ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

2020

"Big Data Content Analysis"
Personal Accident

KAILANI ANASTASIA
KOURTESI IOANNA

Table of Contents

Table of Figures

1.Descriptions

Nowadays, we are living in the data era, which means that a company can collect data from a variety of sources in order to enhance the customer experience. That is a huge advantage in terms of customer service. Collecting and analyzing customer's feedback and comments from social media about companies, services and products, provides them the advantage to have better information in order to make strategic decisions, while having an accurate understanding of what the customer actually wants and, as a result, a better experience for everyone.

We are a business intelligence services provider that has been appointed by an Insurance Company in order to build a model that will classify whether or not we will suggest an insurance contract to a potential customer. The aim of our project is that, the company will increase its profit by taking the right decision.

This document outlines the steps involved with fitting models for Personal Accident coverage. The child accident has been excluded. Models will return either a yes or no if it will be offered.

The covers that will be modeled are:

1) Chewing Damage (Tyggeskade)
2) Disease (Sygdom)
3) Critical Illness (Kritisk Sygdom)
4) Extended Help (Udvidet hjælp)

The following sections outline the parameters of the problem.

2. Mission

The potential customers that are shopping for personal accident coverage traditionally are offered all of the available covers. To reduce the possibility of offering a cover that the customer does not want and degrading their user experience, predictive models are to be used to determine which customers are the most likely to purchase the extra covers. By offering each customer only the products that they are most likely to purchase, the overall customer experience can be improved.

A model will be fit for each of the six different products which will determine whether or not a product will be suggested for purchase.

3. Data

Initially, it is important for the handling of this project to analyze and give more information about the datasets we used. The data have been given to us, as an outsourced employee, from an Insurance's employee, who works at the company's Pre-Sales Department. This company is working according to GDPR. That means that it has a limitation about the independent variables that we can use at our research.

Before the company gave us the datasets, they had done a previous cleansing of them according to their needs. This does not mean that we did not clean further the data. In order to get the data, the company gave us two CSV files, we cleaned them and then we merged them. The names of the csv files are *accidents_data.csv* and *occupation_tariff_codes.csv.* The final number of observations that had been used for the results is 48.620. At all the models we used a train and test dataset, which we split according to 80%-20% rule.

At this point, we should note that after a lot of communications with the contact person at the company, we came up with the final scope of this project. This scope is not to choose the most accurate model, but the one that leads to increasing sales (sales suggestions). Finally, the company had some business rules that we had to follow. These business rules are about the cleansing that we have done and we mention them below.

Since the customers that will be offered the additional coverage consists of shoppers that are not currently customers, the data available for them is limited. The data items included consist of more than is available when the decision has to be made. The data items available for prediction are:

Variable	Type	
Geographic Region	Nominal (Generated from Zip (Postal) Code)	zipcode_factor
Tariff Class	Nominal (Generated from Occupation Code)	Tariff_aggregate_code
Housing Type	Nominal	BOLITYPE_aggregate_code
Socioeconomic Code	Nominal	TYPKODE
Age	Continues	ALDER
Motorcycle	Binary	
Occupation Code	Nominal	BESKTITY

“Motorcycle” is empty and so it cannot be used. It has been removed.

The data cover approximately two months of coverage purchases but it was filtered to make sure each customer was only present once in the data.

The steps for data cleansing were as follows:

1. Indicator Removal – Indicators for other covers were removed from the data (independence assumption)
2. Very low instances of certain variables were removed.
3. Not applicable values of certain features were removed (like age less than 18, etc.)
4. Data imputation in Socio-economic variable
5. Separation of full dataset into subsets to satisfy the requirements of each cover.

Business rules

The business rules about product’s sales (if a product is allowed to be sold) regarding the age are:

1)	Chewing Damage	<70
2)	Disease	<=70
3)	Critical Illness	<=24
4)	Extended	any age

According to the above matrix the data must be divided into the following distinct age groupings (at least for the models that have restrictions):

1) Zero to 17 (must be removed from all models and therefore from the dataset itself)
2) 18 to 24
3) 25 to 69
4) 70 and up

The hit rate for Disease, Critical Illness and Extended Help is almost zero.

Model Assumptions

No model will be developed for Disease, Critical Illness and Extended Help covers. Instead, there will be always a ‘No’ on recommendation from a hypothetical model.

For Chewing, the current hit rate is 34%. The model will suggest “Yes” (recommend this) between 35% of times.

Data Features

The original distribution of the Covers can be seen in the next graphs.

