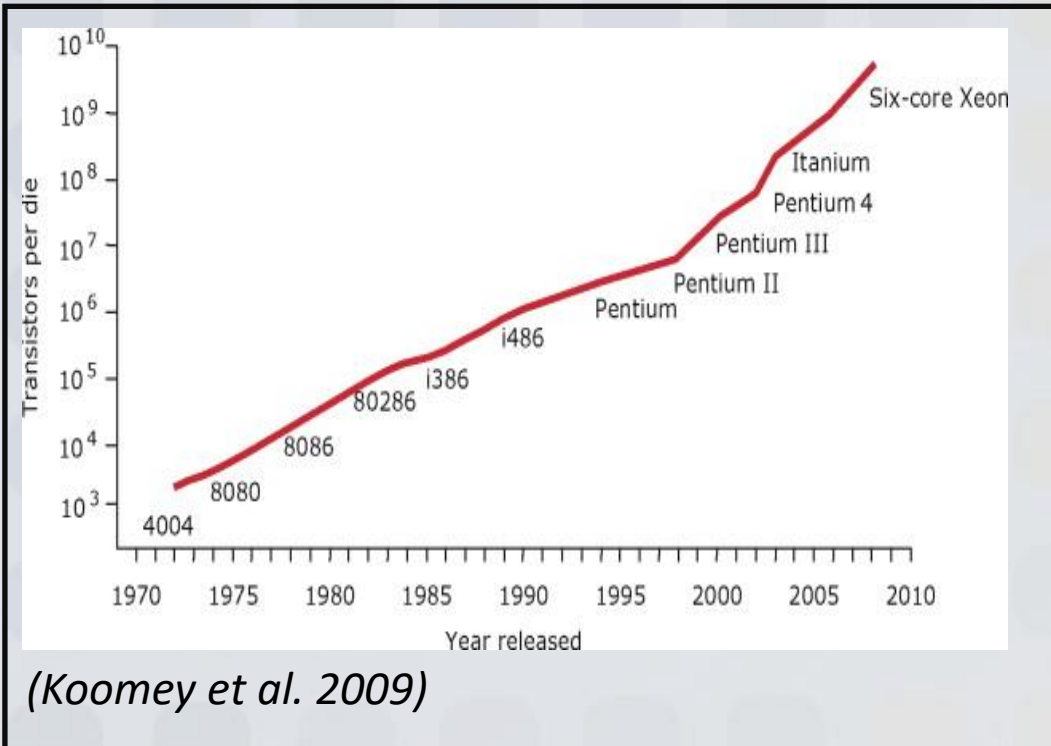


# Introduction to Machine Learning

Dr Mykola Gordovskyy

- What is machine learning?
- ... and what is it for?
- Types of machine learning
  - Linear regression
  - Multi-linear regressions
  - Polynomial regression
- Choosing the optimal model

# ~~Why~~ What is machine learning?



In the last 25 years (widely-)available computational power has increased by factor of 100-1000

(My own observation)

Moore's law: the number of transistors in a dense integrated circuit (IC) doubles about every two years

(G.Moore, Fairchild semiconductors)

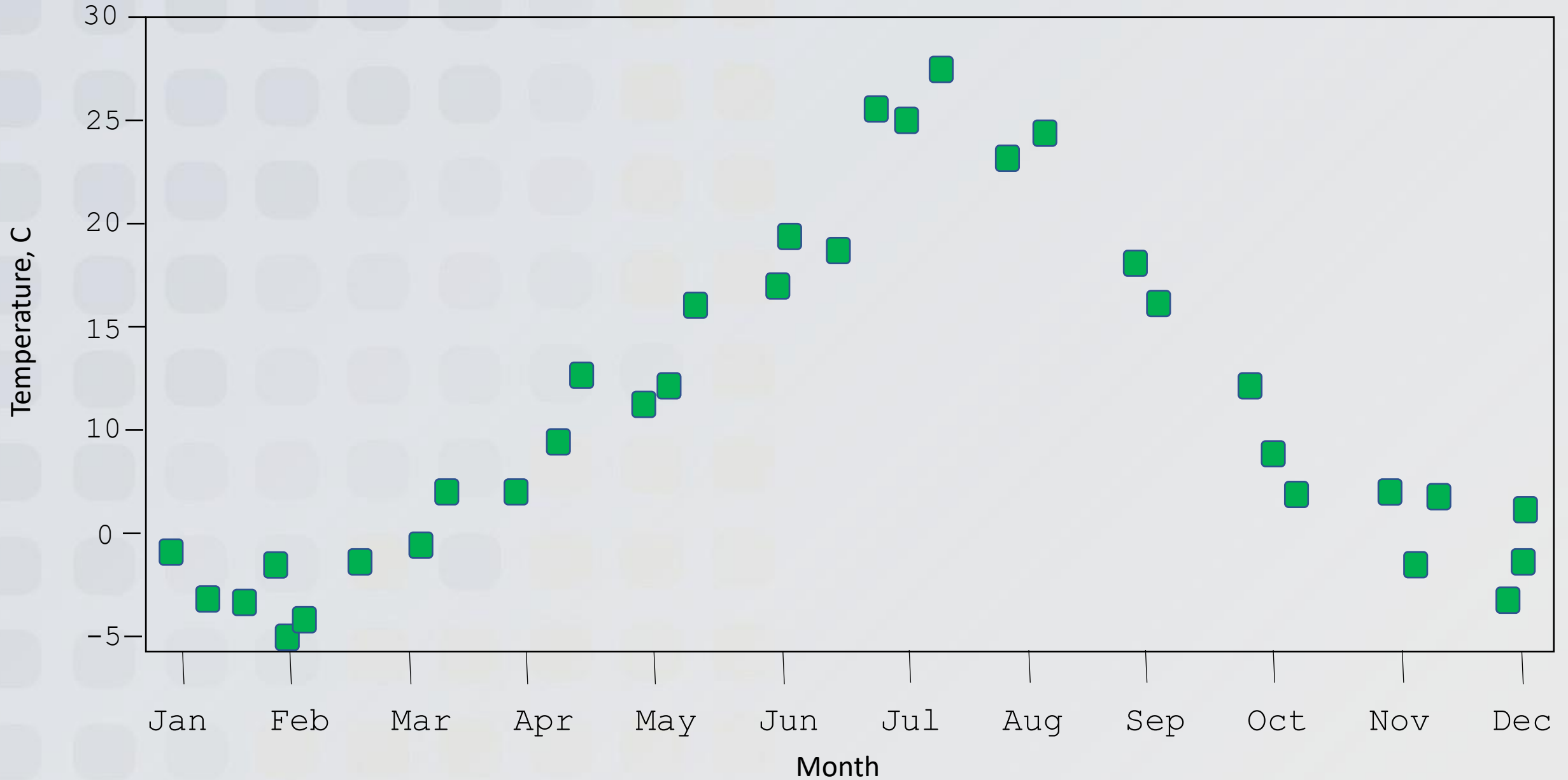
NB: Its not a law, just an observation

Why

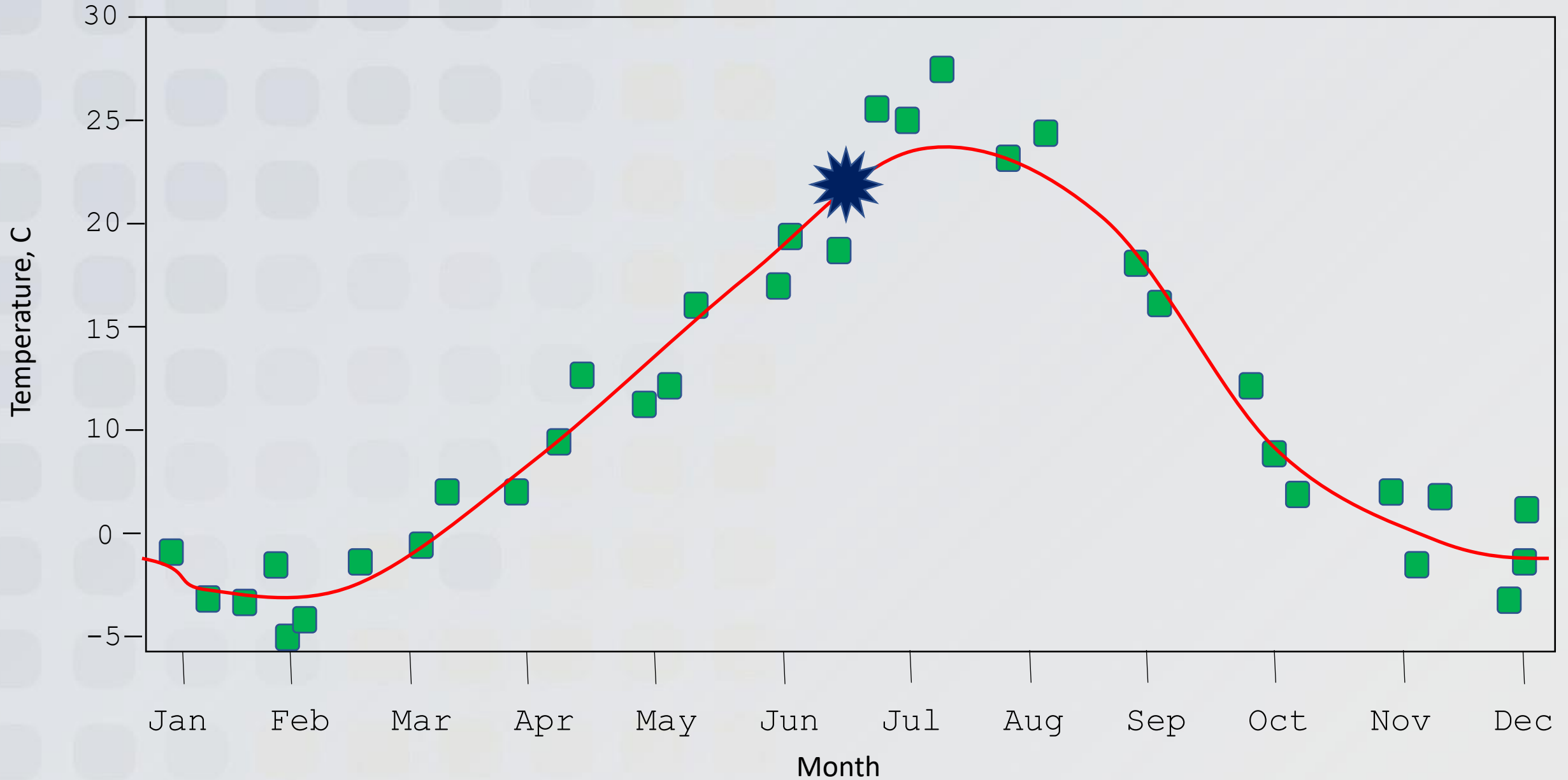
# What is machine learning?

Perhaps, the key reason behind the fast growth of ML use is that our computational powers increase faster than our advance in understanding the physics of things....

# What is machine learning? (example 1)



# What is machine learning? (example 1)



# What is machine learning? (example 2)

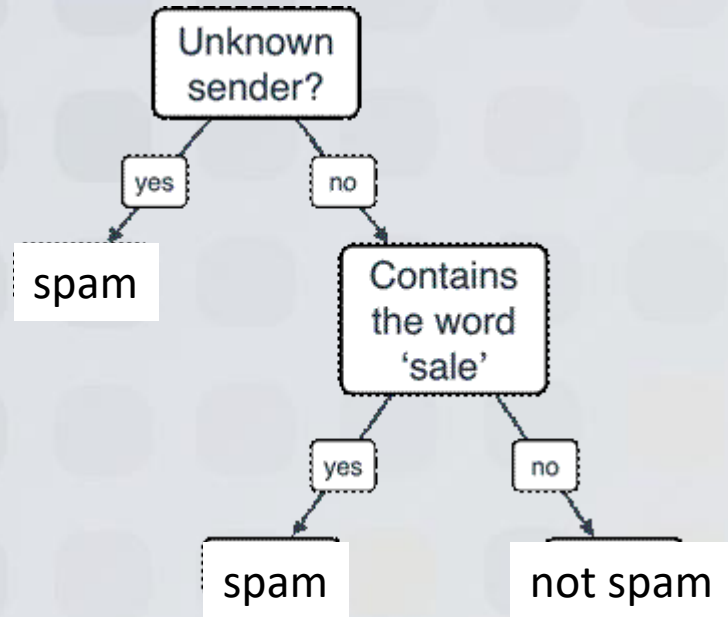


# What is machine learning? (example 3)

## Spam filter

*We can...*

...create the decision tree ourselves



'Rigid' 'one-fits-all' system will falsely classify some emails as spam

Spammers and scammers will learn how to get though the filter

...or 'factorise' emails, and use lots of examples of spam and not-spam emails to create a 'model'/decision tree for email classification

Where do we get examples? – User classifies emails



# What is machine learning?

Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions *without being explicitly programmed to do so*

(A. Samuel)

Machine learning is a branch of *artificial intelligence* and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.

(IBM website)

Machine learning is *a branch of artificial intelligence* that systematically applies algorithms to synthesize the underlying relationships among data and information.

