

# Introduction to Statistical Machine Learning

Réda DEHAK  
<http://ismml.dehak.org>

1

---

---



---

---

---

---

---

## Organisation

- 8 lessons : Introduction to Statistical Machine Learning Methods
  - 8 Lectures :
    - Regression - Gradient descent - Régularization Lasso and Ridge
    - Linear Classification - Logistic Regression - MultiClass Classification
    - Neural Network Methods: Backpropagation Algorithm
    - Support Vector Machine : Linear version
    - Support Vector Machine : Kernel Method
    - Decision Trees, Bagging, Boosting, Random Forest
    - Dimensionality Reduction
    - Unsupervised Learning
  - 8 labs :
    - Python 3
    - Jupiter
    - Numpy
    - Scikit-learn
    - ...
- MidTerm Exam : ? Final Exam : ?

Réda DEHAK 2

2

---

---



---

---

---

---

---

## Why Learn?

- Machine learning is programming computers to **optimize a performance criterion** using **example data or past experience**.
- There is no need to "learn" to calculate payroll, or to sort a list of numbers
- Learning is used when:
  - Human expertise does not exist (navigating on Mars),
  - Humans are unable to explain their expertise (speech recognition)
  - Solution changes in time (routing on a computer network)
  - Solution needs to be adapted to particular cases (user biometrics)

Réda DEHAK 3

3

---

---



---

---

---

---

---

## Machine Learning

- Study of algorithms that [T.Mitchell]:
  - Improve their **performance  $P$** ,
  - At some **task  $T$**
  - With **experience  $E$**
- Well defined learning task:  $\langle P, T, E \rangle$

Réda DENAE

4

4

---

---

---



---

---

---

---

---

## What is Machine Learning?

- Optimize a performance criterion using example data or past experience.
- Role of Statistics: Inference from a sample
- Role of Computer science: Efficient algorithms to
  - Solve the optimization problem
  - Representing and evaluating the model for inference

Réda DENAE

5

5

---

---

---

---

---

---

---

---




## What We Talk About When We Talk About "Learning"

- Learning general models from a data of particular examples
- Data is cheap and abundant (data warehouses, data marts); knowledge is expensive and scarce.
- Example in retail: Customer transactions to consumer behavior:  
*People who bought "Beer" also bought "Chips"*
- Build a model that is *a good and useful approximation* to the data.

Réda DENAE

6

6

---

---

---

---

---

---

---

---





## Learning to Detect Faces in Images




Training images for different pose

Réda DENAE
7

---

---

---

---


---


---

---

---


7





## Text Documents Classifications

Personal home page  
vs.  
Company home page  
vs.  
University home page  
vs.  
...



Réda DENAE
8

---

---

---

---


---


---

---

---

8





## Data Mining

- **Retail:** Market basket analysis, Customer relationship management (CRM)
- **Finance:** Credit scoring, fraud detection
- **Manufacturing:** Control, robotics, troubleshooting
- **Medicine:** Medical diagnosis
- **Telecommunications:** Spam filters, intrusion detection
- **Bioinformatics:** Motifs, alignment
- **Web mining:** Search engines
- ...

Réda DENAE
9

---

---

---

---

---

---

---

---

9

**Machine Learning in Computer Science**

- Machine learning already the preferred approach to
  - Speech recognition, Natural Language processing
  - Computer Vision
  - Medical outcomes analysis
  - Robot control
  - ...
- This ML niche is growing (why?)

Réda DEHAÏE 10

10

---

---

---

---

---

---

---

---

**Machine Learning in Computer Science**

- Machine learning already the preferred approach to
  - Speech recognition, Natural Language processing
  - Computer Vision
  - Medical outcomes analysis
  - Robot control
  - ...
- This ML niche is growing
  - Improved machine learning algorithms
  - Increased data capture, networking, new sensors
  - Software too complex to write by hand
  - Demand for self-customization to user, environment

