

STK2100- Machine Learning and Statistical Methods for Prediction and Classification

Spring 2025

Introduction

Covers Chapter 1 and Sections 2.1-2.6 and 2.9

Ingrid Hobæk Haff
Department of Mathematics
University of Oslo

Statistics/machine learning/data mining

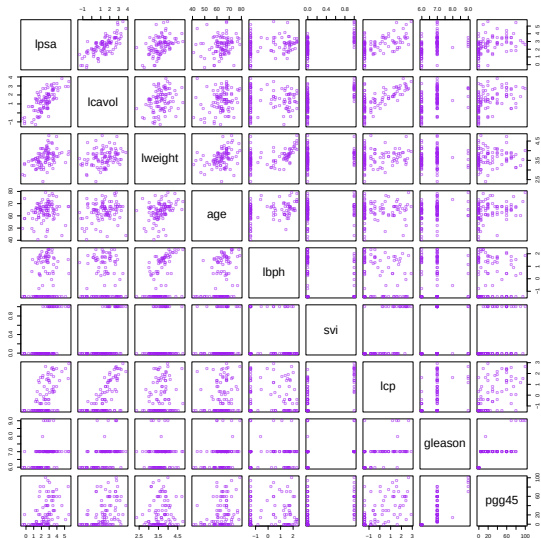
- Hastie et al. (2009):

*The field of Statistics is constantly **challenged by the problems** that science and industry brings to its door. In the early days, these problems often came from agricultural and industrial experiments and were relatively small in scope. With the advent of computers and the information age, **statistical problems have exploded both in size and complexity**. Challenges in the areas of data storage, organization and searching have led to the new field of “data mining”; statistical and computational problems in biology and medicine have created “bioinformatics.” **Vast amounts of data** are being generated in many fields, and the statistician’s job is to **make sense of it all**: to extract important patterns and trends, and understand “what the data says.” We call this **learning from data**.*

- In many cases statistics and machine learning are presented as different things:
 - Statistics is model based
 - Machine learning is algorithmic based
 - <https://www.svds.com/machine-learning-vs-statistics/>
 - Breiman 2001
- In practice, there is considerable overlap and a common goal, namely to **extract knowledge from data**

Example: Prostate cancer data

Problem: predict the (log of the) level of prostate specific antigen (PSA), based on a number of clinical measures



Example: Handwritten digits

Problem: predict, from the 16×16 matrix of pixel intensities, the identity of each image (0, 1, ..., 9)

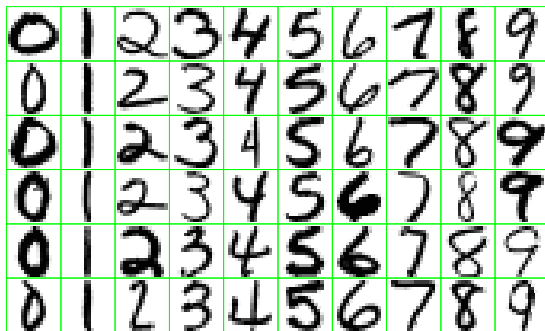
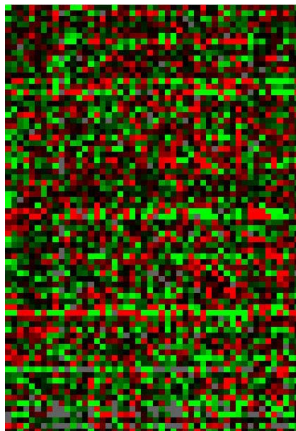


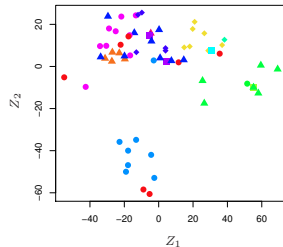
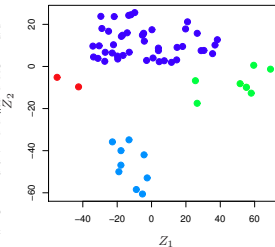
FIGURE 1.2. *Examples of handwritten digits from U.S. postal envelopes.*

Example: DNA microarray data

Problem: understand how the genes and samples are organised, i.e. **group** cells



