POLITECNICO
MILANO 1863

Quality of Transmission (QoT) estimation in optical networks

Francesco Musumeci

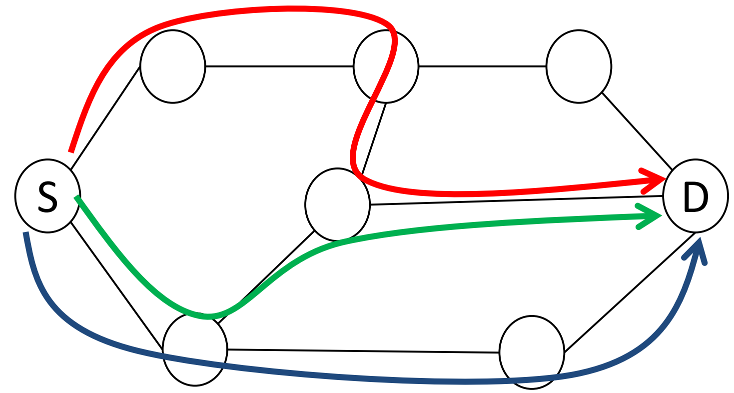
Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB)

Politecnico di Milano, Milano, Italy

Network Data Analysis Laboratory

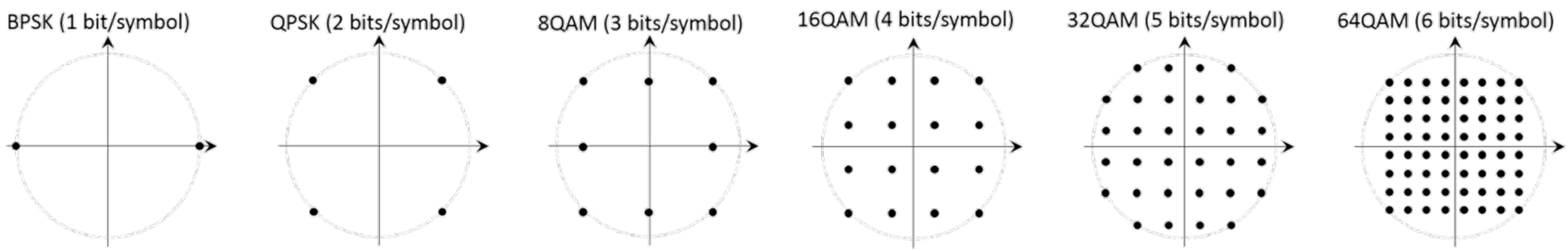
QoT estimation
Background

- When routing a lightpath (optical channel) in an optical network between a source/destination pair, multiple options are available:
  - Physical route
  - Modulation Format (MF)
  - …
- Possible impact from:
  - Other active channels
  - No. of traversed devices (amplifiers, fibers, switching devices, …)
- Each combination provides a different signal quality (e.g., SNR, Bit Error Rate, …)



What is the best combination?

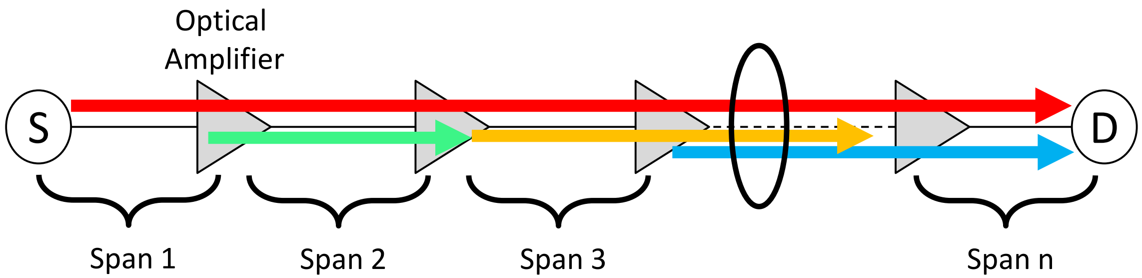
- **Trade off**: high-order MF is desirable to have higher spectral efficiency (higher bit rate in the same spectral resources)…
- …BUT it is more sensitive to noise

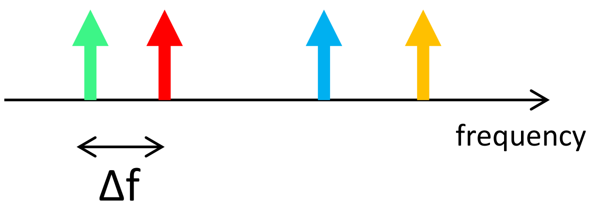


QoT estimation
Background

- How to select path and MF?
- MF is constrained by quality of transmission (QoT) of the received signal (e.g., SNR)
- In turn, signal quality is affected by:
  - Noise of traversed optical amplifiers
  - Interfering channels (co-existing lightpaths),
    - how many? how much close in frequency?

