Latakia University
Information Engineering Faculty
Artificial Intelligence Department
Fourth Year

Artificial Neural Networks

Lecture 3 : Features Scaling & polynomial Regression

Dr. Wisam Ibrahim

2025-2026 Semester 1

FEATURES SCALING

- $X = \begin{bmatrix} 150 & 2 & 2 & 12 \\ 100 & 2 & 1 & 3 \\ 75 & 1 & 2 & 4 \\ 200 & 3 & 3 & 8 \end{bmatrix}$

- $Y = \begin{bmatrix} 15000 \\ 11000 \\ 9000 \\ 19000 \end{bmatrix}$

Area	Bedroom	Floor	Age	Price y' s
150	2	2	12	15,000
100	2	1	3	11,000
75	1	2	4	9,000
200	3	3	8	19,000

FEATURES SCALING

- Feature scaling is unifying all domains of feature values in one domain.
- That is, if there are values in feature vectors within large and small domains, the domains will be unified, therefore all values become consistent within one domain.
- The large disparity between domains in the features makes the Gradient Descent algorithm runs slowly, so we have to apply Feature scaling to unify the values in domain [-3,3]:
- we have the following function:
- $X' = \frac{X-\mu}{\sigma}$
- μ: is the mean of X values.
- σ: is the standard deviation of X values.

FEATURES SCALING

Area
150
100
75
200

- for the Area feature :
- Average $=\mu = \frac{150+100+75+200}{4}$=131.25
- standard deviation $\sigma = \sqrt{\frac{\sum(x-\mu)^2}{n-1}}$
- $= \sqrt{\frac{(150-131.25)^2+(100-131.25)^2+(75-131.25)^2+(200-131.25)^2}{3}}$
- $= \sqrt{\frac{(18.75)^2+(-31.25)^2+(-56.25)^2+(68.75)^2}{3}}$

$= \sqrt{\frac{(351.56)+(976.56)+(3164.06)+(4726.56)}{3}}$ =55.43

- σ=55.43

FEATURES SCALING

- Based on the function:
- $X' = \frac{X-\mu}{\sigma}$

  the feature Area will be :

Area X'	Area X
$\frac{150-131.25}{55.43}$=0.33	150
$\frac{100-131.25}{55.43}$=-0.56	100
$\frac{75-131.25}{55.43}$= -1.01	75
$\frac{200-131.25}{55.43}$=1.24	200

FEATURES SCALING

- The dataset after the feature scaling will be as following:

Area'	Bedroom'	Floor'	Age'	Price y
0.33	0	0	1.27	15,000
-0.56	0	-1.23	-0.91	11,000
-1.01	-1.23	0	-0.67	9,000
1.24	1.23	1.23	0.30	19,000

- If we want to predict the price of house having this feature vector:
- (175,3,2,4)
- We have to apply feature scaling on this feature vector so the result will be:
- (0.79,1.23,0,-0.67)
- Then input this FV in the function *h(x)* to get the estimated output.

