PREDICTIVE LUNG CARE: MODELING EARLY CANCER DETECTION

Table of Contents

Introduction:

Lung cancer is classified histologically into small cell and non–small cell lung cancers. The most common symptoms of lung cancer are cough, dyspnea, hemoptysis, and systemic symptoms such as weight loss and anorexia. High-risk patients who are present with symptoms should undergo chest radiography. We have considered 24 different parameters that have impact on cancer and analyzed the level of cancer that can happen to a person in the future.

The following are the few reasons why we have considered these parameters are: -

- Tobacco use causes 80% to 90% of all lung cancers.
- Secondhand tobacco smoke exposure is also a significant risk factor,
- with younger age at exposure associated with higher risk of lung cancer
- Radon, a naturally occurring radioactive gas found in some homes, is estimated to cause 21,000 cases of lung cancer per year

A combination of intrinsic factors and exposure to environmental carcinogens is involved in the pathogenesis of lung cancer. Therefore, considering these different parameters, we have split the records into train, validate and test records. Using train dataset, we are training the model to generate the rules and verifying its accuracy using validate dataset. At the end, we have considered test data as unforeseen records to find significance of the considered model.

Executive Summary

Cancer ranks as a leading cause of death and an important barrier to increasing life expectancy in every country of the world, and according to available data, those diagnosed early have a 50 percent chance of survival than those diagnosed with late-stage cancer. This means that early detection of cancer is paramount to the survival of the patient and creating an algorithm to perform optimally and accurately predict the detection of cancer early, will by every means help in the reduction of cancer deaths.

This leads to our objective which is to implement different machine learning models on a set of data that has information about patient’s data like; age, gender, weight, genetic risk, etc., clean the data to fit our models, train the models, try to use them to predict patients' likelihood of having lung cancer or not, and then find out which of the algorithms performs optimally.

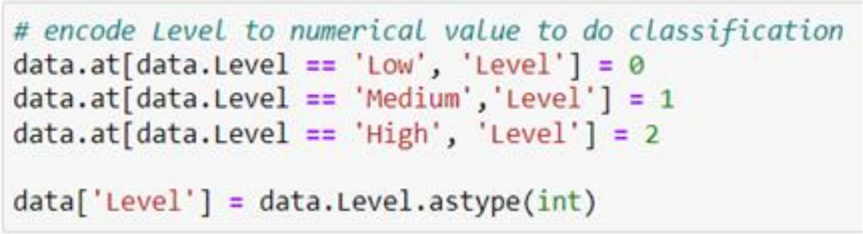
Rational Statement

According to World Health Organization Cancer is the second most common cause of death. Around new, 12.7 million cases were found, and 7.6 million people (about twice the population of Oklahoma) died due to cancer in the whole world. People with insufficient income suffer the most from cancer. It said to be that in the years between 2000 and 2020 it's going to double the cancer reports and more around 2030. Lung cancer takes a couple of years to develop.

- Main reasons due to smoking and vaping
- 25% of all lung cancer are people who started to smoke newly.
- Two people out of three are infected with lung cancer and the common people are above the age of 70.
- Stages of lung cancer
    - Localized – Cancer is in the lungs
    - Regional – Cancer is spread lymph nodes
    - Distant – Cancer is spread to the other parts
- Until the cancer is spread symptoms are not visible such as coughing

Problem Statement

The dataset that we are using is from Kaggle known as the cancer patient data set that holds 1000 records which are classified into 3 different categories of cancer that is low, medium, and high. When looking at the data set the column known as the level contains values such as low, medium, and high, in this case, we are converting the column data into numerical values that are low as 0, medium as 1, and high as 2 in order to do the classification. (Image is attached below)



```
# encode Level to numerical value to do classification
data.at[data.Level == 'Low', 'Level'] = 0
data.at[data.Level == 'Medium','Level'] = 1
data.at[data.Level == 'High', 'Level'] = 2

data['Level'] = data.Level.astype(int)
```

Furthermore, when we look at the metrics that have been used is precision, recall, and F1 score for the random forest. For K-nearest neighbor, the dataset is divided into training and test data and then the model is trained with different values of K to capture the accuracy of the test data, in other words, the classification is based on measuring the distance between the test sample and the training sample to determine the final output. All the algorithms contain the matrix as well as the precision, recall, and F1 score.

Moreover, holdout validation is not used in the model training process and the purpose is to provide an unbiased estimate of the model performance during the training. And most of the AI algorithms are developed with hyperparameter to maximize the model's performance without overfitting or creating too high of a variance.

Data Requirements

We have twelve data requirements for this project.

For the problem of 80% of people are infected with lung cancer by smoking tobacco, the data requirement of large dataset, data diversification, labelled data, data collection and numerical or categorical data are important, so we can get more data to do this project. The problem is about the requirement of below:

- Large dataset: The dataset for this project should be large, as it is better to have more data than less. It should have at least 1000 rows. So, we have enough data to train and test our model. However, it is not necessary to have a huge dataset as it is difficult to work with it and find insights into the data.
- Data Diversification: Data should be diverse. It should include people of different genders and different cancer states. It should account for the different classes of patients in the dataset. So, we can train our model to predict the status of new patients.
- Labelled data: The dataset should be well labelled for this project; we can do classification and regression for supervised learning. Because an unlabeled dataset is difficult to do with algorithms, it has less information about the data and expected results. So, we should have a labelled dataset to train and test our model.
- Data collection: The data collection of the dataset should be clear. We should find a dataset that shows where the data came from. Since patient information is confidential. We should know where the data is to avoid any problems.
- Numerical or categorical data: The data should be numerical or categorical. Because many datasets for cancer are images, image processing is used to detect cancer. However, this project is to predict cancer through numerical or categorical data. So, this dataset should be numerical or categorical.

For the problem of being diagnosed with cancer before it's too late, the data requirement of meaningful features, fewer missing values and outliers are important, so we can do our model easily. The problem is about the requirement of below:

- Sufficient meaningful features: The dataset should have enough meaningful features. We can easily understand the data and spend less time doing data analysis. And it should have enough features for the project to do data analysis.
- Fewer missing values: If possible, the dataset should not have any missing values. Because missing values in the dataset mean that we need to spend time cleaning the data. But it is difficult to find a dataset without any missing values, but this project should still find a dataset with fewer missing values.
- Few outliers: If possible, the dataset should not have any outliers. Because having outliers in the dataset means that the data validation control of the dataset is poor. However, it is difficult to find datasets without outliers, but this project should still find datasets with fewer outliers.

For the problem of saving people from the economic crisis, the data requirement of data format, consistency, accuracy and balance are important, so we can train and test our model to solve the problem. The problem about the requirement of below:

- Data format: The data format should be correct. Because a lot of data comes from various locations or formats. The dataset should have structured data. So, we do not have to spend more time cleaning the data.
- Data consistency: The data should be consistent. It is hard to understand the data if the data is out of the range, such as the value of gender should be male equals 1, female equals 2, but the data has other values than 1 and 2. So we do not have to spend more time cleaning the data.
- Data Accuracy: Data should be correct. It can be difficult to spot if the data has errors. This project should find a reliable source to get the dataset.
- Data balance: The dataset should have enough data for reliable and unreliable data. If the dataset has 900 reliable data but only 100 unreliable data, it is difficult to train a model to understand the unreliable data.

In assumption, we assume that the dataset is trustable in the requirements of below:

- Data collection
- Data Accuracy

In constraints, this project has three data constraints.

First, we cannot change the dataset because the dataset is about patient cancer data. It is difficult for us to change or clean data. It is about the data requirements of below:

- Data format
- Data consistency

Second, we cannot combine one dataset with another. Because there are different datasets with distinctive characteristics. It is about the data requirements of below:

- Data Diversification
- Large dataset

Third, we have no way of knowing the accuracy of the data, we can only make assumptions because the data for this project is about patient information and is confidential. It is about the data requirements of below:

- Data collection
- Data Accuracy

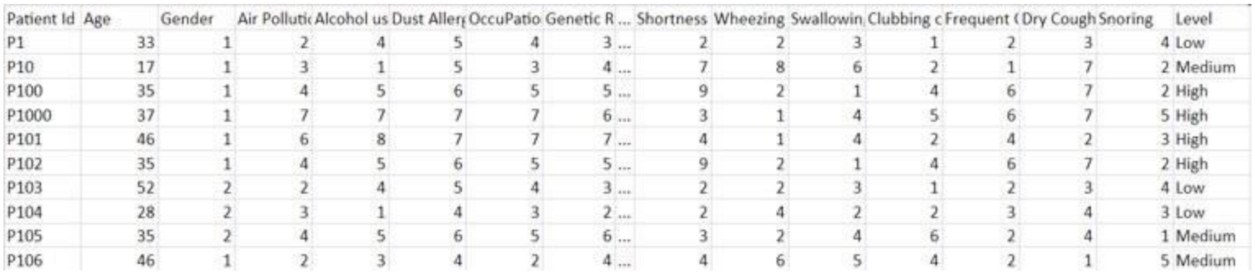
Therefore, we need to find a dataset where we need the most data and the data source should be reliable so that we can avoid constraints.

Insight to Data

We will be using cancer patient data sets.xlsx on Kaggle.com

Link -> https://www.kaggle.com/rishidamarla/cancer-patients-data

The Data holds 1000 observations of which 365 of them are classified as Low while 332 and 303 observations are classified as Medium and High, respectively. The Data has 23 possible features.



