Statistisches Data Mining (StDM)
Woche 7

Zurich University
of Applied Sciences

Oliver Dürr

Institut für Datenanalyse und Prozessdesign

Zürcher Hochschule für Angewandte Wissenschaften

oliver.duerr@zhaw.ch

Winterthur, 1 November 2016

No laptops, no phones, no problems

Multitasking senkt Lerneffizienz:

- **Keine Laptops im Theorie-Unterricht Deckel zu oder fast zu (Sleep modus)**

Don't forget:
ZP nächste Woche

Prüfungsdauer: 60 Minuten (nach Einlesen der Daten)
Erlaubte Hilfsmittel

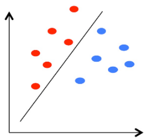
4 Blätter beliebige Zusammenfassung (beidseitig beschrieben)

Kommentiertes R-Skript beliebigen Inhalts, das in R-Studio geöffnet werden kann

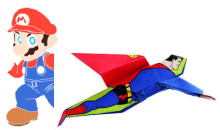
R, R-Studio, Taschenrechner

Overview of classification (until the end to the semester)

Classifiers



K-Nearest-Neighbors (KNN)
Logistic Regression
Linear discriminant analysis
Classification Trees
Support Vector Machine (SVM)
Neural networks NN
Deep Neural Networks (e.g. CNN, RNN)
…



Combining classifiers

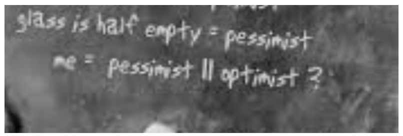
Bagging
Boosting
Random Forest

Evaluation

Cross validation
Performance measures
ROC Analysis / Lift Charts

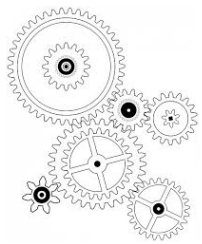
Theoretical Guidance / General Ideas

Bayes Classifier
Bias Variance Trade off (Overfitting)



Feature Engineering

Feature Extraction
Feature Selection



Principal Idea: Classification

Klassifikatoren
•Neuronale Netze
•Entscheidungsbäum
•...

Training Data

id	Type	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
1	setosa	5.1	3.5	1.4	0.2
2	setosa	4.9	3	1.4	0.2
3	virinica	3.3	3.2	1.6	0.5
4	setosa	5.1	3.5	1.4	0.2
...	...	...	...	...	...
150	virinica	4.9	3	1.4	0.2

Learn a classifier

Classifier

Predict

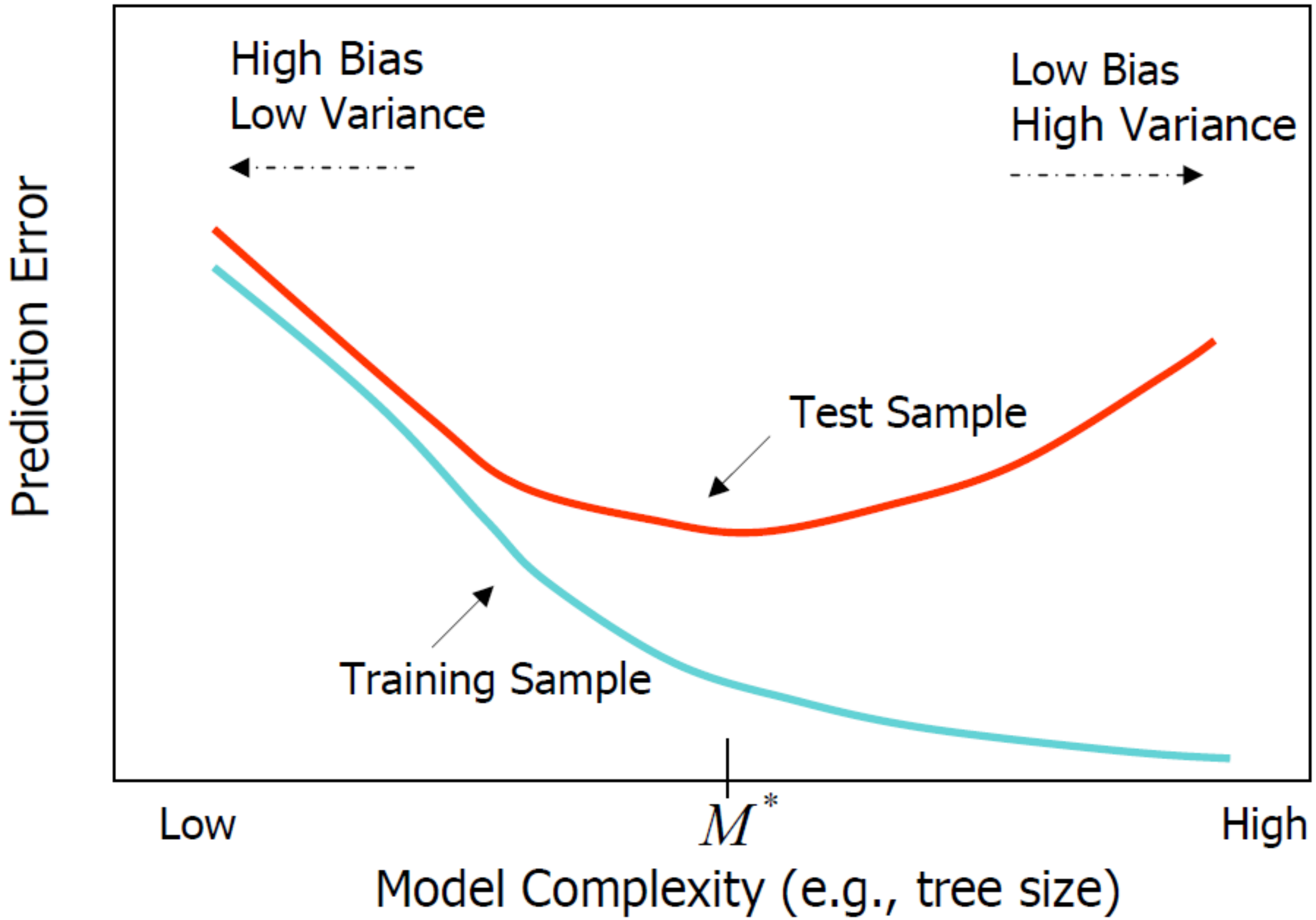
Unknown data / Test data

d	Type	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
1	?	3.1	3.5	1.4	0.2
2	?	4.9	3	1.4	0.2
3	?	3.3	3.2	1.6	0.5
4	?	5.1	3.5	31.4	0.2

Classifier → Type

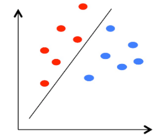
Note:
To evaluate the performance a part of the labelled data not used to train the classifier but left aside to check the performance of the classifier to new data.

What is the right level of complexity



1/k

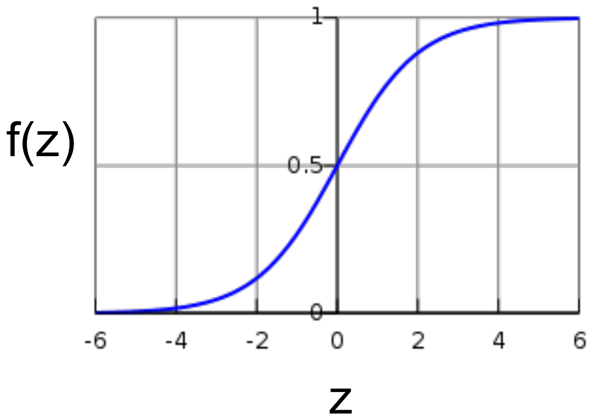
Logistic regression for 2 classes



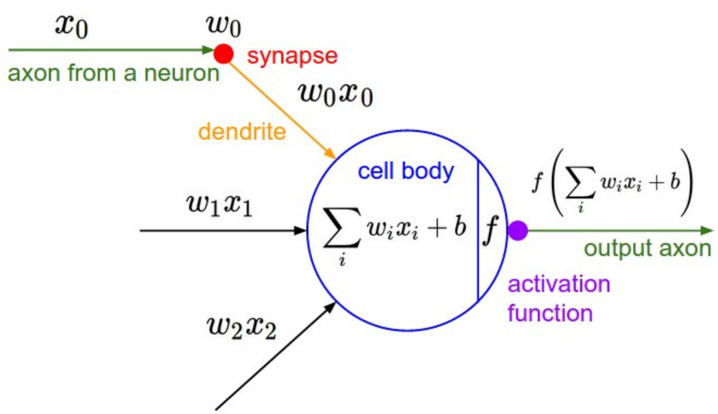
$$z = \beta_0 + x_1\beta_1 + x_2\beta_2 = \beta^T x \quad \in [-\infty, +\infty]$$

$$p_1(z) = P(Y = 1 \mid X = x) = \underbrace{\frac{1}{1+\mathrm{e}^{-z}}}_{\text{f(z)}} \quad \in [0,1]$$

$$p_0(z) = 1 - p_1(z)$$



Symbolic form (taken from [CS231n]())

