FOUNDATIONS OF STATISTICAL ANALYSIS & MACHINE LEARNING

Amric Trudel
amric.trudel@epita.fr

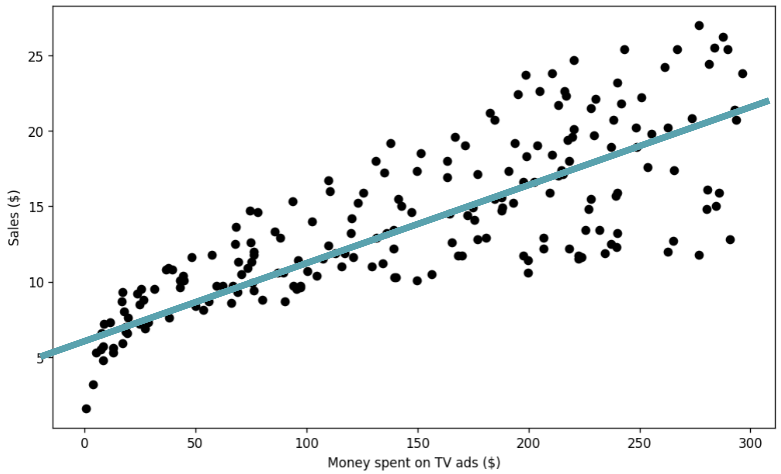
COURSE PROGRAM

Structure

PREPARATION	Data exploration
	Data preprocessing
REGRESSION	Linear regression with one variable
	Multiple and polynomial regression
CLASSIFICATION	Logistic regression
	Classification model assessment
	k-NN, Decision Tree, SVM
CLUSTERING	k-means, hierarchical clustering
DIMENSIONALITY REDUCTION	Principal Components Analysis
ALL NOTIONS	Final assignment

REVIEW OF LAST CLASS

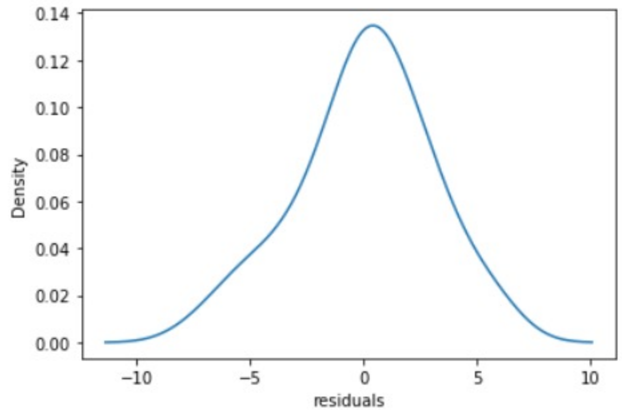
Simple linear regression



- Equation with one feature:

$$Y = \beta_0 + \beta_1 X + e$$

Intercept   Slope   Residual (error term)



MULTIPLE LINEAR REGRESSION

Using more than one feature

Size (feet²)	Number of bedrooms	Number of floors	Age of home (years)	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…

MULTIPLE LINEAR REGRESSION

Model fitting

- We consider n distinct predictors: $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n + e$

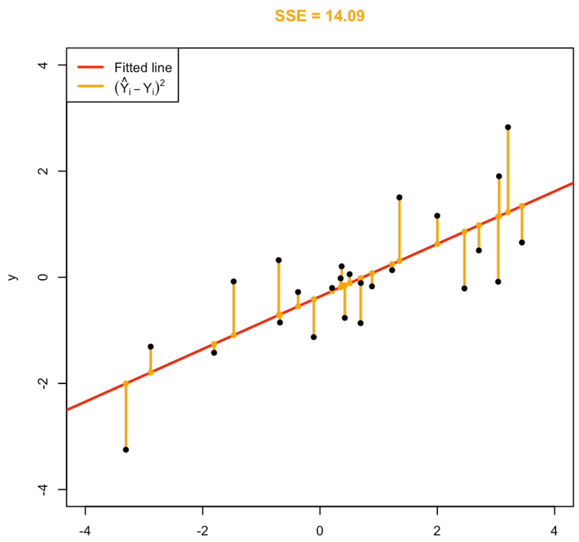
- In matrix terms: $\mathbf{Y} = \mathbf{X}\beta + e$

$$\mathbf{Y} = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} \quad \mathbf{X} = \begin{pmatrix} 1 & x_{11} & x_{12} & \dots & x_{1p} \\ 1 & x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix} \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} \quad e = \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{pmatrix}$$

