Machine Learning

Data Train Workshop
October 2021

Marvin N. Wright
wright@leibniz-bips.de

Lukas Burk
burk@leibniz-bips.de

Example: House Prices

Predict the price for a house in a certain area

Features x				Target y
square footage of the house	number of bedrooms	swimming pool (yes/no)	...	house price in US$
1,180	3	0	...	221,900
2,570	3	1	...	538,000
770	2	0	...	180,000
1,960	4	1	...	604,000

Example: Length of hospital stay

Predict days a patient has to stay in hospital

Features x					Target y
diagnosis category	**admission type**	**gender**	**age**	**...**	**Length-of-stay in the hospital in days**
heart disease	elective	male	75	...	4.6
injury	emergency	male	22	...	2.6
psychosis	newborn	female	0	...	8
pneumonia	urgent	female	67	...	5.5

Introduction

Example: Life Insurance

Predict risk category for a life insurance customer

Features x				Target y
job type	**age**	**smoker**	**...**	**risk group**
carpenter	34	1	...	3
stuntman	25	0	...	5
student	23	0	...	1
white-collar worker	39	0	...	2

Learning outcomes

- Understand basic concepts of machine learning
  - Supervised learning
  - Models and learners
  - Overfitting and underfitting
  - Hyperparameter tuning
  - Performance evaluation
- Know the major machine learning methods
  - k-nearest neighbors
  - Decision trees
  - Random forests
  - Boosting
  - Support vector machines
  - Artificial neural networks
- Be able to perform machine learning analyses in `R`

Outline (Day 1)

Morning

1. k-nearest neighbors (kNN)
2. General concepts
3. Decision trees

Afternoon

4. Random forests
5. Model evaluation and resampling
6. Boosting

Outline (Day 2)

Morning

7. Support vector machines
8. Hyperparameter tuning and performance comparison
9. Artificial neural networks

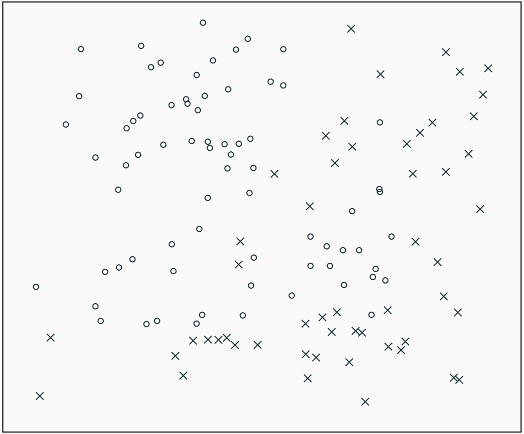
Afternoon

10. Specific endpoints
11. Variable importance and selection
12. Discussion

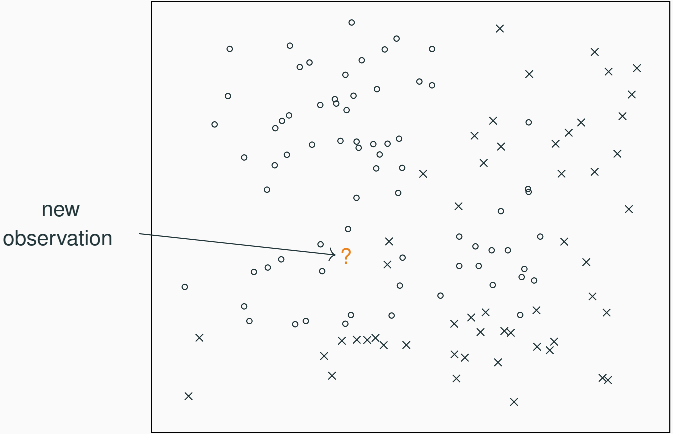
Part 1

k-nearest neighbors (kNN)

What is k-nearest neighbors (kNN)?

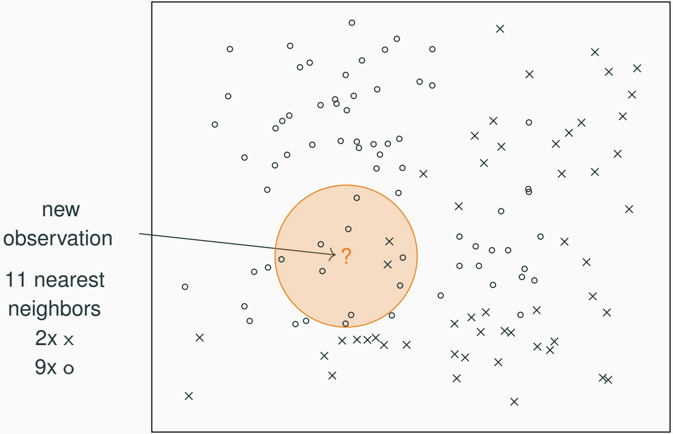


What is k-nearest neighbors (kNN)?



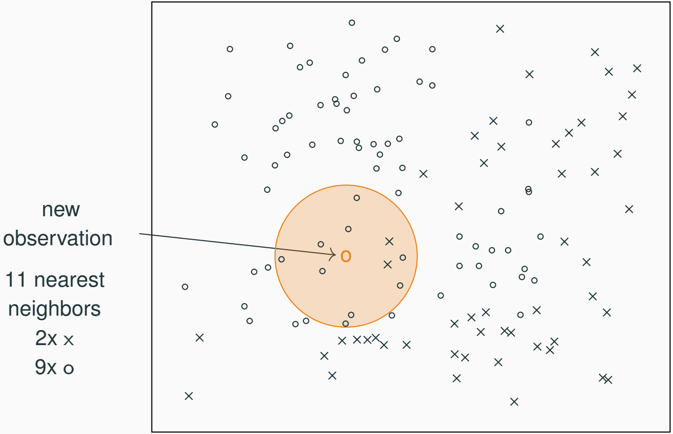
k-nearest neighbors (kNN)

What is k-nearest neighbors (kNN)?



k-nearest neighbors (kNN)

What is k-nearest neighbors (kNN)?



k-nearest neighbors (kNN)

What is kNN formally?

- $N_k(\boldsymbol{x})$ neighborhood of x defined by k closest points $\boldsymbol{x}_i$ in training data
- $\hat{y} = \frac{1}{k} \sum_{\boldsymbol{x}_i \in N_k(\boldsymbol{x})} y_i$
- Closeness implies metric
- Standard metrics: Euclidian, Mahalanobis distance
- Generalization: Weighting schemes, e.g. $w = \frac{1}{d(x, x_i)}$
- kNN assumes: Regression function $\mathbb{E}(y \mid \boldsymbol{x})$ well approximated by locally constant function

Wettschereck et al. 1997 Artif Intell Rev 10:1-37

k-nearest neighbors (kNN)

What are the statistical properties of kNN?

- **Universal consistency (Stone's theorem)**
  kNN is universally consistent if $k \to \infty, n \to \infty, k/n \to 0$
- **Convergence rate**
  Under suitable regularity conditions the rate of convergence of kNN is faster than the minimax rate
- **Asymptotic normality**
  Under suitable regularity conditions the kNN estimator is asymptotically normal

Stone 1977 Ann Stat 5:595 • Györfi et al. 2002 ISBN 0387954414, p. 88 • Biau et al. 2010 IEEE Transact Inform Theor 56:2034 • Biau et al. 2010 J Mach Learn Res 11:687, Corollaries 10 and 14 • Mack 1981 SIAM J Alg Disc Meth 2:311 Biau & Devroye 2015 ISBN 9783319253862, Theorem 14.4

What are the advantages of kNN?

- Fast training
- Simple and easy to understand
- Robust to noisy training data
- Effective if training sample size is large
- Many generalizations available

Guo et al. 2003 LNCS 2888:986-96 • Bhatia & Vandan 2010 Int J Comput Sci Inform Secur 8:302-5

k-nearest neighbors (kNN)

What are the disadvantages of kNN?

- No rule to transfer to new data
- Results sensitive to choice of metric
- Tuning of k required $\rightarrow$ discussed later
- Works not well in high dimensional problems

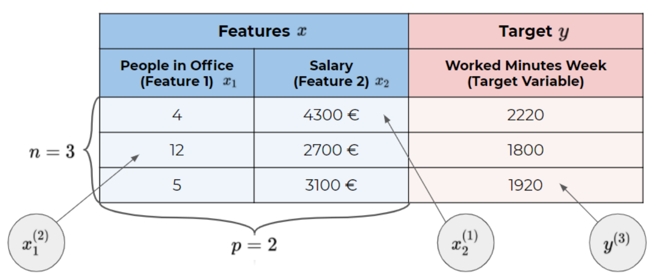
Part 2

General concepts

General concepts

What is supervised learning?

Learn a functional relationship between **features** x and **target** y



Features x		Target y
People in Office (Feature 1) x_1	Salary (Feature 2) x_2	Worked Minutes Week (Target Variable)
4	4300 €	2220
12	2700 €	1800
5	3100 €	1920

What is supervised learning?

Use labeled data to learn a model f

Use model f to predict target y of new data

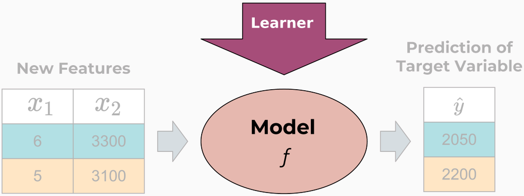
What is the difference between model and learner?

Model
Functional relationship between **features** x and **target** y

Learner (or inducer)
Algorithm for finding model

Train Set

y	x_1	x_2
2200	4	4300
1800	12	2700
1920	15	3100



General concepts

Example

- Learner: Artificial neural network (as a concept)
- Model: Actual network with learned weights

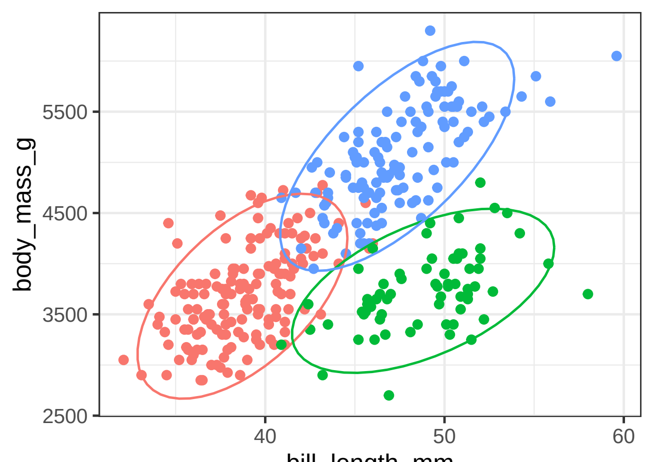
Models differ in size and complexity

- Linear model: Coefficients β
- Neural network: Weights for all units in all layers
- Decision trees: Many binary splits
- k-nearest neighbors: Complete training data

What is unsupervised learning?

No **target** y available

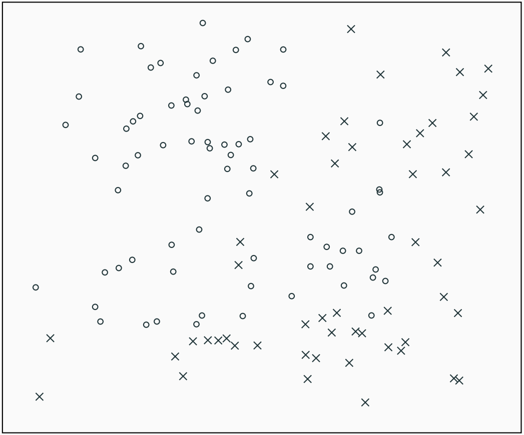
Search for patterns in the data x, e.g. clustering:



Part 3

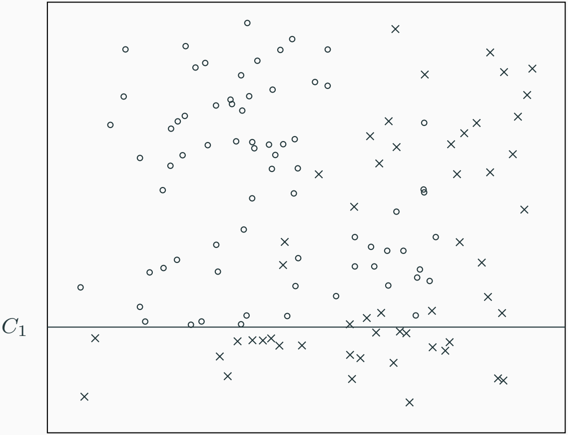
Decision trees

What are classification and probability estimation trees?



Breiman et al. 1984 ISBN 9780412048418

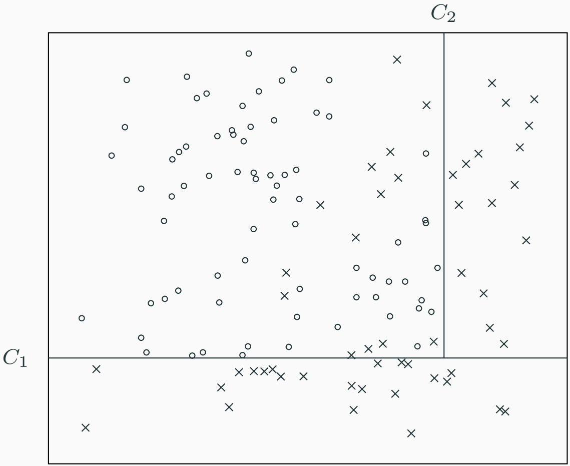
What are classification and probability estimation trees?



Breiman et al. 1984 ISBN 9780412048418

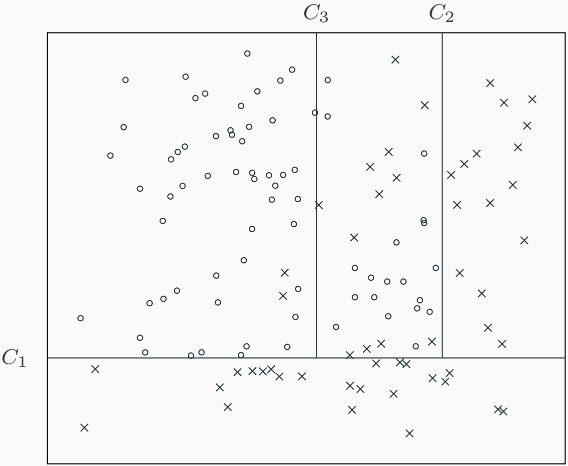
Classification and probability estimation trees

What are classification and probability estimation trees?



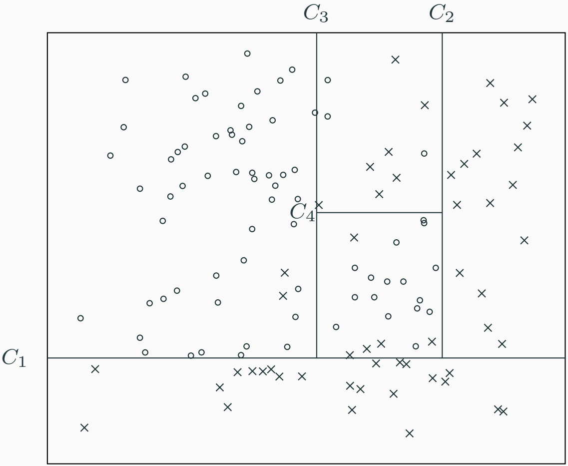
Breiman et al. 1984 ISBN 9780412048418

What are classification and probability estimation trees?



Breiman et al. 1984 ISBN 9780412048418

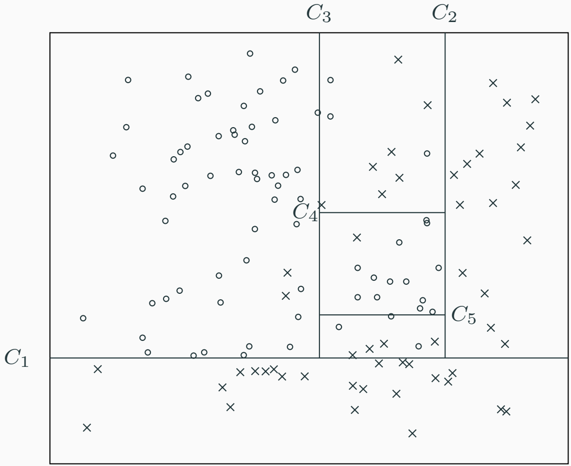
What are classification and probability estimation trees?



Breiman et al. 1984 ISBN 9780412048418

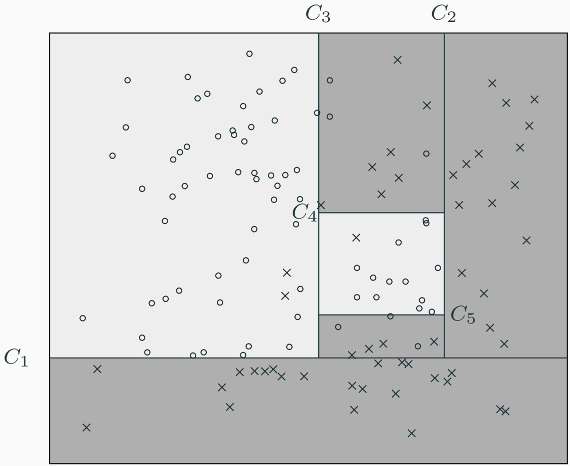
Classification and probability estimation trees

What are classification and probability estimation trees?



Breiman et al. 1984 ISBN 9780412048418

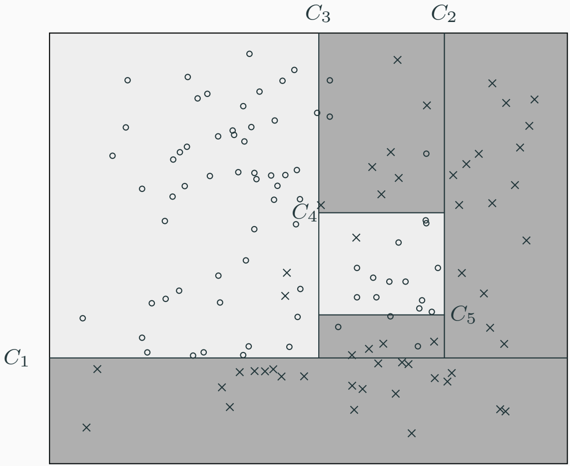
What are classification and probability estimation trees?

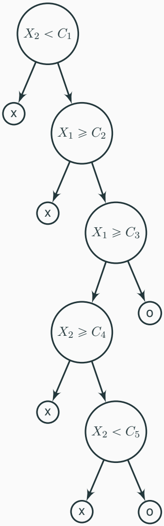


Breiman et al. 1984 ISBN 9780412048418

Classification and probability estimation trees

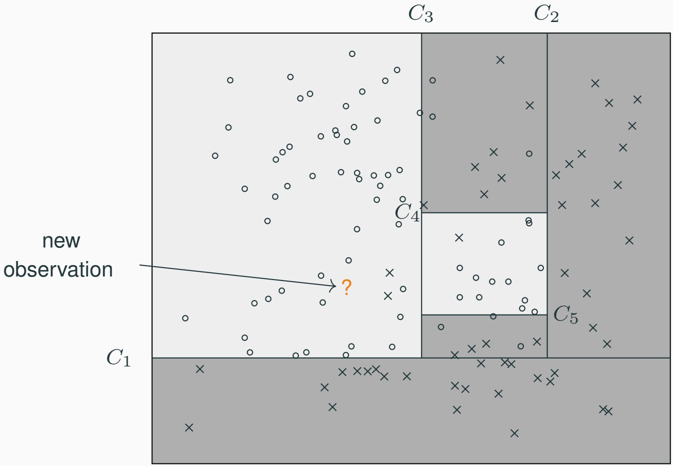
What are classification and probability estimation trees?

