MACHINE LEARNING

Shoaib Farooq

Department of Computer Science

Disclaimer: This presentation includes contents available online including images copied from Google search and contents of presentations of other professors. I don't claim any image or text to be my own. All the credit goes to the original authors.

Instructor Information

Instructor	Shoaib Farooq	E-mail	m.shoaib1050@gmail.com
GitHub	https://github.com/SHOAIB1050		
YouTube	https://www.youtube.com/c/pakithub		
LinkedIn	https://www.linkedin.com/in/shoaib-farooq-b5b190105/		

Let’s Start …..

Lecture #9

GOALS

This Lecture Will Cover:

- Generalization
  - Overfitting vs Underfitting
  - Features Scaling
  - Train, Test and Evaluate model

THE NORMAL EQUATION

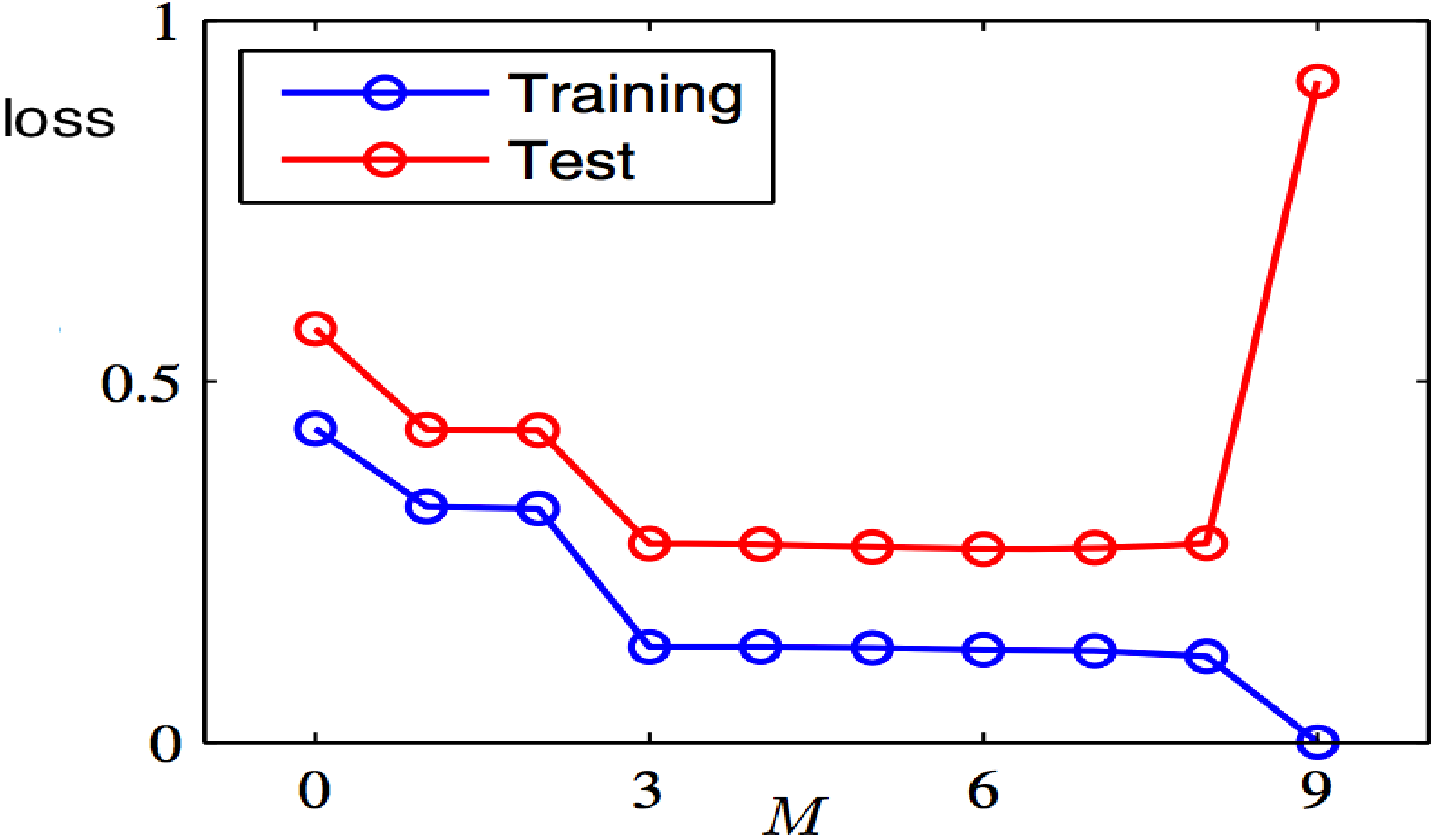
- Find the optimal values of the parameters directly
- The value of **W** that minimizes the cost function
  - closed-form solution
- This is called the *Normal Equation.*

$$W_i = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{y}$$

GENERALIZATION

- The model's ability to adapt properly to new and previously unseen data.
- We expect a model to perform well on both training and test data sets.
- What if model shows high accuracy on Training data and low accuracy on test data?
  - Not generalized well

