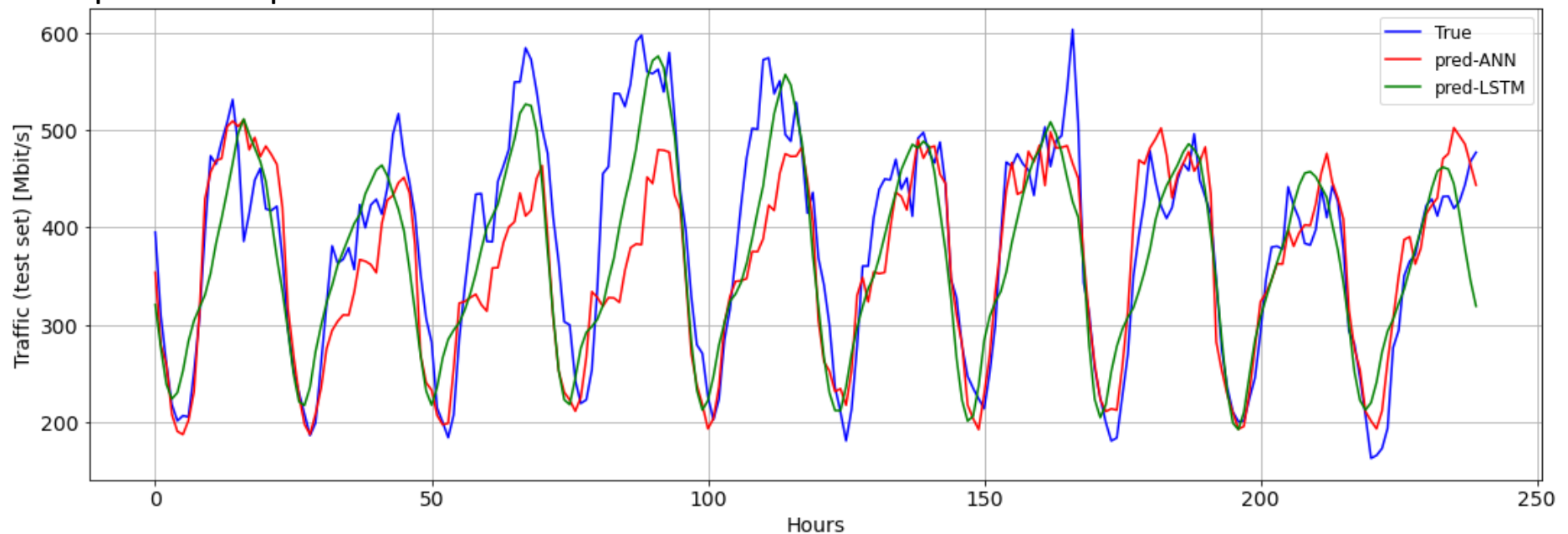


# Traffic prediction

## Task 8

8. Evaluate the impact of traffic over/under-estimation
  - a) Calculate min and max traffic values of the dataframe created in task 2b) and scaled ground-truth, ANN-predicted and LSTM-predicted traffic traces (test set) so as to have maximum traffic = 1 Gbit/s (so far we have worked with a dataset expressed in CDR units). Then, plot the three traffic traces in a single plot

Expected output:



# Traffic prediction

## Task 8a) - hints

1. Ground-truth ( $y_{test}$ ), ANN-predicted ( $y_{pred,ANN}$ ) and LSTM-predicted ( $y_{pred,LSTM}$ ) traffic you have used so far are normalized between 0 and 1

- First scaling step should be to convert  $[0,1]$  range into  $[\min, \max]$  range, (e.g., for  $y_{test}$ )

$$y_{unscaled}[CDR] = y_{test}[CDR] * (maxtraffic - mintraffic) + mintraffic$$

- where:

*maxtraffic = maximum traffic for the cell over the entire 61 – days*

*mintraffic = minimum traffic for the cell over the entire 61 – days*

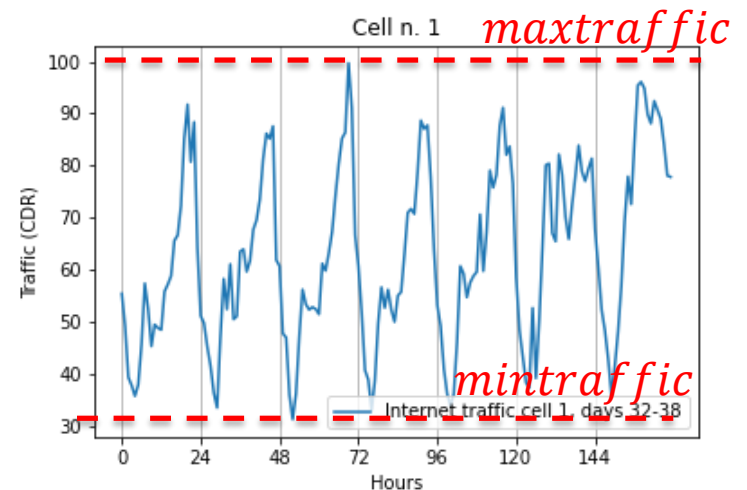
2. Traffic up-scaling (i.e., from CDR into Gbit/s units) must be done so as to have the maximum traffic along the entire period of 61 days equal to 1 Gbit/s

- Example: assume  $maxtraffic = 100$  CDR (see fig.)

- For a generic traffic value  $y$  [CDR], you should obtain the final upscaled value as

- 

$$y [Gbit/s] = y_{unscaled}[CDR] * \frac{1 \text{ Gbit/s}}{100 \text{ CDR}}$$



# Traffic prediction

## Task 8

8. Evaluate the impact of traffic over/under-estimation
  - b) Define function *evaluate\_cost()* that takes in input ground-truth, ANN-predicted and LSTM-predicted traffic traces (scaled as in task 8a) and two cost parameters *alpha* and *beta* for over/under-provisioning and returns cost of over/under-provisioning for the ANN and LSTM cases, assuming a given resource allocation policy
    - See details in the skeleton code
  - c) Use function *evaluate\_cost()* with given over/under-provisioning cost weights using ground-truth, ANN-predicted and LSTM-predicted traffic traces above; plot results in a 3D graph

Expected output:

