



Experiment - 2

WEATHER MODELLING USING QUADRATIC EQUATION

Software Engineering

By

Vulasala Sujan (BU22CSEN0101959)

Meti Chaitanya (BU22CSEN0101523)

Maraka Ganesh (BU22CSEN0101803)

J Bhargav Reddy (BU22CSEN0101198)

Under the Guidance of

Kerenalli Sudarshana (700542)

Gandhi Institute of Technology and Management

(DEEMED TO BE UNIVERSITY)

BENGALURU, KARNATAKA, INDIA

Academic Year 2024-25

INDEX

- **Aim**
- **Algorithm Development**
- **Execution**
- **Result**

Aim: - To implement weather modelling using the quadratic solution across three software development approaches:

1. **Waterfall Model:** Sequential implementation of stages.
2. **Iterative Model:** Incremental development with refinements at each iteration.
3. **Agile Model:** Collaborative and adaptive development in sprints.

The implementation includes:

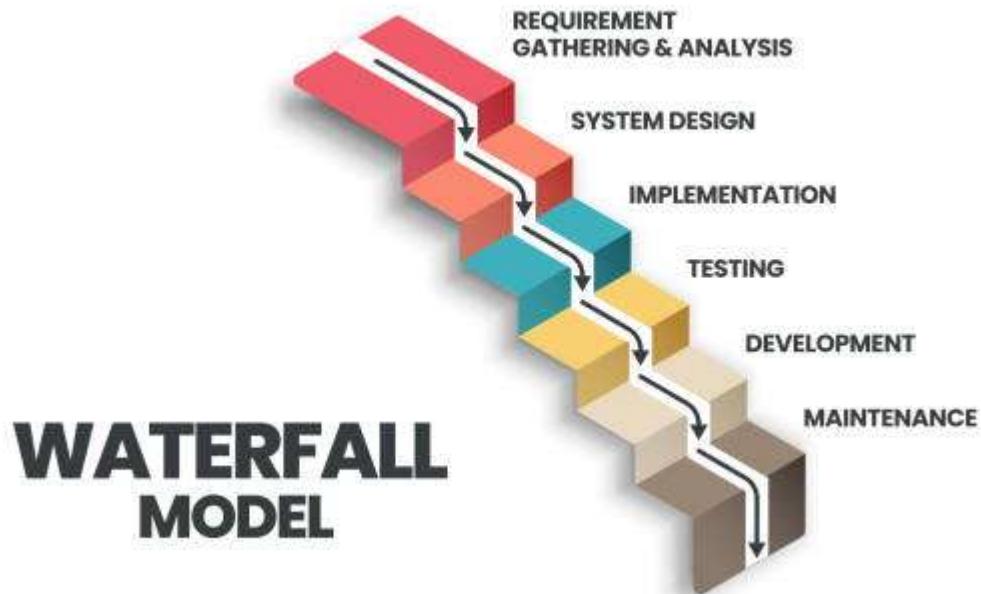
- Hard-coding variables.
- Keyboard input.
- File input for a single set of data.
- File input for multiple sets of data.
- All versions will be tested, debugged, and managed using GitHub for version control.

Algorithm Development: -

1. Waterfall Model

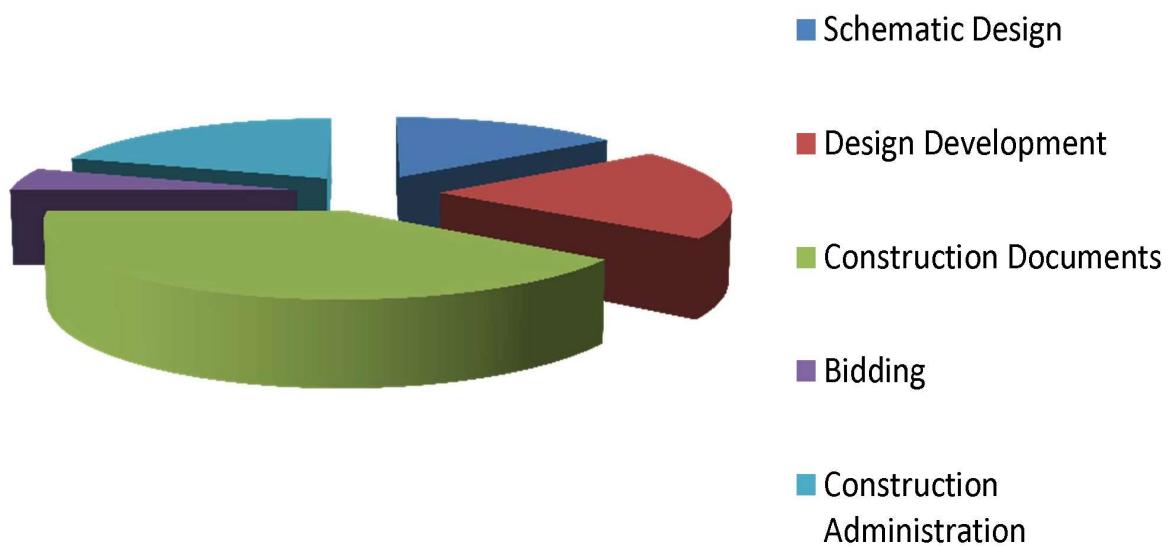
1. **Step 1:** Analyse and finalize requirements for the weather modelling system.
2. **Step 2:** Design the solution and prepare the algorithm for quadratic modelling.

3. **Step 3:** Implement Version 1 (hard-coded variables).
Test and debug the solution.
4. **Step 4:** Implement Version 2 (keyboard input). Test and debug the solution.
5. **Step 5:** Implement Version 3 (file input for a single set of data). Test and debug the solution.
6. **Step 6:** Implement Version 4 (file input for multiple sets of data). Test and debug the solution.
7. **Step 7:** Deliver the final product and document the entire process.



- **Design Phase:** Then develop the architecture and system flow for quadratic modelling.

Design Phases (%)

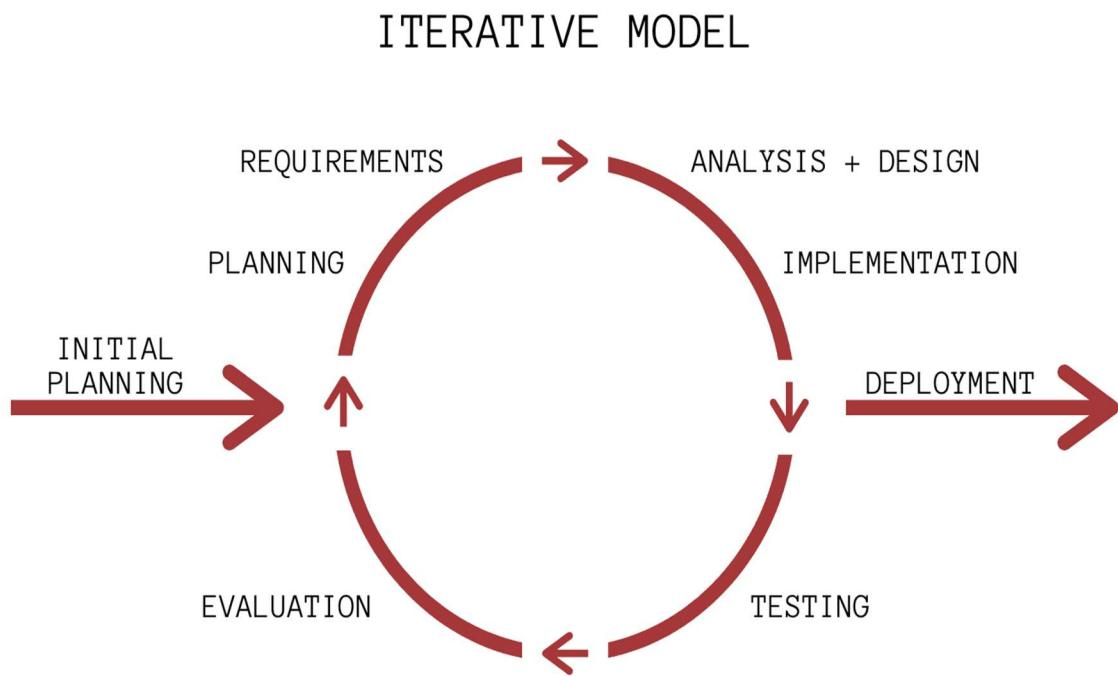


- **Implementation Phase:** Then Code the quadratic model in stages (hard coded, keyboard input, file input).

2. Iterative Model

1. **Step 1:** Identify the core functionality (quadratic modelling with hard-coded variables) and implement Version 1.
2. **Step 2:** Test Version 1 and refine based on feedback.
3. **Step 3:** Add Version 2 functionality (keyboard input) as the next iteration. Test and debug.
4. **Step 4:** Extend to Version 3 (file input for a single set of data). Test and refine.

5. **Step 5:** Extend to Version 4 (file input for multiple sets of data). Test and refine.
6. **Step 6:** Deliver the system incrementally, ensuring thorough testing and feedback at each stage.



3. Agile Model

1. **Sprint 1:** Develop Version 1 (hard-coded variables) and conduct a team review.
2. **Sprint 2:** Implement Version 2 (keyboard input), test, and adapt based on feedback.
3. **Sprint 3:** Extend to Version 3 (file input for a single set of data) with collaborative team effort.

4. **Sprint 4:** Extend to Version 4 (file input for multiple sets of data) and conduct final testing.
5. Deliver the product incrementally after every sprint. Use retrospective meetings to refine the process.



