DiabeteXpert

Ujjwal Dubey, Tejasvi Karthik Kumar, Kavya Sree Nanduru, Sreshta Reddy N

Abstract

The goal of our data mining problem is to use a dataset which contains features that help us to predict whether or not a person is likely to develop diabetes or is already diagnosed with it. The dataset contains information about various features that are known to be the risk factors for diabetes, such as age, body mass index (BMI), blood pressure, and glucose levels. The goal is to use this data to build a model that can accurately predict whether a person is likely to develop diabetes based on their individual risk factors. If an individual is at risk of developing or already has diabetes, it is possible to predict their likelihood of also developing liver, kidney, and heart diseases as well.

Keywords
Data Mining, Machine Learning, Diabetes, Chronic Kidney Disease, Non-Alcoholic Fatty Liver, Random Forest, Neural Network, Decision Tree Classifier, K Nearest Neighbour, Logistic Regression, Data Analysis

[1] *Luddy School of Informatics, Computing, and Engineering, Indiana University, Bloomington, IN, USA*

Contents

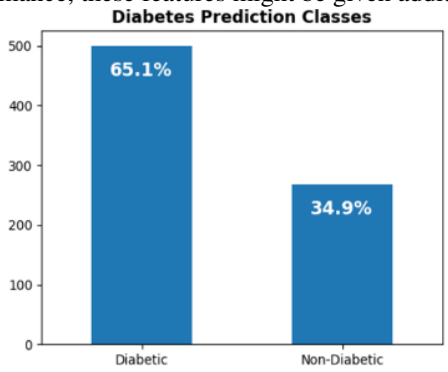
1. Problem and Data Description

1.1 Problem Statement:
Diabetes is a chronic health condition. The number of people with diabetes worldwide has been increasing over the years. According to the International Diabetes Federation, as of 2021, approximately 537 million adults (age 20-79 years) were living with diabetes worldwide. It affects millions of people worldwide and can lead to serious health complications. When diabetes is predicted early, steps can be taken to stop or delay the onset of the condition, which can improve health outcomes and lessen the financial toll that diabetes has on both individuals and society. Second, diabetes prediction can assist in making clear clinical judgments. For instance, if a patient is determined to have a high risk of developing diabetes, their doctor may advise either medication or lifestyle modifications like diet and exercise to stop or delay the onset of the condition. Finally, planning and resource allocation for public health can benefit from diabetes prediction. Public health professionals can tailor interventions to those who are most in need, which can be more efficient and successful than broad-based methods, by identifying groups at high risk of getting diabetes. After receiving a positive diabetes diagnosis, a person is more likely to develop heart, liver, or renal issues. Therefore, we created a model that allows a diabetic patient to additionally examine the health of their heart, liver, and kidneys.

1.2 Data Description:
We have used different datasets which predict the overall health of the patient.

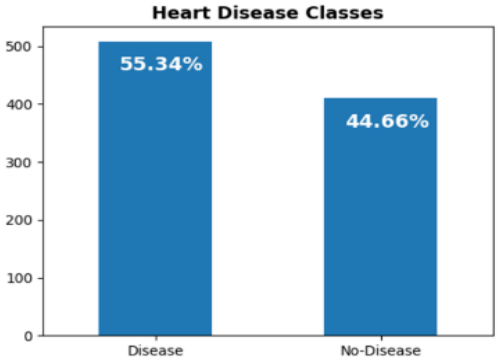
Diabetes dataset:
The Pima Indian Diabetes dataset, which is accessible on a number of data repositories including Kaggle, the UCI Machine Learning Repository, and OpenML, is one of the most frequently used datasets for diabetes prediction. The diabetes prediction dataset consists of 768 patient medical records, each of which includes nine input variables and a binary target variable that indicates whether or not the patient has diabetes. The input elements include the total number of pregnancies, blood glucose levels, blood pressure, skin thickness, insulin levels, body mass index (BMI), function of the diabetes family history, and age. By predicting the result variable (the presence or absence of diabetes) based on the input attributes, machine learning methods may be used to examine this dataset. Glucose concentration, BMI, age, and diabetes pedigree function are the elements of diabetes prediction models that are most frequently utilized because of their strong associations with the onset of diabetes. The outcome variable (the goal variable) is binary and shows whether the patient has diabetes or not. The sample dataset is shown below: The characteristics in this dataset that are frequently utilized to address the diabetes prediction issue include Glucose, body mass index (BMI), age, and DiabetesPedigreeFunction. Many diabetes risk prediction models utilize these factors because they are

