FOUNDATIONS OF STATISTICAL ANALYSIS & MACHINE LEARNING

Amric Trudel
amric.trudel@epita.fr

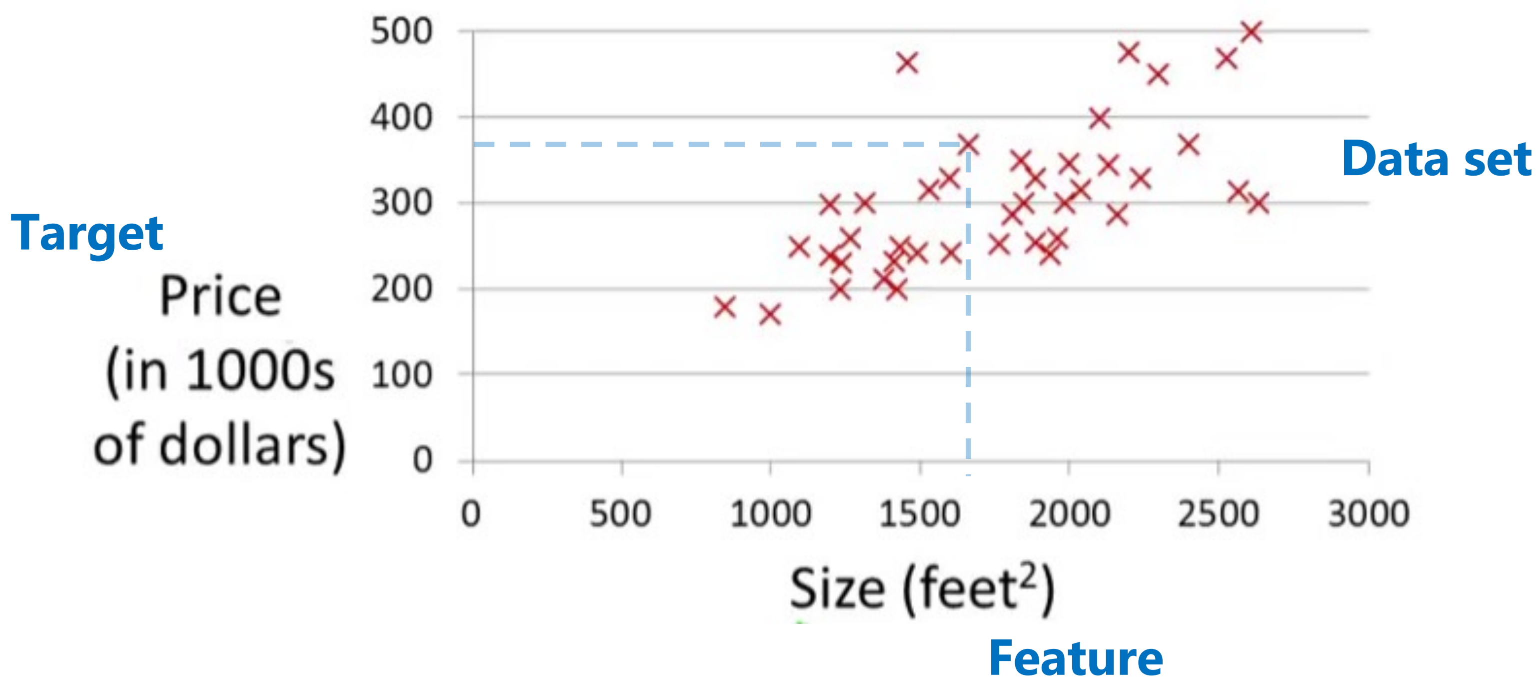
COURSE PROGRAM

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

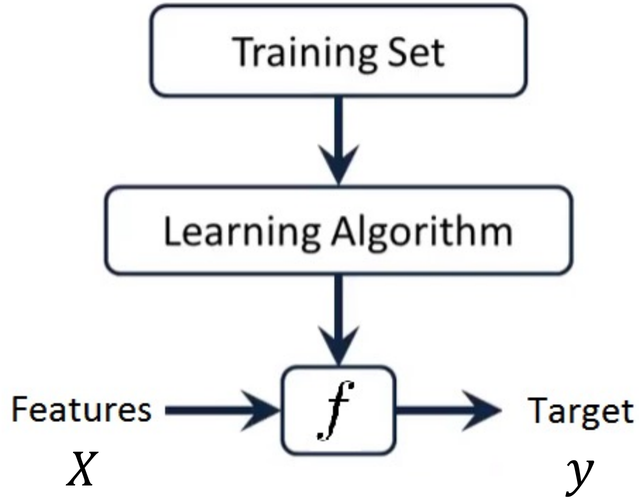
GENERAL APPROACH FOR ML

Problem statement



GENERAL APPROACH FOR ML

Problem statement



- Why estimate f?
  - **Prediction**: provide a response estimation for any feature values X: $(\hat{y} = \hat{f}(X))$.
  - **Inference**: understand the relationship between X and y, i.e. how Y changes as a function of X_1, X_2, …, X_p.