Execution: -

1)Waterfall Model: -

a) Version 1(Hard coding variables): -

```
▶ import numpy as np
temp_coeffs = [0.01, -0.3, 25]
rainfall_coeffs = [0.005, -0.1, 5]
humidity_coeffs = [0.002, -0.05, 80]
days = [1, 100, 200, 365]
def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c
temperature = [quadratic_model(*temp_coeffs, day) for day in days]
rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
print("Waterfall Model - Version 1 (Hardcoded Variables):")
print(f"Days: {days}")
print(f"Temperature: {temperature}")
print(f"Rainfall: {rainfall}")
print(f"Humidity: {humidity}")
```

```
→ Waterfall Model - Version 1 (Hardcoded Variables):
Days: [1, 100, 200, 365]
Temperature: [24.71, 95.0, 365.0, 1247.75]
Rainfall: [4.905, 45.0, 185.0, 634.625]
Humidity: [79.952, 95.0, 150.0, 328.2]
```

b) Version 2 (Keyboard Input):-

```
▶ def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c
print("Enter coefficients for Temperature (a, b, c):")
temp_coeffs = [float(input(f"Enter coefficient {coef}: ")) for coef in ['a', 'b', 'c']]
print("Enter coefficients for Rainfall (a, b, c):")
rainfall_coeffs = [float(input(f"Enter coefficient {coef}: ")) for coef in ['a', 'b', 'c']]
print("Enter coefficients for Humidity (a, b, c):")
humidity_coeffs = [float(input(f"Enter coefficient {coef}: ")) for coef in ['a', 'b', 'c']]
days = [1, 100, 200, 365]
temperature = [quadratic_model(*temp_coeffs, day) for day in days]
rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
print("\nResults:")
print(f"Days: {days}")
print(f"Temperature: {temperature}")
print(f"Rainfall: {rainfall}")
print(f"Humidity: {humidity}")
```

```
▶ Enter coefficients for Temperature (a, b, c):
Enter coefficient a: 0.2
Enter coefficient b: 5
Enter coefficient c: 9
Enter coefficients for Rainfall (a, b, c):
Enter coefficient a: 5.5
Enter coefficient b: 7.2
Enter coefficient c: 9.5
Enter coefficients for Humidity (a, b, c):
Enter coefficient a: 6.2
Enter coefficient b: 45
Enter coefficient c: 25

Results:
Days: [1, 100, 200, 365]
Temperature: [14.2, 2509.0, 9009.0, 28479.0]
Rainfall: [22.2, 55729.5, 221449.5, 735375.0]
Humidity: [76.2, 66525.0, 257025.0, 842445.0]
```

c) Version 3 (Single set of data): -

```
def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c
def read_coefficients(filename):
    with open(filename, 'r') as file:
        lines = file.readlines()
        temp_coeffs = list(map(float, lines[0].strip().split()))
        rainfall_coeffs = list(map(float, lines[1].strip().split()))
        humidity_coeffs = list(map(float, lines[2].strip().split()))
    return temp_coeffs, rainfall_coeffs, humidity_coeffs
filename = "/content/waterfall data ver 3.txt"
temp_coeffs, rainfall_coeffs, humidity_coeffs = read_coefficients(filename)
days = [1, 100, 200, 365]
temperature = [quadratic_model(*temp_coeffs, day) for day in days]
rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
print("\nResults:")
print(f"Days: {days}")
print(f"Temperature: {temperature}")
print(f"Rainfall: {rainfall}")
print(f"Humidity: {humidity}")
```



Results:
Days: [1, 100, 200, 365]
Temperature: [19.62, 180.0, 740.0, 2538.5]
Rainfall: [7.806, 48.0, 208.0, 734.35]
Humidity: [84.933, 108.0, 191.0, 459.125]

d) Version 4 (Multiple sets of data): -

```
import matplotlib.pyplot as plt

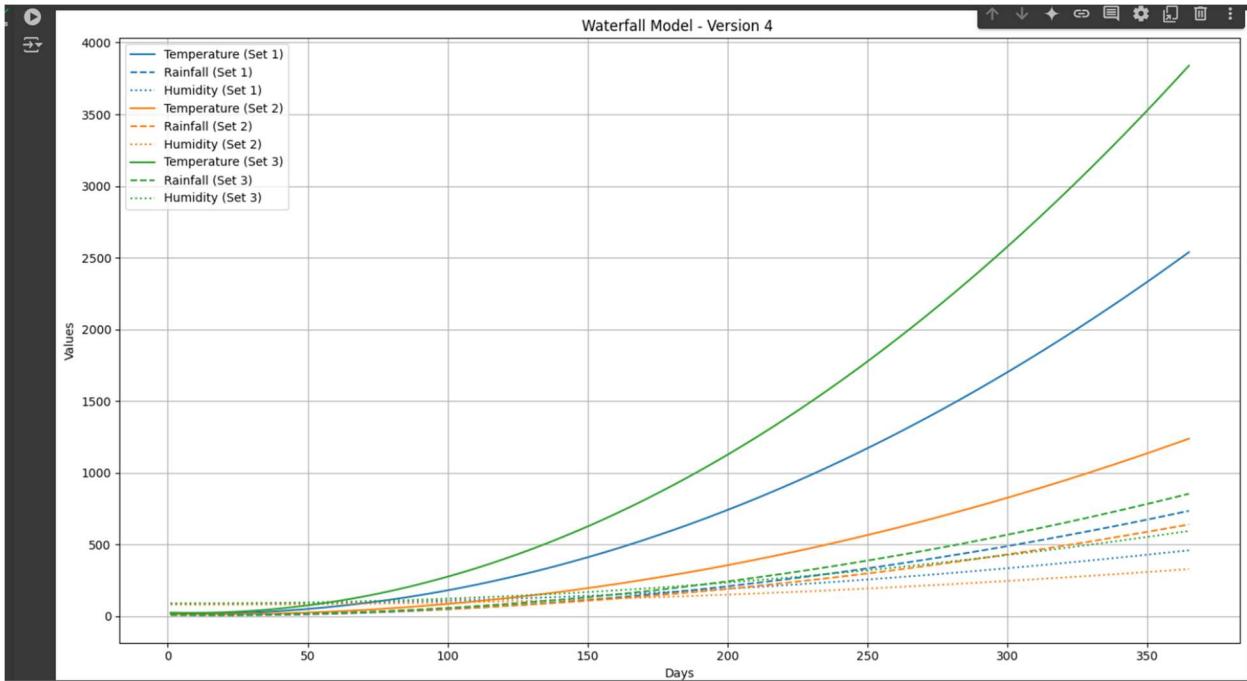
def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c

def read_multiple_coefficients(filename):
    with open(filename, 'r') as file:
        lines = file.readlines()
    coefficients = []
    for line in lines:
        line = line.strip()
        if line and not line.startswith("#"):
            coefficients.append(list(map(float, line.split())))
    return [coefficients[i:i + 3] for i in range(0, len(coefficients), 3)]

filename = "/content/waterfall data ver 4.txt"
coefficients_sets = read_multiple_coefficients(filename)
days = list(range(1, 366))

plt.figure(figsize=(14, 8))
for i, coeffs in enumerate(coefficients_sets):
    temp_coeffs, rainfall_coeffs, humidity_coeffs = coeffs
    temperature = [quadratic_model(*temp_coeffs, day) for day in days]
    rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
    humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
    plt.plot(days, temperature, label=f"Temperature (Set {i + 1})", linestyle='-', color=f"C{i}")
    plt.plot(days, rainfall, label=f"Rainfall (Set {i + 1})", linestyle='--', color=f"C{i}")
    plt.plot(days, humidity, label=f"Humidity (Set {i + 1})", linestyle=':', color=f"C{i}")

plt.title("Waterfall Model - Version 4 ")
plt.xlabel("Days")
plt.ylabel("Values")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```



GitHub Commit History: -

Version 1: -

Filter files... 4/4

1 file changed +24 -0 lines changed

```

+ waterfall_model.py
...
1 + # -*- coding: utf-8 -*-
2 + """Waterfall Ver 1.ipynb
3 +
4 + Automatically generated by Colab.
5 +
6 + Original file is located at
7 +     https://colab.research.google.com/drive/13xdDmz6t_Sb6LYaIM592W4t--lT1G9H2
8 + """
9 +
10 + import numpy as np
11 + temp_coeffs = [0.01, -0.3, 25]
12 + rainfall_coeffs = [0.005, -0.1, 5]
13 + humidity_coeffs = [0.002, -0.05, 80]
14 + days = [1, 100, 200, 365]
15 + def quadratic_model(a, b, c, x):
16 +     return a * x**2 + b * x + c
17 + temperature = [quadratic_model(*temp_coeffs, day) for day in days]
18 + rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
19 + humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
20 + print("Waterfall Model - Version 1 (Hardcoded Variables):")
21 + print(f"Days: {days}")
22 + print(f"Temperature: {temperature}")
23 + print(f"Rainfall: {rainfall}")
24 + print(f"Humidity: {humidity}")

```

Version 2:-

The screenshot shows a code editor interface with a dark theme. A file named 'waterfall_model.py' is open, showing 10 changes made. The code defines a quadratic model and prints hardcoded variable values. The changes are color-coded: red for deletions and green for additions.

