Globant technical test

Jeison Mesa

17 abril, 2021

Introduction

In the workflow for data science professionals, it is important to achieve process automation. Automating processes allows to control human errors made by performing tasks manually. It is proposed to download files using web scraping. In this case, R is connected to the information available for each of the data sets in the web page. It is to recognise the structure of the information. It was possible to identify two types of data structure xls and csv files, for the set of csv files a totally different pattern was found to the xls format files, for this situation only the xls format files were keep. The next step is to have a logical file name. In this case, having files for different months, it is proposed to work with the following format "PCT_year_month" for all the files. This is because the structure presented on the website seemed rather "messy."

The additional step is done to identify the columns that are found for all the files. Resulting in the following variables:

```
colnames(clean_data)
```

```
##  [1] "date"               "TRANS VAT DESC"     "ORIGINAL GROSS AMT"
##  [4] "MERCHANT NAME"      "CARD NUMBER"        "TRANS CAC CODE 1"
##  [7] "TRANS CAC DESC 1"   "TRANS CAC CODE 2"   "TRANS CAC DESC 2"
## [10] "TRANS CAC CODE 3"   "year"               "month"
## [13] "day"                "wday"
```

As can be seen, a date column has been generated in ymd format; in addition to this, year, month and day columns were generated separately in order to have a more universal date format, it is also proposed to calculate the day of the week. In economics, it is usual to find different relationships for weekdays compared to weekends in financial variables.

Quality Control

The inferences that can be by a statistical model are largely influenced by the quality of the information, i.e. we must be able to guarantee that the information was verified and validated; for this purpose the following is generally checked:

1. **Problems associated with the scale of the variable**

Regardless of the type of software that we want to use, it is necessary that at the moment of loading the information those variables that by their typology are strictly numerical, the software can identify it because if we adjust a statistical model with a variable that is numerical and the software detects it as categorical we will reach erroneous conclusions (this is only an example of the many cases that can occur) or vice versa when giving numerical values to categorical variables the software detects them as numerical, in this case it is necessary to perform a transformation.

2. **Missing values**

It is important to be able to check which variables have a high percentage of missing values, since many of the optimisations of statistical models require having the complete data matrix. In this sense, if the information provided presents missing values for some of the variables, a process of imputation of values must be carried out (by some statistical technique ML, pca + NIPALS, KNN, etc...). But it is important to recognize the adjacent stochastic process that arises from credit card transactions. In essence, it is possible to consider a time series with irregular observations over time (Eyheramendy, Elorrieta, and Palma 2016). This is a new approach and can be analyzed by kalman filters and state space representation. In my master thesis I am proposing a regression approach with autocorrelated errors and irregular observations over time, such as credit transactions.

Figure 1: Valores Faltantes

Assuming that traditional methods will be used, the greatest concentration of missing data is found in the "TRANS VAT DESC" variable (Figure 1), and for this reason it will not be used in the statistical inferences to be developed. But it is important to recognize the adjacent stochastic process that arises from credit card transactions. In essence, one can consider a time series with irregular observations over time.

3. **Problems due to the Characteristic of your Variables**.

In many cases we cannot assume that the information is correct, since there may be typing problems or problems due to "outliers" (e.g. imagine a negative precipitation is not possible). When the problem related to the outlier is trivial, i.e. we can know with certainty that it is a value that cannot occur, we can choose to induce a missing value for that data. On the other hand, in order to identify possible outliers it is necessary to use descriptive or inferential statistical techniques; for the descriptive option a box plot can be performed and for the inferential case a probability distribution can be fitted and those values which are not under the distribution curve can be determined.

The Figure 2 shows possible outliers for some transactions. In this case we have a record that may be due to a return of around £5000000 and a value per purchase of more than £250000. These are points that we will later evaluate if they can truly be considered as outliers or fraud issues.

Why is it necessary to develop a deeper analysis? It is necessary because we cannot consider outliers under the assumption of the interquartile ranges of the boxplot. Other types of outlier distributions such as the Gumbel distribution or non-parametric methods can be considered.

