Using Data Science and Big Data Analytics: NBA Player and Team Analysis

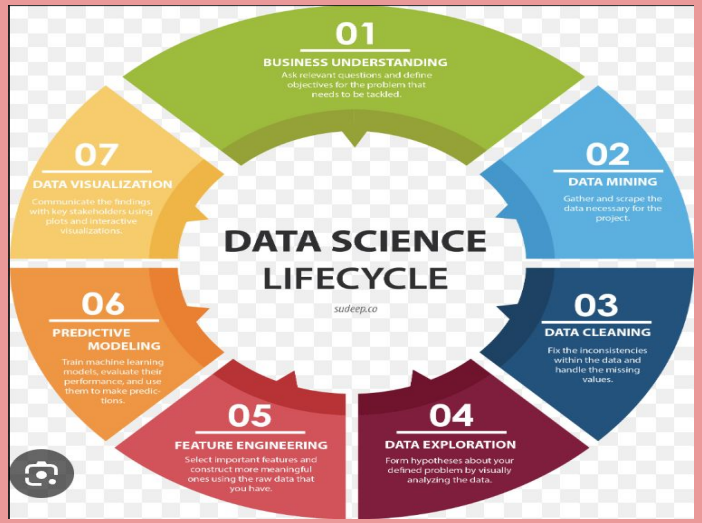
A Deep Dive into Patterns and Insights from NBA Statistics (1996–2022)

Eduardo Polanco
West Los Angeles College
CS 131

Introduction to Data Science and Big Data Analytics

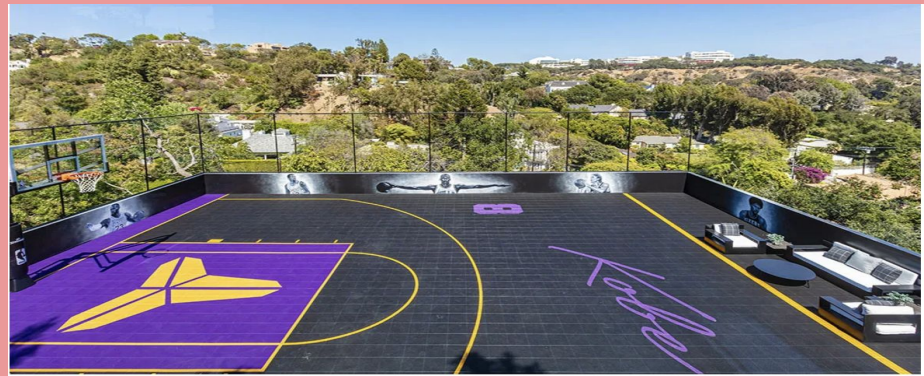
What is Data Science?

- Data Science is the process of analyzing and interpreting data to uncover patterns and insights.
- It combines programming, statistics, and domain knowledge to solace real-world problems and make data-driven decisions.



