# Exploratory Data Analysis (EDA)

- **Definition:** Systematic approach to understand dataset characteristics
- **Purpose:** Discover patterns, spot anomalies, test hypotheses, and check assumptions
- **Modeva's EDA capabilities:** A suite of visualization tools
- **Central component:** The `DataSet` class

# The DataSet Class in Modeva

The `DataSet` class provides five main EDA functions:

- `DataSet.eda_1d` - Generate univariate plots
- `DataSet.eda_2d` - Generate bivariate plots
- `DataSet.eda_3d` - Generate 3D scatter plots
- `DataSet.eda_correlation` - Generate correlation heatmaps
- `DataSet.eda_pca` - Generate PCA plots

# Univariate (1D) Plots

`DataSet.eda_1d` visualizes the distribution of individual features:

- **For categorical features:**
  - Bar charts showing frequency or count
- **For numerical features:**
  - Density plots - smooth representation of distribution
  - Histograms - binned representation of distribution

# Implementing Univariate Analysis

## Basic Usage

```
# Import necessary libraries
from modeva import DataSet
# Load your dataset
dataset = DataSet(data)
# Generate univariate plots
dataset.eda_1d()
```

## Customization Options

```
# Specify plot type for numerical features
dataset.eda_1d(plot_type='histogram')  # or 'density'
# Filter specific features to plot
dataset.eda_1d(features=['feature1', 'feature2'])
```

# Bivariate (2D) Plots

`DataSet.eda_2d` visualizes relationships between pairs of features:

- **Two numerical features:**
  - 2D scatter plots
- **Two categorical features:**
  - Stacked bar plots
- **One numerical and one categorical feature:**
  - Side-by-side box plots

# Implementing Bivariate Analysis

## Basic Usage

```
# Generate bivariate plots for all feature pairs
dataset.eda_2d()
```

## Customization Options

```
# Specify feature pairs to plot
dataset.eda_2d(feature_pairs=[('feature1', 'feature2'),
                              ('feature3', 'feature4')])

# Add color annotation by target variable
dataset.eda_2d(color_by='target_variable')
```

# 3D Scatter Plots

`DataSet.eda_3d` creates interactive 3D visualizations:

- Visualizes relationships between three numerical features
- Optional fourth feature represented by color annotation
- Interactive plot allowing rotation and zooming
- Helpful for discovering clusters and nonlinear relationships

# Implementing 3D Scatter Plots

## Basic Usage

```
# Generate 3D scatter plot for three features
dataset.eda_3d(features=['feature1', 'feature2', 'feature3'])
```

## Adding Color Dimension

```
# Add color based on a fourth feature or target
dataset.eda_3d(features=['feature1', 'feature2', 'feature3'],
               color_by='target_variable')
```

# Correlation Heatmap

`DataSet.eda_correlation` visualizes pairwise relationships:

- Displays strength and direction of relationships between features
- Color intensity represents correlation magnitude
- Supports multiple correlation methods for different relationship types
- Useful for feature selection and multicollinearity detection

# Correlation Methods

`DataSet.eda_correlation` supports four correlation methods:

- **pearson:** Measures linear relationships between continuous variables
  - Range: -1 (perfect negative) to 1 (perfect positive)
  - Sensitive to outliers
- **spearman:** Assesses monotonic relationships based on ranks
  - Range: -1 to 1
  - Robust to outliers, captures non-linear patterns
- **kendall:** Measures association between ranked variables
  - Range: -1 to 1
  - Useful for ordinal data, robust to outliers
- **xicor:** Detects both linear and nonlinear dependencies
  - Range: typically 0 (no dependence) to 1 (strong dependence)
  - More comprehensive view of relationships

# Implementing Correlation Analysis

## Basic Usage

```
# Generate correlation heatmap with default method (pearson)
dataset.eda_correlation()
```

## Specifying Correlation Method

```
# Use Spearman correlation
dataset.eda_correlation(method='spearman')

# Use XiCor for detecting nonlinear relationships
dataset.eda_correlation(method='xicor')
```

# PCA Plot

`DataSet.eda_pca` performs dimensionality reduction:

- Reduces high-dimensional data to principal components
- Visualizes variance explained by each component
- Shows feature loadings (contributions to components)
- Helps identify important features and data structure

**Example:**

# Implementing PCA Analysis

## Basic Usage

```
# Generate PCA plot
dataset.eda_pca()
```

## Customization Options

```
# Specify number of components
dataset.eda_pca(n_components=3)

# Color points by target variable
dataset.eda_pca(color_by='target_variable')

# Select specific features for PCA
dataset.eda_pca(features=['feature1', 'feature2', 'feature3'])
```

# EDA Workflow with Modeva

1. **Start with univariate analysis:**
   - Understand individual feature distributions
   - Identify outliers and data quality issues
2. **Proceed to bivariate analysis:**
   - Explore relationships between feature pairs
   - Identify potential predictive features
3. **Use correlation analysis:**
   - Understand feature interdependencies
   - Detect multicollinearity
4. **Apply 3D visualization and PCA:**
   - Explore higher-dimensional relationships
   - Reduce dimensionality while preserving information

# Complete EDA Example

```
# Import necessary libraries
from modeva import DataSet
# Load your dataset
dataset = DataSet(data)

# Perform comprehensive EDA
dataset.eda_1d()  # Univariate analysis
dataset.eda_2d()  # Bivariate analysis
dataset.eda_correlation(method='pearson')  # Linear correlations
dataset.eda_correlation(method='xicor')  # Nonlinear dependencies
dataset.eda_3d(features=['feature1', 'feature2', 'feature3'],
               color_by='target')  # 3D visualization
dataset.eda_pca(color_by='target')  # Dimensionality reduction
```

# Summary

- **Modeva's DataSet class** provides comprehensive EDA capabilities
- **Five main visualization functions:**
  - `eda_1d`: Univariate distributions
  - `eda_2d`: Bivariate relationships
  - `eda_3d`: 3D visualization
  - `eda_correlation`: Correlation analysis
  - `eda_pca`: Dimensionality reduction
- **Multiple correlation methods** for different relationship types
- **Integrated workflow** from basic to advanced analysis