Norwegian School of Economics

Bergen, Autumn 2020

Distinguishing Potential Child Insurance Customers

A Statistical Investigation

Çaglar Altunel & Hallvard Holte

Supervisor: Geir Drage Berentsen

Master's thesis, MSc in Economics and Business Administration

Major: Business Analytics

NORWEGIAN SCHOOL OF ECONOMICS

This thesis was written as a part of the Master of Science in Economics and Business Administration at NHH. Please note that neither the institution nor the examiners are responsible − through the approval of this thesis − for the theories and methods used, or results and conclusions drawn in this work.

Acknowledgements

Both of us first of all wish to express our deepest gratitude to our supervisor Geir Drage Berentsen. He has consistently provided us not only with exceptionally quick responses but especially helpful guidance. Writing this thesis would have been a much more difficult and stressful process without his help. Secondly, we would like to show our gratitude to the insurance company who suggested that a thesis could be written on this topic. They quickly provided us with a rich data set to work with, and were patient and kind in clarifying our many questions about this data. Without their initiative and trust in us this thesis could not have been written.

Hallvard wants to express his particular gratitude to Çaglar, who, despite having a busy family life raising a young daughter and having recently moved to Norway, never once complained when Hallvard at times could not properly work due to a extended period of illness. His trust and patience avoided turning this into a serious problem during the writing of this thesis. He also wants to thank his parents for their constant patience and support.

Çaglar wants to express his deepest thanks to Hallvard, who gave his 100% to this thesis despite his long illness during the entire semester. Çaglar is also grateful to Hallvard for providing him with perfect studying conditions in his apartment, which was priceless during the ongoing pandemic. Finally, Çaglar would like to thank his wife for her never-ending support, patience and unmatched attitude which made him always feel privileged.

We give no thanks to the following nuisances: the weather of Bergen, Covid-19 and Lyme disease.

Table of Contents

Acknowledgements 2
Table of Contents 3
Executive summary 5
1 Introduction 6
1.1 The problem 6
1.2 The products 6
1.3 A note on the industry 7
1.4 Overview of thesis 7
2 Methodology & literature review 8
2.1 Introduction 8
2.2 Time series methods 8
2.2.1 Autocorrelation (ACF) and Partial Autocorrelation Functions (PACF) 8
2.2.2 Change Point Analysis 10
2.3 Supervised learning methods 12
2.3.1 Supervised vs. unsupervised learning methods 12
2.3.2 Logistic regression 12
2.3.3 XGBoost 17
2.4 Unsupervised learning methods 19
2.4.1 K-means & K-modes clustering 19
2.5 Other methods 22
2.5.1 Mean substitution imputation 22
2.5.2 Stepwise variable selection methods 23
2.5.3 Akaike information criterion (AIC) 24
2.6 Terminology 25
3 Data explanation and exploration 26
3.1 Introduction 26
3.1.2 Explanation of variables in data set 27
3.1.2 Distribution of missing data 29
3.2 Discussion of internal data 31
3.2.1 Correlations: plot and discussion 31
3.2.2 Selected histograms for continuous data 33
3.2.3 Selected bar plots for categorical data 35
3.3 Discussion of external data 38
3.3.1 Correlations: plot and discussion 38
3.3.2 Selected histograms for continuous data 40
3.3.3 Selected bar plots for categorical data 41
3.4 Analysis of binaries indicating external data and sale 44

4 Time series analysis of hit rate 46

4.1 Investigation of the time dependency 46

4.2 Change Point Analysis: detecting the impact of the new product 49

5 Logistic regression model to predict sales 52

5.1 Introduction 52

5.2 Data preparation 53

5.2.1 Sales channels issue 53

5.2.2 Clean-up of selected variables 54

5.2.3 Controlling for time dependency 58

5.3 Variable selection: backwards stepwise 59

5.4 Validation of the model 59

5.4.1 Splitting the data into train and test sets 60

5.4.2 Re-training of model and evaluation 60

5.5 Discussion of final model and findings 61

5.5.1 Presentation of the best-fitted model & evaluation of accuracy 61

5.5.2 Interpretation and conclusion 64

6 Machine learning models to predict product type 69

6.1 Introduction 69

6.2 Data set 69

6.3 Explanation of models selected 69

6.3.1 Supervised learning 1: logistic regression 69

6.3.2 Supervised learning 2: XGBoost 70

6.3.3 Unsupervised learning: K-means and K-modes clustering 70

6.4 Results and evaluation of models selected 71

6.4.1 Supervised learning 1: logistic regression 71

6.4.2 Supervised learning 2: XGBoost 73

6.4.3 Unsupervised learning: K-means and K-modes clustering 74

6.5 Conclusion 77

7 Summary and conclusion 79

7.1 Summary of findings 79

7.2 Synthesis of findings 81

Bibliography 83

Appendix A: table of training model details 86

Appendix B: details of logistic model for product type 87

Appendix C: Figures of the outcome of K-modes clustering 88

Executive summary

In this thesis we try to illuminate possible reasons why the launch of a more affordable child insurance product by an established Norwegian insurance company failed to live up to the company's expectations. We use three main approaches.

First, to better understand the situation, we perform a change point analysis on the ratio of sales to offers from 2014 to 2020. We confirm the company's problem by establishing that the one significant increase detected cannot have been caused by the new product.

Secondly, to understand what drives sales in general, we create an easily interpretable logistic regression model to predict whether an offer is likely to result in a sale, taking into account both product types. Our most surprising finding here is the fact that the presence of certain data that the company gets from a third party business intelligence firm, and only has for about half the customers, by itself is associated with a significantly higher likelihood of purchasing child insurance. Regardless of the content of the data, its presence itself highly affects this likelihood. We suspect this is because only wealthier or more selective customers appear in this external database.

Thirdly, we use two supervised methods to predict whether an offer involves the standard or new product, based on a range of customer characteristics. These fail. We then use two different unsupervised clustering methods, to see if it is possible to identify customer groups with clear preferences for one of the two products. This too fails. None of these statistical methods, successful in predicting and understanding sales, can identify characteristics or profiles associated with the new product.

We interpret these failures as meaning that no customer segment significantly prefers the new product over the old. Slight evidence from one of the cluster analyses also suggests that a more premium product rather than a more economical one could have been more successful. Our final conclusion is therefore that the economy product was unsuccessful because it appealed to a non-existent customer segment.

1 Introduction

1.1 The problem

This thesis is the result of a familiar problem in business: a newly launched product that fails to live up to the company's expectations. The company in question is a large and established Norwegian insurance company that used to sell just one type of child insurance. Believing that it would be possible to increase its sales by launching a second, cheaper child insurance product offering less coverage, they launched such a product in the middle of 2018. Some two years later, disappointing sales forced the company to conclude that this was not the success they had hoped for.

The company therefore contacted NHH to propose that a master's thesis be written on this topic. The purpose of such a thesis would then be to help illuminate some of the reasons why this budget product performed below expectations. Since the writers of this thesis are majoring in the Business Analytics profile, our focus will be on using statistical techniques to give answers to this question. In order to do so, the company provided us with detailed data about each child insurance offer made to customers going back to 2014, giving us a total data set containing approximately 85,000 observations. Since the budget product was launched in the middle of 2018, we hence have data about offers both before and after the product's introduction. Some of these offers resulted in sales, while others did not. On the basis of this large data set, we will perform various investigations and train different predictive models, all of which will be outlined shortly.

1.2 The products

We will here explain the basic differences between the two products, which we will henceforth refer to as the "standard product" and the "economy product" respectively. Both child insurance products offer payouts in the event of accidents or illnesses involving children or young adults up to the age of 18. Some of these payouts are done once, as when an accident occurs, while others are recurring, such as in the case of a lasting or permanent disability. There is a range of different conditions and amounts depending on the specific type of accident or medical incident involved. The essential difference between the two products is that the standard product costs twice as much as the economy product, while in most cases also guaranteeing payouts that are twice as large. In a few cases the payouts involved are identical.

1.3 A note on the industry

During our initial discussions with the company, an important fact about the Norwegian child insurance market came to light: offering price discounts on these products is not legal. This ruled out what would otherwise have been a natural and indeed obvious strategy for dealing with this problem. Had such discounts been legal and regularly applied, we could have analyzed the sales rates associated with different price points to examine the customers' price sensitivity and see whether the economy product simply is overpriced. Since this was not possible, we were instead forced to rely on other techniques to answer why this product performed below the company's expectations. This is what we will now explain.

1.4 Overview of thesis

The investigations contained in this thesis are divided into four main parts, each of which partially builds on the results and conclusions of the preceding ones. Before these investigations, we will in section 2 explain the methods to be used and review some of their previous applications in the academic literature. Then the first main part of the thesis, section 3, will involve describing and exploring the data set. Its purpose is to understand the main features and relationships present in the data, in order to find out what parts of the data are most suited for the analyses to be performed in the later sections. The second main part, section 4, will examine the ratio of sales to offers (the "hit" rate) over time. We here want to find out whether this changes over time, and especially whether any such potential change can be related to the introduction of the new product.

The third and fourth parts of the investigation, sections 5 and 6, involve training various predictive models, and using the results from these to draw real-world conclusions. In section 5, we will train an interpretable model to predict the probability of an offer resulting in a sale. Creating such a model will help us to understand which factors are most important in predicting sales and how they affect that probability. In section 6 we will use three different models to try to predict the type of product involved in the offer, i.e. whether it is the standard or economy product. The point here is to investigate which customer characteristics are most associated with each of the two product types, in order to see whether we can identify separate customer segments. By following such a procedure we hope to gain a number of insights into why the economy product failed to sell as well as the company had hoped.

2 Methodology & literature review

2.1 Introduction

Since this thesis aims to provide insights into the company's problem by statistically analyzing the data set they provided us with, the relevant academic literature will be about statistical methods and their applications. For this reason, we have decided that it makes sense to combine an outline of the methodology to be used in the thesis with a review of the literature concerning these methods. We feel that such a structure is more natural and cohesive than separating this into two sections where we outline the methods and then discuss the literature separately. In the rest of this section, we have grouped the methods according to their basic type, and will for each method first explain it, then discuss some previous applications.

2.2 Time series methods

The data set that we will use was built in such a way that each observation is an offer of a child insurance product. Each of these offers has a time stamp, and the entire period of the observations spans from the beginning of 2014 up to mid-2020. For this reason, we wish to investigate whether the dependent variable we will use in section 5, namely sales, has a temporal pattern or not. To detect the existence of such a time dependency, we will use autocorrelation (ACF) and partial autocorrelation functions (PACF). We will also perform a change point analysis to determine whether the introduction of the new product had an impact on weekly sales per offer or not.

2.2.1 Autocorrelation (ACF) and Partial Autocorrelation Functions (PACF)

The ACF measures the impact of the earlier values of a time series on later values, whilst PACF shows the correlation between a variable and its lagged values that cannot be explained by the correlation of all other low-order lags (Zhang et al., 2014). The ACF formula can be summarized as follows (Zaiontz, 2020):

Let the ACF at lag k: r_k

With the mean of the times series given by:

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

The autocovariance function at lag k (for $k \geq 0$) is given by:

$$s_k = \frac{1}{n} \sum_{i=k+1}^{n} (y_i - \bar{y})(y_{i-k} - \bar{y})$$

Let s_0 be the variance of the time series. Then, the ACF at lag k is:

$$r_k = \frac{s_k}{s_0}$$

Whereas the h^{th} order PACF formula for a time series is (Zaiontz, 2020):

$$\frac{cov(y_i, y_{i-h} | y_{i-1}, \dots, y_{i-h+1})}{\sqrt{var(y_i | y_{i-1}, \dots, y_{i-h+1})} * \sqrt{var(y_{i-h} | y_{i-1}, \dots, y_{i-h+1})}}$$

The ACF and PACF functions have been used to detect temporal patterns in various research fields. Juang et al. (2017) used time-series analysis on a wide data set that spans from January 2009 to December 2016 and built an autoregressive integrated moving average (ARIMA) model to forecast emergency department visits at a medical center in Taiwan. They used the ACF and PACF to determine the orders of the autoregressive and moving-average parts of this model. Shuai et al. (2019) built an ARIMA model using a technique similar to the one used by Juang et al. (2017), to forecast gross domestic product growth rates in Shenzen and Shanghai separately. In order to determine whether our dependent variable has a temporal pattern or not, we will also use ACF and PACF as in the abovementioned examples. However, our final aim in using these functions differs significantly from theirs. Juang et al. (2017) and Shuai et al. (2019) used ACF and PACF correlograms in order to stationarize their dependent variables, as they directly used the lagged values that explain the variations in their dependent variables. However, in our case, we will use ACF and PACF to detect the time dependency of the dependent variable, but will make use of it to create temporal control variables for each offer in our subsequent models. This is because the structure of our data set is different from the ones used in the abovementioned studies. While each time has a unique dependent variable value in those studies, our data set consists of observations which each represent a unique offer, which can hence occur repeatedly at a given time. For this reason, we will create a new times series data set that contains the weekly average of sales per offer for each week. On top of that, we will make use of ACF and PACF to inspect temporal patterns. Detected temporal patterns (in weekly lag terms) will be used as temporal explanatory variables in our predictive models. Such integrated approaches have previously been used in research. To give examples, Abolfazli et al. (2014) used ACF and PACF to determine the best explanatory variables for a neural network model that they constructed to forecast rail transport petroleum consumption. Pethukova et al. (2018) compared the predictive accuracies of ARIMA, generalized linear autoregressive moving-average and random forest time series regression models for predicting influenza A virus frequencies at weekly and monthly intervals in swine in Ontario. The main difference between the method used in our thesis and theirs is that we eventually predict the probability of each offer resulting in a sale (as there is no fixed number of offers per day), while the dependent variables in the abovementioned studies are unique per time period.

2.2.2 Change Point Analysis

As we briefly explained in section 1.1, the company changed its single product policy in child insurance and launched an economy variant after years of selling a well-established standard product. Furthermore, the company wants to know the reasons for the failure in sales of the economy product since its introduction. Theoretically, if the launch of the new product variant had had a positive impact on sales, one could argue that the average sales per offer rate should have demonstrated an increasing trend starting some time after the launch of the product. Therefore, in section 4.2 of this thesis, we we will investigate this issue with a change point analysis, in order to detect whether there is a change point in average sales per offer at any point in time after the introduction of the new product.

The detection of change point in a time series can be formally expressed as follows (Killick & Eckley, 2014):

Let us denote our time series as $y_{1:n} = (y_1, \dots, y_n)$. A change point may be present within the set of time $\tau \in \{1, \dots, n-1\}$, such that the properties of $\{y_1, \dots, y_\tau\}$ and $\{y_{\tau+1}, \dots, y_n\}$ statistically differ in a given way.

For example, two sets can differ in terms of mean, variance or both. In our case, we will look for a change point in terms of both mean and variance. It is also possible to split time series into more than two sequences, however, here we exemplify only the case where it is split into two, as we aim to find a single change point, given that only one new product is introduced.

A test statistic is necessary to detect whether a change point occurs in a given time series or not. In change point analysis, to be able to conclude that there is change point in a series, the difference between the maximum value of the sum of the log-likelihoods of the time series sequences divided by a change point and the maximum log-likelihood of the null hypothesis is expected to exceed a certain threshold. According to Killick & Eckley (2014), this can be formally expressed as follows:

$\log p(y_{1:n}|\hat{\theta})$, where $p(.)$ is the probability density function of the given time series and $\hat{\theta}$ is the maximum likelihood estimate of the parameters.

Now let us suppose there is a change point in the given times series at τ. Then, the maximum value of the sum of the log-likelihoods of the time series sequences is $ML(.)$ and the change point occurs at $\max ML(\tau)$:

$$\max ML(\tau) = \log(\, y_{1:\tau}|\widehat{\theta_1}) + \log\,(y_{\tau+1:n}|\widehat{\theta_2})$$

If λ (formulated in the below) exceeds the certain threshold c:

$$\lambda > c, \text{ where } \lambda = 2(max\ ML(\tau) - \log p(y_{1:n}|\hat{\theta}))$$

Change point analysis covers a wide range of methods which have been used in various research areas. To give examples, Beaulieu et al. (2012) used change point analysis in order to detect abrupt changes in the climate, while Thies & Molnár (2018) used change point analysis techniques to detect the breaking points in the volatility and average return of the Bitcoin price. As stated above, our target in using change point analysis is to determine whether there is a certain change point in average weekly sales per offer that may potentially correspond to the introduction of the economy product. Fader et al. (2004) built a dynamic change point model in order to forecast new product sales that take marketing mix variables and customer-level buying rates into account. Yan & Tu (2012) aimed to forecast short-term sales with change-point evaluation and pattern matching algorithms, with a change-point evaluation approach that determines the number of change-points. Our goal in using change point analysis hence differs significantly from the abovementioned studies. They used change point analysis in such a way that it incorporates the forecasting models that they aimed to build, while ours is completely independent of our later modelling. Instead, we will in this thesis use change point analysis as an independent method of analysis, to see whether the introduction of the economy products impacts average sales per offer or not.

2.3 Supervised learning methods

2.3.1 Supervised vs. unsupervised learning methods

Two of the learning methods to be used in this thesis are supervised, while one is unsupervised. Since these are conceptually distinct, we will here briefly explain the difference[1]. Supervised learning involves separating the data into inputs and outputs, and training a model to use the inputs to predict the output as accurately as possible. The logistic regression we perform in section 5 is hence an example of supervised learning, where we use various data about an offer in order to predict whether or not a sale will be made, an output variable chosen by us. Unsupervised learning does not involve such a separation between inputs and outputs made by the modelers. It is called unsupervised because these methods simply detect patterns in the entire data set, without any such distinction between inputs and outputs. Why this too can give us useful insights in a case where we *do* have an output variable (namely product type, as investigated in section 6) will be explained in section 2.4.1 below, concerning unsupervised learning.

2.3.2 Logistic regression

Explanation and intuition

Logistic regression is a classification method, i.e. one that predicts a qualitative response. This means that the method uses some characteristics of an observation (the explanatory variables) to place that observation within one of two or more classes of observations (the predicted class being the dependent variable). An example would be using age, BMI and blood pressure level to classify a person as either being or not being at risk of developing serious heart disease. In such cases, where there are only two possible outcomes, we speak of a binary logistic model, and encode one outcome as 1 and the other as 0. Since these binary models are most commonly used, and are what we will exclusively use in this thesis, this is the kind of model we will explain here. An important and useful feature of logistic regression is that it does not predict the observation's class directly, but rather the *probability* that the observation belongs to that class. So in the example above, the model would not directly classify a person as, e.g., being at risk of developing serious heart disease, but would rather assign a certain probability to this outcome. Clearly, it is useful to know whether

[1] This and the following section, on logistic regression, are fundamentally based on the explanations in James et al. (2013), especially sections 2.1.3 and 4.3.

the probability underlying such a classification is 51% or 99%, and not just the final classification itself. Having now given an intuitive explanation of what the model does, we will now explain the underlying mathematics.

Equation and parameter estimation

In the logistic regression model, the probability of observing a response variable Y equal to one is linked with the explanatory variable X via the logistic function:

$$p(Y = 1) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

The parameters β_0 and β_1 are estimated using maximum likelihood, which amounts to maximizing the (log) probability of observing the data under this model:

$$L(\beta_1) = log\ P(Y_1, \dots, Y_n) = \sum_{i=1}^{n} log\ P(Y_i = 1)$$

This function must be optimized numerically in order to obtain an estimate of β_1.

Multiple logistic regression, i.e. where we have more than one independent variable, is done in practically the same way. The difference is that the initial equation, now with p variables, becomes:

$$p(Y = 1) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

A logistic regression model is an example of a generalized linear model, since the function is an algebraic transformation of the linear model where $p(Y = 1) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$.

Parameter interpretation

What now remains to be explained is one major strength of logistic regression models, namely their relatively high interpretability. Many sophisticated machine learning models can give highly accurate predictions, while at the same time leaving researchers without many insights into how exactly those predictions are made. Such models are for this reason often rather stereotypically described as being black boxes. Logistic regression is on the other side of this spectrum: it lets the researcher sacrifice some predictive power in return for a better understanding of how each individual independent variable influences the dependent variable.

To understand this relationship, we consider the logit equation. Looking at this will help us to understand exactly what the parameters are doing:

$$\log(odds) = \log\left(\frac{p(Y=1)}{1-p(Y=1)}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

This means that if the independent variable X_1 increases by 1 unit, then the log odds changes by β_1. First of all, β_1 therefore tells us something about the direction of the change: its sign shows whether increasing X_1 will increase or decrease the log odds. It can further be shown that an increase in the log odds implies an increase in the odds, and that this in turn implies an increase in the probability $p(Y=1)$ (and vice versa for a decrease). Hence the sign of β_1 tells us whether an increase in X_1 makes it more or less probable that the observation falls within the given class. Furthermore, an increase in the log odds by β_1 implies that the odds themselves increase by a multiplicative factor e^{β_1}. Exactly this is the great strength of logistic regression in terms of interpretability: we can tell exactly how a change in each of the *p* independent variables will influence the odds.

Previous applications

Logistic regression is commonly used in various industries and fields of research, especially where binary classifications are sought. We will here limit ourselves to some applications from the insurance industry, widely considered. Wilson (2009), Guillen et al. (2002) and Astari & Kismiantini (2019) are all papers whose most important model is a binary logistic model, and where this model is used to study data that comes from a field related to insurance. We will now outline their methods and results to see what we can learn for our own purposes.

Wilson (2009) describes the creation of a binary logistic regression model to predict cases of car insurance fraud, and subsequently evaluates the results of this model. The data set is rather small, consisting of only 100 observations, where 50% of the observations fall within each of the two categories to be predicted (i.e. fraudulent and legitimate claims). The explanatory variables are factors such as the number of years the customer has been with the company, number of claims filed per year and whether the customer is new (took out the insurance policy within a year or less before the claim in question was filed). Before setting up the logistic model, the researcher checks the correlations between each of these variables, in order to find out whether some are especially highly correlated. Since most absolute correlations are found to be no higher than about 0.2, multicollinearity[2] is judged not to be an issue. A logistic model is here used since it does not simply classify each observation, but instead provides probabilities. This is useful in the case of fraud detection, since the insurance company may then decide to further investigate cases which are especially likely to be fraudulent, without wasting time and resources on those cases that are more ambiguous. The result of the logistic regression is that only two explanatory variables are found to have statistically significant coefficients, number of claims filed per year and whether the customer is new. The trained model has an overall accuracy of 0.704, but this quite high score probably involves some overfitting[3]. Due to the small sample size, the researcher decided against using a train/test set split for model validation.

Astari & Kismiantini (2019) use a binary logistic regression model to predict health insurance ownership in Indonesia, based on information from a large survey. The data set consists of 29,508 observations, where approximately the same number are insured and uninsured. Explanatory variables include a range of factors such as age, gender, occupation, education and health status. The researchers state that a logistic regression model was chosen because it is commonly used to model binary response variables, and because it allows them to understand the impact each of these variables has on a person's probability of owning life insurance. They found that the most important explanatory variables, in terms of significance and coefficient size, were higher education, chronic health issues and age. They do not try to validate their results using a test set, and reach an overall accuracy of 0.565 for their model. While this might sound unimpressive, they stress that they have still found causal insights into which factors most affect the probability of owning health insurance.

