Statistical Learning Theory Final Project Report

ML meets MD: Diagnosing Coughs and Wounds by Sound

Medical Symptoms Classification

Author(s): Sahar Ziv
Student ID(s): 312160369
Course Name: Statistical Learning Theory
Submission Date: June 23, 2024

Abstract

This project explores the use of machine learning to classify medical symptoms, specifically coughs and infected wounds, from self-reported audio recordings. Utilizing a dataset of 6661 audio recordings with associated quality measures and transcriptions, the project addresses challenges such as incorrect labels and poor audio quality through thorough data cleaning. Various models, including logistic regression, KNN, SVM, and classification trees, were evaluated using features extracted from audio files and transcriptions. Key findings indicate that NLP models based on audio transcriptions outperform audio-only models, achieving perfect accuracy in binary classification. However, clustering by accent and modeling audio features also yielded promising results. The project concludes that combining speech-to-text and NLP techniques is the most effective approach for symptom classification, with potential implications for enhancing conversational agents in healthcare.

1 Introduction

This project focuses on using machine learning techniques in order to classify medical symptoms based on self-reported audio recordings, with the goal of improving conversational agents in the medical field. The dataset comprises thousands of audio snippets, totaling over 8 hours of recording time, these recordings were created through a multi-step process where contributors first provided textual descriptions of symptoms, followed by recording corresponding audio. However, challenges such as incorrect labels and poor audio quality necessitate thorough data cleaning before model training. By addressing these issues, the project aims to create robust models that can accurately classify medical symptoms, thus enhancing the efficacy of conversational agents and improving healthcare diagnosis.

The research questions are:

- Can we classify the symptom of the patient to cough or infected wound only by the audio file?
- Which model classify better, audio file classification model or NLP model of the audio files transcriptions?

2 Data Description

The data found on Kaggle, the data was labeled using the Figure-Eight platform. The dataset contains 6661 audio recordings and a corresponding CSV file, where each recording has 3 indicators of audio quality measures (audio clipping, background noise audible and quiet speaker) with confidence score for those indications (between 0 and 1), an overall numeric score between 3 and 5 of the quality of the audio which is uncorrelated with the other audio quality measures. Furthermore, for each audio recording the CSV file contains the transcription of the audio, the writer ID, the speaker ID, and the label which is one of 25 medical symptoms. In order to develop the NLP model, I used the English language dictionary. In order to answer the research questions, a sub selection of the data is needed, where the label is either "Cough" or "Infected wound".

3 Exploratory Data Analysis (EDA)

In Figure 1, it's evident that the label frequency is quite balanced, indicating no need to account for unbalanced observations. The subset of data we're interested in comprises 599 observations.

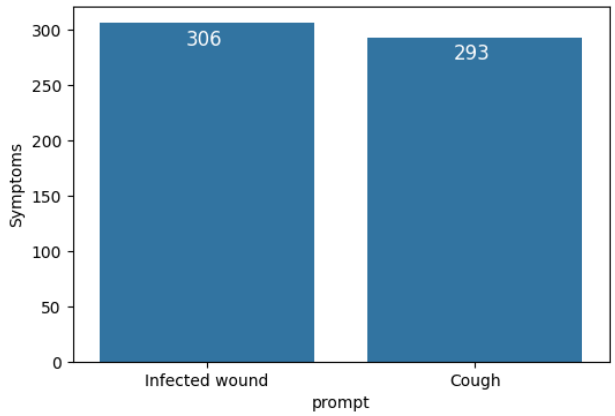

Figure 1: Label frequencies

In Figure 2, we can see the density of the overall quality score of the audio files (left plot) and audio length (right plot) by each label. We can see that there are more audio files with better overall quality for the "Infected wound" label than the "Cough" label and significantly longer audio files too.

