CSQ = str_remove(CSQ, "CSQ="))

  # Verify the CSQ field extraction
  print(head(filtered_data_chr20$CSQ, 10))  # Check the first 10 rows

  # Check if any CSQ values are empty or missing
  print(sum(is.na(filtered_data_chr20$CSQ)))  # Count NA values in CSQ
```

```
print(sum(filtered_data_chr20$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

# Filter out rows where CSQ is missing or empty
filtered_data_chr20 <- filtered_data_chr20 %>%
  filter(!is.na(CSQ) & CSQ != "")

# Recalculate max_fields based on the cleaned data
max_fields_chr20 <- filtered_data_chr20 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%
  pull(max_fields)

# Print max_fields to verify it's a reasonable value
print(max_fields_chr20)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr20 <- filtered_data_chr20 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields_chr20), sep = "\\|", fill = "right")

# Select only the desired columns
desired_columns_chr20 <- dplyr::select(filtered_data_chr20, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

# Rename the selected columns
desired_columns_chr20 <- dplyr::rename(desired_columns_chr20,
                                       Protein_coding = Field8,
                                       Variant_type = Field2,
                                       Gene_symbols = Field4,
                                       Ensembl_Gene_IDs = Field5,
                                       Variant_feature_types = Field6,
                                       rs_Ids = Field18)

# Save the results
write.csv(desired_columns_chr20, file="/sbgenomics/output-files/desired_columns_chr20.csv")


###################### chr22 ######################
# Read in the VEP output file for chromosome 22
vep_data_chr22 <- read.table("/sbgenomics/project-
files/tmplr_subset_chr22.processed.vep.vcf", sep="\t", header=TRUE, comment.char="#")

# Display the names of the first 8 columns
first_8_colnames_chr22 <- colnames(vep_data_chr22)[1:8]
print(first_8_colnames_chr22)

# Automatically make column names unique by appending a suffix
```

```
  vep_data_chr22 <- vep_data_chr22 %>%
    clean_names()  # Ensures no duplicate column names

  # Create the subset with the relevant columns for chromosome 22
  selected_columns_chr22 <- vep_data_chr22[, c(1, 2, 3, 4, 5, 8)]

  # Rename the columns
  colnames(selected_columns_chr22) <- c("CHROM", "POS", "ID", "REF", "ALT", "INFO")

  # If 'combined_data' exists and contains the required columns
  if (exists("combined_data") && all(c("chr", "pos") %in% colnames(combined_data))) {

    # Filter combined_data for chromosome 22
    chr22_data <- combined_data %>%
      filter(chr == 22)

    # Extract unique position numbers from chromosome 22
    chr22_positions <- unique(chr22_data$pos)

    # Display the first few position numbers for chromosome 22
    print(head(chr22_positions))

  } else {
    stop("Error: 'combined_data' is either not loaded or does not contain the required columns
('chr' and 'pos').")
  }

  # Define the positions to filter for chromosome 22
  chr22_positions <- c(26917626)  # Update with relevant positions for chromosome 22

  # Filter the data based on these positions
  filtered_data_chr22 <- selected_columns_chr22[selected_columns_chr22$POS %in%
chr22_positions, ]

  # Check the filtered data
  head(filtered_data_chr22)

  # Ensure unique column names
  colnames(filtered_data_chr22) <- make.unique(colnames(filtered_data_chr22))

  # Extract CSQ field and remove "CSQ=" prefix
  filtered_data_chr22 <- filtered_data_chr22 %>%
    mutate(CSQ = str_extract(INFO, "CSQ=[^;]*")) %>%
    mutate(CSQ = str_remove(CSQ, "CSQ="))

  # Verify the CSQ field extraction
```

```
print(head(filtered_data_chr22$CSQ, 10))  # Check the first 10 rows

# Check if any CSQ values are empty or missing
print(sum(is.na(filtered_data_chr22$CSQ)))  # Count NA values in CSQ
print(sum(filtered_data_chr22$CSQ == "", na.rm = TRUE))  # Count empty strings in CSQ

# Filter out rows where CSQ is missing or empty
filtered_data_chr22 <- filtered_data_chr22 %>%
  filter(!is.na(CSQ) & CSQ != "")

