**PROJECT 3 DOCUMENTATION**

**WINDOWS SECURITY CHECK – MACHINE LEARNING APPROACH**

**Group – 7 : Sai Tarun Sathyan (SS4005)**

**1. Executive Summary**

Computers are a big part of our day-to-day life. We use computers for every task and thus store a lot of personal and sensitive data on our computers, all the way from photos to bank details. Thus, to ensure the safety of our computers I present a tool to evaluate the safety of the computers we use. This tool is used for the security evaluation of the windows operating system in a personal computer or a laptop. The tool analyzes the systems features and parameters that may influence the security of the system, while also going through the installed applications and software that may affect the security of the system. This tool can be used by anyone and does not require expertise in the windows operating system. This tool can be used on stock devices by simply downloading it as an application.

The tool was originally designed as an expert system developed in Python. But such expert systems are resource intensive and not every computer can afford such hardware to smoothly run an expert system. Now I’ve trained a machine learning model using linear regression to predict the final security score based on the input metrics of the expert system.

**2. Requirements**

The expert system has been developed in python so the user must have python (latest version) installed on their device. Python IDEs like Pycharm, VScode or Sublime can also be used to make things easier. Last but not least the user requires a windows device for this expert system to work. A windows 10 system would be preferable, since this system was designed with windows 10 in mind.

The Machine Learning process was coded on Google-Collab in the python programming language. The libraries used for the machine learning project is as follows:-

* Pandas
* Numpy
* Seaborn
* Matplotlib
* Sklearn.linear\_model/preproccessing/postprocessing/model\_selection

**3. Specification**

The security metrics that I’ve finalized for this expert systems analysis are:

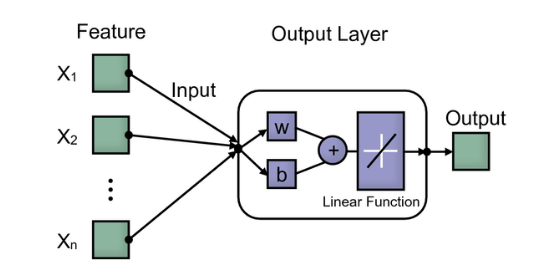
* Device Password Lock => Weight(2)
* Password Strength => Weight(1)
* Anti-Virus Check => Weight(2)
* Update Version => Weight(1)
* Firewall Status => Weight(2)
* Bit-locker Status => Weight(2)

These metrics were decided based on the different types of vulnerabilities a windows device could face. Even though most of it is software based, it is what dictate information security. Each metric has its own purpose to fulfil.

**4. Implementation**

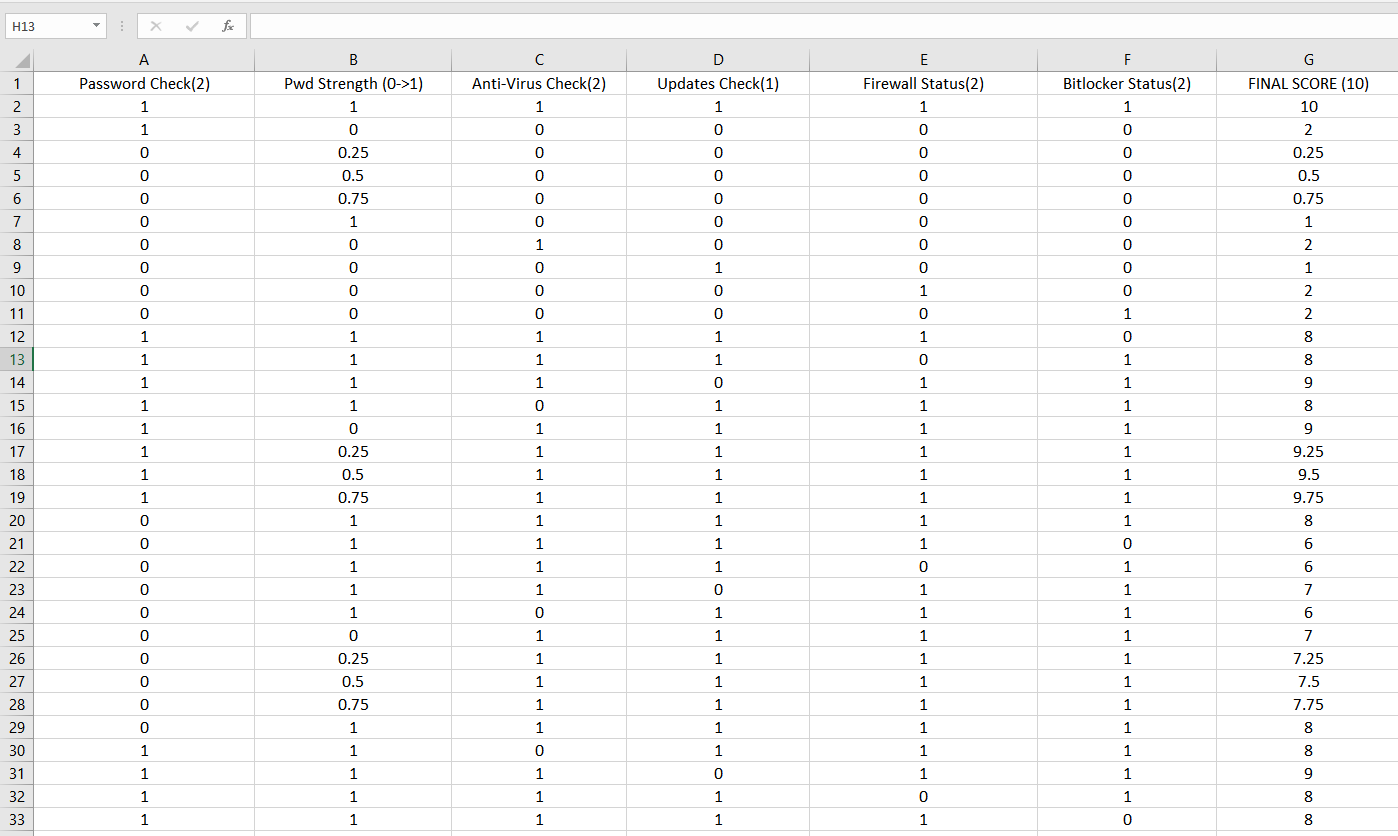
The Machine Learning program was designed using Google Collab using linear regression method. The program was coded in python version 3.7. The machine learning technique used is called simple linear regression, the linear regression model is used to estimate the values used to represent input data that is already present using the formula: y = X**β** + ε

