****

**Fachbereich:** Data Science

**Lehrveranstaltung:** SE Advanced Data Challenge

**LV-Nummer:** 405.982

SS 2020

Projektarbeit zum Thema:

**„Hit or Miss? Ein Projekt zur Vorhersage der Popularität von Liedern mithilfe von Supervised Learning“**

**Vorgelegt von:**

Zivkovic Patrick (01422129)

Angerer Michael (xxxxxxxx)

**Abgabedatum:**

31.07.2020

**Wortzahl:**

xxxx

[Abstract 1](#_Toc47090987)

[Introduction 2](#_Toc47090988)

[Original Dataset 2](#_Toc47090989)

[Spotify 2](#_Toc47090990)

[Loss of Data 3](#_Toc47090991)

[Genres 3](#_Toc47090992)

[Natural Language Processing 4](#_Toc47090993)

[Final Dataset 4](#_Toc47090994)

[Descriptives 5](#_Toc47090995)

[Modelling 7](#_Toc47090996)

[Baseline Model 8](#_Toc47090997)

[Generalized Linear Model 8](#_Toc47090998)

[Neural Network 9](#_Toc47090999)

[Random Forests 9](#_Toc47091000)

[Gradient Boosting 9](#_Toc47091001)

[Stacked Ensemble BOF 10](#_Toc47091002)

[Stacked Ensemble All 10](#_Toc47091003)

[All methods in comparison 10](#_Toc47091004)

[Conclusion and Future Work 11](#_Toc47091005)

# Abstract

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

# Introduction

Our orginal idea was to use the attached *song-lyrics* dataset (songdata.csv) to perform natural language processing techniques on it and hopefully extract some interesting information. We then decided to take an extra step and tried to use the acquired knowledge from the song-lyrics, among other variables, towards a prediction of the overall popularity of a song. The following work will describe all the steps and methods we used to be able to accomplish this goal.

# Original Dataset

The original dataset, that we found on Kaggle, is very simplistic. It lists the title of the song along with the artist and the associated lyrics, mostly without punctuation. Additionally, it provides information about where the lyrics of the songs were retrieved from in the form of a link. The amount of songs the dataset contains amounts to 57650.

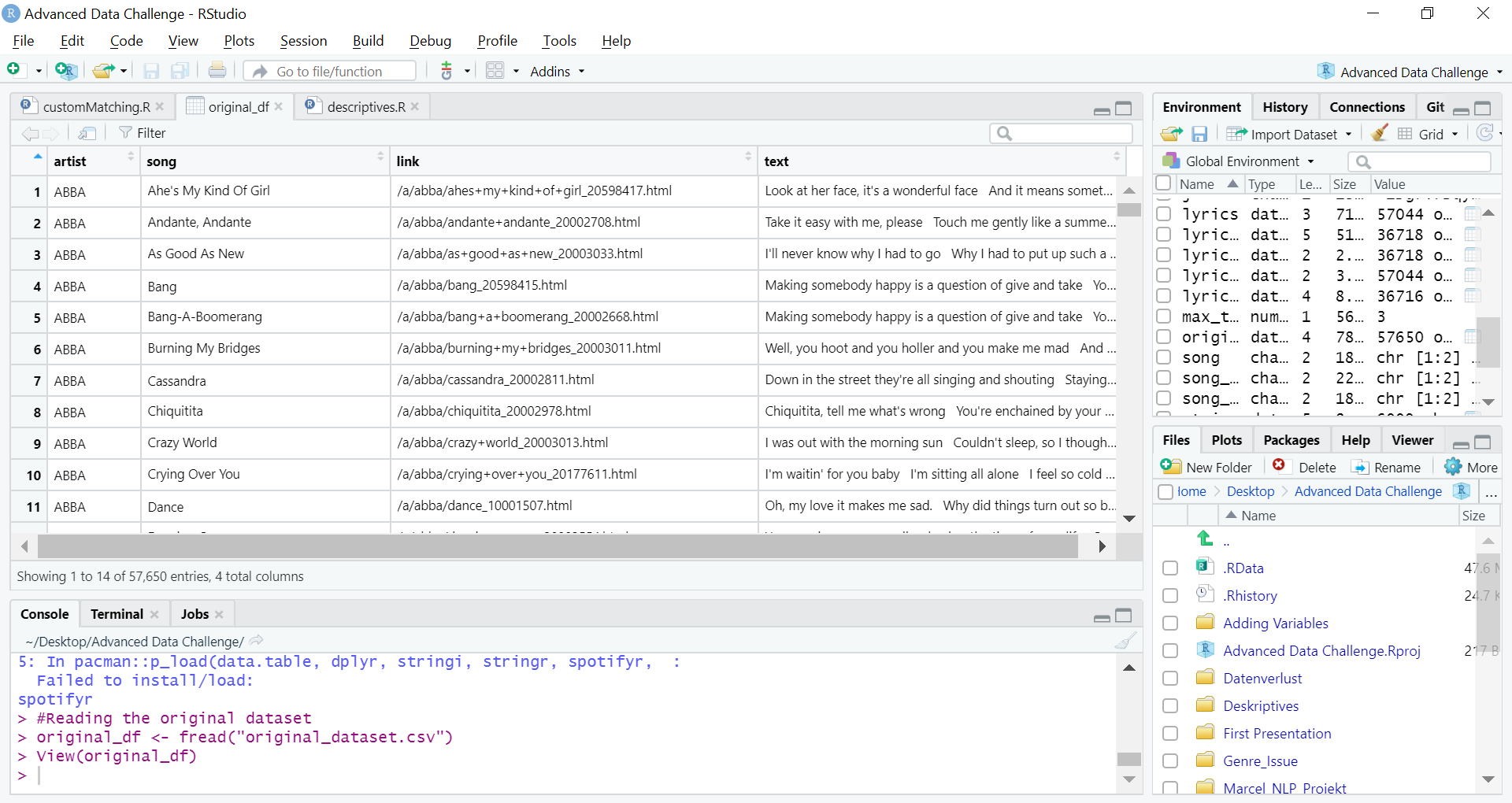


Table 1: Overview of the first few rows of the original dataset

# Spotify

[Hier hab ich ein bisschen was von deinem Part gemacht, damit ich im nächsten Kapitel daran anschließen kann 😉]

Since some song titles from the original dataset were misspelled, like in the example of the Beatles’ song “She’s My Kind Of Girl” that turned out to be “Ahe’s My Kind Of Girl”, we had to use a matching function in order to join the songs from our original dataset with the songs from the Spotify library.

The matching was performed with help of the R package *stringdist* and more specifically the function *amatch*. The integer parameter *maxDist* determines the value of how “similar” the song titles have to be in order to call them a match. The metric used to determine that similarity was the *restricted Damerau-Levenshtein distance.* What it does, in a nutshell, is counting the number of deletions, insertions and substitutions necessary to turn string b into string a.

# Loss of Data

During the intent of matching the song titles from our original dataset with the songs from the Spotify data base, a lot of titles were mismatched or could not find a corresponding match at all. All in all, our original sample size of 57650 songs shrank to a final number of 36889 (about 64% of the original dataset). Some reasons for the loss of data and some problems that led to mismatching are going to be listed in the following.

Initially we chose the *maxDist* parameter of our matching function to a fixed value of 2. That led to problems because all song titles were being treated in the same way disregarding their length. This especially led to mismatches with very short song titles, since the matching function allowed for a fixed amount of 2 differences between the strings. In one example a song named “Hope” was assigned to a song named “Home” and in another one the song title “God” was assigned to the song title “Gold”. This issue was solved by making the *maxDist* parameter dependent on the length of the song title, namely set it to a value equal to 15% (rounded down) of the amount of characters in the string. That way particularly small song titles would need to be exactly equal in order to propagate a match.

Our original dataset contained a comparatively small number of artists considering the large amount of songs. 598 artists on 57650 songs which averages to around 96 songs per artist. For that reason, most of the songs of the whole musical career of these artists were covered in the dataset. Some of those songs, however, were not featured in the Spotify library, so a lot of songs were lost due to a simple lack of availability.

By far the biggest reason why we experienced such a big data loss was due to appendages to the song titles in the Spotify library. There are a lot of songs that add additional information to the song title. For example, another Beatles song from the original dataset "All I’ve Got To Do” was only available on Spotify under the name "All I’ve Got To Do - Remastered 2009". Our matching function could, of course, not deal with such irregularities.

We decided not to deal with this problem in the scope of this project, because we found that our dataset with 36889 songs was still big enough for the later modelling.

