Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

Lisa Erfjord

Statistical Analysis of Interaction Effects Between Environmental and Genetic Factors

Can physical activity reduce the effects of genetic predispositions to cardiovascular disease?

Data set from the HUNT Study

Master's thesis in Industrial Mathematics
Supervisor: Mette Langaas
Co-supervisor: Anja Bye
June 2021

NTNU
Kunnskap for en bedre verden

Lisa Erfjord

Statistical Analysis of Interaction Effects Between Environmental and Genetic Factors

Can physical activity reduce the effects of genetic predispositions to cardiovascular disease?

Data set from the HUNT Study

Master's thesis in Industrial Mathematics
Supervisor: Mette Langaas
Co-supervisor: Anja Bye
June 2021

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

NTNU

Kunnskap for en bedre verden

Preface

This Master's thesis constitutes the course TMA4900 for the Industrial Mathematics program at the Norwegian University of Science and Technology (NTNU). The topic for this analysis evolved from the cooperation between my supervisor Mette Langaas and my co-supervisor Anja Bye, at the Faculty of Medicine and Health Sciences.

I would like to thank my supervisor Mette Langaas for the educational experience, informative discussions, and outstanding guidance in the process of writing this thesis. I would also like to thank Anja Bye for providing the HUNT data set and guidance regarding the medical aspect of the analysis. Lastly, I would like to thank Per Kristian Hove, from IT support at IMF, for helping set up access to the cerg lab in the HUNT cloud.

I would also like to thank my fellow students at NTNU matteland for the many fun mathematical discussions we have had. Lastly, I would like to thank my family for their continuing support.

Lisa Erfjord
June 2021

Statistical Analysis of Interaction Effects Between Environmental and Genetic Factors

Can physical activity reduce the effects of genetic predisposition to cardiovascular disease?

Lisa Erfjord

Abstract

The primary focus of this thesis is to investigate the interaction effects of genetic factors and physical activity on the future risk of developing cardiovascular heart diseases. This includes getting familiar with the data and the theory of both the medical and statistical aspects. It also includes investigating different approaches to analyzing the interaction effect by using several statistical models.

We use the HUNT data set from the Trøndelag Health Study and data on hospital admission from Helse Nord-Trøndelag. Our final data set consists of 41 005 individuals, where 1 303 individuals developed cardiovascular heart disease within nine years. We have eight environmental covariates, including self-reported physical activity. Additionally, we add four principal components as covariates to address population stratification. The genetic factors are 50 different genetic markers that are known to increase the risk of cardiovascular heart disease. The outcome is whether the participant has suffered from cardiovascular heart disease or not.

In this analysis, the interaction effect is modeled using two different approaches for two different types of models. First, we fit two tree ensemble models, namely random forest and extreme gradient boosting. For the tree ensemble models, we investigate the interaction effect by using partial dependence plots. We also fit a logistic regression model, where we investigate the interaction effect in a model with both the main effects and the interaction effects. In the logistic regression, we use information from the tree ensemble model fits to specify the functional relationships between the covariates and the outcome.

From fitting the models, we conclude that being inactive increases the predictive probability of developing cardiovascular heart disease. Furthermore, some of the genetic markers affect the predictive probability of developing cardiovascular heart disease. However, the physical activity-genetic marker interaction effect does not appear to affect the predictive probability of developing CHD for any of the genetic markers. Hence, we cannot conclude that physical activity can reduce the effects of genetic predisposition to cardiovascular disease based on this analysis. Finally, we discuss the strengths and weaknesses of our analysis and present possible future work.

Statistisk analyse av interaksjonseffekt mellom miljø og genetiske faktorer

Kan fysisk aktivitet redusere effekten av genetisk risiko for å utvikle hjerte- og karsykdommer?

Lisa Erfjord

Sammendrag

I denne oppgaven analyseres interaksjonseffekten mellom genetiske faktorer og fysisk aktivitet når det kommer til risiko for å utvikle hjerte- og karsykdommer. Dette inkluderer å lage datasett, bli kjent med teorien fra en medisinsk og statistisk synsvinkel og å undersøke flere mulige metoder for å analysere interaksjonseffekten ved bruk av statistiske modeller.

Vi bruker et datasett fra HUNT, som er en helseundersøkelse fra Nord-Trøndelag, og sykehusdata fra Helse Nord-Trøndelag. Datasettet som brukes i denne analysen inneholder informasjon om 41 005 deltagere, der 1 303 deltagere utvikler en form for hjerte- og karsykdommer i løpet av ni år. Vi bruker åtte miljø-kovariater, inkludert selvrapportert fysisk aktivitet. De fire første prinsipale komponentene er også inkludert som kovariater for å korrigere for genetisk og miljøbasert korrelasjon mellom deltagerne. De genetiske kovariatene er 50 genetiske markører som har vist seg å øke risikoen for å utvikle hjerte- og karsykdommer. Responsen er om deltagerne har utviklet en form for hjerte- og karsykdom eller ikke.

I denne analysen brukes to typer statistiske modeller. Først brukes random forest og extreme gradient boosting, som er tre-ensemble modeller. For disse modellene kan delvis avhengige plot bli brukt for å analysere interaksjonseffekten. Vi bruker også logistisk regresjon, der både hovedeffekten og interaksjonseffekten blir estimert. I logistisk regresjon vil informasjon fra de tilpassede tre-ensemblemodellene bli brukt til å spesifisere den funksjonelle relasjonen mellom variablene og responsen.

Basert på resultatene fra de ulike statistiske modellene konkluderer vi med at fysisk aktivitet reduserer risikoen for å utvikle hjerte- og karsykdommer. Noen av de genetiske markørene hadde også en signifikant effekt på sannsynligheten for å utvikle hjerte- og karsykdommer. Men interaksjonseffekten mellom de genetiske markørene og fysisk aktivitet viste seg å ikke være signifikant. Vi kan derfor ikke konkludere med at fysisk aktivitet reduserer genetisk risiko for å utvikle hjerte- og karsykdommer. Til slutt diskuteres styrker og svakheter ved denne analysen, og vi presenterer muligheter for videre arbeid.

Contents

Chapter 1

Introduction

1.1 Motivation

Cardiovascular disease (CVD) has emerged to be the leading cause of death worldwide. In 2015 CVD caused 45% of deaths in Europe and 31% of all deaths worldwide (Townsend et al., 2015; WHO, 2017). In the next decade, we will most likely have a further increase of people at risk due to the expected increase of diabetes, inactivity, obesity, and an aging population (WHO, 2020a,b, 2018a,b). Hence, there is an urgent need for new prevention strategies to handle the increasing population at risk.

Daily physical activity (PA) has been highlighted as such a prevention strategy for CVD, as it is a cost-effective strategy that improves maximal oxygen uptake (VO_2max). VO_2max has shown to be inversely associated with CVD in population-based studies (Andersen et al., 2015). However, the amount of exercise an individual performs has been challenging to measure accurately and consistently on a large scale. The impact of sedentariness on CVD may be partly determined by a person's genetic constitution. That is, the extent to which the genetic risk for CVD can be compensated with exercise is still not known.

In order to identify whether PA can modify the genetic risk of CVD, we will perform interaction analyses between self-reported PA levels and genetic markers previously associated with CVD. In other words, we are interested in the interaction effect between a genetic factor G and an environmental factor E. The interaction term is often denoted $G \times E$. Analyzing $G \times E$ is a research area within statistical genomics. Statistical genomics is a scientific field concerned with developing statistical methods for drawing inferences from genetic data.

Several models can be used to analyze $G \times E$. We consider different approaches to model the interaction effect of PA and the genetic factors so that it can be used to give information on whether PA can be a good prevention strategy for people with a genetic risk for CVD.

1.2 Outline

In Chapter 2, we will give an introduction to the genetic factor G. Further, we will present the HUNT Data, which is the data set we use in the analysis.

In Chapter 3, we present how we will evaluate the statistical model fits.

In Chapter 4, we will give an overview of the theory behind logistic regression. The theory regarding model evaluation for this specific model will also be included.

In Chapter 5, we present the theory behind the tree ensemble methods we will use in this analysis. We start by giving an overview of classification trees. Then the theory behind random forest and extreme gradient boosting is presented. Additionally, we include theory regarding interpreting the models. We also include the theory on hyperparameter tuning, which is relevant for random forest and extreme gradient boosting.

In Chapter 6, we begin by introducing how we may model the genetic effects. Then we present how we will analyze the interaction effects between the genetic factors and physical activity, first for the logistic regression model and then for tree ensemble models.

In Chapter 7, we present our data analysis. We begin by exploring the data used in our models. Next, a random forest model is fitted, and we present the hyperparameters tuning and the model fit. The second model we fit is the extreme gradient boosting model. For this model, we present the hyperparameter tuning, the model fit, and interaction analysis. The last model presented is logistic regression. We first present how we choose to specify the functional relationship between the covariates and the outcome for this model. Then we present the model fit and the interaction analysis.

Lastly, in Chapter 8, we discuss our results from both a statistical and medical perspective. We also discuss possibilities for further work.

A detailed description of how we construct our data set can be found in Appendix A. Additional tables and figures of results from the models fitted in the data analysis in Chapter 7 can be found in Appendix B.

Chapter 2

Background

2.1 GWAS and SNPs

There is strong evidence of a genetic contribution to the risk of developing CVD. A genome-wide association study (GWAS) is an approach used in genetic research to associate genetic variations with a particular disease (NIH, 2019). The approach involves genotyping and investigating genomes, which is the complete set of genes, from a large number of people. This data can be used to look for genetic markers in the form of single-nucleotide polymorphisms (SNPs), which may be associated with an increased risk of a disease.

SNPs are the most common type of genetic variation among people. Each SNP represents a difference in a single DNA building block at a specific locus, which is a position on a chromosome (NIH, 2020). A DNA building block is called a nucleotide and can be any of the four bases adenine (A), cytosine (C), guanine (G), and thymine (T). More specifically, if there is a single nucleotide substitution at a specific locus in the genome, which is present in more than 1% of the population, there is a SNP at this specific locus. The possible nucleotide variations for a position are called *alleles*. Most human SNPs are what we call biallelic, which means that two allelic variants are segregated in the population. Additionally, the minor allele frequency (MAF) is the frequency at which the minor allele occurs in a population.

Multiple SNPs have been identified to have an association with CVD (Nikpay et al., 2015). We will investigate whether physical activity influences the SNPs of interest and contributes to a significant reduction change in the genetic risk of CVD.

2.2 HUNT Data

The data set we will analyze in this thesis is from the HUNT study (The Nord-Trøndelag Healthy Study). HUNT is a large population-based cohort including 125 000 Norwegian participants (NTNU, 2020a). The study was carried out in the Nord-Trøndelag county of Norway, and every citizen over 20 years was invited. Four waves of the study have been conducted, namely HUNT1 (1984-1986), HUNT2 (1995-1997), HUNT3 (2006-2008),

and HUNT4 (2017-2019). The study includes data from surveys, interviews, clinical measurements, and biological samples. For all participants where this was feasible, DNA was extracted from blood samples, and GWAS analyses were performed (Krokstad et al., 2013). The data we will analyze is from the third survey (HUNT3), where there are around 50 000 genotyped participants.

Baseline Data

The environmental covariates are chosen from a medical perspective and are collected from the HUNT Databank. The environmental covariates are sex, age, smoking status, body mass index, serum cholesterol level, serum high-density lipoprotein cholesterol level, systolic blood pressure and physical activity. The physical activity level is self-reported based on three questions with response options:

Question 1: "How frequently do you exercise?" "Never" (0), "Less than once a week" (0), "Once a week" (1), "2-3 times a week" (2.5), and "Almost every day" (5).

Question 2: "If you exercise as frequently as once or more times a week: How hard do you push yourself?" "I take it easy without breaking a sweat or losing my breath" (1), "I push myself so hard that I lose my breath and break into sweat" (2), and "I push myself to near exhaustion" (3).

Question 3: "How long does each session last?" "Less than 15 minutes" (0.1), "16-20 minutes" (0.38), "30-60 minutes" (0.75) and "More than an hour" (1.0).

The physical activity level is categorized using a physical activity index score called the Kurtze score, defined by Rangul et al. (2008). Each participant's response to the three questions is multiplied using the score in the parenthesis. As the second and third questions only address people who exercise at least once a week, both "Never" and "Less than once a week" yield a score of zero. A descriptive analysis of the Kurtze score in the HUNT data is presented in Appendix A.1.

Another way of categorizing physical activity is categorizing the participants as either active or inactive, where a score under a certain value is categorized as inactive, and a higher score is categorized as active. We will however use the Kurtze score in this analysis.

Additionally, we add the first four principal components (PC) as covariates. We do this because GWAS studies are susceptible to bias due to population stratification and participants may be related to eachother. Adding principal components as covariates is a standard method to correct this bias (Zhao et al., 2018). There may also be a correlation between the participants due to environmental similarities. Hence this will also be corrected for, to some degree, when using PCs as covariates. The PCs in HUNT are a projection in the Human Genome Diversity Project (HGDP) implemented by Taliun et al. (2017). Based on the eigenvalues from HGDP, the four first PCs explain 45.8% of the HGDP variability.

SNPs

A study by Holmen et al. (2014) investigated whether rare SNPs affect the risk of developing cardiovascular heart disease (CHD). The result from the study was that none of the rare SNPs had a significant effect on the risk of developing CHD. However, to do quality control of the methods, 54 known GWAS SNPs that increase the risk of developing CHD were also analyzed. Holmen et al. (2014) selected 54 SNPs from Deloukas et al. (2013), Kathiresan (2008), and Schunkert et al. (2011). We found 50 of the 54 SNPs in the HUNT Databank for HUNT3. These 50 SNPs are the SNPs we choose to use as genetic covariates in this analysis. An overview of the 50 SNPs can be found in Table A.2 in Appendix A.2. The genetic position, the rs number, the risk allele, and the results from GWAS studies and the HUNT study for each SNP can also be found in Table A.2. The table is copied from the supplementary of the HUNT study by Holmen et al. (2014).

Cardiovascular Heart Disease

This analysis will use cardiovascular heart disease (CHD) as the outcome since this is what our genetic factors are associated with. CHD is a particular case of CVD, and occurs if a participant has suffered from acute myocardial infarction or subsequent myocardial infarction. International classification of diseases (ICD) codes are a system created by WHO, where diagnoses are coded and used for statistics of diseases (ehelse, 2021). Acute myocardial infarction has ICD code I21, and subsequent myocardial infarction has ICD code I22 (WHO, 2016). We will exclude the participants with angina pectoris, which has ICD code I20, as this is a less severe variant of CHD. This definition of CHD is identical to the one used by Holmen et al. (2014).

We identify the participants that have suffered from CHD within the following ten years using Hospital Data from *Helse Nord-Trøndelag* (HNT). Furthermore, the participants will be categorized as a case if they have suffered from CHD or control if they have not. For more details see Appendix A.3. We will further denote the outcome as CHD in this analysis.

HUNT Cloud

All medical and genetic data are available to us through the HUNT cloud (NTNU, 2020b). The HUNT cloud delivers a digital infrastructure that enables researchers to analyze sensitive data in controlled environments. For this analysis, we require the genetic data of the participants. However, it is possible to identify a participant from the SNP data. Thus, in order to follow the guidelines on handling patient data, see for example NTNU (2020c), we do not download the data we are using. For this reason, we do all the coding and analysis on a virtual machine in the HUNT cloud.

Chapter 3

Model Evaluation

In order to measure the performance of the statistical models, we need tools for model assessment and evaluation. This section will therefore introduce performance measures used in this analysis.

3.1 Training, Validation and Test Set

When measuring the performance of a statistical method on a given data set, we are interested in how well the model performs on new data. A common practice is to randomly divide the data set into a training set and a test set. The training set is used to fit the model, while the test set is used to evaluate the performance of the fitted model (Ch.7 by Hastie et al. (2001)).

Since we are interested in measuring how the statistical model performs on new data, we will not use the test sample until we do the final evaluation. In other words, we will treat the test error as if it was an error on unexplored data. Hence, the goal is to minimize the test error. As we have a classification problem, one choice of error is the misclassification rate, which is the proportion of mistakes made by the predictor.

An ideal predictor will capture the patterns in the data while ignoring the noise. In order to capture the general patterns, the model has to be complex enough. When the model becomes more complex, the training error decreases, and hence we get a decrease in bias. However, this also gives an increase in the variance. In other words, the training error consistently decreases when the model gets more complex, but it may lead to overfitting. Overfitting the training data will capture noise instead of actual patterns in the data. It will typically give a minimal training error but a considerable test error. Thus, by using a test set, we can choose a model that is not too complex in order to prevent overfitting the training data.

3.2 Sensitivity, Specificity, and AUC

In our analysis, we have a binary outcome given by

$$Y = \begin{cases} 0 & \text{for participants who do not suffer from CHD,} \\ 1 & \text{for participants who suffer from CHD.} \end{cases}$$

The participants classified as 0 are denoted controls, while participants classified as 1 are denoted cases. Based on the predicted outcome from some fitted model and the actual outcome, it is common to define a confusion matrix as in Table 3.1.

	Predicted 0	Predicted 1	Total
True 0	True Negative (TN)	False Positive (FP)	N
True 1	False Negative (FN)	True Positive (TP)	P
Total	N*	P*	

Table 3.1: Confusion matrix

Here 0 denotes individuals not suffered from CHD and 1 denotes suffered from CHD. Moreover, let the sensitivity, also called the true positive rate (TPR), and specificity, which is 1 minus the false positive rate (FPR), be given as

$$\text{sensitivity} = \text{TPR} = \frac{\text{TP}}{\text{P}},$$
$$\text{specificity} = 1 - \text{FPR} = \frac{\text{TN}}{\text{N}}.$$

Then, the goal for the classification rule is to have both high sensitivity and specificity.

A graphical display of the sensitivity against specificity as a function of the possible cut-off values on the probability of disease is called a ROC curve. A straight line as a ROC curve will then represent a model that classifies the outcome randomly. As the goal is to have high sensitivity and specificity, the ideal ROC curve hugs the top left corner. A visualization of the ROC curve for a random and a perfect classifier, inspired by the figures made by Saito and Rehmsmeier (2015), is presented in Figure 3.1.

Furthermore, the AUC score is the area under the ROC curve, ranging from 0 to 1. It is a measure of how well the model performs. AUC is a helpful score for comparing the performance of different classification models, where a higher score indicates a better classifier. The AUC score and ROC curve should be estimated using the test set, where the model we evaluate is fitted using the training set.

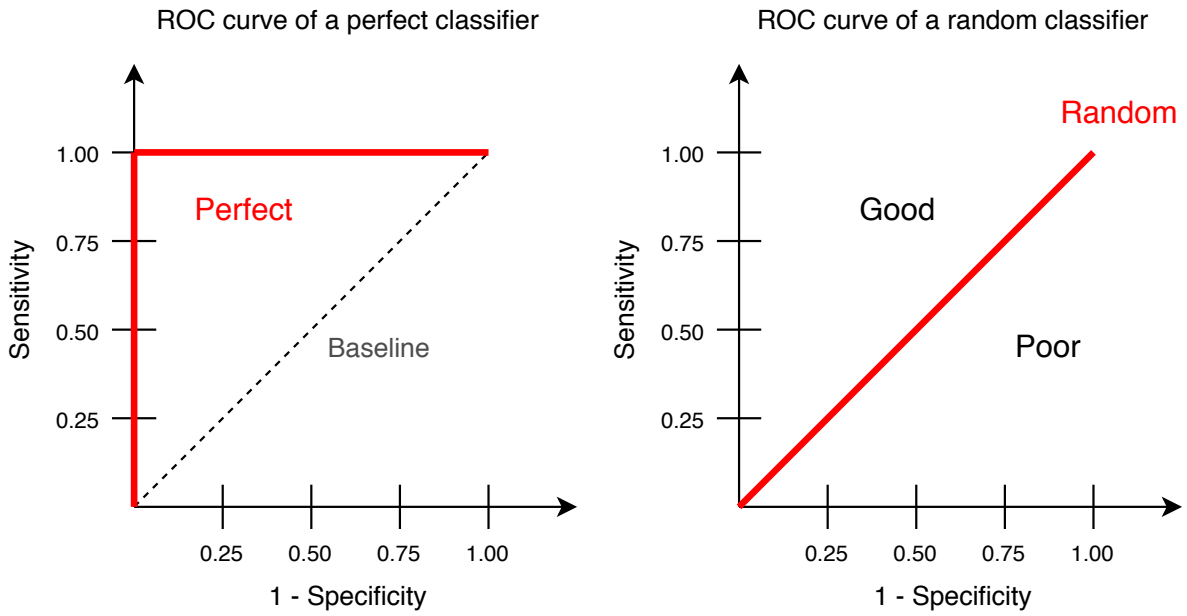


Figure 3.1: A graphical display of the ROC curve for a perfect and a random classifier.

3.2.1 DeLong Test

If two ROC curves are constructed from the same data set, we denote the two curves as paired. Two paired ROC curves can be compared by using a method called DeLong. The method was developed by DeLong et al. (1988) and tests whether one model has a statistically significantly different AUC score from an alternative model. The test is based on U statistics theory and asymptotic normality.

Denote the empirical AUC score of a model to be $\hat{\theta}^{(A)}$, and the empirical AUC score of an alternative model to be $\hat{\theta}^{(B)}$. The null hypothesis is then $H_0 : \hat{\theta}^{(A)} - \hat{\theta}^{(B)} = 0$. In order to test whether model A is better than model B in terms of the AUC score, we calculate the z score as follows

$$z = \frac{\hat{\theta}^{(A)} - \hat{\theta}^{(B)}}{\sqrt{\mathrm{Var}\left[\hat{\theta}^{(A)} - \hat{\theta}^{(B)}\right]}} = \frac{\hat{\theta}^{(A)} - \hat{\theta}^{(B)}}{\sqrt{\mathrm{Var}\left[\hat{\theta}^{(A)}\right] - \mathrm{Var}\left[\hat{\theta}^{(B)}\right] - 2\mathrm{Cov}\left[\hat{\theta}^{(A)}, \hat{\theta}^{(B)}\right]}}.$$

Here $\mathrm{Var}[\cdot]$ denotes the variance and $\mathrm{Cov}[\cdot, \cdot]$ denotes the covariance. To find the z score, we have to calculate the empirical AUC scores, the variances, and the covariance. Under H_0, the z score can be approximated by the standard normal distribution. Thus, if the z score deviates significantly from zero, we can conclude that $\hat{\theta}^{(A)} \neq \hat{\theta}^{(B)}$ at a certain significance level. The DeLong method can also be used to construct confidence intervals for the AUC.

3.2.2 Precision Recall AUC

In our data set, only a few participants are categorized as a case. That is, only a few participants have suffered from CHD, and we will therefore refer to the data as imbalanced. In a strongly imbalanced data set, where the number of controls outweighs the number of cases significantly, it may be misleading to look at the specificity. Another version of the

AUC can be used instead, called the precision-recall AUC score (PR AUC) (Saito and Rehmsmeier, 2015). PR AUC looks at the positive predictive value (PPV) instead of the false positive rate (FPR). With notation from Table (3.1) the PPV is given as

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}}.$$

A graphical display of sensitivity, also referred to as recall, against PPV, also referred to as precision, is called the precision-recall (PR) curve. A classifier with a random performance level will have a horizontal line at $\frac{P}{P+N}$. Then the area above that line will be the area of good performance levels, and the area below will be the area of poor performance levels. Thus, a perfect PR curve will hug the upper right corner. Visualization of two PR curves for a random and a perfect classifier, inspired by the figures made by Saito and Rehmsmeier (2015), are presented in Figure 3.2.

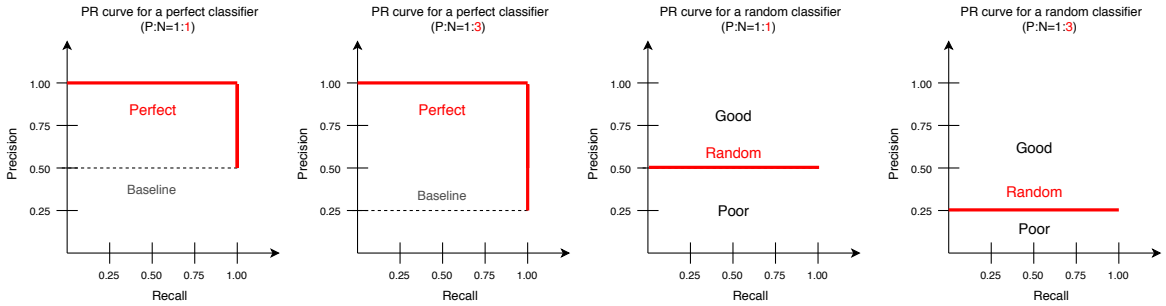


Figure 3.2: A graphical display of the PR curve for two perfect and two random classifiers. The horizontal line is given as $\frac{P}{P+N}$.

The PR AUC score is the area under the PR curve and ranges from 0 to 1. The PR AUC score and PR curve are estimated using the training or test set, where the model we evaluate is fitted using the training set.

Chapter 4

Logistic Regression

This chapter presents a statistical model where the interaction effects between physical activity and the genetic covariates can be modeled and analyzed, namely logistic regression. Logistic regression is a frequently used statistical method for analyzing binary data in biostatistics and statistical genomics. For this reason, logistic regression is one of the statistical models used in the analysis. The logistic regression model is a special case of a generalized linear model (GLM), and an introduction to GLM will thus be presented first. Additionally, tools for evaluating the model are also presented.

4.1 Generalized Linear Models

The GLM consists of two elements, a random component and a systematic component described below (Ch.5 by Dunn and Smyth (2018)).

The Random Component

Assume that the probability distribution belongs to a family of distributions called the exponential dispersion model family. Consider n independent observations $y_1, y_2, ..., y_n$, where y_i is a realization of a random variable Y_i when $i = 1, 2, ..., n$. Assume that Y_i follows a distribution in this family with a probability function of the form

$$Y_i \sim f_{Y_i}(y_i|\theta_i, \phi),$$

where θ_i is a vector of parameters and

$$f(y_i|\theta_i, \phi) = a(y_i, \phi) \exp\left\{\frac{y_i\theta_i - \kappa(\theta_i)}{\phi}\right\}. \tag{4.1}$$

Here $\kappa(\theta_i) < \infty$ is a known function called the cumulant function, $\phi > 0$ is the dispersion parameter and $a(y, \phi)$ is a normalizing function.

