Machine Learning 2 - Project Report

Jan Dudzik & Mateusz Cedro

Regression project

In this notebook we would like to present our approach to the regression problems.

Goal

The goal was to find out the way in which insurance companies charge their clients with regard to their social status and health conditions.

Dataset

For the regression task we took the US Health Insurance Dataset[1], which contains data on 1338 people with regard to Age**,** Sex**,** BMI, number of Children, if the person is a Smoker, and Region of living. Charges were the target variable which indicates the cost of insurance levy.

Models

We ahve undertaken three models for regression problem:

- Random Forest (RF)
- Extreme Gradient Boosting (XGB)
- MultiLayer Perceprton DNN model (MLP)

All of the models have been traind on the same train and test datasets.

Approach

Our apporach was as follows: firstly, we built base models with default parameters, and afterwards we did a Grid Search with 3 fold cross-validation to check if we can gain a better performing model.

Evaluation Metric

For the evaluation metric for the model performance we decided to consider the **Root Mean Squared Error** (**RMSE**) loss metrics.

[1] https://www.kaggle.com/datasets/teertha/ushealthinsurancedataset

Conclusion and final remarks

The best performing model turned out to be **Random Forest (RF)** model with following parameters: max_depth=50, max_features=5, min_samples_leaf=5, min_samples_split=2, n_estimators=50, random_state=42, max_samples=553.

The RMSE on the test dataset for three models was as follows:

- Random Forest 4944
- XGBoost 5007
- MLP 5337

The mean value for of charges equals 13270 (USD dollars).

The MLP model turned out to be the one which tend to overfit fastly (perhaps if the dataset was larger, MPL will perform better). Also the MLP model was the one which learning process took the longest amount of time (not mentioning the time consumed on Grid Search with multiple parameters to check on MLP model...)

However, which is lacked in the whole approach is the lack of the data on earnings and the education status of the insured people. To the best of our knowledge such information is considered when computing the insurance levy.

Reproducibility

Note: All model have been trained with set random state, therefore every computation can be fully reproduced. Also data split was done with set random state.

1. Data Description

Dependent Variables

- Continuous variables
    - Age, BMI
- Discrete variables
    - Sex, Smoker, Children, Region,

