جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

قسم الانظمة الطبية الذكية

المرحلة الثالثة

Subject: Artificial Intelligence AII

Class: Third

Lecturers: Dr. Muneera Abed Hadi, M. Sc. Rouaa Safi , M. Sc. Ansam Ali

University of Information Technology and Communications
College of Medical Informatics
Intelligent Medical Systems Department

Machine Learning Course

Lecture Three

By: Dr. Muneera Abed Hmdi

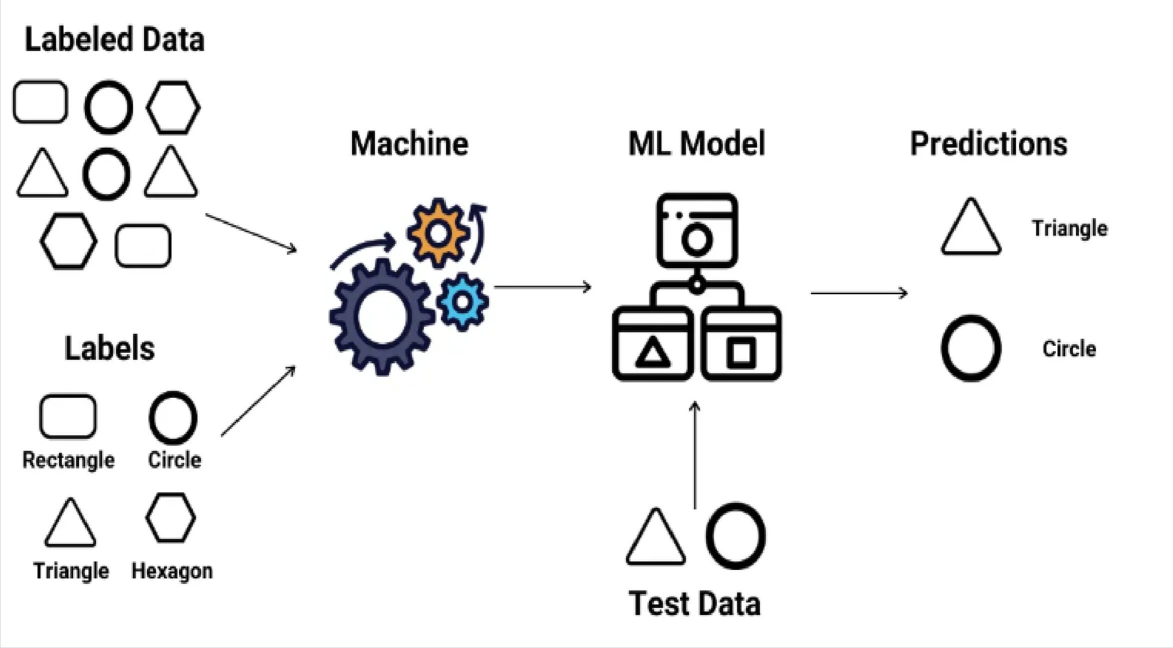
What is Supervised Machine Learning?

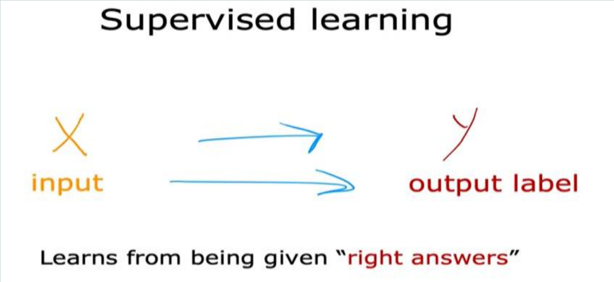
- ***Machine learning*** is creating enormous economic value today, and it consist of two main types of learning are supervised and unsupervised. About 99% of the economic value created by machine learning today is through supervised machine learning.

- ***Supervised learning*** forms the foundation of many machine learning applications, enabling computers to learn from labeled examples and make predictions on unseen data. By understanding the principles and techniques of supervised learning, we can leverage its power to solve a wide range of real-world problems and drive innovation across various industries.