In Guillen et al. (2002) the researchers use customer data from a large Danish insurance company in order to train a logistic regression model to predict customer retention for the following three months, i.e. whether the customer is going to cancel their insurance plan with the company or not

[2] This concept is defined below, in section 2.6 on terminology.
[3] This too is defined in section 2.6.

within this timeframe. They specifically mention that similar techniques have been used for fraud detection, as we have seen. They have a data set of 232,043 customers, where only about 5% are lapses, i.e. customers not retained. Explanatory variables include factors such as customer age, years since the customer's first insurance policy with the company, whether the customer owns home or car insurance, claims within the last 12 months, and others. They also check for multicollinearity, as above. The results of their final model show that having only car insurance highly increases the probability of a lapse, while higher age and having many products with the company decreases it. An important reason for choosing a logistic regression model is that it quite straightforwardly allows us to come to such conclusions about the impact of specific variables. These coefficients are furthermore all statistically significant. Given the large data set, the researchers in this case split the data into train and test sets in order to validate their results. Due to the highly different number of observations in each of the two classes, the classification threshold was tweaked by the researchers based on their domain knowledge. The final test set accuracy obtained was then 0.653.

Insights from previous applications

We will now summarize these findings to see what we can learn for our own model building. First of all, using a logistic regression model seems very appropriate in our case since we are interested not only in making accurate predictions, but also in real-world interpretability. Beyond just classifying an observation as likely leading to a sale, or likely being of one product type or the other, we are interested in knowing which factors influence this. So, as for Guillen et al. (2002) and Astari & Kismiantini (2019), interpretability is for our purposes an important advantage in using a logistic regression model. Furthermore, as emphasized by Wilson (2009), it is advantageous that the model outputs a probability rather than classifying an observation directly. As in that case, it would for our company be interesting and useful to know whether certain customer profiles are *highly* likely to result in a sale, or *highly* likely to pick one product type over the other, instead of just knowing that one classification is *more* likely than the other. So we see that our analyses will benefit from the same features of logistic regression models as have been emphasized in the literature reviewed.

In terms of the data sets used to train the models, we can also draw certain conclusions. We have seen logistic regression models trained on data sets ranging from 100 observations to approximately 232,000 observations. Our data set of approximately 85,000 observations should then plainly have more than enough data for the model to be trained. Since we have such a large number of observations, we see no reason not to follow Guillen et al. (2002) in using a test set to validate our model. But we note that even without doing this, we can, like Astari & Kismiantini (2019), derive insights into patterns in the data. With a large data set such as ours and that of Guillen et al. (2002) we expect a smaller degree of overfitting[4] than would be the case when using a smaller data set. Before training our model, we should follow Wilson (2009) and Guillen et al. (2002) in noting that including explanatory variables that are highly correlated leads to the problem of multicollinearity. Hence we should check whether any such high correlations exist between our explanatory variables, and if so consider not including sets which are highly correlated.

After training our model, we should check whether the coefficients thus found for each explanatory variable are statistically significant. If so, we have by now established that interpreting their effect is relatively straightforward (the coefficient indicates how much the log odds increase given a one-unit increase in the associated explanatory variable). Given that the economy product constitutes only 13% of total sales from the time it was introduced, we are facing a situation similar to that of Guillen et al. (2002), where the number of observations differs greatly between the two classes. This means that we should study which classification threshold makes most sense from the company's real-world perspective, and not only consider the optimal accuracy score. Finally, we have seen that accuracy scores ranging from 0.565 to 0.704 (probably with some overfitting) are found in these studies. We therefore note that such a range seems to indicate reasonable predictive accuracy, although it is not clear to what extent we can truly generalize between these cases.

2.3.3 XGBoost

Eventually, as per the company's main research question, our focus will be narrowed down to trying to find the reasons why the economy product underperformed compared to the standard one. In section 6, we will approach this problem from several perspectives, using both supervised and unsupervised learning methods. Regarding supervised learning methods, in order to inspect whether there is a statistical way of distinguishing economy product buyers and standard product buyers, we will use logistic regression and XGBoost. XGBoost is a boosting method that has been

[4] Explained below in section 2.6 on terminology.

widely used in data science due to the fact that it has been proven very versatile and effective in terms of predictive performance (Nielsen, 2016). The exact technical details are beyond the scope of our research question, but it is worth mentioning that its main difference from other well-known boosting techniques, such as gradient boosting, is that second-order Taylor expansion in linear approximation constitutes the base of its loss function, instead of first-order in gradient boosting (Zhang et al., 2018). The reason why we do not spend time on the technical details is that our main target is not building the most accurate possible predictive model. Rather, an XGBoost model will only be one part of several statistical experiments conducted to find out whether it is possible to statistically distinguish different product buyers. In other words, as eventually we must find a way to causally interpret how the economy product buyers differ from the other, XGBoost will only be used to approach the problem of whether these products' buyers are statistically seperable or not and to compare its accuracy with the outcome of the logistic regression, given that it is not straightforward to interpret the outcome of boosting models such as XGBoost.

Comparing the accuracies of generalized linear models such as logistic regression and boosting methods is common in business science. To exemplify, Pesantez-Narvaez et al. (2019) compared the predictive performances of logistic regression and XGBoost models using a data set that covers individuals' driving patterns in order to predict the occurrence of accident claims, and concluded that the two models demonstrated a similar test accuracy performance after the problem of overfitting is corrected. Xia et al. (2020) used several predictive modelling methods including logistic regression and XGBoost to find the best-performing method in predicting passenger car sales and concluded that XGBoost demonstrated the lowest test logarithmic difference square root score, which made it the best-performing method out of all of the selected ones, including logistic regression. Similar to those studies, we will compare the accuracies and true negative rates of logistic regression and XGBoost models. However, our goal differs from the abovementioned cases, as we will eventually assess whether there is a significant difference between these methods in terms of classification performances or not, to check how far we can go in terms of distinguishing the economy product buyers from buyers of the standard products.

2.4 Unsupervised learning methods

2.4.1 K-means & K-modes clustering

When investigating ways of statistically distinguishing buyers of the economy and standard product, we will use unsupervised learning methods in addition to the supervised ones that we have discussed above. If there really exist separate customer segments for buyers of the standard and economy products, an unsupervised learning method should plausibly be able to detect these. As will be discussed in a more detailed way in section 5.2, we will prepare a cleaned data set that contains both continuous and categorical explanatory variables for the modelling phase. And given that we *a priori* know the number of clusters to investigate, one for each of the company's two child insurance products, we will make use of K-means clustering. According to James et al. (2013), it is ideal to use K-means clustering if one knows the number of clusters before the analysis. On the other hand, we know that K-means clustering cannot be used on categorical data (Khan & Ahmad, 2013). For this reason, we will pursue two separate unsupervised learning methods on our explanatory variables. We will use K-means clustering on continuous variables, while we will independently treat the categorical ones and perform K-modes clustering on those.

K-means clustering algorithm can be formally expressed as follows (James et al., 2013):

Let us first denote each cluster as $C_1, \dots . C_K$. In our case, $K = 2$ as we know that the company has two different product types. Therefore, our target will be dividing our data set of continuous explanatory variables into two clusters, C_1, C_2. We also have the condition that $C_1 \cap C_2 = \emptyset$, which means each observation have to be assigned to exactly one cluster. The idea is to make within-cluster variation as small as possible, and the measure it is used for this is squared Euclidean distance[5]. Therefore, for each cluster, the optimization problem is:

$$minimize_{C_1,C_2} \left\{ \sum_{k=1}^{2} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \right\}$$

[5] According to James et al. (2013), there are several ways of doing it but the most usual metric that is used to minimize within cluster variation is squared Euclidean distance.

Where $|C_k|$ is the cardinality of the set C_k and p is the all pairs within kth cluster and $(x_{ij} - x_{i'j})^2$ is the squared Euclidean distance of each pair. Accordingly, given that there are 2^n ways to separate n observations into two clusters, the below algorithm simplifies the process and provides a solution that is reasonably good enough to keep the squared Euclidean distances as minimized as possible:

1. To each observation, randomly assign a cluster number (1 or 2, in our case).
2. Iterate the below until the cluster number being assigned to each observation does not change
    a. Cluster centroid is computed for each cluster. Each cluster's centroid is a vector of the means of variables for the observations in each cluster.
    b. Assign each observation to a cluster (1 or 2, in our case) whose centroid is closest, by taking squared Euclidean distances as a benchmark.

K-modes uses a similar algorithm but in order to circumvent the limitations of the use of means and squared Euclidean distances on the categorical data, modes as cluster centers and dissimilarity measure (Hamming distance) are used instead (Khan & Ahmad, 2013). According to Khan & Ahmad (2013), the K-modes process offered by Huang (1998) can be formally expressed as follows:

Let us assume X and Y are two categorical data objects being explained by m categorical attributes. The Hamming distances of X and Y can be expressed by the total amount of mismatches. The higher the Hamming distances are, less similar X and Y are. The similarity (or dissimilarity) of the observations (X and Y) can be denoted as $\partial(x_j, y_j)$. $\partial(x_j, y_j) = 1$ if $x_j = y_j$, $\partial(x_j, y_j) = 0$ if $x_j \neq y_j$. Therefore, the dissimilarity measure ($d(X, Y)$) is:

$$d(X, Y) = \sum_{j=1}^{m} \partial(x_j, y_j)$$

When this dissimilarity measure is for the all categorical observations over each attribute, we reach the so-called cost function $C(Q)$ to minimize, that is:

$$minimize_{C(Q)}\left\{\sum_{i=1}^{n} d(N_i, Q_i)\right\}$$

Where N_i is the ith element and Q_i is the cluster center defined by the modes of given attributes within N_i. As discussed before, $n = 2$ in our case, given that we already know that we have two product types, therefore we need to build two clusters to examine. Khan & Ahmad (2013) explains Huang's (1998) algorithm that minimizes the above function as good as possible as in the below[6]:

1. Assign initially $K = 2$, given that we have two product types, cluster centers for each observation with respect to their distance (similarity in this case).
2. Recalculate the dissimilarity of observations against given modes. If it is found out that an observation's nearest mode belongs to the other cluster, then reallocate the observation to the other cluster.
3. Iterate the second step until there is no change in assigned clusters for each observation.

There is a rich literature concerning the use of unsupervised learning methods, including K-means and K-modes clustering in customer segmentation. Ezenkwu et al. (2015) applied K-means clustering, where the number of clusters is set at $K = 4$, on a data set being gathered from a retailer and identified four clusters, which are classified with respect to how often they visit the store and how much they buy in each visit, with 95% accuracy. Ye et al. (2012) enjoyed the advantage that the telecom industry is heavily data-driven and applied K-means clustering for customer segmentation on the data set of Changzou telecom in Jiangsu province. The authors targeted specifically clustering small-business customers, for whom marketing services are relatively weaker, and they *a priori* determined to cluster them into 8 groups and concluded that customer groups are effectively seperable with respect to consumption characteristics. Kamthania et al. (2018) used a reduced data set of customers' interest and location by principal component analysis to apply K-modes clustering for customer segmentation in the e-commerce business. As with K-means clustering, initial specification of the number of clusters is necessary in order to perform K-modes clustering, so the authors iterated K-modes clustering algorithm for the number of clusters from 2 to 60, and reached an accurate segmentation at $K = 31$. This led the authors to conclude

[6] Khan & Ahmad's (2013) expression of the algorithm is converted to a specific case where $K = 2$ by us, as we already know the number of the clusters to be built.

that the approach they pursued can provide internet business owners and growing start-ups a simplified way to cluster their customers in a detailed manner, which can help them to pursue custumer-based marketing strategies.

In this thesis, while using unsupervised methods to segment customers, our target will be significantly different compared to the abovementioned studies. To make it clear, we eventually will use these methods to detect whether the economy product has a customer segment at all, while defining the customer segments is the main target in the abovementioned studies. Another difference of our thesis in terms of approaching the problem with unsupervised learning methods compared to those is that we will enjoy the advange of having both continuous and categorical variables, so that we will be able to treat them seperately. Showing a similarity to our approach, Aliyev et al. (2020) used numerous methods to reach an accurate customer segmentation for bank customers. They used K-modes clustering in their demographic model but could not reach a satisfactory outcome, so they increased their focus on the other models they built. To be more specific, like Aliyev et al. (2020), we will be able to compare the outcomes of K-means clustering, which will be applied on continuous variables and K-modes clustering, and draw a conclusion accordingly.

2.5 Other methods

2.5.1 Mean substitution imputation

Imputation is the general name for any method that fills missing data using values estimated (or “imputed”) from other parts of the data set. Many such methods exist, ranging from the computational simple to the highly complex. We will only use one such method, namely what is called mean substitution imputation or mean imputation overall, as it is referred to in Kalton & Kasprzyk (1982). This straightforward method simply involves filling any missing values using the arithmetical mean of the existent values for the variable in question. The primary benefit of this method is that it does not change the sample mean for the variable in question. Clearly, adding the mean of a set to that set an arbitrary number of times does not give the resulting set a new mean.

2.5.2 Stepwise variable selection methods

A variable selection algorithm aims to find the optimal set of explanatory variables to include in a model[7]. This is done according to some criterion, such as adjusted R^2 or AIC, which will be explained in the immediately following section. The simplest variable selection algorithm, called best subset selection, simply tries every possible combination of explanatory variables, and selects the best one according to the criterion chosen. Yet when we are dealing with data sets containing a large number of explanatory variables, this approach can quickly become computationally infeasible[8].

Backward stepwise subset selection therefore involves a compromise, where the algorithm is designed to drastically reduce the number of possible models to be considered while ideally still identifying the most important explanatory variables. The algorithm works as follows. First the model is trained using all p explanatory variables, and the chosen criterion (in our case the AIC score) is calculated for this model. Then p new models are trained, where the first, second, third, …, $p - th$ explanatory variable is each in turn left out. This leaves us with p models containing $p – 1$ explanatory variables, and each of these are again scored using the desired criterion. The model with the best score is then reduced according to the same procedure: $p – 1$ models are trained containing the now $p – 2$ remaining explanatory variables. In this way the algorithm at each step discards the explanatory variable which least reduces the chosen criterion. This is not guaranteed to give the *best* possible model (as best subset selection is), but only involves training a total of $1 + p(p + 1)/2$ models. That this may sometimes be a necessary trade-off when p is very high is clear.

Forward step-wise selection uses an analogous algorithm involving the same number of possible models, but works forward from a model with zero explanatory variables instead of backwards from a model with all p explanatory variables. Since we will be using backward stepwise subset selection, and since the principle should be quite clear, we will not describe it in detail here[9]. We will instead explain the main difference between the cases when forward and backward stepwise selection are used. Forward stepwise selection, unlike backward stepwise selection, can be used

[7] This section follows the description given in James et al. (2013), section 6.1.

[8] For p explanatory variables, the method involves considering 2^p possible models. Our raw data set has nearly 40 variables, which means that even if we could eliminate half, we would have to check about 2^{20} possible models, a number that exceeds one million. This would not be computationally feasible with the hardware available to us.

[9] The details can be found in James et al. (2013), section 6.1.2, Algorithm 6.2.

even when $n < p$, i.e. when there are more explanatory variables than observations. Otherwise, backward stepwise selection is preferable, since it starts with a model that includes and therefore considers all explanatory variables simultaneously. If some of the explanatory variables are correlated (multicollinearity), forward stepwise selection might lead to all of these being excluded from the model, while backward stepwise selection would keep all of them, according to Mantel (1970). Since for our data set $n \gg p$, we will therefore opt for backward stepwise selection.

2.5.3 Akaike information criterion (AIC)

In the preceding section about finding the optimal explanatory variables to include in a model, we have been referring to an arbitrary criterion to be used to score and rank different possible models. One such criterion, the AIC score, will now be described. We will also explain why this criterion was chosen. AIC is a more sophisticated counterpart to the familiar adjusted R^2 score, which essentially adds a term to the R^2 statistic that penalizes the addition of relatively unnecessary variables to the model. While adjusted R^2 has a simple equation, the score does not have such a rigorous foundation in statistical theory as the AIC score, according to James et al. (2013).

Since the AIC criterion is suitable for models fit using maximum likelihood[10], as we have seen in the preceding section that logistic regression is, we have opted to use AIC as the criterion in this thesis. This is because we want a criterion that, like adjusted R^2, penalizes the addition of variables to our models, but resting on a sounder basis of statistical theory. We will now state and explain the equation for AIC:

$$AIC = \frac{1}{n\hat{\sigma}^2}(RSS + 2d\hat{\sigma}^2)$$

Here n is the number of observations, $\hat{\sigma}^2$ is the variance of the error term ϵ, RSS is the sum of the squares of the residuals for the trained model, while d is the number of explanatory variables included. We see therefore that the AIC score will increase if the number of included explanatory variables d is increased. Since a smaller AIC score is preferable to a large one, this is how the AIC score penalizes the addition of explanatory variables that are relatively unnecessary in a given model.

[10] James et al. (2013), section 6.1.3.

2.6 Terminology

Multicollinearity

This means the issue where two or more explanatory variables in a regression are highly correlated. When this is the case, the interpretation of the coefficients becomes problematic. If two explanatory variables are themselves highly correlated, it becomes hard to say which one actually influences the dependent variable. Speaking in real-world causal terms, we might not be able to tell whether e.g. one explanatory variable influences the second, which in turn is the one that causally influences the dependent variable.

Overfitting

This refers to when a statistical model incorporates noise rather than real patterns in the data. We can check if a model is overfitting by splitting the data into train and test sets, and seeing if it performs markedly worse on the test set. If so, it has incorporated noise from the train set in addition to any real patterns underlying the whole data set.

Confusion matrix

A confusion matrix represents the outcome of a classification model as a 2x2 matrix, showing true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN).

True positive and true negative rates

The raw numbers above are related to these rates in the following way. The true positive rate is: $\frac{TP}{TP+FN}$. Conversely, the true negative rate is: $\frac{TN}{TN+FP}$.

Accuracy

Accuracy is the total true classification rate, i.e. the sum of true positives and true negatives divided by the total number of observations.

One-hot encoding

One-hot encoding is a method used to encode a categorical variable with k categories as $k-1$ binaries. For example, suppose a variable has the categorical values A, B and C (so $k=3$). Then one-hot encoding turns this into two binaries, which together can represent any of the values: A = (0,0), B = (1,0) and C = (1,1).

Hit rate

This means average sales per offer for a given time period, in our case usually weekly.

3 Data explanation and exploration

3.1 Introduction

The data used in this thesis concerns offers of child insurance made to potential customers, some of which later resulted in sales. The data set stretches back to the beginning of 2014, when only the standard product was available. The less expensive economy product offering less coverage was then launched in the middle of June 2018. The data set contains approximately 85,000 observations representing offers made to customers, with variables representing the offer's characteristics. Some of these characteristics relate to the customer, such as age and income, while others relate to the product, such as the type of product or yearly premium. In addition, each observation contains the offer date and, if the offer resulted in a successful sale, also the associated sales date.

The customer information in each observation can be divided into two large categories: essential characteristics, such as age and income, which all customers must enter before being presented with an offer, and characteristics which the insurance company receives about the customer from a third party. This third party data appears to be estimated to a large degree, so it is less accurate than the data the customers directly enter about themselves. For this reason, we separate these two categories, which we will henceforth refer to as the "internal" and "external" data, respectively. In addition, this external data has only been gathered for roughly half the customers. Hence we feel it is useful for analytical purposes to treat these data as quite distinct from the high-quality internal data.

We will presently give a more detailed explanation of each variable in the data set, starting with the internal variables. The following tables will show whether the variable is continuous or categorical, and display some summary statistics for the continuous variables.

3.1.2 Explanation of variables in data set

Internal variable	Type	Explanation	Mean	Std. dev.	1. quartile	3. quartile
CustomerAge	Cont.	Customer age	35.6	7.2	30.0	40.0
CustomerIncomme	Cont.	Customer income	572,252.1	504,894.3	410,000.0	642,687.0
CustomerDebt	Cont.	Customer debt	1,463,428	2,063,155	-1	2.800,000
CustomerNrOf OtherLifeProducts	Cont.	Life insurance products the customer has with the company	0.7	1.0	0.0	1.0
CustomerNrOf OtherP&CProducts	Cont.	Property/car insurance products the customer has with the company	2.0	2.8	0.0	4.0
CustomerNrOf MonthsActive[11]	Cont.	Months active with the company	336.9	939.9	10.0	57.0
InsuredChildAge	Cont.	Insured child's age	5.2	4.9	1.0	9.0
TypeOfChildInsurance	Cat.	Standard or economy type				
OfferDate	Cat.	Date when the customer received the offer				
SalesDate	Cat.	Date the customer accepted the offer (blank if not accepted)				
CustomerSex	Cat.	Male or female				
CustomerCountyCode	Cat.	Norwegian county code				
CustomerEducation	Cat.	Primary, high school, university (bachelor), and master's degree or higher				
CustomerOccupation	Cat.	Score of occupational risk based on job				
InsuredChildSex	Cat.	Male or female				
YearlyPremium	Cat.	Two different prices corresponding to the standard and economy type				
Salgskanal	Cat.	The channel through which the offer was made (detailed explanation below)				

Table 3.1: Explanation and summary statistics for variables from internal data.

Salgskanal (Norwegian for "Sales channel") merits a more detailed explanation. This explains how the offer was made, and consists of four possibilities: "Eierbank" (owner bank), "Franchise", "Firmaets egne kanaler" (The company's own channels) and "Partner". "Eierbank" means that the offer has been made via banks that are co-owners of the insurance company. "Firmaets egne kanaler" means that the offer was made through the company's own customer center or website. "Franchise" means that the offer was made on behalf of the company by a third party seller, which receives a commission for this. "Partner" means that the offer has been made by a partner bank. Since we cannot rule out differences in the data between sales channels, we deal with this issue in more detail in section 5.2.1, where we discuss how to take this into consideration in our modelling. This is particularly relevant for the "Franchise" category, which involves outbound offers (i.e. offers made directly to the customer by a salesperson). The other channels are inbound, i.e.

[11] We can tell from the summary statistics that something is wrong with *CustomerNrOfMonthsActive*: the standard deviation is for example quite impossible. We deal with this issue in section 5.2.2, on data clean-up.

initiated by interested customers themselves. We hence have an *a priori* suspicion that this sales channel might differ quite significantly from the others.

Now we come to the external data, which the company gets from the third party business intelligence company. We note again that this data is only present for approximately half the offers made. This is one reason why we have decided to treat it separately from the internal data, which is present for virtually all offers. The other reason is that this data is either scraped from public sources or estimated, in contrast to the internal data which is entered by the potential customers themselves.

