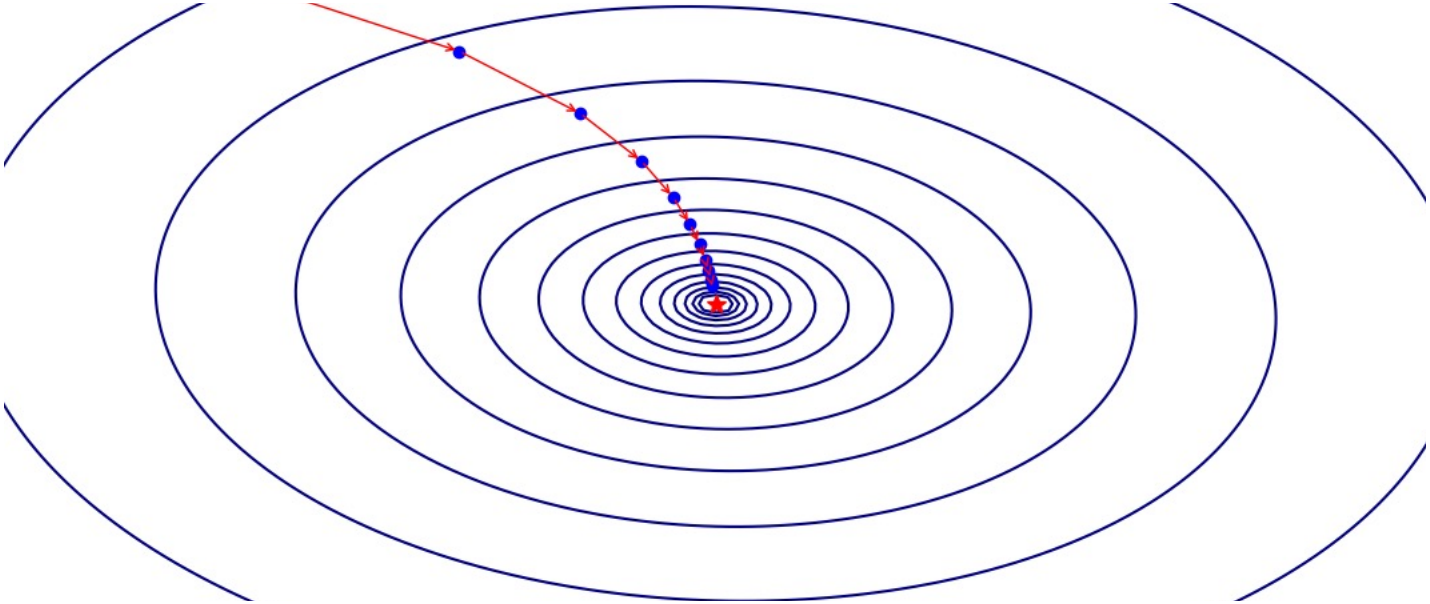


Machine Learning in Signal Processing

Winter Semester 2023/24

5. Performance Evaluation

14.11.2023

Prof. Dr. Vasileios Belagiannis

Chair of Multimedia Communications and Signal Processing

Friedrich-Alexander-Universität
Technische Fakultät

Course Topics

1. Introduction.
2. Basics and terminology.
3. Linear regression.
4. Linear classification.
5. **Performance evaluation.**
6. Neural networks.
7. Deep neural networks.
8. Decision trees.
9. Ensemble models.
10. Random forests.
11. Clustering / Unsupervised learning.
12. Dimensionality reduction.
13. Support vector machines.
14. Recap and Q&A.

- The exam will be written.
- We will have an exam preparation test before the end of the year.

Acknowledgements

Ideas and inspiration from:

- CSC311 Introduction to Machine Learning, University of Toronto.
- Introduction to Machine Learning: LMU Munich.
- Introduction to Machine Learning, CSAIL, MIT.
- CSE 574 Introduction to Machine Learning, University of Buffalo.
- Special thanks Arij Bouazizi, Julia Hornauer, Julian Wiederer, Adrian Holzbock and Youssef Dawoud for contributing to the lecture preparation.

Last Lecture Recap

- Linear classifiers.
- Grouping the classifiers.
- Decision Regions and Boundaries.
- Linear Classifier examples.

Today's Agenda and Objectives

- Generalisation.
- Bias, Variance and Model Complexity.
- Under-fitting and over-fitting.
- Measures of regression.
- Measures of classification.

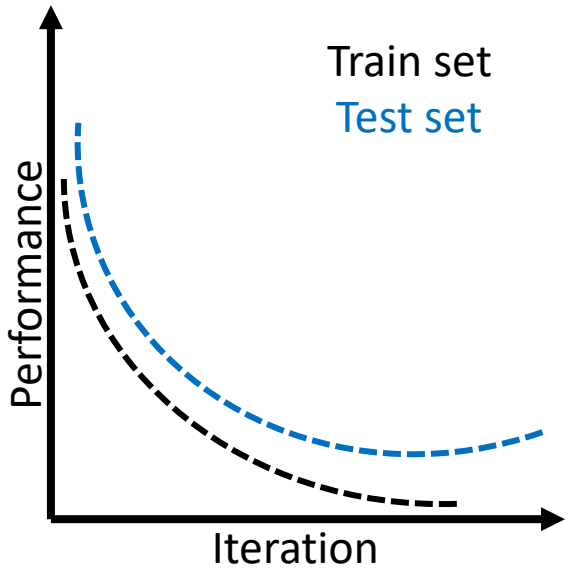
Machine Learning

- Recall the definition by Thomas Mitchell*:
  - "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."
- The performance evaluation refers to the performance measure of the machine learning algorithm.
- For a specific task, our goal is to select the machine learning algorithm or the hyper-parameters of a specific algorithm based on the performance evaluation.
- Overall we seek for machine learning algorithms that <u>generalise well</u>.

*Thomas Mitchell. Machine learning. McGraw-Hill Education, 1997.

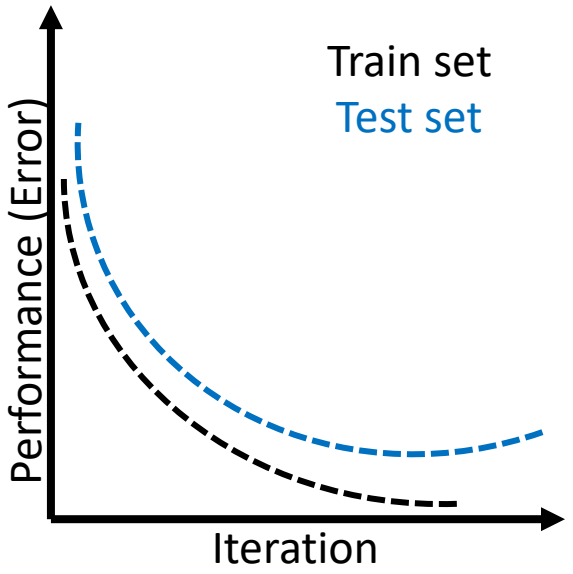
Generalization

- It is the ability to perform well on unobserved samples.
  - Unobserved samples → test set.
  - Observed samples → training (and validation) set.
- The performance is reflected in the training error, as well as test error (known as generalization error).
- Reducing the generalization error is a main challenge in machine learning.



Generalization (Cont.)

- Assumption: training and test sets come from the same distribution.
- The training set is used to learn the parameters of the machine learning algorithm. The algorithm is then evaluated on the test set, separately obtained, to compute the test error (generalization error).
- The overall goal is to minimize the training error, as well as **close the gap** between training and test errors.
- In the **data generation process**, we always make the i.i.d assumption, namely 1. the data samples are independent from each other and 2. the training and test set are identically distributed.



Generalization: Bias and Variance

- Consider the target variable Y, the input features X, and the prediction model $\hat{f}(x)$ trained on the training set T.
- The loss function for measuring errors between Y and $\hat{f}(x)$ is denoted by $L\left(Y, \hat{f}(x)\right)$. For example:
  - $L\left(Y, f(x)\right) = \begin{cases} (Y - \hat{f}(x))^2 & Mean\ Square\ Error \\ \left|Y - \hat{f}(x)\right| & Absolute\ Error \end{cases}$
- The <u>generalization error</u> is measured as the prediction error over an independent test set τ as:
  - $Err = E\left(L\left(Y, \hat{f}(x)\right)\middle|\tau\right)$.

Generalization: Bias and Variance (Cont.)

- Now we $Y = f(x) + \varepsilon$ with ε error.
  - $E[\varepsilon] = 0$.
  - $Var(\varepsilon) = \sigma_\varepsilon^2$.
- Given the input sample x_0 we computer the error.
- $Err(x) = E\left[(Y - \hat{f}(x))^2\right] =$

$$= \sigma_\varepsilon^2 + \left[E[\hat{f}(x)] - f(x)\right]^2 + E\left[\hat{f}(x) - E[\hat{f}(x)]\right]^2$$

$$= \sigma_\varepsilon^2 + Bias^2(\hat{f}(x)) + \mathrm{Var}(\hat{f}(x))$$

$$= Irreducible\ Error + Bias^2 + Variance$$

- $f(x)$ is the true mean.
- $E[\hat{f}(x)]$ expected value of the estimate.

The elements of statistical learning: data mining, inference and prediction, (Section 2.9).

