Library Imports

This cell imports the required Python libraries for the project.

```
import pandas as pd
import os
```

Load & Tag Files

This function loads all CSV files and adds a dataset_type tag (enrollment, demographic, biometric) so we can easily separate them during analysis.

```
def load_and_tag(files, tag):
    dfs = []
    for file in files:
        df = pd.read_csv(file)
        df['dataset_type'] = tag
        dfs.append(df)
    return pd.concat(dfs, ignore_index=True)
```

Identify Dataset Files

This cell scans the working folder and separates files into enrollment, demographic, and biometric datasets for further processing.

```
enrollment_files = [f for f in os.listdir() if 'enrolment' in f.lower()]
demographic_files = [f for f in os.listdir() if 'demographic' in f.lower()]
biometric_files = [f for f in os.listdir() if 'biometric' in f.lower()]

print(enrollment_files)
print(demographic_files)
print(biometric_files)
```

```
['enrolment_3.csv', 'enrolment_1.csv', 'enrolment_2.csv']
['demographic_5.csv', 'demographic_3.csv', 'demographic_4.csv', 'demographic_2.csv', 'demographic_1.csv']
['aadhar_biometric_4.csv', 'aadhar_biometric_1.csv', 'aadhar_biometric_3.csv', 'aadhar_biometric_2.csv']
```

Load Datasets

This cell loads the enrollment, demographic, and biometric files into separate DataFrames and verifies their shapes.

```
enrollment_df = load_and_tag(enrollment_files, 'enrollment')
demographic_df = load_and_tag(demographic_files, 'demographic')
biometric_df = load_and_tag(biometric_files, 'biometric')

print(enrollment_df.shape)
print(demographic_df.shape)
print(biometric_df.shape)
```

```
(1006029, 8)
(2071700, 7)
(1861108, 7)
```

Merge All Datasets

This cell combines the enrollment, demographic, and biometric DataFrames into one unified dataset for analysis.

```
full_df = pd.concat([enrollment_df, demographic_df, biometric_df], ignore_index=True)

full_df.shape
```

```
(4938837, 12)
```

Check Columns

This cell displays all available columns in the merged dataset to confirm the structure before analysis.

```
print(full_df.columns)
```

```
Index(['date', 'state', 'district', 'pincode', 'age_0_5', 'age_5_17',
       'age_18_greater', 'dataset_type', 'demo_age_5_17', 'demo_age_17_',
       'bio_age_5_17', 'bio_age_17_'],
      dtype='object')
```

Calculate Enrollment Total

This cell creates a new column for total Aadhaar enrollment by summing all age group enrollment fields.

```
enroll_cols = ['age_0_5','age_5_17','age_18_greater']
full_df['enrollment_total'] = full_df[enroll_cols].sum(axis=1)
```

Calculate Demographic Total

This cell calculates the total number of demographic updates by summing all demographic age group fields.

```
demo_cols = [col for col in full_df.columns if col.startswith('demo_age')]
full_df['demographic_total'] = full_df[demo_cols].sum(axis=1)
```

Calculate Biometric Total

This cell calculates the total biometric updates by summing all biometric age group fields.

```
bio_cols = [col for col in full_df.columns if col.startswith('bio_age')]
full_df['biometric_total'] = full_df[bio_cols].sum(axis=1)
```

Verify Calculated Totals

This cell displays the newly created enrollment, demographic, and biometric total columns to validate the calculations.

```
full_df[['state','enrollment_total','demographic_total','biometric_total']].head()
```

	state	enrollment_total	demographic_total	biometric_total
0	Andhra Pradesh	1.0	0.0	0.0
1	Andhra Pradesh	1.0	0.0	0.0
2	Andhra Pradesh	1.0	0.0	0.0
3	Andhra Pradesh	1.0	0.0	0.0
4	Andhra Pradesh	1.0	0.0	0.0

State-Level Aggregation

This cell groups the data by state and calculates total enrollment, demographic updates, and biometric updates to identify high-demand states.

```
state_summary = full_df.groupby('state')[[
    'enrollment_total',
    'demographic_total',
    'biometric_total'
]].sum().reset_index()

state_summary.sort_values('enrollment_total', ascending=False).head(10)
```

	state	enrollment_total	demographic_total	biometric_total
54	Uttar Pradesh	1018629.0	8542328.0	9577735.0
7	Bihar	609585.0	4814350.0	4897587.0
32	Madhya Pradesh	493970.0	2912938.0	5923771.0
61	West Bengal	375297.0	3872172.0	2524448.0
33	Maharashtra	369139.0	5054602.0	9226139.0
47	Rajasthan	348458.0	2817615.0	3994955.0
19	Gujarat	280549.0	1824327.0	3196514.0
5	Assam	230197.0	1012578.0	982722.0
27	Karnataka	223235.0	1695285.0	2635954.0
49	Tamil Nadu	220789.0	2212228.0	4698117.0