The figure 3 shows the dynamics associated with the transactions carried out. In particular, some rather "strange" points can be observed according to the records. The red areas indicate two cases: no transactions were made or no information was found for those periods.

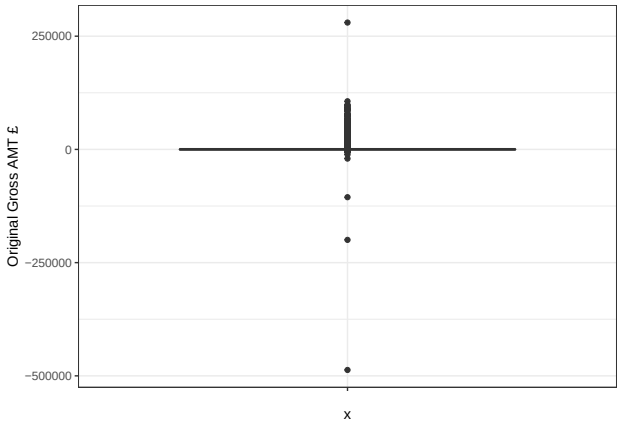

Figure 2: Gross Outlier Detection

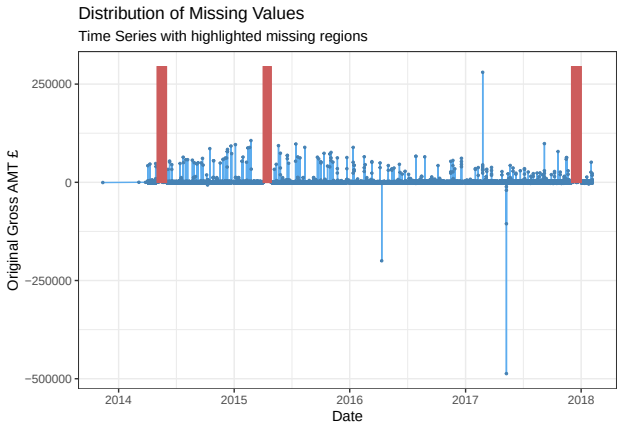

Figure 3: Time Serie Gross AMT

Now it is necessary to perform descriptive statistics that provide valuable information when generating a statistical model.

The most relevant results are shown below.

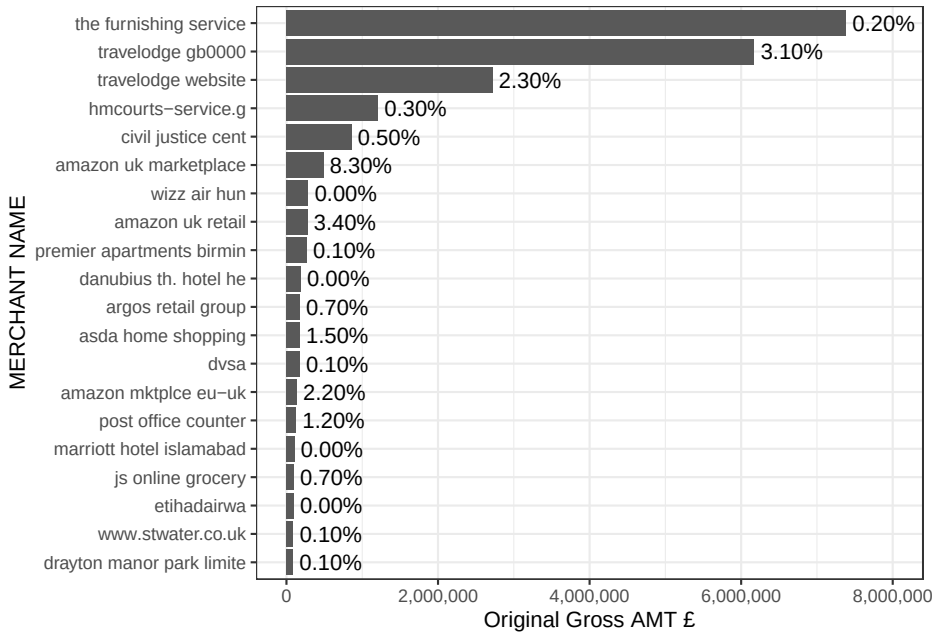

Figure 4: Activity in credit card transaction (percentages are the proportion of the number of times the card was used for the particular merchant, the bars indicate the amount of money used)

