**MOOC 1**

**Exploratory Data Analysis for Machine Learning**

1. **A Brief History of Modern Al and its Applications:**

**1. Machine Learning and Deep Learning:**

\* Understanding Machine Learning

* Machine Learning algorithms learn patterns from data over time, improving with more data, although performance may plateau after a certain point.

\* Types of Machine Learning

* Supervised Learning: Involves labeled datasets to predict outcomes (e.g., fraud detection).
* Unsupervised Learning: Works with unlabeled data to find underlying structures (e.g., customer segmentation).

\* Deep Learning vs. Traditional Machine Learning

* Deep Learning uses complex models (neural networks) to automatically extract features from data, particularly effective for tasks like image classification.
* Traditional Machine Learning requires manual feature selection, which can be challenging, especially with high-dimensional data like images.

\*Understanding Features in Images

* Traditional Machine Learning struggles with defining features in images, such as distinguishing between a cat and a dog.
* Each pixel in an image can be considered a feature, leading to a vast number of features (e.g., 65,000 for a 256x256 image), complicating the analysis.

\* Role of Deep Learning

* Deep Learning utilizes deep neural networks to automatically learn and combine features, capturing spatial relationships between pixels.
* This approach allows for more effective image classification compared to traditional methods, which require manual feature selection.

\*Comparison with Traditional Machine Learning

* In traditional Machine Learning, humans must define features before model training, which can be challenging.
* Deep Learning simplifies this by allowing the model to learn meaningful features directly from the data, enhancing performance, especially with larger datasets.

**2. History of AI:**

\* Early Developments in AI

* The term "artificial intelligence" was coined at the Dartmouth Conference in 1956, marking the official recognition of AI as a field.
* Key milestones include Alan Turing's development of the Turing test in 1950 and Frank Rosenblatt's invention of the perceptron algorithm in 1957, which laid the groundwork for neural networks.

\* AI Winters and Resurgences

* The first AI Winter occurred in the 1960s and 1970s due to unmet expectations and limited progress, leading to reduced funding.
* The 1980s saw a resurgence with the rise of expert systems and the introduction of the Backpropagation algorithm, which allowed for more complex neural networks.

\* Modern AI Breakthroughs

* The late 1990s and early 2000s marked a successful period for machine learning, particularly in applications like speech recognition and search algorithms.
* Today, deep learning has emerged as a powerful tool, overcoming previous limitations and excelling in complex tasks such as image classification and machine translation.

**3. History of Machine Learning and Deep Learning:**

\* Historical Milestones in AI

* In the 1990s and 2000s, classical machine learning techniques gained traction, leading to successes in areas like speech recognition and robotics.
* Notable achievements include Deep Blue defeating a world chess champion in 1996 and Google's PageRank algorithm revolutionizing search engines.

\* Advancements in Deep Learning

* In 2006, breakthroughs in deep learning addressed issues like exploding and vanishing gradients, allowing for deeper neural networks.
* The introduction of the ImageNet database in 2009 provided a vast resource for training algorithms, culminating in the 2012 AlexNet competition that showcased significant improvements in visual recognition.

\* Recent Developments in AI

* From 2013 to 2019, AI made strides in natural language processing, machine translation, and computer vision, with notable projects like DeepMind's AlphaGo and IBM's Project Debater demonstrating advanced capabilities.
* These advancements have led to practical applications in various fields, marking a significant era in AI development.

**4. Modern AI:**

\* Current AI Advancements

* Computer vision is enabling self-driving cars and improving medical imaging diagnostics, often matching or surpassing human experts.
* Natural language processing has seen improvements in translation, sentiment analysis, and content generation.

\* Factors Driving AI Growth

* Larger datasets are now available, allowing for better model training and complex pattern recognition.
* Enhanced cloud infrastructure and powerful hardware have made data processing more efficient and accessible.
* Innovations in deep learning have led to practical applications across various industries.

\* AI Applications Across Industries

* In healthcare, AI aids in medical imaging and drug discovery, enhancing patient care.
* Industrial sectors benefit from automation, predictive maintenance, and optimized agricultural production.
* Finance utilizes AI for algorithmic trading, fraud detection, and risk management.
* Government applications include threat identification, smart city initiatives, and improved citizen engagement.
* Transportation innovations include autonomous vehicles and optimized logistics.
* AI is also transforming personalized advertising, education, gaming, and service industries.

**5. Applications:**

\* Applications of AI in Transportation

* AI enhances navigation by considering traffic data and weather conditions, improving route efficiency (e.g., Google and Waze).
* Ride-sharing platforms like Uber and Lyft utilize AI to adjust pricing based on real-time supply and demand.