Chewing Damage

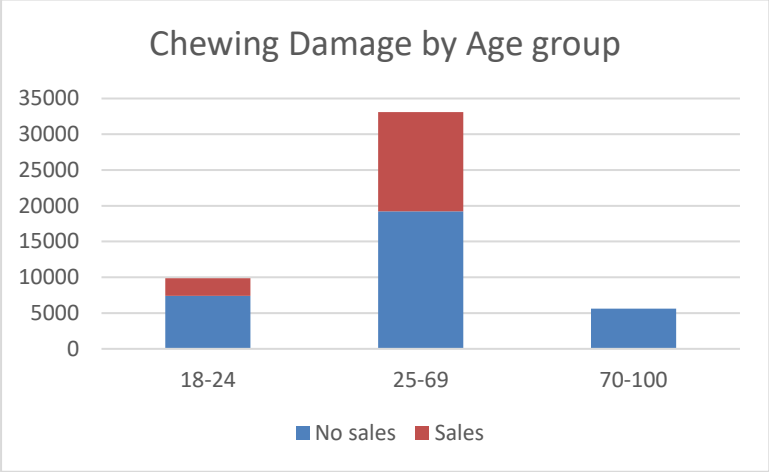

Figure 1 Chewing Damage by Age group and sales vs no sales

Sales vs No sales

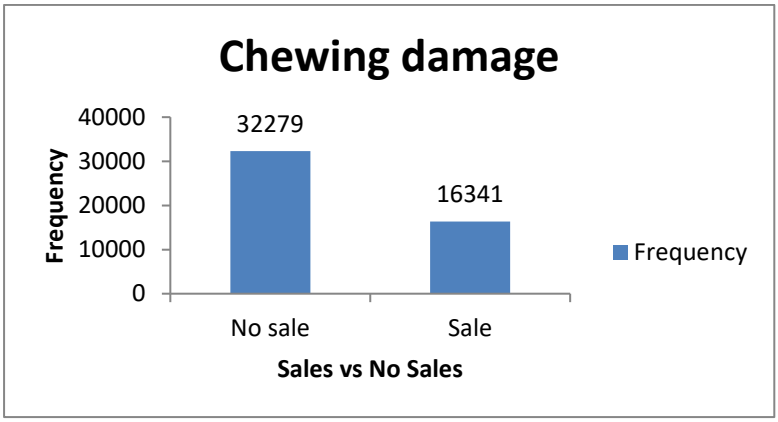

Figure 2 chewing damage and sales vs no sales

Disease

Figure 3 Disease by age group and sales vs no sales

Sales vs No sales

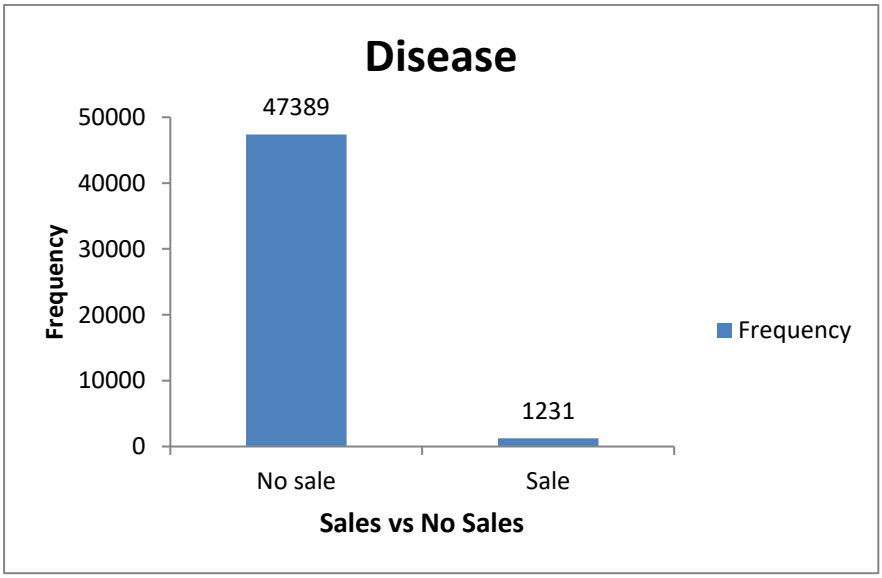

Figure 4 Disease and sales vs no sales

Critical Illness

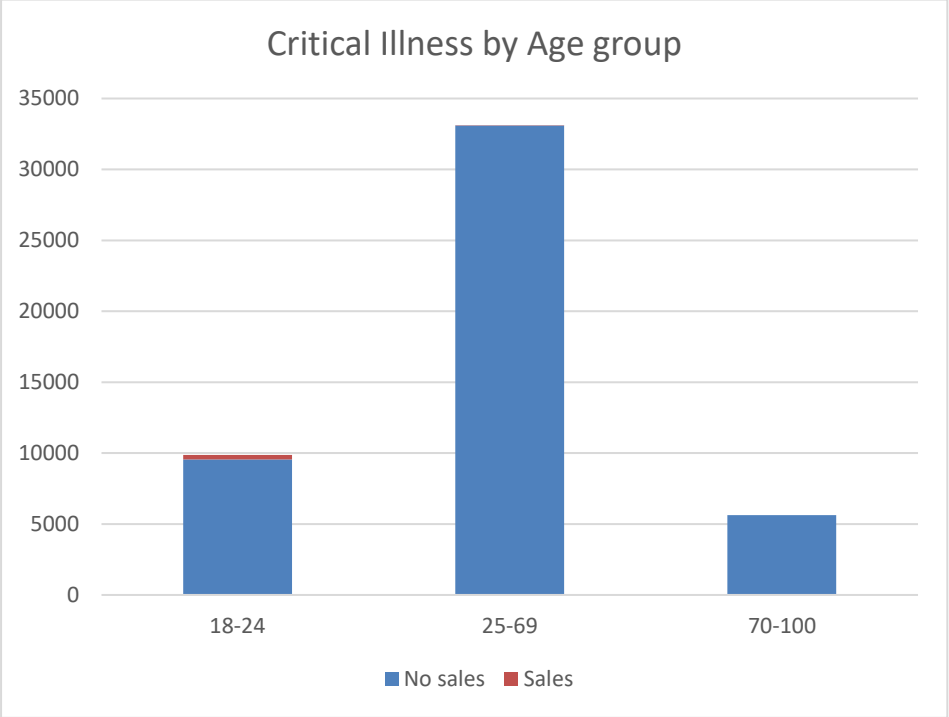

Figure 5 Critical Illness by Age group and sales vs no sales

Sales vs No sales

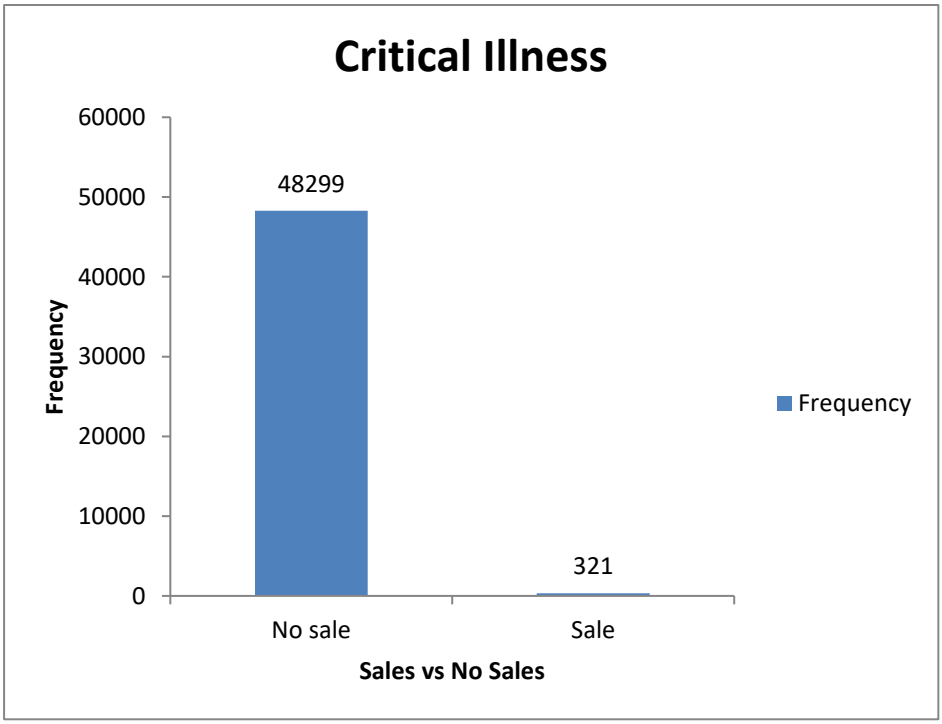