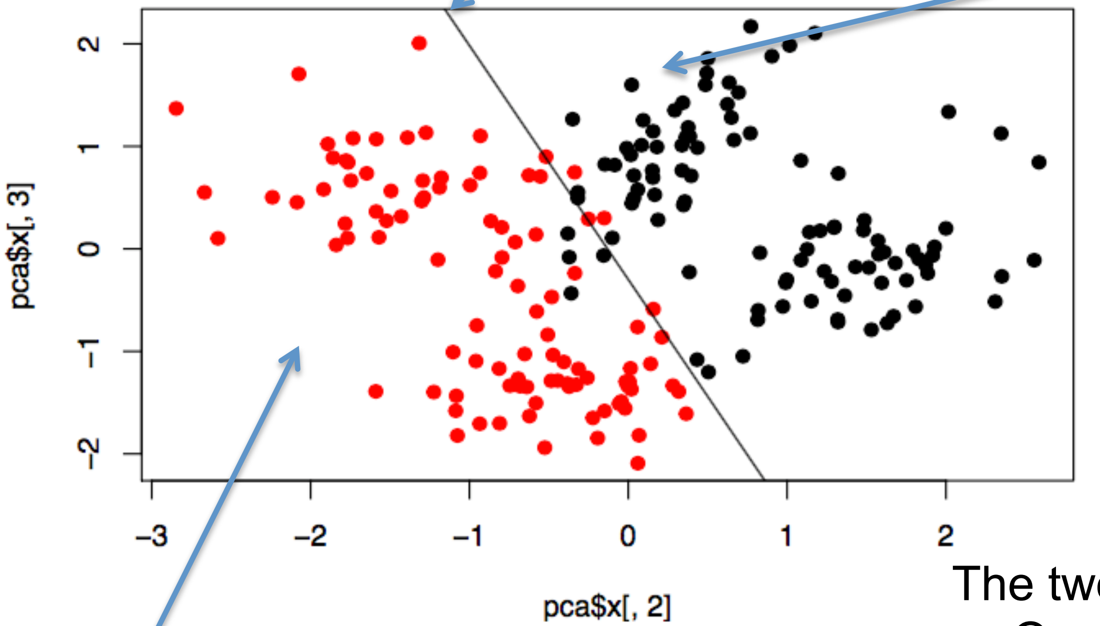


Logistic regression is basic unit for neural networks (see later)

Logistic Regression (2 Classes):
Aufgabe 2 "Nochmals Krabben (Decission Surface)"

At this line $p_1(x_1,x_2)=p_0(x_1,x_2)$

A novel data point on this side would be put to the **black class**



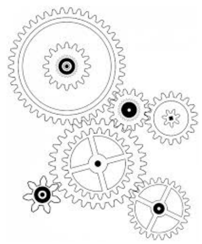
A novel data point on this side would be put to the **red class**

The two classes are separated by:

- One feature: a point
- Two features: a line
- Three features: a plane
- Four features: a hyperplane

Nothing curved ➔Linear Classifiers

Normalisierung / Scaling

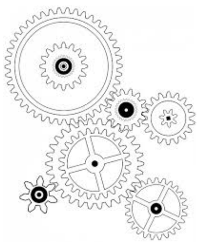


- Unterschiedliche Werte Bereiche
- Daten können Einheiten tragen

Person	Körper Gewicht [kg]	Hirngewicht [g]	Schuhgrösse	Körper Länge [cm]
1	75.1	1400	42	192
2	84.9	2029	47	189
...	...	...	...	
150	50	1780	39	173

- Beliebte Normierungen:
  - Z-Normierung: Danach einheitslos, MW = 0, stddev = 1 (R: scale)
  - Quantil-Normalisierung: Alle Quantile der Verteilung gleich

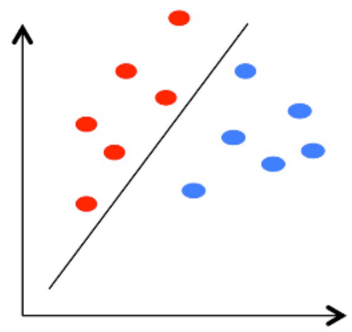
Feature engineering: Categorical Features



- Example green, blue, red how to code?

```
> ############
> # Kategorical Variables
> y = c(0,0,0,1,1,1)
> x = c(0,1,2,0,1,2)
> fit = glm(y ~ x)
> model.matrix(fit)
  (Intercept) x
1           1 0
2           1 1
3           1 2
4           1 0
5           1 1
6           1 2
attr(,"assign")
[1] 0 1
>
> fit = glm(y ~ as.factor(x))
> model.matrix(fit)
  (Intercept) as.factor(x)1 as.factor(x)2
1           1             0             0
2           1             1             0
3           1             0             1
4           1             0             0
5           1             1             0
6           1             0             1
```

Ende der Wiederholung



Linear Discriminant Analysis

Book: ISLR 4.4
Videos (from ISLR)
Linear Discriminant Analysis and Bayes Theorem (7:12)
Univariate Linear Discriminant Analysis (7:37)
Multivariate Linear Discriminant Analysis and ROC Curves (17:42)

Skript: Decision Theory (in more detail) und 6.A (allerdings andere Ableitung)

LDA

- Principle:
  - Model distribution of X in each of the classes (we assume for the time being that we know all the parameters)
  - Use Bayes theorem to get Y from X (see also decision theory)
  - Here we use Gaussians
- Bayes Theorem

$$\Pr(Y=k|X=x) = \frac{\Pr(X=x|Y=k)\cdot\Pr(Y=k)}{\Pr(X=x)}$$

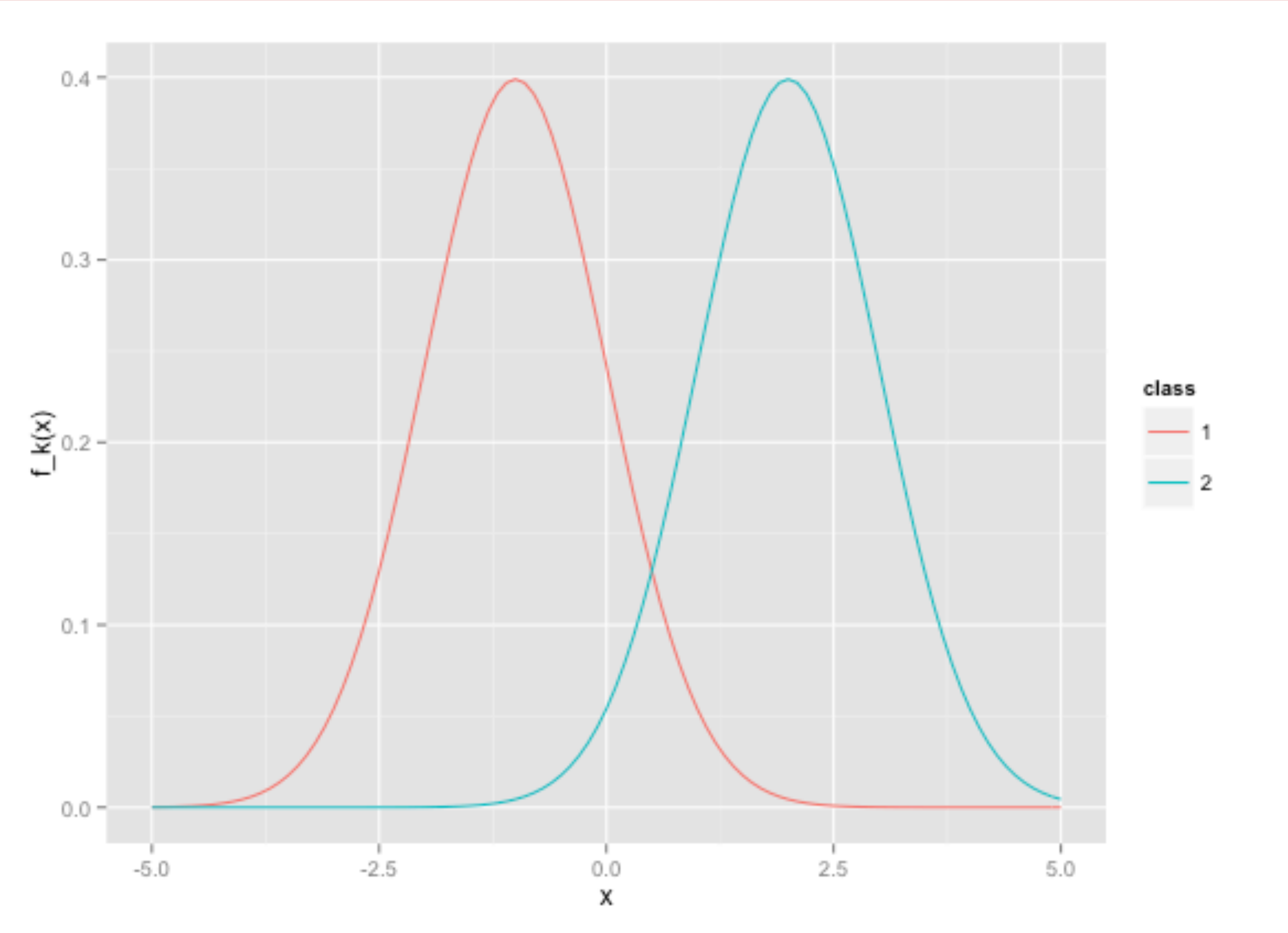
- For Classification

$$\Pr(Y=k|X=x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K}\pi_l f_l(x)}$$

$f_k(x) = \Pr(X=x|Y=k)$ is the *density* for X in class k

$\pi_k = \Pr(Y=k)$ is the marginal or *prior* probability for class k.