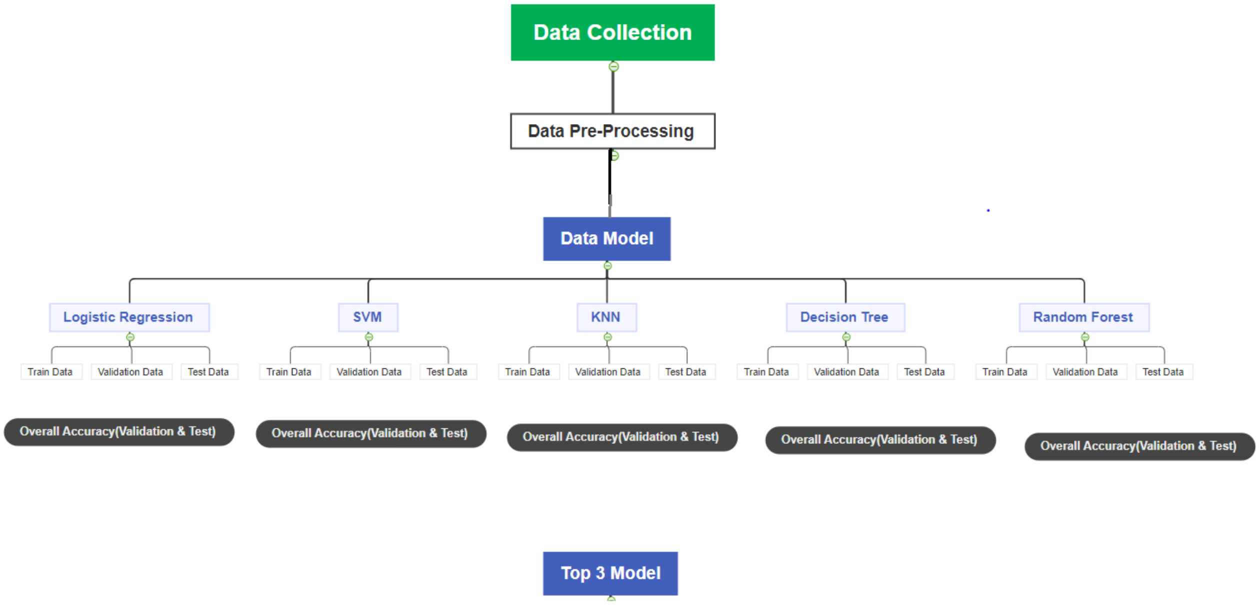
Patient Id	Age	Gender	Air Polluti	Alcohol us	Dust Allerg	OccuPatio	Genetic R	...	Shortness	Wheezing	Swallowin	Clubbing c	Frequent (	Dry Cough	Snoring	Level
P1	33	1	2	4	5	4	3	...	2	2	3	1	2	3	4	Low
P10	17	1	3	1	5	3	4	...	7	8	6	2	1	7	2	Medium
P100	35	1	4	5	6	5	5	...	9	2	1	4	6	7	2	High
P1000	37	1	7	7	7	7	6	...	3	1	4	5	6	7	5	High
P101	46	1	6	8	7	7	7	...	4	1	4	2	4	2	3	High
P102	35	1	4	5	6	5	5	...	9	2	1	4	6	7	2	High
P103	52	2	2	4	5	4	3	...	2	2	3	1	2	3	4	Low
P104	28	2	3	1	4	3	2	...	2	4	2	2	3	4	3	Low
P105	35	2	4	5	6	5	6	...	3	2	4	6	2	4	1	Medium
P106	46	1	2	3	4	2	4	...	4	6	5	4	2	1	5	Medium

Model(s) /Architecture Approach:

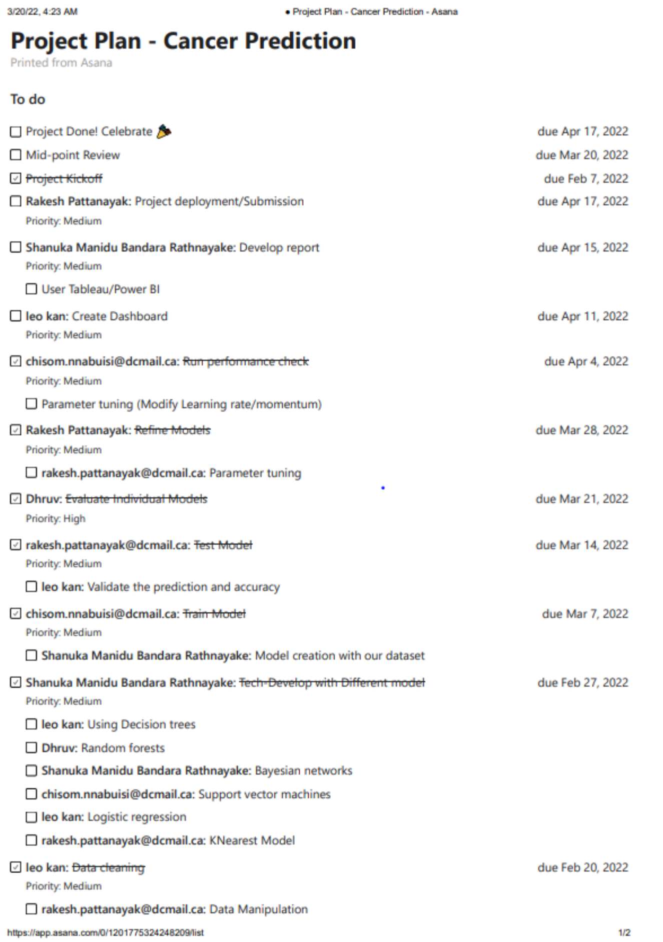
We are trying to compare models using multiple supervised learning models such as:

- Decision trees
- Random forests
- Support vector machines
- Logistic regression
- K-nearest neighbors
- Naïve Bayes

By comparing the precision and recall we will be able to provide the most optimal algorithm for our problem statement.



Project Work Breakdown



3/20/22, 4:23 AM • Project Plan - Cancer Prediction - Asana

Project Plan - Cancer Prediction

Printed from Asana

To do

☐ Project Done! Celebrate 🎉 — due Apr 17, 2022

☐ Mid-point Review — due Mar 20, 2022

☑ ~~Project Kickoff~~ — due Feb 7, 2022

☐ **Rakesh Pattanayak:** Project deployment/Submission — due Apr 17, 2022
Priority: Medium

☐ **Shanuka Manidu Bandara Rathnayake:** Develop report — due Apr 15, 2022
Priority: Medium

- ☐ User Tableau/Power BI

☐ **leo kan:** Create Dashboard — due Apr 11, 2022
Priority: Medium

☑ **chisom.nnabuisi@dcmail.ca:** ~~Run performance check~~ — due Apr 4, 2022
Priority: Medium

- ☐ Parameter tuning (Modify Learning rate/momentum)

☑ **Rakesh Pattanayak:** ~~Refine Models~~ — due Mar 28, 2022
Priority: Medium

- ☐ **rakesh.pattanayak@dcmail.ca:** Parameter tuning

☑ **Dhruv:** ~~Evaluate Individual Models~~ — due Mar 21, 2022
Priority: High

☑ **rakesh.pattanayak@dcmail.ca:** ~~Test Model~~ — due Mar 14, 2022
Priority: Medium

- ☐ **leo kan:** Validate the prediction and accuracy

☑ **chisom.nnabuisi@dcmail.ca:** ~~Train Model~~ — due Mar 7, 2022
Priority: Medium

- ☐ **Shanuka Manidu Bandara Rathnayake:** Model creation with our dataset

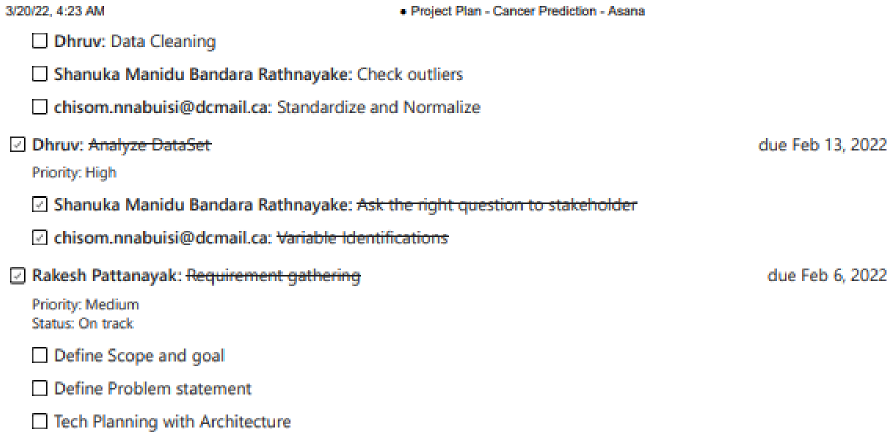
☑ **Shanuka Manidu Bandara Rathnayake:** ~~Tech-Develop with Different model~~ — due Feb 27, 2022
Priority: Medium

- ☐ **leo kan:** Using Decision trees
- ☐ **Dhruv:** Random forests
- ☐ **Shanuka Manidu Bandara Rathnayake:** Bayesian networks
- ☐ **chisom.nnabuisi@dcmail.ca:** Support vector machines
- ☐ **leo kan:** Logistic regression
- ☐ **rakesh.pattanayak@dcmail.ca:** KNearest Model

☑ **leo kan:** ~~Data cleaning~~ — due Feb 20, 2022
Priority: Medium

- ☐ **rakesh.pattanayak@dcmail.ca:** Data Manipulation

https://app.asana.com/0/1201775324248209/list 1/2



3/20/22, 4:23 AM • Project Plan - Cancer Prediction - Asana

- ☐ **Dhruv:** Data Cleaning
- ☐ **Shanuka Manidu Bandara Rathnayake:** Check outliers
- ☐ **chisom.nnabuisi@dcmail.ca:** Standardize and Normalize

☑ **Dhruv:** ~~Analyze DataSet~~ — due Feb 13, 2022
Priority: High

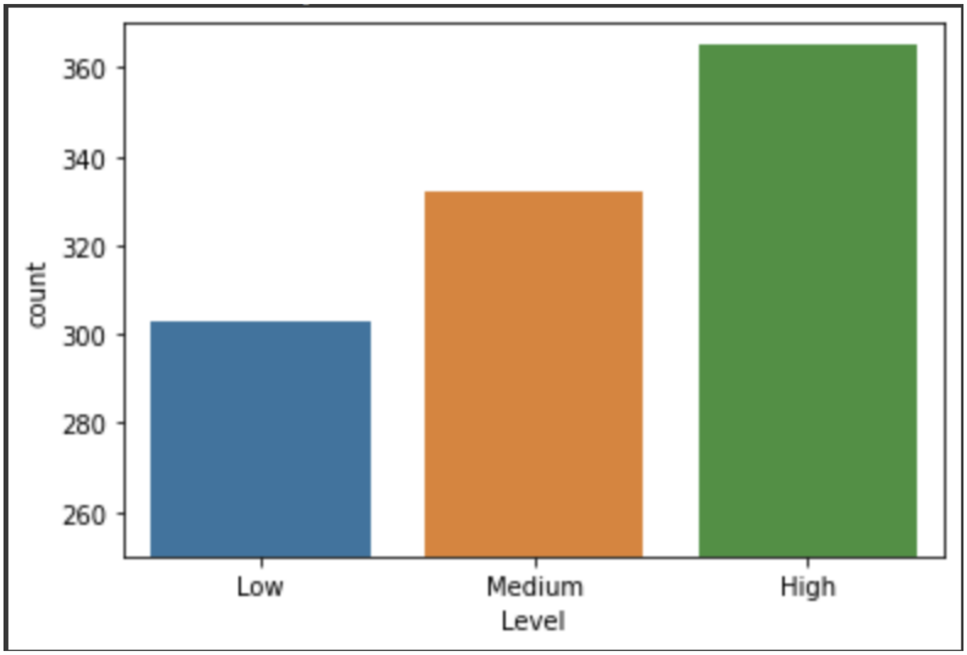
- ☑ **Shanuka Manidu Bandara Rathnayake:** ~~Ask the right question to stakeholder~~
- ☑ **chisom.nnabuisi@dcmail.ca:** ~~Variable Identifications~~

