# Genetic Association Study in Osteoporosis


**Author:** Gabriel Pons Mojena

**Supervisor:** Alexandre Perera i Lluna

June 30, 2016

# Resum

Dia rere dia, la tecnologia i la tècnica avancen a un ritme vertiginós, fent possible l'abaratiment de molts processos i el que aquests comporten. És el cas de la genètica, on fa menys de dues dècades encara semblava impensable que la seqüenciació i genotipació del genoma humà arribés a ser una tècnica tan accessible i barata com ho és ara. Aquest avenç ha permès un desenvolupament excepcional i un augment exponencial de la producció científica en aquest àmbit.

El mètode que es tracta en aquest treball, conegut com "Genome Wide Association Study", o el que és el mateix, estudi d'associació genètica considerant tot el genoma és conseqüència directa del que s'ha comentat en primer lloc. Això és degut a que per portar-lo a terme es necessita disposar d'una enorme quantitat de marcadors genètics coneguts com a "SNPs", que són petites variacions presents en el genoma que donen lloc a una certa variabilitat entre individus. L'objectiu del mètode al final consisteix a trobar una relació entre aquests marcadors i una malaltia complexa, fent ús d'una sèrie de fenotips numèrics o bé de presència/absència de la malaltia. Per tal de realitzar aquesta associació hi ha diverses maneres de procedir, com ara la utilització de casos i controls amb individus no relacionats o bé l'estudi de famílies on hi ha una presència inusual de la malaltia. En aquest treball el disseny experimental coincideix amb la segona opció, complicant el procediment per portar a terme l'associació des d'un punt de vista estadístic. En concret, es fa ús de models lineals mixtos, els quals tenen una matemàtica considerablement més complicada que els models lineals als que estem acostumats.

La implementació i desenvolupament del treball s'han realitzat majoritàriament en el llenguatge de programació estadística R, on a la vegada s'ha fet ús d'un software extern conegut com SOLAR especialitzat en estudis d'associació genètica en famílies. La malaltia objecte d'estudi ha estat l'Osteoporosi, per a la qual s'han intentat trobar possibles variacions genètiques que puguin explicar-ne la seva manifestació i desenvolupament. Si bé s'ha utilitzat a priori un mètode estàndard d'associació genètica, també s'ha proposat l'aplicació d'anàlisis de components principals amb l'objectiu d'incrementar-ne el poder estadístic. Tanmateix, s'ha fet un primer mapeig i valoració biològica dels resultats, deixant la porta oberta a futures investigacions amb els resultats obtinguts.

# Contents



ETSEIB

ETSEIB

# List of figures

# List of tables



ETSEIB

# 1. Introduction to the Genetic Analysis of Osteoporosis (GAO) project

## 1.1. What are complex diseases?

When we hear the word "disease", we tend to associate it with the most frequent causes, which include: bacteria, virus, cell mutation (cancer), or a very concrete genetic disorder (for instance the trisomy of chromosome 21). Although indeed these are some of the most frequent ones, there exists a group of diseases caused by multiple factors. This kind of diseases is called "complex diseases"[1], and their expression is influenced by both genetic and environmental factors. Its genetic basis is not easy though, because commonly many different loci[2] -a position or marker in the genome- are involved in the disease manifestation. Furthermore, in many occasions the genetic basis barely explains the disease or trait variability, which increases the importance of considering environmental factors. In fact, sometimes the latter get to explain rather more variability than genetics does. As a trait example for illustration we have the height[3]. Fifty four loci have been associated with this trait so far, and it's still a better predictive model for the offspring measuring parents' height.

## 1.2. The disease object of study: Osteoporosis

### Overview

Probably most of those who haven't studied any medical discipline know nothing or almost nothing about this disease, due to its low mortality. Although it's not a mortal disease, there are 8.9 million bone fractures per year due to osteoporosis[4]. Despite nobody dies from osteoporosis itself, it's not that rare to die from fractures' complications. However, it's difficult to

find an accurate definition for this condition. It has no symptoms and typically the diagnosis comes after the first osteoporotic fracture. So, what is actually osteoporosis? The World Health Organization defines it as a bone density of 2.5 standard deviations below that of a young adult. This abnormal bone loss may weaken bones to the point that a break may occur with minor stress or even spontaneously. The most common osteoporotic fractures typically occur in the vertebral column, rib, hip and wrist[5].

Those occurred in the hip deserve special attention, because they are associated with a higher risk of mortality.

### Risk factors

As we have described above, when we talked about the definition of complex diseases, Osteoporosis hasn't a unique cause or risk factor. Here are some environmental factors[4] that likely have influence in Osteoporosis expression. Most of them will be object of discussion later on, when choosing the covariates for the association study to maximize the traits' variability explained.

- High alcohol consumption

- Vitamin D deficiency

- Smoking

- Obesity

- Drugs

## 1.3. Motivation and scope of this study

The main reason for carrying out the GAO project is to know more about Osteoporosis' genetic basis considering environmental factors. The dreamed scenario would be to find a strong association between any of the genetic markers that we have already genotyped and any of the different traits related with the disease. It would be fantastic to either find a new locus or

replicate any result of another GWAS of Osteoporosis -in recent GWAS studies replicating known results has become tougher than it was thought-.

However, the size of GAO sample may affect the power of association's test and become an obstacle to find any significant association. This problem will lead to find new methods and ways to improve GWAS power, though nothing is a guaranty of finding any significant and trustable result.

Typically, a Genome Wide Association Study consists of three different parts: genotyping and obtaining of phenotypes' data, (sometimes) imputation, and association. The GAO project was already running before this final degree's project was started, and therefore the first steps were done, which means that are out of the scope of this project. Thus, the initial objectives of this project were doing an imputation of the genotyped data and carrying out a genetic association afterwards. Nevertheless, two main limitations have arisen from the very beginning: my previous knowledge in the field and the project's deadline. Due to these reasons and the fact that imputation is a very technical procedure, often assumed as something known by all genetics research groups, this project is focused on the third part.

## 1.4. Genome Wide Association Studies (GWAS)

### What are GWAS?

Genome Wide Association Studies are a very young method in genetics (the first study considered to be a GWAS was published in 2005), that as it was mentioned in section 1.3 searches the genome for any statistical significant association between a small variants of the genome called SNPs and a disease which is supposed to have a complex genetic basis[2]. Typically, there are two main ways for choosing the individuals participating in a GWAS. The former and most widely used are "cases and controls". Random individuals from the unaffected group (controls) and random individuals from the affected group (cases), as less related as possible between them, are selected for the study[6].

In the other hand, considering that often complex disorders cluster in families, the other possibility is recruiting directly extended families with several individuals affected. The latter is the way this project has been carried out, with 11 extended pedigree recruited. As explained above, in section 1.3, GWAS have one or even two important previous steps

before starting the association. The first step is genotyping the individuals participating in the study, which generally is done using a SNP array[7]. One of the keys for GWAS enhancement and development are the improvements of this kind of DNA microarrays. As long as genotyping techniques get better and the market becomes more competitive, the genotyping cost descends pretty fast, which allows research groups to increase sample sizes in GWAS.

The other highly recommended step is imputation of the missing data[8]. Even though microarrays are getting better and cheaper quite fast, these microchips are designed to provide an entire coverage of the genome by genotyping just a subset of variants, due to linkage disequilibrium [3]. Imputation allows avoiding this lack of genotyped data and some limitations of non-imputed data too. In addition, it also increases the power of the study and the number of SNPs that can be tested for association[8]. Considering that we didn't impute the genotyped data in this study, the limitations commented before will be discussed later on.

## 1.5. Software

### 1.5.1. R

R is an environment as well as programming language mainly used to carry out statistical analysis[9]. R has strong object-oriented programming facilities and is an interpreted language. Nevertheless, the most powerful feature of R are the user-created “packages”, which are built to deal with more specific statistical problems implementing complicated algorithms and techniques. Concretely, in this study we are going to use a package called “solarius” recurrently, in order to perform all the modeling analysis. This package actually is an interface to use external software called SOLAR in R. Actually, when we refer to SOLAR in this study later on, we will be referring always to “solarius”, because the analysis is done in R. In the next section we explain what’s exactly SOLAR and its importance.

### 1.5.2. SOLAR

In order to define SOLAR, we have quoted the introduction that can be found in its web page[10]:

*"SOLAR-Eclipse is an extensive, flexible software package for genetic variance components analysis, including linkage analysis, quantitative genetic analysis, SNP association analysis (QTN and QTLD), and covariate screening. Operations are included for calculation of marker-specific or multipoint identity-by-descent (IBD) matrices in pedigrees of arbitrary size and complexity, and for linkage analysis of multiple quantitative traits and/or discrete traits which may involve multiple loci (oligogenic analysis), dominance effects, household effects, and interactions. Additional features include functionality for mega and meta-genetic analyses where data from diverse cohorts can be pooled to improve statistical significance."*

Actually, this software is especially made to deal with related individuals, concretely with complex pedigrees, as it is in our case.

### 1.5.2. PLINK

PLINK is an open-source software made by Harvard researchers in bioinformatics used to perform genome statistical analysis[11]. It has a wide range of features especially conceived to deal with genetics data. Despite it has so many possibilities (a genetic association study can be completely carried out in PLINK) we are going to use it only to apply a quality control to our data. In fact, PLINK comes with several useful filters implemented that will be applied to our data, in order to increase the rigour and consistency of our final results.

# 2. Genome structure and SNPs

The aim of this section is to provide a basic genetic context to the reader, in order to facilitate the understanding of this project, as well as the objectives pursued. Primarily we are going to explain the genome's structure and the basic functionalities of the elements integrated in it (as the protein encoding) emphasizing the SNPs' role and the variability caused by them.

The human genome is the complete sequence of DNA (deoxyribonucleic acid) contained in 23 pairs of chromosomes (22 autosomal and 1 depending on the sex) in cell nuclei as well as a small DNA molecule present in the mithocondria (cellular organelles responsible of the energy obtain)[12]. All the necessary information that makes possible the correct development and functionality of a human individual is encoded in the DNA sequence. This information is mainly

the “proteome” which is the entire set of proteins that cover all the functions needed to live. In fact, proteins develop a wide range of functions; structural, enzymatic, metabolic, regulative, signaling, and so on, forming a highly complex interactive network. Nevertheless, the DNA sequence contains many elements that are not especially dedicated to protein-coding. In fact, we should split the sequence into two main types: intragenic and intergenic.

## Intragenic DNA

In the one hand, the intragenic DNA contains the protein-coding genes, the RNA (ribonucleic acid) coding genes, regulation sequences and pseudogenes.

There are about 20,000-25,000 human protein-coding genes, and surprisingly they are a small percentage of the overall sequence (1,5%). The protein-coding genes consist of two main parts: a promoter sequence that regulates its expression and a transcription sequence, integrated by UTR sequences (necessaries for the well translation and stability of the RNAm), exons (that codify) and introns (which are eliminated in the transcription process).

Furthermore, the RNA coding genes are responsible of several types of RNA transcription. These types include: the transference RNA (RNAt), the ribosomal RNA (RNAr), the microRNA and other non-coding genes. The first two ones are essential on the proteins transcription and ribosomes constitution, whereas the third one have demonstrated a main role in gene regulation expression.

The regulation sequences are typically short sequences either near or within the genes. In case of those regulation sequences within the gene, they are typically placed in the introns. Actually, we barely know how these regions work, though it’s known that their importance is crucial.

Finally, the pseudo genes are mutated versions of existing genes, that due to their level of mutation they can’t be transcribed.

## Intergenic DNA

These non-coding regions represent the greatest part of the genome, and its function is mainly still unknown. Most of these sequences are composed by repetitive elements, though some of

them don't have a clear and classification pattern. Nevertheless, their importance in gene expression and regulation has been proved.

### The SNPs

So far, we have given an outline of the genome's structure in order to arrive here and understand the importance and different types of existing SNPs.

SNP means "single nucleotide polymorphism" and it's exactly what they are, a variation of a single base. However, their importance resides in their high contribution to phenotypic variability among individuals. In fact the main source of variability in the human race is due to variations of a single nucleotide. Nevertheless, only those variations which their minor allele is present at least 1% within a population can be considered SNPs[13]. If the presence of the minor allele within a population is below this threshold, these variations are considered rare variants or mutations.

Actually, we have presented the structure of the human genome just before this section to state that SNPs can be found in any region of the genome. This fact implies that there are two main types of SNPs: silent SNPs which are placed in non-coding regions and SNPs that may directly interfere with protein-coding. However, the importance of both types could be crucial, with the difference that those placed in the non-coding region would be difficult to interpret, and despite they may be hiding an underlying unknown effect.

It's important to keep in mind that SNPs have survived to natural selection in one way or another because they have still presence within the actual population. From the disease's point of view, in general a single SNP rarely can affect lethally an individual, although they can have a moderately negative effect.

# 3. Materials: GAO Data

## 3.1. Sample description

The GAO project involved 367 individuals from eleven extended Spanish families. To be considered "extended pedigree", a family had to have at least ten living individuals distributed in

three or more generations[4]. The aim of having the individuals clustered in families is to enhance power of “rare variants”. In case there is any rare marker with high presence among the individuals of these families with high incidence of Osteoporosis, we are more likely to find an association with it. In fact, large pedigrees have comparably more power per sampled individual than smaller families, compensating for small sample sizes, as it is in this case[4].

| FAM | N of individuals | Age (mean) | Sex |
|---|---|---|---|
| gao10 | 57 | 37,63 | 1:33 2:24 |
| gao11 | 91 | 39,47 | 1:33 2:58 |
| gao12 | 23 | 42,35 | 1:13 2:10 |
| gao13 | 34 | 49,53 | 1:17 2:17 |
| gao14 | 19 | 36,47 | 1:10 2:9 |
| gao15 | 31 | 35,97 | 1:12 2:19 |
| gao16 | 22 | 43,82 | 1:11 2:11 |
| gao17 | 15 | 50,4 | 1:7 2:8 |
| gao18 | 30 | 32,87 | 1:19 2:11 |
| gao19 | 30 | 50,3 | 1:13 2:17 |
| gao20 | 15 | 37,73 | 1:9 2:6 |

*Table 3.1. Basic description of the sample*

## 3.2. Genotyped data

### 3.2.1. Blood collection and DNA extraction

The DNA used to genotype the individuals was extracted from white blood cells using a standard salting out procedure after separating and storing the plasma for future biochemical assays. The blood used for the DNA extraction was collected in samples of 35 ml of peripheral venous blood from all of the participants in[14]:

1. Citrate-containing tubes for DNA extraction,

2. PAXgene® Blood RNA Tubes (PreAnalytiX GmbH, Hombrechtikon, Switzerland) for storage and further RNA analysis

3. Heparin-containing tubes for further whole blood assays.

### 3.2.2. Genotyping

In this study the SNPs were genotyped by Real-Time PCR, using the standard TaqMan® SNP genotyping assay protocol of Applied Biosystems for a total volume of 5 µl per well. Fluorescence intensity measurements of the final reaction product and data collection were carried out on an Applied Biosystems 7500/7500 Fast Real-Time PCR System[4].

The subset of SNPs chosen determines the coverage provided of the entire genome. In fact, genome-wide association studies rely on linkage disequilibrium and because of this the reported association variants are unlikely to be the actual causal variants. We would need a very dense coverage to find exactly the actual variant directly in the association study. In this project we have 964.193 independent SNPs genotyped in the raw files, and since we are not going to impute the data the final number of SNPs used in the association will be below this amount of markers due to the further quality control. Actually, we are going to use only the SNPs from the autosomal chromosomes. A summary table can be found below these lines:

| Chromosome | N of SNPs |
|---|---|
| chr1 | 83456 |
| chr2 | 74455 |
| chr3 | 61332 |
| chr4 | 50299 |
| chr5 | 53103 |
| chr6 | 63077 |
| chr7 | 48878 |
| chr8 | 45627 |
| chr9 | 42781 |
| chr10 | 48287 |

| | |
|---|---|
| chr11 | 51967 |
| chr12 | 47593 |
| chr13 | 31948 |
| chr14 | 30817 |
| chr15 | 29751 |
| chr16 | 32977 |
| chr17 | 33153 |
| chr18 | 25296 |
| chr19 | 29981 |
| chr20 | 24732 |
| chr21 | 12994 |
| chr22 | 15554 |
| **Total** | **938058** |

*Table 3.2. Number of SNPs per chromosome*

The file encoding of the genotypes is explained in the appendix section A.1.1.

## 3.3. The phenotypes

### 3.3.1. Overview

The tables of phenotype include three different types of traits, as well as additional variables that may be useful for further modeling analysis and a set of mandatory variables requested by the software used in this study.

The different types of traits that we have are: plasma levels of bone metabolism markers related with Osteoporosis, densitometric traits (as Bone Mineral Density), and binary “affected” traits defined following different criteria.

The additional variables contain relevant information about the environment and lifestyle of the individuals participating in the study.

The mandatory variables contain relevant information of the kinship relations among individuals, which is absolutely necessary considering that we are carrying out an association study with extended pedigree. These variables will be extendedly commented in methodology, concretely when we explain the kinship matrix.

### 3.3.2. Bone metabolism markers

The plasma levels of 12 bone metabolism markers have been obtained from the samples of 35ml of blood contained in the heparin-containing tubes from the participants[14]. The analysis of each marker has been carried out in a different place and way. This is summarized below:

1. Sclerostin (Biomedica Medizinprodukte GmbH, Wien), IGF1 (Mediadiagnost, Reutlinger, Germany), Serum Crosslaps and OstaseBAP (Immunodiagnostic Systems Ltd, Fountain Hills, AZ) were analyzed by ELISA
2. 25-hydroxy vitamin D ( Immunodiagnostic Systems Ltd ) was analyzed by EIA
3. Adiponectin, Leptin, TNFalpha, Osteoprotegerin, Osteocalcin, Osteopontin and Parathyroid hormone were analyzed in a Luminex using xMAP® technology (Millipore Corporation, Billerica, MA) following the manufacturer's instructions.

There are also 10 additional variables accounting for external factors of the individuals that are interesting from the modeling perspective.

A summary table of the different variables and a brief description of each one is attached below these lines. Nevertheless, the biology of the markers included in the table is briefly described in the appendix section A.1.2.

| Variables | NAs | Mean | Sd | Frequency | Units | Description |
|---|---|---|---|---|---|---|
| Age | 0 | 4,082834e+01 | 2,095938e+01 | - | Years | Patient age (year of last update - year of birth) |
| Sex | 0 | - | - | 1:177<br>2:190 | - | Patient gender |
| MenopAge | 307 | 4,821667e+0 | 4,621584e+ | - | Years | Menopause age |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 00 | | | for women |
| Alcohol | 8 | - | - | 1:343 2:16 | - | Alcohol intake >30g/day (0=No, 1=yes) |
| Smoking | 13 | - | - | 1:215 2:139 | - | Smoking habit (0=Non-smoker; 1=currently or previously a smoker) |
| SolarExp | 1 | 6,008197e+00 | 6,496991e+00 | - | min/day | Daily solar exposure (min/day) |
| PosDrug | 1 | - | - | 1:319 2:47 | - | Drug administration with a positive effect on the development of osteoporosis |
| NegDrug | 1 | - | - | 1:331 2:35 | - | Drug administration with a negative effect on the development of osteoporosis |
| IPAQ | 109 | 3,411711e+03 | 4,297056e+03 | - | MET-minutes/ week | Total physical activity MET-minutes/week = sum of Walking + Moderate + |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | Vigorous MET-minutes/week scores. |
| Calcium | 0 | 3,405995e+03 | 2,076233e+03 | - | mg/week | Total calcium intake |
| HydVitD | 0 | 2,183161e+01 | 1,020009e+01 | - | ng/mL | Density of 25-hydroxy vitamin D |
| Sclerostin | 0 | 3,032332e+01 | 1,019868e+01 | - | pmol/L | Concentration of sclerostin |
| SerCrossLaps | 1 | 3,415027e-01 | 3,834939e-01 | - | ng/mL | Density of serum beta-crosslaps, a bone turnover marker |
| OstaseBAP | 0 | 2,564635e+01 | 2,792830e+01 | - | ug/L | Density of ostase bone-specific alkaline phosphatase |
| IGF1 | 0 | 2,177134e+02 | 1,165206e+02 | - | ng/mL | Density of insulin-like growth factor 1 |
| Adiponectin | 0 | 2,913246e+07 | 3,910357e+07 | - | pg/mL | Density of adiponectin |
| Leptin | 0 | 6,000868e+03 | 7,243827e+03 | - | pg/mL | Density of leptin |
| Osteocalcin | 0 | 1,668874e+04 | 1,745401e+04 | - | pg/mL | Density of osteocalcin |
| Osteoprotegerin | 0 | 2,414776e+02 | 1,368011e+02 | - | pg/mL | Density of osteoprotegerin |
| Osteopontin | 0 | 1,939970e+04 | 1,536175e+04 | - | pg/mL | Density of osteopontin |

| | | | | | | |
|---|---|---|---|---|---|---|
| Parathyroid | 0 | 2,649147e+01 | 4,990015e+01 | - | pg/mL | Density of parathyroid hormone |
| TNFalpha | 0 | 6,956676e-01 | 6,028639e-01 | - | pg/mL | Density of tumor necrosis factor-alpha |

*Table 3.3. Summary of the bone metabolism markers' table*

## 3.3.3. Densitometric and affected phenotypes

In total we have 23 densitometric traits of high clinical interest and 4 binary "Affected" traits. However, there are 8 additional traits that are derived from the other 23. In fact, these 8 phenotypes are the T and Z scores, which are measures that compare levels of bone mineral density of the patient with those of a healthy individual. Furthermore, we have additional variables that may be useful in further steps of this study.

### Densitometric phenotypes

The densitometric traits of spine, femur and whole body for all participants were obtained using a Discovery DXA system (Dual X-ray absorptiometry technique) with the APEX v2.3 software (Hologic, Bedford, MA, USA), following the manufacturer's recommendations[4]. Although the analysis is restricted to two dimensions, and the resolution of the structural dimensions is low, it seems an acceptable approach to analyze strength and geometrical properties, with the advantages of a relatively low cost and small radiation dose compared to quantitative computed tomography.

In order to analyze strength and geometrical properties of the hip, was used the hip structural analysis (HSA) software included in APEX[4]. Scans were performed and reviewed by the same technician and physician respectively, both of them certified either by the International Society for Clinical Densitometry or by the manufacturer of the densitometer.

The phenotypes include:

1. The bone mineral density (g/cm2) of hip, trochanteric line, intertrochanteric line, femoral neck, spine and whole body;
2. Hip axis length (mm)
3. Femoral neck - femoral shaft angle (degrees)
4. Measures of bone strength, as: average cortical thickness (ACT; cm), buckling ratio (BR; cm3), cross-sectional area (CSA; cm2), (iv) cross- sectional moment of inertia (CSMI; cm4) and section modulus (SM; cm3) of femoral shaft, narrow neck and intertrochanteric line.

## Affected phenotypes

Considering that Osteoporosis is a complex disease difficult to define clearly, four different definitions[4] have been used to deal with this problem in this study, nevertheless isn't still clear which one of them is the best approach for the disease.

The summary table of the densitometric and affected traits, as well as the additional variables is attached below:

| Variables | NAs | Mean | Sd | Frequency | Units | Description |
|---|---|---|---|---|---|---|
| Affected1 | 5 | - | - | 0:292 1:70 | - | Osteoporosis, common definition: Older than 21 years old AND [T-score < -2.5 (column, hip neck or total hip) OR atraumatic fracture OR treatment with bisphosphonates] |
| Affected2 | 5 | - | - | 0:338 1:24 | - | Atraumatic fracture |
| Affected3 | 4 | - | - | 0:157 1:206 | - | Osteoporotic traits (osteoporosis + osteopenia, common definition): Older than 21 years old AND [T-score < -1 (column, hip neck or total hip) OR atraumatic fracture |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | OR treatment with bisphosphonates] |
| Affected4 | 5 | - | - | 0:288 1:74 | - | Osteoporosis, children included: {Older than 21 years old AND [T-score < -2.5 (column, hip neck or total hip) OR atraumatic fracture OR treatment with bisphosphonates]} AND {Younger than 21 years old AND [Z-score < -2.5 (column, hip neck or total hip) OR atraumatic fracture OR treatment with bisphosphonates]} |
| AxisLen | 3 | 107,10372798 | 10,676076683 | - | mm | Hip axis length; distance from pelvic rim to outer margin of greater trochanter along neck axis |
| FemShACT | 4 | 0,616521124 | 0,1463527373 | - | cm | Average cortical thickness of femoral shaft calculated as: (femoral shaft width - femoral shaft endocortical diameter)/2 |
| FemShBR | 4 | 2,427790 48 | 0,6181417474 | - | cm^3 | Buckling ratio of femoral shaft, i.e. the relative thickness of femoral shaft cortex as an estimate of cortical stability in buckling |
| FemShCSA | 4 | 4,16436281 | 1,059929 67 | - | cm^2 | Cross sectional area of femoral shaft |
| FemShCSMI | 4 | 3,14059947 | 1,379289 18 | - | cm^4 | Cross sectional moment of inertia of femoral shaft; index of structural rigidity; reflects |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | distribution of mass about the center of a structural element |
| FeShSMod | 4 | 2,12159240 | 0,72344390 | - | cm^3 | Section modulus of femoral shaft; indicator of bending strength for maximum bending stress in the image plane |
| HipNeckT | 4 | -1,11983471 | 1,0649062 8 | - | t score | t-score of the bone mineral density of femoral neck (number of standard deviations above or below the mean for a healthy 30 year old adult of the same sex and ethnicity as the patient) |
| HipNeckZ | 8 | -0,28189415 | 0,9552447 3 | - | z-score | z-score of the bone mineral density of femoral neck (number of standard deviations above or below the mean for the age, sex and ethnicity of the patient) |
| HipTotBMD | 3 | 0,91049867 | 0,1437997 68 | - | g/cm^3 | Total bone mineral density of hip |
| HipTotT | 4 | -0,56473829 | 1,0023297 68 | - | t-score | t-score of the bone mineral density of hip (number of standard deviations above or below the mean for a healthy 30 year old adult of the same sex and ethnicity as the patient) |
| HipTotZ | 8 | 0,13008357 | 0,9112197 9 | - | z-score | z-score of the bone mineral density of hip (number of standard deviations above or below the mean for the age, sex and ethnicity of the patient) |

| | | | | | | |
|---|---|---|---|---|---|---|
| InterBMD | 3 | 1,07366305 | 0,178070 56 | - | g/cm^3 | Bone mineral density of intertrochanteric area (g/cm^3) |
| IntTrACT | 4 | 0,40524260 | 0,083239 56 | - | cm | Intertrochanteric average cortical thickness calculated as: (intertrochanteric width - intertrochanteric endocortical diameter)/2 |
| IntTrBR | 4 | 7,71376937 | 1,849273 71 | - | cm^3 | Intertrochanteric buckling ratio, i.e. the relative thickness of intertrochanteric cortex as an estimate of cortical stability in buckling |
| IntTrCSA | 4 | 4,67077525 | 1,166120 25 | - | cm^2 | Intertrochanteric cross sectional area |
| IntTrCSMI | 4 | 12,09169816 | 5,030004 93 | - | cm^4 | Intertrochanteric cross sectional moment of inertia; index of structural rigidity; reflects distribution of mass about the center of a structural element |
| IntTrSMod | 4 | 3,88861587 | 1,331292 82 | - | cm^3 | Intertrochanteric section modulus; indicator of bending strength for maximum bending stress in the image plane |
| NeckBMD | 3 | 0,75099010 | 0,132235 58 | - | g/cm^3 | Bone mineral density of femoral neck (g/cm^3) |

| | | | | | | |
|---|---|---|---|---|---|---|
| NNeckACT | 4 | 0,17074681 | 0,029849 83 | - | cm | Average cortical thickness of narrow neck calculated as: (narrow neck width - narrow neck endocortical diameter)/2 |
| NNeckBR | 4 | 10,94453652 | 2,584777 53 | - | cm^3 | Buckling ratio of narrow neck, i.e. the relative thickness of narrow neck cortex as an estimate of cortical stability in buckling |
| NNeckCSA | 4 | 2,80308640 | 0,615174 39 | - | cm^2 | Cross sectional area of narrow neck () |
| NNeckCSMI | 4 | 2,78796462 | 1,124506 29 | - | cm^4 | Cross sectional moment of inertia of narrow neck; index of structural rigidity; reflects distribution of mass about the center of a structural element |
| NNeckSMod | 4 | 1,49993516 | 0,469344 30 | - | cm^3 | Section modulus of narrow neck; indicator of bending strength for maximum bending stress in the image plane |
| ShaftNeck | 4 | 126,04709468 | 5,631400 39 | - | degrees | Femoral neck - shaft angle |
| SpineT | 3 | -1,68736264 | 1,599106 04 | - | t-score | t-score of the bone mineral density of spine (number of standard deviations above or below the mean for a healthy 30 year old adult of the same sex and ethnicity as the patient) |
| SpineZ | 9 | - | 1,26 | - | z- | z-score of the bone mineral |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | 0,5916 2011 | 7007 39 | | score | density of spine (number of standard deviations above or below the mean for the age, sex and ethnicity of the patient) |
| TotBMD | 2 | 0,8817 9108 | 0,17 7021 61 | - | g/cm^ 3 | Total bone mineral density of spine |
| TrochBMD | 3 | 0,6526 7482 | 0,10 5718 15 | - | g/cm^ 3 | Bone mineral density of trochanteric area |
| WBTotBMD | 2 | 1,0503 5345 | 0,13 8909 82 | - | g/cm^ 3 | Total bone mineral density of the whole body () |
| WhBodyT | 4 | - 1,1986 2259 | 1,65 9485 80 | - | t-score | t-score of the bone mineral density of the whole body (number of standard deviations above or below the mean for a healthy 30 year old adult of the same sex and ethnicity as the patient) |
| WhBodyZ | 24 | - 0,3766 7638 | 1,02 8843 40 | - | z-score | z-score of the bone mineral density of the whole body (number of standard deviations above or below the mean for the age, sex and ethnicity of the patient) |
| BMI | 2 | 2,3945 67e+01 | 4,48 0787 e+0 | - | Kg/m^ 2 | Patient body mass index |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | 0 | | | |
| Coffee | 10 | - | - | 1:123 2:165 3:54 4:15 | cups/ day | Coffee intake |
| Diabetes | 2 | - | - | 0:353 1:12 | - | Diabetes Mellitus comorbidity (0 = no; 1 = yes) |
| ERT | 2 | - | - | 0:363 1:2 | - | Estrogen Replacement Therapy in female subjects (0 = no; 1 = yes) |
| Height | 2 | 1,621863e+02 | 1,351166e+01 | - | cm | Patient height |
| Medication | 2 | - | - | -1:21 0: 311 1:33 | - | Drug administration with a positive (1) or negative (-1) effect on the development of osteoporosis. If a patient takes both types of medication, then their effects cancel (0) |
| VitD | 2 | - | - | 0:342 1:23 | - | Vitamin D treatment (0 = no; 1 = yes) |
| WBTotArea | 2 | 1,873226e+03 | 3,146674e+02 | - | cm^2 | Total bone area of the whole body |
| WBTotBMC | 2 | 2,001517e+03 | 5,394567e+02 | - | g | Total bone mineral content of the whole body |
| WBTotFat | 2 | 1,804838e+04 | 7,455101e+0 | - | g | Total fat content of the whole body |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | 3 | | | |
| WBTotLean | 2 | 4,452018e+04 | 1,240816e+04 | - | g | Total lean content of the whole body () |
| Weight | 2 | 6,392137e+01 | 1,666956e+01 | - | Kg | Patient weight |

*Table 3.4. Summary of the densitometric and affected phenotypes table*

The table of clinical phenotypes that has been summarized above was stored in the SQL database of the Biomedical's Research Institute of the "Hospital de la Santa Creu i Sant Pau" and it was extracted using two scripts that can be found in the annex A.1.3.