Systematic component

We also assume a specific form for the systematic component. Namely, a linear predictor

$$\eta_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} = \boldsymbol{\beta}^T \mathbf{x}_i,$$

where $\boldsymbol{\beta} = \{\beta_0, \beta_1, ..., \beta_p\}$ is a vector of unknown coefficients and $\mathbf{x}_i = \{1, x_{i1}, , x_{i2}, ..., x_{ip}\}$ is a vector of p predictors for observation i. The linear predictor is linked to the mean μ through a link function $g(\cdot)$, so that

$$g(\mu) = \eta. \tag{4.2}$$

The link function is a known monotonic, differentiable function that ensures that the function is one-to-one and can be estimated.

4.2 Logistic Regression Models

Since we have a binary response variable in our dataset, a logistic regression model can be used to fit the data (Ch.4 by Hastie et al. (2001)). Denote the probability for an observation to come from the class $Y_i = 1$ to be $\pi_i(\mathbf{x}_i)$ and $Y_i = 0$ to be $1 - \pi_i(\mathbf{x}_i)$. That is,

$$Y = \begin{cases} 0 & \text{with probability } \mathrm{P}(Y_i = 0 | X_i = \mathbf{x}_i) = 1 - \pi_i(\mathbf{x}_i), \\ 1 & \text{with probability } \mathrm{P}(Y_i = 1 | X_i = \mathbf{x}_i) = \pi_i(\mathbf{x}_i). \end{cases}$$

The Random component

Now assume $\{y_1, y_2, ..., y_n\}$ are independent observations. Then the probability mass function for $Y_i, i = 1, 2, ..., n$ is binomially distributed with one trial. This can be expressed as

$$f_i(y_i|\pi_i) = \begin{pmatrix} 1 \\ y_i \end{pmatrix} \pi_i^{y_i} (1 - \pi_i)^{1 - y_i}.$$

The binomial distribution is an exponential family since it can be rewritten as

$$f_i(y_i|\pi_i) = \binom{1}{y_i} \exp\left(y_i \log\left(\frac{\pi_i}{1 - \pi_i} \right) + \log\left(1 - \pi_i\right) \right),$$

which is in the form of Equation (4.1) with $\theta = \log \frac{\pi}{1-\pi}$, $\kappa(\theta) = -\log(1 - \pi)$, $\phi = 1$ and $a(y, \phi) = \binom{1}{y}$.

The Systematic Component

For logistic regression, the link function given by (4.2) is chosen to be the logit function, expressed as

$$g(\pi_i) = \eta_i = \mathrm{logit}(\pi_i) = \log \frac{\pi_i}{1 - \pi_i} = \boldsymbol{\beta}^T \mathbf{x}_i. \tag{4.3}$$

4.3 Interpretation

In order to interpret the probability π_i, observe that

$$\pi_i = \text{logit}^{-1}(\boldsymbol{\beta}^T \mathbf{x}_i) = \frac{e^{\boldsymbol{\beta}^T \mathbf{x}_i}}{1 - e^{\boldsymbol{\beta}^T \mathbf{x}_i}}.$$

Hence, the probability of Y_i can be directly found in logistic regression.

Odds Ratio

In order to interpret the change in the response variable when a predictor $x_{ij}, j = 0, 1, ..., p$ changes, it is useful to look at the odds ratio. The odds for an individual i is the ratio of $Y_i = 1$ to $Y_i = 0$, given by

$$\frac{\pi_i}{1 - \pi_i} = e^{\boldsymbol{\beta}^T \mathbf{x}_i} = e^{\beta_0} e^{\beta_1 x_{i1}} e^{\beta_2 x_{i2}} \cdots e^{\beta_p x_{ip}}$$

The interpretation of a one-unit increase in x_{ij} is that the odds are multiplied by e^{β_j}.

4.4 Parameter Estimation

Maximum Likelihood

The parameters $\boldsymbol{\beta}$ can be estimated by maximizing the likelihood of our model correctly predicting any y_i given $\mathbf{x}_i$, which is equal to maximizing the likelihood function $L(\boldsymbol{\beta})$. This again can easier be estimated by maximizing the log-likelihood function given by

$$\begin{aligned} l(\boldsymbol{\beta}) = \log L(\boldsymbol{\beta}) &= \sum_{i=1}^{n} \Big(y_i \log \pi_i + (1 - y_i) \log(1 - \pi_i) \Big) \\ &= \sum_{i=1}^{n} \left(y_i \log \left(\frac{\pi_i}{1 - \pi_i} \right) + \log(1 - \pi_i) \right) \\ &= \sum_{i=1}^{n} \Big(y_i \boldsymbol{\beta}^T \mathbf{x}_i - \log\big(1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}\big) \Big) \end{aligned}$$

In order to maximize this, we can find the derivative of each β_j and set it equal to zero.

$$\frac{\partial l(\boldsymbol{\beta})}{\partial \beta_j} = \sum_{i=1}^{n} \mathbf{x}_i \left(y_i - \frac{e^{\boldsymbol{\beta}^T \mathbf{x}_i}}{1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}} \right) = 0$$

To solve this, we can use the Newton-Raphson algorithm, at iteration $k+1$ $\boldsymbol{\beta}$ is updated in the following way

$$\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^k - H^{-1} \frac{\partial l(\boldsymbol{\beta}^k)}{\partial \boldsymbol{\beta}^k} \tag{4.4}$$

where H is the Hessian of $\boldsymbol{\beta}$ given by

$$H_{j,l} = \frac{\partial^2 l(\boldsymbol{\beta})}{\partial \beta_j \partial \beta_l} = -\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T \frac{e^{\boldsymbol{\beta}^T \mathbf{x}_i}}{1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}} \left(1 - \frac{e^{\boldsymbol{\beta}^T \mathbf{x}_i}}{1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}}\right), j, l = 0, 1, ..., p.$$

Iteratively Re-Weighted Least Squares

Furthermore, this can be calculated using a method called iteratively re-weighted least squares (IWLS). For that, we need Equation (4.4) in matrix form, which we get by rewriting it as follows

$$\boldsymbol{\beta}^{k+1} = \boldsymbol{\beta}^k + (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T (\boldsymbol{y} - \boldsymbol{\pi}) = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \mathbf{z}.$$

Here $\mathbf{y}$ is a n-dimensional column vector of observations, $\boldsymbol{\pi}$ is a n-dimensional column vector of fitted probabilities π_i, $\mathbf{X}$ is the $n \times (p+1)$ matrix with x_i as column, $\mathbf{W}$ is a diagonal $n \times n$ matrix of weights given by

$$\begin{aligned}
\mathbf{W} &= \frac{e^{\boldsymbol{\beta}^T \mathbf{x}_i}}{1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}} \left(1 - \frac{e^{\boldsymbol{\beta}^T \mathbf{x}_i}}{1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}}\right) \text{ and} \\
\mathbf{z} &= X\boldsymbol{\beta}^k + W^{-1}(\mathbf{y} - \pi), \\
\frac{\partial l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} &= \mathbf{X}^T (\boldsymbol{y} - \boldsymbol{\pi}), \\
\frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} &= -\mathbf{X}^T \mathbf{W} \mathbf{X}.
\end{aligned}$$

Using this, we find the estimate of $\boldsymbol{\beta}$.

Distribution of Parameter Estimates

It is useful to understand which covariates influence the outcome. For this, we can construct confidence intervals for the βs and test whether the parameters are significantly different from zero or not. This can be done by assuming that all the estimated β_j are approximately normally distributed with a mean $\hat{\beta}_j$ and a variance $\widehat{\text{Var}}(\hat{\beta}_j)$, which follows from the fact that the MLE follows an approximately multivariate normal distribution for large sample sizes. That is,

$$\hat{\boldsymbol{\beta}} \sim N_p(\boldsymbol{\beta}, \text{Cov}(\hat{\boldsymbol{\beta}})).$$

Further $\text{Cov}(\hat{\boldsymbol{\beta}})$ can be replaced by $\widehat{\text{Cov}}(\hat{\boldsymbol{\beta}})$, where $\widehat{\text{Cov}}(\hat{\boldsymbol{\beta}}) = \boldsymbol{I}^{-1}(\hat{\boldsymbol{\beta}}) = (\boldsymbol{X}^T \hat{\boldsymbol{W}} \boldsymbol{X})^{-1}$ and $\boldsymbol{I}(\hat{\boldsymbol{\beta}})$ is the Fisher information. Then, with significance level $1 - \alpha$ and $z_{\frac{\alpha}{2}}$ being the critical value in the standard normal distribution, the confidence interval for a parameter $\hat{\beta}_j$ can the be expressed as

$$\left[\hat{\beta}_j - z_{\frac{\alpha}{2}} \sqrt{\widehat{\text{Var}}(\hat{\beta}_j)}, \hat{\beta}_j + z_{\frac{\alpha}{2}} \sqrt{\widehat{\text{Var}}(\hat{\beta}_j)}\right].$$

Here $\widehat{\text{Var}}(\hat{\beta}_j)$ are the diagonal elements of $\widehat{\text{Cov}}(\hat{\boldsymbol{\beta}})$. An interpretation of this can be that the confidence interval is a range of values which is likely to include the parameter β_j with a certain degree of confidence.

4.5 Deviance and AIC

Maximizing the likelihood is equivalent to minimizing the deviance. The deviance is a measure of how much unexplained variation there is in the model. Hence, it is helpful to look at the deviance to ascertain how well our model performs (Ch.5 Fahrmeir et al. (2013)).

When assessing the fit of an estimated model, we can compare the estimated model with the best fit of the data. When the data have been maximally grouped, the group-specific parameter π_i can be estimated by using the mean value $\bar{y}_i$. This corresponds to the best fit of the data, called the saturated model. The saturated model will then serve as a benchmark when evaluating the fit of estimated models. Hence, we can formally test the significance of the departure between the estimated model and the saturated model using deviance.

Now denote p to be the number of predictors for the estimated model and G to be the number of groups. The deviance is then defined by

$$D = -2\sum_{i=1}^{G}\Big(l_i(\hat{\pi}_i) - l_i(\bar{y}_i)\Big), \tag{4.5}$$

where $l_i(\hat{\pi}_i)$ and $l_i(\bar{y}_i)$ are the log-likelihood of group i for the estimated and the saturated model respectively. The deviance compares the log-likelihood of the estimated model with the largest value of the log-likelihood that can be attained. If the number of participants is sufficiently large in each group, the approximate distribution of the deviance is

$$D \sim \chi^2_{G-p}. \tag{4.6}$$

Based on this approximate distribution, the model fit can be evaluated by comparing the observed value of the test statistic to the corresponding quantile of the χ^2_{G-p}-distribution. A lower value of the deviance indicates a better model. However, notice that the deviance does not penalize the number of predictors p in the estimated model. A strategy where we only minimize the deviance will usually result in an overfit model choice. Thus, it is helpful to consider the model complexity in order to avoid overfitting.

Akaike's Information Criterion

A goodness of fit measure which penalizes the number of predictors is Akaike's information criterion (AIC). AIC is defined as

$$AIC = -2l(\hat{\boldsymbol{\beta}}) + 2p.$$

This can also compare models where the data distribution is from the same exponential family and use the same link function. A low AIC value is desirable.

4.6 Likelihood Ratio Test

In order to assess the goodness of fit of different models and hence choose the final model, the likelihood ratio test (LRT) can be used. For instance, in the case where we are interested in which and how many explanatory variables are sufficient.

Consider the case where we are interested in comparing model A with p predictors with model B with q predictors, where A is nested in B. Let the hypothesis be of the form

$$\mathbf{H}_0 : \boldsymbol{\beta} = \boldsymbol{\beta}_A = (\beta_1, ..., \beta_q)^T \text{ vs. } \mathbf{H}_1 : \boldsymbol{\beta} = \boldsymbol{\beta}_B = (\beta_1, ..., \beta_p)^T,$$

where $q < p < n$ and n is the total number of observations. This hypothesis can be tested based on the likelihood ratio given by

$$\lambda = \frac{L(\hat{\boldsymbol{\beta}}_B|\boldsymbol{y})}{L(\hat{\boldsymbol{\beta}}_A|\boldsymbol{y})}.$$

Then letting $\hat{\boldsymbol{\beta}}_\Omega$ denote the saturated model and using Equation (4.5), we get the following

$$\begin{aligned} 2\log\lambda &= 2(\log L(\hat{\boldsymbol{\beta}}_B) - \log L(\hat{\boldsymbol{\beta}}_A)) \\ &= 2(\log L(\hat{\boldsymbol{\beta}}_\Omega) - \log L(\hat{\boldsymbol{\beta}}_A)) - 2(\log L(\hat{\boldsymbol{\beta}}_\Omega) - \log L(\hat{\boldsymbol{\beta}}_B)) \\ &= D_0 - D_1 = \Delta D. \end{aligned}$$

According to Equation (4.6), $D_A \sim \chi^2(n-q)$ and $D_B \sim \chi^2(n-p)$ if both models fit the data well. Hence, $\Delta D \sim \chi^2(p-q)$.

Moreover, this can be used to compute a deviance table. For instance, let $q = 1$ and $p = 2$, then the deviance table would be given as

Model	H_0	H_1	ΔD
A	$\beta_0 + \beta_1$	β_0	$D_0 - D_1$
B	$\beta_0 + \beta_1 + \beta_2$	$\beta_0 + \beta_1$	$D_1 - D_2$

Table 4.1: Deviance table

Here, the LRT statistic is ΔD, which tests the goodness of the fit.

4.7 R^2 McFadden

For logistic regression, McFadden's R^2 is one of several possible measures of explained variation by the model (McFadden, 1973). It is a measure based on the ratio between the log-likelihood of the fitted model and the intercept-only model, defined as

$$R^2_{\text{MF}} = 1 - \frac{\log L(\hat{\boldsymbol{\beta}})}{\log L(\hat{\beta}_0)}.$$

It follows that R^2_{MF} increases when the likelihood of the fitted model increases. Hence R^2_{MF} is a measure of how well the fit of the model is compared to the intercept-only model. R^2_{MF} ranges from 0 to 1, where a higher measure implies that the model explains more of the variance. Hence, a high value signifies a better model. However, R^2_{MF} increases when adding more predictors to the model. The number of predictors must therefore be taken into consideration.

4.8 Generalized Additive Models

For a logistic regression model we assume a linear relationship between the predictors and the log-odds of the outcome, such as presented in Equation (4.3). However, this assumption is not always valid. Generalized additive models (GAMs) are an extended framework of a standard linear model by also allowing non-linear function of each variable while still maintaining additivity (Ch.7 by Efron and Hastie (2016)).

In order to avoid the assumption of a linear relationship, we can replace the linear component with a non-linear function. The model can then be written as

$$g(\pi_i) = \eta_i = \text{logit}(\pi_i) = \beta_0 + \sum_{j=1}^{p} f_j(x_{ij}). \tag{4.7}$$

An advantage of GAMs is that we can automatically model non-linear relationships for the log-odds, giving potentially more accurate predictions for the outcome. Additionally, we can still easily examine the effect of each covariate on the outcome while holding all other variables fixed since the model is additive.

Chapter 5

Tree Ensembles

Other statistical models where the interaction effects between physical activity and the genetic covariates can be modeled and analyzed are tree ensembles. When having a large amount of data and a need for fitting a rich class of functions, tree-based methods are a good solution. Popular tree-based methods are random forests and boosting, which represent the fitted model by a sum of trees. They often have good predictive performance, where interaction terms are included automatically. Furthermore, a popular version of boosting is extreme gradient boosting. As we have a large amount of data and are interested in good predictive performance and interaction terms, the tree ensemble models fit well with this analysis. We will use both the random forest and the extreme gradient boosting model to analyze the interaction effect. Both models use an ensemble of classification trees.

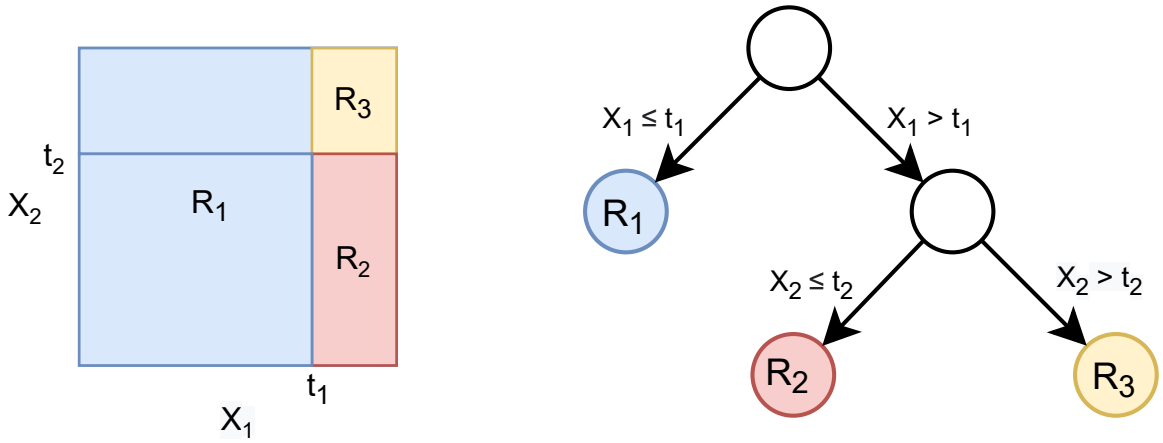
In this chapter, classification trees, random forests, and extreme gradient boosting will therefore be presented. Partial dependence plots and accumulated local effects plots are helpful tools to evaluate covariates and will also be included in the chapter. Lastly, we will discuss how tuning the hyperparameters for the different tree models can be performed using cross-validation.

5.1 Classification Trees

For all tree-based methods, the feature space is partitioned into a set of non-overlapping regions. Every observation that falls into the same region is assigned the same prediction, which is a class for classification trees (Ch. 9 by Hastie et al. (2001)).

For instance, for a two-dimensional case with two covariates, as shown in Figure 5.1, the feature space will be partitioned into a set of rectangles. The predictors in the feature space are translated into a schematic figure in the form of a tree, also shown in Figure 5.1. We use a binary-splitting approach, where we begin at the top of the tree denoted the root node. The split is chosen such that we achieve the best fit. There are now two new nodes, one for each rectangle, which can be either a terminal node or an internal node. If there are no more partitions in the rectangle, this is a terminal node. If more partitions are left, it is an internal node, and we have more binary splits. We then create new branches and nodes until all our nodes are terminal nodes. A classification tree is a model where each internal node represents a "decision", each branch represents the

outcome of the decision, and each terminal node represents a class label. Hence the paths from the root to the terminal nodes represent classification rules.



(a) Partition of a two-dimensional feature space, where $R_m, m \in \{1, 2, 3\}$ represent the regions.

(b) Tree model corresponding to the feature space, where the colored nodes are terminal nodes and the white nodes are internal nodes.

Figure 5.1

The algorithm for creating trees is a greedy algorithm since testing all possible trees is too computationally expensive. Thus, what is considered the best split for the tree at a specific step is determined in that step. That is, the algorithm searches for the local optimum and does not consider future splits. A possible stopping criterion is to perform binary splitting until each region R_m, corresponding to the terminal node m, has fewer than a minimum number of observations N_{min}.

Suppose we have partitioned the space into M regions $R_1, R_2, ..., R_M$, where each region corresponds to a terminal node in the tree, and we have N_m observations in each region. Consider a binary response, where $k = \{0, 1\}$ is the outcome Y, and define the proportion of class k observations in node m to be

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k).$$

The tree-model classifies the observation in node m to class $k(m) = \arg\max_k \hat{p}_{mk}$, which is the most common class in node m.

Gini index

In order to determine the best split, we use the Gini index. Starting with all of the data, consider a splitting variable j and splitting point t, and define the half-planes

$$R_1(j, t) = \{X | X_j \leq t\} \text{ and } R_2(j, t) = \{X | X_j > t\}.$$

We then solve

$$\min_{j,t}\left[\min_{c_1}\sum_{x_i\in R_1(j,t)}(y_i-c_1)^2+\min_{c_2}\sum_{x_i\in R_2(j,t)}(y_i-c_2)^2\right],$$

where $c_m=\frac{1}{N_m}\sum_{x_i\in R_m}y_i$. When we find the best split, the data is partitioned into two resulting regions. We then repeat this process on all the resulting regions until we have the optimal tree.

The tree size is a tuning parameter governing the model's complexity. If the aim is to construct a single tree, the strategy is to grow a large tree T_0, stopping when some minimum node size is reached. We then prune this large tree using cost-complexity pruning. Pruning is performed by first defining a subtree $T\subset T_0$ to be any tree obtained pruning T_0. Recall that the terminal nodes are indexed by m representing region R_m and that M denotes the number of terminal nodes in T. Then define the cost complexity criterion to be

$$C_\alpha(T)=\sum_{m=1}^{M}N_mQ_m(T)+\alpha M,$$

where α is the tuning parameter and Q is the Gini index given as

$$Q_m(T)=\sum_{k=1}^{K}\hat{p}_{mk}(1-\hat{p}_{mk}).$$

For each α we find the subtree $T_\alpha\subseteq T_0$ that minimizes $C_\alpha(T)$. This is done by weakest link pruning, which successively collapse the internal node that produces the smallest per-node increase in $\sum_m N_mQ_m(T)$. Lastly α is estimated by cross-validation such that $\hat{\alpha}$ minimizes the cross-validated misclassification rate. The final tree is then $T_{\hat{\alpha}}$.

5.2 Random Forests

Random forests are ensemble methods that provide a classifier from several classification trees. That is, a *committee* of trees each cast a vote for the predicted class. The essential idea in random forests is to build an extensive collection of de-correlated and unbiased trees from bootstrap samples, then average the trees in order to reduce the variance (Ch.15 by Hastie et al. (2001)).

Each tree in random forests is identically distributed. Let an average of K random variables have variance σ^2, and hence a total variance of $\frac{1}{K}\sigma^2$. For variables that are identically distributed with positive pairwise correlation ρ, the variance of the average can be expressed as

$$\rho\sigma^2+\frac{1-\rho}{K}\sigma^2.$$

Notice that as K increases, the second term decreases, while the first term remains constant. In order to improve the variance reduction, the idea of random forests is to reduce the correlation between the trees without a large increase in the variance. The solution

to this is to make a random selection of the input variables in the tree-growing process. That is, before each split in a tree, select $m \leq p$ of the input variables at random as candidates for splitting. Furthermore, a classification random forest model obtains a class vote from each tree and classifies using a majority vote.

Some tunable hyperparameters for random forests are the number of predictors as candidates for the splits, the minimum node size, and the number of trees to fit. A popular choice for the number of predictors as candidates for the splits denoted m is $\sqrt{p}$. The minimum node size is often set to one for classification. The number of trees has to be sufficiently large, but a larger number is more computationally expensive. Nonetheless, these parameters depend on the data and are often treated as tuning hyperparameters.

Algorithm 1: Random Forest Algorithm for Classification by Hastie et al. (2001)

Input:

- A training set
- Tuning parameters:
  - number of predictors as candidates for split
  - minimum node size
  - number of trees

for $k = 1$ **to** K **do**

1. Draw a bootstrap sample from the training set.
2. Grow a random forest tree $f_k(\boldsymbol{x})$ to the bootstrapped data by recursively repeating the following steps for each terminal node of the tree. Stop when a minimum node size n_{min} is reached.
   (a) From the p variables, select m variables at random.
   (b) Select the best variable among the m variables.
   (c) Split the node based on that variable.

end

Output: Ensemble of trees $\{f_k(\boldsymbol{x})\}_1^K$.

Classification: Let $\hat{f}_k(\boldsymbol{x})$ be the predicted class from the kth random forest tree. Then $\hat{f}^K(\boldsymbol{x})$ is the majority vote from $\{\hat{f}_k(\boldsymbol{x})\}_1^K$.

Variable importance

In order to measure how important a predictor is for the predictions of a random forest model, we can look at the variable importance. Variable importance is a measure that calculates the total decrease in node impurities from splitting on the variable, averaged over all trees. For classification, the node impurity is measured by the Gini index. A predictor is considered important if it has a high mean decrease in Gini.

5.3 Extreme Gradient Boosting

The algorithm for extreme gradient boosting (XGBoost) was first implemented by Chen and Guestrin (2016). XGBoost is an implementation of gradient boosted decision trees, which is designed for speed and performance (Lunde et al., 2020).

Similarly to random forests, boosting is an ensemble method that provides a classifier from several classification trees. However, the ensemble in boosting is done by repeatedly growing shallow trees to the residuals and building an additive model. More specifically, we first fit a model from the training data and then create a second model that attempts to correct the errors from the first model. Adding more models to correct the errors is repeated until the training error is sufficiently low or until a maximum number of models are added.

As the name suggests, extreme gradient boosting is a special case of boosting. Let f be an ensemble model with classification trees $f_k(\boldsymbol{x})$ as ensemble members. Furthermore, the loss is a function that measures the difference between a prediction $\hat{y}_i = f(\mathbf{x}_i)$ and its target y_i. We determine $f(\boldsymbol{x})$ by minimizing the expected loss function. Minimizing this can be viewed as a numerical optimization given as

$$\hat{f} = \arg\min_f E_{\boldsymbol{x},y}\big[l(y, f(\boldsymbol{x}))\big].$$

Now assume that $l(\cdot,\cdot)$ is both differentiable and convex. A prediction from f can be expressed as follows

$$\hat{y}_i = f^{(K)}(\boldsymbol{x}_i) = \sum_{k=1}^{K} f_k(\boldsymbol{x}_i),$$

where $f_k(\boldsymbol{x}_i) = w_{q_k(\boldsymbol{x}_i),k}$. Denote $\mathcal{L}_k$ to be the set of leaf nodes and M_k to be the number of leaf nodes in the k'th tree. Then $q_k : \mathbb{R}^m \to \mathcal{L}_k$ is the feature mapping of the k'th tree that assigns every feature vector to a unique leaf node. Moreover, $\boldsymbol{w}_k = \{w_{m,k}, m \in \mathcal{L}_k\} \in \mathbb{R}^{M_k}$ is the vector of predictions associated with each leaf node.

