Washington State University
School of Electrical Engineering and Computer Science
CptS 315 – Introduction to Data Mining
Online

Ananth Jillepalli

Course Project Report

Name: Nam Jun Lee

TABLE OF CONTENTS

Introduction 3

Data Mining Task 5

Technical Approach 6

Evaluation Methodology 12

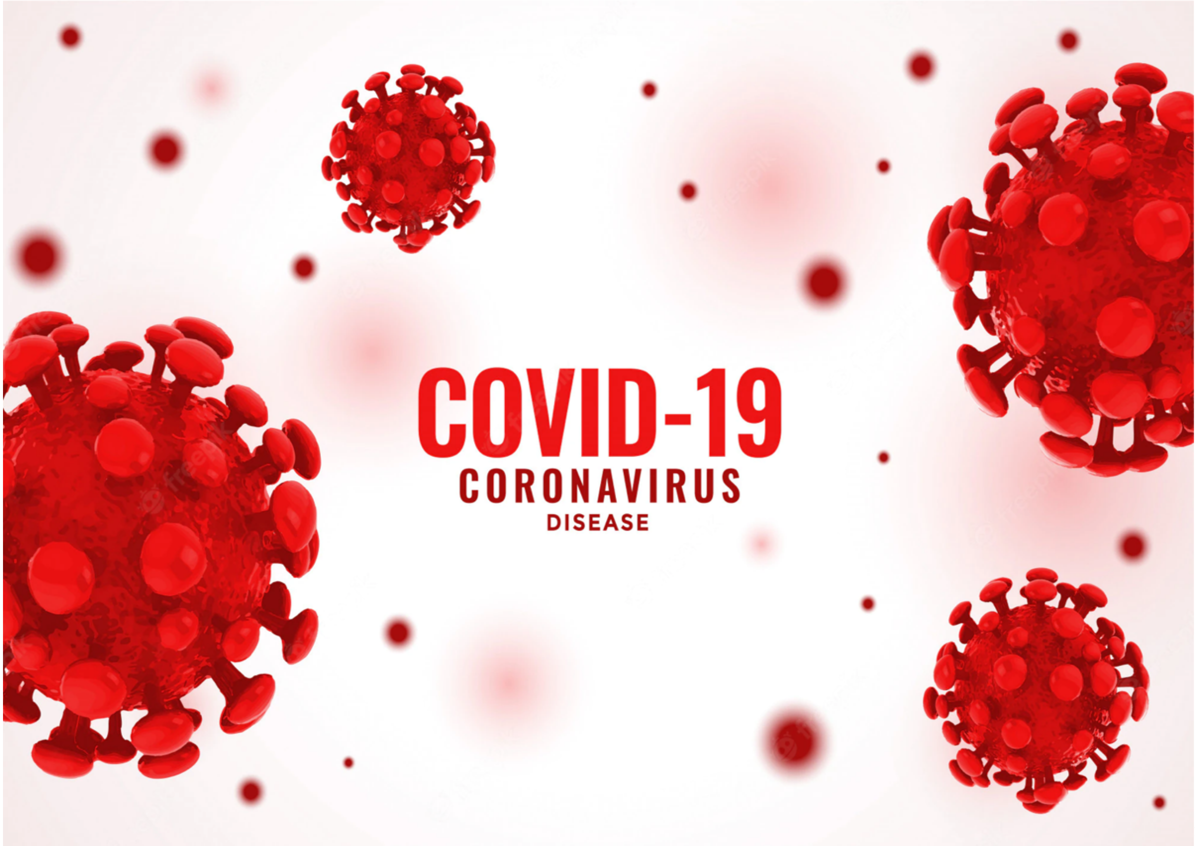
Results and Discussion 14

Lessons Learned 16

Acknowledgments..16

Bibliography 17

Classification of COVID-19



[1]

Introduction

This project aims to develop and evaluate a classification model that can predict whether it is COVID-19 or not through COVID-19 symptoms. Currently, these classifiers are actually applied among many companies. One motivation for the actual application of these predictive classifiers is the detection of fraud in banks. In the banking industry, it is very important to identify people's creditworthiness and lifestyle patterns to determine whether they use money correctly or not as a business that provides customer-centered services through funds. Therefore, based on past fraudulent or non-fraudulent transaction data and machine learning classification models, it can be predicted whether the given credit card will result in fraudulent transactions or not.[2] Based on this, I decided that the primary purpose of my

[1] *Free vector: Covid19 coronavirus red virus cell spread background concept*. Freepik. (2020, April 14) . Retrieved from https://www.freepik.com/free-vector/covid19-coronavirus-red-virus-cell-spread-background-concept_7643643.htm#query=covid&position=35&from_view=keyword

[2] Kumar, A. (2022, May 23). *Classification problems real-life examples*. Data Analytics. Retrieved from https://vitalflux.com/classification-problems-real-world-examples/

project is classification model learning.

The questions I set to achieve the objectives of this project are as follows:

1. What classification model will be used?
2. Is there data that is not needed prior to classification?
3. Which classifier can best classify COVID-19?
4. What symptoms influence COVID-19?

Looking for answers to these set questions, I can achieve the purpose of my project.

