Disease Prediction from Symptoms

Your Own Virtual Doctor!

Team Members

Yugansh Jain	**184506**
Peeyush Thakur	**184510**
Rohit Bhatia	**184518**
Tejesh Reddy	**184528**

Under The Guidance Of : Mr Gagnesh Kumar

Abstract

Disease Prediction plays an important role in health care information. It is important to diagnose the disease early. This project uses selective traits and classification strategies to diagnose and predict diseases. Adequate selection of features plays an important role in improving the accuracy of classification systems. The reduction in size helps to improve the overall performance of the machine learning algorithm. The use of differentiating algorithms in the disease database produces promising results by developing flexible, automated and intelligent diagnostic programs for diseases. Classification systems can be used to speed up the process and improve the efficiency of result statistics. This work introduces a comprehensive overview of the options for selecting various features and their pros and cons. We then analyzed the flexible classification systems and classification systems for the same stages of disease forecasting.

Contents

01 Project Idea & Objective

02 Phases of the project

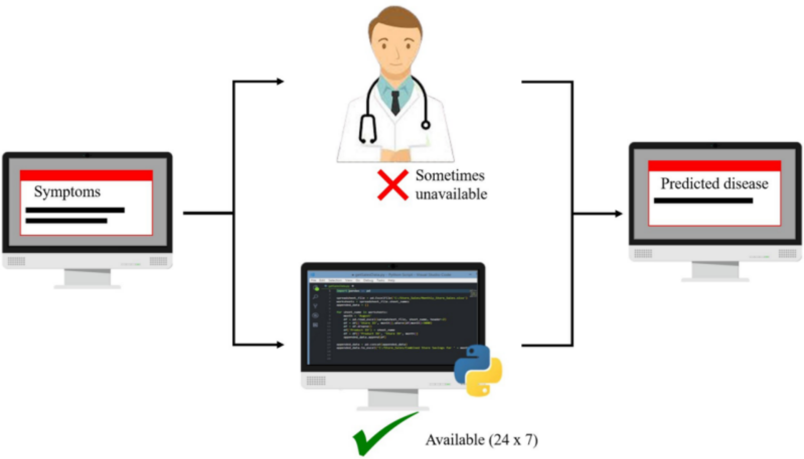
03 Tools & Tech Stacks

04 Briefing of each phase

05 Wrapping things up

Project Inspiration

Medicine and healthcare are some of the most crucial parts of the economy and human life. In this situation, where everything has turned virtual, the doctors and nurses are putting up maximum efforts to save people's lives even if they have to danger their own. Virtual doctors are board-certified doctors who choose to practice online via video and phone appointments, rather than in-person appointments but this is not possible in the case of emergency. Machines are always considered better than humans as, without any human error, they can perform tasks more efficiently and with a consistent level of accuracy.



Objective

The primary goal was to develop multiple models to define which one of them provides the most accurate predictions. While ML projects vary in scale and complexity, their general structure is the same.

A disease predictor can predict the disease of any patient without any human error. Also, in conditions like COVID-19 and EBOLA, a disease predictor can be a blessing as it can identify a human’s disease without any physical contact.

Phases of the project

01

Data Collection & Pre-Processing

The most essential stages to **acquire the fine and final data** that can be taken as correct and suitable for further data mining tasks.

02

Model Training

Determining good values for all the weights and the bias from labeled examples.

We used different classifiers to compare the results namely

- Decision Tree Classifier
- Support Vector Machine

Tools & Tech Stacks

Data Collection & Pre-processing:

- Scikit-learn
- Pandas
- Numpy
- Matplotlib

Model Training:

- Scikit-learn
- Decision Tree
- Support Vector Machine

Collaboration tools:

- Google Colab
- Google Docs
- Latex

Phase 01

Data Collection & Pre-processing

Data Collection : Data collection is the systematic approach to gathering and measuring information from a variety of sources to get a complete and accurate picture of an area of interest.

Data Pre-processing : Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format which can be further feed into a model for training.

Why Data Pre-processing is required?