(M. Awad & R. Khanna “Efficient learning machines”)

# What is machine learning? (naïve, childish definition)

Machine learning =

- *we have some data that we understand,*
- *we assume that this data represents the data we don't understand*
- *we use data that we understand to create a model*
- *and then we use this model to understand the data we don't understand*

# What about AI?

## **Human approach:**

- Systems that think like humans
- Systems that act like humans

## **Ideal approach:**

- Systems that think rationally
- Systems that act rationally

*(S. Russel & P. Norvig, IBM)*

**Artificial intelligence is intelligence demonstrated by machines, as opposed to the natural intelligence displayed by animals and humans.**

*(Wikipedia)*

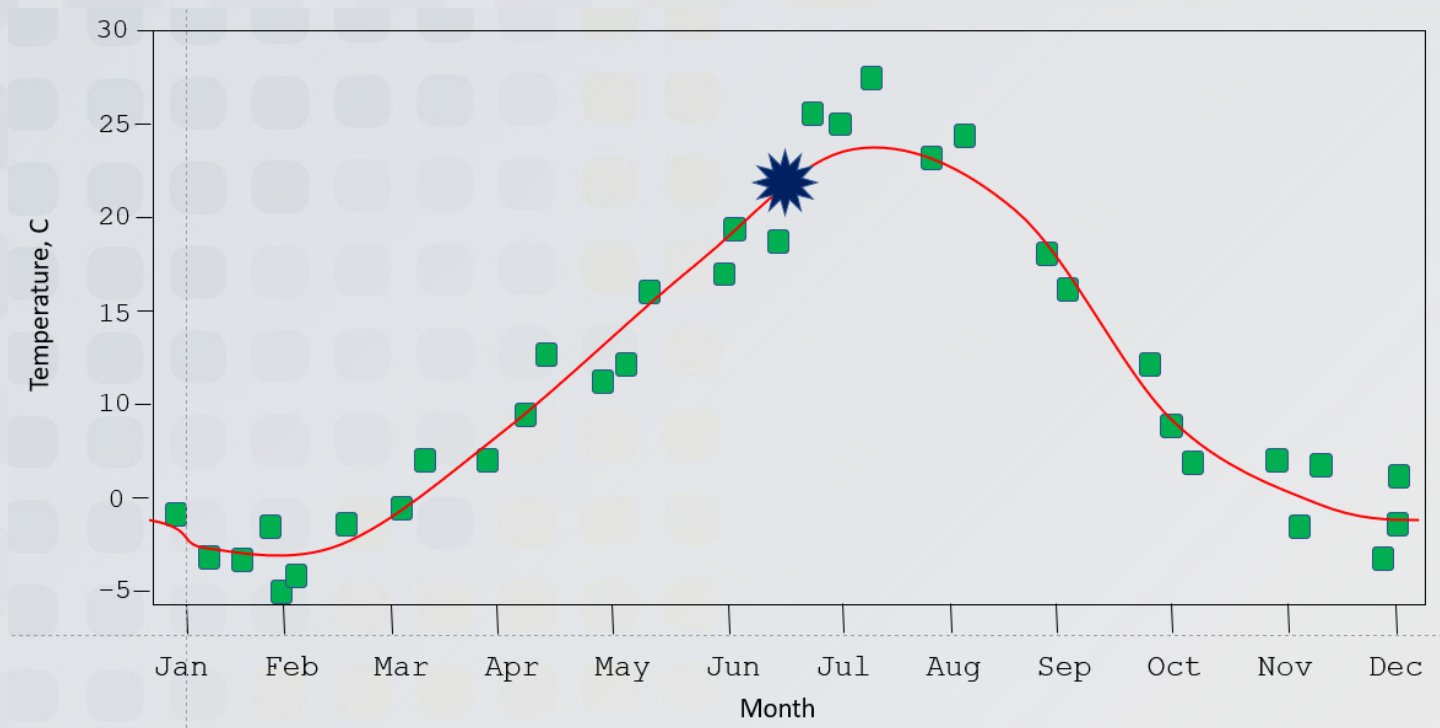
**Can machine replace the question by another, which is closely related to it and is expressed in relatively unambiguous words? (Turing test)**

*(A. Turing)*

***Machine learning is an essential component of AI***

# Benefits v. drawbacks

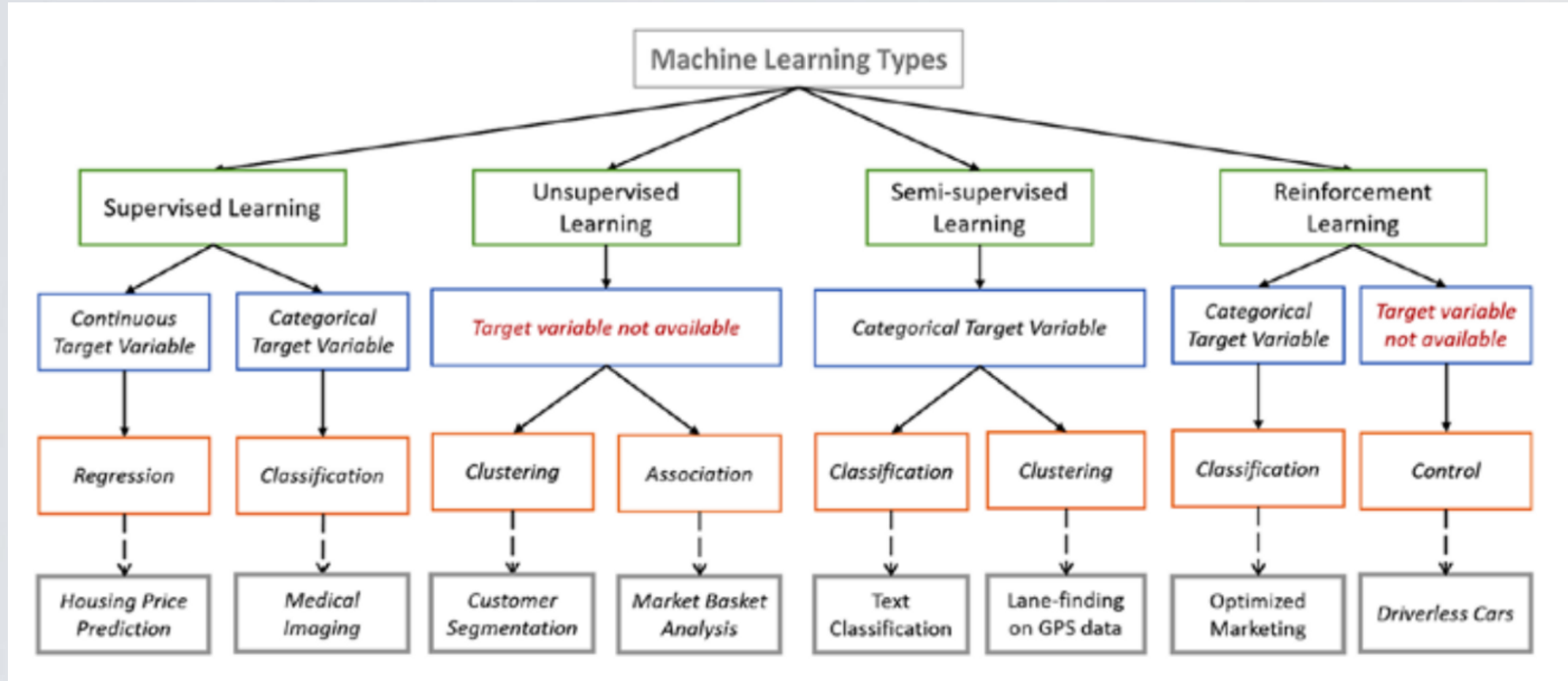
*With the fast increase in available computational power, it might be easier to use purely mathematical methods to do forecasting, without actual understanding how things work. This is both a benefit and a drawback...*



# Benefits v. drawbacks

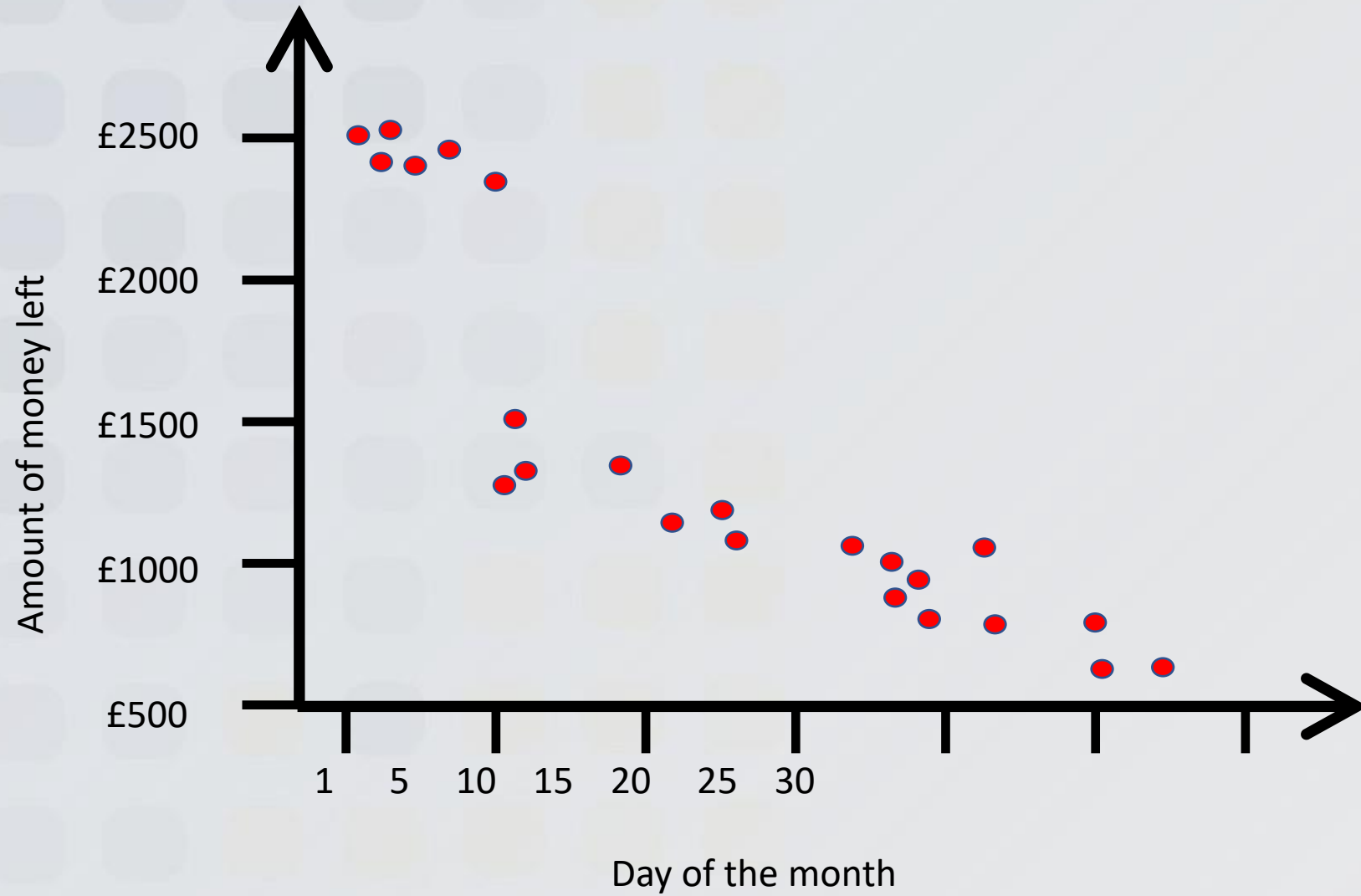
*With the fast increase in available computational power, cheap electronics, machine learning and other elements of AI become ubiquitous in the society. They are widely used in many areas, including education (e.g. AI can be used to chose an optimal teaching approach), HR management (e.g. AI can used to rank job applicants), security (e.g. AI can be used for fast face recognition or detect 'abnormal behaviour')*