My motivation for choosing this project is that COVID-19 is currently one of the serious diseases around the world, and many researchers are trying to solve it. Just as many researchers use artificial intelligence to tell you what symptoms appear when you have COVID-19, I wanted to use machine learning for these world-famous events, so I decided to proceed with this COVID-19 prediction classification project. To proceed with this project, I will apply three classifier models (Logistic Regression, Decision Tree, Support Vector Machine) to evaluate the accuracy of each model, find out what the best classifier is, and then show which particular item has a significant influence on COVID-19.

Briefly speaking, the results of my project show that the Decision Tree classifier has the best accuracy of 98.47% of the three classifiers, and from checking the importance of variables, "Sore throat", "Breathing Problem" and "Aboard travel" has a significant impact on COVID-19.

Data Mining Task

First, I focused on identifying and understanding the details of the dataset used to proceed with this project. This plays a very important role in fitting the model. If unnecessary information or missing values exist in the dataset, this reduces the accuracy of the model. Therefore, all data mining questions set up to investigate this project must be solved sequentially.

Set Data Mining Questions:

1. Is there any problem with this data that the form fits the model?
2. Is there unnecessary data or missing values? How do you plan to confirm this?
3. What models are appropriate to use in the Scikit-learn Library?
4. How will you evaluate the performance of the fitted models?
5. How do we determine whether each variable is a factor that affects the outcome?

The input data used in this project are symptoms and external activities that appear when people have COVID-19. Through these input data, the result of the data mining approach is to classify whether or not there is a corona infection.

In order to solve the data mining questions set by me sequentially, there are nine major challenges.

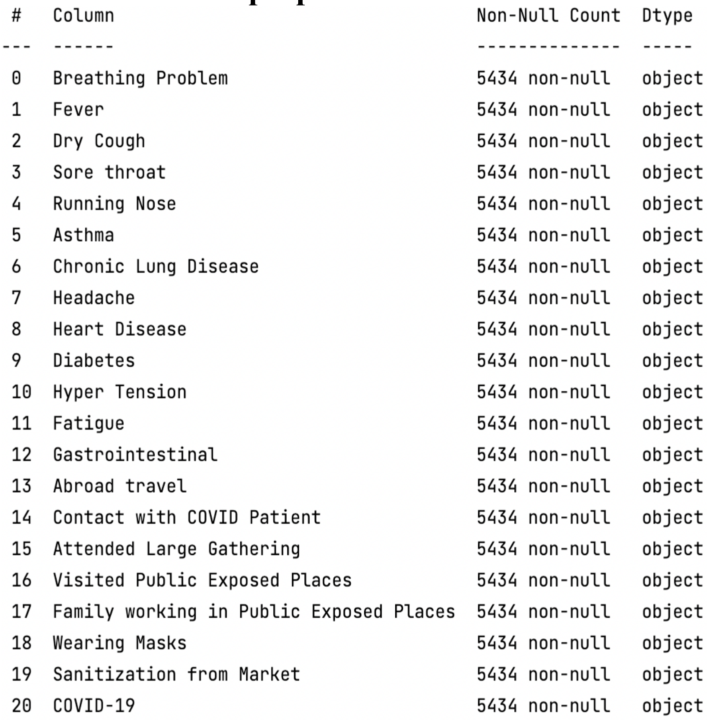
Key Challenges:

1. Check the properties of the data
2. Check correlation between data and remove unnecessary values
3. Check data for missing values
4. Data Segmentation
5. Check Logistic Regression classifier fit and predictive assessment indicators
6. Check Support Vector Machine classifier suitability and predictive evaluation indicators (Setting Linear Kernel)
7. Confirmation of conformity and predictive evaluation indicators of Decision Tree classifiers
8. Compare the accuracy of each of the three classifiers to find the best model
9. Determine the importance of each variable through an optimal model

Technical Approach

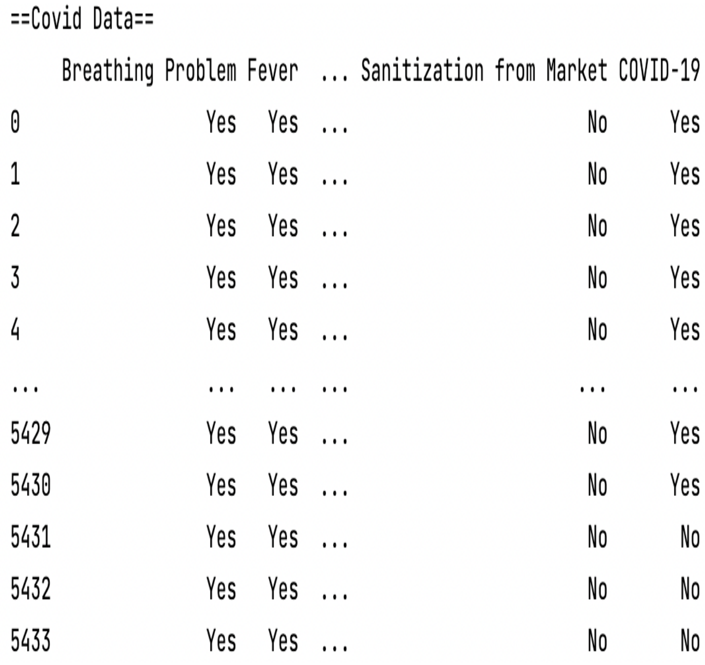
First, to solve the data mining task I set up, I will explain in detail the rest of the main tasks mentioned above except for the evaluation part.