Réda DEHAÏE 11

11

---

---

---

---

---

---

---

---

**Machine Learning**

Réda DEHAÏE 12

12

---

---

---


---

---

---

---

---



## Machine Learning Methods

1. Supervised Methods:
  - a) Regression
  - b) Classification
2. Unsupervised Methods
3. Reinforcement Learning

Réda DENAE 13

13

---

---


---

---

---

---

---



## Supervised Learning

- learning a **function** that **maps** an **input** (features) to an **output** (target or labels) based on **examples input-output pairs**.
- Discrete Output Values → Classification
- Continuous Output Values → Regression
- Examples:
  - Face recognition, Character recognition, Speech and Language recognition
  - Predicting age, nationality and weight of a person, Predicting whether stock price of a company will increase tomorrow

Réda DENAE 14

14

---

---


---

---

---

---

---



## Unsupervised Learning

- Learning “what normally happens”
- No output
- Clustering: Grouping similar instances
- Example applications
  - Customer segmentation in CRM
  - Image compression: Color quantization
  - Bioinformatics: Learning motifs

Réda DENAE 15

15

---

---


---

---

---

---

---



## Reinforcement Learning

- Learning a policy: A *sequence* of outputs
- No supervised output but delayed reward
- Examples:
  - Credit assignment problem
  - Game playing
  - Robot in a maze
  - Multiple agents, partial observability, ...

Réda DENAE 16

16

---

---


---

---

---

---

---



## The four “aspects” of Machine Learning

- **Representation:** How best to represent data for best processing
- **Modeling:** How to *model* the systematic and statistical characteristics of the data
- **Classification:** How do we assign a class to the data?
- **Prediction:** How do we predict new or unseen values or attributes of the data

Réda DENAE 17

17

---

---


---

---

---

---

---



## What we will cover

1. Regression - Gradient descent - Régularization Lasso and Ridge
2. Linear Classification - Logistic Regression - MultiClass Classification
3. Backpropagation Algorithm, Neural Networks
4. Support Vector Machine : Linear version
5. Support Vector Machine : Kernel Method
6. Decision Trees, Bagging, Boosting, Random Forest
7. Dimensionality Reduction
8. Unsupervised Learning

Réda DENAE 18

18

---

---



---

---

---

---

---



## Recommended Background

- Linear Algebra
  - Definitions, vectors, matrices, operations, properties
- Probability
  - Basics: what is a random variable, probability distributions, functions of a random variable
- Machine learning
  - Learning, modelling and classification techniques

19

---

---



---

---

---

---

---



## Data

**Structured Data**

Area	estate type	Distance to center	Energy class	Age	Number bedrooms	Price
100	Apartment	1,1	A	20	3	130000
150	House	5,6	A	21	5	180000
247	House	2,2	C	20	7	250000
987	House	0,5	D	1	10	1250000

features

Targets/labels

20

---

---



---

---

---

---

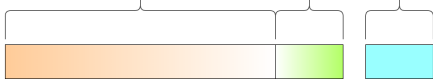
---



## Training, Validation and Testing Datasets

- The available dataset is subdivided into 3 datasets:
  1. Training Dataset (generally 60% for small dataset, 98% for large dataset): sample data used to fit the model.
  2. Validation Dataset (generally 20% for small dataset, 1% for large dataset): used to provide an unbiased evaluation of the model fit when you tune the model's parameters (avoid overfitting).
  3. Test Dataset (generally 20% for small dataset, 1% for large dataset): The sample of data used to provide an unbiased evaluation of a final model fit on the training dataset.

Train      Validation      Test



21

---

---

---

---

---

---

---

**Resampling and K-Fold Cross-Validation**

- The need for multiple training/validation sets  $\{X_i, V_i\}_i$ :  
Training/validation sets of fold  $i$
- K-fold cross-validation: Divide  $X$  into  $k$ ,  $X_i, i=1, \dots, K$ 

$$V_1 = X_1 \quad T_1 = X_2 \cup X_3 \cup \dots \cup X_K$$

$$V_2 = X_2 \quad T_2 = X_1 \cup X_3 \cup \dots \cup X_K$$

$$\vdots$$

$$V_K = X_K \quad T_K = X_1 \cup X_2 \cup \dots \cup X_{K-1}$$
- $T_i$  share  $K-2$  parts

