Lecture 3: Model Evaluation

Andreas Wichert
Department of Computer Science and Engineering
Técnico Lisboa

1

Binary Classification

- **Binary** classification problem in Machine Learning (ML)
  - identifying if a certain patient has *some disease* using his *health record*
  - Credit versus No Credit (*using Decision Tree lecture 2*)
  - Class a versus Class b

- ML trained on a traning set D_t tested on a test set D_{test} with

$$\emptyset = D_t \cap D_{test}$$

$$\boldsymbol{accuracy} = \frac{Correctly\ Classified}{All} \qquad \boldsymbol{error\ rate} = 1 - accuracy$$

2

Evaluating classification models

- Some types of mistakes that are worse than others
- We are choosing between two models A and B that diagnose a given infectious disease
  - positive if the disease present, negative if the disease not
  - present both models have the **same accuracy**, which model is better?
- model A's mistakes are all **false positives**
  - cases where the patient is not sick but the model *predicted disease*
- model B where all mistakes are **false negatives**
  - *contagious people are told they are healthy*

3

Confusion Matrix

		true/actual/target: P	true/actual/target: N	
predicted	P	True Positives (TP)	False Positives (FP)	TP+FP
predicted	N	False Negatives (FN)	True Negatives (TN)	FN+TN
		P=TP+FN	N=FP+TN	All=P+N

Recall/sensitivity

% of positive observations predicted as positive

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

Precision

% of positive observations among the observations predicted as positive

$$Precision = \frac{TP}{TP + FP}$$

4

Precision and Recall

- A high recall value without a high precision does not give us any confidence about the quality of the binary classifier
  - High recall value by classifying all patterns as positive (the recall value will be one); however, the precision value will be very low

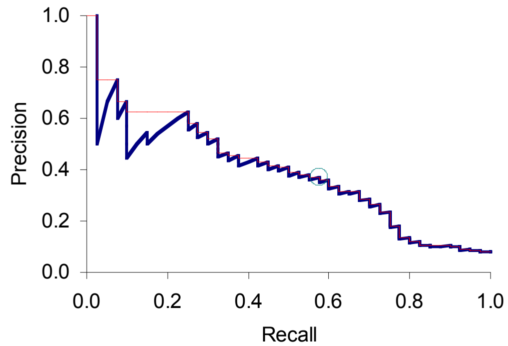
$$Recall = \frac{TP}{P} = \frac{TP}{TP+FN} = \frac{All}{All} = 1$$

  - By classifying only one pattern correctly as positive, we obtain the maximal precision value of one but a low recall value.

$$Precision = \frac{TP}{TP+FP} = \frac{1}{1+0} = 1$$

5

A precision-recall curve



Both values have to be simultaneously interpreted

6

Balanced Measure

- Precision and Recall have to be simultaneously interpreted.
- We can combine both values with the *harmonic mean*

$$F = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

- Both values are evenly weighted.
- This measure is also called the balanced measure.

7

A combined measure: *F*

- Combined measure that assesses this tradeoff is *F* measure (weighted harmonic mean):

$$F = \frac{1}{\alpha \frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R}$$

- However, usually use **balanced F_1 measure**
  - i.e., with β = 1 or α= ½
  - *P=Precision, R=Recall*

8

ROC curve

- For binary classifier indicates the probability of two classes:

  C_1 and ***not C_1*** *positive class:= C_1*

  *p(**C_1**)* and ***not C_1*** *=1- p(**C_1**)* *negative class := not C_1*

  If *p(**C_1**)* $\geq$ *treshold then class **C_1***

  If *p(**C_1**)* < *treshold then class **not C_1***

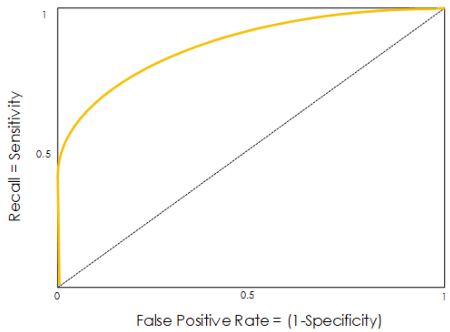
Usually the treshold is 0.5

- *Niave Bayes, Perceptron, Logistic Regression*
  - *introduced later in the course*

