Business and Local Government
Data Research Centre

Introduction to Data Science

EXPLORING DATA
ENHANCING KNOWLEDGE
EMPOWERING SOCIETY

E·S·R·C
ECONOMIC
& SOCIAL
RESEARCH
COUNCIL

University of Essex

Business and Local Government Data Research Centre

Cutting-edge training delivered by leading experts in the field of data analytics brought to you by the Business and Local Government Data Research Centre

- Exploring data
- Enhancing knowledge
- Empowering society

Offering grant funded data analytics research projects, training, events, webinars and data consultation services.

To find out more please contact Laura.brookes@essex.ac.uk

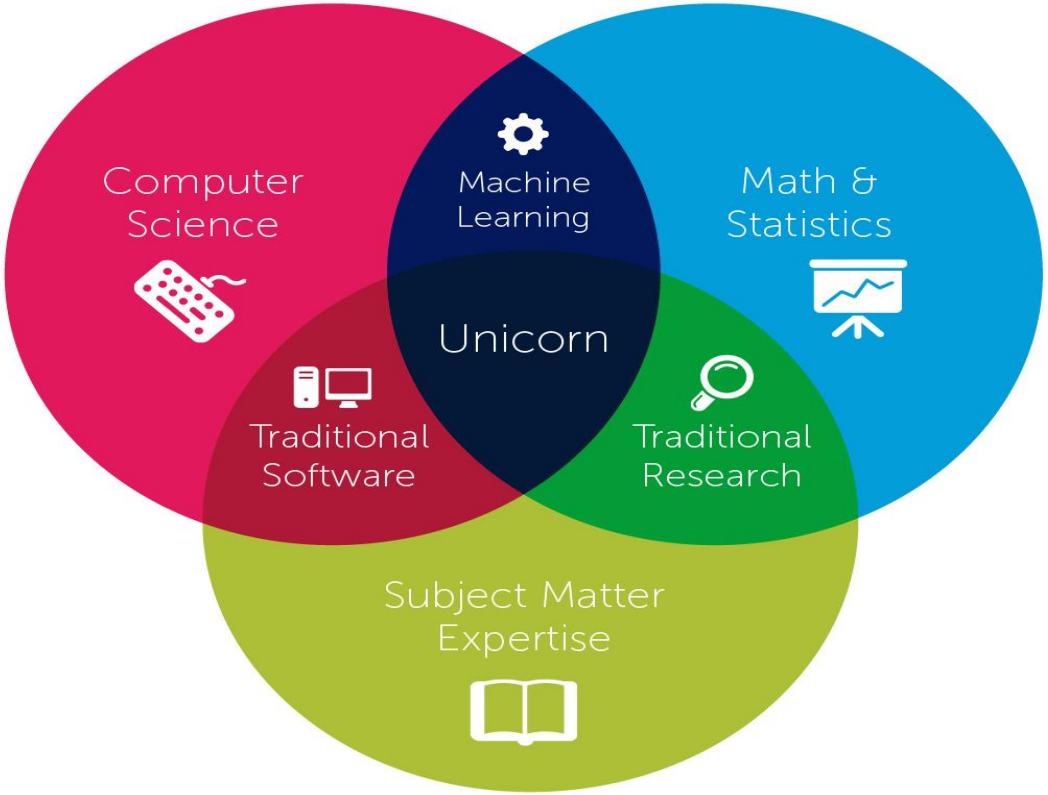
Agenda

1. What is Data Science?
2. Regression
3. Classification
4. Cross-validation
5. Subset selection
6. Non-linear models
7. Tree-based methods

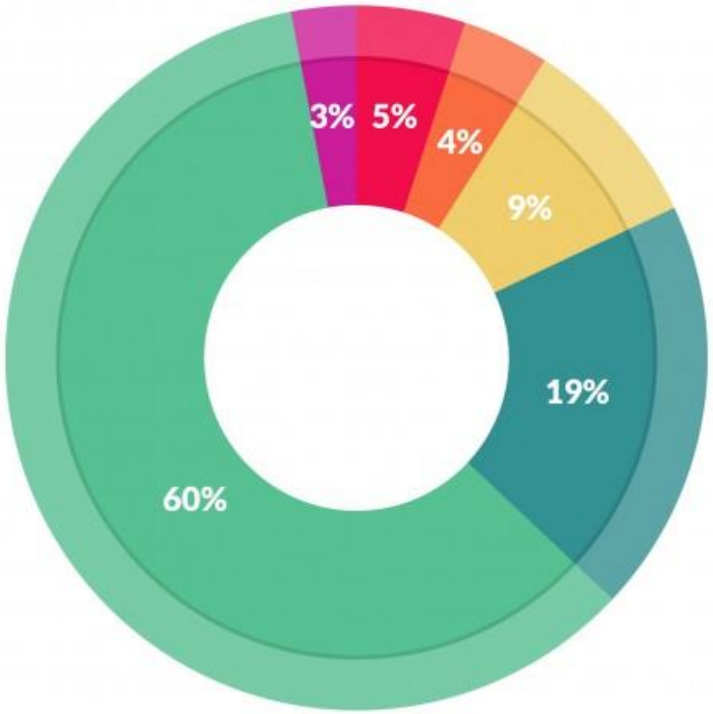
Introduction to the session

- For the hands-on session visit the following web page: https://esrc-blg.github.io/ml101

Data Science



Reality



What data scientists spend the most time doing

- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

Source: https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/#4a79a76c7f75

Machine/Statistical Learning Problems

- Prediction: E.g. heart attacks on the basis of demographic and clinical data
- Prediction: Classify an email into: spam – not spam
- Prediction: Identify the numbers in handwritten post code
- Prediction: Identify the best model to predict turnout and vote choice
- Prediction: Find the best predictors for income among demographic variables in a survey
- But causality is coming back in current ML research

Supervised Learning (Outcome is known)

- An outcome measurement *Y* (aka dependent variable, response, target, left-hand side variable)
- Vector of *p* predictor measurements *X* (aka inputs, regressors, covariates, features, independent variables, right-hand side variables)
- In the regression problem, *Y* takes values in a finite unordered set (e.g. price, blood pressure…)
- In the classification problem, *Y* takes values in a finite, unordered set (survived/died, digit 0-9, cancer class of tissue sample).
- We have training data (x_i, y_i),…,(x_N, y_N). These are observations (instances) of these measurements

Objectives of supervised learning

- On the basis of the training data we would like to:
  - Accurately predict unseen test cases
  - Understand which inputs affect the outcome, and how (although here we are straying into the territory of causal inference)
  - Assess the quality of our predictions and inferences

Unsupervised learning (no outcome measurement)

- No outcome variable, just a set of predictors (features) measured on a set of samples
- Find groups or clusters, were observations are similar within groups but differences across groups are large
- Difficult to assess quality
- Sometimes used as a pre-processing step for supervised learning
- Often used when working with textual data, e.g. to uncover topics in legislative speech

Philosophy

- It is important to understand the ideas behind various techniques, in order to know how and when to use them
- We need to understand the simpler methods first, in order to grasp the more sophisticated ones
- It is important to be able to accurately assess the performance of a method (simpler methods are often competitive and there is no one best method for all problems)
- If you are interested in learning whether a specific intervention has its intended effect, you need a causal inference class

The Netflix prize

- Competition started in Oct 2006. Training data is ratings for 18,000 moves by 400,000 Netflix customers and each rating is between 1 (worst) and 5 (best)
- Training data is very sparse – about 98% missing
- Objective: Predict the rating for a set of 1 million customer-movie pairs that are missing in the training data.
- Netflix's original algorithm achieved a root MSE of 0.953. The first team to achieve a 10% improvement wins $1,000,000
- Is this a supervised or an unsupervised problem?

Statistical learning v. machine learning

- Machine learning arose as a subfield of artificial intelligence
- Statistical learning arose as a subfield of statistics
- Much overlap – both fields focus on supervised and unsupervised problems:
  - Machine learning has a greater emphasis on large scale applications and prediction accuracy
  - Statistical learning emphasises models and their interpretability, precision and uncertainty
- Over time, the differences became less and less pronounced and machine learning has emerged as the general label

Prediction v. explaining relationships

- A causal inference problem is where we want to learn about the effect that *X* exerts on *Y*
  - E.g.: Do stop and searches reduce knife crime?
- A prediction problem is where we want to predict an outcome as accurately as possible
  - E.g.: Does a patient have cancer or not?
  - In a prediction problem, we do not necessarily need to establish causal relationships. However a truly causal model with always be a good predictive model (e.g. in weather forecasts all error comes from inaccurate measurement)

