MID-TERM REPORT

Data science

Summer of science(sos)2023

Lomisa lakshman kumar

210110073

Mentored by – Sanket

What is data science?

Data science is the practice of transforming data into actionable insights to gain a competitive advantage in diverse industries.It involves extracting knowledge and insights from large volumes of data using various techniques such as statistical analysis, machine learning, and data visualization. It combines elements of mathematics, computer science, and domain expertise to uncover patterns, make predictions, and solve complex problems.

Phython

Python is one of the most popular programming languages used in data science due to its extensive range of libraries and tools specifically designed for data analysis, manipulation, and machine learning.

Here are the subtopics in phython :

1. Syntax and Basic Concepts:
    - Variables, data types, and operators
    - Control flow statements (if, for, while)
    - Functions and modules
2. Data Structures:
    - Lists, tuples, and dictionaries
    - Sets and frozensets
    - Strings and string manipulation
3. File Handling:
    - Reading from and writing to files
    - File modes and handling exceptions
    - Working with CSV and JSON files
4. Object-Oriented Programming (OOP):
    - Classes, objects, and instances
    - Inheritance and polymorphism

- Encapsulation and abstraction

5. Error Handling and Exceptions:
    - Try-except blocks
    - Raising and handling exceptions
    - Finally block and resource cleanup
6. Regular Expressions:
    - Pattern matching and searching
    - Metacharacters and quantifiers
    - Substitution and capturing groups
7. Modules and Packages:
    - Importing modules and using functions
    - Creating and organizing packages
    - Understanding namespaces and scope
8. Functional Programming:
    - Lambda functions and higher-order functions
    - Map, filter, and reduce functions
    - List comprehensions
9. Python Standard Library:
    - Commonly used modules (e.g., datetime, math)
    - Working with dates, times, and calendars
    - File and directory manipulation (os, shutil)

Data science libraries

Data science libraries offer a range of functionalities and tools that enable data scientists to analyze and extract insights from large datasets. Here is a summary of subtopics, including syntax, related to data science libraries:

1. Syntax and Basic Concepts:

- Understand the syntax and conventions used in Python programming.
- Familiarize yourself with variables, data types, operators, and control flow statements.

2. NumPy:
    - Learn NumPy's syntax for creating and manipulating multidimensional arrays.
    - Explore mathematical functions, indexing, slicing, and broadcasting operations provided by NumPy.
3. Pandas:
    - Gain proficiency in Pandas' syntax for working with Series and DataFrames.
    - Understand data manipulation techniques such as filtering, sorting, merging, and grouping.
4. Matplotlib:
    - Explore Matplotlib's syntax for creating various types of visualizations, such as line plots, scatter plots, and bar charts.
    - Customize plots by adding labels, titles, legends, and annotations.
5. Seaborn:
    - Understand Seaborn's syntax for creating statistical visualizations, including distribution plots, regression plots, and categorical plots.
    - Utilize Seaborn's high-level functions for styling and enhancing visualizations.
6. Scikit-learn:
    - Learn Scikit-learn's syntax for implementing machine learning algorithms.
    - Explore syntax for data preprocessing, model selection, evaluation, and cross-validation.
7. TensorFlow:

- Understand TensorFlow's syntax for building and training deep learning models.
- Learn how to define computational graphs, work with tensors, and apply various layers and activations.

8. PyTorch:
    - Familiarize yourself with PyTorch's syntax for creating and training neural networks.
    - Explore tensor operations, autograd, and different optimization techniques.