MULTIPLE LINEAR REGRESSION

Cost function

- The same <u>cost functions</u> as before can be used on a multiple linear regression model (as with **any** regression model)
  - Residual Sum of Squares: $RSS = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$



Mean Squared Error	Root mean squared error	Mean absolute error
$MSE = \frac{1}{m} RSS$	$RMSE = \sqrt{MSE}$	$MAE = \frac{1}{m} \sum_{i=1}^{m} \lvert y_i - \hat{y}_i \rvert$

MULTIPLE LINEAR REGRESSION

Model fitting: analytical solving

- We choose $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$ to minimize the residual sum of squares:

$$\mathrm{RSS} = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \hat{\beta}_2 x_{i2} - \cdots - \hat{\beta}_p x_{ip}\right)^2$$

- Linear combination of $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$ ➔ Least Squares (analytical solving):

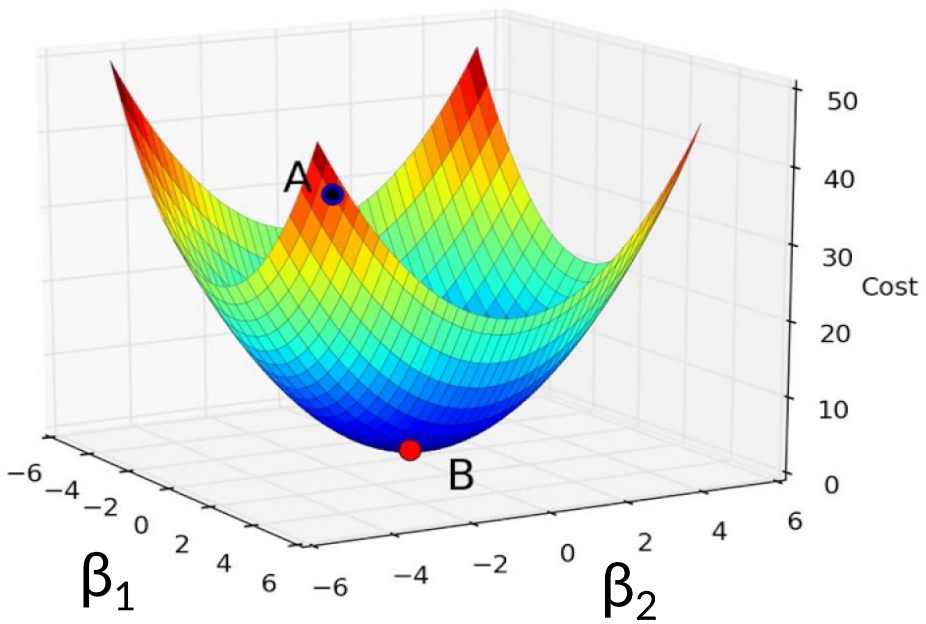
$$\hat{\beta} = \mathbf{X}^{+}\mathbf{Y} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

Pseudoinverse

MULTIPLE LINEAR REGRESSION

Model fitting: numerical solving

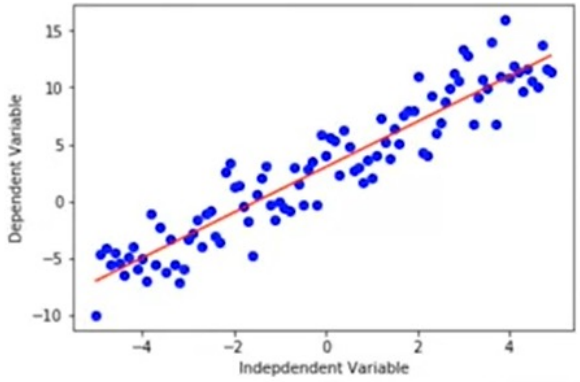
- Gradient descent works with a lot of parameters ($\beta_{0,}\ \beta_1 \ldots \beta_n$)