Since mistakes, redundancies, missing values, and inconsistencies all compromise the integrity of the set, you need to fix all those issues for a more accurate outcome. Thus, before using that data for the purpose you want, you need it to be as organized and “clean” as possible. There are several ways to do so, depending on what kind of problem you’re tackling.

Phase 01

Data Collection & Pre-processing

Data Source : https://people.dbmi.columbia.edu/~friedma/Projects/DiseaseSymptomKB/index.html

Database based on disease-symptom associations generated by an automated method based on information in textual discharge summaries of patients at New York Presbyterian Hospital admitted during 2004.

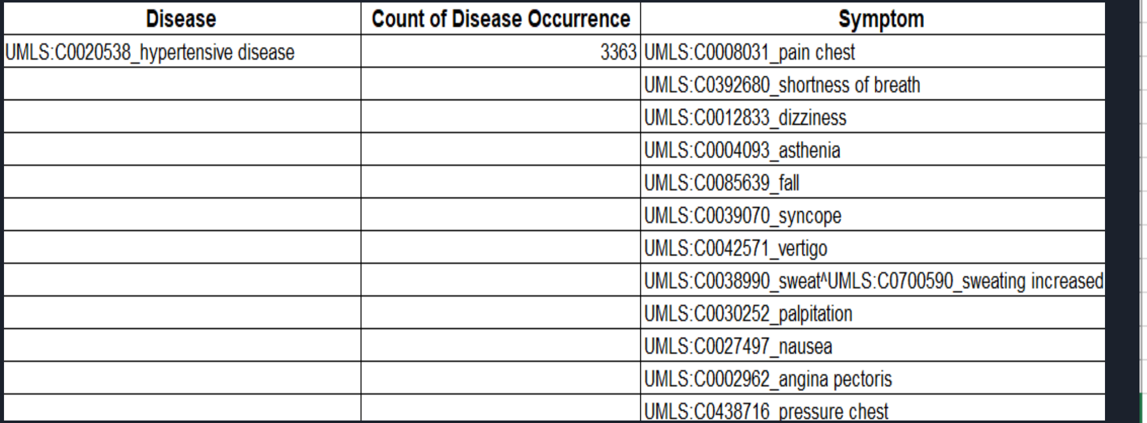
We will use this data by copying it in an excel file and using it.

We will firstly clean the data, so that we can further process the data to feed into the model.

Phase 01

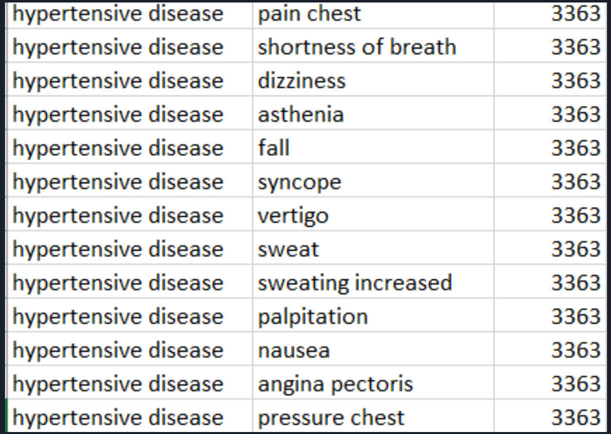
Data Images (Snippet) before and after Cleaning

Before :



Disease	Count of Disease Occurrence	Symptom
UMLS:C0020538_hypertensive disease	3363	UMLS:C0008031_pain chest
		UMLS:C0392680_shortness of breath
		UMLS:C0012833_dizziness
		UMLS:C0004093_asthenia
		UMLS:C0085639_fall
		UMLS:C0039070_syncope
		UMLS:C0042571_vertigo
		UMLS:C0038990_sweat^UMLS:C0700590_sweating increased
		UMLS:C0030252_palpitation
		UMLS:C0027497_nausea
		UMLS:C0002962_angina pectoris
		UMLS:C0438716_pressure chest

