Bern University
of Applied Sciences

Mastering Machine Learning for Spatial Prediction I

Introduction and Overview of Methods

OpenGeoHub
4/5 September 2019

Madlene Nussbaum

Objectives ...

- Make sense of **terms** and **concepts** often heard

- Get an **overview** of machine learning (ML) methods and their strategies

- Learn to **apply** 2-4 ML techniques

- Machine learning will not solve all your problems by one click, so **be critical!**

Be able to judge if computing model averaging on 78 methods found in Package caret is a sensible thing to do …

Content of lecture

Introduction: some terms

Computer
Science

**Machine
Learning**
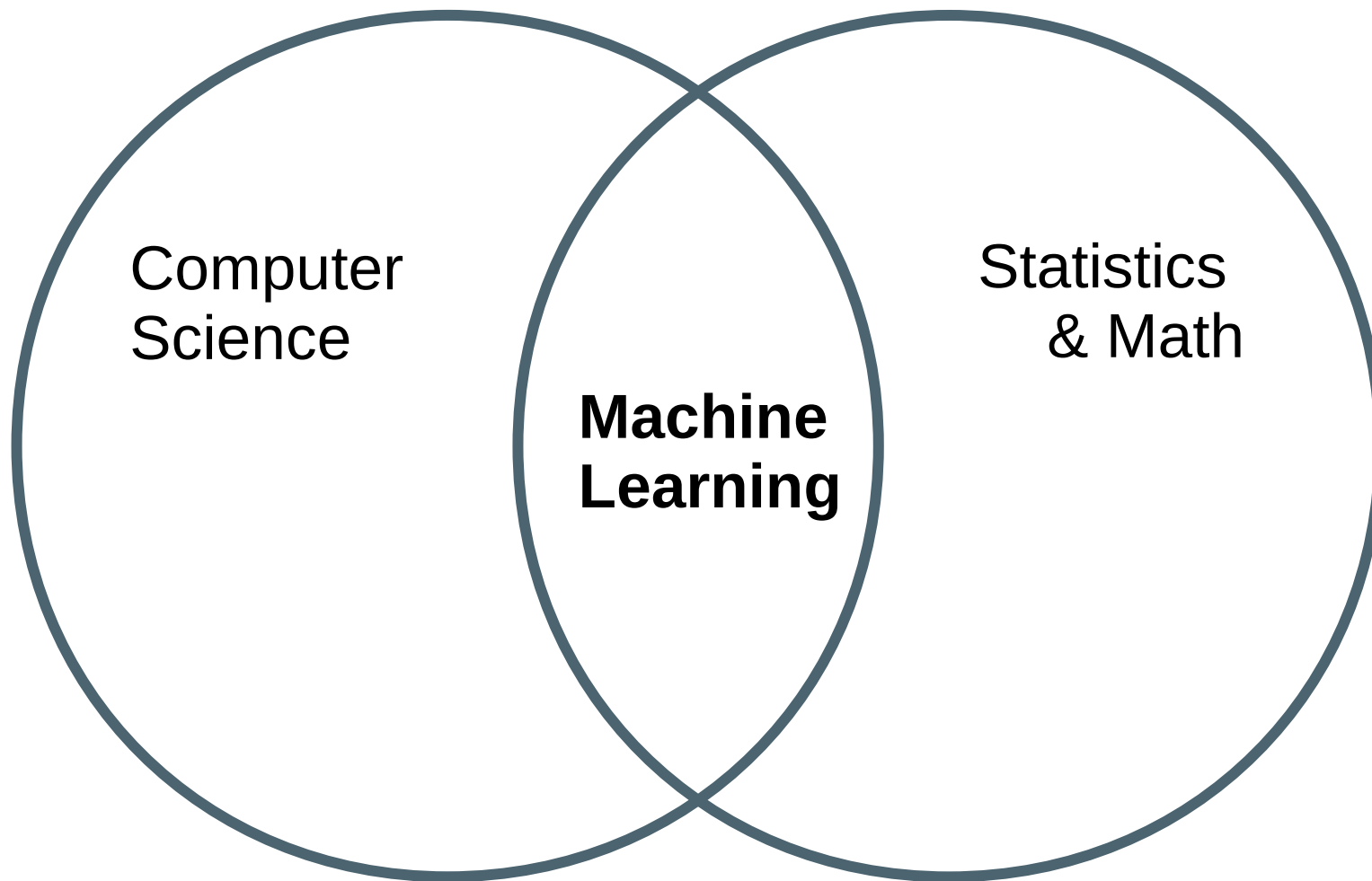
Statistics
& Math

→ For environmental mapping maybe better:
  statistical learning or **computational statistics**

Introduction: some terms

- **Unsupervised learning**
  No response, just "covariates"
    → e.g. clustering of satellite image by similar values
    → operates "blind"

- **Supervised learning**
  For each covariate value x_i there is also a response value y_i
    → e.g. random forest
    → what we usually do for environmental mapping

    – Regression: continuous responses, e.g. soil clay content, rainfall

    – Classification: categorical responses (binary or multinomial),
       e.g. soil type

Huge topic, hence further reading advised:

Gareth et al. 2017, very nice and solid introduction, easy accessible.

Gareth, James, Witten, Daniela, Hastie, Trevor and Tibshirani, Robert. An introduction to statistical learning : With applications in R. 8 edn. New York: Springer, 2017.

Hastie et al. 2009, very good and detailed book, but rather advanced.

Hastie, T., Tibshirani, R., and Friedman, J.: The Elements of Statistical Learning; Data Mining, Inference and Prediction, Springer, New York, 2 edn., 2009. with examples and data in R package ElemStatLearn, https://cran.r-project.org/web/packages/ElemStatLearn/index.html

Kuhn et al. 2013, form the author of the R package caret, focuses a bit more on classification

Kuhn, M., Johnson, K.: Applied predictive modeling, Springer, New York, 2013.

See also caret package website for overviews ans basic explanations: https://topepo.github.io/caret/

Hothorn, 2018, overview of R packages for ML:

Hothorn, Torsten. CRAN Task View: Machine Learning & Statistical Learning https://CRAN.R-project.org/view=MachineLearning, 2018.

Content of lecture

Terms and concepts

Spatial modelling: requirements?

Overview of ML and their strategies

Side note: overfitting

Methods

– Lasso

– Support Vector Machines

– Ensemble Learners

- Random Forest
- Boosting
- Model averaging

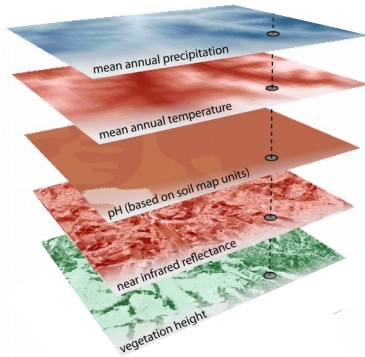# Step back: What do we need for spatial predictions?

My situation …

texture
density
gravel
soil depth
drainage
pH, ECEC
SOC

300-1400
locations with
soil properties in

2–4 **soil depth**
3 **study areas**

mean annual precipitation

mean annual temperature

pH (based on soil map units)

near infrared reflectance

vegetation height

300-500
environmental
covariates

48
statistical models

Requirements
A spatial prediction method should …

- model **nonlinear** relations

- consider **spatial** autocorrelation

- model continuous and categorical responses

- handle **numerous** correlated **covariates** without overfitting calibration data

- **automatically** build models with **good predictive power**

- preferably result in **sparse model**

- accurately quantify **accuracy** of **predictions**

- give prediction **uncertainty**

Content of lecture

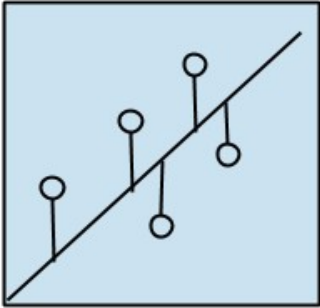Terms and concepts

Spatial modelling: requirements?

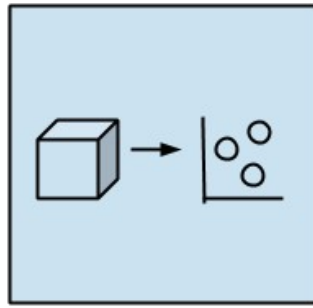Overview of ML and their strategies

Side note: overfitting

Methods

- Lasso

- Support Vector Machines

- Ensemble Learners
  - Random Forest
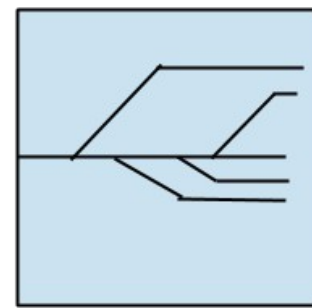 - Boosting
  - Model averaging

I tried to tidy up ...