What is Supervised Machine Learning?

- ***Supervised learning*** involves training a model on a labeled dataset, where each example consists of input data and corresponding output labels. The goal is for the model to learn the mapping between inputs and outputs, enabling it to make predictions on unseen data accurately.

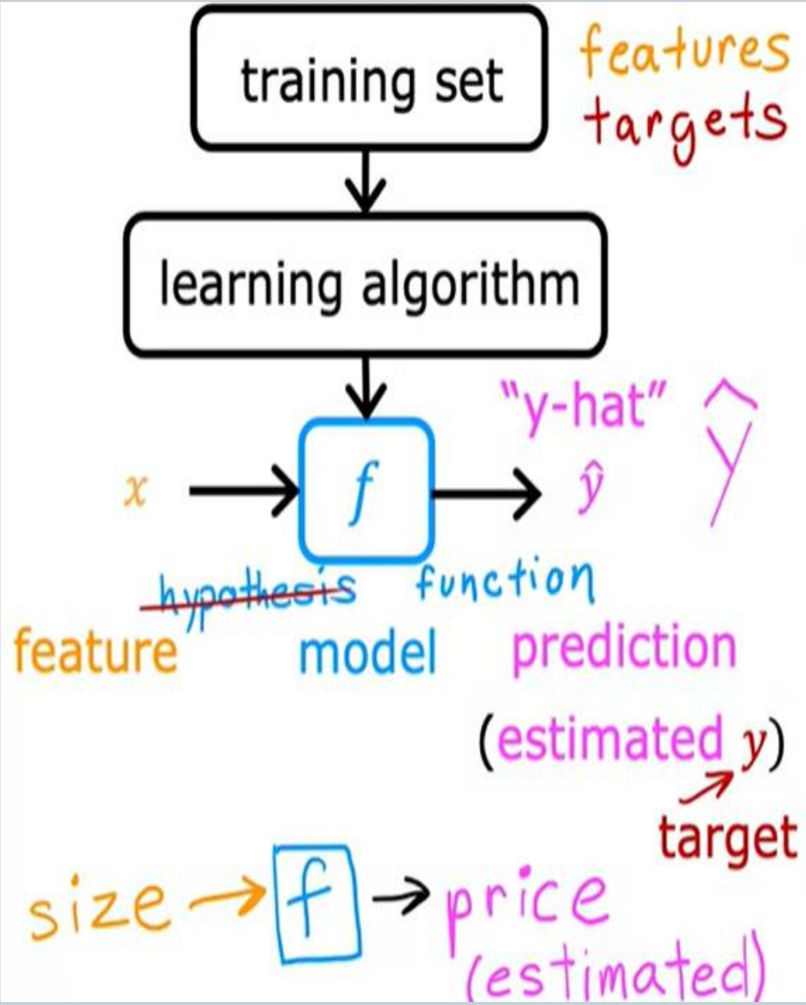




Supervised Machine Learning

- In supervised learning, the model learns by ***comparing its predictions with the actual answers*** provided in the training data.

- Over time, it adjusts itself to ***minimize errors and improve accuracy.***



Supervised Machine Learning

- Goal of ***supervised learning*** is to make accurate predictions when given new, unseen data. ***For example***, if a model is trained to recognize handwritten digits, it will use what it learned to correctly identify new numbers it hasn't seen before.

- It can be applied in various forms, including supervised learning ***classification*** and supervised learning ***regression***, making it a crucial technique in the field of artificial intelligence and supervised data mining.

Supervised Machine Learning

- A fundamental concept in supervised machine learning is learning a class from examples. This involves providing the model with examples where the ***correct label*** is known, such as learning to ***classify images of cats and dogs*** by being shown labeled examples of both. ***The model then learns the distinguishing features of each class and applies this knowledge to classify new images.***

Supervised Machine Examples

Input (X)	Output (Y)	Application
email	spam? (0/1)	spam filtering
audio	text transcripts	speech recognition
English	Spanish	machine translation
ad, user info	click? (0/1)	online advertising
image, radar info	position of other cars	self-driving car
image of phone	defect? (0/1)	visual inspection

How Supervised Machine Learning Works?