# 4. Methodology

## 4.1. Data pre-processing

### 4.1.1. Overview

The data pre-processing consisted in three main steps, which are:

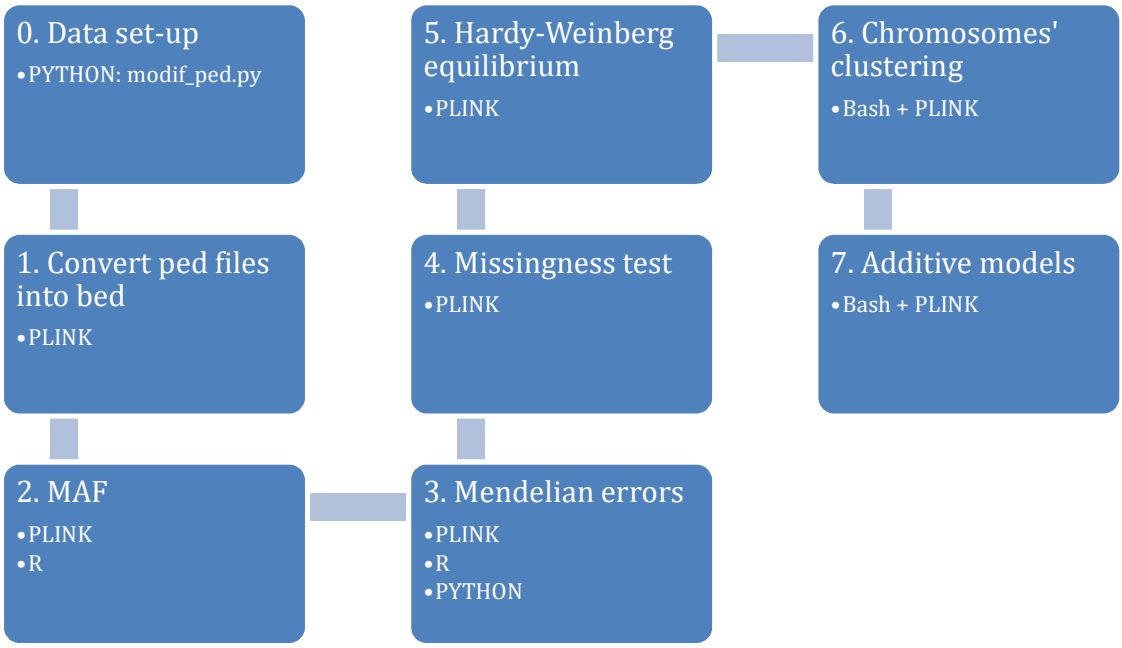
1. Preparation of the data for further quality control
2. The quality control itself
3. Preparation of the data for the further association analysis.

The first step was necessary in order to make possible some analysis done in the QC as well as guarantee that the QC is unbiased. It was also important to enhance the QC performance in computational efficiency terms.

The quality control was also necessary especially from the genetics point of view. The final results obtained in this study have to be consistent with the genetics not just with mathematics. Thus, in order to guarantee its genetic consistency several filters have been applied to the data, which are: Minor Allele Frequency, Mendelian errors and Hardy Weinberg equilibrium. Furthermore, a missingness test has been also performed, to avoid bias in the association analysis.

Finally, it was crucial to let the data prepared for the association analysis, especially from the computational point of view. We clustered the data in small batches to improve computation efficiency as well as transformed the genotypes into numeric data, to make possible the mathematical treatment of the data.

These three steps have been performed using PLINK and R. We have attached a workflow diagram below, with the main operations and the software used in each case.



## 4.1.2. Data set-up for the QC

### 4.1.2.1. Modification of ped file


There are two basic points that have to be modified from this file in order to assure an unbiased and efficient quality control:

1. There are some individuals in this file who actually don't belong to this study. Therefore, we have no phenotypic information for these individuals, and no association can be done. Thus, these individuals have to be removed from that file to guarantee that the QC is unbiased. The script compares the IDs in the pedigree file with those present in the table of phenotypes, and remove the unmatched individuals.

2. On the other hand, as we introduced in section 3.2, there is an important lack of information about the pedigree, which is needed to perform some of the steps of this QC. Concretely, the fields “father” and “mother” in the genotypes are empty, and they are necessary to perform further Mendelian errors analysis. The python script solves this problem.

The code in python of the script used as well as the explanation of how it actually works can be found in the annex A.2.1.

### 4.1.2.2. Binary ped files (*.bed)

Although map and ped files have a very clear structure and are quite easy to understand, working with them is actually very inefficient in PLINK. To save space and time, making a binary file is a highly recommended idea. This procedure splits the data into three different files: *.bed,* .fam and *.bim[15].

The first two files carry the information that was previously contained in the ped file. In the one hand, the bed file is a compressed file which contains the genotypes of every individual. On the other hand, the fam file contains the first six columns of the ped file. The third file is an extended map file, which means that apart from carrying the data of the map file, now it also carries information about the allele names, which would be otherwise be lost in the bed file.

The commands used for this file conversion can be found in the appendix section A.2.1.

## 4.1.3. Quality Control[16]

### 4.1.3.1. Minor Allele Frequency (MAF)


Following the SNP's definition, they are variations in the genome that have to be present in at least 1% of individuals within a population. Variations with prevalence below this threshold are considered specific mutations instead of SNPs. Thus, considering that the minor allele frequency is the lowest allele frequency at a locus that is observed in a particular population[17], applying a MAF threshold prevents from including very rare markers in further association analysis. Considering that all markers are biallellic, the MAF is calculated as:

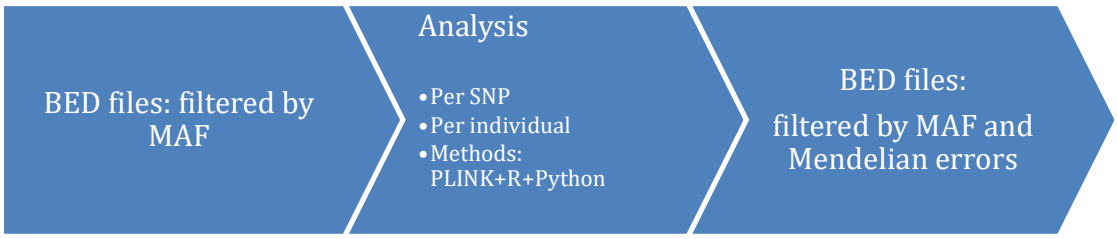
$$MAF = \frac{Number\ of\ minor\ Alleles}{2 * Number\ of\ individuals} \quad (4.1)$$

In this case, we pruned those SNPs whose MAF was below 1%[18]. The total amount of SNPs pruned can be found in the summary table of the QC in the Results' section.

#### Implementation

To filter by MAF we used both PLINK and R. With the first software we generated a file (*.frq) containing all minor allele frequencies for all SNPs[19] (its structure can be found in the appendix) and with R we pruned those markers which were below the chosen threshold. The names of the SNPs pruned have been imported to a file and removed from the original file using PLINK. The code can be found in the annex A.2.2.

### 4.1.3.2. Mendelian errors




A Mendelian error describes an allele in an individual which could not have been received from either of its biological parents by Mendelian inheritance[20]. There are different types of Mendelian errors, each one of them has its corresponding code in PLINK.

Furthermore, there are two different ways to evaluate and filter by Mendelian errors. In the one hand, we can look at them from the SNPs perspective. From this point of view, we prune those SNPs in which the percentage of individuals who present a Mendelian error at that locus is above our threshold.

In the other hand, we can look at the amount of Mendelian errors that each individual presents. This case is such more complicated and controverted, because we can start to discuss if beyond a certain number of Mendelian errors per individual we are not just facing a rare case fruit of randomness. What we are meaning with this, is that in one way or another familiar structure is wrong, and therefore it is easy to infer that either the supposed father or mother actually is not the biological parent.

To carry out both analysis we have used the files: .*.lmendel, *.mendel, *.imendel and *.fmendel generated in PLINK with the command –mendel[21]. The structure of these files is detailed in the annex section A.2.2

### Analysis of Mendelian errors per SNP

In the first place if we filter SNPs by Mendelian errors, the file of interest is.*.lmendel. This file contains the number of Mendelian errors per SNP, which is the same to say the number of individuals who present a Mendelian error at that locus. We have applied this filter using PLINK and R to compare both results, which have to be the same.

The steps followed to filter by Mendelian errors in R have been:

1. Convert the file of interest into the correct format (*.csv) through a python script.
2. Apply a vectorized operation pruning those SNPs that are above the threshold we have chosen. In this case, the threshold has been set at 1%. Nevertheless, we have to consider that some individuals in the sample are founders, and therefore we can't calculate Mendelian errors for them. The consequence of this fact is that PLINK computes the proportion using only non-founders. So if we want to obtain the same results, we have to follow the same criteria in R. There are 281 non-founders, so 1% out of 281 would be 2.81 individuals presenting a Mendelian error. Due to the fact that the number of individuals is a discrete variable, those SNPs with more than 2 individuals presenting a Mendelian error at that locus will be pruned.

Afterwards, we have performed the filtering operation using PLINK, in order to compare with the results obtained in R.

An important observation that should be remarked is that we can apply this filter after pruning by MAF because we are focusing on SNPs. When we analyze the Mendelian errors per individual, it doesn't make sense to eliminate SNPs previously, since we might be eliminating potential discrepancies.

Both PLINK and R command lines can be found in the annex section A.2.2

### Analysis of Mendelian errors per individual

To carry out this analysis the only possibility was doing it in R. In this case, we are interested in the files: *.mendel, *.imendel and *.fmendel (their structure is explained in the annex). The procedure has been the following one:

1. Convert the files of interest into the correct format (*.csv) through a python script
2. Through an R script written for this particular problem (code in the annex A.2.2.), we extracted the IDs of the offspring of the nuclear families that were in the *.fmendel file. Due to the fact that in this file we just had the parents' IDs, a file with complete pedigree information was needed. In total, we extracted 171 IDs.
3. Using the same script, we obtained the Mendelian errors for each ID extracted before. Afterwards, we have plotted several distribution figures to identify possible outliers. The resultant plots are in the section Results.
4. At last, we have analyzed some outliers in detail, using a script written in python for this occasion too (annex A.2.2.). The script, given an individual and one or more types of Mendelian errors, computes how many Mendelian errors of those types the individual present.

   We have attached a table with the different types of Mendelian errors and their corresponding codes [21]:

| Code | Pat | Mat | Offspring |
|---|---|---|---|
| 1 | AA | AA | AB |
| 2 | BB | BB | AB |
| 3 | BB | ** | AA |
| 4 | ** | BB | AA |
| 5 | BB | BB | AA |
| 6 | AA | ** | BB |
| 7 | ** | AA | BB |
| 8 | AA | AA | BB |
| 9 | ** | AA | BB |
| 10 | ** | BB | AA |

*Table 4.1. Types of mendelian errors and their codification. The asterisks mean that the Mendelian error is independent from the genotypes of that individual.*

Finally, to analyze the different types of errors per individual, we have clustered some types of Mendelian errors[21]:

- Errors 1 and 2 affect the whole trio
- Errors 5 and 8 affect only the child
- Errors 3 and 6 affect both child and father
- Errors 4, 7, 9 and 10 affect both child and mother

### 4.1.3.3. Missingness test


The missingness test otherwise known as "call rate test" consists in identifying those loci whose number of missings is above the threshold established. If there are more individuals than allowed presenting a missing in the locus in question, that SNP is automatically pruned. In this case, we set the threshold to 2%, which considering our sample size of 367 individuals implies pruning those SNPs which have more than 7 missings. The procedure to apply this filter is almost the same we have been following so far in previous sections.

1. First, we generate in PLINK two text files through the command –missing[22], one containing missingness per individual and the other containing missingness per SNP.

Since we are not interested in discarding individuals, we are going to focus only in the second one (*.lmiss).

2. Before importing this file in R, we need to convert it to a suitable format (*.csv)
3. Afterwards, we apply a vectorized filter setting the threshold to 7 missings per SNP, and we export those SNPs which failed the test to a text file.
4. Finally, we extract from the original file the SNPs that failed the test using PLINK.

The file structure and the code can be found in the annex A.2.2..

#### 4.1.3.4. Hardy Weinberg equilibrium test


First, before starting to explain the test itself, we need to define the Hardy Weinberg principle[23]. This law states that allele and genotype frequencies will remain constant from generation to generation if none of the following assumptions is violated:

- Infinite population

- Discrete generations

- Random mating

- No natural selection is taking place in the population

- No migration (to avoid genetic flow, which is the transfer of alleles or genes from one population to another)

- No mutation

- Equal initial genotype frequencies in both sexes

The equilibrium is reached after one generation of breeding under the assumptions from above. To put it in mathematical terms, let's consider a parametrical example. Given the frequencies u, v and w for the genotypes AA, Aa and aa respectively, we can directly deduce alleles frequencies, which are $P(A) = u + (1/2) * v$ and $P(a) = w + (1/2) * v$. Considering the different mating possibilities and with genotypes frequencies in hand, we can easily arrive to table 4.2.

| **Parents genotypes** | **Mating frequency** | **Expected offspring** |
|---|---|---|
| AA x AA | $u^2$ | AA |
| AA x Aa | $2 * u * v$ | $(1/2) * AA + (1/2) * Aa$ |
| AA x aa | $2 * u * w$ | Aa |
| Aa x aa | $2 * v * w$ | $(1/2) * Aa + (1/2) * aa$ |
| Aa x Aa | $v^2$ | $(1/4) * AA + (1/2) * Aa + (1/4) * aa$ |
| aa x aa | $w^2$ | aa |

*Table 4.2. Mating frequencies and expected offspring for the different genotypes combination*

From table 4.2. we can calculate the genotypes frequencies for the offspring, which will remain constant from now on. So a certain locus is in Hardy Weinberg equilibrium if the genotypes frequencies are the following ones:

$$P(AA) = (u + (1/2) * v)^2 = P(A)^2$$
$$P(Aa) = 2 * (u + (1/2) * v)((1/2) * v + w) = 2 * P(A) * P(a)$$
$$P(aa) = (w + (1/2) * v)^2 = P(a)^2$$

So far so good, but we need to apply these concepts to our data in order to prune those SNPs which are not in Hardy Weinberg equilibrium. As always in this QC, we are going to apply this filter using PLINK, which performs a Chi-Square goodness of fit test. The test statistic used for it is obtained using some of the concepts presented so far. To obtain the test statistic for a locus, we first need to construct the following table:

| **Genotype** | **Observed** | **Expected** |
|---|---|---|

| | | |
|---|---|---|
| AA | $N_{AA}$ | $N * p^2$ |
| Aa | $N_{Aa}$ | $N * 2 * p * (1 - p)$ |
| aa | $N_{aa}$ | $N * (1 - p)^2$ |

*Table 4.3. Observed vs expected number of genotypes*

Where:

$N = N_{AA} + N_{Aa} + N_{aa}$ (total number of observations)

$p = p(A) = (N_{Aa} + 2 * N_{AA})/(2 * N)$ (allelle frequency of A)

Once we have calculated the observed and the expected number of alleles, we can proceed to obtain the test statistic:

$$X^2 = \sum^{genotypes} (observed - expected)^2/expected$$

$$= n * ((4 * N_{AA}N_{aa} - N_{Aa}^2)/((2 * N_{AA} + N_{Aa}) * (2 * N_{aa} + N_{Aa})))^2$$

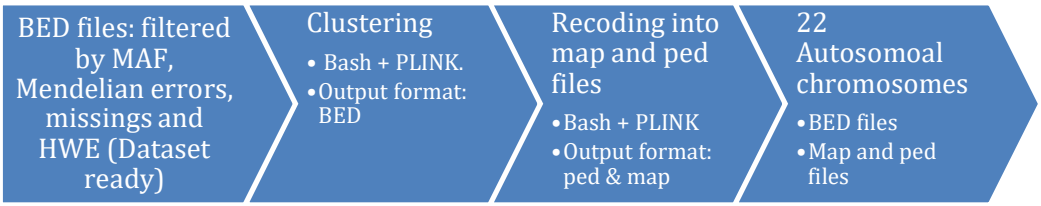
At last, to obtain a p-value we have to consider that the test statistic under the null hypothesis ($H_0$ = the locus is in Hardy-Weinberg equilibrium) follows approximately a Chi-Square distribution with 1 degree of freedom. This implies that using an alpha of 0.05, those values for the test statistic above 3.84 happen to be significant and Hardy Weinberg equilibrium does not hold for that locus.

However, filtering by a p-value of 0.05 in a Hardy-Weinberg equilibrium test could be excessively restrictive. Thus, we have set the threshold to filter p-values of the HWE test to 1e-3.

The commands used in PLINK[24] to carry out this procedure can be found in section A.2.2. of the appendix.

## 4.1.4. Data set-up for association

### 4.1.4.1. Clustering data by chromosome

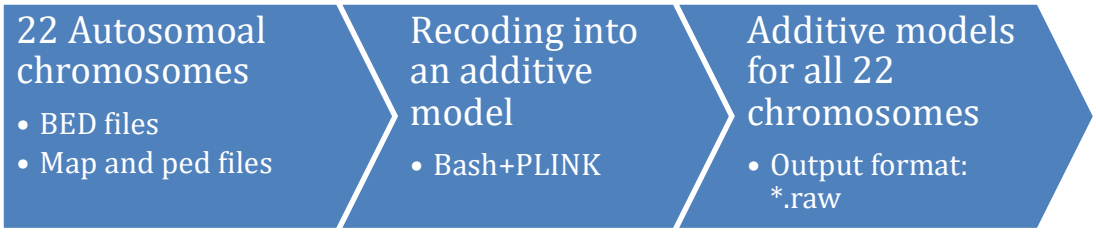



In order to improve computing performance (in terms of RAM and time) during further association, we need to split the data in smaller batches

The suitable clusters in this case happened to be chromosomes so we used a bash script which calls PLINK to cluster data by chromosome[25], and that afterwards recodes the resultant *.bed file into a* .ped and *.map file, since they are the formats required for the association. At the end we should have 22 batches, corresponding to the 22 autosomal chromosomes.

The bash script used to cluster SNPs by chromosome and recode every output file into ped and map format can be found in the appendix section A.2.2.

### 4.1.4.2. Additive models




Although we haven't talked yet about the methodology to carry out the association study, we can assume that it will be performed using linear mixed models. However, whereas traits and covariates are both numeric (either continuous or binary), the genotypes are still coded as letters. Obviously no statistical association can be done with genotypes coded as this, thus a numerical codification is required. The most commonly used way to input genotyped data in SOLAR is through an additive model. Basically, this way of recoding genotyped data consists in counting the number of minor alleles per person. So PLINK (this time wasn't going to be the

exception) first obtains the allele frequency for each allele at a certain locus, and afterwards counts the number of alleles per person for that allele presenting the lowest frequency[26].

Here goes an example:

| SNP | SNP_Additive |
|---|---|
| A A | 2 |
| A C | 1 |
| C C | 0 |
| C C | 0 |
| 0 0 | NA |

*Table 4.4. Example of file encoding transformation from a *.ped file to a *.raw file*

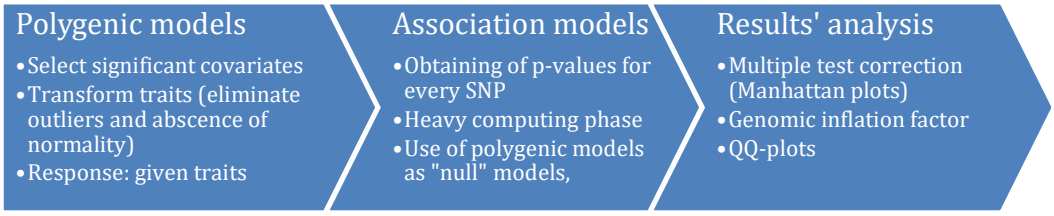
The bash script used to recode every file containing information of autosomal chromosomes can be found in the appendix section A.2.3.

## 4.2. Analysis

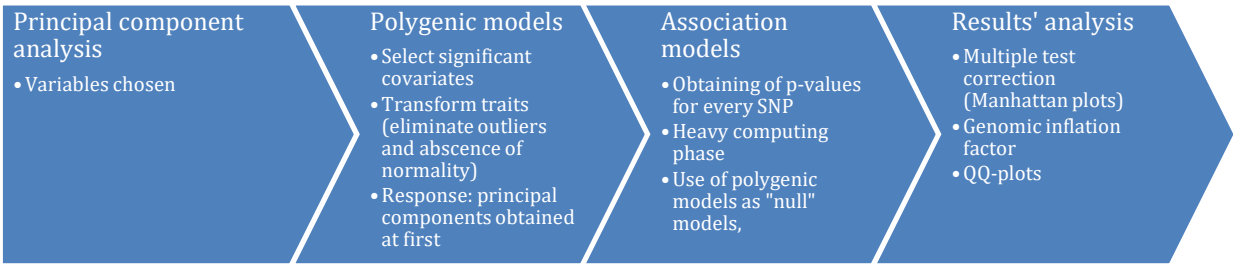
### 4.2.1. Analysis' roadmap

Before we start explaining how we have proceeded, as well as detail the algorithms, the concepts and the mathematics behind this project, it is a good idea to stop for a while and understand at least graphically the process that we have followed to obtain our results.

In a "direct" association (i.e. associating with the traits given directly) we follow the procedure from below:




In an indirect association we first obtain a model for the data, where we can even reduce the dimensionality of it, and afterwards we carry out the association.




However, considering that the basis of this project is linear mixed models (polygenic and association models are LMMs) we are going to introduce them first, giving the most important mathematical concepts and a practical example in order to provide some practical intuition to the reader.

## 4.2.2. Linear mixed models

### 4.2.2.1.. Overview

Although probably everyone who is reading this text has ever heard about linear models, we definitely can't state the same for linear mixed models. The first ones are widely used; in fact they're strongly present in almost every science field. Conversely, the second ones are much less famous despite their flexibility and elegance. When we have clusters or groups in our data, and we suspect that the randomness of the observations could be affected by those groups or clusters, using a mixed model is a highly recommended option[27]. Historically, their presence and use have been restricted by computational limitations, due to the fact that their parameters estimation requires using complicated algorithms which demand high computational efficiency[28]. However, considering recent technological advances and the incredible improvement of computation power, they have become more popular - and even the unique solution to some problems -. Before starting to explain them in detail, we should go back and remember the assumptions needed for linear models, since for mixed models are exactly the same.

The assumptions in linear models can be summarized in five main concepts[29]: linearity, constant variance, absence of colinearity, normality of residuals and independence. By far, the most important one is the latter, and its violation will definitely invalidate the model. Independence implies that the residual errors for the response variables are uncorrelated. As an example, if we have measured a certain trait several times from the same subject, and we have

done it among a group of individuals, our observations between individuals won't be totally independent. It turns out that the mean of observations of each particular subject is slightly different, so therefore the observations are not independent from subjects. It's precisely here where linear mixed models are an excellent solution. In fact, mixed models account for the variance due to two different kinds of variables or "effects": fixed effects and random effects. In the one hand, fixed effects are those ones which influence the variance in a highly systematic and predictable way. In the other hand, random effects tend to influence variance in a very non-systematic and unpredictable manner, as the “by-subject” effect that we have commented as an example. So precisely because they account for both effects linear mixed models are called "mixed".

#### 4.2.2.2. The mathematics behind linear mixed models (LMM)

In this section we are going to introduce from a mathematical and more formal perspective linear mixed models. Although LMM have their basis on very advanced mathematics, here we intend to give to the reader the main concepts as plain as possible, in order to facilitate their understanding as well as improve further comprehension of the methodology presented later on in this project.

##### Mathematical basis

If we express LMM in matrix form we have[30]:

$$y = X\,x\,\beta + Z\,x\,u + \varepsilon \qquad (1)$$

Where X and Z are both known incidence or design matrices of *n x p* and *n x q* dimensions respectively. β is a *p x 1* dimension vector that accounts for fixed effects (as Age or Sex) and u is a *q x 1* dimension vector which accounts partly for random effects, jointly with ε. Nevertheless, *Z x u* gives some structure to random effects, in order to correct the model and let the assumptions of independence intact. We also need to define a covariance matrix for each random effect, one for vector u and another one for vector of residuals ε.

So consider that the covariance matrix for vector u is denoted by G and the covariance matrix for vector ε is denoted by R. The first matrix typically accounts for known random effects, and the second one for residual variance.

The mathematical structure of the matrix G can be defined in different ways, depending on the implementation. The most generic way to define it is:

$$G = \sigma_g^2 A \qquad (2).$$

Where $\sigma_g^2$ is the variance of the random effect to be estimated and $A$ is a generic matrix its structure depends on the prototype for matrix Z chosen. These concepts will be extendedly explained later on, when we comment the different implementations existing for linear mixed models.