Decision Boundary



Falls $\pi_1 = \pi_2 = 0.5$

An welcher Stelle würden Sie x zur Klasse 1 oder 2 rechnen. Wo ist die Grenze?
Wohin verschiebt sich die Grenze wenn $\pi_1 = 0.9$ $\pi_2 = 0.1$?

Discriminant functions

SLDM vermutlich Woche 9

LDA nach Bayes

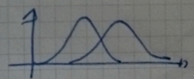
Likelihood Daten X fallt in Klasse Y

$$P(Y=k \mid X=x) = \frac{P(X=x \mid Y=k)\; P(Y=k)}{P(X=x)}$$

Prior W'keit

$$P(Y=k \mid x) = \frac{f_k(x)\,\pi_k}{\sum f_k(x)\,\pi_k}$$

Bsp f_k ist Normalverteilung

$\Rightarrow$ Intuit Beispiel mit  $\Leftarrow$

Rechnung

Betrachten $f_k(x) = \frac{1}{\sqrt{2\pi}\,\sigma_k} e^{-\frac{(x-\mu_k)^2}{2\sigma_k^2}}$

$k = 1 \ldots K$

Nehmen an $\sigma_k = \sigma$ d.h. nicht von Klasse abhängig.

$$\Rightarrow P(Y=k \mid x) = \frac{e^{-\frac{(x-\mu_k)^2}{2\sigma^2}}\,\pi_k}{\sum e^{-(x-\mu_k)^2/2\sigma^2}}$$

Wir müssen nur maximum finden d.h. Log OK

$$\tilde{\delta}_k = -\frac{(x-\mu_k)^2}{2\sigma^2} + \ln \pi_k$$

nur Terme $\sim k$

$$\Rightarrow \delta_k = +\frac{x\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \ln \pi_k$$

↳ Discriminant Score

-LDA 1-

Spezialfall $\pi_k = 1/2$ $k=2$

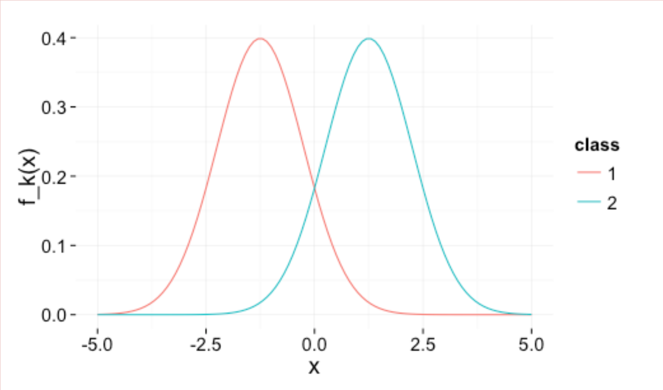
$\tilde{\delta}$: $-(x-\mu_1)^2 = -(x-\mu_2)^2$

$\Rightarrow$ ① $x - \mu_1 = x - \mu_2$

② $x - \mu_1 = -x + \mu_2$ ✓ Auch LSG

$\Rightarrow x = (\mu_1 + \mu_2)/2$

Accuracy of the Bayes Classifier

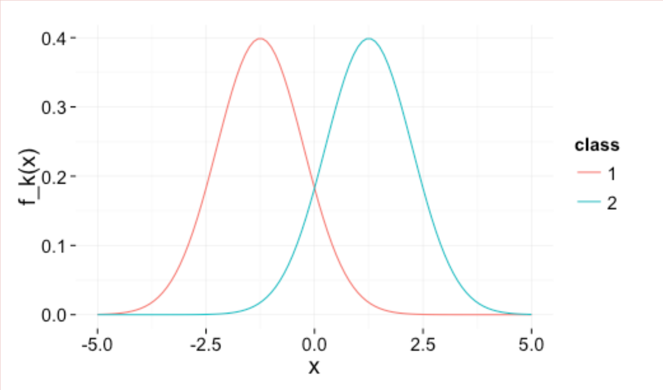


Es gilt:
```
> pnorm(0, -1.25, 1)
[1] 0.8943502
```

	PREDICTED CLASS		
ACTUAL CLASS		Class=1	Class=2
	Class=1		
	Class=2		

Accuracy of the Bayes Classifier

Lösung

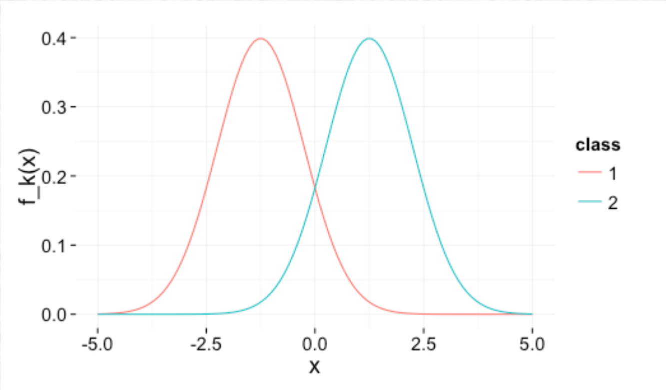


Es gilt:

```
> pnorm(0, -1.25, 1)
[1] 0.8943502
```

	PREDICTED CLASS		
ACTUAL CLASS		Class=1	Class=2
	Class=1	89%	11%
	Class=2	11%	89%

What to do if one only has data



$$\hat{\mu}_k =$$

$$\hat{\sigma}^2 =$$

$$\hat{\pi}_k =$$

Use Training Data set for Estimation

- Usually we don't have the parameters, we have to estimate them.
- The mean μ_k could be estimated by the average of all training observations from the k^{th} class.
- The variance σ^2 could be estimated as the weighted average of variances of all k classes.
- And, π_k is estimated as the proportion of the training observations that belong to the k^{th} class.

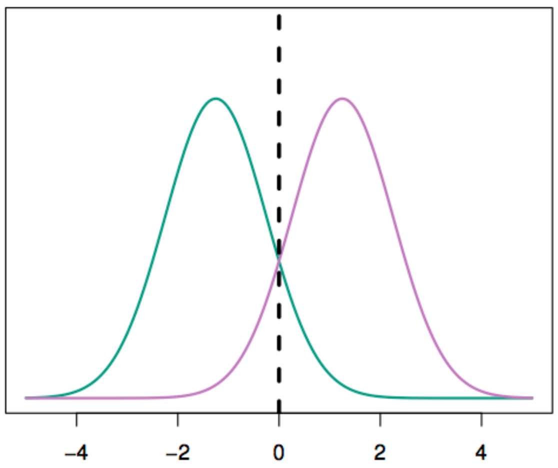
$$
\begin{aligned}
\hat{\mu}_k &= \frac{1}{n_k}\sum_{i:y_i=k} x_i \\
\hat{\sigma}^2 &= \frac{1}{n-K}\sum_{k=1}^{K}\sum_{i:y_i=k}(x_i-\hat{\mu}_k)^2
\end{aligned}
$$

$$\hat{\pi}_k = n_k/n.$$

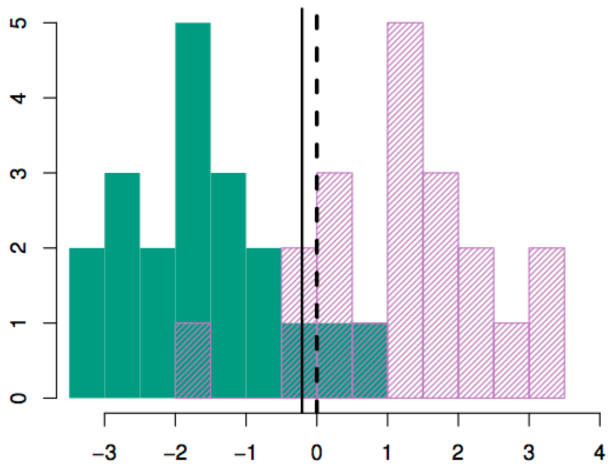
Example

- The dashed vertical line is the Bayes' decision boundary
- The solid vertical line is the LDA decision boundary
  - Bayes' error rate: 10.6%
  - LDA error rate: 11.1%

