10/5/25, 11:02 AM fitbitanalysis

```
In [ ]: import numpy as np
         # 1. Load Dataset
         # Example data (replace with your own CSV load logic)
         # Columns: Date | Steps | Mood | Calories | Sleep | Activity
         data = np.array([
             ["2017-11-01", "3450", "Happy", "200", "6", "Active"],
             ["2017-11-02", "1200", "Sad", "90", "5", "Inactive"],
["2017-11-03", "5600", "Happy", "180", "7", "Active"],
             ["2017-11-04", "2500", "Neutral", "120", "6", "Inactive"],
             ["2017-11-05", "7422", "Happy", "250", "4", "Active"],
             ["2017-11-06", "3000", "Sad", "100", "5", "Inactive"],
             ["2017-11-07", "6700", "Neutral", "210", "8", "Active"],
             ["2017-11-08", "1500", "Sad", "95", "5", "Inactive"],
             ["2017-11-09", "5000", "Happy", "160", "6", "Active"],
["2017-11-10", "2800", "Sad", "110", "5", "Inactive"],
             ["2017-12-12", "6000", "Happy", "300", "7", "Active"],
             ["2017-12-13", "7000", "Happy", "310", "6", "Active"],
             ["2017-12-14", "6500", "Neutral", "290", "6", "Active"]
         ])
         # Extract columns
         date = data[:, 0]
         step_count = data[:, 1].astype(int)
         mood = data[:, 2]
         calories = data[:, 3].astype(int)
         sleep = data[:, 4].astype(int)
         activity = data[:, 5]
         # Masks
         active_mask = (activity == "Active")
         inactive_mask = (activity == "Inactive")
         # 2. Key Analyses
         # -------
         print("\n | FITBIT DATA ANALYSIS RESULTS\n")
         # 1. Average Steps on Active Days
         avg steps active = np.mean(step count[active mask])
         print(f" ✓ Avg Steps on Active Days: {avg_steps_active:.0f}")
         # 2. High-Effort Days (>5000 steps & >150 cal)
         high_effort_days = np.sum((step_count > 5000) & (calories > 150))
         print(f" ✓ High-Effort Days: {high effort days}")
         # 3. Percentage of 'Sad' Mood Days
         sad_pct = np.sum(mood == "Sad") / len(mood) * 100
         print(f"  % Sad Mood Days: {sad_pct:.2f}%")
         # 4. Max Steps with <6 Hours Sleep
         max_steps_low_sleep = np.max(step_count[sleep < 6])</pre>
         print(f" ✓ Max Steps (<6 hrs sleep): {max_steps_low_sleep}")</pre>
         # 5. Avg Calories per Hour (Inactive Days)
```

10/5/25, 11:02 AM fitbitanalysis

```
avg_calories_inactive = np.mean(calories[inactive_mask] / sleep[inactive_mask])
print(f" ✓ Avg Calories/Hour (Inactive): {avg calories inactive:.2f}")
# 6. Mood with Highest Avg Step Count
mood avg steps = {m: np.mean(step count[mood == m]) for m in np.unique(mood)}
max mood = max(mood avg steps, key=mood avg steps.get)
print(f" ✓ Mood with Highest Avg Steps: {max mood} ({mood avg steps[max mood]:.(
# 7. Steps vs Calories Correlation
corr_steps_calories = np.corrcoef(step_count, calories)[0, 1]
print(f" ✓ Correlation (Steps vs Calories): {corr steps calories:.2f}")
# 8. Longest Inactive Low-Calorie Streak (< mean cal)
mean_cal = np.mean(calories)
low cal inactive = (activity == "Inactive") & (calories < mean cal)</pre>
max streak, current streak = 0, 0
for val in low_cal_inactive:
   if val:
       current_streak += 1
       max_streak = max(max_streak, current_streak)
       current_streak = 0
print(f" ✓ Longest Inactive Low-Calorie Streak: {max streak} days")
# 9. Day with Highest Calories-to-Sleep Ratio
cal_sleep_ratio = calories / sleep
best_day = date[np.argmax(cal_sleep_ratio)]
# 10. Top 3-Day Burn Period (Max Calorie Sum)
window_sums = np.array([np.sum(calories[i:i+3]) for i in range(len(calories)-2)]
start_idx = np.argmax(window_sums)
print(f" ▼ Top 3-Day Burn Period Start: {date[start_idx]} (Total: {window_sums[:
# ------
# 3. Summary Insights
# -----
print("\n KEY TAKEAWAYS:")
print(f"  Highest Avg Steps Mood: {max_mood}")
print(f" Sad Mood Days: ~{sad pct:.1f}%")
print(f"① Most Frequent Sleep Duration: {np.bincount(sleep).argmax()} hrs")
print(f"  Max Steps on Low Sleep: {max_steps_low_sleep}")
print(f" Sleep vs Calories Correlation: {np.corrcoef(sleep, calories)[0,1]:.2
print(f"  Longest Inactive Low-Calorie Streak: {max_streak} days")
print(f" Top Calories-to-Sleep Efficiency Day: {best day}")
print(f" Max 3-Day Calorie Burn Starts: {date[start idx]}")
```

10/5/25, 11:02 AM fitbitanalysis

- **II** FITBIT DATA ANALYSIS RESULTS
- Avg Steps on Active Days: 5959
- ☑ High-Effort Days: 6
- ✓ % Sad Mood Days: 30.77%
- ✓ Max Steps (<6 hrs sleep): 7422</p>
- Avg Calories/Hour (Inactive): 19.80
- Mood with Highest Avg Steps: Happy (5745 steps)
- ☑ Correlation (Steps vs Calories): 0.88
- Longest Inactive Low-Calorie Streak: 1 days
- ☑ Day with Highest Calorie-to-Sleep Ratio: 2017-11-05
- ▼ Top 3-Day Burn Period Start: 2017-12-12 (Total: 900 cal)
- KEY TAKEAWAYS:
- Tighest Avg Steps Mood: Happy
- Sad Mood Days: ~30.8%
- Most Frequent Sleep Duration: 6 hrs
- Max Steps on Low Sleep: 7422
- Sleep vs Calories Correlation: 0.36
- 🛅 Longest Inactive Low-Calorie Streak: 1 days
- Top Calories-to-Sleep Efficiency Day: 2017-11-05
- max 3-Day Calorie Burn Starts: 2017-12-12