Chapter 04 – Classifier Evaluation

Trust only the statistics you have faked yourself.

Lecture A2I2

Kai Höfig & Dominik Stecher

Goals

- In this slideset you will learn different performance metrics for classification and how to destroy them or fake them for your advantage.
  - Confusion Matrix
  - Accuracy
  - Sensitivity
  - Specificity
  - Precision
  - F score
  - Informedness
  - Markedness
  - Mathews Correlation Coefficient
- *And some stuff for multiple classes or no classes at all as an outlook*

Performance Metrics

- Large variety of metrics available

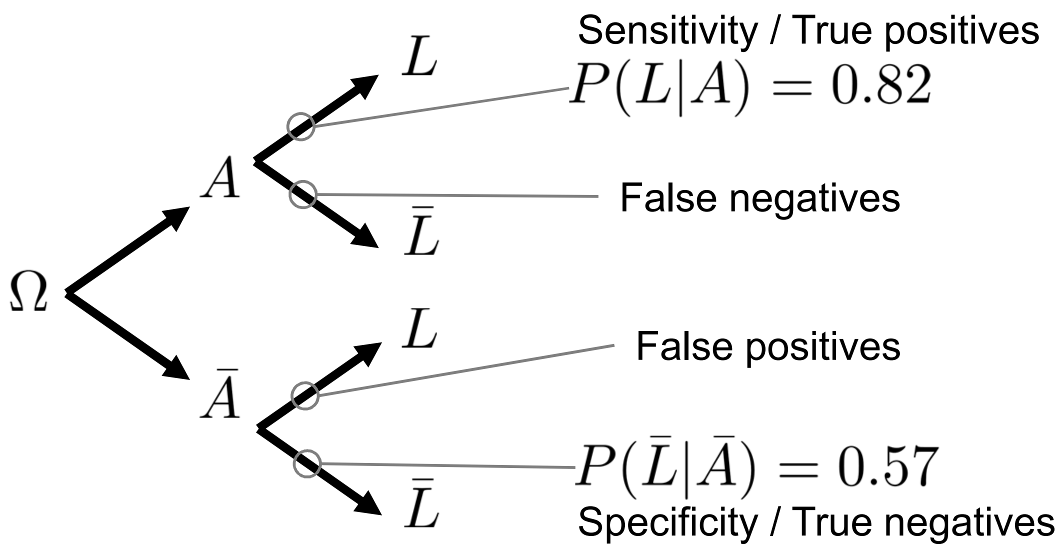
- Choose the right metrics depending on:
  - Dataset attributes
  - Classifier goal (Regression, Classification,…)

- For binary classification:
  - P: All positive samples in the dataset
  - N: All negative samples in the dataset

True positive/negative and False positive/negative

- The probability of increased white blood cells when a person has appendicitis is $P(L|A) = 0.82$ (Sensitivity of the test)

- The probability of normal white blood cell concentration if a person has no appendicitis is $P(\bar{L}|\bar{A}) = 0.57$ (Specificity of the test)