Now suppose a model $f^{(k-1)}$ with $k-1$ trees has been selected. We add another tree to improve the prediction, which allows the expectation to be rewritten as

$$E_{\boldsymbol{x},y}\Big[l\big(y, f^{(k)}(\boldsymbol{x})\big)\Big] = E_{\boldsymbol{x},y}\Big[l\big(y, f^{(k-1)}(\boldsymbol{x}) + f_k(\boldsymbol{x})\big)\Big]. \tag{5.1}$$

This should be minimized with respect to q_k and $\boldsymbol{w}_k$ associated with the model f_k. Next we perform a second order Taylor expansion around $\hat{y} = f^{(k-1)}(\boldsymbol{x})$. This can be expressed as

$$\hat{l}(y, \hat{y} + f_k(\boldsymbol{x})) = l(y, \hat{y}) + g(y, \hat{y}) f_k(\boldsymbol{x}) + \frac{1}{2} h(y, \hat{y}) f_k^2(\boldsymbol{x}),$$

where $g(y, \hat{y}) = \frac{\partial}{\partial \hat{y}} l(y, \hat{y})$ and $h(y, \hat{y}) = \frac{\partial^2}{\partial^2 \hat{y}} l(y, \hat{y})$.

Since the joint distribution of $(\boldsymbol{x}, y)$ is unknown, Equation (5.1) can be approximated using the Taylor expansion as follows

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^{n} l\big(y_i, \hat{y}_i^{(k-1)} + f_k(\boldsymbol{x}_i)\big) &\approx \frac{1}{n}\sum_{i=1}^{n}\left[l\big(y_i, \hat{y}_i^{(k-1)}\big) + g_{ik} f_k(\boldsymbol{x}_i) + \frac{1}{2} h_{ik} f_k(\boldsymbol{x}_i)^2\right] \\ &= \frac{1}{n}\sum_{i=1}^{n} l\big(y_i, \hat{y}_i^{(k-1)}\big) + \frac{1}{n}\sum_{m\in\mathcal{L}_k}\left[\sum_{i\in I_{mk}} g_{ik} w_{mk} + \frac{1}{2} h_{ik} w_{mk}^2\right] \\ &=: \ell_k(q_k, \boldsymbol{w}_k).\end{aligned}$$

Here

$$g_{ik} = g(y_i, f^{(k-1)}(\boldsymbol{x}_i)) \text{ and } h_{ik} = h(y_i, f^{(k-1)}(\boldsymbol{x}_i)). \tag{5.2}$$

Furthermore, I_{mk} is the instance set of leaf $m : I_{mk} = \{i : q_k(\boldsymbol{x}_i) = m\}$. Thus, $\ell_k(q_k, \boldsymbol{w}_k)$ is the training loss approximation of Equation (5.1), which we optimize by using k'th boosting iteration.

When we have a feature mapping q_k we can find the weight estimates $\hat{\boldsymbol{w}}_k$ minimizing $\boldsymbol{w}_k \to \ell_k(q_k, \boldsymbol{w}_k)$, which are given by

$$\hat{w}_{mk} = -\frac{G_{mk}}{H_{mk}}, \text{ where } G_{mk} = \sum_{i\in I_{mk}} g_{ik}, H_{mk} = \sum_{i\in I_{mk}} h_{ik}. \tag{5.3}$$

By using these weights, we can further improve the training loss such that

$$\ell_k(q_k, \hat{\boldsymbol{w}}) - \frac{1}{n}\sum_{i=1}^{n} l(y_i, \hat{y}_i^{(k-1)}) = \frac{1}{2n}\sum_{m=1}^{M_k} \frac{G_{mk}^2}{H_{tk}}. \tag{5.4}$$

When we have explicit expression for Equation (5.3) and (5.4) we can compare a large number of candidate feature maps q_k. However, to consider every possible tree structure is too computationally expensive. Instead, it is common to do recursive binary splitting greedily, which can be performed by doing to following

1. Calculate a constant prediction for all features: $\hat{w} = -\frac{\sum_{i=1}^{n} g_{ik}}{\sum_{i=1}^{n} h_{ik}}$.

2. Choose a leaf node m. For each feature j, calculate the training loss reduction

$$R_m(j, t_j) = \frac{1}{2n}\left[\frac{\big(\sum_{i\in I_L(j,t_j)} g_{ik}\big)^2}{\sum_{i\in I_L(j,t_j)} h_{ik}} + \frac{\big(\sum_{i\in I_R(j,t_j)} g_{ik}\big)^2}{\sum_{i\in I_R(j,t_j)} h_{ik}} - \frac{\big(\sum_{i\in I_L(j,t_j)} g_{ik}\big)^2}{\sum_{i\in I_L(j,t_j)} h_{ik}}\right],$$

   where t_j are different split points, $I_L(j, t_j) = \{i \in I_{mk} : x_{ij} \le t_j\}$ and $I_R(j, t_j) = \{i \in I_{mk} : x_{ij} > t_j\}$. The next split from the old leaf m is chosen such that j and t_j maximize $R_m(j, t_j)$

3. Repeat step 2 iteratively until a tree complexity threshold is reached.

The tree complexity in step 3 can be maximum depth, maximum terminal nodes, minimum number of observations in a node, or a regularized objective. A common strategy to choose m is to build a large tree and then prune it using cost complexity pruning. Another parameter that needs to be determined for extreme gradient boosting is a learning rate $\delta \in (0, 1]$. The learning rate shrinks the effect of each new tree added, which improves the predictive power of the boosting. The algorithm for extreme gradient boosting is presented in Algorithm 2.

Algorithm 2: Extreme Gradient Boosting Algorithm by Chen and Guestrin (2016)

Input:

- A training set $\{(x_i, y_i)\}_{i=1}^n$
- A Differentiable loss $l(\cdot, \cdot)$
- Tuning parameters:
  - Number of trees
  - Maximum tree depth
  - Learning rate (shrinkage)
  - Minimum loss reduction
  - Column sampling
  - Minimum sum of instance weight
  - Row sampling

Initialize $f^{(0)}(\boldsymbol{x}) = \arg\min_\eta \sum_{i=1}^n l(y_i, \eta)$
for $k = 1$ **to** K **do**

1. Compute derivatives from Equation (5.2)
2. Determine q_k by iteratively selecting the binary split that maximizes Equation (2) until a tree complexity criterion is reached
3. Determine leaf weights from Equation (5.3) given q_k
4. Use the learning rate to scale the tree $f_k(\boldsymbol{x}) = \delta w_{q_k(\boldsymbol{x})}$
5. Update $f^{(k)}(\boldsymbol{x}) = f^{(k-1)}(\boldsymbol{x}) + f_k(\boldsymbol{x})$

end
Output: Model $f^{(K)}$

Some hyperparameters for extreme gradient boosting are the number of trees, maximum tree depth, learning rate, minimum loss reduction, column sampling, minimum sum of instance weight, and row sampling. All of these parameters depend on the data and are often treated as tuning hyperparameters.

Variable importance

To measure how important a predictor is for the predictions of an xgboost model, we can use measures denoted *gain* or $frequency$. Gain is the relative contribution of the corresponding predictor to the model, calculated by taking each predictor's contribution for each classification tree in the model. More specifically, when making a split on a new predictor on a certain branch in the classification tree, each new branch is more accurate than the previous branch. Gain then represents the fractional contribution of each predictor to the model, based on the total gain of the splits performed on this predictor. The frequency represents the relative number of times a predictor has been used in all the model trees. For both measures, a higher percentage means a more important predictive variable.

5.4 Partial Dependence Plot

When interpreting the tree ensemble models, it is helpful to investigate the functional relationship between the variables and the predictions. A partial dependence (PD) plot can display this with a small number of variables (Ch. 10 by Hastie et al. (2001)).

Consider a subvector $\boldsymbol{x}_S$ of $l < p$ of the predictor variables. Let $\boldsymbol{x}_C$ be the vector of predictors that is not in $\boldsymbol{x}_S$. In principle the general function $f(\boldsymbol{x})$ will depend on all of the input variables, $f(\boldsymbol{x}) = f(\boldsymbol{x}_S, \boldsymbol{x}_C)$. The partial dependence of $f(\boldsymbol{x})$ on $\boldsymbol{x}_S$ is given as

$$f_S(\boldsymbol{x}_S) = E_{X_C} f(\boldsymbol{x}_S, \boldsymbol{x}_C) = \int f(\boldsymbol{x}_S, \boldsymbol{x}_C) g(\boldsymbol{x}_C) d\boldsymbol{x}_C.$$

This can give a helpful description of the effect of the chosen subset on $f(\boldsymbol{x})$ when, for instance, the variables in $\boldsymbol{x}_S$ do not have strong interactions with those in $\boldsymbol{x}_C$. These partial dependence functions can be used to interpret the results of any black-box learning method, such as random forests and extreme gradient boosting. They can be estimated by

$$\overline{f}_S(\boldsymbol{x}_S) = \frac{1}{N} \sum_{i=1}^{N} f(\boldsymbol{x}_S, x_{iC}),$$

where $\{x_{1C}, x_{2C}, ..., x_{NC}\}$ are the values of $\boldsymbol{x}_C$ that occur in the training data.

In other words, the partial dependence represent the effect of $\boldsymbol{x}_S$ on $f(\boldsymbol{x})$ after accounting for the effects of the other variables $\boldsymbol{x}_C$ on $f(\boldsymbol{x})$.

5.5 Accumulated Local Effects Plot

PD plots build on the assumption of independent predictor variables. However, if the variables are correlated, the PD plot can be misleading. This is because the computation of a partial dependence function for a variable correlated with other variables involves averaging predictions of artificial data instances that are unlikely. This again can give bias in the estimated predictor variable. A solution to this is to use Accumulated Local

Effects (ALE) plots instead (Apley and Zhu, 2020).

ALE plots are an alternative way of visualizing variable effects which do not require unreliable extrapolation with correlated variables. The main effect of variable j at x, in a multivariate case, is then given as

$$\begin{aligned} f_{j,ALE}(x_j) &= \int_{x_{\min,j}}^{x_j} \int p(\boldsymbol{x}_{\setminus j}|z_j) \frac{\delta f(z_j, \boldsymbol{x}_{\setminus j})}{\delta z_1} d\boldsymbol{x}_{\setminus j} dz_1 - \text{constant} \\ &= g_{j,ALE}(x_j) - \int p(z_j) g_{j,ALE}(z_j) dz_j. \end{aligned}$$

Here $\frac{\delta f(z_j, \boldsymbol{x}_{\setminus j})}{\delta z_1}$ is the local effect of x_j on f at $\boldsymbol{x}$. Moreover, $x_{\min,j}$ is a value just below $\min\{x_{i,j}; i = 1, ..., n\}$.

An approximation of the uncentered ALE can be expressed as

$$\hat{g}_{j,ALE}(x) = \sum_{k=1}^{k_{j(x)}} \frac{1}{n_{j(k)}} \sum_{\{i: x_{i,j} \in N_{j(k)}\}} \Big[f(z_{k,j}, \boldsymbol{x}_{i\setminus j}) - f(z_{k-1,j}, \boldsymbol{x}_{i\setminus j}) \Big],$$

where $n_{j(k)}$ is the number of training observations that falls into the kth interval $N_{j(k)}$. Furthermore, $k_{j(x)}$ is the index of the interval where x falls and the sample range of $x_{i,ji} = 1, ..., n$ is split into K intervals with split points $z_{0,j}, ..., z_{K,j}$.

The centered approximation is then obtained by subtracting the estimate of $\mathbb{E}[g_{j,ALE}(x_j)]$ from the uncentered approximation, that is

$$\hat{f}_{j,ALE}(x) = \hat{g}_{j,ALE}(x) - \frac{1}{n} \sum_{i=1}^{n} \hat{g}_{j,ALE}(x_{i,j}) = \hat{g}_{j,ALE}(x) - \frac{1}{n} \sum_{k=1}^{K} n_{j(k)} \hat{g}_{j,ALE}(z_{k,j}).$$

An interpretation of this is that $\hat{f}_{j,ALE}(x)$ is the main effect of the variable j at value x, compared to the average prediction of the data.

5.6 Cross-Validation

Both random forests and extreme gradient boosting have tunable hyperparameters. A popular approach to select the optimal hyperparameters is to perform cross-validation (CV). The approach is a method where the misclassification rate is estimated by holding out a subset of the training observations from the fitting process and then evaluate the model to those held out observations (Ch. 5 James et al. (2014)).

A special case of CV is k-fold CV, which involves randomly dividing the set of observations into k equally sized folds. Then the first fold is treated as a test set, while the remaining $k - 1$ folds are used as a training set to fit the model. Typical values for the number of folds are $k = 5$ or $k = 10$. For classification, the misclassification rate is computed on the first fold to evaluate the model. The approach is repeated k times, where each fold is

treated as a test set once. This results in k estimates of the misclassification rate denoted Err. Then the k-fold CV estimate is the average of the estimates for each fold, given as

$$CV_{(k)} = \frac{1}{k} \sum_{i=1}^{k} Err_i.$$

Thus, CV can give an estimate of how well a statistical model will perform on independent data. CV can therefore be used to evaluate the model fit for a range of hyperparameters. The optimal set of hyperparameters that minimizes $CV_{(k)}$ can then be chosen, as this model is expected to have the best performance on an independent data set.

Chapter 6

Modeling Genetic Interaction Effects

In our analysis, we are interested in measuring the interaction effects between physical activity (PA) and the genetic predisposition of CHD. In this chapter, we will present our different approaches. Two different types of models will be used, namely logistic regression and tree ensembles.

Logistic regression requires that we specify the functional relationship between the covariates and the outcome, in contrast to tree ensemble models. If we fit a tree ensemble to our data, we can next estimate the functional relationship from partial dependence (PD) plots or accumulated local effects (ALE) plots. If the tree ensemble model performs well on our data, we can use the information to fit a logistic regression model. We will first look at the functional relationship of the genetic main effects. There are several possible genetic models, and we will therefore begin the chapter by describing the different alternatives and how we determine which to choose for each SNP in our analysis. Then we will use this information to investigate the interaction effects.

6.1 Genetic Main Effects

The SNPs are the genetic covariates of interest. A person may have 0, 1, or 2 copies of a risk allele of a SNP, which increases the risk of developing diseases. However, the number of risk alleles affects the risk of developing diseases differently depending on the SNP. We will now look at four different genetic models for SNPs: recessive, dominant, additive, or codominant.

If the risk allele is recessive, then the risk allele will only affect the outcome if two risk alleles are present. In contrast, if the risk allele is dominant, then one copy of the risk allele will be enough to affect the outcome. Thus for a recessive or dominant case, the SNP is a binary covariate. An allele is additive if the effect of having two risk alleles is twice as large as the effect of having one risk allele. Finally, the genetic model can also be codominant, which means that 0, 1, or 2 copies of the risk allele affect the outcome in three different ways. Thus, when an allele is additive or codominant, the coded genetic covariate has three levels. An illustration of the four different genetic models can be found in Figure 6.1.

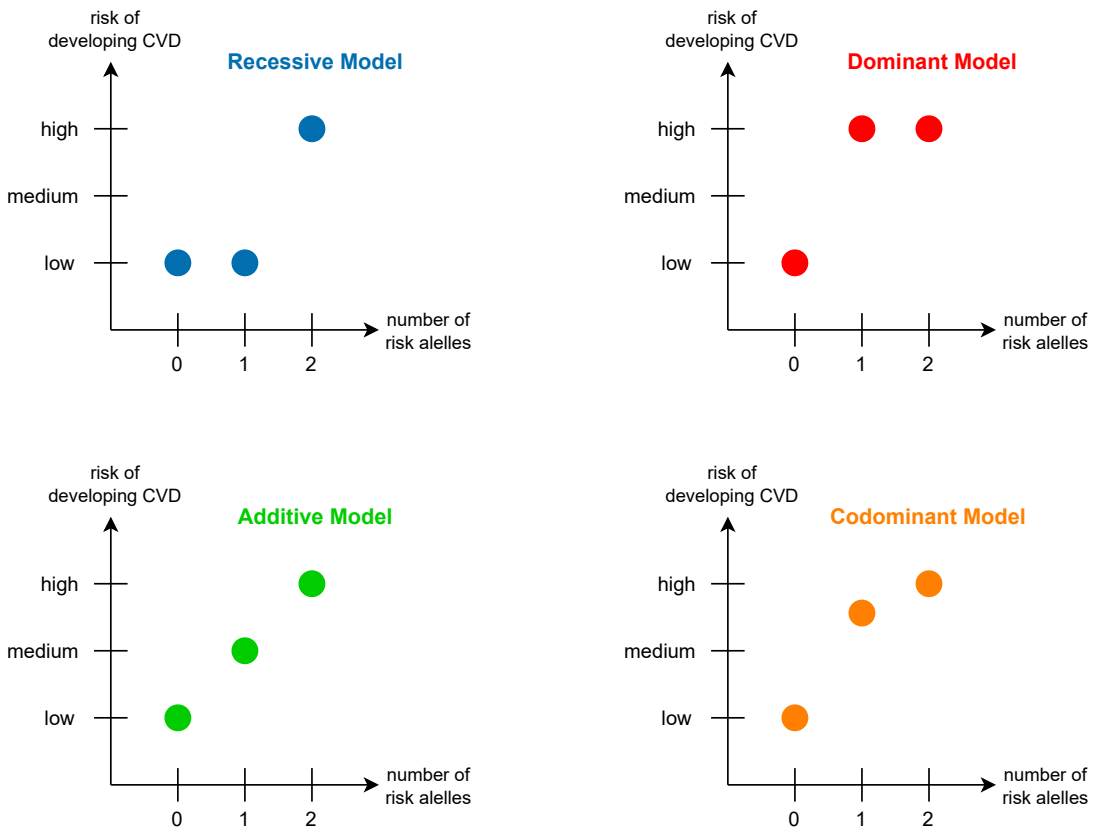


Figure 6.1: The plots represent the behavior of a recessive, dominant, additive, and codominant genetic model as a function of the number of risk alleles for genotypes 0, 1, and 2. The codominant model can be any function of the number of risk alleles. The risk is given on the log-odds scale, as this is the functional relationship the logistic regression requires that we specify.

In order to measure the effects of the SNPs for logistic regression, a genetic model must be chosen for each SNP. According to the literature in statistical genomics, some SNPs have a known relationship with CHD. However, in general the relationship between most SNPs and CHD is unknown. The codominant model makes no assumptions of the genetic relationship between the number of risk alleles. Hence, we can use a codominant coding for every SNP in a tree ensemble model and then plot the functional relationships given by the fit in PD plots. That is, we make PD plots with one variable, explained in Section 5.4. This is feasible since the tree ensemble models require no specification on the functional relationship between the covariates and the outcome. By investigating the PD plots, we can choose which genetic model is the best fit for each SNP. Hence, for logistic regression, we can code the genetic covariates according to the tree ensemble fit.

6.2 Genetic Interaction Effects for Logistic Regression

For logistic regression, we will use an approach where the main and interaction effects are modeled and a stratified approach. Both approaches will be described below.

6.2.1 Modeling the Main and Interaction Effects

The interaction effect can be modeled as an addition to the main effects. Then we can investigate whether the model with the main effects and the interaction effect is significantly different from the model with only the main effects. This method will have different interaction terms depending on the chosen genetic model.

If the SNP is coded as recessive, dominant, or additive, we fit a model for each SNP with the main effects and the interaction effect as follows

$$\beta_0 + \beta_{PA} + \beta_{snp} + \beta_{PA \cdot snp}.$$

Here, the effect of the other covariates are measured in the intercept. Furthermore, the main effects are measured in the terms $\beta_{PA} + \beta_{snp}$ and the interaction effect in $\beta_{PA \cdot snp}$. In this case, we only have one interaction parameter which needs to be estimated.

If the SNP is measured as codominant, we fit a model for each SNP as follows

$$\beta_0 + \beta_{PA} + \beta_{snp(1)} + \beta_{snp(2)} + \beta_{PA \cdot snp(1)} + \beta_{PA \cdot snp(2)}.$$

The effect of the other covariates are measured in the intercept, and $snp_{(\cdot)}$ indicates the number of risk alleles. Then the main effects are measured in the terms $\beta_{PA} + \beta_{snp(1)} + \beta_{snp(2)}$ and the interaction effect in $\beta_{PA \cdot snp(1)} + \beta_{PA \cdot snp(2)}$. Here the interaction effect is measured in two parameters.

In the approach of measuring the interaction effect as an addition to the main effects, we multiply the effects of PA and the SNPs. This results in only measuring the interaction effect when PA and the number of risk alleles both are different from zero.

A likelihood ratio test with the model with both the main effects and the interaction effects and a model with just the main effects can be used to test if an interaction effect should be present. That is, we investigate whether adding the interaction term improves the performance of the model. By looking at the test statistic, we can evaluate whether the interaction term between the genetic covariates and PA is significant.

6.2.2 Stratified Analysis

Another approach to investigate the interaction effects is by stratifying the data set. The idea behind stratified analysis is to stratify the data set on one of the predictors in the interaction effect, which is physical activity or the genetic covariates in our case. We then fit a logistic regression model for each data set and compare the estimated covariates of the predictor in the interaction effect that we do not stratify on. If the estimated covariates are different for each fit, this can indicate an interaction effect.

In order to do the stratified analysis, we have to decide whether to stratify on PA or the SNPs. This choice is based on how we model the main effects, which are PA and the SNPs. We can choose which genetic model to use for each SNP from the information described in the last section. However, we also need to specify the functional relationships

between PA and CVD. Two options for measuring PA that we will consider are treating the covariate as dichotomous or continuous. How we measure the interaction effects in the stratified analysis will depend on this choice.

First, consider the case where we let PA be a dichotomous covariate. Denote the PA covariate to be 0 if a participant is inactive and 1 if a participant is active, based on some criteria. In a dichotomous case, we can stratify the data set based on PA and model the genetic effects in each data set. In order to analyze the interaction effects, it is helpful to compare the SNP covariates and their confidence level in each data set. Denote β^0 and β^1 the coefficients in the data set with PA being 0 and 1, respectively. What we are interested in testing is

$$H_0 : \beta^0_{snp} = \beta^1_{snp} \text{ vs. } H_1 : \beta^0_{snp} \neq \beta^1_{snp},$$

where β_{snp} are the coefficients for a genetic model for a SNP. If the estimated parameters are significantly different in each data set, it indicates that PA affects the SNP's effect on CHD. In other words, if the β_{snp} are different in the two data sets, it can indicate that the effect of the SNP is different for the two levels of PA. In practice, we will prefer to present a confidence interval for the $\beta^0_{snp} - \beta^1_{snp}$ difference instead of the hypothesis test, as this gives a better visualization of the effects. Then, if the confidence interval does not contain zero, this is equivalent to rejecting H_0.

If we prefer to measure PA as a continuous covariate, we cannot stratify the data sets in the manner explained above. A solution to this can be to stratify the data set on the genetic factors instead. This is feasible unless the genetic effect is additive since the genetic covariate will then be continuous. When stratifying on the genetic factor, we perform the stratification for each SNP, fit a logistic regression model for each data set and then compare the PA effect in those model fits. However, the genetic covariates can have two or three categorical levels, depending on the genetic model. If the genetic effect is measured as recessive or dominant, we will divide the data set into two separate data sets. When the genetic effect is codominant, we have to separate the data set into three separate data sets. Hence, how we stratify the data sets depend on the genetic model for the SNPs.

If the genetic covariate is dichotomous, which is in the recessive or dominant case, then we can follow a similar procedure for the stratification on PA and divide the data set in two. We stratify the data sets on whether the participants have an increased risk of developing CVD from a specific SNP or not and fit a logistic regression model on each data set. Then we can compare the PA covariates in each data set by investigating the confidence level for the $\beta^0_{PA} - \beta^1_{PA}$ difference. If the confidence interval does not contain zero, it can indicate an interaction between that SNP and PA. However, suppose the genetic covariate has three levels. In that case, we have to stratify the data set into three separate data sets, fit three different models and compare the three PA covariates to investigate whether there is an interaction effect. We can then use ANOVA or pairwise tests.

A disadvantage of the stratified approach is that we divide the data into smaller data sets. As more extensive data sets give more reliable results than smaller ones, this is a weakness of the stratified approach. This loss of power is especially the case when stratifying the data on a codominant SNP. On the other hand, an advantage of the stratifying approach is that there is no specification of the interaction term.

6.3 Interaction Effect for Tree Ensembles

PD plots with two predictors are used to visualize the interaction among those predictors. Thus, for the tree ensemble models, we will investigate the interaction effect by examining the PD plots with two variables, explained in Section 5.4.

That is, for each SNP, we make a PD plot with that SNP and PA, which yields a visualization of the interaction effect. Based on the plots, we can get more information on the interaction effect. However, it can be challenging to visualize whether there, in fact, is an interaction effect from only visualizing the plots. Therefore, we also perform an informal test on the significance of the interaction effect. That is, we test whether the interaction affects the log-odds of the predicted probability of developing CHD. In order to test this, we can extract the values used to plot the PD plot and create a surrogate data set. The variables are PA and the SNP, and the outcome is the predicted log-odds of developing CHD. Using this surrogate data, we can fit a linear regression with PA, the SNP, and the multiplication of PA and the SNP, which will be the interaction effect, as covariates. That is, we can fit a model for each SNP as follows

$$\beta_0 + \beta_{PA} + \beta_{SNP} + \beta_{PA \cdot SNP}.$$

Here, the SNP covariate is coded according to the genetic model that best fits each SNP. From the model, we can get information on the strength of the interaction term by looking at the coefficient estimate and the test statistics of $\beta_{PA \cdot SNP}$. It is important to mention that the surrogate data set does not have independent observations. This is an informal test to get an improved understanding of the PD plots with two predictors.

