Predicting County Level Cost Differences for Treating Chronic Obstructive Pulmonary Disease

Jonathan Lin
jolituba@stanford.edu

Michael Smith
msmith11@stanford.edu

Aileen Wang
aileen15@stanford.edu

***Abstract*—Chronic Obstructive Pulmonary Disease, or COPD, is the third leading cause of death in the United States. [1] In 2010, the estimated healthcare cost of COPD totaled over $32 billion with Medicare and Medicaid comprising 76% of total COPD spending; by 2020, the CDC predicts COPD will cost the United States over $49 billion. This paper seeks to classify whether or not a given county will have an increase/decrease in average principle cost for COPD in the next year as well as discretize that increase/decrease. To do so, we employ linear least squares as a baseline before testing binary and multiclass classification algorithms such as logistic regression, SVMs, and Naïve Bayes. In our preliminary work, we have found that linear least squares performs comparably to our binary classifiers. We have also found that multiclass classification performs worse than binary classification. Going forward, we seek to improve the complexity of our model to fix underfit.**

I. Introduction

Chronic Obstructive Pulmonary Disease, or COPD, is the third leading cause of death in the United States [1]. The National Institutes of Health state that 16 million Americans have been diagnosed with COPD; however, potentially more may have it but are unaware. Most patients are above the age of 40 [2]. COPD is characterized by varying degrees of both emphysema and chronic bronchitis, and leads progressively to disability [1].

II. Problem

The CDC estimates that COPD cost the United States $32.1 billion in 2010, with 76% of that being attributable to Medicare and Medicaid billings [3]. By 2020, COPD is projected to cost $49 billion [3]. Given such a rapid cost increase, we seek to explore the year-to-year change in average principle cost in Medicare & Medicaid spending at the local (county) level, rather than the national, in order to predict whether or not there will be an increase/decrease in average principle cost for COPD in a given county (binary classification). From this, we will classify how significant that increase/decrease is (multiclass classification). This project builds on similar projects from CS229 in the past which focused on estimating general Medicare costs based on features such as state, county, age, gender, and race [5], [6]. Our key difference is that we look at percentage increase in cost from one year to the next, rather than the general Medicare cost, which represents a new angle to a semi-familiar problem.

III. Data Gathering

We started by using data from the Centers for Medicare and Medicaid Services databases [7]. The data we used from these database came in the form of two documents. The first document contained the average principal cost for Chronic Obstructive Pulmonary Disease, sorted by county and year, which would become the dependent variable in our prediction models (our $y^{(i)}$). The other document contained county profile data over the years 2012-2015, grouped by features, such as unemployment rate, average income, and percent below poverty.

Furthermore, we found external datasets from the Center for Disease Control's National Environmental Public Health Tracking Network [4] that contained additional features that we thought would be relevant to our models, including pollutant levels, tobacco use, and asthma prevalence by county. Together, the average principal cost, county profile data, and external features constitutes our entire dataset.

In order to convert this data to a usable form, we wrote several scripts to split, merge, and reorganize the data such that it was organized by training example. This process involved first calculating the dependent variable, the percentage difference in average principal cost from one year to the next. Then, we split the county profile data into separate documents grouped by feature and year. Finally, another script combined all of the values for each county in a particular year into a single row, which constituted a single training example.

IV. Feature Selection

There are 2118 rows in the dataset. Each row consists of 12 fields. For example, row1 = {year: 2013, county: Houston County, State: GEORGIA, copd_prev: 17, asthma_prev: 6, tobacco_prev: 10, med_income_5y: 54893, perc_below_poverty_1y: 0.1618, perc_below_poverty_5y: 0.1521, unemployment_1y: 0.0823, unemployment_5y: 0.1033, apc_percent_diff: -0.0504}.

We will choose the following features for both binary and multiclass predictions: features = {copd_prev, asthma_prev, tobacco_prev, med_income_5y, perc_below_poverty_1y, perc_below_poverty_5y, unemployment_1y, unemployment_5y}. See Appendix 1 for explanations on feature abbreviations.

For binary classification, we define the target y value as

$$y = \begin{cases} 1 & \text{if apc\_percent\_diff > 0} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

For multiclass classification, we divided the apc_percent_diff decreasing and increasing percentage into the following intervals: [-100%, -20%], [-20%, -10%], [-10%, 0%], [0, 10%], [10%, 20%], [20%, 100%] and define the target y value as

$$y = \begin{cases} -2 & \text{if apc\_percent\_diff < -0.20} \\ -1 & \text{if apc\_percent\_diff < -0.10} \\ 0 & \text{if apc\_percent\_diff < 0} \\ 1 & \text{if apc\_percent\_diff < 0.10} \\ 2 & \text{if apc\_percent\_diff < 0.20} \\ 3 & \text{otherwise} \end{cases} \tag{2}$$

