Patient's age prediction based on medical diagnosis of teeth maturity *

De La Torre Camilo *Universite Toulouse 1 Capitole*
Campan Alexis *Universite Toulouse 1 Capitole*

Our study involves patient's age prediction based on teeth maturity that is assesed by doctors. Teeth are characterised by several levels of maturity (going from A to H), data is often incomplete and for some patients, teeth can't be evaluated because the patient does not have them yet. After several attempts to mitigate the incompleteness of our records, we obtained a very low Mean Absolute Error of 0.9 years in predicting unseen data. The model that best generalized to unseen records was a tuned GBRT (Gradient Boosting Regression Tree). Lastly, we also analyse the importance of different teeth maturities and interestingly discovered that not all teeth have the same predictive power on patient's age.

Keywords: Teeth, Age, Regression, Machine Learning, GBRT

Introduction

The data we study explores teeth maturity of several individuals, specifically, we analysed 2847 records. By teeth maturity we mean that each tooth's condition is analysed by a doctor and classed on one of the possible maturity levels (going from A to H, A being the least mature, and H being the most mature). A total of 8 teeth are analysed by doctors, these involve : two incisors teeth, one canine tooth, two premolar teeth, finally, three molar teeth. In addition to teeth maturity indices, we also know the patient's sex, age and medical ID.

Incompleteness of the data

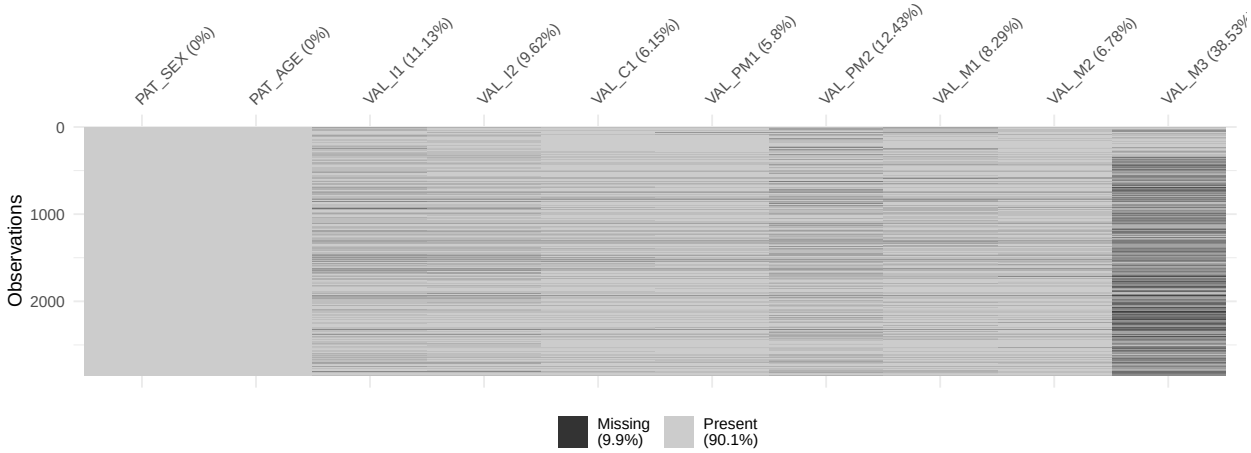
The data at hand is not complete: out of our 2847 records, for 1436 records at least a tooth score is missing (50% of the data). Moreover, not all teeth are equally misrepresented, as we might expect, molar teeth scores are less likely to be present in our records because several patients do not have them yet. In addition, because of multiple reasons, is it possible that a patient of a given age lost or had removed one or several teeth, this, we think, complicates missing values interpretation because several causes, unrelated to age, can affect it.

Figure X shows missing values per column in our data, moreover, it shows the interaction between missing values across columns.

We observe that our dependent variable, patient's age, is always complete. We also note that the last molar has a very high percentage of missing values. Besides the last molar (M3), another two teeth have over 10% of missing values (I1 and PM2).

```
## Warning: ‘gather_()‘ was deprecated in tidyr 1.2.0.
## Please use ‘gather()‘ instead.
```

*We would like to thanks the doctors who delivered their experienced assesment of teeth maturity as well as our teacher who provided us the data to work with.