Regression
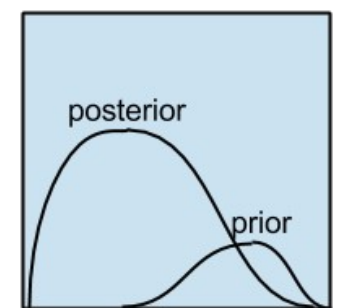linear and non-linear models, geostatistics
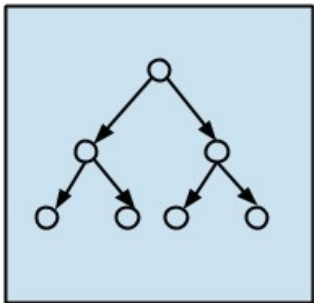
Dimension reduction
PCA, PLS

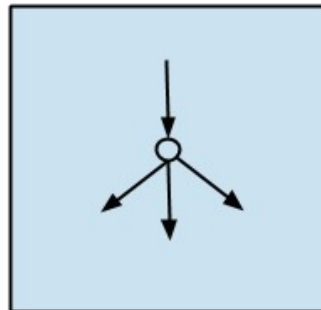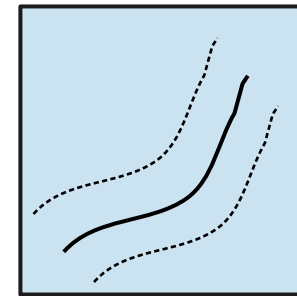**Regularisation Shrinkage**
Lasso

Bayes methods
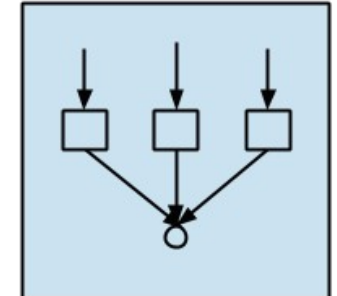
Decision trees
CART

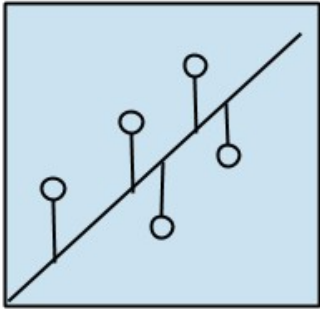**Neuronal networks**

Support vector machines
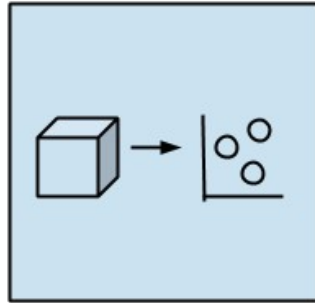kernel methods

Ensembles
bootstrap, boosting, model averaging
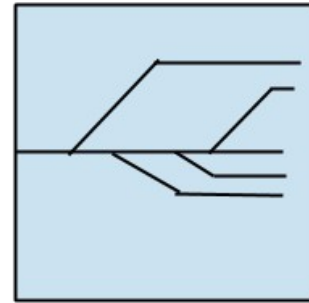
I tried to tidy up ...

Regression
linear and non-linear
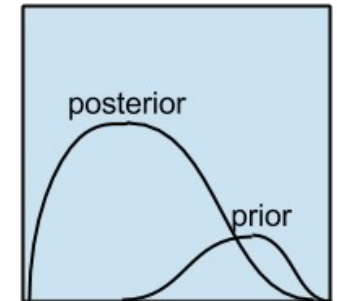models, geostatistics

**Dimension
reduction**
PCA, PLS
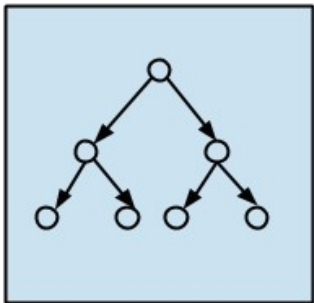
**Regularisation
Shrinkage**
Lasso

**Bayes
methods**

Linear (more or less)

Decision trees
CART
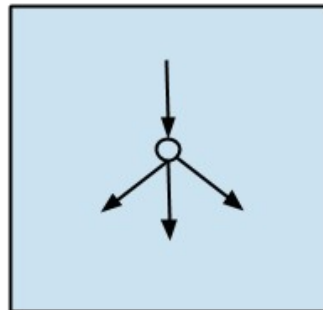
**Neuronal
networks**
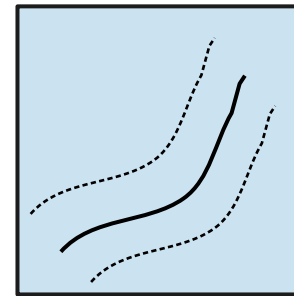
**Support vector
machines**
kernel methods
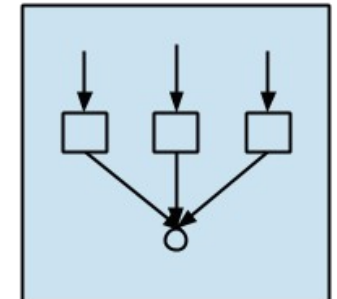
Ensembles
bootstrap, boosting,
model averaging

High complexity

Content of lecture

Terms and concepts

Spatial modelling: requirements?

Overview of ML and their strategies

Side note: overfitting

Methods

– Lasso

– Support Vector Machines

– Ensemble Learners

- Random Forest
- Boosting
- Model averaging

Side note:
Overfitting? Bias-Variance trade-off

High Bias
Low Variance

Low Bias
High Variance

Prediction Error

Test Sample

Training Sample

Low

High

Model Complexity

FIGURE 2.11. *Test and training error as a function of model complexity.*

Hastie et al. 2009, p. 38.

Bias-Variance tradeoff

Linear Regression of 0/1 Response
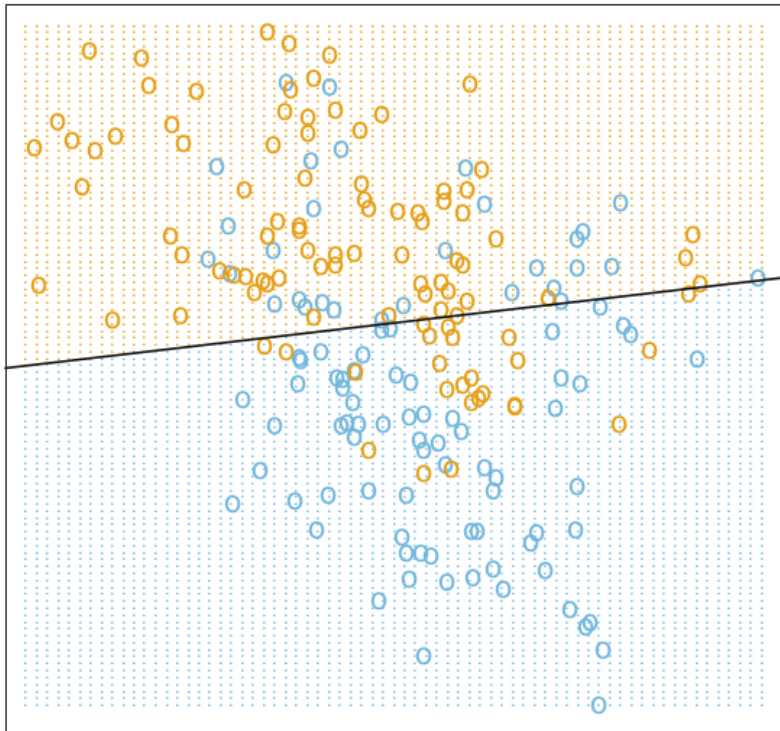
1–Nearest Neighbor Classifier
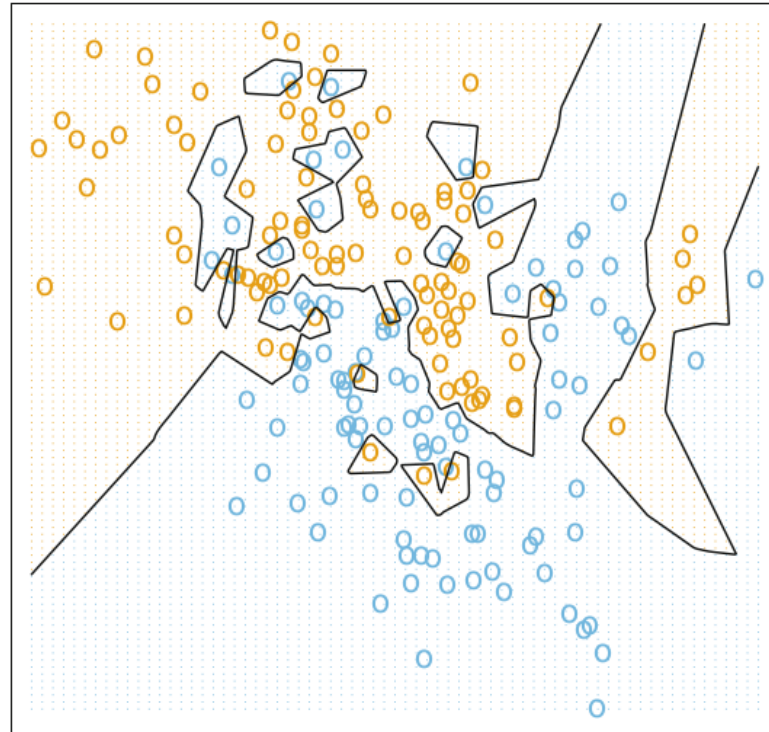
Linear model
high bias, but stable

1-nearest neighbours
low bias, high variance

$$\mathrm{E}\left[\left(y - \hat{f}\left(x\right)\right)^2\right] = \mathrm{Bias}\left[\hat{f}\left(x\right)\right]^2 + \mathrm{Var}\left[\hat{f}\left(x\right)\right] + \sigma^2$$

Bias: erroneous assumptions in the model, miss relevant relationship (underfitting).

Variance: sensitivity to small fluctuations in the calibration data, algorithm models random noise in calibration data, instead of just relevant relationship (overfitting).