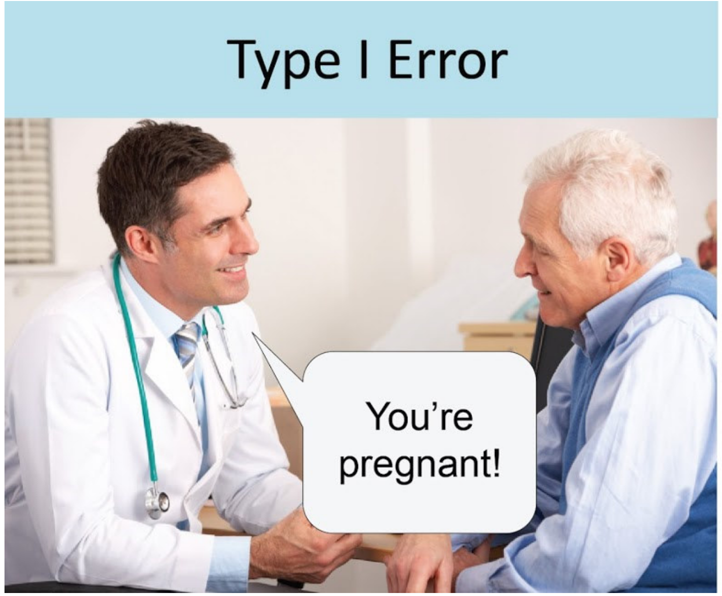


Can be measured during experiments, universal value for the test

Confusion Matrix

- For Classification problems
- Basis for advanced metrics
- False Positive: Type 1 error
  The test erroneously classifies a sample as positive
- False Negative: Type 2 error
  The test erroneously classifies a sample as negative

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)



Sensitivity, recall, hit rate, True Positive Rate (TPR)

- Q: How many P were found?
- Only evaluates P-class

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN}$$

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)

Lets say a perfect test would result in this confusion matrix

10	
	90

But we use a test that always classifies as **true** instead.

10	90

$$TPR = \frac{TP}{P} = \frac{10}{0 + 10} = 100\%$$

Specificity, selectivity, True Negative Rate (TNR)

- Q: How many N were found?
- Only evaluates N-class

$$TNR = \frac{TN}{N} = \frac{\color{red}{TN}}{TN + FP}$$

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)

Lets say a perfect test would result in this confusion matrix

10	
	90

But we use a test that always classifies as **false** instead.

10	90

$$TNR = \frac{TN}{N} = \frac{90}{0 + 90} = 100\%$$

Sensitivity and Specificity belong together and need to be combined for a judgement

Lets say a perfect test would result in this confusion matrix

10	
	90

$$TPR = 100\%$$

$$TNR = 100\%$$

But we use a test that always classifies as **true** instead.

10	90

$$TPR = \frac{TP}{P} = \frac{10}{0+10} = 100\%$$

$$\boldsymbol{TNR = \frac{TN}{P} = \frac{0}{0+90} = 0\%}$$

Lets say a perfect test would result in this confusion matrix

10	
	90

But we use a test that always classifies as **false** instead.

10	90

$$TNR = \frac{TN}{P} = \frac{90}{0+90} = 100\%$$

$$\boldsymbol{TPR = \frac{TP}{P} = \frac{0}{0+10} = 0\%}$$

- Q: How many correct classifications?
- Simple but widespread
- Heavily influenced by the dataset→very misleading

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)

$$ACC = \frac{correctly\ classified}{all\ samples} = \frac{TP + TN}{P + N}$$

Accuracy, easy to trick

- Example HIV-Test
- ~90.000 Infected in GER
- 80.000.000 Healthy people

- $$ACC = \frac{correctly\ classified}{all\ samples}$$

$$= \frac{TP + TN}{P + N}$$

```python
def hiv_classifier(data):
    return False
```

A very bad classifier, but with great accuracy!

90k	80M

$$ACC = \frac{0+80.000.000}{80.000.000+90.000} = 0.998876$$

So, accuracy combines true positives and true negatives, but can be misleading if the dataset is imbalanced.

Precision, Positive/Negative Predictive Rate (PPR/NPR)

- Q: How pure is the positive/negative result?
- Only evaluates positive/negative predictions

$$PPR = \frac{TP}{TP + FP} \qquad NPR = \frac{TN}{FN + TN}$$

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)

Lets say we use a coin flip for classification

25	25
25	25

But we test on a set of only **positives**

50	
50	

$$PPV = 100\%$$

F score, F1 score, F measure (F_1)

- Combines sensitivity (TPR) and precision (PPV) in one value
- Can show similar issues as Accuracy on imbalanced datasets

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)

- $F_1 = 2\frac{PPV * TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$

We classify always true and have 99 cats and one dog in the test-set

99	1
0	0

$$F_1 = \frac{2*99}{2*99+1+0} = 0.995$$

$$TNR = \frac{TN}{N} = \frac{0}{1} = 0\%$$

$$TPR = \frac{TP}{P} = \frac{99}{99} = 100\%$$

Informedness, Bookmaker Informedness (BM)