Réda DENAE 22

22

---

---

---

---

---

---

---

---

**ONE ITERATION OF A 5-FOLD CROSS-VALIDATION:**

Réda DENAE 23

23

---

---

---

---

---

---

---

---

**Under – Overfitting vs. Bias/Variance**

Réda DENAE 24

24

---

---

---

---

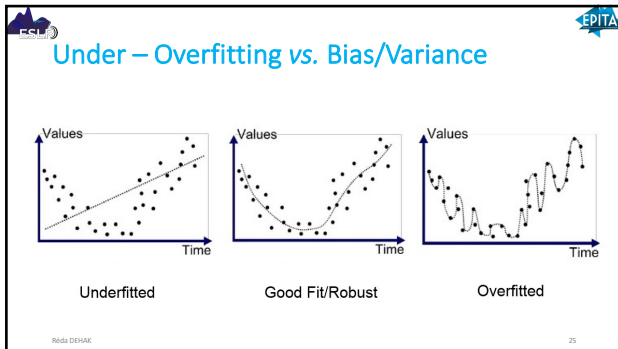
---

---

---

---





25

---

---

---

---

---

---

---

---

**Under – Over fitting**

- **Overfitting:**
  - Model too complex (flexible)
  - Fits « noise » in the training data
  - Don't perform well on new data (bad generalization)
- **Underfitting:**
  - Model too simplistic (too rigid)
  - Not powerful enough to capture salient patterns in data

Réda DENAE 26

26

---

---

---

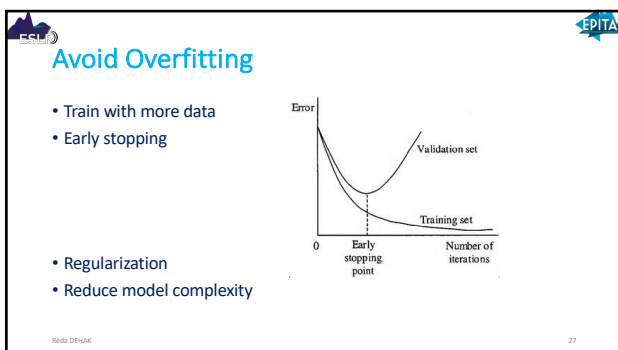
---

---

---

---

---



27

Generally, the error of testset is higher than validation set;  
Do not use validation set too many times to avoid over fitting on it

---

---

---



---

---

---

---

---

## Regression - Gradient descent - Regularization Lasso and Ridge

28

---

---



---

---

---

---

---

## What is a regression

- Analyzing relationship between variables
- Expressed in many forms
- Wikipedia
  - Linear regression, Simple regression, Ordinary least squares, Polynomial regression, General linear model, Generalized linear model, Discrete choice, Logistic regression, Multinomial logit, Mixed logit, Probit, Multinomial probit, ....
- Generally a **tool to predict variables**

Rida DENAE 29

29

---

---



---

---

---

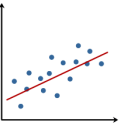
---

---

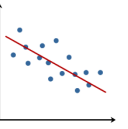



## Regressions for prediction

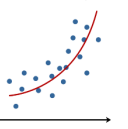
Linear



Linear



No linear relationship



$y = f(x; \theta) + \epsilon$

- $x$  is a scalar or a vector
- $f(\cdot)$  is a linear or affine function  $\rightarrow$  **Linear Regression**
- $f(\cdot)$  is a non-linear function  $\rightarrow$  **Non Linear Regression**
- $\theta$  is the model's ( $f$ ) parameters

Rida DENAE 30

30

---

---

---

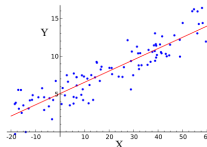
---

---

---

---

**A linear regression**



- Assumption: relationship between variables is linear
  - A linear *trend* may be found relating  $x$  and  $y$
  - $y$  = *dependent* variable
  - $x$  = *explanatory* variable
  - Given  $x$ ,  $y$  can be predicted as an affine function of  $x$