Figure 6 Critical Illness and sales vs no sales

Extended Help

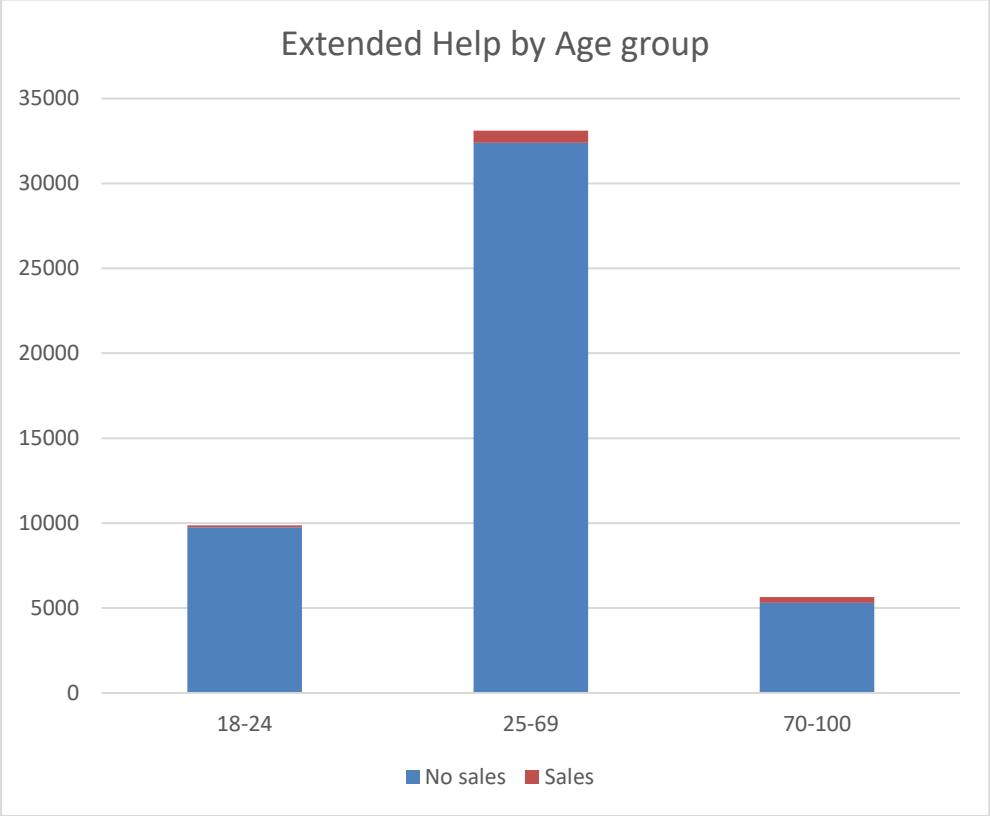

Figure 7 Extended help and sales vs no sales

Sales vs No sales

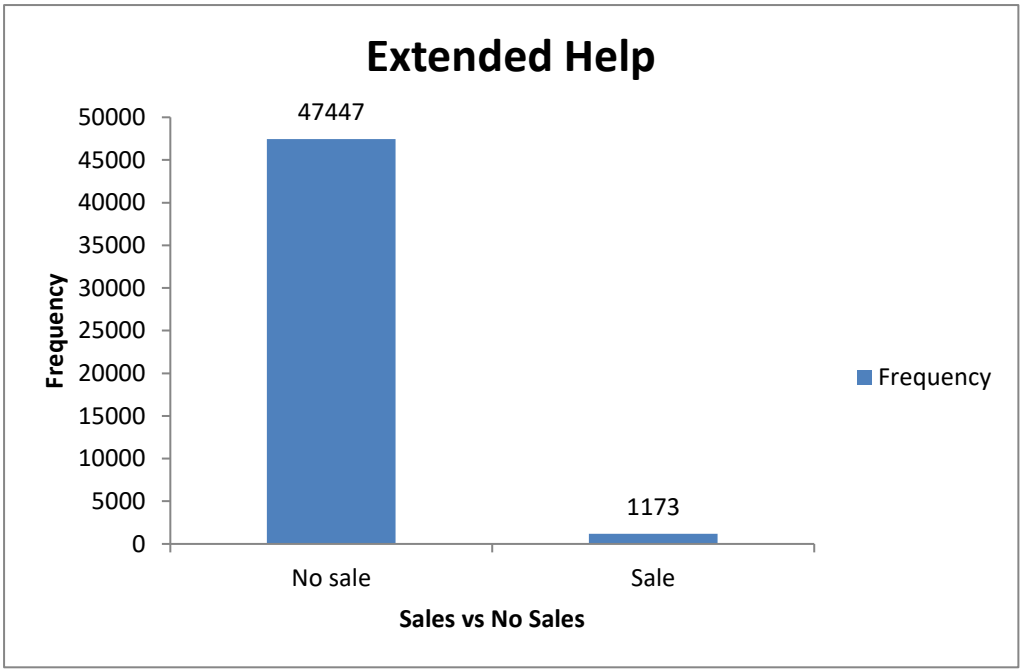

Figure 8 Extended help and sales vs no sales

Observations

As we can see the hit rate from Disease, Critical Illness and Extended Help is so low that is impossible to make predictions.

So, in their place there will be a default value which will be always '*No*'.

For Chewing Damage, the dataset that has been used is a filtered one that contains only ages below or equal with 69 years old.