Assessing model accuracy

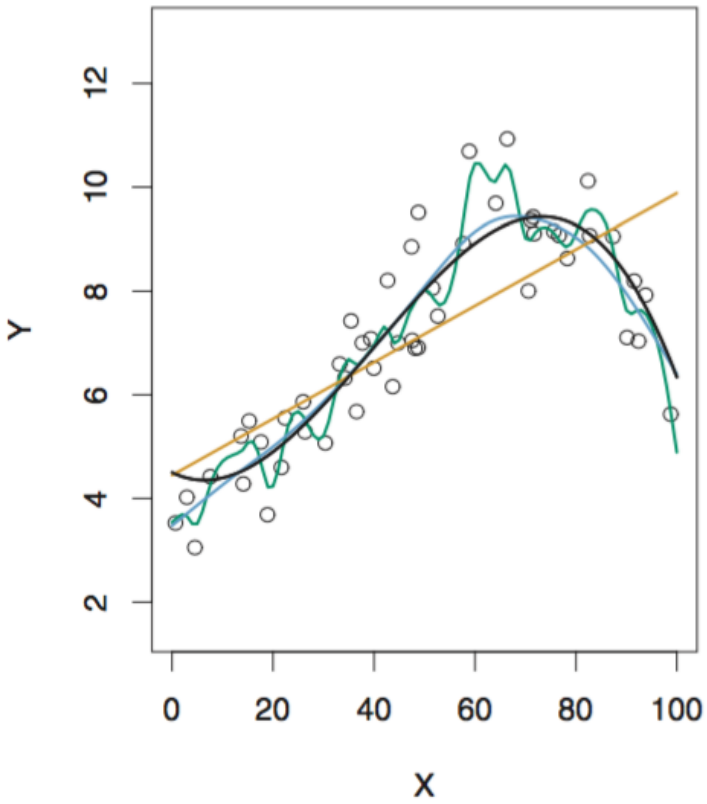
- In order to be able to select the best approach for a specific problem, we need to evaluate performance
- *The more dissimilar our prediction from the real outcome, the worse our performance but how do we assess that difference -> what is our loss function*
- *We can look at the mean squared error:*

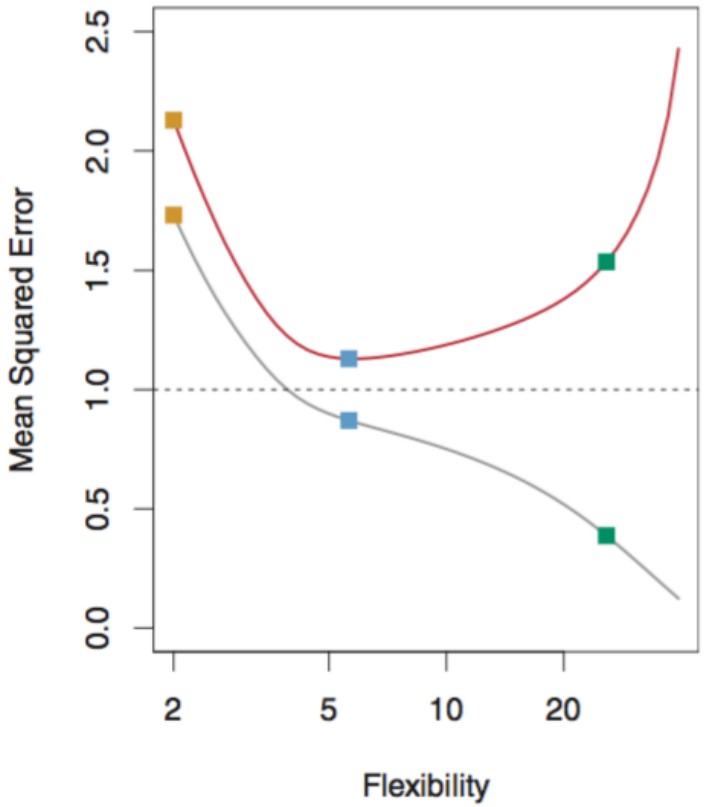
$$MSE = \frac{1}{n}\sum_{i=1}^{n}[y_i - \hat{f}(x_i)]^2$$

  - We determine $\hat{f}(x)$ on the training data and then generate the MSE on the test data

Variance-Bias Tradeoff 1

- If we choose models based on training MSE, we end up with bad predictions
- The problem is known as over-fitting:

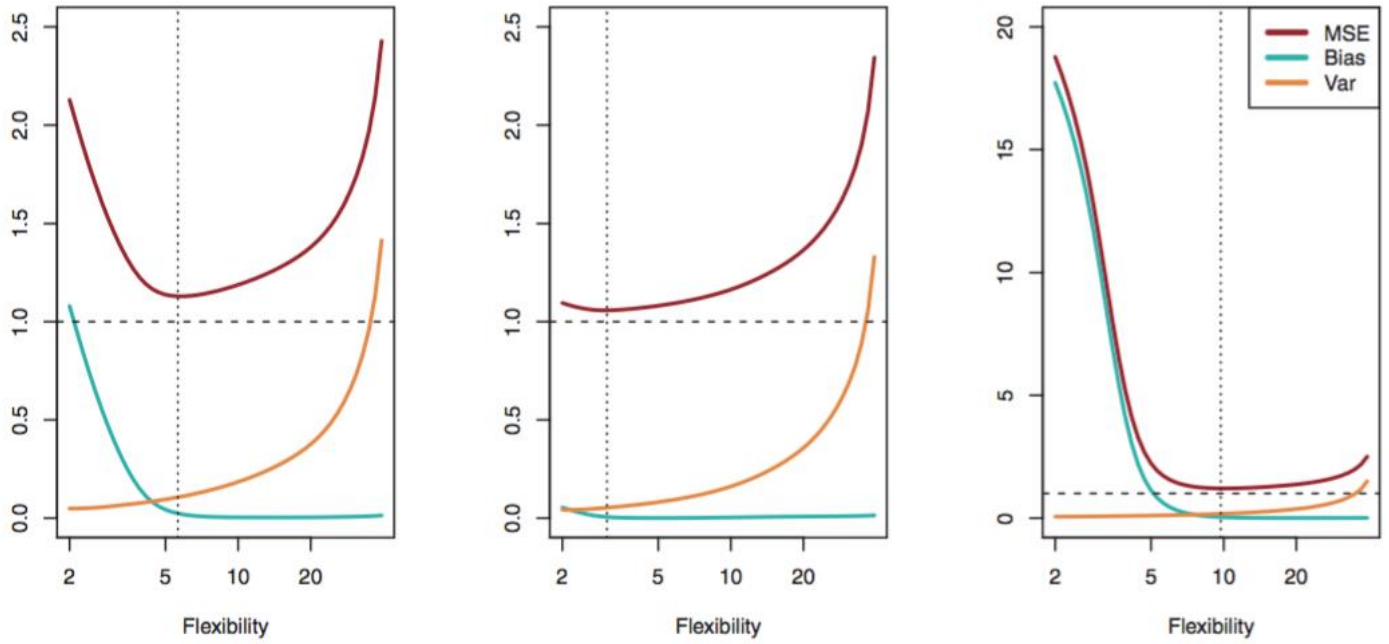




James et al. 2013: 22-24

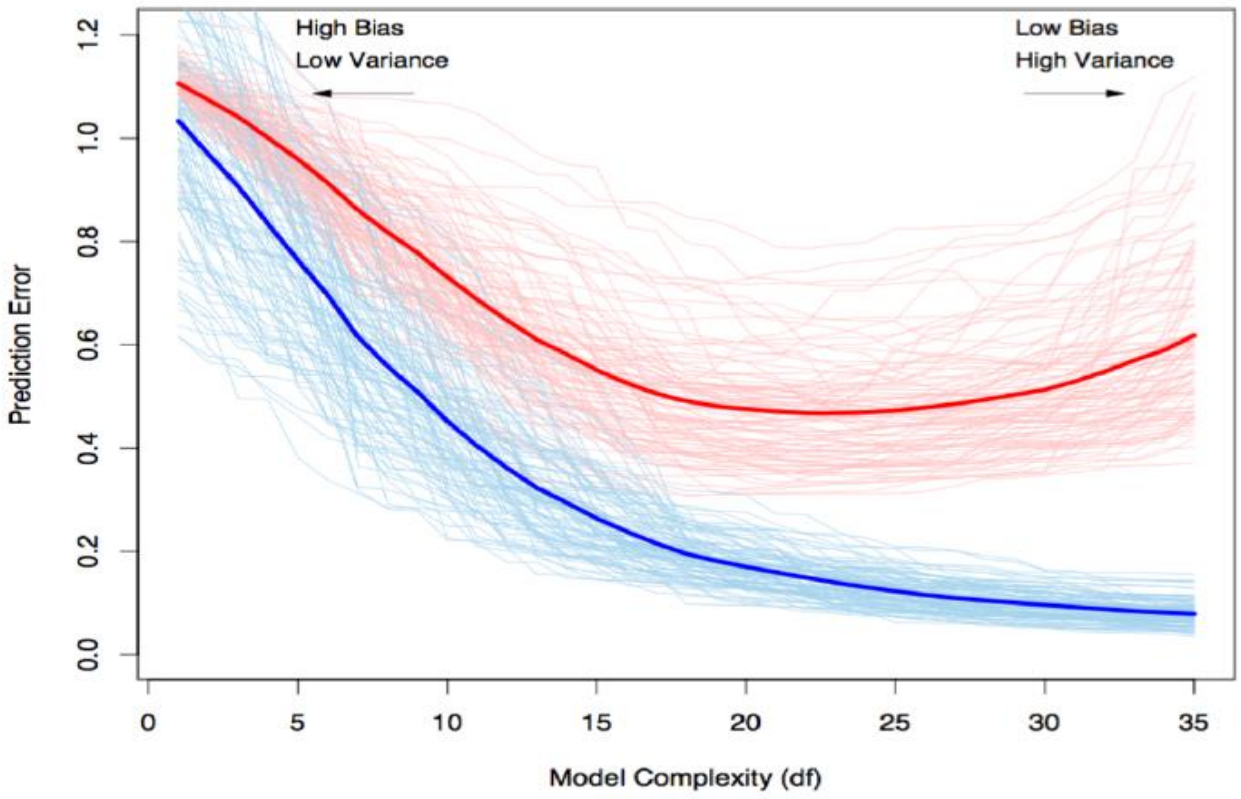
Variance-Bias Tradeoff 2

- Test $MSE = Var\left(\hat{f}(X)\right) + [Bias\left(\hat{f}(X)\right)]^2 + Var(\epsilon)$
- The V-B trade-off exists because there are two opposite principles at work:
- Bias: As the model becomes less complex, bias increases
- Variance: As the model becomes more complex, variance increases