- ***Supervised learning algorithm consists of input features and corresponding output labels, and it works through:***

- ***Training Data:*** The model is provided with a training dataset that includes ***input data (features)*** and corresponding ***output data (labels or target variables)***.

- ***Learning Process:*** The algorithm processes the training data, learning the patterns and relationships between the input features and the output labels. This is achieved by ***adjusting the model's parameters to minimize the difference between its predictions and the actual labels.***

How Supervised Machine Learning Works?

- ***Supervised learning algorithm consists of input features and corresponding output labels, and it works through:***

  - ***Model Building:*** Based on the labeled data, the algorithm builds a model that can generalize from the training examples ***to make predictions on new, unseen data.***

  - ***Prediction Phase:*** Once the model is trained, it can be used to make predictions on new data. ***The model takes an input, processes it through its learned knowledge, and produces an output prediction***.

Supervised Machine Learning

- ***After training,*** the model is evaluated using a test dataset to measure its accuracy and performance. Then the model's performance is optimized by adjusting parameters and using techniques like ***cross-validation*** to balance bias and variance. ***This ensures the model generalizes well to new, unseen data.***

Hint: why we use cross validation?

- ***When the entire data is used for training the model using different algorithms, the problem of evaluating the models and selecting the most optimal model remains.***

- ***Cross validation*** is an important step in the machine learning process and helps to ensure that the model selected for deployment is robust and generalizes well to new data.

Cross-validation Techniques

- ***Cross validation*** is a technique used in machine learning to evaluate the performance of a model on ***unseen data***. It involves dividing the dataset into multiple ***folds or subsets***, using one of these folds as a ***validation set***, and training the model on the ***remaining folds***. This process is repeated multiple times, each time using a different fold as the validation set. Finally, ***the results from each validation step are averaged to produce a more robust estimate of the model's performance.***

What is the main purpose of using cross-validation?

- The main purpose of cross validation is to prevent ***overfitting***, which occurs when a model is trained too well on the training data and performs poorly on new, unseen data. By evaluating the model on multiple validation sets, cross validation provides a more realistic estimate of the model's generalization performance, i.e., ***its ability to perform well on new, unseen data.***

Cross-validation Techniques

- ***There are several types of cross validation techniques such as:***

1. Resubstitution validation

- If all the data is used for training the model and the error rate is evaluated based on outcome vs. actual value from the same training data set, this error is called the resubstitution error. This technique is called the resubstitution validation technique.

2. Hold-out Validation

- It is a mechanism of splitting the dataset into ***training and test datasets***. The model is trained on the training set and then tested on the testing set to get the most optimal model ***(60/40 or 70/30 or 80/20 data splitting)***.

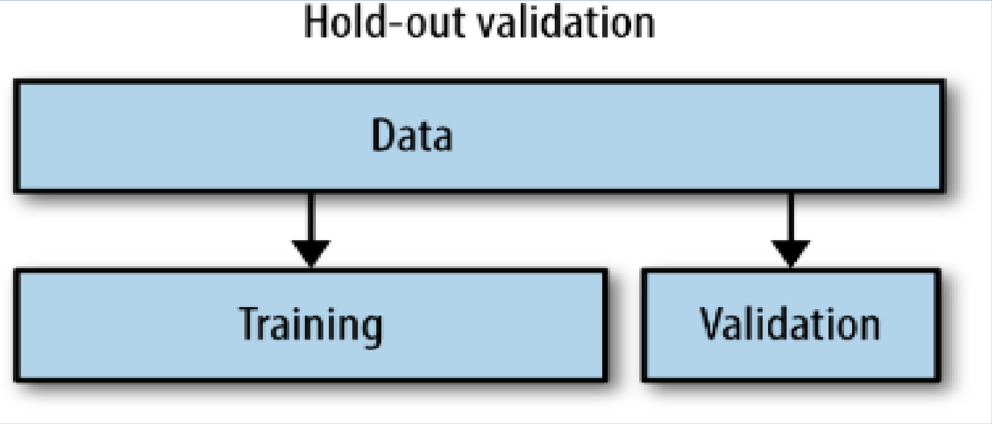
- This approach is often used when the data set is small and there is not enough data to split into three sets (training, validation, and testing).