Target Variable

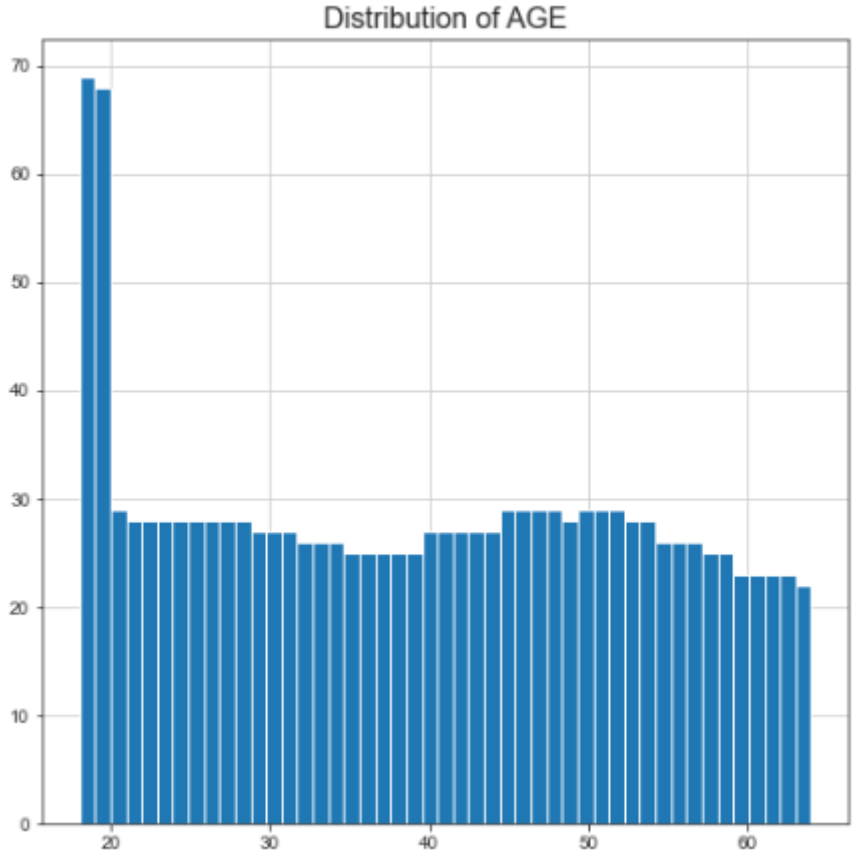
- Charges

Description and Distribution of the variables

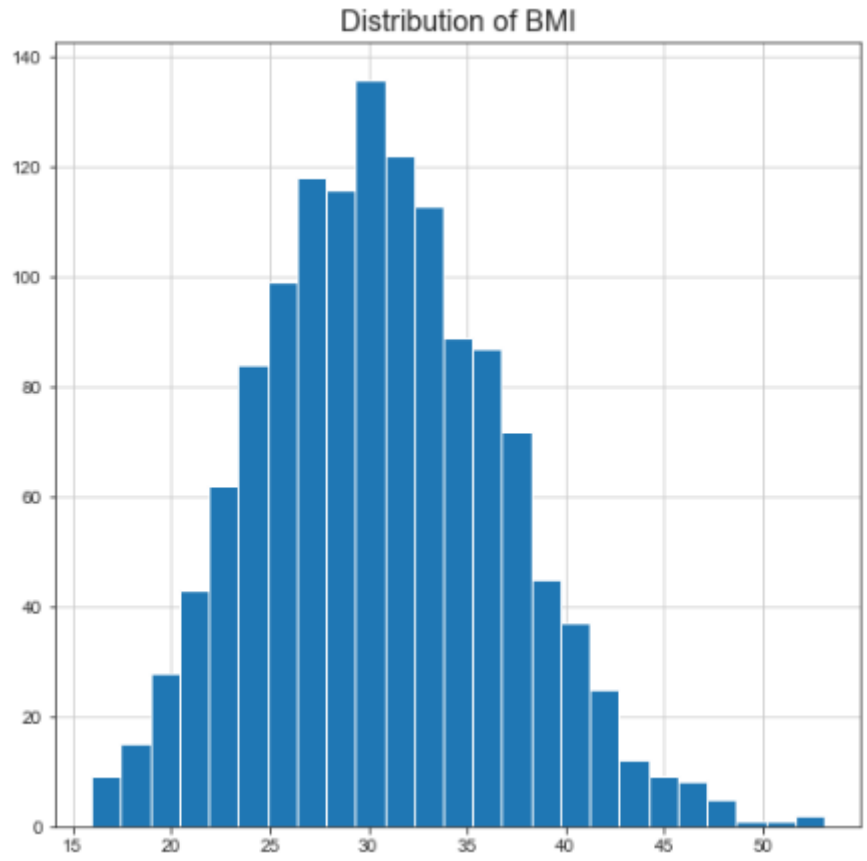
Description

	age	bmi	children	charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.663397	1.094918	13270.422265
std	14.049960	6.098187	1.205493	12110.011237
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.296250	0.000000	4740.287150
50%	39.000000	30.400000	1.000000	9382.033000
75%	51.000000	34.693750	2.000000	16639.912515
max	64.000000	53.130000	5.000000	63770.428010