Chapter 7

Data Analysis

In this chapter, the results from our analysis are presented. First, we present the data set used. Next, we fit a random forest model and extreme gradient boosting (xgboost) model. With the xgboost model fit, we investigate the interaction effect by partial dependence (PD) plots. Lastly, we evaluate the functional relationship between the covariates and the outcome and use this to fit a logistic regression model. The interaction effects are then analyzed by fitting a logistic regression.

7.1 Descriptive Statistics

In the data set from HUNT3, we are only interested in the participants who had not already suffered from CVD when they participated in HUNT3. Hence, we exclude the participants who previously suffered from CVD. Moreover, we only include the covariates given in Table 7.1, which are chosen from a medical point of view. This process is explained in a flow diagram in Figure 7.1. A more detailed description of how we extract the covariates from HUNT can be found in Appendix A.1.

Covariate label	**Variable in HUNT**	**Description**	**Type**
id	PID_106474	Unique id for each participant	
sex	Sex	Biological sex (Female:0, Male:1)	binary
age	PartAg_NT3BLQ1	Age of participant	continuous
smoking	SmoStatNT3_recoded	Smoking status of the participant (1:Not smoking, 2:Smoking)	binary
bmi	BMI_NT3BLM	Body mass index	continuous
seChol	SeChol_NT3BLM	Serum Cholesterol	continuous
seHDLChol	SeHDLChol_NT3BLM	Serum High-density lipoprotein Cholesterol	continuous
bpSyst	BPSystMn23_NT3BLM	Mean systolic pressure measurement 2 and 3	continuous
kurtze	PA_H3_index_K	Kurtze score which measures self reported physical activity	continuous

Table 7.1: The environmental covariates extracted from HUNT and used in the analysis.

Logistic regression requires independent observations. However, we may have correlated observations due to participants being related or due to participants having the same environmental conditions. In order to correct for this, we add principal components as covariates in our data set. The principal components are found in a separate file from HUNT.

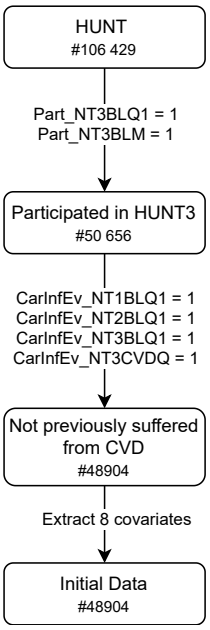


Figure 7.1: Flow diagram: from the HUNT data set to Initial Data. Here # denotes number of participants in sequence of data sets.

Next, we add the genetic covariates, which are the 50 SNPs given in Table A.2 in the Appendix. In our case, the covariates are coded such that the number of minor alleles is counted. A more detailed description of the genetic variables is in Section 2.2. In addition, a description of how we prepare the SNP files from HUNT can be found in Appendix A.2.

Lastly, we add the outcome to our data set, given in Table A.1 in the Appendix. We use CHD as the outcome variable in this project since this is the variable known to be associated with the 50 chosen SNPs. Participants with ICD diagnose code I21 or I22 are coded as CHD cases. The ICD codes are explained in Section 2.2. A description of how we extract the outcome from hospital data from Helse Nord-Trøndelag can be found in Appendix A.3.

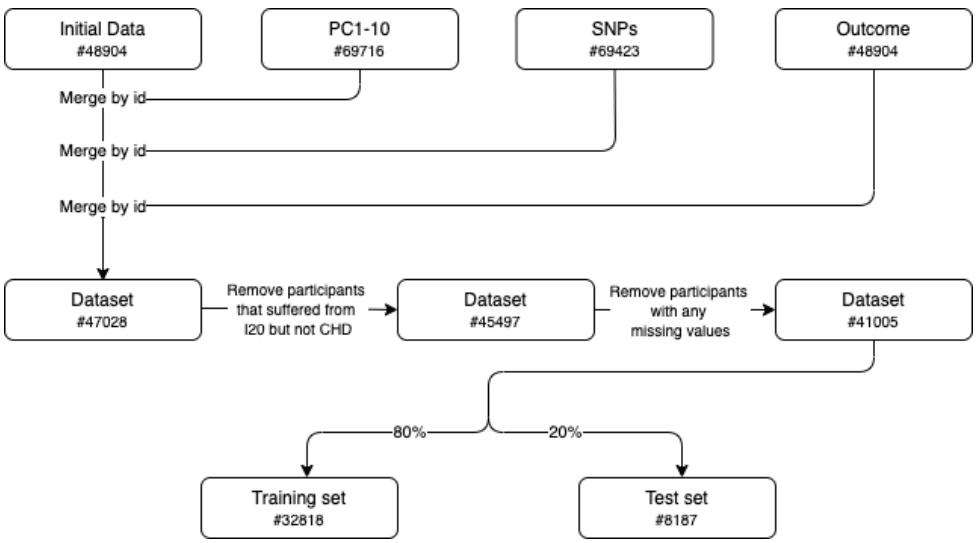


Figure 7.2: Flow diagram: from initial data to training and test set. Here # denotes number of participants in sequence of data sets.

Now we have all our variables of interest. However, before we have our final data set, we remove participants with the ICD code I20. The ICD code I20 represents chest pain, also called angina. Angina is not associated with the 50 chosen SNPs, so we remove those cases from our data set. We also remove participants with missing values for any of the variables. Finally, we subset the data set into a training and test set stratified by the outcome. As explained in Section 3.1, we will use the training set to fit the models and the test set to evaluate the models. The process of constructing the training and test data set is described in a flow diagram in Figure 7.2.

Summary statistics for the environmental covariates for the training and test set can be found in Figure 7.3. From the table, observe that the percentage of participants who suffered from CHD is 3.2%. We have a severely imbalanced data set, which means that the number of cases is significantly lower than the number of controls. Observe that CHD is evenly distributed in the training and test set since we stratify on CHD. Furthermore, the environmental covariates also have similar distributions in the two data sets, which is desirable since each data set should represent the data distribution in the whole data set.

	Overall (N=32818)
chd	
0	31766 (96.8%)
1	1052 (3.2%)
sex	
0	18069 (55.1%)
1	14749 (44.9%)
age	
Mean (SD)	51.5 (15.4)
Median [Min, Max]	51.7 [19.1, 101]
smoking	
1	27060 (82.5%)
2	5758 (17.5%)
bmi	
Mean (SD)	27.1 (4.38)
Median [Min, Max]	26.6 [12.1, 55.9]
seChol	
Mean (SD)	5.51 (1.10)
Median [Min, Max]	5.40 [1.70, 12.3]
seHDLChol	
Mean (SD)	1.35 (0.351)
Median [Min, Max]	1.30 [0.500, 4.00]
bpSyst	
Mean (SD)	130 (18.2)
Median [Min, Max]	128 [60.0, 260]
kurtze	
Mean (SD)	2.65 (2.67)
Median [Min, Max]	1.88 [0, 15.0]

(a) Training set

	Overall (N=8187)
chd	
0	7936 (96.9%)
1	251 (3.1%)
sex	
0	4495 (54.9%)
1	3692 (45.1%)
age	
Mean (SD)	51.6 (15.6)
Median [Min, Max]	51.7 [19.3, 96.7]
smoking	
1	6791 (82.9%)
2	1396 (17.1%)
bmi	
Mean (SD)	27.0 (4.35)
Median [Min, Max]	26.6 [15.0, 51.8]
seChol	
Mean (SD)	5.52 (1.09)
Median [Min, Max]	5.50 [2.00, 11.3]
seHDLChol	
Mean (SD)	1.35 (0.349)
Median [Min, Max]	1.30 [0.500, 3.40]
bpSyst	
Mean (SD)	130 (18.4)
Median [Min, Max]	128 [82.0, 234]
kurtze	
Mean (SD)	2.63 (2.59)
Median [Min, Max]	1.88 [0, 15.0]

(b) Test set

Figure 7.3: Summary statistics for covariates for the training and test set. See Table 7.1 for the description of each covariate label.

A visualization of the association of the continuous environmental covariates with the outcome in the training set is presented in boxplots in Figure 7.4. Observe that age is associated with CHD, where a participant with a higher age is more likely to get CHD. The same holds for BMI, serum cholesterol level, and systolic blood pressure, but with a weaker association. In contrast, the serum high-density lipoprotein cholesterol level has a weak association with CHD in the opposite direction. Lastly, and of importance, self-reported physical activity does not appear to have an association with CHD. Moreover, BMI, serum cholesterol level, serum high-density lipoprotein cholesterol level, and systolic blood pressure have many outliers, especially for the group of controls.

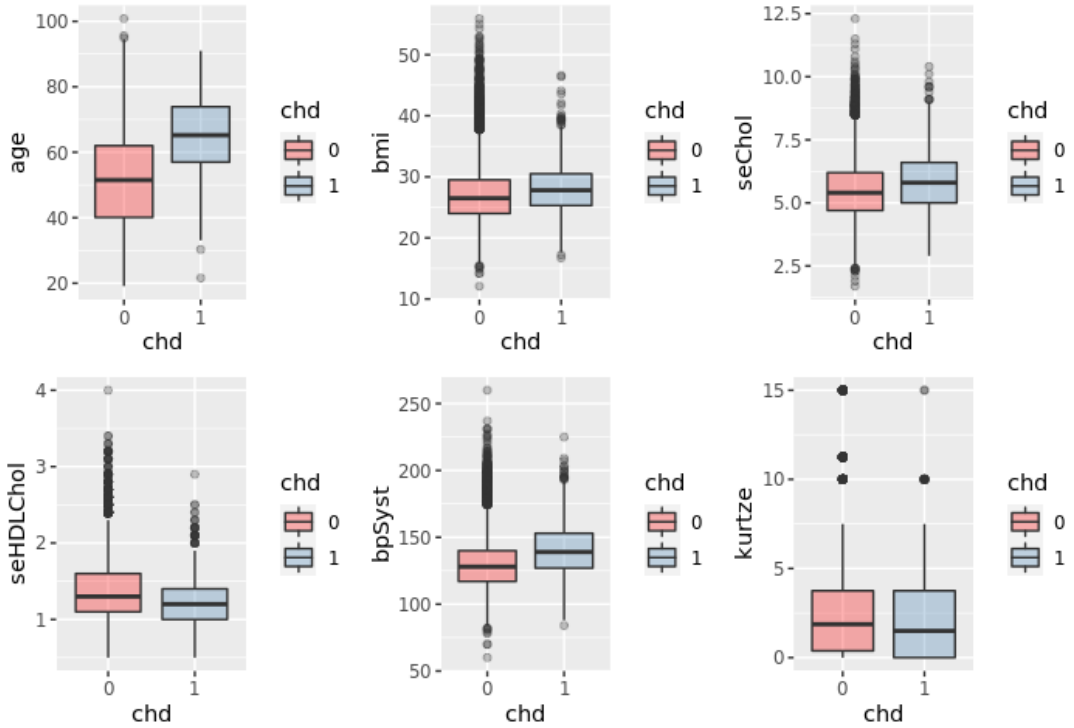


Figure 7.4: Boxplots of all continuous environmental covariates stratified by the outcome on the training set. See Table 7.1 for the description of each covariate label.

A visualization of the association of the categorical environmental covariates with the outcome in the training set is presented in barplots in Figure 7.5. From this figure, we can observe that there are slightly more females than males, around 55% females and 45% males. However, the percentage of males who suffered from CHD is higher than for females. Additionally, around 18% of the participants are smokers, and the percentage of participants who suffered from CHD does not look higher in the smoking group than the non-smoking group.

A Pearson correlation plot of all the environmental covariates can be found in Figure 7.6. Observe that physical activity is correlated with many of the variables. Consequently, some of the effects of being active are measured by the other covariates. This again can lead to an estimated smaller effect of the main effects of physical activity and the interaction effects between physical activity and the genetic factors.

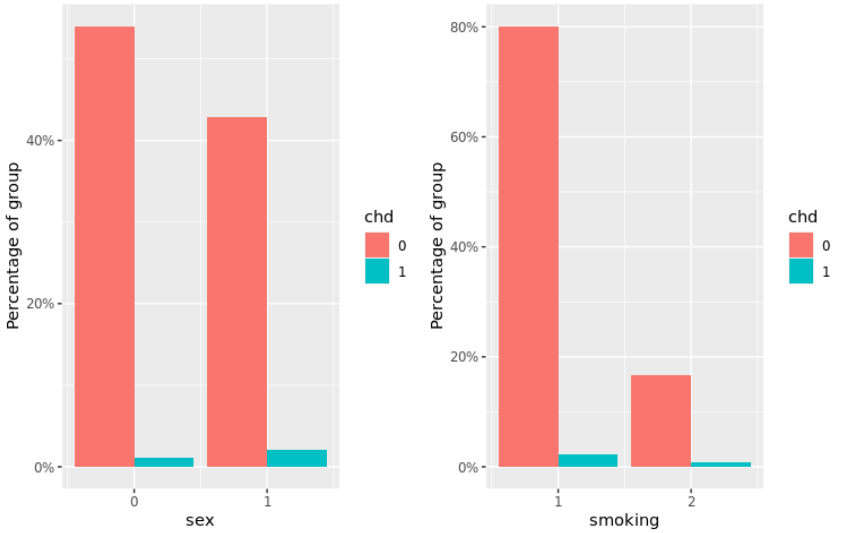


Figure 7.5: Percentage of CHD grouped by the categorical environmental covariates in training set. See Table 7.1 for the description of each covariate. Here Sex = 0 is females and Sex = 1 is males.

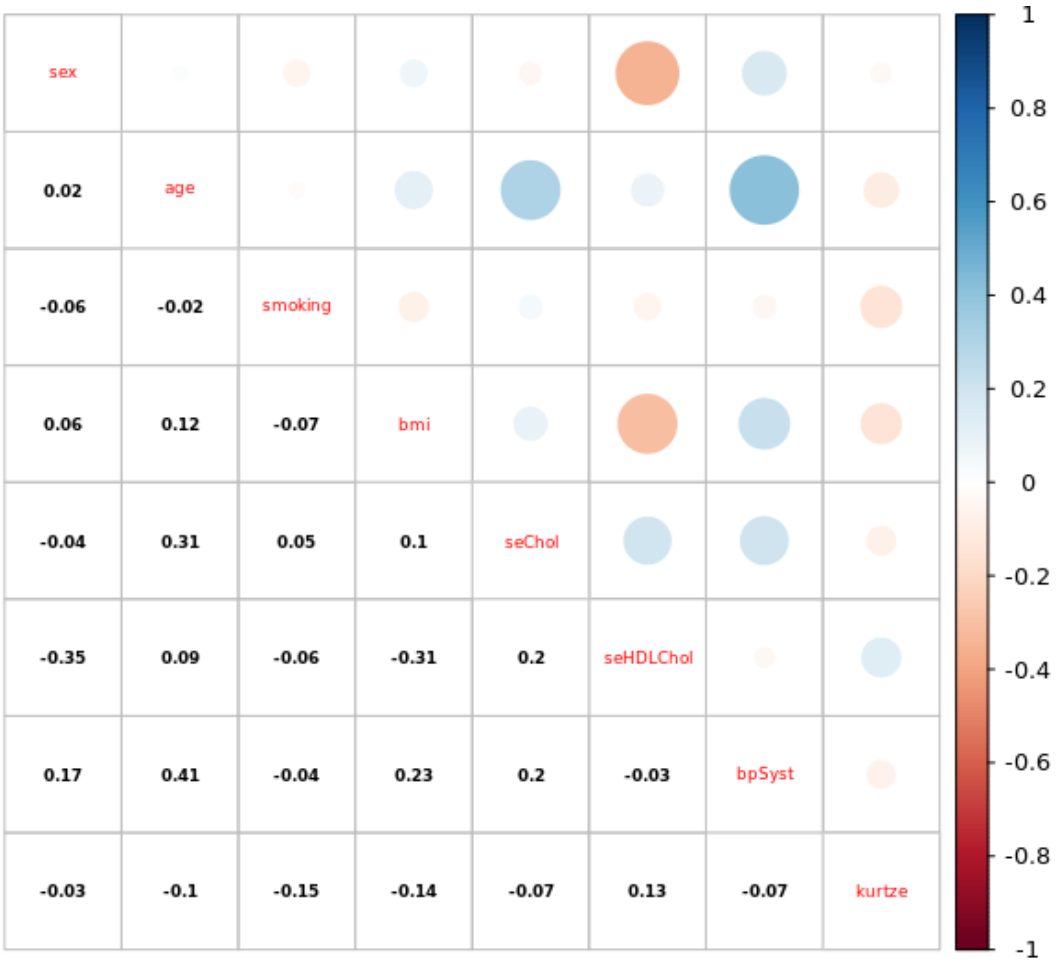


Figure 7.6: Pearson correlation plot of all environmental covariates in the training set. See Table 7.1 for the description of each covariate label.

A correlation plot between the ten first PCs is plotted in Figure 7.7. As mentioned above, the principal components are added to address population stratification and relatedness. We observe some outliers in the PC plots, which indicates that some participants are unlike the others genetically. We choose to use principal components 1 to 4 since this is a standard procedure in statistical genomics. Another visualization of the first four PCs plotted against each other can be seen in Figure 7.8. In this figure, we can also see that the cases and controls are evenly spread out in the population, which is our assumption for the models.

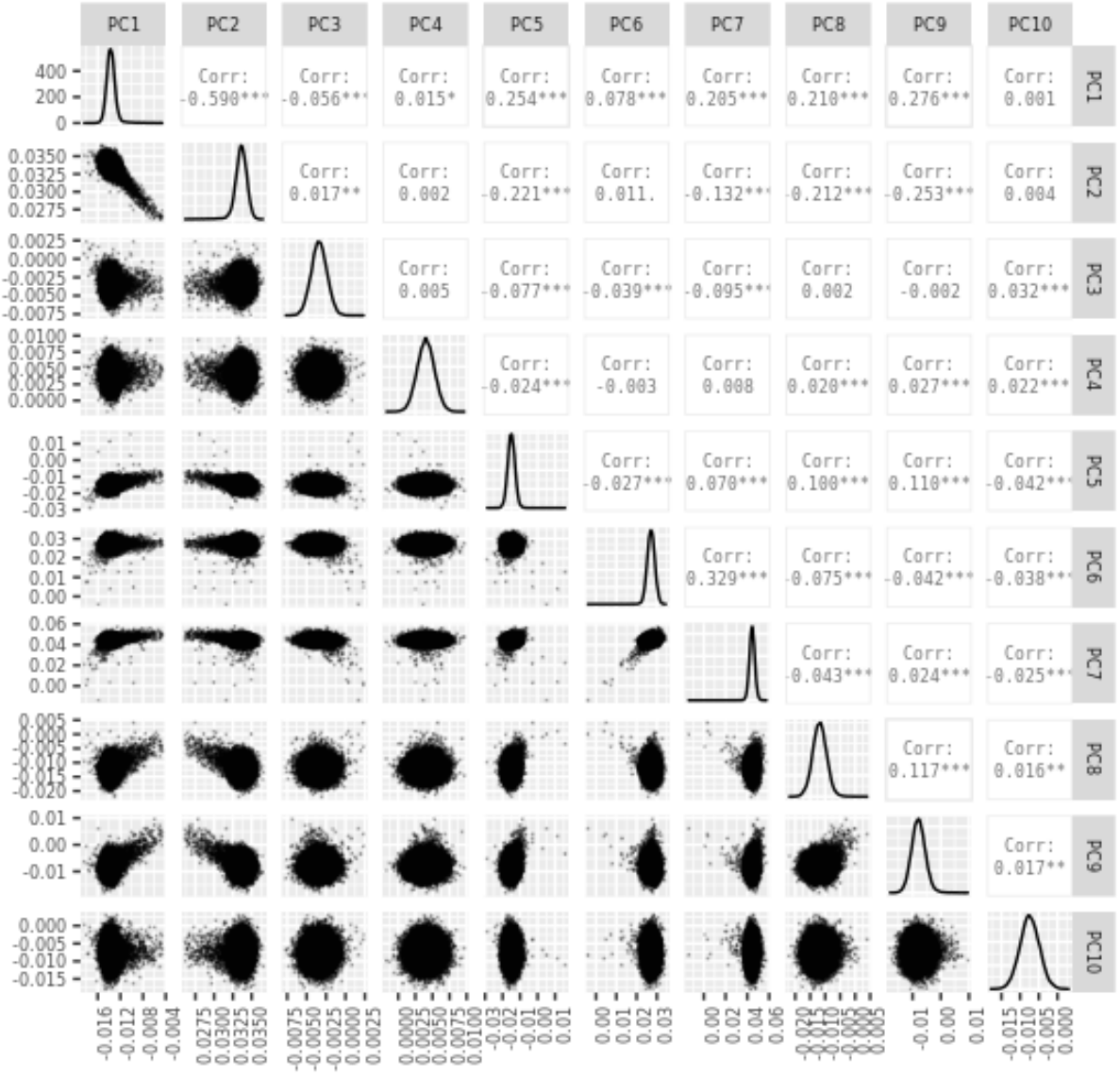


Figure 7.7: The ten first principal components plotted against each other.

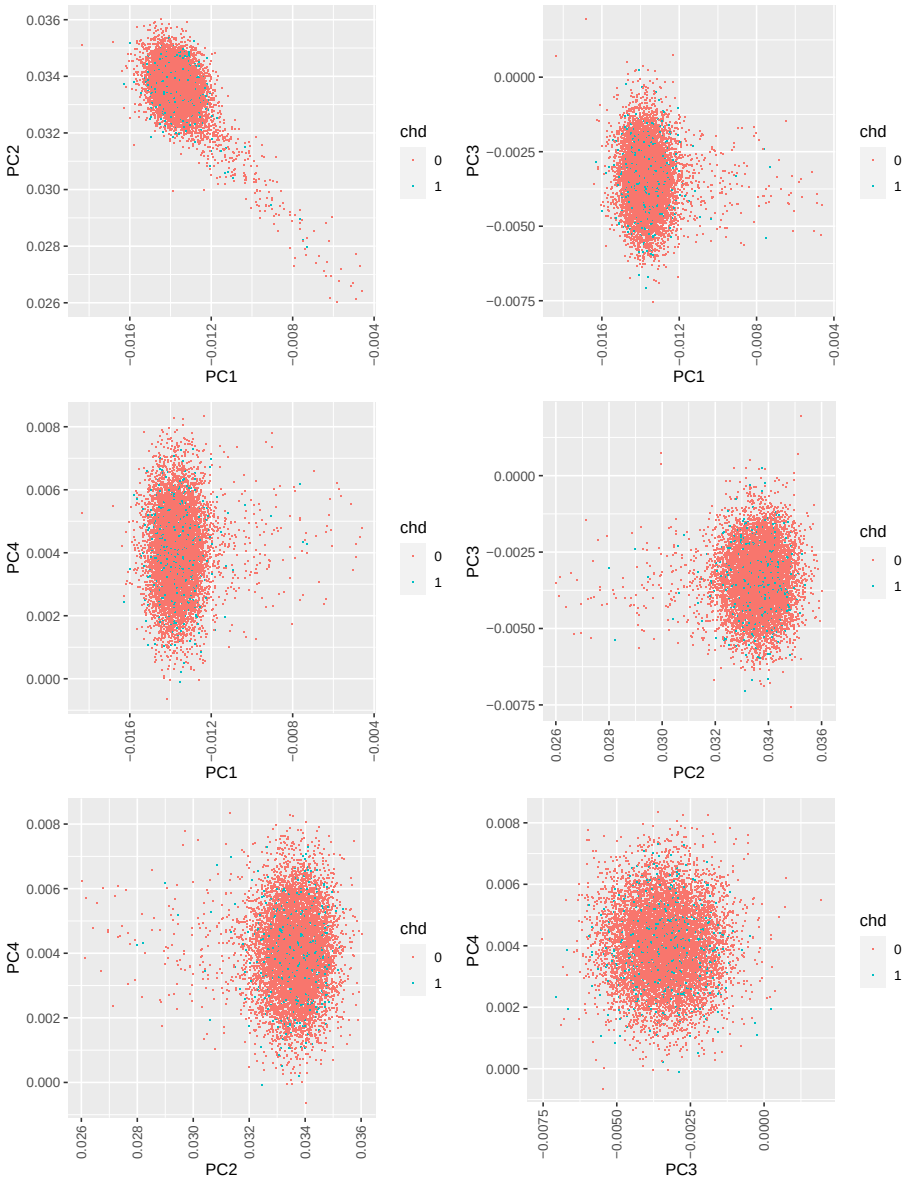


Figure 7.8: The four first principal components plotted against each other, with color based on outcome.

Lastly, we modify the physical activity score. The measure of physical activity is the Kurtze score, defined by Rangul et al. (2008) and explained in Section 2.2. Based on data exploration of physical activity and its functional relationship with CHD, we observe a trend where a higher physical activity decreases the risk of developing CHD. The trend is as hypothesized and an assumption of this analysis of the interaction effect between physical activity and genetic factors. However, for the small number of participants with a Kurtze score higher than five, the functional relationship is more volatile. A Kurtze score of around 5 is the minimum amount of recommended physical activity, based on national guidelines by Helsedirektoratet (2019). Around 11% of the participants have a Kurtze score higher than 5 in our data set. To avoid spurious due to possible noise or because Kurtze is self-reported, we choose to set a score of five for every participant with a Kurtze score higher than 5.

7.2 Random Forests

The first model we fit is the random forest model, which is fitted by using the randomForest package in R (Liaw and Wiener, 2002).

7.2.1 Hyperparameter Tuning

Before fitting the random forest model, some hyperparameters have to be tuned. They are tuned by performing a 5-fold CV and chosen based on the highest AUC score. Since the tuning is a computationally expensive process and the fact that there may be interactions between the hyperparameters, marginal tuning is not optimal. Hence, we first perform separate wide grid searches for each hyperparameter to find the range of optimal values. Then we perform tuning on all the hyperparameters simultaneously with a narrower grid to find the optimal values.

The most important hyperparameter to tune is the number of variables randomly sampled as candidates at each split in the trees, called *mtry* in the R package. For classification, the default is the square root of the number of variables. Our data set has 62 variables, so we tune this parameter with a wide grid, including $[5, 15]$. A plot of the results from the tuning can be found in Figure 7.9, where the optimal value for the parameter is 12 according to the AUC score from the 5-fold CV.