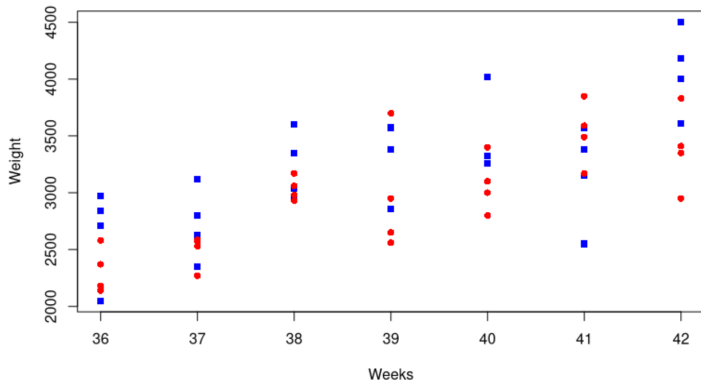
SEW099104
SEW080102
SD17101
QNA
HagmannR
RAGGTPASE
SD07172
ESTs
SEW037402
HummRNA
SEW099084
ESTs
SD471915
MYBP9CTO
ESTs761
SD377651
DNAP-3 TME
SD075912
SEW03489
SD107117
SEW070409
SEW067261
Homo sapiens
SEW0378595
Chr
MITOCHOND
SD47116
ESTsChr 6
SEW099010
SEW080117
SD300147
ESTsChr 2
SD107594
SD300144
PTPRB
SD300143
SEW031041
SEW099083
ESTsChr 3
SD10444
SD0377419
SD097117
SEW0031620
SEW07084
SEW0510534
MALACAL
SEW003464
SD09910
SEW009716
SEW070776
HYPOTHETIC
WAGGAG
SEW032854
SEW0376384
ESTsChr 10
SD099096
ESTsChr 5
SD46036
SEW070815
ESTsChr 2
SEW032866
SD000394
ESTsChr 10
SD084803
SD46148
SD0707905
ESTs
SEW086740
MALACAL
ESTs
SEW060311
SEW067267
SD0379
ESTs



Note that there there is no response in this case

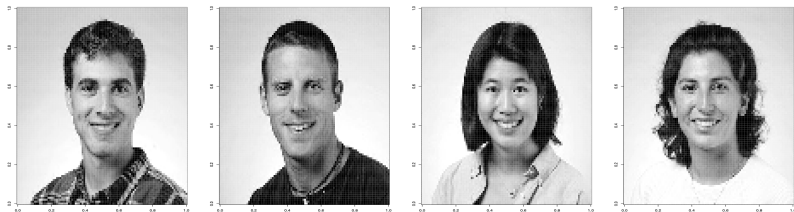
Example: Birthweights of babies

Problem: How does the birthweight of babies depend on sex and duration (boys in blue and girls in red)?



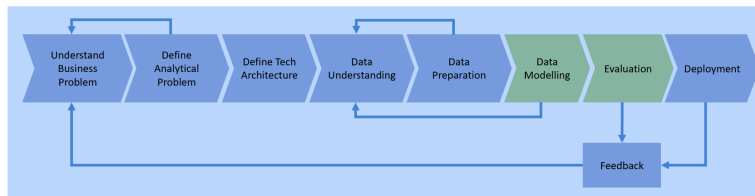
Example: Images of faces

Problem: classify gender based on $200 \times 100 \times 100 = 10\,000$ pixel images



Challenge: many explanatory variables (10 000) compared to the number of individuals (200)

Data Science



The importance of Data Science:

- <https://blog.edx.org/the-importance-of-data-science-in-the-21st-century>
- <http://bigdata.teradata.com/US/Big-Data-Quick-Start/People-And-Roles/Data-Scientist/>
- <https://blog.alexandria.com/know-data-science-important/>
- <https://blog.udacity.com/2014/11/data-science-job-skills.html>

This course

- focuses on **data analysis**
- consists of statistics, machine learning, **statistical learning**

Machine learning and statistical learning

- Machine learning consists of the three components
 - **Supervised learning**: Learning a prediction rule from input and output data
 - **Unsupervised learning**: Identify hidden patterns and relationships of input data
 - **Reinforcement learning**: Construct optimal actions in a stochastic environment
- In statistical learning, the focus is on **prediction** of a response variable based on explanatory variables, and is called
 - **regression** if the response variable is quantitative
 - **classification** if the response variable is categorical
 - data mining/**clustering** if there is no response variable

Machine learning and statistical learning

- Machine learning is concerned with algorithms for prediction
 - Often used as a black box
 - Evaluation often based on a separate test set
- Statistical learning is concerned with methods for prediction
 - More "open box"
 - Emphasis on inference: quantification of uncertainty, evaluation of the method, hypothesis testing, ...
- There is no clear distinction, as both have focus on prediction

This course

- We will concentrate on **prediction** based on **regression/classification** (supervised learning)
- There will also be a bit about **clustering** (unsupervised learning)
- We will to some extent touch upon problems related to **big data**, more specifically
 - statistical issues: p larger than n problems, multiple testing
 - computational issues
- Textbook: Hastie, Tibshirani, Friedman: **The Elements of Statistical Learning**
- Supplementary text: James, Witten, Hastie, Tibshirani: **An Introduction to Statistical learning - with applications in R**
- Supplemented by extra exercises/notes.