James et al. 2013: 36

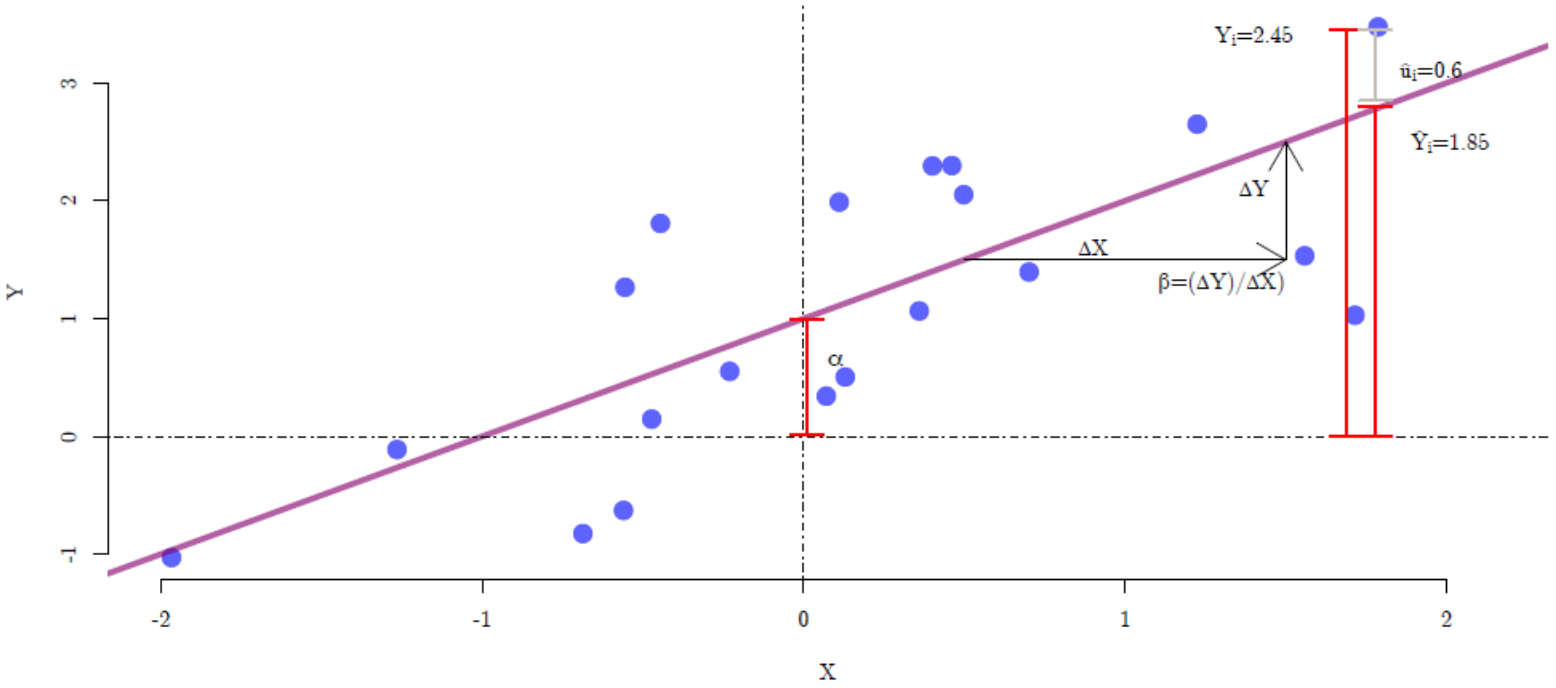
Variance-Bias Tradeoff 3



- Red = test error;
- blue = training error

Hastie et al. 2008: 220

Linear models



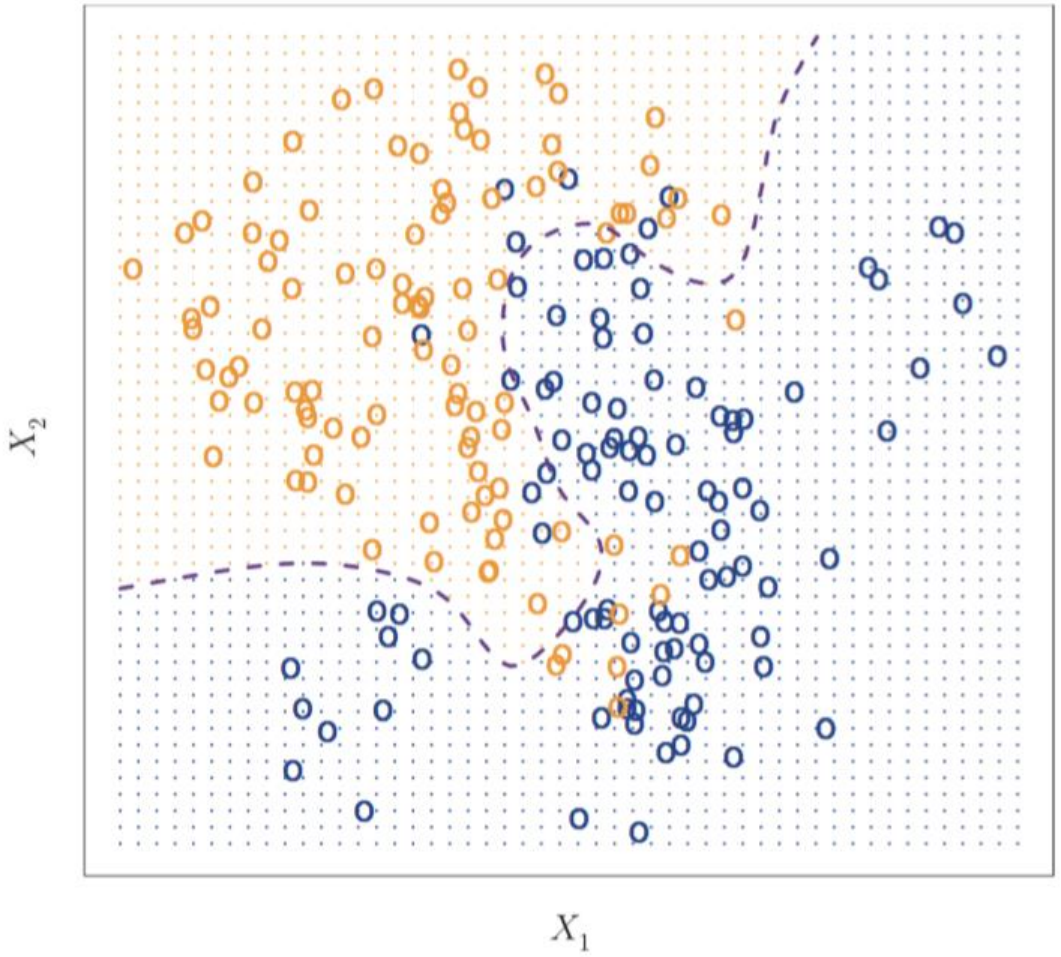
Classification

- When *Y* is not continuous but qualitative, we face a classification problem
- The goal is to predict the correct class of an observation based on context information *X*
- We asses the quality of classification via the error rate:

$$Error\ rate = \frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$$

- We prefer the classification that minimises the error rate in the test set

Classification

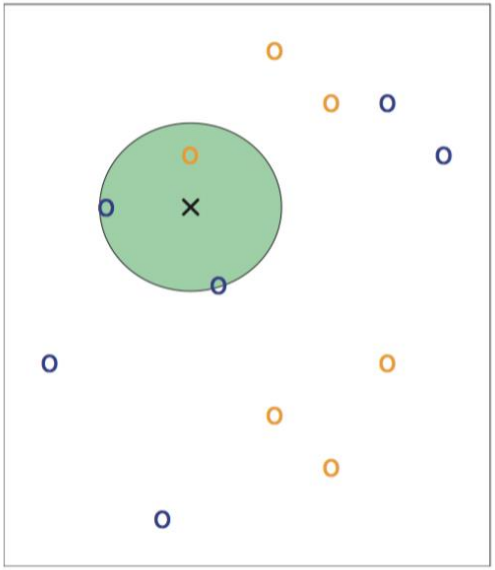


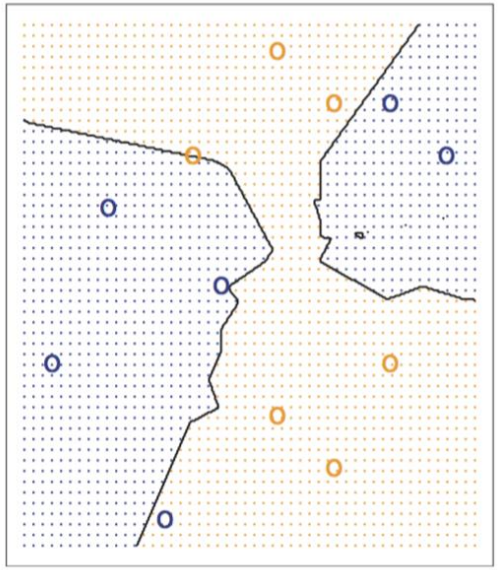
James et al. 2013: 38

Classification with K Nearest Neighbor (KNN)