Cross-validation Techniques

2. Hold-out Validation

- This approach has the advantage of being simple to implement, but it can be sensitive to how the data is divided into two sets. If the split is not random, then the results may be biased.

- In this case, there is a likelihood that uneven distribution of different classes of data is found in training and test dataset. To fix this***, the training and test dataset is created with equal distribution of different classes of data***. This process is called stratification.



Cross-validation Techniques

- ***To perform data splitting while ensuring equal distribution of different classes (also known as stratified sampling), you can use tools like Scikit-learn in Python. This approach maintains the same class proportion in both the training and test datasets, which is crucial for classification tasks.***

```
from sklearn.model_selection import train_test_split

# Example: X = features, y = labels/classes
X = [[1], [2], [3], [4], [5], [6]]
y = [0, 0, 1, 1, 2, 2]  # Classes: 0, 1, 2

# Stratified Split
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.3,          # 30% Test Data
    stratify=y,             # Ensures equal class distribution
    random_state=42         # For reproducibility
)

# Result
print("Train Labels:", y_train)
print("Test Labels:", y_test)
```

Key Parameters

- ***test_size:*** Proportion of the dataset for testing (e.g., ***0.3 for 30%).***
- ***stratify:*** Set this to the ***target variable y*** to maintain class balance.
- ***random_state:*** Ensures you get the same split every time you run the code.

Why Stratification Matters?

- ***Without stratification***, you might end up with ***imbalanced splits***—some classes might be ***overrepresented*** in the training set and ***underrepresented*** in the test set, leading to biased performance metrics.

Real Dataset Example (Iris Dataset)

```python
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
import pandas as pd

# Load Iris dataset
iris = load_iris()
X = iris.data
y = iris.target

# Stratified Split
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.3,
    stratify=y,
    random_state=42
)

# Check Class Distribution
print("Original Class Distribution:\n", pd.Series(y).value_counts())
print("Training Set Class Distribution:\n", pd.Series(y_train).value_counts())
print("Test Set Class Distribution:\n", pd.Series(y_test).value_counts())
```

Expected Output

- ***You'll see that the class distribution in both the training and test sets mirrors the original dataset distribution.***

```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
import pandas as pd

# Load Iris dataset
iris = load_iris()
X = iris.data
y = iris.target

# Stratified Split
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.3,
    stratify=y,
    random_state=42
)

# Check Class Distribution
print("Original Class Distribution:\n", pd.Series(y).value_counts())
print("Training Set Class Distribution:\n", pd.Series(y_train).value_counts())
print("Test Set Class Distribution:\n", pd.Series(y_test).value_counts())
```

Cancer Classification Dataset (Breast Cancer Wisconsin Dataset)

- ***This dataset is used for binary classification (malignant vs. benign). We'll apply the same stratified split.***

```python
from sklearn.datasets import load_breast_cancer

# Load Breast Cancer Dataset
cancer = load_breast_cancer()
X = cancer.data
y = cancer.target  # 0 = malignant, 1 = benign

# Stratified Split
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.3,
    stratify=y,
    random_state=42
)

# Check Class Distribution
print("Original Class Distribution:\n", pd.Series(y).value_counts())
print("Training Set Class Distribution:\n", pd.Series(y_train).value_counts())
print("Test Set Class Distribution:\n", pd.Series(y_test).value_counts())
```

Why This Is Important in Cancer Classification:

- ***Malignant cases** (class 0) are often fewer than **benign cases** (class 1).*
- ***Without stratification**, the test set might miss malignant cases, leading to misleading high accuracy but poor real-world performance.*

