Sunniva Bjelland Strømseng

Predicting Credit Card Activity for Passive Customers

Master's thesis in Applied Physics and Mathematics
Supervisor: John Sølve Tyssedal
Co-supervisor: Christian Meland

June 2023

Master's thesis

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

NTNU
Norwegian University of
Science and Technology

SpareBank 1

Sunniva Bjelland Strømseng

Predicting Credit Card Activity for Passive Customers

Master's thesis in Applied Physics and Mathematics
Supervisor: John Sølve Tyssedal
Co-supervisor: Christian Meland
June 2023

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

NTNU
Norwegian University of
Science and Technology

Preface

This master thesis finalizes my Master of Science (M.Sc.) in Applied Physics and Mathematics, with specialization in statistics at the Norwegian University of Science and Technology (NTNU). The work was completed in the spring of 2023 at the Department of Mathematical Sciences with John Sølve Tyssedal as supervisor. The subject of the thesis and data were provided by Sparebank 1, with Christian Meland as a company supervisor.

I would like to extend my sincere gratitude to my supervisor, John Sølve Tyssedal, for his valuable guidance, expertise, and the pleasant meetings we have had. I also want to thank Sparebank 1 for the opportunity to write this thesis in collaboration with them. Lastly, I am grateful for all the friendships and experiences I have gained the last five years, and the support of my friends and family.

Trondheim, June 8th, 2023
Sunniva Bjelland Strømseng

Abstract

Passive credit card customers, or dormant accounts, are a great problem for banks. Customers who open credit card accounts and either never use them or stop using them are costly, as then, the banks do not earn from interests and must cover the expenses of keeping the accounts open. This thesis's main objective is to build models to predict if passive customers will stay passive or become active within a given number of months. Thus, the problem at hand is a binary classification task where the response is either "passive" or "active". The models were constructed and optimized based on an imbalanced data set, which consisted of historical data of the customers' credit card use, provided by Sparebank 1 Kreditt AS. In addition to evaluating the models' predictive performance, the impact of unique features on the response was also considered to gain insight into which type of customers are more likely to become active.

Logistic regression and adaptive boosting (AdaBoost) were the two learning methods chosen to build the classification models used in this thesis. Logistic regression was chosen to have a benchmark result, in addition to being a well-performing and easily interpretable method. AdaBoost was chosen because it is a well-established boosting technique and has been shown to produce good results in similar studies. Logistic regression was applied to predict one, three, six and twelve months ahead in time, while AdaBoost was used to predict one and twelve months ahead. Hyperparameters of AdaBoost were tuned to improve the models' performance. An initial screening experiment was done with design of experiments, and further tuning was done through response surface methodology. Balanced accuracy was the primary metric used to evaluate the models, but sensitivity was also used to assess the models' ability to correctly classify active customers. After optimizing the models, variable importance was explored based on relative influence and Shapley values.

The logistic regression model used to predict one month ahead obtained a BACC score of 0.6181 with cutoff = 0.5, which was improved to 0.6486 after optimizing the cutoff value. For the logistic model which predicted twelve months ahead, the BACC score increased from 0.5717 to 0.6362 with optimal cutoff value. With AdaBoost, the BACC score of the one month ahead model increased from 0.6018 to 0.6859 after tuning the hyperparameters, and the twelve months ahead obtained a BACC score of 0.6779 with tuned hyperparameters versus 0.5102 with default values. Thus, optimized cutoff and hyperparameter values improved the overall performance of all models, in addition to increasing their sensitivity value, i.e., the ability to classify active customers correctly.

Sammendrag

Passive kredittkortkunder er et stort problem for kredittbanker. Kunder som åpner kredittkortkontoer og enten aldri bruker kredittkortet, eller stopper å bruke det, er kostbare. I slike situasjoner tjener ikke bankene på kunden fra renter og må i tillegg dekke utgifter for å holde kontoen åpen, til tross for at den ikke brukes. Målet med denne oppgaven er å bygge modeller for å predikere om passive kunder kommer til å forbli passive eller gå over til å bli aktive innen ett gitt antall måneder. Dermed er dette et binært klassifiseringsproblem hvor responsen er enten "passiv" eller "aktiv". Modellene ble bygget og optimalisert basert på et ubalansert datasett, som besto av historisk data av kundenes tidligere kredittkortbruk, levert av Sparebank 1 Kreditt AS. I tillegg til å evaluere hvor godt modellene predikerer, vil også de forskjellige variablenes påvirkning på responsen bli vurdert for å få bedre innsikt i hvilken type kunder som har størst sannsynlighet til å bli aktive.

Logistisk regresjon og adaptive boosting (AdaBoost) var de to læringsmetodene valgt til å bygge klassifiseringsmodellene i denne oppgaven. Logistisk regresjon ble hovedsakelig valgt for å ha et referanseresultat, i tillegg til at det er en metode som generelt presterer bra og er lett å tolke. AdaBoost ble valgt på bakgrunn av at det er en velkjent boosting teknikk som har blitt vist å gi gode resultater i lignende studier. Logistisk regresjon ble anvendt for å predikere en, tre, seks og tolv måneder frem i tid, mens AdaBoost ble brukt for å predikere en og tolv måneder frem. Hyperparametere til AdaBoost ble optimert for å forbedre modellenes ytelse. Et innledende screeningseksperiment ble gjort med forsøksplanlegging, og ytterligere optimering ble gjort ved hjelp av responsoverflatemetodikk. For å evaluere modellene var det hovedsakelig balansert nøyaktighet (BACC) som ble brukt, men sensitivitet ble også brukt for å vurdere hvor godt modellene klarte å klassifisere kundene som ble aktive. Etter optimalisering av modellene ble variablenes betydning utforsket basert på relativ innflytelse og Shapley verdier.

Den logistiske regresjonsmodellen som ble brukt til å predikere en måned frem i tid oppnådde en BACC-score på 0.6181 med cutoff = 0.5, som ble forbedret til 0.6486 etter optimalisering av cutoff-verdien. For den logistiske modellen som predikerte tolv måneder frem, økte BACC-score fra 0.5717 til 0.6362 med optimal cutoffverdi. Med AdaBoost økte BACC-verdien for modellen som predikerte én måned frem fra 0.6018 til 0.6859 etter optimering av hyperparametrene, og modellen for tolv måneder frem oppnådde en BACC-verdi på 0.6779 med optimaliserte hyperparametre mot 0.5102 med standardverdier. Dermed, ble den generelle ytelsen til alle modellene forbedret med optimaliserte cutoff- og hyperparameterverdier, i tillegg til å øke deres sensitivitetsverdi, dvs. evnen til å klassifisere aktive kunder korrekt.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Credit cards are for most adults nowadays a staple payment solution. According to Pokora (2023), 84% of U.S. adults had at least one credit card in 2021. An American adult has on average three different credit card accounts. In fact, almost 50% of American adults opened one or more new credit card accounts in 2022. The Federal Reserve Bank of San Francisco started a yearly study in 2016 to map the population's financial habits. The percentage of payments made using a credit card reached its highest level in 2021 with a total of 28%. The study also found that there is a clear correlation between both household income and usage of credit cards, as well as educational degree and credit card use. The study reports that among families with an income less than 25000 USD, only 57% have a credit card, while 98% of the families with a yearly income greater than 100000 USD have a credit card. Regarding educational level, only 52% of those with less than a high school degree have a credit card, on the other hand, 96% of those with at least a bachelor's degree have a credit card. A more recent survey done by Forbes Advisor in February 2023 found that debit cards, either physical or virtual, are the most common payment method with 54% among consumers, and credit cards, physical or virtual, come second with 36% of consumers.

There is no doubt that credit cards are a widely used payment method. Over the past decades the popularity of credit cards has increased. These changes in financial habits have also led to changes in what the issuers offer the consumers. When most of the population already own a credit card, the banks needed new methods to attract new customers, as well as retain a good relationship with their existing ones. Some of those changes are sign-up bonuses and reward programs, (Tsosie 2019). However, with the rising popularity of such bonuses and rewards, a new problem arose; credit card churning. Credit card churning is when a customer opens a new credit card account solely to benefit from the sign-in bonuses, make minimum payments with the card to achieve the benefits, and close the account before the yearly fees are charged to the card.

According to numbers from the central bank of Norway, the use of credit cards has increased also in Norway the last decade (Finanstilsynet 2022). In 2019, Gjeldsregisteret was introduced which gave an overview of unsecured loans. This made it possible for both consumers and financial institutions to see the amount of credit card debt and other consumer loans. Between 2019 and 2021 the total volume of credit decreased 28.6%. This decrease may be a result of several events. Gjeldsregisteret gave banks information about the customers' credit card loans which earlier were unknown if not specified by the customer itself. In addition, the credit limit is now counted as loan in the estimation of mortgage. Thus, some people discarded unnecessary credit cards and reduced their credit limit to be able to purchase real estate. The main reason, however, is most likely the coronavirus pandemic. Uncertainty due to the pandemic lead to reduced consumption. Earlier, the most common sector to pay with credit cards had been for abroad travels, but now, that was no longer possible.

Annual fees on credit cards are not common in Norway. As a result, the problem of credit card churning is not as relevant as in the U.S. However, the lack of fees induces another problem; dormant accounts. These are credit card accounts opened by the customer, and then either never activated the card, or the customer used it for a period then left the account open but inactive for a longer period. An owner of a dormant account is called a passive customer and is costly for the banks as the account is kept open even though it is inactive. Even though inactive accounts are costly for the banks, the possible profit of converting them into active accounts is great. Nie et al. (2011) found that retaining an existing customer can save banks up to 5 times the cost of making a sale to a new customer. Consequently, it has become more common for banks to invest more in their existing customers to keep long-lasting relations and avoid attrition.

Banks possess substantial amounts of data about their customers and the customers' consumption. The combination of substantial data and modern technology give great possibilities to learn more about the customers. Data mining, which is the process of analyzing large data sets to identify patterns and relationships, has become more common to solve many business problems. The process involves several steps where the final goal is to transform raw data into useful information. In addition to having large quantities of data, bank data is also very seldom faulty, which is favorable when the objective is to learn from the data and build appropriate models. In real-world business problems, the goal is not only to make good predictions, but model interpretability is also necessary to make well-reasoned decisions. Thus, as the application of machine learning models has become more common, so has the use of explainable AI (Artificial Intelligence), which are tools to help interpret and understand predictions made by machine learning models.

By analyzing data of dormant account owners and identifying which customers are more likely to become active, there is an immense potential economic gain for banks. Sparebank 1 Kreditt AS has collected data of some of their customers who at one point in time have been passive, in addition to longitudinal data of the relevant customers' earlier payments with the credit card. The objective of this thesis is to exploit and build models based on these data to make predictions of unfamiliar customers if they are likely to become active or not. Before being able to predict new observations, the data is pre-processed, relationships and correlation are analyzed, and hyperparameters optimized. Four different cases are considered; prediction of one, three, six and twelve months ahead in time. The predictive performance of two different models will be examined and compared. Relevant hyperparameters are chosen and tuned to optimize the models and increase the predictive performance. In addition to predicting and classifying customers into active and passive, variable importance and interpretation are evaluated. This is essential to enable Sparebank 1 to learn more about the customers and to find the common factors for those who become active.

1.1 Related Research

As there is great potential earning in identifying churners, several papers have been written where the objective is to study and predict credit card churn. Miao and Wang (2022) investigated the predictive performance of three well-known methods on credit card customer churn. The data set used to train and evaluate the models consisted of 21 explanatory variables and over 10000 observations. The paper explored random forests, linear regression, and K-nearest neighbor (KNN), and performed grid search to tune hyperparameters. Evaluation of the models was done based on AUC and recall ratio with 5-fold cross validation. The results showed that random forests performed best on both metrics, KNN obtained the second highest score of recall ratio, while linear regression had the second highest score of AUC. Furthermore, random forests was used to study the feature importance. This showed that the total transaction amount, and the count of transactions during the last twelve months, in addition to the total revolving balance on the credit card were the most influential features.

In AL-Najjar et al. (2022), feature-selection methods were used together with five different machine learning methods with the aim of predicting credit card customer churn. The five methods were Bayesian network, C5 tree, chi-square automatic interaction detection (CHAID) tree, classification and regression (CR) tree, and neural network. The C5 tree is a specific algorithm for implementation of a decision tree which uses entropy as splitting criteria and a post pruning technique by Binomial Confidence Limit, unlike the CART algorithm which uses the Gini index as splitting criteria and pre pruning by cost complexity, (Patil et al. 2012). All methods were trained, validated and tested based on three different cases; all explanatory variables, selected variables based on two-step clustering and KNN, and selected variables based on a feature-selection method. Accuracy, precision, recall, false omission rate (FOR) and $F1$-score were the metrics used to evaluate the models. All five models on the three different cases performed well with accuracy greater than 0.9. However, the combination of variable clustering with KNN and C5-tree outperformed the other methods. Total transaction count, total revolving balance and change in transaction count were found to be the three most important variables in the best performing model. Reducing the feature dimension was shown to improve the performance of the models, specifically clustering of variables.

Class imbalance in the response of a data set is common in several real-world problems, including customer churning. Geiler et al. (2022) investigated seven different methods to handle imbalanced data sets together with eight supervised machine learning methods; naïve Bayesian classifier, logistic regression, KNN, support vector machine (SVM) with and without kernel, decision trees, and two ensemble methods; random forests and extreme gradient boosting (XGBoost). Evaluation of both the rebalancing methods and learning methods are done based on AUC with 5-fold cross validation.

The results show that a model combining logistic regression, XGBoost and random forests performs best among the models. Comparing the different models applied to seven distinct types of balanced data sets in addition to no balancing, the combined model of logistic regression, random forests and XGBoost obtains highest AUC score in 6 out of 8 cases. The highest value in AUC is obtained by the combined model with a balancing method called Tomek Links applied to the data set. Tomek Links is an undersampling method which removes observations where the Euclidean distance between the majority and minority class is small, (Zeng et al. 2016).

Vafeiadis et al. (2015) did a study on five widely used machine learning methods applied to a customer churning prediction problem from the telecommunications industry. The predictive performance of multi-layer artificial neural networks, decision trees, SVM, naïve Bayes, and logistic regression were tested, as well as a boosting version of the three former methods. Two different SVM models were used, one with polynomial kernel and the other with Gaussian Radial Basis kernel function. To tune the hyperparameters Monte Carlo simulations were performed with a wide range of configurations. In the evaluation of the classifiers, four different metrics calculated based on the confusion matrix were used; precision, recall, accuracy and F-measure. Boosting of each method was done using the AdaBoost.M1 algorithm, i.e., the original AdaBoost algorithm for classification. The results show significant improvement in all methods with boosting compared to without. Specifically, the F-measure increased more than 15% for the two SVM models when boosting was applied. The best performing classifier was boosted SVM with the polynomial kernel.

1.2 Outline

In this thesis, the predictive performance of logistic regression and AdaBoost are investigated based on the problem statement. Logistic regression is chosen as the baseline method because of its simplicity in addition to in general performing quite well on several types of problems. Ensemble methods like random forests and XGBoost are often investigated in studies regarding customer churn. The results from Vafeiadis et al. (2015) showed impressive performance of AdaBoost, which is a different ensemble method. There AdaBoost was used with three different weak learners, where SVM produced the best results. However, Kim et al. (2013) found that in cases with moderate class imbalance, SVM generally performs poorer than other methods. Therefore, the second method is AdaBoost chosen with decision trees as the weak learners.

This thesis's outline is as follows. Chapter 2 describes the theoretical background for the models, in addition to performance metrics, optimization of hyperparameters and methods to calculate variable importance. In Chapter 3, the data set is described together with visualizations of selected features, and the pre-processing of the data. The analysis and results are presented in Chapter 4. Discussion of the results and concluding remarks are given in Chapter 5.

Chapter 2

Theoretical Background

This chapter delves into the theoretical foundations of the methods used in this thesis. Generalized linear models are first described, with binary logistic regression as a special case. Furthermore, adaptive boosting is introduced, along with hyperparameter tuning through response surface methodology. Also, this chapter describes the various methods and metrics used to assess and evaluate the models. Sections 2.1 and 2.2.1 are based on corresponding sections in Strømseng (2022).

2.1 Generalized Linear Model

Generalized linear models (GLMs) are an extension of ordinary linear regression that relate the linear model to the response variables using a link function. They consist of three key components, the first being the random component represented by the probability distribution of the response variable $(y_1, \dots, y_m)$, where m is the number of observations. The response variable is assumed to follow a distribution belonging to the exponential family, which can be expressed in the following form

$$f(y|\theta, \phi; w) = \exp\left(\frac{y\theta - b(\theta)}{\phi} w + c(y, \phi, w)\right),$$

where θ is the canonical parameter, ϕ is a dispersion parameter and w is a known value. The first and second derivative of the function $b(\theta)$ must exist, and $f(y|\theta, \phi; w)$ must be such that it can be normalized, (Fahrmeir et al. 2022).

The second component specifies the linear combination of the explanatory variables in the model. This is called the systematic component,

$$\eta_i = \mathbf{x}_i^T \boldsymbol{\beta} = \sum_{j=1}^{p} \beta_j x_{ij}, \quad i = 1, \ldots, n,$$

where, $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p)^T$ is the vector of unknown regression parameters, p is the number of explanatory variables, and $\mathbf{x}_i$ is a $p \times 1$ vector containing the values of the explanatory variables for y_i.

The last and third component is the link function, which specifies the relationship between the expected value of the response and the linear combination of the explanatory variables, $\eta_i = g(\mu_i)$, (Stanberry 2013).

2.1.1 Binary Logistic Regression

Logistic regression is a type of GLM that is well-suited for problems where the response variable is binary or represents a probability. When the response variable is binary, commonly represented as ”Yes”/”No” or encoded as 1/0, binary logistic regression is employed. In this case, a decision boundary, or cutoff value, is used to determine whether an instance of the response variable y_i should be classified as 0 or 1. A popular choice for the cutoff value is typically 0.5, which means that if the predicted probability of belonging to category 1 is greater than or equal to 0.5, it is classified as 1, and 0 if the probability is less than 0.5. However, in certain situations, such as when the distribution of the response variable is skewed, it may be advantageous to experiment with different cutoff values.

In binary logistic regression, it is assumed that the distribution of the response variable follows a binomial distribution with the number of independent trials, denoted by n, equal to 1 ,

$$y_i \sim \text{Bin}(n_i = 1, p_i).$$

Note that this is the same as a Bernoulli distribution with parameter p_i. The expected value of the response is by definition

$$\mu_i = \text{E}(y_i) = p_i,$$

where,

$$\begin{aligned} p_i &= \text{P}(y_i = 1|\mathbf{x}_i) \\ &= [1 + \exp(-\mathbf{x}_i^T \boldsymbol{\beta})]^{-1}. \end{aligned} \tag{2.1}$$

The systematic component defines that the explanatory variables $\mathbf{x}_i$ are linear in the parameters $\boldsymbol{\beta}$, $\eta_i = \mathbf{x}_i^T \boldsymbol{\beta}$, (Harrell 2015). From Equation 2.1 the odds of $y_i = 1$ occurring is obtained,

$$\frac{p_i}{1-p_i} = \frac{\mathrm{P}\left(y_i = 1|\mathbf{x}_i\right)}{\mathrm{P}\left(y_i = 0|\mathbf{x}_i\right)} = \exp\left(\beta_0\right)\exp\left(\beta_1 x_{i1}\right) \cdot \ldots \cdot \exp\left(\beta_p x_{ip}\right).$$

The logit link function is the natural logarithm of the odds, and by exponentiating both sides, an expression for the odds can be obtained, (Harrell 2015). Consequently, the odds ratio can be expressed as follows,

$$\begin{aligned}\frac{\mathrm{Odds}\{y_i = 1|x_{i1}, \ldots, x_{ij}+1, \ldots, x_{ip}\}}{\mathrm{Odds}\{y_i = 1|x_{i1}, x_{i2}, \ldots, x_{ip}\}} &= \frac{\exp(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_j(x_{ij}+1) + \ldots \beta_p x_{ip})}{\exp(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip})} \\ &= \frac{\exp(\beta_0) \cdot \exp(\beta_1 x_{i1}) \ldots \exp(\beta_j(x_{ij}+1) \ldots \exp(\beta_p x_{ip})}{\exp(\beta_0) \cdot \exp(\beta_1 x_{i1}) \ldots \exp(\beta_p x_{ip})}.\end{aligned} \tag{2.2}$$

From Equation 2.2, a one unit increase in x_{ij} with all other held constant, results in an increase in the odds that $y_i = 1$ by a factor of $\exp(\beta_j)$. Or perhaps a more intuitive interpretation; if x_{ij} increases by one unit, the log-odds will increase by a factor of β_j.

2.1.2 Maximum Likelihood Estimation

To derive estimates for the parameters $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ from the observed data $y_1, \ldots, y_m$, maximum likelihood estimation (MLE) is used. It is assumed that the response variable y_i follows a Binomial$(1, p)$ distribution with a probability density function of $f(y_i; p_i) = p_i^{y_i}[1-p_i]^{1-y_i}$. The likelihood function is given by,

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{m} f(y_i; p_i) = \prod_{i=1}^{m} p_i^{y_i}[1-p_i]^{1-y_i}.$$

Further, the log likelihood function is

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{m} \log\left(L_i(\boldsymbol{\beta}\right) = \sum_{i=1}^{m} \left[y_i \cdot \log(p_i) + (1-y_i) \cdot \log(1-p_i)\right]. \tag{2.3}$$

The MLE of $\boldsymbol{\beta}$ is the value that maximizes the likelihood function, but in practice, it is often easier to work with the log-likelihood function. Since the logarithmic function is monotonic, the value of $\boldsymbol{\beta}$ that maximizes the log-likelihood function also maximizes the likelihood function. Substituting Equation 2.1 into Equation 2.3 results in,

$$
\begin{aligned}
l(\boldsymbol{\beta}) &= \sum_{i=1}^{m} \left[y_i \cdot \log\left([1+\exp(-\mathbf{x}_i^T \boldsymbol{\beta}]^{-1} \right) + (1-y_i) \cdot \log\left(1 - [1+\exp(-\mathbf{x}_i^T \boldsymbol{\beta})]^{-1} \right) \right] \\
&= \sum_{i=1}^{m} \left[y_i \cdot \mathbf{x}_i^T \boldsymbol{\beta} - \log\left(\exp(\mathbf{x}_i^T \boldsymbol{\beta}) + 1 \right) \right].
\end{aligned} \tag{2.4}
$$

To obtain the estimate for $\boldsymbol{\beta}$, the first derivative of Equation 2.3 with respect to the parameters $\boldsymbol{\beta}$ is computed. It is then set equal to zero and solved for $\boldsymbol{\beta}$,

$$
\mathbf{s}(\boldsymbol{\beta}) = \frac{\partial l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_{i=1}^{m} \mathbf{x}_i (y_i - p_i) = \mathbf{0}. \tag{2.5}
$$

The function $\mathbf{s}(\boldsymbol{\beta})$ is called the score function. To solve Equation 2.5 and obtain an estimate for $\boldsymbol{\beta}$, a numerical optimization method must be employed. One widely used method is the Newton-Raphson method, which iteratively optimizes the log-likelihood function,

$$
\hat{\boldsymbol{\beta}}^{(k+1)} = \hat{\boldsymbol{\beta}}^{(k)} + \left[\mathbf{H}\left(\hat{\boldsymbol{\beta}}^{(k)} \right) \right]^{-1} \mathbf{s}\left(\hat{\boldsymbol{\beta}}^{(k)} \right) \tag{2.6}
$$

until the difference between $\hat{\boldsymbol{\beta}}^{(k)}$ and $\hat{\boldsymbol{\beta}}^{(k+1)}$ is insignificant. Here $\mathbf{H}(\boldsymbol{\beta})$ is the observed Fisher information, given by

$$
\mathbf{H}(\boldsymbol{\beta}) = -\frac{\partial \mathbf{s}(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^T}.
$$

Another optimization method is the Fisher scoring algorithm. This method replaces the observed Fisher information with the expected Fisher information,

$$
\mathbf{F}(\boldsymbol{\beta}) = \mathrm{E}\left[\mathbf{H}(\boldsymbol{\beta}) \right].
$$

The Fisher information matrix is required to be invertible for all $\boldsymbol{\beta}$ in order for the Fisher scoring algorithm to converge to the maximum likelihood solution. This requirement is met if the design matrix $\mathbf{X} = (\mathbf{x}_1, \ldots \mathbf{x}_m)$ has full rank. For the logit model, the expected Fisher information is equal to the observed, $\mathbf{F}(\boldsymbol{\beta}) = \mathbf{H}(\boldsymbol{\beta})$. Thus, in this case, the Fisher scoring algorithm corresponds to the Newton-Raphson method.

Iterative Re-weighted Least Squares (IRLS) is the method used in the `glm` function in `R` to solve the optimization problem. For a GLM,

$$\mathbf{H}(\boldsymbol{\beta}) = \mathbf{F}(\boldsymbol{\beta}) = \mathbf{x}^T\mathbf{W}(\boldsymbol{\beta})\mathbf{x}, \tag{2.7}$$

where $\mathbf{W}(\boldsymbol{\beta}) = \operatorname{diag}\left[h(\mathbf{x}_1^T\beta_1)(1-h(\mathbf{x}_1^T\beta_1)), \ldots, h(\mathbf{x}_p^T\beta_p)(1-h(\mathbf{x}_p^T\beta_p))\right]$. Insert this and the expression for the score function, Equation 2.5, in Equation 2.6 to yield,

$$\mathbf{x}^T\mathbf{W}\left(\hat{\boldsymbol{\beta}}^{(k)}\right)\mathbf{x}\hat{\boldsymbol{\beta}}^{(k+1)} = \mathbf{x}^T\mathbf{W}\left(\hat{\boldsymbol{\beta}}^{(k)}\right)\cdot\left[\mathbf{x}\hat{\boldsymbol{\beta}}^{(k)} + \mathbf{W}\left(\hat{\boldsymbol{\beta}}^{(k)}\right)\cdot\left(\mathbf{y} - h\left(\mathbf{x}^T\hat{\boldsymbol{\beta}}^{(k)}\right)\right)\right], \tag{2.8}$$

(Seeber 1993). Equation 2.8 solved with respect to $\hat{\boldsymbol{\beta}}^{(k+1)}$ results in updates which correspond to the Fisher scoring algorithm. For an adequately large sample size, the maximum likelihood estimator can be assumed to follow a normal distribution with $\boldsymbol{\beta}$ as the expected value and the inverse of the expected Fisher information as the covariance matrix,

$$\hat{\boldsymbol{\beta}} \sim \mathcal{N}\left(\boldsymbol{\beta}, F^{-1}(\boldsymbol{\beta})\right). \tag{2.9}$$

As a result, confidence bounds and hypothesis tests can be generated based on the normal distribution.

2.1.3 Model Selection

Model selection is a critical process in statistical analysis, as it enables the selection of the most suitable model for a given data set. The primary goal of model selection is to compare two or more models and determine which one provides the best fit to the observed data. An ideal model should be both accurate and interpretable. In some cases, a simpler model may be preferred over a more complex one, even if this results in a slight loss of fit for the model. Additionally, simpler models tend to have better generalization properties, meaning they can perform well on new, unseen data. The challenge in model selection lies in balancing the trade-off between model complexity and model fit, with the goal of identifying the model that best explains the data while still being interpretable.

For regression, if one candidate model includes a subset of the explanatory variables found in another, the two models are considered nested. In such cases, an analysis of variance (ANOVA) test can be used to determine if the additional variables have a significant impact on the model's fit. This is done by conducting a hypothesis test where the null hypothesis is that the parameters of the variables left out in the simple model are equal to zero, while the alternative hypothesis is that one or more of these parameters are different from zero.

The `anova` function in `R` provides various tests such as the F-test and the likelihood ratio test. The latter is based on the ratio of the log-likelihoods of the two candidate models,

$$LRT = -2\log\left[\frac{L_s(\hat{\boldsymbol{\beta}})}{L_c(\hat{\boldsymbol{\beta}})}\right] = -2\left(\log\left[L_s(\hat{\boldsymbol{\beta}})\right] - \log\left[L_c(\hat{\boldsymbol{\beta}})\right]\right),$$

where L_s and L_c are the likelihood of the simple and complex model, respectively. The LRT statistic converges to a χ^2 distribution with degrees of freedom equal to the difference in parameters between the complex and simple model. If the p-value associated with the test statistic is less than a predetermined significance level α, the null hypothesis is rejected, and the complex model is favored. Conversely, if the p-value exceeds α, the extra variables do not have a significant impact on the model's performance, indicating that the simpler model is preferable.

Subset selection is another method of model selection that aims to identify a subset of explanatory variables that have the greatest impact on the response. One common strategy for subset selection is backward elimination, which begins with the full model using all p explanatory variables and successively eliminates the variable that has the least effect on the model fit. This choice is based on the variable that results in the largest reduction in the AIC, which stands for the Akaike information criterion and is defined as

$$AIC = 2\left[k - l(\hat{\boldsymbol{\beta}})\right],$$

where k is the number of estimated parameters in the model and $l(\hat{\boldsymbol{\beta}})$ is the maximized value of the model's log likelihood function. When using AIC as model choice criteria, the best model is the one with lowest AIC value. The complexity of the model is penalized with the term $2k$, while the goodness of fit is rewarded by the term $-2l(\hat{\boldsymbol{\beta}})$.

Another criterion that can be used is the Bayesian information criterion (BIC), which is defined as,

$$BIC = k\ln(m) - 2l(\hat{\boldsymbol{\beta}}),$$

where m is the number of observations. As for the AIC, k is the number of estimated parameters in the model and $l(\hat{\boldsymbol{\beta}})$ is the maximized value of the model's log likelihood function. BIC uses the same term as AIC to reward the model's goodness of fit but penalizes by $k\ln(m)$ instead of $2k$. Thus, the preferred model is the one with lowest BIC value. The process of the backward elimination is the same regardless of if AIC or BIC is the chosen criterion.

Backward elimination examines $1 + p(1+p)/2$ models, making it a more efficient model selection method compared to other similar techniques, such as best subset selection, particularly when the number of explanatory variables p is large. However, it is a greedy method and does not guarantee to produce the optimal model using a subset of all p explanatory variables, (James et al. 2021).

2.2 Statistical Learning

Statistical learning is a branch of applied mathematics that concerns the development and application of statistical methods to solve problems related to data analysis and prediction. One of the primary objectives is to understand the underlying relationship between input variables and output variables, which can be used to make predictions or classify new observations.

Supervised learning is a specific subfield of statistical learning that deals with predicting an outcome variable based on one or more input variables. The outcome variable is often called a response, denoted y, and is connected to the explanatory variables $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$. This relationship can be expressed by,

$$y = f(\mathbf{x}) + \varepsilon,$$

where the systematic information that $\mathbf{x}$ provides about y is represented by f, and ε is a random error term. Statistical learning can be regarded as the set of techniques to estimate f. There are two types of supervised learning problems: regression and classification. In regression problems, the response is a continuous variable, while in classification problems, the response is a categorical variable. When using a supervised learner, the data is split into a training set and a test set, where the training set is used to fit the model and the test set is used to evaluate the model's performance, (James et al. 2021). The main goal of supervised learning is to minimize the difference between the predicted values and the actual values of the response variable in the test set.

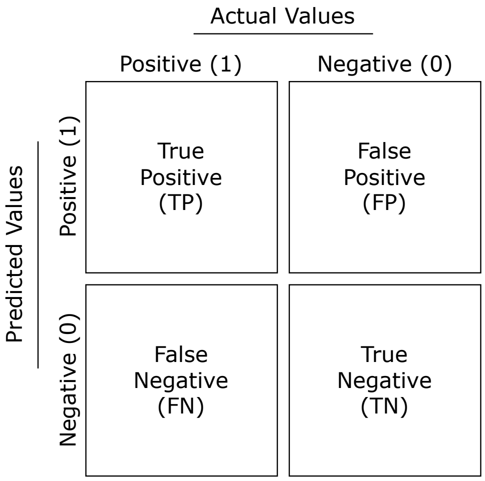
2.2.1 Performance Metrics for Classification

There are two primary considerations when evaluating a model: interpretability and performance. In the case of classification problems, the confusion matrix, shown in Figure 2.1, is often an essential factor in evaluating performance. This matrix summarizes the model's classification performance based on test data. The confusion matrix's elements show the counts of positive classified cases that were correctly identified (true positives or TP) and those that were negative (false positives or FP), as well as the count of negative classified cases that were correctly identified (true negatives or TN) and those that were incorrectly classified (false negatives or FN). A false positive is also called a type-I error, and a false negative is a type-II error. The accuracy (ACC) is one of the possible metrics that can be obtained from the confusion matrix, and it is calculated by,

$$\text{ACC} = \frac{TP + TN}{TP + FP + TN + FN}.$$

It has been established that evaluating classification performance using accuracy on an imbalanced data set presents certain issues. For instance, if the response of 80% of the observations belong to the negative class, then classifying all observations as negative will give an accuracy of 80%.

However, none of the observations that belong to the positive class has then been correctly classified. Therefore, it is advisable to consider alternative metrics when working with data where the response classes are imbalanced, (Luque et al. 2019).



Predicted Values \ Actual Values	Positive (1)	Negative (0)
Positive (1)	True Positive (TP)	False Positive (FP)
Negative (0)	False Negative (FN)	True Negative (TN)

Figure 2.1: Illustration of a general confusion matrix for classification.

The ability of a model to accurately classify the positive class is measured by sensitivity, also known as the true positive rate or recall. Sensitivity can be expressed as

$$\text{Sensitivity} = \frac{TP}{TP + FN}.$$

A value of sensitivity close to 1 indicates that the model performs well at classifying the positive class. Specificity is a model's ability to classify the negative class. Specificity is given by

$$\text{Specificity} = \frac{TN}{TN + FP},$$

and is also called the true negative rate. Like sensitivity, if the value of specificity is close to 1, then most of the negative instances are correctly classified. Sensitivity and specificity are useful for establishing the threshold for classifying a response as positive or negative. In situations where the data is imbalanced, such as in medical diagnosis, and fraud detection, the minority class is often the one of greater interest. Then, it is important to have a classifier with a strong ability to recognize and correctly classify the minority class, implying a high sensitivity while maintaining a reasonable specificity value.

Precision evaluates the number of the instances classified as positive, were actually positive,

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

The positive predictive value is another term for precision. Conversely, the negative predictive value estimates the proportion of negative instances that were accurately classified as negative,

$$\text{Negative predictive value} = \frac{TN}{TN + FN}.$$

Balanced accuracy is a metric used to evaluate classification models in situations where the distribution of the response classes in the data set is imbalanced. It provides an overall assessment of the model's performance as it measures the average of sensitivity and specificity,

$$\text{Balanced accuracy} = \frac{\text{Sensitivity} + \text{Specificity}}{2} = \frac{1}{2}\left[\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right].$$

By considering both sensitivity and specificity, balanced accuracy addresses the potential bias towards the majority class that can occur with traditional accuracy metrics. In this way, balanced accuracy gives a more reasonable estimation of the model's ability to classify both positive and negative cases. The balanced accuracy score ranges between 0 and 1, with 1 indicating perfect classification performance.

Another commonly used metric for evaluating model performance is the Receiver Operating Characteristic (ROC) curve. Consider a set of l observations to be classified as either positive or negative, each with a corresponding predicted probability $(p_1^*, p_2^*, \ldots, p_l^*)$. A threshold value $c \in (0, 1)$ is applied to determine whether an observation is classified as positive or negative, depending on whether the predicted probability is greater or less than c. To construct the ROC curve, the true positive rate (sensitivity) and false positive rate, defined as $(1 - \text{specificity})$, are computed for each value of c, and plotted with the false positive rate on the x-axis and the true positive rate on the y-axis, (Nam and D'Agostino 2002). Some examples of ROC curves are shown in Figure 2.2. A diagonal line from $(0, 0)$ to $(1, 1)$ represents a model that performs no better than random guessing, while a curve below this line indicates worse performance. The ROC curves in Figure 2.2 illustrate different scenarios of predictive performance, ranging from somewhat better than random guessing (pink curve) to perfect prediction (green curve) to a more realistic scenario of a well-performing predictor (orange curve).

The Area Under the ROC Curve (AUC) is a metric used to assess the classification performance of a model. A high AUC score close to 1 indicates a better model performance in correctly classifying positive and negative examples. On the other hand, an AUC score closer to 0.5, represented by the dashed diagonal line in Figure 2.2, suggests that the model has no discriminative ability. One way to interpret AUC is that it represents the probability of a randomly selected positive instance being ranked higher than a randomly selected negative instance by the model's classification output, (Mandrekar 2010).

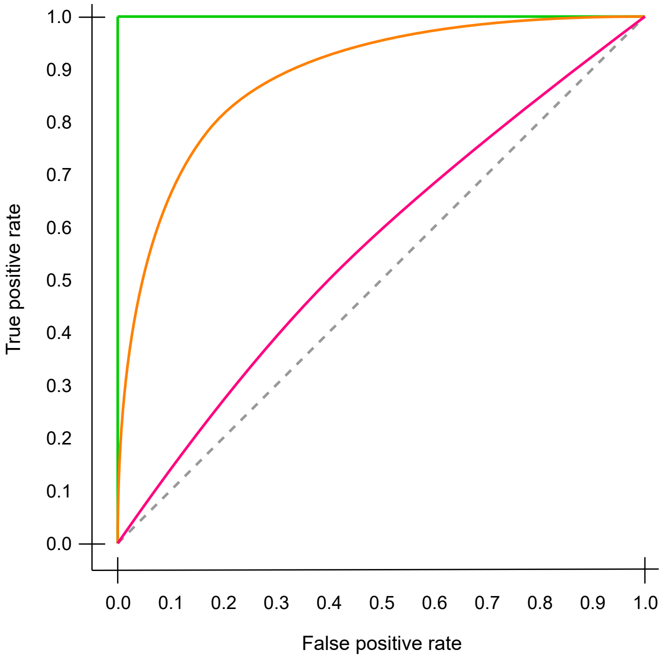


Figure 2.2: Illustration of two different ROC curves, in addition to the case of random guessing shown as a gray dashed line, and a perfect predictor shown as green. Obtained from Strømseng (2022).

2.2.2 K-Fold Cross-Validation

When a statistical model's performance is tested, it is beneficial to have a large test set to get a good estimate. However, in most cases the access to data is limited and some techniques which use the training data to estimate this quantity are helpful. k-fold cross-validation is one such approach. It randomly splits the observations in the training set into k approximately equal parts, or folds. The first fold is used as a validation set to estimate the performance metric, while the other $k - 1$ folds are used as a training set to fit the statistical model. This procedure is repeated k times, with a different fold used as validation set each time. In the end, the k different measures of the performance metric are averaged, and a final estimate of the model's performance is obtained. In practice, any $k \leq n$ can be used, where n is the number of observations. The case where $k = n$ is called leave-one-out cross-validation (LOOCV). This choice of k requires the model to be fitted n times, which can be computationally expensive, (James et al. 2021). To ease the computational cost, $k = 5$ and $k = 10$ are two popular choices. Figure 2.3 visualizes the 5-fold cross-validation.

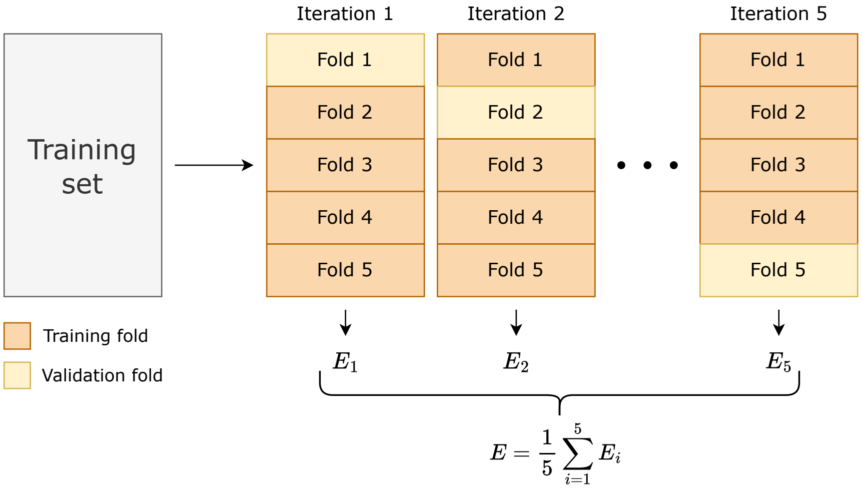


Figure 2.3: Illustration of 5-fold cross-validation.

2.3 Tree-Based Methods

Tree-based methods are supervised learning algorithms and can be applied to both regression and classification problems. The predictor space is divided into smaller sub spaces based on a set of splitting criteria. Several splits of the predictor space construct a tree like structure. This is called a decision tree, a simple and interpretable method which is the base of several powerful machine learning methods, (James et al. 2021).

2.3.1 Decision Trees

Decision trees are the foundation of tree-based methods and can solve both classification and regression problems. Regression trees predict a quantitative response, while classification trees predict a qualitative response. Because of this, the process of building the two types is done a bit differently, even though they are remarkably similar. The first step in the process of building a decision tree is to separate the predictor space, i.e., the set of possible values for the predictors, into J distinct sub spaces, or regions, $R_1, R_2, \ldots, R_j$, which also must be non-overlapping. For a regression tree, these regions are most often constructed by dividing the predictor space into high-dimensional rectangles, where the objective is to find those that minimize the RSS. Let y_i be the response of observation i in the region R_j and $\hat{y}_{R_j}$ be the mean response for the training instances within rectangle j, then the RSS is given by

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2. \tag{2.10}$$

The first split is the one that minimizes the RSS the most, then the splitting process continues at the child nodes. RSS cannot be used as a criterion to make binary splits. Thus, classification trees use other criteria that are more suited. For a classification tree, the predicted response of an observation is decided by the most commonly occurring class of training instances in that region. There are several alternatives to RSS, and one of them is the classification error rate which is the fraction of training instances which is not in the most common class in the given region. The classification error rate is given by

$$E_m = 1 - \max_k(\hat{p}_{mk}), \tag{2.11}$$

where $\hat{p}_{mk}$ is the portion of training instances in the m'th region that belong to class k. For tree-growing, however, the classification error is not adequately sensitive. For this reason, the measures Gini index and entropy are preferred. The Gini index is a measure across all K classes of the total variance and is given by

$$G_m = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}). \tag{2.12}$$

If the values of all $\hat{p}_{mk}$ are close to either zero or one, then the value of Gini index will be small. Thus, for a region, or leaf node, with most instances from the same class, the value is small. Therefore, the Gini index is called a node purity measure, (James et al. 2021). The split with the lowest Gini index is chosen as the first split, and the process is continued at the child nodes. In the case of binary classification, with p being the proportion in the second class, the Gini index can be expressed as $2p(1 - p)$.

Entropy is an alternative measure to Gini index which is quite similar, as this also will have a small value in the case where all $\hat{p}_{mk}$ are close to zero or one. Entropy is defined by

$$D_m = -\sum_{k=1}^{K} \hat{p}_{mk} \log(\hat{p}_{mk}). \tag{2.13}$$

Like for the Gini index, if the m'th leaf node is pure, then the value of entropy is small. Thus, the splits are done to minimize the entropy. Both entropy and Gini index are differentiable, and therefore suited for numerical optimization, (Hastie et al. 2009).

To consider all possible ways to divide the predictor space is computationally unattainable. Therefore, recursive binary splitting is applied. It starts at the top of the tree and divides the predictor space consecutively, a top-down approach. Recursive binary splitting is greedy as it at each step, performs the split, which is considered best at that specific step, instead of making the split which in the end will lead to an improved tree. The best split is the one which results in the greatest decrease of the chosen criteria.

The first split is done by considering all predictors $X_1, \ldots, X_p$, and all cut point values s for each predictor and selecting the predictor and cut point which yield the tree with lowest criteria value. For the next step, the algorithm evaluates the potential splits for each of the new regions. The splitting criteria for each potential split is calculated, and the split which leads to the lowest criteria value is chosen. The process is repeated until some stopping criteria is met. The response of a test instance can then be predicted by passing down its set of predictor values until a region R_j is reached. For regression, the response of that instance is predicted to be the mean of all response values in region R_j from the training instances. While for classification, it is predicted as the most frequent class of training instances in region R_j, (James et al. 2021).

Even though decision trees are nice in terms of interpretability, there are some issues and limitations related to them. Firstly, categorical predictors with many classes tend to be favored over those with less by the criterion used to split the predictor space. Cases where the number of classes in one predictor is large can lead to serious overfitting. Another issue is that they are unstable, a consequence of their hierarchical structure. Slight changes in the data can result in different trees, which is the reason why trees have high variance, (Hastie et al. 2009).

2.3.2 Ensemble Learning

Ensemble learning refers to a wide range of methods which combine the predictions of several weak learners. A weak learner can be any machine learning method, where a model is built based on input data. Let the function f represent the true relationship between the input and the response. The goal of a learning method is to find a function h which is a good approximation to f, (Dietterich et al. 2002). Each weak learner makes a prediction of a new observation using the produced model, and the ensemble method combines the predictions of all weak learners to make the final prediction, which is the output. It is an intuitive concept as humans consult others to obtain different viewpoints to make well-rounded decisions. One model may be good at predicting some observations, while have difficulties predicting others. Thus, by combining the predictions of several models, the overall error will be reduced, (Sagi and Rokach 2018).

Dietterich et al. (2002) lists three problems with methods based on one single weak learner, which can be partially solved by ensemble methods; the statistical problem, the computational problem, and the representation problem. The former occurs if the number of observations in the training data is too small for the space of weak learners under consideration to work well. Such a method has high variance. If the method cannot guarantee to find the optimal weak learner in the space of possible learners, the computational problem occurs. Lastly, if a good approximation to the true unknown function f is not contained in the space of weak learners, the representation problem occurs. The method will then have high bias. Thus, ensemble methods have the potential of reducing both the variance and bias, (Dietterich et al. 2002).

