Introduction to Statistical Learning

Eltecon Data Science Course by Emarsys

Péter Lukács

November 6, 2019

About me

- Eltecon BSc
- University of Amsterdam MSc in Economics
- Last 6+ years working with data
  - 2.5 year @ Emarsys as a Data Scientist
- lukacs.peter.andras@gmail.com

Goal of the lesson

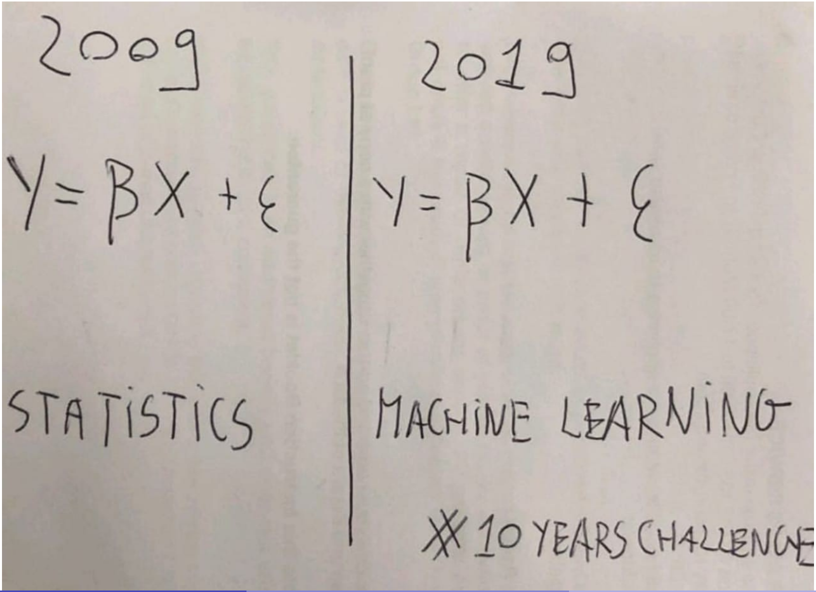
Section 1

Statistical Learning in General

Introduction to Statistical Learning

- tell about the book and what chapters are covered

What is Statistical Learning



What is Statistical Learning

*"**Machine learning** is all about results, it is likely working in a company where your worth is characterized solely by your performance. Whereas, **statistical modeling** is more about finding relationships between variables and the significance of those relationships, whilst also catering for prediction"*

source

What is Statistical Learning

Assumption:

$$Y = f(X) + \epsilon$$

- We **assume** a systematic relationship between X and Y
- f is generally unknown
- **Statistical Learning refers to a set of approaches for estimating f based on the available observations (X)**

What is Statistical Learning

Assumption:

$$Y = f(X) + \epsilon$$

- ϵ is assumed to have mean 0
- ϵ is assumed to be independent of X
  $\Rightarrow$ **otherwise** could be modeled through f

Why estimate f?

- Causality/Inference (more in Econ, e.g. What drives unemployment?)
- Prediction (more in Business, e.g. How much Happy Socks are we selling next month?)

Prediction: Reducible error/Irreducible error

$$Y = f(X) + \epsilon$$

$$\begin{aligned} E(Y - \hat{Y}) &= E[f(X) + \epsilon - \hat{f}(X)]^2 \\ &= \underbrace{[f(X) - \hat{f}(X)]^2}_{\text{reducible error}} + \underbrace{Var(\epsilon)}_{\text{irreducible error}} \end{aligned}$$

- the aim is to estimate f by reducing the reducible error
- What about the irreducible error? Can't do anything about that.
  - Didn't measure :(
  - Can't measure: e.g. mood of a buyer on the day she's buying the house

How to estimate f?

