POLITECNICO MILANO 1863

Explainable AI in Telecommunications

Francesco Musumeci

Oleg Karandin

May 2023

Explainable AI in Telecommunications
Outline

- **Why should we explain ML decisions?**
- How can we explain ML decisions?
  - SHAP framework
- Applications
  - Feature selection
  - Quality of Transmission estimation
  - Fault localization in optical networks
  - Fault identification in microwave links

Shap

```
pip install shap
```

Explainable AI in Telecommunications
Why should we explain ML decisions?

(a) Husky classified as wolf (b) Explanation

Snow-classifier:
"If there is snow, this is a wolf"

Correct prediction for the wrong reason

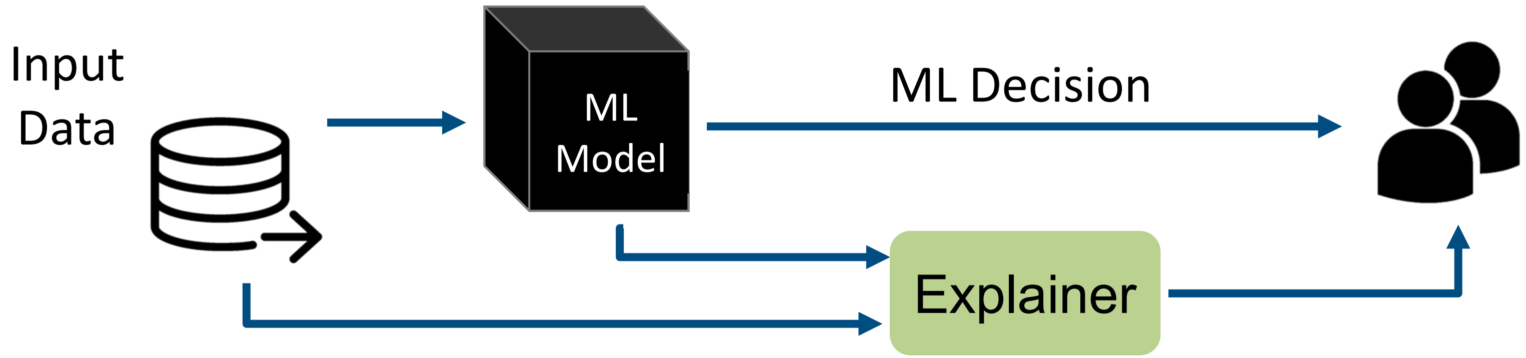
Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "" Why should i trust you?" Explaining the predictions of any classifier." *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016.

Explainable AI in Telecommunications
Why should we explain ML decisions?

- Machine Learning (ML) models are complex black boxes
- Guaranteeing high accuracy is not enough
- Operators want to understand ML reasoning to trust its decisions in mission-critical scenarios

> "The need for increased explainability to enable trust is crucial for the applicability of AI in network OAM"
> [1] **5G PPP Technology Board**

> "For a system to be trustworthy, operators and developers must be able to understand why it behaved in a certain way in a given situation" [2] **Deutsche Telekom**



[1] https://5gppp.eu/wp-content/uploads/2021/05/AI-MLforNetworks-v1-0.pdf
[2] https://www.telekom.com/resource/blob/630952/37534d39c5184b251c235d93edfd6f91/dl-210702-professionsethik-data.pdf

Explainable AI in Telecommunications

Why should we explain ML decisions?

Debugging

Informing feature engineering

Increase trust in the model

Informing human decision-making

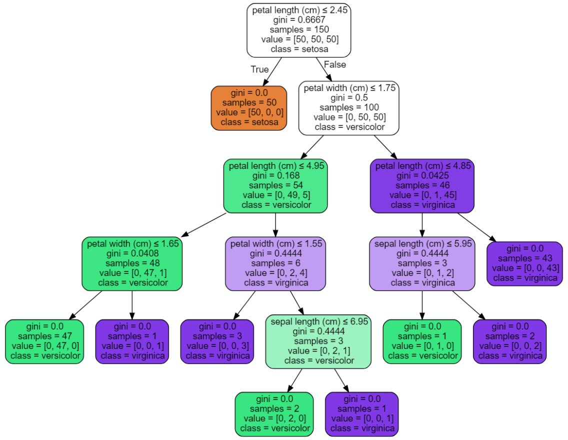
Directing future data collection

Detecting bias

Explainable AI in Telecommunications
Outline

- Why should we explain ML decisions?
- **How can we explain ML decisions?**
  - **SHAP framework**
- Applications
  - Feature selection
  - Quality of Transmission estimation
  - Fault localization in optical networks
  - Fault identification in microwave links

Explainable AI in Telecommunications
Interpretability and Explainability



Interpretable ML: The extent to which an observer can understand the cause of a decision

e.g., linear models, decision trees

- Allow observer to trace individual predictions
- Provide insights to model internals. Useful for model understanding and debugging

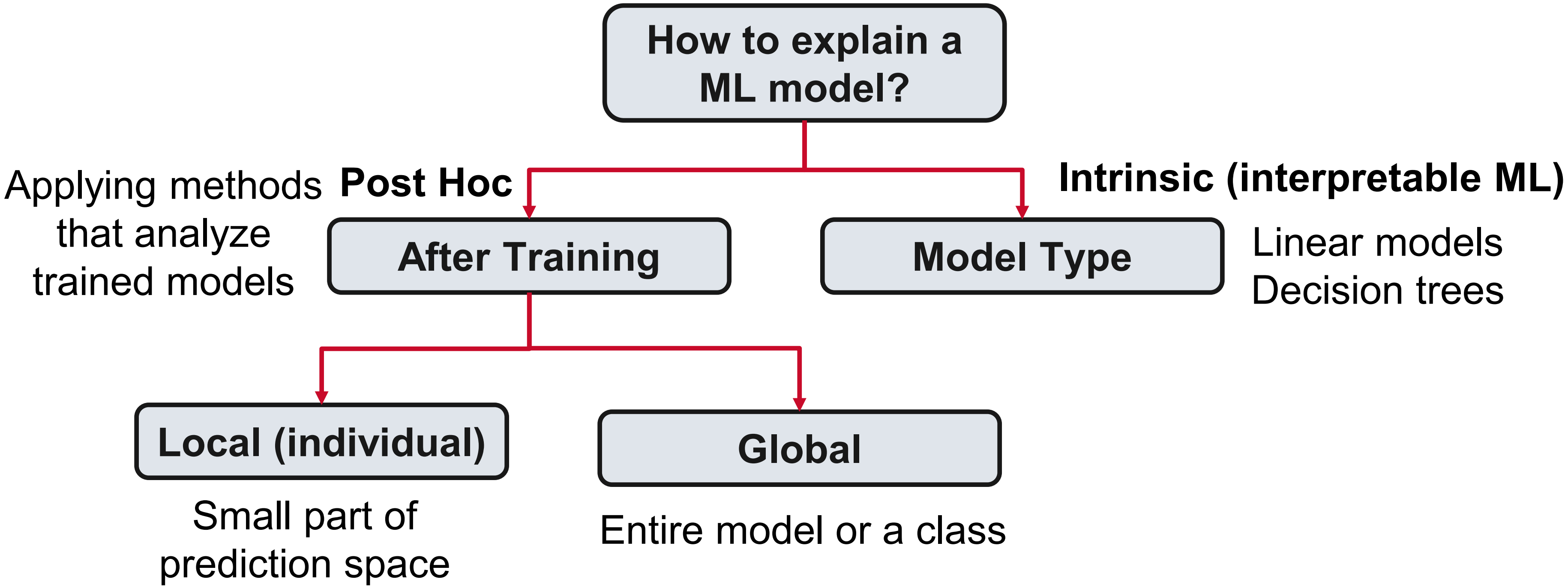
Explainability goes a step further beyond interpretability and looks at how the ML model arrived at the result

