

```
import pandas as pd

file_path = "/content/DS_Python_Assignment.xlsx"
data = pd.read_excel(file_path)

print("Dataset loaded successfully!")
data.info()

Dataset loaded successfully!
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Columns: 117 entries, CUSTOMERID to MDAYSHOPPER_L4Y
dtypes: float64(49), int64(65), object(3)
memory usage: 8.9+ MB
```

List Down all the columns with missing values bold text

```
missing_values = data.isnull().sum()
missing_columns = missing_values[missing_values > 0]
print("Columns with missing values:")
print(missing_columns)
```

Columns with missing values:	
STATE	59
TENURE_IN_MONTHS	66
CLOSESTSTOREDISTANCE	1545
AGE	5945
INCOME	3259
LENGTH OF RESIDENCE	2872
NUMBER OF PERSONS IN LIVING UNIT	2872
NUMBER OF ADULTS IN LIVING UNIT	2872
MOSAIC	1868
CAPE: AGE: POP: MEDIAN AGE	2872
CAPE: AGE: POP: % 0-17	2872
CAPE: AGE: POP: % 18-99+	2872
CAPE: AGE: POP: % 65-99+	2872
CAPE: ETHNIC: POP: % WHITE ONLY	2872
CAPE: ETHNIC: POP: % BLACK ONLY	2872
CAPE: ETHNIC: POP: % ASIAN ONLY	2872
CAPE: ETHNIC: POP: % HISPANIC	2872
CAPE: DENSITY: PERSONS PER HH FOR POP IN HH	2872
CAPE: HHSIZE: HH: AVERAGE HOUSEHOLD SIZE	2872
CAPE: TYP: HH: % MARRIED COUPLE FAMILY	2872
CAPE: CHILD: HH: % WITH PERSONS LT18	2872
CAPE: CHILD: HH: % MARR COUPLE FAMW- PERSONS LT18	2872
CAPE: CHILD: HH: % MARR COUPLE FAMW-0 PERSONS LT18	2872
CAPE: LANG: HH: % SPANISH SPEAKING	2872

CAPE: EDUC: POP25+: MEDIAN EDUCATION ATTAINED	2872
CAPE: HOMVAL: OOHU: MEDIAN HOME VALUE	2872
CAPE: BUILT: HU: MEDIAN HOUSING UNIT AGE	2872
CAPE: TENANCY: OCCHU: % OWNER OCCUPIED	2872
CAPE: TENANCY: OCCHU: % RENTER OCCUPIED	2872
CAPE: EDUC: ISPSA	2872
CAPE: EDUC: ISPSA DECILE	2872
CAPE: INC: FAMILY INC STATE DECILE	2872
CAPE: INC: HH: MEDIAN FAMILY HOUSEHOLD INCOME	2872