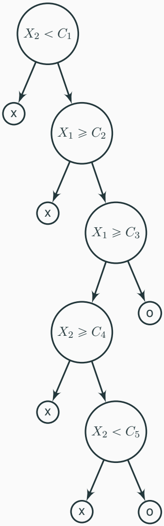




Breiman et al. 1984 ISBN 9780412048418

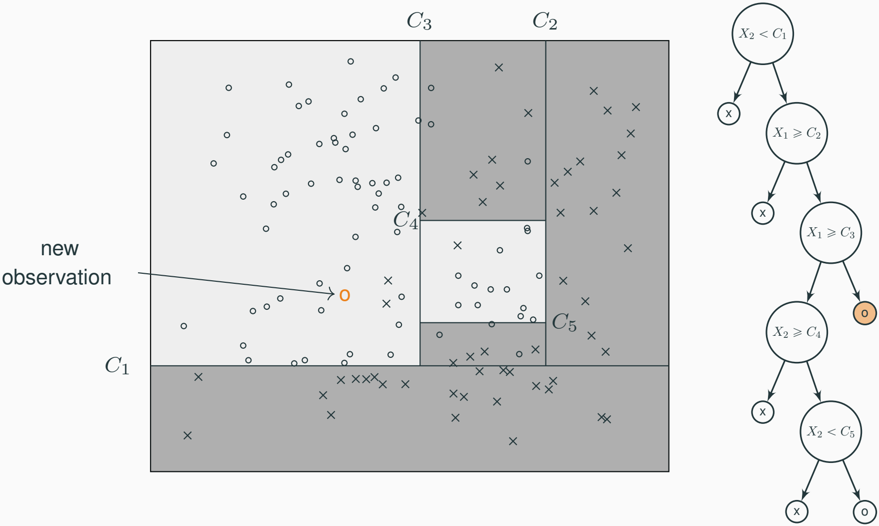
What are classification and probability estimation trees?





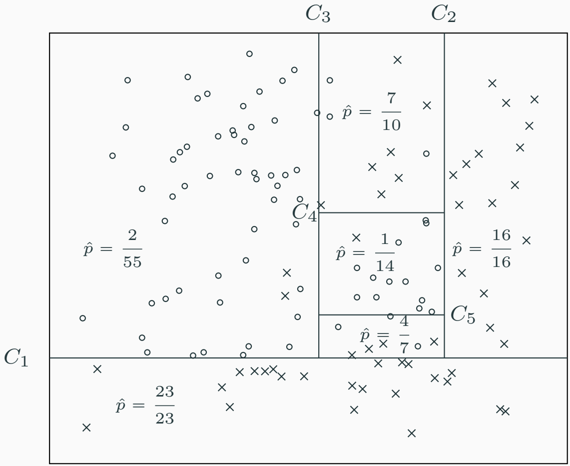
Breiman et al. 1984 ISBN 9780412048418

What are classification and probability estimation trees?



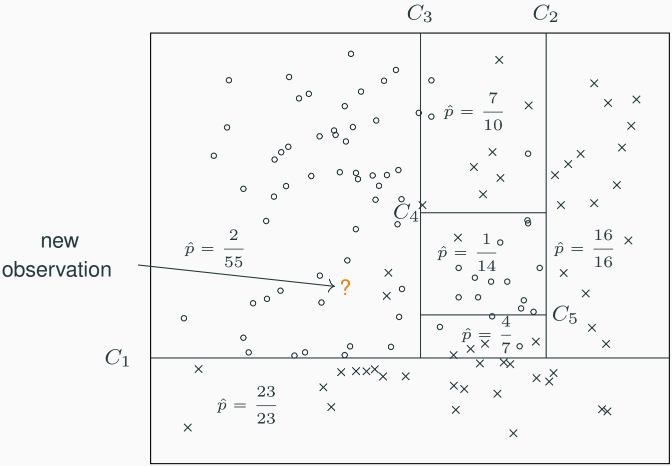
Breiman et al. 1984 ISBN 9780412048418

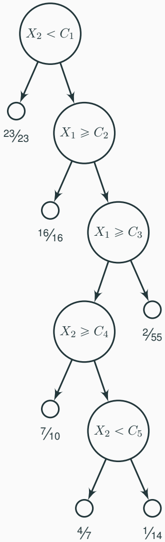
What are classification and probability estimation trees?



Breiman et al. 1984 ISBN 9780412048418

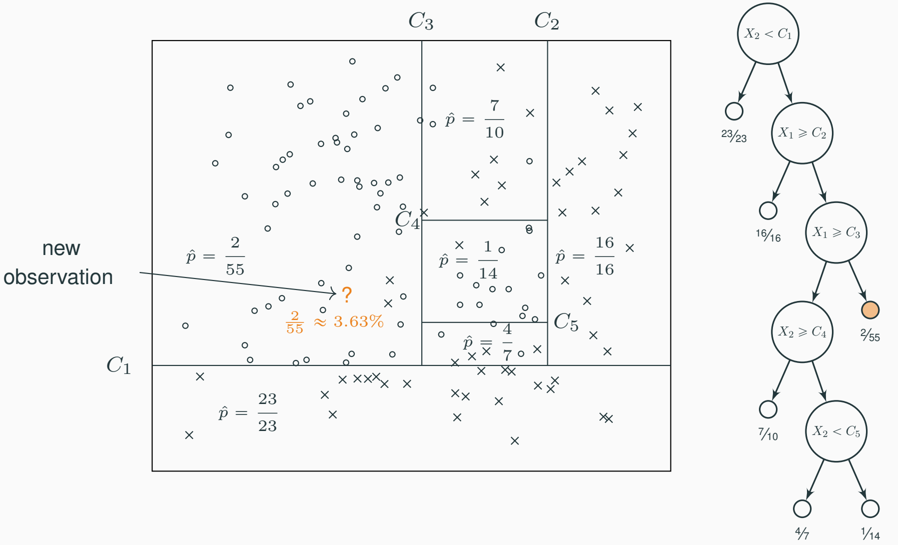
What are classification and probability estimation trees?





Breiman et al. 1984 ISBN 9780412048418

What are classification and probability estimation trees?



Breiman et al. 1984 ISBN 9780412048418

Classification and probability estimation trees

What is the standard algorithm for classification and probability estimation trees?

1. Grow tree
2. Stop tree growing process
3. Prune back branches
4. Select optimal tree

Breiman et al. 1984 ISBN 9780412048418 • König et al. 2008 Int J Data Min Bioinform 2:289

Classification and probability estimation trees

What is the standard algorithm for classification and probability estimation trees?

1. Grow tree : Recursively split nodes
   a) For each independent variable x_j, consider each possible binary split (partition), compute child node impurity
   b) Select variable x_j and split point yielding largest decrease in impurity
   c) Split in exactly two child nodes at optimal split point
2. Stop tree growing process
3. Prune back branches
4. Select optimal tree

Breiman et al. 1984 ISBN 9780412048418 • König et al. 2008 Int J Data Min Bioinform 2:289

Classification and probability estimation trees

What is the standard algorithm for classification and probability estimation trees?

1. Grow tree
2. Stop tree growing process
   a) In a node with only identical outcome (pure node)
   b) In a node with only identical variable values (no split possible)
   c) If external limit on tree complexity, tree depth or node size reached
3. Prune back branches
4. Select optimal tree

Breiman et al. 1984 ISBN 9780412048418 • König et al. 2008 Int J Data Min Bioinform 2:289

Classification and probability estimation trees

What is the standard algorithm for classification and probability estimation trees?

1. Grow tree
2. Stop tree growing process
3. Prune back branches
    a) Tradeoff between complexity and accuracy
    b) Estimate accuracy in test data set
    c) Time consuming
4. Select optimal tree

Breiman et al. 1984 ISBN 9780412048418 • König et al. 2008 Int J Data Min Bioinform 2:289

Classification and probability estimation trees

What are standard split criteria for classification trees?

Measures of impurity for classification trees

- Misclassification error: $1 - max(p, 1 - p)$
- Gini index: $2p(1 - p)$
- Deviance: $-p \cdot ln(p) - (1 - p) \cdot ln(1 - p)$

p: proportion of subjects with $y = 1$ in node

Dichotomous probability estimation

Gini index splitting and variance splitting identical

König et al. 2008 Int J Data Min Bioinf 2:289-341

Classification and probability estimation trees

What is the standard algorithm for classification and probability estimation trees?

Number of possible splits for a variable

- Dichotomous categorical variable, e.g., sex:
  1 possible split point
- Ordered categorical variable, e.g., age in years:
  $m - 1$ possible split points
- Continuous independent variable, e.g., biomarker:
  $n - 1$ possible split points
- Unordered categorical variable, e.g., hair color:
  $2^m - 1$ possible split points

Breiman et al. 1984 ISBN 9780412048418 • Fisher 1958 J Am Stat Assoc 53:789
Hastie et al. 2009 ISBN 9780387848570, p. 310 • Ripley 1996 ISBN 9780521717700, p. 238

How can the computational complexity be reduced for unordered categorical variables?

- Idea: separate high case prob. categories from low case prob. categories
- Order categories by proportion of 1's
- Split as ordered categorical variable
- Split points shown to be optimal for binary classification and regression
- Approximations for multiclass classification available

Breiman et al. 1984 ISBN 9780412048418 • Fisher 1958 J Am Stat Assoc 53:789
Hastie et al. 2009 ISBN 9780387848570, p. 310 • Ripley 1996 ISBN 9780521717700, p. 238

Classification and probability estimation trees

What are the advantages of classification and probability estimation trees?

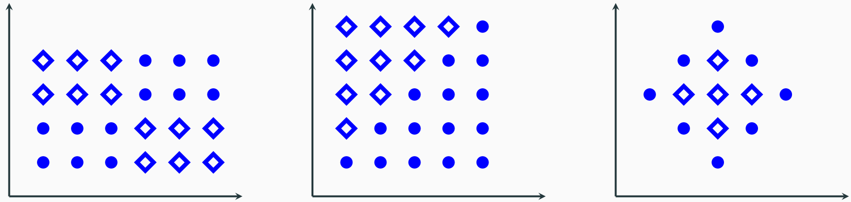
- Procedure intuitive
- Small trees simple to interpret
- Intrinsic variable selection
- Simple handling of outliers
- Fast training
- Usually better prediction performance than kNN

Steinberg 2009 ISBN 9781420089646, pp. 179

Classification and probability estimation trees

What are the disadvantages of classification and probability estimation trees?

- Trees unstable
- Pruning can be computationally intensive
- Usually worse prediction performance than random forests and boosted trees (covered later)
- Problematic data sets

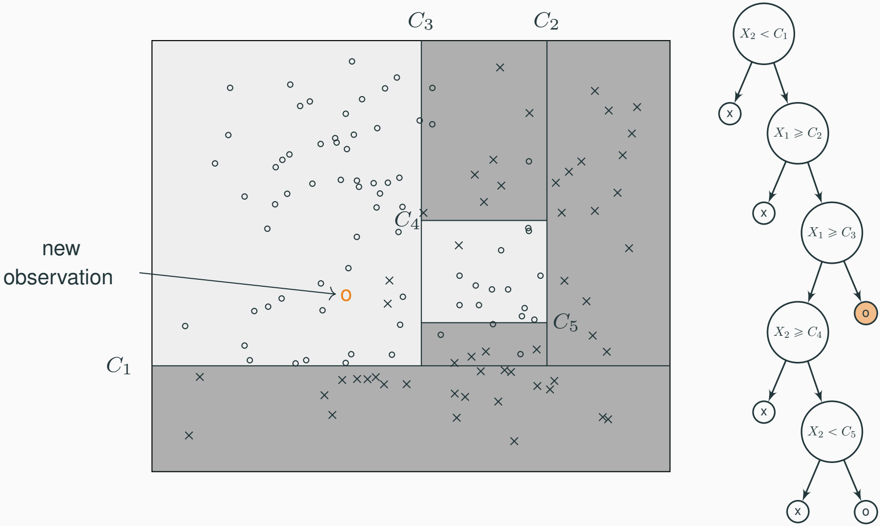


Steinberg 2009 ISBN 9781420089646, pp. 179

Part 4

Random forests

What are classification and probability estimation trees?



Breiman et al. 1984 ISBN 9780412048418

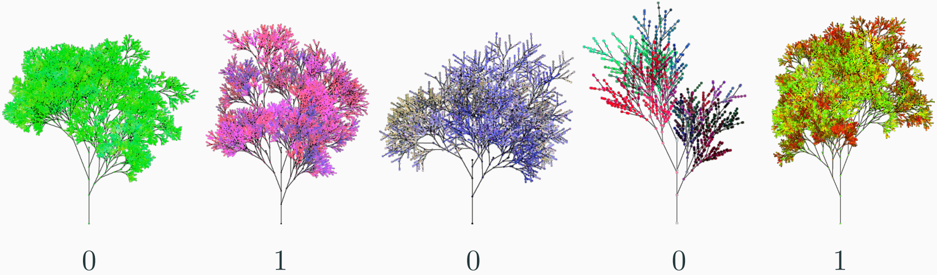
What are random forests?

0

Breiman 2001 Mach Learn 45:5-32 • Malley et al. 2012 Methods Inf Med 51:74-81 • luc.devroye.org

Random forests

What are random forests?



Random forests

What are random forests?

$$\underbrace{0 \qquad 1 \qquad 0 \qquad 0 \qquad 1}_{0}$$

Classification: **majority vote** over all trees

Breiman 2001 Mach Learn 45:5-32 • Malley et al. 2012 Methods Inf Med 51:74-81 • luc.devroye.org

Random forests

What are random forests?

$$\underbrace{0 \qquad 1 \qquad 0 \qquad 0 \qquad 1}_{0.4 < 0.5 \Rightarrow 0}$$

Classification: **majority vote** over all trees

Identical to average over all trees, cut point 0.5

Breiman 2001 Mach Learn 45:5-32 • Malley et al. 2012 Methods Inf Med 51:74-81 • luc.devroye.org

Random forests

What are random forests?

$$\underbrace{8.7\% \qquad 52.6\% \qquad 21.3\% \qquad 3.1\% \qquad 69.6\%}_{31.1\%}$$

Probability estimation: Average over all trees

Breiman 2001 Mach Learn 45:5-32 ● Malley et al. 2012 Methods Inf Med 51:74-81 ● luc.devroye.org

What are random forests?

Two components of randomization

- Data manipulation in rows: bootstrapping / subsampling
- Data manipulation in columns: feature subsampling

Breiman 2001 Mach Learn 45:5

What is bootstrap aggregating (bagging)?

- Ensemble = committee of experts
- Single weak learner = single committee member
- Ensemble decision = committee decision

Fundamental idea of boosting (later) and bagging (bootstrap aggregating)

Any machine can be used as *base learner*, e.g. kNN or tree

What are bootstrapping and subsampling?

- Bootstrapping
  - Sampling **with** replacement
  - Original sample size n, resampled sample size n
  - On average $lim_{n\to\infty}\left(1-\frac{1}{n}\right)^n \approx 0.632 \approx 2/3$ resampled
- Subsampling
  - Sampling **without** replacement
  - Original sample size n, resampled sample size $< n$
  - Standard: resampling of $0.632n$

Malley et al. 2002 ISBN 0387954414, p. 93 • Ziegler & König 2014 Wiley Interdiscip Rev Data Mining Knowl Discov 4:55-63

What is feature subsampling?

At a node consider only subset of features

- Trees vary
- “Experts” differ in their opinion
- Reduce correlation between trees

Number of features considered at split

$\texttt{mtry} = \sqrt{d}, \ln d$ or $d/3 \rightarrow$ Tuning possible (later)

Breiman 2001 Mach Learn 45:5-32

What is the standard random forest algorithm?

For each tree

1. Draw bootstrap sample with replacement
2. Grow tree
   a) Use random subset of variables (`mtry`) at each node
   b) Stop if minimum node size reached
3. Determine proportion of ‘1’ in each terminal node

New subject

1. Drop down subject in each single tree
2. Store proportion from all trees
3. Average proportion of ‘1’s over all trees

Kruppa et al. 2013 Expert Sys Appl 40:5125-31

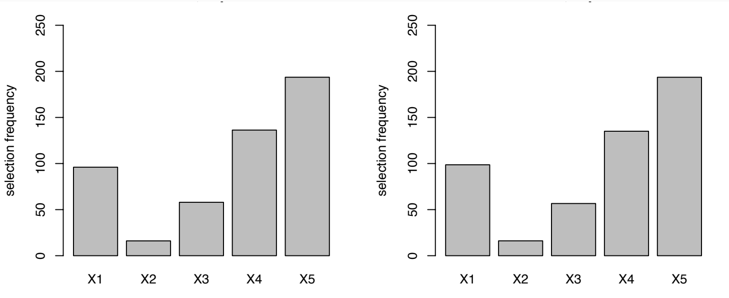
Which variables are selected if they have no effect?

Standard random forest: Bias in split variable selection

- All independent variables unimportant
- X1 $\sim$ N(0,1), X2 $\sim$ M(2), X3 $\sim$ M(4), X3 $\sim$ M(10), X5 $\sim$ M(20)

Bootstrap

Subsample



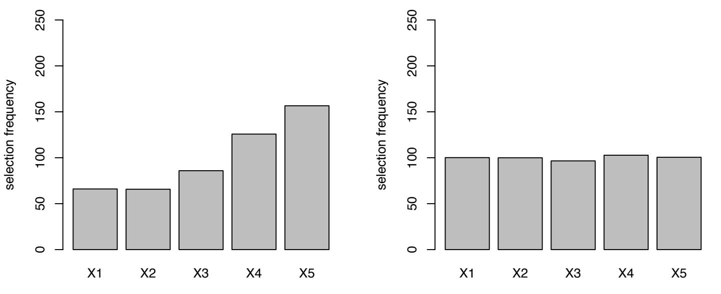
Strobl et al. 2007 BMC Bioinformatics 8:25

Which variables are selected if they have no effect?

Conditional inference forest with subsampling: NO bias in split variable selection

Bootstrap

Subsample



Strobl et al. 2007 BMC Bioinformatics 8:25

What are conditional inference forests?

Standard random forest

- One-step procedure: maximize gain in purity over all independent variables and all possible split points
- No natural stop criterion

Conditional inference forest (CIF)

- Two-step procedure:
  Step 1: select variable to split on (association test)
  Step 2: select optimal split point
- Natural stop through p-value criterion
- Adjustment for multiple testing of split points

Breiman et al. 1984 ISBN 9780412048418 • Hothorn et al. 2006 J Comput Graph Stat 16:651-74
Strobl et al. 2007 BMC Bioinformatics 8:25 • Wright et al. 2017 Stat Med 36:1272

Random forests

What are the statistical properties of trees and RF?

- **Consistency**
  - Single tree not consistent
  - RF consistent if $a_n/n \to 0$ and $a_n \to \infty$ (a_n/n: Subsampling rate)
- **Convergence rate**
  - Single trees slower than minimax rate
  - RF achieves minimax rate. If more than 54% of variables have no effect, convergence rate faster than minimax
- **Asymptotic normality**
  - Single tree predictions asymptotically normally distributed
  - RF predictions asymptotically normally distributed for subsampling

Few results for sampling with replacement

Biau et al. 2008 J Mach Learn Res 9:2015-33 • Scornet 2016 J Multiv Anal 146:72-83
Wager et al. 2014 J Mach Learn Res 15:1625-51 • Wager & Walther 2016 arXiv 1503.06388v3

What are the advantages of random forests?

- As with trees: Procedure intuitive, intrinsic variable selection, simple handling of outliers, fast training
- Work well with high dimensional data
- Work well without (or with only a little) tuning
- Usually better prediction performance than a single tree

Random forests

What are the disadvantages of random forests?

- Not simple to interpret
- Usually worse prediction performance than well tuned boosted trees

Part 5

Model evaluation and resampling

What is model evaluation?

How goood is a prediction model?

Compare true target y with predicted target $\hat{y}$

Examples

- How many patients correctly diagnosed?
- How many emails correctly detected as ham or spam?
- How close is the predicted price of a house to the true value?
- How close is the length of hospitalization to the true value?

Model evaluation and resampling

What are standard performance measures?

