High-Dimensional Statistics and Applications in Insurance

November 2019
Masaryk University, Brno
Ivana Milović, MAS PhD

Introducing myself

Ivana Milović, MAS PhD

Non-Life Pricing Actuary - Group P&C Pricing

Lecturer - University of Vienna

Prior experience

- Prae and Post-Doc Researcher - Department of Statistics, University of Vienna

Education

- PhD in Statistics (Univ. of Vienna, 2016)
- Master of Advanced Studies in Mathematics (Univ. of Cambridge, 2011)
- BSc in Mathematics and Computer Science (Univ. of Belgrade, 2010)

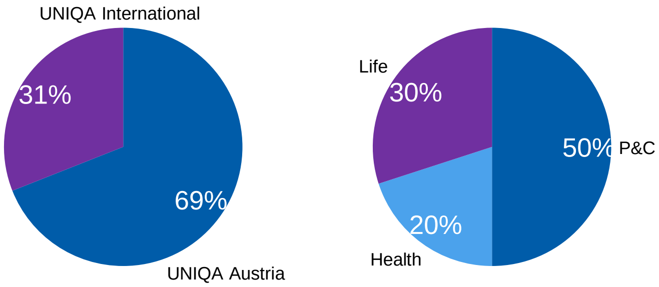
Introduction to Uniqa

UNIQA at a glance

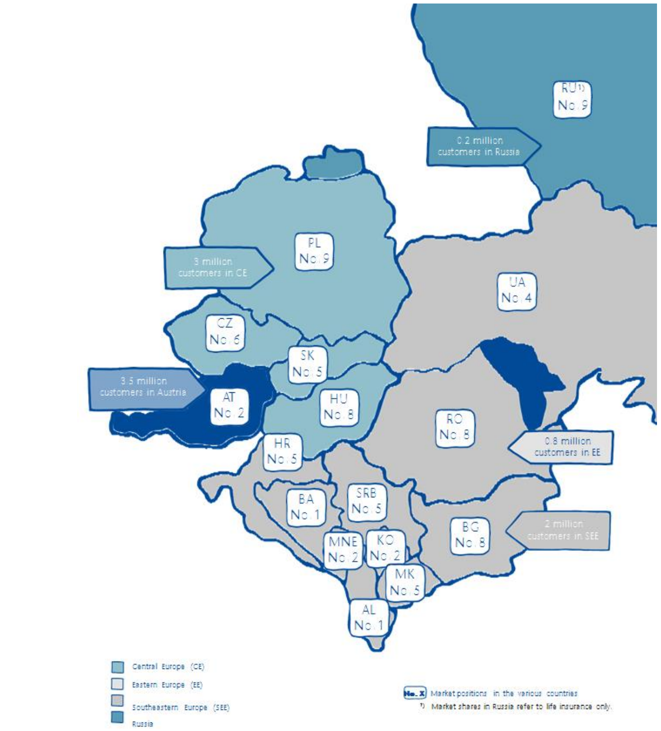
Key financials EURm

	2013	2014	2015	2016[c]	2017
Gross written premiums[a]	5,886	6,064	6,325	5,048	5,293
Premiums earned (retained)[a]	5,641	5,839	6,102	4,443	6,628
Earnings before taxes	308	378	423	225	242
Consolidated net profit	285	290	331	148	161
Combined ratio (net) (P&C)	99.8%	99.6%	97.8%	98.1%	97.5%
Return on Equity	11.9%	9.9%	10.9%	4.7%	5.1%

Diversification by regions and products (GWP[a][b] FY17)



UNIQA's geographical footprint


RU1)
No. 9
0.2 million customers in Russia
PL
No. 9
3 million customers in CE
UA
No. 4
CZ
No. 6
SK
No. 5
3.5 million customers in Austria
AT
No. 2
HU
No. 8
RO
No. 8
0.8 million customers in EE
HR
No. 5
BA
No. 1
SRB
No. 5
BG
No. 8
2 million customers in SEE
MNE
No. 2
KO
No. 2
MK
No. 5
AL
No. 1
Central Europe (CE)
Eastern Europe (EE)
Southeastern Europe (SEE)
Russia
No. X Market positions in the various countries
1) Market shares in Russia refer to life insurance only.

(a) Including savings portion of premiums from unit- and index-linked life insurance,
(b) Excluding consolidation and UNIQA Reinsurance, (c) UNIQA signed contract to sell Italian operations on Dec 2, therefore FY16 IFRS figures excluding Italy

Think UNIQA 4WARD

What are Shared Services?

> “ A central service unit is an **entity within a multi-unit organization** responsible for **supplying** the business units, respective divisions and departments with **specific operational tasks & processes** (eg accounting, payroll, IT, compliance or as in UNIQA's case actuarial and risk)“

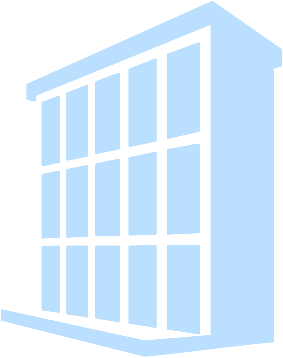
UNIQA operating countries and HQ

Local UNIQA Business Units (BUs) and UNIQA Group are **customers** of U4W and will **outsource** specific **processes to U4W**

U4W **performs the process** for the customers according to commonly defined Service Levels

Benefits through UNIQA 4WARD

- Standardization
- Specialization
- Speed



UNIQA 4WARD (U4W)

Benefits with UNIQA 4WARD

Development of personal and professional skills with UNIQA

Your Development

What can UNIQA 4WARD offer you?

Actuarial education and continuous professional & soft-skill training

1 General onboarding training with focus on UNIQA tools and standards as well as intercultural awareness

2 Function specific training in relevant Group department in Vienna – partially spending time in Vienna and in Bratislava with a strong "applied learning" (learning by doing approach)

3 Mentoring program and on-the-job knowledge transfer

4 International working culture and positive working atmosphere

5 Start-up environment with the stability of an international insurance company in the background.

Various employee benefits

- Bonus payments
- Language courses
- 25 vacation days
- Flexible working times & Home Office

Mathematics Challenge

https://www.uniqa4ward.com/en/challenge.html#Challenge

UNIQA 4WARD
MATH CHALLENGE

Introduction

Introduction

- Topics
  - Model assessment and selection
  - Cross validation, AIC, BIC
  - Linear Models
  - PCR, Regularization methods
  - Generalized Linear models
  - Pricing process
  - Machine Learning in Insurance

Introduction

pringer Texts in Statistics
Gareth James
Daniela Witten
Trevor Hastie
Robert Tibshirani
An Introduction to Statistical Learning
with Applications in R
Sprin
Hastie · Tibshirani
An Introduction to Statistical Learning
THE STRANGE ORDER of THINGS

Introduction

- Let Y be a quantitative response and $X = (X_1, \dots, X_p)$ be a set of regressors and suppose: $\boxed{Y = f(X) + \epsilon,}$ for some fixed (but unknown) function f .
- ϵ has mean 0 and is independent of X. Often we assume normality.
- Note: X can be fixed or random
- Example: Y is the number of claims and X are the characteristics of a driver and his car

Introduction

- **Statistical learning** is a set of approaches for estimating f by $\hat{f}$ from the data.
- Estimation goals can be:
  - Prediction
  - Inference

Introduction

- **Prediction**: $\hat{Y} = \hat{f}(X)$, for some estimate $\hat{f}$.
- If prediction is our only goal and we do not have interest in the form of f, then many modern techniques give good results: random forests, gradient boosting trees, etc.
- Example: predicting prices on the stock exchange. Here the interpretation is not important, as long as the results are good.

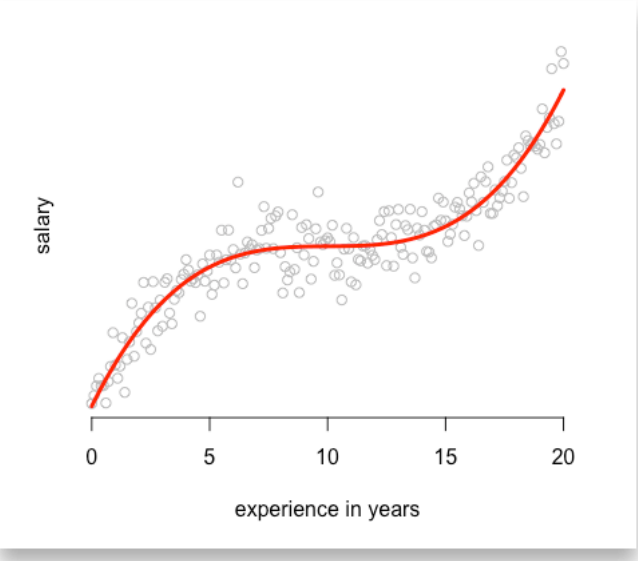
Introduction

- The accuracy of $\hat{Y}$ depends on two quantities:
  - reducible error – coming from approximating f by $\hat{f}$
  - irreducible error – the error coming from ϵ
- We measure the accuracy by the **expected prediction error**
- $$E(Y-\hat{Y})^2 = \underbrace{E(f(X)-\hat{f}(X))^2}_{\text{reducible}} + \underbrace{Var(\epsilon)}_{\text{irreducible}}$$

- Goal: to find a method that has small reducible error

Introduction

- **Inference**: we want to also understand the form of f, i.e. the relationship between Y and $X = (X_1, \dots, X_p)$.
- Is f linear or more complex?
- Which regressors are associated with Y?
- What is their relationship?



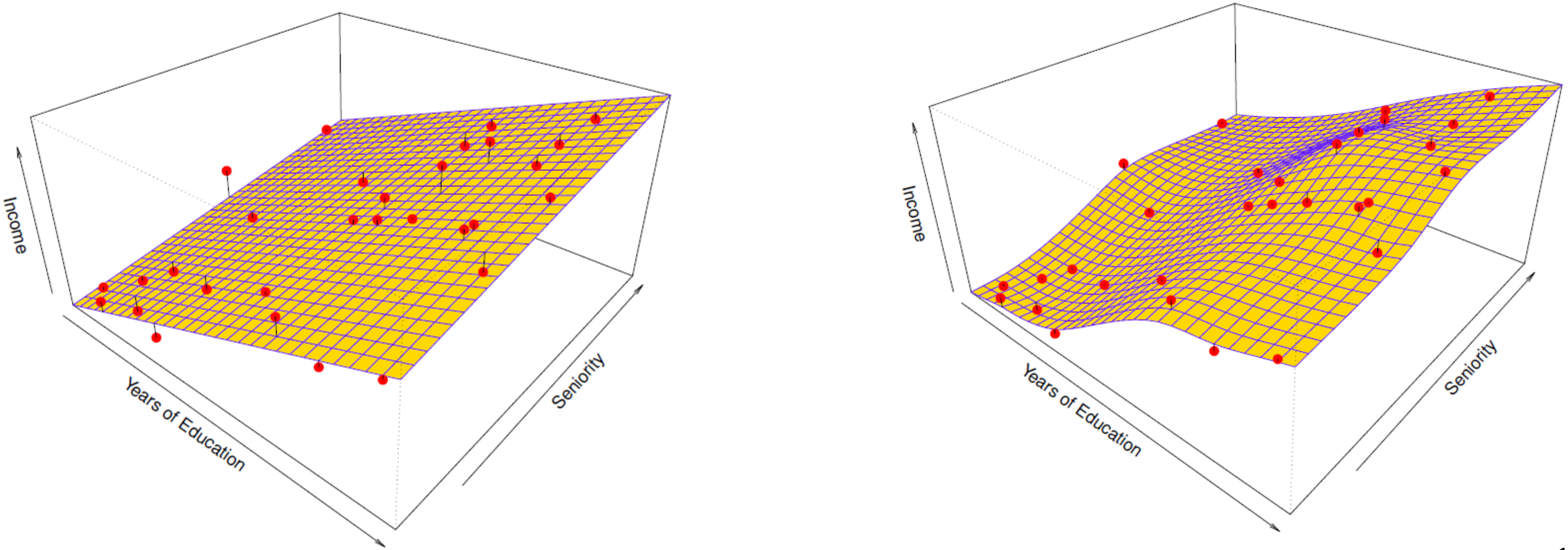
Choice of Model

Choice of Model

- We may choose our model based on what we are more interested in: prediction or inference
- Example:
  - **Parametric** models like linear models and GLMs: simple and interpretable, but not always very accurate
  - **Non-parametric** models like splines, GBM, random forests: better predictions but much less interpretable
- Factors like sample size, computational power, etc. also play a significant role in making a decision.

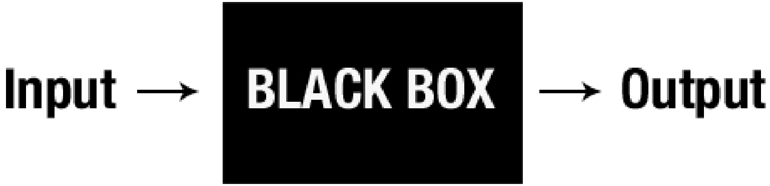
Choice of Model

Example: Linear regression vs. splines



Machine learning controversy

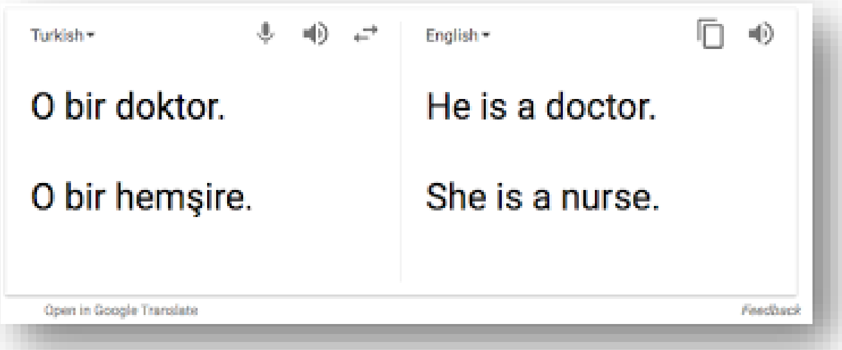
- Many machine learning techniques offer fully automatized routines for calculating prices, insurance premiums, etc. or clustering data into different segments (for example: brands or regions)
- But if the interpretability is missing, many problems might occur



Input → BLACK BOX → Output

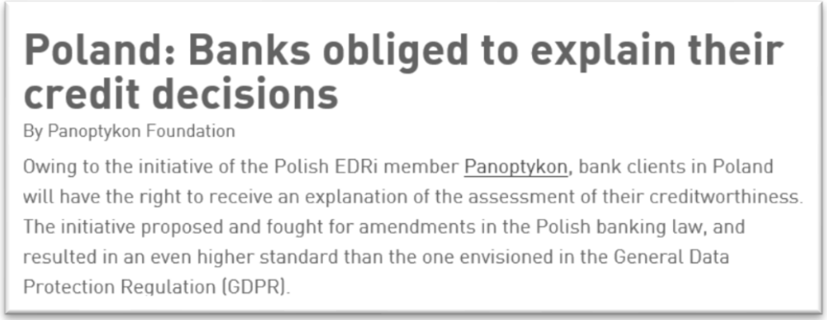
Machine learning controversy

- Certain companies have sparked controversy as ethnic, gender or ‘unethical’ variables slipped into their models, often because data bias was not corrected



O bir doktor. | He is a doctor.