```
1 # -*- coding: utf-8 -*-
2 - """Waterfall Ver 1.ipynb
3
4 Automatically generated by Colab.
5
6 Original file is located at
7 https://colab.research.google.com/drive/1Jxd0wzef_5b6lYah592wt--lTtGK2
8 ***
9
10 - import numpy as np
11 - temp_coeffs = [0.01, -0.3, 25]
12 - rainfall_coeffs = [0.005, -0.1, 5]
13 - humidity_coeffs = [0.002, -0.05, 80]
14 - days = [1, 100, 200, 365]
15 def quadratic_model(a, b, c, x):
16     return a * x**2 + b * x + c
17
18 temperature = [quadratic_model(*temp_coeffs, day) for day in days]
19 rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
20 humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
21 print("Waterfall Model - Version 1 (Hardcoded Variables):")
22 print(f"Days: {days}")
23 print(f"Temperature: {temperature}")
24 print(f"Rainfall: {rainfall}")
25 ***
```

```
1 # -*- coding: utf-8 -*-
2 + """Waterfall Ver 2.ipynb
3
4 Automatically generated by Colab.
5
6 Original file is located at
7 + https://colab.research.google.com/drive/1S803QjUS1B4y2Hmz6nNg4Sg4d8e
8 ***
9
10 def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12 + print("Enter coefficients for Temperature (a, b, c):")
13 + temp_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
14 + print("Enter coefficients for Rainfall (a, b, c):")
15 + rainfall_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
16 + print("Enter coefficients for Humidity (a, b, c):")
17 + humidity_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
18 + days = [1, 100, 200, 365]
19 + temperature = [quadratic_model(*temp_coeffs, day) for day in days]
20 + rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
21 + humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
22 + print("\nResults:")
23 print(f"Days: {days}")
24 print(f"Temperature: {temperature}")
25 print(f"Rainfall: {rainfall}")
```

Version 3:-

The screenshot shows a code editor interface with a dark theme. A file named 'waterfall_model.py' is open, showing 11 changes made. The code defines a quadratic model and reads coefficients from a file. The changes are color-coded: red for deletions and green for additions.

```
1 # -*- coding: utf-8 -*-
2 - """Waterfall Ver 2.ipynb
3
4 Automatically generated by Colab.
5
6 Original file is located at
7 - https://colab.research.google.com/drive/1S803QjUS1B4y2Hmz6nNg4Sg4d8e
8 ***
9
10 def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12 - print("Enter coefficients for Temperature (a, b, c):")
13 - temp_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
14 - print("Enter coefficients for Rainfall (a, b, c):")
15 - rainfall_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
16 - print("Enter coefficients for Humidity (a, b, c):")
17 - humidity_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
18
19 days = [1, 100, 200, 365]
20 temperature = [quadratic_model(*temp_coeffs, day) for day in days]
21 rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
```

```
1 # -*- coding: utf-8 -*-
2 + """Waterfall Ver 3.ipynb
3
4 Automatically generated by Colab.
5
6 Original file is located at
7 + https://colab.research.google.com/drive/1iu2JMLiUxqY-kCjPFY85vIxMpfTAk
8 ***
9
10 def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12 + def read_coefficients(filename):
13 +     with open(filename, 'r') as file:
14 +         lines = file.readlines()
15 +         temp_coeffs = list(map(float, lines[0].strip().split()))
16 +         rainfall_coeffs = list(map(float, lines[1].strip().split()))
17 +         humidity_coeffs = list(map(float, lines[2].strip().split()))
18 +     return temp_coeffs, rainfall_coeffs, humidity_coeffs
19 + filename = "/content/waterfall data ver 3.txt"
20 + temp_coeffs, rainfall_coeffs, humidity_coeffs = read_coefficients(filename)
21 days = [1, 100, 200, 365]
22 temperature = [quadratic_model(*temp_coeffs, day) for day in days]
23 rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
```

Data ver 3:-

The screenshot shows a Jupyter Notebook cell with the following content:

```
Code Blame 3 lines (3 loc) • 40 Bytes
```

```
1 0.02 -0.4 20
2 0.006 -0.2 8
3 0.003 -0.07 85
```

Version 4:-

The screenshot shows a GitHub commit interface for a file named `waterfall_model.py`. The commit summary indicates 1 file changed with 34-18 lines changed.

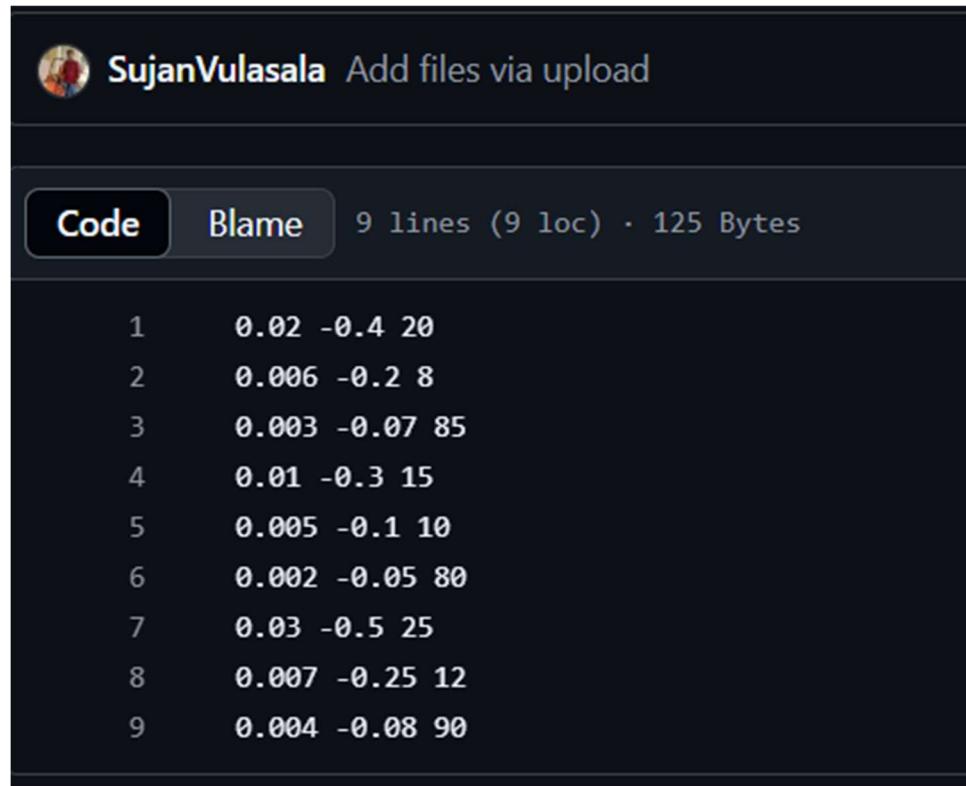
```
waterfall_model.py
```

```
1 file changed +34-18 lines changed
```

```
diff --git a/waterfall_model.py b/waterfall_model.py
--- a/waterfall_model.py
+++ b/waterfall_model.py
@@ -1,12 +1,45 @@
 3
 4 Automatically generated by Colab.
 5
 6 Original file is located at
 7 - https://colab.research.google.com/drive/1iuQ23M11uqV-k3cJFYv5v1xvleptAK
 8 +++
 9
10 + def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12 + def read_coefficients(filename):
13
14     with open(filename, "r") as file:
15         lines = file.readlines()
16         temp_coeffs = list(map(float, lines[0].strip().split()))
17         rainfall_coeffs = list(map(float, lines[1].strip().split()))
18         humidity_coeffs = list(map(float, lines[2].strip().split()))
19
20     return temp_coeffs, rainfall_coeffs, humidity_coeffs
21
22 filename = "/content/waterfall data ver 3.txt"
23 temp_coeffs, rainfall_coeffs, humidity_coeffs = read_coefficients(filename)
24 days = [1, 100, 200, 365]
25 temperature = [quadratic_model(*temp_coeffs, day) for day in days]
26 rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
27 humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
28
29 print("Results:")
30 print(f"Days: {days}")
31 print(f"Temperature: {temperature}")
32 print(f"Rainfall: {rainfall}")
33 print(f"Humidity: {humidity}")