☑ **Rakesh Pattanayak:** ~~Requirement gathering~~ — due Feb 6, 2022
Priority: Medium
Status: On track

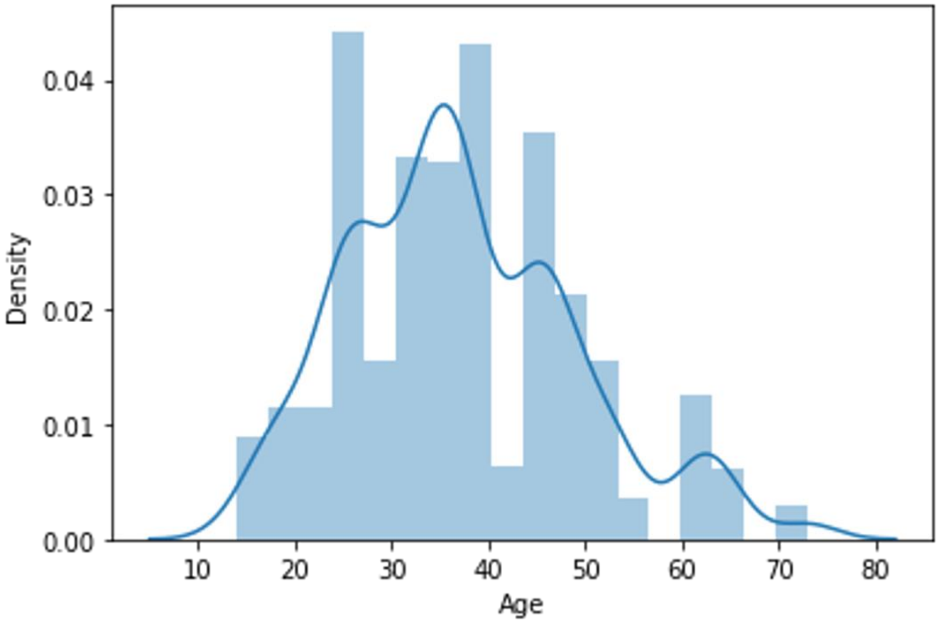
- ☐ Define Scope and goal
- ☐ Define Problem statement
- ☐ Tech Planning with Architecture

Exploratory Data Analysis

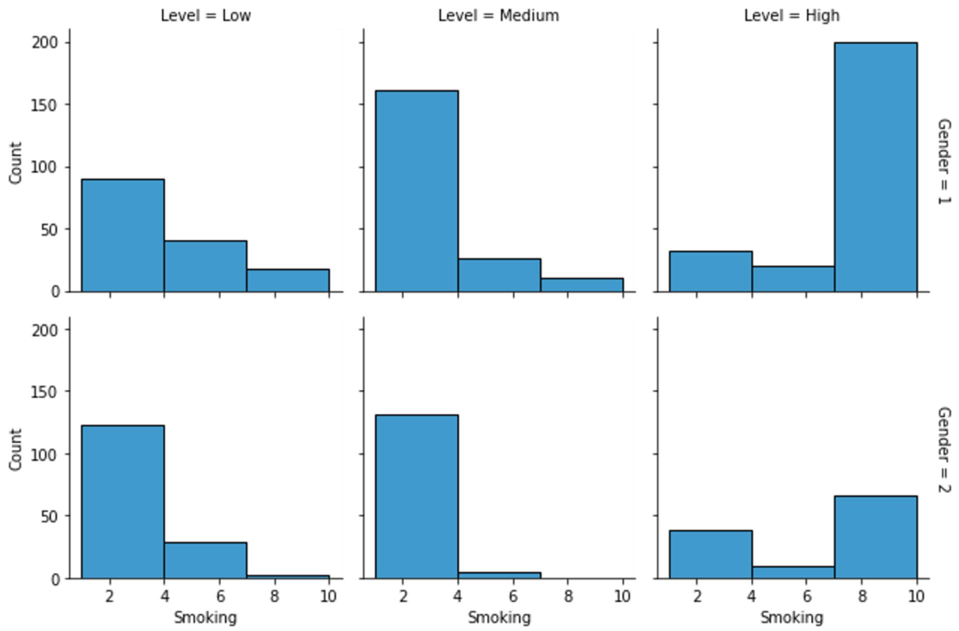
Number of data at each level. We can see that the dataset is a balanced dataset.



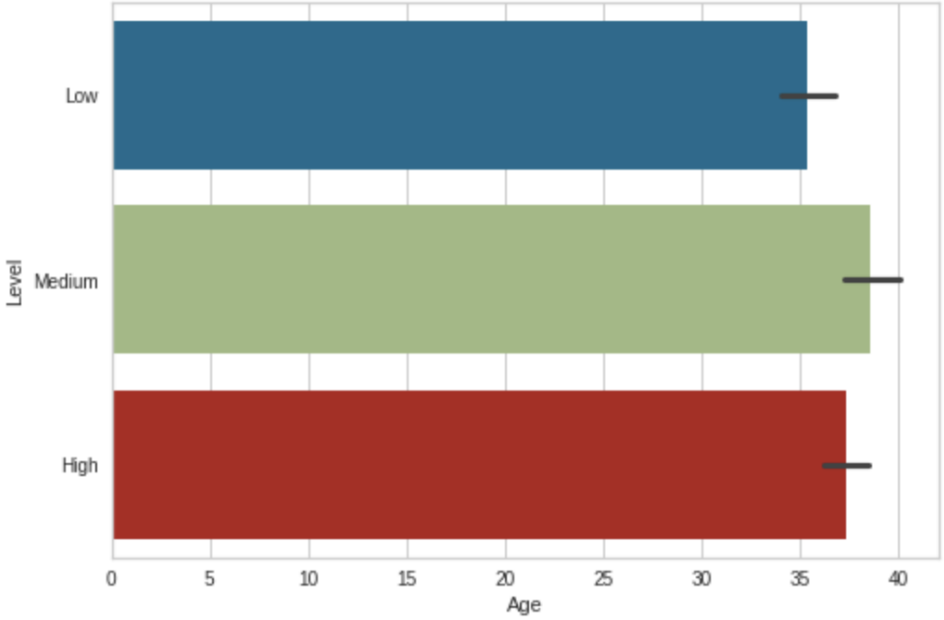
We can know the age distribution. We can see that the highest distribution is between 30 and 40 years old.



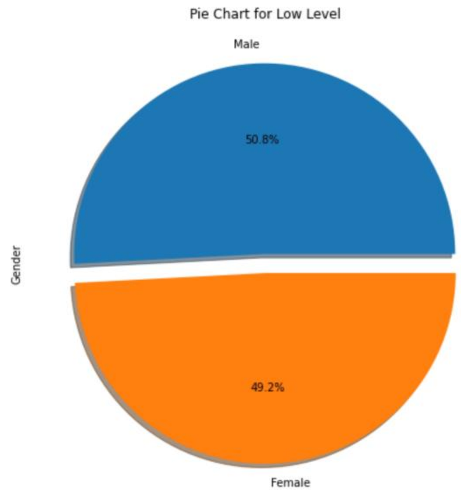
We can learn about cancer levels at different levels of smoking by gender. We can see that if smoking levels are high, men have high cancer levels.

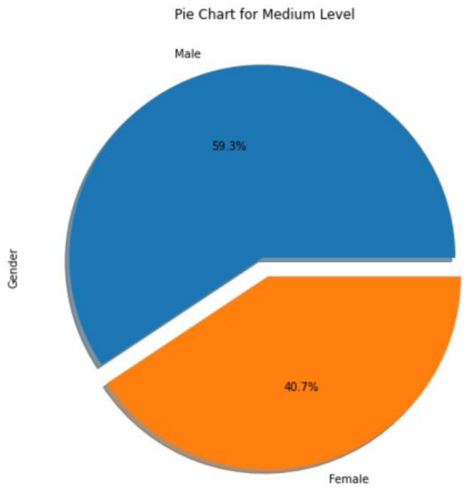


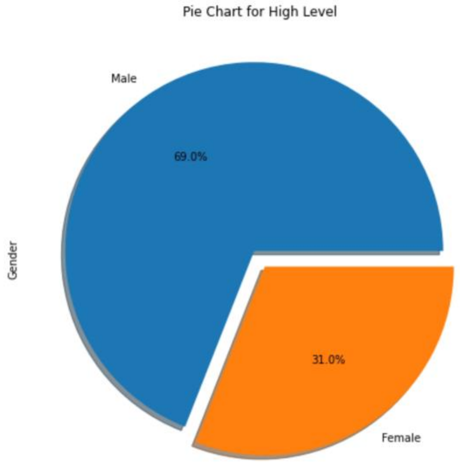
We can know the level of lung cancer by age. We can see that the odds of developing lung cancer are similar at different ages.

