

# Monthly Household Electricity Analysis: Appliance-Wise Usage, Trends & Bill Reduction Recommendations

## Objectives:

- Estimate each appliances energy usage using assumed wattages and usage time.
- Compare total estimated kWh with actual units consumed.
- Identify high-consuming appliances.
- Visualize monthly trends and seasonal peaks.
- Provide data-driven recommendations to reduce usage and save money.

```
1 import pandas as pd
2 df = pd.read_csv('/content/electricity.csv')
3 df.head()
```

	Month	Units_Used	Bill_Amount	People	Fan_Hours	AC_Hours	Fridge_Watt	Washing_Hours	Cooking_Mins
0	Jan 2024	200	878	4	1079	0	160	4	92
1	Feb 2024	201	757	4	1070	0	160	6	108
2	Mar 2024	385	1747	4	1359	109	160	4	91
3	Apr 2024	583	2243	4	1744	221	160	6	85
4	May 2024	670	3122	4	1478	292	160	6	107

Next steps: [Generate code with df](#) [View recommended plots](#) [New interactive sheet](#)

```
1 df["Month"] = pd.to_datetime(df["Month"], format="%b %Y", errors='coerce')
2 df["Cooking_Hours"] = df["Cooking_Mins"] / 60
3 df = df.drop(columns=["Comments"]) #not need for analysis
4 df.head()
```

	Month	Units_Used	Bill_Amount	People	Fan_Hours	AC_Hours	Fridge_Watt	Washing_Hours	Cooking_Mins
0	2024-01-01	200	878	4	1079	0	160	4	92
1	2024-02-01	201	757	4	1070	0	160	6	108
2	2024-03-01	385	1747	4	1359	109	160	4	91
3	2024-04-01	583	2243	4	1744	221	160	6	85
4	2024-05-01	670	3122	4	1478	292	160	6	107

Next steps: [Generate code with df](#) [View recommended plots](#) [New interactive sheet](#)

```

1 df.columns.tolist()

2 ['Month',
3  'Units_Used',
4  'Bill_Amount',
5  'People',
6  'Fan_Hours',
7  'AC_Hours',
8  'Fridge_Watt',
9  'Washing_Hours',
10 'Cooking_Mins',
11 'Cooking_Hours']

12
13
14 # Estimate Appliance-wise Energy Usage (in kWh)
15
16 def watt_to_kwh(watt, hours_per_day, days=30):
17     return (watt * hours_per_day * days) / 1000
18
19 # Assumed wattages
20 WATTAGE = {
21     "AC": 1500,
22     "Fan": 75,
23     "Washing": 500,
24     "Cooking": 1000,
25 }
26
27 df["Fridge_kWh"] = watt_to_kwh(df["Fridge_Watt"], 24 * 0.4)
28 df["AC_kWh"] = watt_to_kwh(WATTAGE["AC"], df["AC_Hours"])
29 df["Fan_kWh"] = watt_to_kwh(WATTAGE["Fan"], df["Fan_Hours"])
30 df["Washing_kWh"] = watt_to_kwh(WATTAGE["Washing"], df["Washing_Hours"])
31 df["Cooking_kWh"] = watt_to_kwh(WATTAGE["Cooking"], df["Cooking_Hours"])
32
33 # Total estimated usage vs reported usage
34 df["Estimated_kWh"] = df[["Fridge_kWh", "AC_kWh", "Fan_kWh", "Washing_kWh", "Cooking_kWh"]].sum(axis=
35 1)
36 df.head()
37
38