1. **Check the properties of the Covid-19 dataset:**



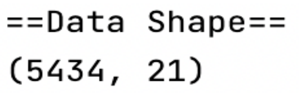
```
 #   Column                                   Non-Null Count  Dtype
---  ------                                   --------------  -----
 0   Breathing Problem                        5434 non-null   object
 1   Fever                                    5434 non-null   object
 2   Dry Cough                                5434 non-null   object
 3   Sore throat                              5434 non-null   object
 4   Running Nose                             5434 non-null   object
 5   Asthma                                   5434 non-null   object
 6   Chronic Lung Disease                     5434 non-null   object
 7   Headache                                 5434 non-null   object
 8   Heart Disease                            5434 non-null   object
 9   Diabetes                                 5434 non-null   object
 10  Hyper Tension                            5434 non-null   object
 11  Fatigue                                  5434 non-null   object
 12  Gastrointestinal                         5434 non-null   object
 13  Abroad travel                            5434 non-null   object
 14  Contact with COVID Patient               5434 non-null   object
 15  Attended Large Gathering                 5434 non-null   object
 16  Visited Public Exposed Places            5434 non-null   object
 17  Family working in Public Exposed Places  5434 non-null   object
 18  Wearing Masks                            5434 non-null   object
 19  Sanitization from Market                 5434 non-null   object
 20  COVID-19                                 5434 non-null   object
```

Fig 1. Data Attributes 1



```
==Covid Data==
     Breathing Problem Fever  ... Sanitization from Market COVID-19
0                  Yes   Yes  ...                       No      Yes
1                  Yes   Yes  ...                       No      Yes
2                  Yes   Yes  ...                       No      Yes
3                  Yes   Yes  ...                       No      Yes
4                  Yes   Yes  ...                       No      Yes
...                ...   ...  ...                      ...      ...
5429               Yes   Yes  ...                       No      Yes
5430               Yes   Yes  ...                       No      Yes
5431               Yes   Yes  ...                       No       No
5432               Yes   Yes  ...                       No       No
5433               Yes   Yes  ...                       No       No
```

Fig 2. Data Type 1



```
==Data Shape==
(5434, 21)
```

Fig 3. Data Shape 1

As shown Fig 1 to 3, this dataset has 5434 columns and 21 rows, all of which are categorical variables. In this process, it can be seen that all data have two categorical characteristics, Yes or No.

2. Check correlation between data and remove unnecessary values

First, since all of these datasets have categorical characteristics, the types of all variables were converted into integers and then worked to confirm the correlation. Detailed information can be found in the Evaluation Methodology section.

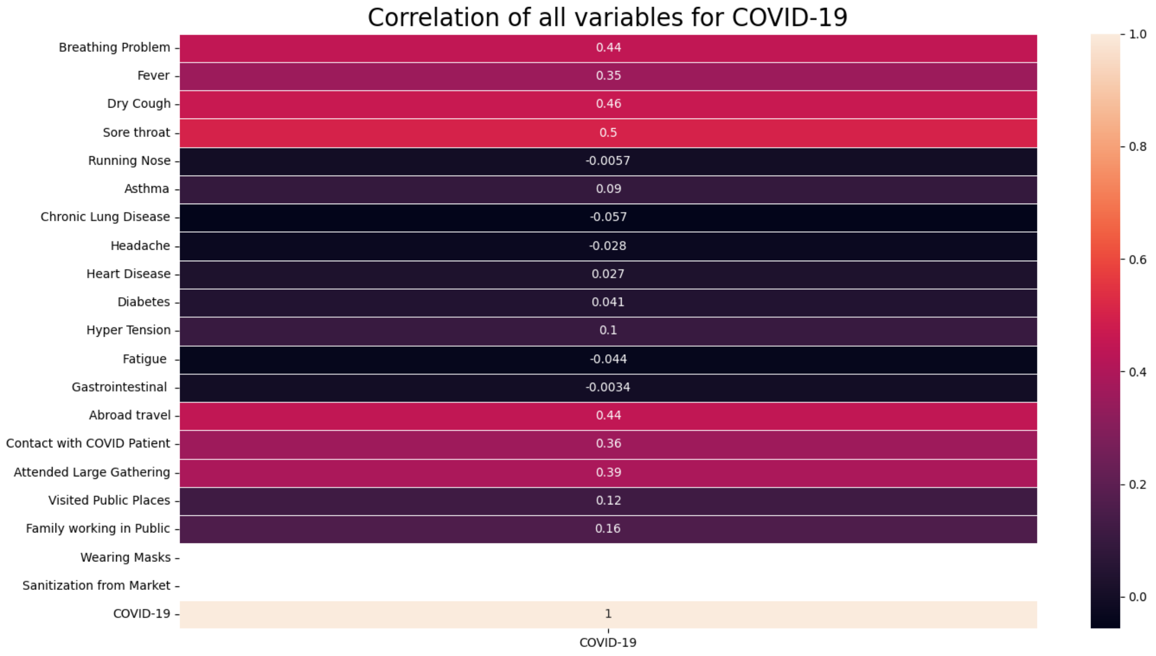
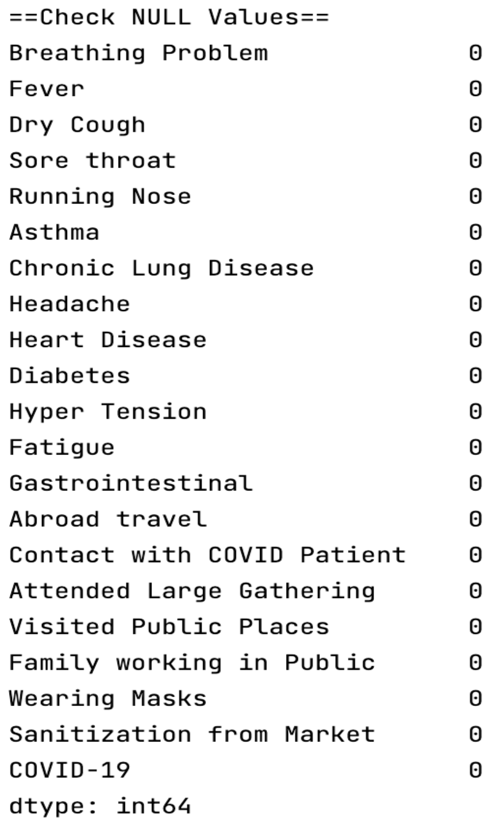