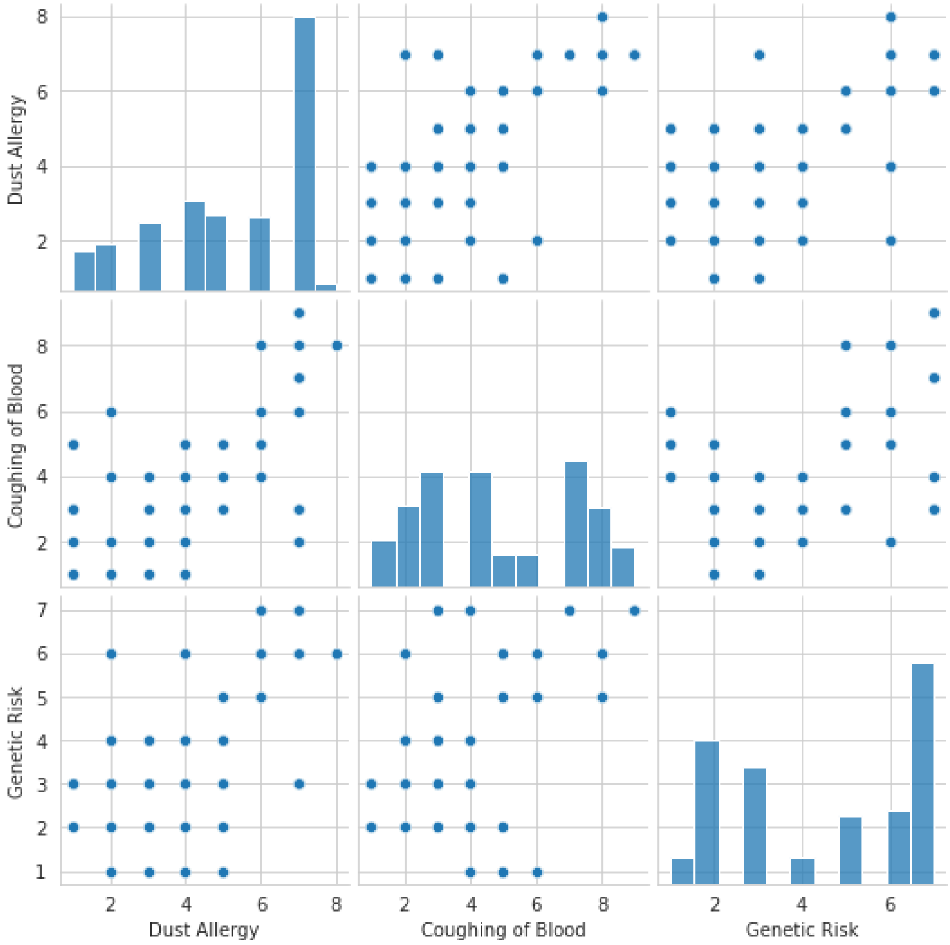


We can know the extent of lung cancer by gender. We can see that men are more likely to develop lung cancer than women.

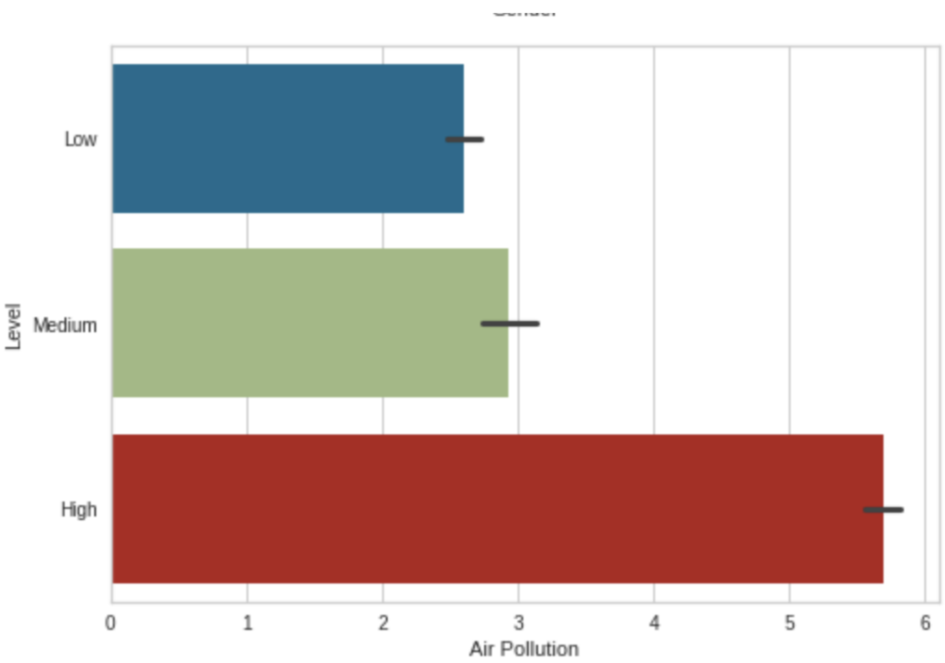

Pie Chart for Low Level


Pie Chart for Medium Level

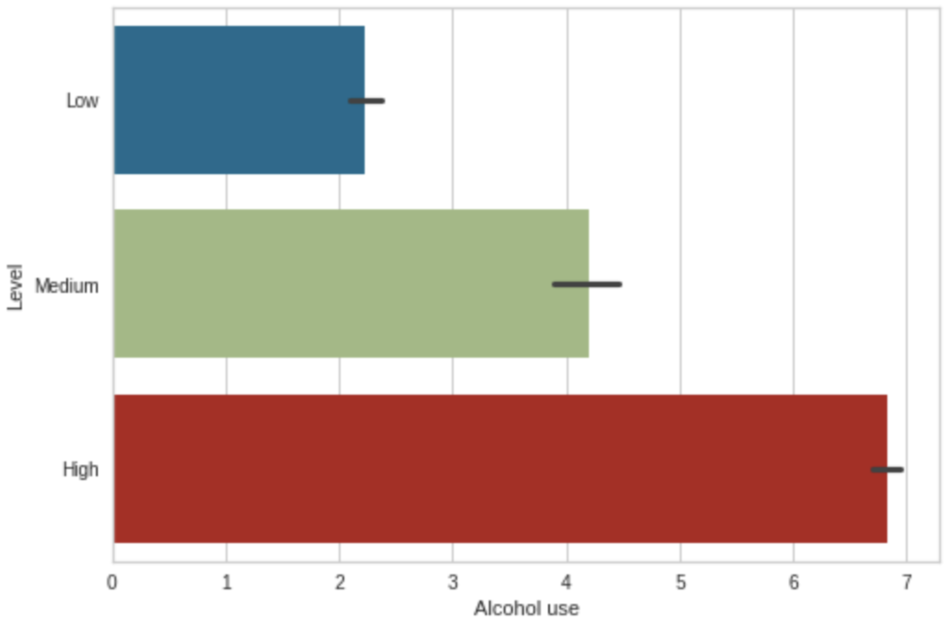

Pie Chart for High Level



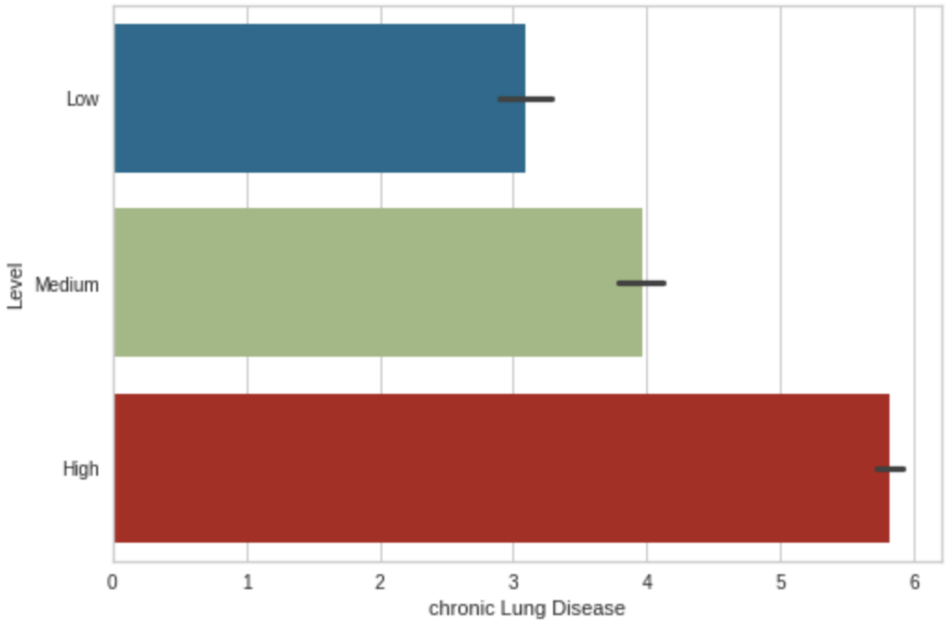
We can learn about the extent of lung cancer through air pollution. We can see that air pollution can lead to higher incidence of lung cancer than other levels.



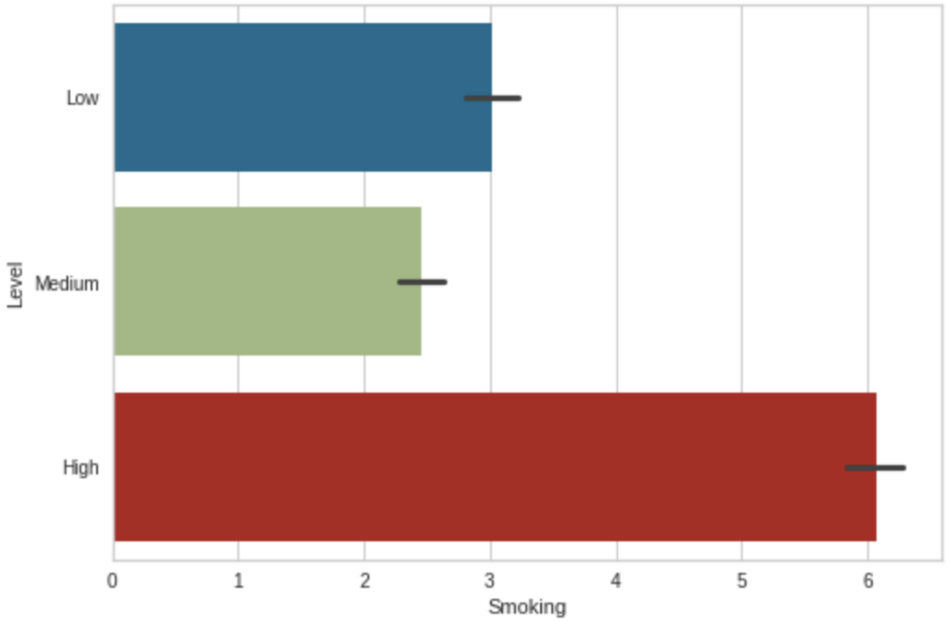
We can understand the extent of lung cancer by drinking alcohol. We can see that drinking alcohol leads to higher rates of lung cancer than other levels.