There are both dependent and independent ensemble methods. The latter is called bagging and is when each weak learner is constructed independently from the rest, random forests is one example of such methods. This makes it possible to implement the method in a parallel approach where the weak learners can be trained at the same time. In the case of dependent methods, the output of one weak learner affects how the next is produced, (Sagi and Rokach 2018). Boosting is a dependent technique which starts by building a weak learner from the training data. The next weak learner is then built to correct the errors of the previous. The procedure is repeated until either all observations in the training data are predicted correctly, or the maximum number of weak learners is reached.

2.3.3 Adaptive Boosting

Adaptive boosting (AdaBoost) is a dependent ensemble method which was proposed by Yoav Freund and Robert Shapire in 1995, (Rojas et al. 2009). It was originally introduced for binary classification, which will be the focus here, although generalizations of the algorithm have been made for multi class and bounded real valued output. AdaBoost generates a strong learner from a weighted sum of weak learners, which for $\mathbf{x}_i$ is given by

$$F(\mathbf{x}_i) = \sum_{l=1}^{L} \alpha_l h_l(\mathbf{x}_i), \tag{2.14}$$

where the α_l's are the weights and the sign of $F(\mathbf{x}_i)$ is the predicted class of $\mathbf{x}_i$. Let the input of the AdaBoost algorithm be a training set of N observations $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$. Each $\mathbf{x}_i$ contains feature values which belong to an instance space $\mathbf{X}$, and y_i is the corresponding label from a label set $Y = -1, +1$. Let $h_1, \ldots, h_L$ be the set of weak learners, where the output of each weak learner is a classification $h_j(\mathbf{x}_i) \in \{-1, +1\}$ for each $i = 1, 2, \ldots, N$. After $(m-1)$ iterations, the current strong learner is a linear combination of $m-1$ weak learners,

$$F_{(m-1)}(\mathbf{x}_i) = \alpha_1 h_1(\mathbf{x}_i) + \cdots + \alpha_{m-1} h_{m-1}(\mathbf{x}_i),$$

which is extended further in iteration m to

$$F_{(m)}(\mathbf{x}_i) = F_{(m-1)}(\mathbf{x}_i) + \alpha_m h_m(\mathbf{x}_i),$$

where α_m and h_m need to be determined in an optimal way. At the first iteration, $F_{(m-1)}(\mathbf{x}_i)$ is the zero function. The total cost of the strong learner is defined as the exponential loss

$$E = \sum_{i=1}^{N} \exp\left(-y_i \left[F_{(m-1)}(\mathbf{x}_i) + \alpha_m h_m(\mathbf{x}_i)\right]\right),$$

which is rewritten as

$$E = \sum_{i=1}^{N} w_i^{(m)} \exp\left(-y_i \alpha_m h_m(\mathbf{x}_i)\right), \tag{2.15}$$

where

$$w_i^{(m)} = \exp\left[F_{(m-1)}(\mathbf{x}_i)\right] \tag{2.16}$$

for $i = 1, 2, \ldots, N$. Here, $\mathbf{w}^{(m)}$ is the vector of weights for all observations in the training set at iteration m. At the first iteration all $w_i^{(m)}$'s are equal to 1. The sum in Equation 2.15 can be split into the total cost of correctly classified observations, and the total cost of wrongly classified observations,

$$\begin{aligned} E &= \sum_{y_i = h_m(\mathbf{x}_i)} w_i^{(m)} \exp(-\alpha_m) + \sum_{y_i \neq h_m(\mathbf{x}_i)} w_i^{(m)} \exp(\alpha_m) \\ &= \sum_{i=1}^{N} w_i^{(m)} \exp(-\alpha_m) + \sum_{y_i \neq h_m(\mathbf{x}_i)} w_i^{(m)} (\exp(\alpha_m) - \exp(-\alpha_m)). \end{aligned}$$

Then, $\sum_{y_i \neq h_m(\mathbf{x}_i)} w_i^{(m)}$ is the only part of the right-hand side that depends on h_m. Thus, assuming $\alpha_m > 0$, the h_m that minimizes E is the same that minimizes $\sum_{y_i \neq h_m(\mathbf{x}_i)} w_i^{(m)}$, which is the weak learner with the lowest weighed error. For simplicity, we write $W_c = \sum_{y_i = h_m(\mathbf{x}_i)} w_i^{(m)}$ and $W_e = \sum_{y_i \neq h_m(\mathbf{x}_i)} w_i^{(m)}$. The corresponding weight is found by differentiating E with respect to α_m,

$$\frac{dE}{d\alpha_m} = -W_c e^{-\alpha_m} + W_e e^{\alpha_m},$$

where the total cost of correctly and wrongly classified observations was simplified to $W_c e^{-\alpha_m}$ and $W_e e^{\alpha_m}$ respectively. Then, set $dE/d\alpha_m = 0$ and solve for α_m to obtain

$$\alpha_m = \frac{1}{2} \ln\left(\frac{W_c}{W_e}\right).$$

With $W = (W_c + W_e)$ as the total sum of weights, which is assumed constant in each iteration, this can be written as

$$\alpha_m = \frac{1}{2} \ln \left(\frac{W - W_e}{W_e} \right) = \frac{1}{2} \ln \left(\frac{1 - \varepsilon_m}{\varepsilon_m} \right),$$

where $\varepsilon_m = W_e / W$ is the weighted error of the weak learner h_m, (Rojas et al. 2009).

Instead of restricting the range of the weak learners to $[-1, 1]$, R. E. Schapire and Singer (1998) suggest a generalization where h_t can have range all over $\mathbb{R}$. This is an extension of AdaBoost to handle real-valued output of weak learners, called confidence-rated predictions. Let $h_t(\mathbf{x}_i)$ be the weak learner of a given instance $\mathbf{x}_i$. They interpret the sign of $h_t(\mathbf{x}_i)$ to be the predicted response of instance $\mathbf{x}_i$, equal to -1 or $+1$. The magnitude $|h_t(\mathbf{x}_i)|$ is interpreted as the level of confidence of the relevant prediction.

For a binary classification problem, a random classifier will have an error rate of $1/2$. Thus, by letting the error ε_t of a weak learner h_t be written as $\varepsilon_t = \frac{1}{2} - \gamma_t$, then γ_t is a measure of to which degree the predictions made by h_t outperform those of a random classifier. Freund and R. E. Schapire (1997) proved an upper bound of the training error ε for the final strong learner F. In the case where all weak learners perform slightly better than a random classifier, then the training error will decrease exponentially, (Freund, R. Schapire et al. 1999).

Overfitting is a recurrent challenge when training machine learning models. Although overfitting can occur when applying AdaBoost, several experiments have found that this does not happen, (R. E. Schapire 2013). Some proposal and theory in order to explain this can be found in Bartlett et al. (1998).

Many machine learning problems can be converted into optimization problems. The methods can be defined such that the objective is to minimize a loss function. This loss function should measure the goodness of fit of the model on the observed data. Even though not intended to, AdaBoost can also be seen as a procedure to minimize the exponential loss function in a greedy manner. The exponential loss is given by

$$\frac{1}{N} \sum_{i=1}^{N} \exp \{-y_i F(\mathbf{x}_i)\}, \tag{2.17}$$

where $F(x)$ is defined in Equation 2.14. As stated earlier, the final prediction is determined by the sign of F. By minimizing the exponential loss, selecting a function F which sign is probable to correspond with the correct label of $\mathbf{x}_i$ is favored. Thus, this procedure aligns with the objective of reducing the number of misclassifications. By considering AdaBoost as a greedy procedure to minimize the exponential loss, it can be regarded as a variation of functional gradient descent, which has led to generalization of boosting to other learning methods, (R. E. Schapire 2013).

2.4 Tuning of Hyperparameters

Machine learning algorithms have been applied in a wide range of application domains. They can solve a variety of problems by recognizing relationships in, often substantial amounts of, data. Problems of various kinds often require different algorithms. In general, machine learning methods have two types of parameters; model parameters and hyperparameters. The former is automatically estimated by the algorithm from the data, while hyperparameters are set manually prior to training a model, as they define the structure of the model. Hyperparameter tuning is the process of finding the optimal configuration of hyperparameters which results in building the best model, (Yang and Shami 2020). The optimal configuration is specific for each case and depends on the type of problem and data set at hand.

There are several techniques to perform optimization of hyperparameters without any deep understanding of the algorithm or its possible settings of hyperparameter values. Some of those techniques define a search space for the hyperparameters, and then search for the hyperparameter configuration which performs best, in that space. Such methods are called decision-theoretic, and grid search is one of the most used methods among the techniques within that area. Grid search evaluates all combinations of hyperparameters given to the configuration grid and can thus be regarded as a brute-force method, (Yang and Shami 2020). Grid search is convenient in the sense that it is easy to implement and can be parallelized. However, there are several disadvantages, where the main one is connected to the methods lack of efficiency when there are many hyperparameters to tune. As the number of hyperparameters increases, the number of points to evaluate in the configuration space increase exponentially, which is called the curse of dimensionality, (Yang and Shami 2020). Therefore, unless the number of hyperparameters to tune is small, such that the corresponding configuration space also is small, grid search is highly inefficient.

Another decision-theoretic method is random search. Unlike grid search which evaluates all values in the defined configuration space, random search samples a predetermined number of values from a uniform density within the upper and lower bounds, i.e., in the same space. Random search can be more effective than grid search in high dimensional spaces, specifically when the function of interest has low effective dimensionality. A function of two variables $f(x, y)$ is said to have low effective dimensionality if it can be approximated by another function of only one variable, $f(x, y) \approx g(x)$. Figure 2.4 shows how a grid of points project inefficiently onto the subspace of either parameter, in terms of the coverage of the subspace. While even though the uniformly random points are more uneven in the original configuration space, the coverage of the subspaces are much improved, (Bergstra and Bengio 2012).

Although random search can be more efficient than grid search, the approach still has some disadvantages in that each point is evaluated independent of the previous, thus time and computational effort can be wasted in less favorable regions. In addition, if the optimal configuration lies outside the defined space, this point will not be found in that search, neither will the methods give any indication that the search space should be moved to another region or in which direction that region is. Therefore, the focus in this thesis will be to optimize hyperparameters in a more systematic manner through design of experiments and the method of steepest ascent.

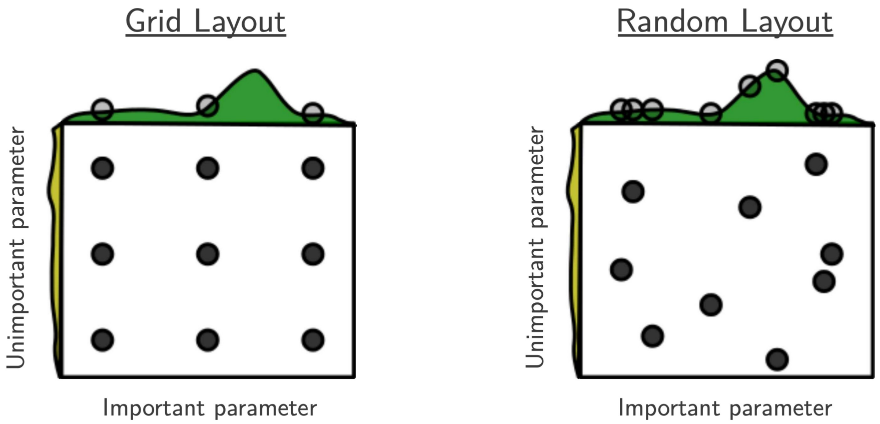


Figure 2.4: Layout of grid search to the left and random search to the right, both with nine configuration points to optimize a function with low effective dimensionality, $f(x, y) = g(x)+h(y) \approx g(x)$. Above each square layout, $g(x)$ is shown in green, and to the left of each square layout, $h(x)$ is shown in yellow. The circles on the green curve visualize the distinct values of $g(x)$ which are evaluated by the nine trial points. Obtained from Bergstra and Bengio (2012).

2.4.1 Design of Experiments

Design of Experiments (DOE) is a systematic process of constructing, conducting, and analyzing a series of experiments. An experiment is designed by selecting which variables and their range to explore, and the number of times to run it. The primary goal of DOE is to optimize the response variable by identifying the factors with greatest impact on the response. D. C. Montgomery (2017) lists three basic principles of DOE; replication, randomization and blocking. Replication is essential to estimate internal standard error. Experiments should ideally be done in a randomized order to avoid the results being affected by external factors. Blocking is useful to deal with variation in the experiment that is known but is not able to be controlled, (Lujan-Moreno et al. 2018). In the case of hyperparameter tuning where all computations are done on the same machine and software, randomization and blocking is generally not needed. One Factor At a Time (OFAT) is a method which as the name indicates, varies one factor while keeping the rest constant, (Yuangyai and Nembhard 2015), while factorial experiments vary the factors together and can thus detect relationships between factors.

2.4.2 Two Level Factorial Design

A two-level, or 2^k, factorial design is an experimental design where each factor has two levels. The k represents the number of factors in the design, and each factor has one low level and one high level, commonly denoted as -1 and 1 respectively. Every combination of factors and their respective levels are evaluated to discover both main effects and interaction effects. For each experiment, a response is obtained, thus, for a design with n experiments, there are n responses, $y_1, y_2, \dots, y_n$, which are observed when conducting the experiments. In the case where the objective is to tune hyperparameters, the hyperparameters are represented by the factors and the levels are the values of the corresponding hyperparameter to be evaluated. The response is then an evaluation metric which value is obtained by testing a trained model with hyperparameter values determined by the design. A linear regression model is fitted with the factors as explanatory variables and lastly analyzed to detect the factors with most impact on the response and the most beneficial value combination of those.

Table 2.1: Standard form of a general design of a 2^3 factorial design with three factors A, B and C.

Experiment No.	A	B	C	AB	AC	BC	ABC	Level code	y
1	-1	-1	-1	1	1	1	-1	l	y_1
2	1	-1	-1	-1	-1	1	1	a	y_2
3	-1	1	-1	-1	1	-1	1	b	y_3
4	1	1	-1	1	-1	-1	-1	ab	y_4
5	-1	-1	1	1	-1	-1	1	c	y_5
6	1	-1	1	-1	1	-1	-1	ac	y_6
7	-1	1	1	-1	-1	1	-1	bc	y_7
8	1	1	1	1	1	1	1	abc	y_8

Suppose a design with $k = 3$ factors, that is a 2^3 factorial design. Let the three factors be denoted as A, B and C, each with two levels. A full 2^3 factorial design has 8 different combinations and consists therefore of 8 experiments. Table 2.1 shows the standard form of a 2^3 factorial design with one replicate. A sign matrix determines the levels of the factors in each experiment, where -1 represents a factor's low level, and 1 represents the high level. The level code is denoted by lowercase letters and illustrate which factors are at their high level, e.g. l is the level code where all factors are at their low level, a is the level code with A at its high level while B and C are on their low level, and ab represents A and B at their high level with C on its low level. From a 2^3 factorial design, there are a total of 7 effects that can be estimated with the use of the observed responses $y_1, \dots y_8$; three main effects A B and C, three two-factor interaction effects AB, AC and BC, and one three-factor interaction effect ABC. For a two-level design, the main effect of a factor is defined as the expected average response when the relevant factor is at the high level, minus the expected average response when the factor is at the low level. A two-factor interaction is defined as half the main effect of one factor when the other is at the high level, minus half the main effect of the same factor when the other factor is at the low level.

The main effect of a single factor can be estimated by the difference of the average response when the factor is at its high level, $\bar{y}_h$, and the average response when the factor is at its low level, $\bar{y}_l$. E.g., in a 2^3 factorial design, the main effect of factor A is defined as

$$\hat{A} = \bar{y}_{A_h} - \bar{y}_{A_l} = \frac{1}{4}\left[(y_2 + y_4 + y_6 + y_8) - (y_1 + y_3 + y_5 + y_7)\right].$$

An estimate of a two-factor interaction is computed by the difference between half the main effect of one factor when the other factor is on the high level, and half the main effect of the former factor when the other is at its low level. This is equivalent to summing the responses when the two factors are at the same level, and subtract by the sum of the responses when the factors are at opposite levels, and divide by half the number of observations. The estimated interaction between A and B in a 2^3 factorial design is then given by,

$$\widehat{AB} = \frac{1}{4}\left[(y_1 + y_4 + y_5 + y_8) - (y_2 + y_3 + y_6 + y_7)\right].$$

Finally, the three-factor interaction ABC is estimated by the average difference when the interaction AB and C are at the same level, and when AB and C are at different levels,

$$\widehat{ABC} = \frac{1}{4}\left[(y_2 + y_3 + y_5 + y_8) - (y_1 + y_4 + y_6 + y_7)\right].$$

In a factorial experiment, the response y can be expressed by a linear regression model. For a 2^3 factorial design this model is given by,

$$\begin{aligned} y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_{12} x_1 x_2 + \beta_{13} x_1 x_3 \\ + \beta_{23} x_2 x_3 + \beta_{123} x_1 x_2 x_3 + \varepsilon. \end{aligned} \tag{2.18}$$

Here, β's are the regression parameters to estimate, and ε is the error term. The error terms ε_i are assumed to be independent with $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, $i = 1, 2, \ldots, 8$. The variables x_1, x_2 and x_3 represent the factors A, B and C coded to -1 and 1 such that the factor columns are orthogonal. The main effect of a factor is a measure of how much the expected response will change when the factor is moved from low to high level, that is from -1 to 1. A regression coefficient on the other hand, measures how much the expected response will change with a one-unit change in the factor, from 0 to 1. Thus, the regression coefficients in Equation 2.18 are half of their factor's corresponding effects. A transformation of the factors can be done to obtain coded variables with levels 1 and -1. For factor A this is done by,

$$x_1 = \frac{A - (A_h + A_l)/2}{(A_h - A_l)/2},$$

where the low and high level of A are represented by A_l and A_h respectively.

One of the three principles of DOE is replication. One replicate of the experiment results in two response values with the same expected value for each level combination. Then, a model independent variance can be estimated. Let two observed response values for the first level combination be denoted by y_{11} and y_{12}. The variance of the observations is then estimated by,

$$\begin{aligned}\sum_{j=1}^{2}(y_{1j} - \bar{y}_1)^2 &= \left(y_{11} - \frac{y_{11} + y_{12}}{2}\right)^2 + \left(y_{12} - \frac{y_{11} + y_{12}}{2}\right)^2 \\ &= \left(\frac{y_{11} - y_{12}}{2}\right)^2 + \left(\frac{y_{12} - y_{11}}{2}\right)^2 = \frac{(y_{11} - y_{12})^2}{2}.\end{aligned}$$

Such estimates are obtained for each level combination, i.e., 8 in a 2^3 factorial design. The final estimate of the variance is computed by averaging over all estimates. More generally, if an experiment is replicated $(m-1)$ times, each level combination has m observed response values, an estimate for the variance of the observation i is defined by

$$\hat{\sigma}_i^2 = \sum_{j=1}^{m} \frac{\left(Y_{ij} - \bar{Y}_i\right)^2}{m-1}, \quad i = 1, 2, \ldots, n,$$

and these can be averaged to obtain a final estimate, (Tyssedal n.d.).

2.4.3 Two Level Fractional Factorial Design

If the number of factors, k, becomes large, then the number of runs needed in a 2^k factorial design increases quickly, and the full design may be infeasible to perform. For that reason, running a fraction of the full factorial design can be beneficial to save time and computational effort. A fraction of a two-level design with k factors is referred to as a 2^{k-p} fractional factorial design, where 2^{k-p} is the number of level combinations to observe. Here, the focus will be on a half fraction of the 2^k factorial design, often called a 2^{k-1} fractional factorial design. Such a design is created with one of the factors' columns in the design matrix being defined by an interaction column not containing that factor. Pairs like that are called aliases, and independent estimation of their linear effects is not possible. In general, the main effects and lower order interactions are more significant compared to higher order interactions which often can be negligible. A design's resolution is defined by the number of letters in the shortest term of the defining relation. Constructing a $2^k - 1$ fractional factorial design with the k'th factor's column in the design matrix defined by the column of the interaction of the other $k-1$ factors, gives a design with the highest feasible resolution, (Goos 2002).

Now, let $k = 5$ and consider a 2^{5-1} fractional factorial design with factors A, B, C, D and E. Following the design described above, factor E should then be defined by the four-factor interaction $ABCD$, which is called the generator. Table 2.2 shows the design of a 2^{5-1} fractional factorial experiment with one replicate. As E has the same sign as $ABCD$, the design's defining relation is defined as

$$I = ABCDE,$$

where I is called the identity element. This design is of resolution V since a main effect is aliased with a four-factor interaction, and two-factor interactions are aliased with three-factor interactions. In general, a resolution of a fractional factorial design either greater than or equal to V, assures that estimation of all main and two-factor interaction effects is feasible, given that interactions of order three or higher are negligible, (Goos 2002).

Table 2.2: Design of a 2^{5-1} fractional factorial design of one replicate with five factors A, B, C, D and E.

Experiment No.	A	B	C	D	E = ABCD	Level code
1	-1	-1	-1	-1	1	*e*
2	1	-1	-1	-1	-1	*a*
3	-1	1	-1	-1	-1	*b*
4	1	1	-1	-1	1	*abe*
5	-1	-1	1	-1	-1	*c*
6	1	-1	1	-1	1	*ace*
7	-1	1	1	-1	1	*bce*
8	1	1	1	-1	-1	*abc*
9	-1	-1	-1	1	-1	*d*
10	1	-1	-1	1	1	*ade*
11	-1	1	-1	1	1	*bde*
12	1	1	-1	1	-1	*abd*
13	-1	-1	1	1	1	*cde*
14	1	-1	1	1	-1	*acd*
15	-1	1	1	1	-1	*bcd*
16	1	1	1	1	1	*abcde*

2.4.4 Adding Center Points to a 2^k Design

A two-level factorial design assumes the factor effects to be linear and is therefore valuable to capture significant main effects and interactions. In fact, a general first-order model including interaction terms given by

$$y = \beta_0 + \sum_{j=1}^{k} \beta_j x_j + \sum_{i<j} \sum_{=2}^{k} \beta_{ij} x_i x_j + \varepsilon \tag{2.19}$$

is able to represent some curvature in the response function. This is a result of the interaction terms $\beta_{ij} x_i x_j$ which induce a twist of the plane, (Myers and Montgomery 2002). However, if quadratic effects are present, the curvature captured by Equation 2.19 is not adequate, and a second-order response surface model

$$y = \beta_0 + \sum_{j=1}^{k} \beta_j x_j + \sum_{i<j} \sum_{j=2}^{k} \beta_{ij} x_i x_j + \sum_{j=1}^{k} \beta_{jj} x_j^2 + \varepsilon \tag{2.20}$$

should be considered. One method to test if the second-order model in Equation 2.20 is more suitable, is to add center points to the 2^k design. Suppose n_C observations at the center point are added to the 2^k design. Let $\bar{y}_C$ be the average response of the n_C center point runs, and $\bar{y}_F$ be the average response of the factorial point runs. If the response of the center points lies near the plane constructed by the factorial points, then the difference $\bar{y}_F - \bar{y}_C$ will be small. However, if the distance between the center point responses and the plane through the factorial points is large, then the difference $\bar{y}_F - \bar{y}_C$ is large. This may mean that the curvature in the response function is quadratic, and a second-order response surface model should be used, (Myers and Montgomery 2002). Analysis of variance can be used to determine if the difference is significantly large.

Let m be the number of distinct points in the design, and r_i is the number of observations for level combination i. Then, for each point in the design, the residuals are given by

$$y_{ij} - \hat{y}_i = (y_{ij} - \bar{y}_i) - (\hat{y}_i - \bar{y}_i),$$

where $i = 1, 2, \ldots m$ and $j = 1, 2, \ldots r_i$. By inserting this expression into the equation for residual sum of squares yields

$$
\begin{aligned}
SSE &= \sum_{i=1}^{m}\sum_{j=1}^{r_i}(y_{ij}-\hat{y}_i)^2 \\
&= \sum_{i=1}^{m}\sum_{j=1}^{r_i}(y_{ij}-\bar{y}_i)^2 + \sum_{i=1}^{m} r_i(\bar{y}_i-\hat{y}_i)^2 \\
&= SS_{PE} + SS_{LOF}.
\end{aligned}
$$

Meaning, the residual sum of squares can be written as the sum of the pure error sum of squares and the lack of fit sum of squares. Suppose p terms are fitted to the data, and n is the total number of runs. The residual sum of squares then has $(n-p)$ degrees of freedom. The pure error sum of squares has $(n-m)$ degrees of freedom, thus the lack of fit sum of squares has the resulting $(m-p)$ degrees of freedom.

In a design matrix, the low level of a factor is coded as -1 and the high level as 1, while center points are coded as 0. A result of this is that the center points have no impact on the computation of the contrasts, which further results in two objectives of the addition of center points. The first is based on the variation of the response in the center points, which can be used to estimate the pure error, (Lujan-Moreno et al. 2018). The second objective is a test for lack of fit, related to the difference $\bar{y}_F - \bar{y}_C$. To perform a test for lack of fit, the F-statistic

$$
F = \frac{SS_{LOF}/(m-p)}{SS_{PE}/(n-m)}
$$

can be used. The null and alternative hypotheses tested by the lack of fit test is

$$
\begin{aligned}
H_0: &\quad \sum_{j=1}^{k}\beta_{jj} = 0 \\
H_1: &\quad \sum_{j=1}^{k}\beta_{jj} \neq 0.
\end{aligned}
$$

Thus, if the result of the F-test is to reject the null hypothesis, the lack of fit is significant and there is need for a response surface model with quadratic terms. While if the value of the F-statistic is small, there is not sufficient evidence to conclude that quadratic terms are needed and a model with main effects and interactions may be suitable.

2.4.5 Response Surface Methodology

Response Surface Methodology (RSM) is a set of statistical and mathematical techniques whose objective is to develop, analyze and optimize processes. It is an efficient method convenient for hyperparameter optimization, specifically in the case where several hyperparameters are assumed to affect the response of interest. The relationship between the independent input variables $x_1, x_2, \dots, x_k$ and the response y can be represented by $y = f(x_1, x_2, \dots, x_k) + \varepsilon$, where f is the response function, and ε represents the error not accounted for in f. In general, the relationship is visualized graphically through response surface and contour plots. The true form of f is unknown, and it must therefore be approximated. If the region of interest is relatively small, a first-order model with interaction terms in Equation 2.19 may be adequate, (Myers and Montgomery 2002). While if the curvature is stronger than represented by the interaction terms in the first-order model, the second-order response surface model in Equation 2.20 is more appropriate.

Applications of RSM are most often done in a sequential manner. Firstly, a screening experiment is done to identify the most influential variables. This is typically done using DOE, with the assumption that a first-order model is suitable. With the most important variables established, the response surface study can start. In the first phase, the current variable levels must be considered to determine if these lead to a response value near the optimum, or if the process should be moved to a new region closer to the optimum. In the case of the latter, the method of steepest ascent, which is an optimization technique, can be used. Once the process is near the optimum, the next phase is to approximate the response function in a smaller region. As the response function often is curved in the area around an optimum, a second-order model is a good approximation. Phase one and two of the response surface study may be repeated until the optimum is reached.

Method of Steepest Ascent

When a two-level design is executed, it is necessary for the practitioner to determine the low and high levels for each factor. These levels will therefore often be determined based on educated guesses, and it may be beneficial to move into a new region where the response is improved before the conduction of experiments is continued. The method of steepest ascent is a first-order gradient based optimization technique, (Myers and Montgomery 2002), with the objective of exploring a new region where the response is improved, and not to discover which variable configuration that obtains an optimal response. The method of steepest ascent proceeds sequentially along the direction of maximum increase in the response. Figure 2.5 illustrates the path of steepest ascent in the case with two variables.

Consider a first-order regression model fitted with the use of an orthogonal design,

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k.$$

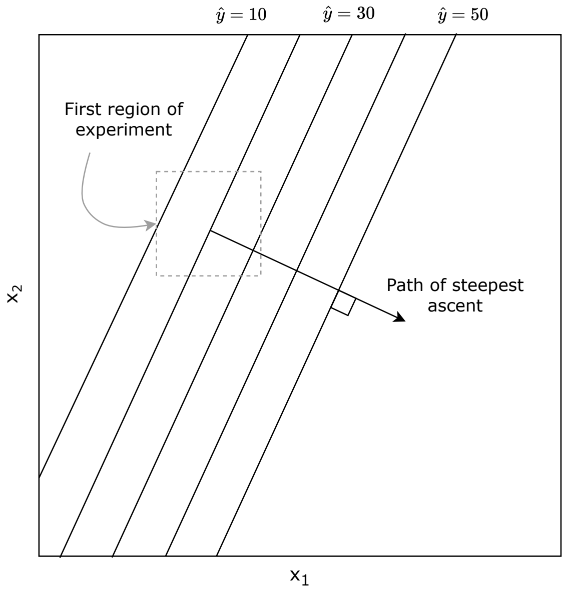


Figure 2.5: Illustration of the path of steepest ascent, where the contours are the expected response.

Given a fixed distance r from the center of the design, the path of steepest ascent is the direction of where the point which provides the maximum expected response $\hat{y}$ lies. Mathematically, this is expressed in the following way

$$\max_{(x_1,x_2,\ldots,x_k)} \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \quad \text{subject to} \quad \sum_{i=1}^{k} x_i^2 = r^2. \tag{2.21}$$

The center point of the design is $(0, 0, \ldots, 0)$, thus the constraint resembles a sphere with radius r. The optimization problem in Equation 2.21 is solved by using Lagrange multipliers. The maximum of the Lagrange function

$$L = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k - \lambda \left(\sum_{i=1}^{k} x_i^2 - r^2 \right),$$

is obtained by calculating the partial derivatives of L with respect to x_j,

$$\frac{\partial L}{\partial x_j} = \beta_j - 2\lambda x_j, \qquad j = 1, 2, \ldots, k.$$

By setting the expression of the partial derivatives equal to zero and solving for x_j, yields a coordinate of x_j in the direction of steepest ascent,

$$x_j = \frac{\beta_j}{2\lambda}, \qquad j = 1, 2, \ldots, k.$$

The final coordinate of steepest ascent is then

$$x_1 = \rho\beta_1, \quad x_2 = \rho\beta_2, \ldots x_k = \rho\beta_k,$$

where $\rho = 1/2\lambda$ is a positive constant of proportionality, which is decided by the practitioner and regulates the distance from the center point. Thus, the distance x_j is moved, is proportional to the absolute value of the regression coefficient β_j, and the direction is determined by the coefficient sign. Experiments are conducted along the path until there is no more improvement in the observed response value. The location of the maximum response value along the path is then the base for a second experimental region. A new first-order design for estimating main effects and interactions is then implemented. Often it is favorable to add center point to the design to test for lack of fit. If the test is not rejected, the new model is used to find a second path of steepest ascent, referred to as a mid-course correction. After conducting some experiments, the improvement will likely be limited. Thus, a new base is obtained to conduct a more refined experiment and optimization process, (Myers and Montgomery 2002). Usually, this is done by modifying the design to fit a second-order model.

Central Composite Design - Experimental Design for Fitting Second-Order Models

The second-order model is given in Equation 2.20 and contains $1 + 2k + k(k-1)/2$ parameters. Among the designs to fit second-order models, central composite design (CCD) is the most popular. A CCD involves three main components. The first is a two-level factorial, alternatively a two-level fraction preferably of resolution V, used to estimate the linear terms and two-factor interactions in a variance-optimal way. The second component is the n_c center runs. These enable a form of internal estimate of error, called the pure error, along with recognizing existence of curvature and contribute to the estimation of quadratic terms. The main contributor in estimating the quadratic terms however is the $2k$ axial points, which are the last component.

There are two important decisions to consider when constructing a CCD; the axial distance α, and the number of center points n_c. The choice of the former depends on both the region of operability and interest. The latter tends to influence the distribution of the scaled prediction variance $N \cdot \mathrm{Var}\left[\hat{y}(\mathbf{x})\right]/\sigma^2$ in the region of interest. The axial distance is often set to be between 1.0 and $\sqrt{k}$, where $\alpha = 1.0$ places all axial points on the face of the cube or hypercube, while $\alpha = \sqrt{k}$ places all axial points on a sphere. Table 2.3 shows a CCD with three variables, one center run, and axial distance given by α. For a spherical region, each factor has five levels; the center point 0, low and high level, -1 and 1, from the factorial and the axial points $-\sqrt{k}$ and $+\sqrt{k}$. A CCD for $k = 3$ and $\alpha = \sqrt{3}$ is visualized in Figure 2.6.

Table 2.3: A central composite design with variables x_1, x_2 and x_3, axial distance α and one center run.

Experiment No.	x_1	x_2	x_3
1	-1	-1	-1
2	1	-1	-1
3	-1	1	-1
4	1	1	-1
5	-1	-1	1
6	1	-1	1
7	-1	1	1
8	1	1	1
9	-α	0	0
10	α	0	0
11	0	-α	0
12	0	α	0
13	0	0	-α
14	0	0	α
15	0	0	0

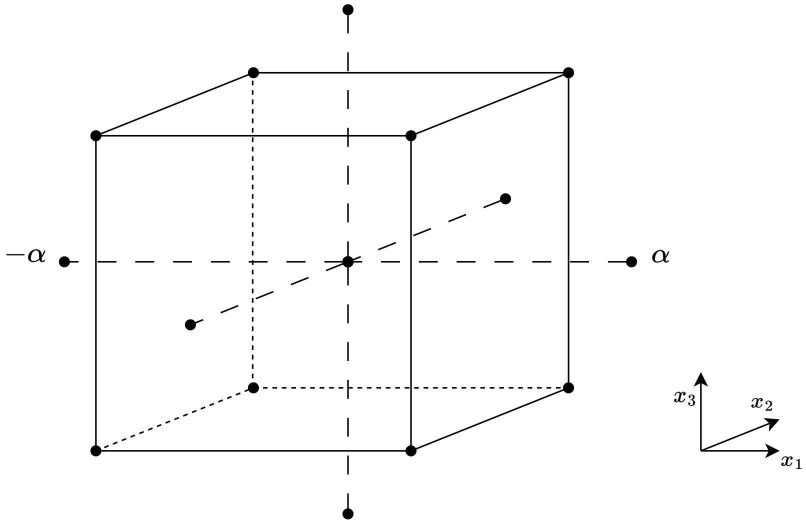


Figure 2.6: Illustration of a central composite design with three variables, x_1, x_2 and x_3, and $\alpha = \sqrt{3}$.

To guarantee that $\hat{y}(\mathbf{x})$ is a good estimator for $\mathrm{E}[y(\mathbf{x})]$ in the whole region of interest, the scaled prediction variance should be reasonably stable. For that reason, the idea of design rotatability was introduced by Box and Hunter (1957). It states that a rotatable design is one where two locations at the same distance from the design center have the same value of $N \cdot \mathrm{Var}\left[\hat{y}(\mathbf{x})\right]/\sigma^2$. The objective of this idea is to produce stability in the way that $N \cdot \mathrm{Var}\left[\hat{y}(\mathbf{x})\right]/\sigma^2$ is constant on a sphere. Fitting a second-order model with a spherical design leads $\mathbf{X}^T\mathbf{X}$ to be singular, which further causes $N \cdot \mathrm{Var}\left[\hat{y}(\mathbf{x})\right]/\sigma^2$ to be infinite. By adding enough center runs, the stability of $N \cdot \mathrm{Var}\left[\hat{y}(\mathbf{x})\right]/\sigma^2$ becomes reasonable in the design region, (Myers and Montgomery 2002). Thus, the design for a spherical region is either rotatable or near rotatable, and the use of three to five center runs is recommended in practice.

Analysis of Second-Order Response Surfaces

Experimental data with presence of curvature can be described by the second-order response surface model in Equation 2.20, which is both flexible and easily fitted using designs like CCD. Through second-order analysis of the response surface, the stationary point can be found, and the point can be classified. The stationary point is where all the partial derivatives of $\hat{y}$ are equal to zero, i.e., the solution of

$$\frac{\partial \hat{y}}{\partial x_1} = \frac{\partial \hat{y}}{\partial x_2} = \dots \frac{\partial \hat{y}}{\partial x_k} = 0.$$

By analyzing the plot of the contours of the second-order system, a stationary point can be identified as a maximum, minimum or saddle point. These depend on the model coefficients which are estimates of the β's in Equation 2.20. Consequently, these contours are based on estimated responses. Let the fitted second-order response surface model be written as

$$\hat{y} = b_0 + \mathbf{x}^T\mathbf{b} + \mathbf{x}^T\hat{\mathbf{B}}\mathbf{x}, \tag{2.22}$$

where b_0 represent the estimate of the intercept, $\mathbf{x}^T = [x_1, x_2, \dots, x_k]$, $\mathbf{b}^T = [b_1, b_2, \dots, b_k]$ is the estimate of the linear coefficients, and the second-order coefficients are represented by

$$\hat{\mathbf{B}} = \begin{bmatrix} b_{11} & b_{12}/2 & \dots & b_{1k}/2 \\ & b_{22} & \dots & b_{2k}/2 \\ & & \ddots & \vdots \\ \text{sym.} & & & b_{kk} \end{bmatrix}.$$

The location of the stationary point $\mathbf{x}_s$ is found by differentiating Equation 2.22 with respect to $\mathbf{x}$,

$$\frac{\partial \hat{y}}{\partial \mathbf{x}} = \mathbf{b} + 2\hat{\mathbf{B}}\mathbf{x}.$$

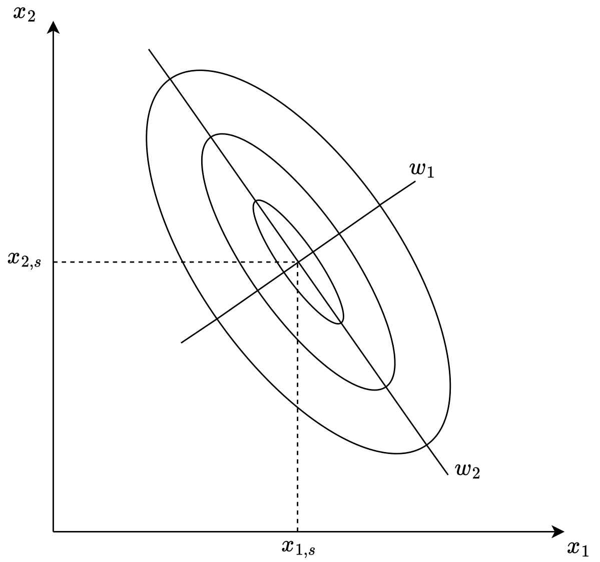


Figure 2.7: Canonical form of the second-order model with two variables, x_1 and x_2. The stationary point is represented by $(x_{1,s}, x_{2,s})$.

Then, the derivatives are set equal to zero and solved for $\mathbf{x}_s$ to obtain the system's stationary point,

$$\mathbf{x}_s = -\frac{1}{2}\hat{\mathbf{B}}^{-1}\mathbf{b}.$$

The corresponding estimated response is given by

$$\hat{y}_s = b_0 + \frac{1}{2}\mathbf{x}_s^T\mathbf{b}.$$

By considering the signs of the eigenvalues of matrix $\hat{\mathbf{B}}$, the character of the stationary point can be determined. First, Equation 2.22 must be transformed into the canonical form, where the new center is the stationary point, and the axes are rotated to coincide with the principal axes of the contour system. Figure 2.7 shows an illustration of the second-order model with two variables in canonical form. The canonical translation yields

$$\hat{y} = \hat{y}_s + \sum_{i=1}^{k} \lambda_i w_i^2, \tag{2.23}$$

where $\hat{y}_s$ is the estimated response at the stationary point, and $\lambda_1, \lambda_2, \ldots, \lambda_k$ are the eigenvalues of $\hat{\mathbf{B}}$. The variables $w_1, w_2, \ldots, w_k$ are given by

$$\mathbf{w} = \mathbf{P}^T(\mathbf{x} - \mathbf{x}_s)$$

and are called canonical variables. Here $\mathbf{P}$ is a $k \times k$ matrix where the columns correspond to the normalized eigenvectors related to the eigenvalues of $\hat{\mathbf{B}}$. Let $\mathbf{\Lambda}$ be the diagonal matrix of the eigenvalues of $\hat{\mathbf{B}}$, then

$$\mathbf{P}^T\hat{\mathbf{B}}\mathbf{P} = \mathbf{\Lambda}.$$

The signs of the eigenvalues of $\hat{\mathbf{B}}$ determine the nature of the stationary point $\mathbf{x}_s$, (Myers and Montgomery 2002). In the case of all $\lambda_1, \lambda_2, \ldots, \lambda_k$ are negative, then the stationary point is a maximum. If all eigenvalues are positive, the stationary point is a minimum. While if the eigenvalues have both negative and positive signs, the stationary point is a saddle point. The relative magnitude of the eigenvalues indicates the sensitivity of the response system, e.g., if the absolute values are close to zero there is a plane with approximately the same response value, while large absolute values may indicate there is solely one point with the optimal response value. In the case of a saddle point, new experiments should be conducted along a linear path in the direction of increasing or decreasing response, depending on the objective. The eigenvectors can be used to suggest which direction to follow.

2.5 Explainable AI

For simple models, the original model is its own best explanation. However, this does not hold for more complex models as they are difficult to understand. Machine learning models are often represented as a black box. The model processes the input data and returns a prediction, but the reason for the prediction is unknown. However, in most real-world cases it is preferable to understand how the unique features affect the outcome. Explainable AI are different methods and processes which make it possible for humans to gain knowledge of the relation between the input data and the predictions.

2.5.1 LIME

Local interpretable model-agnostic explanations (LIME) is a technique to interpret single predictions of a machine learning model by approximating a local surrogate model. Two important criterion of LIME is interpretability and local fidelity. The idea of the latter is that although it can be impossible to obtain a globally reliable explanation, it is sufficient with local reliability to obtain useful information. Thus, the aim is to find an interpretable model which is locally reliable.

Let $f(x)$ be the machine learning model being explained for instance x, and $g(x)$ the explanation model which is in the class of potential interpretable models $g \in G$. These potential models can have various levels of complexity. In general, less complex models are easier to interpret. Denote $\Omega(g)$ as the complexity of the explanation model $g \in G$. Let $\mathcal{L}$ be the loss function which measure how faithful g is at approximating the prediction of f in the nearby area defined by π_x, (Ribeiro et al. 2016). The explanation of instance x obtained by LIME is given by,

$$\arg\min_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g). \tag{2.24}$$

As the M in LIME implies, the explanation g needs to be model-agnostic. Thus, the loss function $\mathcal{L}(f, g, \pi_x)$ should be minimized while avoiding any assumptions about f. Samples are drawn uniformly at random from non-zero elements around x, and then perturbed. Let Z be the data set of perturbed samples z. Black box predictions $f(z)$ are obtained as labels for the explanation model g. Further, the perturbed samples are weighted according to their proximity to x, which is the instance of interest, (Ribeiro et al. 2016). Closer sample instances are weighted higher. Lastly, an interpretable model is trained on the data set Z and used to explain the prediction of instance x.

There are several advantages of LIME. One of them is the fact that the local interpretable model given by LIME does not depend on the underlying machine learning model used. This gives flexibility in the way that the model which makes the predictions may be changed to improve performance, while the same local model can be used to interpret the predictions. Another advantage is that LIME allows the use of other features than those used to train the underlying machine learning model. This makes it possible to have a model trained on complex and less interpretable features, e.g., components of principal component analysis, while the local interpretable model is trained on the original, more interpretable features. Of course, the interpretable features are assumed derived from the same data set as those used to train the machine learning model. In addition, LIME can be applied to tabular data, text, and images, which few other methods for variable explanation can.

Despite the advantages, there are also some disadvantages related to LIME. In the case of applying LIME to tabular data, defining the correct neighborhood is a big unsolved problem. In addition, the explanations can be very unstable. For instance, Alvarez-Melis and Jaakkola (2018) showed great variation in the explanation of two nearby points in a simulated study. Also, the explanations made by LIME can be manipulated to hide biases, which can make LIME explanations less trustworthy.

2.5.2 Shapley Values

Another method to explain the predictions made by machine learning models is Shapley values. This is a concept from co-operative game theory originally used to assign payouts to players according to their contribution towards the total payout. In the case of prediction explanation by machine learning models, the players are replaced with features and the total payout by the prediction. The Shapley value of a feature is given by the average expected marginal contribution to the prediction, after all feature combinations have been considered, (Aas et al. 2021). Thus, a feature's Shapley value reflects how much the prediction made by the model is affected by adding the relevant feature.

Let F be the set of all features. The Shapley value method retrains the model on all possible feature coalitions $S \subseteq F = \{1, \ldots, F\}$ and based on the effect of including the relevant feature has on the prediction, the feature is given an importance value. The effect of a feature is computed by training two models, one including the relevant feature and another excluding it, $v(S \cup \{i\})$ and $v(S)$ respectively, where $v()$ is the characteristic function. The predictions made by the two models are then compared on the values of the input features in the relevant subset, $v(S \cup \{i\}) - v(S)$. This difference is computed for all permutations of all possible coalitions to capture the effect of withholding a feature. The Shapley value is then given by,

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!\,(F - |S| - 1)!}{F!} \left[v(S \cup \{i\}) - v(S) \right], \quad i = 1, \ldots, F, \tag{2.25}$$

which is the weighted average of all differences, (Lundberg and Lee 2017).

Consider an example with three features, $F = \{1, 2, 3\}$. For three features there are eight different subsets, where the model output of each of them are given by,

$$\begin{gathered} v(\emptyset) = 0, \quad v(\{1\}) = 0, \quad v(\{2\}) = 0, \quad v(\{3\}) = 0 \\ v(\{1,2\}) = 60, \quad v(\{1,3\}) = 40, \quad v(\{2,3\}) = 70, \quad v(\{1,2,3\}) = 100. \end{gathered}$$

Table 2.4: The six different coalitions with three features with corresponding computed differences and Shapley values denoted by ϕ.

