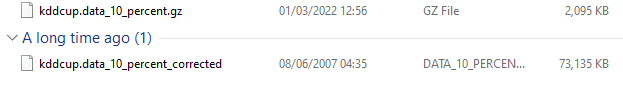
Element 1 ICA – KDD Cup Data from 1999 – Report

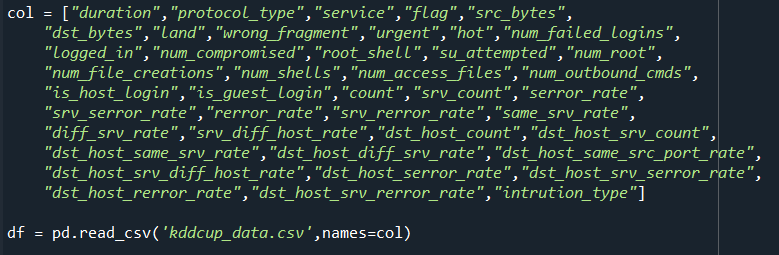
During the first element of the ICA, I was tasked with downloading a dataset and cleaning it, as well as doing an exploratory analysis. I countered many problems, from the first issue being separating the data into columns on Excel, to figuring out what the column header names would be.

Before I could start writing code to start analysing the dataset, I needed to separate it into columns. To do this, I downloaded the dataset file, extracted it to get this file (corrected file):



Once I did this, I opened the file in Excel and used the comma delimited tool to separate the data into columns to make it readable. The next issue I had to tackle was finding the suitable column headers, these were found on the website I downloaded the dataset itself (found here - <http://kdd.ics.uci.edu/databases/kddcup99/kddcup.names>).

Once I separated the columns and found suitable header names, I named each column through Spyder using this code:



After assigning the columns headers, I had to start attempting at visualising the data, to make the graphs for the user so they can visualise it better. I struggled with making the graphs and using the matplotlib module, although I had some help from the internet and produced a couple of graphs as shown here. There is one bar chart, one pie chart, and a line chart that is configurable to the user so they can filter the data to see what data they want. An example of one of the graphs are shown here:

Chart, bar chart, histogram

Description automatically generated

Main Findings

When working through this element of the ICA, I countered a lot of information regarding this dataset. Firstly, the website given to us itself contains a lot of data, including the headers of the columns. This was used to name the columns as I said above. Secondly, after doing a lot of research into what the data is and meant, I found out that it contains a standard set of data to be audited. This includes a wide variety of intrusions simulated in a military network environment. This is military data and data about simulated attacks used.

Having this knowledge about what the data frame, meant I could go into this project having a better understanding what I was plotting for my graphs, and it helped with splitting the data into columns to make it easier to read.

Issues I Faced

With regards to the issues, I faced plenty that I managed to overcome with a little bit of effort and research. One issue I got stuck with right at the start was cleaning the dataset and separating them into columns. I solved this with using Excel’s comma delimited tool which separated the data based on where they were with commas. Here is the tool:

Graphical user interface, text, application

Description automatically generated

Another issue I faced with this task was analysing the full dataset and knowing which commands to use. I solved this issue with doing each week’s tutorials and this helped me better understand what the functions meant. For example, using functions such as “df.head” (returns the first ‘n’ number of rows) or “df.tail” (returns the last ‘n’ number of rows). This can be seen here when doing a full analysis of the dataset:

Text

Description automatically generated

In addition to these two issues, another one I faced was using matplotlib. Matplotlib is a Python module used to visualise data in graphs. I had little experience with using this module so getting the hang of using it was a bit tricky. However, using the internet to help me with understanding how the module works I managed to make various graphs with my program. I managed to make a bar chart as shown above, a pie chart and a line chart that has custom filtering options so the user can choose what data they want to plot from the data frame.

One more issue I faced in this task, was the loading times for the graph to show. The loading time for the graphs were about 20 mins, and I couldn’t seem to find a quicker way since the dataset is so big. The graph did eventually appear, however.

Presentation and Analysis of Results

To present the results I collected from this task, what I designed was a console menu. The user would start the program and be welcomed into a simple dashboard consisting of five options each doing a different task. Before presenting the results to the user, the dataset was cleaned, so when visualising the results all the data was readable.

The five options included displaying a bar chart with pre-set plots (duration against protocol type), a pie chart with pre-set plots (number of file creations against count) and finally a line graph that has a custom filter option. This allows the user to pick any two headers and allow it to be plotted onto a dotted line graph. The fourth option is to print a full analysis of the data frame by using the “df.‘n’” functions. Finally, the fifth is a simple but essential option to exit the program.

Conclusion

In conclusion, during element one of this ICA I managed to analyse and clean a data frame, then plot the data visually in Spyder using Python modules such as Matplotlib. I was not surprised at any of my findings, except at first when reading the data frame and realising I could only find forty-one headers instead of forty-two since there were forty-two columns for the data. I discussed this with my professor and concluded that I had to make up a column name instead which came to “intrusion\_type.”

Aside from that, I was happy with my results and the finished product I came out with. I put as much effort in as I could with what resources I managed to access. In the end I came out with a working program that a user could successfully use to visualise data from that given data frame. Things I were surprised at when going through this task, was how much data was collected. There’s a lot of data to comb through, hence why my graphs take such a long time to load.