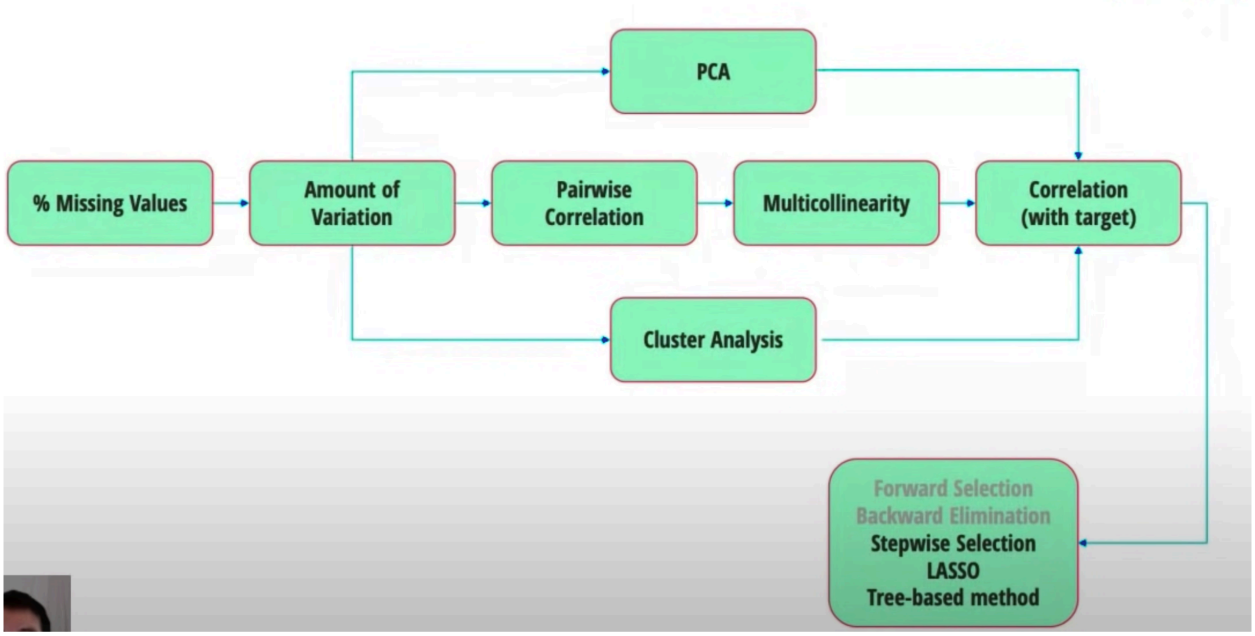
Feature Selection



Machine learning Models

Open Book Exam Templates for MGT301

Section 1: Conceptual Questions

1. **General Understanding**:
   - Explain the difference between `DataFrame` and `Series` in pandas. Provide an example of when you would use each.
   - Discuss three methods to handle outliers in a dataset. Provide an example for each.
2. **Data Cleaning**:
   - What is the importance of handling missing values in data preprocessing? Illustrate with examples.
   - Describe the process of using the `z-score` method for identifying outliers. What are the advantages and disadvantages of this method?
3. **Descriptive Statistics**:
   - Define skewness and kurtosis. What do they indicate about a dataset?
   - Explain the difference between variance and covariance. Provide a scenario where each is relevant.
4. **Grouping and Aggregation**:
   - Explain the purpose of the `groupby` function in pandas. Provide an example of its usage.

- How would you calculate multiple aggregations (e.g., mean, median) for specific columns in a grouped dataset?

Section 2: Code Implementation Questions

1. **Basic Data Operations**:
    - Write code to create a pandas DataFrame from a dictionary. Include columns for 'Name', 'Age', and 'Department'.
    - Modify a given list by appending and removing elements, then converting it into a tuple.
2. **Data Analysis**:
    - Given a dataset, write code to calculate the mean, variance, and standard deviation of a numerical column.
    - Implement code to filter rows in a DataFrame where a numerical column exceeds a specified value.
3. **Outlier Handling**:
    - Use the `IQR` method to remove outliers from a numerical column in a DataFrame. Provide a snippet of code to illustrate this.
    - Write a function to replace values greater than the 99th percentile or less than the 1st percentile with the boundary values.
4. **Visualization**:
    - Create a bar plot to display the mean values of a categorical column grouped by another categorical column.
    - Generate a histogram for a numerical column and discuss its skewness.

Section 3: Problem Solving Questions

1. **Application of Concepts**:
    - Suppose you are given a dataset of insurance claims. Write code to:
        - Group data by 'Region' and calculate the total claims per region.
        - Identify regions with total claims exceeding 50,000.
    - You have sales data for multiple stores. Write code to:
        - Calculate the monthly average sales for each store.
        - Identify the month with the highest sales for each store.
2. **Advanced Data Analysis**:
    - Write code to create a pivot table from a dataset containing 'Date', 'Product', and 'Sales'. The pivot table should show total sales for each product by month.
    - Implement a solution to identify top 3 most frequent categories in a categorical column.
3. **Scenario-Based Analysis**:
    - Given a dataset with 'Age', 'Gender', and 'Purchase Amount', analyze the relationship between age and purchase amount. Write code to:
        - Bin ages into intervals (e.g., 18-25, 26-35, etc.).
        - Calculate the average purchase amount for each age group.

