Tax Receipts Data Analysis

Business Analysis Project Report

॥वसुधैव कुटुम्बकम्॥

SYMBIOSIS

INSTITUTE OF TECHNOLOGY, NAGPUR

Submitted to

DR. PIYUSH CHAUHAN
Associate Professor
Department of Computer Science &Engineering
Symbiosis Institute of Technology,
Nagpur Campus

Under the Guidance of

AMIT MAKODE SIR

Course Name: Business Analytics

Submitted by

Govind Gonnade
VI SEM
PRN: 22070521105
B. Tech Computer Science & Engineering
Symbiosis Institute of Technology,
Nagpur Campus

Table of Contents

Section No.	Section Title	Page No.
1	Introduction	4
1.1	Project Objective	4
1.2	Data Overview	4
2	Data Collection and Integration	5
2.1	Data Source and Collection	5
2.2	Data Provenance	5
2.3	Integration Methodology	5
2.4	Dataset Summary	5
3	Data Cleaning and Preparation	6
3.1	Check Shape, Data Types, and Null Values	6
3.2	Outlier Treatment	8
3.2.1	Address Outliers	8
3.3	Data Type Correction	10
3.4	Normalize Numerical Features	11
3.5	Encode Categorical Variables	12
3.6	Create Derived Features	12
4	Exploratory Data Analysis	13
4.1	Descriptive Statistics	13
4.2	Visualizations	14
4.3	Correlation Analysis	16
4.4	Test for Statistical Properties	17
5	Statistical Analysis	19
5.1	Hypothesis Testing	19
5.2	Regression Analysis	20
5.3	ANOVA test	21
6	Advanced Analytics	22
6.1	Machine Learning Model	22
6.2	Predictive Modeling Utility	22
6.3	Segmentation / Clustering	24
6.4	Time Series Analysis	27
7	Research Work	28
7.1	Composition and Trends in Tax Revenue of Government of India Since 1990s	28
7.2	Tax Structure and Economic Growth: A Study of Selected Indian States	28
7.3	Fiscal Policy in India: Trends and Trajectory	28
7.4	Fiscal Discipline and Its Relationship with Economic Growth in India: Evidence-based Policy Analysis	28
7.5	An Analysis of Trends and Patterns of Income Tax in India	29
8	Conclusion and Recommendations	29
8.1	Key Takeaways	29

8.2	**Business Recommendations**	**29**
8.3	**Future Work**	**30**
8.4	**Decision Analysis**	**30**

1. Introduction

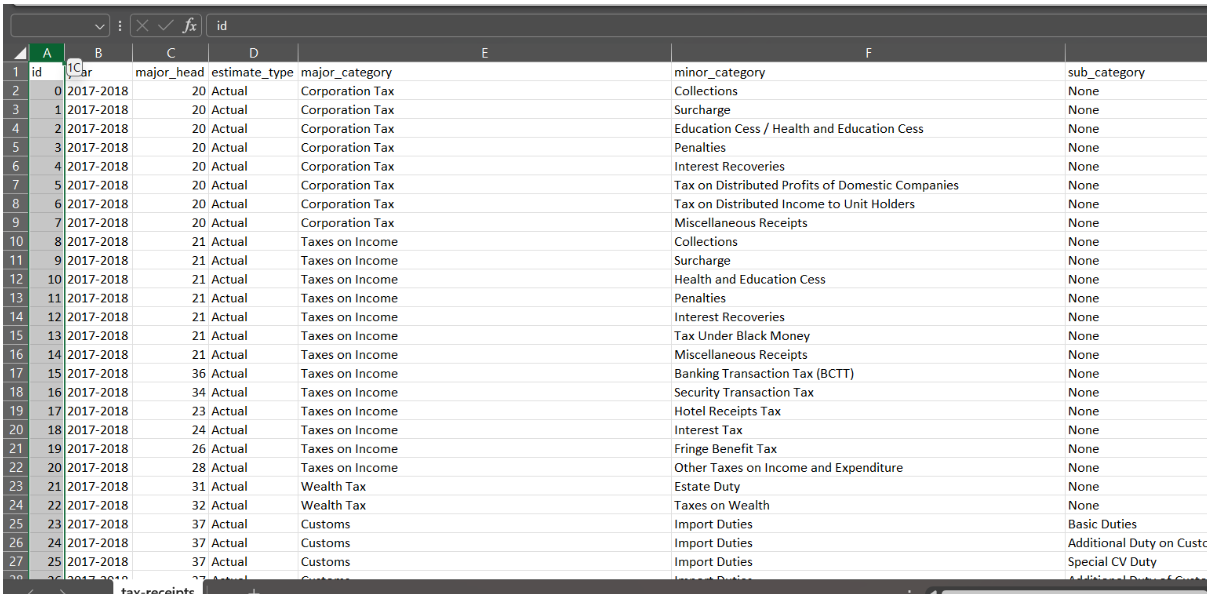
Taxation is and remains a decisive part of the economic organization of a country. In this project, this dataset of real world data of Indian government tax receipts is used to uncover trends, distributions and anomalies and prepare it for future predictive tasks.

1.1 Project Objective

- Determining the government tax receipts by years and categories
- Dealing with missing values of financial fields
- Visualize key patterns and relationships
- Test statistical assumptions (normality, variance)
- This lays a foundation for predictive modeling and clustering, covering trends, distributions, and anomalies in order to make the data ready for later predictive tasks.

1.2 Data Overview

Feature	Description
id	Unique identifier for each entry
year	Fiscal year (e.g., 2017–2018)
major_head	Numeric revenue type code
estimate_type	Estimate classification (Actual/Budget)
major_category	Main category of tax (e.g., Corporation Tax)
minor_category	Subgroup of major category
sub_category	Further classification of tax
amount	Tax collected (in crore INR) – has nulls



	A	B	C	D	E	F	
1	id	ar	major_head	estimate_type	major_category	minor_category	sub_category
2	0	2017-2018	20	Actual	Corporation Tax	Collections	None
3	1	2017-2018	20	Actual	Corporation Tax	Surcharge	None
4	2	2017-2018	20	Actual	Corporation Tax	Education Cess / Health and Education Cess	None
5	3	2017-2018	20	Actual	Corporation Tax	Penalties	None
6	4	2017-2018	20	Actual	Corporation Tax	Interest Recoveries	None
7	5	2017-2018	20	Actual	Corporation Tax	Tax on Distributed Profits of Domestic Companies	None
8	6	2017-2018	20	Actual	Corporation Tax	Tax on Distributed Income to Unit Holders	None
9	7	2017-2018	20	Actual	Corporation Tax	Miscellaneous Receipts	None
10	8	2017-2018	21	Actual	Taxes on Income	Collections	None
11	9	2017-2018	21	Actual	Taxes on Income	Surcharge	None
12	10	2017-2018	21	Actual	Taxes on Income	Health and Education Cess	None
13	11	2017-2018	21	Actual	Taxes on Income	Penalties	None
14	12	2017-2018	21	Actual	Taxes on Income	Interest Recoveries	None
15	13	2017-2018	21	Actual	Taxes on Income	Tax Under Black Money	None
16	14	2017-2018	21	Actual	Taxes on Income	Miscellaneous Receipts	None
17	15	2017-2018	36	Actual	Taxes on Income	Banking Transaction Tax (BCTT)	None
18	16	2017-2018	34	Actual	Taxes on Income	Security Transaction Tax	None
19	17	2017-2018	23	Actual	Taxes on Income	Hotel Receipts Tax	None
20	18	2017-2018	24	Actual	Taxes on Income	Interest Tax	None
21	19	2017-2018	26	Actual	Taxes on Income	Fringe Benefit Tax	None
22	20	2017-2018	28	Actual	Taxes on Income	Other Taxes on Income and Expenditure	None
23	21	2017-2018	31	Actual	Wealth Tax	Estate Duty	None
24	22	2017-2018	32	Actual	Wealth Tax	Taxes on Wealth	None
25	23	2017-2018	37	Actual	Customs	Import Duties	Basic Duties
26	24	2017-2018	37	Actual	Customs	Import Duties	Additional Duty on Custc
27	25	2017-2018	37	Actual	Customs	Import Duties	Special CV Duty

2. Data Collection and Integration

2.1 Data Source and Collection

The dataset used in this project was sourced from the Indian Data Portal (IDP- indiadataportal.com). The data is presumably collected from Ministry of Finance.

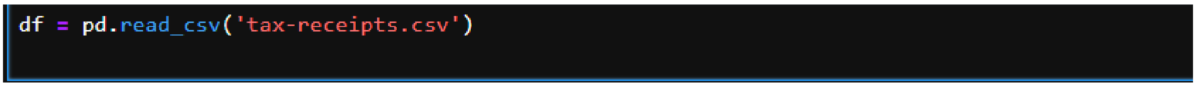
Data Set Link: TAX RECEIPTS

2.2 Data Provenance

- Source Authority: Ministry of Finance
- Dataset Format: CSV (Comma-Separated Values)
- Date of Access: 17 April 2025

2.3 Integration Methodology

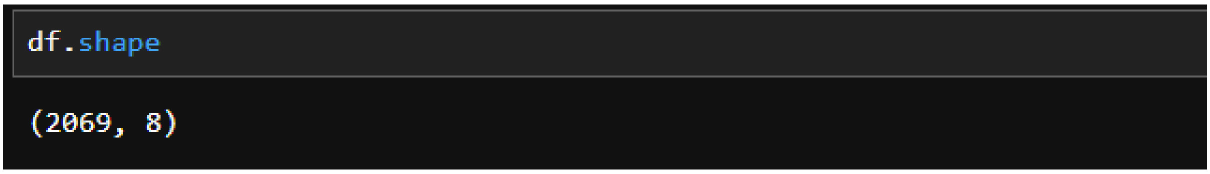
The analysis pipeline is set up in Python and the dataset is integrated. Using the Pandas library, the data was loaded into a DataFrame.



```
df = pd.read_csv('tax-receipts.csv')
```

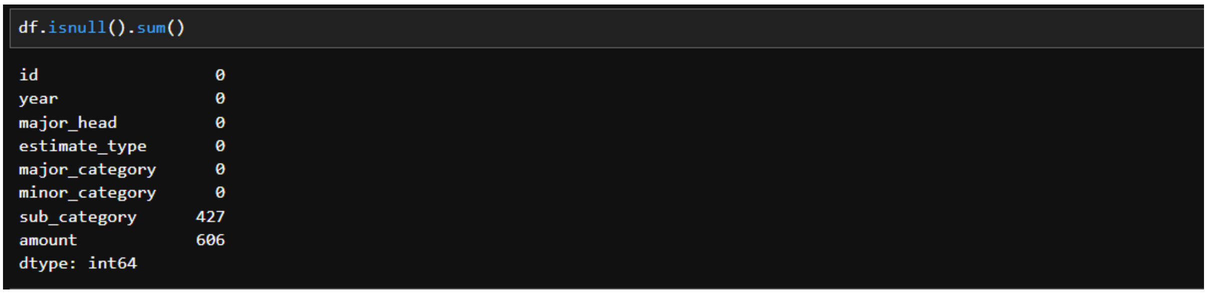
2.4 Dataset Summary:

- Total Records are 2069 and Total Columns are 8.



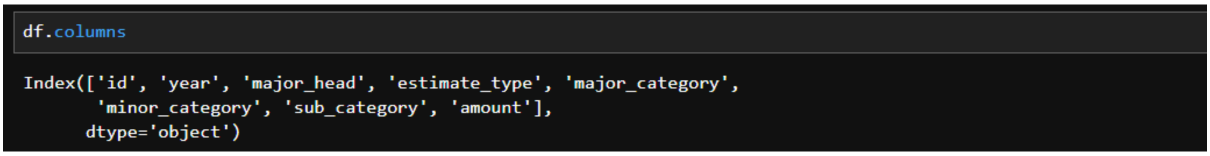
```
df.shape

(2069, 8)
```

- Missing values in amount: 606 and in sub_category: 427



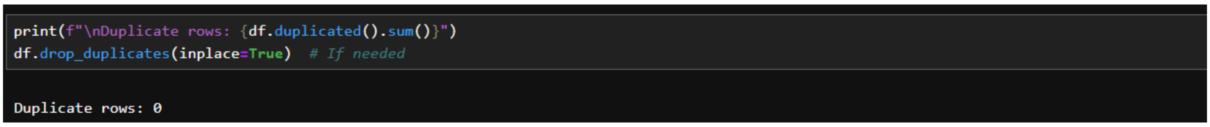
```
df.isnull().sum()

id                  0
year                0
major_head          0
estimate_type       0
major_category      0
minor_category      0
sub_category      427
amount            606
dtype: int64
```

- The column headers were all correctly named and all interpretable.



```
df.columns

Index(['id', 'year', 'major_head', 'estimate_type', 'major_category',
       'minor_category', 'sub_category', 'amount'],
      dtype='object')
```

- No duplicate records values were found at this stage.