MF	Required SNR
QPSK	11 dB
8QAM	15 dB
16QAM	18 dB
32QAM	20.8 dB
64QAM	23.7 dB





→ It's desirable to know the QoT (SNR) ***before*** establishing a lightpath

- Different ways to solve the problem:
  1. **Analytical models**: time consuming, involve huge set of parameters to be known, often uncertain
  2. **Design with margin** (simpler approximate models): fast but less accurate (→margin is required), lead to resource underutilization; not developed for a specific network topology → may fail in generalizing
  3. **ML predictors**: can be trained for the specific network and approximate better the required *margin*

QoT estimation
Dataset

- 1034 lightpaths routed in different network topologies
- Raw data:

One row = one lightpath

```
60 60 60 60 60; 5; 50.0129; 21.4016
70 70 60; 14; 50.0129; 21.7621
50 50 50 50 50 50 50 50 50 50 50 50 50 50 50; 17; 50.0129; 19.3127
70 70 70 70 70; 29; 50.0129; 21.0628
70 70 70 70 70; 9; 50.0129; 21.2568
...
```

Set of fiber spans w/ length [km]

Max no. of interferers (other lightpaths) along the route

Δf from the closest interferer [GHz]

Lightpath SNR [dB]

QoT estimation
Dataset

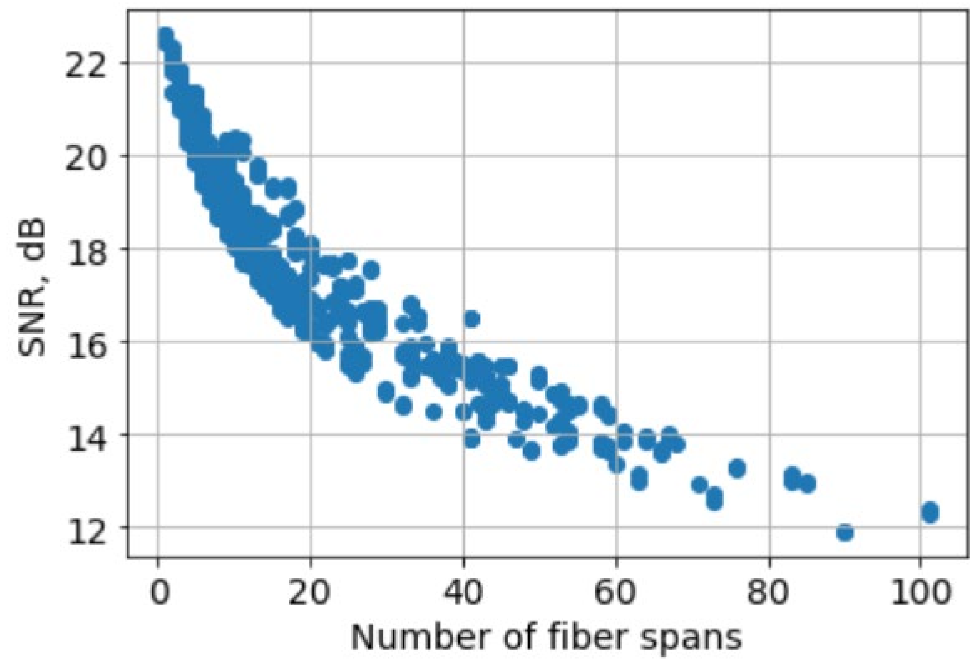
- 1034 lightpaths routed in different network topologies
- Raw data:

One row = one lightpath

```
60 60 60 60 60; 5; 50.0129; 21.4016
70 70 60; 14; 50.0129; 21.
50 50 50 50 50 50 50 50 50
70 70 70 70 70; 29; 50.012
70 70 70 70 70; 9; 50.0129
```

High correlation between no. of spans and lightpath SNR

Max no. of interferers
(other lightpaths)
along the route



QoT estimation
Lab overview

1. Task 1. Raw data visualization
2. Task 2. Dataset generation and visualization
3. Task 3. ML algorithms training/testing
4. Task 4. Evaluate the impact of using one single feature
5. Task 5. Evaluate the impact of inaccurate predictions
6. Task 6. Performance evaluation
7. Task 7 (**HOMEWORK**). Evaluate the impact of *uncertain* fiber span length

QoT estimation
Task 1

1. Observe raw data characteristics
   a) Define function *read_dataset()* that reads a file passed in input and returns 3 lists with lightpaths characteristics (spans, interferers, SNR)
      - See details in the skeleton code
   b) Call function *read_dataset()* using input file in the skeleton code, check dimensions of returned values and calculate/print mean, variance and std dev for
      a) number of spans,
      b) lightpath length,
      c) number of interferers
      d) snr

      across all lightpaths
      - **Already given in skeleton code**

QoT estimation
Task 1a)-b): expected outputs

```
Length of span list: 1034
Length of interferers list: 1034
Length of snr list: 1034
**********
Number of spans: mean=15.83, var=243.23, std=15.6
Lightpath length: mean=945.05, var=721041.32, std=849.14
Number of interferers: mean=41.52, var=231.66, std=15.22
SNR: mean=18.37, var=4.46, std=2.11
```