Dichotomous (binary) outcome

- Proportion of correct classifications (PC):
  $\widehat{PC} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{y_i = \hat{y}_i}$
- ROC, AUC: $\hat{\mathbb{P}}(y = 1 \mid x)$
- Brier score (BS), i.e., MSE of probability estimates; also probability score (PS): $\widehat{BS} = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{\mathbb{P}}\left(y_i = 1 \mid x_i\right)\right)^2$
- Root mean square error (RMSE): $\widehat{RMSE} = \sqrt{\widehat{BS}}$
- Brier skill score (BSS), i.e., Brier score normalized to reference BS_{ref}: $\widehat{BSS} = \frac{\widehat{BS}}{BS_{ref}}$
- Mean absolute error (MAE):
  $\widehat{MAE} = \frac{1}{n} \sum_{i=1}^{n} \left| y_i - \hat{\mathbb{P}}(y_i = 1 \mid x_i) \right|$

König et al. 2008 Int J Data Min Bioinform 2:289-341 • Kruppa et al. 2014 Biom J 56:564-83

Model evaluation and resampling

What are standard performance measures?

Multicategory outcome

- Proportion of correct classifications (PC)
- Averaged class-wise PC
- ROC, AUC: several extensions

Continuous outcome

- MSE: $\widehat{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2$
- MAE: $\widehat{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|$
- RMSE: $\widehat{RMSE} = \sqrt{\widehat{MSE}}$
- Explained variance: $\hat{R}^2 = \frac{1-\widehat{MSE}}{\widehat{\mathbb{V}ar}(y)}$

Mossmann 1999 Med Decis Making 19:78–89 • Hand & Till 2001 Mach Learn 45:171–86
Provost & Domingos 2003 Mach Learn 52:199–215

What are standard performance measures?

Survival outcome

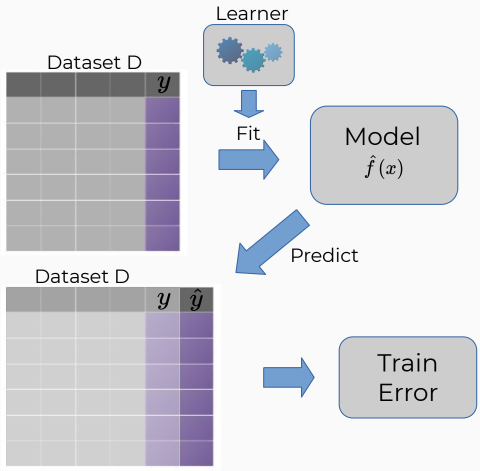
- C-Index
- Time-dependent Brier Score
- Integrated Brier score

Harrell et al. 1982 JAMA 247:2543–6 • Graf et al. 1999 Stat Med 18:2529–45
Mogensen et al. 2012 J Stat Softw 50:1–23

What is model evaluation?

Training error

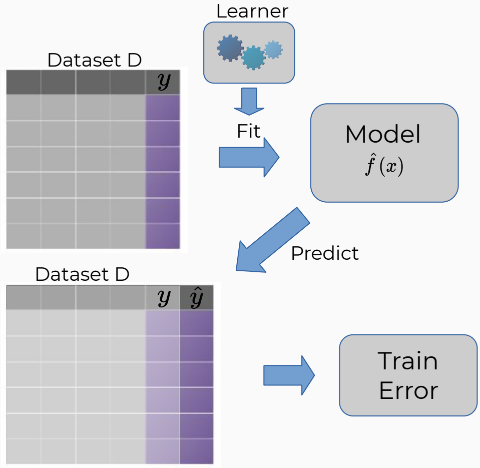
Evaluate performance on training data



What is model evaluation?

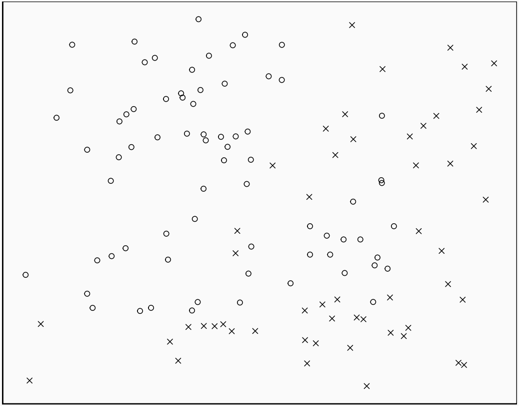
Training error

Evaluate performance on training data

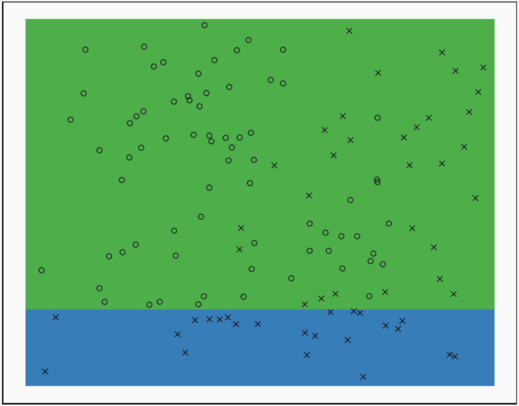


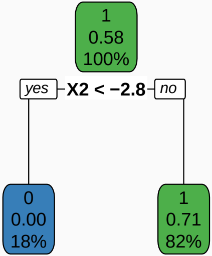
Problem: Overfitting

What is overfitting?

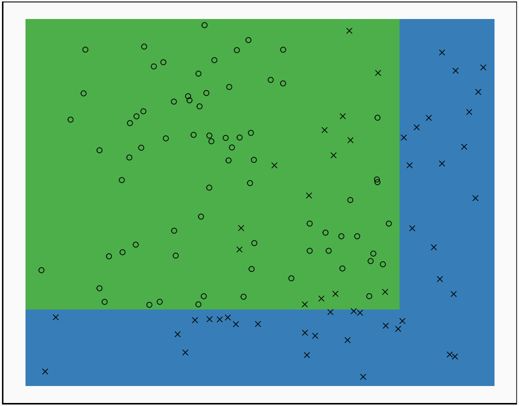


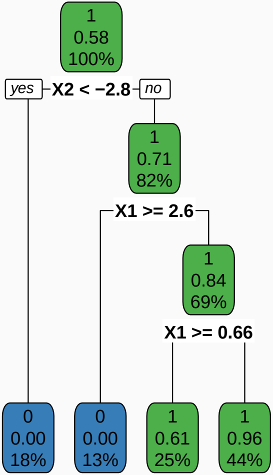
What is overfitting?



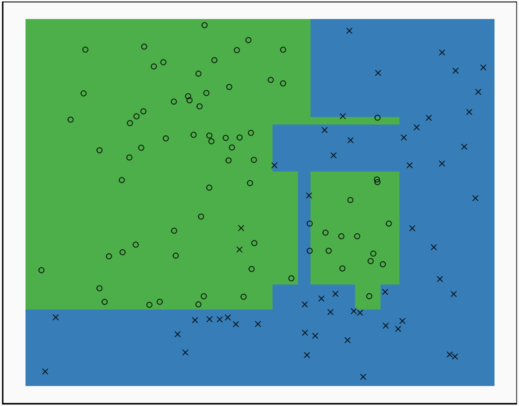


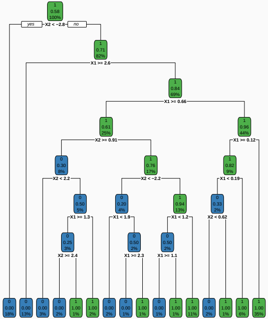
What is overfitting?



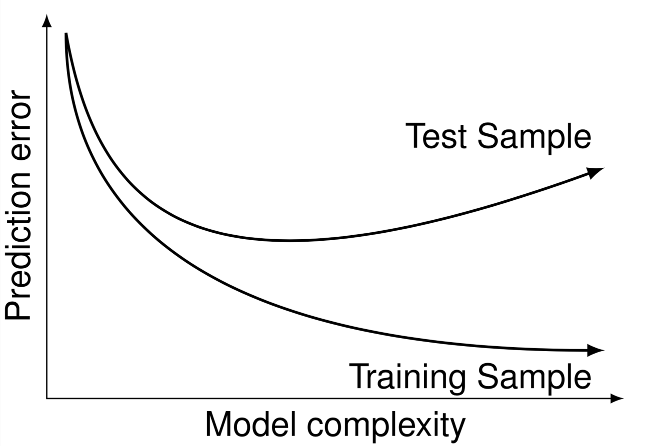


What is overfitting?





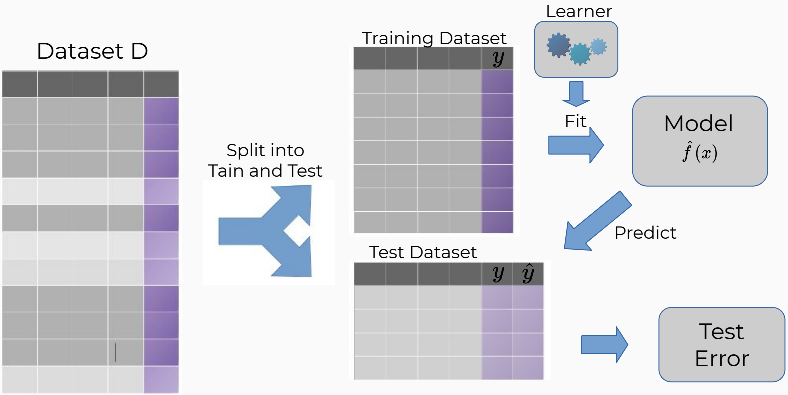
What is overfitting?



Model evaluation and resampling

What is model evaluation?

Test error



Model evaluation and resampling

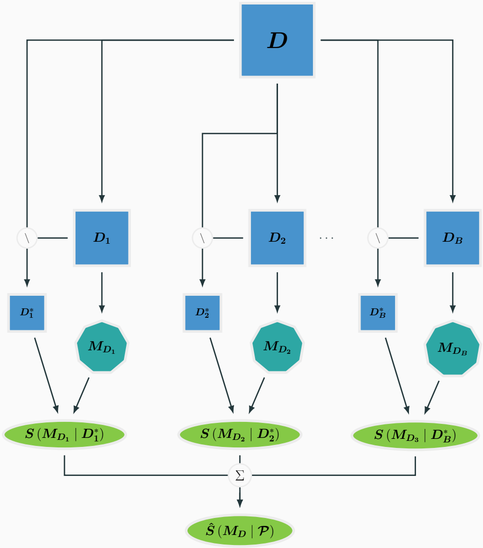
What is model evaluation?

Training and test error

- Training error heavily biased
- Test error (almost) unbiased but variance unknown

Model evaluation and resampling

What is resampling?



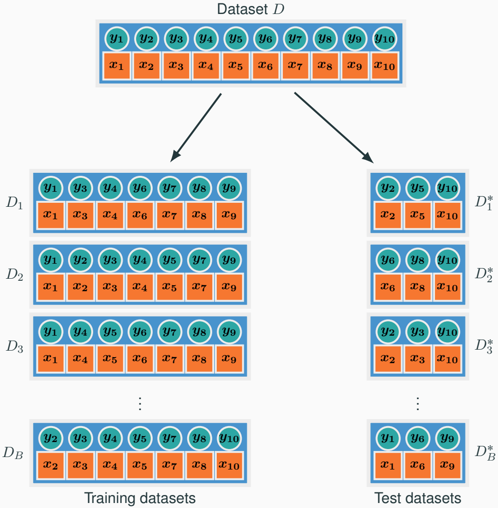
What is resampling?

- Estimate performance on independent data
- Used for
  - Performance estimation
  - Hyperparameter tuning
  - Model selection
- Resampling based performance estimation
  1. Split dataset in several (smaller) datasets D_b
  2. On each dataset D_b:
     2.1 Train machine
     2.2 Estimate performance on $D_b^* = D \backslash D_b$
  3. Aggregate performance estimates

Simon 2007 ISBN 9780387475080 • Bischl et al. 2012 Evol Comput 20:249–75

What is resampling?

Subsampling



Model evaluation and resampling

What is resampling?

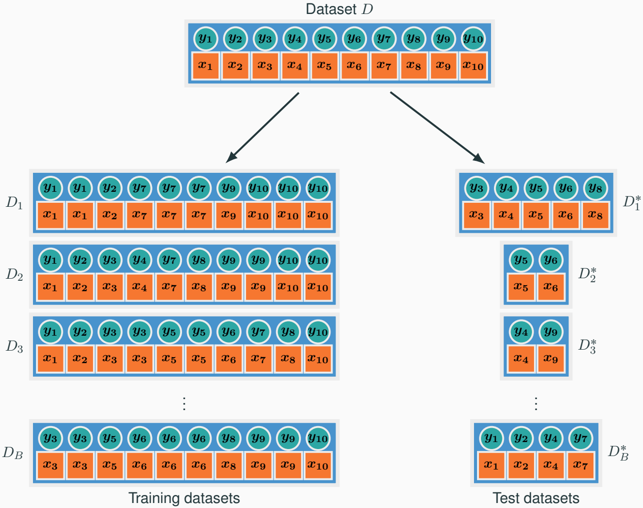
Subsampling

- Sample B training datasets D_b from D without replacement, usually $n_b = \frac{2}{3}n$
- Use $D_b^* = D \backslash D_b$ as test datasets
- D_b and D_b^* disjunct
- D_1 and D_2 not disjunct
- D_1^* and D_2^* not disjunct
- Performance estimator biased
- No optimal B, usually $100 < B < 1000$
- Special case with $B = 1$: Single train/test split (holdout)

Molinaro et al. 2005 Bioinformatics 21:3301–07 • Bischl et al. 2012 Evol Comput 20:249–75

What is resampling?

Bootstrapping



What is resampling?

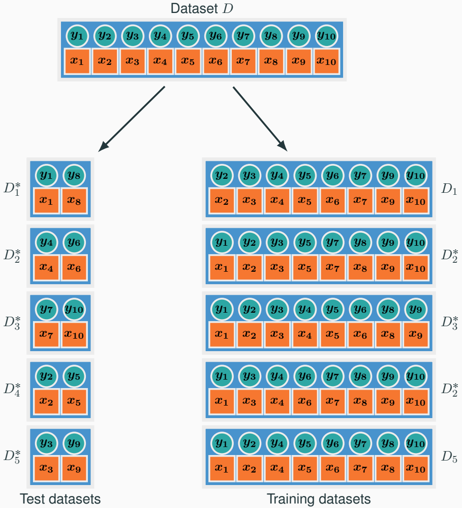
Bootstrapping

- Sample B training datasets D_b from D with replacement, usually $n_b = n$
- Use $D_b^* = D \backslash D_b$ as test datasets
- D_b and D_b^* disjunct
- D_1 and D_2 not disjunct
- D_1^* and D_2^* not disjunct
- Performance estimator biased
- Adaptive weighting of performance estimators to reduce bias (.632+ bootstrap)
- Small variance (large B)
- No optimal B, usually $100 < B < 1000$

Molinaro et al. 2005 Bioinformatics 21:3301–07 • Efron & Tibshirani 1997 JASA 92:548–60 • Bischl et al. 2012 Evol Comput

What is resampling?

Cross validation (CV)



What is resampling?

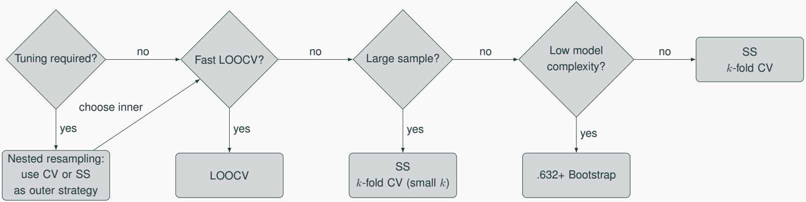
Cross validation (CV)

- Split D in B test datasets D_b^*
- Use $D_b = D \backslash D_b^*$ as training datasets
- D_b and D_b^* disjunct
- D_1 and D_2 not disjunct
- D_1^* and D_2^* disjunct
- Special case with $B = n$: Leave-one-out CV (LOOCV)
  - Small bias, high variance
  - Long runtime
- No optimal B, usually $B = 5, 10$
  - Slightly more bias than LOOCV, but lower variance
  - Lowest B of all resampling methods → fast computation

Stone 1974 J Roy Stat Soc B Met 36:111–47 • Molinaro et al. 2005 Bioinformatics 21:3301–07
Bischl et al. 2012 Evol Comput 20:249–75

Model evaluation and resampling

How to choose the resampling method?



Bischl et al. 2012 Evol Comput 20:249–75

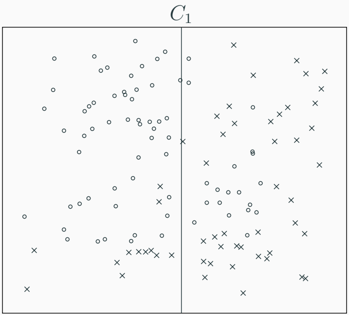
Part 6

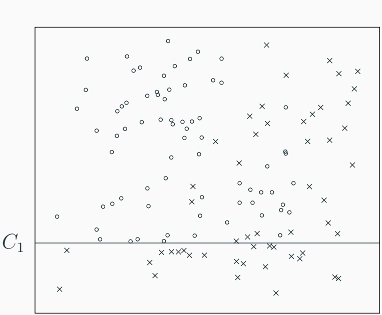
Boosting

Boosting stumps

What is boosting stumps?

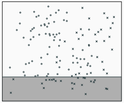
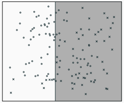
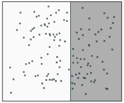
- Use 1-level decision tree, i.e., a stump
  Geometrically: only one horizontal or one vertical split

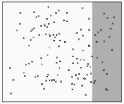
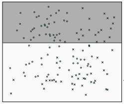
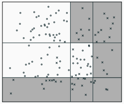




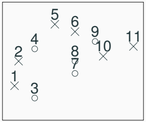
What is boosting stumps?

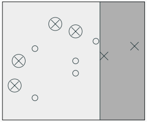
- Stump might not be able to classify all data points correctly
- If stumps are done repeatedly, classification might be improved
- Stumps often used as base learner in boosting

$H = \text{sign}(0.4 \times$  $+ 0.3 \times$  $+ 0.2 \times$  $+$

$0.34 \times$  $+ \cdots + 0.92 \times$  $) =$ 

What is discrete AdaBoost?

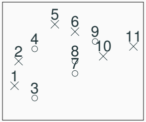


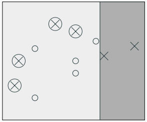


i	1	2	3	4	5	6	7	8	9	10	11	sum	err	alpha
$w_{i,1}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	1	$0.\overline{36}$	0.5596
$w_{i,2}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$			

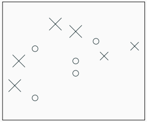
http://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/boosting.pdf

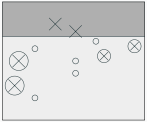
What is discrete AdaBoost?





i	1	2	3	4	5	6	7	8	9	10	11	sum	err	alpha
$w_{i,1}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	1	$0.\overline{36}$	0.5596
$w_{i,2}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$			

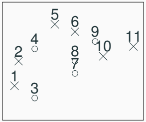


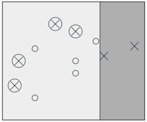


i	1	2	3	4	5	6	7	8	9	10	11	sum	err	alpha
$w_{i,2}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	1.27	0.3928	0.4356
$w_{i,3}$	0.2458	0.2458	$0.\overline{09}$	$0.\overline{09}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	0.1405	0.1405			

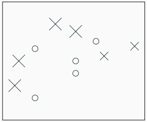
http://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/boosting.pdf

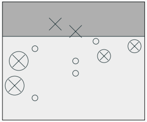
What is discrete AdaBoost?



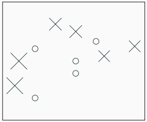


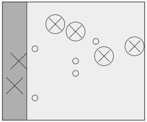
i	1	2	3	4	5	6	7	8	9	10	11	sum	err	alpha
$w_{i,1}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	1	$0.\overline{36}$	0.5596
$w_{i,2}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$			





i	1	2	3	4	5	6	7	8	9	10	11	sum	err	alpha
$w_{i,2}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	1.27	0.3928	0.4356
$w_{i,3}$	0.2458	0.2458	$0.\overline{09}$	$0.\overline{09}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	0.1405	0.1405			



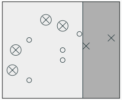


i	1	2	3	4	5	6	7	8	9	10	11	sum	err	alpha
$w_{i,3}$	0.2458	0.2458	$0.\overline{09}$	$0.\overline{09}$	0.159	0.159	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	0.1405	0.1405	1.5451	0.3877	0.457
$w_{i,4}$	0.2458	0.2458	$0.\overline{09}$	$0.\overline{09}$	0.2511	0.2511	$0.\overline{09}$	$0.\overline{09}$	$0.\overline{09}$	0.2219	0.2219			

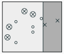
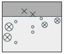
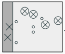
`http://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/boosting.pdf`

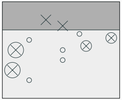
AdaBoost

What is discrete AdaBoost?