External variable	**Type**	**Explanation**	**Mean**	**Std. dev.**	**1. quartile**	**3. quartile**
FamilyAssets	Cont.	Household assets	846,346	2,775,415	0	692,500
FamilyIncome	Cont.	Household income	724,328	508,757	440,000	900,000
PersonAssets	Cont.	Customer's assets	356,783	2,021,293	0	130,000
PersonIncome	Cont.	Customer's income	417,026	312,913	260,000	510,000
DisposableShare	Cont.	Percentage of income left after fixed expenses	59.2	22.3	53.0	74.0
HousingExpense	Cont.	Percentage of income spent on housing	4.7	3.2	3.0	6.0
HousingValue	Cont.	Value of house where the customer lives	4,028,692	2,211,392	2,561,289	4,890,907
LivingExpense	Cont.	Percentage of income spent on living expenses	41.7	75.7	19.0	39.0
PurchasePower	Cont.	Savings available for unexpected expenses	24,301.7	24,154.2	11,771.0	31,380.5
EducationYears	Cont.	Total years of education	15.2	1.6	14.0	16.3
ReliabilityScore	Cont.	Probability of getting a payment remark this year	42.4	21.6	25.0	57.0
CarOwner	Cont.	Type of car(s) owned (how expensive)	1.2	1.2	1.0	1.0
Companyrole	Cat.	Whether the customer has a management/board position in a registered company				
Companyboard	Cat.	Whether the customer sits on the board of a company				
SharePayment Remark	Cat.	Probability that the customer has a payment remark (i.e. an ongoing debt collection case, bankruptcy, or similar)				
HousingOwner	Cat.	Whether the customer owns the property where they are living				
OwnsVacationHome	Cat.	Whether the customer owns property used only as a vacation home				
OwnsResidenceType	Cat.	The kind of property that the person owns				
PropertyOwnership	Cat.	Whether the customer owns any kind of property				
TypeOfHousing	Cat.	The kind of property the person is living in				
LifeStage	Cat.	Several categories from youth to senior				
LevelOfEducation	Cat.	Categories from no education to researcher				

Table 3.2: Explanation and summary statistics for variables from external data.

We have, in addition to the raw data, created two new binary variables which indicate whether an offer resulted in a successful sale (is a “hit”) and whether the offer contains external data from the third party data source. These two variables hence do not add any information which was not previously present in the raw data, which is why we mention them here. They simply make it easier to at a glance see these two facts about any given offer. These binaries were created according to a procedure explained in detail in section 5.2.2.

3.1.2 Distribution of missing data

We will now give a brief summary of how missing data is distributed between different variables. There are large differences here, with some variables containing data for absolutely all observations. Other variables lack data for up to 90% of observations. The existence of such gaps, and of such size, is clearly of importance to our future analyses of, and models created using, the data set. We will follow the by now regular distinction between internal and external data, by first presenting the former and commenting on any interesting observations. Tables 3.3 and 3.4 show the percentage of data missing, organized by variable and sorted in increasing order:

Internal data

Internal variable	**Data missing (%)**
CustomerCountyCode	0.1%
CustomerAge	0.2%
InsuredChildAge	0.2%
YearlyPremium	0.5%
CustomerDebt	4.4%
CustomerIncomme	32.0%
CustomerEducation	33.3%
CustomerOccupation	33.3%
SalesDate	60.2%
CustomerNrOfMonthsActive	73.1%

Table 3.3: Percentage of data missing for variables from the company’s internal data.

We see that for the variables up to *CustomerDebt* there is virtually no missing data at all. This is one characteristic of the internal part of the data set. In the case of *CustomerDebt*, this low percentage of strictly missing data is actually deceptive: we are informed by the company that the substantial number of observations for which this variable is 1 or -1 should also be treated as being unknown or missing. This is something we will deal with in more detail in section 5.2.2, where we perform a clean-up of the data before modelling.

We also note that the number of blanks is very similar for *CustomerIncomme*, *CustomerEducation***,** and *CustomerOccupation*. For most observations, these three variables are either all missing or not missing at all. *SalesDate* is a crucial variable here, as the percentage of blanks here expresses the rate of unsuccessful sales (60.2%). Hence the hit rate, the rate of successful sales, is 39.8% for the entire data set. In the case of *CustomerNrOfMonthsActive*, a missing value at least in some cases must express that a customer is new, i.e. has been active for zero months, but here we suspect that this cannot be the case for 73.1% of customers. This is something we will investigate more closely in section 5.2.2, where we will use imputation to fill some anomalous values for this variable.

External data

External variable name	**Data missing (%)**
Companyrole	55.4 %
Companyboard	55.4 %
SharePaymentRemark	55.4 %
HousingOwner	55.4 %
OwnsVacationHome	55.4 %
OwnsResidenceType	55.4 %
PropertyOwnership	55.4 %
TypeOfHousing	55.4 %
LifeStage	55.4 %
LevelOfEducation	55.4 %
ReliabilityScore	55.4 %
PersonAssets	55.5 %
PersonIncome	55.5 %

EducationYears	55.5 %
FamilyAssets	55.6 %
FamilyIncome	55.6 %
DisposableShare	55.6 %
LivingExpense	55.6 %
PurchasePower	55.6 %
HousingExpense	60.8 %
HousingValue	90.2 %

Table 3.4: Percentage of data missing for variables in external data set.

The most striking thing to note is that all but two of these variables (*HousingExpense* and *HousingValue*) contain data in approximately 55,5% of cases. As noted previously, this means that much fewer observations have this external data compared with the internal data variables. Only roughly half of all offers made have this data associated with them. In addition, a rather small proportion of these lack information about *HousingExpense*. This is in contrast to *HousingValue*, which we have information about only in very few cases.

3.2 Discussion of internal data

Now that we have set out the meaning and type of each variable, we will move on to examining the actual shape of the data contained therein. As usual, we will begin by discussing the internal variables. Some of these, as mentioned, contain continuous variables while others are categorical. These will here be treated separately. First, we discuss the continuous variables, then the categorical ones. The order is hence the same as shown in tables 3.1 and 3.2 in the preceding section.

3.2.1 Correlations: plot and discussion

Beginning with the continuous variables, we first present a correlation plot, showing the correlations between each of these:

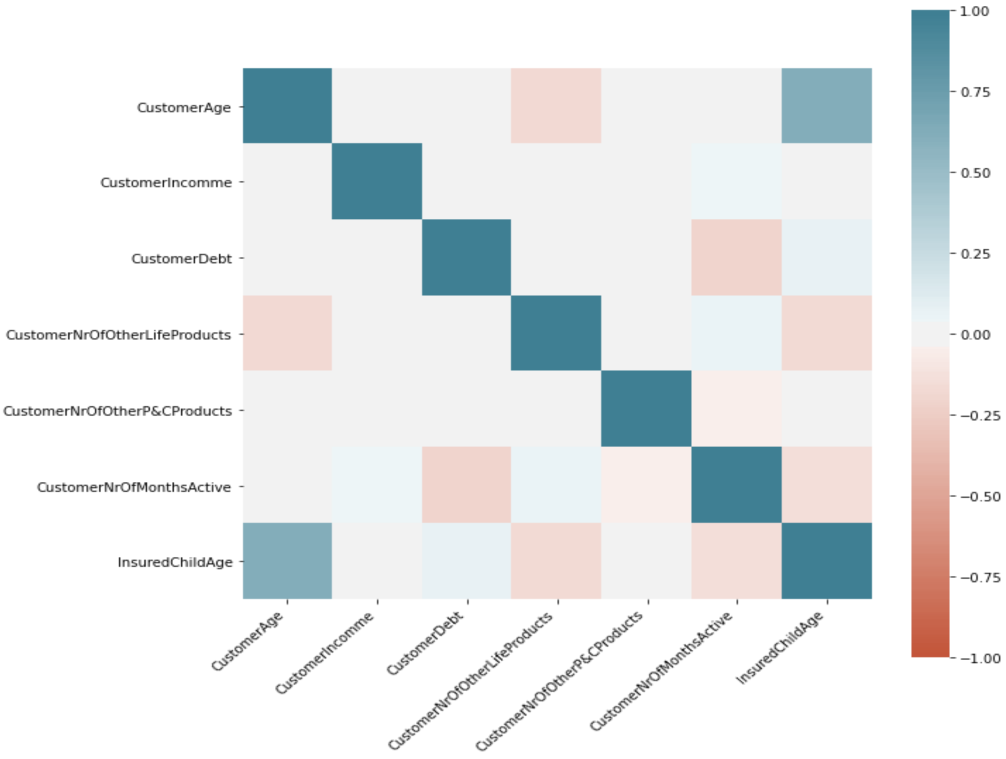


Figure 3.1: Correlation plot for internal data.

We note immediately that many of these continuous variables are not highly correlated. The exceptions, those that do show important correlations, are the following, listed in order of increasing absolute correlation:

Variable 1	**Variable 2**	**Correlation**
CustomerNrOfMonthsActive	*InsuredChildAge*	-0.141
CustomerNrOfOtherLifeProducts	*InsuredChildAge*	-0.168
CustomerAge	*CustomerNrOfOtherLifeProducts*	-0.175
CustomerDebt	*CustomerNrOfMonthsActive*	-0.204
CustomerAge	*InsuredChildAge*	0.616

Table 3.5: List of strongly correlated variables within the company's own data.

Almost all of these are intuitively surprising. For example, it is hard to explain why more indebted customers tend to have been customers for a shorter amount of time. Perhaps highly indebted customers switch around more between competitors. That older customers have purchased fewer life products is also surprising, since one would intuitively assume that one would buy precisely things like child and life insurance as one grows older. At least one of these may be less puzzling than it seems. It is for example hard to explain why a relatively new customer (someone with a low number for *CustomerNrOfMonthsActive*) tends to insure a relatively older child. This does make more sense if viewed the other way around: an older customer (someone with a high number for *CustomerNrOfMonthsActive*) might immediately insure a newly born (i.e. young) child. The high correlation between *CustomerAge* and *InsuredChildAge* is strange indeed, and not something one would intuitively expect. Given the problems associated with multicollinearity, we should therefore not include both of these highly correlated variables in our models. But aside from this pair, all others are below or around the 0.2 correlation that Wilson (2009) deemed acceptable.

3.2.2 Selected histograms for continuous data

We will now show and discuss some histograms made from the same data. Only the most interesting examples have been selected, and the same procedure will be repeated below for the external data.

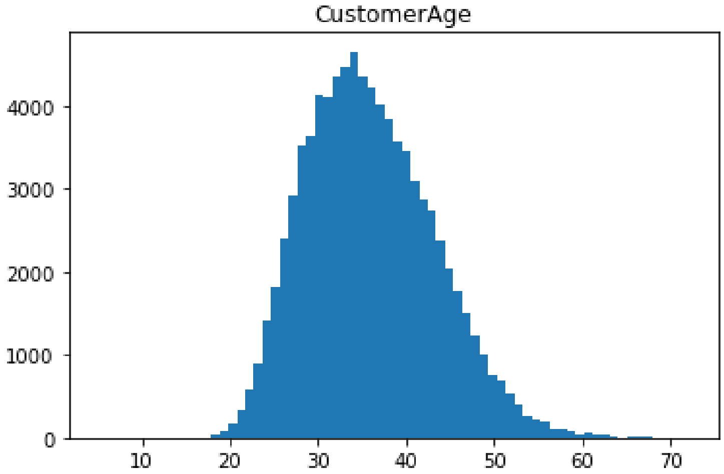


Figure 3.2: Distribution plot for *CustomerAge*.

We here note that the shape of this data very closely resembles a normal distribution. There is a slight skew, but it is almost negligible. This skew is intuitive and unsurprising since there is a clear lower bound for who might purchase child insurance and in practice no upper bound (since e.g. grandparents can buy this for grandchildren or similar situations).

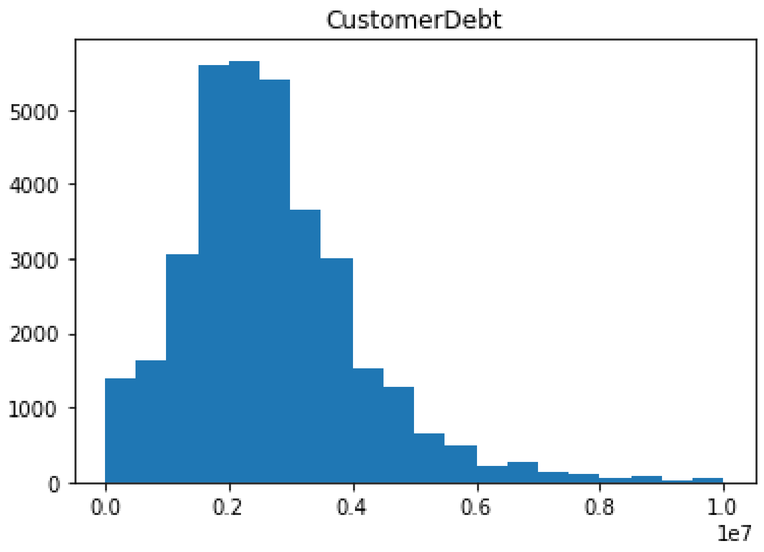


Figure 3.3: Distribution plot for *CustomerDebt*.

As noted when discussing this in section 3.1.2, *CustomerDebt* has many more instances of missing data if we include the values -1 and 1 as such. Hence we have excluded these to show the distribution for the data that is actually present. The figure now appears quite normally distributed, though perhaps somewhat skewed to the left, and naturally containing no values below zero.

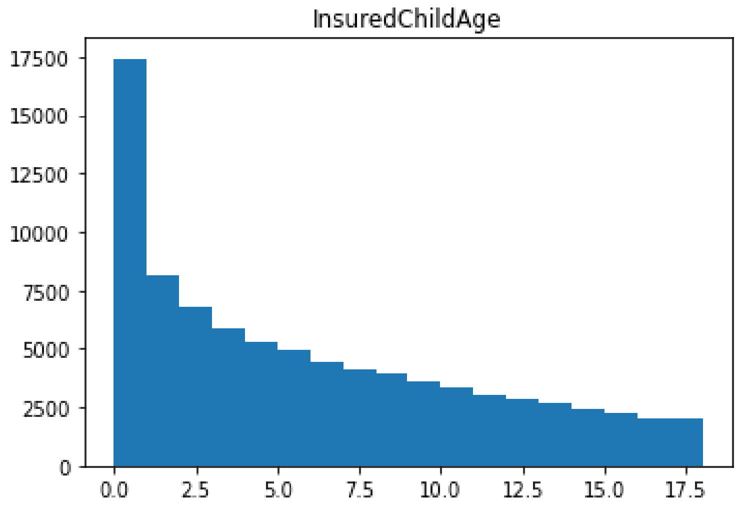


Figure 3.4 Distribution plot for *InsuredChildAge*.

This is one of the less surprising histograms. In shape it strongly resembles a negative exponential distribution. We are not surprised that more children are insured while they are young, and that there is a decline as the children get older.

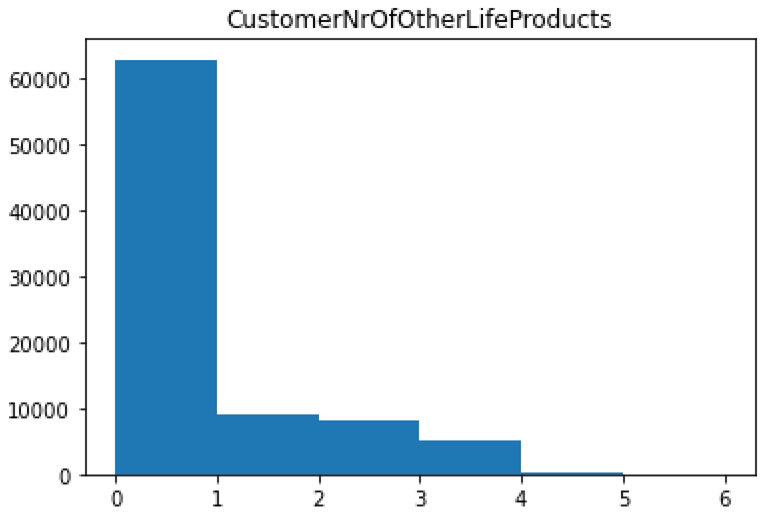


Figure 3.5: Distribution plot for *CustomerNrOfOtherLifeProducts*.

This histogram shows a feature that recurs in the data set, namely that the majority of observations have a value of zero. What this means is that most customers do not already own any other life insurance products when they receive the offer of (i.e. express some interest in) child insurance. This is an important point to note, especially as it will present us with a potential problem when comparing the two different child insurance products. Not only do we here see that most customers own no other life insurance products, but the newer economy product represents an even smaller fraction of sales. For now, we only make a note of this, and defer a fuller discussion to later sections.

3.2.3 Selected bar plots for categorical data

We will here use bar plots to visualize some of the more interesting variables containing categorical data. Again we are here dealing only with the internal data that the company gathers for virtually all sales.

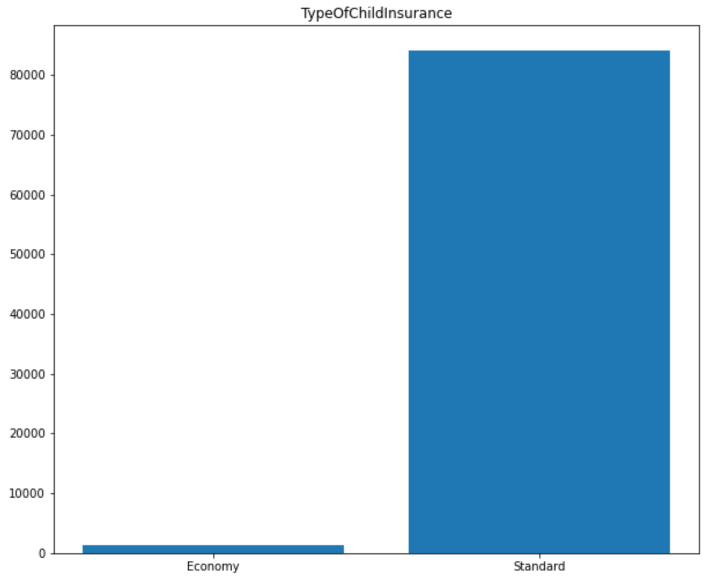


Figure 3.6: Distribution plot for *TypeOfChildInsurance*.

This is the point referred to in the paragraph above, that the number of offers made for the standard product vastly exceeds that of the economy product. Of course, this is to a certain extent to be expected given that we are dealing with offer data going back to the beginning of 2014 for the standard product, while the economy product was launched in the middle of 2018. Yet even for the later periods, the consistent underperformance of the new economy product forms the basic business problem that this thesis aims to illuminate. In section 6 we will study the data set exclusively from the launch of the economy product onwards.

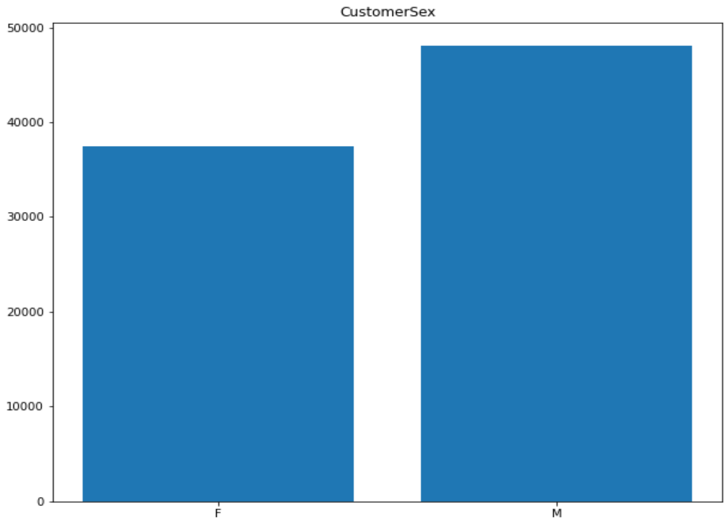


Figure 3.7: Distribution plot for *CustomerSex*.

This plot, though not hard to interpret, is still somewhat unexpected from an intuitive perspective. We see a clear discrepancy between the number of offers provided to men and women. Somewhat less than 40% of offers are made to women. This possibly has something to do with the fact that even today, financial decisions are more likely to be taken by the male in a household (and purchasers of child insurance are naturally somewhat likely to be couples). At this point, we only note this surprising finding, which we will return to in section 5.5.2, when we discuss the results of our first logistic regression model.

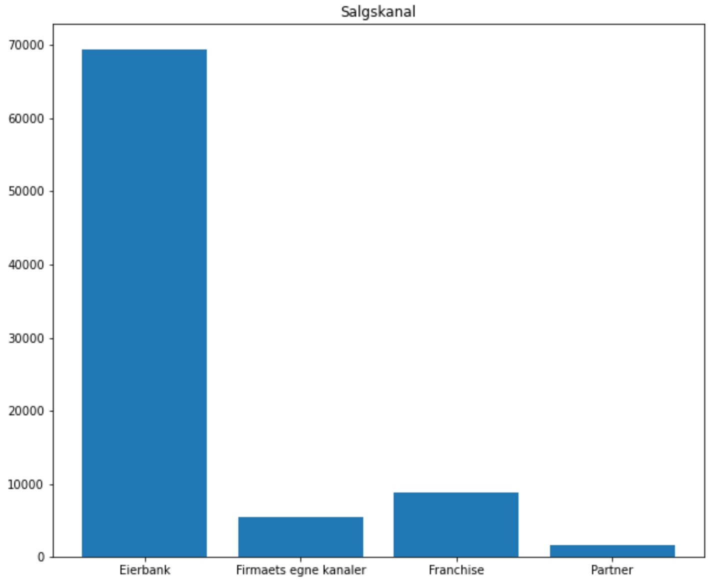


Figure 3.8: Distribution plot for *Salgskanal*.

This relates to the sales channels described previously. We see here that there is a far from even distribution of sales between these: approximately 85% of all offers are made through banks that are co-owners of the insurance company. The second-largest category, containing those made by a third party sales agency that receives a commission for each sale made, accounts for only around 10% of all the data. This unevenness, and potential differences between these subsets, is something we need to take into account in the modelling phase of this thesis. It will be dealt with in more detail in section 5.2.1, before we clean up the data.

3.3 Discussion of external data

We are now going to consider the external data set, which the company gets from a third party business information partner. As previously mentioned, it will become apparent that many of these data are not gathered directly from the individual in question, but instead scraped or estimated based on whichever data is publicly available or gathered by this partner company. In practical terms, this means that the data is less reliable on a person-to-person basis, and as we will see, many of the variables show strong correlations. Some of this probably reflects real-world correlations (e.g. between income and debt), while others are likely the result of the underlying algorithms used by the partner to estimate the customer’s income, life stage etc. Without finely grained personalized information about each customer, the associated profile must necessarily be based on a population average to a large extent, which would explain the high correlations we see below. As above, we first consider the continuous variables and start by looking at a correlation plot.

