University of Science, VNU-HCM
Faculty of Information Technology

Introduction to Data Science Course

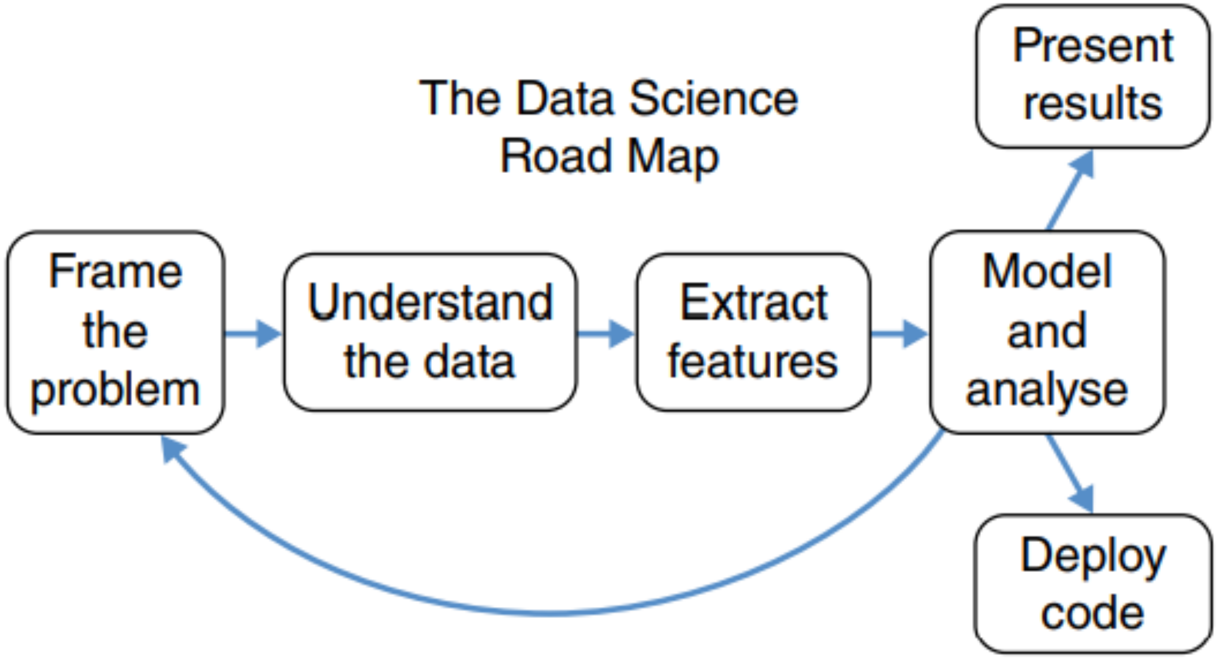
Data Modeling (Part 1)

Le Ngoc Thanh
lnthanh@fit.hcmus.edu.vn
Department of Computer Science

Ho Chi Minh City

Contents

Process



The Data Science Road Map
Present results
Frame the problem
Understand the data
Extract features
Model and analyse
Deploy code

After preprocessing

Data Science Process

- Give the question to answer
- Collecting data
- Data Discovery & preprocessing to obtain data that can be analyzed
- Data analysis (in visualizations, statistics,machine learning)
  → answers (hypotheses) for the question
- Evaluation
- Decision Making

Data Science vs. Machine Learning

Data Science

Field that determines the processes, systems, and tools needed to transform data into insights to be applied to various industries.

Skills needed:

- Statistics
- Data visualizatiom
- Coding skills (Python/R)
- Machine learning
- SQL/NoSQL
- Data wrangling

Machine learning is part of data science. Its algorithms train on data delivered by data science to "learn."

Skills needed:

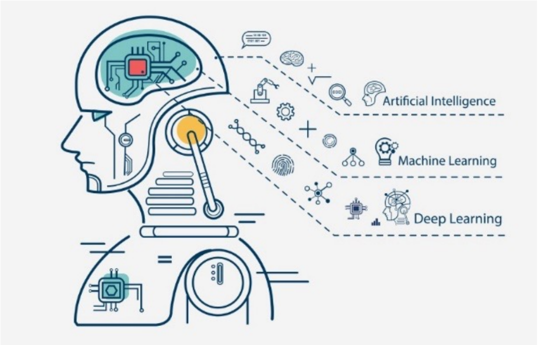
- Math, statistics, and probability
- Comfortable working with data
- Programming skills

Machine Learning

Field of artificial intelligence (AI) that gives machines the human-like capability to learn and adapt through statistical models and algorithms.

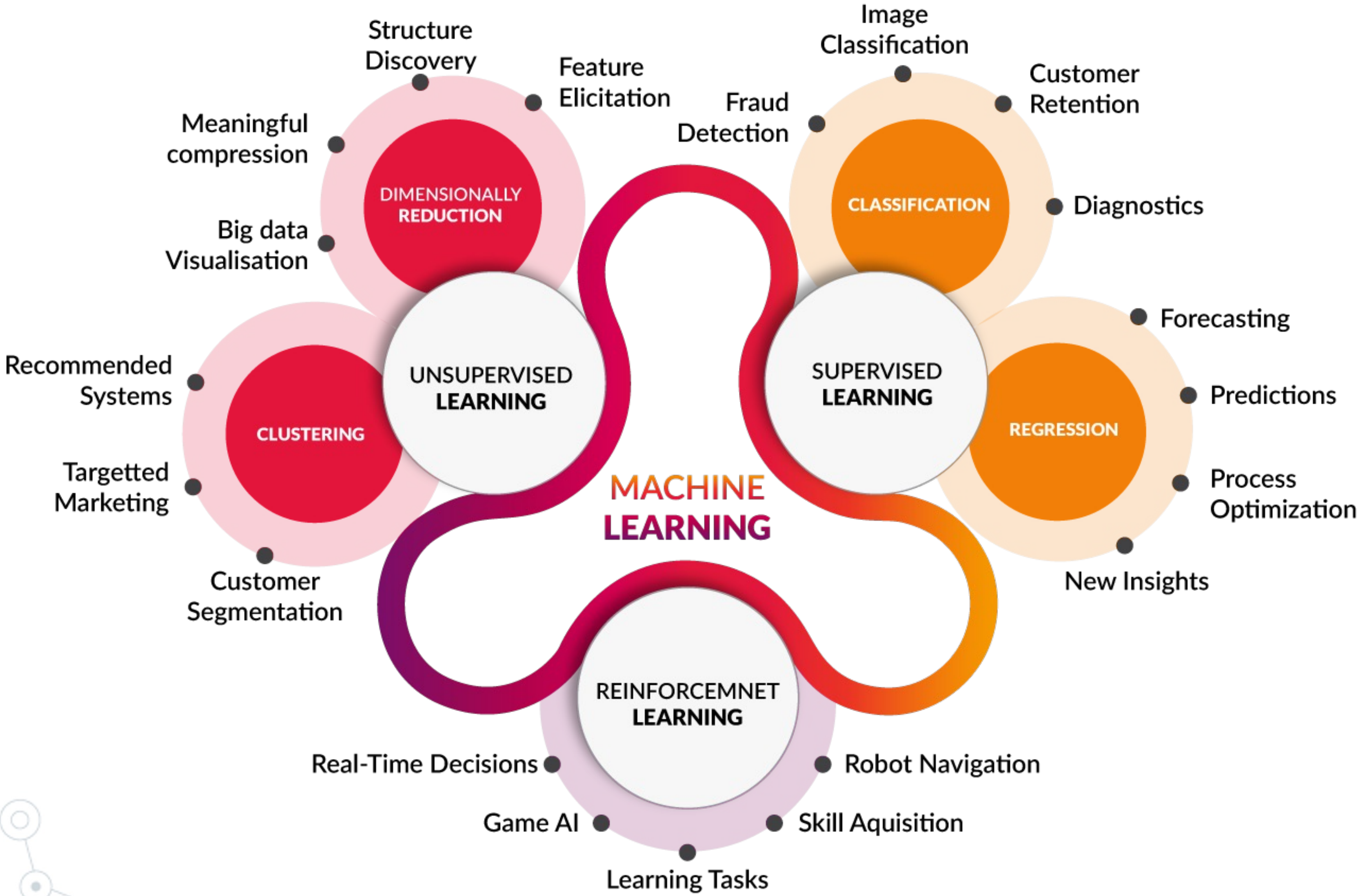
Skills needed:

- Programming skills (Python, SQL, Java)
- Statistics and probability
- Prototyping
- Data modeling