- parametric models

  + less parameters to learn (needs less training data)

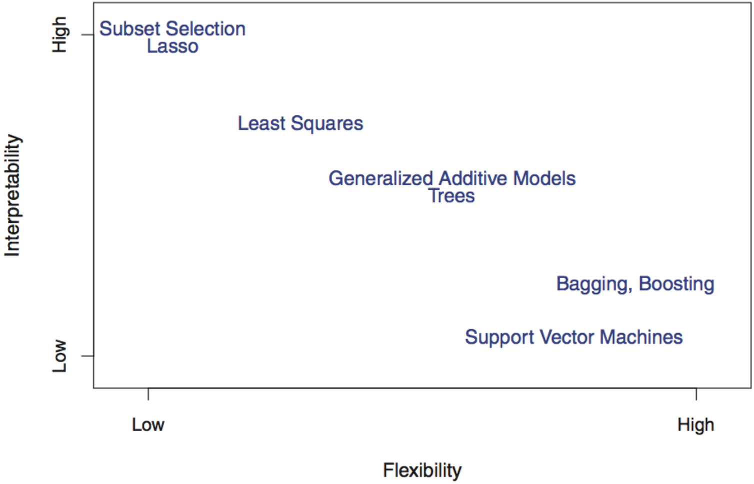
  - can erroneously assume f
- non-parametric models

  + more flexible

  - more parameters to learn (needs more training data)

  - can overfit the data

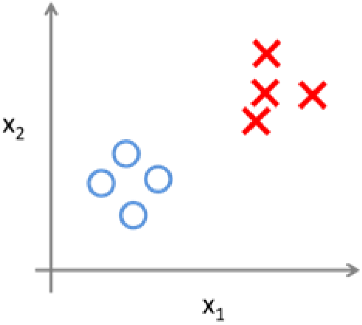
Prediction Accuracy vs. Model Interpretability



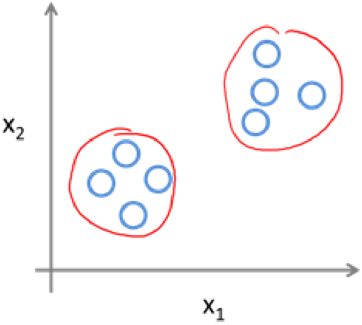
source: ISLR, p.25.

Supervised vs. Unsupervised Learning

- Supervised: has response variable (Y)
  - linear reg., logistic reg., GAM, SVC
- Unsupervised: no supervisor response variable
  - cluster analysis