# ML diagram

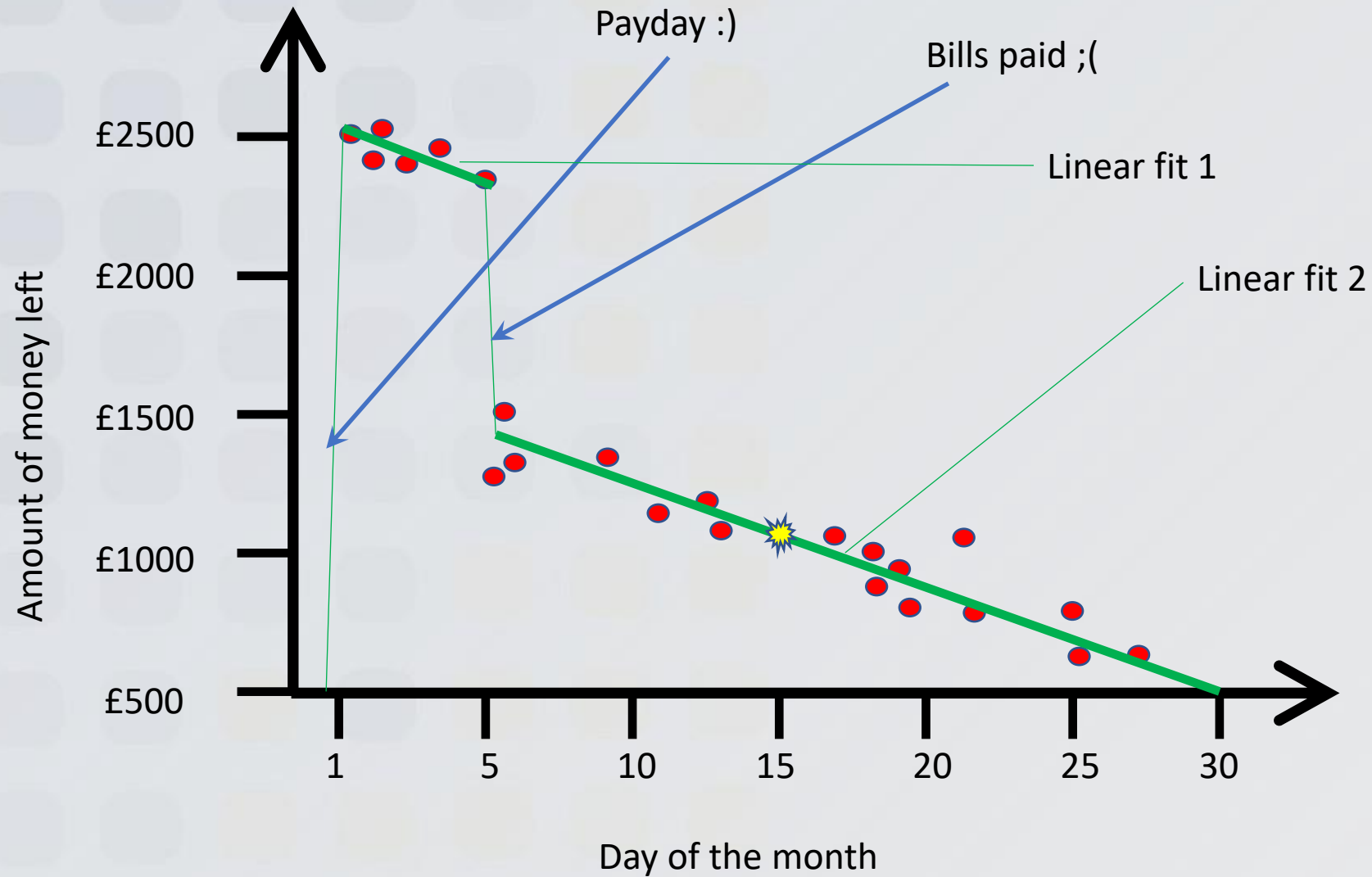


(from Sengupta et al. 2020)

## Another realistic scenario...



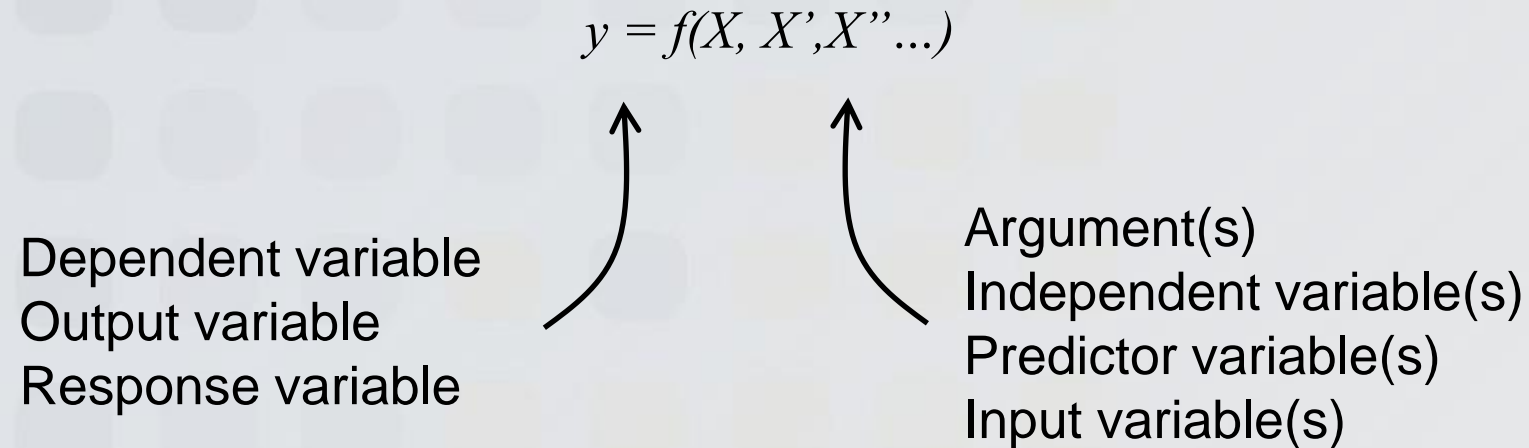
# Another realistic scenario...





# Regression as ML approach

- Regression can be used to estimate the relationship between two or more variables so that we can gain information about (or predict) one of them through knowing values of the other
- The idea is, given a relational database of values  $X, X', X'' \dots Y$ , to find an analytical function  $y = f(X, X', X'' \dots)$ , such that  $y$  is a good approximation of  $Y$



# Linear regression

- Most used type of regression, a powerful tool...
- Linear regression can be used if
  - we expect two variables in our data to be linearly dependent (e.g. if we expect X and Y to be related as

$$Y = a + bX$$

or

- we have no better idea and a linear function gives an acceptable approximation to our data

# Regression

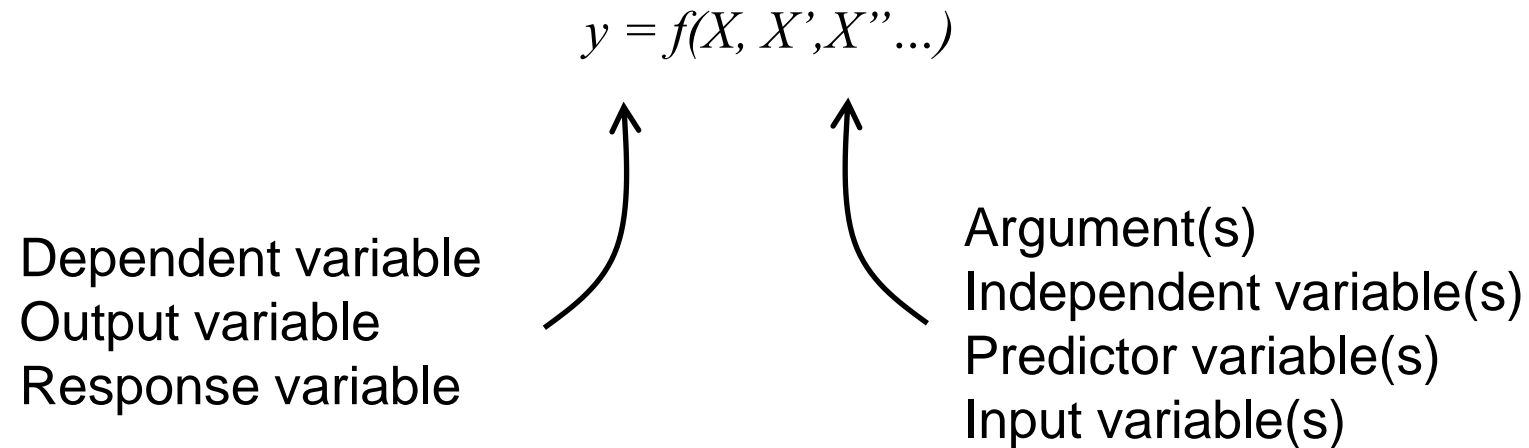
- Instead of using the tabulated data directly (e.g. for differentiation, integration etc), we can first fit an analytical function to our data, and then use that function (for differentiation, integration etc, respectively)
- Benefits: (a) regularisation (getting rid of random noise), (b) use as a forecasting model, (c) 'analytical' use etc

# Regression

- Instead of using the tabulated data directly (e.g. for differentiation, integration etc), we can first fit an analytical function to our data, and then use that function (for differentiation, integration etc, respectively)
- Benefits: (a) regularisation (getting rid of random noise), (b) use as a forecasting model, (c) 'analytical' use etc

# Regression

- Regression can be used to estimate the relationship between two or more variables so that we can gain information about (or predict) one of them through knowing values of the other
- The idea is, given a relational database of values  $X, X', X'' \dots Y$ , to find an analytical function  $y = f(X, X', X'' \dots)$ , such that  $y$  is a good approximation of  $Y$



# Linear regression

- Most used type of regression, a powerful tool...
- Linear regression can be used if
  - we expect two variables in our data to be linearly dependent (e.g. if we expect X and Y to be related as

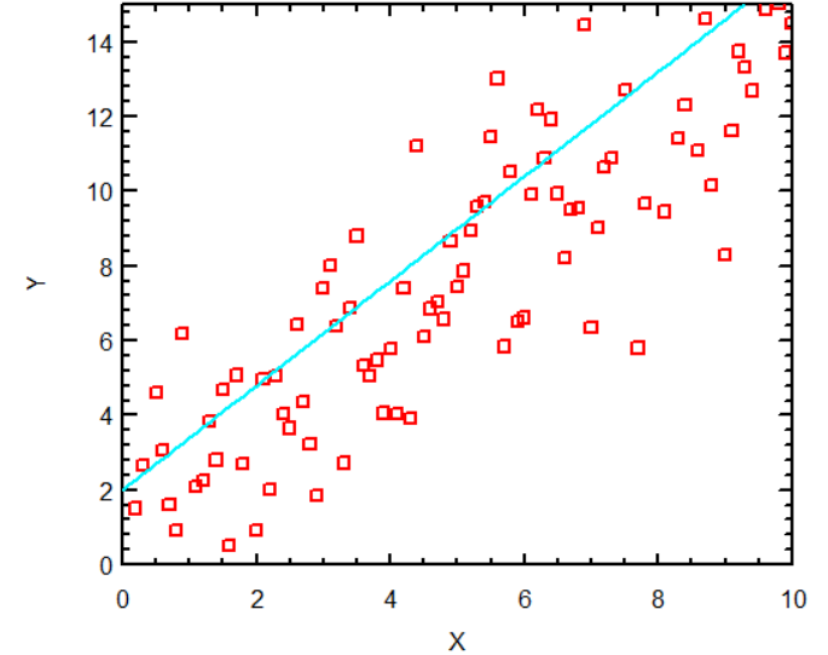
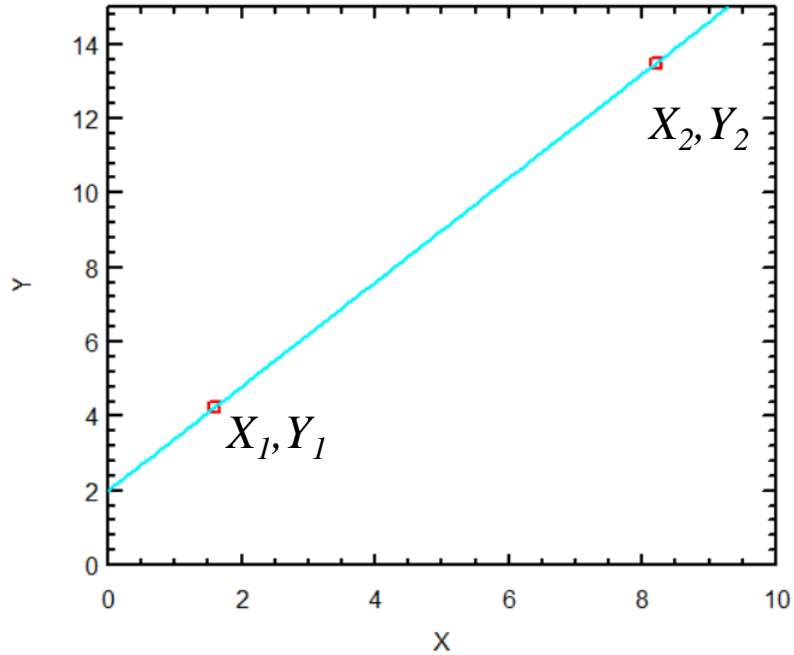
$$Y = a + bX$$

or

- we have no better idea and a linear function gives an acceptable approximation to our data

