Assignment 1

Jakob og Oskar

2025

Assignment 1 in interpretable machine learning

We are tasked with building a predictive regression model, with the best possible prediction. This submission will be split up into several parts:

- Initial data preprocessing and overview
- Introduction into the mathematics
- Modelling and justification
- Comparative discussion

Initial data preprocessing and overview

The given data has the form:

```
##  [1] "ID"                  "Date_start_contract"  "Date_last_renewal"
##  [4] "Date_next_renewal"   "Date_birth"           "Date_driving_licence"
##  [7] "Distribution_channel" "Seniority"           "Policies_in_force"
## [10] "Max_policies"        "Max_products"         "Lapse"
## [13] "Date_lapse"          "Payment"              "Premium"
## [16] "Cost_claims_year"    "N_claims_year"        "N_claims_history"
## [19] "R_Claims_history"    "Type_risk"            "Area"
## [22] "Second_driver"       "Year_matriculation"   "Power"
## [25] "Cylinder_capacity"   "Value_vehicle"        "N_doors"
## [28] "Type_fuel"           "Length"               "Weight"
```

We are asked to predict the **Cost_claims_year** given the rest of the covariate-vector. Initially it is important to note that out data is a classical insurance dataset, where we are given rows corresponding to insurance periode for a given contract. There are several issues with this, since some contracts might overlap into multiple contracts, which can be identified by the ID. It is however very difficult to locate these policies, and we overlook this issue.

There are numerous char. vectors in the data, which can be seen here:

```
## Classes 'data.table' and 'data.frame':   105555 obs. of  30 variables:
##  $ ID                  : int  1 1 1 1 2 2 3 3 3 3 ...
##  $ Date_start_contract : chr  "05/11/2015" "05/11/2015" "05/11/2015" "05/11/2015" ...
##  $ Date_last_renewal   : chr  "05/11/2015" "05/11/2016" "05/11/2017" "05/11/2018" ...
##  $ Date_next_renewal   : chr  "05/11/2016" "05/11/2017" "05/11/2018" "05/11/2019" ...
##  $ Date_birth          : chr  "15/04/1956" "15/04/1956" "15/04/1956" "15/04/1956" ...
##  $ Date_driving_licence: chr  "20/03/1976" "20/03/1976" "20/03/1976" "20/03/1976" ...
```

```
##  $ Distribution_channel: chr  "0" "0" "0" "0" ...
##  $ Seniority           : int  4 4 4 4 4 4 15 15 15 15 ...
##  $ Policies_in_force   : int  1 1 2 2 2 2 1 1 1 1 ...
##  $ Max_policies        : int  2 2 2 2 2 2 2 2 2 2 ...
##  $ Max_products        : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Lapse               : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Date_lapse          : chr  "" "" "" "" ...
##  $ Payment             : int  0 0 0 0 1 1 0 0 0 0 ...
##  $ Premium             : num  223 214 215 217 214 ...
##  $ Cost_claims_year    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ N_claims_year       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ N_claims_history    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ R_Claims_history    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Type_risk           : int  1 1 1 1 1 1 3 3 3 3 ...
##  $ Area                : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Second_driver       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Year_matriculation  : int  2004 2004 2004 2004 2004 2004 2013 2013 2013 2013 ...
##  $ Power               : int  80 80 80 80 80 80 85 85 85 85 ...
##  $ Cylinder_capacity   : int  599 599 599 599 599 599 1229 1229 1229 1229 ...
##  $ Value_vehicle       : num  7068 7068 7068 7068 7068 ...
##  $ N_doors             : int  0 0 0 0 0 0 5 5 5 5 ...
##  $ Type_fuel           : chr  "P" "P" "P" "P" ...
##  $ Length              : num  NA NA NA NA NA ...
##  $ Weight              : int  190 190 190 190 190 190 1105 1105 1105 1105 ...
##  - attr(*, ".internal.selfref")=<externalptr>
## NULL
```

One could, model some of the char. vectors like proposed in the lectures, by

$$\begin{aligned} X_i &= E(Y \mid X_i) \\ &= \int_Y y \; \mu(x, dy) \end{aligned}$$

where μ is the appropriate probability kernel, and we let y be our response. However, we choose to take the our char. vectors and round them to yearly values, which then has a ordinal ordering and can thus be used as features. Further we take and one-hot encode *Distribution_channel* by creating three new features which are either 1 or 0. Same for the type of fuel.