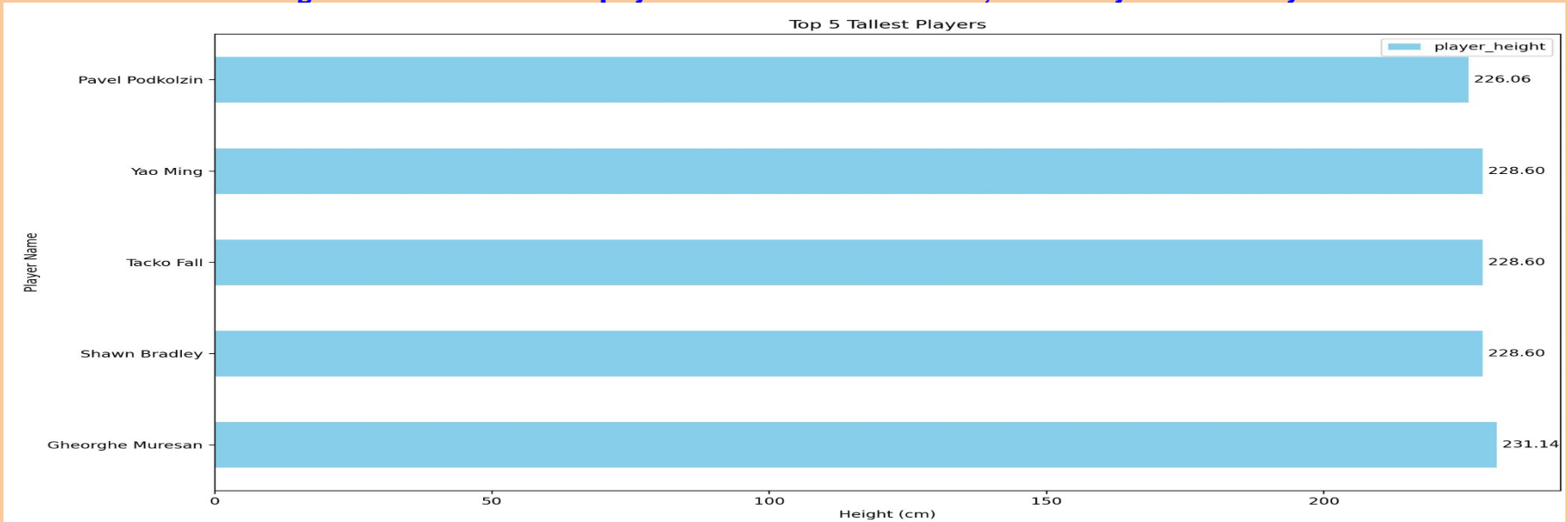
Big Data Analytics and my project

- Big Data Analytics involves analyzing massive datasets to uncover trends and make data-driven decisions.
- This project uses NBA data spanning 26 years (1996-2022) with thousands of rows of player and team stats.
- By applying Python, I performed analyses like player performance, team trends, and demographic distributions.



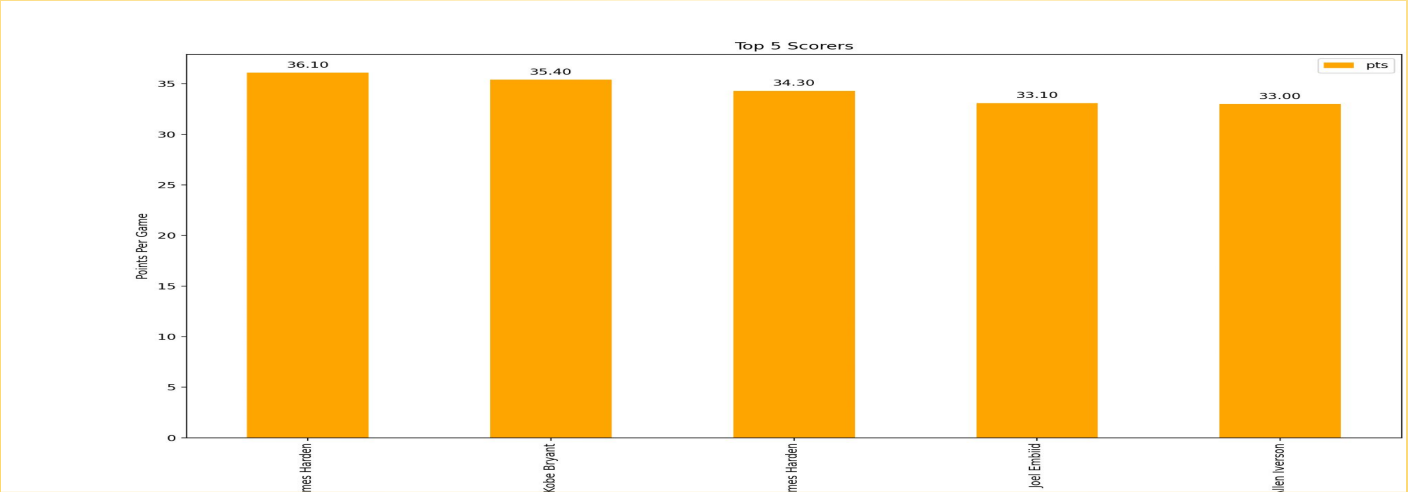
Top 5 Tallest Players

- **Purpose of the Analysis:**
  - This analysis identifies the tallest players in the NBA dataset, showcasing their unique physical attributes.
  - Height is a key advantage in basketball particularly for positions like center.
- **Data Science Connection:**
  - I applied sorting and filtering techniques to analyze player heights.
  - This demonstrates how data science can extract meaningful insights from large datasets.
- **Key Insight:**
  - ***Gheorghe Muresan*** **is the tallest player in the dataset at 231.14 cm, followed by** ***Shawn Bradley*** **at 228.60 cm.**