Assuming that residual errors have constant variance and are uncorrelated, the matrix R can be written as:

$$R = \sigma_e^2 * I \qquad (3),$$

Where I is the identity matrix and $\sigma_e^2$ is the parameter to be estimated. However, we need to define a few more equations before presenting the final multivariate normal distribution of our trait vector $y$. Considering that u and ε are both random effects, their expected values have to be equal to 0:

$$E(u) = 0 \qquad (4)$$

and

$$E(\varepsilon) = 0 \qquad (5)$$

Conversely, the expected value for fixed effects has to be equal to $X\,x\,\beta$. Furthermore, assuming that random effects $u$ and $\varepsilon$ are uncorrelated, we can define the variance for vector y as:

$$Var(y) = V = Z\,x\,G\,x\,Z^T + R \qquad (6)$$

In the end, the distribution for vector $y$ is:

$$y \sim (X\,x\,\beta,\ Z\,x\,G\,x\,Z^T + R) \qquad (7)$$

ETSEIB

Summarizing, the elements that we know a priori are y, X and Z, and those that we have to estimate are β, u, G and R. The parameters that interest us the most are β and u, though to obtain them there is a previous step which consists in estimating both matrices G and R.

Actually, estimating the matrices G and R is the most complicated step. We need to use complicated mathematic techniques and complex algorithms[31]. In fact, the procedures to obtain these parameters are based on maximum-likelihood estimation techniques, which consist in the iterative optimization of a log-likelihood function. These are functions of the parameters of a statistical model, and there are two of them which deserve a mention due to their importance: Maximum Likelihood (ML) and Restricted Maximum Likelihood (REML). In order to optimize these functions there is a list of algorithms capable to do it, though Expectation-Maximization algorithm is the most widely used[32]. However, we are not going to dig deeper into these concepts, because they are out of the scope of this study.

Nevertheless, we need to talk a bit more about mathematics of LMM, but now focusing on a question of high interest considering our purpose. The issue that should be discussed now is the significance of the parameters found when we fit a mixed model. Whereas in linear simple regression models we were used to check the significance of a set of parameters using test statistics and looking at the adjusted R-squared directly from the models, in linear mixed models this is not feasible. Thus, we need to find a way to discuss significance for LMM, and the solution is Likelihood Ratio Test (LRT)[31]. The basic working principle for these tests is to obtain a "relative" significance between two models. In fact, we are not going to obtain an overall p-value for a certain parameter when comparing two different models: the "null" model and the "full" model. As we can imagine, the full model will use more variables than the null one, concretely, it will include those variables that we want to test their significance. Therefore, the meaning of the p-value obtained, in case to be significant, is that the bunch of variables included in the full model is contributing to fit a better model. Although so far everything seems to work properly, we should read the last statement carefully, since it can be misunderstood. It turns out that not all the variables included in the full model which are not present in the null one, are significant and contribute to fit a better model. This is what we meant with the word "bunch", and implies that when we test two models and we include more than one different variable in the full model, we

can't know which of those variables are significant. In consequence, it's highly recommended to focus on a single variable at the same time when testing mixed models.

### Illustrative example to give some intuition for LMM

In this section we are going to give some practical intuition about linear mixed models through an example applied to the GAO data. However, all the results obtained here won't be used later on. The aim of this section is just giving a clear idea of what linear mixed models are. The example presented here has been carried out using a package called "lme4" in R.

Suppose we want to build a model where the response variable is a certain level of a protein (in this example: "Sclerostin"), and our covariates are 'Age' and 'Sex'. However, we know that our individuals are clustered in families, which might be affecting the independence of observations, since each family could have a slightly different "baseline" for this trait[27]. That is, we expect from each family to have a different average value for the trait in question. To be precise, we are referring now to the intercept term of the model, which will be different for each family if we have a look at the by-family coefficients after computing the model. In addition, the "overall" intercept term given by R is the mean of these by-family intercepts.

First, we have plotted a boxplot of the Sclerostin level against different families:

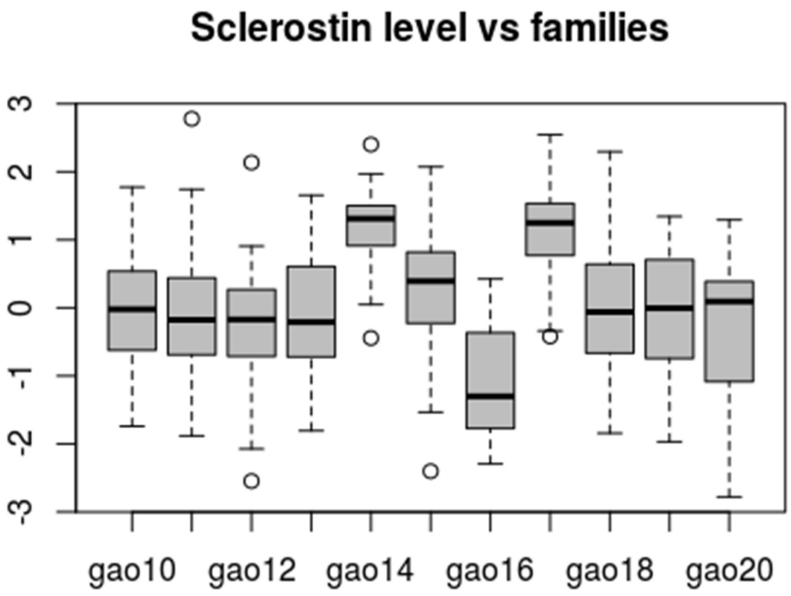



*Figure 4.1. Boxplots of Sclerostin levels per family*

As we can see above in figure 4.1, as it was expected there are obvious differences between families for this protein level. Thus, it seems an unbeatable occasion to fit a linear mixed model. The notation for LMM with random intercepts in R is (1|"Clustering_variable"), where the 1 indicates that the model only has to considerate different intercepts for each cluster. So in this case the formula for the mixed model in R will be something similar to this: Sclerostin ~ Age + Sex + (1|FAM) (actually we applied an inverse normal transformation to the response variable, though it will be explained and justified later on). Once we had the model fit in R, let's have a look at the table of coefficients by-family:

| FAM | (Intercept) | Age | Sex |
|---|---|---|---|
| gao10 | 0,24014312 | 0,007654035 | -0,4008496 |
| gao11 | 0,26786011 | 0,007654035 | -0,4008496 |

| | | | |
|---|---|---|---|
| gao12 | 0,04526834 | 0,007654035 | -0,4008496 |
| gao13 | 0,13284431 | 0,007654035 | -0,4008496 |
| gao14 | 1,34367063 | 0,007654035 | -0,4008496 |
| gao15 | 0,64053023 | 0,007654035 | -0,4008496 |
| gao16 | -0,80328248 | 0,007654035 | -0,4008496 |
| gao17 | 1,20360548 | 0,007654035 | -0,4008496 |
| gao18 | 0,40835251 | 0,007654035 | -0,4008496 |
| gao19 | 0,21483031 | 0,007654035 | -0,4008496 |
| gao20 | 0,03335406 | 0,007654035 | -0,4008496 |

*Table 4.5. Coefficients of the linear mixed models calculated clustering by families and using a different intercept in each one*

However, the most interesting and powerful idea of LMM is yet to come. So far, we have assumed that the way that mixed models have to account for the differences between clusters is to fit a different intercept for each cluster. Actually, this is partially true because mixed models also can fit different coefficients for fixed effects. They are called "random slopes", and it has been demonstrated that they significantly reduce type I errors when they can be applied (not always makes sense to fit them)[27].

In our present example, we can guess that Age might be different for different families. Some of them will have either older or younger individuals than others, and therefore incorporating a random slope for Age doesn't seem a bad idea. In R notation, we would write it as: Sclerostin ~ Age + Sex +(1+ Age|FAM).

Before to have a look at the new coefficients by-family for intercepts and slopes, let's see if age differences between families could be significant:

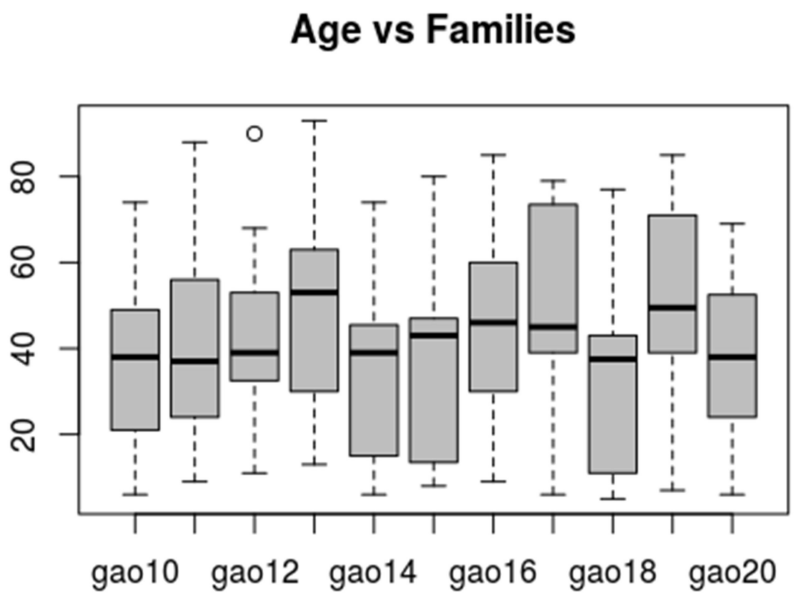



*Fig. 4.2. Boxplots of age stratified by family*

In fact, we can see in the figure above that our assumption was correct and that makes sense to add a random slope for Age to the linear mixed model. Now it is the turn of coefficients, let's have a look on them.

| FAM | (Intercept) | Age | Sex |
|---|---|---|---|
| gao10 | -0,00778901 | 0,014111344 | -0,4090415 |
| gao11 | 0,35933799 | 0,005749535 | -0,4090415 |
| gao12 | 0,10181807 | 0,006819606 | -0,4090415 |
| gao13 | -0,05711655 | 0,011517653 | -0,4090415 |
| gao14 | 1,58249182 | 0,001773113 | -0,4090415 |
| gao15 | 0,81552514 | 0,003503418 | -0,4090415 |
| gao16 | -0,95951546 | 0,011588629 | -0,4090415 |
| gao17 | 1,38137812 | 0,004192465 | -0,4090415 |

| | | | |
|---|---|---|---|
| gao18 | 0,43926883 | 0,007060266 | -0,4090415 |
| gao19 | 0,30787256 | 0,006178744 | -0,4090415 |
| gao20 | 0,04119108 | 0,007894150 | -0,4090415 |

*Table 4.6. Coefficients of the linear mixed models calculated clustering by families and using a different intercept and random slope for Age in each one*

Note that despite the coefficients for age vary, they are actually quite the same and always positive. This is because there is still consistency in how Age affects the response[27].

## 4.2.3. Polygenic models

### 4.2.3.1. The kinship matrix

Before we explain the specific mixed models that we are going to use in this study to carry out the association we need to introduce a new concept: the kinship matrix. In fact, this matrix is going to be the main key of this project, since we will use it almost for everything. Thus, polygenic models won't be understood if we don't define the kinship matrix previously.

The kinship matrix is actually a matrix of similarity between individuals. The coefficients of this matrix are known as kinship coefficients (Φij) and can be defined as follows: is the probability that a random gene from subject i is identical with a gene at the same locus from subject j[33]. In fact, this matrix is accounting for the proportion of the genome shared by individuals, or what is the same: the degree of relatedness.

#### Algorithm

First of all, there are some conditions that our pedigree has to satisfy[33]:

-Any person should have either both or neither of their parents in the pedigree
-The members in the pedigree have to be numbered in such a way that every parent has a lower number than his or her children

The kinship coefficients between any two individuals in the pedigree are computed in a symmetric matrix from left top downwards recursively. Thus, there is only one coefficient that we need to know: the parent-offspring coefficient, which is equal to 1/4. From this datum we can trivially obtain another useful coefficient: the one for siblings, which is also equal to 1/4. The recursive instructions are[33]:

-For $\Phi_{i,i}$:

if i is a founder:

$\Phi_{i,i} = 1/2,$

else:

$\Phi_{i,i} = 1/2 + (1/2) * \Phi_{k,l}$, where k and l are parents of i.

-For $\Phi_{i,j}$, (i>j):

if i is a founder:

$\Phi_{i,j} = 0,$

else:

$\Phi_{i,j} = (1/2) * \Phi_{j,k} + (1/2) * \Phi_{j,l}$, where k and l are parents of i.

### Implementation

When we compute both polygenic and association model through solarius, SOLAR itself estimates the double Kinship matrix based on the identifier fields that have to be present in the tables of phenotypes passed to R. These fields required by SOLAR [34] are: ID, SEX, FAM, FA and MO, as well as MZTWIN. Where FAM contains family information, FA father's ID of the subject, MO mother's ID of the subject and MZTWIN the ID of the monozygotic twin (only in case the subject has a monozygotic twin, obviously).

Actually, there is a gap between the available phenotypic data and the number of IDs. In fact, these six fields contain information of 576 individuals whereas we only have phenotypic data of 367 subjects. This difference is due to the presence of "ficticial" individuals that are necessary to complete the familiar structure and make SOLAR compute correctly the kinship matrix. These "ficticial" individuals either exist or have existed in real life (some of them may be dead), though they have never been part of the study. Before I set off this project and as I said in the introduction there were already people working on this project, and this completion of the familiar structure was one of their contributions.

Finally, we can also estimate the double Kinship matrix through the function "solarKinship2", and obtain two interesting figures using "plotKinship2". In fact, in the results section the reader can found the global kinship matrix for the eleven families and a histogram accounting for the kinship

coefficients frequencies. The code used to obtain these figures, as well as the code used in the illustrative example explained right after this section can be found in the appendix section A.3.1.

## Illustrative example

Finally, in order to have a better idea of how the algorithm to calculate kinship coefficients works we are going to present an example using the pedigree from the 11th family, which is the smallest one and therefore it doesn't need so many recursive operations. However, before calculating any coefficient manually we have plotted the genealogical tree and the "double" kinship matrix (two times the matrix of kinship coefficients), to give an idea of the structure of the pedigree.

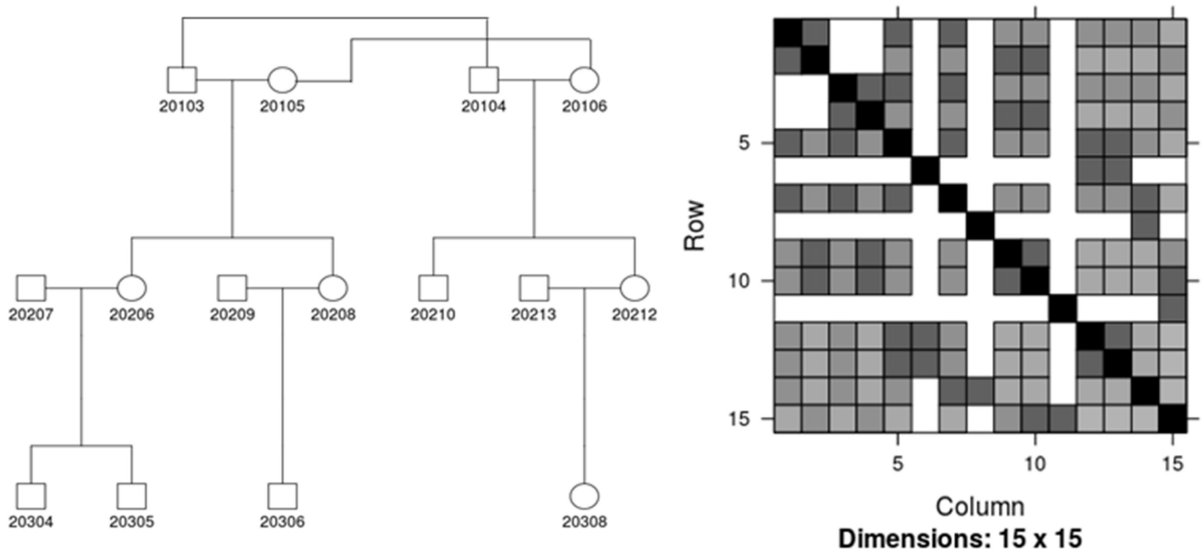



*Fig. 4.3. On the left genealogical tree of the 11th family. On the right, the kinship matrix sorted from the eldest individuals to the youngest ones.*

The kinship coefficient that we decided to calculate is the one below:

$\Phi_{20308,20103} = ?$

Individuals 20213 and 20212 are both parents of 20308, so 20308 is the "i"" individual due to 20308 > 20103. Thus, the j individual here is 20103.

$\Phi_{20308,20103} = (1/2) * \Phi_{20103,20213} + (1/2) * \Phi_{20103,20212}$

ETSEIB

The individual 20213 has neither mother nor father in the pedigree, so there is no relation between 20103 and 20213

$\Phi_{20103,20213} = 0$

$\Phi_{20103,20212} = (1/2) * \Phi_{20103,20104} + (1/2) * \Phi_{20103,20106}$

Individuals 20103 and 20104 are siblings, so the coefficient is equal to 1/4.

$\Phi_{20103,20104} = 1/4$

Individuals 20103 and 20106 are both founders uncorrelated so they are actually unrelated.

$\Phi_{20103,20106} = 0$

If we substitute the values we have found so far into the first formula:

$\Phi_{20308,20103} = (1/2) * 0 + (1/2) * ((1/2) * (1/4) + (1/2) * 0)$

We obtain that the kinship coefficient is:

$\Phi_{20308,20103} = 1/16$

And finally the double kinship coefficient is equal to 1/8 (=0.125).

We have checked if this value corresponds with the computed coefficient by SOLAR and in fact both values are the same.

### 4.2.3.2. Linear mixed models into genetics' context

Fitting and understanding polygenic models is the last step before starting the association study. In the previous sections we have introduced general linear mixed models, but we haven't specified which random and fixed effects should be considered in our study. In fact, although a

polygenic model is a mixed one, there are some differences between them. In matrix form we can write it as[30]:

$$y = X\,x\,\beta + g + c + \varepsilon \quad (1)$$

Where β corresponds to fixed effects and the other three terms account for random effects: g for the additive genetic effect, c for the "household effect" and ε for the residuals. In our particular case, we don't have information for the household effect, which implies that our model will have only two random effects. In the one hand, we have the additive genetic effect that we haven't defined yet. This term gives structure to the random part of the model through individual relationships, also known as "kinship". Instead of clustering by families as we did in our first example when we presented mixed models, now we are giving to the model the full familiar structure, using the kinship matrix presented in section 4.2.3.1. In fact, the covariance matrix $G$ (defined in the previous section) for this random effect g can be written as $Var(g) = \sigma_g^2 * A$, where $A = 2K$ (two times the kinship matrix). Thus, this term accounts for the "proportion" of the genome that indviduals share between them. In consequence we can expect that those individuals who are closer in genetic terms present similar levels for the trait under study, which avoids a violation of independence between observations.

In the other hand, we have the residual random effect ε whose variance can be expressed exactly as we explained in the previous section: $Var(\varepsilon) = \sigma_e^2 * I$. Hence, the multivriate normal distribution for trait $y$ finally is:

$$y \sim \left(X\,x\,\beta, \sigma_g^2 * A + \sigma_e^2 * I\right) \quad (2)$$

.

### Implementation of polygenic models

Polygenic models in SOLAR aren't fitted exactly as LMM in R. In this section we have been insisting on the kinship matrix and its importance to calculate matrix G. Conversely, in the introduction of LMM we have just passed the formula to R, and we didn't need any matrices to estimate matrix G a priori. That is because SOLAR and the package lme4 use different prototypes for matrices G and Z.

In the one hand, the package lme4 in R computes G as $\sigma^2 * I$ (in this case we denote $\sigma$ as the variance of a generic random effect) and Z is “free”. The meaning of “free” in this case is that matrix Z has no defined dimension or structure a priori. Therefore, matrix Z is calculated following the formula passed to R and the clustering data given (for instance the “FAM” field).

On the other hand, SOLAR uses a prototype for matrix Z, setting it equal to the identity matrix I, and allows a much more complicated structure for matrix G using the kinship matrix: $G = \sigma_g^2 * 2 * K$ [35]. So, in the particular case of the polygenic model in SOLAR, we can write the total variance $y$ as:

$$Var(y) = G + R = \sigma_g^2 * 2 * K + \sigma_e^2 * I \ (3).$$

### 4.2.3.3. Utility of Polygenic models

So far, we have given an outline of what are polygenic models and their main differences against general mixed models, though we have commented nothing about the possibilities they have and their usefulness. As we will see in section 4.2.4, when we explain associations models, polygenic models are basically their basis. Therefore, we will use them to prepare the land for the main purpose of this study, which is genetic association. Concretely, we will use them to choose the suitable covariates and estimate the heritability for each trait. In order to perform these tasks, two new and crucial concepts have to be explained, which are: the inverse normal transformation and heritability. Although the likelihood ratio test (LRT) it's also crucial, it has been already explained when we talked about the mathematical basis of LMM.

#### The inverse normal transformation

When we fit any model, we have to pay special attention to the distribution of the response variable. Extreme values or a clear absence of normality may lead to fit the wrong model, which will give us wrong results as well, converting our study into nonsense. However, historically outliers have always been a controverted and recurrent problem, and therefore so many techniques and procedures have been described and purposed in scientific literature so far to deal with them. However, for this particular study we have decided to use an "inverse normal transformation" due to their advantages and easy implementation [36].

- The first main advantage that we can mention is that we don't have to worry about which distribution has the trait before transforming it: for sure its final distribution will be a normal one. This is especially useful in clinical and biological studies, where sometimes the distribution of some traits is far from being normally.
- In second place, the standard deviation of the final distribution is known (equal to 1) and it is not considerably low, which prevents from calculus errors when fitting models.

- We don't have to worry about outliers. Since what is kept is the rank of our data, it is almost impossible to have very extreme values in the final distribution. The logarithmic transformation also reduces differences between outliers and normal points but sometimes the standard deviation of the transformed distribution is too low, which leads to calculus errors.
- Another advantage consequence of the last one is that the Kurtosis obtained when fitting a model is always very low. If there are not extreme values, obviously the shape of the normal distribution will be as an usual one, with a value for "tailedness" within normal range.

Finally, the formula used in this case to carry out the transformation is the following one:

$$Y_i^t = \Phi^{-1}((r_i - c)/(N - 2 * c + 1)) \quad (1)$$

Where:

$\Phi^{-1}$: Inverse probit function.

$r_i$: Rank for the ith observation.

N: Total number of observations which are not missings.

c: A parameter to modify slightly the normal curve. Whereas in literature the common value for it is 3/8, we set it to 0.

There is only one handicap to be considered in using this transformation: it reduces a bit the statistical power of the study.

In order to test that the inverse normal transformation calculated by SOLAR is the as the one we have presented here, we have implemented the code in R, which can be found in the appendix A section A.3.2.

### Heritability

Heritability is a statistic that estimates how much variation in a phenotypic trait in a population is due to genetic variation among individuals in that population[37]. The concept of heritability applies only to traits that differ between individuals, because in case that the trait is the same from across all individuals, it might be inherited, but it is not heritable. Nevertheless, as higher is the percentage of heritability for a certain trait, as higher is the interest to study that trait from the genetics' perspective. In mathematical terms, the heritability defined as the percentage of phenotypic variance due to the additive genetic effect:

$$h^2 = \frac{Var(A}{Var(P)} \quad (2)$$

Where Var(A) is the variance due to the additive genetic effect and Var(P) is the overall phenotypic variance.

## 4.2.3.4. Polygenic models in GAO

The procedure that we followed to fit polygenic models has been slightly different for each group of traits. In fact, we have used different covariates to fit a polygenic model for proteins, clinical phenotypes or affected phenotypes. In addition, the algorithm used by SOLAR to estimate parameters when traits are binary (as affected phenotypes) isn't exactly the same either. Nevertheless, all models obtained in this section have been fitted using the package "solarius" in R, that is an interface more user-friendly to run genetic analysis in SOLAR. Concretely in this section we basically use a single function, which is "solarPolygenic". An important detail about this function is that the p-values reported for covariates can be directly interpreted, because SOLAR automatically performs a Likelihood Ratio Test for each one of them. Furthermore, we take advantage from the fact that we are working in R whereby we can organize and deal with the data in a very flexible way.

All the polygenic models presented in this results' section have been used as "null" models in the further association analysis.

## Polygenic models for bone metabolism markers

### Methods

First of all, we should have a look at the covariates considered for this particular case. As we can see in the table 3.3, the possible covariates available in the table of bone metabolism makers are the following ones: Age, Sex, MenopAge, Alcohol, Smoking, SolarExp, PosDrug, NegDrug, IPAQ and Calcium. However, before using them to fit our models we should check the number of missing values for each one, since we can only consider those individuals who have information for all the covariates under consideration. Looking at the table 3.3 again we can rapidly conclude that covariates MenopAge and IPAQ contain an excessive number of missings and that they have to be discarded. In fact, using the data from 60 individuals to infer the parameters of our model instead of 367 may change the resultant model so much. Although the case for IPAQ is not that exaggerated 258 individuals in front of 367 is also a very significant difference. Once we have discarded both two problematic covariates, in order to find the suitable covariates for these traits and estimate their heritabilities, we have split the analysis in two steps. The first step has consisted in testing all the covariates considered jointly with a quadratic element for Age ($Age^2$). Afterwards, we fitted a second model with those covariates that were significant in the first analysis. Nevertheless, in case that "Age" was significant and its quadratic element wasn't in the first attempt, we considered it again in the second try due to the fact that using a few more individuals may change the resultant model and covariates significance. Also, in case that "$Age^2$" happened to be significant and "Age" didn't, we directly remove the covariate.

In the other hand, it could seem that we are insisting very hard on using a quadratic element for Age. In fact we are, because the effect of age, as many other things in biology, rarely can be explained with a straight line. Finally, since all traits in this section are quantitative, we have applied the inverse normal transformation to all of them in order to deal with possible outliers and non-normal distributions. The summary table containing the models for each trait is the one below:

### Implementation

In order to follow the methods presented above we have used a script to fit the polygenic models that can be found in the appendix section 3.3.

Considering that all the traits are quantitative in this section, SOLAR works as usual, using the algorithms commented in section 4.2.2 to optimize the likelihood cost function and estimate the parameters.

At last but not least, apart from the script used to obtain a first approach of the models, we have dealt manually with some the models of some traits when we saw something strange in them. Concretely, for the Adiponectin model, in the first step we had a bunch of variables that seemed to be significant. However, when we fit the supposed final model using them, the sample size increased (the covariates remaining had less missings) and in consequence the covariate Age happened to be non-significant. The problem was that the covariate $Age^2$ was apparently significant. Therefore, we removed them both. But it doesn't end here, when we did that, one covariate from the two remaining started to be non-significant. So finally, we ended fitting a model with just one covariate, that was sex and very significant.

## Polygenic models for clinical phenotypes

### Methods

To fit the models for clinical phenotypes, instead of considering a bunch of covariates as we did in the case of proteins we have just used three of them: Age, Age^2 and Sex. The main reasons for choosing only these three variables have been the following ones:

- First, there were so many traits; therefore we would have needed a lot of time to fit a very accurate model for each trait. Despite accuracy it's important, in many occasions using covariates which are not Age, Age^2 and Sex barely increases the proportion of variance explained. Apart from this fact, if we look at the models fitted for proteins, just three models out of twelve use a covariate which is not among Age, Age^2 and Sex. Thus, we can save a lot of time fitting more basic models which at the same time explain a reasonable amount of variance.

- Secondly, these three covariates don't have heritability which avoids interfering with the resultant model. For instance, when we use covariates as Body Mass Index (BMI) we have to be

very careful, because they have heritability and the estimation of random effects of the models may be affected leading to an inconsistent model.

- Finally, these variables are totally uncorrelated and can't be confounded between them. In fact, there are some covariates that can be confounded, resulting in a model sometimes difficult to interpret, as well as poorly fitted.

### Implementation

The script used in this case (Annex A.3.3.) is similar to that one used to fit polygenic models for the bone metabolism markers. In this case, traits are still quantitative and therefore SOLAR works as usual too, optimizing the likelihood functions to estimate the parameters of the models through the common algorithms.

## Polygenic models for affected phenotypes

### Methods

As we did in last case, we fitted models for the Affected traits using just three covariates, which are Age, $Age^2$ and Sex. The main reason for that is the same we argued for clinical phenotypes. These three covariates can't be confounded between them, they don't present heritability and almost always they explain a considerable proportion of variance. In addition, these covariates present no missings, meaning that we have the whole sample available. Furthermore, in this section we are dealing with binary traits, so an abuse of controverted covariates is not recommended.

