Principals and Elementary Models

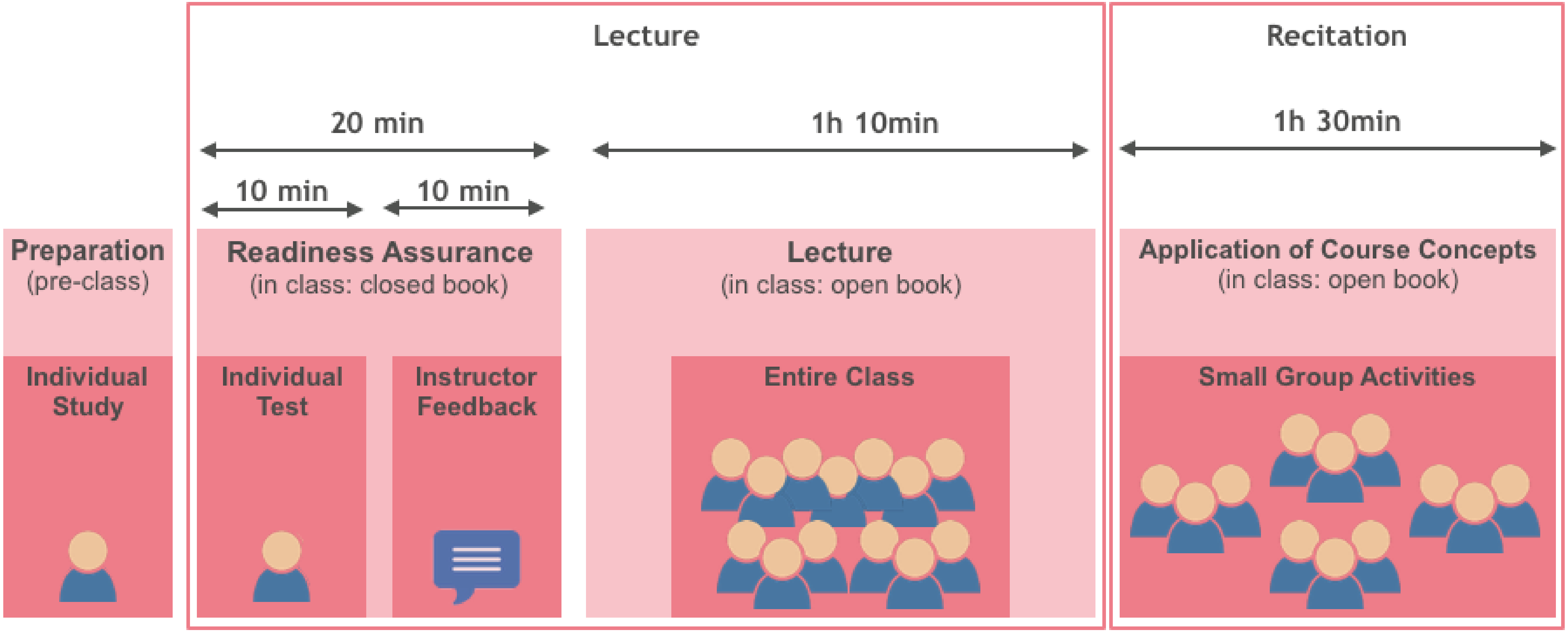
Franck JAOTOMBO

em lyon business school