- Estimate contribution of each feature to the prediction
- Estimate how feature values affect the prediction

https://www.ibm.com/watson/explainable-ai

Explainable AI in Telecommunications

How can we explain ML decisions?



Explainable AI in Telecommunications
Permutation importance

Test dataset			
F1	F2	…	Label

- Randomly shuffle a single column of the **test** data
- Leave the label and all other columns in order
- How would that affect the accuracy of predictions?

- Model accuracy
  - **Changes little** if the shuffled column is **not important** for model predictions
  - **Decreases significantly** if shuffled column **is important** for model predictions

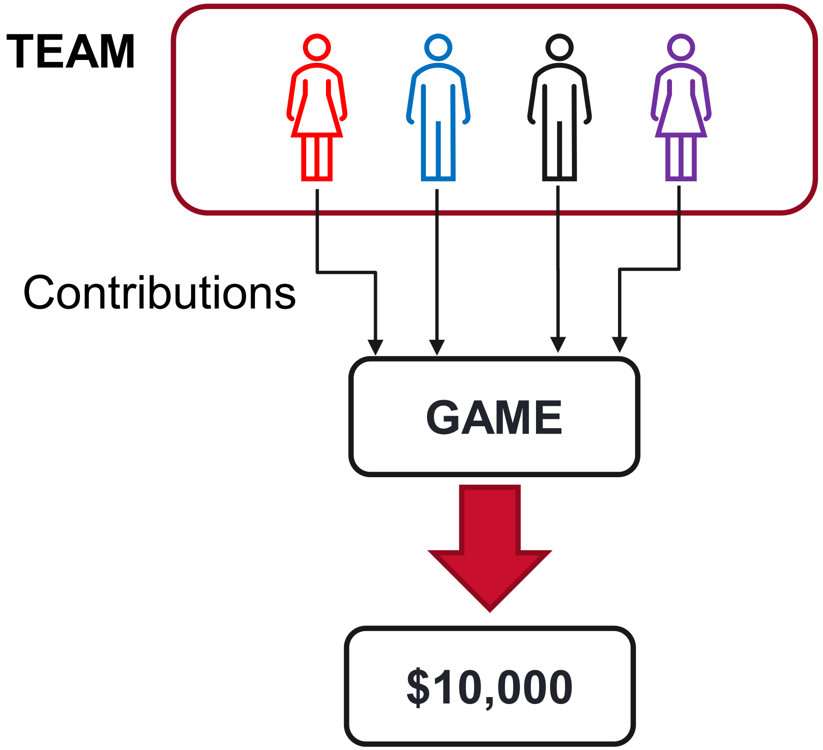
- Repeat for each column – estimate importance of every feature

https://www.kaggle.com/code/dansbecker/permutation-importance

Explainable AI in Telecommunications

SHAP: Shapley-Additive Explanations

- Shapley Additive exPlanations (SHAP) algorithm is model-agnostic
- The idea for SHAP algorithm comes from cooperative game theory (Lloyd Shapley)

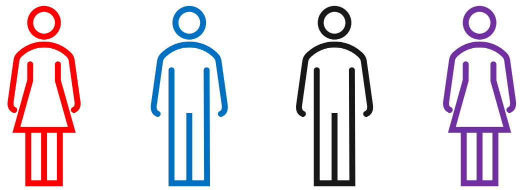


Q.: How can the prize be distributed fairly?
A.: Players' contributions can be fairly estimated by Shapley values

Instead of players in a game, we have features in a ML model

Explainable AI in Telecommunications
SHAP: Shapley-Additive Explanations

TEAM



But it is not this simple, as we need to consider interactions between players

$3,000

$7,000

To find out true contribution, we need to consider all subsets of players (features) and find player contribution in each subset and then average over all contributions

Marginal Contribution

Explainable AI in Telecommunications

SHAP: Shapley-Additive Explanations

ML model

Input data point

Output of ML model with the feature *i*

$$\phi_i(f, x) = \sum_{z' \subseteq x} \frac{|z'|!(M - |z'| - 1)!}{M!} \boxed{[f_x(z') - f_x(z' \setminus i)]}$$

Shapley value for the feature *i* in data point *x*

All possible subsets, (combinations) of features

Output of ML model without the feature *i*

Excluded features are set to their mean values in the dataset

How is the prediction driven by the feature value vs. baseline mean value?

Computing Shapley value for **feature 4** in data point **x**

	Feature 1	Feature 2	Feature 3	Feature 4
x	50	23	34	70
z' = x \ 1	E[Feature 1]	23	34	70
z' \ 4	E[Feature 1]	23	34	E[Feature 4]

https://shap.readthedocs.io/en/latest/example_notebooks/overviews/An%20introduction%20to%20explainable%20AI%20with%20Shapley%20values.html

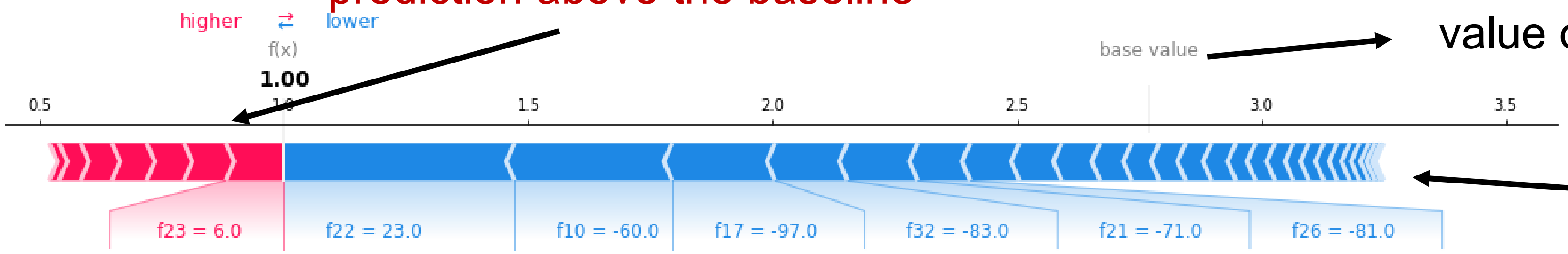
Explainable AI in Telecommunications
SHAP: Shapley-Additive Explanations