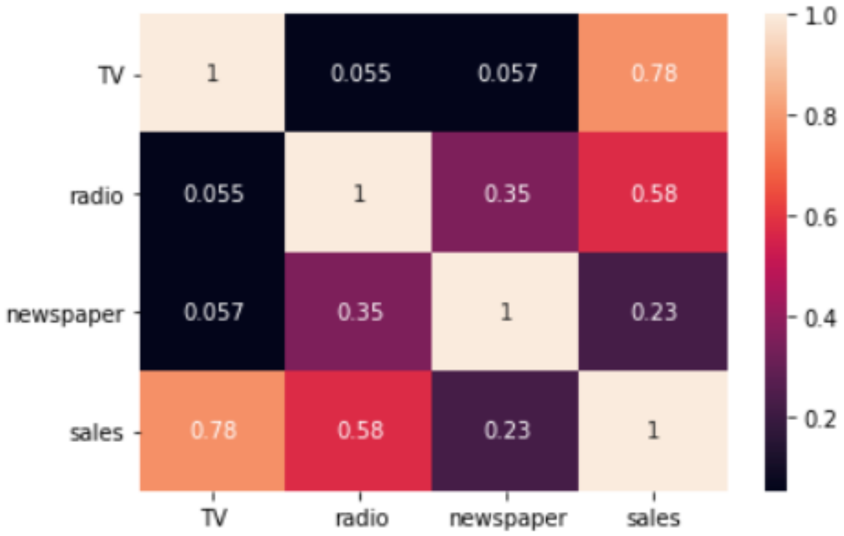
GENERAL APPROACH FOR ML

Solving process

1. Get some intuition from **data inspection** (visualization, correlation, etc.)
2. **Choose a model**
3. **Find the parameters** that minimize a criterion (cost function)
4. **Evaluate the performance**

