Al-Mustaqbal University
College of Sciences
Intelligent Medical System Department

جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

كلية العلوم
قسم الانظمة الطبية الذكية

Lecture: (2)

Supervised Learning: Algorithms and Techniques for Predictive Modeling

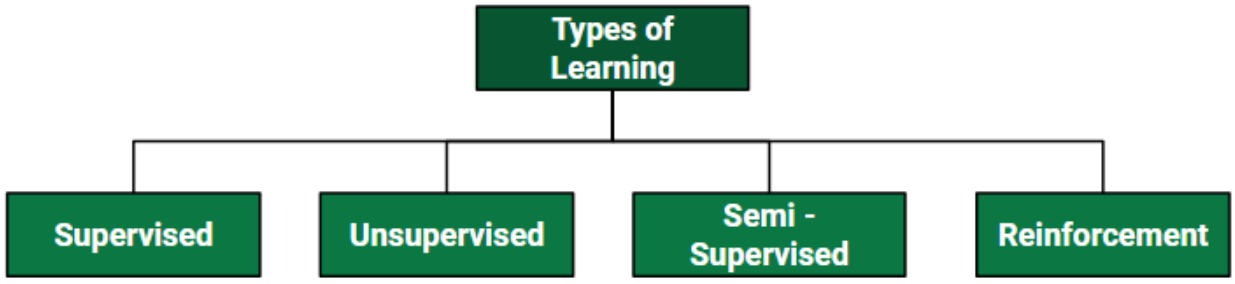
Subject: Artificial Intelligence
Class: Third
Lecturer: Dr. Maytham N. Meqdad

Supervised Learning: Algorithms and Techniques for Predictive Modeling

Supervised Machine Learning

A machine is said to be learning from **Past Experiences** (data feed-in) with respect to some class of **tasks** if its **Performance** in a given Task improves with the Experience.

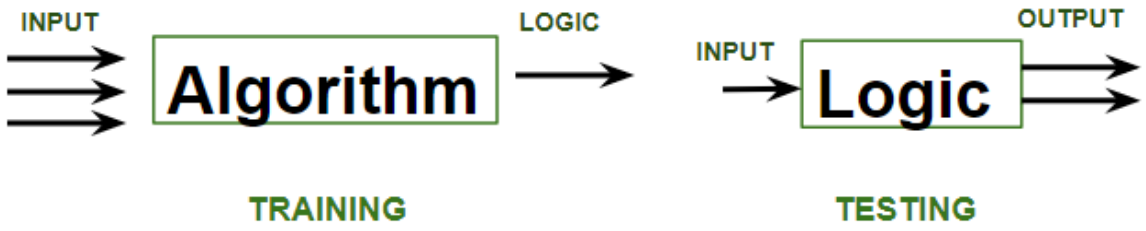
For example, assume that a machine has to predict whether a customer will buy a specific product let's say "Antivirus" this year or not. The machine will do it by looking at the **previous knowledge/past experiences** i.e. the data of products that the customer had bought every year and if he buys an Antivirus every year, then there is a high probability that the customer is going to buy an antivirus this year as well. This is how machine learning works at the basic conceptual level.



What is Supervised Machine Learning?

Supervised learning is a machine learning technique that is widely used in various fields such as finance, healthcare, marketing, and more. It is a form of machine learning in which the algorithm is trained on labeled data to make predictions or decisions based on the data inputs.In supervised learning, the algorithm learns a mapping between the input and output data. This mapping is learned from a labeled dataset, which consists of pairs of input and output data. The algorithm tries to learn the relationship between the input and output data so that it can make accurate predictions on new, unseen data.

Let us discuss what learning for a machine is as shown below media as follows:



Supervised learning is where the model is trained on a labelled dataset. A **labelled** dataset is one that has both input and output parameters. In this type of learning both training and validation, datasets are labelled as shown in the figures below.

The labeled dataset used in supervised learning consists of input features and corresponding output labels. The input features are the attributes or characteristics of the data that are used to make predictions, while the output labels are the desired outcomes or targets that the algorithm tries to predict.

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

Both the above figures 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.

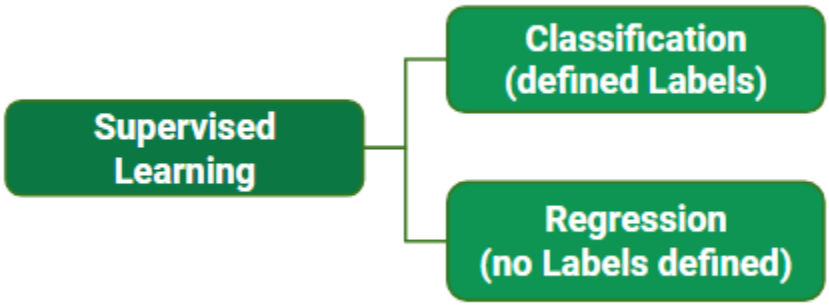
- **Figure B:** It is a Meteorological dataset that serves the purpose of predicting wind speed based on different parameters.
  Input: Dew Point, Temperature, Pressure, Relative Humidity, Wind Direction
  Output: Wind Speed

Training the system: 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 machine learning algorithms(which we will discuss in detail in the next articles) to build our model. Learning means that the model will build some logic of its own.
Once the model is ready then it is good to be tested. At the time of testing, the input is fed from the remaining 20% of data that the model has never seen before, the model will predict some value and we will compare it with the actual output and calculate the accuracy.