frequently linked to the onset of diabetes. Additionally, research has shown that glucose and BMI are the key indicators for predicting diabetes. In order to enhance the model's performance, these features might be given additional weight.



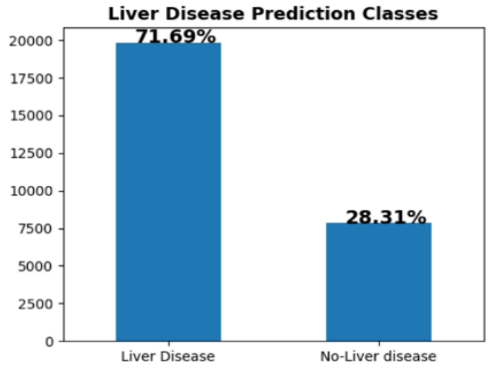
Heart dataset:

Heart Failure Prediction Dataset is taken from Kaggle. The goal is to determine whether a patient has heart disease based on diagnostic parameters. The dataset has records of 918 patients, and it has 12 attributes which help to predict heart disease. The input elements are Age, Sex, ChestPainType, RestingBP, Cholesterol, FastingBS, RestingECG, MaxHR, ExerciseAngina, Oldpeak, STSlope. The target variable "HeartDisease" is binary which shows whether the patient is likely to develop heart disease or not.



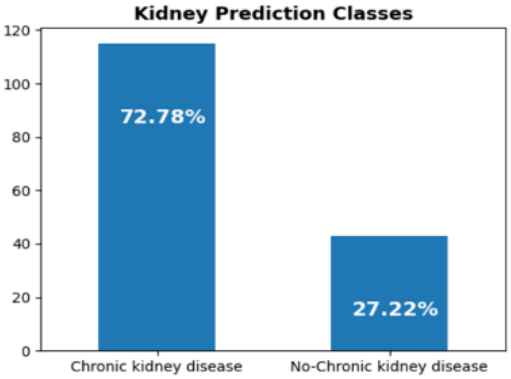
Liver dataset:

Liver Dataset is taken from Kaggle. Liver Patient Records Patient records collected from patients in India. This data collection includes 167 non-liver patient records and 416 liver patient records that were gathered from Indian medical records. To categorize groups as liver patients (liver disease) or not (no disease), the "Dataset" column is used as a class label. There are 142 female patient records and 441 male patient records in this data collection. The input elements are Age, Gender, TotalBilirubin, DirectBilirubin,AlkalinePhosphotase,AlamineAminotransferase, Aspartate Aminotransferase, TotalProtiens, Albumin, AlbuminandGlobulinRatio.



Kidney dataset:

Kidney Dataset is taken from Kaggle. Kidney Patient Records Patient records collected from patients in India. This dataset contains 25 input features which are considered the medical diagnostic parameters which affect the functioning of kidneys. The dataset consists of 400 rows. The output column "classification" tells whether the patient has kidney problems or not. The dataset consists of many missing and categorical values which are to be handled properly to obtain accurate results.