\* AI in Social Media

* AI algorithms identify relevant content, suggest connections, and deliver targeted advertisements.

\* Voice Recognition and Object Detection

* Products like Siri and Alexa use natural language processing to understand and respond to user commands.

**6. Machine Learning Workflow:**

\* Basic Background and Tools

* Familiarity with Python libraries such as NumPy, Pandas, Matplotlib, Seaborn, Scikit-Learn, TensorFlow, and Keras is assumed.

\* Typical Machine Learning Workflow

* The workflow begins with defining the problem statement, followed by data collection and exploration.
* Data preprocessing is crucial for cleaning data, followed by modeling, validation, and deployment of the solution.

\* Key Machine Learning Vocabulary

* Target variable: The value being predicted (e.g., species of iris flowers).
* Features: The explanatory variables used to predict the target variable (e.g., sepal and petal dimensions).
* Observations: Individual rows in the dataset, each representing a single example.
* Labels: The specific values of the target variable associated with observations.

1. **Retrieving and Cleaning Data:**

**1. Retrieving data from csv and json files:**

Retrieving Data from CSV Files

* CSV (Comma Separated Values) files consist of rows of data separated by commas, easily read using Pandas with pd.read\_csv().
* Useful arguments for read\_csv() include specifying delimiters, handling headers, and defining null values.

Working with JSON Files

* JSON (JavaScript Object Notation) files are structured like Python dictionaries and are commonly used for NoSQL databases and APIs.
* To read JSON files, use pd.read\_json(), and for writing, use data.to\_json(), with various arguments available for different JSON structures.

**2. Retrieving Data from Databases, APIs, and the Cloud:**

* SQL Databases: Relational, fixed schema (e.g., MySQL, SQL Server, Oracle). Python tools: sqlite3, SQLAlchemy.
* NoSQL Databases: Non-relational, flexible formats (e.g., JSON). Types include document (MongoDB), graph, and wide column databases.
* APIs & Cloud Data: Provide access to external/cloud data sources. Data can be loaded into Python (e.g., Pandas with pd.read\_csv from a URL).

**3. Data Cleaning:**

* Importance of Data Cleaning: Clean data ensures accurate machine learning; bad data = bad results.
* Messy Data Issues: Includes duplicates, inconsistent text, missing values, outliers, and challenges like scattered or low-quality data.
* Handling Duplicates: Not all duplicates are errors; evaluate necessity, filter carefully, and always keep the original dataset.

**4. Handling Missing Values and Outliers:**

Handling Missing Values

* Removing Data: You can remove entire rows to clean the dataset quickly, but this may lead to loss of important information if many rows are missing values.
* Imputation: Replacing null values with the mean, median, or estimates helps retain data but introduces uncertainty into the model.

Dealing with Outliers

* Definition: Outliers are distinct observations that can skew predictions, such as an unusually high sales figure.
* Identification: Use plots like histograms, density plots, and box plots to detect outliers, and consider their potential informative value before removal.

**5. Handling Missing Values and Outliers using Residuals:**

* Residuals: Differences between actual and predicted values; standardized residuals make comparisons meaningful.
* Handling Outliers: Techniques include deleted residuals and studentized residuals to gauge observation impact.
* Managing Outliers: Strategies include removal, replacement, transformation, regression-based prediction, or robust modeling—each with trade-offs.
* Key Point: Outlier management is essential for reliable data analysis and model accuracy.

1. **Exploratory Data Analysis and Feature Engineering:**

**1. Introduction to Exploratory Data Analysis (EDA):**

* EDA Purpose: First step in data analysis to explore data, spot patterns/trends, and assess cleaning or data needs.
* Techniques: Use summary stats (mean, median, min, max, correlations) and visualizations (histograms, scatter plots, box plots).
* Sampling: Random sampling reduces large data size for efficiency; stratified sampling preserves class proportions for accurate analysis.

**2. EDA with Visualization:**

* Matplotlib: Core Python plotting library; very flexible. Use %matplotlib inline in Jupyter for display.
* Pandas & Seaborn: Pandas simplifies plotting (less flexible), Seaborn (built on Matplotlib) makes statistical and attractive plots easier.
* Visualizations: Scatter plots, histograms, bar plots; can layer plots and customize with labels/colors.
* EDA Role: Visualizations + summary statistics help reveal data distributions, patterns, and relationships.

**3. Grouping Data for EDA:**

* Pandas Plotting: Group by category (e.g., species), calculate means, and visualize with dot plots (custom colors per feature).
* Seaborn Tools: Use pair plots for feature relationships (colored by category) and hex bin plots for density, with histograms for distributions.
* Facet Grid: Break data into subplots by category (e.g., species-specific histograms) for detailed comparison.
* Key Point: EDA combines summary statistics and visualizations to uncover patterns and insights.