Theoretical Curve (Bayes Classifier)



20 randomly drawn from each distribution



LDA reaches Bayes for n→∞

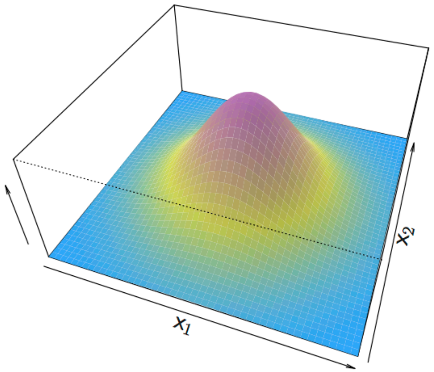
Siehe auch Praktikum für Simulationsstudie

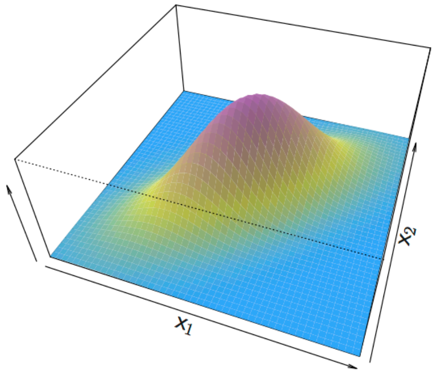
LDA for p>1

- If X is multidimensional ($p > 1$), we use exactly the same approach except the density function *f(x)* is modeled using the multivariate normal density

No correlations

$$\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$





Correlations between x_1 and x_2

$$\Sigma = \begin{pmatrix} 1 & 0.7 \\ 0.7 & 1 \end{pmatrix}$$

$$\text{Density: } f(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \mathbf{\Sigma}^{-1}(x-\mu)}$$

$$\text{Discriminant function: } \delta_k(x) = x^T \mathbf{\Sigma}^{-1}\mu_k - \frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k + \log \pi_k$$

Σ not class depended

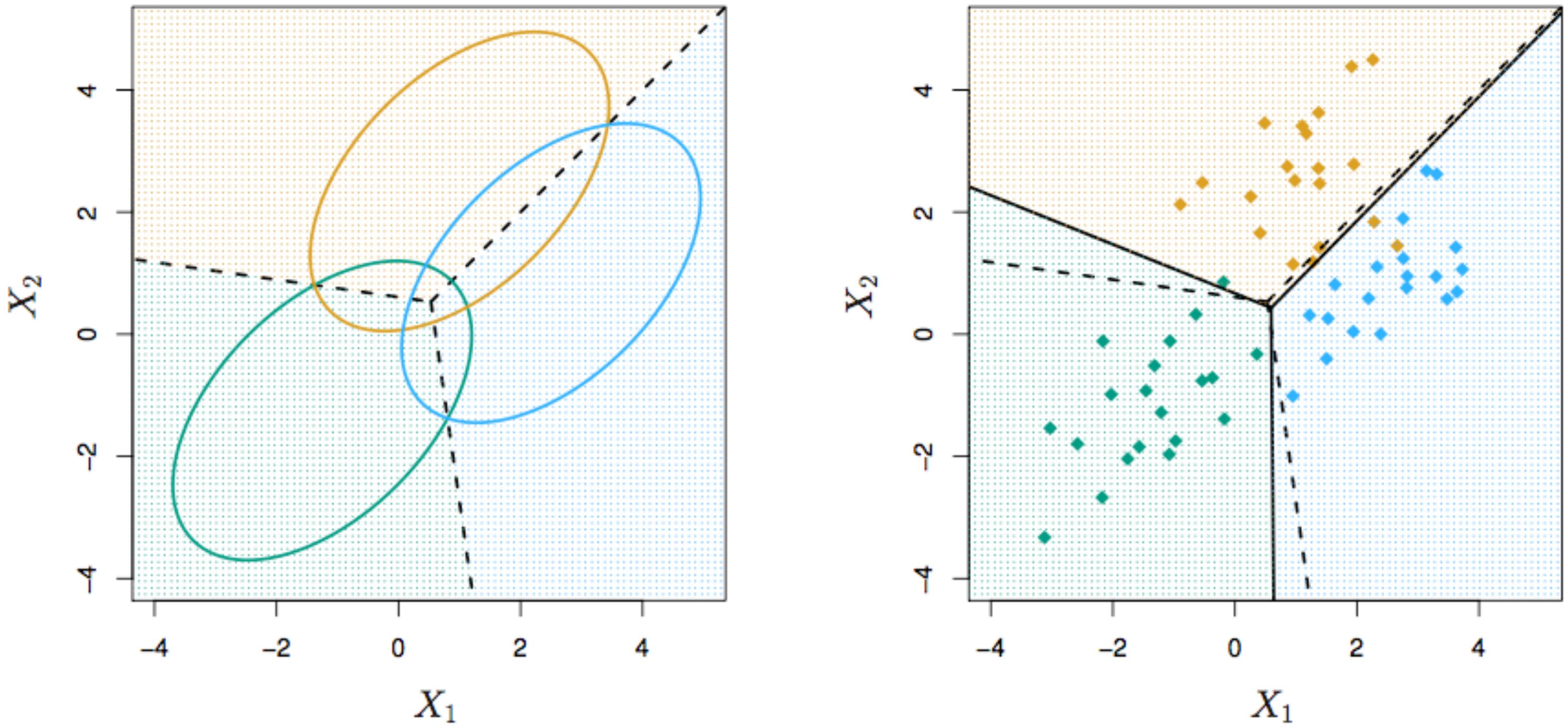
Despite its complex form,

$\delta_k(x) = c_{k0} + c_{k1}x_1 + c_{k2}x_2 + \ldots + c_{kp}x_p$ — a linear function.

p=1

$$\delta_k = + \frac{x\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \ln \pi_k$$

Example: p = 2 K = 3



Here $\pi_1 = \pi_2 = \pi_3 = 1/3$.
The dashed lines are known as the *Bayes decision boundaries*.
Were they known, they would yield the fewest misclassification errors, among all possible classifiers.

Example LDA: Predicting Species

Iris Setosa

…

Iris Virginica

…

Iris Versicolor

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
61	5.0	2.0	3.5	1.0	versicolor
63	6.0	2.2	4.0	1.0	versicolor
69	6.2	2.2	4.5	1.5	versicolor
120	6.0	2.2	5.0	1.5	virginica
42	4.5	2.3	1.3	0.3	setosa
94	5.0	2.3	3.3	1.0	versicolor
54	5.5	2.3	4.0	1.3	versicolor