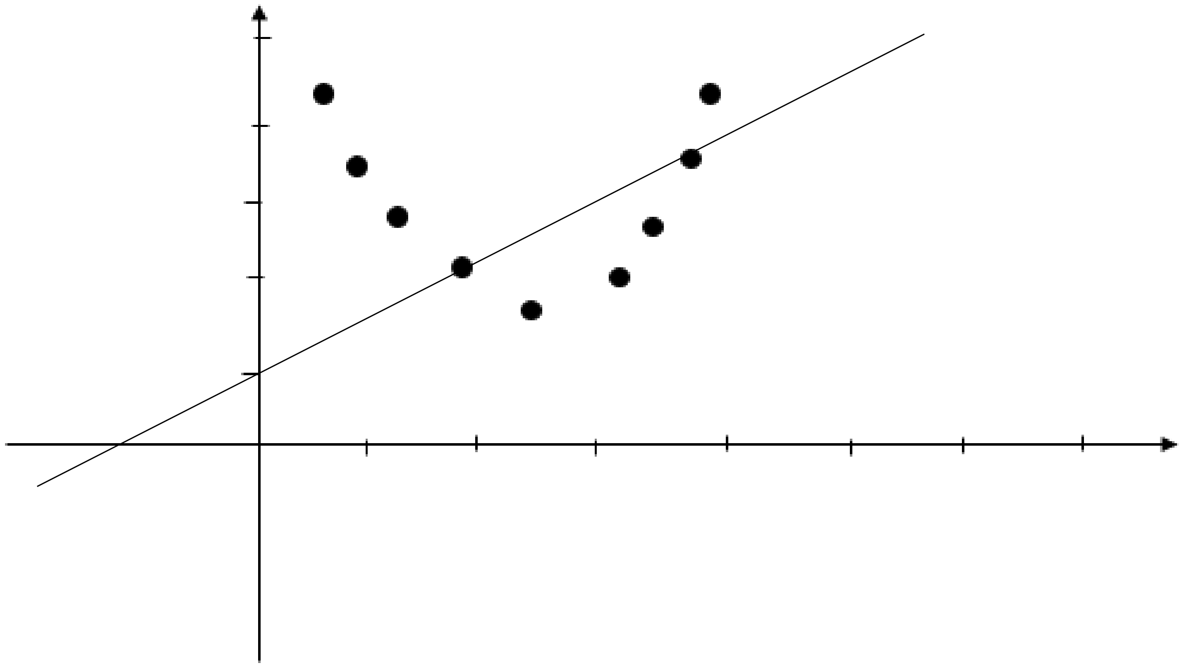
FEATURES SCALING-NORMALIZATION

- Another type of features scaling is Normalization.
- The normalization formula is a statistics formula that can transform a data set so that all of its variations fall between zero and one [0, 1].
- $x_{normalized} = (x - x_{minimum}) / range\ of\ x$

- $range\ of\ x\ values = x_{maximum} - x_{minimum}$
- **Normalization formula for custom ranges:**
- Puts all data within a custom range where the lowest value is ***a*** and the highest value is ***b*** i.e. [a, b]:
- $x_{normalized} = a + (((x - x_{minimum}) * (b - a)) / range\ of\ x)$

POLYNOMIAL REGRESSION

- Let's suppose the samples in the previous training dataset (house) are shown as follows:

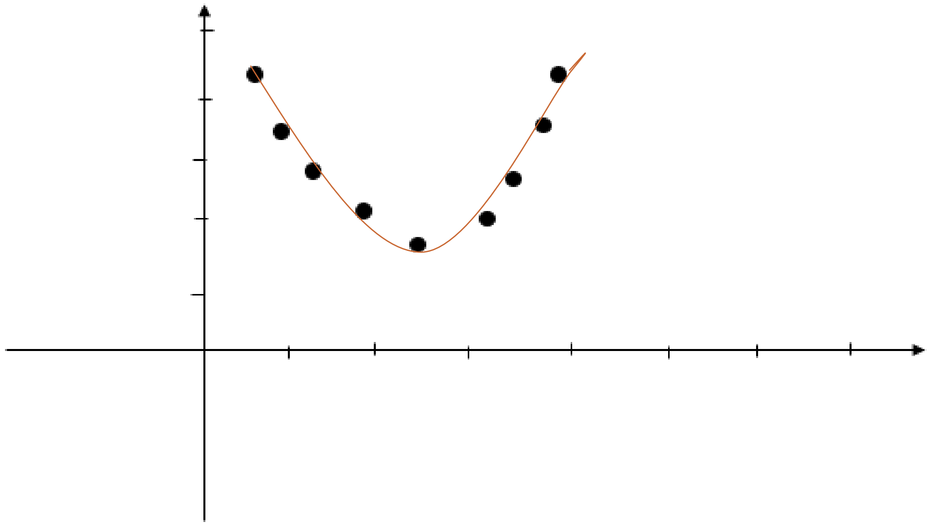


- By applying Linear regression on this dataset ($h\ (x) = \theta_0 + \theta_1 x$), we find that the function doesn't converge with most of the existing samples, therefore we have to search for another function which is Polynomial regression.

POLYNOMIAL REGRESSION

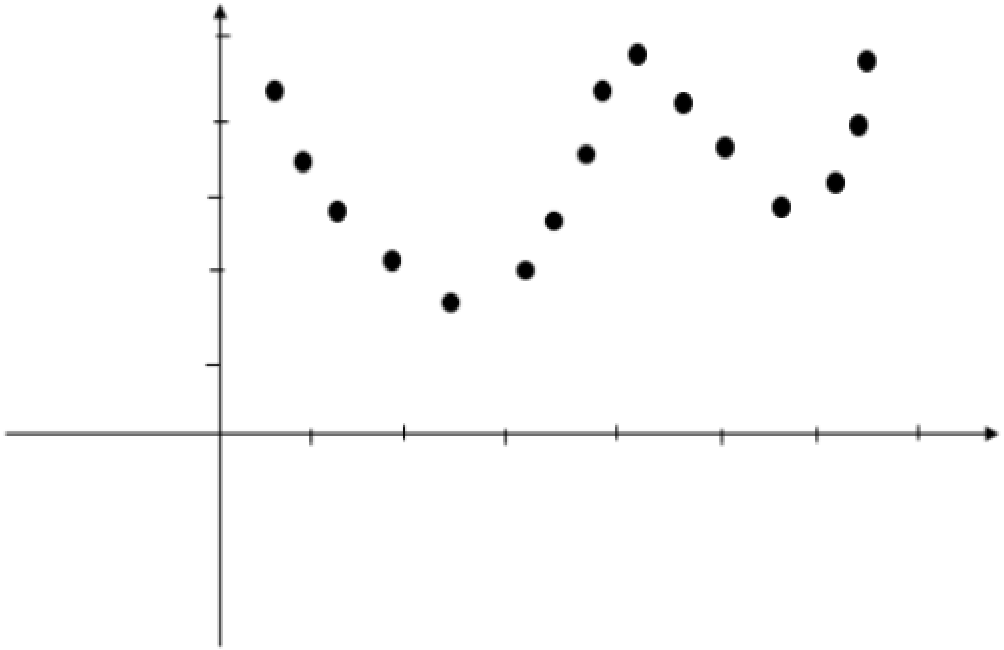
- We modify the equation of the previous function to become as follows:
- $h\,(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 \quad : x_2 = x_1^2$
- $h\,(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2$