Content of lecture

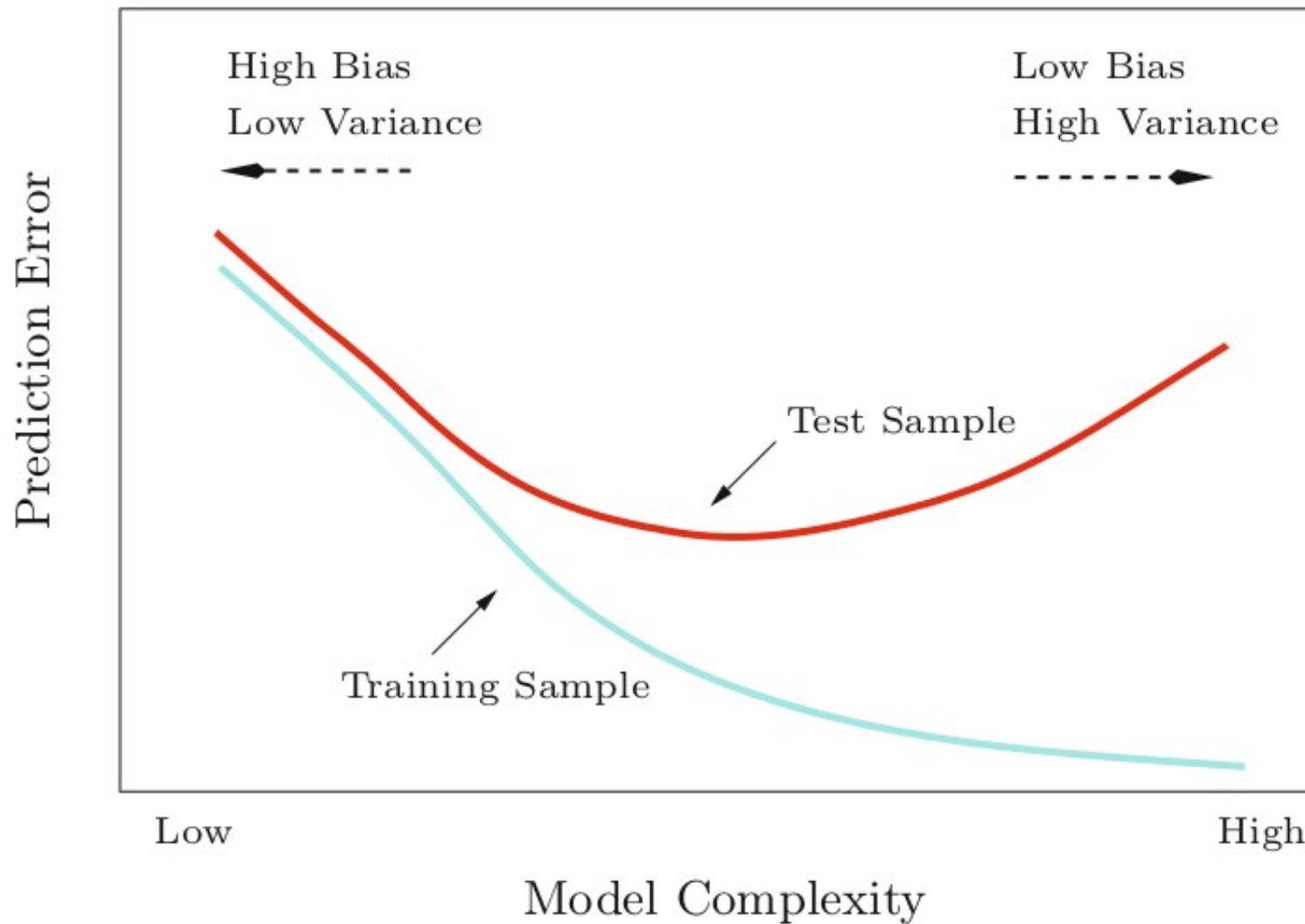Terms and concepts

Spatial modelling: requirements?

Overview of ML and their strategies

Side note: overfitting

Methods
- Lasso
- Support Vector Machines
- Ensemble Learners
    - Random Forest
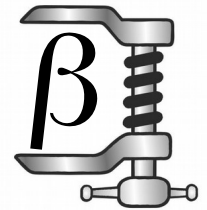   - Boosting
    - Model averaging

Lasso: ML for linear models

- Select linear regression with stepwise forward/backward, best subset: Most often does not find true model, does overfit, selection is binary – either in or out

- **Shrinkage:** include a covariate, but with smaller / downweighted coefficients

- Different approaches (ridge regression etc.), most promising: Lasso: least absolute shrinkage and selection operator

$$\hat{\beta}^{\text{lasso}} = \underset{\beta}{\text{argmin}} \left\{ \frac{1}{2} \boxed{\sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij} \beta_j \right)^2} + \boxed{\lambda \sum_{j=1}^{p} |\beta_j|} \right\}.$$

OLS **Lasso penalty**

- Thus the lasso does a kind of continuous subset selection.

- Tuning Parameter λ, find by cross validation

Lasso: ML for linear models

Path of coefficents for increasing tuning parameter

FIGURE 3.10. *Profiles of lasso coefficients, as the tuning parameter t is varied. Coefficients are plotted versus $s = t / \sum_1^p |\hat{\beta}_j|$. A vertical line is drawn at $s = 0.36$, the value chosen by cross-validation. Compare Figure 3.8 on page 65; the lasso profiles hit zero, while those for ridge do not. The profiles are piece-wise linear, and so are computed only at the points displayed; see Section 3.4.4 for details.*

Hastie et al. 2009, p. 70

Model selection with lasso

FIGURE 3.10. *Profiles of lasso coefficients, as the tuning parameter t is varied. Coefficients are plotted versus $s = t/\sum_{1}^{p} |\hat{\beta}_j|$. A vertical line is drawn at $s = 0.36$, the value chosen by cross-validation. Compare Figure 3.8 on page 65; the lasso profiles hit zero, while those for ridge do not. The profiles are piece-wise linear, and so are computed only at the points displayed; see Section 3.4.4 for details.*

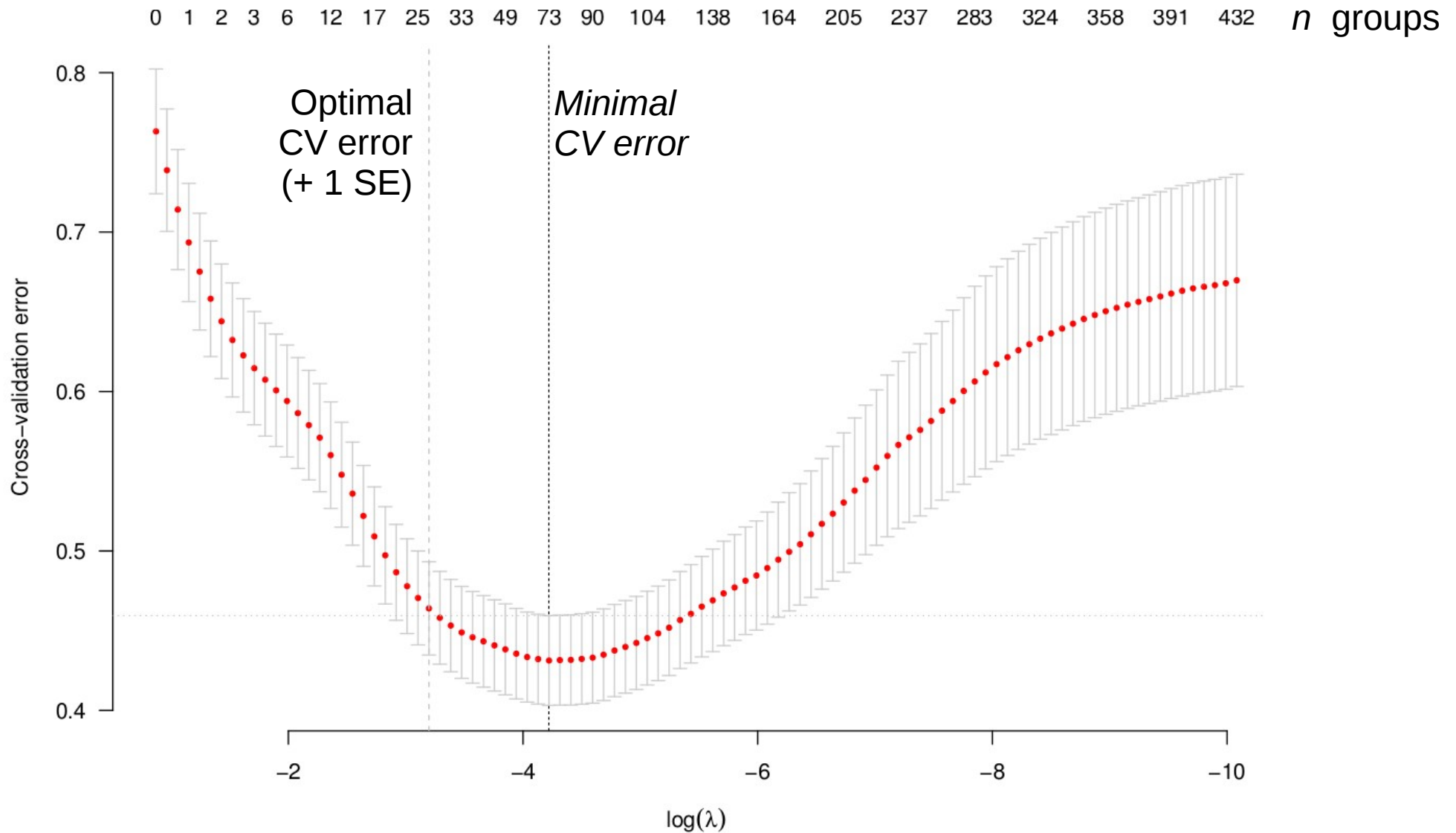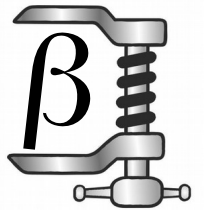Path of coefficients for increasing tuning parameter

Coefficient becomes 0, meaning the covariate is removed from the model

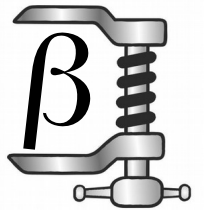With lambda = 1, there is no shrinkage, and we have the normal ordinary least squares linear model fit