Next there are some missing values.

```
##       Type_fuel Length
## 95226         1      1     0
## 8565          1      0     1
## 1764          0      0     2
##            1764  10329 12093
```

It becomes apparent that there are missing values in the length and type_fuel variable. We can see that the missingness is overlapping in 1764 rows. However the length variable suffers way heavier from missingnes compared to type_fuel. In order to impute values, we assume the missingnes it completely at random for both features. We consider correlated features to do imputation. We see from the correlation plot (fig. 3) that type_fuel is mostly correlated with Cylinder_capacity, Value_vehicle and Weight. The same goes for the feature Length. Therefore, we fit a multivariate linear regression model with these 3 covariates to

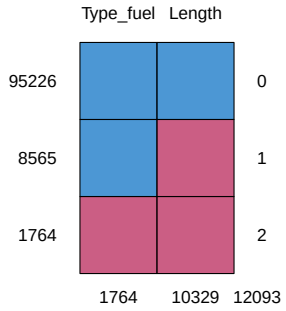

Figure 1: Missing data plot, right axis shows numer of missing columns in that row, and the left axis show how many rows have this missingness pattern

predict both type_fuel and Length (using cbind in the response formula). Finally, we predict the missing values using this trained model.

Our respone variable **Cost_claims_year** suffers from a few extreme values. We decide to remove these values to later on acheive a better model fit. In fig. 2, **N_claims_year** is plotted against **Cost_claims_year**, where at least 5-10 extreme values of **Cost_claims_year* are spotted.

Finally we look at the correlation between the covariates.

We notice some clusters, most meaningfull between *Cylinder_capacity*, *Value_vehicle* and *Weight* which is expected. We deem these to have significant predictive ability, and thus we choose to not remove these. The top left cluster, will be ignored for now, since we will later introduce a data-transformation which affects this cluster in a high degree.

Data Aggregation For our data aggregation we want to have the ability to assume independence, which we have if we aggregate the rows into being one policy. There are several problems to tacle to achive data in the desired format. We have to both choose an appropriate aggreations function, and we have to consider events where the insured object change. We start off by identifying some unique-identifiers. If theme columns change, we deem the contract to insure a new car, or at least insure something that is independent of what was previously insured. Amongst these unique identifiers are Length, Weight, Power etc. so things which change, probably mean that the thing that is insured has changes. \ Next, we look at how we have aggregated data. Firstly we sum the exposure for each contract, since we will use this to weight in the models later on. Next we use the max function on the number of policies, lapse, date of birth, date of driving licence, start of contract, products, payments, ratio of claims history, type of risk,. For The mean premium we use the mean of the premium without the forward premium, that is the premium for the time we want to predict for. For the value of the veichle, and number of doors we use mean. For the length we use sum since this is just a scaling. For the variable *cost_claims_this_year* we use the cost of the claims in $\mathcal{F}_t$ where we are to predict data on $\mathcal{F}_{t+1}$.

So we have $Z_i(\omega_i) := (X_i(\omega_i), Y_i(\omega_i)) : (\Omega_i, \mathcal{F}_i) \to (\mathcal{X} \times \mathcal{Y}, \mathcal{B}(\mathcal{X} \times \mathcal{Y}))$, where by aggregating data row-wise we obtain

$$P\big(Z_i \in A,\ Z_k \in B\big) = P\big(Z_i \in A\big)\, P\big(Z_k \in B\big), \quad A, B \in \mathcal{B}(\mathcal{X} \times \mathcal{Y}).$$

However this is an empirical assumption which can be violated by catastrophe events.