District-Level Aggregation

This cell groups the data by state and district to calculate total enrollment, demographic updates, and biometric updates, helping identify high-pressure districts within each state.

```
district_summary = full_df.groupby(['state','district'])[
    ['enrollment_total','demographic_total','biometric_total']
].sum().reset_index()

district_summary.sort_values('enrollment_total', ascending=False).head(10)
```

```
district_summary.sort_values('demographic_total', ascending=False).head(10)
district_summary.sort_values('biometric_total', ascending=False).head(10)
```

	state	district	enrollment_total	demographic_total	biometric_total
578	Maharashtra	Pune	31763.0	438478.0	605762.0
574	Maharashtra	Nashik	22368.0	246100.0	576606.0
587	Maharashtra	Thane	43688.0	447253.0	571273.0
562	Maharashtra	Jalgaon	13260.0	151076.0	417384.0
241	Gujarat	Ahmedabad	19130.0	267884.0	405490.0
566	Maharashtra	Mumbai	14552.0	135483.0	404359.0
539	Maharashtra	Ahmadnagar	11836.0	227667.0	363561.0
800	Rajasthan	Jaipur	31146.0	275340.0	355884.0
570	Maharashtra	Nagpur	11828.0	158901.0	350923.0
29	Andhra Pradesh	Kurnool	11770.0	177645.0	350633.0

Age Group Summary

This cell calculates total Aadhaar activity for each age group (0–5, 5–17, and 18+), helping understand which age segment contributes most to enrollment.

```
age_summary = full_df[[
    'age_0_5',
    'age_5_17',
    'age_18_greater'
]].sum()

age_summary
```

	0
age_0_5	3546965.0
age_5_17	1720384.0
age_18_greater	168353.0

dtype: float64

Dataset Type Comparison

This cell groups the data by dataset_type and calculates total enrollment, demographic updates, and biometric updates to compare overall activity across all dataset categories.

```
type_summary = full_df.groupby('dataset_type')[[
    'enrollment_total',
    'demographic_total',
    'biometric_total'
]].sum().reset_index()

type_summary
```

	dataset_type	enrollment_total	demographic_total	biometric_total
0	biometric	0.0	0.0	69763095.0
1	demographic	0.0	49295187.0	0.0
2	enrollment	5435702.0	0.0	0.0

Chart: Top States by Enrollment

This cell visualizes the top states with the highest Aadhaar enrollment totals using a horizontal bar chart.

```
import matplotlib.pyplot as plt

top_states = full_df.groupby('state')['enrollment_total'].sum().sort_values(ascending=False).head(10)

plt.figure(figsize=(8,6))
plt.barh(top_states.index, top_states.values)
plt.gca().invert_yaxis()
plt.title("Top 10 States by Aadhaar Enrollment")
plt.xlabel("Enrollment Count")
plt.show()
```