9

ROC Curve

Receiver Operating Characteristic



Recall/sensitivity

- % of positive observations predicted as positive

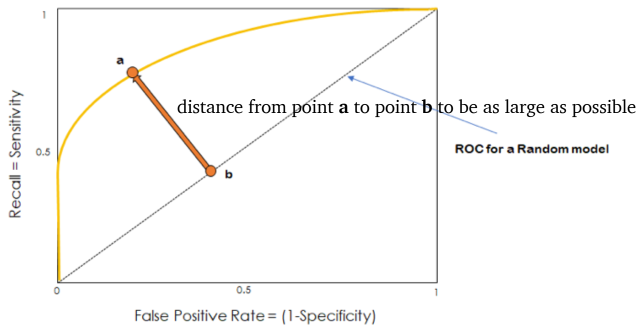
$$Recall = \frac{TP}{P} = \frac{TP}{TP+FN}$$

Fallout/specificity

- % of negative observations predicted as negative

$$False\ Positive\ Rate = Specificity = \frac{TN}{N} = \frac{TN}{TN+FP}$$

10



- To plot the ROC curve, we must first calculate the *Recall* and the $Specificity$ for **various thresholds**, and then plot them against each other
- The further away we are to the curve of the random model, the better

11

Various thresohlds vor ROC curve

- For binary classifier indicates the probability of two classes:

  C₁ and ***not C₁*** — *positive class:= C_1*

  *p(**C₁**)* and ***not C₁*** *=1- p(**C₁**)* — *negative class := not C_1*

  If $p(C_1) \geq$ *treshold* *then class* C_1

  If $p(C_1) <$ *treshold* *then class* ***not*** C_1

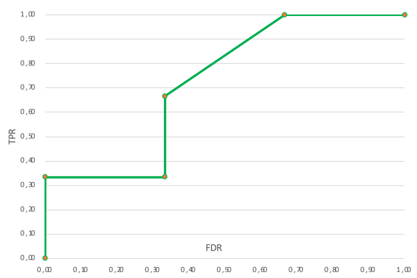
  Usually the treshold is 0.5

- *To compute the ROC curve qe chose* **various thresholds** $\in$ *[0,1]*
- *We chose threshold=0, then threshold=0.1,…, threshold=0.9, threshold=1*

12

z is the true value and $\hat{z}$ the classifier prediction

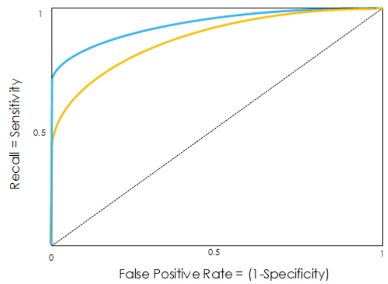
z	$\hat{z}$	0	>0.3	>0.4	>0.45	>0.6	>0.8
1	0.5	TP	TP	TP	FN	FN	FN
1	0.8	TP	TP	TP	TP	TP	FN
1	0.45	TP	TP	FN	FN	FN	FN
0	0.4	FP	FP	TN	TN	TN	TN
0	0.3	FP	TN	TN	TN	TN	TN
0	0.6	FP	FP	FP	FP	TN	TN
FPR=FP/N		1.00	0.67	0.33	0.33	0.00	0.00
TPR=TP/P		1.00	1.00	0.67	0.33	0.33	0.00



13

AUC metric (Area Under the Curve)

- *ACU* quantifies in a **single metric** how well our model classifies the True and False data points.
- *AUC* goes from values of 0.5(random classifier) to 1 (perfect classifier)



14

Lift Charts

- Comparing classifiers:
  - 1,000,000 prospective respondents
  - prediction that 0.1% of **all households** (1,000,000) will respond
  - prediction that 0.4% of a **specified** 100,000 homes will respond.
  - lift factor=increase in response rate=4
  - Given a classifier that outputs *probabilities* for the predicted class value for each test instance, what to do?