2. Data Preprocessing & Exploratory Data Analysis

2.1 Handling Missing Values

Before we find the missing values in the dataset, The Pima Indian Diabetes dataset must first be loaded into the software environment .csv format is used to load the dataset. To make the data appropriate for analysis, it must first be cleaned and transformed, a process known as data preprocessing. Scaling the data and handling outliers and missing values are included in this. Checking for missing values is crucial to determine whether or not the dataset contains any missing values. This is done using diabetesdataset.isnull().sum().sum() command. After analysing, we come to a conclusion that there are no missing values in the dataset.

```
import numpy as np
#Finding out the number of missing data
diabetes_dataset.replace('0',np.nan, inplace=True)
missing_values = diabetes_dataset.isnull().sum().sum()
print("Number of unknown/missing values in dataset:", missing_values)
```

```
Number of unknown/missing values in dataset: 0
```

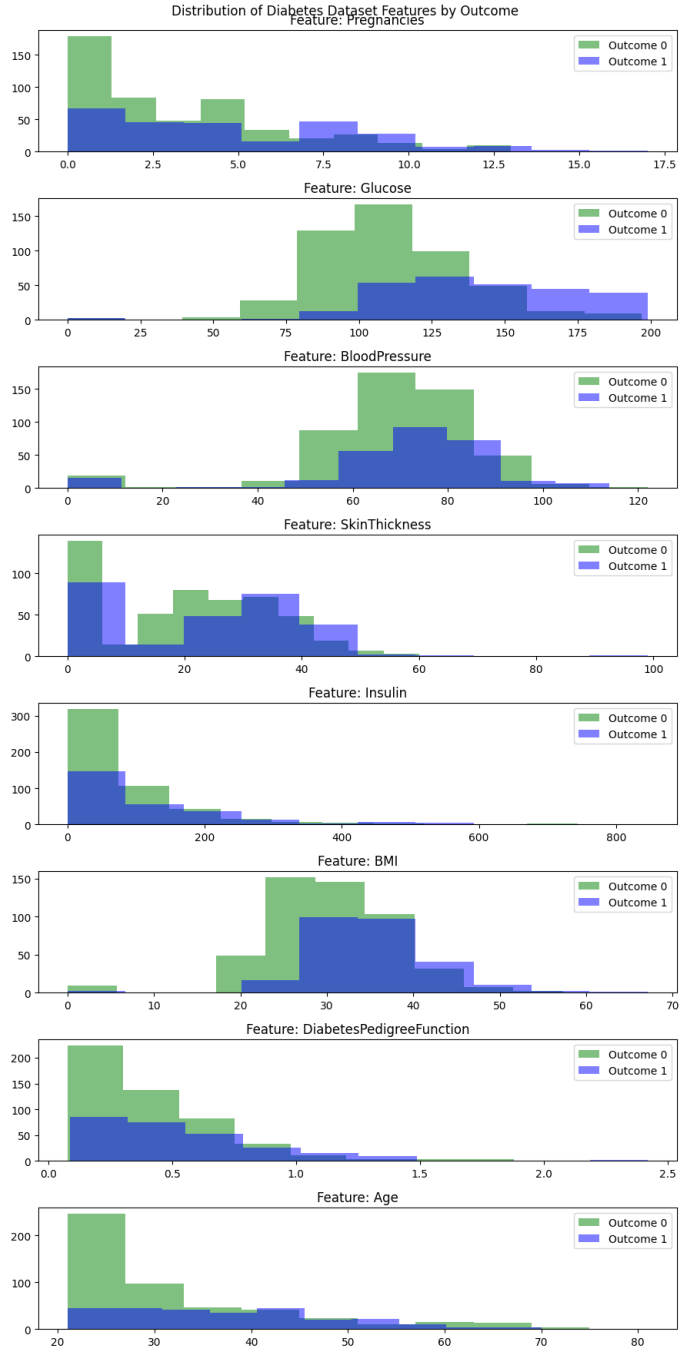
Now we are checking if there are any categorical variables in the dataset. Categorical variables are variables that can take on a limited number of values, such as 'Male' or 'Female'. We can check for categorical variables using the dtypes function in pandas. This will print the data type for each column in the dataset. If any of the columns are of type 'object', we can assume that they are categorical variables and handle them appropriately. In our case, we don't have any categorical variables in the dataset.

```
diabetes_dataset.dtypes
```