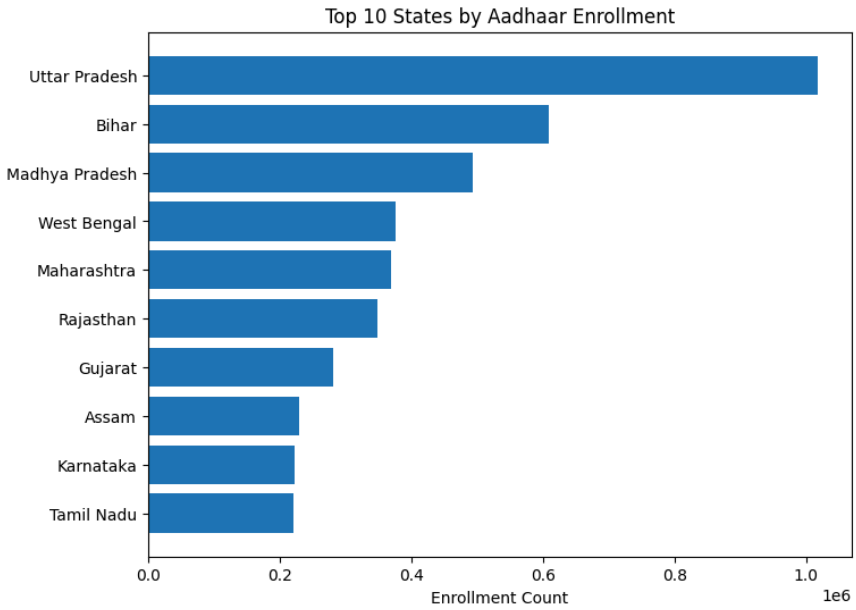


Chart: Top States by Demographic Updates

This cell visualizes the states with the highest demographic update counts using a horizontal bar chart.

```
top_states_demo = full_df.groupby('state')['demographic_total'].sum().sort_values(ascending=False).head(10)

plt.figure(figsize=(8,6))
plt.barh(top_states_demo.index, top_states_demo.values)
plt.gca().invert_yaxis()
plt.title("Top 10 States by Demographic Updates")
plt.xlabel("Demographic Update Count")
plt.show()
```

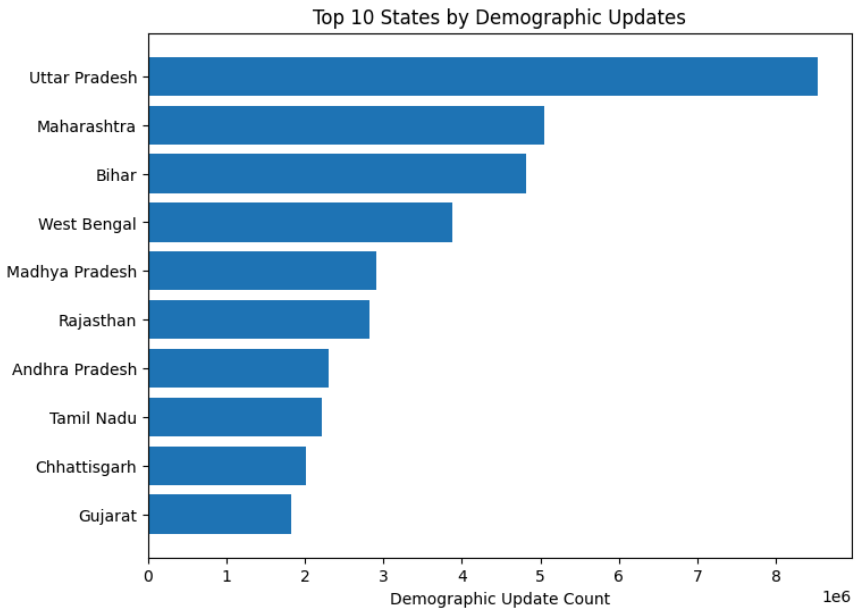
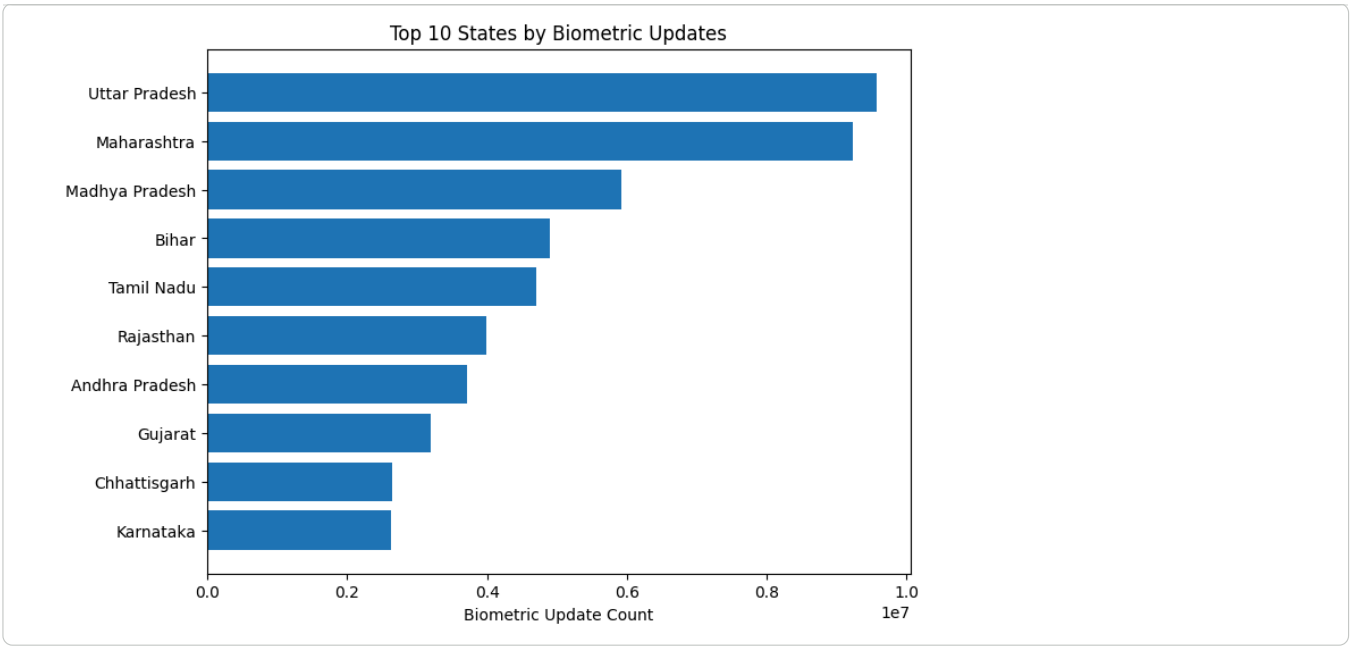