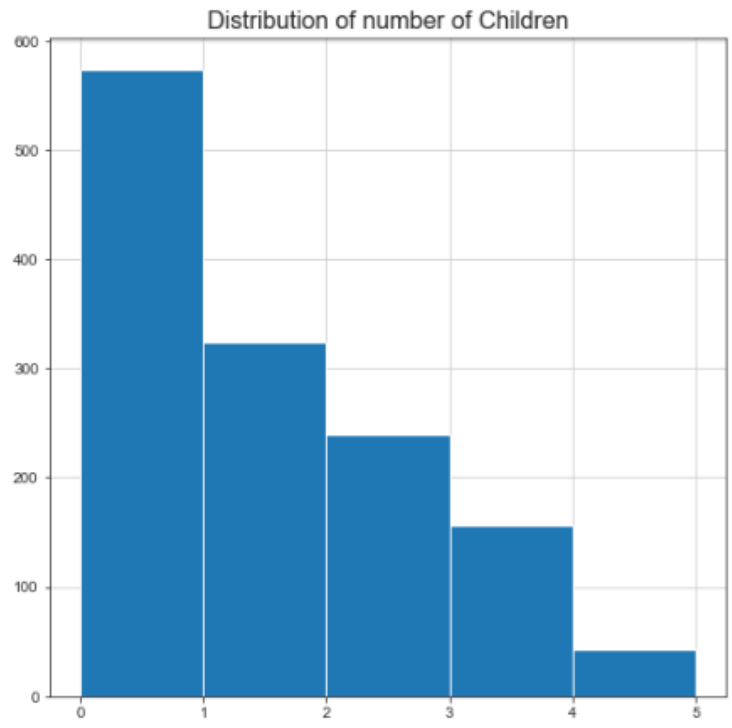
Age



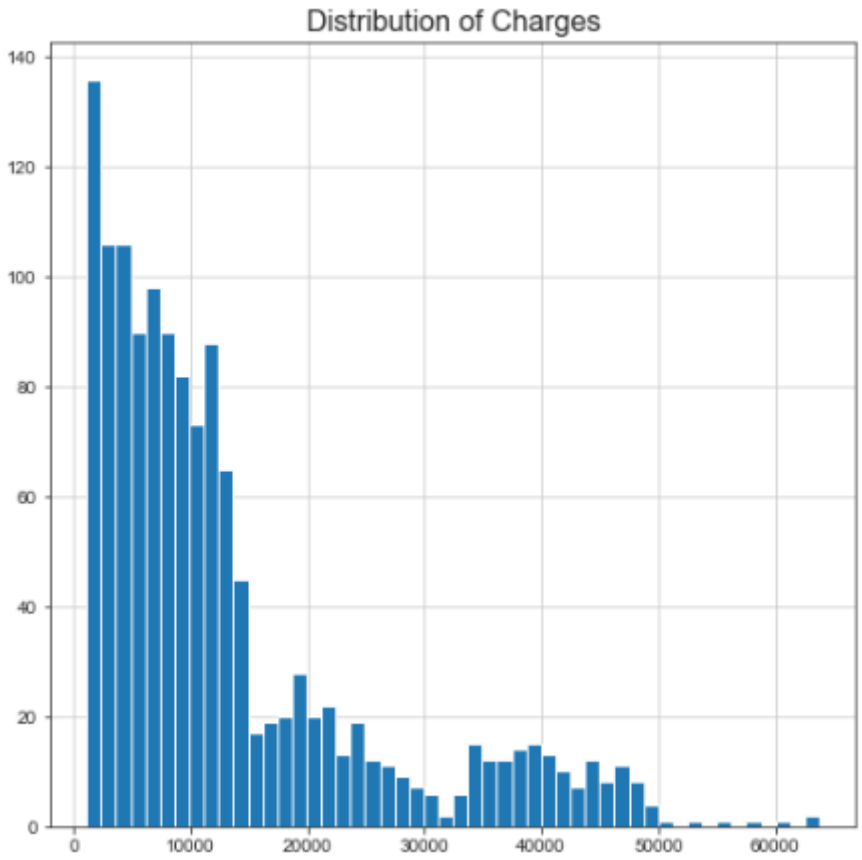
BMI



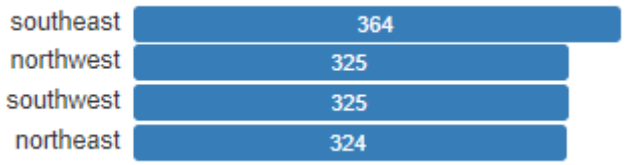
Number of Children



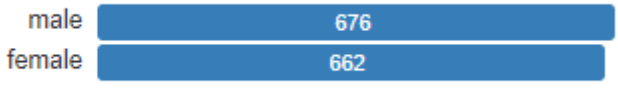
Charges



Region



Sex



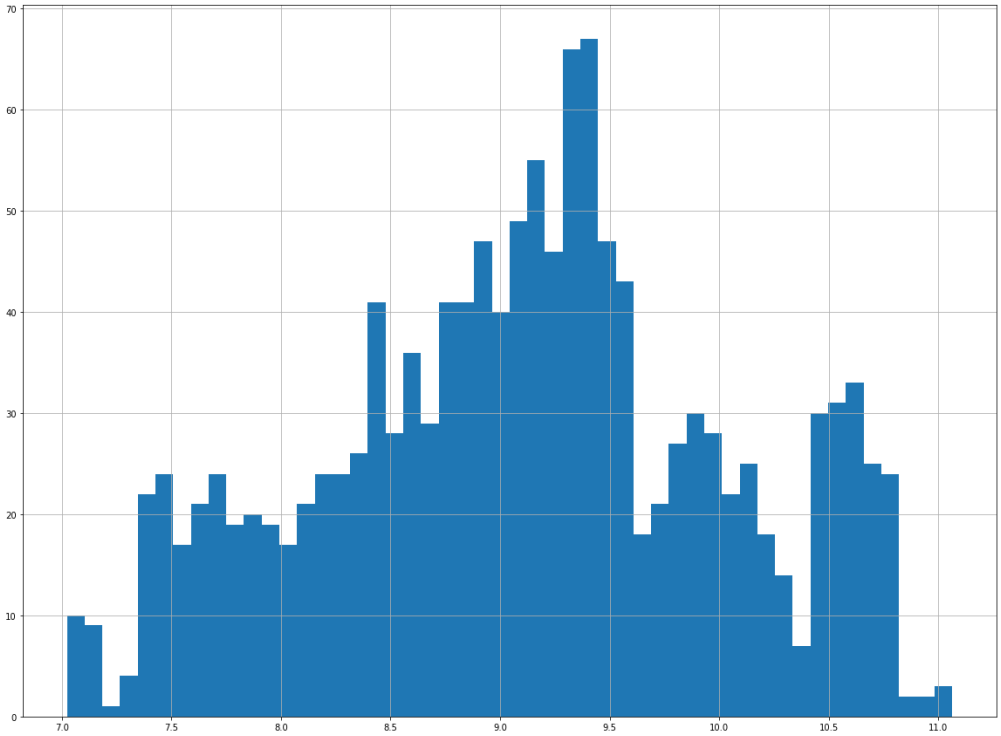
Smoker

2. Data Transformation

We have discovered that the target variable **Charges** is right skewed, therefore we did the logarithm transformation of it.

Log Charges

After the transformation the distribution of Charges looks as follows.



After the transformation the distribution of the charges variable is more normally distributed, however, we have performed the XGBoost and Random Forest on both normal target variable and log targeted variable and surprisingly the RMSE on log target variable was a little bit higher than on not logged, therefore in the following model computation we decided to use the not targeted charges variable.

One hot ecoding

The discrete (cathegorical/factor) variables have been one hot encoded for the ease of the computations - some tree based models has problems with encoding factor variables.

After one hot encoding and deviding the dataframe into X (dependent) and y (target) the data looks like this:

```
1 X
```

	age	bmi	children	sex_male	smoker_yes	region_northeast	region_northwest	region_southeast	region_southwest
0	19	27.900	0	0	1	0	0	0	1
1	18	33.770	1	1	0	0	0	1	0
2	28	33.000	3	1	0	0	0	1	0
3	33	22.705	0	1	0	0	1	0	0
4	32	28.880	0	1	0	0	1	0	0
...	...	...	...	...	...	...	...	...	...
1333	50	30.970	3	1	0	0	1	0	0
1334	18	31.920	0	0	0	1	0	0	0
1335	18	36.850	0	0	0	0	0	1	0
1336	21	25.800	0	0	0	0	0	0	1
1337	61	29.070	0	0	1	0	1	0	0

1338 rows × 9 columns

```
1 y
```

```
0       16884.92400
1        1725.55230
2        4449.46200
3       21984.47061
4        3866.85520
            ...
1333    10600.54830
1334     2205.98080
1335     1629.83350
1336     2007.94500
1337    29141.36030
Name: charges, Length: 1338, dtype: float64
```

3. Data Split

Before computing the models, we have splitted the data into training (80%) and testing (20%). The data split was done with set random state so that the subsets can be fully reproduced. Each model was performed on the same train and test subset.

4. Modelling

As we have mentioned before, three regression models have been performed Random Forest, XGBoost, and MLP. On each of models Grid Search with multiple sets of hiperparameters was checked and 3 fold cross-validation was performed.

4.1 Random Forest

First base RF model had RMSE equal to: 5236.43 on test dataset.

After that, the Grid Search with following set parameters was computed:

- max_depth: [30, 40, 50],
- max_features: [2, 3, 4, 5],
- min_samples_leaf: [1, 2, 3, 4, 5],
- min_samples_split: [2, 4, 6, 8],
- n_estimators: [30, 40, 50]

With 3 fold cross validation, we have 720 candidates, and in total 2160 fits was done.