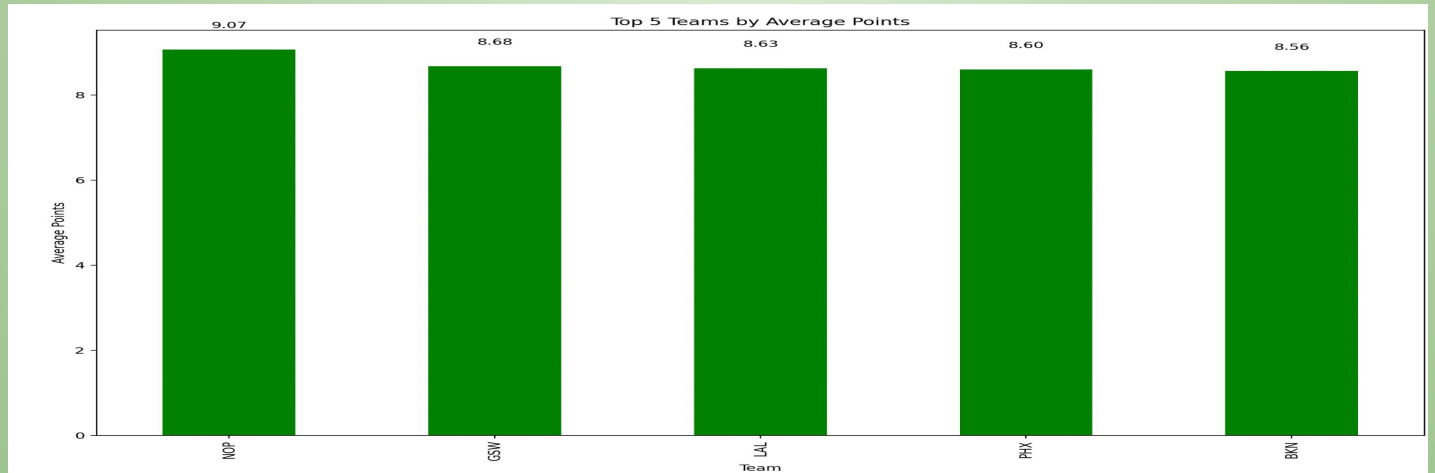
Top 5 Scorers

- **Purpose of the Analysis:**
  - This analysis identifies the top 5 NBA players with the highest points per game(PPG).
  - Points per game is a critical metric to measure a player's offensive performance.
- **Data Science Connection:**
  - I used Python to sort and filter the dataset to identify top-performing players.
  - Visualization techniques help highlight key patterns in player performance.
- **Key Insight:**
  - James Harden leads the list with 36.10 points per game, showcasing exceptional scoring ability, followed by Kobe Bryant with 35.40 points per game and other top scorers demonstrating elite performance. .