	1	2	3
$1 \leftarrow 2 \leftarrow 3$	0	60	40
$1 \leftarrow 3 \leftarrow 2$	0	60	40
$2 \leftarrow 1 \leftarrow 3$	60	0	40
$2 \leftarrow 3 \leftarrow 1$	30	0	70
$3 \leftarrow 1 \leftarrow 2$	40	60	0
$3 \leftarrow 2 \leftarrow 1$	30	70	0
Sum	160	250	190
ϕ	26.67	41.67	31.67

Table 2.4 shows the six different coalitions with the respective Shapley values. Consider the calculation of the predicted differences in the first row. With only feature 1 present, the predicted value is 0 from $v(\{1\}) = 0$. The effect of adding feature 2 is equal to 60, which is computed by the difference $v(\{1,2\}) - v(\{1\}) = 60 - 0$. Adding feature 3 yields an effect of 40, since

$v(\{1,2,3\}) - v(\{1,2\}) = 100 - 60$. The Shapley values are computed by weighting the computed differences according to Equation 2.25, which for feature 1 yields

$$\begin{aligned}\phi_1 &= \frac{1}{3}\left[v(\{1,2,3\}) - v(\{2,3\})\right] + \frac{1}{6}\left[v(\{1,2\}) - v(\{2\})\right] \\ &+ \frac{1}{6}\left[v(\{1,3\}) - v(\{3\})\right] + \frac{1}{3}\left[v(\{1\}) - v(\emptyset)\right] \\ &= \frac{1}{3}(100-70) + \frac{1}{6}(60-0) + \frac{1}{6}(40-0) + \frac{1}{3}(0-0) \\ &= 26.67.\end{aligned}$$

There are two main advantages with Shapley values. The first is that it is the only explanation method with a mathematically solid theory. Secondly, it guarantees that the difference between the prediction and the average prediction is fairly distributed across the features. However, the complexity of the method is a great drawback. The number of possible coalitions of the feature values grows exponentially. Thus, in most real-world problems, the exact solution is not feasible. Because of this, several methods to approximate the Shapley values have been proposed. One of them is SHAP.

2.5.3 SHAP

SHAP (SHapley Additive exPlanations) estimates the Shapley values from co-operative game theory to explain individual predictions. SHAP was proposed by Lundberg and Lee (2017) as a unified approach to interpret predictions made by machine learning models.

Kernel SHAP is one method to estimate Shapley values which can be regarded as two distinct parts; (1) approximating the Shapley values in Equation 2.25, and (2) estimating $v(S)$. Consider part (1) and assume $v(S)$ to be known. The Shapley values can be defined as the optimal solution to the weighted least squares problem of minimizing

$$\sum_{S \subseteq F}\left[v(S) - \left(\phi_0 + \sum_{i \in S}\phi_i\right)\right]^2 k(F, |S|) \tag{2.26}$$

with respect to $\phi_0, \ldots, \phi_F$, where the Shapley kernel weights are

$$k(F,S) = (F-1)/\left[\binom{F}{|S|}|S|(F-|S|)\right].$$

Let $\mathbf{Z}$ be a $2^F \times (F+1)$ binary matrix which represents all coalitions of including/excluding the different features, where the first column contains all 1's, while entity $i+1$ in row j is 1 if feature

i is present in coalition j, and 0 if excluded. Let $v(S)$ be contained in the vector $\mathbf{v}$, and $\mathbf{W}$ be a $2^F \times 2^F$ diagonal matrix which contains $k(F, |S|)$. For both $\mathbf{v}$ and $\mathbf{W}$, S is the feature coalition of the corresponding row in $\mathbf{Z}$. The weighted least squares problem in Equation 2.26 can be written as

$$(\mathbf{v} - \mathbf{Z}\boldsymbol{\phi})^T \mathbf{W} (\mathbf{v} - \mathbf{Z}\boldsymbol{\phi}), \tag{2.27}$$

where $\boldsymbol{\phi}$ is the vector containing $\phi_0, \ldots, \phi_F$, and the solution is

$$\boldsymbol{\phi} = (\mathbf{Z}^T \mathbf{W} \mathbf{Z})^{-1} \mathbf{Z}^T \mathbf{W} \mathbf{v}. \tag{2.28}$$

Instead of using the whole feature set F, Kernel SHAP samples a subset D of F according to a probability distribution which follows the Shapley weighting kernel, and use only the rows of $\mathbf{Z}$ and elements of $\mathbf{v}$ corresponding to the subset, $\mathbf{Z}_D$ and $\mathbf{v}_D$. The sampled subsets are equally weighted in the new least squares problem. Thus, an approximation to Equation 2.28 is obtained,

$$\boldsymbol{\phi} = (\mathbf{Z}_D^T \mathbf{W}_D \mathbf{Z}_D)^{-1} \mathbf{Z}_D^T \mathbf{W}_D \mathbf{v}_D. \tag{2.29}$$

The next part is to estimate $v(S)$. Consider a predictive model $f(\mathbf{x})$ trained on a training set $\{y^i, \mathbf{x}^i\}_{i=1,\ldots,n}$, and let $\mathbf{x} = \mathbf{x}^*$ be a specific feature vector. For a subset S, the contribution value is given by

$$v(S) = \mathrm{E}\left[f(\mathbf{x}) | \mathbf{x}_S = \mathbf{x}_S^*\right].$$

Let $\bar{S}$ be the complement of S for the purpose of writing $\mathbf{x}_{\bar{S}}$ as the part of $\mathbf{x}$ which is not contained in $\mathbf{x}_S$. The right-hand side can then be computed by

$$\begin{aligned}
\mathrm{E}\left[f(\mathbf{x}) | \mathbf{x}_S = \mathbf{x}_S^*\right] &= \mathrm{E}\left[f(\mathbf{x}_{\bar{S}}, \mathbf{x}_S) | \mathbf{x}_S = \mathbf{x}_S^*\right] \\
&= \int f(\mathbf{x}_{\bar{S}}, \mathbf{x}_S^*) p(\mathbf{x}_{\bar{S}} | \mathbf{x}_S = \mathbf{x}_S^*) d\mathbf{x}_{\bar{S}}.
\end{aligned}$$

The conditional distribution $p(\mathbf{x}_{\bar{S}} | \mathbf{x}_S = \mathbf{x}_S^*)$ is seldom known, (Aas et al. 2021). The Kernel SHAP assumes the features to be independent, and thus avoids this problem. The integral can then be approximated with Monte Carlo integration.

Chapter 3

The Data Set

The data set used in this thesis is provided by Sparebank 1 Kreditt AS. It was retrieved in February 2023 and consists of four data tables; *passive_table*, *application_table*, *purchase_history_table*, and *purchase_history_segment_table*. Table 3.1 gives a summary of the four tables. All tables contain different information about the customers, and are connected through the variable BK_ACCOUNT_ID, which is the personal account number for a customer. Each BK_ACCOUNT_ID represents a customer who at some point stopped using their credit card and has been inactive for at least six months. The response AktivEtterPassiv denotes if the given customer has become an active credit card user after the period of being passive. An active customer is defined as an individual who uses their credit card. An inactive customer is defined as someone who has not used their card the last month, and a passive customer is defined as someone who has been inactive for at least six months.

Table 3.1: An overview of the data tables with description and number of explanatory variables.

Data table	Description	Number of variables
Passive	Information about the customers' credit card use and transactions	10
Application	Information about the customers' application data	21
Purchase history	Information about the customers' payments	8
Purchase history segment	Information about the customers' purchase history split into segments	21

This chapter will introduce the explanatory variables and response before the data pre-processing performed will be described. Finally, visualizations of some explanatory variables will be presented to improve the understanding of their impact on the response.

3.1 Explanatory Variables and Response

The data set contains 60 explanatory variables. Of these, some variables contain information of credit limit, age, when the card was last used, and the customer's marital status at the time of application. All variables are listed together with explanations in Appendix A. The passive_table has 10 explanatory variables, in addition to the response. Among these are variables describing when the credit card was used first and last, gender and type of credit card. This table contains longitudinal data of 27485 unique BK_ACCOUNT_ID's with a total of 260959 observations. The number of observations for one customer differs, but consistent for them all is that at the time of the first observation for each customer, it has been at least six months since last time they used their credit card. The next observations are monthly updates of the customer's credit card use and transactions, until they become active. If a customer did not become active, the last observation is from the same month as the data was retrieved by Sparebank 1. Transactions can happen while the customer is passive, even though they do not use their card. Some customers become active and then stopped using their card for at least six months. In such cases, there are no observations after they were active until they have been inactive for six months, then there are monthly observations until they once again start to use their card, or the month of data retrieval is reached. This leads to some customers having several passive periods.

The application_table contains application data of 20933 unique BK_ACCOUNT_ID's, and 21 explanatory variables. These include information of the customer from the time they applied for the credit card, such as amount of student loan, type of employment and number of children. The purchase_history_table contains payment history for 5081 unique BK_ACCOUNT_ID's from the last twelve months before the customer became passive and consists of 8 explanatory variables. These include payments to external repayment loan accounts, external collection accounts and external credit card accounts. This table contains a total of 5475 observations, as some customers have several periods of being passive and can thus have one observation before each passive period. And lastly, purchase_history_segment_table contains monthly longitudinal purchase history for 28375 unique BK_ACCOUNT_ID's which is divided into different segments; airline, food stores, hardware, etc. This table has a total of 174492 observations and 21 different segments.

The response AktivEtterPassiv denotes whether a given customer has become active after the passive period or not. Thus, it is a binary response which equals 1 if the customers once again started to use their card, and 0 if they are still categorized as a passive customer.

3.2 Data Pre-Processing

Both the passive_table and purchase_history_segment_table contain longitudinal data. To model data on this form can be challenging. Therefore, data pre-processing is necessary to transform the data into a more convenient form; one row per BK_ACCOUNT_ID. This was done using the software `R` in R studio.

The aim is to compare the performance of models when predicting one, three, six and twelve months ahead. Four data frames were created, one for each situation. Let us consider the preparation of the *one_month_ahead* data frame. For each BK_ACCOUNT_ID in the passive_table; first, the relevant observations according to the number of months to predict ahead, called the valid observations, were found for the one month ahead prediction. These are all observations but the last. The last observation of the customer reflects if they have become active or not and is thus only used to set the value of the response. Next, information from the valid observations were extracted by adding new variables. Some examples are the number of passive periods, number of transactions, and number of months between the first and last time the credit card was used. Several of the variables in the passive_table were just dates. New variables were added to describe the number of months between events, instead of the date of the event as dates are not informative for a model. For the purchase_history_table, only the last valid observation for each BK_ACCOUNT_ID was kept. This was done by comparing the dates of these observations to the date of the customer's last valid observation in the passive_table. Each observation in purchase_history_table is a summary of the last twelve months based on the date of the observation. Including observations further back in time is assumed to add little to no additional information. Similarly for the purchase_history_segment_table, only the last valid observation for each BK_ACCOUNT_ID was kept, as these observations also are summaries of the last twelve months. Lastly, these three modified tables and the application_table were merged into the one_month_ahead data frame.

A similar procedure was done to the *three_months_ahead*, *six_months_ahead* and the *twelve_months_ahead* data frames, with updated number of valid observations depending on the number of months to predict ahead. As the number of observations for each BK_ACCOUNT_ID in the passive_table differs, not all customers have enough observations to be included in all four data frames. Table 3.2 shows an overview of the number of instances in each data frame, as well as the portion of customers that are categorized as active at the end of the (last) period.

After the transformation was done, the BK_ACCOUNT_ID variable was removed, as it did not contain any relevant information for the models, and missing values could be handled. Indicator variables were added to specify which customers lack observations from the application_table, purchase_history_table or purchase_history_segment_table. Some customers have application data, but do not have values for DebtRegisterNum/ DebtRegisterIELA or SumAvailable, which are three of the variables in the application_table. Thus, indicator variables are added to specify cases with application data where one of these is missing. Missing values in variables regarding purchase history or payments are set to zero, assuming it has not occurred. Missing values in the variable NoOfChildren is also set to zero, as these are assumed to not have children. Missing values in

Table 3.2: An overview of the four data frames with total number of observations, number of observations where AktivEtterPassiv = 1, and percentage of observations where AktivEtterPassiv = 1.

		AktivEtterPassiv = 1	
Data frame	Total no. of obs.	Frequency	Percentage
One month	27485	11232	40.9%
Three months	20527	7295	35.5%
Six Months	15553	4714	30.3%
Twelve months	9395	2624	27.9%

the rest of the numeric variables, are set to be the median of the relevant column. Finally, missing values in categorical variables are categorized as ”Unknown”. This handling of missing values is applied to all four data frames, one_month_ahead, three_months_ahead, six_months_ahead and twelve_months_ahead.

The next step was to dummy encode categorical variables. A new column was added for each category, equal to 1 if that category was present and 0 else. The original variable was removed together with the majority category of each variable, which acts as the reference for the remaining categories. The data frames were then split into training and test sets. The split was done randomly, with 75% of observations in each of the four training sets, one for each data frame, and the remaining 25% in the four test sets.

Logistic regression requires the data to be standardized. This is important to avoid vastly different magnitudes in the explanatory variables, which could lead to some variables wrongly dominating the model. Standardization was done by finding the mean and standard deviation of each column in the training set, then each column in the training set was subtracted by its mean and divided by its standard deviation. As the test set should be new and unknown data, it was standardized using the same method, with the mean and standard deviation of the training set. All four data frames were standardized.

3.3 Visualization

Visualizations of the variables can be a helpful tool to get an improved understanding of the relations between the response and explanatory variables. These relations are assumed to be consistent in all four data tables. All visualizations in this section are based on the one_month_ahead data frame, before the handling of missing values, as it is the data frame with the most observations of the four.

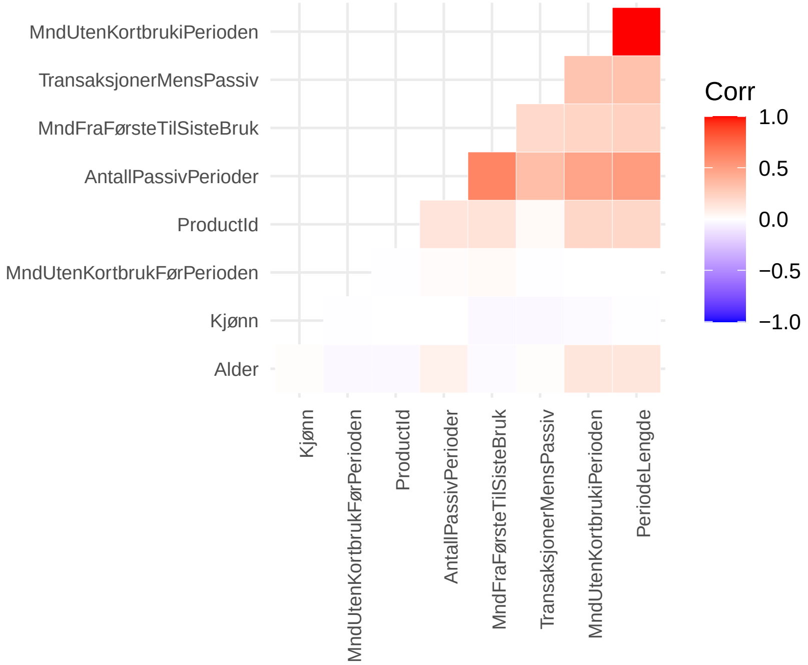


Figure 3.1: Correlation matrix of the one_month_ahead data frame, with explanatory variables from the passive_table.

Figure 3.1 shows the correlation between explanatory variables from, or made of variables in, the passive_table. The variable PeriodeLengde reflects the number of months in which information of the customers' credit card usage is used to predict their future activity. MndUtenKortbrukiPerioden is the number of months in which the card has not been used in that same period. The correlation between these two variables is close to 1. Since the data set consists of passive customers, it is reasonable that a substantial portion of the customers have not used their card in the relevant period. There is also some correlation between AntallPassivPerioder, which reports the number of distinct times the customer has been passive in the period, and MndFraFørsteTilSisteBruk, which is the number of months between the customer's first and last credit card use.

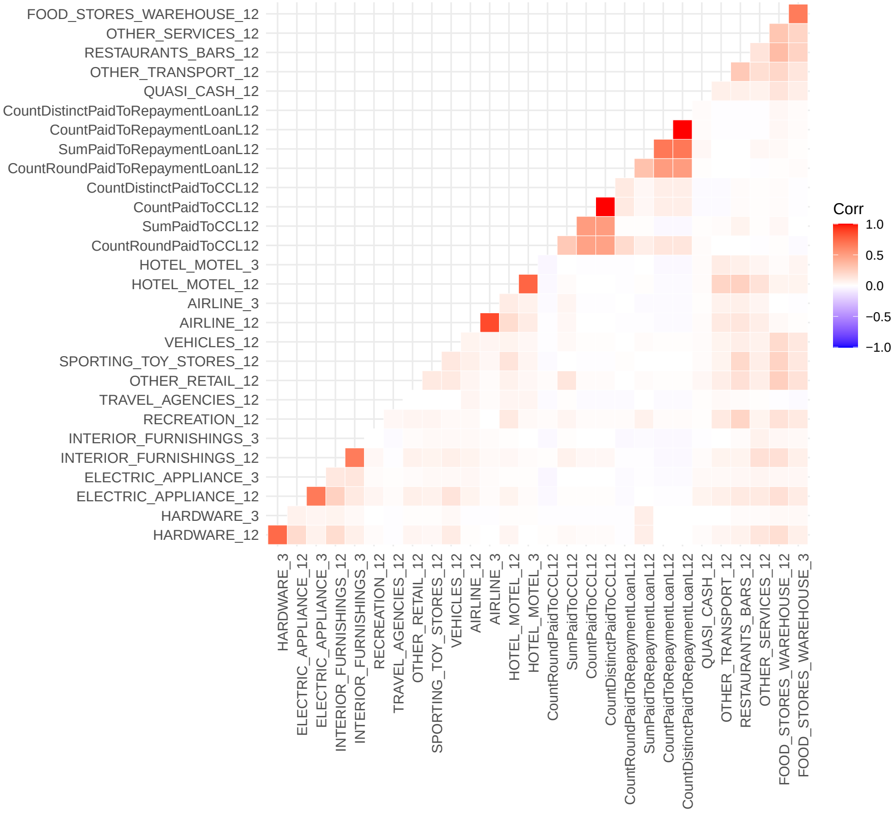


Figure 3.2: Correlation matrix of the one_month_ahead data frame, with explanatory variables from the purchase_history_table and purchase_history_segment_table.

Figure 3.2 shows the correlation between explanatory variables from both the purchase_history_table and purchase_history_segment_table. Some of the segments contained in the latter table have two features, one which sums the last twelve months and one summing the last three months. Among the segments containing both are hardware, hotel/motel, and airline. The twelve- and three-month summary of these segments are correlated as seen from the figure. Apart from these, the most correlated variables are those related to payments to the same type of external accounts, such as the variables describing payments to repayment loan accounts; CountDistinctPaidToRepaymentLoanL12, CountPaidToRepaymentLoanL12 and SumPaidToRepaymentLoanL12.

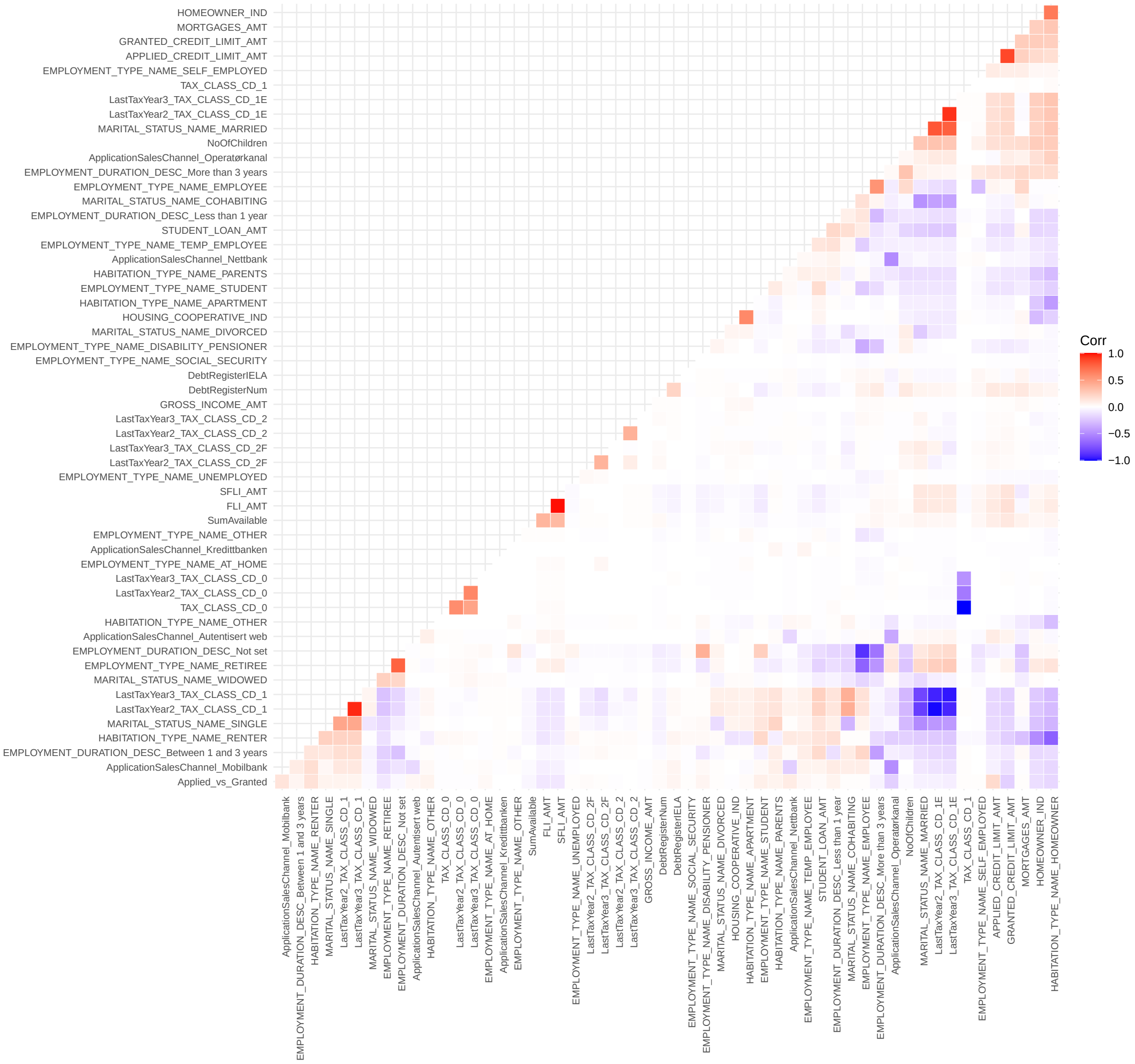


Figure 3.3: Correlation matrix of the one_month_ahead data frame, with explanatory variables from the application_table after dummy encoding.

The last correlation matrix is of the variables in the application_table where categorical variables are dummy encoded, shown in Figure 3.3. The variables FLI_AMT and SFLI_AMT are strongly correlated. This is to be expected as they both describe a customer's simplified liquidity indicator, where SFLI_AMT is the simplified liquidity indicator based on a 5% increase in interest. The simplified liquidity indicator is a measure of a customer's ability to pay. APPLIED_CREDIT_LIMIT_AMT and GRANTED_CREDIT_LIMIT_AMT are also correlated. The two variables reflect the amount of credit limit the customer applied for, and the amount they were granted. From Figure 3.3 it can also be seen that applications where the employment duration is not set are positively correlated with retirees, while it is negatively correlated with those who are employees. The rest of the most correlated variables are related to tax class. It is reasonable that the tax class generally is consistent from one year to the next, i.e., positive correlation between LastYear2_TAX_CLASS_CD and LastYear3_TAX_CLASS_CD of same category, and negative correlation of dissimilar categories.

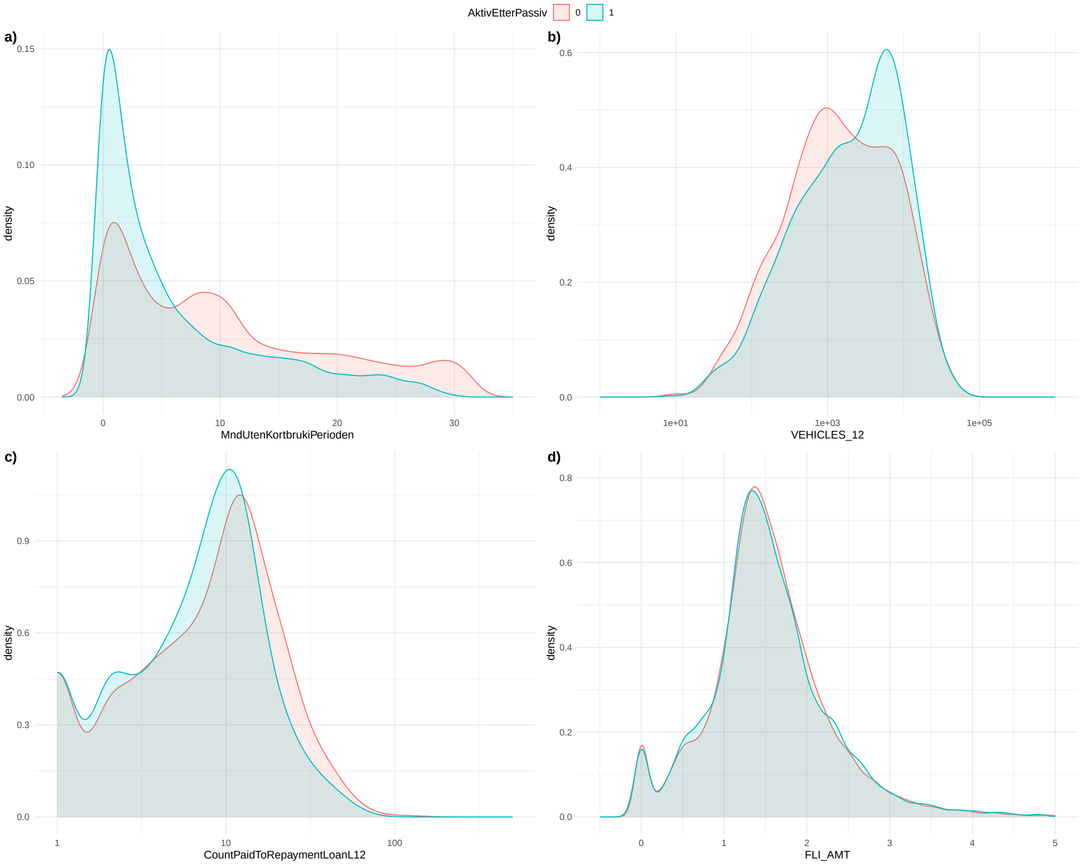


Figure 3.4: Density plots of four selected explanatory variables. **a)** Number of months without use of credit card in the period. **b)** Sum of purchases the last 12 months in the vehicles segment. **c)** Sum of amount paid to external repayment loan accounts the last 12 months. **d)** Customer's simplified liquidity indicator.

Inspection of density plots can improve the understanding of how the variables are distributed. Figure 3.4 shows density plots of four selected variables, divided by the response, AktivEtterPassiv. The curve of those who become active is displayed in light blue, while those who stay passive are displayed by a light red colored curve. Plot **a)** in Figure 3.4 shows that the portion of customers that become active is greater the fewer months where the credit card has not been used in the period. Plot **b)** indicates that customers who have bought a greater amount in the vehicles segment tend to become active more often than customers who bought less in the same category. The curves in plot **c)** show a similar distribution of the active and passive customers. The curve of customers who become active is slightly left-skewed compared to the one of passive customers, which suggests customers with a lower count of payments made to external repayments loan accounts are somewhat more likely to become active. In plot **d)** of FLI_AMT, there is no noteworthy distinction between active and passive customers. The two curves follow each other very closely, which indicates that separating customers who stay passive from those who become active based on their value of FLI_AMT is difficult.

Chapter 4

Analysis and Results

This chapter will briefly describe how the models were implemented before presenting and analyzing the obtained results. Model outputs and results from `R` for logistic regression are included in Appendix B, and for AdaBoost in Appendix C.

4.1 Logistic Regression

The logistic regression models were fitted using the `glm()` function in `R`. First, the data was pre-processed as described in Chapter 3.2, which resulted in four data frames; one_month_ahead, three_months_ahead, six_months_ahead, and twelve_months_ahead. Collinearity between variables leads to the design matrix $\mathbf{X}$ not having full rank which further results in $\mathbf{X}^T\mathbf{X}$ being singular. This is a problem in estimation of the coefficients, thus, the `alias()` function in `R` was used to find linearly dependent variables which needed to be removed. This resulted in discarding PeriodeLengde from all four data frames, and ‘EMPLOYMENT_DURATION_DESC_Not set‘ from six_months_ahead and twelve_months_ahead, as well as CountDistinctPaidToRepaymentLoanL12 and CountDistinct-PaidToCCL12 from twelve_months_ahead.

The first model was fitted by applying logistic regression to the one_month_ahead data frame, this will be referred to as the oneMo model. Summary of the full model output is found in Appendix B.1. Table 4.1 shows estimate of the intercept together with the variables whose p-value is less than 0.001. The p-value is a measure of a variable's statistical significance, and various levels of significance is represented by; (***) for 0.001, (**) for 0.01, (*) for 0.05, and (.) for 0.1.

Remember, Equation 2.2 showed that a one-unit increase in x_{ij} leads to an increase in the odds that $y_i = 1$ of $\exp(\beta_j)$. Thus, the estimates of the coefficients represent the effect each variable has on the log-odds ratio that the response belongs to class 1. Consider the variable MndUten-

Table 4.1: Partial model output of oneMo with estimated coefficients and p-value for the variables with p-value less than 0.001, and the intercept.

Coefficients	Estimate	$\Pr(>\|z\|)$
(Intercept)	-6.651e-01	0.004153
MndUtenKortbrukiPerioden	-9.872e-02	< 2e-16
MndFraFørsteTilSisteBruk	-1.849e-02	1.07e-11
RESTAURANTS_BARS_12	-1.546e-01	4.14e-12
TRAVEL_AGENCIES_12	-6.494e-02	9.46e-05
HOTEL_MOTEL_3	-8.952e-02	0.000854
AktiviPerioden	6.838e-01	6.11e-07
Missing_purchaseSeg	-5.381e-01	6.18e-11
Missing_application	5.609e-01	5.99e-06
Missing_sumAvail	9.225e-01	< 2e-16
ApplicationSalesChannel_Kredittbanken	-2.897e+00	9.33e-05
ApplicationSalesChannel_Mobilbank	1.756e-01	0.000570
ApplicationSalesChannel_Nettbank	2.846e-01	1.07e-08
HABITATION_TYPE_NAME_RENTER	-2.837e-01	7.79e-05

KortbrukiPerioden, which is the number of months where the credit card was not used in the considered period. The estimated coefficient of this variable is equal to -0.0987, as seen in the partial model output in Table 4.1. Thus, a one-unit increase in MndUtenKortbrukiPerioden while keeping all other variables constant, decreases the log-odds of the response belonging to class 1 by 0.0987. Or a one-unit increase in the same variable results in the odds decreasing by a factor of $\exp(-0.0987) = 0.9060$. This means that a customer who has more inactive months is less likely to become active.

Logistic regression was then fitted to the three_months_ahead, six_months_ahead and twelve_months_ahead data frames, which resulted in three models that will be referred to as the threeMos, sixMos and twelveMos models, respectively. Recall, as the models are supposed to predict different numbers of months ahead, they have distinct training and test sets. Thus, the number of observations used to train and test the models differs. In addition, the number of explanatory variables differs slightly; in the one_month_ahead and three_months_ahead data frames there are 94, while the six_months_ahead has 92, and the twelve_months_ahead has 90. Full model outputs are found in Appendix B.1, which show that many of the same variables are regarded as statistically significant in the four models. If a significance level of 0.05 is considered then the oneMo model has 27 significant explainable variables, the threeMos model has 22, the sixMos model has 25, and the twelveMos model has 23. With this level of significance, 29% of the explainable variables in the oneMo model is regarded as significant, 22% in threeMos, 27% in sixMos, and 26% in twelveMos. For instance, ProductId, which is a numeric variable corresponding to which type of credit card the customer has, is one of the variables with lowest p-value in all four models, together with MndUtenKortbrukiPerioden, which is the number of months in the considered period where the credit card was not used. However, there are also some differences, e.g., MndUtenKortbrukFørPerioden, which is the number of months between the passive period and the last time the credit card was used before that period, is statistically significant at a level 0.001 in twelveMos, and 0.05 in threeMos, while in oneMo and sixMos that variable is not regarded as significant.

In Table 4.2, confusion matrices are shown of all four models applied to their respective test sets, with cutoff at 0.5. In each table, the top left is the number of observations correctly classified as passive, and the bottom right is the number of observations correctly classified as active, while the top right and bottom left are the number of observations wrongly classified as passive and active, respectively. All four models have trouble classifying the observations that become active, which is the minority class as there are more observations where the customer stays passive. I.e., the minority class is active and the majority class is passive. The oneMo model is able to classify the largest part of the minority class in the test set, while the sixMos and twelveMos models perform approximately equally poorly. This is expected as the one_month_ahead data frame has the highest percentage of the minority class, and the further ahead the model should predict, the more imbalanced the data set is, as seen in Table 3.2. It is advantageous to identify which customers are more likely to become active, such that the account of a customer who would become active is not closed. The type II error is high when a lot of positive observations are falsely classified as negative. Thus, reducing the type II error is desired, and a method to achieve this will be applied and discussed later.

Table 4.2: Confusion matrices after using the different models to classify the corresponding test set, with cutoff = 0.5.

		Actual	
		0	1
Predicted	0	3145	1560
	1	879	1300

(a) OneMo

		Actual	
		0	1
Predicted	0	2775	1207
	1	466	526

(b) ThreeMos

		Actual	
		0	1
Predicted	0	7579	2739
	1	567	747

(c) SixMos

		Actual	
		0	1
Predicted	0	1558	514
	1	114	138

(d) TwelveMos

Based on the confusion matrices, the metrics sensitivity, specificity, and balanced accuracy (BACC) are computed. Table 4.3 presents these metrics together with the area under the ROC curve (AUC) for each model. All four models have an AUC value between 0.69 and 0.70, and their ROC curves, shown in Figure 4.1, are very similar. The twelveMos model has the highest value in specificity, but in return, the lowest value in sensitivity, which means that the model performs poorly at classifying the positive class, as seen from the confusion matrix in Table 4.2 (d). The small value of sensitivity results in the twelveMos model having the lowest BACC value among the four models. As the months to predict ahead decrease, the models' sensitivity increases and thus also the BACC value, despite the specificity decreasing. Hence, the oneMo model has the smallest value of specificity, but the largest of sensitivity and BACC. Based on the observations from the confusion matrices and the percentage of imbalance in the different data frames, this was expected.

Table 4.3: Reported metrics of the four models applied to their respective test set, cutoff = 0.50.

Model	Sensitivity	Specificity	AUC	BACC
oneMo	0.4545	0.7816	0.6917	0.6181
threeMos	0.3035	0.8562	0.6897	0.5799
sixMos	0.2143	0.9304	0.6992	0.5723
twelveMos	0.2117	0.9318	0.6931	0.5717

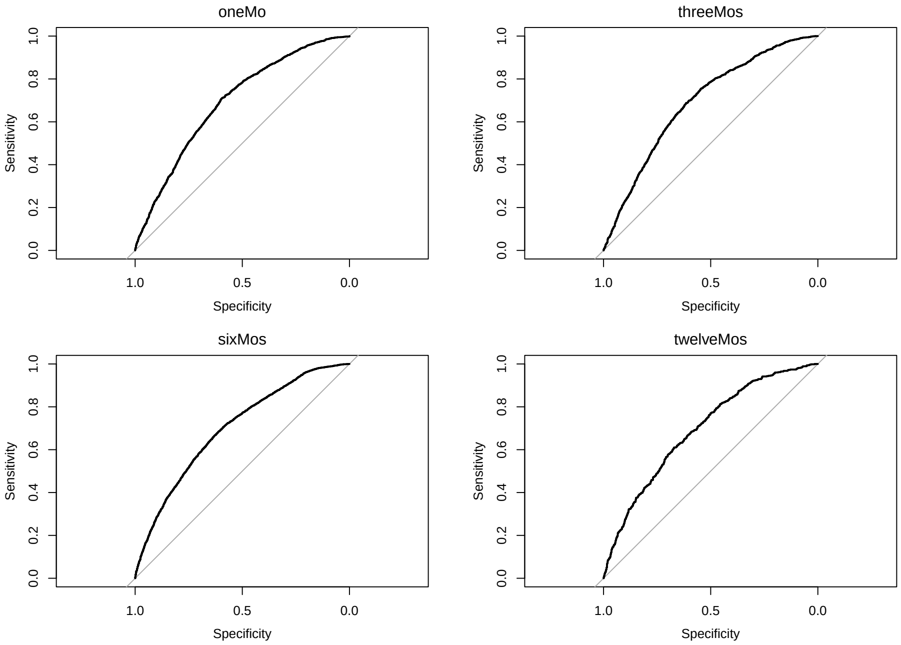


Figure 4.1: ROC curve for the oneMo, threeMos, sixMos and twelveMos models.

4.1.1 Feature Selection

Among the four models, the percentage of explanatory variables regarded as significant was between 22% and 29%. Thus, several of the variables contribute with little to no effect on the response. This suggests that reducing the variable dimension is reasonable. Backward elimination (BE) was therefore performed on the four models with both BIC and AIC, using the function `step()` with $k = ln(m)$ to define the penalty term for BIC and $k = 2$ for AIC, where m is the number of observations in the training set.

BE with BIC was applied to the four models. Applied to the oneMo model resulted in a new model with 18 explanatory variables, to the threeMos model there were 14 variables remaining, 16 variables for the sixMos model, and 15 for the twelveMos model. Model outputs are found in Appendix B.2. Now, all variables in these models are regarded as statistically significant at level 0.05. Compared to the original models which all had at least 90 explanatory variables, these are considerable reductions. Reported metrics for the four BIC-reduced models are shown in Table 4.4. Compared to the reported metrics of the original models in Table 4.3, the specificity values have increased slightly, while all other values have reduced a little.

Table 4.4: Reported metrics of the four BE models with BIC when applied to their respective test set, with cutoff = 0.50.

Model	Sensitivity	Specificity	AUC	BACC
oneMo	0.4381	0.7920	0.6891	0.6151
threeMos	0.2874	0.8713	0.6839	0.5793
sixMos	0.1951	0.9358	0.6878	0.5654
twelveMos	0.1887	0.9474	0.6919	0.5680

As the BIC-reduced models are nested in their corresponding original models, ANOVA analysis is performed. The ANOVA analysis tests the null hypothesis that the coefficient of variables found in the full model, but not in the reduced model, are equal to zero, against the alternative hypothesis that at least one of those coefficients are different from zero. That way, the test evaluates if the reduced or full model is preferred. The ANOVA analysis of oneMo, threeMos, sixMos and twelveMos between their full and BIC-reduced models resulted in p-values of $1.457e{-}07$, $9.722e{-}09$, $1.313e{-}05$ and 0.0001217, respectively, and can be found in Appendix B.4. These are all smaller than 0.05, thus, the null hypotheses are rejected, and the full models are preferred.

BE applied to the oneMo model with AIC resulted in a model of 46 explanatory variables, which is more than twice the number of variables in the BIC-reduced oneMo model. Similarly, BE with AIC applied to the threeMos, sixMos and twelveMos models resulted in reduced models of 49, 43 and 41 explanatory variables, respectively. The AIC-reduced models have remarkably more variables than the corresponding BIC-reduced models. However, the number of variables is still greatly reduced from the full original models. Outputs of the AIC-reduced models are found in Appendix B.3.

These show that most of the variables included are regarded as statistically significant, but not all as for the BIC-reduced models. Table 4.5 reports the metrics of the four AIC-reduced models after applied on the test sets. Compared to metrics of the BIC-reduced models, the specificity values are somewhat smaller for the AIC-reduced models, on the contrary, the sensitivity and BACC values are slightly improved.

Table 4.5: Reported metrics of the four BE models with AIC when applied to their respective test set, with cutoff = 0.50.

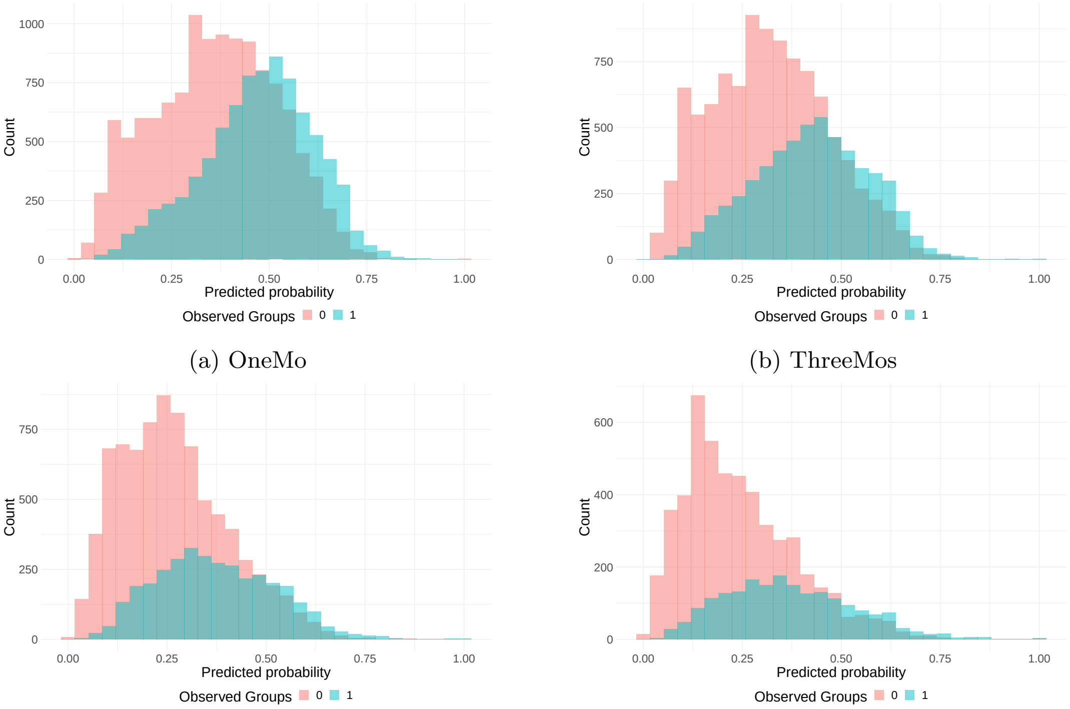
Model	Sensitivity	Specificity	AUC	BACC
oneMo	0.4500	0.7823	0.6907	0.6162
threeMos	0.3001	0.8587	0.6890	0.5794
sixMos	0.2117	0.9288	0.6973	0.5703
twelveMos	0.2086	0.9324	0.6930	0.5705

ANOVA analysis is also performed between the four original models and their corresponding AIC-reduced models, shown in Appendix B.5. The four ANOVA analyses all resulted in p-values close to 1; 0.9989, 0.9961, 0.9904 and 0.9983 for oneMo, threeMos, sixMos and twelveMos, respectively. Since the p-values are large there is not sufficient evidence to reject the null hypotheses. Considering the four AIC-reduced models have fewer variables than their respective full models, they could be regarded as the preferred models. Though, as the full models had higher values in sensitivity and BACC, and thus performed better at predicting the minority class, these will be considered further.

4.1.2 Optimize the Cutoff Value

In cases of imbalanced data set, logistic regression with default cutoff at 0.5 may produce quite poor results. Finding the optimal cutoff value can therefore improve the predictive performance of the model fairly much. Before optimizing the cutoff, consider the classification plots in Figure 4.2. Here, the predicted probability of all observations made by the model is represented by the x-axis, and the y-axis shows the count. The pink histograms visualize the distribution of the true negative class's predicted probability, while the blue histograms visualize the distribution of the true positive class's predicted probability according to the relevant model. There is substantial amount of overlap in all four plots. Thus, the models will never completely separate the two classes, but it is obvious that a cutoff value of 0.5 is not optimal for any of the models, and optimizing the cutoff is expected to improve the predictive performance.

The function `performance()` with `"sens"` and `"spec"` was used to calculate the sensitivity and specificity values of 6999 different cutoff points which further was used to find the cutoff that maximized the average of those, i.e., the balanced accuracy. The confusion matrices are presented in Table 4.6, with the estimated optimal cutoff value for each model.



(a) OneMo

(b) ThreeMos

(c) SixMos

(d) TwelveMos

Figure 4.2: Classification plot of the four full logistic regression models, with observations belonging to the negative class shown in pink, and positive class as blue. The x-axis represent the predicted probability of belonging to the positive class according to the model.

Table 4.7 reports the optimal cutoff value for each of the models, together with classification metrics. The found optimal cutoff for all four models is close to the portion of minority observations in the respective training sets, reported in Table 3.2. The oneMo model performs best at classifying the positive class with a sensitivity value of 0.6906, and the highest score of BACC. The threeMos model has the largest value of specificity, but the smallest of sensitivity, while the sixMos has the second largest sensitivity value, which results in the second highest BACC score. However, the difference in BACC between oneMo, threeMos and sixMos are only at the level of thousandths. The twelveMos model has the second smallest value of sensitivity and smallest specificity value of all the models, resulting in the lowest BACC score.

Thus, the oneMo model performs best and the twelveMos model worst among the four as anticipated. However, a bigger difference in BACC was expected as predicting one month ahead was envisioned easier than twelve. Therefore, only the one_month_ahead and twelve_months_ahead data frames will be used in the further modeling and analyses of adaptive boosting.

Table 4.6: Confusion matrices after using the different models to classify their corresponding test set, with optimized cutoff values.

		Actual	
		0	1
Pred.	0	2441	885
	1	1583	1975

(a) OneMo, cutoff = 0.41

		Actual	
		0	1
Pred.	0	2071	599
	1	1170	1134

(b) ThreeMos, cutoff = 0.37

		Actual	
		0	1
Pred.	0	4967	1097
	1	3179	2389

(c) SixMos, cutoff = 0.29

		Actual	
		0	1
Pred.	0	1012	217
	1	660	435

(d) TwelveMos, cutoff = 0.26

Table 4.7: Reported metrics of the four models when applied to their respective test set, with optimized cutoff values.