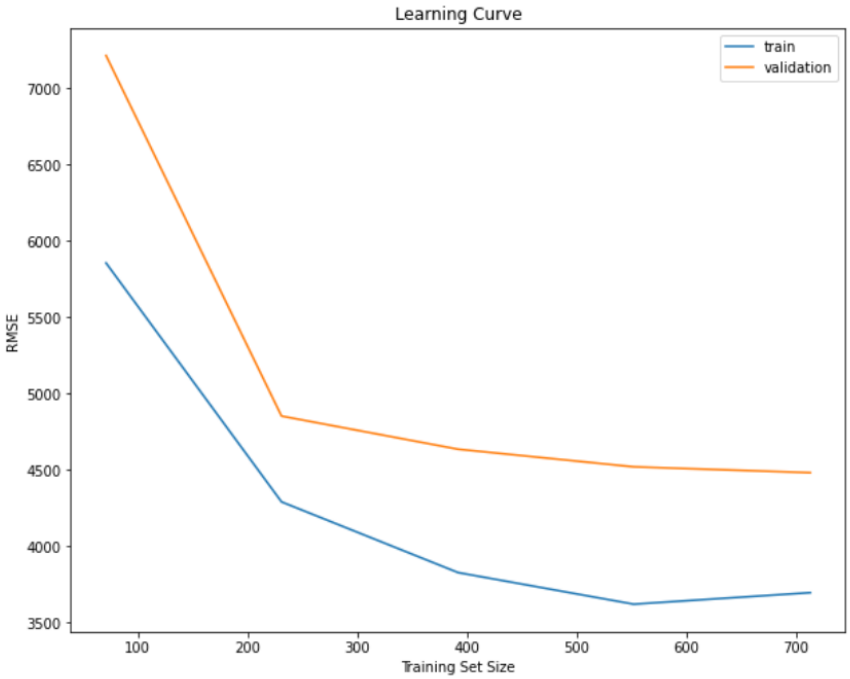
On best performing model on training size with parameters:

- bootstrap=True
- max_depth=30
- max_features=5
- min_samples_leaf=5
- min_samples_split=2
- n_estimators=50

Test RMSE is equal to 4964.49

However, after plotting the learning curve, we have discovered, that filling the model with all observations from the training sample, model tends to slightly overfit when training size is higher than 553 samples.

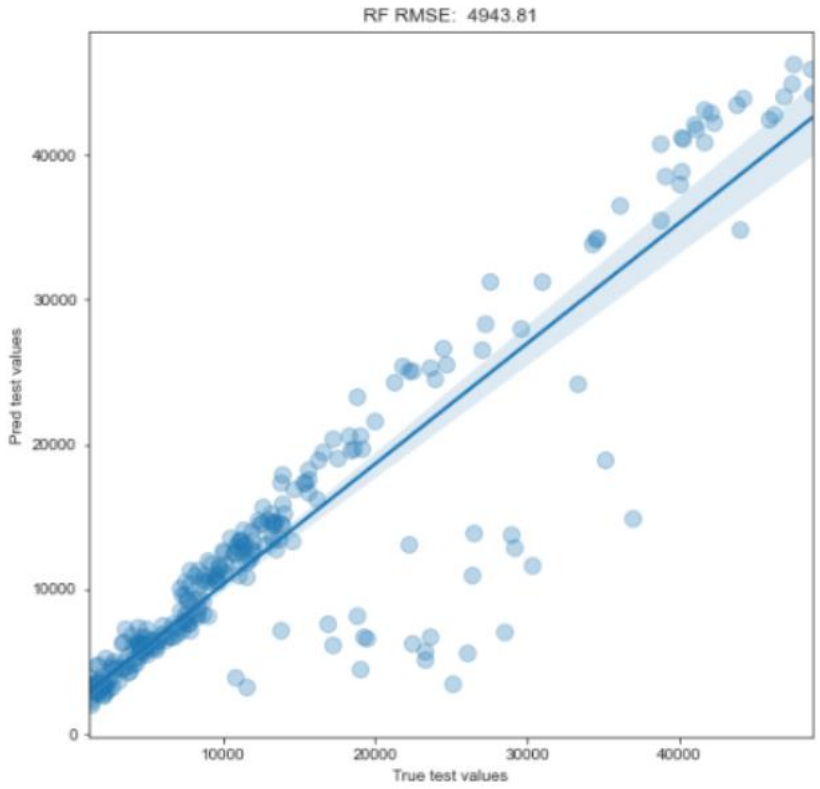
Learning Curve



So afterwards we added to the model the max_samples=553 parameter, and the RMSE dropped to 4943.81

Y predicted and Y true

The plot between true test y values and predicted y values look like this:



4.2 XGBoost

After FR model, the XGBoost model was performed. The RMSE on base XGBoost model was equal to: 5063.90 on test dataset.

After that, the Grid Search with following set parameters was computed:

- max_depth: [2,4,6,8],
- min_child_weight: [1,3,5,7,9],
- n_estimators: [50, 100, 150, 200],
- learning_rate: [.01, .03, 0.05, .07],

With 3 fold cross validation, we have 320 candidates, and in total 960 fits was done.

