```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
```

```
# File lists
bio_files = [
    'api_data_aadhar_biometric_0_500000.csv',
    'api_data_aadhar_biometric_500000_1000000.csv',
    'api_data_aadhar_biometric_1000000_1500000.csv',
    'api_data_aadhar_biometric_1500000_1861108.csv'
]
demo_files = [
    'api_data_aadhar_demographic_0_500000.csv',
    'api_data_aadhar_demographic_500000_1000000.csv',
    'api_data_aadhar_demographic_1000000_1500000.csv',
    'api_data_aadhar_demographic_1500000_2000000.csv',
    'api_data_aadhar_demographic_2000000_2071700.csv'
]
enroll_files = [
    'api_data_aadhar_enrolment_0_500000.csv',
    'api_data_aadhar_enrolment_500000_1000000.csv',
    'api_data_aadhar_enrolment_1000000_1006029.csv'
]
```

```
# Loading and Aggregating
print("Aggregating Biometric data...")
bio_df = pd.concat([pd.read_csv(f) for f in bio_files]).groupby(['pincode', 'state', 'district']).sum().reset_index()
bio_df = bio_df[['pincode', 'state', 'district', 'bio_age_5_17', 'bio_age_17_']]

print("Aggregating Demographic data...")
demo_df = pd.concat([pd.read_csv(f) for f in demo_files]).groupby(['pincode', 'state', 'district']).sum().reset_index()
demo_df = demo_df[['pincode', 'state', 'district', 'demo_age_5_17', 'demo_age_17_']]

print("Aggregating Enrollment data...")
enroll_df = pd.concat([pd.read_csv(f) for f in enroll_files]).groupby(['pincode', 'state', 'district']).sum().reset_index()
enroll_df = enroll_df[['pincode', 'state', 'district', 'age_0_5', 'age_5_17', 'age_18_greater']]
enroll_df.rename(columns={'age_0_5': 'enroll_0_5', 'age_5_17': 'enroll_5_17', 'age_18_greater': 'enroll_18_plus'}, inplace=T
```

```
Aggregating Biometric data...
Aggregating Demographic data...
Aggregating Enrollment data...
```

```
# Merge all three
print("Merging all datasets...")
master_df = pd.merge(bio_df, demo_df, on=['pincode', 'state', 'district'], how='outer')
master_df = pd.merge(master_df, enroll_df, on=['pincode', 'state', 'district'], how='outer').fillna(0)
```

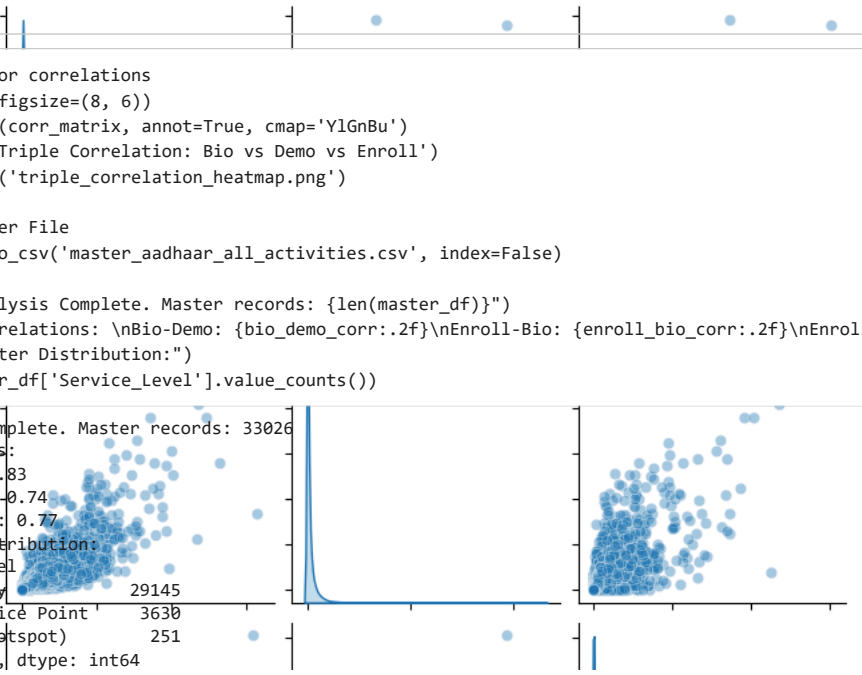
```
Merging all datasets...
