EDS PROJECT

GUIDED BY MADHAVI NIMKAR

PRESENTED BY
GAYATRI NAROTE(643)
VAISHNAVI PAWAR(649)
KANCHAN GAIKWAD(652)
KIRAN SHINDE(655)

INTRODUCTION

- Python has a simple syntax similar to the English language. Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- This project involves data manipulation, data visualization and linear regression.
- We choose cricket dataset for this project

MOTIVATION

- The cricket dataset is always big because its involves sixes, fours, century and so many things.
- The curiosity of manipulating such a big dataset.
- Its include numerical as well as float datatypes

DETAILS OF DATASET

- The provided data appears to be a list of cricket players with their respective statistics in Twenty20 International matches
- The table includes information such as the player's name, their country or representation, the span of their career, the number of matches played (Mat), innings batted (Inns), not outs (Not out), total runs scored (Runs), highest score (HS), batting average (Ave), balls faced, strike rate (SR), number of centuries (100), number of half-centuries (50), number of times dismissed without scoring (0), number of fours hit (4s), and number of sixes hit (6s).

DATA MANIPULATION

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

df=pd.read_csv('/content/cricket.csv')

df
```

Player	Span	Mat	Inns	Not out	Runs	HS	Ave	ball faced
	SR	100	50	0	4s	6s		
0	V Kohli (INDIA)		2010-2019	75	70	20	2633	94*
	52.66	1907	138.07	0	24	2	247	71
1	RG Sharma (INDIA)		2007-2019	104	96	14	2633	118
	32.10	1905	138.21	4	19	6	234	120
2	MJ Guptill (NZ)		2009-2019	83	80	7	2436	105
	33.36	1810	134.58	2	15	2	215	113
3	Shoaib Malik (ICC/PAK)		2006-2019	111	104	30	2263	75
	30.58	1824	124.06	0	7	1	186	61
4	BB McCullum (NZ)		2005-2015	71	70	10	2140	123
	35.66	1571	136.21	2	13	3	199	91
5	DA Warner (AUS)		2009-2019	76	76	8	2079	100*
	30.57	1476	140.85	1	15	5	203	86
6	EJG Morgan (ENG)		2009-2019	86	84	17	2002	91
	29.88	1475	135.72	0	11	3	151	96

```
# Perform data manipulation operations

# Example: Sorting the dataframe by Runs in descending order

df_sorted = df.sort_values(by='Runs', ascending=False)