 3
 4 Automatically generated by Colab.
 5
 6 Original file is located at
 7 + https://colab.research.google.com/drive/1gtpf799r-sMzDmz3rdhkp4TFk16Z6b
 8 +++
 9
10 + import matplotlib.pyplot as plt
11 +
12 + def quadratic_model(a, b, c, x):
13     return a * x**2 + b * x + c
14 +
15 + def read_multiple_coefficients(filename):
16     with open(filename, "r") as file:
17         lines = file.readlines()
18         coefficients = []
19         for line in lines:
20             if line and not line.startswith("#"):
21                 coefficients.append(list(map(float, line.split())))
22
23     return [coefficients[i:i+3] for i in range(0, len(coefficients), 3)]
24
25 + filename = "/content/waterfall data ver 4.txt"
26 + coefficients_sets = read_multiple_coefficients(filename)
27 + days = list(range(1, 366))
28 +
29 + plt.figure(figsize=(14, 8))
30 + for i, coeffs in enumerate(coefficients_sets):
31     temp_coeffs, rainfall_coeffs, humidity_coeffs = coeffs
32     temperature = [quadratic_model(*temp_coeffs, day) for day in days]
33     rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
34     humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
35
36     plt.plot(days, temperature, label=f"Temperature (Set {i+1})", linestyle='--', color=f"C{i+1}")
37     plt.plot(days, rainfall, label=f"Rainfall (Set {i+1})", linestyle='-.', color=f"C{i+1}")
38     plt.plot(days, humidity, label=f"Humidity (Set {i+1})", linestyle=':', color=f"C{i+1}")
39
40 + plt.title("Waterfall Model - Version 4")
41 + plt.xlabel("Days")
42 + plt.ylabel("Values")
43 + plt.legend()
44 + plt.grid(True)
45 + plt.tight_layout()
46 + plt.show()
```

Data Ver 4:-



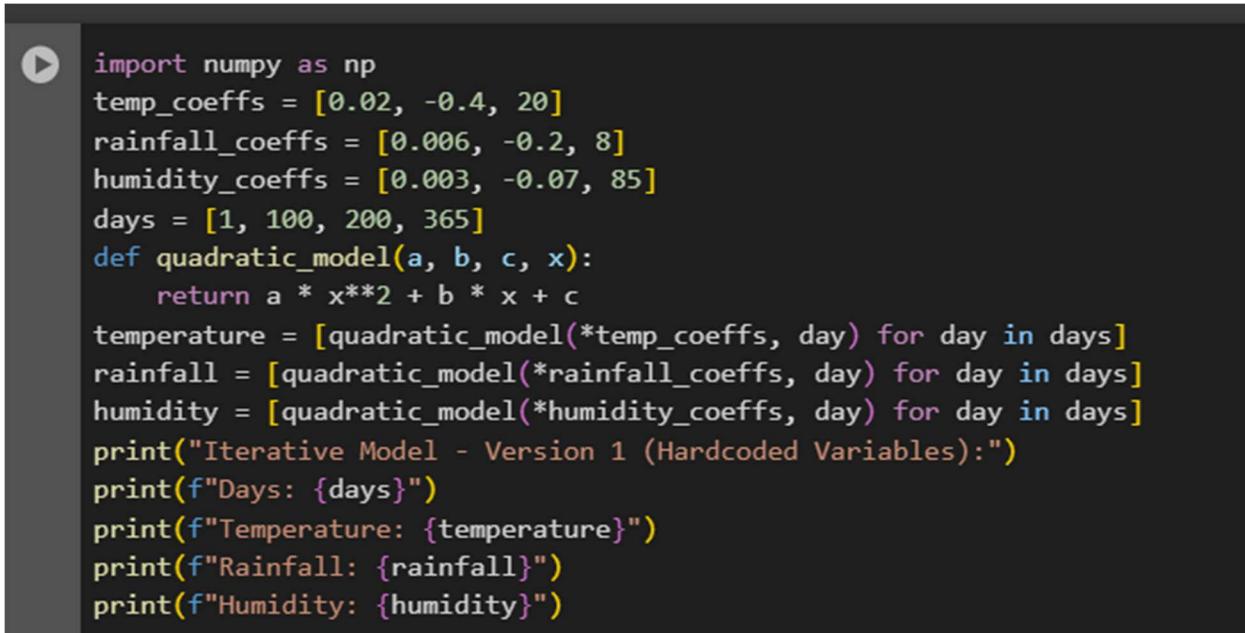
SujanVulasala Add files via upload

Code Blame 9 lines (9 loc) · 125 Bytes

```
1 0.02 -0.4 20
2 0.006 -0.2 8
3 0.003 -0.07 85
4 0.01 -0.3 15
5 0.005 -0.1 10
6 0.002 -0.05 80
7 0.03 -0.5 25
8 0.007 -0.25 12
9 0.004 -0.08 90
```

2)Iterative Model :-

a)Version 1(Hard coding variables):-



```
▶ import numpy as np
temp_coeffs = [0.02, -0.4, 20]
rainfall_coeffs = [0.006, -0.2, 8]
humidity_coeffs = [0.003, -0.07, 85]
days = [1, 100, 200, 365]
def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c
temperature = [quadratic_model(*temp_coeffs, day) for day in days]
rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
print("Iterative Model - Version 1 (Hardcoded Variables):")
print(f"Days: {days}")
print(f"Temperature: {temperature}")
print(f"Rainfall: {rainfall}")
print(f"Humidity: {humidity}")
```

```
→ Iterative Model - Version 1 (Hardcoded Variables):  
Days: [1, 100, 200, 365]  
Temperature: [19.62, 180.0, 740.0, 2538.5]  
Rainfall: [7.806, 48.0, 208.0, 734.35]  
Humidity: [84.933, 108.0, 191.0, 459.125]
```

b) Version 2 (Keyboard Input) :-

```
def quadratic_model(a, b, c, x):  
    return a * x**2 + b * x + c  
print("Enter coefficients for Temperature (a, b, c):")  
temp_coeffs = [float(input(f"Enter coefficient {coef}: ")) for coef in ['a', 'b', 'c']]  
print("Enter coefficients for Rainfall (a, b, c):")  
rainfall_coeffs = [float(input(f"Enter coefficient {coef}: ")) for coef in ['a', 'b', 'c']]  
print("Enter coefficients for Humidity (a, b, c):")  
humidity_coeffs = [float(input(f"Enter coefficient {coef}: ")) for coef in ['a', 'b', 'c']]  
days = [1, 100, 200, 365]  
temperature = [quadratic_model(*temp_coeffs, day) for day in days]  
rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]  
humidity = [quadratic_model(*humidity_coeffs, day) for day in days]  
print("\nIterative Model - Version 2 (Keyboard Input):")  
print(f"Days: {days}")  
print(f"Temperature: {temperature}")  
print(f"Rainfall: {rainfall}")  
print(f"Humidity: {humidity}")
```

```
→ Enter coefficients for Temperature (a, b, c):  
Enter coefficient a: 0.5  
Enter coefficient b: 65  
Enter coefficient c: 65  
Enter coefficients for Rainfall (a, b, c):  
Enter coefficient a: 9.5  
Enter coefficient b: 6  
Enter coefficient c: 45  
Enter coefficients for Humidity (a, b, c):  
Enter coefficient a: 0.9  
Enter coefficient b: 6  
Enter coefficient c: 12
```

```
Iterative Model - Version 2 (Keyboard Input):  
Days: [1, 100, 200, 365]  
Temperature: [130.5, 11565.0, 33065.0, 90402.5]  
Rainfall: [60.5, 95645.0, 381245.0, 1267872.5]  
Humidity: [18.9, 9612.0, 37212.0, 122104.5]
```

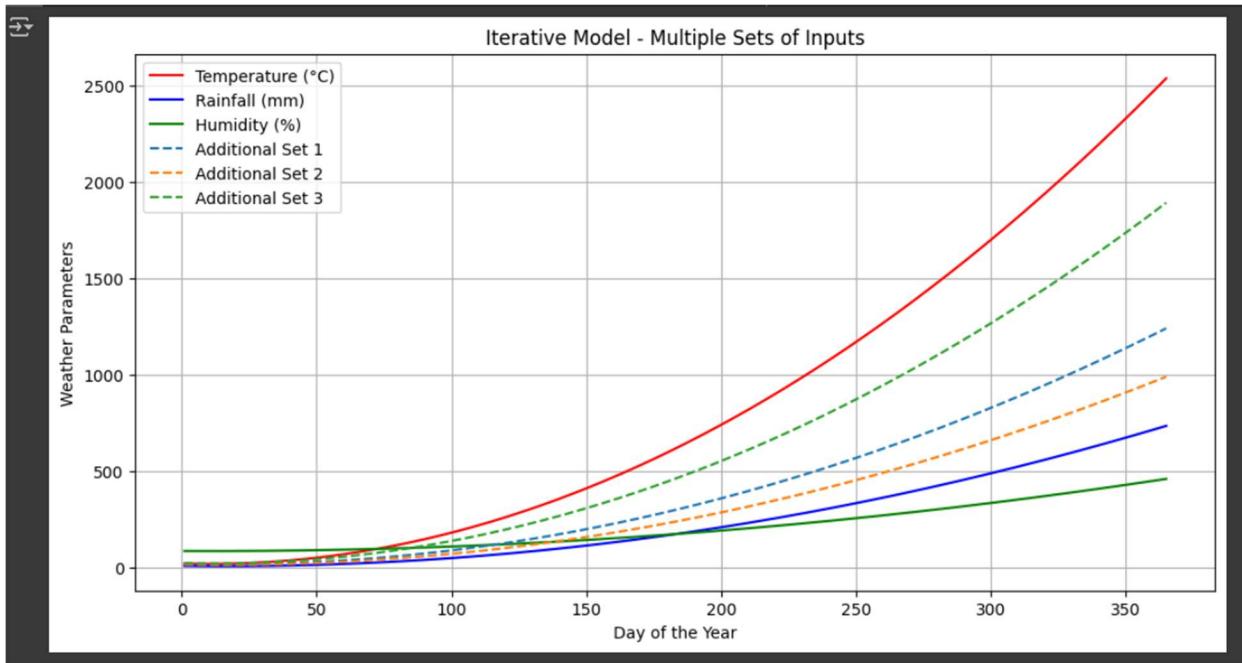
c) Version 3 (Single set of data):-

```
def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c
def read_coefficients_from_file(filename):
    with open(filename, 'r') as file:
        lines = file.readlines()
    coefficients = []
    for line in lines:
        line = line.strip()
        if line and not line.startswith("#"):
            coefficients.append(list(map(float, line.split())))
    return coefficients
filename = "/content/Iterative data ver 3.txt"
coefficients = read_coefficients_from_file(filename)
temp_coeffs, rainfall_coeffs, humidity_coeffs = coefficients
days = [1, 100, 200, 365]
temperature = [quadratic_model(*temp_coeffs, day) for day in days]
rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
print("\nIterative Model - Version 3 :")
print(f"Days: {days}")
print(f"Temperature: {temperature}")
print(f"Rainfall: {rainfall}")
print(f"Humidity: {humidity}")
```

Iterative Model - Version 3 :
Days: [1, 100, 200, 365]
Temperature: [19.62, 180.0, 740.0, 2538.5]
Rainfall: [7.806, 48.0, 208.0, 734.35]
Humidity: [84.933, 108.0, 191.0, 459.125]

d) Version 4 (Multiple sets of data):-