Fig 4. Correlation Plot 1

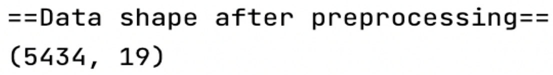
As a result of checking Fig 4 above, which created the correlation heat map of each variable in COVID-19, it can be seen that "Wearing Masks" and "Sanitization from Market" have no effect on COVID-19. Therefore, I decided to remove only those two variables, which have no influence at all, because deleting too much input data, alth ough it does not have a significant impact except for a few other symptoms, can lea d to underfitting.

3. Check data for missing values



```
==Check NULL Values==
Breathing Problem             0
Fever                         0
Dry Cough                     0
Sore throat                   0
Running Nose                  0
Asthma                        0
Chronic Lung Disease          0
Headache                      0
Heart Disease                 0
Diabetes                      0
Hyper Tension                 0
Fatigue                       0
Gastrointestinal              0
Abroad travel                 0
Contact with COVID Patient    0
Attended Large Gathering      0
Visited Public Places         0
Family working in Public      0
Wearing Masks                 0
Sanitization from Market      0
COVID-19                      0
dtype: int64
```

Fig 5. NULL Check 1

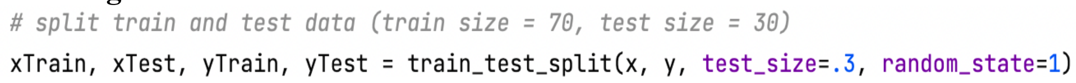


```
==Data shape after preprocessing==
(5434, 19)
```

Fig 6. Data Shape After Convert 1

According to Fig 5 and 6, it can be confirmed that there are no missing values of all variables, and the data will be used for total model learning are columns 5434 and 19 rows.

4. Data Segmentation



```
# split train and test data (train size = 70, test size = 30)
xTrain, xTest, yTrain, yTest = train_test_split(x, y, test_size=.3, random_state=1)
```

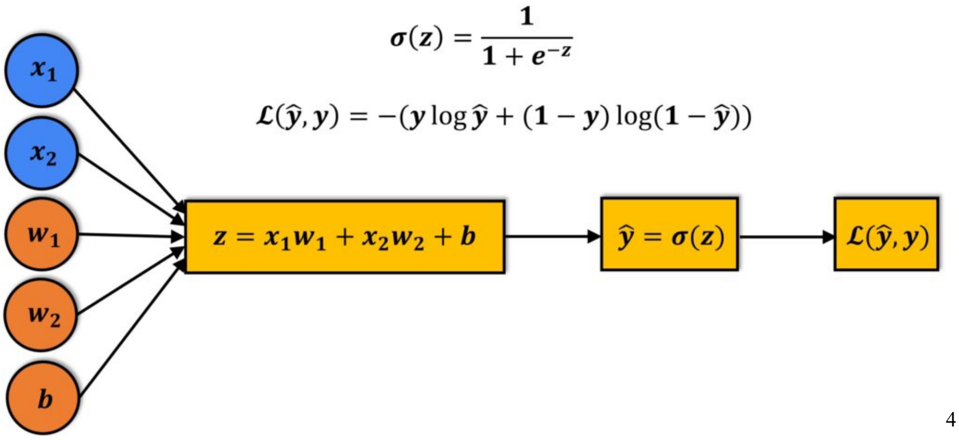
Fig 7. Split Train and Test 1

Prior to modeling, input variables were applied to x and COVID-19 was set as an independent variable to separate training data and test data. In addition, the train data were divided into 70% and the rest into test data using the train_test_split function built into the Scikit-learn. Details of this function can be found in Sklearn train_test_split[3] and applied based on this.