15

Lift Factor

sample success proportion=
(number of positive instances in sample) / sample size

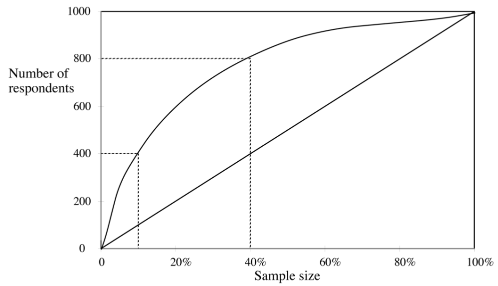
lift factor=
(sample success proportion) / (total test set success proportion)

- Plot the number of respondents as a function of the number of mailings
- Why is response rate dropping with increasing number of mailings?

16

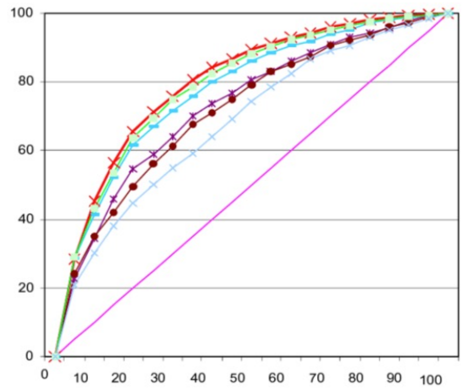
Evaluation of Lift Chart

- Two extreme points:
  - Lower left: if no solicitations are sent - no respondents
  - Upper right: if all households receive offers - 1000 respondents
- What is the ideal point in the chart?
- Best to be in the upper left-hand corner of the chart: mail only to those 1000 (out of a million!) who would respond.



17

Lift charts of two classifiers: which one is better?



18

Evaluating multiclass classifiers

- Most real-world classification problems have more than two classes
  - e.g. identifying risk groups, categorizing documents, recommending products
- Extend binary confusion matrices

		true/actual/target		
		A	B	C
predicted	P	True A (TA)	False A (FA)	False A (FA)
	B	False B (FB)	True B (TB)	False B (FB)
	C	False C (FC)	False C (FC)	True C (TC)

- Accuracy is the % of observations along the diagonal

19

Example of MNIST digits represented by gray images of size 28 × 28

20

predicted class

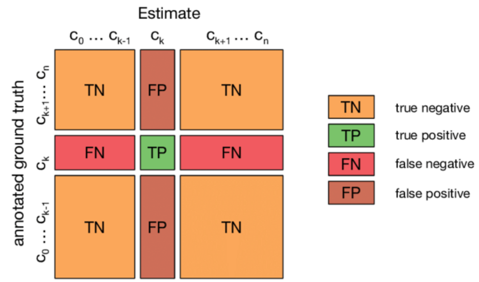
actual class	0	1	2	3	4	5	6	7	8	9	
0	931	0	3	3	0	30	8	2	3	0	980
1	0	1091	3	3	0	3	5	1	29	0	1135
2	16	2	896	14	16	5	19	24	30	10	1032
3	6	1	24	909	0	19	6	17	18	10	1010
4	1	3	6	1	900	2	14	2	15	38	982
5	15	3	8	57	6	728	18	8	39	10	892
6	22	3	10	1	17	19	879	3	4	0	958
7	4	15	28	6	8	0	0	925	4	38	1028
8	11	6	14	29	10	28	11	17	832	16	974
9	12	7	6	12	48	18	0	24	7	875	1009
	1018	1131	998	1035	1005	852	960	1023	981	997	

Confusion Matrix for 10 Classes (Error Matrix)

21

Evaluating multiclass classifiers

- Recall/sensitivity, specificity and precision *per* class
  - the target class is seen as positive
  - the negative class is the **union** of the remaining classes



22

Overfiting

- The training data contains information about the regularities in the mapping from input to output, but it also contains noise
- There is sampling error and a flexible architecture can model the sampling error really well
- However, we cannot tell which regularities are real and which are caused by sampling error

23

- In general, we try to learn a function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$

$$\mathbf{y} = f(\mathbf{x})$$

- that is described by a sample of training data D_t of the labeled data set

$$D_t = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \cdots,, (\mathbf{x}_N, \mathbf{y}_N)\}$$

- Labels can include multiple things like faces vs. non-faces or man-made objects vs. non-man-made objects