https://www.coursera.org/articles/data-science-vs-machine-learning

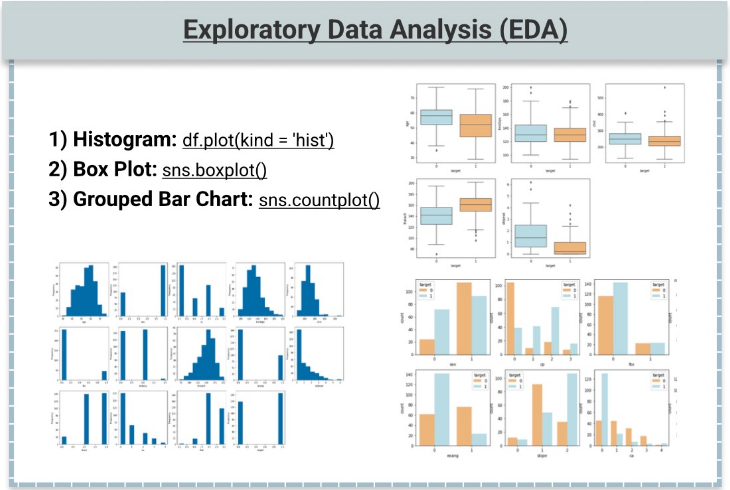
ML Tasks



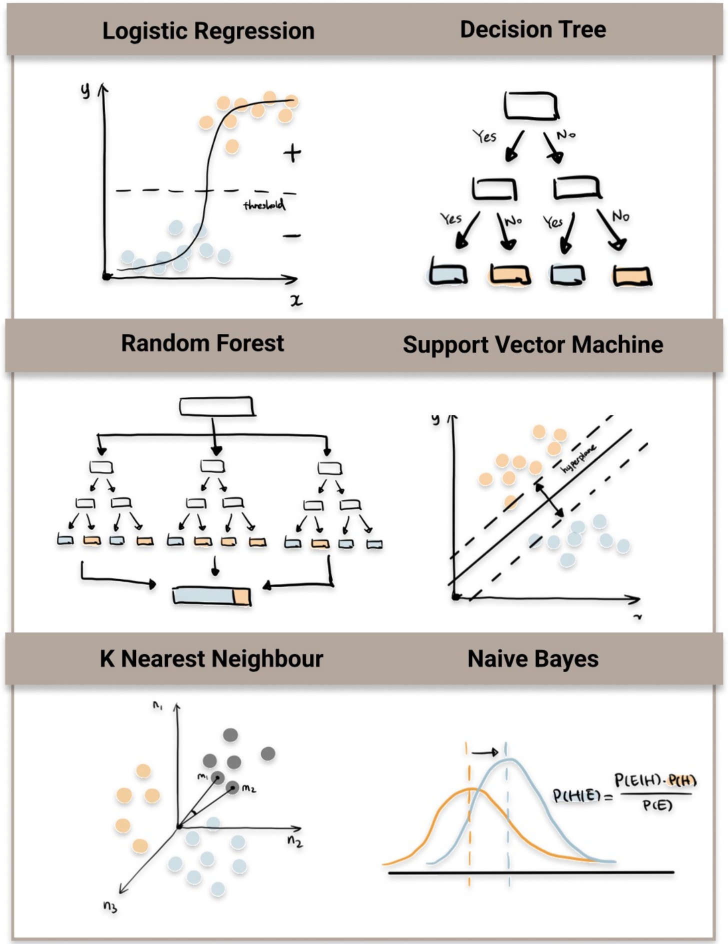
Structure Discovery
Feature Elicitation
Meaningful compression
DIMENSIONALLY REDUCTION
Big data Visualisation
Recommended Systems
CLUSTERING
Targetted Marketing
Customer Segmentation
UNSUPERVISED LEARNING
Image Classification
Customer Retention
Fraud Detection
CLASSIFICATION
Diagnostics
SUPERVISED LEARNING
Forecasting
Predictions
REGRESSION
Process Optimization
New Insights
MACHINE LEARNING
REINFORCEMNET LEARNING
Real-Time Decisions
Robot Navigation
Game AI
Skill Aquisition
Learning Tasks

Machine Learning Choice

- Before implementing the machine learning (ML) model, the data scientist needs to identify (several) branches in ML that can solve the given problem.



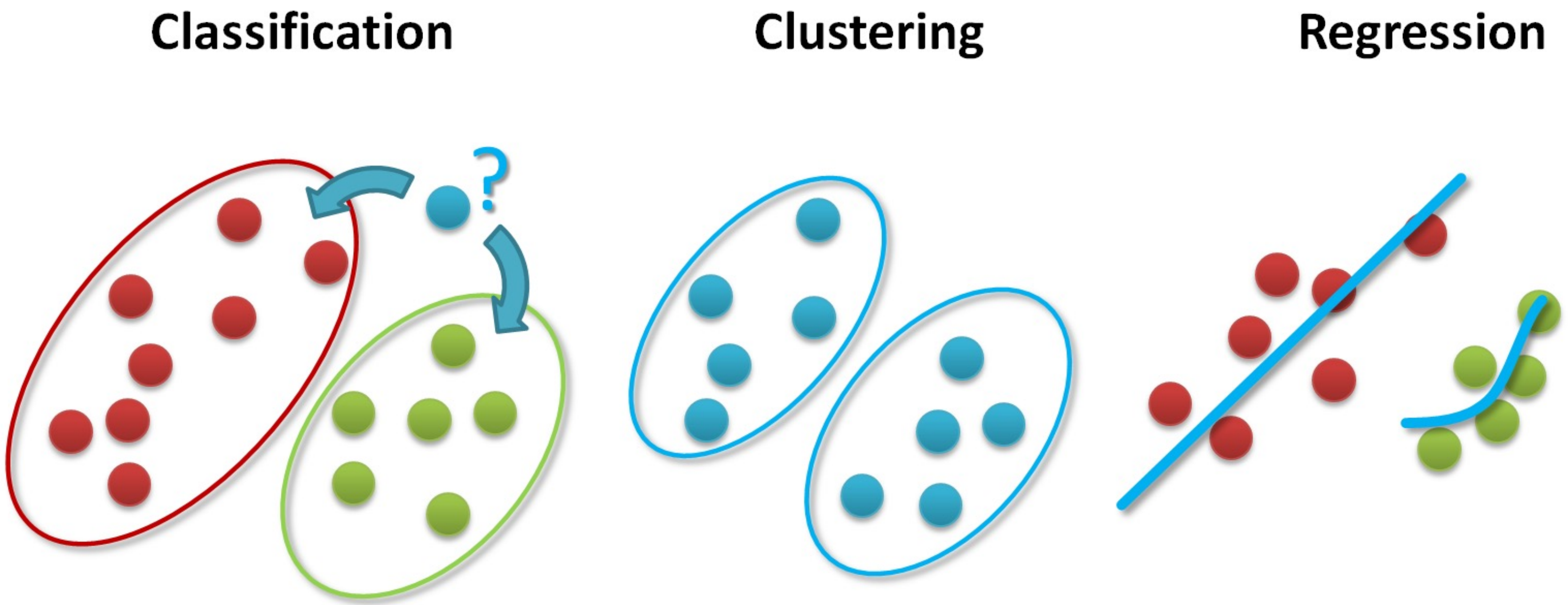
Visualization and Statistics



Machine Learning

The course's focus

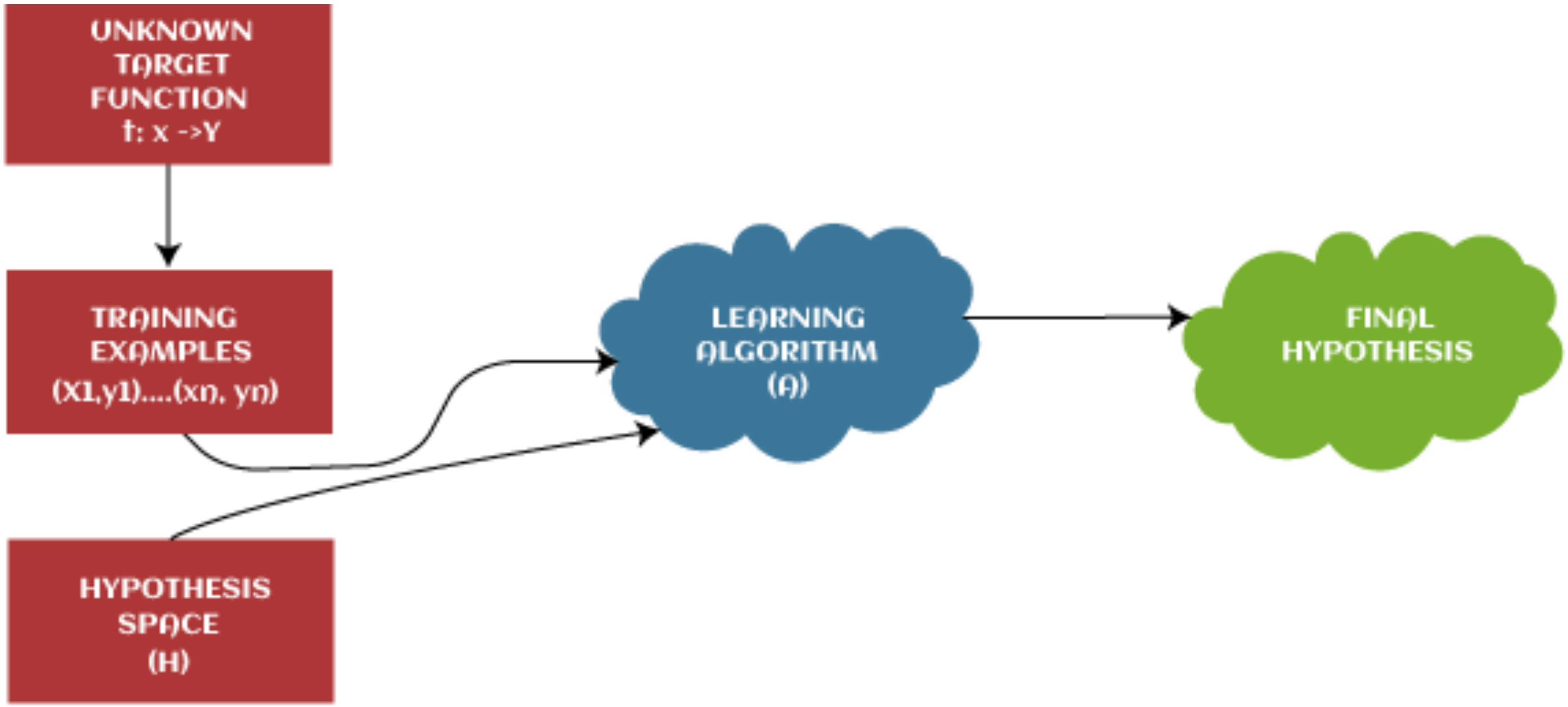
- In this course, we focus on three main groups of ML:
    - Regression
    - Classification
    - Clustering



Contents

After hypothesis

- The job of a learning algorithm to find the best suitable hypothesis for a problem.



