Age estimation regression

Marco Galano, Davide Marzano
Politecnico di Torino
Student id: s338658, s344814
s338658@studenti.polito.it
s344814@studenti.polito.it

***Abstract*—In this report we aim to show a possible approach for the *Age Estimation* regression task. In particular, our solution consists of both using the features provided with the dataset and extracting some statistics from the spectrogram of each audio file. The submissions of the prediction of each model are evaluated in terms of root mean square error (RMSE). Our model has been put in competition with other models and obtained overall good results, outperforming the naive baseline.**

I. Problem overview

The assigned task consists in building a regression pipeline aimed at estimating the age of each speaker, given a list of audio recordings. In addition to the audio files, some features related to each of them were provided. The dataset is divided as follows:

- a *Development Set*, consisting of 2933 recordings, with a target value "Age";
- an *Evaluation Set*, consisting of 691 recordings, for which we try to estimate the value "Age".

An immediate consideration that can be done is that the development set is heavily unbalanced: 1980 samples out of 2933 are in the range of age between 18 and 34 years: those samples account for $\approx 67.5\%$ of the development set. In Figure 1 it is shown the distribution of ages for each group. Another observation regarding both the provided sets can be made: *430* records of the evaluation set have an ethnicity that is not present in the development set, which is used to train the model.

We can examine audio recordings looking at both the time or frequency domains, and this information is well represented by the features provided in the dataset. However, while there are various different durations between all the recordings (shown in Figure 2), every audio is sampled with the same sampling rate, that is 22.05 kHz. This suggests that we could extract important information from the spectrogram of each audio signal. For example, a common practice in the task of age estimation is to extract the *Mel Frequency Cepstral Coefficient* (*MFCC*) [1].

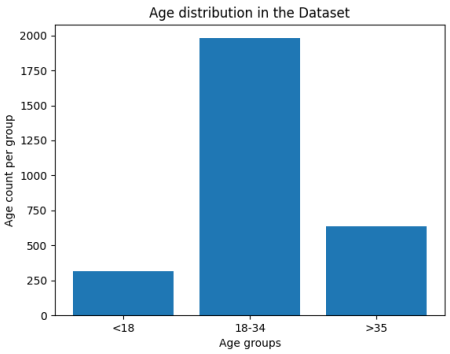

Fig. 1: Age distribution per groups

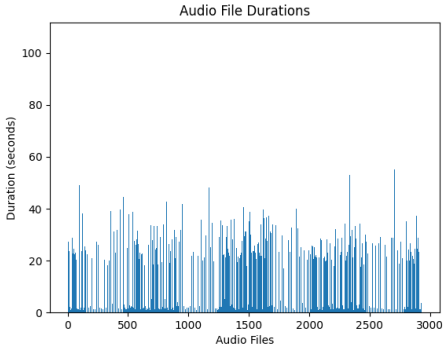

Fig. 2: Audio files durations in seconds

II. Proposed approach

A. Preprocessing

Following an initial inspection of the dataset, we could see that there were no missing values to handle. The first approach adopted was to leverage our domain knowledge to analyze the features. We observed that some of them could be redundant; in particular, we expected these two groups of features to carry very similar information:

- NUM WORDS, NUM CHARACTERS: the number of words and characters in the spoken sentence;
- MEAN PITCH, MAX PITCH, MIN PITCH: mean, maximum, and minimum pitch of the speech signal, in Hz;

for example, we expected NUM WORDS and NUM CHARACTERS to have a strong correlation, because a higher number of

words is very likely to lead to a higher number of characters. This assumption was confirmed by computing the Pearson correlation, that returned a correlation of *0.99989*. The reasoning applied was analogous for MEAN PITCH, MAX PITCH, MIN PITCH. In order to reduce dimensionality, we have decided to keep only NUM WORDS and MEAN PITCH, and exclude the others.

Furthermore, we noticed that SAMPLING RATE and PATH do not carry useful informations and are therefore irrelevant for our regression purposes, so we removed them from the dataframe.

