METRICS AND EVALUATION

Machine Learning for Autonomous Robots

Dr. Su-Kyoung Kim
DFKI, Robotics Innovation Center

November 1, 2022 – Bremen, Deutschland

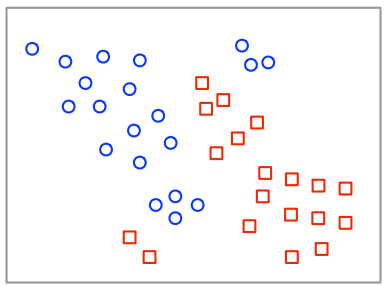
University of Bremen

German Research Center for Artificial Intelligence

Confusion Matrix

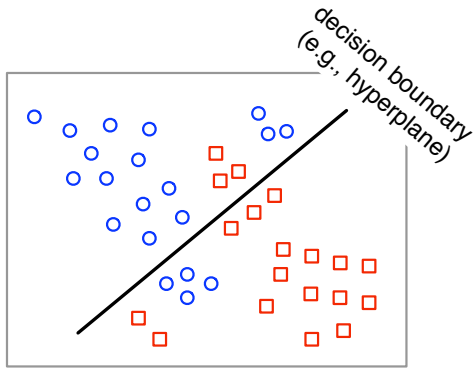
Binary classification

- There are two classes.
  - Class A: *circle*
  - Class B: *square*



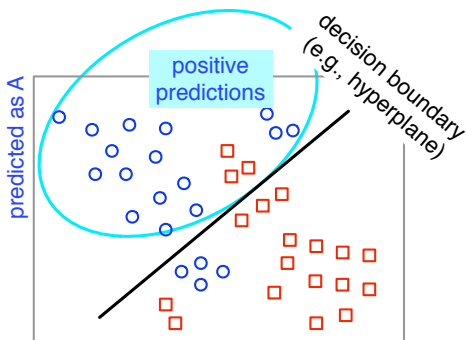
Binary classification

- A classifier is trained to distinguish between two classes.



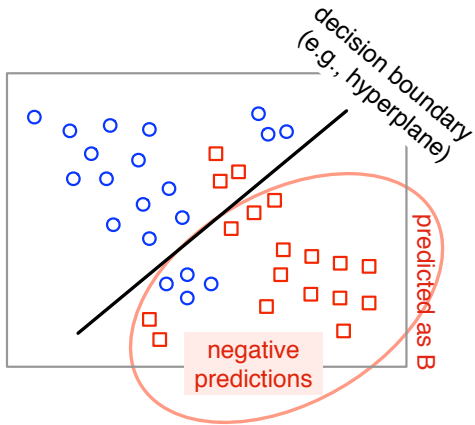
Binary classification

☐ Instances are predicted as positive; they are positive predictions.



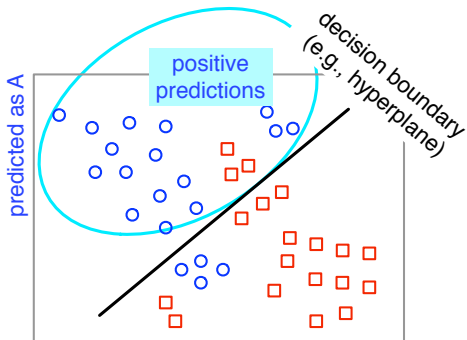
Binary classification

☐ Instances are predicted as negative, they are negative predictions.



Binary classification

☐ Instances are predicted as positive, they are positive predictions.



Binary classification

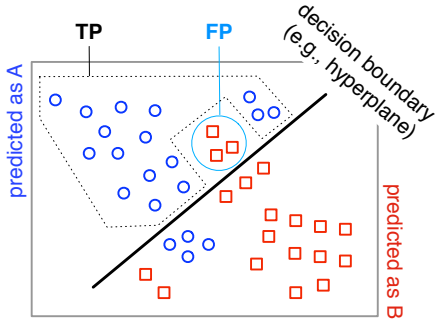
Positive prediction

- Two kinds of positive predictions
  - A is classified as $A \rightarrow$ correct classification
  - B is classified as $A \rightarrow$ wrong classification

  Note that A stands for the positive class

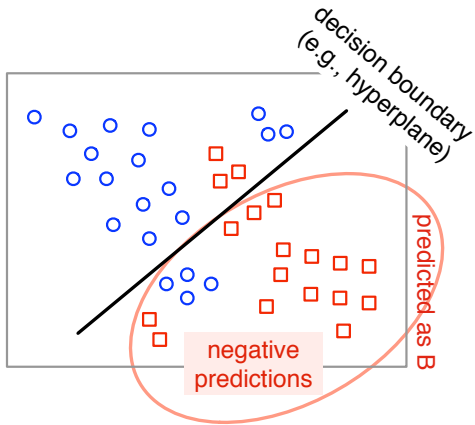
Binary classification

- ☐ Instances are correctly classified as positive. → True Positive (TP)
- ☐ Instances are wrongly classified as positive. → False Positive (FP)



Binary classification

☐ Instances are predicted as negative; they are negative predictions.



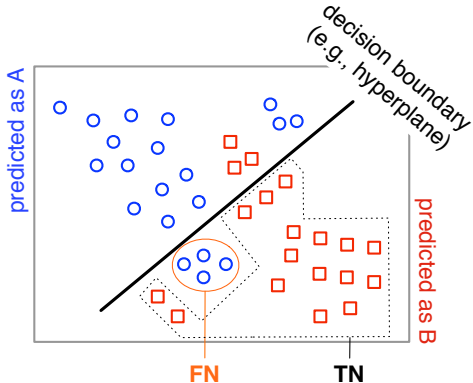
Negative prediction

- Two kinds of negative predictions
  - B is classified as $B \rightarrow$ correct classification
  - A is classified as $B \rightarrow$ wrong classification

  Note that B stands for the negative class

Binary classification

- ☐ Instances are correctly classified as negative. → True Negative (TN)
- ☐ Instances are wrongly classified as negative. → False Negative (FN)



Confusion matrix

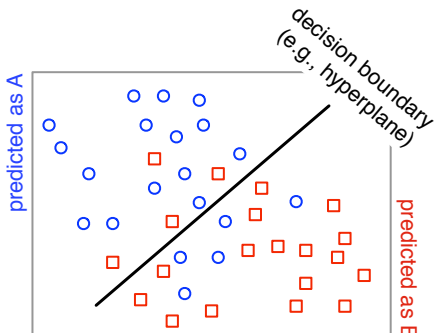
Confusion matrix is a table that describes classification performances.

		prediction (classified as)	
		positive	negative
actual label (labelled as)	positive	TP (# of TPs)	FN (# of FNs)
	negative	FP (# of FPs)	TN (# of TNs)

Exercise 1 (binary classification)

The classifier predicted 15 as positive from the 20 actual positives and 16 as negative from the 20 actual negatives. The circle (class A) stands for the positive class. The square (class B) stands for the negative class.

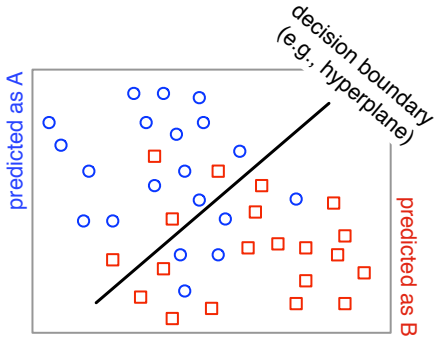
Calculate the number of TPs, FPs, TNs, and FNs!