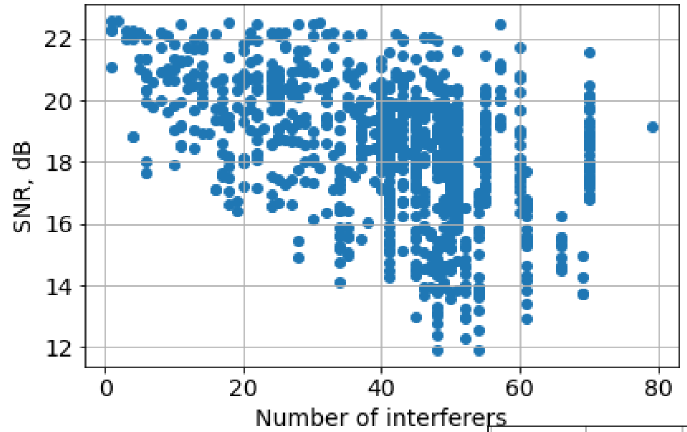
QoT estimation
Task 2

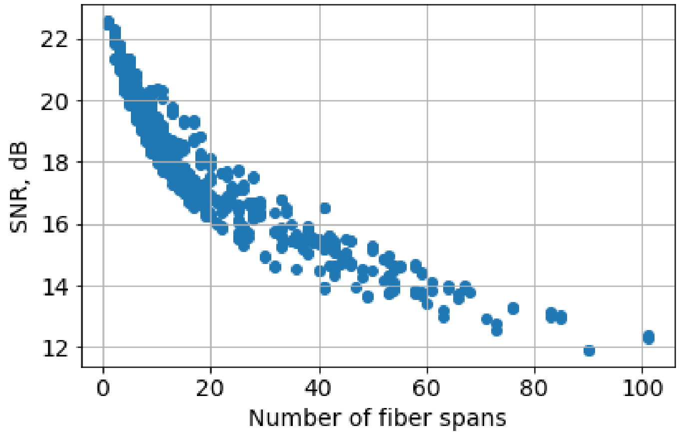
2. Features matrix generation and data visualization
   a) Define function *extract_features()* that takes in input the lists of features (i.e., excluding SNR) obtained in task 1b) and returns a numpy array including the features as columns
      - See details in the skeleton code
   b) Call function *extract_features()* using the lists obtained in task 1b), verify shape of the returned numpy array and calculate mean, var and std dev for all features across all the lightpaths
   c) Draw 5 scatterplots with distribution of SNR values against each feature
      - **Already given in skeleton code**

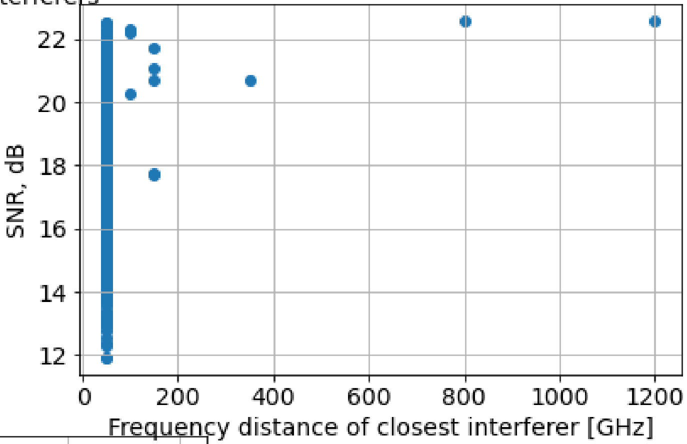
QoT estimation
Task 2a)-b): expected outputs

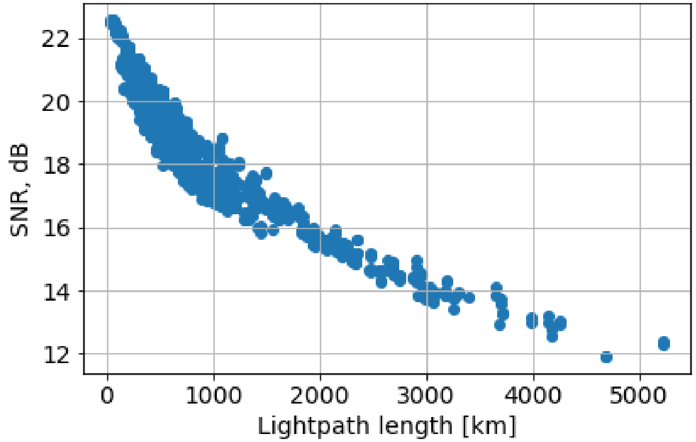
```
Shape of features matrix: (1034, 5)
*************
Number of spans: mean=15.83, var=243.23, std=15.6
Lightpath length: mean=945.05, var=721041.32, std=849.14
Longest span length: mean=71.57, var=76.2, std=8.73
Number of interferers: mean=41.52, var=231.66, std=15.22
Freq distance from closest interferer: mean=52.8, var=1960.32, std=44.28
```

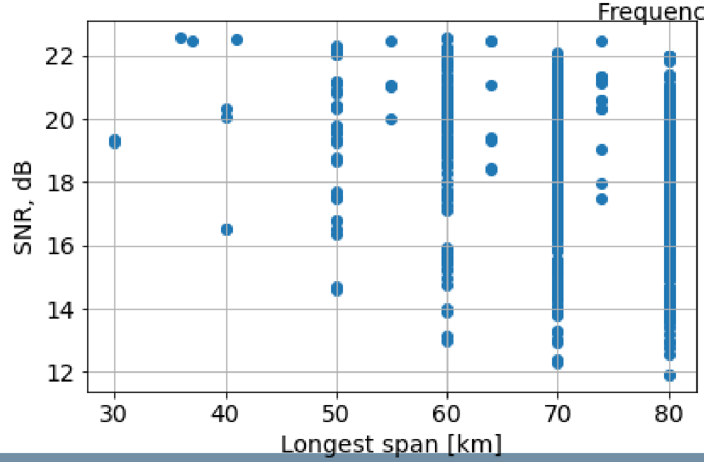
QoT estimation
Task 2c): expected outputs











