**MAKERERE  UNIVERSITY**

**COLLEGE OF COMPUTING AND INFORMATION SCIENCES**

**SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY**

**(YEAR II) RECESS TERM**

**SYTEM DESIGN SPECIFICATIONS DOCUMENT(SDD)**

**PROJECT MEMBERS**

|  |  |  |
| --- | --- | --- |
| **NAME** | **REG NO** | **STUDENT NUMBER** |
| Nagaba Angel | 17/U/726 | 217000189 |
| Okello Marvin Kevin Ochira | 17/U/9569/PS | 217017015 |
| Karungi Lydia | 17/U/4676/PS | 217002012 |
| Wepukhulu Bruno | 17/U/10891/PS | 217012574 |

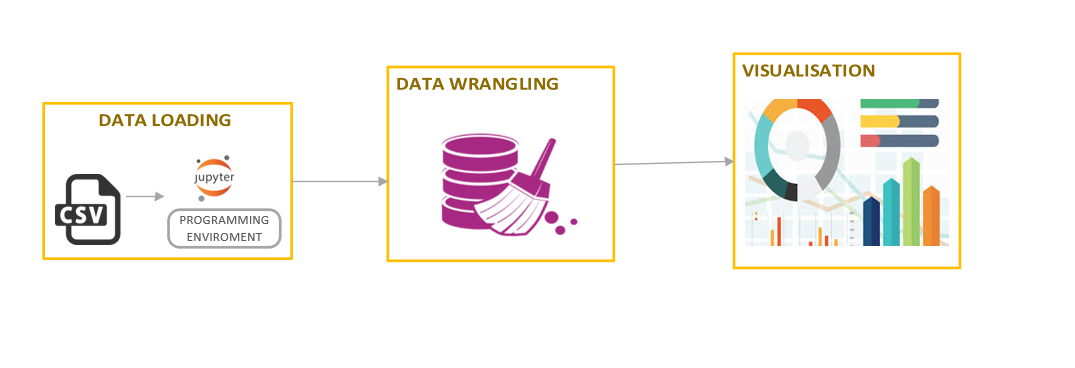
**PROJECT COORDINATOR**: MR. KAMULEGEYA GRACE

Introduction

The system design specifications document describes the details of the pipeline all the way from datasources to the visualisation. This document gives a detailed explanation of each of the components of the pipeline and how each of the activities involved shall be performed.

It also explains the libraries that support each of the activities in the pipeline and why those libraries are used. It furthermore explains the viualisation tools that we shall use for our accidents dataset and why those visualisation tools are used.

Representation of the pipeline from datasources to visualisation



*Source: this diagram was creatively generated by the group using microsoft visio*

Components of the pipeline

**Data loading**

The first step is data collection, in this case we collected our dataset from [www.kaggle.com](http://www.kaggle.com), which was in csv format.

After, the data is loaded into a dataframe from the csv file in the python notebook.

The is done using the method **“read\_csv”** which is a data structure supported by the pandas library.Since we have three datasets we are going to combine them into one dataset. This component of the pipeline is supported by the python pandas library. The pandas library provides tools for writing and reading the data.

**Data wrangling**

The process of data wrangling(formerly known as data cleaning) involves activities like label encoding, checking and eliminating missing values as well as feature scaling.

Missing values

The first step is to check whether missing values exist in the dataset and this is done by running the line of code “**dataframeName.isnull().values.any()”** if it evaluates to true, it implies that there are missing values. The pandas library provides various methods to deal with missing values namely: foward fill, backward fill, filling with zero, mean or median and interpolation. We shall use the interpolation method to remove missing values in our data is its close to accurate.

Label encoding

This refers to converting the labels of textual columns into numeric form so as to make it machine-readable. Our dataset doesn’t require this part of the pipeline since its already encoded.

Feature scaling

It basically helps to normalise the data within a particular range(removing outliers). Sometimes, it also helps in speeding up the calculations in an algorithm.

It is done by methods provided by the done by **scikit-learn.** This package contains a module sklearn.preprocessing that helps us to carry out machine learning. Feature scaling is done becausemany machine learning algorithms perform better or converge faster when features are on a relatively similar scale and/or close to normally distributed.

From the above package we can either use the standard scaler or robust scaler for feature scaling.

Robustscaler is used when we want to reduce the effect of the outliers. Minmax scaler doesnt reduce the effect of the outliers.

[RobustScaler](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.RobustScaler.html?source=post_page---------------------------) transforms the feature vector by subtracting the median and then dividing by the interquartile range (75% value — 25% value).