GENERALIZATION



OVERFITTING

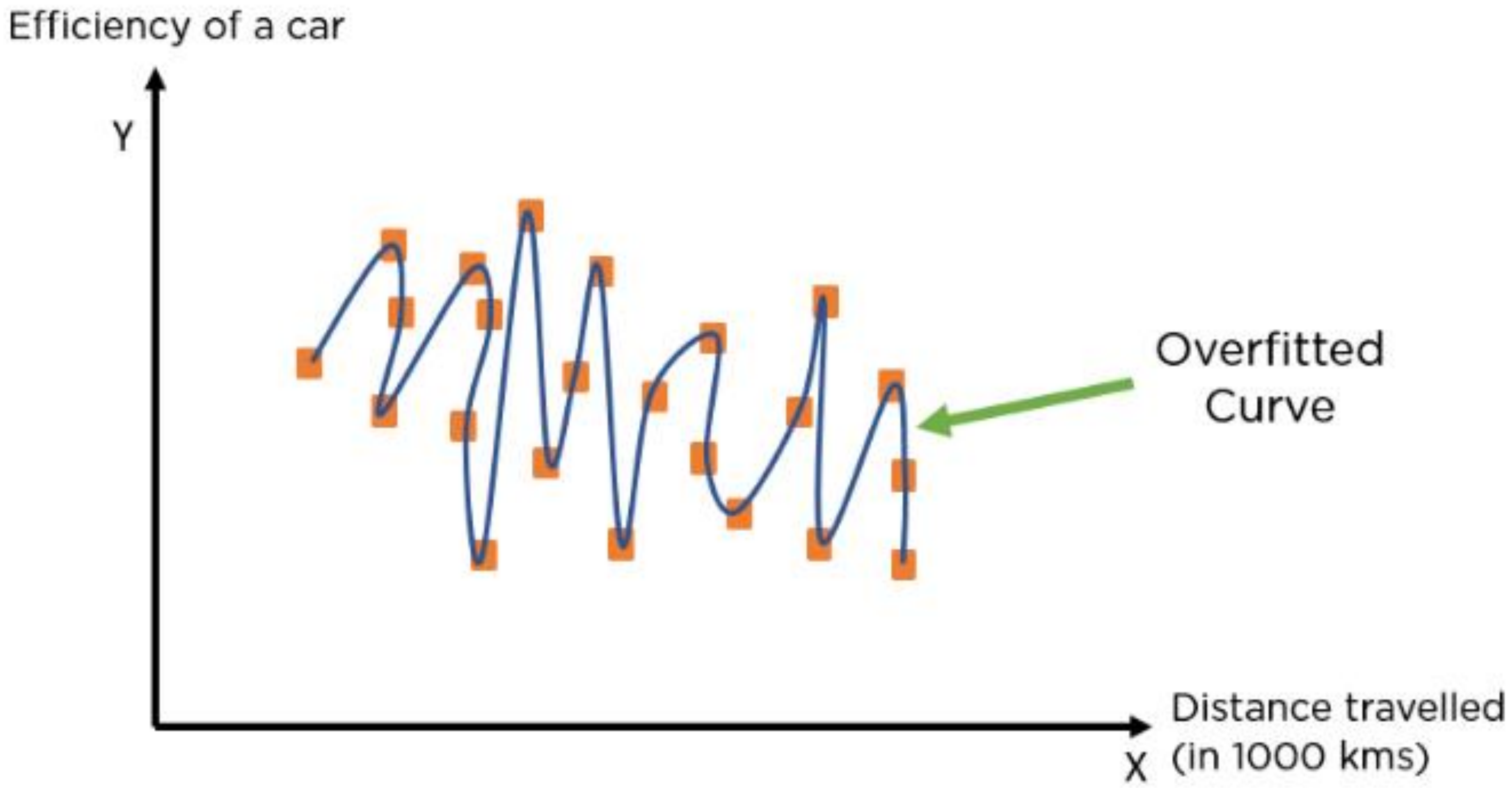
- **Definition**: Overfitting occurs when a model learns the training data too well, including its noise and outliers. This results in a model that performs exceptionally on training data but poorly on new, unseen data.
- **Causes**:
  - The model is too complex (e.g., too many parameters).
  - The training data is not representative of the overall data distribution.
  - The model is trained for too long.
- **Indicators**:
  - High accuracy on training data but low accuracy on validation/test data.