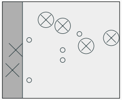


$$\frac{\text{alpha}}{0.5596}$$

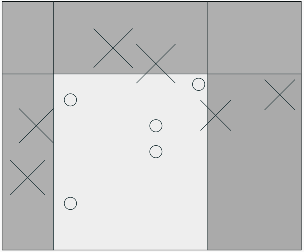
$H = (0.5596 \times$  $+ 0.4356 \times$  $+ 0.457 \times$  $)$



$$\frac{\text{alpha}}{0.4356}$$



$$\frac{\text{alpha}}{0.457}$$



http://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/boosting.pdf

AdaBoost

What is discrete AdaBoost?

1. Start with equal weights w_i for all subjects
2. For all boosting steps $b = 1$ to B do
   a) Fit a classifier $G_b(x)$ to the raining data using the above weights
   b) Compute error weights $err_b = \frac{1}{\sum_i w_i} \sum_{i:G_b(x_i) \neq y_i} w_i$
   c) Compute log odds of non error $\alpha_b = \ln(\frac{1-err_b}{err_b})$
   d) For all misclassified subjects increase weights using α_b
3. Output: weighted mean of classifiers

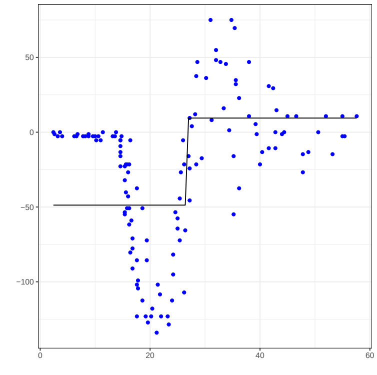
$$G(x) = \text{sign}\left(\sum_{b=1}^{B} \alpha_b G_b(x)\right)$$

Hastie et al. 2009 ISBN 9780387848570 • König et al. 2008 Int J Data Min Bioinf 2:289-341

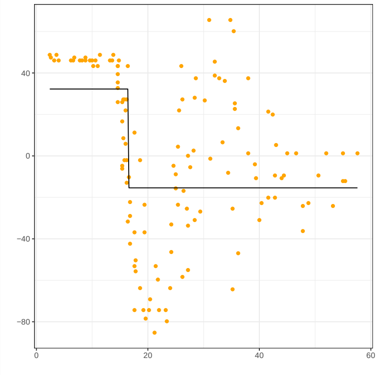
What is gradient boosting?

- Learn a dependent variable **using any learner**
- Compute residual (more general: any loss function)
- Learn the residual

Learn a simple model - here: a stump



Try to correct its error

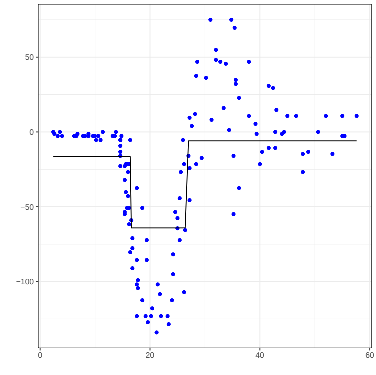


Ihler 2012 `https://www.youtube.com/watch?v=sRktKszFmSk`

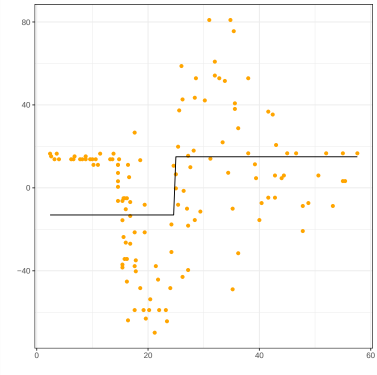
What is gradient boosting?

- Learn a dependent variable **using any learner**
- Compute residual (more general: any loss function)
- Learn the residual

Combining improves prediction



Try to correct its error once again

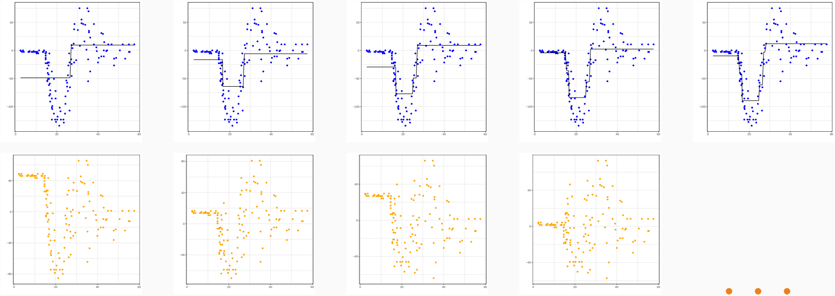


Ihler 2012 `https://www.youtube.com/watch?v=sRktKszFmSk`

What is gradient boosting?

- Learn a dependent variable **using any learner**
- Compute residual (more general: any loss function)
- Learn the residual

Data and **prediction** function and **residual**



Ihler 2012 `https://www.youtube.com/watch?v=sRktKszFmSk`

What are similarities and differences of AdaBoost and gradient boosting?

Identical fundamental idea

Boost performance of simple base-learner by iteratively shifting the focus towards problematic observations that are 'difficult' to predict

Difference

- AdaBoost: up-weight observations misclassified before
- Gradient boosting: difficult observations have large residuals

Mayr et al. 2014 Methods Inf Med 53:419-27

What are extensions of gradient boosting?

Component-wise boosting

Fit an additive model, e.g. regression

- Base learner: regression of single independent variable
- Select base learner with best fit among all single components, add to model

Likelihood boosting

Maximize likelihood with penalty term

Bühlmann & Yu 2003 J Am Stat Assoc 98:324-8 • Mayr et al. 2014 Methods Inf Med 53:419

Gradient boosting

What are the advantages of boosting?

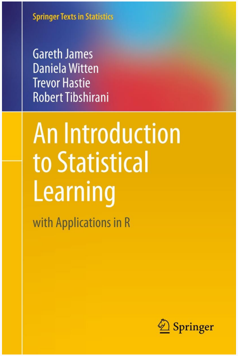
- High prediction accuracy
- Fast training
- Optimize any differentiable loss function
- Intrinsic variable selection

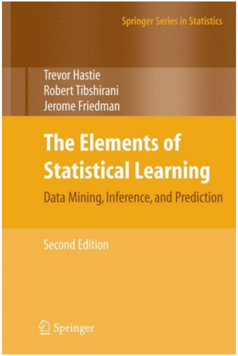
Gradient boosting

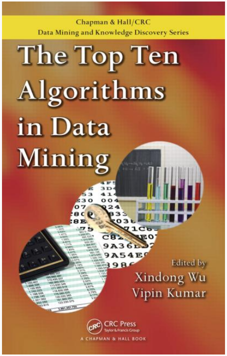
What are the disadvantages of boosting?

- Not simple to interpret (except component-wise boosting)
- Not easily parallelized (but possible)
- Prone to overfitting
- Requires proper parameter tuning

Literature







`https://www-bcf.usc.edu/~gareth/ISL/` (free!)

`https://web.stanford.edu/~hastie/ElemStatLearn/` (free!)

`http://www.crcpress.com/product/isbn/9781420089646`

Outline (Day 2)

Morning

7. Support vector machines
8. Hyperparameter tuning and performance comparison
9. Artificial neural networks

Afternoon

10. Specific endpoints
11. Variable importance and selection
12. Discussion

Part 7

Support vector machines

What are support vector machines?

Fundamental idea

- Find the hyperplane which best separates the data
- SVM similar to discriminant analysis, but SVM works without likelihood

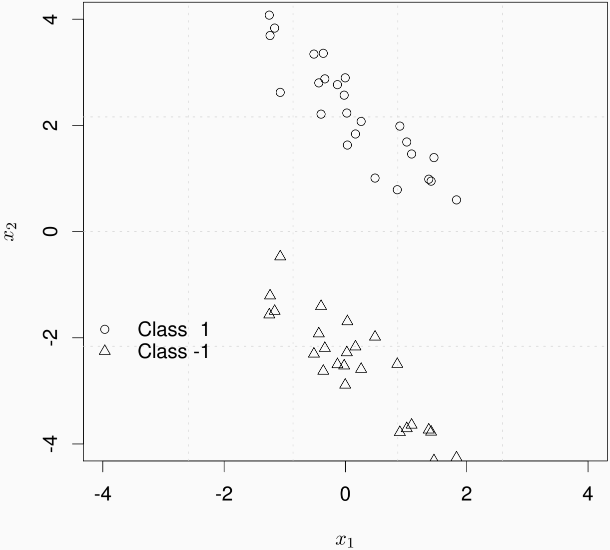
Warning

Mathematically most complex, not possible without formulae!

Schölkopf & Smola 2002 ISBN 9780262194754 ● Cristianini & Shawe-Taylor 2000 ISBN 9780521780193 ● Herbrich 2002 ISBN 9780262083065

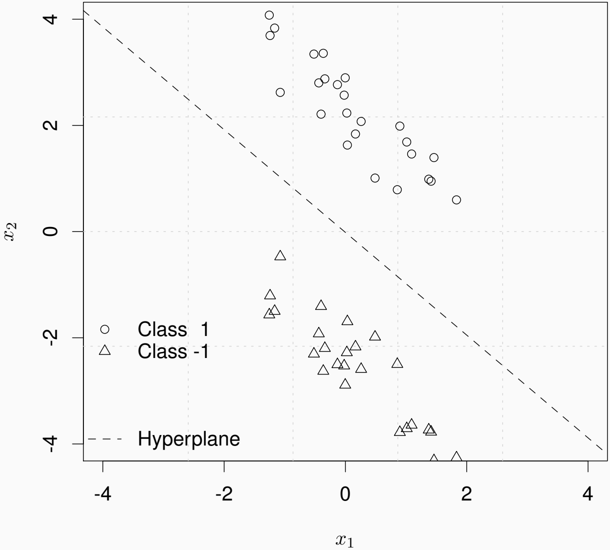
Support vector machines

What are support vector machines?



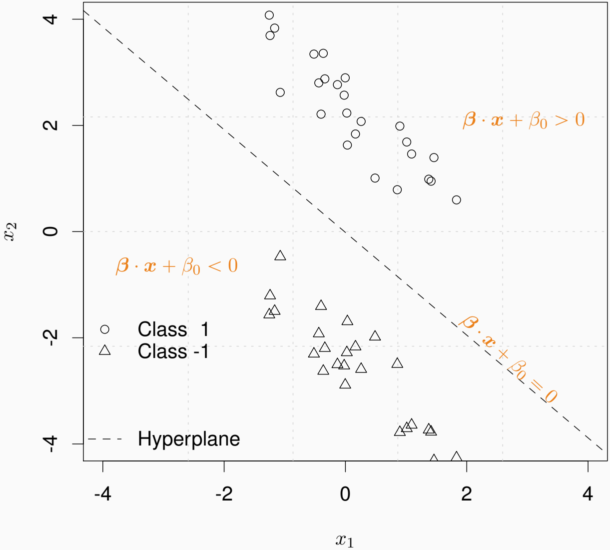
König et al. 2008 Int J Data Min Bioinf 2:289-341 • Fouodo et al. 2018 R J in press

What are support vector machines?



König et al. 2008 Int J Data Min Bioinf 2:289-341 • Fouodo et al. 2018 R J in press

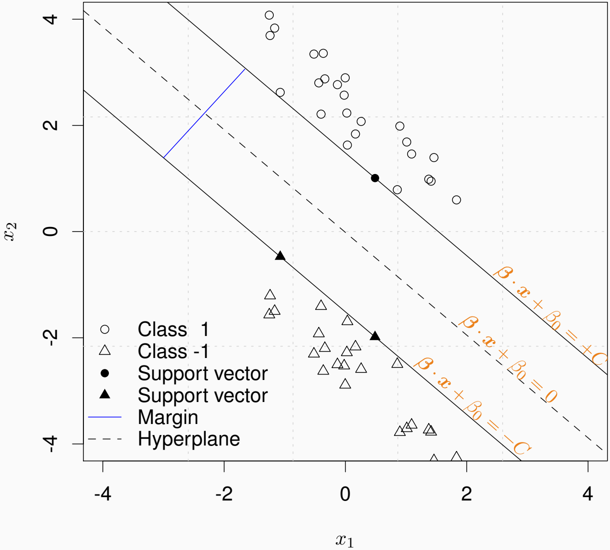
What are support vector machines?



König et al. 2008 Int J Data Min Bioinf 2:289-341 • Fouodo et al. 2018 R J in press

Support vector machines

What are support vector machines?



König et al. 2008 Int J Data Min Bioinf 2:289-341 • Fouodo et al. 2018 R J in press

What are support vector machines?

- Hyperplane given by $f(\boldsymbol{x}) = \boldsymbol{\beta} \cdot \boldsymbol{x} + \beta_0$
- Separation given by $y_i f(\boldsymbol{x}_i) > 0$ for $i = 1, ..., n$
- Number of Hyperplanes: **infinite**
- Uniqueness guaranteed by maximizing the **margin** C between groups
- Width of margin : $2C$
- Maximization problem:

$$max_{\boldsymbol{\beta}, \beta_0, ||\boldsymbol{\beta}||=1} C$$

subject to $y_i(\boldsymbol{\beta} \cdot \boldsymbol{x}_i + \beta_0) \geqslant C, \quad i = 1, ..., n$

König et al. 2008 Int J Data Min Bioinf 2:289-341

Support vector machines

What are support vector machines?

- Maximization problem:

$$max_{\boldsymbol{\beta},\beta_0,||\boldsymbol{\beta}||=1} C$$

  subject to $y_i(\boldsymbol{\beta} \cdot \boldsymbol{x}_i + \beta_0) \geqslant C, \quad i = 1, ..., n$
  is equivalent to

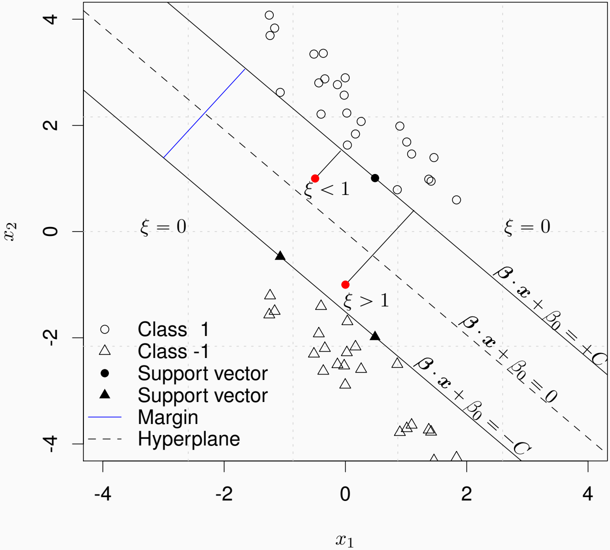
$$max_{\boldsymbol{\beta},\beta_0} \boldsymbol{\beta}$$

  subject to $y_i\left(\boldsymbol{\beta} \cdot \boldsymbol{x}_i + \beta_0\right) \geqslant C, \quad i = 1, ..., n$
- Classification rule for an observation: $\boldsymbol{x}^*$

$$\operatorname{sign}\left(f\left(\boldsymbol{x}^*\right)\right) = \operatorname{sign}\left(\boldsymbol{\beta} \cdot \boldsymbol{x}^* + \beta_0\right)$$

König et al. 2008 Int J Data Min Bioinf 2:289-341

What are support vector machines?



König et al. 2008 Int J Data Min Bioinf 2:289-341 • Fouodo et al. 2018 R J in press

Support vector machines

What are support vector machines?

- Slack variables ξ_i
- Subject on correct side of margin $\Rightarrow \xi_i = 0$
- Subject on correct side, but within margin $\Rightarrow 0 < \xi_i < 1$
- Subject on wrong side of the margin $\Rightarrow \xi_i \geqslant 1$

Restriction: $\sum_{i=1}^{n} \xi_i \leqslant \gamma$

- Tuning Parameter γ
- Proportion of predictions on the wrong side of the margin

König et al. 2008 Int J Data Min Bioinf 2:289-341

Support vector machines

What are support vector machines?

Maximization problem:

$$max_{\boldsymbol{\beta},\beta_0,\|\boldsymbol{\beta}\|=1} C$$

subject to $y_i\left(\boldsymbol{\beta} \cdot \boldsymbol{x}_i + \beta_0\right) \geqslant 1 - \xi_i$
and

$$\xi_i \geqslant 0$$

$$\sum_{i=1}^{n} \xi_i \leqslant \gamma \quad i = 1, ..., n$$

König et al. 2008 Int J Data Min Bioinf 2:289-341

Support vector machines

What are support vector machines (SVMs)?

Idea for solving optimization problem

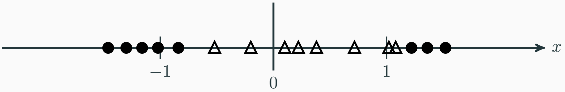
- Quadratic problem with linear inequalities
  → convex optimization problem
- Simplify optimization problem by transfer to the dual
- Together with Karush-Kuhn-Tucker conditions
  → unique solution

Advantages of the dual form

- Does not involve $\boldsymbol{\beta}$
- Involves maximization problem with inner product $\boldsymbol{x}_i \cdot \boldsymbol{x}_j = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$
- Quadratic programming solver with Lagrange multipliers

König et al. 2008 Int J Data Min Bioinf 2:289-341

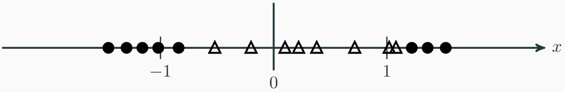
What is the kernel trick?

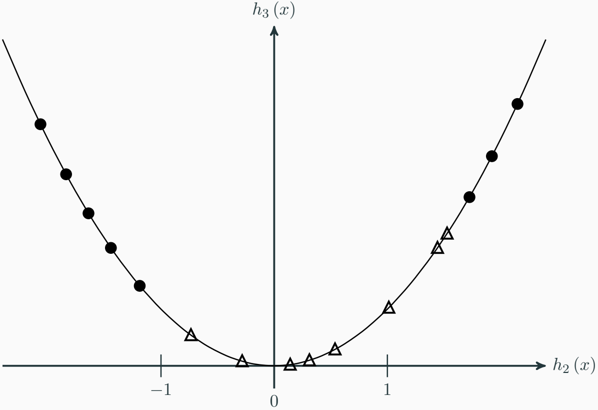


König et al. 2008 Int J Data Min Bioinf 2:289-341

Support vector machines

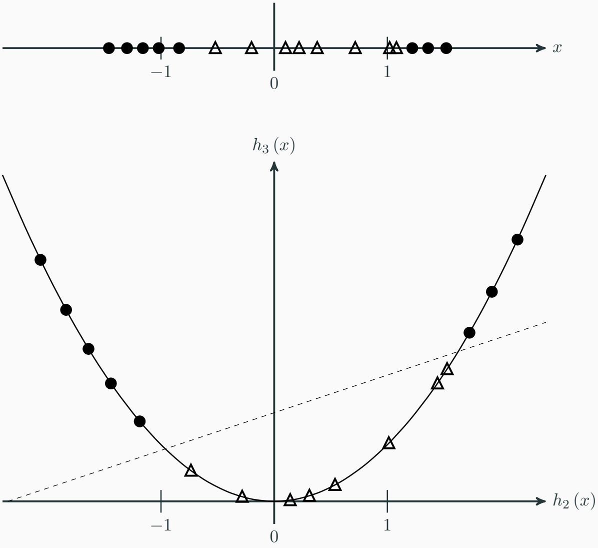
What is the kernel trick?