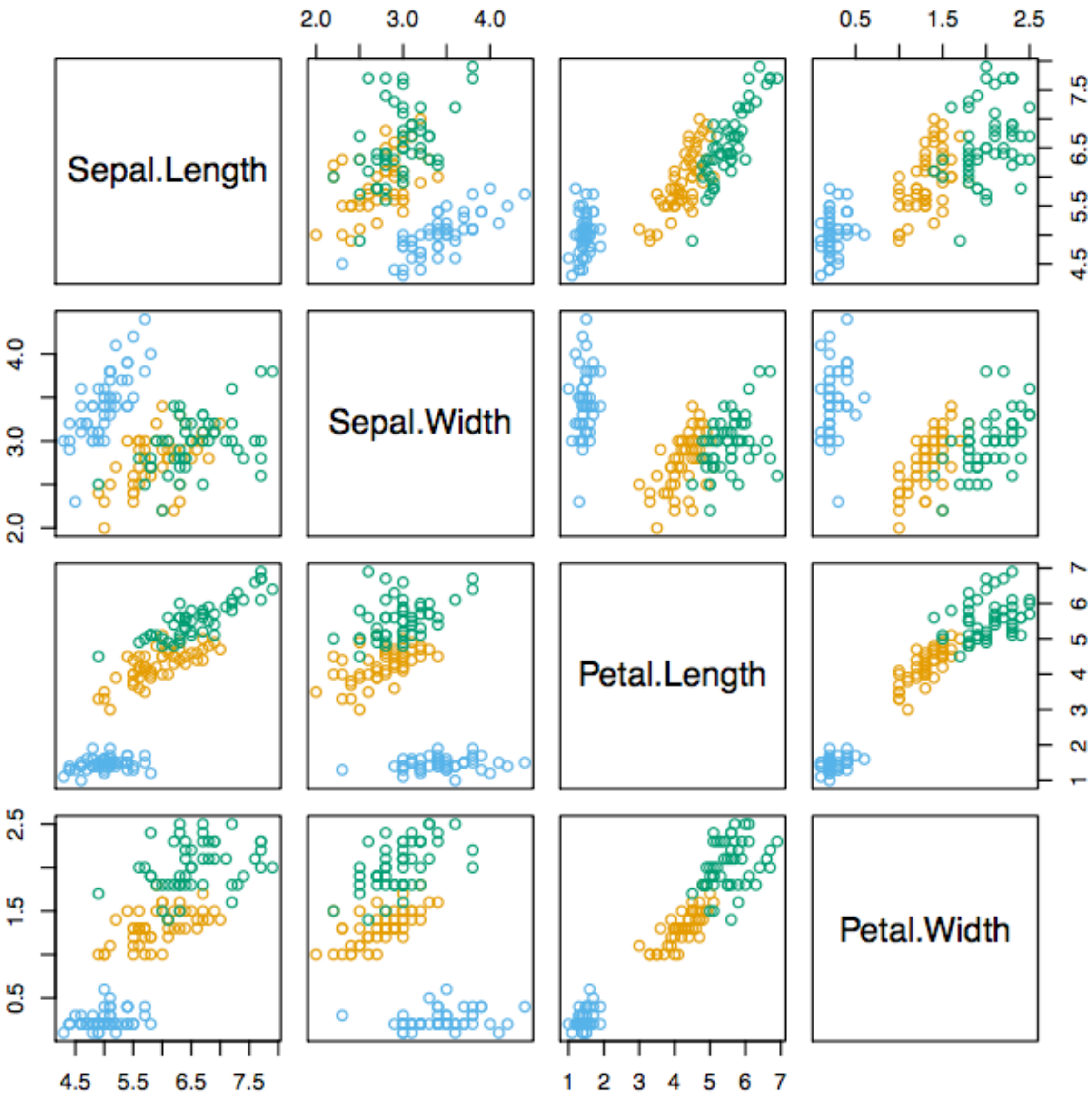
Iris Data Set: Fisher 1936. In R iris

Example Iris

4 variables
3 species
50 samples/class

- Setosa
- Versicolor
- Virginica

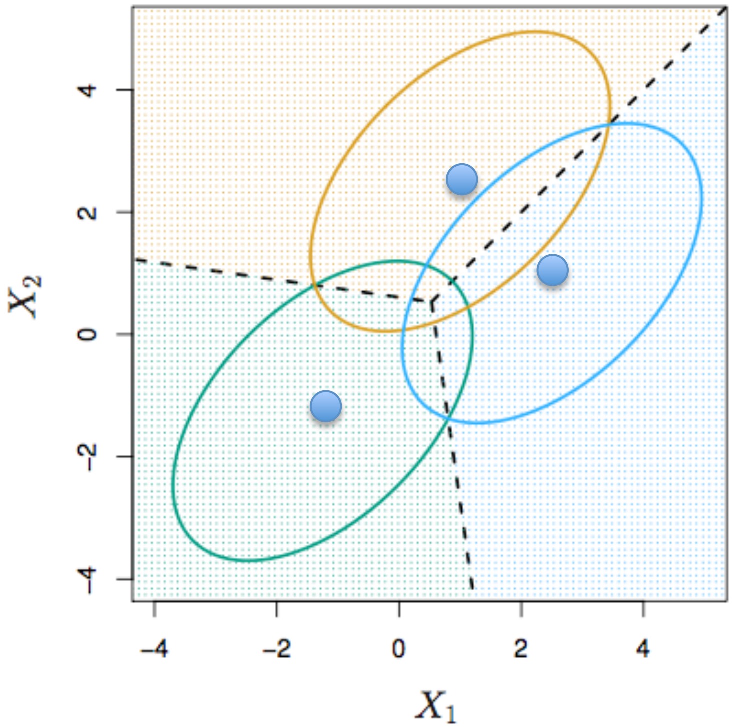
LDA classifies all but 3 of the 150 training samples correctly.



Example Iris

```
> fit = lda(Species ~ ., data = iris)
> res = predict(fit, iris)
> sum(res$class == iris$Species)
[1] 147
```

LDA as dimension reduction



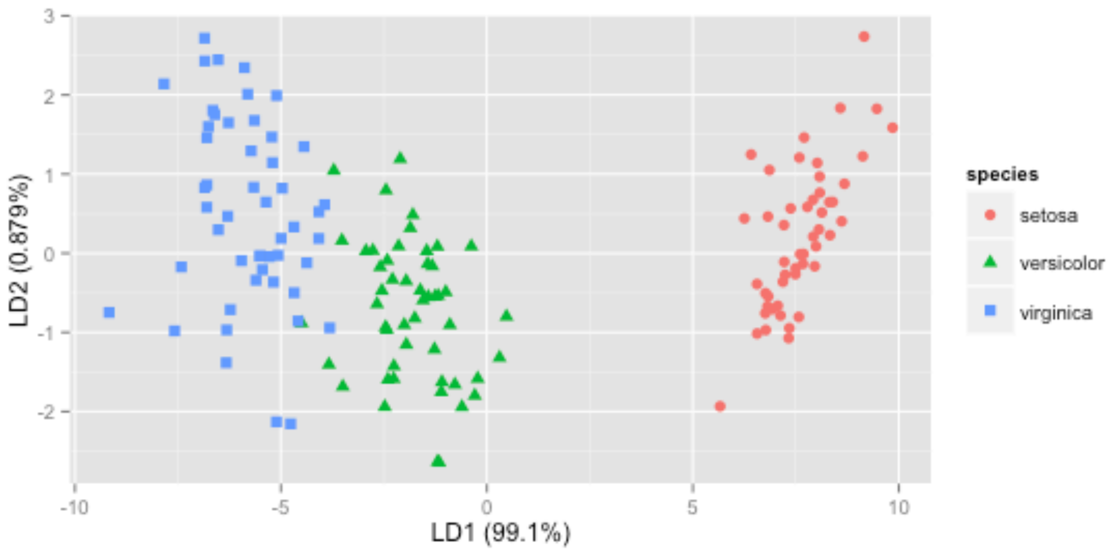
Important for classification is only the (non-isotrop) distance to the centres μ_k

Visualization of the distances to the centres ➔Possibility for dimension reduction

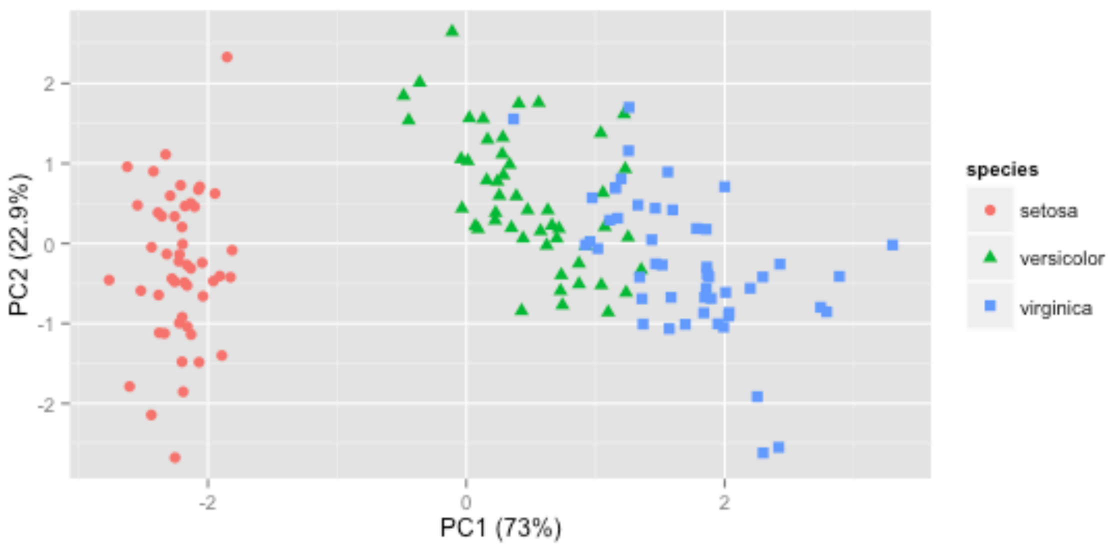
2 classes in can be shown 1D

3 classes can be shown in 2D

Example Iris Data

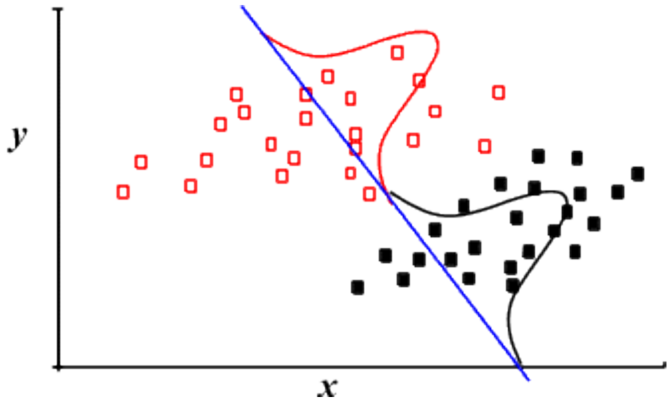


Projection with LDA



Projection with PCA

Fisher's original approach for 2 classes



The data is projected to a line. The orientation is chosen for best discrimination (separation).