```
▶ import numpy as np
import matplotlib.pyplot as plt
with open("/content/Iterative data ver 4.txt", "r") as file:
    lines = file.readlines()
temp_coeffs = list(map(float, lines[0].strip().split(',')))
rainfall_coeffs = list(map(float, lines[1].strip().split(',')))
humidity_coeffs = list(map(float, lines[2].strip().split(',')))
additional_coeffs = [list(map(float, line.strip().split(','))) for line in lines[3:]]
days = np.linspace(1, 365, 365)
def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c
temperature = quadratic_model(*temp_coeffs, days)
rainfall = quadratic_model(*rainfall_coeffs, days)
humidity = quadratic_model(*humidity_coeffs, days)
plt.figure(figsize=(12, 6))
plt.plot(days, temperature, label="Temperature (°C)", color="red")
plt.plot(days, rainfall, label="Rainfall (mm)", color="blue")
plt.plot(days, humidity, label="Humidity (%)", color="green")
for idx, coeffs in enumerate(additional_coeffs):
    temp = quadratic_model(*coeffs, days)
    plt.plot(days, temp, label=f"Additional Set {idx + 1}", linestyle="--")
plt.title("Iterative Model - Multiple Sets of Inputs")
plt.xlabel("Day of the Year")
plt.ylabel("Weather Parameters")
plt.legend()
plt.grid()
plt.show()
```



GitHub Commit History:-

Version 1:-

1 file changed +24 -0 lines changed

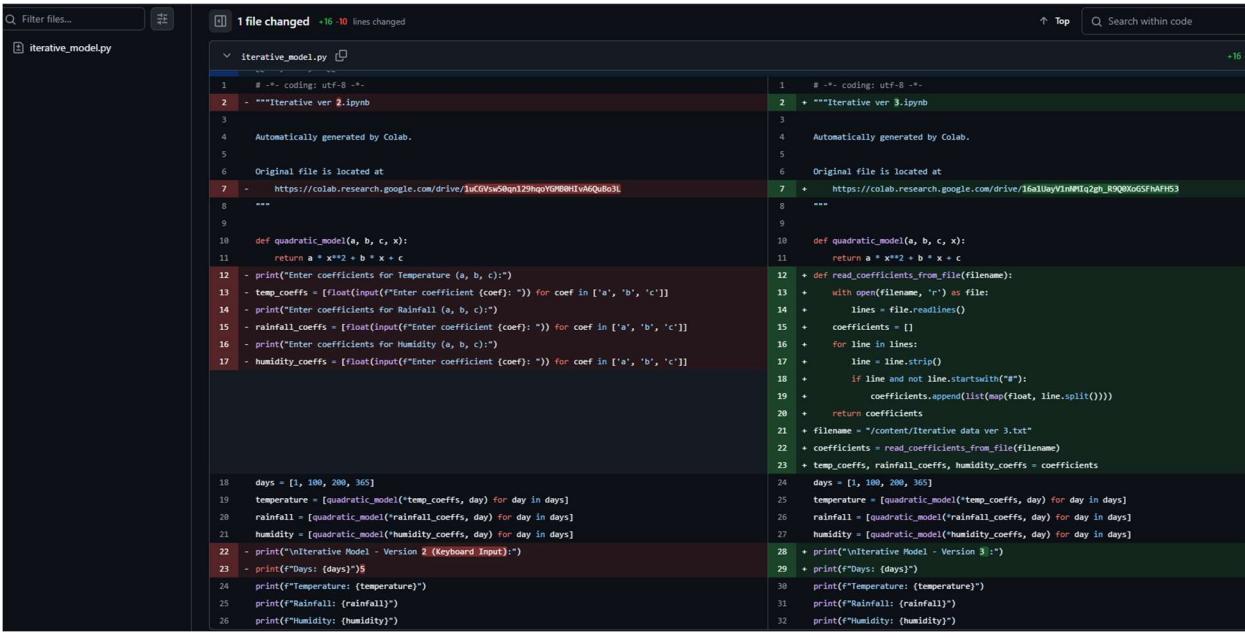
```
iterative_model.py
...
@@ -0,0 +1,24 @@
1 + # -*- coding: utf-8 -*-
2 + """Iterative ver 1.ipynb
3 +
4 + Automatically generated by Colab.
5 +
6 + Original file is located at
7 +     https://colab.research.google.com/drive/1cJ0WDSF4gnO5jL5ZHIAwv8kmAxZs4rR
8 + """
9 +
10 + import numpy as np
11 + temp_coeffs = [0.02, -0.4, 20]
12 + rainfall_coeffs = [0.006, -0.2, 8]
13 + humidity_coeffs = [0.003, -0.07, 85]
14 + days = [1, 100, 200, 365]
15 + def quadratic_model(a, b, c, x):
16 +     return a * x**2 + b * x + c
17 + temperature = [quadratic_model(*temp_coeffs, day) for day in days]
18 + rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
19 + humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
20 + print("Iterative Model - Version 1 (Hardcoded Variables):")
21 + print(f"Days: {days}")
22 + print(f"Temperature: {temperature}")
23 + print(f"Rainfall: {rainfall}")
24 + print(f"Humidity: {humidity}")
```

Version 2:-

1 file changed +11 -9 lines changed

```
iterative_model.py
...
@@ -0,0 +1,24 @@
1 + # -*- coding: utf-8 -*-
2 + """Iterative ver 2.ipynb
3 +
4 + Automatically generated by Colab.
5 +
6 + Original file is located at
7 +     https://colab.research.google.com/drive/1cJ0WDSF4gnO5jL5ZHIAwv8kmAxZs4rR
8 + """
9 +
10 - import numpy as np
11 - temp_coeffs = [0.02, -0.4, 20]
12 - rainfall_coeffs = [0.006, -0.2, 8]
13 - humidity_coeffs = [0.003, -0.07, 85]
14 - days = [1, 100, 200, 365]
15 - def quadratic_model(a, b, c, x):
16 -     return a * x**2 + b * x + c
17 -
18 -     temperature = [quadratic_model(*temp_coeffs, day) for day in days]
19 -     rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
20 -     humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
21 -     print("Iterative Model - Version 1 (Hardcoded Variables):")
22 -     print(f"Days: {days}")
23 -     print(f"Temperature: {temperature}")
24 -     print(f"Rainfall: {rainfall}")
25 -     print(f"Humidity: {humidity}")
26 +     def quadratic_model(a, b, c, x):
27 +         return a * x**2 + b * x + c
28 +         print("Enter coefficients for Temperature (a, b, c):")
29 +         temp_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
30 +         print("Enter coefficients for Rainfall (a, b, c):")
31 +         rainfall_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
32 +         print("Enter coefficients for Humidity (a, b, c):")
33 +         humidity_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
34 +         days = [1, 100, 200, 365]
35 +         temperature = [quadratic_model(*temp_coeffs, day) for day in days]
36 +         rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
37 +         humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
38 +         print("Iterative Model - Version 2 (Keyboard Input):")
39 +         print(f"Days: {days}")
40 +         print(f"Temperature: {temperature}")
41 +         print(f"Rainfall: {rainfall}")
42 +         print(f"Humidity: {humidity}")
```

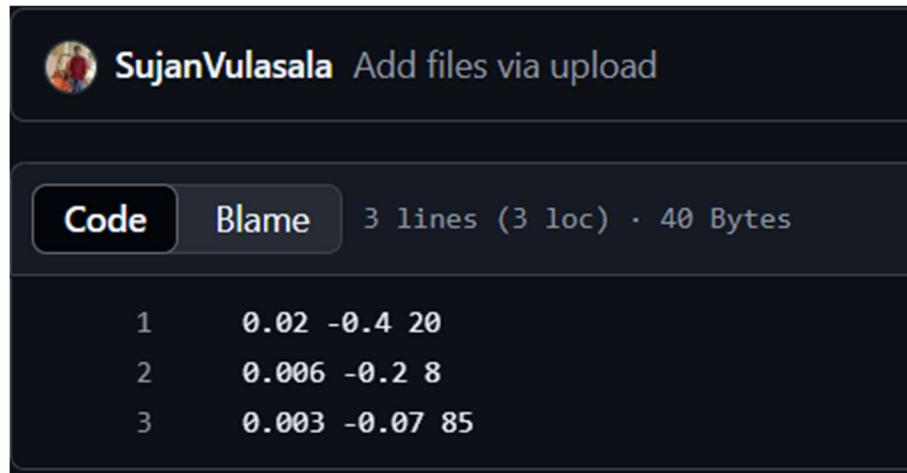
Version 3:-



```
1 # -*- coding: utf-8 -*-
2 # """Iterative ver 2.ipynb
3 
4 Automatically generated by Colab.
5 
6 Original file is located at
7 https://colab.research.google.com/drive/1uGKvsw5Qpn129hpqYGM8RltvA6Qub03l
8 """
9 
10 def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12 - print("Enter coefficients for Temperature (a, b, c):")
13 - temp_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
14 - print("Enter coefficients for Rainfall (a, b, c):")
15 - rainfall_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
16 - print("Enter coefficients for Humidity (a, b, c):")
17 - humidity_coeffs = [float(input("Enter coefficient (coef): ")) for coef in ['a', 'b', 'c']]
18 
19 days = [1, 100, 200, 365]
20 temperature = [quadratic_model(*temp_coeffs, day) for day in days]
21 rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
22 humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
23 - print("Iterative Model - Version 2 (Keyboard Input):")
24 - print(f"Days: {days}")
25 - print(f"Temperature: {temperature}")
26 - print(f"Rainfall: {rainfall}")
27 - print(f"Humidity: {humidity}")
28 + print("Iterative Model - Version 3:")
29 + print(f"Days: {days}")
30 + print(f"Temperature: {temperature}")
31 + print(f"Rainfall: {rainfall}")
32 + print(f"Humidity: {humidity}")