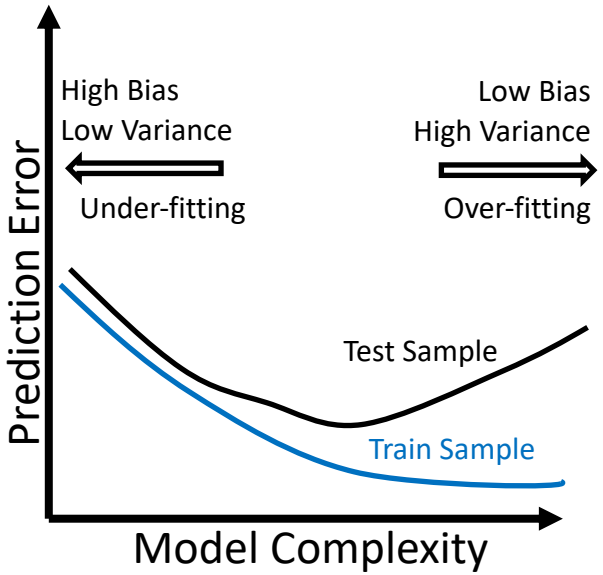
Bias, Variance and Model Complexity

- $Err(x_0) = Irreducible\ Error + Bias^2 + Variance$
  - The irreducible error can not be minimized by training better models. It measures the amount of noise in the data, which is inherent in any real-world dataset.
  - The squared bias refers to the amount by which the average of our estimate differs from the true mean $f(x_0)$. It shows whether the model predictions approximate the real model well. Models with high capacity have low bias and models with low capacity have a high bias.
  - The last term is the variance, namely the expected squared deviation of $\hat{f}(x_0)$ around its mean. It shows the gap between the real model and predictor. Models with high capacity have a high variance (neural networks), where models with low capacity (logistic regression) have a low variance.

Online reference: https://towardsdatascience.com/understanding-the-bias-variance-tradeoff-165e6942b229

Bias, Variance and Model Complexity (Cont.)

- A simple model with few number of parameters for a complex task tend to have high bias and low variance.
- A complex model with many parameters for a simple task tend to have high variance and low bias.
- The goal is to find a balance between bias and variance, such that it minimizes the total error.



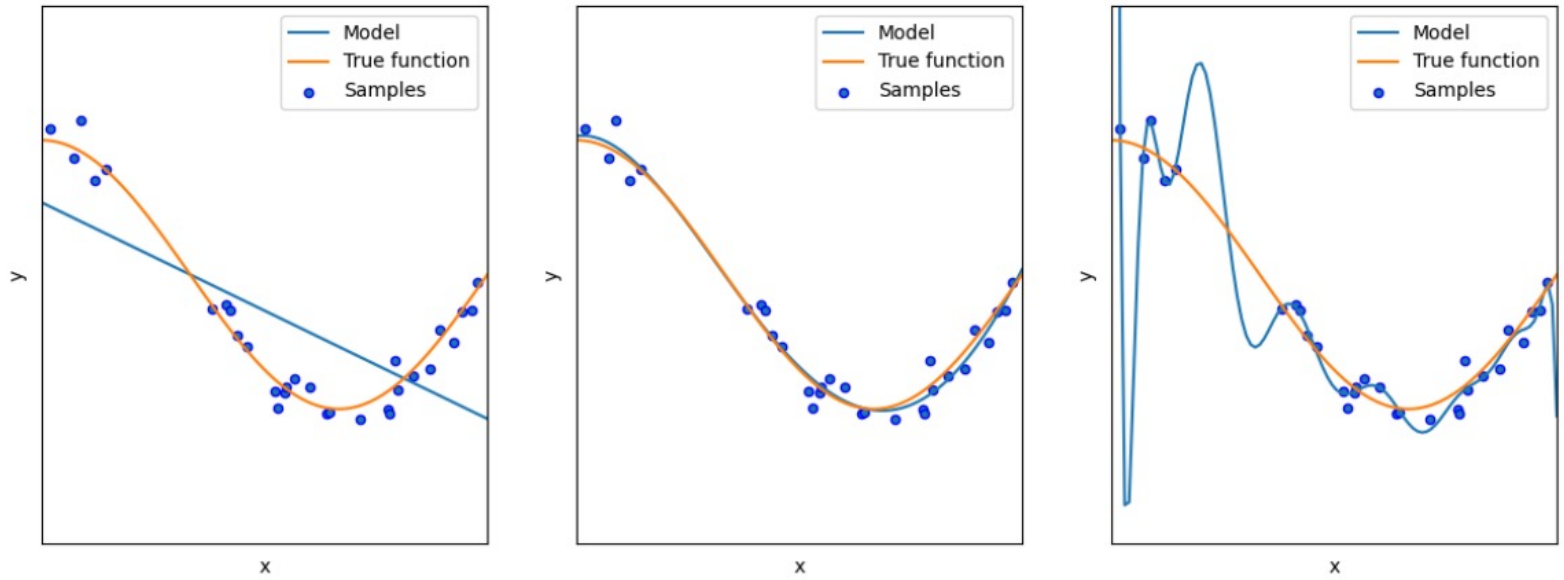
The elements of statistical learning: data mining, inference and prediction, (Section 2.9).

Overfitting and Underfitting

- The under-fitting occurs if the model does not reflect the true shape of the underlying function.
  - It can be recognized with a high train error and high test error.
  - The reason can be too few model parameters.
- The over-fitting occurs when the model reflects noise in the training data, which do not generalize.
  - It can be recognized with a low train error and high test error.
  - It can be due to too many model parameters.
- Model capacity: It determines how complex mappings the model learns. It can help to avoid under-fitting and prevent over-fitting.

Overfitting and Underfitting (Cont.)

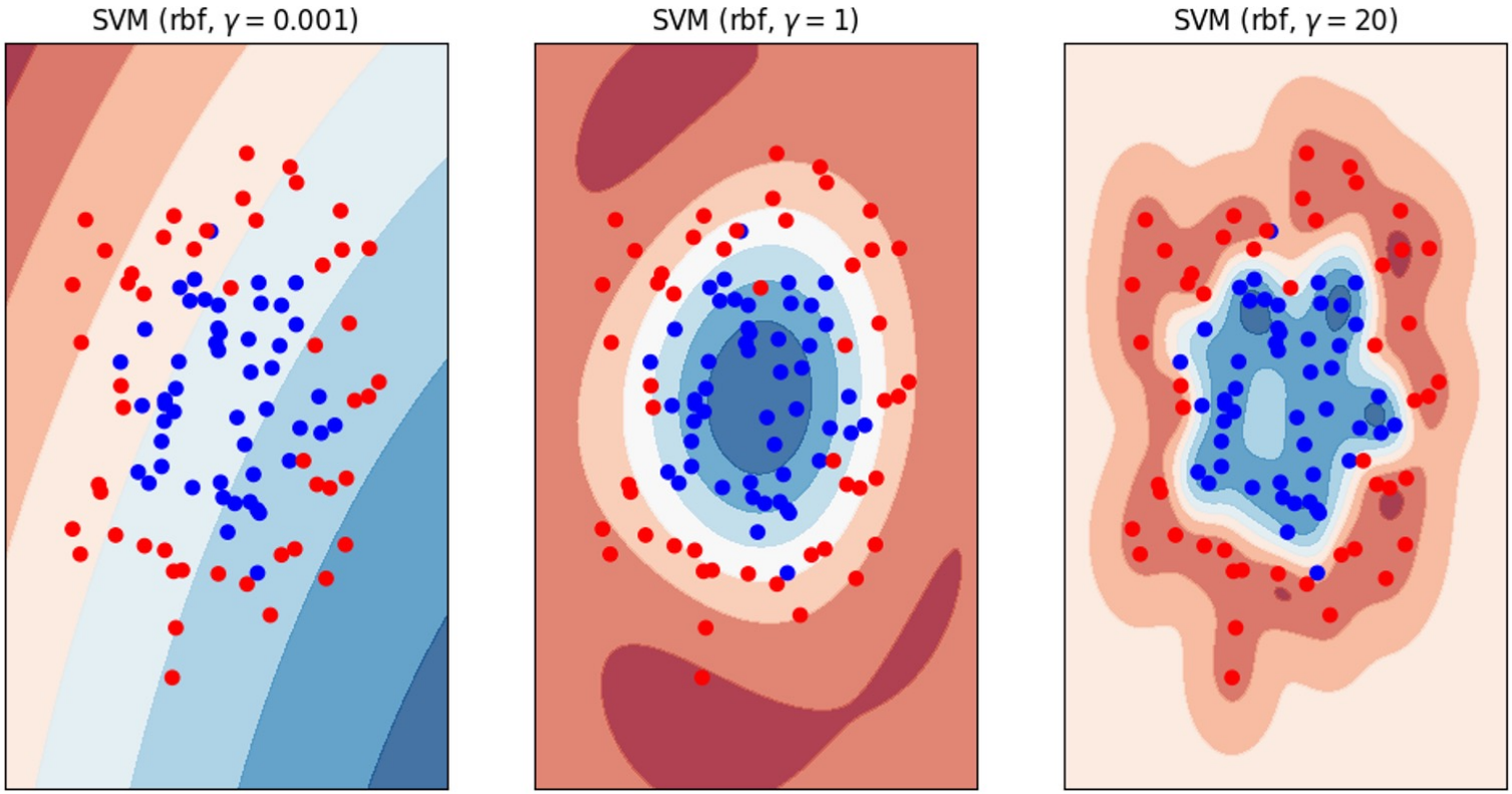
- Under-fitting, Normal fitting, Over-fitting.



- To prevent overfitting (and select hyper-parameters), A validation set is constructed from the training set.
  - Split the training set into two disjoint sets, e.g. 80% for training and 20% for validation. The validation error is usually closer to the test error, compared to the training error.