Hastie et al. 2009, p. 70

Lasso: ML for linear models

Berne data set, subspoil pH, >400 partly higly correlated and noisy covariates

Lasso: Pros and Cons

✔ Very fast

✔ Selects covariates

✔ No problems with colinearity

✔ Easy interpretation (linear relationships)

✔ Linear regression with a lot of covariates, even $n>>p$

✖ Response transformation needed (assumption of Gaussian errors)

✖ Linear only, no interactions if not added explicitly
(if n>>>p becomes nonlinear again)

✖ Take care, not always stable

✖ Rather underfitting
(possible solution: relaxed Lasso with a second fit on non-zero covariates only)

✖ Standard errors not defined, prediction uncertainty only with bootstrap

✖ No direct spatial modelling, only via workaround

Content of lecture
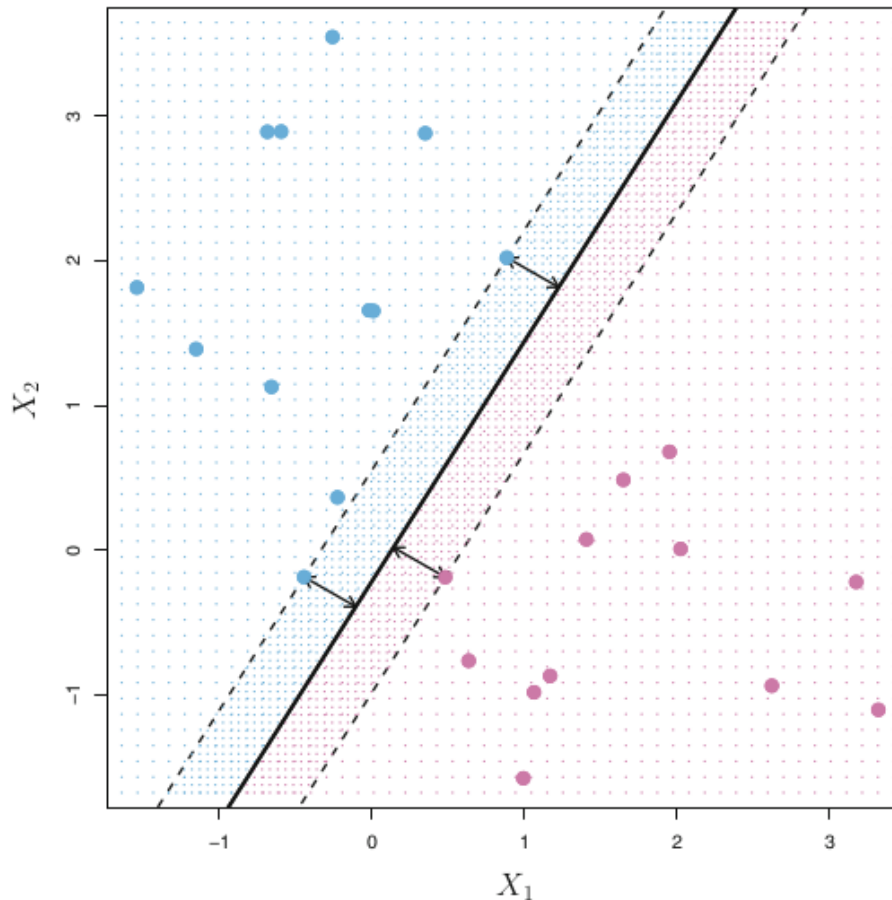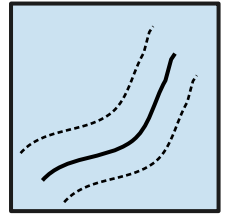
Terms and concepts

Spatial modelling: requirements?

Overview of ML and their strategies

Side note: overfitting

Methods
- Lasso
- Support Vector Machines
- Ensemble Learners
  - Random Forest
  - Boosting
  - Model averaging

Support vector classifier – a linear classifier

Split data by a separating line (2 covariates) or **separating hyperplane** (>2 covariates) that is the **farthest away** from the observations (maximal margin hyperplan).

FIGURE 9.3. *There are two classes of observations, shown in blue and in purple. The maximal margin hyperplane is shown as a solid line. The margin is the distance from the solid line to either of the dashed lines. The two blue points and the purple point that lie on the dashed lines are the support vectors, and the distance from those points to the hyperplane is indicated by arrows. The purple and blue grid indicates the decision rule made by a classifier based on this separating hyperplane.*

Gareth et al. 2013

SV classifier – Tuning parameter C
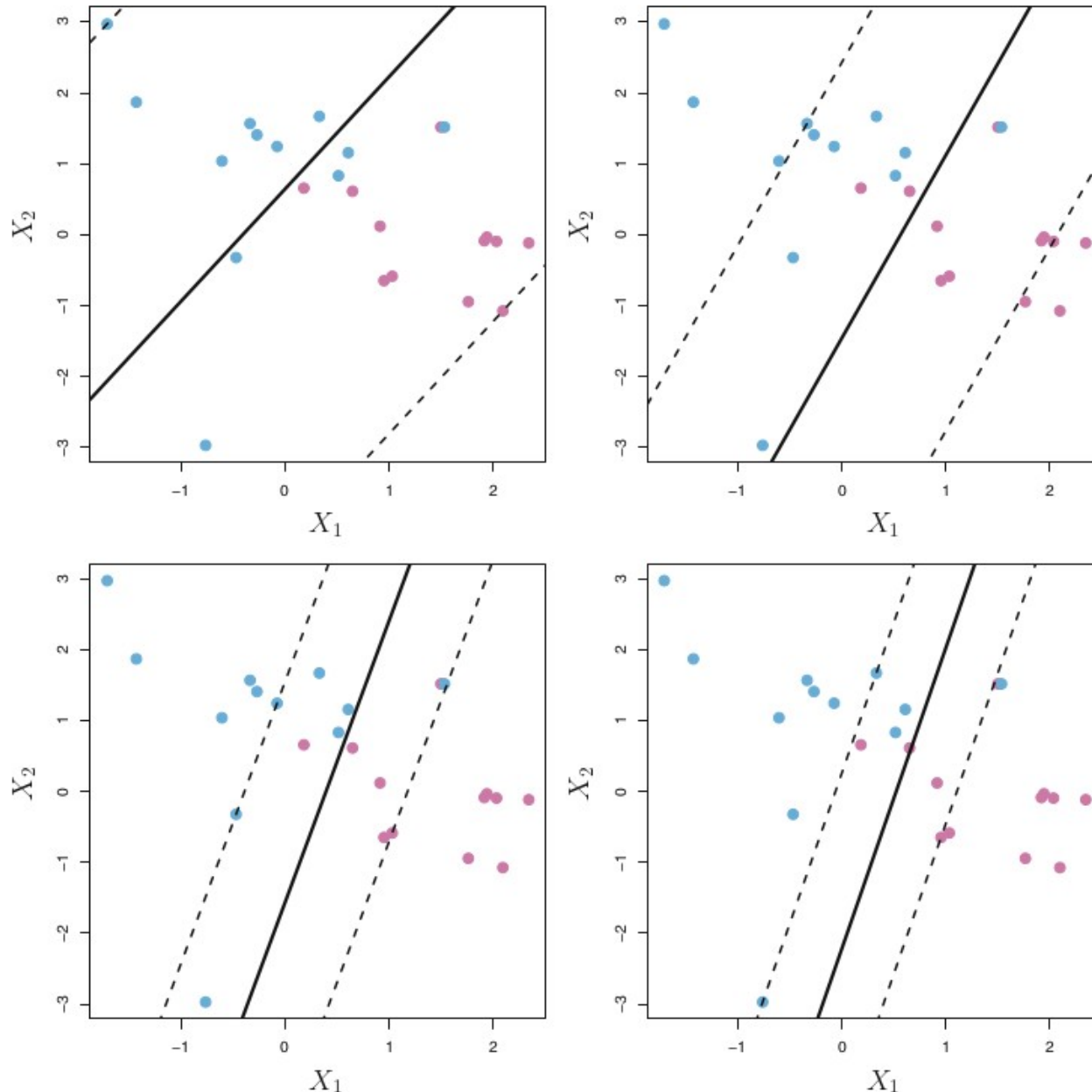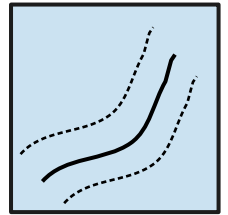
Soft margin classifier

A support vector classifier was fit using four different values of the tuning parameter C. The largest value of C was used in the top left panel. When C is large, then there is a **high tolerance for observations being on the wrong side of the margin and the hyperplane**, and so the margin will be large.