König et al. 2008 Int J Data Min Bioinf 2:289-341

What is the kernel trick?



König et al. 2008 Int J Data Min Bioinf 2:289-341

Support vector machines

What is the kernel trick?

- Separation by hyperplane impossible
- BUT separation by hyperplane might be possible in higher dimensional space → feature space $\boldsymbol{h}(\boldsymbol{x})$
- Separating hyperplane given by

$$\boldsymbol{\beta} \cdot \boldsymbol{h}(\boldsymbol{x}) + \beta_0$$

Disadvantage

- Drastic increase in CPU time

Kernel trick

- Reduction of CPU time by using specific kernel functions

König et al. 2008 Int J Data Min Bioinf 2:289-341

Support vector machines

What is the kernel trick?

Solution of the optimization problem f for new observation $\boldsymbol{x}^*$

$$\begin{aligned}\operatorname{sign}\left(f\left(\boldsymbol{x}^*\right)\right) &= \operatorname{sign}\left(\boldsymbol{\beta}\cdot\boldsymbol{x}^* + \beta_0\right) \\ &= \operatorname{sign}\left(\sum_{i=1}^{n}\alpha_i\, y_i\, \boldsymbol{x}_i\cdot\boldsymbol{x}^* + \beta_0\right) \\ &= \operatorname{sign}\left(\sum_{i=1}^{n}\alpha_i\, y_i\, \langle\boldsymbol{x}^*, \boldsymbol{x}_i\rangle + \beta_0\right)\end{aligned}$$

With feature space as input space

$$\operatorname{sign}\left(f\left(\boldsymbol{x}^*\right)\right) = \operatorname{sign}\left(\sum_{i=1}^{n}\alpha_i\, y_i\, \langle\boldsymbol{h}\left(\boldsymbol{x}^*\right), \boldsymbol{h}\left(\boldsymbol{x}_i\right)\rangle + \beta_0\right)$$

König et al. 2008 Int J Data Min Bioinf 2:289-341

Support vector machines

What is the kernel trick?

Trick: choose kernel function to evaluate

$$K\left(\boldsymbol{x}^{*}, \boldsymbol{x}_{i}\right)=\left\langle\boldsymbol{h}\left(\boldsymbol{x}^{*}\right), \boldsymbol{h}\left(\boldsymbol{x}_{i}\right)\right\rangle$$

without explicit calculation of $h\left(\cdot\right)$

König et al. 2008 Int J Data Min Bioinf 2:289-341

What is the kernel trick?

Standard properties of kernel functions $K(\cdot, \cdot)$

- Symmetric
- Positive definite

What is the kernel trick?

Classical choices of kernel functions

- Inhomogeneous polynomial kernel
  $K(\boldsymbol{x}^*, \boldsymbol{x}_i) = (c + \langle \boldsymbol{x}^*, \boldsymbol{x}_i \rangle)^b$ with $b \in \mathbb{N}$ and $c \geqslant 0$
- Sigmoid kernel (artifical neural network kernel)
  $K(\boldsymbol{x}^*, \boldsymbol{x}_i) = \tanh(\kappa \langle \boldsymbol{x}^*, \boldsymbol{x}_i \rangle + \upsilon)$ with $\kappa > 0$ and $\upsilon < 0$
- Gaussian radial kernel
  $K(\boldsymbol{x}^*, \boldsymbol{x}_i) = \exp\left(\frac{1}{2\sigma^2} \|\boldsymbol{x}^* - \boldsymbol{x}_i\|\right)$, with $\sigma^2 > 0$
- Bessel kernel
  $K(\boldsymbol{x}^*, \boldsymbol{x}_i) = -\operatorname{Bessel}^n_{\nu+1}\left(\sigma \|\boldsymbol{x}^* - \boldsymbol{x}_i\|^2\right)$

König et al. 2008 Int J Data Min Bioinf 2:289-341

How can SVMs be used for probability estimation?

Approach 1

Fit parametric sigmoid link function between conditional probability and SVM classification function

$\Rightarrow$ Imposes parametric assumptions, probably unrealistic for applications

Kruppa et al. 2014 Biom J 56:534-63 • Platt 2000 ISBN 9780262194488, p. 61-74 • Vapnik 1998 ISBN 9780471030034

Support vector machines

How can SVMs be used for probability estimation?

Approach 2

- SVM consistent estimation of Bayes rule
- Weighted SVM allows consistent estimation of probabilities at other thresholds

Concept

- Fit weighted SVM classifiers
- SVM estimates correspond to point probabilities
- **Bracket probability**: observation exactly between two SVM estimates
- Average probability of lower and upper limits

Kruppa et al. 2014 Biom J 56:534-63 ● Lin 2002 Data Min Knowl Discov 6:259-75 ● Wang et al. 2008 Biometrika 95:149-167

What are the statistical properties of SVMs?

- Summary of findings: Moguerra & Muñoz
- Consistency for binary SVM: Cucker & Smale
- Consistency of multicategory classification SVM: Doğan et al.
- Convergence rate:
  - Surrogate loss: Zhang
  - Hinge loss: Bartlett et al.
  - Gaussian kernel binary SVM: Steinwart & Scovel
  - Multicategory: Zhang, Wang & Shen, Zhang & Liu
  - Probability SVM: Wang et al.
  - Multicategory probability SVM: Not aware of any result

Bartlett et al. 2006 J Am Stat Assoc 101:138-56 • Cucker & Smale 2002 Bull Am Math Soc 39:1-49 • Doğan et al. 2016 J Mach Learn Res 17:1-32 • Moguerra & Muñoz 2006 Stat Sci 21:322-36 • Steinwart & Scovel 2007 Ann Stat 35:575-607 • Wang & Shen 2007 J Am Stat Assoc 102:595-602 • Wang et al. 2008 Biometrika 95:149-67 • Wu et al. 2010 J Am Stat Assoc 105:424-36 • Zhang 2004 Ann Stat 32:56-85 • Zhang & Liu 2013 J Mach Learn Res 14:1349-86

What are the advantages of SVMs?

- Work well with high dimensional data
- Work well on non-linear data (kernel trick)
- Simple handling of outliers
- Beautiful theory

Support vector machines

What are the disadvantages of SVMs?

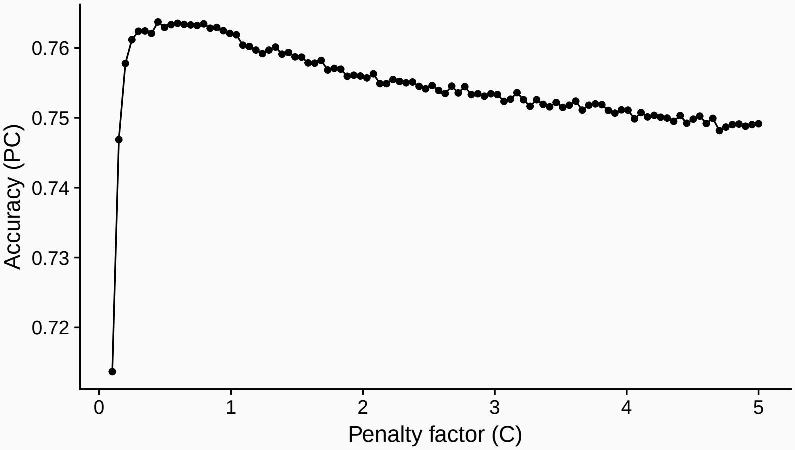
- Not simple to interpret
- Difficult to guess good kernel on real data
- Require proper hyperparameter tuning
- Computationally intensive on large data sets
- Optimized for binary classification and numerical features

Part 8

Hyperparameter tuning and performance comparison

Hyperparameter tuning

Why is parameter tuning necessary?



Hyperparameter tuning

What is parameter tuning?

Hyperparameters

Learners have hyperparameters, e.g.:

- Number of nearest neighbors k
- Depth of a tree
- Number of features to consider in each split of a random forest (mtry)
- Number of boosting iterations
- Kernel of SVM

Most learners have several hyperparameters

Have to be jointly optimized

Hyperparameter tuning

What is parameter tuning?

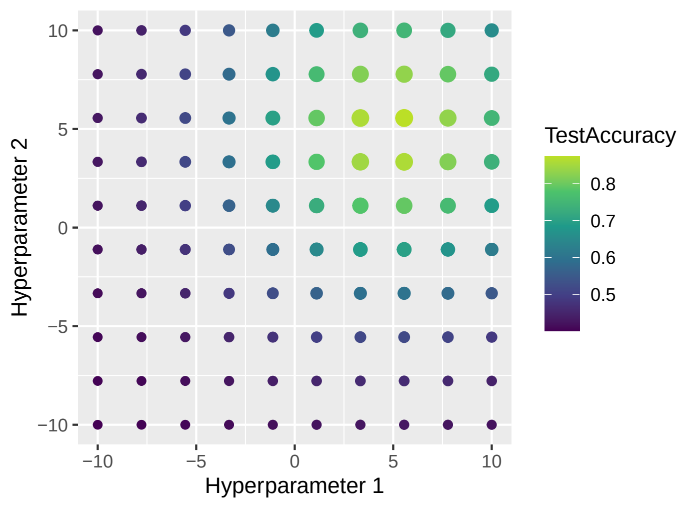
Search entire parameter space

- All possible combinations
- Grid search
- Randomly select combinations
- Model-based optimization

Use resampling

- Evaluate each parameter combination on all resampling iterations/folds
- Choose parameter maximizing aggregated performance measure

What is grid search?



Hyperparameter tuning

Grid search

Advantages

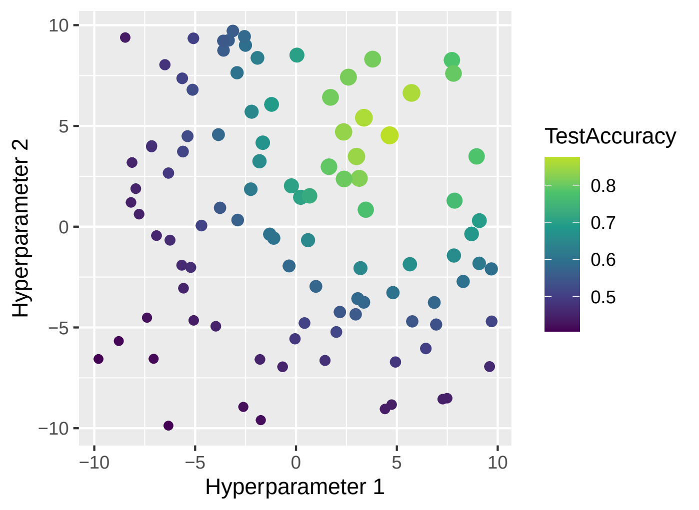
- Easy to implement
- All parameter types possible
- Easily parallelized

Disadvantages

- Computationally intensive
- Inefficient: Searches large irrelevant areas
- Arbitrary: Which values / discretization?

Hyperparameter tuning

What is random search?



Hyperparameter tuning

Random search

Advantages

- Same as grid search: Easy to implement, all parameter types possible, trivial parallelization
- Easy to adjust to computational budget
- No discretization
- Superior performance compared to grid search

Disadvantages

- Computationally intensive
- Inefficient: Searches large irrelevant areas

What is model-based optimization?

Surrogate model

Learn relationship between hyperparameters and prediction performance

Algorithm

1. Pick initial configuration (e.g. random)
2. Learn surrogate model
3. Predict new configuration with surrogate model
4. Repeat steps 2 and 3

Hyperparameter tuning

Model-based optimization

Advantages

- All parameter types possible
- Efficient: Focus on promising areas
- Superior performance compared to grid and random search

Disadvantages

- Computationally intensive
- Non-trivial parallelization
- Harder to implement

Turner et al. 2020 NeurIPS PMLR 133:3

Performance comparison

How can performance be compared?

Be fair!

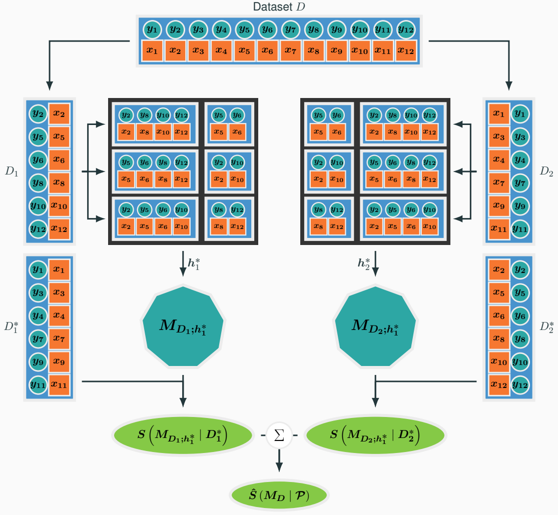
- Compare all learners and models on same data
- Tune parameters of all learners
- Don’t overfit
- Don’t publish over-optimistic results

Never learn, tune or evaluate on same data!

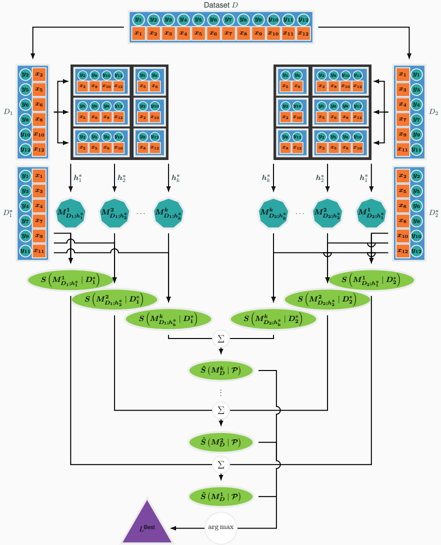
Performance comparison

How can performance be compared?

Nested resampling



How can performance be compared?



How to build a final model?

1. Select best learner with nested resampling
2. Find optimal hyperparameters of best learner with resampling
3. Train best learner with optimal hyperparameters on full data

Bischl et al. 2012 Evol Comput 20:249–75

Performance comparison

How can differences in performance be statistically tested?

One dataset

- McNemar test for simple holdout validation
- 5×2 CV test
- Corrected t-test for (repeated) CV

Several datasets

- 2 machines: Wilcoxon signed rank test
- Several machines: Friedman test

Dietterich 1998 Neural Comput 10:1895 • Goldstein et al. 2016 Int J Biostat 12:117
Nadeau & Bengio 2003 Mach Learn 52:239 • Demšar 2006 J Mach Learn Res 7:1

Performance comparison

How can differences in performance be statistically tested?

Problems

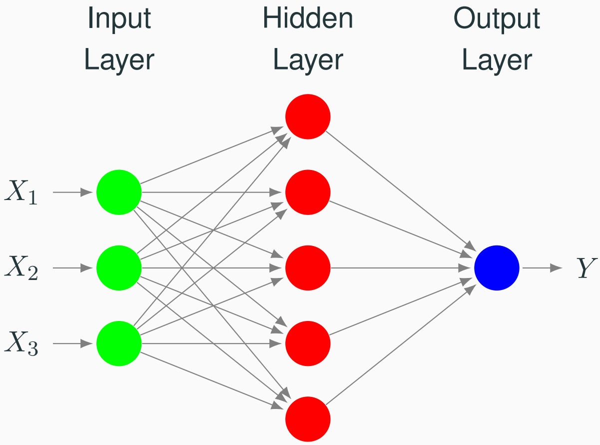
- Same as statistical vs. clinical relevance
- Many other parameter to consider
    - number of hyperparameters
    - number of variables
    - runtime
    - ...

$\Rightarrow$ **Practical relevance?**

Part 9

Artificial neural networks

What are artificial neural networks?



What are artificial neural networks?

For each hidden unit:

Activation function

Unit input

$$z = \sigma(\mathbf{w} \cdot \mathbf{x} + b)$$

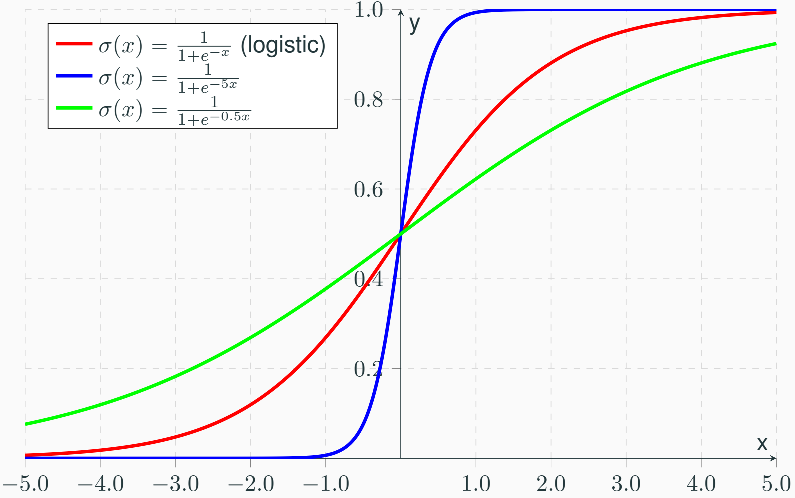
Unit output

Bias

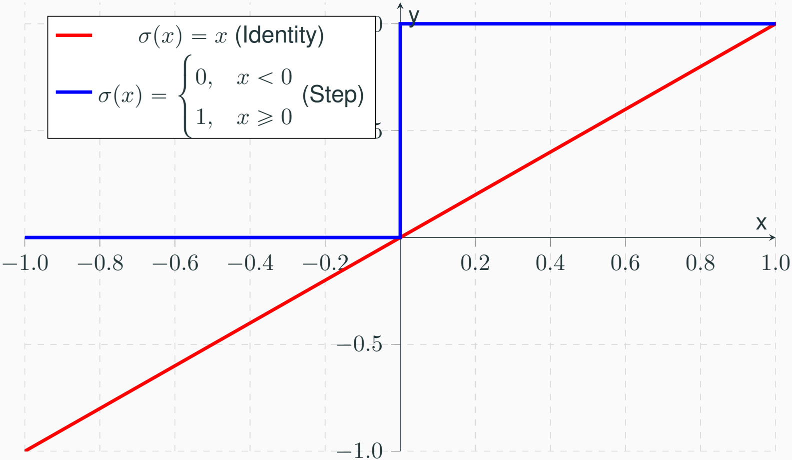
Weights

What is an activation function?

Typical choice: Sigmoid (e.g. logistic)



What are special cases of activation functions?



Identity: Linear model

Step: Perceptron

Artificial neural networks

How to fit a neural network?

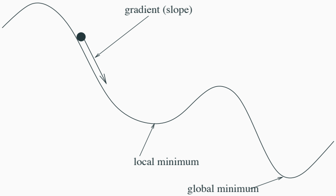
Loss function: Error as function of network weights, e.g,

$$L(\mathcal{W}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Aim: Find weights that minimize error

Gradient descent

Adjust weights in direction with steepest descent

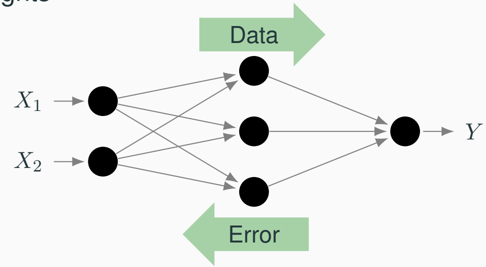


http://bengio.abracadoudou.com/lectures/old/tex_ann.pdf

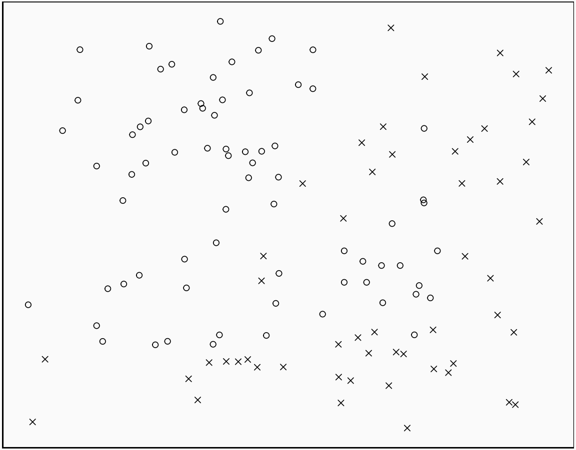
How to fit a neural network?