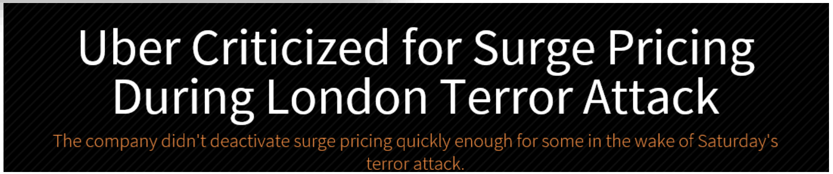
O bir hemşire. | She is a nurse.



Poland: Banks obliged to explain their credit decisions

By Panoptykon Foundation

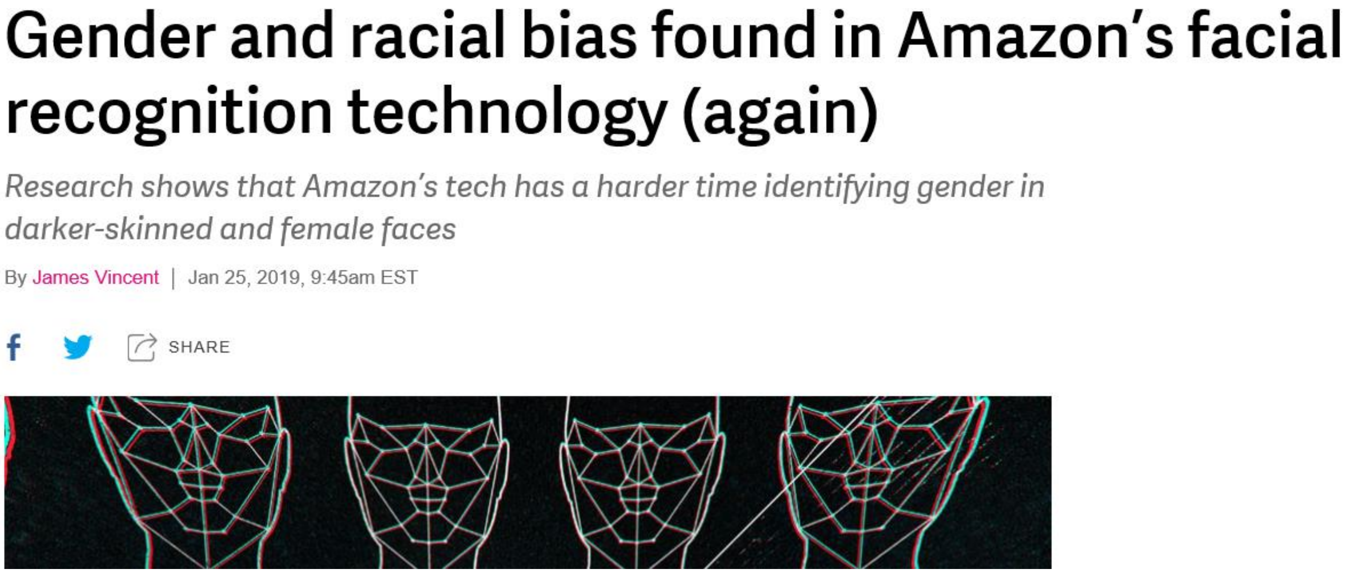
Owing to the initiative of the Polish EDRi member Panoptykon, bank clients in Poland will have the right to receive an explanation of the assessment of their creditworthiness. The initiative proposed and fought for amendments in the Polish banking law, and resulted in an even higher standard than the one envisioned in the General Data Protection Regulation (GDPR).



Uber Criticized for Surge Pricing During London Terror Attack

The company didn't deactivate surge pricing quickly enough for some in the wake of Saturday's terror attack.

Machine learning controversy



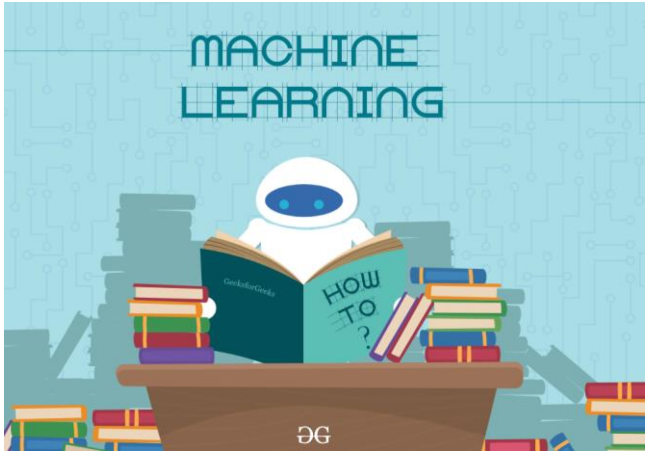
Gender and racial bias found in Amazon's facial recognition technology (again)

Research shows that Amazon's tech has a harder time identifying gender in darker-skinned and female faces

By James Vincent | Jan 25, 2019, 9:45am EST

Machine learning controversy

- What about the insurance industry?
- Current standard: GLM models
- Can Machine learning replace them?
- Later on that!



Assessing Model Accuracy

- No model dominates all other models over all possible data sets. We need to decide which model is most suitable based on the data set given
- The prediction error $E(Y - \hat{f}(X))^2$ can be estimated by the mean-squared error (**MSE**)

$$\boldsymbol{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{f}(X_i))^2}$$

given a sample $(X_i, Y_i)_{i=1}^{n}$.

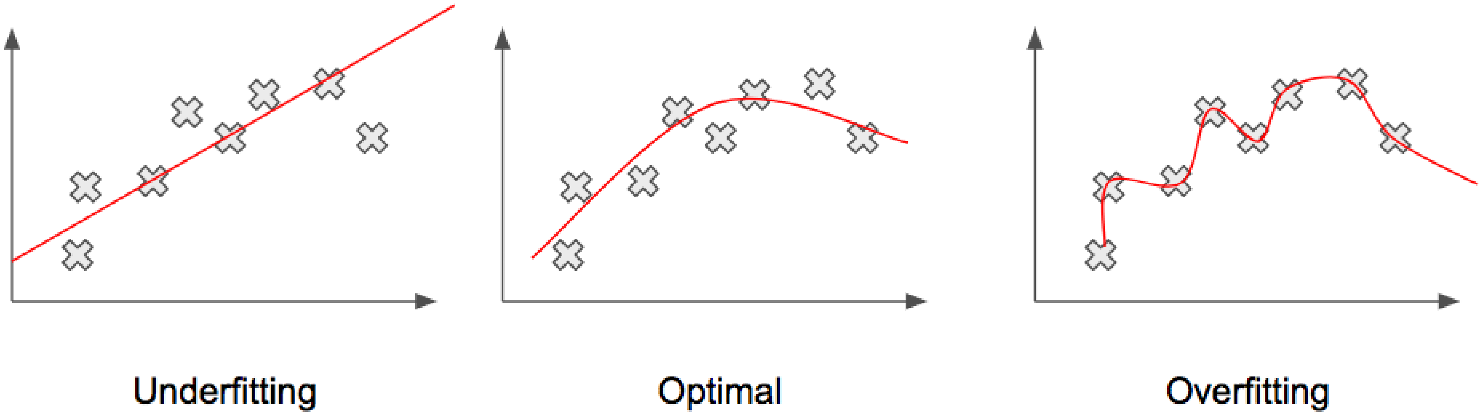
- Here X_i denotes a p –vector of regressors for the i-th data point

Assessing Model Accuracy

- But we do not want to predict the model accuracy on the data we already observed. $\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{f}(X_i))^2$ is actually <u>in-sample (training) MSE</u>.
- We want our model to perform well on the **future data**,
- For a new (unseen) observation (X_0, Y_0), it should hold that $\hat{f}(X_0) \approx Y_0$.
- In general, when considering all new data points: $\underbrace{Average}_{(X_0,Y_0)}(Y_0 - \hat{f}(X_0))^2$ should be small. This is <u>out-of-sample (testing) MSE</u>

Assessing Model Accuracy

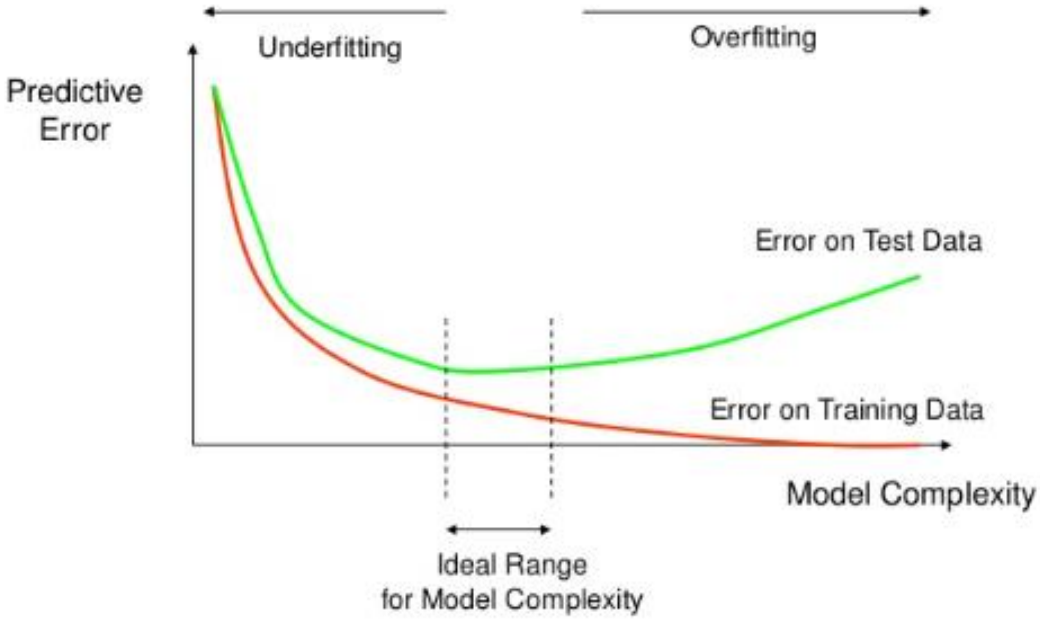
- There is no guarantee that a model with a small training MSE will also have a small testing MSE. This leads to concepts of underfitting and overfitting.



Assessing Model Accuracy

- As the model complexity increases, the training error gets smaller but the testing error increases.
- **Underfitting**: the model is too simple and performs badly on the training data, and consequently on the testing data
- **Overfitting**: the training data is modelled too well, because non-existing patterns in the data are found (coming from the noise). Therefore the performance on the future data is poor.

Assessing Model Accuracy



Bias-variance trade-off

- Let X_0 be fixed. Note that the **test MSE** can be written as

$$E(Y_0 - \hat{f}(X_0))^2 = \underbrace{\left(Bias\left(\widehat{f}\,(X_0)\right)\right)^2 + Var\left(\hat{f}(X_0)\right)}_{\text{reducible}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{irreducible}}.$$

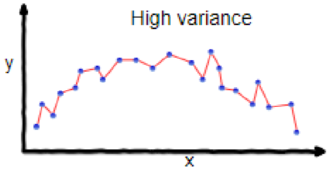
- Bias: Error introduced by approximating f by $\hat{f}$
- Variance: how much $\hat{f}$ changes if we use different data sets for training

Bias-variance trade-off

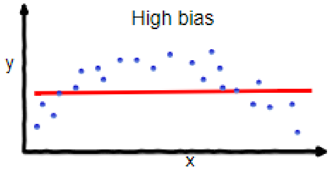
- Easy to find a method with low bias and high variance, just use a curve that connects all the points
- Easy to find a method with low variance and high bias, just take a flat line through the data
- But, we want a method that simultaneously has low bias and low variance.

Bias-variance trade-off

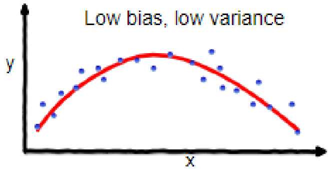
- Example:



overfitting



underfitting



Good balance

Test MSE Estimation

- But in real-life situations it is not possible to compute the test MSE, because f is unknown, so we need to estimate it.
- This could be done in the following ways:
  - Cross-Validation: directly estimating test MSE by using resampling
  - Indirect way of estimating test error: adjust the training error by a penalty term which takes the model dimension into account

Cross Validation

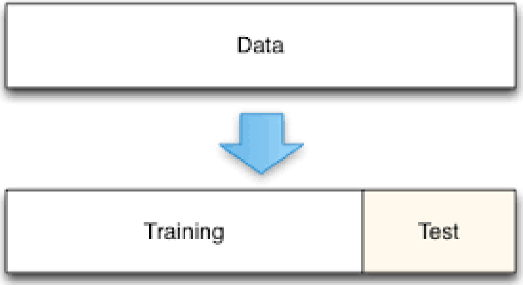
Cross-Validation

- Used to estimate the test MSE, for a given statistical model
- It tells us how our model performs on an unseen data set
- When comparing several competing models, the one with the **smallest** cross-validation error (CV) is preferred.
- It can also be used for selecting tuning parameters for a chosen model (Ridge, Lasso, etc.)

Cross-Validation

There are 3 ways in which CV can be done:

1. **Validation set approach**: divide the data **randomly** into two data sets: training and testing. Usually a 80-20% split is done. The model is then fitted using the training set and the prediction error $\boldsymbol{\frac{1}{m}\sum_{i=1}^{m}\left(Y_i - \widehat{Y_i}\right)^2}$ is calculated on the testing data



Cross-Validation

Example:

- The model trained on 80% of the data gives the following prediction: $\hat{Y} = 2X$.
- The test data is:

Y	X
5	2
9	5
10	4

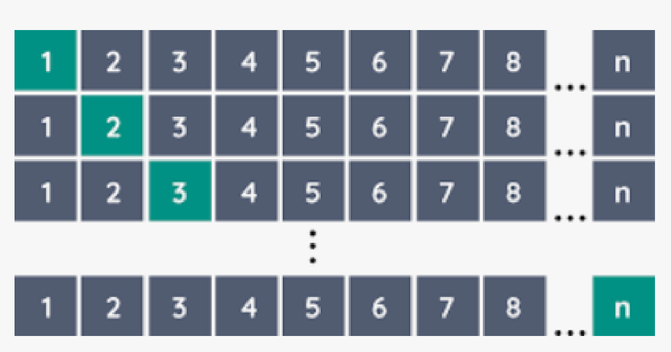
- CV equals: $\frac{1}{3}[(5-4)^2+(9-10)^2+(10-8)^2] = \frac{6}{3} = 2$

Cross-Validation

- Drawbacks:
  - CV error can be extremely variable, depending on how the data was split
  - Only a subset of the data was used for training, this introduces a lot of bias so we might overestimate the testing error

2. **Leave-one-out cross-validation (LOOCV):** Dataset with n sample points is split into $n-1$ data points, on which model training is done and the testing is done on the remaining one data point. This is then repeated n times, so that each point gets to be in the training and the validation data set. The prediction errors are then averaged out.