The Figure 4 shows that the client for which the most money was spent was "the furnishing service" but not as many transactions were recorded as in the case of "travelodge gb0000." It is also possible to look at several transactions for amazon but different divisions within amazon, now we can aggregate for the amazon category (additionally if we have a team it is possible to collect other variables).

The Figure 5 shows that amazon moved up in the ranking, but the associated spend per transaction is still higher for the "the furnishing service" category. This analysis can be extended to other variables.

Table 1: Sector top 10

TRANS CAC DESC 1	n	prop
Equip Operational	33699	18.59%
Vehicle Fuel	22227	12.26%
Purchases Food	20017	11.05%
Books	11142	6.15%
Supplies & Sev Mic	10499	5.79%
Other Third Parties	9330	5.15%
Mat'l Raw/Drct	7600	4.19%
Conference Fees Subs UK	6444	3.56%
Equip Other	5448	3.01%
Hospitality	4826	2.66%

The Table 1 shows that the sector with the highest transaction is "Equip Operational." In addition, another important use of the card is for "Vehicle Fuel" and "Purchases Food," which represent a person's daily expenses.

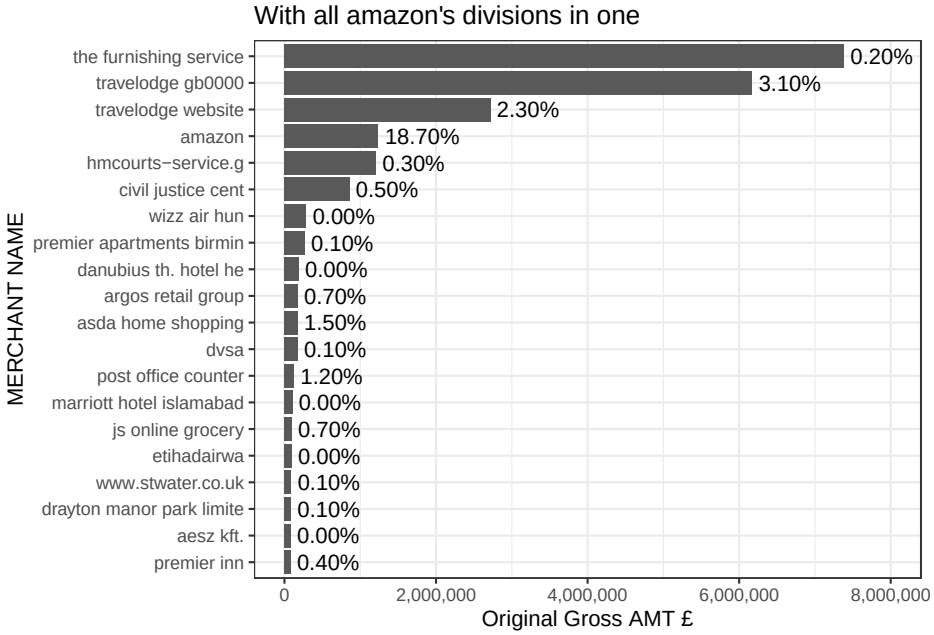

Figure 5: Activity in credit card transaction (percentages are the proportion of the number of times the card was used for the particular merchant, the bars indicate the amount of money used) (Amazon)