Chart: Top States by Biometric Updates

This cell visualizes the states with the highest biometric update counts using a horizontal bar chart.

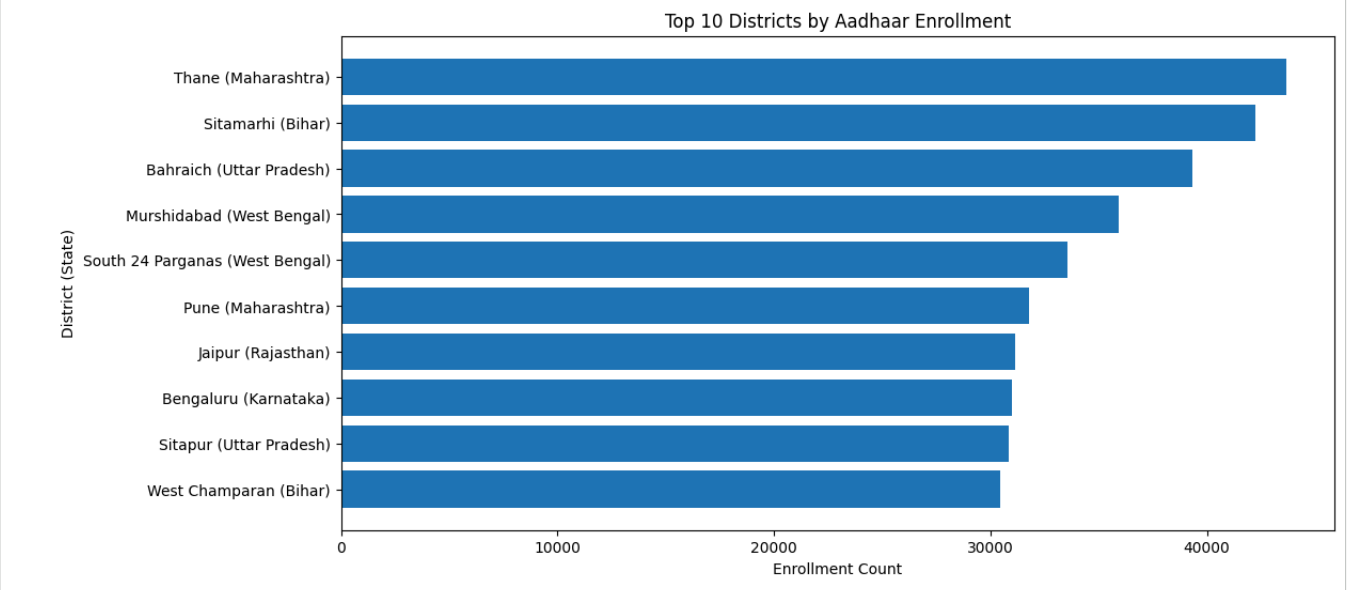
```
top_states_bio = full_df.groupby('state')['biometric_total'].sum().sort_values(ascending=False).head(10)

plt.figure(figsize=(8,6))
plt.barh(top_states_bio.index, top_states_bio.values)
plt.gca().invert_yaxis()
plt.title("Top 10 States by Biometric Updates")
plt.xlabel("Biometric Update Count")
plt.show()
```