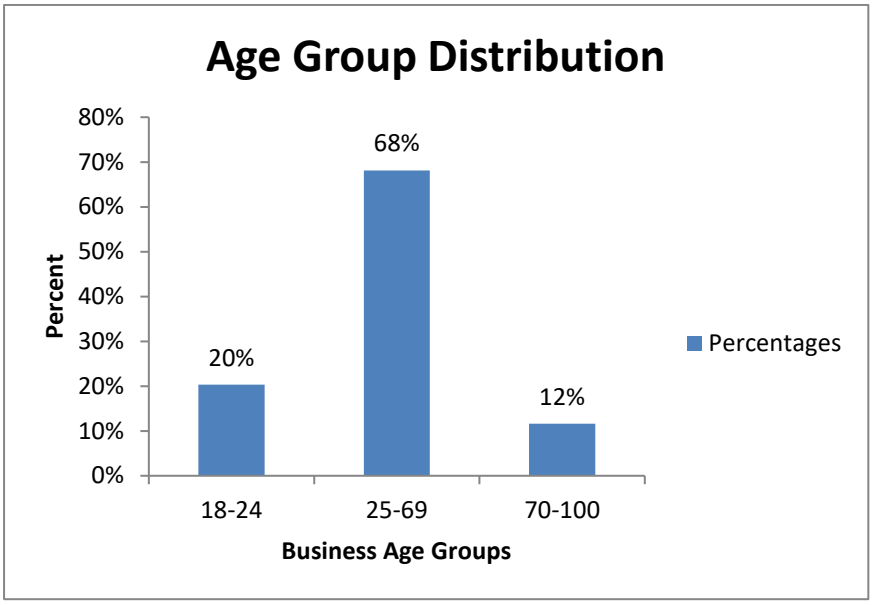

Figure 9 Age group Distribution

The Tariff Codes distribution is as follows

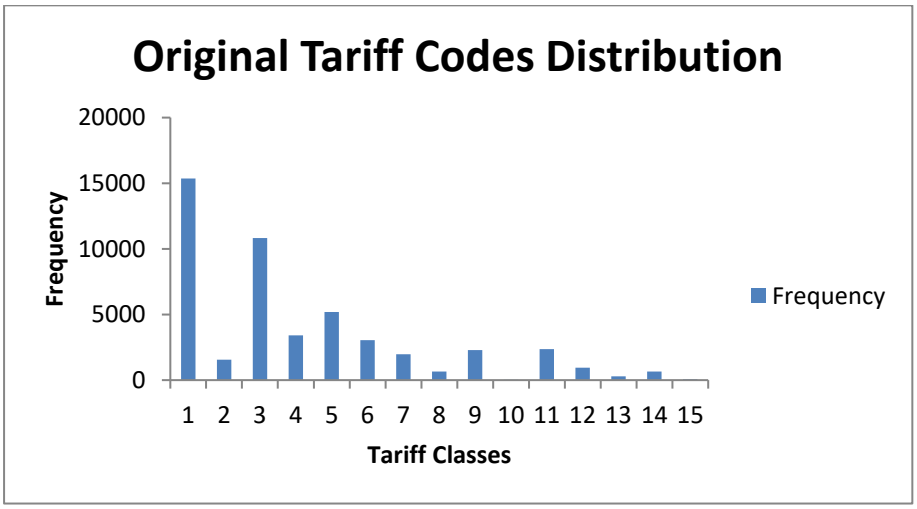

Figure 10 Original Tarrif Codes Distribution

The original distribution of Housing Type follows

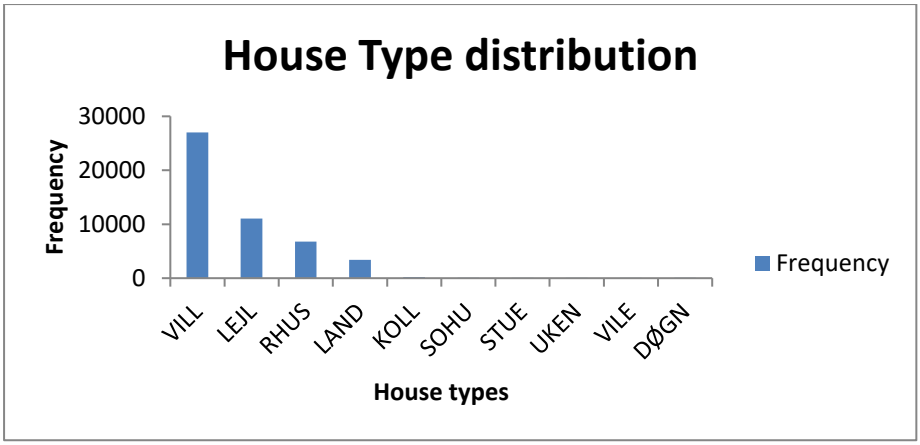

Figure 11 House Type distribution

The original distribution of Socio-Economic Types follows

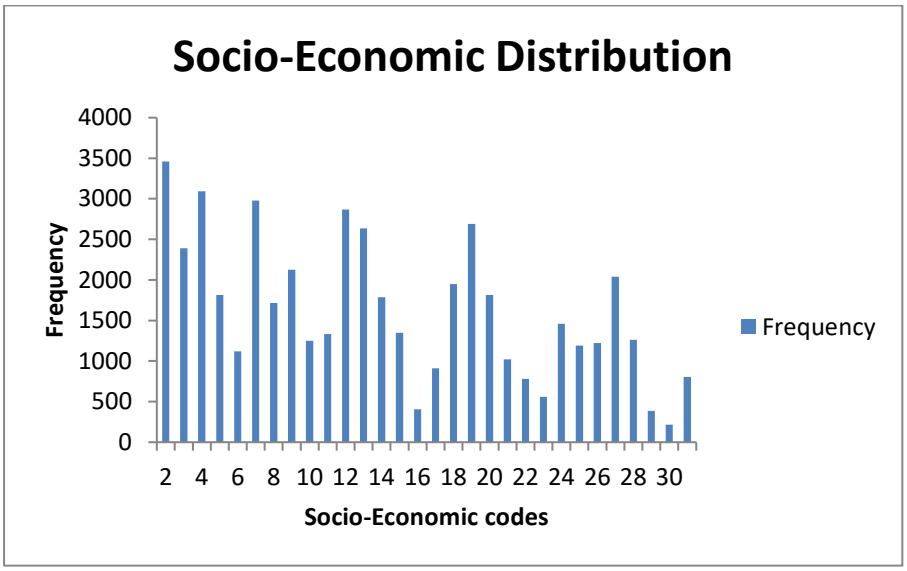

Figure 12 Socio - Economic distribution

Feature Engineering

After cleansing there had to be done some feature engineering.

A new variable was created with values of Tariff Codes aggregated as below (according to the graph above):

- Values with Tariff Class 2-7 remained as were
- Values with Tariff Class greater or equal to 8 became 8 and took a general description like “All others”

The new distribution follows

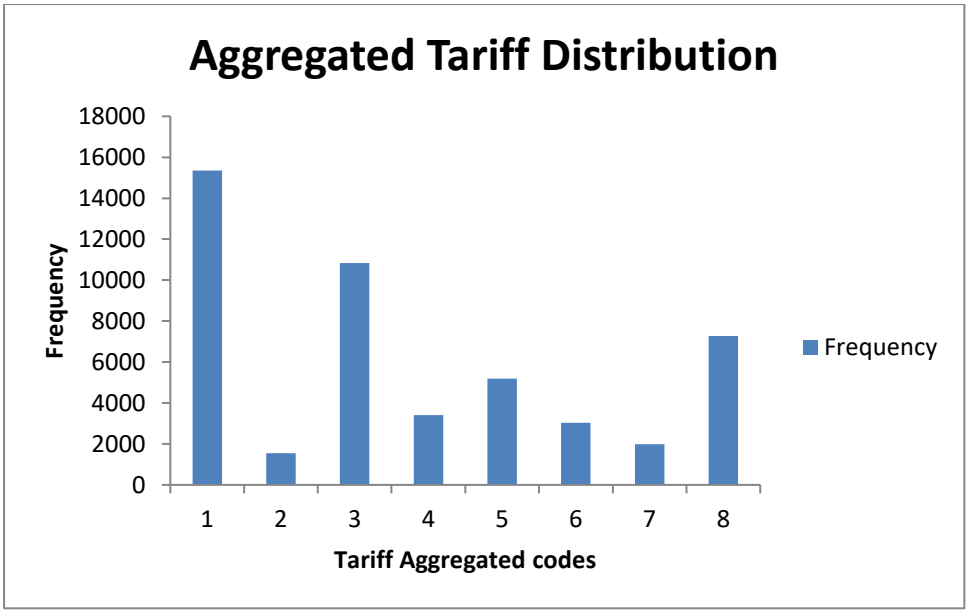

Figure 13 Aggregated Tariff distribution

Due to the very low rations of House codes from "KOLL" and above, a new variable was created with values of House Type aggregated as below (according to the graph in previous paragraph):

