Norwegian School of Economics
Bergen, Fall 2018

Method for Fusing Predictor Levels with Application to Insurance Data

Henriette Nøkleberg Jensen and Mathias William Rafn

Supervisor: Jonas Andersson

Master thesis in Economics and Business Administration
Profile: Business analysis and performance management with specialization in business analytics

NORWEGIAN SCHOOL OF ECONOMICS

Confidential until 20.12.2023

Acknowledgements

This thesis is based on a collaboration with PwC and Gjensidige related to an ongoing project on developing Gjensidige's framework for predictive modeling. We feel fortunate to have been able to research a topic that is of relevance to a large company like Gjensidige. This has both increased our motivation during our work and challenged us.

We would like to thank Gjensidige for giving us access to their resources, facilities, and data. In particular, we thank Mikkel Hinnerichsen for raising constructive questions and showing enthusiasm related to our work.

We would also like to thank Thomas Langeland, our contact person at PwC Bergen, for facilitating the process of writing this thesis, in addition to being available for questions and support throughout the semester.

It has been challenging to use a newly developed package. We would therefore like to thank Tom Reynkens, one of the developers of the *smurf* package in R, for taking the time to help us solve initial challenges related to using the package.

Lastly, we would like to thank our supervisor Jonas Andersson for guiding us through the process of writing a master thesis and for constructive criticism. He was our lecturer in the course "Predictive Analytics with R" during the spring semester of 2018, which is what originally aroused our interest in the topic of predictive modeling.

Oslo, 20th of December 2018

Mathias William Rafn

Henriette Nøkleberg Jensen

Summary

The aim of this thesis is to determine whether the prediction accuracy of a model can be improved by using a data-driven method to bin continuous variables and group the levels of categorical variables. We use data on the policyholders of one of Gjensidige's insurance products to perform our analysis, and specifically aim to improve Gjensidige's Poisson regression model for predicting claim frequency, where the predictors are binned and grouped manually today.

We analyze the effect of using a regularization framework that combines the Lasso method and generalizations of the method that have been adapted to nominal and ordinal predictors. These generalizations constrain coefficients and the differences between them, effectively fusing and selecting predictor levels. By optimizing the resulting objective function in R using the newly developed *smurf* package (Reynkens, Devriendt & Antonio, 2018), we estimate a penalized Poisson regression model.

We reestimate a Poisson regression model using the selected and fused predictor levels as input in order to reduce the bias of the estimates. The resulting model is compared with the model Gjensidige currently uses for predicting claim frequency, to determine the effect of using the data-driven approach. We validate the performance of the prediction models using MSE and AIC as performance measures and find that our reestimated model performs slightly better in terms of prediction accuracy, in addition to reducing the number of parameters used in the model. We conclude that regularization can be used as a data-driven method of binning and grouping predictor levels to improve prediction accuracy.

Contents

1. Introduction

1.1 Motivation

The motivation for writing this thesis is based on a request from Gjensidige, related to their ongoing project on improving their framework for predictive models. In initial meetings regarding the content of this thesis, our contact person from Gjensidige presented an interesting challenge concerning how they treat their variables prior to modeling. Firstly, Gjensidige uses nominal and ordinal variables in their prediction models, but before they are included as predictors in the models, the number of categories within the variable is usually reduced by *grouping* some of the categories together. Secondly, Gjensidige recodes some of the continuous variables to ordinal variables by dividing them into intervals. These processes are similar as they both concern determining which values of a predictor that bear resemblance to each other and can be treated together. Therefore, these processes will often be discussed as one throughout the thesis.

The challenge with Gjensidige's approach today is that these processes and the decisions related to them are performed manually. Analysts choose the groups and set the intervals using their intuition and experience. It is a time-consuming task that could reduce the prediction ability of the model if done poorly. As automating the processes of their prediction framework is an important aspect of the ongoing project, studying how to treat these variables in a more data-driven and automated way is a relevant project both for us and for Gjensidige. Based on this challenge, Gjensidige's request was for us to find a new approach for choosing groups and setting intervals that can save time for the analysts and possibly improve the prediction accuracy of the models. Improved prediction models can lead to a fairer pricing for their customers, as it will better reflect the likelihood of them using their insurance, which is one of the reasons why this topic is of interest to us.

Clarifications

Categorical variables can take a limited number of different values that are commonly referred to as categories, groups or levels. Throughout this thesis, we will mainly refer to these values as levels, both for the nominal variables, ordinal variables and the continuous variables recoded to ordinal variables. Being the main purpose and motivation for the thesis,

the earlier mentioned *grouping* of these levels will be discussed frequently and referred to as grouping, combining or fusing levels.

Why group at all?

Many of the categorical predictors Gjensidige currently uses have many levels. Problems arise when some of the levels contain few observations, as this leads to estimates with high variance. The variables can still provide valuable information, and a possible solution is to fuse together some levels of the variable. Another reason to group the levels became clear to us when we tried to run a model where all levels were included as dummy variables and ended up being contacted by Gjensidige who warned us that we were taking up too much of their server capacity.

A common challenge

Reducing the number of levels of categorical variables is a common challenge, but there are not any well-known best practice solutions. Therefore, during the last couple of decades, many new approaches and methods have been suggested. This is another reason why this is an exciting topic to study at this point in time.

1.2 Background

In this section we will provide background information about Gjensidige and common concepts within the insurance industry. We will also explain how insurance companies use predictive modeling to calculate premiums based on data on their clients, in order to illustrate the context of the task from Gjensidige.

1.2.1 **Insurance concepts**

Insurance companies form an important fundament of a functioning economy, as they secure the financial stability of households and firms (ECB, 2009). Every day, households and firms face risk and uncertainty, and insurance companies can help manage this uncertainty by offering products that provide financial protection against potential economic losses (Iowa Insurance Institute, 2017). These products and their terms are outlined in contracts, called *insurance policies*.

The party holding the insurance is called a *policyholder*. The policyholder pays a fee to the insurance company and is then compensated if losses incur according to the policy. Insurance companies thus stimulate economic activity by ensuring that the policyholders continue to purchase and invest despite the risks they encounter. The fee policyholders pay is called the *premium* and is usually paid monthly or yearly. If the policyholder experiences a loss potentially covered by the insurance policy, they may submit a *claim*, which is then examined by the insurance company. If the claim is indeed covered by the insurance policy, the insurance company compensates the policyholder (Iowa Insurance Institute, 2017).

1.2.2 About Gjensidige

Gjensidige is a Nordic insurance group that offers insurance products in Norway, Sweden, Denmark and the Baltic countries. In Norway, the company also offers services within banking, pension and savings (Gjensidige, 2018a). In 2017, Gjensidige was the largest insurance company in Norway with a market share of 25.5% and had an operating income of 27 billion (Gjensidige, 2018b). Gjensidige offers a range of insurance products in Norway, covering cars, homes, boats, travel, pets, life and health, valuables and personal property (Gjensidige, 2018c).

1.2.3 The importance of predictive modeling

For insurance companies like Gjensidige to be profitable, it is vital that the premiums they charge are at a competitive level, but still cover the losses they have on their clients' claims. Due to asymmetrical information between the insurance company and the policyholder regarding the risk of the client, problems with adverse selection arise (Finkelstein & Poterba, 2000). Within the insurance industry, the problem of adverse selection is related to the tendency of high-risk clients to be more likely than the average client to buy insurance. To avoid taking losses on these clients, the insurance companies have to take their clients' risk profiles into account when setting premiums. Therefore, the insurer will usually charge different premiums across the customer base, increasing the premium for clients that are considered high-risk.

However, they also need to keep the premiums sufficiently low so that the low risk clients have incentives to buy insurance at all. This helps the insurance companies obtain a larger and more differentiated customer base. When insurance companies calculate the premiums

for different products, predictive modeling is a valuable tool for determining the risk profile of their clients, and thereby reducing the problem of asymmetrical information that leads to adverse selection. It is therefore essential that the prediction models perform well, which makes it a natural priority for Gjensidige to allocate resources into the research of possible improvements to their models. As the attributes of the policyholders are valuable indicators of their risk profile, it is important to find out how to handle the variables in order to take advantage of this information.

1.2.4 Calculating the premium

The predictive model for a specific insurance product is designed to predict the amount of money an individual, with a specific set of attributes, will claim yearly. This is referred to as the expected loss for this customer. The expected loss will then form the basis for the premium that a customer with these attributes will have to pay to be covered by the insurance company. On top of this price, insurance companies like Gjensidige may add administrative fees and discounts depending on which customers they want to attract and retain (Parodi, 2016).

There are two possible approaches for predicting the expected loss of a policyholder. One option is to build a model that directly estimates the expected loss of the individual. Another option is an indirect approach where one model is built for predicting claim frequency, which is the number of claims in a year, and another model is built for predicting claim severity, the total loss per claim. If one chooses to use the indirect approach, the two models can be combined by multiplying the expected claim frequency by the expected claim severity to get the expected loss for the customer (Goldburd, Khare & Tevet, 2016).

Building two models instead of one will most likely demand more resources, but there are several advantages to this approach that make it a common choice. For one, it often provides more insight than predicting the expected loss directly, as it is possible to distinguish which factors affect the frequency of claims and which factors affect the severity of these claims. In some cases, some effects may even disappear completely when predicting the expected loss directly, as it is possible that some attributes have a positive effect on frequency but an equally negative effect on severity (Goldburd et al., 2016).

1.3 Research question

The purpose of this thesis is to find a method for grouping the levels of categorical variables in a data-driven way, rather than doing it manually. For the method to be valuable for Gjensidige, it must prove to be better than their current one in some way. The method can benefit Gjensidige through both increased prediction accuracy and reduction in time spent on grouping the variable levels. However, prior to a potential implementation of the method, the change in prediction accuracy is the only available measure and will be the focus of this thesis. Therefore, the research question of this thesis is:

Can the prediction accuracy of Gjensidige's claim frequency models be increased by using a data-driven method for the fusion of levels of categorical predictors?

1.3.1 **Delimitation**

Originally, Gjensidige asked us to find a way to fuse the levels of the predictor representing vehicle brand. However, they were also interested in a method for fusing any other type of categorical predictor and for performing variable selection. Through our research we have been able to find a relevant method developed by Devriendt, Antonio, Reynkens & Verbelen (2018), which is able to do all of this simultaneously. In order to answer our research question, we will use this method on Gjensidige's data on their policyholders and investigate the effect on prediction accuracy.

Gjensidige's framework for prediction models involves predicting the expected loss for policyholders indirectly through separate models for predicting claim frequency and claim severity. To limit the scope of this thesis, we focus on improving the model for predicting claim frequency, but our potential findings may be transferred to the other model types. We further limit the scope by only modeling the claim frequency for one insurance product. The insurance products related to motor vehicles have the second-most claims of all types of insurance in Norway (SSB, 2018). These products are therefore important and as Gjensidige has high exposure and relatively high claim frequency for most of them, they are suitable for our project. To perform our analysis, we use data on the policyholders of *comprehensive motor vehicle coverage* where the claims are related to windscreen coverage. Comprehensive

motor vehicle coverage is an expensive insurance product that covers a range of damages and earns insurance companies high premiums.

1.4 Structure

Chapter 2 outlines the methods that will be used to answer our research question. In chapter 3 we describe the dataset we perform our analysis on and explain how we have treated the variables differently than Gjensidige prior to modeling. In chapter 4 we explain how we have performed our analysis and present our results. Further, in chapter 5 we discuss the theoretical background and implications of our results, before we make our final conclusions in chapter 6.

2. Methods

In this chapter, we describe our selected method further, where we employ some of the benefits that can be obtained when using regularization methods to select and fuse levels. By adding a penalty term which constrains the coefficients of the Poisson regression model Gjensidige uses today, prediction accuracy can be improved. We go through the components of the objective function we minimize, before explaining what algorithm we use for optimization. In the last section of the chapter, we will explain what type of dataset and which performance criteria will be used for the validation and comparison of the models.

2.1 Selection of method to fuse and select levels

When selecting a method to answer our research question, it was important for us to find a method which could be used for both nominal, ordinal and continuous predictors. In addition, it was important that it could possibly improve prediction accuracy; simply choosing a method based on being data-driven was not enough. Furthermore, Gjensidige uses large datasets to create their models, so the method had to work for large datasets.

For our purpose, the use of regularization techniques to shrink coefficients is likely the approach where the most research has been performed (Tibshirani, 1996; Tibshirani, Saunders, Rosset, Zhu & Knight, 2005; Bondell & Reich, 2009; Gertheiss & Tutz, 2010; Oelker & Tutz, 2017; Devriendt et al., 2018). Through extensions of the original regularization methods like Ridge regression and Lasso, they are now able to bin continuous variables and fuse categorical predictor levels, which means we can use the techniques for our purpose. The latest method developed by Devriendt et al. (2018) can be used for categorical variables on large datasets and has shown to improve prediction accuracy. Before we expand on this method, we will present some alternative approaches considered in the literature to fuse levels of categorical variables.

2.1.1 Alternative methods

Traditionally, the most common approach for handling categorical predictors for regression purposes has been converting each level of the variable into a dummy variable (Johnson & Kuhn, 2013) and occasionally also reducing the number of levels by combining those with

few observations into an “Other” category. However, the prior is computationally demanding if there are many levels, while the latter does not take the levels’ effect on the dependent variable into account, likely reducing the prediction accuracy of the estimated model.

Consequently, as we have discovered through the process of selecting a method, research has been conducted throughout the last two decades to find better methods for collapsing levels of categorical variables. It is still an emerging field of research of which few approaches have been tested extensively, which makes implementation challenging. An ad-hoc method is to use classification and regression trees (CART) (Hastie, Tibshirani & Friedman, 2009). The clustered categories can then be found by analyzing the tree nodes, which contain information on how the predictors are divided into regions depending on their effect on the dependent variable (Hastie, James, Tibshirani & Witten, 2017). Berger & Tutz (2014) systemized this approach by introducing tree-structured clustering to obtain clusters of categorical data, allowing other types of variables to be included in the model. The non-categorical variables will then have a linear or additive effect on the response.

Another suggested approach is using Tukey’s test to perform pairwise comparisons within predictors (Tukey, 1949; Hothorn, Bretz & Westfall, 2008) to determine which levels differ from each other simultaneously. Sparse Bayesian modeling of the effects of categorical variables within a regression framework has been considered by Pauger & Wagner (2017), where a spike and slab prior is placed on differences between regression coefficients. Unfortunately, all the aforementioned methods are problematic to implement in R. Either they cannot handle predictors with many levels, the corresponding packages are not maintained, or they are too computationally intensive. Therefore, we elect to use a regularization method.

2.2 Objective function of the regularized model

To create the regularized model, we minimize the penalized objective function (Devriendt et al., 2018)

$$\mathcal{O}(\boldsymbol{\beta}; \boldsymbol{X}, \boldsymbol{y}) = f(\boldsymbol{\beta}; \boldsymbol{X}, \boldsymbol{y}) + \lambda \sum_{j=0}^{J} g_j(\boldsymbol{\beta}_j), \qquad (2.1)$$

where $\boldsymbol{y}$ is the response vector, $\boldsymbol{\beta}$ is the parameter vector, while $\boldsymbol{X}$ is the corresponding model matrix. In $\boldsymbol{X}$, continuous and binary predictors are represented by one column since they are coded with one parameter, while nominal and ordinal predictors are dummy-coded, and are therefore represented by as many columns as they have levels. *f* refers to the loss function, measuring the distance between the observed and fitted data. It can for example be the least squares criterion or minus the log-likelihood (Devriendt et al., 2018).

The second term of the objective function (2.1) is the penalty term. The vector $\boldsymbol{\beta}$ has been partitioned into a subvector $\boldsymbol{\beta}_j$ for each predictor *j* and contains all the parameters used to code the predictor. For a continuous variable, there will only be one coefficient in subvector $\boldsymbol{\beta}_j$, while for a categorical variable with many levels, there will be one coefficient for each level. g_j represents different types of penalties and penalty weights and is chosen depending on the predictor type of each predictor *j* that is penalized.

In the next sections, we elaborate on the different components of the objective function. We begin by describing our loss function, the negative Poisson log-likelihood. Further, we describe what regularization methods, penalty terms and penalty weights are and how they can be used.

2.3 Poisson regression

The objective function (2.1) includes a loss function *f* that measures the distance between the observed and fitted data. Gjensidige uses a Poisson regression model to predict claim frequency, making the negative Poisson log-likelihood the loss function. As our aim is to find out if new groupings of variables improve prediction accuracy rather than changes in the loss function, we will use the same loss function as Gjensidige for our model. A Poisson regression model can be estimated using the *stats* package (R Core Team, 2018) in R.

A Poisson regression model assumes that the random component of the regression model has the Poisson probability distribution (Dunteman & Ho, 2006). It is the main tool used for estimating claim frequency in the insurance industry, because the distribution is well suited for a situation where there are few occurrences of the event compared to the amount of trials, but the event can happen in any of the trials (Goldburd et al., 2016). This description is usually suitable for an insurance model as the vast majority of the policyholders do not have any claims.

If the random dependent variable Y_i conditioned by the vector of predictors X_i is assumed to be Poisson distributed, the probability density function of Y_i is (David & Jemna, 2015)

$$f(y_i|x_i) = \frac{\mathrm{e}^{-\mu_i}\mu_i^{y_i}}{y_i!}, \qquad (2.2)$$

where e is the base of the natural logarithm and μ is the distribution parameter. μ represents the average number of events in the given time interval, for example the number of claims. Therefore, Equation 2.2 represents the probability that Y_i will take the value y_i ($y_i \in \mathbb{N}$), dependent on the attributes of policyholders. The mean and variance of the true Poisson distribution are equal, so the distribution parameter μ represents both the mean and the variance of the dependent variable.

For the Poisson distribution, the mean of the dependent variable is related to the linear predictor through the natural logarithmic function. A linear model can be used to estimate the relationship between the predictors and $log(\mu)$. However, we are not interested in the transformed μ, but the predicted μ itself, which is derived by applying the inverse link function g to the calculated linear predictors (Goldburd et al., 2016)

$$\log(\mu) = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} \Rightarrow \mu_i = e^{x_i^t \beta}. \qquad (2.3)$$

Estimation of the parameters is done by maximum likelihood estimation. To find the maximum likelihood of Equation 2.2, the likelihood function is defined as (David & Jemna, 2015)

$$L(\beta) = \prod_{i=1}^{n} \frac{e^{-\mu_i}\mu_i^{y_i}}{y_i!} = \prod_{i=1}^{n} \frac{e^{-e^{x_i^t\beta}}\left(e^{x_i^t\beta}\right)^{y_i}}{y_i!}. \qquad (2.4)$$

By using a logarithm on both sides of the equation, the log-likelihood function is found (David & Jemna, 2015)

$$LL(\beta) = \sum_{i=1}^{n} [y_i \ln \mu_i - \mu_i - \log y_i!] = \sum_{i=1}^{n} \left[y_i x_i^t \beta - e^{x_i^t \beta} - \log y_i! \right]. \qquad (2.5)$$

The scaled negative of the log-likelihood function will be the loss function in our objective function (2.1). To create the scaled negative of the log-likelihood function, we divide by the number of policyholders and negate the equation

$$-\frac{1}{n} \sum_{i=1}^{n} \left[y_i x_i^t \beta - e^{x_i^t \beta} - \log y_i! \right]. \qquad (2.6)$$

2.3.1 **Offset**

When predicting the number of claims for a policyholder within a certain time interval, it is relevant to include which proportion of this time period the individual held the insurance. To account for this in the model, an offset is included. An offset is defined as a predictor whose coefficient is constrained to be equal to 1 (Goldburd et al., 2016). By including exposure as an offset, the estimated coefficients of the other predictors are affected to take the exposure into account

$$log(\mu) = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \log(exposure), \qquad (2.7)$$

which can be re-written as

$$log\left(\frac{\mu}{exposure}\right) = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}. \qquad (2.8)$$

In Equation 2.8, the left-hand side of the equation is the rate of claims per unit exposure. Therefore, by including exposure as an offset in the objective function, the predicted number of claims will be equal to be the rate of claims per unit exposure. When including an offset, it is important that it is on the same scale as the linear predictor. Therefore, as we use a log-link model, the offset must be logged (Goldburd et al., 2016). By including the offset in the objective function, we ensure that short-time policyholders are not weighted

disproportionately much, which would lead to us systematically underpredicting the number of claims.

As we use the scaled negative of the log-likelihood function and include exposure as an offset, the first term of the objective function using the notation of Devriendt et al. (2018) becomes

$$-\frac{1}{n}\sum_{i=1}^{n}(y_i(x_i\beta + \log(expo_i)) - e^{(x_i\beta+\log(expo_i))} - \log(y_i!)). \quad (2.9)$$

2.4 Regularization methods

The second term in our objective function is a regularization term that constrains coefficients. Its components are the tuning parameter λ, a penalty function and penalty weights. Firstly, we explain what regularization methods are in general before we in later sections explain each component in detail.

To explain what regularization methods are, we will show an example which includes a penalty term and a tuning parameter λ. One of the most well-known penalty terms is the term used in Ridge regression (Hoerl & Kennard, 1970)

$$\mathcal{O}(\boldsymbol{\beta}; \boldsymbol{X}, \boldsymbol{y}) = -\frac{1}{n}\sum_{i=1}^{n}(y_i(x_i\beta + \log(expo_i)) - e^{(x_i\beta+\log(expo_i))} - \log(y_i!)) + \boldsymbol{\lambda}\sum_{\boldsymbol{j=1}}^{\boldsymbol{p}}\boldsymbol{\beta_j^2}, \quad (2.10)$$

where the penalty term is the sum of all the model's coefficients. In the case of Ridge regression, an L2 penalty is used, meaning the sum of the penalty is squared. However, in later examples we will see that this is not the case for all regularization techniques. The size of the penalty term is small when the beta estimates are closer to zero, which means that the penalty term shrinks the estimates of β_j towards zero when the expression is minimized.

To what extent the coefficients will be constrained depends on the tuning parameter λ, which the penalty term is multiplied by. As the size of λ decides the relative strength of the penalty compared to model fit, a higher value of λ will increase the impact of the penalty term and the coefficient estimates will approach zero. If λ is set to 0, the method will produce the same estimates as the objective function without the penalty term (Hastie et al., 2017).

Hence, the size of λ is selected based on our preference between model fit and shrinkage of coefficients.

Even though the main purpose of regularization methods is to increase interpretability by creating less complicated models, regularized models can sometimes improve prediction accuracy if the variance is reduced more than bias increases (Hastie, Tibshirani & Wainwright, 2015). A model with high bias is trying to explain a complicated relationship with a model which is too simple. For example, if trying to explain a non-linear relationship with a linear model, the number of parameters should be increased to create a more flexible model.

A more flexible model can take on more functional forms because it can choose between more parameters, which makes it able to explain a more complicated relationship (Hastie et al., 2017). A higher variance means the estimated model would differ to a greater extent if it was used on different datasets, which means the number of parameters should be decreased to create a less flexible model. Consequently, there is a trade-off between the bias and the variance.

As λ increases, coefficients are constrained, leading to a less flexible model. As the flexibility of the model decreases, the variance of the estimates is reduced, while the bias increases (Hastie et al., 2017). It is the relative change of variance and bias which decides whether prediction accuracy is increased. In many cases, a small increase in bias can lead to a larger reduction in variance, especially if the model overfits the data (Johnson & Kuhn, 2013).

2.5 Penalty types

The datasets Gjensidige use to predict claim frequency include several different predictor types, including binary, nominal, ordinal and continuous predictors. Gertheiss & Tutz (2010) were the first to introduce regularized regression for multiple predictor types. In their method, penalties adapted to each predictor type are combined to act on the objective function as a sum of sub-penalties. The method can be used for datasets that include all the predictor types needed by Gjensidige. In this section, we will describe each of these penalty types and what predictor types they are suitable for.