**4. Feature Engineering and Variable Transformation - Background:**

* Feature Engineering: Improves model performance through encoding, scaling, and transformations.
* Variable Transformation: Adjusts distributions (e.g., log transform) to handle outliers and skewness.
* Linear Relationships: Many models (e.g., linear regression) assume predictors relate linearly to the target.
* Model Example: Features like budgets can predict outcomes (e.g., box office) using coefficients.
* Key Point: Even transformed features can preserve linear relationships, enabling flexible yet valid modeling.

**5. Variable Transformation:**

* Data Transformations: Fix skewed raw data to improve linear regression.
* Log/Box-Cox: Normalize positively skewed data, making relationships more linear.
* Polynomial Features: Add higher-order terms to capture non-linear patterns while staying within a linear regression framework.
* Key Point: Transformations and polynomial features expand linear regression’s ability to handle skewness and complexity.

**6. Feature Encoding:**

* Feature Selection & Transformation: Picking the right features is vital; transformations (log, polynomial, scaling, encoding) prepare data for modeling.
* Categorical Encoding: Converts non-numeric data into numeric form.
* Nominal: No natural order (e.g., colors).
* Ordinal: Ordered categories (e.g., rankings).
* Encoding Methods:
* Binary: Two values → 0/1.
* One-Hot: New column per category (for nominal data).
* Ordinal Encoding: Assigns integers to ordered data (must consider distance meaning).

**7. Feature Scaling:**

* Feature scaling adjusts the scale of variables to allow for meaningful comparisons, especially when dealing with continuous features that have different ranges.
* An example illustrates how age measured in seconds versus the number of surgeries can lead to misleading groupings if not scaled properly.

**8. Common Variable Transformations in Python:**

Feature Transformations:

* Numerical: Scale with StandardScaler, MinMaxScaler, or RobustScaler (sklearn.preprocessing).
* Categorical: Encode with LabelEncoder, OneHotEncoder, or pd.get\_dummies.

Feature Encoding:

* Ordinal data: Use OrdinalEncoder or DictVectorizer to preserve order.
* Converts categories → numbers for model use.

Feature Scaling:

* Ensures features are on the same scale, critical for many ML algorithms.
* Techniques: Standardization (mean=0, std=1) and Normalization (values in [0,1]).

1. **Inferential Statistics and Hypothesis Testing:**

**1. Estimation and Inference - Introduction:**

Estimation vs. Inference:

* *Estimation*: Calculates parameter values (e.g., mean) from a sample.
* *Inference*: Draws conclusions about the population, considering uncertainty (e.g., standard error).

Parametric vs. Non-Parametric:

* *Parametric*: Assumes a specific data distribution.
* *Non-Parametric*: Makes no distributional assumptions.
* Choice depends on data and goals.

Frequentist vs. Bayesian:

* *Frequentist*: Based on long-run event frequencies.
* *Bayesian*: Combines prior beliefs with observed evidence.

**2. Estimation and Inference - Example:**

Data Overview: Contains customer traits, churn outcomes (e.g., cancellations, non-renewals), account types, revenue, satisfaction, and lifetime value.

EDA Findings:

* Payment type: Credit card users churn less.
* Tenure: Newer customers churn more.

Visualizations:

* Pair plots show feature relationships with churn colored.
* Hexbin plots reveal tenure–charges patterns linked to retention.

**3. Estimation and Inference - Parametric vs. Non-Parametric:**

Parametric Models: Rely on fixed parameters and distribution assumptions (e.g., normal distribution with mean & standard deviation).

Non-Parametric Models: More flexible, no strict distribution assumptions (e.g., histograms).

Common Distributions:

* Uniform: Equal probability for all values in a range.
* Normal: Bell-shaped, centered on mean.
* Log-normal: Variable’s log is normal (e.g., incomes).
* Exponential: Time until next event, skewed toward lower values.
* Poisson: Counts events in fixed intervals, based on rate (λ).

**4. Estimation and Inference - Commonly Used Distributions:**

* Uniform Distribution: All outcomes equally likely (e.g., fair die roll).
* Normal (Gaussian) Distribution: Bell curve around mean; defined by mean & std. Central Limit Theorem supports its ubiquity.
* Log-Normal Distribution: If log(variable) is normal → variable is log-normal. Common in finance (e.g., incomes, stock prices).
* Exponential Distribution: Skewed left; models time until next event (shorter intervals more frequent).
* Poisson Distribution: Models count of events in fixed intervals, based on rate (λ) (e.g., number of arrivals/views).