- Combines sensitivity (TPR) and specificity (TNR) in one value
- Avoids problems on imbalanced datasets

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)

BM = TPR + TNR -1

We classify always true and have 99 cats and one dog in the test-set

99	1
0	0

$$TNR = \frac{TN}{N} = \frac{0}{1} = 0\%$$

$$TPR = \frac{TP}{P} = \frac{99}{99} = 100\%$$

$$BM = 1 + 0 - 1 = 0\%$$

Markedness (MK)

- Combines precision (PPR) and NPR in one value
- Avoids problems on imbalanced datasets
- „Informedness" of the negative class

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)

MK = PPR + NPR -1

$$PPR = \frac{TP}{TP + FP} \qquad NPR = \frac{TN}{FN + TN}$$

We classify always true and have 99 cats and one dog in the test-set

99	1
0	0

$$MK = 99\% + 0\% - 100\% = -1\%$$

Matthews correlation coefficient (MCC)

- Correlation between prediction and observation
- Works well on imbalanced datasets
- Somehow mixes all together.

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

What to do for multiple classes?

- MPCA: Mean Per Class Accuracy
- MPCE: Mean Per Class Error

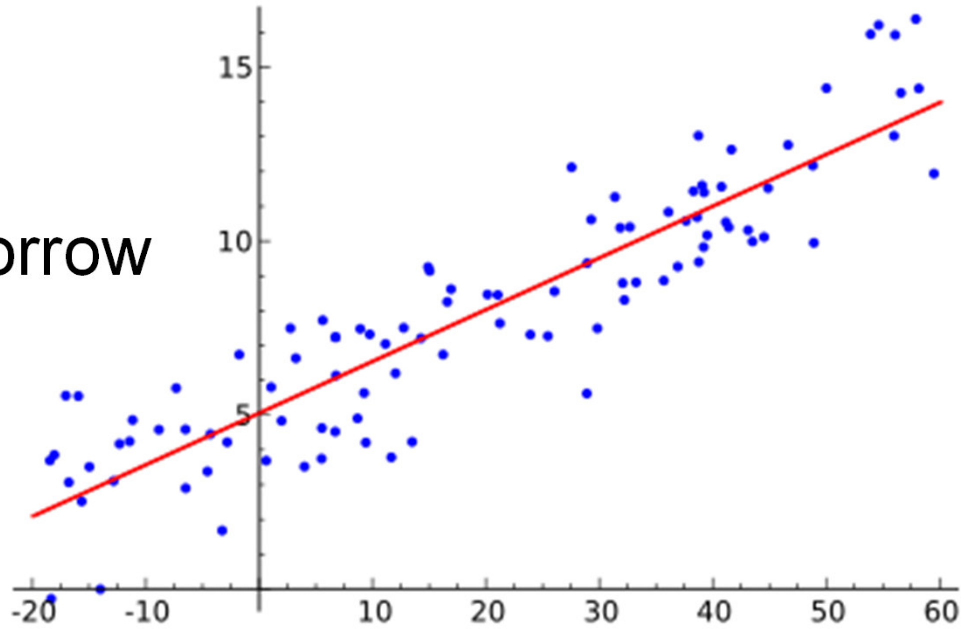
1. Calculate metric per class
2. Calculate mean over n classes

		Actual Class	
		Cat (P)	Not a cat (N)
Predicted Class	Cat	True Positive (TP)	False Positive (FP)
	Not a cat	False Negative (FN)	True Negative (TN)

$$\text{MPCA} = \frac{1}{n} \sum_{i=0}^{n} ACC_i$$

Outlook: What to do without classes?

- Continuous values
- E.g. when using linear Regression
  - Predicting the rent for a flat from other flats using m² and rent.
- Or forecasting in general
  - Predicting the temperature tomorrow from weather data of tody.