2.5.1 Lasso

The least absolute shrinkage and selection operator (Lasso), introduced by Tibshirani (1996), applies a penalty term similar to the one used in Ridge regression. Adopting the notation of Devriendt et al. (2018), the Lasso penalty can be expressed as the following

$$g_{Lasso}(\boldsymbol{\beta}_j) = \sum_{i=1}^{p_j} w_{j,i}|\beta_{j,i}|, \qquad (2.11)$$

where j represents predictors and i represents coefficients. Therefore, p_j is the number of individual coefficients $\beta_{j,i}$, while $w_{j,i}$ is the penalty weight for each coefficient of each predictor. The Lasso uses an L1 penalty and each individual coefficient is multiplied by its corresponding penalty weight and added to the total sum of coefficients. The L1 penalty is equal to the absolute value of the sum of the coefficients, differing from the L2 penalty of Ridge regression where the sum of the coefficients is squared (Hastie et al., 2017). Using the L1 penalty, the coefficient estimates are shrunk towards zero, and some may even be set to zero.

The Lasso is suitable as a selection tool for binary and continuous predictors as they only have one coefficient. Therefore, as the Lasso is applied, only the most important predictors receive non-zero coefficients, while the rest are removed from the model. In the case of categorical predictors, if the coefficient estimate of a level is set to zero, the level is removed from the model. If all levels are set to zero, the predictor is removed entirely. The limitation of using the Lasso for categorical variables is that it does not fuse levels together, and only works as a selection tool. Consequently, Tibshirani et al. (2005) introduced the Fused Lasso.

2.5.2 Fused Lasso

The Fused Lasso is designed for models containing features that can be ordered in a meaningful way, namely ordinal variables or continuous variables recoded as ordinal variables. The method applies a penalty on both the coefficients themselves and the differences between coefficients of subsequent levels. As a result, it can perform both variable selection and clustering of the categories of variables. Using the same notation as we did for the Lasso, the Fused Lasso applies an L1 penalty to the differences between subsequent coefficients

$$g_{fLasso}(\boldsymbol{\beta}_j) = \sum_{i=2}^{p_j} w_{j,i-1}|\beta_{j,i} - \beta_{j,i-1}|, \qquad (2.12)$$

so that consecutive levels within predictors may be fused. Because the Fused Lasso only regularizes differences, the predictor being penalized needs to have a reference level for the penalty to work as a variable selection tool. The coefficient corresponding to the level which is fused with the reference level is then set to zero.

For high values of λ, the differences between all subsequent coefficients of a predictor become zero. All the levels will then be fused with the reference level, which is equal to the predictor being removed from the model. The Fused Lasso can thus also be used for variable selection (Devriendt et al., 2018).

2.5.3 Generalized Fused Lasso

The Fused Lasso is not suited for regularization of nominal predictors since there is no intrinsic ordering to their categories. Therefore, Bondell & Reich (2009) introduced a penalty for nominal variables that could perform factor selection and level fusion through *analysis of variance* (ANOVA). Gertheiss & Tutz (2010) later adapted the penalty to the regression setting. The penalty is expressed as

$$g_{gfLasso}(\boldsymbol{\beta}_j) = \sum_{i>l} w_{j,il}|\beta_{j,i} - \beta_{j,l}|, \qquad (2.13)$$

where the sum is over all coefficients i, $l \geq 0$. Not only differences $\beta_{j,i} - \beta_{j,i-1}$ are considered for penalization like for the Fused Lasso, but rather all differences $\beta_{j,i} - \beta_{j,l}$. The Generalized Fused Lasso thus penalizes the sum of the differences between the coefficients of all the levels within the predictor. Consequently, the Generalized Fused Lasso enforces the building of clusters of all levels that share the same effect, not just those who are in sequence. Similar to the Fused Lasso, a reference category is needed for the penalty to work as a variable selection tool (Devriendt et al., 2018). If a reference level is present, the Generalized Fused Lasso is suitable for both variable selection and fusion of levels.

2.6 Penalty weights

As we only apply one tuning parameter λ on all the sub-penalties, incorporating individual penalty weights w_j to each sub-penalty can improve their performance. In the datasets Gjensidige uses to create models to predict claim frequency, each level of a given predictor may have differently sized coefficients and a different amount of observations. Both adaptive penalty weights and standardization penalty weights have been proposed to account for these differences to improve performance (Devriendt et al., 2018).

Penalty name	$\boldsymbol{w}_j^{(ad)}$	$\boldsymbol{w}_j^{(st)}$
Lasso	$w_{j,i}^{(ad)} = \lvert\hat{\beta}_{j,i}\rvert^{-1}$	$w_{j,i}^{(st)} = 1$
Fused Lasso	$w_{j,i-1}^{(ad)} = \lvert\hat{\beta}_{j,i} - \hat{\beta}_{j,i-1}\rvert^{-1}$	$w_{j,i-1}^{(st)} = \sqrt{\frac{n_{j,i}+n_{j,i-1}}{n}}$
Generalized Fused Lasso	$w_{j,il}^{(ad)} = \lvert\hat{\beta}_{j,i} - \hat{\beta}_{j,l}\rvert^{-1}$	$w_{j,il}^{(st)} = (k_j+1)^{-1}\sqrt{\frac{n_{j,i}+n_{j,l}}{n}}$

Table 2.1 – Penalty weights

The adaptive (*ad)* weights are based on initial estimates of β, obtained from running an initial regression. Coefficients that are initially estimated as large could be in danger of being too heavily regularized, but by including the adaptive weights, coefficients that are initially estimated as small will be regularized relatively more than large ones (Devriendt et al., 2018). The weight for the Lasso penalty is defined as

$$w_{j,i}^{(ad)} = \lvert\hat{\beta}_{j,i}\rvert^{-\gamma}, \qquad (2.14)$$

where $\gamma > 0$ is a tuning parameter that both Gertheiss and Tutz (2010) and Devriendt et al. (2018) set equal to 1. They also adopt the adaptive weights formulated for each penalty listed in the table from Rinaldo (2009) and Viallon, Lambert-Lacriox, Höfling & Picard (2016). We have focused on how these articles contribute to the method through Devriendt et al. (2018), rather than understanding their theoretical background, which we consider outside the scope of this thesis.

When applying the Lasso penalty, the variables should be centered and standardized to account for the effect of different measuring scales of different predictors, which may lead to an uneven number of observations per level (Tibshirani, 1997). This is not possible for ordinal and nominal variables, as the levels would lose their interpretation (Devriendt et al., 2018). Therefore, Bondell and Reich (2009) and Gertheiss & Tutz (2010) proposed the following standardization (*st*) weight for ordinal variables

$$w_{j,i-1}^{(st)} = \sqrt{\frac{n_{j,i} + n_{j,i-1}}{n}}, \qquad (2.15)$$

which takes the number of observations of each level into account. The standardization penalty weights thus adjust for the imbalances resulting from some levels having more observations than others. To extend the standardization weight to nominal predictors, the amount of regularized differences for a nominal predictor relative to an ordinal one needs to be considered. Gertheiss and Tutz (2010) adapted the weight to the Generalized Fused Lasso by adding the factor $(k_j + 1)^{-1}$, where $k + 1$ is the number of levels of predictor j. Without this factor, the Generalized Fused Lasso would be artificially larger than the Fused Lasso, because while the Fused Lasso for a predictor of p levels only includes p-1 differences, the Generalized Fused Lasso includes the differences between all the different levels within the predictor. Therefore, nominal predictors would likely be regularized to a greater extent than ordinal predictors without this factor.

As shown in Equation 2.16, by multiplying the two weights and using a combination of them, it is possible to apply the objectives of both weights.

$$\boldsymbol{w}_j = \boldsymbol{w}_j^{(ad)} \cdot \boldsymbol{w}_j^{(st)} \qquad (2.16)$$

2.7 Resulting objective function

In this thesis we estimate a penalized Poisson regression model to predict claim frequency while selecting and fusing predictor levels to increase prediction accuracy. We use the scaled negative log-likelihood as our loss function while including exposure as an offset. Since the dataset includes binary, nominal, binned continuous and ordinal predictors, we use the regularization framework developed by Gertheiss and Tutz (2010) to penalize different predictor types.

Our objective function's second term therefore consists of sub-penalties adapted to each predictor type, stating how each predictor should be penalized. We apply the Lasso to binary predictors, the Fused Lasso to ordinal and binned continuous predictors and the Generalized Fused Lasso to nominal predictors.

Through combining the chosen loss function, offset, penalty types and penalty weights, we end up with minimizing the objective function

$$\mathcal{O}(\boldsymbol{\beta}; \boldsymbol{X}, \boldsymbol{y}) = -\frac{1}{n}\sum_{i=1}^{n}(y_i(x_i\beta + \log(expo_i)) - e^{(x_i\beta + \log(expo_i))} - \log(y_i!)) + \lambda\left(\sum_{j\epsilon bin}\sum_{i=1}^{p_j} w_{j,i}|\beta_{j,i}| + \sum_{j\epsilon ord}\sum_{i=2}^{p_j} w_{j,i-1}|\beta_{j,i} - \beta_{j,i-1}| + \sum_{j\epsilon nom}\sum_{i>l} w_{j,il}|\beta_{j,i} - \beta_{j,l}|\right), \quad (2.17)$$

where we use the combined penalty weights $\boldsymbol{w}_j = \boldsymbol{w}_j^{(ad)} \cdot \boldsymbol{w}_j^{(st)}$. The combined penalty weights are products of the adaptive and standardization weights and take both different sizes of coefficients and different amounts of observations per level into account. In the simulation study of Devriendt et al. (2018), the combined weights perform the best in terms of prediction accuracy.

After the objective function is developed, the next stage is to identify a suitable estimation procedure.

2.8 Optimization

Traditionally, *least angle regression* (LARS) (Efron, Hastie, Johnstone & Tibshirani, 2004) has been the most common estimation procedure used for Lasso-type penalties. Starting with all coefficients set to zero, the algorithm works similarly to *Forward Stepwise Selection* and iteratively searches for the predictor with the highest correlation with the dependent variable, increasing its coefficient. The process is repeated until all predictors are included in the model (Tibshirani, 2003). However, even though it is well suited for estimation using only one predictor type and thereby one type of penalty term, it cannot be used when working with a dataset with several predictor types and a corresponding number of penalty terms (Devriendt et al., 2018).

Oelker & Tutz (2017) used local quadratic approximations of the penalties to be able to apply the *Penalized Iteratively Reweighted Least Squares* (PIRLS) algorithm in a regularization setting. The procedure can be used for datasets with different types of predictors with corresponding penalty terms. However, using local quadratic approximations of the penalties leads to non-exact collapsing and selection of levels, and as the PIRLS algorithm requires creating large matrix inverses, the procedure is computationally intensive (Devriendt et al., 2018).

Devriendt et al. (2018) introduced the *Sparse Multitype Regularized Feature* (SMuRF) algorithm, which applies the theory of proximal operators on Lasso-type penalties, which was first done by Beck & Teboulle (2009) and Xin, Kawahara, Wang & Gao (2014). By using proximal operators, the algorithm can solve the subproblems per penalty type exactly, instead of using approximations. As the algorithm creates a set of smaller subproblems to be optimized, it can also use parallel computing. Since the SMuRF algorithm works for several penalty types, does not use any approximations and can be used for large datasets, we elect to use it for optimization. In the following sections, we will describe the SMuRF method, including how it tunes the λ parameter and reestimates the model.

2.8.1 The SMuRF algorithm

The SMuRF algorithm uses a gradient descent approach to minimize the objective function (Devriendt et al., 2018). The gradient of a function *f* with several variables is a vector of the partial derivatives of the function with respect to all the variables (Sydsæter, Seierstad & Strøm, 2002). For a specific point, the gradient is the slope of the function. The gradient descent approach uses the gradient to find the minimum of the function (Donges, 2018).

Figure 2.1 lists the steps taken for each iteration of the algorithm. Prior to these steps, the parameter vector $\beta^{(0)}$ is filled with initial random parameter estimates. Together with the predictors, the dependent variable, the chosen value of lambda, and a step size *s*, $\beta^{(0)}$ is used as input to the algorithm. For each of the *m* iterations of the algorithm, the estimates of the parameter vector are updated, and the new estimates are based on the step size *s* and the gradient, which together signals the length and direction of our steps. For each iteration we get closer to the minimum, as the gradient assigns the direction of the steepest descent (Donges, 2018).

Naïve SMuRF algorithm	
1: Input: $\beta^{(0)}$, X, y, s, lambda	
2: **For** k =1 ... M do:	
3: $\tilde{\beta} \leftarrow \beta^{(k-1)} - s\nabla f(\beta^{(k-1)})$	Gradient update
4:$(\tilde{\beta}_0, \tilde{\beta}_1, \dots, \tilde{\beta}_J) \leftarrow \tilde{\beta}$	Partition full vector in components for each predictor
5:$\beta_j^{(k)} \leftarrow prox_{s\lambda g_j}(\tilde{\beta}_j)$	Calculate the Proximal Operator for all predictors j in {0, ... ,J}
6:$\beta^{(k)} \leftarrow (\beta_0^{(k)}, \beta_1^{(k)}, \dots, \beta_J^{(k)})$	Recombine to full vector
7: **End for**	
8: Return $\beta^{(m)}$	

Figure 2.1 – The steps of the SMuRF algorithm in its naïve form

Each time the parameter estimates are updated, the parameter vector is partitioned into separate components for each predictor, because the proximal operator (PO) will be calculated and solved separately for each predictor. Generally, POs are used for approximating a value, with the combined goal of approximating it accurately and minimizing a cost associated with the chosen value (Devriendt et al., 2018). In our case, we want to estimate the coefficients that minimize the differences between the observed and predicted values of the dependent variable, number of claims.

However, included in the PO is also the cost associated with choosing a specific coefficient value, where the cost is adapted to the different predictor types through the penalty terms. This way, the SMuRF algorithm is able to take both prediction accuracy and the regularization terms into account during the estimation procedure. When the POs have been calculated, the coefficient estimates are again combined to a full vector, and the algorithm returns the coefficient estimates for the given iteration.

The algorithm can be implemented in R using the *smurf* package (Reynkens, Devriendt & Antonio, 2018). Figure 2.1 illustrates the steps of the algorithm in its naïve form. The version of the algorithm included in the *smurf* package has been improved in terms of computational efficiency. When the number of iterations *k* approaches infinity, the algorithm converges to the optimal solution (Devriendt et al., 2018).

2.8.2 Tuning and reestimation

The objective function

$$\mathcal{O}(\boldsymbol{\beta};\boldsymbol{X},\boldsymbol{y}) = -\frac{1}{n}\sum_{i=1}^{n}(y_i(x_i\beta + \log(expo_i)) - e^{(x_i\beta+\log(expo_i))} - \log(y_i!)) + \lambda\left(\sum_{j\epsilon bin}\sum_{i=1}^{p_j} w_{j,i}|\beta_{j,i}| + \sum_{j\epsilon ord}\sum_{i=2}^{p_j} w_{j,i-1}|\beta_{j,i} - \beta_{j,i-1}| + \sum_{j\epsilon nom}\sum_{i>l} w_{j,il}|\beta_{j,i} - \beta_{j,l}|\right) \quad (2.18)$$

includes a tuning parameter λ, which determines the relative strength of the penalties compared to model fit. In contrast to the model's coefficient estimates, which are learned by the model, λ must be chosen prior to estimation. It is difficult to identify a suitable value for λ before estimation, but the *smurf* package (Reynkens et al., 2018) allows it to be selected by evaluating performance criteria for in-sample training, out-of-sample training on a test dataset or stratified K-fold cross validation. Stratified K-fold cross validation with deviance as criterion using the one standard error rule performs the best in Devriendt et al. (2018), and is therefore our chosen method for selecting λ.

Cross-validation can be used to evaluate a range of different values for λ based on some performance criterion, which will indicate the optimal value for λ (Hastie et al., 2017). When using K-fold cross validation, the dataset is divided into *K folds* or groups of observations, of similar size. The partitioning creates stratified folds, meaning that the mean of the dependent variable is approximately equal in all folds. *K-1* of these folds are then used to estimate the model. The fold of observations that is not included is used as a validation group to calculate an estimate of the average deviance (Devriendt et al., 2018). The deviance is a performance criterion which is calculated as negative two times the maximum log-likelihood, where a smaller deviance indicates a better model fit.

The procedure is completed *K* times, as each fold of observations is used as the validation fold once, giving us *K* estimates of the average deviance as a function of λ. As we use the "one standard error rule", we select the highest λ where the average deviance is within one standard error of its minimum. This will result in the simplest model which is within one standard error of the minimum average deviance (Hastie et al., 2015).

Compared to non-penalized models, regularization methods like SMuRF normally return models with decreased variance in the estimates and predictions at the expense of an increase

in bias. It is therefore common to reestimate the model without penalties, where the coefficients from the regularized model estimation are used to reduce this newly introduced bias (Devriendt et al., 2018). The variables of which the coefficients are estimated to zero are removed, and levels which were fused by the regularization procedure are collapsed. The reestimated model will therefore have the same non-zero and fused coefficients as the original model, but the results will not be biased (Devriendt et al., 2018). To perform this reestimation, we use the R package *smurf* (Reynkens et al., 2018).

2.9 Model validation

To compare the model we create to the reference model Gjensidige already uses, the models have to be validated using performance measures. Several approaches can be used to estimate performance, depending on the type of dataset and performance criteria used for measurement. For Gjensidige, the goal is to earn a profit from the insurance policies they offer. For this to happen, it is important that the premiums they charge are at a competitive level, while still covering the losses they have on their client's claims.

As part of calculating these premiums, Gjensidige must predict the number of claims from each person as accurately as possible. Consequently, for model validation we use criteria which can measure this prediction accuracy, as this will uncover whether our method can help Gjensidige earn a higher profit. In the next sections we will describe the type of dataset and performance criteria we will use to compare the models.

2.9.1 **Validation dataset**

The prediction accuracy of a model is highly sensitive to the dataset used for validation. The preferred method to use for validating a model is to separate the data by creating a training set used for model estimation and a test set only used for validation. This test set can be created several different ways. When predicting future observations, an out-of-time test set is suitable. Creating an out-of-time test set means the dataset is divided based on the time of the observation, for example the day, month or year.

An alternative method is to create the test set by sampling randomly from the full dataset (Johnson & Kuhn, 2013). If interested in predictions in the same population of policyholders, the alternative method would be most suitable. However, as the aim of the

model created in this thesis is to predict the number of claims registered in a year from future policyholders, it is important the model translates well to the future. By training the model using data from previous years and validating using data from a later year, we can replicate how the model will be used by Gjensidige.

When splitting the data, it is also important to have enough observations to properly train and test the model. If the dataset used for modeling is small, the decisions regarding the split of the data are critical. However, when using a larger dataset, there will be enough observations to split the data without the results changing excessively. Therefore, there will not be a need to use resampling techniques to validate the model (Johnson & Kuhn, 2013).

2.9.2 **MSE**

To validate the estimated models, we measure how well the predicted number of claims fit the observed data (Hastie et al., 2017). For regression models, the mean-squared error (MSE) is the most commonly-used measure

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2, \qquad (2.19)$$

where n is the number of observations, y_i is the actual value of the ith observation, and $f(x_i)$ is the prediction that f gives the ith observation. The MSE will be small if the predicted values are close to the observed data (Hastie et al., 2017). Therefore, the model with the lowest MSE is usually preferred.

The quality of fit of a model can either be measured using the training dataset used for model estimation or a separate test dataset. Using the training set will produce the training MSE, as in Equation 2.19. If more parameters are included in the model, increasing its flexibility and allowing it to search for more patterns in the data, the training MSE will decrease because the model can explain a greater part of the variation in the training set. However, the test MSE will not necessarily decrease as much as the training MSE. If this is the case, the model is said to be *overfitted*, as it seems to have been fitted to noise in the training set (Hastie et al., 2017). Therefore, the training error is an underestimation of the test error, and the test MSE is more interesting to us (Hastie et al., 2017)

$$Ave\left(y_0 - \hat{f}(x_0)\right)^2. \qquad (2.20)$$

The observation (x_0, y_0) is a previously unseen test observation, and Equation 2.20 thus calculates the average squared prediction for the observations. When using test MSE as the performance criterion, the model which minimizes Equation 2.20 will be selected. To be able to minimize the equation, it is valuable to understand how it is composed. It can be shown that the expected test MSE is the sum of the variance of $\hat{f}(x_0)$, the squared bias of $\hat{f}(x_0)$ and the variance of the error term E (Hastie et al., 2017)

$$E\left(y_0 - \hat{f}(x_0)\right)^2 = Var\left(\hat{f}(x_0)\right) + \left[Bias\left(\hat{f}(x_0)\right)\right]^2 + Var(\in). \qquad (2.21)$$

To minimize the expected error, we therefore wish to minimize the variance and bias of the model simultaneously. However, since there is a trade-off between the variance and bias of the model, it is the relative change of the variance and bias which decides if the test MSE increases or decreases (Hastie et al., 2017).

To calculate the test MSE, a designated test set has to be available. If not, other approaches have to be taken to estimate the test MSE, like the validation-set approach or cross-validation (Hastie et al., 2017). A downside of using test MSE as the only performance criterion is that it does not consider that models have different amounts of predictors. Therefore, we also use AIC.

2.9.3 AIC

When considering two models with the same test MSE, the one with the fewest predictors will always be preferred (Johnson & Kuhn, 2013). However, using MSE to choose between models with different amounts of parameters is not optimal. Adding additional predictors to a model will generally return a lower test MSE, but the increase in predictors may be higher than the relative gain of a lower test MSE (Hastie et al., 2017). The Akaike Information Criterion (Akaike, 1974) is a performance measurement which was created to address this issue. We seek to minimize

$$AIC = -2\log(L) + 2K = Deviance + 2K, \qquad (2.22)$$

where L represents the likelihood function and K is the number of parameters used in the model. When the number of parameters increases, the first term decreases the AIC, while the second term increases the AIC. Therefore, there is a trade-off when increasing the number of

parameters, as the second term penalizes a more flexible model. Consequently, it is a suitable measure when comparing models which have a different number of parameters.

3. Dataset

To find an improved way for Gjensidige to group and bin variables used in their predictive models, we study Gjensidige's insurance product *comprehensive motor vehicle coverage*. The predictive models built to study our research question are therefore trained using data on the policyholders of this product. The same dataset was used to train Gjensidige's current frequency model for this insurance product, which enables us to evaluate the effect on prediction accuracy of our model. In this chapter we will describe the dataset, present descriptive statistics of some of its variables, consider the quality of the data and present how we process the data prior to modeling.

The dataset spans the time period from 2012 to 2017 and consists of 8 446 547 observations and 31 variables. Most of the variables provide information on the policyholders' personal characteristics, information about their insurance policy or the attributes of their vehicle. A variable signaling each policyholder's *exposure* (*Ekspo_faktor)* is also included in the dataset. *Ekspo_faktor* indicates the proportion of the year the policyholder was insured and, in that sense, *exposed* Gjensidige to the risk that they would make a claim.

The R packages *ggplot2* (Wickham, 2016) and *graphics* (R Core Team, 2018) have been used to create the plots we present in this chapter.

3.1 Delimitation

The variables that are not used in Gjensidige's model have been excluded from our model as well. The same 14 variables have therefore been used as the basis to create the predictors included in both models, and potential variable selection is only performed among those 14 variables. In addition, most data pre-processing performed besides fusing variables has been equal for both models. This includes the removal of NA's and aggregation of the dataset, further described in 3.5. Using the same variables as a basis and doing the same data pre-processing is a way of validating comparisons between the models' performance, in addition to limiting the scope of the thesis.

However, we have also performed some separate data pre-processing for the two models. To be able to let the SMuRF algorithm perform the binning of continuous variables in a data-

driven way, some manual data pre-processing has been performed to prepare the variables. How this differentiates between the two models is further described in 3.6.

3.2 Variables

Table 3.1 lists the dependent variable, exposure variable and independent variables used for modeling in this thesis. Out of the 14 independent variables, two are ordinal, seven are nominal, one is binary, and four are continuous. Overall, the variables describe traits of the policyholder and the vehicle, the policyholder's use of the vehicle, and geographical factors.