After hypothesis

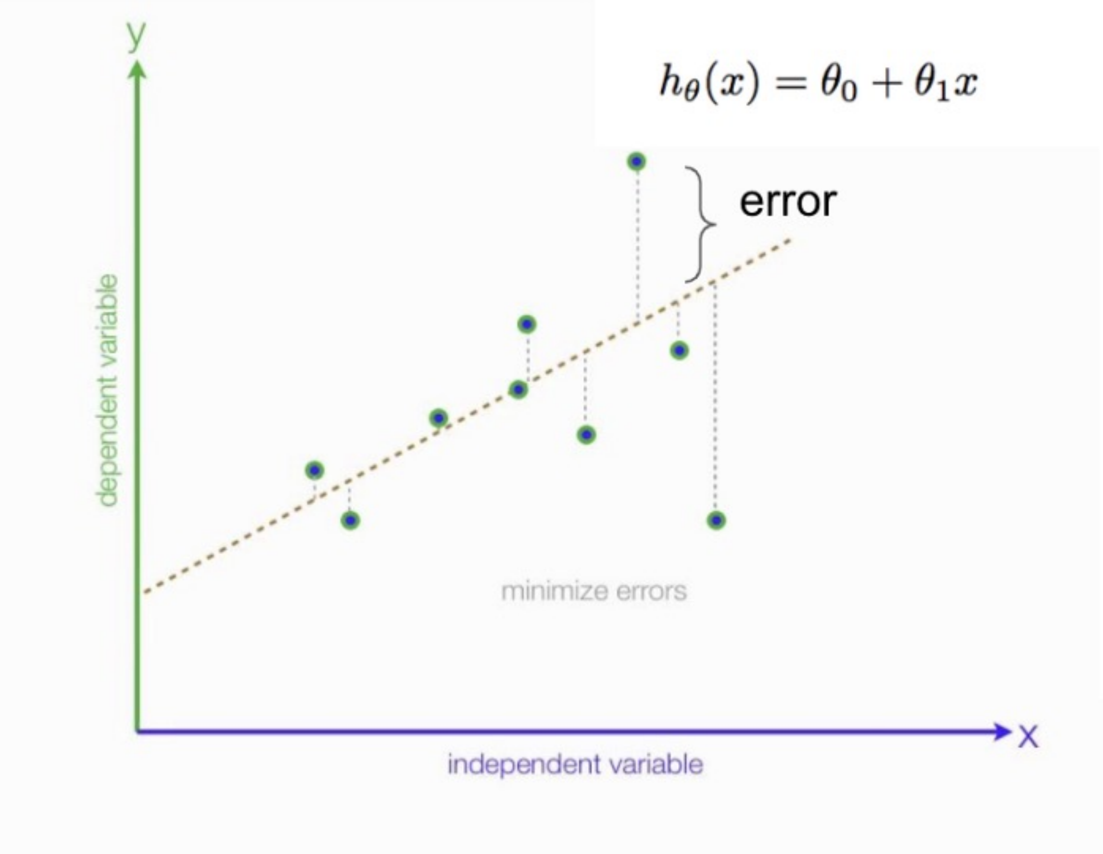
- To choose the suitable hypothesis, we need to define the loss function.

$$\mathcal{L}(y - \hat{y}) = \sum_{i=1}^{n} (y - \hat{y_i})^2$$

Machine learning = *iterative procedure to find a minimum of loss for the given data.*

After loss function design

- We are looking for what parameters to produce the lowest loss rate for given dataset, so we need the process to optimize the function (fitting).



Hypothesis:

$$h_\theta(x) = \theta_0 + \theta_1 x$$

Parameters:

$$\theta_0, \theta_1$$

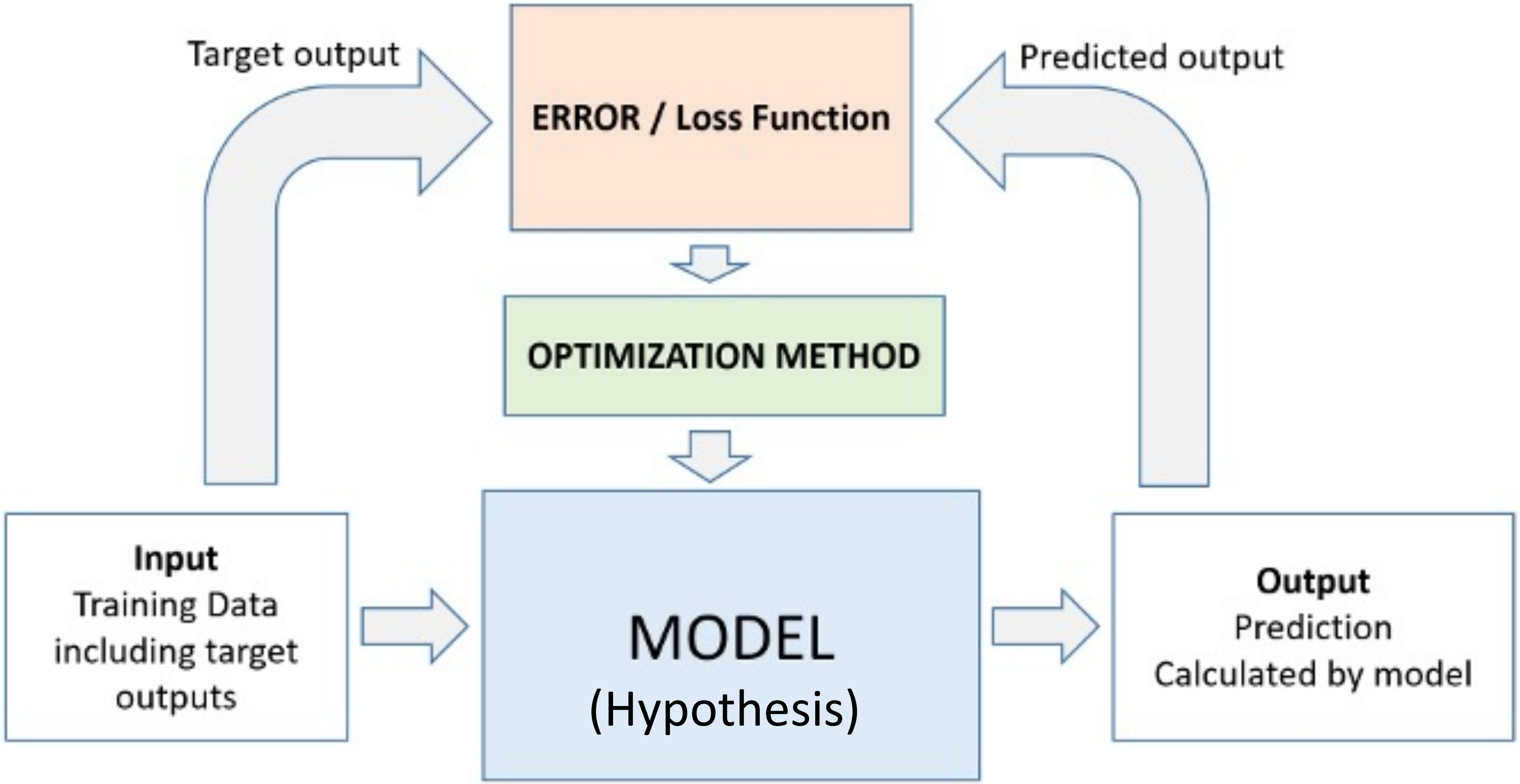
Cost Function:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

Goal:

$$\underset{\theta_0, \theta_1}{minimize} J(\theta_0, \theta_1)$$

General model learning architecture



Contents

- Data science and machine learning
- Machine learning architecture
- **Regression model**
  - Linear regression
  - Non-linear regression
  - Over- and Under-Determined Systems
  - Model selection
  - Overfitting

Regression

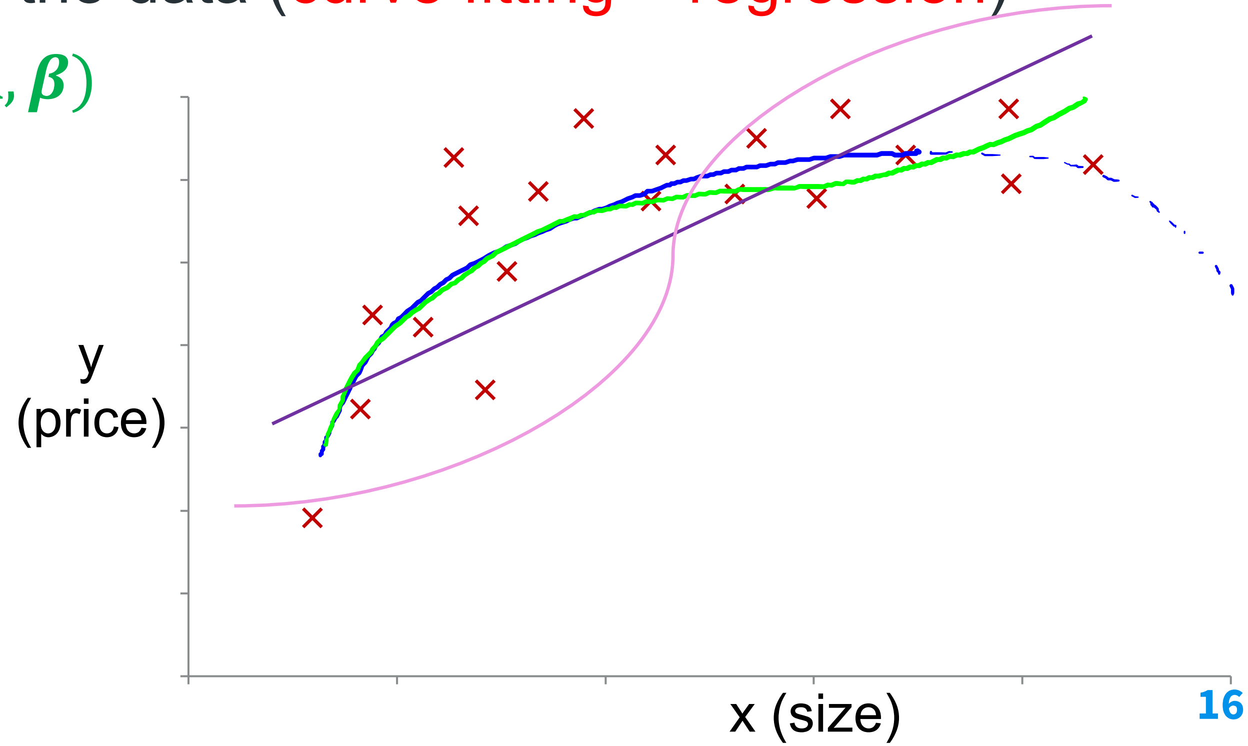
- Consider a set of n data points:

$$(x_1, y_1), (x_2, y_2), (x_3, y_3), \ \dots, (x_n, y_n)$$

- Purpose:
  - Select a function f (·) and fit it to the data (curve fitting = regression)

$$\mathbf{Y} = f(\mathbf{A}, \boldsymbol{\beta})$$

Size in feet² (x)	Price ($) in 1000's (y)
100	10
800	150
1534	315
852	178



Linear regression

- Assume that a line is fitted through the points (hypothesis)

$$f(x) = \beta_1 x + \beta_2$$

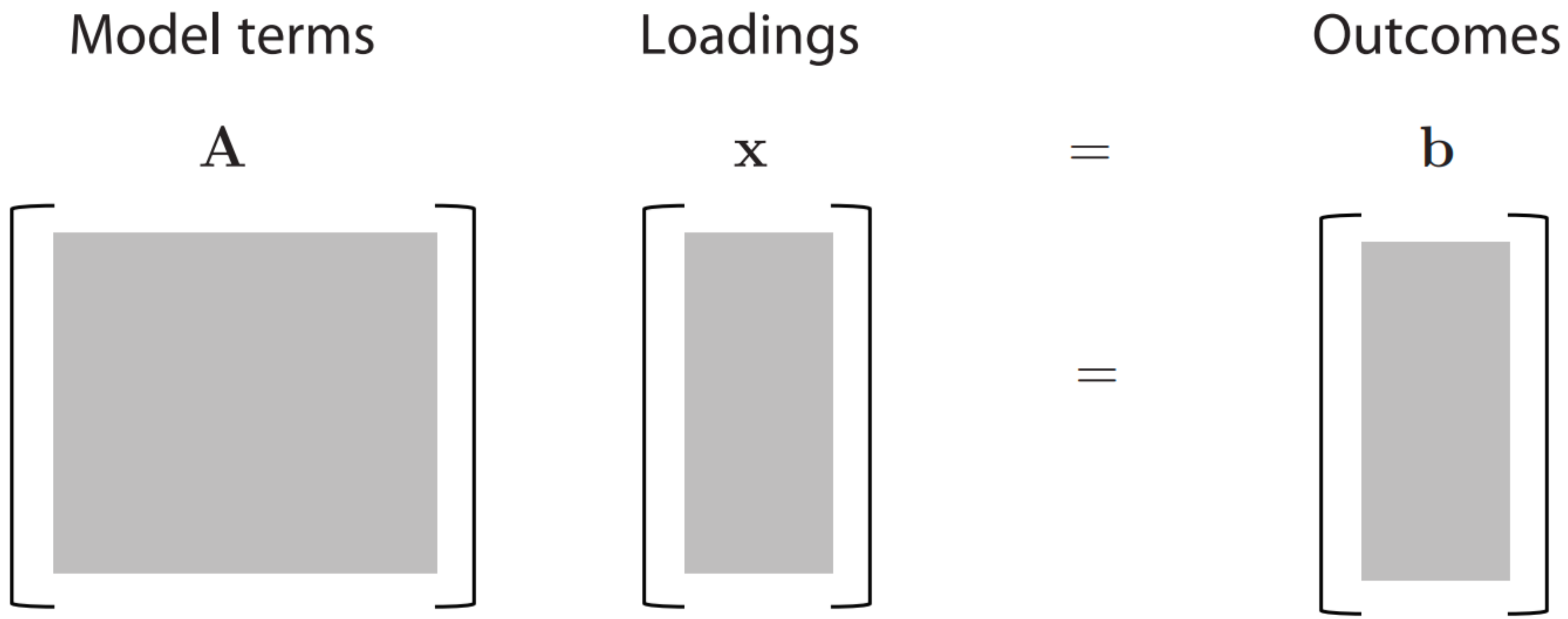
- The loss function is MSE (mean-squares error)

$$E(f) = \frac{1}{n}\sum_{k=1}^{n}(f(x_k) - y_k)^2 = \frac{1}{n}\sum_{k=1}^{n}(\beta_1 x_k + \beta_2 - y_k)^2$$

Linear regression

- The optimization method: derivatives
- Generalization, the 2×2 system:

$$\mathbf{Ax} = \mathbf{b}$$



Contents

- Data science and machine learning
- Machine learning architecture
- **Regression model**
  - Linear regression
  - Non-linear regression
    - Fit Function
    - Gradient descent
  - Over- and Under-Determined Systems
  - Model selection
  - Overfitting

Nonlinear regresstion

- How with nonlinear regresstion? For example:

$$f(x) = \beta_2 \exp(\beta_1 x)$$

- The MSE function:

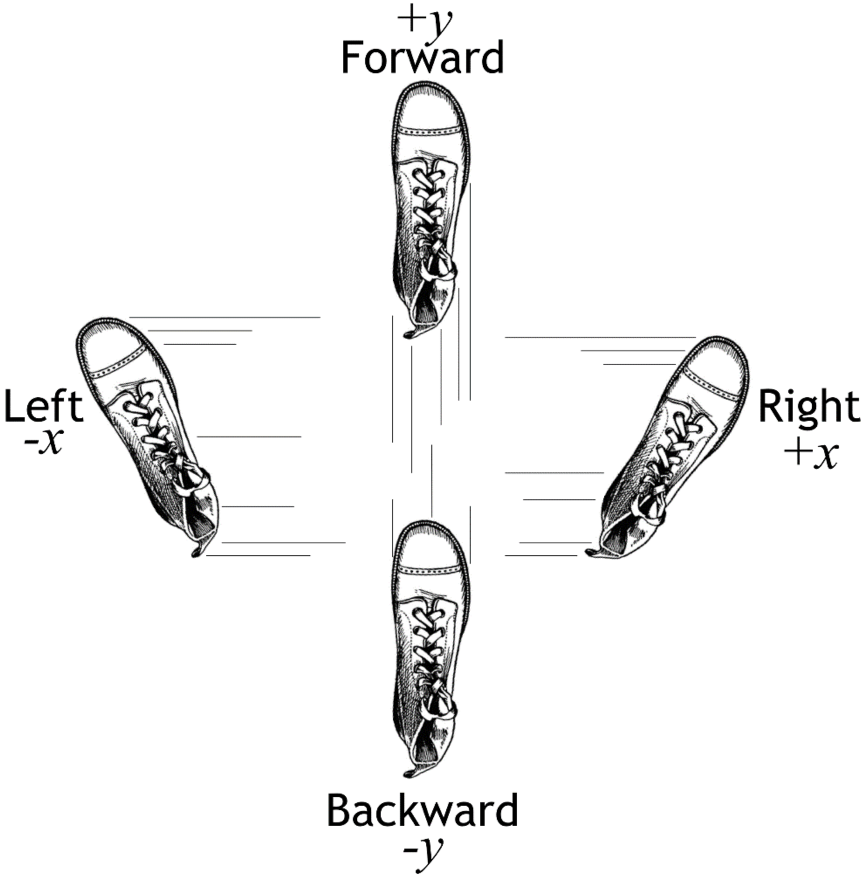
$$E(\beta_1, \beta_2) = \sum_{k=1}^{n} (\beta_2 \exp(\beta_1 x_k) - y_k)^2$$

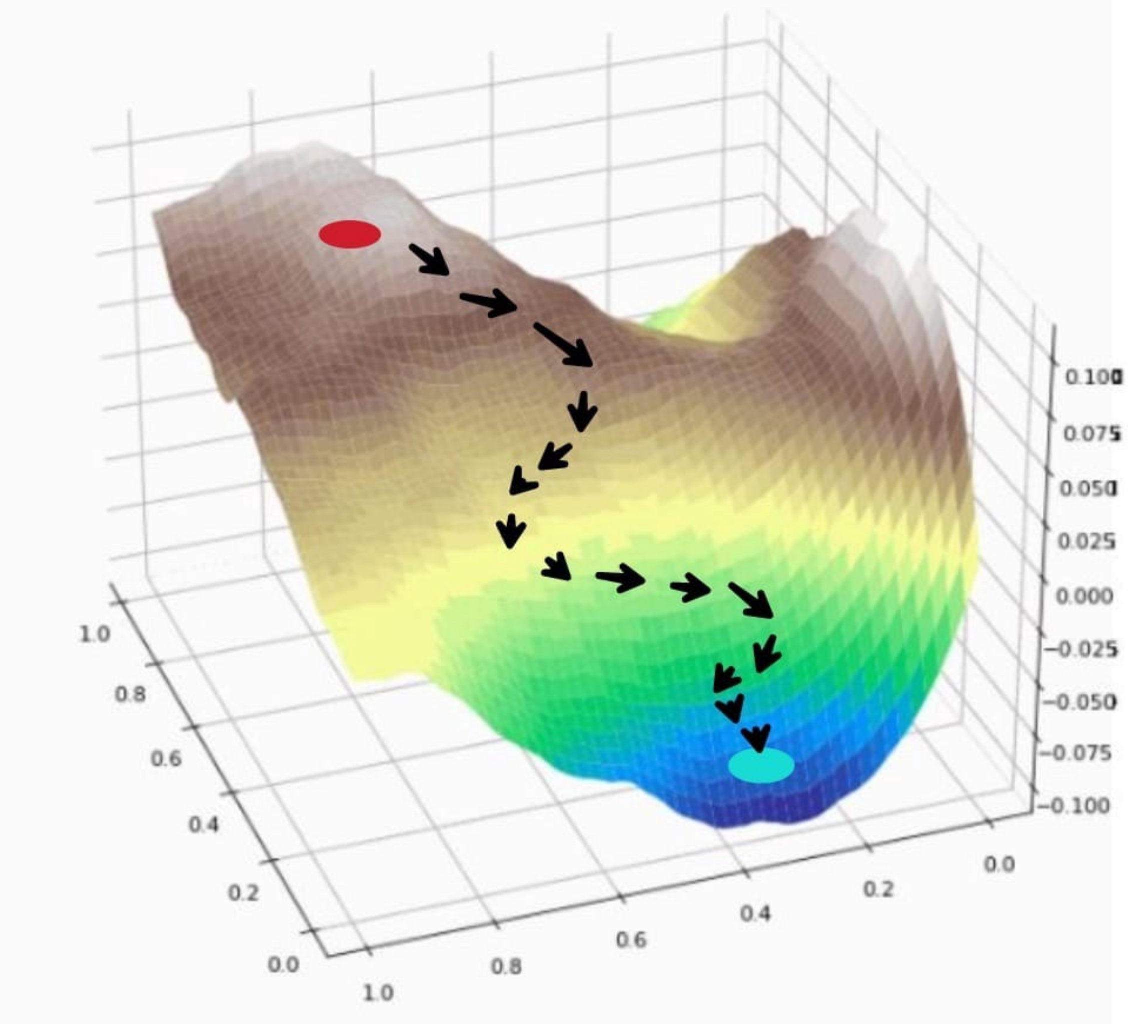
Contents

- Data science and machine learning
- Machine learning architecture
- **Regression model**
  - Linear regression
  - Non-linear regression
    - Fit Function
    - Gradient descent
  - Over- and Under-Determined Systems
  - Model selection
  - Overfitting

