MPA-MLF - Miniproject

Classification of wireless transmitters

Date: 11.3.2023

1 Task description

Your task will be to **determine the number of wireless transmitters** based on the hardware imperfections. You are provided with the dataset. The dataset consists of 19200 samples and 9 features in total. The features represent the main radio frequency impairments, such as:

- **cfo_meas** - Carrier Frequency Offset (CFO) between transmitter and receiver. Measured via vector spectrum analyzer, in Hz
- **cfo_demod** - Carrier Frequency Offset (CFO) between transmitter and receiver, measured after demodulation, in Hz
- **gain_imb** - Gain imbalance of modulator defined as:

$$gain\_imb = 20\log_{10}\left(\frac{g_i}{g_q}\right), \tag{1}$$

  where g_i, g_q are gains in I and Q path

- **iq_imb** - combination of gain imbalance and quadrature imbalance aggregated into one parameter. More information can be found e.g. in a document from Rhode & Schwarz company FSQ-K70/FSMR/FSU-B73 Vector Signal Analysis Software Manual (page 87)
- **or_off** - Origin offset, often known as DC offset. Represents how the constellation diagram is shifted from point [0+0j], expressed in dB's
- **quadr_err** - quadrature error imbalance - deviation of phase shift between I and Q components from 90 degrees
- **ph_err** - represents phase difference between received $Y(n)$ and ideal $X(n)$ constellation points:

$$ph\_err = \arg(Y(n) - X(n)) \tag{2}$$

- **mag_err** - Magnitude error between received $Y(n)$ and ideal $X(n)$ constellation points in QAM constellation:

$$mag\_err(n) = ||Y(n)| - |X(n)|| \tag{3}$$

- **evm** - Error Vector Magnitude measurements, representing the RMS (root mean square) error between received $Y(n)$ and ideal $X(n)$ constellation points in QAM constellation (see figure 1.) The EVM is computed for N total symbols such as:

$$\text{EVM} = \frac{\sum_{n=1}^{N}(Y(n) - X(n))^2}{\sum_{n=1}^{N} X(n)^2} \tag{4}$$

  EVM is an aggregated feature, and several transceiver impairments can contribute to it. The most important source of EVM increase is power amplifier nonlinearity.

The effect of gain imbalance, quadrature error and origin offset on the QPSK constellation is illustrated in Figure 2.

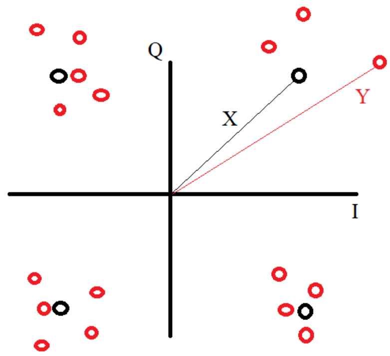

Figure 1: Original (black) and distorted (red) constellation points

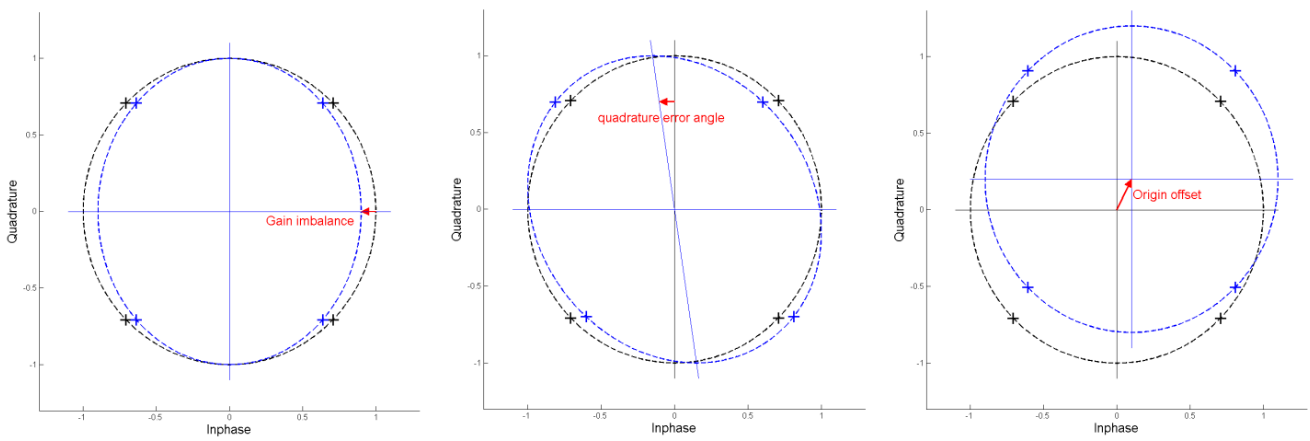

Figure 2: Effect of selected imbalances on QPSK constellation diagram

2 Steps

You are expected to perform the following steps:

- **Data examination** - Examine what data types you have in your dataset. What preprocessing steps appear to be the best ones?
- **Data preprocessing** - Perform the preprocessing,
- **Model selection** - Based on examined data and your task, select the optimal model/algorithm to complete the task
- **Model building and model training** - Build the model and fit it to the data

3 Submission and grading

You are required to work in pairs. Please make your pairs on your own and write down the teams in the following document: Link to the document Create your teams until **17.3.2023**. The deadline for the project submission is **31.3.2023**. You are expected to write a report(a maximum of 5 pages) describing the essential steps and strategies. You can receive a maximum of 15 points.
Your solutions will be submitted in two different ways:

- **Report, e-learning**. You will upload your report into the e-learning. Please upload your report in .pdf format
- **Code, GitHub, e-learning**. Please create a new folder in the repo you have used for MPA-MLP. Do not push the input dataset to your GitHub repo. **Link the folder with the code into your reports.**

4 General comments

- The report should have all the necessary formalities that a report of a similar type has (Introduction, problem description, the main body of your work, conclusions. . . ., all figures and tables should have labels and numbering and should be referenced in the text, etc..). You should be familiar with this from your bachelor's thesis.
- Do not include the screenshots of your code in the report. If you wish to describe some of your algorithms, use pseudo-code/flow charts.., Do not describe well-known algorithms.
- Don't present your work in the report as plain text. Use graphs, figures, and tables to show and present the relevant information.
- You are required to do all of your coding in Python. You don't have to do all coding from scratch, but you are all allowed to use Python's libraries and frameworks (For instance, *Keras*, *PyTorch*, *Scikit-learn..*).
- Usage of Google Collab is strongly recommended but not required.
- Ensure you properly describe your work in your report; we are more interested in the process of your work than in a correct result.