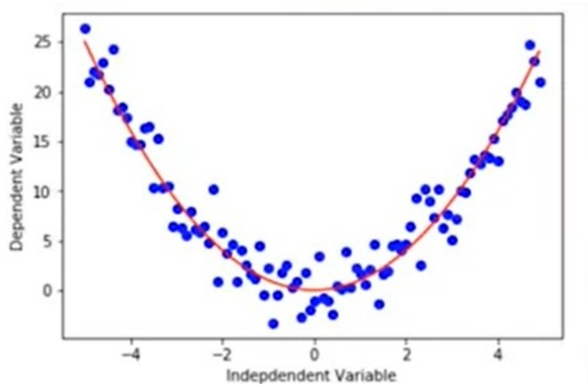


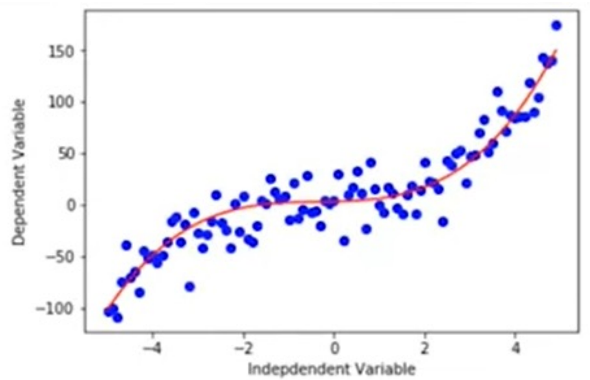
MULTIPLE LINEAR REGRESSION

Polynomial regression

- Some curvy data can be modeled by a polynomial regression:







- It can be transformed into a linear regression model. E.g.:

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + e$$

$$\Rightarrow Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + e \quad \text{with } X_1 = X, X_2 = X^2, X_3 = X^3$$

MULTIPLE LINEAR REGRESSION

Linear model extensions

- Categorical variables. E.g.:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + e \quad \text{with } X_1 = \begin{cases} 0 \text{ if male} \\ 1 \text{ if female} \end{cases}$$

- Interaction terms. E.g.:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + e$$

MULTIPLE LINEAR REGRESSION

Python implementation

- Creating the polynomial features:

```
from sklearn.preprocessing import PolynomialFeatures
poly = PolynomialFeatures(degree=2)
X_poly = poly.fit_transform(X)
```

- Training the linear regression model:

```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
```

- Using the model for predicting:

```
y_pred = regressor.predict(X_test)
```

$$\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \longrightarrow \begin{bmatrix} [1 & v_1 & v_1^2] \\ [1 & v_2 & v_2^2] \\ \vdots & \vdots & \vdots \\ [1 & v_n & v_n^2] \end{bmatrix}$$

$$\begin{bmatrix} 2. \\ 2.4 \\ 1.5 \\ \vdots \end{bmatrix} \longrightarrow \begin{bmatrix} [1 & 2. & 4.] \\ [1 & 2.4 & 5.76] \\ [1 & 1.5 & 2.25] \\ \vdots & \vdots & \vdots \end{bmatrix}$$

Remember to **scale** your polynomial features before fitting the model.

A **pipeline** can be useful!

REGRESSIONS

Model choice

We are left with many choices when building a model.

- Linear regression (one single predictor, or more)
- Polynomial regression
- Non-linear regression model
- Data transformation

