Project Title: AI-Based Number Guessing Game

Problem Statement: The goal of this project is to create an AI-based number guessing game that uses Python and libraries such as NumPy, Pandas, Matplotlib, and Seaborn to generate, analyze, and visualize the AI's guessing performance.

Name: Ayushi Mishra

Roll No: 202401100400069

Introduction

The AI-based number guessing game is a simple yet engaging problem where an AI attempts to guess a randomly generated number within a defined range. The AI makes guesses iteratively until it finds the correct number. The performance of the AI is visualized using various types of graphs, such as line plots, scatter plots, histograms, and box plots. This project helps in understanding random number generation, data visualization, and Python programming concepts.

Methodology

1. **Define the Problem:** The AI must guess a randomly generated number between a given range.
2. **Generate Random Numbers:** The actual number is randomly generated using NumPy.
3. **AI Guessing Mechanism:** The AI makes random guesses within the range until it finds the correct number.
4. **Data Storage:** Each attempt is recorded in a Pandas DataFrame.
5. **Visualization:** Various plots (line, scatter, histogram, box) are created using Matplotlib and Seaborn to analyze the AI's performance.
6. **Execution & Analysis:** The number of attempts and guessing pattern are examined.

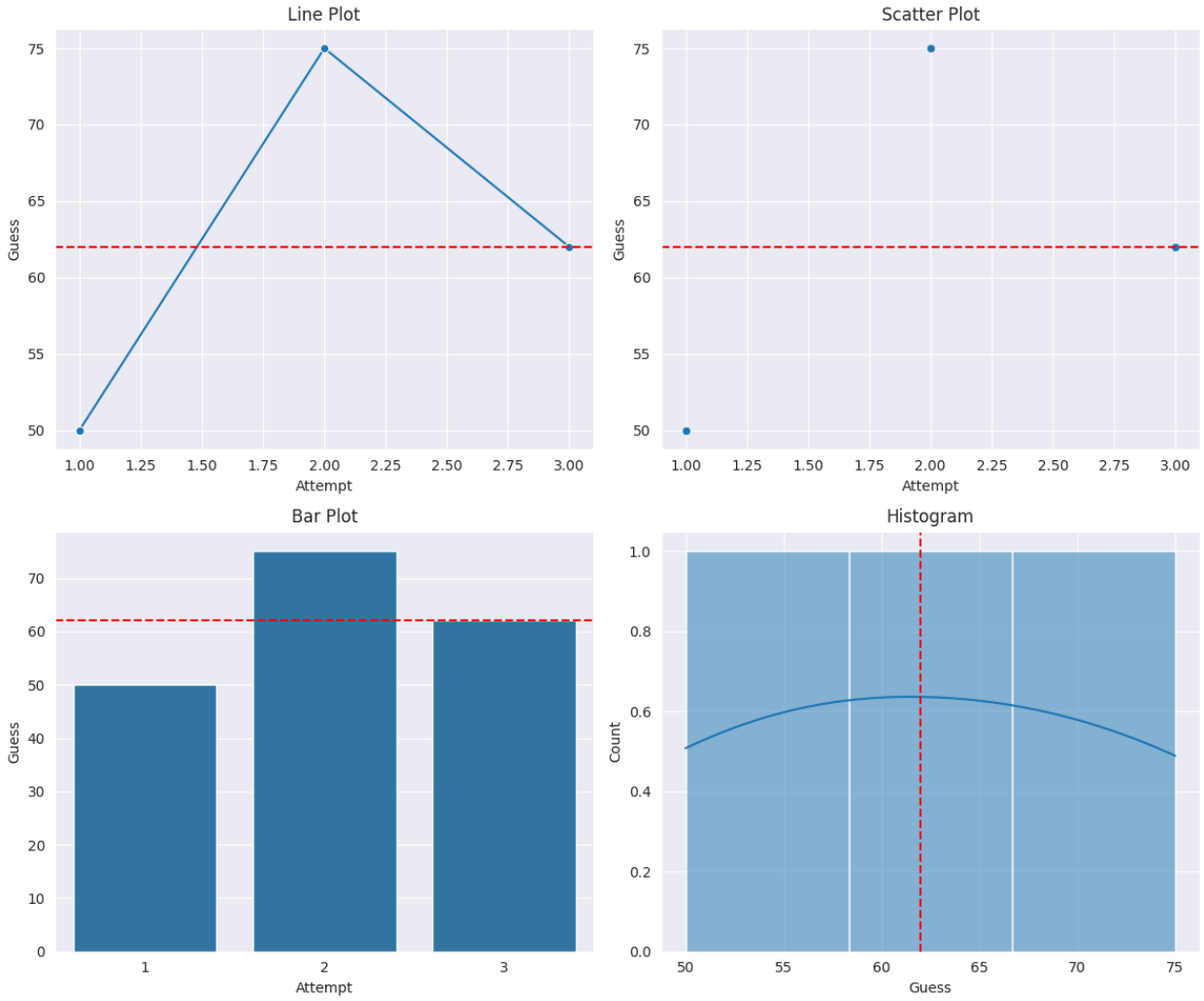
Code

```
import numpy as np
import pandas as pd
```

```
import matplotlib.pyplot as plt
import seaborn as sns

def ai_guess_number(low, high, actual_number):
    attempts = []
    guess = None
    while guess != actual_number:
        guess = np.random.randint(low, high + 1)
        attempts.append(guess)
    return attempts

def plot_attempts(attempts, actual_number):
    df = pd.DataFrame({'Attempt': range(1, len(attempts) + 1),
'Guess': attempts})

    plt.figure(figsize=(15, 10))

    # Line Plot
    plt.subplot(2, 2, 1)
    sns.lineplot(data=df, x='Attempt', y='Guess', marker='o',