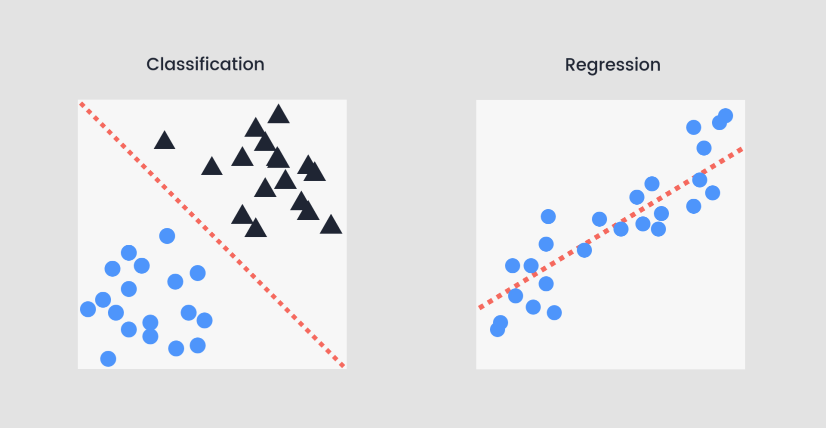
Supervised Learning



Unsupervised Learning

Regression vs. Classification

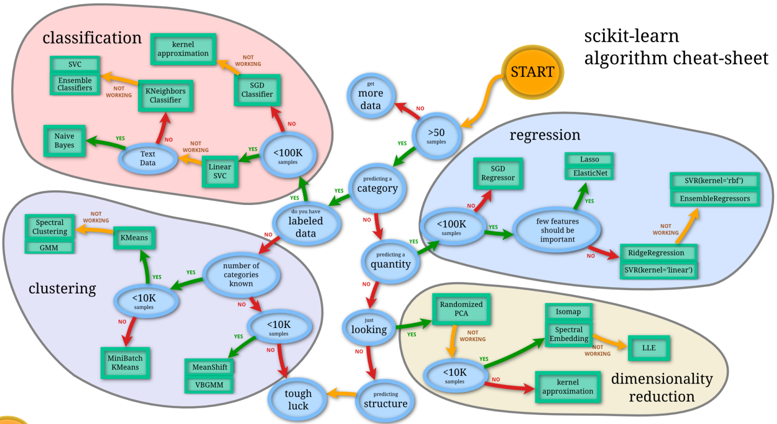
- Regression: quantitative response (e.g. market price prediction)
- Classification: qualitative response (e.g. male/female based on purchase patterns)



Statistical Learning Dimensions Summarized

- Goal: inference vs. prediction
- Model interpretability vs. Prediction Accuracy
- Supervised vs. Unsupervised
- Regression vs. Classification

Other model selection decision points



Back

source

Section 2

Linear Regression

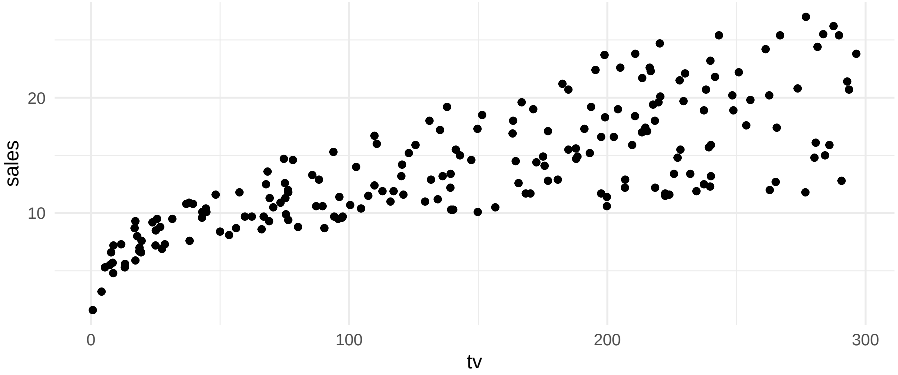
Simple Linear Regression Formula

- assumes an approximate linear relationship between X and Y

$$Y \approx \beta_0 + \beta_1 X$$

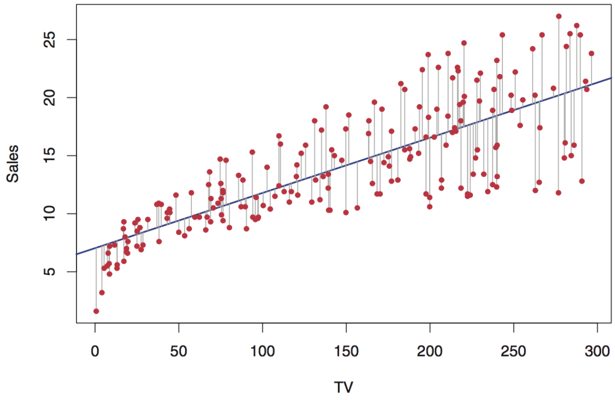
Simple Linear Regression: Advertising Data

```r
adv <- fread("./data/advertising_data.csv")
ggplot(adv, aes(tv, sales)) + geom_point() + theme_minimal()
```



Estimating Coefficients

We want to find the coefficients so that the resulting line is as “close” to the observations as possible.



source: ISLR, p.62.