Go to downhill

Go to downhill





Go to downhill

- What means if direction vector is:

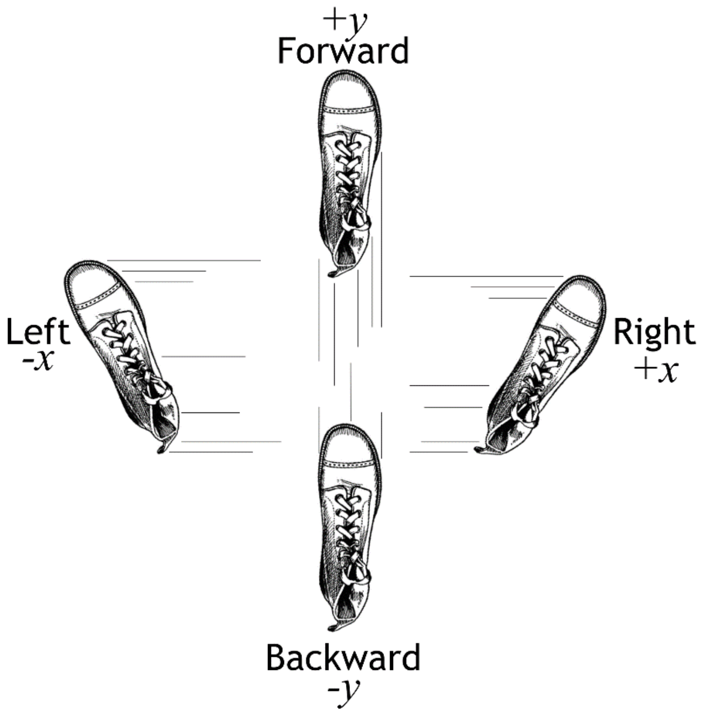
$[x, y]$

$= [which\ way\ is\ down\ in\ x\ direction, which\ way\ is\ down\ in\ y\ direction]$

$= [-1,1]$

- To actually move downhill, we move to:

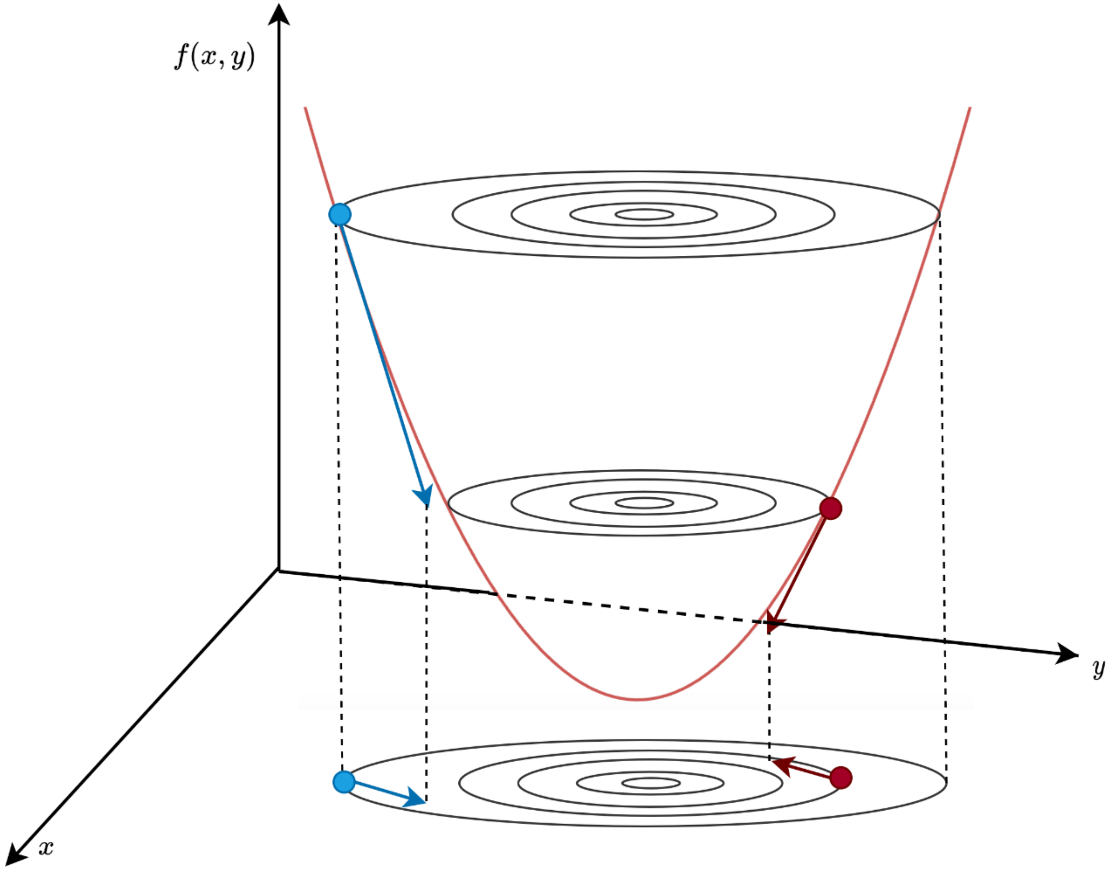
$\Rightarrow [x_{new}, y_{new}] = [x, y] + [-1, 1]$



Go to downhill

- Generally, to move in xy space toward the minimum point, we need identify:
  - Moving direction (increase/descrease x and y)
  - Rate of change (based on slope)

$\Rightarrow$ It is a direction vector

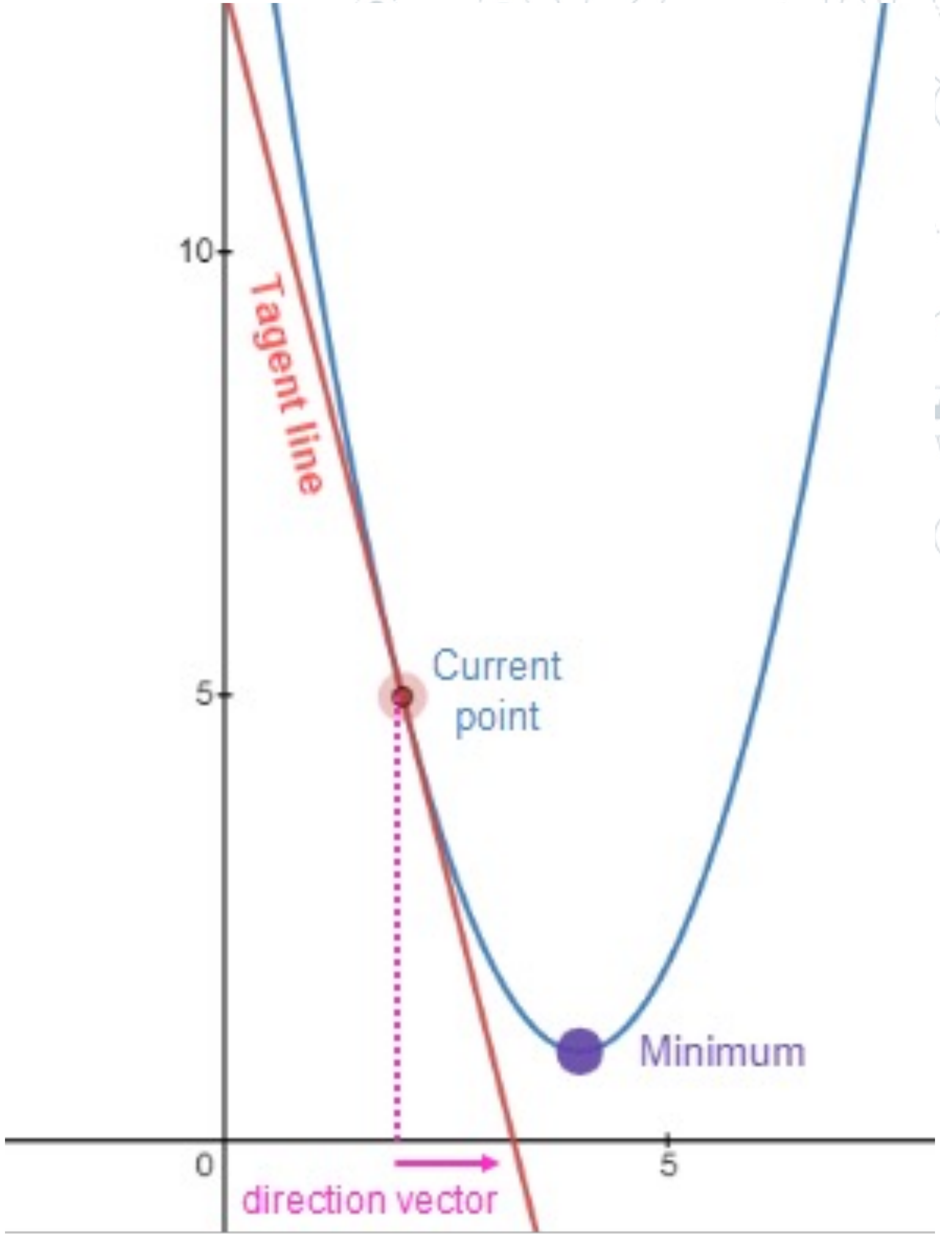


Direction vector

- The derivative of a function at a specific point gives the slope of the tangent line.

$$f'(x) = \lim_{(x_1 - x_0)\to 0} \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

- Why is the tangent line considered as a direction vector?