QoT estimation
Task 3

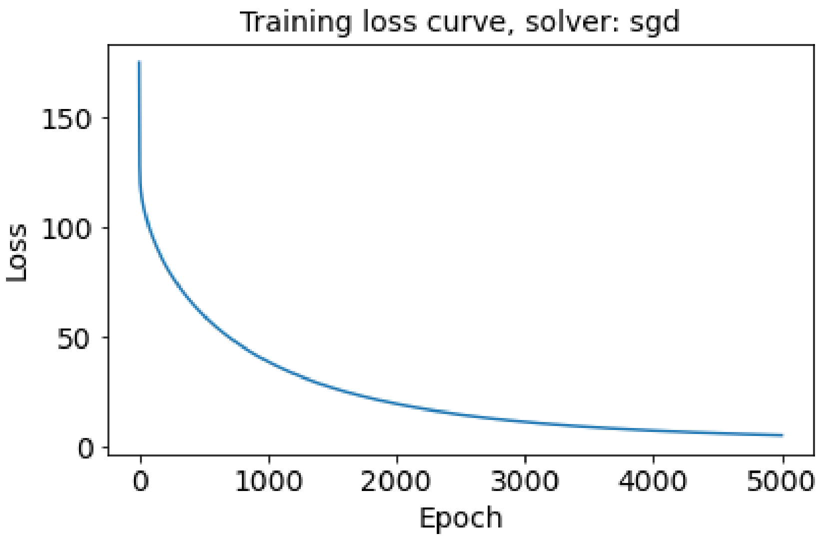
3. ML algorithms training/testing (Neural Network)
   a) Define function *train_NN()* that takes in input NN hyperparameters, performs training using a training set passed in input and provides training results
      - See details in the skeleton code
   b) Scale features so as to have 0 mean and unit variance.
      Split dataset (features matrix and output vector (SNR) retrieved in tasks 1b) and 2b)) into training and test sets (80/20%).
      Call function *train_NN()* using scaled features matrix and output (SNR) and with two different solvers (‘sgd’, 'adam‘*)

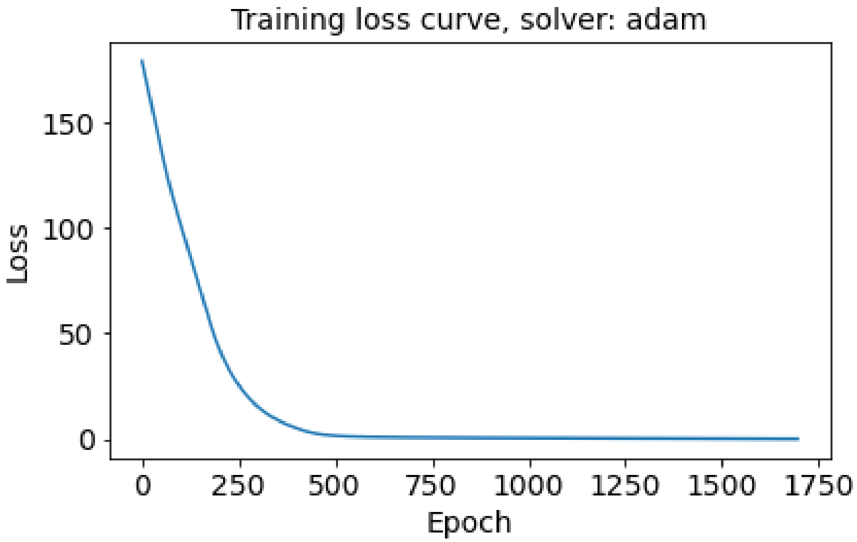
*Diederik P. Kingma, Jimmy Ba, "Adam: A Method for Stochastic Optimization“, available at https://arxiv.org/abs/1412.6980

QoT estimation
Task 3a)-b): expected outputs

Training with solver: sgd
Final loss: 5.179096527026564
Total number of iterations: 5000
Current loss: 5.179
Best loss: 5.179
Training time [s]: 15.482
Final training R2 score is: -1.322
Final training MSE is: 10.356

Training with solver: adam
Final loss: 0.06068748657274561 4
Total number of iterations: 1696
Current loss: 0.061
Best loss: 0.061
Training time [s]: 4.222
Final training R2 score is: 0.973
Final training MSE is: 0.121





QoT estimation
Task 3

3. ML algorithms training/testing (Linear Regression)
   c) Define function *train_linreg()* that performs training using a training set passed in input and provides training results
      - See details in the skeleton code
   d) Call function *train_linreg()* to fit a linear regressor using the same training set as for the NN
      - **Already given in skeleton code**

Expected output:

```
Training time [s]: 0.006
Final training R2 score is: 0.897
Final training MSE is: 0.459
```

QoT estimation
Task 3

3. ML algorithms training/testing (NN vs Linear Regression)
   e) Compare results (MSE and R2 score) for NN and Linear Regression in both train and test sets
      - **Already given in skeleton code**

Expected output:

```
********** Training scores **********
Linear Regression
Final training R2 score is: 0.897
Final training MSE is: 0.459

Neural Network
Final training R2 score is: 0.973
Final training MSE is: 0.121
```

