SIT719 Security and Privacy Issues in Analytics

Distinction/Higher Distinction Task 5.1 End-to-end project delivery on cyber-security data analytics

Overview

Do you know what is an end-to-end data science project? See the lifecycle of an end-to-end data science project. If you are doing data science application for security analysis, your problem will be related to the cybersecurity and your data analysis needs to follow the below steps. See the task description for the detailed instructions.

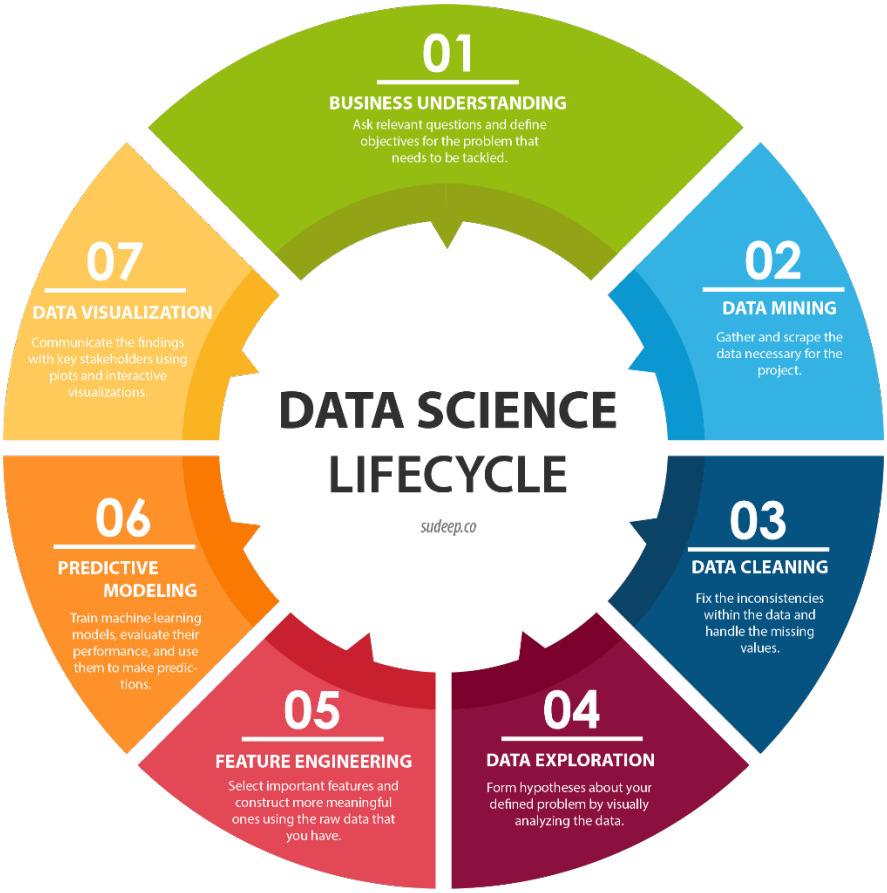


Figure 1: Data Science Lifecycle [source: Sudeep, 2019 (accessed Jan 2020)]

In this *Distinction Task*, you will experiment with Machine Learning classification algorithms. Please see more details in the Task description. Before attempting this task, please make sure you are already up to date with all previous ***Credit and Pass tasks***.

Task Description

Instructions:

Suppose, you are working in an organization as a security analyst. You need to conduct an end to end project on “cyber-attack classification in the network traffic database”. To complete the project you follow the steps in Figure 1. Here, **some of the steps are already solved for you (by the teaching team, you don’t need to take any action**) and the remaining (Step 4 and 6, 7) you need to complete (highlighted in blue) by yourself to submit this task.

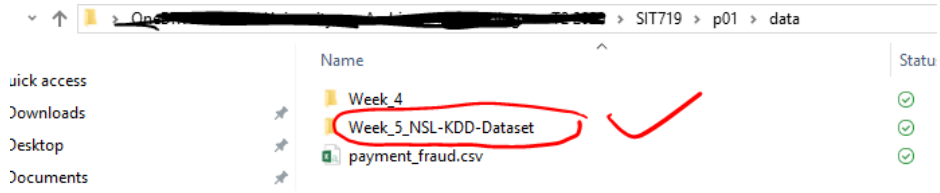
Step 1: Business Understanding (Problem Definitions)

Your task is to develop a 5-class machine learning-based classification model to identify the normal network traffics and attack classes.