Plan for the course

- Chapter 2: Overview of Supervised Learning
- Chapter 3: Linear methods for regression: least squares, subset selection, shrinkage methods, principal components
- Chapter 4: Linear methods for classification: linear regression for an indicator response, linear discriminant analysis, logistic regression
- Chapter 5: Splines
- Chapter 6: Kernel smoothing methods
- Chapter 7: The bias-variance decomposition
- Chapter 9: Additive models
- Chapter 10: Boosting
- Chapter 11: Neural networks
- Chapter 12: Support vector machines
- Chapter 13: Nearest neighbour
- Chapter 14: Cluster analysis
- Chapter 15: Random forests

STK-IN4300: More details, remaining chapters

More about the course

- Required background:
 - Basic programming (IN1900 or IN1000)
 - Calculus/linear algebra (MAT1100/1110/1120)
 - Probability theory, statistical inference (STK1100/1110)
- Focus on **understanding** methods
 - There will be some mathematics
 - We will go through R commands/functions for practical use: [James et al. \(2013\)](#) will be useful here
 - Exercises will be a mix of theoretical and practical tasks
- There will be 2 **mandatory assignments**:

*To qualify for final examination, both mandatory assignments must be approved. You have **one** attempt to pass an assignment. Mandatory assignments are not part of the grading in our courses.*

Prediction and estimation of f

- Assume that $Y = f(\mathbf{x}) + \varepsilon$, $E(\varepsilon) = 0$, where f is unknown, so that

$$E(Y|\mathbf{x}) = f(\mathbf{x})$$

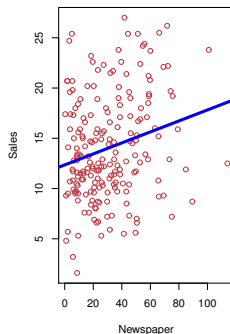
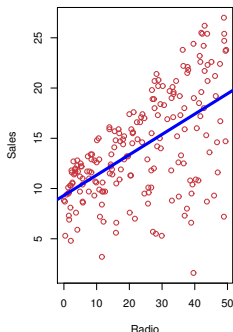
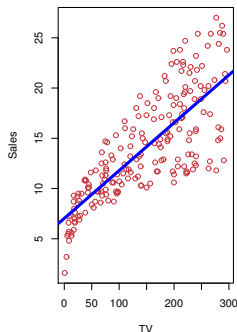
- Why do we want to estimate f ?
- The prediction of Y is given by $\hat{y} = \hat{f}(\mathbf{x})$
- The estimate $\hat{f}$ can give us an understanding of how y is influenced by $x_1, \dots, x_p$:
 - Which variable(s) influence the response?
 - What is the relationship between a variable and the response? Is it linear?
- The **precision** of an estimate (for given $\mathbf{x}$ and $\hat{f}$) can be quantified by

$$E(Y - \hat{Y})^2 = \underbrace{E\left(\left(f(\mathbf{x}) - \hat{f}(\mathbf{x})\right)^2\right)}_{\text{Reducible}} + \underbrace{\text{Var}(\varepsilon)}_{\text{Non-reducible}}$$

- The choice of model and estimation method will influence the **reducible** part of the error in prediction

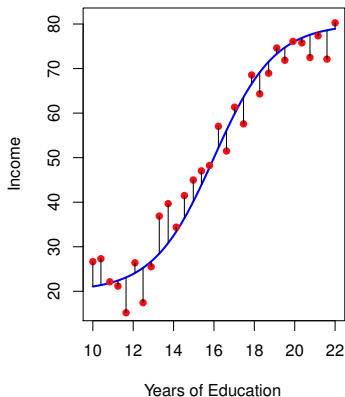
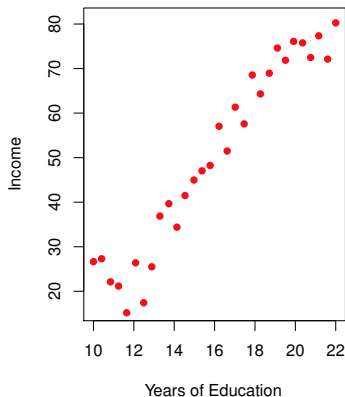
Example: advertising data

- Response: Sale of product in 200 different markets (`sales`)
- Explanatory variables:
 - Advertising budget in tv (`TV`)
 - Advertising budget in radio (`radio`)
 - Advertising budget in newspapers (`newspaper`)



Example: wage data

- Response: Income (`income`)
- Explanatory variables: years of education



Estimation of f - parametric methods

Linear regression: Assume

$$f(\mathbf{x}) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$

- The model is described by the parameters $\beta = (\beta_0, \dots, \beta_p)$
- The estimation of f is done through the estimation of the parameters β

Another example of a parametric model: Assume $p = 1$ and

$$f(x) = \beta_1 e^{\lambda_1 x} + \beta_2 e^{\lambda_2 x}$$

- The model is described by the parameters $\theta = (\beta_1, \beta_2, \lambda_1, \lambda_2)$
- The estimation of f is done through the estimation of the parameters θ

Estimation of f - non-parametric methods

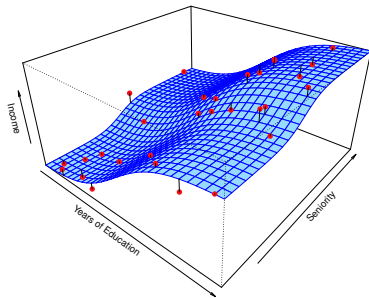
Non-parametric methods: Assume $f(\mathbf{x})$ is a smooth function in $\mathbf{x}$