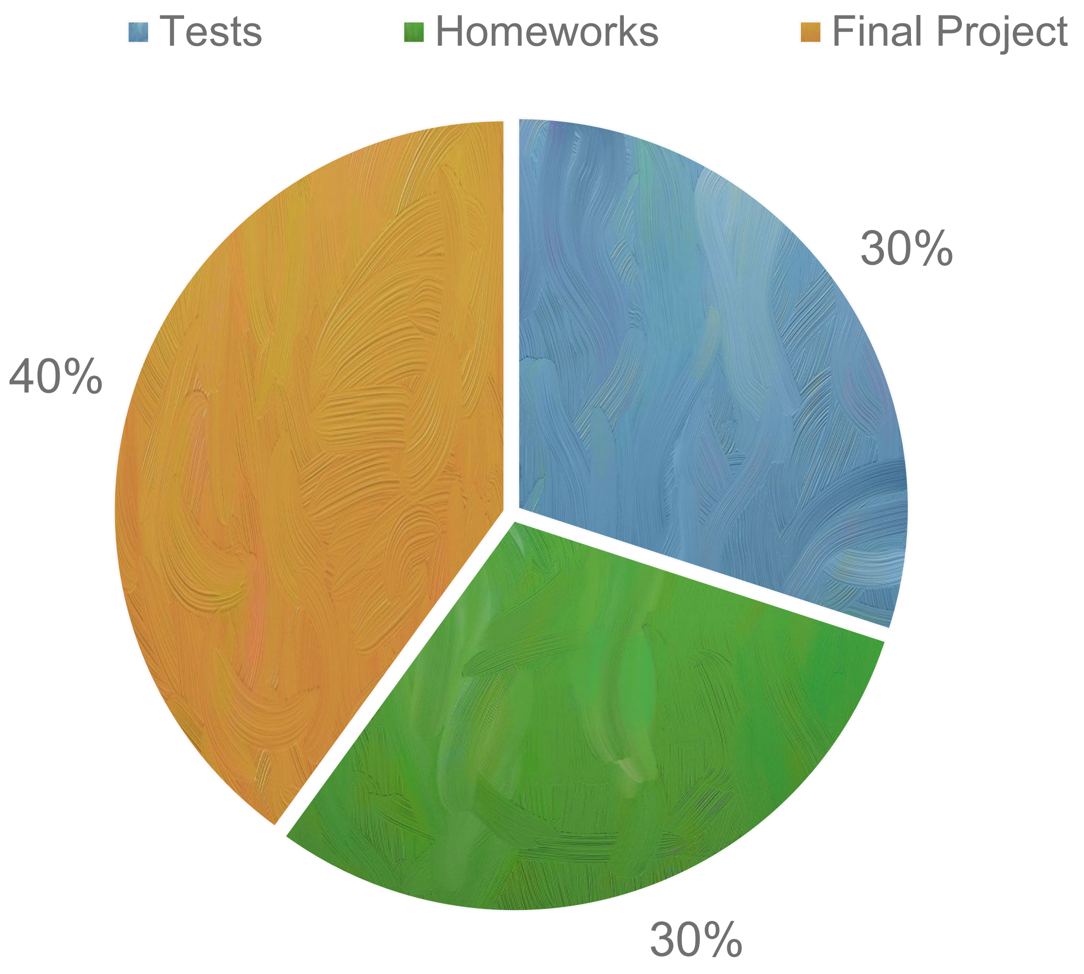
Session 6 – Gradient Descent

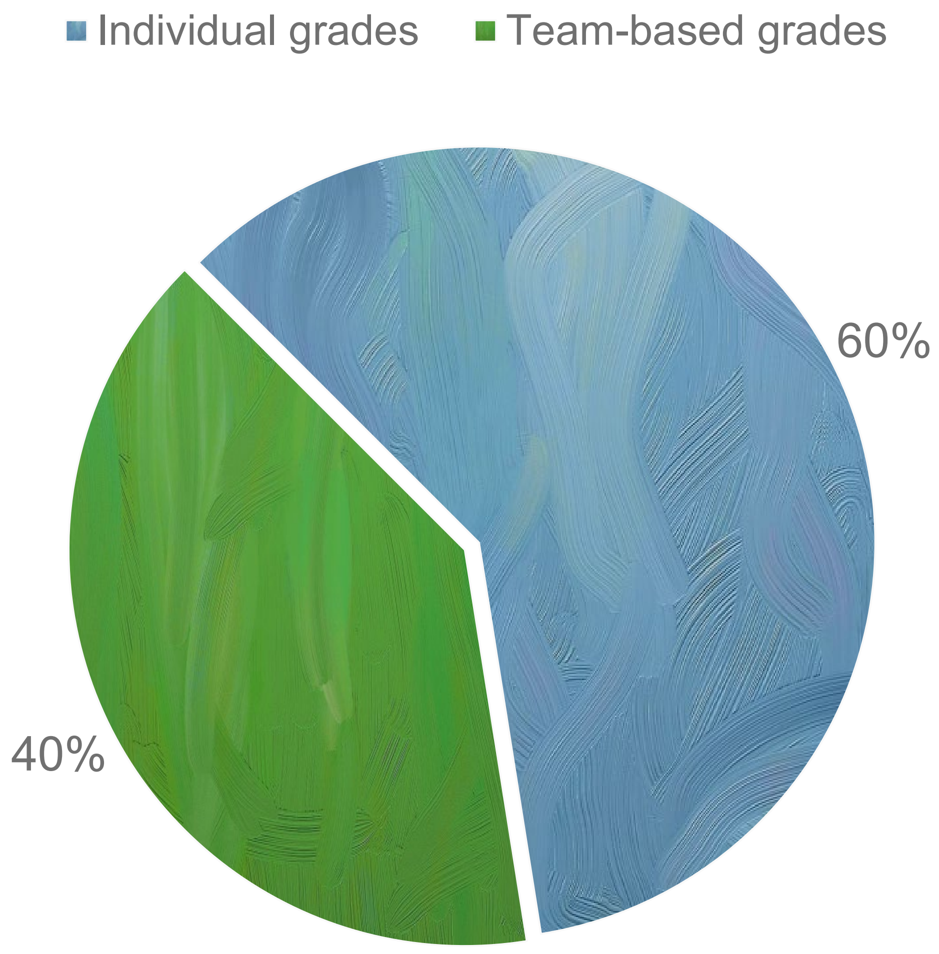
How do we minimize Loss in Machine Learning