### Implementation

The algorithm to estimate the parameters is not exactly the same used for quantitative traits. In fact, this algorithm has to perform a few more steps rather than the common algorithm does, demanding a huge amount of power computation and needing almost 2.5 more time to estimate the parameters. This is not a problem when we fit a small set of models, but it becomes problematic when we need to fit 650.000 models (for instance when we associate). About these new steps included in the algorithm, they don't convert the procedure into a logistic regression because the optimization is still carried out through the Maximum Likelihood method, though the probit function is used to transform quantitative predictions of the trait into binary ones.

The script used to fit the polygenic models for affected traits can be found in the appendix A.3.3.

### 4.2.4. Association

When we introduced polygenic models, the first thing we commented was that they were the basis of association models. In fact, the random and fixed effects considered to fit the polygenic model are also present in the association model. The only difference between both models is that in the association model we add a new fixed effect, which is the SNP object of study. Actually, there are hundreds of thousands of SNPs which are under study in a GWAS, though we add just one SNP at the same time, at least in this project. The PhD* of Dr. Helena Brunel precisely focus on a new kind of association models, where several SNPs are included in a single association model, though this new method is absolutely out of the scope of this study. In matrix form we can write the association model as[30]:

$$y = X\ x\ \beta + \beta_{snp}\ x\ snp + Z\ x\ u + \varepsilon$$

where the element $\beta_{snp}\ x\ snp$ accounts for the fixed effect produced by the SNP in question. Now, let's move back for a while to last section of the quality control, when we explained how to transform each genotype into an additive model. In fact, the second part of the product of the new fixed effect noted as "snp" is exactly what we calculated when we transformed each genotype from letters to numbers. However, the reader might be wondering how we can extract from this model the precious p-value to prove SNP significance, which is the main aim of this study. The answer for this question is again Likelihood Ratio Test (LRT). Essentially, what we are doing when we run an association is fitting two models and comparing them. In the one hand, we fit the "null" model, which actually corresponds to a polygenic model with their fixed and random effects (for instance age and the additive genetic effect respectively). In the other hand, the "full" model, including a new fixed effect that is actually the SNP. It's important to mention that in all association models we have applied the inverse normal transformation to the response variables. Once both models are fitted, we apply a Likelihood Ratio Test extracting a p-value for the SNP tested.

Nevertheless, this p-value can't be interpreted as usual do to determine significance. In this case, the threshold for the p-values is set using Bonferroni's correction for multiple

testing[38]. The motivation for applying this correction is that as we increase the number of hypothesis being tested, we also increase the probability of a rare event, and therefore the likelihood of incorrectly rejecting null hypothesis (type I error). The principle of multiple testing corrections is to set a threshold for the pvalues to avoid Type I error inflation. In this study we use a particular kind of Bonferroni's correction, that despite it's not the most widely used, it makes much sense in this scenario. Typically, Bonferroni's correction sets a new threshold for p-values as follows: $\alpha = \alpha_0/m$, where m is the number of hypothesis being tested. For our particular case, $m$ is equal to the total amount of independent SNPs found in genetics, instead of our number of SNPs being tested. In scientific literature*, $m$ has been approached to at least 1M of independent SNPs, which sets the threshold for p-values to 5e-08 (considering $\alpha_0 = 0.05$).

### Implementation

Now that we have defined how the association algorithm works, it seems a good idea to summarize in the table below the different standard associations that we carried out in this study.

| **Phenotypes** | **Covariates (Y/N)** | **Covariates considered** |
|---|---|---|
| Bone metabolism markers (12) | No | - |
| Bone metabolism markers (12) | Yes | Age, Age^2, Sex, Alcohol, Smoking, SolarExp, PosDrug, NegDrug, Calcium |
| Densitometric traits (23+8) | Yes | Age, Age^2, Sex |
| Affected (4) | Yes | Age, Age^2, Sex |

*Table 4.7. Summary of the different associations carried out and the covariates considered in each case to fit the models*

It has to be remarked that covariates mentioned in the third column from the table above weren't always used to fit the "null" association model. Precisely, the aim of fitting polygenic models was choosing those significant covariates among a group of candidate variables.

The scripts in R used to carry out the different association studies shown in the table 4.7 from above can be found in the appendix. Concretely, we can distinguish two main scripts: the first one used to apply the inverse normal transformation to the phenotypes and obtain transformed tables; the second one the script to carry out the association itself and export the results to files. They can be found in sections A.3.2. and A.3.4. respectively.

As a general note for the association scripts, just mention that they have been performed using "parallel computation" in a server with RAM 128GB (64 CPU x 2,3GB) which has increased considerably calculation speed as well as reduced the proportion of RAM (even though we had a large amount of it, we should take care of this aspect, considering that actually we are sharing the server with the people from the research's department).

## 4.2.5. PCAs

### 4.2.5.1. Overview

In this section we are not going to explain in detail what Principal Component Analysis are assuming that the reader is familiar with them. Nevertheless, the code written in R that can be found in the appendix is very clear and the steps followed to carry out the analysis can be easily understood.

Before we continue, it is important to emphasize that all Principal Component Analysis in this study have been performed using only the densitometric traits (23+8) and two interesting covariates: WBTotArea and WBTotBMC, due to their high degree of correlation. So, we should explain why it makes sense to perform a PCA in this study and how are we going to use the results obtained. Actually, the association study that we explained previously has low power due to sample size. In fact, 367 individuals is a quite small sample size, considering that nowadays there are research groups performing GWAS with sample sizes of thousands of individuals. Although our sample is clustered in families, which can increase power in case there is a rare variant associated with osteoporosis with high presence within a family, the fact is that the average power is low. Thus, considering the low power and that the clinical phenotypes are highly correlated, an association with the resultant eigenvectors obtained in the PCA seemed to be a proper method to enhance power[39].

### 4.2.5.2. PCA of the clinical phenotypes

In the first PCA that has been performed in this study we haven't considered the familiar structure. We have just centered and scaled the matrix of phenotypes, in order to obtain a clear covariance matrix and avoid measuring the variance of the means of the different traits. Afterwards, we have obtained the eigenvalues and eigenvectors of the covariance matrix, where the eigenvalues account for the proportion of variance explained by principal components and the eigenvectors are the directions of principal components. The main drawback of this PCA is that whereas we are trying to find a "model" of the data that accounts for its internal structure, we might be overlooking or confounding the variance due to the degree of relatedness among individuals. Because of this fact, in the next section we will explain the correction by degree of relatedness that we have carried out, and why we decided to associate with the resultant principal components of the latter PCA.

Finally, we should consider that the matrix of phenotypes contains 33 features, since the 4 binary affected traits have been removed as well as the 3 continuous affected traits. The decision to exclude these phenotypes has its basis on the fact that their nature is so different from the other traits, especially the binary ones, which might lead to an incorrect model of the data.

### 4.2.5.3. Correction of the PCA by the kinship matrix

As we have mentioned several times in the last two sections, we have "hacked"" the PCA correcting by the degree of relatedness among individuals. To perform this correction we have used the kinship matrix one more time. Nevertheless, due to a dimensional problem we have had to slightly modify the kinship matrix. For a better understanding of this problem we need to know that the formula applied to obtain the "corrected" covariance matrix is:

$$\Sigma = \left(\frac{1}{N-1}\right) * X^T x\, C^{-1} x\, X \qquad (1)$$

Where:

N: number of observations

X: matrix of features, preferably centered and scaled

C: kinship matrix

So the dimensional problem resides in the fact that the kinship matrix a priori was 608x608 and the matrix of phenotypes was 576x33. Concretely, when SOLAR computes the kinship matrix, it could consider more than the 576 IDs present in the matrix of phenotypes. That is, because there can be individuals whose mother or father are not among the IDs of the table of phenotypes. In this case, SOLAR includes these individuals in the kinship matrix, increasing its dimension and leading to the actual problem. In order to solve this problem, we have removed those columns and rows corresponding to those individuals who are not among the IDs of the table of phenotypes. Previously we have eliminated from the table of phenotypes those IDs presenting missings.

The objective of carrying out the procedure commented so far is obtaining a better model of the internal structure of the data, more suitable to associate with the genotypes in order to enhance the overall power. Later on we will see if this proposal actually works, comparing both PCAs as well as applying this procedure to the GAIT (genetic association in idiopathic thrombophilia) data.

Finally, just mention that this is not the only way that we could have followed in order to obtain a better internal model of the data. Another possibility would have been choosing a set of individuals unrelated from the different families, where the source of variance due to the familiar structure didn't exist. The problem of this alternative is that we would have ended with a very reduced set of individuals, as well as selecting them wouldn't have been easy at all, and quite ambiguous.

#### 4.2.5.4. Association with resultant Principal Components set of vectors

After computing the corrected PCA, the last step is the association between the genotypes and the principal components. Whereas typically a principal component analysis is used to make a dimensionality reduction, which in our case would imply associating only with the principal components accounting for a high percentage of the variance, we are going to consider all of them in order to gain power. This power gain is primarily due to increased power to detect genetic variants with opposite effects on positively correlated traits and variants that are exclusively associated with a single trait[39]. Nevertheless, we should be careful because in

general the last principal components only account for noise, which leads to control the results by looking at the "loadings". The loadings are the weights that each feature has in each principal component, and generally are not easy to be interpreted. In this case we have just considered the loadings for the principal components where significant SNPs have been found, analyzing their absolute values.

Finally, mention that in the association with Principal Components we have used the three main covariates of this study: Age, Age^2 and Sex. However, instead of using different covariates for each Principal Component, we have used all three of them.

## 4.3. Main exploratory tools

### 4.3.1. Manhattan plot

This figure is probably the most famous and representative one in Genome Wide Association Studies. The logarithms base 10 of the p-values of a particular trait are plotted against their position in the genome. So it gives a first idea of where are the most significant SNPs and which places in the genome should be revised deeply. In order to split the SNPs into significant, suggestive and neither suggestive nor significant there are usually plotted two lines in the graph. These lines are plotted following Bonferroni theory which has been explained in section 5.3. The threshold for both significant and suggestive lines have been calculated considering that among all the SNPs known so far one million of them are independent, which means that they are not in linkage disequilibrium. For the significant line the type I error rate has been set to 0.05 and for the suggestive line it has been set to 0.1. However, in some Manhattan plots it appears an additional line which in this case is placed between the significant and the suggestive lines. This line corresponds to the threshold calculated following the classical Bonferroni expression, where the number of independent SNPs is not considered and the type I error rate is divided by the number of total SNPs tested in the study.

### 4.3.2. Genomic inflation factor

The genomic inflation factor is a parameter used to estimate how inflated the number of false positives is. Generally, a high inflation factor may be due to unknown familiar relationships, a poorly calibrated test statistic, systematic technical bias, or gross population stratification.

Nevertheless, none of these factors is present in our study, so an inflated distribution of p-values has to be due to other underlying effects. In fact, has been recently proved that in presence of polygenic inheritance (as we have in this association study) substantial genomic inflation is expected. Its magnitude depends on sample size, heritability, linkage disequilibrium and the number of causal variants[40]. The calculation of the inflation factor is performed comparing the median of the quantile chi-square with 1 degree of freedom transformation of the p-values against the median of a chi-square distribution with 1 degree of freedom.

Finally, we should mention the thresholds chosen for the inflation factor. Actually, we have just followed the criteria usually used in scientific literature, which says that an inflation factor of 1±0.1 is considered unacceptable and the results have to be discarded.

The formula used by the function from the GenABEL package in R [41] to estimate the genomic inflation factor is the following one:

$$\frac{median\big(qchisq(pvector,1)\big)}{qchisq(0.5,1)} \qquad (1)$$

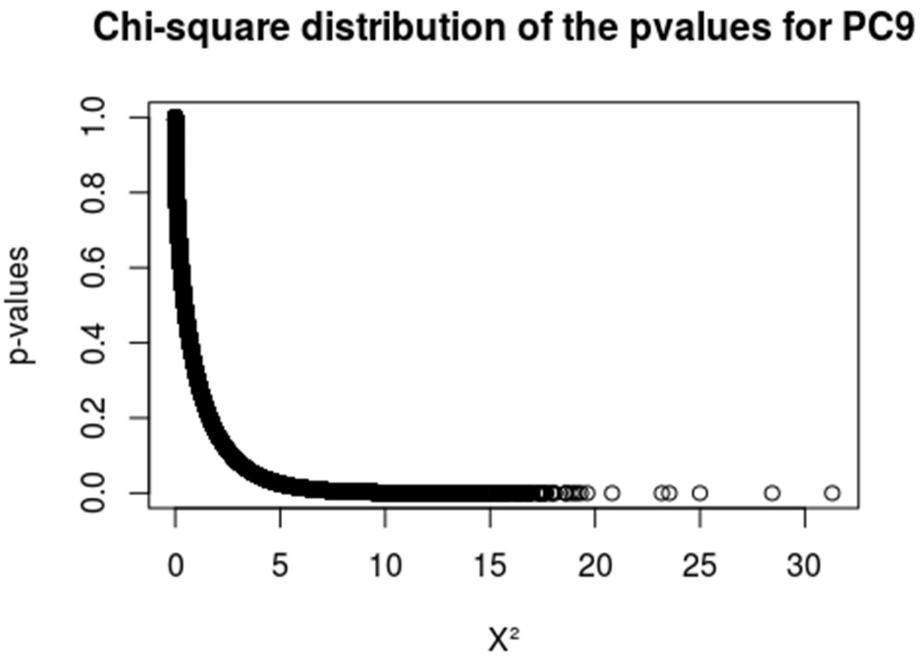



*Figure 4.4.Example: Chi-square distribution of transformed p-values after applying a quintiles' transformation.*

### 4.3.3. Q-Q plots

Quantile - Quantile plots are actually highly related with the genomic inflation factor since they are another tool to visualize if the distribution of the data is correct or not. The distribution of p-values itself has to follow a uniform distribution between 0 and 1, and it's exactly what the Q-Q plot shows. It plots a theoretical uniform distribution against the p-values obtained in the association study[42]. If the p-values are correct, they have to follow the straight line plotted of slope 1 and intercept 0. In case they are concentrated above this line, it means that they are inflated, and the analogous case if they are concentrated below the line.

# 5. Results

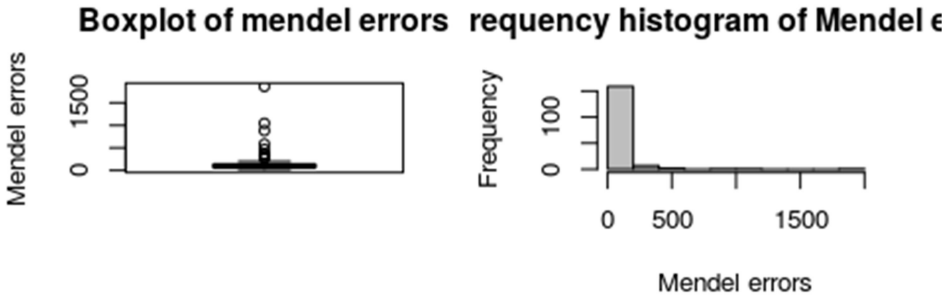
## 5.1. Quality control

### 5.1.1. Mendelian errors per individual

From the individuals' perspective, we are just going to present the results we have obtained, though we have decided not to exclude any individual since the results are not conclusive enough.

#### Mendelian errors distribution per individual

In order to analyse the distribution of Mendelian errors per individual we have plotted several figures that include: a boxplot, an histogram with all the data, a density histogram with two fitted density lines (those points considered outliers in the boxplot are now removed) and a frequency histogram with a normal curve of mean and standard deviation the same as the data (with outliers removed too). We plotted the last two histograms to show that errors seem to be normally distributed when outliers have been removed, which implies randomness.



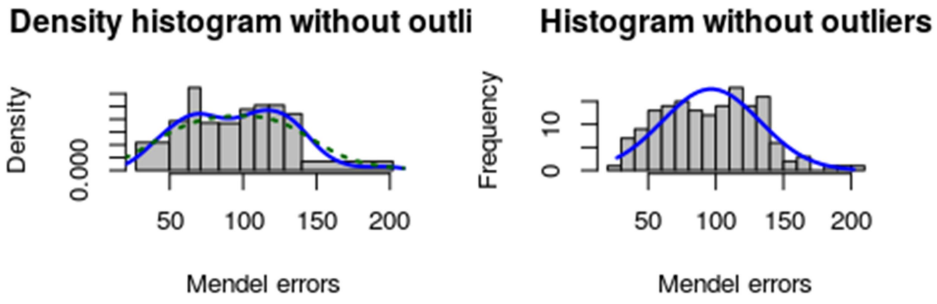



*Figure 5.1. Different plots showing the distribution of Mendelian errors in different scenarios.*

### Mendelian errors outliers' analysis per individual

Here we present the results obtained after having analysed the most important outliers in detail.

| ID | Errors 1 & 2 | Errors 5 & 8 | Errors 3 & 6 | Errors 4,7,9 & 10 | Total |
|---|---|---|---|---|---|
| 13301 | 889 | 2 | 70 | 85 | 1046 |
| 13302 | 233 | 3 | 46 | 73 | 355 |
| 13305 | 1654 | 1 | 116 | 92 | 1863 |
| 13306 | 1 | 0 | 39 | 42 | 82 |

*Table 5.1 Analysis of the different types of mendelian errors of the most problematic individuals*

In the table above we have included both greatest outliers (positions 1 and 3), and their brothers (positions 2 and 4 respectively). Even though the first brother is also an outlier (355), it is still quite far from 1046. A peculiar observation is that the brother of the second outlier, who seems to have been well genotyped and whose number of Mendelian errors can easily be attributed to randomness, barely has errors of type 1 & 2, and the errors affecting just one of the parents, seem to be equally distributed. Although the following statement might seem very speculative, in the event that the offspring was not the child of both assumed parents, the most likely case is that the father was another person. Considering that, errors 1 and 2 have to be assumed as father's errors, which implies that as higher is the number of these errors as higher is the likelihood of a paternity problem. Nevertheless, we should keep in mind that the numbers in the table presented above are still very far away from being a significant proportion considering the total amount of SNPs genotyped per individual. So from the individual's point of view, no individuals are going to be discarded from the study.

## 5.1.2. Overall filters

### Summary table of the filters applied

In the table below we summarize the effects produced by each filter applied to the dataset.

| Filter | SNPs pruned | SNPs remaining |
|---|---|---|
| MAF | 279.009 | 685.184 |
| Mendelian errors per SNP | 3383 | 681.801 |
| Missingness test | 14615 | 667.186 |
| HWE | 2815 | 664.371 |

*Table 5.2. The different filters applied in the quality control and their effects on the dataset*

## Table of SNPs per chromosome

After performing the chromosome's clustering, we obtained 110 files corresponding to 22 autosomal chromosomes. The numbers of SNPs per chromosome are collected in the table below:

| Chromosome | Num. of SNPs |
|---|---|
| chr1 | 53750 |
| chr2 | 52541 |
| chr3 | 43154 |
| chr4 | 37227 |
| chr5 | 38933 |
| chr6 | 46295 |
| chr7 | 34965 |
| chr8 | 34107 |
| chr9 | 30404 |
| chr10 | 35351 |
| chr11 | 33917 |
| chr12 | 32490 |
| chr13 | 24777 |
| chr14 | 21324 |

| | |
|---|---|
| chr15 | 19924 |
| chr16 | 21026 |
| chr17 | 18785 |
| chr18 | 19421 |
| chr19 | 14478 |
| chr20 | 16928 |
| chr21 | 9476 |
| chr22 | 9808 |
| **Total** | **649081** |

*Table 5.3. Final distribution of SNPs per chromosome*

Note that there is a discrepancy between the final number of SNPs obtained after pruning, and the total amount of SNPs contained in the 22 autosomal chromosomes. That is because in the original file we had SNPs from the chromosomes of sex X and Y, as well as from the mitochondria.

## 5.2. Kinship matrix

In this section we present the global kinship matrix for the eleven families and a histogram accounting for the frequencies of the double kinship coefficients. The dimensions of the kinship matrix correspond with the number of individuals that have information of the 6 necessary identifier fields (576 subjects). Actually, the initial kinship matrix computed by default in SOLAR has greater dimensions, since there are individuals that just exist in the mother or father fields of other individuals and that SOLAR takes in consideration. Therefore, the initial dimensions for the kinship matrix were 608x608.

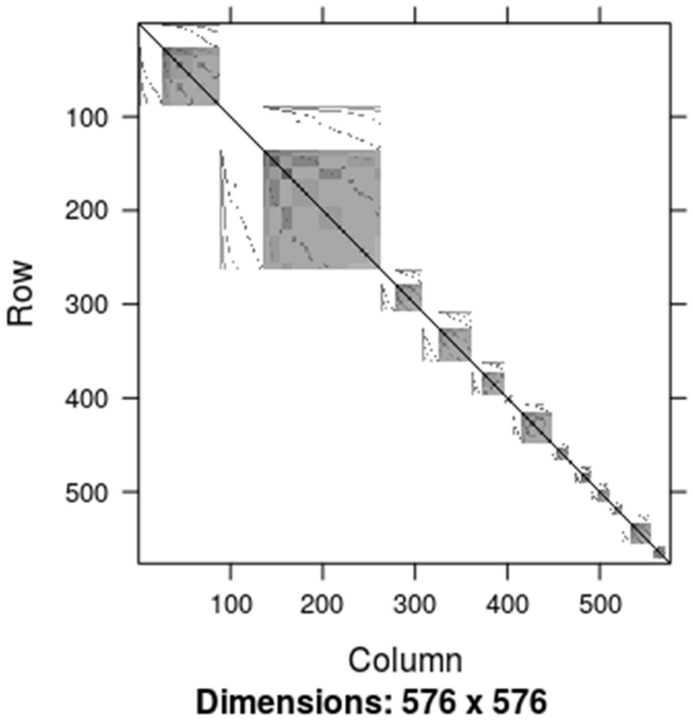



*Figure 5.2. Global kinship matrix*

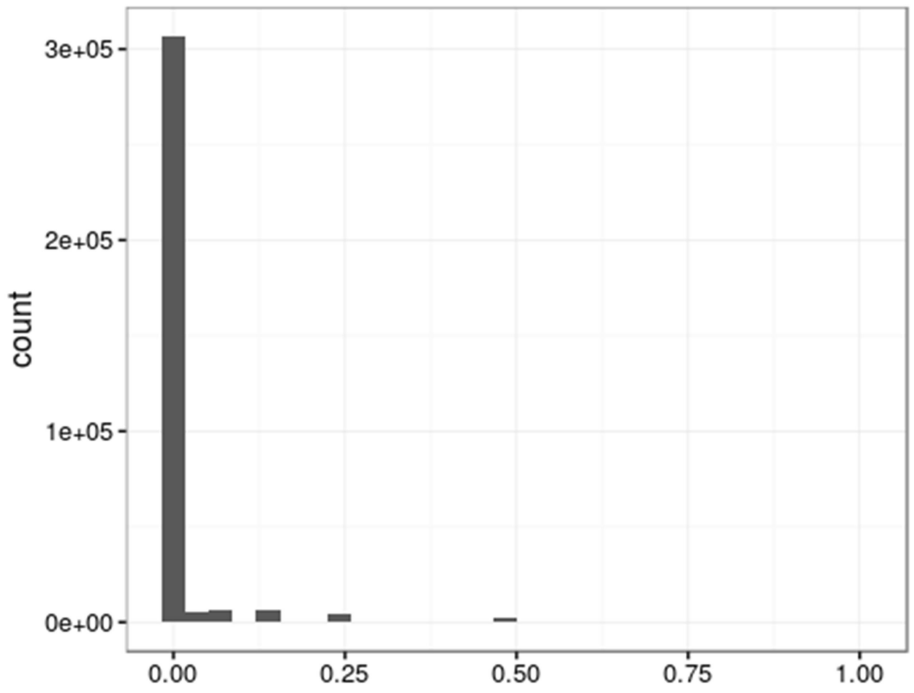



*Figure 5.3. Histogram showing the distribution of the kinship coefficients of the whole set of individuals*

## 5.3. Polygenic models

For each different group of traits we report a summary table of:

1. The significant covariates included in the model of each trait
2. The number of individuals considered in each model
3. The heritability of each trait and its significance
4. Either the proportion of variance explained or the adjusted R-squared (for quantitative and binary traits respectively)

All the polygenic models summarized in the tables of this section will be used in the association analysis as the “null” models, in order to test SNPs significance.

### 5.3.1. Polygenic models for bone metabolism markers

The first table of resultant polygenic models doesn't present any remarkable surprise. The heritabilities are moderately high in general whereas the average proportion of variance explained is rather low (actually as it was expected).

| **Trait** | **Covariates** | **N of individuals considered** | **Heritability** $h^2 \pm std$ | **P-value heritability** | **Proportion of variance** |
|---|---|---|---|---|---|
| HydVitD | PosDrug | 366 | 0,428 ± 0,0776 | 8,951e-12 | 0,0314 |
| Sclerostin | Age, Age$^2$, Sex | 367 | 0,517 ± 0,0882 | 2,912e-11 | 0,0683 |
| SerCrossLaps | Age, Age$^2$ | 366 | 0,299 ± 0,0744 | 2,00e-7 | 0,360 |
| OstaseBAP | Age, Age$^2$, Sex | 367 | 0,353 ± 0,0703 | 2,343e-12 | 0,377 |
| IGF1 | Age, Alcohol, Calcium | 359 | 0,593 ± 0,0843 | 1,218e-15 | 0,251 |
| Adiponectin | Sex | 367 | 0,384 ± 0,0953 | 6,00e-7 | 0,170 |
| Leptin | Age, Sex | 367 | 0,230 ± 0,0931 | 0,00132 | 0,401 |
| Osteocalcin | Age, Age$^2$ | 367 | 0,258 ± 0,0880 | 6,240e-5 | 0,314 |
| Osteoprotegerin | Age, Age$^2$, Sex | 367 | 0.544 ± 0,0773 | 2,362e-18 | 0,406 |
| Osteopontin | Age, Age$^2$, | 367 | 0,282 ± | 1,080e-5 | 0,285 |

| | | | | | |
|---|---|---|---|---|---|
| | Sex | | 0,0885 | | |
| Parathyroid | Age, Age$^2$, PosDrug | 366 | 0,296 ± 0,0790 | 6,00e-7 | 0,0801 |
| TNFalpha | Age, Age$^2$, Sex | 367 | 0,712 ± 0,0642 | 2,954e-38 | 0,191 |

*Table 5.4. Resultant polygenic models obtained for bone metabolism markers*

## 5.3.2. Polygenic models for densitometric traits

All the results obtained for these traits can be directly interpreted. In general, the heritabilities reported are quite high and most of them strongly significant.

| **Trait** | **Covariates** | **N of individuals considered** | **Heritability** $h^2 \pm std$ | **P-value heritability** |
|---|---|---|---|---|
| AxisLen | Age, Age^2, Sex | 364 | 0,383 ± 0,0794 | 7,016e-11 |
| FemShACT | Age, Age^2, Sex | 363 | 0,376 ± 0,0918 | 7e-07 |
| FemShBR | Age, Age^2, Sex | 363 | 0,535 ± 0,0895 | 3,229e-12 |
| FemShCSA | Age, Age^2, Sex | 363 | 0,227 ± 0,0907 | 2,056e-03 |
| FemShCSMI | Age, Age^2, Sex | 363 | 0,344 ± 0,0902 | 1,7e-06 |
| FemShSMod | Age, Age^2, Sex | 363 | 0,282 ± 0,0907 | 1,017e-04 |
| HipNeckT | Age, Age^2 | 363 | 0,486 ± 0,0992 | 1e-07 |
| HipNeckZ | | 359 | 0,686 ± 0,0864 | 5,887e-15 |
| HipTotBMD | Sex | 364 | 0,238 ± 0,0868 | 1,126e-03 |
| HipTotT | Sex | 363 | 0,260 ± 0,0864 | 3,217e-04 |

| | | | | |
|---|---|---|---|---|
| HipTotZ | Age, Age^2 | 359 | 0,626 ± 0,0898 | 6,137e-14 |
| InterBMD | Age, Age^2, Sex | 364 | 0,403 ± 0,0976 | 1,4e-06 |
| IntTrACT | Age, Age^2, Sex | 363 | 0,490 ± 0,0907 | 1,655e-09 |
| IntTrBR | Age, Sex | 363 | 0,609 ± 0,0833 | 2,665e-14 |
| IntTrCSA | Age, Age^2, Sex | 363 | 0,369 ± 0,0925 | 2,8e-06 |
| IntTrCSMI | Age, Age^2, Sex | 363 | 0,310 ± 0,0905 | 2,03e-05 |
| IntTrSMod | Age, Age^2, Sex | 363 | 0,345 ± 0,0952 | 1,27e-05 |
| NeckBMD | Age, Age^2, Sex | 364 | 0,473 ± 0,0976 | 1e-07 |
| NNeckACT | Age, Age^2, Sex | 363 | 0,451 ± 0,0988 | 3e-07 |
| NNeckBR | Age, Sex | 363 | 0,619 ± 0,0904 | 3,437e-12 |
| NNeckCSA | Age, Age^2, Sex | 363 | 0,282 ± 0,0917 | 2,816e-04 |
| NNeckCSMI | Age, Age^2, Sex | 363 | 0,292 ± 0,0888 | 1,204e-04 |
| NNeckSMod | Age, Age^2, Sex | 363 | 0,243 ± 0,0874 | 1,009e-03 |
| ShaftNeck | Age, Sex | 363 | 0,509 ± 0,1090 | 1,51e-09 |
| SpineT | Age, Age^2 | 364 | 0,451 ± 0,0991 | 1e-07 |
| SpineZ | | 358 | 0,680 ± 0,0858 | 5,205e-15 |
| TotBMD | Age, Age^2, | 365 | 0,465 ± 0,0968 | 4,591e-08 |

| | | | | |
|---|---|---|---|---|
| | Sex | | | |
| TrochBMD | Sex | 364 | 0,360 ± 0,0952 | 1,34e-05 |
| WBTotBMD | Age, Age^2, Sex | 365 | 0,295 ± 0,0875 | 6,92e-05 |
| WhBodyT | Age, Age^2 | 363 | 0,313 ± 0,0921 | 3,96e-05 |
| WhBodyZ | | 343 | 0,761 ± 0,0855 | 1,56e-17 |

*Table 5.5. Resultant polygenic models obtained for densitometric traits*

## 5.3.3. Polygenic models for affected traits

Now, if we look at the column of heritabilities we can note that the standard deviation has increased alarmingly for all traits. That is because the statistical power is significantly lower than in quantitative traits. In fact, our sample size is too small to determine accurately the heritability for each trait. The variability for binary traits is substantially lower, and therefore the differences among individuals aren't evident. Thus, we need a greater sample to infer the parameters accurately (as always, but in this case it is even more necessary).

| **Trait** | **Covariates** | **N of individuals considered** | **Heritability** $h^2 \pm std$ | **P-value heritability** | **R-squared** |
|---|---|---|---|---|---|
| Affected1 | Age, Sex | 362 | 0,494 ± 0,273 | 0,0264 | 0,249 |
| Affected2 | Age | 362 | 0,469 ± 0,389 | 0,103 | 0,123 |
| Affected3 | Age, $Age^2$ | 363 | 0,779 ± 0,196 | 5,10e-6 | 0,289 |
| Affected4 | Age | 362 | 0,568 ± 0,221 | 0,0042 | 0,178 |

*Table 5.6. Resultant polygenic models obtained for affected traits*

Furthermore, as we mentioned in the overview, now instead of "Proportion of variance", we have "R-squared". In fact, the feature to compute the proportion of variance is not available in SOLAR yet. In consequence, we should be careful with the R-squared since it has to be properly interpreted. Although the R-squared can be an indicator of how well fitted is our model, it always increases as more covariates we include in the model. Thus, if we obtain a high value for the R-

squared but we are using a large bunch of covariates the indicator might lead to wrong conclusions.