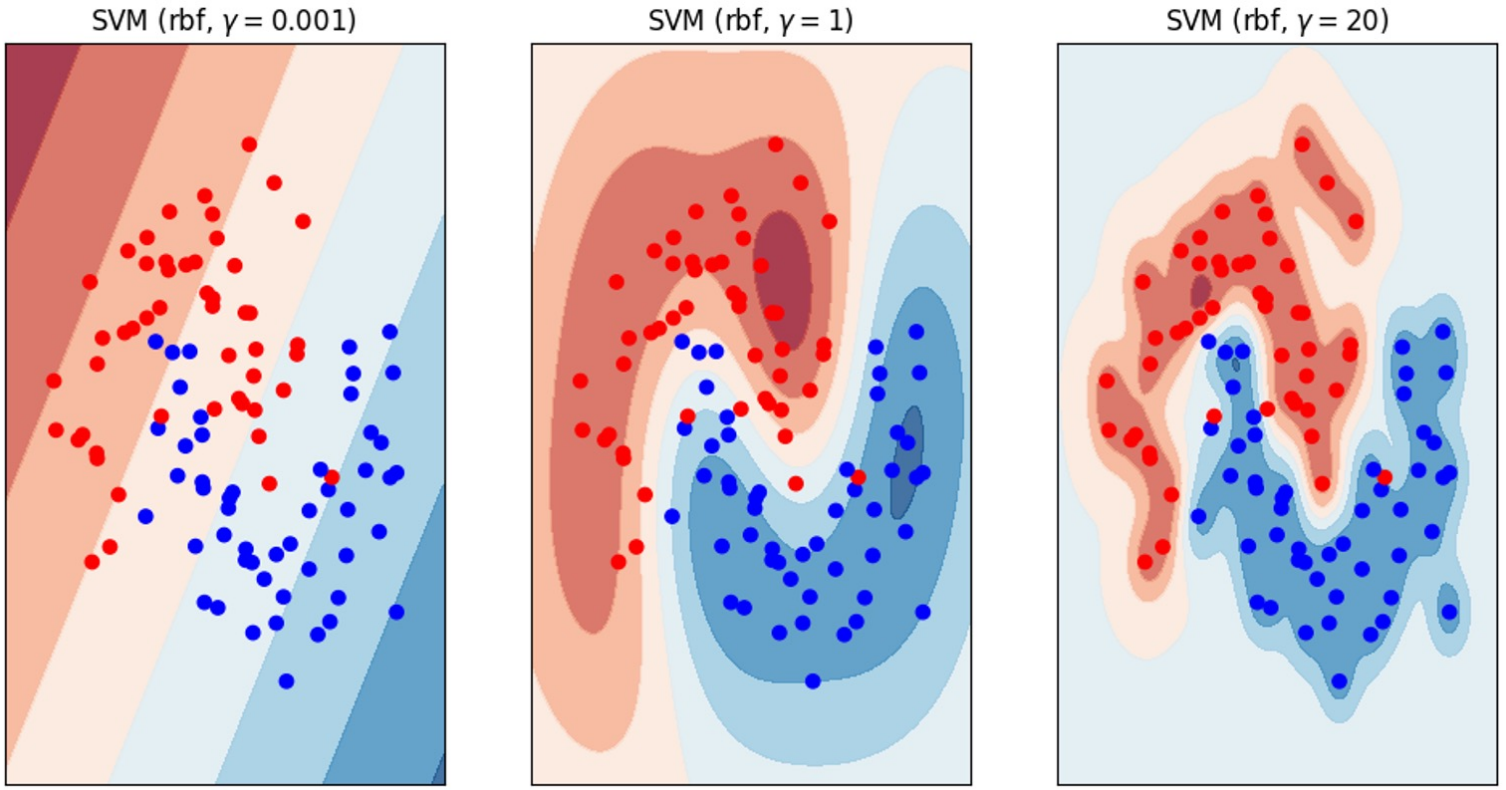
Overfitting and Underfitting (Cont.)

- The decision boundaries generated by under-fitting and over-fitting models on a two-dimensional data.



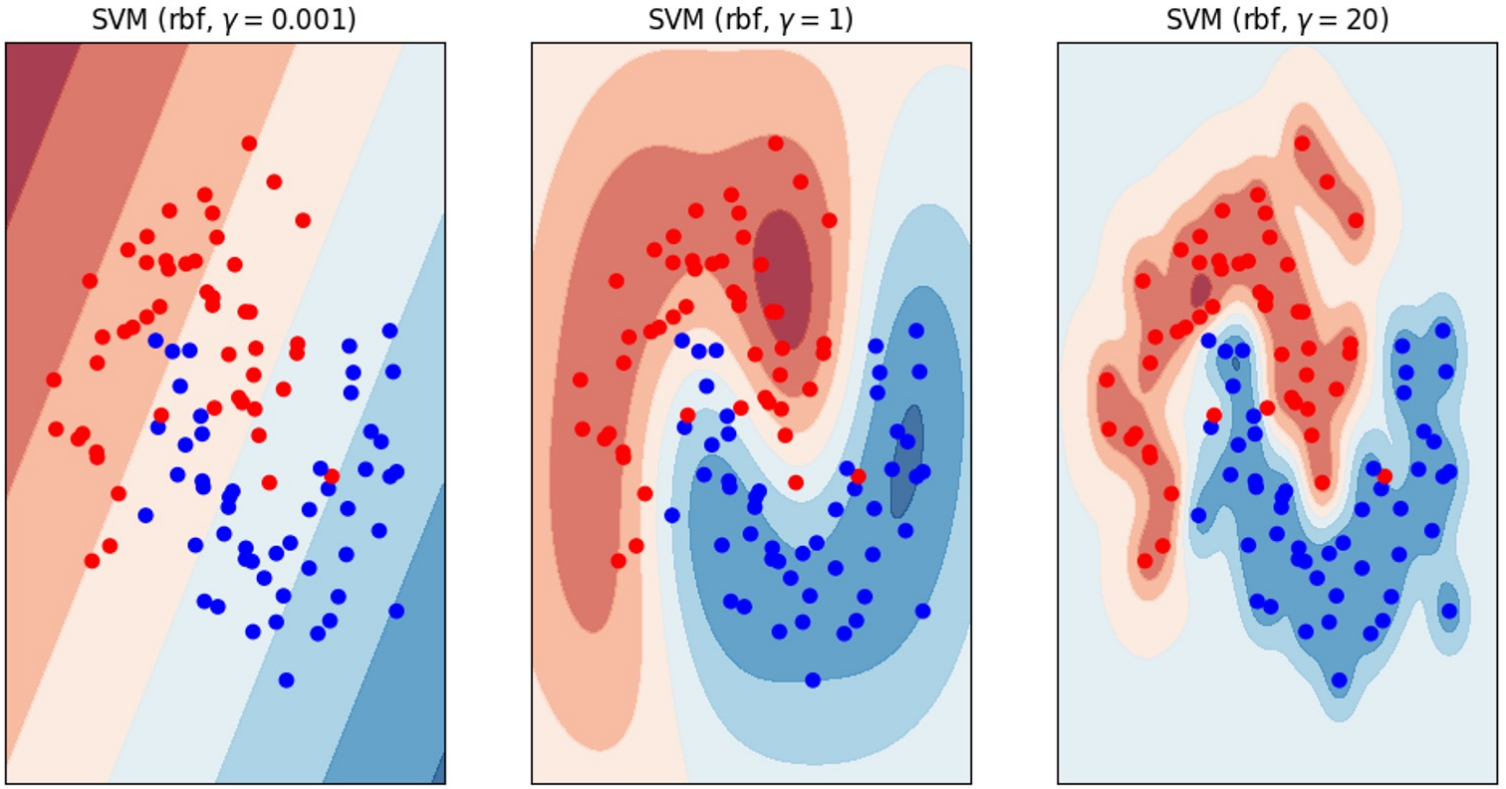
Overfitting and Underfitting

- The decision boundaries generated by under-fitting and over-fitting models on a two-dimensional data.