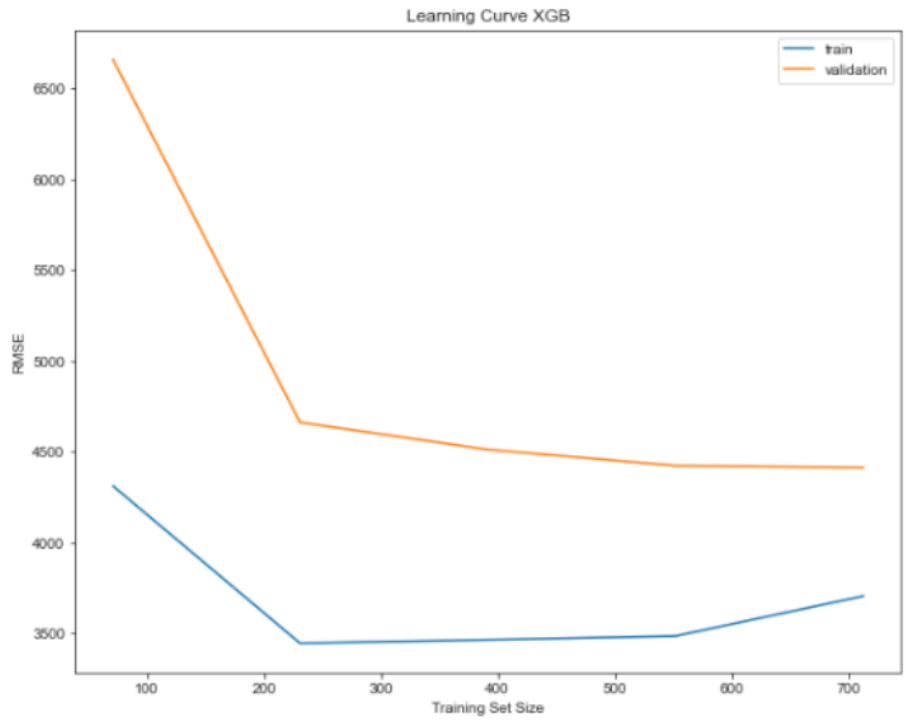
On best performing model on training size with parameters:

- learning_rate=0.05
- max_depth=4
- min_child_weight=9
- n_estimators=100

Train RMSE is evual to 4415.24 However, on test RMSE is evual 5007.14 (quite a big difference!)

After ploting the leraning curve, we have discovered, that filling the model with all observations from the training sample, model tends to slightely overfit when training size is higher than 234 samples.

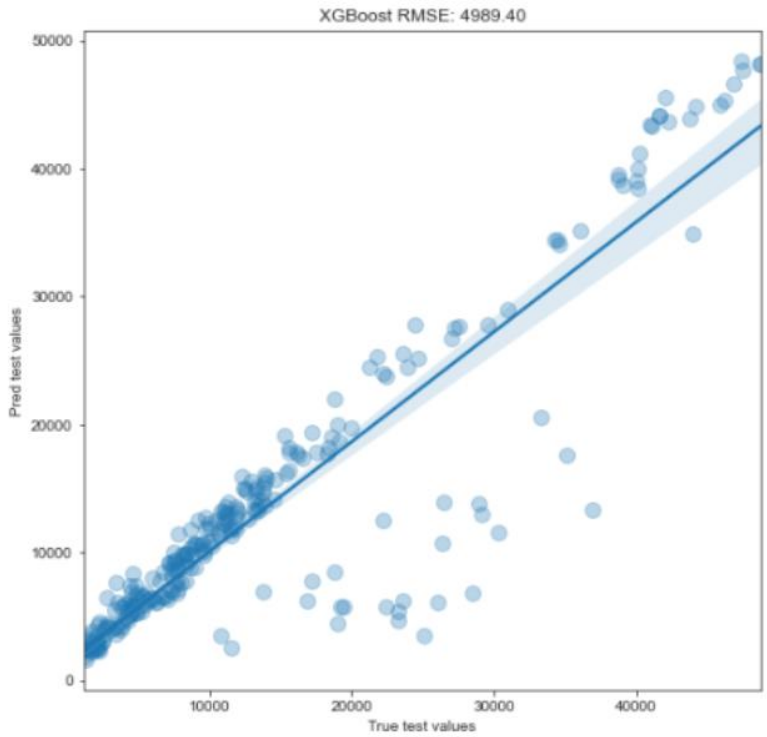
Learning Curve



So afterwards we added to the model the max_samples=234 parameter, and the test RMSE dropped to 4989.40

Y predictd and Y true

The plot between true test y values and predicted y values look like this:



4.3 MultiLayer Perceptron

After tree based models, we have decided to check the Deep Neural Network MultiLayer Perceptron model.

For the base model we decided to have MLP model with three hidden layers which contains of 150, 100, and 50 neurons respectively. For the activation function we decided to have ReLu, Adaptive Moment Estimation (Adam) optimizer, and 500 iterations.

Base model had RMSE 6247.46 on test set. Quite big difference when comparing to the tree models. Perhaps if we have more data, then the DNN model would outperform tree based models.

Then two Grid Searches was performed.

In the first Grid Search, the following set parameters were as follows:

- hidden_layer_sizes: (150,100,50), (180,90,45), (120,80,40),
- max_iter: 500, 700, 900, 1100,
- activation: tanh, relu,
- solver: sgd, adam,
- alpha: 0.0001, 0.05,
- learning_rate: constant, adaptive,

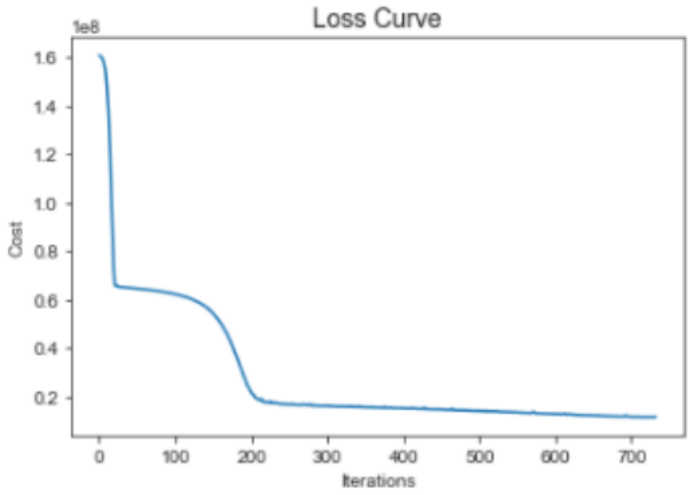
With 3 fold cross validation, we have 192 candidates, and in total 576 fits was done.