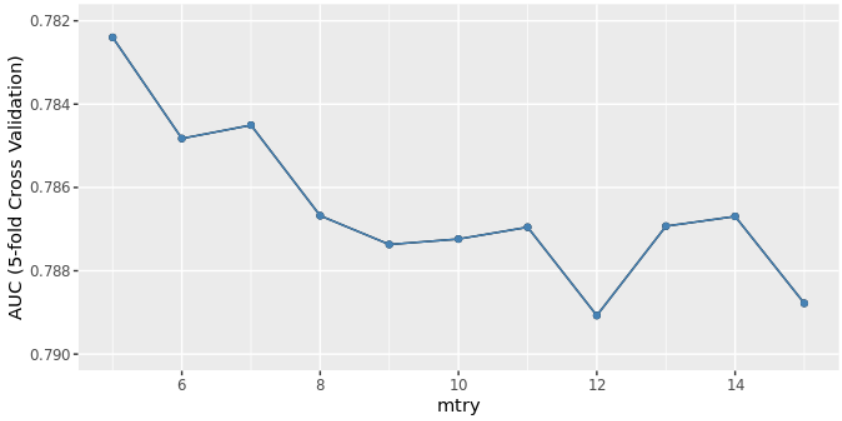


Figure 7.9: Hyperparameter tuning of mtry, the number of variables randomly sampled as candidates at each split in the trees. According to the tuning, the optimal value for mtry is 12 as this gave the highest AUC score.

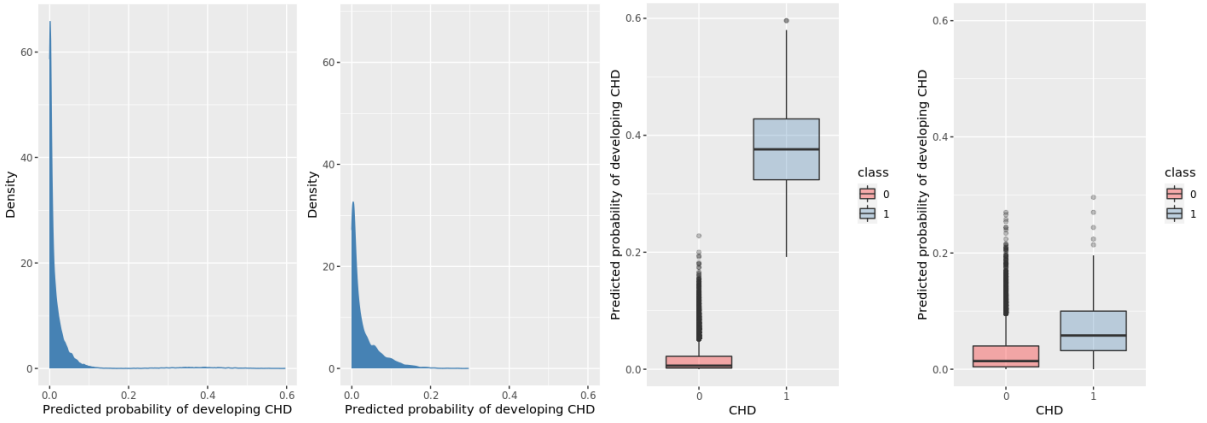
Another hyperparameter that can be tuned is the node size. Node size has a default value of 1 for classification. Using a 5-fold CV based on the AUC score with a wide grid including the values $[1, 15]$, the optimal value of the node size according to the tuning is 9. The last hyperparameter to tune is the number of trees to be fitted. This hyperparameter has to be sufficiently large, but a larger number is more computationally expensive. The default value is 500, and this is also what we use in our model.

Finally, to find the optimal combination of the values for the hyperparameters, we also perform a 5-fold CV with a bivariate grid search with both node size and the number of variables randomly sampled as candidates at each split in the trees. The ranges of optimal values found from the wide grid searches are $[6, 10]$ and $[10, 15]$ respectively, as

they contain the optimal values found from the separate marginal grid searches. Based on the AUC score, the final value for the node size is still 9, while the final value for the number of variables randomly sampled as candidates at each split in the trees is changed to 13. Our final values for our hyperparameters are then 13 for the number of variables randomly sampled as candidates at each split, 9 for the node size, and 500 for the number of trees.

7.2.2 Model Fit

Now that we have our hyperparameters, we fit the random forest model using the training set. In Figure 7.10 there are two density plots of the predicted probability of developing CHD, using the training and test set. From these plots, we see that the predicted probability of developing CHD is low. This is expected, as we observed a low number of cases compared to controls in our data set. Furthermore, the distribution of the predicted probability is larger when the prediction is made using the training set than the test set.



(a) Training set (b) Test set (c) Training set (d) Test set

Figure 7.10: Model fit on training and test set. The two left plots are density plots of the predicted probability of developing CHD. The two right plots are boxplots of the predicted probability of developing CHD stratified by CHD.

In Figure 7.10 there are also two boxplots of the predicted probability of developing CHD stratified by the outcome for the training and test set. Observe that the predicted probability is higher for the cases than the controls, indicating that the random forest model captures some data set patterns. Similarly to the density plots, the distribution of the predicted probability has a lower central trend on the test set than on the training set. Additionally, the difference of the average predicted probability for the two groups of CHD is large when using the training set, with almost 40 percentage points. On the other hand, for the prediction on the test set, the difference is less than 5 percentage points. This difference in the predictions in the data sets can indicate that the model may overfit the training data.

Another indication that the model may have overfitted the training data can be observed by investigating the AUC score for the two data sets. Specifically, the AUC score is 1.00 for the training set, meaning that the model classifies perfectly when predicting the training set. However, the AUC score is 0.78 for the test set. Another measure that highlights a possible overfit even more is the PR AUC score. The prediction using the training

set yields a PR AUC score of 1.00 while using the test set yields a much lower value of 0.10.

The random forest model also gives an overview of which predictors are considered the most important when predicting the outcome. A predictor is considered important if it has a high mean decrease in Gini. A variable importance plot of the 20 most important variables is presented in Figure 7.11. Age is considered the most important variable, which is expected. Moreover, the random forest model considers all the PCs to be important. Also, observe that physical activity is in the top 10 most important variables. Lastly, the genetic predictors are generally the least important variables, according to the random forest model.

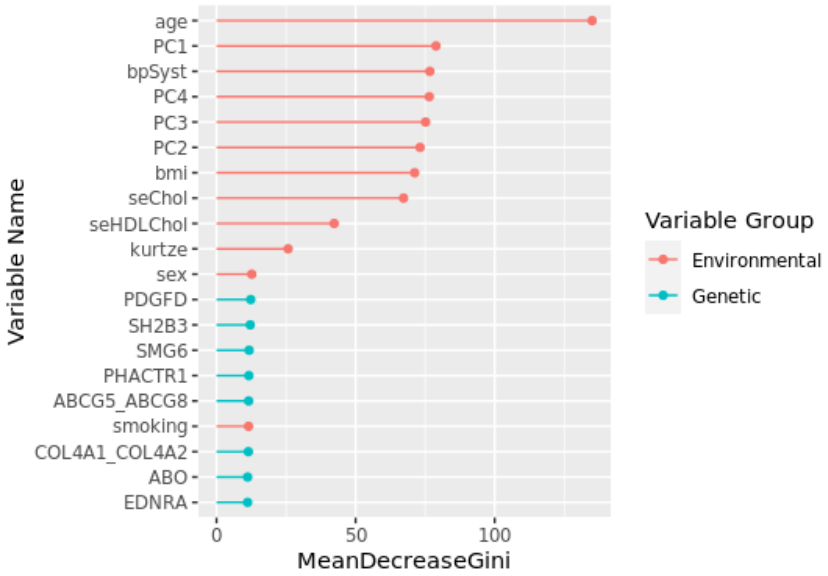


Figure 7.11: Variable importance plot of the 20 most important variables, based on the mean decrease in Gini. See Table 7.1 for the description of the environmental covariate labels and Table A.2 for the genetic covariate labels.

7.3 Extreme Gradient Boosting

The next model we fit is the extreme gradient boosting model, which is fitted using the xgboost package in R (Chen and Guestrin, 2016).

7.3.1 Hyperparameter Tuning

The hyperparameters we tune for extreme gradient boosting can be found in Table 7.2. The table also contains the tuning grids, the default values, and what the tuning suggests as the optimal value for each hyperparameter. The hyperparameters are tuned by performing a 5-fold CV based on the AUC score, using the caret package in R (Kuhn, 2008). Since the computational complexity of tuning all the hyperparameters at the same time is high, we break down the tuning into five consequential steps:

1. Shrinkage (learning rate), maximum tree depth and number of trees
2. Minimum sum of instance weight and number of trees

3. Column and row sampling and number of trees

4. Minimum loss reduction and number of trees

5. Reducing the learning rate and number of trees

We use a higher learning rate to tune the hyperparameters due to this being a computationally heavy process. That is, we fit a model for each configuration of the different hyperparameters with a higher learning rate and then use those hyperparameters to tune the final model with a lower learning rate grid. Furthermore, a smaller grid for the hyperparameter with the number of trees with values $[100, 1000]$ by 100 is included in every step. The optimal value for that hyperparameter is not selected until the final step. A plot of the final step is presented in Figure 7.12, while the plots for the previous steps can be found in Appendix B.1.1.

Hyperparameter in xgboost	**Description**	**Default value**	**Grid**	**Optimal value**
nrounds	Number of trees	100	[100, 1 000] by 100, [100, 5 000] by 100	900
max_depth	Maximum tree depth	6	[2, 3, 4, 5, 6]	2
eta	Shrinkage (learning rate)	0.3	[0.025, 0.05, 0.1, 0.3], [0.01, 0.015, 0.025, 0.05, 0.1]	0.015
gamma	Minimum loss reduction	0	[0, 0.05, 0.1, 0.5, 0.7, 0.9, 1.0]	0.1
colsample_bytree	Column sampling	1	[0.4, 0.6, 0.8, 1.0]	0.8
min_child_weight	Minimum sum of instance weight	1	[1, 2, 3]	3
subsample	Row sampling	1	[0.5, 0.75, 1.0]	0.5

Table 7.2: The hyperparameters used in tuning the xgboost model. The grid is the values used for tuning. The default values are the defaults for classification in the xgboost package. The optimal values are the results from hyperparameter tuning.

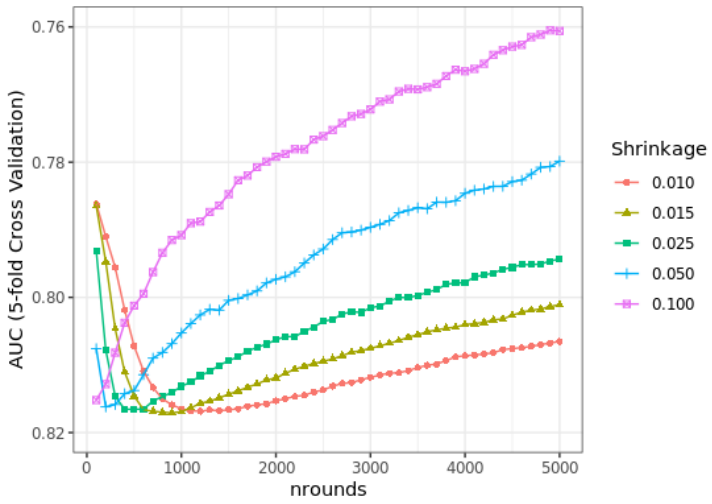
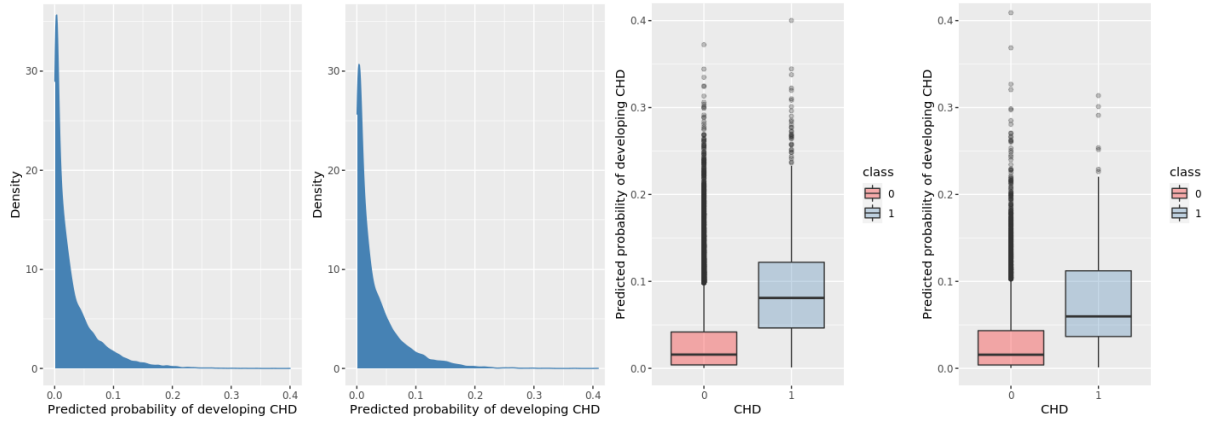


Figure 7.12: Step 5 of hyperparameter tuning, with shrinkage (*eta*) and number of trees (*nrounds*), suggesting $eta = 0.015$ and $nrounds = 900$ as optimal values based on AUC.

7.3.2 Model Fit

The xgboost model with the optimal values from the tuning is fitted using the training set. In Figure 7.13 there are two density plots of the predicted probability of developing CHD, using the training and test set. As for the random forest model fit, we observe that the predicted probability of developing CHD is low because of the observed low number of cases in our data set. However, the predicted probability density is similar when the prediction is performed using the training and test set.



(a) Training set (b) Test set (c) Training set (d) Test set

Figure 7.13: Xgboost model fit on training and test set. The two left plots are density plots of the predicted probability of developing CHD. The two right plots are boxplots of the predicted probability of developing CHD stratified by the outcome the outcome.

Boxplots of the predicted probability of developing CHD stratified by the outcome for the training and test set are also presented in Figure 7.13. For both boxplots, the predicted probability is higher for the case group than the group of controls. As for the random forest model, this indicates that the xgboost model captures some patterns in the data set. The difference of the average predicted probability for the two groups of CHD when using the training set is slightly more than 5%, while the difference of the prediction on the test set is slightly less than 5%. In contrast to the random forest model, the xgboost model has a more comparable average predicted probability for the two groups of the outcome on the training and test set.

For extreme gradient boosting, we will use the variable importance measure denoted gain. The variable importance plot of the 20 most important variables according to the xgboost model is presented in Figure 7.14. Observe that the order of the importance of the predictors is similar to the order of the random forest variable importance plot. However, a difference is that the xgboost model considers the PCs to be less important than what the random forest model did.

The predictions from the xgboost model using the training set yield an AUC score of 0.85 and a PR AUC score of 0.16. The predictions from using the test set yield an AUC score of 0.80 and a PR AUC score of 0.10. The ROC and PR curves from the test set are plotted in Figure 7.15. Based on the AUC and PR AUC scores from the test set, and from the boxplots, the random forest model and the xgboost model appear to perform similarly. However, the results from the predictions on the training set differ in the two models.

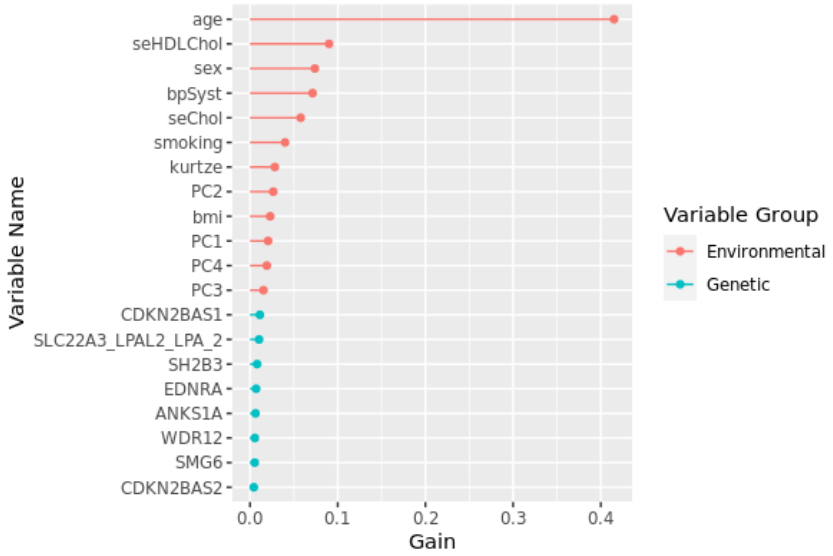


Figure 7.14: Variable importance plot of the 20 most important variables, based on gain in the xgboost model. See Table 7.1 for the description of the environmental covariate labels and Table A.2 for the genetic covariate labels.

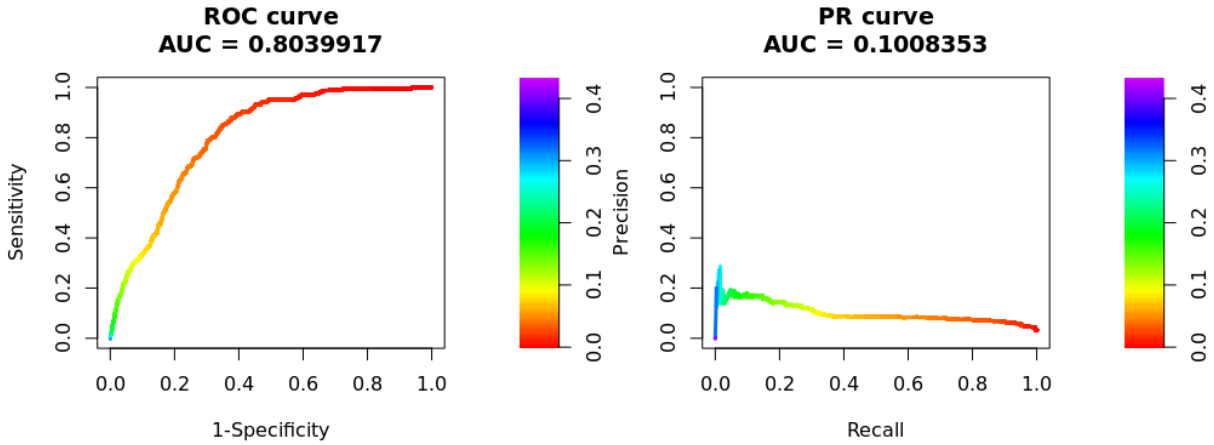


Figure 7.15: The plot to the left is the ROC curve and to the right is the PR curve, both from using the test set. The color scales on the right side of the plots give an indication which classification threshold results in a certain point on the curve.

Since it appears like the random forest model has overfitted the training data, and since the xgboost model is less computationally expensive, we choose to use the xgboost model fit in the further analysis of the genetic predictors and the interaction between physical activity and the genetic predictors.

7.3.3 The Interaction Effects

In order to evaluate the interaction effects between the genetic covariates and physical activity (PA), it is helpful to investigate the functional relationship between the genetic covariates and CHD. That is, we are interested in investigating the genetic models for each SNP, which is described in Section 6.1. The genetic effect can be visualized by partial dependence (PD) plots or accumulated local effects (ALE) plots with one variable, explained in Section 5.4. PD and ALE plots give similar results, which is why only the

PD plots are shown. The PD plots of the three most important SNPs, according to the xgboost model, are presented in Figure 7.16. From this figure, we observe that the genetic effect of the first SNP behaves dominantly, the second SNP behaves additively, and the third SNP behaves recessively.

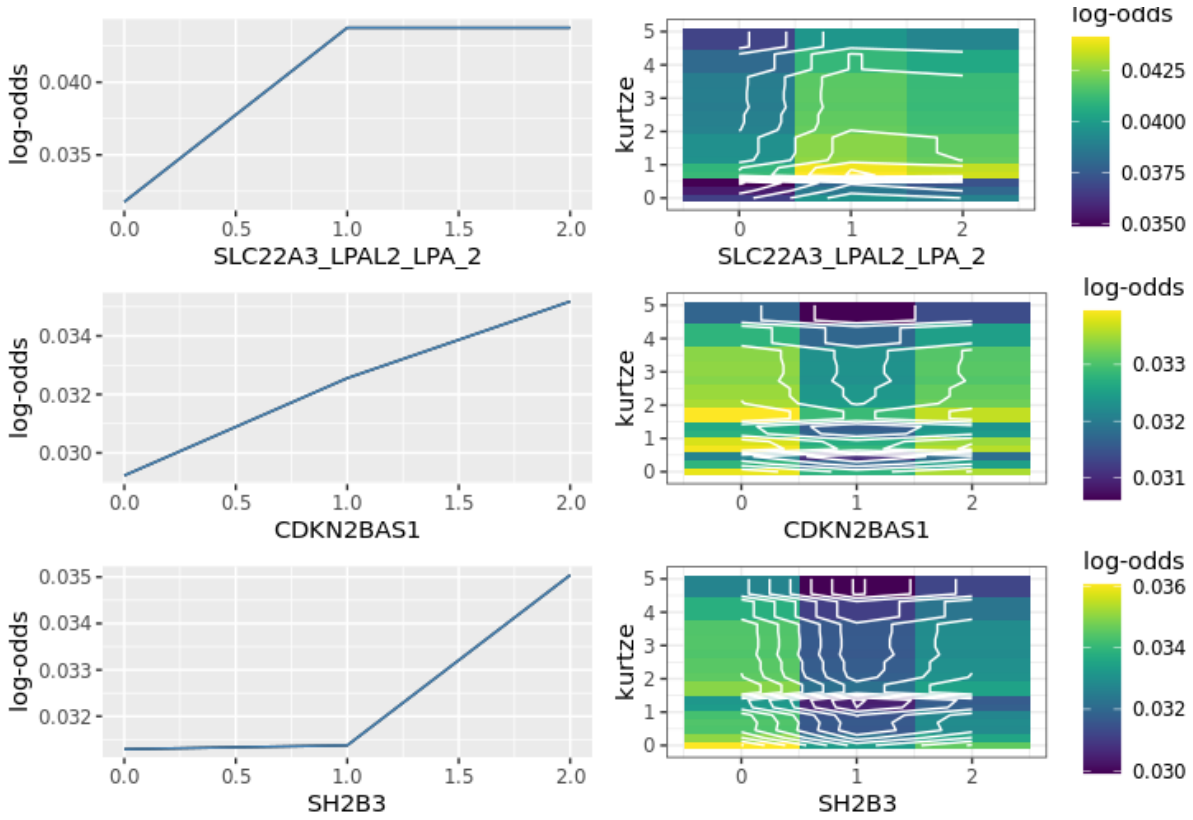


Figure 7.16: To the left are partial dependence plots of the three most important genetic predictors, according to the xgboost model. To the right are partial dependence plots of the same genetic predictors with physical activity. See Table A.2 for the covariate labels.

We have 50 SNPs in our data set, but we choose to investigate a subset of these that appear to have the most considerable effect on the probability of developing CHD. We use two different estimates of how important the SNPs are, which we get from the variable importance plot and the PD plots. We can look at the difference in the log-odds of developing CHD when the participants have 0 and 2 risk alleles from the PD plots. The subset of SNPs is chosen by taking the union of the ten most important SNPs according to these two estimates, which results in 11 SNPs. The PD plots of the rest of the SNPs in this subset can be found in the Appendix, in Figure B.5 and B.6.

The functional relationship between the SNPs, PA, and CHD can also be visualized in a PD plot. That is, we can visualize the interaction effect between PA and the SNPs in a PD plot with two variables. Such plots are included in Figure 7.16, B.5 and B.6. We are interested in whether there is an interaction effect between the genetic covariates and PA. However, observe that the difference of the log-odds of developing CHD is small in the interaction PD plots. Thus, it is challenging to observe whether there is a large interaction effect from visualizing the PD plots.

In order to investigate the PD plots of the interaction effect further, we fit a linear regression based on the values from the plot, as described in Section 6.3. The covariates in

the linear regression are PA, the SNP, and the interaction, which is added by multiplying the two covariates. The SNP covariate is coded according to the genetic model that fits best. The response variable is the log-odds values extracted from the plot. It is worth mentioning that the test is an informal way of investigating the PD plots beyond visualizing. According to the linear regression, the interaction effect between PA and each SNP is not significant for any of the SNPs. That is, from the tree ensemble approach of investigating the interaction effect, it appears like there is not a significant interaction effect between PA and the SNPs. In other words, we cannot reject the null hypothesis that there is no interaction effect between PA and the SNPs.

7.4 Logistic Regression

The last model we fit is a logistic regression model. In order to fit the model, we will first investigate the functional relationship between the covariates and the outcome from the xgboost model fit. Then we will present the model fit and investigate the interaction effects.

7.4.1 Functional Relationship Between Covariates and Outcome

For simplicity, we will only investigate the subset of the 11 most important SNPs according to the xgboost model for logistic regression as well. We model the genetic effects for logistic regression according to the genetic model that fits best based on the PD plots from the xgboost model fit.

For the environmental covariates, we find the functional relationship by investigating accumulated local effects (ALE) plots and GAM plots, explained in Section 5.5 and 4.8 respectively. In Figure 7.17, 7.18, and 7.19 are ALE plots from the random forest and xgboost model fits as well as GAM plots for all the environmental covariates. To see the variability of the data or the functional relationship in the probability scale instead of the log-odds scale, see the ICE plots shown in Appendix B.2.

First, we investigate age, BMI, and serum cholesterol based on the plots in Figure 7.17. Age appears to behave linearly between the ages of 40 and 80. However, to measure the non-linearity for ages larger than 80, we choose to add a second-order term with a center at 80 as an addition to the linear term. Furthermore, we assume that a linear covariate is sufficient for BMI and serum cholesterol when measuring the patterns we observe from the plots.