Réda DENAE 31

31

---

---

---

---

---

---

---

---

**Linear Regressions**

$$y = ax + b + \varepsilon$$

$\varepsilon$  : prediction error

- Given a "training" set of  $\{x, y\}$  values: estimate  $a$  and  $b$ 
  - $y_1 = ax_1 + b + \varepsilon_1$
  - $y_2 = ax_2 + b + \varepsilon_2$
  - $y_3 = ax_3 + b + \varepsilon_3$
  - ...
- If  $a$  and  $b$  are well estimated, prediction error will be small

Réda DENAE 32

32

---

---

---

---

---

---

---

---

**Matrix representation of Linear Regression**

- Define:
 
$$Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \end{bmatrix} \quad X = \begin{bmatrix} x_1 & x_2 & x_3 & \dots \\ 1 & 1 & 1 & \dots \end{bmatrix}$$

$$e = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \end{bmatrix} \quad A = \begin{bmatrix} a \\ b \end{bmatrix}$$

$$Y = X^T A + e$$

Réda DENAE 33

33

---

---

---

---

---

---

---

---

**Learning the Parameters**

- $Y = X^T A + e$
- Learning the parameters  $\rightarrow$  minimize cost function
 
$$E = \frac{1}{N} \sum_{i=1}^N e_i^2 = \frac{\|e\|^2}{N} = \frac{e^T e}{N}$$
- $e = Y - X^T A$
- $E = \frac{\|e\|^2}{N} = \frac{1}{N} (Y - X^T A)^T (Y - X^T A)$
- $E = \frac{1}{N} (Y^T Y - 2 A^T X Y + A^T X X^T A)$
- $\frac{dE}{dA} = \frac{1}{N} (-2 X Y + 2 X X^T A)$
- $\frac{dE}{dA} = 0 \Rightarrow A = (X X^T)^{-1} X Y$

Réda DENAE 34

34

---

---

---

---

---

---

---

---

**Learning Parameters**

- Linear Regression :
 
$$A = (X X^T)^{-1} X Y$$
- If the dimension of  $(X X^T)$  is important, we can use numeric solution rather than analytic one
- Gradient Descent on the Error formula :
  - $\text{Argmin}_A \frac{1}{N} (Y^T Y - 2 A^T X Y + A^T X X^T A) = \text{Argmin}_A \frac{1}{N} (A^T X X^T A - 2 A^T X Y)$

Cost function  $\Rightarrow \frac{1}{N} (A^T X X^T A - 2 A^T X Y)$

Réda DENAE 35

35

---

---

---

---

---

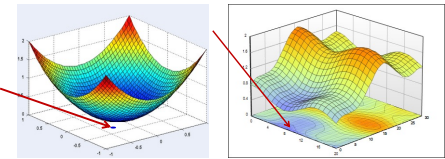
---

---

---

**Examples of Optimization : Multivariate functions**

- Find the optimal point in these functions



Réda DENAE 36

36

---

---

---


---

---

---

---

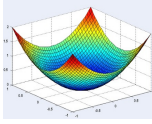
---



## Gradients of scalar functions with multi-variate inputs

- Consider  $f(X) = f(x_1, x_2, \dots, x_n)$

$$\nabla f(X) = \begin{bmatrix} \frac{\partial f(X)}{\partial x_1} \\ \frac{\partial f(X)}{\partial x_2} \\ \vdots \\ \frac{\partial f(X)}{\partial x_n} \end{bmatrix}$$



- The function  $f(X)$  increases most rapidly if the input increment  $\Delta X$  is perfectly aligned to  $\nabla f(X)$
- The gradient is the direction of fastest increase in  $f(X)$**