On best performing model on training size with parameters:

- hidden_layer_sizes: (180,90,45), - The highest possible
- max_iter: 1100, - The highest possible
- activation: relu,
- solver: adam,
- alpha: 0.05,
- learning_rate: constant,

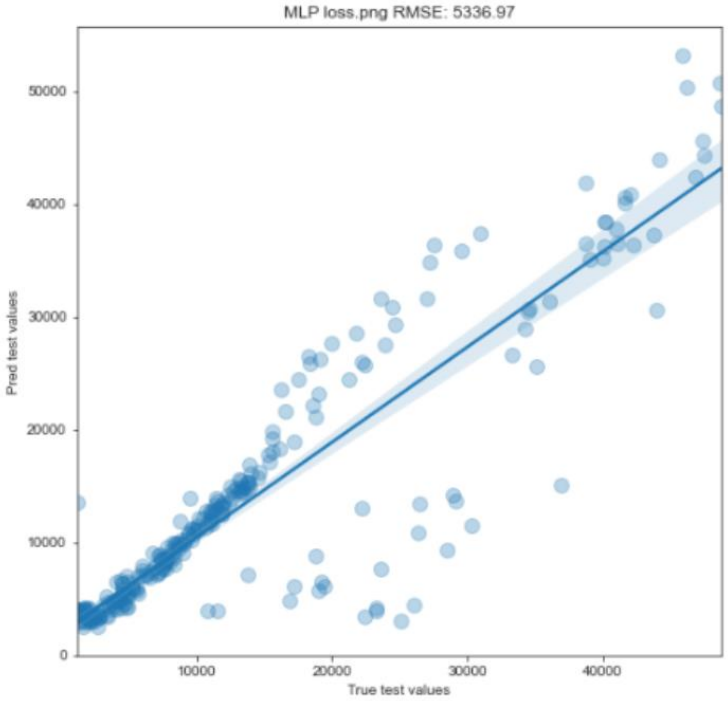
Test RMSE is equal to 5336.97

Learning Curve



Y predicted and Y true

The plot between true test y values and predicted y values look like this:



Because the best performing architecture was the one which contains the highest values (most complex architecture and max iterations), we have decided to perform second Grid Search with four hidden layers and higher max iterations.

Second Grid Search contains such a set parameters:

- hidden_layer_sizes: (180,90,45), (180,90,45,23), (240,120,90), (240,120,90,45), - higher architectures and 4 hidden layers in two of them
- max_iter: 300, 500, 700, 900, 1100, 1300, 1500
- activation: relu,
- solver: adam,
- alpha: 0.05,
- learning_rate: constant,

With 3 fold cross validation, we have new 28 candidates, and in total 84 fits was done.

The best one turned out to be:

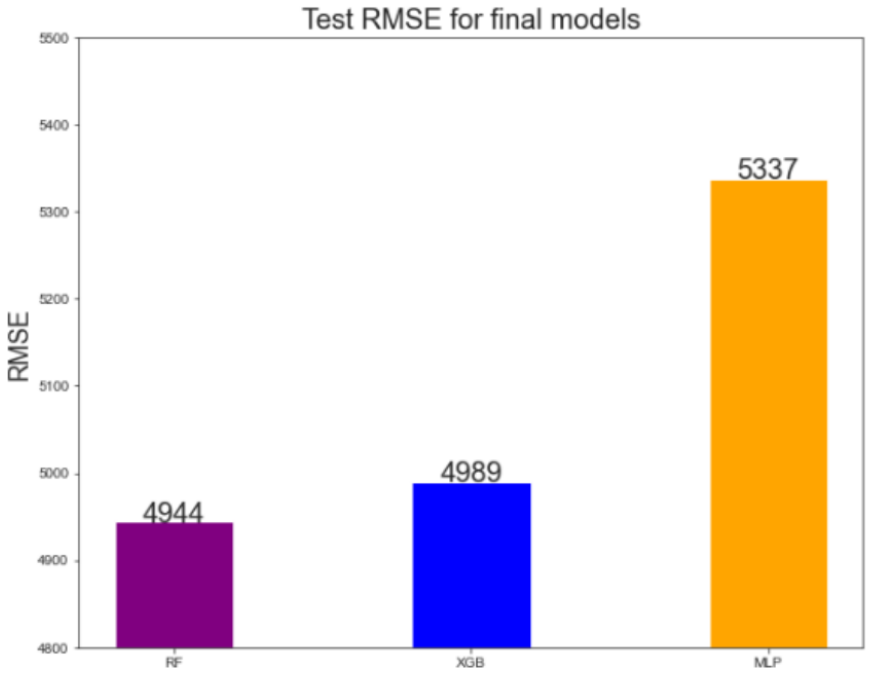
- hidden_layer_sizes: (240,120,90,45),
- max_iter: 700
- activation: relu,
- solver: adam,
- alpha: 0.05,

- learning_rate: constant,

However on the test set we had poorer REMS result 5460.39 - this is perhaps because we have 240*120*90*45 = 116 640 000 hidden neurons on relatively small dataset.