## 5.4. PCA comparison

### 5.4.1. Non-corrected vs corrected PCA in GAIT

Despite we are not going to work with the results of these PCAs obtained for the GAIT data, they are the perfect example to illustrate the difference between a common PCA and a "corrected" one by using the kinship matrix. In order to note the differences between them we have plotted a scoreplot of the two first principal components. In addition, we have given a different color to each family, for the purpose of visualizing how the individuals of each family are distributed. In fact, in the common PCA we can note that families are slightly clustered, because this source of non-random variance hasn't been modeled. As we can see in the first graph, families tend to be distributed along diagonal lines, instead of the second graph, where no patterns can be found. That is, applying the correction we are modelling accurately other internal sources of variance and eliminating non-randomness due to familiy clustering.

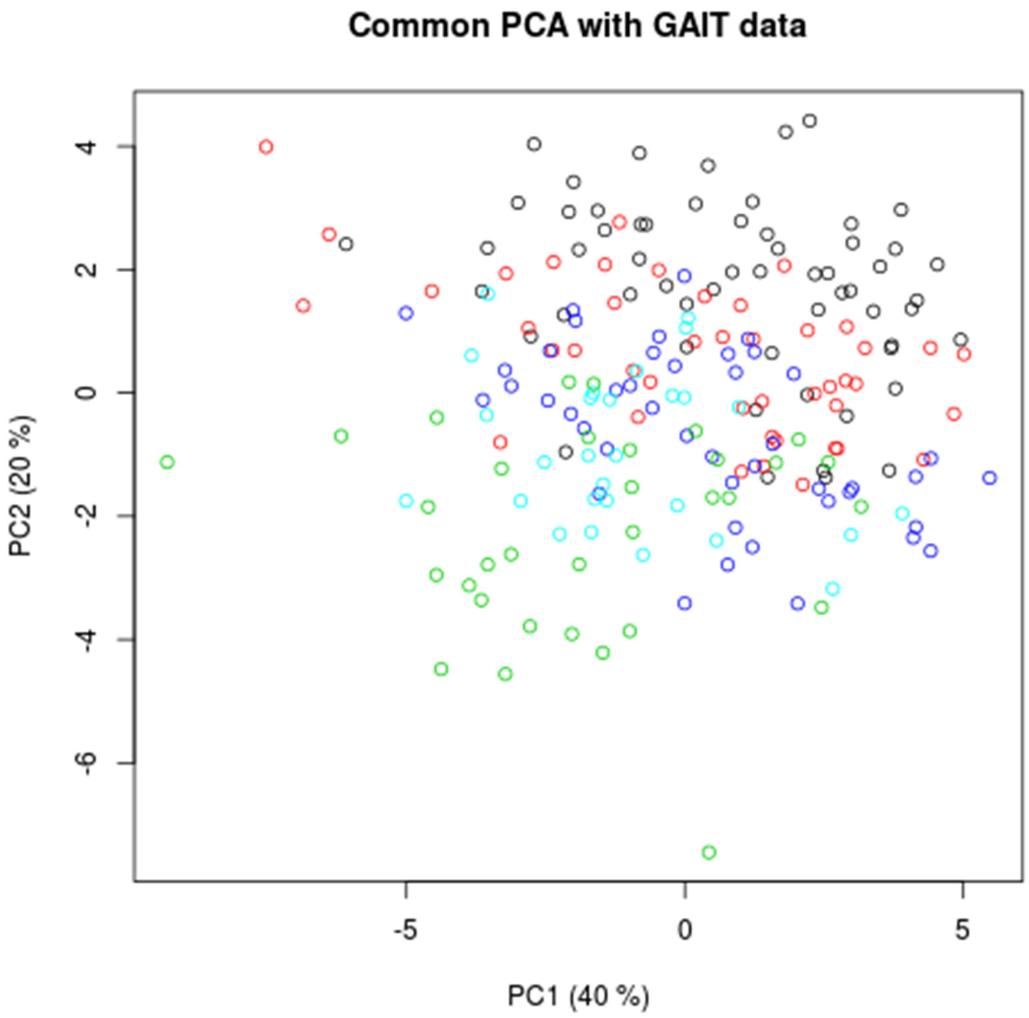



*Figure 5.4. PCA non-corrected for the GAIT data. Each color corresponds to a different family.*

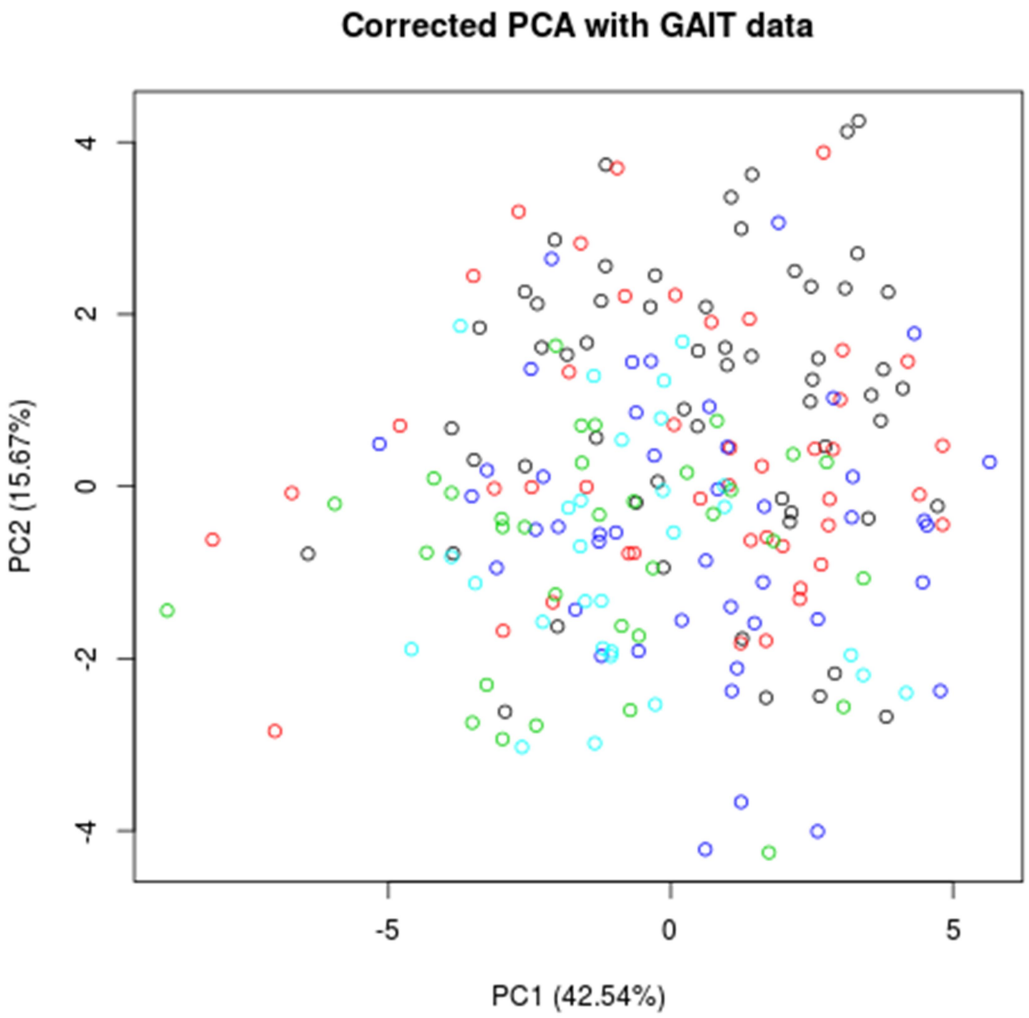



*Figure 5.5. PCA corrected by the kinship matrix for the GAIT data. Each color corresponds to a different family.*

## 5.4.2. Heat map for the densitometric traits

In order to visualize graphically the correlations of the variables included in the PCA made for the GAO data we have plotted a heat map of the traits. As we can see, there are traits which are strongly correlated (as FemShBR and NNeckBR), and we would state that the average

correlation is moderately high, which is a positive thing for the PCA (more variance explained by the first principal components and therefore more power).

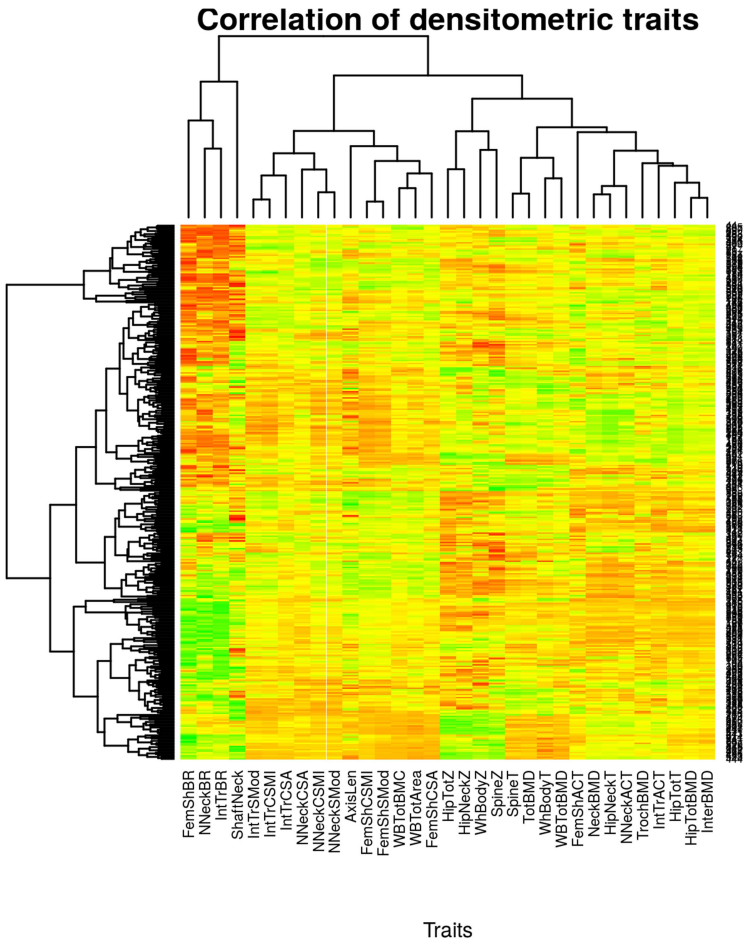



*Figure 5.6. Heat map showing the correlations existing among the densitometric traits included in the PCA.*

## 5.4.3. Non-corrected vs corrected PCA in GAO

In this case the differences between the corrected PCA and the common one are not as evident as they were for the GAIT data. However, there are still differences between both PCAs, so the effect of correcting by the kinship matrix is also significant.

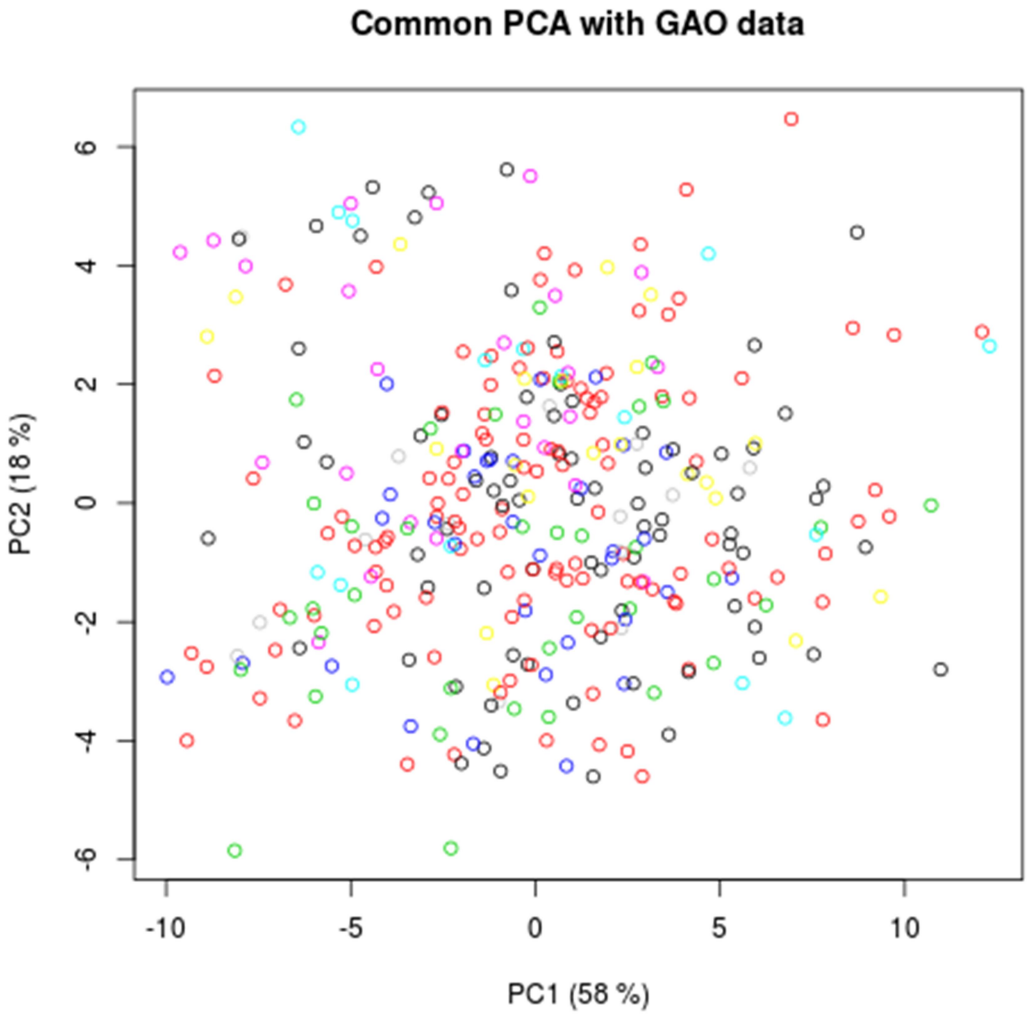



*Figure 5.7. PCA non-corrected for the GAO data. Each color corresponds to a different family*

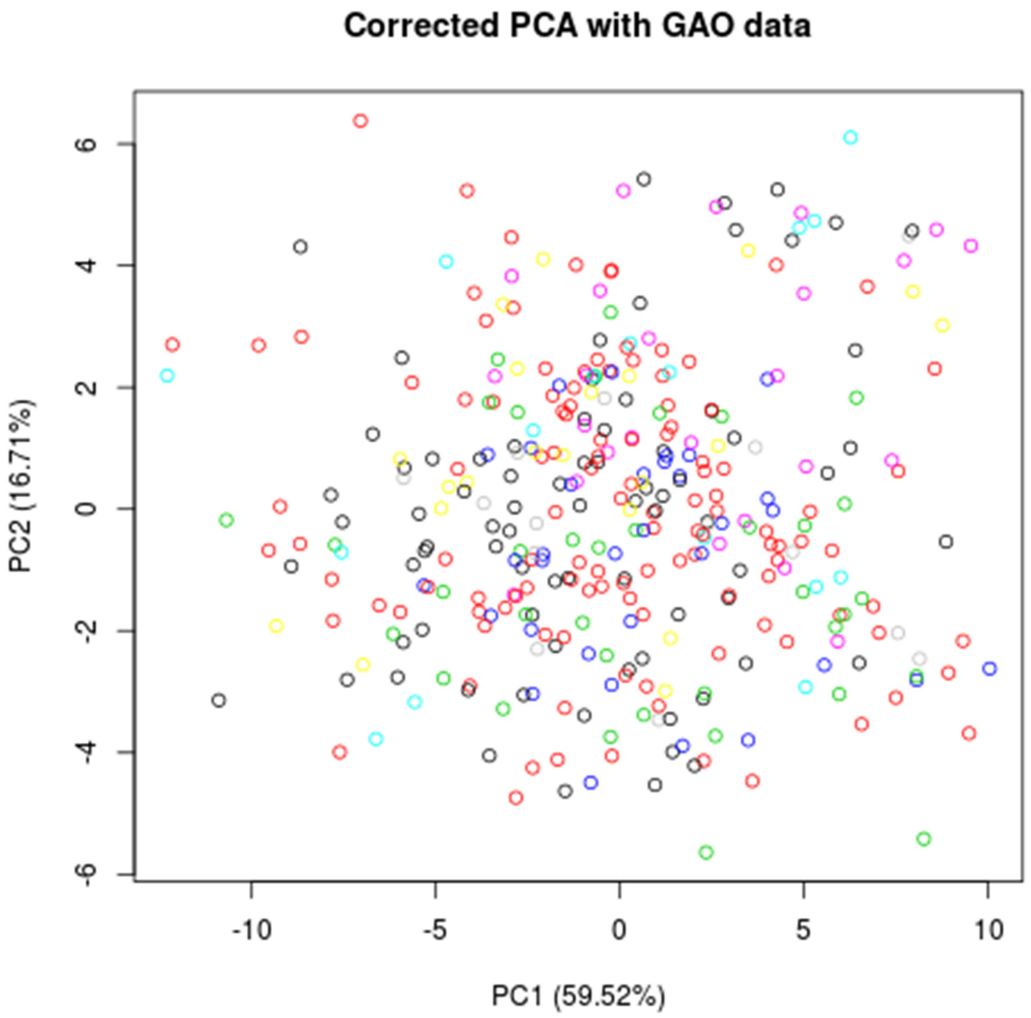



*Figure 5.8. PCA corrected by the kinship matrix for the GAO data. Each color corresponds to a different family.*

## 5.5. Results from associations

### 5.5.1. Direct associations

In this section we are going to present the results obtained in direct associations with the phenotypes. We are going to show mainly those associations where we have found significant

SNPs, nevertheless in some cases we haven't found anything significant, which leads to present the most interesting suggestive results. Typically the tables presented in this section contain the SNP with the lowest pvalue (significant) and a bunch of non-significant (generally suggestive) SNPs that are in high linkage disequilibrium with the significant SNP. Furthermore, we have attached a table with genomic inflation factors for the pvalues of the traits of interest. The manhattan and QQ plots used to find and verify significant SNPs are collected in the annex section A.4.

## Bone metabolism markers (no covariates)

### Leptin

For this protein one SNP happened to be between the genome wide association line and the common Bonferroni line.

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP |
|---|---|---|---|---|---|---|---|---|---|
| rs714962 | 22 | 6,5027e-08 | snp | LOC105377195 | C/T | C | T | 0,499 | 29397845 |
| rs3788412 | 22 | 3,352e-06 | snp | AP1B1 | C/T | T | C | 0,432 | 29354196 |
| rs5752907 | 22 | 3,3252e-06 | snp | LOC105377195 | C/T | T | C | 0,441 | 29393031 |
| rs2301587 | 22 | 4,8910e-06 | snp | AP1B1 | C/T | T | C | 0,149 | 29341833 |

*Table 5.7. Top SNPs' table of chromosome 22 for Leptin's association*

### OstaseBAP

Apart from the Leptin protein, we haven't found more significant SNPs associated with the proteins traits. Actually, these are probably the second most interesting results for these traits, despite they are suggestive.

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP |
|---|---|---|---|---|---|---|---|---|---|
| rs2659187 | 5 | 2,272 2e-07 | snp | | C/T | C | T | 0,480 | 173857929 |
| rs359470 | 5 | 1,236 9e-06 | snp | | C/T | C | T | 0,452 | 173873683 |
| rs359467 | 5 | 8,647 4e-06 | snp | CPEB 4 | A/G | A | G | 0,448 1 | 173889725 |

*Table 5.8. Top SNPs' table of chromosome 5 for OstaseBAP's association*

### Genomic control

Looking at the inflation factor of both traits, as well as the Q-Q plots, all is within normal parameters, validating the results presented before.

| Traits | Inflation factor |
|---|---|
| HydVitD | 1,01872 |
| Sclerostin | 1,046627 |
| SerCrossLaps | 1,042657 |
| OstaseBAP | 1,042691 |
| IGF1 | 1,014283 |
| Adiponectin | 1,054512 |
| Leptin | 1,010495 |
| Osteocalcin | 1,040014 |
| Osteoprotegerin | 1,041537 |

| | |
|---|---|
| Osteopontin | 1,034599 |
| Parathyroid | 1,025272 |
| TNFalpha | 1,029288 |

*Table 5.9. Genomic inflation factors of the pvalues obtained in the association without covariates for bone metabolism markers*

## Bone metabolism markers (covariates)

### SerCrossLaps

We haven't found any significant SNP in the analysis with covariates. However, the most interesting suggestive SNPs that we found are presented here.

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP |
|---|---|---|---|---|---|---|---|---|---|
| rs3217 | 4 | 7,6407e-07 | snp | ZNF518B | C/T | C | T | 0,2183 | 10443025 |
| rs442241 | 4 | 7,6407e-07 | snp | ZNF518B | G/T | T | G | 0,2187 | 10441759 |
| rs17467273 | 4 | 1,8701e-06 | snp | CLNK | C/T | T | C | 0,2035 | 10498806 |

*Table 5.10. Top SNPs' table of chromosome 4 for SerCrossLaps association*

### Genomic control

And the inflation factor of the p-values obtained for this trait is:

| Traits | Inflation factor |
|---|---|
| SerCrossLaps | 1,04562 |

*Table 5.11. Genomic inflation factor of SerCrossLaps trait*

## Densitometric traits

### FemShBR

We have found a clear association for this trait. An indicator which shows that the association in this case is quite strong is the fact that there is a bunch of non-significant SNPs in linkage disequilibrium with the main SNP that are above the suggestive line. It should be commented as well that the main SNP found here also has a very small p-value for the trait FemShACT, which is a trait very similar to this one. In fact, the SNP rs11060592 happened to be significant almost twice, which validates its significance (unintentionally we have almost replicated the result obtained here). Considering that we have found a strong association for this trait, we have plotted the Manhattan plot, apart from attaching the top SNPs table.

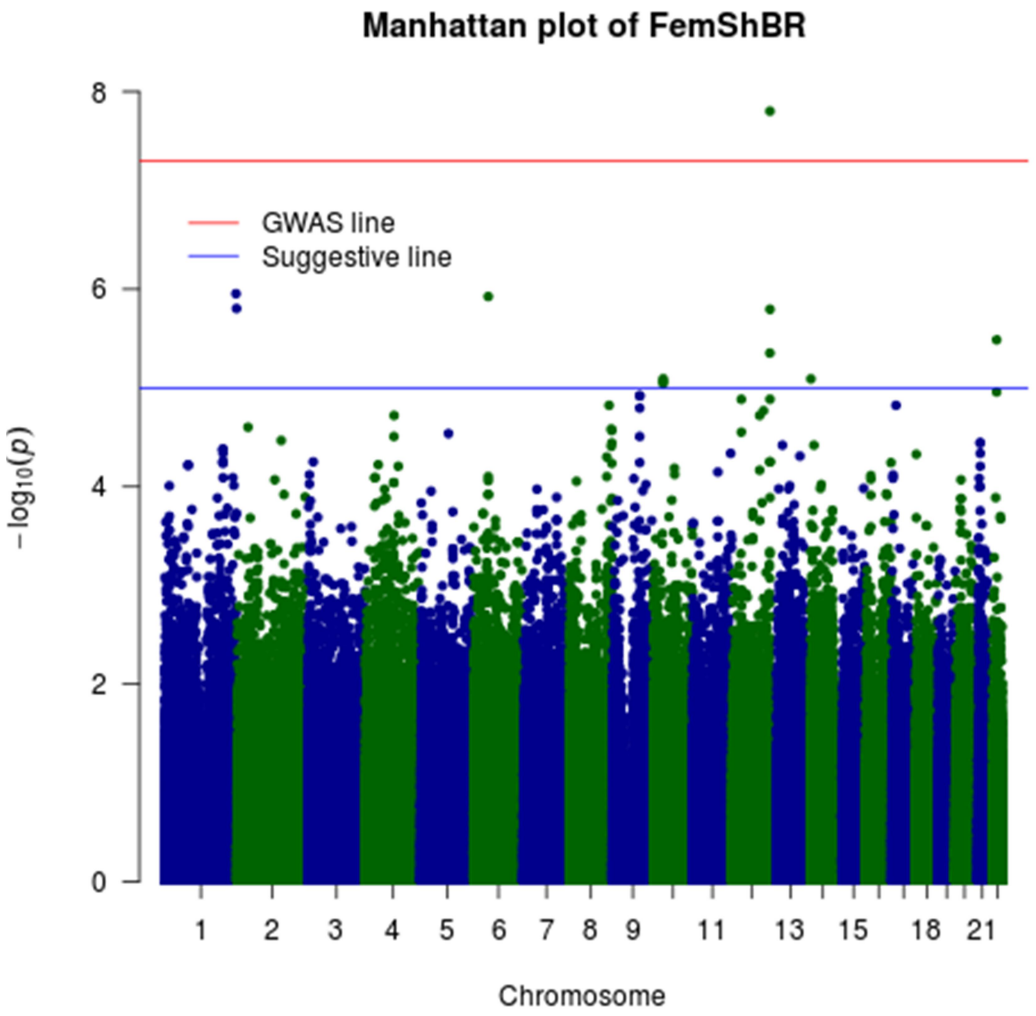



*Figure 5.9. Manhattan plot of FemShBR trait*

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP |
|---|---|---|---|---|---|---|---|---|---|
| rs11060592 | 12 | 1,5756e-08 | snp | TMEM132D | C/T | T | C | 0,348 | 129864271 |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| rs10847960 | 12 | 1,5989e-06 | snp | TMEM132D | A/G | G | A | 0,188 | 129842878 |
| rs10847961 | 12 | 4,4347e-06 | snp | TMEM132D | A/G | A | G | 0,356 | 129869803 |
| rs10847963 | 12 | 1,3000e-05 | snp | TMEM132D | C/T | T | C | 0,201 | 129876987 |

*Table 5.12. Top SNPs' table of chromosome 12 for FemShBR's association*

### NNeckBR

For this trait we have found a single SNP with a p-value almost equal to the genome wide threshold but above the common Bonferroni line. However, this SNP will be difficult to interpret, since it is isolated and there are no suggestive SNPs in high linkage disequilibrium with it.