Type	Name	Description
Dependent	*Claims*	Observed number of claims of the policyholder: 0-5
Exposure	*Ekspo_faktor*	Fraction of the year policy was active: 0-1
Ordinal	*Kjor_lengde_kode*	Code for distance driven: 13 levels
	Drivstoff_kode	Code for type of fuel: 13 levels
Nominal	*Merke_klasse_kode_ny*	Code for vehicle brands collected from the Norwegian Register of Motor Vehicles: 45 levels
	Ekspo_aar	The year the policy was valid: 2012-2017
	Subcluster	Gjensidige's own code for combination of geography and demography: 12 levels
	Band_AK_G	Gjensidige's own geographic variable: 26 levels
	Divisjon_kode	Where the policy was registered: 6 levels
	Leasing_flagg	Whether the insured vehicle is leased: Yes/No/Unknown
	Import_flagg	Whether the insured vehicle is imported: Yes/No/Unknown
Binary	*Forer_23_aar_flagg*	Indicates whether all drivers of the vehicle are over 23 years old: Yes/No
Continuous	*Effekt_HK*	Vehicle horsepower: 1 – 193 000
	MV_alder	Vehicle age: 0 – 97
	Alder_ftaker	Age of policyholder: 1 – 110
	Egenvekt	Weight of vehicle measured in kilograms: 1 – 99 805

Table 3.1 - Variable descriptions

3.3 Descriptive statistics

3.3.1 **Dependent variable**

In this thesis, we model the claim frequency of *comprehensive motor vehicle coverage*. The purpose is to predict the number of claims during the insured period of a future policyholder. The model's dependent variable is *Claims*, which is the number of claims reported by the policyholder throughout the year. *Claims* is a discrete variable, having only integer values. As shown in Table 3.2, most policyholders, 98.06%, do not submit a claim during their policy period. A few, 1.86%, submit one claim, while the remaining minority, 0.08%, submit between two and five claims.

Number of Claims	Number of Policyholders	Proportion of Policyholders
0	8 282 630	98.06%
1	157 292	1.86%
2	6310	0.075%
3	287	0.0034%
4	25	0.0003%
5	3	0.00004%

Table 3.2 - Number of policyholders for each number of claims

3.3.2 **Exposure**

Ekspo_faktor measures the proportion of the year that the observed policyholder was insured, ranging from 0 to 1. The mean exposure is 0.34 and the average duration of a policy is therefore slightly longer than four months. The average duration of a policy seems low, but the explanation can be found in how the dataset is structured. An observation, or a row, in the dataset represents an individual. As time passes, its features will change, and probably not simultaneously. A change in just one of these variables will generate a new row in the dataset with the updated information on the individual.

In that sense, the exposure indicates how long Gjensidige was exposed to the risk of that one individual while its features remained the same. The distribution of *Ekspo_faktor* in average for a year with months as intervals is shown in Figure 3.1.

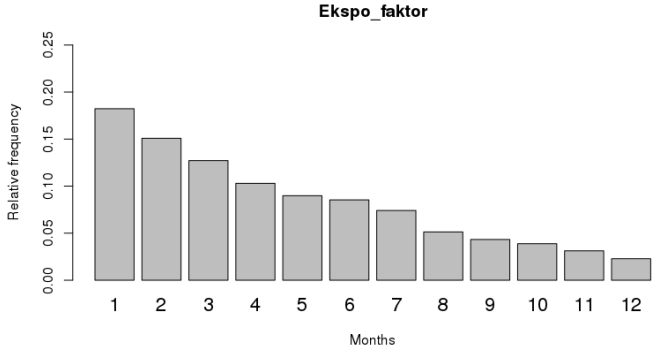


Figure 3.1 – Distribution of exposure

The mean number of claims in the dataset is approximately 0.02. The mean of *Claims* is based on the mean exposure of the dataset which is 0.35. It does not necessarily indicate that an average policyholder files 0.02 claims a year, but rather indicates that the average policyholder would file 0.02 claims during their average exposure of 4 months, equaling about 0.05 over a year. In this chapter, we will use the average claim number as a reference with the purpose to analyze how each variable, and each level within it, affects the mean. We have been informed by representatives from Gjensidige that the real average exposure is between 0.7 and 0.9 for different products. This means that policyholders on average are insured for periods of eight to eleven months, approximately.

In the following section, we will present the predictors which we believe have an interesting relationship with claims. Plots of the rest of the predictors can be found in Appendix A1 and A2.

3.3.3 Independent variables

Missing values

There are some variable values in the dataset which are 0, -1 or -2, indicating missing values. *Band_AK_G* and *Subcluster* have about 13 percent missing values out of the total number of observations, while *Divisjon_kode* has about 10 percent. Besides these three variables, the variable with the most missing values is *MV_alder* with about 2 percent. How these missing values are handled is explained in 3.5. In addition to the mentioned missing values, some of the variables have values which we consider odd, for example when horsepower is 1 or 193 000. Whether these represent missing values or not is unknown.

Kjor_lengde_kode: Mileage

In the dataset, mileage is divided into intervals that are each given a code. As seen from the left-hand plot of Figure 3.2, most of the vehicles have codes between 005 and 030, with the most common code being 012. In Figure 3.2 we plot the mileage codes against the mean number of claims. It seems that the mean number of claims increases with mileage, but the standard errors also increase with mileage. The most common mileage codes have a mean number of claims close to the overall mean of the dataset (0.02), while mileage codes of higher value than these seem to indicate an above average number of claims.

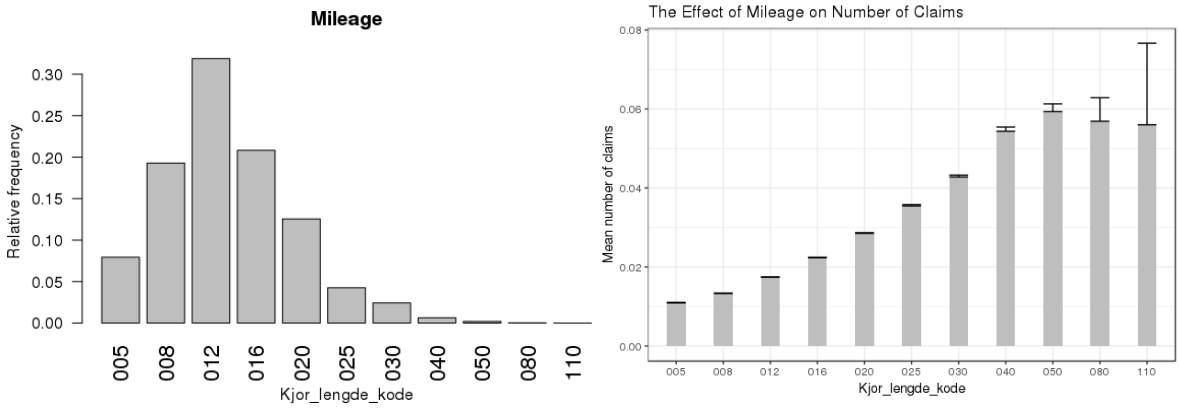


Figure 3.2 - Relative frequency and mean number of claims for mileage

Drivstoff_kode: Fuel code

The different types of fuel are also divided into different codes, and we observe that approximately 95 percent of the observations have fuel code 001 or 002 as seen in Figure 3.3. We observe that the codes with the highest relative frequency of policyholders have the lowest standard error, while it seems that for four of the codes, there are no claims among all policyholders with that particular type of fuel.

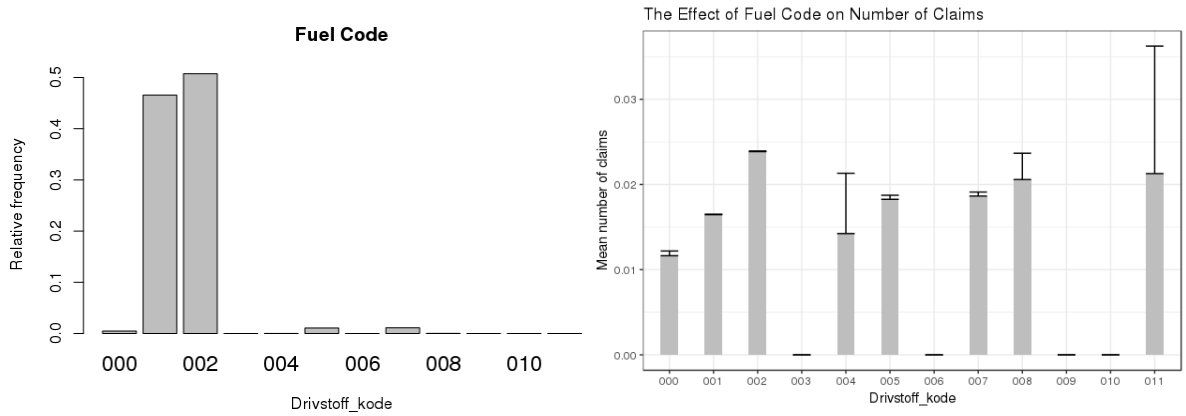


Figure 3.3 - Relative frequency and mean number of claims for fuel type

Merke_klasse_kode_ny: Vehicle brand

We observe in Figure 3.4 that the number of vehicles insured within each brand differ a fair amount, and again we see that the categories with many observations have smaller standard errors. We observe that the average number of claims differ between different vehicle brands, indicating that the variable could be a good predictor.

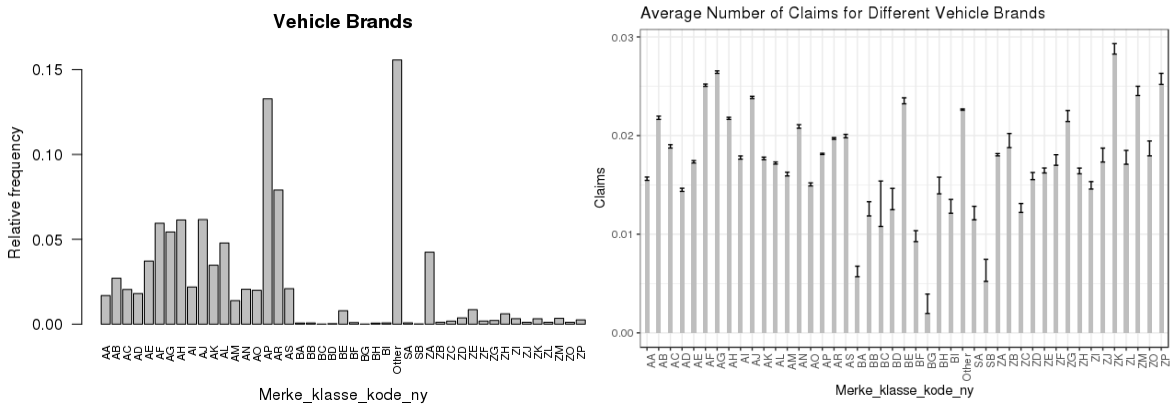


Figure 3.4 - Relative frequency and mean number of claims for vehicle brand

Band_AK_G: Geographic variable

In Figure 3.5, we observe that the relative frequency of *Band_AK_G* is quite evenly distributed over the different categories, and the standard errors of the calculated means are small. There is a close to linear trend for each category's effect on the mean, which is surprising for a nominal variable. This variable was created by Gjensidige themselves, and so it seems they may have sorted the categories according to their effect on the dependent variable.

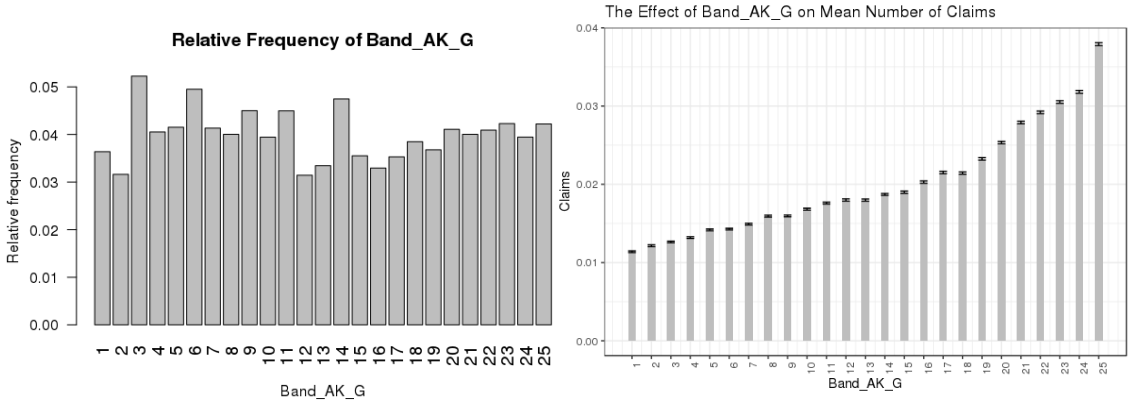


Figure 3.5 - Relative frequency and mean number of claims for Band_AK_G

Effekt_HK: Horsepower

Figure 3.6 shows how the average horsepower of the vehicles of the policyholders differs depending on how many claims they have filed. The figures show that the vehicles of policyholders who have submitted claims have lower horsepower on average than the policyholders who have not submitted claims. In 2016, the average car in Norway had 133 horsepower (Korsvoll, 2016). From the figure we can therefore observe that the policyholders that have filed at least one claim on average own cars with average horsepower, while the policyholders who have not filed claims, own vehicles with horsepower above average. This indicates that horsepower could be a useful variable to include when predicting claims. Overall, the standard errors are small, indicating that the sample mean is reliable. We do however observe larger standard errors for the observations with no claims and the observations with a high number of claims.

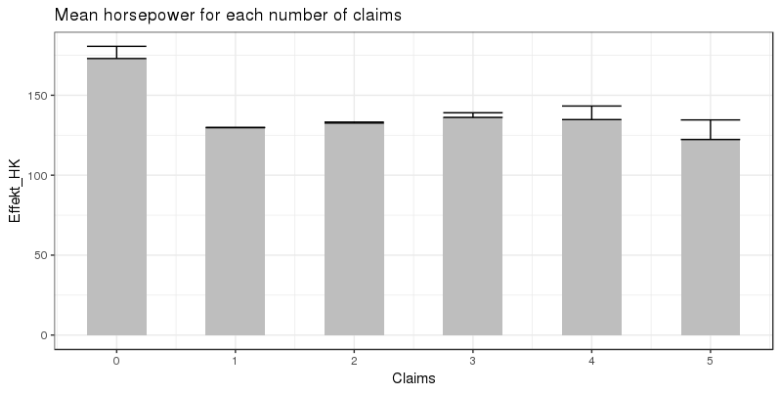


Figure 3.6 - Mean horsepower for each number of claims

MV_alder: Age of vehicle

Figure 3.7 shows the mean and standard error of *Claims* for different ages of the vehicles that are insured by the policyholders. The line represents the mean values, while the grey ribbon surrounding it represents the standard errors of the mean of *Claims* for each age. The plot shows that the mean number of claims increases with age and reaches its peak for vehicles that are four years old. After that, the average number of claims decreases steadily with age.

After 25 years the standard errors increase. This is probably due to fewer observations, as there are not many vehicles that are that old, and these old vehicles are probably not insured with the product *comprehensive motor vehicle coverage*, which is expensive and more suitable for new vehicles. There are also some very old vehicles which mostly have zero claims. These kinds of vehicles are probably well maintained veteran vehicles that are treated carefully and seldom driven.

The plot on the left shows why the standard error of the mean of older vehicles are higher, as there are relatively few policyholders who have vehicles which are older than 30 years.

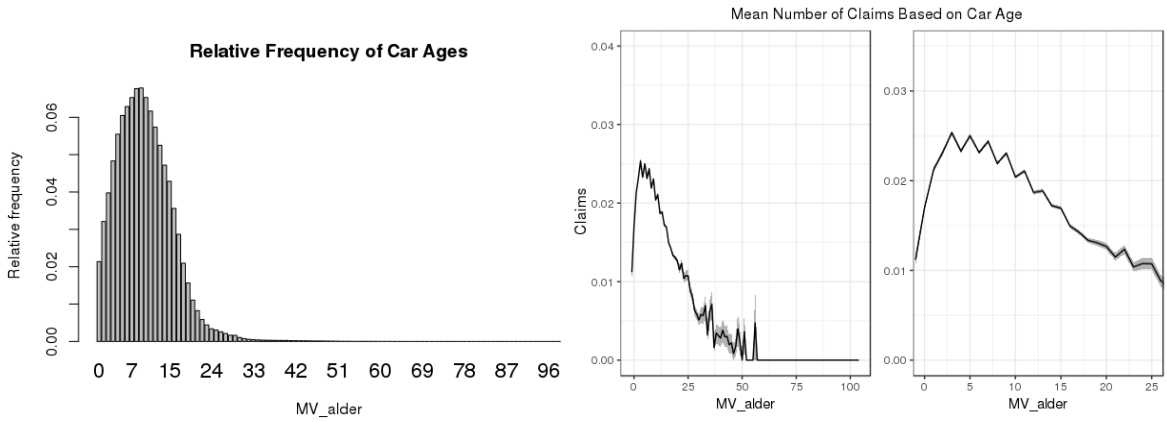


Figure 3.7 - Relative frequency and mean number of claims for vehicle age

3.4 Data quality

Gjensidige possesses a large data warehouse with information on the policyholders of all their insurance products. When building models for predicting claim frequency, claim severity, or the risk premium directly, data on the policyholders of the relevant product and the variables desired is retrieved from the data warehouse and imported into R. As we have received the data directly from Gjensidige and their data warehouse, the data quality should

be high. It is in their interest to use data that reflects the attributes of their policyholders as closely as possible, to ensure their models are of high quality. At the same time, there is some potential of error in the data as much of the data is collected by Gjensidige's employees and therefore prone to human mistakes.

Even though the dataset from Gjensidige is of high quality, we perform some data pre-processing before modeling. The pre-processing is divided between the pre-processing which is common for both models and the pre-processing which differs between them. The pre-processing which is common for both models is the aggregation of the dataset and recoding of NA's, while the difference is in how continuous variables are binned and ordinal variables are grouped. In the next two sections we will describe how this pre-processing is performed. We primarily use the R package *dplyr* (Wickham, François, Henry & Müller, 2018) for data manipulation.

3.5 Common pre-processing

To increase computational efficiency, the dataset is aggregated prior to modeling. Aggregating the dataset means combining observations that have the exact same values for the predictors into one. All values of the predictors are kept the same, while their number of claims and exposure are added together. The effect on the dependent variable will be exactly the same as if keeping the observations separate, as the combined observation will have a higher number of claims and exposure than the observations had separately beforehand. This is common practice in the insurance industry, since datasets are often quite large and increasing computational efficiency is a priority. After aggregating the dataset, the number of observations is reduced from 8 446 547 to 6 726 978.

Concerning the odd values we have found for some variables in the dataset, we do not perform any pre-processing because Gjensidige does not and we wish to ensure model similarity. However, in addition to aggregating the dataset, we recode observations that include NA's. We recode NA's to −1, because this is Gjensidige's current approach for treating missing data and we want to ensure our model is similar to the one currently used. The reason Gjensidige recodes missing values to −1 is to avoid deleting observations. Deleting observations removes information from the model, and should be avoided if possible (Goldburd et al., 2016). These recoded observations are combined with other

observations into groups at a later point, and it is therefore not important that their value is –1.

A better solution would perhaps be to impute values using information from the rest of the predictors. This is done by creating a second model using the predictor with missing values as the dependent variable. A subset of the data which includes the other predictors and only observations without missing data is then used to train the model (Goldburd et al., 2016). This would lessen the information lost by simply recoding the missing value field to –1. Nevertheless, we use the same procedure as Gjensidige to ensure model similarity. The consequence of this may be that observations end up in the wrong group and therefore increase the bias of predictions.

The two variables *Leasing_flagg* and *Import_flagg* are originally nominal with three levels representing "yes", "no" and "unknown". For both variables, the relative frequency of "unknown" is very low, and we would therefore like to convert these variables into binary variables. We do not want to delete the observations that include this level, but rather impute them. For both variables, it is reasonable to believe that they would be registered as "yes" if they were leased or imported, while it may be forgotten if they are not, as this is the most common. Therefore, all observations of "unknown" are changed to "no" for both variables.

3.6 Differentiated pre-processing

To answer our research question, we want to find out whether going from manually grouping and binning variables to using regularization methods to do it can improve the prediction accuracy of the model. This section covers what binning is, how Gjensidige groups and bins their variables today, and lastly how we prepare the variables for using the regularization methods for fusion. Even though the purpose of regularizing the model is to let the method handle the variables, some pre-processing of the variables is done before using the method.

3.6.1 **Binning**

As part of the data pre-processing, Gjensidige manually *bins* their continuous variables into categorical variables before using them as predictors in their frequency model. Consequently, a coefficient is estimated for each bin, which applies to all observations falling within it. Manual *binning* refers to the pre-categorization of data into two or more

bins, which simplifies the dataset and increases interpretability (Johnson & Kuhn, 2013). In addition, it enables the model to capture non-linear effects which would not be possible if the variable was kept continuous. By binning the variable, the model is freed of needing to constrain its assumed relationship with the dependent variable to any particular shape, enabling it to capture non-linear effects it otherwise would not (Goldburd et al., 2016).

However, there are also several drawbacks to manual binning. As each interval has its own coefficient, the estimates will not behave in a continuous fashion, meaning some estimates may be inconsistent with others due to random noise. Also, variation within each bin is ignored, which means there is a loss of information that could have been used to fit the model. A possible solution is to separate the data into even more bins but doing so will reduce the credibility of each estimate (Goldburd et al., 2016). Lastly, dividing the bins manually makes it very difficult to find the optimal bins to maximize prediction accuracy. Many variables must be evaluated simultaneously, which is difficult to do manually (Johnson & Kuhn, 2013).

3.6.2 Gjensidige's method of binning and grouping variables

The reference model predicts claim frequency based on 13 categorical predictors. In this section, we describe how the 14 variables listed in table 3.1 are used to create these 13 predictors. The dataset includes ordinal, nominal, binary and continuous variables, and the variable types are treated differently in preparation for model estimation. Continuous variables are manually binned, converting them to ordinal variables, where the bins are equivalent of levels of ordinal variables. The intervals of values that are used as limits for the bins are not necessarily of equal size, and the limits have been set manually based on observation of the data. For example, the continuous variable *Egenvekt* is divided into 17 bins and called *Egenvekt_gruppe*, converting it into an ordinal predictor with 17 levels. Three of the four continuous variables, are simply binned and used as ordinal predictors in the model, creating four of the 13 predictors.

Original variable	Predictor	Number of bins/levels
Effekt_HK	*Effekt_HK_gruppe*	25
Egenvekt	*Egenvekt_gruppe*	17
MV_alder	*MV_alder_gruppe*	24

Table 3.3 - Number of levels after binning

The last continuous variable, *Alder_ftaker*, is also binned manually first. However, it is its interaction with the binary variable *Forer_23_aar_flagg* which is used as a predictor for the model. In addition to the binning of the continuous variables, one ordinal (*Kjor_lengde_kode*) and two nominal variables (*Drivstoff_kode and Divisjon_kode)* are grouped as shown in Table 3.4.

Original variable	**Levels**	**Predictor**	**Levels**
Kjor_lengde_kode	13	*Kjor_lengde_kode_gruppe*	10
Divisjon_kode	6	*Divisjon_kode_gruppe*	3
Drivstoff_kode	13	*Drivstoff_kode_gruppe*	2

Table 3.4 - Number of levels before and after grouping

These groupings are also performed manually, and the background for why the particular levels are fused together is unknown, as the groupings were done many years ago. The rest of the variables in the dataset are not treated in any way and are included in the model in their original form.

3.6.3 Preparing the variables for the SMuRF algorithm

One of the penalty types we use is the Fused Lasso, which is especially suited for ordinal predictors or continuous predictors that have been recoded as ordinal predictors to capture their non-linear effect. Therefore, to fuse and select levels of continuous variables, they firstly must be recoded to ordinal variables. This recoding should bin the continuous variable very crudely, so that an ordinal variable with many levels is created. This allows the SMuRF algorithm to choose between many levels to fuse, preserving more of the information the continuous variable originally provides. In the extreme example of only creating two bins, only two levels can potentially be fused, and too much information is lost from the continuous predictor. The cruder the bins are, the more similar it will be to fusing the original continuous predictor.

