Final Report
Disease prediction using Machine Learning

By Lakshika Jain

Problem Statement

One of the challenges the healthcare providers face currently is the accuracy of disease prediction. The healthcare providers are very overworked and can give very little time and attention to the patients. In this case, to accurately predict the disease ahead of time with just a few symptoms can help with treating the possible disease proactively, thus increasing the treatment outcome in patients. To intervene in time and improve the disease outcome, using Machine learning models to predict the diseases can be very helpful. This will save a lot of time for healthcare providers to accurately identify the disease and also save money by decreasing the number of tests that need to be run. In disease prediction, symptoms are very informative, they can narrow down on what the underlying issue is and sometimes symptoms can be seen very early on thus improving the treatment outcomes.

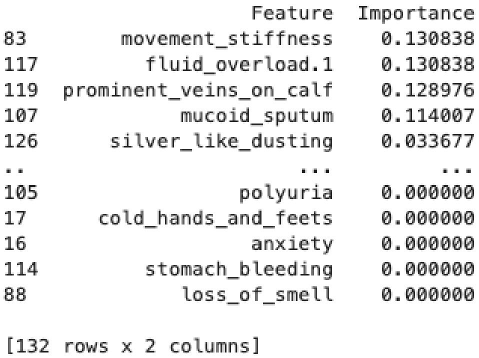
Data Wrangling

The raw test dataset from Kaggle on the disease prediction had 134 columns and 4920 rows. The columns had prognosis and 132 symptoms with 1 or 0 in the row to identify if the symptom was present or not for the prognosis. Since there were 132 symptoms/parameters to consider, reducing the dimensionality was a very important step. Thus I focussed only on the parameters that were positive for most diseases/prognosis. The most commonly occurring symptoms for the prognosis were Fatigue, vomiting, high fever, loss of appetite, nausea, headache, abdominal pain etc that occurred over 1000 times in the data. The least common symptoms that I didn’t focus on to reduce the noise were Foul smell of urine, swollen blood vessels, pus filled pimples, blackheads etc.

There was an unnamed:133 column that had nulls which I got rid of.

Exploratory Data Analysis

Since my dataset had either 0 or 1 as values, it was more like boolean data instead of numerical data which was challenging to draw any kind of statistical inference on. My approach to the exploratory data analysis for this project was to identify which symptoms were correlated with one another to narrow down on the features.
For example, itching was correlated with skin rash, nodal skin eruptions. Continuous sneezing was correlated with Shivering, chills etc.
I identified the top 5 symptoms for each prognosis and most commonly occurring symptoms across all diseases. The most commonly occuring symptoms across prognosis were:
Vomiting = 11
Itching = 7
Fatigue = 7
Skin Rash = 6
Headache = 5
I also calculated the feature importance using XGBoost,



```
                      Feature  Importance
83          movement_stiffness    0.130838
117           fluid_overload.1    0.130838
119  prominent_veins_on_calf    0.128976
107              mucoid_sputum    0.114007
126         silver_like_dusting    0.033677
..                         ...         ...
105                   polyuria    0.000000
17        cold_hands_and_feets    0.000000
16                     anxiety    0.000000
114           stomach_bleeding    0.000000
88               loss_of_smell    0.000000

[132 rows x 2 columns]
```

Modeling

I used different machine learning models to test which model is predicting the disease accurately. The models that I used were: Random forest, XG Boost, Decision Tree.

All models are performed exceptionally well, which could indicate that the dataset is well-structured for classification, with distinct patterns between the symptoms and diagnoses.

XGBoost, also had a perfect AUC score of 1.0, reinforcing that these models are highly capable of handling the dataset.
This high performance across all models may also indicate that the dataset is relatively simple for these algorithms to model, likely due to the clear separation between classes based on the given features.

RandomForest and XGBoost both achieved a perfect AUC of 1.0 on the test data. This suggests that these models are highly effective at distinguishing between the different classes (diagnoses). A perfect AUC indicates that the models made no classification errors, meaning they perfectly separate the various medical conditions based on the provided symptoms.
DecisionTree had a slightly lower AUC of 0.9936, which is still a strong performance. This indicates that the DecisionTree model was almost as effective as the ensemble models but may have struggled with a few instances where it couldn't perfectly classify the conditions. Decision trees are prone to overfitting, which might explain this slight drop in performance compared to the ensemble models.

Conclusion:

- Both RandomForest and XGBoost are highly reliable models for this dataset, with perfect classification performance on the test data.
- DecisionTree performed slightly worse but is still a strong model with near-perfect results.
- This dataset lacked diversity which could be the reason why the models fit perfectly.
- For future work, feature selection techniques could be applied to reduce the dimensionality and improve interpretability.
- Data augmentation for underrepresented classes could be considered to improve model performance on rare diagnoses.