Delft University of Technology

Reliability of Sustainable Power Systems
EE3065TU

# Practicum 7: Security Assessment of Electrical Power Systems

October 17, 2023

# Session 7: Security assessment of electrical power systems

This practicum session is mainly related to the course reader for lecture 7. Python will be used to construct the machine learning model and perform the power system security analysis. *Sckit learn library* will be used as well. Scikitlearn is a machine learning Python library that provides efficient models for both supervised and unsupervised learning methods. Nevertheless, a coding environment will be used which has already installed the required packages. A brief description on how to download the initial files from Brightspace, and how to upload and work from Google Colab as follows:

1. The file as well as the corresponding data-sets can be found in *Brightspace* > *Content* > *Practicum* > *Practicum 7 material*.
2. The coding environment to be used is Google Colab. You can access through the following link: https://colab.research.google.com.
3. Once in Google Colab, you can press *File* > *Upload Notebook* and select the provided .jpynb file.
4. In the Files tab (see Figure 1) you can upload the data-set files.
5. Now everything is set up to address the assignment tasks.

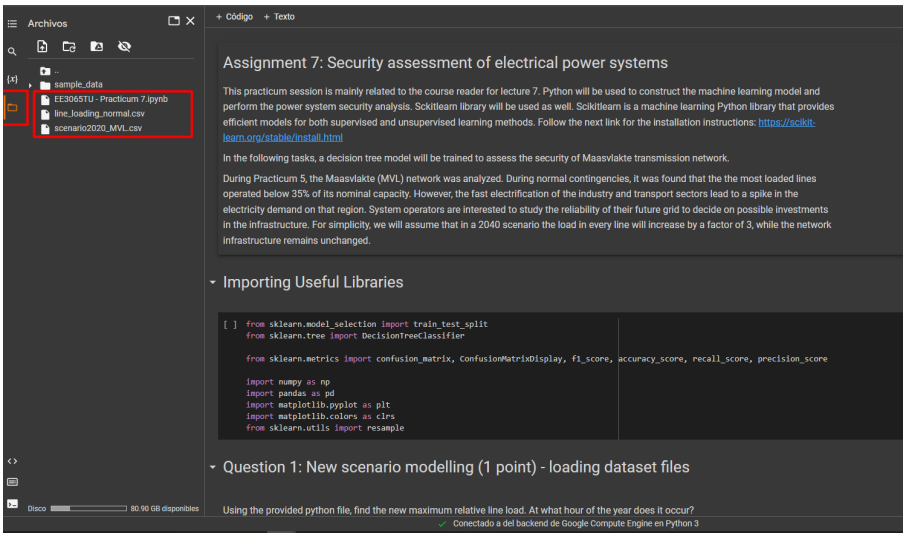


Figure 1: Google Colab - Files tab

In the following tasks, a decision tree model will be trained to assess the security of Maasvlakte transmission network (the same network as in Practicum 5 and Figure 5.4 of the course reader - see Figure 2).