# Linear regression

Data:  $X_i, Y_i$



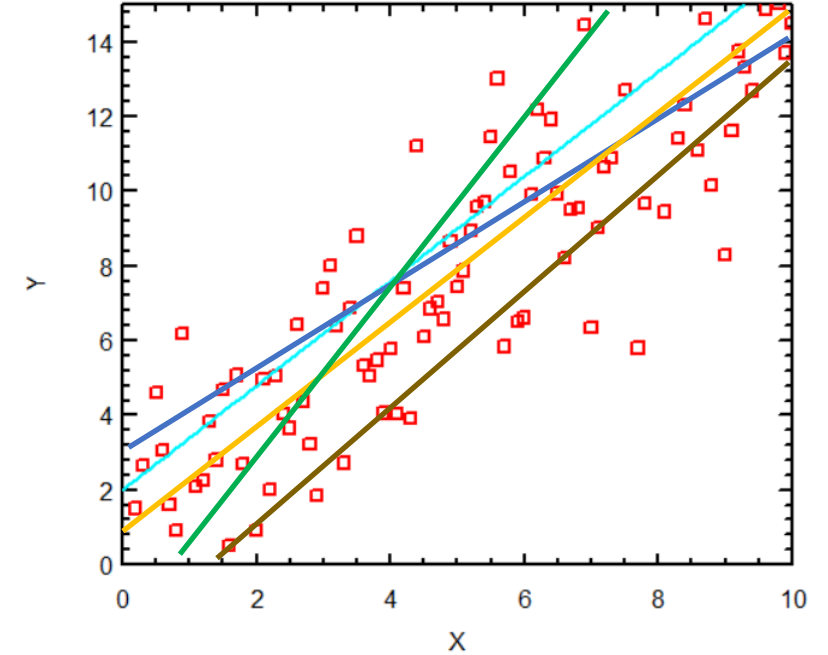
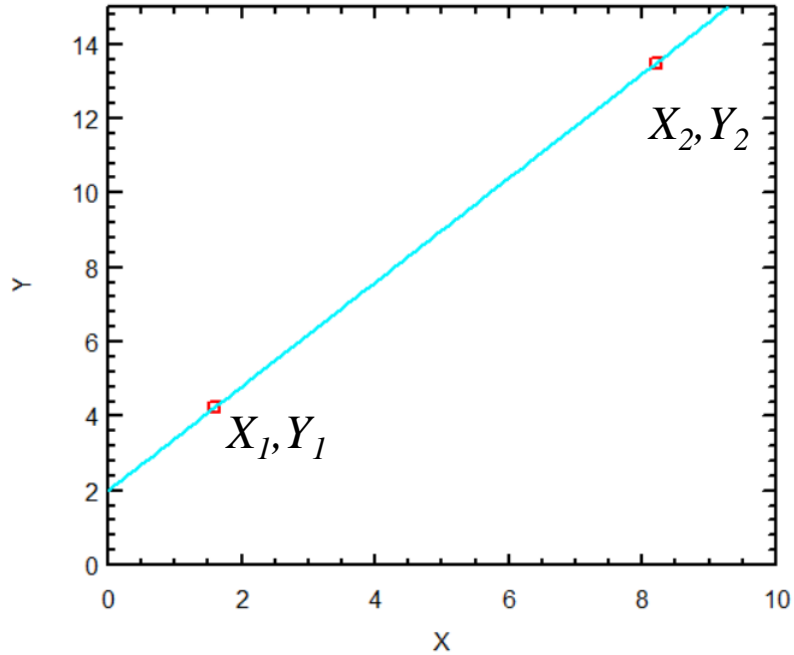
Linear fit  $Y = a + bX$  can be easily found  
because two points can always be  
connected by a line

$$\begin{cases} Y_1 + bX_1 = a \\ Y_2 + bX_2 = a \end{cases}$$

?

# Linear regression

Data:  $X_i, Y_i$



Linear fit  $Y = a + bX$  can be easily found because two points can always be connected by a line

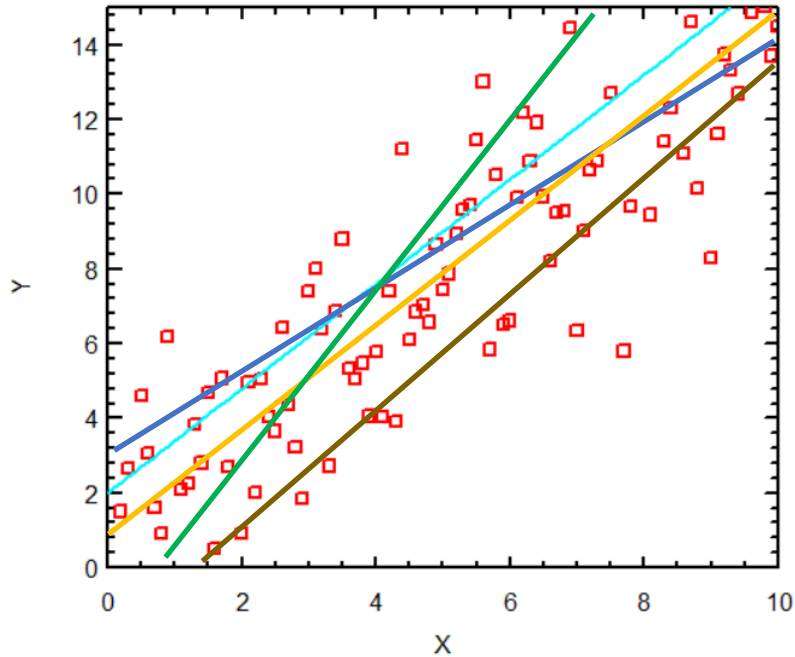
$$\begin{cases} Y_1 + bX_1 = a \\ Y_2 + bX_2 = a \end{cases}$$

?



# Linear regression

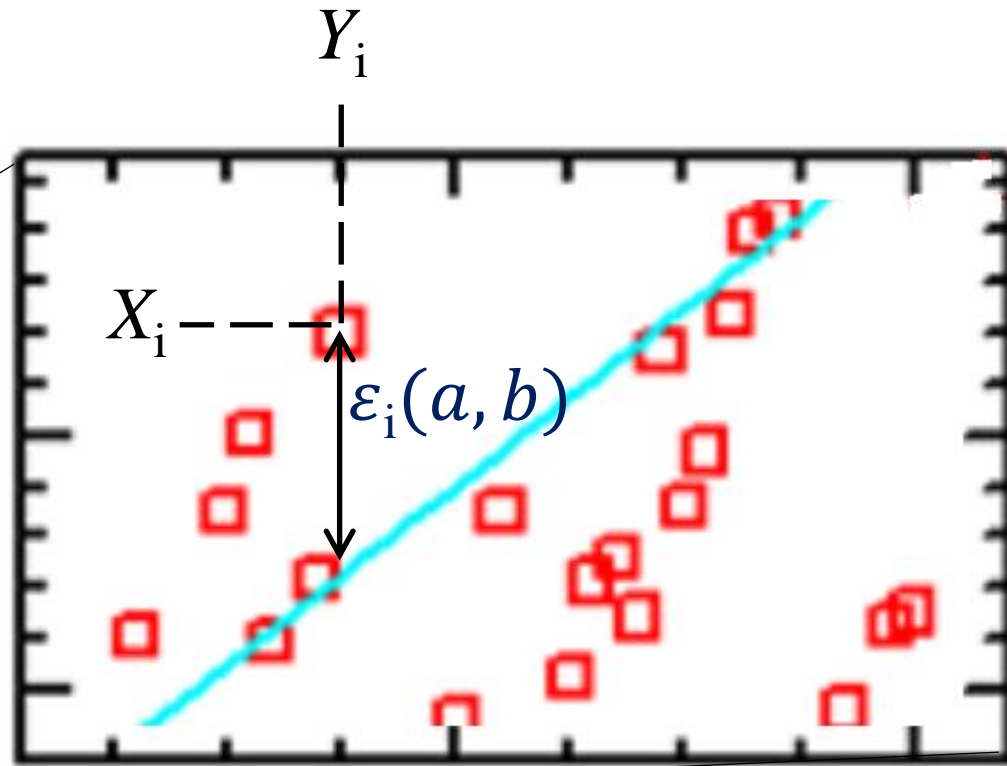
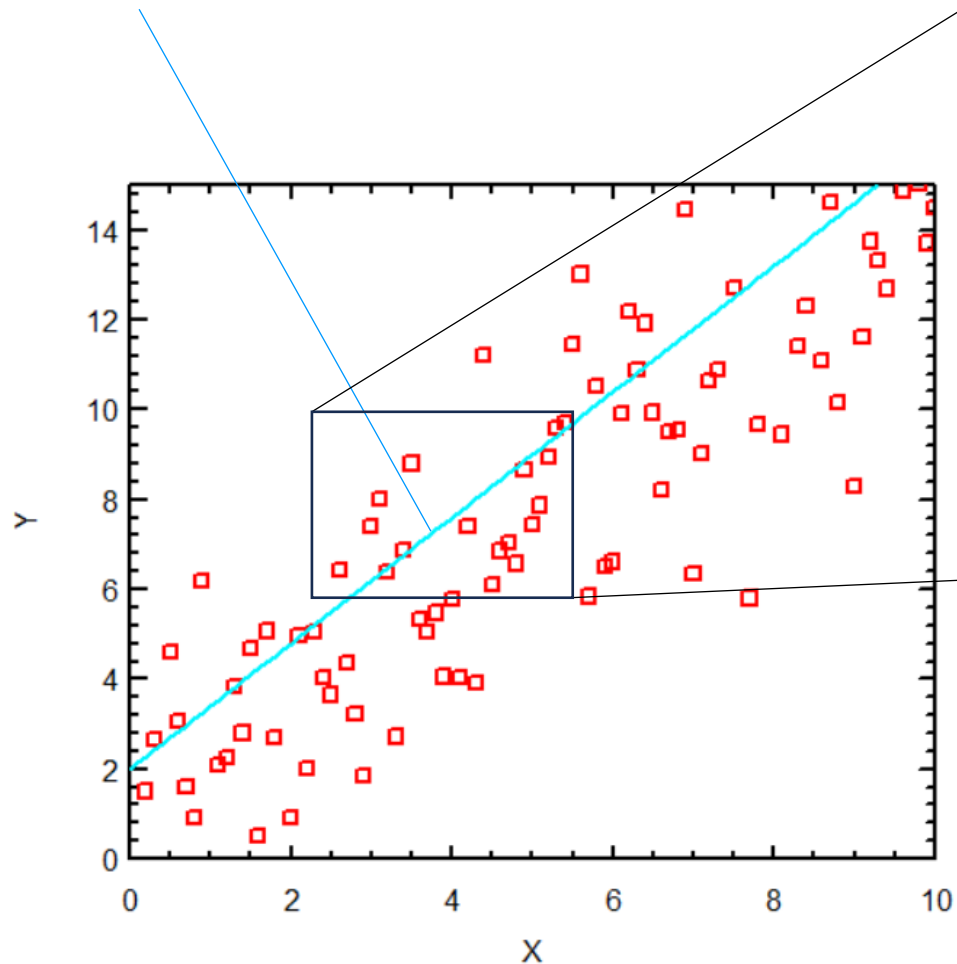
Data:  $X_i, Y_i$



- All these lines 'fit' the data. But we need only one, best fit
- But which fit is the 'best' (or, what is the criteria for the 'best' fit?)

# Linear regression

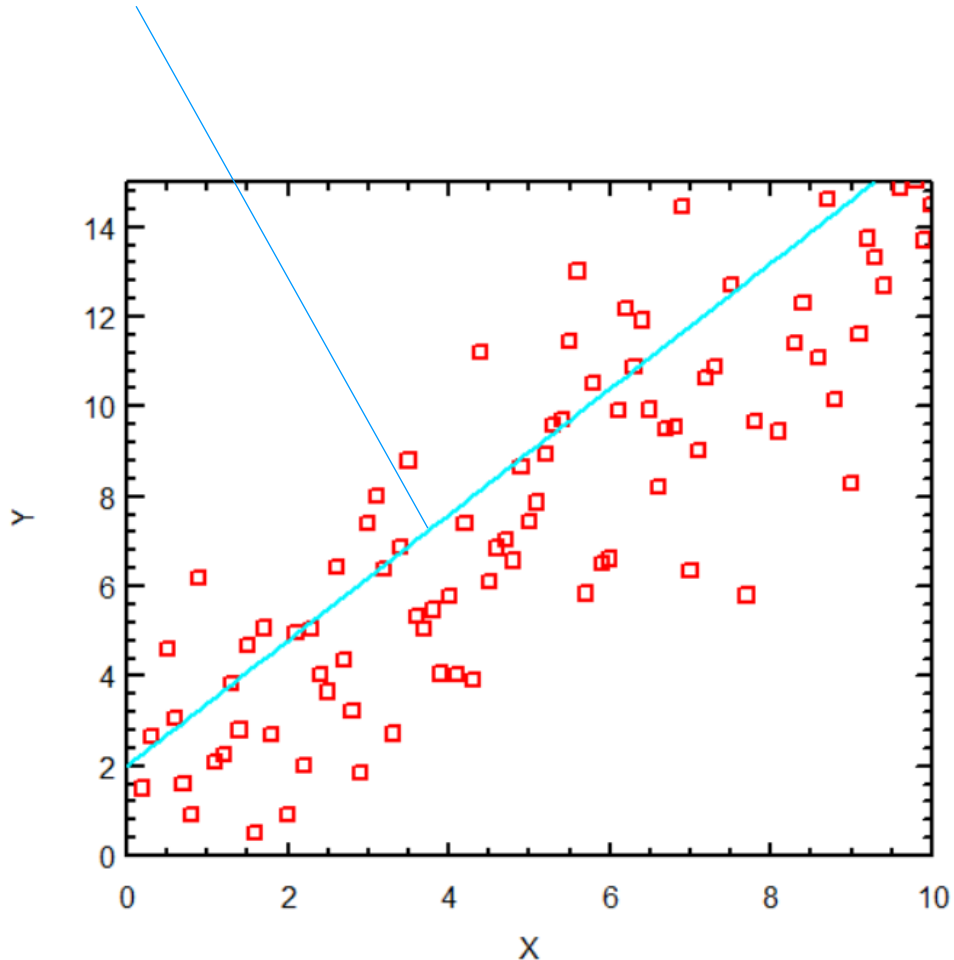
Model (or 'fit'),  $y = a + bx$



Residual (or deviation)  $\varepsilon_i$  depends on (or is a function of)  $a$  and  $b$

# Linear regression

Model (or 'fit'),  $y = a + bx$



- We assume that our data variables are related as

$$Y_i = a + bX_i + \varepsilon_i$$

with deviation values (or **residuals**) forming normal (Gaussian) distribution  $n(\varepsilon) \sim \exp(-\varepsilon^2/\sigma^2)$

- To satisfy the above assumption, we need to find the parameters  $a$  and  $b$  so that

$$\sum_i \varepsilon_i^2(a, b) = \min$$

**Least Squares method**

# Linear regression

## *How to find the linear fit?*

The total squared deviation can be represented by

$$E(a, b) = \sum_i \varepsilon_i^2 = \sum_i (Y_i - a - bX_i)^2 = \min$$

How do we find the minimum of function  $E(a, b)$ ? – By using basic calculus: find the values of  $a$  and  $b$  so that

$$\frac{\partial E}{\partial a} = 0$$

and

$$\frac{\partial E}{\partial b} = 0$$

The above two equations form a set  $\rightarrow$  solving it gives us the model parameters.

