|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Register Number** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



**SRM Institute of Science and Technology**

Set -

**College of Engineering and Technology**

**School of Computing**

SRM Nagar, Kattankulathur – 603203, Chengalpattu District, Tamil Nadu

# Academic Year: 2024-25 (EVEN)

Test: FT4 Date: 29-04-2025

Course Code & Title: 21CSS303T-Data Science Duration: Two periods

Year& Sem: III Year /VI Sem Max.Marks:50

Course Articulation Matrix:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course  Outcome | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO3 | - | - | - | - | 1 | - | - | - | - | - | - | - |
| CO4 | - | - | - | - | 1 | - | - | - | - | - | - | - |
| CO5 | - | - | - | - | 1 | - | - | - | - | - | - | - |

**Note:** CO3 – To identify data manipulation and cleaning techniques using pandas

CO4 – To constructs the Graphs and plots to represent the data using python packages

CO5 – To apply the principles of the data science techniques to predict and forecast the outcome of real- world problem

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Part – A** (10 x 1 = 10 Marks)  *Instructions:*  1) Answer **ALL** questions.  2) The duration for answering Part A is **15 minutes** (this sheet will be collected after 15 minutes).  3**) Encircle the correct answer**. | | | | | | |
| S.No | Question | Marks | BL | CO | PO | PI Code |
| 1 | Which of the following tools is used for compactly storing large arrays and supports memory-mapping?  A) NumPy  B) Matplotlib  C) Bcolz  D) Seaborn | 1 | 1 | 3 | 5 |  |
| 2 | What does the method combine\_first () do in data wrangling?  A) Deletes duplicate data  B) Fills missing values with data from another DataFrame  C) Joins two DataFrames based on index  D) Sorts data in ascending order | 1 | 1 | 3 | 5 |  |
| 3 | Which of these is not a general technique to handle large datasets?  A) Data compression  B) Parallel processing  C) Data visualization  D) Batch learning | 1 | 1 | 3 | 5 |  |
| 4 | Which Python library enables parallel execution and optimization of computation flow?  A) Pandas  B) Matplotlib  C) Dask  D) Numexpr | 1 | 2 | 3 | 5 |  |
| 5 | What is the main purpose of using the melt() function in pandas?  A) To remove duplicates  B) To convert wide data into long format  C) To merge datasets  D) To perform statistical analysis | 1 | 2 | 3 | 5 |  |
| 6 | Which function is used to create a histogram in Matplotlib?  A) plot()  B) hist()  C) bar()  D) scatter() | 1 | 1 | 4 | 5 |  |
| 7 | What does plt.legend() do in a Matplotlib plot?  A) Sets the plot title  B) Adds a legend to the plot  C) Changes the axis labels  D) Saves the figure | 1 | 1 | 4 | 5 |  |
| 8 | Which of the following Seaborn functions helps to visualize pairwise relationships in a dataset?  A) jointplot()  B) pairplot()  C) distplot()  D) catplot() | 1 | 1 | 4 | 5 |  |
| 9 | Which parameter controls the resolution of the saved figure using savefig()?  A) dpi  B) bbox\_inches  C) pad\_inches  D) format | 1 | 2 | 5 | 5 |  |
| 10 | Which type of annotation includes text with arrows to highlight specific points?  A) Tick  B) Title  C) Callout  D) Label | 1 | 2 | 5 | 5 |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Register Number** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