The machine learning architecture looks like this:-



The inputs are the security metrics such as firewall status, anti-virus check, password check etc. Whereas the output layer is where the machine learning software computes the raw data and tries to make sense out of it. The output is the final security score that represents the cumulative score of the inputs.

The dataset used were the security metrics and the final score in excel format. The data set that was used to train looks like the following:



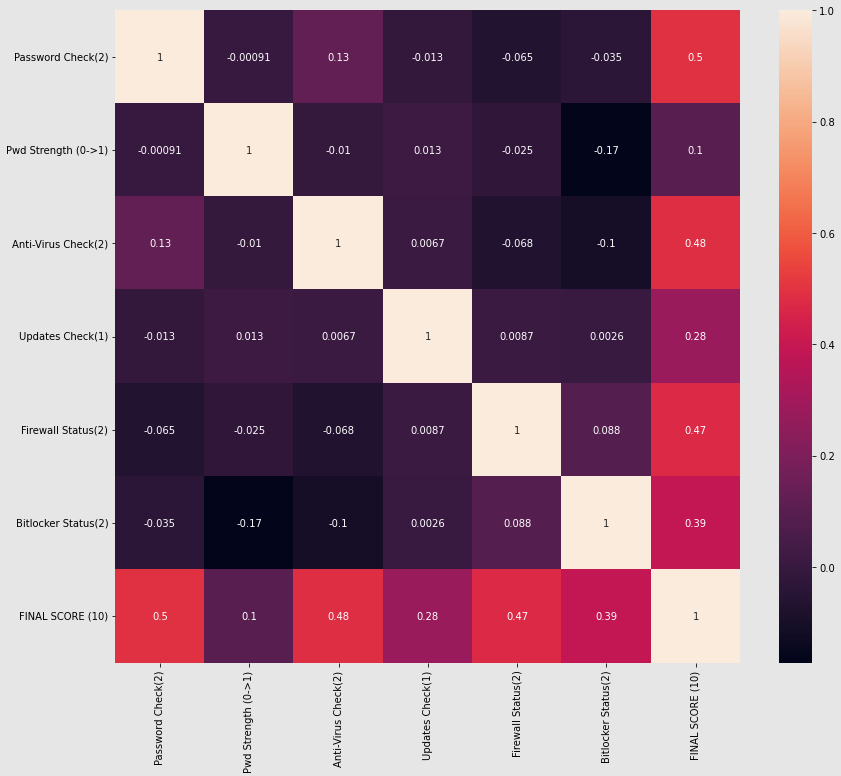
User Interface: UI for the project is just simple text and few plot showcasing the correlation between the security metrics and the final score.

Limitations: The limitations for this part of the project are not hardware based but I did face a time constraint, given more time I would like to add four more security metrics and implement a multivariate regression model or an artificial neural network model to identify more patterns in the data. But due to the lower amount of security metrics and the simplicity of the data at hand Linear regression seemed to be the right choice for the machine learning model.

Software & Hardware requirements: In terms of software any python IDE would do. Having said that I developed this project in the Google Colab IDE since my laptops CPU and GPU are on the weaker side, google provides the use of free GPU through Colab notebooks so training these Machine learning programs would be hassle free while using it. If you have a good PC then you can run this program on your system locally.