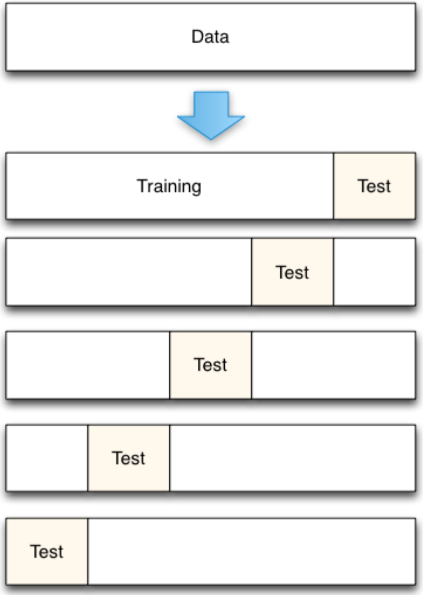
Cross-Validation



- Now there is no randomness in data splits, and there is much less bias compared to the previous method, because $n-1$ points are used for training
- Problem: we have to fit the model n times. Computationally extensive.

Cross-Validation

3. **K-fold cross-validation:** Randomly divide the data set into k parts of (approximately) equal size. Then train the model on $k-1$ parts and test on the remaining part. Repeat k times and average out the testing error.



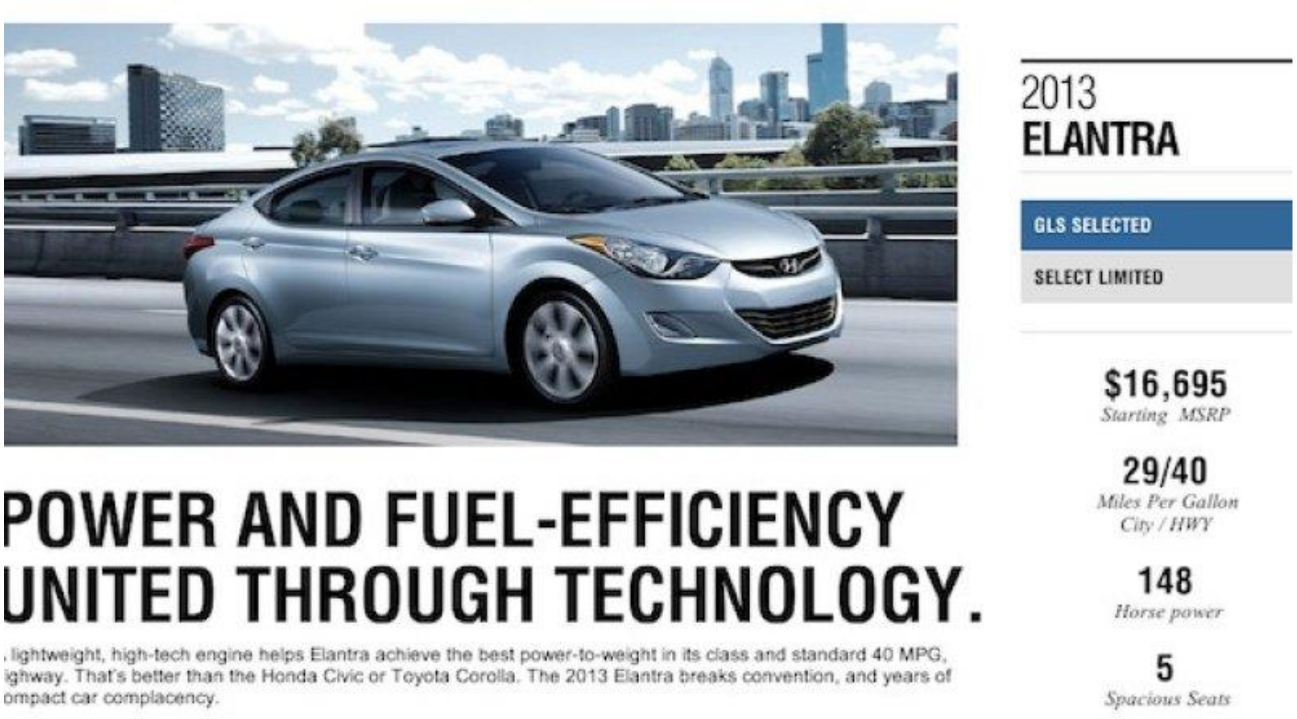
Cross-Validation

- How big should k be? Experience shows that $\boldsymbol{k=5}$ **or** $\boldsymbol{k=10}$ show best results.
- We fit the model only k times
- Bias remains small, because we fit on almost all data and variability of the CV estimate gets smaller compared to LOOCV, because the outputs for each fit are less correlated
- This method corrects the disadvantages over the previous two.

Example

- Response variable **mpg – miles per gallon**
- Polynomial regression is performed with the regressor **horsepower**. But which degree to take?
- Cross-validation can give us an answer

Example



2013
ELANTRA

GLS SELECTED

SELECT LIMITED

$16,695
Starting MSRP

29/40
Miles Per Gallon City / HWY

148
Horse power

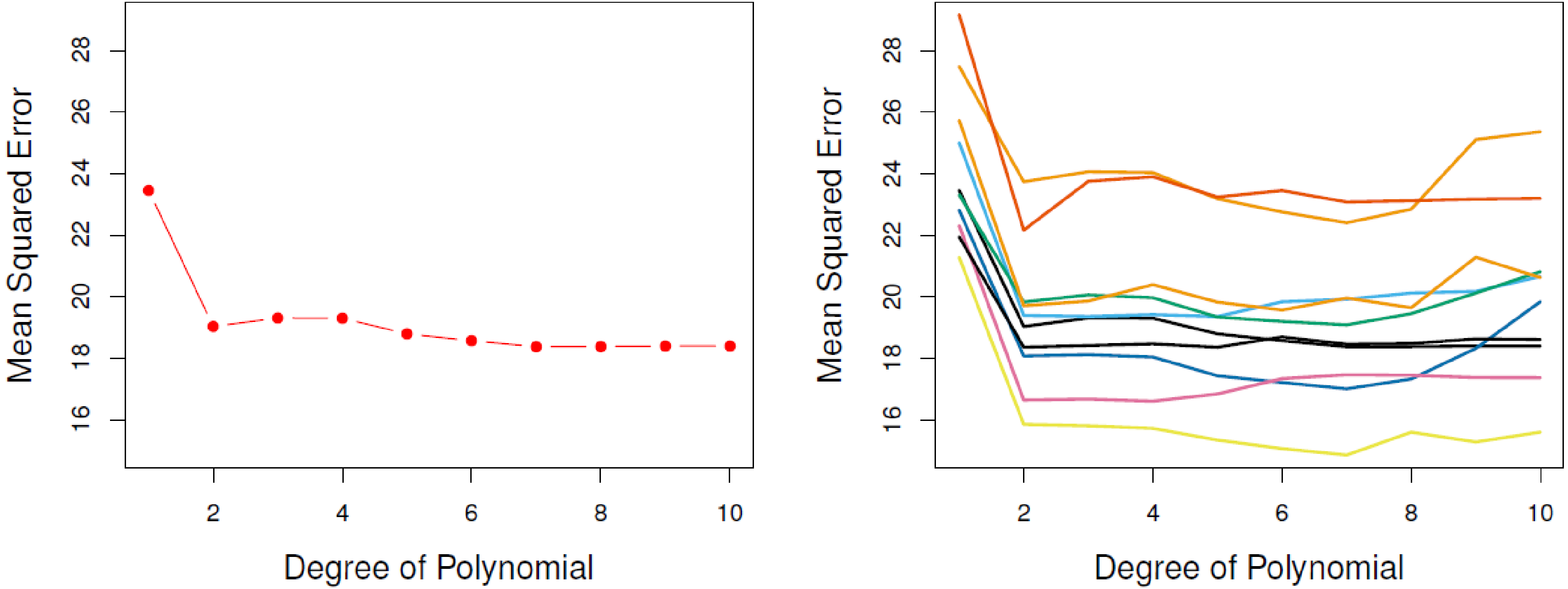
5
Spacious Seats

POWER AND FUEL-EFFICIENCY UNITED THROUGH TECHNOLOGY.

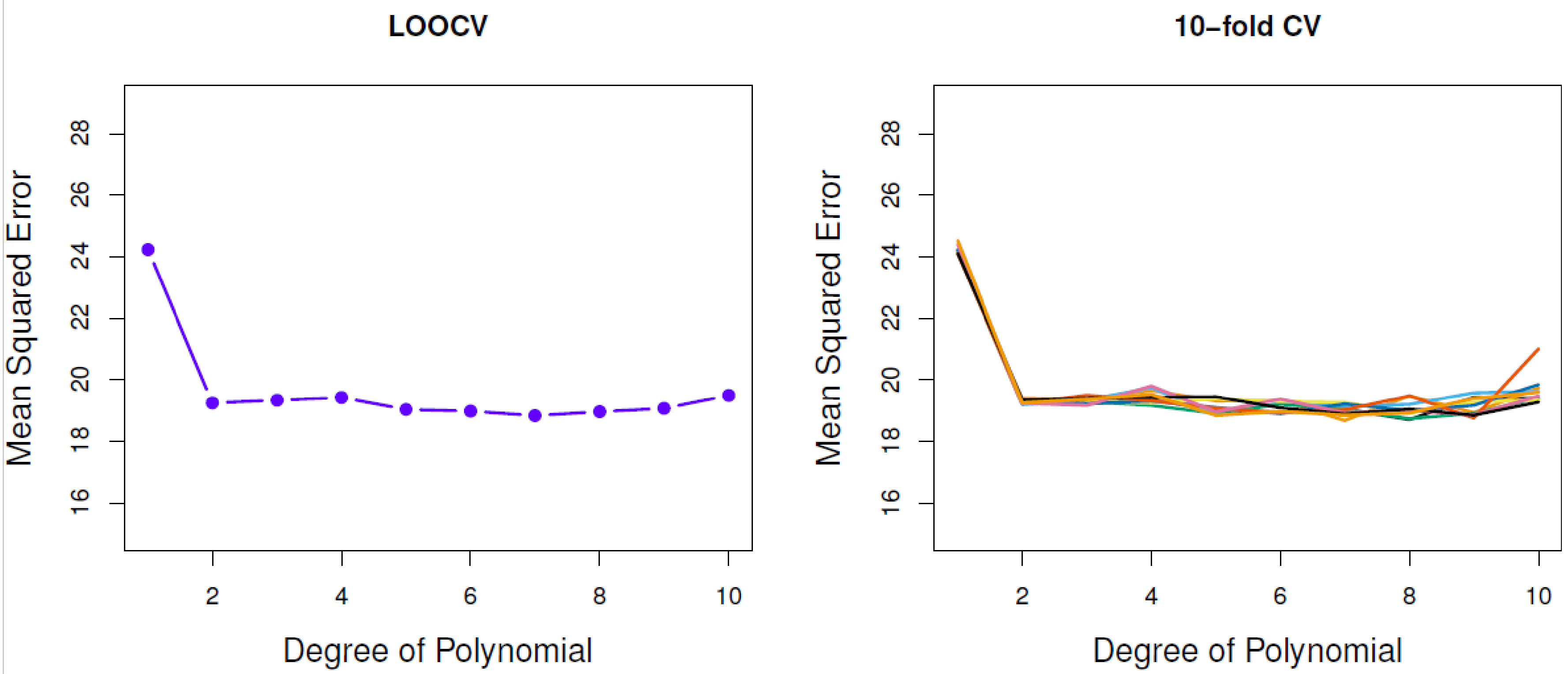
. lightweight, high-tech engine helps Elantra achieve the best power-to-weight in its class and standard 40 MPG, ighway. That's better than the Honda Civic or Toyota Corolla. The 2013 Elantra breaks convention, and years of ompact car complacency.

Example

Validation set approach



Example


LOOCV
10–fold CV
Mean Squared Error
16
18
20
22
24
26
28
2
4
6
8
10
Degree of Polynomial

AIC, BIC, etc.

AIC, BIC, etc.

Other way of estimating the test MSE error is by **adjusting the training MSE.**

- **AIC** (Akaike Information Criterion) is an estimator for out-of-sample prediction error and thereby for the relative quality of a statistical model for a given set of data.
- Given a collection of models, AIC estimates the quality of each model. Thus, AIC provides a means for model selection.

AIC, BIC, etc

- Akaike extends the concept of the maximum likelihood estimation to the case where the number of parameters p is also unknown. A penalty is introduced, depending on p. **So a parameter is added to the model, only if it leads to a significant improvement in the fit.**

AIC, BIC, etc

- Let $f(y|\theta)$ be a candidate model for estimating Y, for $\theta \in R^p$. For example: $f(y|\theta)$ is the density of $N(X\theta, I)$
- Let $\hat{\theta} = \hat{\theta}(Y)$ be the MLE estimator, given the data $Y \in R^n$.
- Then, $\boldsymbol{AIC = -2logf\left(Y\middle|\widehat{\theta}\right) + 2\mathrm{p}}$ is the estimate of the test MSE
- Model with the smallest AIC is chosen

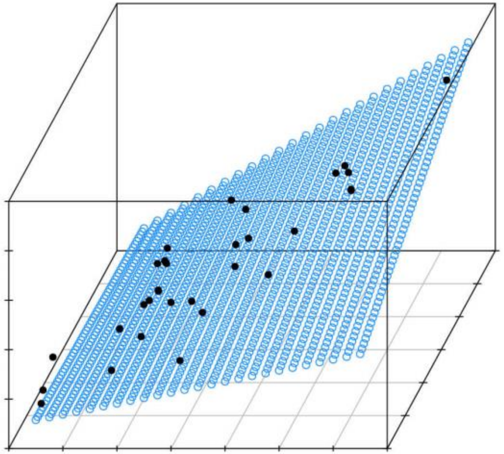
AIC, BIC, etc.

- BIC (Bayesian Information Criterion) is a similar method to AIC.
  - The model with the smallest $\boldsymbol{BIC = -2logf\left(Y\middle|\hat{\theta}\right)} + \mathbf{p\ log(n)}$ is chosen.
  - Since the penalty term here is larger, sparser models are selected than with AIC.

- In the linear regression model with normal errors: AIC and BIC have the following forms:

$$\boldsymbol{AIC = n\,\mathrm{log}(\,MSE) + 2p}\ \text{ and } \boldsymbol{BIC = n\,log(MSE) + plog(n)}$$

Linear Models



Model selection and regularization

- **Linear models** (and generalized linear models: GLMs), though simple, turn out to be surprisingly competitive in real-world problems, compare to more complex models
- Reason for that lies in their simplicity and interpretability
- GLMs are the standard in the insurance business and most of the results for linear models can be naturally generalized
- But what is their prediction accuracy and what happens when the number of parameters $\boldsymbol{p}$ **is large compared to the sample size** $\boldsymbol{n}$?

Model selection and regularization

- Let us focus on linear models, for demonstration
- Assume that: $\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{\epsilon}$, for some $\beta \in R^p$
- $E(\epsilon) = 0$ and $Var(\epsilon) = \sigma I$.
- Also, $\mathrm{Y} \in R^n$ and $\mathrm{X} \in R^{n \times p}$.

Model selection and regularization

- OLS estimator $\hat{\beta} = (X'X)^{-1}X'Y$ is well-defined for $n \geq p$ and it is unbiased. Therefore, the estimates $\hat{Y} = X\hat{\beta}$ are unbiased.
- For $p > n,$ OLS is not even defined. Therefore, we have to come up with some other estimators.

Model selection and regularization

But what about the **variance** of these estimates?