Model	Cutoff value	Sensitivity	Specificity	BACC
oneMo	0.41	0.6906	0.6066	0.6486
threeMos	0.37	0.6544	0.6390	0.6467
sixMos	0.29	0.6853	0.6097	0.6475
twelveMos	0.26	0.6672	0.6053	0.6362

4.2 Adaptive Boosting

In this section, oneMo and twelveMos will refer to AdaBoost models fitted to the one_month_ahead and twelve_months_ahead data frames, respectively. If referring to the logistic regression models, this will be specified.

AdaBoost was fitted using the function `gbm()` in `R` with `distribution = "adaboost"`, which is an extended implementation to the AdaBoost algorithm introduced by Freund and Schapire in 1995, (Greenwell et al. 2022). Before optimizing the hyperparameters, AdaBoost was fitted with default values for all hyperparameters to have reference points for comparison of the results. Two default models were fitted, one with AdaBoost applied to the one_month_ahead data frame, and the other applied to the twelve_months_ahead data frame. For logistic regression, the data frames needed to be normalized, this is not required for AdaBoost. Therefore, the data frames used to fit the

AdaBoost models are not normalized. After training, the two default models were evaluated on their respective test set. Table 4.8 shows the confusion matrix of both the oneMo and twelveMos models with default hyperparameter values.

Table 4.8: Confusion matrices after using the two default AdaBoost models to classify their corresponding test set, with cutoff = 0.5.

		Actual	
		0	1
Predicted	0	3365	1750
	1	761	1110

(a) OneMo

		Actual	
		0	1
Predicted	0	1696	659
	1	8	17

(b) TwelveMos

Reported classification metrics are shown in Table 4.9. The different amount of imbalance is even more clear than for the logistic models, and the sensitivity values show that both default AdaBoost models have great difficulties at predicting the positive class. Both AUC scores are higher for the default AdaBoost models compared to their corresponding logistic models, while the sensitivity and BACC values are reduced. Specifically, the default twelveMos model has an extremely low value of sensitivity while the specificity is close to 1. This indicates that most observations are classified as the negative class, this can also be seen in the confusion matrix in Table 4.8b. The BACC value of 0.5102 shows that the model performs almost equivalent to random guessing. Even though the default oneMo model is able to classify more of the minority class, many of them are still misclassified as seen in the confusion matrix in Table 4.8a.

Table 4.9: Reported metrics of AdaBoost on the one_month_ahead and twelve_months_ahead data frames, with default values on all hyperparameters and cutoff = 0.5.

Model	Sensitivity	Specificity	AUC	BACC
Default oneMo	0.3881	0.8156	0.6947	0.6018
Default twelveMos	0.0251	0.9953	0.7135	0.5102

Four hyperparameters from the `gbm()` function in addition to the cutoff are chosen for optimization to improve the results. The parameters and their corresponding default value are presented in Table 4.10.

Table 4.10: Chosen hyperparameters with default values.

Hyperparameter	n.trees	interaction.depth	shrinkage	bag.fraction	cutoff
Default value	100	1	0.1	0.5	0.5

The `gbm()` function returns a value between 0 and 1 for each observation, a predicted probability to belong to the positive class, similar to logistic regression. The cutoff is not a built-in parameter in the `gbm()` function, however, as the response is binary it is needed to define which observations to classify as positive and negative. Optimizing the cutoff showed quite impressive improvements in the logistic models and has in general a profound effect on imbalanced classification problems.

The value of n.trees specifies the number of trees to fit. The larger number of trees to grow, the longer the model needs to run. The interaction.depth parameter is the maximum depth of the trees, which corresponds to the upper bound of level of variable interaction. Thus, a larger value of interaction.depth leads to a more complex model which can be more prone to overfitting. The shrinkage parameter, also called learning rate, specifies how fast the model should learn. In general, a value between 0.001 and 0.1 is advised for this hyperparameter, and a smaller value will make the learning process slower and take more time to run. In addition, a small shrinkage value often requires more trees, increasing the run time even more. Last, bag.fraction determines the fraction of randomly chosen observations from the training set that should be used to build the next tree. If the value of bag.fraction is set equal to 1, then all observations will be used.

4.3 Tuning Hyperparameters in Adaptive Boosting

Tuning of the hyperparameters is done through Response Surface Methodology. This method consists of several steps, where the first is a screening experiment performed with Design of Experiments. Then, steepest ascent is used to explore a different experimental region, where a second-order response surface model is fitted. First, the optimization of hyperparameters in the oneMo model will be considered, then the twelveMos model.

4.3.1 Optimizing the One-Month-Ahead Model through Response Surface Methodology

Since there are five hyperparameters to tune, a 2^{5-1} fractional factorial design is chosen for the screening experiment to reduce the number of runs needed. First, low and high levels of each parameter must be set. The default value of n.trees is 100, which in general is considered a quite small amount. Thus, the low level of n.trees is set equal to 100, and the high level to 600. The interaction.depth's default value is 1, which is the smallest possible. Therefore, 1 is set to be the low level and 5 to be the high level of interaction.depth. The shrinkage parameter is advised to be between 0.001 and 0.1, the low and high levels are therefore decided to be 0.03 and 0.07. The low and high levels of bag.fraction are set around its default value, 0.5 ± 0.1. As the optimal cutoff point often corresponds with the fraction of minority observations, the low level of cutoff is set equal 0.4 and high level equal 0.45. The low and high levels of each hyperparameter are shown in Table 4.11, together with the factor names.

Table 4.11: Selected low and high levels of each hyperparameter.

Factor	Hyperparameter	Low Level	High Level
A	n.trees	100	600
B	interaction.depth	1	5
C	shrinkage	0.03	0.07
D	bag.fraction	0.4	0.6
E	cutoff	0.4	0.45

Table 2.2 shows the setup for the 2^{5-1} fractional factorial experiment which was conducted and replicated one time. The results from the full experiment are found in Table C.1 in the appendix. 5-fold cross-validation was performed at each experiment run and the reported response is the mean BACC. Table 4.12 shows the mean BACC of the identical runs. The highest BACC value is 0.6752 and was obtained at level code *abd*.

Table 4.12: Results of the 2^{5-1} fractional factorial design of AdaBoost on the one_month_ahead data frame, with mean BACC from the duplicate runs.

Experiment No.	A	B	C	D	E	Level code	Mean BACC
1 & 17	-1	-1	-1	-1	1	*e*	0.6213
2 & 18	1	-1	-1	-1	-1	*a*	0.6532
3 & 19	-1	1	-1	-1	-1	*b*	0.6586
4 & 20	1	1	-1	-1	1	*abe*	0.6722
5 & 21	-1	-1	1	-1	-1	*c*	0.6365
6 & 22	1	-1	1	-1	1	*ace*	0.6536
7 & 23	-1	1	1	-1	1	*bce*	0.6657
8 & 24	1	1	1	-1	-1	*abc*	0.6743
9 & 25	-1	-1	-1	1	-1	*d*	0.6225
10 & 26	1	-1	-1	1	1	*ade*	0.6483
11 & 27	-1	1	-1	1	1	*bde*	0.6542
12 & 28	1	1	-1	1	-1	*abd*	0.6752
13 & 29	-1	-1	1	1	1	*cde*	0.6275
14 & 30	1	-1	1	1	-1	*acd*	0.6574
15 & 31	-1	1	1	1	-1	*bcd*	0.6672
16 & 32	1	1	1	1	1	*abcde*	0.6745

A model with main effects and interactions was fitted to the 2^{5-1} fractional factorial design with BACC as response. Table 4.13 shows the coefficient estimates, while the full model summary is displayed in Appendix C.1. Excluding the intercept, 12 of the coefficients are regarded significant at a level of 0.01. Considering the p-values, factors A and B have the smallest values and are therefore those of most statistical significance. In addition, the p-value of factor C and interactions AB and AC are also quite small. The residuals of the model with main effects and interactions are assumed to be independent and approximately normally distributed. To see if this assumption holds, the normal Q-Q plot of the residuals in Figure C.1 is considered. The residuals lie quite nicely on the line around 0, while around ±1 they deviate a bit but still follow the line closely. The endpoints deviate most from the line. In Figure C.2, the residuals are plotted and there does not seem to be any pattern. This suggests that the model assumption of independent and normally distributed residuals holds.

Table 4.13: Coefficient estimates of the model with main effects and interactions fitted to the 2^{5-1} fractional factorial design for the one_month_ahead data frame with AdaBoost.

	Estimate	Std. Error	t value	Pr(> |t|)	
(Intercept)	6.539e-01	1.646e-04	3971.860	< 2e-16	***
A	9.684e-03	1.646e-04	58.826	< 2e-16	***
B	1.384e-02	1.646e-04	84.096	< 2e-16	***
C	3.183e-03	1.646e-04	19.336	1.61e-12	***
D	-5.498e-04	1.646e-04	-3.340	0.004156	**
E	-1.734e-03	1.646e-04	-10.531	1.33e-08	***
A:B	-3.402e-03	1.646e-04	-20.666	5.77e-13	***
A:C	-1.829e-03	1.646e-04	-11.112	6.22e-09	***
A:D	8.131e-04	1.646e-04	4.939	0.000148	***
A:E	3.030e-04	1.646e-04	1.840	0.084318	.
B:C	-5.135e-04	1.646e-04	-3.119	0.006605	**
B:D	5.602e-04	1.646e-04	3.403	0.003640	**
B:E	6.365e-04	1.646e-04	3.867	0.001367	**
C:D	9.699e-05	1.646e-04	0.589	0.563979	
C:E	-1.644e-05	1.646e-04	-0.100	0.921695	
D:E	-4.940e-04	1.646e-04	-3.001	0.008468	**

Figure 4.3 shows the main effects of the five factors A, B, C, D and E. The mean response for the low and high levels of each factor are displayed as small squares with a line connecting them. The slope of the line represents the effect of moving the factor from low to high level has on the response. Thus, a steeper slope means that the factor has a greater effect on the response. If the

line connecting the mean response for the low and high level is approximately horizontal, then the factor has no to little effect on the response. From Figure 4.3, factors A and B both have steep and positive slopes and affect the response the most. C also has a positive slope, while E has a negative. From Table 4.13, factor B has the largest estimated coefficient, which agrees with the observation that B has the steepest slope in Figure 4.3. The magnitude of D's slope and estimated coefficient are both relatively small, thus, D has little effect on the response.

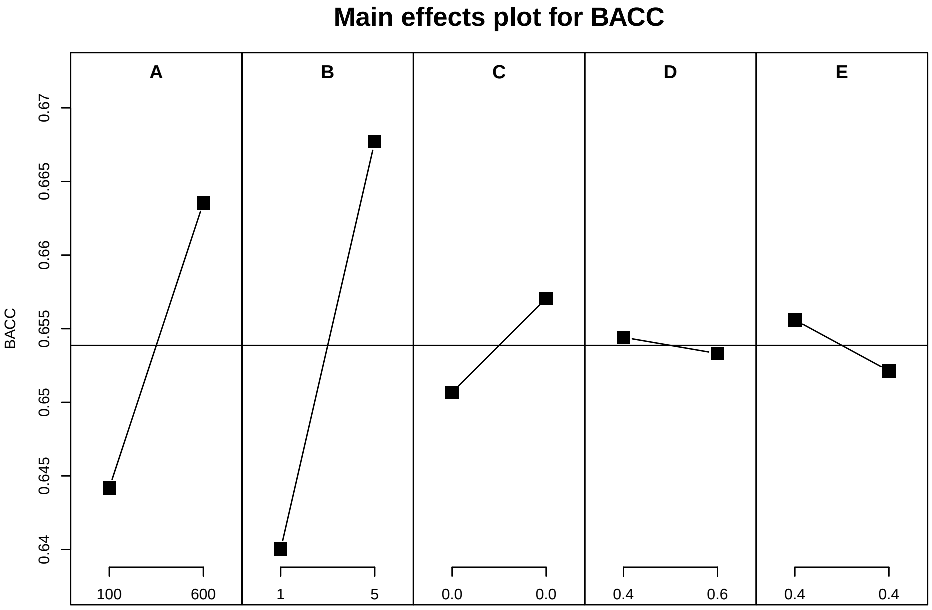


Figure 4.3: Main effects plot of factors A, B, C, D and E after performing the 2^{5-1} fractional factorial screening experiment for AdaBoost on the one_month_ahead data frame.

Figure 4.4 displays the interaction effects between the distinct factors. An interaction plot shows how the relationship between one factor and the response can depend on a second factor. The second factor's low and high levels are represented by a red dotted line and a black solid line, respectively. The mean response for the low and high levels of the first factor are displayed as red squares and black triangles for the low and high level of the second factor, respectively, with the lines connecting them. If the two resulting lines are parallel, then there is no interaction between the two factors. However, the more non-parallel the lines are, the greater is the interaction. In Figure 4.4, there are interactions between A and B, and A and C. This is confirmed by Table 4.13, where these interactions have among the largest estimated coefficients and smallest p-values. From the model summary, interactions between A and D, B and C, B and D, B and E, and D and E are significant at level 0.05. However, these interactions have smaller estimated coefficients. Thus, they have less impact on the response compared to other effects.

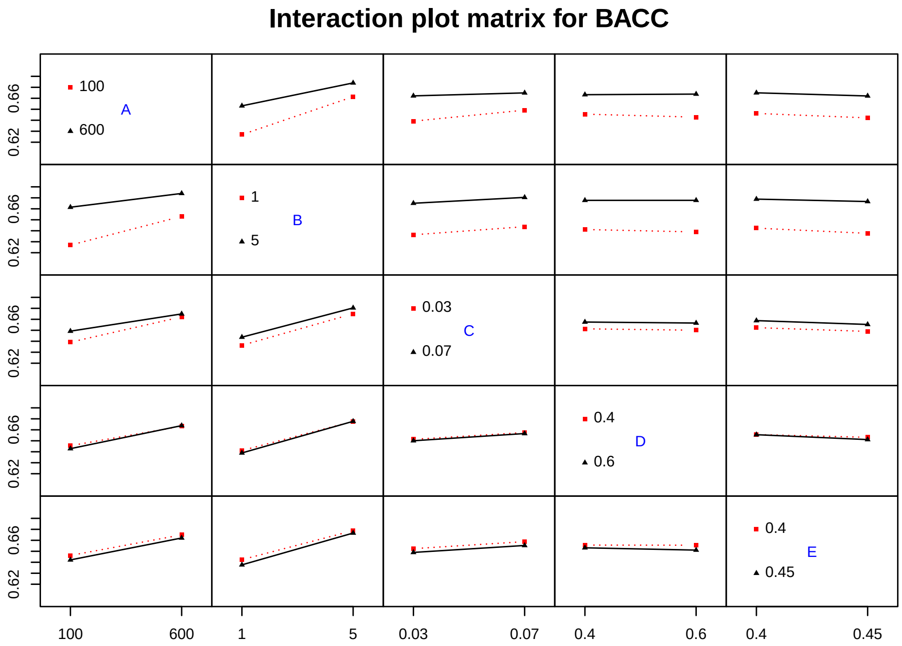


Figure 4.4: Interaction plots of factors A, B, C, D and E after performing the 2^{5-1} fractional factorial screening experiment for AdaBoost on the one_month_ahead data frame.

Figure 4.5 shows the normal plot of the estimated main and interaction effects. Under the assumption that the response is normally distributed, the effects are also normally distributed. The effects that form a straight line are assumed normally distributed with zero mean and can often be neglected, while those furthest away from the line are assumed to have nonzero mean and are regarded significant. From Figure 4.5, factors A, B and C, in addition to the interactions between A and B, and A and C are furthest from the line. Thus, these are the factors considered significant at a significance level of 0.05. This side with the findings from the main and interaction effects. Based on this analysis of the main and interaction effects, A, B and C are the factors that will be considered for further optimization.

Since the slope of D was quite gentle, and the interaction between the other factors were less noteworthy, D was set at its default value 0.5 in the continued optimization. The effect of moving factor E from high level to low level was substantial, while the interaction effects with the other factors were not. Thus, E was set at its low level, 0.4. To decide the levels of A, B, and C, interaction effects should be considered. First, from the main effects plot in Figure 4.3, the highest response is obtained when both A and B are at their high level, while the interaction effect between the two factors was negative. Looking at the interaction plot in Figure 4.4, even though the interaction effect is negative, the highest BACC value is obtained when both A and B are at their high levels. Considering A and C, these also have positive main effects while the interaction effect between them is negative. However, from the interaction plot, the highest value of BACC is also here obtained with both A and C at high level. Therefore, all three factors should be at their high level. Hence, from the screening experiment, the hyperparameter values are established as $A = 600$, $B = 5$, $C = 0.07$, $D = 0.4$, and $E = 0.5$.

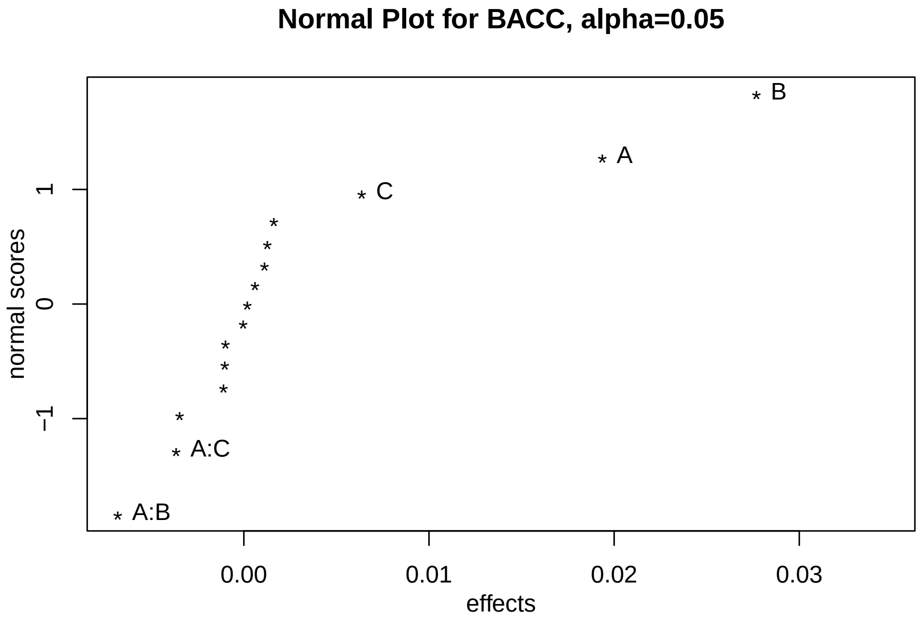


Figure 4.5: Normal plot of estimated main and interaction effects after performing the 2^{5-1} fractional factorial screening experiment for AdaBoost on the one_month_ahead data frame.

In the next part of the optimization process, the experimental region is moved through the method of steepest ascent where the goal is to approach the optimum. Based on the findings of the screening experiment, where A, B and C were the most influential factors, the movement will be along the vector where the slope of all three factors increases. From the model output in Table 4.13, the estimated slope of A is 0.009684, 0.01384 for B, and 0.003183 for C. This means that the magnitude of the slope of n.trees is 3 times larger than shrinkage, and the slope of interaction.depth has a magnitude which is more than 4 times larger than shrinkage. In the 2^{5-1} fractional factorial design, a one-unit increase in n.trees, interaction.depth and shrinkage was 250, 2 and 0.02, respectively. The shrinkage parameter was determined to start in 0.05 with a step size of 0.01. The step sizes of n.trees and interaction.depth were then determined based on shrinkage's step size value. Thus, starting at 3, interaction.depth was increased with 4 at each step, while n.trees started at 350 with a step size of 375. The cutoff was held constant at 0.4 and bag.fraction at the default value of 0.5. The path of steepest ascent with resulting response is shown in Table 4.14, and the response is the mean BACC value of 5-fold cross-validation. The highest value of BACC was 0.6785, which was obtained at the first step from the center point, with n.trees = 725, interaction.depth = 7 and shrinkage = 0.06. It is assumed that the improvement stopped early because of the negative interaction effects between A and B, and A and C.

Table 4.14: Path of steepest ascent for n.trees, interaction.depth and shrinkage in AdaBoost with cutoff = 0.4 and resulting BACC.

n.trees	interaction.depth	shrinkage	cutoff	BACC
350	3	0.05	0.4	0.6684
725	7	0.06	0.4	**0.6785**
1100	11	0.07	0.4	0.6769
1475	15	0.08	0.4	0.6771
1850	19	0.09	0.4	0.6662

In the new experimental region, a 2^3 factorial experiment was conducted with center points for n.trees, interaction.depth and shrinkage. Their center points were set equal to the configuration which obtained highest BACC score along the path of steepest ascent. The low and high levels of n.trees were chosen to be 725 ± 75. For interaction.depth the respective levels were set equal 7 ± 2, and for shrinkage 0.06 ± 0.02. The experiment was replicated one time, and center points were added to detect if a second-order model should be used. The full experiment is shown in rows 1 to 22 of Table C.2 in the appendix, where the reported BACC values are the mean after 5-fold cross-validation, and Table 4.15 shows the mean BACC of identical runs. A model with first order and two-way interactions was fitted to the 2^3 factorial design with center points, using the `rsm()` function in `R`, and BACC as response. The full model summary is shown in Listing 2 in Appendix C.1. The lack of fit test has 2 degrees of freedom since there are 9 distinct points, including a center point, in the design, and 7 terms are fitted to the data; three first order terms, three two-way interaction terms, and one intercept. The null hypothesis of the test is that a model with main effects and two-factor interactions is a good approximation to the data. Since the resulting p-value is small, equal to 0.0467, and significant at a level of 0.05, the null hypothesis is rejected, and the test suggests lack of fit.

Thus, a second-order model should be fitted to the data. Axial points at $\alpha \approx \pm\sqrt{3}$ are therefore added for each factor to construct a central composite design according to Table 2.3. The axial points of n.trees were 595 and 855, for interaction.depth they were 4 and 10, and for shrinkage they were 0.025 and 0.095. The resulting mean BACC after 5-fold cross-validation is shown in rows 23-28 in both Table 4.15 and C.2 in the appendix.

The full summary of the second-order response model is presented in Listing 3 in Appendix C.1. Now, the p-value of the lack of fit test is 0.7395, i.e., not significant at a level of 0.05, which suggests that the second-order model with quadratic terms is a more suitable fit to the true response

Table 4.15: Results of the central composite design with three factors for AdaBoost on the one_month_ahead data frame.

Experiment No.	A	B	C	BACC
1 & 12	-1	-1	-1	0.6763
2 & 13	1	-1	-1	0.6768
3 & 14	-1	1	-1	0.6791
4 & 15	1	1	-1	0.6797
5 & 16	-1	-1	1	0.6768
6 & 17	1	-1	1	0.6760
7 & 18	-1	1	1	0.6760
8 & 19	1	1	1	0.6755
9 & 20	0	0	0	0.6778
10 & 21	0	0	0	0.6797
11 & 22	0	0	0	0.6783
23	-1.73	0	0	0.6781
24	1.73	0	0	0.6779
25	0	-1.5	0	0.6752
26	0	1.5	0	0.6769
27	0	0	-1.75	0.6762
28	0	0	1.75	0.6752

surface. At a significance level of 0.05, the first and second order effects of interaction.depth and shrinkage are significant, in addition to the interaction effect between the two. The stationary point proposed by the model is at n.tree = 701, interaction.depth = 8 and shrinkage = 0.04448, with eigenvalues 0.00000000, -0.00042269 and -0.00126178. Since the first eigenvalue is zero and the other two are negative, the stationary point is essentially a line maximum. Figure 4.6 shows contour plots of the response surface, and perspective plots are shown in Figure C.3 in the appendix. The plot of n.trees and interaction.depth sliced at shrinkage = 0.06 visualize the maximum on a line within the design region. Such a response surface is called a stationary ridge system. The plot of n.trees and shrinkage taken at slice interaction.depth = 7 show a weakly rising ridge system. 5-fold cross-validation was performed at the suggested stationary point which obtained BACC values of 0.6902, 0.6883, 0.6684, 0.6822, and 0.6753. The mean of these five runs is 0.6809, which is a small improvement compared to earlier obtained results. Thereby, all values of n.trees, interaction.depth and shrinkage on the line passing through the stationary point with a direction given by the eigenvector $[0.9558, 0.1511, -0.2522]^T$ are potential candidates for an optimum.

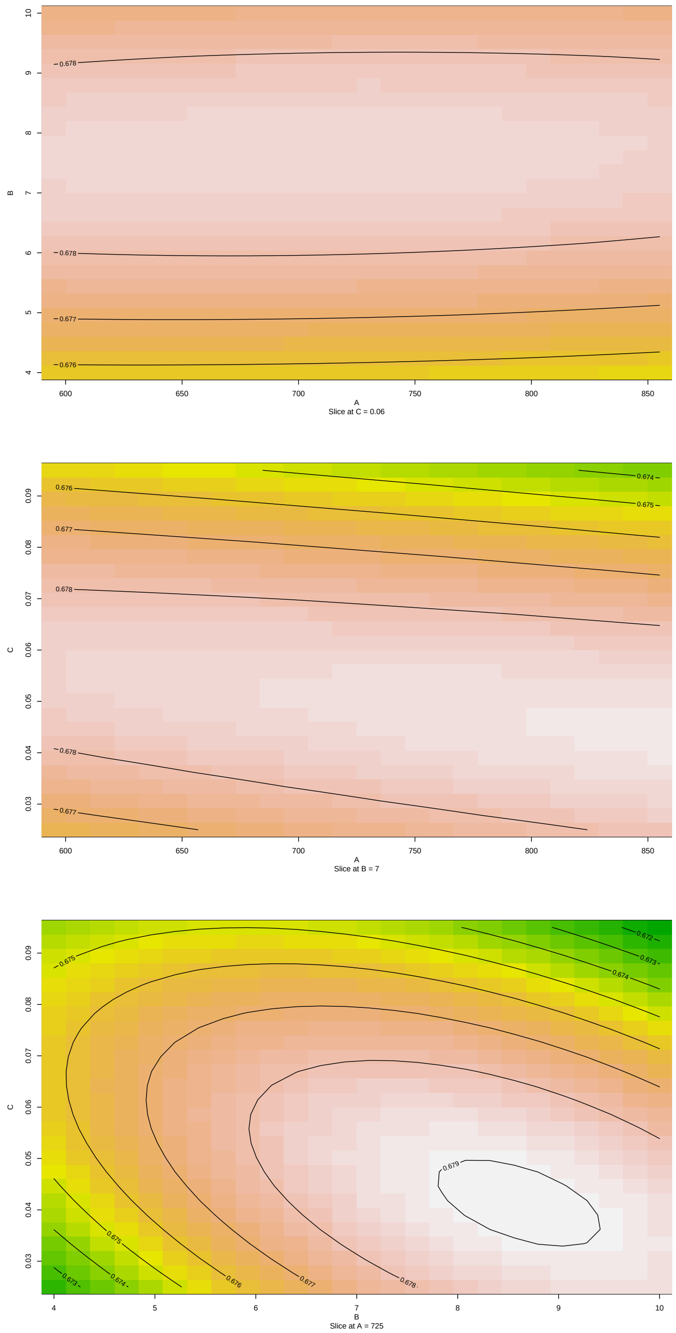


Figure 4.6: Contour plots of the first fitted second-order response surface model with AdaBoost.

A linear path in the direction of the eigenvector with origin at the proposed stationary point is found using the function `canonical.path()` in `R`. Table 4.16 shows the estimated canonical path for distances between -2 and 5 from the stationary point with mean BACC after 5-fold cross-validation.

Table 4.16: Results of AdaBoost performed along the canonical path starting from the stationary point proposed by the first second-order model.

Distance	n.trees	interaction.depth	shrinkage	BACC
-2	558	8	0.05458	0.6799
-1	630	8	0.04952	0.6813
0	701	8	0.04448	0.6808
1	773	9	0.03944	0.6794
2	845	9	0.03440	**0.6821**
3	916	9	0.02934	0.6810
4	988	10	0.02430	0.6798
5	1060	10	0.01926	0.6809

The configuration n.trees = 845, interaction.depth = 9 and shrinkage = 0.03440 obtained the highest BACC score of 0.6821, which is an improvement from earlier obtained results. A new second-order response surface was fitted with this new configuration as center point. Axial points with $\alpha \approx \sqrt{3}$ was used, and a total of six center runs. For n.trees, the low and high levels were 770 and 920, respectively, with axial points 715 and 975. The low and high levels of interaction.depth were 9 ± 1 with axial points 7 and 11. For shrinkage, the respective low and high levels were 0.02940 and 0.03940, with axial points 0.02574 and 0.04306. Table C.3 shows the mean BACC after 5-fold cross-validation, and the model summary is displayed in Listing 4, both in Appendix C.1. The lack of fit test has a p-value of 0.4886, indicating that the second-order model with quadratic terms still is a suitable approximation to the data. None of the estimated coefficients are statistically significant at level 0.05. The estimated stationary point is at n.trees = 851, interaction.depth = 8, and shrinkage = 0.02202, with eigenvalues 0.0002652, -0.0001944 and -0.0004517. Since the eigenvalues have opposite signs, the stationary point is a saddle point. Contour and perspective plots are shown in Figures C.4 and C.5 in the appendix. These plots indicate that the optimum may lie outside the design region.

The response surface suggests following the direction of the eigenvector corresponding to the positive eigenvalue. However, after repeating the process of moving along the eigenvector and fitting a new second-order response surface two times, the stationary point was still a saddle point. Instead, a new second-order response surface was fitted with center point in the configuration which obtained highest BACC score from the previous central composite design, which is shown in Table C.3. The largest value of BACC was 0.6834 and was obtained with n.trees = 770, interaction.depth = 10, and shrinkage = 0.03940. Low and high levels for n.trees were 695 and 845, respectively, with axial points 640 and 900. For interaction.depth, the low and high levels were 9 and 11, with axial points 8 and 12. The low and high levels for shrinkage were set to 0.03440 and 0.04440, and axial points 0.03074 and 0.04806. The second-order response surface was fitted with $\alpha \approx \sqrt{3}$ and six center runs. In Appendix C.1, the mean BACC value after 5-fold cross-validation of each configuration is presented in Table C.4, and the model summary is displayed in Listing 5. There are no significant estimated coefficients at level 0.05. The p-value of the lack of fit test is 0.3307, which means there is no significant lack of fit at level 0.05.

The stationary point is estimated at n.trees = 759, interaction.depth = 10, and shrinkage = 0.03884, which after 5-fold cross-validation the following values were obtained, 0.6859, 0.6847, 0.6823, 0.6892 and 0.6685, resulting in a mean BACC of 0.6821. The eigenvalues were 0.00000000, −0.00040159 and −0.00124404, which, once again, means that the response surface is a ridge system.

Table 4.17: Chosen hyperparameters with optimal values for the oneMo model.

Hyperparameter	Optimal value
n.trees	759
interaction.depth	10
shrinkage	0.03884
bag.fraction	0.5
cutoff	0.4

The optimal hyperparameters found through response surface methodology are shown in Table 4.17. The value of bag.fraction is kept at its default value, while the rest were adjusted. The OneMo AdaBoost model was trained with these values for the hyperparameters and evaluated on the one_month_ahead test set. The resulting confusion matrix is shown in Table 4.18 together with the confusion matrix of the OneMo model with default values on the hyperparameters. The number of observations wrongly classified as passive, i.e., the top right element in the confusion matrices, decreased from 1750 to 764 with optimized hyperparameter values, and the number of correctly classified active observations increased from 1110 to 2096.

Table 4.18: Confusion matrices after using AdaBoost with default and optimized hyperparameter values to classify the one_month_ahead test set.

		Actual	
		0	1
Predicted	0	3365	1750
	1	761	1110

(a) Default

		Actual	
		0	1
Predicted	0	2636	764
	1	1490	2096

(b) Optimized

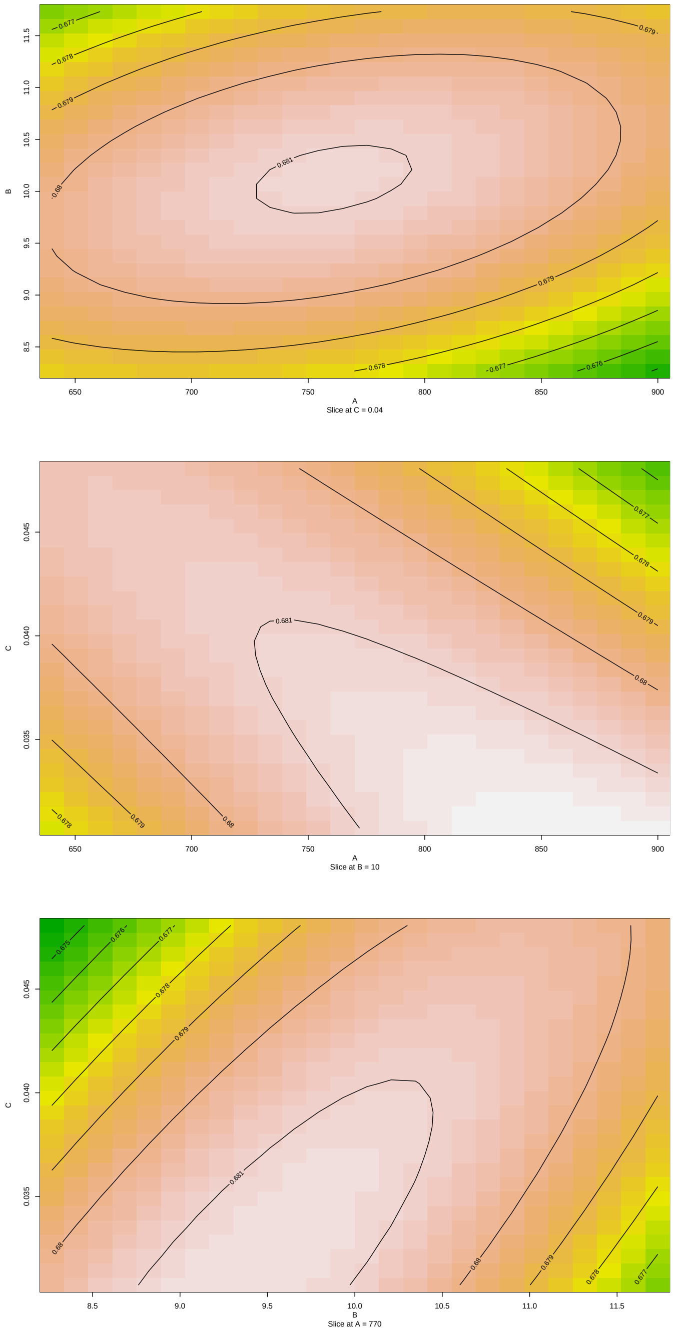


Figure 4.7: Contour plots of the third fitted second-order response surface model with AdaBoost.

Table 4.19 shows the resulting classification metrics of the default and optimized oneMo models evaluated on the test set. The specificity has decreased a substantial amount, however, the sensitivity has increased from 0.3881 to 0.7329 which leads to an increase in BACC score from 0.6018 to 0.6859. The BACC score obtained on the test set was in fact larger than the score obtained on the training set. A large value of sensitivity is of interest as this value reflects the portion of active customers the model can classify correctly.

Table 4.19: Reported metrics of AdaBoost on the one_month_ahead data frame, with default and optimized hyperparameter values.

Model	Sensitivity	Specificity	AUC	BACC
Default oneMo	0.3881	0.8156	0.6947	0.6018
Optimized oneMo	0.7329	0.6389	0.7577	0.6859

4.3.2 Optimizing the Twelve-Months-Ahead Model though Response Surface Methodology

Also, for the twelve_months_ahead data frame, a screening experiment was the first step in the optimization process. Similar to the optimization of the one_month_ahead data frame, a 2^{5-1} fractional factorial experiment was conducted with the same five hyperparameters; n.trees, interaction.depth, shrinkage, bag.fraction and cutoff. Based on the observations of the optimal cutoff for the logistic twelveMos model compared to the logistic oneMo model, the levels of the cutoff were decreased compared to the values used in the oneMo screening experiment. The low and high levels of the cutoff were decided to be 0.26 and 0.3, respectively, while the rest were kept the same. Table 4.20 shows the low and high levels of each hyperparameter used in the screening experiment for AdaBoost on the twelve_months_ahead data frame.

Table 4.20: Selected low and high levels of each hyperparameter.

Factor	Hyperparameter	Low Level	High Level
A	n.trees	100	600
B	interaction.depth	1	5
C	shrinkage	0.03	0.07
D	bag.fraction	0.4	0.6
E	cutoff	0.26	0.3

The 2^{5-1} fractional factorial experiment was conducted according to the setup from Table 2.2 and replicated one time. 5-fold cross-validation was performed for each configuration, with mean BACC as the response. Table C.6 in the appendix shows the results from the full experiment. Table 4.21 shows results of the same experiment, however, the reported BACC is the mean of the duplicate runs. The highest BACC score was 0.6772, obtained for level code *abd*.

Table 4.21: Results of the 2^{5-1} fractional factorial design of AdaBoost on the twelve_months_ahead data frame with mean BACC of the duplicate runs.

Experiment No.	A	B	C	D	E	Level code	Mean BACC
1	-1	-1	-1	-1	1	*e*	0.6085
2	1	-1	-1	-1	-1	*a*	0.6530
3	-1	1	-1	-1	-1	*b*	0.6692
4	1	1	-1	-1	1	*abe*	0.6733
5	-1	-1	1	-1	-1	*c*	0.6361
6	1	-1	1	-1	1	*ace*	0.6545
7	-1	1	1	-1	1	*bce*	0.6680
8	1	1	1	-1	-1	*abc*	0.6662
9	-1	-1	-1	1	-1	*d*	0.6075
10	1	-1	-1	1	1	*ade*	0.6573
11	-1	1	-1	1	1	*bde*	0.6687
12	1	1	-1	1	-1	*abd*	0.6772
13	-1	-1	1	1	1	*cde*	0.6456
14	1	-1	1	1	-1	*acd*	0.6526
15	-1	1	1	1	-1	*bcd*	0.6734
16	1	1	1	1	1	*abcde*	0.6672

A model with main effects and interactions was fitted to the 2^{5-1} fractional factorial design with BACC as response. Table 4.22 displays the coefficient estimates, and the full model summary is shown in Appendix C.2. Omitting the intercept, 9 of the coefficient estimates are considered significant at a level 0.05. Factors A and B have the smallest p-values and greatest absolute value in their estimated coefficients. Also, factor C and the interactions between A and B, A and C, and B and C have relatively small p-values. The normal Q-Q plot of the residuals in Figure C.7 is studied to assess if the assumption of independent and approximately normally distributed residuals holds for the model with main effects and interactions. The residuals fall nicely on a line, with only the end points deviating a bit. Figure C.8 shows the plotted residuals which look randomly distributed. Thus, the assumption of independent and normally distributed residuals seems to hold.

The main effects of factors A, B, C, D and E are shown in Figure 4.8. Like before, the small squares represent the mean BACC at the low and high levels of each factor, and the slope of the line which connects the two squares represents the effect on the response when one factor is changed from

Table 4.22: Coefficient estimates of the model with main effects and interactions fitted to the 2^{5-1} fractional factorial design for the twelve_months_ahead data frame with AdaBoost.

	Estimate	Std. Error	t value	Pr(> |t|)	
(Intercept)	6.549e-01	5.714e-04	1146.010	< 2e-16	***
A	7.774e-03	5.714e-04	13.605	3.27e-10	***
B	1.551e-02	5.714e-04	27.139	8.28e-15	***
C	3.068e-03	5.714e-04	5.369	6.27e-05	***
D	1.299e-03	5.714e-04	2.272	0.0372	*
E	5.007e-04	5.714e-04	0.876	0.3938	
A:B	-7.201e-03	5.714e-04	-12.603	1.01e-09	***
A:C	-5.606e-03	5.714e-04	-9.811	3.58e-08	***
A:D	-3.836e-04	5.714e-04	-0.671	0.5116	
A:E	-8.881e-05	5.714e-04	-0.155	0.8784	
B:C	-4.748e-03	5.714e-04	-8.310	3.38e-07	***
B:D	-7.852e-05	5.714e-04	-0.137	0.8924	
B:E	-1.590e-03	5.714e-04	-2.782	0.0133	*
C:D	4.566e-04	5.714e-04	0.799	0.4360	
C:E	3.659e-04	5.714e-04	0.640	0.5310	
D:E	3.029e-03	5.714e-04	5.301	7.17e-05	***

low to high level. Factor B has the steepest slope and A has the second steepest, where moving both factors from low to high level increases the response. This is also reflected in their estimated coefficients in Table 4.22, where A and B have the largest values. Factors D and E have relatively horizontal slopes compared to the other factors, which implies that the main effects of D and E alone have minor impact on the response.

Figure 4.9 shows interaction effects between each factor pair. Such plots are useful to analyze how the relationship between one factor and the response can be affected by a second factor. The interaction plot of A and B has the most non-parallel lines, which indicates that the interaction between A and B is the strongest. This is also seen in Table 4.22, where $A : B$ has the smallest p-value among the estimated interaction coefficients. The p-values of $A : C$ and $B : C$ are not much larger, which also are reflected in their interaction plots. For both interactions, the lines are non-parallel and intersect when A and B are moved to their high levels. Some of the other interactions have slightly non-parallel lines, e.g., the interaction plot of D and E shows intersecting lines. However, the main effects of these factors are not as significant and have smaller estimated coefficients in Table 4.22, and the small interaction between the two is not regarded important.

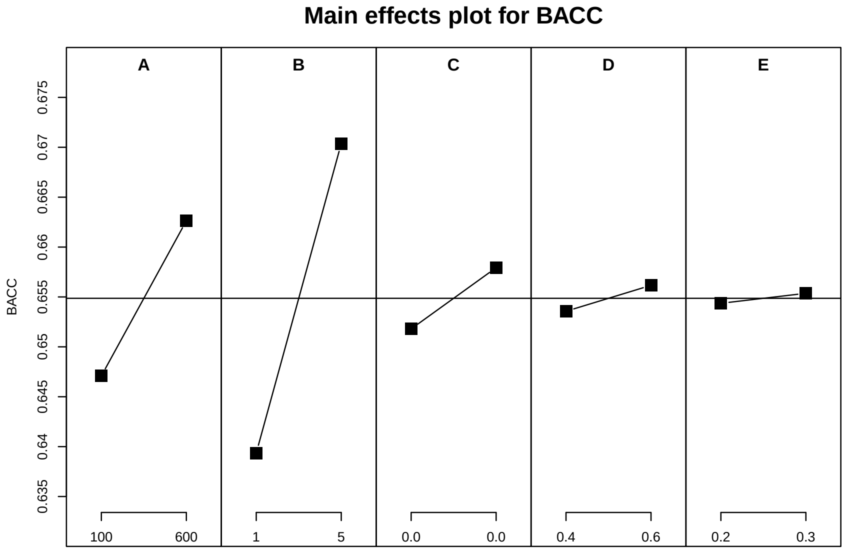


Figure 4.8: Main effects plot of factors A, B, C, D and E after performing the 2^{5-1} fractional factorial screening experiment for AdaBoost on the twelve_months_ahead data frame.

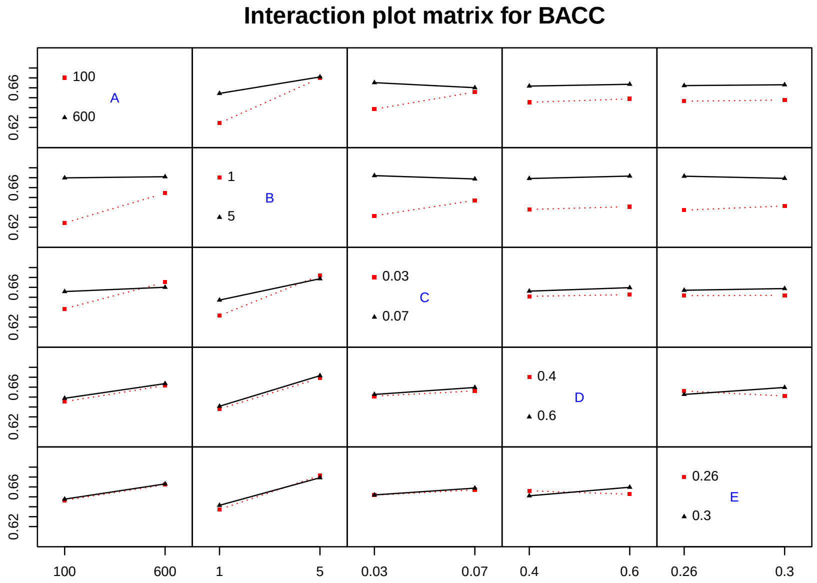


Figure 4.9: Interaction plots of factors A, B, C, D and E after performing the 2^{5-1} fractional factorial screening experiment for AdaBoost on the twelve_months_ahead data frame.

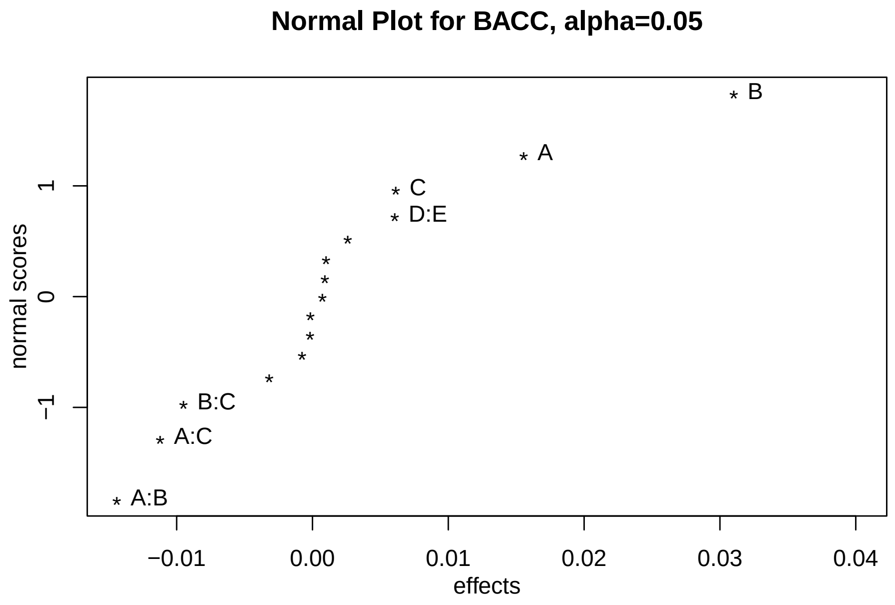


Figure 4.10: Normal plot of estimated main and interaction effects after performing the 2^{5-1} fractional factorial screening experiment for AdaBoost on the twelve_months_ahead data frame.