[3] *Sklearn.model_selection.train_test_split*. scikit. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

5. Check Logistic Regression classifier fit and predictive assessment indicators

Logistic Regression operation principle


$\sigma(z) = \frac{1}{1+e^{-z}}$
$\mathcal{L}(\hat{y}, y) = -(y \log \hat{y} + (1-y)\log(1-\hat{y}))$
x_1
x_2
w_1
w_2
b
$z = x_1 w_1 + x_2 w_2 + b$
$\hat{y} = \sigma(z)$
$\mathcal{L}(\hat{y}, y)$

[4]

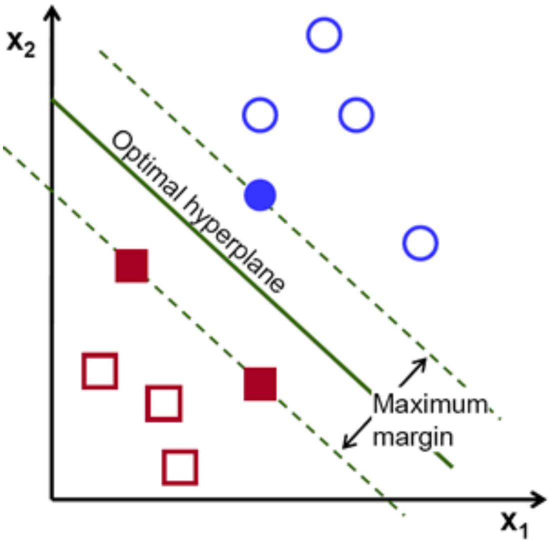
Used to measure the performance of a classification model whose output is a probability value using a cross-entropy loss function. This logistic regression model binary classification and multi-classification, which can be used to solve two problems, shows that the output data value of my project is a suitable model for use given that it consists of two categories. So, I used the LogisticRegression[5] library that is built in to Scikit-learn.

[4] Bonthu, H. (2021, July 11). *An introduction to logistic regression*. Analytics Vidhya. Retrieved from https://www.analyticsvidhya.com/blog/2021/07/an-introduction-to-logistic-regression/

[5] *Sklearn.linear_model.logisticregression*. scikit. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

6. Check Support Vector Machine classifier suitability and predictive evaluation indicators (Setting Linear Kernel)

Support Vector Machine operation principle



[6]

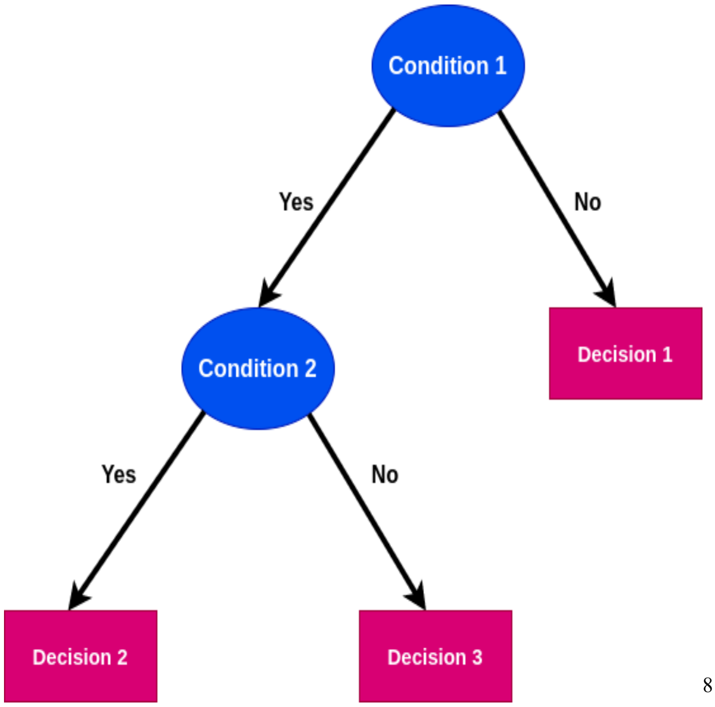
The SVM can measure performance by first finding a dividing line between the two classes and then checking the hyperplane where each input data belongs. So, I used the SVC[7] library built into the Scikit-learn and applied the most commonly used linear kernel.

[6] *Introduction to support Vector Machines*. OpenCV. (n.d.). Retrieved from https://docs.opencv.org/3.4/d1/d73/tutorial_introduction_to_svm.html

[7] *Sklearn.svm.SVC*. scikit. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

7. Confirmation of conformity and predictive evaluation indicators of Decision Tree classifiers

Decision Tree operation principle



[8]

Decision Tree algorithms correspond to supervised learning and are an analysis method that combines patterns existing between data into predictable rules and charts them into decision tree structures, as shown in the figure above. So, I used DecisionTreeClassifier[9] library built into the Scikit-learn.