Réda DENAE  
37

---

---

---

---


---

---

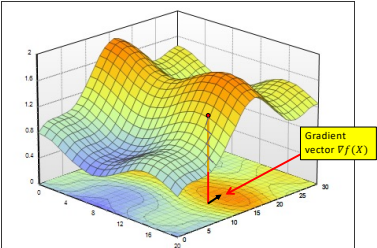
---

---

37



## Gradient



Réda DENAE  
38

---

---

---

---


---

---

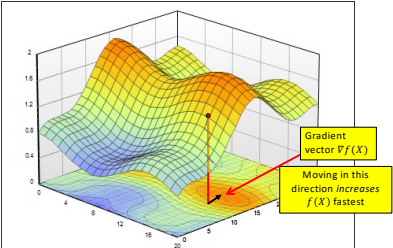
---

---

38



## Gradient



Réda DENAE  
39

---

---

---

---

---

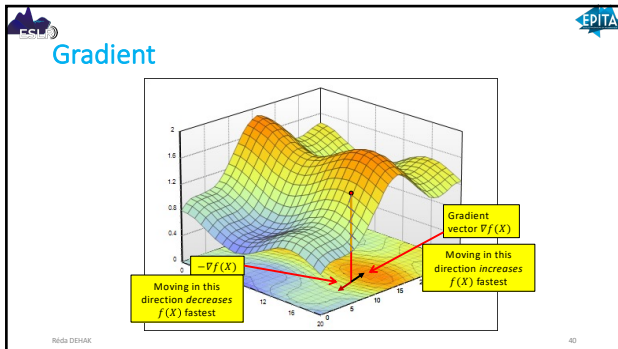
---

---

---

39

13




---

---

---

---

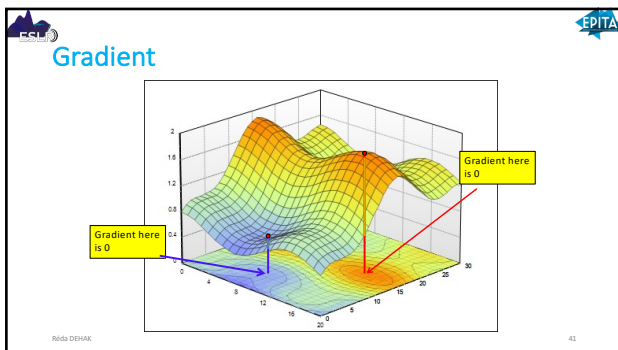
---

---

---

---

40




---

---

---

---

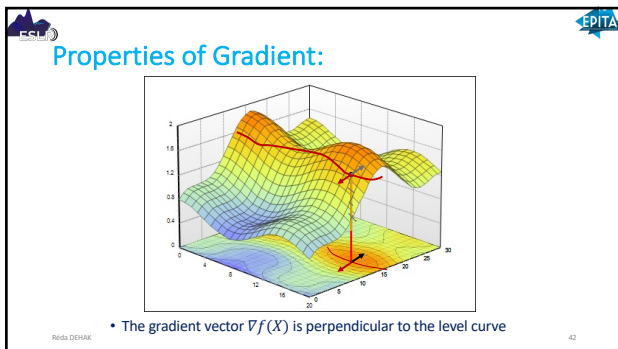
---

---

---

---

41




---

---

---

---

---

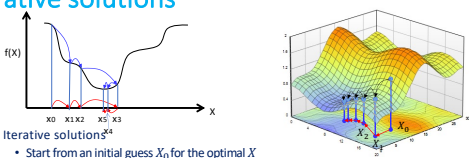
---

---

---

42

**Iterative solutions**



- Iterative solutions<sup>44</sup>
  - Start from an initial guess  $X_0$  for the optimal  $X$
  - Update the guess towards a (hopefully) "better" value of  $f(X)$
  - Stop when  $f(X)$  no longer decreases
- Problems:
  - Which direction to step in
  - How big must the steps be

Réda DENAE 43

43

---

---

---

---

---

---

---

---

**Descent methods**

- Iterative solutions that attempt to "descend" the function in steps to arrive at the minimum
- Based on the first order derivatives (gradient) and in some cases the second order derivatives (Hessian).
  - **Gradient descent** is based only on the first derivative
  - **Newton's method** is based on both first and second derivatives

Réda DENAE 44

44

---

---

---

---

---

---

---

---

**Gradient descent/ascent**

- The gradient descent/ascent method to find the minimum or maximum of a function  $f$  iteratively
  - To find a *maximum* move *in the direction of the gradient*

$$x^{k+1} = x^k + \eta^k \nabla f(x^k)$$
  - To find a *minimum* move *exactly opposite the direction of the gradient*

$$x^{k+1} = x^k - \eta^k \nabla f(x^k)$$
- What is the step size  $\eta^k$  (Learning rate)