Nonetheless, because of the first reason of incompleteness we presented (that teeth grow at different stages in life) we think that imputing missing values makes sense, our approach is explained when we discuss about our methodology.

Methodology

As predictive power is our main goal, we are interested in finding the model that minimizes an error score on unseen data. The score we chose is the Mean Absolute Error defined as $mae = (\frac{1}{n}) \sum_{i=1}^{n} |y_i - f(x_i)|$ where $f(x_i)$ is our predicted age.

To do so, we adopt a classical random split of our data to avoid any data leakage and test model generalization. For algorithms that do not make a validation set right out the bat, i.e. *XGBoost*, we make it ourselves to ensure that we do not overfit our model on test data because of Early Stopping and other techniques.

Ordinal encoding

Teeth maturity comes in a mixture of numbers and letters levels. Maturity scores ranges from 0, 1, A,. . . , H, 0 being the least mature and H being the most mature. Several algorithms do not work with ordinal data or perform poorly after one hot encoding of such data.

In order to compare algorithms but also to facilitate convergence on some algorithms we adopt the strategy of ordinal encoding. That is, we assign an integer value to each categorical level. The transformation is monotonic for the purpose of keeping the order of teeth maturity scores.

We assume that maturity scores are based on a linear scale, in other words, the difference between A and B is of the same magnitude as the difference between G and H.

Handling missing values

Removing very incomplete data

As discussed before, several samples of the data contain missing values. Figure X showed that for several patients, multiple teeth scores where missing, i.e. some samples presented a continuous black line, indicating that for all teeth, the maturity scores are not available. Because of this observation, we decided to exclude some samples from the training data in order to prevent

the imputation technique, discussed below, to base its neighboring on an insufficient subset of columns.

Specifically, we removed samples in the training set for which information about 7, out of 8, teeth is missing. That was the case for 108 samples in the training set, reducing its size from 2135 observations to 2027 observations.

K Nearest Neighbors imputation

We believe that, in general, patients of similar age have similar teeth maturities, hence, we are convinced that for our prediction task, samples that are alike will tend to have the same age. Therefore, if tooth information is missing from a sample, we believe that it could be estimated by looking at analogous samples existing in the training data. This motivates us to use an imputing algorithm that performs some multivariate estimation of missing values using complete observations.

We performed K Nearest Neighbors imputation on the missing values in the training set and testing set. We set the vicinity used for imputation to 40 neighbors, these are weighted by euclidean distance.

Because we believe that the missingness of a tooth score is informative about a patient's age, e.g. wisdom tooth removal, we created a binary features indicating the presence of missing values for each imputed column and sample.

Outlier removal

After all these steps, our last process before training is to remove the outliers. Removing outliers consist of exclude unusual values and so reduce the variability in our data. We have tried several techniques for detect these outliers: - PCA: we fit_transform our data and then we calculate the MSE score for each point in order to drop the 10% lower (we have tested 10% upper and 5%-5% but lower values was the best to remove). - Isolation Forest: this algorithm is based on tree algorithms it calculate an anomalie score for each point, the easier it is to isolate the data, the more likely it is that the data is an anomaly. After performance comparisons we kept the Isolation Forest to remove the outliers. Remove outliers is one of our most important techniques of preprocessing to reduce our mean absolute error.

Model developement

Baseline results on several algorithms

As stated before, our main goal is finding the optimum predictive power. To do so, we tried several algorithms on the processed data, i.e. following the treatment described before.

We report the scores obtained by the following models: Linear regression, Random Forest, Lasso regression and Elastic regression. All data sets are scored, that is, the train, validation and test sets. The models where not *fine tuned* because we were mostly interested in defining baselines score to improve upon.