- With KNN we look at the K closest observations and base our classification on them
- We assign the class for which this quantity is largest:

$$P(Y = j \mid X = x_0 = \frac{1}{k} \sum_{i \in N_0} y_i \in j)$$

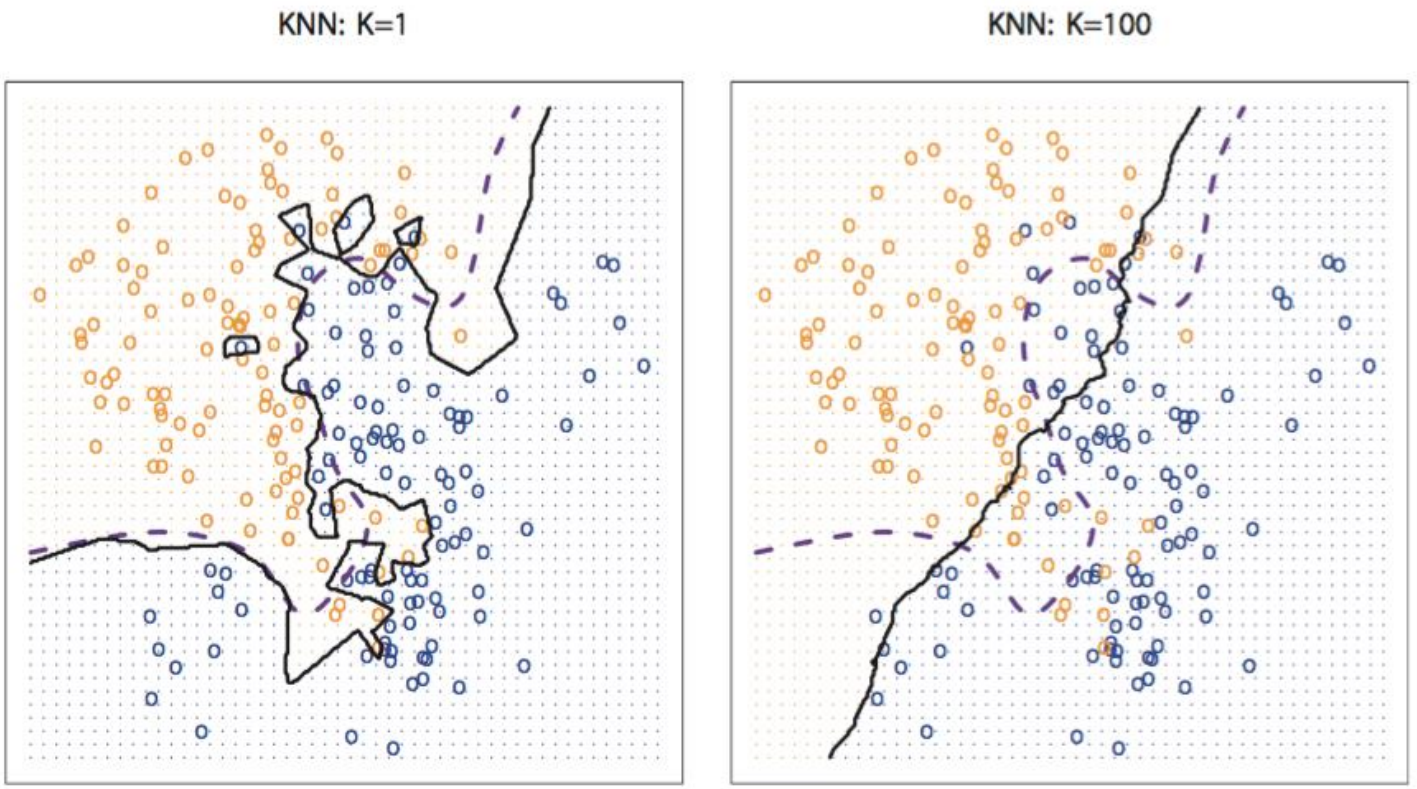




James et al. 2013: 40

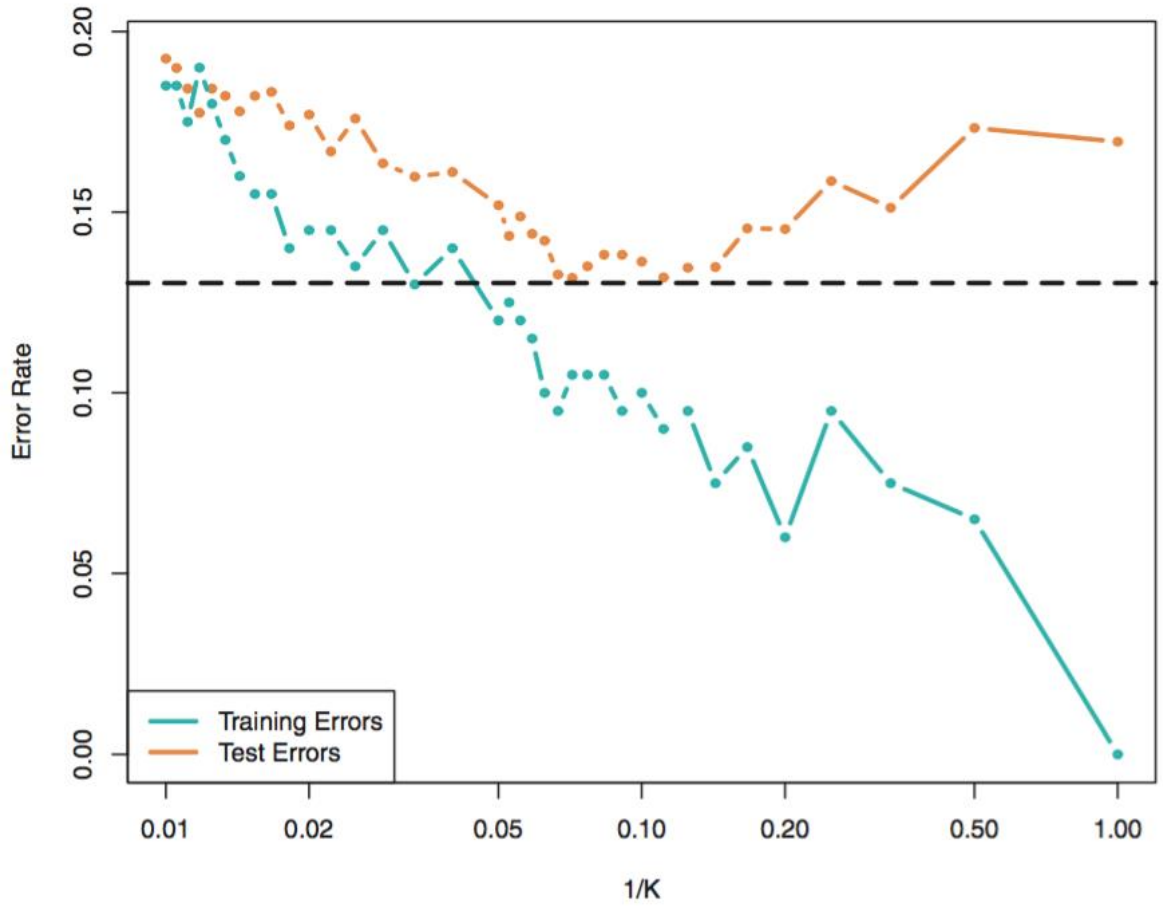
Classification with KNN 2

- The choice of *K* matters:



James et al. 2013: 41

KNN and the variance-bias tradeoff



James et al. 2013: 42

Cross-validation

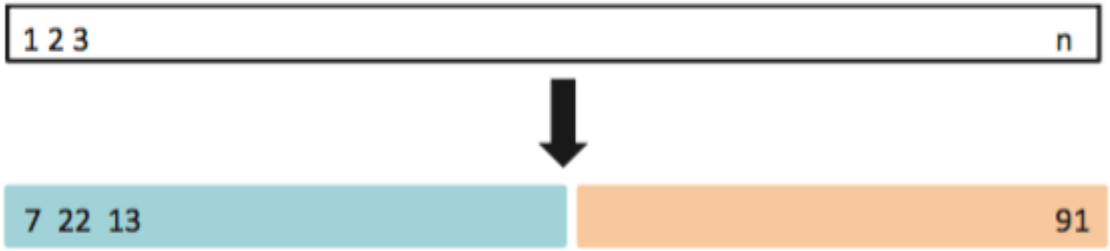
- Without having new data, we can split the data we have into training and test data – this called re-sampling
  - Re-sampling is computationally expensive