- If $n \gg p,$ the variance is usually small and our estimates are accurate
- But if two or more variables are **highly correlated**, this could lead to high variance and therefore unstable estimates. This happens, because $\det(X'X)$ is almost 0 and the matrix inversion becomes very unstable

Model selection and regularization

- Example of (potentially) **highly-correlated variables** in Motor Insurance
  - entry user age and current user age
  - vehicle age and contract age
  - population density and regional segmentation variables

Model selection and regularization

- Also if n is not much larger than p, the estimates can get very unstable.
- Example: if all regressors are i.i.d. N(0,1) the variance of the predictions equals $\sigma \frac{p}{n-p-1}$.
- This is problematic for p large compared to n.

Model selection and regularization

- Alternatives to OLS in linear regression:
  - Subset selection (best subset and stepwise)
  - Dimension reduction (PCA, for example)
  - Shrinkage methods (Ridge, Lasso, etc.)

Subset Selection

Best subset selection: for a linear model with p predictors do

- Let M_0 be the null model with zero regressors, i.e. sample mean of Y is used as a predictor
- For $k = 1,2, \ldots, p$
  1. Fit all $\binom{p}{k}$ models that contain exactly k predictors
  2. Pick the best among these $\binom{p}{k}$ models and call it M_k. I.e., choose the model with the largest R^2.
- Select the best model from $M_0, M_1, \ldots, M_p$ using cross-validation, AIC, BIC, etc.
- Note: here you cannot use R^2 because then the largest model would always be chosen.

Subset Selection

- This method is conceptually very simple to understand
- Problem? Too many models to fit! How many?
- 2^p models to fit.
- For example: for $p = 40$, there are 1 073 741 824 models to fit!
- So we need another solution.

Subset Selection

2. Stepwise selection
   - Forward
   - Backward

Forwards stepwise selection

- Computationally efficient alternative to the best subset selection
- Here we begin with the null model and add predictors **one at the time** until we get the full model (or some stopping rule is applied)
- Then we choose among these models using cross-validation, AIC, BIC, etc.

Subset Selection

More formally:

Forwards stepwise selection: for a linear model with p predictors do

- Let M_0 be the null model with zero regressors, i.e. sample mean of Y is used as a predictor
- For $k = 0,1, \dots, p-1$
    1. Consider all $p-k$ models that add one additional predictor to the model M_k
    2. Pick the best among these $p-k$ models and call it M_{k+1}. I.e. choose the model with the largest R^2.
- Select the best model from $M_0, M_1, \dots, M_p$ using cross-validation, AIC, BIC, etc.
- Note: here you cannot use R^2 because then the largest model would always be chosen.

Subset Selection

- Here we fit only $1 + \sum_{k=0}^{p-1}(p-k) = 1 + \frac{p\,(p+1)}{2}$ models
- For example: for $p = 40$, there are **466** models to fit. Much better than before.
- This procedure works well in practice, but now there is no guarantee that we will select the best method overall

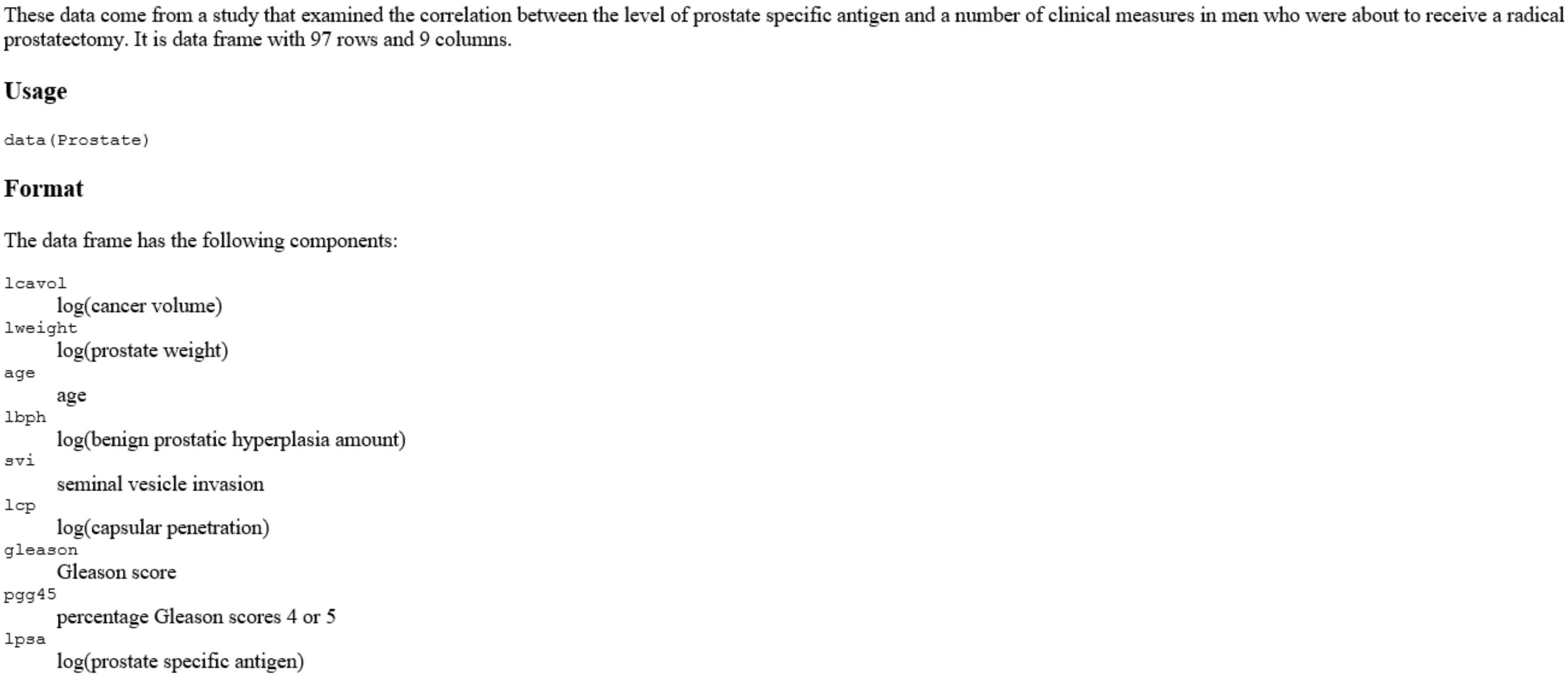
Backwards stepwise selection:

- Similar: here you start with the full model and delete regressors one at the time

Example: Prostate cancer

- The data come from a study that examined the correlation between the level of **prostate specific antigen (response variable)** and a number of clinical measures **(regressors)** in men who were about to receive a radical prostatectomy.
- It is data frame with 97 rows and 9 columns.

Example: Prostate cancer



These data come from a study that examined the correlation between the level of prostate specific antigen and a number of clinical measures in men who were about to receive a radical prostatectomy. It is data frame with 97 rows and 9 columns.

Usage

```
data(Prostate)
```

Format

The data frame has the following components:

`lcavol`
    log(cancer volume)

`lweight`
    log(prostate weight)

`age`
    age

`lbph`
    log(benign prostatic hyperplasia amount)

`svi`
    seminal vesicle invasion

`lcp`
    log(capsular penetration)

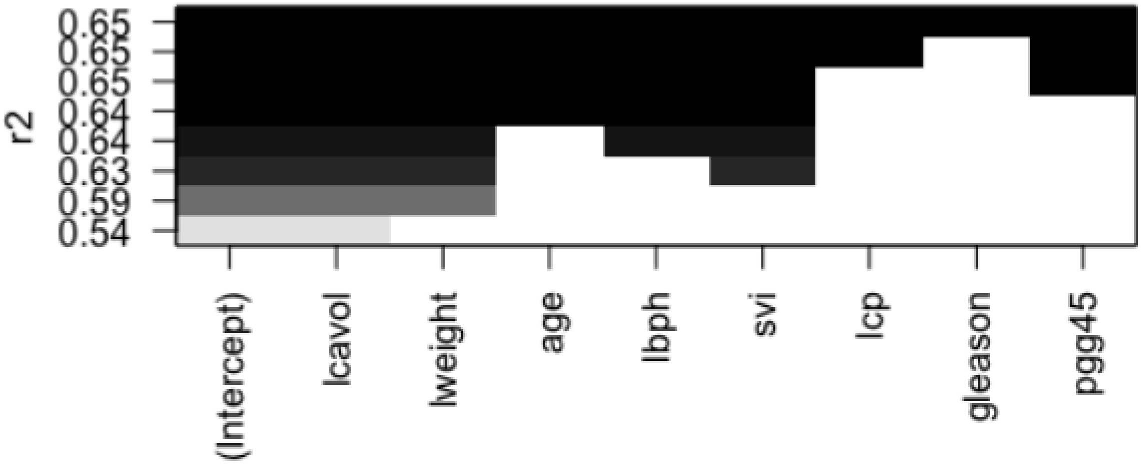
`gleason`
    Gleason score

`pgg45`
    percentage Gleason scores 4 or 5

`lpsa`
    log(prostate specific antigen)

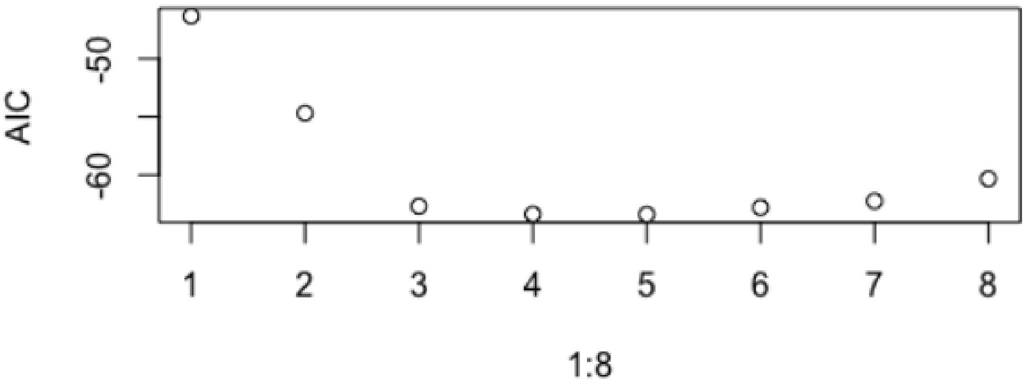
Example: Prostate cancer

- R Package **Leaps** is used to select the best model (based on R^2) of each size

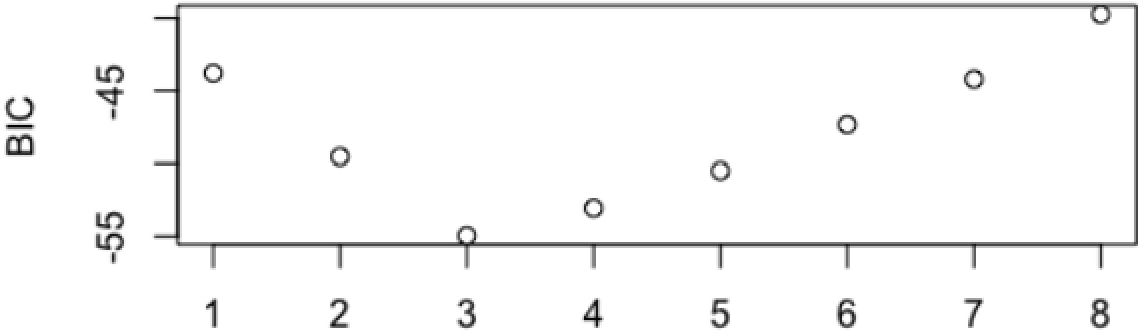


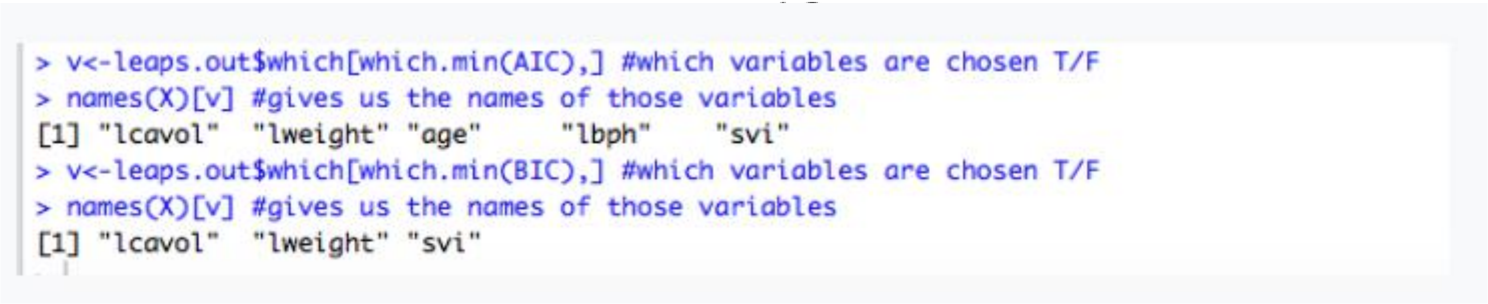
Example: Prostate cancer

- Then AIC and BIC are calculated for each of these models, based on the formula for linear regression with normal errors.



Example: Prostate cancer





```
> v<-leaps.out$which[which.min(AIC),] #which variables are chosen T/F
> names(X)[v] #gives us the names of those variables
[1] "lcavol"  "lweight" "age"     "lbph"    "svi"
> v<-leaps.out$which[which.min(BIC),] #which variables are chosen T/F
> names(X)[v] #gives us the names of those variables
[1] "lcavol"  "lweight" "svi"
```

Summary of Day 1

Summary

- We assess the model quality by its **prediction error**

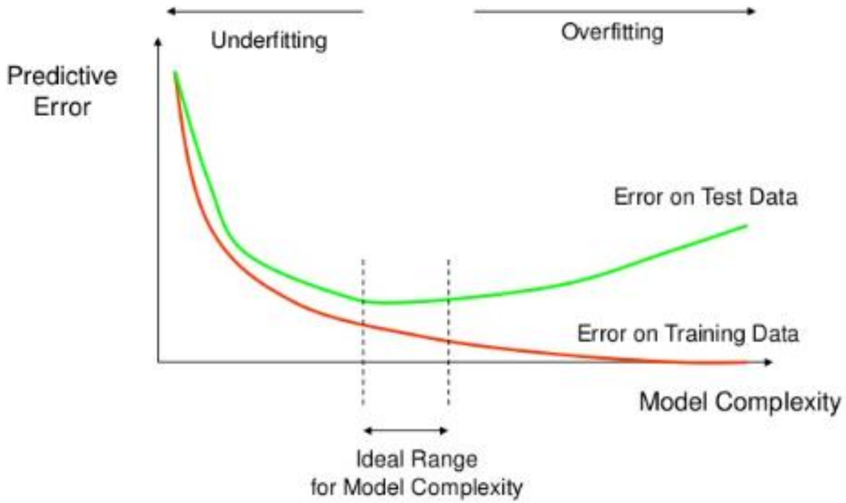
$$\boldsymbol{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{f}(X_i))^2}$$

given a sample $(X_i, Y_i)_{i=1}^{n}$.