In addition, we performed k-fold (k=10) cross validation for each algorithm to improve our training *mae* estimate. The results of such cross validation strategy are shown in Figure X:

```
autoplot(fit_rs)
```

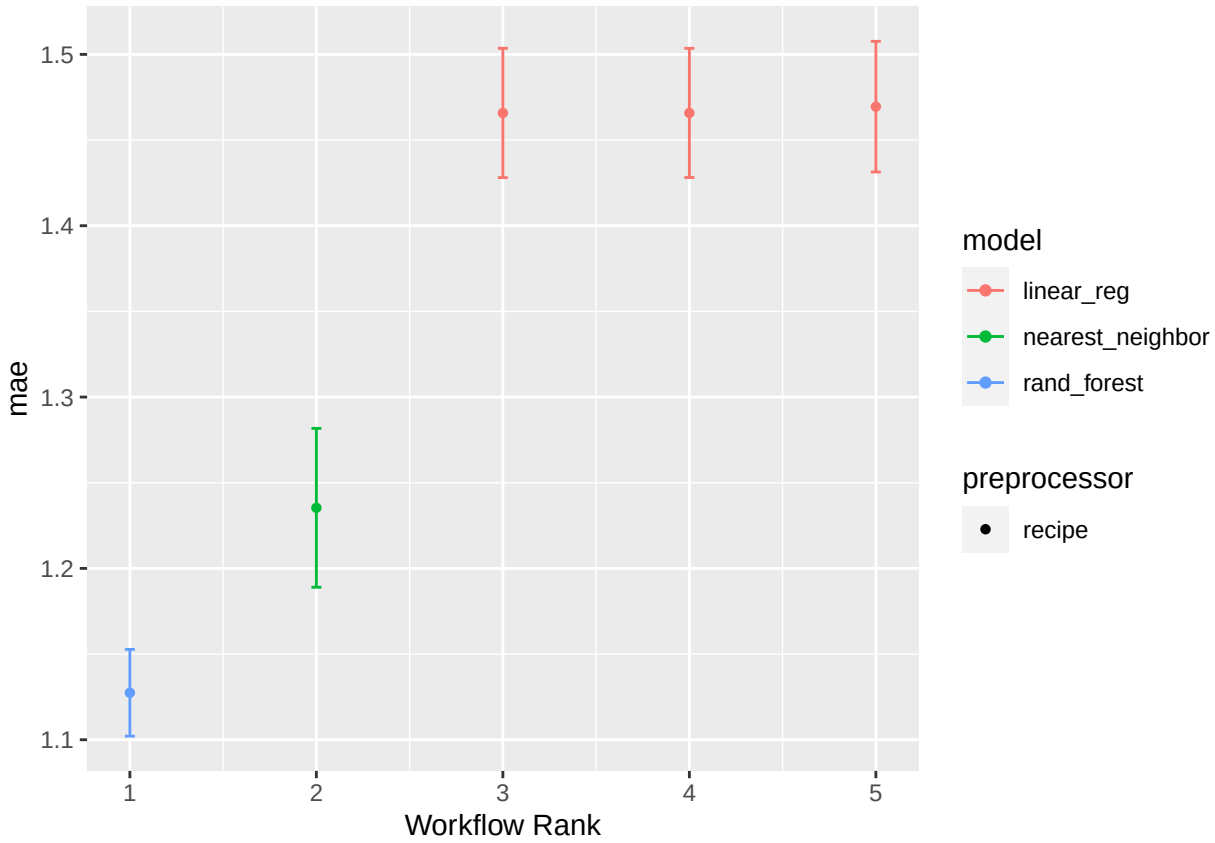


Table X shows all mean absolute error scores on all data sets from all models.

```
print(results)
```

```
## # A tibble: 3 x 6
##   data       lm    rf lasso elastic   knn
##   <chr>   <dbl> <dbl> <dbl>   <dbl> <dbl>
## 1 X_train  1.46  1.02  1.46    1.46  1.01
## 2 X_val    1.53  1.12  1.53    1.53  1.44
## 3 X_test   1.55  1.24  1.55    1.55  1.47
```

Results on best tuned GBRT

Feature importances

Conclusion