- Cross-validation methods:
  - Validation set approach
  - Leave-one-out cross-validation (LOOCV)
  - *k*-fold cross-validation

James et al. 2013: 42

Validation set approach

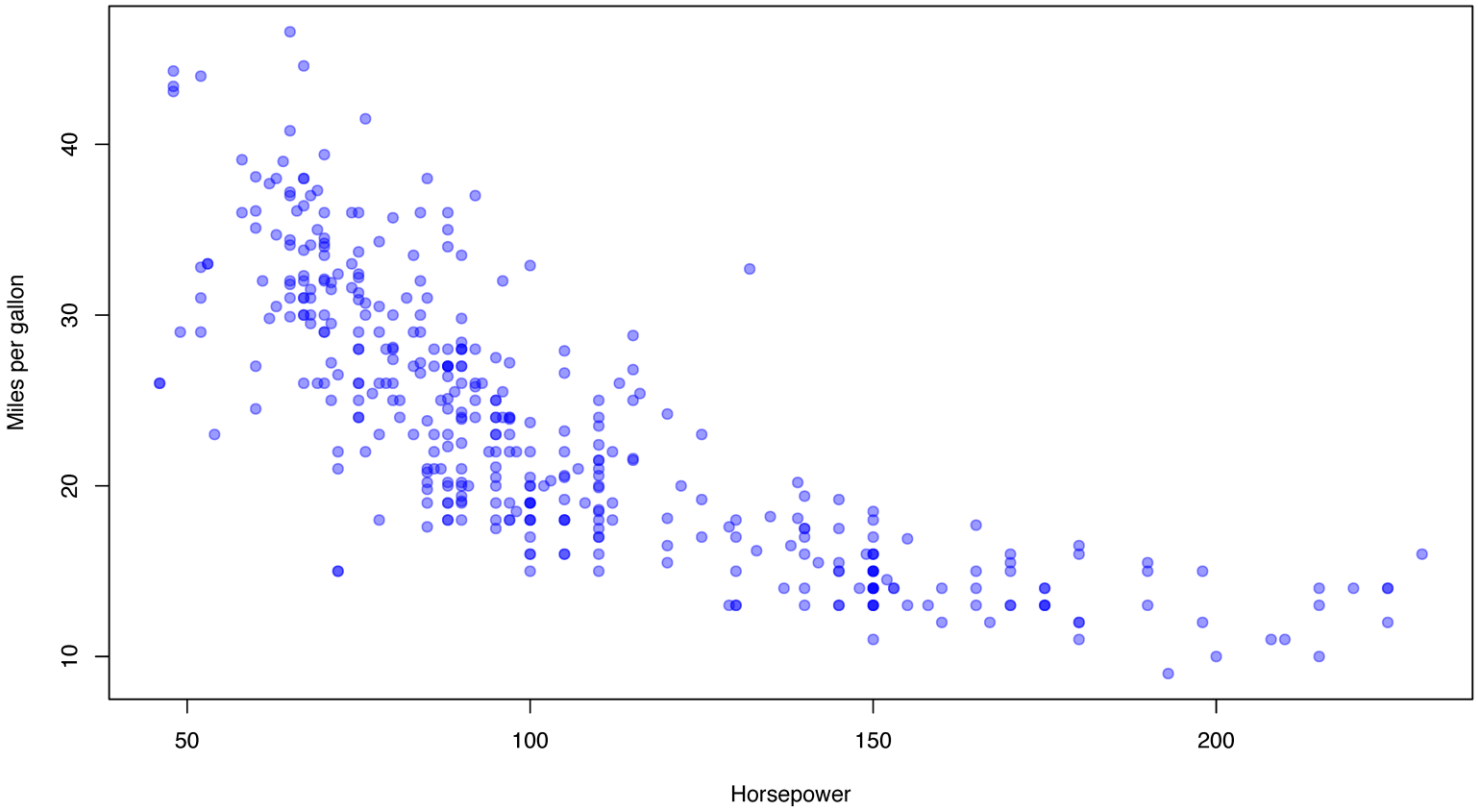
- Step 1: Split data in training and test sets at random
- Step 2: Pick the optimal model in the training set
- Step 3: Determine its performance on the test set



James et al. 2013: 177

Auto Example (James et al., chapter 3)

- Predict mpg with horsepower. But: How complex is the relationship?

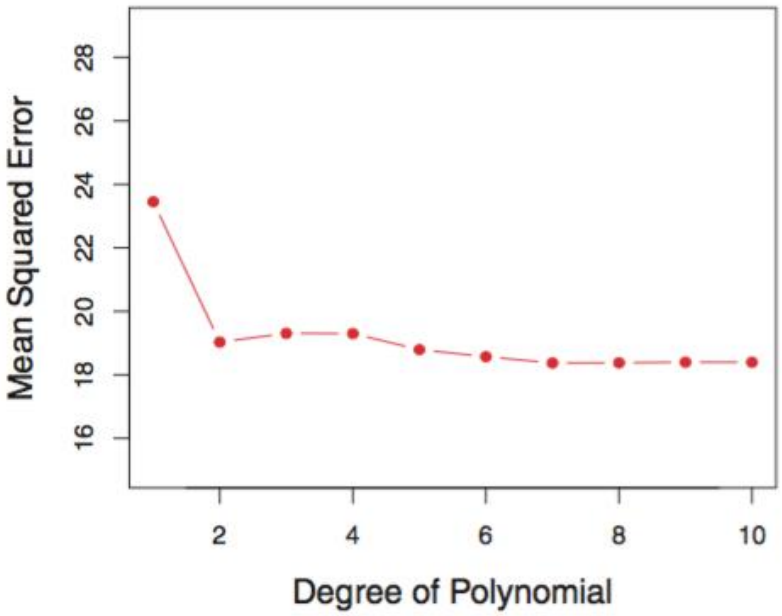


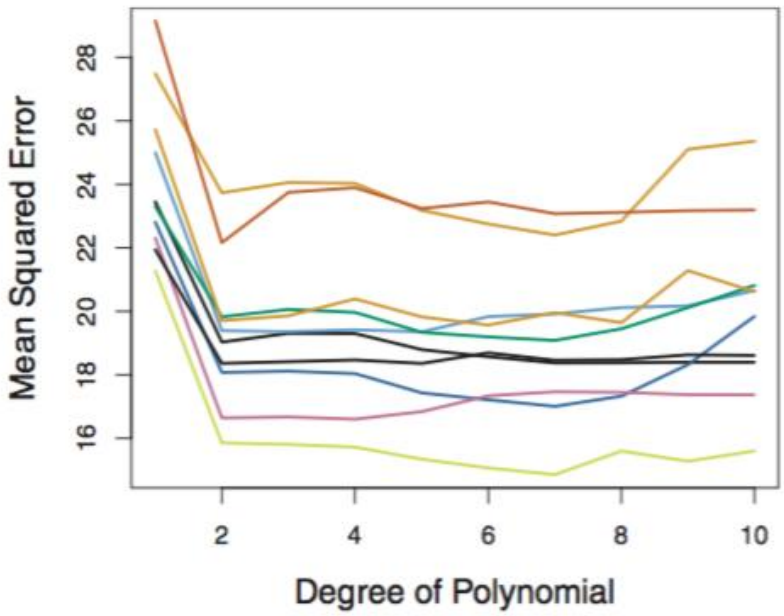
How many polynomials should we use?

	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7
(Intercept)	39.94 ***	56.90 ***	60.68 ***	47.57 ***	-32.23	-162.14 *	-489.06 *
	(0.72)	(1.80)	(4.56)	(11.96)	(28.57)	(71.43)	(189.83)
horsepower	-0.16 ***	-0.47 ***	-0.57 ***	-0.08	3.70 **	11.24 **	33.25 **
	(0.01)	(0.03)	(0.12)	(0.43)	(1.30)	(4.02)	(12.51)
horsepower2		0.00 ***	0.00 *	-0.00	-0.07 **	-0.24 **	-0.85 *
		(0.00)	(0.00)	(0.01)	(0.02)	(0.09)	(0.34)
horsepower3			-0.00	0.00	0.00 **	0.00 *	0.01 *
			(0.00)	(0.00)	(0.00)	(0.00)	(0.00)
horsepower4				-0.00	-0.00 **	-0.00 *	-0.00 *
				(0.00)	(0.00)	(0.00)	(0.00)
horsepower5					0.00 **	0.00 *	0.00 *
					(0.00)	(0.00)	(0.00)
horsepower6						-0.00 *	-0.00
						(0.00)	(0.00)
horsepower7							0.00
							(0.00)
R^2	0.61	0.69	0.69	0.69	0.70	0.70	0.70
RMSE	4.91	4.37	4.37	4.37	4.33	4.31	4.30