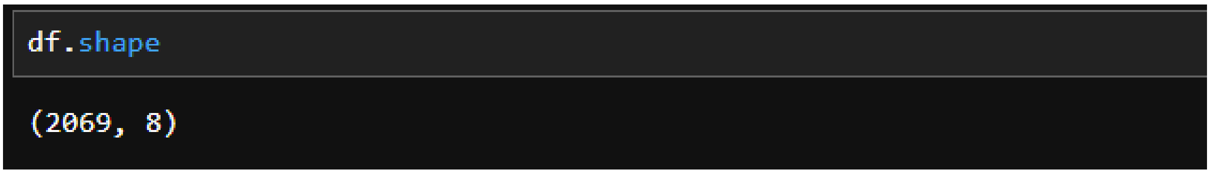


```
print(f"\nDuplicate rows: {df.duplicated().sum()}")
df.drop_duplicates(inplace=True)  # If needed

Duplicate rows: 0
```

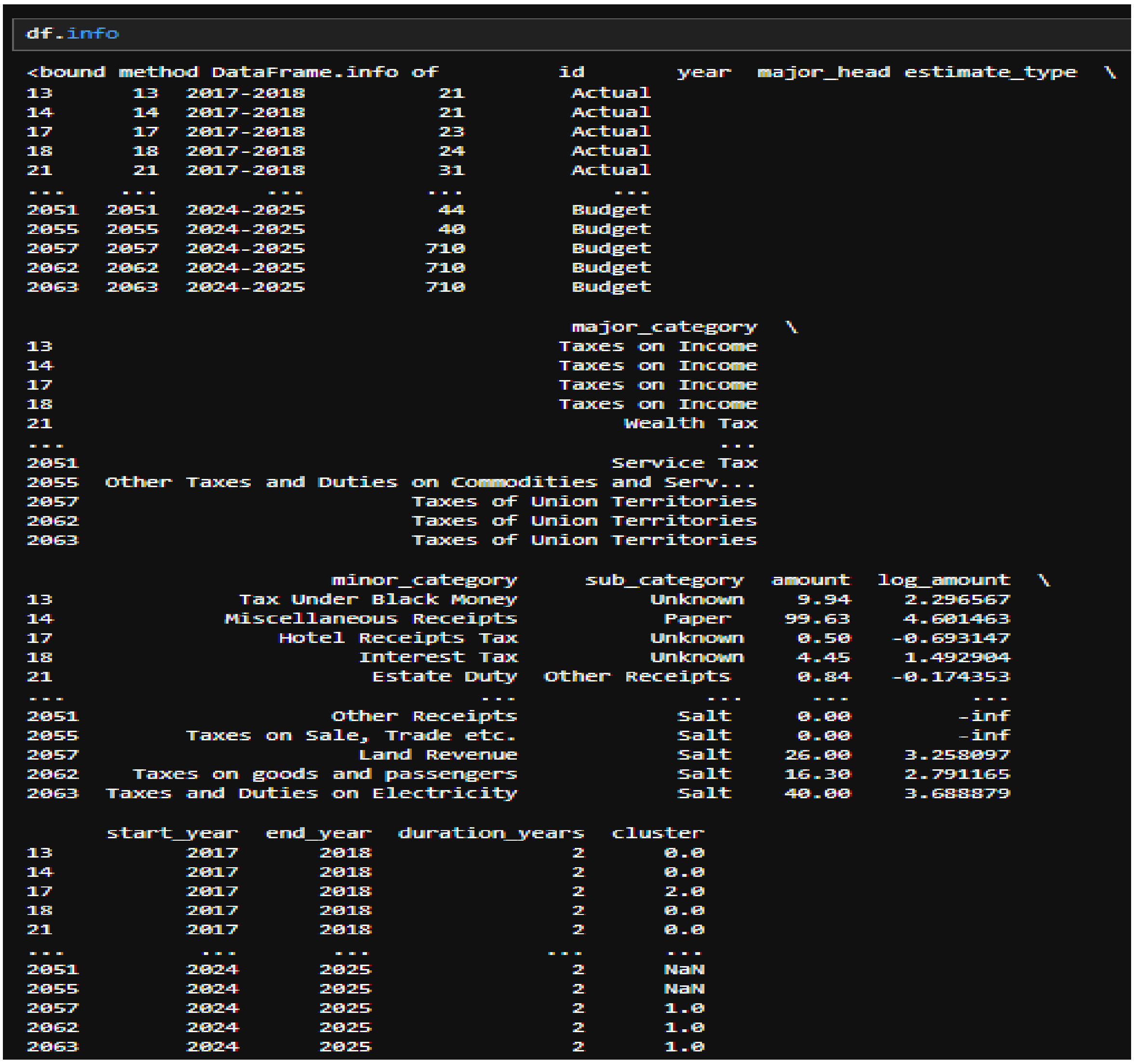
3. Data Cleaning and Preparation

3.1 Check Shape, Data Types, and Null Values

For the initial phase of data cleaning, it was evaluated for structure and integrity of the dataset. dataset is 2,069 rows by 8 columns which are tax receipts by multiple fiscal years and categories.

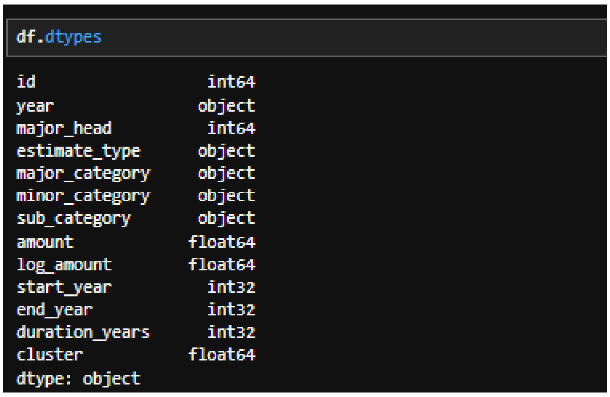


```
df.shape

(2069, 8)
```



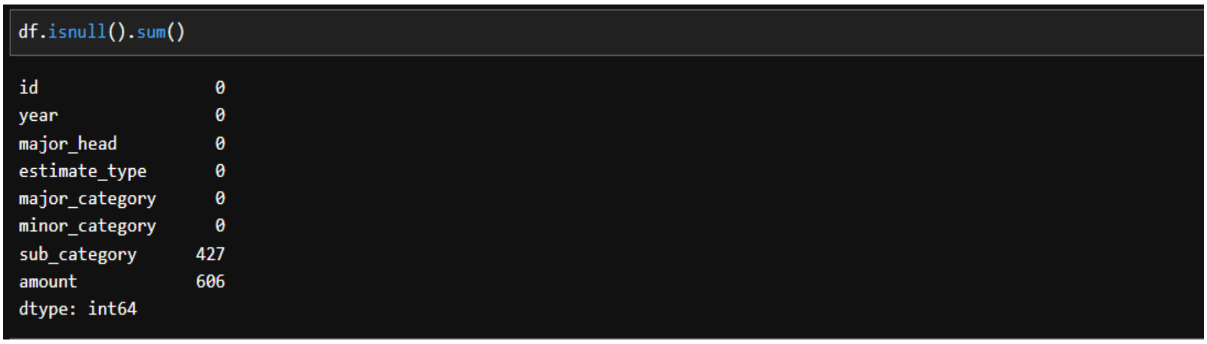
```
df.info

<bound method DataFrame.info of          id       year  major_head estimate_type  \
13      13  2017-2018          21        Actual
14      14  2017-2018          21        Actual
17      17  2017-2018          23        Actual
18      18  2017-2018          24        Actual
21      21  2017-2018          31        Actual
...    ...        ...         ...           ...
2051  2051  2024-2025          44        Budget
2055  2055  2024-2025          40        Budget
2057  2057  2024-2025         710        Budget
2062  2062  2024-2025         710        Budget
2063  2063  2024-2025         710        Budget

                                         major_category  \
13                                      Taxes on Income
14                                      Taxes on Income
17                                      Taxes on Income
18                                      Taxes on Income
21                                           Wealth Tax
...                                                 ...
2051                                        Service Tax
2055  Other Taxes and Duties on Commodities and Serv...
2057                         Taxes of Union Territories
2062                         Taxes of Union Territories
2063                         Taxes of Union Territories

                          minor_category     sub_category  amount  log_amount  \
13                 Tax Under Black Money          Unknown    9.94    2.296567
14               Miscellaneous Receipts            Paper   99.63    4.601463
17                     Hotel Receipts Tax          Unknown    0.50   -0.693147
18                           Interest Tax          Unknown    4.45    1.492904
21                           Estate Duty   Other Receipts    0.84   -0.174353
...                                  ...              ...     ...         ...
2051                      Other Receipts             Salt    0.00        -inf
2055              Taxes on Sale, Trade etc.          Salt    0.00        -inf
2057                        Land Revenue             Salt   26.00    3.258097
2062        Taxes on goods and passengers            Salt   16.30    2.791165
2063   Taxes and Duties on Electricity              Salt   40.00    3.688879

      start_year  end_year  duration_years  cluster
13          2017      2018               2      0.0
14          2017      2018               2      0.0
17          2017      2018               2      2.0
18          2017      2018               2      0.0
21          2017      2018               2      0.0
...          ...       ...             ...      ...
2051        2024      2025               2      NaN
2055        2024      2025               2      NaN
2057        2024      2025               2      1.0
2062        2024      2025               2      1.0
2063        2024      2025               2      1.0
```

The data types were verified so that they would become compatible for further analysis. Amount and major_head are stored as float64 and int64 so as to allow precise statistics to be done. int is the type of id column and is mapped to MySQL auto increment integer field, being a unique identifier for every record. All the descriptive fields like year, estimate_type, major_category, minor_category, sub_category are stored as object (string) types. In case these fields become categorical it is possible to cast them to categorical types later for optimizing processing and modeling performance.



```
df.dtypes

id                  int64
year               object
major_head          int64
estimate_type      object
major_category     object
minor_category     object
sub_category       object
amount            float64
log_amount        float64
start_year          int32
end_year            int32
duration_years      int32
cluster           float64
dtype: object
```

A thorough null value inspection confirmed that the dataset is mostly complete, with the exception of the amount column. Specifically:



```
df.isnull().sum()

id                  0
year                0
major_head          0
estimate_type       0
major_category      0
minor_category      0
sub_category      427
amount            606
dtype: int64
```

606 out of 2,069 entries in the amount field are missing, accounting for approximately 29.3% of the data.

This incomplete financial field requires appropriate handling before further analysis, particularly since amount is a primary variable of interest in revenue analysis.

3.2 Outlier Treatment

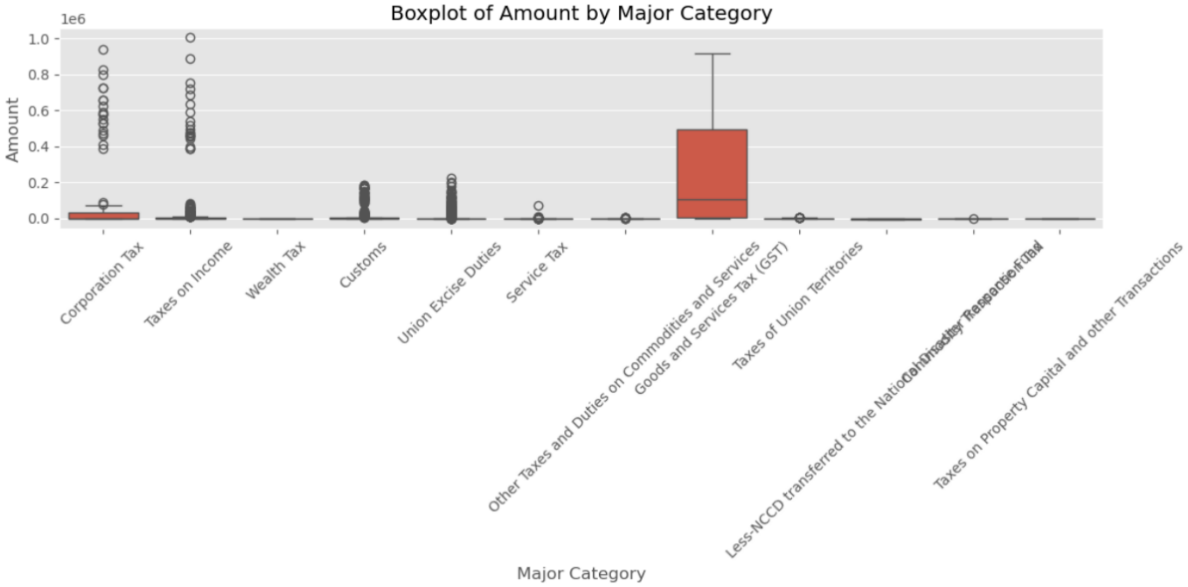
To deal with the presence of extreme values capable of distorting statistical analysis, I applied a comprehensive outlier treatment process to the amount column (secondary numeric feature of the dataset).