# Recalculate max_fields based on the cleaned data
max_fields_chr22 <- filtered_data_chr22 %>%
  mutate(field_count = str_count(CSQ, "\\|") + 1) %>%
  summarise(max_fields = max(field_count, na.rm = TRUE)) %>%
  pull(max_fields)

# Print max_fields to verify it's a reasonable value
print(max_fields_chr22)

# Separate the CSQ column into multiple fields based on the max_fields
filtered_data_chr22 <- filtered_data_chr22 %>%
  separate(CSQ, into = paste0("Field", 1:max_fields_chr22), sep = "\\|", fill = "right")

# Select only the desired columns
desired_columns_chr22 <- dplyr::select(filtered_data_chr22, CHROM, POS, ID, REF, ALT,
Field5, Field18, Field6, Field4, Field8, Field2)

# Rename the selected columns
desired_columns_chr22 <- dplyr::rename(desired_columns_chr22,
                                       Protein_coding = Field8,
                                       Variant_type = Field2,
                                       Gene_symbols = Field4,
                                       Ensembl_Gene_IDs = Field5,
                                       Variant_feature_types = Field6,
                                       rs_Ids = Field18)

# Save the results
write.csv(desired_columns_chr22, file="/sbgenomics/output-files/desired_columns_chr22.csv")
```

Appendix E: Descriptive Statistics