*** p < 0.001, ** p < 0.01, * p < 0.05

Validation set approach applied to Auto data





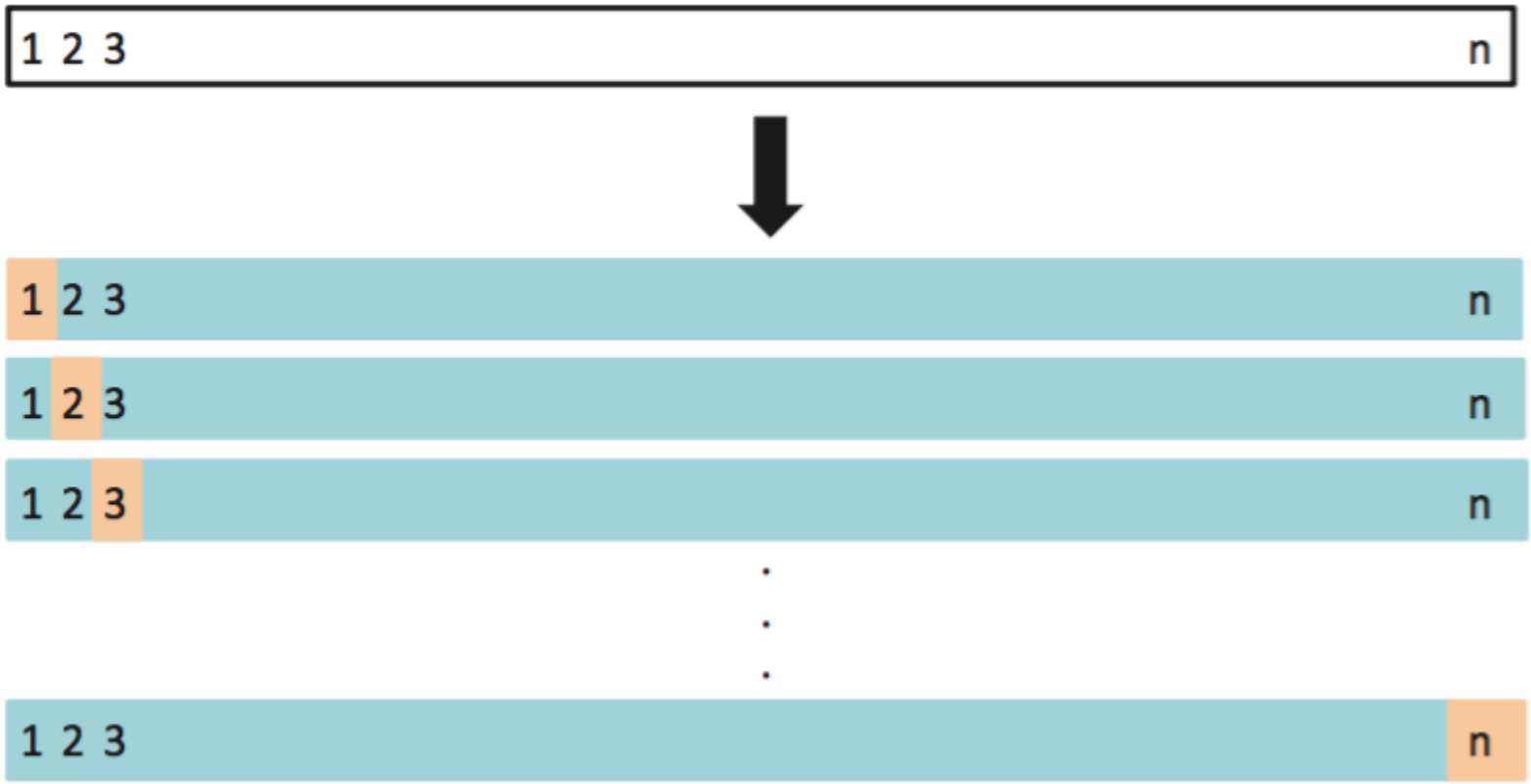
James et al. 2013: 178

- Validation approach: highly variable results (right plot)
- Validation approach may tend to over-estimate test error due to small size of the training data

LOOCV

- Validation set approach had 2 disadvantages: (1) the error rate is highly variable and (2) a large part of the data is not used in training (fitting) the model
- Alternative approach: Leave-one-out cross-validation
- Leave out 1 observation and estimate model, assess the error rate (MSE_i)
- Average over all n steps, $CV = \frac{1}{n} \sum_{i=1}^{n} MSE_i$

LOOCV 2



James et al. 2013: 179

LOOCV 3

- Advantages:
  - Less bias than validation set approach – will not over-estimate the test error
  - The *MSE* of *LOOCV* does not vary over several attempts
- Disadvantage:
  - Model needs to be estimated as many times as we have observations in the dataset, i.e. n times.
  - However for LS linear or polynomial models there is a shortcut for LOOCV:

$$CV_{LOOCV} = \frac{1}{n}\left(\frac{y_i - \hat{y}_i}{1 - h_i}\right)$$

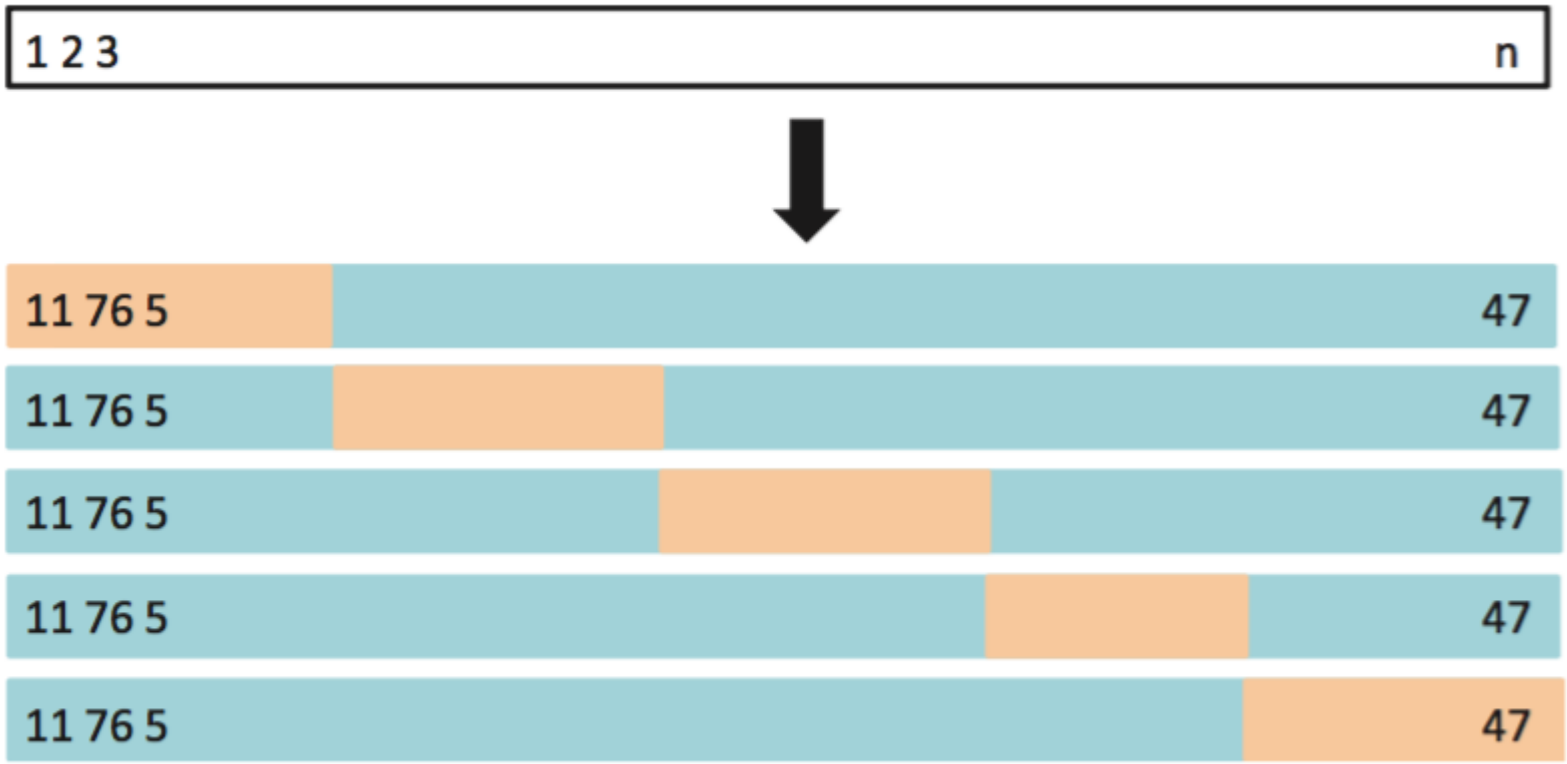
  - Where h_i is the leverage of observation i