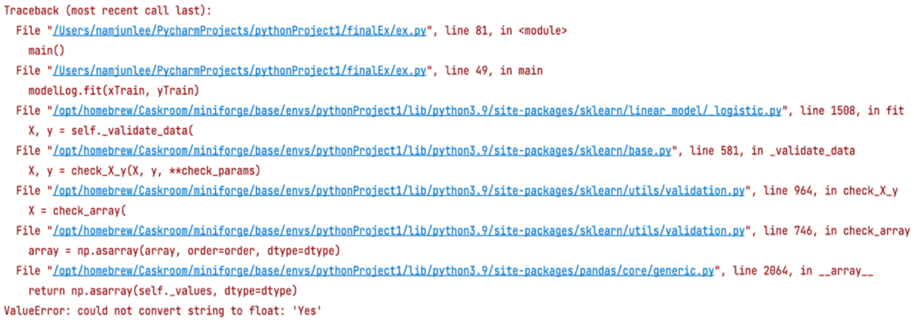
Of the major tasks mentioned, the evaluation methods used in 5-9 can be found in the evaluation methodology section below.

[8] Roy, A. (2020, November 6). *A dive into decision trees*. Medium. Retrieved from https://towardsdatascience.com/a-dive-into-decision-trees-a128923c9298

[9] *Sklearn.tree.decisiontreeclassifier*. scikit. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

Evaluation Methodology

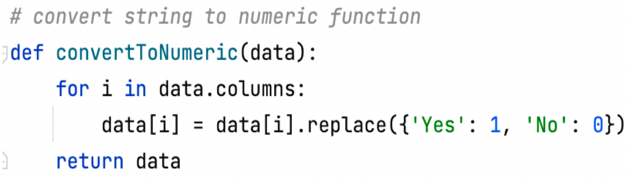
The data sets used to carry out this project are from Kaggle. This dataset includes the presence of various symptoms and whether people are infected with COVID-19. This data set has a total of 5434 columns and 21 rows, each with two string data types: Yes or No.
In the process of fitting this data to the model, the problem occurred as shown in Fig 8 below.



```
Traceback (most recent call last):
  File "/Users/namjunlee/PycharmProjects/pythonProject1/finalEx/ex.py", line 81, in <module>
    main()
  File "/Users/namjunlee/PycharmProjects/pythonProject1/finalEx/ex.py", line 49, in main
    modelLog.fit(xTrain, yTrain)
  File "/opt/homebrew/Caskroom/miniforge/base/envs/pythonProject1/lib/python3.9/site-packages/sklearn/linear_model/_logistic.py", line 1508, in fit
    X, y = self._validate_data(
  File "/opt/homebrew/Caskroom/miniforge/base/envs/pythonProject1/lib/python3.9/site-packages/sklearn/base.py", line 581, in _validate_data
    X, y = check_X_y(X, y, **check_params)
  File "/opt/homebrew/Caskroom/miniforge/base/envs/pythonProject1/lib/python3.9/site-packages/sklearn/utils/validation.py", line 964, in check_X_y
    X = check_array(
  File "/opt/homebrew/Caskroom/miniforge/base/envs/pythonProject1/lib/python3.9/site-packages/sklearn/utils/validation.py", line 746, in check_array
    array = np.asarray(array, order=order, dtype=dtype)
  File "/opt/homebrew/Caskroom/miniforge/base/envs/pythonProject1/lib/python3.9/site-packages/pandas/core/generic.py", line 2064, in __array__
    return np.asarray(self._values, dtype=dtype)
ValueError: could not convert string to float: 'Yes'
```

Fig 8. Error Record 1

Many machine learning algorithms cannot operate directly from label data.
All input and output variables must be numeric.[10] So I implemented a function to change the data types of all input and output variables and applied it when performing data preprocessing tasks.



```
# convert string to numeric function
def convertToNumeric(data):
    for i in data.columns:
        data[i] = data[i].replace({'Yes': 1, 'No': 0})
    return data
```

Fig 9. Convert Function 1

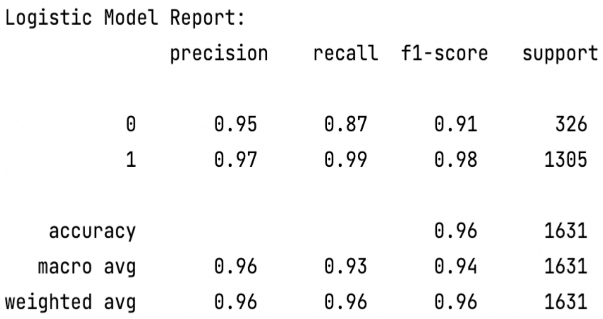
As shown in Fig 9 above, this function could solve the problem of data specifications suitable for use in models to be used in my project. Three models were then formed through separate test data and training data and then used to evaluate the performance of each model using the classification_report library provided by Scikit-learn. However, comparing the accuracy of

[10] Brownlee, J. (2017, July 28). *Why one-hot encode data in machine learning?* Machine Learning Mastery. Retrieved from https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/

each of the three models, all of them had similar accuracy, so it was found that the decision tree was the optimal model using the accuracy_score library provided by Scikit-learn, which shows the accuracy in more detail. As shown in Fig 9 above, this function could solve the problem of data specifications suitable for use in models to be used in my project. Three models were then formed through separate test data and training data and then used to evaluate the performance of each model using the classification_report library provided by Scikit-learn. However, comparing the accuracy of each of the three models, all of them had similar accuracy, so it was found that the decision tree was the optimal model using the accuracy_score library provided by Scikit-learn, which shows the accuracy in more detail. Here, this property returns the importance through the Gini importance.[11] Details can be found in the Results and Discussion section.