This appendix gives a full description of the descriptive statistics of the participants in TMPLR (**Table E1**). Figure E1 – E6 shows the distribution of all the obesity phenotypes namely, total, android, gynoid, arms, legs and trunk fat mass.

Table E1: Mean and SD of participants in tmplr

Gender	Count	Height Mean	Height SD	Height Min	Height Max	Weight Mean	Weight SD	Weight Min	Weight Max	Age Mean	Age SD	Age Min	Age Max
Female	168	165.2	6.8	146.3	186.4	75.7	19.2	48.2	155.6	38.7	4.7	30	46
Male	223	178.5	7.5	128.1	194.6	88.0	15.5	54.2	144.4	37.4	4.8	30	46

Figure E1: Distribution of android fat mass phenotype

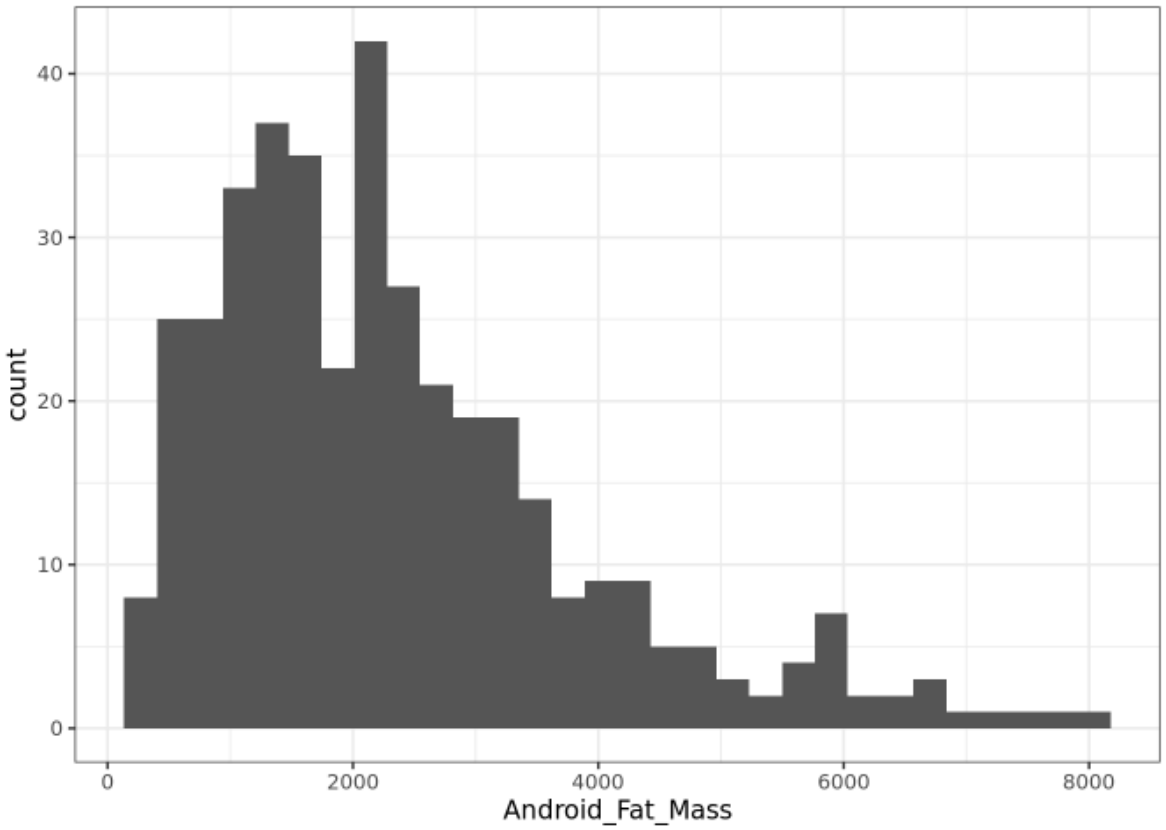


Figure E2: Distribution of android fat mass phenotype