In a normal plot, factors with estimated coefficients close to zero fall on an approximate line, while those further from zero fall off the line and are regarded significant. In the normal plot shown in Figure 4.10, factors A and B fall furthest off the line with positive effects, in addition to C and the interaction $D : E$. Also, the interactions $A : B$, $A : C$ and $B : C$ fall off the line with negative effects, at a significance level at 0.05. All other main and interaction effects are here seemingly normally distributed with zero mean, and thus less important than these.

The levels of all factors were decided based on main and interaction effects. Firstly, bag.fraction and cutoff, factors D and E, had negligible effects on the response both as main effects and in interaction with other factors, except the interaction with each other. Therefore, bag.fraction was kept at its default value 0.5 and cutoff was set to 0.28. The factors with most impact on the response were A, B and C, i.e., n.trees, interaction.depth and shrinkage. All three factors had positive main effects, while the three interaction effects between them were negative. From the interaction plot between A and B, the highest BACC score was obtained with both at their high level. Thus, appropriate values of n.trees and interaction.depth are assumed around their high levels.

Since n.trees, interaction.depth and shrinkage were found to have the most impact on the response, new experiments are conducted along the gradient of these three to examine if the response can be improved further. From the estimated coefficients, the gradient of (n.trees, interaction.depth, shrinkage) is approximately $[0.5, 1, 0.2]$, which in original units means that when increasing interaction.depth by 2, n.trees should be increased by 125 and shrinkage by 0.004. In the interaction plots between A and C, and B and C, C at its low level yielded a slightly larger value of BACC. However, the difference is not substantial and the value of shrinkage is increased according to the gradient from the center value. Table 4.23 shows resulting mean BACC after 5-fold cross-validation at each step along the gradient, starting in 350 for n.trees, 3 for interaction.depth and 0.05 for shrinkage. The starting values correspond to the center value of each hyperparameter from the 2^{5-1} fractional factorial experiment. Cutoff was held constant at 0.28, and bag.fraction at 0.5. The largest BACC score was obtained at the center point. Thus, because of negative interaction effects, experiments along the gradient did not improve the response.

Table 4.23: Optimizing n.trees, interaction.depth and shrinkage along the gradient for the twelveMos model.

n.trees	interaction.depth	shrinkage	cutoff	BACC
350	3	0.050	0.28	**0.6786**
475	5	0.054	0.28	0.6677
600	7	0.058	0.28	0.6715
725	9	0.062	0.28	0.6614

A 2^3 factorial experiment with a total of eight center runs was conducted with factors for n.trees, interaction.depth and shrinkage. The center point was decided as the configuration yielding highest BACC score in Table 4.23. The cutoff was held constant at 0.28, and bag.fraction at 0.5. The low and high levels of n.trees were set to 350 ± 75, for interaction.depth the low level was set to 2 and high level to 4, and for shrinkage, the respective low and high levels were 0.05 ± 0.01. The mean BACC value after 5-fold cross-validation for each configuration is shown in Table C.7 in the appendix. The model summary is presented in Listing 7 in Appendix C.2. The p-value of the test for lack of fit is 0.1132, which means at a level 0.05 there is no significant lack of fit and the model with main effects and interactions is an appropriate approximation. The estimated coefficient of interaction.depth is the only significant effect at a level of 0.05.

Thus, new experiments were conducted along the gradient of interaction.depth, while the values of n.trees and shrinkage were kept constant at 350 and 0.05, respectively. Table 4.24 shows the resulting mean BACC value after 5-fold cross-validation at each step. The largest BACC value was obtained with n.trees = 360, interaction.depth = 5 and shrinkage = 0.0554.

A new 2^3 factorial experiment with eight center runs was conducted with center point at the configuration yielding the highest BACC score in Table 4.24. Low and high levels of n.trees was set at 350 ± 50, for interaction.depth they were set to 5 and 7, respectively, and 0.050 ± 0.005 for shrinkage. Table C.8 in the appendix shows the mean BACC value after 5-fold cross-validation for each configuration. The model summary is displayed in Listing 8 in Appendix C.2. The p-value of the test for lack of fit is 0.9361, which means that there is no evidence of lack of fit, and the

Table 4.24: Optimizing interaction.depth along the gradient for the twelveMos model.

n.trees	interaction.depth	shrinkage	cutoff	BACC
350	3	0.0500	0.28	0.6746
350	4	0.0500	0.28	0.6698
350	5	0.0500	0.28	0.6719
350	6	0.0500	0.28	**0.6758**
350	7	0.0500	0.28	0.6757
350	8	0.0500	0.28	0.6757
350	9	0.0500	0.28	0.6701

model with main effects and interactions is an appropriate approximation. None of the estimated coefficients are significant at a level 0.05, thus, there is no clear direction the experimental region could be moved to improve the response further. Optimal values for the hyperparameters were therefore decided to be the center values from the 2^3 factorial experiment, which are presented in Table 4.25.

Table 4.25: Chosen hyperparameters with optimal values for the twelveMos model.

Hyperparameter	Optimal value
n.trees	350
interaction.depth	6
shrinkage	0.050
bag.fraction	0.5
cutoff	0.28

The value of bag.fraction was kept at its default, while the four others were adjusted from their default value. The twelveMos AdaBoost model was trained with optimal hyperparameter values and then evaluated on the twelve_months_ahead test set. Table 4.26 presents the confusion matrix of twelveMos with default and optimized hyperparameters. The number of observations wrongly classified as passive has decreased from 659 to 234, and the number of correctly classified active observations has increased from 17 to 442. However, the number of wrongly classified active observations has also increased, from 8 to 508.

The classification metrics of the two twelveMos models, with default and optimized hyperparameters, evaluated on the test set is presented in Table 4.27. The default twelveMos model had an exceedingly small value of sensitivity while the specificity was close to 1. This indicates that the model classified almost all observations to belong to the passive class. This is also seen in the

Table 4.26: Confusion matrices after using AdaBoost with default and optimized hyperparameter values to classify the twelve_months_ahead test set.

		Actual	
		0	1
Predicted	0	1696	659
	1	8	17

(a) Default

		Actual	
		0	1
Predicted	0	1196	234
	1	508	442

(b) Optimized

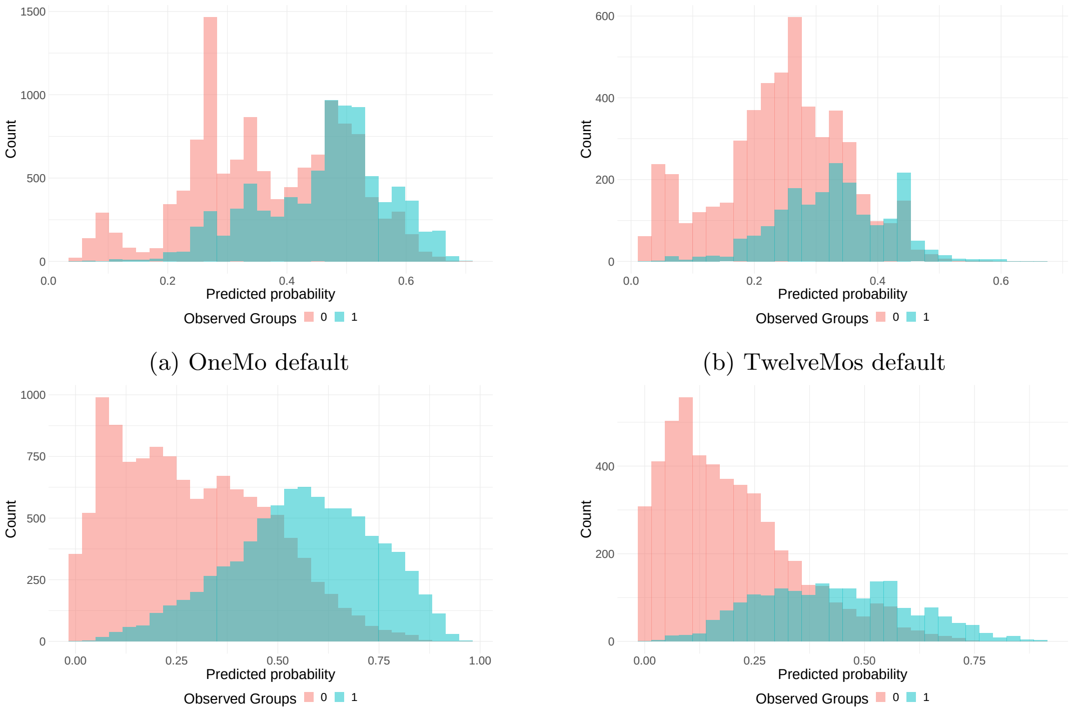
corresponding confusion matrix in Table 4.26a, where only 25 observations were classified as active. For the optimized twelveMos model, the specificity decreased from 0.9953 to 0.7019. However, the sensitivity increased from 0.0251 to 0.6538, which lead to an increase in the BACC to 0.6779.

Table 4.27: Reported metrics of AdaBoost on the twelve_months_ahead data frame, with default and optimized hyperparameter values.

Model	Sensitivity	Specificity	AUC	BACC
Default twelveMos	0.0251	0.9953	0.7135	0.5102
Optimized twelveMos	0.6538	0.7019	0.7503	0.6779

Figure 4.11 shows classification plot of the default oneMo and twelveMos models, and the optimized oneMo and optimized twelveMos models, before using the cutoff to classify the observations. The x-axis represents the model's predicted probability of an observation, and the y-axis is the count. Like the classification plots of the logistic models, the pink histograms visualize the true negative class, while the blue visualize the true positive class, i.e., passive and active observations, respectively. Specifically for the optimized oneMo model, the separation of the two classes is greatly improved compared to the default oneMo in Figure 4.11a and the oneMo logistic model, shown in Figure 4.2a. For the twelveMos model, there is still considerable overlap between the two classes, which is reflected in the confusion matrix and in the results in Table 4.27. However, the optimized twelveMos model is more confident in predicting the passive observations compared to the default twelveMos model.

Instead of classifying an observation as active or passive based on a chosen cutoff value, an alternative approach could be used. That is, if the predicted value is below a specific threshold, the customer will remain passive with a certain probability. Conversely, if the predicted value exceeds a certain threshold, the customer will become active with a certain probability. By adopting this perspective, it is possible to select a desired probability and determine the minimum predictive value, a threshold, at which a customer is likely to become active with that probability. This could also be applied to the logistic regression models.



(a) OneMo default

(b) TwelveMos default

(c) OneMo optimal

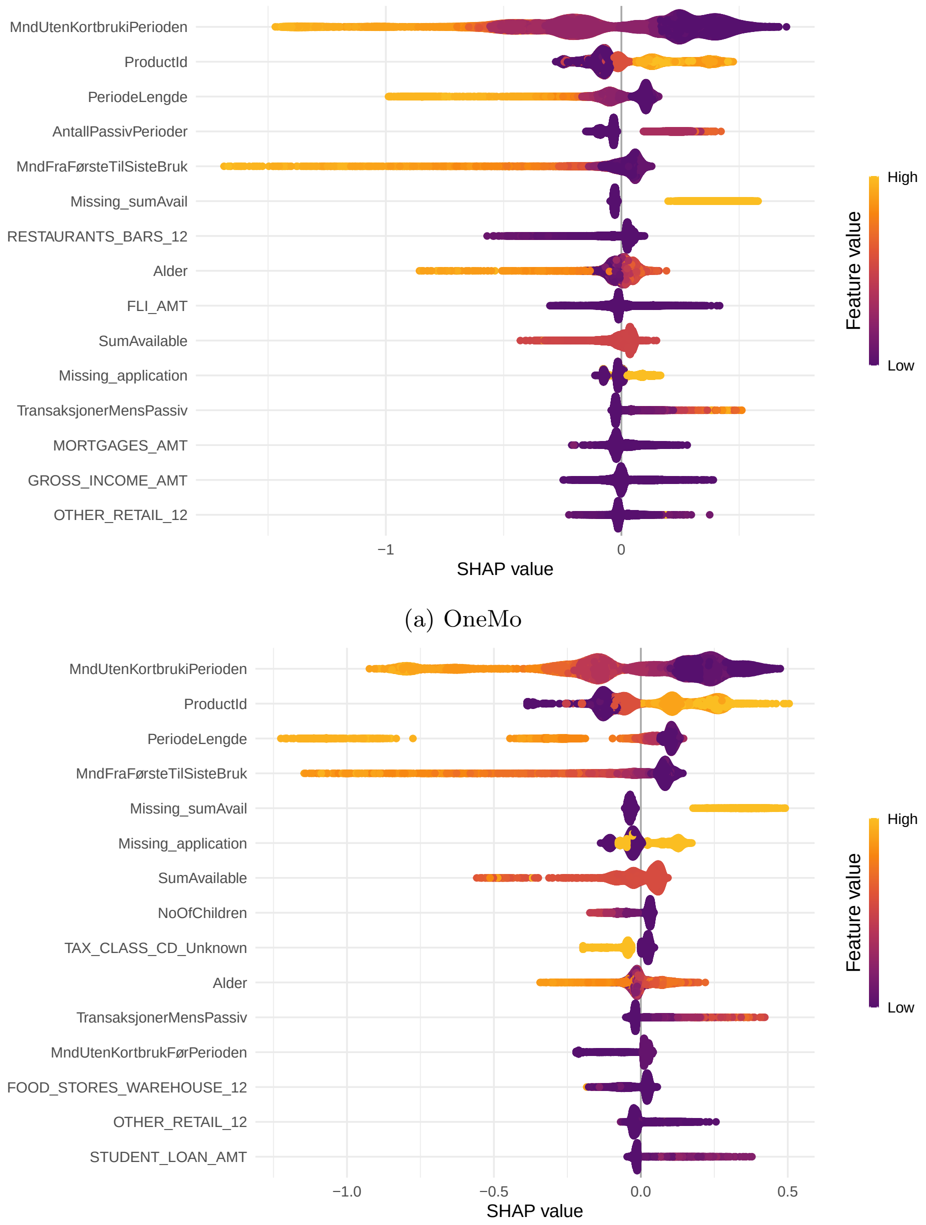
(d) TwelveMos optimal

Figure 4.11: Classification plot of the oneMo and twelveMos models with default and optimizes hyperparameters. Observations belonging to the negative class are shown in pink, and positive class as blue. The x-axis represents the predicted probability of belonging to the positive class according to the model, with optimized hyperparameters.

4.4 Variable Importance

The importance of unique features, or variables, and how they affect the response is examined for the two optimized AdaBoost models, both through the relative influence, which is calculated by the `gbm()` function, and SHAP. At each split in each tree constructed by `gbm()`, the improvement in split criterion is computed. The improvement each variable has contributed to, is averaged across all trees where that variable was used. The resulting value is the relative influence of the relevant variable.

Figure 4.12 visualizes beeswarm plots of the top 15 features in the optimized AdaBoost oneMo and twelveMos models. In a beeswarm plot, each observation in the data set is given an explanation represented by a dot on each feature row. The color of the dot represents the true feature value of that observation, and the dot's position on the x-axis is decided by its estimated Shapley value. Gatherings of dots along a feature row show the density of the observations' Shapley values.



(a) OneMo

(b) TwelveMos

Figure 4.12: "Beeswarm" plot. Visualization of the variables' influence on the response based on their Shapley values computed on the respective training sets.

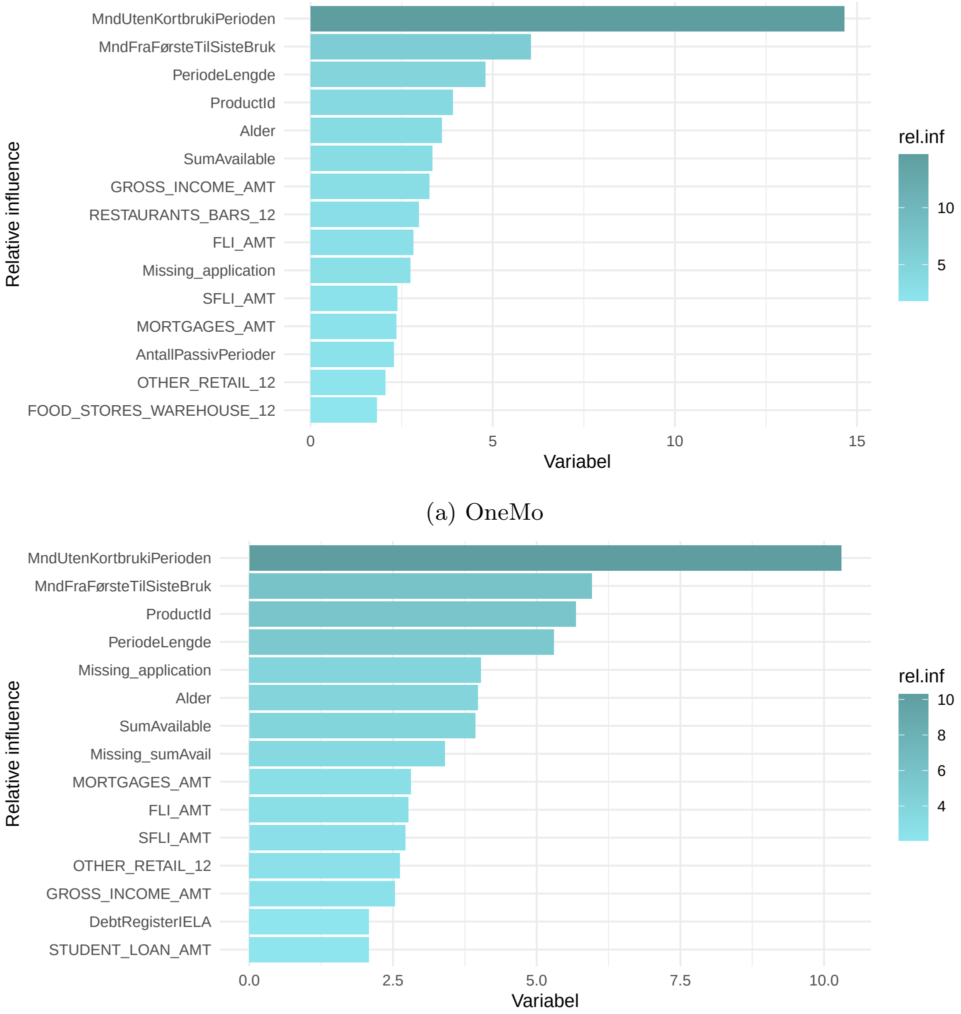
The top three feature is the same for the two models; MndUtenKortbrukiPerioden, ProdictId and PeriodeLengde. Further down we find several of the same variables, however the order is changed slightly. The Shapley value distribution of the same features is very similar for the two models. Consider MndUtenKortbrukiPerioden, observations with high values of that feature have negative Shapley values and are quite evenly spread out creating a line. The observations with lower values of MndUtenKortbrukiPerioden have larger Shapley values. In addition, there are several of these observations which create clusters along the feature row.

Figure 4.13 shows the relative influence of the top 15 explanatory variables in the optimized oneMo and twevleMos models. For the oneMo model, the top two features are the same as in the beeswarm plot in Figure 4.12a; MndUtenKortbrukiPerioden and PeriodeLengde. Among all 15 features visualized, 12 of them are presented in both the beeswarm and the relative influence plot. Even though the variable importance is computed differently for the two types of plots and the ordering of the features is different, this still shows that most of the same features are considered important in relation to the response. For the twelveMos model, the top four variables are the same in the beeswarm plot in Figure 4.12b and in the relative influence plot in Figure 4.13b, with some difference in the ordering. Here, ten of the fifteen features are the same in both plots with some difference in ordering. Again, this shows that the features regarded to have most influence on the response by the two methods are much the same.

Hence, the features considered most important based on Shapley values and relative influence are remarkably similar. However, if we compare the top 15 variables by relative influence from the oneMo AdaBoost model, to the 15 variables with lowest p-value from the oneMo logistic model, only five variables are listed to have significant effect on the response in both models. In the same situation for the respective twelveMos models, six of the same variables are considered to have significant impact on the response in the models. I.e., among the top fifteen most important variables, more than half of them are different for the two methods.

Figure 4.14 visualizes dependence plots of the feature Alder, which is the age of the customers, in the two models. Again, each observation is given an explanation represented by a dot. Here, the dot's position on the x-axis is that observation's, or customer's, age, and the y-axis is the computed Shapley value. The color of the dot here represents the observation's value of PeriodeLengde. The value of PeriodeLengde reflects the number of months in which information of the customers' credit card usage is used to predict their future activity. The two plots show a similar trend up to the age of 75, from there the Shapley value decrease for both models, however, the decrease is more apparent for the oneMo model than for the twelveMos model. This can be a result of less observations in the data set of the latter model.

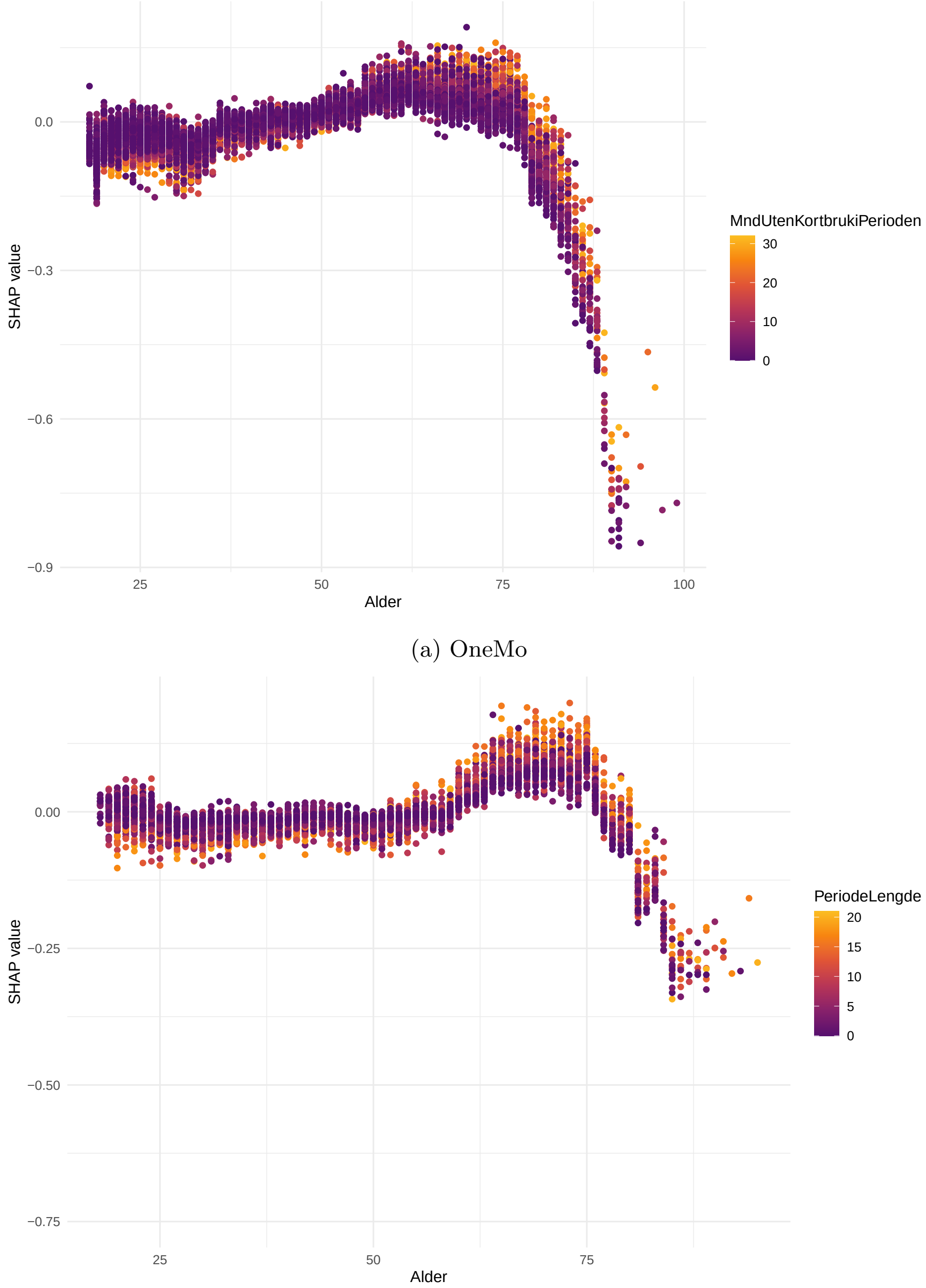
Specifically in Figure 4.14b of twelveMos, it is interesting to note that observations where the customer is younger and the value of PeriodeLengde is larger, then the Shapley value is smaller, and with a smaller value of PeriodeLengde, the Shapley value is generally a bit larger. While for customers of age 50 to 75 with larger value of PeriodeLengde, the Shapley value is also larger compared to customers of the same age with smaller value of PeriodeLengde.



(a) OneMo

(b) TwelveMos

Figure 4.13: Relative influence of the variables in the optimized AdaBoost models computed on the respective training sets.



(a) OneMo

(b) TwelveMos

Figure 4.14: Dependence plot of age where the color represent the value of PeriodeLengde, computed on the respective training sets.

Chapter 5

Discussion and Concluding Remarks

In this thesis, the predictive performance of logistic regression and AdaBoost have been investigated on imbalanced data sets. The main objective has been to predict and classify credit card use of passive customers. Logistic regression was applied on four distinct data sets to predict one, three, six and twelve months ahead, while AdaBoost was applied on two data sets to predict one and twelve months ahead. Different techniques were tested to improve the response; balanced accuracy (BACC).

First, logistic regression was fitted to the four data sets of one, three, six and twelve months ahead with cutoff = 0.5. The model that predicted one month ahead, oneMo, obtained the highest BACC score of 0.6181, and the longer ahead the models predicted, the more decreased the BACC score. The twelveMos model obtained a BACC score of 0.5717. This was expected as the one month ahead data set was the least imbalanced with 41% belonging to the positive class, while in the other data sets the level of imbalance increased together with the number of months to predict ahead. Feature selection through backward elimination was tested on all four models, both with BIC and AIC. However, neither method improved the BACC score of any of the models. Next, the cutoff values were optimized. For all four models, the resulting cutoff value was close to the percentage of minority instances in the respective data sets. With optimized cutoff value, the BACC score increased for all models, in addition to greatly improved sensitivity values. The resulting oneMo model had a BACC score of 0.6486, and 0.6362 for the twelveMos model.

AdaBoost was the second method fitted to the one and twelve months ahead training sets. The various levels of imbalance in the two data sets became obvious when AdaBoost was trained with default hyperparameter values. The oneMo model obtained a BACC of 0.6018 with quite small value of sensitivity and quite large value of specificity, while the twelveMos model obtained a BACC score of 0.5102 with an extremely small value of sensitivity and a specificity value close to 1. Then, four hyperparameters in addition to the cutoff value were optimized for both oneMo and twelveMos. For the oneMo model, the optimization resulted in an improved BACC score of 0.6859, and the sensitivity increased from 0.3881 to 0.7329. This means that the model's overall performance and ability to correctly classify the minority class were greatly improved. For the twelveMos model, both sensitivity and BACC increased a considerable amount with a resulting BACC of 0.6779 and sensitivity of 0.6538.

For the logistic models with optimized cutoff, there was a slight difference in the overall performance. The BACC score of the oneMo, threeMos and sixMos differed only with thousandths, and of the twelveMos with hundredths. Beforehand, the predictive performance of one month ahead was expected to be superior compared to twelve months ahead. This has turned out to not be the case. Also, for the optimized AdaBoost models, the difference in BACC score of oneMo and twelveMos was less than 0.01.

The distinction of logistic regression's and AdaBoost's predictive performance is noteworthy. The oneMo logistic model obtained a BACC score of 0.6486, while the oneMo AdaBoost model obtained 0.6859. Also, the sensitivity value of the logistic model was 0.6906, and for AdaBoost it was 0.7329. These results show quite impressive performance of AdaBoost compared to logistic regression, which was also visualized in the classification plots in Figure 4.11c versus Figure 4.2a. The two classes are better separated in the classification plot of the AdaBoost model than for the logistic model. The reason for this may be that the "S" shaped sigmoid function used in logistic regression fits the data set poorly, which limits the model's performance. If we consider the twelveMos models, AdaBoost still exceeds the overall performance of logistic regression, with a BACC score of 0.6779 over 0.6362. However, as seen in their classification plots, logistic in Figure 4.2d and AdaBoost in Figure 4.11d, there is a large amount of overlap between the two classes for both models. It is also worth noticing that the logistic twelveMos model obtained a larger value of sensitivity, 0.6672 versus 0.6538 for the AdaBoost model, which introduce another aspect to consider when working with a classification problem; choosing suitable metrics.

The choice of classification metric decides how the models are evaluated. In addition, when a model is optimized with a classification metric as response, the chosen metric also influences the resulting model. There is not a global agreement on which metric is best, and choosing a suitable metric depends on the problem at hand and if a response class is considered most valuable. In this thesis, all optimization of models was done to maximize the BACC score. BACC was chosen as it measures the model's overall performance and accounts for imbalanced response. Sensitivity has also been considered in the analysis and discussion of the results, as it measures a model's ability to classify the positive class, which in this case is the customers who become active. For the twelveMos models, where the AdaBoost model provides the best overall performance, while the logistic model performs better at classifying the positive class, the question of what is valued the most becomes relevant. Also, if predicting the positive class is valued over a better overall prediction, optimization of the cutoff values could have been done through a cost function instead of maximizing the BACC.

Another aspect to consider is the models' interpretability. This is often viewed in context of performance and the trade-off between the two. Often, machine learning methods, like AdaBoost, perform better than statistical methods, like logistic regression. In return, the latter are usually easier to interpret than machine learning methods. Interpretation is often an important aspect to consider in classification problems as it gives the end user valuable information of unique features' impact on the response. In this case, an interpretable model can help Sparebank 1 gain insight to which characteristics are repetitious for the customers who become active. However, several techniques within explainable AI have made it possible to explain the relationship between features and the response, even for "black box" machine learning methods. Among those techniques are Shapley values, used in this thesis to explain the AdaBoost models. As seen, the most important variables according to Shapley values and relative influence were mostly in agreement. However, the top 15 variables by relative influence from the AdaBoost models, differed quite a lot from the variables with lowest p-value in the corresponding logistic models. This demonstrates that the importance of features is calculated differently for the distinct methods.

5.1 Recommendations for Further Work

The goal of the models is essentially to model human behavior, which is a challenging task. For further work, including more variables, both personal and economical, could be beneficial to better capture the customers' behavior and improve the models. Adding personal variables that comply with GDPR can be challenging, however, educational level and the number of times they have moved in the last 5 years are some variables that could be of interest.

Even though optimizing hyperparameters led to improved performance, neither method performed optimally for this classification problem. Other techniques to tune the hyperparameters could be used. In addition, AdaBoost could be tested with different weak learners, e.g., support vector machines (SVM) which obtained best results in Vafeiadis et al. (2015). Adaptive boosting is also possible to perform with neural networks. Taherkhani et al. (2020) applied AdaBoost with a convolutional neural network on an imbalanced data set, which achieved almost 17% higher accuracy compared to classical AdaBoost with decision trees.

Regarding variable importance, it is also possible to compute Shapley values for logistic regression. These would be different from the interpretation of the estimated coefficients, and could be used to discuss the difference in how the methods calculate the importance of variables.

Bibliography

Aas, K., M. Jullum and A. Løland (2021). 'Explaining individual predictions when features are dependent: More accurate approximations to Shapley values'. In: *Artificial Intelligence* 298, p. 103502.

Alvarez-Melis, D. and T. S. Jaakkola (2018). 'On the Robustness of Interpretability Methods'. In: *arXiv preprint arXiv:1806.08049*.

Bartlett, P. et al. (1998). 'Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods'. In: *The Annals of Statistics* 26.5, pp. 1651–1686.

Bergstra, J. and Y. Bengio (2012). 'Random Search for Hyper-Parameter Optimization.' In: *Journal of Machine Learning Research* 13.2.

Box, G. E. P. and J. S. Hunter (1957). 'Multi-Factor Experimental Designs for Exploring Response Surfaces'. In: *The Annals of Mathematical Statistics*, pp. 195–241.

Dietterich, T. G. et al. (2002). 'Ensemble Learning'. In: *The Handbook of Brain Theory and NeuralNetworks* 2.1, pp. 110–125.

Fahrmeir, L. et al. (2022). 'Correction to: Regression'. In: *Regression: Models, Methods and Applications*. Springer.

Finanstilsynet (2022). *Tilleggsfordeler ved kredittopptak*. Høringsnotat.

Freund, Y., R. Schapire and N. Abe (1999). 'A Short Introduction to Boosting'. In: *Journal-Japanese Society For Artificial Intelligence* 14.771-780, p. 1612.

Freund, Y. and R. E. Schapire (1997). 'A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting'. In: *Journal of Computer and System Sciences* 55.1, pp. 119–139.

Geiler, L., S. Affeldt and M. Nadif (2022). 'An effective strategy for churn prediction and customer profiling'. In: *Data & Knowledge Engineering* 142, p. 102100.

Goos, P. (2002). 'Two-Level Factorial and Fractional Factorial Designs'. In: *The Optimal Design of Blocked and Split-Plot Experiments*. Springer, pp. 217–228.

Greenwell, B. et al. (2022). 'Generalized Boosted Regression Models - Package 'gbm''. In: *R package version* 2.8.

Harrell, F. E. (2015). 'Binary Logistic Regression'. In: *Regression modeling strategies*. Springer.

Hastie, T., R. Tibshirani and J. H. Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Vol. 2. Springer.

James, G. et al. (2021). *An Introduction to Statistical Learning with Applications in R*. 2nd ed. Springer. DOI: https://doi.org/10.1007/978-1-0716-1418-1.

Kim, G., B. K. Chae and D. L. Olson (2013). 'A support vector machine (SVM) approach to imbalanced datasets of customer responses: comparison with other customer response models'. In: *Service Business* 7, pp. 167–182.

Lujan-Moreno, G. A. et al. (2018). 'Design of experiments and response surface methodology to tune machine learning hyperparameters, with a random forest case-study'. In: *Expert Systems with Applications* 109, pp. 195–205.

Lundberg, S. M. and S. Lee (2017). 'A Unified Approach to Interpreting Model Predictions'. In: *Advances in Neural Information Processing Systems* 30.

Luque, A. et al. (2019). 'The impact of class imbalance in classification performance metrics based on the binary confusion matrix'. In: *Pattern Recognition* 91, pp. 216–231.

Mandrekar, J. N. (2010). 'Receiver Operating Characteristic Curve in Diagnostic Test Assessment'. In: *Journal of Thoracic Oncology* 5.9, pp. 1315–1316.

Miao, X. and H. Wang (2022). 'Customer Churn Prediction on Credit Card Services using Random Forest Method'. In: *2022 7th International Conference on Financial Innovation and Economic Development (ICFIED 2022)*. Atlantis Press, pp. 649–656.

Montgomery, D. C. (2017). *Design and Analysis of Experiments*. John Wiley & Sons.

Myers, R. H. and Montgomery (2002). *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. John Wiley & Sons.

AL-Najjar, D., N. Al-Rousan and H. AL-Najjar (2022). 'Machine Learning to Develop Credit Card Customer Churn Prediction'. In: *Journal of Theoretical and Applied Electronic Commerce Research* 17.4, pp. 1529–1542.

Nam, B. and R. B. D'Agostino (2002). 'Discrimination Index, the Area Under the ROC Curve'. In: *Goodness-of-Fit Tests and Model Validity*. Springer, pp. 267–279.

Nie, G. et al. (2011). 'Credit card churn forecasting by logistic regression and decision tree'. In: *Expert Systems with Applications* 38.12, pp. 15273–15285.

Patil, N., R. Lathi and V. Chitre (2012). 'Comparison of C5. 0 & CART Classification algorithms using pruning technique'. In: *International Journal of Engineering Research and Technology* 1.4, pp. 1–5.

Pokora, B. (Mar. 2023). *Credit Card Statistics And Trends 2023*. URL: https://www.forbes.com/advisor/credit-cards/credit-card-statistics/. (accessed: May 1st 2023).

Ribeiro, M. T., S. Singh and C. Guestrin (2016). '" Why Should I Trust You?" Explaining the Predictions of Any Classifier'. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144.

Rojas, R. et al. (2009). 'AdaBoost and the Super Bowl of Classifiers A Tutorial Introduction to Adaptive Boosting'. In: *Freie University, Berlin, Tech. Rep.*

Sagi, O. and L. Rokach (2018). 'Ensemble Learning: A Survey'. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.4, e1249.

Schapire, R. E. (2013). 'Explaining AdaBoost'. In: *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik*, pp. 37–52.

Schapire, R. E. and Y. Singer (1998). ‘Improved Boosting Algorithms Using Confidence-rated Predictions’. In: *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pp. 80–91.

Seeber, G. U. H. (1993). ‘Iteratively Re-Weighted Least Squares and its Implementation in GLIM4 and S’. In.

Stanberry, L. (2013). *Generalized Linear Models*. In: *Encyclopaedia of Systems Biology*.

Strømseng, S. B. (2022). *Prediction of Credit Card Activity for Passive Customers*. NTNU, project thesis.

Taherkhani, A., G. Cosma and T. M. McGinnity (2020). ‘AdaBoost-CNN: An adaptive boosting algorithm for convolutional neural networks to classify multi-class imbalanced datasets using transfer learning’. In: *Neurocomputing* 404, pp. 351–366.

Tsosie, C. (Nov. 2019). *3 Major Ways Credit Cards Changed This Decade*. URL: https://www.forbes.com/sites/clairetsosie/2019/11/27/3-major-ways-credit-cards-changed-this-decade/. (accessed: May 1st 2023).

Tyssedal, J. (n.d.). *Design of Experiments*. Last accessed 21 April 2023. URL: https://folk.ntnu.no/mettela/TMA4267/DOE.pdf.

Vafeiadis, T. et al. (2015). ‘A comparison of machine learning techniques for customer churn prediction’. In: *Simulation Modelling Practice and Theory* 55, pp. 1–9.

Yang, L. and A. Shami (2020). ‘On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice’. In: *Neurocomputing* 415, pp. 295–316.

Yuangyai, C. and H. B. Nembhard (2015). *Chapter 8. Design of Experiments: A Key to Innovation in Nanotechnology, Second Edition*.

Zeng, M. et al. (2016). ‘Effective prediction of three common diseases by combining SMOTE with Tomek links technique for imbalanced medical data’. In: *2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS)*, pp. 225–228. DOI: 10.1109/ICOACS.2016.7563084.

Appendix

A Variables with explanation

Table A.1: All variables in the original data set provided by Sparebank 1, with a short explanation.

Variable	**Explanation**
Utgangspunkt	Start date of the period which passive customers were predicted to become active or not
BK_ACCOUNT_ID	Internal account ID
Alder	Age of customer
Kjønn	Gender of customer
ProductId	Type of credit card
Revolver	Customer who pays nothing or part of the monthly invoice
Fullpayer	Customer who pays the whole monthly invoice
FørsteBrukt	Date of which the customer used their card for the first time
SisteTransaksjon	Date of the last transaction
SisteKortbruk	Date of the last card use
ApplicationSalesChannel	Sales channel which the customer applied for credit card
APPLIED_CREDIT_LIMIT_AMT	Credit limit the customer applied for in NOK

GRANTED_CREDIT_LIMIT_AMT	Credit limit the customer was granted in NOK
GROSS_INCOME_AMT	Customer's gross income in NOK
STUDENT_LOAN_AMT	Customer's amount of student loan in NOK
MORTGAGES_AMT	Customer's amount of mortages in NOK
EMPLOYMENT_TYPE_NAME	Categorical variable of customer's type of employment
EMPLOYMENT_DURATION_DESC	Categorical variable of the duration the customer has been in current employment
HABITATION_TYPE_NAME	Customer's type of habitation
MARITAL_STATUS_NAME	Customer's marital status
DebtRegisterNum	Number of different credit card debt and consumer loans the customer have
DebtRegisterIELA	Amount of credit card debt and consumer loans the customer have
TAX_CLASS_CD	Customer's tax class last year
LastTaxYear2_TAX_CLASS_CD	Customer's tax class two years ago
LastTaxYear3_TAX_CLASS_CD	Customer's tax class three years ago
HOMEOWNER_IND	Binary variable defining if the customer is a homeowner or not
HOUSING_COOPERATIVE_IND	Binary variable defining if the customer lives in a housing cooperative or not
NoOfChildren	Number of children the customer have
FLI_AMT	Simplified market liquidity indicator (customer's ability to pay)
SFLI_AMT	Simplified market liquidity indicator based in a 5% increase in interests
AIRLINE_12	Sum of transactions in given class last 12 months
ELECTRIC_APPLIANCE_12	Sum of transactions in given class last 12 months
FOOD_STORES_WAREHOUSE_12	Sum of transactions in given class last 12 months

HOTEL_MOTEL_12	Sum of transactions in given class last 12 months
HARDWARE_12	Sum of transactions in given class last 12 months
INTERIOR_FURNISHINGS_12	Sum of transactions in given class last 12 months
OTHER_RETAIL_12	Sum of transactions in given class last 12 months
OTHER_SERVICES_12	Sum of transactions in given class last 12 months
OTHER_TRANSPORT_12	Sum of transactions in given class last 12 months
RECREATION_12	Sum of transactions in given class last 12 months
RESTURANT_BARS_12	Sum of transactions in given class last 12 months
SPORTING_TOY_STORES_12	Sum of transactions in given class last 12 months
TRAVEL_AGENCIES_12	Sum of transactions in given class last 12 months
VEHICLES_12	Sum of transactions in given class last 12 months
QUASI_CASH_12	Sum of transactions in given class last 12 months
AIRLINE_3	Sum of transactions in given class last 3 months
ELECTRIC_APPLIANCE_3	Sum of transactions in given class last 3 months
FOOD_STORES_WAREHOUSE_3	Sum of transactions in given class last 3 months
HOTEL_MOTEL_3	Sum of transactions in given class last 3 months
HARDWARE_3	Sum of transactions in given class last 3 months
INTERIOR_FURNISHINGS_3	Sum of transactions in given class last 3 months
SumPaidToCCL12	Sum paid from bank account to known external credit card accounts last 12 months
SumPaidToRepaymentLoan12	Sum paid from bank account to known external repayment loan accounts last 12 months
CountPaidToRepaymentLoan12	Number of payments from bank account to known external repayment loan accounts last 12 months
CountPaidToCCL12	Number of payments from bank account to known external credit card accounts last 12 months
CountDistinctPaidToRepaymentLoan12	Number of payments from bank account to known distinct external repayment loan accounts last 12 months

CountDistinctPaidToCCL12	Number of payments from bank account to known distinct external credit card accounts last 12 months
CountRoundPaidToRepaymentLoan12	Number of round (whole 100 nok) payments from bank account to known distinct external repayment loan accounts last 12 months
CountRoundPaidToCCL12	Number of round (whole 100 nok) payments from bank account to known external credit card accounts last 12 months
AktivEtterPassiv	Binary response variable, defines if the customer became active or not within the relevant period

B Model Output from R for Logistic Regression

B.1 Full Logistic Models

One Month Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ ., family = binomial,
    data = oneMo_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9936  -1.0114  -0.6334   1.1425   2.9078

Coefficients:
                                         Estimate Std. Error z value Pr(>|z|)
(Intercept)                            -6.651e-01  2.320e-01  -2.866 0.004153 **
Alder                                   2.408e-03  1.383e-03   1.742 0.081578 .
Kjønn                                  -2.216e-02  3.146e-02  -0.704 0.481176
ProductId                               3.475e-02  2.179e-03  15.948  < 2e-16 ***
TransaksjonerMensPassiv                 1.867e-02  7.553e-03   2.472 0.013452 *
AntallPassivPerioder                    2.493e-01  1.200e-01   2.078 0.037754 *
MndUtenKortbrukiPerioden               -9.872e-02  2.579e-03 -38.273  < 2e-16 ***
MndFraFørsteTilSisteBruk               -1.849e-02  2.720e-03  -6.797 1.07e-11 ***
MndUtenKortbrukFørPerioden             -1.466e-02  2.323e-02  -0.631 0.528063
APPLIED_CREDIT_LIMIT_AMT                7.916e-02  4.459e-02   1.776 0.075814 .
GRANTED_CREDIT_LIMIT_AMT               -4.343e-02  4.460e-02  -0.974 0.330209
GROSS_INCOME_AMT                        1.947e+00  2.868e+00   0.679 0.497221
STUDENT_LOAN_AMT                        2.838e-02  1.724e-02   1.646 0.099698 .
MORTGAGES_AMT                           1.707e-02  2.565e-02   0.665 0.505878
DebtRegisterNum                        -3.388e-03  1.637e-02  -0.207 0.836083
DebtRegisterIELA                        2.940e-08  2.603e-07   0.113 0.910081
HOMEOWNER_IND                          -8.571e-03  2.837e-02  -0.302 0.762537
HOUSING_COOPERATIVE_IND                -1.381e-02  1.997e-02  -0.691 0.489358
NoOfChildren                            4.537e-03  2.125e-02   0.214 0.830885
FLI_AMT                                -6.021e-01  9.321e-01  -0.646 0.518311
SFLI_AMT                                5.817e-01  9.290e-01   0.626 0.531240
SumAvailable                           -8.240e-03  2.124e-02  -0.388 0.698061
Applied_vs_Granted                     -1.217e-02  2.419e-02  -0.503 0.614985
SumPaidToCCL12                         -4.741e-02  2.006e-02  -2.364 0.018089 *
SumPaidToRepaymentLoanL12              -2.445e-03  2.341e-02  -0.104 0.916847
```