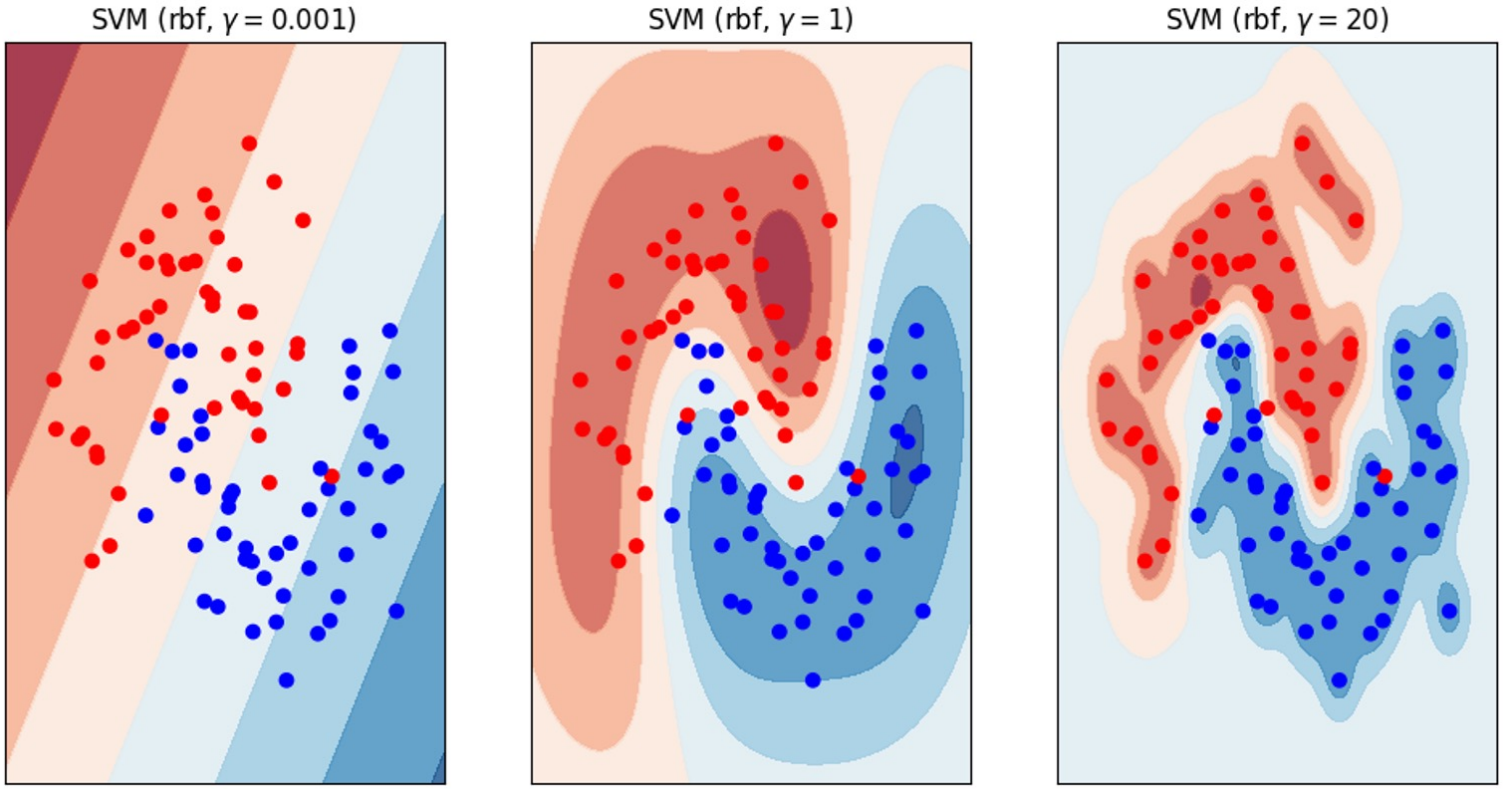
Overfitting and Underfitting

- Underfitting: The model on the left is not complex enough to model the underlying data The model predictions can either be trusted and nor deployed in real-world scenarios.



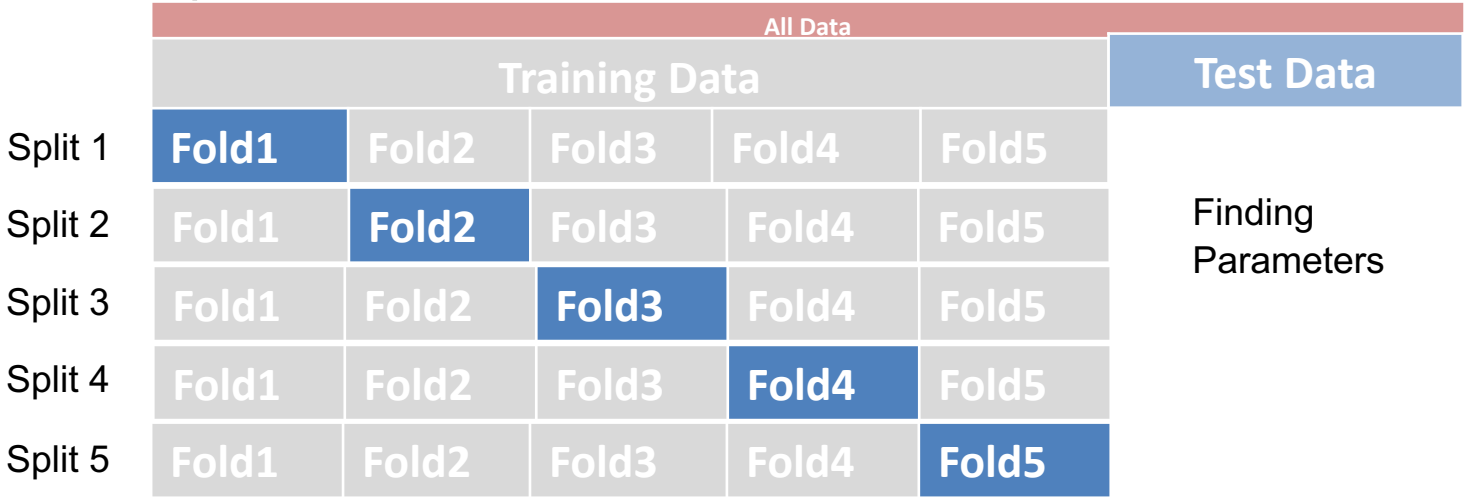
Overfitting and Underfitting

- Overfitting: The right model fits exactly the training data. The decision boundary wraps exactly around each sample of the data. This is a complex decision boundary, where the model reflects the noise in the training data, but do not generalize.



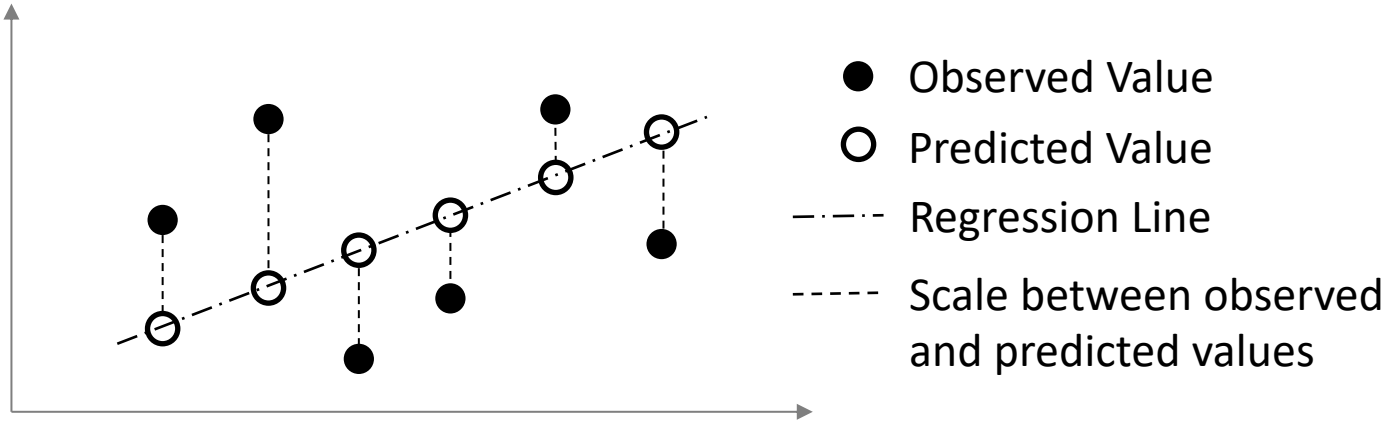
Cross Validation

- K-Fold Cross Validation: the most widely used method for estimating the prediction error.
  - Split the data into k roughly equally-sized partitions.
  - Each part is test set once, join $k - 1$ parts for training.
  - Obtain k test errors and average.
  - Fraction $(k - 1)/k$ is used for training and each observation is tested exactly once.



Measures for Regression

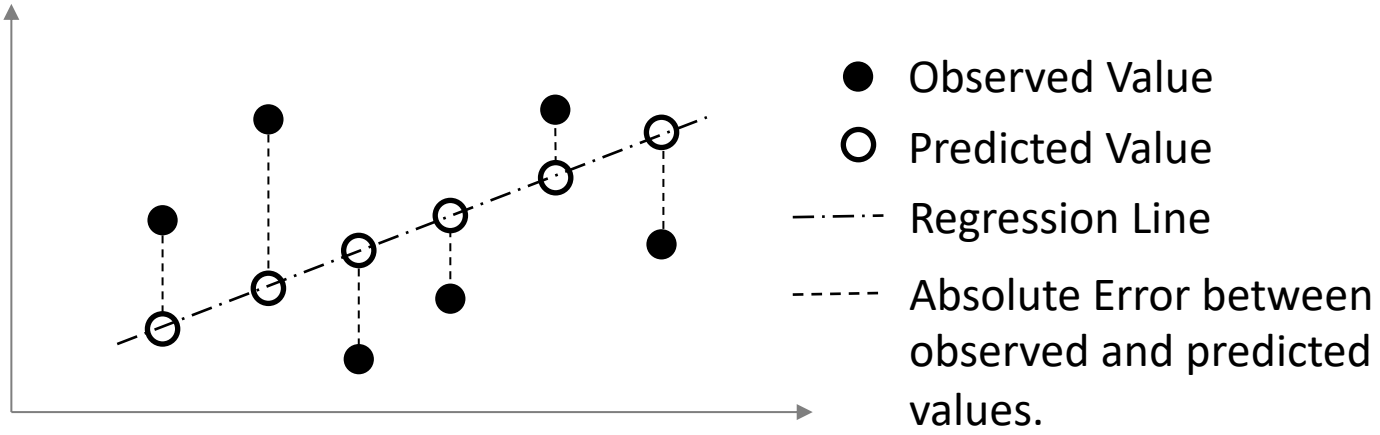
- Mean Square Error (MSE):
  - $\rho_{MSE}(y, F) = \frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)^2 \in [0\,;\,\infty)$ for m samples with prediction $\hat{y}^{(i)}$ and $y^{(i)}$ ground-truth / target.
  - It is known as L2 loss.



- Similar Measures:
  - Sum of Squared Errors: $\rho_{MSE}(y, F) = \sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)^2$.
  - Root MSE (org. Scale): $\rho_{RMSE}(y, F) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y^{(i)} - \hat{y}^{(i)})^2}$.

Measures for Regression (Cont.)

- Mean Absolute Error (MAE):
  - $\rho_{MAE}(y, F) = \frac{1}{m}\sum_{i=1}^{m}\left|y^{(i)} - \hat{y}^{(i)}\right| \in [0\,;\,\infty)$.
  - It is known as L1 loss.