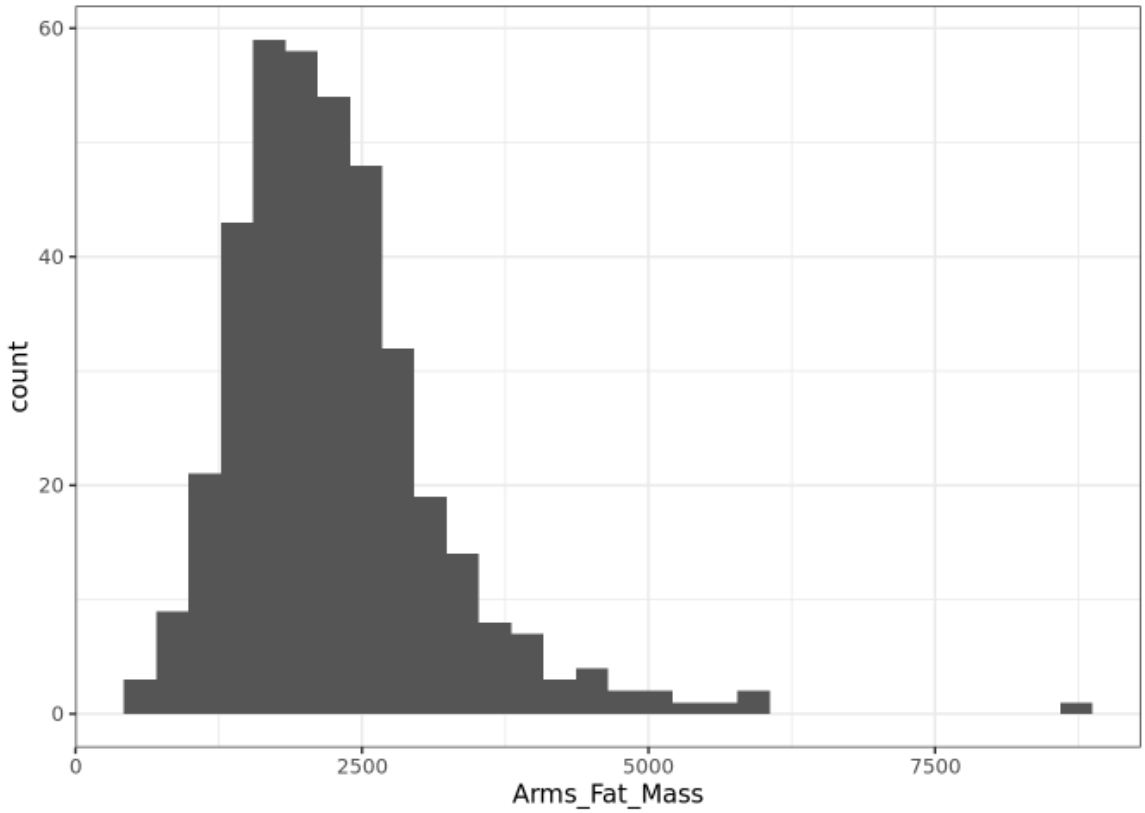


Figure E3: Distribution of android fat mass phenotype

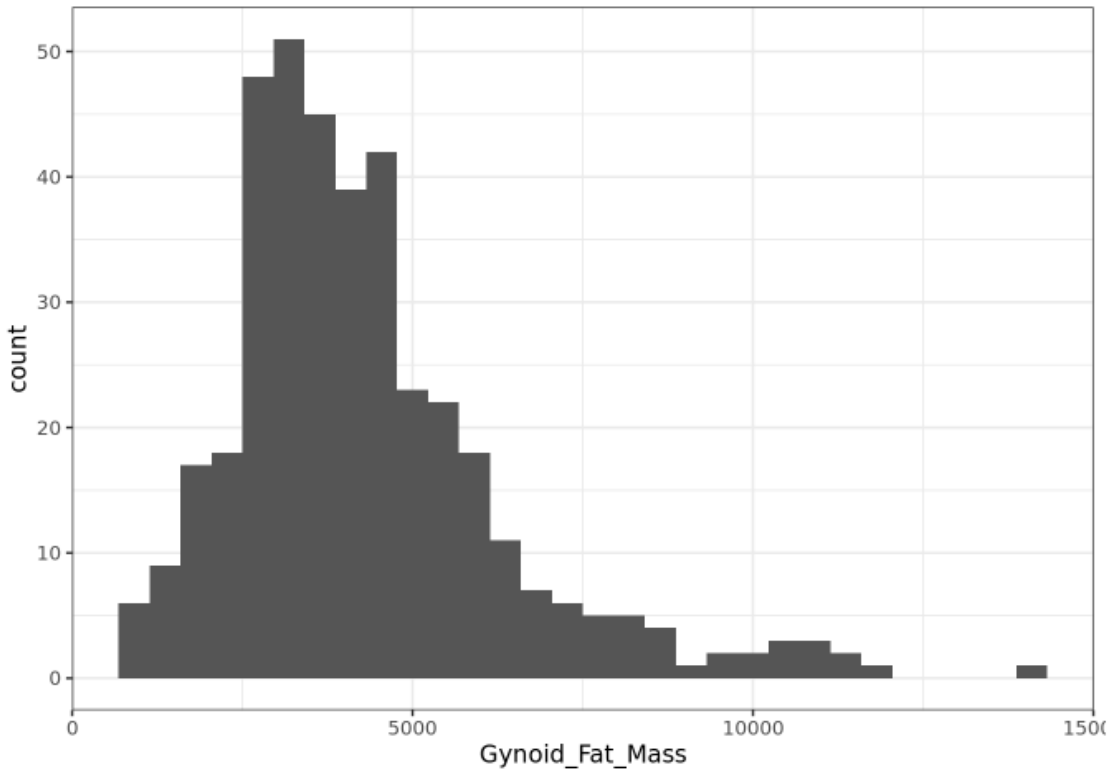


Figure E4: Distribution of android fat mass phenotype

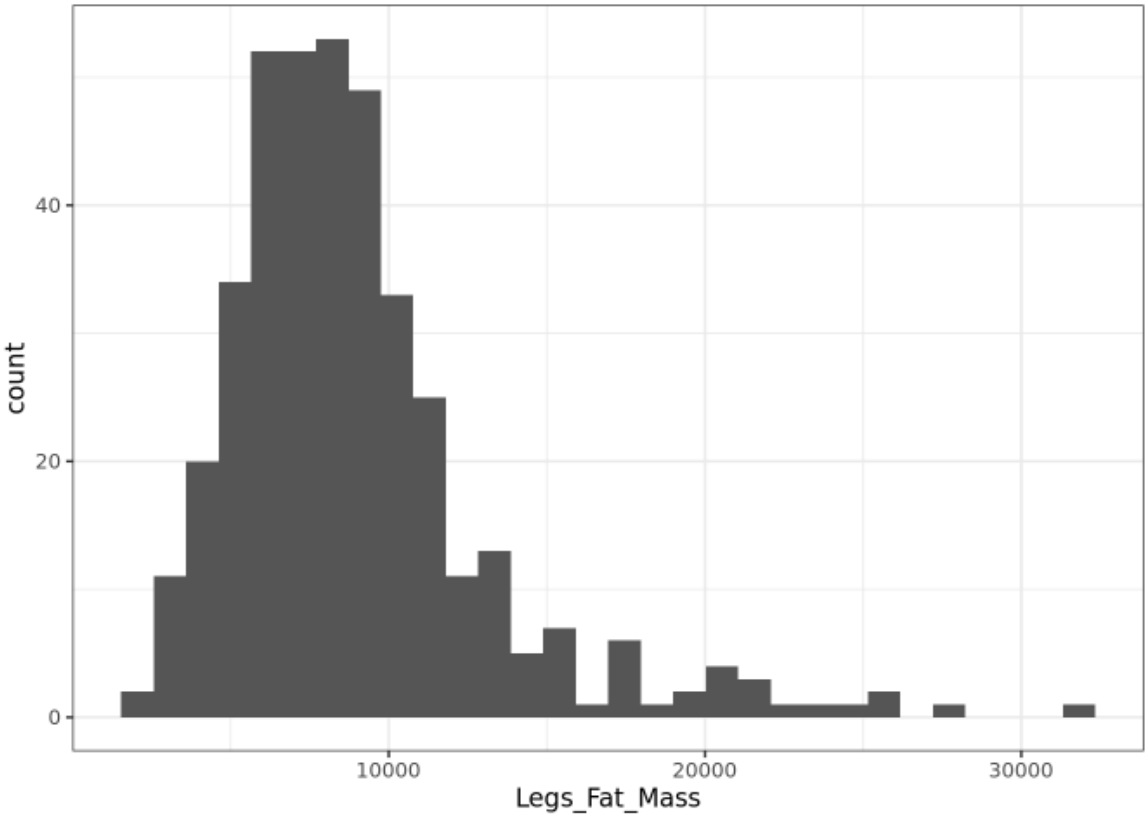


Figure E5: Distribution of android fat mass phenotype

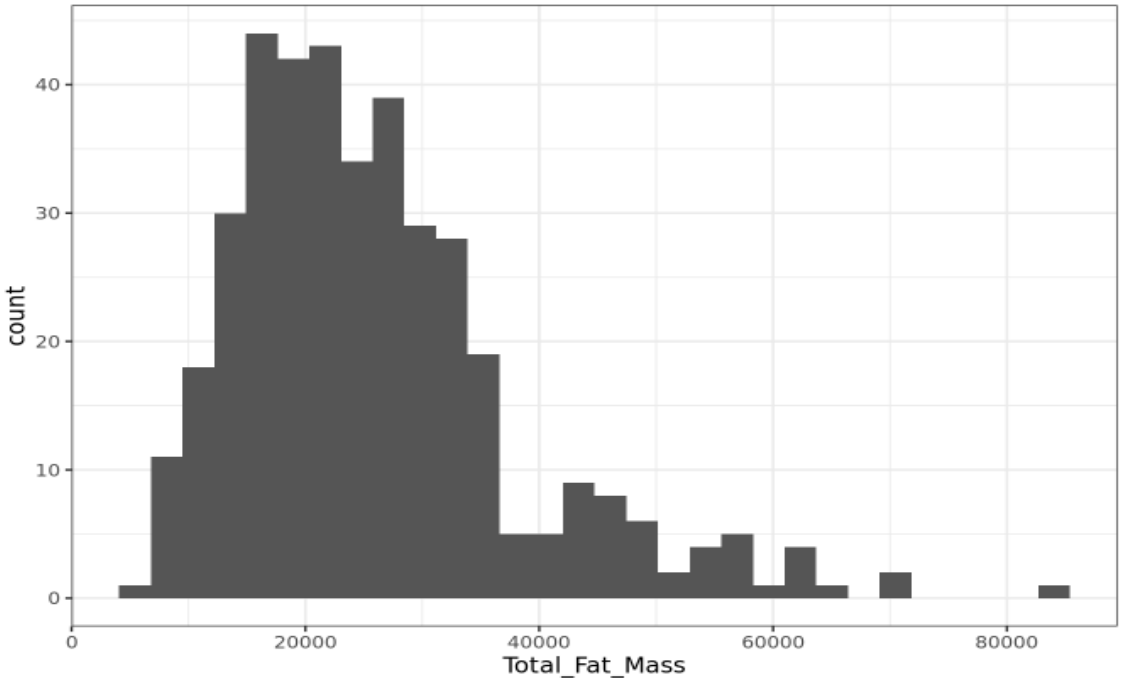


Figure E6: Distribution of android fat mass phenotype

Appendix F: Kindship coefficient

Figure F1 shows the kindship coefficient for the participants of TMPLR. It determines whether individuals in the population are related and if they come from similar ancestry.

Figure F1: Kindship coefficient from PC relate application