Visualization:Top 10 Districts by Aadhar Enrollment

This chart displays the districts with the highest Aadhaar enrollment demand.
Each district is labeled along with its state to clearly represent regional activity levels and support UIDAI infrastructure planning.

```
top_districts['district_state'] = top_districts['district'] + " (" + top_districts['state'] + ")"

import matplotlib.pyplot as plt

plt.figure(figsize=(12,6))

plt.barh(top_districts['district_state'], top_districts['enrollment_total'])

plt.xlabel("Enrollment Count")
plt.ylabel("District (State)")
plt.title("Top 10 Districts by Aadhaar Enrollment")

plt.gca().invert_yaxis()
plt.show()
```

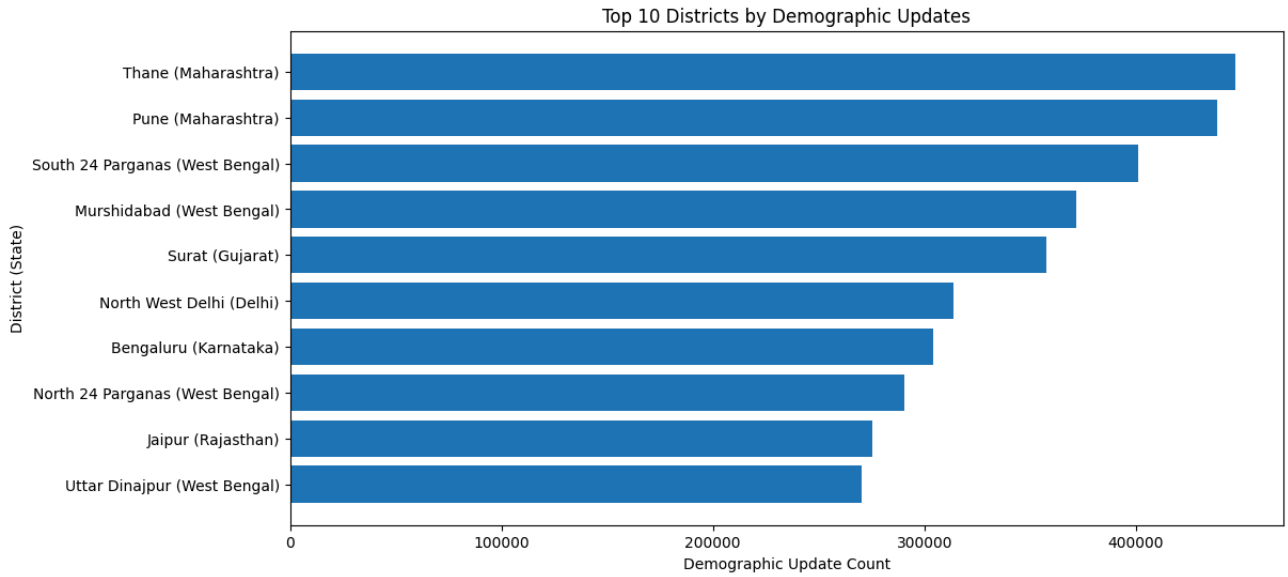


Visualization: Top 10 Districts by Demographic Updates

This chart shows districts with the highest number of demographic updates. District names are displayed along with their respective states to clearly identify regional demand patterns. High demographic updates often indicate migration, address changes, or data correction requirements.

```
top_demo_districts = district_summary.sort_values(
    'demographic_total', ascending=False
).head(10)
```