We have previously outlined the disadvantages of binning continuous variables, and with our chosen method we could re-evaluate Gjensidige's decision to perform binning, as it allows us to simply use the continuous variables instead of binning them into ordinal variables.

However, as our research question concerns comparing ways to group and bin variables, this assessment is outside the scope of this thesis. Nevertheless, the manual binning of variables performed for the regularized model is much cruder, as each variable is split into a larger number of intervals. For example, *Effekt_HK* is binned into 25 bins when used for Gjensidige's model, while it is divided into 52 bins for the regularization model.

We create many bins manually because we want the method to perform the binning for us having many bins to possible fuse. Binning the levels in a crude way gives the penalty term more levels to fuse, and therefore more influence on which ones should be fused together. Consequently, we have manually binned each of the continuous variables into ordinal variables crudely, before including them in the regularized model. The rest of the variables are left as they were in the original dataset, except that for each variable where zero has no meaning, any zero or negative values are gathered in a separate group, as they represent NAs.

4. Analysis

The research question of this thesis is: "Can the prediction accuracy of Gjensidige's claim frequency models be increased by using a data-driven method for the fusion of levels of categorical predictors?" To study this question, we analyze the data described in *Dataset* using the methods explained in *Methods*. In this chapter we estimate a reference model and a penalized model that can select and fuse predictor levels. In addition, as explained in *Methods* we reestimate a model with the new parameters to reduce the bias of the estimates, and then compare its results to the reference model.

To isolate the effect of fusing predictor levels using regularization, we strive to keep all other specifications identical for the models. Consequently, we use the same training dataset for model estimation and the same test dataset for model validation. To replicate how Gjensidige uses the predictive models to predict future policyholders' number of claims, we use an out-of-time test set with observations from 2017, while the training set consists of observations from 2012 - 2016. The test set contains 1 131 099 observations, which is approximately 20% of the training set's 5 595 879 observations.

4.1 Reference model

To evaluate the performance of the regularized model, we use Gjensidige's current model for predicting claim frequency of the policyholders of comprehensive motor vehicle insurance as a reference model. To ensure that the reference model we estimate is equal to Gjensidige's current model, we replicate how they manually bin and group variables as explained in *Gjensidige's method of binning and grouping variables*.

The resulting objective function to be minimized is the scaled negative Poisson log-likelihood with 13 predictors. For two of these 13 predictors, *Alder_ftaker* and *Forer_23_aar_flagg*, only the interaction between the two is used. In addition, the logged version of the variable representing exposure, *Ekspo_faktor*, is used as an offset in the model. We estimate the reference model by minimizing the objective function

$$\mathcal{O}(\boldsymbol{\beta}; \boldsymbol{X}, \boldsymbol{y}) = -\frac{1}{n}\sum_{i=1}^{n}(y_i(x_i\beta + \log(expo_i)) - e^{(x_i\beta + \log(expo_i))} - \log(y_i!)), \quad (4.1)$$

where y_i is the number of observed claims of a policyholder during the insured period $expo_i$. In section 4.3.1, the results of the reference model estimation are presented and compared to the results of the regularization method.

4.2 The regularized model

To fuse and select the levels of the variables used for predicting the number of claims, we estimate a regularized model. The model is based on the objective function from the reference model, but there are two important differences between the models. First, the variables have been treated differently prior to being included in the different models, as explained in *Differentiated pre-processing*.

Furthermore, a penalty term has been added to the objective function which creates the regularized model. The penalty term consists of sub-penalties adapted to each predictor type, stating how each predictor should be penalized. The same predictors which are used in the reference model are used as input for the regularized model, but as the regularization method performs variable selection, it may exclude some of the predictors in the process. Estimating the regularized model, we minimize the objective function

$$\mathcal{O}(\boldsymbol{\beta}; \boldsymbol{X}, \boldsymbol{y}) = -\frac{1}{n}\sum_{i=1}^{n}(y_i(x_i\beta + \log(expo_i)) - e^{(x_i\beta+\log(expo_i))} - \log(y_i!)) + \lambda\left(\sum_{j\epsilon bin}\sum_{i=1}^{p_j} w_{j,i}|\beta_{j,i}| + \sum_{j\epsilon ord}\sum_{i=2}^{p_j} w_{j,i-1}|\beta_{j,i} - \beta_{j,i-1}| + \sum_{j\epsilon nom}\sum_{i>l} w_{j,il}|\beta_{j,i} - \beta_{j,l}|\right). \quad (4.2)$$

Within the penalty term, we apply the penalty types that are most suitable to penalize the parameters of each predictor. Generally, we apply the Lasso for the binary predictors, and its generalizations Fused Lasso and Generalized Fused Lasso for the ordinal and nominal predictors respectively. The interaction between *Alder_ftaker and Forer_23_aar_flagg* is not penalized because the SMuRF method requires the predictors used for the interaction to be included without an interaction as well if they are to be penalized. Table 4.1 summarizes which penalty types are used for the different predictors.

Variable type	Predictor name	Penalty
Binary	*Import_flagg*	Lasso
	Leasing_flagg	
Ordinal	*Kjor_lengde_kode_gruppe*	Fused Lasso
	Effekt_HK_gruppe	
	Egenvekt_gruppe	
	MV_alder_gruppe	
Nominal	*Merke_klasse_kode_ny*	Generalized Fused Lasso
	Subcluster_gruppe	
	Divisjon_kode_gruppe	
	Band_AK_G_gruppe	
	Drivstoff_kode_gruppe	

Table 4.1 - Penalty type for each predictor

To regulate the relative importance of each penalty term, we apply combined penalty weights to each penalty term. To select and fuse predictor levels, the SMuRF algorithm is used to minimize the objective function of the regularized model, estimating the coefficients that will display which levels are selected and fused.

4.2.1 Selection of λ

Using the *smurf* package (Reynkens et al., 2018), we tune λ using 15-fold stratified cross-validation. The value of λ is selected using cross-validation, where *lowest deviance using the one standard error rule* is used as the criterion for selection.

Figure 4.1 was created using the *smurf* package (Reynkens et al., 2018) and shows how the deviance is changed while the logarithm of λ increases, indicating more heavily regularized models. The logarithm of λ is used to increase interpretability of the plot, as the relevant values are very small.

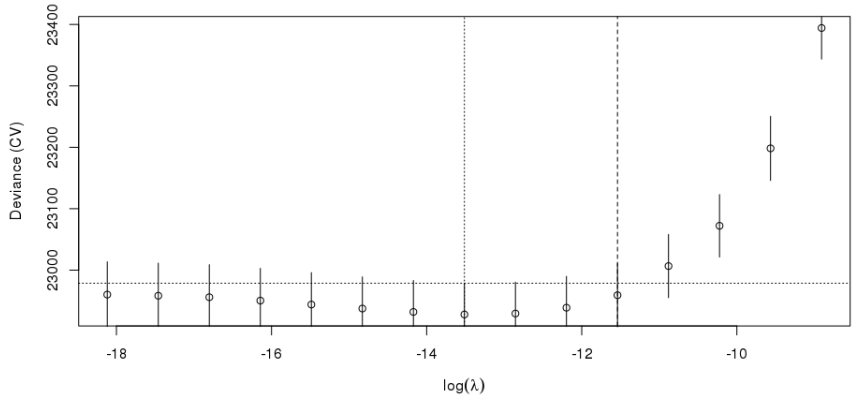


Figure 4.1 - Selection of λ

The most prominent stapled vertical line (on the right-hand side of the plot) indicates the chosen λ. The vertical stapled line located towards the center of the plot indicates the λ with the lowest deviance, while the shorter vertical lines represent the standard errors for the deviance of each λ. Even though the chosen λ did not produce the lowest deviance, it is chosen because it is the highest value of λ that still produces a deviance within one standard error of the λ with the lowest deviance. The figure also shows how increasing λ further would produce significantly worse results in terms of deviance, as it increases rapidly if λ is increased past a certain point. However, for lower values of λ, the deviance is low and stable as the penalization of the objective function is decreased.

4.2.2 The regularized predictors

In this section we will present plots of the estimated coefficients for selected predictors that we consider suitable for illustrating how the regularization method select and fuses levels. Similar plots for the remaining predictors are either shown in a later section or can be found in Appendix A5, while complete lists of all coefficients of the models are shown in Appendix A3 and A4. Levels with equal coefficient estimates are fused, while levels with coefficient estimates set to zero are removed, as they are fused with the reference category and consequently the intercept of the entire model. If all parameters of a predictor are set to zero, the variable would be removed from the model. However, in our case, the algorithm did not remove any of the variables. Still, as we will observe from the plots, the method has fused levels accounting for a reduction in parameters from 365 to 282.

The coefficient plots were created using the R package *ggplot2* (Wickham, 2016) and show the estimated coefficient for each level of each predictor for the reference model and the

regularized model. The standard errors of the estimates from the reference model are included as vertical shaded lines to illustrate which estimates have high variance. The standard errors are included because, as we will observe, they often affect the grouping of levels.

Ordinal predictors

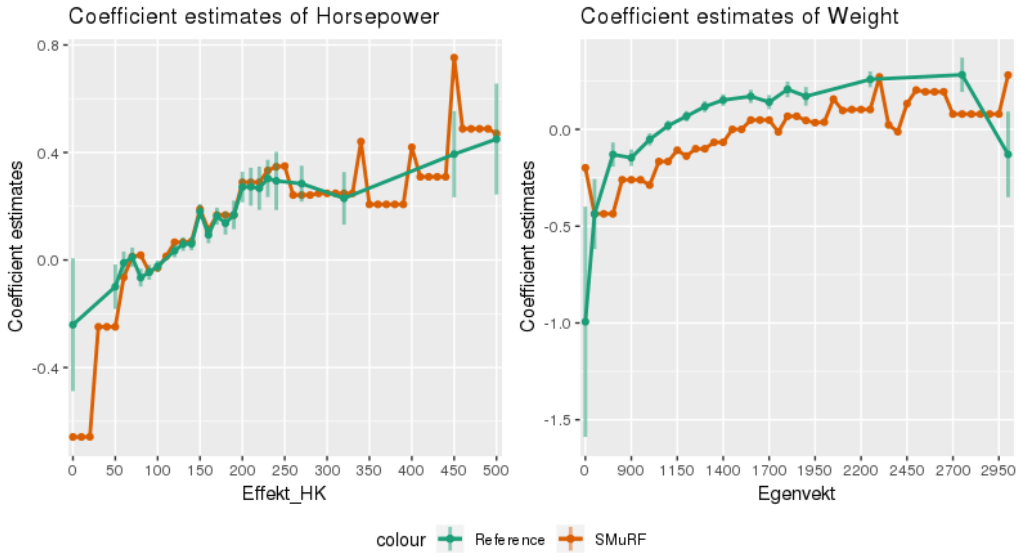


Figure 4.2 - Coefficient estimate for each level for the SMuRF model and the reference model for horsepower and weight

Figure 4.2 compares the coefficient estimates of the reference model and the SMuRF model for two of the ordinal predictors included in the model; horsepower and weight. The two plots illustrate how the number of parameters is reduced through the SMuRF estimation, as many coefficient estimates are equal, fusing the levels, and a few are zero, fusing them with the reference level. However, since horsepower and weight are ordinal predictors, only sequential levels are fused.

As the predictors are binned more crudely prior to the SMuRF estimation than the reference model, there is initially a higher number of coefficients to be estimated for the SMuRF model than the reference model. For example, 52 levels of horsepower were used as input for the SMuRF model, while 25 were used for the reference model. However, after using the SMuRF algorithm, the number of unique coefficients for horsepower is 28. The two models therefore end up with a similar number of levels, but the binning performed by the SMuRF model has been data-driven rather than performed manually.

The difference in binning is especially apparent on the right half of both plots, where there is a large difference in the number of levels for the two different models. We also observe that more of the levels on the right side of the plots have been fused than on the left, which indicates it was a correct decision by Gjensidige to create larger bins for the higher values. Still, the SMuRF model includes more of these levels than the reference model, which means that this data contains some information deemed valuable enough by the SMuRF algorithm to be signaled with several coefficients. The coefficient estimates of the two models are quite similar between about 75 and 325 of horsepower, while they differ more for most other values. In contrast, the coefficient estimates for weight are considerably different for the two models, as the reference model estimates a higher number of claims for most levels of the variable.

Nominal predictors

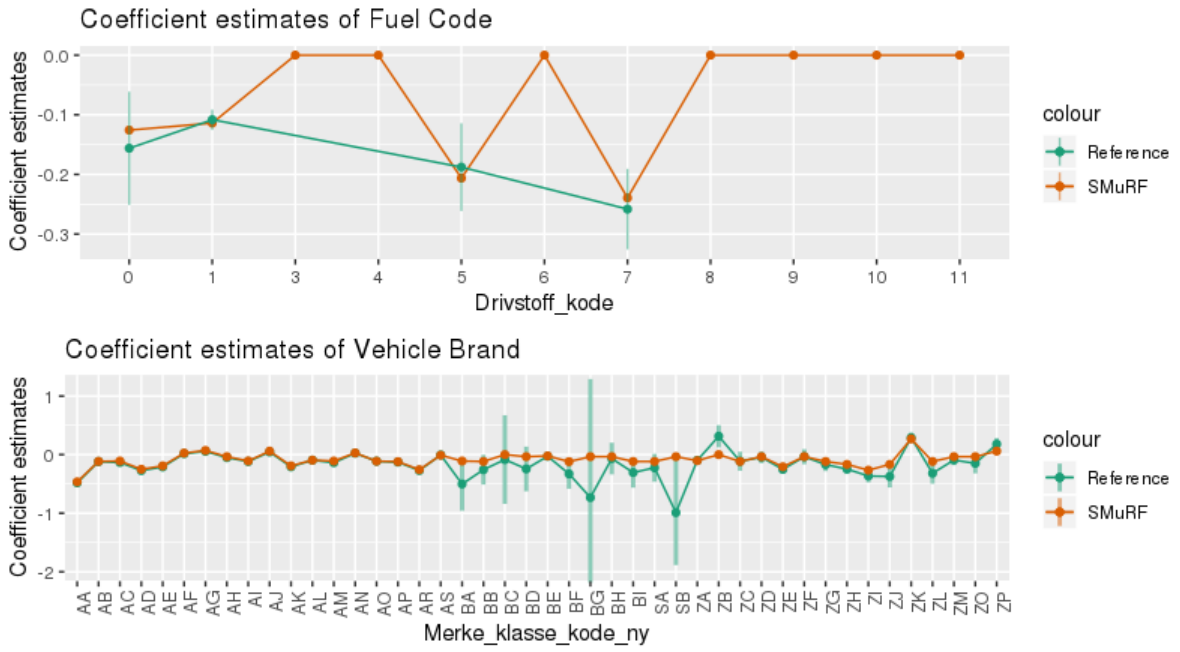


Figure 4.3 - Coefficient estimate for each level for the SMuRF model and the reference model for fuel code and vehicle brand

Figure 4.3 compares the coefficient estimates of the two models for two of the nominal predictors included in the model; fuel code and vehicle brand. As they are nominal, all levels can potentially be fused together. Since there is no logical sequence of the levels, they have been sorted in ascending and alphabetical order respectively.

The two plots show that many coefficient estimates have been set to zero for the regularized model. For fuel code, seven of eleven levels have been set to zero and are therefore fused

with the reference level, *2*. These seven levels are the same levels Gjensidige manually group before estimating the reference model, meaning the two models end up with the same number of unique coefficients and similar estimates. As we observed from the descriptive analysis, fuel code has several levels with close to zero observations, which could be why they are set to zero, and why Gjensidige has fused them with other categories.

The same 44 levels of vehicle brand were used as input for both models. However, the SMuRF algorithm set two of its coefficients to zero and fused several others together, ending up with 27 unique coefficients. The coefficient estimates in the reference model with the highest standard errors seem to have been shrunk the most, seen mostly in the middle of the plot from *BA* to *ZB*. Two of these, *BC* and *ZB*, have been set to zero and are therefore fused with the reference level.

Binary predictors

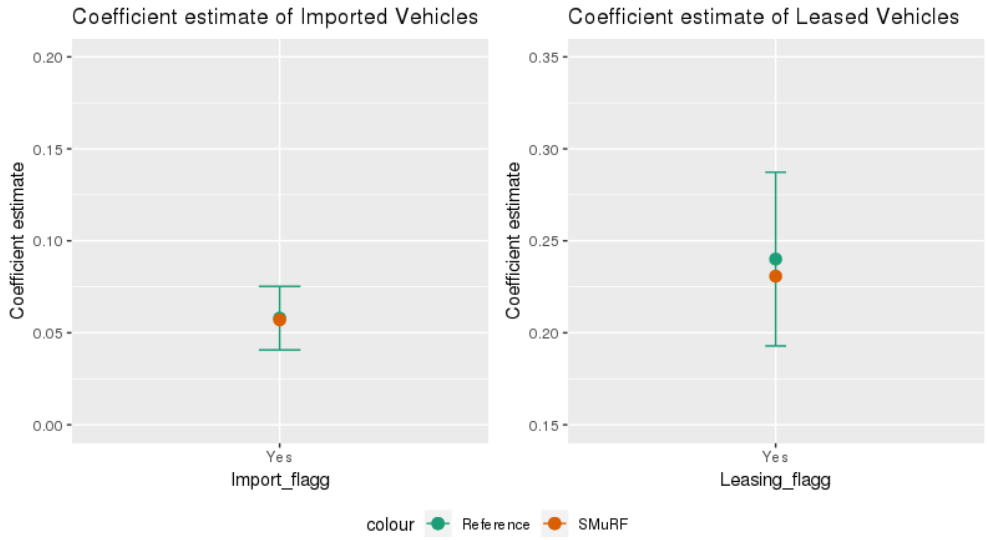


Figure 4.4 - Coefficient estimate for each level for the SMuRF model and the reference model for imported vehicles and leased vehicles

Figure 4.4 compares the coefficient estimates of the two models for the two binary predictors penalized in the model, imported vehicles and leased vehicles, which cannot be grouped as they only have one parameter. They could have been removed from the model if the one parameter was set to zero, but they are not removed in our model. We can see how the coefficient estimate has been constrained towards zero by the SMuRF for both predictors. The coefficient estimate for leased vehicles has been constrained more than the estimate of imported vehicles, which may be because its standard error is higher.

Predictors not suited for level fusion

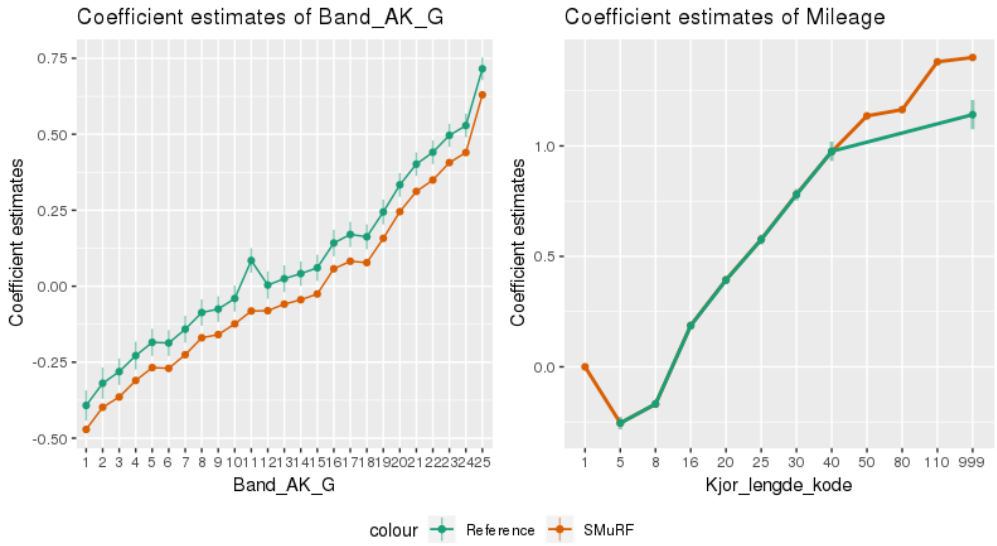


Figure 4.5 - Coefficient estimate for each level for the SMuRF model and the reference model for Band_AK_G and mileage

Figure 4.5 compares the coefficient estimates for the two models for a nominal and an ordinal predictor for which levels have not been grouped. These two predictors are *Band_AK_G*, a predictor constructed by Gjensidige based on geographic and demographic factors, and *Kjor_lengde_kode*, a constructed predictor for mileage.

Both predictors seem to be constructed to have an increasingly positive effect on claims, beginning with a negative effect and ending up with a positive effect. As shown in *Dataset,* the mean number of claims increases for each level for both predictors, and the variable has an approximately linear relationship with number of claims. Consequently, the two predictors are already constructed to have levels that share the same effect on number of claims and is a good example of how predictors could be grouped ideally. Therefore, it is not surprising the levels have not been grouped by the SMuRF algorithm.

By including a regularization term in the objective function, some predictor levels have been grouped while some have been removed entirely. Regularization improves prediction accuracy by reducing variance, but this comes at the cost of an increase in bias. In the following section, we introduce a new model to address this.

4.2.3 **Reestimated model**

To counter the increased bias introduced in the model by the regularization term, we reestimate the regularized model without penalties, where the coefficients estimated by the SMuRF algorithm are used as input to select and group levels. The predictors for which the coefficients were estimated to be zero are removed, while any fused coefficients are included in their collapsed form. The reestimation is done using the *smurf* package (Reynkens et al., 2018).

Reestimating the regularized model without the penalty term effectively means using the same objective function

$$\mathcal{O}(\boldsymbol{\beta}; \boldsymbol{X}, \boldsymbol{y}) = -\frac{1}{n}\sum_{i=1}^{n}(y_i(x_i\beta + \log(expo_i)) - e^{(x_i\beta+\log(expo_i))} - \log(y_i!)) \quad (4.3)$$

as the reference model, but with differently coded predictors

4.3 Validation and comparison of models

To find out whether selecting and fusing predictors using the SMuRF algorithm can lead to improved prediction accuracy compared to Gjensidige's current method, we validate and compare the reference model, the first regularized model and the reestimated model using selected performance criteria presented in *Methods*. To be able to analyze why the different models perform as they do, we also study the coefficient estimates of the levels of some of the predictors. In addition, to illustrate the effect of the regularization term, we illustrate how the MSE, bias, variance and the reduction in degrees of freedom changes as the strength of the penalty changes. The test dataset including observations from 2017 and the R package *stats* (R Core Team, 2018) was used for calculating test MSE.

4.3.1 **Comparison of models**

Criteria	Model: Reference	Estimated SMuRF	Reestimated SMuRF
Reduction in DF	298	365	282
AIC	1 240 098	1 239 809	**1 239 746**
Training MSE	0.0264971	0.0264963	**0.0264962**
Test MSE	0.0266903	**0.0266876**	0.0266879

Table 4.2 - Performance of the different models using different criteria

The reestimated SMuRF performs the best for AIC and training MSE, while the estimated SMuRF performs the best for test MSE. The two SMuRF models perform better than the reference model for every criterion.

Degrees of freedom (DF)

As the SMuRF method collapses categories, the number of degrees of freedom used is reduced from 365 to 282. Compared to the reference model, the reduction is from 298 to 282. This means a simpler model has been used to increase prediction accuracy. In addition, the reduction in used degrees of freedom means interpretability increases, as the number of coefficients is reduced.

AIC

The calculated AIC is lower for the reestimated SMuRF than for the reference model, which indicates a better model. This can either be due to a lower deviance, a reduced number of parameters or a combination of the two. As it takes the number of parameters in the models into account, it favors the SMuRF method relatively higher than the MSE results do.