Estimating Coefficients: Least Squares

- Minimize the *Residual Sum of Squares* (*RSS*)

$$RSS = (y_1 - \hat{\beta}_0 - \hat{\beta}_1 x_1)^2 + (y_2 - \hat{\beta}_0 - \hat{\beta}_2 x_2)^2 + ... + (y_n - \hat{\beta}_0 - \hat{\beta}_n x_n)^2$$

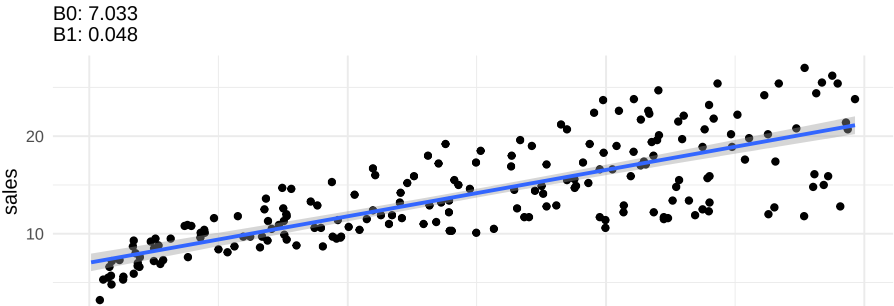
```
slm <- lm(formula = sales ~ tv, data = adv)
slm$coefficients
```

```
## (Intercept)          tv
##  7.03259355  0.04753664
```

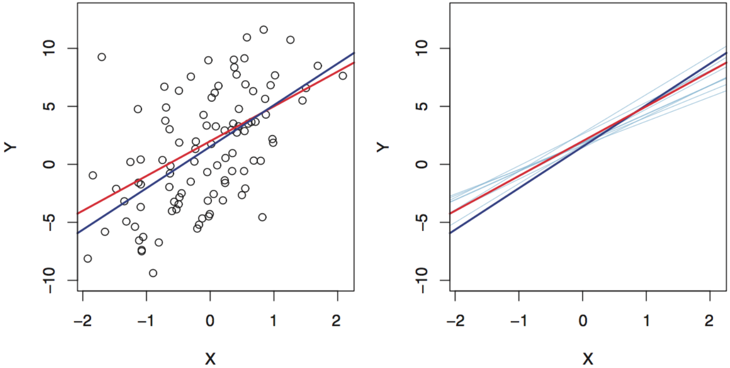
Type `names(slm)` to the console to see `slm`'s other attributes

Estimating Coefficients: Least Squares

```
int <- slm$coefficients["(Intercept)"]
b1  <- slm$coefficients["tv"]
ggplot(adv, aes(tv, sales)) + geom_point() +
geom_smooth(method = "lm") + labs(subtitle = glue(
        "B0: {round(int, digits = 3)}\n",
        "B1: {round(b1, digits=3)}"
    )) + theme_minimal()
```



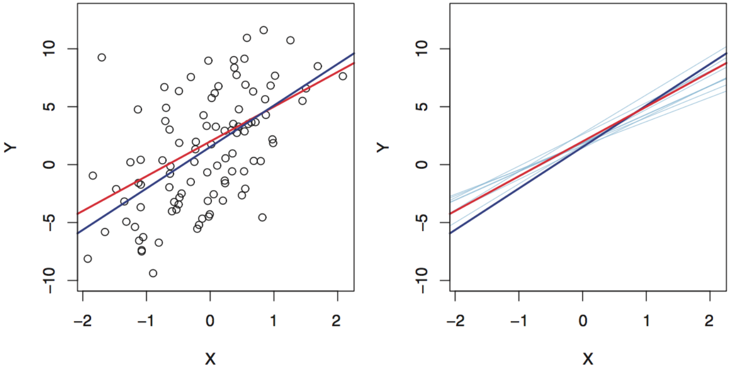
Assessing the Coefficient Estimation Accuracy



We only have one data set, and so what does it mean that two different lines describe the relationship between the predictor and the response?

source: ISLR, p.64.

Assessing the Coefficient Estimation Accuracy



source: ISLR, p.64.