# Print the manipulated dataframe

print(df_sorted)
```

OUTPUT

```
0          V Kohli (INDIA)  2010-2019   75    70      20  2633   94*
1        RG Sharma (INDIA)  2007-2019  104    96      14  2633   118
2          MJ Guptill (NZ)  2009-2019   83    80       7  2436   105
3     Shoaib Malik (ICC/PAK)  2006-2019  111   104      30  2263    75
4          BB McCullum (NZ)  2005-2015   71    70      10  2140   123
5          DA Warner (AUS)  2009-2019   76    76       8  2079  100*
6          EJG Morgan (ENG)  2009-2019   86    84      17  2002    91
7     Mohammad Shahzad (AFG)  2010-2018   65    65       3  1936  118*
8            JP Duminy (SA)  2007-2019   81    75      25  1934   96*
9          PR Stirling (IRE)  2009-2019   72    71       6  1929    91
10    Mohammad Hafeez (PAK)  2006-2018   89    86       8  1908    86
11         TM Dilshan (SL)  2006-2016   80    79      12  1889  104*
12          AJ Finch (AUS)  2011-2019   58    58       9  1878   172
13         LRPL Taylor (NZ)  2006-2019   95    87      19  1743    63
14          Umar Akmal (PAK)  2009-2019   84    79      14  1690    94
15       AB de Villiers (SA)  2006-2017   78    75      11  1672   79*
16        H Masakadza (ZIM)  2006-2019   66    66       2  1662   93*
17          AD Hales (ENG)  2011-2019   60    60       7  1644  116*
18           CH Gayle (WI)  2006-2019   58    54       4  1627   117
```

DATA VISUALIZATION

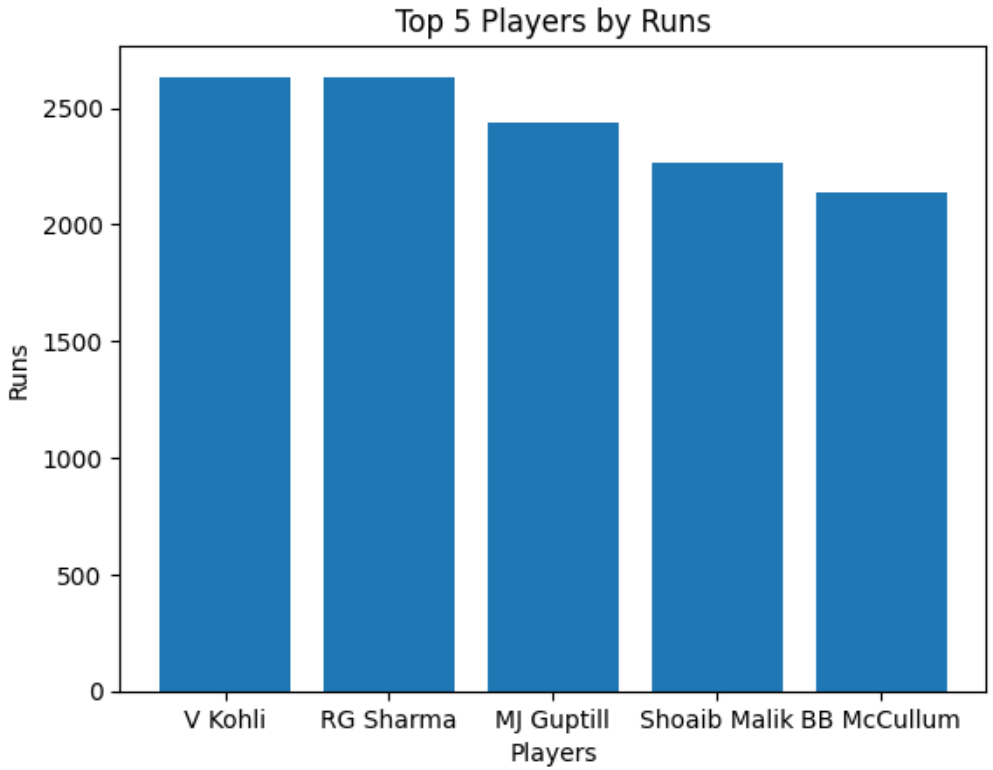
```
import matplotlib.pyplot as plt

# Prepare data
players = ['V Kohli', 'RG Sharma', 'MJ Guptill', 'Shoaib Malik', 'BB McCullum']
runs = [2633, 2633, 2436, 2263, 2140]