Types of Supervised Learning Algorithm

Supervised learning is typically divided into two main categories: regression and classification. In regression, the algorithm learns to predict a continuous output value, such as the price of a house or the temperature of a city. In classification, the algorithm learns to predict a categorical output variable or class label, such as whether a customer is likely to purchase a product or not.

One of the primary advantages of supervised learning is that it allows for the creation of complex models that can make accurate predictions on new data. However, supervised learning requires large amounts of labeled training data to be effective. Additionally, the quality and representativeness of the training data can have a significant impact on the accuracy of the model.
Supervised learning can be further classified into two categories:



Regression

Regression is a supervised learning technique used to predict continuous numerical values based on input features. It aims to establish a functional relationship between independent variables and a dependent variable, such as predicting house prices based on features like size, bedrooms, and location.
The goal is to minimize the difference between predicted and actual values using algorithms like Linear Regression, Decision Trees, or Neural Networks, ensuring the model captures underlying patterns in the data.

Classification

Classification is a type of supervised learning that categorizes input data into predefined labels. It involves training a model on labeled examples to learn patterns between input features and output classes. In classification, the target variable is a categorical value. For example, classifying emails as spam or not.
The model's goal is to generalize this learning to make accurate predictions on new, unseen data. Algorithms like Decision Trees, Support Vector Machines, and Neural Networks are commonly used for classification tasks.

NOTE: There are common Supervised Machine Learning Algorithm that can be used for both regression and classification task.

Supervised Machine Learning Algorithm

Supervised learning can be further divided into several different types, each with its own unique characteristics and applications. Here are some of the most common types of supervised learning algorithms:

- **Linear Regression** : Linear regression is a type of regression algorithm that is used to predict a continuous output value. It is one of the simplest and most widely used algorithms in supervised learning. In linear regression, the algorithm tries to find a linear relationship between the input features and the output value. The output value is predicted based on the weighted sum of the input features.

- **Logistic Regression** : Logistic regression is a type of classification algorithm that is used to predict a binary output variable. It is commonly used in machine learning applications where the output variable is either true or false, such as in fraud detection or spam filtering. In logistic regression, the algorithm tries to find a linear relationship between the input features and the output variable. The output variable is then transformed using a logistic function to produce a probability value between 0 and 1.

- **Decision Trees** : Decision tree is a tree-like structure that is used to model decisions and their possible consequences. Each internal node in the tree represents a decision, while each leaf node represents a possible outcome. Decision trees can be used to model complex relationships between input features and output variables.
  A decision tree is a type of algorithm that is used for both classification and regression tasks.
  - **Decision Trees Regression:** Decision Trees can be utilized for regression tasks by predicting the value linked with a leaf node.
  - **Decision Trees Classification:** Random Forest is a machine learning algorithm that uses multiple decision trees to improve classification and prevent overfitting.

- **Random Forests** : Random forests are made up of multiple decision trees that work together to make predictions. Each tree in the forest is trained on a different subset of the input features and data. The final prediction is made by aggregating the predictions of all the trees in the forest.
  Random forests are an ensemble learning technique that is used for both classification and regression tasks.
  - **Random Forest Regression** : It combines multiple decision trees to reduce overfitting and improve prediction accuracy.
  - **Random Forest Classifier:** Combines several decision trees to improve the accuracy of classification while minimizing overfitting.

- **Support Vector Machine(SVM)** : The SVM algorithm creates a hyperplane to segregate n-dimensional space into classes and identify the correct category of new data points. The extreme cases that help create the hyperplane are called support vectors, hence the name Support Vector Machine.
  A Support Vector Machine is a type of algorithm that is used for both classification and regression tasks
  - **Support Vector Regression:** It is a extension of Support Vector Machines (SVM) used for predicting continuous values.
  - **Support Vector Classifier:** It aims to find the best hyperplane that maximizes the margin between data points of different classes.