Another issue we had to address was the handling of two categorical features, GENDER and ETHNICITY:

- For the GENDER column, we identified an incorrect value, that is 'famale'; we considered it a spelling error and converted it to correct value 'female' and then we used *one-hot* encoding, that generated two columns as a result, *gender_male* and *gender_female*.
- Regarding the ETHNICITY feature, it requires further elaboration for better clarity. About the ethnicity, one might argue that different ethnicities might reflect distinct vocal timbres and languages, thereby offering additional context alongside the other features. However, other factors need to be taken into account: firstly, as it was introduced in the problem overview, only around 37.8% of the evaluation set contains records whose ethnicity is also present in the development set; secondly, since it is a categorical feature with a high cardinality of distinct values, it is important to consider the consequent dimensionality increase given by the encoding of the ethnicities. On account of these considerations, we decided to discard this feature.

This analysis had a clear benefit: reducing the dimensionality and multicollinearity in order to improve the performances of the regression models.

Another observation made was that TEMPO, while it should have been represented as a floating point value, was actually written as a string in the format *'[999.9999]'*; therefore, it was stripped of the square brackets and converted to float.

Finally, as it was previously introduced, we decided to extract some meaningful features from the audio files. We computed the *MFCC* for each recording, resulting in a matrix of coefficients for each audio file with n rows, where n is a parameter to be tuned based on the performances of our models. We represented these coefficients through the mean value of the matrix rows. By doing so, our dimensionality increases by n, but the just computed coefficients carry enough meaningful information to justify the augmentation. We also considered performing a dimensionality reduction for one of our tested models using *PCA*, that is properly examined in the following subsections after the discussion of the number of *MFCC* computed.

B. Model selection

In order to perform this task, we opted to use the following two regressors:

- *RandomForestRegressor*: we decided to use this algorithm as it performs quite well with an unnormalized dataset, meaning that it can run with the selected features without further elaboration. Moreover, a reason behind this choice is that it does not struggle with a large number of dimensions due to its implementation;
- *MLPRegressor*: this model is a regression model built on the multi-layer perceptron (MLP) architecture, which uses a feedforward neural network to predict continuous target values. We have chosen it because of its wide variety of customizable parameters, that make it suitable for our application and improve the overall effectiveness. Additionally, it has been shown to perform well with MFCC in [2].

More precisely, *MLPRegressor* is part of a pipeline, in which it is preceded by additional steps used for dimensionality reduction. Let us observe the pipeline in detail:

- *StandardScaler*: A scaling method that centers data around a mean value and normalizes it by a common standard deviation. This standardization is crucial to perform dimensionality reduction through PCA later;
- *PCA*: a dimensionality reduction method, that takes into account both statistical and algebraic characteristics of the dataset, and reduces the number of total features by a linear combination of the original ones [3];
- *MLPRegressor*: it is used to predict the target value based on the features transformed in the previous steps of the pipeline.

For both our approaches, a grid search with cross validation has been used to find the best set of hyperparameters.

C. Hyperparameters tuning

We need to determine the hyperparameters for:

- *MFCC*
- *RandomForestRegressor*
- *MLPregressor*

To decide what would be the optimal number of *MFCC*, we tested some default RandomForestRegressor and MLPregressor without any additional tuning or preprocessing.

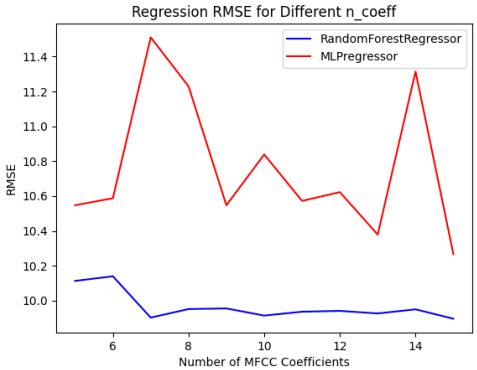

Fig. 3: Regression with a variable number of mean *MFCC*

We can already see that while RandomForestRegressor performs quite well with the provided dataset, MLPregressor requires some additional preprocessing. In fact, once we decided how many additional features we would add to our dataset, we performed dimensionality reduction through PCA for the MLPregressor; to identify the ideal number of Principal Components (PCs) to be computed, we analyzed the graph of the cumulative explained variance, setting a threshold of the 90% of retained explained variance.

