Predicting Lung Cancer with Machine Learning

September 23, 2022

1 Problem Formulation

In this paper, I will try to predict, if a patient with specific attributes has lung cancer or not.

1.1 Datapoints

The dataset consists of 309 entries in an execel sheet, with patients between 21 and 87 years. Each datapoint describes a patients with various attributes. The dataset is licsensed under the *CCO: Public Domain* and was found on *Kaggle* [1]. For transparency reasons, this dataset is used in another project which can be found on *here* [2], but since this this work was done independantly it was by no means copied. The dataset is complete and has no missing features. Each datapoint is depicted with 16 different attributes (Gender, Age, Smoking, Yellow Fingers, Chronic Desease etc..).

The label for this prediction will be wether or not a patient has lungcancer (yes/no), which implys that this problem is a classification problem. I chose to change the dataset for the labels from *yes* and *no* to *-1* and *1*, for easier prediction and because I will be using the SVC hinge loss funciton in the second part. All attributes except for the gender and age are represented with *1* or *2* which correspond to *no* and *yes* respectively. The Gender is represented with M or F which corresponds to male and female, respectively. The age is the actual age as a number. The histogramm, shows that the data is mostly balanced except for few outliers like *Fatigue* and *Lung cancer*. Since I will be using logistic regression these outliers will not impact our results significantly [3][4].

Additionally, I was able to find 33 duplicates which I got rid of. As one may have already realized, the dataset is not all too big, as there are just 276 entries.

1.2 Feature selection

For the features I selected all attributes apart from the lungcancer column, as it is used for the label. For the selected features I chose to change the datapoint from 1 and 2 to 0 and 1 for convention and for the sake of understandability. Important to note here, is that except for the age, all features are representable in binary, meaning a normalization is not needed. After looking at the correlation heatmap of the feature, I could not identify any bad datasets. All datapoints seem to be in range of -0.75 and 0.75 which is a common indicator for correlation. All labels, gender, age, smoking, yellow fingers, (. . .) intuitively seem to be important as features of wether a patient has lung cancer or not. After some minor testing with *Variance Inflation Factor (VIF)* I could validate that all 15 features seem to help the accuracy of the prediction.

1.3 Data Splitting

As already mention, the dataset is not very big, which is why I chose to use the k-fold cross validation method. This method is suitable for datasets with fewer entries. The algorithm works by randomly splitting up the data sets in equal parts and using one set as the test set and the rest as training set. I decided to split the data in 5 parts (k=5) to achieve a training to test ratio of 80% to 20% which is common for achieving good results.

1.4 Machine learning model

Our goal is to classify wether a patient has lungcancer. This can be described as a binary classification problem.

For the first machine learning model I chose a logistic regresion, as it seemed to be a reasonable choice for a simple binary classification problem. Logistic regression uses a linear hypothesis space and works by setting a demlimiter between the datapoints. On a 2D space one would put a line between the given datapoints. In 3D space the points would be seperated by a plane and in higher dimension a hyperplane would be used to describe the separation. As for the loss function, I chose the logistic loss function as it is a very common and established function for logistic regresion.

2 Problem statement

In this paper I will try to predict whether a patient with certain characteristics has lung cancer or not.

2.1 Data points

The data set consists of 309 entries in an Excel spreadsheet, with patients between 21 and 87 years old. Each data point describes a patient with different characteristics. The dataset is licensed under the *CCO: Public Domain* and was found on *Kaggle*. For transparency, this dataset is used in another project found on *here*, but as this work was done independently, it has not been copied in any way. The dataset is complete and has no missing features. Each data point is mapped with 16 different attributes (gender, age, smoking, yellow fingers, chronic disease, etc. . .).

The label for this prediction is whether a patient has lung cancer or not (yes/no), which means that this problem is a classification problem. I decided to change the dataset for the labels from *yes* and *no* to *-1* and *1* to make the prediction easier and because I will use the SVC hinge loss function in the second part. All attributes except gender and age are represented with *1* or *2*, which corresponds to *no* and *yes* respectively. Gender is represented with *M* or *F*, which means male or female respectively. The age is the actual age in the form of a number. The histogram shows that the data is largely balanced, except for a few outliers such as *fatigue* and *lung cancer*. Since I will be using logistic regression, these outliers will not significantly affect our results.

Additionally, I was also able to find 33 duplicates, which I have eliminated. As you may have already noticed, the dataset is not too large, now containing only 276 entries.

2.2 Feature selection

For the characteristics, I selected all attributes except for the lung cancer column, as this is used for the label. For the selected characteristics, I have changed the data points from *1* and *2* to *0* and *1* for convention and ease of understanding. It is important to note here that with the exception of age, all characteristics can be represented in binary, meaning, a normalisation is not required. After looking at the correlation heat map of the characteristic, I could not identify any bad data sets. All data points appear to be in the range of -0.75 and 0.75, which is a common indicator of correlation. Also, all the identifiers, gender, age, smoking, yellow fingers, (...) seem intuitively important in determining whether a patient has lung cancer or not. After some small tests with *Variance Inflation Factor (VIF)* I could confirm that all 15 characteristics seem to improve the accuracy of the prediction.

2.3 Data breakdown

As mentioned earlier, the data set is not very large, so I decided to use the k-fold cross-validation method. This method is suitable for data sets with fewer entries. The algorithm works by randomly dividing the datasets into equal parts and using one set as the test set and the rest as the training set. I decided to split the data into 5 parts (k=5) to achieve an 80% to 20% ratio, which is very common.

2.4 Machine learning model

Our goal is to classify whether a patient has lung cancer. This can be described as a binary classification problem.

For the first machine learning model, I chose logistic regression because it seemed reasonable for a simple binary classification problem. Logistic regression uses a linear hypothesis space and works by setting a limiter between the data points. In a 2D space, you would put a line between the given data points. In 3D space, the points would be separated by a plane, and in higher dimensions you would use a hyperplane to describe the separation. For the loss function, I chose the logistic loss function, as it is a very common and proven function for logistic regression.Some text in which I cite an author.

3 References

- [1] Data set from Kaggle: `https://www.kaggle.com/datasets/mysarahmadbhat/lung-cancer`
- [2] Other Kaggle project with same data set: `https://www.kaggle.com/code/gaganmaahi224/lung-cancer-5ml-models-full-analysis-plotly`
- [3] How to handle unbalanced sets tutorial : `https://www.kdnuggets.com/2017/06/7-techniques-handle-imbalanced-data.html`
- [4] Unbalanced data in Logistic Regression: `https://stats.stackexchange.com/questions/6067/does-an-unbalanced-sample-matter-when-doing-logistic-regression`
- [5] Heatmap Tutorial *Medium* an seaborn library: `https://medium.com/@szabo.bibor/how-to-\\create-a-seaborn-correlation-heatmap-in-python-834c0686b88e`

- [6] Virtual Inflation Factor tutorial on Geeks for Geeks: https://www.geeksforgeeks.org/detecting-multicollinearity-with-vif-python/

4 Code Appendics