Solution (exercise 1)

The classifier predicted 15 as positive from the 20 actual positives and 16 as negative from the 20 actual negatives. The circle (class A) stands for the positive class. The square (class B) stands for the negative class.

15 TPs, 4 FPs, 16 TNs, and 5 FNs



Exercise 2 (confusion matrix)

A classifier was trained on a large set of training data. The trained classifier was evaluated on the test dataset which contains 100 instances. The classifier predicted 40 as positive from the 50 actual positives and 45 as negative from the 50 actual negatives.

Build a confusion matrix!

Solution (exercise 2)

The trained classifier was evaluated on the test dataset which contains 100 instances. The classifier predicted 40 as positive from the 50 actual positives and 45 as negative from the 50 actual negatives.

40 TPs, 5 FPs, 10 FNs, 45 TNs

		prediction (classified as)	
		positive	negative
actual label (labelled as)	positive	40	10
	negative	5	45

Multi-class classification

Three classes: A, B, C (correct classifications: diagonal entries)

		prediction (classified as)		
		A	B	C
actual label (labelled as)	A	TP_{aa}		
	B		TP_{bb}	
	C			TP_{cc}

Multi-class classification: Class A

A stands for the positive class; B and C stand for the negative class.

		prediction (classified as)		
		A	B	C
actual label (labelled as)	**A**	TP_{aa}	FN_{ab}	FN_{ac}
	B	FP_{ba}	TN_{bb}	FN_{bc}
	C	FP_{ca}	FN_{cb}	TN_{cc}

Multi-class classification: Class B

B stands for the positive class; A and C stand for the negative class.

		prediction (classified as)		
		A	**B**	C
actual label (labelled as)	A	TN_{aa}	FP_{ab}	FN_{ac}
	B	FN_{ba}	TP_{bb}	FN_{bc}
	C	FN_{ca}	FP_{cb}	TN_{cc}

Multi-class classification: Class C

C stands for the positive class; A and B stand for the negative class.

		prediction (classified as)		
		A	B	**C**
actual label (labelled as)	A	TN_{aa}	FN_{ab}	FP_{ac}
	B	FN_{ab}	TN_{bb}	FP_{bc}
	C	FN_{ac}	FN_{cb}	TP_{cc}

Metrics

Metrics: accuracy, TPR, TNR, FPR, FNR

- Accuracy (ACC) $= \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP} + \mathrm{TN}}$

  The number of all correct predictions (TP+TN) divided by the total number of dataset.

- True positive rate (TPR) $= \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$, i.e., sensitivity or recall

  The number of correct positive predictions (TP) divided by the total number of (actual) positives (TP+FN)

- True negative rate (TNR) $= \frac{\mathrm{TN}}{\mathrm{FP} + \mathrm{TN}}$, i.e., specificity

  The number of correct negative predictions (TN) divided by the total number of (actual) negatives (TN+FP)

Metrics: accuracy, TPR, TNR, FPR, FNR (continued)

- False positive rate (FPR) $= \frac{\text{FP}}{\text{FP} + \text{TN}}$, i.e., 1-TNR or α error

  The number of incorrect positive predictions (FP) divided by the total number of (actual) negatives (TN+FP).

- False negative rate (FNR) $= \frac{\text{FN}}{\text{TP} + \text{FN}}$, i.e. 1-TPR or β error

  The number of incorrect negative prediction (FN) divided by the total number of actual positives (TP+FN).

Metrics: precision, recall, F-measure, bACC

- Precision $= \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FP}}$, i.e., positive predictive value

  The number of correct positive predictions divided by the total number of positive predictions

- Recall $= \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FN}}$, i.e., TPR

  The number of correct positive predictions divided by the total number of (actual) positives.

Metrics: precision, recall, F-measure, bACC (continued)

- F-measure $= \frac{(1+\beta^2)\cdot\text{precision}\cdot\text{recall}}{\beta\cdot(\text{precision}+\text{recall})}$

  β is commonly $0.5, 1, 2$. Example: β is 1, $F_1 = 2 \cdot \frac{\text{precision}\cdot\text{recall}}{\text{precision}+\text{recall}}$

  A harmonic mean of precision and recall (TPR).

  Precision $= \frac{\text{TP}}{\text{TP}+\text{FP}}$; Recall $= \frac{\text{TP}}{\text{TP}+\text{FN}}$, i.e., TPR

- Balanced accuracy (bACC) $= \frac{(\text{TPR} + \text{TNR})}{2}$

  The arithmetic mean of TPR and TNR

Exercise 3 (metrics)

Calculate the following metrics based on the given confusion matrix: Recall (TPR; sensitivity), precision, F_1-measure, specificity (TNR), accuracy, and balanced accuracy (bACC)!

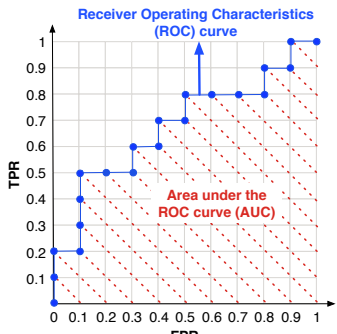
		prediction (classified as)	
		positive	negative
actual label (labelled as)	positive	40	10
	negative	5	45

Solution (exercise 3)

- ☐ Recall (TPR)= $\frac{\mathsf{TP}}{\mathsf{TP}+\mathsf{FN}} = 40/50 = 0.8$
- ☐ Precision = $\frac{\mathsf{TP}}{\mathsf{TP}+\mathsf{FP}} = 40/45 \approx 0.89$
- ☐ F_1-measure = $\frac{(1+\beta^2)\cdot\mathsf{precision}\cdot\mathsf{recall}}{\beta\cdot(\mathsf{precision}+\mathsf{recall})} \approx 0.84$
- ☐ Specificity (TNR) = $\frac{\mathsf{TN}}{\mathsf{FP}+\mathsf{TN}} = 45/50 = 0.9$
- ☐ Accuracy (ACC) = $\frac{\mathsf{TP}+\mathsf{TN}}{\mathsf{TP}+\mathsf{FN}+\mathsf{FP}+\mathsf{TN}} = 85/100 = 0.85$
- ☐ Balanced ACC (bACC) = $\frac{(\mathsf{TPR}+\mathsf{TNR})}{2} = (0.8+0.9)/2 = 0.75$

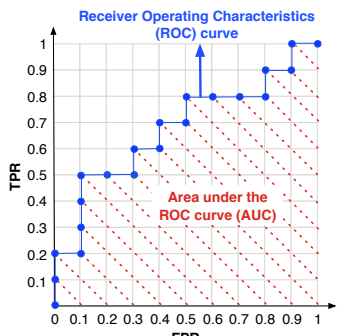
Area under the ROC curve (AUC)

- AUC is a area under the ROC curve.
- ROC: Receiver operating characteristics (ROC)



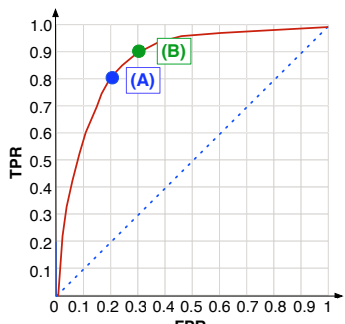
Area under the ROC curve (AUC)

- Receiver operating characteristics (ROC) is a probability curve (CDF of TPR and FPR).
- AUC represents a degree of distinction between two classes and measures performances of classification models.