- K-Nearest Neighbors **(KNN)** : KNN works by finding k training examples closest to a given input and then predicts the class or value based on the majority class or average value of these neighbors. The performance of KNN can be influenced by the choice of k and the distance metric used to measure proximity. However, it is intuitive but can be sensitive to noisy data and requires careful selection of k for optimal results.
  A K-Nearest Neighbors (KNN) is a type of algorithm that is used for both classification and regression tasks.
  - **K-Nearest Neighbors Regressio** n: It predicts continuous values by averaging the outputs of the k closest neighbors.

- **K-Nearest Neighbors Classification:** Data points are classified based on the majority class of their k closest neighbors.

- Gradient Boosting : Gradient Boosting combines weak learners, like decision trees, to create a strong model. It iteratively builds new models that correct errors made by previous ones. Each new model is trained to minimize residual errors, resulting in a powerful predictor capable of handling complex data relationships.
  A Gradient Boosting is a type of algorithm that is used for both classification and regression tasks.
  - **Gradient Boosting Regression:** It builds an ensemble of weak learners to improve prediction accuracy through iterative training.
  - **Gradient Boosting Classification:** Creates a group of classifiers to continually enhance the accuracy of predictions through iterations

Dimensions of Supervised Machine Learning Algorithm

When discussing the dimensions of supervised machine learning algorithms, we refer to various aspects that characterize and influence the performance, complexity, and applicability of these algorithms. These dimensions provide a framework for understanding how an algorithm operates, its strengths and weaknesses, and how it can be optimized for specific tasks. Here's an explanation of key dimensions:

1. Complexity

- **Model Complexity:** This refers to the intricacy of the algorithm's structure. Simple models like linear regression are easy to understand and interpret, but they may not capture complex relationships in data. On the other hand, more complex models like deep neural networks can capture intricate patterns but are harder to interpret and require more computational resources.

- **Overfitting and Underfitting:** Complexity is closely tied to overfitting (where the model learns the noise in the training data) and underfitting (where the model is too simple to capture the underlying pattern). Balancing complexity is crucial for creating a model that generalizes well to unseen data.

2. Interpretability

- **Transparency:** Some algorithms, like decision trees, are highly interpretable, meaning their decision-making process can be easily understood by humans. Other algorithms, like neural networks, operate as "black boxes," where the reasoning behind a prediction is not easily discernible.

- **Feature Importance:** In interpretability, understanding which features (input variables) are most influential in making predictions is important. This can be straightforward in algorithms like linear regression, where coefficients directly indicate feature importance, but more challenging in complex models like ensemble methods or neural networks.

3. Scalability

- **Data Size:** Scalability refers to the algorithm's ability to handle large datasets. Some algorithms, like K-Nearest Neighbors (KNN) , may become computationally expensive as the dataset grows, while others, like linear models or decision trees, can scale more efficiently with large datasets.

- **Dimensionality:** Scalability also considers how well an algorithm performs as the number of features (dimensions) increases. High-dimensional data can lead to challenges like the "curse of dimensionality," where the data becomes sparse, and distance measures (used in algorithms like KNN) become less meaningful.

4. Flexibility

- **Adaptability to Different Data Types:** Flexibility indicates how well an algorithm can be adapted to different types of data (e.g., categorical, continuous, or mixed data types). Some algorithms are inherently more flexible, like decision trees , which can handle both categorical and continuous data.

- **Handling Missing Data:** Flexibility also involves how well an algorithm can handle incomplete data. Some algorithms can naturally handle missing values, while others may require data imputation or other preprocessing steps.

5. Training Time and Computational Efficiency

- **Algorithm Efficiency:** This dimension measures the time and computational resources required to train the model. Algorithms like linear regression or Naive Bayes are generally fast to train, while more complex models like support vector machines (SVM) or neural networks might require significant computational power and time, especially on large datasets.

- **Resource Requirements:** The efficiency of an algorithm also depends on the hardware and software environment. Some algorithms can be parallelized to speed up training, while others may be limited by memory or processing power.

Training a Model for Supervised Learning

Training a model for supervised learning involves several crucial steps, each designed to prepare the model to make accurate predictions or decisions based on labeled data. Below is a detailed explanation of the process:

1. Data Collection and Preprocessing

- **Data Collection:** The first step is gathering the data that will be used to train the model. In supervised learning, this data must be labeled, meaning that each data point should have corresponding input features and an associated output label (the correct answer).

- **Data Preprocessing:** Raw data often contains noise, missing values, or irrelevant features. Preprocessing involves cleaning the data, handling missing values, normalizing or scaling the data, and selecting the most relevant features. This step is critical because the quality of the input data directly impacts the model's performance.

2. Splitting the Data

- **Training and Testing Split:** The dataset is typically split into two parts: the training set and the testing set. A common split ratio is 80:20, where 80% of the data is used for training the model, and 20% is reserved for testing. The training set is used to teach the model, while the testing set is used to evaluate its performance on unseen data.

- **Validation Set (Optional):** Sometimes, the training data is further split into a training set and a validation set. The validation set is used to fine-tune the model parameters during training without exposing the model to the testing set.

