**MAKERERE  UNIVERSITY**

**COLLEGE OF COMPUTING AND INFORMATION SCIENCES**

**(YEAR II) RECESS TERM**

**SOFTWARE DESIGN DOCUMENT**

**FOR: DIABETES PREDICTION MODEL**

**PROJECT MEMBERS**

|  |  |  |
| --- | --- | --- |
| **NAME** | **REGISTRATION NUMBER** | **STUDENT NUMBER** |
| Kabahinda Rovence | 17/U/4261/PS | 217003233 |
| Meleby Nyika Jino | 17/x/18244/PS | 217019537 |
| Arikosi Michael Okurut | 17/U/2900/PS | 217004064 |
| Mwanje Mike | 17/U/6856/PS | 217004074 |

**PROJECT COORDINATOR**: MR. KAMULEGEYA GRACE

**INTRODUCTION**

The System Design Document consists of the whole data pipeline from data sources all the way to visualization, a description of all the key components in this data pipeline, what it is, what each component does and how it does it. It also includes the description of different python libraries that we intend to use during the implementation of our project and the reasons as to why we chose to use these particular libraries. It also includes the visualization components for our pipeline, the visualizations we intend to use and why we think they are the best among other visualization tools.

**DIAGRAM SHOWING DATA PIPELINE UP TO VISUALIZATION**

**\***Load kidney\_disease.csv

into the jupyter notebook using

Pandas library.

**\***Address missing values using

Backward fill method

**\***Label encode columns with string

values

**\***Feature scaling

**\***Using libraries matplotlib plot

A bar graph, Line graph,

Scatter plot, Histogram,

regression line.

*Source: diagram generated by group members using Microsoft word*

**DATA LOADING**

We intend to load the kidney\_disease.csv dataset into the jupyter notebook using *pandas library*, a data manipulation library.

This we shall do by importing pandas library first by using the statement “*import pandas as pd*” and so after this statement, the jupyter notebook will understand that from then every time we type pd, we are referring to pandas. After importing pandas we shall read the dataset into a pandas data structure that is; a dataframe (*df*) using *read\_csv()* function so that every time we run the code we do not have to run the *read\_csv()* function but instead simply call data frame (df).

And for us not to flood the whole screen with data, we decided that we use functions *df\_read.tail()* and *df\_read.head().* These help to automatically generate the last and first 5 elements of the data frame respectively.

***Why use pandas and then pandas dataframe?***

Pandas library provides data structures (*pandas dataframes* and *pandas series*) and data analysis tools for different functions.

A pandas data frame is a 2 or (more dimensional) data structure basically a table with rows and columns. The columns have names and the rows have indices

Just like excel files, pandas data frames provide various functionalities that will be used in the analysis, changing and extracting of valuable information from the kidney\_disease.csv dataset.

Data frames also provide various column helper functions which shall extremely be useful for extracting valuable information from the column for example; *unique* - provides unique elements from a column by removing duplicates, *mean* - provide the mean value of all the items in the column and among others.

**DATA CLEANING/ WRANGLING**

Our data cleaning process shall involve fine tuning the kidney\_disease.csv dataset to transform it to a state that will help us attain the best possible results in the grand scope of the diabetes detection problem. Since our data set has missing values, we shall address this problem by using **interpolation**for the numeric columns and backward fill for the nominal (columns with string values) columns. The interpolation method uses the interpolate() function which fills in a missing value basing on a specified domain of values around it.

**Backward fill** on the other hand uses bfill() function on available values in a column to fill in precedent missing values in the same column. Both functions utilize the pandas library and NumPy libararies and the major reasons for the use of these two libraries will be discussed in the tail of this section.

The next step of our data cleaning process shall be **label encoding**. This means mapping different string values of a feature to distinct numeric values and it shall be achieved using the *.map()* function in the *numpy library* which lets a user specify strings and assign them to a distinct corresponding numeric values. This operation will be carried out on the rbc(reb blood cells), pc(pus cell), pcc(pus cell count), ba(bacteria), htn(hypertension) and dm(diabetes mellitus) features of the kidney\_disease.csv dataset to encode their nominal values to binary 0’s and 1’s since their values vary between normal or abnormal, present or non-present and yes or No.

The final phase shall involve handling outliers in each column by **feature scaling**the numeric columns of wc , bgr and bu which stand for white blood cell count, blood glucose and blood urea respectively whose values span over a wide range. This step will help our machine learning algorithm K-Nearest Neighbor (KNN) to perform better in the diabetes disease prevalence problem and this shall be made possible by the help of the *scikit learn library* using the *Robust Scaler function* which works by removing the median and using the interquartile range (Q1-Q3) of the values hence it focuses on the parts where the bulk of the data is.

***Reasons for using the libraries mentioned above:***

* **Pandas–**it is an open source and high performance library that is easy to learn, easy to use and also easy to maintain. It also will us provide a multitude of functions for data manipulation, analysis and modelling hence allow us to focus more on developing an accurate model for the diabetes disease prevalence model rather than spending time programming functions that have already been coded and tested. Pandas also provides a fast and efficient data frame object for data manipulation, less relevant columns such as pc(pus cell) from our kidney\_disease dataset can be deleted/ dropped . Pandas also has widespread community support so in case of any difficulties concerning usage of some its functions then we can get help online.
* **Numpy –**it is the fundamental package for scientific computing with python. It will provide us with a wide variety of sophisticated functions to carry out mathematical computations and analysis on our kidney\_disease.csv dataset. It is also widely used hence there is support available online in case of challenges when using it during our implementation.
* **Scikit learn** – it shall provide us with a range of learning algorithms through a consistent interface in python. It will help us to model data and do feature selection to identify meaningful attributes from which to create our best possible model. It will help us in parameter tuning to get the most accuracy out of our model and also in splitting our dataset to train our model.

**DATA VISUALIZATION**

This section is a brief description of how we intend to help our users (medical personnel, supervisor, data scientists) and the system administrator to understand the significance of the kidney\_disease.csv dataset by placing it in a visual context. In other words, it describes the movement from raw data that is; kidney\_disease.csv dataset to meaningful insights and learning

In the first place, we shall have to import ***matplotlib*** into the workspace (jupyter). This matplotlib library has a module called ***pyplot*** which is used to create figures, create a plotting area within the figure, plot some lines or dots within the figure, decorate plots among other functionalities.

We shall import the pyplot module into our workspace by adding a statement; “ *from matplotlib import pyplot as plt* ” or “*import matplotlib.pyplot as plt*” incase matplotlib is already imported

***Why use matplotlib.pyplot?***

* It supports a very wide variety of plots and graphs for example scatter plots, regression lines, histograms and bar graphs.

***Why plot the graphs mentioned above?***

* Scatter plot
* Barplot
* Histogram
* Regression line

*Scatter Plot*: this describes the value of any ­­­­two sets of data on two dimensions. Each dot represents an observation. It is relevant in our diabetes disease prevalence problem because it is good for the study of the relationship between two variables. In addition, it shall provide more information about our data by using colors or shapes to show groups or any other third variable

*Histogram*: This is made up of columns plotted on a graph. Usually there is no space between these columns. This shall be used to show the skewdness of the data

*Bar plot:* This is made up of columns plotted on a graph also just like a histogram. Usually there’s space between the columns. This shall be used to show the distribution of the data

*Regression line*: This shall be used for prediction and forecasting where its use has a significant overlap with machine learning. It shall be used to understand which among the independent variables are related to the dependent variable and to explore the forms of these relationships.

REFERENCE:

1. <https://archive.ics.uci.edu/ml/datasets/chronic_kidney_disease>