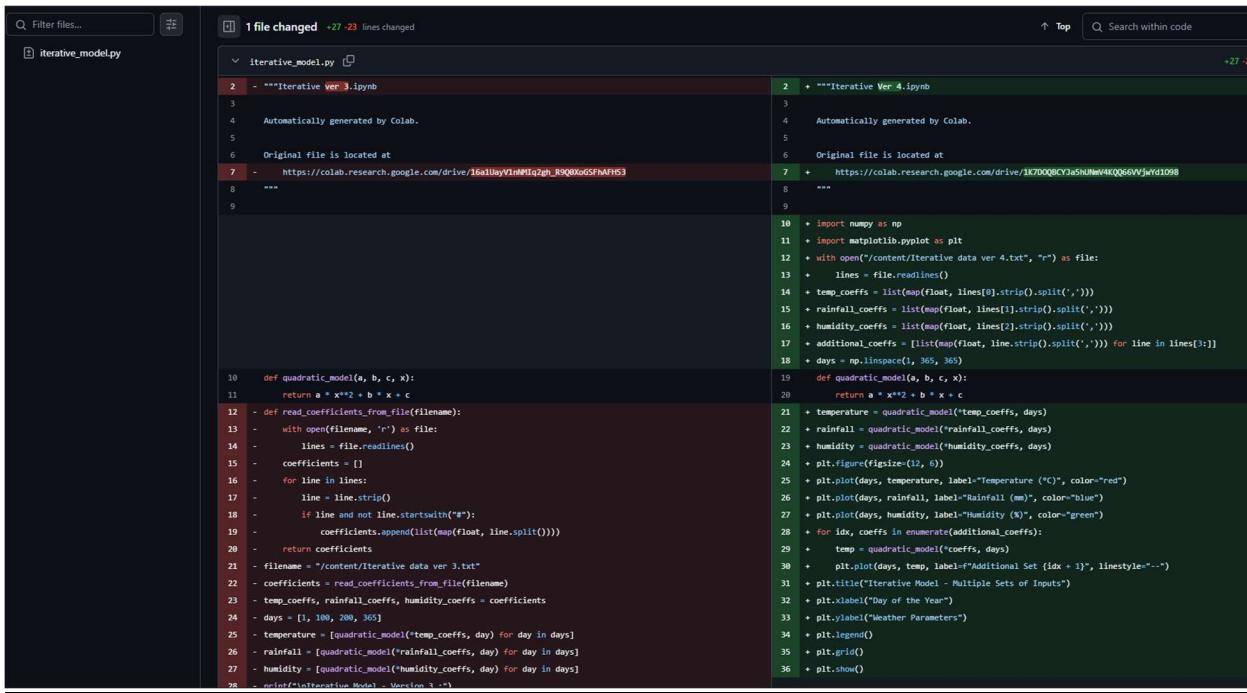
1 # -*- coding: utf-8 -*-
2 + """Iterative ver 3.ipynb
3 
4 Automatically generated by Colab.
5 
6 Original file is located at
7 https://colab.research.google.com/drive/16a1lkyV1nHtq2gh_R9QX0oGShAHHS3
8 """
9 
10 def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12 + def read_coefficients_from_file(filename):
13 +     with open(filename, 'r') as file:
14 +         lines = file.readlines()
15 +         coefficients = []
16 +         for line in lines:
17 +             line = line.strip()
18 +             if line and not line.startswith("#"):
19 +                 coefficients.append(list(map(float, line.split())))
20 + 
21 +     return coefficients
22 + filename = "/content/Iterative data ver 3.txt"
23 + coefficients = read_coefficients_from_file(filename)
24 + temp_coeffs, rainfall_coeffs, humidity_coeffs = coefficients
25 + days = [1, 100, 200, 365]
26 + temperature = [quadratic_model(*temp_coeffs, day) for day in days]
27 + rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
28 + humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
29 + print("Iterative Model - Version 3:")
30 + print(f"Days: {days}")
31 + print(f"Temperature: {temperature}")
32 + print(f"Rainfall: {rainfall}")
33 + print(f"Humidity: {humidity}")
```

Data ver 3:-



Code	Blame	3 lines (3 loc) • 40 Bytes
1	0.02 -0.4 20	
2	0.006 -0.2 8	
3	0.003 -0.07 85	

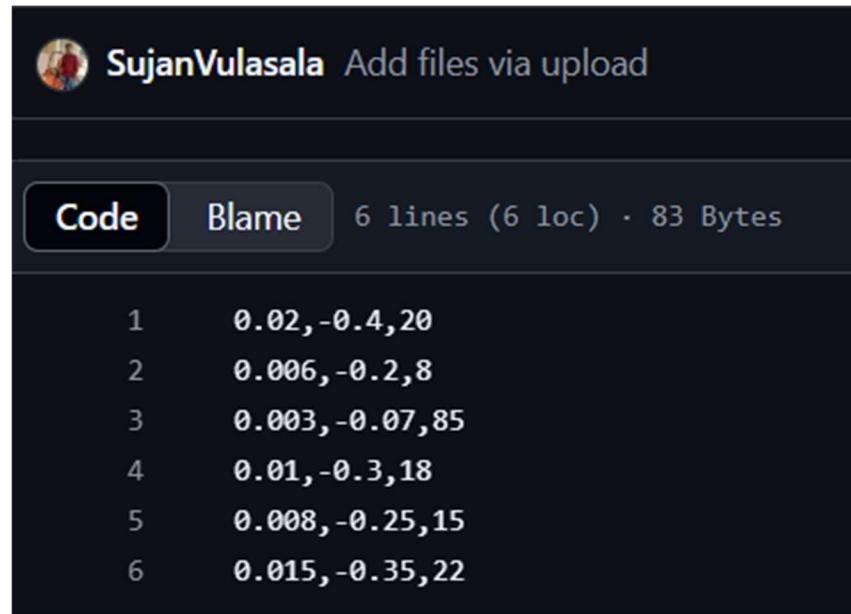
Version 4:-



```
diff --git a/iterative_model.py b/iterative_model.py
--- a/iterative_model.py
+++ b/iterative_model.py
@@ -2,7 +2,7 @@
 2  - """Iterative Ver 3.ipynb
 3
 4  Automatically generated by Colab.
 5
 6  Original file is located at
 7  - https://colab.research.google.com/drive/16a1IuyVlnM1q2gh_R9QXoG5FHF53
 8  ***
 9
10 def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12 - def read_coefficients_from_file(filename):
13 -     with open(filename, 'r') as file:
14 -         lines = file.readlines()
15 -         coefficients = []
16 -         for line in lines:
17 -             line = line.strip()
18 -             if line and not line.startswith("#"):
19 -                 coefficients.append(list(map(float, line.split())))
20 -     return coefficients
21 - filename = "/content/Iterative data ver 3.txt"
22 - coefficients = read_coefficients_from_file(filename)
23 - temp_coeffs, rainfall_coeffs, humidity_coeffs = coefficients
24 - days = [1, 100, 200, 365]
25 - temperature = [quadratic_model(*temp_coeffs, day) for day in days]
26 - rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
27 - humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
28 - print("Iteration Model - Version 3 :")
```

```
2  + """Iterative Ver 4.ipynb
 3
 4  Automatically generated by Colab.
 5
 6  Original file is located at
 7  + https://colab.research.google.com/drive/1K700QCYJaSHUwV4QQ66VjwYd1098
 8  ***
 9
10 + import numpy as np
11 + import matplotlib.pyplot as plt
12 + with open("/content/Iterative data ver 4.txt", "r") as file:
13 +     lines = file.readlines()
14 +     temp_coeffs = list(map(float, lines[0].strip().split(' ')))
15 +     rainfall_coeffs = list(map(float, lines[1].strip().split(',')))
16 +     humidity_coeffs = list(map(float, lines[2].strip().split(',')))
17 +     additional_coeffs = [list(map(float, line.strip().split(','))) for line in lines[3:]]
18 +     days = np.linspace(1, 365, 365)
19 + def quadratic_model(a, b, c):
20 +     return a * x**2 + b * x + c
21 + temperature = quadratic_model(*temp_coeffs, days)
22 + rainfall = quadratic_model(*rainfall_coeffs, days)
23 + humidity = quadratic_model(*humidity_coeffs, days)
24 + plt.figure(figsize=(12, 6))
25 + plt.plot(days, temperature, label="Temperature (°C)", color="red")
26 + plt.plot(days, rainfall, label="Rainfall (mm)", color="blue")
27 + plt.plot(days, humidity, label="Humidity (%)", color="green")
28 + for idx, coeffs in enumerate(additional_coeffs):
29 +     temp = quadratic_model(*coeffs, days)
30 +     plt.plot(days, temp, label=f"Additional Set {idx+1}", linestyle="--")
31 + plt.title("Iterative Model - Multiple Sets of Inputs")
32 + plt.xlabel("Day of the Year")
33 + plt.ylabel("Weather Parameters")
34 + plt.legend()
35 + plt.grid()
36 + plt.show()
```

Data Ver 4:-



3)Agile Method :-

a)Version 1(Hard coding variables):-