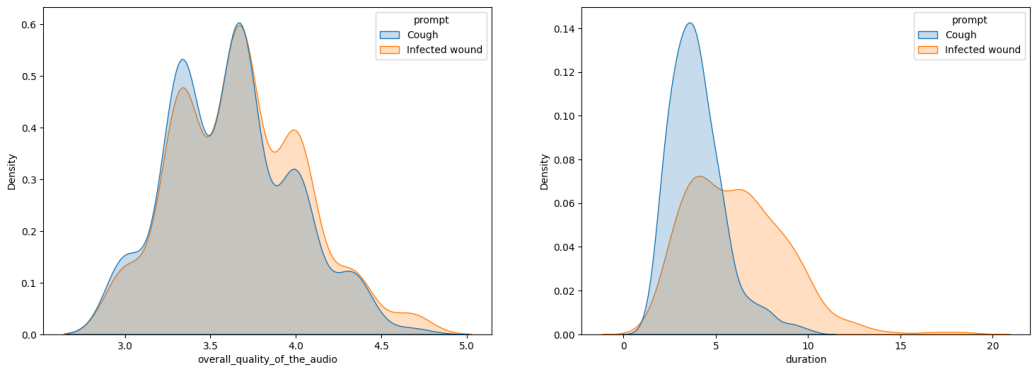

Figure 2: Left plot: Density of the overall quality score of the audio file for each label. Right plot: Density of the duration of the audio file for each label.

In Figure 3, we can see the absolute correlation between all the features engineered in the Feature Engineering section. We can see really high correlation between some features (in bright red), which could result in bad predictions of our label. Removing high correlated features using LASSO and fitting the remaining features in the other models preformed poorly, therefore, I didn't removed any feature.