- But this only one part of it – **training (in-sample) error**
- It is necessary to estimate this error for new (unseen) data – **testing (out-of-sample) error**

Summary

- A model (and its complexity) should be chosen based on these two prediction errors:



Summary

- The training error we can estimate from the sample directly
- There are two types of methods for estimating the testing error
  1. Cross – validation: based on **resampling**
  2. AIC, BIC, etc.: based on **testing error** ≈ **training error + dimension penalty**

Summary

- **Linear models**: simple but widely-used because of it simplicity and interpretability
- OLS well-defined for $n \geq p$
- But if performs badly if
  - p is large compared to n
  - some of the regressors are highly correlated

Summary

- Some methods to reduce the number of parameters:
  1. Best subset selection: **all submodels** are considered, but this is computationally infeasible
  2. Stepwise-regression: regressors are added **one at the time.** Once a regressor is chosen, it stays

Other Methods

Preview

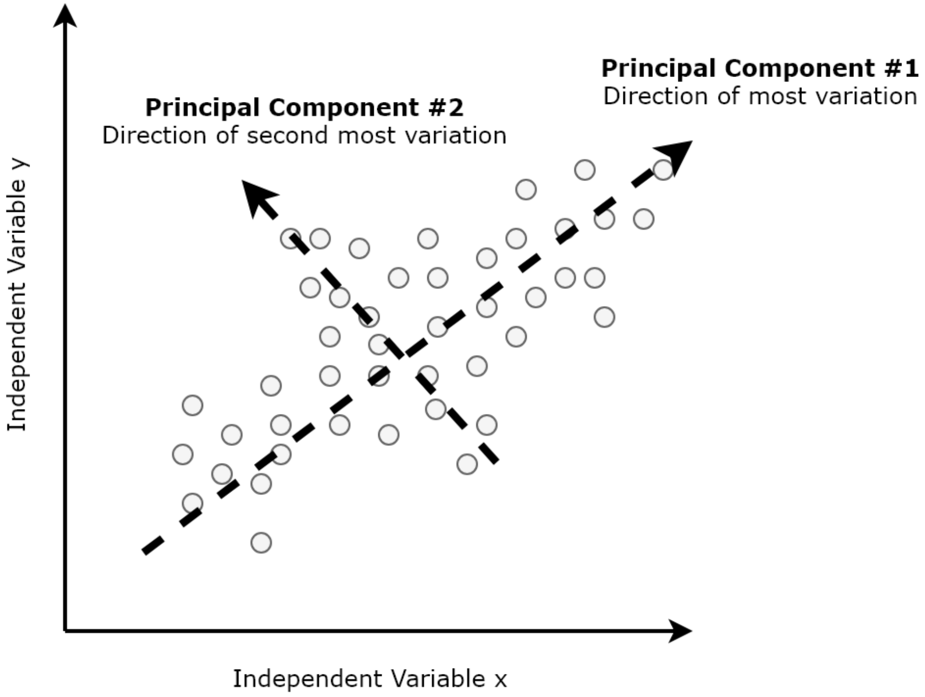
- We are still to see:
  - Some other methods that do model selection for linear models
  - How to deal with correlations
  - How to deal with $p > n$ case?

Principal Component Analysis

Principal Component Regression

- PCA uses an orthogonal transformation to convert a set of possibly correlated variables into a set linearly uncorrelated variables called **principal components**.
- This transformation is defined in such a way that the first principal component has the largest variance, the second principal component the second largest, etc.
- This way a dimension reduction can be performed and consequently OLS can be fitted using the newly obtained regressors.
- One can show that this reduces the variance of the OLS estimator

Principal Component Regression



Principal Component Regression

- The only issue with this procedure is that the new regressors have lost the interpretability, because they are linear combinations or the original regressors.
- But if the prediction is the only goal, then this procedure is more than suitable.

Shrinkage Methods

Shrinkage Methods

- We have already mentioned that if p is relatively large compared to n, or if some regressors are highly-correlated then the OLS estimates can be very variable and therefore unstable.
- Also we cannot do OLS for $p > n$.
- In order to tackle these problems, shrinking the regression coefficients is helpful

Shrinkage Methods

- We know that if OLS is defined then (**Gauss-Markov**)
  - it is unbiased
  - has the smallest variance among all **unbiased linear estimators**
- So, if we want to stay in the class of unbiased linear estimators, we cannot further reduce the variance.
- **Idea**: introduce a little bit of bias to decrease the variance significantly

Ridge estimator

- Let $\lambda \geq 0$ be fixed. Then the **Ridge estimator** is defined as:

- $\hat{\beta}_{\lambda} = arg \min_{\beta \in R^p} (||Y - X\beta||_2^2 + \lambda||\beta||_2^2) = \ arg \min_{||\beta||_2^2 \leq c} ||Y - X\beta||_2^2$

  for some c that depends on λ.

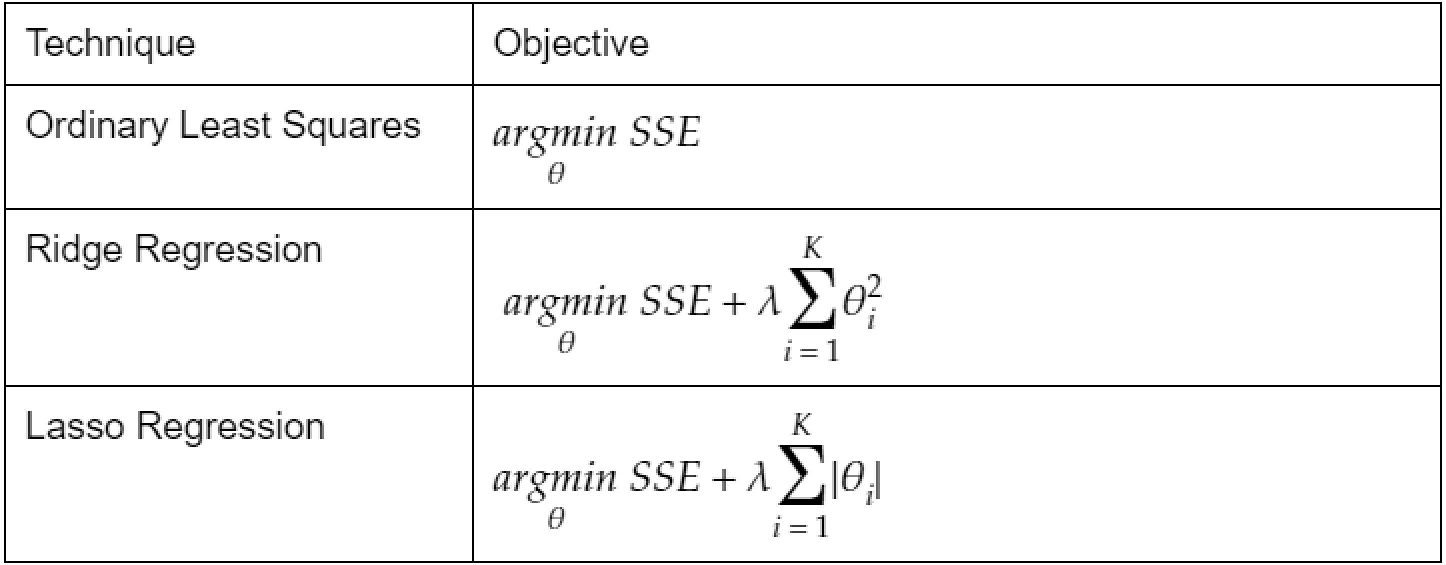
- For $\lambda = 0$, we obtain OLS. Otherwise we obtain a **biased estimator with smaller variance** than OLS

Lasso estimator

- Let $\lambda \geq 0$ be fixed. Then the **Lasso estimator** is defined as:
- $\hat{\beta}_\lambda = arg \min_{\beta \in R^p} (||Y - X\beta||_2^2 + \lambda||\beta||_1) = \ \ arg \min_{||\beta||_1 \leq c} ||Y - X\beta||_2^2$

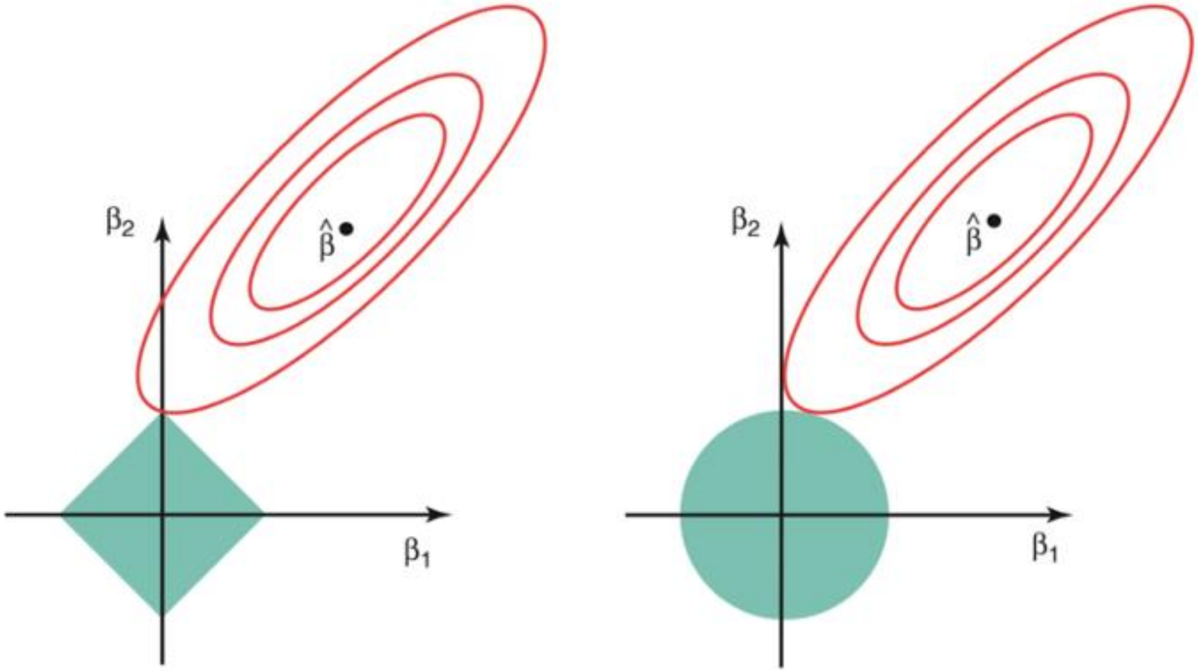
  for some c that depends on λ.
- For $\lambda = 0$, we obtain OLS. Otherwise we obtain a biased estimator than in most cases outperforms the OLS

Shrinkage Methods



Technique	Objective
Ordinary Least Squares	$\underset{\theta}{argmin}\ SSE$
Ridge Regression	$\underset{\theta}{argmin}\ SSE + \lambda \sum_{i=1}^{K} \theta_i^2$
Lasso Regression	$\underset{\theta}{argmin}\ SSE + \lambda \sum_{i=1}^{K} \lvert\theta_i\rvert$

Shrinkage Methods – geometrical interpretation



Shrinkage Methods

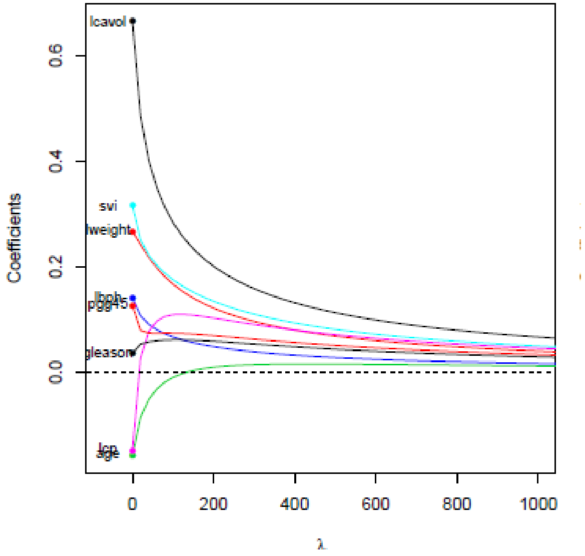
- For both estimators, estimators for β coefficients will be now **bounded**, which means that also the variance of the estimates stays controlled
- How to choose the right λ? Cross validation!

Model selection

- Ridge estimator will almost surely not set any estimated coefficients to zero because of its L2 geometry
- On the other hand, that is exactly what happens with Lasso estimates, because of the L1 norm.
- The larger the λ the more coefficients are set to 0.
- So Lasso performs **model selection and estimation** at the same time

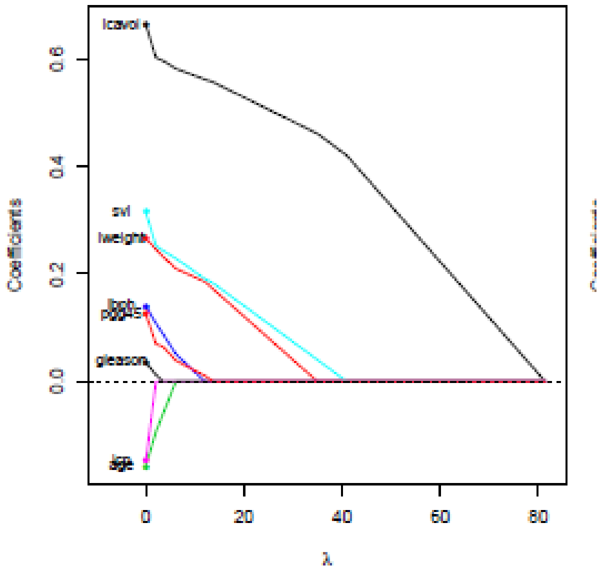
Example – Prostate data

- The more you increase λ, the smaller the estimated coefficients are
- Ridge estimated coefficients:



Example – Prostate data

- The more you increase λ, the smaller the estimated coefficients are
- Lasso estimated coefficients: here they are set to 0 for large λ



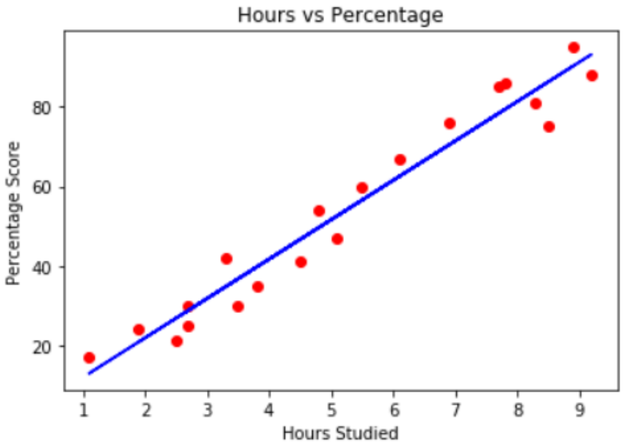
Generalized linear models (GLM)

GLM

- Generalized linear models (GLM) are a natural extension of linear models
- Response variable is now **function** of a **linear combination** of regressors
- Response variable does not have to be distributed normally anymore, it can take on of the distributions from the **exponential family**: Bernoulli, Binomial, Poisson, Gamma, Exponential
- GLMs are widely used in insurance industry and are ideally suited for the analysis of the non-normal data, that is commonly encountered in insurance.

GLM

- More formally: $Y_i \in R$ - response variable, $X_i \in R^p$ - regressors
- Linear regression: $E(Y_i | X_i) = \beta' X_i$ and $\widehat{Y_i} = \hat{\beta}' X_i$.



GLM

- But what if Y_i is a **count** variable, like the number of claims?
- Assume a Poisson distribution for each Y_i, but with a (potentially) different parameter $\lambda_i > 0$. Each customer has different frequency of claims.
- We want to model Y_i in terms of X_i.