# Genres

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

# Natural Language Processing

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

# Final Dataset

The following table gives an overview on our final dataset after all feature collection and extraction steps mentioned above were completed:

|  |  |  |
| --- | --- | --- |
| **variable** | **description** | **example** |
| artist | The artist’s name | ABBA |
| songname | The title of the song | She’s My Kind Of Girl |
| year | The year the song was released in | 1973 |
| popularity | The popularity of the song (0-100) | 20 |
| duration\_s | The duration of the song in seconds | 165 |
| artist\_popularity | The popularity of the artist (0-100) | 80 |
| genre | An overgenre of the artist | pop |
| Coleman\_Liau | Readability index of the lyrics (mean = 2.39) | 1.97 |
| SMOG | Readability index of the lyrics (mean = 5.68) | 4.66 |
| trust | Sentiment indices (ranging from 0 to 1) | 0.23 |
| fear | 0 |
| negative | 0 |
| sadness | 0.08 |
| anger | 0 |
| surprise | 0.08 |
| positive | 0.31 |
| disgust | 0 |
| joy | 0.27 |
| anticipation | 0.04 |
| neg\_sent | How positive/neutral/negative is the song in terms of the sentiment-words and their sentiment value? (starting from 0) | 0 |
| neutral | 8 |
| pos\_sent | 13 |

# Descriptives

In the following we are going to present some interesting descriptives of our dataset, mostly focusing on the variables, that are most relevant for the building of our models in the next chapter.

Figure 2 shows how the songs of our dataset are spread out over the decades. The number of songs gradually increases peaking in the 2000s with just above 10000 songs.

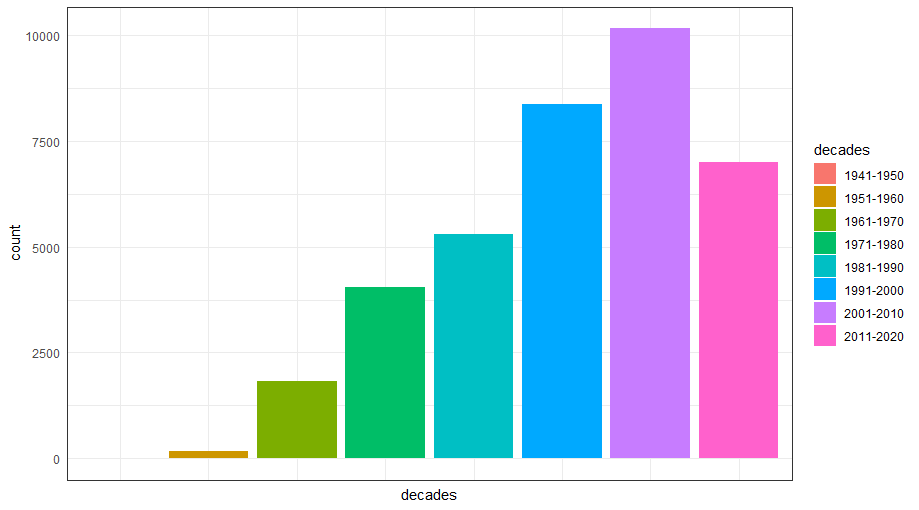
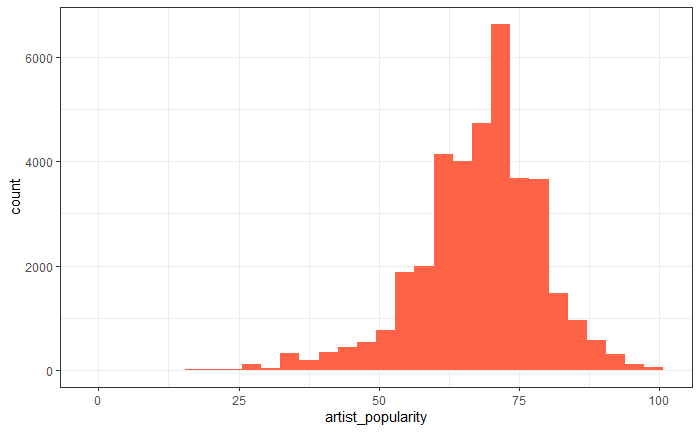
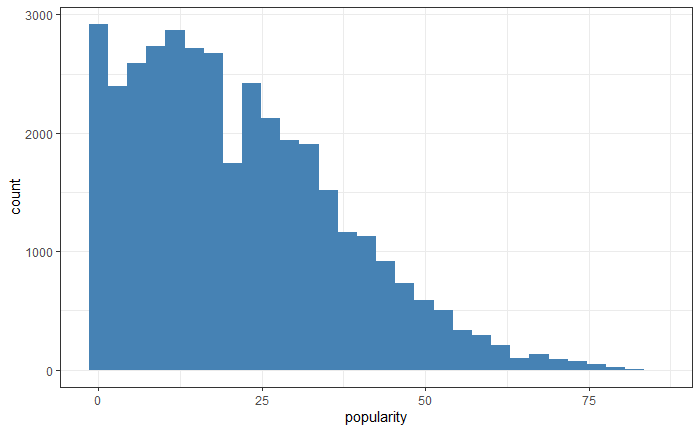
*Figure 2: Number of songs by decade*