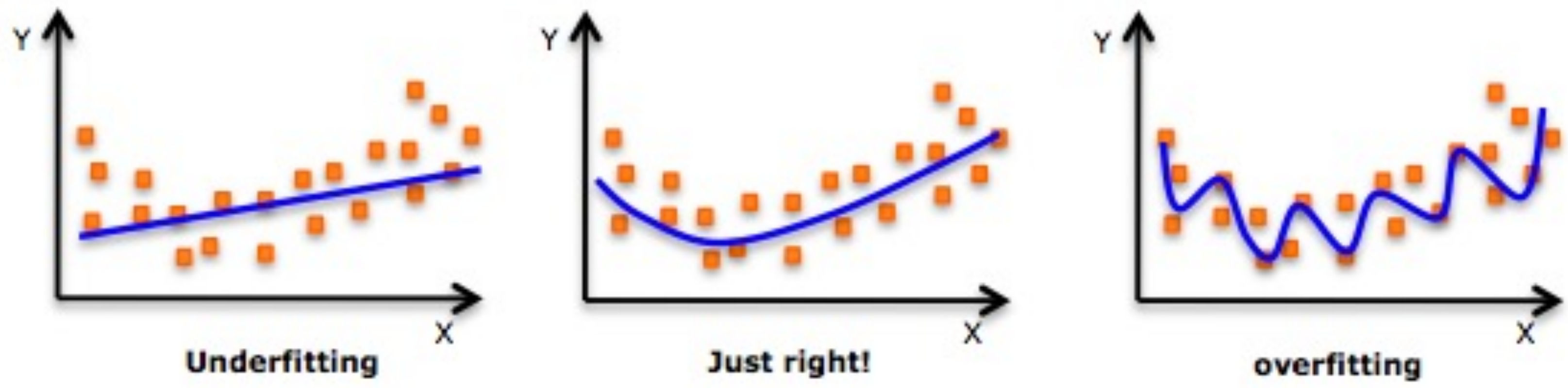
Model **complexity**

MULTIPLE LINEAR REGRESSION

The overfitting problem

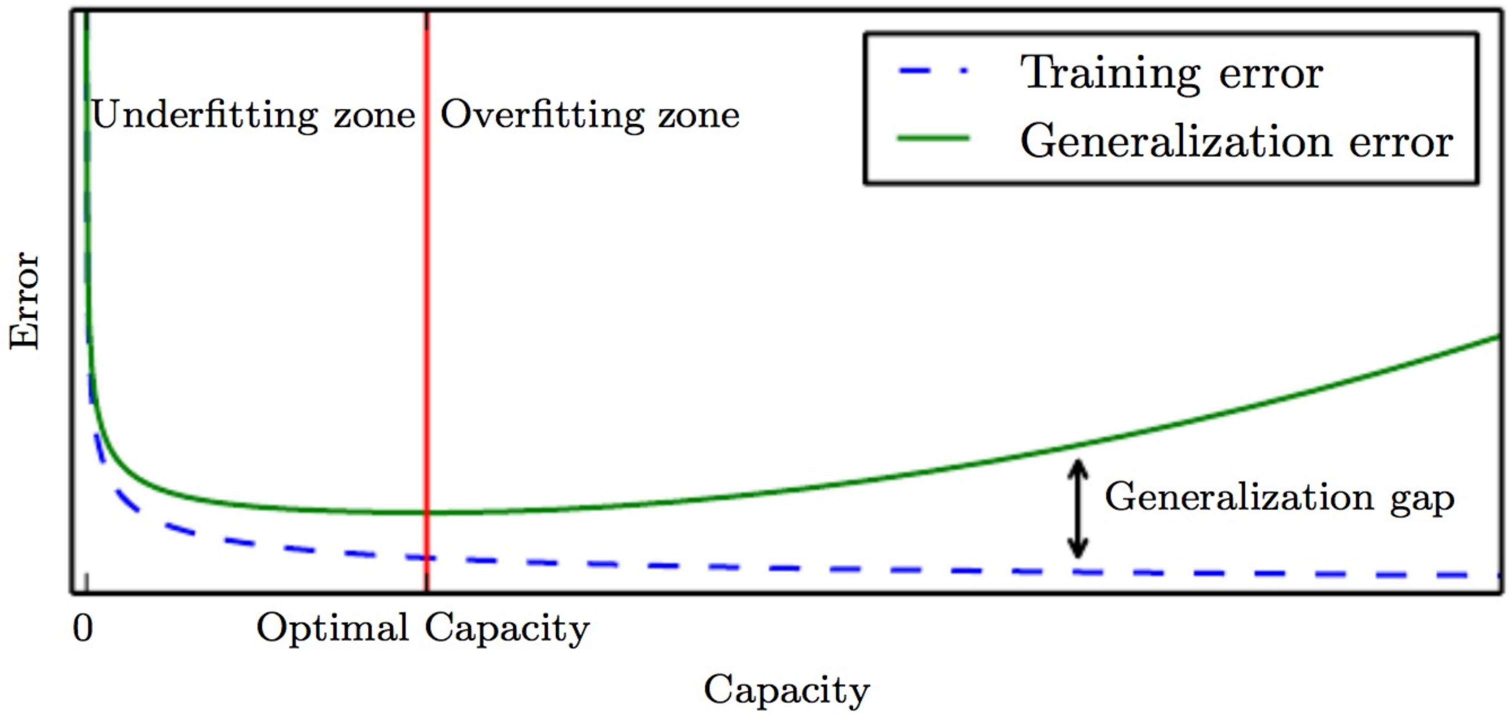
- To what extent should we increase the complexity of the model?
  - Up until you reach overfitting!

  - Overfitting is when the model too complex and overly fitted to the particularities of the dataset, which may result in capturing noise and producing a non-generalized model.



MULTIPLE LINEAR REGRESSION

The overfitting problem



MULTIPLE LINEAR REGRESSION

Controlling overfitting: Hold out validation

- It is common to use a hold out **validation set** on which to evaluate the generalization errors of different models when choosing the best one.

Training set	Validation set	Test set
60 %	20 %	20 %
Train several models with different complexities (minimize trainin error)	Measure generalization error on each option	Measure generalization error final model

MULTIPLE LINEAR REGRESSION

Avoiding overfitting

- One way to reduce overfitting is to simplify the model by reducing the number of features or using a simpler hypothesis.