k-fold cross-validation

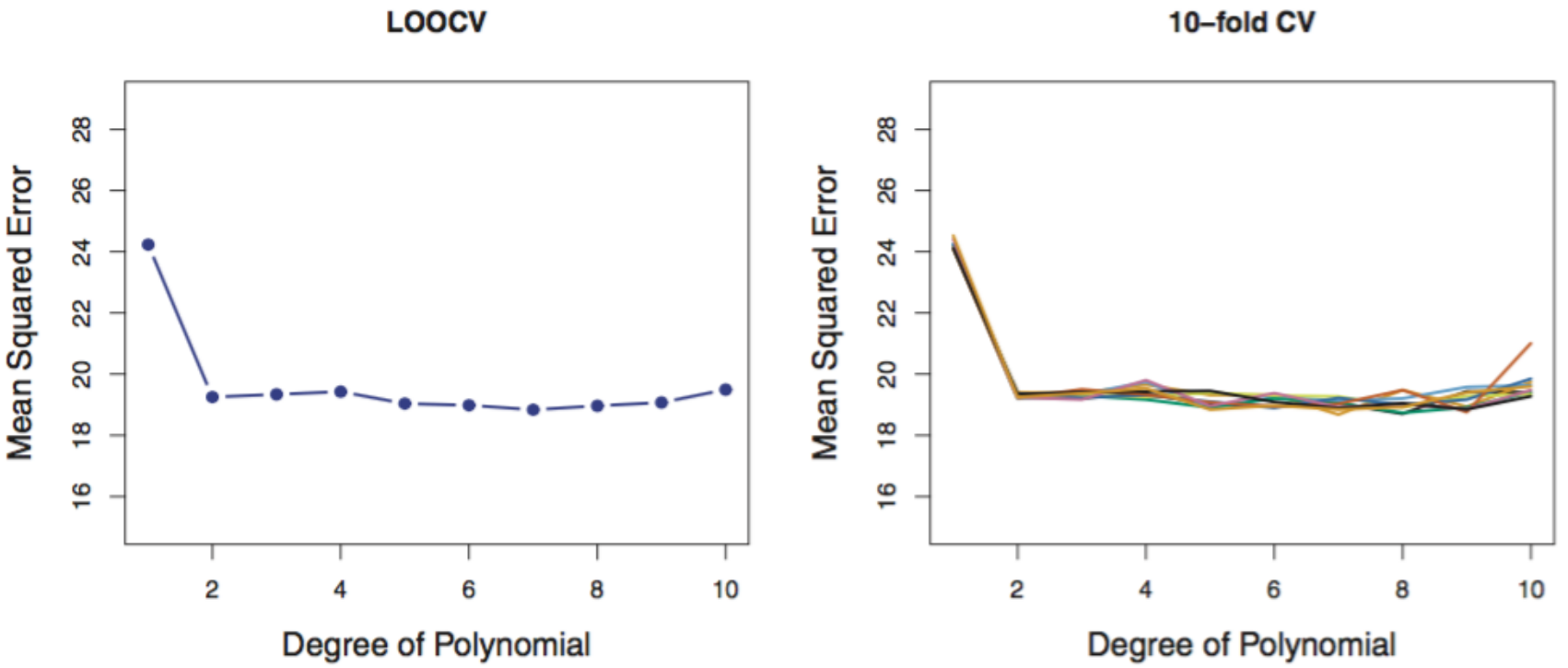
- Compromise between validation set and LOOCV is *k*-fold cross-validation
- We divide the dataset into *k* different *folds*
  - Usually $k = 5$ or $k = 10$
- We then estimate the model on $d - 1$ folds and use the k^{th} fold *as test dataset*

$$CV_k = \frac{1}{k}\sum_{i=1}^{K} MSE_i$$

k-fold cross-validation



k-fold cross-validation v. LOOCV



James et al. 2013: 180

- Similar error rates, but 10-fold CV is much faster

Variance-bias tradeoff

- LOOCV and *k*-fold CV lead to estimates of the test error
- LOOCV has almost no bias, *k*-fold CV has small bias (since not $n-1$ but only $\frac{k-1}{kn}$ observations used for estimation)
- But, LOOCV has higher variance since all n data subsets are highly similar and hence the estimates are more highly correlated than in *k*-fold CV
- Variance-bias tradeoff: We usually rely on *k*-fold for $k = 5$ or $k = 10$

CV the solution to all our problems?

- CV often works very well
- However, CV requires that folds are independent
  - This can lead to problems in hierarchical data where observations are cross-nested and in time series data where observations in t are not independent from observations in $t-1$
- One alternative or complementary approach is model averaging, e.g. using Ensemble Bayesian Model Averaging or Stacking

Regularization

- Ridge Regression & Lasso

- We fit a model on all *p* predictors using a technique that constrains or regularizes the coefficient estimates, or equivalently, that shrinks coefficient estimates towards 0
- It is an empirical finding, that shrinking coefficient estimates can significantly reduce variance

Regularization

- Recall that the least squares fitting procedure estimates $\beta_0, \beta_1, \ldots, \beta_n$ using the parameter values that minimize:

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij})^2 = RSS$$

- In other words, it minimizes the sum of the squared prediction errors (OLS -> ordinary least squares)
- In contrast, the regularization approach minimizes:

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij})^2 + \lambda f(\beta_j) = RSS + \lambda f(\beta_j)$$

- where $\lambda \geq 0$ is a tuning parameter, to estimated separately

Ridge regression

- Ridge regression minimizes the following expression

$$\underbrace{\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij})^2}_{\text{Standard OLS estimate}} + \underbrace{\lambda \sum_{j=1}^{J}\beta_j^2}_{\text{penalty}}$$

- Different values for λ lead to different model predictions (λ is a tuning parameter)
  - When λ is large, estimates get pushed towards 0
  - When λ is 0, ridge regression and OLS are identical
- We can find an optimal value for λ by relying on cross-validation

Ridge regression

- Shrinkage not applied to model constant β_0, model estimate for conditional mean should be un-shrunk
- Ridge regression is an example of ℓ_2 regularization

- $\ell_1 = f(\beta_j) = \sum_{j=1}^{J} |\beta_j|$
- $\ell_2 = f(\beta_j) = \sum_{j=1}^{J} \beta_j^2$

Ridge regression

- The least squares coefficients are scale equivariant: multiplying X_j by a constant c simply leads a scaling of the least coefficients by the factor $1/c$. In other words, regardless of how the j^{th} predictor is scaled, $X_j\hat{\beta}_j$ remains the same
- By contrast, the ridge regression coefficient estimates can change substantially when multiplying a given predictor by a constant, due to the penalty
- Therefore, the predictors must first be standardized before performing ridge (this also applies to the Lasso):

$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n}(\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2}}$$

Ridge regression

- OLS (linear regression) with p predictors has low bias (full information of all predictor variables is used)
- In ridge, predictors are shrunk (depending on the size of the penalty – think of it like a budget constraint)
  - Thereby, ridge reduces variance at the cost of increased bias

The Lasso

- The disadvantage of ridge regression is that all predictors are included in the final model (unlike for instance in best subset selection)
- The Lasso uses the ℓ_1 norm to overcome this problem – depending on the size of λ, predictors with less predictive power may be shrunk to exactly zero (excluded)

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij})^2 + \lambda \sum_{j=1}^{J} |\beta_j| = RSS + \lambda \sum_{j=1}^{J} |\beta_j|$$

Lasso v Ridge

- Lasso: Least absolute shrinkage and selection operator
- Ridge regression includes all variables, Lass will set some $\hat{\beta}_j = 0$, hence it is also a selection estimator (models are easier to interpret)
- Combines subset selection and regularization

- Which is better?
- If DGP has many equally relevant predictors, ridge will be better suited for prediction
- If some variables are highly correlated, Lasso helps with feature (variable) selection

Tree based models

1. Trees
2. Bagging/Random forests
3. Boosting

Tree-based methods

- We stratify or segment the predictor space into a number of regions
- Since a set of splitting rules used to segment the predictor space can be summarised in a tree, these types of approaches are also known as decision-tree methods

Pros and Cons

- Tree-based methods are simple and useful for interpretation
- However, they typically are not competitive with the best supervised learning approaches in terms of prediction accuracy
- Extensions such as bagging, random forests and boosting are very competitive. These methods grow multiple trees (ensembles) which are then combined to yield a single consensus prediction
- Combining a large number of trees can often result in dramatic improvements in prediction accuracy, at the expense of some loss in ease of interpretation