**SRM Institute of Science and Technology**

Set -

**College of Engineering and Technology**

**School of Computing**

SRM Nagar, Kattankulathur – 603203, Chengalpattu District, Tamil Nadu

# Academic Year: 2024-25 (EVEN SEM)

Test: FT4 Date:29-04-2025

Course Code & Title: 21CSS303T-Data Science Duration: Two periods

Year& Sem: III Year /VI Sem Max.Marks:50

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Part – B** (4 x 5 = 20 Marks)  Instructions: Answer **ANY FOUR** Questions | | | | | | |
| Q.  No | Question | Marks | BL | CO | PO | PI Code |
| 11 | Explain the issues faced when handling large datasets and suggest suitable techniques to address them.  **Answer:**  **Issues:**   * **Memory overload**: Large datasets exceed available RAM, causing system slowdown or crashes. * **Slow processing**: Algorithms may become inefficient due to data volume. * **CPU starvation**: Inefficient use of processing power leads to idle CPU time. * **I/O bottlenecks**: Reading/writing large data to/from disk is slow.   **Techniques:**   * **Data compression**: Use tools like Bcolz to reduce memory usage. * **Chunking**: Process data in smaller batches. * **Parallelism**: Tools like Dask allow computations across multiple CPU cores. * **Efficient libraries**: Use optimized Python tools like Numexpr, Numba, and Theano. | 5 | 2 | 3 | 5 |  |
| 12 | Illustrate with examples how missing data is handled using pandas in Python.  **Answer:**  **Techniques:**   1. **Detect missing data**:  * df.isnull().sum()  1. **Drop missing data**:  * df.dropna() – removes rows with any NaNs. * df.dropna(axis=1) – removes columns with NaNs.  1. **Fill missing data**:   df.fillna(0)  ex:  df['Age'].fillna(df['Age'].mean(), inplace=True)  df['Salary'].fillna(df['Salary'].median(), inplace=True)   1. **Forward/Backward fill**:  * df.fillna(method='ffill') # Propagate previous value * df.fillna(method='bfill') # Propagate next value | 5 | 3 | 3 | 5 |  |
| 13 | Explain the difference between merge() and join() functions in pandas with suitable examples.   |  |  |  | | --- | --- | --- | | Feature | merge() | join() | | Basis | Joins on **columns** | Joins on **index** | | Flexibility | SQL-style joins | Simpler syntax | | Use Case | |  | | --- | |  |  |  | | --- | | Dataset joining by keys | | Combining based on index | | Example | merge() on a column  df = pd.merge(df1, df2, on='ID', how='inner') | join() on index  df1.set\_index('ID', inplace=True)  df2.set\_index('ID', inplace=True)  df1.join(df2, how='outer') | | 5 | 2 | 3 | 5 |  |
| 14 | Demonstrate how to create multiple subplots using Matplotlib and annotate a point in the plot.  **Creating subplots:**  import matplotlib.pyplot as plt  fig, axs = plt.subplots(1, 2) # 1 row, 2 columns  axs[0].plot([1, 2, 3], [4, 5, 6])  axs[1].plot([1, 2, 3], [6, 5, 4])  **Annotating a point:**  plt.annotate('Peak', xy=(2, 5), xytext=(2, 6),  arrowprops=dict(facecolor='black', arrowstyle='->')) | 5 | 3 | 4 | 5 |  |
| 15 | Explain the purpose and usage of Pair Plots and Joint Plots in Seaborn with example code.  **pairplot()**:   * Displays pairwise relationships in a dataset. * Useful for exploring patterns and correlations.   **Jointplot()**:   * Combines scatterplot and histograms. * Shows distribution and relationship of two variables.   **Example :**  import seaborn as sns  df = sns.load\_dataset("iris")  # Pair plot  sns.pairplot(df, hue="species")  # Joint plot  sns.jointplot(x='sepal\_length', y='sepal\_width', data=df) | 5 | 3 | 5 | 5 |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Part – C (2 x 10 = 20 Marks)**  Instructions: Answer ALL questions. | | | | | | |
| Q.  No | Question | Marks | BL | CO | PO | PI  Code |
| 16 a | Explain various data wrangling operations such as reshaping, pivoting, and merging in pandas with examples.  (7 Marks)  **Reshaping**:   * pivot() – converts long to wide format. * melt() – converts wide to long format.   **Merging**:   * merge() – combines datasets using key(s). * join() – merges using index. * concat() – appends datasets row/column-wise.   **Pivot**  df.pivot(index='Date', columns='City', values='Sales')  **Melt**  df.melt(id\_vars='Date', var\_name='City', value\_name='Sales')  **Merge**  pd.merge(df1, df2, on='ID', how='inner')  Examples to be given (3 marks) | 10 | 2 | 3 | 5 |  |
| **(OR)** | | | | | | |
| 16 b | Apply different data cleaning techniques such as handling missing data, standardization, and outlier detection using pandas.  **Handling Missing Data**:  dropna(), fillna() with mean/median/mode.  **Standardization**:  from sklearn.preprocessing import StandardScaler  scaler = StandardScaler()  df\_scaled = scaler.fit\_transform(df[['Age']])  **Outlier Detection**:  **IQR** method:  Q1 = df['Age'].quantile(0.25)  Q3 = df['Age'].quantile(0.75)  IQR = Q3 - Q1  outliers = df[(df['Age'] < Q1 - 1.5\*IQR) | (df['Age'] > Q3 + 1.5\*IQR)] | 10 | 3 | 3 | 5 |  |
| 17 a | Demonstrate how to create multiple subplots, control axes, and customize labels and legends using Matplotlib.  **Creating Subplots**: (5 marks)  fig, axs = plt.subplots(2, 2)  axs[0, 0].plot([1, 2, 3], [4, 5, 6])  **Control Axes**:  plt.xlim(0, 5)  plt.ylim(0, 10)   * Add xlabel(), ylabel() * Use legend() with labels * Add title with title()   **Annotation**:  plt.annotate('Point A', xy=(2, 5), xytext=(3, 6),  arrowprops=dict(facecolor='blue', arrowstyle='->'))  All plots to be drawn (5 marks) | 10 | 2 | 4 | 5 |  |
| **(OR)** | | | | | | |
| 17 b | Construct different Seaborn visualizations including pair plots, scatter plots, and joint plots, and explain their use in analysis.  (5 marks)  **Pair Plot**:  Explores multiple variables at once.  sns.pairplot(df)  **Scatter Plot**:   * Visualizes relationship between two variables.   sns.scatterplot(x='Age', y='Income', data=df)  **Joint Plot**:   * Combines scatter and histograms for deeper insight.   sns.jointplot(x='Age', y='Income', data=df)  (5marks)  Explain the plots using diagrams.  **Use cases**:   * Detect correlation * Identify clusters and trends * Explore distributions | 10 | 3 | 5 | 5 |  |

**Course Outcome (CO) and Bloom’s level (BL) Coverage in Questions**

|  |  |
| --- | --- |
|  |  |