```
CountPaidToRepaymentLoanL12             -1.350e+01  1.156e+02  -0.117 0.907021
CountPaidToCCL12                        -4.068e-02  2.903e-01  -0.140 0.888543
CountDistinctPaidToRepaymentLoanL12      1.343e+01  1.155e+02   0.116 0.907391
CountDistinctPaidToCCL12                 5.764e-02  2.902e-01   0.199 0.842543
CountRoundPaidToRepaymentLoanL12        -1.371e-02  1.827e-02  -0.750 0.453024
CountRoundPaidToCCL12                    1.228e-03  1.968e-02   0.062 0.950247
AIRLINE_12                              -7.857e-02  2.695e-02  -2.915 0.003553 **
ELECTRIC_APPLIANCE_12                    6.040e-02  2.265e-02   2.667 0.007652 **
FOOD_STORES_WAREHOUSE_12                 3.897e-02  2.272e-02   1.715 0.086334 .
HOTEL_MOTEL_12                           7.666e-02  2.754e-02   2.784 0.005374 **
HARDWARE_12                              2.162e-02  2.192e-02   0.986 0.324053
INTERIOR_FURNISHINGS_12                  3.105e-02  2.210e-02   1.405 0.160086
OTHER_RETAIL_12                          1.455e-02  1.667e-02   0.873 0.382538
OTHER_SERVICES_12                        4.230e-03  1.569e-02   0.270 0.787455
OTHER_TRANSPORT_12                      -4.669e-02  1.948e-02  -2.396 0.016556 *
RECREATION_12                            2.865e-03  1.621e-02   0.177 0.859747
RESTAURANTS_BARS_12                     -1.546e-01  2.230e-02  -6.932 4.14e-12 ***
SPORTING_TOY_STORES_12                   6.087e-03  1.516e-02   0.401 0.688122
TRAVEL_AGENCIES_12                      -6.494e-02  1.663e-02  -3.904 9.46e-05 ***
VEHICLES_12                              4.224e-02  1.552e-02   2.721 0.006506 **
QUASI_CASH_12                           -3.220e-02  1.658e-02  -1.942 0.052124 .
AIRLINE_3                                3.220e-02  2.519e-02   1.278 0.201129
ELECTRIC_APPLIANCE_3                     3.111e-03  2.141e-02   0.145 0.884508
FOOD_STORES_WAREHOUSE_3                  1.350e-02  1.987e-02   0.679 0.497014
HOTEL_MOTEL_3                           -8.952e-02  2.684e-02  -3.335 0.000854 ***
HARDWARE_3                              -4.816e-03  2.088e-02  -0.231 0.817605
INTERIOR_FURNISHINGS_3                   6.811e-03  2.111e-02   0.323 0.746924
AktiviPerioden                           6.838e-01  1.371e-01   4.988 6.11e-07 ***
Missing_purchaseSeg                     -5.381e-01  8.228e-02  -6.539 6.18e-11 ***
Missing_application                      5.609e-01  1.239e-01   4.527 5.99e-06 ***
Missing_purchaseHist                    -4.922e-02  5.233e-02  -0.940 0.346992
Missing_debt                             5.216e-02  4.401e-02   1.185 0.235947
Missing_sumAvail                         9.225e-01  6.669e-02  13.833  < 2e-16 ***
ApplicationSalesChannel_Autentisert_web -3.802e-02  6.700e-02  -0.567 0.570382
ApplicationSalesChannel_Kredittbanken   -2.897e+00  7.414e-01  -3.907 9.33e-05 ***
ApplicationSalesChannel_Mobilbank        1.756e-01  5.097e-02   3.445 0.000570 ***
ApplicationSalesChannel_Nettbank         2.846e-01  4.976e-02   5.720 1.07e-08 ***
ApplicationSalesChannel_Responsside      1.462e+01  3.247e+02   0.045 0.964081
EMPLOYMENT_TYPE_NAME_AT_HOME            -1.246e+01  3.247e+02  -0.038 0.969395
EMPLOYMENT_TYPE_NAME_DISABILITY_        -1.247e+01  3.247e+02  -0.038 0.969381
PENSIONER
EMPLOYMENT_TYPE_NAME_OTHER              -1.262e+01  3.247e+02  -0.039 0.969005
```

```
EMPLOYMENT_TYPE_NAME_RETIREE            -1.254e+01  3.247e+02  -0.039 0.969195
EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED       2.319e-02  1.090e-01   0.213 0.831442
EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY    -1.089e+01  3.247e+02  -0.034 0.973246
EMPLOYMENT_TYPE_NAME_STUDENT            -1.220e+01  3.247e+02  -0.038 0.970041
EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE       1.555e-01  8.970e-02   1.734 0.082917 .
EMPLOYMENT_TYPE_NAME_UNEMPLOYED         -1.318e+01  3.247e+02  -0.041 0.967617
EMPLOYMENT_DURATION_DESC_Between_1_and_  1.677e-02  5.680e-02   0.295 0.767837
3_years
EMPLOYMENT_DURATION_DESC_Less_than_1_    6.894e-02  6.723e-02   1.025 0.305156
year
EMPLOYMENT_DURATION_DESC_Not_set         1.237e+01  3.247e+02   0.038 0.969604
HABITATION_TYPE_NAME_APARTMENT          -1.253e-01  1.032e-01  -1.214 0.224803
HABITATION_TYPE_NAME_OTHER              -2.013e-01  1.059e-01  -1.901 0.057342 .
HABITATION_TYPE_NAME_PARENTS            -2.105e-01  9.284e-02  -2.268 0.023354 *
HABITATION_TYPE_NAME_RENTER             -2.837e-01  7.182e-02  -3.951 7.79e-05 ***
MARITAL_STATUS_NAME_COHABITING           1.086e-01  5.437e-02   1.998 0.045766 *
MARITAL_STATUS_NAME_DIVORCED             5.517e-03  9.374e-02   0.059 0.953065
MARITAL_STATUS_NAME_MARRIED              7.420e-02  7.526e-02   0.986 0.324218
MARITAL_STATUS_NAME_WIDOWED             -5.724e-02  1.087e-01  -0.527 0.598468
TAX_CLASS_CD_0                          -1.790e-01  3.203e-01  -0.559 0.576325
TAX_CLASS_CD_Unknown                    -2.912e-01  1.106e-01  -2.634 0.008437 **
LastTaxYear2_TAX_CLASS_CD_0             -6.301e-01  4.894e-01  -1.288 0.197913
LastTaxYear2_TAX_CLASS_CD_1E            -9.679e-02  1.311e-01  -0.738 0.460271
LastTaxYear2_TAX_CLASS_CD_2              2.609e-01  5.655e-01   0.461 0.644510
LastTaxYear2_TAX_CLASS_CD_2F             3.959e-01  2.627e-01   1.507 0.131760
LastTaxYear2_TAX_CLASS_CD_Unknown        7.833e-02  1.152e-01   0.680 0.496526
LastTaxYear3_TAX_CLASS_CD_0              1.157e-01  7.602e-01   0.152 0.879038
LastTaxYear3_TAX_CLASS_CD_1              1.361e-01  9.289e-02   1.465 0.143017
LastTaxYear3_TAX_CLASS_CD_1E             2.040e-01  1.516e-01   1.345 0.178503
LastTaxYear3_TAX_CLASS_CD_2              2.528e-01  3.876e-01   0.652 0.514175
LastTaxYear3_TAX_CLASS_CD_2F             5.464e-01  2.113e-01   2.587 0.009695 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 27833  on 20600  degrees of freedom
Residual deviance: 25470  on 20506  degrees of freedom
AIC: 25660

Number of Fisher Scoring iterations: 11
```

Three Months Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ ., family = binomial,
    data = threeMos_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8446  -0.9425  -0.6503   1.1687   2.4216

Coefficients:
                                           Estimate Std. Error z value Pr(>|z|)
(Intercept)                              -9.809e-01  2.867e-01  -3.422 0.000622 ***
Alder                                     4.321e-03  1.596e-03   2.708 0.006763 **
Kjønn                                     6.314e-02  3.702e-02   1.706 0.088072 .
ProductId                                 4.353e-02  2.569e-03  16.941  < 2e-16 ***
TransaksjonerMensPassiv                   2.638e-02  8.478e-03   3.111 0.001864 **
AntallPassivPerioder                      2.756e-01  1.568e-01   1.758 0.078778 .
MndUtenKortbrukiPerioden                 -1.010e-01  3.117e-03 -32.409  < 2e-16 ***
MndFraFørsteTilSisteBruk                 -2.787e-02  3.567e-03  -7.813 5.59e-15 ***
MndUtenKortbrukFørPerioden               -6.354e-02  2.796e-02  -2.273 0.023054 *
APPLIED_CREDIT_LIMIT_AMT                  1.542e-02  5.392e-02   0.286 0.774870
GRANTED_CREDIT_LIMIT_AMT                  2.106e-02  5.398e-02   0.390 0.696495
GROSS_INCOME_AMT                          3.398e+00  3.548e+00   0.958 0.338155
STUDENT_LOAN_AMT                          2.611e-02  1.989e-02   1.312 0.189373
MORTGAGES_AMT                            -1.852e-03  4.518e-02  -0.041 0.967311
DebtRegisterNum                          -1.301e-02  1.948e-02  -0.668 0.504238
DebtRegisterIELA                         -2.418e-07  3.294e-07  -0.734 0.462890
HOMEOWNER_IND                             8.500e-03  3.396e-02   0.250 0.802380
HOUSING_COOPERATIVE_IND                   9.571e-03  2.368e-02   0.404 0.686099
NoOfChildren                             -2.342e-02  2.509e-02  -0.934 0.350550
FLI_AMT                                  -2.614e-01  1.982e+00  -0.132 0.895047
SFLI_AMT                                  2.330e-01  1.979e+00   0.118 0.906268
SumAvailable                             -3.755e-02  3.899e-02  -0.963 0.335592
Applied_vs_Granted                       -2.445e-03  3.015e-02  -0.081 0.935375
SumPaidToCCL12                           -2.112e-02  2.376e-02  -0.889 0.374200
SumPaidToRepaymentLoanL12                 2.297e-02  2.773e-02   0.828 0.407491
CountPaidToRepaymentLoanL12               3.069e+00  5.216e+01   0.059 0.953075
CountPaidToCCL12                         -3.889e-02  2.984e-01  -0.130 0.896304
CountDistinctPaidToRepaymentLoanL12      -3.131e+00  5.209e+01  -0.060 0.952062
CountDistinctPaidToCCL12                  1.320e-02  2.982e-01   0.044 0.964698
CountRoundPaidToRepaymentLoanL12         -3.016e-02  2.215e-02  -1.361 0.173418
```

```
CountRoundPaidToCCL12                  6.311e-03  2.383e-02   0.265 0.791152
AIRLINE_12                            -2.797e-02  3.060e-02  -0.914 0.360712
ELECTRIC_APPLIANCE_12                  2.931e-02  2.665e-02   1.100 0.271381
FOOD_STORES_WAREHOUSE_12              -5.194e-02  2.852e-02  -1.821 0.068597 .
HOTEL_MOTEL_12                         2.355e-02  3.080e-02   0.765 0.444458
HARDWARE_12                            6.003e-03  2.475e-02   0.243 0.808362
INTERIOR_FURNISHINGS_12                1.149e-03  2.541e-02   0.045 0.963932
OTHER_RETAIL_12                        3.423e-02  1.895e-02   1.806 0.070888 .
OTHER_SERVICES_12                     -4.823e-03  1.839e-02  -0.262 0.793062
OTHER_TRANSPORT_12                    -1.992e-02  2.140e-02  -0.931 0.351906
RECREATION_12                          4.369e-02  1.912e-02   2.285 0.022334 *
RESTAURANTS_BARS_12                   -5.049e-02  2.401e-02  -2.103 0.035477 *
SPORTING_TOY_STORES_12                -1.862e-02  1.807e-02  -1.030 0.302895
TRAVEL_AGENCIES_12                    -3.541e-02  1.897e-02  -1.867 0.061919 .
VEHICLES_12                            4.256e-02  1.775e-02   2.398 0.016497 *
QUASI_CASH_12                         -2.363e-02  1.946e-02  -1.214 0.224688
AIRLINE_3                             -9.022e-03  2.994e-02  -0.301 0.763185
ELECTRIC_APPLIANCE_3                   1.206e-02  2.594e-02   0.465 0.642063
FOOD_STORES_WAREHOUSE_3                7.080e-02  2.367e-02   2.991 0.002784 **
HOTEL_MOTEL_3                          4.054e-02  2.856e-02   1.420 0.155727
HARDWARE_3                             6.586e-03  2.413e-02   0.273 0.784925
INTERIOR_FURNISHINGS_3                -2.326e-03  2.441e-02  -0.095 0.924098
AktiviPerioden                         7.427e-01  1.763e-01   4.213 2.52e-05 ***
Missing_purchaseSeg                   -6.096e-01  9.863e-02  -6.181 6.39e-10 ***
Missing_application                    6.312e-01  1.506e-01   4.191 2.78e-05 ***
Missing_purchaseHist                  -3.546e-02  6.431e-02  -0.551 0.581380
Missing_debt                           1.624e-01  5.291e-02   3.069 0.002145 **
Missing_sumAvail                       9.106e-01  7.702e-02  11.824  < 2e-16 ***
ApplicationSalesChannel_Autentisert_web -1.235e-01  7.983e-02  -1.547 0.121771
ApplicationSalesChannel_Kredittbanken -1.622e+00  1.171e+00  -1.385 0.166127
ApplicationSalesChannel_Mobilbank      1.985e-01  6.269e-02   3.167 0.001541 **
ApplicationSalesChannel_Nettbank       3.753e-01  6.020e-02   6.234 4.56e-10 ***
ApplicationSalesChannel_Responsside    1.375e+01  1.970e+02   0.070 0.944362
EMPLOYMENT_TYPE_NAME_AT_HOME          -1.149e+01  1.970e+02  -0.058 0.953488
EMPLOYMENT_TYPE_NAME_DISABILITY_      -1.165e+01  1.970e+02  -0.059 0.952852
PENSIONER
EMPLOYMENT_TYPE_NAME_OTHER            -1.122e+01  1.970e+02  -0.057 0.954585
EMPLOYMENT_TYPE_NAME_RETIREE          -1.170e+01  1.970e+02  -0.059 0.952649
EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED     1.019e-01  1.273e-01   0.800 0.423625
EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY  -1.042e+01  1.970e+02  -0.053 0.957809
EMPLOYMENT_TYPE_NAME_STUDENT          -1.116e+01  1.970e+02  -0.057 0.954808
EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE     1.399e-01  1.087e-01   1.288 0.197880
```

```
EMPLOYMENT_TYPE_NAME_UNEMPLOYED          -1.185e+01  1.970e+02  -0.060 0.952033
EMPLOYMENT_DURATION_DESC_Between_1_and_  -2.091e-02  6.983e-02  -0.299 0.764621
3_years
EMPLOYMENT_DURATION_DESC_Less_than_1_     6.290e-02  8.215e-02   0.766 0.443866
year
EMPLOYMENT_DURATION_DESC_Not_set          1.144e+01  1.970e+02   0.058 0.953688
HABITATION_TYPE_NAME_APARTMENT           -3.376e-02  1.223e-01  -0.276 0.782413
HABITATION_TYPE_NAME_OTHER               -1.730e-01  1.334e-01  -1.297 0.194654
HABITATION_TYPE_NAME_PARENTS             -1.361e-01  1.132e-01  -1.203 0.229019
HABITATION_TYPE_NAME_RENTER              -3.206e-01  8.782e-02  -3.651 0.000261 ***
MARITAL_STATUS_NAME_COHABITING            1.286e-01  6.659e-02   1.931 0.053529 .
MARITAL_STATUS_NAME_DIVORCED              5.379e-02  1.113e-01   0.483 0.629016
MARITAL_STATUS_NAME_MARRIED               8.576e-02  9.062e-02   0.946 0.343951
MARITAL_STATUS_NAME_WIDOWED              -1.600e-01  1.309e-01  -1.222 0.221569
TAX_CLASS_CD_0                           -8.166e-01  5.159e-01  -1.583 0.113469
TAX_CLASS_CD_Unknown                     -2.871e-01  1.344e-01  -2.137 0.032631 *
LastTaxYear2_TAX_CLASS_CD_0              -7.983e-01  5.924e-01  -1.347 0.177822
LastTaxYear2_TAX_CLASS_CD_1               5.929e-03  1.390e-01   0.043 0.965979
LastTaxYear2_TAX_CLASS_CD_1E             -2.036e-01  2.031e-01  -1.002 0.316313
LastTaxYear2_TAX_CLASS_CD_2               5.291e-01  5.742e-01   0.921 0.356834
LastTaxYear2_TAX_CLASS_CD_2F              1.032e-01  3.392e-01   0.304 0.760976
LastTaxYear3_TAX_CLASS_CD_0              -4.114e-01  1.121e+00  -0.367 0.713743
LastTaxYear3_TAX_CLASS_CD_1               2.146e-01  1.113e-01   1.928 0.053901 .
LastTaxYear3_TAX_CLASS_CD_1E              3.874e-01  1.799e-01   2.153 0.031306 *
LastTaxYear3_TAX_CLASS_CD_2               2.813e-01  4.526e-01   0.621 0.534313
LastTaxYear3_TAX_CLASS_CD_2F              7.049e-01  2.425e-01   2.906 0.003658 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 20282  on 15552  degrees of freedom
Residual deviance: 18530  on 15458  degrees of freedom
AIC: 18720

Number of Fisher Scoring iterations: 10
```

Six Months Ahead

```
Call:
```

```
glm(formula = as.factor(AktivEtterPassiv) ~ ., family = binomial,
    data = sixMos_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0313  -0.8492  -0.6250   1.1277   2.6728

Coefficients:
                                      Estimate Std. Error z value Pr(>|z|)
(Intercept)                         -1.458e+00  4.366e-01  -3.340 0.000839 ***
Alder                                5.984e-03  1.904e-03   3.143 0.001672 **
Kjønn                                2.943e-02  4.478e-02   0.657 0.510973
ProductId                            5.417e-02  3.163e-03  17.125  < 2e-16 ***
TransaksjonerMensPassiv              3.457e-02  1.064e-02   3.248 0.001163 **
AntallPassivPerioder                -1.211e-02  3.002e-01  -0.040 0.967830
MndUtenKortbrukiPerioden            -9.801e-02  3.914e-03 -25.045  < 2e-16 ***
MndFraFørsteTilSisteBruk            -4.482e-02  5.047e-03  -8.879  < 2e-16 ***
MndUtenKortbrukFørPerioden          -2.856e-02  3.778e-02  -0.756 0.449634
APPLIED_CREDIT_LIMIT_AMT             6.315e-02  6.503e-02   0.971 0.331527
GRANTED_CREDIT_LIMIT_AMT            -3.625e-02  6.499e-02  -0.558 0.576978
GROSS_INCOME_AMT                     5.774e+00  4.354e+00   1.326 0.184843
STUDENT_LOAN_AMT                     4.533e-02  2.379e-02   1.906 0.056695 .
MORTGAGES_AMT                       -1.675e-01  9.755e-02  -1.717 0.086008 .
DebtRegisterNum                     -7.251e-04  2.403e-02  -0.030 0.975922
DebtRegisterIELA                    -2.858e-07  3.555e-07  -0.804 0.421392
HOMEOWNER_IND                        5.801e-03  3.986e-02   0.146 0.884283
HOUSING_COOPERATIVE_IND              3.355e-02  2.863e-02   1.172 0.241304
NoOfChildren                        -6.566e-02  3.059e-02  -2.146 0.031854 *
FLI_AMT                              7.707e-01  3.600e-01   2.141 0.032292 *
SFLI_AMT                            -7.099e-01  3.486e-01  -2.036 0.041734 *
SumAvailable                        -2.305e-01  9.622e-02  -2.396 0.016577 *
Applied_vs_Granted                   2.904e-03  3.624e-02   0.080 0.936140
SumPaidToCCL12                      -9.743e-03  3.085e-02  -0.316 0.752143
SumPaidToRepaymentLoanL12            3.465e-02  4.024e-02   0.861 0.389169
CountPaidToRepaymentLoanL12         -1.458e+01  2.281e+02  -0.064 0.949031
CountPaidToCCL12                    -6.528e-01  9.656e-01  -0.676 0.498992
CountDistinctPaidToRepaymentLoanL12  1.451e+01  2.278e+02   0.064 0.949218
CountDistinctPaidToCCL12             6.207e-01  9.632e-01   0.644 0.519277
CountRoundPaidToRepaymentLoanL12    -8.831e-02  3.758e-02  -2.350 0.018787 *
CountRoundPaidToCCL12               -2.243e-02  3.169e-02  -0.708 0.479111
AIRLINE_12                           4.252e-02  3.402e-02   1.250 0.211451
ELECTRIC_APPLIANCE_12                3.605e-02  3.181e-02   1.133 0.257099
```

```
FOOD_STORES_WAREHOUSE_12                -4.835e-02  3.687e-02  -1.312 0.189678
HOTEL_MOTEL_12                           2.588e-02  3.735e-02   0.693 0.488289
HARDWARE_12                             -4.488e-02  3.529e-02  -1.272 0.203402
INTERIOR_FURNISHINGS_12                  1.634e-04  3.444e-02   0.005 0.996215
OTHER_RETAIL_12                          2.316e-02  2.167e-02   1.069 0.285255
OTHER_SERVICES_12                       -5.500e-03  2.245e-02  -0.245 0.806458
OTHER_TRANSPORT_12                      -1.420e-03  2.396e-02  -0.059 0.952750
RECREATION_12                            2.492e-02  2.122e-02   1.174 0.240256
RESTAURANTS_BARS_12                     -3.173e-02  2.601e-02  -1.220 0.222519
SPORTING_TOY_STORES_12                  -2.085e-02  2.237e-02  -0.932 0.351268
TRAVEL_AGENCIES_12                       2.445e-02  2.061e-02   1.186 0.235543
VEHICLES_12                              3.198e-02  2.117e-02   1.511 0.130900
QUASI_CASH_12                           -6.333e-03  2.398e-02  -0.264 0.791726
AIRLINE_3                               -4.305e-02  3.445e-02  -1.250 0.211345
ELECTRIC_APPLIANCE_3                     1.268e-02  3.086e-02   0.411 0.681189
FOOD_STORES_WAREHOUSE_3                  5.093e-02  2.940e-02   1.733 0.083162 .
HOTEL_MOTEL_3                            3.703e-02  3.481e-02   1.064 0.287490
HARDWARE_3                               5.253e-02  3.086e-02   1.702 0.088738 .
INTERIOR_FURNISHINGS_3                   1.410e-02  3.243e-02   0.435 0.663713
AktiviPerioden                           9.680e-01  3.179e-01   3.045 0.002326 **
Missing_purchaseSeg                     -9.925e-01  1.284e-01  -7.727 1.10e-14 ***
Missing_application                      5.712e-01  1.875e-01   3.046 0.002316 **
Missing_purchaseHist                     1.033e-01  8.506e-02   1.214 0.224570
Missing_debt                             1.917e-01  6.531e-02   2.934 0.003342 **
Missing_sumAvail                         1.051e+00  8.713e-02  12.064  < 2e-16 ***
ApplicationSalesChannel_Autentisert_web -2.067e-01  9.335e-02  -2.214 0.026813 *
ApplicationSalesChannel_Mobilbank        3.298e-01  7.894e-02   4.178 2.94e-05 ***
ApplicationSalesChannel_Nettbank         4.039e-01  7.553e-02   5.347 8.94e-08 ***
ApplicationSalesChannel_Responsside      1.568e+01  5.354e+02   0.029 0.976632
EMPLOYMENT_TYPE_NAME_AT_HOME            -4.061e-01  9.464e-01  -0.429 0.667895
EMPLOYMENT_TYPE_NAME_DISABILITY_        -1.221e-01  1.309e-01  -0.933 0.350822
PENSIONER
EMPLOYMENT_TYPE_NAME_OTHER               1.334e-01  2.555e-01   0.522 0.601594
EMPLOYMENT_TYPE_NAME_RETIREE            -2.219e-01  1.074e-01  -2.067 0.038774 *
EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED       2.804e-02  1.555e-01   0.180 0.856854
EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY    -1.207e+01  3.783e+02  -0.032 0.974548
EMPLOYMENT_TYPE_NAME_STUDENT             4.344e-01  1.236e-01   3.516 0.000439 ***
EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE       1.848e-01  1.367e-01   1.351 0.176535
EMPLOYMENT_TYPE_NAME_UNEMPLOYED          7.418e-02  5.506e-01   0.135 0.892836
EMPLOYMENT_DURATION_DESC_Between_1_and_  8.184e-02  8.735e-02   0.937 0.348803
3_years
EMPLOYMENT_DURATION_DESC_Less_than_1_    4.941e-02  1.027e-01   0.481 0.630291
```

```
year
HABITATION_TYPE_NAME_APARTMENT          -8.610e-02  1.514e-01  -0.569 0.569506
HABITATION_TYPE_NAME_OTHER              -1.834e-01  1.636e-01  -1.121 0.262260
HABITATION_TYPE_NAME_PARENTS            -2.807e-01  1.396e-01  -2.010 0.044379 *
HABITATION_TYPE_NAME_RENTER             -3.962e-01  1.088e-01  -3.641 0.000272 ***
MARITAL_STATUS_NAME_COHABITING           9.084e-02  8.234e-02   1.103 0.269936
MARITAL_STATUS_NAME_DIVORCED             1.512e-01  1.408e-01   1.074 0.282885
MARITAL_STATUS_NAME_MARRIED              1.026e-01  1.138e-01   0.901 0.367556
MARITAL_STATUS_NAME_WIDOWED             -1.757e-01  1.610e-01  -1.092 0.275026
TAX_CLASS_CD_0                          -1.318e-01  5.753e-01  -0.229 0.818781
TAX_CLASS_CD_Unknown                    -4.355e-01  1.670e-01  -2.607 0.009133 **
LastTaxYear2_TAX_CLASS_CD_0             -1.568e+00  1.072e+00  -1.463 0.143469
LastTaxYear2_TAX_CLASS_CD_1             -1.638e-01  1.717e-01  -0.954 0.340065
LastTaxYear2_TAX_CLASS_CD_1E            -2.659e-01  2.481e-01  -1.072 0.283743
LastTaxYear2_TAX_CLASS_CD_2              5.985e-01  7.590e-01   0.789 0.430345
LastTaxYear2_TAX_CLASS_CD_2F             4.478e-01  3.685e-01   1.215 0.224274
LastTaxYear3_TAX_CLASS_CD_0              2.009e-01  1.163e+00   0.173 0.862921
LastTaxYear3_TAX_CLASS_CD_1              3.231e-01  1.419e-01   2.277 0.022789 *
LastTaxYear3_TAX_CLASS_CD_1E             4.125e-01  2.193e-01   1.881 0.060008 .
LastTaxYear3_TAX_CLASS_CD_2              4.695e-01  4.852e-01   0.968 0.333232
LastTaxYear3_TAX_CLASS_CD_2F             5.253e-01  2.850e-01   1.843 0.065292 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14205  on 11631  degrees of freedom
Residual deviance: 12939  on 11539  degrees of freedom
AIC: 13125

Number of Fisher Scoring iterations: 12
```

Twelve Months Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ ., family = binomial,
    data = twelveMos_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
```

```
-3.4488  -0.8089  -0.5739   1.0233   2.8925

Coefficients:
                                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                             -3.126e+00  9.350e-01  -3.344 0.000826 ***
Alder                                    3.289e-03  2.547e-03   1.291 0.196648
Kjønn                                   -3.017e-02  5.931e-02  -0.509 0.610981
ProductId                                4.130e-02  4.729e-03   8.735  < 2e-16 ***
TransaksjonerMensPassiv                  4.804e-02  1.445e-02   3.325 0.000885 ***
AntallPassivPerioder                    -7.070e-01  7.724e-01  -0.915 0.360003
MndUtenKortbrukiPerioden                -1.249e-01  6.043e-03 -20.668  < 2e-16 ***
MndFraFørsteTilSisteBruk                -7.304e-02  8.410e-03  -8.685  < 2e-16 ***
MndUtenKortbrukFørPerioden               3.479e-01  7.871e-02   4.420 9.85e-06 ***
APPLIED_CREDIT_LIMIT_AMT                -2.460e-02  9.258e-02  -0.266 0.790500
GRANTED_CREDIT_LIMIT_AMT                 1.121e-01  9.240e-02   1.213 0.225227
GROSS_INCOME_AMT                         1.523e-01  3.537e+00   0.043 0.965659
STUDENT_LOAN_AMT                         9.323e-02  3.248e-02   2.870 0.004103 **
MORTGAGES_AMT                           -9.804e-02  5.125e-02  -1.913 0.055747 .
DebtRegisterNum                          2.282e-02  3.198e-02   0.713 0.475578
DebtRegisterIELA                        -3.879e-02  4.403e-02  -0.881 0.378268
HOMEOWNER_IND                            3.596e-02  5.680e-02   0.633 0.526654
HOUSING_COOPERATIVE_IND                  6.208e-02  3.982e-02   1.559 0.119004
NoOfChildren                            -1.192e-01  4.191e-02  -2.845 0.004435 **
FLI_AMT                                  5.245e-01  5.597e-01   0.937 0.348730
SFLI_AMT                                -4.304e-01  5.281e-01  -0.815 0.415091
SumAvailable                            -2.357e-01  9.552e-02  -2.467 0.013618 *
Applied_vs_Granted                       1.514e-02  5.058e-02   0.299 0.764726
SumPaidToCCL12                           3.557e-02  3.992e-02   0.891 0.372990
SumPaidToRepaymentLoanL12                1.276e-02  5.785e-02   0.221 0.825467
CountPaidToRepaymentLoanL12              5.459e-02  5.683e-02   0.961 0.336733
CountPaidToCCL12                        -4.598e-02  5.200e-02  -0.884 0.376564
CountRoundPaidToRepaymentLoanL12        -9.196e-02  5.028e-02  -1.829 0.067419 .
CountRoundPaidToCCL12                   -4.900e-02  4.786e-02  -1.024 0.305946
AIRLINE_12                              -1.659e-02  3.955e-02  -0.419 0.674922
ELECTRIC_APPLIANCE_12                    2.979e-02  4.125e-02   0.722 0.470245
FOOD_STORES_WAREHOUSE_12                -1.896e-02  4.346e-02  -0.436 0.662626
HOTEL_MOTEL_12                           1.309e-02  5.970e-02   0.219 0.826436
HARDWARE_12                             -1.011e-01  6.743e-02  -1.499 0.133847
INTERIOR_FURNISHINGS_12                  1.964e-02  4.758e-02   0.413 0.679711
OTHER_RETAIL_12                         -3.736e-04  2.810e-02  -0.013 0.989391
OTHER_SERVICES_12                        1.239e-02  2.994e-02   0.414 0.678951
OTHER_TRANSPORT_12                       2.766e-03  2.976e-02   0.093 0.925952
```

```
RECREATION_12                            2.442e-02  2.765e-02   0.883 0.377040
RESTAURANTS_BARS_12                      6.658e-02  3.118e-02   2.135 0.032749 *
SPORTING_TOY_STORES_12                  -2.938e-02  2.931e-02  -1.003 0.316099
TRAVEL_AGENCIES_12                       5.573e-02  2.622e-02   2.125 0.033563 *
VEHICLES_12                              3.145e-02  2.801e-02   1.123 0.261549
QUASI_CASH_12                            1.480e-02  2.875e-02   0.515 0.606596
AIRLINE_3                                5.701e-02  3.951e-02   1.443 0.149014
ELECTRIC_APPLIANCE_3                     1.551e-02  4.001e-02   0.388 0.698295
FOOD_STORES_WAREHOUSE_3                  4.061e-02  3.817e-02   1.064 0.287313
HOTEL_MOTEL_3                            4.934e-02  5.414e-02   0.911 0.362197
HARDWARE_3                               5.877e-02  4.768e-02   1.233 0.217752
INTERIOR_FURNISHINGS_3                  -1.903e-02  4.502e-02  -0.423 0.672590
AktiviPerioden                           1.571e+00  7.910e-01   1.986 0.046999 *
Missing_purchaseSeg                     -9.886e-01  1.750e-01  -5.648 1.62e-08 ***
Missing_application                      1.642e+00  2.656e-01   6.182 6.33e-10 ***
Missing_purchaseHist                     3.895e-01  1.380e-01   2.823 0.004757 **
Missing_debt                             2.177e-01  8.822e-02   2.467 0.013609 *
Missing_sumAvail                         1.225e+00  1.057e-01  11.586  < 2e-16 ***
ApplicationSalesChannel_Autentisert_web  7.771e-02  1.213e-01   0.641 0.521674
ApplicationSalesChannel_Mobilbank        5.352e-01  1.097e-01   4.880 1.06e-06 ***
ApplicationSalesChannel_Nettbank         6.060e-01  1.003e-01   6.043 1.51e-09 ***
ApplicationSalesChannel_Responsside      1.598e+01  5.354e+02   0.030 0.976186
EMPLOYMENT_TYPE_NAME_AT_HOME            -1.258e+00  1.292e+00  -0.974 0.330138
EMPLOYMENT_TYPE_NAME_DISABILITY_         1.321e-01  1.790e-01   0.738 0.460406
PENSIONER
EMPLOYMENT_TYPE_NAME_OTHER              -2.749e-01  3.788e-01  -0.726 0.467971
EMPLOYMENT_TYPE_NAME_RETIREE             1.001e-02  1.424e-01   0.070 0.943917
EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED       2.964e-01  2.188e-01   1.355 0.175484
EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY     1.430e+00  1.308e+00   1.093 0.274282
EMPLOYMENT_TYPE_NAME_STUDENT             3.066e-01  1.671e-01   1.835 0.066498 .
EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE       5.204e-01  1.806e-01   2.882 0.003955 **
EMPLOYMENT_TYPE_NAME_UNEMPLOYED          5.999e-01  6.575e-01   0.912 0.361535
EMPLOYMENT_DURATION_DESC_Between_1_      1.069e-01  1.196e-01   0.894 0.371101
and_3_years
EMPLOYMENT_DURATION_DESC_Less_than_     -9.837e-03  1.435e-01  -0.069 0.945360
1_year
HABITATION_TYPE_NAME_APARTMENT          -2.233e-01  2.137e-01  -1.045 0.296141
HABITATION_TYPE_NAME_OTHER              -2.638e-01  2.234e-01  -1.181 0.237644
HABITATION_TYPE_NAME_PARENTS            -2.937e-01  1.953e-01  -1.504 0.132502
HABITATION_TYPE_NAME_RENTER             -3.764e-01  1.540e-01  -2.444 0.014509 *
MARITAL_STATUS_NAME_COHABITING           1.695e-01  1.131e-01   1.499 0.133832
MARITAL_STATUS_NAME_DIVORCED             3.221e-01  1.887e-01   1.707 0.087821 .
```

```
MARITAL_STATUS_NAME_MARRIED              3.405e-01  1.510e-01   2.255 0.024118 *
MARITAL_STATUS_NAME_WIDOWED             -8.560e-02  2.096e-01  -0.408 0.682962
TAX_CLASS_CD_0                          -2.889e-01  6.028e-01  -0.479 0.631755
TAX_CLASS_CD_Unknown                    -9.054e-01  2.292e-01  -3.950 7.82e-05 ***
LastTaxYear2_TAX_CLASS_CD_0             -1.256e+01  2.289e+02  -0.055 0.956234
LastTaxYear2_TAX_CLASS_CD_1             -8.804e-02  2.141e-01  -0.411 0.680969
LastTaxYear2_TAX_CLASS_CD_1E            -2.371e-01  3.178e-01  -0.746 0.455558
LastTaxYear2_TAX_CLASS_CD_2              4.471e-02  1.252e+00   0.036 0.971525
LastTaxYear2_TAX_CLASS_CD_2F             3.310e-01  4.712e-01   0.703 0.482358
LastTaxYear3_TAX_CLASS_CD_0             -1.201e+01  3.784e+02  -0.032 0.974683
LastTaxYear3_TAX_CLASS_CD_1              2.201e-02  1.775e-01   0.124 0.901322
LastTaxYear3_TAX_CLASS_CD_1E            -1.504e-02  2.813e-01  -0.053 0.957352
LastTaxYear3_TAX_CLASS_CD_2             -2.919e-01  7.619e-01  -0.383 0.701625
LastTaxYear3_TAX_CLASS_CD_2F            -2.629e-01  3.845e-01  -0.684 0.494212
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8370.6  on 7070  degrees of freedom
Residual deviance: 7473.4  on 6980  degrees of freedom
AIC: 7655.4

Number of Fisher Scoring iterations: 12
```

B.2 BIC-Reduced Logistic Models

One Month Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ ProductId +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    CountPaidToRepaymentLoanL12 + ELECTRIC_APPLIANCE_12 +
    RESTAURANTS_BARS_12 + TRAVEL_AGENCIES_12 +
    VEHICLES_12 + AktiviPerioden + Missing_purchaseSeg + Missing_application +
    Missing_sumAvail + ApplicationSalesChannel_Kredittbanken +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    TAX_CLASS_CD_Unknown, family = binomial, data = oneMo_train.stand)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max  
-1.8738  -1.0217  -0.6442   1.1523   2.5874  

Coefficients:
                                       Estimate Std. Error z value Pr(>|z|)    
(Intercept)                           -0.328382   0.042806  -7.671 1.70e-14 ***
ProductId                              0.033482   0.001964  17.045  < 2e-16 ***
MndUtenKortbrukiPerioden              -0.095664   0.002465 -38.802  < 2e-16 ***
MndFraFørsteTilSisteBruk              -0.016161   0.002520  -6.414 1.42e-10 ***
CountPaidToRepaymentLoanL12           -0.055812   0.015613  -3.575 0.000351 ***
ELECTRIC_APPLIANCE_12                  0.077437   0.015299   5.062 4.16e-07 ***
RESTAURANTS_BARS_12                   -0.137324   0.019213  -7.147 8.85e-13 ***
TRAVEL_AGENCIES_12                    -0.067830   0.016563  -4.095 4.21e-05 ***
VEHICLES_12                            0.051524   0.015252   3.378 0.000730 ***
AktiviPerioden                         0.935135   0.058692  15.933  < 2e-16 ***
Missing_purchaseSeg                   -0.552491   0.081244  -6.800 1.04e-11 ***
Missing_application                    0.537391   0.091566   5.869 4.39e-09 ***
Missing_sumAvail                       0.924734   0.065749  14.065  < 2e-16 ***
ApplicationSalesChannel_Kredittbanken -2.937734   0.738662  -3.977 6.98e-05 ***
ApplicationSalesChannel_Mobilbank      0.200721   0.047693   4.209 2.57e-05 ***
ApplicationSalesChannel_Nettbank       0.303978   0.047265   6.431 1.26e-10 ***
HABITATION_TYPE_NAME_PARENTS          -0.216195   0.062398  -3.465 0.000531 ***
HABITATION_TYPE_NAME_RENTER           -0.232375   0.045329  -5.126 2.95e-07 ***
TAX_CLASS_CD_Unknown                  -0.309417   0.083260  -3.716 0.000202 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 27833  on 20600  degrees of freedom
Residual deviance: 25627  on 20582  degrees of freedom
AIC: 25665

Number of Fisher Scoring iterations: 4
```

Three Months Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ ProductId + 
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    CountDistinctPaidToRepaymentLoanL12 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_sumAvail +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    HABITATION_TYPE_NAME_RENTER + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2F,
    family = binomial, data = threeMos_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7648  -0.9528  -0.6626   1.1821   2.3459

Coefficients:
                                     Estimate Std. Error z value Pr(>|z|)
(Intercept)                         -0.931161   0.080196 -11.611  < 2e-16 ***
ProductId                            0.041939   0.002288  18.332  < 2e-16 ***
MndUtenKortbrukiPerioden            -0.096625   0.002972 -32.511  < 2e-16 ***
MndFraFørsteTilSisteBruk            -0.026899   0.003203  -8.399  < 2e-16 ***
CountDistinctPaidToRepaymentLoanL12 -0.070014   0.019459  -3.598 0.000321 ***
AktiviPerioden                       1.070996   0.070581  15.174  < 2e-16 ***
Missing_purchaseSeg                 -0.653350   0.097150  -6.725 1.75e-11 ***
Missing_application                  0.460440   0.072250   6.373 1.86e-10 ***
Missing_sumAvail                     0.902312   0.075089  12.017  < 2e-16 ***
ApplicationSalesChannel_Mobilbank    0.202315   0.058351   3.467 0.000526 ***
ApplicationSalesChannel_Nettbank     0.386424   0.056812   6.802 1.03e-11 ***
HABITATION_TYPE_NAME_RENTER         -0.239493   0.054578  -4.388 1.14e-05 ***
LastTaxYear3_TAX_CLASS_CD_1          0.314073   0.062697   5.009 5.46e-07 ***
LastTaxYear3_TAX_CLASS_CD_1E         0.302232   0.071652   4.218 2.46e-05 ***
LastTaxYear3_TAX_CLASS_CD_2F         0.698322   0.177502   3.934 8.35e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 20282  on 15552  degrees of freedom
Residual deviance: 18703  on 15538  degrees of freedom
AIC: 18733

Number of Fisher Scoring iterations: 4
```

Six Months Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ ProductId +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk + FLI_AMT +
    SFLI_AMT + SumAvailable + CountPaidToCCL12 +
    CountRoundPaidToRepaymentLoanL12 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_sumAvail +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    HABITATION_TYPE_NAME_RENTER + TAX_CLASS_CD_Unknown, family = binomial,
    data = sixMos_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8332  -0.8528  -0.6386   1.1521   2.6377

Coefficients:
                                   Estimate Std. Error z value Pr(>|z|)
(Intercept)                       -1.074176   0.065616 -16.371  < 2e-16 ***
ProductId                          0.051171   0.002854  17.930  < 2e-16 ***
MndUtenKortbrukiPerioden          -0.093470   0.003751 -24.918  < 2e-16 ***
MndFraFørsteTilSisteBruk          -0.044191   0.004487  -9.850  < 2e-16 ***
FLI_AMT                            1.012790   0.296969   3.410 0.000649 ***
SFLI_AMT                          -0.965572   0.291578  -3.312 0.000928 ***
SumAvailable                      -0.160942   0.071157  -2.262 0.023711 *
CountPaidToCCL12                  -0.083715   0.026047  -3.214 0.001309 **
CountRoundPaidToRepaymentLoanL12  -0.114942   0.035112  -3.274 0.001062 **
AktiviPerioden                     1.017772   0.088727  11.471  < 2e-16 ***
Missing_purchaseSeg               -1.059101   0.126964  -8.342  < 2e-16 ***
Missing_application                0.558695   0.131255   4.257 2.08e-05 ***
Missing_sumAvail                   1.050138   0.085348  12.304  < 2e-16 ***
ApplicationSalesChannel_Mobilbank  0.307946   0.072269   4.261 2.03e-05 ***
ApplicationSalesChannel_Nettbank   0.441822   0.070639   6.255 3.98e-10 ***
HABITATION_TYPE_NAME_RENTER       -0.241074   0.069461  -3.471 0.000519 ***
TAX_CLASS_CD_Unknown              -0.504223   0.123455  -4.084 4.42e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14205  on 11631  degrees of freedom
Residual deviance: 13078  on 11615  degrees of freedom
```

```
AIC: 13112

Number of Fisher Scoring iterations: 5
```

Twelve Months Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ ProductId +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + STUDENT_LOAN_AMT +
    RESTAURANTS_BARS_12 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_sumAvail + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + HABITATION_TYPE_NAME_RENTER +
    TAX_CLASS_CD_Unknown, family = binomial,
    data = twelveMos_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.5085  -0.8144  -0.5969   1.0738   2.8695

Coefficients:
                                   Estimate Std. Error z value Pr(>|z|)
(Intercept)                       -3.813302   0.460020  -8.289  < 2e-16 ***
ProductId                          0.038854   0.004270   9.098  < 2e-16 ***
MndUtenKortbrukiPerioden          -0.117014   0.005756 -20.328  < 2e-16 ***
MndFraFørsteTilSisteBruk          -0.071509   0.007573  -9.443  < 2e-16 ***
MndUtenKortbrukFørPerioden         0.375998   0.079058   4.756 1.98e-06 ***
STUDENT_LOAN_AMT                   0.087110   0.028026   3.108  0.00188 **
RESTAURANTS_BARS_12                0.085582   0.027691   3.091  0.00200 **
AktiviPerioden                     0.919066   0.130306   7.053 1.75e-12 ***
Missing_purchaseSeg               -1.023434   0.171819  -5.956 2.58e-09 ***
Missing_application                1.600242   0.190185   8.414  < 2e-16 ***
Missing_purchaseHist               0.510372   0.106488   4.793 1.64e-06 ***
Missing_sumAvail                   1.240472   0.102757  12.072  < 2e-16 ***
ApplicationSalesChannel_Mobilbank  0.477960   0.098934   4.831 1.36e-06 ***
ApplicationSalesChannel_Nettbank   0.581828   0.093229   6.241 4.35e-10 ***
HABITATION_TYPE_NAME_RENTER       -0.278761   0.090187  -3.091  0.00200 **
TAX_CLASS_CD_Unknown              -0.961818   0.177992  -5.404 6.53e-08 ***
---
```

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8370.6  on 7070  degrees of freedom
Residual deviance: 7601.9  on 7055  degrees of freedom
AIC: 7633.9

Number of Fisher Scoring iterations: 4
```

B.3 AIC-Reduced Logistic Models

One Month Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ Alder + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + APPLIED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT +
    STUDENT_LOAN_AMT + SumPaidToCCL12 + CountPaidToRepaymentLoanL12 +
    CountDistinctPaidToRepaymentLoanL12 + AIRLINE_12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HOTEL_MOTEL_12 + INTERIOR_FURNISHINGS_12 +
    OTHER_TRANSPORT_12 + RESTAURANTS_BARS_12 + TRAVEL_AGENCIES_12 +
    VEHICLES_12 + QUASI_CASH_12 + HOTEL_MOTEL_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_sumAvail +
    ApplicationSalesChannel_Kredittbanken + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER + EMPLOYMENT_TYPE_NAME_OTHER +
    EMPLOYMENT_TYPE_NAME_RETIREE + EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE +
    EMPLOYMENT_TYPE_NAME_UNEMPLOYED + `EMPLOYMENT_DURATION_DESC_Not set` +
    HABITATION_TYPE_NAME_APARTMENT + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + TAX_CLASS_CD_Unknown +
    LastTaxYear2_TAX_CLASS_CD_0 + LastTaxYear2_TAX_CLASS_CD_2F +
    LastTaxYear3_TAX_CLASS_CD_2F,
    family = binomial, data = oneMo_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0298  -1.0124  -0.6342   1.1442   2.8914
```