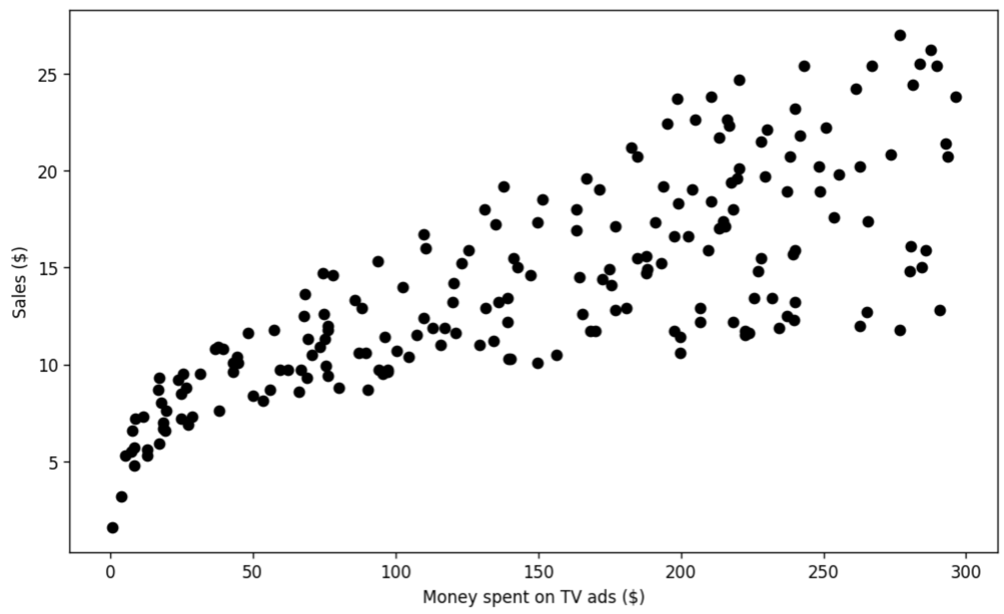
REGRESSIONS

Data inspection



Correlation matrix

Target
Continuous values

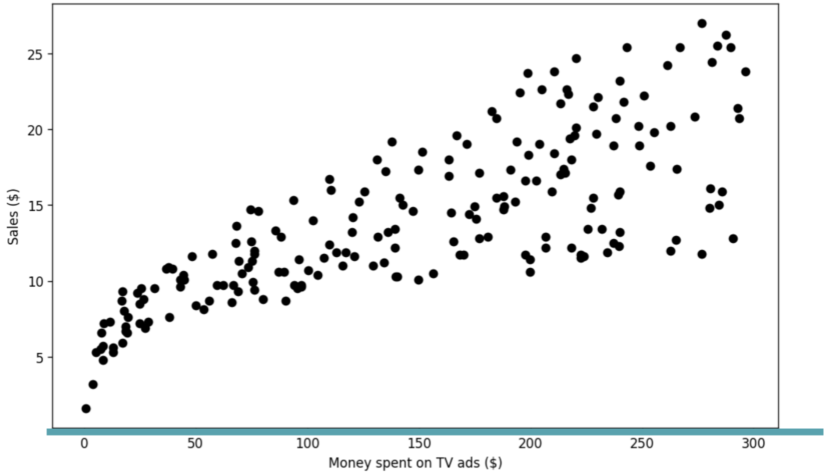


Feature

SIMPLE LINEAR REGRESSION

Model definition

- Inspection:



What other assumptions does this model rely on?

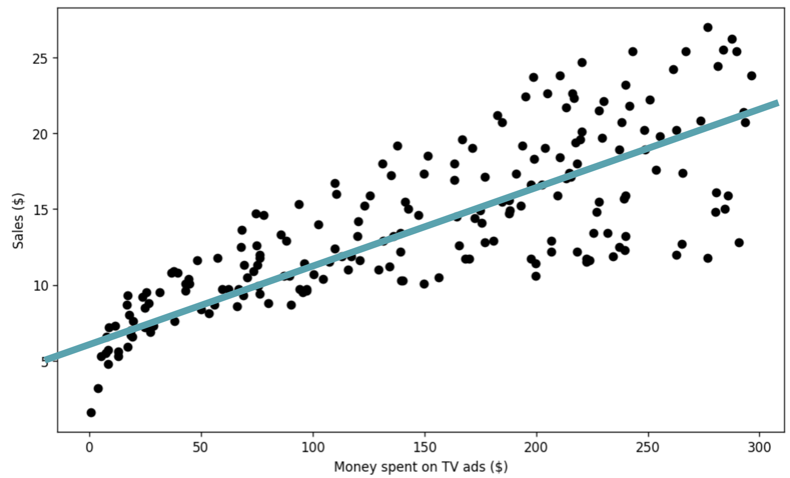
- Assumption of a linear relationship between the response Y and a single predictor variable X:

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

Intercept Slope Residual (error term)

SIMPLE LINEAR REGRESSION

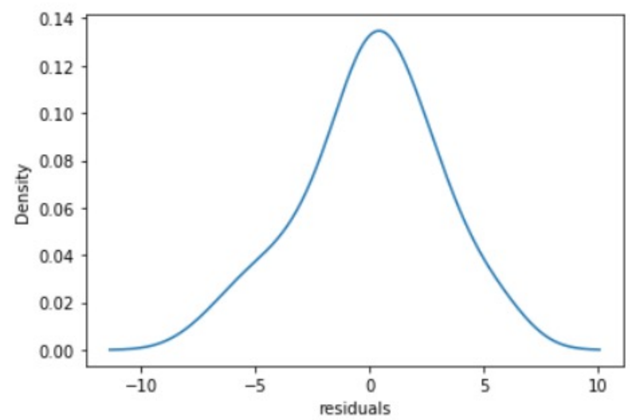
Model definition



- The residuals should have a mean of 0

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

Intercept Slope Residual (error term)