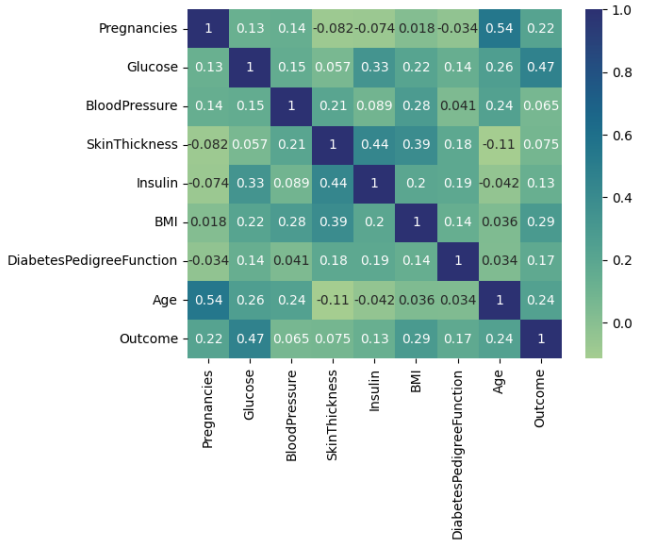
```
Pregnancies                 int64
Glucose                     int64
BloodPressure               int64
SkinThickness               int64
Insulin                     int64
BMI                       float64
DiabetesPedigreeFunction  float64
Age                         int64
Outcome                     int64
dtype: object
```

2.2 Exploratory Data Analysis

Exploratory data analysis is an iterative process, and it can be repeated multiple times until we get satisfied results. The dataset's summary statistics need to be calculated. It includes each variable's mean, median, mode, minimum, maximum, and standard deviation. We have used summary() function in Python for this purpose. Exploratory data analysis requires the use of data visualization. It helps to understand the relationship between different attributes and the distribution of data. We have created histograms, box plots, scatter plots, and heat maps to visualize the data. The diabetes dataset exhibits clear patterns in its variables. Pregnancies range from 0 to 18, with a peak at 1. Glucose ranges from 0 to 200, with a peak at 80 to 130. Blood pressure ranges from 0 to 120, with a peak from 50 to 90. Skin thickness ranges from 0 to 100, with a peak at 0 and a smaller peak at 20. Insulin ranges from 0 to 800, with a peak at 50. BMI ranges from 0 to 70, with the maximum value at 25 to 40. The DiabetesPedigree Function ranges from 0 to 2.5, with a peak at 0.25. Age ranges from 20 to 80, with a peak at 25.

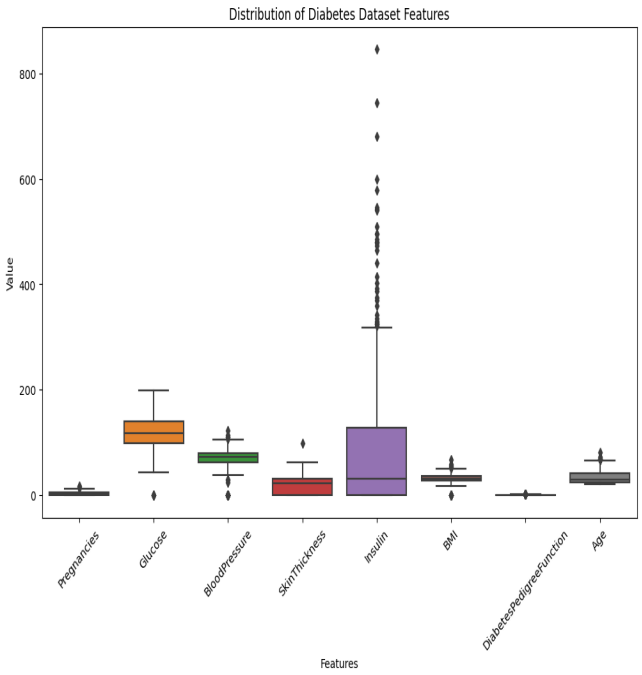


Correlation Analysis is used in understanding the relationship between several variables and it is easier to analyze considering this. We have calculated the correlation coefficient between each pair of variables using the corr() function in Python and used heatmap and pair plot to represent the correlation between the attributes. By considering the heatmap as mentioned below, there is a positive correlation between BMI and SkinThickness and Insulin. Pregnancies and Age are highly correlated. Additionally, there is a weak positive correlation between Glucose and Outcome, but no significant correlation between DiabetesPedigreeFunction and any of the other features.

