

**Department of Computer Science and Engineering**

**29th Batch**

**Lab Final Report**

|  |  |
| --- | --- |
| Course title | : Artificial Intelligence Lab |
| Course Code | : CSE - 414 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Submitted By** | | **Submitted To** | |
|  |  |  |  |
| Name | : Md. Nahid Hasan | Name | : Md. Mahfujur Rahman |
| ID | : 221311131 | Designation | : Lecturer,  Varendra University,  Rajshahi. |
| Section | : D |  |
| Semester | : 8th |  |
| Batch | : 29th | Name | : D.M. Asadujjaman |
|  |  | Designation | : Lecturer,  Varendra University,  Rajshahi. |
|  |  |  |

-----------

|  |  |
| --- | --- |
| Signature | Signature |
|  |  |

1. **a) Using the Covertype[fetch\_covtype] available from Scikit-learn, perform Exploratory Data Analysis (EDA). Your analysis should include:**
2. Loading the dataset and converting it into a Pandas DataFrame.

|  |
| --- |
| **#** **Load libraries**  import pandas as pd  import numpy as np  import matplotlib.pyplot as plt  import seaborn as sns  from collections import Counter  **#** **Load and convert the dataset into DataFrame**  covtype = fetch\_covtype()  df = pd.DataFrame(covtype.data, columns=covtype.feature\_names)  df['Cover\_Type'] = covtype.target |

1. Displaying basic information such as shape, column names, data types, and the first few records.

|  |  |
| --- | --- |
| **#shape**  df.shape  **# column names**  print("\nColumn names:\n", df.columns)  **# data types**  **# first few records**  df.head(10) |  |

1. Checking for missing values.

|  |  |
| --- | --- |
| **#** **missing values**  print("Missing values:\n", df.isnull().sum()) |  |

1. Checking for missing values.

|  |  |
| --- | --- |
| **#** **distribution of the target variable**  X = covtype.data  plt.figure(figsize=(8, 4))  sns.countplot(x='Cover\_Type', data=df,color='blue', edgecolor='black')  plt.title("Distribution of Cover Type")  plt.show() |  |

1. Displaying descriptive statistics of the dataset.

|  |  |
| --- | --- |
| **#** **descriptive statistics of the dataset.**  df.describe() |  |

1. Checking for missing values.

|  |  |
| --- | --- |
| **#** **distribution of the target variable**  X = covtype.data  plt.figure(figsize=(8, 4))  sns.countplot(x='Cover\_Type', data=df,color='blue', edgecolor='black')  plt.title("Distribution of Cover Type")  plt.show() |  |

1. Checking for missing values.

|  |  |
| --- | --- |
| **#** **distribution of the target variable**  X = covtype.data  plt.figure(figsize=(8, 4))  sns.countplot(x='Cover\_Type', data=df,color='blue', edgecolor='black')  plt.title("Distribution of Cover Type")  plt.show() |  |

* **Conclusion:**

In this lab, I created a dataset of 150 samples and applied the KNN algorithm to classify sports from height and weight data. I implemented a loop that takes user input repeatedly and provides predictions with real-time plots. This helped me understand both algorithm logic and user interaction in a data-driven system.