SIMPLE LINEAR REGRESSION

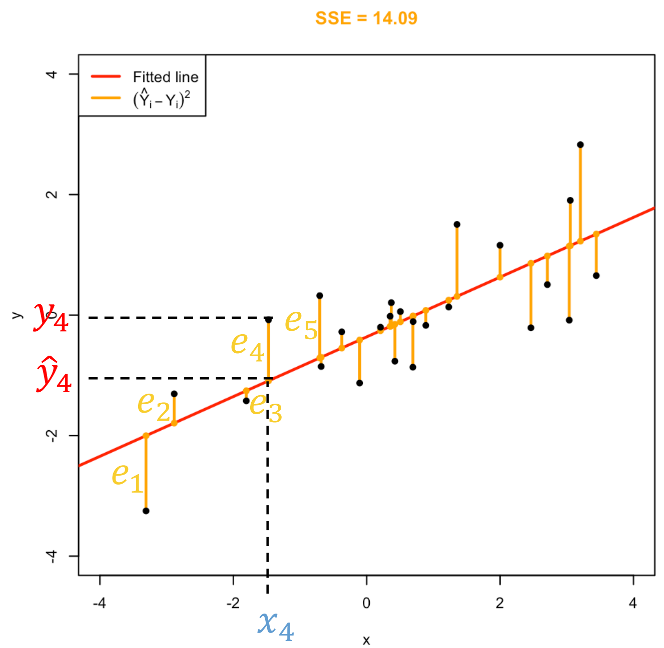
Cost function

- Cost function: Residual Sum of Square (RSS or SSE)

$$\text{RSS} = \sum_{i=1}^{m} e_i^2$$

with e_i the i^{th} residual: $e_i = y_i - \hat{y}_i$

$$\text{RSS} = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$$



REGRESSIONS

Model parameter optimization

- **Analytical solving** through Ordinary Least Squares (linear algebra operations)
  - More common in statistics
  - Works with small samples

- **Numerical solving** through an optimization algorithm, e.g. *Gradient Descent*
  - More common in Machine Learning
  - Works with large datasets

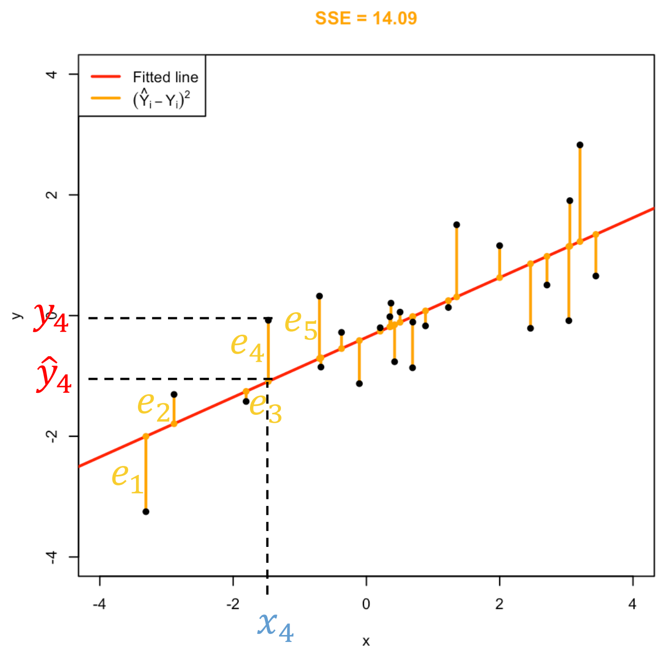
SIMPLE LINEAR REGRESSION

Model parameter optimization

- **Analytical solving:** Minimization through Least Squares approach:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \frac{s_{xy}}{s_x^2}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