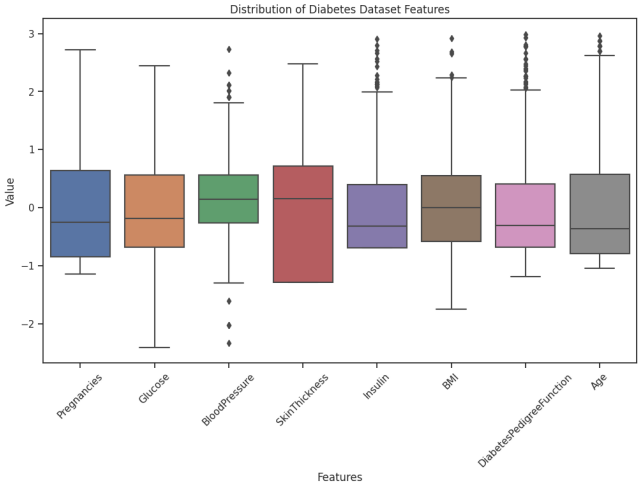


The heatmap can help us identify which features are more strongly correlated with each other and with the outcome variable, which can help us in selecting relevant features for building the predictive model. A data point is considered an outlier if it differs significantly from the rest. We have used box plots to identify outliers in the dataset. We have calculated the Z-score for each data point in the diabetes dataset and removed data points with a Z-score greater than 3. The Z-score is a measure of how many standard deviations a data point is away from the mean.

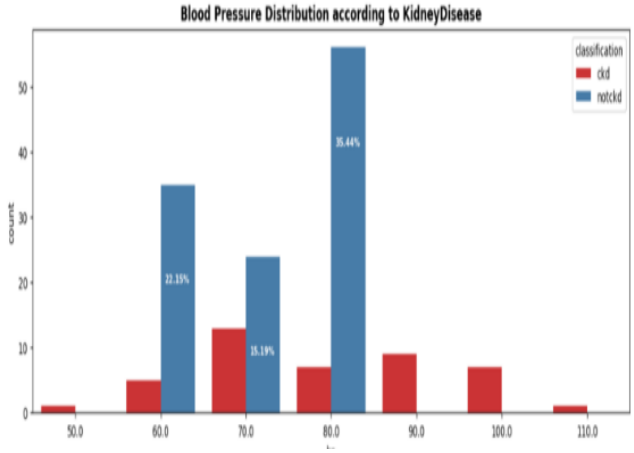
In order to show any anomalies, a box diagram for each number column in the dataset is created. Based on our research, we can then determine how to deal with the outliers.



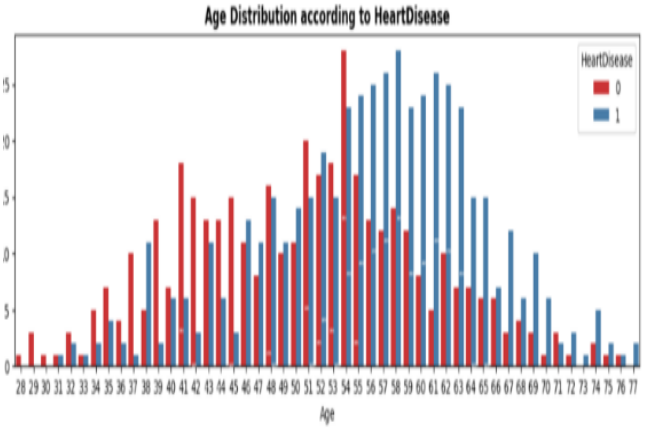
Re-Analyzing Outliers, after outlier reduction:



As we can see below is the data analysis for the kidney dataset. The majority of the patients not having kidney disease have a blood pressure of 80.