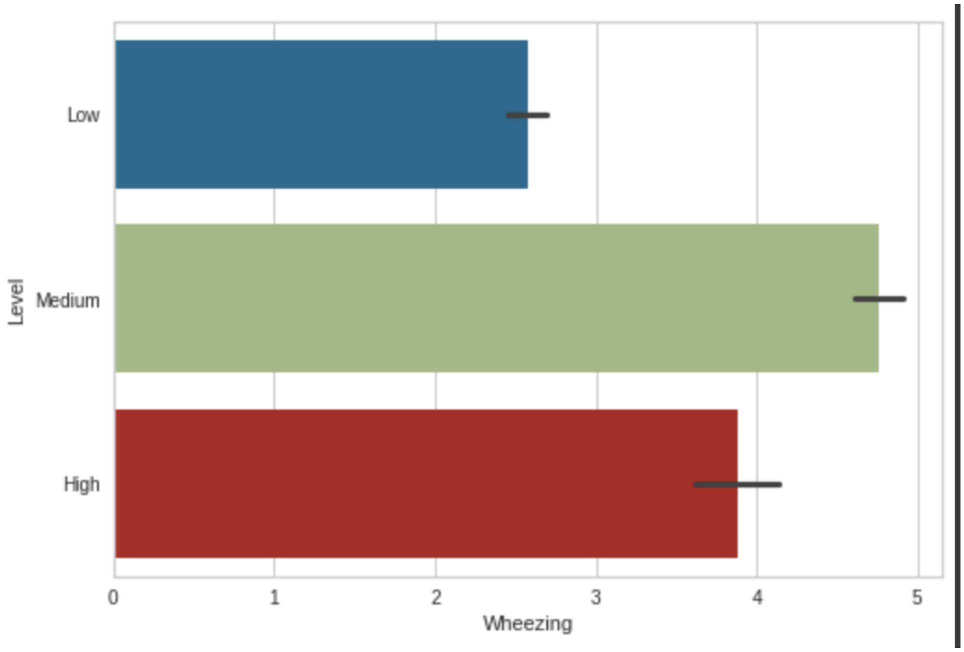
We can understand the extent of lung cancer through chronic lung disease. We can see that the incidence of lung cancer due to chronic lung disease is higher than other levels.



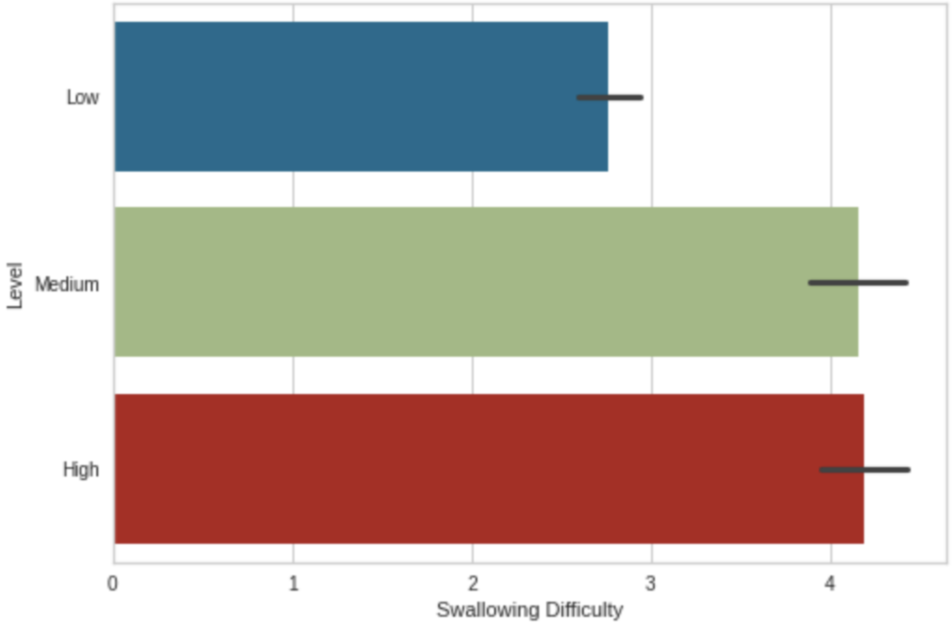
We can learn about the extent of lung cancer through smoking. We can see that the incidence of lung cancer caused by smoking is higher than other levels.



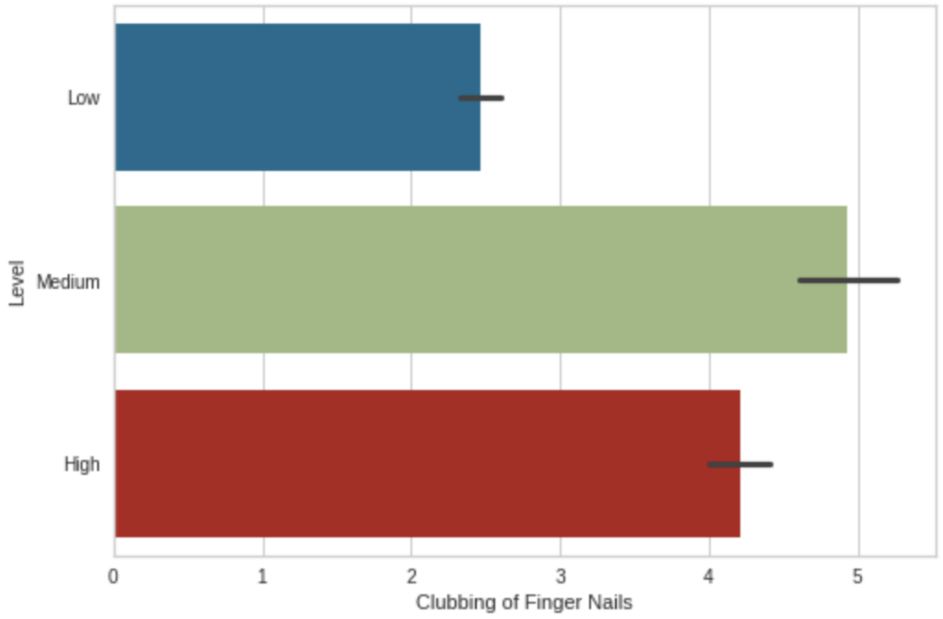
We can understand the extent of lung cancer by wheezing. It can be seen that the incidence of lung cancer caused by wheezing is at a moderate level.



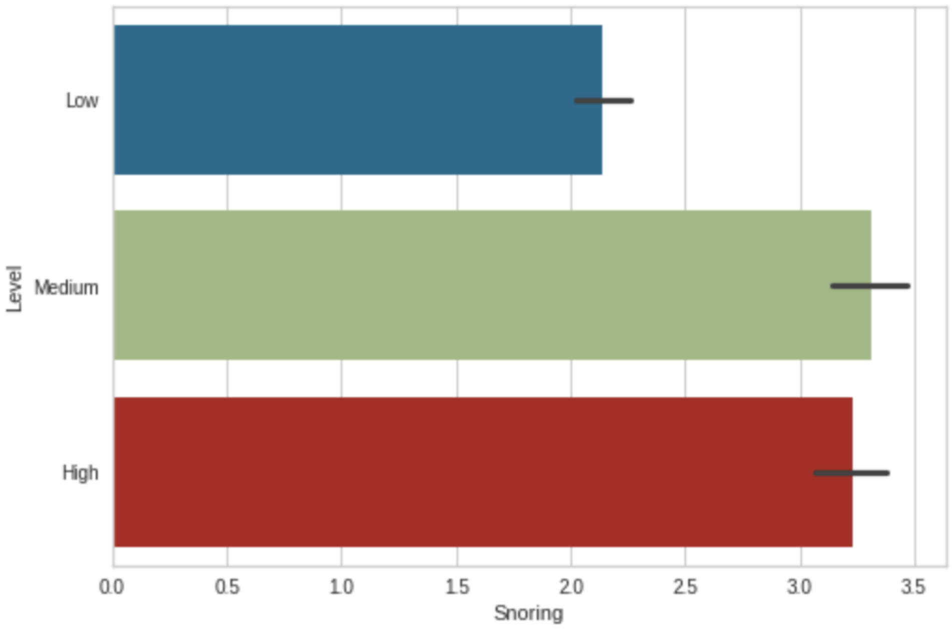
We can understand the extent of lung cancer by dysphagia. It can be seen that the incidence of lung cancer caused by dysphagia of high and medium are similar

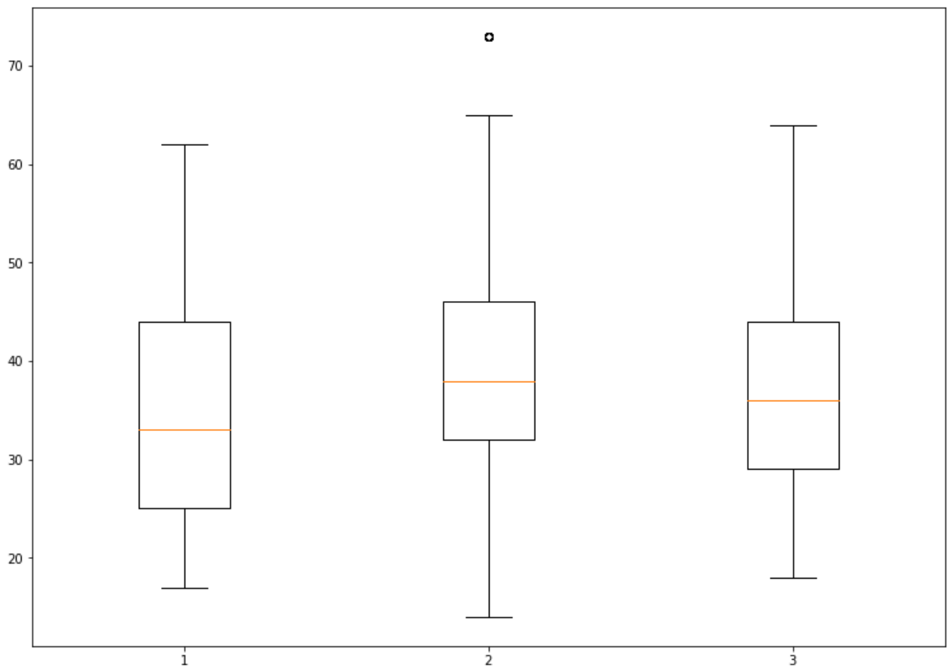


We can get a sense of the extent of lung cancer by looking at the clubbing. Visible moderate nail clubbing has the highest incidence of lung cancer



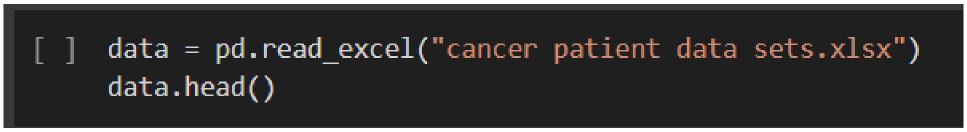
We can understand the extent of lung cancer by snoring. It can be seen that the incidence of lung cancer caused by snoring of high and medium are similar.