3.3.1 Correlations: plot and discussion

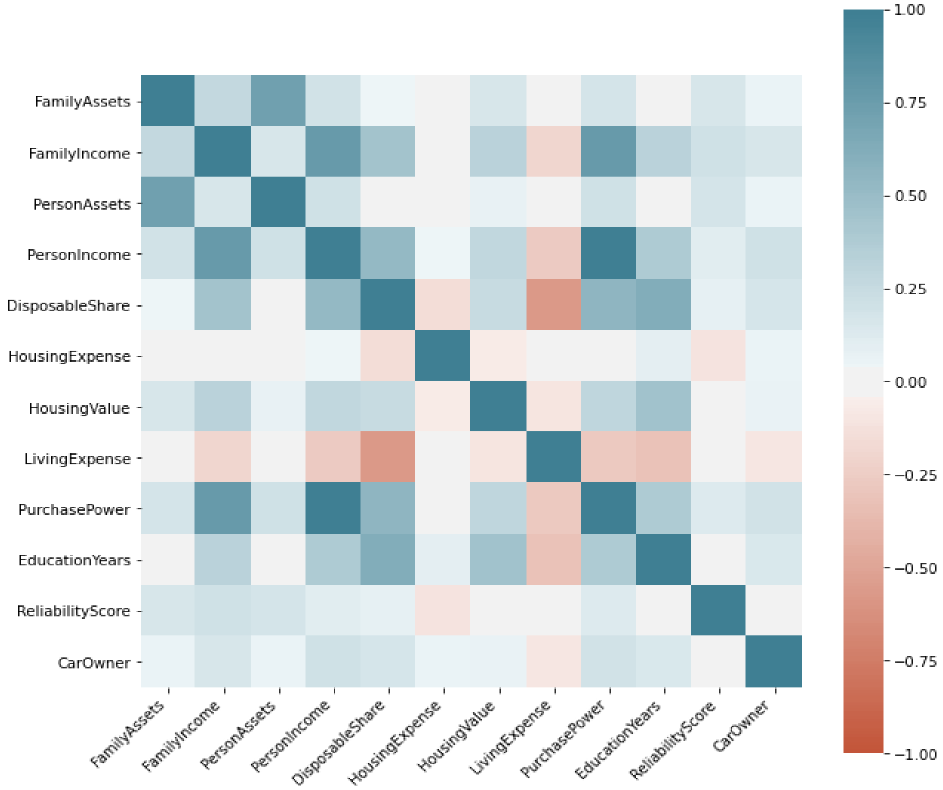


Figure 3.9: Correlation plot for external data.

What we immediately see here is that many more variables are much more strongly correlated than for the internal data, whether negatively or positively. The reason for this is what we have outlined in the previous paragraph, namely that we assume each customer's profile is based on a population average that is then adjusted in various ways depending on what additional information is known about the customer. The existence of such a generic profile would explain why we see such high correlations. We will now list and briefly discuss some of the higher and more interesting correlations.

Variable 1	**Variable 2**	**Correlation**
PersonIncome	*PurchasePower*	0.993
PurchasePower	*FamilyIncome*	0.769
PersonIncome	*FamilyIncome*	0.766
PersonAssets	*FamilyAssets*	0.719
DisposableShare	*EducationYears*	0.622
DisposableShare	*PersonIncome*	0.528
DisposableShare	*LivingExpense*	-0.573

Table 3.6: List of strongly correlated variables within the external source.

The first three rows are an example of precisely this phenomenon. We find that *PersonIncome*, *PurchasePower*, and *FamilyIncome* are all highly correlated, with *DisposableShare* showing a somewhat weaker correlation with *PersonIncome*. While we would intuitively expect to find robust correlations between such variables, a correlation higher than 0.99 leads us to believe the data was generated in the manner described above. For modelling purposes, this leads us to question the usefulness of many of these variables. Including more than one would undoubtedly lead to the problem of multicollinearity. We also find a correlation of comparable strength between *FamilyAssets* and *PersonAssets* as we did between *FamilyIncome* and *PersonIncome*. Again it is plausible and even likely that this reflects a real-world phenomenon, though perhaps to a somewhat lesser extent. *DisposableShare*, which we have already seen has a fairly strong correlation with *PersonIncome*, also has a comparably strong positive correlation with *EducationYears* and negative with *LivingExpense*. Our conclusion here is therefore what we have already noted, that many of the variables in the external data set are very highly correlated among themselves.

3.3.2 Selected histograms for continuous data

As was done above for the internal data, we will now show and discuss some histograms made from this data set. We will pick representative or otherwise interesting examples that illustrate features of the external data.

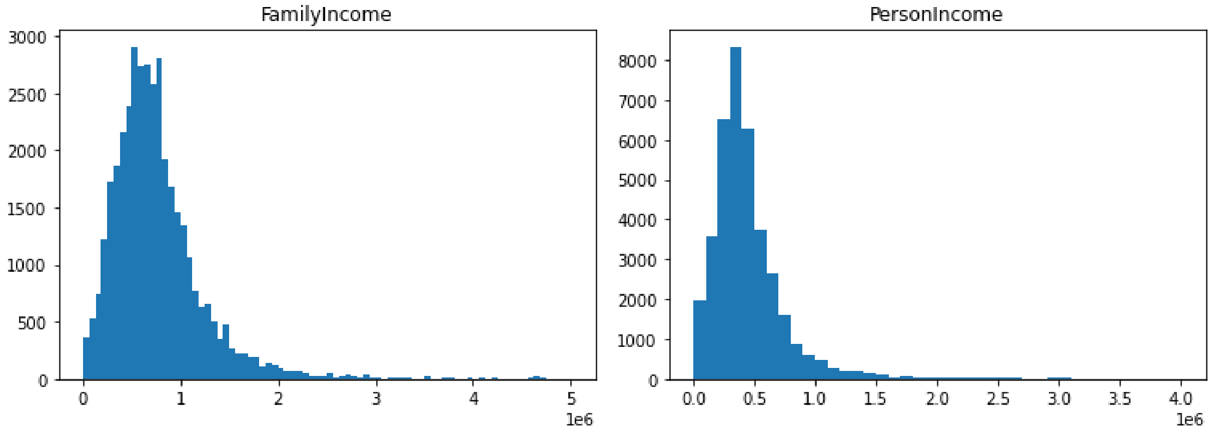


Figure 3.10: Distribution plots for *FamilyIncome* and *PersonIncome*.

We have already noted from the correlation plot that the variable *FamilyIncome* is highly correlated with *PersonIncome*, and, as we might expect, here we see that the two also have distributions that look similar. Both resemble normal distributions that are perhaps to a small extent skewed to the left.

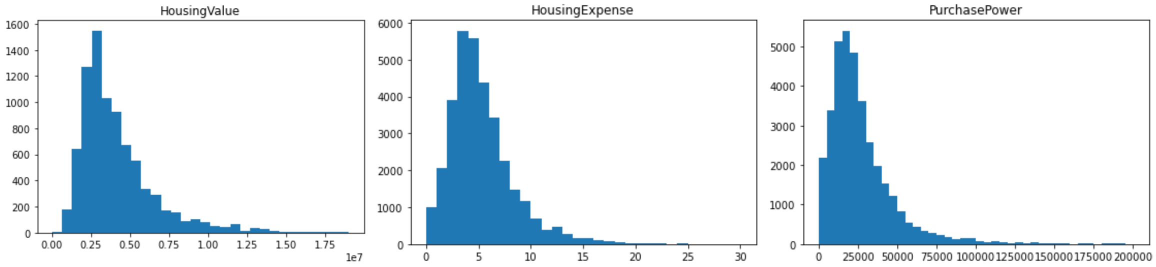


Figure 3.11: Distribution plots for *HousingValue*, *HousingExpense* and *PurchasePower*.

HousingExpense, *HousingValue* and *PurchasePower* also have the same general shape, but are clearly skewed to the left. The point to note here is how the external data is of a different kind than the internal, since it is much less differentiated and indeed for many variables seems to have even a comparable distribution, only scaled differently.

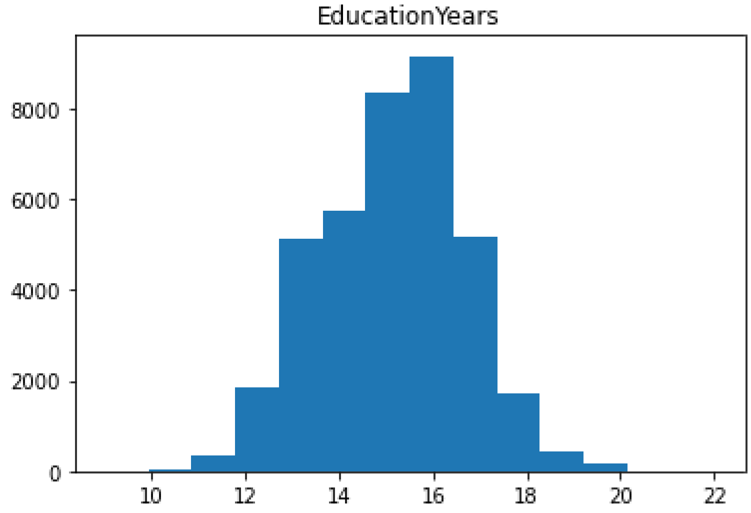


Figure 3.12: Distribution plot for *EducationYears*.

But that is not to say that all the external data have such a shape. For example, we see that the variable *EducationYears* has a normal distribution with no significant skew. This no doubt reflects the real-world variable it is meant to estimate: there are indeed people with zero income, but hardly anyone with zero education. We can also note that even though this histogram visually appears quite different from that of e.g. *PersonIncome* (shown above), there is in fact a 0.391 correlation between them. The fact that the shapes differ somewhat does not mean that there is an especially low correlation. All the correlations discussed in this section are above the 0.2 threshold which Wilson (2009) considered acceptable in terms of not causing significant multicollinearity.

3.3.3 Selected bar plots for categorical data

Now we will visualize the categorical variables in the external data set using bar plots and discuss what we find. As usual, we will select the more interesting variables for discussion, as many present us with few surprises. We begin by noting that, as discussed initially, the external data set is missing for approximately 55% of all entries. This is what is meant by the bar with the caption “nan”, which is therefore present in each of the figures below. An important question which will be discussed in more detail in section 3.4 is whether the roughly 55% of offers containing the external data are randomly distributed, or whether there is some kind of skew towards a certain profile. That question will begin to be raised here.

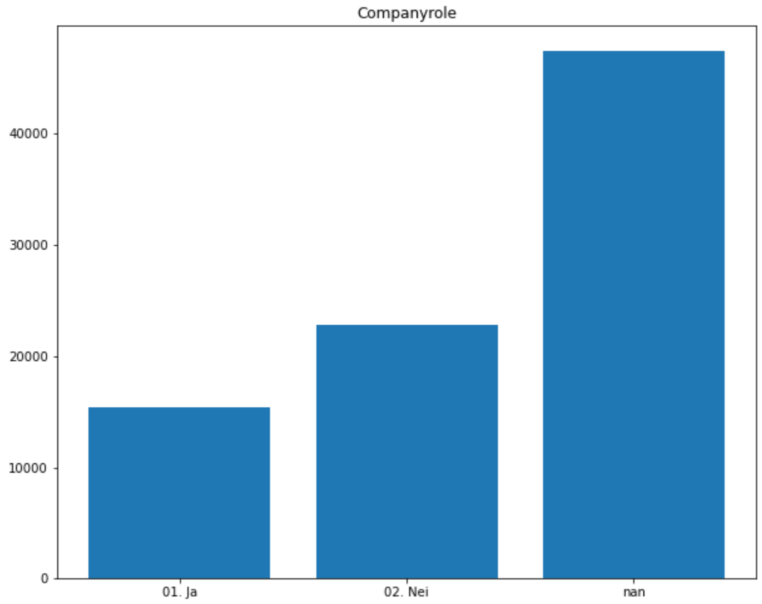


Figure 3.13: Distribution plot for *CompanyRole*.

Already here we are faced with something that is probably not representative of the general population, namely that roughly 40% of the customers in the external data set have a role in a company, i.e. a management or board position. Surely a much smaller proportion of the population actually has such a role. In addition, because such data is publicly available in Norway, this is likely to have been scraped instead of estimated. Hence we are dealing with an actual instance of skewed data, not only the potential for one.

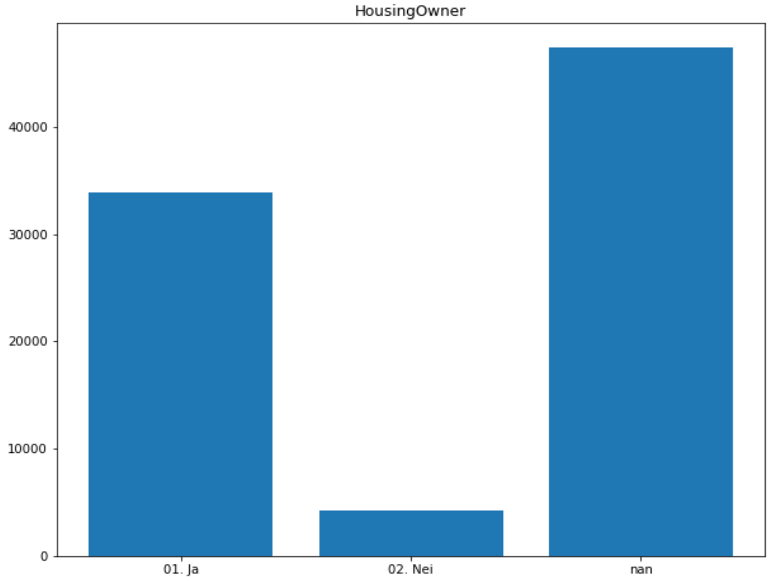


Figure 3.14: Distribution plot for *HousingOwner*.

Here we find that approximately 85% of the customers are also homeowners. This too is based on public records, not on any kind of statistical estimation. This is clearly not representative either, since the home ownership rate in Norway is 76.8%, according to SSB (2020). Hence we are led more firmly to believe that whether or not there is external data about a customer at all can say something significant about the customer. It seems that such data is generally only present when the customer is somewhat more wealthy than the average person.

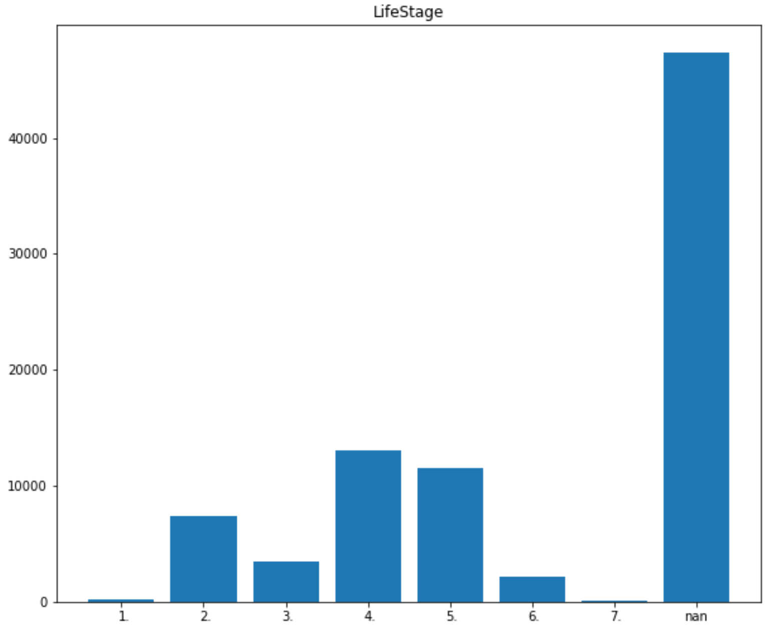


Figure 3.15: Distribution plot for *LifeStage*.

Code	**Meaning**
1	Youth
2	Single
3	Couples without children
4	Families with small children
5	Established families with children
6	Middle aged
7	Elderly

Table 3.7: Meaning of codes for external variable *LifesStage.*

There are a number of things to comment on here. Since we are dealing with offers of child insurance, it is unsurprising that the biggest categories are “Families with small children”, closely followed by “Established families with children”. Here the intelligence firm has correctly gathered or estimated that these are families with children. It is also not especially surprising that the third largest category is “Single”, since more children are now being raised by single parents. What is harder to explain, and which probably indicates that the external intelligence firm is making a mistake, is that the fourth largest category is “Couples without children”. In the entirely plausible cases where somebody pays child insurance for a child that is not theirs, one would intuitively imagine that those making the payment would be older, falling into the category of either “Middle aged” or “Elderly”. It is naturally possible for a childless couple to pay child insurance for the child of somebody else, but here it seems more plausible that we are dealing with a (perhaps unavoidable) error by the firm that provides these external data.

3.4 Analysis of binaries indicating external data and sale

Further examinations of the data set are necessary, given that it is gathered from two different sources in our case. Having data merged from multiple sources may cause several problems during modelling. Selection bias is one such problem, as is well described by Wachinger et al. (2008), and occurs when subjects included in the study are not representative of the aspects of the entire population. In addition, having missing observations is another aspect that potentially causes selection bias. Therefore, given that our data set is both gathered from two different sources and particularly since one source has a significant number of missing observations, the occurrence of selection bias in the data must be investigated before modelling. Another type of bias that might occur while merging data sets from two different sources is information bias. Information bias occurs when there is a mismeasurement in any of the variables (Mehta et al., 2016), or the data is inaccurately collected. In our case, we know that the internal data we gathered is self-reported by the potential customers during the offer phase. However, there is an ambiguity regarding how the external data is collected, as we previously showed that very strong correlation among the variables being represented in this data set makes us suspect that some of the continuous variables are generated by the use of other continuous variables within the data, while some of the categorical variables are almost certainly scraped from third party sources. On the whole, and for these reasons, it is necessary for us to carefully treat the external data in order to prevent systematic errors during modelling.

The most crucial and therefore the first thing that we must investigate is whether those offers that have external data are biased from the total population in terms of hit rate. To investigate this, we create two binary variables, called *Hit* and *External*, which indicate whether the offer resulted in a sale and whether it has data from the external business intelligence firm. This was done according to a procedure explained in more detail in section 5.2.2. After creating these two binary variables, we find that for customers without information gathered by the third party company (i.e. customers for whom the *External* binary is zero) the hit rate is 30.28%, whilst this rate is 51.50% for those customers who do have information from the external source. This makes us suspect that the data set gathered from the external source is biased at a certain level.

In order to show this statistically, we train a simple logistic regression model with those two binaries. This simple model can be defined as; $p(x)/(1-p(x)) = e^{(\beta_0+\beta_1 X_1)}$, where $p(x)/(1-p(x))$ is the odds that an offer is accepted, β_0 is the intercept term, $X_1 = 1$ for the *External* binary, that is if the customer has any information in the data set being gathered from the external source, and $X_1 = 0$ otherwise, and β_1 is the coefficient for the *External* binary. The outcome of this logistic regression suggests that *External* is a significant explanatory variable of the odds that an offer is accepted, given the high z-value of 62.454, and Pearson's $\chi^2 = 8.55e + 04$. We can therefore interpret the outcome as when $X_1 = 1$, the odds of the offer being accepted increases by a factor of $e^{0.8942}$, which equals 2.445. We see that the external binary significantly impacts the odds of an offer resulting in a sale.

As a result of this investigation, we first of all conclude that completely omitting all observations containing external data would cause a bias in the hit rate. We would be leaving out data with a clearly higher hit rate than the remaining data. Therefore, our main target will be making use of all observations during the modelling phase, which will start from section 5 onwards. On top of this conclusion, with respect to the outcome of our investigation, we ended up with two feasible strategies in terms of how to handle the external data while modelling. One would be to completely disregard the data from the external source during modelling. However, as a result of the investigation in this section, we opt to pursue an alternative strategy: we conclude that the *External* binary itself is not only significantly explanatory, but is also information that the company can easily observe retrospectively and gather simultaneously, which makes the use of the *External* binary as an explanatory variable realistic and logical. In other words, rather than using explanatory variables from the external data set for modelling purposes, which would be problematic due to their very high correlations, we conclude that using only whether such external information is available or not for each observation makes sense. This is mainly because it is based on a logical and realistic scenario that the company is able to have the knowledge of, and also because it gives us the advantage of converting missing values to an external binary with a value of zero, so that we will be able to use the absence of information from the external source as itself a piece of information.

4 Time series analysis of hit rate

In this section, we will investigate the temporal development in our data. We will inspect time dependency in the number of weekly offers and sales as well as the weekly hit rate, but we will eventually increase our focus on the hit rate, given that our main target is to assign a probability of purchasing child insurance to each offer. The insights that we gather in this section will constitute a base in terms of how to handle time dependency during the modelling phase and will be discussed in this context in section 5.2.3. Later on, we will take a first step towards evaluating whether introducing the economy product did have an impact on sales or not. To that end, we will try to establish whether there exists a change point in the hit rate, corresponding to the introduction of the new and more affordable product. For this purpose, we will perform a change point analysis.

4.1 Investigation of the time dependency

Having concluded our initial data exploration in section 3, we now look for the existence of time components in our data set. This is important since any sort of systematic and cyclical trend in the dependent and/or any of the explanatory variables results in endogeneity problems[12], so that the estimated effects of the explanatory variables become biased. We inspect weekly offers and sales, considering the fact that the daily data set is too detailed for us to detect any sort of trend or seasonality visually. The number of offers and sales realized per week can be seen in the figure below:

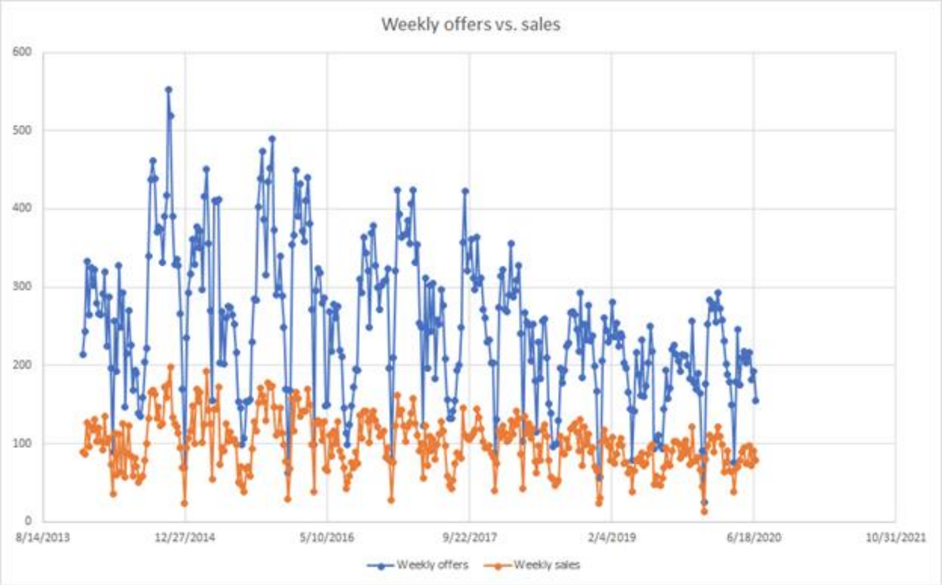


Figure 4.1: The number of offers and sales realized each week.