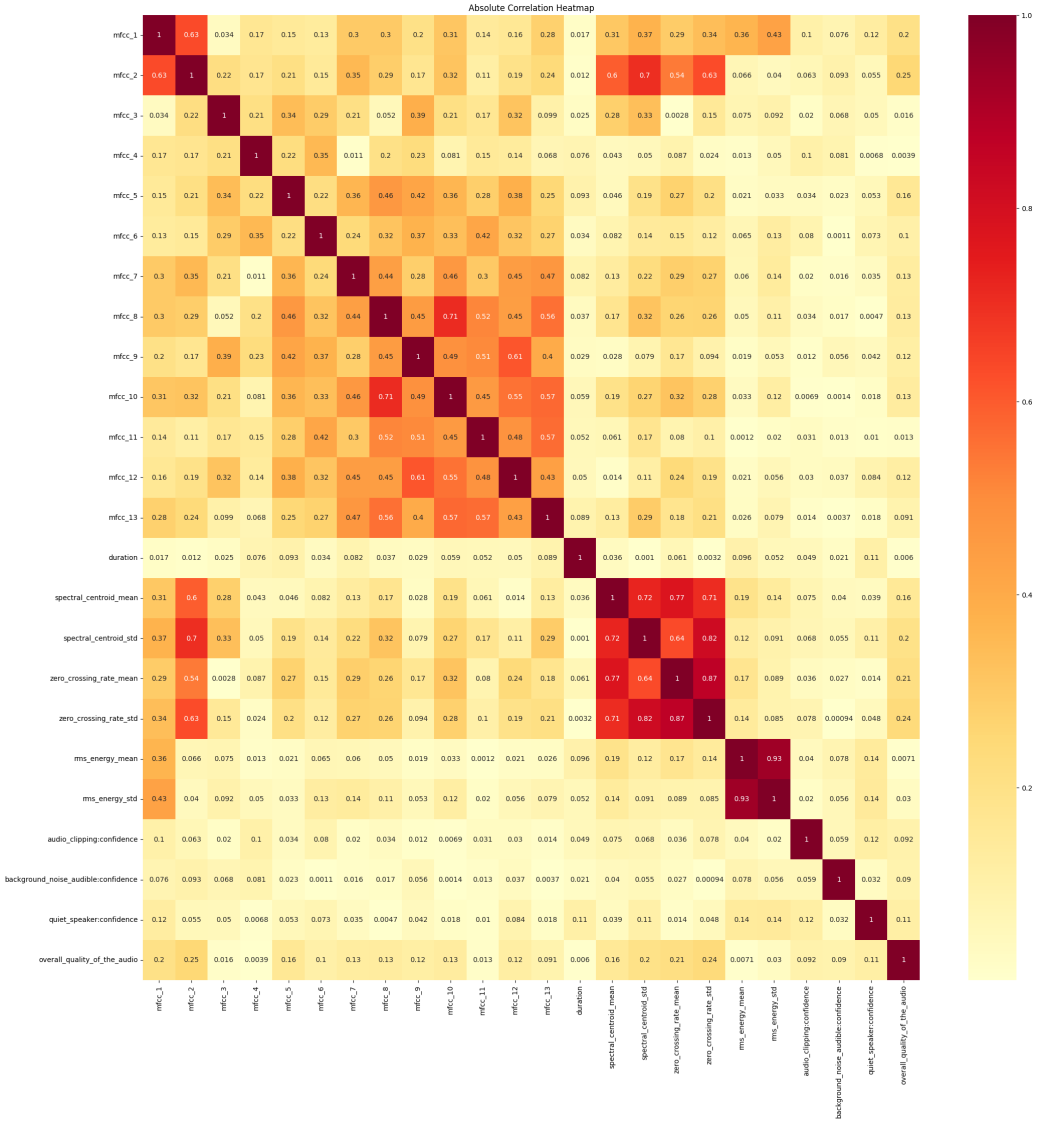

Figure 3: Absolute correlation matrix between all the features

In Figure 4, we can see a word cloud of the non-stop words in the transcription for the audio files. The size of the word is proportional to its frequency in all the transcriptions.

Figure 4: Word cloud of non-stop words in the data

3.1 Clustering Method

In addition, I tried to use clustering methods to see any interesting patterns in the data (without the label), such as K-Means, Gaussian Mixture Model and hierarchical clustering, before and after Principal Component Analysis. The most interesting finds are from fitting the Gaussian Mixture Model (GMM) which assumes K groups which distribute Multivariate Normal with expectation μ_k, covariance Σ_k for each $k \in K$, in addition, each observation is randomly assigned to one group and the proportion the group π_k is estimated as well as μ_k and Σ_k. In the next step the Expectation-Maximization (EM) algorithm is preformed.

The GMM algorithm is:

1. Initialization Step: Estimate the parameters for each group k as follow:
   - $\hat{\mu}_k$ - group k mean vector.
   - $\hat{\Sigma}_k$ - group k sample covariance matrix.
   - $\hat{\pi}_k$ - group k sample proportion.
2. Do Expectation-Maximization:

1. Expectation Step: For each data point X_i, we calculate the probability that the data point belongs to group (c) using Bayes equation:
$$r_{ik} = \frac{\hat{\pi}_k \cdot f(x_i|\hat{\mu}_k, \hat{\Sigma}_k)}{\sum_{c=1}^{K} \hat{\pi}_c \cdot f(x_i|\hat{\mu}_c, \hat{\Sigma}_c)}$$
where $f(x_i|\hat{\mu}_k, \hat{\Sigma}_k)$ is the PDF of the Multivariate Normal distribution given by:
$$f(x_i|\hat{\mu}_k, \hat{\Sigma}_k) = \frac{1}{(2\pi)^{\frac{n}{2}} \cdot |\hat{\Sigma}|^{\frac{1}{2}}} \cdot \exp\left\{-\frac{1}{2} \cdot (x_i - \hat{\mu}_k)^T \cdot \hat{\Sigma}_k^{-1} \cdot (x_i - \hat{\mu}_k)\right\}$$
2. Maximization Step: When maximizing the likelihood function (which is not specified here) we get that the new estimations for the parameters are:
$$\tilde{\mu}_k = \frac{\sum_{i=1}^{m} r_{ik} \cdot x_i}{\sum_{i=1}^{m} r_{ik}}, \quad \tilde{\Sigma}_k = \frac{\sum_{i=1}^{m} r_{ik} \cdot (x_i - \tilde{\mu}_k)^2}{\sum_{i=1}^{m} r_{ik}}, \quad \tilde{\pi}_k = \frac{\sum_{i=1}^{m} r_{ik}}{m}$$
where m is the number of data points in group k. Then, we do the Expectation-Maximization steps until convergence.