70
60
50
40
30
20
1
2
3

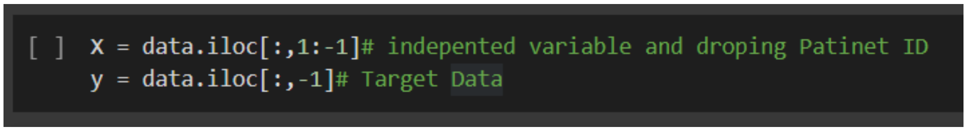
Data Preprocessing pipeline

Importing the dataset



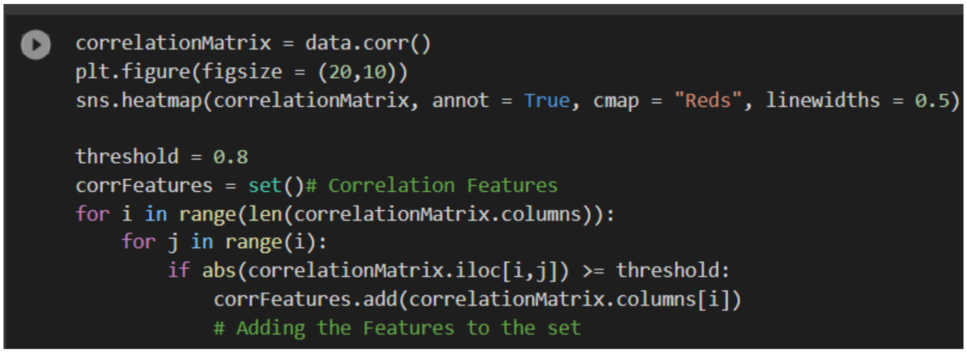
```
[ ]  data = pd.read_excel("cancer patient data sets.xlsx")
     data.head()
```

Separating the target value from the independent variable



```
[ ]  X = data.iloc[:,1:-1]# independented variable and droping Patinet ID
     y = data.iloc[:,-1]# Target Data
```

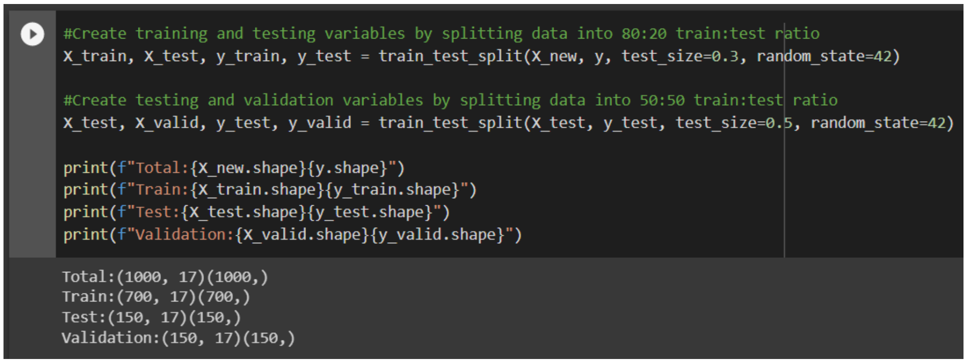
Finding Correlation between columns



```
correlationMatrix = data.corr()
plt.figure(figsize = (20,10))
sns.heatmap(correlationMatrix, annot = True, cmap = "Reds", linewidths = 0.5)

threshold = 0.8
corrFeatures = set()# Correlation Features
for i in range(len(correlationMatrix.columns)):
    for j in range(i):
        if abs(correlationMatrix.iloc[i,j]) >= threshold:
            corrFeatures.add(correlationMatrix.columns[i])
            # Adding the Features to the set
```

Splitting Data into train and test



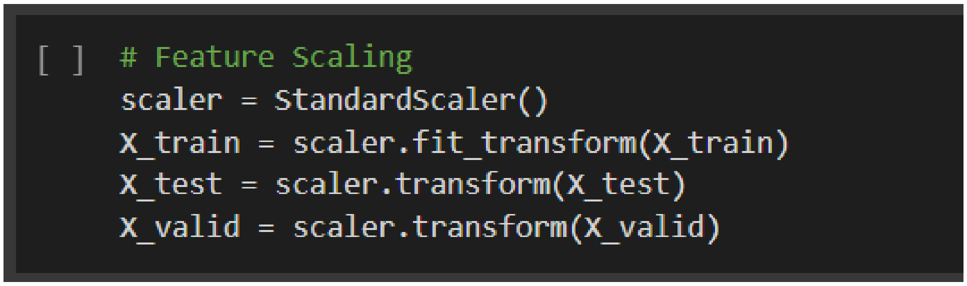
```
#Create training and testing variables by splitting data into 80:20 train:test ratio
X_train, X_test, y_train, y_test = train_test_split(X_new, y, test_size=0.3, random_state=42)

#Create testing and validation variables by splitting data into 50:50 train:test ratio
X_test, X_valid, y_test, y_valid = train_test_split(X_test, y_test, test_size=0.5, random_state=42)

print(f"Total:{X_new.shape}{y.shape}")
print(f"Train:{X_train.shape}{y_train.shape}")
print(f"Test:{X_test.shape}{y_test.shape}")
print(f"Validation:{X_valid.shape}{y_valid.shape}")
```

```
Total:(1000, 17)(1000,)
Train:(700, 17)(700,)
Test:(150, 17)(150,)
Validation:(150, 17)(150,)
```

Feature Scaling



```
# Feature Scaling
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
X_valid = scaler.transform(X_valid)
```

Algorithms Evaluation

I. Support vector machines

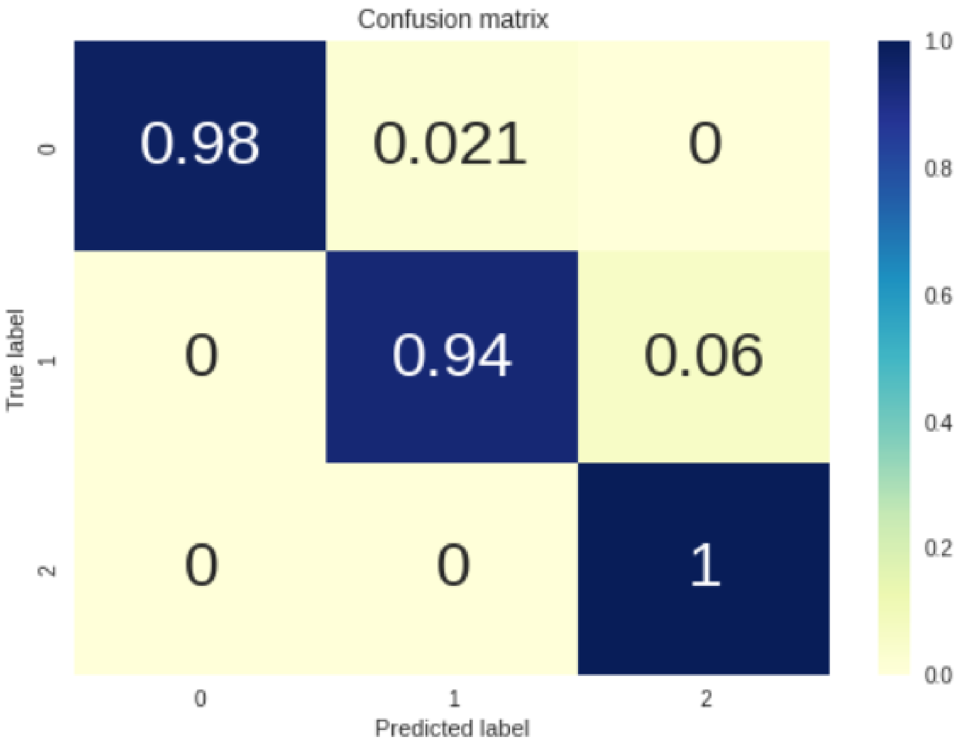
Pros

- SVM uses a small amount of memory.
- SVM is helpful in high-dimensional spaces.
- SVM predictions are quite fast.

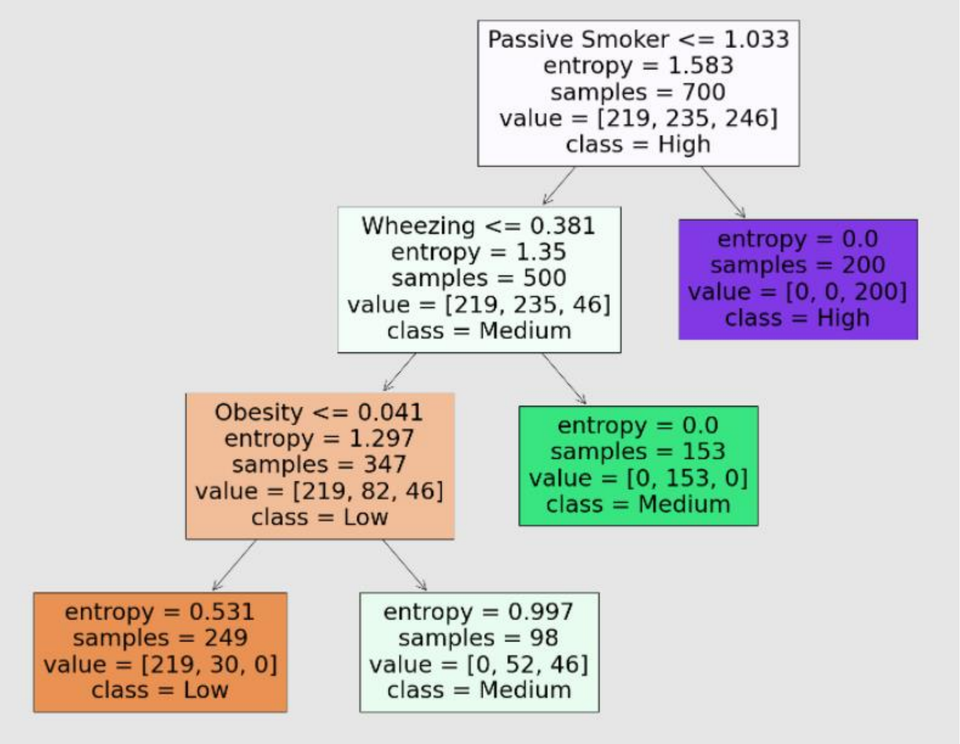
Cons

- SVM is not great for large data sets.
- When the data set has noise, it does not perform well
- SVM does not work well when features for each data point exceeds the number of training data