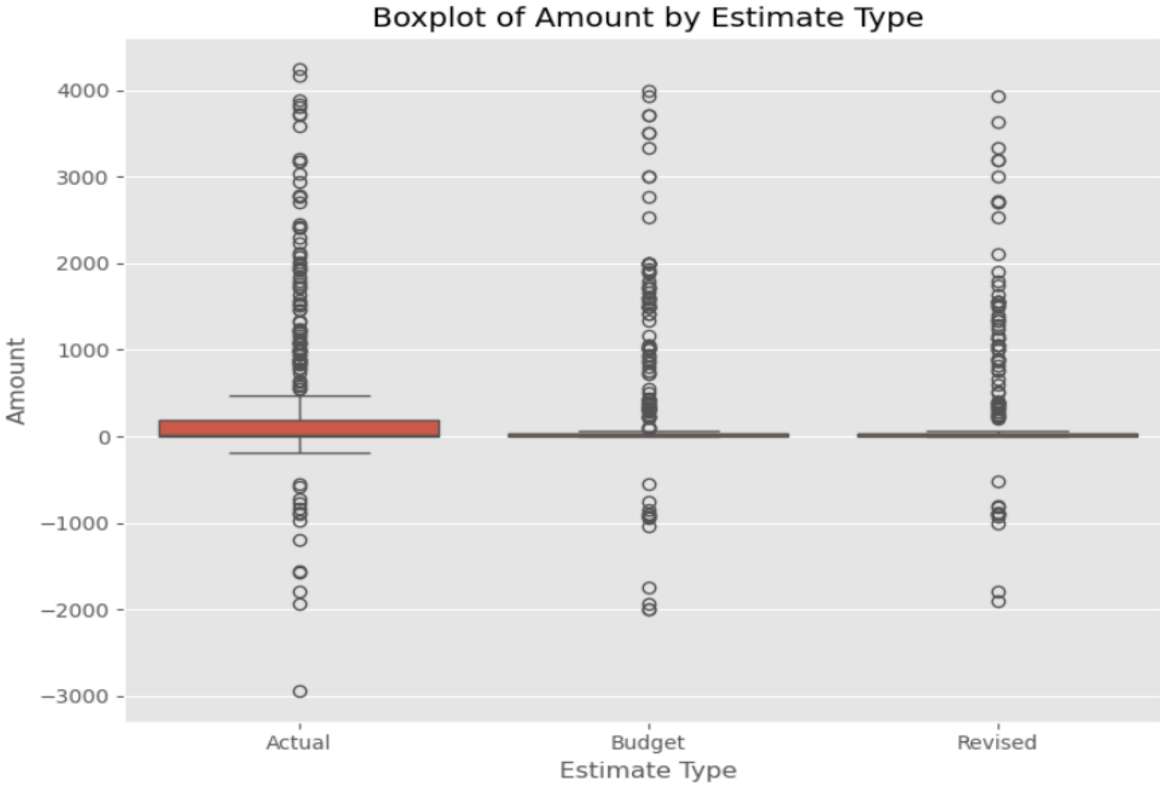
3.2.1 Address Outliers

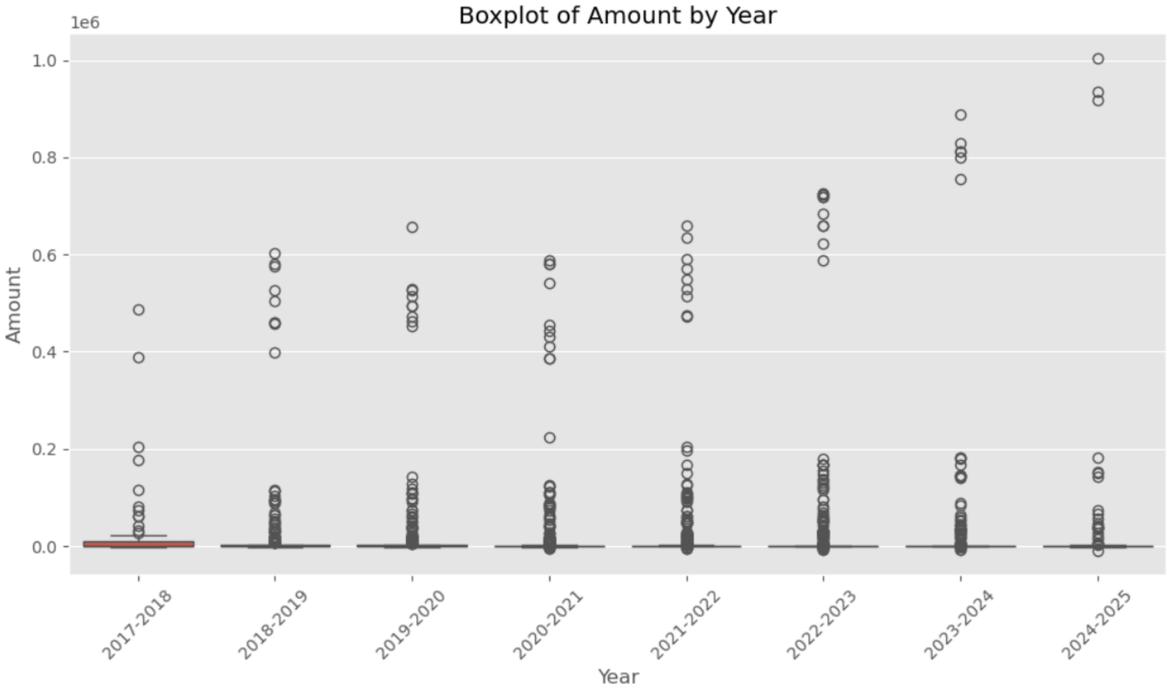
Visual initial outlier detection was performed using seaborn based boxplots. As we can observe the amount column is quite highly right skewed, we transformed it to be log transformed to stabilize the variance and to have the extremes values. Key categorical dimensions included and boxplots were generated across them:

- major_category
- estimate_type
- year

It enabled a clearer view of the underlying distribution pattern and the obvious presence of high outliers.







3.3 Data Type Corrections

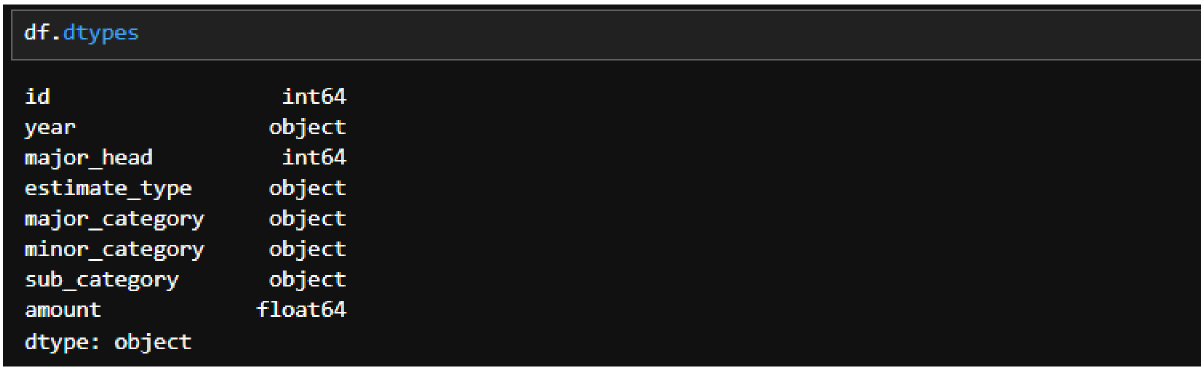
We first began by performing an initial audit on what the column data types are using the DataFrame.info() method. That confirmed that all columns are typed in the correct way according to their semantic meaning:

- amount: float64 (correct for financial magnitude)
- id, major_head: int64 (valid identifiers)
- year, estimate_type, major_category, minor_category, sub_category: object (categorical or textual)

Strings were not stored as numeric data (no inconsistencies, e.g. numeric data as strings and categorical values encoded as numerics). For analytical efficiency and improved plotting though, it is better to:

- It was explicitly converted to a categorical data type, a year.
- Amount was validated to be of float type to support continuous numerical operations.

Sufficient data hygiene (in schema design) was confirmed by the fact that there were no further corrections requested.



```
df.dtypes

id                  int64
year               object
major_head          int64
estimate_type      object
major_category     object
minor_category     object
sub_category       object
amount            float64
dtype: object
```

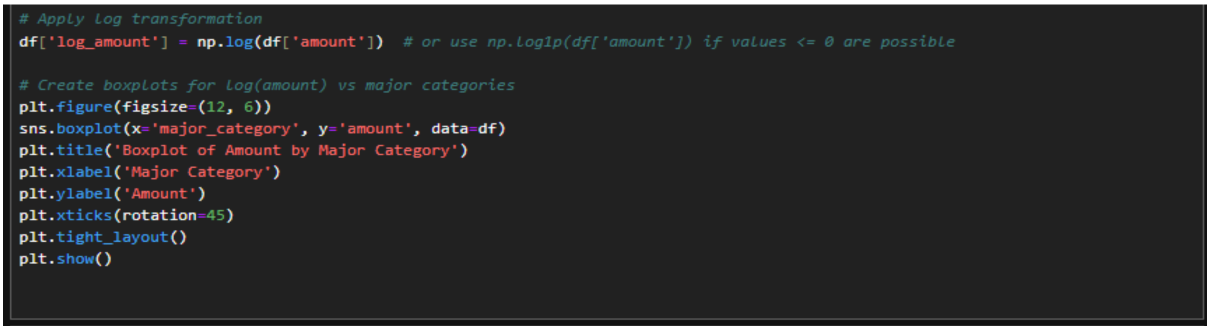
3.4 Normalize Numerical Features

To decide whether normalization was needed, it was found that the only significant numerical feature (amount) was not used in scale, where standardized scale is required (e.g. distance based machine learning models). Instead, amount served primarily in:

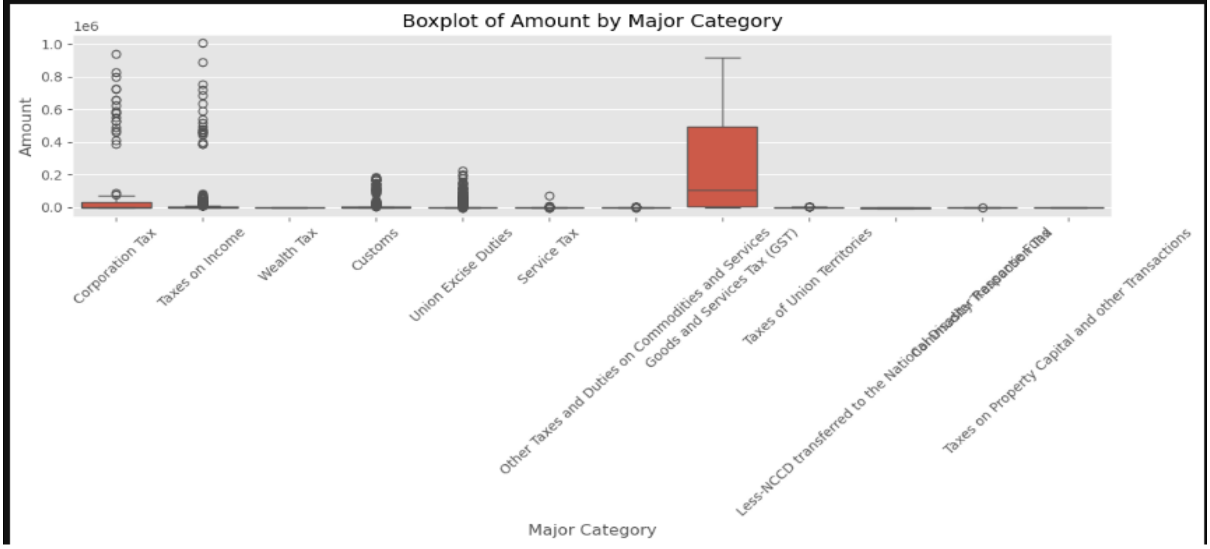
- Aggregations (e.g., total tax per category/year)
- Statistical analysis (e.g., outlier detection, boxplots)
- Correlation and distributional study

As clustering models were usually applied on categorical encodings or summarized groupings (as opposed to raw continuous features), normalization parameters such as Min Max or StandardScaler were not needed in this stage.