        - Visualize the results.
    - A movie dataset contains 'Genres' and 'Revenue'. Write code to:
        - Split the 'Genres' column into individual genres.
        - Calculate the average revenue for each genre.
        - Identify the top 5 genres by average revenue.

. Supervised Learning

Supervised learning models require labeled data and are used for prediction tasks like classification and regression.

1.1. Classification Models

Used when the output variable is categorical.

- **Logistic Regression**: Binary or multi-class classification. Used for problems like spam detection or customer churn.
- **Support Vector Machines (SVM)**: Classification tasks with clear margins between classes, e.g., image classification.
- **k-Nearest Neighbors (k-NN)**: Non-parametric method for classification based on similarity to neighbors. Suitable for smaller datasets.
- **Decision Trees**: Simple, interpretable models for classification.
- **Random Forest**: Ensemble of decision trees. Used for both classification and regression, reducing overfitting compared to single trees.
- **Gradient Boosting (e.g., XGBoost, LightGBM)**: Boosted trees for highly accurate models in competitions or structured data.
- **Naive Bayes**: Text classification, spam detection, and sentiment analysis.
- **Linear Discriminant Analysis (LDA)**: Classification with normally distributed classes.
- **Quadratic Discriminant Analysis (QDA)**: Like LDA but allows different covariance for each class.

1.2. Regression Models

Used when the output variable is continuous.

- **Linear Regression**: Predicting continuous values, e.g., house prices.
- **Ridge/Lasso Regression**: Linear regression with regularization to prevent overfitting.
- **Polynomial Regression**: Extends linear regression by adding polynomial terms.
- **Decision Tree Regressor**: Predicting continuous outputs with a tree structure.
- **Random Forest Regressor**: Ensemble model for robust regression tasks.

- **Support Vector Regressor (SVR)**: Regression tasks with non-linear relationships.
- **ElasticNet**: Combines Ridge and Lasso for robust feature selection.
- **Gradient Boosting Regressor (e.g., XGBoost, LightGBM)**: High-performing regression tasks.
- **k-Nearest Neighbors Regressor**: Predicting based on nearby points.

---

2. Unsupervised Learning

No labeled output, used for clustering, dimensionality reduction, and density estimation.

2.1. Clustering

Group similar data points.

- **k-Means**: Partition data into kkk clusters. Used in customer segmentation.
- **DBSCAN**: Density-based clustering, useful for irregular clusters.
- **Agglomerative Clustering**: Hierarchical clustering for structured datasets.
- **Gaussian Mixture Models (GMM)**: Probabilistic clustering using Gaussian distributions.

2.2. Dimensionality Reduction

Reduce the number of features in a dataset.

- **Principal Component Analysis (PCA)**: Projects data into fewer dimensions. Used for visualization or noise reduction.
- **t-SNE**: Visualizing high-dimensional data in 2D or 3D.
- **Linear Discriminant Analysis (LDA)**: Both a classifier and dimensionality reduction technique.
- **Truncated SVD**: Similar to PCA but used for sparse matrices.

---

3. Semi-Supervised Learning

Handles datasets with both labeled and unlabeled data.

- **Label Propagation**: Propagates labels in a graph-based approach.
- **Self-training Classifier**: Iteratively adds pseudo-labeled data.

---

4. Ensemble Models

Combine multiple models to improve predictions.

- **Bagging (e.g., BaggingClassifier, BaggingRegressor)**: Reduces variance by training on random subsets.
- **Boosting (e.g., AdaBoost, GradientBoosting)**: Focuses on misclassified points in subsequent models.
- **Voting Classifier**: Combines predictions from multiple models for improved classification.
- **Stacking**: Combines multiple models by using their predictions as input for another model.

---

5. Neural Networks

Used for non-linear, high-dimensional data.

- **MLPClassifier**: Multi-layer perceptron for classification tasks.
- **MLPRegressor**: Multi-layer perceptron for regression tasks.

---

6. Preprocessing and Pipelines

Used to prepare data for machine learning models.

- **StandardScaler**: Standardizes features to zero mean and unit variance.
- **MinMaxScaler**: Scales features to a fixed range, e.g., [0, 1].
- **OneHotEncoder**: Encodes categorical variables.
- **Imputer**: Handles missing values.
- **Pipeline**: Combines multiple steps, e.g., scaling and model training.

When to Use Each Model

1. **Classification Tasks**:
    - **Text data**: Naive Bayes, SVM.
    - **Image data**: SVM, Random Forest, Neural Networks.
    - **Small datasets**: k-NN, Decision Trees.
2. **Regression Tasks**:
    - **Linear relationships**: Linear Regression, Ridge, Lasso.
    - **Non-linear relationships**: Random Forest, SVR, Gradient Boosting.
3. **Clustering**:
    - **Structured data**: k-Means.

    - **Irregular clusters**: DBSCAN.
4. **Dimensionality Reduction**:
    - **Data visualization**: PCA, t-SNE.
5. **Time Series**:
    - Gradient Boosting, Random Forest, or SVR (after creating time-lag features).

Let me know if you'd like more details or examples for specific models!