Data science lab: process and methods Winter project

Gaudino Andrea, Grivet Talocia Lorenzo
Politecnico di Torino
Student id: s346119, s346559
s346119@studenti.polito.it, s346559@studenti.polito.it

***Abstract*—In this report we introduce a possible approach to the age prediction from a speaking voice. The approach is based on the feature extraction from the spectrogram of the recordings. The algorithm preprocess the data and trains a linear regressor to compute the prediction of the age.**

I. Problem overview

The goal of the project is to estimate the age of a speaker from the audio recording of their voice. In order to predict the outcome we analyzed the acoustic properties of the audio files and inspected their correlation to the age of the speaker.

Our data set is composed of two sections:

- *Development set*, which includes the age of the speakers along with the features extracted from the recordings. It contains 2933 audio files.
- *Evaluation set*, which contains only the audio features and will be used to perform our prediction. It comprises 691 records.

We can make some considerations by looking at the data distribution in the development set. In particular, observing the Figure 1, concerning our target variable, we can clearly see that its distribution is primarily concentrated within a range of values between twenty and thirty. As the age increases, there are progressively fewer data points, therefore the error of our model could increase for these values.

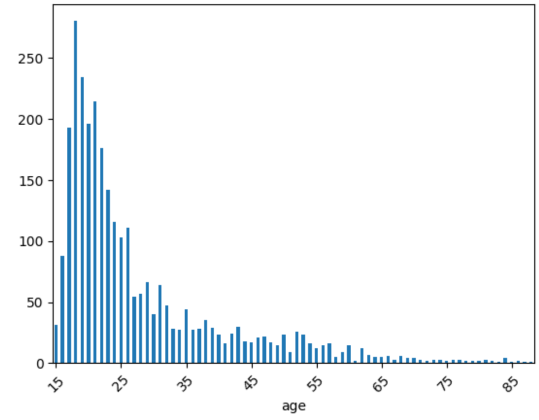

Fig. 1. Age distribution of the development set

II. Proposed approach

The model we were asked to employ is based on regression analysis.

A. Preprocessing

For our regressor to function effectively, a careful preprocessing phase is required. Firstly, we focused on the detection of missing values that could have prevented the execution of the regressor model. In our case the dataset does not contain any null value, therefore we did not need to discard any of the recordings.

As discussed in section I, the edge age values contain few data points, and their presence could negatively impact the effectiveness of our model. Therefore, we decided to remove these recordings from the dataset.

The regressor can only work with numerical values, thus we looked for categorical data and we found out that two features (*ethnicity* and *gender*) contained strings of characters. In order to overcome this setback, we applied the *One-Hot encoding* method on these attributes.

Regarding the *ethnicity* feature, we observed the following issue: the distribution of ethnicities differs between the development and evaluation datasets. In general, we think that potential inconsistencies between sets could lead the model to learning patterns that do not generalize well, causing a potential mismatch in predictive accuracy and a biased model performance. Given that, other than the fact that the number of different ethnicities is huge, we thought it could be better to remove this feature.

The features contained in the original datasets were: *sampling rate, gender, ethnicity, min pitch, max pitch, mean pitch, jitter, shimmer, energy, zcr mean, spectral centroid mean, tempo, hnr, num words, num characters, num pauses, silence duration*. We decided to consider additional features related to the acoustic properties, using the Python *librosa* package. Specifically, we included MFCC (Mel-Frequency Cepstral Coefficients) and Mel spectrogram features.

The Mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear Mel scale of frequency. MFCCs are coefficients that collectively make up the Mel-Frequency Cepstrum. We took into account the mean and the standard deviation of each coefficient to summarize their behavior over time.

The Mel spectrogram, meanwhile, is a time-frequency representation of the audio signal, where the frequency axis is mapped to the Mel scale [1]. Both features are derived from the Mel scale, which approximates how the human ear discriminates frequencies by emphasizing lower frequencies and compressing higher ones:

$$mel(f) = \begin{cases} f & \text{se } f \leq 1kHz \\ 2595 \cdot \log\left(1 + \frac{f}{700}\right) & \text{se } f > 1kHz \end{cases}$$

In addition, we also added the spectral rolloff and spectral bandwidth. Spectral rolloff [2] is defined as the frequency below which a certain percentage of the total energy of the spectrum is contained (we used the default value of 85%). Spectral bandwidth represents the width of the frequency band for each frame in each recording.