- Data:
 $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$
- $f(\mathbf{x}) = E(Y|\mathbf{X} = \mathbf{x})$ can be estimated by

$$\hat{f}(\mathbf{x}) = \frac{\sum_{i=1}^N I(\mathbf{x}_i = \mathbf{x}) y_i}{\sum_{i=1}^N I(\mathbf{x}_i = \mathbf{x})}$$

- There can be very few observations with $\mathbf{x}_i = \mathbf{x}$
- Since $f(\mathbf{x}) \approx f(\mathbf{x}_i)$ for $\mathbf{x}_i$ close to $\mathbf{x}$, we can instead use

$$\hat{f}(\mathbf{x}) = \frac{\sum_{i=1}^N I(\mathbf{x}_i \approx \mathbf{x}) y_i}{\sum_{i=1}^N I(\mathbf{x}_i \approx \mathbf{x})}$$



Parametric or non-parametric?

- Recall that the precision of the estimate can be quantified by

$$E(Y - \hat{Y})^2 = E \underbrace{\left(f(\mathbf{x}) - \hat{f}(\mathbf{x})\right)^2}_{\text{reducible}} + \underbrace{\text{Var}(\varepsilon)}_{\text{Non-reducible}}$$

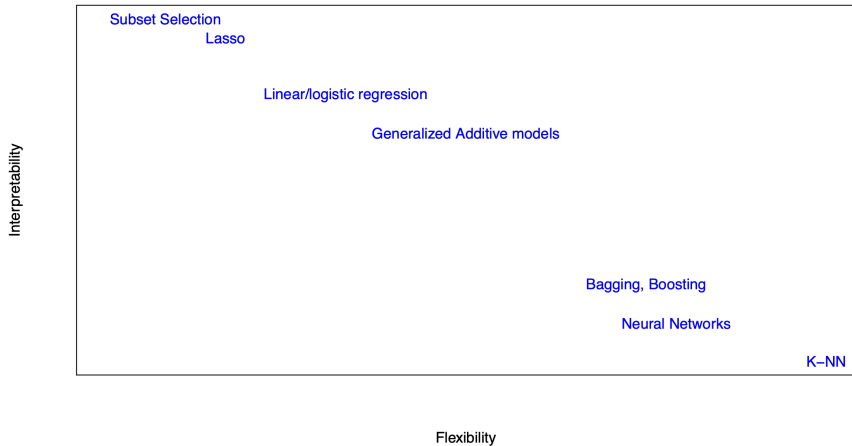
- The reducible part can further be decomposed as

$$\begin{aligned} E \left(f(\mathbf{x}) - \hat{f}(\mathbf{x}) \right)^2 &= E \left(f(\mathbf{x}) - E \left(\hat{f}(\mathbf{x}) \right) + E \left(\hat{f}(\mathbf{x}) \right) - \hat{f}(\mathbf{x}) \right)^2 \\ &= \left(f(\mathbf{x}) - E \left(\hat{f}(\mathbf{x}) \right) \right)^2 + E \left(\left(E \left(\hat{f}(\mathbf{x}) \right) - \hat{f}(\mathbf{x}) \right)^2 \right) + \\ &\quad 2 \left(f(\mathbf{x}) - E \left(\hat{f}(\mathbf{x}) \right) \right) \underbrace{E \left(E \left(\hat{f}(\mathbf{x}) \right) - \hat{f}(\mathbf{x}) \right)}_{=0} \\ &= \underbrace{\left(f(\mathbf{x}) - E \left(\hat{f}(\mathbf{x}) \right) \right)^2}_{\text{Bias}^2} + \underbrace{\text{Var} \left(\hat{f}(\mathbf{x}) \right)}_{\text{Variance}} \end{aligned}$$

Parametric or non-parametric?

- Few assumptions about f gives small bias, but can give high variance
- Harder assumptions on f can give small variance, but high bias (if the assumptions are wrong)
- For interpretation puposes, simpler forms of f are preferable

Overview of different methods



Evaluation of precision

- No statistical learning method will be the best in all situations, it depends on how complex f is, how much data is available etc.
- In a specific problem setting, one should try out several methods
- How does one evaluate which one is the best?
- A possible goal in regression could be

$$\text{Minimise } E \left(\left(Y^* - \hat{f}(\mathbf{x}^*) \right)^2 \right),$$

where $\mathbf{x}^*$ are values of explanatory variables where we want to predict a new Y^* .