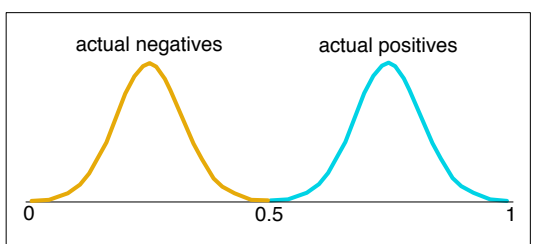
Receiver operating characteristics (ROC) analysis

- The ROC curve is plotted with TPR against the FPR.
- The ROC curve measures performances of classification models at various thresholds settings, e.g., (A) and (B).



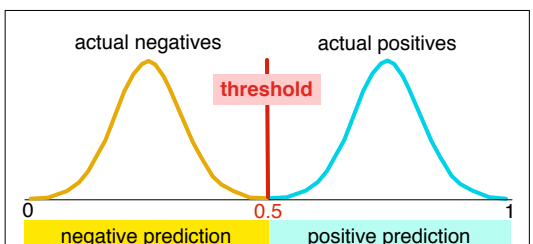
ROC: binary classification

- The dataset has N instances, which consists of two **actual** classes: actual negatives and actual positives.
- Two distributions (for each **actual** class)
  - Yellow distribution stands for actual negatives.
  - Skyblue distribution stands for actual positives.



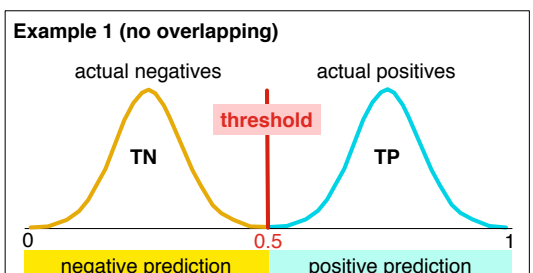
ROC: binary classification

- Each instance has a classification score (x).
- Threshold is given, e.g., threshold of 0.5
  - if $x > 0.5$, the instance is predicted as positive (positive prediction).
  - if $x < 0.5$, the instance is predicted as negative (negative prediction).

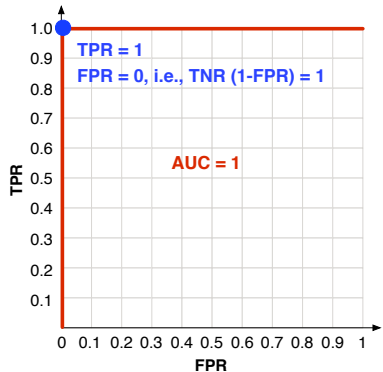


ROC: Example 1 (no overlap; AUC = 1)

- No overlap between two distributions.
- Perfect to distinguish between positive class and negative class.
- All actual positives are predicted as positive and all actual negatives are predicted as negative.
- TPR = 1, FPR = 0, i.e., TNR (1-FPR) = 1, AUC = 1



ROC: Example 1 (no overlap; AUC = 1)

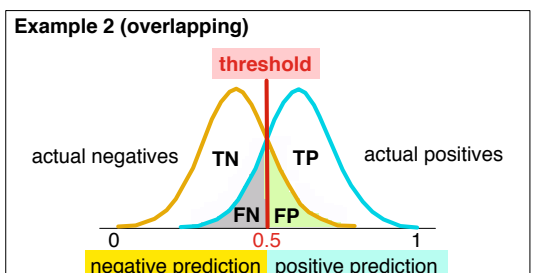


ROC: Example 2 (overlap)

- ☐ Overlap between two distributions (threshold of 0.5).
- ☐ If $x > 0.5$, the instance is predicted as positive. $\rightarrow$ TP or FP
- ☐ If $x < 0.5$, the instance is predicted as negative. $\rightarrow$ TN or FN

Remember: x is a classification score

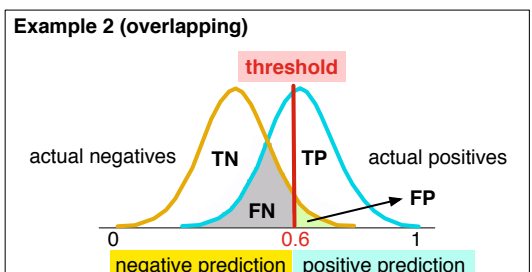
- ☐ The same size of FN and FP in this example.



ROC: Example 2 (overlap; threshold shift 1)

- ☐ We can change the number of FN and FP by shifting the threshold
- ☐ Example: threshold of 0.6
- ☐ If $x > 0.6$, the instance is predicted as positive. $\rightarrow$ FP $\downarrow$
- ☐ If $x < 0.6$, the instance is predicted as negative. $\rightarrow$ FN $\uparrow$

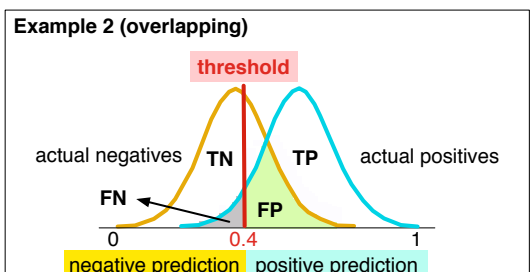
Remember: x is a classification score



ROC: Example 2 (overlap; threshold shift 2)

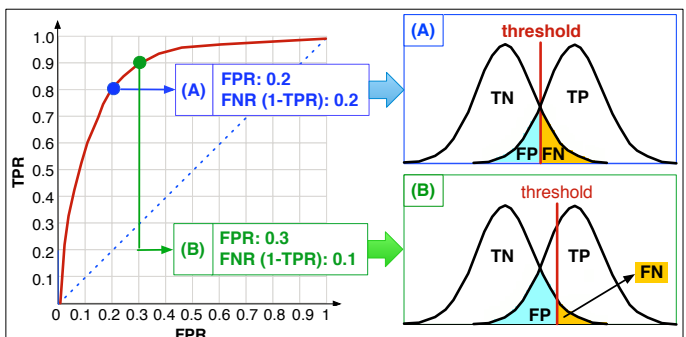
- ☐ We can change the number of FN and FP by shifting the threshold
- ☐ Example: threshold of 0.4
- ☐ If $x > 0.4$, the instance is predicted as positive. $\rightarrow$ FP $\uparrow$
- ☐ If $x < 0.4$, the instance is predicted as negative. $\rightarrow$ FN $\downarrow$

Remember: x is a classification score



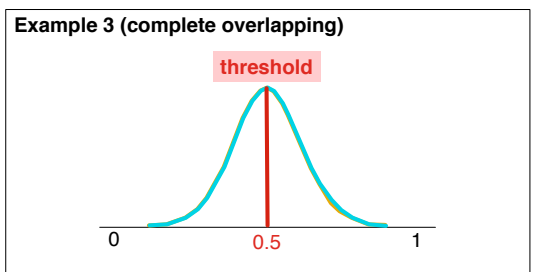
ROC: Example 2 (overlap)

- The ROC curve measures performances of classification models at various thresholds settings, e.g., (A) and (B).