MSE

The test MSE of the SMuRF method is slightly lower than the reference model, indicating increased prediction accuracy as a result of using the SMuRF algorithm to treat the variables. However, the results are very similar. The calculated MSE for the models consists of the estimation error attributed to both bias and variance. As the MSE for the SMuRF models are lower, a reduction in variance has more than offset a potential increase in bias.

4.3.2 Groups created with the different methods

The performance criteria used to compare the models conclude that the SMuRF models perform better than the reference model in terms of prediction accuracy. As the difference between the models is how the predictor's levels are grouped, we will now analyze the estimated coefficients of the levels of some of the predictors. Figure 4.6 shows the estimated coefficients for all three models for four of the predictors, horsepower, vehicle age, imported vehicles and *Subcluster*. Similar plots for the remaining predictors can be found in Appendix A5.

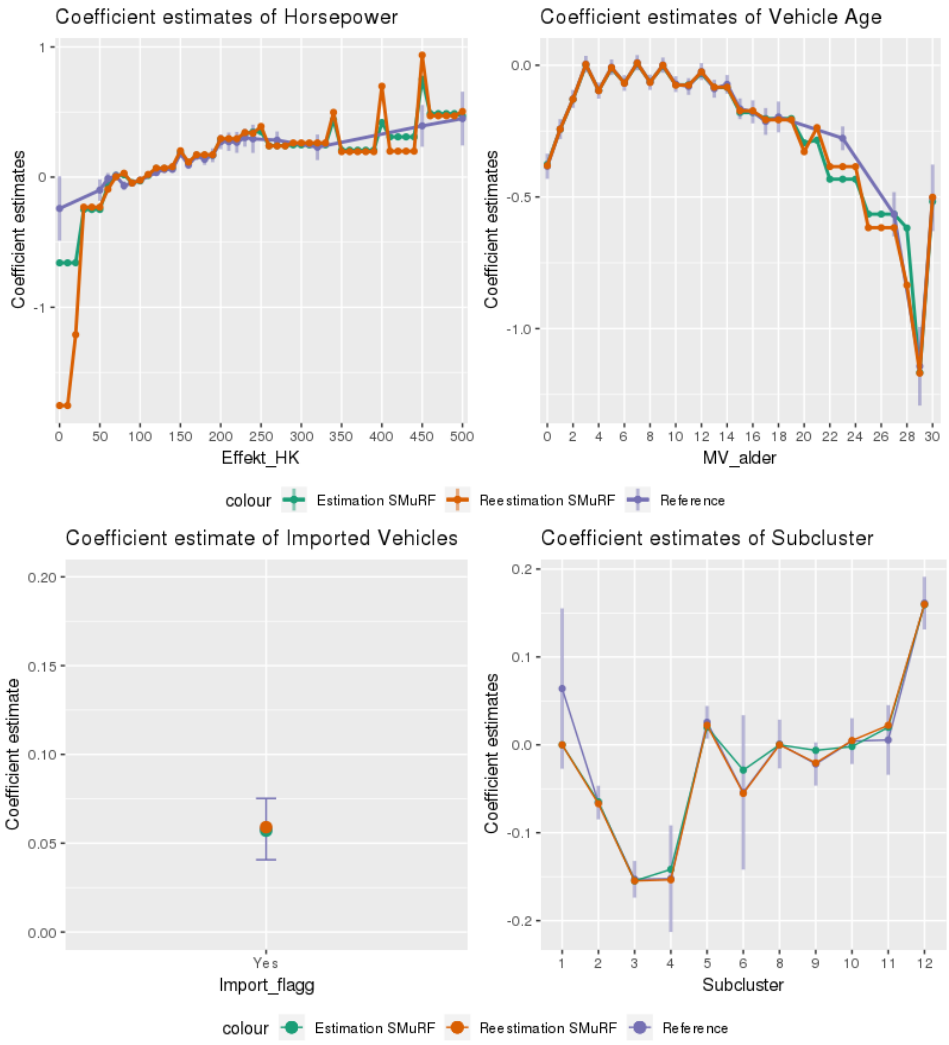


Figure 4.6 - Coefficient estimate for each level for the first SMuRF model, reestimated SMuRF model and the reference model for horsepower, vehicle age, imported vehicles and Subcluster

The trends of the three lines representing the coefficients estimated for each model mostly follow each other. The differences mainly appear when the standard errors of the reference model estimates are higher. Comparing the coefficient plots to the descriptive analysis, this mostly concerns levels containing few observations. In addition, the graphs show that the variation in the reestimated estimates are higher than in the first estimation of the SMuRF algorithm. For example, in the interval from 325 to 450 of *Horsepower*, there are three clear spikes in the coefficient estimates. The estimates of the reestimated coefficients are further away from zero than the first SMuRF estimation for almost every level where the estimate is not identical. For the last plot, imported vehicles, the reestimated coefficient is also further away from zero. This all indicates that the variance has increased as a consequence of reducing the bias when reestimating the coefficients.

In addition, the SMuRF algorithm has removed some levels with high variance from the model. For example, the level *1* of *Subcluster* seems to have a large standard error. As the SMuRF method has removed it from the model entirely, variance is likely reduced at the cost of some bias, since the model is simplified.

However, for another example like the level *6* of subcluster, the first SMuRF estimation has shrunk the coefficient some to reduce variance, while the reestimation has undone the shrinkage and ends up with the same estimate as the reference model. This increase in variance is likely the reason why the reestimation performs worse for test MSE than the first SMuRF estimation, as the reduction in bias has not been large enough to compensate for the increase in variance.

4.3.3 The effect of regularization

As we have observed that regularization can improve the test MSE compared to the reference model, it is interesting to analyze how different values of λ changes test MSE. Therefore, we create Figure 4.7 which illustrates how the test MSE and degrees of freedom differentiate for different values of λ. We have also included the reference model and the SMuRF model with $\lambda = 0$ for comparison of reduction of degrees of freedom and test MSE, even though the logarithm of zero is $-\infty$. We have therefore labelled their λ to 0.00000001 and 0.00000001 to be able to include them in the plot.

The plot shows that the number of degrees of freedom is lower for the reference model than the regularized models with the lowest λ's, because of the difference in data pre-processing.

Also, we observe that the MSE decreases by creating more bins in preparation for the SMuRF model. In addition, regularizing the model decreases the MSE steadily until a certain point, where the MSE begins to increase rapidly. If the value of λ is higher than at this point, the degrees of freedom will be reduced further but the test MSE will increase. This illustrates the trade-off between prediction accuracy and interpretability, as the lower values of λ give a more flexible model while increased λ reduces the variance at the cost of increased bias. The increase in test MSE especially occurs when reducing the number of parameters from 192 to 163, which indicates that some of the variables or levels removed in that interval are important for prediction accuracy.

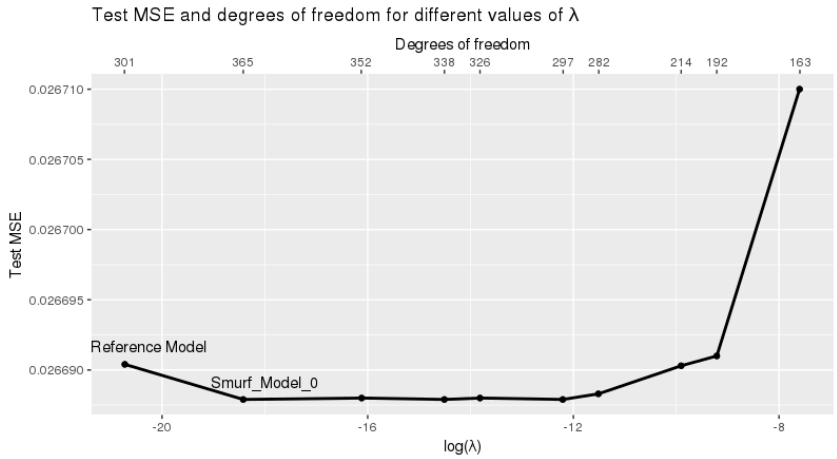


Figure 4.7 - MSE for models created with different values of λ

Figure 4.8 shows how the variance and squared bias change with an increasing λ. The results mostly follow what we expected, as the trend in the plot is that an increased λ leads to an increase in bias and a reduction in variance. Surprisingly, there is a spike in the variance for the model which uses 192 degrees of freedom. However, the variance can sometimes increase when the number of parameters in the model is reduced.

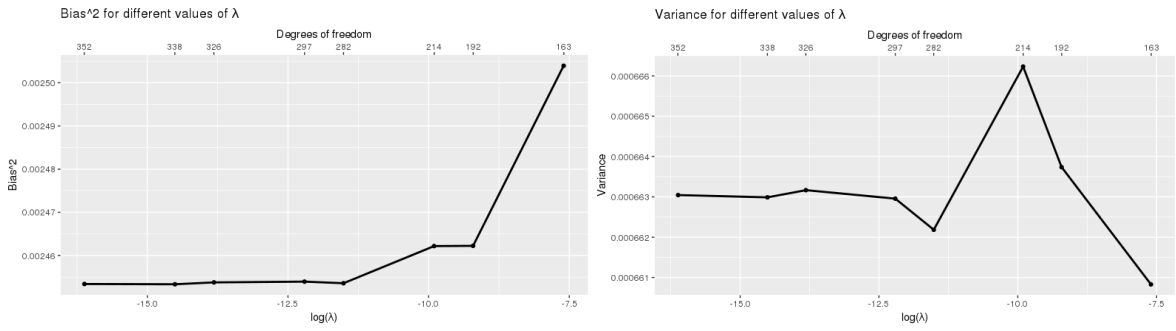


Figure 4.8 - Bias and variance for different values of λ

5. Discussion

The results of our analysis indicate that grouping and binning the variables used in Gjensidige's claim frequency model by applying the SMuRF method, rather than doing it manually, results in a slightly better model in terms of both test MSE and AIC. Surprisingly, the first estimation of the regularized model performs better than the reestimated model, even though the purpose of the reestimated model is to reduce bias introduced to the model during regularization. Whether the improvement from the reference model to our models is large enough for Gjensidige to consider implementing it, and whether the results are robust over different datasets remains to be concluded. However, our results show that the newly developed method is promising.

To gain a better understanding of our results, we discuss the mechanisms of the regularization method used and how the constant tradeoff between minimizing the distance between observed and predicted values of the dependent variable and penalizing coefficients has played a role. The difference in MSE, between the two models' predictions can be attributed to both the differences in fusion of levels and how the binning was done differently prior to model estimation.

5.1 Results

The choice of manually binning continuous variables prior to modelling introduce bias to both the reference model and the models estimated with the SMuRF algorithm, as we lose some information on specific values of a variable by combining them. However, in the preparation for the SMuRF algorithm, the bins are much narrower than for the reference model. By allowing more parameters in the model and increasing thereby flexibility, this bias is reduced compared to the reference model. On the other hand, this comes at the cost of increased variance. As our analysis show, some of the additional parameters we introduce with this crude binning are not needed. Some groups of levels in the reference model seem to have been formed by Gjensidige for this reason. They have discovered that the levels are not valuable for the model, for example due to few observations or that the effect of several levels on number of claims is very similar. Still, the cruder binning lead to a reduced MSE, meaning the model was already improved before applying the SMuRF algorithm. This was

somewhat surprising to us, as it indicates that simply creating more bins without considering their limits gave better results than grouping manually while considering the data.

However, the reduction in MSE caused by the reduction in bias from the cruder binning is offset by an increase in the use of degrees of freedom, as more bins means a more flexible model. The next step of applying the SMuRF algorithm to regularize the model counteracts this problem, as the main idea is to reduce excess parameters. When using a regularization method, the choice of λ is essential because it decides the tradeoff between a flexible model and a simple model by regulating the number and size of parameters in the model. As our analysis show, the estimated coefficients of the models are quite similar. This signals that while it is optimal to constrain some coefficients, the information is valuable enough for the method not to penalize a great amount.

In our study, increasing λ more than the optimal value leads to a large decrease in number of parameters, but an increase in MSE. Up to a certain point, regularizing the coefficients leads to a reduction in variance that dominates the increase in bias, because initially, levels that are not important for the model performance are removed. However, with an increasing value of λ, MSE can increase as levels with high importance to the performance of the model are removed or fused in a way so that information is lost. As shown in *Analysis*, the likelihood of fusion and removal of levels is much higher when the standard error of the coefficient estimate is higher, meaning the SMuRF algorithm recognizes that the estimate introduces too much error to be included.

Penalizing the coefficients reduces the flexibility of the model as the number of parameters is reduced, which increases the bias. The model is therefore reestimated using the selected and fused parameters in order to reduce the bias introduced by penalizing the coefficients. Consequently, the variance of the model increases, which could be observed in the coefficient plots in *Analysis.* The estimates of the reestimated model often reversed some of the penalization performed by the regularized model, which results in a more flexible model. As it is often recommended to perform this reestimation to reduce bias, it was surprising that the reestimated model ended up performing poorer, but it indicates that decrease in bias has not been large enough to overcome the increase in variance from the regularization of the coefficients.

5.2 Implications

One of the aims of this study has been to find and employ a method that can improve Gjensidige's prediction models in some way, if implemented. It was therefore important for us to find a method suitable for Poisson regression. The output of a model estimated using the *smurf* package (Reynkens et al., 2018) is almost identical to the output from the *stats* package (R Core Team, 2018), and this familiarity will likely simplify the implementation process, which is an important aspect because radical changes take time in large organizations.

For Gjensidige to consider implementing the SMuRF method, there has to be a potential economic gain related to the method that compensates them for the effort of implementing a new system. In this thesis, prediction accuracy is used to evaluate whether there is an economic gain, as the performance of the predictive models is for Gjensidige's overall profit. However, it is difficult to draw conclusions on the value that the small change in MSE constitutes from the MSE numbers reported in Analysis. For 2017, the improvement in the number of claims predicted correctly would be 8.4, out of a total of 29 217 claims. Therefore, the improvement in prediction accuracy is not very large, but 8.4 claims can still account for a large sum of money when considering claims connected to vehicles.

If Gjensidige decides to implement the method, they can also consider to what extent they want to implement it. As the *smurf* package estimates and reestimates the model using *speedglm*, it is possible for Gjensidige to use the *smurf* package for model estimation in the same dynamic way they estimate the Poisson regression models today, using the *stats* package (R Core Team, 2018). In Gjensidige's prediction framework the models are updated continuously, and by fully implementing the smurf package, they can ensure that the parameters they use are up-to-date when new levels, variables or observations are introduced to the dataset. Another option is to implement the new method in a more static way by continuing to estimate the prediction models using the *stats* package (R Core Team, 2018) and to group the levels of the categorical variables manually. The method can still be somewhat implemented by running the SMuRF algorithm using the *smurf* package (Reynkens et al., 2018) whenever the analysts consider changing the level groups. Even though the process of grouping the levels will be done manually, the groups will still be chosen in a data-driven way, and the analysts can spend less time researching the different categories. Another benefit of this approach is that it does not necessarily require getting the

entire organization on board before using it. This approach is the most relevant one if Gjensidige considers the increase in prediction accuracy too small to spend time and resources on implementing it, but still consider it beneficial to save time on grouping the predictor levels manually. If we trust that our results from *Analysis* are robust, the more dynamic alternative will likely provide the best prediction accuracy and be most time-effective. However, Gjensidige must still consider whether the time required to develop and implement the new method is worth the small increase in prediction accuracy it can provide.

After discussing the results with our persons of contact in Gjensidige, the main potential of the model seems to be related to the possibility of reducing the time analysts spend grouping predictor levels. The groups used today are results of many years of research and decisions by the analysts. In some instances, the SMuRF algorithm has chosen almost exactly the same groups as the analysts, as shown in the coefficient plots in *Analysis*. This is perhaps the most interesting finding from Gjensidige's perspective, as it illustrates the potential to save time and resources on finding suitable groups manually. Considering the need to update the models frequently as new observations, variables or levels are introduced, a method which can do group the variable levels automatically can prove to be very valuable.

5.3 Limitations and weaknesses

As in any other study, there are several limitations and weaknesses to our research. First of all, our results are dependent on the performance measures we have chosen to use, as other performance measures could possibly consider the reference model superior as the margins are very slim. The results are also dependent on our choice of test set. Our test set only consists of observations from 2017, which may have been a year different from the ordinary. In that case, our model may have a superior ability to predict the number of claims for 2017 compared to the reference model, but potentially not for other future years. However, our model also performed better on the training set which contains observations from 2012-2016.

Exploring different ways of improving the prediction accuracy of Gjensidige's models has not been the focus of our analysis. For example, it might be of interest to attempt to estimate these models without binning continuous variables beforehand, but this has not been investigated. Our focus has rather been on fusing predictor levels differently to improve

prediction accuracy, without considering other ways it could be improved. Therefore, it might be the case that none of these models perform particularly well.

In addition, when searching for the optimal λ using cross-validation, we were only able to use a subset of the observations as the cross-validation function of the *smurf* package (Reynkens et al., 2018) did not work on Gjensidige's server. We further used the λ chosen based on the subset as input for the SMuRF algorithm, which was run on the entire dataset. This means the optimal λ for the whole dataset may not have been chosen, but through trial-and-error we identified that the MSE of the λ we used was one of the lowest of all λ, as we also show in our analysis.

6. Conclusion

In this thesis we have researched whether the prediction accuracy of Gjensidige's models for predicting claim frequency can be increased by using a data-driven method for the fusion of categorical predictor levels. We therefore searched for a method that could handle all predictor types and was compatible with large datasets. The method we selected was the SMuRF method, which relies on regularization to select variables and fuse levels of categorical predictors. To assess its performance, we estimate a reference model identical to the model currently employed by Gjensidige to predict claim frequency.

The results of our analysis show that the model estimated using the SMuRF method performs slightly better in terms of MSE and AIC than the reference model. It also reduces the number of degrees of freedom used to create the model, as only levels important to the model's performance are included. The reestimated model we create using selected and fused levels performs better than the reference model but has a lower prediction accuracy than the first regularized model.

However, the most important finding for Gjensidige was perhaps not the slightly improved performance of the model. It is reassuring that the models perform at a similar level, but the fact that the SMuRF algorithm is able to automatically group levels as well as the analysts is of greater interest. The new method can therefore have a positive effect on the tasks and resource usage of the department developing the prediction models.

References

Akaike, H. (1974). A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control, 19* (6), 716-723. Retrieved from https://ieeexplore.ieee.org/document/1100705

Beck, A. & Teboulle, M. (2009). A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences, 2* (1), 183–202. Retrieved from https://epubs.siam.org/doi/pdf/10.1137/080716542

Berger, M. & Tutz, G. (2014). Tree-Structured Modeling of Categorical Predictors in Regression. (Technical Report 169). Retrieved from https://epub.ub.uni-muenchen.de/21358/1/tr169.pdf

Bondell, H. & Reich, B. (2009). Simultaneous Factor Selection and Collapsing Levels in ANOVA. *Biometrics, 65* (1), 169-177. Retrieved from https://onlinelibrary.wiley.com/doi/full/10.1111/j.1541-0420.2008.01061.x

David, M. & Jemna, D. (2015). Modeling the Frequency of Auto Insurance Claims by Means of Poisson and Negative Binomial Models. *Scientific Annals of the "Alexandru Ioan Cuza" University of Iaşi Economic Sciences, 62* (2), 151-168. Retrieved from https://www.degruyter.com/downloadpdf/j/aicue.2015.62.issue-2/aicue-2015-0011/aicue-2015-0011.pdf

Devriendt, S., Antonio, K., Reynkens, T. & Verbelen, R. (2018). Sparse Regression with Multi-type Regularized Feature Modeling. Retrieved from https://arxiv.org/pdf/1810.03136.pdf

Donges, N. (2018, March 7). Gradient Descent in a Nutshell. Retrieved from https://towardsdatascience.com/gradient-descent-in-a-nutshell-eaf8c18212f0

Dunteman, G. & Ho, M. (2006). *An Introduction to Generalized Linear Models*. Montreal: Sage Publications Inc.

Efron, B., Hastie, T., Johnstone, I. & Tibshirani, R. (2004). Least Angle Regression. *The Annals of Statistics, 32* (2), 407-499. Retrieved from https://projecteuclid.org/download/pdfview_1/euclid.aos/1083178935

European Central Bank (ECB). (2009). The importance of insurance companies for financial stability. *Financial Stability Review, 6* (2), 160-168. Retrieved from https://www.ecb.europa.eu/pub/pdf/fsr/art/ecb.fsrart200912en_05.pdf?016079a6d617b8c3c04b41995a92fe61

Finkelstein, A. & Poterba, J. (2004). Adverse Selection in Insurance Markets: Policyholder Evidence from the U.K. Annuity Market. *Journal of Political Economy, 112* (1), 183-208. Retrieved from https://www.jstor.org/stable/10.1086/379936

Gertheiss, J. & Tutz, G. (2010). Sparse Modeling of Categorial Explanatory Variables. *The Annals of Applied Statistics, 4* (4), 2150-2180. Retrieved from

https://projecteuclid.org/download/pdfview_1/euclid.aoas/1294167814

Gjensidige. (2018a). Gjensidige Group. Retrieved from https://www.gjensidige.no/group/about-us

Gjensidige. (2018b). *Annual Report 2017*. Retrieved from https://www.gjensidige.no/group/annual-report

Gjensidige. (2018c). Insurance. Retrieved from https://www.gjensidige.no/group/english

Goldburd, M., Khare, A & Tevet, D. (2016). *Generalized linear models for insurance rating* (CAS Monograph Series 5). Retrieved from https://www.casact.org/pubs/monographs/papers/05-Goldburd-Khare-Tevet.pdf

Hastie, T., James, G., Tibshirani, R. & Witten, D. (2017). *An Introduction to Statistical Learning with Applications in R*. Retrieved from https://wwwbcf.usc.edu/~gareth/ISL/ISLR%20Seventh%20Printing.pdf

Hastie, T., Tibshirani, R. & Friedman, J. (2009). *The Elements of Statistical Learning.* Retrieved from https://web.stanford.edu/~hastie/ElemStatLearn/printings/ESLII_print12.pdf

Hastie, T., Tibshirani, R. & Wainwright, M. (2015). *Statistical Learning with Sparsity.* Retrieved from https://web.stanford.edu/~hastie/StatLearnSparsity_files/SLS_corrected_1.4.16.pdf

Hoerl, A. & Kennard, R. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics, 12* (1), 55-67. Retrieved from https://math.arizona.edu/~hzhang/math574m/Read/RidgeRegressionBiasedEstimatiForNonorthogonalProblems.pdf

Hothorn, T., Bretz, F. & Westfall, P. (2008). Simultaneous Inference in General Parametric Models. *Biometrical Journal, 50* (3), 346-363. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1002/bimj.200810425

Iowa Insurance Institute. (2017). How the insurance industry helps stimulate the economy. Retrieved from https://iowains.org/how-the-insurance-industry-helps-stimulate-the-economy/

Johnson, K. & Kuhn, M. (2013). *Applied Predictive Modeling*. New York: Springer-Verlag

Korsvoll, R. (2016, December 30). Antall hestekrefter i bilene oker kraftig. *Motor.* Retrieved from https://www.motor.no/artikler/2017/januar/antall-hestekrefter-i-bilene-oker-kraftig/?fbclid=IwAR0eLM35YmosZI_wWeH4sUb7hIWeuQRiOMkv8B5rZfbft6Js2tbl9bJjDSY

Oelker, M. & Tutz, G. (2017). A Uniform Framework for The Combination of Penalties in Generalized Structured Models. *Advances in Data Analysis and Classification, 1* (1), 97-120. Retrieved from https://link.springer.com/article/10.1007/s11634-015-0205-y

Parodi, P. (2016). From the expected losses to the technical premium. Retrieved from

https://www.cii.co.uk/knowledge/resources/articles/basic-concepts-and-techniquesof-the-pricing-process/43681#fro

Pauger, D. & Wagner, H. (2017). Bayesian Effect Fusion for Categorical Predictors. Retrieved from https://projecteuclid.org/download/pdfview_1/euclid.ba/1527213627

R Core Team (2018). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.