- **A Shapley Value for every feature per data point**
  - Positive or negative
  - Absolute value indicates influence (importance)

- Additive property:
  - sum(SHAP values for all features of datapoint x) = prediction_for_datapoint_x – mean_prediction

- **Explaining one instance (local explanation)**
  - SHAP values of all features sum up to explain why the prediction is different from the baseline

Red features are pushing the prediction above the baseline



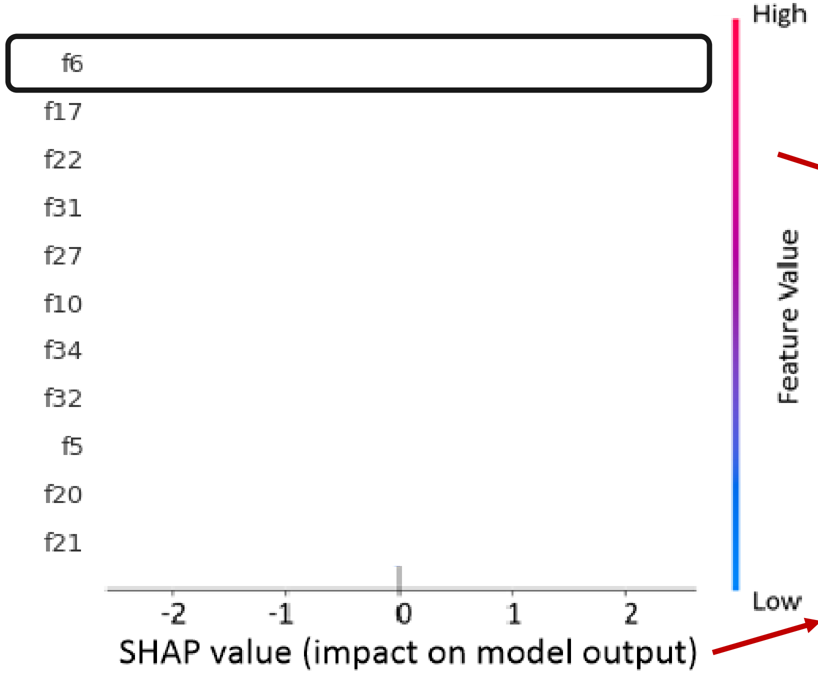
Base value can be seen as the expected value of model's output if we knew nothing about the data point

Blue features are pushing the prediction below the baseline

Explainable AI in Telecommunications
SHAP: Shapley-Additive Explanations

- **Global explanation**
  - find Shapley values for all data points in the dataset
  - plot their values towards a particular class
- We can correlate *feature importance* (Shapley value) with *feature value* for a specific class

Class being explained: Class 1



High value of f6 steers decision towards Class 1.
Low value of f6 steers decision against Class 1

Most impactful feature at the top

Color shows if feature has high or low value

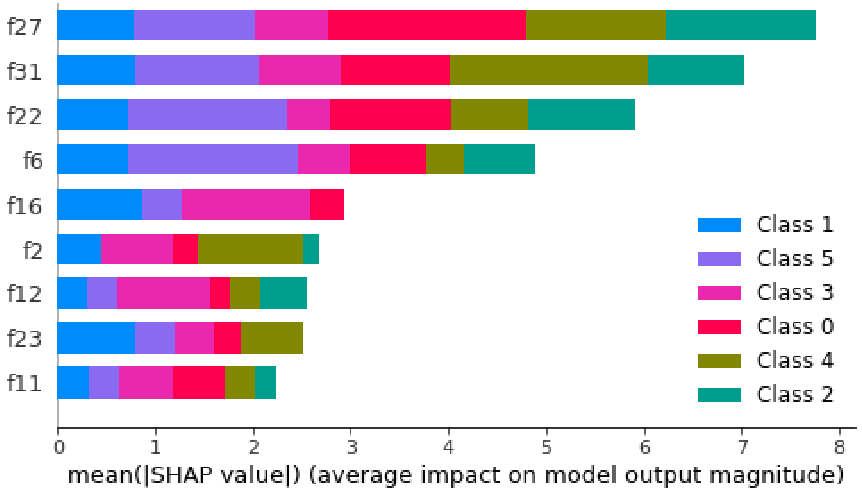
Feature importance according to SHAP

Explainable AI in Telecommunications
Outline

- Why should we explain ML decisions?
- How can we explain ML decisions?
  - SHAP framework
- **Applications**
  - **Feature selection**
  - Quality of Transmission estimation
  - Fault localization in optical networks
  - Fault identification in microwave links

Explainable AI in Telecommunications
SHAP: Shapley-Additive Explanations

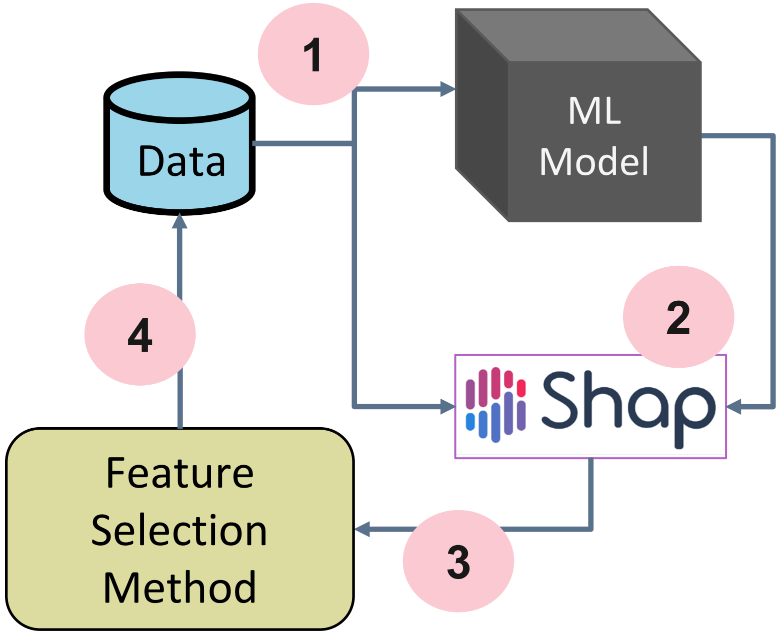
Global per-class feature importance



Some features might be globally less important than other but may be very important for one particular class

Feature Selection with SHAP

1. Start with an initial set of features, train a model and find SHAP values to rank features
2. Then, eliminate less important features to produce a new feature set
3. Train a new model with a new set of features
4. Repeat, based on model accuracy