After :



hypertensive disease	pain chest	3363
hypertensive disease	shortness of breath	3363
hypertensive disease	dizziness	3363
hypertensive disease	asthenia	3363
hypertensive disease	fall	3363
hypertensive disease	syncope	3363
hypertensive disease	vertigo	3363
hypertensive disease	sweat	3363
hypertensive disease	sweating increased	3363
hypertensive disease	palpitation	3363
hypertensive disease	nausea	3363
hypertensive disease	angina pectoris	3363
hypertensive disease	pressure chest	3363

Phase 01

Encoding

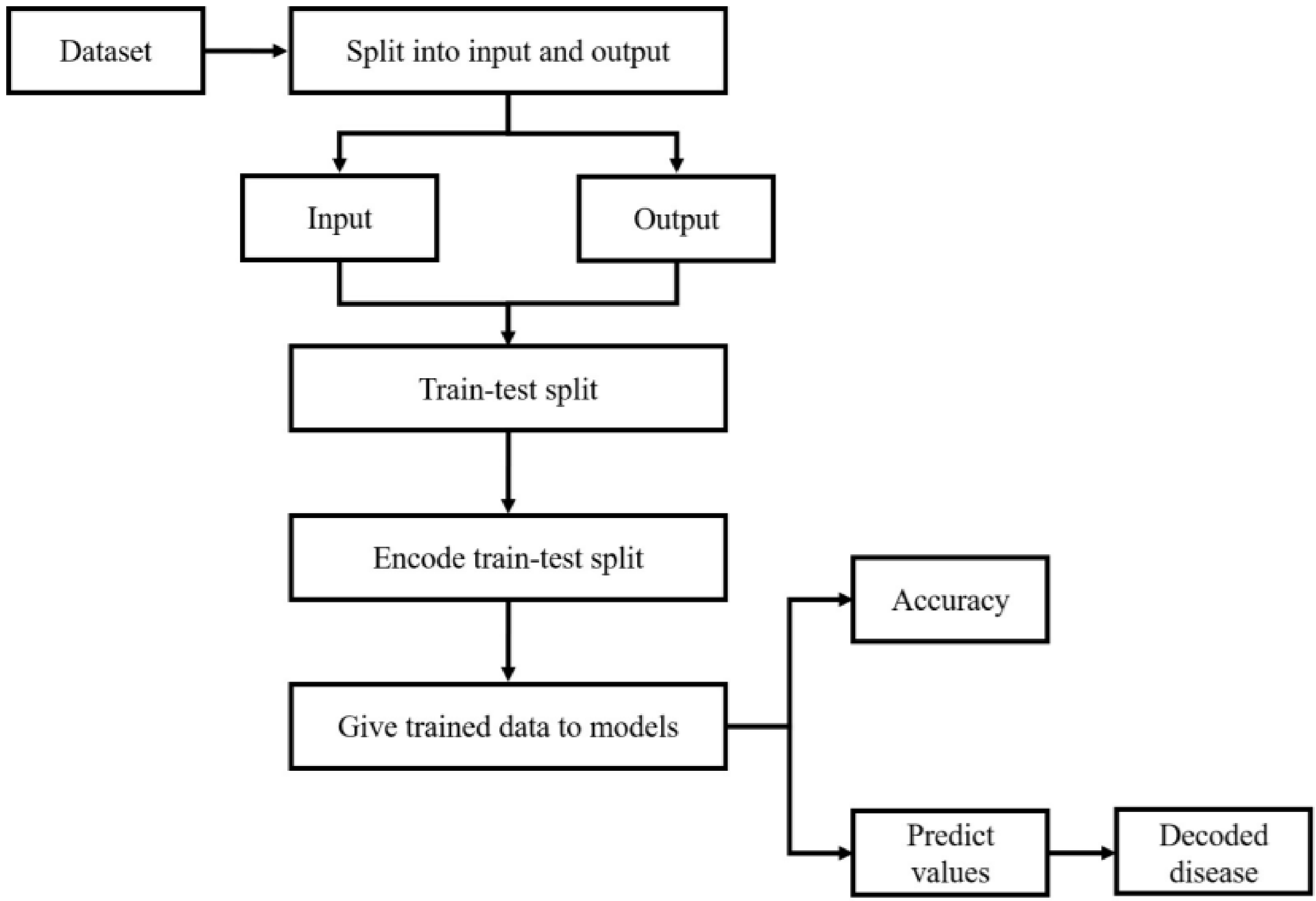
After cleaning data we will use Label Encoder and One Hot Encoder. These two encoders are parts of the SciKit Learn library in Python, and they are used to convert categorical data, or text data, into numbers, which our predictive models can better understand.