Reynkens, T., Devriendt, S. & Antonio, K. (2018). smurf: Sparse Multi-Type Regularized Feature Modeling. R package version 1.0.0. https://CRAN.R-project.org/package=smurf

Rinaldo, A. (2009). Properties and Refinements of the Fused Lasso. *Annals of Statistics, 37* (5B), 2922-2952. Retrieved from https://arxiv.org/pdf/0805.0234.pdf

Statistisk Sentralbyrå (SSB). (2018). Non-life insurance companies. Breakdown of premiums and claims (2014-2017). Retrieved from https://www.ssb.no/en/bank-og-finansmarked/statistikker/forsikring/aar

Sydsæter, K., Seierstad, A. & Strøm, A. (2002). *Matematisk Analyse Bind 2* (4th edition). Oslo: Gyldendal Norsk Forlag

Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society, 58* (1), 267-288. Retrieved from http://www.math.yorku.ca/~hkj/Teaching/6621Winter2013/Coverage/lasso.pdf

Tibshirani, R. (1997). The Lasso Method for Variable Selection in The Cox Model. *Statistics in Medicine, 16,* 385-395. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.123.2680&rep=rep1&type=pdf

Tibshirani, R. (2003). A Simple Explanation of the Lasso and Least Angle Regression. Retrieved from https://statweb.stanford.edu/~tibs/lasso/simple.html

Tibshirani, R., Saunders, M., Rosset, S., Zhu, K. & Knight, K. (2005). Sparsity and Smoothness via the Fused Lasso. *Journal of the Royal Statistical Society Series B, 67* (1), 91–108. Retrieved from https://web.stanford.edu/group/SOL/papers/fused-lasso JRSSB.pdf

Tukey, J. (1949). Comparing Individual Means in the Analysis of Variance. *Biometrics, 5* (2), 99-114. Retrieved from http://webspace.ship.edu/pgmarr/Geo441/Readings/Tukey%201949%20%20Comparing%20Individual%20Means%20in%20the%20Analysis%20of%20Variance.pdf

Viallon, V., Lambert-Lacriox, S., Höfling, H., & Picard, F. (2014). On the Robustness of the Generalized Fused Lasso to Prior Specifications. *Statistics and Computing, 26* (1), 285– 301. Retrieved from http://pbil.univ-lyon1.fr/members/fpicard/franckpicard_fichiers/pdf/viallon.2014.pdf

Wickham, H. (2016). ggplot2: Elegant Graphics for Data Analysis. New York: Springer Verlag. https://CRAN.R-project.org/web/packages/ggplot2/index.html

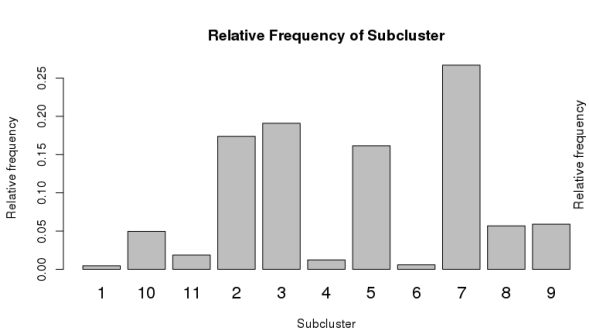
Wickham, H., François, R., Henry, L. & Müller, K. (2018). dplyr: A Grammar of Data Manipulation. R package version 0.7.6. https://CRAN.R-project.org/package=dplyr

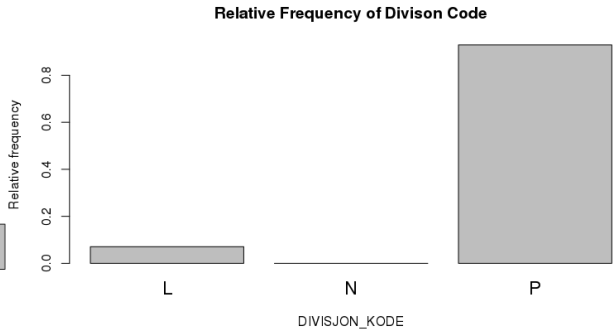
Xin, B., Kawahara, Y., Wang, Y., & Gao, W. (2014). Efficient Generalized Fused Lasso and its application to the diagnosis of Alzheimer's disease. Proceedings of the TwentyEighth AAAI Conference on Artificial Intelligence, 2163–2169. Retrieved from https://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8261/8862

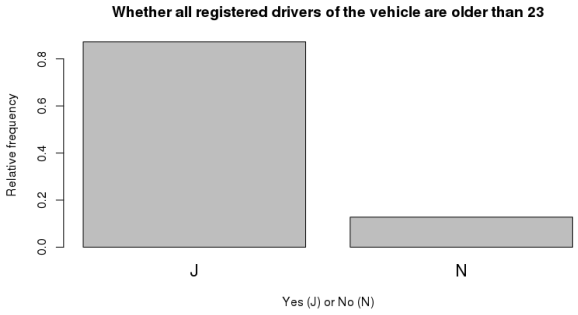
Appendix

A1

Table and plots of relative frequencies of the remaining variables, excluding the numerical ones.







	Import_flagg			***Leasing_flagg***		
Levels	Frequency	Relative frequency	**%**	Frequency	Relative frequency	**%**
Yes	1 071 152	0.12682	**12.682**	102 649	0.0121528	**1.2153**
No	7 375 203	0.87316	**87.316**	8 343 852	0.9878418	**98.784**
Unknown	192	0.00002	**0.002**	46	0.000005	**0.0005**

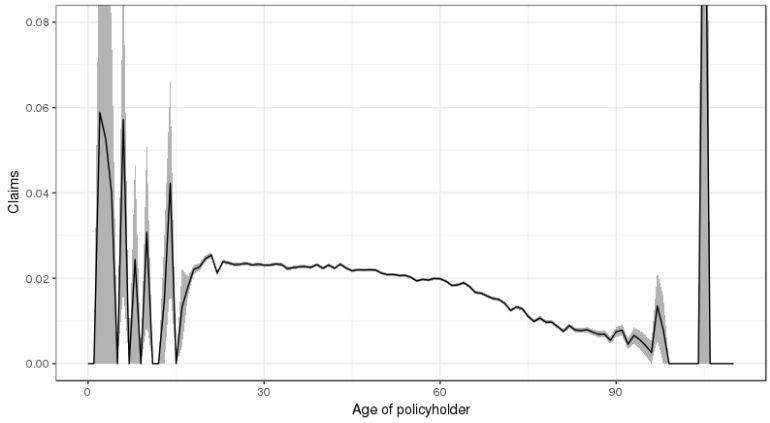
A2

This section shows the mean of claims for variable levels for the remaining variables.

Alder_ftaker: Age of the policyholder

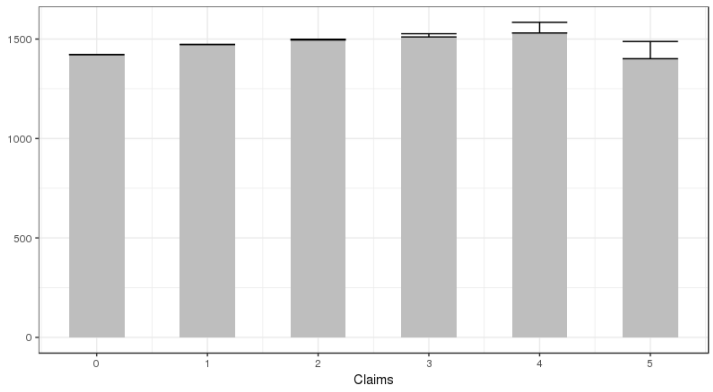
The figure below shows the mean and standard error of Claims for different ages of policyholders. The line represents the mean values, while the grey ribbon surrounding it represents the standard errors of the mean of Claims for each age. The plot includes data on policyholders under the age of 18, which does not make much sense as it is not possible to have a driver's license in Norway before the age of 18. As there are very few observations for these ages, the standard errors are very high.

The standard errors are also very high for ages above 80, as there are few people who drive vehicles at this age. We observe that the mean number of claims decreases with age. The number of claims peaks around 20 years of age, and then quite steadily declines up to the age of approximately 65, after which it declines with a faster rate. It not surprising that young drivers cause many claims.



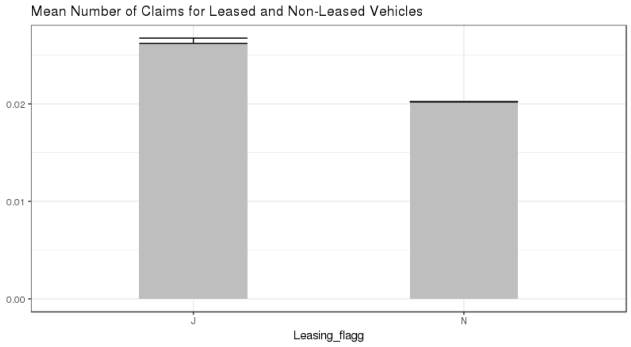
Egenvekt: Weight

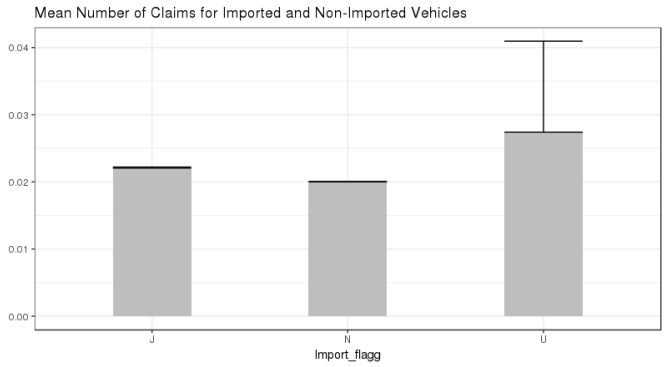
The figure below shows the mean weight in kilograms for vehicles of policyholders from zero to five claims. It seems that the weight of the car increases slightly when the number of claims increase, before decreasing for observations with five claims.



Leasing_flagg and Import_flagg: Leased cars and imported cars

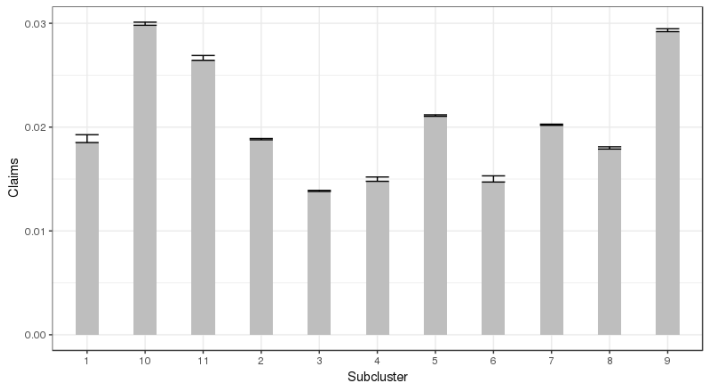
The figures below shows the mean number of claims for policyholders depending on whether their vehicle is imported or not, in addition to the mean number of claims for policyholders depending on whether their vehicle is leased or not. It seems that the average number of claims are somewhat higher for cars that are imported or leased compared with vehicles that are not imported or not leased. The last group for both variables refers to the cases where it is "unknown" whether the vehicle of the policyholder is imported or leased. It seems like the "unknown" vehicles have a mean number of claims between the two groups, which is to be expected, but we also observe that the standard errors are large because very few observations are "unknown".





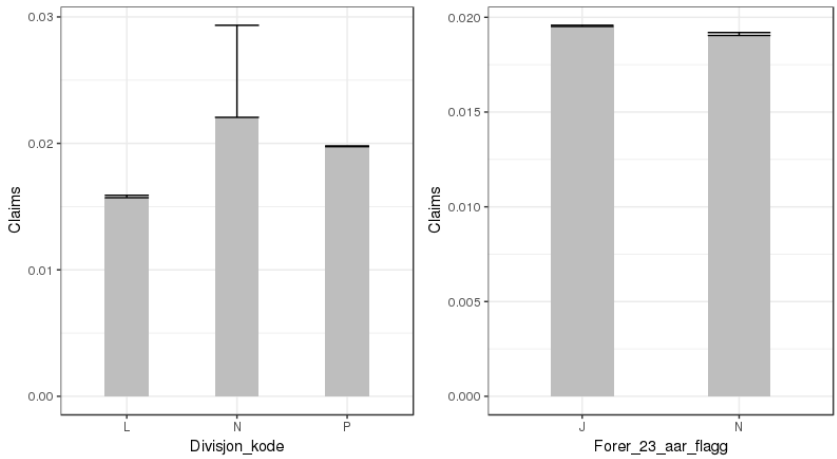
Subcluster: Combination of geography and demography

The figure below shows the mean number of claims for policyholders for *Subcluster*. There is a quite high difference between the different levels, and the standard errors are also relatively small.



Divisjon_kode and Forer_23_aar_flagg: Policy registration and whether vehicle is driven by someone below 23 years old

The figure below shows the mean number of claims for policyholders for *Divisjon_kode* and *Forer_23_aar_flagg*. There is a difference between the mean number of claims for *Divisjon_kode*, but *N* has a large standard error. Still, it seems it can have an effect on the number of claims. For *Forer_23_aar_flagg*, the difference is very small between *J* and *N*.



A3

Coefficients of the estimated and reestimated SMuRF models, where '*' indicates a zero coefficient and removal from the model.

	Estimated	Reestimated
Intercept	-2.548046	-2.467170
BAND_AK_G_GRUPPE_AK_G_Annen"1	-0.471258	-0.475545
BAND_AK_G_GRUPPE_AK_G_Annen"10	-0.123976	-0.124514
BAND_AK_G_GRUPPE_AK_G_Annen"11	-0.081610	-0.084170
BAND_AK_G_GRUPPE_AK_G_Annen"12	-0.080577	-0.080320
BAND_AK_G_GRUPPE_AK_G_Annen"13	-0.058987	-0.058977
BAND_AK_G_GRUPPE_AK_G_Annen"14	-0.044527	-0.042550
BAND_AK_G_GRUPPE_AK_G_Annen"15	-0.025521	-0.023399
BAND_AK_G_GRUPPE_AK_G_Annen"16	0.057408	0.058419
BAND_AK_G_GRUPPE_AK_G_Annen"17	0.082107	0.086446
BAND_AK_G_GRUPPE_AK_G_Annen"18	0.077550	0.078025
BAND_AK_G_GRUPPE_AK_G_Annen"19	0.157571	0.158924
BAND_AK_G_GRUPPE_AK_G_Annen"2	-0.398347	-0.404033
BAND_AK_G_GRUPPE_AK_G_Annen"20	0.245327	0.246867
BAND_AK_G_GRUPPE_AK_G_Annen"21	0.312045	0.315322
BAND_AK_G_GRUPPE_AK_G_Annen"22	0.349446	0.353853
BAND_AK_G_GRUPPE_AK_G_Annen"23	0.406719	0.409964
BAND_AK_G_GRUPPE_AK_G_Annen"24	0.439909	0.442866
BAND_AK_G_GRUPPE_AK_G_Annen"25	0.629796	0.630005
BAND_AK_G_GRUPPE_AK_G_Annen"3	-0.364579	-0.364941
BAND_AK_G_GRUPPE_AK_G_Annen"4	-0.310364	-0.312917
BAND_AK_G_GRUPPE_AK_G_Annen"5	-0.267842	-0.267505
BAND_AK_G_GRUPPE_AK_G_Annen"6	-0.270246	-0.271586
BAND_AK_G_GRUPPE_AK_G_Annen"7	-0.225430	-0.225926
BAND_AK_G_GRUPPE_AK_G_Annen"8	-0.169719	-0.171225
BAND_AK_G_GRUPPE_AK_G_Annen"9	-0.159237	-0.159134
DIVISJON_KODE_GRUPPEL	-0.169660	-0.172296
DIVISJON_KODE_GRUPPEZ	0.096027	0.097740
DRIVSTOFF_KODE_GRUPPE0	-0.125689	-0.146155
DRIVSTOFF_KODE_GRUPPE1	-0.113906	-0.111516
DRIVSTOFF_KODE_GRUPPE3	*	*

```
DRIVSTOFF_KODE_GRUPPE4                          *         *
DRIVSTOFF_KODE_GRUPPE5                         -0.206291 -0.211370
DRIVSTOFF_KODE_GRUPPE6                          *         *
DRIVSTOFF_KODE_GRUPPE7                         -0.239376 -0.259897
DRIVSTOFF_KODE_GRUPPE8                          *         *
DRIVSTOFF_KODE_GRUPPE9                          *         *
DRIVSTOFF_KODE_GRUPPE10                         *         *
DRIVSTOFF_KODE_GRUPPE11                         *         *
IMPORT_FLAGGN                                  -0.057047 -0.059030
IMPORT_FLAGGJ                                   *         *
LEASING_FLAGGN                                 -0.230779 -0.239951
LEASING_FLAGGJ                                  *         *
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AA -0.465145 -0.484312
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AB -0.115576 -0.126060
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AC -0.110021 -0.135362
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AD -0.251177 -0.266476
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AE -0.192167 -0.204560
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AF  0.026190  0.013063
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AG  0.070364  0.058632
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AH -0.034939 -0.055894
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AI -0.108343 -0.115675
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AJ  0.059323  0.046946
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AK -0.191936 -0.206165
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AL -0.094761 -0.097181
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AM -0.110021 -0.135362
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AN  0.029198  0.021847
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AO -0.112461 -0.114242
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AP -0.119724 -0.131233
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AR -0.257010 -0.267892
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"AS -0.010197 -0.007216
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"BA -0.110021 -0.135362
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"BB -0.117391 -0.207873
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"BC  *         *
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"BD -0.034939 -0.055894
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"BE -0.022333 -0.051574
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"BF -0.119768 -0.421467
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"BG -0.034939 -0.055894
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"BH -0.034939 -0.055894
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"BI -0.117391 -0.207873
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"SA -0.117391 -0.207873
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"SB -0.034939 -0.055894
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZA -0.103127 -0.096900
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZB  *         *
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZC -0.117391 -0.207873
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZD -0.034939 -0.055894
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZE -0.207443 -0.249617
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZF -0.033723 -0.031709
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZG -0.117391 -0.207873
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZH -0.167135 -0.274037
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZI -0.265857 -0.363983
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZJ -0.167135 -0.274037
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZK  0.267929  0.282186
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZL -0.117391 -0.207873
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZM -0.034939 -0.055894
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZO -0.034939 -0.055894
MERKE_KLASSE_KODE_NY_KLASSE_KODE_Annen"ZP  0.059323  0.046946
```

SUBCLUSTER_GRUPPE1	*	*
SUBCLUSTER_GRUPPE10	-0.001890	0.004906
SUBCLUSTER_GRUPPE11	0.019969	0.022225
SUBCLUSTER_GRUPPE12	0.159429	0.159846
SUBCLUSTER_GRUPPE2	-0.064786	-0.066908
SUBCLUSTER_GRUPPE3	-0.154768	-0.154773
SUBCLUSTER_GRUPPE4	-0.141647	-0.153514
SUBCLUSTER_GRUPPE5	0.019969	0.022225
SUBCLUSTER_GRUPPE6	-0.028744	-0.055242
SUBCLUSTER_GRUPPE8	*	*
SUBCLUSTER_GRUPPE9	-0.006282	-0.020608
EFFEKT_HK_GRUPPE0	-0.658565	-1.754997
EFFEKT_HK_GRUPPE1	-0.658565	-1.754997
EFFEKT_HK_GRUPPE2	-0.657857	-1.210773
EFFEKT_HK_GRUPPE3	-0.248731	-0.230512
EFFEKT_HK_GRUPPE4	-0.248731	-0.230512
EFFEKT_HK_GRUPPE5	-0.248731	-0.230512
EFFEKT_HK_GRUPPE6	-0.064490	-0.093134
EFFEKT_HK_GRUPPE7	0.012725	0.002717
EFFEKT_HK_GRUPPE8	0.017814	0.029897
EFFEKT_HK_GRUPPE9	-0.043023	-0.045318
EFFEKT_HK_GRUPPE10	-0.029219	-0.023949
EFFEKT_HK_GRUPPE12	0.013187	0.020775
EFFEKT_HK_GRUPPE13	0.065891	0.069394
EFFEKT_HK_GRUPPE14	0.065891	0.069394
EFFEKT_HK_GRUPPE15	0.068651	0.080292
EFFEKT_HK_GRUPPE16	0.189300	0.202445
EFFEKT_HK_GRUPPE17	0.111134	0.115120
EFFEKT_HK_GRUPPE18	0.167182	0.172065
EFFEKT_HK_GRUPPE19	0.167182	0.172065
EFFEKT_HK_GRUPPE20	0.167182	0.172065
EFFEKT_HK_GRUPPE21	0.289124	0.295559
EFFEKT_HK_GRUPPE22	0.289124	0.295559
EFFEKT_HK_GRUPPE23	0.289124	0.295559
EFFEKT_HK_GRUPPE24	0.332981	0.342290
EFFEKT_HK_GRUPPE25	0.346922	0.336804
EFFEKT_HK_GRUPPE26	0.349266	0.389487
EFFEKT_HK_GRUPPE27	0.241474	0.239626
EFFEKT_HK_GRUPPE28	0.241474	0.239626
EFFEKT_HK_GRUPPE29	0.241474	0.239626
EFFEKT_HK_GRUPPE30	0.247771	0.262910
EFFEKT_HK_GRUPPE31	0.247771	0.262910
EFFEKT_HK_GRUPPE32	0.247771	0.262910
EFFEKT_HK_GRUPPE33	0.247771	0.262910
EFFEKT_HK_GRUPPE34	0.247771	0.262910
EFFEKT_HK_GRUPPE35	0.440519	0.497546
EFFEKT_HK_GRUPPE36	0.206845	0.195553
EFFEKT_HK_GRUPPE37	0.206845	0.195553
EFFEKT_HK_GRUPPE38	0.206845	0.195553
EFFEKT_HK_GRUPPE39	0.206845	0.195553
EFFEKT_HK_GRUPPE40	0.206845	0.195553
EFFEKT_HK_GRUPPE41	0.419297	0.698508
EFFEKT_HK_GRUPPE42	0.309308	0.199257
EFFEKT_HK_GRUPPE43	0.309308	0.199257
EFFEKT_HK_GRUPPE44	0.309308	0.199257
EFFEKT_HK_GRUPPE45	0.309308	0.199257