- ***For more Information related to Python Pandas Series and all the basic operations which can be performed on Pandas Series pass through the below link:***

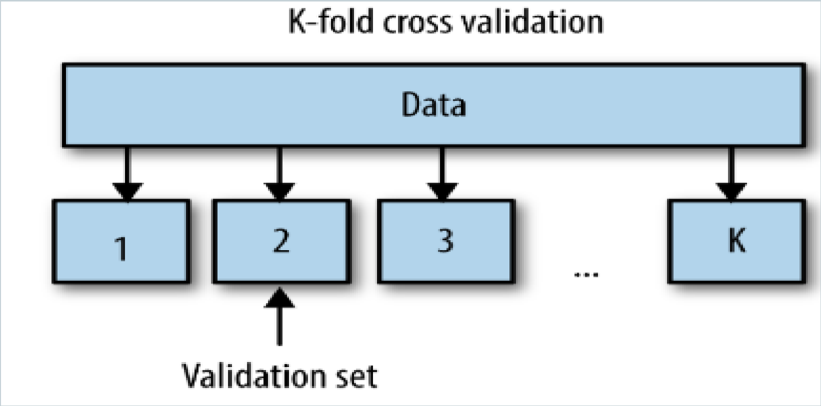
https://www.geeksforgeeks.org/python-pandas-series/

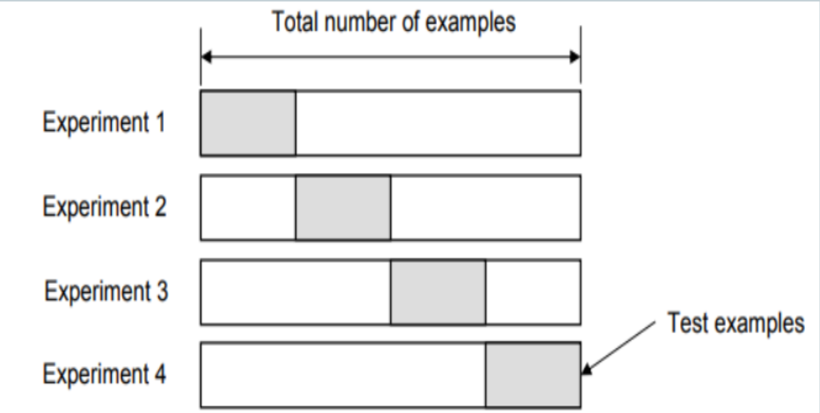
Cross-validation Techniques

3. K-Fold Cross-Validation

- In this technique, ***k-1 folds*** are used for training and the ***remaining one*** is used for testing as shown in the picture given below.

- The advantage is that entire data is used for training and testing. The error rate of the model is average of the error rate of each iteration. The error rate could be improved by using ***stratification technique***.