GLM - Poisson regression

- We know that

$$P(Y_i = y|\lambda_i) = (e^{-\lambda_i}{\lambda_i}^y)/y! \text{ for each } y \in \{0,1,2,\ldots\}.$$

- Also $E(Y_i|X_i) = \lambda_i$. We want to model λ_i in terms of $\beta' X_i$. Since $\lambda_i > 0$, it makes sense to do the following parametrization

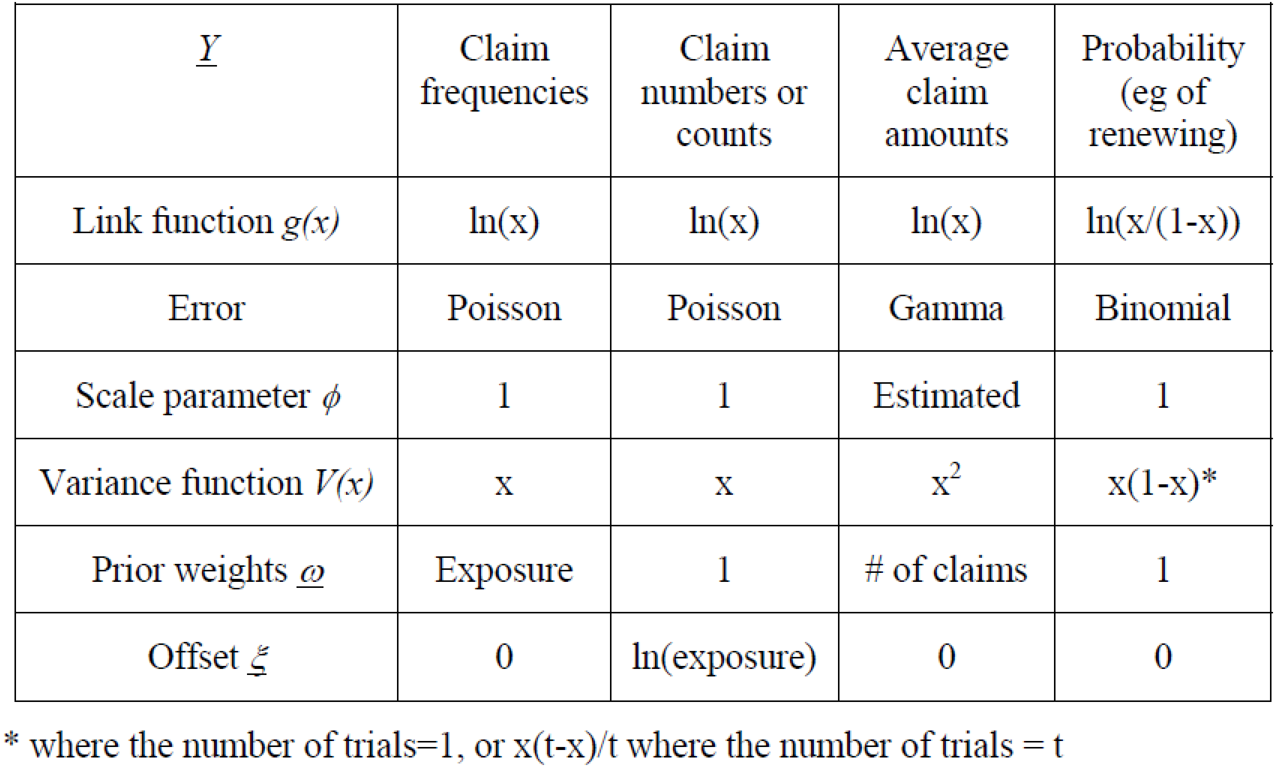
$$E(Y_i|X_i) = \lambda_i = e^{\beta' X_i}$$

- Estimator: $\widehat{Y}_i = e^{\widehat{\beta}' X_i} = e^{\widehat{\beta}_1' X_{i1}} \ldots e^{\widehat{\beta}_p' X_{ip}}$.

Multiplicative structure

- GLM: a generalization of this, to also allow for other distributions in the exponential family: Normal, Exponential, Gamma, Bernoulli, Binomial, etc.

GLM - Model choice



$\underline{Y}$	Claim frequencies	Claim numbers or counts	Average claim amounts	Probability (eg of renewing)
Link function $g(x)$	ln(x)	ln(x)	ln(x)	ln(x/(1-x))
Error	Poisson	Poisson	Gamma	Binomial
Scale parameter ϕ	1	1	Estimated	1
Variance function $V(x)$	x	x	x^2	x(1-x)*
Prior weights $\underline{\omega}$	Exposure	1	# of claims	1
Offset $\underline{\xi}$	0	ln(exposure)	0	0

* where the number of trials=1, or x(t-x)/t where the number of trials = t

Source: Willis Towers Watson

$$\mathrm{E}(\mathbf{Y}) = \boldsymbol{\mu} = g^{-1}(\mathbf{X}\boldsymbol{\beta})$$

GLM

- Generalized Linear Models serve as the industry standard for non-life insurance pricing
  - Multiplicative output remains understandable also for non-actuaries
  - Range of professional insurance software dedicated to GLM
  - GLM is also possible in, for example, R
- **Burning costs** are defined as **Frequency** × **Severity**
- One can model the average frequency of claims, the average claim amounts (severity) or (directly) the average burning costs
- Burning costs are then the basis for the (net) risk premium

Criteria for GLM

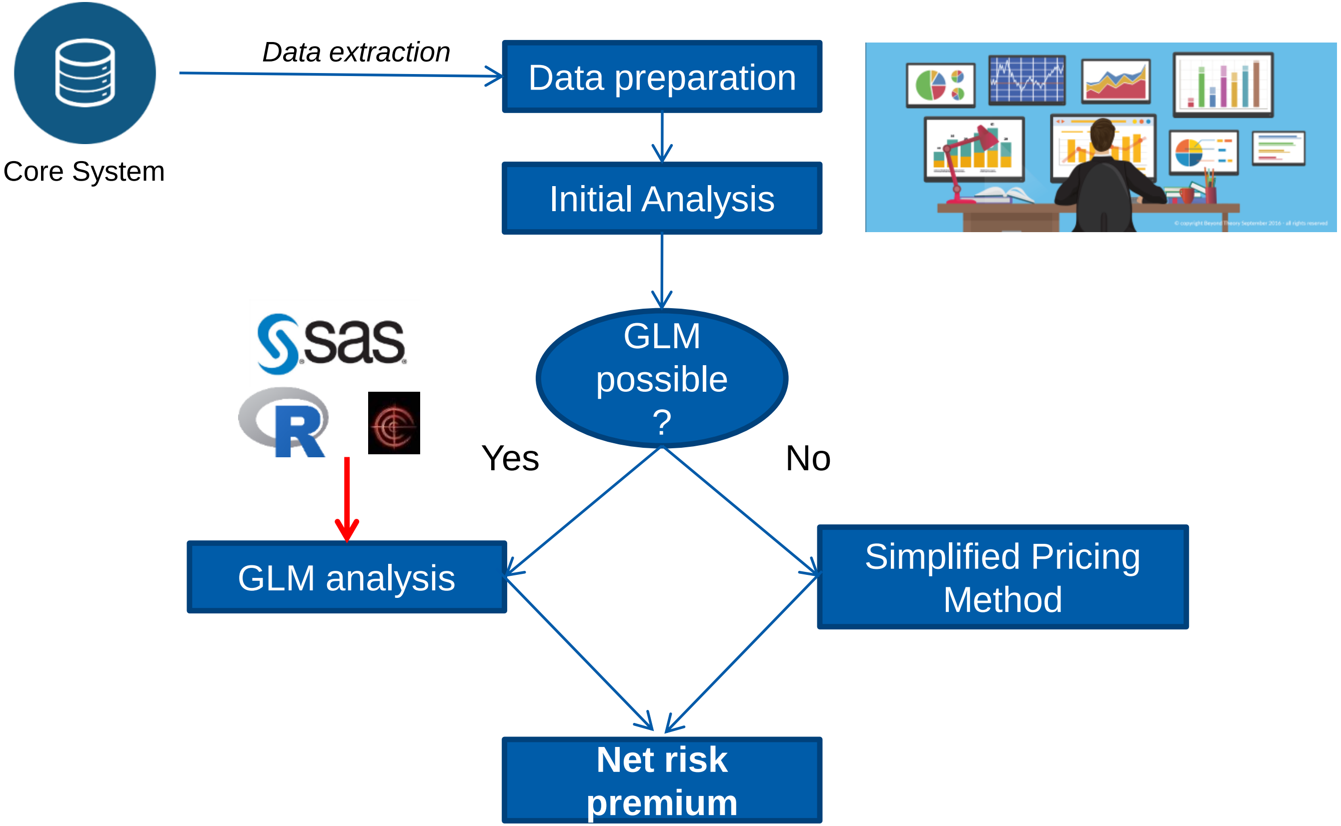
- Portfolio size
  - 150.000 exposure rows is seen as a minimum
  - A significant number of claims is required as well
- Homogeneity of the risks in the portfolio
- Possibility to segment the risks
  - Available risk factors

Often these criteria are met for a part of, but not for the full portfolio

- Alternative methods
  - Other pricing techniques
  - Flat premium or premium influenced by one risk factor
  - Individual underwriting

Pricing Process

Risk Modelling Process



Core System

Data extraction

Data preparation

Initial Analysis

GLM possible ?

Yes

No

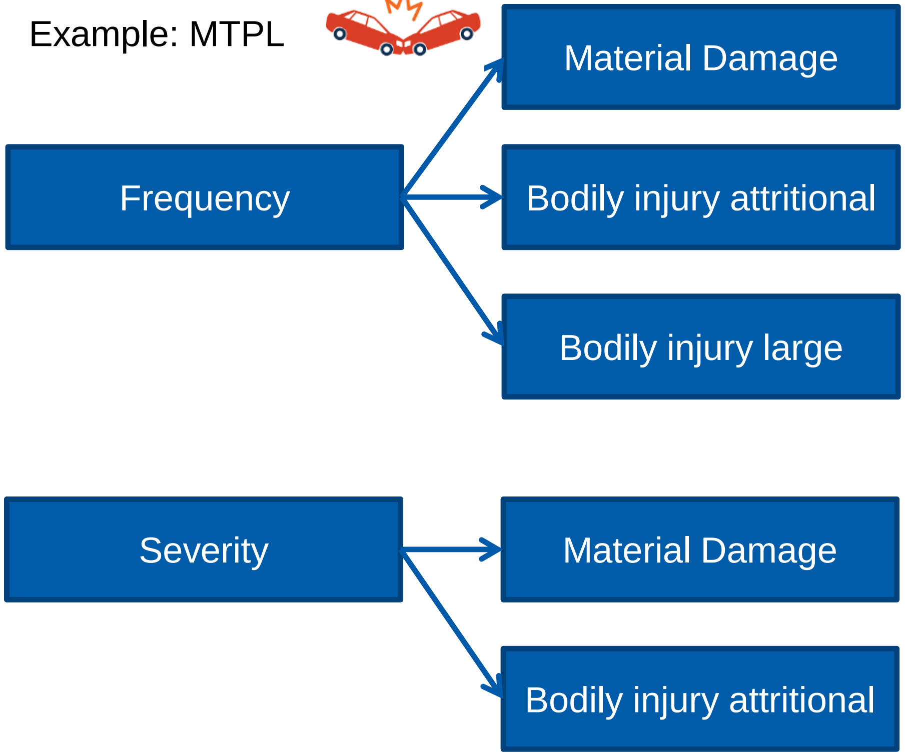
GLM analysis

Simplified Pricing Method

Net risk premium

You could end up with a multitude of models

Example: MTPL



Frequency

Material Damage

Bodily injury attritional

Bodily injury large

Severity

Material Damage

Bodily injury attritional

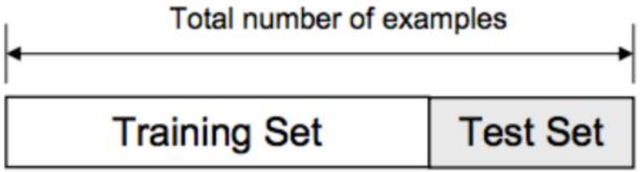
Separate models possible for private persons, fleets, leasing, etc

And this is just for passenger cars!

In this example the severity of large BI claims is not modelled, but taken as a fixed amount per claim

Validation of a GLM model

- Split the dataset in two
  - Usually a 80-20% split or out-of-time
  - Check how the model performs on unseen data
  - Avoid overfitting

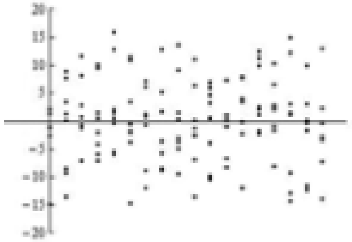


Total number of examples

Training Set

Test Set

- Significance tests
  - Significance of a parameter in the model
  - Significance of levels of a parameter against each other (how granular should a variable be)
- Temporal stability
  - To be significant, an effect must be stable over the years
- Residual analysis
  - To test the distribution
  - On real data no distribution works perfectly

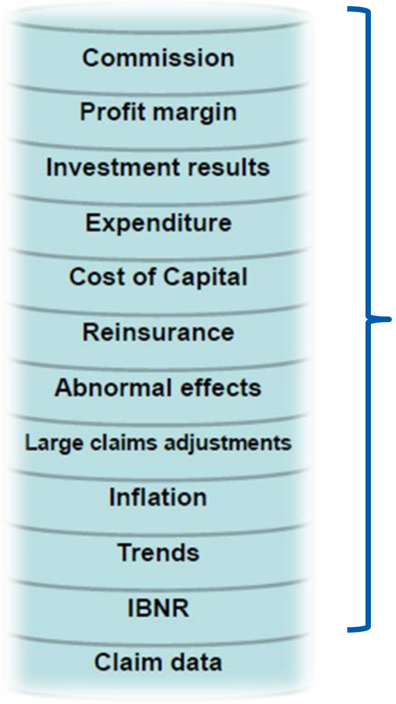


Combining Risk Models

- In the end we need to deliver a final risk premium
  - We should combine all models we made
  - Necessary to understand the total effect

- Result: net risk premium!