The figure below is data analysis for the heart dataset. The majority of the patients having heart disease lie in the age group of 55-65.



3. Algorithm and Methodology

Health prediction web applications use a variety of algorithms and methodologies to predict the likelihood of an individual developing health issues based on their risk factors and other data. Here are some of the common algorithms and methodologies used in our predictions:

Logistic Regression: Logistic regression is a statistical method that is used to analyze the relationship between a set of independent variables and a binary outcome variable. It is used in many fields like medical research, marketing, finance etc. In the context of health prediction or a disease prediction, the independent variables may include factors such as age, body mass index (BMI), family history, blood pressure, glucose levels, Cholesterol etc. The outcome variable would be whether the individual is at risk of developing health issues or not.

Decision Trees: Decision trees are a type of machine learning algorithm that is used for classification and regression analysis. They are particularly useful for identifying complex patterns in large datasets. In the context of health prediction, decision trees can be used to identify the most important risk factors for causing a health issue and to predict the likelihood of an individual developing the condition based on those factors.

Random Forest: Random Forest is an ensemble machine learning algorithm that uses multiple decision trees to make predictions. It is particularly effective in dealing with noisy data and handling large datasets. In the context of health condition prediction, Random Forest can be used to identify the most important risk factors related to health and predicts the likelihood of an individual developing the condition based on those factors.

Support Vector Machines: Support Vector Machines (SVMs) are a type of machine learning algorithm that is used for classification and regression analysis. They work by creating a hyperplane that separates the data into different classes.

Neural Networks: Neural networks are a type of machine learning algorithm that is modeled after the structure of the human brain. They are particularly effective in handling complex patterns in large datasets.

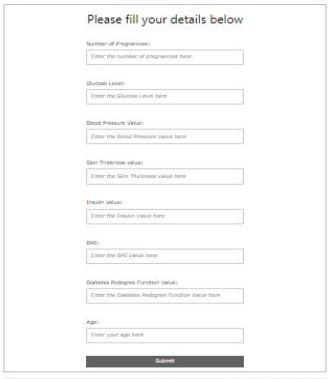
K-Nearest Neighbor: K-Nearest Neighbor is a type of machine learning algorithm that is used for classification and regression analysis. It works by finding the K-nearest points in the dataset to a given point and then classifying or predicting the outcome based on those points. In the context of health prediction, K-Nearest Neighbor can be used to identify the most important risk factors and predict the likelihood of an individual developing the condition based on those factors.

In summary, diabetes prediction web applications use a range of algorithms and methodologies to predict the likelihood of an individual developing health issues. These algorithms and methodologies are based on statistical and machine learning techniques, and they analyze various risk factors such as age, BMI, family history, blood pressure, glucose levels, cholestrol etc to make predictions. The choice of algorithm or methodology depends on the specific requirements of the web application and the dataset being analyzed.

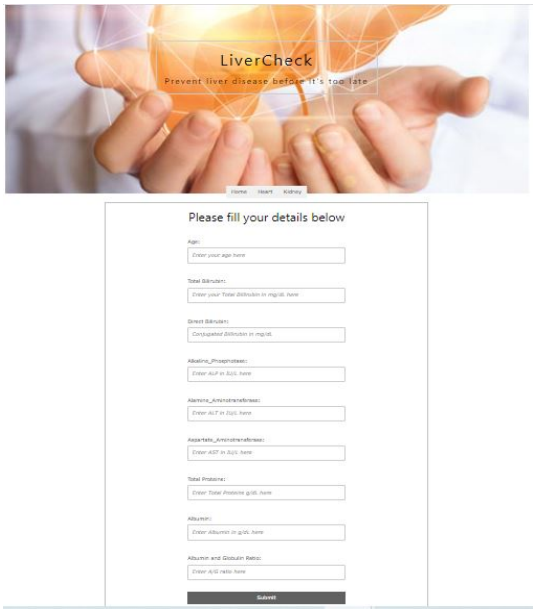
4. Experiments and Results

The following classifier algorithms performed exceptionally well for the respective datasets:

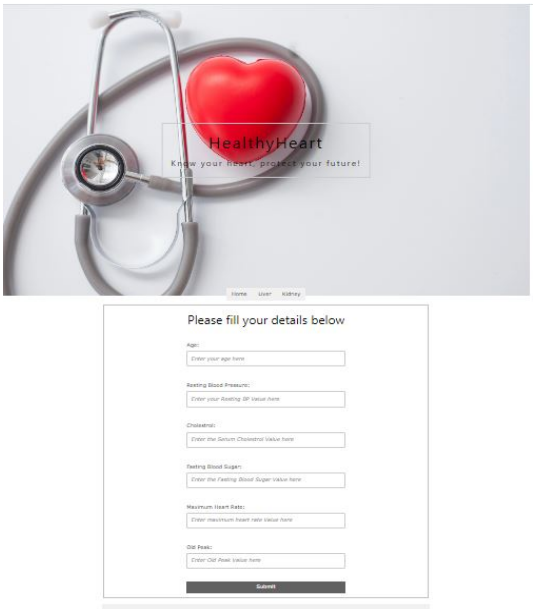
Diabetes dataset: KNN, SVC,Logistic Regression and Random Forest were used for this dataset and logistic regression gave the highest accuracy out of them at 77%. Logistic Regression can be regularized to prevent overfitting, which is important if the dataset is small or has a lot of noise. We used L1 (Lasso) and L2(Ridge) penalty functions to improve the model's generalization performance.



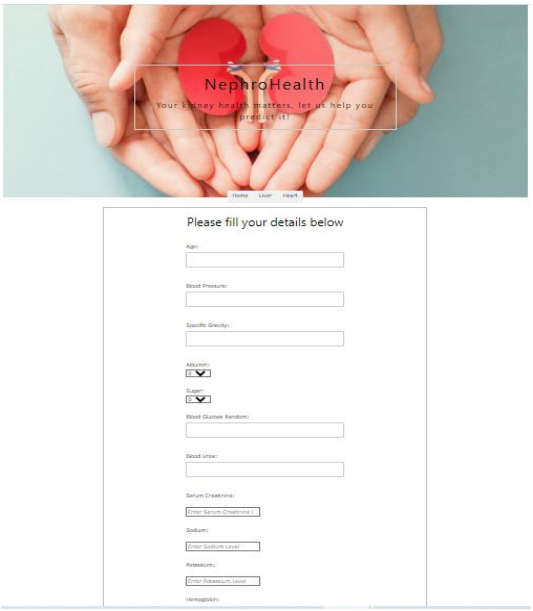
Liver Dataset: Random Forest and Decision Tree were considered better for the Liver Dataset because they can handle non-linear relationships between variables effectively, which is common in real-world datasets. They can also capture interactions between features, unlike Logistic Regression, which assumes that the effect of each feature is independent of other features. This makes Random Forest and Decision Tree the better choice and it gave us an accuracy of 90%.



Heart dataset: KNN, SVC, Neural Network and Random Forest were performed on the dataset. Random forest gave us an accuracy of 86% making it the best choice. This is because Random Forest has built-in feature selection, while Neural Networks require manual selection.



Kidney dataset: Both Neural Network and Random Forest were performed on this. Although both the algorithms gave us high accuracy, the accuracy of neural network was not stable, whereas random forest gave a constant accuracy of 93%. This is because Random Forest is less sensitive to outliers: Random Forest is a collection of decision trees, and the decision at each split is based on a subset of the features. This reduces the impact of outliers on the overall prediction. Neural Networks, on the other hand, are more sensitive to outliers, which can impact the accuracy of the model.



5. Deployment and Maintenance

We have deployed our web application on Heroku by following these steps:

Deployment Steps:

1. Signed up for a Heroku account.
2. Installed the Heroku CLI tool on our computer.
3. Opened the terminal/command prompt and navigated to our project directory.
4. Created a new Git repository using the command: git init.
5. Created a new Heroku app using the command: heroku create.
6. Created a new requirements file using the command: pip freeze ¿ requirements.txt.
7. Created a new Procfile using the command: echo web: gunicorn app:app - Procfile.
8. Committed our changes using the command: git add . and git commit -m ”Initial commit”.
9. Pushed our code to Heroku using the command: git push heroku master.