V. Baseline: Linear Least Squares Classification

We used this simple baseline as the lower bound check for our classification accuracy in the more complex models that we implemented. We implemented a simple linear least squares classifier, using a classification function of

$$\hat{y}^{(i)} = \textbf{sign}\left(x^{(i)^T}\theta + \theta_0\right)$$

(Note that this model results in classifications of 1 and -1, rather than 1 and 0 as defined previously.) We trained this simple classifier on 1482 counties and tested the classifier for 636 counties. For our baseline, we achieved

Test	$\hat{y} = 1$	$\hat{y} = -1$
y = 1	78	574
y = -1	60	770

Train	$\hat{y} = 1$	$\hat{y} = -1$
y = 1	33	257
y = -1	30	316

Which corresponds to a train accuracy of 57.22% and a test accuracy of 54.87%. We will use these accuracies as a baseline evaluation metric for experiments using more advanced models.

VI. Classification Algorithms

All classification algorithms described below were implemented in binary and multiclass classification, as described in the feature selection section. Binary classification predicted whether the APC of each county for treating COPD increases or decreases, while multiclass predicts the bucket of APC percentage increase or decrease. We began by implementing many different algorithms to see which performed well and which didn't. This will allow us to perform further work after the milestone in enhancing one or two well-performing algorithms and decrease error rates.

A. Logistic Regression

We used the logistic regression method in the linear model class of SKLearn, with the parameters of regularization $C = 1e5$ and Solver = 'lbfgs'. While logistic regression is primarily binary, we used the multiclass variant of the method with an additional parameter of multi_class = 'multinomial' for predicting the percentage increase or decrease bucket. Logistic Regression assumes that features are roughly linear and the problem is linearly separable. We varied the strength of regularization, finding that less regularization - a larger C-value - yielded slightly higher test accuracy for our mode. We observed the train and test accuracy of logistic regression both for binary and muticlass as listed in the table below:

Accuracy	Train	Test
Binary	0.5553	0.5660
Multiclass	0.3704	0.4056

B. Multi-layer Neural Network

We used the MLPClassifier (multi-layer perceptron) class in the neural_network package of SKLearn with a quasi-Newton solver 'lbfgs'. Neural networks use multiple hidden layers to learn a non-linear function. We observed the train and test accuracy for MLP as follows:

Accuracy	Train	Test
Binary	0.8083	0.4764
Multiclass	0.6605	0.3270

C. Support Vector Machines

We used the SVC class in the svm package of SKLearn. SVMs treats feature vectors as high dimensional points in space, which it tries to separate with a hyperplane in order to create the largest margin between the points and the decision boundary. We observed the train and test accuracy for SVM as follows:

Accuracy	Train	Test
Binary	0.5533	0.5597
Multiclass	0.3670	0.4119

D. Naive Bayes

We used the GaussianNB class in the naive_bayes package of SKLearn. Naive Bayes looks at each feature independently and assigns probabilities to each feature value based on the y value. We observed the train and test accuracy for Naive Bayes as follows:

Accuracy	Train	Test
Binary	0.5506	0.5487
Multiclass	0.3535	0.3584

VII. Evaluation Metrics

We are using score and accuracy_score methods (see equation (3) and (4) below) in SKLearn package to measure the training and test accuracy of the above classification algorithms, which are calculated by dividing the number of correct predictions by the total number of train and test data examples respectively.

$$Train\_Accuracy = Fit.score(X\_train, Y\_train) \tag{3}$$

$$Test\_Accuracy = accuracy\_score(pred, Y\_test) \tag{4}$$

VIII. Results and Analysis

We first use the train_test_split method in the SKLearn package to randomly select 1482 (70%) for the train dataset and 636 (30%)for the test dataset from the dataset with 2118 elements defined in Section VI. Then we run the classification algorithms in Section VI on this dataset configuration and obtained the following results:

- After running our binary and multiclass classifiers on the selected features, we observed that logistic regression and SVM performed the best (See Figure 1 and 2). Since these algorithms expect linear features or features that interact linearly, it implies that the selected features in our model are more linearly correlated.
- Although MLP has the highest train accuracy, it performed the worst in term of the test accuracy. This is expected since MLP algorithm is more suitable for non-linear data models.
- We also tried to add more features, such as, num_days_rain, perc_cultivated_land_use, perc_forest_cover, perc_park_proximity, rural_urban, total_pollutants, acetaldehyde, benzene, formaldehyde. But these features were not changing year by year and seem to be not correlated with the features in section IV. It has degraded classification performance both for binary and multiclass.

