**UNIT II: Supervised Learning (9 hours)**

**1. Linear Regression**

* **Linear Basis Function Models**:
  + Linear models that use basis functions (like polynomials) to transform input data into higher-dimensional space. They fit a linear model in the transformed space.
  + Example: y=β0+β1ϕ1(x)+β2ϕ2(x)+...+ϵy = \beta\_0 + \beta\_1 \phi\_1(x) + \beta\_2 \phi\_2(x) + ... + \epsilony=β0​+β1​ϕ1​(x)+β2​ϕ2​(x)+...+ϵ, where ϕ(x)\phi(x)ϕ(x) represents basis functions.
* **Simple Linear Regression**:
  + The simplest form of regression, predicting a dependent variable yyy from one independent variable xxx using the formula:

y=β0+β1x+ϵy = \beta\_0 + \beta\_1 x + \epsilony=β0​+β1​x+ϵ

* + Where:
    - β0\beta\_0β0​: Intercept
    - β1\beta\_1β1​: Slope (regression coefficient)
    - ϵ\epsilonϵ: Error term
* **Multiple Linear Regression**:
  + Extends simple linear regression to predict a dependent variable yyy from multiple independent variables x1,x2,…,xnx\_1, x\_2, \dots, x\_nx1​,x2​,…,xn​:

y=β0+β1x1+β2x2+...+βnxn+ϵy = \beta\_0 + \beta\_1 x\_1 + \beta\_2 x\_2 + ... + \beta\_n x\_n + \epsilony=β0​+β1​x1​+β2​x2​+...+βn​xn​+ϵ

* **Bayesian Linear Regression**:
  + A probabilistic approach to linear regression. Instead of estimating a single set of coefficients, it computes a distribution over the parameters, incorporating prior knowledge or assumptions.
  + The result is a distribution of predictions rather than a single value.

**2. Classification**

* **Logistic Regression**:
  + A classification algorithm that models the probability of a binary outcome using the logistic function:

P(y=1∣x)=11+e−(β0+β1x)P(y = 1 | x) = \frac{1}{1 + e^{-(\beta\_0 + \beta\_1 x)}}P(y=1∣x)=1+e−(β0​+β1​x)1​

* + Outputs probabilities between 0 and 1, which are then mapped to classes.
* **k-Nearest Neighbors (k-NN)**:
  + A non-parametric method for classification. The class of a data point is determined by the majority class among its k nearest neighbors in the feature space.
  + Simple and intuitive, but computationally expensive for large datasets.
* **Decision Trees**:
  + A tree-like structure where each node represents a decision based on a feature, and branches represent possible outcomes.
  + A recursive process that splits the data based on feature thresholds, creating a hierarchy of rules for classification.
* **Random Forest Model**:
  + An ensemble of decision trees. Each tree is trained on a random subset of the data, and predictions are made by averaging (regression) or voting (classification) from all trees.
  + Helps reduce overfitting and increases model accuracy.
* **Support Vector Machines (SVM)**:
  + A classifier that finds the optimal hyperplane that separates the data into different classes, maximizing the margin between the classes.
  + Effective for high-dimensional spaces and can handle non-linear classification through the use of kernel functions.
* **Bayesian Networks**:
  + A graphical model that represents probabilistic relationships among variables. The nodes are variables, and the edges represent conditional dependencies.
  + Used for reasoning under uncertainty and decision-making.

**UNIT III: Unsupervised Learning (9 hours)**

**1. Principal Component Analysis (PCA)**

* A technique for dimensionality reduction that transforms the data into a new coordinate system, where the greatest variance in the data is captured by the first principal component, the second greatest by the second component, and so on.
* PCA helps reduce complexity while retaining most of the information, making it useful for visualization and feature extraction.

**2. Linear Discriminant Analysis (LDA)**

* A supervised technique for dimensionality reduction. Unlike PCA, LDA tries to find a projection that maximizes the separability of different classes.
* It minimizes the variance within each class and maximizes the variance between different classes, making it suitable for classification tasks.

**3. Clustering**

* **Introduction to Clustering**:
  + Clustering involves grouping similar data points together based on similarity measures (distance, density, etc.), with no prior labels.
* **K-means Clustering**:
  + A partition-based algorithm that divides data into kkk clusters. It iteratively assigns each data point to the nearest centroid and updates centroids until convergence.
  + Simple, but sensitive to the initial placement of centroids.
* **K-Mode Clustering**:
  + A variation of K-means, used for categorical data. Instead of means, it uses the mode (most frequent value) to represent each cluster.
* **Hierarchical Clustering**:
  + A bottom-up (agglomerative) or top-down (divisive) method of clustering. The algorithm creates a tree-like structure (dendrogram) to represent the nested clusters.
  + No need to predefine the number of clusters.