```
Coefficients:
                                         Estimate Std. Error z value Pr(>|z|)
(Intercept)                             -0.643676   0.144590  -4.452 8.52e-06 ***
Alder                                    0.002275   0.001298   1.752 0.079775 .
ProductId                                0.033517   0.002029  16.521  < 2e-16 ***
TransaksjonerMensPassiv                  0.018717   0.007500   2.495 0.012579 *
AntallPassivPerioder                     0.249558   0.119817   2.083 0.037267 *
MndUtenKortbrukiPerioden                -0.098869   0.002569 -38.488  < 2e-16 ***
MndFraFørsteTilSisteBruk                -0.018672   0.002664  -7.008 2.42e-12 ***
APPLIED_CREDIT_LIMIT_AMT                 0.042843   0.016955   2.527 0.011510 *
GROSS_INCOME_AMT                         1.457986   2.825269   0.516 0.605818
STUDENT_LOAN_AMT                         0.031700   0.016287   1.946 0.051619 .
SumPaidToCCL12                          -0.033037   0.015815  -2.089 0.036714 *
CountPaidToRepaymentLoanL12            -13.456352 115.127981  -0.117 0.906954
CountDistinctPaidToRepaymentLoanL12     13.388415 115.005962   0.116 0.907324
AIRLINE_12                              -0.051040   0.016499  -3.094 0.001978 **
ELECTRIC_APPLIANCE_12                    0.064854   0.015604   4.156 3.24e-05 ***
FOOD_STORES_WAREHOUSE_12                 0.054000   0.017537   3.079 0.002076 **
HOTEL_MOTEL_12                           0.073784   0.027363   2.696 0.007007 **
INTERIOR_FURNISHINGS_12                  0.038066   0.015661   2.431 0.015075 *
OTHER_TRANSPORT_12                      -0.047288   0.019447  -2.432 0.015029 *
RESTAURANTS_BARS_12                     -0.152637   0.021741  -7.021 2.21e-12 ***
TRAVEL_AGENCIES_12                      -0.064443   0.016573  -3.889 0.000101 ***
VEHICLES_12                              0.045607   0.015392   2.963 0.003047 **
QUASI_CASH_12                           -0.032552   0.016496  -1.973 0.048461 *
HOTEL_MOTEL_3                           -0.087751   0.026645  -3.293 0.000990 ***
AktiviPerioden                           0.688626   0.136870   5.031 4.87e-07 ***
Missing_purchaseSeg                     -0.545571   0.081793  -6.670 2.56e-11 ***
Missing_application                      0.495368   0.097019   5.106 3.29e-07 ***
Missing_sumAvail                         0.923957   0.066280  13.940  < 2e-16 ***
ApplicationSalesChannel_Kredittbanken   -2.880769   0.740546  -3.890 0.000100 ***
ApplicationSalesChannel_Mobilbank        0.186465   0.049053   3.801 0.000144 ***
ApplicationSalesChannel_Nettbank         0.297267   0.048048   6.187 6.14e-10 ***
ApplicationSalesChannel_Responsside     14.608146 324.743717   0.045 0.964120
EMPLOYMENT_TYPE_NAME_DISABILITY_        -0.236025   0.103988  -2.270 0.023224 *
PENSIONER
EMPLOYMENT_TYPE_NAME_OTHER              -0.401415   0.167033  -2.403 0.016252 *
EMPLOYMENT_TYPE_NAME_RETIREE            -0.316626   0.096512  -3.281 0.001035 **
EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE       0.164424   0.087903   1.871 0.061412 .
EMPLOYMENT_TYPE_NAME_UNEMPLOYED         -0.956326   0.418732  -2.284 0.022380 *
EMPLOYMENT_DURATION_DESC_Not_set         0.135832   0.067944   1.999 0.045589 *
HABITATION_TYPE_NAME_APARTMENT          -0.164521   0.083541  -1.969 0.048913 *
```

```
HABITATION_TYPE_NAME_OTHER            -0.189757   0.097013  -1.956 0.050466 .
HABITATION_TYPE_NAME_PARENTS          -0.214892   0.074379  -2.889 0.003863 **
HABITATION_TYPE_NAME_RENTER           -0.266087   0.051821  -5.135 2.83e-07 ***
MARITAL_STATUS_NAME_COHABITING         0.085858   0.046597   1.843 0.065392 .
TAX_CLASS_CD_Unknown                  -0.290187   0.085961  -3.376 0.000736 ***
LastTaxYear2_TAX_CLASS_CD_0           -0.709482   0.448860  -1.581 0.113963
LastTaxYear2_TAX_CLASS_CD_2F           0.461706   0.243735   1.894 0.058186 .
LastTaxYear3_TAX_CLASS_CD_2F           0.376683   0.167460   2.249 0.024488 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 27833  on 20600  degrees of freedom
Residual deviance: 25493  on 20554  degrees of freedom
AIC: 25587

Number of Fisher Scoring iterations: 11
```

Three Months Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ Alder + Kjønn +
    ProductId + TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + GRANTED_CREDIT_LIMIT_AMT +
    GROSS_INCOME_AMT + STUDENT_LOAN_AMT + SumAvailable +
    SumPaidToCCL12 + CountDistinctPaidToRepaymentLoanL12 +
    CountRoundPaidToRepaymentLoanL12 + AIRLINE_12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + OTHER_RETAIL_12 + RECREATION_12 +
    RESTAURANTS_BARS_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    FOOD_STORES_WAREHOUSE_3 + HOTEL_MOTEL_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_debt +
    Missing_sumAvail + `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Kredittbanken +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER +
    EMPLOYMENT_TYPE_NAME_RETIREE + EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Not set` + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_WIDOWED +
    TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_2 + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2F,
    family = binomial, data = threeMos_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8169  -0.9439  -0.6524   1.1681   2.4275

Coefficients:
                                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                              -1.060485   0.256646  -4.132 3.59e-05 ***
Alder                                     0.004484   0.001518   2.954 0.003140 **
Kjønn                                     0.065303   0.036501   1.789 0.073602 .
ProductId                                 0.043433   0.002446  17.756  < 2e-16 ***
TransaksjonerMensPassiv                   0.025601   0.008448   3.031 0.002441 **
AntallPassivPerioder                      0.276579   0.156491   1.767 0.077164 .
MndUtenKortbrukiPerioden                 -0.100823   0.003104 -32.477  < 2e-16 ***
MndFraFørsteTilSisteBruk                 -0.028295   0.003492  -8.102 5.41e-16 ***
MndUtenKortbrukFørPerioden               -0.063284   0.027763  -2.279 0.022642 *
GRANTED_CREDIT_LIMIT_AMT                  0.036949   0.020753   1.780 0.075004 .
GROSS_INCOME_AMT                          3.126856   3.509776   0.891 0.372983
STUDENT_LOAN_AMT                          0.032414   0.019241   1.685 0.092063 .
SumAvailable                             -0.037428   0.027425  -1.365 0.172328
SumPaidToCCL12                           -0.029530   0.018814  -1.570 0.116522
CountDistinctPaidToRepaymentLoanL12      -0.048619   0.022032  -2.207 0.027334 *
CountRoundPaidToRepaymentLoanL12         -0.030705   0.021740  -1.412 0.157841
AIRLINE_12                               -0.033301   0.018534  -1.797 0.072370 .
ELECTRIC_APPLIANCE_12                     0.038668   0.017419   2.220 0.026431 *
FOOD_STORES_WAREHOUSE_12                 -0.056318   0.027782  -2.027 0.042648 *
OTHER_RETAIL_12                           0.032982   0.018655   1.768 0.077068 .
RECREATION_12                             0.044233   0.019008   2.327 0.019961 *
RESTAURANTS_BARS_12                      -0.055033   0.023613  -2.331 0.019775 *
TRAVEL_AGENCIES_12                       -0.035708   0.018914  -1.888 0.059036 .
VEHICLES_12                               0.042410   0.017590   2.411 0.015908 *
FOOD_STORES_WAREHOUSE_3                   0.070320   0.023411   3.004 0.002667 **
HOTEL_MOTEL_3                             0.056398   0.017785   3.171 0.001519 **
AktiviPerioden                            0.752574   0.175927   4.278 1.89e-05 ***
Missing_purchaseSeg                      -0.606229   0.098245  -6.171 6.80e-10 ***
Missing_application                       0.654684   0.120262   5.444 5.22e-08 ***
Missing_debt                              0.159305   0.051329   3.104 0.001912 **
```

```
Missing_sumAvail                          0.915657   0.076654  11.945  < 2e-16 ***
ApplicationSalesChannel_Autentisert_web  -0.132594   0.078057  -1.699 0.089378 .
ApplicationSalesChannel_Kredittbanken    -1.621884   1.172523  -1.383 0.166590
ApplicationSalesChannel_Mobilbank         0.188112   0.061032   3.082 0.002055 **
ApplicationSalesChannel_Nettbank          0.372059   0.058696   6.339 2.32e-10 ***
ApplicationSalesChannel_Responsside      12.771746 119.468129   0.107 0.914864
EMPLOYMENT_TYPE_NAME_DISABILITY_         -0.437763   0.123506  -3.544 0.000393 ***
PENSIONER
EMPLOYMENT_TYPE_NAME_RETIREE             -0.438856   0.112030  -3.917 8.95e-05 ***
EMPLOYMENT_TYPE_NAME_UNEMPLOYED          -0.662715   0.433442  -1.529 0.126274
`EMPLOYMENT_DURATION_DESC_Not set`        0.204036   0.077637   2.628 0.008587 **
HABITATION_TYPE_NAME_RENTER              -0.257861   0.056815  -4.539 5.66e-06 ***
MARITAL_STATUS_NAME_COHABITING            0.121696   0.059069   2.060 0.039375 *
MARITAL_STATUS_NAME_WIDOWED              -0.182017   0.124137  -1.466 0.142577
TAX_CLASS_CD_0                           -0.882844   0.503709  -1.753 0.079656 .
TAX_CLASS_CD_Unknown                     -0.293733   0.114929  -2.556 0.010595 *
LastTaxYear2_TAX_CLASS_CD_0              -0.826435   0.571894  -1.445 0.148434
LastTaxYear2_TAX_CLASS_CD_2               0.711913   0.489702   1.454 0.146011
LastTaxYear3_TAX_CLASS_CD_1               0.224174   0.083150   2.696 0.007017 **
LastTaxYear3_TAX_CLASS_CD_1E              0.253619   0.096086   2.639 0.008303 **
LastTaxYear3_TAX_CLASS_CD_2F              0.643217   0.187194   3.436 0.000590 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 20282  on 15552  degrees of freedom
Residual deviance: 18554  on 15503  degrees of freedom
AIC: 18654

Number of Fisher Scoring iterations: 9
```

Six Months Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ~ Alder + ProductId +
    TransaksjonerMensPassiv + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + APPLIED_CREDIT_LIMIT_AMT +
    GROSS_INCOME_AMT + STUDENT_LOAN_AMT + MORTGAGES_AMT + NoOfChildren +
    FLI_AMT + SFLI_AMT + SumAvailable + CountPaidToCCL12 +
    CountRoundPaidToRepaymentLoanL12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HARDWARE_12 + VEHICLES_12 +
    FOOD_STORES_WAREHOUSE_3 + HOTEL_MOTEL_3 + HARDWARE_3 +
    AktiviPerioden + Missing_purchaseSeg + Missing_application +
    Missing_purchaseHist + Missing_debt + Missing_sumAvail +
    `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    ApplicationSalesChannel_Responsside + EMPLOYMENT_TYPE_NAME_RETIREE +
    EMPLOYMENT_TYPE_NAME_STUDENT + HABITATION_TYPE_NAME_PARENTS +
    HABITATION_TYPE_NAME_RENTER + MARITAL_STATUS_NAME_WIDOWED +
    TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_2F + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2F,
    family = binomial, data = sixMos_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0722  -0.8490  -0.6262   1.1326   2.6461

Coefficients:
                                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                              -1.578772   0.161477  -9.777  < 2e-16 ***
Alder                                     0.005650   0.001836   3.078  0.00209 **
ProductId                                 0.053622   0.003121  17.183  < 2e-16 ***
TransaksjonerMensPassiv                   0.032023   0.010576   3.028  0.00246 **
MndUtenKortbrukiPerioden                 -0.097710   0.003892 -25.105  < 2e-16 ***
MndFraFørsteTilSisteBruk                 -0.044289   0.004853  -9.126  < 2e-16 ***
APPLIED_CREDIT_LIMIT_AMT                  0.038435   0.024005   1.601  0.10935
GROSS_INCOME_AMT                          5.165930   4.376575   1.180  0.23786
STUDENT_LOAN_AMT                          0.057274   0.023151   2.474  0.01336 *
MORTGAGES_AMT                            -0.178094   0.095897  -1.857  0.06329 .
NoOfChildren                             -0.059203   0.029355  -2.017  0.04372 *
FLI_AMT                                   0.921039   0.349880   2.632  0.00848 **
SFLI_AMT                                 -0.856644   0.339196  -2.526  0.01155 *
SumAvailable                             -0.243705   0.094153  -2.588  0.00964 **
CountPaidToCCL12                         -0.047769   0.030108  -1.587  0.11261
CountRoundPaidToRepaymentLoanL12         -0.104598   0.034557  -3.027  0.00247 **
ELECTRIC_APPLIANCE_12                     0.044685   0.020453   2.185  0.02890 *
FOOD_STORES_WAREHOUSE_12                 -0.056608   0.034469  -1.642  0.10053
HARDWARE_12                              -0.048602   0.034478  -1.410  0.15864
VEHICLES_12                               0.031079   0.020960   1.483  0.13814
FOOD_STORES_WAREHOUSE_3                   0.051834   0.029044   1.785  0.07432 .
```

```
HOTEL_MOTEL_3                            0.054278   0.020220   2.684  0.00727 **
HARDWARE_3                               0.055762   0.030143   1.850  0.06433 .
AktiviPerioden                           0.956526   0.092434  10.348  < 2e-16 ***
Missing_purchaseSeg                     -1.008413   0.127627  -7.901 2.76e-15 ***
Missing_application                      0.513700   0.161417   3.182  0.00146 **
Missing_purchaseHist                     0.133626   0.080870   1.652  0.09846 .
Missing_debt                             0.199647   0.063970   3.121  0.00180 **
Missing_sumAvail                         1.061934   0.086457  12.283  < 2e-16 ***
ApplicationSalesChannel_Autentisert_web -0.207036   0.092028  -2.250  0.02447 *
ApplicationSalesChannel_Mobilbank        0.333297   0.077030   4.327 1.51e-05 ***
ApplicationSalesChannel_Nettbank         0.410816   0.073877   5.561 2.68e-08 ***
ApplicationSalesChannel_Responsside     12.684270 119.468131   0.106  0.91545
EMPLOYMENT_TYPE_NAME_RETIREE            -0.201002   0.103593  -1.940  0.05234 .
EMPLOYMENT_TYPE_NAME_STUDENT             0.330019   0.100717   3.277  0.00105 **
HABITATION_TYPE_NAME_PARENTS            -0.250499   0.114370  -2.190  0.02851 *
HABITATION_TYPE_NAME_RENTER             -0.365549   0.084182  -4.342 1.41e-05 ***
MARITAL_STATUS_NAME_WIDOWED             -0.231914   0.152268  -1.523  0.12774
TAX_CLASS_CD_Unknown                    -0.398041   0.146916  -2.709  0.00674 **
LastTaxYear2_TAX_CLASS_CD_0             -1.408737   1.065091  -1.323  0.18595
LastTaxYear2_TAX_CLASS_CD_2F             0.717781   0.302572   2.372  0.01768 *
LastTaxYear3_TAX_CLASS_CD_1              0.183632   0.103732   1.770  0.07669 .
LastTaxYear3_TAX_CLASS_CD_1E             0.207843   0.123565   1.682  0.09256 .
LastTaxYear3_TAX_CLASS_CD_2F             0.358123   0.236121   1.517  0.12934
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14205  on 11631  degrees of freedom
Residual deviance: 12968  on 11588  degrees of freedom
AIC: 13056

Number of Fisher Scoring iterations: 9
```

Twelve Months Ahead

```
Call:
glm(formula = as.factor(AktivEtterPassiv) ˜ Alder + ProductId +
    TransaksjonerMensPassiv + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + MndUtenKortbrukFørPerioden +
    GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    MORTGAGES_AMT + NoOfChildren + SumAvailable +
    CountPaidToRepaymentLoanL12 + CountRoundPaidToRepaymentLoanL12 +
    CountRoundPaidToCCL12 + ELECTRIC_APPLIANCE_12 + RESTAURANTS_BARS_12 +
    TRAVEL_AGENCIES_12 + AIRLINE_3 + HOTEL_MOTEL_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_debt + Missing_sumAvail + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED + EMPLOYMENT_TYPE_NAME_STUDENT +
    EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_DIVORCED +
    MARITAL_STATUS_NAME_MARRIED + TAX_CLASS_CD_Unknown +
    LastTaxYear2_TAX_CLASS_CD_0 + LastTaxYear2_TAX_CLASS_CD_1E,
    family = binomial, data = twelveMos_train.stand)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.4909  -0.8119  -0.5792   1.0316   2.8944

Coefficients:
                                      Estimate Std. Error z value Pr(>|z|)
(Intercept)                          -3.939538   0.488662  -8.062 7.51e-16 ***
Alder                                 0.003258   0.002150   1.516 0.129618
ProductId                             0.041622   0.004473   9.304  < 2e-16 ***
TransaksjonerMensPassiv               0.048634   0.014371   3.384 0.000714 ***
MndUtenKortbrukiPerioden             -0.123659   0.005992 -20.638  < 2e-16 ***
MndFraFørsteTilSisteBruk             -0.073224   0.007722  -9.483  < 2e-16 ***
MndUtenKortbrukFørPerioden            0.356055   0.078481   4.537 5.71e-06 ***
GRANTED_CREDIT_LIMIT_AMT              0.093554   0.033849   2.764 0.005712 **
GROSS_INCOME_AMT                      0.498816   8.416553   0.059 0.952740
STUDENT_LOAN_AMT                      0.092155   0.030298   3.042 0.002353 **
MORTGAGES_AMT                        -0.050686   0.039323  -1.289 0.197413
NoOfChildren                         -0.114638   0.037678  -3.043 0.002346 **
SumAvailable                         -0.119885   0.048568  -2.468 0.013573 *
CountPaidToRepaymentLoanL12           0.062232   0.036718   1.695 0.090099 .
CountRoundPaidToRepaymentLoanL12     -0.089410   0.049173  -1.818 0.069025 .
CountRoundPaidToCCL12                -0.061370   0.043466  -1.412 0.157972
ELECTRIC_APPLIANCE_12                 0.040500   0.026474   1.530 0.126070
RESTAURANTS_BARS_12                   0.074161   0.028445   2.607 0.009129 **
TRAVEL_AGENCIES_12                    0.055719   0.026115   2.134 0.032879 *
AIRLINE_3                             0.046686   0.028340   1.647 0.099487 .
```

```
HOTEL_MOTEL_3                         0.060163   0.026489   2.271 0.023130 *
AktiviPerioden                        0.857955   0.132616   6.469 9.84e-11 ***
Missing_purchaseSeg                  -1.003063   0.173800  -5.771 7.86e-09 ***
Missing_application                   1.603937   0.222330   7.214 5.42e-13 ***
Missing_purchaseHist                  0.399962   0.125576   3.185 0.001447 **
Missing_debt                          0.216962   0.085942   2.525 0.011586 *
Missing_sumAvail                      1.229772   0.104318  11.789  < 2e-16 ***
ApplicationSalesChannel_Mobilbank     0.529393   0.105932   4.997 5.81e-07 ***
ApplicationSalesChannel_Nettbank      0.597462   0.096606   6.184 6.23e-10 ***
ApplicationSalesChannel_Responsside  15.974622 535.411221   0.030 0.976198
EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED    0.330799   0.212946   1.553 0.120318
EMPLOYMENT_TYPE_NAME_STUDENT          0.263377   0.138084   1.907 0.056473 .
EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE    0.527986   0.176814   2.986 0.002826 **
HABITATION_TYPE_NAME_OTHER           -0.321286   0.203601  -1.578 0.114562
HABITATION_TYPE_NAME_PARENTS         -0.344628   0.161299  -2.137 0.032633 *
HABITATION_TYPE_NAME_RENTER          -0.444230   0.115165  -3.857 0.000115 ***
MARITAL_STATUS_NAME_COHABITING        0.190046   0.108830   1.746 0.080765 .
MARITAL_STATUS_NAME_DIVORCED          0.334454   0.183054   1.827 0.067687 .
MARITAL_STATUS_NAME_MARRIED           0.374733   0.138134   2.713 0.006671 **
TAX_CLASS_CD_Unknown                 -0.906375   0.191052  -4.744 2.09e-06 ***
LastTaxYear2_TAX_CLASS_CD_0         -12.482261 229.838510  -0.054 0.956689
LastTaxYear2_TAX_CLASS_CD_1E         -0.216528   0.135968  -1.592 0.111273
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8370.6  on 7070  degrees of freedom
Residual deviance: 7498.4  on 7029  degrees of freedom
AIC: 7582.4

Number of Fisher Scoring iterations: 12
```

B.4 ANOVA Analysis of BIC-Reduced and Full Logistic Models

One Month Ahead

```
Analysis of Deviance Table

Model 1: as.factor(AktivEtterPassiv) ~ ProductId + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + CountPaidToRepaymentLoanL12 +
    ELECTRIC_APPLIANCE_12 + RESTAURANTS_BARS_12 + TRAVEL_AGENCIES_12 +
    VEHICLES_12 + AktiviPerioden + Missing_purchaseSeg + Missing_application +
    Missing_sumAvail + ApplicationSalesChannel_Kredittbanken +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    TAX_CLASS_CD_Unknown
Model 2: as.factor(AktivEtterPassiv) ~ Alder + Kjønn + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + APPLIED_CREDIT_LIMIT_AMT +
    GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    MORTGAGES_AMT + DebtRegisterNum + DebtRegisterIELA + HOMEOWNER_IND +
    HOUSING_COOPERATIVE_IND + NoOfChildren + FLI_AMT + SFLI_AMT +
    SumAvailable + Applied_vs_Granted + SumPaidToCCL12 +
    SumPaidToRepaymentLoanL12 + CountPaidToRepaymentLoanL12 +
    CountPaidToCCL12 + CountDistinctPaidToRepaymentLoanL12 +
    CountDistinctPaidToCCL12 + CountRoundPaidToRepaymentLoanL12 +
    CountRoundPaidToCCL12 + AIRLINE_12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HOTEL_MOTEL_12 + HARDWARE_12 +
    INTERIOR_FURNISHINGS_12 + OTHER_RETAIL_12 + OTHER_SERVICES_12 +
    OTHER_TRANSPORT_12 + RECREATION_12 + RESTAURANTS_BARS_12 +
    SPORTING_TOY_STORES_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    QUASI_CASH_12 + AIRLINE_3 + ELECTRIC_APPLIANCE_3 + FOOD_STORES_WAREHOUSE_3 +
    HOTEL_MOTEL_3 + HARDWARE_3 + INTERIOR_FURNISHINGS_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_debt + Missing_sumAvail + `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Kredittbanken + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_AT_HOME + EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER +
    EMPLOYMENT_TYPE_NAME_OTHER + EMPLOYMENT_TYPE_NAME_RETIREE +
    EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED + EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY +
    EMPLOYMENT_TYPE_NAME_STUDENT + EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE +
    EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Between 1 and 3 years` +
    `EMPLOYMENT_DURATION_DESC_Less than 1 year` +
    `EMPLOYMENT_DURATION_DESC_Not set` + HABITATION_TYPE_NAME_APARTMENT +
    HABITATION_TYPE_NAME_OTHER + HABITATION_TYPE_NAME_PARENTS +
    HABITATION_TYPE_NAME_RENTER + MARITAL_STATUS_NAME_COHABITING +
    MARITAL_STATUS_NAME_DIVORCED + MARITAL_STATUS_NAME_MARRIED +
    MARITAL_STATUS_NAME_WIDOWED + TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown +
    LastTaxYear2_TAX_CLASS_CD_0 + LastTaxYear2_TAX_CLASS_CD_1E +
    LastTaxYear2_TAX_CLASS_CD_2 + LastTaxYear2_TAX_CLASS_CD_2F +
    LastTaxYear2_TAX_CLASS_CD_Unknown + LastTaxYear3_TAX_CLASS_CD_0 +
    LastTaxYear3_TAX_CLASS_CD_1 + LastTaxYear3_TAX_CLASS_CD_1E +
    LastTaxYear3_TAX_CLASS_CD_2 + LastTaxYear3_TAX_CLASS_CD_2F
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1     20582      25626
2     20506      25470 76   156.87 1.457e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Three Months Ahead

```
Analysis of Deviance Table

Model 1: as.factor(AktivEtterPassiv) ~ ProductId + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + CountDistinctPaidToRepaymentLoanL12 +
    AktiviPerioden + Missing_purchaseSeg + Missing_application +
    Missing_sumAvail + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + HABITATION_TYPE_NAME_RENTER +
    LastTaxYear3_TAX_CLASS_CD_1 + LastTaxYear3_TAX_CLASS_CD_1E +
    LastTaxYear3_TAX_CLASS_CD_2F
Model 2: as.factor(AktivEtterPassiv) ~ Alder + Kjønn + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + MndUtenKortbrukFørPerioden +
    APPLIED_CREDIT_LIMIT_AMT + GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT +
    STUDENT_LOAN_AMT + MORTGAGES_AMT + DebtRegisterNum + DebtRegisterIELA +
    HOMEOWNER_IND + HOUSING_COOPERATIVE_IND + NoOfChildren + FLI_AMT + SFLI_AMT +
    SumAvailable + Applied_vs_Granted + SumPaidToCCL12 +
    SumPaidToRepaymentLoanL12 + CountPaidToRepaymentLoanL12 + CountPaidToCCL12 +
    CountDistinctPaidToRepaymentLoanL12 + CountDistinctPaidToCCL12 +
    CountRoundPaidToRepaymentLoanL12 + CountRoundPaidToCCL12 + AIRLINE_12 +
    ELECTRIC_APPLIANCE_12 + FOOD_STORES_WAREHOUSE_12 + HOTEL_MOTEL_12 +
    HARDWARE_12 + INTERIOR_FURNISHINGS_12 + OTHER_RETAIL_12 +
    OTHER_SERVICES_12 + OTHER_TRANSPORT_12 + RECREATION_12 + RESTAURANTS_BARS_12 +
    SPORTING_TOY_STORES_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    QUASI_CASH_12 + AIRLINE_3 + ELECTRIC_APPLIANCE_3 + FOOD_STORES_WAREHOUSE_3 +
    HOTEL_MOTEL_3 + HARDWARE_3 + INTERIOR_FURNISHINGS_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_debt + Missing_sumAvail + `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Kredittbanken + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_AT_HOME + EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER +
    EMPLOYMENT_TYPE_NAME_OTHER + EMPLOYMENT_TYPE_NAME_RETIREE +
    EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED + EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY +
    EMPLOYMENT_TYPE_NAME_STUDENT + EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE +
    EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Between 1 and 3 years` +
    `EMPLOYMENT_DURATION_DESC_Less than 1 year` +
    `EMPLOYMENT_DURATION_DESC_Not set` +
    HABITATION_TYPE_NAME_APARTMENT + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_DIVORCED +
    MARITAL_STATUS_NAME_MARRIED + MARITAL_STATUS_NAME_WIDOWED +
    TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_1 + LastTaxYear2_TAX_CLASS_CD_1E +
    LastTaxYear2_TAX_CLASS_CD_2 + LastTaxYear2_TAX_CLASS_CD_2F +
    LastTaxYear3_TAX_CLASS_CD_0 + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2 +
    LastTaxYear3_TAX_CLASS_CD_2F
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1     15538      18703
2     15458      18530 80   172.45 9.722e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Six Months Ahead

```
Analysis of Deviance Table

Model 1: as.factor(AktivEtterPassiv) ~ ProductId + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + FLI_AMT + SFLI_AMT + SumAvailable +
    CountPaidToCCL12 + CountRoundPaidToRepaymentLoanL12 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_sumAvail +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    HABITATION_TYPE_NAME_RENTER + TAX_CLASS_CD_Unknown
Model 2: as.factor(AktivEtterPassiv) ~ Alder + Kjønn + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + APPLIED_CREDIT_LIMIT_AMT +
    GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    MORTGAGES_AMT + DebtRegisterNum + DebtRegisterIELA + HOMEOWNER_IND +
    HOUSING_COOPERATIVE_IND + NoOfChildren + FLI_AMT + SFLI_AMT +
    SumAvailable + Applied_vs_Granted + SumPaidToCCL12 +
    SumPaidToRepaymentLoanL12 + CountPaidToRepaymentLoanL12 +
    CountPaidToCCL12 + CountDistinctPaidToRepaymentLoanL12 +
    CountDistinctPaidToCCL12 + CountRoundPaidToRepaymentLoanL12 +
    CountRoundPaidToCCL12 + AIRLINE_12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HOTEL_MOTEL_12 + HARDWARE_12 +
    INTERIOR_FURNISHINGS_12 + OTHER_RETAIL_12 + OTHER_SERVICES_12 +
    OTHER_TRANSPORT_12 + RECREATION_12 + RESTAURANTS_BARS_12 +
    SPORTING_TOY_STORES_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    QUASI_CASH_12 + AIRLINE_3 + ELECTRIC_APPLIANCE_3 + FOOD_STORES_WAREHOUSE_3 +
    HOTEL_MOTEL_3 + HARDWARE_3 + INTERIOR_FURNISHINGS_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_debt + Missing_sumAvail + `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    ApplicationSalesChannel_Responsside + EMPLOYMENT_TYPE_NAME_AT_HOME +
    EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER + EMPLOYMENT_TYPE_NAME_OTHER +
    EMPLOYMENT_TYPE_NAME_RETIREE + EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED +
    EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY + EMPLOYMENT_TYPE_NAME_STUDENT +
    EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE + EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Between 1 and 3 years` +
    `EMPLOYMENT_DURATION_DESC_Less than 1 year` +
    HABITATION_TYPE_NAME_APARTMENT + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_DIVORCED +
    MARITAL_STATUS_NAME_MARRIED + MARITAL_STATUS_NAME_WIDOWED +
    TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_1 + LastTaxYear2_TAX_CLASS_CD_1E +
    LastTaxYear2_TAX_CLASS_CD_2 + LastTaxYear2_TAX_CLASS_CD_2F +
    LastTaxYear3_TAX_CLASS_CD_0 + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2 +
    LastTaxYear3_TAX_CLASS_CD_2F
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1     11615      13078
2     11539      12939 76   139.28 1.313e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Twelve Months Ahead

```
Analysis of Deviance Table
```

```
Model 1: as.factor(AktivEtterPassiv) ~ ProductId + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + MndUtenKortbrukFørPerioden +
    STUDENT_LOAN_AMT + RESTAURANTS_BARS_12 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_sumAvail + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + HABITATION_TYPE_NAME_RENTER +
    TAX_CLASS_CD_Unknown
Model 2: as.factor(AktivEtterPassiv) ~ Alder + Kjønn + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + APPLIED_CREDIT_LIMIT_AMT +
    GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    MORTGAGES_AMT + DebtRegisterNum + DebtRegisterIELA + HOMEOWNER_IND +
    HOUSING_COOPERATIVE_IND + NoOfChildren + FLI_AMT + SFLI_AMT +
    SumAvailable + Applied_vs_Granted + SumPaidToCCL12 +
    SumPaidToRepaymentLoanL12 + CountPaidToRepaymentLoanL12 +
    CountPaidToCCL12 + CountRoundPaidToRepaymentLoanL12 +
    CountRoundPaidToCCL12 + AIRLINE_12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HOTEL_MOTEL_12 + HARDWARE_12 +
    INTERIOR_FURNISHINGS_12 + OTHER_RETAIL_12 + OTHER_SERVICES_12 +
    OTHER_TRANSPORT_12 + RECREATION_12 + RESTAURANTS_BARS_12 +
    SPORTING_TOY_STORES_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    QUASI_CASH_12 + AIRLINE_3 + ELECTRIC_APPLIANCE_3 + FOOD_STORES_WAREHOUSE_3 +
    HOTEL_MOTEL_3 + HARDWARE_3 + INTERIOR_FURNISHINGS_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_debt + Missing_sumAvail + `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    ApplicationSalesChannel_Responsside + EMPLOYMENT_TYPE_NAME_AT_HOME +
    EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER + EMPLOYMENT_TYPE_NAME_OTHER +
    EMPLOYMENT_TYPE_NAME_RETIREE + EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED +
    EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY + EMPLOYMENT_TYPE_NAME_STUDENT +
    EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE + EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Between 1 and 3 years` +
    `EMPLOYMENT_DURATION_DESC_Less than 1 year` +
    HABITATION_TYPE_NAME_APARTMENT + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_DIVORCED +
    MARITAL_STATUS_NAME_MARRIED + MARITAL_STATUS_NAME_WIDOWED +
    TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_1 + LastTaxYear2_TAX_CLASS_CD_1E +
    LastTaxYear2_TAX_CLASS_CD_2 + LastTaxYear2_TAX_CLASS_CD_2F +
    LastTaxYear3_TAX_CLASS_CD_0 + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2 +
    LastTaxYear3_TAX_CLASS_CD_2F
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      7055     7601.9
2      6980     7473.4 75   128.42 0.0001217 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

B.5 ANOVA Analysis of AIC-Reduced and Full Logistic Models

One Month Ahead

```
Analysis of Deviance Table

Model 1: as.factor(AktivEtterPassiv) ~ Alder + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    APPLIED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    SumPaidToCCL12 + CountPaidToRepaymentLoanL12 +
    CountDistinctPaidToRepaymentLoanL12 + AIRLINE_12 +
    ELECTRIC_APPLIANCE_12 + FOOD_STORES_WAREHOUSE_12 +
    HOTEL_MOTEL_12 + INTERIOR_FURNISHINGS_12 + OTHER_TRANSPORT_12 +
    RESTAURANTS_BARS_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    QUASI_CASH_12 + HOTEL_MOTEL_3 + AktiviPerioden + Missing_purchaseSeg +
    Missing_application + Missing_sumAvail +
    ApplicationSalesChannel_Kredittbanken +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER +
    EMPLOYMENT_TYPE_NAME_OTHER + EMPLOYMENT_TYPE_NAME_RETIREE +
    EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE + EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Not set` + HABITATION_TYPE_NAME_APARTMENT +
    HABITATION_TYPE_NAME_OTHER + HABITATION_TYPE_NAME_PARENTS +
    HABITATION_TYPE_NAME_RENTER + MARITAL_STATUS_NAME_COHABITING +
    TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_2F + LastTaxYear3_TAX_CLASS_CD_2F
Model 2: as.factor(AktivEtterPassiv) ~ Alder + Kjønn + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + APPLIED_CREDIT_LIMIT_AMT +
    GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    MORTGAGES_AMT + DebtRegisterNum + DebtRegisterIELA + HOMEOWNER_IND +
    HOUSING_COOPERATIVE_IND + NoOfChildren + FLI_AMT + SFLI_AMT +
    SumAvailable + Applied_vs_Granted + SumPaidToCCL12 +
    SumPaidToRepaymentLoanL12 + CountPaidToRepaymentLoanL12 +
    CountPaidToCCL12 + CountDistinctPaidToRepaymentLoanL12 +
    CountDistinctPaidToCCL12 + CountRoundPaidToRepaymentLoanL12 +
    CountRoundPaidToCCL12 + AIRLINE_12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HOTEL_MOTEL_12 + HARDWARE_12 +
    INTERIOR_FURNISHINGS_12 + OTHER_RETAIL_12 + OTHER_SERVICES_12 +
    OTHER_TRANSPORT_12 + RECREATION_12 + RESTAURANTS_BARS_12 +
    SPORTING_TOY_STORES_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    QUASI_CASH_12 + AIRLINE_3 + ELECTRIC_APPLIANCE_3 + FOOD_STORES_WAREHOUSE_3 +
    HOTEL_MOTEL_3 + HARDWARE_3 + INTERIOR_FURNISHINGS_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_debt + Missing_sumAvail + `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Kredittbanken + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_AT_HOME + EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER +
    EMPLOYMENT_TYPE_NAME_OTHER + EMPLOYMENT_TYPE_NAME_RETIREE +
    EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED + EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY +
    EMPLOYMENT_TYPE_NAME_STUDENT + EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE +
    EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Between 1 and 3 years` +
    `EMPLOYMENT_DURATION_DESC_Less than 1 year` +
    `EMPLOYMENT_DURATION_DESC_Not set` +
    HABITATION_TYPE_NAME_APARTMENT + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_DIVORCED +
    MARITAL_STATUS_NAME_MARRIED + MARITAL_STATUS_NAME_WIDOWED +
    TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_1E + LastTaxYear2_TAX_CLASS_CD_2 +
    LastTaxYear2_TAX_CLASS_CD_2F + LastTaxYear2_TAX_CLASS_CD_Unknown +
    LastTaxYear3_TAX_CLASS_CD_0 + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2 +
    LastTaxYear3_TAX_CLASS_CD_2F
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1     20554      25493
2     20506      25470 48   23.499   0.9989
```

Three Months Ahead

```
Analysis of Deviance Table

Model 1: as.factor(AktivEtterPassiv) ~ Alder + Kjønn + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + GRANTED_CREDIT_LIMIT_AMT +
    GROSS_INCOME_AMT + STUDENT_LOAN_AMT + SumAvailable + SumPaidToCCL12 +
    CountDistinctPaidToRepaymentLoanL12 + CountRoundPaidToRepaymentLoanL12 +
    AIRLINE_12 + ELECTRIC_APPLIANCE_12 + FOOD_STORES_WAREHOUSE_12 +
    OTHER_RETAIL_12 + RECREATION_12 + RESTAURANTS_BARS_12 + TRAVEL_AGENCIES_12 +
    VEHICLES_12 + FOOD_STORES_WAREHOUSE_3 + HOTEL_MOTEL_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_debt +
    Missing_sumAvail + `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Kredittbanken + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER + EMPLOYMENT_TYPE_NAME_RETIREE +
    EMPLOYMENT_TYPE_NAME_UNEMPLOYED + `EMPLOYMENT_DURATION_DESC_Not set` +
    HABITATION_TYPE_NAME_RENTER + MARITAL_STATUS_NAME_COHABITING +
    MARITAL_STATUS_NAME_WIDOWED + TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown +
    LastTaxYear2_TAX_CLASS_CD_0 + LastTaxYear2_TAX_CLASS_CD_2 +
    LastTaxYear3_TAX_CLASS_CD_1 + LastTaxYear3_TAX_CLASS_CD_1E +
    LastTaxYear3_TAX_CLASS_CD_2F
Model 2: as.factor(AktivEtterPassiv) ~ Alder + Kjønn + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + APPLIED_CREDIT_LIMIT_AMT +
    GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    MORTGAGES_AMT + DebtRegisterNum + DebtRegisterIELA + HOMEOWNER_IND +
    HOUSING_COOPERATIVE_IND + NoOfChildren + FLI_AMT + SFLI_AMT +
    SumAvailable + Applied_vs_Granted + SumPaidToCCL12 +
    SumPaidToRepaymentLoanL12 + CountPaidToRepaymentLoanL12 +
    CountPaidToCCL12 + CountDistinctPaidToRepaymentLoanL12 +
    CountDistinctPaidToCCL12 + CountRoundPaidToRepaymentLoanL12 +
    CountRoundPaidToCCL12 + AIRLINE_12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HOTEL_MOTEL_12 + HARDWARE_12 +
    INTERIOR_FURNISHINGS_12 + OTHER_RETAIL_12 + OTHER_SERVICES_12 +
    OTHER_TRANSPORT_12 + RECREATION_12 + RESTAURANTS_BARS_12 +
    SPORTING_TOY_STORES_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    QUASI_CASH_12 + AIRLINE_3 + ELECTRIC_APPLIANCE_3 + FOOD_STORES_WAREHOUSE_3 +
    HOTEL_MOTEL_3 + HARDWARE_3 + INTERIOR_FURNISHINGS_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_debt + Missing_sumAvail + `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Kredittbanken + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_AT_HOME + EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER +
    EMPLOYMENT_TYPE_NAME_OTHER + EMPLOYMENT_TYPE_NAME_RETIREE +
    EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED + EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY +
    EMPLOYMENT_TYPE_NAME_STUDENT + EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE +
    EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Between 1 and 3 years` +
    `EMPLOYMENT_DURATION_DESC_Less than 1 year` +
    `EMPLOYMENT_DURATION_DESC_Not set` +
    HABITATION_TYPE_NAME_APARTMENT + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_DIVORCED +
    MARITAL_STATUS_NAME_MARRIED + MARITAL_STATUS_NAME_WIDOWED +
    TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_1 + LastTaxYear2_TAX_CLASS_CD_1E +
    LastTaxYear2_TAX_CLASS_CD_2 + LastTaxYear2_TAX_CLASS_CD_2F +
    LastTaxYear3_TAX_CLASS_CD_0 + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2 +
    LastTaxYear3_TAX_CLASS_CD_2F
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1     15503      18554
2     15458      18530 45   23.784   0.9961
```

Six Months Ahead

```
Analysis of Deviance Table

Model 1: as.factor(AktivEtterPassiv) ~ Alder + ProductId +
    TransaksjonerMensPassiv + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + APPLIED_CREDIT_LIMIT_AMT +
    GROSS_INCOME_AMT + STUDENT_LOAN_AMT + MORTGAGES_AMT + NoOfChildren +
    FLI_AMT + SFLI_AMT + SumAvailable + CountPaidToCCL12 +
    CountRoundPaidToRepaymentLoanL12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HARDWARE_12 +
    VEHICLES_12 + FOOD_STORES_WAREHOUSE_3 + HOTEL_MOTEL_3 + HARDWARE_3 +
    AktiviPerioden + Missing_purchaseSeg + Missing_application +
    Missing_purchaseHist + Missing_debt + Missing_sumAvail +
    `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_RETIREE + EMPLOYMENT_TYPE_NAME_STUDENT +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_WIDOWED + TAX_CLASS_CD_Unknown +
    LastTaxYear2_TAX_CLASS_CD_0 + LastTaxYear2_TAX_CLASS_CD_2F +
    LastTaxYear3_TAX_CLASS_CD_1 + LastTaxYear3_TAX_CLASS_CD_1E +
    LastTaxYear3_TAX_CLASS_CD_2F
Model 2: as.factor(AktivEtterPassiv) ~ Alder + Kjønn + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + APPLIED_CREDIT_LIMIT_AMT +
    GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    MORTGAGES_AMT + DebtRegisterNum + DebtRegisterIELA + HOMEOWNER_IND +
    HOUSING_COOPERATIVE_IND + NoOfChildren + FLI_AMT + SFLI_AMT +
    SumAvailable + Applied_vs_Granted + SumPaidToCCL12 +
    SumPaidToRepaymentLoanL12 + CountPaidToRepaymentLoanL12 +
    CountPaidToCCL12 + CountDistinctPaidToRepaymentLoanL12 +
    CountDistinctPaidToCCL12 + CountRoundPaidToRepaymentLoanL12 +
    CountRoundPaidToCCL12 + AIRLINE_12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HOTEL_MOTEL_12 + HARDWARE_12 +
    INTERIOR_FURNISHINGS_12 + OTHER_RETAIL_12 + OTHER_SERVICES_12 +
    OTHER_TRANSPORT_12 + RECREATION_12 + RESTAURANTS_BARS_12 +
    SPORTING_TOY_STORES_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    QUASI_CASH_12 + AIRLINE_3 + ELECTRIC_APPLIANCE_3 + FOOD_STORES_WAREHOUSE_3 +
    HOTEL_MOTEL_3 + HARDWARE_3 + INTERIOR_FURNISHINGS_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_debt + Missing_sumAvail + `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    ApplicationSalesChannel_Responsside + EMPLOYMENT_TYPE_NAME_AT_HOME +
    EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER + EMPLOYMENT_TYPE_NAME_OTHER +
    EMPLOYMENT_TYPE_NAME_RETIREE + EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED +
    EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY + EMPLOYMENT_TYPE_NAME_STUDENT +
    EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE + EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Between 1 and 3 years` +
    `EMPLOYMENT_DURATION_DESC_Less than 1 year` +
    HABITATION_TYPE_NAME_APARTMENT + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_DIVORCED +
    MARITAL_STATUS_NAME_MARRIED + MARITAL_STATUS_NAME_WIDOWED +
    TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_1 + LastTaxYear2_TAX_CLASS_CD_1E +
    LastTaxYear2_TAX_CLASS_CD_2 + LastTaxYear2_TAX_CLASS_CD_2F +
    LastTaxYear3_TAX_CLASS_CD_0 + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2 +
    LastTaxYear3_TAX_CLASS_CD_2F
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1     11588      12968
2     11539      12939 49   28.837   0.9904
```

Twelve Months Ahead