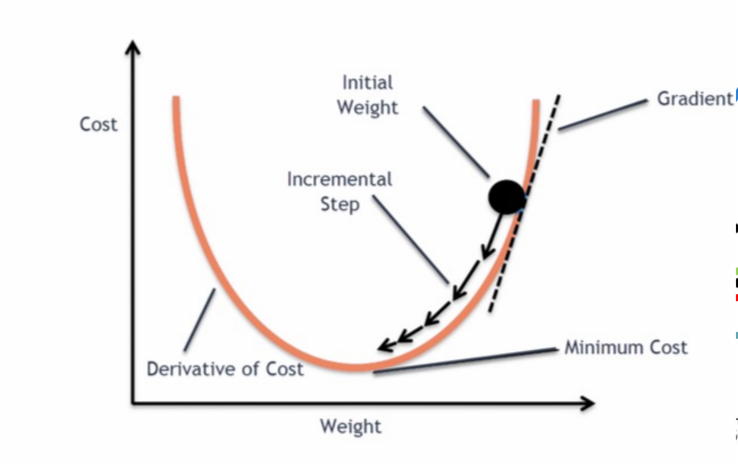


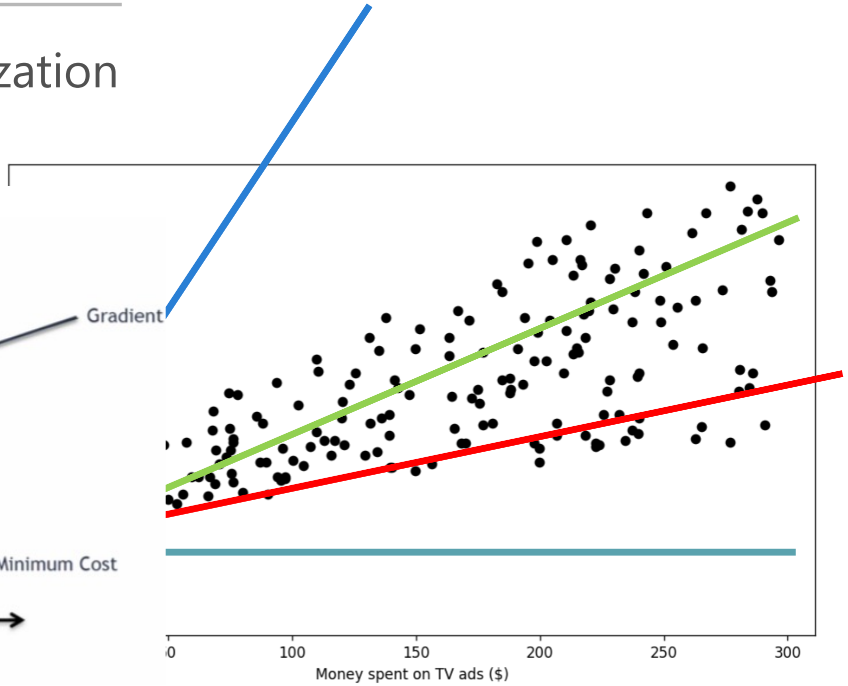
- Sample mean: $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$
- Sample variance: $s_x^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$
- Sample covariance: $s_{xy} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$