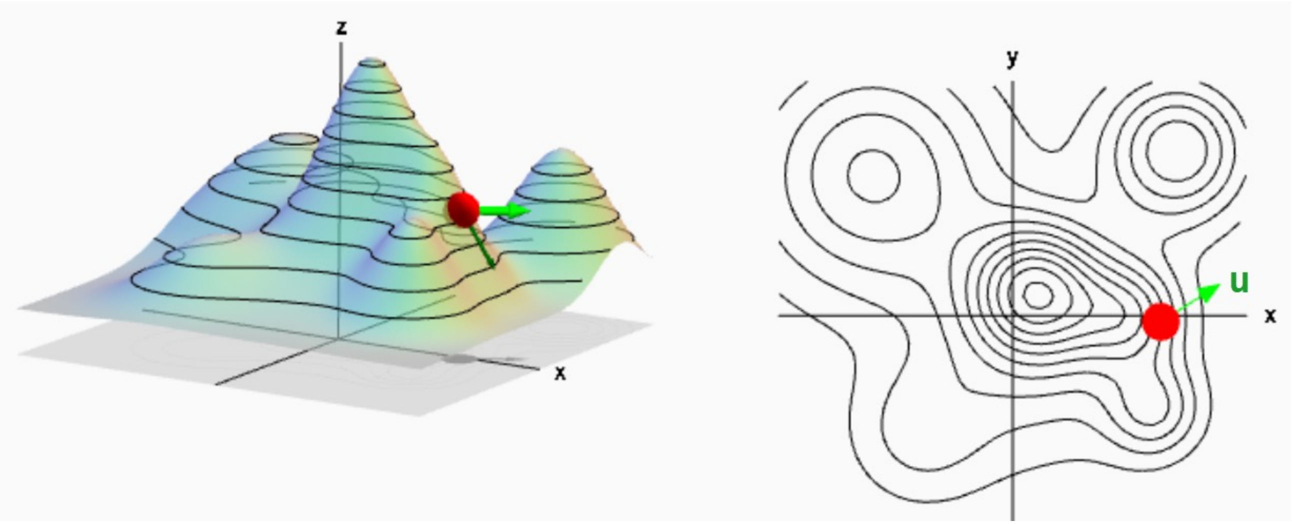


Directional derivative

- If you stand at some point $\mathbf{a} = (x_0, y_0)$, the slope of the ground in front of you will depend on the direction you are facing.
- To calculate the slope in any direction, we derivative in this direction.
  $\Rightarrow$ called the directional derivative.

$$D_{\mathbf{u}} f(x_0, y_0)$$

where $\mathbf{u} = (u_1, u_2)$ is an unit vector that points in the direction in which we want to compute the slope.



Gradient

- The gradient of f at any point tells you:
  - a direction is the steepest from that point with respect to the x,y plane
  - how steep it is (the slope of the hill in that direction)

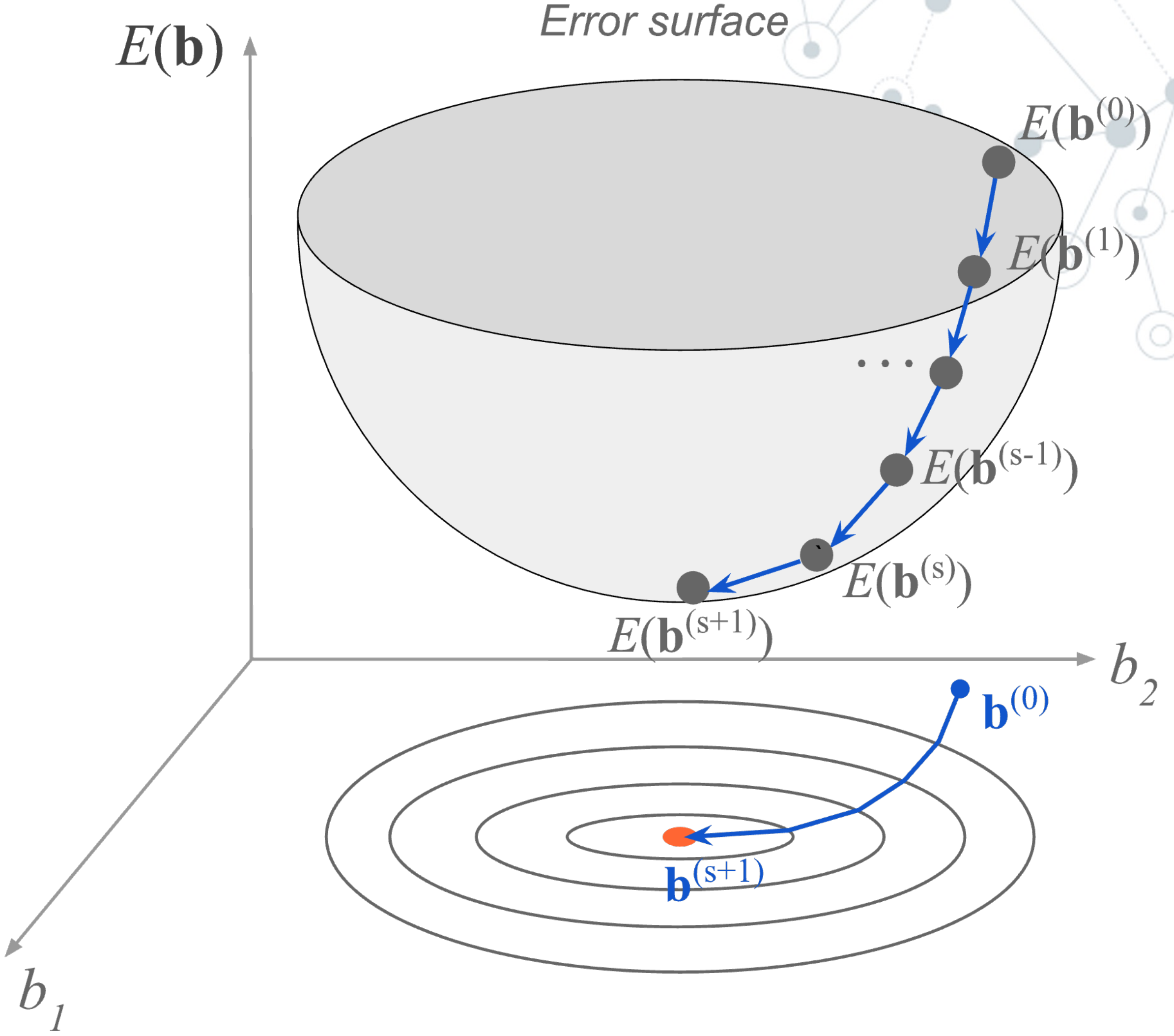
$$\nabla f(x,y) = \begin{bmatrix} \frac{\partial f(x,y)}{\partial x} \\ \frac{\partial f(x,y)}{\partial y} \end{bmatrix} = \frac{\partial f(x,y)}{\partial x}\hat{\mathbf{x}} + \frac{\partial f(x,y)}{\partial y}\hat{\mathbf{y}}$$

- The partial derivatives give the slope in the **positive** x direction and the slope in the **positive** y direction.

Gradient Descent

- As we update, we want the value of $f(x, y)$ to decrease.
  - When it stops decreasing, (x_0, y_0) will have arrived at the position giving the minimum value of $f(x, y)$.
- The next position at time step t:

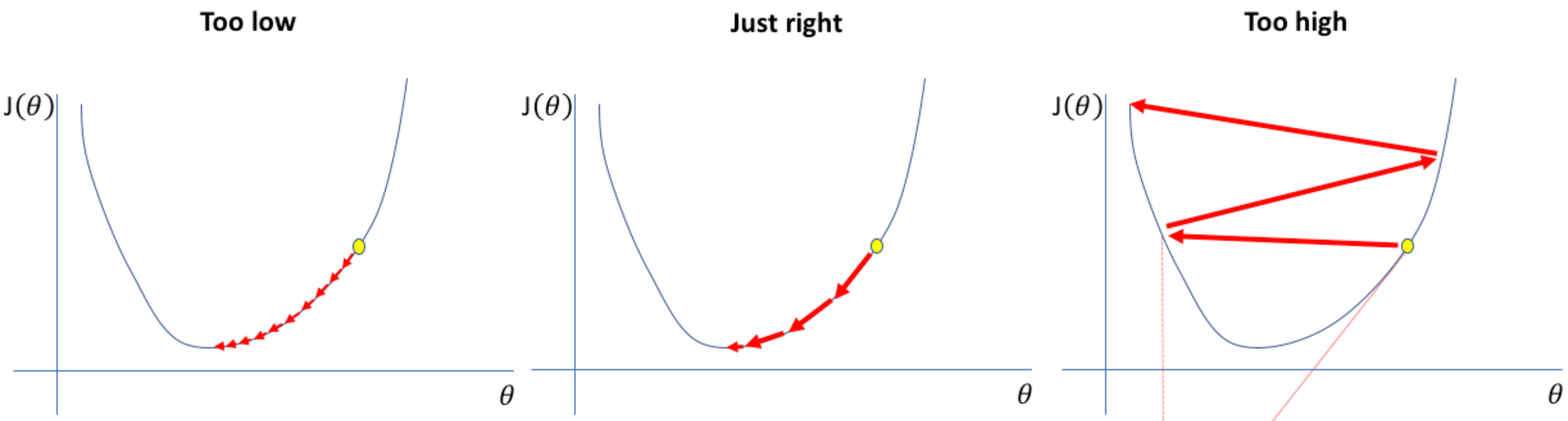
$$\mathbf{x}_{t+1} = \mathbf{x}_t - \nabla f(\mathbf{x}_t)$$