Other important descriptive statistics are included

Table 2: TRANS VAT DESC top 10

TRANS VAT DESC	n	prop
VR	86222	50.24%
VZ	83691	48.77%
VL	754	0.44%
VE	618	0.36%
VS	297	0.17%
VT	29	0.02%
6.65%	5	0%

Table 3: TRANS CAC DESC 2 top 10

TRANS CAC DESC 2	n	prop
Illegal Money Lending T Stds Comm Inv.	13450	7.43%
Homeless Private Sector Accom	8558	4.73%
The City of Birmingham School	4904	2.71%
Camborne House HLDC	3065	1.69%
Technical Unit	2581	1.43%
Ward End Junior & Infant (NC)	1777	0.98%
West Heath Primary	1649	0.91%
Corp Inbound Post	1511	0.83%
Selly Oak Nursery School	1509	0.83%
Warwick Hse HLDC, 938 Warwick Rd B27	1448	0.8%

Table 4: Day of the week for the transaction

wday	n	prop
Wed	36781	20.28%
Tue	36236	19.98%
Thu	34736	19.16%
Mon	32155	17.73%
Fri	29855	16.46%
Sat	6583	3.63%
Sun	4989	2.75%

The Table 4 shows something very important, the transactions are being made on weekdays, since there is little use on weekends. This may indicate the fact that the most of the transactions are for "Equip Operational" as these are being made on weekdays.

Model Analysis

In many data anlaysis tasks, outlier detection plays an important role in modelling, inference, and even data processing because otuliers could adversely lead to model misspecification, biased parameter estimation, and poor predictions. The original outlier detection methods were arbitrary like using boxplot analys (interquartile range). But recognizing the pattern of transactions, outlier analysis can be performed with probability distributions for extreme values such as the gumbel distribution.

```
library(readr)
library(glue)
library(FactoMineR)
library(tidyr)
library(factoextra)
library(sparklyr)
library(dplyr)
library(caret)
library(tidymodels)
library(C50)
library(parsnip)
library(ranger)
library(ggpubr)
library(skimr)
library(extRemes)


clean_data <-  read_csv(file = glue::glue("data/cleaning/sequence_purchase_transactions.csv")) %>%
  mutate(week_type = case_when(wday == "Sat" ~ "weekend",
                               wday == "Sun" ~ "weekend",
                               TRUE ~ "Week")) %>%
  dplyr::select(-wday)
names(clean_data) <- gsub(" ", "_", names(clean_data))

clean_data <- clean_data %>%
  mutate(type = case_when(ORIGINAL_GROSS_AMT >0 ~ "purchase",
                          ORIGINAL_GROSS_AMT <=0 ~ "return"))
```

```r
# Hampel filter

lower_bound <- median(clean_data$ORIGINAL_GROSS_AMT, na.rm = T) - 3 * mad(clean_data$ORIGINAL_GROSS_AMT
                                                                          constant = 1, na.rm = T)

upper_bound <- median(clean_data$ORIGINAL_GROSS_AMT, na.rm = T) + 3 * mad(clean_data$ORIGINAL_GROSS_AMT
                                                                          constant = 1, na.rm = T)


complete_data <- clean_data %>%
  mutate(condition = case_when(ORIGINAL_GROSS_AMT <= lower_bound ~ "outlier",
                               ORIGINAL_GROSS_AMT >= upper_bound ~ "outlier",
                               TRUE ~ "normal")) %>%
  drop_na()

complete_data %>%
  filter(row_number()==1 | row_number()==n()) %>%
  dplyr::select(date)
```

```
## # A tibble: 2 x 1
##   date
##   <date>
## 1 2014-04-29
## 2 2018-01-13
```