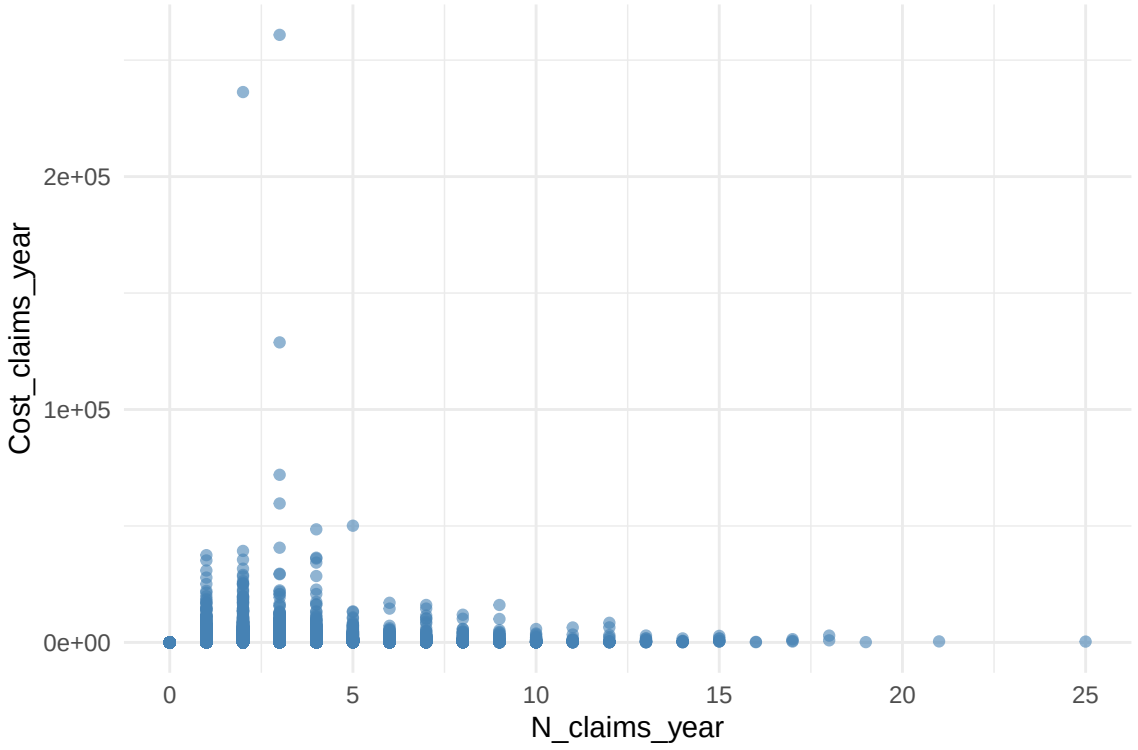

Figure 2: Claim costs over number of claims

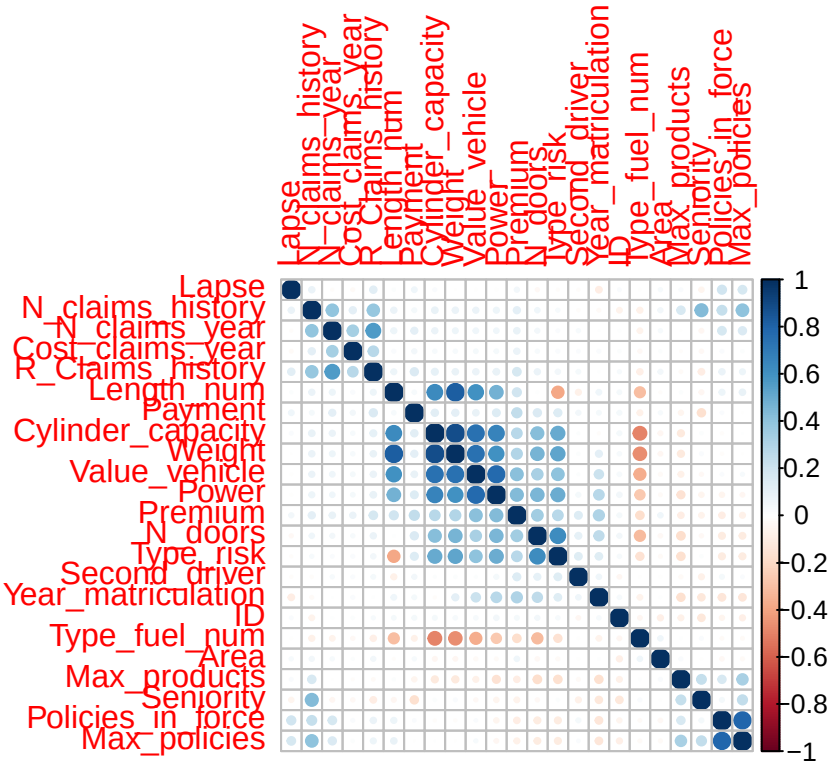

Figure 3: Correlation plot between the continious covariates

The aggregating is done by formatting the date covariates such that we can calculate the contract exposure, where we note that one year means $E = 1$, and then sum over the entire *ID*. Futher we take the premium, for the second to last entry, since we dont want to have forward-leakage in our data. The same is done for *cost_claims_history*, and related rows. We apply a mix of sum and max functions on the remaining rows to ensure that we aggregate data.