# Linear regression

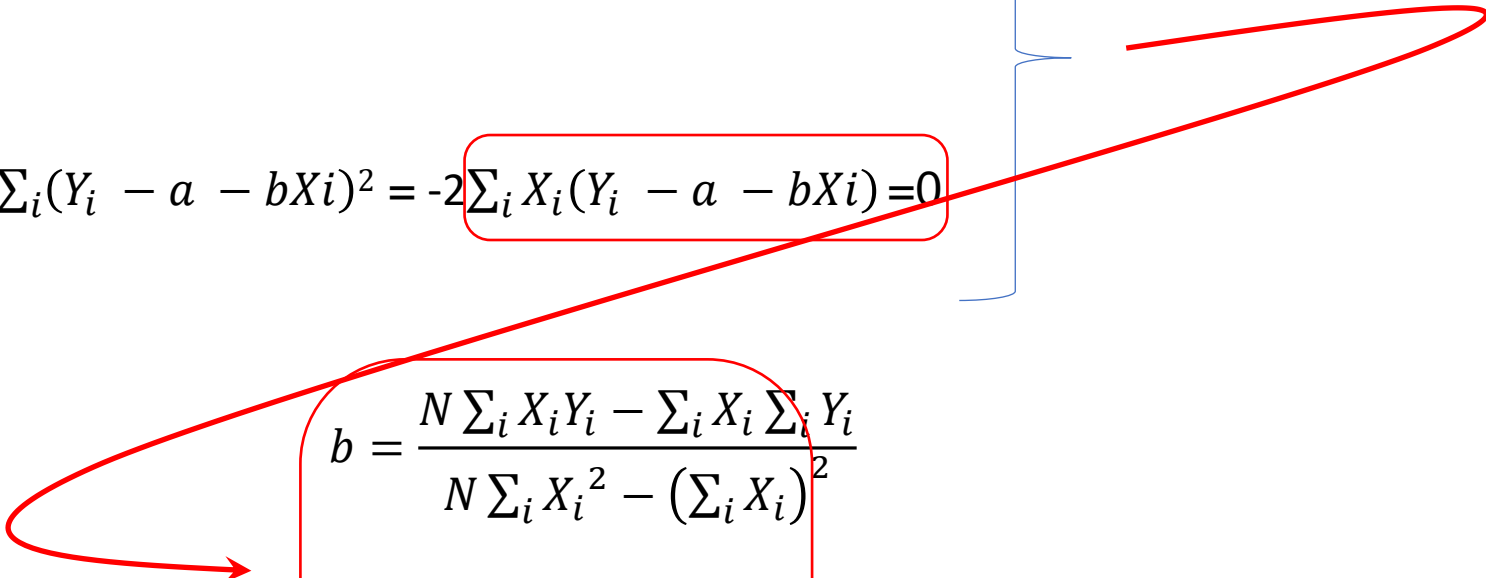
## *How to find the linear fit?*

Find the values of  $a$  and  $b$  so that

$$\frac{\partial E}{\partial a} = \frac{\partial}{\partial a} \sum_i (Y_i - a - bX_i)^2 = -2 \sum_i (Y_i - a - bX_i) = 0$$

and

$$\frac{\partial E}{\partial b} = \frac{\partial}{\partial b} \sum_i (Y_i - a - bX_i)^2 = -2 \sum_i X_i (Y_i - a - bX_i) = 0$$


$$b = \frac{N \sum_i X_i Y_i - \sum_i X_i \sum_i Y_i}{N \sum_i X_i^2 - (\sum_i X_i)^2}$$

$$a = \frac{1}{N} \sum_i Y_i - b \frac{1}{N} \sum_i X_i$$

# Multi-linear regression – why?

Will the client renew their subscription?

- it will depend on their income (car class, price?)
- satisfaction with our service
- age
- education
- .....

Using too many predictor parameters -> Overfitting, noise, computationally expensive

Using too few predictor parameters -> Low-quality model

} Prioritise predictors

# Linear regression (multi-dimensional case)

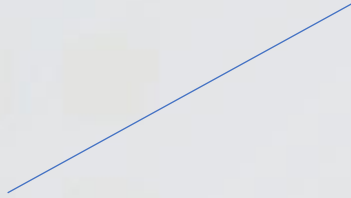
... is a case of multi-dimensional regression.

- In multi-linear regression we assume that our data variables are related as

$$Y_i = a + b^{(I)}X_i^{(I)} + b^{(II)}X_i^{(II)} + b^{(III)}X_i^{(III)} + \dots + \varepsilon_i$$

- We can go beyond linearity assuming

$$Y_i = a + b^{(I)}X_i^{(I)} + b^{(II)}X_i^{(II)} + b^{(III)}X_i^{(I)} X_i^{(II)} + \dots + \varepsilon_i$$



Cross-terms

# Linear regression (multi-dimensional case)

- Linear regression can be used when dealing with more than one input variable
- In multi-linear regression we assume that our data variables are related as

$$Y_i = a + b^{(I)}X_i^{(I)} + b^{(II)}X_i^{(II)} + b^{(III)}X_i^{(III)} + \dots + \varepsilon_i$$

with deviation values forming normal (Gaussian) distribution  $n(\varepsilon) \sim \exp(-\varepsilon^2/\sigma^2)$

- To satisfy the above assumption, we need to find the parameters  $a$  and  $b^{(I)}$ ,  $b^{(II)}$ ,  $b^{(III)}$  and so on so that

$$E(a, b^{(I)}, b^{(II)}, b^{(III)} \dots) = \sum_i \varepsilon_i^2 = \sum_i \left( Y_i - a - b^{(I)}X_i^{(I)} - b^{(II)}X_i^{(II)} - b^{(III)}X_i^{(III)} \dots \right)^2 = \min$$

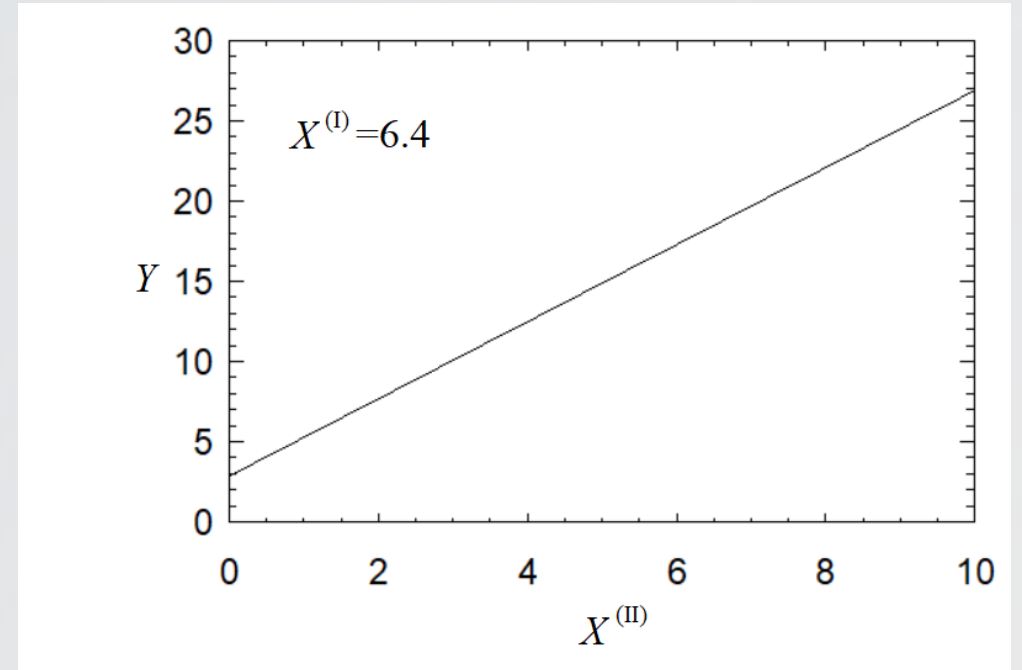
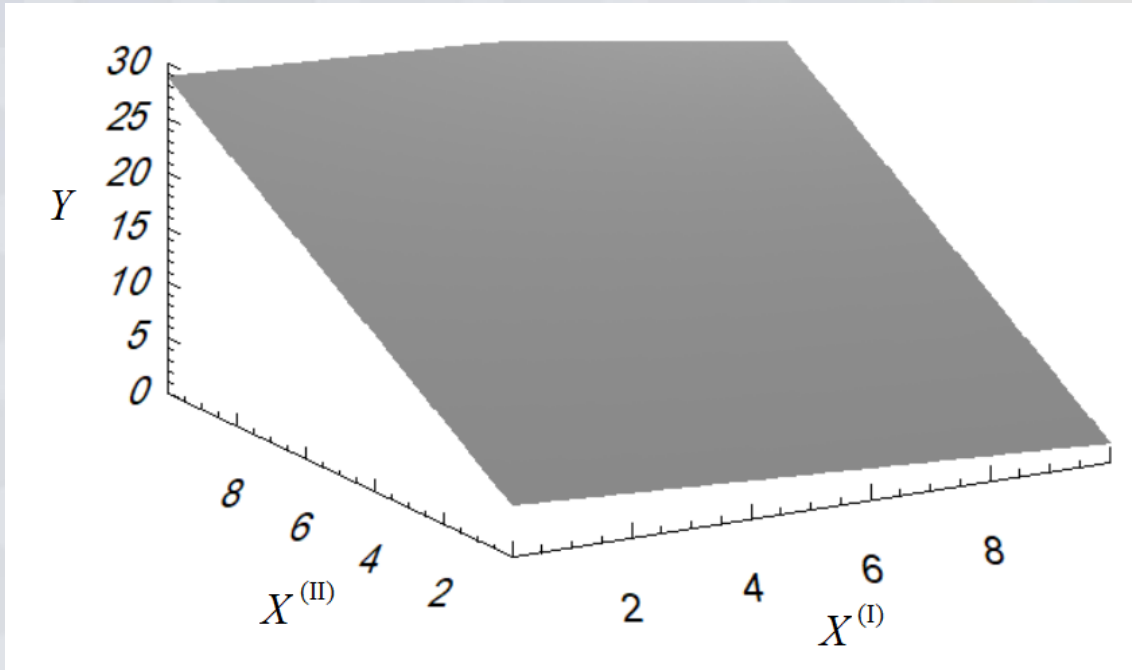
-> **least square method**



# Linear regression (multi-dimensional case)

- Geometrically, the fit (i.e. the model) is a multi-dimensional surface
- In any 'slice' of that multi-dimensional cube, this surface is a line