From net risk premium to gross risk premium



Commission

Profit margin

Investment results

Expenditure

Cost of Capital

Reinsurance

Abnormal effects

Large claims adjustments

Inflation

Trends

IBNR

Claim data

- A whole range of effects is to be added to the net risk premium
- Most loadings will be added through an increase in the intercept, but there are other possibilities
- Loading for discounts to be added

Gross Risk Premium

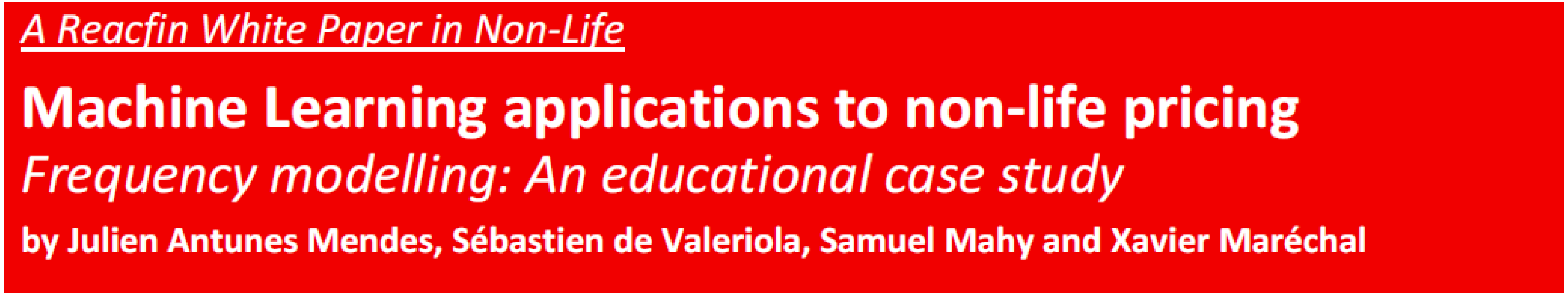
Additional Topics - Interactions

Interactions and GLM

- An **interaction** effect exists when the effect of an independent variable on a dependent variable changes, depending on the value(s) of one or more other independent variables
- In that case an interaction term(s) has to be added to the model
- Example: gene A and gene B may contribute to developing a certain disease, but in combination they are fatal

Interactions and GLM

- The problem? GLM models do not detect interactions automatically
- Then can be added to the model, but this has to be done ‘manually’
- Example taken from:



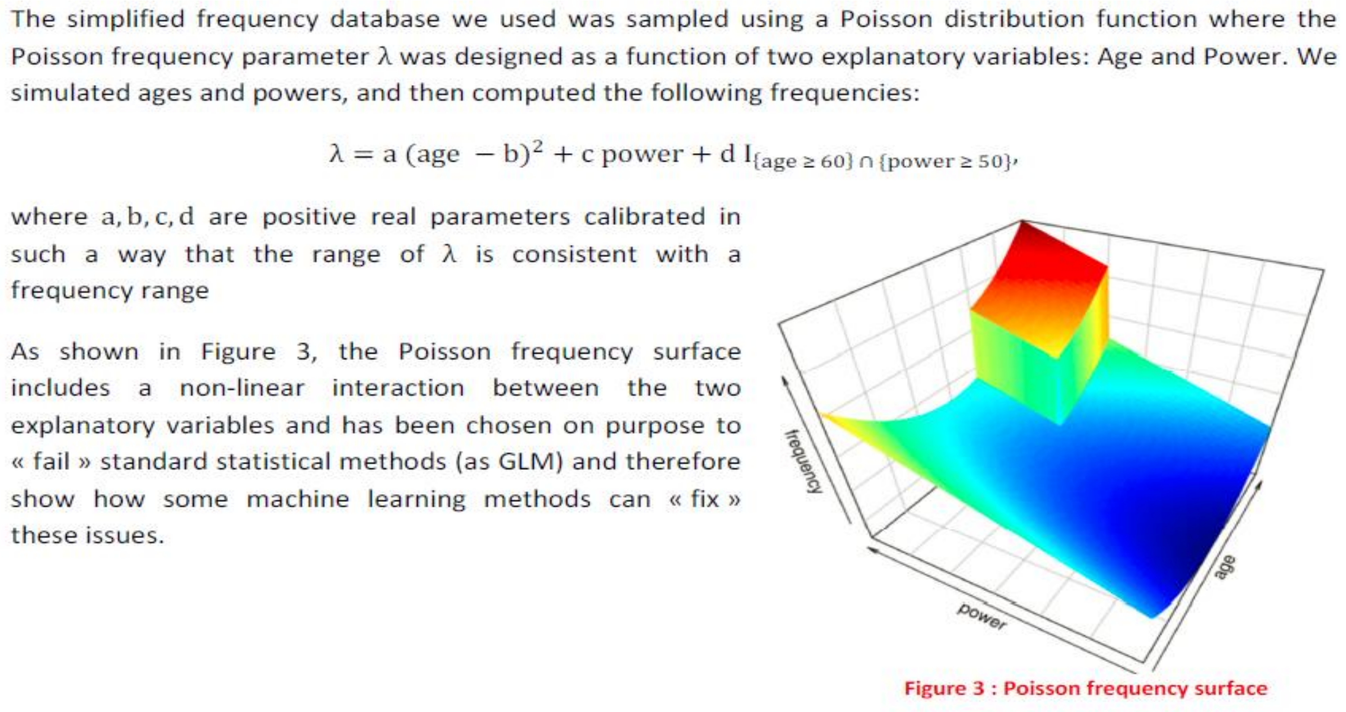
A Reacfin White Paper in Non-Life

Machine Learning applications to non-life pricing

Frequency modelling: An educational case study

by Julien Antunes Mendes, Sébastien de Valeriola, Samuel Mahy and Xavier Maréchal

Interactions and GLM



The simplified frequency database we used was sampled using a Poisson distribution function where the Poisson frequency parameter λ was designed as a function of two explanatory variables: Age and Power. We simulated ages and powers, and then computed the following frequencies:

$$\lambda = a\,(\text{age} - b)^2 + c\,\text{power} + d\,I_{\{\text{age} \geq 60\} \cap \{\text{power} \geq 50\}},$$

where a, b, c, d are positive real parameters calibrated in such a way that the range of λ is consistent with a frequency range

As shown in Figure 3, the Poisson frequency surface includes a non-linear interaction between the two explanatory variables and has been chosen on purpose to « fail » standard statistical methods (as GLM) and therefore show how some machine learning methods can « fix » these issues.

Figure 3 : Poisson frequency surface

Interactions and GLM

- In this example: there is an interaction of age and engine power
- $\boldsymbol{Age \geq 60\ and\ Engine\ Power \geq 50}$
- But if this effect is not noticed and included in the model, the GLM fit is poor

Interactions and GLM

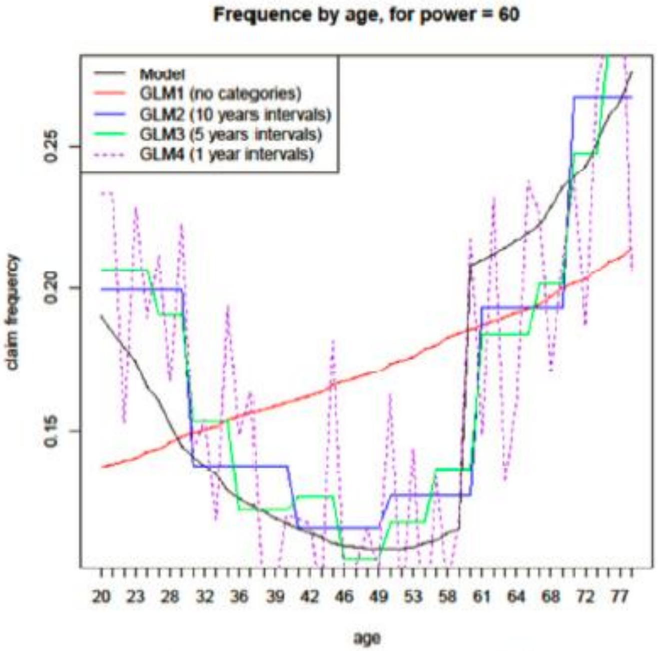


Figure 8: Frequency by age with GLM

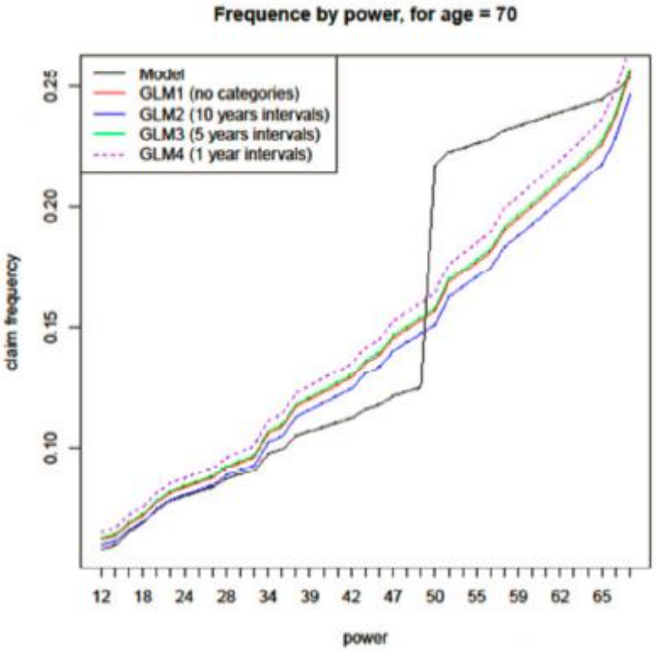


Figure 9: Frequency by power with GLM

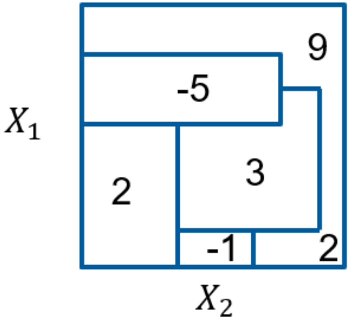
Machine Learning

Boosting

- But many machine learning algorithms can automatically capture these effects
- Let us take Gradient Boosting Trees for example
- How does this algorithm work?
- Let us present some basics

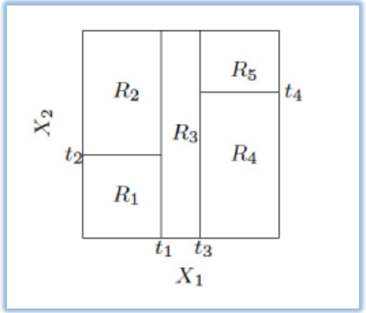
Tree based methods

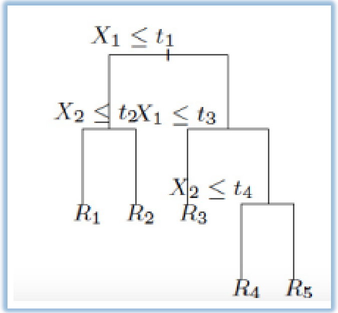
- Tree-based methods partition the feature space into a set of rectangles and then fit a simple model (typically a constant) in each region
- Consider a regression problem with continuous response Y and continuous regressors $X_1, X_2 \in (0,1)$.
- For example, this partition is simple but cannot be obtained by recursive binary splitting, i.e. represented by a tree.



Tree based methods

- So, let us restrict our attention to **recursive binary partitions**, like this one:

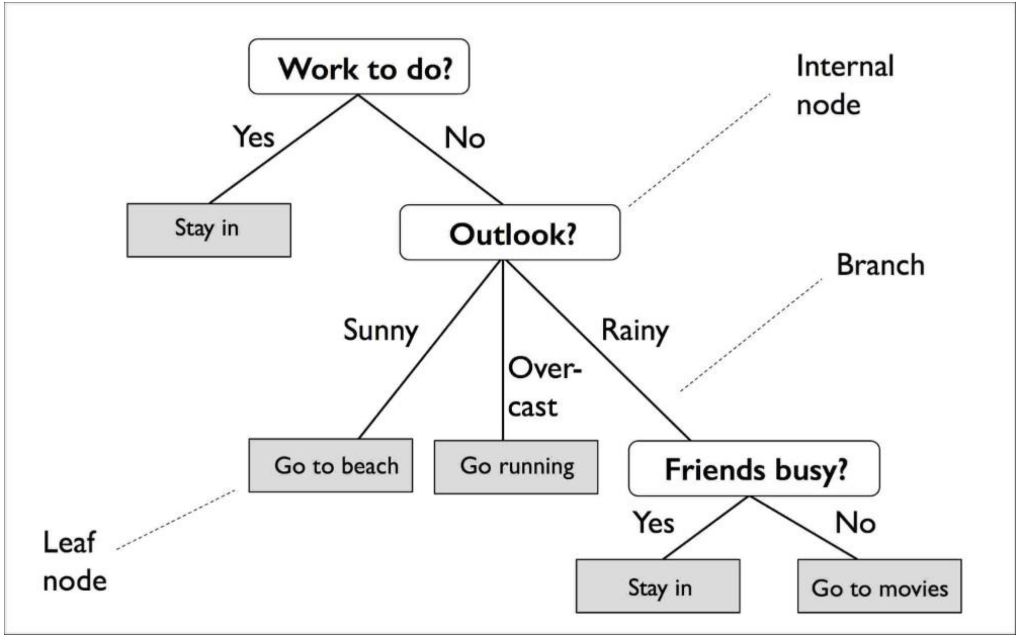




- First split the space into two regions and model the response by the mean of Y in each region. Choose the split variable and split-point to achieve the optimal split.
- Then one or both regions are further split in the same fashion iteratively until some stopping rule is applied.

Tree based methods

- The corresponding regression model predicts Y with a constant c_m if the inputs X are in region R_m, i.e.

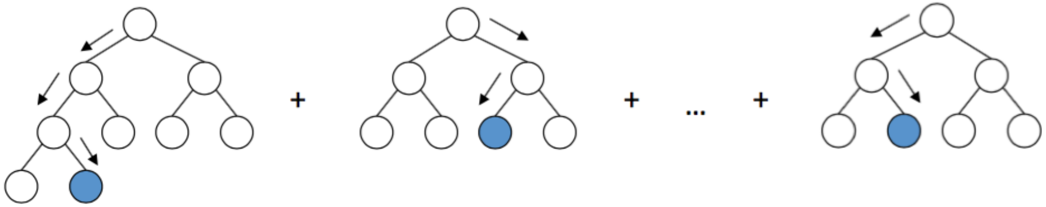


Tree based methods

- These trees can now further be used for boosting
- What is boosting?

Boosting

- Gradient boosting is one of the most powerful techniques for building predictive models. It is proven successful in many areas and is one of the leading methods for winning Kaggle competitions



Boosting

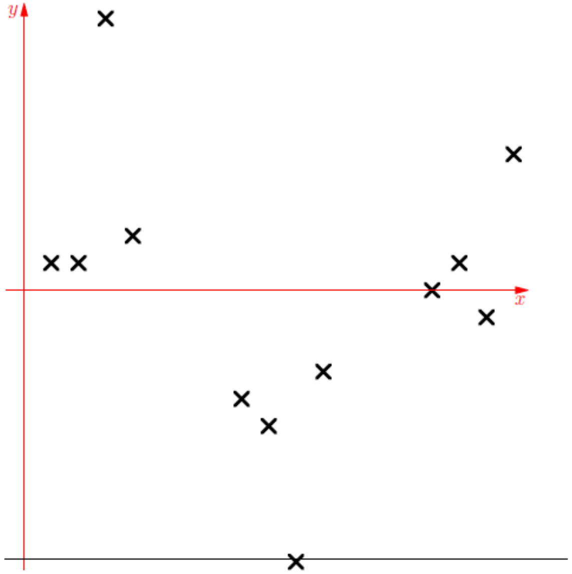
- In general: models can be fitted to data individually or combined in an ensemble – a combination of simple individual models (usually trees) that together create a more powerful model
- Boosting is a method that builds the model in a **stage-wise fashion**.
- It starts by fitting an initial model.
- The second model focuses then on accurately predicting the cases where the first model performed badly
- The third model focuses on correcting the faults of the previous stage, etc.