Data Manipulation

Data manipulation is a critical component of data science, involving various techniques to transform, clean, and prepare data for analysis.. Here are key subtopics in data manipulation:

1. Data Cleaning:
    - Identify and handle missing values, outliers, and inconsistencies.
    - Impute or remove missing values using appropriate techniques.
    - Standardize formats, correct data types, and address inconsistencies.
2. Data Filtering and Selection:
    - Filter data based on specific conditions or criteria.
    - Select relevant columns or rows for analysis.
    - Use logical and comparison operators to filter data.
3. Data Transformation:
    - Create new variables or features from existing ones.
    - Apply mathematical or statistical transformations to variables.
    - Normalize, scale, or encode categorical variables.
4. Data Aggregation and Grouping:

- Group data based on variables or categories.
- Perform aggregation operations like sum, average, count, etc.
- Compute group-level statistics and insights.

5. Data Reshaping:
    - Transform data between wide and long formats.
    - Use techniques like pivoting, stacking, and melting.
    - Reshape data for specific analysis requirements.
6. Data Joining and Merging:
    - Combine multiple datasets based on common columns or keys.
    - Perform different types of joins to merge data.
    - Handle duplicates, overlapping values, and conflicting data.
7. Feature Engineering:
    - Create meaningful features from raw data.
    - Derive new variables using domain knowledge.
    - Engineer features for improved predictive performance.

Visualization using Matplotlib

Matplotlib is a powerful Python library for creating visualizations and plots. It offers a range of subtopics for effective data visualization:

1. Line Plots: Display trends and patterns using lines.
2. Scatter Plots: Explore relationships between variables.
3. Bar Plots: Compare categorical data using bars.
4. Histograms: Visualize data distribution.
5. Pie Charts: Show proportions or percentages.
6. Box Plots: Analyze data variability.
7. Heatmaps: Represent data using colors in a grid-like format.

8. Subplots and Figures: Create multiple plots within a figure.
9. Customization: Customize visual elements and styles.

Machine Learning

Mathematics: Mathematics forms the foundation of machine learning in data science. summary of important mathematical concepts and formulas required for understanding and implementing machine learning algorithms:

1. Linear Algebra:
    - Vectors: Representing data points.
    - Matrices: Representing datasets.
    - Matrix Multiplication: C = AB.
    - Transpose: A^T.
2. Calculus:
    - Derivative: Measures rate of change.
    - Gradient: ∇ denotes partial derivatives.
    - Chain Rule: Computes derivatives of composite functions.
    - Gradient Descent: $\theta \leftarrow \theta - \alpha * \nabla(\text{loss})$.
3. Probability and Statistics:
    - Probability: Quantifies uncertainty.
    - Probability Distributions.
    - Mean: $\mu = (x_1 + x_2 + ... + x_n) / n$.
    - Variance: $\sigma^2 = \Sigma((x_i - \mu)^2) / n$.
    - Covariance.
    - Normal Distribution (Gaussian).
    - Bayes' Theorem: $P(A|B) = (P(B|A) * P(A)) / P(B)$.
4. Optimization Techniques:
    - Regularization.

- L1 Regularization (Lasso).
- L2 Regularization (Ridge).

Fundamentals of ML

Machine learning is a powerful field in data science that involves the development of algorithms capable of learning patterns from data and making predictions or decisions. Here are key fundamental concepts:

1. Supervised Learning:
    - Supervised learning involves training a model on labeled data with known outcomes.
    - The model learns the relationship between input features (X) and target variables (y) to make predictions on new, unseen data.
    - Formula for the predicted target variable ($\hat{y}$) in linear regression: $\hat{y} = \beta_0 + \beta_1X_1 + \beta_2X_2 + ... + \beta_nX_n$, where $\beta_0, \beta_1, \beta_2, ..., \beta_n$ are the coefficients learned during training.
2. Unsupervised Learning:
    - Unsupervised learning deals with unlabeled data and aims to discover patterns or structures in the data.
    - Clustering algorithms group similar data points together, while dimensionality reduction techniques capture the most important information in a lower-dimensional space.
    - Formula for K-means clustering: Minimize the within-cluster sum of squared distances: $J = \Sigma_i\Sigma_j ||x_i - \mu_j||^2$, where x_i represents a data point, μ_j is the centroid of cluster j.
3. Linear Regression:
    - Linear regression is a supervised learning algorithm used for predicting continuous target variables.
    - It models the relationship between input features (X) and the target variable (y) using a linear equation.