- Similar Measures:
  - Median Absolute Error (even for more robustness).
  - *What is the additional robustness of the median absolute error?*

Measures for Regression (Cont.)

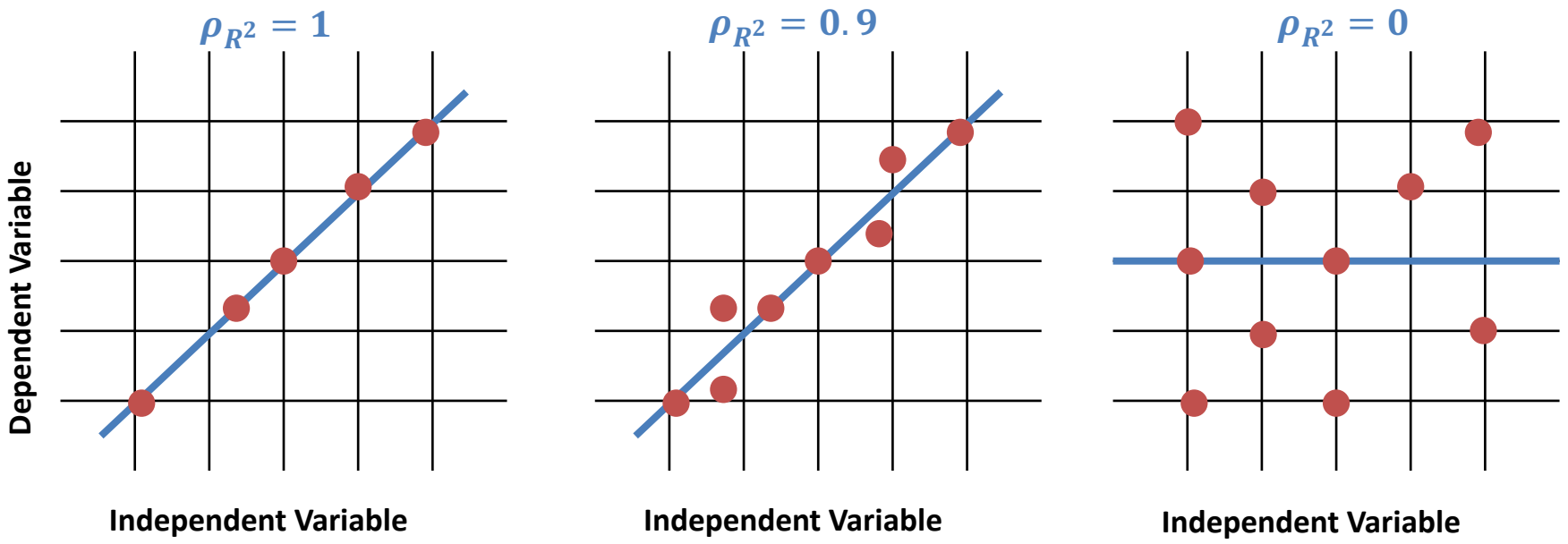
- Coefficient of Determination: R-squared (R^2)
  - $\rho_{R^2}(y, F) = 1 - \frac{\sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)^2}{\sum_{i=1}^{m}\left(y^{(i)} - \bar{y}\right)^2} = 1 - \frac{SSE}{SST}.$

- $\bar{y}$ is the mean.
- $y^{(i)} - \hat{y}^{(i)}$ is the residual.
- Well-known classical measure for regression.
- It measures how strong is the linear relationship between to variables.
- $SSE = \sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)^2$ is the sum of squares of residuals.
- $SST = \sum_{i=1}^{m}\left(y^{(i)} - \bar{y}\right)^2$ is the total sum of squares (like the data variance).
- ρ_{R^2} = 1: all residuals are 0, we predict perfectly.
- ρ_{R^2} = 0.9: the trained model reduces SSE by factor of 10.
- ρ_{R^2} = 0: the trained model makes bad predictions.

Measures for Regression (Cont.)

- Coefficient of Determination: R-squared (R^2) for the linear relationship between two variables.

$$- \rho_{R^2}(y, F) = 1 - \frac{\sum_{i=1}^{m}\left(y^{(i)} - \hat{y}^{(i)}\right)^2}{\sum_{i=1}^{m}\left(y^{(i)} - \bar{y}\right)^2} = 1 - \frac{SSE}{SST}.$$



Measures for Classification: MCE & ACC

- Misclassification Error Rate (MCE): Number of incorrect predictions and presents them as a rate:
  - $\rho_{MCE} = \frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)} \neq \hat{y}^{(i)}\right] \in [0,1]$.

- Accuracy (ACC): Number of correct predictions and presents them as a rate:
  - $\rho_{MCE} = \frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)} = \hat{y}^{(i)}\right] \in [0,1]$.

Measures for Classification: Confusion Matrix

- It is an error matrix with the following form:

	Predicted (False)	Predicted (True)
Actual (False)	True Negative (TN)	False Positive (FP)
Actual (True)	False Negative (FN)	True Positive (TP)

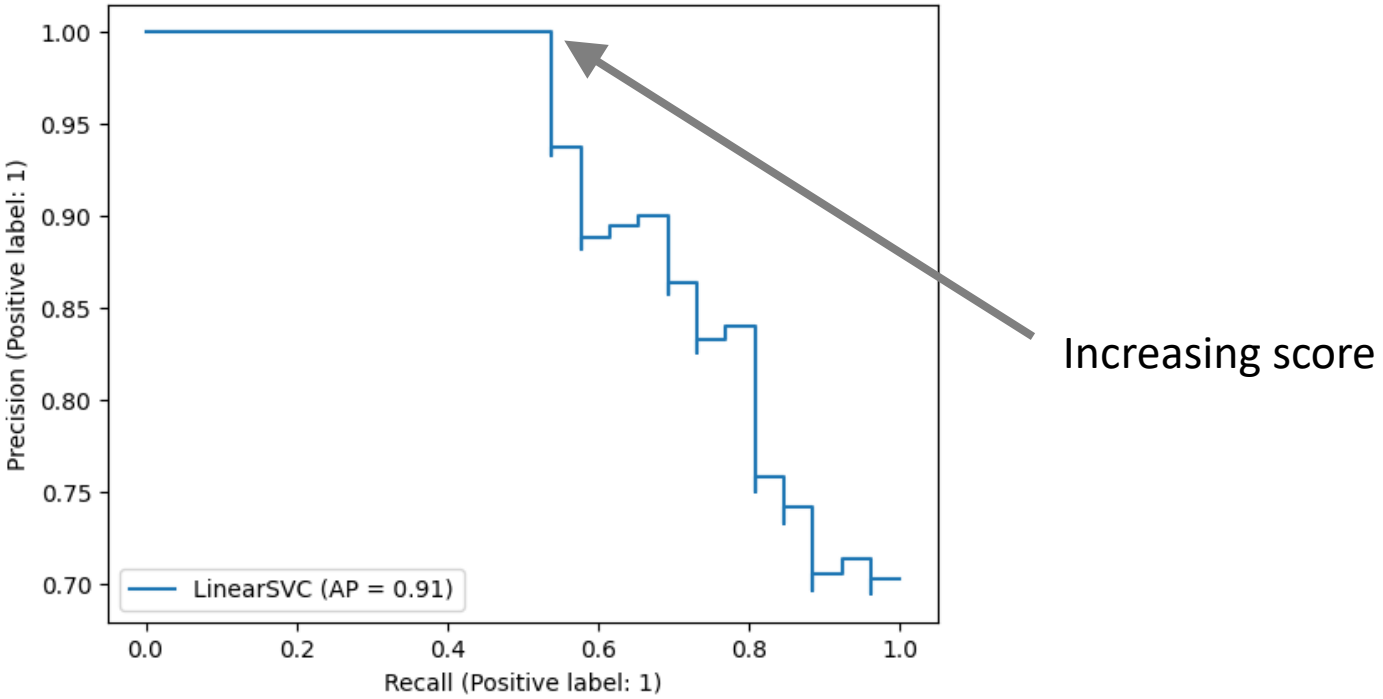
- True positive (TP): the number of correctly predicted samples.
- True negative (TN): the number of correctly predicted samples, as not the considered class.
  - An instance is classified as negative, and it is negative.
- False positive (FP): the number of samples incorrectly predicted as the considered class.
- False negative (FN): the number of samples incorrectly not predicted as the considered class.

Measures for Classification: Precision & Recall

- Consider a binary classifier with the positive and negative class.
- Precision is the ratio between the true positive samples and all the positive samples. It can be defined as:
- $$Precision = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Positive\ (FP)}$$
- Recall is the measure of the correctly classified true positive samples. It can be defined as:
- $$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP) + False\ Negative\ (FN)}$$
- Related measures:
  - Sensitivity is the recall. It corresponds to the proportion of the positive class that got correctly classified.
  - Specificity the proportion of the negative class of the negative class that go correctly classified. It is given by: $Specificity = \frac{True\ Negative\ (TN)}{True\ Negavtive\ (TN) + False\ Positive\ (FP)}$

Measures for Classification: Precision & Recall (Cont.)