Course Architecture



Grades


Tests
Homeworks
Final Project
30%
30%
40%


Individual grades
Team-based grades
60%
40%

em lyon business school

Grades : extra 10-20% bonus points

- **You may be in one or several of the following cases :**
  - You love challenges and excellence
  - You already have a good background in Python / Machine Learning
  - You want to tackle a real-life problem head on and hone your skills in doing so
  - You love Coding / Machine Learning

- **Then you may ask optional topics and subjects which will yield as many as 20% of extra points**
  - So, in short if you take this option, you will be graded on 110-120 points over 100
  - Come and see me if you are interested, we will select something for you
  - Please, do not consider this option unless you are confident that you can really do well on the other parts of the class

Rules 1

- **Attendance**
  - One seriously justified missed class may be tolerated : beyond = FAIL course
  - On campus attendance is the rule – not online

- **Quizzes**
  - Quizzes are to be taken in class only – not remotely
  - Missing a quiz = 0

- **Homework**
  - Each homework is to be submitted individually and on time
  - Failing to submit a homework = 0

em lyon business school

Rules 2

- **Group Presentations**
  - Every team member must be able to explain each part of the project (code and concepts)
    - I will interrogate each of you on differents parts of the code and of the theoretical concepts
  - The Project report is graded as a group work (20% of the final grade)
  - The Presentation is graded individually (20% of the final grade)
  - There is a bonus for having fun
  - There is a bonus for asking good questions

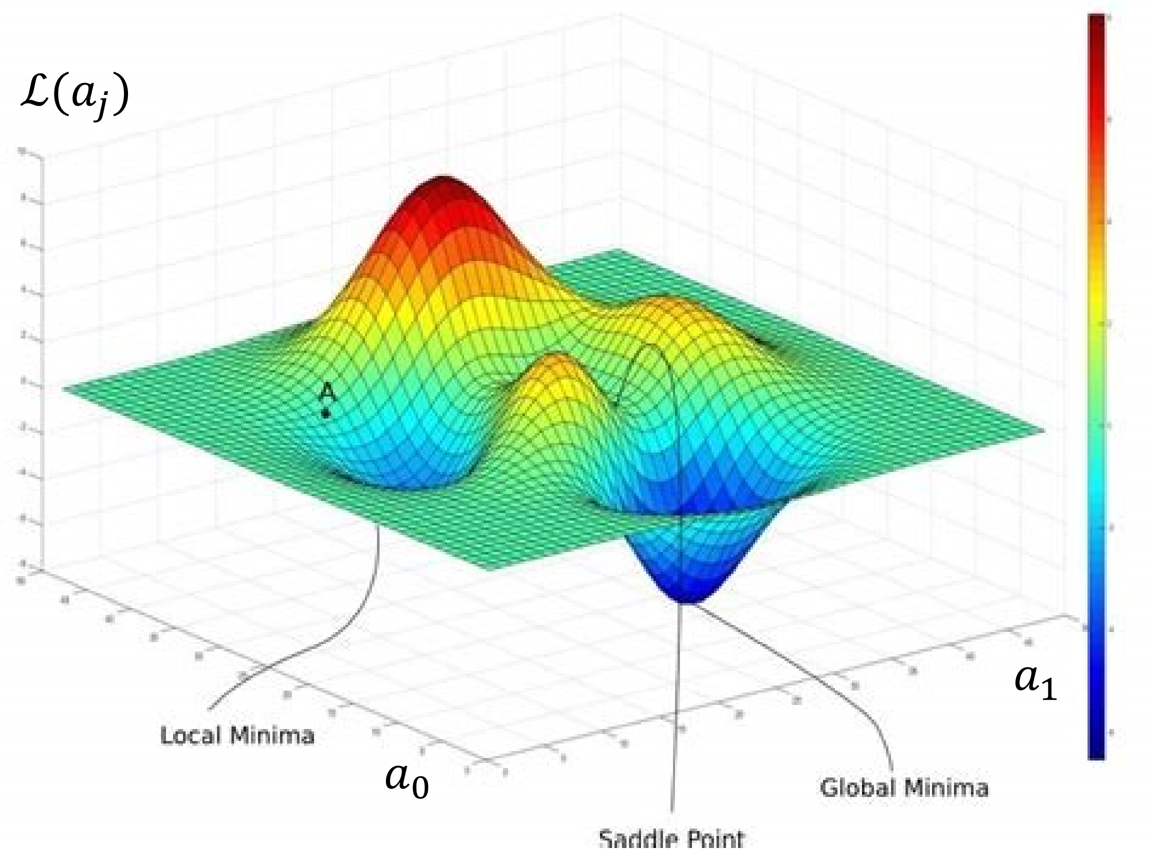
- **Plagiarism**
  - The individual homeworks are ***individual …***!
  - Plagiarism : 0 for the related works + possible disciplinary board
  - Using Generative AI & Large Language Model (ChatGPT or else)
    - I have nothing against it, however it must be acknowledged
    - Please mention it in the introduction of the related work or as a disclaimer