4.4 Model Comparison

Final model comparison

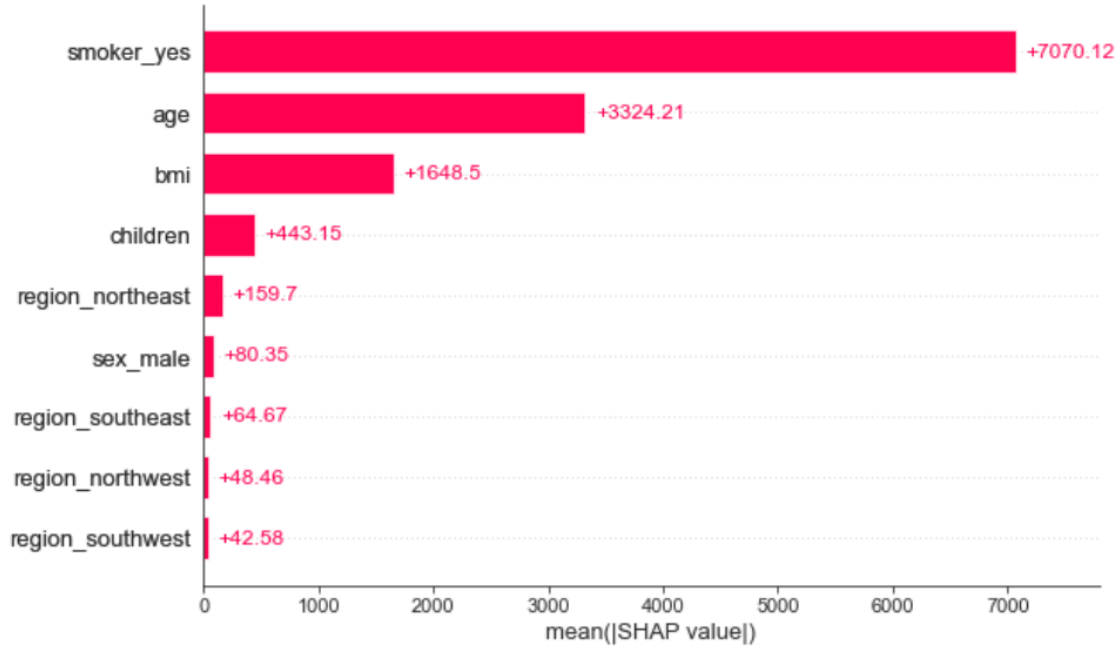


The best model with the lowest RMSE which has been performed is Random Forest model with RMSE equal to 4944 on test size.

5. eXplainable Artificial Intelligence (XAI)

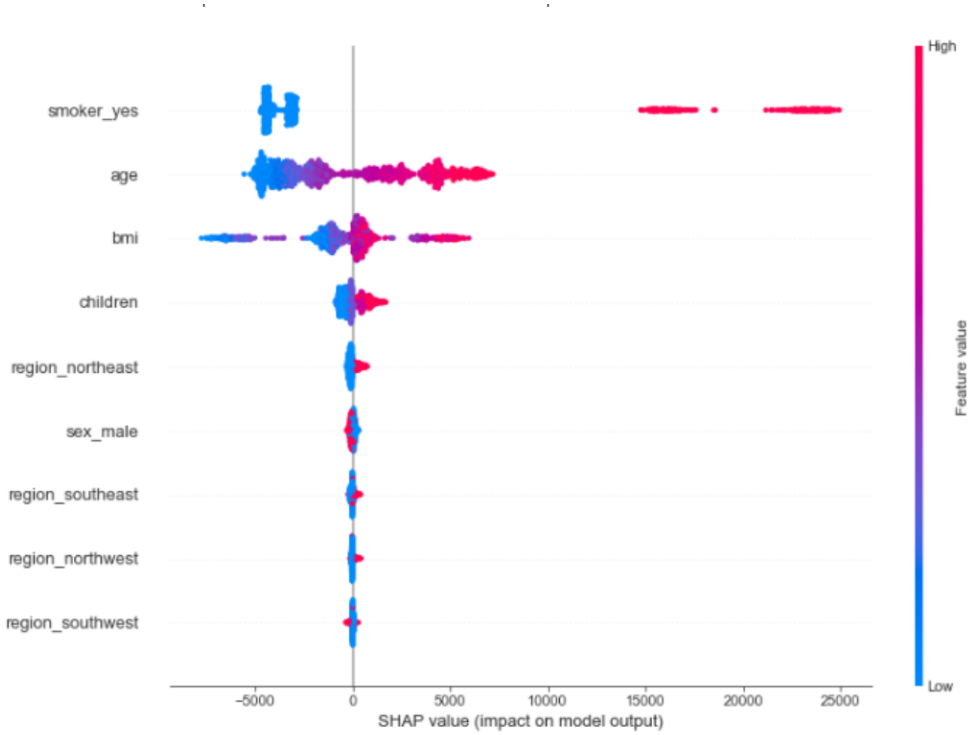
For the best performing model (FR) we have done the XAI model explanations with SHAP library which helps in model explanations basing on Shapley values and feature attribution.

The plot of variable importance is presented below:



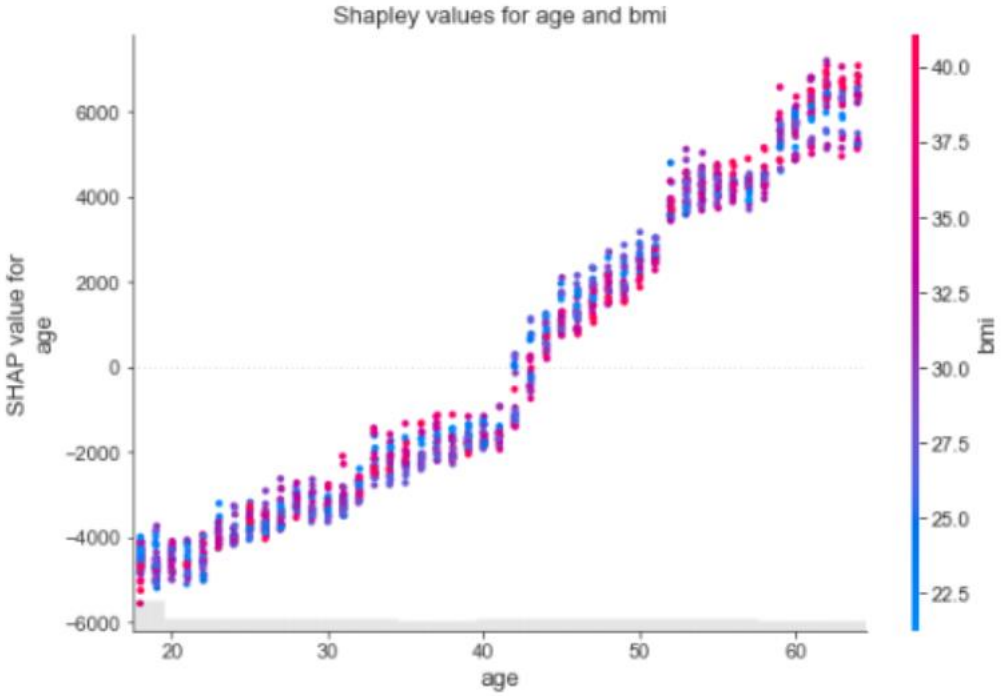
We can clearly see that the variables which contributes most in calculating mean Shapley values for the Charges levy are Somker_yes, AGE, and BMI, where the smoker contributes twice as much as AGE, and fourth as much as BMI. Interestingly, Sex variable contributes less than region of Northeast of living.