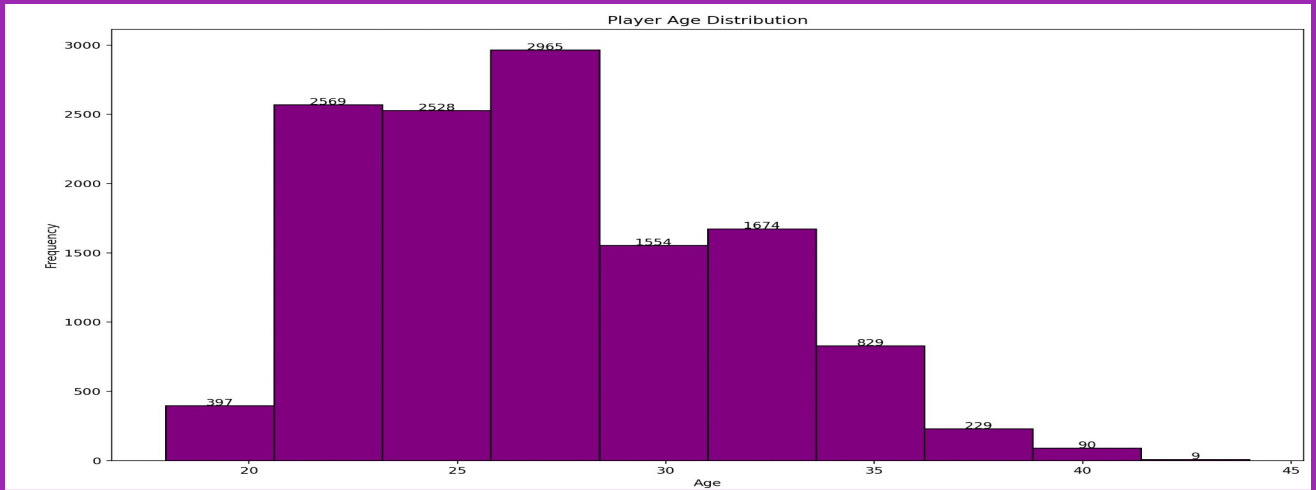
Team Comparisons - Average Points

- **Purpose of the Analysis:**
  - This analysis identifies the top 5 NBA teams based on their average points per game across all seasons.
  - Team performance trends provide insight into overall competitiveness and offensive capabilities.
- **Data Science Connection:**
  - I used Python to group the dataset by teams and calculated average points per game.
  - This demonstrates how aggregation techniques in data science can highlight team-level performance from player-level data.
- **Key Insight:**
  - The New Orleans Pelicans (NOP) lead the list with an average of 9.07 points per game, followed by the Golden State Warriors (GSW) with 8.68 and the Los Angeles (LAL) with 8.63 points per game.



Player Age Distribution

- **Purpose of the Analysis:**
  - This analysis examines the distribution of NBA player ages across the dataset.
  - Age plays a crucial role in determining player performance, career longevity, and peak years.
- **Data Science Connection:**
  - I used Python's histogram function to visualize how player ages are distributed.
  - This demonstrates how data science can uncover demographic trends within a dataset.
- **Key Insight:**
  - The majority of players fall between 25 and 30 years old, with 2.965 players in the 30-year age group being the highest frequency.
  - Fewer players are seen at the extremes, with only 397 players under 20 and just 9 players aged 40 or older.



Python Code: A quick Breakdown

Key Components of the code

- **Data Import and preview:**
  - Dataset loaded using pandas for structured data analysis.
  - Used .head( ) to explore the dataset's structure and ensure data accuracy.
- **Analysis Workflow:**
  - Tallest Players: Grouped players, sorted by height, and plotted a horizontal bar chart with labels.
  - Top Scorers: Filtered the data set for players with the highest points per game and visualized it using labeled bar chart.
  - Team Averages: Used groupby( ) to calculate average points per team and highlighted top-performing teams.
  - Player Age Distribution: created a histogram to illustrate the spread of player ages, annotated with frequices for clarity.
- **Visualization Enhancements:**
  - Added numerical labels directly on charts for better interpretation.
  - Used color coding for distinct data categories.

How the Code Supports the Analysis

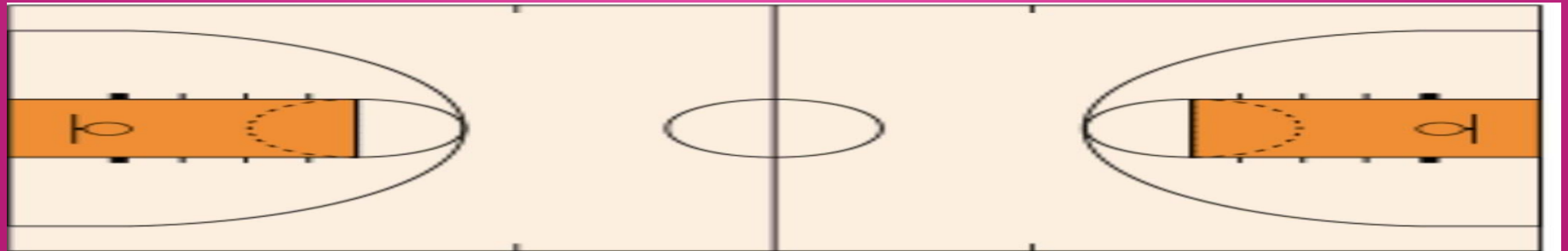
- **Python's flexibility and librairies like pandas and matplotlib enabled efficient data manipulation and visualization.**
- **Automated tasks like grouping, filtering, and annotations charts saved time and ensured accuracy.**