Label Encoding is a popular encoding technique for handling categorical variables. In this technique, each label is assigned a unique integer based on alphabetical ordering.

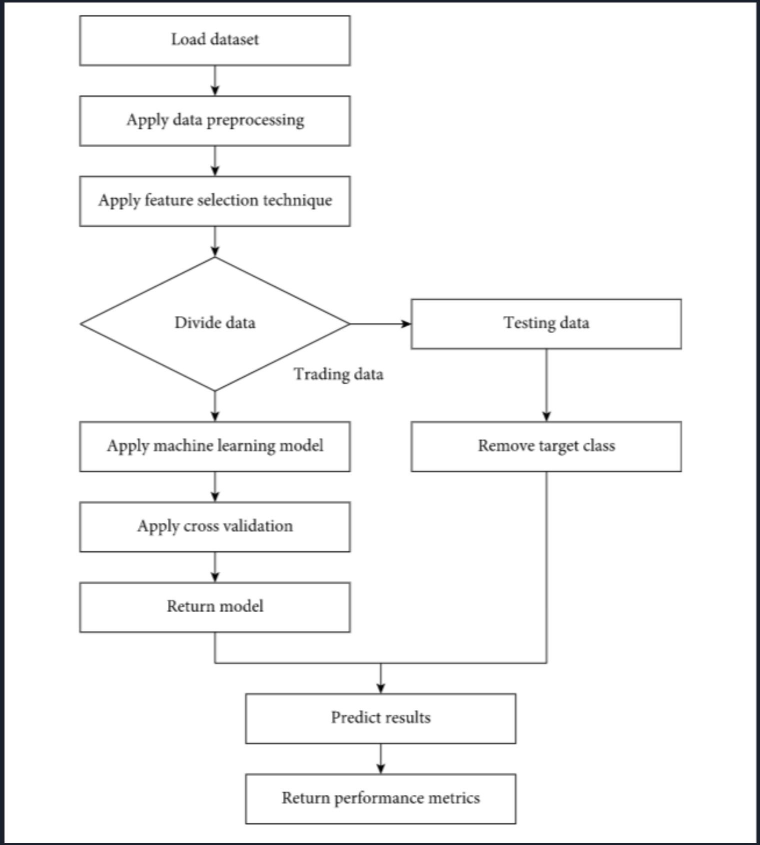
With one-hot, we convert each categorical value into a new categorical column and assign a binary value of 1 or 0 to those columns.

After, encoding our data will be ready to feed in model for training process.

Flow chart


Dataset
Split into input and output
Input
Output
Train-test split
Encode train-test split
Give trained data to models
Accuracy
Predict values
Decoded disease

Principal Aspects



Load dataset

Apply data preprocessing

Apply feature selection technique

Divide data

Testing data

Trading data

Apply machine learning model

Remove target class

Apply cross validation

Return model

Predict results

Return performance metrics

Phase 02

Model Training

How to choose the right model for the project? Things to consider

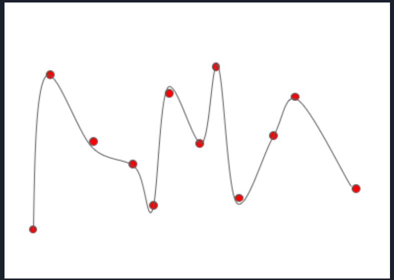
01 Size of the dataset

02 Training time

03 Number of features

04 Availability of computational time and power

Too many features can cause the model to overfit and make it imprecise.