[12] This means the situation when the error term and an explanatory variable are correlated.

Visually we detect no clear upward or downward trend in weekly offers and sales. However, we must also search for the potential presence of autocorrelation, to see whether time dependency should be taken into account during the modelling phase. Here, we particularly focus on weekly offers and hit rate, as aggregating the data set in weekly terms reduces the variances and gives a better general impression, given that daily offers and hit rates contain some extremes, such as days with zero offers, and therefore zero hit rate, and days with very few offers along with 100% hit rate. Before investigating autocorrelation in weekly hit rate, we perform a visual inspection, similar to what we did for weekly offers and sales. The weekly hit rate figure can be seen below:

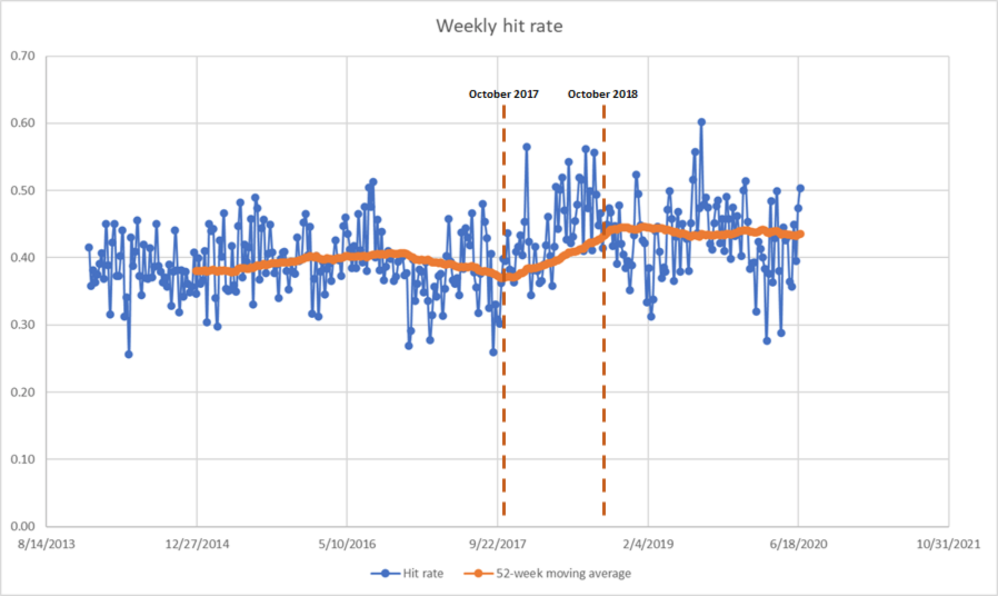


Figure 4.2: Weekly hit rate & 52-week moving average of the week hit rate.

Visually, the weekly hit rate data has some variation over time, however it is not possible to see whether it contains systematic temporal patterns or not. We added the 52-week moving average of weekly hit rates in order to visualize the trend component. It can be seen that weekly hit rates have a roughly flat trend from the beginning of 2014 until October 2017. Following this, there is a one-year period where the weekly hit rate has a trend with a positive slope, that is between October 2017 and October 2018. After this, it seems weekly hit rates again start to show a flat trend from October 2018 onwards. In general, visual inspection suggests there is a weak trend component within the period of the beginning of 2014 and June 2020. Within this section, we use this figure only for the purpose of visually inspecting the occurrence of the time component at any level in the data set. On the whole, these visual inspections lead us to the conclusion that testing the existence of autocorrelation is necessary before the modelling phase. ACF and PACF[13] plots of weekly offers and weekly hit rate can be seen in the below figures:

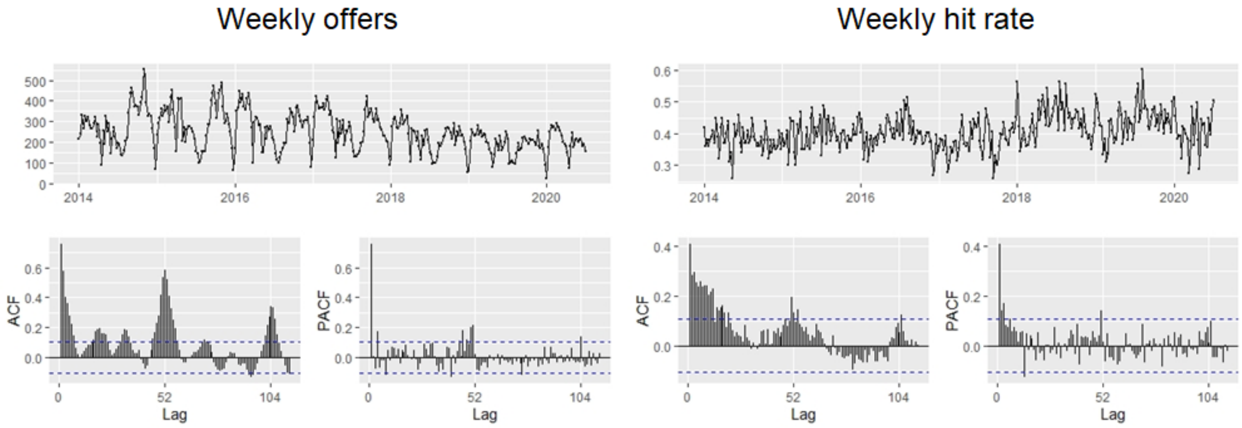


Figure 4.3: ACF and PACF plots of weekly offers and weekly hit rate.

After concluding that there is no significant trend in weekly offers and a weak positive trend in weekly hit rates, these ACF and PACF plots make it apparent that weekly offers and weekly hit rates possess temporal patterns that cannot be ignored, given that both are very significant, particularly at lag 1. In addition to this, there is a certain level of significance at and around lag 52, which leads us to suspect that weekly offers and weekly hit rates possess time dependency in annual terms as well.

Given the fact that particularly weekly hit rate has significant time dependency, we end up considering two main possibilities regarding how to reflect this time component in our modelling. The first is an integrated time series and predictive model, where a predictive model that regards customers' aspects can be built on the top of an autoregressive model. However, we must bear in mind that during modelling, in section 5 and section 6, our main aim will be deriving interpretable predictive models, given that the company's main research question is to find out why sales of the economy product underperformed. Integrating a predictive model on top of an autoregressive time series model would have given a more accurate prediction, but it would at the same time reduce interpretability. In addition, another approach would be building a typical ARMA model for

[13] Introduced and explained in section 2.2.1.

weekly hit rate, but it would be useless in our case given that our main target is not estimating the hit rate for a given period of time. Rather than this, we in section 5 aim to predict the probability of each individual offer resulting in a sale. Therefore, in the modelling phase, we will make use of the time dependency as an extra explanatory variable added to our predictive model, in order not to reduce interpretability, while at the same time controlling for the time dependency of the hit rate for each offer for a given period of time. As will be discussed in section 5.2.3, we will take a number of temporal factors into account, such as what the hit rate was in general in previous weeks and in the corresponding week of the previous year, in the form of additional explanatory variables. In section 6, our problem will be reduced to whether there is a way to build an accurate and interpretable model that predicts the customer's product decisions, that is, whether they opt for the standard or the economy product. We consider the time dependency of the hit rate in this phase as well.

4.2 Change Point Analysis: detecting the impact of the new product

As we discussed in section 1.1, the company has one major research question: why have sales for its economy product performed far below expectations since its launch in July 2018? In this section, we aim to take the first step of examining the potential impact (if any) of launching this new and more affordable product variant on the company's overall child insurance sales. In section 6, we will have a more incisive approach towards this problem, as we are going to investigate the potential reasons why sales of the economy product did not meet expectations with the help of several predictive modelling techniques. But in the present section, we will approach the problem from a time-series perspective only, and the investigation will be limited to whether introducing a new product variant has impacted sales or not. For that purpose, we will make use of change point analysis.

As explained in section 2.2.2, change point analysis has been widely used in various areas to determine whether there is a certain point of time when the mean and/or variance of the dependent variable changes. In other words, it aims to detect a breaking point in time where the dependent variable changes in terms of the abovementioned aspects. In our case, we will use change point analysis to determine whether there exists a change in the mean and variance of the weekly hit rate at any point of time. If there is any, we will compare this point in time with the launch date of the new product variant, in order to detect whether that impact may indeed stem from it or not.

Recalling the information from Figure 2.2, we know that the 52-week moving average of the weekly hit rate has a positive slope from October 2017 to October 2018. Considering that the company launched its new economy product variant in late July 2018, we have a strong intuitive suspicion that the increasing trend in the hit rate is independent of the introduction of the new product. This is simply because the upward trend had already started by the time that the new product was introduced. It should also be taken into account that we would not expect a point of change in the hit rate to be found at exactly the same time as the introduction of the economy product, given that for a period of time customers will have limited knowledge of this product. We further investigate this by making use of change point analysis on the entire data set. Afterwards, in order to see whether the introduction of the new product really matters or not, we use the same technique on the weekly hit rate of the standard product only. The comparison of these two will allow us to come to a conclusion. The outcome of the change point analysis in these two cases, i.e. when the entire data set is taken into consideration and when only the sales of the standard product are considered, can be seen below:

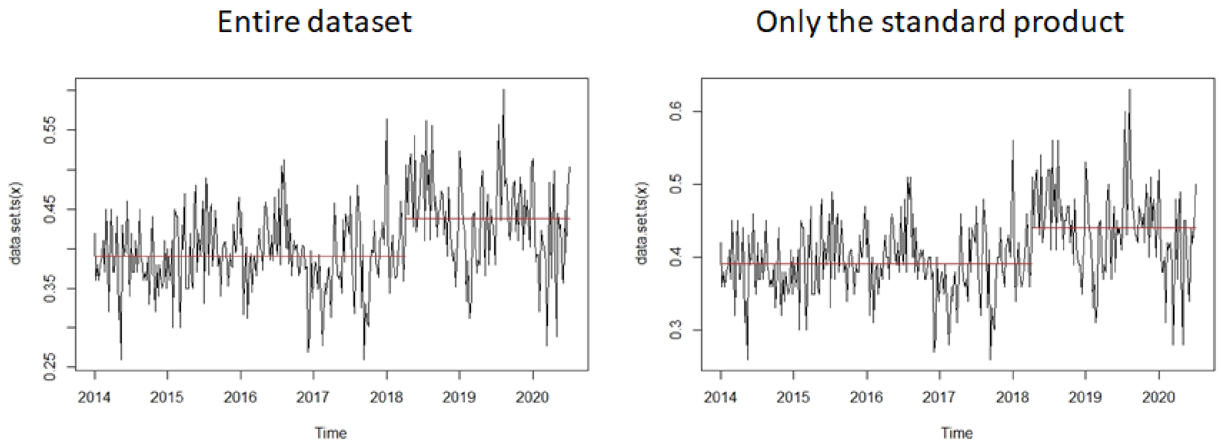


Figure 4.4: Change point analysis figure. Change point in the weekly hit rate when the entire data set is taken into consideration (Left figure) vs. when only the sales of the standard product are taken into account (right figure).

To elaborate, the figure on the left-hand side suggests a change point in the mean and variance of the weekly hit rate that corresponds to the week of 29th of March, 2018. The mean hit rate is 39% before this week, while it is 43.8% from the week of 29th of March, 2018 onwards. Our initial finding of the change point analysis therefore suggests that we certainly cannot argue that a significant change in the weekly hit rate has taken place *after* the introduction of the new product, since the change point identified in fact comes before the new product was launched. The change that we captured must hence have stemmed from something else. In order to double-check, we have run the same analysis on a data set containing only the sales of the standard product, whose outcome can be seen on the right-hand side of the above figure. The outcome shows no change in the change point, while a very marginal increase (43.8% to 44%) in the mean weekly hit rate from the week of March 29th, 2018 onwards. This leads us to conclude that introducing the new and more affordable economy product variant indeed made no impact on the weekly hit rate of the child insurance sales of the company, and statistically confirms the claim of the company. In section 6, we will further investigate this issue by making use of a number of predictive modelling techniques in order to find out potential reasons why the introduction of the economy product did not make the expected impact on sales.

5 Logistic regression model to predict sales

5.1 Introduction

In this section, we will build an interpretable predictive model to take the first step towards finding answers to the company's research question. Compared to more advanced statistical learning models, the model that will be presented below, the logistic regression model, is well suited for investigating *why* customers opt for child insurance. Other modelling techniques such as Random Forest and XGBoost were considered but disregarded for the purposes of this section, due to their hard-to-interpret nature. Classification trees could have been an option in this case as these are indeed easy to interpret. However, since this method is very vulnerable to marginal changes in the data, in other words, since small changes cause significant differences in the decision tree, we discarded this option as well. Considering that we are faced with a binary classification problem, we end up building a logistic regression model, which will assign a probability that a given child insurance offer results in a sale, as explained in section 2.3.2. Thanks to the interpretable nature of logistic regression, we will then be able to show in section 5.5 how each variable affects the probability of an offer being successful.

We discarded some possible explanatory variables for a number of reasons. Some were disregarded as they do not serve the company's research question, such as *TypeOfChildInsurance*, *OfferDate*, *SalesDate* and *YearlyPremium* (which in practice only takes two values, since discounts and other forms of price discrimination are illegal, as noted in section 1.3). We also disregarded *InsuredChildAge*, since we noticed in section 3.2.1 that it is highly correlated with *CustomerAge*. Including both would hence have led to the problems associated with multicollinearity, outlined in section 2.6. We instead opt for including *CustomerAge* as an explanatory variable candidate since it is normally distributed, unlike *InsuredChildAge*. This is something we noted in section 3.2.2. In addition, based on the analysis of the *External* binary performed in section 3.4, we come to the conclusion that it makes more sense to include this as an explanatory variable rather than including some of the highly correlated external data variables themselves. Doing so would again have presented issues of multicollinearity.

Before we move on to creating the model, we will first clarify how we cleaned up the data, explained in section 5.2 which immediately follows this introduction. This involves explaining the following: which observations are deleted, our process of imputation (how we filled in missing data), what other binaries are used including the target variable and how we performed one-hot encoding, outlined in section 2.6.

For variable selection, we pursued a backward stepwise subset selection strategy using the AIC criterion, as explained in sections 2.5.2 and 2.5.3. We chose this approach because it has a computationally feasible algorithm. Given that after one-hot encoding[14] the categorical variables we end up with 30 possible explanatory variables, it would not be feasible to check all possible candidate models[15]. After finding the best-fitted model, we split our data into train and test sets in order to detect whether the best-fitted model is overfitting or not. During the splitting phase, we pursued a different imputation strategy in order not to cause any sort of information leakage from training data to test data, or vice versa. We will discuss this in section 5.4, where we validate the model. Then, along with the results of the cross-validation process, we will discuss the model's results in terms of both overall accuracy and the confusion matrix. Finally, in section 5.5, we will discuss how to interpret the best-fitted model and which conclusions can be drawn from it.

5.2 Data preparation

5.2.1 Sales channels issue

We here discuss how to treat the fact that the offers have been made through four different sales channels. We already noted in section 3.1.2 that these *a priori* cannot be assumed to be comparable, since e.g. banks that are co-owners of the company, the largest of the sales channels, may have a customer base with a profile different from those who access the company's website directly. In particular, we would expect the sales channel called "Franchise" to potentially be quite different, since these offers are made **to** the customer by a salesperson, while the offers in the other categories are made **by** the customers themselves. What we in practice find is that the data contained in each sales channel is comparable, though not identical. The customers do not generally have radically different profiles. We therefore want to find a middle path between either making separate models for each sales channel or, on the other hand, trying to normalize the data between them. Since we

[14] Introduced and explained in section 2.6.

[15] Doing so would require us to fit 2^{30} models, as explained in section 2.5.2.

want the final model to be as general and inclusive as possible, we also decide it would be counterproductive to exclude the data from certain sales channels.

Our approach to this issue will therefore be to use one-hot encoding for the sales channels. As outlined in section 2.6, this means creating dummy binary variables that indicate whether or not each given offer has been made through each sales channel. Naturally, each offer can only be made through one of the sales channels, so each offer will have only one of these binaries set to 1, while the rest will be set to 0. The advantage in using this approach is that we leave the raw data itself untouched, so that we neither introduce any mistakes or biases to it, while also not leaving out the information about sales channels, which we believe might be valuable. Hence, if a certain sales channel has an impact on the probability of an offer being a hit, this should be reflected in the size and significance of the associated effect in the model. This last point will naturally be explained in further detail when we are discussing the final results of the model, in section 5.5 below.

5.2.2 Clean-up of selected variables

Here we will explain the various ways in which we processed the raw data before training a logistic regression model on it. The procedures can be organized into four categories: removing observations, filling in missing or erroneous values with estimations (imputation), creation of binary variables, and one-hot encoding. Each of these will be introduced and explained in more detail shortly.

We have three main purposes when performing such a clean-up. The first is to remove or fix data which is nonsensical, which should be self-explanatory. The second and third involve filling in missing data and encoding categorical data as numerical values. We will now explain each operation in more detail.

Removing observations

This procedure involves dropping whole observations, i.e. all data about a specific offer. We only do this in one case, namely when *CustomerAge* is either blank, 5 or 6. The latter two are clearly errors, presumably where the customer mistakenly has entered the child’s age instead. The point here is hence both to remove missing data and to eliminate nonsensical values. Normally, we would avoid dropping whole observations since this involves the deletion of more data than the missing or problematic value of the variable in question, in this case *CustomerAge*. But we make an exception here, since only a very small proportion of these values are blank, 5 or 6: this involves only 145 offers out of a total of 85,489, or 0.17%. The loss of a minuscule amount of potentially valuable data for other explanatory variables here seems justified since this ensures that *CustomerAge* now will be entirely filled with user-entered data. We hence make a trade-off where we sacrifice some data to avoid having to impute the missing or erroneous values.

Imputation

For other variables, the trade-off involved in dropping whole observations that contain missing or erroneous data becomes unreasonably expensive in terms of the data that would be sacrificed. In such cases it is preferable to estimate what the missing values are likely to be, rather than to delete entire observations. This is imputation, as introduced in section 2.5.1. We now explain how we implemented this.

We have previously remarked on the high proportion of blank values found in *CustomerIncome* and *CustomerNrOfMonthsActive*, in section 3.1.2. For the latter, the number reaches 73.1% (reflecting the fact that many offers are made to potential rather than current customers). For *CustomerDebt* the number of blanks is only 4.4%, but it rises dramatically to 58% if we count values of 1 and -1 as meaning the same as a blank, i.e. unknown. The company informed us that these values should in fact be given such an interpretation; hence we will treat them as blanks. It is clearly not a viable option to discard half of our data set in these cases, as we could do with *CustomerAge* above.

For *CustomerIncome* and *CustomerDebt* we apply the computationally simple method of mean substitution imputation. The imputed means used to fill the blanks are the following:

Variable	**Mean of existing values**
CustomerIncomme	572,111
CustomerDebt	2,697,883

Table 5.1: Mean values of CustomerIncomme and CustomerDebt.

For *CustomerNrOfMonthsActive* the situation is somewhat different. Here we are dealing with both blanks and a value which is clearly a mistake. We noted already in section 3.1.2 that something seemed to be wrong with this variable. The problem is that for 5.6% of offers

CustomerNrOfMonthsActive has a value of 3213, i.e. around 268 years. Luckily, we are able to make some inferences about both the blanks and this anomalous value based on both *CustomerNrOfOtherLifeProducts* and *CustomerNrOfOtherP&CProducts*. We find that when *CustomerNrOfMonthsActive* is blank, both of these were always zero. Hence we conclude that a blank value for this variable really means zero, i.e. that the potential customer in question is a new customer[16]. For this reason, we fill all blank values in *CustomerNrOfMonthsActive* with zero.

In the case of 3213, we combine two approaches. First, we try to find out whether, as with the blanks above, we can establish if these at least are current customers. By inspecting the variables *CustomerNrOfOtherLifeProducts* and *CustomerNrOfOtherP&CProducts* we find that this is indeed the case. Whenever *CustomerNrOfMonthsActive* is 3213, the customer in question already has one or more active products with the company. Hence we can conclude that 3213 cannot correspond to 0. We therefore use the same method of imputation as before, namely mean substitution, but with one crucial difference. In this case, we know that these are current customers, and hence calculate the mean for current customers (i.e. all existing values of *CustomerNrOfMonthsActive* excluding 0 and 3213). This means that we, in the end, fill those values that previously contained 3213 with a newly calculated mean for existing customers of 43.27. Given that this involves changing roughly 5.5% of the values to such a drastic extent (from 3213 to 43.27), it is clear that not dealing with these anomalous 3213 values would have significantly skewed the mean and standard deviation of this variable.

Encoding of categorical to numerical binaries

As described in section 2.3.2, a logistic regression model uses numerical input values, while several binary variables in the raw data set are encoded as text. Hence we convert these into numerical binaries, i.e. 0 or 1. For two variables, *CustomerSex* and *InsuredChildSex* the binary distinction is quite arbitrary: we simply define male as 1 and female as 0.

Addition of new binary variables

The next step is creating the two binary variables *External* and *Hit*. In both cases, a value of 0 indicates absence and 1 presence. *External* hence refers to whether or not the observation includes

[16] There are in fact eight occurrences of 0 as the value of *CustomerNrOfMonthsActive*, which slightly undermines this line of argument. But this is a vanishingly low proportion, and we feel quite confident in our interpretation given the existence of the clearly erroneous value of 3213 with a much higher proportion.

information from the third party business intelligence company about the customer, which we in section 3.1.2 mentioned was the case for only about 50% of offers. *Hit* indicates whether or not the offer eventually resulted in a sale.

These binary variables were created according to the same method. For *Hit* we know that a sale has occurred if the *SalesDate* associated with the offer is filled. Hence *Hit* is 1 if this is the case, and 0 otherwise. This allows us to convert the information in *SalesDate*, which is either a date or blank, into a binary variable without any blanks. *External* is filled in according to an analogous procedure: here we discover through inspection that if *Companyrole* is filled then virtually all the other external variables are as well, and, crucially, in no case was any other external variable filled while *Companyrole* was not. Hence the presence or absence of a value for *Companyrole* works as a proxy for whether or not any external data is present for an offer. This information is then recoded as the binary variable *External*.

One-hot encoding

This method, as outlined in section 2.6, allows categorical variables with more than two possible values to be encoded numerically as a set of binaries. Essentially, it encodes each possible value within each categorical variable as a new dummy binary. This was done for the following variables: *CountyCode*, *SalesChannels* and *CustomerEducation*.

In the case of *CountyCode***,** this means that we will be able to see from the final model how each different county code affects the probability of a hit, if at all. We have already discussed in section 5.2.1 why we decided that this was the best method for dealing with *SalesChannels* and the potential differences between them.