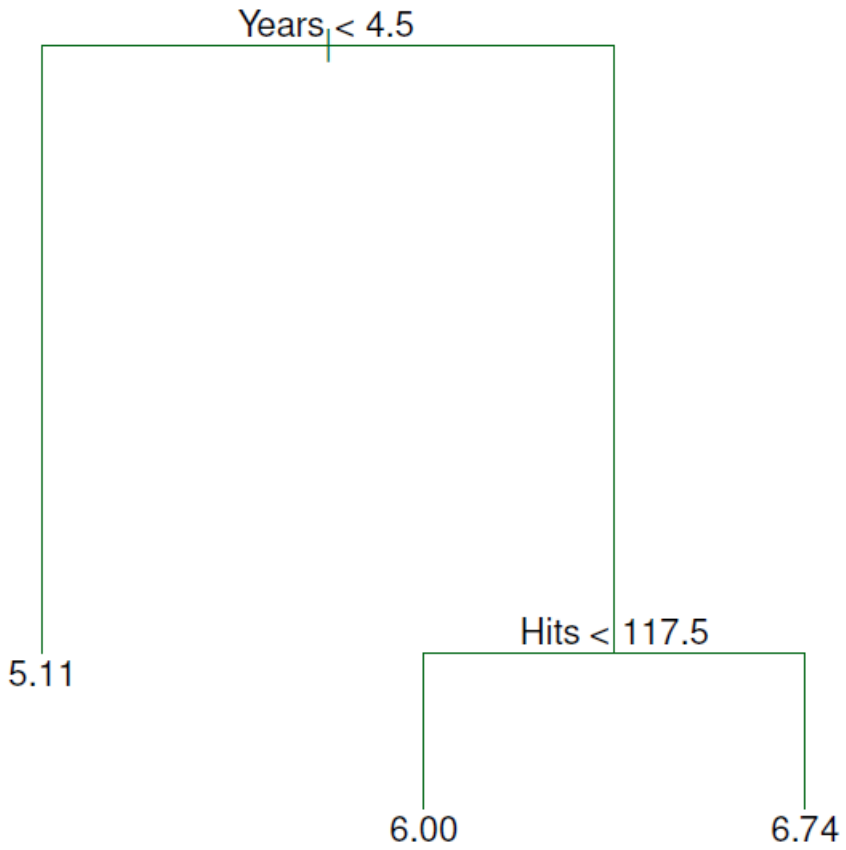
The basics of decision trees

- Decision trees can be applied to both regression and classification problems
- We first consider regression and then move to classification

Baseball salary data

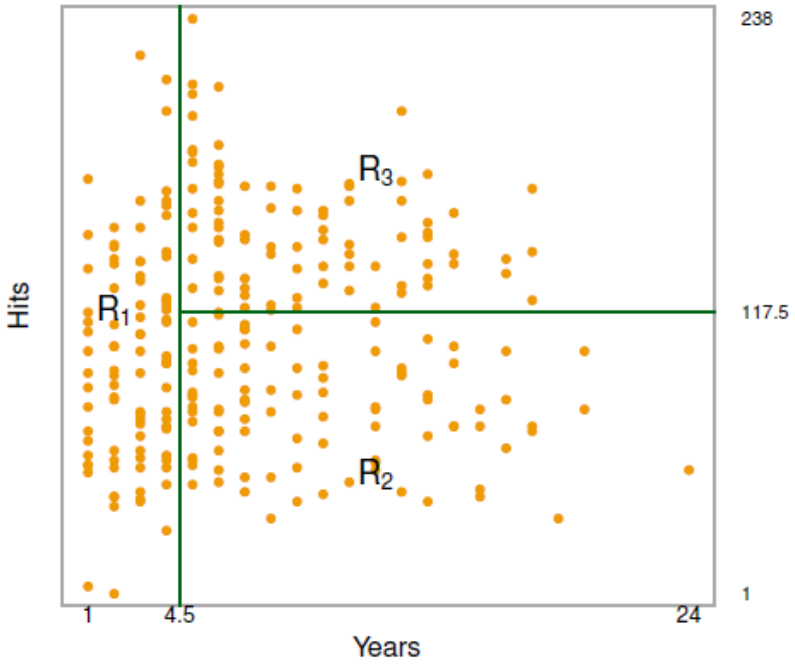
- We predict hitters logged salary based on the number of years played in major leagues and number of hits in previous year
- At a tree's node a condition is defined (e.g. Years < 4.5). At the left branch the condition is fulfilled and at the right branch the condition is not fulfilled
- The following tree has 3 terminal nodes or leaves and the number in each leaf is the mean of the response for the observations that fall there

Baseball salary data



James et al. 2013: 304

Results



Hastie et al. 2008: 305

- Tree stratifies players into regions of the predictor space:
  - $R_1 = \{X|Years < 4.5\}$
  - $R_2 = \{X|Years > 4.5, Hits < 117.5\}$
  - $R_3 = \{X|Years > 4.5, Hits \geq 117.5\}$

Tree terminology

- The regions R_1, R_2, R_3 are known as terminal nodes or leafes
- Decision trees are typically drawn upside down
- The points where the predictor space is split are referred to as internal nodes
  - In the baseball example, the internal nodes are indicated by the text $Years < 4.5$ and $Hits < 117.5$

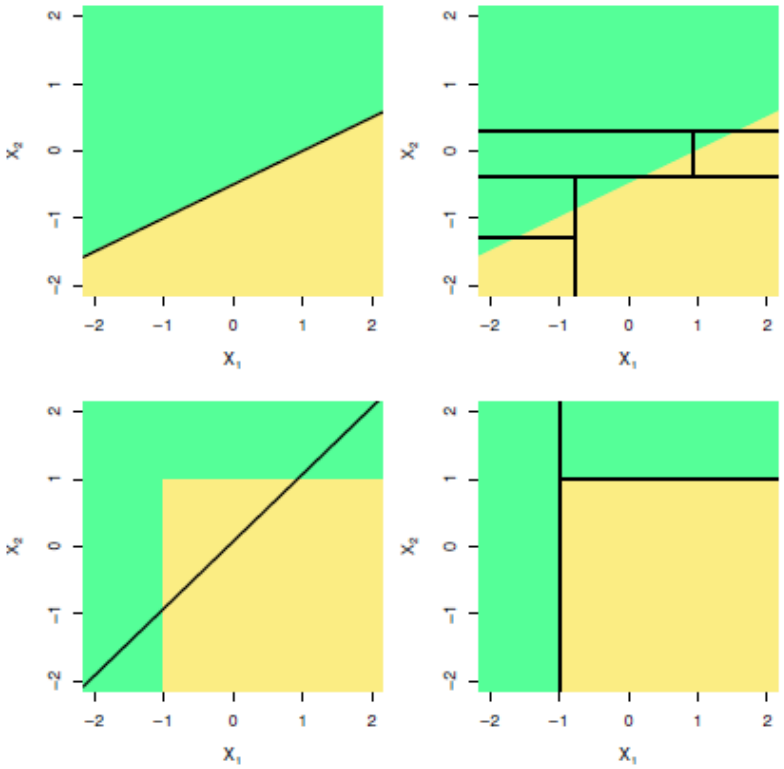
Interpretation

- *Years* is the most important factor; players with less experience earn less
- For less experienced players, the number of *Hits* in the previous year does not make a difference
- *But players who have been in the major leagues for five or more years, earn more when they hit more than 117.5 hits*
- *While this might be an over-simplification, the interpretation is clear*

Pruning a tree

- Trees are prone to over-fitting leading to poor test set performance
- A smaller tree with fewer splits might lead to lower variance and better ease of interpretation at the cost of some bias
- In cost-complexity pruning, we grow a full tree and then remove splits that add the smallest improvement
- The tuning parameter α controls the tradeoff between a subtree's complexity and its fit to the training data
- We select α in cross-validation

Trees v linear models



Hastie et al. 2008: 315

- Top row: True linear boundary; bottom row: True non-linear boundary
- Left column: linear model; right column: tree-based model

Bagging

- Bootstrap aggregation, or bagging, is a general-purpose procedure for reducing the variance of a statistical learning method; it is particularly often used with trees
- Given a set of n independent observations $Z_1, \ldots, Z_n$ each with variance σ^2, the variance of the mean $\bar{Z}$ of the observations is given by σ^2/n
  - In words, averaging a set of observations reduces variance. That approach is not typical because we generally do not have access to multiple training sets

Bagging 2

- Instead, we can bootstrap
  - Bootstrapping means repeatedly sampling from a sampling with replacement where the sample has the same size as the data it is sampled from (hence the need for sampling with replacement)
- In this approach, we generate a large number of bootstrapped datasets and at fit a tree to each of them. Then we average over all the predictions

Out-of-bag (OOB) estimation

- It turns out that we get the test error for free in bagging
- Each tree uses on average 2/3 of the observations; the remaining 1/3 are the out-of-bag (OOB) observations
- For any observation i, we can use the predictions from the trees in which that observations was OOB and then average over the predictions

Random Forests

- Random forests (RF) is bagging with a small tweak that decorrelates trees and thereby reduces variance
- In RF each time a split is made in a tree only a subset of the predictor space is considered for the split where the subset is chosen at random
- The number of predictors to consider is a tuning parameter which we could pick using cross-validation
- In practice, the number of predictors considered is usually the square root of the number of predictors

Boosting

- Like bagging, boosting is a general approach
- Boosting is similar to bagging except that trees are grown sequentially; each tree uses information from the previously grown trees (it is essentially grown on the residuals of the model)
- Each new tree's contribution is discounted by a learning rate
- The learning rate is a tuning parameter, as is the number of trees to grow, and the number of splits in each tree

- Essentially boosting is an ensemble of sparse trees

Business and Local Government Data Research Centre

Thank you

Join in the conversation online:

@BLGDataResearch #Data2Life

LinkedIn: ESRC Business and Local Government Data Research Centre

YouTube: ESRC Business and Local Government Data Research Centre

Email: BLGDataResearch@essex.ac.uk

EXPLORING DATA
ENHANCING KNOWLEDGE
EMPOWERING SOCIETY

E·S·R·C ECONOMIC & SOCIAL RESEARCH COUNCIL

University of Essex