Backpropagation

1. Initialize weights randomly
2. For k iterations repeat
   a) Compute error function
   b) Adjust weights in output layer
   c) Propagate error backwards through network and adjust weights

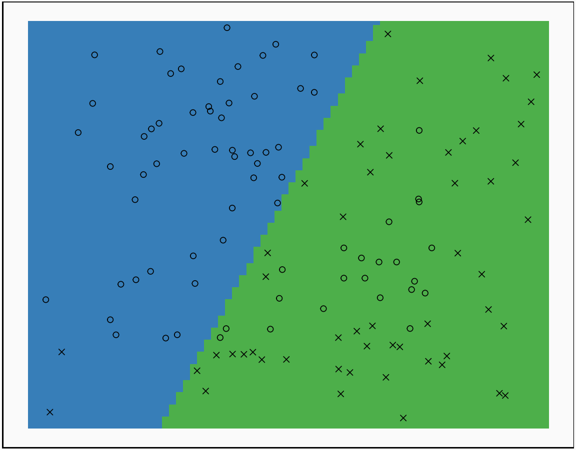


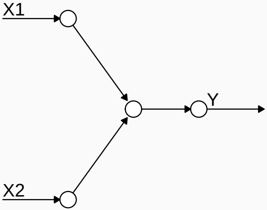
Do neural networks overfit?



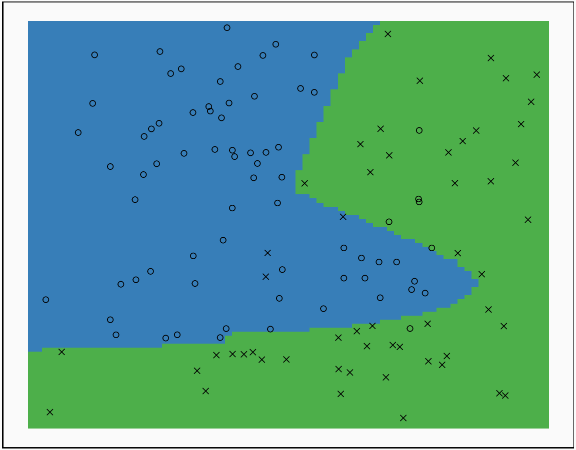
Artificial neural networks

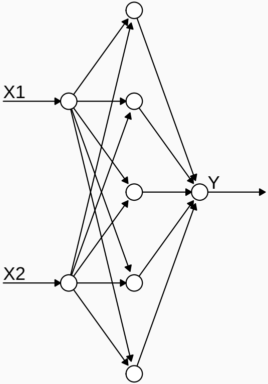
Do neural networks overfit?



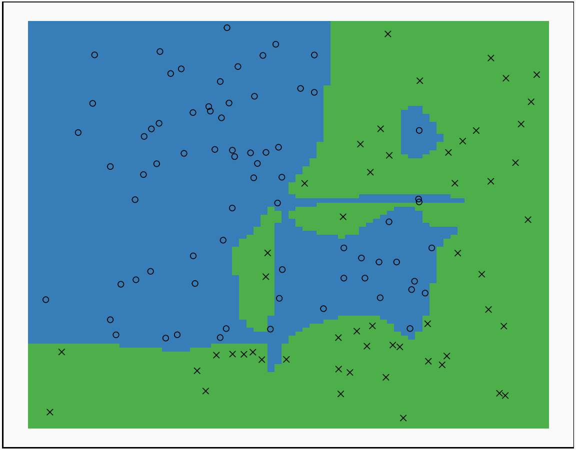


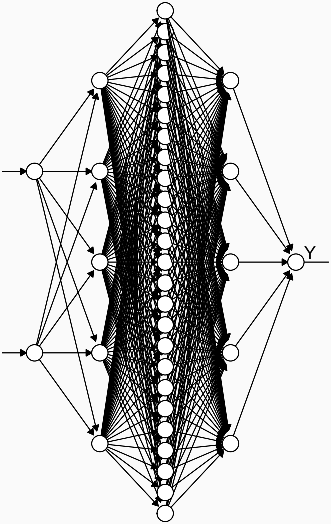
Do neural networks overfit?





Do neural networks overfit?





What is regularization?

L2 regularization

$$L(\mathcal{W}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{\mathbf{w}\in\mathcal{W}} \mathbf{w}^2$$

Penalty term

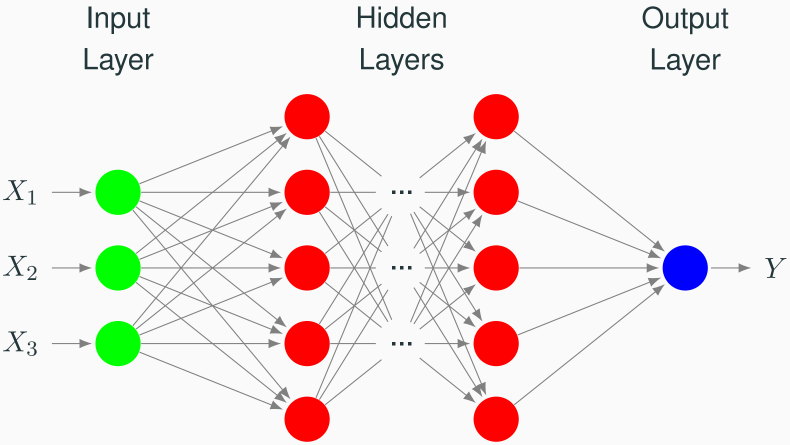
Alternative: Dropout

Temporarily remove units while fitting

Other alternative: Early stopping

Artificial neural networks

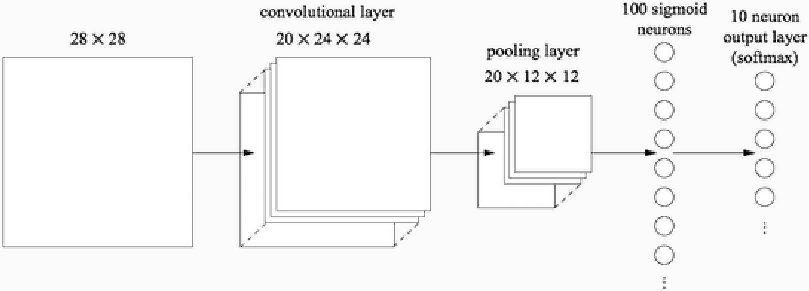
What is deep learning?



What is deep learning?

Idea: Build hierarchy of concepts (representations)

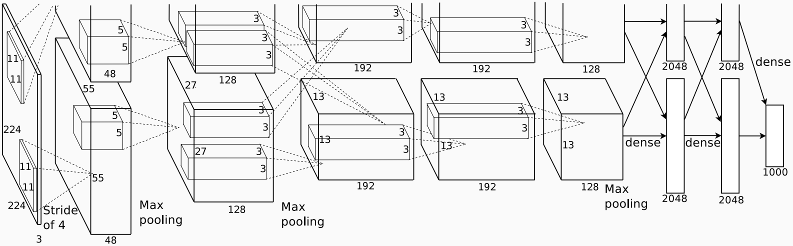
Convolutional neural networks



http://neuralnetworksanddeeplearning.com/chap6.html

What is deep learning?

Example



Krizhevsky et al. 2012 NIPS 25:1097

Artificial neural networks

What are the advantages of neural networks?

- Can fit any complicated function almost perfectly
- Learns representations
- Online learning possible
- Prediction very fast (matrix multiplication)
- Can be parallelized (GPU computing)

Artificial neural networks

What are the disadvantages of neural networks?

- Prone to overfitting
- Difficult to design good networks
- Interpretation very difficult (black box)
- Learning can be slow
- Statistical properties not well studied

Part 10

Specific endpoints

Machine learning for multi-category (multi-class) endpoints

How can kNN be used with multi-category outcome?

- Algorithms unchanged
- Classification: most frequent category among kNN
- Probability estimation: estimate proportions of categories among kNN

Kruppa et al. 2014 Biom J 56:534-63

Machine learning for multi-category (multi-class) endpoints

How can random forests be used with multi-category outcome?

- Random forest algorithm unchanged
- Classification: use of extension of impurity measure to ordered and unordered multi-category outcome
  - Entropy $\rightarrow$ straightforward
  - Gini Index $\sum_{c=1}^{C} p_c(1 - p_c) = 1 - \sum_{c=1}^{C} p_c^2$
- Probability estimation: analogously to classification with Gini Index as criterion

Breiman et al. 1984 ISBN 9780412048418, 103 • Kruppa et al. 2014 Biom J 56:534-63

Machine learning for multi-category (multi-class) endpoints

How can boosting be used with multi-category outcome?

AdaBoost

Essentially the same, but use base learner better than C-class random guessing, i.e., accuracy better than $1/C$.

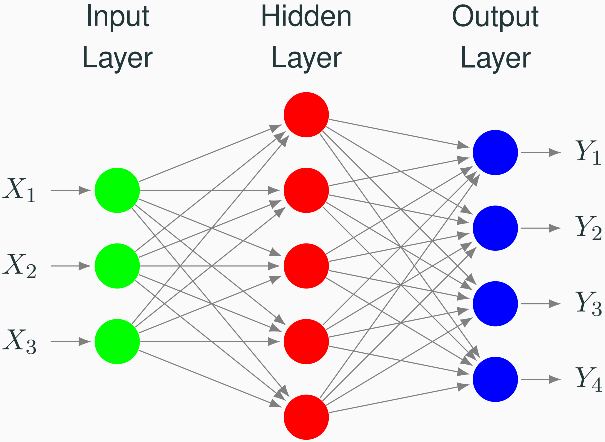
Gradient boosting

- Use base learner supporting multi-category outcomes
- Use e.g. multinomial loss function

Zhu 2009 Stat Interface 2:349–60

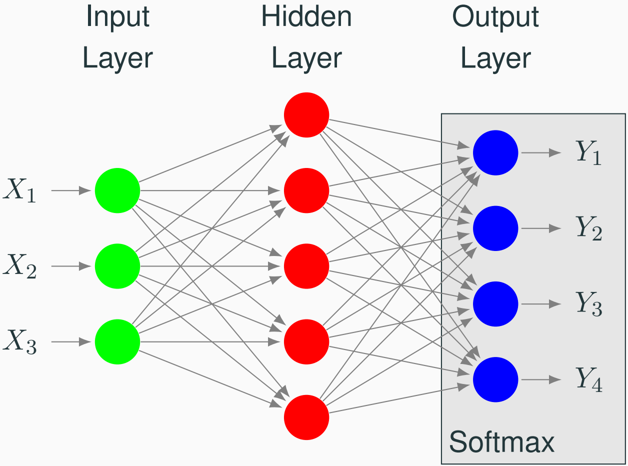
Machine learning for multi-category (multi-class) endpoints

How can neural networks be used with multi-category outcome?



Machine learning for multi-category (multi-class) endpoints

How can neural networks be used with multi-category outcome?



Machine learning for multi-category (multi-class) endpoints

How can neural networks be used with multi-category outcome?

- One output unit per class
- Use softmax activation function in output layer:

$$\sigma(x)_j = \frac{e^{x_j}}{\sum_{c=1}^{C} e^{x_c}}, j = 1, \ldots, C$$

- Outputs $Y_1, \ldots, Y_C$ guaranteed to sum to 1

Machine learning for multi-category (multi-class) endpoints

How can support vector machines be used with multi-category outcome?

Two approaches in case of C classes

1. One versus all(OVA)
   a) Fit c different 2-class SVM classifiers $\hat{f}_c(x)$
   b) Each class versus rest
   c) Classify x to the class c for which $\hat{f}_c(x)$ is largest
2. One versus one (OVO) ”pairwise coupling approach”
   a) Fit all $\binom{C}{2}$ pairwise classifiers $\hat{\hat{f}}_{cc'}(x)$
   b) Classify x to the class that wins the most pairwise competitions

If C is not to large, use OVO.

Doğan et al. 2016 J Mach Learn Res 17:1-32

Machine learning for multi-category (multi-class) endpoints

How can support vector machines be used for probability estimation in the multi-category case?

Approach 1

Step 1 OVO aaproach (pairwise coupling)

Step 2 Combine probability estimates

Approach 2

Extend bracketing approach to multicategory probability estimation

Approach 3

Re-fit specific Large-margin Unified Machine (LUMs)

Kruppa et al. 2014 Biom J 56:534-63 • Liu et al. 2011 J Am Stat Assoc 106:166-77
Wu et al. 2004 J Mach Learn Res 5:975-1005 • Wu et al. 2010 J Am Stat Assoc 105:424-36

Machine learning for multi-category (multi-class) endpoints

Do the statistical properties of binary classification problems automatically generalize to the multi-category setting?

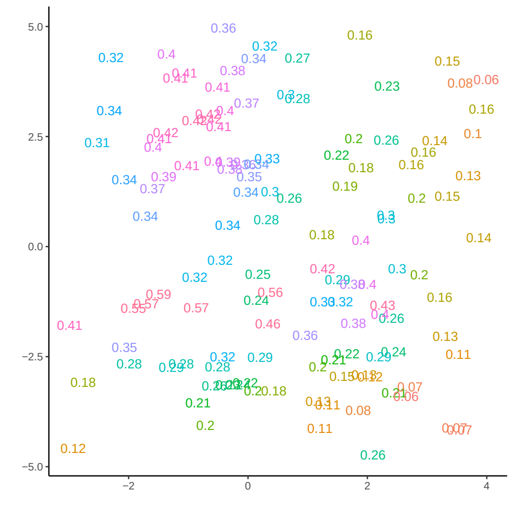
- No

- No explicit proofs available for kNN, random forests
- Erroneous proof available for support vector machines
- Sufficient and necessary conditions available for consistency and wide class of models

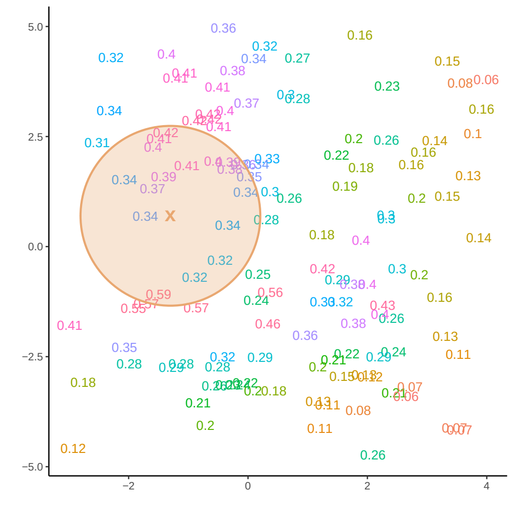
Tewari & Bartlett 2007 J Mach Learn Res 8:1007-25

Machine learning for continuous endpoints

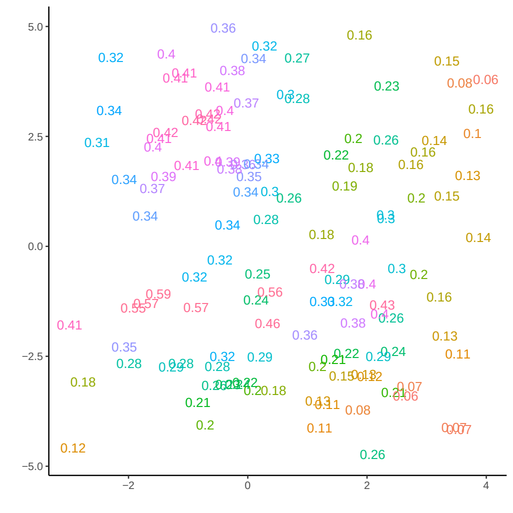
How can continuous endpoints be analyzed with kNN?



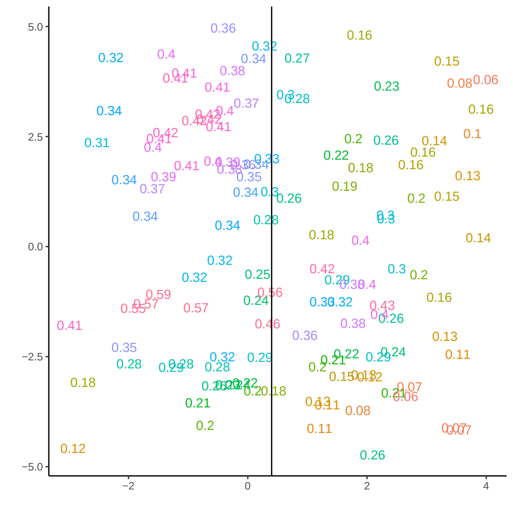
How can continuous endpoints be analyzed with kNN?



How can continuous endpoints be analyzed with random forests?

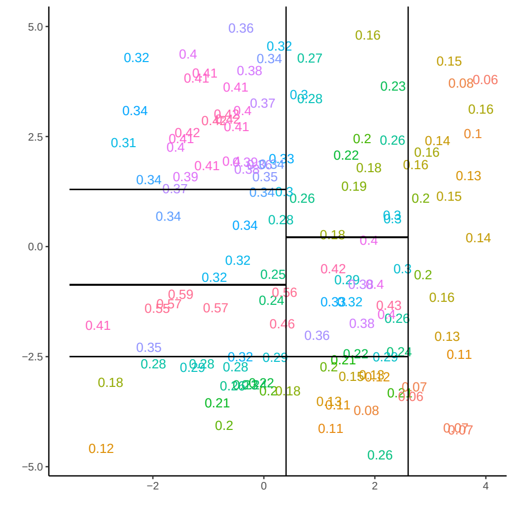


How can continuous endpoints be analyzed with random forests?

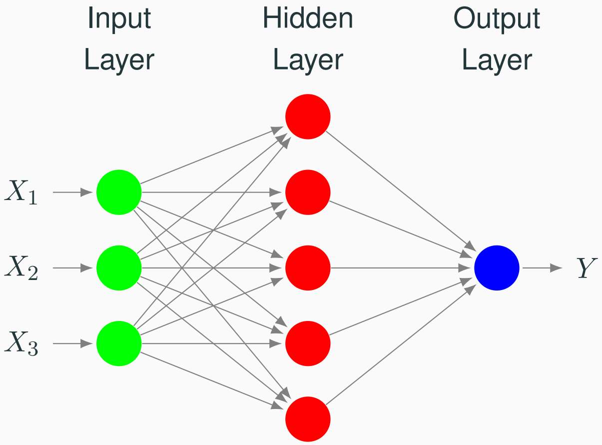


Machine learning for continuous endpoints

How can continuous endpoints be analyzed with random forests?



How can continuous endpoints be analyzed with neural networks?



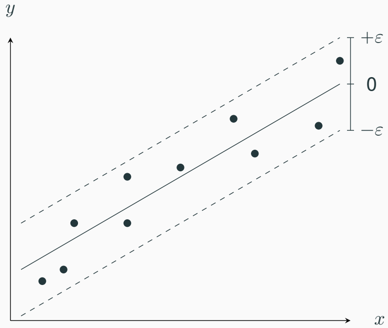
Machine learning for continuous endpoints

How can continuous endpoints be analyzed with neural networks?

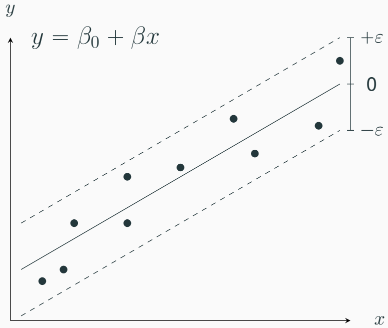
- One output unit
- Use identity activation function in output layer:

$$\sigma\left(x\right) = x$$

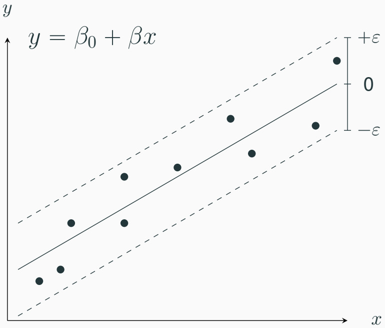
How can continuous endpoints be analyzed with support vector regression?



How can continuous endpoints be analyzed with support vector regression?



How can continuous endpoints be analyzed with support vector regression?