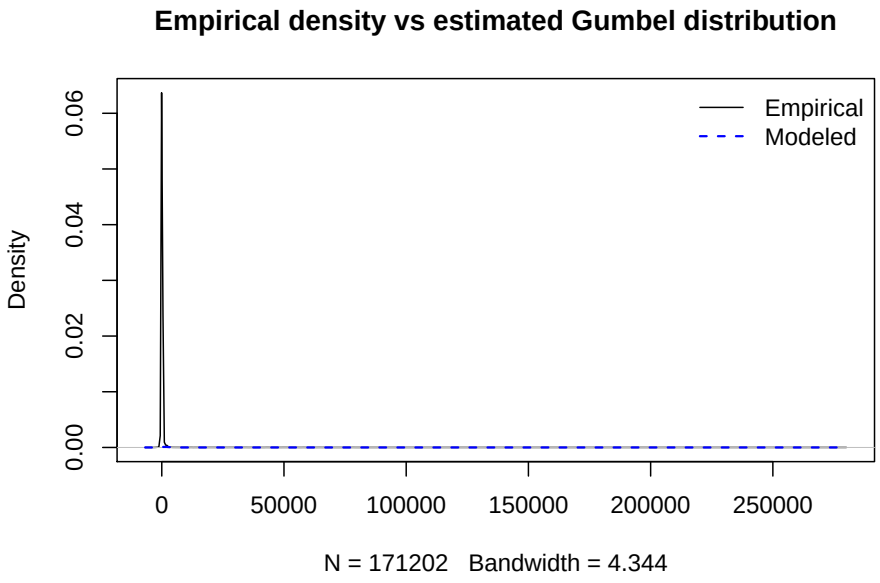

Figure 6: Gumbel

Values are concentrated around low amounts (Figure 6). Therefore, large amounts of money used in transactions can be considered as fraud problems. Additionally, it is possible to use the "Hampel filter" metric to determine the bands in which possible outliers are found (this metric is useful as it is a non-parametric method) and onsists of considering as outliers the values outside the interval (I) formed by the median, plus or minus 3 median absolute deviations.

```
lower_bound
```

```
## [1] -55.01
```

```
upper_bound
```

```
## [1] 136.15
```

Values below -55.01 and values above 136.15 are considered "outliers" which represent information different from the information expected to be found. For now they will be considered as possible outliers, but we will adjust decision trees model to contrast the fitted values of the model with the possible leverage points of the line. In regression analysis, leverage points cause poor modeling. In this sense those leverage points will be considered to be outside the expected value of the distribution.

Description

The term machine learning encompasses the set of algorithms that identify patterns in data and create structures (models) that represent them. Once the models have been generated, they can be used to predict information about facts or events that have not yet been observed. It is important to remember that machine learning systems are only able to memorize patterns that are present in the data they are trained on, so they can only recognize what they have seen before. By using systems trained with past data to predict the future, it is assumed that the behavior will be the same in the future, which is not always the case.

Modelling Stages

1. Preparring the strategy for evaluating the model: separate the observations into a training set, a validation (or cross-validation) set and a test set. It is very important to ensure that no information from the test set is involved in the model training process.
2. Preprocessing the data: apply the necessary transformations so that the data can be interpreted by the selected machine learning algorithm.
3. Adjust a first model capable of overcoming minimum results.
4. Gradually improve the model by incorporating-creating new variables or optimizing the hyperparameters.
5. Evaluating the capacity of the final model with the test set to have an estimate of the capacity of the model when predicting new observations.
6. Train the final model with all available data.