Gareth et al. 2013

SV classifier – linearity challenged
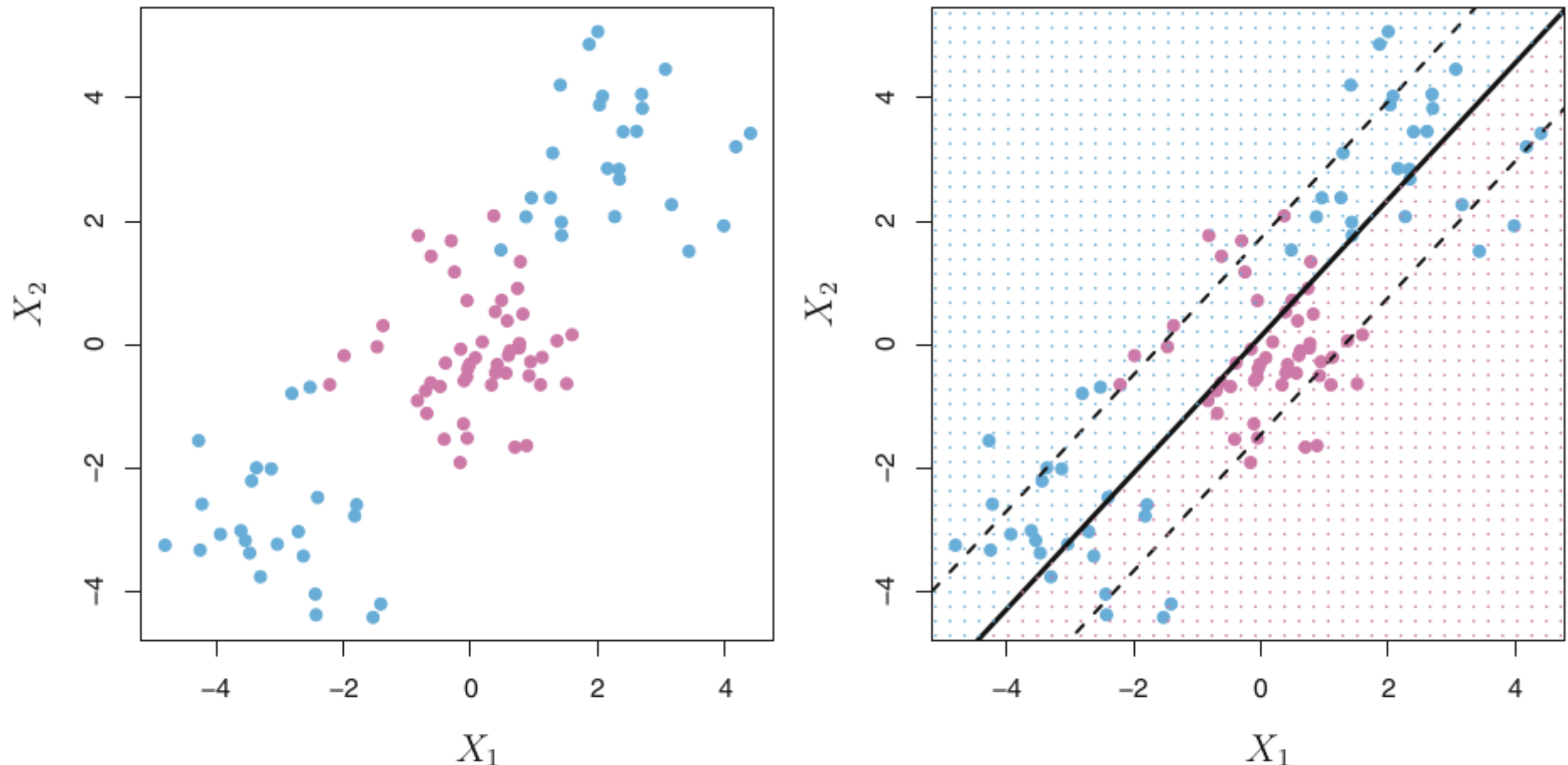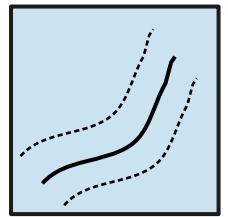
FIGURE 9.8. Left: *The observations fall into two classes, with a non-linear boundary between them.* Right: *The support vector classifier seeks a linear boundary, and consequently performs very poorly.*

Gareth et al. 2013

Support vector machine with kernels
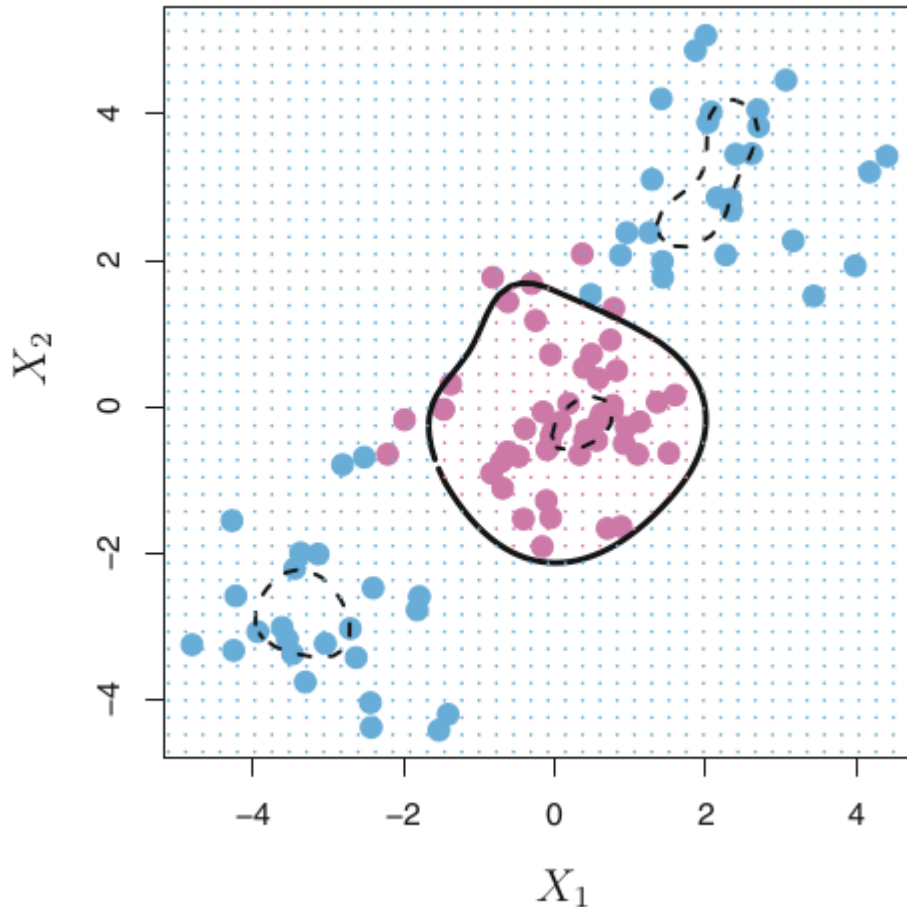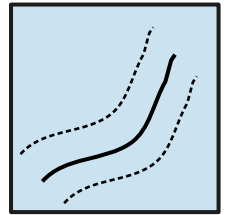
SVM with a radial kernel

High-dimensional classifier

Mainly depends on data points in the center of the distribution, hence is robust to outliers

Tuning parameters
C: how many observations are allowed to be on the wrong side or within the margins
Sigma: degree of smooting of kernels / how much non-linearity is allowed

Gareth et al. 2013

SVM: Pros and Cons
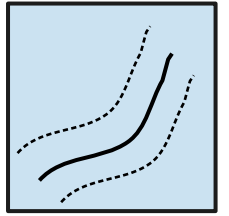
✔ Quite fast

✔ Robust to outliers

✔ Handles non linear relationships

✔ Good predictive power expected

✔ Easy to apply, only 1-2 tuning parameters

✖ Does not select covariates

✖ Difficult to interpret

✖ Standard errors not defined, prediction uncertainty only with bootstrap

✖ No direct spatial modelling, only via workaround

Content of lecture

Terms and concepts

Spatial modelling: requirements?

Overview of ML and their strategies

Side note: overfitting

Methods
- Lasso
- Support Vector Machines
- Ensemble Learners
  - Random Forest
  - Boosting
  - Model averaging

Ensemble Machine Learners

- Combine predictions of several learners
  (any method)

- Meaningful for low-bias, high-variance procedures
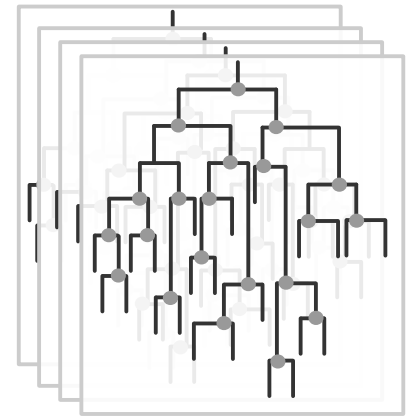
Strategies:

- *Bagging* = *bootstrap aggregation.*
  Uniform *r*esampling the data with replacement (no change of response distribution), fit the data to each resampled set, prediction = average of all single predictions

- Gradient boosting
  Adaptive updating strategy, shrunken stepwise forward selection, fits on residuals → change of distribution

- Model averaging
  Fits on the same response by different methods

random forest

- Ensemble method with CART as base element
  CART: classification and regression tree
  = recursive binary splitting of the dataset

- A large number of trees (ntree) are grown, e.g. 500

- Algorithm ensures trees are decorrelated with

  – Resampling of original dataset with replacement

  – Not all covariates are used at splits of the tree, only a subset (mtry)

- Prediction is the average of all trees (continouous response) or the majority vote (binary or multinomial response)

random forest: algorithm

1) Resample dataset (with replacement)

2) Take a random sample of covariates

3) Test all selected covariates:
   find optimal covariate value to split
   data into 2 portions (minimize
   squared error or wrongly classified)

4) Chose covariate that has the lowest
   error and split data

5) Continue to split the data with (3)-(4)
   until you are left with a small number
   of data points (*nodesize*) in each leaf
   of the tree