Trying too hard to overfit!

Continued.

Phase 02

Model Training

How to tackle the problem of large number of features?

Large number of features can cause the training time of the model to increase exponentially.

Feature selection: If no statistical relationship exists between some features, drop the insignificant one.

Manual Feature Selection: Selected through data analysis.

Feature Engineering: Group certain features which have some relevance.

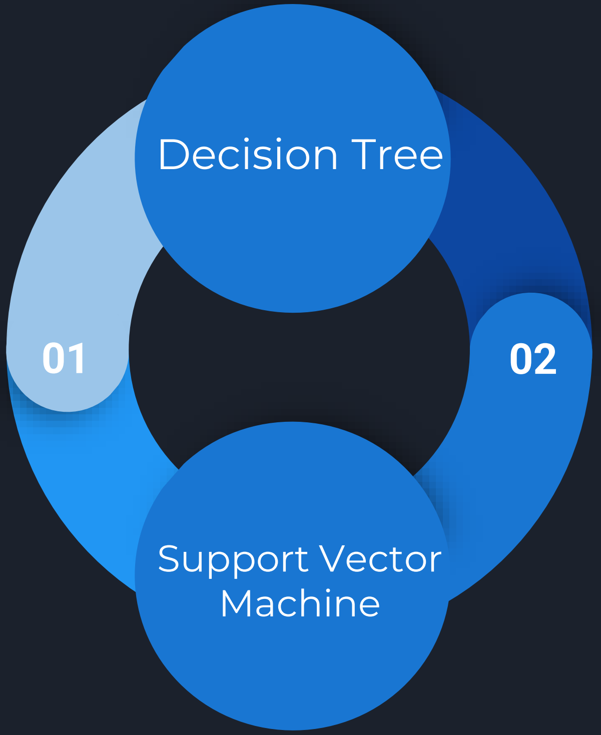
PCA: Reducing the dimensionality of our dataset.

Phase 02

Model Training

Decision Tree

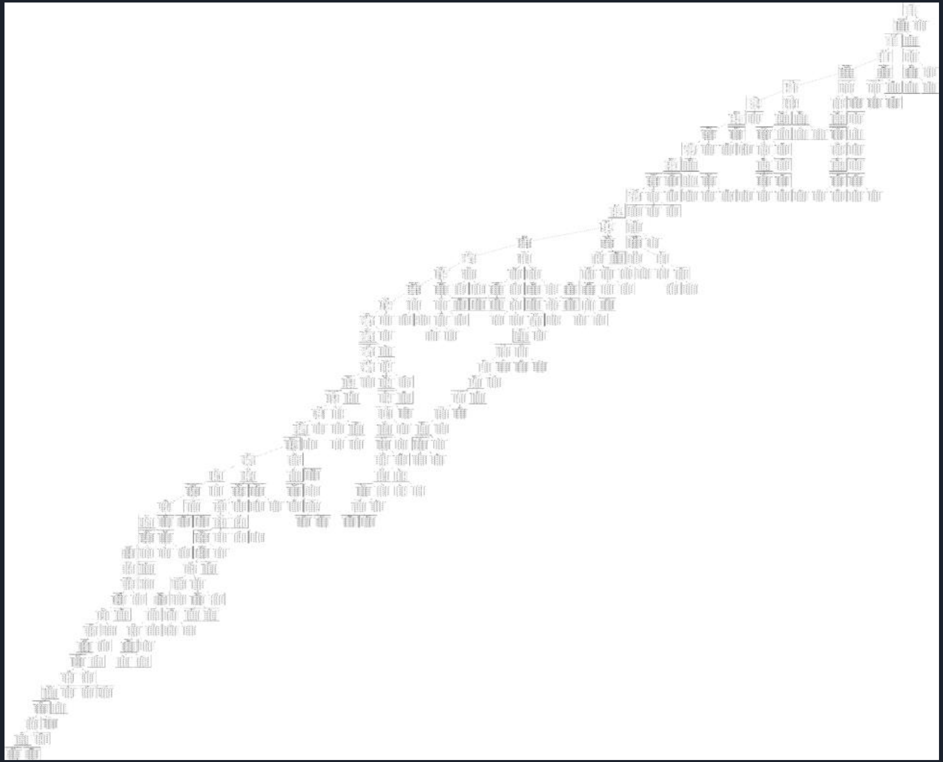
The goal of using a Decision Tree is to create a training model that can **use to predict the class or value of the target variable** by learning simple decision rules inferred from prior data(training data). In Decision Trees, for predicting a class label for a record we start from the root of the tree.