The case of *CustomerEducation*, with categories A, B, C and D, must be discussed in some more detail. For this variable, 33% of values are blank, a far too large proportion for us to simply drop these observations. Nor can we use mean substitution imputation: since this is categorical data, no mean exists. It would theoretically be possible to assign numerical values to each category, and calculate a mean from these, but it would be hard to do this in a non-arbitrary fashion[17]. Hence we decide simply to one-hot encode the blanks as a fifth category, meaning that data about education

[17] For example, is a bachelor degree twice as much “education” as going to high school? Three times more? We avoided this approach due to the many such pitfalls involved.

is unknown. It seems reasonable not to rule out that such a refusal to state one's education status could hold valuable information. So we decide to include this and see what the effect is, if any.

5.2.3 Controlling for time dependency

As we elaborated in section 4.1, our data set has significant time dependency in terms of weekly hit rate. The ACF plot in Figure 4.3 shows significant spikes particularly at lag 1 and 52. As stated in section 4.1, we want to create an interpretable model that is relevant to the company's research question and therefore disregard integrated autoregressive and ARMA models as possible modelling techniques. Another very important reason for eliminating heavy time series modelling is that each observation is a unique offer, most of which share offer date with at least some other offers. We eventually want to predict the probability of each offer resulting in a sale based on the unique characteristics of that offer, not the probability of a product being purchased as a function of time. Hence we create three candidate explanatory variables to control the time dependency in the model. In this way, we will be able to control the time dependency to a certain extent, while still enjoying the advantage of the interpretable nature of the logistic regression model.

For each observation (i.e. offer in our case), we create a lag 1 weekly hit rate variable named *Avg1WkLag*. This variable indicates what the average hit rate was during the last seven days for any given offer. To give a brief example, for an offer that is made on the 8th of January, *Avg1WkLag* is the average hit rate from 1st to 7th of January. So for any offer on any given day this variable answers the question *"What was the average hit rate for the last week?"*. In addition to this, we created another explanatory variable candidate named *Avg4WkLag*, which is created by following the same logic that we did for *Avg1WkLag*, but taking the last 28 days into account. Finally, in order to control for the annual temporal pattern, we create a final explanatory variable candidate named *ThreeKNN1YrLag*, which is calculated by the average of 3 K-nearest neighborhood in daily terms with a lag of 52 weeks. To give an example, for an offer that is realized on the 10th of June 2019, *ThreeKNN1YrLag* is calculated for this observation by the average of the hit rates of the 7th through 13th of June 2018. Therefore, this variable stands for the question *"What was the average weekly hit rate on the corresponding week of the previous year?"*.

5.3 Variable selection: backwards stepwise

After creating binary variables out of the categorical and adding three additional explanatory variables to control for time dependency, we end up with a total of 30 explanatory variable candidates. As explained in section 2.5.2, the explanatory variables to be included in the best possible model can theoretically be found using the best subset selection method, which exhaustively tries out every single possibility to find the optimal one. However, if we were to employ this method, we would have to try all 2^{30} possible combinations of explanatory variables. We therefore opt for a more computationally feasible variable selection algorithm, namely the backwards stepwise variable selection method. We choose AIC as the criterion, since it penalises the number of explanatory variables to be used, as explained in section 2.5.3. In other words, it enables the researcher to find a more generalized model with less number of explanatory variables and therefore reduces the risk of overfitting.

The outcome of the backwards stepwise subset selection method suggests that the best-fitted model has 22 explanatory variables, 5 continuous variables along with 17 binaries. We present the equation of the fitted model in section 5.5, where we will discuss and interpret the coefficients. Before presenting these results, we will first, in the following section, investigate whether the model is overfitting or not, by splitting our data set into train and test sets. Along with this, we will be able to test the accuracy of the model on the test set, by treating this as a classification problem.

5.4 Validation of the model

Now that we have selected the relevant explanatory variables and fitted the model based on the full data set, we are faced with a typical problem when dealing with statistical models: is the high accuracy simply the result of overfitting? This is the question we will now try to answer.

To give an answer to this question, we will use a classic model validation approach, involving two steps. The first is to split the full data set into train and test sets. The second is to train the model, using the variables identified above but only using the train data, and then to see how it performs in predicting hits for the test set. This will generally always give a worse accuracy score compared to the model fitted using the full data set, as the degree of overfitting is reduced. This means that if the scores are comparable, i.e. if there is only a small reduction in accuracy when running the model on test data compared to train data, then we can conclude that the model trained using the full data set only involves modest overfitting. That is exactly what we hope to find in this section.

5.4.1 Splitting the data into train and test sets

We here opt for a realistic approach when splitting the data into train and test sets, dividing the two sets according to a chronological cut-off point. The advantage of this is precisely its realism: it mirrors the situation of a company training its models using all available data up to the present and then naturally being interested in the accuracy of its predictions in the future. To simulate such a situation, we simply divide the full data set into two sets, before and after the beginning of 2019. This means that 82% of the data falls in the train set, and the remaining 18% in the test set. Such a proportion of observations between train and test sets is considered reasonable[18], and this date gives us a clear and round cut-off point.

One further set of actions was necessary before re-training our model. As described in section 5.2.2 above, we used mean substitution imputation to fill the missing data for several continuous variables. Since we now have two separate data sets, and more importantly, since there should be no information leakage between train and test sets, these were recalculated and added separately. For example, while we previously calculated one mean using the whole data set for *CustomerDebt*, we now calculate two separate means, one for each data set. This was repeated for each imputed value, as described above.

5.4.2 Re-training of model and evaluation

We now re-train the model using this training data, and evaluate how it performs using the test set. Since we are here mostly interested in determining the degree of overfitting, we leave the exact coefficients of this new model to the end, in **Appendix A**. The results are as follows:

Data set	**Accuracy**
Full data	0.6574
Train set	0.6575
Test set	0.6504

Table 5.2: Model accuracies with respect to the data set used.

[18] There are several approaches regarding how train-test ratio should be in various research fields. To exemplify, Pawluszek-Filipiak et al. (2020) concluded in their study on automatic landslide detection that train data should be as large as test data. Mohanty et al. (2016) tried several train-test ratios in their study about image-based plant disease detection and reached robust accuracies at the train-test ratio of 80-20. Thanks to such examples, we concluded that a train-test ratio of 82-18 is reasonable.

The accuracy is astoundingly similar between the models trained using the full data set and with the train subset. We would expect comparable results, since logistic models are not highly flexible and the training data is 80% of the full data set, but such a degree of similarity is indeed surprising. Furthermore, and fortunately for our initial model, the accuracy is only slightly worse for the test set. The difference in accuracy between train and test set accuracy is less than 0.7%, which indicates a minimal degree of overfitting. We have therefore established that the original model trained on the full data set is very likely to have discovered important relationships within the data, which are stable and hence generalizable also for new data about future customers. That means we can now discuss the findings from the model trained on the full data set, having now established that they involve minimal overfitting.

5.5 Discussion of final model and findings

5.5.1 Presentation of the best-fitted model & evaluation of accuracy

Before discussing the accuracy of the best-fitted model, we will here briefly introduce it in equation form, in order to give a general impression of the explanatory variables, of which interpretations will be discussed in more details in section 5.5.3. (For convenience, the linearized equation is given, where the target variable is the log of the odds.)

$$\log\left(\frac{p(Hit)}{1-p(Hit)}\right)$$

$$\begin{aligned}
&= 0.0346 - 0.02788\text{CustomerAge} - 0.3375\text{CustomerSex} \\
&- 0.00000004401\text{CustomerDebt} + 0.1928\text{CustomerNrOfOtherLifeProducts} \\
&+ 0.00866\text{CustomerNrOfMonthsActive} + 0.7976\text{External} \\
&+ 0.2791\text{Avg1WkLag} + 0.1537\text{CustomerCountyCode\_11.0} \\
&+ 0.1875\text{CustomerCountyCode\_15.0} + 0.257\text{CustomerCountyCode\_18.0} \\
&+ 0.2671\text{CustomerCountyCode\_30.0} + 0.4383\text{CustomerCountyCode\_34.0} \\
&+ 0.3839\text{CustomerCountyCode\_38.0} + 0.137\text{CustomerCountyCode\_42.0} \\
&+ 0.3244\text{CustomerCountyCode\_50.0} + 0.1413\text{CustomerCountyCode\_54.0} \\
&+ 0.1357\text{CustomerEducation\_A} + 0.07144\text{CustomerEducation\_B} \\
&- 0.09229\text{CustomerEducation\_C} + 0.608\text{Salgskanal\_Franchise} \\
&+ 0.09355\text{Salgskanal\_Firmaets\_egne\_kanaler} - 0.214\text{Salgskanal\_Partner}
\end{aligned}$$

Now, we present a confusion matrix that gives us a more detailed breakdown of what exactly makes up the accuracy score. This shows us how well the model performs in classifying an offer as resulting in either a sale or not. The results are as follows:

Predicted \ Actual	**0**	**1**
0	42245	20127
1	9115	13857

Table 5.3: Confusion matrix of the model (For the entire data set).

The previously reported accuracy score of 0.6574 is calculated by adding the correct classifications (true positives and true negatives, i.e. actual non-sales classified as such and actual sales classified as such) and dividing this by the total number of observations: (42245 + 13857) / (42245 + 20127 + 9115 + 13857) = 0.6574. But from this confusion matrix we can also calculate other interesting metrics. For example, from a business perspective, one might also be interested in the false positive and false negative rates, i.e. the share of offers *wrongly* classified as sales, and the share of offers wrongly classified as *not* being sales.

From this perspective, while the overall accuracy score is quite good, maximizing accuracy may not be optimal for the company. The accuracy score assigns equal weights to true positives and true negatives, and we note that most of the correct predictions are true *negatives*, i.e. predictions that a sale will not result. But it is plausible that the company is more interested in correctly identifying probable hits than non-hits. This is why not only the overall accuracy score is a relevant metric, but also the share of true positives. We will now describe how the output of the model can be tweaked to illustrate the trade-off that exists between these two metrics.

A logistic model does not itself produce a classification as an output, but rather the probability that an observation falls into a given class. Concretely, our logistic model does not itself classify whether an offer is a sale or not, but rather gives the probability from 0 to 1 that the offer results in a sale. We hence maximize accuracy by subsequently classifying each offer as a sale or not depending on whether the probability is above or below the "neutral" cut-off probability of 0.5. If the offer is more likely to be a sale, i.e. if the probability of this is over 0.5, then we classify it as a sale, and vice versa. But given that the company might not want *only* to maximize accuracy, it is possible to set this cutoff at a different level. It might make perfect sense to classify an offer as a likely sale even if the actual probability is only 0.4, because it is more valuable to catch more true positives this way, even if it also increases the number of false positives and hence decreases the overall accuracy. We illustrate the trade-off in Figure 5.1 below:

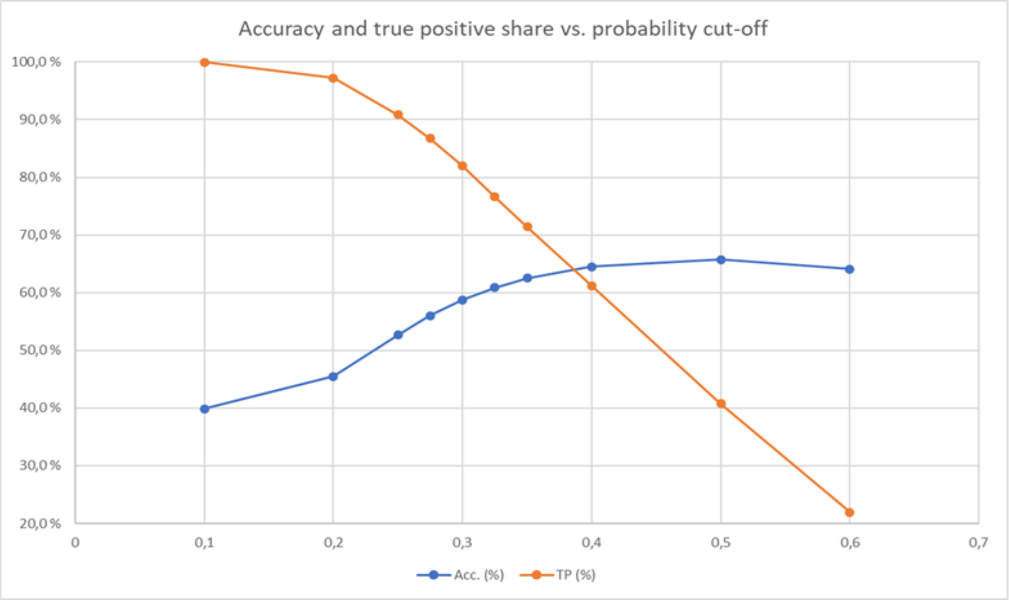


Figure 5.1: Accuracy vs. true positive share (%).

The fact that a trade-off exists is seen in how decreasing the cut-off probability decreases accuracy, but increases the share of true positives. The fact that the slopes are drastically different illustrates another feature: a small sacrifice in accuracy dramatically increases the number of sales correctly identified as such. We present the same results in the table below:

Probability threshold	**Accuracy (%)**	**True positive share (%)**
0.1	39.8 %	100.0 %
0.2	45.5 %	97.2 %
0.25	52.7 %	90.9 %
0.275	56.0 %	86.7 %
0.3	**58.7 %**	**82.0 %**
0.325	**60.8 %**	**76.6 %**
0.35	62.5 %	71.4 %
0.4	64.5 %	61.2 %
0.5	65.7 %	40.8 %
0.6	64.1 %	21.9 %

Table 5.4:Accuracy and true positive share given various classification thresholds.

We have chosen to highlight in bold typeface the results of applying a cut-off probability of 0.3 and 0.325, as these seem to be in the neighborhood that would be optimal in the real world. For the case where the cut-off is set at 0.325, we find that the overall accuracy of the model falls from the optimal level (0.5) by only 4.9 percentage points, but the share of true sales identifies rises by a massive 35.8 percentage points. This means that with the final model configured in this way, 76.6% of offers classified as sales in fact turn out to be so. From a business perspective, this seems highly satisfactory.

5.5.2 Interpretation and conclusion

We now present the results of the model, which was provided in equation for in section 5.5.1, ranked by the significance of each explanatory variable (z value, ordered in absolute terms while keeping the sign in the table):

Explanatory variable	**Coefficient**	**Std. Error**	**z value**	**Pr(>|z|)**
External	0.7976	1.56E-02	51.16	<2e-16
CustomerAge	-0.0279	1.07E-03	-26.04	<2e-16
CustomerSex	-0.3375	1.56E-02	-21.62	<2e-16
Salgskanal_Franchise	0.6080	2.86E-02	21.30	<2e-16
CustomerNrOfMonthsActive	0.0087	5.05E-04	17.15	<2e-16
CustomerNrOfOtherLifeProducts	0.1928	1.21E-02	15.96	<2e-16
CustomerCountyCode_38.0	0.3839	3.96E-02	9.71	<2e-16
CustomerCountyCode_30.0	0.2671	3.06E-02	8.73	<2e-16
CustomerCountyCode_18.0	0.2570	3.01E-02	8.53	<2e-16
CustomerCountyCode_34.0	0.4383	5.62E-02	7.80	6.14E-15
CustomerCountyCode_50.0	0.3244	4.78E-02	6.79	1.13E-11
CustomerDebt (million NOK)	-0.0440	6.50E-09	-6.77	1.31E-11
CustomerCountyCode_42.0	0.1370	2.15E-02	6.37	1.91E-10
CustomerCountyCode_11.0	0.1537	2.62E-02	5.87	4.25E-09
CustomerEducation_C	-0.0923	1.83E-02	-5.03	4.88E-07
Avg1WkLag	0.2791	6.63E-02	4.21	2.56E-05
Salgskanal_Partner	-0.2140	5.68E-02	-3.77	1.66E-04

CustomerEducation_B	0.0714	1.99E-02	3.59	3.32E-04
Salgskanal_Firmaets_egne_kanaler	0.0936	3.20E-02	2.92	3.50E-03
CustomerCountyCode_15.0	0.1875	6.61E-02	2.84	4.53E-03
CustomerEducation_A	0.1357	5.05E-02	2.69	7.15E-03
CustomerCountyCode_54.0	0.1413	6.71E-02	2.10	3.53E-02
(Intercept)	0.0347	4.99E-02	0.69	4.88E-01

Table 5.5: Summary of the model.

The order in which the variables are ranked, as mentioned, reflects how unlikely the relationship between the variable and the hit probability is to be random. In other words, the higher a variable is on the list, the more likely it is to *truly* influence the probability that an offer will result in a sale, regardless of the size of that effect. The variables which we find at the top of the list are a mixed bag in terms of how surprising they are from an intuitive perspective. It does not surprise us that factors such as *CustomerAge*, *CustomerNrOfMonthsActive* and *CustomerNrOfOtherLifeProducts* are high on the list. These are variables which one could *a priori* expect to influence the likelihood of a successful sale or not (as they are associated respectively with higher wealth and being an established customer of the company).

Yet other results are indeed more surprising. By far the most significant factor is the *External* binary. It is crystal clear that whether or not the third party business intelligence provider has information about the potential customer, *regardless of what that information is*, in itself influences the likelihood that an offer will become a sale. While probable reasons for this will be discussed shortly, this result is hardly something that we could have assumed or even expected beforehand. It is also surprising to find that *CustomerSex* makes a difference. Men and women apparently have different inclinations towards accepting the offers they are presented with. Finally, as noted in section 3.1.2, it is unsurprising that the sales channel “Franchise” (indicated by the one-hot encoded dummy variable *Salgskanal_Franchise*) should also make a difference. This is because offers made through this channel, unlike the other three, are out-bound, i.e. not initiated by the customer themselves but by a seller that contacts them. It is precisely because we could not rule out that this would make a difference that we decided to encode the information in this way.

Now we will discuss the effects associated with the most important variables. The effects describe how much the explanatory variable influences the odds of a sale, according to the relationship described in section 2.3.2. In the following sections we will therefore show and discuss the effect on the log odds e^{β_x} rather than the coefficient β_x itself, as this is somewhat more intuitive. When interpreting these effects, we distinguish between binary and continuous variables. The former have a simpler interpretation since they can only have two values. Hence the effect straightforwardly indicates how the odds of an offer being a sale change if the underlying variable is true (such as the presence of external data). Binary explanatory variables also have no units. Both of these facts are different for continuous variables: firstly, they depend on the magnitude of the continuous variable, and secondly they for this reason have a unit. This is why we treat these two types separately. We begin with the binary variables:

Binary explanatory variable	**Effect on the odds of sale**
External	2.2255
Salgskanal_Franchise	1.8404
CustomerCountyCode_34.0	1.5527
CustomerCountyCode_38.0	1.4623
CustomerSex	0.7118
CustomerCountyCode_50.0	1.3771
CustomerCountyCode_30.0	1.3100
CustomerCountyCode_18.0	1.2969
Salgskanal_Partner	0.8106
CustomerCountyCode_15.0	1.2092
CustomerCountyCode_11.0	1.1618
CustomerCountyCode_54.0	1.1503
CustomerCountyCode_42.0	1.1503
CustomerEducation_A	1.1503
Salgskanal_Firmaets_egne_kanaler	1.0942
CustomerEducation_C	0.9139
CustomerEducation_B	1.0725

Table 5.6: Effect on the odds of sale for each categorical explanatory variable.

As we initially discussed, we can see that amongst all other binary variables, *External* seems to have the largest impact on the odds $(p(x)/(1-p(x)))$. This can be interpreted as follows: if the third party business intelligence company has information about the customer who received the offer, then the odds of a sale resulting increases by $e^{0.80}$, which is 2.22. In terms of the magnitude of the impact on the odds, *Salgskanal_Franchise*, which means whether the offer has been made via a franchise or not, follows *External*, as its impact on the odds is 1.836. Following the same logic, the customers who live in the counties encoded as 34 or 38 have a stronger tendency to purchase child insurance than the others. Another interesting result is that the likelihood of buying child insurance differs significantly between male and female customers. Given that we encoded the binary *CustomerSex* as 1 if the customer is male and 0 otherwise, we see that if the customer is male, then the odds of a successful sale tend to decrease by a factor of 0.71. This obviously means that female customers have a higher tendency to purchase the product. Noting this is interesting, as in section 3.2.3 we saw that males were overrepresented among offer recipients. So fewer women receive offers, but the ones that do are apparently more liable to accept them. Regarding sales channels, the fact that a sale has been made through a partner company (*Salgskanal_Partner*) has the most negative impact on the odds. Among all binary variables, those related to the customer's education have relatively lower impact on the odds compared to sales channel and county.

The effect on the odds of sale for each continuous variable can be seen in the below:

Continuous explanatory variable	**Effect on the odds of sale**	**Unit**
CustomerAge	0.9725	years
CustomerDebt	0.9569	million NOK
CustomerNrOfOtherLifeProducts	1.2126	products
CustomerNrOfMonthsActive	1.0087	products
Avg1WkLag	1.3219	sales/offer

Table 5.7: Effect on the odds of sale for each continuous explanatory variable.

The logistic regression model suggests that younger customers are more likely to purchase child insurance, as *CustomerAge* negatively impacts the odds of a sale. The same relationship holds for *CustomerDebt*, which is not surprising as customers with less debt have a higher tendency to purchase the product. Meanwhile, *CustomerNrOfOtherLifeProducts*, *CustomerNrOfMonthsActive* and *Avg1WkLag* are positively correlated with the hit rate. To interpret this briefly, the longer the person has been a customer of the company, the more likely it is for the person to purchase child insurance, while those having been subscribed to a higher number of other life products have a higher likelihood of purchasing the product. Meanwhile, as time is also a significant determinant of the hit rate, the average hit rate of the previous week has a positive relationship with the probability of an offer being accepted, according to the logistic regression model.

6 Machine learning models to predict product type

6.1 Introduction

We are now ready to tackle the fourth and final major part of this thesis: whether we can predict the type of child insurance a customer will be interested in based on the offer characteristics explained in section 3. Technically speaking, the question is analogous to the one treated previously: given that a customer purchases a product and a set of characteristics about that customer, we want to predict whether the product type involved is standard or economy (i.e. a binary classification problem, as performed above for sales). We will attempt to make these predictions using three different methods to be outlined below. The results obtained will then be interpreted to give insights into whether separate customer profiles really can be identified. If not, then this would strongly suggest that no separate customer segment exists for the recently launched economy product, and that this could help explain its underwhelming performance.

6.2 Data set

We here start out with the same cleaned data set with 30 explanatory variables (explained in section 3) that we used to create the logistic model in section 5. But, given that the economy product was introduced in the second half of July 2019, we now only include observations from this date onwards. Moreover, we also deleted all observations where the offers were rejected and ended up with 10,403 observations. To address our new binary problem, we created a new target binary variable, *Standard*, where the value is 1 when the standard product is purchased, whilst it is 0 when the customers buy the economy one. This is the data set that will be used in all of the models discussed in this section.