Finally, we took into account the *chroma* feature based on the *Short-Time Fourier Transformation*. It provides a spectral representation that captures intensity patterns over time in an audio signal. It focuses on the distribution of energy across different frequency bands, which can be interpreted as a form of harmonic content. By using the Short-Time Fourier Transform (STFT), it analyzes the signal in small time windows and computes the intensity of various frequency bands within each window.

These newly selected features help the model better understand the acoustic properties of speech, improving its ability to estimate the age of the speaker.

We decided to remove some feature in order to reduce dimensionality of our datasets:

- *Id*: it has no correlation at all with the age of the speaker.
- *Sampling rate*: since the sampling rate is identical across all recordings, it does not contribute any valuable insights.
- *Min and max pitch*: we already include the *mean pitch* feature, which captures the relevant information, so the min and max values were deemed unnecessary.

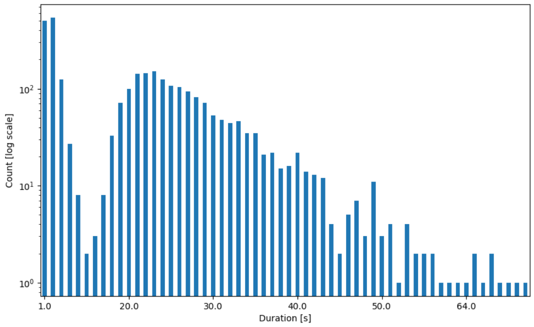

Fig. 2. Audio files duration

The graph in Figure 1 shows a wide range of durations, which suggests significant variability in the audio data. This variability could indicate that the audio files differ considerably in length. In this case, using a variety of features (such as spectral, temporal, or statistical characteristics) could help capture the characteristics in the data more effectively and improve the model's ability to classify the audio files accurately.

Finally, we decided to detect and eliminate outliers by setting a threshold for each feature:

$$t = \mu \pm 3 \cdot \sigma$$

and removed any recordings that had multiple features outside of these limits from the dataset.

A fundamental part of the preprocessing we made concerns the scaling of the data. Scaling ensures that all features contribute equally to the analysis by standardizing their ranges or distributions. We decided to inspect two different types of standardization: the first one based on the computation of the columns' norm of the train dataset (from now on, we will refer as ”column norm” standardization for simplicity) and for the second one we applied a standard scaling.

After scaling the data, the next step is to apply PCA (*Principal Component Analysis*) in order to reduce the dimensionality of the dataset while retaining most of the meaningful information. This process helps simplify the analysis, improving computational efficiency and making the data easier to interpret without sacrificing the quality of the information. PCA transforms the original features into a new set of uncorrelated components, effectively eliminating any correlations between the features. This is achieved by finding the directions along which the data varies the most, then it projects the data onto these new axes. Removing correlated feature could improve the performance of the regressor model and simplify data analysis by reducing redundancy.

B. Model selection

We chose to base our algorithm on Ridge Regression, a method that incorporates a regularization term to the model. This approach is particularly useful for addressing two common challenges in regression models:

- Preventing overfitting: Ridge Regression penalizes large regression coefficients, preventing the model from fitting the training data too closely. This allows the model to generalize better to unseen data.
- Handling multicollinearity: when independent variables are highly correlated, multicollinearity can make the regression coefficients unstable in standard linear regression models. Ridge Regression mitigates this issue by shrinking the coefficients, reducing their variance, and improving the stability and robustness of the model.

In summary, Ridge Regression not only enhances model stability when dealing with correlated features but also ensures better predictive performance on new data by striking a balance between bias and variance through regularization.

C. Hyperparameters tuning

The principal issue of ridge regression is the choice of the appropriate value for the regularization parameter *alpha*. With the aim of inspecting the best value of α for our model, we conducted several trials. The graph in Figure 3 shows the impact of the regularization parameter α on *RMSE* in

Ridge Regression using the two different data normalization methods: Column norm (in blue) and Standardization (in orange). As the value of α increases, the *RMSE* rises for both methods, but the Column norm method deteriorates more significantly. Standardization achieves lower and more stable *RMSE*, suggesting it is less sensitive to large regularization values. The optimal α lies in the lower range, where *RMSE* is minimized before increasing due to excessive regularization.