EFFEKT_HK_GRUPPE46	0.752780	0.937438
EFFEKT_HK_GRUPPE47	0.487874	0.471029
EFFEKT_HK_GRUPPE48	0.487874	0.471029
EFFEKT_HK_GRUPPE49	0.487874	0.471029
EFFEKT_HK_GRUPPE50	0.487874	0.471029
EFFEKT_HK_GRUPPE51	0.470396	0.504333
EGENVEKT_GRUPPE0	-0.198391	-0.139170
EGENVEKT_GRUPPE0.5	-0.436435	-0.470030
EGENVEKT_GRUPPE1	-0.436435	-0.470030
EGENVEKT_GRUPPE2	-0.436435	-0.470030
EGENVEKT_GRUPPE3	-0.260489	-0.240131
EGENVEKT_GRUPPE4	-0.260489	-0.240131
EGENVEKT_GRUPPE5	-0.260489	-0.240131
EGENVEKT_GRUPPE6	-0.287824	-0.295166
EGENVEKT_GRUPPE7	-0.166171	-0.163682
EGENVEKT_GRUPPE8	-0.166171	-0.163682
EGENVEKT_GRUPPE9	-0.107896	-0.099092
EGENVEKT_GRUPPE10	-0.137988	-0.148789
EGENVEKT_GRUPPE11	-0.100351	-0.101600
EGENVEKT_GRUPPE12	-0.100351	-0.101600
EGENVEKT_GRUPPE13	-0.066981	-0.063508
EGENVEKT_GRUPPE14	-0.066981	-0.063508
EGENVEKT_GRUPPE15	*	*
EGENVEKT_GRUPPE16	*	*
EGENVEKT_GRUPPE18	0.048111	0.049953
EGENVEKT_GRUPPE19	0.048111	0.049953
EGENVEKT_GRUPPE20	0.048111	0.049953
EGENVEKT_GRUPPE21	-0.013300	-0.015942
EGENVEKT_GRUPPE22	0.067622	0.077657
EGENVEKT_GRUPPE23	0.067622	0.077657
EGENVEKT_GRUPPE24	0.045788	0.046997
EGENVEKT_GRUPPE25	0.034618	0.017683
EGENVEKT_GRUPPE26	0.037172	0.053741
EGENVEKT_GRUPPE27	0.156071	0.173508
EGENVEKT_GRUPPE28	0.097226	0.084427
EGENVEKT_GRUPPE29	0.102430	0.111835
EGENVEKT_GRUPPE30	0.102430	0.111835
EGENVEKT_GRUPPE31	0.102430	0.111835
EGENVEKT_GRUPPE32	0.270583	0.294438
EGENVEKT_GRUPPE33	0.023052	0.038198
EGENVEKT_GRUPPE34	-0.012323	-0.102010
EGENVEKT_GRUPPE35	0.132421	0.154441
EGENVEKT_GRUPPE36	0.202970	0.288565
EGENVEKT_GRUPPE37	0.194000	0.219570
EGENVEKT_GRUPPE38	0.194000	0.219570
EGENVEKT_GRUPPE39	0.194000	0.219570
EGENVEKT_GRUPPE40	0.078895	0.087301
EGENVEKT_GRUPPE41	0.078895	0.087301
EGENVEKT_GRUPPE42	0.078895	0.087301
EGENVEKT_GRUPPE43	0.078895	0.087301
EGENVEKT_GRUPPE44	0.078895	0.087301
EGENVEKT_GRUPPE45	0.078895	0.087301
EGENVEKT_GRUPPE46	0.280227	0.455678
KJOR_LENGDE_KODE_GRUPPE1	*	*
KJOR_LENGDE_KODE_GRUPPE5	-0.254146	-0.254660
KJOR_LENGDE_KODE_GRUPPE8	-0.169096	-0.167840

KJOR_LENGDE_KODE_GRUPPE16	0.187419	0.186480
KJOR_LENGDE_KODE_GRUPPE20	0.394051	0.392997
KJOR_LENGDE_KODE_GRUPPE25	0.577148	0.576531
KJOR_LENGDE_KODE_GRUPPE30	0.781146	0.779745
KJOR_LENGDE_KODE_GRUPPE40	0.973634	0.975742
KJOR_LENGDE_KODE_GRUPPE50	1.135958	1.133393
KJOR_LENGDE_KODE_GRUPPE80	1.164295	1.084264
KJOR_LENGDE_KODE_GRUPPE110	1.380942	1.381617
KJOR_LENGDE_KODE_GRUPPE999	1.400398	1.450097
MV_ALDER_GRUPPE0	-0.376189	-0.381391
MV_ALDER_GRUPPE1	-0.243635	-0.242718
MV_ALDER_GRUPPE2	-0.129841	-0.128869
MV_ALDER_GRUPPE3	-0.000168	0.003416
MV_ALDER_GRUPPE4	-0.095245	-0.096965
MV_ALDER_GRUPPE5	-0.013514	-0.008849
MV_ALDER_GRUPPE6	-0.066124	-0.068585
MV_ALDER_GRUPPE7	0.004291	0.008554
MV_ALDER_GRUPPE8	-0.062780	-0.065323
MV_ALDER_GRUPPE9	-0.004568	0.000123
MV_ALDER_GRUPPE10	-0.074153	-0.075772
MV_ALDER_GRUPPE12	-0.074153	-0.075772
MV_ALDER_GRUPPE13	-0.032363	-0.025140
MV_ALDER_GRUPPE14	-0.084460	-0.083419
MV_ALDER_GRUPPE15	-0.084460	-0.083419
MV_ALDER_GRUPPE16	-0.179677	-0.172866
MV_ALDER_GRUPPE17	-0.179677	-0.172866
MV_ALDER_GRUPPE18	-0.202709	-0.207128
MV_ALDER_GRUPPE19	-0.202709	-0.207128
MV_ALDER_GRUPPE20	-0.202709	-0.207128
MV_ALDER_GRUPPE21	-0.294810	-0.328313
MV_ALDER_GRUPPE22	-0.284352	-0.236883
MV_ALDER_GRUPPE23	-0.432580	-0.384985
MV_ALDER_GRUPPE24	-0.432580	-0.384985
MV_ALDER_GRUPPE25	-0.432580	-0.384985
MV_ALDER_GRUPPE26	-0.565371	-0.616900
MV_ALDER_GRUPPE27	-0.565371	-0.616900
MV_ALDER_GRUPPE28	-0.565371	-0.616900
MV_ALDER_GRUPPE29	-0.618002	-0.835072
MV_ALDER_GRUPPE30	-1.166946	-1.169100
MV_ALDER_GRUPPE31	-0.517436	-0.501153
ALDER_FORER_Interaction19.J	-0.047756	-0.118618
ALDER_FORER_Interaction20.J	0.308490	0.201202
ALDER_FORER_Interaction21.J	0.452943	0.367004
ALDER_FORER_Interaction22.J	0.487536	0.417482
ALDER_FORER_Interaction23.J	0.473268	0.405131
ALDER_FORER_Interaction24.J	0.382637	0.314753
ALDER_FORER_Interaction25.J	0.345222	0.277250
ALDER_FORER_Interaction26.J	0.306408	0.238734
ALDER_FORER_Interaction27.J	0.293594	0.226154
ALDER_FORER_Interaction28.J	0.251223	0.183773
ALDER_FORER_Interaction29.J	0.254270	0.186930
ALDER_FORER_Interaction30.J	0.227439	0.160191
ALDER_FORER_Interaction31.J	0.195221	0.127829
ALDER_FORER_Interaction32.J	0.193660	0.126225
ALDER_FORER_Interaction33.J	0.193844	0.126302
ALDER_FORER_Interaction34.J	0.159565	0.092118

ALDER_FORER_Interaction35.J	0.163903	0.096416
ALDER_FORER_Interaction36.J	0.153896	0.086429
ALDER_FORER_Interaction37.J	0.153136	0.085607
ALDER_FORER_Interaction38.J	0.130409	0.062782
ALDER_FORER_Interaction39.J	0.137439	0.069748
ALDER_FORER_Interaction40.J	0.107298	0.039643
ALDER_FORER_Interaction41.J	0.150036	0.082398
ALDER_FORER_Interaction42.J	0.098644	0.031032
ALDER_FORER_Interaction43.J	0.155167	0.087573
ALDER_FORER_Interaction44.J	0.115577	0.047770
ALDER_FORER_Interaction45.J	0.074612	0.007051
ALDER_FORER_Interaction46.J	0.085328	0.017808
ALDER_FORER_Interaction47.J	0.076738	0.009249
ALDER_FORER_Interaction48.J	0.095353	0.027885
ALDER_FORER_Interaction49.J	0.078747	0.011327
ALDER_FORER_Interaction50.J	0.045224	-0.022058
ALDER_FORER_Interaction51.J	0.031316	-0.035880
ALDER_FORER_Interaction52.J	0.062813	-0.004359
ALDER_FORER_Interaction53.J	0.005233	-0.061937
ALDER_FORER_Interaction54.J	-0.005895	-0.073034
ALDER_FORER_Interaction55.J	-0.030500	-0.097529
ALDER_FORER_Interaction56.J	-0.101417	-0.168520
ALDER_FORER_Interaction57.J	-0.059915	-0.127067
ALDER_FORER_Interaction58.J	-0.070922	-0.138118
ALDER_FORER_Interaction59.J	-0.040201	-0.107649
ALDER_FORER_Interaction60.J	-0.033637	-0.101034
ALDER_FORER_Interaction61.J	-0.070815	-0.138081
ALDER_FORER_Interaction62.J	-0.109369	-0.176498
ALDER_FORER_Interaction63.J	-0.093953	-0.161265
ALDER_FORER_Interaction64.J	-0.071868	-0.139237
ALDER_FORER_Interaction65.J	-0.121797	-0.189256
ALDER_FORER_Interaction66.J	-0.202681	-0.270094
ALDER_FORER_Interaction67.J	-0.205555	-0.273182
ALDER_FORER_Interaction68.J	-0.216225	-0.283499
ALDER_FORER_Interaction69.J	-0.283108	-0.350765
ALDER_FORER_Interaction70.J	-0.268698	-0.335490
ALDER_FORER_Interaction71.J	-0.330460	-0.398193
ALDER_FORER_Interaction72.J	-0.465766	-0.532755
ALDER_FORER_Interaction73.J	-0.356500	-0.423556
ALDER_FORER_Interaction74.J	-0.408203	-0.474152
ALDER_FORER_Interaction75.J	-0.532282	-0.600553
ALDER_FORER_Interaction76.J	-0.619720	-0.687148
ALDER_FORER_Interaction77.J	-0.510273	-0.577386
ALDER_FORER_Interaction78.J	-0.561314	-0.625154
ALDER_FORER_Interaction79.J	-0.561182	-0.630354
ALDER_FORER_Interaction80.J	-0.654844	-0.722765
ALDER_FORER_Interaction85.J	-0.738321	-0.805188
ALDER_FORER_Interaction90.J	-0.741514	-0.793752
ALDER_FORER_Interaction18.N	0.586950	0.514485
ALDER_FORER_Interaction19.N	0.610906	0.539619
ALDER_FORER_Interaction20.N	0.604745	0.536879
ALDER_FORER_Interaction21.N	0.576861	0.509078
ALDER_FORER_Interaction22.N	0.465628	0.397001
ALDER_FORER_Interaction23.N	0.304277	0.233272
ALDER_FORER_Interaction24.N	0.331953	0.267438
ALDER_FORER_Interaction25.N	0.336585	0.264335

ALDER_FORER_Interaction26.N	0.283006	0.215501
ALDER_FORER_Interaction27.N	0.138076	0.070681
ALDER_FORER_Interaction28.N	0.304113	0.238653
ALDER_FORER_Interaction29.N	0.055802	-0.012764
ALDER_FORER_Interaction30.N	0.164084	0.092422
ALDER_FORER_Interaction31.N	0.151002	0.083867
ALDER_FORER_Interaction32.N	0.211487	0.146252
ALDER_FORER_Interaction33.N	0.092693	0.028244
ALDER_FORER_Interaction34.N	0.236888	0.179456
ALDER_FORER_Interaction35.N	-0.097074	-0.173361
ALDER_FORER_Interaction36.N	0.170948	0.108616
ALDER_FORER_Interaction37.N	0.222321	0.153078
ALDER_FORER_Interaction38.N	0.055404	-0.011685
ALDER_FORER_Interaction39.N	0.335572	0.270413
ALDER_FORER_Interaction40.N	0.139655	0.073232
ALDER_FORER_Interaction41.N	0.272206	0.199990
ALDER_FORER_Interaction42.N	0.214804	0.148543
ALDER_FORER_Interaction43.N	0.332027	0.260453
ALDER_FORER_Interaction44.N	0.165865	0.095906
ALDER_FORER_Interaction45.N	0.232906	0.164914
ALDER_FORER_Interaction46.N	0.187873	0.119680
ALDER_FORER_Interaction47.N	0.152082	0.084672
ALDER_FORER_Interaction48.N	0.143962	0.076767
ALDER_FORER_Interaction49.N	0.140262	0.072849
ALDER_FORER_Interaction50.N	0.055971	-0.011609
ALDER_FORER_Interaction51.N	0.061410	-0.006088
ALDER_FORER_Interaction52.N	0.020156	-0.047450
ALDER_FORER_Interaction53.N	0.048668	-0.018919
ALDER_FORER_Interaction54.N	0.063660	-0.004051
ALDER_FORER_Interaction55.N	0.042993	-0.024864
ALDER_FORER_Interaction56.N	0.009302	-0.059034
ALDER_FORER_Interaction57.N	-0.049806	-0.118419
ALDER_FORER_Interaction58.N	0.023577	-0.045375
ALDER_FORER_Interaction59.N	-0.047938	-0.116237
ALDER_FORER_Interaction60.N	-0.025602	-0.092639
ALDER_FORER_Interaction61.N	-0.061467	-0.127544
ALDER_FORER_Interaction62.N	-0.068710	-0.137859
ALDER_FORER_Interaction63.N	-0.122440	-0.194265
ALDER_FORER_Interaction64.N	0.048269	-0.019601
ALDER_FORER_Interaction65.N	0.064299	-0.004023
ALDER_FORER_Interaction66.N	-0.042985	-0.110284
ALDER_FORER_Interaction67.N	-0.270217	-0.330576
ALDER_FORER_Interaction68.N	-0.074215	-0.138723
ALDER_FORER_Interaction69.N	-0.046585	-0.112058
ALDER_FORER_Interaction70.N	-0.348595	-0.439165
ALDER_FORER_Interaction71.N	-0.054032	-0.124289
ALDER_FORER_Interaction72.N	-0.304944	-0.386964
ALDER_FORER_Interaction73.N	-0.187172	-0.264283
ALDER_FORER_Interaction74.N	-0.404225	-0.461397
ALDER_FORER_Interaction75.N	-0.263390	-0.327191
ALDER_FORER_Interaction76.N	-0.406314	-0.445597
ALDER_FORER_Interaction77.N	-0.181213	-0.242607
ALDER_FORER_Interaction78.N	-1.130259	-1.101445
ALDER_FORER_Interaction79.N	-0.230937	-0.281295
ALDER_FORER_Interaction80.N	-0.702745	-0.784770
ALDER_FORER_Interaction85.N	-0.623503	-0.713142

ALDER_FORER_Interaction90.N	-0.329895	-0.362217

A4

Coefficients of the reference model.

	Reference
Intercept	-2.9676086250
BAND_AK_G_Grouping1Z	0.0844519409
BAND_AK_G_Grouping19	-0.0752009127
BAND_AK_G_Grouping12	-0.3193137431
BAND_AK_G_Grouping118	0.1625651981
BAND_AK_G_Grouping13	-0.2811221975
BAND_AK_G_Grouping122	0.4408350498
BAND_AK_G_Grouping17	-0.1414771713
BAND_AK_G_Grouping121	0.4016074879
BAND_AK_G_Grouping16	-0.1867964887
BAND_AK_G_Grouping123	0.4964815824
BAND_AK_G_Grouping113	0.0247466902
BAND_AK_G_Grouping112	0.0036810828
BAND_AK_G_Grouping15	-0.1847284054
BAND_AK_G_Grouping119	0.2443255539
BAND_AK_G_Grouping125	0.7153991287
BAND_AK_G_Grouping18	-0.0867514737
BAND_AK_G_Grouping117	0.1703833318
BAND_AK_G_Grouping124	0.5286775109
BAND_AK_G_Grouping114	0.0415004758
BAND_AK_G_Grouping116	0.1421402881
BAND_AK_G_Grouping14	-0.2284501524
BAND_AK_G_Grouping115	0.0605512248
BAND_AK_G_Grouping120	0.3335742429
BAND_AK_G_Grouping110	-0.0404221034
BAND_AK_G_Grouping11	-0.3923106690
DIVISJON_KODE_Grouping1Z	0.0980167596
DIVISJON_KODE_Grouping1L	-0.1721597172
DRIVSTOFF_KODE_Grouping8DRIVSTOFF_KODE_G01	-0.1561331656
DRIVSTOFF_KODE_Grouping8DRIVSTOFF_KODE_G02	-0.1082853799
DRIVSTOFF_KODE_Grouping8DRIVSTOFF_KODE_G03	-0.1877912701
DRIVSTOFF_KODE_Grouping8DRIVSTOFF_KODE_G04	-0.2582087311
IMPORT_FLAGG_Grouping1J	0.0579595944
LEASING_FLAGG_Grouping1J	0.2400657536
MERKE_KLASSE_KODE_NY_Grouping1AA	-0.4817853359
MERKE_KLASSE_KODE_NY_Grouping1AB	-0.1185160198
MERKE_KLASSE_KODE_NY_Grouping1AC	-0.1346975146

MERKE_KLASSE_KODE_NY_Grouping1AD	-0.2734796042
MERKE_KLASSE_KODE_NY_Grouping1AE	-0.2109871528
MERKE_KLASSE_KODE_NY_Grouping1AF	0.0158292890
MERKE_KLASSE_KODE_NY_Grouping1AG	0.0547557452
MERKE_KLASSE_KODE_NY_Grouping1AH	-0.0517705448
MERKE_KLASSE_KODE_NY_Grouping1AI	-0.1179918071
MERKE_KLASSE_KODE_NY_Grouping1AJ	0.0438746782
MERKE_KLASSE_KODE_NY_Grouping1AK	-0.2087193566
MERKE_KLASSE_KODE_NY_Grouping1AL	-0.0955918581
MERKE_KLASSE_KODE_NY_Grouping1AM	-0.1388757111
MERKE_KLASSE_KODE_NY_Grouping1AN	0.0237822574
MERKE_KLASSE_KODE_NY_Grouping1AO	-0.1127384457
MERKE_KLASSE_KODE_NY_Grouping1AP	-0.1287337756
MERKE_KLASSE_KODE_NY_Grouping1AR	-0.2715051484
MERKE_KLASSE_KODE_NY_Grouping1AS	-0.0062452968
MERKE_KLASSE_KODE_NY_Grouping1BA	-0.5023721652
MERKE_KLASSE_KODE_NY_Grouping1BB	-0.2577364256
MERKE_KLASSE_KODE_NY_Grouping1BC	-0.0844411045
MERKE_KLASSE_KODE_NY_Grouping1BD	-0.2444026808
MERKE_KLASSE_KODE_NY_Grouping1BE	-0.0282357282
MERKE_KLASSE_KODE_NY_Grouping1BF	-0.3295652120
MERKE_KLASSE_KODE_NY_Grouping1BG	-0.7327637062
MERKE_KLASSE_KODE_NY_Grouping1BH	-0.0672665516
MERKE_KLASSE_KODE_NY_Grouping1BI	-0.3059777091
MERKE_KLASSE_KODE_NY_Grouping1SA	-0.2248378915
MERKE_KLASSE_KODE_NY_Grouping1SB	-0.9901560679
MERKE_KLASSE_KODE_NY_Grouping1ZA	-0.0976040734
MERKE_KLASSE_KODE_NY_Grouping1ZB	0.3133385624
MERKE_KLASSE_KODE_NY_Grouping1ZC	-0.1135169024
MERKE_KLASSE_KODE_NY_Grouping1ZD	-0.0448643643
MERKE_KLASSE_KODE_NY_Grouping1ZE	-0.2524021275
MERKE_KLASSE_KODE_NY_Grouping1ZF	-0.0365767977
MERKE_KLASSE_KODE_NY_Grouping1ZG	-0.1708782646
MERKE_KLASSE_KODE_NY_Grouping1ZH	-0.2469513478
MERKE_KLASSE_KODE_NY_Grouping1ZI	-0.3653278637
MERKE_KLASSE_KODE_NY_Grouping1ZJ	-0.3733106579
MERKE_KLASSE_KODE_NY_Grouping1ZK	0.2898278079
MERKE_KLASSE_KODE_NY_Grouping1ZL	-0.3189539660
MERKE_KLASSE_KODE_NY_Grouping1ZM	-0.0927670166
MERKE_KLASSE_KODE_NY_Grouping1ZO	-0.1494628319
MERKE_KLASSE_KODE_NY_Grouping1ZP	0.1791593400
SUBCLUSTER_Grouping112	0.1612286111
SUBCLUSTER_Grouping12	-0.0656034321
SUBCLUSTER_Grouping13	-0.1529261742
SUBCLUSTER_Grouping19	-0.0218009691
SUBCLUSTER_Grouping11	0.0640302164
SUBCLUSTER_Grouping111	0.0055672732
SUBCLUSTER_Grouping18	0.0009105251

SUBCLUSTER_Grouping110	0.0042430586
SUBCLUSTER_Grouping15	0.0256452344
SUBCLUSTER_Grouping14	-0.1522708111
SUBCLUSTER_Grouping16	-0.0540593534
EFFEKT_HK_Grouping1EFFEKT_HK_G01	-0.2409274646
EFFEKT_HK_Grouping1EFFEKT_HK_G02	-0.0994246779
EFFEKT_HK_Grouping1EFFEKT_HK_G03	-0.0109132382
EFFEKT_HK_Grouping1EFFEKT_HK_G04	0.0104021789
EFFEKT_HK_Grouping1EFFEKT_HK_G05	-0.0657068107
EFFEKT_HK_Grouping1EFFEKT_HK_G06	-0.0456724959
EFFEKT_HK_Grouping1EFFEKT_HK_G07	-0.0235560260
EFFEKT_HK_Grouping1EFFEKT_HK_G08	0.0344946323
EFFEKT_HK_Grouping1EFFEKT_HK_G09	0.0590623399
EFFEKT_HK_Grouping1EFFEKT_HK_G10	0.0601125458
EFFEKT_HK_Grouping1EFFEKT_HK_G11	0.1797179479
EFFEKT_HK_Grouping1EFFEKT_HK_G12	0.0936014584
EFFEKT_HK_Grouping1EFFEKT_HK_G13	0.1629708124
EFFEKT_HK_Grouping1EFFEKT_HK_G14	0.1368303256
EFFEKT_HK_Grouping1EFFEKT_HK_G15	0.1677638944
EFFEKT_HK_Grouping1EFFEKT_HK_G16	0.2712279159
EFFEKT_HK_Grouping1EFFEKT_HK_G17	0.2722784506
EFFEKT_HK_Grouping1EFFEKT_HK_G18	0.2670458075
EFFEKT_HK_Grouping1EFFEKT_HK_G19	0.3032943403
EFFEKT_HK_Grouping1EFFEKT_HK_G20	0.2943361826
EFFEKT_HK_Grouping1EFFEKT_HK_G21	0.2839250434
EFFEKT_HK_Grouping1EFFEKT_HK_G22	0.2294898432
EFFEKT_HK_Grouping1EFFEKT_HK_G23	0.3938214473
EFFEKT_HK_Grouping1EFFEKT_HK_G24	0.4497735931
EGENVEKT_Grouping1EGENVEKT_G01	-0.9940052903
EGENVEKT_Grouping1EGENVEKT_G02	-0.4373411729
EGENVEKT_Grouping1EGENVEKT_G03	-0.1300383047
EGENVEKT_Grouping1EGENVEKT_G04	-0.1467152207
EGENVEKT_Grouping1EGENVEKT_G05	-0.0518729194
EGENVEKT_Grouping1EGENVEKT_G06	0.0190214043
EGENVEKT_Grouping1EGENVEKT_G07	0.0677554681
EGENVEKT_Grouping1EGENVEKT_G08	0.1178102137
EGENVEKT_Grouping1EGENVEKT_G09	0.1513739853
EGENVEKT_Grouping1EGENVEKT_G10	0.1704854166
EGENVEKT_Grouping1EGENVEKT_G11	0.1405956199
EGENVEKT_Grouping1EGENVEKT_G12	0.2065242142
EGENVEKT_Grouping1EGENVEKT_G13	0.1707134416
EGENVEKT_Grouping1EGENVEKT_G14	0.2580567519
EGENVEKT_Grouping1EGENVEKT_G15	0.2817406761
EGENVEKT_Grouping1EGENVEKT_G16	-0.1293204124
KJOR_LENGDE_KODE_Grouping2KJOR_LENGDE_KODE_G01	-7.5053961221
KJOR_LENGDE_KODE_Grouping2KJOR_LENGDE_KODE_G02	-0.2549354525
KJOR_LENGDE_KODE_Grouping2KJOR_LENGDE_KODE_G03	-0.1679325095
KJOR_LENGDE_KODE_Grouping2KJOR_LENGDE_KODE_G04	0.1864156465