- Data Generated: $f(X) = 2 + 3X + \epsilon$
- Population regression line (red): $f(X) = 2 + 3X$
- Least Squares regression line (blue)
- Unbiased estimation

Assessing the Coefficient Estimation Accuracy: Standard Error

$$SE(\hat{\beta}_0)^2 = \sigma^2 \left(\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \right)$$

$$SE(\hat{\beta}_1)^2 = \frac{\sigma^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$\sigma^2 = Var(\epsilon)$$

$$\hat{\sigma} = RSE = \sqrt{RSS/(n-2)}$$

Assessing the Coefficient Estimation Accuracy: Confidence Intervals

- A 95% confidence interval is defined as a range of values such that with 95% probability, the range will contain the true unknown value of the parameter
- For linear regression, the 95% confidence interval for β_1 approximately takes the form

$$\hat{\beta}_1 \pm 2 \cdot SE(\hat{\beta}_1)$$

Assessing the Coefficient Estimation Accuracy: Hypothesis test

$$H_0 : \hat{\beta}_1 = 0$$

$$H_a : \hat{\beta}_1 \neq 0$$

$$t = \frac{\hat{\beta}_1 - 0}{SE(\hat{\beta}_1)}$$

- t measures the number of standard deviations that β_1 is away from 0
- the p value tells you how likely it is to observe such t value given $\hat{\beta}_1 = 0$

Assessing the Coefficient Estimation Accuracy: Hypothesis test

	Coefficient	Std. error	t-statistic	p-value
Intercept	7.0325	0.4578	15.36	< 0.0001
TV	0.0475	0.0027	17.67	< 0.0001

source: ISLR, p.68.

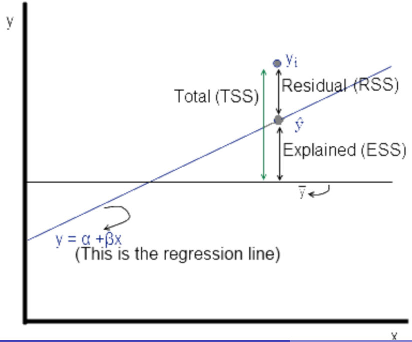
Assessing the Accuracy of the Model: Residual Standard Error

- RSE (Residual Standard Error)
- Roughly speaking, it is the average amount that the response will deviate from the true regression line
- Sales in each market deviate from the true regression line by approximately 3,260 units, on average
- The RSE is considered a measure of the lack of fit of the model to the data

Assessing the Accuracy of the Model: R^2

$$R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS}$$

- where $TSS = \sum(y_i - \bar{y})^2$ is the *total sum of squares*
- R^2 measures the proportion of variability in Y that can be explained using X



R Syntax: Multiple Linear Regression

- example: median Boston house prices

```
?MASS::Boston
boston <- MASS::Boston
names(boston)
mlm <- lm(medv ~ lstat, data = boston)
mlm <- lm(medv ~ lstat + age, data = boston)
summary(mlm)
mlm <- lm(medv ~ ., data = boston)
mlm <- lm(medv ~ . - indus -age, data = boston)
cor(boston)
mlm <- lm(medv ~ . + zn*chas - indus -age, data = boston)
mlm <- lm(medv ~ . + I(lstat^2) - indus -age, data = boston)
```

R Syntax: Multiple Linear Regression

```r
car <- ISLR::Carseats
summary(lm(Sales ~ ShelveLoc, data = car))
contrasts(car$ShelveLoc)
```

R Syntax: Multiple Linear Regression

```r
mlm <- lm(medv ~ . - indus -age, data = boston)
pred <- predict(mlm)
```

Section 3

Binary Classification

Section 4

Hands on Exercises

R Commands

Great resources

- Casuality: http://nickchk.com/causalgraphs.html

Reproducing a graph from the book

```
ls_lines <- map(1:20, ~{
    adv_s <- adv[sample(.N, 10)]
    stat_smooth(data = adv_s, mapping = aes(tv, sales), methoc
        se = FALSE, alpha = .3, geom = 'line', color = "blue")
})

ggplot(adv, aes(tv, sales)) +
    geom_point(alpha = .5) +
    geom_smooth(method = lm, color = "red", se = FALSE) +
    ls_lines
```