Department of Computer Science

Student Number	1812755	Supervisor	Martin Shepperd
Programme	Computer Science	Specialism	Software Engineering
Provisional Title	Predication models of COVID disease outcome		

Problem Definition

This project will focus on delivering a piece of software which will address to issue with current virus COVID-19, the main purpose is to provide information on what will happen next in regards to how many infected cases, number of deaths or what could be an outcome of infected patients. Also, it is important to consider additional information like preexisting medical conditions, disease parameters, demographic information, etc. and how they impact the infected patients. This will be helpful when making models as it would allow for better accuracy. Moreover, this is a serious issue that needs addressing as software like this could give important information when making decisions for businesses, by government or hospitals. This problem will focus on using historical data from pass events, data analytics and Machine learning. All these concepts will focus on understanding the virus by analyzing data and try to produce visualization that can be used to have a better understanding of what is the current situation and what the situation could be in particular amount of time in the future.

This system will also provide visualization of the data, for example, using time series plots, choropleth maps to indicate effected areas. These will help users to understand analyzed data in clear way. Moreover, the historical data will be used to predict future events and it will be used to build models that highlight important trends. Furthermore, these models are used on train data which will try to predict on what will happen next or give an estimation on what might be optimal outcome.

Aims and Objectives

The aim of this project is to produce a software which creates prediction based on historical data that has information about pass events which can be used in machine learning to create different types of models and compare them to determinate which one is more accurate or suitable. These models also will visualize the predicted outcome.
It will also give an option to user to see current data which will be visualized clearly using graphs. Moreover, the objectives to meet this aim are following:

- **Collect data -** This is important to find source of information that will be used to train algorithms, These data sets will also be used to give insights on the current situation by displaying charts, graphs, maps etc. This will allow to understand current situation and paint an image in user's head to be able see how the prediction models work and predict the outcomes. These data sets will also be used to check how accurate a particular model is by using test sets. This data will need to be cleaned, meaning it will have to be analyzed for any missing values, normalization, quality check, duplication etc.
- **Create UI –** This is also important as all this information has to be presented in a elegant way to help the user understand it in simple way as some of this can be complicated information and some users may find it complex, Therefore it's important that this is addressed to all users like, younger users as well as elderly users or even medical departments. This can be implemented for example using Shiny R package.

- **Visualize data –** Display graphs, charts, maps to help the user visually illustrate the relationship in the data that might be complex or very big.
- **Creating prediction models using different algorithms –** This part is very important as this will be used to predict the outcome. Therefore, it is a key element to make sure that they will be as accurate as possible. This can be done by analyzing available prediction algorithms to help understand what output they give and how accurate there are using particular data set. These algorithms may include classification, clustering, K-means, Forecast Model, time series model, random forest, Prophet.
- **Compare algorithms –** after analyzing the algorithms they should be compared to check for accuracy and determinate which one should be used in the final system.
- **Visualize models –** The prediction models should be presented in the UI in clear way and easy to understand with explanation of what they represent and mean.
- **Testing –** This is also important and especially if this system would be used by medical specialists or government as decision making tool. It would be essential that the system would try to produce 100% accurate prediction models and would have as little incorrect data as possible. Therefore, this system will focus more on general users, but it is still important to make sure that the system is free of bugs and produces mostly accurate results.
- **Version control -** This will help to keep track of the project and version. This will be done using GitHub as it is a good tool for controlling projects.

Background Sources

One of the main sources I have found which is "Worldometer", it visualizes collected data regarding COVID- 19 in very elegant way and easy to understand for all users. It also breaks down the data by categorizing it which allows easy browsing through the data. This is one of the examples which demonstrates how they have approached this problem to help understand complex and large data. Another example of how this problem was addressed already is the government website. They have collected data and made time plot series to visualize the data on daily basis. This was done for daily cases, daily deaths and the number of tests that was done in particular day compared to the capacity of testing. This is a good approach for this problem as its interactive and color coded which gives good feel of the system**. However, these systems do not show any predictions, they only analyze current data**.

Therefore, I have also found source that describes how they have approached similar problem if machine learning can be used for prediction of disease outcome. They path they took involved cleaning gathered data. Then using different algorithms to make models to compare them for accuracy. Lastly, they produced comparisons with different elements of predicted models.

This problem will involve looking into areas like machine learning algorithms that will be used to make models. I will approach this using a strategy of implementing most relevant algorithms and see how they perform. It might be that some algorithms work better for some prediction models then others. Decision trees would be useful to visualize how some concepts contribute into prediction. This needs to be analyzed in detail by doing research on how these algorithms perform in different projects and how they can be adjusted to be beneficial in this particular project.

***All found sources are listed at the bottom of this file in bibliography section*

Approach

This project will be approached using iterative methodology as it will require lots of testing and going back and improving the features making sure that it produces correct/optimal results. As from managing side of the project, I will keep track of the project and its progress using GitHub which also helps with version control and to keep track of what was modified and when. I will also keep records on spreadsheet with deliverables that should be achieved by specific date. The requirements will be broken down into sub requirements which can be implemented using 2 weeks sprints. These sprints will be helpful to manage project as it will allow to review at end of each 2 weeks sprints and analyze at what stage the project is currently and how to progress further. This approach will help me with achieving the aim. As from developing side of the project, it will be approached using historical data from pass events, data analytics and Machine learning. The implementation will be done using R language in R studio IDE.

Moreover, I will submit an ethical approval to University as I will be using data which will be used for analyzing and modeling. Also, I might use participation of users to test UI of the system to gain feedback which will help me to make improvements.

Evaluation

As to evaluate the project, I will try to break it down into individual components to make the evaluation easier. Firstly, the UI of the system will need evaluating to ensure that its consistent, easy and smooth to use. This can be done by doing user testing and see how they perform using the system. Another section for evaluation would be how the information is presented. Meaning how its analyzed, graphs and other information that is presented should be elegant and free of bugs. Lastly, the prediction models should also be evaluated. This could be done by using test sets of data and compare it to actual outcome. Evaluation should be done towards the end of this project as it would be easier to complete it when more features are implemented and presented as a whole. However, testing should be done throughout the project as it will improve the quality of code and it will reduce the number of bugs in the program.

Bibliography

1. Worldometer (2020). *Coronavirus Toll Update: Cases & Deaths by Country of Wuhan, China Virus - Worldometer*. [online] Worldometers.info. Available at: https://www.worldometers.info/coronavirus/.

2. coronavirus.data.gov.uk. (n.d.). *Coronavirus (COVID-19) in the UK*. [online] Available at: https://coronavirus.data.gov.uk/deaths.

3. Glander, S. (n.d.). *Can we predict flu deaths with Machine Learning and R?* [online] Available at: https://www.r-bloggers.com/2017/06/can-we-predict-flu-deaths-with-machine-learning-and-r-2/?utm_source=feedburner&utm_medium=email&utm_campaign=Feed%3A%20RBloggers%20%28R%20bloggers%29.

4. Shepperd, M. (n.d.). *3.1 Why does it [engineering] matter? | CS5702 Modern Data Book*. [online] *bookdown.org*. Available at: https://bookdown.org/martin_shepperd/ModernDataBook/C3-Matter.html.

Department of Computer Science