Réda DENAE 45

45

---

---

---

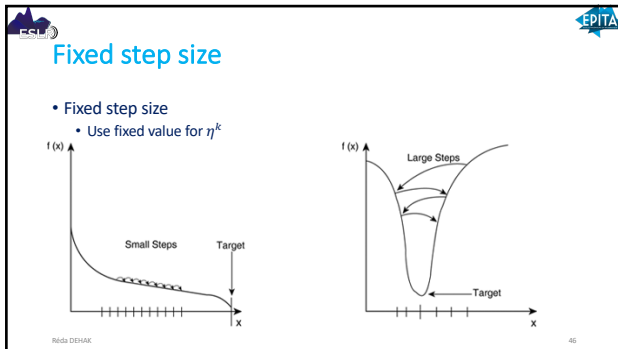
---

---

---

---

---




---

---

---

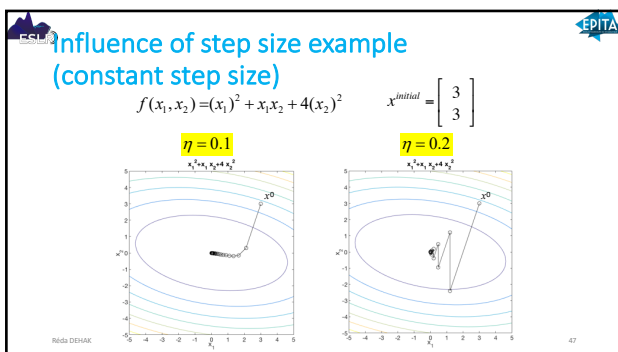
---

---

---

---

---




---

---

---

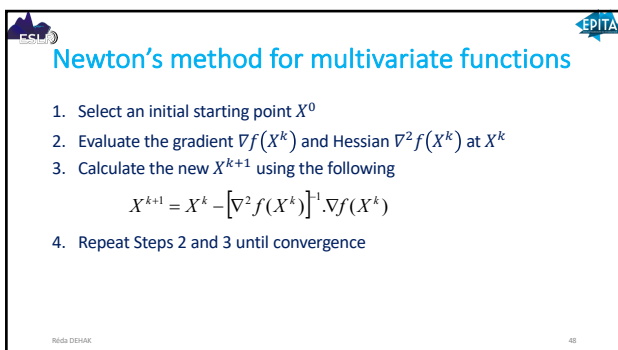
---

---

---

---

---




---

---

---

---

---

---

---

---



**Gradient Descent for linear Regression**

$$E = \frac{1}{N} (Y^T Y - 2 A^T X Y + A^T X X^T A)$$

$$\nabla E = \frac{2}{N} (X X^T A - X Y)$$

$$\nabla^2 E = \frac{2}{N} X X^T$$

Réda DENAE 49

49

---

---

---

---

---

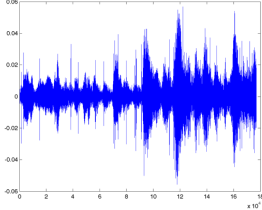
---

---

---

**A Common Problem**

- Can you spot the glitches?



Réda DENAE 50

50

---

---

---

---

---

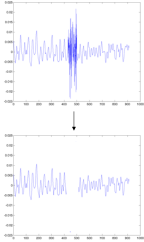
---

---

---

**How to fix this problem?**

- “Glitches” in audio
  - Must be detected
  - How?
- Then what?
- Glitches must be “fixed”
  - Delete the glitch
    - Results in a “hole”
  - Fill in the hole
  - How?