In contrast in PCA where the line is chosen to explain the most variation (independent of class labels)

More on that approach Ruckstuhl 6.A or elements of statistical learning

Concluding Remarks (LDA)

- When the classes are well-separated, the parameter estimates for the logistic regression model are surprisingly unstable. Linear discriminant analysis does not suffer from this problem.
- If n is small and the distribution of the predictors X is approximately normal in each of the classes, the linear discriminant model is again more stable than the logistic regression model.
- Linear discriminant analysis is popular when we have more than two response classes, because it also provides low-dimensional views of the data
- There are more classifiers based on Bayesian Principle.
  - E.g naïve base (See Ruckstuhl 5.4)

Crossvalidation

Outline

- Cross Validation (performance measures and splitting techniques)
  - Measures
    - Accuracy
  - Splitting in Training / Testset
    - The Validation Set Approach
    - Leave-One-Out Cross Validation
    - K-fold Cross Validation
    - Bias-Variance Trade-off for k-fold Cross Validation
  - More Measures
  - Pitfalls of cross validation approach

Accuracy as performance measure

Evaluate prediction accuracy on data

Confusion Matrix:

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

For an ideal classifier the off-diagonal entries should be zero: c=0, b=0, or Accuracy=1

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

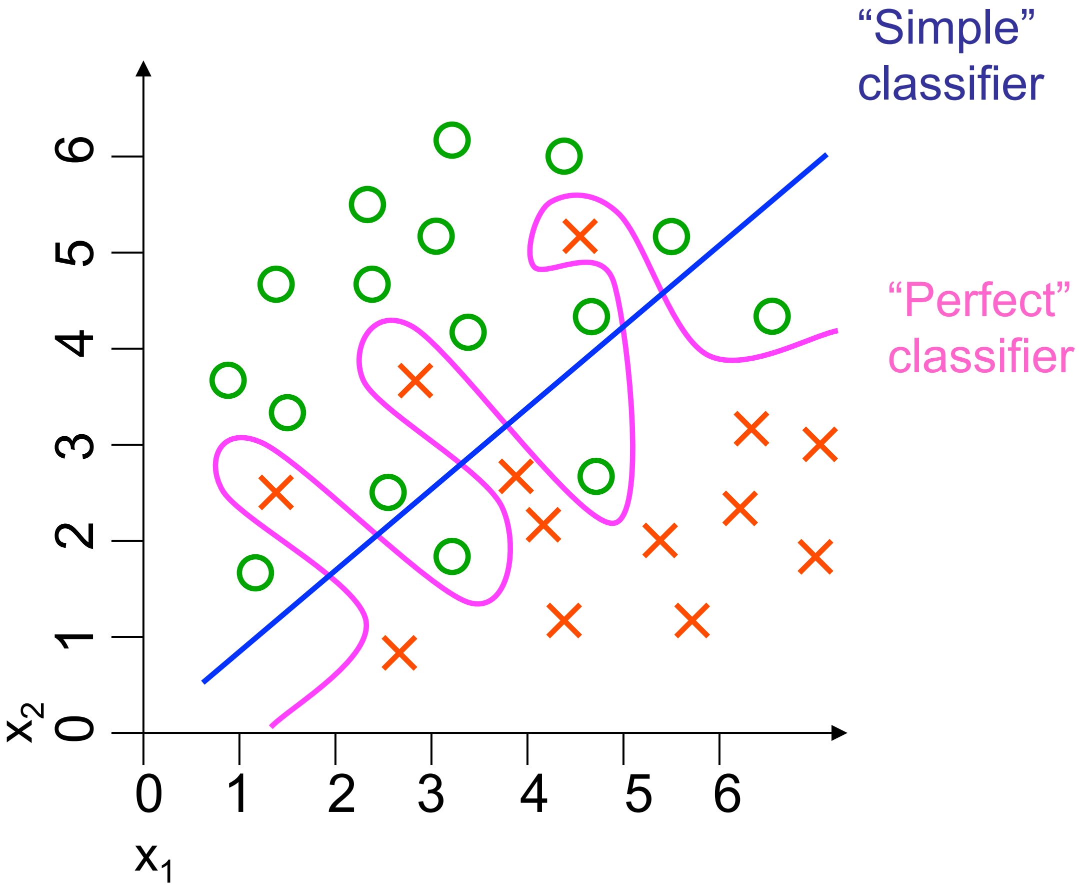
d: TN (true negative)

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Simply count the # correct / all

Source: Tan, Steinbach, Kumar

"Perfect" Vs. "Simple" classifier



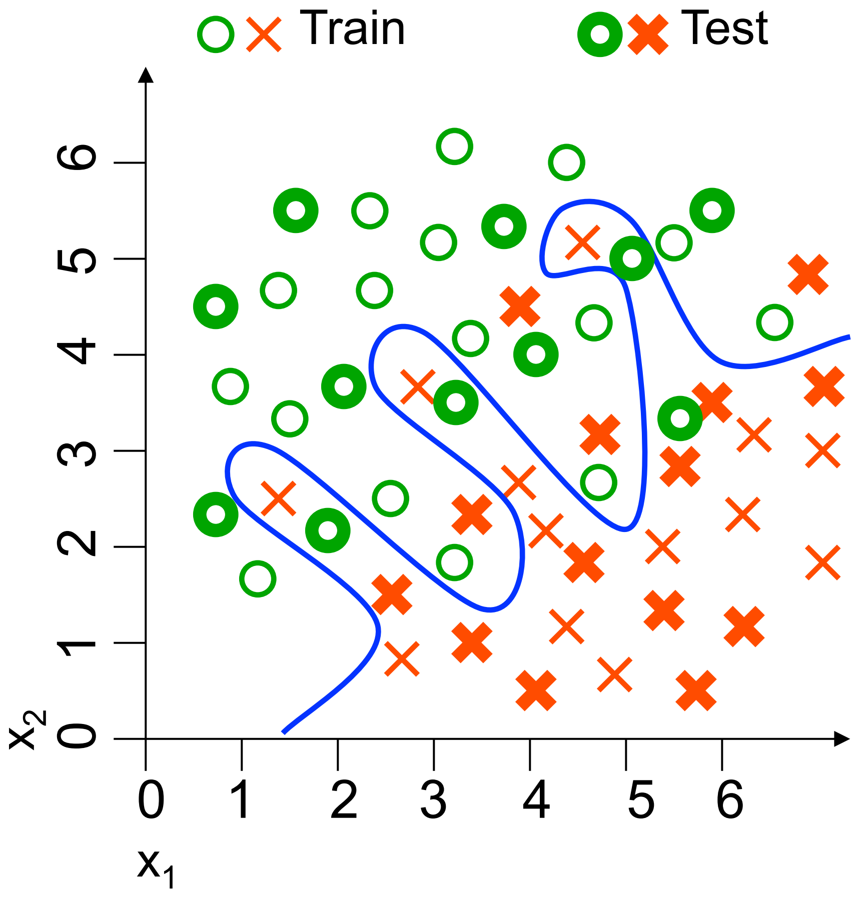
Which is better?

Check on a test-set (don't use all you labeled data to train)

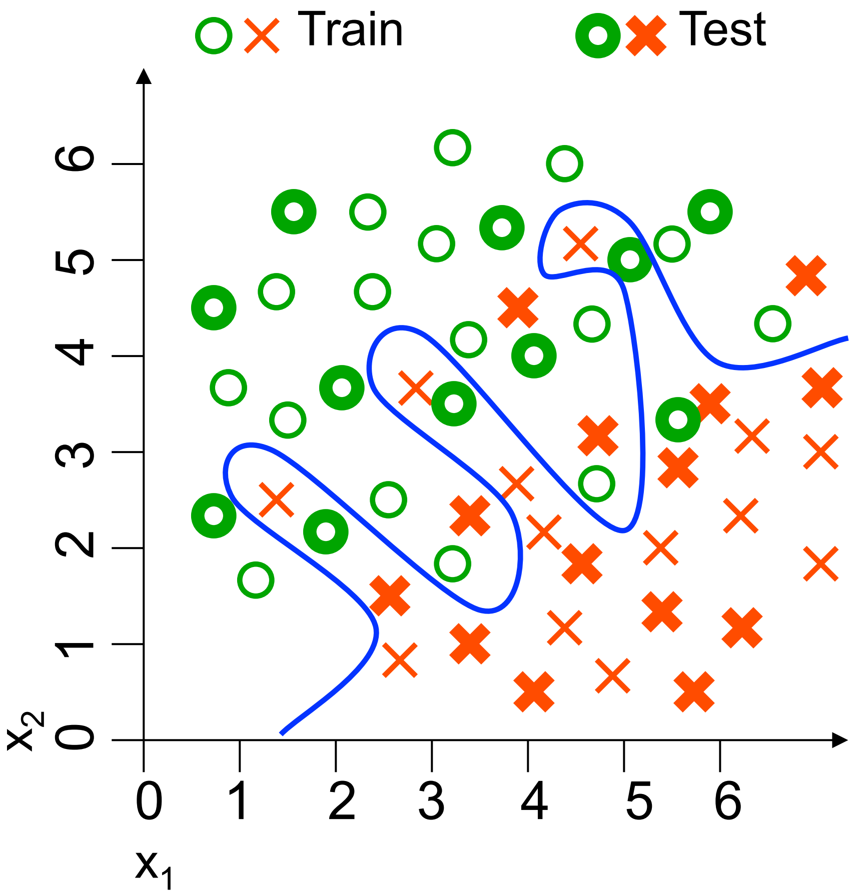
Cross validation of the "Perfect" classifier