Issues: Learning rate

- Need to restrict the size of the steps by shrinking the direction vector using a learning rate η, whose value is less than 1:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \nabla f(\mathbf{x}_t)$$

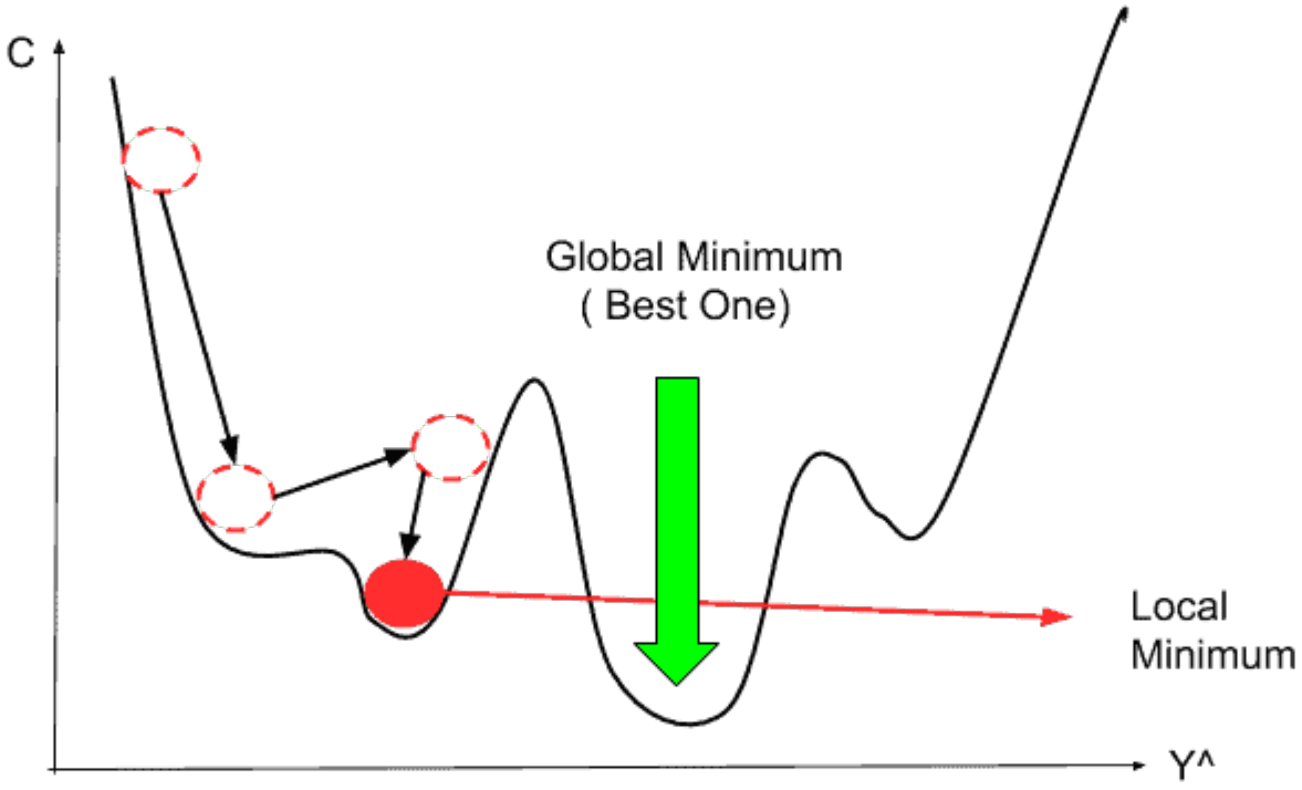


A small learning rate requires many updates before reaching the minimum point

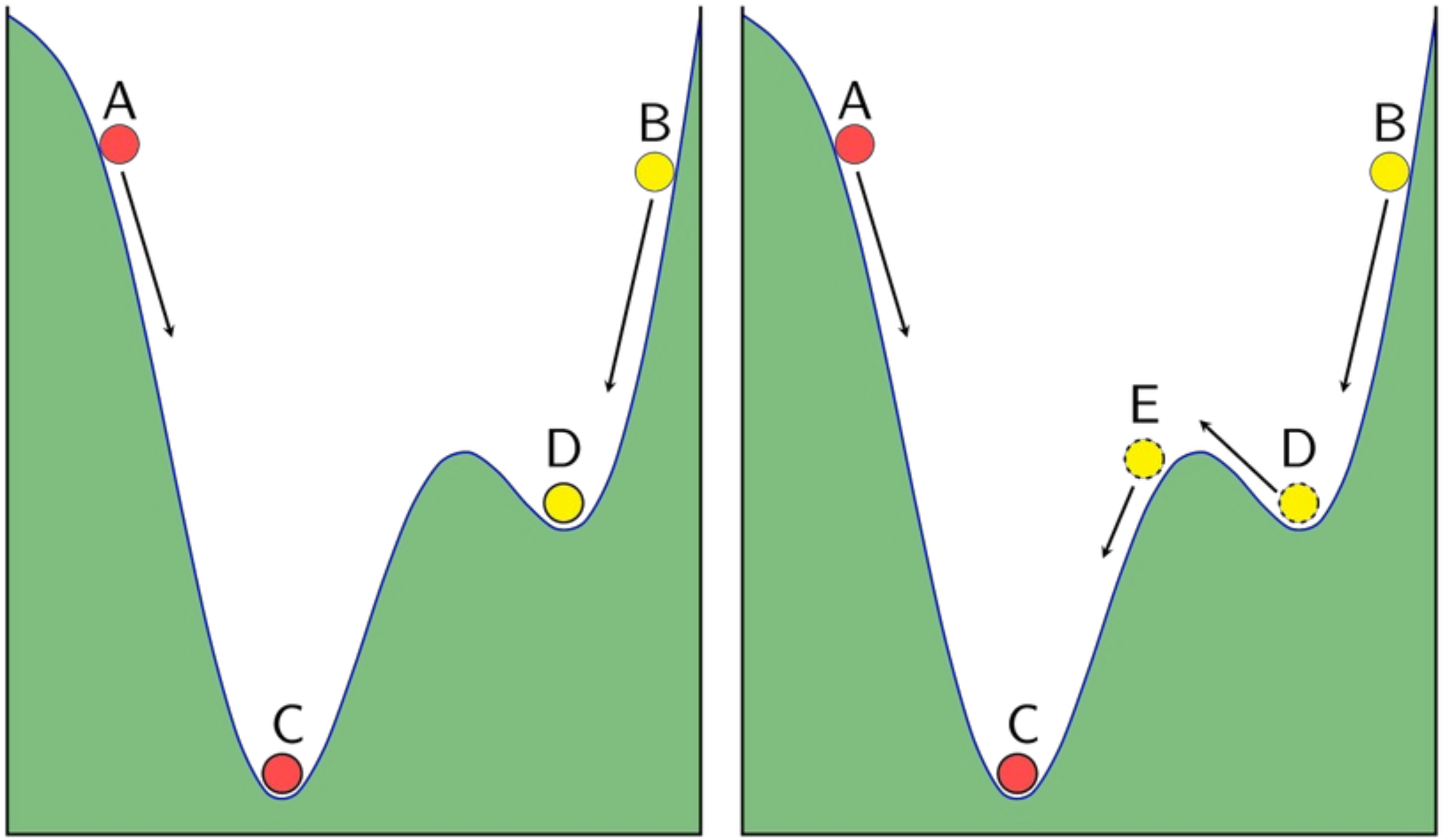
The optimal learning rate swiftly reaches the minimum point

Too large of a learning rate causes drastic updates which lead to divergent behaviors

Issues: Starting point (non-linear function)



Momentum



b) GD

c) GD with momentum

Summary for nonlinear regression

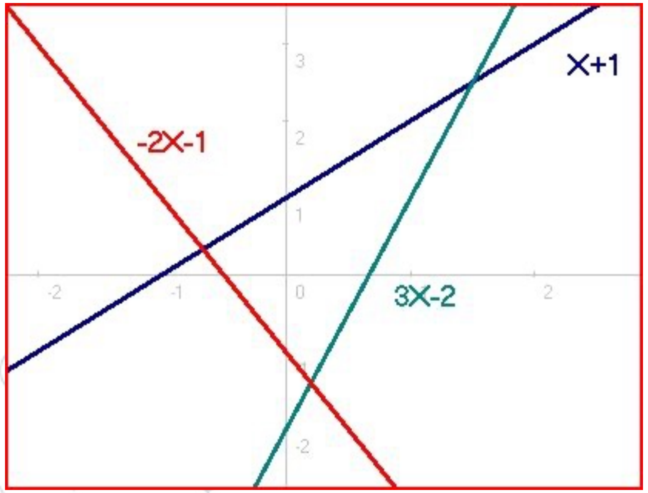
- The nonlinear optimization procedure:
  - The initial guess
  - Step size η
  - Computing the gradient efficiently

Contents

- Data science and machine learning
- Machine learning architecture
- **Regression model**
  - Linear regression
  - Non-linear regression
  - Over- and Under-Determined Systems
  - Model selection
  - Overfitting