- The goal is to minimize the sum of squared residuals (errors) between the predicted values ($\hat{y}$) and the actual values (y).
- Formula for linear regression: $y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n + \varepsilon$, where β_0, β_1, β_2, ..., β_n are the coefficients to be learned, and ε represents the error term.

4. Logistic Regression:
    - Logistic regression is a supervised learning algorithm used for binary classification tasks.
    - It estimates the probability of an input belonging to a certain class using a logistic function.
    - The logistic function maps the linear regression output (z) to a value between 0 and 1, representing the probability.
    - Formula for logistic regression: $p = 1 / (1 + e^{(-z)})$, where p is the probability, $z = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n$, and e is the base of the natural logarithm.
5. Introduction to Random Forests:
    - Random forests are an ensemble learning method that combines multiple decision trees.
    - Each decision tree is trained on a random subset of the data and features, reducing overfitting.
    - Random forests can handle both regression and classification tasks and are known for their robustness and accuracy.
    - No specific formula for random forests, but the underlying decision trees use various splitting criteria, such as Gini impurity or entropy, to make decisions.

Advanced ML:

overview of advanced ML techniques and an introduction to neural networks and deep learning:

1. Support Vector Machines (SVM):

- SVM separates data points by constructing a hyperplane with the largest margin between classes.
- The decision function for a new data point x is given by: f(x) = sign(w^T * x + b), where w represents the weights and b is the bias term.
- The objective function to maximize the margin is: min 0.5 * ||w||^2, subject to y_i(w^T * x_i + b) ≥ 1 for all training samples (x_i, y_i).

2. DBSCAN (Density-Based Spatial Clustering of Applications with Noise):
    - DBSCAN groups data points based on their density and defines clusters as areas of high density separated by areas of low density.
    - The density of a data point is determined by the number of neighboring points within a specified radius ε.
    - Core points have a sufficient number of neighboring points (at least minPts), and density-reachable points belong to the same cluster.
    - The algorithm assigns noise points that do not belong to any cluster.
3. Principal Component Analysis (PCA):
    - PCA reduces the dimensionality of high-dimensional data while preserving the maximum variance.
    - The principal components are obtained by performing an eigendecomposition of the covariance matrix or singular value decomposition (SVD) of the data.
    - The first principal component captures the direction of maximum variance, and subsequent components are orthogonal and capture decreasing variance.
    - The transformed data is obtained by projecting the original data onto the selected principal components.
4. Introduction to Neural Networks:

- Neural networks consist of interconnected layers of nodes (neurons) that process and transform data.
- The output of a neuron is computed by applying an activation function to the weighted sum of its inputs: output = activation(w^T * x + b), where w represents the weights, x is the input, and b is the bias term.
- The most commonly used activation functions include the sigmoid function, tanh function, and rectified linear unit (ReLU) function.

5. Deep Learning:
    - Convolutional Neural Networks (CNN) use convolutional layers with filters/kernels to capture local patterns in images: output = activation(convolution(input, kernel) + bias).
    - Recurrent Neural Networks (RNN) maintain a hidden state that captures temporal dependencies: hidden_state(t) = activation(weight_input * input(t) + weight_hidden * hidden_state(t-1) + bias).
    - Long Short-Term Memory (LSTM) is a type of RNN that uses gates to control information flow and effectively capture long-term dependencies.
    - Generative Adversarial Networks (GAN) consist of a generator network that generates new samples and a discriminator network that tries to distinguish between real and generated samples.

Remaining topics to cover:

Week5: Data science in action like NLP,sentiment and text analysis , introduction to recommendation systems

Week6: Big data and cloud computing

Week7:Development and model management

Week8: Work on a data science real world examples, applying all the skills and techniques learned throughout the course

THANKYOU