Boosting

- Here we do not fit one big decision tree to the model, because this can easily lead to overfitting
- Instead, the boosting algorithm learns slowly
- At each step we fit a decision tree to the residuals from the previous model
- Then new tree is then added to the model

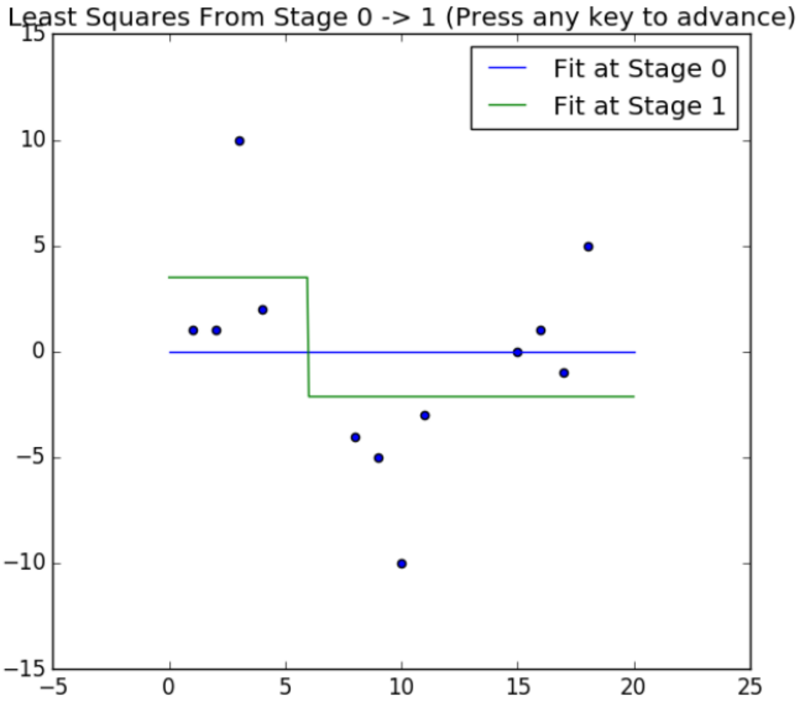
Boosting

- Example: data to be fitted

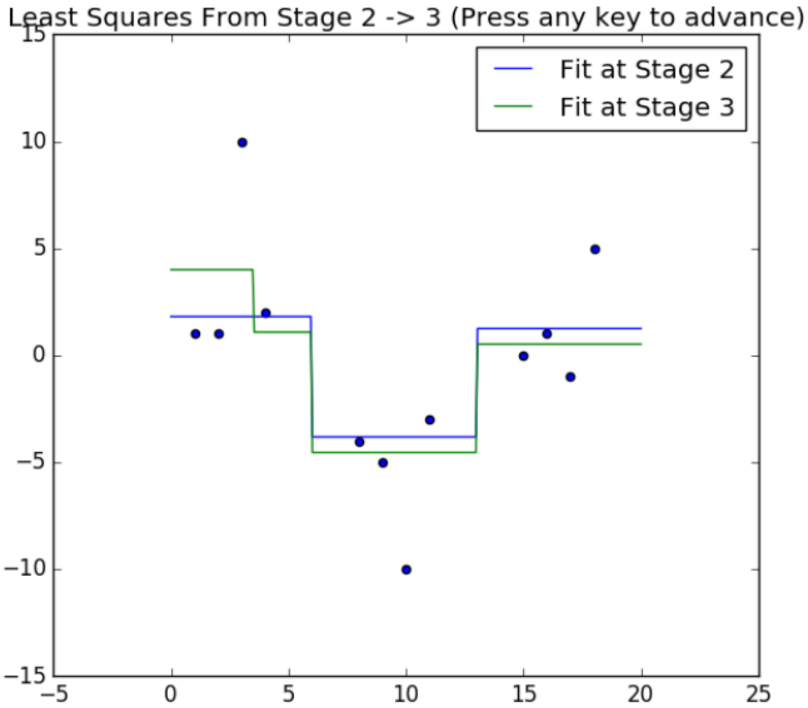


Plot courtesy of Brett Bernstein.

Boosting

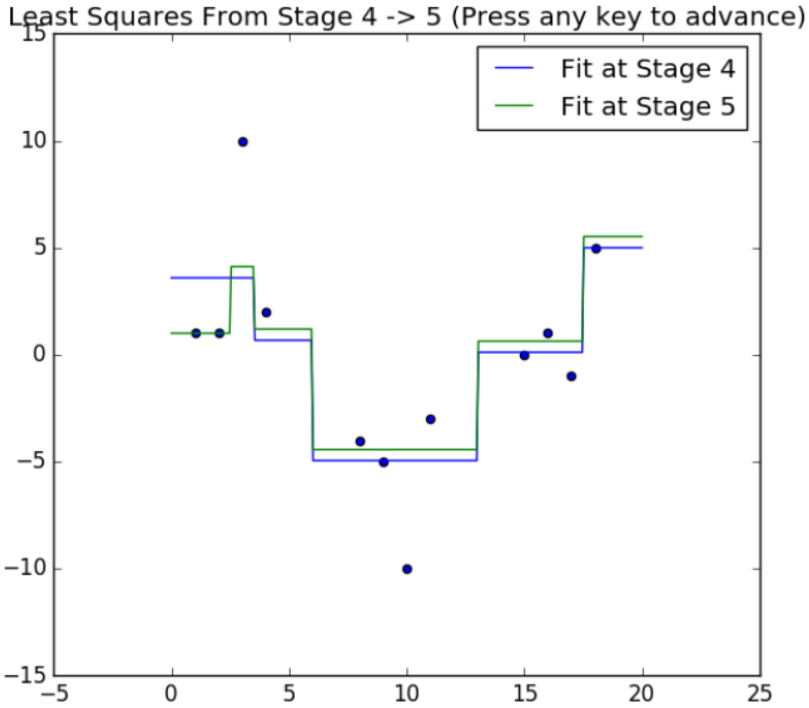

Least Squares From Stage 0 -> 1 (Press any key to advance)
Fit at Stage 0
Fit at Stage 1
15
10
5
0
−5
−10
−15
−5
0
5
10
15
20
25

Boosting



Least Squares From Stage 2 -> 3 (Press any key to advance)
Fit at Stage 2
Fit at Stage 3
15
10
5
0
−5
−10
−15
−5
0
5
10
15
20
25

Boosting


Least Squares From Stage 4 -> 5 (Press any key to advance)
Fit at Stage 4
Fit at Stage 5
15
10
5
0
−5
−10
−15
−5
0
5
10
15
20
25

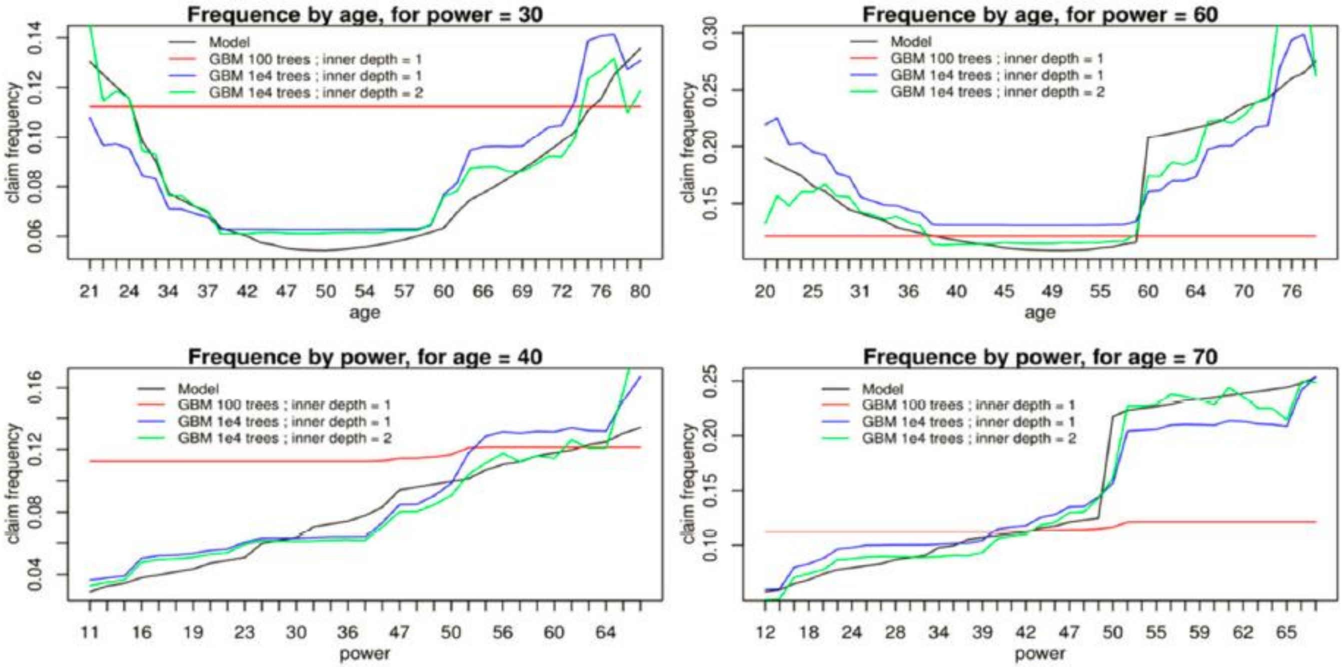
Boosting

- Usually the trees are rather small, but they should be deep enough to capture interactions. Number of splits = 2 is already enough to catch first-order interactions
- There are several parameters that need to be chosen: the number of trees, the number of splits in each tree and the learning rate of the algorithm (usually 0.1 or 0.01)
- For the number of trees cross-validation is used

Example

- Back to our example
- Remember that GLM could not ‘recognize’ the interaction between age and engine power
- But GBMs do, provided that the tuning parameter have be carefully selected

Example


Frequence by age, for power = 30
Model
GBM 100 trees ; inner depth = 1
GBM 1e4 trees ; inner depth = 1
GBM 1e4 trees ; inner depth = 2
claim frequency
0.06 0.08 0.10 0.12 0.14
21 24 34 37 42 47 50 54 57 60 66 69 72 76 80
age
Frequence by age, for power = 60
Model
GBM 100 trees ; inner depth = 1
GBM 1e4 trees ; inner depth = 1
GBM 1e4 trees ; inner depth = 2
claim frequency
0.15 0.20 0.25 0.30
20 25 31 36 40 45 49 55 60 64 70 76
age
Frequence by power, for age = 40
Model
GBM 100 trees ; inner depth = 1
GBM 1e4 trees ; inner depth = 1
GBM 1e4 trees ; inner depth = 2
claim frequency
0.04 0.08 0.12 0.16
11 16 19 23 30 36 47 50 56 60 64
power
Frequence by power, for age = 70
Model
GBM 100 trees ; inner depth = 1
GBM 1e4 trees ; inner depth = 1
GBM 1e4 trees ; inner depth = 2
claim frequency
0.10 0.15 0.20 0.25
12 18 24 28 34 39 42 47 50 55 59 62 65
power

Method comparison

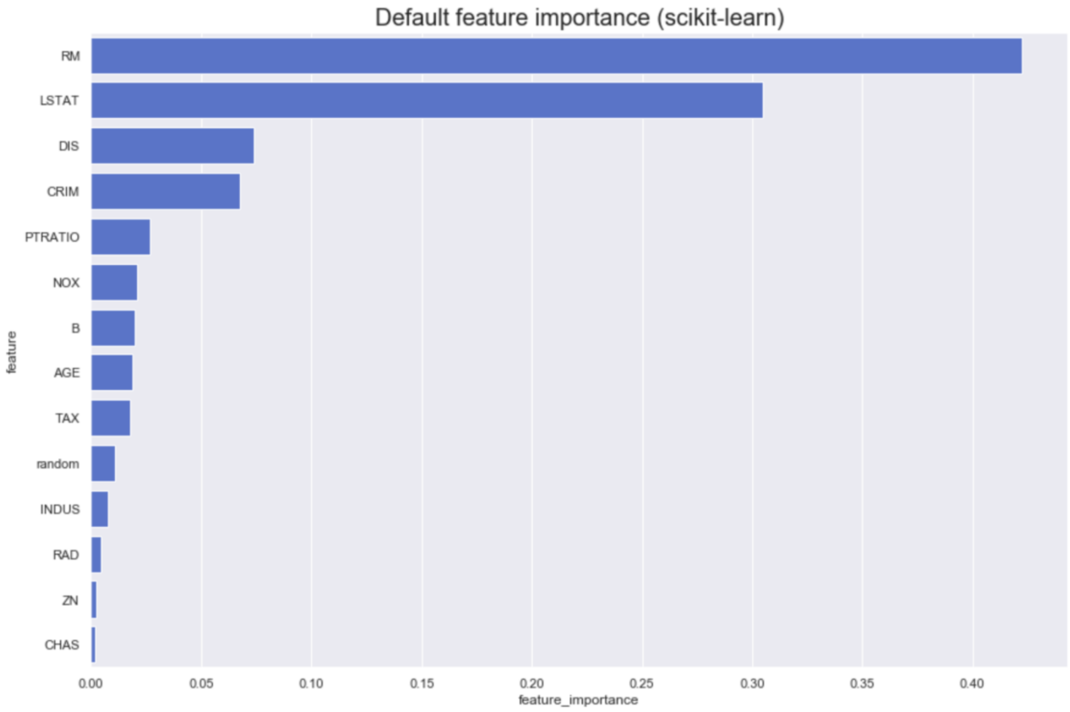
GLM vs Machine Learning

- The problem with these kind of algorithms is that the interpretation is almost completely lost
- It is very unlikely that such models will be approved by regulators, at least in the majority of countries
- And even if they are, then the insurance company runs into the risk of reputational loss, in case some of the ethical problems discussed before emerge

GLM vs Machine Learning

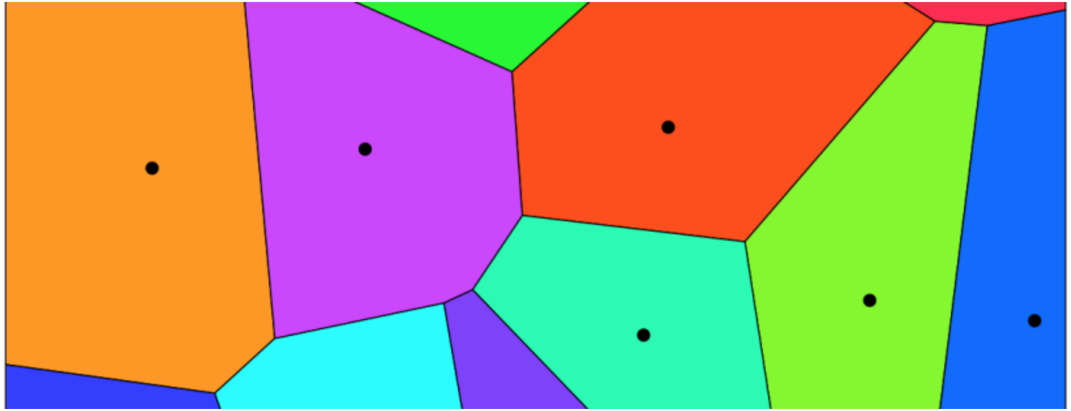
- Also the actuaries want to understand their models and not use black-box alternatives
- So, GLMs will probably not be replaced by Machine Learning algorithms in the near future
- But they can assist the actuaries in spotting interactions, as well as variable significance or perform clustering tasks

GLM vs Machine Learning



GLM vs Machine Learning

- Examples of clustering can be brand or region clustering. Here the black-box nature of the models is not so important, because the model results can usually easily be validated



Literature

Literature

- An Introduction to Statistical Learning with Applications in R - Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani
- https://www.reacfin.com/wp-content/uploads/2016/12/20170914-Machine-Learning-applications-for-non-life-pricing.pdf
- https://www.stat.cmu.edu/~ryantibs/datamining/lectures/17-modr2.pdf