Support Vector Machine

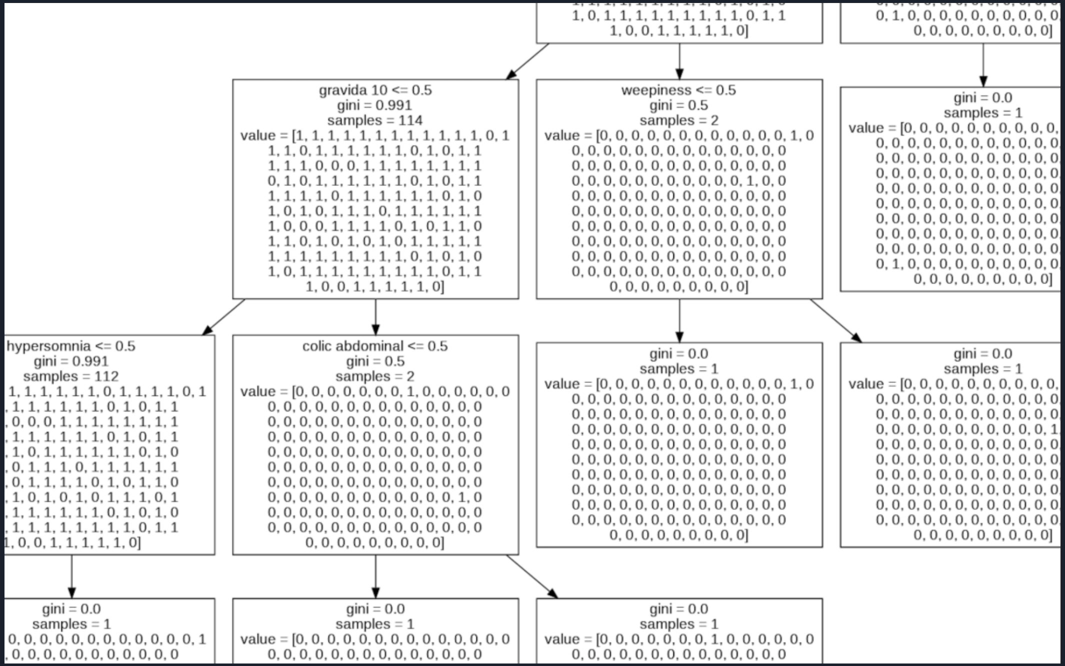
Support-vector machines SVMs, also Support-vector networks are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis.

Results using Decision Tree



Single Block Representation

Accuracy which we got using Decision Tree is 97.3154



We also used SVM to compare the accuracies and the accuracy which we got by SVM is 98.1132

Why did we use two different algorithms ?

- To find and compare the accuracy of our model.
- The goal of using a Decision Tree is to create a training model that can **use to predict the class or value of the target variable** by learning simple decision rules inferred from prior data (training data). In Decision Trees, for predicting a class label for a record we start from the root of the tree.
- Major complexity is with the algorithms and mathematical concepts behind them.

Wrapping Things Up

Potential Uses in day-to-day life

Many types of illness can be prevented by just stopping in its beginning phase. This project envisages to help patients with mild symptoms to check for any possibility of disease.

This early detection can help the patient visit the doctor within time and get treated fast and economically.

Thus this project have potential to accomplish the following:

1. (Primarily) save many lives from diseases,

2. Save time and money by early detection.

Thank you!

We thank you for your time for having a look at our project.