Solution:

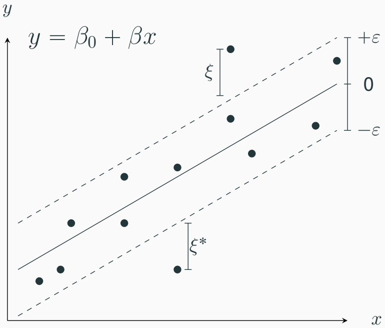
$$\min \frac{1}{2} \|\boldsymbol{\beta}\|^2$$

Constraints:

$$y_i - \beta_0 - \boldsymbol{\beta} \cdot \boldsymbol{x}_i \leqslant \epsilon$$

$$\beta_0 + \boldsymbol{\beta} \cdot \boldsymbol{x}_i - y_i \leqslant \epsilon$$

How can continuous endpoints be analyzed with support vector regression?



Solution:

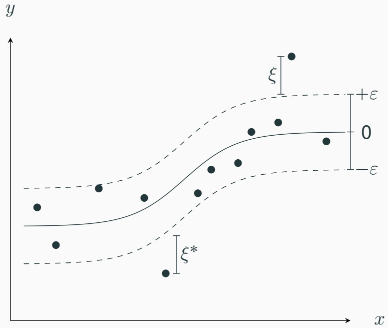
$$\min \frac{1}{2} \|\boldsymbol{\beta}\|^2 + C \sum_{i=1}^{n} (\xi_i + \xi_i^*)$$

Constraints:

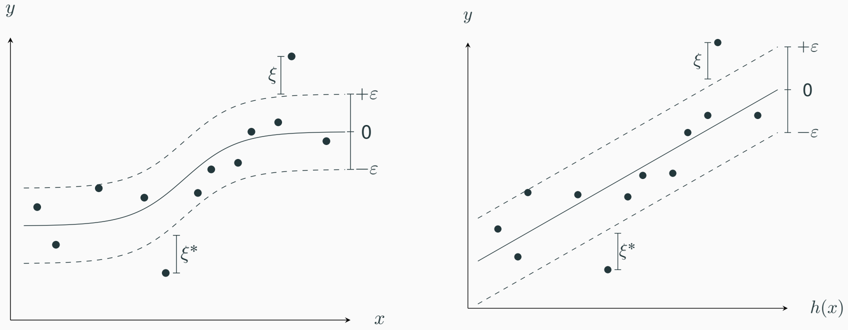
$$y_i - \beta_0 - \boldsymbol{\beta} \cdot \boldsymbol{x}_i \leqslant \epsilon + \xi_i$$

$$\beta_0 + \boldsymbol{\beta} \cdot \boldsymbol{x}_i - y_i \leqslant \epsilon + \xi_i^*$$

How can continuous endpoints be analyzed with support vector regression?



How can continuous endpoints be analyzed with support vector regression?



Machine learning for continuous endpoints

How can continuous endpoints be analyzed with support vector regression?

Linear kernel support vector regression

$$y = \beta_0 + \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) \langle \boldsymbol{x}_i, \boldsymbol{x}_i^* \rangle$$

Non-Linear kernel support vector regression

$$y = \beta_0 + \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) \langle \boldsymbol{h}(\boldsymbol{x}_i), \boldsymbol{h}(\boldsymbol{x}_i^*) \rangle$$

$$y = \beta_0 + \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) K(\boldsymbol{x}_i, \boldsymbol{x}_i^*)$$

Smolua & Schölkopf 2004 Stat Comput 14:199-222 • Drucker et al. 1997 ISBN 9780262100656, pp. 155–61

How can nearest neighbor methods be used for survival analysis?

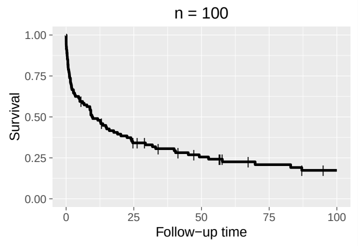
Algorithm unchanged

Estimate function of interest using kNN, e.g.,

- Kaplan-Meier estimator
- Nelson-Aalen estimator
- Median survival time
- ...

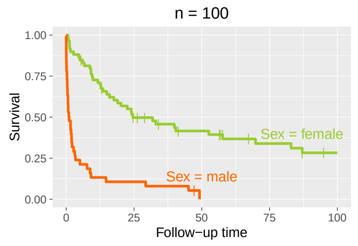
Machine learning for survival endpoints

How can random forests be used for survival analysis?

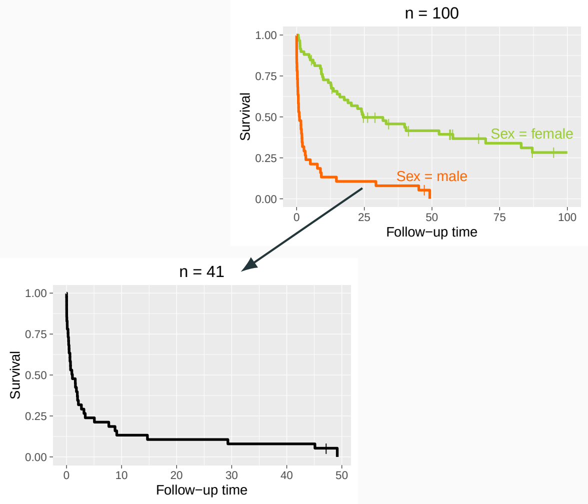


Machine learning for survival endpoints

How can random forests be used for survival analysis?

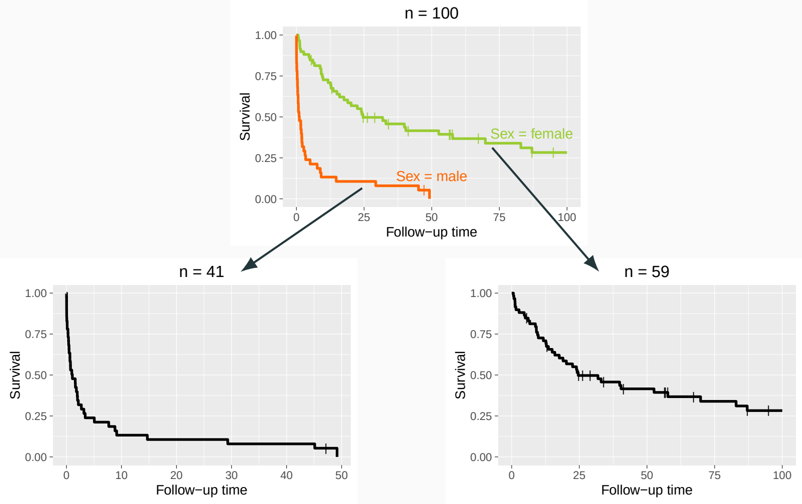


How can random forests be used for survival analysis?

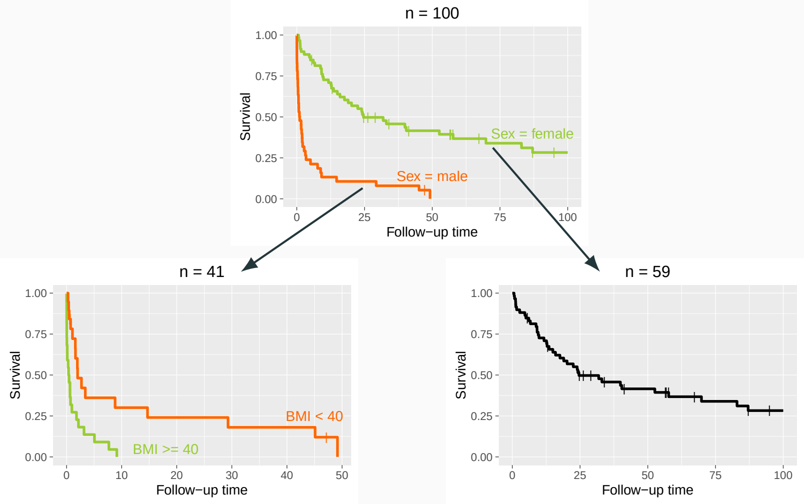


Machine learning for survival endpoints

How can random forests be used for survival analysis?

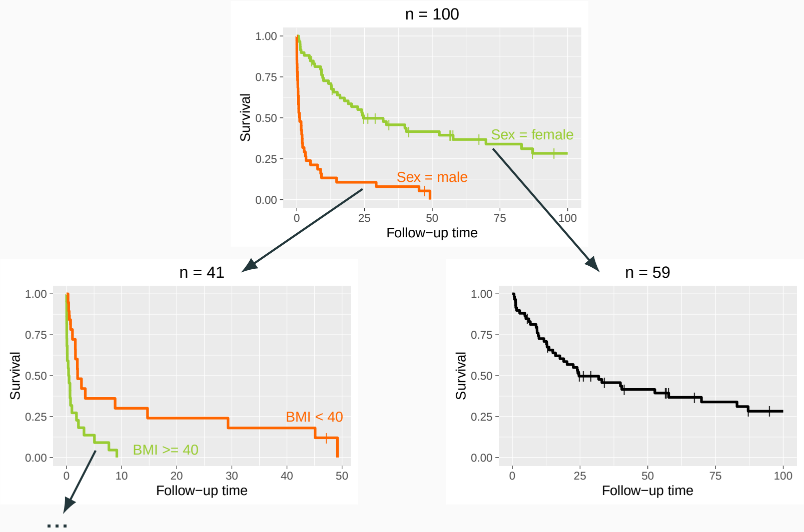


How can random forests be used for survival analysis?

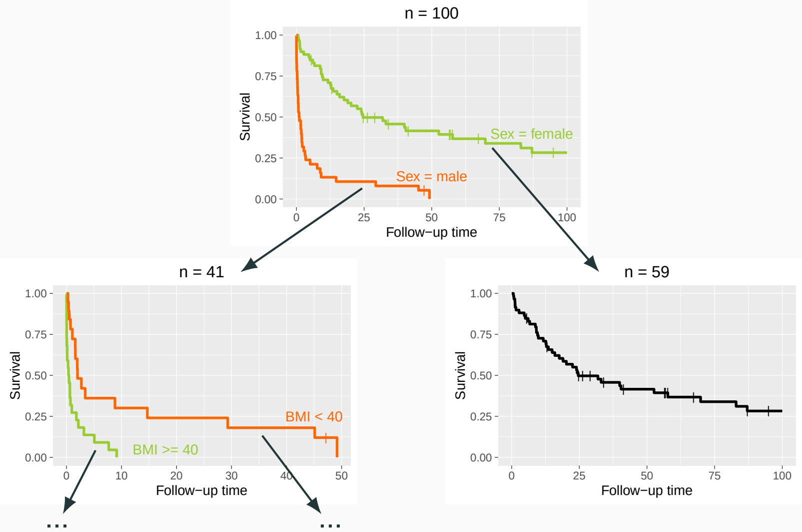


Machine learning for survival endpoints

How can random forests be used for survival analysis?

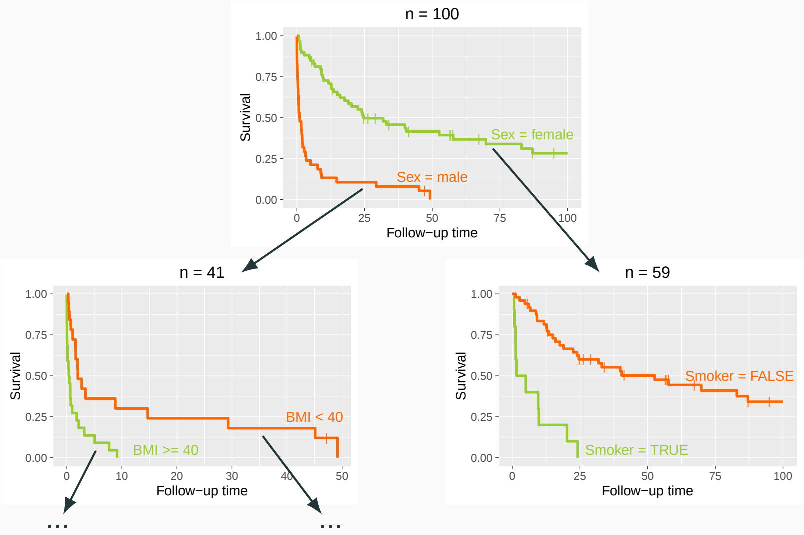


How can random forests be used for survival analysis?



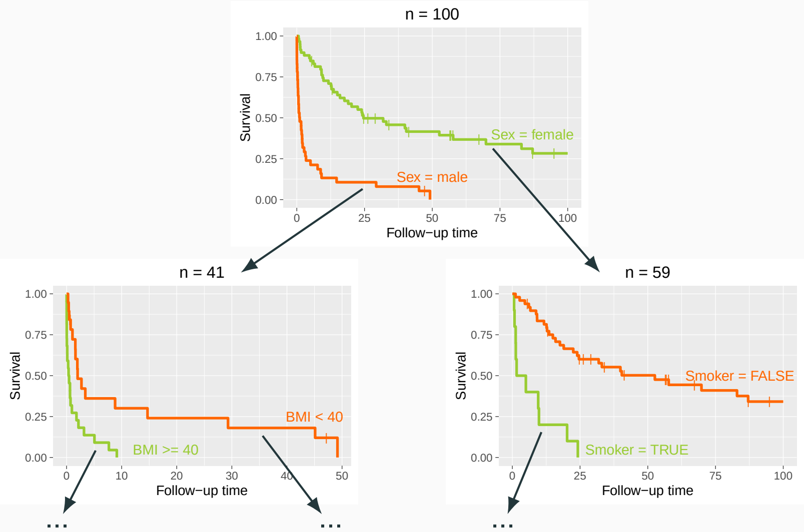
Machine learning for survival endpoints

How can random forests be used for survival analysis?



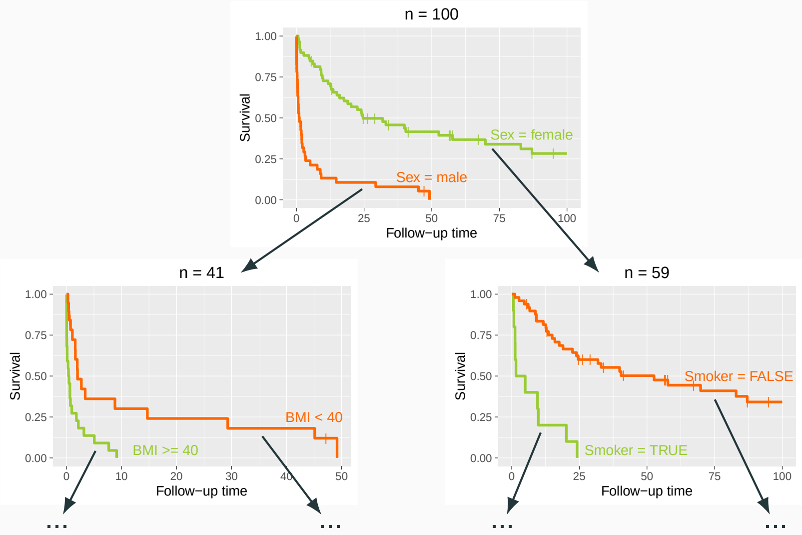
Machine learning for survival endpoints

How can random forests be used for survival analysis?

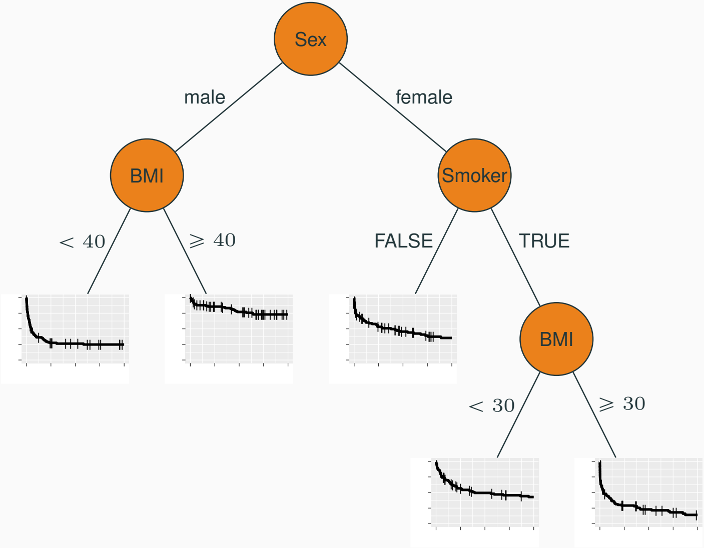


Machine learning for survival endpoints

How can random forests be used for survival analysis?

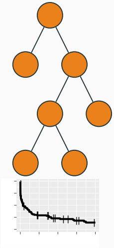


How can random forests be used for survival analysis?



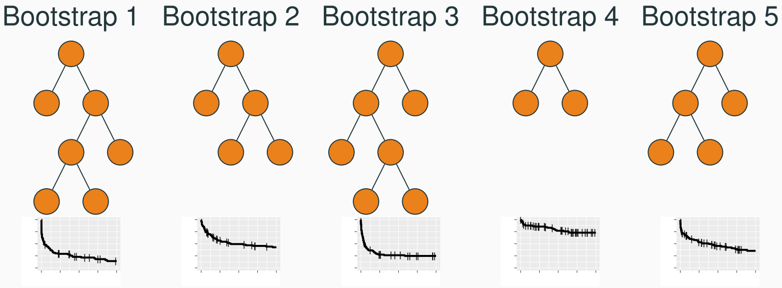
How can random forests be used for survival analysis?

Bootstrap 1



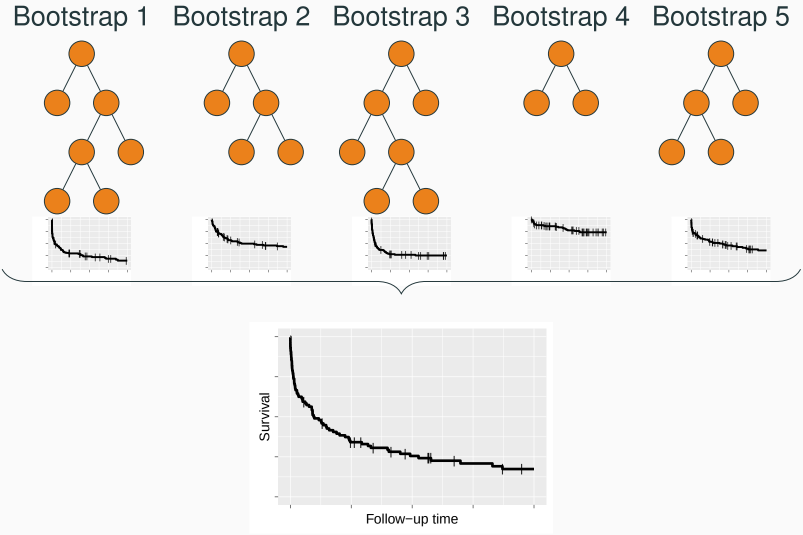
Machine learning for survival endpoints

How can random forests be used for survival analysis?



Machine learning for survival endpoints

How can random forests be used for survival analysis?



Machine learning for survival endpoints

Which random forest approach and which split criterion should be used for random forest with survival outcome?

Biased split point selection

As before: continuous covariates have more possible split points

Examples as before

- Sex: 2 unique values
- Medication type: few unique values
- Age (in years): many unique values
- Biomarker: n unique values

Strobl et al. 2007 BMC Bioinformatics 8:25

Which random forest approach and which split criterion should be used for random forest with survival outcome?