➢ Evaluated Results – 96%



II. Decision trees



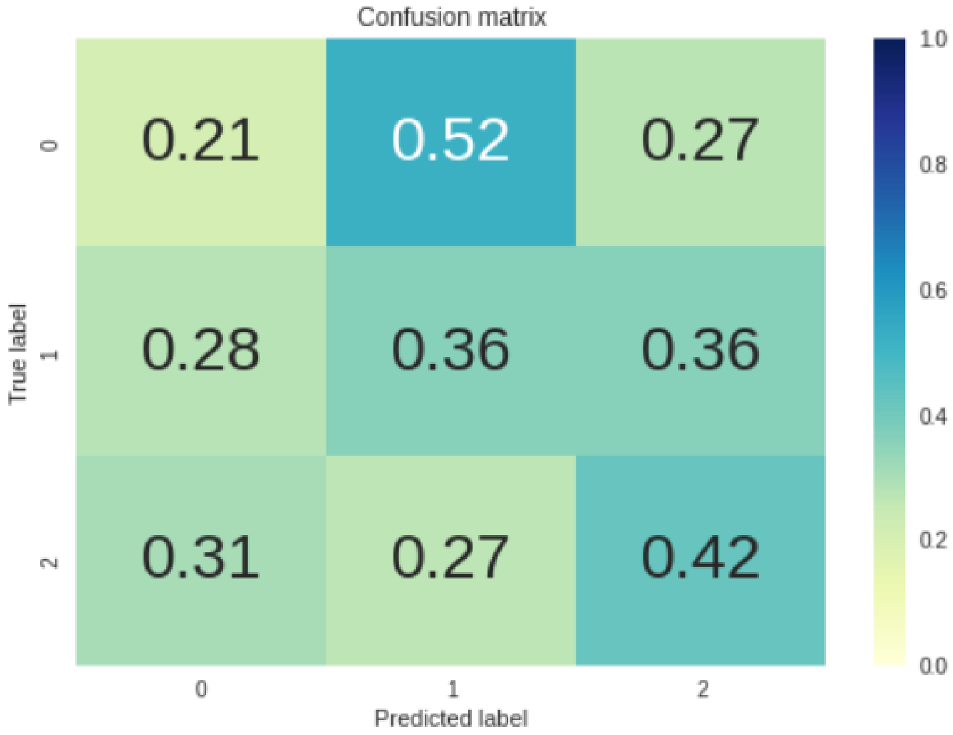
Pros

- Decision trees need less effort for data preparation during pre-processing than the other algorithms.
- Decision tree does not require normalized data.
- Data does not need to be scaled when using a decision tree
- Even if the values are missing data does not affect the process of building a decision tree.

Cons

- Even if a slight change is made to the data, it can cause a momentous change in the structure of the decision tree causing stability.
- Calculations can be more complex than other algorithms.
- Requires more time to train the model.
- More expensive in training the decision tree.

➢ Evaluated Results – 88%



III. Logistic regression

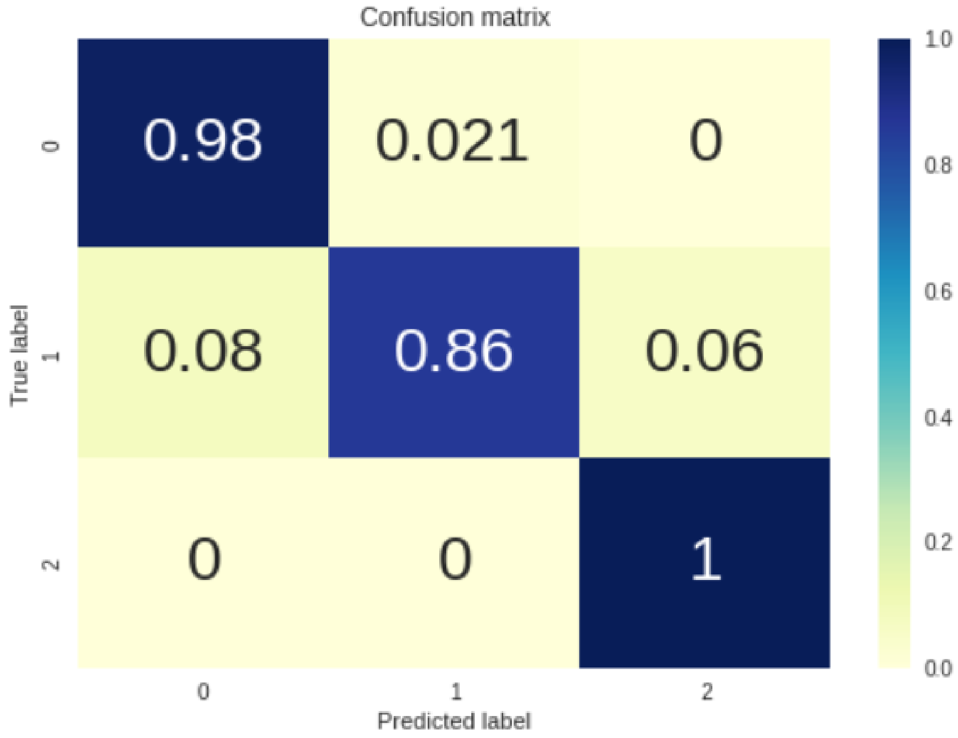
Pros

- Logistic regression is easier to implement.
- Logistic regression is fast at classifying unknown data.
- Logistic regression has particularly good accuracy for simple datasets when the dataset is linearly separable.
- Can extend multiple classes.

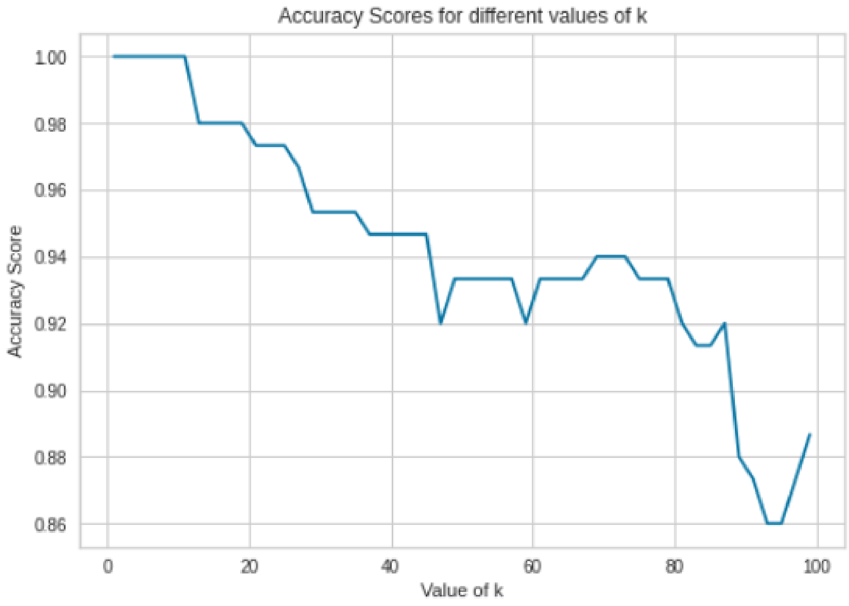
Cons

- If observations are less than the number of features Logistic regression should not be used.
- Difficult to capture complex relationships using logistic regression.
- Logistic regression fails in complex relationships.
- Logistic regression requires a large dataset and needs sufficient training for all the classes and categories.

➢ Evaluated Results – 95%



IV. K-nearest neighbors



- With the increment of k value, we can see the accuracy is decreasing right after the k =12.

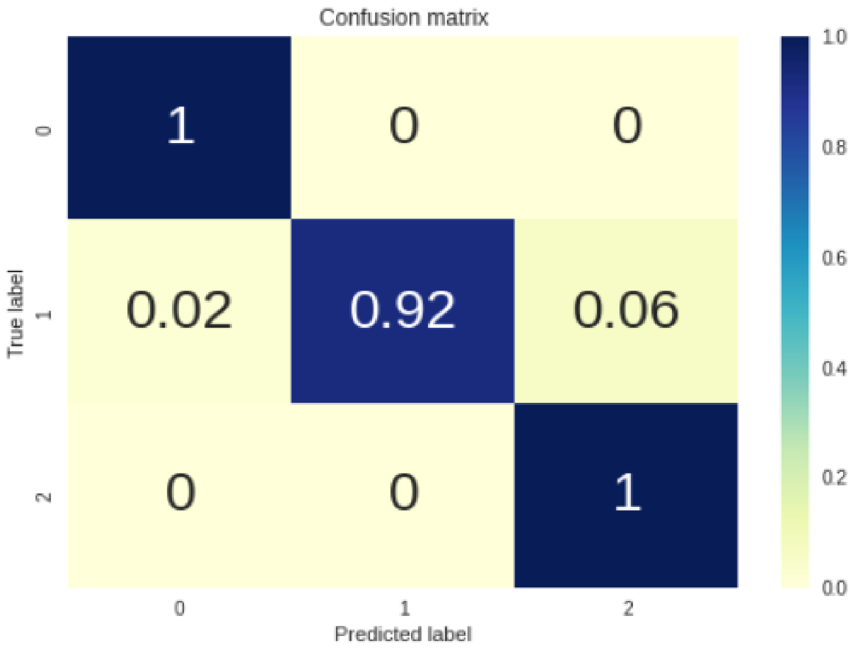
Pros

- The K-NN algorithm is remarkably simple and easy to implement.
- K-NN can be used in both classification and regression problems.
- Without having any effect new data can be added.
- Does not need training period since it sores the training data set and learns only when only making real-time predictions.

Cons

- If the datasets increase the speed of the algorithm decreases.
- Does not perform well on unbalanced data.
- If variables increase KNN struggles to predict the output
- Large datasets have a high prediction complexity.

- Predicted accuracy is 98%



V. Naïve Bayes

Pros

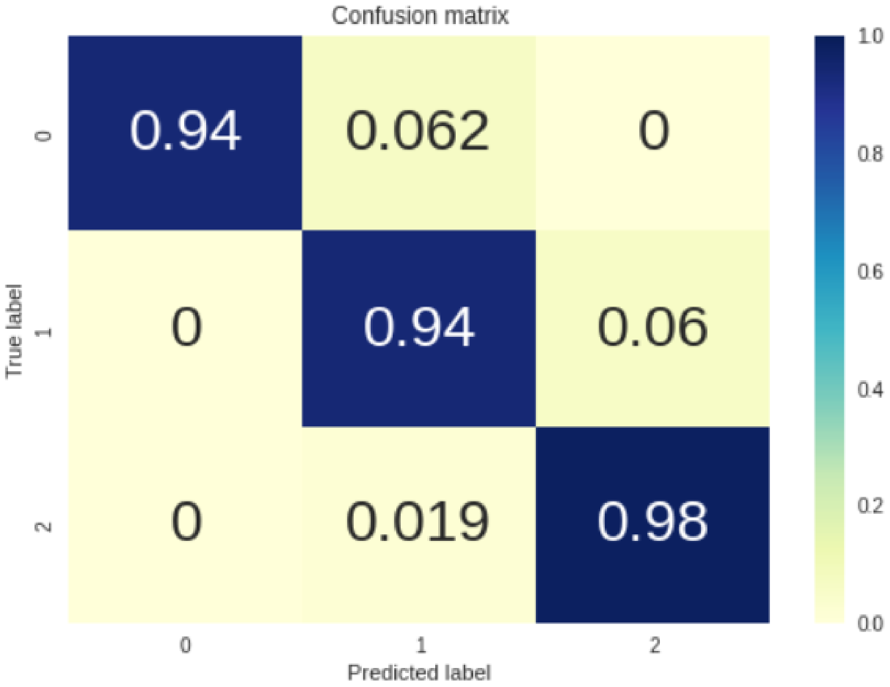
- Naïve Bayes is easy to implement
- Naïve Bayes only requires a small amount of time training data to estimate the test data.