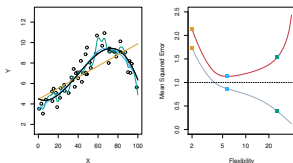
- The problem is that the distribution of $(\mathbf{x}^*, Y^*)$ is unknown

Estimation of prediction error

- A possible estimate of $E \left(\left(Y^* - \hat{f}(\mathbf{x}^*) \right)^2 \right)$ is the **mean squared error**

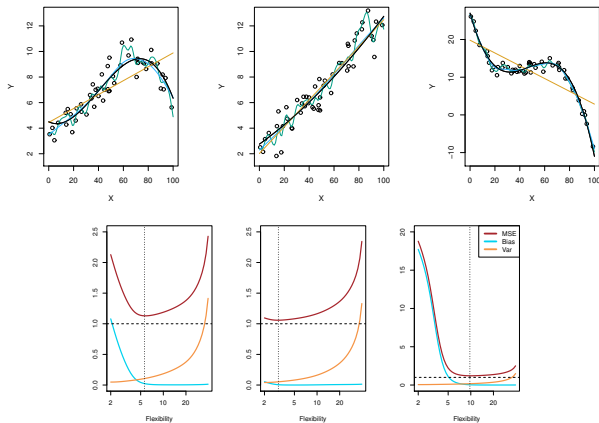
$$MSE = \frac{1}{N} \sum_{i=1}^N \left(Y_i - \hat{f}(\mathbf{x}_i) \right)^2$$

- Then the prediction error is evaluated on the same data as the ones used for estimating f
- However, we are interested in prediction on new data, and use of the same data can give too optimistic answers



- The idea is to divide the data into two parts:
 - **Training data:** Used for estimating f
 - **Test data:** Used for estimating $E \left(\left(Y^* - \hat{f}(\mathbf{x}^*) \right)^2 \right)$

Variance vs bias



$$E \left(\left(f(\mathbf{x}) - \hat{f}(\mathbf{x}) \right)^2 \right) = \underbrace{\left(f(\mathbf{x}) - E \left(\hat{f}(\mathbf{x}) \right) \right)^2}_{\text{Bias}^2} + \underbrace{\text{Var} \left(\hat{f}(\mathbf{x}) \right)}_{\text{Variance}}$$

- Bias **decreases** with higher flexibility
- Variance **increases** with higher flexibility

Classification

- In regression, prediction error can be quantified by $E \left(\left(Y^* - \hat{f}(\mathbf{x}^*) \right)^2 \right)$
- In classification, we can instead use the **error rate** $E \left(I(Y \neq \hat{Y}) \right)$
- This can be estimated by $\frac{1}{N} \sum_{i=1}^N I(\hat{y}_i \neq y_i)$
- However, as before
 - the estimated error rate is too optimistic if estimated from the same data
 - there is a trade-off between bias and variance

Linear models and least squares estimation

- Simple parametric model:

$$\hat{Y} = \hat{f}(\mathbf{X}) = \hat{\beta}_0 + \sum_{j=1}^p X_j \hat{\beta}_j$$

- Available: **Training data** $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$.
- How do we estimate $\beta_0, \dots, \beta_p$?
- A reasonable criterion is

$$\text{Minimise } \text{RSS}(\beta) = \sum_{i=1}^N \left(y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j \right)^2 \text{ w.r.t. } \beta = (\beta_0, \dots, \beta_p)^T$$

- Define

$$\mathbf{X} = \begin{pmatrix} 1 & x_{1,1} & \cdots & x_{1,p} \\ 1 & x_{2,1} & \cdots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N,1} & \cdots & x_{N,p} \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}, \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}$$

Then a vector/matrix formulation is given by

$$\text{RSS}(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta)$$

Least squares estimation

- Minimise $\text{RSS}(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta)$ w.r.t. β
- Differentiation w.r.t. β gives the **normal equations**

$$\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = \mathbf{0}$$

$$\Downarrow$$

$$\mathbf{X}^T\mathbf{X}\beta = \mathbf{X}^T\mathbf{y}$$

- If $\mathbf{X}^T\mathbf{X}$ is nonsingular, a unique solution is given by

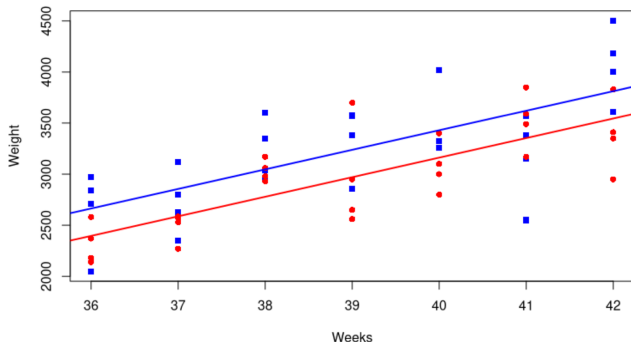
$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

- The **fitted values** are then given by

$$\hat{y}_i = \mathbf{x}_i^T \hat{\beta}$$

Example: Birthweights of babies

- Response: Birthweight in grams
- Explanatory variables: duration of pregnancy and gender