Approaches

- Standard random forests with
  - Log-rank statistic as split criterion
  - Harrell's C index as split criterion
- p-value based 2-step random forests
  - Conditional inference forests with linear rank statistic
  - Maximally selected rank statistic

Wright et al. 2017 Stat Med 36:1272–84

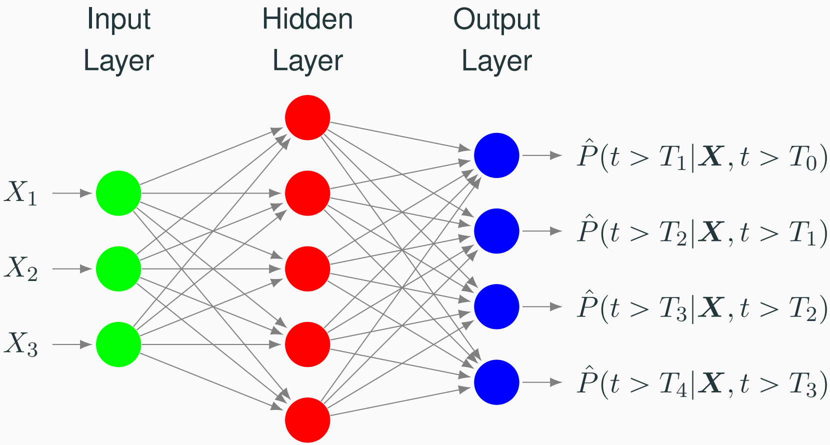
Which random forest approach and which split criterion should be used for random forest with survival outcome?

Summary

1. Log-rank to be used in case of many noise variables
2. Harrell's C index to be used in case of few variables
3. Standard random forest biased
4. Use 2-step subsampling random forest
   - 4.1 CIF slow but excellent
   - 4.2 Maximally selected rank statistics fast but with small bias

Wright et al. 2017 Stat Med 36:1272–84 • Ishwaran 2015 Mach Learn 99:75-118
Schmid et al. 2016 Expert Syst Appl 63:450–9 • Hothorn et al. 2006 J Comput Graph Stat 15:651-74

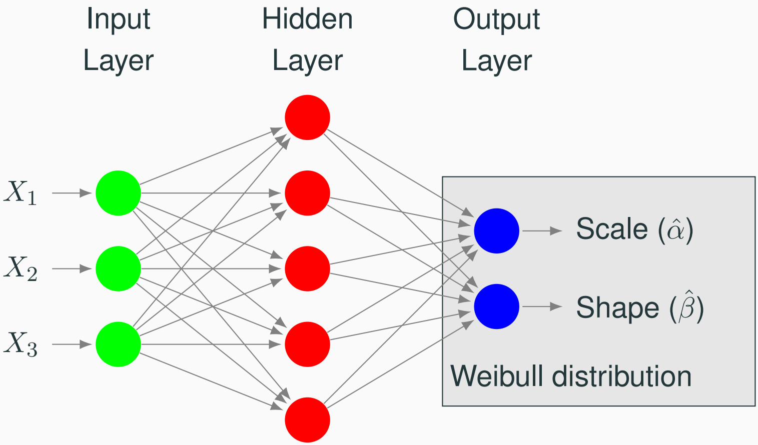
How can neural networks be used for survival analysis?



Loss function: Log-likelihood of discrete hazard model

Gensheimer & Narasimhan 2018 arXiv:1805.00917

How can neural networks be used for survival analysis?



Loss function: Log-likelihood of Weibull distribution

Martinsson 2017 https://github.com/ragulpr/wtte-rnn/

How can SVMs be used for survival analysis?

Approaches

- Ranking: Compare pairs of observations, classify larger survival time
- Regression: Penalization for incorrect predictions of censored survival times if predicted time is lower than observed time
- Hybrid: Minimize weighted sum of ranking and regression functions

Remarks

- Hybrid approach performs best
- Long runtimes for medium and large datasets

Evers & Messow 2008 Bioinformatics 24:1632-8 • Van Belle et al. 2011 Artif Intell Med 53:107-18
Pölsterl et al. 2015 ISBN 9783319234618, pp. 243-59 • Shivaswamy et al. 2007 ISBN 9780769530185:655-60
Fouodo et al. 2018 R J 10:412–423

Part 11

Variable importance
Variable selection

Variable importance

What are general ideas for variable importance?

Permutation importance

Idea: The larger the error after permutation, the more important the variable

Algorithm for variable j:

1. Fit model on original data
2. Determine error e (e.g. test data, cross validation)
3. Permute values x_j of variable j
4. Determine error after permutation e_j
5. Permutation importance: $VIM_j = e_j - e$

Variable importance

What are ideas for model-specific variable importance?

kNN

No specific methods known

Decision trees

- Count number of splits on a variable
- Sum of impurity reduction of all splits on a variable

Boosted trees

Same as single decision trees

Breiman et al. 1984 ISBN 9780412048418

Variable importance

What are ideas for random forest variable importance?

Impurity importance

As in single decision trees → Gini importance

Permutation importance

- Use out-of-bag (OOB) performance estimate for each tree
- Average tree results

Bias-corrected impurity importance

1. Create pseudo-variables by permutation
2. Substract pseudo-variable importance from original importance

Fast but requires re-fitting for prediction

Nembrini et al. 2018 Bioinformatics doi:10.1093/bioinformatics/bty373

What are permutation importance (PI) and scaled PI?

1. For all $b = 1$ to $B =$ `ntree` do
   a) Grow tree
   b) Determine error on OOB data (classification error, Brier score, ...) e_b^{OOB}
   c) Permute values x_j of variable j
   d) Determine error on OOB data using permuted data $e_{b,j}^{OOB}$
   e) Determine difference in errors $e_b^{OOB} - e_{b,j}^{OOB}$
2. **Permutation importance** of variable j
   $PI_j = \frac{1}{B}\sum_{b=1}^{B}(e_b^{OOB} - e_{b,j}^{OOB})$
3. Standard deviation s_j of differences
4. **Scaled importance** (scaled PI) of variable j
   $zPI_j = \frac{PI_j}{s_j/\sqrt{B}}$

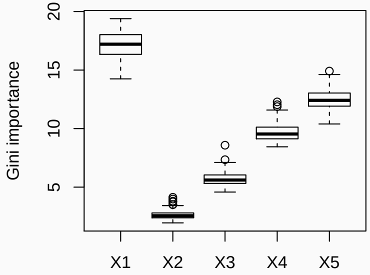
Breiman 2001 Mach LEarn 45:5-32 • Verikas et al. 2011 Pattern Recogn 44:330-49

Variable importance

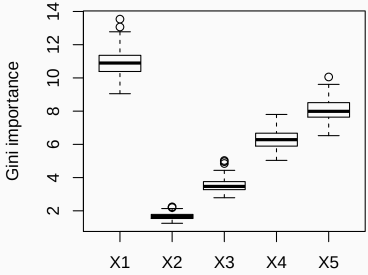
How does the Gini importance perform?

- All independent variables unimportant
- X1 $\sim$ N(0,1), X2 $\sim$ M(2), X3 $\sim$ M(4), X3 $\sim$ M(10), X5 $\sim$ M(20)

Bootstrap



Subsample



Also applies to single and boosted trees

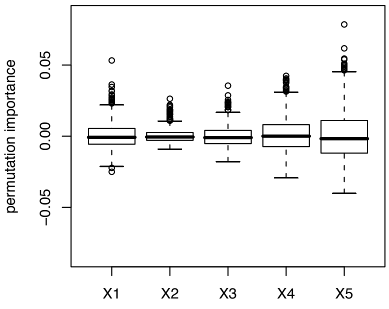
Variable importance

How does the permutation importance perform?

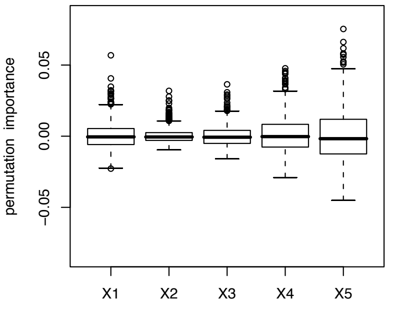
- All independent variables unimportant
- X1 ∼ N(0,1), X2 ∼ M(2), X3 ∼ M(4), X3 ∼ M(10), X5 ∼ M(20)

Standard random forest

Bootstrap



Subsample

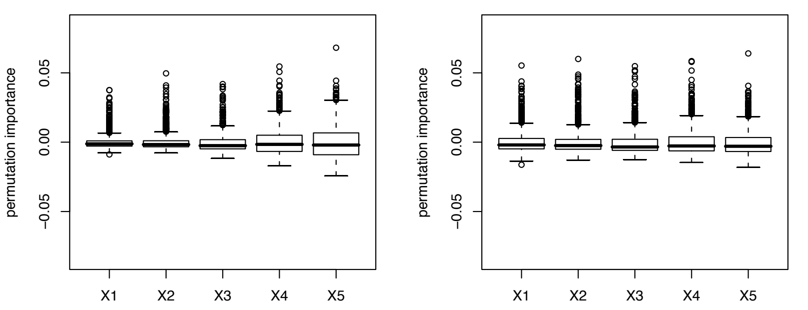


Strobl et al. 2007 BMC Bioinformatics 8:25

How does the permutation importance perform?

Conditional inference forest

Bootstrap Subsample

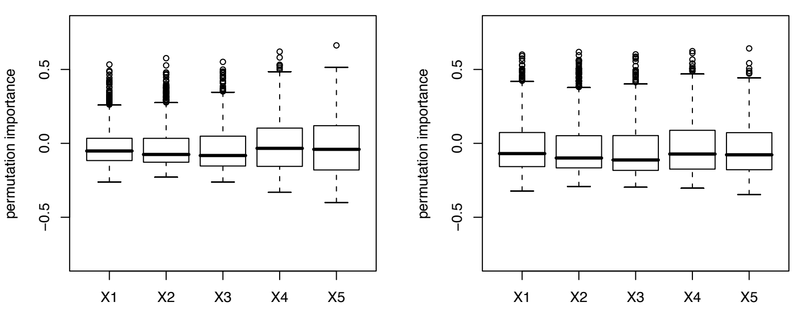


Strobl et al. 2007 BMC Bioinformatics 8:25

How does the scaled importance perform?

- Distribution skewed
- Not asymptotically normal
- Depends on number of trees in random forest
- Magnitude not interpretable

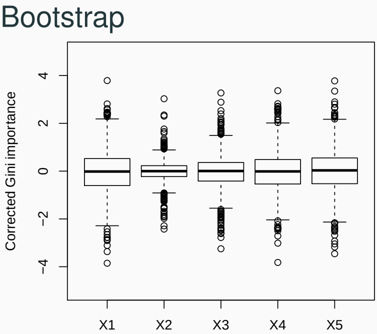
Bootstrap Subsample

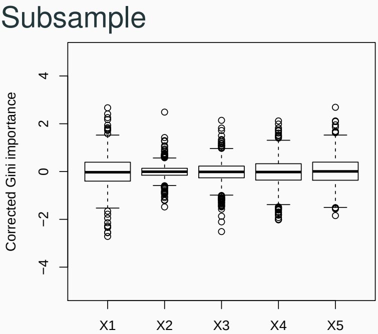


Strobl et al. 2007 BMC Bioinformatics 8:25

Variable importance

How does the corrected Gini importance perform?





Nembrini et al. 2018 Bioinformatics doi:10.1093/bioinformatics/bty373

What are ideas for model-specific variable importance?

SVM

Use parameters β as in regression

$\rightarrow$ Only possible for linear kernel or without kernel trick

Neural networks

Idea: Partition output unit weights to each input unit

Newer methods available

Gaonkar & Davatzikos 2013 NeuroImage 78:270 • Garson 1991 AI Expert 6:46 • Olden et al. 2004 Ecol Model 178:389 • Samek et al. 2021 Proc IEEE 109:247

Which importance measure should be used?

- Random forest measures most elaborated
- Gini importance *not* recommended
- Scaled permutation importance *not* recommended
- Standard permutation importance recommended
- Corrected Gini importance recommended
- Performance depends on number of trees, `mtry` and split criterion

Nicodemus et al. 2010 BMC Bioinformatics 11:110 ● Hapfelmeier & Ulm 2013 Comput Stat Data Anal 60:50-69
Grömping 2007 Am Stat 61:139-147 ● Genuer et al. 2010 R-Journal 7:19-33 ● Probst et al. 2018 arXiv 1804.03515

Variable importance

How can importance be statistically tested?

Permutation approach

1. Fit model on original data set
2. Estimate variable importance
3. For $perm = 1$ to $Perm$
   a) Permute y
   b) Fit model on permuted data set
   c) Estimate variable importances
4. p-value obtained as proportion of permuted importance scores at least as large as importance from original data

Only shown for random forests

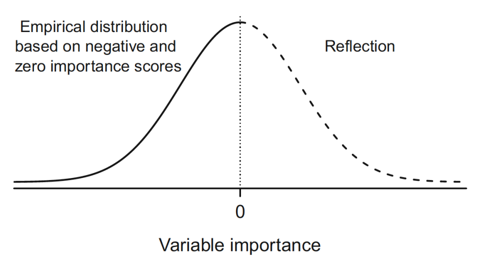
Altmann et al. 2010 Bioinformatics 26:1340

Variable importance

How can importance be statistically tested?

Null importance test

- Under H_0 distribution of importance expected to be symmetric around 0
- Idea: Reconstruct distribution of null importance with importance of variables likely to be irrelevant:
  - Zero importance variables
  - Negative importance variables



Janitza et al. 2016 Adv Data Anal Classif doi:10.1007/s11634-016-0270-x • Nembrini et al. 2018 Bioinformatics

How can importance be statistically tested?

Null importance test

- Only possible for high dimensional data
- Shown to work with random forests
  - Holdout importance (modification of permutation importance)
  - Corrected Gini importance (faster)
- No results for other learners

Janitza et al. 2016 Adv Data Anal Classif doi:10.1007/s11634-016-0270-x
Nembrini et al. 2018 Bioinformatics doi:10.1093/bioinformatics/bty373

Variable selection

What are ideas for variable selection?

- Backward elimination
- Forward selection
- Use variable importance
    - Rank based
    - Test based
- *Intrinsic* variable selection in some learners
- Possible to use different method for variable selection
- Multivariate approaches recommended

Díaz-Uriarte & Alvarez de Andrés 2006 BMC Bioinformatics 7:3 ● Genuer et al. 2010 R-Journal 7:19-33
Hapfelmeier & Ulm 2013 Comput Stat Data Anal 60:50-69 ● Degenhardt et al. 2017 Brief Bioinform doi:10.1093/bib/bbx124

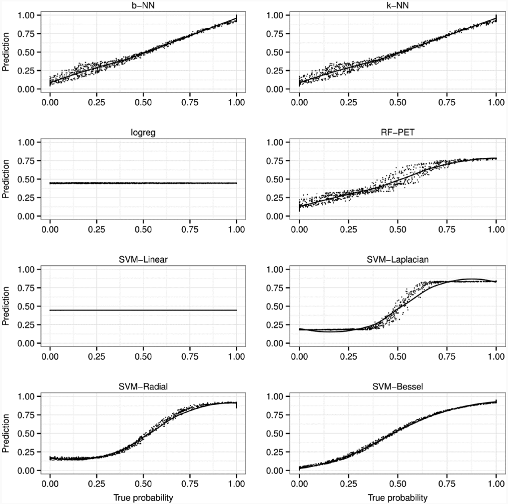
Part 12

Discussion

Discussion

Is there a single best learner?

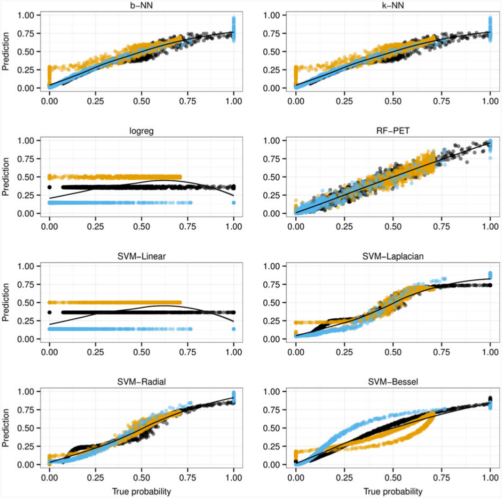
Mease model I



Kruppa et al. 2014 Biom J 56:534-63

Is there a single best learner?

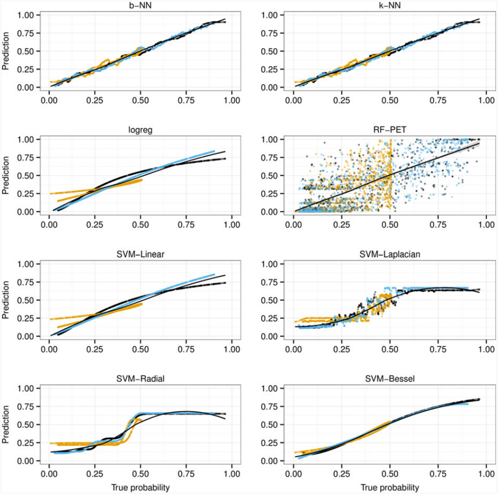
Mease model II



Kruppa et al. 2014 Biom J 56:534-63

Is there a single best learner?

Lee model

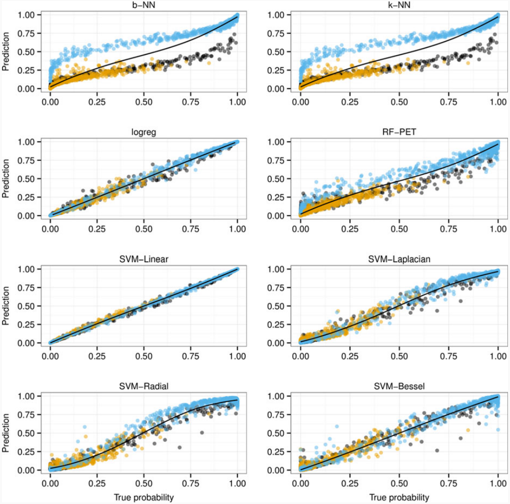


Kruppa et al. 2014 Biom J 56:534-63

Discussion

Is there a single best learner?

Li model I

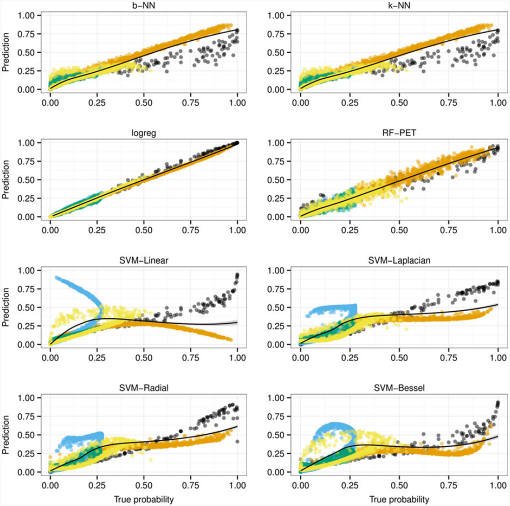


Kruppa et al. 2014 Biom J 56:534-63

Discussion

Is there a single best learner?

Li model II



Kruppa et al. 2014 Biom J 56:534-63

Discussion

Is there a single best learner?

General recommendations

- Typically RF $>$ Tree $>$ kNN
- RF robust, easy to tune and fast
- Boosting often slightly better than RF on tabular data (when properly tuned)
- SVM good alternative for binary classification with numerical features (when properly tuned)
- Image, text and speech data $\rightarrow$ Deep Learning

Discussion

What are important aspects when applying machine learning?

- Subsampling statistically superior over bootstrapping
- Subsampling and bootstrapping lead to same prediction performance

- Interpretability versus performance
- Transferability

Discussion

What are important aspects when applying machine learning?

- Never use default parameter settings
- Tune parameters!
- Tune parameters jointly!
- Parameter tuning simple and straightforward for
  - kNN
  - bNN
  - random forests
- Parameter tuning complex and not straightforward for
  - SVM: parameters depend on kernel
  - ANN: tuning of architecture
- Use adequate resampling strategy
- Gold standard: nested cross validation

Discussion

What are important aspects when applying machine learning?

- Extensions required for
  - Matched samples
  - Longitudinal endpoints
  - Family data
- Development is triggered by other research areas for ANN and deep learning (image, signal)

Three parts of data science

	Statistics	Machine learning
Exploration	+/−	+/−
Effect estimation	+	−
Prediction	−	+

Acknowledgements

Some figures from course Introduction to Machine Learning (I2ML)

Bernd Bischl, Fabian Scheipl, Daniel Schalk, Heidi Seibold et al.

https://github.com/compstat-lmu/lecture_i2ml

Develop lectures similar to open source software