- The figure shows the trade-off between precision and recall for different thresholds.
  - A high area under the curve represents both high recall and high precision ➔ (High precision = Low False Positive Rate)



Measures for Classification

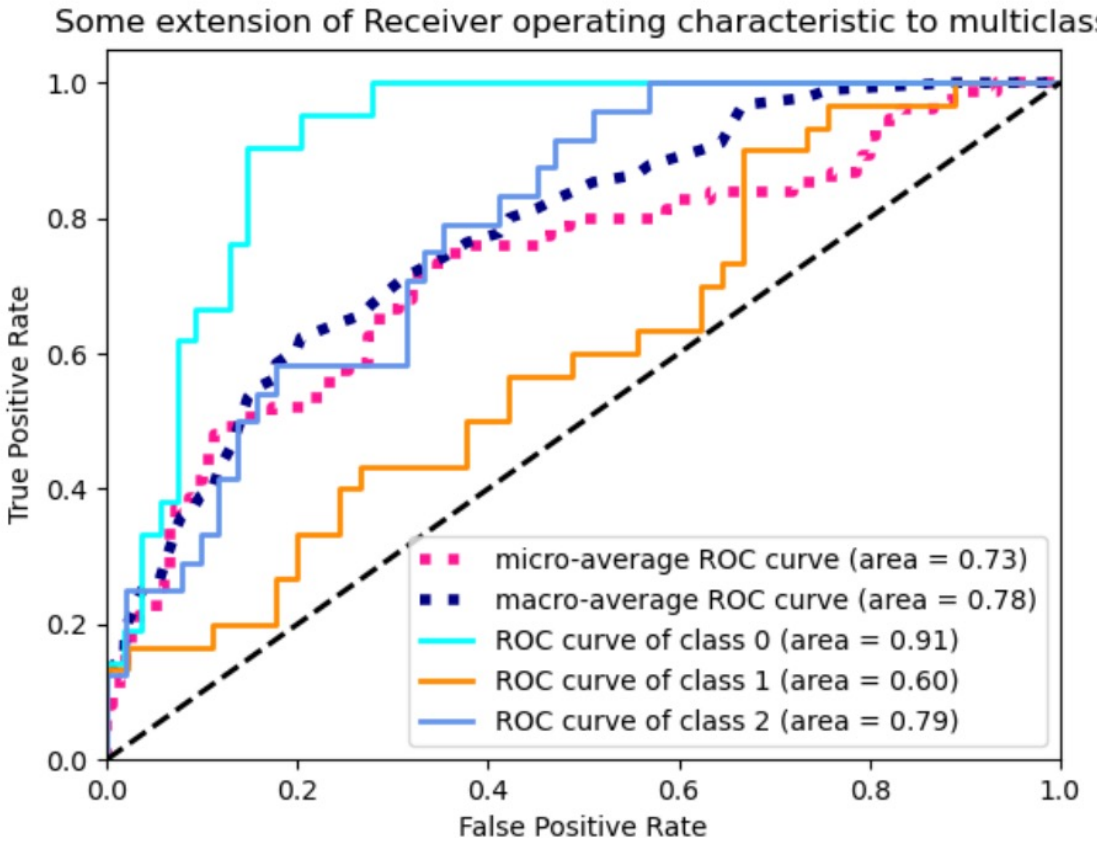
- F1-Score: it is the harmonic mean of the precision and recall. It is given by:

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

- The Receiver Operating Curve (ROC) is a graph showing the classification performance of a model at all classification thresholds. This curve plots two parameters:
  - True Positive Rate (Recall).
  - False Positive Rate (False alarm).

Receiver Operating Curve (ROC)

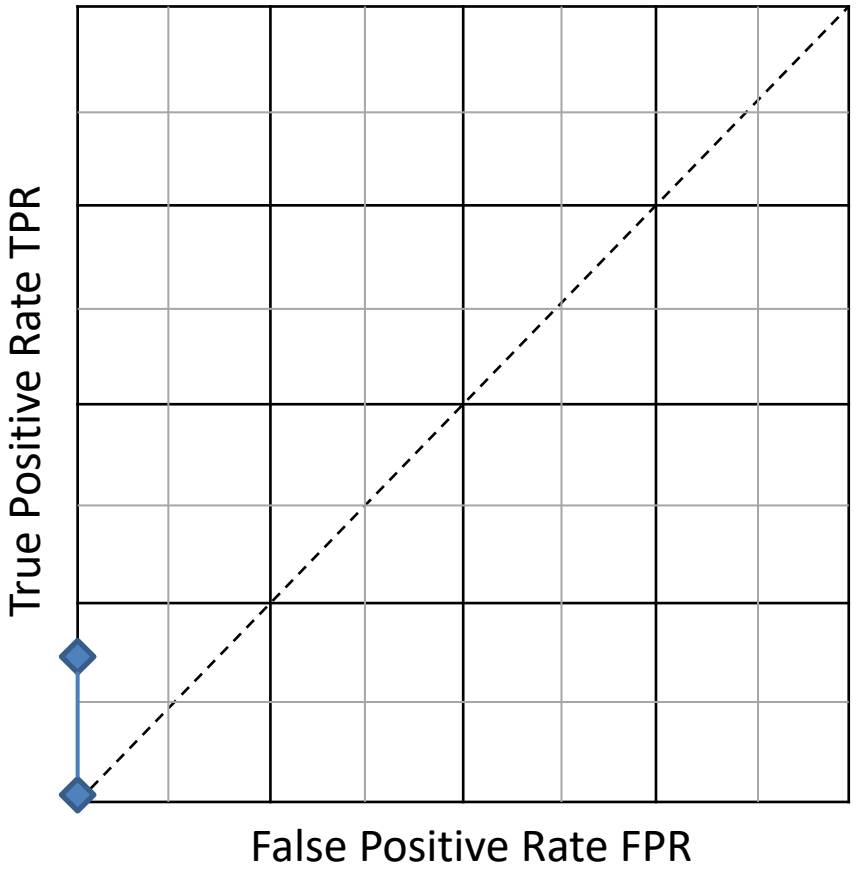
- The diagonal line corresponds to a random classifier (line of no-discrimination).



Computation of ROC Curve

- Example of Receiver Operating Curve (ROC) Computation.

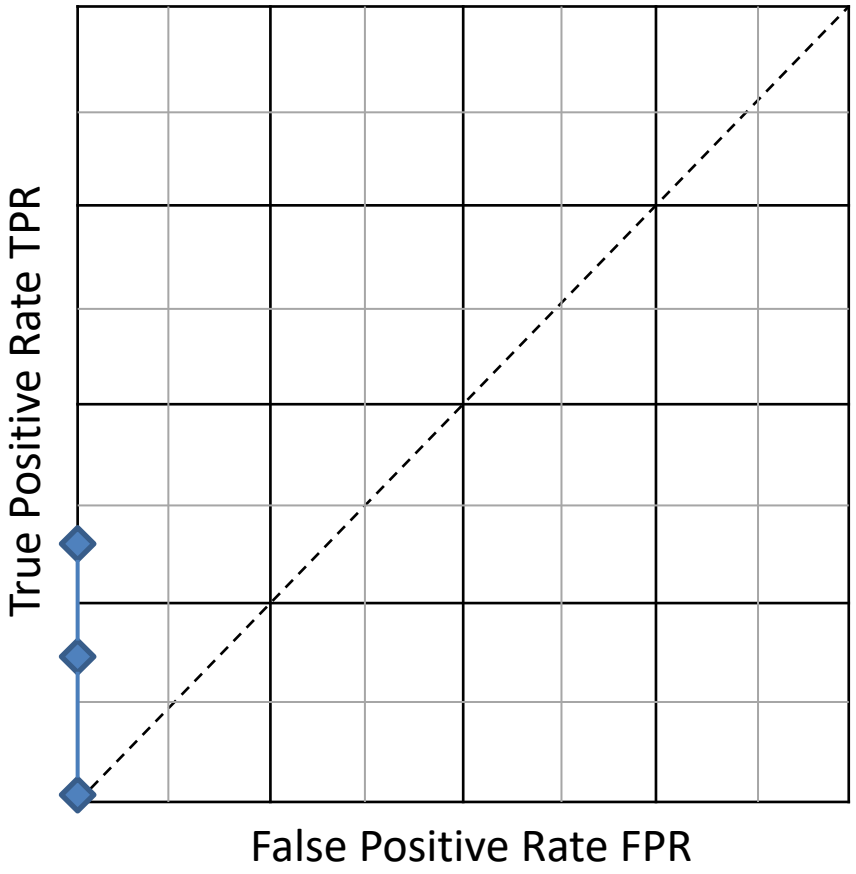
	Predict Correct	Score
1	**Yes**	**0.95**
2	Yes	0.86
3	Yes	0.69
4	No	0.65
5	Yes	0.59
6	No	0.52
7	No	0.39
8	No	0.28
9	Yes	0.15
10	No	0.06



Computation of ROC Curve (Cont.)

- Example of Receiver Operating Curve (ROC) Computation.