The Tidymodels environment will be used for modeling development. Tidymodels is an interface that unifies under a single framework hundreds of functions from different packages, greatly facilitating all stages of preprocessing, training, optimization and validation of predictive models.

```
set.seed(1234)

split_inicial <- initial_split(
  data   = complete_data,
  prop   = 0.8,
  strata = ORIGINAL_GROSS_AMT
)
datos_train <- training(split_inicial)
datos_test  <- testing(split_inicial)
```

```
transformer <- recipe(
  formula = ORIGINAL_GROSS_AMT ~ TRANS_VAT_DESC+TRANS_CAC_CODE_3+TRANS_CAC_CODE_1+week_type+condition,
  data =  datos_train
) %>%
  step_naomit(all_predictors()) %>%
  step_nzv(all_predictors()) %>%
  step_center(all_numeric(), -all_outcomes()) %>%
  step_scale(all_numeric(), -all_outcomes()) %>%
  step_dummy(all_nominal(), -all_outcomes())


transformer_fit <- prep(transformer)

# The transformations are applied to the training and test set.

datos_train_prep <- bake(transformer_fit, new_data = datos_train)
datos_test_prep  <- bake(transformer_fit, new_data = datos_test)

glimpse(datos_train_prep)
```

```
## Rows: 136,965
## Columns: 147
## $ ORIGINAL_GROSS_AMT     <dbl> 52.32, 67.99, 66.60, 27.76, 80.19, 132.55, 50.1~
## $ TRANS_VAT_DESC_VE      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_VAT_DESC_VL      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_VAT_DESC_VR      <dbl> 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1,~
## $ TRANS_VAT_DESC_VS      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_VAT_DESC_VT      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_VAT_DESC_VZ      <dbl> 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0,~
## $ TRANS_CAC_CODE_1_X5C55 <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H000  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H030  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H0M0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H0R0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H0T0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H220  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H240  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H2Q0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H2T0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_H400  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J000  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J010  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J020  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J030  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J040  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J050  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J0A0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J0C0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J0D0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J0L0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J0V0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J0Z0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J100  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
```

```
## $ TRANS_CAC_CODE_1_J1A0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J1F0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J1G0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J200  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J210  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J220  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J240  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J250  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J260  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J280  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J300  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J310  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J3A0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J400  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J500  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J510  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J550  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J560  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J700  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_J760  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K020  <dbl> 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K030  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K050  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K080  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K110  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K170  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K200  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K210  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K240  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K260  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K300  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K310  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K400  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_K600  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_KZZ0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L000  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L030  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,~
## $ TRANS_CAC_CODE_1_L060  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L080  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L0F0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L0G0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L0J0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L0L0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L0M0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L0P0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L0T0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L100  <dbl> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1,~
## $ TRANS_CAC_CODE_1_L110  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L120  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L130  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L180  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L220  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L250  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L270  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
```

```