6) Repeat (1)-(5) *ntree* times

Main tuning parameters:

mtry: number of randomly selected covariates to
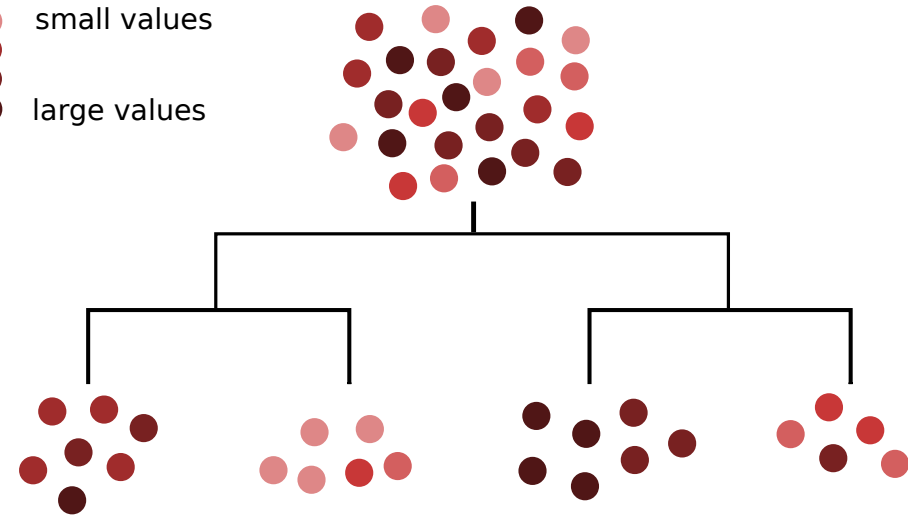try at each split

ntree: number of trees (mostly not sensitive)

nodesize: size of remaining dataset in tree leaf,
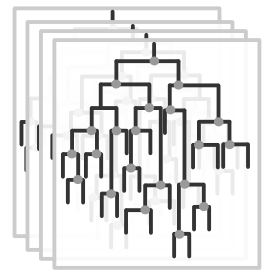when it stops to split (mostly not sensitive)

First resampling:

continouous response
- small values
- large values

Second resampling:
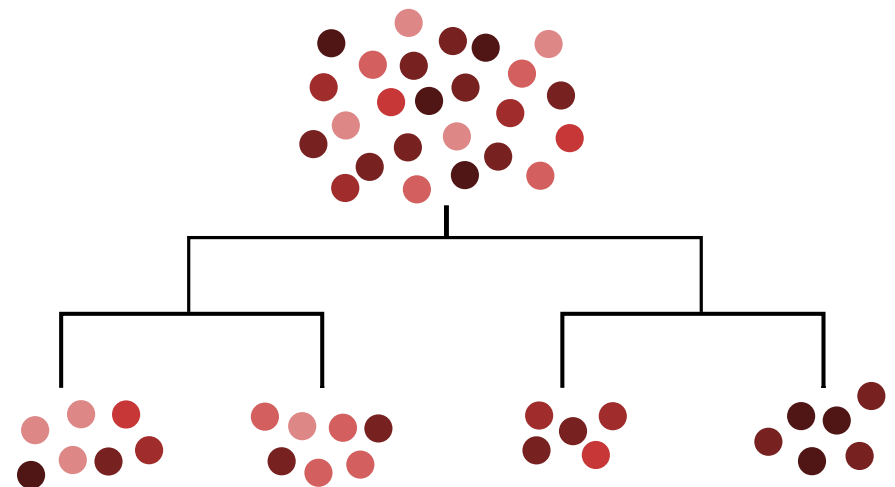
random forest: out-of-bag
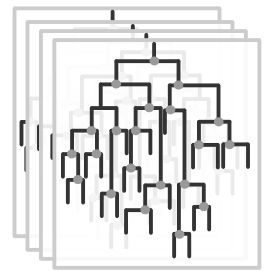
For each observation $z_i = (x_i, y_i)$, construct its random forest predictor by averaging only those trees corresponding to boot-strap samples in which z_i did not appear.   Hastie et al. 2009, p. 593

1. take a random sample of N
(with replacement)
some data points will be duplicated/
triplicated, some will not be chosen
(about 30 % will be left out)

2. Fit tree to resampled dataset of N
Hence, some data points are not used for
model fitting, they are out-of-bag.

dataset of N
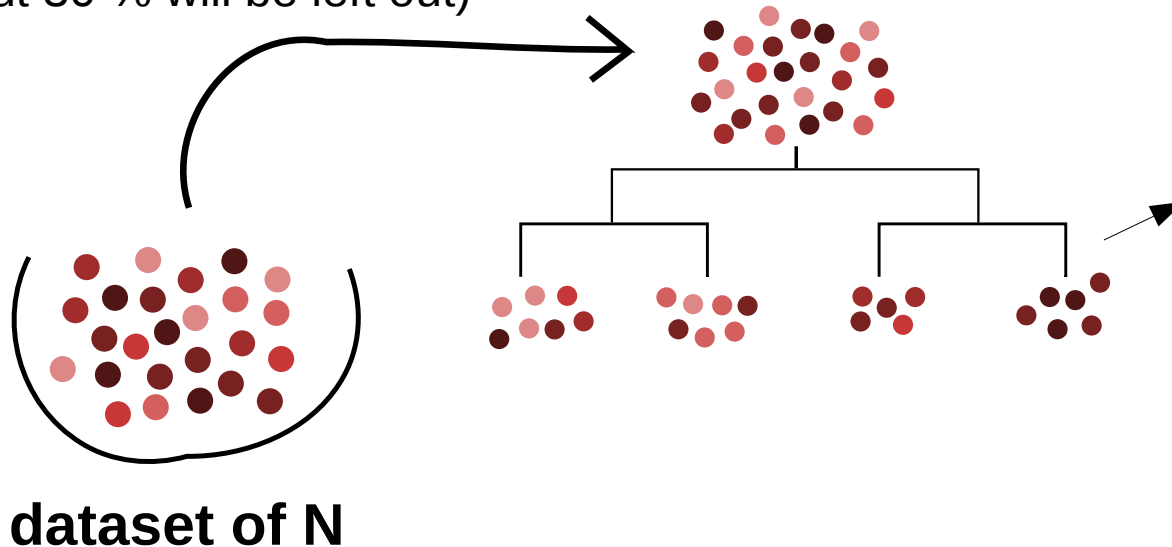
3. Compute predictions for the out-of-bag data points
These predictions can now be compared to the observed values and error statistics can be calculated.

Random forest: Pros and Cons
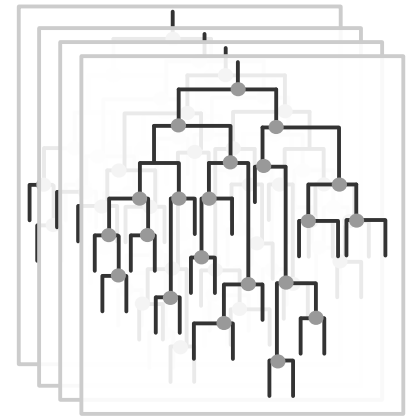
- ✔ Quite fast

- ✔ Models interactions in the data / non linear relationships

- ✔ Very good predictive power expected

- ✔ Yields covariate importance, makes covariate selection possible

- ✔ Prediction uncertainty for continouous responses

- ✖ If no additional covariate selection implemented: difficult to interpret

- ✖ Don't trust every random forest, also this model can overfit

- ✖ No direct spatial modelling, only via workaround

Content of lecture

Terms and concepts

Spatial modelling: requirements?

Overview of ML and their strategies

Side note: overfitting

Methods
- Lasso
- Support Vector Machines
- Ensemble Learners
  - Random Forest
  - Boosting
  - Model averaging

Gradient boosting: Algorithm

model selection for high-dimensional regression

1. compute residuals (gradient)

2. fit base-learners and select best fitting base-learner
$$u^m$$

3. update parameters stepwise as sum of previously fitted estimates of the base-learners

$$f^m = f^{m-1} + v u^m$$
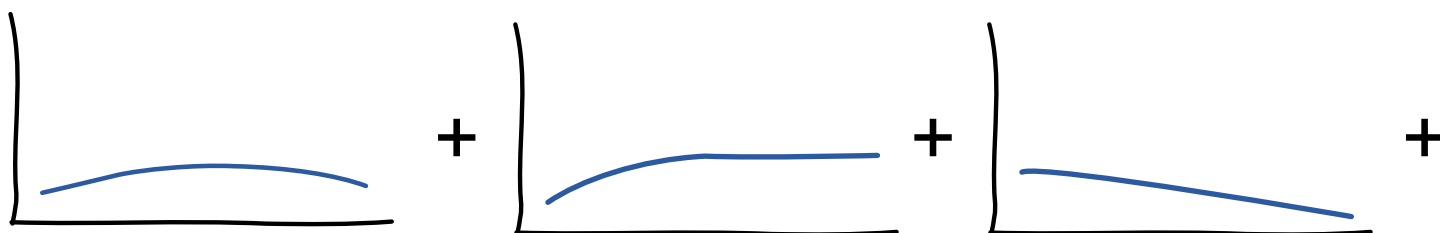
find optimal number of iterations

small step size v = "weak" learner (again shrinkage!)