# Create a bar chart
plt.bar(players, runs)

# Customize the chart
plt.title('Top 5 Players by Runs')
plt.xlabel('Players')
plt.ylabel('Runs')

# Display the chart
plt.show()
```



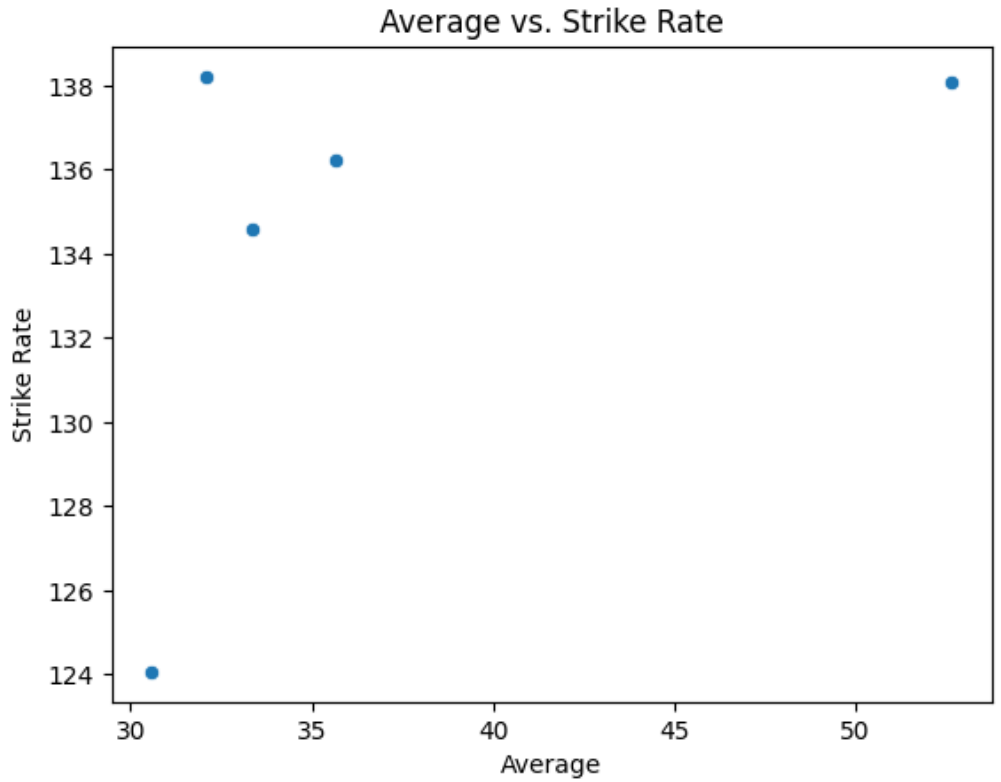
```
import seaborn as sns

# Prepare data
averages = [52.66, 32.1, 33.36, 30.58, 35.66]
strike_rates = [138.07, 138.21, 134.58, 124.06, 136.21]

# Create a scatter plot
sns.scatterplot(x=averages, y=strike_rates)

# Customize the plot
plt.title('Average vs. Strike Rate')
plt.xlabel('Average')
plt.ylabel('Strike Rate')

# Display the plot
plt.show()
```



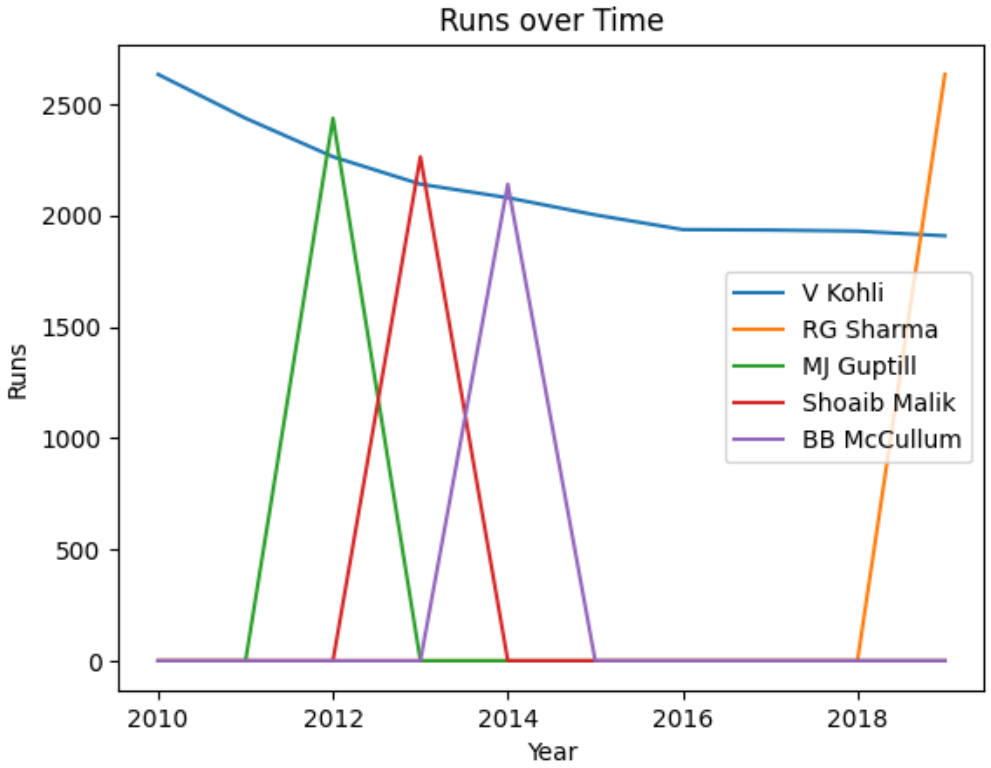
```
import matplotlib.pyplot as plt

# Prepare data
players = ['V Kohli', 'RG Sharma', 'MJ Guptill', 'Shoaib Malik', 'BB McCullum']
years = range(2010, 2020)  # Assuming the data is for 2010-2019
runs = [
    [2633, 2436, 2263, 2140, 2079, 2002, 1936, 1934, 1929, 1908],
    [0, 0, 0, 0, 0, 0, 0, 0, 0, 2633],
    [0, 0, 2436, 0, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 2263, 0, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 2140, 0, 0, 0, 0, 0]
]