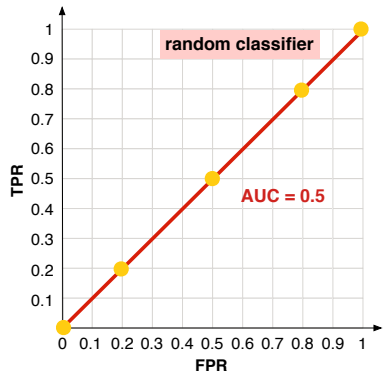
ROC: Example 3 (random classifier)

- Complete overlap between two distributions.
- 50 % chance to distinguish two classes.



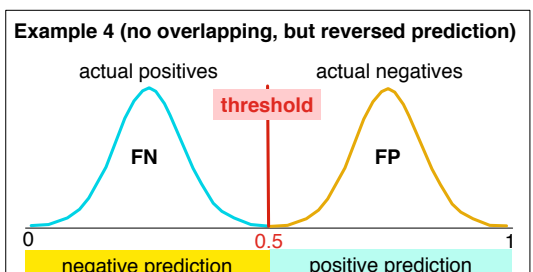
ROC: Example 3 (random classifier)

- Complete overlap between two distributions.
- random classifier
- 50 % chance that to distinguish two classes (AUC = 0.5).



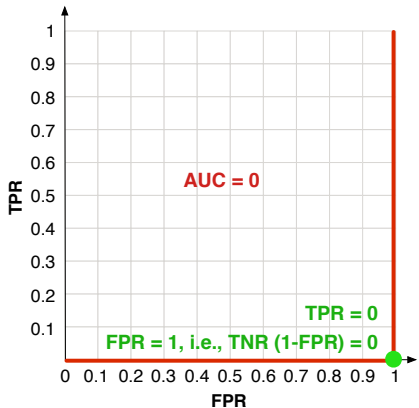
ROC: Example 4 (no overlap; AUC = 0)

- ☐ No overlap between two distributions, but reversed pattern.
- ☐ All actual positives (skyblue distribution) are predicted as negative.
- ☐ All actual negatives (yello distribution) are predicted as positive.
- ☐ TPR = 0, FPR = 1, i.e., TNR (1-FPR) = 0, AUC = 0



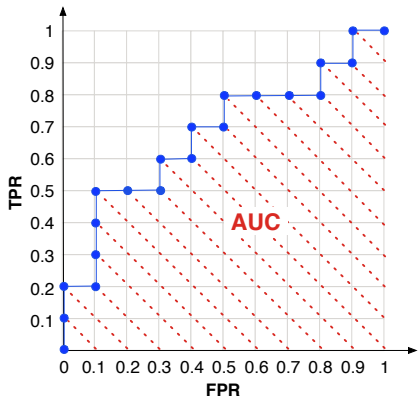
ROC: Example 4 (no overlap; AUC = 0)

- ☐ AUC = 0
- ☐ TPR = 0, FPR = 1, i.e., TNR (1-FPR) = 0, AUC = 0



Area under the ROC curve (AUC) as performance metric

- An excellent model has AUC near to the 1.
- The larger the AUC, the better the performance on average.



Drawing of ROC curve

Recap

- For each instance, a certain score (x) is computed by a classification model.
- Instances are predicted as positive or negative according to the given threshold.
- When the score (x) exceeds the given threshold, the instance is predicted as positive. Otherwise, the instance is predicted as negative.
  - $x > 0.5$, the instance is predicted as positive.
  - $x < 0.5$, the instance is predicted as negative.

Drawing of ROC curve (1a)

Step 1: Sort the scores in descending order!

Drawing of ROC curve (1b)

The scores are sorted in descending order.

instance	score (x)	class	instance	score (x)	class
1	0.90	p	11	0.40	p
2	0.80	p	12	0.39	n
3	0.70	n	13	0.38	p
4	0.65	p	14	0.37	n
5	0.60	p	15	0.36	n
6	0.55	p	16	0.35	n
7	0.54	n	17	0.34	p
8	0.53	n	18	0.33	n
9	0.52	p	19	0.30	p
10	0.51	n	20	0.20	n

Total number of 20 instances; p stands for the actual positive class; n stands for the actual negative class; class (actual class).

Drawing of ROC curve (2)

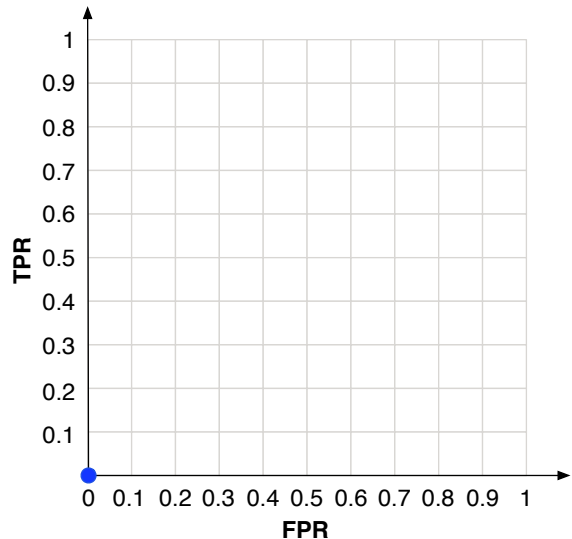
Step 1: Sort the scores in descending order!

Step 2: Start in (0,0)!

Please do not start with the first instance which is sorted in descending order (one of the most frequent mistakes)!

Drawing of ROC curve (1,2)

inst	score	class	inst	score	class
1	0.90	*p*	11	0.40	*p*
2	0.80	*p*	12	0.39	*n*
3	0.70	*n*	13	0.38	*p*
4	0.65	*p*	14	0.37	*n*
5	0.60	*p*	15	0.36	*n*
6	0.55	*p*	16	0.35	*n*
7	0.54	*n*	17	0.34	*p*
8	0.53	*n*	18	0.33	*n*
9	0.52	*p*	19	0.30	*p*
10	0.51	*n*	20	0.20	*n*



Drawing of ROC curve (3)

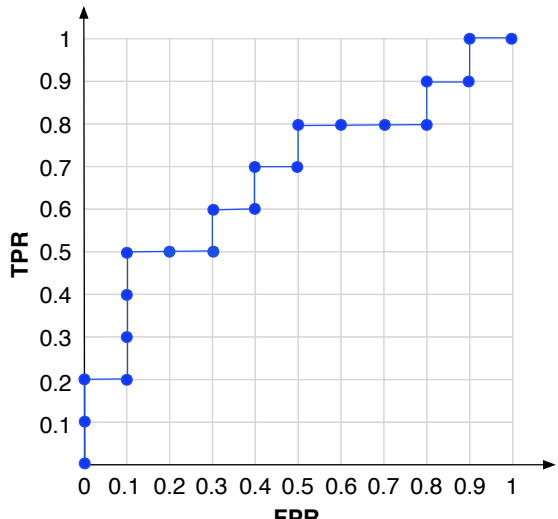
Step 1: Sort the scores in descending order!

Step 2: Start in (0,0)!

Step 3: For each instance, move up when the instance is positive and move to the right when the instance is negative!