<u>Training set:</u>
0%misclassification

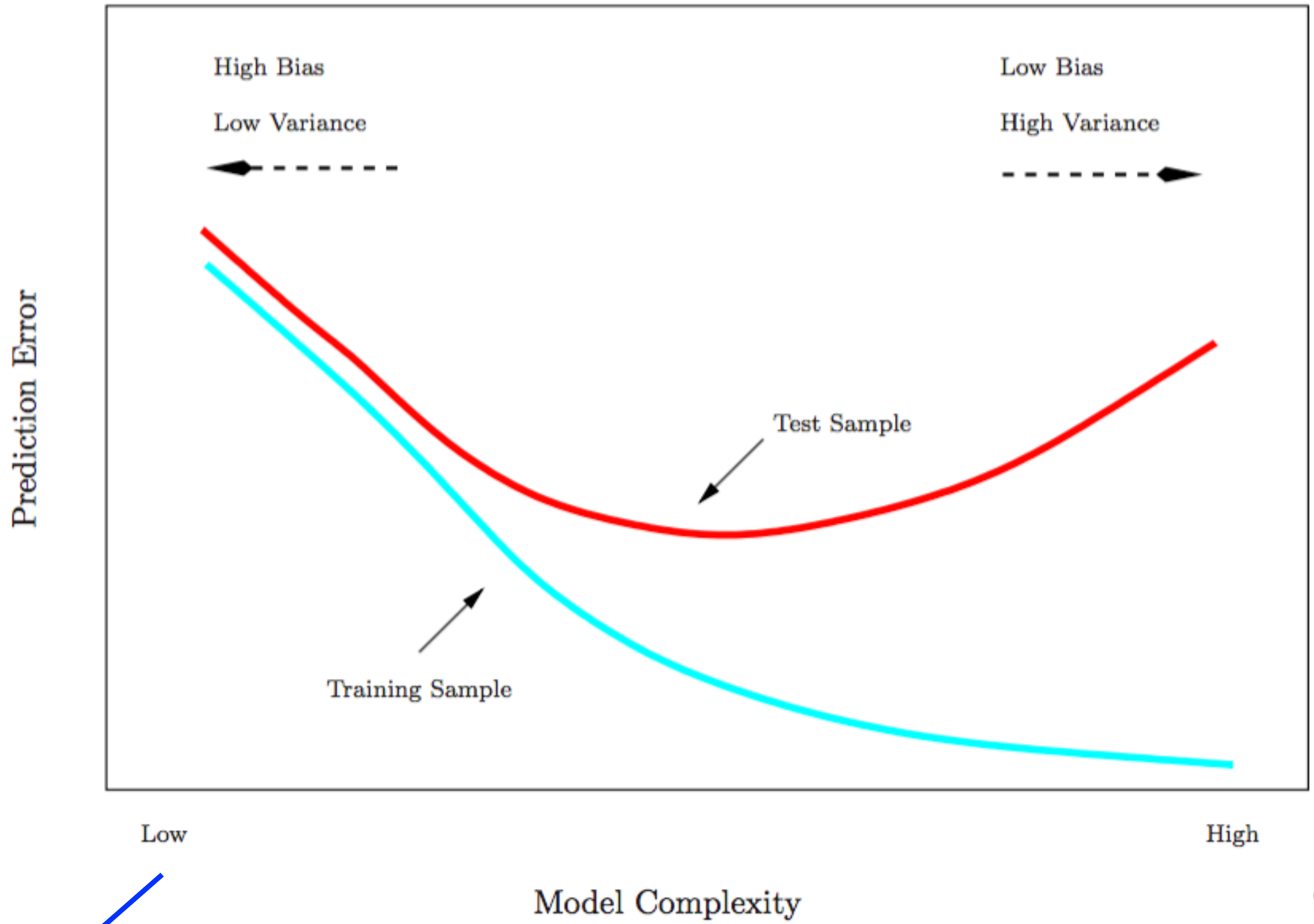
<u>Test set:</u>
8/25=24%
misclassification



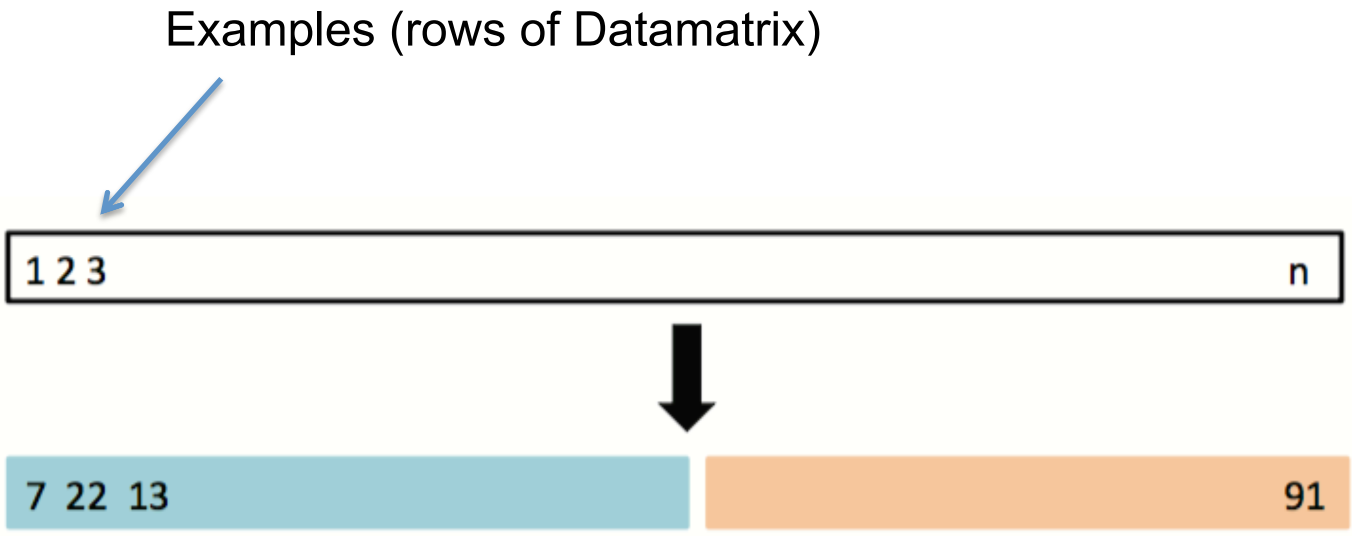
Cross validation of the “Perfect” classifier



First approach validation set approach



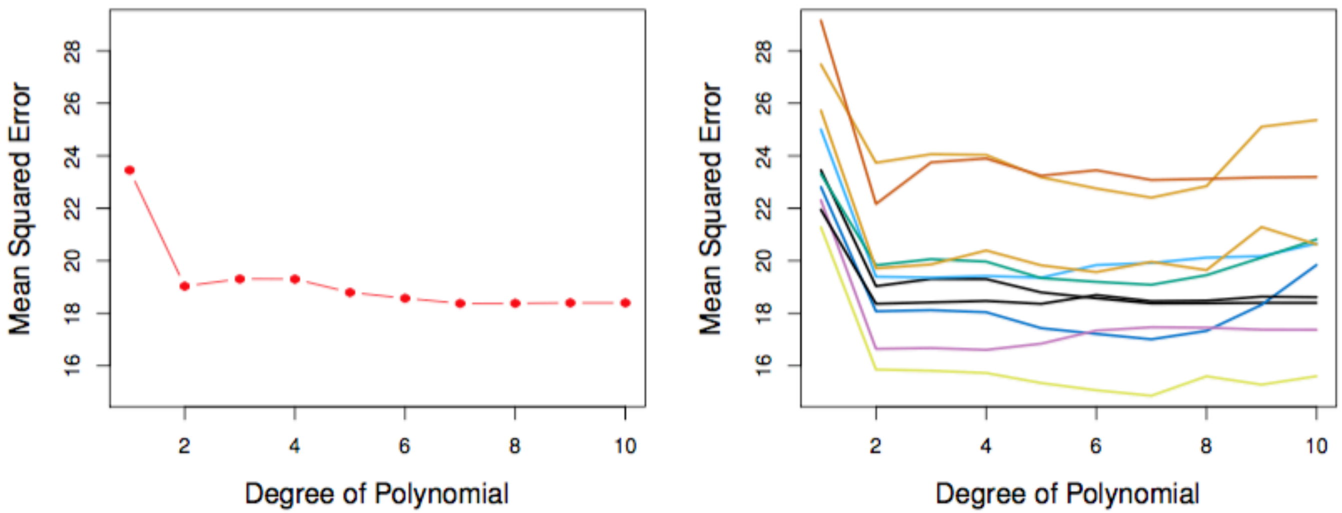
The Validation Set Approach



A random splitting into two halves: left part is training set, right part is validation set

Example

- Want to compare linear vs higher-order polynomial terms in a linear regression
- We randomly split the 392 observations into two sets, a training set containing 196 of the data points, and a validation set containing the remaining 196 observations.