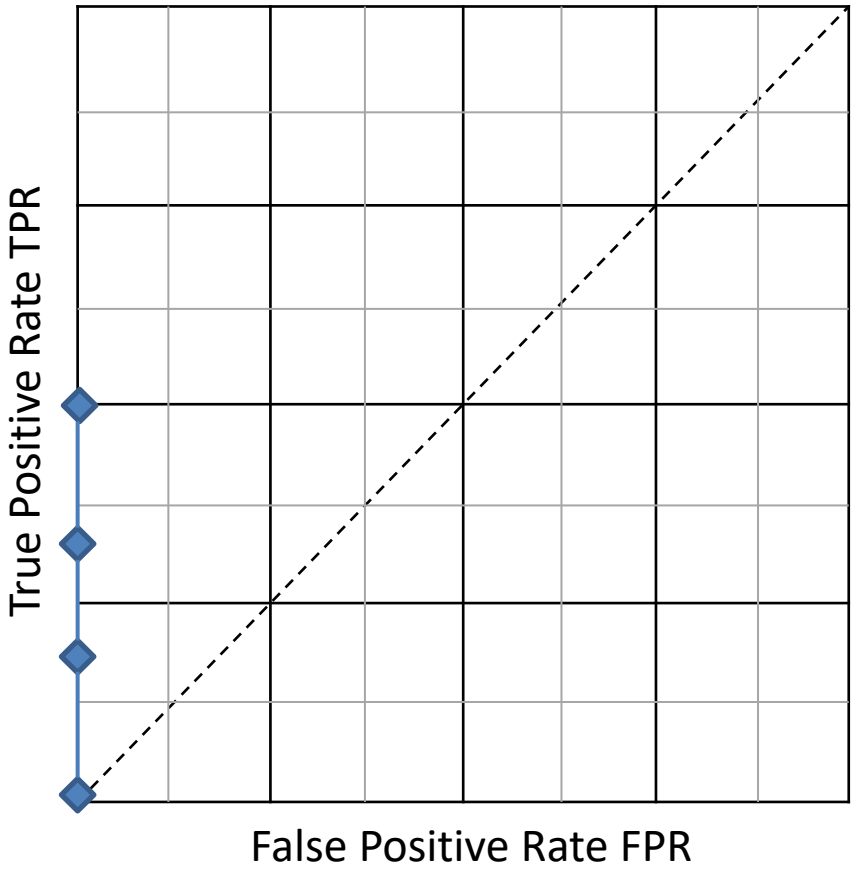
	Predict Correct	Score
1	**Yes**	**0.95**
2	**Yes**	**0.86**
3	Yes	0.69
4	No	0.65
5	Yes	0.59
6	No	0.52
7	No	0.39
8	No	0.28
9	Yes	0.15
10	No	0.06



Computation of ROC Curve (Cont.)

- Example of Receiver Operating Curve (ROC) Computation.

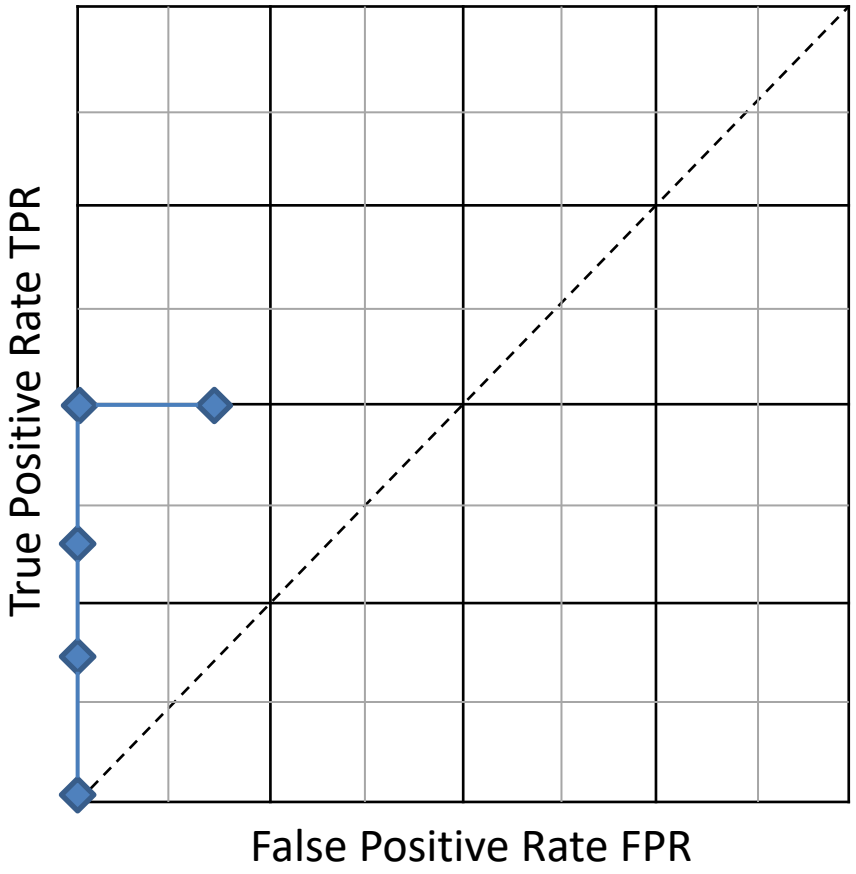
	Predict Correct	Score
1	**Yes**	**0.95**
2	**Yes**	**0.86**
3	**Yes**	**0.69**
4	No	0.65
5	Yes	0.59
6	No	0.52
7	No	0.39
8	No	0.28
9	Yes	0.15
10	No	0.06



Computation of ROC Curve (Cont.)

- Example of Receiver Operating Curve (ROC) Computation.

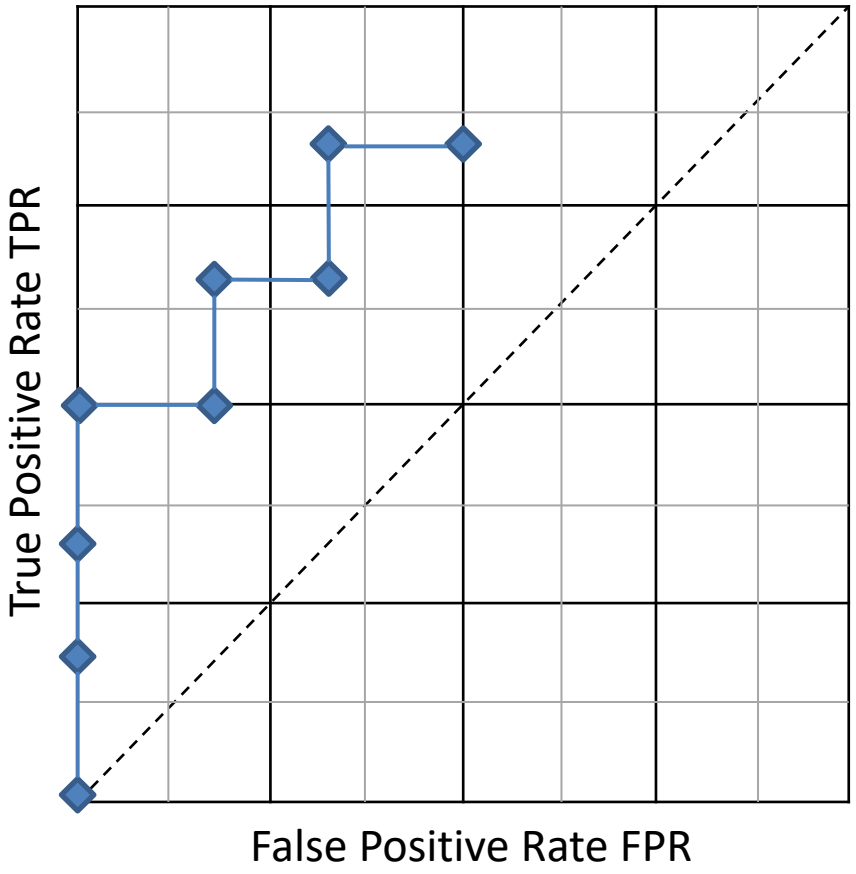
	Predict Correct	Score
1	**Yes**	**0.95**
2	**Yes**	**0.86**
3	**Yes**	**0.69**
4	**No**	**0.65**
5	Yes	0.59
6	No	0.52
7	No	0.39
8	No	0.28
9	Yes	0.15
10	No	0.06



Computation of ROC Curve (Cont.)

- Example of Receiver Operating Curve (ROC) Computation.

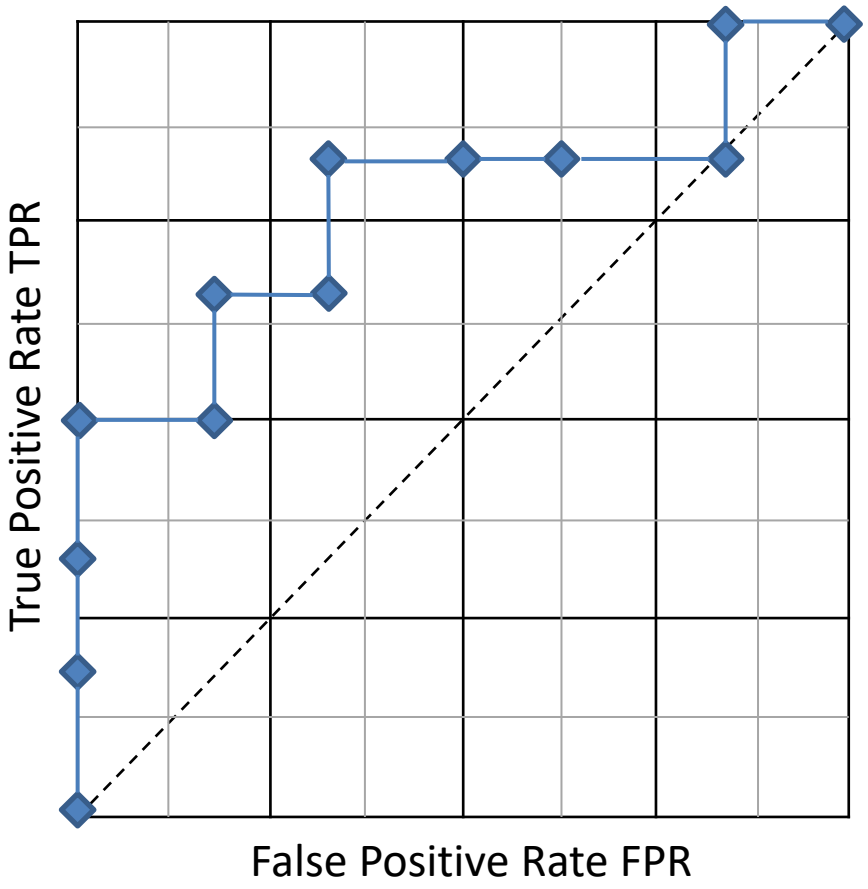
	Predict Correct	Score
1	**Yes**	**0.95**
2	**Yes**	**0.86**
3	**Yes**	**0.69**
4	**No**	**0.65**
5	**Yes**	**0.59**
6	**No**	**0.52**
7	**No**	**0.39**
8	No	0.28
9	Yes	0.15
10	No	0.06



Computation of ROC Curve (Cont.)