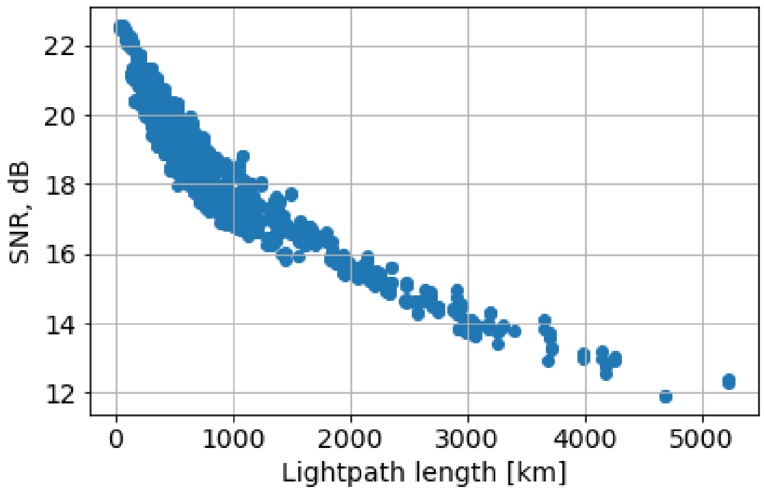
```
********** Test scores **********
Linear Regression
Final test R2 score is: 0.901
Final test MSE is: 0.443

Neural Network
Final test R2 score is: 0.974
Final test MSE is: 0.114
```

QoT estimation
Task 4

4. NN vs LR with one feature only

We had observed a high correlation between SNR and some of the features (e.g., lightpath length)

Do we need all the features?



a) Retrain NN and LR considering a new dataset that only includes **lightpath length** as a feature. Then, compare the performance of the two regressors on the same test set
   - **N.B. Use the SAME train/test sets used in task 3**
   - See details in the skeleton code

QoT estimation

Task 4a)

Expected output:

```
********** Test scores **********
Linear Regression
Final test R2 score is: 0.863
Final test MSE is: 0.61

Neural Network
Final test R2 score is: 0.949
Final test MSE is: 0.229
```

<------------- one feature only (new results)

```
********** Test scores **********
Linear Regression
Final test R2 score is: 0.901
Final test MSE is: 0.443

Neural Network
Final test R2 score is: 0.974
Final test MSE is: 0.114
```

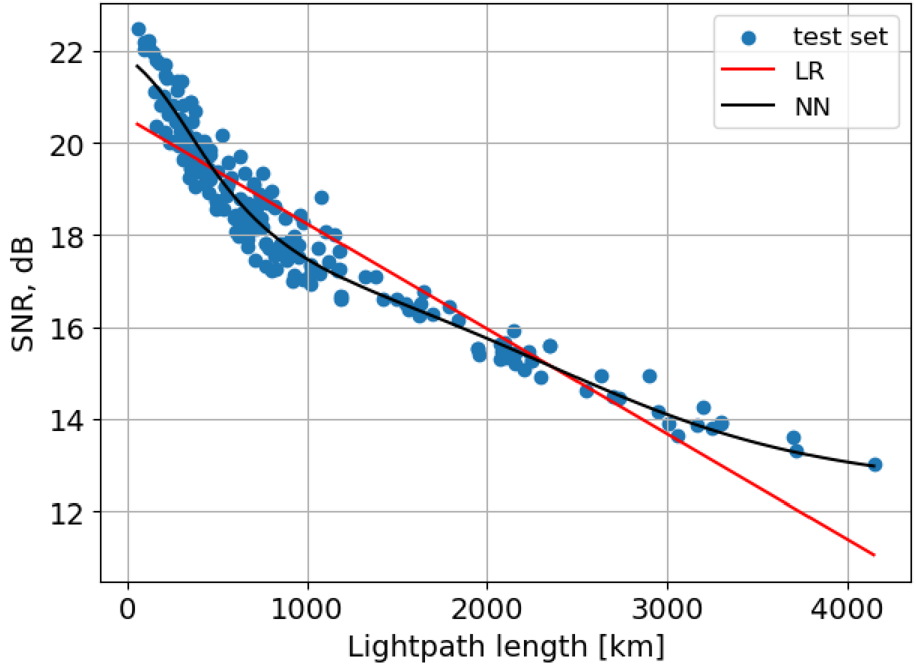
<------------- all features (previous results)

What can we conclude with this analysis?

QoT estimation
Task 4

4. NN vs LR with one feature only
   b) Plot in the same graph the data points of test set and the curves produced by the univariate NN and LR for a range of lightpath length values as in the test set
      - **Already given in skeleton code**

Expected output:



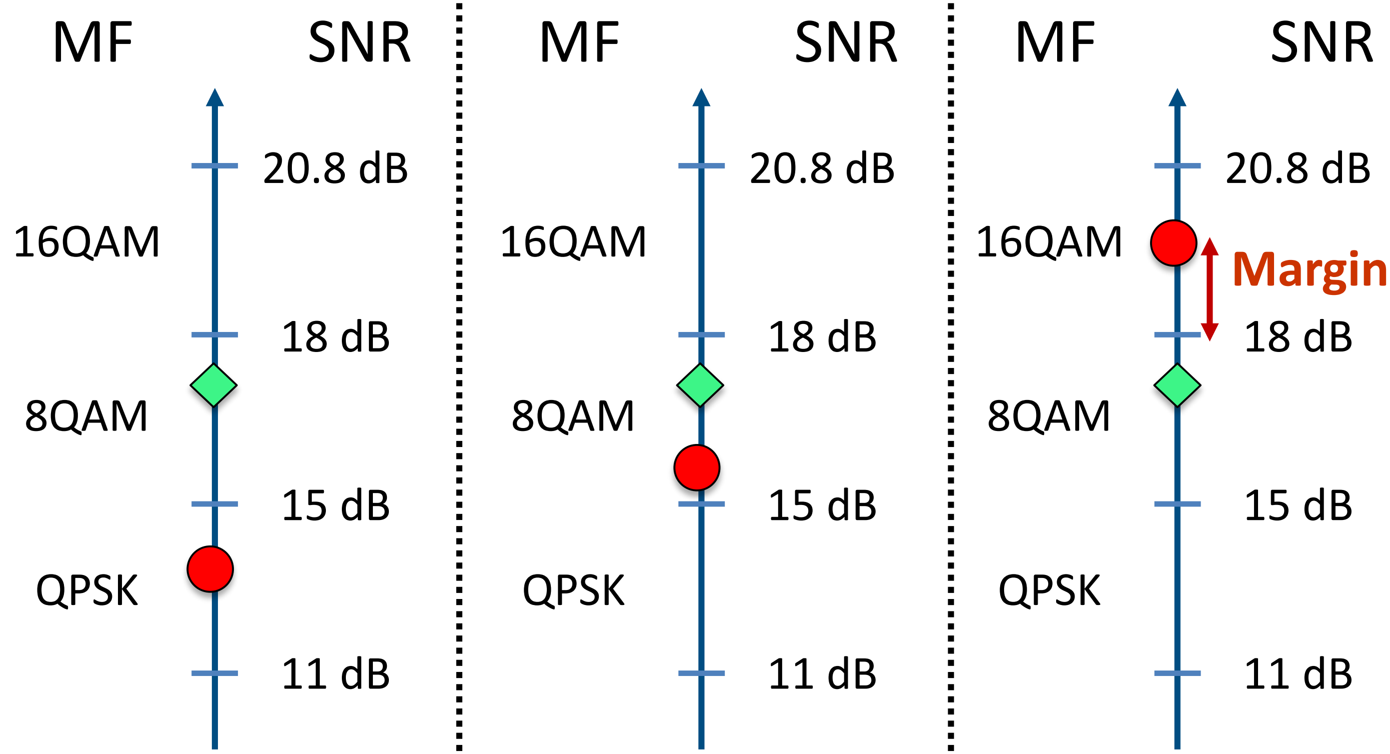
QoT estimation
Optical margin (1/2)

Ground truth SNR

Predicted SNR

MF	Required SNR
QPSK	11 dB +M
8QAM	15 dB +M
16QAM	18 dB +M
32QAM	20.8 dB +M
64QAM	23.7 dB +M

- What is the impact of inaccurate SNR predictions?



$MF_{gt} > MF_p$
QoT Underestimation
Waste of resources

$MF_{gt} = MF_p$
Ideal

$MF_{gt} < MF_p$
QoT Overestimation
Lightpath will be disrupted!!

- To avoid QoT overestimation, we add a safety margin M to required SNR
- Corresponds to subtracting M to the predicted SNR

- We want low M (not conservative) not to waste resources
- Better predictors lead to smaller required margin, closer to real SNR

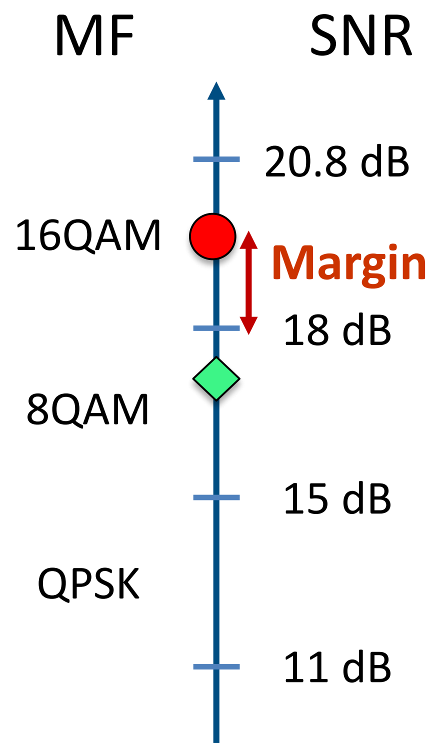
QoT estimation
Optical margin (2/2)

Ground truth SNR

Predicted SNR

MF	Required SNR
QPSK	11 dB +M
8QAM	15 dB +M
16QAM	18 dB +M
32QAM	20.8 dB +M
64QAM	23.7 dB +M

- What is the impact of inaccurate SNR predictions?
- In the lab, we compute minimum margin M *a priori* using predicted and ground truth SNR (test set)
- ***M is the minimum value that guarantees NO DISRUPTION for <u>ALL</u> SNR values in the test set***
- Note that we can still get an underestimated MF (lower than it could be)



$MF_{gt} < MF_p$

QoT Overestimation

Lightpath will be disrupted!!