```

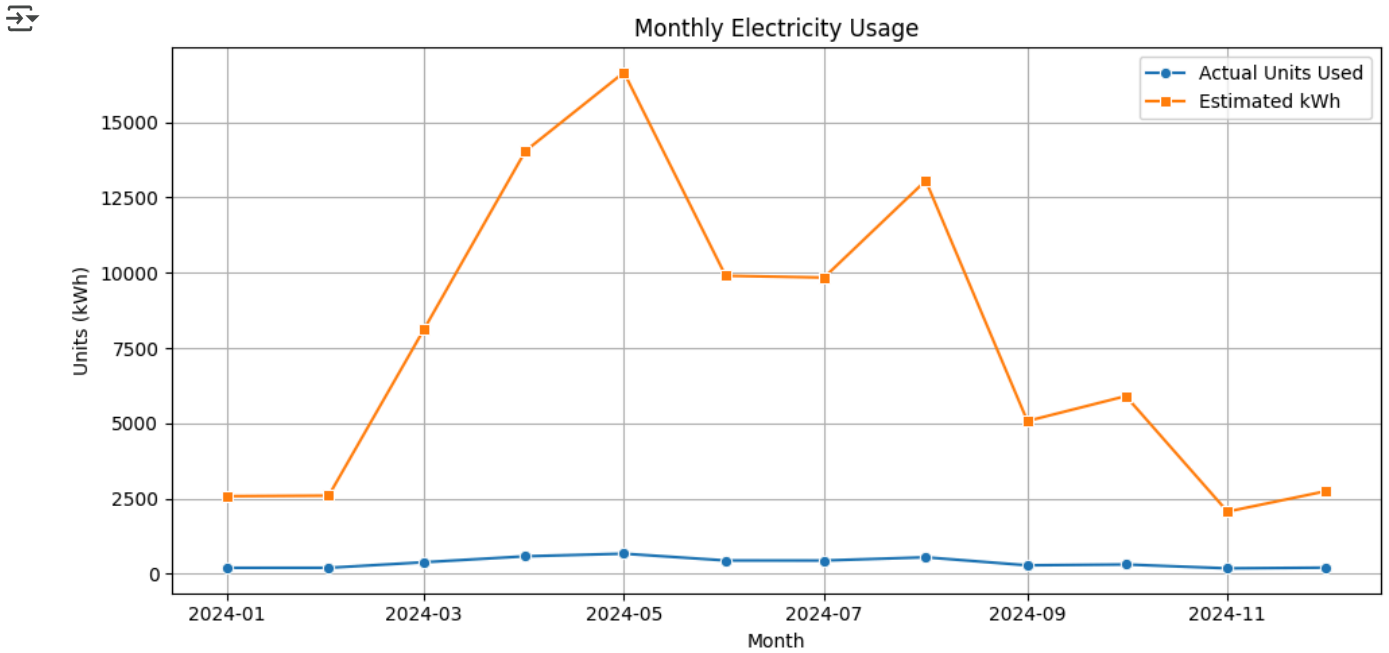
	Month	Units_Used	Bill_Amount	People	Fan_Hours	AC_Hours	Fridge_Watt	Washing_Hours	Cooking_Mins
0	2024-01-01	200	878	4	1079	0	160	4	92
1	2024-02-01	201	757	4	1070	0	160	6	108
2	2024-03-01	385	1747	4	1359	109	160	4	91
3	2024-04-01	583	2243	4	1744	221	160	6	85
4	2024-05-01	670	3122	4	1478	292	160	6	107