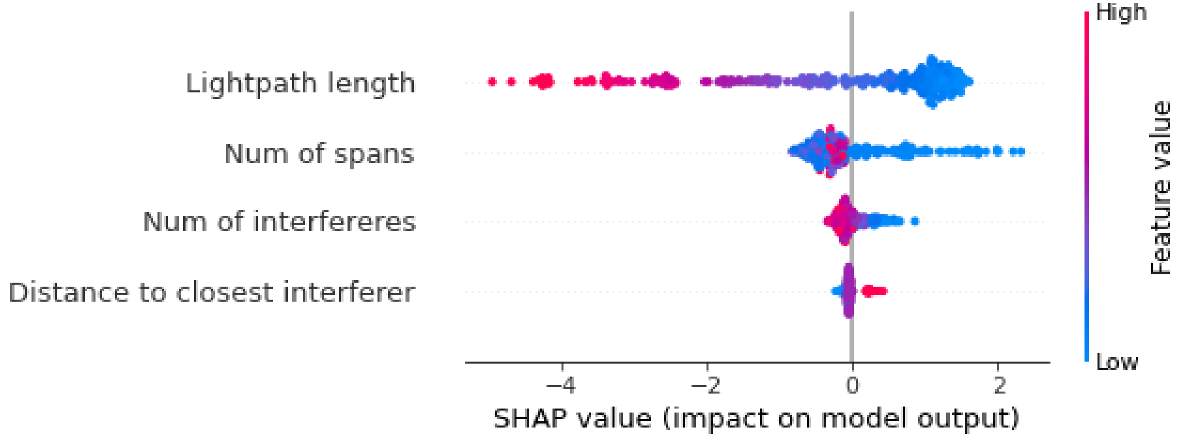
https://github.com/cerlymarco/shap-hypetune

Explainable AI in Telecommunications
Outline

- Why should we explain ML decisions?
- How can we explain ML decisions?
  - SHAP framework
- **Applications**
  - Feature selection
  - **Quality of Transmission estimation**
  - Fault localization in optical networks
  - Fault identification in microwave links

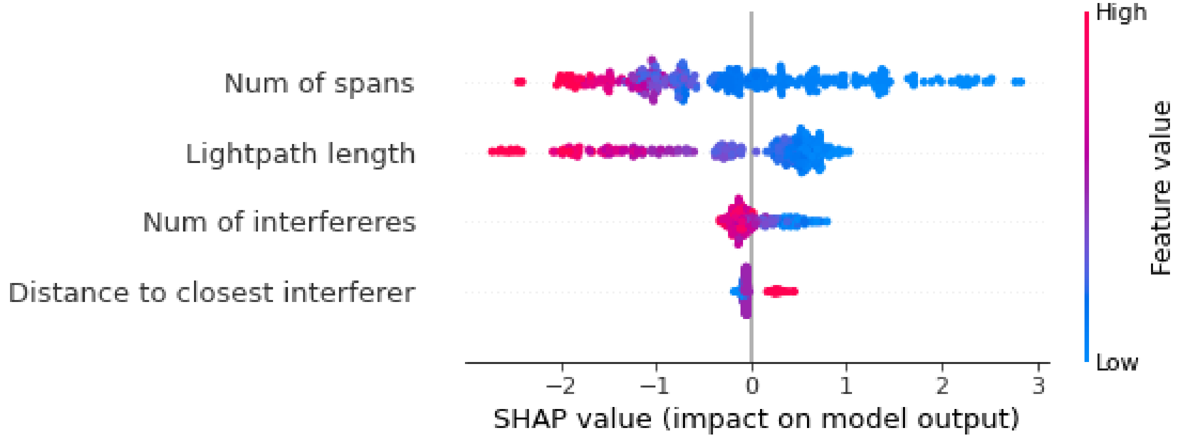
Explainable AI in Telecommunications
Quality of Transmission estimation

- How does ML model reasons when estimating SNR at the optical receiver?



Low lightpath length – higher SNR

- What happens with model reasoning if we add noise to lightpath length (N(0, std = 10% of length)



Model relies more on number of spans

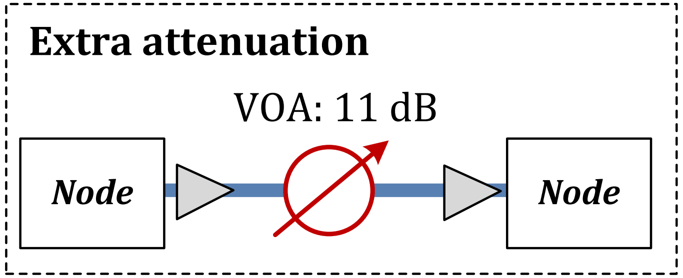
Explainable AI in Telecommunications
Outline

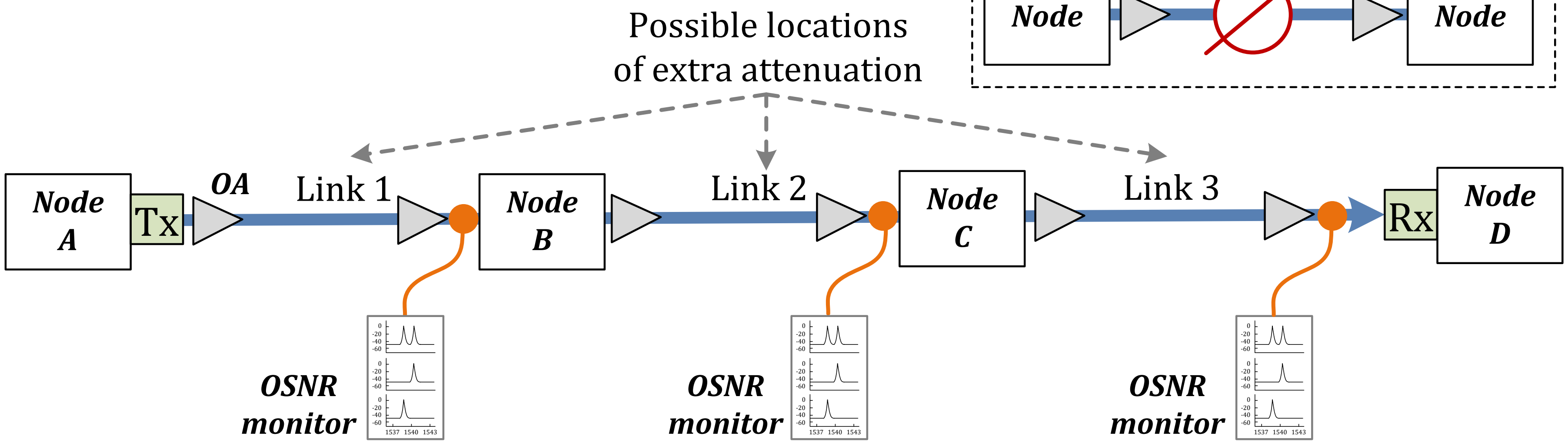
- Why should we explain ML decisions?
- How can we explain ML decisions?
  - SHAP framework
- **Applications**
  - Feature selection
  - Quality of Transmission estimation
  - **Fault localization in optical networks**
  - Fault identification in microwave links

Explainable AI in Telecommunications

Fault localization in optical networks

- How does ML model reason when localizing faults?