- To avoid QoT overestimation, we add a safety margin M to required SNR
- Corresponds to subtracting M to the predicted SNR
- We want low M (not conservative) not to waste resources
- Better predictors lead to smaller required margin, closer to real SNR

QoT estimation
Task 5

5. Evaluate the impact of inaccurate predictions
   a) Define function *SNR_to_MF()* that takes in input SNR value and uses it to return the highest possible modulation format (MF)
      - **Already given in skeleton code**
   b) Define function *find_minimal_margin()* that takes in input predicted and ground truth SNR vectors and find the minimum margin to subtract to prediction in order not to have any disruption due to a higher-order MF in prediction wrt ground truth
   c) Test function *find_minimal_margin()* to check if the returned margin allows no disruption for all elements in a predicted SNR vector (compared to its ground truth). Repeat the check comparing ground truth vs non-margined prediction
      - **Already given in skeleton code**

QoT estimation
Task 5a-b-c): expected outputs

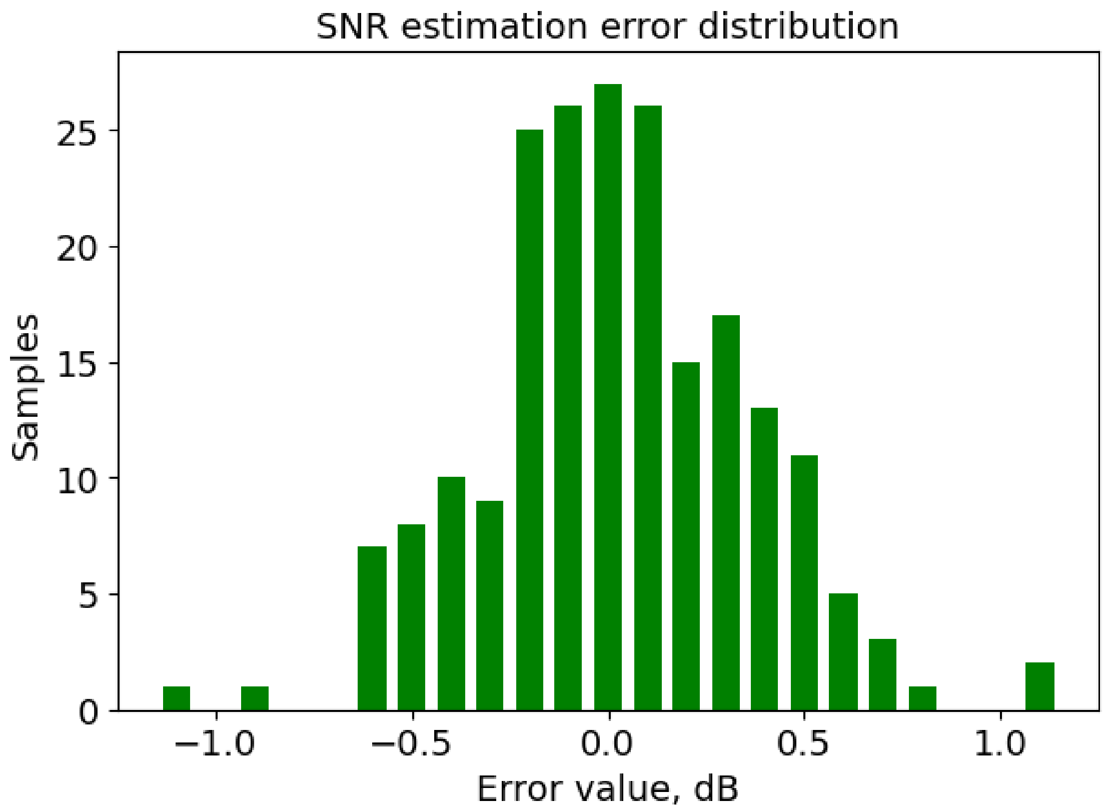
Minimum margin: 1.0999999999999999 dB
***** NON MARGINED PREDICTED SNR *****
Number of incorrectly-assigned MFs: 16
Number of overrated MFs: 9
Number of underrated MFs: 7
******* MARGINED PREDICTED SNR *******
Number of incorrectly-assigned MFs: 86
Number of overrated MFs: 0
Number of underrated MFs: 86

QoT estimation
Task 6

6. Performance evaluation
   a) Define function *perf_eval()* that takes in input real and predicted SNR vectors and provides results
      - See details in the skeleton code
   b) Use the NN model already used in task 3e) to perform prediction for test set and call function *perf_eval()* to evaluate performance
      - **Already given in skeleton code**

QoT estimation
Task 6a)-b): expected outputs

MSE: 0.11 dB
Max error: 1.14 dB
Minimal margin to avoid disruptions 0.4 dB
Error histogram
-1.1 dB: 1 times
-0.9 dB: 1 times
-0.6 dB: 7 times
…
…
0.4 dB: 13 times
0.5 dB: 11 times
0.6 dB: 5 times
0.7 dB: 3 times
0.8 dB: 1 times
1.1 dB: 2 times



QoT estimation
Task 7 – **HOMEWORK (max 1.5 points)**

- What happens with the margin if some features are not known precisely?
- E.g., span lengths are decided at network design phase but the actual values in the field may be slightly different
- Ref paper:

Th3D.5.pdf OFC 2020 © OSA 2020

How Uncertainty on the Fiber Span Lengths Influences QoT Estimation Using Machine Learning in WDM Networks

