CS4622 - Machine Learning
Lab 01 - Feature Engineering

Thevin Senath

190583V

1. Introduction

The primary objective of the lab involves utilizing techniques for feature selection and engineering to decrease the number of features in a provided dataset, all while maintaining the accuracy of the model. The dataset includes two CSV files named "train.csv" and "valid.csv," each containing 256 features and 4 target labels. Additionally, there's a "test.csv" file for evaluating the accuracy of the feature-reduced model.

The initial 256 columns in the dataset hold speaker embedding vectors created using the "wav2vec-base" model. The last 4 columns represent labels linked with these embeddings:

- Label 1: Speaker ID
- Label 2: Speaker age
- Label 3: Speaker gender
- Label 4: Speaker accent

The process of feature reduction is performed separately for each label. Although the general process is similar for each label, some adjustments are necessary due to differing behaviors among labels. The first step involves splitting the dataset into features and labels.

For each label, after reducing the features, the accuracy of the model is tested. A suitable classification model is used for Speaker ID, Gender, and Accent since they are categorical variables. As for the age, a regression model is employed because it's a numerical variable. The goal remains to ensure accuracy while effectively reducing the number of features and maintaining the model's performance.

2. Preprocessing

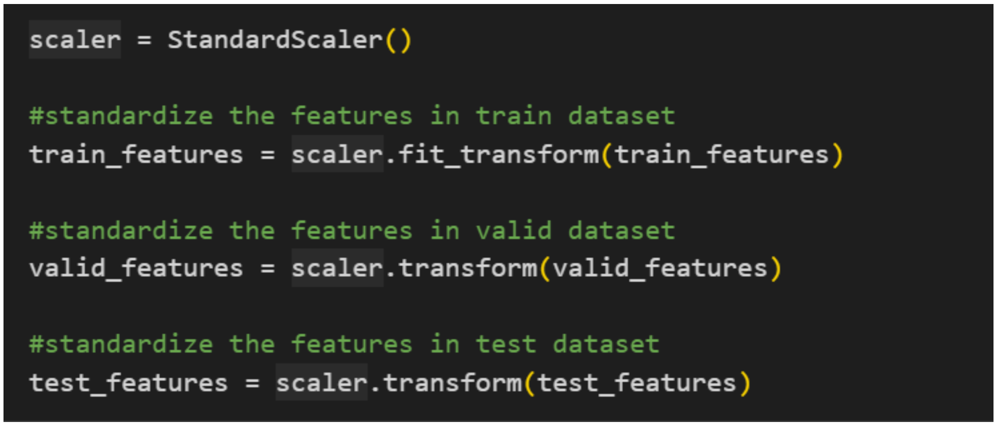
- Scaling

Scaling is a crucial preprocessing step aimed at normalizing the features within your dataset. Its purpose is to ensure that all features share similar scales and ranges. This normalization is vital for the proper functioning of many machine learning algorithms. When features have varying scales, certain algorithms may assign greater importance to features with larger values, potentially leading to biased outcomes.

One specific method of scaling is standard scaling, also referred to as Z-score normalization. This approach involves subtracting the meaning of each feature from its values and then dividing it by the standard deviation of that feature. As a result, the transformed features will have a mean of 0 and a standard deviation of 1. In the context of this lab, the feature set is normalized using a standard scaler.

Before applying feature engineering techniques, scaling the dataset is crucial. This is to ensure that variables with different scales don't unduly influence the analysis. In situations where variables have significantly different scales, those with larger scales could dominate the outcomes, leading to a biased representation of the data's variability.

Moreover, scaling guarantees consistent treatment of variables and mitigates potential problems related to numerical precision. By scaling the data, feature engineering



```
scaler = StandardScaler()

#standardize the features in train dataset
train_features = scaler.fit_transform(train_features)

#standardize the features in valid dataset
valid_features = scaler.transform(valid_features)

#standardize the features in test dataset
test_features = scaler.transform(test_features)
```

techniques able to capture the true relationships among variables, enabling more meaningful interpretation of the results.