Figure 3 and Figure 4 show the scatter of the artist popularity and the song popularity. It is immediately evident, that the song popularity has a distribution that is strongly skewed to the left, meaning that most of the songs of our dataset have a popularity that ranges between 0 and 50 although the variable generally ranges between 0 and 100.

*Figure 3: The distribution of the variable artist popularity*

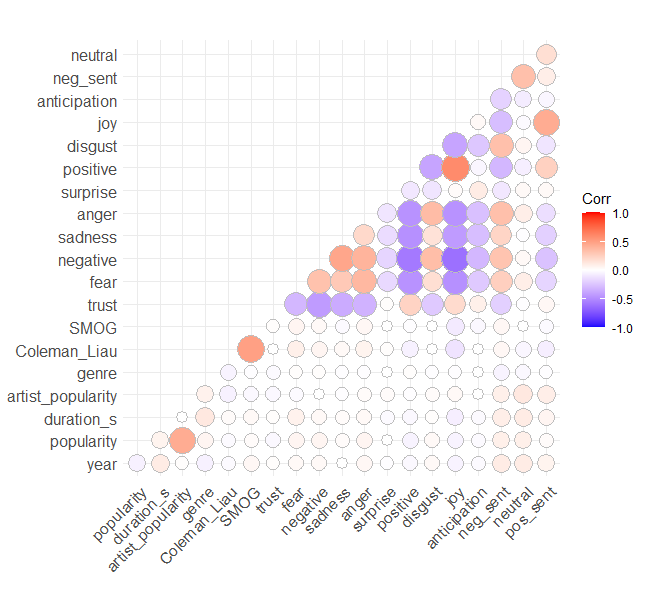


*Figure 4: The distribution of the values of the song popularity*

# Modelling

In this chapter we are going to report the results we achieved trying to predict the popularity of a song with all the variables at our disposal. First, we are going to take a look at a baseline model to set a benchmark for the more complex models that will follow later.

Figure 5 shows a correlation plot illustrating all interdependencies among the variables of our dataset. Strong correlations are found between the variables popularity and artist popularity, the two readability indices Coleman-Liau and SMOG and in general between variables of the same general sentiment.



*Figure 5: Correlation plot with all the variables of the dataset*

For all methods described below, a 10-fold cross-validation was performed in order to get an accurate measure of how well the particular method is suited for this kind of problem. The mean absolute error was used as a metric to determine the quality of the prediction performed on the test set. It was calculated for every single run and then averaged over the 10 different runs for a result, that is comparable with the other methods used.