J. Pesic[(1)], M. Lonardi[(1)], N. Rossi[(2)], T. Zami[(2)], E. Seve[(1)] and Y. Pointurier[(1)]
(1) Nokia Bell Labs France, (2) Nokia France
jelena.pesic@nokia-bell-labs.com

QoT estimation

Task 7 – **HOMEWORK (max 1.5 points)**

7. Evaluate the impact of uncertain fiber span length
   a) Define function *extract_UNCERTAIN_features()* (take inspiration from function *extract_features()* in task 2a), that generates span length features **with a random error chosen in a normal distribution with 0 mean and std dev *sigma* passed in input**
      - See details in the skeleton code

Hints:

To obtain the new dataset with features including the error, you should define new span lengths:

$$new\ span\ length = old\ span\ length + error$$

where

$$error = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2}$$

QoT estimation
Task 7 – **HOMEWORK (max 1.5 points)**

7. Evaluate the impact of uncertain fiber span length
   b) Consider NN algortihm only and redo training and performance evaluation using a new dataset with uncertain features, where error in span length is introduced with std dev = to 5%, 10%, 15% of the maximum span length across all lightpaths. Specifically, after reading the dataset (task 1b), for each error std dev, the steps are:
      - generate features matrix (new function from task 7a)
      - scale, split the dataset and train a new NN (task 3b)
      - predict and evaluate performance (task 6b)

QoT estimation

Task 7a)-b): expected outputs

5%*max_span_length

****
Total number of iterations: 1750
Current loss: 0.064
Best loss: 0.064
Training time [s]: 5.429
Final training R2 score is: 0.971
Final training MSE is: 0.128
****
MSE: 0.12 dB

Max error: 1.28 dB

Minimal margin to avoid disruptions 1.0 dB

Error histogram
-1.1 dB: 1 times
-1.0 dB: 1 times
-0.7 dB: 1 times
…

10%*max_span_length

****
Total number of iterations: 1750
Current loss: 0.064
Best loss: 0.064
Training time [s]: 5.429
Final training R2 score is: 0.971
Final training MSE is: 0.128
****
MSE: 0.12 dB

Max error: 1.28 dB

Minimal margin to avoid disruptions 1.0 dB

Error histogram
-1.1 dB: 1 times
-1.0 dB: 1 times
-0.7 dB: 1 times
…

15%*max_span_length

****
Total number of iterations: 1750
Current loss: 0.069
Best loss: 0.069
Training time [s]: 8.025
Final training R2 score is: 0.969
Final training MSE is: 0.137
****
MSE: 0.13 dB

Max error: 1.34 dB

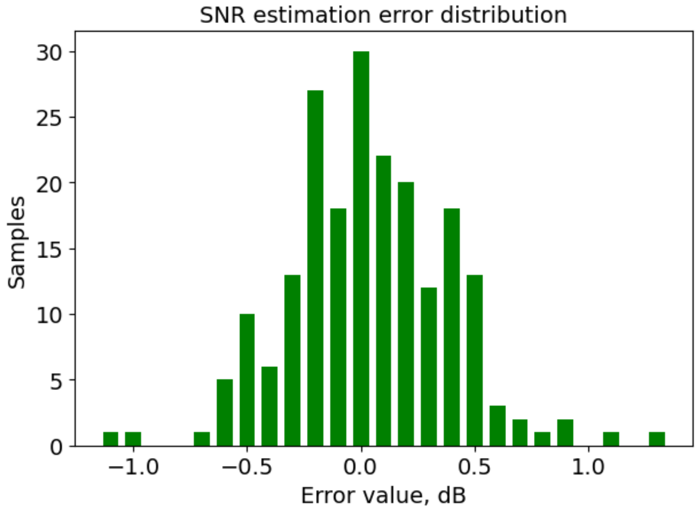
Minimal margin to avoid disruptions 1.0 dB

Error histogram
-1.1 dB: 1 times
-1.0 dB: 1 times
-0.7 dB: 4 times
…

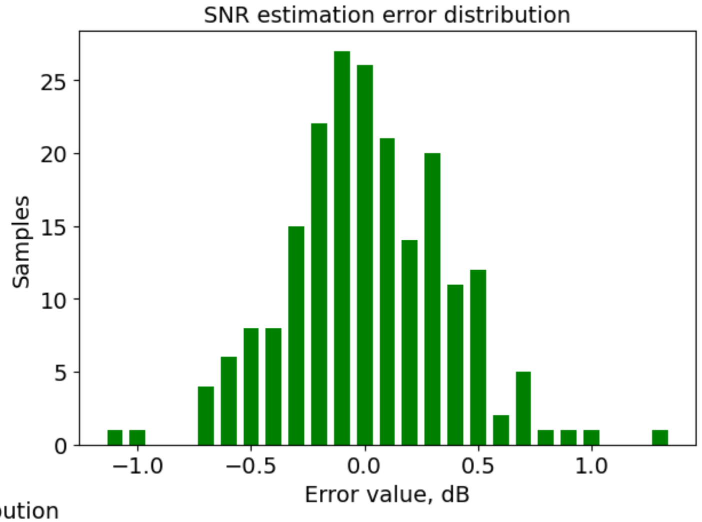
QoT estimation

Task 7a)-b): expected outputs

5%*max_span_length



15%*max_span_length



10%*max_span_length