Next steps: [Generate code with df](#) [View recommended plots](#) [New interactive sheet](#)

```

1 # Visualize Monthly Trends
2
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5
6 plt.figure(figsize=(10, 5))
7 sns.lineplot(x=df["Month"], y=df["Units_Used"], marker='o', label='Actual Units Used')
8 sns.lineplot(x=df["Month"], y=df["Estimated_kWh"], marker='s', label='Estimated kWh')
9 plt.title("Monthly Electricity Usage")
10 plt.ylabel("Units (kWh)")
11 plt.legend()
12 plt.grid(True)
13 plt.tight_layout()
14 plt.show()

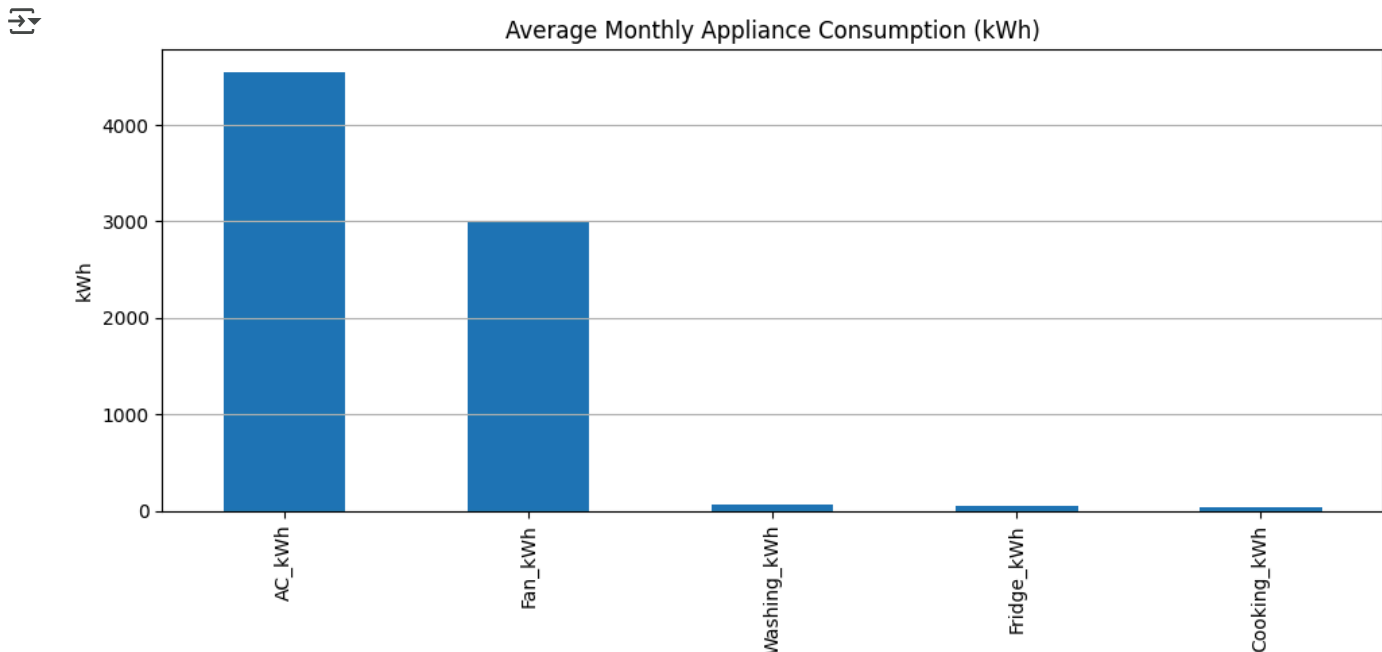
```



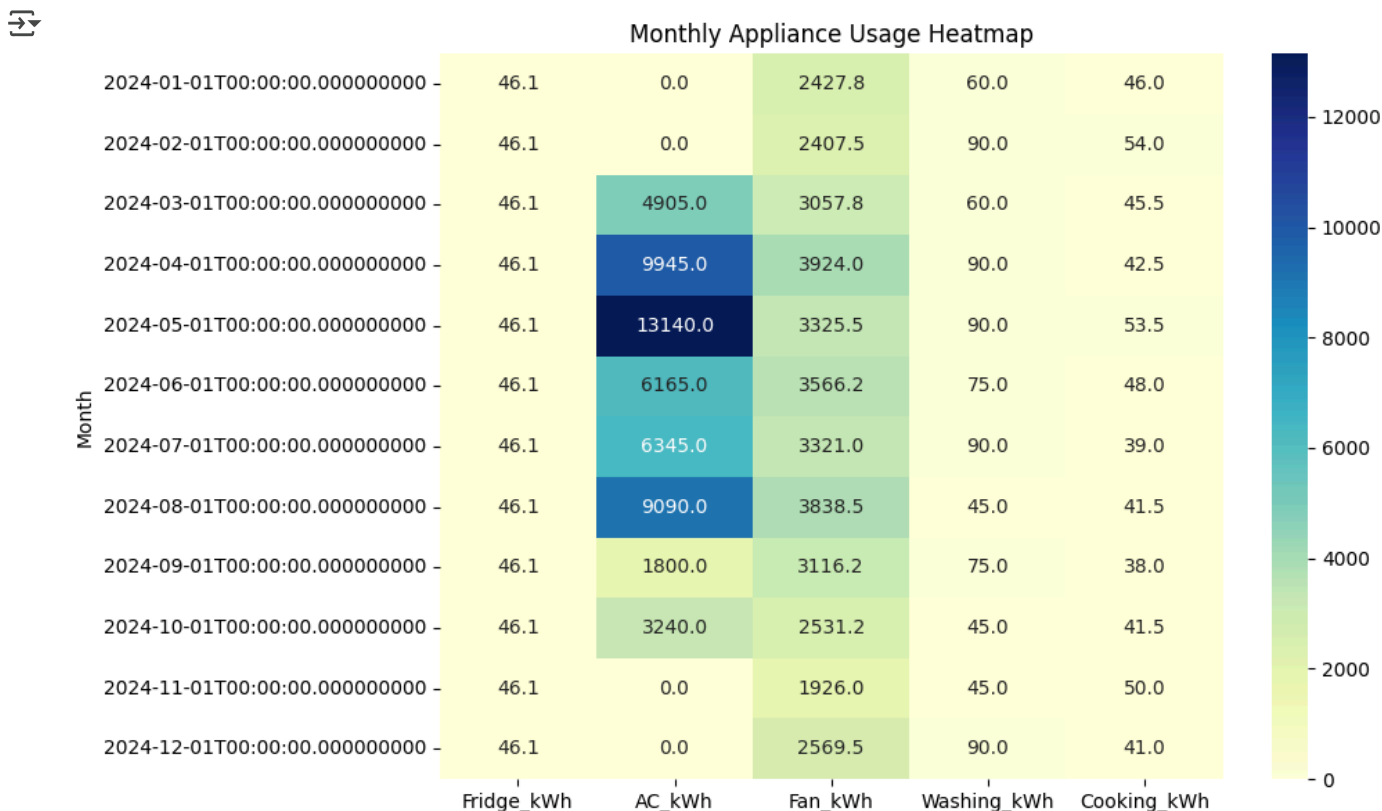
```

1 # Visualize Appliance Contributions
2
3 appliance_avg = df[["Fridge_kWh", "AC_kWh", "Fan_kWh", "Washing_kWh", "Cooking_kWh"]].mean().sort_val
4
5 appliance_avg.plot(kind='bar', title="Average Monthly Appliance Consumption (kWh)", ylabel="kWh", fig
6 plt.grid(axis='y')
7 plt.tight_layout()
8 plt.show()
9

```



```
1 # Heatmap of Appliance Usage by Month
2
3 import seaborn as sns
4
5 heat_df = df.set_index("Month")[["Fridge_kWh", "AC_kWh", "Fan_kWh", "Washing_kWh", "Cooking_kWh"]]
6 plt.figure(figsize=(10, 6))
7 sns.heatmap(heat_df, annot=True, cmap="YlGnBu", fmt=".1f")
8 plt.title("Monthly Appliance Usage Heatmap")
9 plt.tight_layout()
10 plt.show()
11
```



```

1 # Cost Estimation & Appliance Cost Split
2
3 unit_rate = 8 # ₹8 per kWh average
4
5 df["Estimated_Cost"] = df["Estimated_kWh"] * unit_rate
6 df["Cost_Diff"] = df["Bill_Amount"] - df["Estimated_Cost"]
7
8 # Appliance-level average cost
9 avg_costs = (appliance_avg * unit_rate).round(2)
10 print("Estimated Monthly Appliance-wise Cost (INR):")
11 print(avg_costs)

```

```

⇒ Estimated Monthly Appliance-wise Cost (INR):
AC_kWh      36420.00
Fan_kWh     24007.50
Washing_kWh  570.00
Fridge_kWh   368.64
Cooking_kWh  360.33
dtype: float64

```

```

1 # Recommendation
2 print("🔍 Recommendations to Reduce Bill:")
3 top_appliances = appliance_avg.sort_values(ascending=False).head(3)
4
5 for appliance, val in top_appliances.items():
6     print("\n")
7     if appliance == "AC_kWh":
8         print("\t- Use AC efficiently: Set to 24°C, ensure room insulation, use for fewer hours.")

```