6.3 Explanation of models selected

6.3.1 Supervised learning 1: logistic regression

This involves using the same method as implemented in section 5. The technical details of the model have been discussed in section 2.3.2, and so will not be repeated here. Logistic regression was chosen for the attractive features already outlined above, and since the methods and forms of the results will by now be familiar to any reader of the thesis. This method is also very commonly used when predicting a binary outcome, as we are again dealing with in this case.

6.3.2 Supervised learning 2: XGBoost

After using logistic regression, a generalized linear model, we decide to employ another kind of supervised learning method, which uses a different approach. Several tree-based predictive models are widely used, among them decision trees, random forest and boosting techniques such as XGBoost. This was discussed in section 2.3.3. We choose XGBoost as our second supervised candidate for several reasons. One of them is the fact that its tree-based nature means that its precision can be fine-tuned as strongly as possible, so that unlike generalized linear models, it can give very strong train accuracy, almost regardless of how strong the patterns are within the data set. Another advantage of this, this time compared to decision trees, is that XGBoost gives a more robust outcome, as decision trees are very vulnerable against small changes in the data set. XGBoost, as a gradient boosting technique, uses different subsamples for training and selects different subsets of explanatory variables, and provides more robust outcomes since averaging reduces variance. Obviously, the fact that it is capable of providing strong train accuracy indicates that the method is prone to overfitting[19]. This is an example of the bias-variance trade-off, where a more flexible model (with less bias) captures more of the noise in the train set (increasing variance), resulting in lower accuracy in the test set. In our case, we will take advantage of this in order to demonstrate how well a very precisely trained model, that hence manages to capture almost the entire bias in the data set, handles predicting product type per offer in the test data vs. the test accuracy of a fine-tuned and therefore more generalized model. The comparison of these two separate outcomes will give us an idea about whether the XGBoost technique manages to capture a substantial difference between customers receiving offers of the standard product vs. those offered the economy product.

6.3.3 Unsupervised learning: K-means and K-modes clustering

We have chosen this method for the following reason. Since unsupervised learning finds underlying patterns within a data set, without any kind of human input as to the type of pattern sought, we expect the result to be certain distinguishable customer groups, if these indeed exist. Since the actual results remain to be seen, we will invent an example to illustrate the kind of pattern we are looking for. Let us suppose that we find that those who have children in their 20s and those who get them in their 30s constitute two groups, each of which separately have certain shared

[19] The relationship between overfitting and train vs. test accuracy was explained in section 2.6.

characteristics. We can then derive insights by comparing the ratios between the two product types among these different customer groups. This is how unsupervised learning can be used to give us information about whether different customer segments really exist for the two products. In light of the abovementioned explanation, our unsupervised learning strategy will be as follows:

First of all, we must make a separation within our explanatory variables, given that we have a mixture of continuous and categorical ones. Secondly, we know *a priori* the number of clusters we should build, namely two, given that we have two product types in our case. This suggests that we can use K-means clustering and its variations. To elaborate, we will take advantage of the fact that K-means clustering performs well on data containing continuous variables and where the number of clusters is known beforehand. In the first place, we will create a new data set containing only continuous variables and run K-means clustering with $K = 2$. To figure out whether these two groups are distinguishable in terms of product selection, we will compare the ratios between the standard and economy products for each cluster. Later on, we will run K-modes clustering on the categorical data, given that instead of averaging as in K-means, frequency is a sensible measure for clustering categorical data. We will then pursue the same procedure as for K-means clustering, in order to check whether the clusters differ in terms of their product selection. On top of that strategy, we will also discuss some noticeable differences between the two clusters for both K-means and K-modes clustering, which may potentially constitute a base for a more in-depth discussion of what the correct product development strategy should be.

6.4 Results and evaluation of models selected

6.4.1 Supervised learning 1: logistic regression

The difference between the current model and that trained in section 5 is that now the model is trained to predict the binary variable *Standard* (encoded using the same method as before). Again, backwards stepwise subset selection according to AIC score is used to find the optimal variables to include in the model. We do not this time split the data set into train and test sets, for reasons which will become apparent towards the end of this section. The results of the logistic regression model are then as follows.

To get an initial idea of the maximum accuracy, we use the optimal 0.5 probability cut-off for classification purposes, which results in an accuracy score of 0.88. This sounds impressively high, but the confusion matrix reveals a significant problem:

Predicted \ Actual	**0**	**1**
0	201	121
1	1158	8923

Table 6.1: Confusion matrix of the logistic regression model.

We here find the numbers behind the accuracy score broken down according to the two categories, and we note a striking difference in how well each category is predicted: 98.7% of all positives (i.e. standard products) are correctly classified as such, but this share plummets to only 14.8% for the economy category. The accuracy score alone gives a highly imbalanced picture of the situation, since it weights true positives (correct identifications of the standard product) and true negatives (correct identifications of the economy product) equally, even though 86% of all products are of the standard type.

We can illustrate the problem the following way: if we had chosen to predict the product class by simply predicting the most frequent class (i.e. standard) each time, we would be correct in 86% of cases. This would hence result in an accuracy score of 0.86. So our model's optimal accuracy score of 0.88 constitutes an improvement of only 2 percentage points. Now we see that this initially impressive score really is extremely poor. Using such a vast data set and even sophisticated and computationally intensive methods such as subset selection according to AIC, we have hardly improved the accuracy beyond what we could have found using the most naive of approaches.

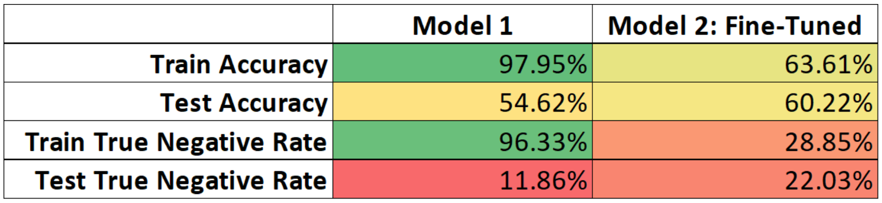
Furthermore, since this model is trained and evaluated using the same data set (from when the economy product was launched and onwards), it will also exhibit a degree of overfitting. This means that the already poor results we have seen would be even worse in the real world. Given the already slender margin between the accuracy score of the model and what we would have achieved using the naive prediction approach mentioned above, it is highly likely that the model would have performed worse than the naive approach in the real world. This seems evident enough that we find it unnecessary to demonstrate this by re-training and evaluating the model using a test set.

Hence we conclude that it would hardly be an exaggeration to say that this model is worthless for predictive purposes. Using such a complex method to achieve such a marginal improvement, which in reality is likely to vanish when applied in the real world, cannot be recommended. Given that the model is virtually useless, we have placed the coefficients and associated model details in **Appendix B** instead of discussing something that is unlikely to give any real insights into the data. But while the *model* may be a failure in terms of prediction, the knowledge of this fact is in itself valuable since it strengthens the hypothesis that no separate customer segment *can* be identified for the two products. Hence the failure of the model is not a dead-end in itself. This is something we discuss further in the conclusion, section 6.5.

6.4.2 Supervised learning 2: XGBoost

In order to find out whether we can go any further in terms of distinguishing the economy product buyers and the standard product buyers, we opt to use a gradient boosting method, XGBoost. The reason why we selected a method that has a highly acclaimed predictive performance but weak interpretability due to its black-box nature, is that we want to find out whether a very strong prediction technique can distinguish the buyers of different product types. In other words, in making use of this boosting technique, rather than seeking a causal interpretation, we aim purely to investigate whether buyers of different product types can be distinguished or not.

As mentioned in the explanation of the XGBoost in section 6.3.2, boosting methods like XGBoost are able to capture almost the entire bias in the data set mainly due to the fact that the depth of each tree can be predetermined. We ran the XGBoost algorithm with the help of the *xgboost* package that is available in the Python environment. Initially, we intentionally keep the selected tuning parameters, which are the number of estimators, learning rate, maximum tree depth and the gamma parameter, at a guaranteed level of high precision in order to see how well a model that manages to capture the majority of the bias in training data will perform in the test data. Then we will build a second model where the abovementioned parameters are fine-tuned, and thus there is no overfitting in training data. For both models, our benchmark will be the true negative rate, (corresponding to correct identifications of economy product buyers), given that we are trying to figure out whether the model accurately captures particularly the economy product buyers or not, taking into account their scarcity in the data set. In both cases, as previously mentioned, the tuned parameters will be the number of estimators, learning rate, maximum tree depth and the gamma parameter. To briefly explain, the number of estimators stands for the number of trees to be built for the model, the learning rate is the shrinkage parameter that is imposed after each boosting on explanatory variables, the maximum tree depth is by definition the maximum size of each tree and gamma is a hyperparameter to fine-tune the regularization in between each tree to be built. The explanatory variables and the target variables that we used in this section are provided in section 6.2. The outcome of both XGBoost models can be seen in the below:



	Model 1	Model 2: Fine-Tuned
Train Accuracy	97.95%	63.61%
Test Accuracy	54.62%	60.22%
Train True Negative Rate	96.33%	28.85%
Test True Negative Rate	11.86%	22.03%

Table 6.2: Accuracy and true negative rate comparison for both models. Color-coded to make the difference between models more apparent.

As mentioned, Model 1 is tuned to capture the vast majority of the bias in train data (The number of estimators = 400, learning rate = 0.9, maximum depth = 10, gamma = 0.9). From Table 6.2 we see that its true negative accuracy is poor for the test data, even though it performs very well for the training data. To conclude briefly, Model 1 fails at predicting potential economy product buyers.

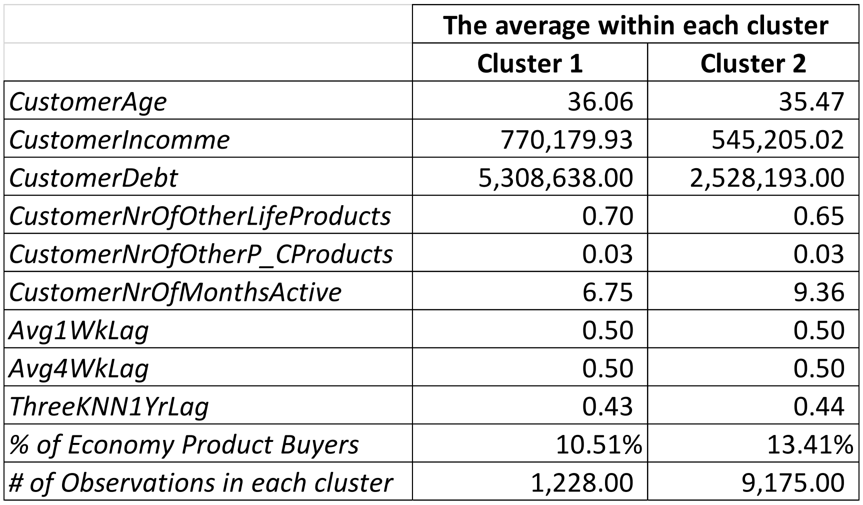
Model 2 is fine-tuned to overcome overfitting (The number of estimators = 400, learning rate = 0.001, maximum depth = 5, gamma = 0). In fact, with newly tuned parameters, we managed to reach a model that does not have significant overfitting, given that the difference between train and test accuracy is relatively low as seen in Table 6.2. However, this model also performs poorly in terms of correctly predicting the potential economy buyers, as it has a test true negative rate of 22%. To sum up, we can undoubtedly conclude that XGBoost could not find a substantial pattern to accurately distinguish potential economy product buyers and standard product buyers.

6.4.3 Unsupervised learning: K-means and K-modes clustering

Finally, we run unsupervised learning on our data set as well to cluster the observations into two groups. This will not only give us information about the differences between clusters, but will also enable us to check whether there is a significant difference between these clusters in terms of product preference. As briefly discussed above in section 6.3.3, we created two different data sets out of our main data set. First, we will take only continuous explanatory variables into consideration and run K-means clustering on it. Second, we will look at categorical explanatory variables using K-modes clustering.

K-means clustering

In this section, we aim to divide our data set into two clusters where only continuous explanatory variables are taken into consideration. Table 6.3 below shows the variables in question and compares the two identified clusters with respect to the average of explanatory variables:



	The average within each cluster	
	Cluster 1	**Cluster 2**
CustomerAge	36.06	35.47
CustomerIncomme	770,179.93	545,205.02
CustomerDebt	5,308,638.00	2,528,193.00
CustomerNrOfOtherLifeProducts	0.70	0.65
CustomerNrOfOtherP_CProducts	0.03	0.03
CustomerNrOfMonthsActive	6.75	9.36
Avg1WkLag	0.50	0.50
Avg4WkLag	0.50	0.50
ThreeKNN1YrLag	0.43	0.44
% of Economy Product Buyers	10.51%	13.41%
# of Observations in each cluster	1,228.00	9,175.00

Table 6.3: Summary of K-means clustering (K=2).

As we can see, K-means clustering where $K = 2$ suggests a small separate cluster (Cluster 1, with 1228 observations) that significantly differs in terms of average income, debt and number of months spent as an active company customer. Customers within this cluster (Cluster 1) tend to have higher income and debt compared to the main cluster (Cluster 2) with 9175 observations. Meanwhile, there are newer customers in Cluster 1, compared to Cluster 2, given that their customers' number of months as an active company customer is roughly around 3 months lower. Finally, we compared these two clusters in terms of the ratio of economy product buyers. Cluster 1 has 3 percentage points lower proportion of economy product buyers compared to the main cluster. On the whole, we can conclude that none of the clusters can be tagged as a cluster for a specific product type, given the low difference of product proportions between clusters.

K-modes clustering

K-modes clustering has been widely used as categorical clustering becomes popular along with the widespread use of data mining techniques, thanks to the fact that it is a simple algorithm to pursue and provides efficient outcomes (Nguyen, 2018). Given that we have a substantial amount of categorical variables in the data set, we opt to use K-modes clustering in order to detect whether there are certain patterns that may potentially lead us to differentiate clusters with respect to their product preference. When running K-modes clustering, we consider the explanatory variables *CustomerCountyCode***,** *Salgskanal***,** *CustomerSex***,** *InsuredChildSex* and *External*.

There are several algorithms offered in the literature to pursue K-modes clustering. The method is first introduced by Huang (1998) as an extension of K-means clustering and as a new dissimilarity measure that can be particularly effective on categorical data (Sajidha et al., 2018). Huang (1998) offers to a random initialization of cluster center selection. In 2009, Cao et al. introduced a new K-modes algorithm, which suggests selecting cluster centers initially by taking into account the distance between objects and the density of each object (Jiang et al., 2015). We know that random initialization is preferred thanks to its simplicity (Khan & Ahmad, 2013). In K-means clustering, we used the most widespread method, which is randomly assigning the initial cluster centers. For the sake of consistency, we pursue the same initial cluster center selection strategy in K-modes clustering.

The outcome of K-modes clustering (whose graphs are shown in **Appendix C**) suggests no significant differentiation in *CustomerCountyCode* and *InsuredChildSex*. That being said, it is observed that the proportion of male customers in Cluster 1, which in total consists of 6127 observations, is significantly higher, whilst the opposite is the case for Cluster 2 (4276 observations). In addition to this, the proportion of *Franchise* as a sales channel is significantly higher in Cluster 1, compared to Cluster 2 and remarkably higher proportion of customers belong to the education level *C* in Cluster 2. Finally, the proportion of customers having any sort of information at the third party business intelligence company (i.e. those for whom *External* is 1) is noticeably higher in Cluster 1. Nevertheless, despite these differences, we find that none of these clusters significantly differ in terms of product preference.

6.5 Conclusion

On the whole, we can undoubtedly conclude that the supervised learning methods employed, logistic regression and XGBoost, fail at accurately classifying true negatives. In other words, these methods cannot accurately separate out the economy product buyers in our data set. Even though further interdisciplinary studies are necessary, the results from the supervised learning methods strongly suggest that there are no significant differences in terms of customer characteristics for the standard and economy products. One should note that this conclusion only holds for customers that have in fact bought one of the two products. Potentially, there might exist a segment of customers where such a distinction is possible, but we cannot detect this in the data set covering the company's current customers. This possibility is discussed in more depth in section 7.

The unsupervised learning methods that we ran on both continuous and categorical variables also do not indicate a marked difference in terms of product preference between clusters. K-modes clustering yields two clusters that show no large distinctions in terms of customer's sex, customer's education, sales channel and whether the customer's information is available at third party business intelligence company or not. However, there is no marked product preference difference between clusters. Similarly, K-means clustering could not find an accurate separation between clusters when product preference is taken into consideration. However, the most striking outcome of K-means clustering is the fact that the smaller cluster (Cluster 1, with 1228 observations) contains customers that have particularly higher income and higher debt. At the same time, the ratio of economy product buyers within this cluster is 3 percentage points smaller compared to the other cluster. Even though further investigation is necessary, it can be argued that Cluster 1 contains customers that potentially would rather have purchased an even more premium product variant compared to the standard one.

To sum up, all of the statistical modelling and clustering methods that we employ in this section find no specific customer group with a marked tendency to purchase the economy product. Meanwhile, our findings from K-means clustering, which would require further investigation to confirm, point to a potential customer segment for a premium child insurance product. The basis for such an argument is that customers in Cluster 1 have markedly higher income and debt compared to Cluster 2. This may sound like it contradicts what we found in section 5, where customer debt and the odds of accepting the sales offer are negatively related. However, the model created in section 5 assigns a probability of being accepted or not for each offer, while in this case we exclude rejected offers and consider only actual buyers. In this case, when only purchasers are taken into consideration, customers who have higher income and debt potentially have a greater willingness to purchase a more premium product, given the lower preference for the economy product within this cluster. This is why, while speculating about this point, we note that further interdisciplinary research on this is necessary to investigate whether this is in fact true or not.

7 Summary and conclusion

7.1 Summary of findings

In section 3, the first main part of the thesis, we in detail explained and explored the data set that we were provided with by the company. We started the investigation by separating our data set into two subsets based on their source: on the one hand, internal data gathered by the company itself, and on the other, external data that comes from the third party business intelligence company. Within the internal data, we investigated continuous and categorical variables separately. In order to start with only potentially useful explanatory variable candidates in the modelling phase, we in this section performed an initial elimination of potential explanatory variable with the help of distribution histograms and correlation plots. While exploring the external data, we discovered that most of these variables are highly correlated, indicating that they are to a large degree estimated and posing a potential problem due to multicollinearity. Luckily, we also discovered that the very presence of such external data, which we encoded as the binary *External*, itself is highly correlated with an offer resulting in a sale. Hence we could avoid the multicollinearity without entirely discarding the external data by including this binary variable rather than the highly correlated continuous ones.

In section 4, before modelling, we investigated the time dependency of the weekly hit rate, in order to figure out whether candidate models should take a time component into consideration or not. Visual inspection suggested that the weekly hit rate demonstrated a positive trend for only a limited period of time. That being said, temporal patterns were discovered by means of ACF and PACF plots. Overall, the outcome of these plots suggested that we should indeed take temporal patterns into consideration during the following modelling phase. Consequently, we created three candidate variables in order to control time dependency during modelling: *Avg1WkLag*, to control for the average weekly hit rate of the week preceding the offer in question, *Avg4WkLag* to do the same for the preceding month, and *ThreeKNN1YrLag*, in order to control for the annual pattern. After investigating time dependency, we began deriving answers to the company's research question by investigating whether the introduction of the new product variant in fact had an impact on the hit rate or not. Change point analysis led us to the conclusion that the company's hypothesis, that introducing the new and more affordable product variant did not have the expected positive impact on sales, cannot be rejected. In other words, thanks to change point analysis, we conclude that we cannot say there is a change in the hit rate related to the introduction of the new product variant.

After investigating time dependency in the weekly hit rate and completing the change point analysis, we in section 5 built an interpretable classification model to assign a probability of an offer resulting in a sale with a fair amount of accuracy. After carefully treating the missing values, we started by using the entire data set to train the model and used a backwards stepwise variable selection method with AIC as the criterion to find the best-fitted model. The outcome of the in-sample validation of the best-fitted model led us to conclude that it has a fair accuracy, in line with what we have found in comparable academic research. Later on, we split the data into a train and test set, based on a specific point in time, in order to see the out-of-sample accuracy of the best-fitted model that we reached by using the entire dataset. We treated missing values from scratch at this stage, in order not to cause an information leakage between train and test data. We could here conclude that the best-fitted model does not overfit, in other words, out-of-sample validation provided us fair accuracy as well. This led us to the conclusion that it is possible to distinguish offers with respect to whether they are accepted or not, with a fair accuracy and with balanced true positive and negative rates.

Finally, in section 6, we narrowed down our investigation by considering only accepted offers from the launch of the economy product onwards, in order to investigate whether it is possible to distinguish the buyers of the two different product types. We started with the same approach that we employed in section 5, namely by creating a logistic regression model. The resulting model could not satisfactorily distinguish the buyers of the economy and standard products. Afterward, we pursued another supervised learning method, XGBoost, to find out whether a tree-based classification method would perform better compared to a generalized linear model. We built two models using the XGBoost method. One was intentionally constructed to be very sensitive to biases within the train data, and hence to provide an excellent in-sample fit, in order to see the acccuracy difference when it is tested on a new data set. A train and test split was performed as in section 5, and led to the conclusion that the very sensitive model that we built undoubtedly failed in terms of test accuracy. Later on, we built a second, fine-tuned XGBoost model in order to minimize overfitting and see whether this would improve the out-of-sample accuracy. As in the previous case, the fine-tuned model performed poorly on the test data in terms of accuracy. Finally, we employed unsupervised learning methods to inspect whether it is possible to separate customers into two groups. As we *a priori* knew the number of clusters, we performed K-means clustering on the continuous explanatory variables and K-modes clustering on the categorical ones. Both of these methods failed in terms of finding a specific customer group with a higher tendency to purchase the economy product. On the other hand, the findings from K-means clustering indicate the existence of a customer group that potentially might be interested in a more premium product variant, rather than a less premium one.

7.2 Synthesis of findings

Our first striking finding is that whether or not the third party business intelligence company provides any information about a customer itself highly influences the probability that an offer will result in a sale. If there is any such external data about a customer, regardless of the content, the offer is much more likely to result in a sale. We also discovered that the hit rate has a temporal pattern. As seen in Figure 4.2, the weekly hit rate demonstrates an upward trend between October 2017 and October 2018, whilst the trend is flat before and after this period. Our change point analysis suggested that the breaking point in the hit rate occurs the 29th of March 2018. Given that the economy product variant was introduced in the second half of July 2018, we conclude that the increasing trend in the hit rate cannot have been caused by the introduction of the new product. Further research is required to analyze the potential reasons behind the increase in the hit rate within the abovementioned period. There could be external factors underlying this, such as an overall rise in the popularity of the product. In addition to this, internal factors can play a role in such an increase, for instance, there could have been an efficiency increase in the performance of the sales team, whether due to the use of new sales methods, marketing strategies etc.