[11] *Sklearn.tree.decisiontreeclassifier*. scikit. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

Results and Discussion

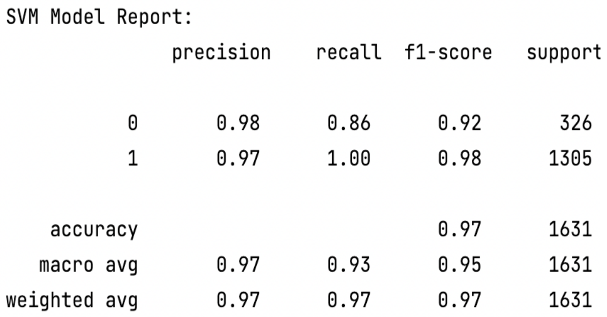


```
Logistic Model Report:
              precision    recall  f1-score   support

           0       0.95      0.87      0.91       326
           1       0.97      0.99      0.98      1305

    accuracy                           0.96      1631
   macro avg       0.96      0.93      0.94      1631
weighted avg       0.96      0.96      0.96      1631
```

Fig 10. Logistic Report 1

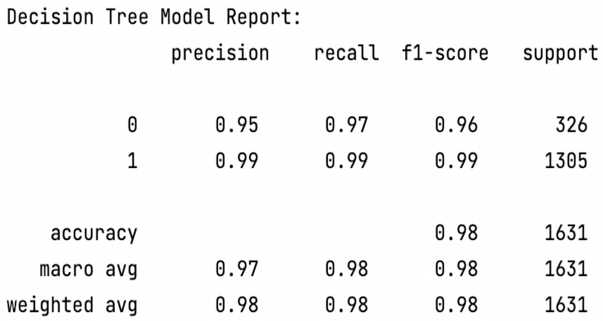


```
SVM Model Report:
              precision    recall  f1-score   support

           0       0.98      0.86      0.92       326
           1       0.97      1.00      0.98      1305

    accuracy                           0.97      1631
   macro avg       0.97      0.93      0.95      1631
weighted avg       0.97      0.97      0.97      1631
```

Fig 11. Linear Kernel SVM Report 1



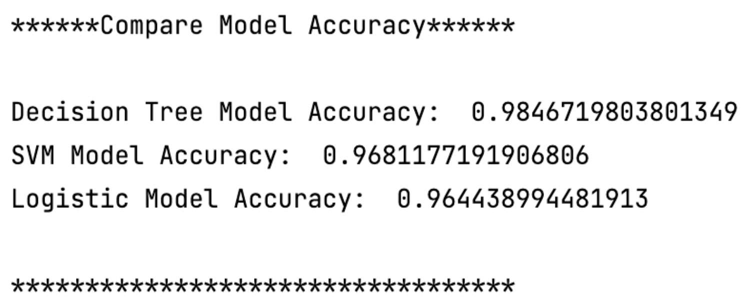
```
Decision Tree Model Report:
              precision    recall  f1-score   support

           0       0.95      0.97      0.96       326
           1       0.99      0.99      0.99      1305

    accuracy                           0.98      1631
   macro avg       0.97      0.98      0.98      1631
weighted avg       0.98      0.98      0.98      1631
```

Fig 12. Decisioin Tree Report 1

Through the predictive evaluation indicators of each model in Fig 10 to 12, it is possible to check the precision and recall rate of each model. Given that the precision and reproducibility of these three models are at least 85%, all of them are considered suitable for use. However, it can be seen that the precision and recall of the Decision Tree are relatively slightly higher than those of the two models.



```
******Compare Model Accuracy******

Decision Tree Model Accuracy:  0.9846719803801349
SVM Model Accuracy:  0.9681177191906806
Logistic Model Accuracy:  0.964438994481913

**********************************
```

Fig 13. Compare Model 1

From Figure 13 above, it can be seen that the performance accuracy of each of the three models is accurately expressed. It can seem that all three models are highly accurate prediction models with an accuracy of 96% or more, but since the Decision Tree model shows slightly higher accuracy than the two models, the Decision Tree model is the optimal model.