3. Feature Selection

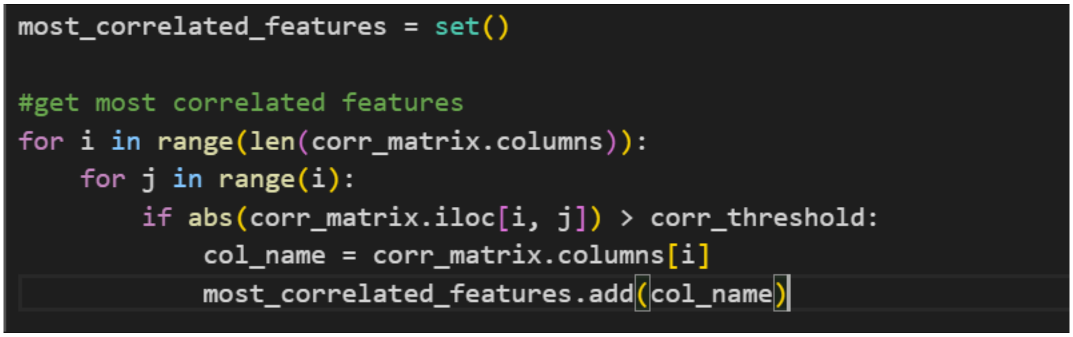
- Correlation Analysis

Correlation serves as a metric that quantifies the strength of a linear relationship between two or more variables. A strong correlation indicates the potential ability to predict one variable based on another, emphasizing a linear connection between them. When two attributes exhibit significant correlation, including both in a model might not provide substantial benefits. Moreover, it's important for attributes to demonstrate correlation with the target variable.

Assessing correlation is essential for understanding complex relationships within datasets. In cases where two attributes are highly correlated, it implies that changes in one attribute are likely to correspond with predictable changes in the other, along a linear pattern. However, correlation doesn't imply a causal relationship; it solely highlights the degree of linear association.

In this lab, the process of reducing features involves analyzing correlations between features and between features and labels. To accomplish this, a correlation matrix is employed.

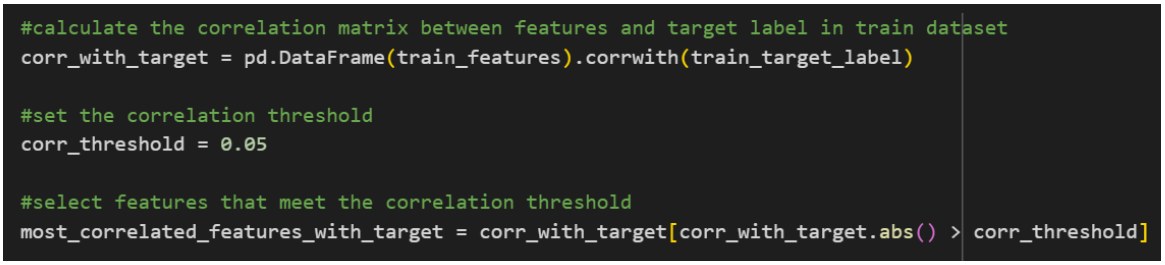
Initially correlation between the features were analyzed. After analyzing the correlation matrix, 0.9 was selected as the threshold. If a correlation between two features is greater



```
most_correlated_features = set()

#get most correlated features
for i in range(len(corr_matrix.columns)):
    for j in range(i):
        if abs(corr_matrix.iloc[i, j]) > corr_threshold:
            col_name = corr_matrix.columns[i]
            most_correlated_features.add(col_name)
```

than the threshold, it can be determined that both the features have the same characteristics so that one of them can be removed from the dataset.

Then the features with low correlations to a specific label are removed. The correlation matrix is examined to guide this process. A threshold of 0.05 is chosen based on the analysis of the correlation matrix. If the correlation between a feature and a label is below this threshold, it suggests that the feature has minimal impact on the corresponding label.