The GMM divided the data in two groups:

- Group 0: Speakers with heavy Indian accent.
- Group 1: Speakers with light or none Indian accent.

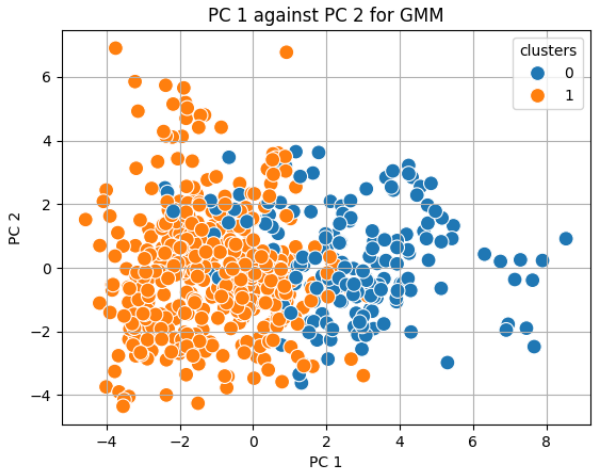

Figure 5: Clustering using GMM visualizing on PC1 vs. PC2

4 Feature Engineering

In order to develop a model for classification from the audio files I converted each audio file to the following features:

- The duration of the audio file in seconds.
- The mean value of the 13 MFCCs coefficient values across all time series frames each divided by its standard deviation.
- The mean and standard deviation spectral centeroid across all time series frames, calculated by

$$\text{Centroid}(t) = \sum_k \frac{S[k,t] \cdot freq[k]}{\sum_j S[j,t]}$$

  where, t is the time frame, S is a magnitude spectrogram and freq is the array of frequencies of the rows of S.
- The mean and standard deviation zero crossing rate across all time series frames, which means the mean rate at which the signal changes its sign (the number of times the audio waveform crosses the horizontal axis).
- The mean and standard deviation RMS (root mean square) energy across all time series frames.

To that I added the quality measures confidence score and the overall quality score, which resulted in 24 features.

In order to develop a model for classification from the transcriptions I used some conventional method in language process, I deconstructed words (I'm=I am, I've=I have...), lower cased and strip out punctuation. Then I removed the stop-words, lemmatized and removed the words 'cough', 'coughing', 'infected', 'infection', 'wound' and 'cut' and did one-hot encoding, to that again I added the quality measures confidence score and the overall quality score, which resulted in 174 features.

5 Modeling

5.1 Modeling Process

Three modeling processes were compared:

1. One audio model of all 24 features.
2. Two audio models of all 24 features each for the different groups of accent resulted in the Clustering Method section.
3. One transcript model of the 174 words frequency features from the NLP.

Each process is 10-fold Cross-validated for the same range of parameters in the same training models and would compared by accuracy of the test set (for the second process a weighted average of accuracies was compared).

5.2 Training Models

For all modeling processes, the data was split to train and test 70%-30% (239-180 observations) respectively, a scaler was trained using only the train set and transforming both train and test set and 4 models were fitted: Logistic Regression, KNN and SVM and Classification Tree. For each model there is the following tuning parameter grid:

- Logistic Regression
  - 'Penalty': LASSO (L1) or Ridge (L2).
  - 'C': $\in [e^{-1}, 1]$ – penalty inverse parameter, as it is smaller the stronger the regularization.
- KNN
  - 'K': number of nearest neighbors $\in [3, m]$ in odds numbers. Where, m is the number of observations in the test set.
- SVM
  - 'kernel': Linear, Polynomial, or Radial.

- 'C': $\in [e^{-1}, 1]$ – penalty inverse parameter, as it is smaller the stronger the regularization.
- 'degree': $\in [3, 6]$ for the polynomial kernel.
- 'gamma': $\in \{0.001, 0.01, 0.1, 1\}$ for the radial kernel.