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP |
|---|---|---|---|---|---|---|---|---|---|
| rs11770631 | 7 | 5,0906e-08 | snp | | C/T | T | C | 0,187 | 155864785 |
| rs12701863 | 7 | 7,8729e-07 | snp | LINC01449 | C/T | T | C | 0,439 | 41101669 |
| rs12701864 | 7 | 7,8729e-07 | snp | LINC01449 | A/G | A | G | 0,370 | 41103345 |

*Table 5.13. Top SNPs' table of chromosome 7 for NNeckBR's association*

### Genomic control

The inflation factors are very near to 1 for all traits and the qq-plots have a nice aspect.

| Phenotype | Inflation factor |
|---|---|
| FemShBR | 1,022833 |
| FemShACT | 1,0308015 |
| NNeckBR | 1,0180046 |

*Table 5.14. Genomic inflation factors of the pvalues obtained in the association with covariates for densitometric traits*

## Phenotypes affected

### Affected3

For the phenotypes "affected" no sihgnificant associations have been found, probably because our sample size is too reduced considering that they are binary traits. Nevertheless, as we pointed out at first, we should also report those suggestive SNPs that could be of interest. In this case, we have considered the snp rs3827306 of the Affected3 trait as the most interesting one, because the results found for the Affected2 phenotype are unacceptable due to their high inflation factor.

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP |
|---|---|---|---|---|---|---|---|---|---|
| rs3827306 | 22 | 3,7243e-07 | snp | LL22NC03-63E9.3 | A/G | G | A | 0,1611 | 22561382 |

*Table 5.15. Top SNPs' table of chromosome 22 for Affectetd3's association*

### Genomic control

As we can see, the results obtained from the association with the Affected2 trait are inflated and can't be considered. In consequence, we have to look for other suggestive interesting SNPs in the remaining associations.

| Phenotype | Inflation factor |
|---|---|
| Affected1 | 1,040944 |
| Affected2 | 1,147128 |
| Affected3 | 1,017010 |
| Affected4 | 1,040732 |

*Table 5.16. Genomic inflation factors of the pvalues obtained in the association with covariates for Affected traits*

## 5.5.2. Association with Principal Components

The results obtained in the otherwise known as “indirect association study” are presented in this section, nevertheless the most important thing that should be considered are the loadings of the Principal Components presenting significant SNPs, in order to interpret if we are facing noise or a signal actually. Apart from this, the tables shown in this section are the same that we reported for the direct associations results. The Manhattan plots and Q-Q plots used to explore these results are collected in the annex A.4.

### PC9

The first principal component under analysis, PC9 seems to have reasonable loadings, where traits very related with bone structure rather than bone density play an important role. In fact, NNecKCSMI, FemShCSMI, NNecrBR, IntTrACT, FemShMod, NNeckSMod, FemShCSA... are all of them measures of bone structure, for instance: bulking ratio, section, moment of inertia, binding strength, and so on. Furthermore, the third greatest weight is NNeckBR, which is one of the traits where we had found significant SNPs when we associated with clinical phenotypes.

| Trait | Weights PC9 |
|---|---|
| NNeckCSMI | 0,338217849 |
| FemShCSMI | 0,326602419 |
| NNeckBR | 0,305341243 |
| IntTrACT | 0,291263757 |
| FemShSMod | 0,289369048 |
| WhBodyZ | 0,285243416 |
| NNeckSMod | 0,260604975 |
| FemShCSA | 0,225532524 |
| HipTotT | 0,224035789 |
| NNeckCSA | 0,190987503 |
| InterBMD | 0,186517061 |
| IntTrBR | 0,176040543 |

| | |
|---|---|
| HipTotZ | 0,162602002 |
| FemShBR | 0,137584142 |
| SpineT | 0,132662124 |
| ShaftNeck | 0,131395514 |
| AxisLen | 0,124794969 |
| WBTotBMD | 0,113769458 |
| NNeckACT | 0,105832037 |
| NeckBMD | 0,096597680 |
| HipTotBMD | 0,088816483 |
| TrochBMD | 0,078289102 |
| WBTotBMC | 0,075928590 |
| SpineZ | 0,062852295 |
| IntTrCSA | 0,053255783 |
| TotBMD | 0,052968488 |
| HipNeckZ | 0,048648064 |
| WhBodyT | 0,043527953 |
| IntTrCSMI | 0,032938807 |
| WBTotArea | 0,017912065 |
| IntTrSMod | 0,010259006 |
| FemShACT | 0,007758922 |
| HipNeckT | 0,007567443 |

*Table 5.17. Sorted absolute values of loadings of the 9$^{th}$ Principal Component*

In fact, is for this component that we have found the strongest association of the whole study. Because of this, we have decided to plot the Manhattan plot as well, apart from attaching the table with the top associated SNPs.

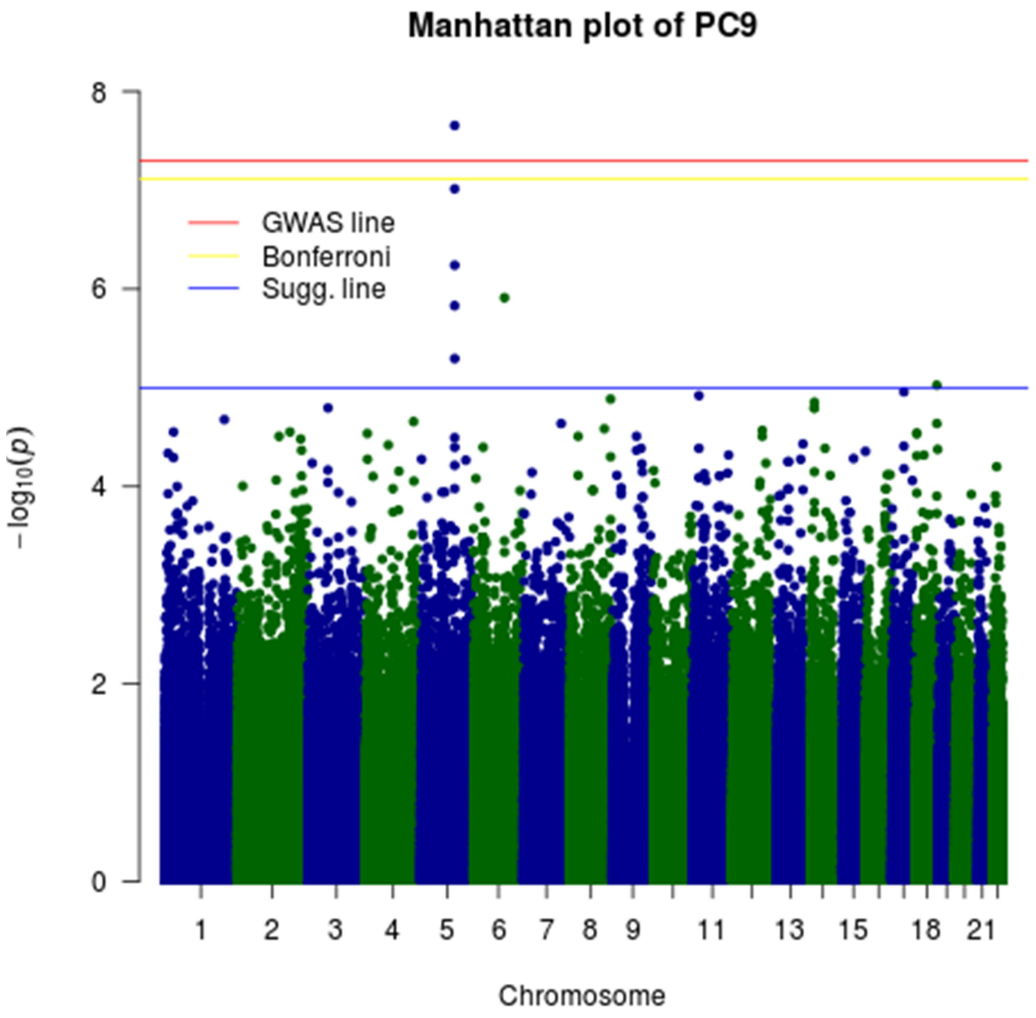



*Figure 5.10. Manhattan plot of the 9th Principal Component*

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP |
|---|---|---|---|---|---|---|---|---|---|
| rs17139820 | 5 | 2,2099e-08 | snp | SEMA6A | C/T | C | T | 0,3003 | 116452715 |
| rs26602 | 5 | 9,6752e-08 | snp | SEMA6A | A/G | A | G | 0,3768 | 116452407 |
| rs2303752 | 5 | 5,7414e-07 | snp | SEMA6A | C/T | C | T | 0,1721 | 116449815 |
| rs17139903 | 5 | 1,4740e-06 | snp | SEMA6A | C/T | C | T | 0,2432 | 116485518 |

*Table 5.18. Top SNPs' table of chromosome 5 for PC9's association*

## PC12

The loadings for this principal component aren't as clear as they were for component 9, nevertheless it seems that in this case Bone Mineral Density has an important role. In fact, T and Z measurements, which are scores of bone mineral density as well as Troch BMD hold 5 out of the first 8 greatest loadings. The snps that happen to be significant in this association might be more related with bone mineral density rather than the structure of bones.

| Trait | Weights PC12 |
|---|---|
| HipNeckT | 0,357463138 |
| NNeckBR | 0,352246345 |
| AxisLen | 0,347644031 |
| HipNeckZ | 0,283301438 |
| HipTotT | 0,270856207 |
| NNeckSMod | 0,251393899 |
| TrochBMD | 0,244458980 |

| | |
|---|---|
| WhBodyT | 0,227444453 |
| WBTotArea | 0,222087318 |
| WBTotBMD | 0,216416796 |
| NNeckACT | 0,189471695 |
| IntTrBR | 0,181925479 |
| SpineZ | 0,156657468 |
| FemShACT | 0,153828255 |
| NNeckCSA | 0,151474785 |
| TotBMD | 0,135789497 |
| SpineT | 0,118593544 |
| HipTotZ | 0,087830685 |
| NeckBMD | 0,078114807 |
| NNeckCSMI | 0,069784590 |
| FemShCSA | 0,063335570 |
| HipTotBMD | 0,055295171 |
| WhBodyZ | 0,038988223 |
| IntTrSMod | 0,032903758 |
| FemShBR | 0,030837568 |
| FemShSMod | 0,017813748 |
| IntTrACT | 0,013005477 |
| IntTrCSMI | 0,010855905 |
| IntTrCSA | 0,010032353 |
| InterBMD | 0,009861414 |
| FemShCSMI | 0,005398266 |
| ShaftNeck | 0,003722286 |
| WBTotBMC | 0,003399836 |

*Table 5.19. Sorted absolute values of loadings of the 12$^{th}$ Principal Component*

And the top SNPs table:

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP |
|---|---|---|---|---|---|---|---|---|---|
| rs11770919 | 7 | 9,0854e-08 | snp | MAGI2 | A/G | A | G | 0,0645 | 78283558 |
| rs699332 | 7 | 5,5483e-06 | snp | MAGI2 | G/T | T | G | 0,3017 | 78246092 |
| rs1852008 | 7 | 3,2000e-05 | snp | MAGI2 | A/C | A | C | 0,1813 | 78237393 |

*Table 5.20. Top SNPs' table of chromosome 7 for PC12's association*

## PC29

This is one of the last principal components and therefore it is a strong candidate to be capturing noise rather than a real signal. In fact, the loadings for this component are the most confusing ones, because two kinds of traits are strongly mixed. The greatest loadings in this case are bone mineral density indicators as well as structural traits, which implies that maybe the SNP found in the association with this component has a more overall effect in Osteoporosis.

| Trait | Weights PC29 |
|---|---|
| FemShSMod | 0,500219420 |
| FemShCSMI | 0,448357811 |
| WBTotBMC | 0,294602145 |
| TotBMD | 0,250753808 |
| WhBodyT | 0,248423402 |
| IntTrCSA | 0,241072832 |
| IntTrACT | 0,211662790 |
| SpineT | 0,198085336 |

| | |
|---|---|
| WBTotArea | 0,188652553 |
| HipTotT | 0,172509185 |
| FemShCSA | 0,158568019 |
| IntTrSMod | 0,157609897 |
| WBTotBMD | 0,146318578 |
| HipTotBMD | 0,110560657 |
| IntTrCSMI | 0,106694295 |
| NNeckCSMI | 0,103441389 |
| NNeckSMod | 0,083551416 |
| HipNeckT | 0,069289935 |
| NeckBMD | 0,061930883 |
| NNeckACT | 0,056618995 |
| NNeckCSA | 0,037459990 |
| IntTrBR | 0,037212340 |
| FemShACT | 0,027802841 |
| TrochBMD | 0,019111740 |
| InterBMD | 0,017027120 |
| SpineZ | 0,016847032 |
| AxisLen | 0,016348641 |
| HipTotZ | 0,014071812 |
| HipNeckZ | 0,012454937 |
| WhBodyZ | 0,011937685 |
| ShaftNeck | 0,006088069 |
| FemShBR | 0,005893771 |
| NNeckBR | 0,003221282 |

*Table 5.21. Sorted absolute values of ladings of the 29$^{th}$ Principal Component*

The table of SNPs presenting the lowest pvalues:

ETSEIB

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP |
|---|---|---|---|---|---|---|---|---|---|
| rs4271181 | 1 | 8,3389e-08 | snp | | A/G | G | A | 0,3776 | 22607 8327 |

*Table 5.22. Top SNPs' table of chromosome 1 for PC29's association*

### Genomic Control

The genomic control of these three associations is within the correct values, which validates the results. Although the inflation factor for the PC12 is a bit higher than usual (considering that the limit is on 1.1), is still far from the undesired 1.1. The Q-Q plots also have normal aspect, with the pvalues either concentrated over the straight line or near to it.

| Phenotype | Inflation factor |
|---|---|
| PC9 | 1,033591 |
| PC12 | 1,0748962 |
| PC29 | 1,0357043 |

*Table 5.23. Genomic inflation factors of the pvalues obtained in the association with covariates for the resultant Principal Components*

# 6. Discussion

In this section we are going to discuss the results obtained in the association analysis from the biological point of view. Nevertheless, in some cases the results found despite their mathematical consistency don't make sense in biologic terms.

The markers that we are going to discuss are collected in the following table:

| Marker | Chr | p-value | Class | Gene | Alleles | Major | Minor | MAF | BP | Trait |
|---|---|---|---|---|---|---|---|---|---|---|
| rs7149 | 22 | 6,502 | snp | LOC1 | C/T | C | T | 0,49 | 29397 | Leptin |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 62 | | 7e-08 | | 05377195 | | | | 90 | 845 | |
| rs11060592 | 12 | 1,575 6e-08 | snp | TMEM132D | C/T | T | C | 0,3478 | 129864271 | FemShBR |
| rs11770631 | 7 | 5,090 6e-08 | snp | NA | C/T | T | C | 0,1869 | 155864785 | NNeckBR |
| rs17139820 | 5 | 2,209 9e-08 | snp | SEMA6A | C/T | C | T | 0,3003 | 116452715 | PC9 |
| rs11770919 | 7 | 9,085 4e-08 | snp | MAGI2 | A/G | A | G | 0,0645 | 78283558 | PC12 |
| rs4271181 | 1 | 8,338 9e-08 | snp | NA | A/G | G | A | 0,3776 | 226078327 | PC29 |

*Table 6.1. The most significant SNPs found in the whole set of analysis*

In order to find biological sense to the most significant SNPs found, we have first looked at the regions where the SNPs are placed, paying special attention at those ones which are placed in a coding region. In that case, we comment the function of the genes, we look for possible related pathways and we suggest a possible relation with Osteoporosis. However, if the SNP is not placed in a coding gene we comment the type of region where it is placed, and the role that this region may be performing (for instance regulation of gene's expression).

In the first place we have the snp rs17139820 (chromosome 5), which is placed within the gene SEMA6A. The transmembrane semaphorin SEMA6A is a protein-coding gene, expressed in development of neural tissue, concretely of the thalamocortical projection [43]. This gene is especially present during the embryo development, thus only small amounts of SEMA6A transcripts were detected in human adult tissues. This gene belongs to the axon guidance pathway and seems to have no relation with Osteoporosis. This is a typical case of mathematical consistency but biological inconsistency.

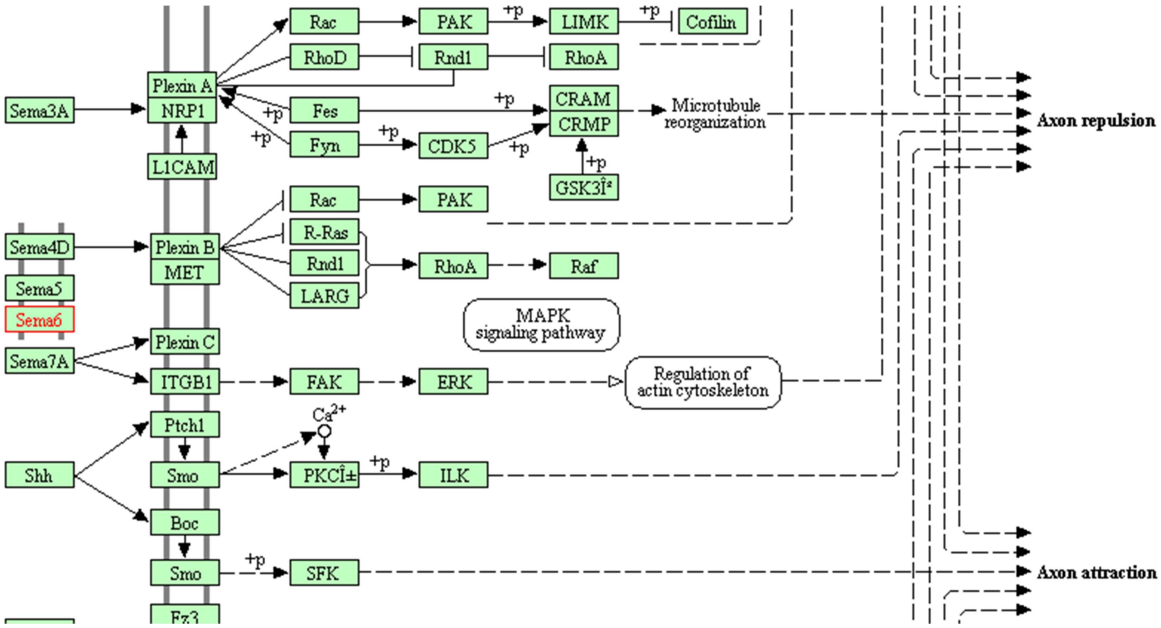



*Figure 6.1. Axon guidance pathway of the SEMA6A. Image extracted from KEGG pathways (http://www.kegg.jp/kegg-bin/highlight_pathway?scale=1.0&map=hsa04360&keyword=SEMA6A)*

In the second place, we have the snp rs11770919 (chromosome 7), which is also placed within a coding gene, concretely the MAGI2. The proteins encoded by this gene, known as membrane-associated guanylate kinase are essential for development and maintenance of synapses, including receptor endocytosis and postendocytotic trafficking. MAGI2 dependent endocytosis is also essential for ciliogenesis [44]. Cilioginesis is defined as the building of the cell's antenna (primary cilia) or extracellular fluid mediation mechanism (motile cilium). Cilia are important organelles of cells that are involved in numerous activities, as cell signaling, processing developmental signals, and directing the flow of fluids such as mucus over and around cells. Therefore, a dysfunction of the cilia may lead to problems of cell-adhesion and cell-junction formation. If we move to our context (Osteoporosis) we can suggest that bone related cells may be affected by these problems. That is, the bone cells may not be able to get stuck to the bone as they usually do. In fact, if we look at the related pathway of the MAGI2, we can see that the gene is near to the calcium signaling pathway. This one has been the most conclusive biological result that we have found in this study,

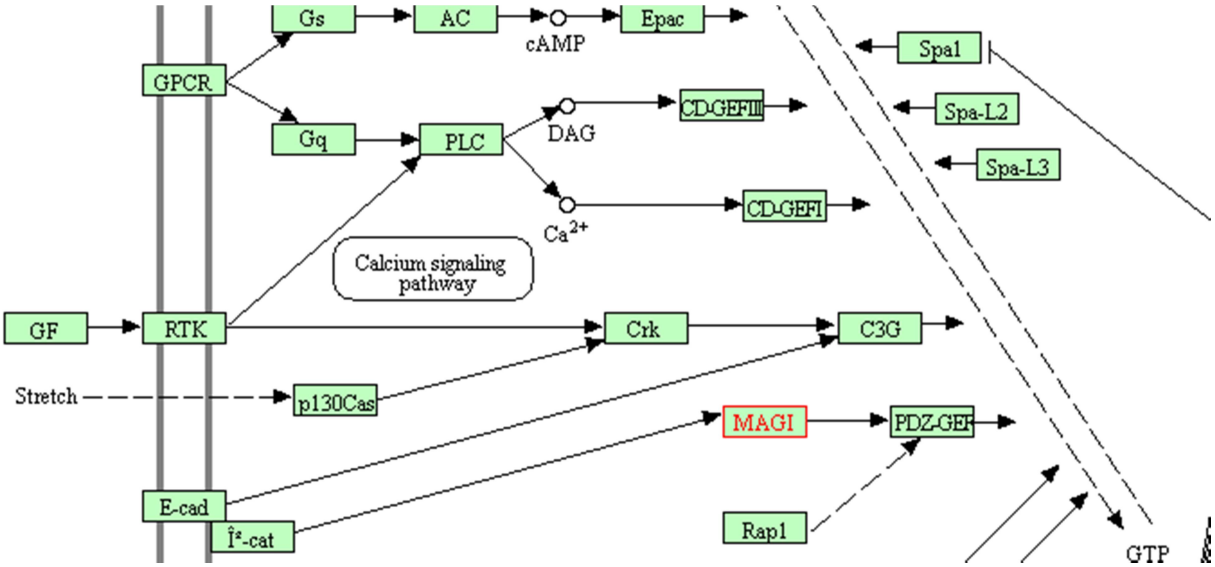



*Figure 6.2. Part of the Rap1 signaling pathway. Image extracted from KEGG pathways (http://www.kegg.jp/kegg-bin/highlight_pathway?scale=1.0&map=hsa04015&keyword=MAGI2)*

In fact the rs11770631 (chromosome 7) and rs427181 (chromosome 1) are placed in intergenic regions, far away from any gene, thus no function could have been associated to them so far. The rs714962 (chromosome 22) is placed within LOC105377195, which is a RNA gene[45], that may have some regulatory effect, though it is unlikely to be related with Osteoporosis considering the zone of the genome where it is placed. Finally, despite the rs11060592 (chromosome 12) is placed within a coding gene, concretely the TMEM132D, no relation with Osteoporosis has been found. Actually, this gene is related with the conformation of the cerebral cortex, primarily in the white matter formation[46].

At last, just comment that we have compared our results with the actual State of the art in Osteoporosis through the GWAS catalog and there were no coincidences.

# 7. Economic, environmental and social impact

## 7.1. Economic analysis

In this section we have made an approach of the overall cost of the project, making some assumptions about the prices of the elements involved. Also, we have considered that projected lasted 15 weeks.

- The student work lasted for 7h, five days a week valued at 8€ per hour (if the student would have done an internship in an enterprise, the legal agreement rules that the maximum compensation could be 8€/h).
- The supervisor work involved 2h per week due to project guidance and orientation. Every hour is worth 40€.
- The personal computer was acquired for 750 € and have a 5 years lifespan
- The computational cluster cost is 10.000€ and will be replaced after 5 years from its acquisition.
- Office furniture was purchased at 150€ and will last for 15 years
- Office material was bought for 40€ and will be consumed for 2 years.
- The average energy consumption is estimated by 350W. It includes a barebone, one screen, and the proportional part of office lights, cluster processing units and cluster cooling system consumption. We assumed that 1kWh costs 0.13907 € after taxes.

| Concept | Lifespan [y] | Acquisition cost [€] | Fixed cost [€] | Usage [h] | Variable cost[€/h] | Total |
|---|---|---|---|---|---|---|
| Student work | | | | 525 | 8 | **4200** |
| Supervisor work | | | | 30 | 40 | **1200** |
| Personal computer | 5 | 750 | 43,3 | | | **43.3** |
| Cluster | 5 | 10000 | 576,92 | | | **576.92** |
| Office furniture | 15 | 300 | 5.77 | | | **5.77** |
| Office material | 2 | 30 | 4,33 | | | **4.33** |
| Energy consumption | | | | 525 | 0,048674 | **25,55** |
| **TOTAL** | | | | | | **6055,87** |

*Table 7.1. Economic analysis estimation*

## 7.2. Environmental impact

Considering that this project has been entirely developed using computer systems, its environmental impact is minimal. So for this case we only have to consider the CO2 emissions due to the production of our energy consumption (400W for 7h per day, 5 days per week and 15 weeks in total).

| Contaminant | Energy Consumption | Specific amounts | Total amounts |
|---|---|---|---|
| Carbon dioxide | 183,75 kWh | 0,302kg/kWh | **55,49 kg** |
| Radioactive waste | 183,75 kWh | 0,56mg/kWh | **102,9 mg** |

*Table 7.2. Pollution analysis estimation. Specific amounts have been extracted from www.gencat.cat.*

## 7.3. Social impact

This project does not directly create a negative impact on any sector or group. On the contrary, the results obtained in this project may help in the future to improve life's quality of people suffering from Osteoporosis. Nevertheless, since this is a biomedical project with real data from different patients, we have to properly preserve their anonymity. Our procedures included safe programming environments as well as encrypted *ssh* connections in order to protect the data. This work meets the requirements for biomedical investigation in Spain.

# 8. Conclusions

## 8.1. Objectives reviewing

In general terms, we can state that we have fulfilled the objectives presented in section 1.3. In fact, the methods applied to perform the Genome-Wide Association Study have succeeded in finding possible markers related with Osteoporosis. Despite we haven't imputed the genotyped data, losing statistical power, we have overcome this problem. Actually, imputation is a highly technical procedure that has no interest itself, though it requires time and expertise to be performed correctly. In our case, we had neither of these things, and even though it was a bit daring to try an association with no imputation done, at last it has been worth the risk.

However, the path hasn't been clear at all, and the most conclusive results have arrived when we embraced the Principal Component Analsys technique. Thus, this procedure has increased the statistical power of the analysis, which was quite limited considering our sample size. Furthermore, the results obtained apart from being consistent from the mathematical point of view, may make biological sense. Although they haven't been overwhelming, they have presented credentials for a suggestive relation with Osteoporosis.

In conclusion, all the procedures that we have followed and proposed in this project have led to interesting results that need further investigation and interpretation.

## 8.2. Further work

In this section we suggest possible ways to dig deeply in this study that may lead to new and exciting results.