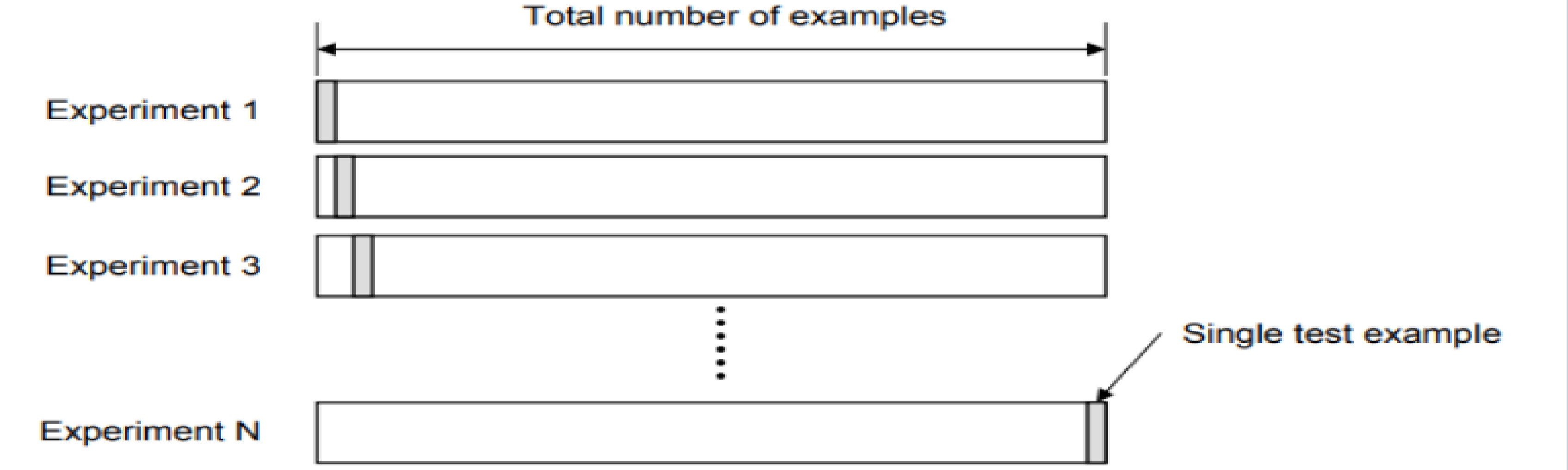




Cross-validation Techniques

4. LOOCV (Leave One Out Cross Validation)

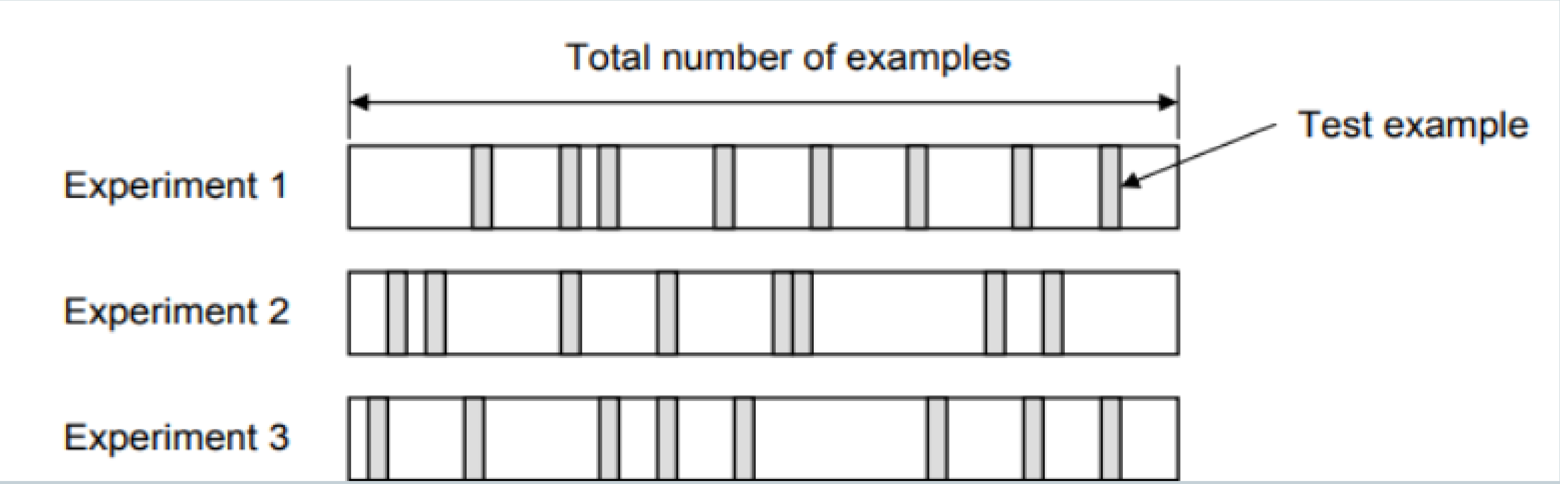
- In this technique, all the data except one record is used for training and one record is used for testing. This process is repeated for N times if there are N records. The advantage is that entire data is used for training and testing. The error rate of the model is average of the error rate of each iteration.