- Values with House Type ("VILL", "LEJL", "RHUS", "LAND") remained as were
- Values with the rest House Types took a general description like "All others"

The new distribution is as below

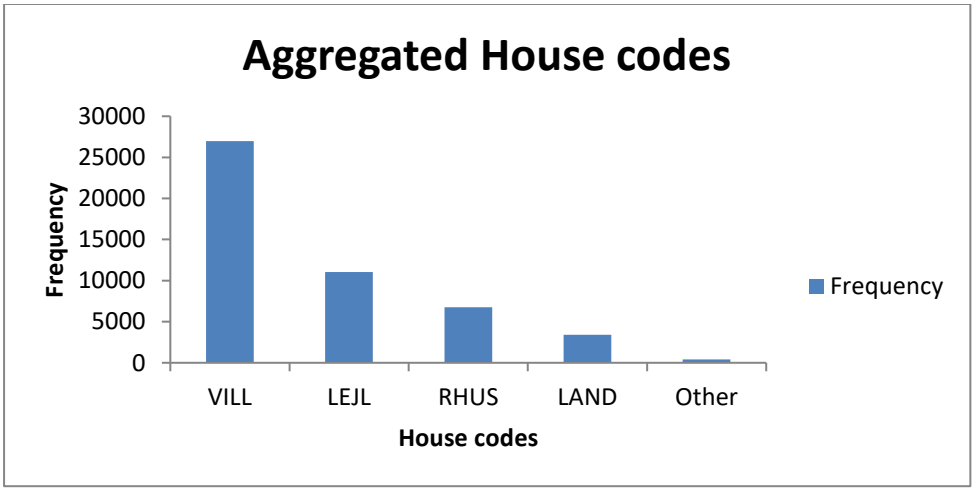

Figure 14 Aggregated House codes

A new variable has been created transforming Age into Age groups according to pattern:

- Ages 18 – 24
- Ages 25 – 69
- Ages 70 – 100 (100 is the maximum value)

Finally, a new scaled dataset was created for testing purposes according to minmax scale. This is equal:

$$Variable_{scaled} = \frac{Variable_{unscaled} - Variable_{Minimum}}{Variable_{Maximum} - Variable_{Minimum}}$$

All models have been tested on this dataset as well.

4. Methodology and Results

Analysis and Model Development

Below there is the problem analysis the model development. Before finalizing the models, there has been an extensive research of the problem with many other algorithms. We have tested:

- Logistic Regression
- Decision Trees
- Random Forest
- Neural Networks

Form this research the best models are logistic Regression for Chewing Damage.

All the results have been taken from the test datasets. But not only from one test. Cross-Validation has been used, so the results like ROC, AUC, etc. are the average from the all the test datasets.

In the Cross-Validation procedure the percentages are 80% for Training and 20% for Test.

Chewing Damage

Here there is a recording of model tests. The method that was used in order to asses if each method is correct for the problem is Cross-Validation in Cover4 data set with 5 repetitions. Chewing model was developed with the use of scaled data.

Logistic Regression

Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

Results from Logistic Regression application.

The data elements that were significant for Chewing Damage were:

- Insurer's Age
- Zip code
- Aggregated Housing Type
- Aggregated Tariff Code
- Socio-Economic factor
- Occupation code

The general qualitative characteristics are as below:

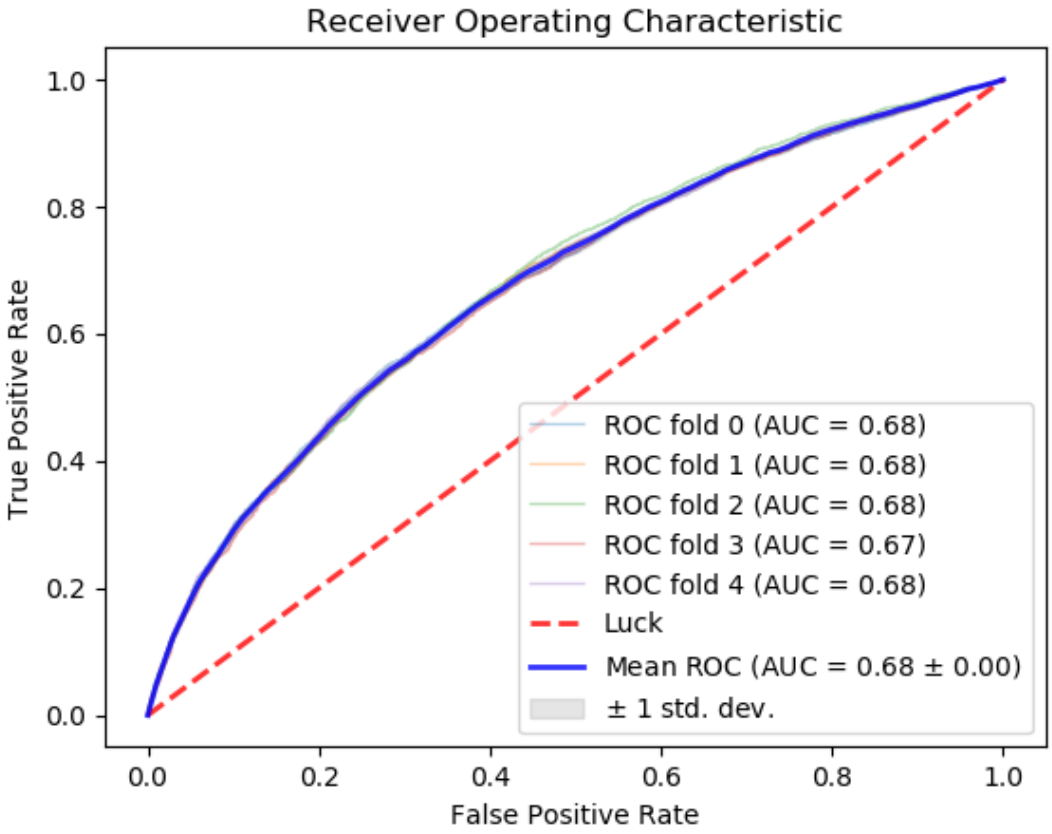

Figure 15 Roc curve

The ROC and the AUC are good as we can see.