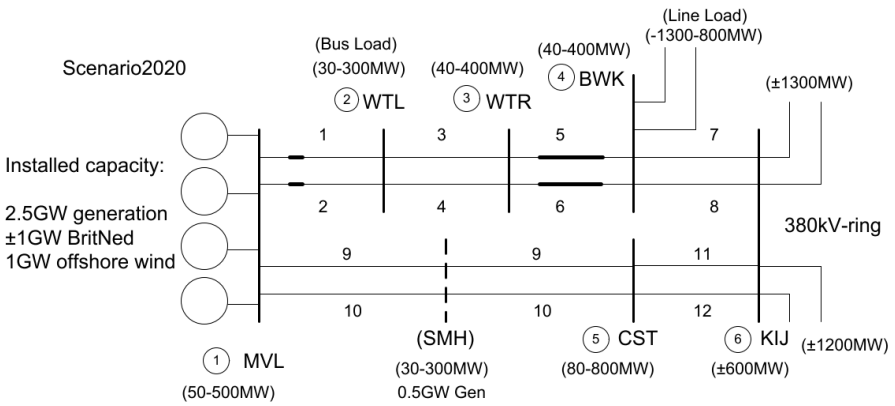


Figure 2: Massvlakte Transmission Network

# Assignments:

During Practicum 5, the Maasvlakte (MVL) network was analyzed. During normal contingencies, it was found that the the most loaded lines operated below 35% of its nominal capacity. However, the fast electrification of the industry and transport sectors lead to a spike in the electricity demand on that region. System operators are interested to study the reliability of their future grid to decide on possible investments in the infrastructure. For simplicity, we will assume that in a 2040 scenario the load in every line will increase by a factor of 3, while the network infrastructure remains unchanged.

## Question 1: New scenario modelling (1 point) - loading dataset files

Using the provided Jupyter Notebook file, find the new maximum relative line load. At what hour of the year does it occur?

## Question 2: Database pre-processing - creating system security labels (2 points)

The Transmission System Operator (TSO) has set the line loading operational limit to be 70%. Therefore, lines operating above that threshold will be considered as insecure, while those above 100% are on unstable region. Implement a script to label the operating conditions in terms of its security level. From the training data, how many operation points represent an insecure condition (answer in %)? How many hours does the system operates in the insecure region (answer in hours)?

## Question 3: Training and testing data-set (1 point)

Within the provided script, the following code line is included:

*X_train, X_test, y_train, y_test = train_test_split(X_dataset, Y_dataset, test_size=0.3, random_state=3065, shuffle=True)*

Describe what the function does, and why we need the variables X_train, X_test, y_train, y_test. You can use Scikit learn package available documentation (you can use the following link: https://scikit-learn.org/stable/index.html and use the search bar to find the aforementioned function).

## Question 4: Assess the performance of ML model with the Confusion matrix (2 points)

In order to assess the performance of the Decision Tree model, calculate the recall, the precision and the F1 score for both the training and the testing data-sets.
Which metric would be more suitable to analyze the impact in the context of the Maasvlakte network? Motivate your answer.

Using the provided plot function create two independent plots to show the Predicted Classes and the Actual Classes scattered plots. Compare both figures and explain whether there is any difference between them.

## Question 5: Risk of applying ML models (1 point)

From the TSO point of view, a false alarm (False Positive -FP-) does not have the same impact as an insecure operation that was not predicted (False Negative -FN-). Lets assume that the impact of a FP is 100 $EUR/FP$, while for FN is 1000000 $EUR/FN$. What is the risk of applying this DT model? What is the contribution of FP to that risk (answer in %)?

## Question 6: Sensitivity of the insecurity-threshold (1 point)

The regulatory introduces very high penalty costs for a loss of loads and requires the TSOs to operate with a high security of energy supply. As a result, the TSO operates more conservatively the grid. The new operational

limit for the lines should be set to be 50 % for each line.

Re-do questions 3 to 5 with the new operational limit. What are the consequences of lowering the threshold? How does the change impact affect the DT performance? Base your answer on the performance metrics as well as in the risk of applying the new model.

## Question 7: Dealing with imbalanced data-set (2 points)

Aiming to improve the model performance, the TSO suggests to implement either a down-sampling, up-sampling or over-weighting approach. Implement one of the three techniques and analyze the results. Explain the benefits and drawbacks of the selected approach, including a comparison to the base case. Use the original threshold of 70%.
A hint on how to implement each approach as follows:

- **Option A - Up-sampling**: As in the provided data-set the insecure are a much less-populated data points, one could create artificial data point when sampling real information is not feasible. You can use existing insecure data-points to create new artificial information to be used as input data for the training set.
- **Option B - Down-sampling**: Another approach would be to decrease the over-populated data points. Therefore, one could delete some of the secure data points so the population of each class can be balanced.
- **Option C - Over-weighting**: Check *DecisionTreeClassifier* parameters in the Scikit learn website (https://scikit-learn.org/stable/index.html). The overall idea is make insecure points have a higher class weight over the secure points.