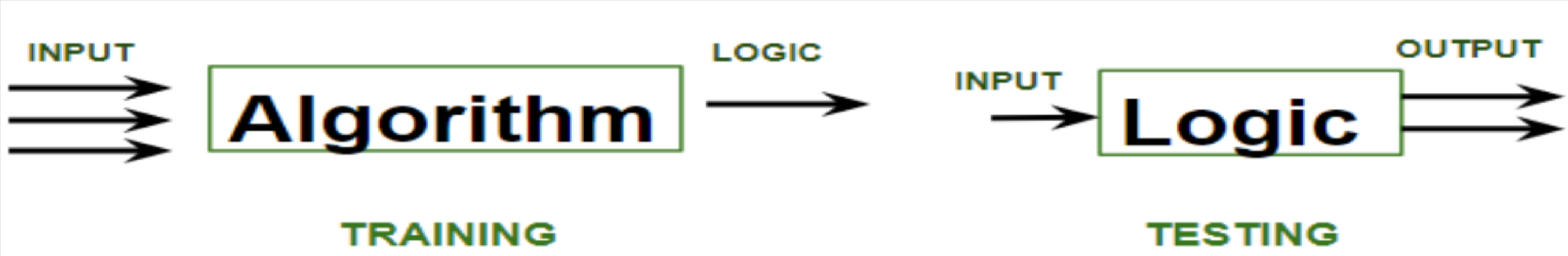
Types of Cross-validation process in ML?

5. Random Subsampling

- In this technique, multiple sets of data are randomly chosen from the dataset and combined to form a test dataset. The remaining data forms the training dataset. The following diagram represents the random subsampling validation technique. The error rate of the model is the average of the error rate of each iteration.



Training Phase vs. Testing Phase



- ***Training phase*** involves feeding the algorithm labeled data, where each data point is paired with its correct output. The algorithm learns to ***identify patterns and relationships between the input and output data***.

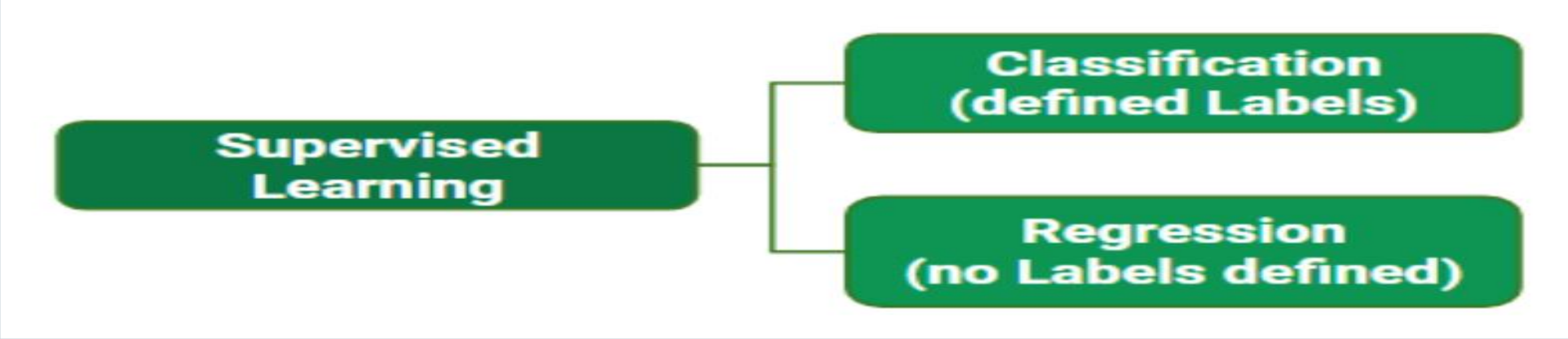
- ***Testing phase*** involves feeding the algorithm new, unseen data and ***evaluating its ability to predict the correct output*** based on the learned patterns.