KJOR_LENGDE_KODE_Grouping2KJOR_LENGDE_KODE_G05	0.3924743799
KJOR_LENGDE_KODE_Grouping2KJOR_LENGDE_KODE_G06	0.5761008250
KJOR_LENGDE_KODE_Grouping2KJOR_LENGDE_KODE_G07	0.7791373328
KJOR_LENGDE_KODE_Grouping2KJOR_LENGDE_KODE_G08	0.9757461831
KJOR_LENGDE_KODE_Grouping2KJOR_LENGDE_KODE_G09	1.1416531225
MV_ALDER_Grouping10	-0.3836161367
MV_ALDER_Grouping11	-0.2423227689
MV_ALDER_Grouping110	-0.0717086408
MV_ALDER_Grouping112	-0.0805578889
MV_ALDER_Grouping113	-0.0240763748
MV_ALDER_Grouping114	-0.0891989710
MV_ALDER_Grouping115	-0.0726886562
MV_ALDER_Grouping116	-0.1655244053
MV_ALDER_Grouping117	-0.1769695626
MV_ALDER_Grouping118	-0.2134672360
MV_ALDER_Grouping119	-0.1963917017
MV_ALDER_Grouping12	-0.1277186575
MV_ALDER_Grouping13	0.0041021660
MV_ALDER_Grouping14	-0.0960722423
MV_ALDER_Grouping15	-0.0076292131
MV_ALDER_Grouping16	-0.0674647295
MV_ALDER_Grouping17	0.0096461265
MV_ALDER_Grouping18	-0.0644954431
MV_ALDER_Grouping19	0.0003951305
MV_ALDER_Grouping1MV_ALDER_G01	-0.5034236592
MV_ALDER_Grouping1MV_ALDER_G02	-0.2773883876
MV_ALDER_Grouping1MV_ALDER_G03	-0.5660267602
MV_ALDER_Grouping1MV_ALDER_G04	-1.1429010938
ALDER_FORER_Interaction19.N	0.5512270280
ALDER_FORER_Interaction20.N	0.5476926021
ALDER_FORER_Interaction21.N	0.5201113981
ALDER_FORER_Interaction22.N	0.4079626926
ALDER_FORER_Interaction23.N	0.2447602470
ALDER_FORER_Interaction24.N	0.2785020135
ALDER_FORER_Interaction25.N	0.2765059649
ALDER_FORER_Interaction26.N	0.2259044414
ALDER_FORER_Interaction27.N	0.0815283651
ALDER_FORER_Interaction28.N	0.2494172408
ALDER_FORER_Interaction29.N	-0.0007962183
ALDER_FORER_Interaction30.N	0.1048624314
ALDER_FORER_Interaction31.N	0.0959737373
ALDER_FORER_Interaction32.N	0.1561961856
ALDER_FORER_Interaction33.N	0.0373751860
ALDER_FORER_Interaction34.N	0.1912016352
ALDER_FORER_Interaction35.N	-0.1609059075
ALDER_FORER_Interaction36.N	0.1223096324
ALDER_FORER_Interaction37.N	0.1641404634
ALDER_FORER_Interaction38.N	0.0006367598

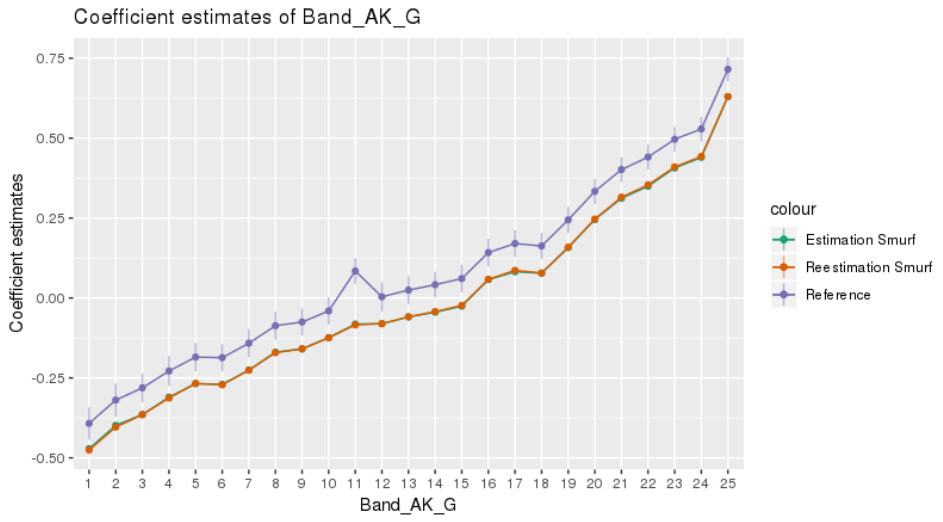
ALDER_FOREROR_Interaction39.N	0.2802015163
ALDER_FOREROR_Interaction4.N	-7.9166630986
ALDER_FOREROR_Interaction40.N	0.0836642006
ALDER_FOREROR_Interaction41.N	0.2105510646
ALDER_FOREROR_Interaction42.N	0.1590702734
ALDER_FOREROR_Interaction43.N	0.2715080471
ALDER_FOREROR_Interaction44.N	0.1069423336
ALDER_FOREROR_Interaction45.N	0.1762531660
ALDER_FOREROR_Interaction46.N	0.1310537966
ALDER_FOREROR_Interaction47.N	0.0956154296
ALDER_FOREROR_Interaction48.N	0.0881181041
ALDER_FOREROR_Interaction49.N	0.0844668593
ALDER_FOREROR_Interaction5.N	-6.1380647744
ALDER_FOREROR_Interaction51.N	0.0056635331
ALDER_FOREROR_Interaction52.N	-0.0356932825
ALDER_FOREROR_Interaction53.N	-0.0069448439
ALDER_FOREROR_Interaction54.N	0.0077419365
ALDER_FOREROR_Interaction55.N	-0.0137628862
ALDER_FOREROR_Interaction56.N	-0.0477238296
ALDER_FOREROR_Interaction57.N	-0.1064556023
ALDER_FOREROR_Interaction58.N	-0.0336747073
ALDER_FOREROR_Interaction59.N	-0.1052388119
ALDER_FOREROR_Interaction6.N	-6.1649902175
ALDER_FOREROR_Interaction60.N	-0.0804836191
ALDER_FOREROR_Interaction61.N	-0.1154042416
ALDER_FOREROR_Interaction62.N	-0.1256229361
ALDER_FOREROR_Interaction63.N	-0.1814343619
ALDER_FOREROR_Interaction64.N	-0.0080437745
ALDER_FOREROR_Interaction65.N	0.0078079844
ALDER_FOREROR_Interaction66.N	-0.0982840596
ALDER_FOREROR_Interaction67.N	-0.3188132206
ALDER_FOREROR_Interaction68.N	-0.1258196084
ALDER_FOREROR_Interaction69.N	-0.0994992150
ALDER_FOREROR_Interaction7.N	-6.5594002478
ALDER_FOREROR_Interaction70.N	-0.4265145066
ALDER_FOREROR_Interaction71.N	-0.1138060357
ALDER_FOREROR_Interaction72.N	-0.3752468144
ALDER_FOREROR_Interaction73.N	-0.2537065807
ALDER_FOREROR_Interaction74.N	-0.4492363650
ALDER_FOREROR_Interaction75.N	-0.3158276268
ALDER_FOREROR_Interaction76.N	-0.4352614019
ALDER_FOREROR_Interaction77.N	-0.2333732388
ALDER_FOREROR_Interaction78.N	-1.0897875572
ALDER_FOREROR_Interaction79.N	-0.2709617806
ALDER_FOREROR_Interaction8.N	-7.2515899398
ALDER_FOREROR_InteractionALDER_FTAKER_G01.N	0.5308909543
ALDER_FOREROR_InteractionALDER_FTAKER_G02.N	-0.7745870128
ALDER_FOREROR_InteractionALDER_FTAKER_G03.N	-0.7004405530

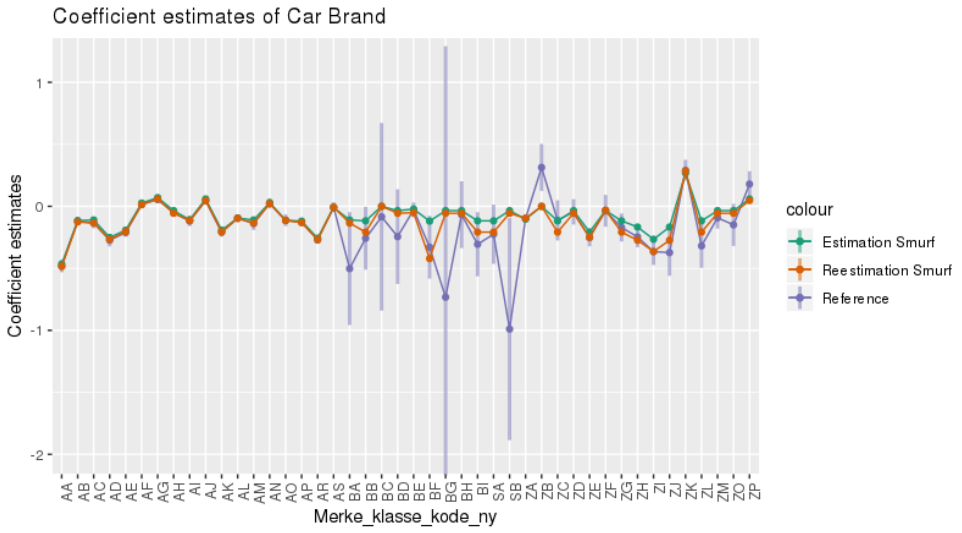
ALDER_FORER_InteractionALDER_FTAKER_G04.N	-0.4088095171
ALDER_FORER_Interaction50.J	-0.0110232381
ALDER_FORER_Interaction19.J	-0.1075474734
ALDER_FORER_Interaction2.J	1.0747518868
ALDER_FORER_Interaction20.J	0.2054193793
ALDER_FORER_Interaction21.J	0.3739767152
ALDER_FORER_Interaction22.J	0.4283439286
ALDER_FORER_Interaction23.J	0.4161701972
ALDER_FORER_Interaction24.J	0.3256360849
ALDER_FORER_Interaction25.J	0.2883979261
ALDER_FORER_Interaction26.J	0.2501422207
ALDER_FORER_Interaction27.J	0.2374356958
ALDER_FORER_Interaction28.J	0.1949091821
ALDER_FORER_Interaction29.J	0.1979893515
ALDER_FORER_Interaction3.J	1.4123335733
ALDER_FORER_Interaction30.J	0.1711448022
ALDER_FORER_Interaction31.J	0.1387997813
ALDER_FORER_Interaction32.J	0.1374674501
ALDER_FORER_Interaction33.J	0.1375252940
ALDER_FORER_Interaction34.J	0.1029643419
ALDER_FORER_Interaction35.J	0.1071369936
ALDER_FORER_Interaction36.J	0.0969251484
ALDER_FORER_Interaction37.J	0.0961849293
ALDER_FORER_Interaction38.J	0.0727741050
ALDER_FORER_Interaction39.J	0.0799005354
ALDER_FORER_Interaction4.J	1.8048501102
ALDER_FORER_Interaction40.J	0.0497049459
ALDER_FORER_Interaction41.J	0.0926555183
ALDER_FORER_Interaction42.J	0.0410382686
ALDER_FORER_Interaction43.J	0.0977203411
ALDER_FORER_Interaction44.J	0.0581012438
ALDER_FORER_Interaction45.J	0.0176736554
ALDER_FORER_Interaction46.J	0.0284301736
ALDER_FORER_Interaction47.J	0.0202138966
ALDER_FORER_Interaction48.J	0.0389873408
ALDER_FORER_Interaction49.J	0.0223311207
ALDER_FORER_Interaction5.J	-7.2183397797
ALDER_FORER_Interaction51.J	-0.0247876321
ALDER_FORER_Interaction52.J	0.0067589263
ALDER_FORER_Interaction53.J	-0.0506516669
ALDER_FORER_Interaction54.J	-0.0616035158
ALDER_FORER_Interaction55.J	-0.0862359165
ALDER_FORER_Interaction56.J	-0.1566520525
ALDER_FORER_Interaction57.J	-0.1151298847
ALDER_FORER_Interaction58.J	-0.1264621407
ALDER_FORER_Interaction59.J	-0.0954639821
ALDER_FORER_Interaction6.J	1.4806219567
ALDER_FORER_Interaction60.J	-0.0889089510

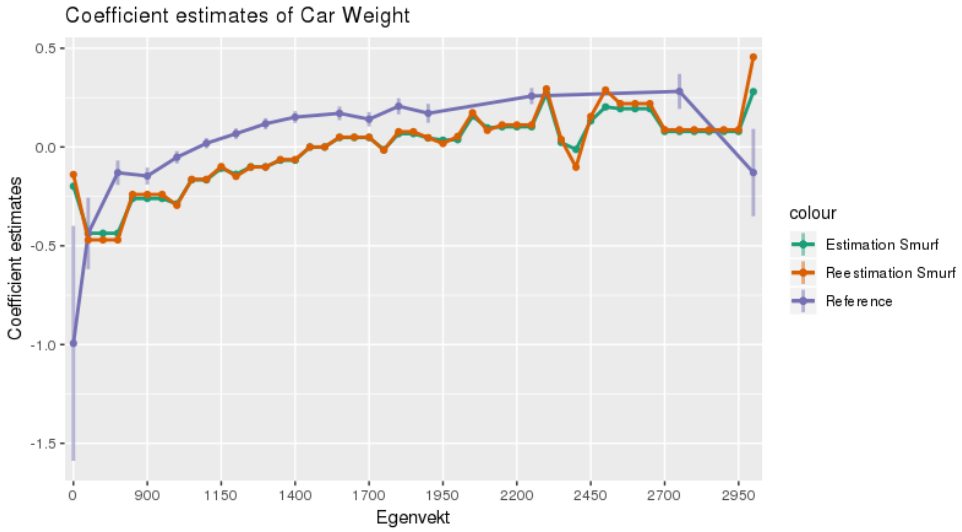
ALDER_FORER_Interaction61.J	-0.1262064881
ALDER_FORER_Interaction62.J	-0.1647287699
ALDER_FORER_Interaction63.J	-0.1494483174
ALDER_FORER_Interaction64.J	-0.1273154778
ALDER_FORER_Interaction65.J	-0.1773746203
ALDER_FORER_Interaction66.J	-0.2581408052
ALDER_FORER_Interaction67.J	-0.2607849148
ALDER_FORER_Interaction68.J	-0.2712930697
ALDER_FORER_Interaction69.J	-0.3386968640
ALDER_FORER_Interaction7.J	-7.0208919639
ALDER_FORER_Interaction70.J	-0.3236806095
ALDER_FORER_Interaction71.J	-0.3863763406
ALDER_FORER_Interaction72.J	-0.5210396152
ALDER_FORER_Interaction73.J	-0.4119993955
ALDER_FORER_Interaction74.J	-0.4628636045
ALDER_FORER_Interaction75.J	-0.5891204540
ALDER_FORER_Interaction76.J	-0.6756901470
ALDER_FORER_Interaction77.J	-0.5660134477
ALDER_FORER_Interaction78.J	-0.6133837234
ALDER_FORER_Interaction79.J	-0.6194511007
ALDER_FORER_Interaction8.J	0.4904805544
ALDER_FORER_InteractionALDER_FTAKER_G01.J	-0.1609776843
ALDER_FORER_InteractionALDER_FTAKER_G02.J	-0.7109276587
ALDER_FORER_InteractionALDER_FTAKER_G03.J	-0.7938495021
ALDER_FORER_InteractionALDER_FTAKER_G04.J	-0.7753851540

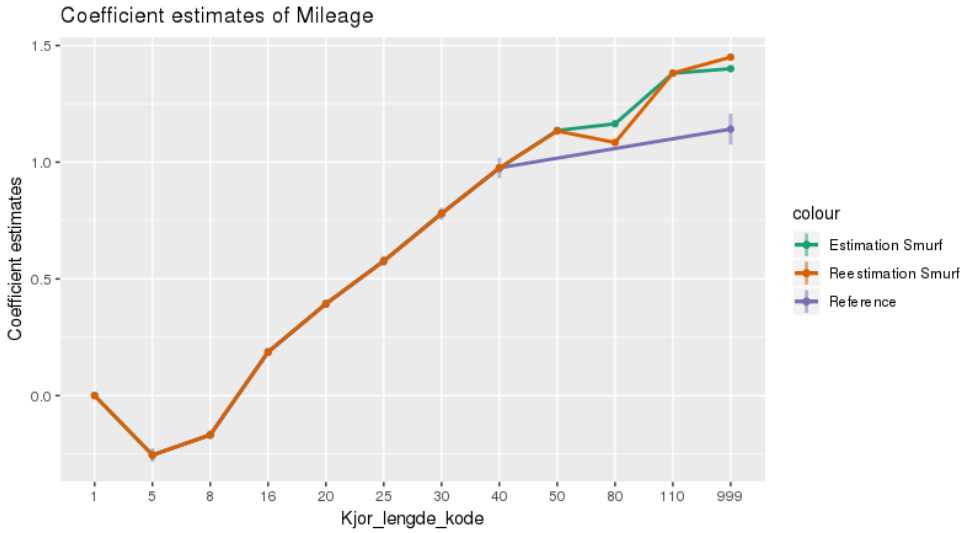
A5

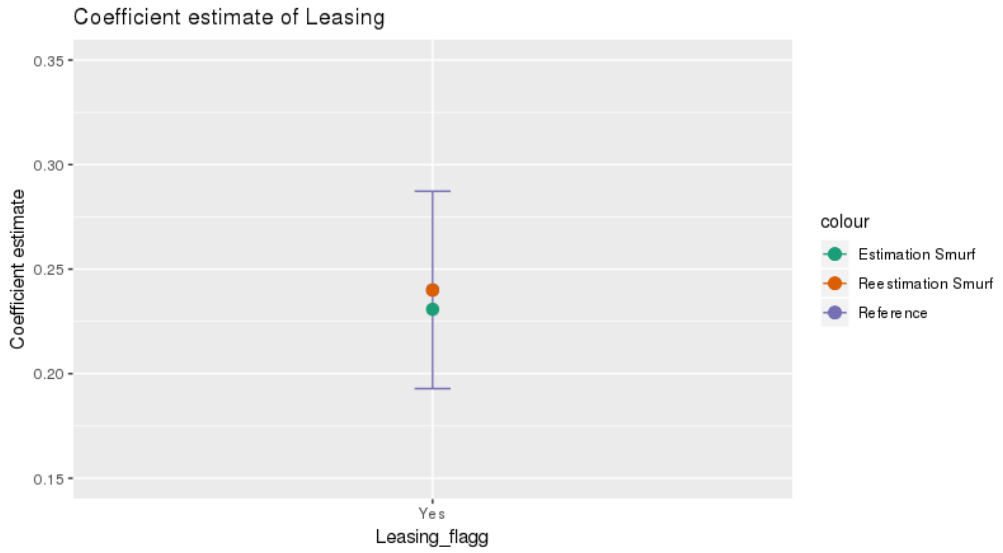
Plots of coefficient estimates for all three models of remaining variables.

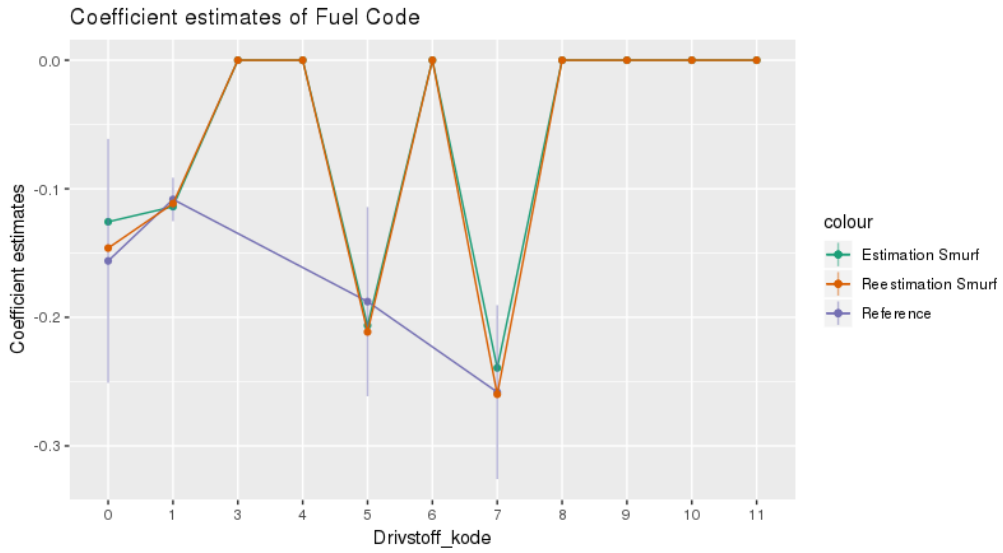


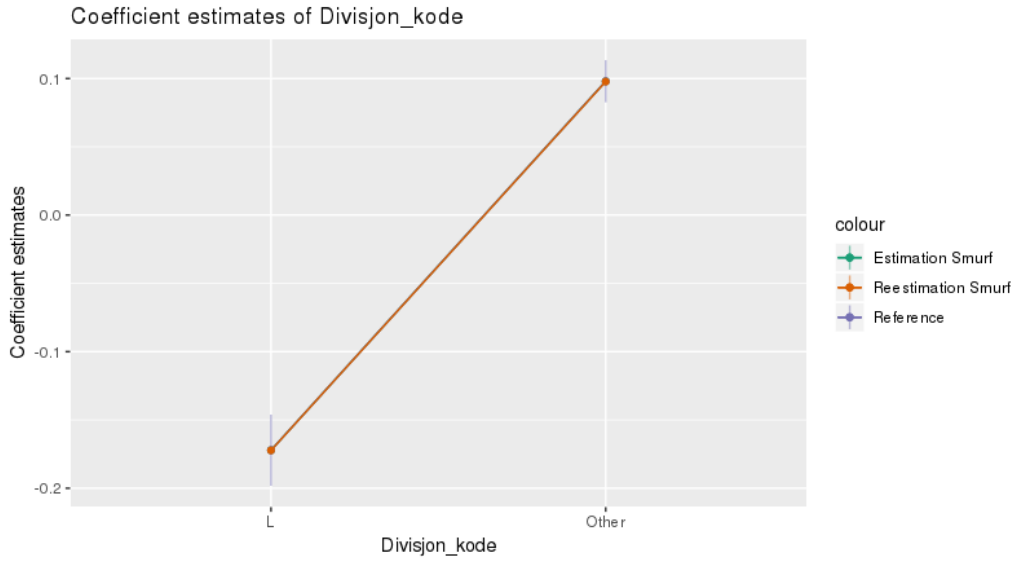

Coefficient estimates of Car Brand
Coefficient estimates
1
0
-1
-2
AA AB AC AD AE AF AG AH AI AJ AK AL AM AN AO AP AR AS BA BB BC BD BE BF BG BH BI SA SB ZA ZB ZC ZD ZE ZF ZG ZH ZI ZJ ZK ZL ZM ZO ZP
Merke_klasse_kode_ny
colour
Estimation Smurf
Reestimation Smurf
Reference


Coefficient estimates of Car Weight
Coefficient estimates
0.5
0.0
-0.5
-1.0
-1.5
0 900 1150 1400 1700 1950 2200 2450 2700 2950
Egenvekt
colour
Estimation Smurf
Reestimation Smurf
Reference


Coefficient estimates of Mileage
Coefficient estimates
1.5
1.0
0.5
0.0
1 5 8 16 20 25 30 40 50 80 110 999
Kjor_lengde_kode
colour
Estimation Smurf
Reestimation Smurf
Reference


Coefficient estimate of Leasing
Coefficient estimate
0.35
0.30
0.25
0.20
0.15
Yes
Leasing_flagg
colour
Estimation Smurf
Reestimation Smurf
Reference


Coefficient estimates of Fuel Code
Coefficient estimates
0.0
-0.1
-0.2
-0.3
0
1
3
4
5
6
7
8
9
10
11
Drivstoff_kode
colour
Estimation Smurf
Reestimation Smurf
Reference


Coefficient estimates of Divisjon_kode
Coefficient estimates
0.1
0.0
-0.1
-0.2
L
Other
Divisjon_kode
colour
Estimation Smurf
Reestimation Smurf
Reference