Introduction into the mathematical framework

Here we desire some mathematics before we proceed. It could be desireable to look at a classical insurance related result

$$\begin{aligned} E(S(t) \mid Z = z) &= E\left(\sum_{i=1}^{N(t)} X_i \mid Z = z\right) \\ &= E\left(\left(\sum_{i=1}^{N(t)} X_i \mid Z = z \cap \mathcal{F}(t)\right)\right) \\ &= E\left(NE\left(X_i \mid Z = z\right) \mid Z = z\right) \\ &= E(N \mid Z = z)E(X_1 \mid Z = z) \end{aligned}$$

Further we define the Tweedie law as

$$P_{\theta,\sigma^2}(Y \in A) = \int_A \exp\left\{\frac{\theta z - \kappa_p(\theta)}{\sigma^2}\right\} \nu_y(dz)$$

since it can accommodate a zero-inflated distribution very well. Lastly we introduce the so-called *bbrier* measure, for classification models.

$$\frac{1}{n}\sum_{i=1}^{n} w_i(1\{X_i = 1\} - P(X_i = 1))^2$$

Which is the mean weighted square euclidean distance between the probability and the true value. ## Modelling and justification

Tweedie Initially we want to fit a Tweedie model on the entire data, since the Tweedie model is known for handeling zero-inflated distributions well. We model with a gradient boosting machine, since we have a couple of parameters and the *lightgbm* provides a solid bias-variance trade-off. The additive structure of the gradient boosting is preferable over the random forest. We minimize the mse, and find that the optimal variance power through cross-validation is approximately 1.82. This makes the tweedie a compound Poisson-Gamma. This optimal model is found with 19 leaves.

The random search is not optimal, and one could have used a better optimization algorithm. We can look at the model fit, on the whole dataset.

This fit is somewhat disappointing since we severely underestimate the larger claims. It is also evident that our data is very heavy tailed, and the tweedie does not fully capture the more extreme events.