- We note that the equation is a parabola function. After determining the parameters (θ_j) of the function by the Gradient Decent algorithm , the function can be represented as follows:



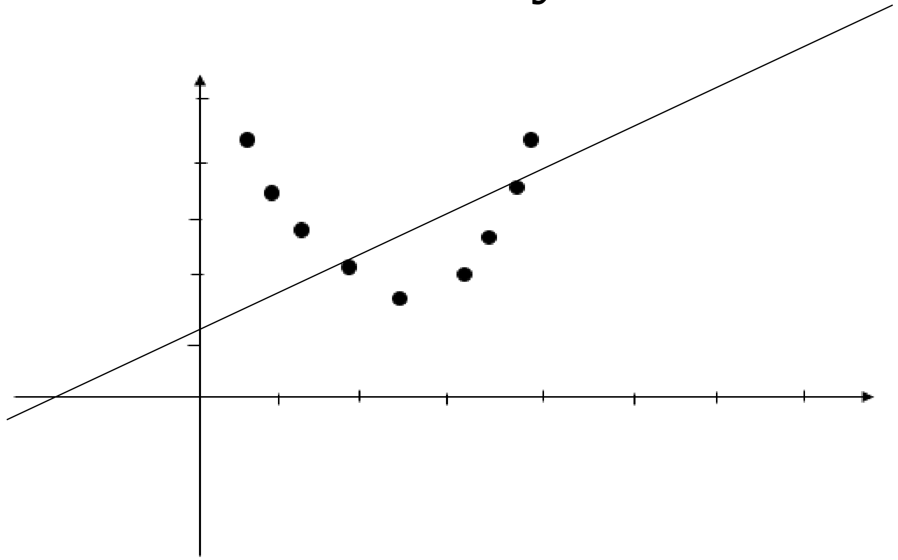
POLYNOMIAL REGRESSION

- If the samples in the data set are represented as follows:



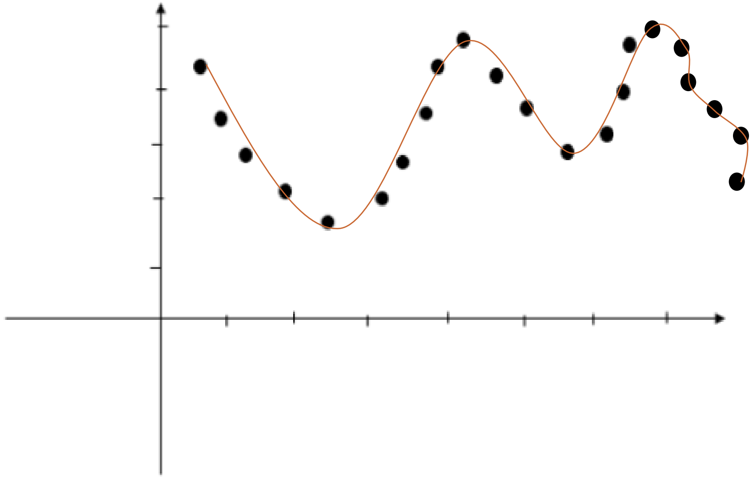
- Then we can say that the complexity of the function has increased.
- More features in the dataset, more complexity of the function.

UNDER FITTING & OVER FITTING

- Under fitting: means that the function (model) accuracy in the training phase (training data) is so low. Thus, the accuracy in the test phase (new data) will be low too. as follows:
- 
- For example :In the face recognition problem:
- If we trained the model on five faces and tested it for a new different face, it may give a result that the new face is one of the five faces, although there is very little similarity between them. ***Generalization***

UNDER FITTING & OVER FITTING

- Over Fitting: means that the function's (model) accuracy in the training phase (training data) is very High. However, the accuracy in the test phase (new data) will be low.



- For example :In the face recognition problem:
- If we trained the model on five faces and tested it for one of them (with a little change), it may give a result that the face is new, because there is very simple difference between them.
- ***Unimportant details.***

UNDER FITTING & OVER FITTING

- **In other words:**
- When we increase the complexity of the function, we make our model more representative of the data and takes the unimportant details into account. This increases the possibility of getting into Over Fitting problem.
- Conversely, when we reduce the complexity of the function, we make our model less representative of the data, so its accuracy is low. This increases the possibility of getting into Under Fitting problem .
- We gradually increase the complexity of the function so that we get good accuracy in the training phase without getting into the Over Fitting problem.