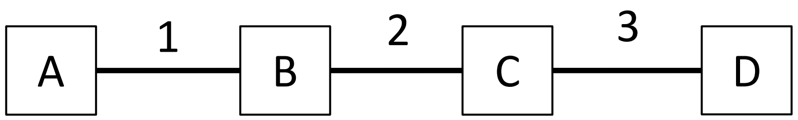


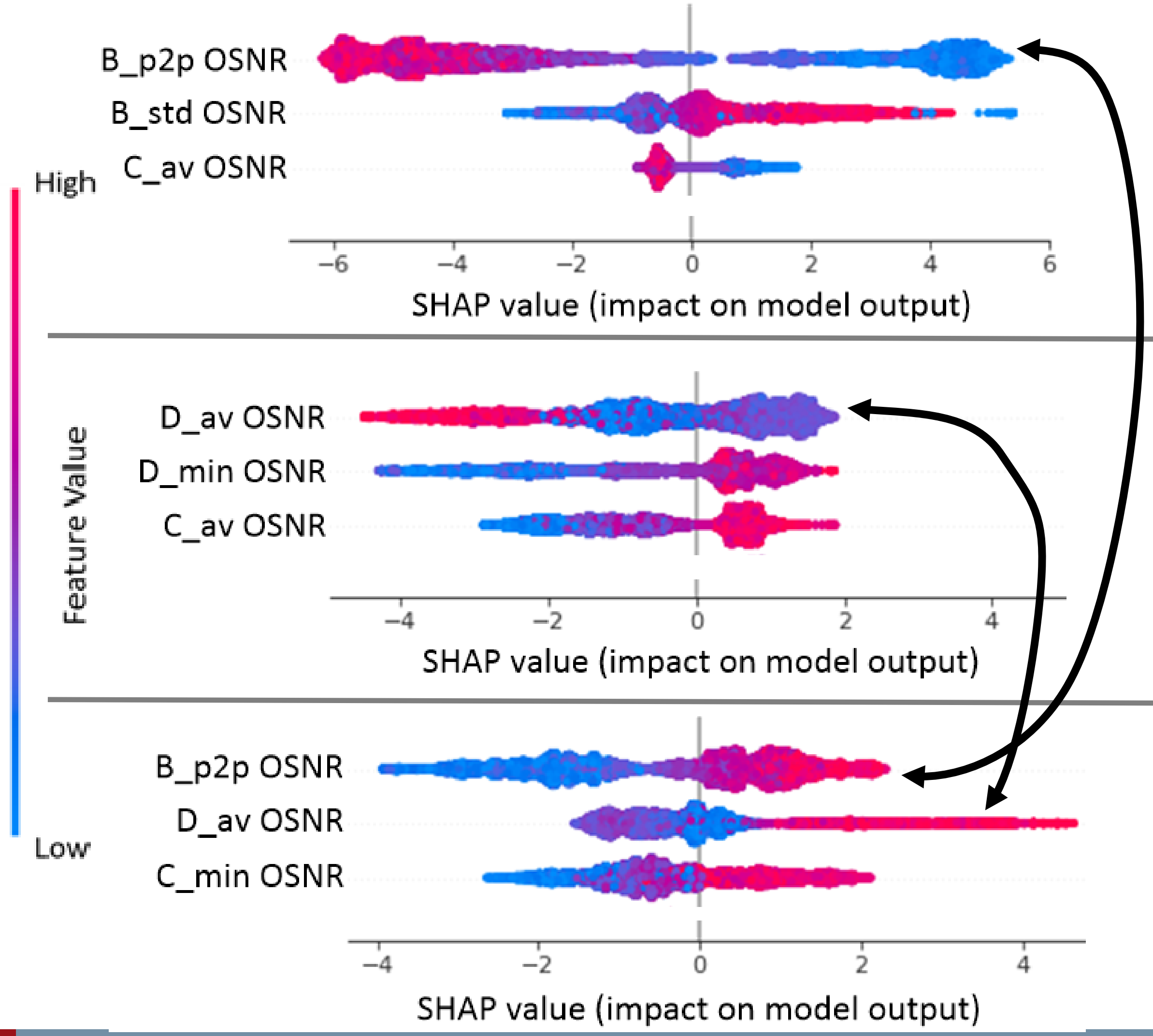
- We use 5 features (statistics of an OSNR window) from each of the 3 monitors

X			y
Features of OSNR windows measured at link 1	Features of OSNR windows measured at link 2	Features of OSNR windows measured at link 3	Fault location

https://ieeexplore.ieee.org/document/9782859

Explainable AI in Telecommunications
Fault localization in optical networks





Fault at link 1:
ML-model reasons using OSNR statistics at the monitor after the fault

Fault at link 3:
ML-model reasons using mostly OSNR statistics at the monitor after the fault

Fault at link 2:
ML-model reasons by contradiction: if the fault is not at Link 1 or Link 3 – it is at Link 2

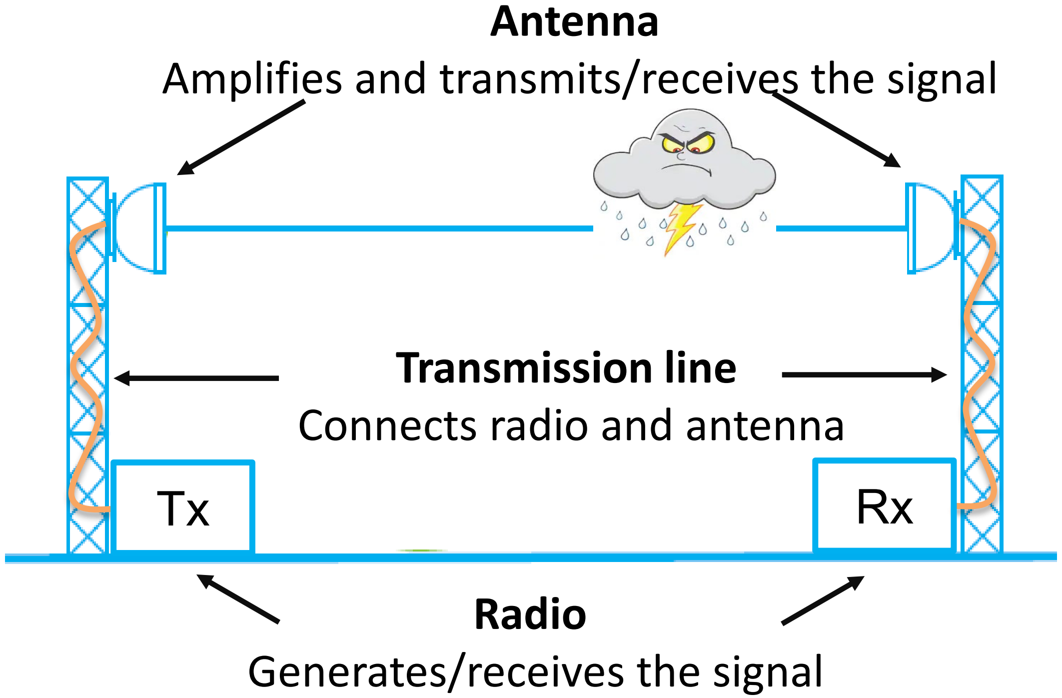
Explainable AI in Telecommunications
Outline