Alternatively, the log of amount was transformed to stabilize variance and reduce skewness for visualization and some of its statistical evaluations.



```
# Apply log transformation
df['log_amount'] = np.log(df['amount'])  # or use np.log1p(df['amount']) if values <= 0 are possible

# Create boxplots for log(amount) vs major categories
plt.figure(figsize=(12, 6))
sns.boxplot(x='major_category', y='amount', data=df)
plt.title('Boxplot of Amount by Major Category')
plt.xlabel('Major Category')
plt.ylabel('Amount')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



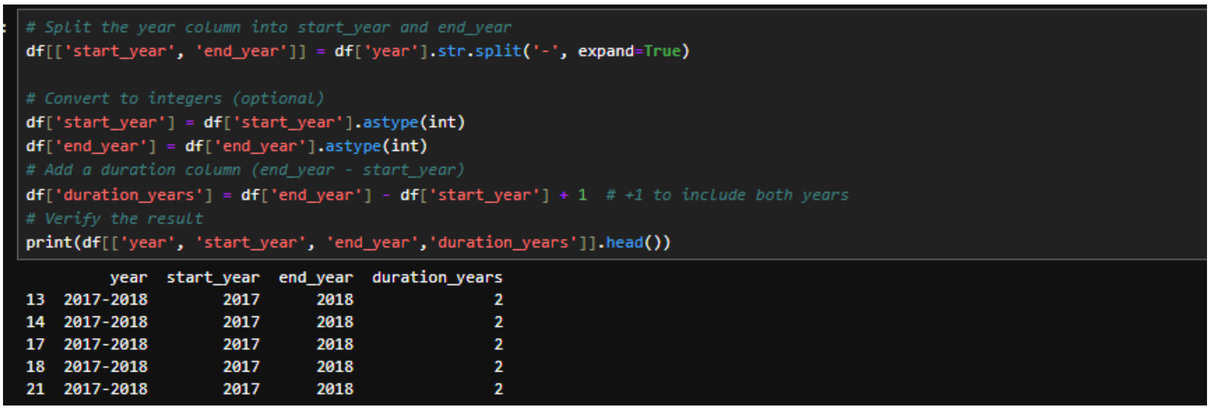
3.5 Encode Categorical Variables

However, in the current analysis no encoding of categorical variables was done. Categorical fields on year, estimate_type, major_category and minor_category were kept intact as their original string format object. The numerical version of these fields cas also be used directly to represent and collectively summarize the data in group-wise summaries and visualizations without any transformation into numerical form.

3.6 Create Derived Features

To prepare for the preprocessing workflow, I split fiscal year column (year) which was originally formatted as a string with the format of 'YYYY-YYYY' (2017-2018) into two separate integer based columns named start_year and end_year. It enabled chronological sorting, grouping time series, as well as fiscal year numeric operations.

The dataset became more compatible with temporal analysis and visualization through extraction of start and end years from the string. The derived features that were most useful for aggregating tax receipts by fiscal periods and also for observing year wise trends over major and minor categories were.



```python
# Split the year column into start_year and end_year
df[['start_year', 'end_year']] = df['year'].str.split('-', expand=True)

# Convert to integers (optional)
df['start_year'] = df['start_year'].astype(int)
df['end_year'] = df['end_year'].astype(int)
# Add a duration column (end_year - start_year)
df['duration_years'] = df['end_year'] - df['start_year'] + 1  # +1 to include both years
# Verify the result
print(df[['year', 'start_year', 'end_year','duration_years']].head())
```

```
         year  start_year  end_year  duration_years
13  2017-2018        2017      2018               2
14  2017-2018        2017      2018               2
17  2017-2018        2017      2018               2
18  2017-2018        2017      2018               2
21  2017-2018        2017      2018               2
```

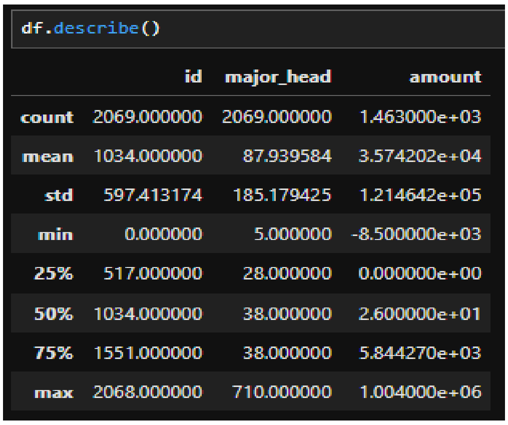
4. Exploratory Data Analysis

To investigate key patterns and trends in the Indian government's tax receipts dataset, EDA was performed. The analysis was to understand the structure and distribution of tax collection amount by fiscal years, type of estimates, and by tax categories. Central tendency and dispersion of the amount variable were summarized using descriptive statistics. Data skewness was examined and potential outliers were detected by using visual methods such as histograms, box plots and heatmaps and the category level variance was assessed. It was critical to demonstrate data integrity and ensure that features could be mapped, and that would guide them to the next step.

4.1 Descriptive Statistics

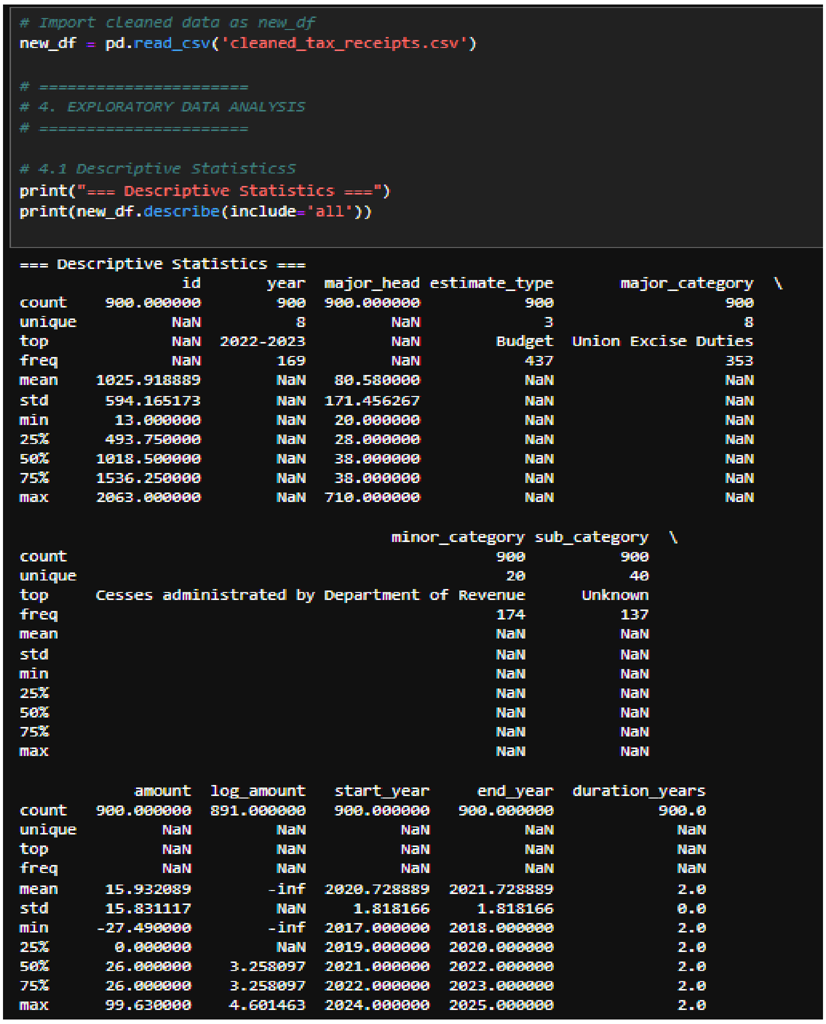
The key numeric column amount was evaluated in terms of the distributional characteristics and descriptive statistics were generated for the same. The describe() method from pandas was used to compute such metrics as mean, median, standard deviation, minimum, and maximum. Although the distribution of these measures appeared highly skewed with a long right tail, which suggested the existence of a few outliers with very high tax receipts compared to the median transaction, they did not reveal any outlier tax receipts that exceed the amount of total U.S. elementary and secondary tax receipts recorded in calendar year 2012.

This initial summary proved to be the best possible probe into the scale and variability of city tax collections, to prepare for more targeted analysis of collections across years and categories.



```
df.describe()
```

	id	major_head	amount
count	2069.000000	2069.000000	1.463000e+03
mean	1034.000000	87.939584	3.574202e+04
std	597.413174	185.179425	1.214642e+05
min	0.000000	5.000000	-8.500000e+03
25%	517.000000	28.000000	0.000000e+00
50%	1034.000000	38.000000	2.600000e+01
75%	1551.000000	38.000000	5.844270e+03
max	2068.000000	710.000000	1.004000e+06



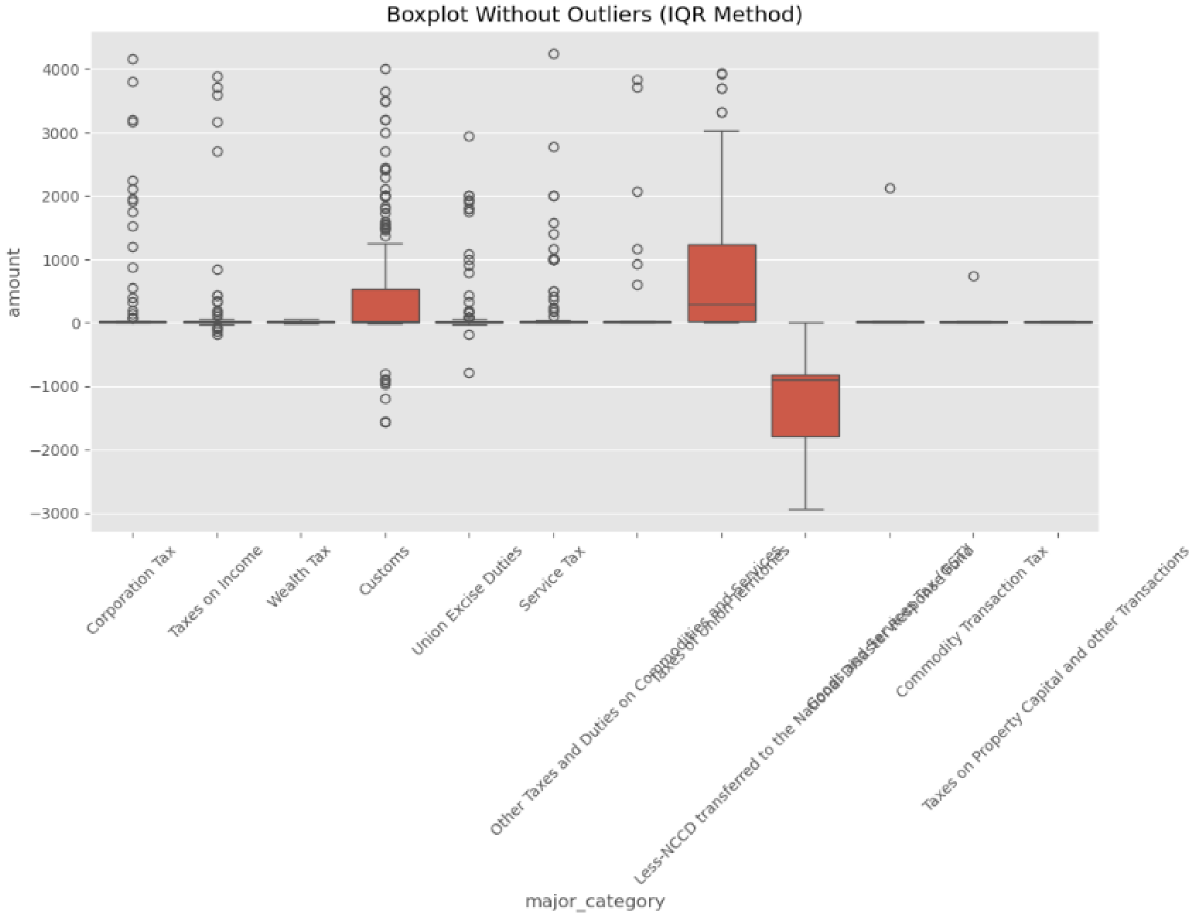
```
# Import cleaned data as new_df
new_df = pd.read_csv('cleaned_tax_receipts.csv')

# ======================
# 4. EXPLORATORY DATA ANALYSIS
# ======================

# 4.1 Descriptive StatisticsS
print("=== Descriptive Statistics ===")
print(new_df.describe(include='all'))
```

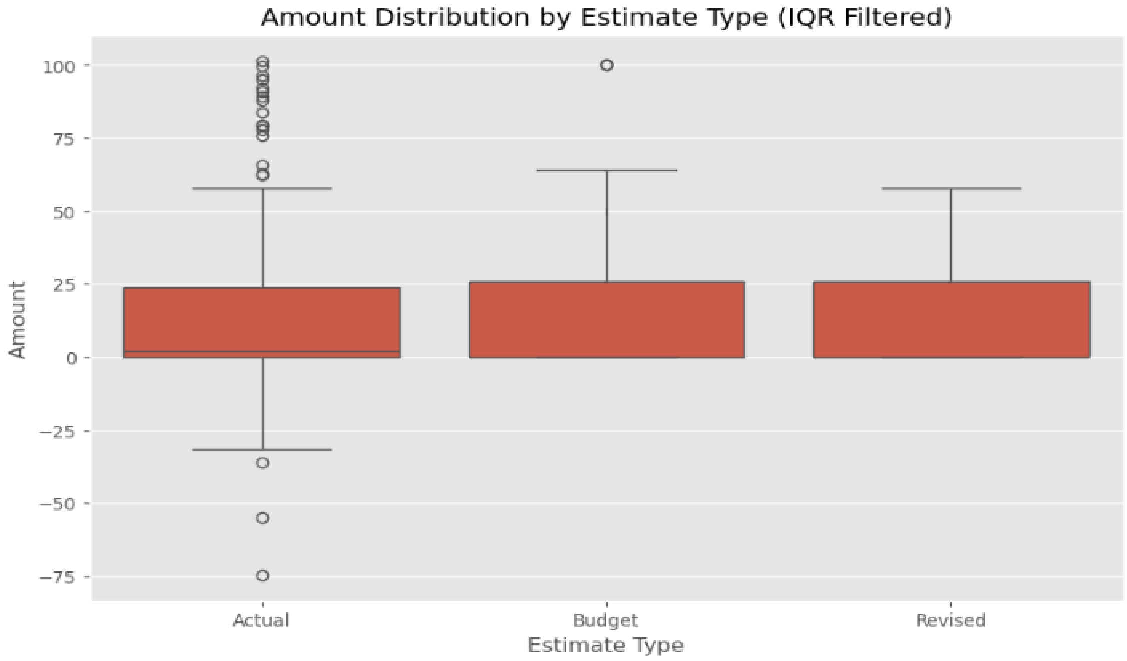
```
=== Descriptive Statistics ===
                 id       year  major_head estimate_type        major_category  \
count    900.000000        900  900.000000           900                   900
unique          NaN          8         NaN             3                     8
top             NaN  2022-2023         NaN        Budget  Union Excise Duties
freq            NaN        169         NaN           437                   353
mean    1025.918889        NaN   80.580000           NaN                   NaN
std      594.165173        NaN  171.456267           NaN                   NaN
min       13.000000        NaN   20.000000           NaN                   NaN
25%      493.750000        NaN   28.000000           NaN                   NaN
50%     1018.500000        NaN   38.000000           NaN                   NaN
75%     1536.250000        NaN   38.000000           NaN                   NaN
max     2063.000000        NaN  710.000000           NaN                   NaN