3. Choosing the Model

- **Algorithm Selection:** Depending on the type of problem (regression or classification), an appropriate supervised learning algorithm is selected. Common algorithms include:
  - **Linear Regression** for predicting continuous values.
  - **Logistic Regression** for binary classification tasks.
  - **Decision Trees** for both regression and classification.
  - **Support Vector Machines (SVM)** for classification and regression.

- **Random Forests** and **Gradient Boosting Machines (GBM)** for ensemble learning.

- **Model Initialization:** The model is initialized with default parameters, or initial parameters are set based on prior knowledge or experiments.

4. Training the Model

- **Feeding Data:** The training data (input features and output labels) is fed into the model. The model uses this data to learn patterns and relationships between the inputs and outputs.

- **Learning Process:** The model adjusts its internal parameters (e.g., weights in a linear model or node decisions in a tree) to minimize the difference between its predictions and the actual output labels. This process is often guided by a loss function (e.g., Mean Squared Error for regression, Cross-Entropy Loss for classification) that quantifies the error.

- **Optimization:** The model uses an optimization algorithm, such as **Gradient Descent** , to iteratively update its parameters and reduce the loss. This step is repeated for multiple iterations, or epochs, until the model's performance stabilizes.

5. Evaluating the Model

- **Testing:** Once the model is trained, it is evaluated on the testing set. The input features from the testing set are fed into the model, which predicts the outputs. These predicted outputs are then compared to the actual labels to assess the model's accuracy.

- **Performance Metrics:** Several metrics are used to evaluate the model's performance:
  - **Accuracy:** The proportion of correct predictions.
  - **Precision, Recall, and F1-Score:** Metrics used in classification tasks to evaluate the model's ability to correctly identify positive instances.
  - **Mean Squared Error (MSE):** A common metric for regression tasks to measure the average squared difference between predicted and actual values.
  - **Confusion Matrix:** A table used in classification to visualize the performance of the model across different classes.

6. Hyperparameter Tuning

- **Hyperparameter Optimization:** Hyperparameters are model parameters set before training (e.g., the learning rate in gradient descent, the number of trees in a random forest). Tuning these hyperparameters can significantly impact the model's performance.

Techniques like **Grid Search** or **Random Search** are commonly used to find the best hyperparameter values.

- **Cross-Validation:** To ensure that the model generalizes well to new data, cross-validation is often used. This involves dividing the training data into several subsets and training the model multiple times, each time using a different subset as the validation set. The results are averaged to provide a more robust evaluation.

7. Final Model Selection and Testing

- **Final Training:** After hyperparameter tuning, the model is retrained on the entire training set (including the validation set, if used) using the best-found hyperparameters.
- **Final Testing:** The model is then tested on the testing set to evaluate its final performance. This step determines how well the model is likely to perform on real-world, unseen data.

8. Model Deployment

- **Deployment:** Once the model has been thoroughly tested and validated, it is ready to be deployed into a production environment where it can make predictions on new, unseen data.
- **Monitoring:** Post-deployment, the model's performance should be continuously monitored to ensure it continues to perform well. Over time, the model may need to be retrained with new data to adapt to changes in the underlying patterns.

9. Iterative Improvement

- **Feedback Loop:** In practice, model training is an iterative process. Based on the performance metrics and feedback, the model may need to be refined or retrained with more data, different features, or alternative algorithms to achieve better results.

By following these steps, a supervised learning model can be effectively trained to make accurate predictions, whether it's classifying emails as spam or not, predicting house prices, or any other task that requires mapping inputs to outputs based on labeled data.

A advantages of Supervised Learning

The power of supervised learning lies in its ability to accurately predict patterns and make data-driven decisions across a variety of applications. Here are some advantages listed below:

1. Labeled training data benefits supervised learning by enabling models to accurately learn patterns and relationships between inputs and outputs.

2. Supervised learning models can accurately predict and classify new data.

3. Supervised learning has a wide range of applications, including classification, regression, and even more complex problems like image recognition and natural language processing.

4. Well-established evaluation metrics, including accuracy, precision, recall, and F1-score, facilitate the assessment of supervised learning model performance.

Disadvantages of Supervised Learning

Although supervised learning methods have benefits, their limitations require careful consideration during problem formulation, data collection, model selection, and evaluation. Here are some disadvantages listed below:

1. **Overfitting :** Models can overfit training data, which leads to poor performance on new, unseen data due to the capture of noise.

2. **Feature Engineering :** Extracting relevant features from raw data is crucial for model performance, but this process can be time-consuming and may require domain expertise.

3. **Bias in Models:** Training data biases can lead to unfair predictions.

4. Supervised learning heavily depends on labeled training data, which can be costly, time-consuming, and may require domain expertise.