CODE USED

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the dataset
data = pd.read_csv('all_seasons.csv')

# Preview the dataset
print("Dataset Preview:")
print(data.head())

# ----- Analysis 1: Top 5 Tallest Players -----
# Group data by player_name and retain the max height only for numeric columns
unique_tallest = data.groupby('player_name', as_index=False).agg({
    'player_height': 'max',
    'team_abbreviation': 'first'  # Keep the first team abbreviation
})

# Get the top 5 tallest players
top_tallest = unique_tallest.nlargest(5, 'player_height')

print("Top 5 Tallest Players:")
print(top_tallest[['player_name', 'player_height', 'team_abbreviation']])

# Horizontal bar chart of the top 5 tallest players with values
ax = top_tallest.plot(x='player_name', y='player_height', kind='barh', title='Top 5 Tallest Players', color='skyblue')
plt.xlabel("Height (cm)")
plt.ylabel("Player Name")
for i, v in enumerate(top_tallest['player_height']):
    ax.text(v + 1, i, f"{v:.2f}", color='black', va='center')  # Add values
plt.show()

# ----- Analysis 2: Top 5 Players by Points Per Game -----
top_scorers = data.nlargest(5, 'pts')
print("Top 5 Scorers:")
print(top_scorers[['player_name', 'pts', 'team_abbreviation']])

# Bar chart of the top 5 scorers with values
ax = top_scorers.plot(x='player_name', y='pts', kind='bar', title='Top 5 Scorers', color='orange')
plt.xlabel("Player Name")
plt.ylabel("Points Per Game")
for i, v in enumerate(top_scorers['pts']):
    ax.text(i, v + 0.5, f"{v:.2f}", color='black', ha='center')  # Add values
plt.show()

# ----- Analysis 3: Team Comparisons - Average Points -----
team_avg_pts = data.groupby('team_abbreviation')['pts'].mean().sort_values(ascending=False).head(5)
print("Top 5 Teams by Average Points:")
print(team_avg_pts)

# Bar chart of average points by team with values
ax = team_avg_pts.plot(kind='bar', title='Top 5 Teams by Average Points', color='green')
plt.xlabel("Team")
plt.ylabel("Average Points")
for i, v in enumerate(team_avg_pts):
    ax.text(i, v + 0.5, f"{v:.2f}", color='black', ha='center')  # Add values
plt.show()

# ----- Analysis 4: Player Age Distribution -----
print("Player Age Distribution:")
ax = data['age'].plot(kind='hist', bins=10, title='Player Age Distribution', color='purple', edgecolor='black')
plt.xlabel("Age")
plt.ylabel("Frequency")
# Optional: Add frequency values above each bar in the histogram
n, bins, patches = plt.hist(data['age'], bins=10, color='purple', edgecolor='black')
for i in range(len(patches)):
    ax.text(bins[i] + (bins[1] - bins[0]) / 2, n[i] + 0.5, int(n[i]), ha='center', color='black')
plt.show()
```

Conclusion & Key Takeaways

- **Summary of Findings:**
  - This project analyzed NBA data spanning 26 years (1996 - 2023) to uncover key trends and insights.
  - Key analyses included player height, scoring performance, team averages, and age distribution.
- **Data Science Connections:**
  - Using Python, I demonstrated data cleaning, sorting, grouping, and visualization to analyze a large dataset.
  - These techniques reflect how data science can turn raw data into actionable insights.
- **Reflection:**
  - This project enhanced my skills in Python programming, data visualization, and storytelling through analytics.
  - It also highlighted the importance of data integrity and visualization in communicating findings effectively.
- **Next Steps or Applications:**
  - These analyses could be expanded to predict future player performance trends using machine learning.
  - Additional factors like injuries, player positions, or team strategies could be incorporated.



References

1. Justin Cirtautas. “NBA Players Dataset.” Kaggle.

   Biometric, biographic, and basic box score stats from 1996 to 2022 season.

   https://www.kaggle.com/datasets/justinas/nba-players-data?select=all_seasons.csv