24

- After learning, the trained network can be seen as an hypothesis *h* that tries to represent the function *f* and it can be then used for mapping new examples
- The hypothesis *h* should represent the function *f* well on the training set. However, ideally, it should generalize from the training data set to unseen **future data points**.
- To try to make sure this is the case, we can validate on an unseen validation (or test set) data set D_v

$$D_v = \{(\mathbf{x}'_1, \mathbf{y}'_1), (\mathbf{x}'_2, \mathbf{y}'_2), \cdots, , (\mathbf{x}'_M, \mathbf{y}'_M)\} \qquad \emptyset = D_t \cap D_v.$$

25

Mean Squared Error (MSE)

- The validation of the model is done by comparing the hypothesis *h* outputs

$$\mathbf{o}_k = h(\mathbf{x}'_k)$$

- with the correct values y'_k of the validation data set D_v by the mean squared error

$$MSE_{Dv}(h) = \sum_{k=1}^{M} \frac{1}{M} \; \|\mathbf{y}'_k - \mathbf{o}_k\|^2.$$

- The smaller the $MSE(D_v)$ the better the hypothesis *h* describing the function *f*

26

- We can define the mean squared error for the training data set D_t

$$MSE_{Dt}(h) = \frac{1}{N} \cdot \sum_{k=1}^{N} \|\mathbf{y}_k - \mathbf{o}_k\|^2,$$

- usually

$$MSE_{Dv}(h) > MSE_{Dt}(h).$$

27

- If we have two hypothesis h_1 and h_2 with

$$MSE_{Dt}(h_1) < MSE_{Dt}(h_2), \quad MSE_{Dv}(h_1) > MSE_{Dv}(h_2).$$

- then we say that the hypothesis h_1 overfits the training data set D_t
  - h_1 fits better the training examples than $h_2$performs more poorly over examples it didn't learn.
- It seems as if h_1 learned D_t **by heart** and not the topological structure that describes the function f
- h_2 learned the corresponding structure and can **generalize**

28

Cross-Validation

- Estimate the accuracy of a hypothesis induced by a supervised learning algorithm
- Predict the accuracy of a hypothesis over future unseen instances
- Select the optimal hypothesis from a given set of alternative hypotheses
  - Model selection
  - Feature selection
- Combining multiple classifiers (boosting)

29

Holdout Method

- Partition data set $D = \{(v_1,y_1),\ldots,(v_n,y_n)\}$ into *training* D_t and *validation* set $D_h=D\backslash D_t$

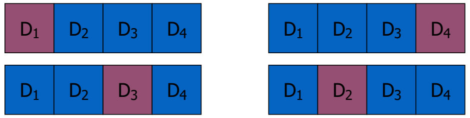
Problems:
- makes insufficient use of data
- training and validation set are correlated

30

Cross-Validation

- k-fold cross-validation splits the data set D into k mutually exclusive subsets $D_1, D_2, \ldots, D_k$

- Train and test the learning algorithm k times, each time it is trained on $D \setminus D_i$ and tested on D_i

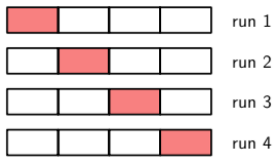


31

Cross-Validation

- Uses all the data for training and testing
- Complete *k*-fold cross-validation splits the dataset of size *m* in all (m over *m/k*) possible ways (choosing *m/k* instances out of *m*)
- Leave *n*-out cross-validation sets *n* instances aside for testing and uses the remaining ones for training (leave one-out is equivalent to *n*-fold cross-validation)
- In stratified cross-validation, the folds are stratified so that they contain approximately the same proportion of labels as the original data set

32



- One major drawback of cross-validation is that the number of training runs that must be performed is increased by a factor of *k*
- How to Evaluate cross-validation for different models (h_1, h_2, h_3)?
  - We will use t-statistics

33

The logic of hypothesis testing

- Example: toss a coin ten times, observe eight heads. Is the coin fair (i.e., what is it's long run behavior?) and what is your residual uncertainty?
- You say, "If the coin were fair, then eight or more heads is pretty unlikely, so I think the coin isn't fair."
- Like proof by contradiction: Assert the opposite (the coin is fair) show that the sample result (≥ 8 heads) has low probability *p*, **reject** the assertion, with residual uncertainty related to *p*.
- Estimate p with a *sampling distribution*.