Next we investigate Figure 7.18. From the ALE plot based on the xgboost model fit, we observe that serum high-density lipoprotein cholesterol behaves linearly for values up to two. For values larger than two, the log-odds are constant. Hence, for this covariate, we set all values over two to be equal to two. For blood pressure and physical activity, we decide that a linear covariate is sufficient in order to measure the patterns.

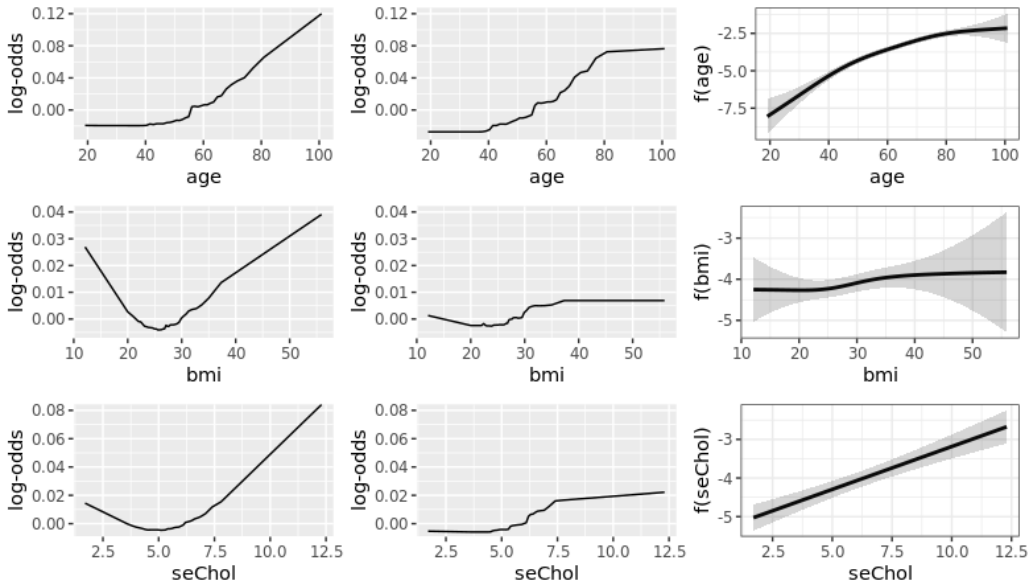


Figure 7.17: Plots for estimating the functional relationship between age, BMI, and serum cholesterol with CHD. To the left and middle are ALE plots made from the random forest model fit and the xgboost model fit, respectively. The y-axis represents the outcome given on the log-odds scale, as this is the functional relationship the logistic regression requires that we specify. To the right are GAM plots, where the relationship with the outcome is also given in the log-odds scale. However, for the GAM plots, the y-axis is the function of the variable in addition to the intercept, given in Equation (4.7).

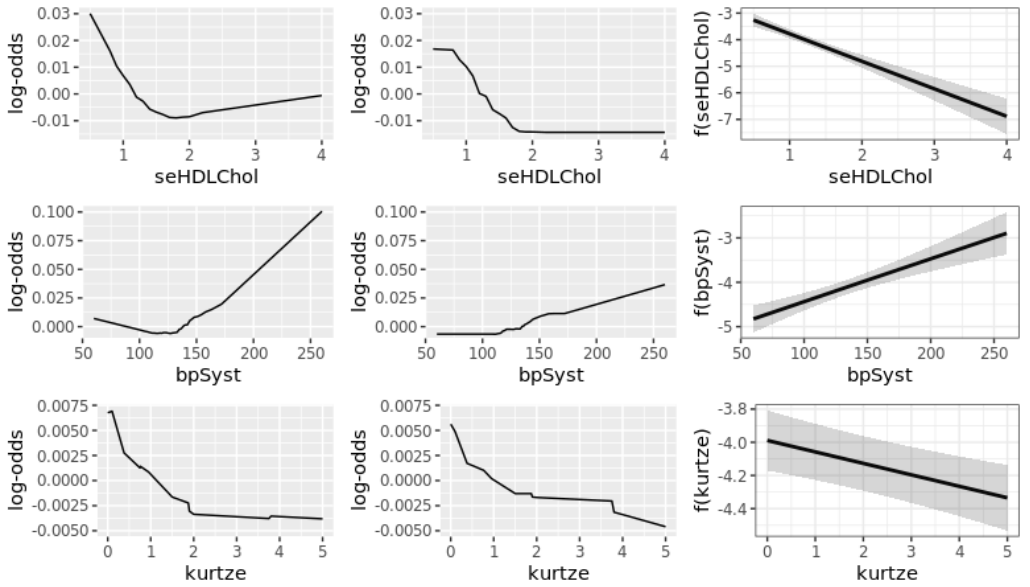


Figure 7.18: Plots for estimating the functional relationship between serum high-density lipoprotein cholesterol, blood pressure, and physical activity with CHD. To the left and middle are ALE plots made from the random forest model fit and the xgboost model fit, respectively. The y-axis represents the outcome given on the log-odds scale, as this is the functional relationship the logistic regression requires that we specify. To the right are GAM plots, where the relationship with the outcome is also given in the log-odds scale. However, for the GAM plots, the y-axis is the function of the variable in addition to the intercept, given in Equation (4.7).

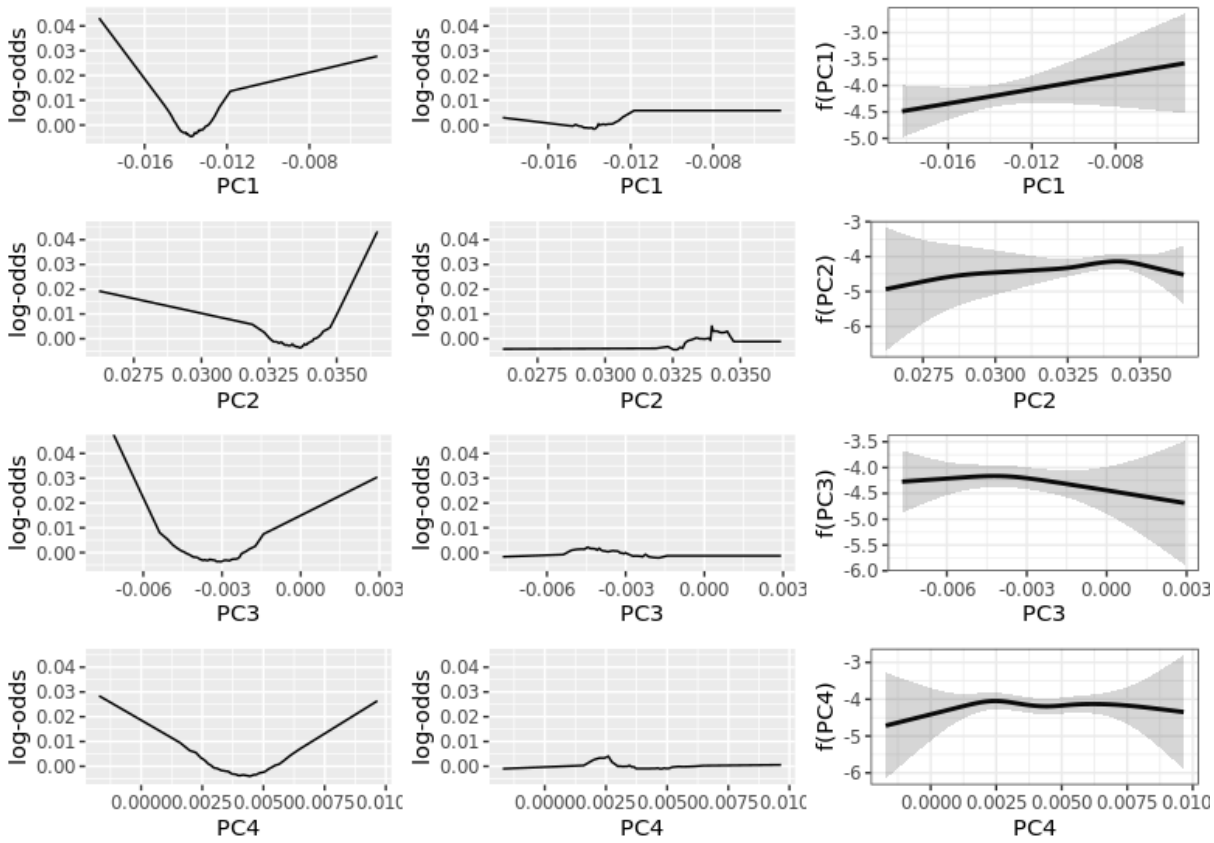


Figure 7.19: Plots for estimating the functional relationship between the PC and CHD. To the left and middle are ALE plots made from the random forest model fit and the xgboost model fit, respectively. The y-axis represents the outcome given on the log-odds scale, as this is the functional relationship the logistic regression requires that we specify. To the right are GAM plots, where the relationship with the outcome is also given in the log-odds scale. However, for the GAM plots, the y-axis is the function of the variable in addition to the intercept, given in Equation (4.7).

Lastly, we investigate the functional relationship between the PC covariates and CHD in Figure 7.19. We do not observe any particular patterns and will therefore measure the PCs as linear covariates. Observe from the ALE plots that the random forest model estimates a much larger effect for all PCs than what the xgboost model does. Recall from the variable importance plot that the PCs were considered some of the most important variables for the random forest model, in contrast to the results from the variable importance plot from the xgboost model. These results can indicate that the random forest model overestimates the effects of the PCs, which leads to some of the overfitting of the training set.

7.4.2 Model Fit with Main and Interaction Effects

Now that we know how to model each covariate, we can fit the logistic regression model. We fit the model using glm() in R with the training set. One model is fitted for each SNP. Moreover, to model the interaction effect, an interaction covariate between physical activity and the SNP is added by multiplying those covariates. What we are interested in is whether adding the interaction covariate improves the performance of the model significantly. We follow the approach where we model the main effects in addition to the interaction effects, explained in Section 6.2.1. We will not perform the stratified analysis since we see it is as sufficient to do the other approach.

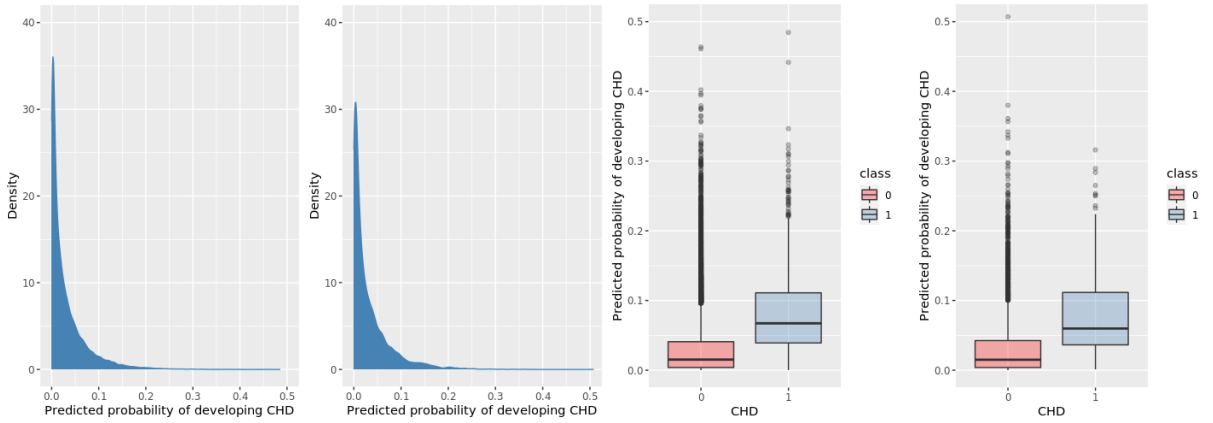
A summary of the covariate statistics for the model with the most important SNP, according to the xgboost model, is presented in Table 7.3. Notice that all covariates are significant at a significance level of 0.05, except the PCs and the interaction between physical activity and the genetic covariate. The covariate estimates for physical activity and the SNP indicate that the main effects significantly affect the probability of developing CHD. However, the covariate estimate for the interaction indicates that there is no interaction effect. In other words, based on this we cannot reject the null hypothesis that the interaction effect between physical activity and the SNP is zero.

	Estimate	Std. Error	z value	p-value
(Intercept)	-8.6612	1.4959	-5.79	$7.04 \cdot 10^{-9}$
sex	0.7886	0.0732	10.78	$2.00 \cdot 10^{-16}$
age	0.0284	0.0068	4.20	$2.62 \cdot 10^{-5}$
I((age - 80)^2)	-0.0010	0.0002	-5.70	$1.19 \cdot 10^{-8}$
smoking	0.6858	0.0780	8.79	$2.00 \cdot 10^{-16}$
bmi	0.0214	0.0084	2.53	0.0113
seChol	0.2120	0.0298	7.11	$1.13 \cdot 10^{-12}$
seHDLChol	-1.1262	0.1220	-9.23	$2.00 \cdot 10^{-16}$
bpSyst	0.0095	0.0017	5.56	$2.74 \cdot 10^{-8}$
kurtze	-0.0682	0.0193	-3.54	0.0004
PC1	50.4379	44.4013	1.14	0.2560
PC2	59.9710	46.6413	1.29	0.1985
PC3	-40.9468	31.9572	-1.28	0.2001
PC4	-17.6199	25.8973	-0.68	0.4963
SLC22A3_LPAL2_LPA_2	0.5796	0.2037	2.85	0.0044
kurtze:SLC22A3_LPAL2_LPA_2	-0.0103	0.0816	-0.13	0.8999

Table 7.3: Summary statistics of logistic regression model fit with interaction covariate between physical activity and a genetic covariate. The genetic covariate is the most important SNP according to the xgboost model, which is SLC22A3_LPAL2_LPA_2. See Table 7.1 for the description of the environmental covariate labels.

In Figure 7.20 there are two density plots of the predicted probability of developing CHD, using the training and test set for the model with the same SNP. The predicted probability of developing CHD is low since we observed a low number of cases in our data set. Boxplots of the predicted probability of developing CHD stratified by the outcome for the training and test set are also presented in Figure 7.20. For both boxplots, the predicted probability is higher for the case group than for the group of controls. As for the tree ensemble models, this indicates that the logistic regression model captures some of the

patterns in the data set.



(a) Training set (b) Test set (c) Training set (d) Test set

Figure 7.20: Logistic regression model fit on training and test set. The two left plots are density plots of the predicted probability of developing CHD. The two right plots are boxplots of the predicted probability of developing CHD stratified by the outcome. The fit is from the model with the most important SNP, according to the xgboost model.

We are interested in how well the logistic regression models perform. For the model with the most important SNP, the predictions using the training set yield an AUC score of 0.82 and a PR AUC score of 0.12. The predictions from using the test set yield an AUC score of 0.80 and a PR AUC score of 0.10. The ROC and PR curves from the test set are plotted in Figure 7.21. Additionally, the R^2_{MF} score for the model fit is 0.16.

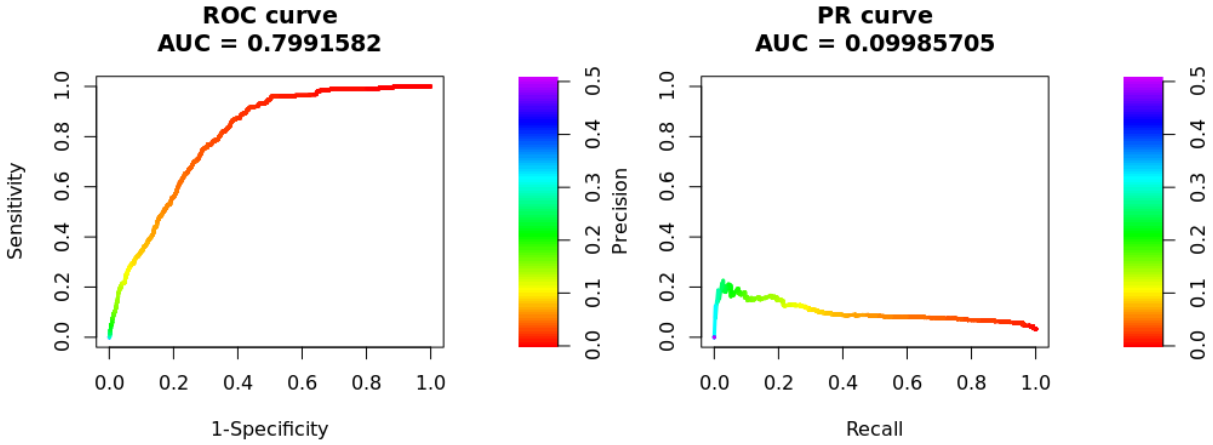


Figure 7.21: The plot to the left is the ROC curve and to the right is the PR curve, both from using the test set. The fit is from the model with the most important SNP, according to the xgboost model. The color scales on the right side of the plots give an indication which classification threshold results in a certain point on the curve.

In order to compare the model performance for logistic regression and xgboost, we can compare the AUC scores as explained in Section 3.2.1. We perform a DeLong test by using the R package pROC (Robin et al., 2011). From the test, we cannot reject H_0, which suggests that the difference of the AUC scores is not different from zero. In other

words, the test indicates that the models perform similarly in regards to the AUC score. The models also appear to perform similarly according to the density and boxplots in Figure 7.13 and 7.20. It is worth mentioning that the xgboost model has all the 50 SNPs as covariates in one model, whereas the logistic regression model only has one SNP as the genetic covariate.

The results from fitting the logistic regression model with the other 10 SNPs as the genetic covariate give similar results. The value for the genetic covariate varies, but the interaction covariate is not significant, at a significance level of 0.05, for any of the models. A plot of the 95% confidence intervals of the genetic covariate and the interaction covariate for the models with the three most important SNPs according to the xgboost model is presented in Figure 7.22. Observe that the confidence intervals for the genetic covariates do not contain zero, indicating that they are significant with a significance level of 0.05. In contrast, the confidence intervals for the interaction covariates all contain zero. As a result, we cannot reject the null hypothesis that the interaction effect is insignificant based on these results.

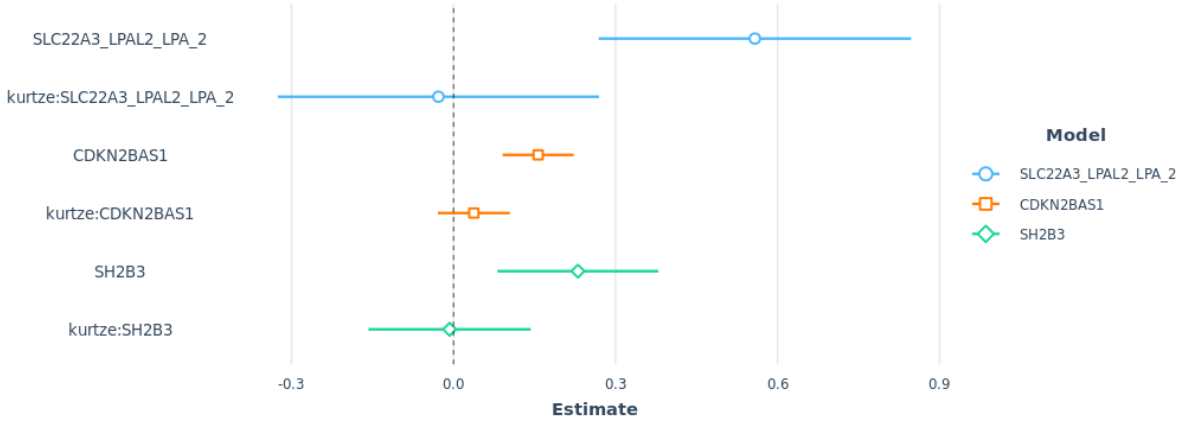


Figure 7.22: 95% confidence interval plots of genetic covariates and interaction covariates. Kurtze is the measure of physical activity. The genetic covariates are the SNPs that are considered the most important, according to the xgboost model. For description of the genetic covariate labels see Table A.2.

Another approach for investigating whether the interaction effect is significant is to perform a likelihood ratio test with the model with only the main effects and the model with both the main effects and the interaction effect. The results from the three tests for the models with the three most important SNPs, according to the xgboost model, is presented in a deviance table in Table 7.4. According to the tests, the models with the interaction covariate do not perform significantly better than the models without the interaction covariate, at significance level 0.05. These results indicate that there are no interaction effects between physical activity and the SNPs. The LRT is also performed on the models with the other SNPs, yielding the same results.

Model	**Residuals**	**Residual Deviance**	**Df**	**Deviance**	**Pr(>Chi)**
SLC22A3_LPAL2_LPA_2	32804	7896.6			
kurtze:SLC22A3_LPAL2_LPA_2	32803	7896.5	1	0.033844	0.854
CDKN2BAS1	32804	7889.8			
kurtze:CDKN2BAS1	32803	7888.6	1	1.2328	0.2669
SH2B3	32804	7900.7			
kurtze:SH2B3	32803	7900.6	1	0.0087375	0.9255

Table 7.4: Results from three Likelihood Ratio Tests, which is the deviance table from comparing fit with only main effects and fit with main effects and interaction effects. The three models have the three most important SNPs as genetic covariates according to the xgboost model. See Table A.2 for description of the genetic covariate labels.

Chapter 8

Discussion and Future Work

In this chapter, we discuss the results from the data analysis in the previous chapter. We also discuss the strengths and limitations and present possibilities for future work.

8.1 The Interaction Effect

The main result from the data analysis in Chapter 7 was that there was no significant or important effect of the interaction between physical activity and genetic factors. However, the main effect of physical activity had a significant effect on CHD according to the logistic regression model and was considered an important predictor according to the extreme gradient boosting model. The main effect for some of the SNPs also affected the outcome. There are several possible reasons why the interaction effects did not affect the outcome, which will be presented next.

Firstly, the physical activity level was self-reported, which may not be the most accurate measure of the actual physical activity level. Secondly, from the correlation plot of the environmental covariates, we observed that physical activity was correlated with many of the other environmental covariates. Consequently, some of the effects of being physically active may be measured in other covariates, such as BMI or blood pressure. When we investigated the interaction effect between physical activity and the SNPs, we may have gotten a more negligible effect since the estimated effect of physical activity may be smaller than the actual effect.

Moreover, we had a highly imbalanced data set where only 3% was classified as cases. The small proportion of cases can have led to poorer prediction performance by the models, which again can have led to a smaller measured interaction effect. Additionally, we had as little as 251 cases in the test set. The small number of cases made it challenging to perform a proper evaluation of the performance of the models and even more challenging to evaluate the interaction effects between physical activity and the SNPs. The imbalanced data set was especially challenging when fitting the random forest model, as this model appeared to overfit the training data.

8.2 Strengths

Even though the interaction effect between physical activity and the SNPs did not appear to affect the outcome, both the xgboost model and the logistic regression model had predictive power. Both models had an AUC score of around 0.8 on the test set. All of the environmental covariates and some of the genetic covariates were important or significant according to the models.

Extreme gradient boosting has empirically proven to be a highly effective approach to predictive modeling and wins "every" machine learning competition, as explained by Nielsen (2016). One of the reasons for this is that the functional relationship between the covariates and the outcome does not need to be specified for tree ensemble models, in contrast to the logistic regression model. Additionally, in xgboost the interaction effects are included automatically, whereas it has to be specified to be included for logistic regression. For this reason, we chose to use the xgboost model in our analysis. However, the logistic regression model is a lot more interpretable than xgboost. Especially those from a medical background prefer to interpret a logistic regression model over a black box tree ensemble model. Thus, we also chose to use a logistic regression model. We used a block box model, namely xgboost, to estimate the functional relationship between the covariates and the outcome by partial dependence plots and accumulated local effects plots. The information from the plots was again used to fit a logistic regression model. Consequently, we got some of the benefits from the xgboost model while still having interpretability. The approach was performed on both the environmental and genetic covariates and gave a logistic regression model that predicted almost as well as the xgboost model. From the DeLong test performed, the AUC scores from the two models were not significantly different, indicating that the models performed similarly. Thus, a strength of this analysis was that we used explainable AI on medical data.

Another strength is the large sample of cohort data available from the HUNT data set. We had extensive information about each participant's genetics, lifestyle habits, family history, and medical condition. The information about both the genetic and environmental variables was collected at HUNT test stations, except for the physical activity level, which was self-reported. Most of the information used in this analysis is therefore highly reliable.

8.3 Limitations

One of the main limitations of this analysis was that we used the training data to fit the tree ensemble models, which we again used to fit the logistic regression model. That is, we used the training data to fit the logistic regression model, but we have already used this data to find the functional relationships between the covariates and the outcome. In other words, we performed selective inference, which is a weakness when it comes to replicability as explained by Benjamini (2020). Consequently, we cannot trust the p-values from the logistic regression. However, we still fitted the logistic regression model due to the advantage of the interpretability.

Logistic regression requires that we have independent observations. However, there may be positive correlation between the feature vectors of some participants due to genetic

relations or environmental similarities in our data set. We adjusted for genetic relation and environmental similarities by adding the four first principal components (PC), as this is a common approach. We did that without testing whether it resulted in a less dependent data set, and this is therefore a weakness with our analysis. It can be helpful to investigate whether adding four PCs is the optimal approach to correct the correlation between participants. An alternative approach that can correct for the correlation is to remove the participants that are closely related. Another alternative is to reduce bias due to the relation between participants and environmental similarities by using genomic control or using a generalized linear mixed model (Zhou et al., 2018). This is a statistical method commonly used to control for the confounding effects of population stratification in genetic association studies. However, it requires that we have a large set of SNPs.

8.4 Future work

There are multiple possible directions for future work. One of the limitations of this analysis was the highly imbalanced data set. A possibility for handling this is to perform oversampling or undersampling, which are techniques used to adjust the ratio between the cases and controls. Oversampling is the most relevant approach in our case, which is when the size of the minority class is increased by making copies of observations from the minority class and adding them to the data set. Oversampling could lead to better predictions and may give a better insight into the interaction effect between physical activity and the SNPs.

Another option to handle the imbalanced data set can be to remove the related participants as long as they are controls. That is, if two participants are related and one of them developed CHD, we remove the other participant. Consequently, we both correct the correlation between participants due to genetic relations and get a less imbalanced data set.