A possible model is a linear regression of the form

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \epsilon_i$$

with

$$x_{i1} = \text{weeks} \quad x_{i2} = I(\text{baby } i \text{ is a girl})$$

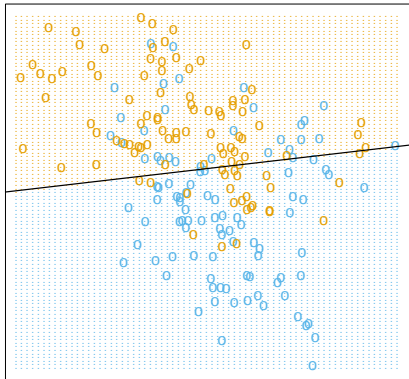
Example: Classification

- **Classification** problem with two classes (Orange and Blue) and two covariates
- We define a **numeric** response: $Y = I(G = \text{Orange})$
- A possible model is a **linear regression** $Y = \mathbf{x}^T \boldsymbol{\beta} + \varepsilon$ to obtain fitted values $\hat{Y} = \mathbf{x}^T \hat{\boldsymbol{\beta}}$
- We then convert to classification according to

$$\hat{G} = \begin{cases} \text{Orange} & \text{if } \hat{Y} > 0.5 \\ \text{Blue} & \text{if } \hat{Y} \leq 0.5 \end{cases}$$

Example: Classification

The decision boundary $\{\mathbf{x} : \mathbf{x}^T \hat{\boldsymbol{\beta}} > 0.5\}$ is then linear



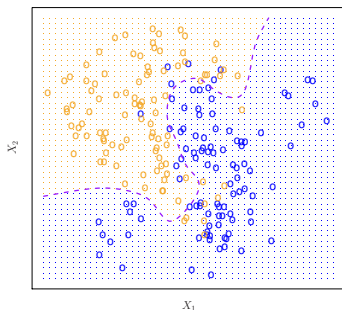
Is the boundary good?

The Bayes classifier

- It can be shown that the classification which **minimises the error rate** is

$$\hat{Y} = \arg \max_j \Pr(Y = j | \mathbf{X} = \mathbf{x})$$

- This is called the **Bayes classifier**



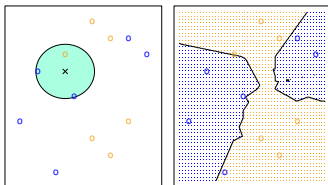
The borders between regions are called **Bayesian decision borders**

K-nearest neighbours

- The ideal is to use the Bayes classifier
- In practice, $\Pr(Y = k|\mathbf{X} = \mathbf{x}^*)$ is unknown and needs to be estimated.
- One option is K -nearest neighbour
- Let $\mathcal{N}_0 \subset \{1, \dots, N\}$ be the K indices with $\mathbf{x}_i$ nearest to $\mathbf{x}^*$.
- The estimate of $\Pr(Y = k|\mathbf{X} = \mathbf{x}^*) = E(I(Y = k|\mathbf{X} = \mathbf{x}^*))$ is then given by

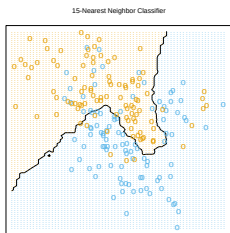
$$\hat{\Pr}(Y = k|\mathbf{X} = \mathbf{x}^*) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} I(y_i = k)$$

- The classification rule is to classify the test point $\mathbf{X} = \mathbf{x}^*$ to the class with the highest number of cases among the K nearest points

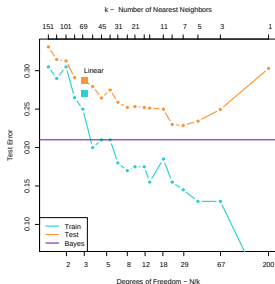
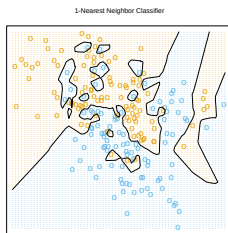


K-nearest neighbour for different K :

$K = 15$



$K = 1$



Misclassification curves for the test (orange) and the training error (teal) for K -nearest neighbour classification as a function of the degrees of freedom N/K . The results for linear regression are the orange and blue squares at three degrees of freedom. The purple line is the optimal Bayes error rate.

Statistical decision theory

- Consider now a general setting where $(\mathbf{X}, Y) \sim p(\mathbf{x}, y)$
- Given $\mathbf{X}$, what is the **optimal** way of predicting Y ?
- This can be formalised through **statistical decision theory**
- Define a **loss function** $L(Y, f(\mathbf{X}))$, that measures the loss of predicting Y by $f(\mathbf{X})$.
- A typical choice is the **squared error loss**
 $L(Y, f(\mathbf{X})) = (Y - f(\mathbf{X}))^2$
- Since $(\mathbf{X}, Y)$ are **stochastic**, we will minimise the **expected prediction error** (EPE)

$$\text{EPE}(f) = E((Y - f(\mathbf{X}))^2) = \int \int (y - f(\mathbf{x}))^2 p(\mathbf{x}, y) d\mathbf{x} dy$$