Réda DENAE 51

51

---

---

---

---

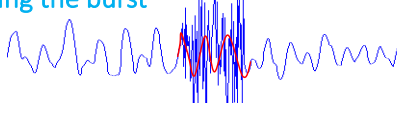
---

---

---

---

**Finding the burst**



- At each time
  - Learn a "forward" predictor  $a_k$
  - At each time, predict next sample  $x_{t+1}^{est} = \sum a_{k,k} x_{t-k}$
  - Compute error:  $error = |x_t - x_{t+1}^{est}|^2$
  - Learn a "backward" predictor and compute backward error
    - $b_{k,k}$
  - Compute average prediction error over window, threshold
- If the error exceeds a threshold, identify burst

Réda DENAE 52

52

---

---

---

---

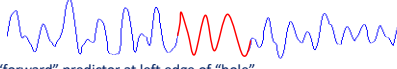
---

---

---

---

**Filling the hole**



- Learn "forward" predictor at left edge of "hole"
  - For each missing sample
  - At each time, predict next sample  $x_{t+1}^{est} = \sum a_{k,k} x_{t-k}$ 
    - Use estimated samples if real samples are not available
- Learn "backward" predictor at left edge of "hole"
  - For each missing sample
  - At each time, predict next sample  $x_{t+1}^{est} = \sum b_{k,k} x_{t-k}$ 
    - Use estimated samples if real samples are not available
- Average forward and backward predictions

Réda DENAE 53

53

---

---

---

---

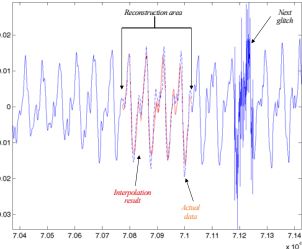
---

---

---

---

**Reconstruction zoom in**



Réda DENAE 54

54

---

---

---



---

---

---

---

---

## Linear Regression

- **Goal :** find a linear relationship between the dependent target variable  $y$  and the regression value  $x$ 

$$y = ax + b + \varepsilon = \hat{x}^T A + \varepsilon$$

Where  $A = \begin{bmatrix} a \\ b \end{bmatrix}$  and  $\hat{x} = \begin{bmatrix} x \\ 1 \end{bmatrix}$
- **Solution:** Find the optimal value of the cost function MSE (Mean Squared Error)
$$E = MSE = \frac{1}{N} \sum_{i=1}^N \varepsilon_i^2 = \frac{1}{N} (Y^T Y - 2 A^T X Y + A^T X X^T A)$$

Where  $Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \end{bmatrix}$      $X = \begin{bmatrix} x_1 & x_2 & x_3 & \dots \\ 1 & 1 & 1 & \dots \end{bmatrix}$

$$A = (X X^T)^{-1} X Y$$

Réda DENAE
55

---

---

---

---



---

---

---

---

55

## Multiple Linear Regression

- **Goal:** find a linear relationship between the dependent target variable  $y$  and the regression  $d$  vector  $x$ 

$$y = a_1 x_{(1)} + a_2 x_{(2)} + \dots + a_d x_{(d)} + b + \varepsilon = \hat{x}^T A + \varepsilon$$

Where  $A = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_d \\ b \end{bmatrix}$  and  $\hat{x} = \begin{bmatrix} x_{(1)} \\ x_{(2)} \\ \vdots \\ x_{(d)} \\ 1 \end{bmatrix}$
- **Solution:** Find the optimal value of the cost function MSE (Mean Squared Error)
$$E = MSE = \frac{1}{N} \sum_{i=1}^N \varepsilon_i^2 = \frac{1}{N} (Y^T Y - 2 A^T X Y + A^T X X^T A)$$

Where  $Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \end{bmatrix}$      $X = \begin{bmatrix} x_{1(1)} & x_{2(1)} & x_{3(1)} & \dots \\ \vdots & \vdots & \vdots & \vdots \\ x_{1(d)} & x_{2(d)} & x_{3(d)} & \dots \\ 1 & 1 & 1 & \dots \end{bmatrix}$

$$A = (X X^T)^{-1} X Y$$

Réda DENAE
56

---

---

---

---



---

---

---

---

56

## Regularization

- **Goal:** Avoid Overfitting
- Add penalty to the optimization process of the cost function

Réda DENAE
57

---

---

---

---

---

---

---

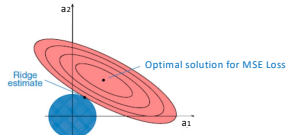
---

57

**Regularization: Ridge Regression**

- The **Ridge Regression** is a **regularization** technique that uses  **$L_2$  regularization** to impose a penalty on the size of coefficients.