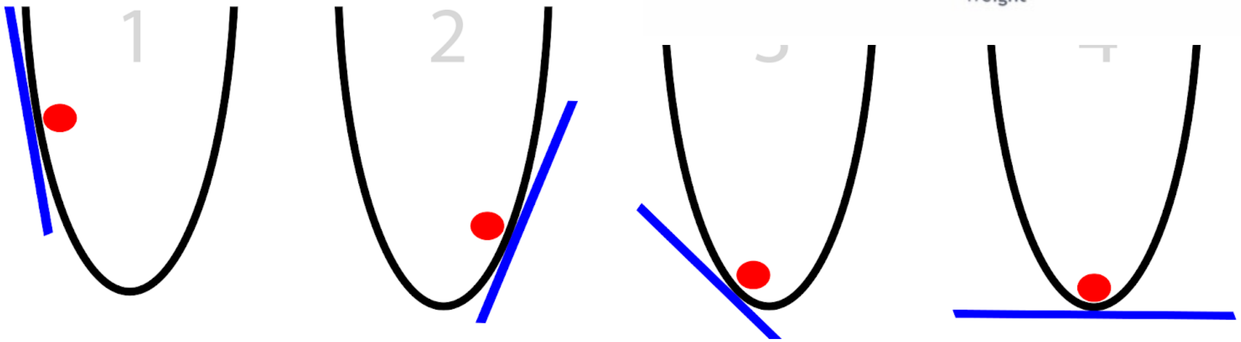
SIMPLE LINEAR REGRESSION

Model parameter optimization

- **Numerical solving:**
  - Iteratively adjusting ea
    e.g. $\hat{\beta}_1$

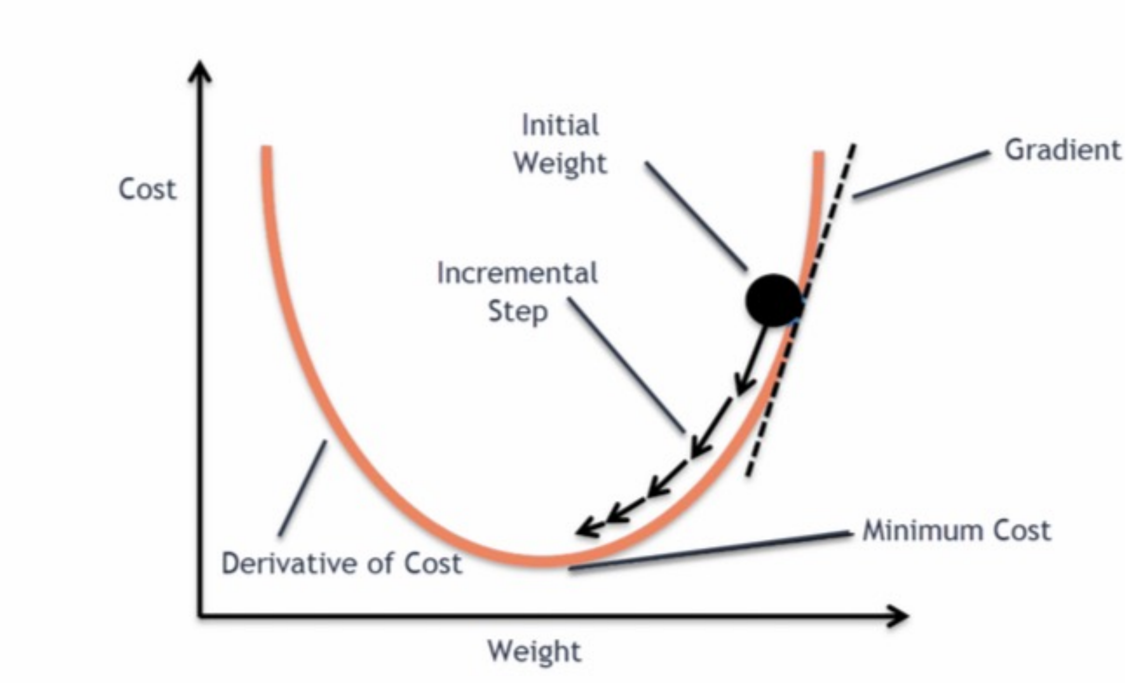






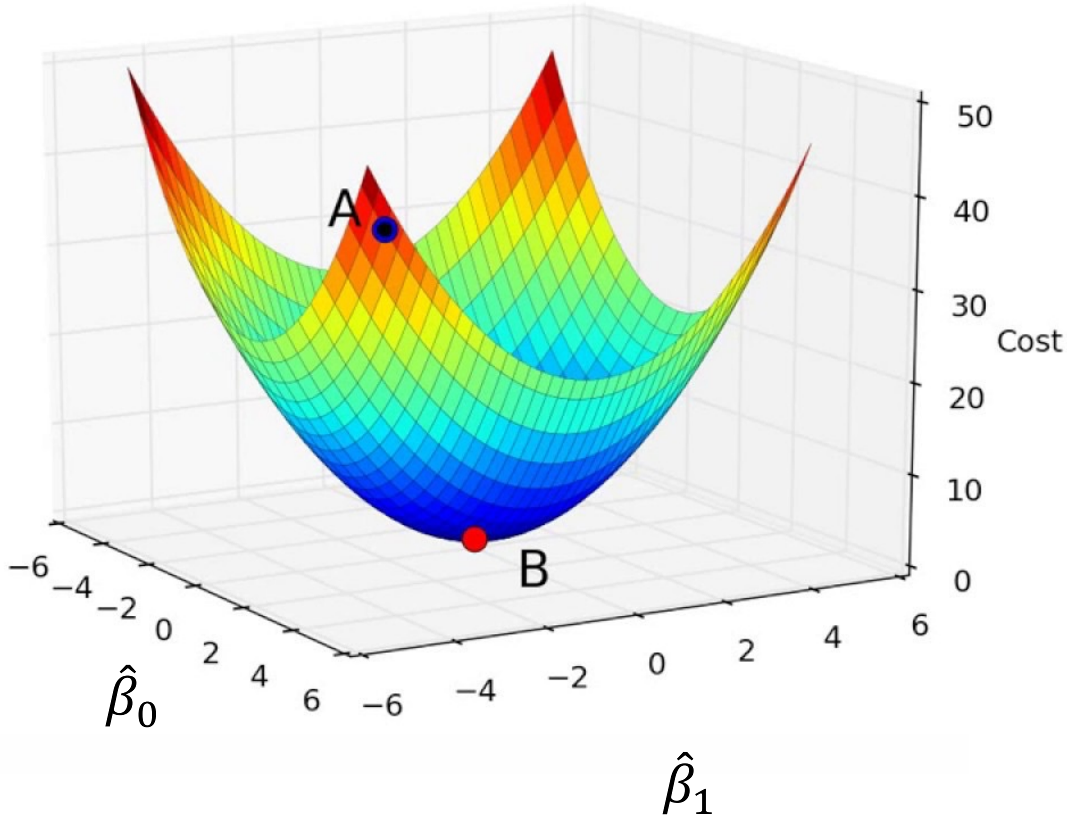
SIMPLE LINEAR REGRESSION

Gradient Descent



$\hat{\beta}_1$

SIMPLE LINEAR REGRESSION

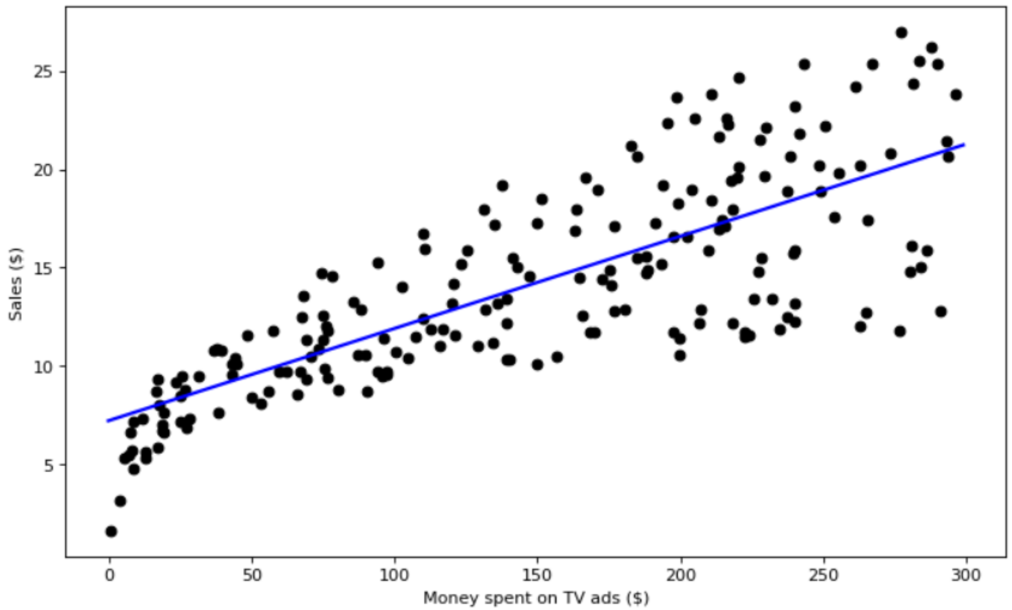


SIMPLE LINEAR REGRESSION

Model fitting

- Prediction:

$$\hat{y} = 7.22 + 0.047x$$



REGRESSIONS

Performance evaluation

- MSE, RMSE, MAE, …
- R^2 score

SIMPLE LINEAR REGRESSION

Model performance assessment

- Accuracy of the prediction:
  - Sum of Squared Errors or Residual Sum of Squares: $\mathrm{RSS} = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$
  - Mean Squared Error: $\mathrm{MSE} = \frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 = \frac{1}{m} RSS$
  - Root Mean Squared Error: $\mathrm{RMSE} = \sqrt{\frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2}$
  - Mean Absolute Error: $\mathrm{MAE} = \frac{1}{m} \sum_{i=1}^{m} |y_i - \hat{y}_i|$

SIMPLE LINEAR REGRESSION

Python implementation

- Training the linear regression model:

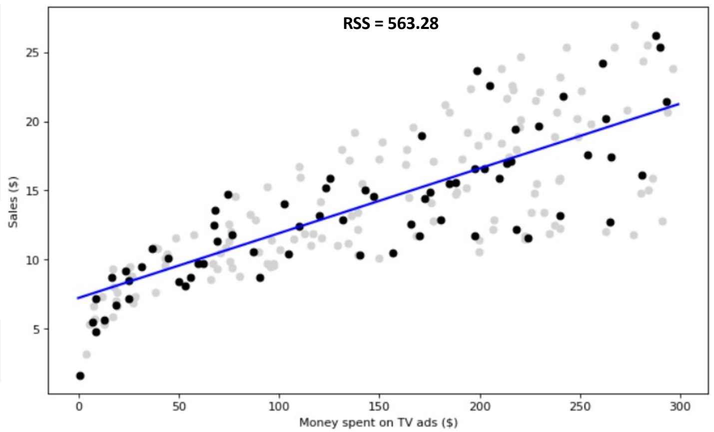
```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
```

- Using the model for predicting:

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

- Assessing the accuracy:

```
MSE = np.mean((y_pred - y_test) ** 2)
RMSE = np.sqrt(np.mean((y_pred - y_test) ** 2)))
MAE = np.mean(abs(y_pred - y_test))
```



REGRESSION MODEL ASSESSMENT

Variance decomposition

- The variance of Y can be decomposed into a part corresponding to the regression and a part corresponding to the error: $\text{TSS} = \text{ESS} + \text{RSS}$

$$\text{TSS} = \sum_{i=1}^{n} (Y_i - \bar{Y})^2$$

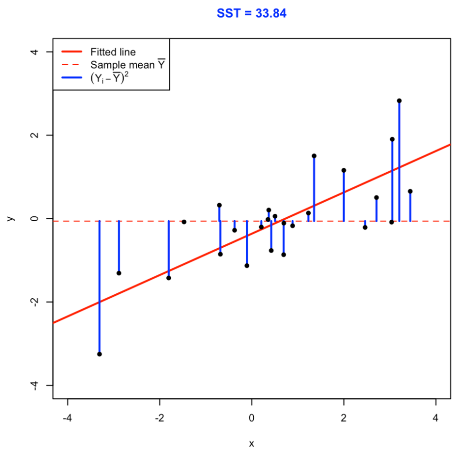
$$\text{ESS} = \sum_{i=1}^{n} \left(\hat{Y}_i - \bar{Y}\right)^2$$

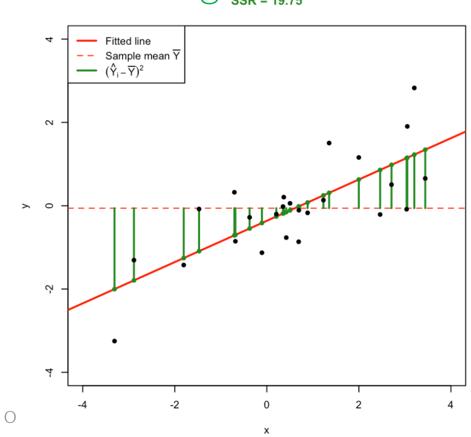
$$\text{RSS} = \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2$$

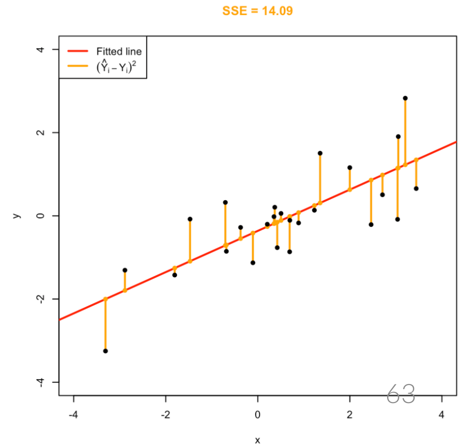
Total Sum of Squares.
This is the total variation of $Y_1 \dots Y_n$.

Explained Sum of Squares.
This is the variation explained by the regression line.

Residual Sum of Squares.
This is the unexplained variation.







REGRESSION MODEL ASSESSMENT

ANOVA

- The **coefficient of determination R^2** measures the proportion of variability in Y that can be explained using X:

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

  - When close to 1: a large proportion of the variability of Y has been explained by the regression model.
  - When close to 0: the regression did not explain much of the variability in the response. This can occur when the linear model is wrong, when the inherent error is high, or when there is no linear relationship between X and Y.

REGRESSION MODEL ASSESSMENT

Python implementation

- Computing the coefficient of determination:

```
from sklearn.metrics import r2_score

print("R2-score: %.2f" % r2_score(y_test , y_pred))
```

- More metrics available with scikit-learn.

SIMPLE LINEAR REGRESSION

Example of implementation

- Data set: product sales w.r.t. ads expenditures
- Objectives:
  - Inspect correlation between candidate features and the target
  - Train a simple linear regression with one feature
  - Evaluate the model accuracy

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
199.8	2.6	21.2	10.6
66.1	5.8	24.2	8.6
214.7	24	4	17.4

SIMPLE LINEAR REGRESSION

Practice

- Data set: CO_2 emission w.r.t. vehicle characteristics
- Objectives:
  - Check for possible correlations
  - Train simple linear regressions with one feature
  - Assess and compare the accuracy of the regressions

	YEAR	MAKE	MODEL	VEHICLECLASS	ENGINESIZE	CYLINDERS	:MISSION	FUELTYPE	ON_CITY	I_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