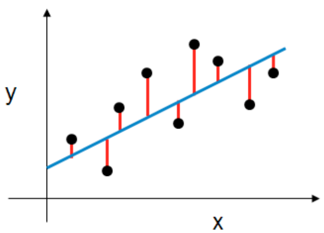
- Another method that can be used to reduce overfitting and avoid much of the trial and error is called **regularization**.
  Its strategy is to penalize the model for having large parameters.

MULTIPLE LINEAR REGRESSION

Avoiding overfitting: Regularization

- When a linear regression model overfits a dataset, its parameters usually become very large ($|\beta_n| >> 1$).
- We can penalize large parameters by modifying the **cost function.**

$$Cost \quad = \quad RSS \quad + \quad Regularization\ term$$



+

$$Cost \quad = \quad \sum_{i=1}^{m}(y_i - \hat{y}_i)^2 \quad + \quad \alpha \sum_{j=1}^{n}(\beta_n)^2$$

MULTIPLE LINEAR REGRESSION

Python implementation of regularized regression

- The regularization technique we learned about is called Ridge regression:

```
from sklearn.linear_model import Ridge

regularized_regressor = Ridge(alpha=1.0)
regularized_regressor.fit(X_train, y_train)
```

- A well-tuned Ridge regression model will most probably have a lower generalization error than a non-regularized linear regression model that overfits.

MULTIPLE LINEAR REGRESSION

Example of implementation

- Data set: product sales w.r.t. ads expenditures (again)
- Objectives:
  - Train a multiple linear regression
  - Train a polynomial regression
  - Compare the performance
  - Try out Ridge regularization

TV	radio	newspaper	sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	9.3
151.5	41.3	58.5	18.5
180.8	10.8	58.4	12.9
8.7	48.9	75	7.2
57.5	32.8	23.5	11.8
120.2	19.6	11.6	13.2
8.6	2.1	1	4.8

MULTIPLE LINEAR REGRESSION

Student practice

- Data set: CO_2 emission w.r.t. vehicle characteristics (again)
- Objectives:
  - Check for possible correlations
  - Train multiple linear regressions
  - Assess and compare the performance of the regressions

	YEAR	MAKE	MODEL	VEHICLECLASS	ENGINESIZE	CYLINDERS	MISSION	FUELTYPE	ON_CITY	_HWY	OMB	MPG	CO2EMISSIONS
1	2014	ACURA	ILX	COMPACT	2	4	AS5	Z	9.9	6.7	8.5	33	196
2	2014	ACURA	ILX	COMPACT	2.4	4	M6	Z	11.2	7.7	9.6	29	221
3	2014	ACURA	ILX HYBRID	COMPACT	1.5	4	AV7	Z	6	5.8	5.9	48	136
4	2014	ACURA	MDX 4WD	SUV - SMALL	3.5	6	AS6	Z	12.7	9.1	11.1	25	255
5	2014	ACURA	RDX AWD	SUV - SMALL	3.5	6	AS6	Z	12.1	8.7	10.6	27	244
6	2014	ACURA	RLX	MID-SIZE	3.5	6	AS6	Z	11.9	7.7	10	28	230
7	2014	ACURA	TL	MID-SIZE	3.5	6	AS6	Z	11.8	8.1	10.1	28	232
8	2014	ACURA	TL AWD	MID-SIZE	3.7	6	AS6	Z	12.8	9	11.1	25	255
9	2014	ACURA	TL AWD	MID-SIZE	3.7	6	M6	Z	13.4	9.5	11.6	24	267
10	2014	ACURA	TSX	COMPACT	2.4	4	AS5	Z	10.6	7.5	9.2	31	212
11	2014	ACURA	TSX	COMPACT	2.4	4	M6	Z	11.2	8.1	9.8	29	225
12	2014	ACURA	TSX	COMPACT	3.5	6	AS5	Z	12.1	8.3	10.4	27	239
13	2014	ASTON MARTIN	DB9	MINICOMPACT	5.9	12	A6	Z	18	12.6	15.6	18	359
14	2014	ASTON MARTIN	RAPIDE	SUBCOMPACT	5.9	12	A6	Z	18	12.6	15.6	18	359
15	2014	ASTON MARTIN	V8 VANTAGE	TWO-SEATER	4.7	8	AM7	Z	17.4	11.3	14.7	19	338
16	2014	ASTON MARTIN	V8 VANTAGE	TWO-SEATER	4.7	8	M6	Z	18.1	12.2	15.4	18	354
17	2014	ASTON MARTIN	V8 VANTAGE S	TWO-SEATER	4.7	8	AM7	Z	17.4	11.3	14.7	19	338