## $ TRANS_CAC_CODE_1_L300  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L310  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L410  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L420  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L4H0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L510  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L530  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L540  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L570  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L580  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L5A0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L5B0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L5D0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L5E0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L5F0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L5G0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L5H0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L5J0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L5R0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L670  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L680  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L690  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L6C0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L6E0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L700  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L750  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L800  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L880  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L990  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L991  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_L9Y0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M222  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M310  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M600  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M650  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M660  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M670  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M900  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M960  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M980  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_M990  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_MA20  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_MA50  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_MC70  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_MCJ0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_MCK0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_N050  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_N060  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_N070  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_N0B0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_N110  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_N1F0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_P060  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_P150  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
```

```
## $ TRANS_CAC_CODE_1_P180  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_P190  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_P1D0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_P1G0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_rg10  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ TRANS_CAC_CODE_1_XOA0  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ week_type_weekend      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ condition_outlier      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1,~
```

```
modelo_tree <- decision_tree(mode = "regression") %>%
  set_engine(engine = "rpart")
modelo_tree
```

```
## Decision Tree Model Specification (regression)
##
## Computational engine: rpart
```

```
response_variable <- "ORIGINAL_GROSS_AMT"

predictor_variables <- setdiff(colnames(datos_train_prep), response_variable)

modelo_tree_fit <- modelo_tree %>%
  fit_xy(
    x = datos_train_prep[, predictor_variables],
    y = datos_train_prep[[response_variable]]
  )

modelo_tree_fit$fit
```

```
## n= 136965
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 136965 466767400000  154.28280
##    2) condition_outlier< 0.5 113588    123719300   37.57409 *
##    3) condition_outlier>=0.5 23377 457578800000  721.36640
##      6) TRANS_CAC_CODE_1_L100< 0.5 19294 134218100000  496.03340 *
##      7) TRANS_CAC_CODE_1_L100>=0.5 4083 317751800000 1786.16600
##       14) TRANS_VAT_DESC_VR< 0.5 1584    421317800  246.27950 *
##       15) TRANS_VAT_DESC_VR>=0.5 2499 311193600000 2762.22800 *
```

```
cv_folds <- vfold_cv(
  data    = datos_train,
  v       = 5,
  repeats = 10,
  strata  = ORIGINAL_GROSS_AMT
)
head(cv_folds)
```

```
## # A tibble: 6 x 3
##   splits                id       id2
```

```
##   <list>                 <chr>    <chr>
## 1 <split [109570/27395]> Repeat01 Fold1
## 2 <split [109571/27394]> Repeat01 Fold2
## 3 <split [109573/27392]> Repeat01 Fold3
## 4 <split [109573/27392]> Repeat01 Fold4
## 5 <split [109573/27392]> Repeat01 Fold5
## 6 <split [109570/27395]> Repeat02 Fold1
```

```
modelo_tree <- decision_tree(mode = "regression") %>%
  set_engine(engine = "rpart")

validacion_fit <- fit_resamples(
  object       = modelo_tree,
  preprocessor = transformer,
  resamples    = cv_folds,
  metrics      = metric_set(rmse, mae),
  control      = control_resamples(save_pred = TRUE)
)

head(validacion_fit)
```

```
## # A tibble: 6 x 6
##   splits          id      id2   .metrics        .notes        .predictions
##   <list>          <chr>   <chr> <list>          <list>        <list>
## 1 <split [109570~ Repeat~ Fold1 <tibble[,4] [2~ <tibble[,1] ~ <tibble[,4] [27,3~
## 2 <split [109571~ Repeat~ Fold2 <tibble[,4] [2~ <tibble[,1] ~ <tibble[,4] [27,3~
## 3 <split [109573~ Repeat~ Fold3 <tibble[,4] [2~ <tibble[,1] ~ <tibble[,4] [27,3~
## 4 <split [109573~ Repeat~ Fold4 <tibble[,4] [2~ <tibble[,1] ~ <tibble[,4] [27,3~
## 5 <split [109573~ Repeat~ Fold5 <tibble[,4] [2~ <tibble[,1] ~ <tibble[,4] [27,3~
## 6 <split [109570~ Repeat~ Fold1 <tibble[,4] [2~ <tibble[,1] ~ <tibble[,4] [27,3~
```

```
validacion_fit %>%
  collect_metrics(summarize = TRUE)
```

```
## # A tibble: 2 x 6
##   .metric .estimator  mean     n std_err .config
##   <chr>   <chr>      <dbl> <int>   <dbl> <chr>
## 1 mae     standard    177.    50   0.974 Preprocessor1_Model1
## 2 rmse    standard   1781.    50  43.2   Preprocessor1_Model1
```

```
validacion_fit %>%
  collect_metrics(summarize = FALSE) %>% head()
```

```
## # A tibble: 6 x 6
##   id       id2   .metric .estimator .estimate .config
##   <chr>    <chr> <chr>   <chr>          <dbl> <chr>
## 1 Repeat01 Fold1 rmse    standard       1639. Preprocessor1_Model1
## 2 Repeat01 Fold1 mae     standard        173. Preprocessor1_Model1
## 3 Repeat01 Fold2 rmse    standard       1769. Preprocessor1_Model1
## 4 Repeat01 Fold2 mae     standard        183. Preprocessor1_Model1
## 5 Repeat01 Fold3 rmse    standard       1738. Preprocessor1_Model1
## 6 Repeat01 Fold3 mae     standard        186. Preprocessor1_Model1
```