- In comparison to numerical data, the Naive Bayes method works remarkably well with categorical input variables.

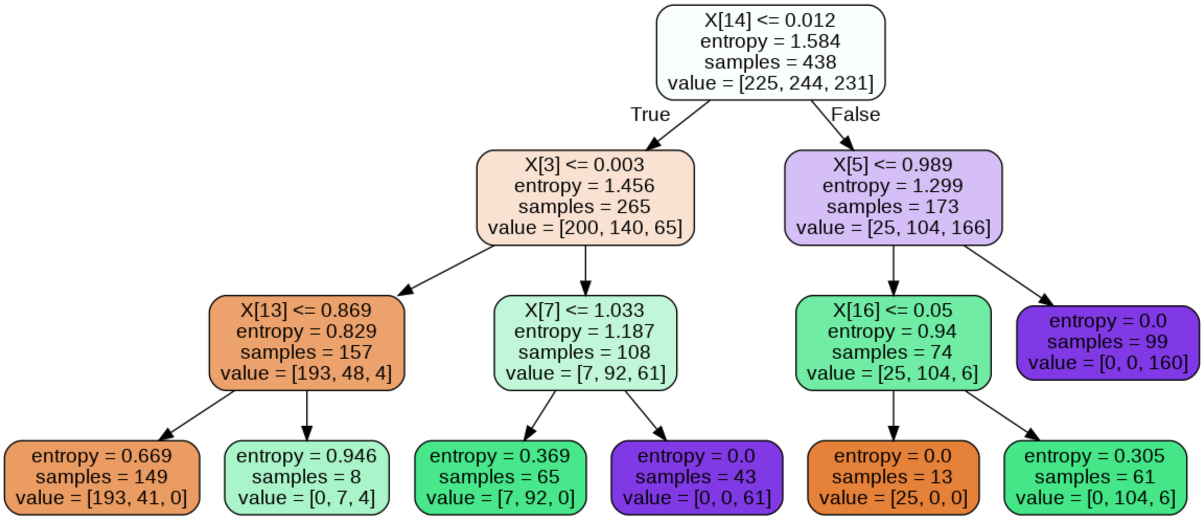
Cons

- Naïve Bayes assumes that all the predictors are independent because of this it limits the real-life use cases.
- Sometimes the estimations can be wrong so as a result, its probability outputs should not be taken seriously.
- Naïve Bayes assigns zero probability to variables where the test dataset is not available.

➢ Evaluated Results – 94%



VI. Random forest



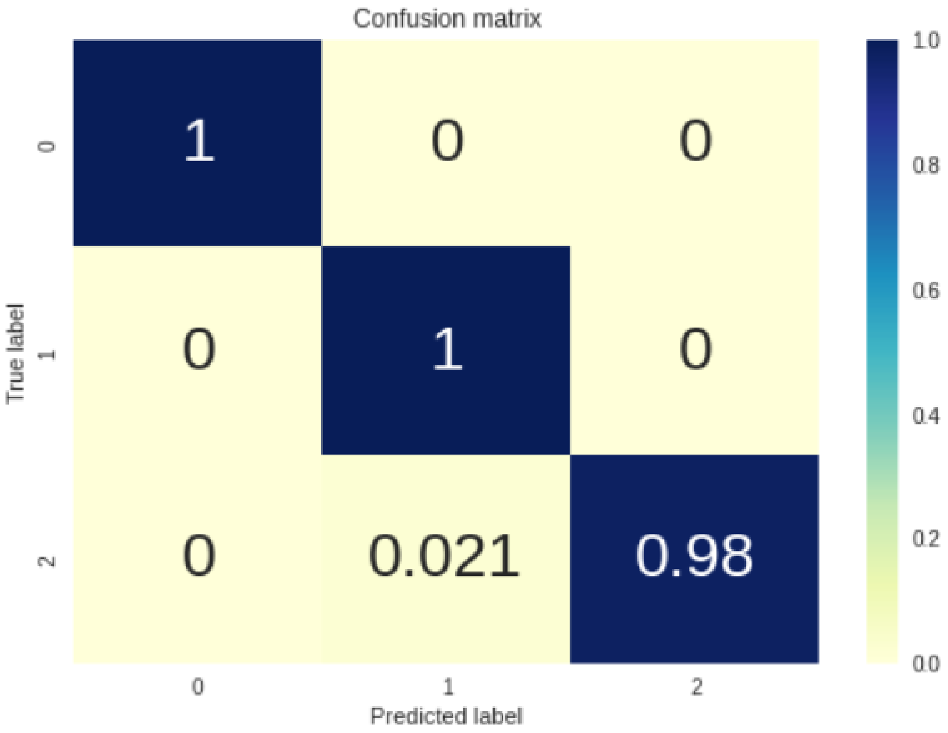
Pros

- Random forest can solve both classification and regression.
- Random forests can handle large data sets with higher dimensionality.
- Random forests have a strategy for guessing missing data that is accurate even when a big percentage of the data is missing.
- Random forests have a strategy for balancing errors in data set where classes are imbalanced.

Cons

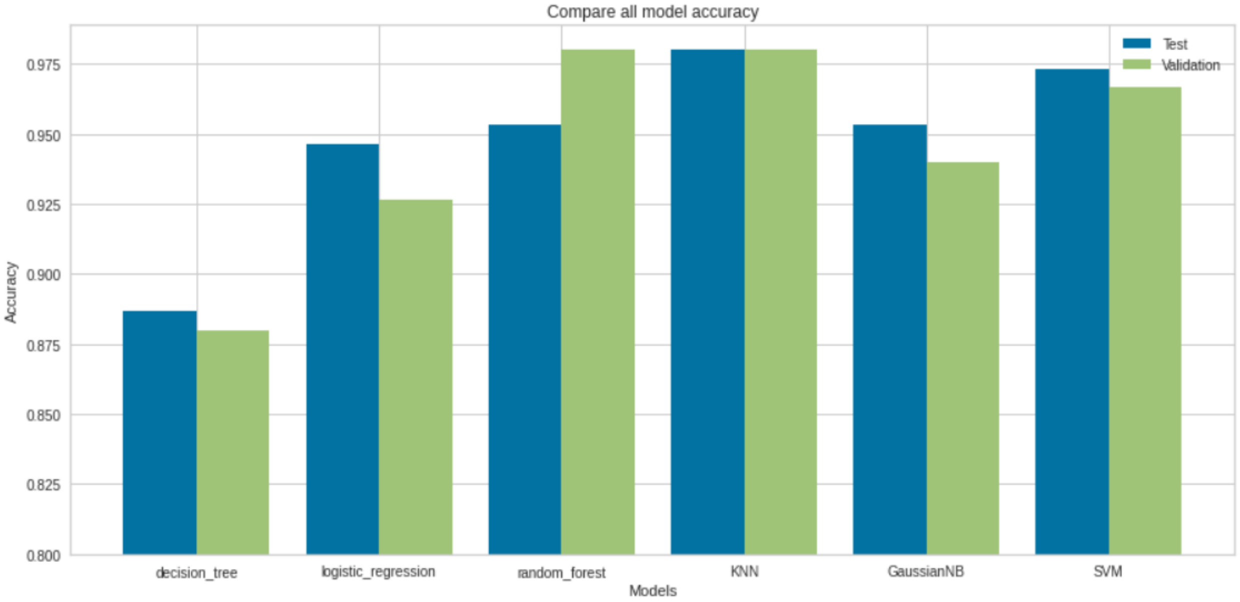
- No good at regression problems since it does not provide a precise prediction.
- If the tress of random forest is large it can make the algorithm slow.

➢ Evaluated Results – 98%


Confusion matrix
True label
Predicted label
0
1
2
0
1
2
1
0
0
0
1
0
0
0.021
0.98
1.0
0.8
0.6
0.4
0.2
0.0

Candidate Algorithm Selection and Rationale

The 3 Algorithms that work best for our dataset are SVM, K-Nearest Neighbor and Random Forest Algorithm.



- SVM is better than most of the other algorithms used as it has better accuracy in results. SVM works relatively well when there is a clear margin of separation between classes.

- The KNN algorithm can compete with the most accurate models because it makes highly accurate predictions. The algorithm is suitable for applications for which sufficient domain knowledge is available.

- The Random Forest Algorithm is the best choice to avoid increasing generalization errors when overfitting of the model occurs. It makes multiple decision trees to predict thereby having a more accurate result than the decision tree.

Model		Accuracy	Precision	Recall
Decision Tree	Validation	0.88	0.88	0.90
	Test	0.88	0.89	0.89
Random Forest	Validation	0.98	0.98	0.98

	Test	0.98	0.97	0.97
Bayesian	Validation	0.94	0.94	0.93
	Test	0.94	0.95	0.95
SVM	Validation	0.97	0.97	0.96
	Test	0.96	0.97	0.97
Logistic Regression	Validation	0.93	0.92	0.92
	Test	0.95	0.95	0.94
KNN	Validation	0.98	0.98	0.98
	Test	0.98	0.97	0.97

Conclusion

On this report, we have provided an overview of six machine learning approaches and architecture used for lung cancer prediction.

From this overview, we drew inferences; one of which was, varying different features on our data categorized under diagnostic risk factors (age, gender, alcohol use, air pollution, balanced diet, obesity, smoking, passive smoker) and symptoms (fatigue, weight loss, shortness of breath, wheezing, swallowing difficulty, clubbing of finger nails, frequent cold, dry cough, snoring) that indicates presence of cancer while building the models; we inferred that those that has the symptom features prior to diagnosis had the highest chance of being diagnosed with high level of cancer.

Also, with sufficient training data, a good level of prediction was achieved using optimized Random Forest, KNN and SVM models out of the six models. These algorithms achieved accuracies above 95% in classifying the level of cancer.

Furthermore, given an acceptable level of performance on the training data, we tried generalization of the models by testing them on unseen data. This also gave us high accuracies within the 95% region for the 3 selected best models.