Drawing of ROC curve (1,2,3)

inst	score	class	inst	score	class
1	0.90	*p*	11	0.40	*p*
2	0.80	*p*	12	0.39	*n*
3	0.70	*n*	13	0.38	*p*
4	0.65	*p*	14	0.37	*n*
5	0.60	*p*	15	0.36	*n*
6	0.55	*p*	16	0.35	*n*
7	0.54	*n*	17	0.34	*p*
8	0.53	*n*	18	0.33	*n*
9	0.52	*p*	19	0.30	*p*
10	0.51	*n*	20	0.20	*n*



Drawing of ROC curve

☐ score: classification score (prediction) ☐ class: actual class (actual positives or negatives) ☐ cm: confusion matrix. ☐ Threshold of 0.5

inst	score	class	cm	inst	score	class	cm
1	0.90	*p*	TP	11	0.40	*p*	FN
2	0.80	*p*	TP	12	0.39	*n*	TN
3	0.70	*n*	FP	13	0.38	*p*	FN
4	0.65	*p*	TP	14	0.37	*n*	TN
5	0.60	*p*	TP	15	0.36	*n*	TN
6	0.55	*p*	TP	16	0.35	*n*	TN
7	0.54	*n*	FP	17	0.34	*p*	FN
8	0.53	*n*	FP	18	0.33	*n*	TN
9	0.52	*p*	TP	19	0.30	*p*	FN
10	0.51	*n*	FP	20	0.20	*n*	TN

Drawing of ROC curve

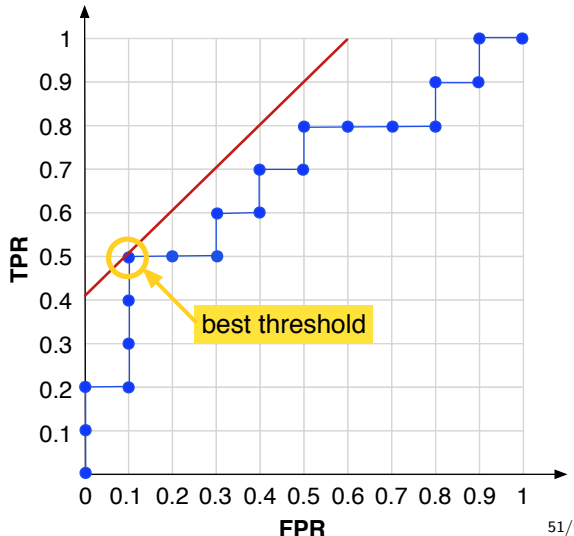
- ☐ Please do not draw a ROC curve based on confusion matrix (another frequent mistake)!

Notes: For calculation of confusion matrix, the threshold should already be given (e.g., threshold of 0.5). The presented confusion matrix is calculated based on the threshold 0.5

Best trade-off of TPR and FPR

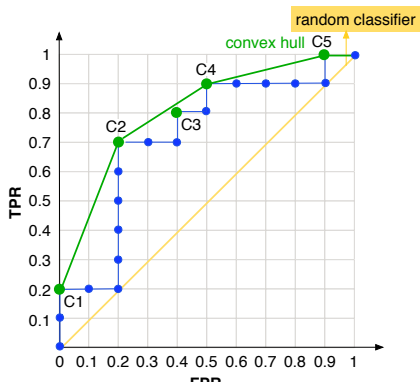
We select the threshold of 0.55 (6th instance) to obtain the best performance (the highest accuracy).

inst	score	class	inst	score	class
1	0.90	*p*	11	0.40	*p*
2	0.80	*p*	12	0.39	*n*
3	0.70	*n*	13	0.38	*p*
4	0.65	*p*	14	0.37	*n*
5	0.60	*p*	15	0.36	*n*
6	0.55	*p*	16	0.35	*n*
7	0.54	*n*	17	0.34	*p*
8	0.53	*n*	18	0.33	*n*
9	0.52	*p*	19	0.30	*p*
10	0.51	*n*	20	0.20	*n*



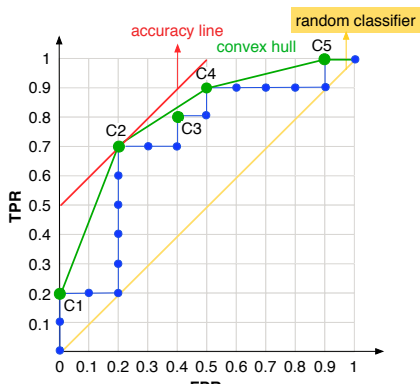
Selection of the best threshold

Classifier under the convex hull: the worst case (C_3)



Selection of the best threshold

Finding of the best classifier (C_3) using accuracy line



Accuracy line

☐ Accuracy line: $\text{TPR} = \frac{neg}{pos} * FPR + \frac{ACC - neg}{pos}$

☐ N: the total number of instances
☐ $N = POS + NEG$
☐ POS: the total number of (actual) positives ($TP + FN$)
☐ NEG: the total number of (actual) negatives ($FP + TN$)
☐ neg: $\frac{NEG}{N}$; pos: $\frac{POS}{N}$

Accuracy line

$$
\begin{aligned}
ACC &= \tfrac{TP+TN}{N} \\
&= \tfrac{TP}{N} + \tfrac{TN}{N} \\
&= (\tfrac{TP}{POS} \cdot \tfrac{POS}{N}) + (\tfrac{NEG-FP}{N}) \\
&= (\tfrac{TP}{POS} \cdot \tfrac{POS}{N}) + (\tfrac{NEG}{N} - (\tfrac{FP}{NEG} \cdot \tfrac{NEG}{N})) \\
&= (TPR \cdot pos) + (neg - (neg \cdot FPR))
\end{aligned}
$$

$$\text{TPR} = \tfrac{ACC-neg}{pos} + \tfrac{neg}{pos} \cdot FPR$$

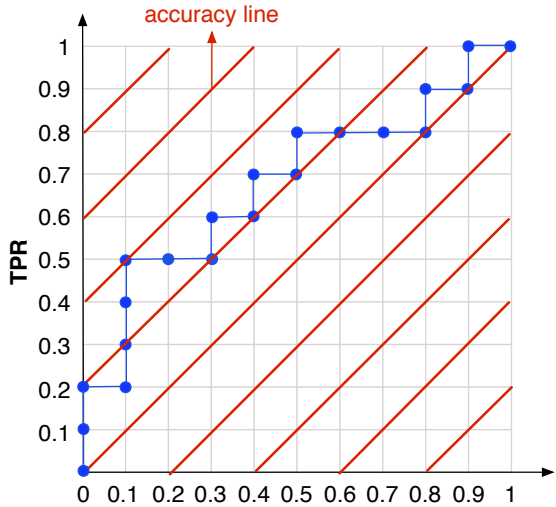
- N: the total number of instances; $N = POS + NEG$
- POS: the total number of (actual) positives ($TP + FN$)
- NEG: the number of (actual) negatives ($FP + TN$)
- neg: $\tfrac{NEG}{N}$; pos: $\tfrac{POS}{N}$

Accuracy line

- TPR $= \frac{neg}{pos} * FPR + \frac{ACC - neg}{pos}$
- y = ax + b, where $y = TPR, a = \frac{neg}{pos}, x = FPR, b = \frac{ACC - neg}{pos}$
- The slope of accuracy line (a) is the ratio of *neg* and *pos* ($\frac{neg}{pos}$).
  - If $\frac{neg}{pos} = 1$, the slope of accuracy line is 1.
  - If $\frac{neg}{pos} = 4$, the slope of accuracy line is 4.