In section 5, we found that the logistic regression model is able to classify offers as being accepted or not with a fair accuracy and balanced true positive and negative rates. The relationship between the hit rate and the following variables were found to almost certainly not be random: whether the third party business intelligence company has provided information about the customer, whether the offer is made via the franchise sales channel, the customer's age, sex, number of months active and number of other subscribed life products. Our conclusion in this section is that it is possible to statistically distinguish buyers and non-buyers of child insurance in general, without considering product type.

That being said, both this approach using logistic regression and one using XGBoost failed to accurately distinguish standard and economy products buyers out of all offers that were made after the economy product had been introduced. On the other hand, K-means and K-modes clustering provided us with meaningful outcomes, but the clusters that those techniques identified did not differ in terms of the ratio between the standard and economy products. However, both K-means and K-modes clustering distinguish a customer group that might have a higher inclination towards purchasing a more premium product. We suspect this since K-means gave us a minority cluster where both customer income and debt are significantly higher, and K-modes identified a cluster where where most customers have information from the third party business intelligence company.

Obviously, we believe that further research can make significant contributions to our findings. To give examples, a survey-based data gathering process would give much more detailed insight about the clusters that we identified using K-means and K-modes clustering. Given such information, further studies can be conducted to find out the optimum price for each product, as it would then be feasible to derive demand functions for these. A demand function would also give us important insights about how price-sensitive child insurance customers are in Norway. In addition to this, such information would allow us to quantify the loss from the sales of the economy product instead of the more premium one.

In summary, our findings show that introducing an economy product variant brought no benefits to the company because the product has no clearly identifiable customer base. This explains why we find no change in the hit rate after the company introduced the economy product. Conceivably, a significant portion of present economy product sales could have been realized as standard product sales if no such product had been introduced, resulting in higher turnover for the company from child insurance. In fact, if the company were to introduce a new product, our findings using unsupervised learning indicate that it should probably have been a more premium product instead of an economy one.

Bibliography

Abolfazli, H., Asadzadeh, S. M., Nazari-Shirkouhi, S., Asadzadeh, S. M., & Rezaie, K. (2014). Forecasting rail transport petroleum consumption using an integrated model of autocorrelation functions-artificial neural network. *Acta Polytechnica Hungarica*, *11*(2), 203-214.

Aliyev, M., Ahmadov, E., Gadirli, H., Mammadova, A., & Alasgarov, E. (2020). Segmenting Bank Customers via RFM Model and Unsupervised Machine Learning. *arXiv preprint arXiv:2008.08662*.

Astari, D. W. & Kismiantini (2019, October). Analysis of Factors Affecting the Health Insurance Ownership with Binary Logistic Regression Model. In *Journal of Physics: Conference Series* (Vol. 1320, No. 1, p. 012011). IOP Publishing.

Beaulieu, C., Chen, J., & Sarmiento, J. L. (2012). Change-point analysis as a tool to detect abrupt climate variations. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *370*(1962), 1228-1249.

Cao, F., Liang, J., & Bai, L. (2009). A new initialization method for categorical data clustering. *Expert Systems with Applications*, *36*(7), 10223-10228.

Ezenkwu, C. P., Ozuomba, S., & Kalu, C. (2015). Application of K-Means algorithm for efficient customer segmentation: a strategy for targeted customer services.

Fader, P. S., Hardie, B. G., & Huang, C. Y. (2004). A dynamic changepoint model for new product sales forecasting. *Marketing Science*, *23*(1), 50-65.

Guillen, M., Parner, J., Densgsoe, C., & Perez-Marin, A. M. (2002). Customer loyalty in the insurance industry: a logistic regression approach. In *II Conference in Actuarial Science and Finance on Samos, Karlovasi-Samos, Greece*.

Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data mining and knowledge discovery*, *2*(3), 283-304.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R* (Springer Texts in Statistics) (1st ed. 2013, Corr. 7th printing 2017 ed.). Springer.

Jiang, F., Liu, G., Du, J., & Sui, Y. (2016). Initialization of K-modes clustering using outlier detection techniques. *Information Sciences*, *332*, 167-183.

Juang W-C, Huang S-J, Huang F-D, et al. (2017). Application of time series analysis in modelling and forecasting emergency department visits in a medical centre in Southern Taiwan. BMJ Open 2017;7:e018628. doi:10.1136/ bmjopen-2017-018628

Kalton, G., & Kasprzyk, D. (1982, August). Imputing for missing survey responses. In *Proceedings of the section on survey research methods, American Statistical Association* (Vol. 22, p. 31). American Statistical Association Cincinnati.

Kamthania, D., Pawa, A., & Madhavan, S. S. (2018). Market Segmentation Analysis and Visualization Using K-Mode Clustering Algorithm for E-Commerce Business. *Journal of computing and information technology*, *26*(1), 57-68.

Khan, S. S., & Ahmad, A. (2013). Cluster center initialization algorithm for K-modes clustering. *Expert Systems with Applications*, *40*(18), 7444-7456.

Killick, R., & Eckley, I. (2014). changepoint: An R package for changepoint analysis. *Journal of statistical software*, *58*(3), 1-19.

Mantel, N. (1970). Why stepdown procedures in variable selection. *Technometrics*, *12*(3), 621-625.

Mehta, A., Malley, B., & Walkey, A. (2016). Formulating the research question. In *Secondary Analysis of Electronic Health Records* (pp. 81-92). Springer, Cham.

Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in plant science*, *7*, 1419.

Nguyen, H. H. (2018). Privacy-preserving mechanisms for k-modes clustering. *Computers & Security*, *78*, 60-75.

Nielsen, D. (2016). *Tree boosting with xgboost-why does xgboost win" every" machine learning competition?* (Master's thesis, NTNU).

Pawluszek-Filipiak, K., & Borkowski, A. (2020). On the Importance of Train–Test Split Ratio of Datasets in Automatic Landslide Detection by Supervised Classification. *Remote Sensing*, *12*(18), 3054.

Pesantez-Narvaez, J., Guillen, M., & Alcañiz, M. (2019). Predicting Motor Insurance Claims Using Telematics Data—XGBoost versus Logistic Regression. *Risks*, *7*(2), 70. doi:10.3390/risks7020070

Petukhova, T., Ojkic, D., McEwen, B., Deardon, R., & Poljak, Z. (2018). Assessment of autoregressive integrated moving average (ARIMA), generalized linear autoregressive moving average (GLARMA), and random forest (RF) time series regression models for predicting influenza A virus frequency in swine in Ontario, Canada. *PloS one*, *13*(6), e0198313.

Sajidha, S. A., Chodnekar, S. P., & Desikan, K. (2018). Initial seed selection for K-modes clustering–A distance and density based approach. *Journal of King Saud University-Computer and Information Sciences*.

Shuai, Y., & Zhou, Z. (2019). GDP Analysis and Comparison in Coastal Cities Based on Time Series Analysis. *Journal of Coastal Research*, *98*(sp1), 402-406.

SSB. (2020). Retrieved from https://www.ssb.no/: https://www.ssb.no/en/bygg-bolig-og-eiendom/statistikker/boforhold?fbclid=IwAR2bksbtOzLFWDTA0aoTUZcAOrvMvXYtX8idOX_rrJiGNi5W8_YvlibaRk

Thies, S., & Molnár, P. (2018). Bayesian change point analysis of Bitcoin returns. *Finance Research Letters*, 27, 223-227.

Wachinger, C., Becker, B. G., & Rieckmann, A. (2018). Detect, quantify, and incorporate dataset bias: A neuroimaging analysis on 12,207 individuals. *arXiv preprint arXiv:1804.10764*.

Wilson, J. H. (2009). An analytical approach to detecting insurance fraud using logistic regression. *Journal of Finance and accountancy*, *1*, 1.

Xia, Z., Xue, S., Wu, L., Sun, J., Chen, Y., & Zhang, R. (2020). ForeXGBoost: passenger car sales prediction based on XGBoost. *DISTRIBUTED AND PARALLEL DATABASES*.

Yan, H. S., & Tu, X. (2012). Short-term sales forecasting with change-point evaluation and pattern matching algorithms. *Expert systems with applications*, *39*(5), 5426-5439.

Ye, L., Qiu-ru, C., Hai-xu, X., Yi-jun, L., & Zhi-min, Y. (2012, July). Telecom customer segmentation with K-means clustering. In *2012 7th International Conference on Computer Science & Education (ICCSE)* (pp. 648-651). IEEE.

Zaiontz, C. (2020). *Real Statistics Using Excel*. Retrieved from: https://www.real-statistics.com/time-series-analysis/stochastic-processes/autocorrelation-function/

Zaiontz, C. (2020). *Real Statistics Using Excel*. Retrieved from: https://www.real-statistics.com/time-series-analysis/stochastic-processes/partial-autocorrelation-function/

Zhang, D., Qian, L., Mao, B., Huang, C., Huang, B., & Si, Y. (2018). A data-driven design for fault detection of wind turbines using random forests and XGboost. *IEEE Access*, *6*, 21020-21031.

Zhang X, Zhang T, Young AA, Li X (2014) Applications and Comparisons of Four Time Series Models in Epidemiological Surveillance Data. PLoS ONE 9(2): e88075. doi:10.1371/journal.pone.0088075

Appendix A: table of training model details

We find that the *z* values and coefficient estimates are all roughly comparable to the full model:

Explanatory variable	**Estimate**	**Std. Error**	**z value**	**Pr(>|z|)**
External	7.38E-01	1.73E-02	42.80	<2e-16
CustomerAge	-2.77E-02	1.17E-03	-23.58	<2e-16
CustomerSex	-3.21E-01	1.72E-02	-18.68	<2e-16
Salgskanal_Franchise	6.00E-01	3.29E-02	18.26	<2e-16
CustomerNrOfMonthsActive	8.30E-03	5.74E-04	14.45	<2e-16
CustomerNrOfOtherLifeProducts	1.59E-01	1.45E-02	10.93	<2e-16
CustomerCountyCode_38.0	4.87E-01	4.60E-02	10.58	<2e-16
CustomerCountyCode_30.0	3.12E-01	3.38E-02	9.26	<2e-16
CustomerEducation_B	1.88E-01	2.20E-02	8.55	<2e-16
CustomerCountyCode_18.0	2.61E-01	3.31E-02	7.88	3.27E-15
CustomerCountyCode_34.0	4.65E-01	6.11E-02	7.61	2.75E-14
CustomerCountyCode_50.0	4.02E-01	5.41E-02	7.42	1.15E-13
CustomerCountyCode_42.0	1.55E-01	2.40E-02	6.44	1.18E-10
CustomerDebt	-4.45E-08	7.27E-09	-6.12	9.41E-10
CustomerCountyCode_11.0	1.75E-01	2.90E-02	6.05	1.45E-09
CustomerEducation_A	2.57E-01	5.50E-02	4.67	3.01E-06
Avg1WkLag	3.21E-01	8.45E-02	3.80	1.47E-04
Salgskanal_Partner	-2.76E-01	8.78E-02	-3.14	1.67E-03
Salgskanal_Firmaets_egne_kanaler	-1.07E-01	3.65E-02	-2.94	3.31E-03
CustomerCountyCode_54.0	1.33E-01	7.51E-02	1.78	7.59E-02
CustomerCountyCode_15.0	6.71E-02	7.62E-02	0.88	3.79E-01
CustomerEducation_C	7.74E-03	2.02E-02	0.38	7.02E-01
(Intercept)	-5.43E-03	5.59E-02	-0.10	9.23E-01

Appendix B: details of logistic model for product type

This model is fitted using the binary variable *Standard* as the dependent variable.

Explanatory variable	**Estimate**	**Std. Error**	**z value**	**Pr(>|z|)**
(Intercept)	4.137e+00	3.944e-01	10.488	<2e-16
CustomerAge	-2.607e-02	4.473e-03	-5.829	5.58e-09
CustomerSex	1.511e-01	6.954e-02	2.173	0.029784
CustomerIncomme	3.921e-07	1.501e-07	2.612	0.009004
CustomerDebt	6.996e-08	3.005e-08	2.328	0.019928
CustomerNrOfOtherP_CProducts	-1.242e-01	7.192e-02	-1.727	0.084203
CustomerNrOfMonthsActive	9.786e-03	2.370e-03	4.129	3.65e-05
External	-1.304e-01	6.641e-02	-1.964	0.049541
Avg1WkLag	5.447e-01	3.832e-01	1.422	0.155123
Avg4WkLag	-1.916e+00	7.337e-01	-2.612	0.009006
CustomerCountyCode_11.0	-6.076e-01	1.237e-01	-4.912	9.01e-07
CustomerCountyCode_15.0	-1.179e+00	2.148e-01	-5.489	4.05e-08
CustomerCountyCode_38.0	-5.161e-01	1.346e-01	-3.833	0.000126
CustomerCountyCode_42.0	-7.513e-01	1.041e-01	-7.215	5.41e-13
CustomerCountyCode_46.0	-5.808e-01	1.047e-01	-5.547	2.90e-08
CustomerCountyCode_54.0	6.265e-01	3.267e-01	1.918	0.055121
CustomerEducation_A	5.778e-01	3.231e-01	1.788	0.073708
CustomerEducation_B	-1.899e-01	8.270e-02	-2.296	0.021682
CustomerEducation_C	-1.399e-01	7.757e-02	-1.804	0.071272
Salgskanal_Franchise	-2.075e+00	8.760e-02	-23.685	<2e-16
Salgskanal_Firmaets_egne_kanaler	3.499e-01	1.776e-01	1.971	0.048759
Salgskanal_Partner	-1.297e+00	1.113e-01	-11.656	<2e-16
(Intercept)	4.137e+00	3.944e-01	10.488	<2e-16

Appendix C: Figures of the outcome of K-modes clustering

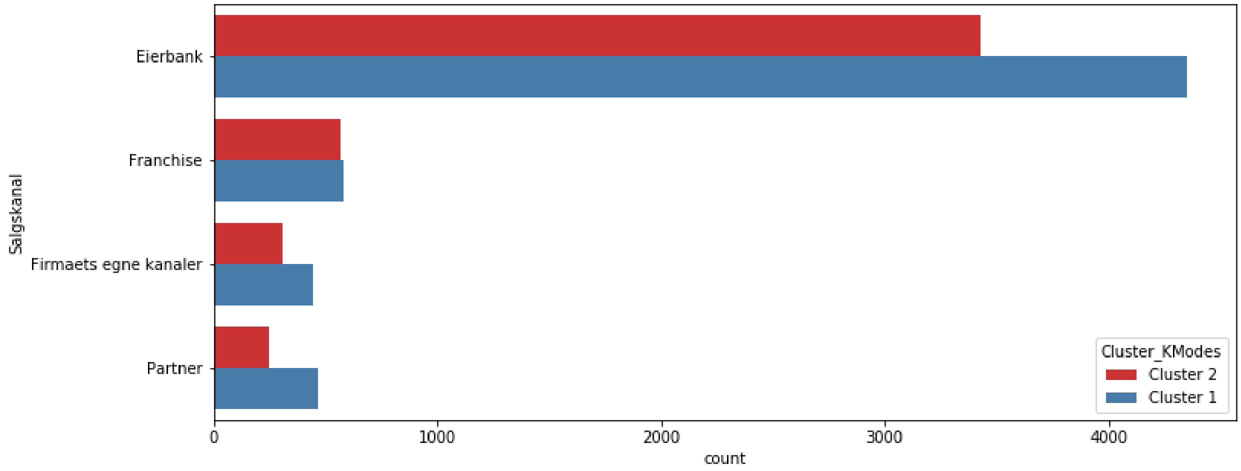


Figure AC.1: Sales channel distribution in absolute terms (In terms of number of observations) - Cluster 1 vs. Cluster 2.

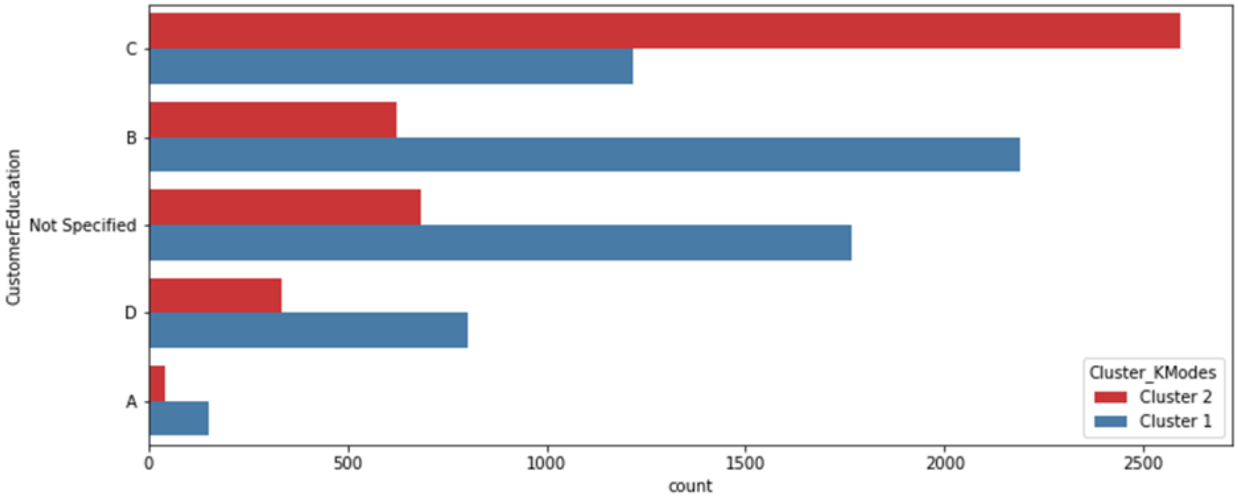


Figure AC.2: Customer education distribution in absolute terms (In terms of number of observations) - Cluster 1 vs. Cluster 2.

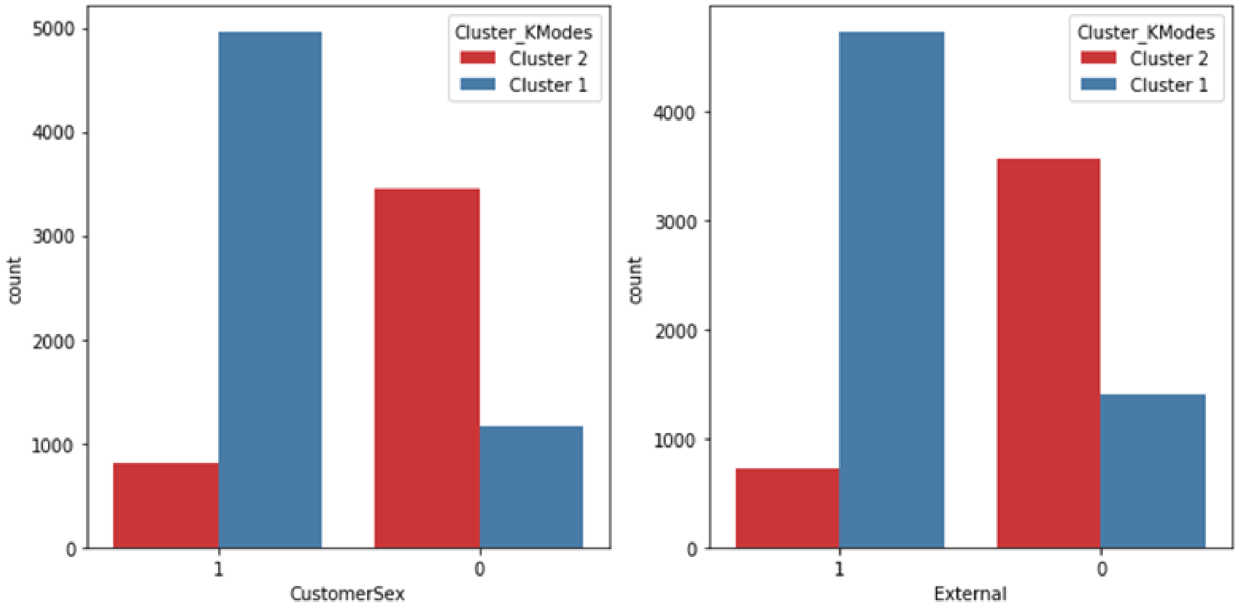


Figure AC.3: Customer sex and ***External*** distribution in absolute terms (In terms of number of observations) - Cluster 1 vs. Cluster 2. CustomerSex = 1 if male, 0 otherwise.

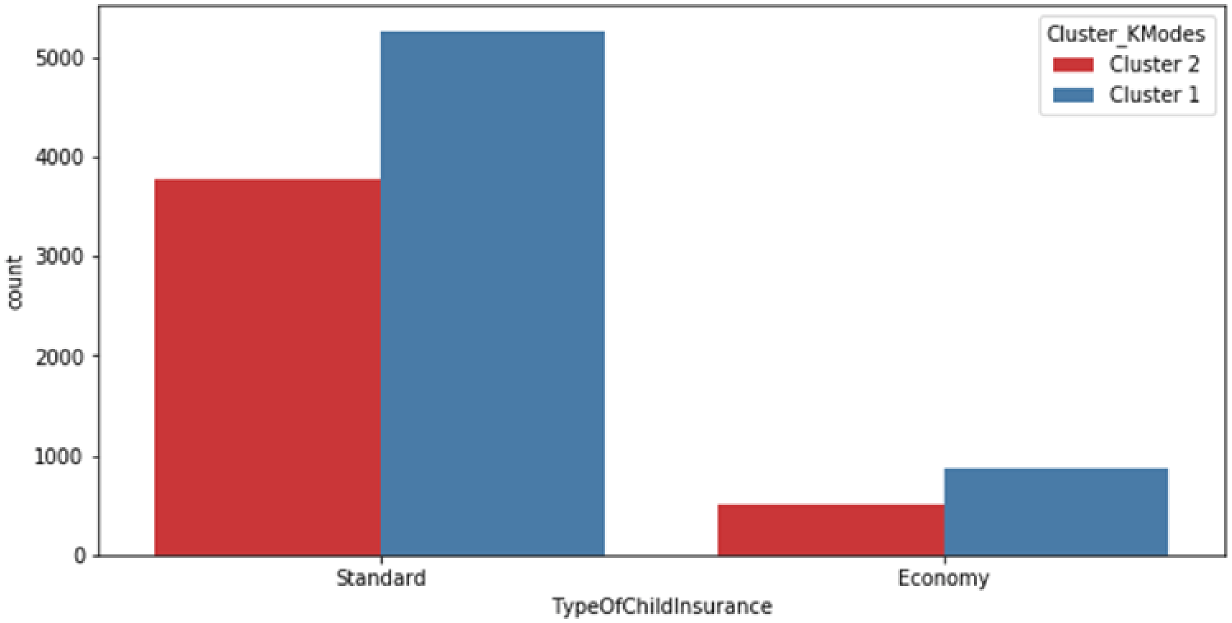


Figure AC.4: Product type distribution in absolute terms (In terms of number of observations) - Cluster 1 vs. Cluster 2.