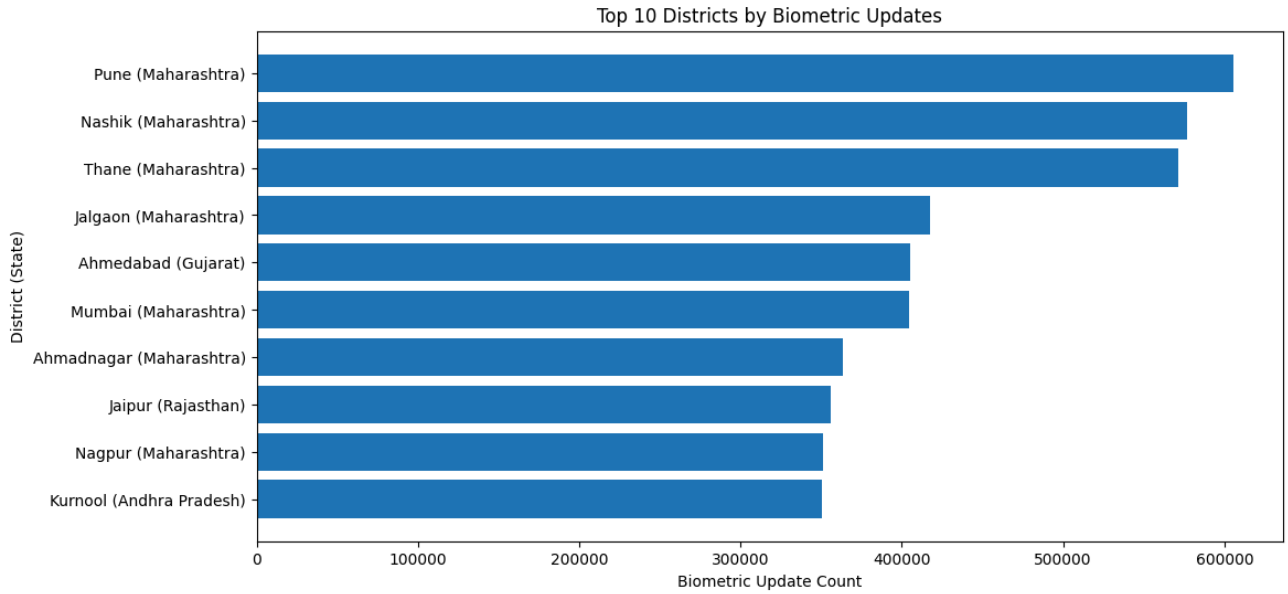
```
top_demo_districts['district_state'] = (
    top_demo_districts['district'] + " (" + top_demo_districts['state'] + ")"
)

import matplotlib.pyplot as plt

plt.figure(figsize=(12,6))
plt.barh(
    top_demo_districts['district_state'],
    top_demo_districts['demographic_total']
)

plt.xlabel("Demographic Update Count")
plt.ylabel("District (State)")
plt.title("Top 10 Districts by Demographic Updates")

plt.gca().invert_yaxis()
plt.show()
```



Visualization: Top 10 Districts by Biometric Updates

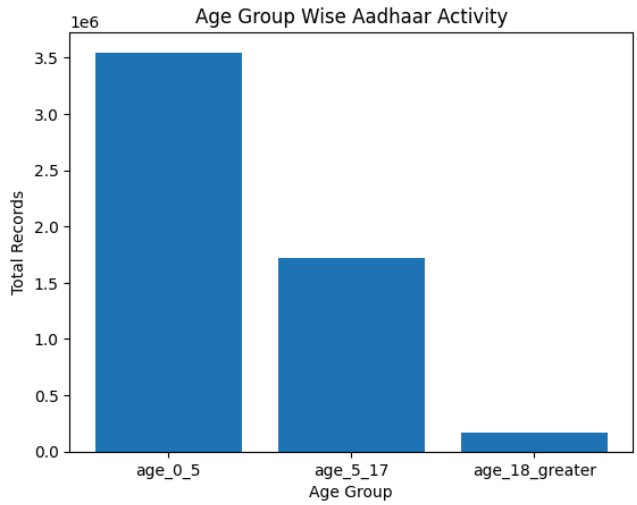
This chart highlights districts with the highest biometric update activity. A high biometric update count indicates frequent Aadhaar authentication corrections and highlights the need for better biometric capture quality and infrastructure.

```
top_bio_districts = district_summary.sort_values(
    'biometric_total', ascending=False
).head(10)

top_bio_districts['district_state'] = (
    top_bio_districts['district'] + " (" + top_bio_districts['state'] + ")"
)

plt.figure(figsize=(12,6))
plt.barh(
    top_bio_districts['district_state'],
    top_bio_districts['biometric_total']
)

plt.xlabel("Biometric Update Count")
plt.ylabel("District (State)")
plt.title("Top 10 Districts by Biometric Updates")

plt.gca().invert_yaxis()
plt.show()
```