Step 2: Data Gathering (Identify the source of data)

In the industry/real-world, you need to communicate either with your manager, client, other stakeholders and/or IT team to understand the source of data and to gather it.

Here, the teaching team already gathered data for you. You can access the dataset from the given github data folder. Within the data folder, go to the “Week_5_NSL-KDD-Dataset” subfolder.



If you are interested to learn more about the dataset, please visit the website below (not mandatory for the HD task):

https://www.unb.ca/cic/datasets/nsl.html

*****A starting example code for the 5 class classification is also given for your convenience, where some of the steps are already implemented. Please see the “SIT719_Prac05_Task02_HD_task_sample_done” notebook file (obtain from the github link).**

Step 3: Data Cleaning (Filtering anomalous data)

In a typical analysis, you may need to take care of missing values and inconsistent data. In week 2, you have learnt how to deal with missing values and manipulate a database. **Here, it has already been taken care of for this dataset (so no action is needed for this task).**

Step 4: Data Exploration (Understanding the data)

Here, you need to do the following tasks and write in your report:

1. Identify the attribute names (Header)
2. Check the length of the Train and Test dataset
3. Check the total number of samples that belong to each of the five classes of the training dataset.

Step 5: Feature Engineering (Select Important Feature)

In a typical setup, you may need to do feature extraction or selection during your data analysis process. Here, relevant feature engineering is already done for you in the sample code. So, **no action is needed for this task**

Step 6: Predictive Modelling (Prediction of the classes)

The DecisionTreeClassifier has been implemented for you. Now, you need to implement other techniques and compare. Please do the following tasks:

1. Implement **at least 5 benchmark classification** algorithms.
2. Tune the parameters if applicable to obtain a good solution.
3. Obtain the confusion matrix for each of the scenarios.
4. Calculate the performance measures for the each of the classification algorithms that includes Accuracy (%), Precision (%), Recall (%), F-Score (%), False Alarm- FPR (%)

You need to compare the results following the table below. Create one table for each algorithm.

Attack Class	Accuracy (%)	Precision (%)	Recall (%)…	…	…	…	...
DoS							
Normal							
Prob							
R2L							
U2R							

Finally, you summarize the results similar to the below table:

Algorithms	Accuracy (%)	Precision (%)	Recall (%)…	…	…	…	...
Alg 1							
Alg 2							
…							
…							
…							

Your results need to be comparable against benchmark algorithms. For example, see the below results obtained from a recent article “An Adaptive Ensemble Machine Learning Model for Intrusion Detection” published in IEEE ACCESS, July 2019.

TABLE 6. Result of each algorithm on KDDTest+.

Algorithms	Accuracy	Precision	Recall	F1	Time(S)
DeciTree	79.71%	83.51%	79.72%	77.31%	6.34
RanForest	76.64%	81.85%	76.64%	72.17%	1.86
kNN	75.51%	80.97%	75.51%	71.41%	86.49
LR	73.58%	74.65%	73.58%	69.13%	43.77
SVM	74.09%	80.91%	74.09%	70.38%	1785.2
DNN	81.6%	84%	81.6%	80.18%	227.8
Adaboost	76.02%	81.82%	76.02	72.12%	265.1

Step 7: Data Visualization

Perform the following tasks:

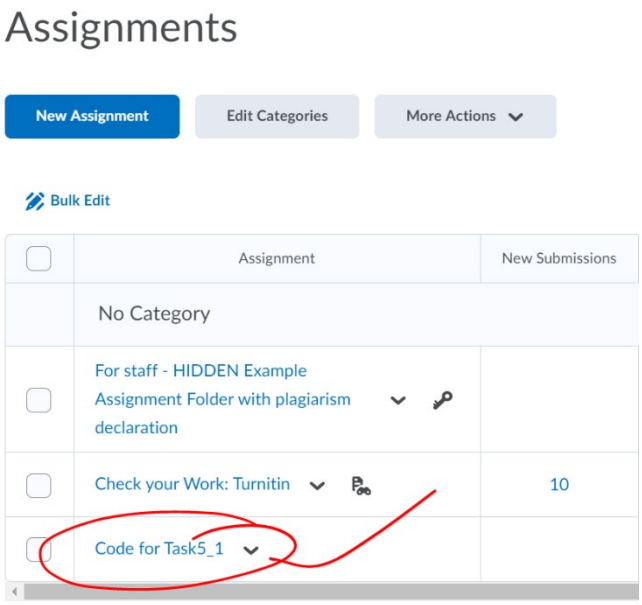
1. Visualize and compare the accuracy of different algorithms.
2. Plot the confusion matrix for each scenarios.