label='AI Guess')
    plt.axhline(actual_number, color='r', linestyle='--',
label=f'Actual Number: {actual_number}')
    plt.xlabel("Attempt Number")
    plt.ylabel("Guessed Number")
    plt.title("AI Number Guessing Progress (Line Plot)")
    plt.legend()

    # Scatter Plot
    plt.subplot(2, 2, 2)
    sns.scatterplot(data=df, x='Attempt', y='Guess', color='b')
    plt.axhline(actual_number, color='r', linestyle='--',
label=f'Actual Number: {actual_number}')
    plt.xlabel("Attempt Number")
    plt.ylabel("Guessed Number")
    plt.title("AI Number Guessing Progress (Scatter Plot)")
    plt.legend()

    # Histogram
```

```
    plt.subplot(2, 2, 3)
    sns.histplot(df['Guess'], bins=10, kde=True, color='g')
    plt.axvline(actual_number, color='r', linestyle='--',
label=f'Actual Number: {actual_number}')
    plt.xlabel("Guessed Number")
    plt.ylabel("Frequency")
    plt.title("Distribution of Guessed Numbers (Histogram)")
    plt.legend()

    # Box Plot
    plt.subplot(2, 2, 4)
    sns.boxplot(y=df['Guess'], color='purple')
    plt.axhline(actual_number, color='r', linestyle='--',
label=f'Actual Number: {actual_number}')
    plt.ylabel("Guessed Number")
    plt.title("Guess Spread (Box Plot)")
    plt.legend()

    plt.tight_layout()
    plt.show()

# Run the game
low, high = 1, 100
actual_number = np.random.randint(low, high + 1)

print(f"AI is trying to guess a number between {low} and {high}...")

attempts = ai_guess_number(low, high, actual_number)
print(f"AI guessed the number {actual_number} in {len(attempts)}
attempts.")

plot_attempts(attempts, actual_number)
```

Output/Result



References/Credits

- NumPy Documentation: https://numpy.org/doc/
- Pandas Documentation: https://pandas.pydata.org/docs/
- Matplotlib Documentation: https://matplotlib.org/stable/contents.html
- Seaborn Documentation: https://seaborn.pydata.org/
- Concept Image: [Image Source if applicable]

This report outlines the complete implementation of the AI-based number guessing game, detailing the problem, methodology, code, and analysis.