# Create a line chart for each player
for i in range(len(players)):
    plt.plot(years, runs[i], label=players[i])

# Customize the chart
plt.title('Runs over Time')
plt.xlabel('Year')
plt.ylabel('Runs')
plt.legend()

# Display the chart
plt.show()
```


Runs over Time
Runs
Year
2500
2000
1500
1000
500
0
2010
2012
2014
2016
2018
V Kohli
RG Sharma
MJ Guptill
Shoaib Malik
BB McCullum

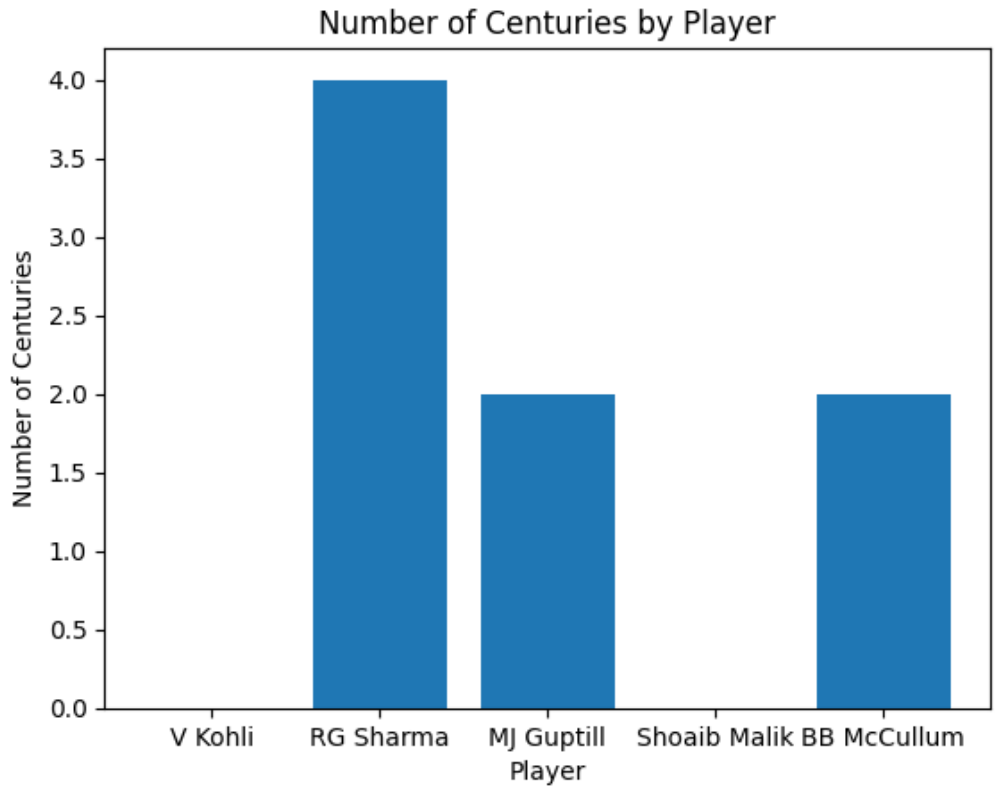
```
import matplotlib.pyplot as plt

# Prepare data
players = ['V Kohli', 'RG Sharma', 'MJ Guptill', 'Shoaib Malik', 'BB McCullum']
centuries = [0, 4, 2, 0, 2]