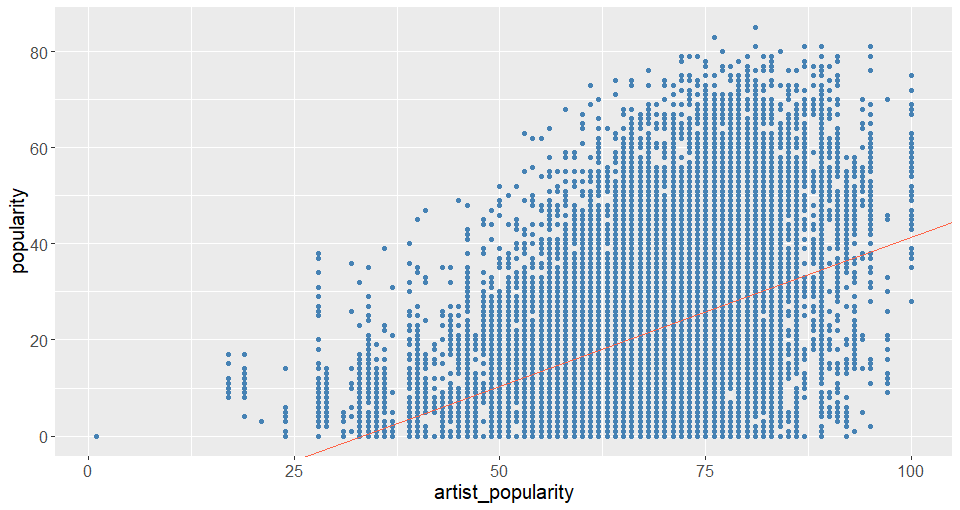
## Baseline Model

For some baseline results, we simply calculated the mean value of the song popularity for a training set and predicted the same value for all the songs of the test set. The resulting average mean absolute error was **12.758** with a standard deviation of 0.089 across the 10 runs of the cross-validation. This result already appears to be pretty good, since it suggests that the popularity of any given song can be predicted with an accuracy of ±12.758 in a space that ranges between 0 and 100. On closer inspection we can see that the predictions are only accurate in the subspace where most values are populated, which is between 0 and 50 (see: Figure 4).

## Generalized Linear Model

As our first method, that also took our independent variables into account, we chose a simple multivariate regression. We performed a forward selection in order to determine which variables have the largest impact on our dependent variable song popularity and to incorporate them into our model. Judging by the adjusted R-squared values, the four most important variables were artist popularity, the sentiment variable joy, the release year and the duration of the song. Figure 6 shows that the strongest independent variable artist popularity is highly correlated with the song popularity (r = 0.427) and contributes the most towards good predictability. This does not come as a surprise, since already popular artists tend to release songs, that become popular as well.

The cross-validation showed that the model with the best result was the one, that used a total number of 12 variables of the 19 variables our dataset contains. The average mean absolute error for that model turned out to be **11.301** with a standard deviation of 0.029.

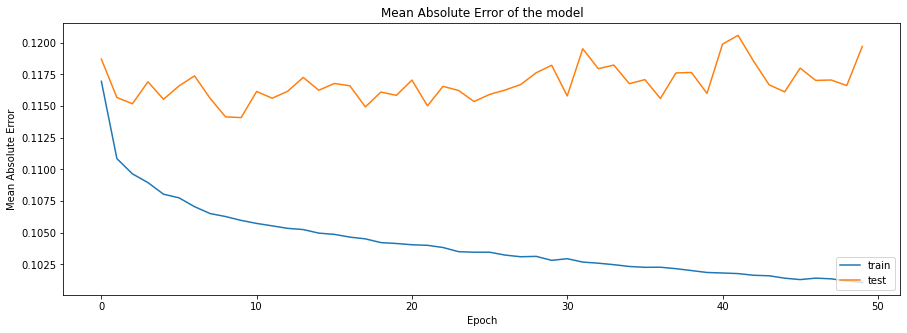


*Figure 6: Scatterplot showing the relationship between the artist popularity and the song popularity*

## Neural Network

Next, we trained a fully-connected neural network in the Keras environment in Python. We set it up by building two hidden layers with 6 hidden neurons in total, because that number seemed to produce the best results for this particular problem. We were training the model for 50 epochs, using the technique of early stopping if the validation error did not substantially change for 10 epochs.

Figure 7, however, shows that a neural network approach is not suitable for this problem, since the test error does not considerably decrease. The result of the approach shows almost no improvement compared to the linear regression having an average mean absolute error of **11.066** and a standard deviation of 0.69.



## Random Forests

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

## Gradient Boosting

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

## Stacked Ensemble BOF

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

## Stacked Ensemble All

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

## All methods in comparison

|  |  |  |
| --- | --- | --- |
| **Model** | **Mean Absolute Error (averaged over 10 folds)** | **Standard  Deviation** |
| Baseline Model | 12.758 | 0.089 |
| GLM | 11.301 | 0.029 |
| Neural Network | 11.066 | 0.69 |
| Random Forests | 9.701 | 0.105 |
| Gradient Boosting | 9.178 | 0.104 |
| Stacked Ensemble BOF | 8.927 | 0.019 |
| **Stacked Ensemble All** | **8.721** | **0.054** |

# Conclusion and Future Work

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.