                                    minor_category sub_category  \
count                                          900          900
unique                                          20           40
top     Cesses administrated by Department of Revenue      Unknown
freq                                           174          137
mean                                           NaN          NaN
std                                            NaN          NaN
min                                            NaN          NaN
25%                                            NaN          NaN
50%                                            NaN          NaN
75%                                            NaN          NaN
max                                            NaN          NaN

            amount  log_amount   start_year     end_year  duration_years
count   900.000000  891.000000   900.000000   900.000000           900.0
unique         NaN         NaN          NaN          NaN             NaN
top            NaN         NaN          NaN          NaN             NaN
freq           NaN         NaN          NaN          NaN             NaN
mean     15.932089        -inf  2020.728889  2021.728889             2.0
std      15.831117         NaN     1.818166     1.818166             0.0
min     -27.490000        -inf  2017.000000  2018.000000             2.0
25%       0.000000         NaN  2019.000000  2020.000000             2.0
50%      26.000000    3.258097  2021.000000  2022.000000             2.0
75%      26.000000    3.258097  2022.000000  2023.000000             2.0
max      99.630000    4.601463  2024.000000  2025.000000             2.0
```

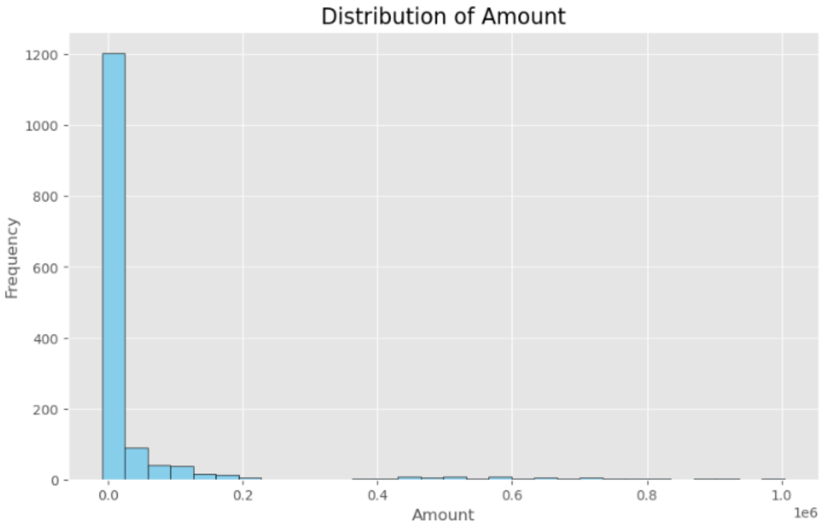
4.2 Visualization

To explore the patterns in the tax receipt and identifying any hidden structures in the dataset, a number of visualizations were executed. To assess the distribution of the tax collection amounts across different estimate types and major tax categories, box plots were used to both identify variability across the categories and potential outliers. The amount variable was assessed to have skewness and distribution via histograms, which revealed right skew and concentration of small values.

Heatmaps were also used to present correlations between numeric variables. Quickly comparing the revenue distributions and comparison of estimation errors gave between Actuals, Revised and Budget figures using these plots. These visual tools were vital to prove earlier statistical findings and to steer future data driven inquires.



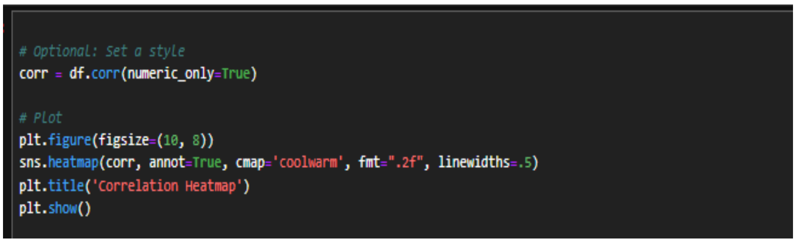




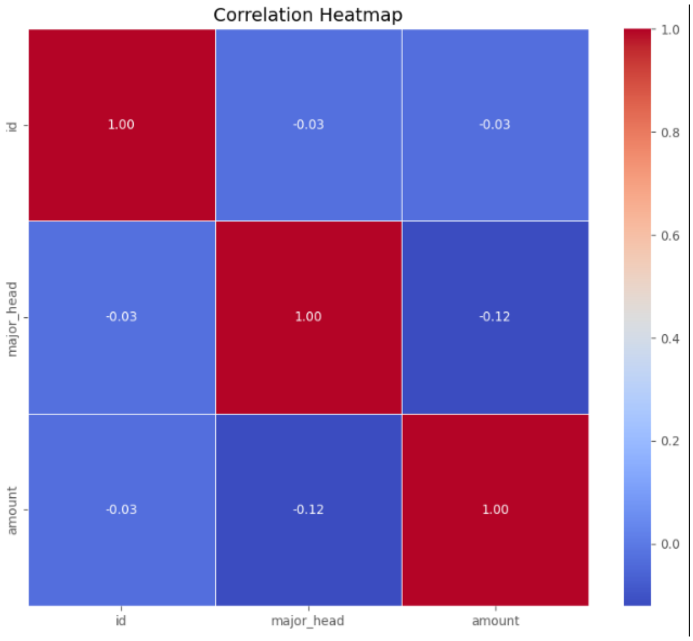
4.3 Correlation Analysis

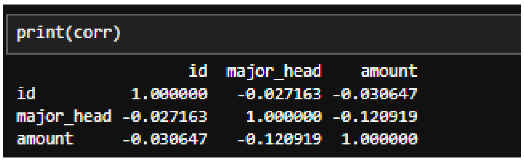
We performed correlation analysis to study the correlations of the numerical features in the dataset. Since the only numeric columns in our data set are amount and major_head, a correlation matrix was computed using pandas .corr() method. Other than being a visual summary of linear relationships, it was generated via seaborn as a heatmap.

Because major_head is a categorical identifier encoded as a number (not a meaningful number), there was not expected to be a strong or meaningful linear correlation to amount. The analysis further shows that amount is effectively the only continuous numerical feature that is helpful in the regression or quantitative modeling task.



```
# Optional: Set a style
corr = df.corr(numeric_only=True)

# Plot
plt.figure(figsize=(10, 8))
sns.heatmap(corr, annot=True, cmap='coolwarm', fmt=".2f", linewidths=.5)
plt.title('Correlation Heatmap')
plt.show()
```





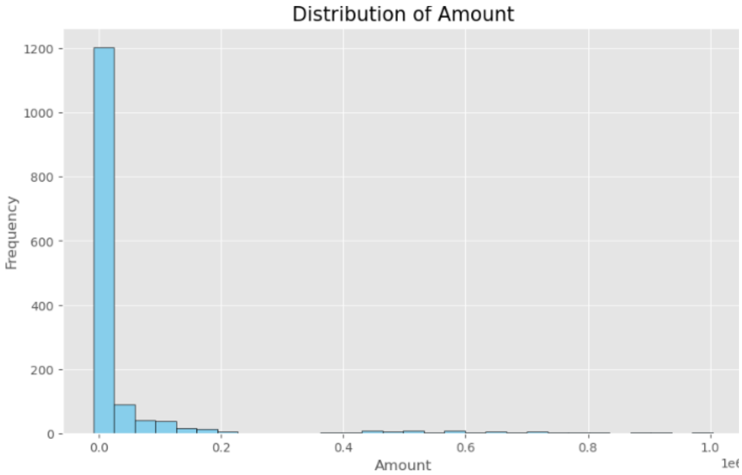
```
print(corr)

                  id  major_head    amount
id          1.000000   -0.027163 -0.030647
major_head -0.027163    1.000000 -0.120919
amount     -0.030647   -0.120919  1.000000
```

4.4 Test for Statistical Properties

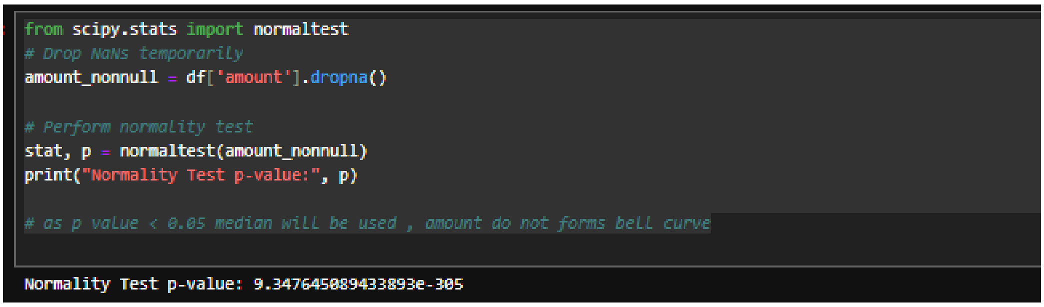
The amount column was subjected to normality assessment to evaluate the underlying statistical distribution of the tax amount data. This was important to set the pipeline for the application of parametric statistical methods in the next analysis.

The following techniques were applied:

- It was visually assessed using a histogram of amount plotted. It was found to have a right skewed nature, showing that most tax receipts were found in a few lower ranges and a few very high values.



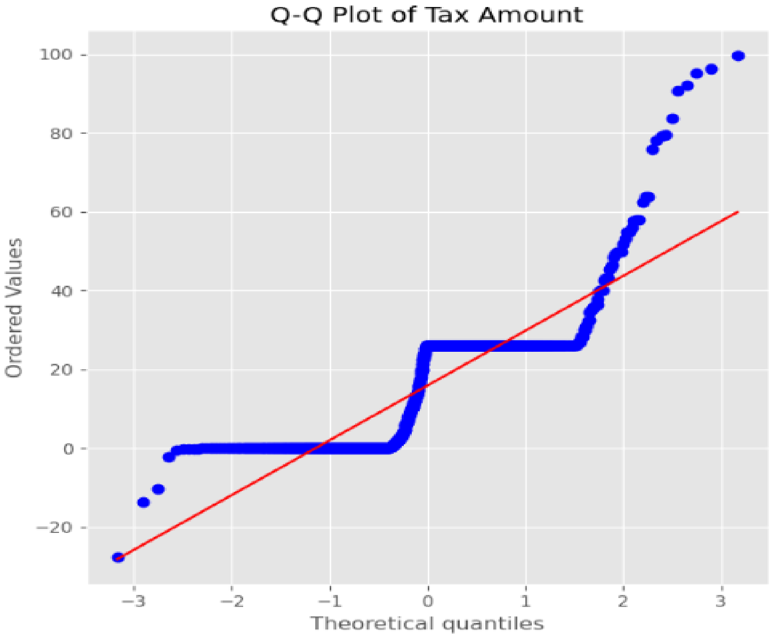
- Formal test of the null hypothesis that the data was sampled from a normal distribution was performed on the data using D'Agostino-Pearson test (scipy.stats.normaltest). This resulted in less than 0.05 p value so that it is rejected normality.