34

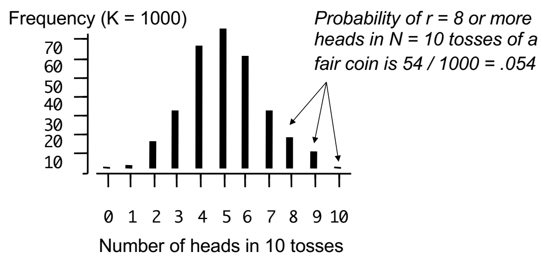
Probability of a sample result under a null hypothesis

- If the coin were fair (p= .5, the *null hypothesis*) what is the probability distribution of r, the number of heads, obtained in *N* tosses of a fair coin? Get it analytically or estimate it by simulation (on a computer):

- Loop K times
  - r := 0 // r is num.heads in N tosses
  - Loop N times // simulate the tosses
    - Generate a random $0 \leq x \leq 1.0$
    - If x >= p increment r // p is the probability of a head
  - Push r onto sampling_distribution
- Print sampling_distribution

35

Sampling distributions



The estimation is constructed by *Monte Carlo sampling*.

36

The t test

- Same logic as the Z test, but appropriate when **population standard deviation** is unknown, samples are small, etc.
- Sampling distribution is t, not normal, but approaches normal as samples size increases
- Test statistic has very similar form but probabilities of the test statistic are obtained by consulting tables of the t distribution, not the normal

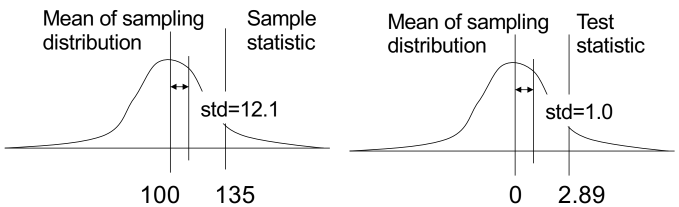
37

The t test

Suppose N = 5 students have mean IQ = 135, std = 27

Estimate the standard deviation of sampling distribution using the sample standard deviation

$$t = \frac{\bar{x} - \mu}{\frac{s}{\sqrt{N}}} = \frac{135 - 100}{\frac{27}{\sqrt{5}}} = \frac{35}{12.1} = 2.89$$



38

p Values

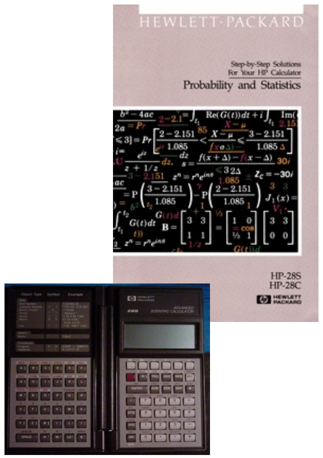
- We find the probabilities by looking them up in tables, or statistics packages provide them
  - The probability of obtaining a particular sample given the null hypothesis is called the *p* value
- Commonly we reject the H0 when the probability of obtaining a *sample statistic* given the null hypothesis is low, say $p < 0.05$
- The null hypothesis is rejected but might be true

39

Paired Sample t Test

- Given a set of paired observations
  - *(from two normal populations)*

A	B	δ=A-B
x1	y1	x1-x2
x2	y2	x2-y2
x3	y3	x3-y3
x4	y4	x4-y4
x5	y5	x5-y5



40

- Calculate the mean $\overline{x}_\delta$ and the standard deviation s_δ of the the differences δ
- H0: μ_δ*=0 (no difference)*
- H0: μ_δ*=k (difference is a constant)*

$$t_\delta = \frac{\overline{x}_\delta - \mu_\delta}{\hat{\sigma}_\delta} \qquad \hat{\sigma}_\delta = \frac{s_\delta}{\sqrt{N_\delta}}$$

41

Paired sample t Test

- We have two rows of data

  94, 86, 12, 90, 66, 40

  10, 20, 22, 26, 6, 18
- Are the two rows significantly different?