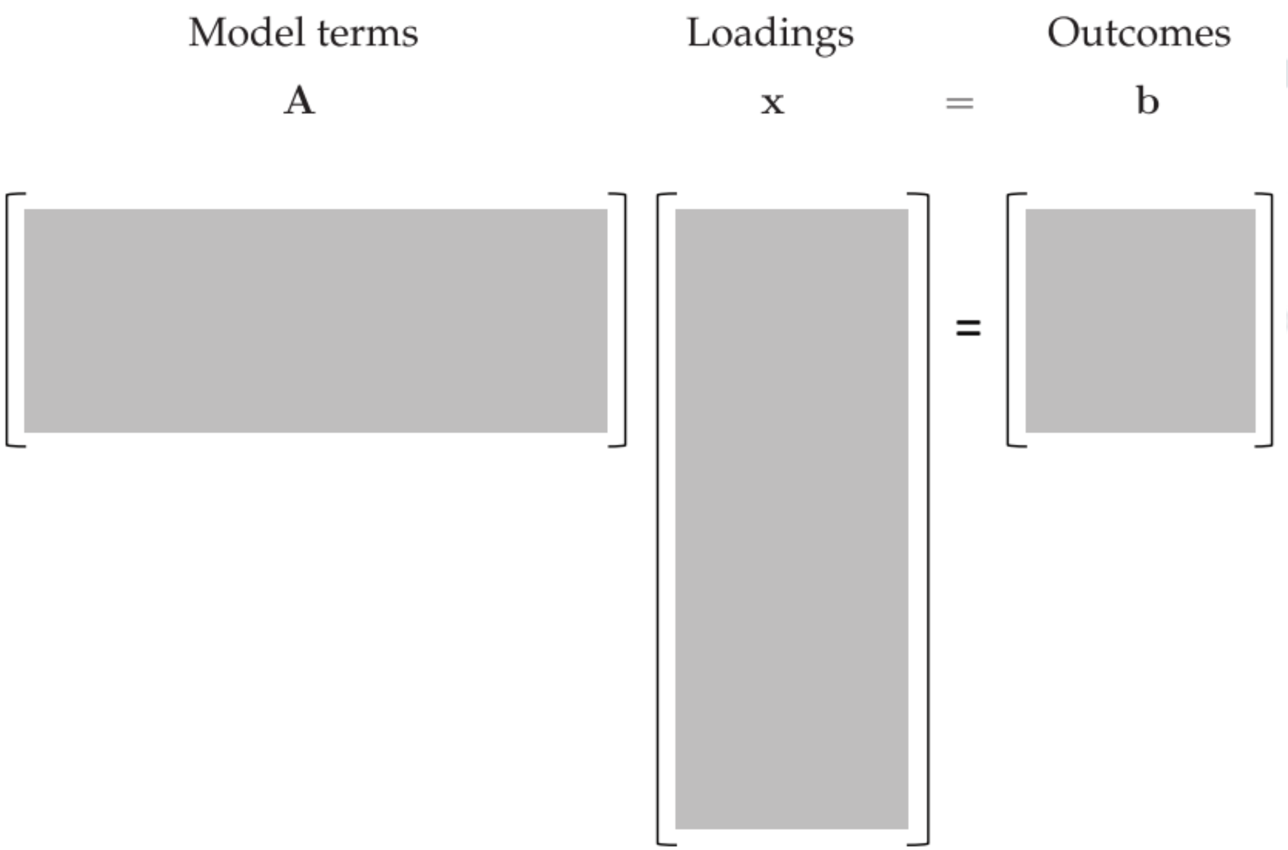
Over-determined systems

- Over-determined systems have more constraints (equations) than unknown variables.
  - No solutions satisfying the linear system.
  - Approximate solutions to minimize a given error.



Under-Determined Systems

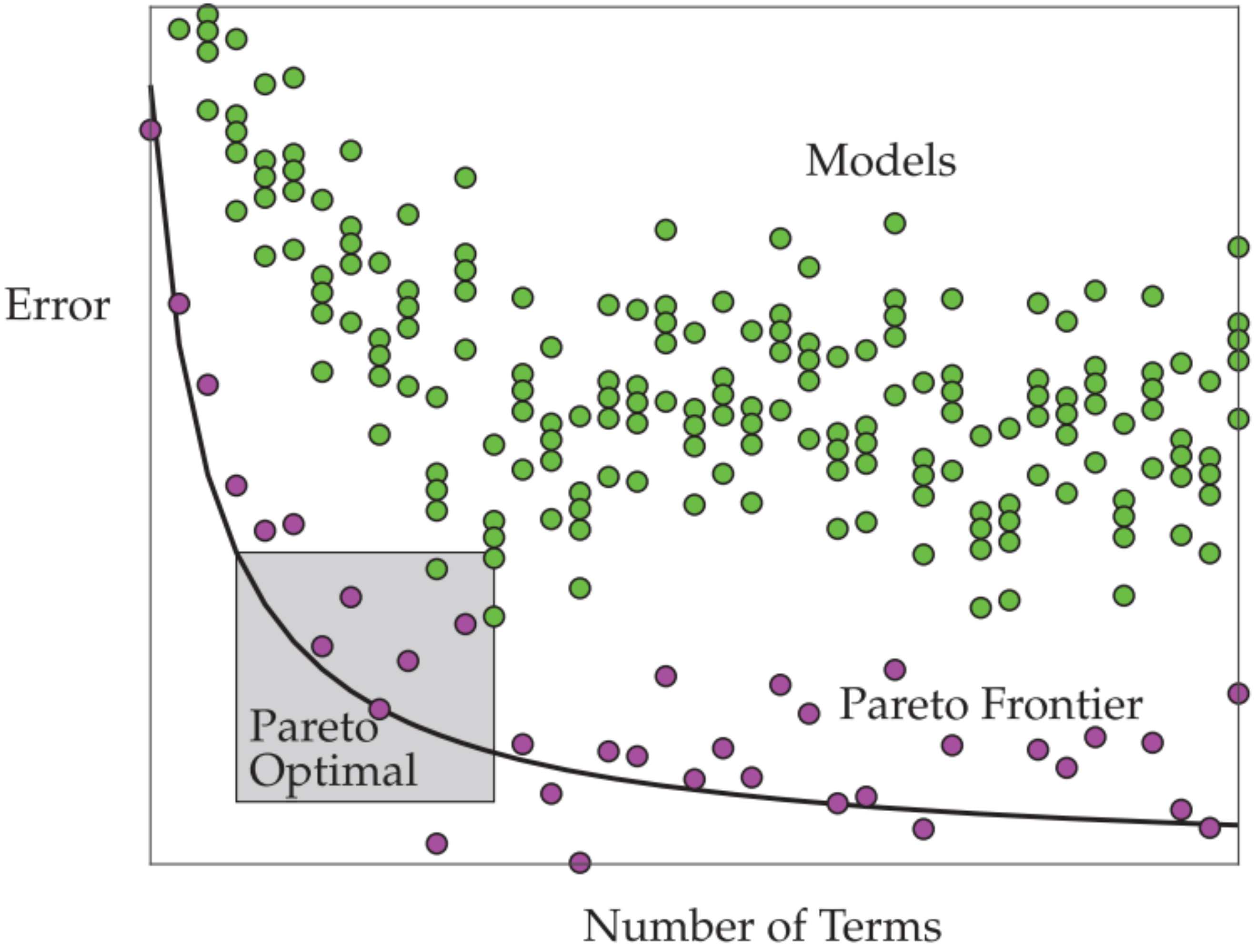
- Under-determined systems have more unknowns than constraints.
  - an infinite number of solutions.
  - some choice of constraint must be made.



Contents

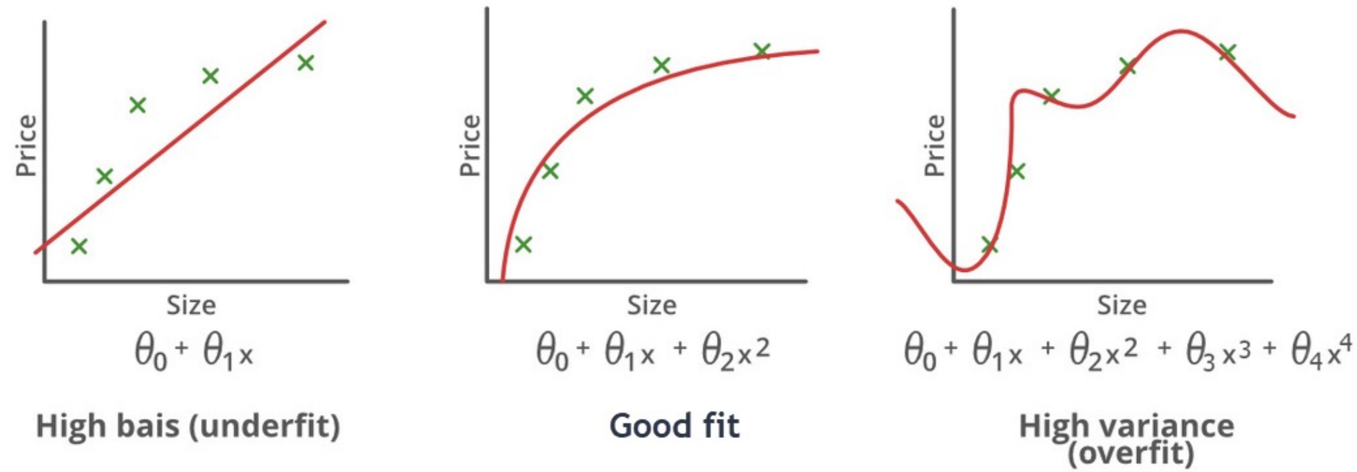
Model Selection

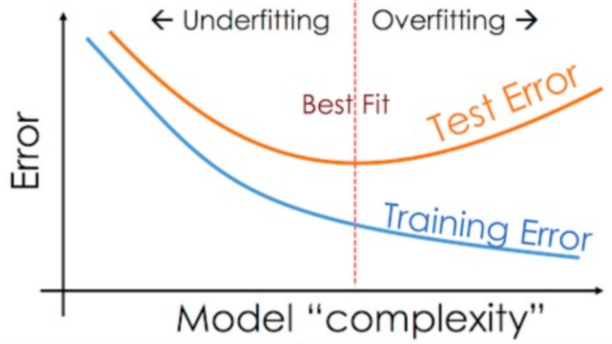
- Model selection is not simply about reducing error, it is about producing a model that has a high degree of interpretability, generalization and predictive capabilities.



Overfitting

- The production is too closely to a particular set of data, and may therefore fail to fit to predict future observations reliably.
  - Overfitting does not allow for generalization.





The End