Below the more detailed plot with distributions of variables are presented:



We can clearly see the distinction for shapley values for each variable

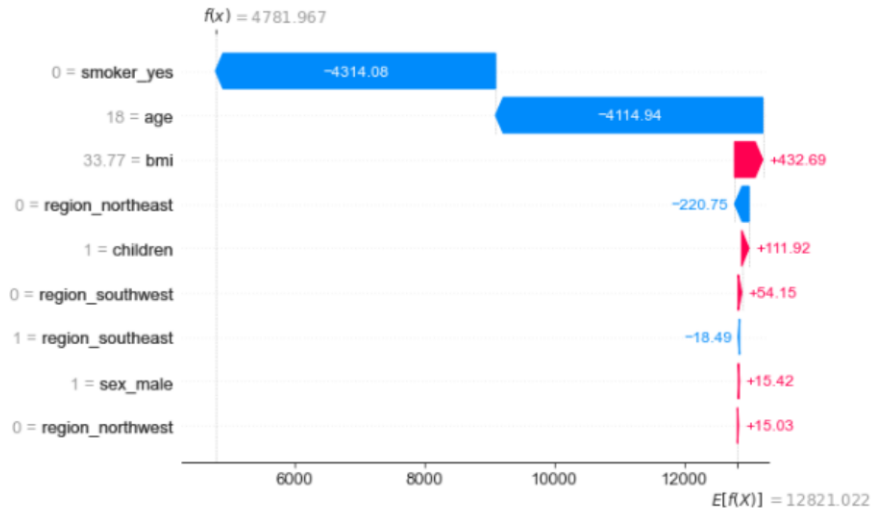
Below we can see the plot describing the AGE and BMI Shapley values:



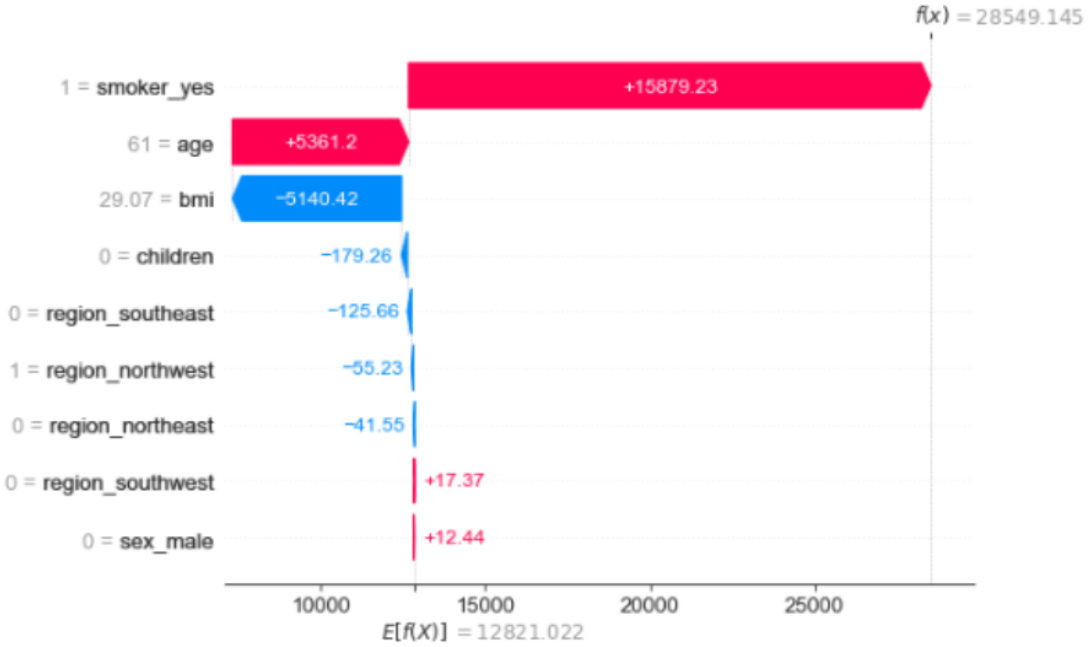
We can clearly see that the age is strictly positively correlated with the Shapley values for additional value of Charges levies. For Age=43 we can see that Shapley values are around zero; age below 43 decreases the insurance charges amount, and higher age than 43 increases the values for insurance levies.

Finally, we would like to present two observations:

- First: 18 years old non-smoker



- Second: 61 years old smoker



We can see the exact differences between these two observations

6. Conclusions

In this report we have presented our approach to developing different regression models, hyper parameter-tuning, evaluation and final model explanations using Shapley values.

The best performing model was Random Forest model which performed better than XGboost and MLP models.