Statistical decision theory

- Now $p(\mathbf{x}, y) = p(\mathbf{x})p(y|\mathbf{x})$, so that

$$\text{EPE}(f) = \int \left(\int (y - f(\mathbf{x}))^2 p(y|\mathbf{x}) dy \right) p(\mathbf{x}) d\mathbf{x}$$

- Hence, we can minimise the EPE **pointwise**

$$f(\mathbf{x}) = \arg \min_c \int (y - c)^2 p(y|\mathbf{x}) dy = \arg \min_c E_{Y|\mathbf{x}} \left((Y - c)^2 | \mathbf{X} = \mathbf{x} \right)$$

- It may be shown that the solution is

$$f(\mathbf{x}) = E(Y|\mathbf{X} = \mathbf{x})$$

- This may for instance be estimated by K -nearest neighbours

Alternative loss function(s)

- **Absolute loss:** $L(Y, f(\mathbf{X})) = |Y - f(\mathbf{X})|$
- Solution: $f(\mathbf{x}) = \text{median}(Y|\mathbf{X} = \mathbf{x})$
- The absolute loss is less sensitive to large errors, hence more robust
- On the other hand, it is mathematically less convenient
- Other choices also possible

Decision theory and classification

- Assume that we have a categorical output G , with $G \in \{1, \dots, K\}$
- Then we need to specify the loss $L(k, g)$ for all combinations k, g , resulting in a $K \times K$ matrix
- $(\mathbf{X}, G)$ is stochastic, and the EPE is given by

$$\begin{aligned} EPE &= E \left(L \left(G, \hat{G}(\mathbf{X}) \right) \right) \\ &= E_{\mathbf{X}} \left(E_{G|\mathbf{X}} \left(L \left(G, \hat{G}(\mathbf{X}) \right) | \mathbf{X} \right) \right) \\ &= E_{\mathbf{X}} \left(\sum_{k=1}^K L \left(k, \hat{G}(\mathbf{X}) \right) \Pr(G = k | \mathbf{X}) \right) \end{aligned}$$

- This may be solved by pointwise minimisation

$$\hat{G}(\mathbf{x}) = \arg \min_g \sum_{j=1}^K L(k, g) \Pr(G = k | \mathbf{X})$$

Decision theory and classification with the 0 – 1 loss

- The simplest (and most used) choice is the 0 – 1 loss

$$L(k, g) = I(g \neq k) = \begin{cases} 1 & \text{if } g \neq k \\ 0 & \text{if } g = k \end{cases}$$

- This gives

$$\begin{aligned} \hat{G}(\mathbf{x}) &= \arg \min_g \sum_{k=1}^K I(g \neq k) \Pr(G = k | \mathbf{X}) \\ &= \arg \min_g \sum_{k=1}^K (1 - I(g = k)) \Pr(G = k | \mathbf{X}) \\ &= \arg \min_g (1 - \Pr(G = g | \mathbf{X})) \\ &= \arg \max_g \Pr(G = g | \mathbf{X}) \end{aligned}$$

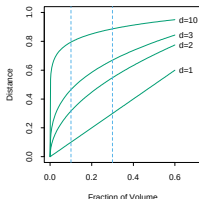
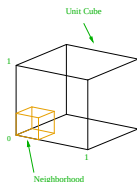
- This is known as the **Bayes classifier**

Local methods in high dimensions

- We need to estimate $E(Y|\mathbf{X} = \mathbf{x})$ or $\Pr(G = g|\mathbf{X} = \mathbf{x})$
- Local (nearest neighbour) methods give

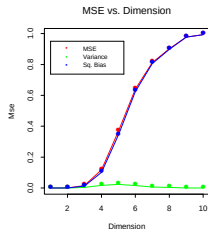
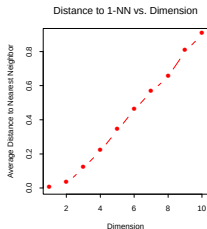
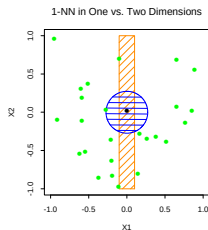
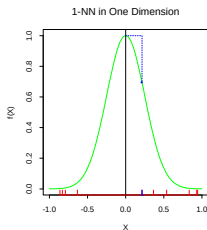
$$\hat{f}(\mathbf{x}) = \frac{\sum_{i=1}^N I(\mathbf{x}_i \approx \mathbf{x}) y_i}{\sum_{i=1}^N I(\mathbf{x}_i \approx \mathbf{x})}$$

- Typically $I(\mathbf{x}_i \approx \mathbf{x})$ is defined by $\|\mathbf{x}_i - \mathbf{x}\|^2 \leq e$, for some number e
- Assume that $\mathbf{x} \in [0, 1]^p$
- We want to specify e , such that a fraction r of data are included
- The problem is that e increases with p



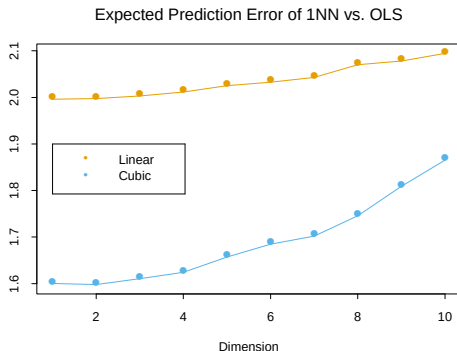
Local methods in high dimensions - Example

- Consider 1000 observations $(y_i, \mathbf{x}_i)$, where $Y = f(\mathbf{x}) = \exp(-8\|\mathbf{x}\|^2)$ and $\mathbf{x} \in [-1, 1]^p$
- We estimate f by 1-NN
- How close is the 1-nearest observation as p increases?