- The first and more obvious one is associating with imputed data
- Association with more depurated polygenic models, using new covariates in order to capture the maximum amount of variance.
- Association with the Principal Components obtained in a PCA where the densitometric traits and possible covariates were different from those chosen in this project.
- Association with the Principal Components obtained in a PCA where only were considered the bone metabolism markers
- Association with the Principal Components obtained in a mixed PCA (using bone metabolism markers and densitometric traits)

# 9. References

[1] Maria Sabater-Lleal, Angel Martinez P. "A genome wide association study identifies KNG1 as a genetic determinant of plasma factor XI level and activated partial thromboplastin time". *Artherioscler Thromb Vasc Biol.*

[2] Georgios Athanasiadis, Alfonso Buil. “A genome-wide association study of the protein C anticoagulant pathway”. *PlosOne.*

[3] Wikipedia, June 2016. https://en.wikipedia.org/wiki/Genome-wide_association_study

[4] Georgios Athanasiadis, Jorge Malouf. “Association and linkage analyses using families identified a locus affecting an osteoporosis-related trait”. *Bone.*

[5] Wikipedia, June 2016. https://en.wikipedia.org/wiki/Osteoporosis

[6] Manolio TA; Guttmacher, Alan E.; Manolio, Teri A. (July 2010). "Genomewide association studies and assessment of the risk of disease". *N. Engl. J. Med*

[7] Sherry, S. T.; Ward, M. H.; Kholodov, M; Baker, J; Phan, L; Smigielski, E. M.; Sirotkin, K (2001). "DbSNP: The NCBI database of genetic variation". *Nucleic Acids Research*

[8] Scheet, Paul; Stephens, Matthew (2006). "A Fast and Flexible Statistical Model for Large-Scale Population Genotype Data: Applications to Inferring Missing Genotypes and Haplotypic Phase". *The American Journal of Human Genetics.*

[9] The R-project for Statistical Computing, June 2016. https://www.r-project.org/

[10] SOLAR-eclipse: an imaging genetics analysis software June 2016. http://solar-eclipse-genetics.org/

[11] PLINK: Whole Genome Analysis Toolset. June 2016.

http://pngu.mgh.harvard.edu/~purcell/plink/

[12] International Human Genome Sequencing Consortium (2001). “Initial sequencing and analysis of the human genome”.. *Nature*

[13] Nature, June 2016. http://www.nature.com/scitable/definition/single-nucleotide-polymorphism-snp-295

[14] Georgios Athanasiadis, Laura Arranz. “Exploring correlation of bone metabolism markers with densitometric traits and osteoporotic disease”. *Bone.*

[15] PLINK: Whole Genome Analysis Toolset. June 2016.

http://pngu.mgh.harvard.edu/~purcell/plink/data.shtml#bed

[16] Carl A. Anderson, Frederik H Petterson, Geraldine M Clarke. “Data Quality Control in genetic case-control association studies”. *Nature Protocol.*

[17] Wikipedia, June 2016. https://en.wikipedia.org/wiki/Minor_allele_frequency

[18] Dr. H. Brunel, Dr. A. Buil. “Technical report. Genotype imputation- The GAIT2 Project”. Geneva, August 2014.

[19] PLINK: Whole Genome Analysis Toolset. June 2016.

http://pngu.mgh.harvard.edu/~purcell/plink/summary.shtml#freq

[20] Wikipedia, June 2016. https://en.wikipedia.org/wiki/Mendelian_error

[21] PLINK: Whole Genome Analysis Toolset. June 2016.

http://pngu.mgh.harvard.edu/~purcell/plink/summary.shtml#mendel

[22] PLINK: Whole Genome Analysis Toolset. June 2016.

http://pngu.mgh.harvard.edu/~purcell/plink/summary.shtml#missing

[23] Summer Institute in Statistical Genetics (2013). Allelle frequencies and HWE.

[24] PLINK: Whole Genome Analysis Toolset. June 2016.

http://pngu.mgh.harvard.edu/~purcell/plink/thresh.shtml#hwd

[25] PLINK: Whole Genome Analysis Toolset. June 2016.

http://pngu.mgh.harvard.edu/~purcell/plink/dataman.shtml#extract

[26] PLINK: Whole Genome Analysis Toolset. June 2016.

http://pngu.mgh.harvard.edu/~purcell/plink/dataman.shtml#recode

[27] Bob Winter. “Linear models and linear mixed effects models in R with linguistic applications”.

[28] Wikipedia, June 2016. https://en.wikipedia.org/wiki/Multilevel_model

[29] Departament d’Estadística i investigación operativa. “Tema 12. Regressió múltiple”. *Atenea UPC.*

[30] GitHub, June 2016. Tutorial on “solarius” R package.

http://ugcd.github.io/solarius/vignettes/tutorial.html

[31] Lynch and Walsh 1998. “Genetics and Analysis of Quantitative Traits”. Chapters 26-27.

[32] Lisa M.Sullivan, Kimberly A.Dukes, Elena Losina. “Tutorial in biostatistics: an introduction to hierarchical linear modeling”.

[33] Lange K[1], Sinsheimer JS (1992). Calculation of genetic identity coefficients

[34] John Blangero et Al. Solar manual.

http://www.biostat.wustl.edu/genetics/geneticssoft/manuals/solar210/00.contents.html

[35] L. Almasy and John Blangero 1998 .“Multipoint quantitative-trait linkage analysis in general pedigrees”.

[36] T. Mark Beasley and Stephen Erickson (2009).

“Rank-Based Inverse Normal Transformations are Increasingly Used, But are They Merited?”. *Behaviour Genetics.*

[37] Wikipedia, June 2016. https://en.wikipedia.org/wiki/Heritability

[38] Goeman, Jelle J.; Solari, Aldo (2014). "Multiple Hypothesis Testing in Genomics". *Statistics in Medicine.*

[39] Hugues Aschard, Bjarni J. Vilhjálmsson (2014). “Maximizing the Power of Principal – Component Analysis of Correlated Phenotypes”. *The American Journal of Human Genetics.*

[40] Jian Yang, Michael N Weedon, Shaun Purcell (2011). “Genomic Inflation Factor under polygenic inheritance”. *European Journal of Genetics.*

[41] GitHub, June 2016.GenABEL package. https://github.com/cran/GenABEL

[42] Wikipedia, June 2016. https://en.wikipedia.org/wiki/Q%E2%80%93Q_plot

[43] Omim, June 2016.

http://www.omim.org/entry/605885?search=SEMA6A&highlight=sema6a

[44] Omim, June 2016.

http://www.omim.org/entry/606382?search=MAGI2.&highlight=magi2

[45] GeneCards, June 2016.

http://www.genecards.org/cgi-bin/carddisp.pl?gene=LOC105377195&keywords=LOC105377195

[46] Omim, June 2016

http://www.omim.org/entry/611257?search=TMEM132D&highlight=tmem132d

# 10. Appendix

# A.1. Materials

## A.1.1. File encoding of the genotypes

### Map files

On the one hand, we have map files, which contain all information about SNPs. In these text files, the number of rows is equal to the amount of SNPs we have, and the number of columns is always equal to 4. These four fields give us information about which chromosome the SNP belongs to, the SNP identifier, the genetic distance (in morgans) and the base-pair position (bp units). Although the meaning of last two fields may seem a bit complicated, we are only interested in the last one, which just indicates the physical position of the SNP. Conversely, genetic distance is a quite more complicated concept, which accounts for genetic linkage and the probability of recombination for a certain loci. Even though, this field is out of the scope of this study, since it is widely used in linkage studies but rarely used in association studies. Finally, mention that fields are separated from each other using the tab space '\t'. This detail will be useful if we want to either import these files or make our own scripts to extract information.

Here there is an example of a map file:

```
1    exm-IND1-200449980     204.38440   202183358
1    exm-IND1-85310248          109.70290   85537661
10   exm-IND10-102817747    121.18120   102827757
10   exm-IND10-18329639     42.83914    18289633
10   exm-IND10-27476467     51.52821    27436462
10   exm-IND10-27727540     51.78007    27687534
```

### Ped files

On the other hand, we have ped files which contain the genotypes of the SNPs present in the map file. In this case, the number of rows is equal to the number of individuals, and the number

of columns is the sum of 6 mandatory columns and two times the number of SNPs, because all markers are biallelic (i.e. there are two copies for each genotype, one inherited from mother and the other one inherited from father). They can be any character, but in our case the genotypes are filled using A,C,G,T and I,D for insertions/deletions. The first five out of six mandatory columns bring information about the individual, and if wanted, the 6th column can bring information of a trait or phenotype. So the first six fields are the following ones:

**Family ID**

**Individual ID**

**Paternal ID (=0 if he is a founder) Maternal ID (=0 if she is a founder) Sex (1=male, 2=female, other=unknown)**

**Phenotype**

However, before ending this section with an example we should stop for a while to define the meaning of the characters used to fill the genotypes. The A,C,G,T are the four nucleotide bases of a DNA strand: adenine, cytosine, guanine and thymine as we already explained when we defined singular nucleotide polymorphisms. Finally, a very little portion of the global ped file of the GAO project looks as follows (notice that the rows have been chosen randomly):

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 7 | 10210 | 0 | 0 | 1 | 0 | A | A |
| 101 | 11406 | 0 | 0 | 1 | 0 | A | G |
| 159 | 12302 | 0 | 0 | 1 | 0 | A | G |
| 206 | 13318 | 0 | 0 | 2 | 0 | G | G |
| 257 | 15320 | 0 | 0 | 2 | 0 | G | G |
| 312 | 18309 | 0 | 0 | 2 | 0 | A | G |
| 318 | 18316 | 0 | 0 | 2 | 0 | A | G |
| 321 | 18403 | 0 | 0 | 1 | 0 | G | G |
| 345 | 19208 | 0 | 0 | 1 | 0 | G | G |
| 359 | 19315 | 0 | 0 | 1 | 0 | A | A |

As we can see, very important information of the pedigree is missing. To PLINK these individuals are unrelated: a priori we know nothing about the family, father and mother of each individual. Nevertheless we are not going to carry out the association using plink, we will need this information to perform the Quality Control. In section 4 is described how we obtained the appropriate *.ped file to perform a successful QC.

## A.1.2. Biology's information of the bone metabolism markers

**25-hydroxy vitamin D:** the most stable and plentiful metabolite of vitamin D in human serum;

**Sclerostin:** a glycoprotein secreted almost exclusively by osteocytes that decreases osteoblastogenesis and bone formation

**Osteocalcin:** important bone turnover markers for formation

**Serum beta-crosslaps:** important bone turnover marker for resorption,

**Osteoprotegerin:** a key inhibitor of bone resorption;

**Bone-specific alkaline phosphatase (OstaseBAP**): an enzyme that plays an essential role in the regulation of tissue mineralization

**Leptin and insulin-like growth factor 1 (IGF1**): two markers with opposite effects on the stimulation of skeletal muscle, cartilage and bone growth, among other tissues;

**Adiponectin**: a protein that potentially influences directly bone cell function;

**Osteopontin:** a bone turnover marker that plays a key role in anchoring osteoclasts to the mineral matrix of bones

**Parathyroid hormone**: a polypeptide that increases the concentration of Ca2+ in the blood; and (xii)

**Tumor necrosis factor alpha (TNFalpha)**: a protein capable of regulating bone turnover, formation and resorption.

## A.1.3. Scripts to obtain the table of clinical phenotypes

In this section there is the code used to extract the table of densitometric traits from a SQL database.

### R script

```
library(RMySQL)
con <- dbConnect(MySQL(), group = 'yamenaGao')
phenotype <- dbReadTable(con, 'phenotype')
f <- as.factor(phenotype$ph_trait)
noms_fenotips <- levels(f)
n <- 0
for(fenotipo in noms_fenotips){
  tabla_fenotipo <- read.table(file= paste(fenotipo, '.csv', sep=''),header=
TRUE, sep= ' ')
  if(n==0){
    attach(tabla_fenotipo)
    daf <- data.frame(ID)
    daf <- cbind(daf, tabla_fenotipo[[fenotipo]])
    detach(tabla_fenotipo)
  }
  else{
    attach(tabla_fenotipo)
    daf <-cbind(daf, tabla_fenotipo[[fenotipo]])
  }
  n <- n+1
}
names(daf)[2:length(daf)] <- noms_fenotips
write.csv(daf, file='tabla_final_fenos.csv')
```

### Python script

```
def fenotipos():
    f= open('fenotipos_completo.csv','r')
    f.readline()
    lista_diccionarios=[]
```

```
    d={}
    n=0
    for linea in f:
        linea=linea.strip()
        lista_linea=linea.split(',')
        if lista_linea[1] not in d.values():
            if n>0:
                lista_diccionarios= lista_diccionarios+[d]
            d={}
            d['ID']=lista_linea[1]
        d[lista_linea[2]]=lista_linea[3]
        n=n+1
    f.close()
    return lista_diccionarios

def lista_todos_fenotipos(lista_diccionarios):
    lista_fenos=[]
    for individuo in lista_diccionarios:
        for clave in individuo:
            if clave not in lista_fenos:
                lista_fenos.append(clave)
    return lista_fenos

def fenos_csv(lista_diccionarios):
    lista_claves=lista_todos_fenotipos(lista_diccionarios)
    lista_claves_reducidas=lista_claves[1:4]
    for fenotipo in lista_claves:
        f=open(fenotipo.replace('"','')+'.csv','w')
        n=0
        print fenotipo
        for individuo in lista_diccionarios:
            print n
```

# A.2. Data pre-processing

## A.2.1. Data set-up for the Quality Control

### Modification of *.ped file

In this annex we give a more detailed explanation of how the python script used in the data set-up works.

First of all, the script extracts the IDs from the initial .ped file and save them in a list. After that, the script looks into one of the tables of phenotypes and extracts the first six columns of those individuals whose IDs are in the list obtained before. This information is saved in a list of lists, where each sublist corresponds to one individual. However, before the sublist is appended to the global one, the sex field for each individual is recoded. The reason for this change is that PLINK identifies males as 1 and females as 2; conversely, in the table of phenotypes the gender is coded as 'M' for males and 'F' for females. Likewise, the scrpit permutes the order of first two columns: ID and FAM, to put it as PLINK's required structure: FAM and ID.

Finally, the script merges this accurate data of individuals with the genotypes in the initial *.ped file.

Below, the 'raw' data extracted from the table of phenotypes. Notice that now we have information about the father and mother of each individual.

['10202', 'gao10', '10101', '10102', 'F']

['10205', 'gao10', '0', '0', 'M']
['10206', 'gao10', '10101', '10102', 'F']

['10207', 'gao10', '10101', '10102', 'M']
['10208', 'gao10', '0', '0', 'F']

['10209', 'gao10', '0', '0', 'F']
...

And the data ready to be merged with the genotypes:

['gao10', '10202', '10101', '10102', '2']

['gao10', '10205', '0', '0', '1']
['gao10', '10206', '10101', '10102', '2']

['gao10', '10207', '10101', '10102', '1']
['gao10', '10208', '0', '0', '2']

['gao10', '10209', '0', '0', '2']
...

Finally, the aspect of the new *.ped file:

| FAM | ID | FA | MO | SEX | PHENOTYPE | Marker 1 | Marker 2 |
|---|---|---|---|---|---|---|---|
| gao10 | 10202 | 10101 | 10102 | 2 | 0 | A A | G G |
| gao10 | 10205 | 0 | 0 | 1 | 0 | A G | G G |
| gao10 | 10206 | 10101 | 10102 | 2 | 0 | A A | G G |
| gao10 | 10207 | 10101 | 10102 | 1 | 0 | A G | G G |
| gao10 | 10208 | 0 | 0 | 2 | 0 | G G | A G |
| gao10 | 10209 | 0 | 0 | 2 | 0 | G G | G G |

The code in python to carry out this procedure is the following one:

```
def datos_ind():
    g=open('/home/gabriel/tabla_fenos/gao.proteins.csv')
    f=open('/home/gabriel/plink/Todas.ped','r')
    IDs=[]
# Extracting the IDs from the ped file
    for linia in f:
        linia=linia.strip()
        lista=linia.split('\t')
        IDs.append(lista[1])
# Due to the fact that the tables of phenotypes bring more accurate
information about the pedigreee, we have extracted the first six columns
# of the table of proteins for each individual whose ID was in the ped file.
Afterwards, we organised each indvidual's information in the ap
# propiate PLINK order, which has been already explained in section 3.3.
# Although this is not the only change we have done below. The gender
information in the table of phenotypes is coded differently from PLINK
```

```
# standards, therefore we have recoded this field before appending
individuals into the global list.
    info_ped=[]
    i=0
    for linia in g:
        linia=linia.strip()
        lista=linia.split(',')
        ind=[lista[1]]+[lista[0]]+lista[2:5]
        if i>=1 and ind[1] in IDs:
            if ind[4]=='M':
                ind[4]='1'
            elif ind[4]=='F':
                ind[4]='2'
            print ind
            info_ped.append(ind)
        i=i+1
    g.close()
    f.close()
# Finally, we rewrite the ped file joining the data contained in the list of
lists with the genotyped data present in the ped file.
    f=open('/home/gabriel/plink/Todas.ped','r')
    t=open('/home/gabriel/prueba/Todas2.ped','w')
    for linia in f:
        linia=linia.strip()
        lista=linia.split('\t')
        #v='F'
        for i in range(len(info_ped)):
            if info_ped[i][1]==lista[1]:
                #v='T'

t.write('\t'.join(info_ped[i])+'\t'+'0'+'\t'+'\t'.join(lista[6:])+'\n')
        if v=='F':
            t.write('\t'.join(lista)+'\n')
    f.close()
    t.close()
```

### Bed files

```
bash_bed <- 'plink --file /home/gabriel/plink/Todas --make-bed --out Todas2'
system(bash_bed)
```

## A.2.2. Quality Control

## Minor Allelle Frequency

In this section of the annex we show the aspect of the files obtained in PLINK through the command –freq as well as the code used to carry out the analsysis.

First, the *.frq file used to filter by MAF looks like this:

| CHR | SNP | A1 | A2 | MAF | NCHROBS |
|---|---|---|---|---|---|
| 0 | exm-rs10862691 | 0 | G | 0.000000 | 768 |
| 0 | exm-rs11136341 | G | A | 0.454300 | 744 |
| 0 | exm-rs1799853 | A | G | 0.151000 | 768 |
| 0 | exm-rs2015062 | A | G | 0.362000 | 768 |
| 0 | exm-rs7164335 | A | G | 0.342500 | 762 |
| 0 | exm-rs9380254 | G | C | 0.033940 | 766 |
| 0 | exm2216283 | G | A | 0.067890 | 766 |
| 0 | exm2216291 | 0 | G | 0.000000 | 768 |
| 0 | exm2216292 | G | A | 0.005208 | 768 |
| 0 | exm2226201 | A | C | 0.001302 | 768 |

Once we have the *.frq file, we enter in R and we import the data. Afterwards, we apply a vectorized operation and we export the resulting SNPs to a text file. In the end, we use PLINK again to extract those SNPs in the text file from the pedigree data. The code in R and the command lines in PLINK can be found below:

```
#First we generate a text file with all snps' frequency information
bash_frq <- 'plink --bfile /home/gabriel/plink/Todas --freq --out
/home/gabriel/plink/Todas_freq'
system(bash_frq)

#Vectorized operation
freq <- read.table('Todas_freq.frq', header=T)
lowMAFsnps <- as.vector(freq$SNP[freq$MAF < 0.01])
write.table(lowMAFsnps, '/home/gabriel/plink/lowMAFsnps.txt', quote=F,
col.names=F, row.names=F)
```

```
#Extract those snps having a MAF below 1% (thus are also excluded monoallelic
snps)
bash_filter <-'plink --bfile /home/gabriel/plink/Todas --exclude
/home/gabriel/plink/lowMAFsnps.txt --make-bed --out
/home/gabriel/plink/filt_maf'
system(bash_filter)
```

## Mendelian errors

In this section we present the structure of the files obtained in PLINK through the command –mendel as well as the code used to carry out the analysis.

The aspect of the 4 files generated in PLINK through the –mendel command is presented below:

***The *.mendel file:***

FID Family ID

KID Child individual ID

CHR Chromosome

SNP SNP ID

CODE A numerical code indicating the type of error

ERROR Description of the actual error

***The *.lmendel file*:**

CHR Chromosome

SNP SNP ID

N Number of Mendel errors for this SNP

***The *.imendel file:***

FID Family ID

IID Individual ID

N Number of errors this individual was implicated in

***The *.fmendel file:***

FID Family ID

PAT Paternal individual ID

MAT Maternal individual ID

CHLD Number of offspring in this (nuclear) family

N Number of Mendel errors for this (nuclear) family

### Conversion to the appropriate format (.*.csv)

The python script written to perform this conversion is the following one:

```
def convert_to_csv():
    mendel=open('Todas2.mendel','r')
    lmendel=open('Todas2.lmendel','r')
    imendel=open('Todas2.imendel','r')
    fmendel=open('Todas2.fmendel','r')
    n_mendel=open('mendel.csv','w')
    n_lmendel=open('lmendel.csv','w')
    n_imendel=open('imendel.csv','w')
    n_fmendel=open('fmendel.csv','w')
    m=0
    for linia in mendel:
        linia=linia.strip()
        lista=linia.split(' ')
        lista=[x for x in lista if x!='']
        if m==0:
            n_mendel.write(','.join(lista[:-1])+'\n')
        elif m >0:
            n_mendel.write(','.join(lista[:-5])+'\n')
        m+=1
    l=0
    for linia in lmendel:
        linia=linia.strip()
        lista=linia.split(' ')
        lista=[x for x in lista if x!='']
        if l==0:
            n_lmendel.write(','.join(lista)+'\n')
        elif l >0:
            n_lmendel.write(','.join(lista)+'\n')
        l+=1
    i=0
    for linia in imendel:
        linia=linia.strip()
        lista=linia.split(' ')
```

```
        lista=[x for x in lista if x!='']
        if i==0:
            n_imendel.write(','.join(lista)+'\n')
        elif i >0:
            n_imendel.write(','.join(lista)+'\n')
        i+=1
    f=0
    for linia in fmendel:
        linia=linia.strip()
        lista=linia.split(' ')
        lista=[x for x in lista if x!='']
        if f==0:
            n_fmendel.write(','.join(lista)+'\n')
        elif f >0:
            n_fmendel.write(','.join(lista)+'\n')
        f+=1
```

### Analysis of Mendelian errors per SNP

The code in R to carry out the analysis of Mendelian errors per SNP:

```
df_lmendel <- read.csv('/home/gabriel/prueba/mendel/lmendel.csv')
SNPs_pruned_mendel <- df_lmendel$SNP[df_lmendel$N  > 2]
length(SNPs_pruned)
write.table(SNPs_pruned_mendel, '/home/gabriel/plink/SNPs_pruned_mendel.txt',
quote=F, col.names=F, row.names=F)
bash_filter <-'plink --bfile /home/gabriel/plink/filt_maf --exclude
/home/gabriel/plink/SNPs_pruned_mendel.txt --make-bed --out
/home/gabriel/plink/filt_maf_mendel'
system(bash_filter)
```

The code in PLINK to replicate the analysis done in R:

```
bash_mendel <- 'plink --bfile /home/gabriel/plink/filt_maf --me 1 0.01 --
make-bed --out /home/gabriel/plink/filt_maf_mendel'
system(bash_mendel)
```

### Analysis of Mendelian errors per individual

The script in R used to obtain the multichart plot of Mendelian errors distribution per individual is the following one:

```
library(lattice)
df_familia <- read.csv('/home/gabriel/prueba/mendel/fmendel.csv')
df_individuos <- read.csv('/home/gabriel/prueba/mendel/imendel.csv')
df_info_ped <- read.table('/home/gabriel/plink/info_ped.csv', header=T)
analysis_mendel <- function(df_fam, df_ind, df_ped){
  errors <- c()
  for(i in 1:dim(df_fam)[1]){
    num_child <- df_fam[i,4]
    ids_offspring <- df_ped$ID[df_ped$FA==df_fam[i,2] &
df_ped$MO==df_fam[i,3]]
    errors <- append(errors, df_ind$N[df_ind$IID %in%
ids_offspring][1:num_child])
  }
  err_sorted <- sort(errors)
  par(mfrow=c(2,2))

  box <- boxplot(err_sorted, ylab='Mendel errors', col='pink', main='Boxplot
of mendel errors')

  h1 <- hist(errors, xlab='Mendel errors' ,col='gray', main='Frequency
histogram of Mendel errors')


  err_no_outliers <- errors[-which(errors %in% box$out)]
  cuts <- quantile(err_no_outliers, seq(0,1,0.1))
  h2 <- hist(err_no_outliers,xlab='Mendel errors', breaks=cuts, col='grey',
main='Density histogram without outliers')
  lines(density(err_no_outliers), col="blue", lwd=2)
  lines(density(err_no_outliers, adjust=2), lty="dotted", col="darkgreen",
lwd=2)
  h3 <- hist(err_no_outliers,xlab='Mendel errors',
breaks=round(sqrt(length(err_no_outliers))), col='grey', main='Histogram
without outliers and a normal curve')
  xfit<-seq(min(err_no_outliers),max(err_no_outliers),length=40)
  yfit<-dnorm(xfit,mean=mean(err_no_outliers),sd=sd(err_no_outliers))
  yfit <- yfit*diff(h3$mids[1:2])*length(err_no_outliers)
  lines(xfit, yfit, col="blue", lwd=2)
  return(list(tots_errors=err_sorted, no_outliers = err_no_outliers))
}
```

And the function in python to perform the outliers' analysis of Mendelian errors per individual is the following one:

```
def mendel(ID, tipo_errores):
    f=open('/home/gabriel/prueba/mendel/Todas2.mendel','r')
    l_errores=[]
    n=0
    for linia in f:
        linia=linia.strip()
        lista=linia.split(' ')
        lista=[x for x in lista if x !='']
        l_errores.append(lista)
    j=0
    for error in l_errores:
        if error[1]==ID:
            if error[4] in  tipo_errores:
                n=n+1
        j+=1
    return n
```

## Missingness

The fields contained in the file *.lmiss are the following ones:

**SNP:** SNP identifier

**CHR:** Chromosome number

**N_MISS**: Number of individuals missing this SNP

**N_GENO:** Number of non-obligatory missing genotypes

**F_MISS:** Proportion of sample missing for this SNP

And the code used to apply the filter to the data in R:

```
df_lmiss <- read.csv('/home/gabriel/prueba/lmiss.csv')
SNPs_removed_miss <- df_lmiss$SNP[df_lmiss$N_MISS > 7]
length(SNPs_removed)
```

```
write.table(SNPs_removed_miss, '/home/gabriel/plink/SNPs_removed_miss.txt',
quote=F, col.names=F, row.names=F)
bash_filter <-'plink --bfile /home/gabriel/plink/filt_maf_mendel --exclude
/home/gabriel/plink/SNPs_removed_miss.txt --make-bed --out
/home/gabriel/plink/filt_maf_mendel_miss'
system(bash_filter)
```

### Hard-Weinberg equilibrium

Commands in plink used to apply the HWE filter

```
bash_last_filter <- 'plink --bfile /home/gabriel/plink/filt_maf_mendel_miss -
-nonfounders  --hwe 1e-3 --make-bed --out /home/gabriel/plink/data_ready'
system(bash_last_filter)
```

## A.2.3. Data set-up for association

In this section of the annex we attach the code used in the operations performed to let the data ready for the association analysis, which are clustering by chromosome and transforming the genotypes into numeric data (additive models).

### Code used to cluster by chromosome

```
bash_split <- 'for chr in $(seq 1 22);
do
plink --bfile /home/gabriel/plink/data_ready --chr $chr --make-bed --out
/home/gabriel/plink/datachr$chr;
plink --bfile /home/gabriel/plink/datachr$chr --recode --out
/home/gabriel/plink/datachr$chr;
done'
system(bash_split)
```

### Code used to transform the genotypes into numeric data (additive models)

```
bash_additive <-  'for chr in $(seq 1 3);
do
```

```
plink --bfile /home/gabriel/plink/datachr$chr --recodeA --out
/home/gabriel/plink/datachr$chr;
done'
system(bash_additive)
```

# A.3. Analysis

## A.3.1. The kinship matrix

In this section the reader can have a look at the code used to obtain the global kinship matrix, as well as the histogram of kinship coefficients and the procedure followed to obtain the Kinship matrix sorted of the 11th family.

```
library(solarius)
library(kinship2)

df_fenos <- read.table('/home/gabriel/tabla_fenos/gao.proteins.csv', header=T
RUE, sep=',')

#Estimation of the kinship matrix
kinship_i <- solarKinship2(df_fenos)

#We eliminate those individuals whose IDs are not in the table of phenotypes.
Notice that we have done it deleting symmetrically the rows and columns of th
e unmatched individuals
ids <- df_fenos[,'ID']
ids_kinship <- rownames(kinship_i)
indices <- which(!(ids_kinship %in% ids))
kinship <- kinship_i[-indices,-indices]

#Plotting the global kinship matrix
plotKinship2(kinship, y='image')


#Plotting an histogram which shows the frequencies for the different degrees
of pairwise relation
plotKinship2(kinship, y='hist')
```

```
#Extracting the subset of indices for the 11th family
familia11 <- c()
for(i in 1:576){
  if(substr(ids[i],start=0,stop=2)=='20'){
    familia11 <- append(familia11, ids[i])
  }
}

ind_fam11 <- which(ids_kinship %in% familia11)

kinship_fam11 <- kinship_i[ind_fam11,ind_fam11]

#Sorting the kinship matrix by ID, which implies that the older inviduals wil
l be the top rows and the younger ones will be the bottom rows
kinship_sorted <- kinship_fam11[order(attr(kinship_fam11, 'dimnames')[[1]]),
order(attr(kinship_fam11, 'dimnames')[[2]])]


#Plotting the sorted kinship matrix of the 11th family

plotKinship2(kinship_sorted, y='image')
```

## A.3.2. Traits' transformations

In this section we have attached our own implementation of the inverse normal transformation function and a full script that transforms a bunch of chosen traits from a table of phenotypes and builds a new table with the transformed traits. Although in polygenic models this transformation can be done automatically while fitting the model, in the association model the traits given have to be already transformed.