**5. Machine Learning Test Data**

This is a heatmap showing the correlation between each and every value including the final score:

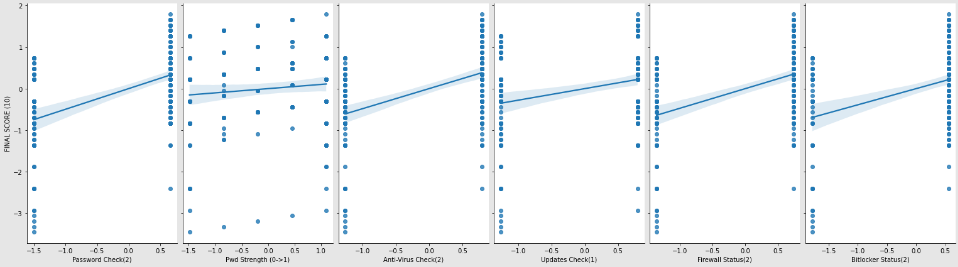


From this heatmap we can note how much each security metric is correlated to the final score and thus understand how important each metric is to overall security of the windows 10 system. The relation can go from 0 -> 1 and the correlation rating to the final score is as follows:

* Password Check -> 0.5
* Password Strength -> 0.1
* Anti-Virus Check -> 0.48
* Updates Check -> 0.28
* Firewall Status -> 0.47
* Bit-locker Status -> 0.39

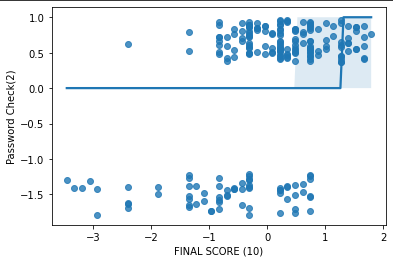
As we can see from the correlation data obtained the password being enabled for the device affects security the most. Obtaining 0.5 points of correlations to the final security score. The second most important security metric seems to be the Anti-virus software being installed and enabled on a windows device. The security metric that has the least effect on the overall security score is the password strength have a correlation rating of 0.1 to the final security score.

The linear pair plot for each security metric looks like this:

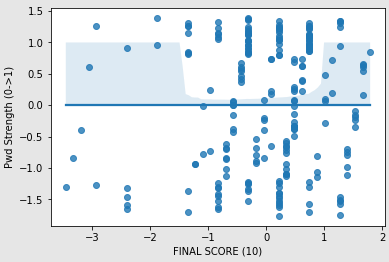


Due to most of the security metrics being binary the data we obtain from this does not have much value, but yet it is good to visualize how each security metric interacts with the final security metric in linear regression.

Since the device password check had the highest correlation to the final security metric, I decided to create a logistic plot to show how it varies. As the graph suggests the password check is a binary value starting at 0 and ending at 1. The higher the security metric the more we can see the password check being closer to 1. The password check logistic graph looks like this:



Similarly, I decide to create a logistic plot for password strength as well. Password strength is a security metric that is not binary an has multiple scores depending on how easy it is to predict your password. The password metric has four major output values they are 0, 0.25, 0.5, 0.75 and 1. The password strength logistic graph looks like this:



**6.Results**

The expert system code was tested with various conditions, such as switching off the firewall, disabling password protection, uninstalling antivirus software and updating the windows software to the latest version. Since most of these security metrics produces binary results such as Fire wall enabled or Fire wall disabled being the only two options. The output was either true or false. So, depending on these there are different possibilities for security scores. Password strength being the multi-variate metric resulted in four possibilities are 0, 0.25, 0.5, 0.75 and 1. Now having six different metric I added weights to each metric to make the final security score add up to a total of 10. The list of security metrics and their weights are as follows:

* Device Password Lock = 2
* Password Strength = 1
* Anti-Virus Check = 2
* Update Version = 1
* Firewall Status = 2
* Bit-locker Status = 2

A machine learning program using linear regression was used to analyse a huge list of input and output possibilities for these security metrics.

The results I obtained for the machine learning tests suggest that having a password lock and an anti-virus software installed matter the most amongst all other security metrics. As this pattern is correlated to a higher final score. From the tests conducted we can also say that the strength of your password is the least important security metric.

The results indicate a pattern of which matters the most and which metric matters the least. I personally agree with the results and in future I hope to add more security metrics to this expert system as well as create a more complex machine learning analysis of this input. Something like a multivariate regression or a neural network would provide much more insight into these security metrics.

**7. User’s Guide**

The user does not have to put any special input for the program. The user simply needs a windows device and python to be installed. Running the script would produce this output. That gives a security score and graph plots.

**8. Development Proccess**

The project was done by me alone, since I am working by myself in group 7. Pywin32 and Windows\_tools libraries were used to allow interaction with windows api through python. For the Machine learning part, I used google-colab online ide since my laptop was a bit slow. The following libraries: pandas, numpy, seaborn, matplotlib were used to code up the linear regression model and plot the pattern graphs. The machine learning analysis took me longer than expected but in the end I managed to complete it with detailed insight about all the security metrics and how they affect the overall security of a windows system.