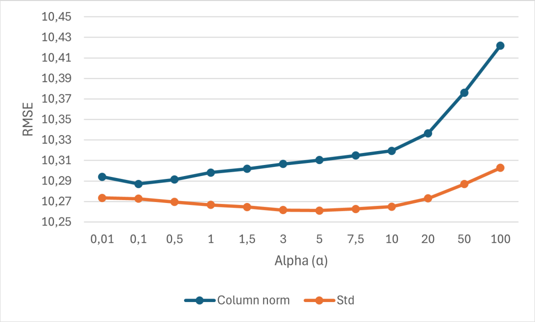

Fig. 3. Performance of the model as α changes

Another hyperparameter to be aware of is *fit_intercept*, that determines whether to fit the intercept of the model. This parameter acquires crucial importance especially having scaled the data. The prediction $\tilde{y}_i$ will be a linear function of the input vector $\bar{x}_i$, therefore the values of $\tilde{y}$, which do not have zero mean, will deviate significantly from the ground truth. To solve this problem, we set the parameter *fit_intercept* to automatically estimate an intercept b_0 to adjust the function.

III. Results

The best results were achieved employing PCA, suggesting that it is effective in reducing correlations between variables while disregarding less important ones. Additionally, a low value of α in a Ridge regression model implies a weaker penalization of the coefficients, allowing the model to better fit the data, particularly when the data is not noisy. A low α, in fact, reduces the introduction of bias in the model. It is interesting to observe, in Figure 4, how the α parameter varies as a function of the number of principal components selected during PCA.

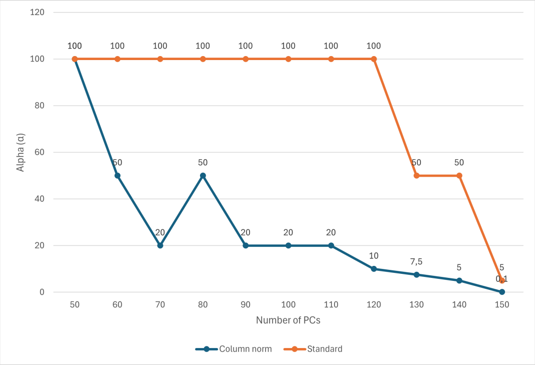

Fig. 4. Value of α as the number of PC increases

As we can detect from the graph the value of α decreases rapidly for the highest values of PCs, therefore the best performing model is trained on a dataset with a large number of principal components. In Figure 5 we display the magnitude of the weights for each feature in the PC space. As expected, the first weights are significantly lower than the others because their corresponding PCs contain most of the information, hence the less important PCs require higher regularization as they represent noise data.

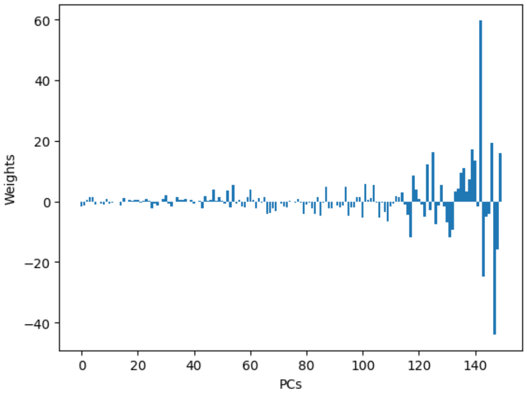

Fig. 5. Best model weights

The score we obtained, respectively for the column norm and standardized datasets, are 10.70 and 10.68. The public score we obtained are of 9.277 and 9.264, significantly lower than the one obtained on development data.

IV. Discussion

The difference observed between the two scores, private and public, denote the presence of overfitting of our model that has not been resolved by the prepocessing steps and PCA analysis. However, the low public score obtained implies that the algorithm is still returning quite a precise prediction.

References

[1] J. Gowdy, “Mel-scaled discrete wavelet coefficients for speech recognition,” *IEEE*, 2000.

[2] F. Eyben, *Real-time Speech and Music Classification by Large Audio Feature Space Extraction*. Springer, 2016.