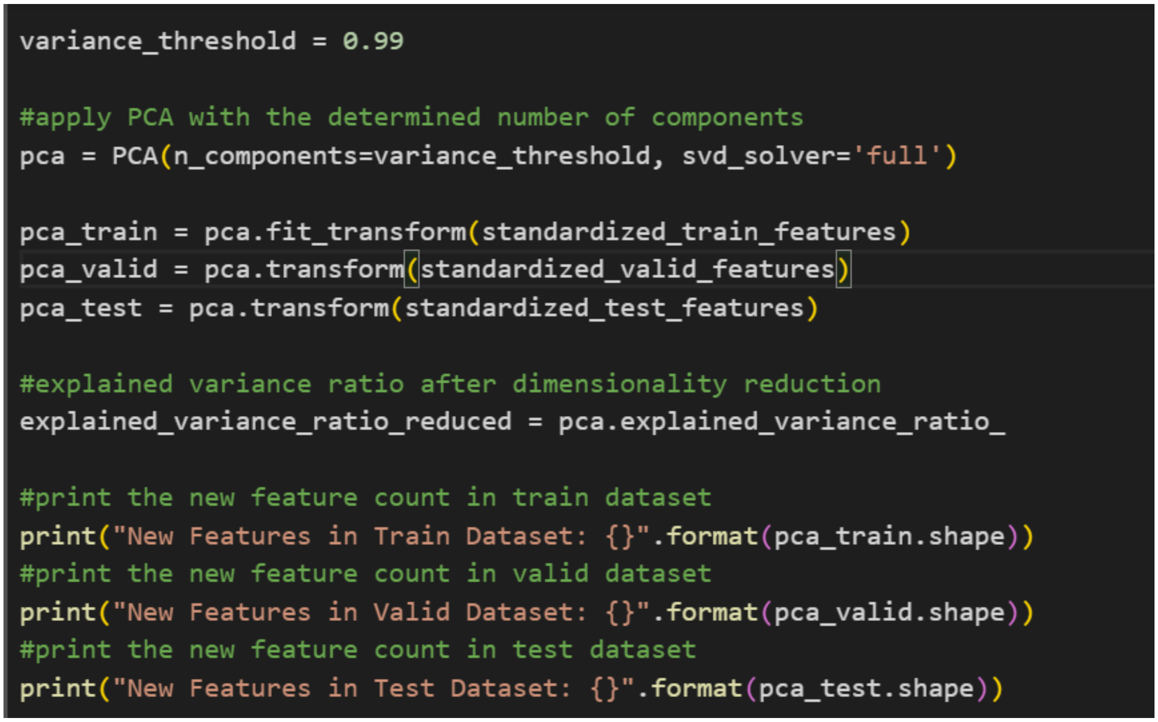
```
#calculate the correlation matrix between features and target label in train dataset
corr_with_target = pd.DataFrame(train_features).corrwith(train_target_label)

#set the correlation threshold
corr_threshold = 0.05

#select features that meet the correlation threshold
most_correlated_features_with_target = corr_with_target[corr_with_target.abs() > corr_threshold]
```

4. Feature Extraction

- Principal Component Analysis(PCA)

Principal Component Analysis (PCA) is a dimensionality reduction technique used in data analysis and machine learning. By identifying new uncorrelated axes, called principal components, PCA transforms high-dimensional data into a lower-dimensional representation while retaining maximum original variability. It begins with calculating the covariance matrix and then performing eigenvalue decomposition to find the eigenvectors and eigenvalues. These are sorted, and the top ones are chosen as principal components. This process reduces dimensionality by capturing the most important information. The number of principal components selected balances dimensionality reduction and preserving data variance.

In this lab, the PCA had been applied for feature extraction.

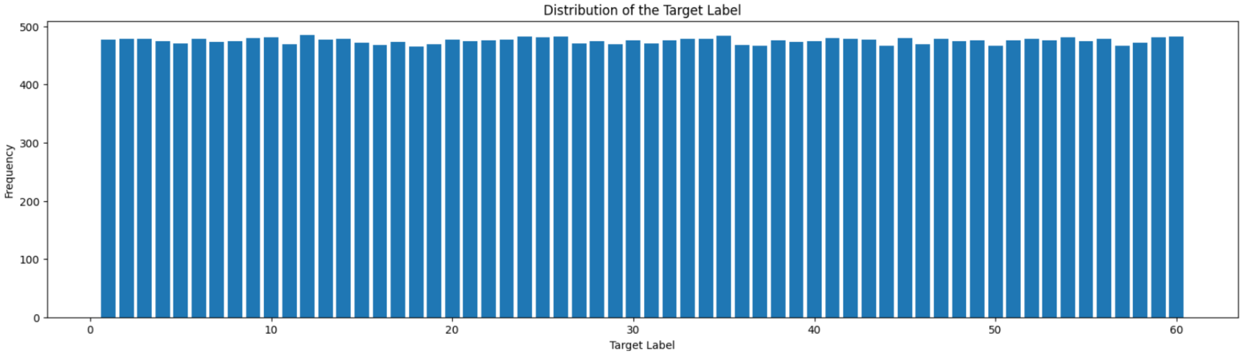


```
variance_threshold = 0.99

#apply PCA with the determined number of components
pca = PCA(n_components=variance_threshold, svd_solver='full')

pca_train = pca.fit_transform(standardized_train_features)
pca_valid = pca.transform(standardized_valid_features)
pca_test = pca.transform(standardized_test_features)

#explained variance ratio after dimensionality reduction
explained_variance_ratio_reduced = pca.explained_variance_ratio_

#print the new feature count in train dataset
print("New Features in Train Dataset: {}".format(pca_train.shape))
#print the new feature count in valid dataset
print("New Features in Valid Dataset: {}".format(pca_valid.shape))
#print the new feature count in test dataset
print("New Features in Test Dataset: {}".format(pca_test.shape))
```

5. Labels

- Label 1- Speaker ID

The distribution of the dataset after preprocessing can be represented as follows.