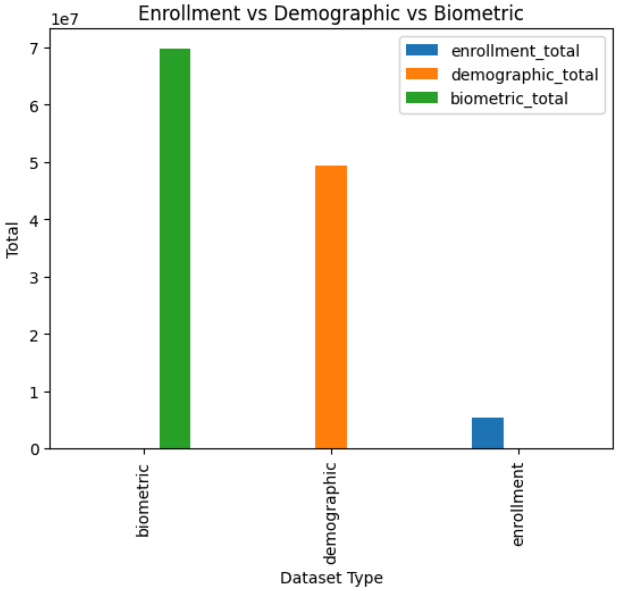
Age Group Distribution

```
age_totals = full_df[['age_0_5','age_5_17','age_18_greater']].sum()

plt.figure()
plt.bar(age_totals.index, age_totals.values)
plt.title("Age Group Wise Aadhaar Activity")
plt.xlabel("Age Group")
plt.ylabel("Total Records")
plt.show()
```



Dataset Type Comparison

```
type_summary = full_df.groupby('dataset_type')[['enrollment_total','demographic_total','biometric_total']].sum()

type_summary.plot(kind='bar')
plt.title("Enrollment vs Demographic vs Biometric")
plt.xlabel("Dataset Type")
plt.ylabel("Total")
plt.show()
```



District Level Aggregation

This cell groups the Aadhaar dataset by state and district to identify which districts have the highest enrollment, demographic update, and biometric update activity.
It helps highlight regional demand patterns at the district level.

```
district_summary = full_df.groupby(['state','district'])[
    ['enrollment_total','demographic_total','biometric_total']
].sum().reset_index()

district_summary.sort_values('enrollment_total', ascending=False).head(10)
```

	state	district	enrollment_total	demographic_total	biometric_total
587	Maharashtra	Thane	43688.0	447253.0	571273.0
161	Bihar	Sitamarhi	42232.0	169848.0	182000.0
946	Uttar Pradesh	Bahraich	39338.0	129697.0	181078.0
1095	West Bengal	Murshidabad	35911.0	371953.0	219079.0
1110	West Bengal	South 24 Parganas	33540.0	401137.0	224624.0
578	Maharashtra	Pune	31763.0	438478.0	605762.0
800	Rajasthan	Jaipur	31146.0	275340.0	355884.0
410	Karnataka	Bengaluru	30980.0	303924.0	297075.0
1024	Uttar Pradesh	Sitapur	30854.0	169322.0	264311.0
165	Bihar	West Champaran	30438.0	233389.0	166817.0

```
district_summary.sort_values('demographic_total', ascending=False).head(10)
```

	state	district	enrollment_total	demographic_total	biometric_total
587	Maharashtra	Thane	43688.0	447253.0	571273.0
578	Maharashtra	Pune	31763.0	438478.0	605762.0
1110	West Bengal	South 24 Parganas	33540.0	401137.0	224624.0
1095	West Bengal	Murshidabad	35911.0	371953.0	219079.0
273	Gujarat	Surat	25469.0	357582.0	281599.0
229	Delhi	North West Delhi	16043.0	313989.0	325293.0
410	Karnataka	Bengaluru	30980.0	303924.0	297075.0
1099	West Bengal	North 24 Parganas	28606.0	290477.0	223910.0
800	Rajasthan	Jaipur	31146.0	275340.0	355884.0
1116	West Bengal	Uttar Dinajpur	26892.0	270232.0	91756.0

```
district_summary.sort_values('biometric_total', ascending=False).head(10)
```

	state	district	enrollment_total	demographic_total	biometric_total
578	Maharashtra	Pune	31763.0	438478.0	605762.0
574	Maharashtra	Nashik	22368.0	246100.0	576606.0
587	Maharashtra	Thane	43688.0	447253.0	571273.0
562	Maharashtra	Jalgaon	13260.0	151076.0	417384.0
241	Gujarat	Ahmedabad	19130.0	267884.0	405490.0
566	Maharashtra	Mumbai	14552.0	135483.0	404359.0
539	Maharashtra	Ahmadnagar	11836.0	227667.0	363561.0
800	Rajasthan	Jaipur	31146.0	275340.0	355884.0
570	Maharashtra	Nagpur	11828.0	158901.0	350923.0
29	Andhra Pradesh	Kurnool	11770.0	177645.0	350633.0