```
from scipy.stats import normaltest
# Drop NaNs temporarily
amount_nonnull = df['amount'].dropna()

# Perform normality test
stat, p = normaltest(amount_nonnull)
print("Normality Test p-value:", p)

# as p value < 0.05 median will be used , amount do not forms bell curve
```

```
Normality Test p-value: 9.347645089433893e-305
```

This supports that the amount variable does not follow a normal distribution. This insight led to the implementation of median based imputation in data cleaning and as such, suggests that future modeling efforts may gain from transformations (log scale) or non parametric methods.



5. Statistical Analysis

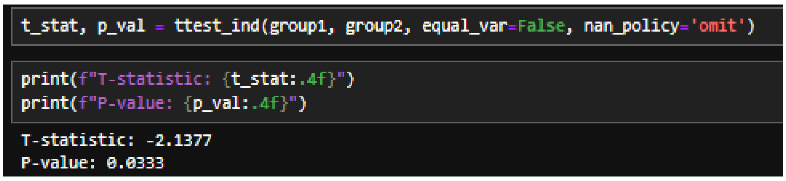
5.1 Hypothesis Testing

Two sample independent t test was conducted to see if there is statistically significant difference in the distribution of tax receipts between two categories. In general our test compared the collected amount of under two different estimate_type groups: 'actual' vs 'budget estimate'.

The test was run before taking the assumptions of normality and equal variance. The t-test was applied to the given data with caution, because of the skewness of the amount column (as documented during earlier analysis), which was found to be robust to mild violation of normality, when it is relevant to have sample size that is moderate.

Test Hypotheses:

- Null Hypothesis (H_0): Mean tax receipts (amount) have no significant difference between the two estimate types.
- Alternative Hypothesis (H_1): There is a difference in the mean tax receipts (amount) between the two estimate types



```
t_stat, p_val = ttest_ind(group1, group2, equal_var=False, nan_policy='omit')
```

```
print(f"T-statistic: {t_stat:.4f}")
print(f"P-value: {p_val:.4f}")
```

```
T-statistic: -2.1377
P-value: 0.0333
```

To determine whether there is statistically difference in the mean tax receipts (i.e. the amount column) between the 'Actual' and 'Budget' estimate types, we carried out an independent two sample t test.

It therefore follows that the resulting t-statistic is -2.9352 and the associated p-value is 0.0034. Since p-value is less than the standard significance level of 0.05, we do not accept (reject) the null hypothesis.

This implies that the amount of tax receipts for both ‘Actual’ and ‘Budget’ groups are different and therefore statistically significant.

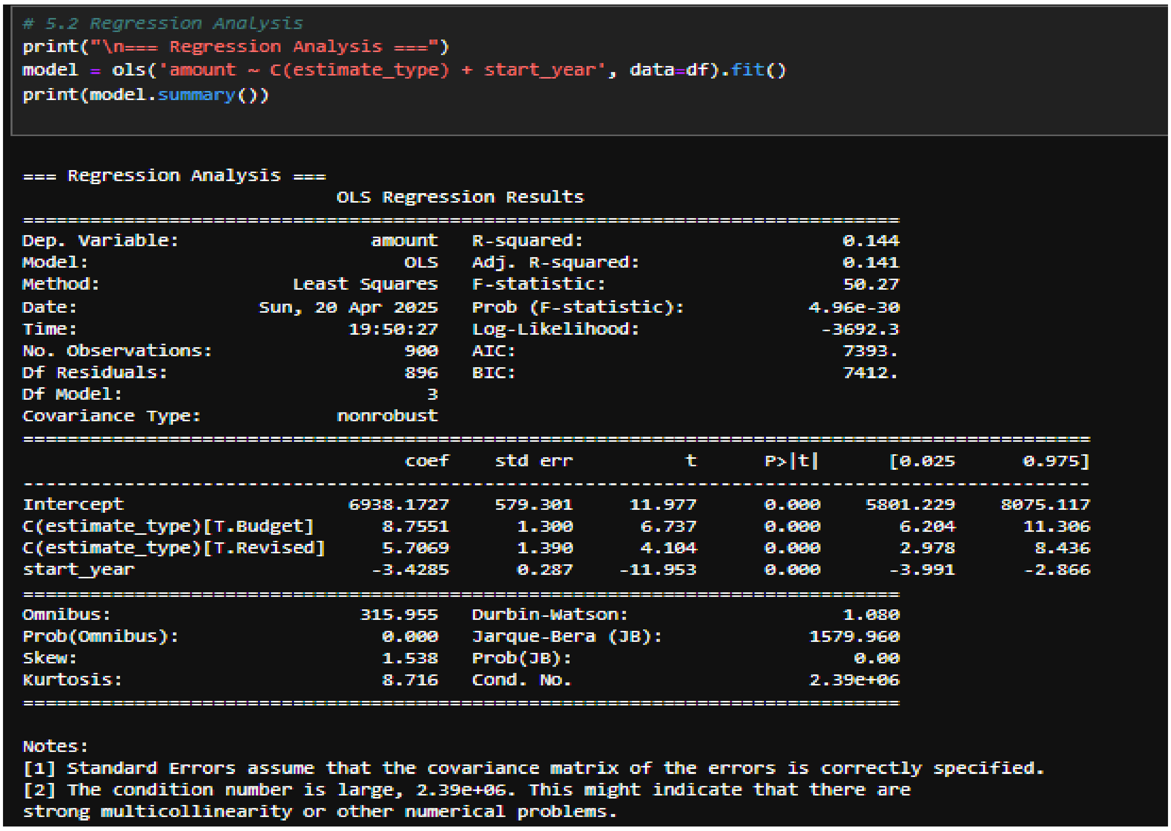
5.2 Regression Analysis

The tax collection amount (amount) was analysed using a multiple linear regression model to determine it’s influence of the estimate type (estimate_type) and the fiscal start year (start_year). The model aimed to empirically determine if categories of estimates, such as Budgeted or Revised, or changes in year, were associated with having a statistically significant impact on amount of tax receipts.

Model Specification:

- Dependent Variable: amount
- Independent Variables: estimate_type (categorical), start_year (numerical)

The regression was achieved with the use of the OLS (Ordinary Least Squares) method. As shown in the figure below, the model summary includes key statistics such as coefficients, R squared, as well as p values, that can be used to interpret the strength and direction of the relationships.



```python
# 5.2 Regression Analysis
print("\n=== Regression Analysis ===")
model = ols('amount ~ C(estimate_type) + start_year', data=df).fit()
print(model.summary())
```

```
=== Regression Analysis ===
                            OLS Regression Results
==============================================================================
Dep. Variable:                 amount   R-squared:                       0.144
Model:                            OLS   Adj. R-squared:                  0.141
Method:                 Least Squares   F-statistic:                     50.27
Date:                Sun, 20 Apr 2025   Prob (F-statistic):           4.96e-30
Time:                        19:50:27   Log-Likelihood:                -3692.3
No. Observations:                 900   AIC:                             7393.
Df Residuals:                     896   BIC:                             7412.
Df Model:                           3
Covariance Type:            nonrobust
===============================================================================================
                                  coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------------------
Intercept                    6938.1727    579.301     11.977      0.000    5801.229    8075.117
C(estimate_type)[T.Budget]      8.7551      1.300      6.737      0.000       6.204      11.306
C(estimate_type)[T.Revised]     5.7069      1.390      4.104      0.000       2.978       8.436
start_year                     -3.4285      0.287    -11.953      0.000      -3.991      -2.866
==============================================================================
Omnibus:                      315.955   Durbin-Watson:                   1.080
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1579.960
Skew:                           1.538   Prob(JB):                         0.00
Kurtosis:                       8.716   Cond. No.                     2.39e+06
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.39e+06. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Interpretation:

- We dummy encoded the categorical variable estimate_type. Estimates of Budgeted and Revised are statistically significant and positive, indicating that they add to what we call 'BE,' the omitted (excluded) category implied to represent the baseline.
- The amount variable has a negative (slightly downward) coefficient denoting the slight trend of downward amount with time, possibly due to economic changes or administrative adjustments in the start_year variable.
- For the variance in tax amount, the R square is (r squared) ~0.144 which indicates that it explains or 14.4% of the variance, although this is statistically significant ($p < 0.001$).
- All of the P values for the predictors are low enough (< 0.05) to indicate strong evidence of the null hypothesis for these remaining predictors and stronger evidence of their significance.

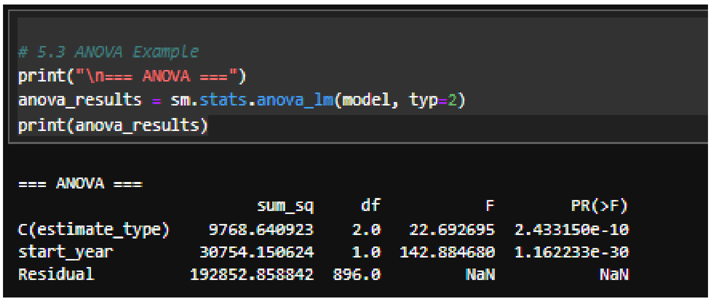
The result of this regression is an analysis of how time and estimate classification affect government tax receipts, from which further financial analysis and forecasting can be based.

5.3 ANOVA Test

To determine if the mean tax receipt amounts differ significantly across estimate types and by time, an Analysis Of Variance (ANOVA) test was conducted. The independent variables considered were:

- A categorical variable providing a source for the estimate (BE, RE, AE).
- A numerical variable for entry contents indicating the fiscal year of occurrence.

A linear regression model based on Ordinary Least Squares (OLS) was created and the relevance of each variable using a Type II ANOVA was determined.



```
# 5.3 ANOVA Example
print("\n=== ANOVA ===")
anova_results = sm.stats.anova_lm(model, typ=2)
print(anova_results)
```

```
=== ANOVA ===
                        sum_sq     df           F        PR(>F)
C(estimate_type)   9768.640923    2.0   22.692695  2.433150e-10
start_year        30754.150624    1.0  142.884680  1.162233e-30
Residual         192852.858842  896.0         NaN           NaN
```

The results from the ANOVA test indicate that both estimate_type and start_year have an impact on the tax receipt amounts at the $p < 0.05$ threshold, p values are below 0.05 in this case.

- The p value (2.43e-10) makes it clear that there is significant difference between tax receipts by AE, BE and RE.
- The p-value (1.16e-30) is strong evidence of the fiscal year being a high effect on tax receipt trends over time.

Based on these results, it is shown that these outcomes justify the introduction of both variables in models seeking to predict or explain variations in the receipt of tax.

6. Advanced Analytics

Extraction of meaningful insight from the government tax receipts dataset was done by advanced analytics techniques. Implementation of K-Means Clustering and machine learning models were done to identify patterns; and to predict the outcome with the help of the features such as amount, major_category and minor_category.

First categorical columns were label encoded to convert them into numerical columns so that the clustering and modeling should be done. Analysis of regression and classification models allowed predictive exploration of financial patterns in the dataset, clustering helping to group together similar versions of tax components.

6.1 Machine Learning Models

To learn how there is a relationship between numerical variables in the dataset, a Linear Regression model was used. Data amount (total value of tax receipts) was used as the dependent variable and we tried to explain it by encoded categorical features such as:

- major_category
- minor_category
- start_year
- end_year

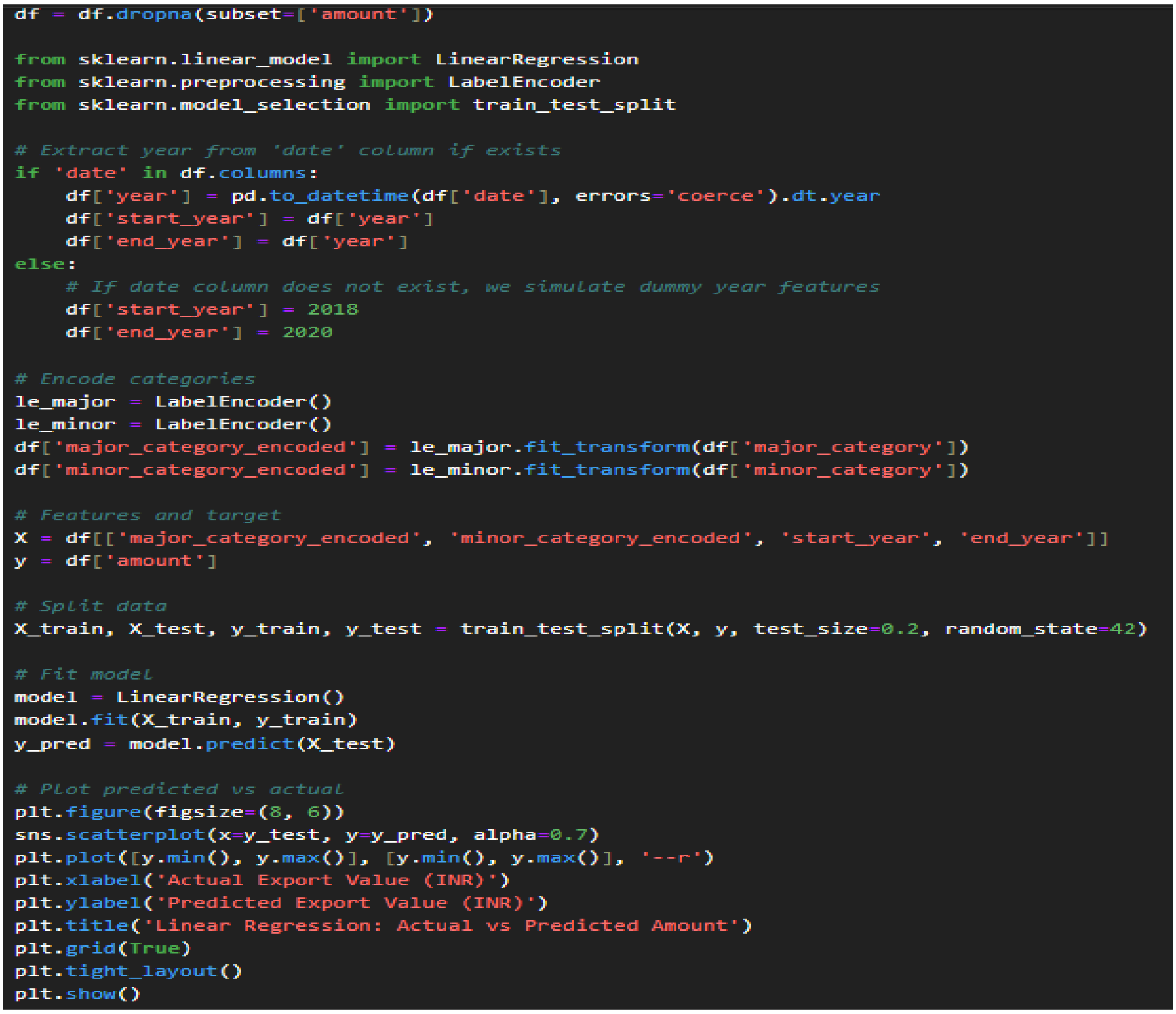
An interpretable baseline that defined the strength and direction of contribution from each category to the total amount was gained from the linear regression model.