From Confusion Matrices with several thresholds we get the results.

F1_score	MCC	Specificity	Sensitivity	Threshold
0.582	0.212	0.390	0.812	0.3
0.563	0.268	0.678	0.678	0.4
0.440	0.244	0.862	0.345	0.5

If we continue to increase the threshold, we will end up with almost zero recommendations.

The threshold suitable for us is 0.3

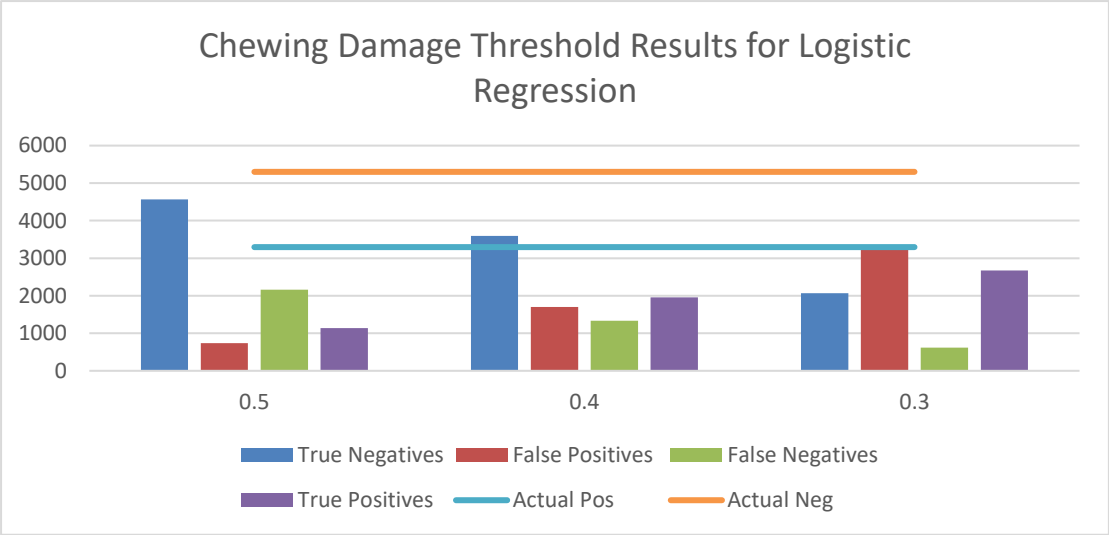

Figure 16 Chewing Damage Threshold for logistic Regression

The purchase probability distribution shows (below table) that with a threshold of 0.3 will result in about 85% of the people being given the offer. Figure is a graph of the table values which shows that the actual hit rate follows the predicted values across the board.

	Actual Hit rate	Avg. Predicted Sales	Max	Min
1	73%	70%	82%	64%
2	65%	61%	64%	58%
3	57%	56%	58%	54%
4	52%	52%	54%	51%
5	42%	49%	51%	48%
6	47%	46%	48%	45%
7	41%	44%	45%	43%
8	40%	41%	43%	40%
9	40%	39%	40%	38%
10	31%	37%	38%	36%
11	33%	35%	36%	35%
12	32%	34%	35%	33%
13	31%	32%	33%	31%
14	35%	30%	31%	29%
15	32%	27%	29%	26%
16	25%	25%	26%	24%
17	25%	23%	24%	21%
18	21%	20%	21%	19%
19	19%	18%	19%	16%
20	21%	12%	16%	7%

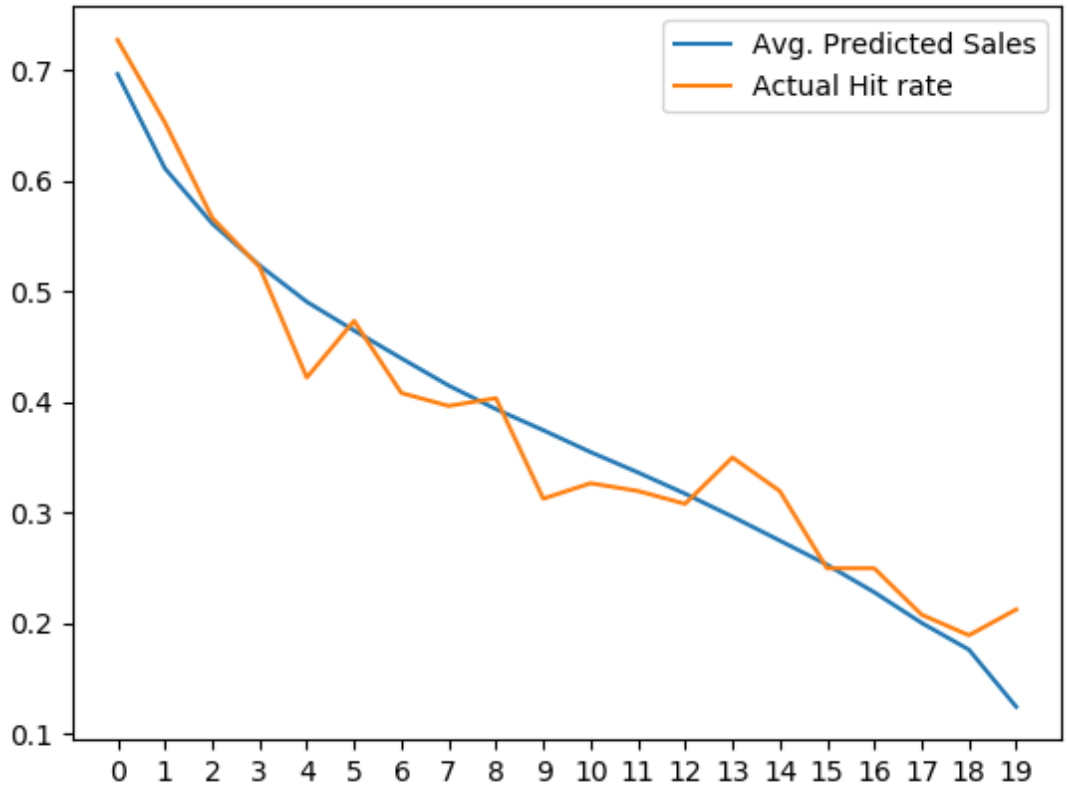

Figure 17 Average Predicted Sales vs Actual Hit rate

Neural Network

Multilayer Perceptrons (MLP)

An MLP consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLPs are suitable for classification prediction problems where inputs are assigned a class or label. They are also suitable for regression prediction problems where a real-valued quantity is predicted given a set of inputs.

