Lecture 1.3

- Normalization and Standardization
- Overfitting and Underfitting

Normalization and Standardization

- Both Normalization and Standardization are techniques used to adjust the scale of features in a dataset
- They are crucial in machine learning to ensure that all features contribute equally to the model and prevent any feature from dominating due to its scale

Normalization

- **Normalization** (also called Min-Max Scaling) is the process of transforming features such that they lie within a specific range, typically [0, 1] or [-1, 1]
- This is done by scaling the data to a fixed range based on the minimum and maximum values of the feature
- Formula:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

where x is the original value, min(x)is the minimum value, and max(x) is the maximum value in the dataset

- **Usage**: Algorithms like k-Nearest Neighbors (k-NN), and Neural Networks, which are sensitive to the scale of features.

Normalization Example

SL	Values	$\overline{x} = \dfrac{x - \min(x)}{\max(x) - \min(x)}$	Normalized Values
1	10	$\dfrac{10 - 10}{50 - 10}$	**0**
2	20	$\dfrac{20 - 10}{50 - 10}$	**0.25**
3	30	$\dfrac{30 - 10}{50 - 10}$	**0.50**
4	40	$\dfrac{40 - 10}{50 - 10}$	**0.75**
5	50	$\dfrac{50 - 10}{50 - 10}$	**1.00**

Standardization

- Standardization: Transforming data to have a mean of 0 and a standard deviation of 1 (also known as Z-Score Scaling)
- It centers the data and scales it based on standard deviation
- Formula:

$$x' = \frac{x - \mu}{\sigma}$$

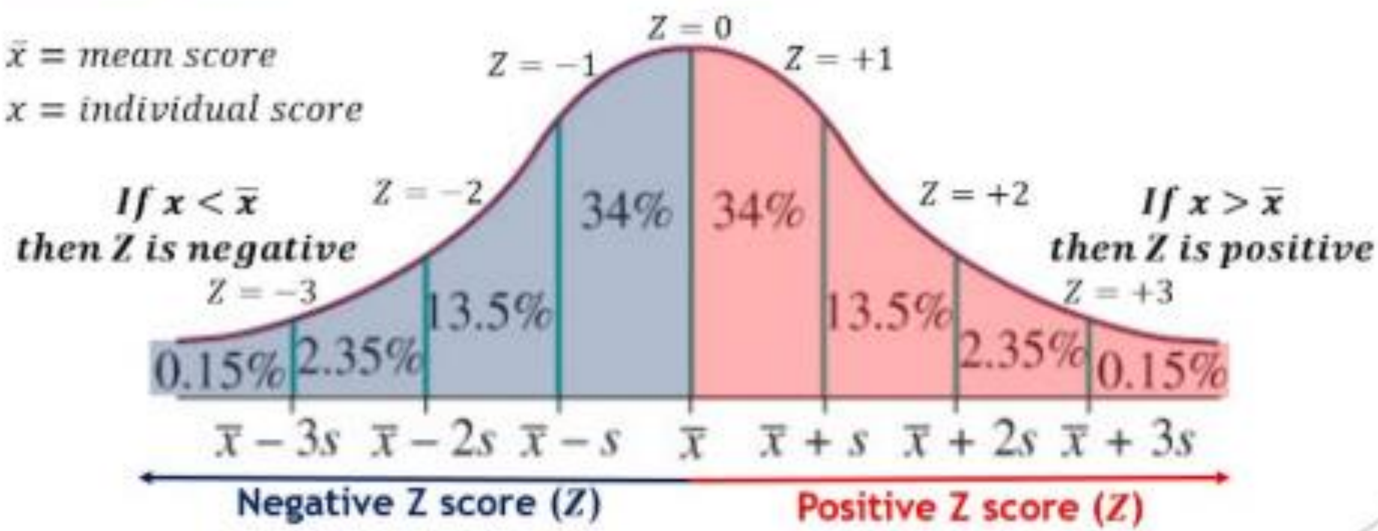
where μ is the mean and σ is the standard deviation of the dataset

- **Usage:** Algorithms like Support Vector Machines (SVM), Logistic Regression, and Principal Component Analysis (PCA) which assume a normal distribution or work better with data centered around 0.

Z Scores

Problem solving

A **Z score** or a **"standardised score"** is a numerical measure of how far an **individual score** is away from the **mean score**, within a normal distribution.



Standardization Example

SL	Values	Mean (μ)	Standard Deviation (σ)	$x' = \frac{x-\mu}{\sigma}$	Standardized Values
1	10	$(10 + 20 + 30 + 40 + 50)/5$ = **30**	$\sqrt{\frac{(\mathbf{10-30})^2 + (\mathbf{20-30})^2 + (\mathbf{30-30})^2 + (\mathbf{40-30})^2 + (\mathbf{50-30})^2}{\mathbf{5}}}$ $=\sqrt{\mathbf{200}} = \mathbf{14.14}$	$\frac{\mathbf{10-30}}{\mathbf{14.14}}$	**−1.41**
2	20			$\frac{\mathbf{20-30}}{\mathbf{14.14}}$	**−0.71**
3	30			$\frac{\mathbf{30-30}}{\mathbf{14.14}}$	**0.00**
4	40			$\frac{\mathbf{40-30}}{\mathbf{14.14}}$	**0.71**
5	50			$\frac{\mathbf{50-30}}{\mathbf{14.14}}$	**1.41**

Overfitting and Underfitting

- Overfitting and Underfitting are concepts in machine learning that describe how well a model generalizes to new data
- They are often indicators of how effectively a model has learned patterns from the training data

Overfitting

- **Overfitting** occurs when a model learns not only the underlying patterns in the training data but also the noise and details that do not generalize to unseen data
- Symptoms
  - High accuracy on training data
  - Poor performance on validation or test data
- Causes
  - Model is too complex (e.g., too many parameters or layers)
  - Insufficient training data
  - Training for too many epochs without regularization
- Prevention
  - Use regularization techniques
  - Reduce the model's complexity
  - Use more training data or data augmentation

Underfitting

- **Underfitting** occurs when a model is too simple to capture the underlying patterns in the data
- Symptoms
  - Poor performance on both training and validation/test data
  - Model fails to capture the complexity of the data
- Causes
  - Model is too simple
  - Insufficient training time
  - Features used in the model are not relevant or sufficient
- Prevention
  - Use a more complex model
  - Train the model for more epochs
  - Provide better or more features to the model

Differences

Aspect	Overfitting	Underfitting
Performance on Training Data	High accuracy	Low accuracy
Performance on Test Data	Poor	Poor
Model Complexity	Too complex	Too simple
Generalization	Poor	Poor