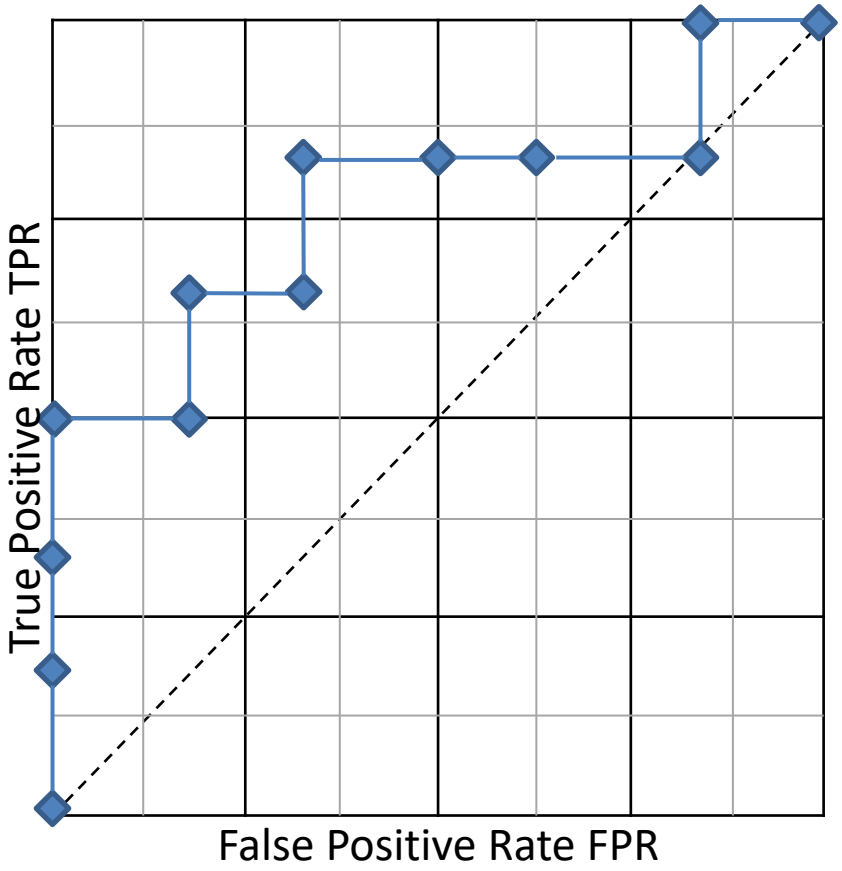
- Example of Receiver Operating Curve (ROC) Computation.

	Predict Correct	Score
1	**Yes**	**0.95**
2	**Yes**	**0.86**
3	**Yes**	**0.69**
4	**No**	**0.65**
5	**Yes**	**0.59**
6	**No**	**0.52**
7	**No**	**0.39**
8	**No**	**0.28**
9	**Yes**	**0.15**
10	**No**	**0.06**



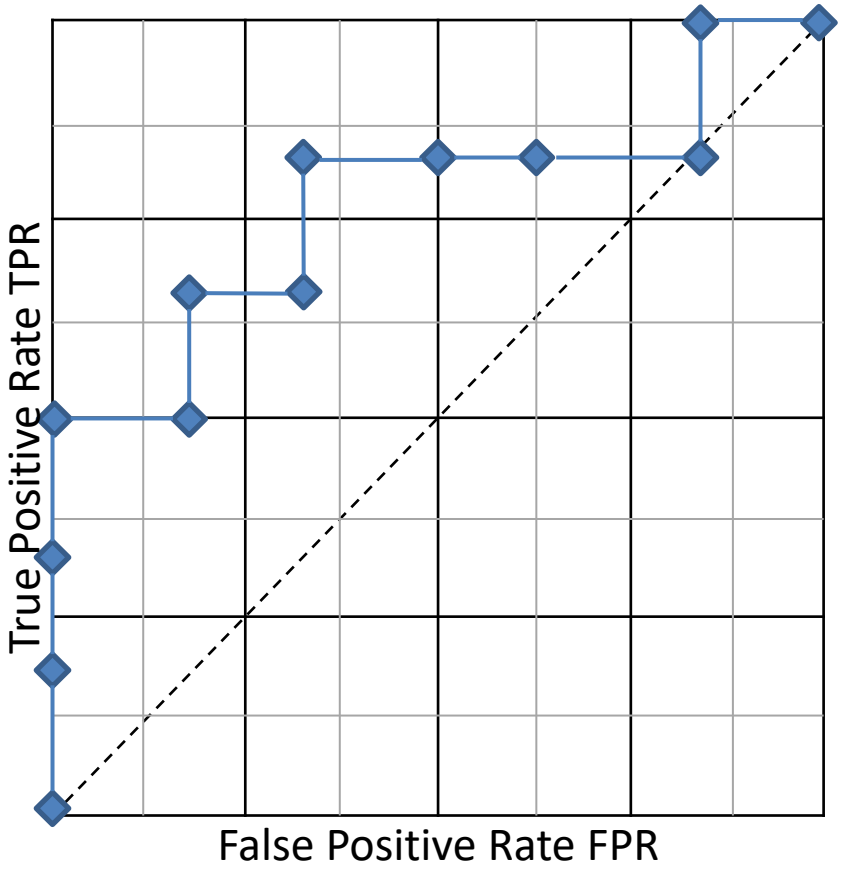
Computation of ROC Curve (Cont.)

- The best classifier is on the top-left corner, where the False Positive Rate FPR equals 0 and the True Positive Rate TPR is maximal.
- The diagonal corresponds to a classifier producing random labels.
- The closer the curve to the top-left corner, the better the model is.



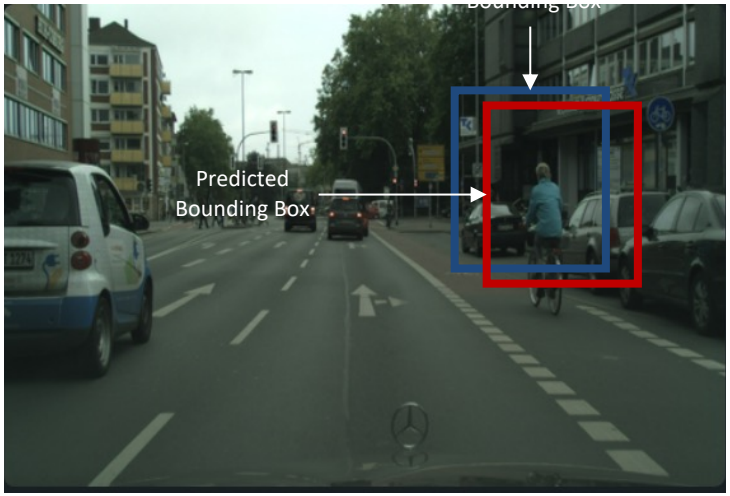
Area Under the ROC Curve (ROC-AUC)

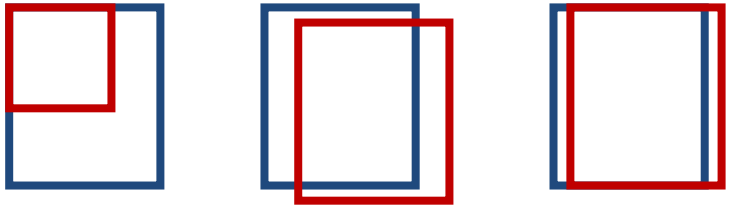
- Area Under the Curve (AUC)
  - $AUC \in [0,1]$ is a single metric to evaluate classifiers. The higher the AUC value, the better the performance of the model at distinguishing between classes.
  - AUC = 1: perfect classifier.
  - AUC = 0.5: random classifier.
  - AUC = 0: perfect, with inverted labels.



Object Detection: Intersection Over Union

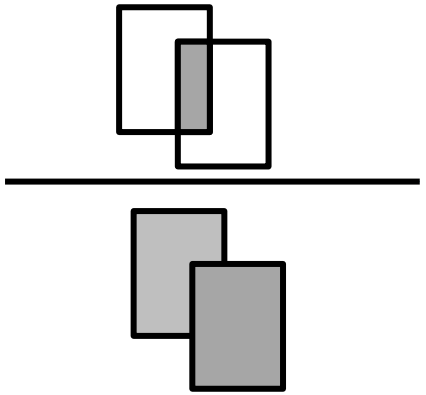
- In object detection, the ground-truth and prediction are described by a bounding box.
- Intersection Over Union (IoU).





IoU = 0.43 IoU = 0.73 IoU = 0.93

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union}$$



- A detection is considered successful if IoU>0.5

Object Detection: Mean Average Precision

- For the multi-class problem, we can computer the mean average precision by applying the precision equation to all classes.
  - $mAP = \frac{1}{classes} \sum_{c \in classes} \frac{TP\,(c)}{TP\,(c)+FP\,(c)}$
- True Positive (TP) : the number of the objects correctly detected (IoU > 0.5).
- False Positive (FP): the number of wrongly detected objects.
- False Negative (FN) : the number of the objects not detected (IoU < 0.5)

Study Material

Not for sharing (LMS, Friedrich-Alexander-Universität Erlangen-Nürnberg)

- *The Elements of Statistical Learning, Trevor Hastie et. al, Chapter 7.*
- *Understanding Machine Learning, Shai Shalev-Schwarz, Chapter 5*
- *Pattern Recognition and Machine Learning, Christopher Bishop, Chapter 1, Section 1.3.*

Next Lecture

Neural Networks