```
def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c
temperature_coeffs = [0.02, -0.4, 20]
rainfall_coeffs = [0.006, -0.2, 8]
humidity_coeffs = [0.003, -0.07, 85]
days = [1, 100, 200, 365]
temperature = [quadratic_model(*temperature_coeffs, day) for day in days]
rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
print("Day\tTemperature\tRainfall\tHumidity")
for day, temp, rain, hum in zip(days, temperature, rainfall, humidity):
    print(f"{day}\t{temp:.2f}\t{rain:.2f}\t{hum:.2f}")
```

Day	Temperature	Rainfall	Humidity
1	19.62	7.81	84.93
100	180.00	48.00	108.00
200	740.00	208.00	191.00
365	2538.50	734.35	459.12

b) Version 2 (Keyboard Input) :-

```
def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c

temperature_coeffs = [float(input("Enter coefficient a for Temperature: ")),
                      float(input("Enter coefficient b for Temperature: ")),
                      float(input("Enter coefficient c for Temperature: "))]

rainfall_coeffs = [float(input("Enter coefficient a for Rainfall: ")),
                   float(input("Enter coefficient b for Rainfall: ")),
                   float(input("Enter coefficient c for Rainfall: "))]

humidity_coeffs = [float(input("Enter coefficient a for Humidity: ")),
                   float(input("Enter coefficient b for Humidity: ")),
                   float(input("Enter coefficient c for Humidity: "))]

days = [1, 100, 200, 365]

temperature = [quadratic_model(*temperature_coeffs, day) for day in days]
rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
humidity = [quadratic_model(*humidity_coeffs, day) for day in days]

print("Day\tTemperature\tRainfall\tHumidity")
for day, temp, rain, hum in zip(days, temperature, rainfall, humidity):
    print(f"{day}\t{temp:.2f}\t{rain:.2f}\t{hum:.2f}")
```

```
→ Enter coefficient a for Temperature: 0.05
Enter coefficient b for Temperature: 7
Enter coefficient c for Temperature: 6
Enter coefficient a for Rainfall: 5.2
Enter coefficient b for Rainfall: 6.8
Enter coefficient c for Rainfall: 9.2
Enter coefficient a for Humidity: 5.6
Enter coefficient b for Humidity: 4.0
Enter coefficient c for Humidity: 7.9
Day      Temperature      Rainfall      Humidity
1        13.05          21.20          17.50
100      1206.00         52689.20        56407.90
200      3406.00         209369.20       224807.90
365      9222.25         695261.20       747527.90
```

c) Version 3 (Single set of data):-

```
def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c

with open("/content/Agile data ver 3.txt", "r") as file:
    coefficients = file.readlines()

temperature_coeffs = list(map(float, coefficients[0].strip().split()))
rainfall_coeffs = list(map(float, coefficients[1].strip().split()))
humidity_coeffs = list(map(float, coefficients[2].strip().split()))

days = [1, 100, 200, 365]

temperature = [quadratic_model(*temperature_coeffs, day) for day in days]
rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
humidity = [quadratic_model(*humidity_coeffs, day) for day in days]

print("Day\tTemperature\tRainfall\tHumidity")
for day, temp, rain, hum in zip(days, temperature, rainfall, humidity):
    print(f"{day}\t{temp:.2f}\t{rain:.2f}\t{hum:.2f}")
```

Day	Temperature	Rainfall	Humidity
1	19.62	7.81	84.93
100	180.00	48.00	108.00
200	740.00	208.00	191.00
365	2538.50	734.35	459.12

d)Version 4 (Multiple sets of data):-

```
import matplotlib.pyplot as plt

def quadratic_model(a, b, c, x):
    return a * x**2 + b * x + c

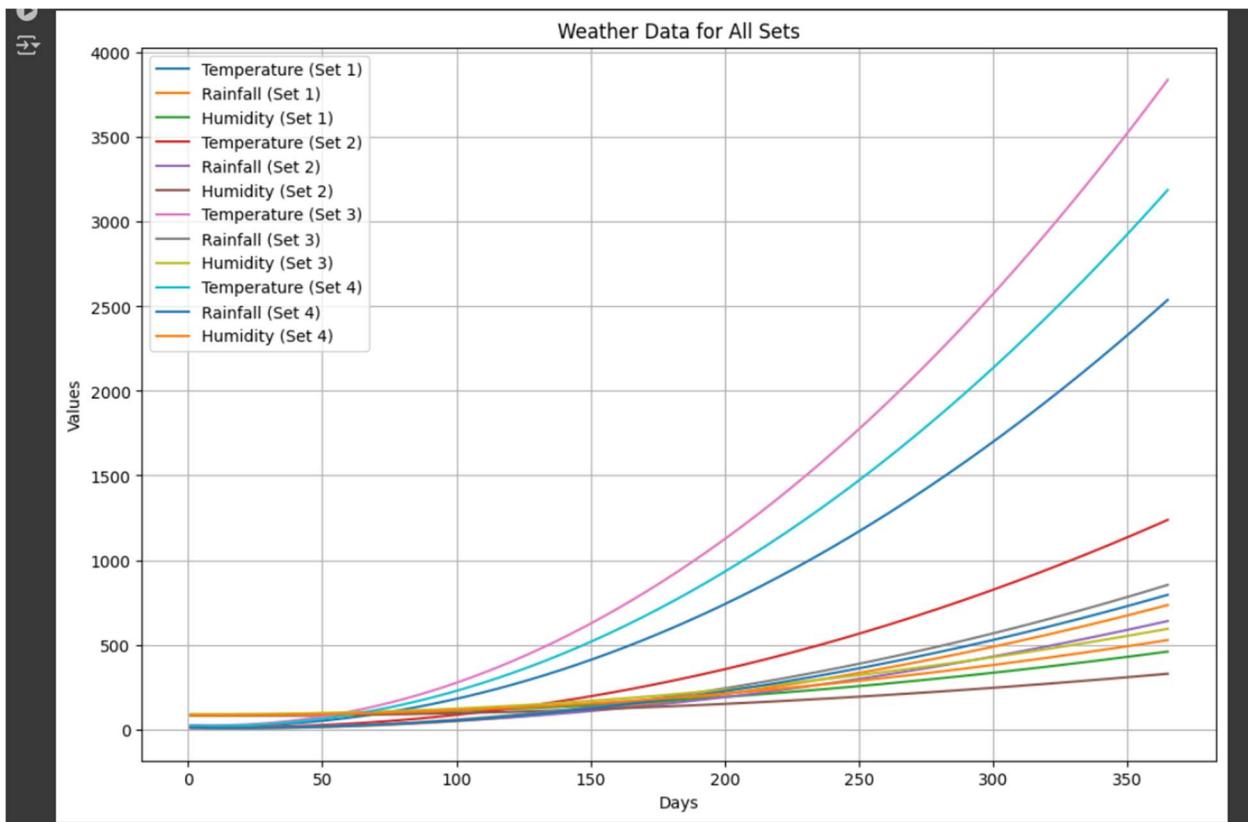
with open("/content/Agile data ver 4.txt", "r") as file:
    coefficient_sets = [list(map(float, line.strip().split())) for line in file]

days = list(range(1, 366))

plt.figure(figsize=(12, 8))
for idx, coeffs in enumerate(coefficient_sets):
    temp_coeffs, rainfall_coeffs, humidity_coeffs = coeffs[:3], coeffs[3:6], coeffs[6:9]
    temperature = [quadratic_model(*temp_coeffs, day) for day in days]
    rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
    humidity = [quadratic_model(*humidity_coeffs, day) for day in days]

    plt.plot(days, temperature, label=f"Temperature (Set {idx + 1})")
    plt.plot(days, rainfall, label=f"Rainfall (Set {idx + 1})")
    plt.plot(days, humidity, label=f"Humidity (Set {idx + 1})")

plt.xlabel("Days")
plt.ylabel("Values")
plt.title("Weather Data for All Sets")
plt.legend()
plt.grid()
plt.show()
```



GitHub Commit History:-

Version 1:-

Filter files... Filter

Agile_model.py

1 file changed +21 -0 lines changed

```

Agile_model.py
...
@@ -0,0 +1,21 @@
+ # -*- coding: utf-8 -*-
+ """Agile ver 1.ipynb
+
+ Automatically generated by Colab.
+
+ Original file is located at
+ https://colab.research.google.com/drive/1MBCSrVnDDesy2T4V2qUsOy8AzmEkyigp
+
+
+ def quadratic_model(a, b, c, x):
+     return a * x**2 + b * x + c
+ temperature_coeffs = [0.02, -0.4, 28]
+ rainfall_coeffs = [0.006, -0.2, 8]
+ humidity_coeffs = [0.003, -0.07, 85]
+ days = [1, 100, 200, 365]
+ temperature = [quadratic_model(*temperature_coeffs, day) for day in days]
+ rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
+ humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
+ print("Day\tTemperature\tRainfall\tHumidity")
+ for day, temp, rain, hum in zip(days, temperature, rainfall, humidity):
+     print(f"{day}\t{temp:.2f}\t{rain:.2f}\t{hum:.2f}")

```

Version 2:-

The screenshot shows a code diff interface comparing two files: Agile_model.py and Agile_ver_2.ipynb. The interface highlights changes in three colors: red for deleted code, green for added code, and blue for modified code. The code itself is a quadratic model for temperature, rainfall, and humidity.