```

```
# Total columns
master_df['total_bio'] = master_df['bio_age_5_17'] + master_df['bio_age_17_']
master_df['total_demo'] = master_df['demo_age_5_17'] + master_df['demo_age_17_']
master_df['total_enroll'] = master_df['enroll_0_5'] + master_df['enroll_5_17'] + master_df['enroll_18_plus']
```

```
# 1. Correlation Analysis
corr_cols = ['total_bio', 'total_demo', 'total_enroll']
corr_matrix = master_df[corr_cols].corr()

# 2. Relationship Viz
plt.figure(figsize=(10, 6))
sns.pairplot(master_df[corr_cols], diag_kind='kde', plot_kws={'alpha': 0.4})
plt.savefig('triple_relationship_pairplot.png')

# 3. Clustering - Identifying the "Super Hubs"
scaler = StandardScaler()
X_scaled = scaler.fit_transform(master_df[corr_cols])
kmeans = KMeans(n_clusters=3, random_state=42, n_init=10)
master_df['master_cluster'] = kmeans.fit_predict(X_scaled)

# Name Clusters
cluster_means = master_df.groupby('master_cluster')[['total_bio', 'total_demo', 'total_enroll']].mean()
cluster_ranking = cluster_means.sum(axis=1).sort_values().index
cluster_map = {cluster_ranking[0]: 'Low Activity', cluster_ranking[1]: 'Active Service Point', cluster_ranking[2]: 'Mega Hub
master_df['Service_Level'] = master_df['master_cluster'].map(cluster_map)

# 4. State Comparison - Total Volume
```

```
state_master = master_df.groupby('state')[['total_bio', 'total_demo', 'total_enroll']].sum().sort_values(by='total_bio', asc
state_master.plot(kind='bar', stacked=True, figsize=(12, 6), color=['#3498db', '#9b59b6', '#2ecc71'])
plt.title('Top 10 States: Total Aadhaar Activities Breakdown')
plt.ylabel('Activity Count')
plt.savefig('triple_state_breakdown.png')

# 5. Finding the "Best Relationship"
bio_demo_corr = corr_matrix.loc['total_bio', 'total_demo']
enroll_bio_corr = corr_matrix.loc['total_enroll', 'total_bio']
enroll_demo_corr = corr_matrix.loc['total_enroll', 'total_demo']
```

```
<Figure size 1000x600 with 0 Axes>
```



```
# Heatmap for correlations
plt.figure(figsize=(8, 6))
sns.heatmap(corr_matrix, annot=True, cmap='YlGnBu')
plt.title('Triple Correlation: Bio vs Demo vs Enroll')
plt.savefig('triple_correlation_heatmap.png')

# Save Master File
master_df.to_csv('master_aadhaar_all_activities.csv', index=False)

print(f"Analysis Complete. Master records: {len(master_df)}")
print(f"Correlations: \nBio-Demo: {bio_demo_corr:.2f}\nEnroll-Bio: {enroll_bio_corr:.2f}\nEnroll-Demo: {enroll_demo_corr:.2f
print("Cluster Distribution:")
print(master_df['Service_Level'].value_counts())
```

```
Analysis Complete. Master records: 33026
Correlations:
Bio-Demo: 0.83
Enroll-Bio: 0.74
Enroll-Demo: 0.77
Cluster Distribution:
Service_Level
Low Activity                29145
Active Service Point         3630
Mega Hub (Hotspot)            251
Name: count, dtype: int64
```