- Why should we explain ML decisions?
- How can we explain ML decisions?
  – SHAP framework
- **Applications**
  – Feature selection
  – Quality of Transmission estimation
  – Fault localization in optical networks
  – **Fault identification in microwave links**

Explainable AI in Telecommunications

Hands-On: Fault identification in microwave links

- Goal: automate the identification of failures in a microwave network
- We can use XAI to make sure that the model takes correct decisions based on correct reasons



https://www.sciencedirect.com/science/article/pii/S138912862200500X

Explainable AI in Telecommunications
Hands-On: Task 1a) and 1b)

- Task 1a): load dataset
  - One datapoint – a 45-minutes window (3 consecutive slots of 15 minutes), with unavailability seconds in the last slot

	acmEngine	esN-2	sesN-2	txMaxAN-2	txminAN-2	rxmaxAN-2	rxminAN-2	txMaxBN-2	txminBN-2	rxmaxBN-2	...	rxmaxAN	rxminAN	txMaxBN	txminBN	rxr
0	1	0.0	0.0	18.0	18.0	-49.0	-49.0	18.0	18.0	-50.0	...	-52.0	-97.0	22.0	18.0	
1	1	0.0	0.0	18.0	18.0	-49.0	-53.0	18.0	18.0	-50.0	...	-50.0	-96.0	22.0	18.0	
2	1	0.0	0.0	18.0	18.0	-45.0	-47.0	18.0	18.0	-47.0	...	-52.0	-91.0	22.0	18.0	
3	1	0.0	0.0	18.0	18.0	-45.0	-52.0	18.0	18.0	-47.0	...	-80.0	-95.0	22.0	22.0	
4	1	0.0	0.0	18.0	18.0	-45.0	-46.0	18.0	18.0	-47.0	...	-45.0	-93.0	22.0	18.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
2508	1	0.0	0.0	18.0	18.0	-58.0	-60.0	18.0	18.0	-52.0	...	-60.0	-97.0	22.0	18.0	
2509	0	0.0	0.0	18.0	18.0	-40.0	-41.0	18.0	18.0	-39.0	...	-48.0	-87.0	18.0	18.0	
2510	0	0.0	0.0	18.0	18.0	-39.0	-40.0	18.0	18.0	-38.0	...	-48.0	-86.0	18.0	18.0	
2511	1	0.0	0.0	17.0	17.0	-55.0	-68.0	17.0	17.0	-55.0	...	-47.0	-71.0	17.0	17.0	
2512	1	0.0	0.0	14.0	14.0	-39.0	-50.0	18.0	18.0	-38.0	...	-41.0	-93.0	18.0	14.0	

2513 rows × 35 columns

- Task 1b): train Random Forest model and check performance

```
Accuracy:  0.9443339960238568
Precision:  [0.86 0.94 0.82 0.84 0.97 0.98]
Recall:  [0.86 0.93 0.9  0.82 0.97 0.99]
F1-score:  [0.86 0.93 0.9  0.82 0.97 0.99]
```

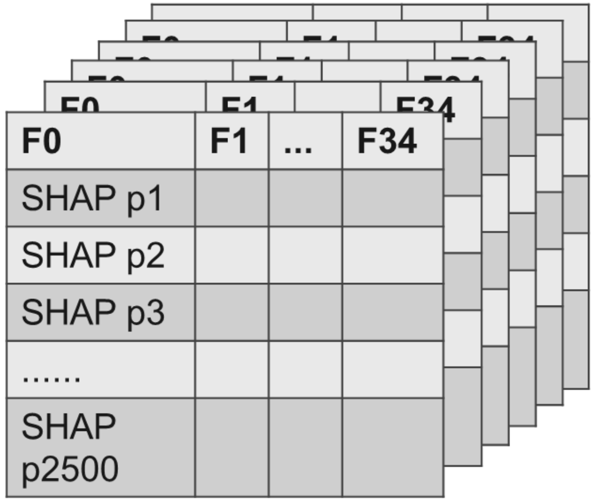
Explainable AI in Telecommunications

Hands-On: Task 2a)

```python
#create and initialize explainer, we use TreeExplainer and pass the rf classifier as an argument
explainer_shap = shap.TreeExplainer(rf)

#Calculating the SHAP values by using shap_values method. It takes training data as an argument
shap_values = explainer_shap.shap_values(data[:shappoints])
```

- Initialize the SHAP explainer and then calculate SHAP values using TreeExplainer
- SHAP values will be represented in a 3-D Matrix of size (n. classes x n. features x n. data points)
  – In our case: 6 x 35 x 2500