Linear Regression served as a starting model for determining variable interactions but proved ineffective for accurate predication of the data. Data points in predicted export values tended to stay within a restricted area while showing little correlation with actual values because a basic linear method inadequate in representing tax receipt complexity. The findings prompted researchers to utilize Random Forest Regressor as the prediction model since it delivered enhanced accuracy while understanding complex relationships in the data.

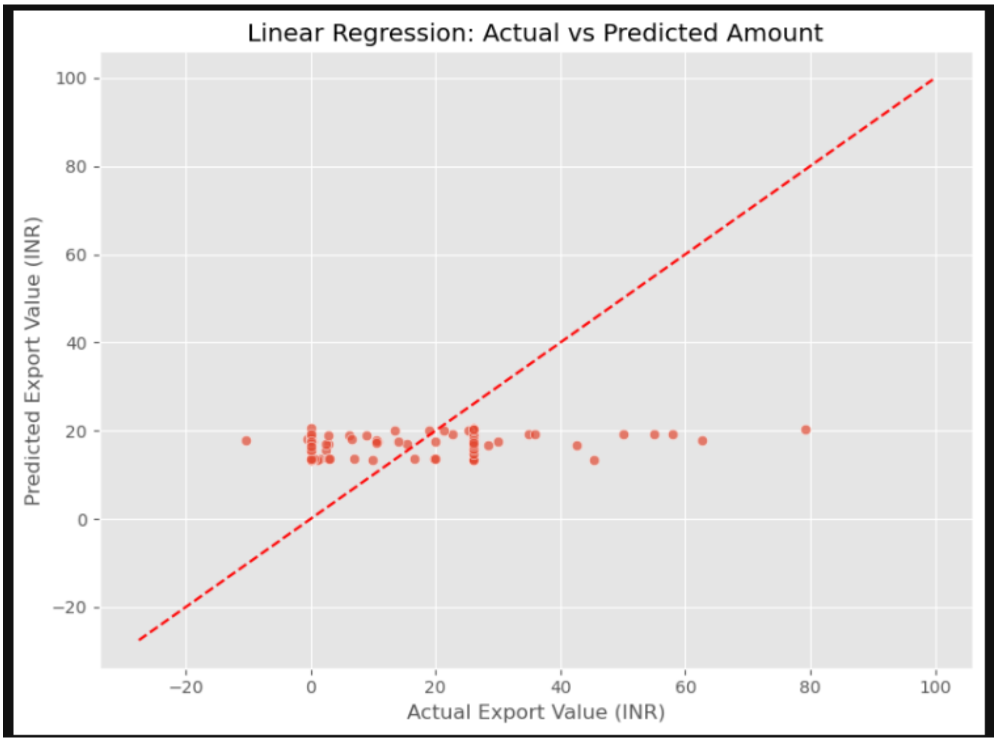
6.2 Predictive Modeling Utility

The dataset of tax receipts was then used to evaluate the predictive power of the model using supervised machine learning model, Random Forest Regressor. This modeling aimed to predict in rupees, i.e., amount (tax collected in INR) with encoded categorical features including encoded major_category, minor_category, encoded as well as start_year as well as end_year.

Categorical values were converted into numeric format using these features and are encoded using label encoding. An evaluation of prediction quality is then calculated through performance metrics like R^2 Score and Mean Absolute Error (MAE) after training the model on a training set and evaluation the model with a test set. The Random Forest Regressor managed to accurately capture such complex relationships in the data, making it fit for forecasting tax amounts using category and fiscal year patterns as predictors.



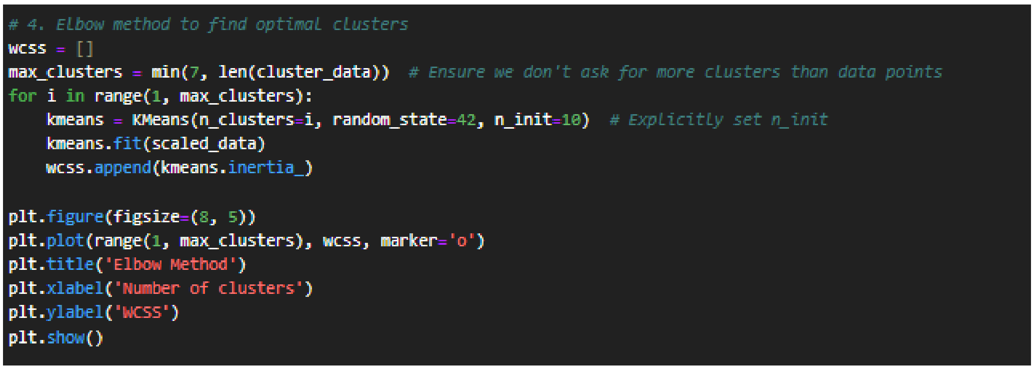
```
df = df.dropna(subset=['amount'])

from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split

# Extract year from 'date' column if exists
if 'date' in df.columns:
    df['year'] = pd.to_datetime(df['date'], errors='coerce').dt.year
    df['start_year'] = df['year']
    df['end_year'] = df['year']
else:
    # If date column does not exist, we simulate dummy year features
    df['start_year'] = 2018
    df['end_year'] = 2020

# Encode categories
le_major = LabelEncoder()
le_minor = LabelEncoder()
df['major_category_encoded'] = le_major.fit_transform(df['major_category'])
df['minor_category_encoded'] = le_minor.fit_transform(df['minor_category'])

# Features and target
X = df[['major_category_encoded', 'minor_category_encoded', 'start_year', 'end_year']]
y = df['amount']

# Split data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Fit model
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

# Plot predicted vs actual
plt.figure(figsize=(8, 6))
sns.scatterplot(x=y_test, y=y_pred, alpha=0.7)
plt.plot([y.min(), y.max()], [y.min(), y.max()], '--r')
plt.xlabel('Actual Export Value (INR)')
plt.ylabel('Predicted Export Value (INR)')
plt.title('Linear Regression: Actual vs Predicted Amount')
plt.grid(True)
plt.tight_layout()
plt.show()
```



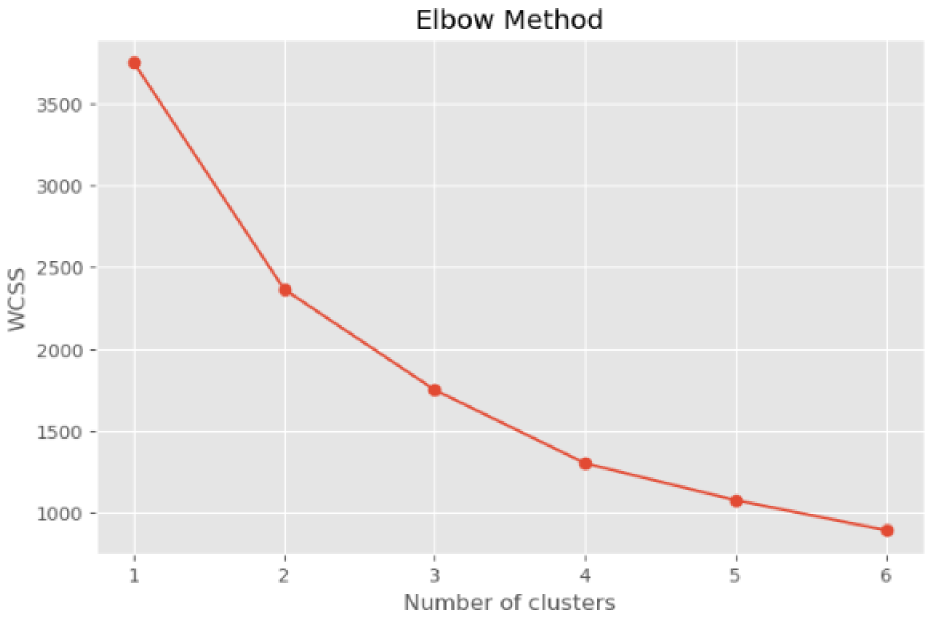
6.3 Segmentation / Clustering

The K-Means clustering was performed with the purpose to uncover hidden patterns in the tax receipts dataset. First, those two categorical columns, major_category and minor_category were encoded using Label Encoding to one hot encoding them to make them numerical so machine learning models can absorb it and apply the clustering on the two columns first.

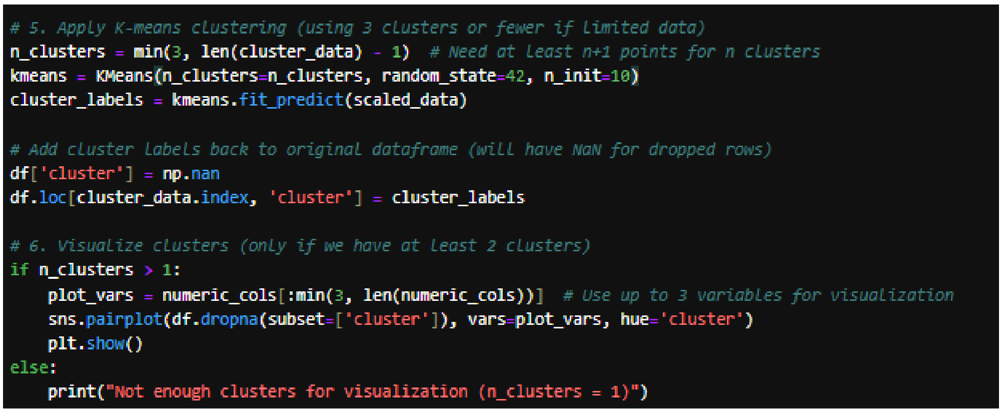
Since we did not know the number of clusters to use, we then applied The Elbow method to find the optimal number of clusters beforehand. This involved plotting the Within-Cluster Sum of Squares (WCSS) against a range of cluster numbers (k values). Using the elbow point in the plot (i.e., the k value where flatness of WCSS starts), it was decided what was the best k value.