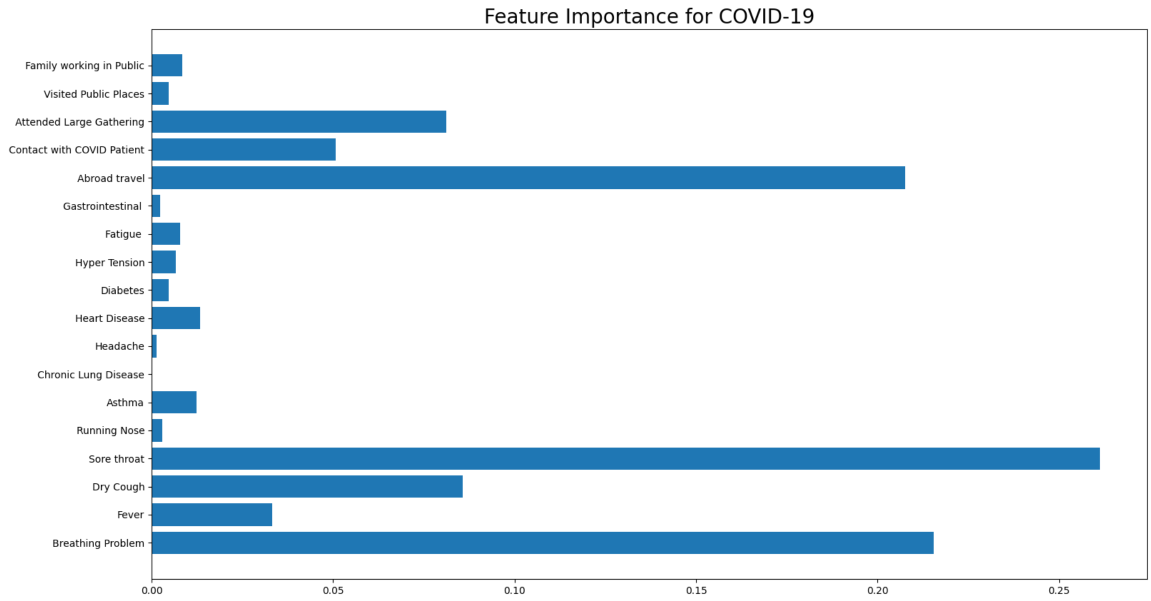


Fig 14. Feature Importance Plot 1

Finally, looking at Figure 14 above, this is the answer to the last part of the main ta sk in my project. As previously shown in Figure 13, this graph uses the most optima l model, the decision-making model, to show which symptoms are influential on COV ID-19. Through this, it can be seen that there is a high probability of being infected with COVID-19 when there are physical symptoms such as "Sore throat" and "Breathi ng Problem" and that if you have been on an "Aboard travel", you are also likely to be infected, with COVID-19. But on the contrary, It is also shown that if you have physical symptoms such as "Headache", "Fatigue" and "Running Nose", you are less l ikely to be infected with COVID-19.

Lessons Learned

Through the project implementing this COVID-19 Symptom Prediction Model, I was able to learn from visualizing data to how to use the Scikit-learn library, which is frequently used in machine learning. Currently, the use of big data and machine learning play an important role in many fields, so it was a good experience to proceed with data analysis projects in the future. And through this project, I was able to learn with certainty which algorithms are appropriate when applying predictive models for categorical output data. In addition, I could learn how to implement a more accurate model by learning that unnecessary data or missing values present in the data are influential in model accuracy.

If I think about it later, it still shows high accuracy on all three models, but it would have been better if tuning was applied to the model to further develop this project. For example, in the case of the decision tree, I think it would have been a good idea to solve the overfitting that would exist by setting the depth of the tree or hyperparameters.

Acknowledgments

To complete this project, I read so many Internet sources and used them for my project. First, I could see how the classification model I used is used in public institutions such as real banks. Also, the main source of use was Scikit-learn, where I could get a lot of help on how to use the library provided by Scikit-learn. In addition, it was also possible to know why categorical data should be converted into numerical data and then suitable for the model. It was also a good opportunity to theoretically know how each algorithm works by getting help from several sources about the Logistic Regression, Support Vector Machine, and Decision Trees that I used in this project.

Bibliography

Bonthu, H. (2021, July 11). *An introduction to logistic regression*. Analytics Vidhya. Retrieved from https://www.analyticsvidhya.com/blog/2021/07/an-introduction-to-logistic-regression/

Brownlee, J. (2017, July 28). *Why one-hot encode data in machine learning?* Machine Learning Mastery. Retrieved from https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/

Free vector: Covid19 coronavirus red virus cell spread background concept. Freepik. (2020, April 14). Retrieved from https://www.freepik.com/free-vector/covid19-coronavirus-red-virus-cell-spread-background-concept_7643643.htm#query=covid&position=35&from_view=keyword

Harikrishnan, H. (2020, August 18). *Symptoms and COVID presence (May 2020 data)*. Kaggle. Retrieved from https://www.kaggle.com/datasets/hemanthhari/symptoms-and-covid-presence

Introduction to support Vector Machines. OpenCV. (n.d.). Retrieved from https://docs.opencv.org/3.4/d1/d73/tutorial_introduction_to_svm.html

Kumar, A. (2022, May 23). *Classification problems real-life examples*. Data Analytics. Retrieved from https://vitalflux.com/classification-problems-real-world-examples/

Roy, A. (2020, November 6). *A dive into decision trees*. Medium. Retrieved from https://towardsdatascience.com/a-dive-into-decision-trees-a128923c9298

Sklearn.linear_model.logisticregression. scikit. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Sklearn.svm.SVC. scikit. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

Sklearn.tree.decisiontreeclassifier. scikit. (n.d.). Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html