```
Analysis of Deviance Table

Model 1: as.factor(AktivEtterPassiv) ~ Alder + ProductId +
    TransaksjonerMensPassiv + MndUtenKortbrukiPerioden +
    MndFraFørsteTilSisteBruk + MndUtenKortbrukFørPerioden +
    GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    MORTGAGES_AMT + NoOfChildren + SumAvailable + CountPaidToRepaymentLoanL12 +
    CountRoundPaidToRepaymentLoanL12 + CountRoundPaidToCCL12 +
    ELECTRIC_APPLIANCE_12 + RESTAURANTS_BARS_12 + TRAVEL_AGENCIES_12 +
    AIRLINE_3 + HOTEL_MOTEL_3 + AktiviPerioden + Missing_purchaseSeg +
    Missing_application + Missing_purchaseHist + Missing_debt +
    Missing_sumAvail + ApplicationSalesChannel_Mobilbank +
    ApplicationSalesChannel_Nettbank + ApplicationSalesChannel_Responsside +
    EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED + EMPLOYMENT_TYPE_NAME_STUDENT +
    EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_DIVORCED +
    MARITAL_STATUS_NAME_MARRIED + TAX_CLASS_CD_Unknown +
    LastTaxYear2_TAX_CLASS_CD_0 + LastTaxYear2_TAX_CLASS_CD_1E
Model 2: as.factor(AktivEtterPassiv) ~ Alder + Kjønn + ProductId +
    TransaksjonerMensPassiv + AntallPassivPerioder +
    MndUtenKortbrukiPerioden + MndFraFørsteTilSisteBruk +
    MndUtenKortbrukFørPerioden + APPLIED_CREDIT_LIMIT_AMT +
    GRANTED_CREDIT_LIMIT_AMT + GROSS_INCOME_AMT + STUDENT_LOAN_AMT +
    MORTGAGES_AMT + DebtRegisterNum + DebtRegisterIELA + HOMEOWNER_IND +
    HOUSING_COOPERATIVE_IND + NoOfChildren + FLI_AMT + SFLI_AMT +
    SumAvailable + Applied_vs_Granted + SumPaidToCCL12 +
    SumPaidToRepaymentLoanL12 + CountPaidToRepaymentLoanL12 +
    CountPaidToCCL12 + CountRoundPaidToRepaymentLoanL12 +
    CountRoundPaidToCCL12 + AIRLINE_12 + ELECTRIC_APPLIANCE_12 +
    FOOD_STORES_WAREHOUSE_12 + HOTEL_MOTEL_12 + HARDWARE_12 +
    INTERIOR_FURNISHINGS_12 + OTHER_RETAIL_12 + OTHER_SERVICES_12 +
    OTHER_TRANSPORT_12 + RECREATION_12 + RESTAURANTS_BARS_12 +
    SPORTING_TOY_STORES_12 + TRAVEL_AGENCIES_12 + VEHICLES_12 +
    QUASI_CASH_12 + AIRLINE_3 + ELECTRIC_APPLIANCE_3 + FOOD_STORES_WAREHOUSE_3 +
    HOTEL_MOTEL_3 + HARDWARE_3 + INTERIOR_FURNISHINGS_3 + AktiviPerioden +
    Missing_purchaseSeg + Missing_application + Missing_purchaseHist +
    Missing_debt + Missing_sumAvail +
    `ApplicationSalesChannel_Autentisert web` +
    ApplicationSalesChannel_Mobilbank + ApplicationSalesChannel_Nettbank +
    ApplicationSalesChannel_Responsside + EMPLOYMENT_TYPE_NAME_AT_HOME +
    EMPLOYMENT_TYPE_NAME_DISABILITY_PENSIONER + EMPLOYMENT_TYPE_NAME_OTHER +
    EMPLOYMENT_TYPE_NAME_RETIREE + EMPLOYMENT_TYPE_NAME_SELF_EMPLOYED +
    EMPLOYMENT_TYPE_NAME_SOCIAL_SECURITY + EMPLOYMENT_TYPE_NAME_STUDENT +
    EMPLOYMENT_TYPE_NAME_TEMP_EMPLOYEE + EMPLOYMENT_TYPE_NAME_UNEMPLOYED +
    `EMPLOYMENT_DURATION_DESC_Between 1 and 3 years` +
    `EMPLOYMENT_DURATION_DESC_Less than 1 year` +
    HABITATION_TYPE_NAME_APARTMENT + HABITATION_TYPE_NAME_OTHER +
    HABITATION_TYPE_NAME_PARENTS + HABITATION_TYPE_NAME_RENTER +
    MARITAL_STATUS_NAME_COHABITING + MARITAL_STATUS_NAME_DIVORCED +
    MARITAL_STATUS_NAME_MARRIED + MARITAL_STATUS_NAME_WIDOWED +
    TAX_CLASS_CD_0 + TAX_CLASS_CD_Unknown + LastTaxYear2_TAX_CLASS_CD_0 +
    LastTaxYear2_TAX_CLASS_CD_1 + LastTaxYear2_TAX_CLASS_CD_1E +
    LastTaxYear2_TAX_CLASS_CD_2 + LastTaxYear2_TAX_CLASS_CD_2F +
    LastTaxYear3_TAX_CLASS_CD_0 + LastTaxYear3_TAX_CLASS_CD_1 +
    LastTaxYear3_TAX_CLASS_CD_1E + LastTaxYear3_TAX_CLASS_CD_2 +
    LastTaxYear3_TAX_CLASS_CD_2F
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1      7029     7498.4
2      6980     7473.4 49   24.918   0.9983
```

C Model Output from R for Adaptive Boosting

C.1 One Month Ahead Hyperparameter Tuning

Design of Experiments

Listing 1: Model with main effects and interactions fitted to the 2^{5-1} fractional factorial design

```
Call:
lm.default(formula = BACC ~ .^2, data = plan.one)

Residuals:
       Min         1Q     Median         3Q        Max
-0.0016448 -0.0003315  0.0000000  0.0003315  0.0016448

Coefficients:
             Estimate Std. Error  t value Pr(>|t|)
(Intercept)  6.539e-01  1.646e-04 3971.860  < 2e-16 ***
A1           9.684e-03  1.646e-04   58.826  < 2e-16 ***
B1           1.384e-02  1.646e-04   84.096  < 2e-16 ***
C1           3.183e-03  1.646e-04   19.336 1.61e-12 ***
D1          -5.498e-04  1.646e-04   -3.340 0.004156 **
E1          -1.734e-03  1.646e-04  -10.531 1.33e-08 ***
A1:B1       -3.402e-03  1.646e-04  -20.666 5.77e-13 ***
A1:C1       -1.829e-03  1.646e-04  -11.112 6.22e-09 ***
A1:D1        8.131e-04  1.646e-04    4.939 0.000148 ***
A1:E1        3.030e-04  1.646e-04    1.840 0.084318 .
B1:C1       -5.135e-04  1.646e-04   -3.119 0.006605 **
B1:D1        5.602e-04  1.646e-04    3.403 0.003640 **
B1:E1        6.365e-04  1.646e-04    3.867 0.001367 **
C1:D1        9.699e-05  1.646e-04    0.589 0.563979
C1:E1       -1.644e-05  1.646e-04   -0.100 0.921695
D1:E1       -4.940e-04  1.646e-04   -3.001 0.008468 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.0009313 on 16 degrees of freedom
Multiple R-squared:  0.9986,    Adjusted R-squared:  0.9973
F-statistic: 776.8 on 15 and 16 DF,  p-value: < 2.2e-16
```

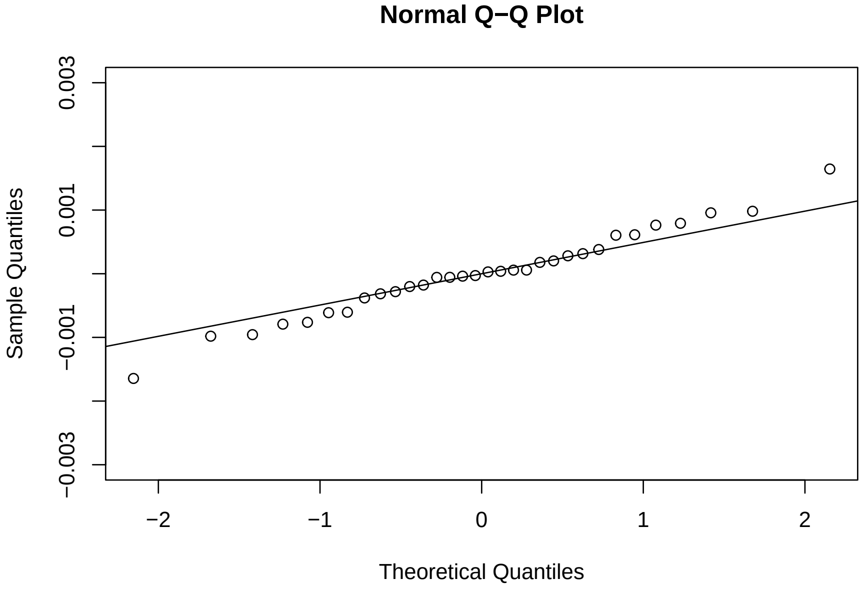


Figure C.1: Normal Q-Q plot of the residuals from the model with main effects and interactions fitted to the 2^{5-1} fractional factorial design for the one_month_ahead data frame with AdaBoost.

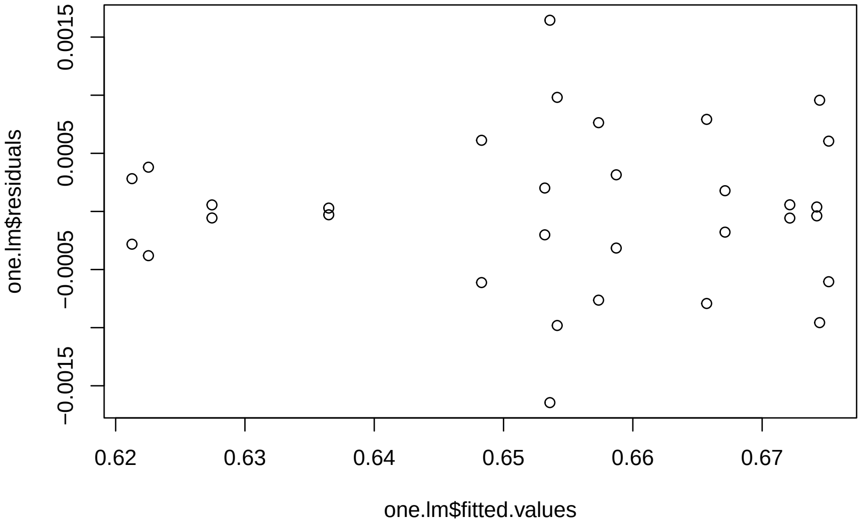


Figure C.2: Plot of the residuals from the model with main effects and interactions fitted to the 2^{5-1} fractional factorial design for the one_month_ahead data frame with AdaBoost.

Table C.1: Results of the 2^{5-1} fractional factorial design with all 32 runs of AdaBoost on the one_month_ahead data frame.

Experiment No.	A	B	C	D	E	Level code	BACC
1	-1	-1	-1	-1	1	*e*	0.6210
2	1	-1	-1	-1	-1	*a*	0.6534
3	-1	1	-1	-1	-1	*b*	0.6590
4	1	1	-1	-1	1	*abe*	0.6721
5	-1	-1	1	-1	-1	*c*	0.6365
6	1	-1	1	-1	1	*ace*	0.6552
7	-1	1	1	-1	1	*bce*	0.6649
8	1	1	1	-1	-1	*abc*	0.6743
9	-1	-1	-1	1	-1	*d*	0.6222
10	1	-1	-1	1	1	*ade*	0.6489
11	-1	1	-1	1	1	*bde*	0.6551
12	1	1	-1	1	-1	*abd*	0.6745
13	-1	-1	1	1	1	*cde*	0.6274
14	1	-1	1	1	-1	*acd*	0.6581
15	-1	1	1	1	-1	*bcd*	0.6673
16	1	1	1	1	1	*abcde*	0.6754
17	-1	-1	-1	-1	1	*e*	0.6215
18	1	-1	-1	-1	-1	*a*	0.6530
19	-1	1	-1	-1	-1	*b*	0.6584
20	1	1	-1	-1	1	*abe*	0.6722
21	-1	-1	1	-1	-1	*c*	0.6365
22	1	-1	1	-1	1	*ace*	0.6519
23	-1	1	1	-1	1	*bce*	0.6665
24	1	1	1	-1	-1	*abc*	0.6742
25	-1	-1	-1	1	-1	*d*	0.6229
26	1	-1	-1	1	1	*ade*	0.6477
27	-1	1	-1	1	1	*bde*	0.6532
28	1	1	-1	1	-1	*abd*	0.6758
29	-1	-1	1	1	1	*cde*	0.6275
30	1	-1	1	1	-1	*acd*	0.6566
31	-1	1	1	1	-1	*bcd*	0.6670
32	1	1	1	1	1	*abcde*	0.6735

Listing 2: First-order model with two-way interaction terms fitted to the 2^3 factorial design with center points

```
Call:
rsm(formula = BACC ~ FO(x1, x2, x3) + TWI(x1, x2, x3), data = centerPlan.one.coded)

               Estimate  Std. Error   t value Pr(>|t|)
(Intercept)  6.7742e-01  2.9413e-04 2303.1205  < 2e-16 ***
x1          -2.6899e-05  3.4490e-04   -0.0780  0.93887
x2           5.6103e-04  3.4490e-04    1.6267  0.12463
x3          -9.6147e-04  3.4490e-04   -2.7877  0.01380 *
x1:x2        4.2012e-05  3.4490e-04    0.1218  0.90467
x1:x3       -2.8285e-04  3.4490e-04   -0.8201  0.42500
x2:x3       -8.6672e-04  3.4490e-04   -2.5130  0.02389 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Multiple R-squared:  0.5374,    Adjusted R-squared:  0.3524
F-statistic: 2.904 on 6 and 15 DF,  p-value: 0.04389

Analysis of Variance Table

Response: BACC
                Df     Sum Sq    Mean Sq F value  Pr(>F)
FO(x1, x2, x3)   3 1.9839e-05 6.6129e-06  3.4745 0.04283
TWI(x1, x2, x3)  3 1.3328e-05 4.4425e-06  2.3341 0.11523
Residuals       15 2.8549e-05 1.9033e-06
Lack of fit      2 1.0732e-05 5.3662e-06  3.9154 0.04667
Pure error      13 1.7817e-05 1.3705e-06

Stationary point of response surface:
        x1         x2         x3
-0.3425763 -0.9874422  0.5469884

Stationary point in original units:
           A            B            C
699.30677652   5.02511557   0.07093977

Eigenanalysis:
eigen() decomposition
$values
[1]  0.0004623965  0.0000000000 -0.0004500230
```

```
$vectors
         [,1]        [,2]      [,3]
x1 -0.2450392  0.95031124 0.1920010
x2 -0.6688808 -0.30906604 0.6760744
x3  0.7018221  0.03723894 0.7113783
```

Table C.2: Results of the first central composite design with three factors for AdaBoost on the one_month_ahead data frame.

Experiment No.	A	B	C	BACC
1	-1	-1	-1	0.6761
2	1	-1	-1	0.6771
3	-1	1	-1	0.6784
4	1	1	-1	0.6801
5	-1	-1	1	0.6759
6	1	-1	1	0.6762
7	-1	1	1	0.6764
8	1	1	1	0.6743
9	-1	-1	-1	0.6765
10	1	-1	-1	0.6765
11	-1	1	-1	0.6798
12	1	1	-1	0.6792
13	-1	-1	1	0.6776
14	1	-1	1	0.6757
15	-1	1	1	0.6755
16	1	1	1	0.6767
17	0	0	0	0.6764
18	0	0	0	0.6797
19	0	0	0	0.6769
20	0	0	0	0.6791
21	0	0	0	0.6796
22	0	0	0	0.6797
23	-1.73	0	0	0.6781
24	1.73	0	0	0.6779
25	0	-1.5	0	0.6752
26	0	1.5	0	0.6769
27	0	0	-1.75	0.6762
28	0	0	1.75	0.6752

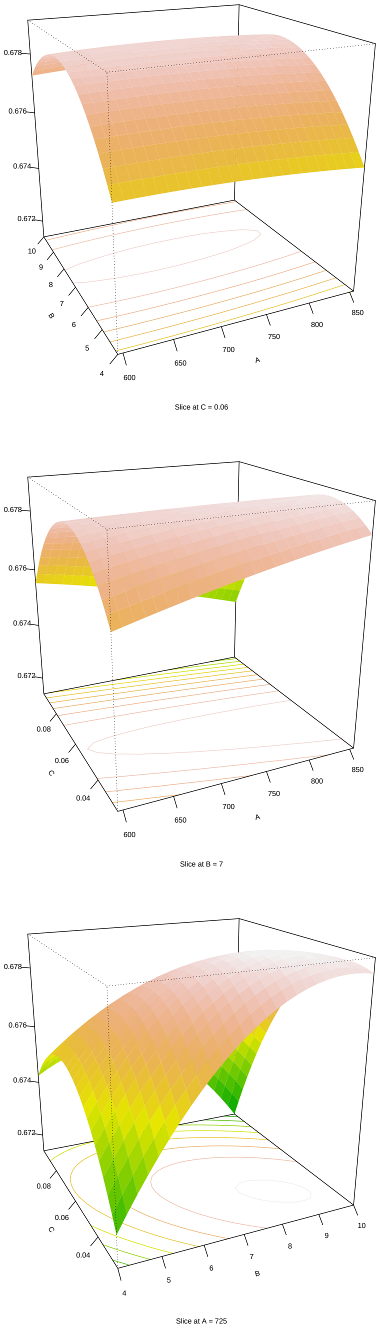
Listing 3: Second-order model fitted to the first central composite design

```
Call:
rsm(formula = BACC ~ SO(x1, x2, x3), data = centerPlan.wAxial.coded.one)

               Estimate  Std. Error   t value  Pr(>|t|)
(Intercept)  6.7851e-01  4.4523e-04 1523.9664 < 2.2e-16 ***
x1          -3.1120e-05  2.3324e-04   -0.1334  0.895339
x2           5.6368e-04  2.4167e-04    2.3324  0.031486 *
x3          -7.7827e-04  2.3263e-04   -3.3456  0.003600 **
x1:x2        4.2012e-05  2.7355e-04    0.1536  0.879651
x1:x3       -2.8285e-04  2.7355e-04   -1.0340  0.314834
x2:x3       -8.6672e-04  2.7355e-04   -3.1684  0.005319 **
x1^2        -3.6604e-05  2.6824e-04   -0.1365  0.892971
x2^2        -8.5235e-04  3.2824e-04   -2.5967  0.018224 *
x3^2        -7.9149e-04  2.6418e-04   -2.9960  0.007752 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Multiple R-squared:  0.7063,    Adjusted R-squared:  0.5595
F-statistic:  4.81 on 9 and 18 DF,  p-value: 0.002263

Analysis of Variance Table

Response: BACC
                Df     Sum Sq    Mean Sq F value   Pr(>F)
FO(x1, x2, x3)   3 1.9936e-05 6.6454e-06  5.5504 0.007076
TWI(x1, x2, x3)  3 1.3328e-05 4.4425e-06  3.7105 0.030766
PQ(x1, x2, x3)   3 1.8565e-05 6.1882e-06  5.1685 0.009430
Residuals       18 2.1551e-05 1.1973e-06
Lack of fit      5 3.7341e-06 7.4680e-07  0.5449 0.739540
Pure error      13 1.7817e-05 1.3705e-06

Stationary point of response surface:
        x1         x2         x3
-0.3146176  0.6951044 -0.7759167

Stationary point in original units:
           A            B            C
701.40367873   8.39020876   0.04448167

Eigenanalysis:
```

```
eigen() decomposition
$values
[1]  0.0000000000 -0.0004226938 -0.0012617767

$vectors
           [,1]       [,2]       [,3]
x1  0.9558047  0.2862929 0.06688671
x2  0.1510755 -0.6734386 0.72364122
x3 -0.2522174  0.6815547 0.68692762
```

Table C.3: Results of the second central composite design with three factors for AdaBoost on the one_month_ahead data frame.

Experiment No.	A	B	C	BACC
1	-1	-1	-1	0.6792
2	1	-1	-1	0.6782
3	-1	1	-1	0.6809
4	1	1	-1	0.6769
5	-1	-1	1	0.6824
6	1	-1	1	0.6781
7	-1	1	1	0.6834
8	1	1	1	0.6808
9	0	0	0	0.6801
10	0	0	0	0.6787
11	0	0	0	0.6830
12	-1.73	0	0	0.6783
13	1.73	0	0	0.6795
14	0	-2	0	0.6774
15	0	2	0	0.6827
16	0	0	-1.73	0.6794
17	0	0	1.73	0.6814
18	0	0	0	0.6823
19	0	0	0	0.6793
20	0	0	0	0.6783



Slice at C = 0.06

Slice at B = 7

Slice at A = 725

Figure C.3: Perspective plots of the first fitted second-order response surface model with AdaBoost.

Listing 4: Second-order model fitted to the second central composite design

```
Call:
rsm(formula = BACC ~ SO(x1, x2, x3), data = ccd2.one)

               Estimate  Std. Error  t value Pr(>|t|)
(Intercept)  6.8027e-01  8.0009e-04 850.2327   <2e-16 ***
x1          -7.0713e-04  5.2378e-04  -1.3500   0.2068
x2           9.4394e-04  5.2378e-04   1.8022   0.1017
x3           9.3041e-04  5.2378e-04   1.7763   0.1061
x1:x2       -1.4615e-04  6.9290e-04  -0.2109   0.8372
x1:x3       -2.4029e-04  6.9290e-04  -0.3468   0.7359
x2:x3        4.1505e-04  6.9290e-04   0.5990   0.5625
x1^2        -4.2249e-04  4.9383e-04  -0.8555   0.4123
x2^2        -3.8804e-05  4.9383e-04  -0.0786   0.9389
x3^2         8.0394e-05  4.9383e-04   0.1628   0.8739
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Multiple R-squared:  0.4886,    Adjusted R-squared:  0.02839
F-statistic: 1.062 on 9 and 10 DF,  p-value: 0.4596

Analysis of Variance Table

Response: BACC
                Df     Sum Sq    Mean Sq F value  Pr(>F)
FO(x1, x2, x3)   3 3.1594e-05 1.0531e-05  2.7419 0.09893
TWI(x1, x2, x3)  3 2.0110e-06 6.7030e-07  0.1745 0.91120
PQ(x1, x2, x3)   3 3.0950e-06 1.0317e-06  0.2686 0.84660
Residuals       10 3.8409e-05 3.8409e-06
Lack of fit      5 1.9462e-05 3.8923e-06  1.0271 0.48864
Pure error       5 1.8947e-05 3.7895e-06

Stationary point of response surface:
         x1          x2          x3
 0.08110566 -1.23527388 -2.47668778

Stationary point in original units:
           A            B            C
851.08292433   7.76472612   0.02201656

Eigenanalysis:
```

```
eigen() decomposition
$values
[1]  0.0002652016 -0.0001944120 -0.0004516854

$vectors
         [,1]        [,2]      [,3]
x1  0.1995111 -0.05080095 0.9785778
x2 -0.5847279 -0.80753906 0.0772917
x3 -0.7863133  0.58762227 0.1908177
```

Table C.4: Results of the third central composite design with three factors for AdaBoost on the one_month_ahead data frame.

Experiment No.	A	B	C	BACC
1	-1	-1	-1	0.6801
2	1	-1	-1	0.6812
3	-1	1	-1	0.6793
4	1	1	-1	0.6783
5	-1	-1	1	0.6812
6	1	-1	1	0.6754
7	-1	1	1	0.6796
8	1	1	1	0.6796
9	0	0	0	0.6815
10	0	0	0	0.6838
11	0	0	0	0.6800
12	-1.73	0	0	0.6789
13	1.73	0	0	0.6808
14	0	-2	0	0.6772
15	0	2	0	0.6800
16	0	0	-1.73	0.6812
17	0	0	1.73	0.6798
18	0	0	0	0.6808
19	0	0	0	0.6799
20	0	0	0	0.6804

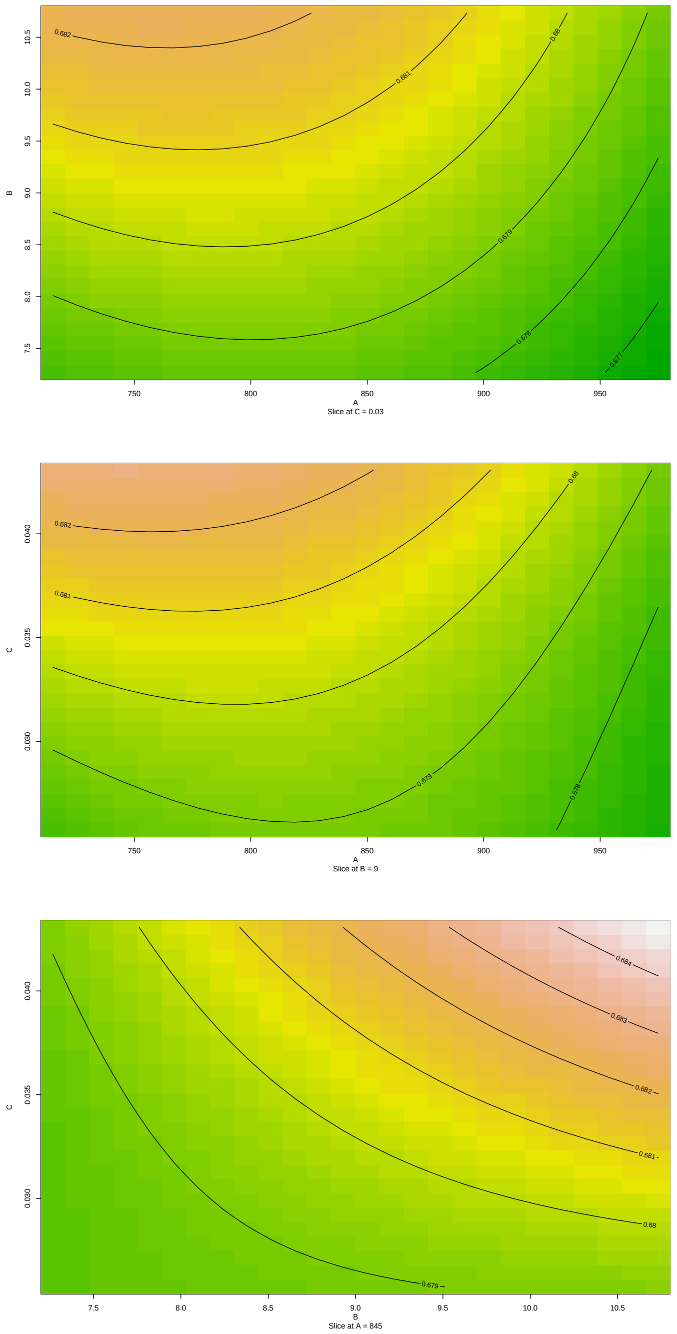
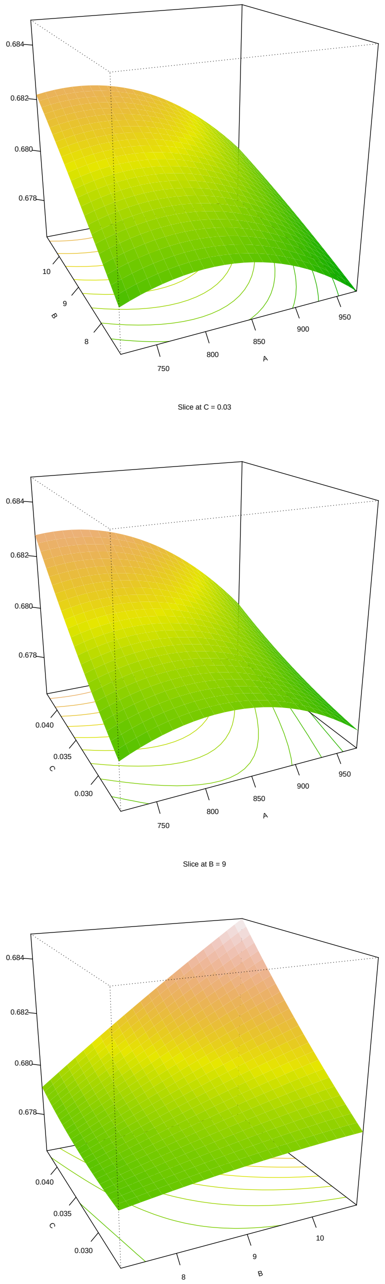


Figure C.4: Contour plots of the second fitted second-order response surface model with AdaBoost.



Slice at C = 0.03

Slice at B = 9

Slice at A = 845

Figure C.5: Perspective plots of the second fitted second-order response surface model with AdaBoost.

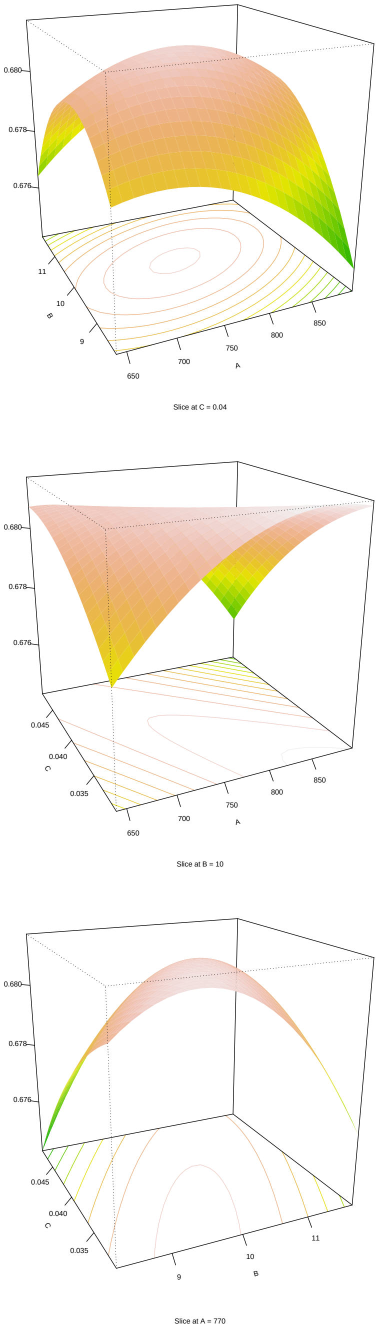
Listing 5: Second-order model fitted to the third central composite design

```
Call:
rsm(formula = BACC ~ SO(x1, x2, x3), data = ccd.best.one)

               Estimate  Std. Error   t value Pr(>|t|)
(Intercept)  0.68105646  0.00066806 1019.4476  < 2e-16 ***
x1          -0.00017035  0.00043735   -0.3895  0.70507
x2           0.00026420  0.00043735    0.6041  0.55924
x3          -0.00039815  0.00043735   -0.9104  0.38405
x1:x2        0.00046821  0.00057856    0.8093  0.43719
x1:x3       -0.00072876  0.00057856   -1.2596  0.23642
x2:x3        0.00078537  0.00057856    1.3575  0.20448
x1^2        -0.00046147  0.00041234   -1.1192  0.28923
x2^2        -0.00086595  0.00041234   -2.1001  0.06207 .
x3^2        -0.00025735  0.00041234   -0.6241  0.54651
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Multiple R-squared:  0.5159,    Adjusted R-squared:  0.08028
F-statistic: 1.184 on 9 and 10 DF,  p-value: 0.3954

Analysis of Variance Table

Response: BACC
                Df     Sum Sq    Mean Sq F value Pr(>F)
FO(x1, x2, x3)   3 3.6028e-06 1.2009e-06  0.4485 0.7239
TWI(x1, x2, x3)  3 1.0937e-05 3.6456e-06  1.3614 0.3100
PQ(x1, x2, x3)   3 1.4002e-05 4.6674e-06  1.7430 0.2213
Residuals       10 2.6779e-05 2.6779e-06
Lack of fit      5 1.6117e-05 3.2233e-06  1.5116 0.3307
Pure error       5 1.0662e-05 2.1324e-06

Stationary point of response surface:
         x1          x2          x3
-0.15261610  0.08670294 -0.11157355

Stationary point in original units:
           A            B            C
758.55379237  10.08670294   0.03884213

Eigenanalysis:
```

```
eigen() decomposition
$values
[1]  0.0000000000 -0.0004015866 -0.0012440404

$vectors
          [,1]      [,2]       [,3]
x1  0.4828926 0.7534786  0.4461891
x2 -0.2353936 0.6024729 -0.7626377
x3 -0.8434481 0.2632420  0.4682937
```

Table C.5: Results of AdaBoost performed along the canonical path starting from the stationary point proposed by the third second-order model.

Distance	n.trees	interaction.depth	shrinkage	BACC
-2	686	11	0.04728	0.6809
-1	722	10	0.04306	0.6810
0	759	10	0.03884	0.6819
1	795	10	0.03463	0.6794
2	831	10	0.03041	0.6803
3	867	9	0.02619	0.6805
4	903	9	0.02198	0.6793
5	940	9	0.01776	0.6806



Slice at C = 0.04

Slice at B = 10

Slice at A = 770

Figure C.6: Perspective plots of the third fitted second-order response surface model with AdaBoost.

C.2 Twelve Months Ahead Hyperparameter Tuning

Design of Experiments

Table C.6: Results of the 2^{5-1} fractional factorial design with all 32 runs of AdaBoost on the twelve_months_ahead data frame.

Experiment No.	A	B	C	D	E	Level code	BACC
1	-1	-1	-1	-1	1	*e*	0.6107
2	1	-1	-1	-1	-1	*a*	0.6535
3	-1	1	-1	-1	-1	*b*	0.6686
4	1	1	-1	-1	1	*abe*	0.6746
5	-1	-1	1	-1	-1	*c*	0.6351
6	1	-1	1	-1	1	*ace*	0.6533
7	-1	1	1	-1	1	*bce*	0.6738
8	1	1	1	-1	-1	*abc*	0.6649
9	-1	-1	-1	1	-1	*d*	0.6064
10	1	-1	-1	1	1	*ade*	0.6561
11	-1	1	-1	1	1	*bde*	0.6667
12	1	1	-1	1	-1	*abd*	0.6733
13	-1	-1	1	1	1	*cde*	0.6417
14	1	-1	1	1	-1	*acd*	0.6525
15	-1	1	1	1	-1	*bcd*	0.6739
16	1	1	1	1	1	*abcde*	0.6656
17	-1	-1	-1	-1	1	*e*	0.6062
18	1	-1	-1	-1	-1	*a*	0.6525
19	-1	1	-1	-1	-1	*b*	0.6697
20	1	1	-1	-1	1	*abe*	0.6719
21	-1	-1	1	-1	-1	*c*	0.6370
22	1	-1	1	-1	1	*ace*	0.6556
23	-1	1	1	-1	1	*bce*	0.6622
24	1	1	1	-1	-1	*abc*	0.6675
25	-1	-1	-1	1	-1	*d*	0.6085
26	1	-1	-1	1	1	*ade*	0.6585
27	-1	1	-1	1	1	*bde*	0.6707
28	1	1	-1	1	-1	*abd*	0.6810
29	-1	-1	1	1	1	*cde*	0.6495
30	1	-1	1	1	-1	*acd*	0.6527
31	-1	1	1	1	-1	*bcd*	0.6729
32	1	1	1	1	1	*abcde*	0.6688

Listing 6: Linear model fitted to the 2^{5-1} fractional factorial design

```
Call:
lm.default(formula = BACC ~ .^2, data = plan.twelve)

Residuals:
      Min        1Q    Median        3Q       Max
-0.005753 -0.001214  0.000000  0.001214  0.005753

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept)  6.549e-01  5.714e-04 1146.010  < 2e-16 ***
A1           7.774e-03  5.714e-04   13.605 3.27e-10 ***
B1           1.551e-02  5.714e-04   27.139 8.28e-15 ***
C1           3.068e-03  5.714e-04    5.369 6.27e-05 ***
D1           1.299e-03  5.714e-04    2.272   0.0372 *
E1           5.007e-04  5.714e-04    0.876   0.3938
A1:B1       -7.201e-03  5.714e-04  -12.603 1.01e-09 ***
A1:C1       -5.606e-03  5.714e-04   -9.811 3.58e-08 ***
A1:D1       -3.836e-04  5.714e-04   -0.671   0.5116
A1:E1       -8.881e-05  5.714e-04   -0.155   0.8784
B1:C1       -4.748e-03  5.714e-04   -8.310 3.38e-07 ***
B1:D1       -7.852e-05  5.714e-04   -0.137   0.8924
B1:E1       -1.590e-03  5.714e-04   -2.782   0.0133 *
C1:D1        4.566e-04  5.714e-04    0.799   0.4360
C1:E1        3.659e-04  5.714e-04    0.640   0.5310
D1:E1        3.029e-03  5.714e-04    5.301 7.17e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.003232 on 16 degrees of freedom
Multiple R-squared:  0.988,     Adjusted R-squared:  0.9768
F-statistic: 87.86 on 15 and 16 DF,  p-value: 1.774e-12
```

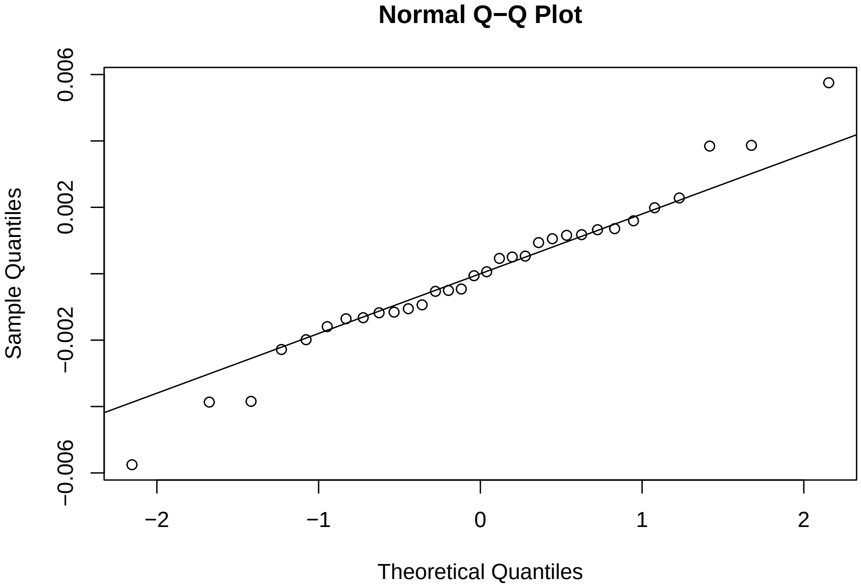


Figure C.7: Normal Q-Q plot of the residuals from the linear model fitted to the 2^{5-1} fractional factorial design for the twelve_months_ahead data frame with AdaBoost.

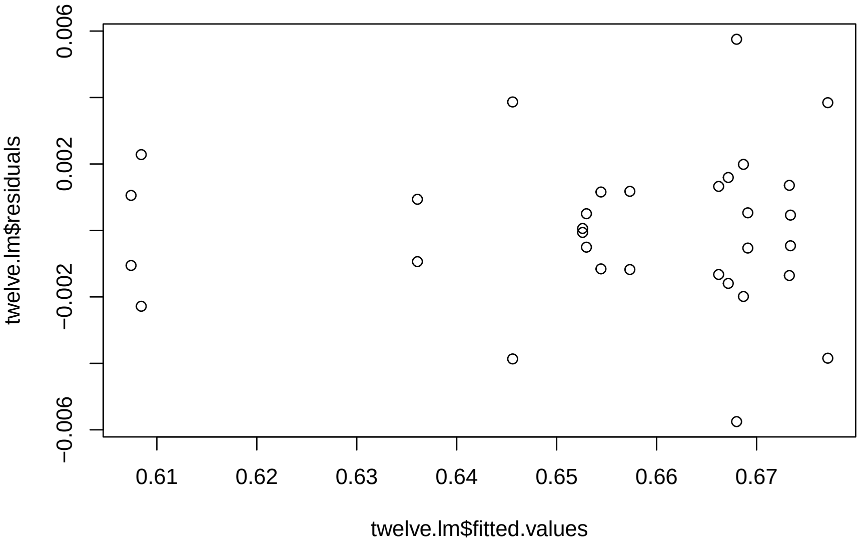


Figure C.8: Plot of the residuals from the linear model fitted to the 2^{5-1} fractional factorial design for the twelve_months_ahead data frame with AdaBoost.

Table C.7: Results of the first 2^3 factorial design with center runs, with three factors for AdaBoost on the twelve_months_ahead data frame.

Experiment No.	A	B	C	BACC
1	-1	-1	-1	0.6712
2	1	-1	-1	0.6690
3	-1	1	-1	0.6748
4	1	1	-1	0.6738
5	-1	-1	1	0.6717
6	1	-1	1	0.6662
7	-1	1	1	0.6720
8	1	1	1	0.6745
9	-1	-1	-1	0.6657
10	1	-1	-1	0.6671
11	-1	1	-1	0.6680
12	1	1	-1	0.6698
13	-1	-1	1	0.6691
14	1	-1	1	0.6674
15	-1	1	1	0.6723
16	1	1	1	0.6752
17	0	0	0	0.6738
18	0	0	0	0.6734
19	0	0	0	0.6712
20	0	0	0	0.6764
21	0	0	0	0.6754
22	0	0	0	0.6712
23	0	0	0	0.6702
24	0	0	0	0.6694

Listing 7: Model with first-order and interaction effects fitted to the first 2^3 factorial design with center points

```
Call:
rsm(formula = BACC ~ FO(x1, x2, x3) + TWI(x1, x2, x3), data = centerPlan.twelve.coded)

               Estimate  Std. Error   t value  Pr(>|t|)
(Intercept)  0.67120077  0.00056044 1197.6306 < 2.2e-16 ***
x1          -0.00012483  0.00068640   -0.1819  0.857841
x2           0.00206863  0.00068640    3.0138  0.007822 **
x3           0.00055647  0.00068640    0.8107  0.428733
x1:x2        0.00088695  0.00068640    1.2922  0.213586
x1:x3       -0.00011514  0.00068640   -0.1677  0.868759
x2:x3        0.00039649  0.00068640    0.5776  0.571081
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Multiple R-squared:  0.4098,    Adjusted R-squared:  0.2015
F-statistic: 1.967 on 6 and 17 DF,  p-value: 0.1274

Analysis of Variance Table

Response: BACC
                Df     Sum Sq    Mean Sq F value  Pr(>F)
FO(x1, x2, x3)   3 7.3672e-05 2.4557e-05  3.2577 0.04739
TWI(x1, x2, x3)  3 1.5314e-05 5.1047e-06  0.6772 0.57791
Residuals       17 1.2815e-04 7.5382e-06
Lack of fit      2 3.2312e-05 1.6156e-05  2.5286 0.11315
Pure error      15 9.5838e-05 6.3892e-06

Stationary point of response surface:
        x1         x2         x3
-1.9049418 -0.2910015 -0.8099988

Stationary point in original units:
           A            B            C
207.12936649   2.70899854   0.04190001

Eigenanalysis:
eigen() decomposition
$values
[1]  0.0004664568  0.0000000000 -0.0005090857
```

```
$vectors
          [,1]        [,2]       [,3]
x1 0.6566514  0.40037629  0.6391461
x2 0.7198802 -0.08001975 -0.6894703
x3 0.2249033 -0.91285030  0.3407680
```

Table C.8: Results of the second 2^3 factorial design with center runs, with three factors for AdaBoost on the twelve_months_ahead data frame.

Experiment No.	A	B	C	BACC
1	-1	-1	-1	0.6744
2	1	-1	-1	0.6712
3	-1	1	-1	0.6793
4	1	1	-1	0.6745
5	-1	-1	1	0.6757
6	1	-1	1	0.6654
7	-1	1	1	0.6775
8	1	1	1	0.6670
9	-1	-1	-1	0.6767
10	1	-1	-1	0.6689
11	-1	1	-1	0.6717
12	1	1	-1	0.6646
13	-1	-1	1	0.6689
14	1	-1	1	0.6778
15	-1	1	1	0.6695
16	1	1	1	0.6722
17	0	0	0	0.6697
18	0	0	0	0.6770
19	0	0	0	0.6678
20	0	0	0	0.6751
21	0	0	0	0.6748
22	0	0	0	0.6748
23	0	0	0	0.6688
24	0	0	0	0.6727

Listing 8: Model with first-order and interaction effects fitted to the second 2^3 factorial design with center points

```
Call:
rsm(formula = BACC ~ FO(x1, x2, x3) + TWI(x1, x2, x3), data = centerPlan2.twelve.coded.2)

               Estimate  Std. Error  t value Pr(>|t|)
(Intercept)  6.7233e-01  8.7500e-04 768.3720  < 2e-16 ***
x1          -2.0087e-03  1.0717e-03  -1.8744  0.07817 .
x2          -1.7364e-04  1.0717e-03  -0.1620  0.87320
x3          -4.5279e-04  1.0717e-03  -0.4225  0.67795
x1:x2       -4.7381e-04  1.0717e-03  -0.4421  0.66396
x1:x3        8.5773e-04  1.0717e-03   0.8004  0.43453
x2:x3       -2.8067e-05  1.0717e-03  -0.0262  0.97941
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Multiple R-squared:  0.2113,    Adjusted R-squared:  -0.06705
F-statistic: 0.7591 on 6 and 17 DF,  p-value: 0.6114

Analysis of Variance Table

Response: BACC
                Df     Sum Sq    Mean Sq F value Pr(>F)
FO(x1, x2, x3)   3 6.8318e-05 2.2773e-05  1.2393 0.3263
TWI(x1, x2, x3)  3 1.5376e-05 5.1252e-06  0.2789 0.8398
Residuals       17 3.1237e-04 1.8375e-05
Lack of fit      2 2.7400e-06 1.3700e-06  0.0664 0.9361
Pure error      15 3.0963e-04 2.0642e-05

Stationary point of response surface:
       x1         x2         x3
 0.2693378 -0.9911022  1.7872070

Stationary point in original units:
          A            B            C
363.46689096   5.00889779   0.05893603

Eigenanalysis:
eigen() decomposition
$values
[1]  0.0004959790  0.0000000000 -0.0004841025
```

```
$vectors
         [,1]       [,2]       [,3]
x1  0.7027295 0.01526496 -0.7112933
x2 -0.3531387 0.87540100 -0.3301002
x3  0.6176279 0.48315637  0.6205608
```

NTNU
Norwegian University of
Science and Technology

SpareBank 1