**5. Frequentist vs. Bayesian Statistics:**

Frequentist Statistics

* Relies on repeated observations to estimate probabilities without prior knowledge of true frequencies.
* Uses large sample sizes to derive estimates directly from data, providing confidence levels based on how well the sample represents the population.

Bayesian Statistics

* Allows parameters to have their own probability distributions, incorporating prior beliefs into the analysis.
* Updates prior distributions with new data to form posterior distributions, refining estimates as more information becomes available.

**6. Introduction to Hypothesis:**

Hypothesis Testing Basics:

* A hypothesis is a statement about a population parameter (e.g., mean).
* Two types: Null (H₀) = specific value; Alternative (H₁) = different/less specific value.

Procedure:

* Use sample data to test H₀.
* If evidence is strong, reject H₀ in favor of H₁.
* Important: rejecting H₀ ≠ fully accepting H₁, only that H₁ is more plausible given the test.

Bayesian Approach:

* Instead of a fixed decision, compute posterior probabilities of both H₀ and H₁.
* Chooses the more likely hypothesis given prior beliefs + observed data.

**7. Hypothesis Testing Example:**

Setup:

* Coin 1 → 70% heads.
* Coin 2 → 50% heads.
* Toss one coin 10 times → decide which coin based on observed heads.

Probability Analysis:

* Table shows probabilities of each head count under both coins.
* Few heads → more likely Coin 2.
* Many heads → more likely Coin 1.

Likelihood Ratio:

* Compares probabilities under Coin 1 vs. Coin 2.
* Example: 3 heads → outcome is 13× more likely under Coin 1 than Coin 2.
* Demonstrates likelihood ratios in hypothesis testing.

**8. Bayesian Interpretation of Hypothesis Testing Example:**

Priors:

* Assign initial probabilities to hypotheses (e.g., fair coin = 0.5, biased coin = 0.7).
* Without data, assume equal prior weights (50/50).

Likelihood & Posterior:

* Posterior = Prior × Likelihood (normalized).
* Likelihood ratio updates the priors based on observed data.

Key Point:

* Bayesian hypothesis testing combines priors (beliefs before data) with likelihoods (data evidence) to form posteriors (updated beliefs).
* Coin toss example shows how evidence shifts probabilities toward one hypothesis.

**9. Type 1 vs Type 2 Error:**

Type I and Type II Errors

* A type I error occurs when the null hypothesis is incorrectly rejected, suggesting a false positive.
* A type II error happens when the null hypothesis is incorrectly accepted, indicating a false negative.

**10. Hypothesis Testing Terminology:**

**T**est Statistic and Hypothesis Testing

* The test statistic is calculated from sample data to determine whether to accept or reject the null hypothesis.
* The rejection region consists of values for the test statistic that lead to rejecting the null hypothesis, while the acceptance region includes values that support accepting it.

**11. Significance Level and P-Values:**

Significance Level (α):

* Set *before* testing to prevent P-hacking.
* Lower α → fewer Type I errors (false positives), important in high-stakes cases (e.g., medicine).

P-values:

* Probability of seeing data (or more extreme) if H₀ is true.
* Thresholds: 0.1, 0.05, 0.01 (smaller = stricter rejection rule).

Examples:

* *Marketing*: At α=0.05, campaign results must show strong evidence vs. “no effect” to reject H₀.
* *Coin toss*: P-value calculation decides whether observed outcomes reject H₀.

**12. Significance Level and P-Values and the F Statistic:**

F-Statistic and Null Hypothesis

* The null hypothesis states that all regression coefficients (betas) are zero, indicating that the features do not improve the model compared to using the mean of the outcome variable.
* A small p-value for the F-Statistic suggests rejecting the null hypothesis, indicating that at least one coefficient has a significant effect on the outcome.

Type I Error and Sample Size

* Type I error occurs when the null hypothesis is incorrectly rejected; running multiple tests increases the likelihood of this error.
* The probability of at least one Type I error can be approximated as 1 - (1 - 0.05)^(# tests), leading to a higher chance of error with more tests.

Bonferroni Correction

* To control Type I error rates, the Bonferroni Correction adjusts the p-value threshold based on the number of tests, typically using a threshold of 0.05 / (# tests).
* This correction reduces the likelihood of Type I errors but may decrease the power to detect true effects, requiring larger sample sizes or stronger effects to maintain significance.

**13. Correlation vs Causation:**

* Correlation indicates a relationship between two variables but does not imply that one causes the other.
* Understanding the underlying mechanisms is essential to avoid misinterpretation of data.