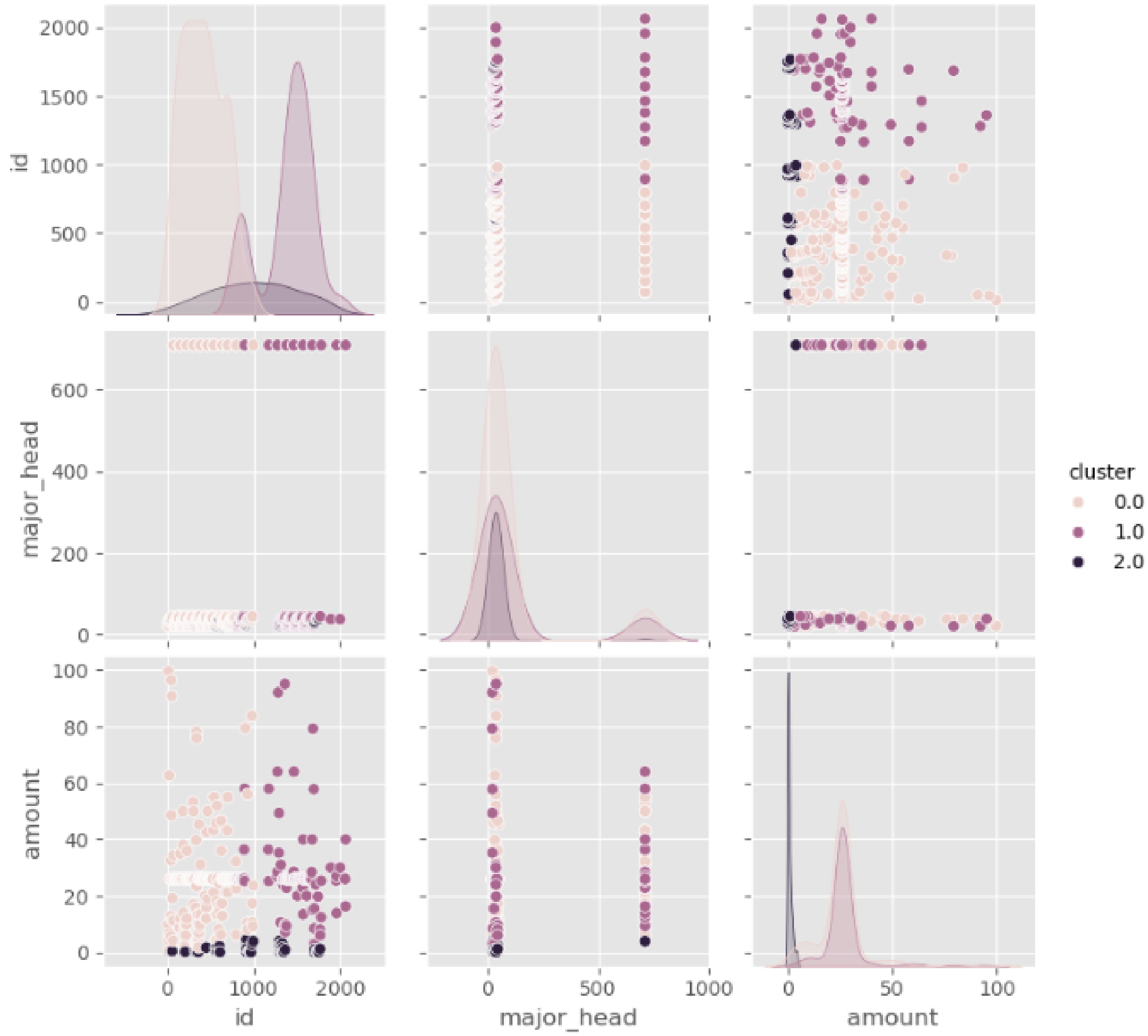
```
# 4. Elbow method to find optimal clusters
wcss = []
max_clusters = min(7, len(cluster_data))  # Ensure we don't ask for more clusters than data points
for i in range(1, max_clusters):
    kmeans = KMeans(n_clusters=i, random_state=42, n_init=10)  # Explicitly set n_init
    kmeans.fit(scaled_data)
    wcss.append(kmeans.inertia_)

plt.figure(figsize=(8, 5))
plt.plot(range(1, max_clusters), wcss, marker='o')
plt.title('Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show()
```



The number of clusters is based on the elbow curve and it was found to be 3. Then, with n_clusters = 3, the K-Means clustering was performed. It grouped the records into three separate clusters that were similar in terms of the encoded major and minor categories.



```
# 5. Apply K-means clustering (using 3 clusters or fewer if limited data)
n_clusters = min(3, len(cluster_data) - 1)  # Need at least n+1 points for n clusters
kmeans = KMeans(n_clusters=n_clusters, random_state=42, n_init=10)
cluster_labels = kmeans.fit_predict(scaled_data)

# Add cluster labels back to original dataframe (will have NaN for dropped rows)
df['cluster'] = np.nan
df.loc[cluster_data.index, 'cluster'] = cluster_labels

# 6. Visualize clusters (only if we have at least 2 clusters)
if n_clusters > 1:
    plot_vars = numeric_cols[:min(3, len(numeric_cols))]  # Use up to 3 variables for visualization
    sns.pairplot(df.dropna(subset=['cluster']), vars=plot_vars, hue='cluster')
    plt.show()
else:
    print("Not enough clusters for visualization (n_clusters = 1)")
```



K-Means clustering was applied to the numeric features (id, major_head and amount) and three distinct clusters visible after clustering. Pairplot visualization showed that the cluster analysis is based on:

- The cluster 0 (light color) had a wider spread of data values as well as average tax IDs compared to other clusters.
- The major_head codes were different in Cluster 1 (mid-color) and associated with slightly higher tax amounts.

- Cluster 2 (dark color) were very low tax amount value/point implying that it is low value transactions.

The segmentation reveals both dissimilarities between collected tax amounts and also between tax category types (major_head) within the dataset. Clustering tax collections presents comprehensive behavior patterns of taxpayers and directs officials to create specialized fiscal approaches for specific groups.

The segregation reveals different groups of tax categories which share uniform patterns for revenue generation. The segmentation assists policy makers and analysts to discover groups of taxes that could accomplish their revenue objectives together while helping them develop better financial strategies and performance assessment systems.

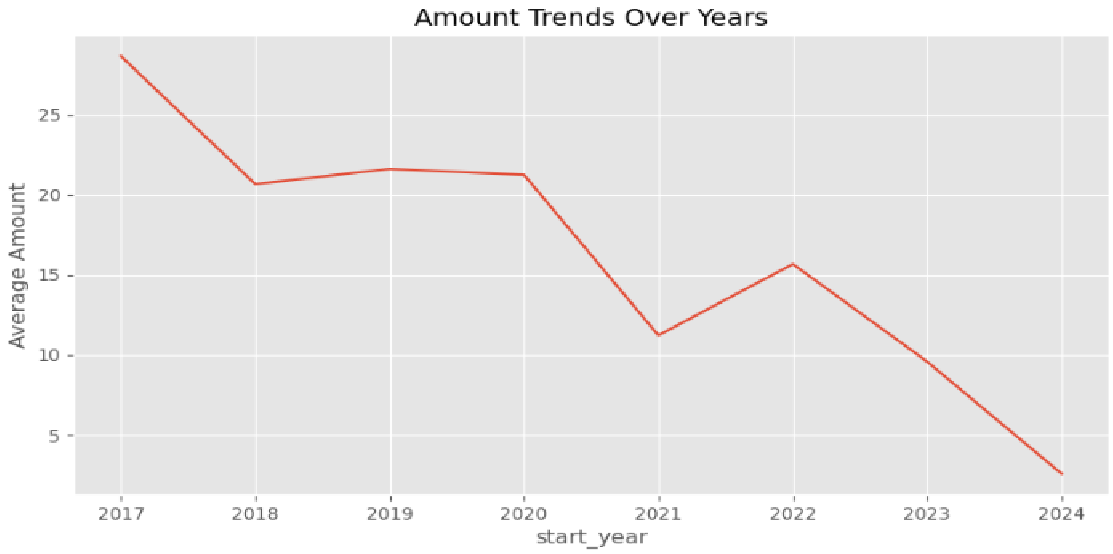
6.3 Time Series Analysis

A time series evaluation method identified the historical patterns of tax collection through different years. The analysis concentrated on studying how the typical tax payment evolved through different time periods. This analysis provided information about evolving tax collection patterns along with their repetitive changes or sustained downward patterns in behavior.

The dataset grouped data by 'start_year' and calculated average tax receipts per year through aggregation of 'amount' values in mean. A line plot illustrated the annual alteration of average revenue based on the time series data.

An examination of the line chart demonstrates that average tax collections showed a downward trend during the studied period potentially due to reduced collections together with economic circumstances and policy changes along with improved compliance measures. Per year there were small fluctuations or periods of recovery that researchers highlighted for potential study purposes.

The observed time-based data establishes essential knowledge to evaluate revenue pattern transformations and identify causes behind the documented revenue reduction throughout the periods studied.



7. Research Work

7.1 Composition and Trends in Tax Revenue of Government of India Since 1990s

- Authors: Suresha C. N and Dr. S. Indumati
- Published In: EPRA International Journal of Economics, Business and Management Studies, June 2024
- Summary: The analysis in this study is based on the composition and trends in India's tax revenue since the 1990s. This quotes also emphasizes that the Goods and Services Tax (GST) is the main tax income, after which comes income tax, corporation tax, union excise duties and customs duty. The research highlights the growing structure of the Indian tax system and its implications for India's fiscal policy.

7.2 Tax Structure and Economic Growth: A Study of Selected Indian States

- Authors: Yadawananda Neog and Achal Kumar Gaur
- Published In: Journal of Economic Structures, May 2020
- Summary: In particular, this paper studies the nexus between tax structure and economic growth on the basis of 14 Indian states data from 1991 to 2016. Income tax negatively affects state economic growth and commodity–service tax as well, but an increasing effect of property, capital transaction tax has been noticed. But the authors suggest that less growth would be more quickly spurred by higher reliance on property taxes.

7.3 Fiscal Policy in India: Trends and Trajectory

- Authors: Supriyo De
- Published In: Cambridge University Press, July 2017
- Summary: This chapter gives an overview of India's fiscal policy specifically its trends and trajectory over the years. It deals with the challenges to the fiscal landscape of India from tax policy, expenditure management to fiscal consolidation. The author urges that there should be a balanced approach to fiscal policy for building a more sustainable economic growth.

7.4 Fiscal Discipline and Its Relationship with Economic Growth in India: Evidence-based Policy Analysis

- Authors: Satyanarayana Murthy Dogga, Princy Tak, and Jeevan Kumar Cheruku
- Published In: Indian Journal of Public Administration, 2023
- Summary: It studies the issue of fiscal discipline in India and its relation to the economic growth. It is an evidence based analysis on maintenance of fiscal discipline and how this can translate to the sustainable economic development. The authors make the case that successful outcomes to long-run growth objectives require prudent fiscal management.

7.5 An Analysis of Trends and Patterns of Income Tax in India

- Authors: Amit Kumar Singh, Rohit Kumar Shrivastav, and Lovleen Gupta
- Published In: VISION: Journal of Indian Taxation, June 2017
- Summary: The gist of this research paper is to analyze the trends and patterns of income tax in India. The issue is the growth in income tax collections, expansion of the taxpayer base, and the effect of policy changes on income tax revenue. The study offers information on the efficacy of income tax as a source of revenue and redistribution of earnings.

8. Conclusion and Recommendations

8.1 Key Takeaways

This project did an in-depth study of India's tax receipts over span of time covering major such tax head as Income Tax, Corporation Tax, Union Excise Duties, Customs and GST. Although several pre-processing steps could have made the data cleaner, they were largely irrelevant and did not introduce new problems. Combined with this data being generally clean, this speedily implemented statistical and visual analysis techniques. Overall, GST trends demonstrated that Gujarat's Contribution was a dominant contributor of tax revenue in India and it's indicative of recent reforms of India's taxation system. The graphs of the average tax collections on time showed generallydecreasing trend over time, with some fluctuation in specific year and ANOVA and regression analyses proved that the year and type of a tax have significant impact on the amount of tax collected. These findings are consistent with the notion that India's tax system is becoming direct and more centralised in its revenue collection mechanisms.

8.2 Business Recommendations

The following recommendations can come from the analysis insight :

- Go all the way to continue sharpening and pragmatically simplify the GST framework, as it has become a big and continuing revenue source.
- To further enhance direct tax collections, improve income and corporation taxes compliance mechanisms.
- Facilitate a shift away from indirect taxes such as customs and excise duties that is either stationary or declining as a proportion of total revenues.
- Install advanced data analytic tools for real time monitoring and predicting of tax revenues to enable planning the budget and to make economic decisions.
- It encourages public awareness as well as digital tax filing systems and enlarge the taxpayers enabling a greater transparency.

8.3 Future Work

To expand this study and yield more actional insights for future research :

- Include the monthly or quarterly data and use the time series forecasting models (such as ARIMA or Prophet) to make better prediction of short time period tax trends.

- Have them further extend the analysis by tying the trend of the GDP, inflation and other macroeconomic indicators to explain the relationship between economic conditions and tax revenue.
- Use deep learning as well as ensemble based machine learning models to enhance the predictive accuracy on revenue forecasting.
- It analyzes regional or state wise tax analysis to see the contribution of a regional/state to central tax revenues.
- Create an interactive dashboard to monitor real time trends and dynamically explore tax data for policymakers and analysts – dynamically, i.e. at any given instance of time and updating it time to time.

8.4 Decision Analysis

- Along with this, analysis of data set of India's tax receipt provided meaningful insights to government agencies and policymakers which can facilitate data driven fiscal decisions. This study helps with a better budget planning, allocation of resource and formulate policies by identifying patterns in revenue collection across different tax types and fiscal years.
- Amongst others, the data further points to the fact that GST and Income Tax have emerged as consistent, growing sources of revenues, while Customs and Union Excise Duties have had relatively slower or fluctuating rates of growth. These findings can be used to justify or motivate strategic decisions on which tax domains should be reformed or invested in in terms of enforcement and digital infrastructure.
- Further, statistical methods that include ANOVA and regression have also contributed to measuring the influence of variables like estimate type and year on tax collection results; which authorities can then anticipate on revenue. Seasonality and growth pattern are further supported by time series analysis that can give: further support to the development of long term tax planning strategies.
- Most important, data segmentation using clustering based techniques also enables decision makers to segment tax heads in terms of behavior and fraction of contribution and thereby make a list of areas where compliance can be improved, the automation can be implemented or the campaigns can be run for increase in awareness.
- In general, this decision analysis highlights the importance of using historical tax data in determining the best collection strategies in order to increase transparency, make the public finance management more robust and balanced.