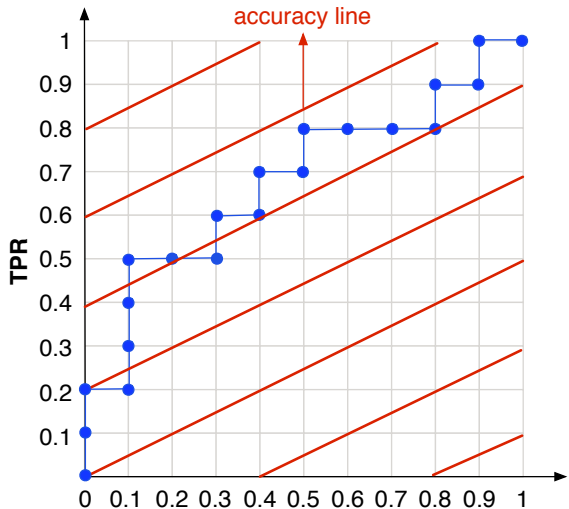
Accuracy line

In case of $\frac{neg}{pos} = 1$, i.e., the slope $a = 1$



Accuracy line

In case of $\frac{neg}{pos} = \frac{1}{2}$, i.e., the slope $a = \frac{1}{2}$



Exercise 5 (ROC)

1. Draw a ROC curve based on the given data!
2. Which threshold should be chosen to obtain the best performance?
3. Calculate the number of TPs, FPs, TNs, and FNs (the given threshold: 0.5)!

inst	score	class	inst	score	class
1	0.95	*p*	11	0.10	*n*
2	0.28	*p*	12	0.59	*p*
3	0.90	*p*	13	0.57	*n*
4	0.62	*n*	14	0.29	*n*
5	0.85	*n*	15	0.56	*p*
6	0.64	*p*	16	0.36	*n*
7	0.53	*p*	17	0.80	*n*
8	0.38	*n*	18	0.43	*n*
9	0.68	*p*	19	0.76	*p*
10	0.44	*n*	20	0.71	*p*

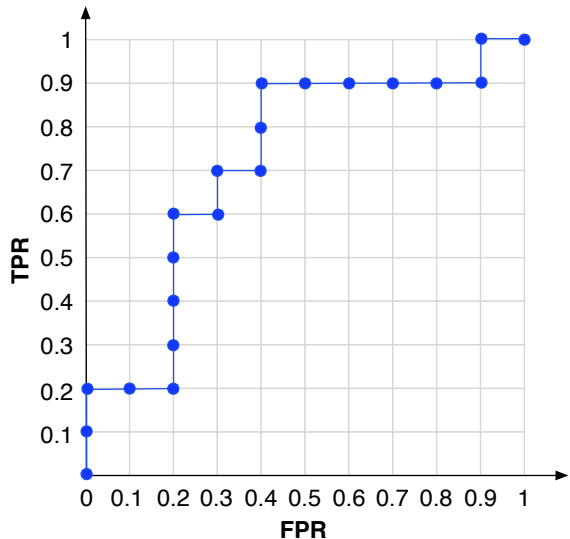
Solution (exercise 5a)

The scores are sorted in descending order.

inst	score	class	inst	score	class
1	0.95	*p*	13	0.57	*n*
3	0.90	*p*	15	0.56	*p*
5	0.85	*n*	7	0.53	*p*
17	0.80	*n*	10	0.44	*n*
19	0.76	*p*	18	0.43	*n*
20	0.71	*p*	8	0.38	*n*
9	0.68	*p*	16	0.36	*n*
6	0.64	*p*	14	0.29	*n*
4	0.62	*n*	2	0.28	*p*
12	0.59	*p*	11	0.10	*n*

Solution (exercise 5b)

inst	score	class	inst	score	class
1	0.95	*p*	11	0.57	*n*
2	0.90	*p*	12	0.56	*p*
3	0.85	*n*	13	0.53	*p*
4	0.80	*n*	14	0.44	*n*
5	0.76	*p*	15	0.43	*n*
6	0.71	*p*	16	0.38	*n*
7	0.68	*p*	17	0.36	*n*
8	0.64	*p*	18	0.29	*n*
9	0.62	*n*	19	0.28	*p*
10	0.59	*p*	20	0.10	*n*

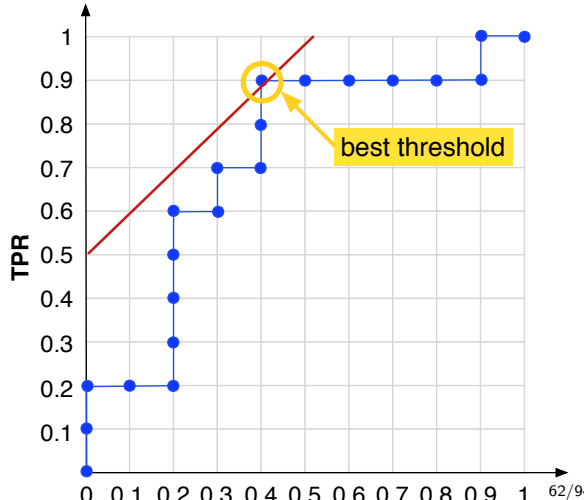


Solution (exercise 5c)

Best threshold: 0.53

$\frac{neg}{pos} = 1$, i.e., the slope $a = 1$

inst	score	class	inst	score	class
1	0.95	*p*	11	0.57	*n*
2	0.90	*p*	12	0.56	*p*
3	0.85	*n*	13	0.53	*p*
4	0.80	*n*	14	0.44	*n*
5	0.76	*p*	15	0.43	*n*
6	0.71	*p*	16	0.38	*n*
7	0.68	*p*	17	0.36	*n*
8	0.64	*p*	18	0.29	*n*
9	0.62	*n*	19	0.28	*p*
10	0.59	*p*	20	0.10	*n*



Solution (exercise 5d)

☐ The threshold of 0.5 is applied.

☐ score: classification scores (predicted class) ☐ class: actual class (actual positives or negatives) ☐ cm: confusion matrix

9 TPs, 5 FPs, 0 FNs, 6 TN

inst	score	class	cm	inst	score	class	cm
1	0.95	*p*	TP	11	0.57	*n*	FP
2	0.90	*p*	TP	12	0.56	*p*	TP
3	0.85	*n*	FP	13	0.53	*p*	TP
4	0.80	*n*	FP	14	0.44	*n*	TN
5	0.76	*p*	TP	15	0.43	*n*	TN
6	0.71	*p*	TP	16	0.38	*n*	TN
7	0.68	*p*	TP	17	0.36	*n*	TN
8	0.64	*p*	TP	18	0.29	*n*	TN
9	0.62	*n*	FP	19	0.28	*p*	FP
10	0.59	*p*	TP	20	0.10	*n*	TN

Evaluation techniques

Limited data

- A large number of data is provided in some research areas, e.g., marketing. However, in some other areas (e.g., medicine), the amounts of data is not sufficient.
- In general, data acquisition is very time-consuming issue, especially data obtained by human.
- Small sample size of data can lead to overfitting.

General methods for limited data

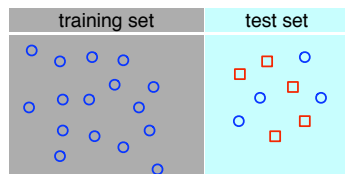
Methods: Split data into training and test set

- Training set: A classifier is trained based on training data.
- Validation set: Parameters of classifier are optimized based on validation data.
- Test set: The trained classifier is evaluated based on test data.

Holdout methods

- 2/3 is randomly chosen for training and 1/3 for testing.