Gradient boosting: mini example
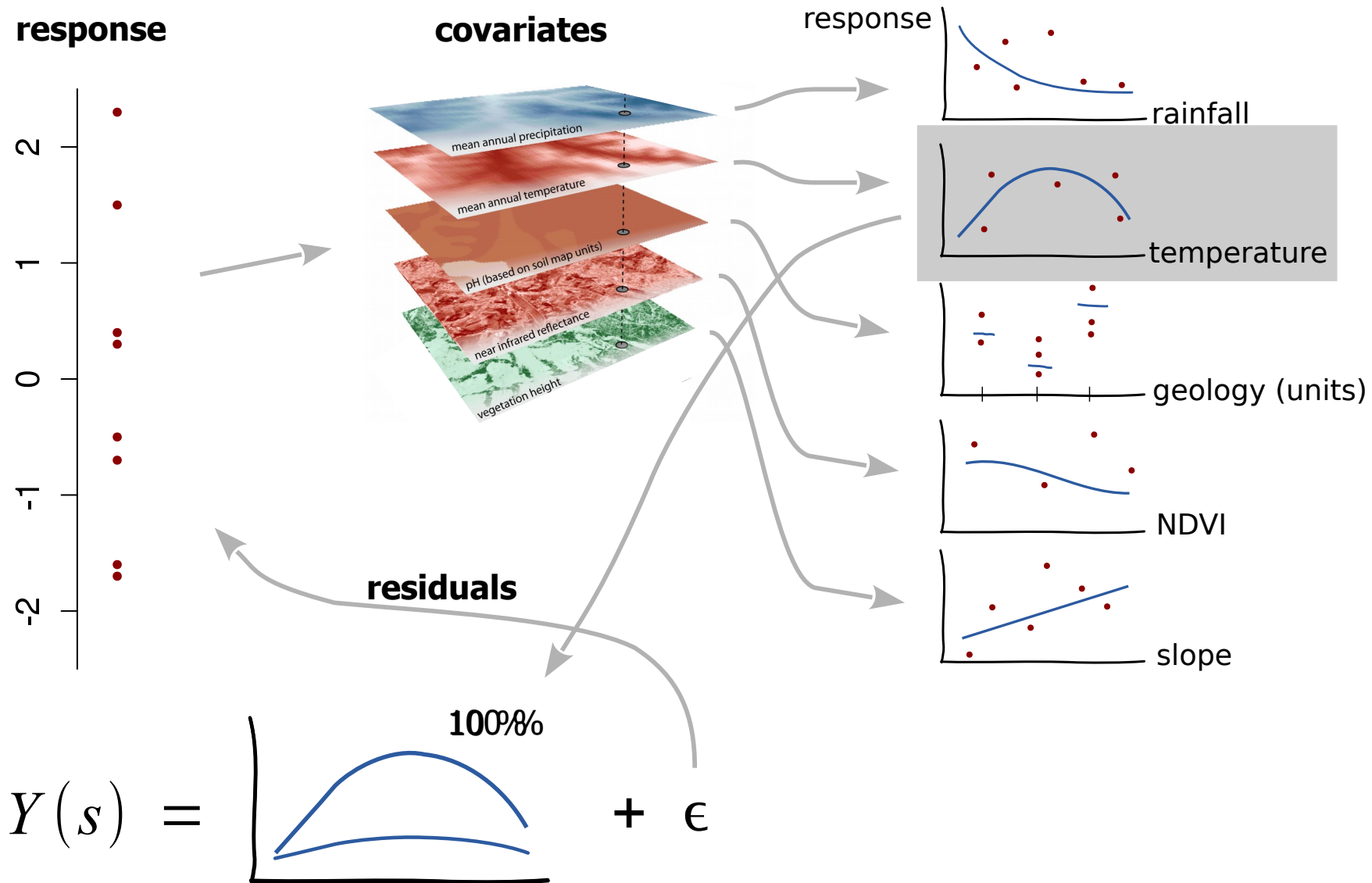
Gradient boosting: mini example

$$Y(s) = + + +$$

Gradient boosting:
linear, splines and spatial baselearners

linear non-linear spatial

$$Y(s) = f_{env}(X) + f_s(s) + f_{ns}(X,s)... + \epsilon$$

partial residuals

SD of slope in rad.20 m [deg.]
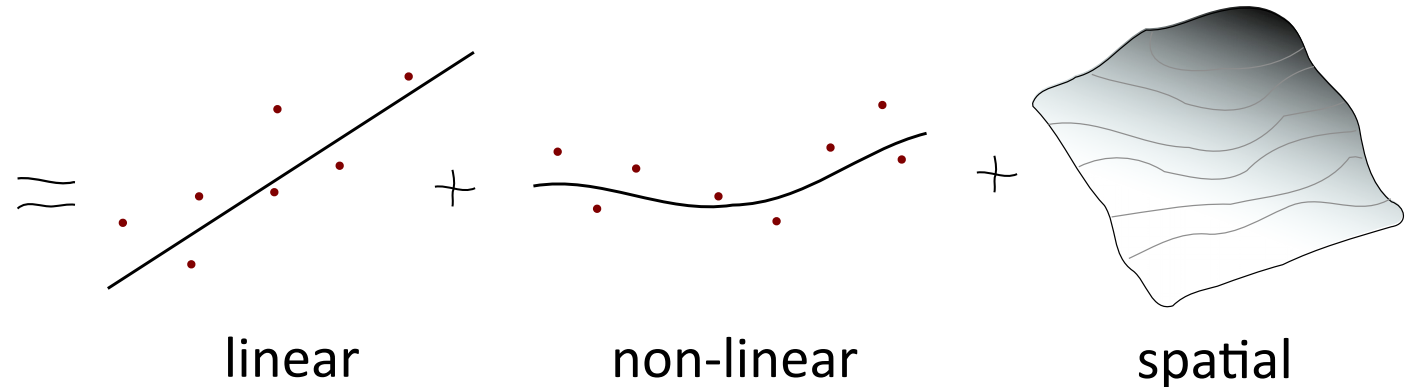
see e.g. Hothorn et al. 2011
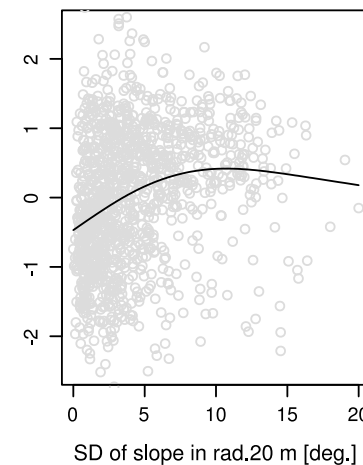
Gradient boosting: Spatial modelling with splines

Spatial autocorrelation can be modelled by including a „smooth spatial surface" as baselearner, non-stationary effects by creating interactions with the spatial surface
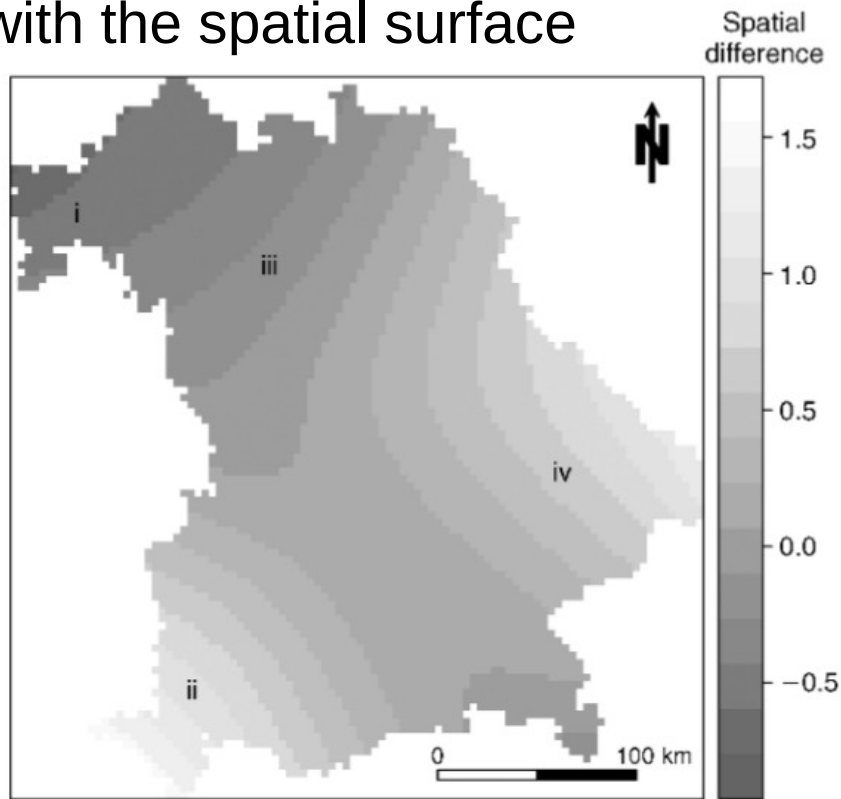
FIG. 6. Spatial difference in Red Kite breeding between 1979–1983 and 1996–1999 for model (add/vary). The breeding probabilities in the northwestern part decreased, while the southwestern part goes with increased breeding probabilities. For the four selected areas [(i) Unterfranken, (ii) Schwaben, (iii) Mittelfranken, and (iv) Niederbayern], the variability of the estimated spatial difference is shown in Fig. 7. Spatial differences can be interpreted as difference in log-odds ratios.
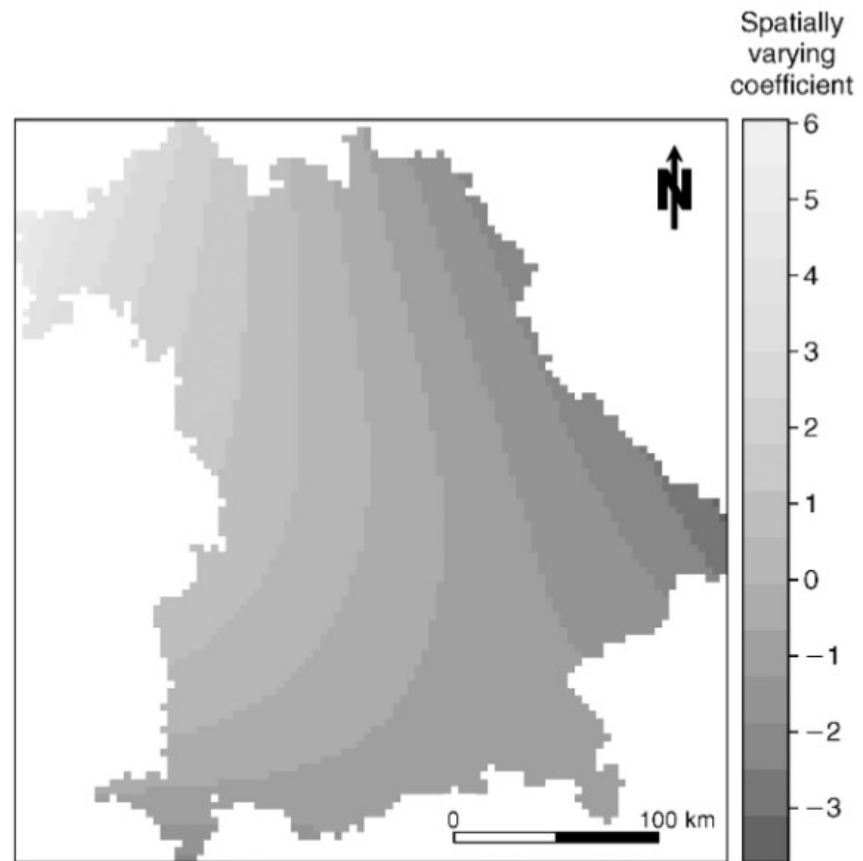
FIG. 8. Spatially varying coefficients for altitude in Red Kite breeding model (add/vary); here altitude was standardized to the unit interval. Altitude has a positive effect in the western and northwestern part, while its effect is zero or even negative in the rest of Bavaria.