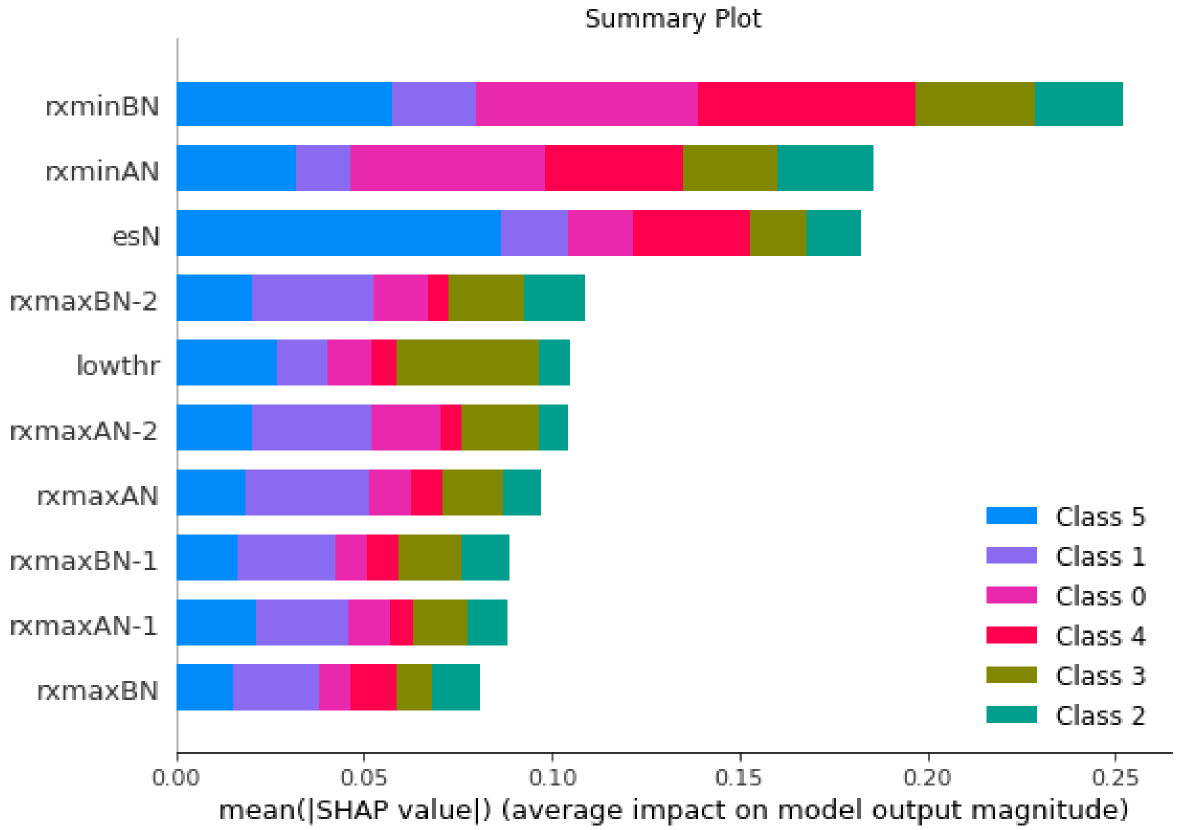



```
[array([[ 0.03,  0.01,  0.01, ...,  0.02, -0.  ,  0.  ],
        [ 0.04,  0.01,  0.  , ...,  0.02, -0.01,  0.  ],
        [ 0.04,  0.01,  0.01, ...,  0.02, -0.  ,  0.  ],
        ...,
        [-0.  ,  0.  ,  0.  , ...,  0.01, -0.01, -0.  ],
        [-0.  ,  0.  ,  0.  , ...,  0.01, -0.01, -0.  ],
        [-0.  ,  0.  ,  0.  , ...,  0.01, -0.01, -0.  ]]),
 array([[ 0.01, -0.  , -0.  , ..., -0.  ,  0.01,  0.  ],
        [ 0.01, -0.  , -0.  , ..., -0.  ,  0.01,  0.  ],
        [ 0.  , -0.  , -0.  , ..., -0.  , -0.  , -0.  ],
        ...,
        [-0.01, -0.  , -0.  , ..., -0.  , -0.01, -0.  ],
        [-0.01, -0.  , -0.  , ..., -0.  , -0.01, -0.  ],
        [-0.01, -0.  , -0.  , ..., -0.  , -0.01, -0.  ]]),
```

Explainable AI in Telecommunications
Hands-On: Task 2b)

```
shap.summary_plot(shap_values, feature_names=list(data.columns), show = False, max_display=10)
plt.title("Summary Plot")
```

- Plot SHAP Global Summary Plot – Global feature importance

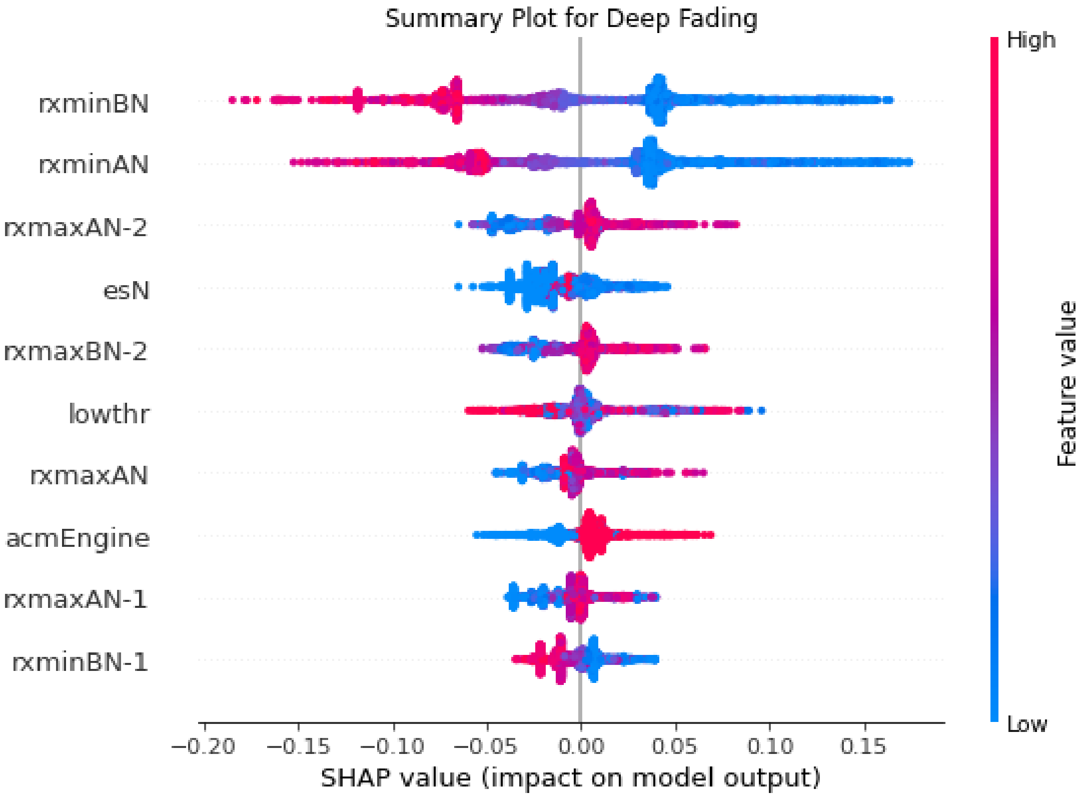


Explainable AI in Telecommunications

Hands-On: Task 2c)

```
for class_ind in range(n_label):
    shap.summary_plot(shap_values[class_ind], features=data[:shappoints], feature_names=list(data.columns),
                      show = False, max_display=10)
```