Least squares versus 1-NN

- Consider two examples, $f_1(\mathbf{x}) = x_1$, $f_2(\mathbf{x}) = 0.5(x_1 + 1)^3$
- Plot of the relative prediction error of 1-NN vs least squares (OLS):



- There is gain in using a linear model compared to 1-NN even when this is wrong
- Further, the gain increases with the dimension

Supervised learning

- Assume the model $Y = f(\mathbf{X}) + \varepsilon$ with $E(\varepsilon) = 0$, so that $f(\mathbf{x}) = E(Y|\mathbf{X} = \mathbf{x})$
- Supervised learning consists in learning about f through a **training set** $\{(\mathbf{x}_i, y_i), i = 1, \dots, N\}$
- NN methods give direct estimates of $f(\mathbf{x})$, however
 - they fail if the dimension of the input space is high
 - if a special structure is known to exist, this can be used to reduce both the bias and the variance
- The typical criterion is to minimise the **residual sum of squares** $RSS(f) = \sum_{i=1}^N (y_i - f(\mathbf{x}_i))^2$
- This results in the **least squares** estimate of f

Supervised learning

- This can be seen as a **function approximation** problem
- Typical restrictions:
 - $f(\mathbf{x})$ is **linear**
 - $f(\mathbf{x}) = \beta_0 + \sum_{j=1}^p f_j(x_j)$, **additive model**
 - $f(\mathbf{x}) = \beta_0 + \sum_{k=1}^K h_k(\mathbf{x})\theta_k$, **linear basis expansions**
 - $f(\mathbf{x})$ is specified through a **neural network**
 - $f(\mathbf{x})$ is **smooth**
- Such restrictions encourage geometrical concepts of Euclidean spaces and mathematical concepts of probabilistic inference to be applied to the problem
- When $f(\mathbf{x})$ is linear, the least squares estimate is a **projection** down to the column space spanned by $\mathbf{X}$

Least squares and maximum likelihood

- Assume now that $\varepsilon \sim N(0, \sigma^2)$ and the ε s are independent.
- Then $Y \sim N(f(\mathbf{x}), \sigma^2)$
- Further, assume that $f(\mathbf{x}) = f(\mathbf{x}; \theta)$ with f known and θ unknown, so that

$$\begin{aligned} p(\mathbf{y}|\mathbf{x}) &= \prod_{i=1}^N p(y_i|\mathbf{x}_i) = \prod_{i=1}^N \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(y_i - f(\mathbf{x}_i))^2\right) \\ &= \frac{1}{(2\pi)^{N/2}\sigma^N} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^N (y_i - f(\mathbf{x}_i))^2\right) \end{aligned}$$

- Maximising $p(\mathbf{y}|\mathbf{x})$ w.r.t. f is **equivalent** to minimising $\text{RSS}(f)$
- In addition, we can obtain

$$\hat{\sigma}_{ML}^2 = \frac{1}{N} \sum_{i=1}^N \left(y_i - \hat{f}(\mathbf{x}_i)\right)^2$$

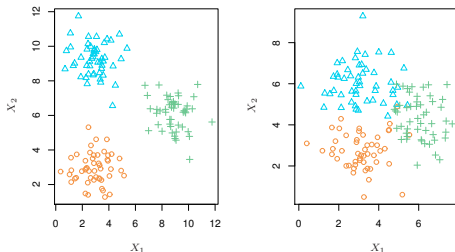
- The maximum likelihood approach is **more general**, so we can also consider other distributions

Regression vs classification

- Variables are often divided into two characteristics:
 - **Quantitative**: Numerical values (ordered)
 - **Qualitative**: Categorical, no ordering
- In **regression**, the response is quantitative
- In **classification**, the response is qualitative
 - There can be qualitative **explanatory variables** within regression
 - There can be quantitative **explanatory variables** within classification

Unsupervised learning

- So far we have considered a response Y and explanatory variable(s) $\mathbf{X}$
- This corresponds to supervised learning
- In many situations, we only have the explanatory variables $\mathbf{X}$, example: the gene expression data
- The aim is then to find relations between variables
- Common approach : **Cluster/group** data



- This is **unsupervised learning**

References

- T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer series in statistics, Springer New York, 2 edition, 2009.
- G. James, D. Witten, T. Hastie, and R. Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.