• Classification Tree

- 'Maximal Depth': $\in [3, 7]$ - the maximal depth of the tree.
- 'Minimum Samples in Leaf': $\in [5, 8]$ - the minimum observations in a leaf node.
- 'Criterion': Gini or Entropy.

5.2.1 Mathematical Formulation

Denote $i = \{1, ..., 429\}$ to be index for each medical self-reported phrase recorded in th train set. Denote a_{ij} to be each quality measure confidence score

where $j = \begin{cases} 1 & \text{Audio Clipping} \\ 2 & \text{Quiet Speaker} \\ 3 & \text{Background Noise} \end{cases}$, for each observation i. Denote b_i to be a matrix of Mel-frequency cepstral coefficient size $\ell_i \times 13$, where $\ell_i = \left\lfloor \frac{\text{signal length}_i}{512} \right\rfloor + 1$ and $k = \{1, ..., 13\}$ is the number of the coefficient, for each observation i. Denote c_{im} to be vectors of conversions of each audio file i,

where $m = \begin{cases} 1 & \text{Spectral Centroid} \\ 2 & \text{Zero Crossing Rate} \\ 3 & \text{RMSE} \end{cases}$, each in length ℓ_i.

The logistic regression equation is

$$\begin{aligned} \text{logit}[P(Y = \text{Cough}|X = x_i)] = \; & \beta_0 + \beta_j \cdot a_{ij} + \beta_4 \cdot \text{Overall quality score}_i \\ & + \beta_{4+k} \cdot \frac{\underset{\text{over } \ell_i}{\text{mean}}(b)_{ik}}{\underset{\text{over } \ell_i}{\text{std}}(b)_{ik}} + \beta_{18} \cdot \text{duration}_i \\ & + \beta_{18+m} \cdot \underset{\text{over } \ell_i}{\text{mean}}(c)_{im} + \beta_{21+m} \cdot \underset{\text{over } \ell_i}{\text{std}}(c)_{im} \end{aligned} \tag{1}$$

Given a positive integer K and a test observation x_0, the KNN classifier first identifies the K points in the training data that are closest to x_0, represented by $\mathcal{N}_0$. It then estimates the conditional probability for "Cough" as

the fraction of points in $\mathcal{N}_0$ whose response values is "Cough":

$$P(Y = \text{Cough}|X = x_0) = \frac{1}{K} \cdot \sum_{i \in \mathcal{N}_0} \mathbb{I}(y_i \text{ is Cough}) \tag{2}$$

Finally, KNN classifies the test observation x_0 to the class with the largest probability.

The SVM equation is

$$\begin{aligned} &\text{Maximize } M \\ &\text{subject to } y_i \cdot \left(\beta_0 + \textstyle\sum_{i \in \mathcal{S}} \alpha_i \cdot K(x, x_i)\right) \geq M \cdot (1 - \varepsilon_i) \end{aligned} \tag{3}$$

where $K(x, x_i)$ is a kernel function:

- linear - $K(x_{ij}, x_{i'j}) = \sum_{j=1}^{p} x_{ij} \cdot x_{i'j}$
- polynomial - $K(x_{ij}, x_{i'j}) = (1 + \sum_{j=1}^{p} x_{ij} \cdot x_{i'j})^d$
- radial - $K(x_{ij}, x_{i'j}) = \exp(-\gamma \cdot \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2)$

6 Results

6.1 Full Audio Model

The results for this process are shown in Table 2, the classification tree with the entropy criterion, maximum depth of 2 and minimum 5 samples in leaf node is the most accurate over the test set (73.9%) but holds the lowest F1 score, which means the model is not balanced in its prediction and indeed, the sensitivity is very high (93.6%) and the specificity is very low (52.3%) which means it would classify true "infected wounds" to "cough" (which is really bad). So, for this process, I would prefer the logistic regression model with parameters $C = e^{-0.98}$, $Penalty = LASSO$, with lower test accuracy (71.7%) but the highest F1 score (71.2%). The logistic regression model is easy to understand, the model estimates the probability that a give input belongs to particular class, it works pretty well when the data is not completely separated between the classes. The train set accuracy is 76.1%, which means the model is not overfitting. Moreover, we can see that the SVM with radial kernel has lower test set accuracy (68.9%) then the train set accuracy (90.2%), which indicate of strong overfitting of this model which is really not good.