$$Y = 4.8 - 0.3 X^{(\text{I})} + 2.4 X^{(\text{II})}$$

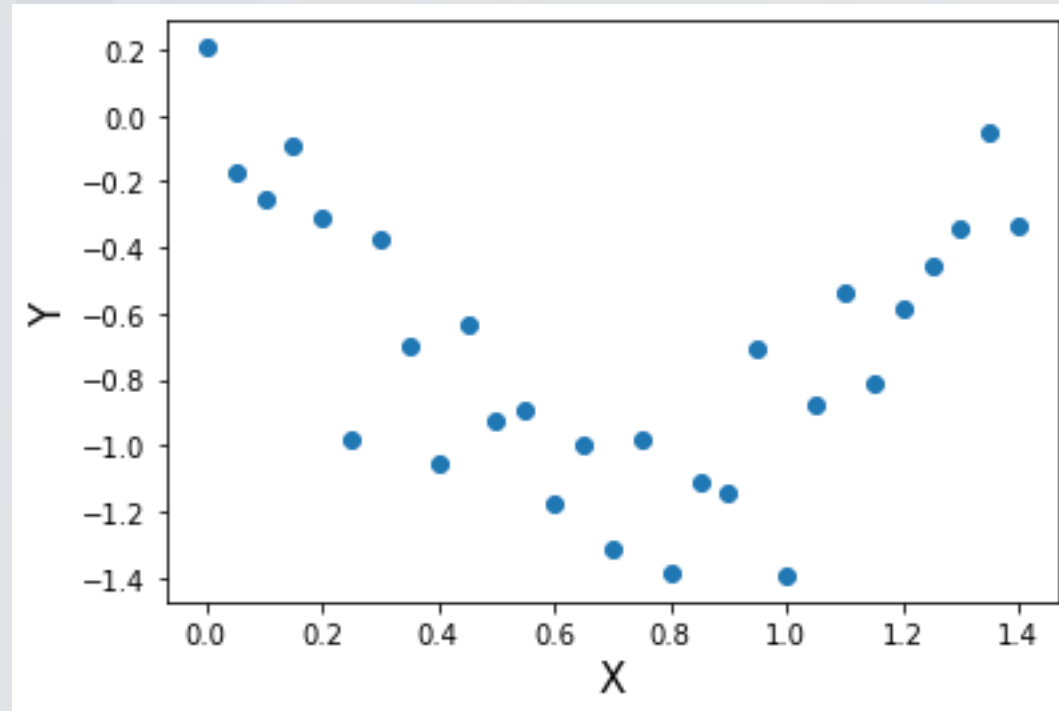


# What if the linear fit does not work?

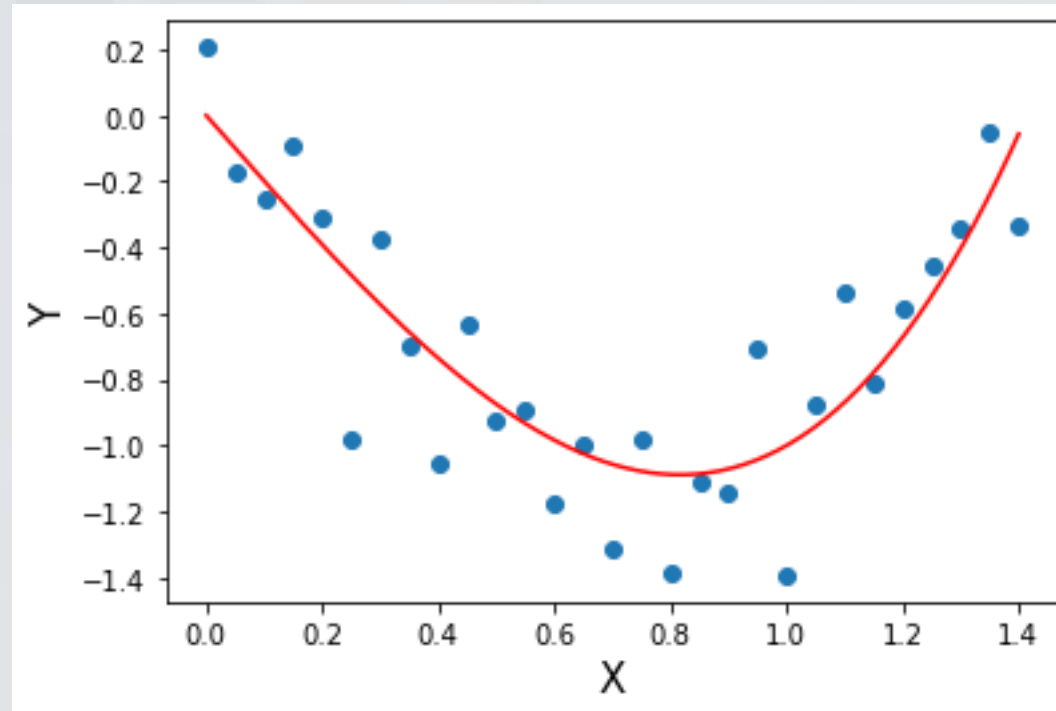
- If the relation between the variables is expected to be some other function or some other function fits your data better then a non-linear fit is used