[StandardScaler](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html?source=post_page---------------------------) standardizes a feature by subtracting the mean and then scaling to unit variance. Unit variance means dividing all the values by the standard deviation. StandardScaler does not meet the strict definition of *scale* I introduced earlier.

**Why scikit-learn for feature scaling**

Scikit-learn is a free machine learning library for Python. It features various algorithms like support vector machine, random forests, and k-neighbours, and it also supports Python numerical and scientific libraries like Numpy and Scipy.

**N.B:** Both data loading and data wrangling are supported by the pandas and numpy libraries

**Why pandas**

-Pandas provides an organised form of data representation. This helps to analyze and understand data better. It can present data in a way that is suitable for data analysis via its Series and DataFrame data structures.

-The package contains multiple methods for convenient data filtering.

-Pandas has a variety of utilities to perform Input/Output operations in a seamless manner. It can read data from a variety of formats such as CSV, TSV, MS Excel, etc.

**Why numpy(numerical python) library**

Numpy (numerical python) is a library that helps to deal with scientific computations.

It provides a high-performance multidimensional array and basic tools to compute with and manipulate these arrays.

**Data Visualisation**

After the data cleaning/wrangling exercise is done, the data is visualised in order to come up with plots, pictorial representations, that will help us to come up with various insights and conclusions. This exercise is done by importing the matplotliblibrary and seaborn library and then using the various methods to plot. These libraries can be intergrated by importing them before starting the actual visualisation.

The data can be visualised to come up with visualisations like scatter plots, Histogram, maps, multiple linear regression, logistic regression e.t.c

**Why matplotlib library**

It has a module named pyplot which makes things easy for plotting by providing feature to control line styles, font properties, formatting axes etc. It supports a very wide variety of graphs and plots namely - histogram, bar charts, scatter plots e.t.c which charts happen to be part of our visualisation plan.

**Why seaborn library**

Seaborn is a [Python](https://www.packtpub.com/tech/Python" \o "Python eBooks and Videos) library created for enhanced data visualization.It is built on top of [matplotlib](https://matplotlib.org/) and closely integrated with [pandas](https://pandas.pydata.org/) data structures.

-seaborn provides an API that is dataset-oriented which helps in examining relationships between multiple variables

- Seaborn makes it easier to carry out the data visualisation exercise because it provides automatic estimation and plotting of linear regression models for different kinds of dependent variables

-seaborn library also provides tools for choosing color palettes that faithfully reveal patterns in your data

**Reasons for the above mentioned visualisation tools**

**Scatter plot**

-It shows the relationship between two variables

-It shows us the non-regression pattern

-It shows the range of data i.e. Maximum or minimum value can be determined

-Observation and reading is straight forward

**Histogram**

-This Is an accurate representation of the distribution of numerical data. It relates one variable.

-It helps to visualize the distribution of data

-It identifies different data, the frequency of the data occurring in the dataset and categories which are difficult to interpret in tabular form.

**Maps**

In our dataset we shall draw a map to show the distribution of accidents.

Maps add value to the reader and helps them visualize and understand the data geographically

**Multiple linear regression**

* Since we have more than one independent variable in our dataset multiple linear regression is the best type of linear regression to use. It establishes a relationship between dependent variables and the independent variables using the best fit straight line (y = mx + c).
* It has the ability to determine the relative influence of the predictor variables to the criterion value for example we could find that the vehicle type and the Age of the driver have a strong correlation to the rate of accidents while the sex of the driver has no correlation
* It also has the ability to identify outliers that is to say the variable that comes out to be out of range compared to others is considered as an outlier.

**Logistic regression**

Here we use data to find the probability of event=success(less fatalities when an accident happens) and event=failure(many fatalities when an accident happens). And it is used when the dependent variable is binary.

Since of dependent variable is accident severity and is not binary i.e it includes slight, serious and fatal, we shall combine slight and serious to form non-fatal so as to use this model.

And in order to avoid over fitting and under fitting in this case, we shall use a stepwise method to estimate the logical regression.

References

<https://dev.to/marsja/essential-python-libraries-for-data-science-machine-learning-and-statistics-5175>

<https://towardsdatascience.com/scale-standardize-or-normalize-with-scikit-learn-6ccc7d176a02>

[http://pandas.pydata.org](http://pandas.pydata.org/)

<http://scikit-learn.org/>

<https://seaborn.pydata.org/introduction.html>