Example: Imagine a model designed to predict house prices. If it memorizes the training data, it might predict the price of houses in the training set perfectly but fail to generalize to new houses

OVERFITTING

- Noise in the data
- The model has a high variance
- Small size of the training dataset
- The model is too complex

OVERFITTING



SOLUTION OF OVERFITTING

- **Add More Data:** Increasing the size of your training dataset can help the model generalize better.
- **Data Augmentation:** For image data, techniques like flipping, rotating, or cropping can create more training examples1.
- **Regularization:** Techniques like L1 or L2 regularization add a penalty for larger coefficients, discouraging the model from fitting the noise in the training data1.
- **Simplify the Model:** Reducing the complexity of the model by removing some features or using a simpler algorithm can help prevent overfitting.
- **Cross-Validation:** Using k-fold cross-validation ensures that the model's performance is tested on different subsets of the data, helping to detect overfitting

UNDERFITTING

- **Definition**: Underfitting happens when a model is too simple to capture the underlying patterns in the data. This results in poor performance on both training and new data.
- **Causes**:
  - The model is too simple (e.g., not enough parameters).
  - Insufficient training time.
  - Inadequate features to capture the complexity of the data.
- **Indicators**:
  - Low accuracy on both training and validation/test data.
- **Example**: Using a linear model to predict house prices when the relationship between features and prices is non-linear. The model fails to capture the complexity of the data, leading to poor predictions

UNDERFITTING

- The model in unable to learn the training data well.
  - Low accuracy on training data.
  - May not generalize well on the new data
- Underfitting occurs due to high bias and low variance of model.
- The size of the training dataset used is not enough
- The model is too simple
- Not enough iterations

SOLUTION OF UNDERFITTING

- **Increase Model Complexity:** Use a more complex model that can capture the underlying patterns in the data.
- **Feature Engineering:** Adding more relevant features or transforming existing ones can help the model learn better.
- **Reduce Regularization:** If regularization is too strong, it can prevent the model from learning the data properly.
- **Increase Training Time:** Sometimes, training the model for more epochs can improve its performance

FEATURE SCALING

- Feature scaling in machine learning is an important pre-processing steps
  - Affect performance of the model
- The difference in range of values of features may cause one feature to dominate other.
- The most commonly used techniques:
  - Normalization
  - Standardization.