*Function ZOO ...*

# Function zoo

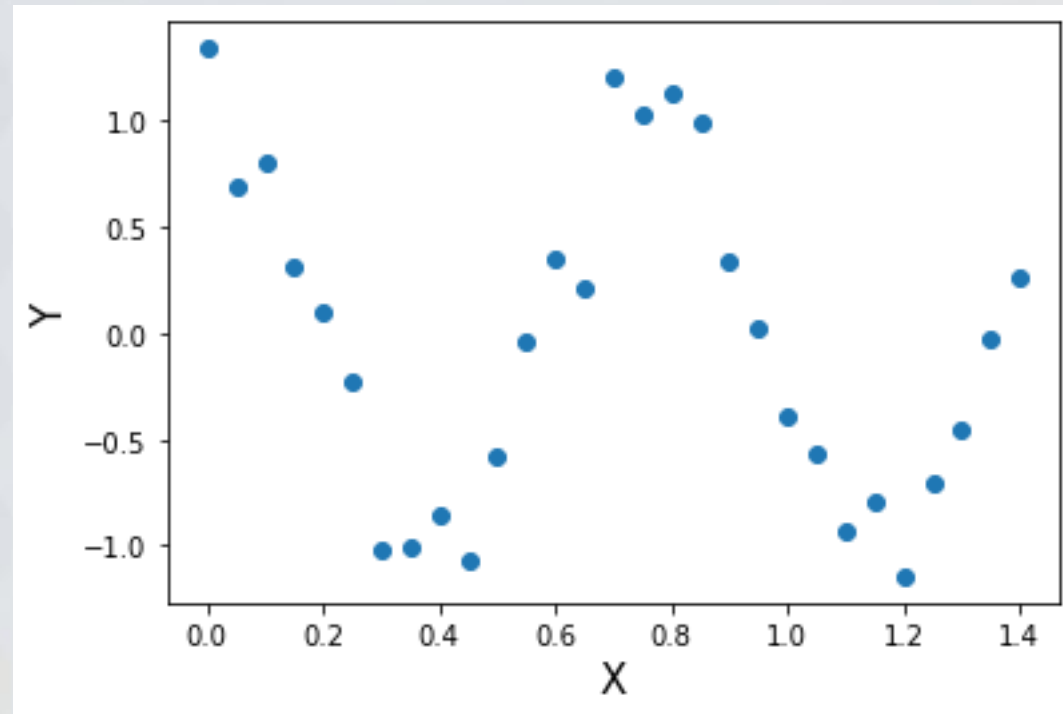


# Function zoo

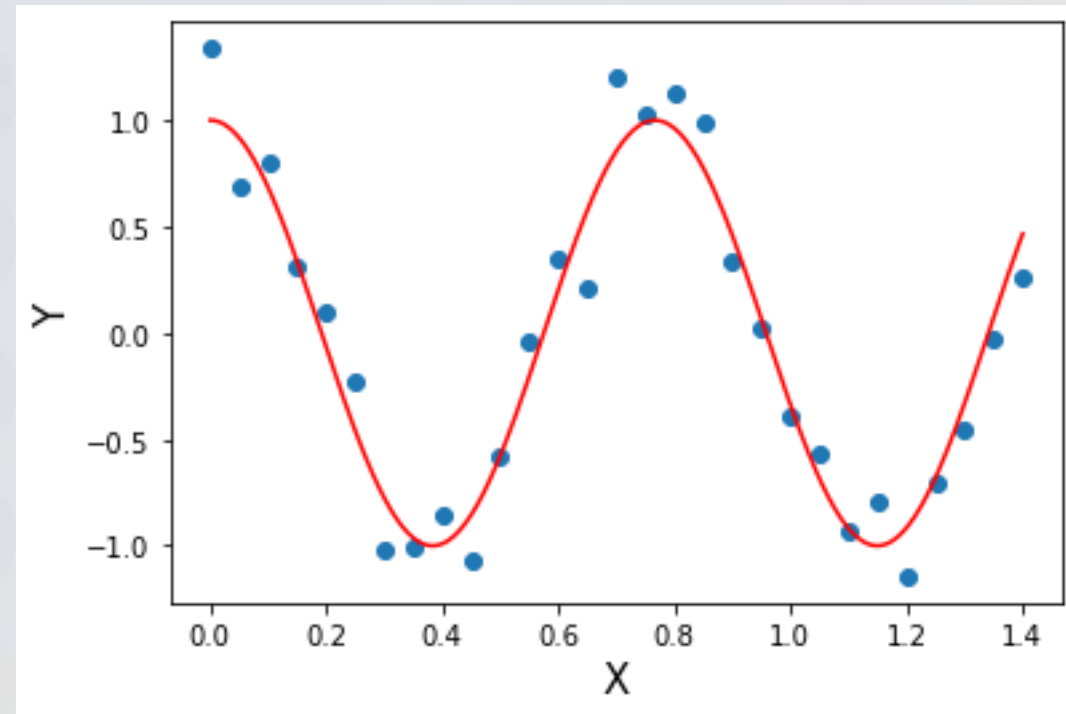


Polynomial, i.e.  $y(x) = a + bx + cx^2 + cx^3 + \dots$

# Function zoo

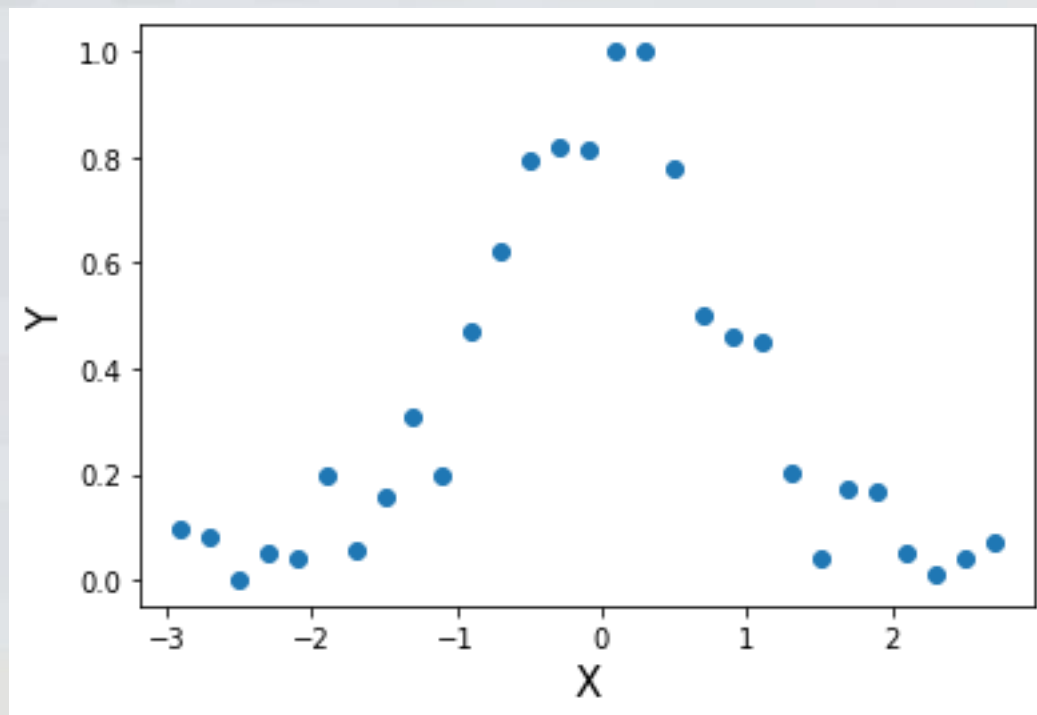


# Function zoo

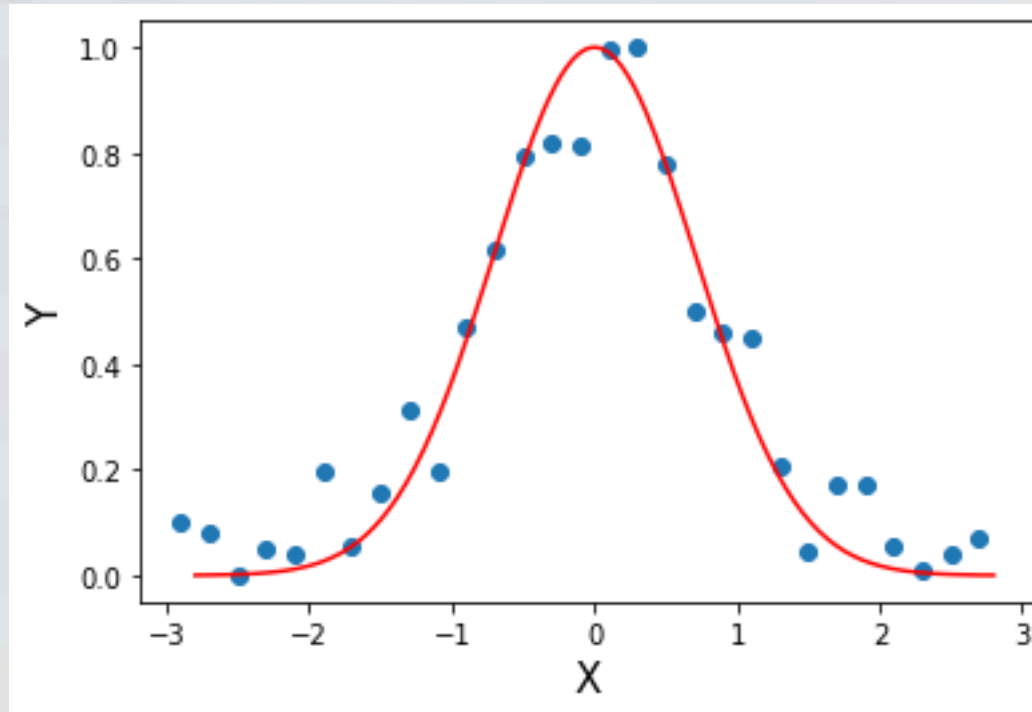


Harmonic function  $y(x) = \sin(x)$  or  $\cos(x)$

# Function zoo



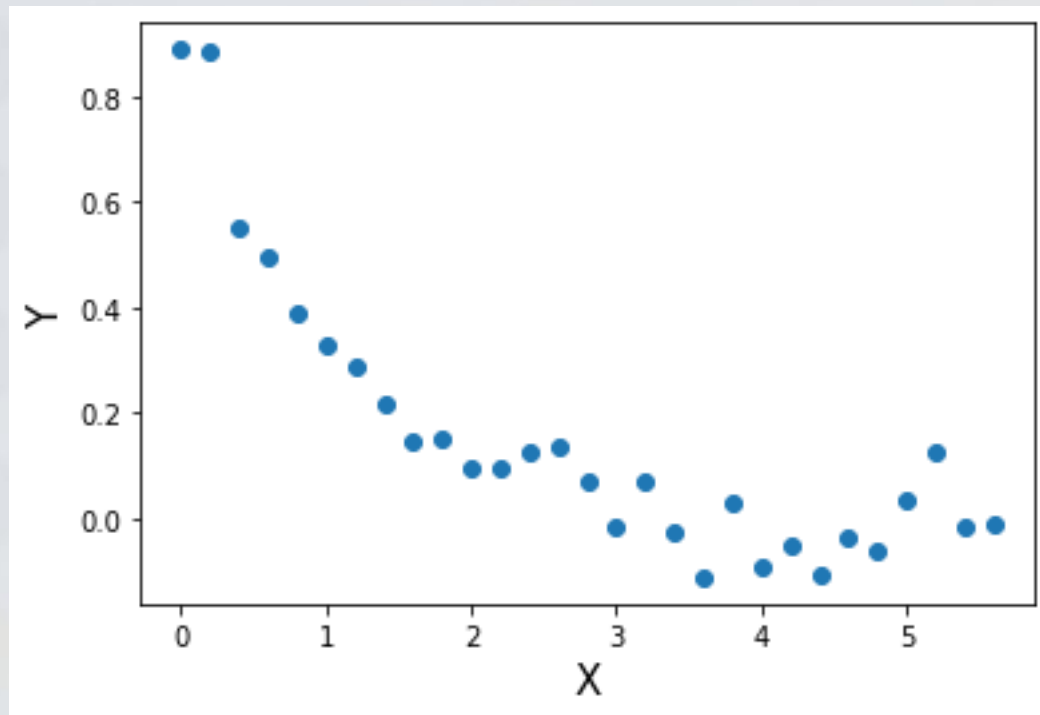
# Function zoo



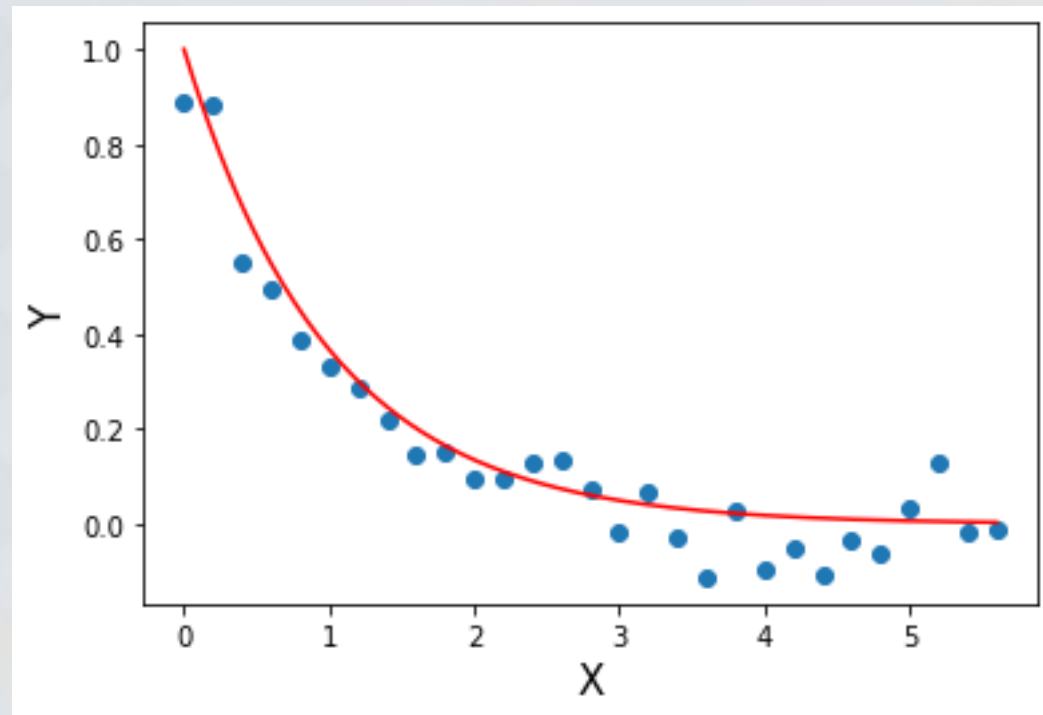
Bell-shaped function, e.g. Gaussian  $y(x) = \exp(-x^2) = e^{-x^2}$   
or Loretzian  $y(x) = \frac{1}{1+x^2}$   
etc etc



# Function zoo

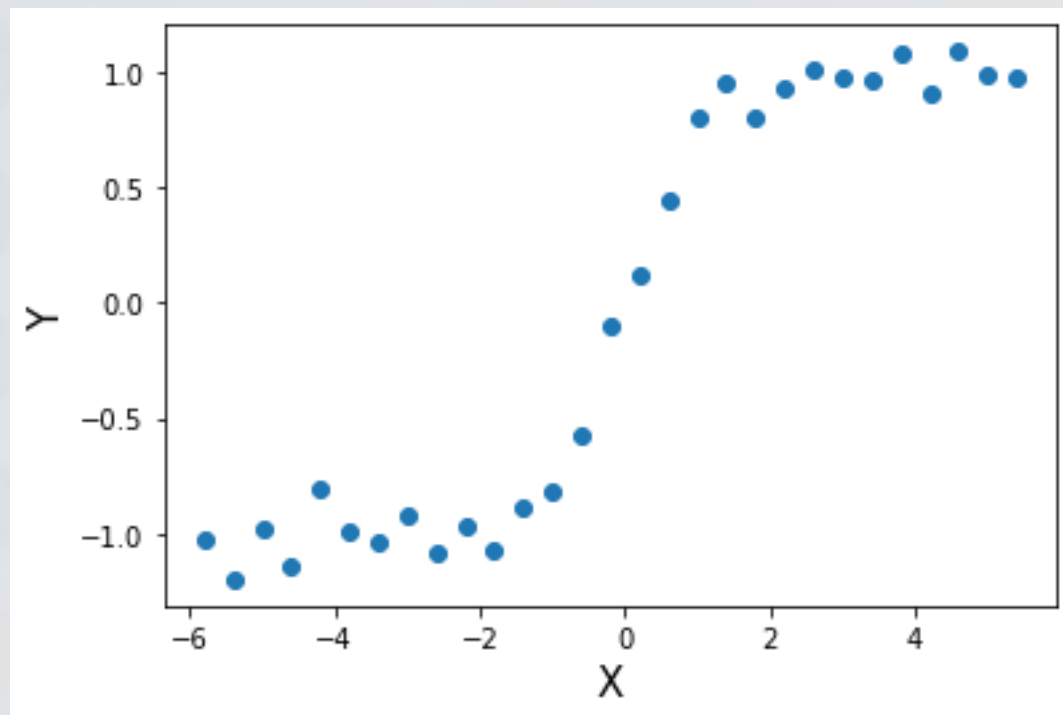


# Function zoo

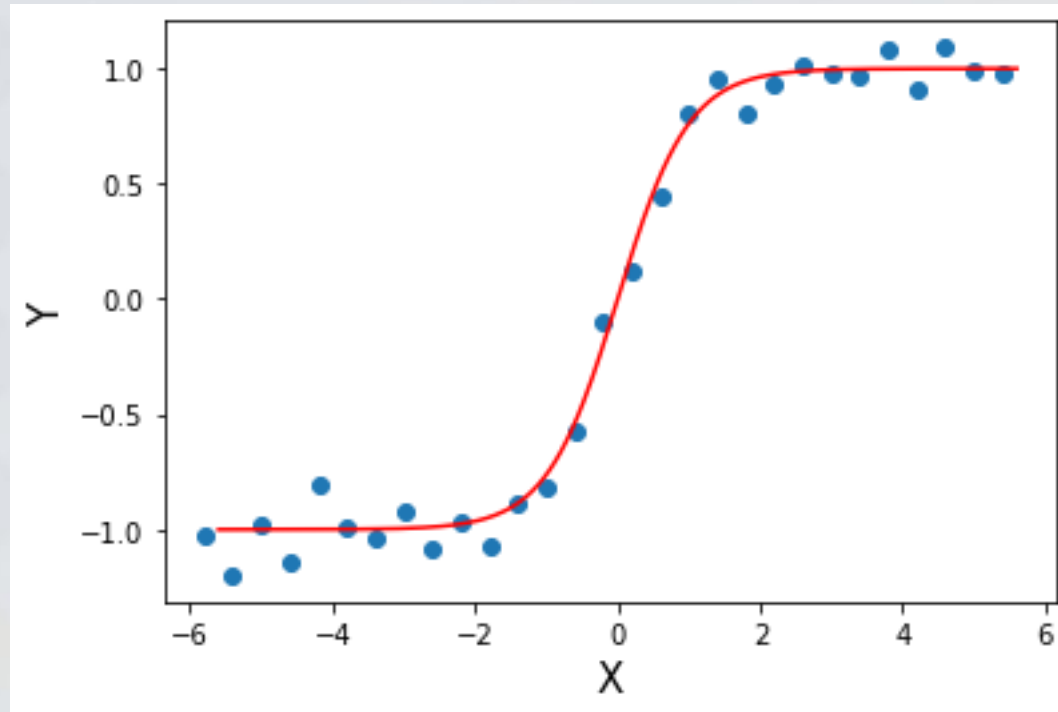


Decay function, e.g.  $y(x) = \exp(-x)$

# Function zoo



# Function zoo



Step-like function, e.g.  $y(x) = \tanh(x)$  or  $y(x) = \operatorname{erf}(x)$

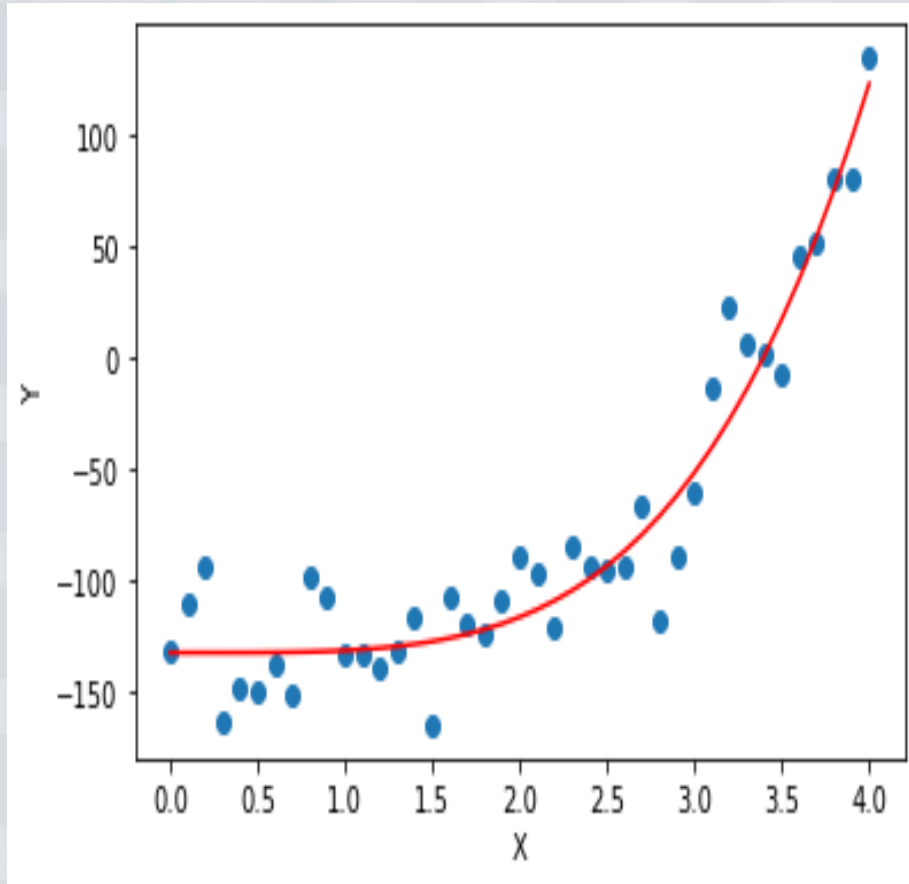
# Function zoo

Parameters add flexibility

$$y(x) = \exp(-x^2)$$

$$y(x) = A + B \exp\left(-\frac{(x-C)^2}{D}\right)$$

# Polynomial regression



- Probabilistic model

- We assume that our data variables are related as

$$Y_i = a + bX_i + cX_i^2 + dX_i^3 + \dots + \varepsilon_i$$

with deviation values forming normal (Gaussian) distribution  $n(\varepsilon) \sim \exp(-\varepsilon^2/\sigma^2)$

- To satisfy the above assumption, we need to find the parameters a and b so that

$$\sum_i \varepsilon_i^2 = \min$$

aka **least square method**

# Polynomial regression – computational aspect

$$f(x) = a + bx + cx^2 + \textcircled{d}x^3$$

*This is nothing to do with derivatives, here  $d$  is a constant*

I'm given data in form of  $[F_i, X_i]$

I need to fit  $F_i = a + bX_i + cX_i^2 + dX_i^3 + \varepsilon_i$

$$E(a, b, c, d) = \sum_i \varepsilon_i^2 = \sum_i (F_i - a - bX_i - cX_i^2 - dX_i^3)^2 = \min$$

$$f(x, y, z) = a + bx + cy + \textcircled{d}z$$

I'm given data in form of  $[F_i, X_i, Y_i, Z_i]$

I need to fit  $F_i = a + bX_i + cY_i + dZ_i + \varepsilon_i$

$$E(a, b, c, d) = \sum_i \varepsilon_i^2 = \sum_i (F_i - a - bX_i - cY_i - dZ_i)^2 = \min$$