# Create a bar chart
plt.bar(players, centuries)

# Customize the chart
plt.title('Number of Centuries by Player')
plt.xlabel('Player')
plt.ylabel('Number of Centuries')

# Display the chart
plt.show()
```



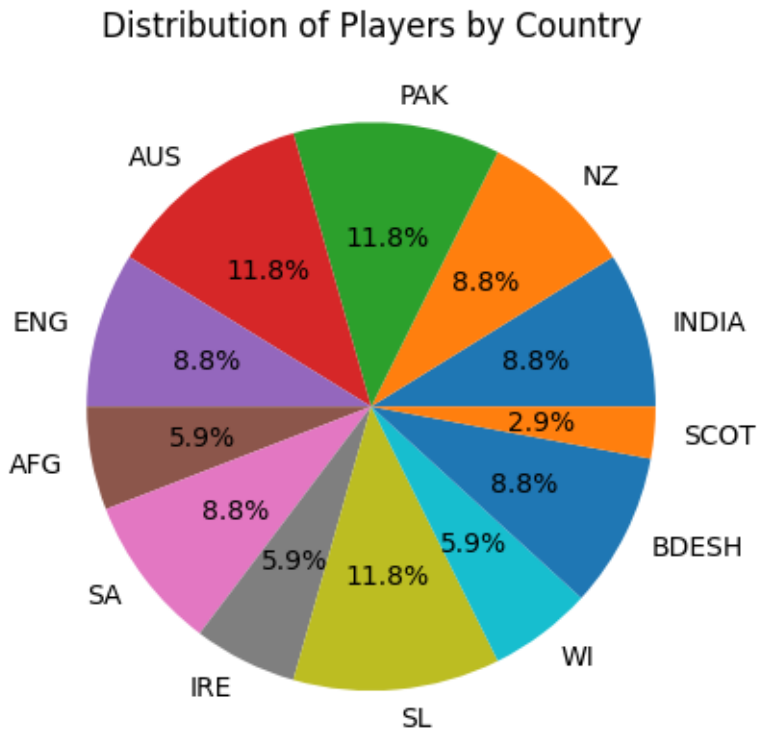
```
import matplotlib.pyplot as plt

# Data for the pie chart
countries = ['INDIA', 'NZ', 'PAK', 'AUS', 'ENG', 'AFG', 'SA', 'IRE', 'SL', 'WI', 'BDESH', 'SCOT']
counts = [3, 3, 4, 4, 3, 2, 3, 2, 4, 2, 3, 1]

# Create the pie chart
plt.pie(counts, labels=countries, autopct='%1.1f%%')

# Add a title
plt.title('Distribution of Players by Country')

# Display the chart
plt.show()
```



LINEAR REGRESSION (LR)

```
import pandas as pd

from sklearn.linear_model import LinearRegression


# Create a dataframe from the given data

data = {

    'Player': ['V Kohli', 'RG Sharma', 'MJ Guptill', 'Shoaib Malik', 'BB McCullum', 'DA Warner', 'EJG Morgan'],

    'Inns': [70, 96, 80, 104, 70, 76, 84],

    'Runs': [2633, 2633, 2436, 2263, 2140, 2079, 2002]

}


df = pd.DataFrame(data)


# Prepare the data for linear regression

X = df[['Inns']]  # Independent variable (number of innings)

y = df['Runs']    # Dependent variable (runs)


# Create a linear regression model

model = LinearRegression()


# Fit the model to the data

model.fit(X, y)


# Predict runs for a player with 90 innings

predicted_runs = model.predict([[90]])


print(f"Predicted runs for a player with 90 innings: {predicted_runs[0]:.2f}")
```

OUTPUT

```
Predicted runs for a player with 90 innings: 2327.58
```

```
import pandas as pd

from sklearn.linear_model import LinearRegression


# Create a dataframe from the given data

data = {

    'Player': ['V Kohli', 'RG Sharma', 'MJ Guptill', 'Shoaib Malik', 'BB McCullum', 'DA Warner', 'EJG Morgan'],

    '4s': [247, 234, 215, 186, 199, 203, 151],

    '6s': [71, 120, 113, 61, 91, 86, 96],

    'Runs': [2633, 2633, 2436, 2263, 2140, 2079, 2002]

}


df = pd.DataFrame(data)
```

```
# Prepare the data for linear regression

X = df[['4s', '6s']]  # Independent variables (number of 4s and 6s)

y = df['Runs']        # Dependent variable (runs)


# Create a linear regression model

model = LinearRegression()


# Fit the model to the data

model.fit(X, y)


# Predict runs for a player with 200 4s and 100 6s

predicted_runs = model.predict([[200, 100]])


print(f"Predicted runs for a player with 200 4s and 100 6s: {predicted_runs[0]:.2f}")
```

OUTPUT

```
Predicted runs for a player with 200 4s and 100 6s: 2286.65
```

APPLICATION

- A DML (data manipulation language) refers to a computer programming language that allows you to add (insert), delete (delete), and alter (update) data in a database.
- The graphical depiction of information and data is known as data visualisation. Data visualisation tools make it easy to view and comprehend trends, outliers, and patterns in data by utilising visual components like charts, graphs, and maps.
- The data manipulation helps to understand data easily.
- Data manipulation helps to sort the data and to find the highest and average runs of a player
- Linear Regression helps to predict the runs of a player.

CONCLUSION

- In conclusion ,we can analysis data of cricket dataset in various way.
- Data manipulation, Data visualization and linear regression helps to analysis the data and it easier to understand.
- And we can get exact meaning of data by using this function.