Enrollment vs Update Ratio by District

This cell calculates how heavily each district relies on Aadhaar updates compared to new enrollments.
A higher ratio indicates stronger update demand and potential infrastructure pressure in that district.

```
district_summary['update_ratio'] = (
    district_summary['demographic_total'] + district_summary['biometric_total']
) / (district_summary['enrollment_total'] + 1)

district_summary.sort_values('update_ratio', ascending=False).head(10)
```

	state	district	enrollment_total	demographic_total	biometric_total	update_ratio
809	Rajasthan	Khairthal-Tijara	0.0	1057.0	15.0	1072.000000
811	Rajasthan	Kotputli-Behror	0.0	529.0	7.0	536.000000
900	Telangana	Medchal?malkajgiri	2.0	350.0	856.0	402.000000
196	Chhattisgarh	ManendragarhChirmiriBharatpur	0.0	0.0	312.0	312.000000
783	Rajasthan	Beawar	1.0	510.0	8.0	259.000000
779	Rajasthan	Balotra	1.0	503.0	6.0	254.500000
796	Rajasthan	Didwana-Kuchaman	2.0	720.0	8.0	242.666667
814	Rajasthan	Phalodi	0.0	204.0	2.0	206.000000
216	Daman & Diu	Daman	9.0	377.0	1412.0	178.900000
538	Maharashtra	Ahilyanagar	13.0	2418.0	17.0	173.928571

Age Group Activity Summary

This cell summarizes Aadhaar activity across different age groups (0–5, 5–17, 18+).
It helps understand which age segment contributes the most to enrollment and update demand.

```
age_summary = full_df[['age_0_5','age_5_17','age_18_greater']].sum()

age_summary
```

	0
age_0_5	3546965.0
age_5_17	1720384.0
age_18_greater	168353.0

dtype: float64

Recommendation Framework

This cell analyzes Aadhaar enrollment and update demand at the state level.
By combining demographic and biometric update totals, it identifies high-load states that require better infrastructure, faster processing, and increased service capacity.

The output of this cell is used to propose actionable recommendations for UIDAI, such as expanding Aadhaar service centers, improving biometric capture systems, and focusing on child enrollment and update awareness.

```
recommendation_df = full_df.groupby('state')[
    ['enrollment_total','demographic_total','biometric_total']
].sum().reset_index()

recommendation_df['update_load'] = (
    recommendation_df['demographic_total'] + recommendation_df['biometric_total']
)
```

	state	enrollment_total	demographic_total	biometric_total	update_load
54	Uttar Pradesh	1018629.0	8542328.0	9577735.0	18120063.0
33	Maharashtra	369139.0	5054602.0	9226139.0	14280741.0
7	Bihar	609585.0	4814350.0	4897587.0	9711937.0
32	Madhya Pradesh	493970.0	2912938.0	5923771.0	8836709.0
47	Rajasthan	348458.0	2817615.0	3994955.0	6812570.0
61	West Bengal	375297.0	3872172.0	2524448.0	6396620.0
3	Andhra Pradesh	127681.0	2295505.0	3714592.0	6010097.0
19	Gujarat	280549.0	1824327.0	3196514.0	5020841.0
10	Chhattisgarh	103219.0	2005434.0	2648729.0	4654163.0

```
recommendation_df.sort_values('update_load', ascending=False).head(10)
```

	state	enrollment_total	demographic_total	biometric_total	update_load
54	Uttar Pradesh	1018629.0	8542328.0	9577735.0	18120063.0
33	Maharashtra	369139.0	5054602.0	9226139.0	14280741.0
7	Bihar	609585.0	4814350.0	4897587.0	9711937.0
32	Madhya Pradesh	493970.0	2912938.0	5923771.0	8836709.0
49	Tamil Nadu	220789.0	2212228.0	4698117.0	6910345.0
47	Rajasthan	348458.0	2817615.0	3994955.0	6812570.0
61	West Bengal	375297.0	3872172.0	2524448.0	6396620.0
3	Andhra Pradesh	127681.0	2295505.0	3714592.0	6010097.0
19	Gujarat	280549.0	1824327.0	3196514.0	5020841.0
10	Chhattisgarh	103219.0	2005434.0	2648729.0	4654163.0

```
district_summary.sort_values('enrollment_total', ascending=False).head(10)
district_summary.sort_values('update_ratio', ascending=False).head(10)
```

	state	district	enrollment_total	demographic_total	biometric_total	update_ratio
809	Rajasthan	Khairthal-Tijara	0.0	1057.0	15.0	1072.000000
811	Rajasthan	Kotputli-Behror	0.0	529.0	7.0	536.000000