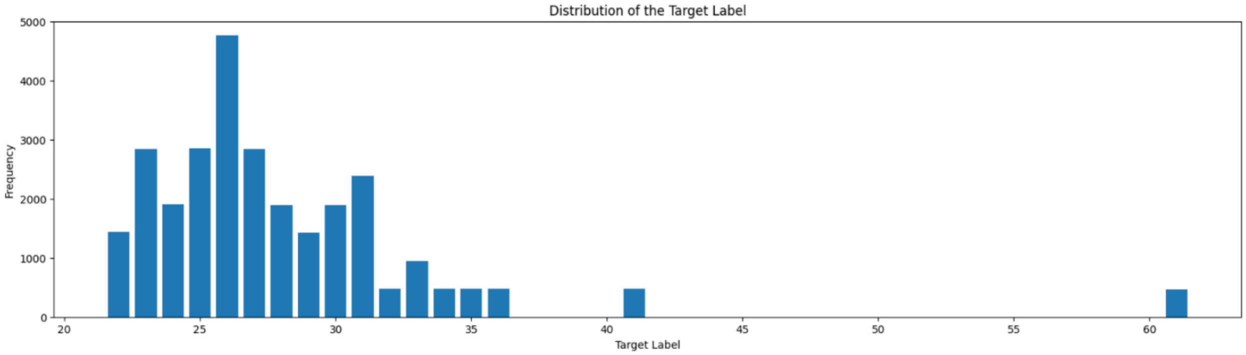
According to the plot, it was identified that the Speaker ID is a categorical variable which has 60 cluster values.

Three classifiers named Support Vector Classifier(SVC), Random Forest Classifier and K-Nearest Neighbors(KNN) had been used and compared to check the accuracy. According to the results the SVC classifier gave better predictions.

- Accuracy Score for valid dataset without feature engineering was 0.991.
- Accuracy Score for valid dataset with feature engineering was 0.984.
- The feature count was reduced to 97.

Label 2- Age of the speaker

The distribution of the dataset after preprocessing can be represented as follows.