Gradient Descent : general concept

- **Algorithm**
  - Given a learning rate $\alpha > 0$ and a Loss $\mathcal{L}(\boldsymbol{a})$
  - Repeat until convergence :

$$a_j \leftarrow a_j - \alpha \frac{\partial \mathcal{L}(\boldsymbol{a})}{\partial a_j}$$

Where : $\boldsymbol{a} = (a_0, \ldots, a_p)$



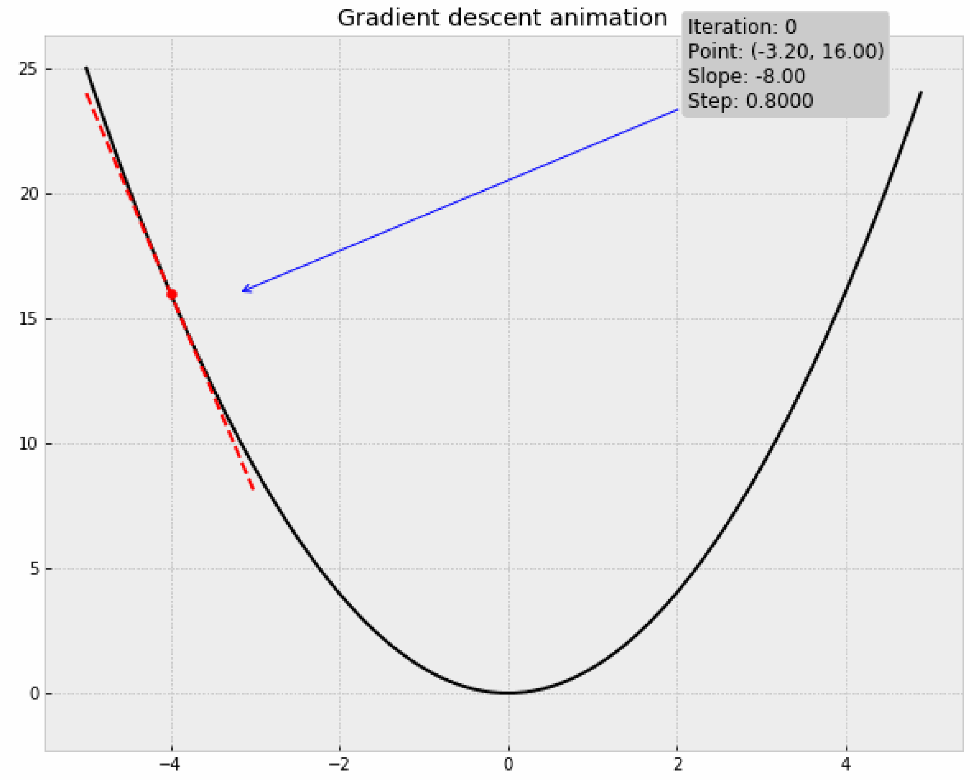
Gradient Descent : linear regression

em lyon business school

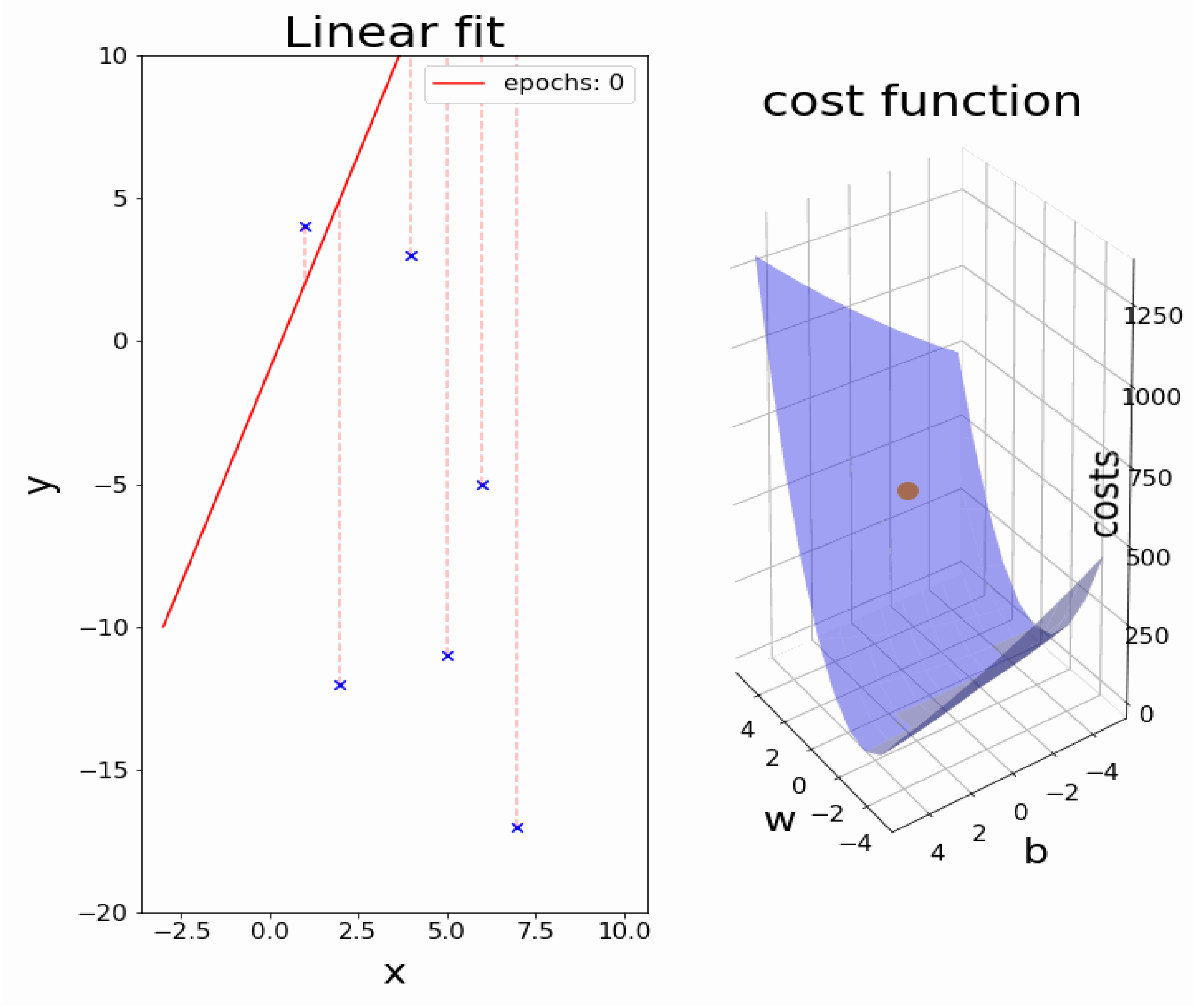
- **Loss function for Linear Regression**

$$h(\boldsymbol{X}) = a_0 + a_1 X_i^1 + \cdots + a_p X_i^p$$

$$\mathcal{L}(\boldsymbol{a}) = \frac{1}{2n} \sum_{i=1}^{n} [y - h(\boldsymbol{X})]^2$$



Gradient Descent in Action



Gradient Descent for Logistic Regression

- **Loss Function**

$$h(\boldsymbol{X}) = \; a_0 + a_1 X_i^1 + \cdots + a_p X_i^p$$

$$\pi(x) = \sigma(\boldsymbol{Xa}) = \frac{1}{1 + e^{-h(\boldsymbol{X})}}$$

$$\mathcal{L}(\boldsymbol{a}) = \frac{1}{n}\sum_{i=1}^{n}[\, y_i \, ln\, \pi_i(x) + \; (1 - y_i) ln(1 - \pi_i(x))]$$