Step 8: Results delivery:

Once you have completed the data analysis task for your security project, you need to deliver the outcome. In real-world, results are typically delivered as a product/tool/web-app or through a presentation or by submitting the report. However, in our unit we will consider a report based submission only.

Here, **you need to write a report (at least 3000 word)** based on the outcome and results you obtained by performing the above steps. The report will describe the algorithms used, their working principle, key parameters, and the results. Results should consider all the key performance measures and comparative results in the form of tables, graphs, etc.

Submit the PDF report through onTrack. You also need to submit the code separately (within the “Code for task 5.1” folder) under the assignment tab of the CloudDeakin python script during submission.



Please note, it is a graded task where you will receive some feedback and marks. Your tutor/marker will assign you some marks based on the quality of your submission, performance of your algorithms, selection and novelty in your algorithm, tuning and understanding the algorithms, data exploration, how well you have explained the results, your usage of scientific language, authenticity of the claims and finally the aesthetic look of your submission and reflection of the quality of your work from the tutor's judgement. You will receive the feedback based on the following marking rubric. The marker will judge how you have performed in the following categories.

Marking Rubric:

Criteria	Unsatisfactory – Beginning	Developing	Accomplished	Exemplary	Total
Report Focus: Purpose/ Position Statement	**0-7 points**	**8-11 points**	**12-15 points**	**16-20 points**	/20
	Fails to clearly relate the report topic or is not clearly defined and/or the report lacks focus throughout.	The report is too broad in scope (outside of the title topic) and/or the report is somewhat unclear and needs to be developed further. Focal point is not consistently maintained throughout the report.	The report provides adequate direction with some degree of interest for the reader. The report states the position, and maintains the focal point of the analysis for the most part.	The report provides direction for the discussion part of the analysis that is engaging and thought provoking, The report clearly and concisely states the position, and consistently maintain the focal point.	
Comparative analysis and Discussion	**0-14 points**	**15-24 points**	**25-39 points**	**40-50 points**	/50
	Demonstrates a lack of understanding and inadequate knowledge of the topic. Analysis is very superficial and contains flaws. The report is also not clear.	Demonstrates general understanding of python scripting. Analysis is good and has addressed all criteria. Comparative analysis is presented. Sufficient discussion is also presented.	Demonstrates good level of understanding of python scripting. Algorithms are fine-tuned and comprise good selection of algorithms. Comparative results are presented using standard performance measures.	Demonstrates superior level of understanding of python scripting and algorithms. Algorithms are fine-tuned with some novelty or hybridization or advanced and/or recent algorithm. Comparative results are presented using performance measures in a way that it provides very clear and meaningful insights of the output.	
Writing	**0-10 points**	**11-17 points**	**18-21 points**	**22-30 points**	/30

Quality & Adherence to Format Guidelines	Report shows a below average/poor writing style lacking in elements of appropriate standard English. Frequent errors in spelling, grammar, punctuation, spelling, usage, and/or formatting.	Report shows an average and/or casual writing style using standard English. Some errors in spelling, grammar, punctuation, usage, and/or formatting.	Report shows above average writing style (can be considered good) and clarity in writing using standard English. Minor errors in grammar, punctuation, spelling, usage, and/or formatting. Author has demonstrated the use of scientific language and results are well explained.	Article is well written and clear and standard English characterized by elements of a strong writing style. Basically free from grammar, punctuation, spelling, usage, or formatting errors. Author has demonstrated advanced use of scientific language and results are well explained with insights.	

Rubric adopted from: Denise Kreiger, Instructional Design and Technology Services, SC&I, Rutgers University, 4/2014