Maintenance Steps:

1. Regularly monitor our application for errors or bugs.
2. Keep our dependencies up-to-date by regularly updating our requirements file.

3. Back up our data regularly to avoid data loss in case of any unexpected issues.

4. Optimize our code for performance to ensure that our application runs smoothly.

5. Stay up-to-date with security patches and updates to avoid any security vulnerabilities.

6. Regularly test our application to ensure that it is functioning correctly and meeting user needs.

7. Respond promptly to user feedback and complaints to improve the user experience.

6. Summary and Conclusions

Our web application is a useful tool for medical professionals and patients alike. With four separate web pages(DiabeteXpert, LiverCheck, HealthyHeart, NephroHealth), each dedicated to predicting a specific disease, it covers a range of health concerns. The front-end is designed using JavaScript, HTML, and CSS, providing an intuitive and user-friendly interface. The backend code, written in Python, uses machine learning models to make predictions for each disease.

The Random Forest model was chosen as the best model for all datasets, as it consistently produced high accuracy scores. This is likely due to the Random Forest model's ability to handle complex data and find non-linear relationships between the features and the target variable.

This web application will be useful for physicians' assistants, medical interns, and trainees, as it will help them quickly and accurately predict the risk of various diseases in their patients. Additionally, diabetic patients can also use this application to assess their risk of developing other diseases such as kidney, liver, and heart disease.

In conclusion, our web application is a valuable tool that can save time and improve accuracy in diagnosing diseases. The Random Forest model's ability to handle complex data makes it a good choice for predicting a range of diseases. Our application's user-friendly interface will be useful for medical professionals and patients alike, making it a valuable addition to the healthcare industry.

Acknowledgments

We would like to express our sincere gratitude to Professor Hasan Kurban, who guided us throughout the project. His expertise and knowledge in the field of Data Mining were invaluable in helping us understand the complex concepts involved in this project.
His guidance and support have been instrumental in the success of this project, and we are grateful for his mentorship. He was always available to answer our questions and provide feedback, which helped us improve the quality of our work. We would also like to thank him for providing us with the opportunity to work on this project and for giving us the resources and tools to complete it successfully.
Thank you, Professor Hasan Kurban, for your support and guidance.

References

VanderPlas, J. (2016). Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly Media

Raschka, S., Mirjalili, V. (2017). Python Machine Learning - Second Edition. Packt Publishing.

Witten, I. H., Frank, E., Hall, M. A., Pal, C. J. (2016). Data Mining: Practical Machine Learning Tools and Techniques (Morgan Kaufmann Series in Data Management Systems) 4th Edition. Morgan Kaufmann Publishers.

Tan, P.-N., Steinbach, M., Kumar, V. (2005). Introduction to Data Mining (2nd Edition) (What's New in Computer Science) 2nd Edition. Pearson. National Institute of Diabetes and Digestive and Kidney Diseases. (n.d.). Pima Indians Diabetes Database. Retrieved from https://www.kaggle.com/uciml/pima-indians-diabetes-database

Farjana, A., Das, M. C., Hossen, M. H., Liza, F. T., Hasan, M., Pandit, P. P., Tabassum, F. (2021). Predicting Chronic Kidney Disease Using Machine Learning Algorithms. Journal of Information and Organizational Sciences, 45(2), 181-200.

Fernandez de Soriano, F. (2021). Heart Failure Prediction. Retrieved from https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction Mansoor Daku. CKDisease Dataset. Kaggle, 2021. https://www.kaggle.com/datasets/mansoordaku/ckdisease

Abhinav Shrivastava. (2021). Liver Disease Patient Dataset. Retrieved from Kaggle: https://www.kaggle.com/datasets/abhi8923shriv/liver-disease-patient-dataset