Results from Multilayer Perceptrons (MLP).

The data elements that were significant for Chewing Damage were:

- Insurer's Age
- Zip code
- Aggregated Housing Type

- Aggregated Tariff Code
- Socio-Economic factor
- Occupation code

From Confusion Matrices with several thresholds we get the results.

F1_score	MCC	Specificity	Sensitivity	Threshold
0.562	0.207	0.497	0.714	0.3
0.526	0.218	0.665	0.558	0.4
0.478	0.231	0.793	0.423	0.5

The threshold suitable for us is 0.3.

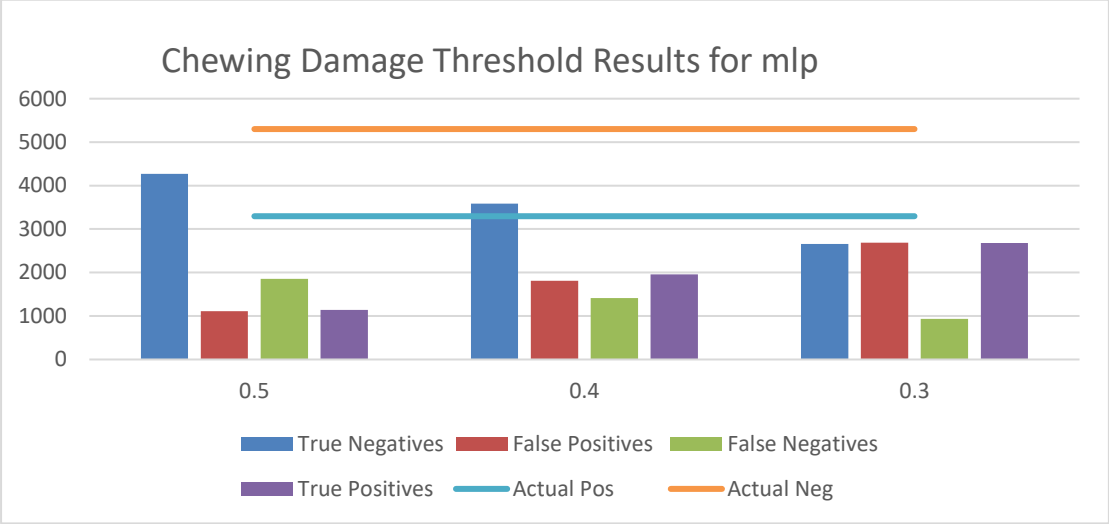

Figure 18 Chewing Damage Threshold Results for mlp

The time required for our model to train was 205 sec on an Intel Core i5 8250U machine with 8 GBytes of RAM.

Recurrent Neural Networks (RNN)

Recurrent Neural Networks, or RNNs, were designed to work with sequence prediction problems. Sequence prediction problems come in many forms and are best described by the types of inputs and outputs supported. Some examples of sequence prediction problems include:

- One-to-Many: An observation as input mapped to a sequence with multiple steps as an output.
- Many-to-One: A sequence of multiple steps as input mapped to class or quantity prediction.
- Many-to-Many: A sequence of multiple steps as input mapped to a sequence with multiple steps as output.
- The Many-to-Many problem is often referred to as sequence-to-sequence, or seq2seq for short.

The Long Short-Term Memory, or LSTM, network is perhaps the most successful RNN because it overcomes the problems of training a recurrent network and in turn has been used on a wide range of applications.

Results from Recurrent Neural Networks (RNN).

The data elements that were significant for Chewing Damage were:

- Insurer's Age
- Zip code
- Aggregated Housing Type
- Aggregated Tariff Code
- Socio-Economic factor
- Occupation code

On training the above Rnn model, we get this output:

We can see below the learning curves of our model. We used 20 epochs. The model could probably use a few more epochs of training and may achieve a higher skill.

```
Train on 27508 samples, validate on 6878 samples
Epoch 1/20
27508/27508 [==============================] - 1s 30us/step - loss: 0.6727 - acc: 0.6220 - val_loss: 0.6602 - val_acc: 0.6149
Epoch 2/20
27508/27508 [==============================] - 0s 13us/step - loss: 0.6499 - acc: 0.6228 - val_loss: 0.6457 - val_acc: 0.6157
Epoch 3/20
27508/27508 [==============================] - 0s 14us/step - loss: 0.6377 - acc: 0.6390 - val_loss: 0.6351 - val_acc: 0.6445
Epoch 4/20
27508/27508 [==============================] - 0s 15us/step - loss: 0.6292 - acc: 0.6546 - val_loss: 0.6293 - val_acc: 0.6499
Epoch 5/20
27508/27508 [==============================] - 1s 18us/step - loss: 0.6248 - acc: 0.6604 - val_loss: 0.6266 - val_acc: 0.6551
Epoch 6/20
27508/27508 [==============================] - 0s 18us/step - loss: 0.6235 - acc: 0.6612 - val_loss: 0.6244 - val_acc: 0.6608
Epoch 7/20
27508/27508 [==============================] - 1s 18us/step - loss: 0.6209 - acc: 0.6638 - val_loss: 0.6228 - val_acc: 0.6627
Epoch 8/20
27508/27508 [==============================] - 1s 18us/step - loss: 0.6206 - acc: 0.6641 - val_loss: 0.6216 - val_acc: 0.6652
Epoch 9/20
27508/27508 [==============================] - 1s 18us/step - loss: 0.6182 - acc: 0.6665 - val_loss: 0.6211 - val_acc: 0.6631
Epoch 10/20
27508/27508 [==============================] - 1s 18us/step - loss: 0.6173 - acc: 0.6677 - val_loss: 0.6198 - val_acc: 0.6675

Epoch 11/20
27508/27508 [==============================] - 1s 22us/step - loss: 0.6155 - acc: 0.6685 - val_loss: 0.6173 - val_acc: 0.6698
Epoch 12/20
27508/27508 [==============================] - 1s 22us/step - loss: 0.6149 - acc: 0.6696 - val_loss: 0.6166 - val_acc: 0.6701
Epoch 13/20
27508/27508 [==============================] - 1s 22us/step - loss: 0.6139 - acc: 0.6693 - val_loss: 0.6163 - val_acc: 0.6705
Epoch 14/20
27508/27508 [==============================] - 1s 22us/step - loss: 0.6138 - acc: 0.6696 - val_loss: 0.6160 - val_acc: 0.6714
Epoch 15/20
27508/27508 [==============================] - 1s 22us/step - loss: 0.6134 - acc: 0.6702 - val_loss: 0.6157 - val_acc: 0.6704
Epoch 16/20
27508/27508 [==============================] - 1s 23us/step - loss: 0.6122 - acc: 0.6723 - val_loss: 0.6153 - val_acc: 0.6721
Epoch 17/20
27508/27508 [==============================] - 1s 26us/step - loss: 0.6116 - acc: 0.6735 - val_loss: 0.6150 - val_acc: 0.6719
Epoch 18/20
27508/27508 [==============================] - 1s 26us/step - loss: 0.6122 - acc: 0.6717 - val_loss: 0.6149 - val_acc: 0.6730
Epoch 19/20
27508/27508 [==============================] - 1s 25us/step - loss: 0.6102 - acc: 0.6738 - val_loss: 0.6145 - val_acc: 0.6735
Epoch 20/20
27508/27508 [==============================] - 1s 24us/step - loss: 0.6101 - acc: 0.6725 - val_loss: 0.6143 - val_acc: 0.6733

Accuracy  0.673791659396266
```

Figure 19 validation accuracy and loss

The maximum value we get on the validation set is 67.35%.