**4. Anomaly Detection**

* Identifying data points that deviate significantly from the rest of the dataset. This is useful in fraud detection, network security, and monitoring systems.

**5. Association Algorithms**

* **Apriori Algorithm**:
  + A classic algorithm used for mining frequent itemsets in transactional databases. It iteratively finds subsets of items that frequently appear together and generates association rules.
* **F-P Growth Algorithm**:
  + An efficient algorithm for frequent itemset mining that avoids candidate generation, which is a limitation of Apriori.
* **Eclat Algorithm**:
  + A fast algorithm for frequent itemset mining that uses a vertical data format, which helps reduce the time complexity compared to Apriori.

**6. Ensemble Methods**

* **Bootstrap Aggregation (Bagging)**:
  + Combines the results of multiple models (usually decision trees) trained on random subsets of the data. Helps reduce overfitting and variance.
* **Boosting**:
  + A sequential ensemble technique that focuses on correcting the errors made by the previous model. Each new model is trained to reduce the residual error of the prior model.
* **Gradient Boosting Machines (GBM)**:
  + A type of boosting that minimizes the loss function by using gradient descent. It is highly effective for regression and classification tasks.
* **Stacking**:
  + An ensemble method where predictions from multiple models are used as input for a meta-model (usually a logistic regression or another classifier), which makes the final prediction.

**UNIT IV: Metrics, Model Evaluation, and Reinforcement Learning (9 hours)**

**1. Evaluation Metrics**

* **Bias-Variance Trade-off**:
  + The bias-variance trade-off refers to the balance between a model's ability to generalize (low bias) and its ability to adapt to the training data (low variance).
  + **Overfitting**: When a model learns too much from the training data, including noise, leading to poor performance on new data.
  + **Underfitting**: When a model is too simple and fails to capture the underlying data patterns, leading to poor performance on both training and test data.

**2. Classification Metrics**

* **Precision**:
  + The ratio of correctly predicted positive observations to the total predicted positives:

Precision=TPTP+FP\text{Precision} = \frac{TP}{TP + FP}Precision=TP+FPTP​

* **Recall**:
  + The ratio of correctly predicted positive observations to all actual positives:

Recall=TPTP+FN\text{Recall} = \frac{TP}{TP + FN}Recall=TP+FNTP​

* **F1-score**:
  + The harmonic mean of precision and recall:

F1=2×Precision×RecallPrecision+RecallF1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}F1=2×Precision+RecallPrecision×Recall​

* **ROC Curve**:
  + A graph of the true positive rate (sensitivity) vs. false positive rate (1 - specificity) for different classification thresholds.
* **AUC (Area Under Curve)**:
  + Measures the ability of the model to discriminate between classes. A higher AUC indicates a better model.
* **Confusion Matrix**:
  + A table that summarizes the performance of a classification model by showing the true positives, false positives, true negatives, and false negatives.

**3. Regression Metrics**

* **Mean Absolute Error (MAE)**:
  + The average of the absolute differences between predicted and actual values:

MAE=1n∑i=1n∣yi−y^i∣MAE = \frac{1}{n} \sum\_{i=1}^{n} |y\_i - \hat{y}\_i|MAE=n1​i=1∑n​∣yi​−y^​i​∣

* **Mean Squared Error (MSE)**:
  + The average of the squared differences between predicted and actual values:

MSE=1n∑i=1n(yi−y^i)2MSE = \frac{1}{n} \sum\_{i=1}^{n} (y\_i - \hat{y}\_i)^2MSE=n1​i=1∑n​(yi​−y^​i​)2

* **R-squared**:
  + The proportion of the variance in the dependent variable that is predictable from the independent variables. R-squared values closer to 1 indicate a better fit.

**4. Cross-Validation Techniques**

* **K-Fold Cross-Validation**:
  + The data is divided into k subsets. The model is trained k times, each time using a different fold as the test set and the remaining data for training.
* **Stratified Cross-Validation**:
  + Ensures that each fold has the same proportion of classes, which is important for imbalanced datasets.
* **Leave-One-Out Cross-Validation (LOOCV)**:
  + A special case of k-fold where k is equal to the number of data points. It trains the model on all but one data point and tests it on the left-out point.

**5. Reinforcement Learning**

* **Introduction**:
  + Reinforcement learning involves an agent learning to make decisions by interacting with an environment. The agent receives feedback in the form of rewards or penalties, aiming to maximize cumulative reward over time.