There are no □-instances in the training data.



- Stratification: Each class should be represented in the right proportion in the training and testing set.

Holdout methods

- Stratified holdout method:

  Training and test set are randomly chosen, but the class ratio should be considered (ideally the equal ratio of both classes in the training and test data.

- Repeated holdout method:

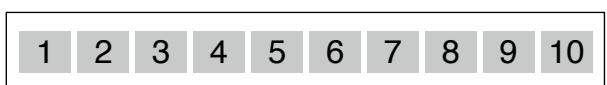
  Repetition of training and testing with different random samples (possibly with stratification) to handle sampling bias.

- A specific form of repeated holdout method is a cross validation.

k-fold cross validation (1)

(1) Data is spilt into k subsets.

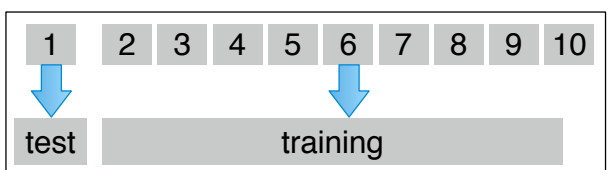
Example: 10-fold cross validation ($k = 10$ in this example)



k-fold cross validation (2)

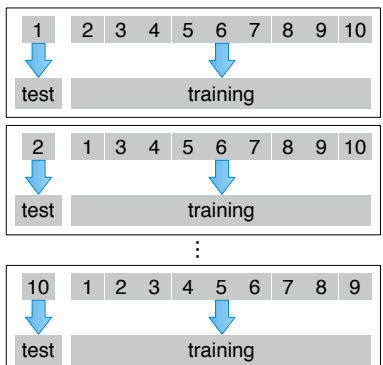
(2) 1 subset is used as test data and k-1 subsets are used as training data (holdout method).

Example: 10-fold cross validation ($k = 10$ in this example)



k-fold cross validation (3)

(3) The holdout method is repeated k times ($k = 10$).



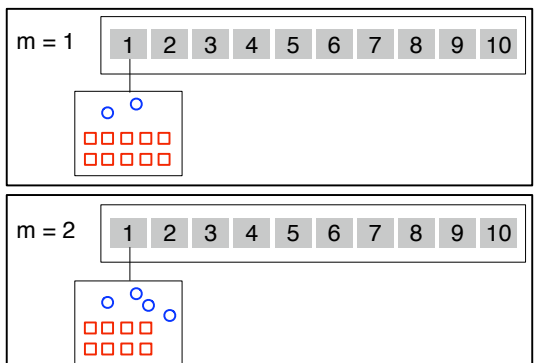
k-fold cross validation (4)

(4) k-fold cross validation is repeated m times.

- $m \times k$-fold cross validation
- Examples: 5×10-fold or 10×10-fold cross validation
- Note: The samples of the first subset in the first iteration is different from the samples of the first subset in the second iteration.

k-fold cross validation (5)

$m \times k$-fold cross validation, e.g., m = 2; k = 10 ($m \times k$ iterations)



Leave-One-Out cross validation (LOOCV)

k-fold cross validation, where $k = N$ (N: number of samples), i.e., the data is not spilt into k folds.

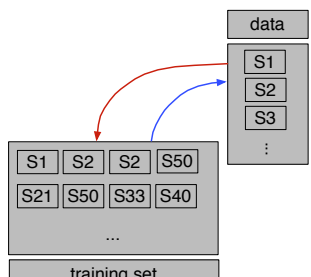
Example: Leave-One-Subject-Out cross validation

- 10 subjects ($N = 10$), Remember: $k = N$
- Data obtained by 9 subjects are used to train a classifier
- Data obtained by 1 subject is used for evaluation of the trained classifier

Bootstrap method

Random sampling with replacement

- Test data: Instances that are not sampled in the training set
- Training data: bootstrapping is repeated m times



Exercise 7 (cross validation)

Exercise is found in the exercise sheet 1.

Implementation of a function that generates train/test data pairs according to the cross-validation method. Integrate stratification, randomization, and repetition in your function and test your implementation on the *IRIS* data set using the template classifier which is found in the script *evaluation.py*.

Statistical analysis (basic concepts)

Independent samples

Datasets of two independent groups (datasets) are collected.

Example: $n = 6$, where n is the size of sample (instance)

n	group 1	group 2
1	x_1	y_1
2	x_2	y_2
3	x_3	y_3
4	x_4	y_4
5	x_5	y_5
6	x_6	y_6

Comparison between two independent groups

(1) Separate computation of mean for each group (μ_x and μ_y)

$$\mu_x = \frac{1}{n}\sum_{i=1}^{n} x_i$$
$$\mu_y = \frac{1}{n}\sum_{i=1}^{n} y_i$$

(2) Separate computation of variance for each group (σ_x^2 and σ_y^2)

$$\sigma_x^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu_x)^2$$
$$\sigma_y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \mu_y)^2$$

Comparison between two independent groups

n	group 1	group 2
1	x_1	y_1
2	x_2	y_2
3	x_3	y_3
4	x_4	y_4
5	x_5	y_5
6	x_6	y_6
mean (μ)	$\frac{1}{n}\sum_{i=1}^{n} x_i$	$\frac{1}{n}\sum_{i=1}^{n} y_i$
variance (σ^2)	$\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\mu_x)^2$	$\frac{1}{n-1}\sum_{i=1}^{n}(y_i-\mu_y)^2$

Comparison between two independent groups

(3) Computation of mean difference

$$\mu_x - \mu_y = \frac{1}{n}\sum_{i=1}^{n} x_i - \frac{1}{n}\sum_{i=1}^{n} y_i$$

(4) Computation of pooled variance

$$\sigma^2 = \frac{(n_x-1)\sigma_x^2+(n_y-1)\sigma_y^2}{n_x+n_y-2}$$

Dependent samples (repeated measure design)

The same data (or person) is repeatedly measured, e.g., data1 and data2 or day1 and day2.

Example: Comparison of two different algorithms.

- ☐ The same data should be used for comparison of two algorithms.
- ☐ Otherwise it makes no sense to use two independent data, since the performance of algorithm is affected by which data set is used (from two independently different data sets)

Comparison in repeated measure design

(1) Computation of difference between paired samples

$$d_i = x_i - y_i.$$

(2) Computation of mean of difference between paired samples

$$\bar{d} = \frac{1}{n} \sum_{i=1}^{n} d_i$$

(3) Computation of variance of difference between paired samples

$$\sigma^2 = \frac{1}{n-1} \sum_{i=1}^{n} (d_i - \bar{d})^2$$

Comparison in repeated measure design

sample	method 1	method 2	diff. between both methods
n	x_i	y_i	$d_i = x_i - y_i$
1	x_1	y_1	$d_1 = x_1 - y_1$
2	x_2	y_2	$d_2 = x_2 - y_2$
3	x_3	y_3	$d_3 = x_3 - y_3$
4	x_4	y_4	$d_4 = x_4 - y_4$
5	x_5	y_5	$d_5 = x_5 - y_5$
6	x_6	y_6	$d_6 = x_6 - y_6$
mean ($\bar{d}$)	–	–	$\frac{1}{n}\sum_{i=1}^{n} d_i$
variance (σ^2)	–	–	$\frac{1}{n-1}\sum_{i=1}^{n}(d_i - \bar{d})^2$

Statistical tests (basics)