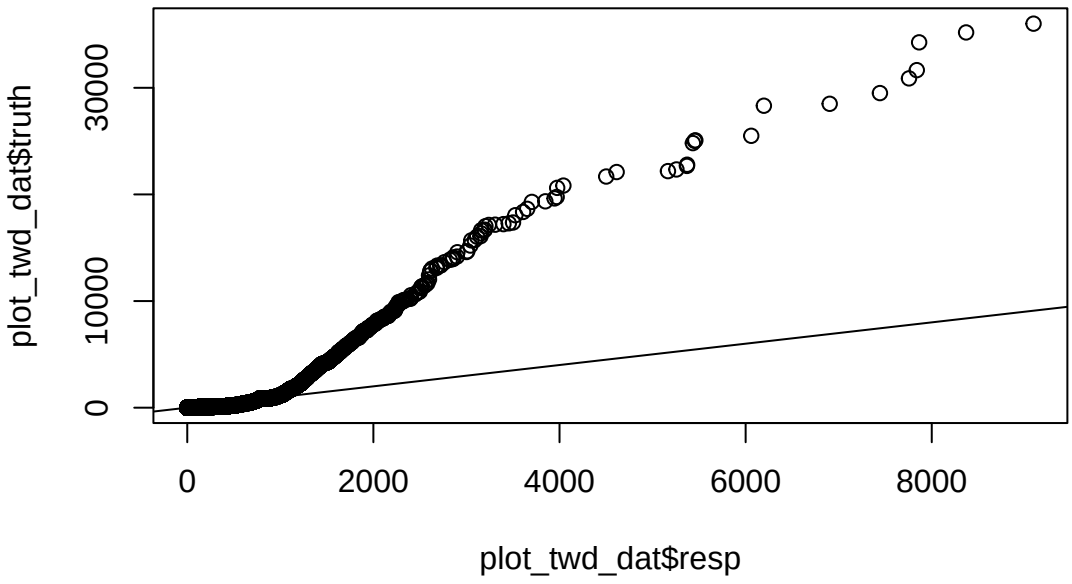

Figure 4: Residual plot, histogram for residuals, QQ-plot

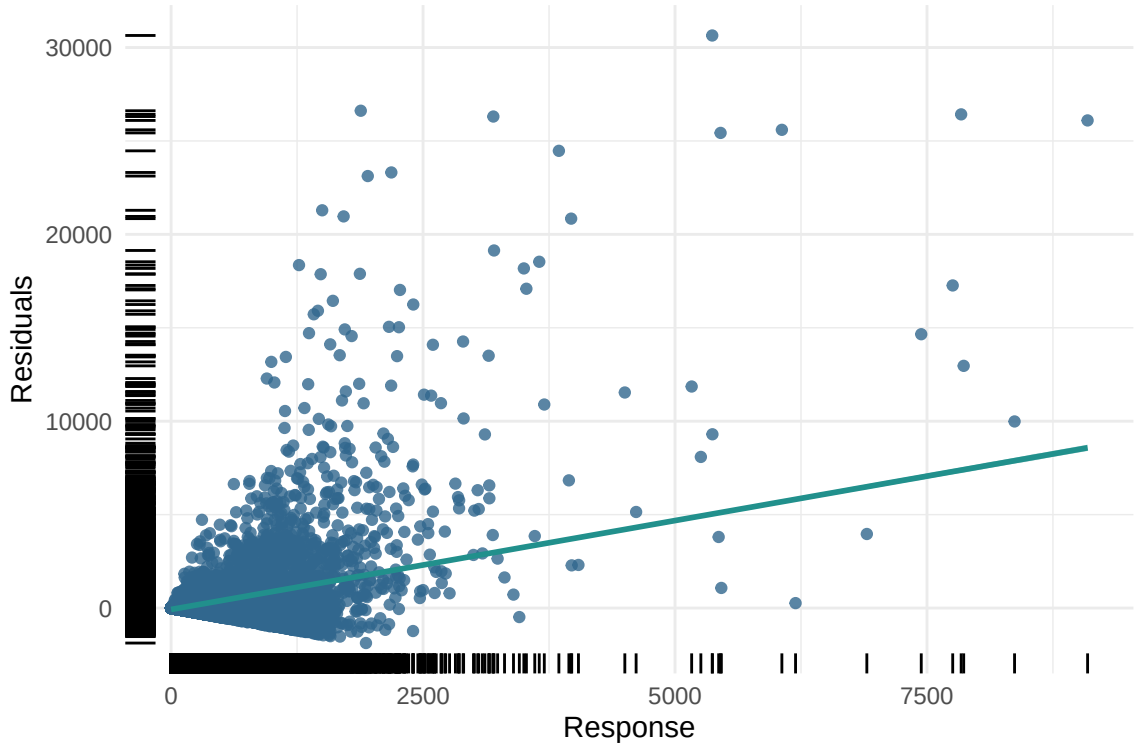

Figure 5: Residual plot, histogram for residuals, QQ-plot

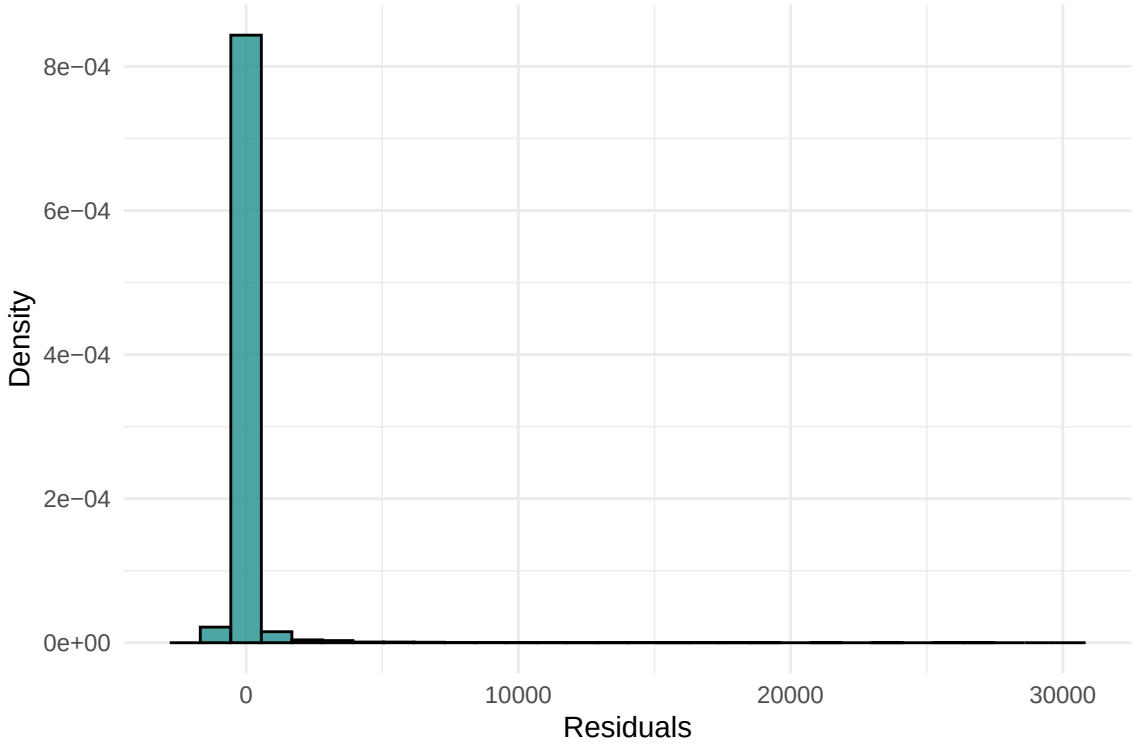

Figure 6: Residual plot, histogram for residuals, QQ-plot