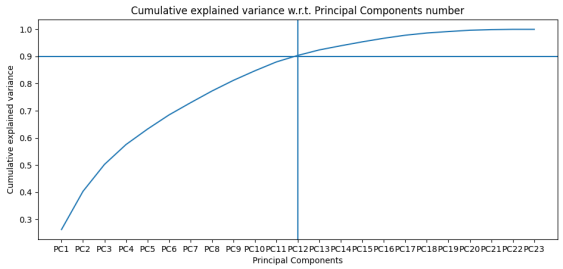

Fig. 4: Cumulative explained variance

Finally, we used two grid searches in order to find the best sets of parameters in terms of performances, one for Random Forest and one for MLPregressor. The grid search has been run with the hyperparameters defined in the table I.

Model	Parameters	Values
MFCC	n	$5 \rightarrow 15$
Random Forest	n_estimators	{100, 250, 500, 650}
	criterion	{squared_error, poisson, friedman_mse, absolute_error}
	max_features	{sqrt, log2}
	min_impurity_decrease	{0, 0.01, 0.001}
	min_samples_leaf	{1, 5, 10}
	random_state	42
MLP	hidden_layer_sizes	{(100,), (100,50) (150,100,50)}
	activation	{relu, identity, logistic, tanh}
	solver	{adam, sgd}
	learning_rate	{constant, invscaling, adaptive}
	random_state	42
	max_iter	7000

Table I: Hyperparameters tested

III. Results

From an inspection of the results shown in figure 3, we chose to compute 10 mean *MFCC* as it seems the best compromise to have good performances without increasing too much the dimensionality of our model. Then, as we can see in figure 4, we succeed in retaining approximately 90% of explained variance by computing 12 PCs, reducing the total number of features almost by half.

By the results of the two grid searches performed, we identified the best configurations for both our models; these were:

- *RandomForestRegressor*: {n_estimators: 650, criterion: 'poisson', max_features: 'sqrt', min_impurity_decrease: 0.001, min_samples_leaf: 5}
- *MLPregressor*: {hidden_layer_sizes: (150,100,50), activation: 'relu', solver: 'sgd', learning_rate: 'invscaling'}

The results of the GridSearch were evaluated with respect to the mean RMSE of the cross validation for each configuration. The best result for RandomForestRegressor in the GridSearch was $RMSE \approx 10.48$, while for MLPregressor it was $RMSE \approx 10.65$, both below the threshold of the naive baseline. We trained both models on the whole development set and then used the models to make a regression on the evaluation set, obtaining in the public leaderboard the following scores[1]:

- *RandomForestRegressor*: 9.966;
- *MLPregressor*: 9.886.

We assume that similar scores could be achieved also in the private leaderboard.

IV. Discussion

As it was discussed in the previous section, the results obtained by our models comfortably outperformed the naive baseline indicated on the public leaderboard in terms of RMSE score.

However, although the results obtained were definitely positive, it is still important to identify the improvements that can be applied to the proposed solution:

- In the problem overview we introduced the issue of an unbalanced dataset with respect to the target value. A potential improvement might be achieved by emphasizing the importance of certain underrepresented values and by generating synthetic data; this approach for regression is discussed in [4].
- Other feature representations or regression models that have not been taken into consideration could achieve better results than the ones obtained.

We find the results satisfactory, but as analyzed here there is always space for refinements that can lead to a better model.

References

[1] Z. K. Abdul and A. K. Al-Talabani, "Mel frequency cepstral coefficient and its applications: A review," *IEEE Access*, vol. 10, pp. 122136–122158, 2022.

[2] Z. Tüske, C. Plahl, and R. Schlüter, "A study on speaker normalized mlp features in lvcsr," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[3] S. Karamizadeh, S. M. Abdullah, A. A. Manaf, M. Zamani, and A. Hooman, "An overview of principal component analysis," *Journal of signal and information processing*, vol. 4, no. 3, pp. 173–175, 2013.

[4] P. Branco, L. Torgo, and R. P. Ribeiro, "Smogn: a pre-processing approach for imbalanced regression," in *First international workshop on learning with imbalanced domains: Theory and applications*, pp. 36–50, PMLR, 2017.

[1]The results were obtained from the submissions of the student s338658.