Left panel shows single split; right panel shows multiple splits

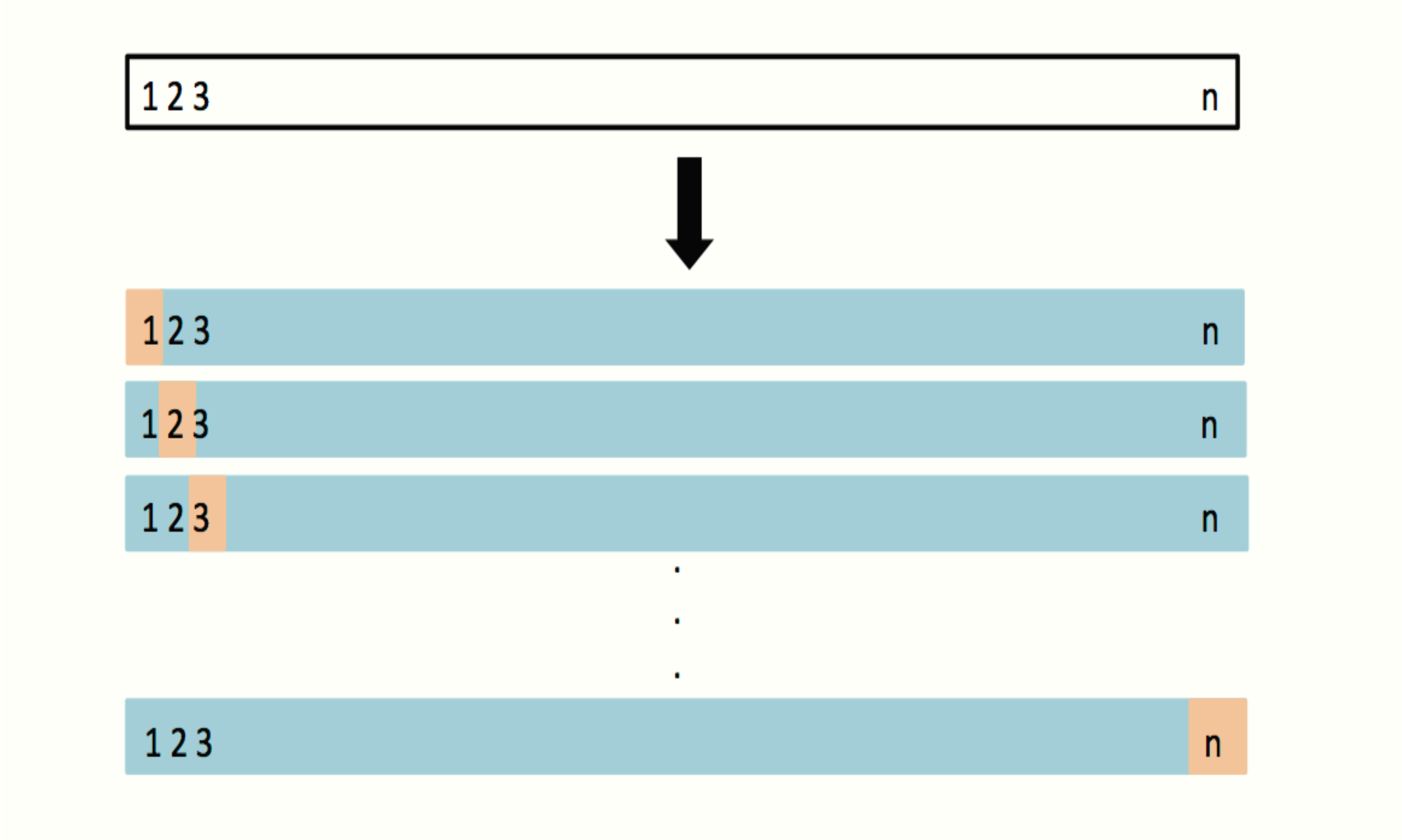
Drawbacks of validation set approach

- The validation estimate of the test error can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set.

- In the validation approach, **only a subset of the observations** — those that are included in the training set rather than in the validation set — are used to fit the model.

- This suggests that the validation set error may tend to overestimate the test error for the model fit on the entire data set. **WHY?**

Leave-One-Out Cross Validation (LOOCV)

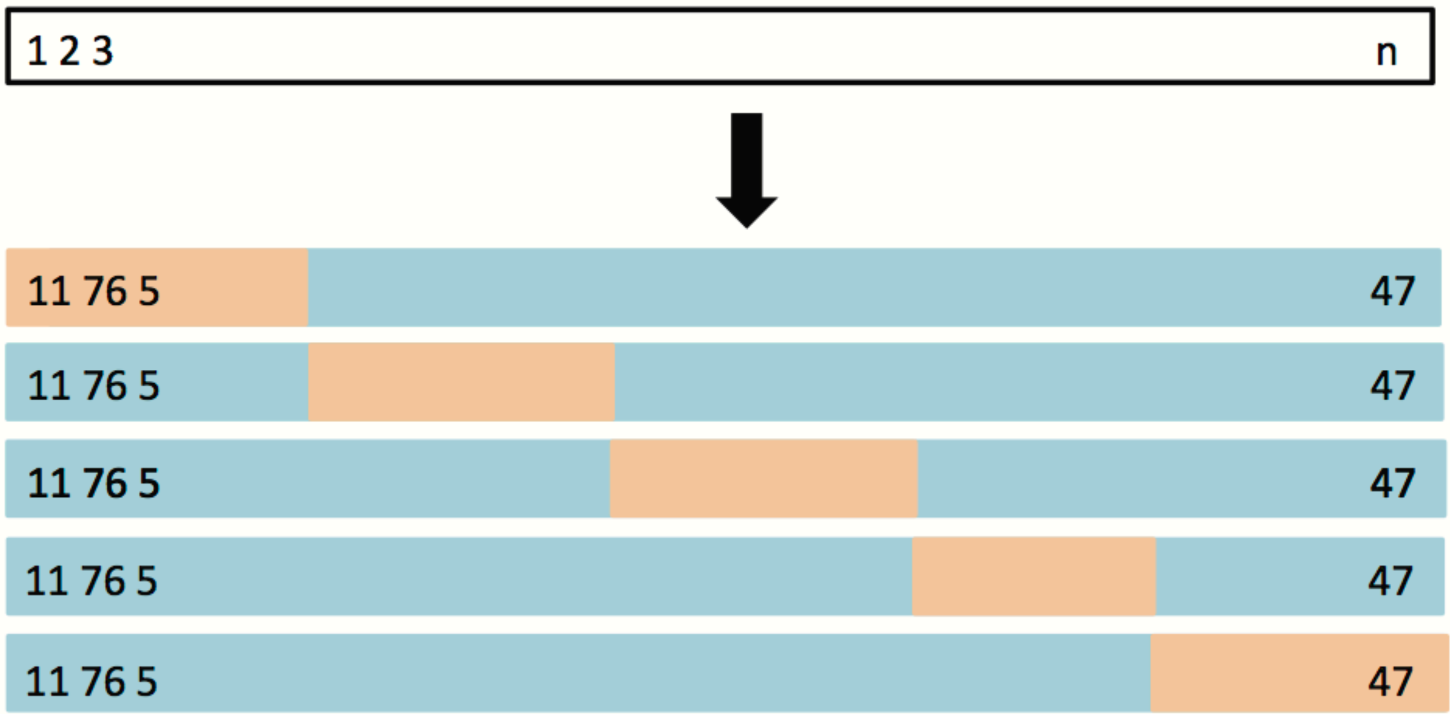


Fit w/o red sample and predict the red sample. Average over all n repeats

LOOCV vs. the Validation Set Approach

- LOOCV has less bias
  - We repeatedly fit the statistical learning method using training data that contains n-1 obs., i.e. almost all the data set is used
- LOOCV produces a less variable MSE
  - The validation approach produces different MSE when applied repeatedly due to randomness in the splitting process, while performing LOOCV multiple times will always yield the same results, because we split based on 1 obs. each time
- LOOCV is computationally intensive (disadvantage)
  - We fit the each model n times!
  - However, certain classifiers can compute LOOCV very fast
    - LDA see Aufgabe

K-fold Cross Validation



Fit w/o red samples and predict the red samples. Average over all k repeats. Do a weighted average if folds do not have the same size.

Question: What happens if k=n?

Ende Woche 9 2015