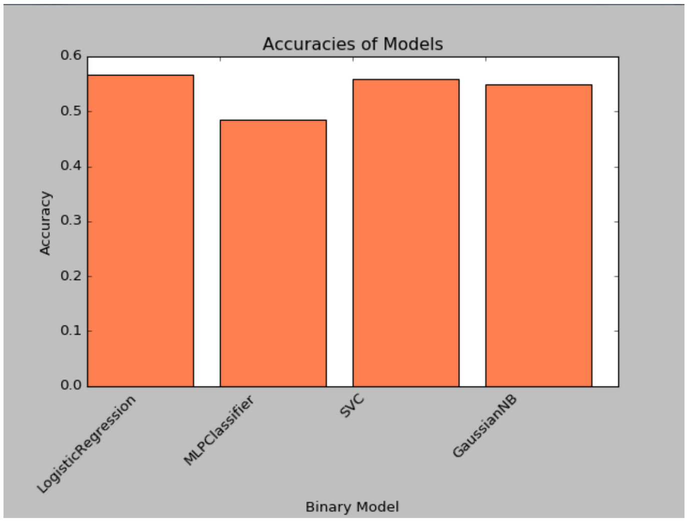

Fig. 1. Test accuracy of Binary Classifications

IX. Future Work

Looking forward to the final submission of this project, we have several goals. First, we would like to implement a few more learning algorithms to have a more robust set of algorithms in our toolbox. Then, we plan to revisit the algorithms already implemented and perform error diagnosis and improve the performance. Based on the work done thus far, we have what looks to be the same issue across all models: high bias and underfitting. In order to remedy this, we plan to improve our models by creating larger models and adding more features, as discussed in class. Also, we will use Chi-squared as the scoring function to determine the optimal number of features should be used. We plan to take our best performing models (Logistic Regression and SVM) and focus on their performance, in order to hopefully achieve a model that performs at an acceptable level, rather than have multiple models with sub-desirable performance.

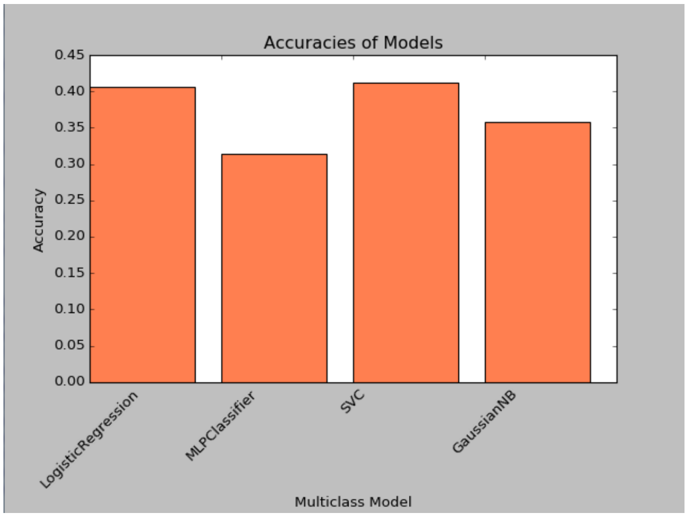

Fig. 2. Test accuracy of Multiclass Classifications

X. Contributions

All of us equally contributed to different parts of the milestone. Everyone was involved in extracting features from different sources and merging the datasets together. Jonathan focused on implementing the linear least squares classification and error analysis/future work, while Aileen and Michael worked on the binary and multiclass classification algorithms.

XI. Appendix I: Feature Abbreviations

- year = year
- county = county
- state = state
- copd_prev = COPD prevalence (percent)
- asthma_prev = asthma prevalence (percent)
- tobacco_prev = tobacco usage prevalence (percent)
- med_income_5y = median income (5 year average)
- perc_below_poverty_1y = percent below poverty (1 year average)
- perc_below_poverty = percent below poverty (5 year average)
- unemployment_1y = percent unemployed (1 year average)
- unemployment_5y = percent unemployed (5 year average)
- apc_perc_diff = percent difference in average principal cost for treating COPD (y value)

REFERENCES

[1] What Is COPD? - NHLBI, NIH, Nhlbi.nih.gov, https://www.nhlbi.nih.gov/health/health-topics/topics/copd/

[2] Risk Factors - NHLBI, NIH, Nhlbi.nih.gov, https://www.nhlbi.nih.gov/health/health-topics/topics/copd/atrisk

[3] CDC Features - Increase expected in medical care costs for COPD, Cdc.gov, https://www.cdc.gov/features/ds-copd-costs/index.html

[4] National Environmental Public Health Tracking Network Query Tool, Ephtracking.cdc.gov, https://ephtracking.cdc.gov/DataExplorer/#/

[5] J. Louie and A. Wells, "Predicting Medicare Costs Using Non-Traditional Metrics", http://cs229.stanford.edu/proj2016spr/report/029.pdf, 2016.

[6] S. Rosston and S. Steele, "The Price Is Right? Estimating Medicare Costs with Machine Learning", http://cs229.stanford.edu/proj2015/291_report.pdf, 2016.

[7] "Mapping Medicare Disparities", https://data.cms.gov/mapping-medicare-disparities, 2017