**The implementation of the inverse normal transformation:**

```
#When there is a tie, by default it computes the average. In SOLAR, the inorm
al transformation uses c=0 (despite the most common value is 3/8) and 'averag
e' for ties.
```

```
inormal <- function(trait, c, tie='average'){
  rk <- rank(trait, na.last='keep', ties.method = tie)
  N=length(which(!is.na(trait)))
  trait_in <- qnorm((rk-c)/(N-2*c+1))
  return(trait_in)
}
```

**The script that transforms tables of phenotypes:**

```
transform.phenotypes <- function(df, traits, type='inormal', write=FALSE,
path='/home/gabriel/Documentos/'){
  library(solarius)
  new_df <- df
  nom_new <- c()
  for(trait in traits){
    transformed_trait <- transformTrait(df[,trait], type, mult= 1)
    nom_new <- append(nom_new, paste(trait, '_',substr(type,1,2), sep=''))
    new_df <- cbind(new_df, transformed_trait)
  }
  names(new_df)[length(names(new_df))-length(nom_new):length(names(new_df))]
<- nom_new
  if(write==TRUE){
    wirte.csv(new_df, file(paste(path,'TABLA_converted_', type,
'.csv',sep='')))
  }
  return(new_df)
}
```

## A.3.3. Polygenic models

The two main scripts that have been used in order to fit polygenic models for the two main types of traits are attached in this section.

### Models for bone metabolism markers

```
lista_cf_mod1 <- list()
lista_vcf_mod1 <- list()
lista_cf_mod2 <- list()
lista_vcf_mod2 <- list()
list_signif_mod1 <- list()
list_signif_mod2 <- list()
for(element in fenos){
  mod1 <- solarPolygenic(traits= element, covlist = append(covs, 'Age^2'),
```

```
transforms='inormal', covtest=T, data= proteins)
  lista_cf_mod1 <- append(lista_cf_mod1, list(mod1$cf))
  lista_vcf_mod1 <- append(lista_vcf_mod1, list(mod1$vcf))
  significant <- as.character(mod1$cf$covariate[!is.na(mod1$cf$pval) &
mod1$cf$pval <= 0.05 ])
  list_signif_mod1 <- append(list_signif_mod1, list(significant))
  if('Age' %in% significant){
    if('Age^2' %in% significant){
       mod2 <- solarPolygenic(traits = element, covlist = significant,
transforms = 'inormal', covtest= T, data = proteins)
    }
    else{
       mod2 <- solarPolygenic(traits = element, covlist = append(significant,
'Age^2'), transforms = 'inormal', covtest= T, data = proteins)
    }
  }
  else{
    if('Age^2' %in% significant){
      ind <- which('Age^2'==significant)
      significant <- significant[-ind]
    }
    mod2 <- solarPolygenic(traits = element, covlist = significant,
transforms = 'inormal', covtest= T, data = proteins)
  }
  signif2 <- as.character(mod2$cf$covariate[!is.na(mod2$cf$pval) &
mod2$cf$pval <= 0.05 ])
  list_signif_mod2 <- append(list_signif_mod2, list(signif2))
  lista_cf_mod2 <- append(lista_cf_mod2, list(mod2$cf))
  lista_vcf_mod2 <- append(lista_vcf_mod2, list(mod2$vcf))
}
```

## Models for clinical phenotypes

```
polygenic.models <- function(phenotypes, phenodata, covariates){
  library(solarius)
  no.missings <- apply(phenodata, 2, function(x){length(na.omit(x))})
  df <- data.frame(t(c(0,0,0,0,0)))
  names(df) <- c('Trait', 'Covariates', 'N indiv', 'Heritability', 'P-value')
  for(phen in phenotypes){
    mod1 <- solarPolygenic(traits=phen, covlist=covariates,
transforms='inormal', covtest=T, data=phenodata)

    cov.signif <-  as.character(mod1$cf$covariate[mod1$cf$pval < 0.05 &
!is.na(mod1$cf$pval)])
```

```
    if('Age^2' %in% cov.signif & !'Age' %in% cov.signif){
      cov.signif <- cov.signif[-which(cov.signif=='Age^2')]
    }

    mod2 <- solarPolygenic(traits=phen, covlist=cov.signif,
transforms='inormal', covtest=T, data=phenodata)
    if(!is.null(cov.signif)){
      cov.def <- as.character(mod2$cf$covariate[mod2$cf$pval < 0.05 &
!is.na(mod2$cf$pval)])
    }


    if('Age^2' %in% cov.def & !'Age' %in% cov.def){
      cov.def[-which(cov.def=='Age^2')]
    }

    names <- append(phen, cov.def)
    print(names)
    num.indiv <- min(na.omit(no.missings[names]))
    heritability <- paste(substr(toString(mod2$vcf[1,'Var']),1,5), '±',
substr(toString(mod2$vcf[1,'SE']),1,6), sep=' ')
    pvalue <- format(mod2$vcf[1,'pval'], scientific=TRUE, digits=4)
    covs <- toString(cov.def)

    row <- c(phen, covs, num.indiv,  heritability, pvalue)
    df <- rbind(df, row)
  }
  return(list(dat=df[-1,], covar=cov.def, N=num.indiv, h2r=heritability,
pval= pvalue))
}
```

## A.3.4. Association models

The function present in the code below was the main one to carry out the different associations in this study (nevertheless exactly this script was used to associate with the bone metabolism markers):

```
library(solarius)

phenos <- read.csv('/home/gabriel/filtered data/tablas
fenos/gao.proteins.inormal.csv')
```

```
nombres <- names(phenos)[34:45]

file_path <- '/home/gabriel/filtered data/tablas fenos/'

load('/home/gabriel/filtered data/tablas fenos/covariates.RData')


association <- function(phenotypes, phenodata, path, list.covariates,
CORES=64){

  n <- 1
  for(phen in phenotypes){

    output.file=paste(phen, '.csv',sep='')

    covariates <- list.covariates[[n]]

    for(i in 1:22){

      # Genotyped data
      num.chr <- toString(i)
      chr.raw <- paste(num.chr,'.raw',sep='')
      chr.map <- paste(num.chr,'.map', sep='')
      plink.raw <- file.path('/home/gabriel/filtered data/plink_asociacion',
chr.raw)
      plink.map <- file.path('/home/gabriel/filtered data/plink_asociacion',
chr.map)

      # Association model
      mod <- solarAssoc(traits=phen, covlist = covariates, covtest=T, data=
phenodata, plink.raw= plink.raw, plink.map= plink.map, plink.raw.append= F,
cores = CORES)

      # Exporting data

      if(i==1){
        write.table(mod$snpf, file= paste(path, output.file, sep=''), append=
F, col.names= T, row.names= F)
      }
      else{
        write.table(mod$snpf, file= paste(path, output.file, sep=''), append=
T, col.names = F, row.names = F)
```

```
      }
    }
    n <- n+1
  }
}

association(nombres, phenos, file_path, list.covariates=covariates, CORES=64)
```

## A.3.5. PCAs

In this section we attach all the code that we have used in the principal components analysis. In first place there is the code of the PCAs correction by the kinship matrix, and afterwards the code of the PCAs comparison.

### Script for PCA correction

```
library(MASS)
relprcomp <- function(X, center = TRUE, scale = FALSE, ncomp,
  relmat)
{
  # testing code
  # 1)
  # X <- iris[1:10, -5]
  # C <- diag(seq(0.5, 1, length = 10), 10)
  # 2)
  # X <- iris[, -5]
  # C <- diag(seq(0.5, 1, length = nrow(X)), nrow(X))

  ### arguments
  if(missing(relmat)) { relmat <- diag(1, nrow(X)) }
  if(missing(ncomp)) { ncomp <- ncol(X) }

  ### data
  X <- as.matrix(X)
  C <- relmat

  ### pre-processing
  nobs <- nrow(X)
```

```
  Xmean <- apply(X, 2, mean)
  Xscale <- apply(X, 2, sd)

  if(center) { X <- sweep(X, 2, Xmean, "-") }
  if(scale) { X <- sweep(X, 2, Xscale, "/") }

  ### prepare matrix `R`
  Cinv <- solve(C)
  #Cinv <- ginv(C)
  print(dim(Cinv))
  R <- (1 / (nobs - 1)) * t(X) %*% Cinv %*% X
  print(R)

  ### SVD
  # A = V Lmbd V^(-1), where Lmbd = diag(vectors)
  out <- eigen(R)
  d <- out$values
  V <- out$vectors

  S <- X %*% V

  ### form output
  out <- list(nobs = nobs, center = Xmean, scale = Xscale,
    relmat = relmat,
    sdev = sqrt(d), x = S, rotation = V)

  oldClass(out) <- "relprcomp"
  return(out)
}

#---------------------
### Class functions
#---------------------

print.relprcomp <- function(x, ...)
{
  cat("Standard deviations:\n")
  print(x$sdev)
  cat("\n")

  cat("Rotation:\n")
  print(x$rotation)
}
```

```
scores.relprcomp <- function(object, ...) object$x

loadings.relprcomp <- function(object, ...)
{
  X <- object$x %*% t(object$rotation)
  Cinv <- solve(object$relmat)
  K <- (1 / (object$nobs - 1)) * t(X) %*% Cinv %*% object$x

  R <- (1 / (object$nobs - 1)) * t(X) %*% Cinv %*% X
  r <- diag(R)
  d <- object$sdev
  RD <- sapply(1:4, function(i) r[i] * d)

  L <- K / sqrt(RD)

  return(L)
}

scoreplot.relprcomp <- function(object, comps = 1:2,
  labels,
  xlab, ylab, type = "p", ...)
{
  S <- scores(object)[, comps, drop = FALSE]

  varlab <- compnames(object, comps, explvar = TRUE)
  if(missing(xlab)) { xlab <- varlab[1] }
  if(missing(ylab)) { ylab <- varlab[2] }

  ### plot
  plot(S, xlab = xlab, ylab = ylab, type = type, ...)

  # labels
  if(!missing(labels)) text(S[, 1], S[, 2], labels, ...)
}

#---------------------
### Support functions
#---------------------

explvar <- function(object)
{
  switch(class(object)[1],
    relprcomp = object$sdev^2 / sum(object$sdev^2),
    stop("error in explvar"))
}
```

```
compnames <- function(object, comps, explvar = TRUE)
{
  compnames <- paste("PC", comps, sep = "")

  if(explvar) {
    vars <- 100 * explvar(object)[comps]

    compnames <- paste(compnames, " (", round(vars, 2), "%)", sep = "")
  }

  return(compnames)
}
```

## PCAs comparison

```
library(solarius)
library(pls)
source('/home/gabriel/filtered data/scripts/multivar.lib.R')
df.proteins <- read.csv('/home/gabriel/filtered data/gao.proteins.csv')
df.clinical <- read.csv('/home/gabriel/filtered data/fenos_limpios_id.csv')
load('/home/gabriel/filtered data/RData/K_gao.RData')
load('/home/gabriel/filtered data/GAIT2-dat-platelets/K.RData')
df.GAIT <- read.csv('/home/gabriel/filtered data/GAIT2-dat-
platelets/14.04.09.MMTPEXTOK.csv')

PCA_comparativa <- function(fenos_df, K){

  #Removing missings
  indices_NA <- as.numeric(attributes(na.omit(fenos_df))$na.action)
  fenos_df <- fenos_df[-indices_NA,]

  #Preparing matrix X and Kinship
  X <- as.matrix(fenos_df[,-1])
  ind <- rownames(K) %in% fenos_df[,1]
  K <- K[ind,ind]

  #PCAs
  pca_corregido <- relprcomp(X, center=TRUE, scale=TRUE,relmat=K);
  pca_normal <- prcomp(~.,data = as.data.frame(X),center=TRUE, scale=TRUE);
  FAM <- as.factor(substr(fenos_df[,1], 1, 2))

  #Plotting
  dev.new()
  scoreplot(pca_normal, main='Non-corrected PCA', col=FAM)
```

```
  dev.new()
  scoreplot.relprcomp(pca_corregido, main='Corrected PCA',col=FAM)

  #Out
  out <- list(pca_corregido = pca_corregido, pca_normal=pca_normal, FAM= FAM,
K=K)
  return(out)
}
```

## A.3.6. Exploratory tools

In this section of the annex the reader can have a look at two scripts used to analyse the results from the association studies. The first script gives a general idea of the results, computing the Manhattan plots, the Q-Q plots and calculating the genomic inflation factor. In addition, the second script digs a bit deeply on the results and for a given chromosome it returns a first approach of the mapped SNPs.

### Inflation factor and figures

```
results.plot <- function(names, path, plot=FALSE){
  library(qqman)
  library(GenABEL)
  inflation.factor <- c()
  for(element in names){
    df <- read.table(paste(path,element,'.csv',sep=''), header=TRUE, sep=' ')

    #Computing inflation factor for each trait

    inflation.factor <- append(inflation.factor, estlambda(data=df$pSNP,
method='median')$estimate)

    if(plot==TRUE){

      #Calculating Bonferroni common threshold
```

```
      log_line <- -log((0.05/length(df$SNP)), base=10)

      #Plotting Manhattan and QQ plot

      png(filename=paste(path,'manhattan_',element,'.png',sep=''), width=
480, height= 480, bg = 'white', pointsize = 12)

      manhattan(df, chr='chr', bp='pos',snp = 'SNP',p = 'pSNP', col =
c('darkblue','darkgreen'), main=paste('Manhattan plot of ', element, sep=''))
      abline(log_line, 0, col='yellow')
      legend('topleft',bty='n',legend= c('GWAS line','Suggestive line'),
col=c('red','blue'), lty= 1)

      dev.off()

      png(filename=paste(path,'qq_',element,'.png',sep=''), width= 480,
height= 480, bg = 'white', pointsize = 12)

      qq(pvector=df$pSNP, main=paste('QQ plot of ', element, sep=''))

      dev.off()

    }
  }
  #Building a data frame with inflation factors

  df_inflation <- data.frame(names, inflation.factor)
  names(df_inflation) <- c('Names', 'Inflation factor')
```

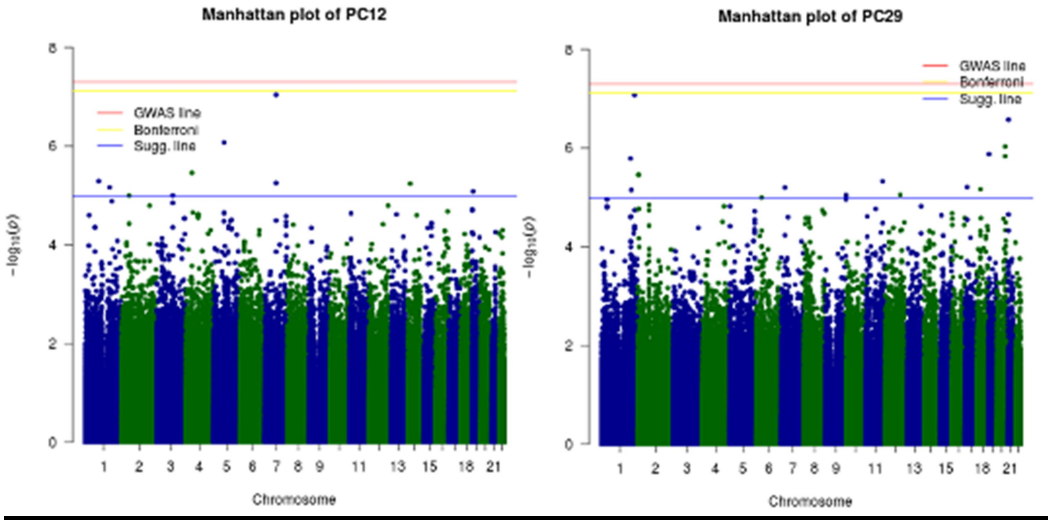
```
  return(df_inflation)
}
```

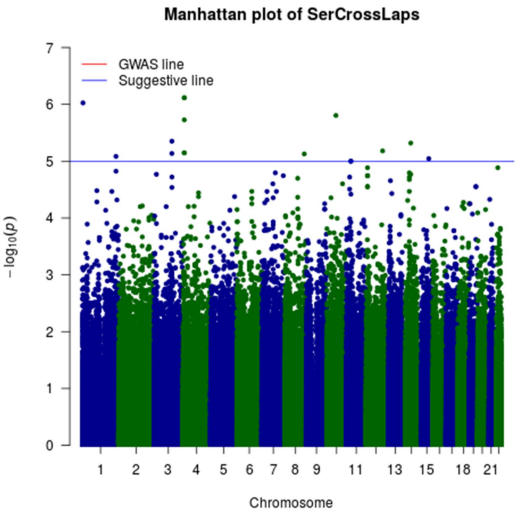
## First mapping approach

```
top.snps <- function(df, CHR, num=5){
  library(data.table)
  library(rsnps)

  #Conversion to a data table
  dat.tab <- data.table(df)
  setkey(dat.tab, pSNP)

  #Top snps
  snps <- as.character(dat.tab[chr==CHR]$SNP[1:num])

  #Filter rs snps
  snps_rs <- c()
  for(snp in snps){
    if(substr(snp,1,2)=='rs'){
      snps_rs <- append(snps_rs, snp)
    }
  }
  #Query for mapping
  tabla_NCBI <- NCBI_snp_query(snps_rs)

  #Out
  lista <-
list(tabla_global=dat.tab[1:num,],tabla_chr=dat.tab[chr==CHR][1:num,],
SNPs=snps, NCBI=tabla_NCBI)
  return(lista)
}
```

# A.4. Results

## A.4.1. Manhattan plots



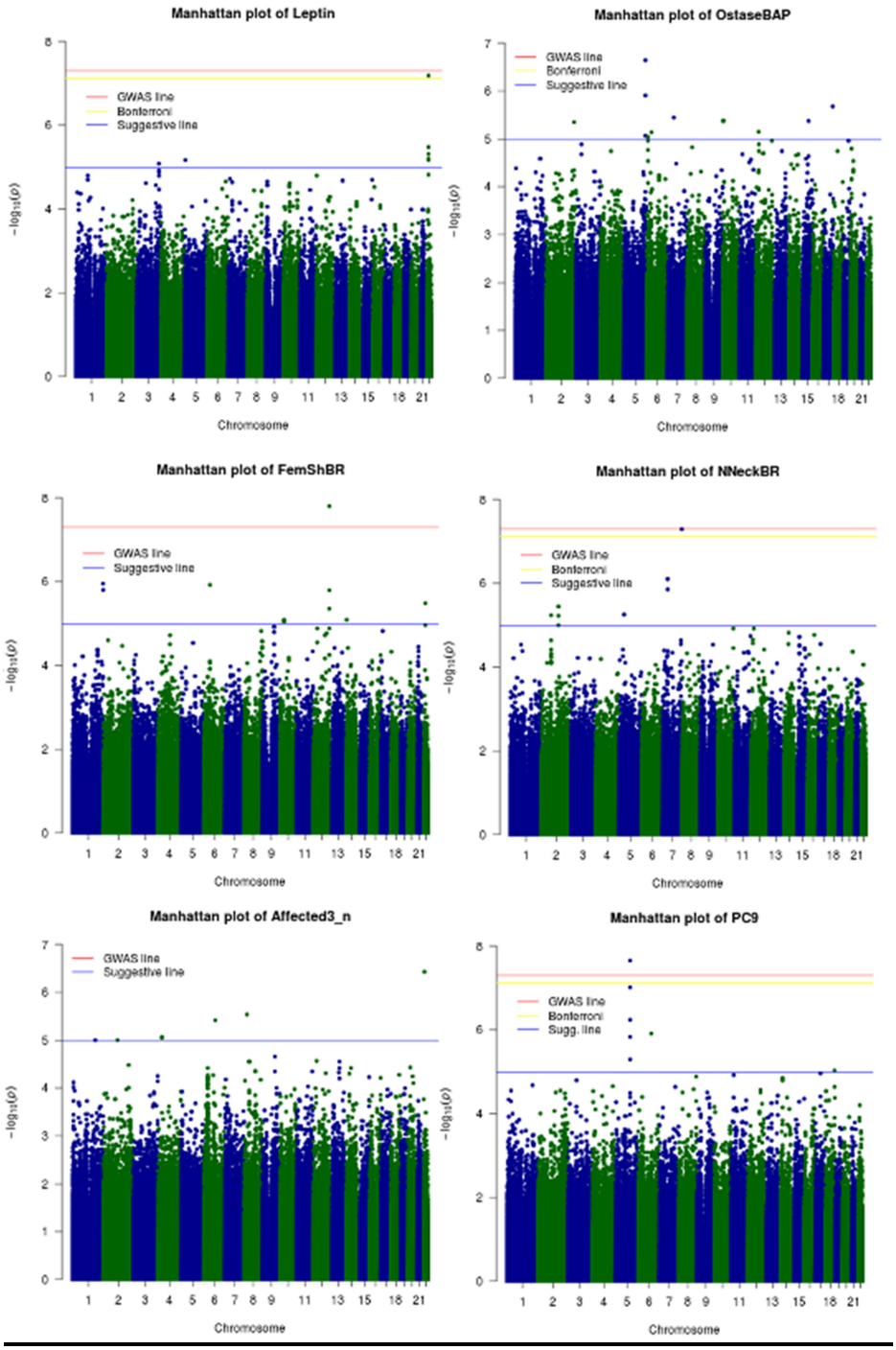
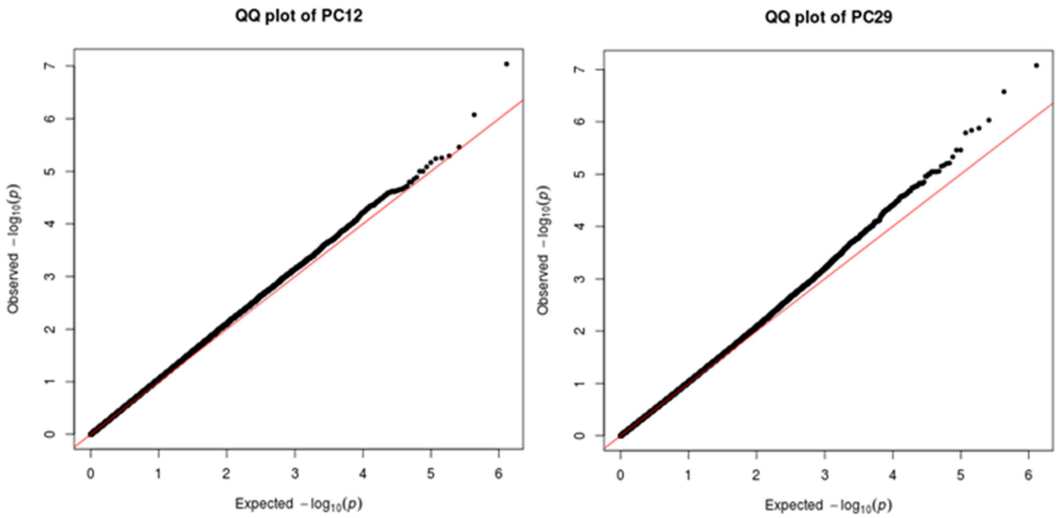



*Figure A.4.1. Manhattan plots of the traits where significant or suggestive markers were found in the association study*

ETSEIB

## A.4.2. Q-Q plots


QQ plot of PC12
Observed $-\log_{10}(p)$
Expected $-\log_{10}(p)$
QQ plot of PC29
Observed $-\log_{10}(p)$
Expected $-\log_{10}(p)$

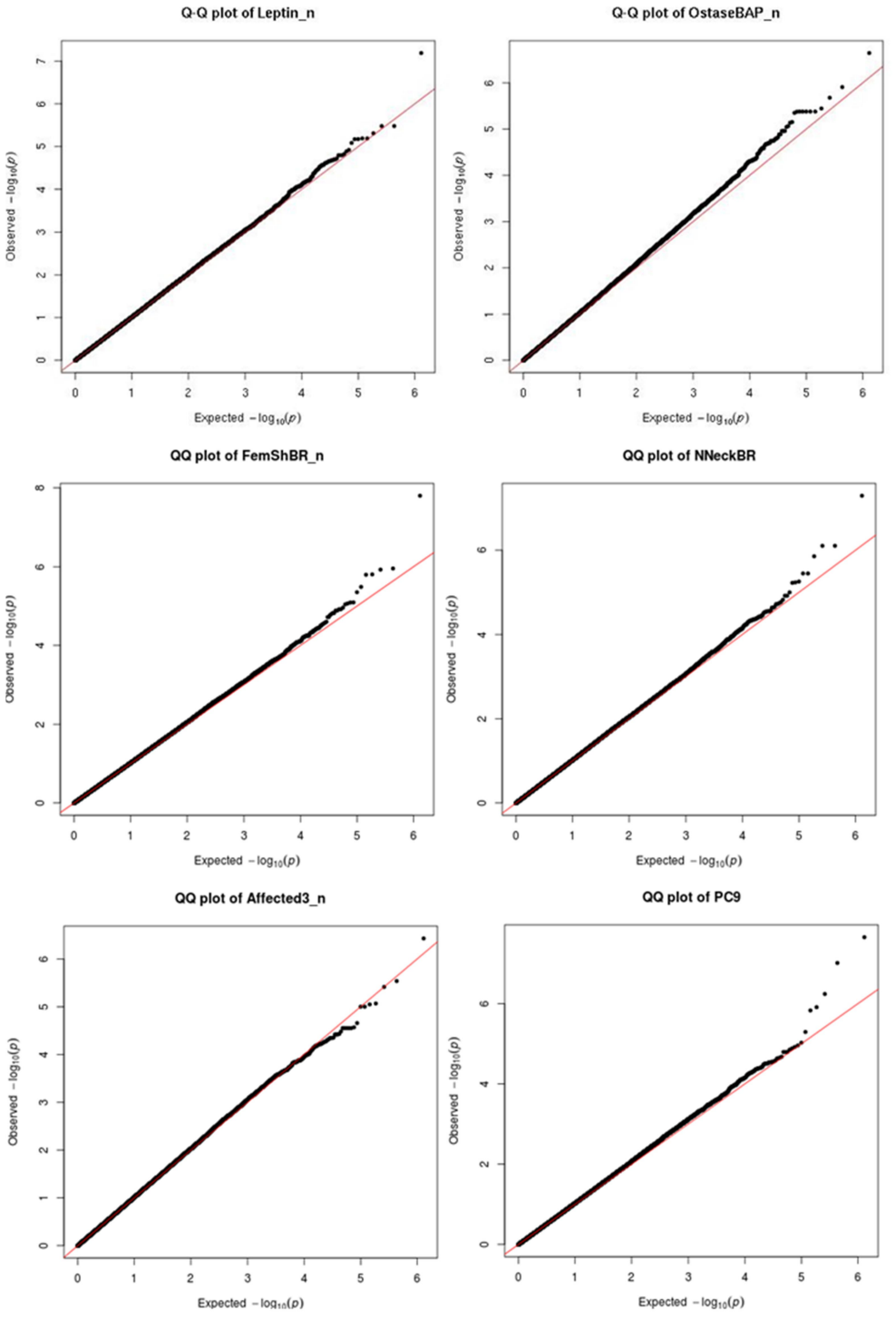



*Figure A.4.2. Q-Q plots of the traits where significant or suggestive markers were found in the association study*