We performed a complete case analysis in this thesis, which means that we removed participants that had missing values from the data set. However, a third alternative to handle the imbalanced data set can be to perform imputation on the cases. Imputation preserves the observations with missing data by estimating the missing values based on the rest of the data set. This will lead to a higher number of cases in our data set.

Another direction for future work can be to analyze more SNPs. We only analyzed 11 out of the 50 SNPs in our data set. Hence, for further work, it can be interesting to investigate the rest of these SNPs. It would also be interesting to analyze other SNPs associated with CVD. It has previously been hypothesized that SNPs with the strongest associations with the outcome variable in GWAS may be the least sensitive to environmental and lifestyle influences (Scott et al., 2012). For this reason, it would be interesting to study SNPs previously found to have a modest association with CVD as well.

Investigating the interaction effect with other null models could also gain insight into the significance of the interaction between physical activity and the SNPs. We selected variables for the null model based on medical advice, but there may be better alterna-

tives. For instance, adding other variables and interaction terms could be an option. As explained in Section 8.1, some of the effects of being physically active can have been measured by other variables in the null model due to the correlation between these variables. Hence, substituting or removing the variables that were associated with physical activity could lead to other results. Substituting the self-reported physical activity measure with another variable that measures the physical condition can also lead to other results. For instance, maximal oxygen uptake, often denoted VO_2max, has shown to be inversely associated with CVD in population-based studies (Andersen et al., 2015) and is not self-reported. Thus, it may lead to a more reliable measure of how physically active a participant is.

Finally, men and women respond differently to diseases and pharmaceuticals used to treat it (Brazil, 2020). For this reason, it may increase the predictive power to analyze men and women separately. Both physical activity and the SNPs may have different effects depending on whether the participant is male or female. Consequently, analyzing men and women separately may lead to a different result of the interaction effect between physical activity and the SNPs. For this reason it is another interesting possibility for future work.

Bibliography

K. Andersen, F. Rasmussen, C. Held, M. Neovius, P. Tynelius, and J. Sundström. Exercise capacity and muscle strength and risk of vascular disease and arrhythmia in 1.1 million young swedish men: cohort study. *BMJ*, 351, 2015. doi: 10.1136/bmj.h4543. URL `https://www.bmj.com/content/351/bmj.h4543`.

D. W. Apley and J. Zhu. Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(4):1059–1086, 2020. doi: https://doi.org/10.1111/rssb.12377. URL `https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12377`.

Y. Benjamini. Selective inference: The silent killer of replicability. *Harvard Data Science Review*, 2(4), 12 2020. doi: 10.1162/99608f92.fc62b261. URL `https://hdsr.mitpress.mit.edu/pub/l39rpgyc`. https://hdsr.mitpress.mit.edu/pub/l39rpgyc.

R. Brazil. Why we need to talk about sex and clinical trials. *The Pharmaceutical Journal*, 304, 2020. doi: 10.1211/PJ.2020.20207976. URL `https://pharmaceutical-journal.com/article/feature/why-we-need-to-talk-about-sex-and-clinical-trials`.

T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939785. URL `https://doi.org/10.1145/2939672.2939785`.

E. R. DeLong, D. M. DeLong, and D. L. Clarke-Pearson. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. *Biometrics*, 44(3):837–845, 1988. ISSN 0006341X, 15410420. URL `http://www.jstor.org/stable/2531595`.

P. Deloukas, S. Kanoni, C. Willenborg, M. Farrall, T. L. Assimes, J. R. Thompson, E. Ingelsson, D. Saleheen, J. Erdmann, and et al. Large-scale association analysis identifies new risk loci for coronary artery disease. *Nat Genet*, 45(1):25–33, Jan 2013.

P. K. Dunn and G. K. Smyth. *Generalized Linear Models With Examples in R*. Springer, New York, NY, 2018. ISBN 978-1-4419-0118-7.

B. Efron and T. Hastie. *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science*. Cambridge University Press, USA, 1st edition, 2016. ISBN 1107149894.

ehelse. Kodeverket ICD-10 (og ICD-11). `https://www.ehelse.no/kodeverk/kodeverket-icd-10-og-icd-11`, 2021. [Accessed 17-June-2021].

L. Fahrmeir, T. Kneib, S. Lang, and B. Marx. *Regression: Models, Methods and Applications*. 01 2013. ISBN 978-3-642-34332-2. doi: 10.1007/978-3-642-34333-9.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

Helsedirektoratet. Fysisk aktivitet for voksne og eldre. https://www.helsedirektoratet.no/faglige-rad/fysisk-aktivitet-for-barn\-unge-voksne-eldre-og-gravide/fysisk-aktivitet-for-voksne-og-eldre#:~:text=Voksne%20og%20eldre%20b%C3%B8r%20v%C3%A6re,av%20moderat%20og%20h%C3%B8y%20intensitet, 2019. [Accessed 17-June-2021].

O. L. Holmen, H. Zhang, W. Zhou, E. Schmidt, D. H. Hovelson, A. Langhammer, M.-L. Løchen, S. K. Ganesh, E. B. Mathiesen, L. Vatten, C. Platou, T. Wilsgaard, J. Chen, F. Skorpen, H. Dalen, M. Boehnke, G. R. Abecasis, I. Njølstad, K. Hveem, and C. J. Willer. No large-effect low-frequency coding variation found for myocardial infarction. *Human Molecular Genetics*, 23(17):4721–4728, 04 2014. ISSN 0964-6906. doi: 10.1093/hmg/ddu175. URL https://doi.org/10.1093/hmg/ddu175.

G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning: With Applications in R*. Springer Publishing Company, Incorporated, 2014. ISBN 1461471370.

S. Kathiresan. A PCSK9 missense variant associated with a reduced risk of early-onset myocardial infarction. *N Engl J Med*, 358(21):2299–2300, May 2008.

S. Krokstad, A. Langhammer, K. Hveem, T. Holmen, K. Midthjell, T. Stene, G. Bratberg, J. Heggland, and J. Holmen. Cohort Profile: The HUNT Study, Norway. 2013. ISSN 0300-5771. doi: 10.1093/ije/dys095. URL https://doi.org/10.1093/ije/dys095.

M. Kuhn. Building predictive models in R Using the caret Package. *Journal of Statistical Software, Articles*, 28(5):1–26, 2008. ISSN 1548-7660. doi: 10.18637/jss.v028.i05. URL https://www.jstatsoft.org/v028/i05.

A. Liaw and M. Wiener. Classification and regression by randomForest. *R News*, 2(3): 18–22, 2002. URL https://CRAN.R-project.org/doc/Rnews/.

B. A. S. Lunde, T. S. Kleppe, and H. Skaug. An information criterion for automatic gradient tree boosting. *arXiv: Methodology*, 2020.

D. McFadden. Conditional logit analysis of qualitative choice behaviour. In P. Zarembka, editor, *Frontiers in Econometrics*, pages 105–142. Academic Press New York, New York, NY, USA, 1973.

D. Nielsen. Tree Boosting With XGBoost - Why Does XGBoost Win "Every" Machine Learning Competition? http://hdl.handle.net/11250/2433761, 2016. [Accessed 18-June-2021].

NIH. Genome-Wide Association Studies (GWAS) . https://www.genome.gov/genetics-glossary/Genome-Wide-Association-Studies#:~:text=A%20genome%2Dwide%20association%20study,the%20presence%20of%20a%20disease., 2019. [Accessed 4-September-2020].

NIH. What are single nucleotide polymorphisms (SNPs)? https://ghr.nlm.nih.gov/primer/genomicresearch/snp, 2020. [Accessed 4-September-2020].

M. Nikpay et al. A comprehensive 1,000 Genomes-based genome-wide association meta-analysis of coronary artery disease. *Nat Genet*, 47(10):1121–1130, Oct 2015.

NTNU. About HUNT. https://www.ntnu.edu/web/hunt/about-hunt/, 2020a. [Accessed 4-September-2020].

NTNU. HUNT cloud. https://www.ntnu.edu/mh/huntcloud, 2020b. [Accessed 4-January-2021].

NTNU. Dmp guidance. https://innsida.ntnu.no/wiki/-/wiki/English/DMP+guidance, 2020c. [Accessed 4-January-2021].

S. Purcell. Plink. http://pngu.mgh.harvard.edu/purcell/plink, 2007. [Accessed 20-October-2020].

V. Rangul, T. L. Holmen, N. Kurtze, K. Cuypers, and K. Midthjell. Reliability and validity of two frequently used self-administered physical activity questionnaires in adolescents. *BMC Med Res Methodol*, 8:47, Jul 2008.

X. Robin, N. Turck, A. Hainard, N. Tiberti, F. Lisacek, J.-C. Sanchez, and M. Müller. pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics*, 12:77, 2011.

T. Saito and M. Rehmsmeier. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS One*, 10(3): e0118432, 2015.

H. Schunkert, I. R. König, S. Kathiresan, M. P. Reilly, T. L. Assimes, H. Holm, M. Preuss, A. F. Stewart, M. Barbalic, C. Gieger, and et al. Large-scale association analysis identifies 13 new susceptibility loci for coronary artery disease. *Nat Genet*, 43(4):333–338, Mar 2011.

R. Scott, A. Chu, and et al. No interactions between previously associated 2-hour glucose gene variants and physical activity or BMI on 2-hour glucose levels. *Diabetes*, 61(5): 1291–1296, May 2012. ISSN 0012-1797. doi: 10.2337/db11-0973.

D. Taliun, S. P. Chothani, S. Schönherr, L. Forer, M. Boehnke, G. R. Abecasis, and C. Wang. LASER server: ancestry tracing with genotypes or sequence reads. *Bioinformatics*, 33(13):2056–2058, 02 2017. ISSN 1367-4803. doi: 10.1093/bioinformatics/btx075. URL https://doi.org/10.1093/bioinformatics/btx075.

N. Townsend, M. Nichols, P. Scarborough, and M. Rayner. Cardiovascular disease in Europe — epidemiological update 2015. *European Heart Journal*, 36(40):2696–2705, 08 2015. ISSN 0195-668X. doi: 10.1093/eurheartj/ehv428. URL https://doi.org/10.1093/eurheartj/ehv428.

WHO. Ischaemic heart diseases (i20-i25). https://icd.who.int/browse10/2016/en#/I20-I25, 2016. [Accessed 11-June-2021].

WHO. Cardiovascular Diseases (CVDs). https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds), 2017. [Accessed 4-September-2020].

WHO. Ageing and health. https://www.who.int/news-room/fact-sheets/detail/ageing-and-health, 2018a. [Accessed 4-September-2020].

WHO. Physical activity. https://www.who.int/news-room/fact-sheets/detail/physical-activity, 2018b. [Accessed 4-September-2020].

WHO. Diabetes. https://www.who.int/news-room/fact-sheets/detail/diabetes, 2020a. [Accessed 4-September-2020].

WHO. Obesity and overweight. https://www.who.int/news-room/fact-sheets/detail/obesity-and-overweight, 2020b. [Accessed 4-September-2020].

H. Zhao, N. Mitra, P. A. Kanetsky, K. L. Nathanson, and T. R. Rebbeck. A practical approach to adjusting for population stratification in genome-wide association studies: principal components and propensity scores (PCAPS). *Stat Appl Genet Mol Biol*, 17 (6), 12 2018.

W. Zhou, J. B. Nielsen, L. G. Fritsche, R. Dey, M. E. Gabrielsen, B. N. Wolford, J. LeFaive, P. VandeHaar, S. A. Gagliano, A. Gifford, L. A. Bastarache, W.-Q. Wei, J. C. Denny, M. Lin, K. Hveem, H. M. Kang, G. R. Abecasis, C. J. Willer, and S. Lee. Efficiently controlling for case-control imbalance and sample relatedness in large-scale genetic association studies. *bioRxiv*, 2018. doi: 10.1101/212357. URL https://www.biorxiv.org/content/early/2018/04/24/212357.

Appendix A

Dataset Construction

This appendix shows how we constructed our data set based on data from HUNT3 and Helse Nord-Trøndelag. The data set was analyzed in the HUNT Cloud.

A.1 Environmental Covariates

In the data set from HUNT we are only interested in the participants who participated in HUNT3, and we exclude the participants who have previously suffered from CVD. Using the variable names from HUNT, that is

$$\begin{aligned}
&\text{Part_NT3BLQ1} = 1,\\
&\text{Part_NT3BLM } = 1,\\
&\text{CarInfEv_NT3BLQ1} = 1,\\
&\text{CarInfEv_NT1BLQ1} = 1,\\
&\text{CarInfEv_NT2BLQ1} = 1,\\
&\text{CarInfEv_NT3CVDQ} = 1.
\end{aligned}$$

In addition, since we are only interested in a subset of the variables, we only included the variables

variables	**variables from HUNT**
id	PID_106474
sex	Sex
age	PartAg_NT3BLQ1
smoking	SmoStatNT3_recoded
bmi	BMI_NT3BLM
seChol	SeChol_NT3BLM
seHDLChol	SeHDLChol_NT3BLM
bpSyst	$\text{BPSystMn23}_N T3BLM$
kurtze	PA_H3_index_K

where the Kurtze score is calculated as described in Section 2.2.

A.2 Genetic Covariates

Holmen et al. (2014) analyzed 54 SNPs that were found to be associated with CHD in previously published GWAS studies (Deloukas et al. (2013), Kathiresan (2008), and Schunkert et al. (2011)). We extract these 54 SNPs, which can be found in the supplementary of the article. From the GWAS summary statistics presented in Holmen et al. (2014), the genomic position (GrCh37) of each SNP is retrieved. An overview of the 50 of the 54 SNPs we found in HUNT is presented in Table A.2.

First, we check for the existence of the SNPs in the HUNT Databank and record the position for each SNP. This is done using the genomic position (GrCh37), as this is what the HUNT databank uses to identify the SNPs. The software PLINK (Purcell, 2007) is used for this purpose. The input file is the PLINK .bed file together with the .bim and .fam files. The location of the PLINK files is in the CERG lab within the HUNT Cloud, in the directory archive/genotype/plink.

We check for existence of the SNPs and record the position by making a file for each chromosome denoted "i" $\in \{1:21\}$, which is done by running the following command in R

```
system(paste("plink1.9 --bfile
    ../../archive/genotypes/plink/genotyped_PID106474
    --chr ",i, " --freq --out chr", i, "freq",sep=""))
```

If information about a SNP is in HUNT, this commando will give a file with information on what the minor allele is in addition to the MAF. The output is in the form

CHR	SNP	A1	A2	MAF	NCHROBS
1	1:762320:C:T	T	C	0.0003241	138844
1	1:798959:G:A	A	G	0.2016	138844
1	1:846808:C:T	T	C	0.1805	138838
1	1:861349:C:T	T	C	0.000108	138844

If a SNP is in the HUNT database, the information needed is stored in an output matrix, where each row represents a SNP. In our case, the information we need is the name of the SNP, the location, and the minor and major alleles. We find 50 out of the 51 SNPs in the HUNT Databank.

Once the SNPs are identified in the HUNT Databank, and we use PLINK to recode the SNPs at each chromosome i. The following code is executed in R

```
system(paste("plink1.9 --bfile
  ../../archive/genotypes/plink/genotyped_PID106474
  --chr ",i," --recode A --out thischr",sep=""))
```

The commando – recode A is a modifier that gives an additive (0/1/2) component file, suitable for loading from R, to be generated. – recode A creates a file with SNP genotypes

coded as a single dosage number. For each SNP, each individual has 0, 1, or 2 of Allele 1 (A1). By default, the A1 alleles are counted. In PLINK, the coding 0-1-2 refers to the number of A1. Furthermore, from the executed code, the information we extract will be stored in a file called *thischr*, and one file is made per chromosome.

Moreover, we only extract the columns where our SNPs are located. Denote these columns colids, which are the row numbers in the matrix +6. We save the wanted information in files called snpfile1.txt, snpfile2.txt, ..., snpfile21.txt, by executing

```
dollars=paste("$2,", paste("$",colids,sep="",collapse=" , "))
system(paste("cat thischr.raw | awk '{print ",dollars,"}'
    >snpfile",i,".txt",sep=""))
```

The final files contain the genotype data in the recoded form: 0 1 2. The rows are PIDs, and then each SNP has its column. Lastly, we read in those files created above and remove observations not used by merging with existing data.

A.3 Outcome

The outcome is extracted from Hospital data from Helse Nord-Trøndelag. Furthermore, the outcome is categorized as 1 if the participant has suffered from Coronary heart disease (CHD). We only categorize the outcome as 1 if the participant suffered from CHD within the time interval where the HUNT3 study was relevant, which is after the HUNT3 study and before the HUNT4 study. That is between 31.12.2008 and 31.07.2017.

Outcome	**Variable in Helse Nord-Trøndelag**	**Description**	**Type**
chd	ICD_10[I21, I22]	Coronary heart disease	binary
st	ICD_10[I60, I61, I62, I63]	Stroke	binary
hf	ICD_10[I50]	Heart failure	binary
af	ICD_10[I48]	Atrial fibrillation	binary
all		Suffered from chd, st, hf or af	binary

Table A.1: The outcome variables extracted from Helse Nord-Trøndelag, where chd is used as outcome in the models.

				GWAS			HUNT		
Gene	**Position: GrCh37**	**SNP: rsID**	**Effect/ Non-effect allele**	**Effect allele freq**	**OR**	**p-value**	**Effect allele freq**	**OR**	**p-value**
SORT1b	1:109821511	rs602633	G/T	0.77	1.12	$1.47{\cdot}10^{-25}$	0.78	1.03	0.42
IL6R	1:154422067	rs4845625	T/C	0.47	1.04	$3.64{\cdot}10^{-10}$	0.43	1.05	0.16
MIA3	1:222762709	rs17464857	T/G	0.87	1.05	$6.06{\cdot}10^{-5}$	0.87	1.15	$6.3{\cdot}10^{-3}$
PCSK9	1:55496039	rs11206510	T/C	0.84	1.06	$1.79{\cdot}10^{-5}$	0.86	1.08	0.12
PCSK9	1:55505647	rs11591147	T/G	0.017	0.40	$2.00{\cdot}10^{-5}$	0.009	0.71	0.05
PPAP2B	1:56962821	rs17114036	A/G	0.91	1.11	$5.80{\cdot}10^{-12}$	0.93	1.16	0.02
ZEB2-AC074093.1	2:145801461	rs2252641	C/T	0.46	1.04	$5.30{\cdot}10^{-8}$	0.42	1.01	0.71
WDR12*	2:203880992	rs2351524	T/C	0.15	1.14	$1.12{\cdot}10^{-9}$	0.13	1.17	$1.5{\cdot}10^{-3}$
APOB	2:21286057	rs515135	C/T	0.83	1.08	$2.56{\cdot}10^{-10}$	0.86	1.05	0.30
ABCG5-ABCG8	2:44073881	rs6544713	T/C	0.30	1.06	$2.12{\cdot}10^{-9}$	0.27	1.06	0.11
VAMP5-VAMP8-GGCX	2:85809989	rs1561198	T/C	0.45	1.05	$1.22{\cdot}10^{-10}$	0.47	1.07	0.048
MRAS	3:138122122	rs9818870	T/C	0.14	1.07	$2.62{\cdot}10^{-9}$	0.15	1.02	0.73
COLQ/HACL1/ BTD/ANKRD28	3:15648004	rs7651039	C/T	0.54	1.06	$1.64{\cdot}10^{-6}$	0.48	1.09	0.01
EDNRA	4: 148393664	rs1878406	T/C	0.15	1.09	$2.54{\cdot}10^{-8}$	–	–	–
GUCY1A3	4:156635309	rs7692387	G/A	0.81	1.06	$2.65{\cdot}10^{-11}$	0.81	0.99	0.83
SLC22A4-SLC22A5	5:131667353	rs273909	G/A	0.14	1.09	$9.62{\cdot}10^{-10}$	0.14	1.10	0.04
PHACTR1	6:12901441	rs9369640	A/C	0.65	1.09	$7.53{\cdot}10^{-22}$	0.68	1.04	0.25
TCF21	6:134210947	rs12190287	C/G	0.59	1.07	$4.94{\cdot}10^{-13}$	–	–	–
SLC22A3-LPAL2-LPA	6:160863532	rs2048327	C/T	0.35	1.06	$6.86{\cdot}10^{-11}$	0.40	1.10	0.004
SLC22A3-LPAL2-LPA	6:160961137	rs3798220	C/T	0.10	1.28	$4.90{\cdot}10^{-5}$	0.019	1.67	$7.9{\cdot}10^{-6}$
PLG	6:161143608	rs4252120	T/C	0.73	1.06	$4.88{\cdot}10^{-10}$	0.71	1.09	0.02
ANKS1A	6:34898455	rs12205331	C/T	0.81	1.04	$4.18{\cdot}10^{-5}$	0.76	1.05	0.25
KCNK5	6:39174922	rs10947789	T/C	0.76	1.06	$9.81{\cdot}10^{-9}$	0.76	1.02	0.58
7q22	7:106938420	rs3815148	C/A	0.19	1.08	$5.33{\cdot}10^{-4}$	0.23	0.97	0.444
ZC3HC1	7:129663496	rs11556924	C/T	0.65	1.09	$6.74{\cdot}10^{-17}$	0.64	1.00	0.99
HDAC9	7:19036775	rs2023938	C/T	0.10	1.07	$4.94{\cdot}10^{-8}$	0.099	1.05	0.37
TRIB1	8:126490972	rs2954029	A/T	0.55	1.04	$4.75{\cdot}10^{-9}$	0.47	1.05	0.12
LPL	8:19813180	rs264	G/A	0.86	1.05	$2.88{\cdot}10^{-9}$	0.87	1.01	0.92
ABO	9:136154168	rs579459	C/T	0.21	1.07	$2.66{\cdot}10^{-8}$	0.23	1.08	0.048
CDKN2BAS2	9:22003223	rs3217992	T/C	0.38	1.16	$7.75{\cdot}10^{-57}$	0.34	1.15	$6.4{\cdot}10^{-5}$
CDKN2BAS1	9:22125503	rs1333049	C/G	0.47	1.23	$1.39{\cdot}10^{-52}$	0.48	1.20	$6.0{\cdot}10^{-8}$
CYP17A1-CNNM2-NT5C2	10:104719096	rs12413409	G/A	0.89	1.10	$6.26{\cdot}10^{-8}$	0.92	1.06	0.30
KIAA1462	10:30335122	rs2505083	C/T	0.42	1.06	$1.35{\cdot}10^{-11}$	0.41	1.07	0.04
CXCL12	10:44753867	rs501120	T/C	0.83	1.07	$1.79{\cdot}10^{-9}$	0.87	1.15	$4.9{\cdot}10^{-3}$
LIPA	10:90989109	rs11203042	T/C	0.44	1.04	$6.08{\cdot}10^{-6}$	0.44	1.08	0.02
LIPA*	10:91004886	rs2246942	G/A	0.38	1.06	$9.49{\cdot}10^{-6}$	0.37	1.10	$6.2{\cdot}10^{-3}$
PDGFD	11:103660567	rs974819	T/C	0.29	1.07	$3.55{\cdot}10^{-11}$	0.23	1.06	0.15
ZNF259-APOA5-APOA1	11:116611733	rs9326246	C/G	0.1	1.09	$1.51{\cdot}10^{-7}$	0.074	1.05	0.49
SH2B3	12:111884608	rs3184504	T/C	0.40	1.07	$5.44{\cdot}10^{-11}$	0.48	1.08	0.02
COL4A1-COL4A2	13:110960712	rs4773144	G/A	0.42	1.07	$1.43{\cdot}10^{-11}$	0.43	1.10	$4.0{\cdot}10^{-3}$
FLT1	13:28973621	rs9319428	A/G	0.32	1.05	$7.32{\cdot}10^{-11}$	0.33	1.06	0.11
HHIPL1	14:100133942	rs2895811	C/T	0.43	1.06	$4.08{\cdot}10^{-10}$	0.42	1.06	0.08
ADAMTS7	15:79141784	rs7173743	T/C	0.58	1.07	$6.74{\cdot}10^{-13}$	–	–	–
FURIN-FES	15:91416550	rs17514846	A/C	0.44	1.05	$9.33{\cdot}10^{-11}$	0.44	1.04	0.27
RAI1-PEMT-RASD1	17:17543722	rs12936587	G/A	0.59	1.06	$1.24{\cdot}10^{-9}$	0.51	0.99	0.85
SMG6	17:2117945	rs2281727	G/A	0.36	1.05	$7.83{\cdot}10^{-9}$	0.39	1.08	0.02
UBE2Z	17:47005193	rs15563	G/A	0.52	1.04	$9.37{\cdot}10^{-6}$	0.54	1.04	0.28
LDLR	19:11163601	rs1122608	G/T	0.76	1.10	$6.33{\cdot}10^{-14}$	0.77	1.04	0.37
ApoE-ApoC1	19:45395619	rs2075650	G/A	0.14	1.11	$5.86{\cdot}10^{-11}$	0.15	1.07	0.13
ApoE-ApoC2	19:45415640	rs445925	G/A	0.90	1.13	$8.76{\cdot}10^{-9}$	0.88	1.10	0.07
Gene desert (KCNE2)	21:35599128	rs9982601	T/C	0.13	1.13	$7.67{\cdot}10^{-17}$	0.14	1.09	0.07

Table A.2: The 50 SNPs which are used as genetic covariates. All the information in this table is copied from the supplementary material of 54 SNP ids in the article by Holmen et al. (2014), with GWAS SNPs from Deloukas et al. (2013), Kathiresan (2008), and Schunkert et al. (2011). 4 of the 54 SNPs were not present in HUNT, which is why we investigate 50 SNPs.

Appendix B

Additional Figures and Tables

This appendix includes additional figures and tables from the data analysis presented in Chapter 7.

B.1 Extreme Gradient boosting

B.1.1 Hyperparameter Tuning

The hyperparameter tuning for extreme gradient boosting is explained in Section 7.3. In Figure B.1 there is a visualization of the first step of the tuning process. The hyperparameters tuned in step 1 is the learning rate and maximum tree depth. Step 2 is tuning the minimum sum of instance weight hyperparameter, which is plotted in Figure B.2. Furthermore, step 3 tunes the column and row samples, plotted in Figure B.3. Next, step 4 is visualized in Figure B.4 and tunes the minimum loss reduction. The fifth and final step can be seen in Section 7.3 and tunes the learning rate and the number of trees.