δ: *84, 66, -10, 64, 60, 22*

$$\frac{47.6667}{34.8119/\sqrt{6}} = 3.3540$$

- For five degrees of freedom in t-student table between p=0.01 and p=0.02, which is less then 0.05, for this reason we have to reject H0! The two rows are significantly different!

42

Paired sample t-test

Partition Index	Partition Size	Test	Measure	Value
1	125	Classification	True Positive	79
2	125	Classification	True Positive	79
3	125	Classification	True Positive	72
4	125	Classification	True Positive	80
5	125	Classification	True Positive	75
6	125	Classification	True Positive	81
7	125	Classification	True Positive	64
8	125	Classification	True Positive	72
classifier *A*			Average	75.25
			Standard Deviation	5.3794

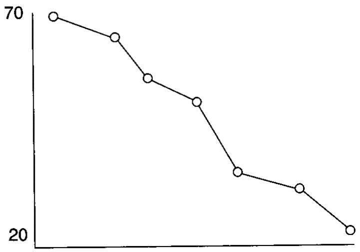
Partition Index	Partition Size	Test	Measure	Value
1	125	Classification	True Positive	75
2	125	Classification	True Positive	73
3	125	Classification	True Positive	80
4	125	Classification	True Positive	71
5	125	Classification	True Positive	75
6	125	Classification	True Positive	80
7	125	Classification	True Positive	67
8	125	Classification	True Positive	77
classifier *B*			Average	74.75
			Standard Deviation	4.1458

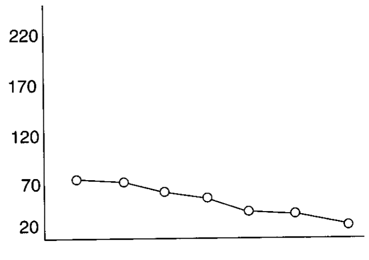
Partition Index	Partition Size	Test	Measure	Value
1	125	Classification	True Positive	63
2	125	Classification	True Positive	55
3	125	Classification	True Positive	70
4	125	Classification	True Positive	58
5	125	Classification	True Positive	67
6	125	Classification	True Positive	70
7	125	Classification	True Positive	55
8	125	Classification	True Positive	61
classifier *C*			Average	62.375
			Standard Deviation	5.7866

using Crosss Validation, determine if the classifier *A, B, C* are significanly different
Compare *(A,B), (A,C), (B,C)*

43

Confidence Intervals





- Just looking at a figure representing the mean values, we can not see if the differences are significant

44

Confidence Intervals (σ known)

- Standard error from the standard deviation

$$\sigma_{\bar{x}} = \frac{\sigma_{Population}}{\sqrt{N}}$$

- 95 Percent confidence interval for normal distribution is about the mean

$$\bar{x} \pm 1.96 \cdot \sigma_{\bar{x}}$$

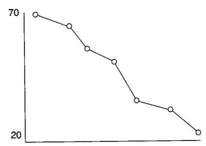
45

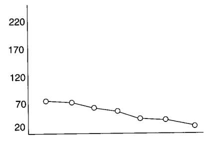
Confidence interval when (σ unknown)

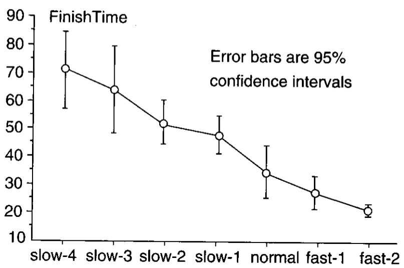
- Standard error from the sample standard deviation
- 95 Percent confidence interval for t distribution ($t_{0.025}$ from a table) is

$$\bar{x} \pm t_{0.025} \cdot \hat{\sigma}_{\bar{x}}$$

$$\hat{\sigma}_{\bar{x}} = \frac{s}{\sqrt{N}}$$

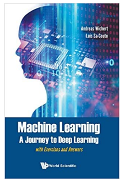






46

Literature



- Machine Learning - A Journey to Deep Learning, A. Wichert, Luis Sa-Couto, World Scientific, 2021
  - Chapter 8

47