- Parametric or non-parametric tests depending on sample distribution (e.g., normal distribution)
  - Parametric tests: t-test, paired t-test, ANOVA, rm ANOVA, etc.
  - Non-parametric tests: Mann-Whiteny Wilcoxon rank-sum test, Wilcoxon sign-rank test, Fridmann test

- Example for parametric tests if levels of independent variable < 2:
  - t-test: $\sqrt{\frac{n_x n_y}{n_x + n_y}} \frac{\mu_x - \mu_y}{\sigma^2}$
  - paired t-test: $t = \sqrt{n} \frac{\bar{d}}{\sigma^2}$

Literature

- Machine Learning. Tom Mitchell, McGraw Hill, 1997
- An introduction to ROC analysis. T. Fawcett, In: Pattern Recognition Letters Vol. 27 (8), p. 861 - 874, Elservier, 2006
- Pattern recognition. Bishop. 1995.
- Data Mining: Practical Machine Learning Tools and Techniques. Ian H. Witten, Eibe Frank, Morgan Kaufmann, 2005

Degree of freedom

- Question in the previous lecture (24th of October):

  Why do variance have $n-1$ degree of freedom (not n)?

- Answer: Only $n-1$ deviations can freely varied.

- Degree of freedom (df): The number of value that can varied for computation.

Degree of freedom

Example:

- 4 Samples ($n = 4$): $x_1 = 1$, $x_2 = 2$, $x_3 = 3$, $x_4 = 4$
- Mean: $\mu_x = 2.5$
- The sum of deviations $(x_i - \mu_i)$ is zero: $\sum_{i=1}^{n}(x_i - \mu_i) = 0$
- Deviations: $\mu_x - x_1 = -1.5$, $\mu_x - x_2 = -0.5$, $\mu_x - x_3 = 1.5$
- That means, $\mu_x - x_4$ should be $= 1.5$, i.e., $\mu_x - x_4 = 1.5$
- Hence, only $n - 1$ can freely varied,
  i.e., variance has n-1 (e.g., $4 - 1$) degree of freedom

Matthews correlation coefficient

- Matthews correlation coefficient (MCC)

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

$$\frac{(\text{correct predictions}) - (\text{incorrect predictions})}{\sqrt{(\text{positive prediction}) \cdot (\text{actual positives}) \cdot (\text{negative prediction}) \cdot (\text{actual negatives})}}$$

- Equivalent to Pearson's correlation coefficient:

  Empirical covariance and empirical variance of each random variable

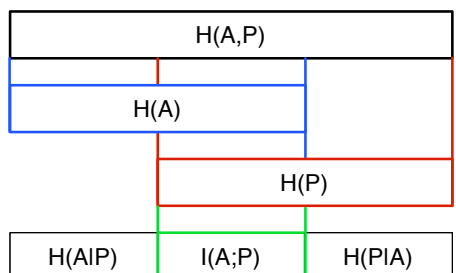
$$r_{x,y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Mutual information based on confusion matrix

Mutual information $I(A;P)$: Difference between a-priory entropy $H(A)$ which is computed based on class ratio and entropy of classification results $H(A|P)$ which is computed based on confusion matrix

$$I(A;P) = H(A) - H(A|P) = \sum_{a \in A} \sum_{g \in P} p(a,g) \log_2 \frac{p(a,g)}{p(a) \cdot p(g)}$$

- c_{ag}: confusion matrix entry
- m: the total number of instances
- $p(a,g) = p(A=a, P=g) = \frac{c_{ag}}{m}$
- $p(a) = p(A=a) = 1/m \sum_g c_{ag}$
- $p(g) = p(P=g) = 1/m \sum_a c_{ag}$



$I(A;P)$ measures a relationship between two random variables.

Exercise 6 (mutual information)

Calculate mutual information based on confusion matrix!

		prediction (classified as)	
		positive	negative
actual label (labelled as)	positive	40	10
	negative	5	45

Solution (exercise 6a)

Step 1: a-priory entropy $H(A)$ based on confusion matrix: $p(a, g) = p(A = a, P = g) = \frac{c_{ag}}{m}$

A: actual label, P: predicted label, m: the total number of instances (N)

- ☐ $P(A = p, P = p) = 40/100$
- ☐ $P(A = p, P = n) = 10/100$
- ☐ $P(A = n, P = p) = 5/100$
- ☐ $P(A = n, P = n) = 45/100$

		prediction (classified as)	
		positive	negative
actual label (labelled as)	positive	40	10
	negative	5	45

Solution (exercise 6a continued)

Step 1: a-priory entropy $H(A)$ based on confusion matrix: $p(a,g) = p(A = a, P = g) = \frac{c_{ag}}{m}$

A: actual label, P: predicted label, m: the total number of instances (N)

- ☐ $P(A = p, P = p) = 40/100 = 0.4$ (TP)
- ☐ $P(A = p, P = n) = 10/100 = 0.1$ (FN)
- ☐ $P(A = n, P = p) = 5/100 = 0.05$ (FP)
- ☐ $P(A = n, P = n) = 45/100 = 0.45$ (TN)

Solution (exercise 6b)

Step 2: actual classes/labels: $p(a) = p(A = a) = 1/m \sum_g c_{ag}$

A: actual label, P: predicted label, m: the total number of instances (N)

- ☐ All actual positives/N: $P(A = p) = (40+10)/100 = 0.5$
- ☐ All actual negatives/N: $P(A = n) = (5+45)/100 = 0.5$

		prediction (classified as)	
		positive	negative
actual label (labelled as)	positive	40	10
	negative	5	45

Solution (exercise 6c)

Step 3: predicted classes/labels: $p(g) = p(P = g) = 1/m \sum_{g} c_{ag}$

A: actual label, P: predicted label, m: the total number of instances (N)

- ☐ All positive predictions: $P(P = p) = (40+5)/100 = 0.45$
- ☐ All negative predictions: $P(P = n) = (10+45)/100 = 0.55$

		prediction (classified as)	
		positive	negative
actual label (labelled as)	positive	40	10
	negative	5	45

Solution (exercise 6d)

$$I(A;P) = H(A) - H(A|P) = \sum_{a \in A} \sum_{g \in P} p(a,g) \log_2 \frac{p(a,g)}{p(a) \cdot p(g)}$$

- ☐ $\mathrm{TP}/\log_2(\mathrm{TP}/((\text{actual positives}) \cdot (\text{positive predictions})))$
- ☐ $\mathrm{FN}/\log_2(\mathrm{FN}/((\text{actual negatives}) \cdot (\text{negative predictions})))$
- ☐ $\mathrm{FP}/\log_2(\mathrm{FP}/((\text{actual positives}) \cdot (\text{positive predictions})))$
- ☐ $\mathrm{TN}/\log_2(\mathrm{FN}/((\text{actual negatives}) \cdot (\text{negative predictions})))$

all actual positives: 0.05, all actual negatives: 0.05, all positive prediction: 0.045, all negative predictions: 0.55, TP: 0.4, FN: 0.1, FP: 0.05, TN: 0.45

$$I(A;P) = 0.4 \log_2(0.4/0.5 * 0.45) + 0.1 \log_2(0.1/(0.5 * 0.55)) + 0.05 \log_2(0.05/(0.5 * 0.45)) + 0.45 \log_2(0.45/(0.5 * 0.55))$$

$$I(A;P) \approx 0.3973$$

Thank You!
Please feel free to ask questions in the forums.