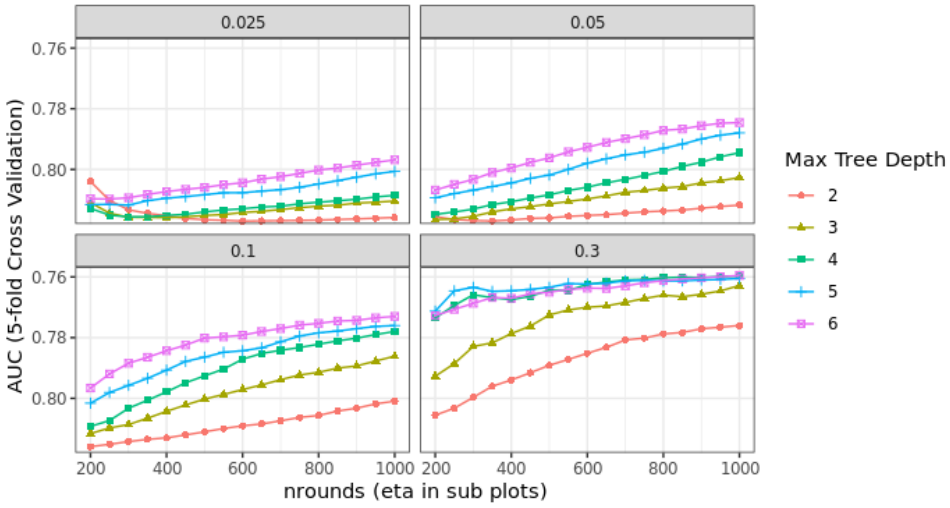


Figure B.1: Step 1 of hyperparameter tuning, with learning rate (*eta*) in the sub plots, maximum tree depth (*max_depth*) and number of trees (*nrounds*).

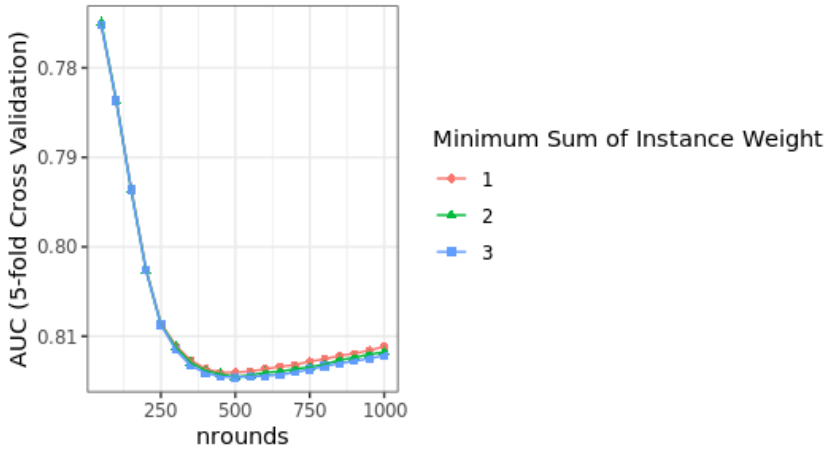


Figure B.2: Step 2 of hyperparameter tuning, with minimum sum of instance weight (*min_child_weight*) and number of trees (*nrounds*).

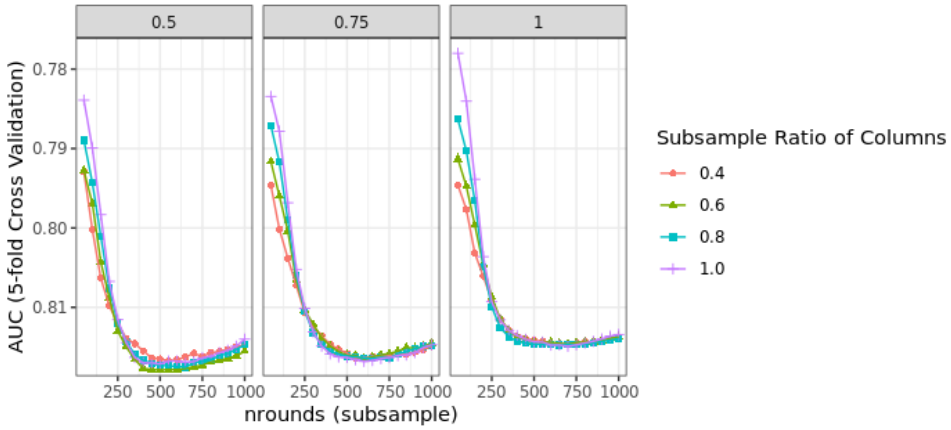


Figure B.3: Step 3 of hyperparameter tuning, with subsample ratio of columns (also denoted column sampling, *colsample_bytree*), row sampling (*subsample*) in the sub plots and number of trees (*nrounds*).

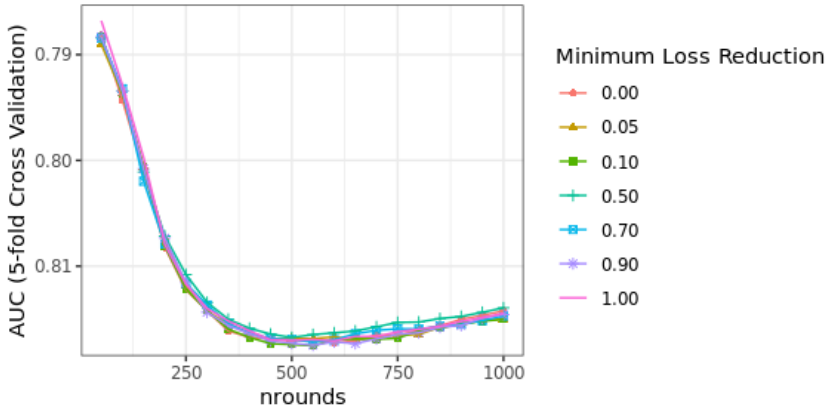


Figure B.4: Step 4 of hyperparameter tuning, with minimum loss reduction (*gamma*) and number of trees (*nrounds*).

B.1.2 Genetic Covariates and the Interaction Term

In Figure B.5 and B.6 are PD plots of the rest of the 11 SNPs we investigate in this analysis. There are PD plots of the SNP and PD plots of the interaction between the SNPs and physical activity. From the PD plots of the SNPs we choose a genetic model that is the best fit for measuring the genetic effect. The chosen genetic model for each SNP is presented in Table B.1.

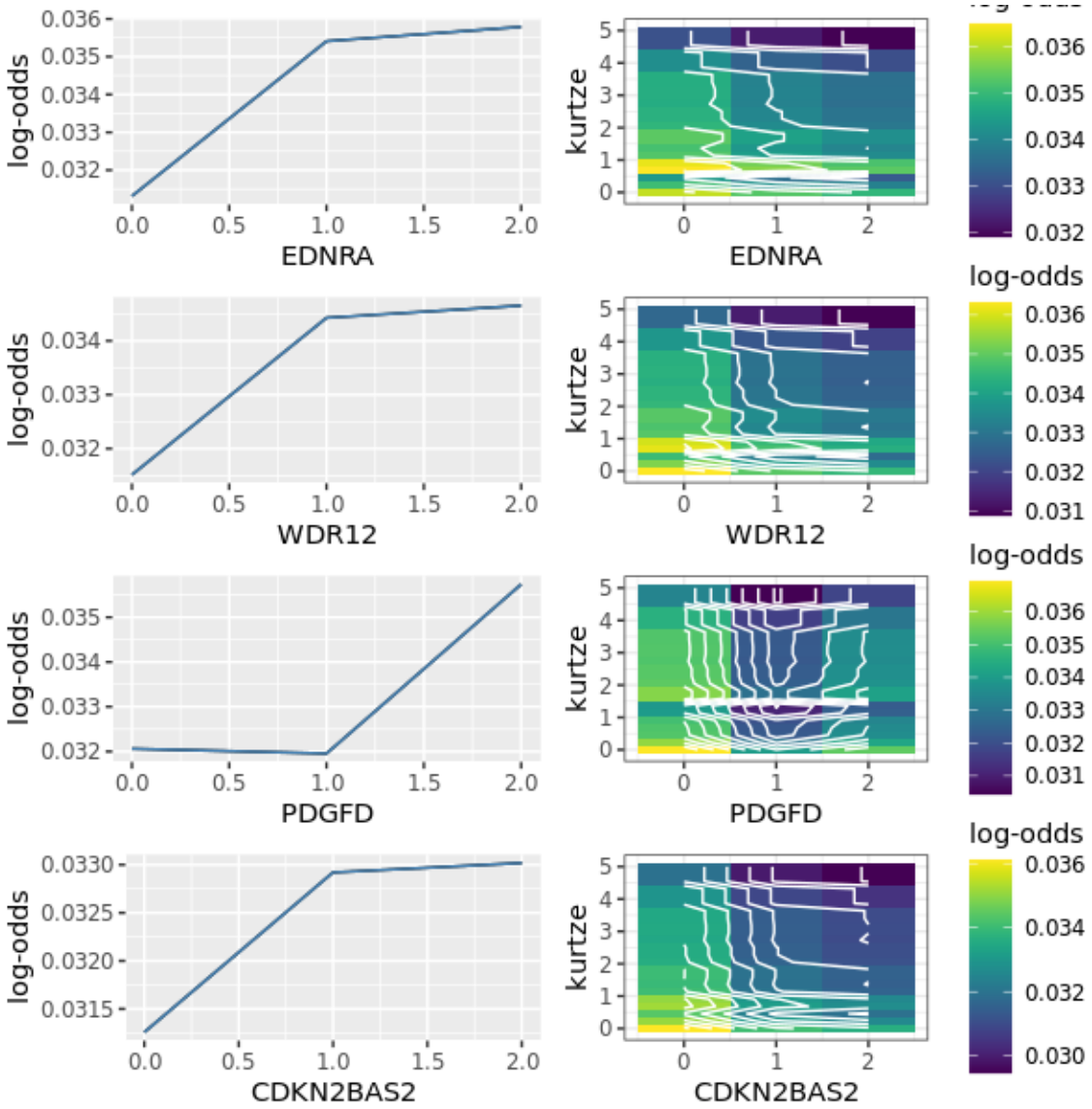


Figure B.5: To the left are partial dependence plots of four of the genetic predictors. To the right are partial dependence plots of the same genetic predictors with physical activity. See Table A.2 for description of the genetic covariate labels.

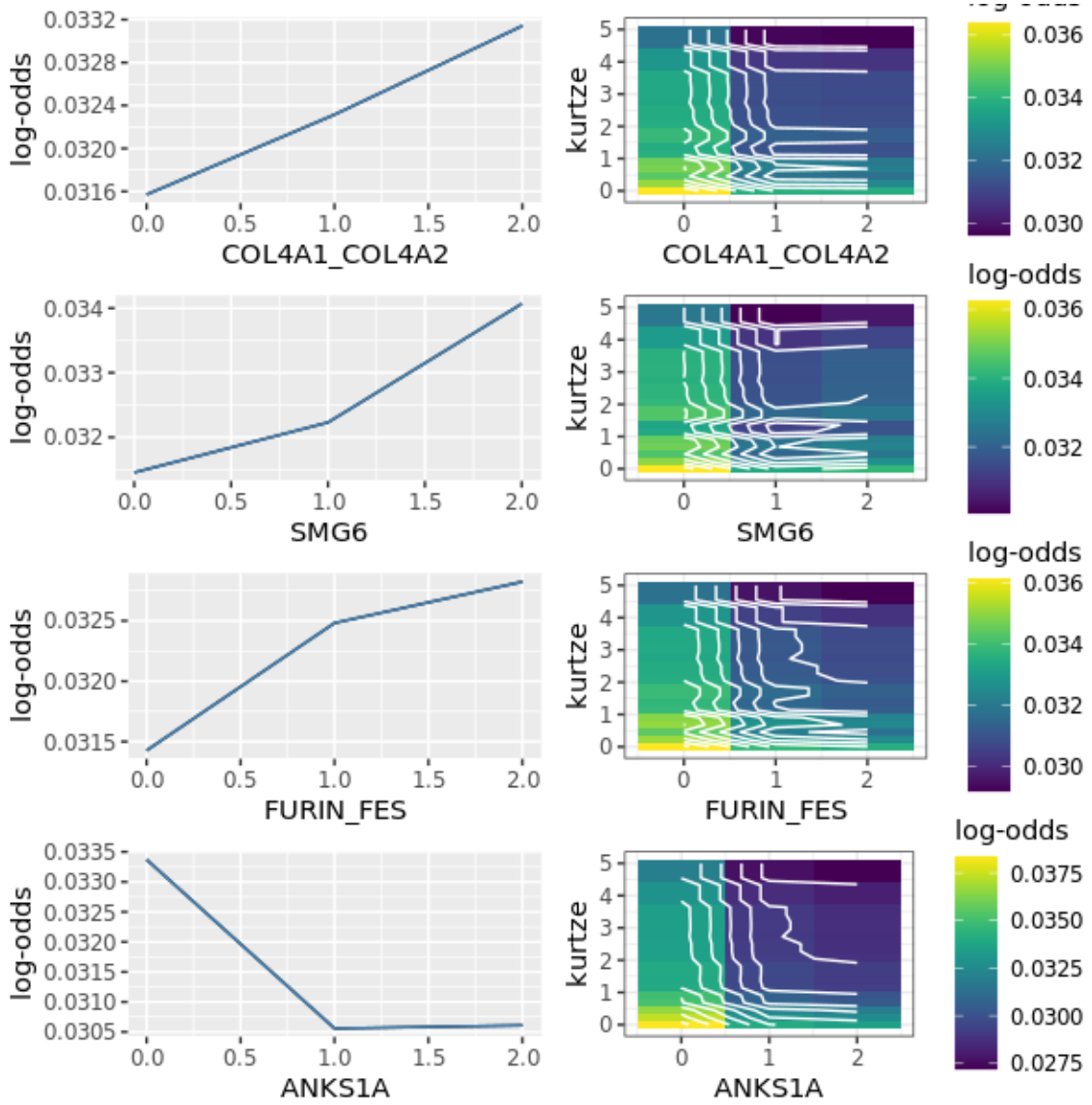


Figure B.6: To the left are partial dependence plots of four of the genetic predictors. To the right are partial dependence plots of the same genetic predictors with physical activity. See Table A.2 for description of the genetic covariate labels.

SNP	**Genetic effect**
SLC22A3_LPAL2_LPA_2	dominant
CDKN2BAS1	additive
SH2B3	recessive
EDNRA	dominant
WDR12	dominant
PDGFD	recessive
CDKN2BAS2	dominant
COL4A1_COL4A2	additive
SMG6	codominant
FURIN_FES	codominant
ANKS1A	codominant

Table B.1: The genetic model for each SNP, based on the PD plots of the SNPs. See Table A.2 for description of the genetic covariate labels.

B.2 ICE plots

In Figure B.7, B.8, and B.9 are ICE plots of the environmental and PC covariates. They are made by using 5% of the training data on the xgboost model fit. ICE plots show the functional relationship between covariates and the outcome. ICE plots also show the variability of the data. Hence, in this case, it shows the relationship and variability on 5% of the data. There are plots with both log-odds and probability scales.

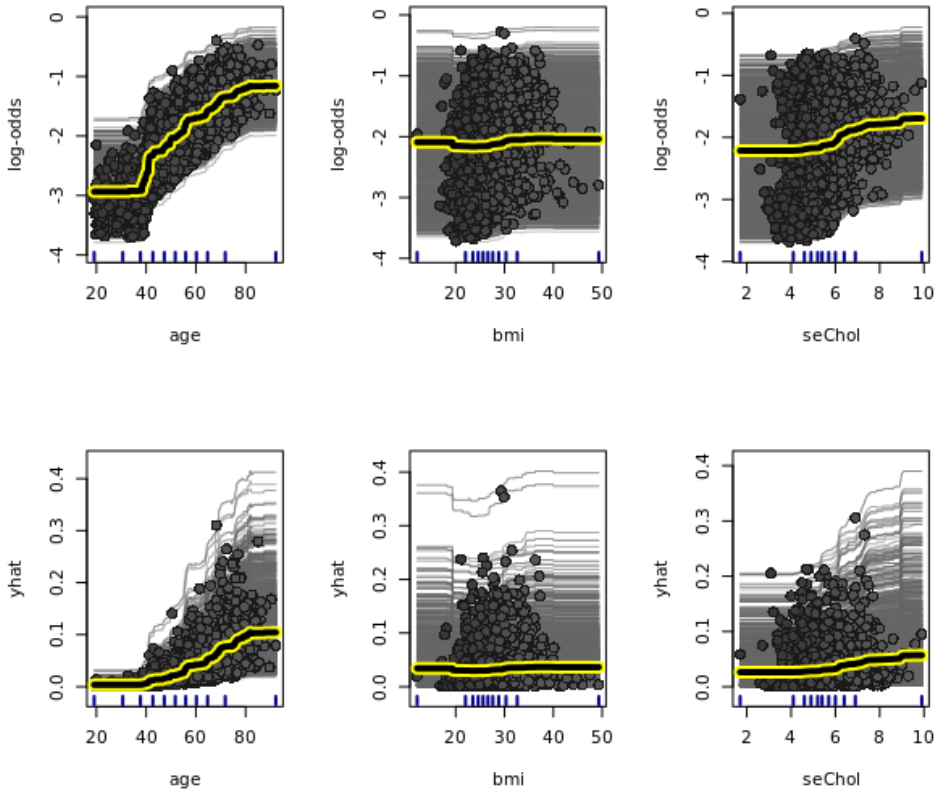


Figure B.7: ICE plots of age, BMI and serum cholesterol, made using 5% of the training data on the xgboost model fit. The top figures show the ICE plots on a log-odds scale, while the bottom figures show the ICE plots on a probability scale. See Table 7.1 for the description of the environmental covariate labels.

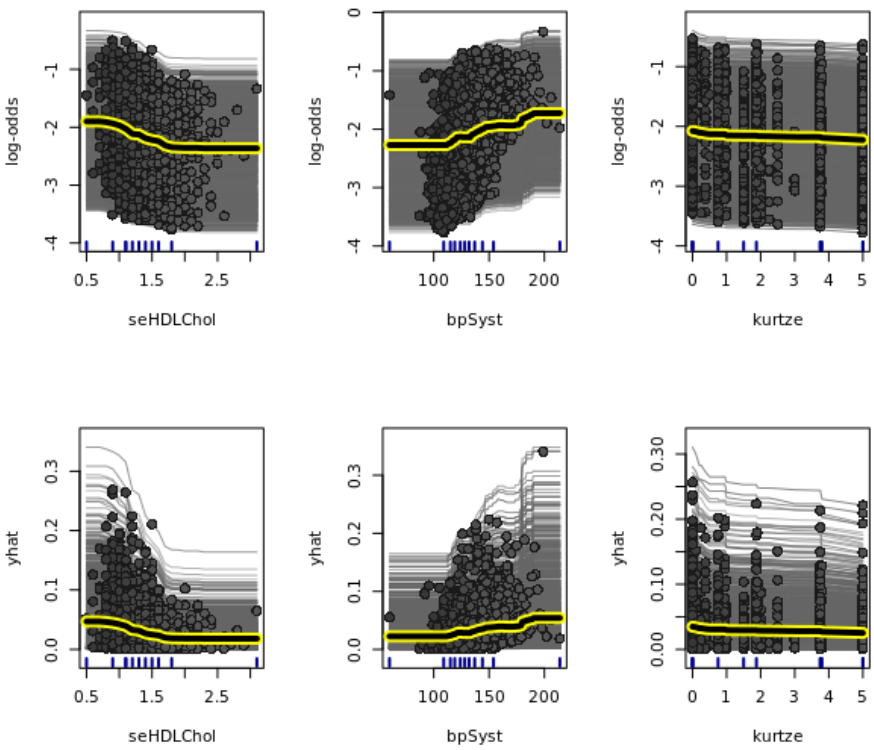


Figure B.8: ICE plots of serum high-density lipoprotein cholesterol, blood pressure, and physical activity, made using 5% of the training data on the xgboost model fit. The top figures show the ICE plots on a log-odds scale, while the bottom figures show the ICE plots on a probability scale. See Table 7.1 for the description of the environmental covariate labels.

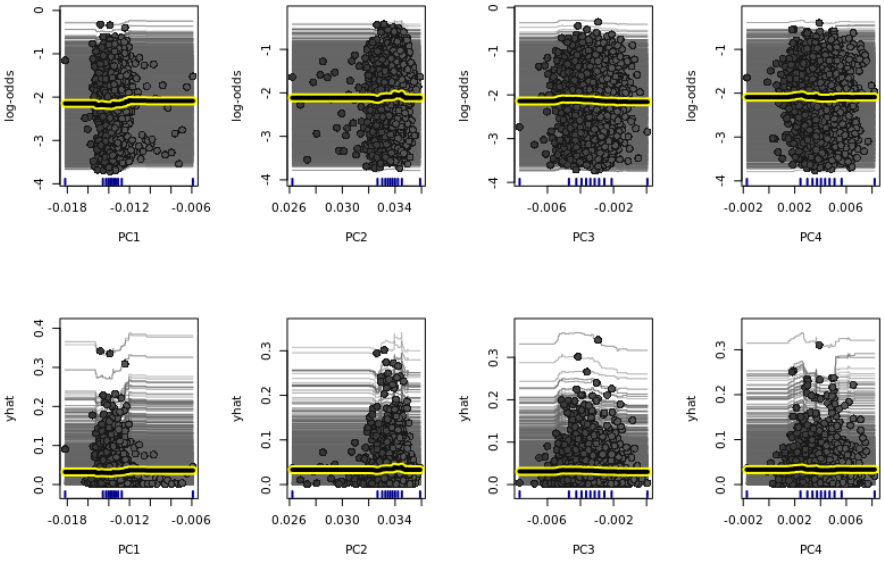
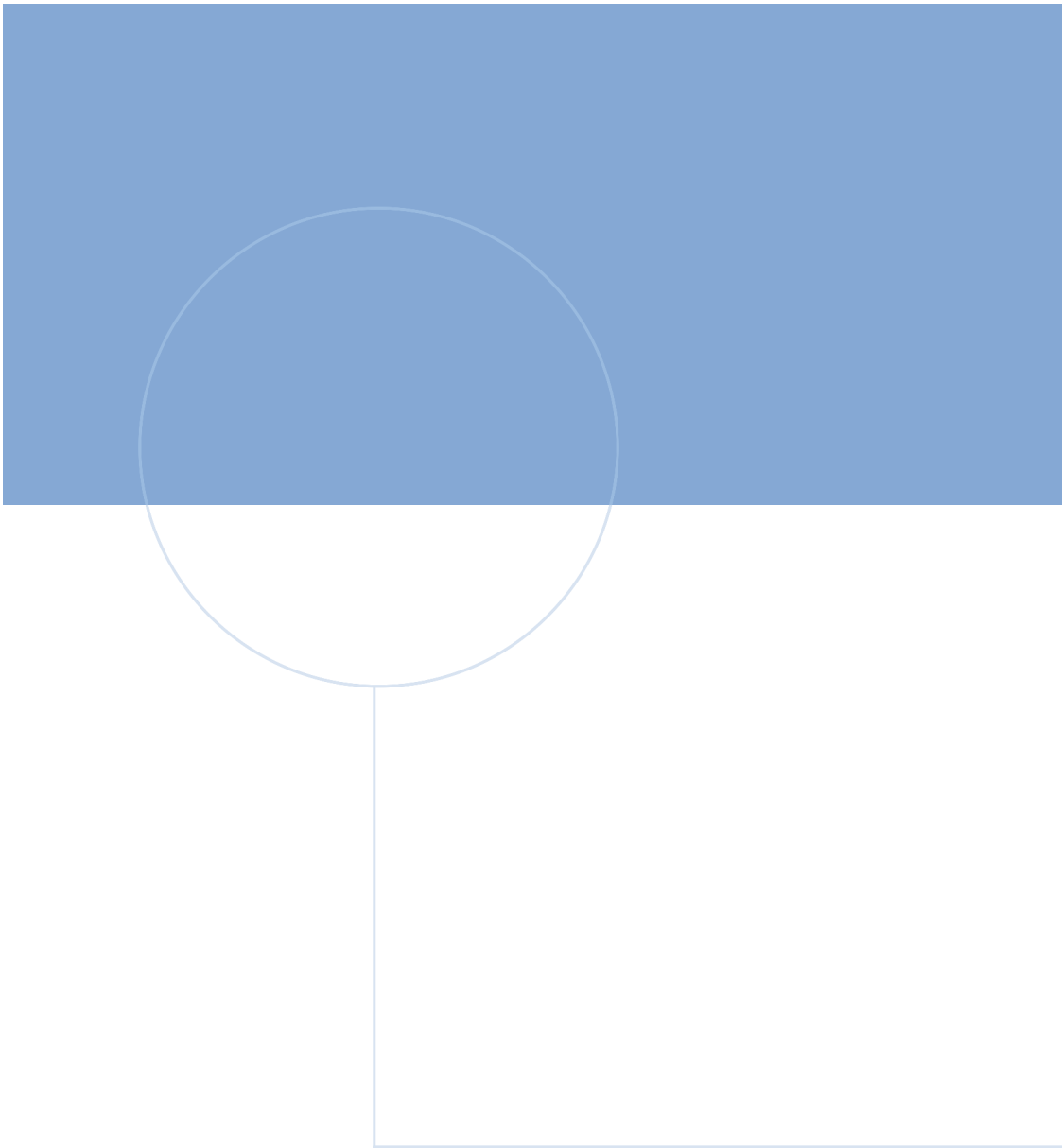


Figure B.9: ICE plots of PCs, made using 5% of the training data on the xgboost model fit. The top figures show the ICE plots on a log-odds scale, while the bottom figures show the ICE plots on a probability scale. See Table 7.1 for the description of the environmental covariate labels.



NTNU

Kunnskap for en bedre verden