$$E_{\text{Ridge}} = \underset{A}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^N \|\tilde{x}^T A - y_i\|^2 \quad \text{MSE Loss}$$

$$\|A\|^2 = \sum_{i=1}^d a_i^2 \leq \tau \quad \text{Penalty}$$


Ridge estimate

Optimal solution for MSE Loss

58

---

---

---

---

---

---

---

---

58

**Regularization: Ridge Regression**

- The **Ridge Regression** is a **regularization** technique that uses  **$L_2$  regularization** to impose a penalty on the size of coefficients.

$$E_{\text{Ridge}} = \underset{A}{\operatorname{argmin}} \underbrace{\frac{1}{N} \sum_{i=1}^N \|\tilde{x}^T A - y_i\|^2}_{\text{MSE Loss}} + \underbrace{\lambda \|A\|^2}_{\text{Penalty}} \quad \|A\|^2 = \sum_{i=1}^d a_i^2$$

- Here  $\lambda \geq 0$  is a tuning parameter, which controls the strength of the penalty term. Note that:
  - When  $\lambda = 0$  ( $\tau = +\infty$ ), we get the linear regression estimate
  - When  $\lambda = +\infty$  ( $\tau = 0$ ) we get  $A_{\text{Ridge}} = 0$
  - For  $\lambda$  in between, we are balancing two ideas: fitting a linear model of  $y$  on  $x$ , and shrinking the coefficients

59

---

---

---

---

---

---

---

---

59

**Regularization: Ridge Regression**

- The  **$x$  vectors must be centered and normalized** to prevent influence of high variance features
- The target variable  **$y$  must be centered to remove** the constant of the regression
- Solution:**

$$A = (XX^T + \lambda I_d)^{-1}XY$$
 Where  $I_d$  is the identity matrix

`sklearn.linear_model.Ridge`

60

---

---

---

---

---

---

---

---

60

**Regularization: Lasso Regression**

*Least Absolute Shrinkage and Selection Operator*

- The **Lasso Regression** is a **regularization** technique that uses  **$L_1$  regularization** to impose a penalty on the size of coefficients.

$$A_{\text{Lasso}} = \underset{w}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^N \|\tilde{x}^T A - y_i\|^2 \quad \text{MSE LOSS}$$

$$\|A\|_1 = \sum_{i=1}^d |a_i| \leq \tau \quad \text{Penalty}$$

Optimal solution for MSE Loss

parameters chosen by ridge

parameters chosen by LASSO

61

---

---

---

---

---

---

---

---

**Regularization: Lasso Regression**

*Least Absolute Shrinkage and Selection Operator*

- The **Lasso Regression** is a **regularization** technique that uses  **$L_1$  regularization** to impose a penalty on the size of coefficients.

$$A_{\text{Lasso}} = \underset{w}{\operatorname{argmin}} \underbrace{\frac{1}{N} \sum_{i=1}^N \|\tilde{x}^T A - y_i\|^2}_{\text{MSE Loss}} + \underbrace{\lambda \|A\|_1}_{\text{Penalty}} \quad \|A\|_1 = \sum_{i=1}^d |a_i|$$

- Here  $\lambda \geq 0$  is a tuning parameter, which controls the strength of the penalty term. Note that:
  - When  $\lambda = 0$  ( $\tau = +\infty$ ), we get the linear regression estimate
  - When  $\lambda = +\infty$  ( $\tau = 0$ ), we get  $W_{\text{Lasso}} = 0$
  - For  $\lambda$  in between, we are balancing two ideas: fitting a linear model of  $y$  on  $X$ , and shrinking the coefficients

62

---

---

---

---

---

---

---

---

**Regularization: Lasso Regression**

- The lasso regression performs a **feature selection**
- No analytic formula: Gradient descent  
`Sklearn.linear_model.Lasso`

63

---

---

---



---

---

---

---

---

## Conclusions:

- Linear Regression is simple and useful method
- Two training algorithms:
  - Analytic solution
  - Numeric solution: Gradient descent
- Question: Relationships are not always linear, so how do we model these cases?

Roba DENAE
64

---

---

---

---

---

---

---