Hothorn et al. 2011

Gradient boosting with splines baselearner

- ✔ Finally a ML method that explicitly models spatial surfaces and non-stationarity!

- ✔ Selects covariates (but not very rigorous)

- ✔ Simple interpretation of non-linear relationships

- ✖ Not so fast, needs a lot of setup for fitting
  Because of the difficult setup: not part of the excercises of this course.

- ✖ Unfair/biased selection of categorical covariates

- ✖ Interpretation of covariate importance difficult, if no strong selection

- ✖ Parametric method: transformations, extrapolation errors

- ✖ Prediction uncertainty only with bootstrap

Should I use gradient boosted trees or random forests?

Boosted trees
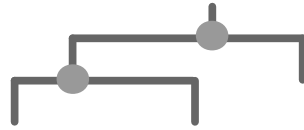
- ✔ Selects covariates weakly

- ✔ Covariate importance
  for interpretation and maybe selection

- ✖ Predictive accuracy slightly lower than random forest

- ✖ Prediction uncertainty only by bootstrapping

- ✔ Reduces bias by fitting on residuals

Speed?

Do some benchmarking if interested ;-)

Random forest
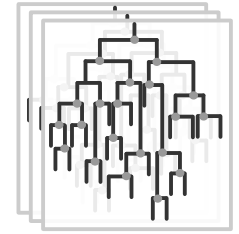
- ✖ Does not select covariates (only with extra implementation)

- ✔ Covariate importance
  for interpretation and maybe selection

- ✔ From my datasets on average most often best performance
  (best for 50 responses of 85 tested for soil mapping)

- ✔ Prediction uncertainty with quantile regression forest

- ✖ Always fits on data with same distribution

Content of lecture

Terms and concepts

Spatial modelling: requirements?

Overview of ML and their strategies

Side note: overfitting

Methods
- Lasso
- Support Vector Machines
- Ensemble Learners
  - Random Forest
  - Boosting
  - Model averaging

Model averaging
# Ensembles of different methods

- Create predictions from different (ML) methods and **combine them**.

- Idea: each (ML) method as a mean of **reducing dimensions** in the dataset capturing different properties of the dataset → used methods should not be similar.

- Mathematical proofs show that combinations of different linear models result always in better performance. For other methods thats not a priori given, but very likely.

- <u>Strategies</u>
    - just take mean for every prediction
    - weighted mean, weights from model performance e.g. $\dfrac{1}{MSE}$
    - local weights with uncertainties of each method and prediction
    - linear fit with predictions as covariates and original data as response → but take car, never fit on validation set!!
    - or stacked generalisation, Bayesian approach

One last problem:
ML methods are not spatial ...
how to deal with spatial autocorrelation?

- Apply geostatistics on residuals
  often named "regression kriging", requires two
  independent model fits

- Add spatial coordinates as covariates to model
  - Add X and Y coordinates, might lead to artefacts
    → predicted maps look like a chessboard
  - Add rotated coordinates (ad-hoc method)

- Add smooth surface of coordinates (tensor splines),
  Works for generalized additive models (GAM) or boosting with splines
  baselearner.
  Might overfit or if constrained not catch the spatial structure in the
  data.

- Preliminary conclusion:
  None of these options is satisfactory.
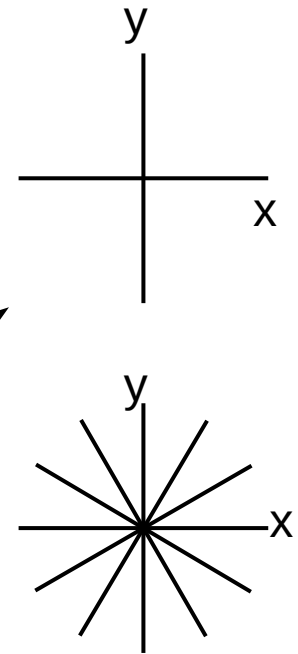
Summary
Can you make sense of all these words now?

boosting

lasso

weak learner

shrinkage   cross-validation

forward selection

support vector machines

ensemble learners
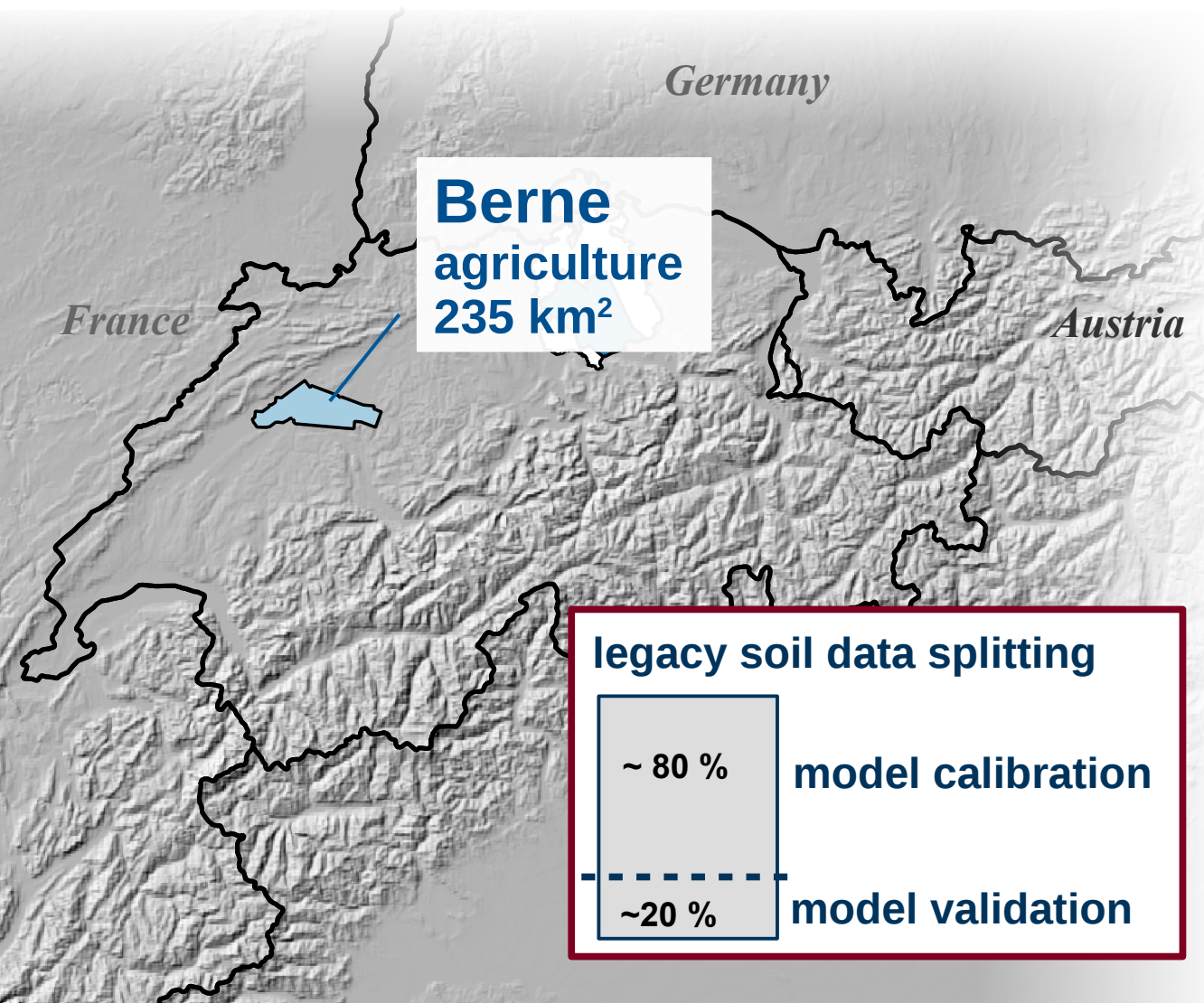
bias-variance tradeoff   **random forest**

bagging

overfitting   **model averaging**

Be able to judge if computing model averaging on 78 methods found in Package caret is a sensible thing to do …

Exercise:
Berne soil mapping study

~ 1000 sites with legacy soil data from 1970-1980
Nussbaum et al. 2017b

Berne
agriculture
235 km²

France

Germany

Austria

legacy soil data splitting

~ 80 % **model calibration**

~20 % **model validation**

Numerous covariates

Climate
different data sets
(monthly resolution)

Soil
soil overview map
historic wetlands
anthropogenic soil interventions
drainage networks

Parent material
(hydro)geological maps
and derivates

Vegetation
Landsat, SPOT5, DMC mosaic
forest vegetation map and
species composition

Terrain
90 derived attributes
(multiple scales)

Exercises: Solve at your own speed...

OpenGeoHub-machine-learning-training-1.R   R: Plain R code

OpenGeoHub-machine-learning-training-1.Rnw   Rnw: knitr-file to create PDF

OpenGeoHub-machine-learning-training-1.pdf   PDF: Instructions

get materials:
git clone
https://github.com/mnocci/2019_OpenGeoHub_machine-learning-madlene.git