A plot of accuracy on the training and validation datasets over training epochs.

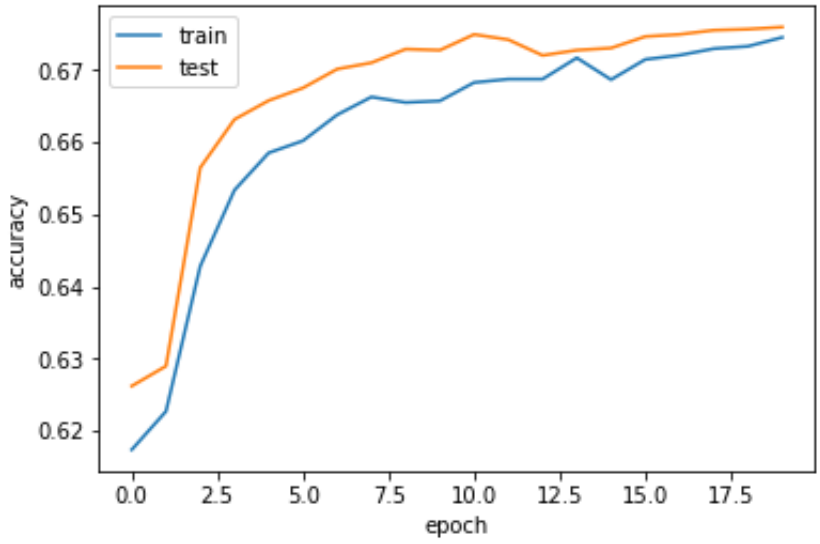

Figure 20 Accuracy on training and validation datasets over epochs

From the plot of accuracy, we can see that the model could probably be trained a little more as the trend for accuracy on both datasets is still rising for the last few epochs. We can also see that the model has not yet over-learned the training dataset, showing comparable skill on both datasets.

A plot of loss on the training and validation datasets over training epochs.

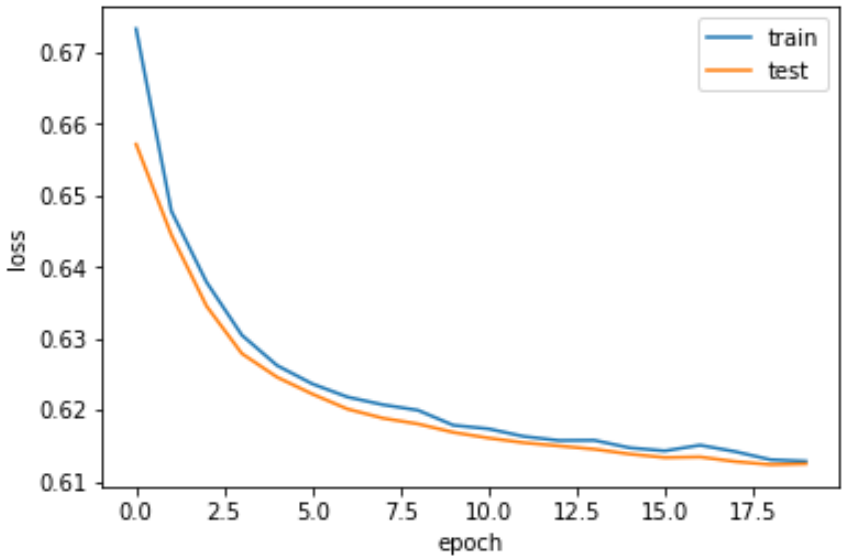

Figure 21 loss on the training and validation datasets over epochs

From the plot of loss, we can see that the model has comparable performance on both train and validation datasets. From this plot we can see if there is overfitting. In this case seems to be a good fit.

The time required for our model to train was 25 sec on an Intel Core i5 8250U machine with 8 GBytes of RAM.

Summary

The model and the main parameters that we have used in order to produce reliable results are as following:

- Chewing Damage
    - Binary Logistic Regression
    - Threshold = 0.3
    - Multilayer Perceptions
    - Recurrent Neural Networks

We conclude that the results of logistic regression are approximately the same as in Rnn. So, we choose the simplest model (this mindset exists in all companies) which is Logistic Regression.

5.Members/Roles

Our group consists of 2 members: Kailani Anastasia and Kourtesi Ioanna. We tried to divide the needs for this project into two major categories.

1) Developers and Software engineering

   Both members had to deal with this part. Most specifically:

   Kailani Anastasia: Graduate student of Statistics at Athens University Economics and Business. Currently working as Business Intelligence and Data Analyst Engineer at Bewise.

   Kourtesi Ioanna: Graduate student of Department of Economics University of Patras. Currently working as Business Intelligence Engineer at Accepted Ltd.

2) Statistical Analysis and Models Implementation
   Kailani Anastasia.

3) Business Development and Project Manager
   Kourtesi Ioanna.

6. Bibliography

[1] [online]. Available: https://pandas.pydata.org/pandas-docs/version/0.23.4/generated/pandas.get_dummies.html

[2] [online]. Available: https://scikit-learn.org/stable/modules/cross_validation.html

[3] [online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html

[4] [online]. Available: https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-classification-in-python/

[5] [online]. Available: https://machinelearningmastery.com/how-to-develop-multilayer-perceptron-models-for-time-series-forecasting/

[6] [online]. Available: https://www.programcreek.com/python/example/93780/sklearn.neural_network.MLPClassifier

[7] [online]. Available: https://en.wikipedia.org/wiki/Matthews_correlation_coefficient

[8] [online]. Available: https://machinelearningmastery.com/diagnose-overfitting-underfitting-lstm-models/

[9] [online]. Available: https://machinelearningmastery.com/display-deep-learning-model-training-history-in-keras/

7.Contact person

Kourtesi Ioanna email: ioannakourtesi95@gmail.com