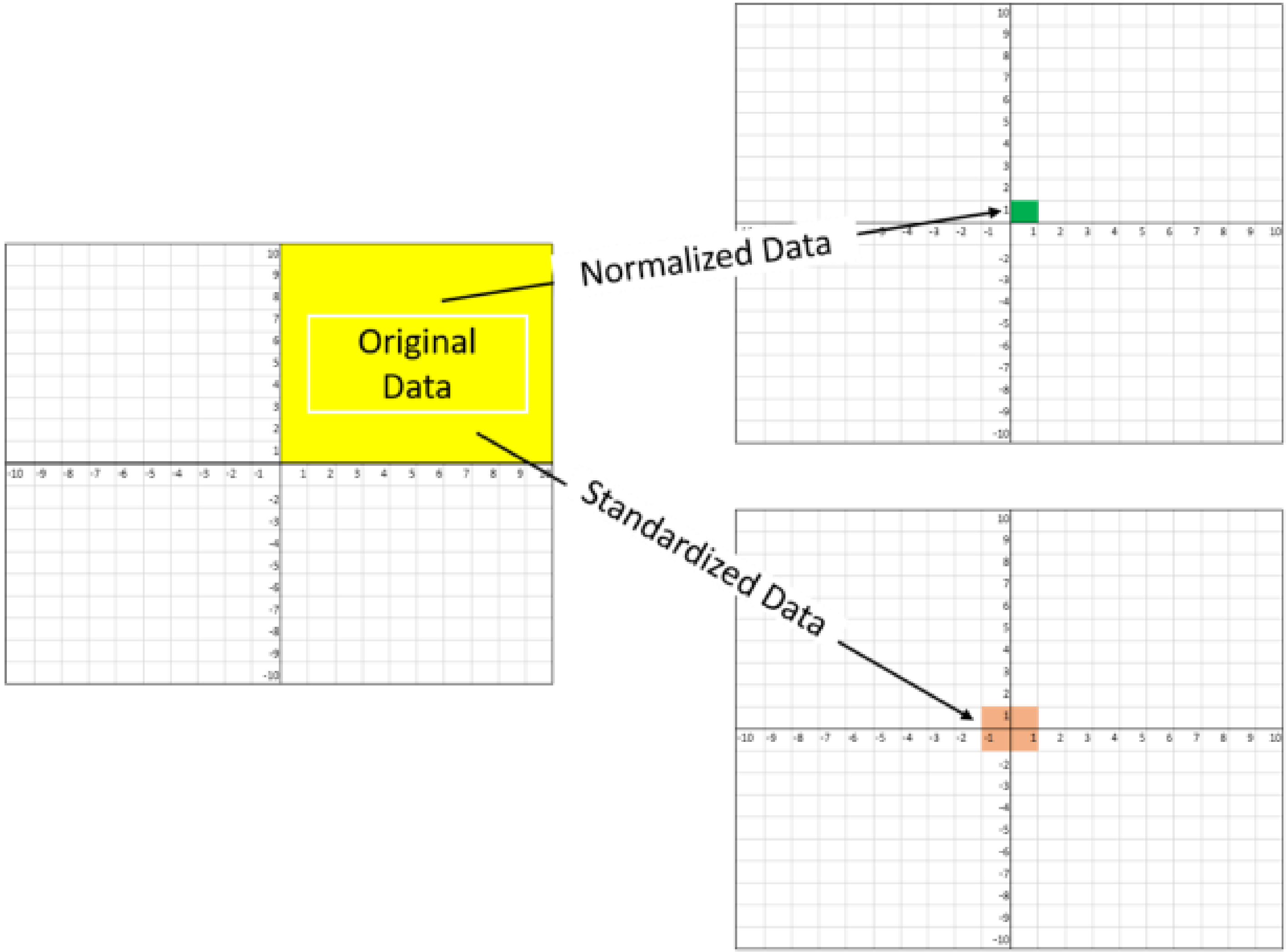
FEATURE SCALING

- **Normalization:** The values of each feature are bound between two numbers, e.g. [0,1] or [-1,1].
  - Min-Max Normalization

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

- **Standardization** transforms the data to have zero mean and a variance of 1
  - Make our data **unitless**.

$$x_{new} = \frac{x - \mu}{\sigma}$$


Original Data
Normalized Data
Standardized Data

TRAIN, TEST AND EVALUATE MODEL

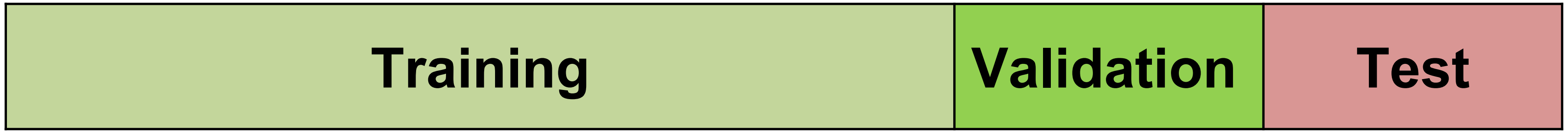
- **Training Data Set**: This dataset trains the model by helping it adjust parameters to minimize errors and learn patterns and relationships.

- **Validation Data Set:** After training, the model is evaluated on validation data to fine-tune hyperparameters and prevent overfitting. Validation set helps decide when to stop training and select the best model version.

- **Test Data Set:** The test dataset provides an unbiased evaluation of the model's final performance on unseen data and is not used during training or validation.

TRAIN, TEST AND EVALUATE MODEL

- Generally, the data is split into 3 three subsets:
  - **Training**, **Validation** and **Testing data sets**.
- **Training data set**: Used to train the model
- **Validation data set:** Tune the parameters of the model.
- **Test data set:** Test the performance of classifier on unseen data