Mean Errors

Y: Predicted value X: Observed value n: Number of values

- **Mean Absolute Error (MAE)**

- $MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n}$

- **Mean Absolute Percentage Error (MAPE)**

- $MAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{X_i - Y_i}{X_i} \right|$

- Multiply by 100 for percentage values

Mean Squared Error (MSE)

Y: Predicted value X: Observed value n: Number of values

- Weighted error
  - Many small errors become irrelevant
  - Few large errors are heavily weighted

- $MSE = \frac{1}{n} \sum_{i=1}^{n} (X_i - Y_i)^2$

Compare to the standard deviation

$$\sigma_X := \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{X})^2}$$

Coefficient of determination (R^2)

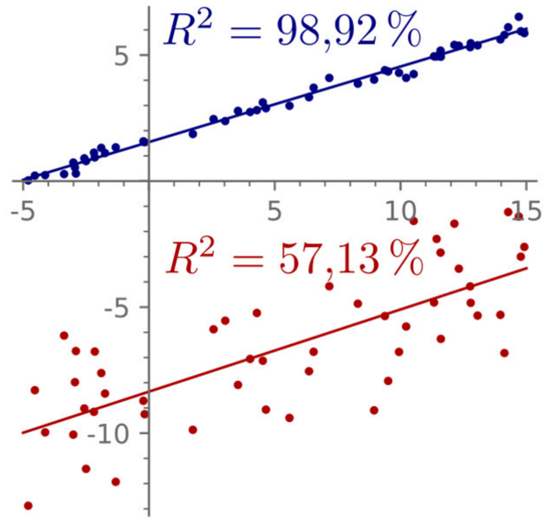
Y: Predicted value   X: Observed value   n: Number of values

- How well does the model predict/describe the dependent variable

- $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ *Mean*

- $SQR = \sum_{i=1}^{n} (x_i - y_i)^2$ ***S**um of **S**quares **R**esidual*

- $SQT = \sum_{i=1}^{n} (x_i - \bar{x})^2$ ***S**um of **S**quares **T**otal*



- $R^2 = 1 - \frac{SQR}{SQT}$

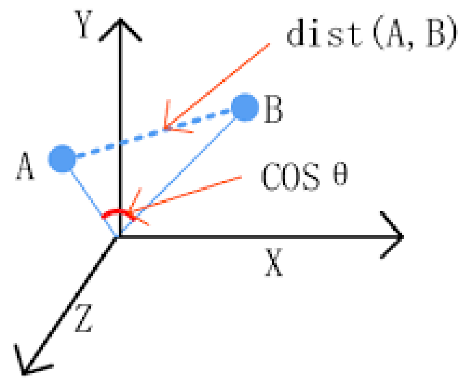
  Deviations from prediction
  Divided by
  Deviations from mean in reality

- Interpretation: $R^2 = 0.67$ → 67% of the variability is fitted well to the model.

Cosine Similarity

Y: Predicted value X: Observed value n: Number of values

- Measures distance **between two vectors**
  - -1: vectors pointing in opposing directions → maximum dissimilarity
  - 0: vectors at 90° → no correlation
  - 1: vectors collinear → perfect match



- $$similarity = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Exercise

1. Use the following measures to judge the quality of the classifications you did so far in the notebooks for Golf, SMS Spam and Titanic

   - Confusion Matrix
   - TPR
   - TNR
   - Accuracy
   - PPR
   - NPR
   - F-Score
   - BM
   - MK
   - MCC

   Interpret the results. Which values are most important to you for the individual classification problems?

2. Apply MPCA to the Iris Notebook

   What if Setosa and Virginica are for Salad and Versicolor is poisonous?

Summary of Chapter 2,3,4

- Naïve Bayes Classification
  - What is classification in general?
  - What is the naïve assumption in the classifier?
  - How does it classify?
- Decision Trees
  - What are the elements of a decision tree?
  - What is the basic principle to learn a tree from data?
  - What are the benefits if a decision tree is used?
  - *..and there is this entropy thing.*
- Classifier Evaluation
  - What are the classic evaluation techniques for classification
  - The more the values of a confusion matrix are aggregated, the harder is an interpretation
  - The application domain is required to evaluate a classifier