- Plot SHAP Summary Plot for Deep Fading class



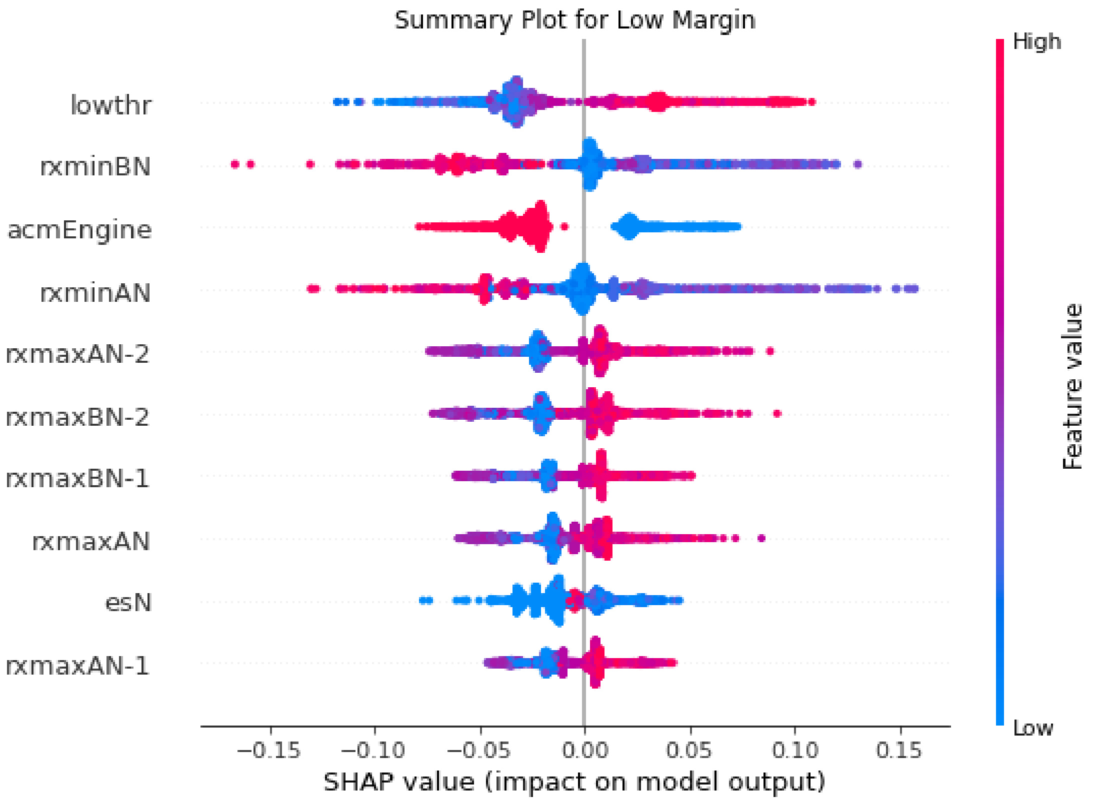
rxminBN and rxminAN:
Low values steer decision towards Deep Fading.
High values steer decision towards “Not Deep Fading”

This behavior can be verified by the domain expert

Explainable AI in Telecommunications
Hands-On: Task 2c)

```
for class_ind in range(n_label):
    shap.summary_plot(shap_values[class_ind], features=data[:shappoints], feature_names=list(data.columns),
                      show = False, max_display=10)
```

- Plot SHAP Summary Plot for Low Margin class



lowthr:
High value steers decision towards Low Margin.
Low/medium value steers decision towards "Not Low Margin"

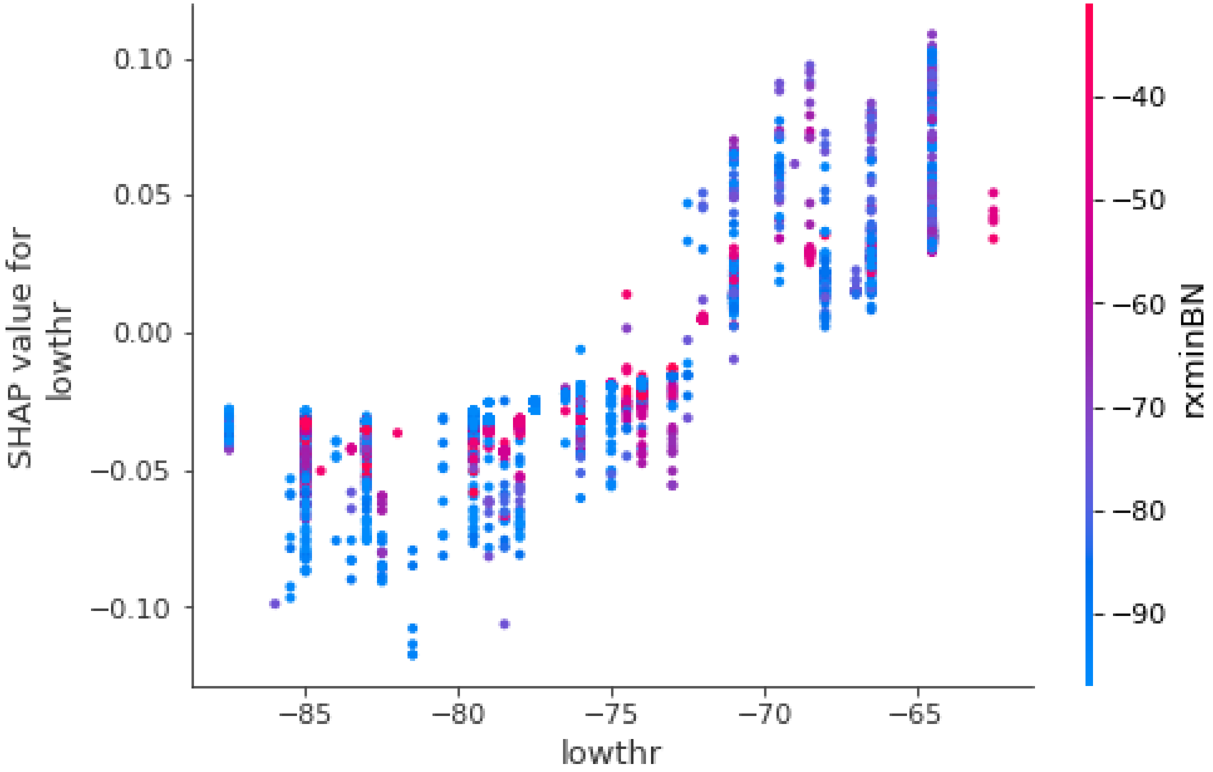
This behavior can be verified by the domain expert

Explainable AI in Telecommunications

Hands-On: Task 2d)

```
# plot SHAP values for lowthr feature in Low margin class (index 3) and use color to code value of rxminBN feature
shap.dependence_plot('lowthr', shap_values[3], features=data[:shappoints], feature_names=list(data.columns),
                     interaction_index = 'rxminBN')
```

- Dependence plot:
  - X-axis: values of feature A (lowthr)
  - Y-axis: SHAP values of feature A
  - Color: another feature B (rxminBN)



lowthr > -70 steers the decision towards Low Margin, also for high/medium values of rxminBN

Explainable AI in Telecommunications
Future directions

Privacy/explainability interplay in collaborative learning scenarios

Explainability in Reinforcement Learning

Explainability in models trained on raw data (e.g., OSNR signal or EDFA power profile)