Model	Parameters	Accuracy	Sensitivity	Specificity	F1 Score	Train Accuracy
LASSO Logistic Regression	C: exp{-0.98}	71.7%	75.5%	67.4%	71.2%	70.2%
KNN	K: 67	67.8%	73.4%	61.6%	67.7%	64.7%
SVM - Radial Kernel	C: exp{-0.2}; Gamma: 0.1	68.9%	70%	67.4%	68.8%	90.2%
Classification Tree - Entropy Metric	Max Depth: 2; Min samples Leaf: 5	73.9%	93.6%	52.3%	67.1%	70.2%

Table 1: Test set accuracy, sensitivity, specificity, F1 score and train set accuracy of various models for the full audio modeling process, without respect to the accent groups. Test set size = 180.

6.2 Two Audio Models

The results for this process are shown in Table 2, for the heavy accent group the most accurate model, in terms of test set accuracy, is the SVM model with linear kernel and regularization parameter $C = e^{-0.4}$ (87.5%) and it is fairly balanced between the sensitivity and specificity. For the light or none accent group we can see that all the best models fitted are overfitting, so the logistic regression model with parameters $C = e^{-1}$, $Penalty = LASSO$ which has the highest test accuracy is chosen for the best in classification for this group. The two test accuracy metrics were combined to one metric as follows:

$$Accuracy = \frac{56}{180} \cdot 87.5\% + \frac{124}{180} \cdot 66.9\% = 73.3\%$$

which is higher then the best model chosen for the full Audio model process (71.7%).

Model	Parameters	Accuracy	Sensitivity	Specificity	F1 Score	Train Accuracy
Within the heavy accent speakers						
LASSO Logistic Regression	C: exp{-0.16}	83.9%	85.2%	82.1%	84.0%	76.2%
KNN	K: 51	64.3%	81.5%	48.3%	60.7%	62.3%
SVM - Linear Kernel	C: exp{-0.4}	87.5%	85.2%	89.7%	87.4%	77.7%
Classification Tree - Entropy Metric	Max Depth: 5; Min Samples Leaf: 7	64.2%	63.0%	65.5%	64.2%	86.2%
Within the light accent speakers						
LASSO Logistic Regression	C: exp{-1}	66.9%	75.9%	60.0%	67.0%	73.4%
KNN	K: 47	61.3%	79.6%	47.1%	66.1%	69.6%
SVM - Radial Kernel	C: exp{-0.7}; Gamma: 0.01	66.1%	88.9%	48.6%	72.8%	92.2%
Classification Tree - Entropy Metric	Max Depth: 6; Min Samples Leaf: 6	64.5%	68.5%	61.4%	64.8%	83.0%

Table 2: Test set accuracy, sensitivity, specificity, F1 score and train set accuracy of various models for the audio modeling process, with respect to the accent groups. Test set size for the heavy accent group = 56 and for the light accent group = 124.

6.3 Transcript NLP Model

In this case the prediction is perfect with accuracy 100%, and it might be because of the small sample size. Also, NLP models are extraordinarily strong in accuracy if preprocessed correctly but since the model have feature for each non-stop word it takes a long time to train the model (around 30_{min}).

7 Conclusions and Future Work

In conclusion, we can say that the best way to predict a symptom using self-reported audio recordings is by transcript it to phrase using speech-to-text software and use the NLP logistic regression model for binary classification, but if we can't do it for some reason, the classification using the two models for each accent group on the audio files themselves could give good results too. For the future work, it would be more correct to do multiclass classification over all the data and use more models for classification.

References

[1] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. An Introduction to Statistical Learning : with Applications in R. New York :Springer, 2013.

[2] T. K. Moon, "The expectation-maximization algorithm," in IEEE Signal Processing Magazine, vol. 13, no. 6, pp. 47-60, Nov. 1996, doi: 10.1109/79.543975.