Training

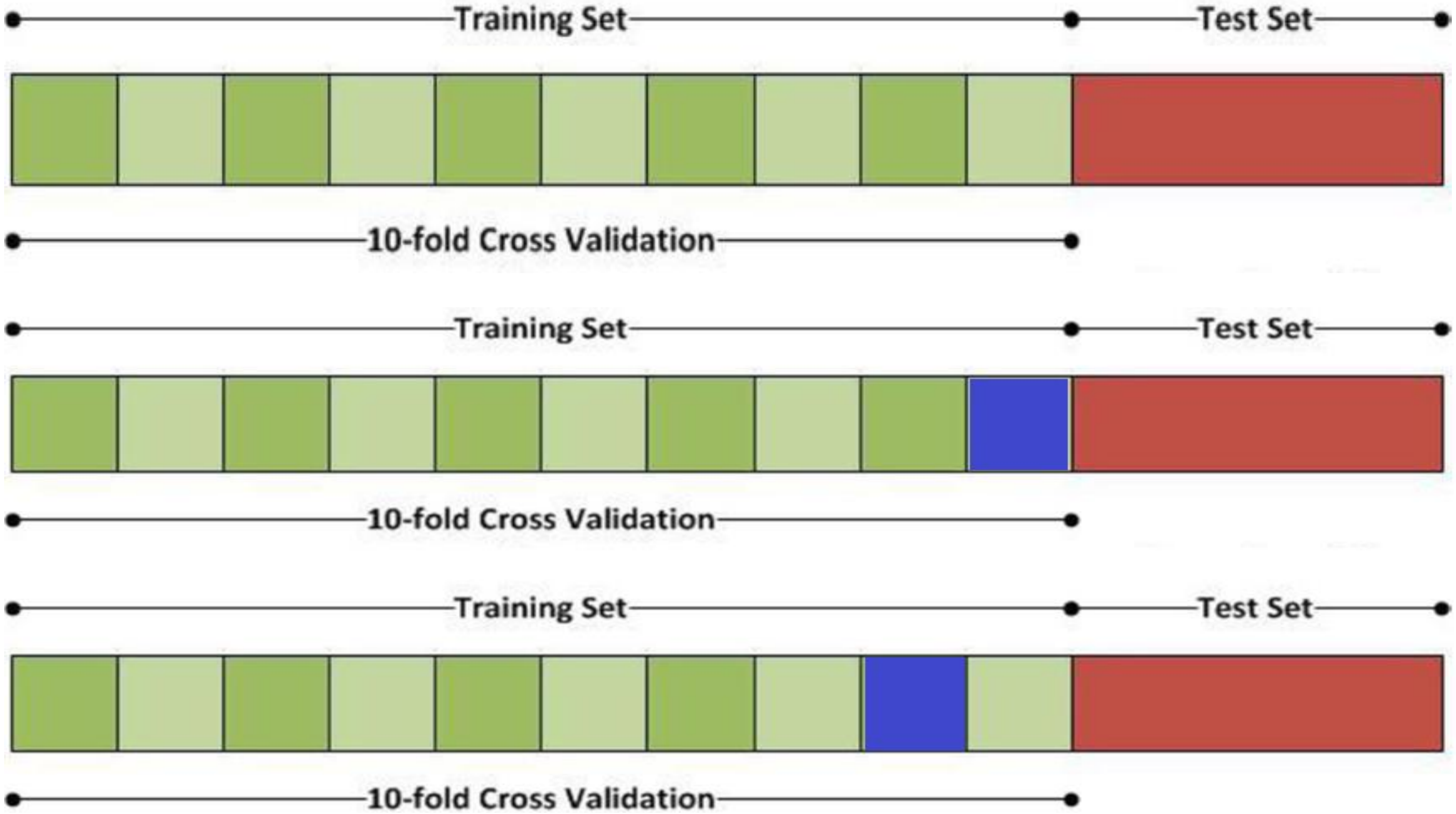
Validation

Test

TRAIN, TEST AND EVALUATE MODEL

- Cross-Validation
- Set aside some portion of the data for validation and Train on rest of it.
- **LOOCV (Leave One Out Cross Validation)**
  - Perform training on the whole training data set but leaves only one sample for validation
- **K-Fold Cross Validation**
  - The data-set into split into k subsets(folds)
  - Perform training on the all the subsets but leave one(k-1)
  - Iterate for all folds

TRAIN, TEST AND EVALUATE MODEL



CLASSIFICATION VS LOGISTIC REGRESSION

Classification

- **Purpose:** Classification is used to categorize data into predefined classes or groups. For example, it can classify emails as spam or not spam, or diagnose diseases based on symptoms.
- **Applications:** It's widely used in various fields such as medical diagnosis, fraud detection, and image recognition.
- **Types:** There are several types of classification algorithms, including decision trees, support vector machines, and neural networks.

Thank You ☺