```
diff --git Agile_model.py Agile_ver_2.ipynb
--- Agile_model.py
+++ Agile_ver_2.ipynb
@@ -1,21 +1,33 @@
1 # -*- coding: utf-8 -*-
2 - """Agile ver 1.ipynb
3
4 Automatically generated by Colab.
5
6 Original file is located at
7 - https://colab.research.google.com/drive/1W6CSrVn00esy2I4YzqUsy8Azmkyip
8 ***
9
10 def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12 - temperature_coeffs = [0.02, -0.4, 20]
13 - rainfall_coeffs = [0.006, -0.2, 8]
14 - humidity_coeffs = [-0.003, -0.07, 85]
15
16 days = [1, 100, 200, 365]
17
18 temperature = [quadratic_model(*temperature_coeffs, day) for day in days]
19 rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
20 humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
21
22 print("Day\Temperature\Rainfall\Humidity")
23 for day, temp, rain, hum in zip(days, temperature, rainfall, humidity):
24     print(f"{day}\t{temp:.2f}\t{rain:.2f}\t{hum:.2f}")
25
26
27 temperature = [quadratic_model(*temperature_coeffs, day) for day in days]
28 rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
29 humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
30
31 print("Day\Temperature\Rainfall\Humidity")
32 for day, temp, rain, hum in zip(days, temperature, rainfall, humidity):
33     print(f"{day}\t{temp:.2f}\t{rain:.2f}\t{hum:.2f}")


```

Version 3:-

The screenshot shows a code diff interface comparing two files: Agile_model.py and Agile_ver_3.ipynb. The interface highlights changes in three colors: red for deleted code, green for added code, and blue for modified code. The code includes a section for reading coefficients from a text file.

```
diff --git Agile_model.py Agile_ver_3.ipynb
--- Agile_model.py
+++ Agile_ver_3.ipynb
@@ -1,26 +1,21 @@
1 # -*- coding: utf-8 -*-
2 - """Agile ver 2.ipynb
3
4 Automatically generated by Colab.
5
6 Original file is located at
7 - https://colab.research.google.com/drive/1xWhZc-3uvmp3lM8D-nq_el0nThdak-8
8 ***
9
10 def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12
13 - temperature_coeffs = [float(input("Enter coefficient a for Temperature: ")),
14 -                         float(input("Enter coefficient b for Temperature: ")),
15 -                         float(input("Enter coefficient c for Temperature: "))]
16
17 - rainfall_coeffs = [float(input("Enter coefficient a for Rainfall: ")),
18 -                     float(input("Enter coefficient b for Rainfall: ")),
19 -                     float(input("Enter coefficient c for Rainfall: "))]
20
21 - humidity_coeffs = [float(input("Enter coefficient a for Humidity: ")),
22 -                      float(input("Enter coefficient b for Humidity: ")),
23 -                      float(input("Enter coefficient c for Humidity: "))]
24
25 days = [1, 100, 200, 365]
26
27
28 temperature = [quadratic_model(*temperature_coeffs, day) for day in days]
29 rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
30 humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
31
32 print("Day\Temperature\Rainfall\Humidity")
33 for day, temp, rain, hum in zip(days, temperature, rainfall, humidity):
34     print(f"{day}\t{temp:.2f}\t{rain:.2f}\t{hum:.2f}")


```

Data ver 3:-

The screenshot shows a GitHub commit interface. At the top, it displays the user's profile picture and name, "SujanVulasala", followed by the text "Add files via upload". Below this, there are two tabs: "Code" and "Blame". The "Code" tab is selected, showing the following three lines of code:

```
1  0.02 -0.4 20
2  0.006 -0.2 8
3  0.003 -0.07 85
```

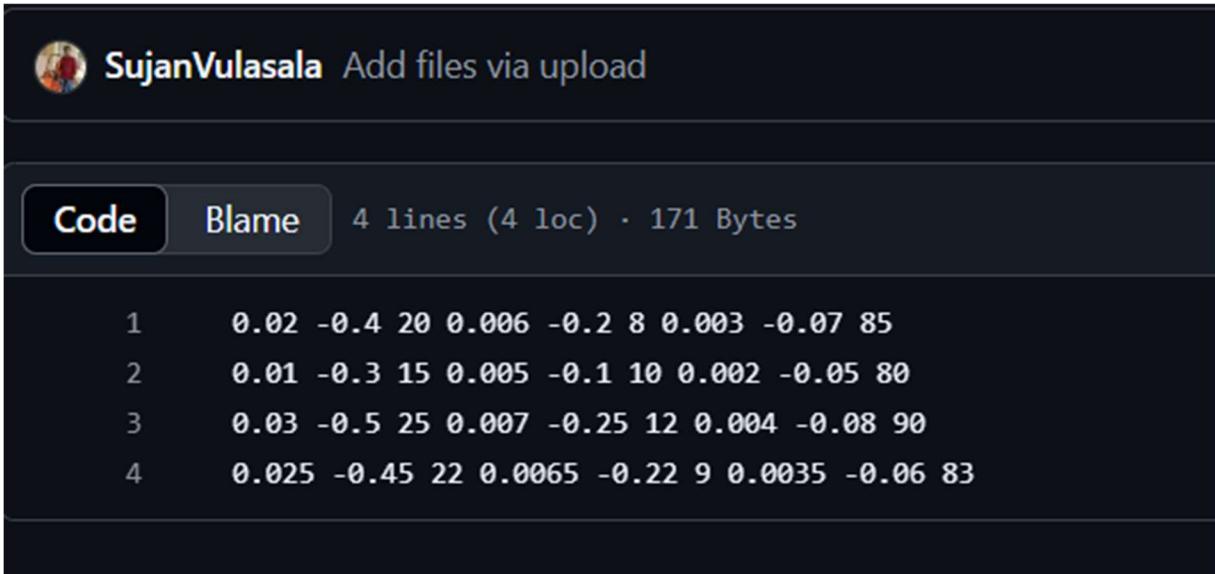
Next to the code, the "Blame" tab is visible, along with the statistics "3 lines (3 loc) · 40 Bytes".

Version 4:-

The screenshot shows a GitHub diff interface comparing two versions of the file "Agile_model.py". The left side shows the content of "Agile_model.py" from version 3, and the right side shows the content from version 4. The diff highlights the changes made between the two versions.

```
diff --git a/Agile_model.py b/Agile_model.py
--- a/Agile_model.py
+++ b/Agile_model.py
@@ -1,14 +1,14 @@
 1 # -*- coding: utf-8 -*-
 2 - """Agile ver 3.ipynb
 3 
 4 Automatically generated by Colab.
 5 
 6 Original file is located at
 7 - https://colab.research.google.com/drive/1Y0s7zHs76K6-EY7JPegeMGR150KFZ2
 8 ===
 9 
10 def quadratic_model(a, b, c, x):
11     return a * x**2 + b * x + c
12 
13 - with open("/content/Agile data ver 3.txt", "r") as file:
14 -     coefficients = file.readlines()
15 
16 -     temperature_coeffs = list(map(float, coefficients[0].strip().split()))
17 -     rainfall_coeffs = list(map(float, coefficients[1].strip().split()))
18 -     humidity_coeffs = list(map(float, coefficients[2].strip().split()))
19 
20 - days = [1, 100, 200, 365]
21 
22 - temperature = [quadratic_model(*temperature_coeffs, day) for day in days]
23 - rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
24 - humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
25 
26 - print("Day\Temperature\tRainfall\tHumidity")
27 - for day, temp, rain, hum in zip(days, temperature, rainfall, humidity):
28 -     print(f"{day}\t{temp:.2f}\t{rain:.2f}\t{hum:.2f}")
29 
30 + import matplotlib.pyplot as plt
31 + 
32 + def quadratic_model(a, b, c, x):
33 +     return a * x**2 + b * x + c
34 + 
35 + with open("/content/Agile data ver 4.txt", "r") as file:
36 +     coefficient_sets = [list(map(float, line.strip().split())) for line in file]
37 + 
38 +     days = list(range(1, 366))
39 
40 + plt.figure(figsize=(12, 8))
41 + for idx, coeffs in enumerate(coefficient_sets):
42 +     temp_coeffs, rainfall_coeffs, humidity_coeffs = coeffs[:3], coeffs[3:6], coeffs[6:9]
43 +     temperature = [quadratic_model(*temp_coeffs, day) for day in days]
44 +     rainfall = [quadratic_model(*rainfall_coeffs, day) for day in days]
45 +     humidity = [quadratic_model(*humidity_coeffs, day) for day in days]
46 
47 +     plt.plot(days, temperature, label=f"Temperature (Set {idx + 1})")
48 +     plt.plot(days, rainfall, label=f"Rainfall (Set {idx + 1})")
49 +     plt.plot(days, humidity, label=f"Humidity (Set {idx + 1})")
50 
51 + plt.xlabel("Days")
52 + plt.ylabel("Values")
53 + plt.title("Weather Data for All Sets")
54 + plt.legend()
```

Data Ver 4:-



The screenshot shows a GitHub commit interface. At the top, it displays the user profile of SujanVulasala and the option to "Add files via upload". Below this, there are two tabs: "Code" (which is selected) and "Blame". The "Code" tab indicates there are 4 lines (4 loc) and 171 Bytes. The code itself is listed below:

```
1  0.02 -0.4 20 0.006 -0.2 8 0.003 -0.07 85
2  0.01 -0.3 15 0.005 -0.1 10 0.002 -0.05 80
3  0.03 -0.5 25 0.007 -0.25 12 0.004 -0.08 90
4  0.025 -0.45 22 0.0065 -0.22 9 0.0035 -0.06 83
```

Result :-

The weather modelling system was successfully implemented using the quadratic solution. Each software development process approach (Waterfall, Iterative, Agile) was applied to demonstrate its implementation, testing, and delivery stages. All versions were saved on GitHub for version control, ensuring traceability and collaboration.