Therefore, if we assume that  $X_i = X_i, Y_i = X_i^2, Z_i = X_i^3$  then there is no difference between multi-linear and polynomial regression

# Polynomial regression

- 1D polynomial regression problem can be reduced to the multi-linear regression problem by treating each  $x^m$  as a separate predictor variable

$$\vec{Y} = \mathbf{X}\vec{b}$$

**Linear regression**

$$\vec{Y} = \begin{pmatrix} y_0 \\ \vdots \\ y_n \end{pmatrix} \quad X = \begin{pmatrix} 1 & x_0 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \quad \vec{b} = \begin{pmatrix} b_0 \\ b_1 \end{pmatrix}$$

**Multi-linear regression**

$$\vec{Y} = \begin{pmatrix} y_0 \\ \vdots \\ y_n \end{pmatrix} \quad X = \begin{pmatrix} 1 & x_0^{(I)} & \cdots & x_0^{(M)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n^{(I)} & \cdots & x_n^{(M)} \end{pmatrix} \quad \vec{b} = \begin{pmatrix} b_0 \\ \vdots \\ b_M \end{pmatrix}$$

**1D polynomial regression**

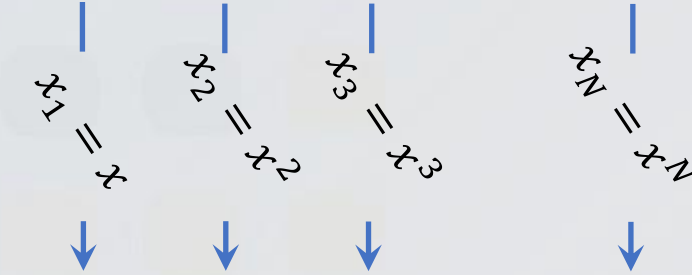
$$\vec{Y} = \begin{pmatrix} y_0 \\ \vdots \\ y_n \end{pmatrix} \quad X = \begin{pmatrix} 1 & x_0 & \cdots & x_0^m \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^m \end{pmatrix} \quad \vec{b} = \begin{pmatrix} b_0 \\ \vdots \\ b_m \end{pmatrix}$$



# Polynomial regression

Practically, we can use multi-linear regression tools for polynomial regression:

- 1) Let's say we have  $[x_i, y_i]$  data and we want to create a polynomial regression model with the degree  $N$
- 2) We will reduce the problem to multi-linear regression by creating  $N$  variables:

$$y(x) = a + k_1x + k_2x^2 + k_3x^3 + \dots + k_Nx^N$$


$$y(x_1, x_2, x_3, \dots, x_N) = a + k_1x_1 + k_2x_2 + k_3x_3 + \dots + k_Nx_N$$

- 3) Perform multi-linear regression
- 4) Use obtained parameters  $a, k_1, k_2 \dots k_N$  for polynomial model

# Finding the optimal model



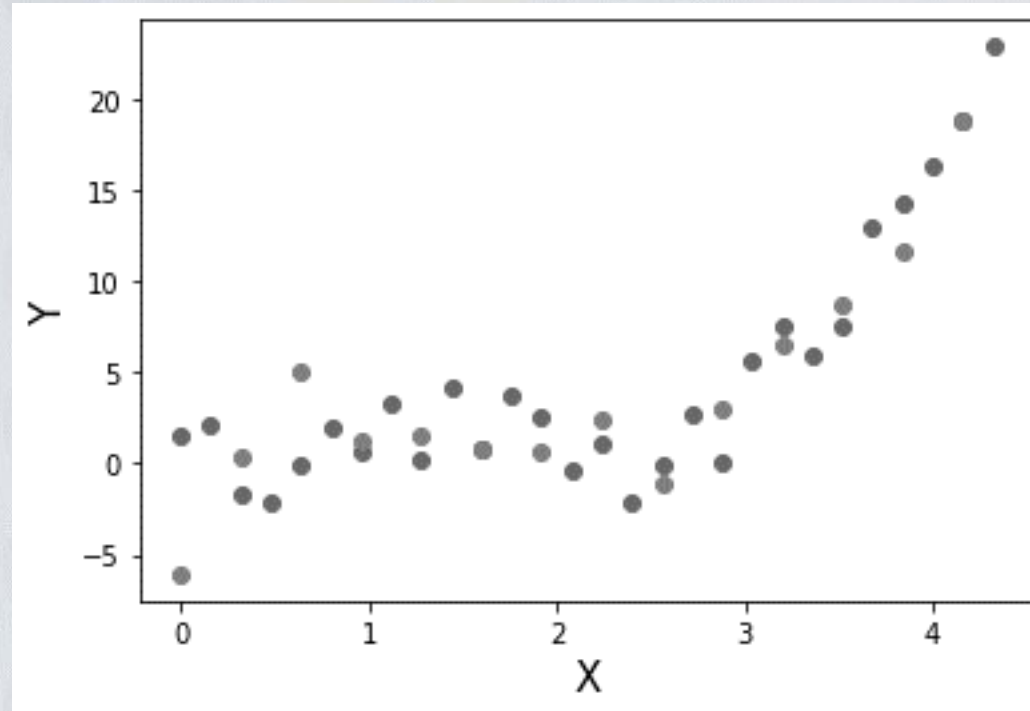
Why? – Because when we are fitting the training data, the MSE for the training data will always decrease with the increasing complexity of the model

With polynomial functions, if we are fitting data containing  $N$  point, then a polynomial function with degree  $N-1$  will yield  $MSE=0$ . Does it mean that the highest polynomial (or the most complex function) is always a best fit? – Obviously, no

We need a separate set of data, coming from the same place as the training data set, in order to test our chosen models and decide

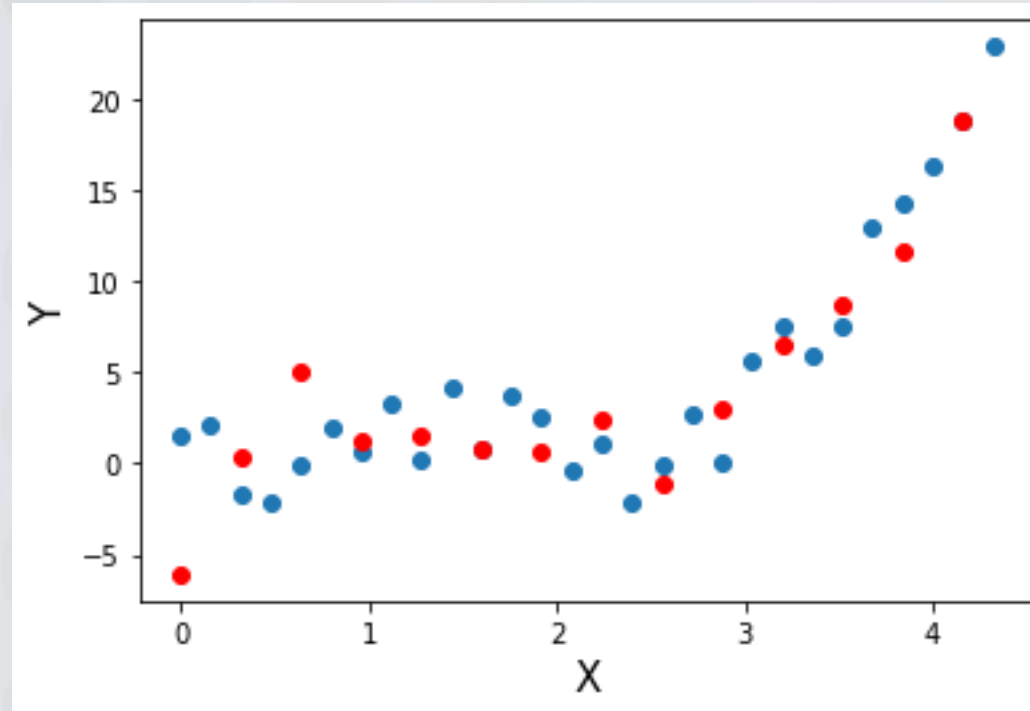
# Finding the optimal model

Data



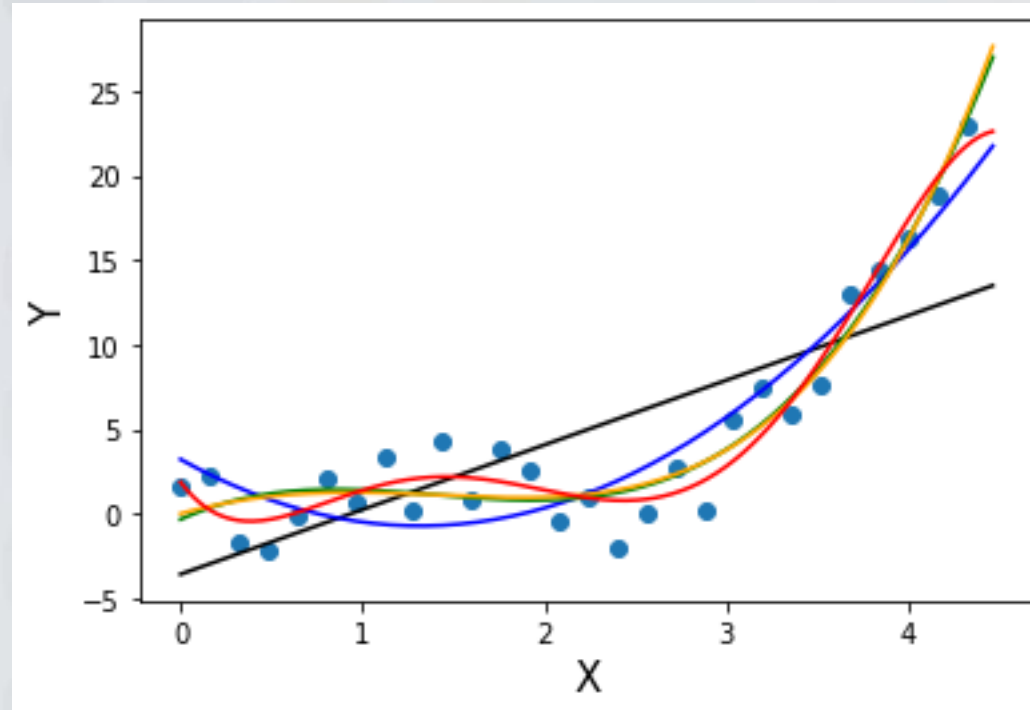
# Finding the optimal model

Data (training and validation)



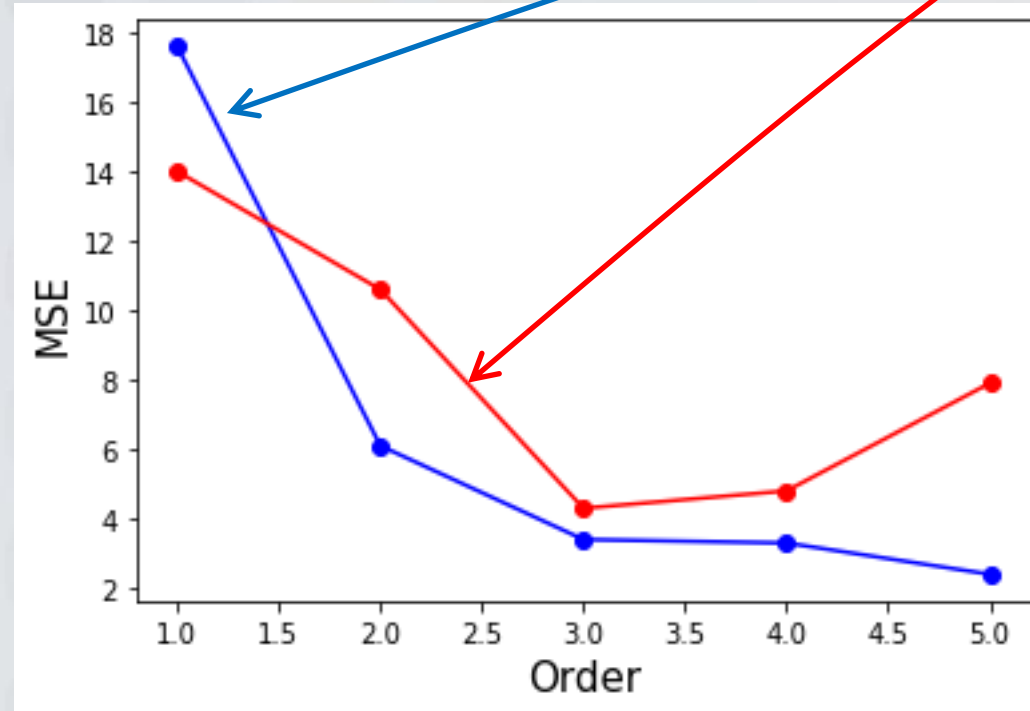
# Finding the optimal model

Data and polynomial regression models



	MSE (training)	MSE (testing)
N=1	17.7	14.0
N=2	6.1	10.6
N=3	3.4	4.3
N=4	3.3	4.8
N=5	2.4	7.9

# Finding the optimal model



Underfitting

Overfitting

Minimum MSE for test data is the optimal choice!

	MSE (training)	MSE (testing)
N=1	17.7	14.0
N=2	6.1	10.6
N=3	3.4	4.3
N=4	3.3	4.8
N=5	2.4	7.9