Types of Supervised Learning

- ***Supervised learning can be applied to solve two main types of problems:***

- ***Classification:*** Where the output is a categorical variable (e.g., spam vs. non-spam emails, yes vs. no).

- ***Regression:*** Where the output is a continuous variable (e.g., predicting house prices, stock prices).



Types of Supervised Learning

- While training the model, ***data is usually split in the ratio of 80:20 i.e., 80% as training data and the rest as testing data. In training data, we feed input as well as output for 80% of data.*** The model learns from training data only. We ***use different supervised learning algorithms to build our model***. First understand the classification and regression data through the table below:

User ID	Gender	Age	Salary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	1
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	1
15728773	Male	27	58000	1
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	1
15727311	Female	35	65000	0
15570769	Female	26	80000	1
15606274	Female	26	52000	0
15746139	Male	20	86000	1
15704987	Male	32	18000	0
15628972	Male	18	82000	0
15697686	Male	29	80000	0
15733883	Male	47	25000	1

Figure A: CLASSIFICATION

Temperature	Pressure	Relative Humidity	Wind Direction	Wind Speed
10.69261758	986.882019	54.19337313	195.7150879	3.278597116
13.59184184	987.8729248	48.0648859	189.2951202	2.909167767
17.70494885	988.1119385	39.11965597	192.9273834	2.973036289
20.95430404	987.8500366	30.66273218	202.0752869	2.965289593
22.9278274	987.2833862	26.06723423	210.6589203	2.798230886
24.04233986	986.2907104	23.46918024	221.1188507	2.627005816
24.41475295	985.2338867	22.25082295	233.7911987	2.448749781
23.93361956	984.8914795	22.35178837	244.3504333	2.454271793
22.68800023	984.8461304	23.7538641	253.0864716	2.418341875
20.56425726	984.8380737	27.07867944	264.5071106	2.318677425
17.76400389	985.4262085	33.54900114	280.7827454	2.343950987
11.25680746	988.9386597	53.74139903	68.15406036	1.650191426
14.37810685	989.6819458	40.70884681	72.62069702	1.553469896
18.45114201	990.2960205	30.85038484	71.70604706	1.005017161
22.54895853	989.9562988	22.81738811	44.66042709	0.264133632
24.23155922	988.796875	19.74790765	318.3214111	0.329656571

Figure B: REGRESSION

Types of Supervised Learning

- ***Both the figures in the previous slied have labelled data set as follows:***

- ***Figure A:*** It is a dataset of a shopping store that is useful in predicting whether a customer will purchase a particular product under consideration or not based on his/ her gender, age, and salary.
  Input: Gender, Age, Salary
  Output: Purchased i.e., 0 or 1; 1 means yes, the customer will purchase and 0 means that the customer won't purchase it.

- ***Figure B:*** It is a Meteorological dataset that serves the purpose of predicting wind speed based on different parameters.
  Input: Dew Point, Temperature, Pressure, Relative Humidity, Wind Direction
  Output: Wind Speed

Key Steps for *Training a Supervised Learning Model*

- These types of supervised learning in machine learning ***vary based on the problem you're trying to solve and the dataset you're working with***.

- In ***classification*** problems, the task is to assign output to predefined classes, while ***regression*** problems involve predicting numerical outcomes.

- Training a model for supervised learning involves crucial steps, each designed ***to prepare the model to make accurate predictions or decisions based on labeled data.***

Key Steps for *Training a Supervised Learning Model*

- ***Below are the key steps involved in training a model for supervised machine learning:***
- ***Data Collection and Preprocessing***: Gather a labeled dataset consisting of input features and target output labels. Clean the data, handle missing values, and scale features as needed to ensure high quality for supervised learning algorithms.
- ***Splitting the Data***: Divide the data into training and test using cross validation technique such as holdout cross validation with training set (80%) and the test set (20%).
- ***Choosing the Model***: Select appropriate algorithms based on the problem type (C/R). This step is crucial for effective supervised learning in ML.
- ***Training the Model***: Feed the model input data and output labels, allowing it to learn patterns and relationship by adjusting internal parameters.