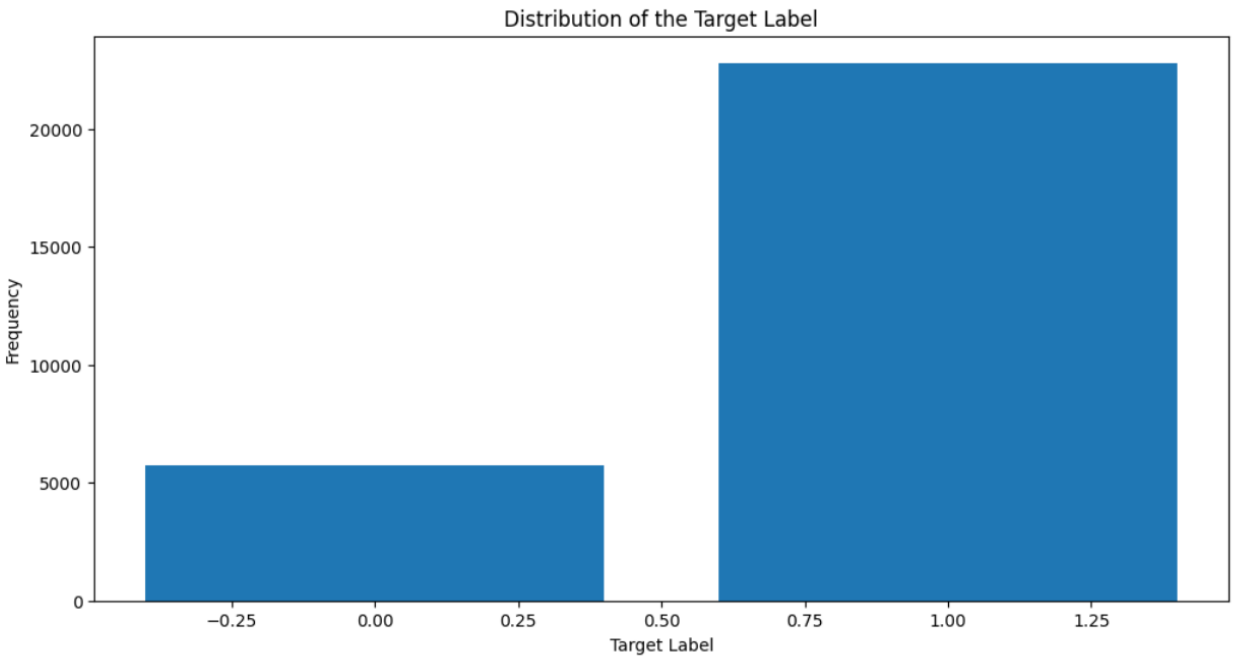


Since age is a numerical variable, instead of a classifier, a regressor is used as the model to train and predict. KNeighbors Regressor is used as the regression model. In here instead of the accuracy*,* R2 Score value is taken as the evaluation metric. When analyzing the dataset, it was found that there are 480 missing values in the label values. Since there is a lot of data, those rows have been dropped from the dataset.

- R2 Score Score for valid dataset without feature engineering was 0.984.
- R2 Score Score for valid dataset with feature engineering was 0.987.
- The feature count was reduced to 84.

Label 3- Gender of the speaker

The distribution of the dataset after preprocessing can be represented as follows.

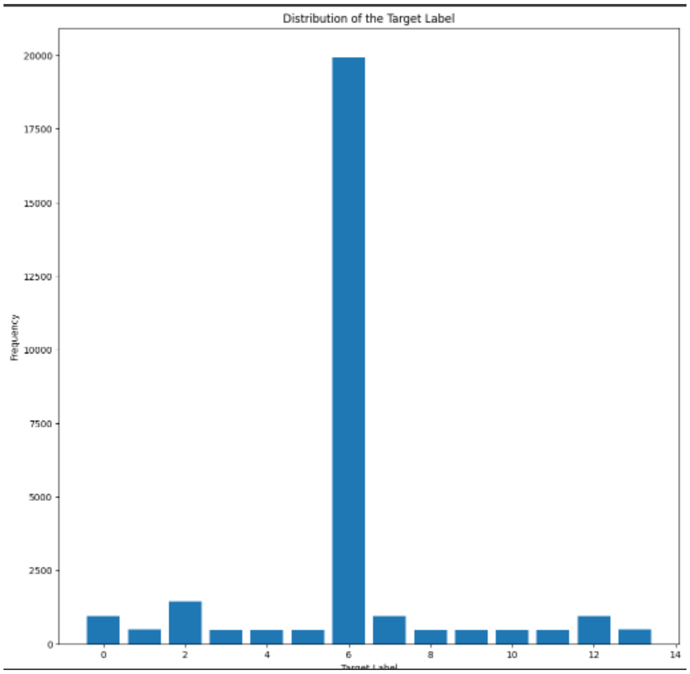


Categorizing in gender can be considered as a binary classification problem where 1s and 0s are used to identify males and females. The same three classifiers that were used check the accuracy of the label 1 had been used here and the SVC classifier gave better predictions too.

- Accuracy Score for valid dataset without feature engineering was 0.999.
- Accuracy Score for valid dataset with feature engineering was 1.0.
- The feature count was reduced to 100.

- Label 4- Accent of the speaker

The distribution of the dataset after preprocessing can be represented as follows.



Here also the same three classifiers that were used check the accuracy of the label 1 had been used. But the KNeighbors classifier gave better predictions too.

- Accuracy Score for valid dataset without feature engineering was 0.995.
- Accuracy Score for valid dataset with feature engineering was 0.991.
- The feature count was reduced to 80.

References

1. https://colab.research.google.com/drive/17xsyxB7vXjXOX2sRXG6GfsUJ1jWMKbzN#scrollTo=a289Ye21LbR0
2. https://colab.research.google.com/drive/1beTfl6P-YnyVy8hdyaohhXi4jCRN_hug#scrollTo=4p51o92-Y9jm
3. https://colab.research.google.com/drive/1m9m_mr01layjyPQ_Xu7yFuTWKcQX-pQQ#scrollTo=F5P90hRcOo8Z
4. https://colab.research.google.com/drive/1giml11-vipYQ139fBu3CePf4oiOd3yf4#scrollTo=pxFmK735ZQtQ