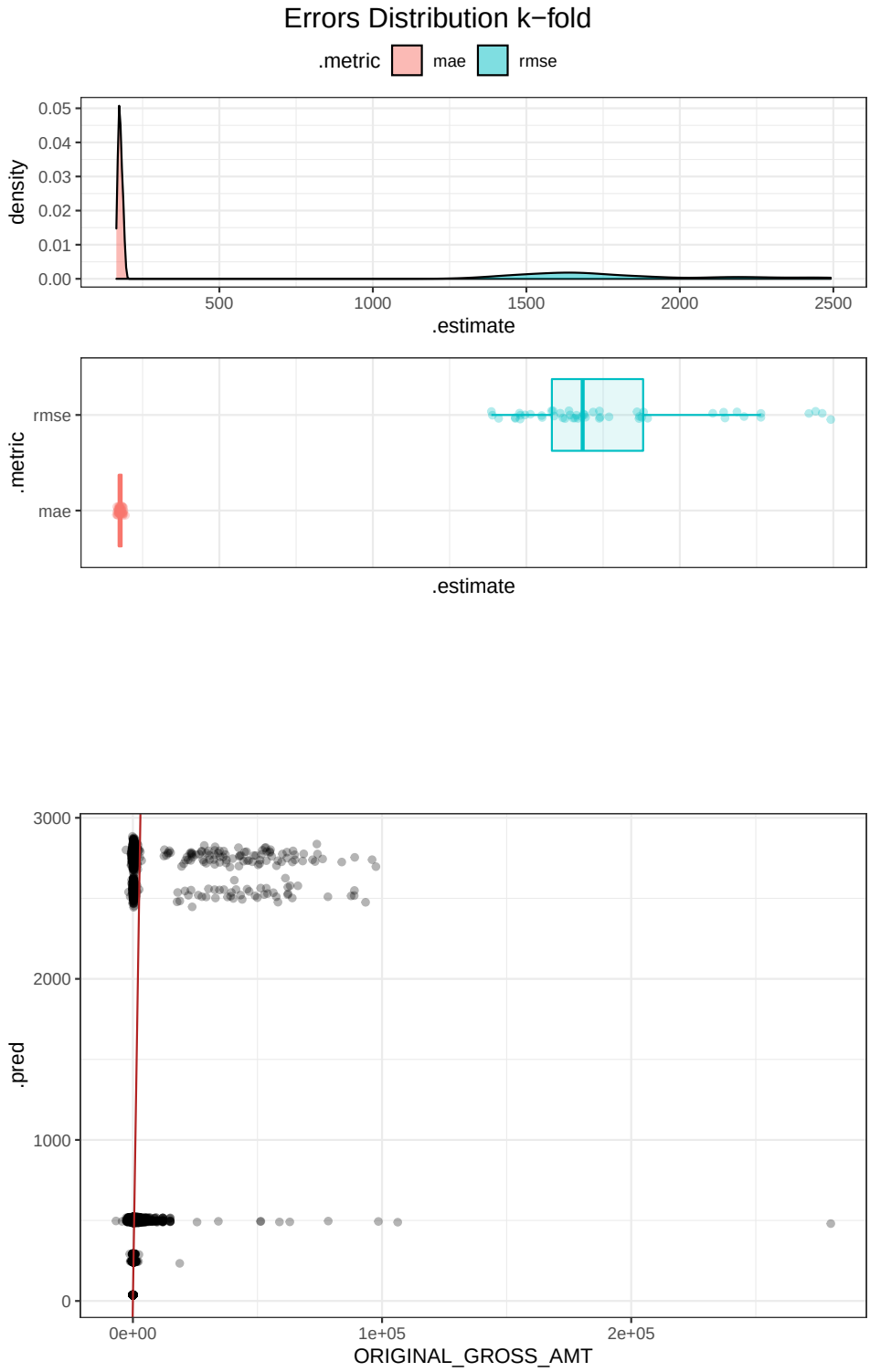

Figure 7: fitted vs Response

It is possible to perform an analysis of extreme events as shown in the Figure 7, given that there are two large concentrations of the amount of money spent per transaction, amounts below 1000 and amounts around 3000. This implies that transactions above 3000 are unusual considering the whole history of transactions.

References

Eyheramendy, Susana, Felipe Elorrieta, and Wilfredo Palma. 2016. "An Autoregressive Model for Irregular Time Series of Variable Stars." *Proceedings of the International Astronomical Union* 12 (S325): 259–62. https://doi.org/10.1017/S1743921317000448.