dtype: int64

Categorize columns based on their data types

```
numeric_columns =
data.select_dtypes(include=['number']).columns.tolist()
categorical_columns =
data.select_dtypes(include=['object']).columns.tolist()

print("Numeric Columns:", numeric_columns)
print("Categorical Columns:", categorical_columns)
```

```
Numeric Columns: ['CUSTOMERID', 'LCPCOUNT', 'TENURE_IN_MONTHS',
'CLOSESTSTOREDISTANCE', 'FEMALE', 'AGE', 'HS_DIPLOMA', 'SOME_COLLEGE',
'BACH_GRAD_DEG', 'LT_HS_DIPLOMA', 'MARRIED', 'MNGMNT_OFFICEADMIN',
'TECH_PROF', 'SALES_JOB', 'BLUE_COLLAR', 'FARMER', 'RETIRED', 'SFDU',
'MFDU', 'HOMEOWNER', 'INCOME', 'MAIL_RESP_MULTI', 'MAIL_RESP_SINGLE',
'LENGTH OF RESIDENCE', 'NUMBER OF PERSONS IN LIVING UNIT', 'NUMBER OF
ADULTS IN LIVING UNIT', 'METRO', 'URBAN', 'MOR BANK: UPSCALE
MERCHANDISE BUYER', 'MOR BANK: MALE MERCHANDISE BUYER', 'MOR BANK:
FEMALE MERCHANDISE BUYER', 'MOR BANK: CRAFTS-HOBBY MERCHANDISE BUYER',
'MOR BANK: GARDENING-FARMING BUYER', 'MOR BANK: BOOK BUYER', 'MOR
BANK: COLLECT-SPECIAL FOODS BUYER', 'MOR BANK: GIFTS AND GADGETS
BUYER', 'MOR BANK: GENERAL MERCHANDISE BUYER', 'MOR BANK: FAMILY AND
GENERAL MAGAZINE', 'MOR BANK: FEMALE ORIENTED MAGAZINE', 'MOR BANK:
MALE SPORTS MAGAZINE', 'MOR BANK: RELIGIOUS MAGAZINE', 'MOR BANK:
GARDENING-FARMING MAGAZINE', 'MOR BANK: CULINARY INTERESTS MAGAZINE',
'MOR BANK: HEALTH AND FITNESS MAGAZINE', 'MOR BANK: DO-IT-
YOURSELFERS', 'MOR BANK: NEWS AND FINANCIAL', 'MOR BANK: PHOTOGRAPHY',
'MOR BANK: OPPORTUNITY SEEKERS AND CE', 'MOR BANK: RELIGIOUS
CONTRIBUTOR', 'MOR BANK: POLITICAL CONTRIBUTOR', 'MOR BANK: HEALTH AND
INSTITUTION CONTRIBUTOR', 'MOR BANK: GENERAL CONTRIBUTOR', 'MOR BANK:
MISCELLANEOUS', 'MOR BANK: ODDS AND ENDS', 'MOR BANK: DEDUPED CATEGORY
HIT COUNT', 'MOR BANK: NON-DEDUPED CATEGORY HIT COUNT', 'MORTGAGE-HOME
PURCHASE: HOME PURCHASE PRICE', 'CHILDREN', 'CAPE: AGE: POP: MEDIAN
AGE', 'CAPE: AGE: POP: % 0-17', 'CAPE: AGE: POP: % 18-99+', 'CAPE:
AGE: POP: % 65-99+', 'CAPE: ETHNIC: POP: % WHITE ONLY', 'CAPE: ETHNIC:
POP: % BLACK ONLY', 'CAPE: ETHNIC: POP: % ASIAN ONLY', 'CAPE: ETHNIC:
POP: % HISPANIC', 'CAPE: DENSITY: PERSONS PER HH FOR POP IN HH',
```

```
'CAPE: HHSIZE: HH: AVERAGE HOUSEHOLD SIZE', 'CAPE: TYP: HH: % MARRIED
COUPLE FAMILY', 'CAPE: CHILD: HH: % WITH PERSONS LT18', 'CAPE: CHILD:
HH: % MARR COUPLE FAMW- PERSONS LT18', 'CAPE: CHILD: HH: % MARR COUPLE
FAMW-0 PERSONS LT18', 'CAPE: LANG: HH: % SPANISH SPEAKING', 'CAPE:
EDUC: POP25+: MEDIAN EDUCATION ATTAINED', 'CAPE: HOMVAL: 00HU: MEDIAN
HOME VALUE', 'CAPE: BUILT: HU: MEDIAN HOUSING UNIT AGE', 'CAPE:
TENANCY: OCCHU: % OWNER OCCUPIED', 'CAPE: TENANCY: OCCHU: % RENTER
OCCUPIED', 'CAPE: EDUC: ISPSA', 'CAPE: EDUC: ISPSA DECILE', 'CAPE:
INC: FAMILY INC STATE DECILE', 'CAPE: INC: HH: MEDIAN FAMILY HOUSEHOLD
INCOME', 'FREQUENCY', 'QUANTITY', 'TOTALSALES', 'FREQUENCY_2Y',
'QUANTITY_2Y', 'TOTALSALES_2Y', '1-Engagement Spend', '2-Wedding Bands
Spend', '3-Fashion Diamonds Spend', '4-Fashion Jewelry Spend', '5-
Close Out Spend', '6-Promotional Items Spend', '7-Cost Only Spend',
'8-Marketing Premium SKUs Spend', '9-Repairs & Appraisals Spend', '10-
Pre Owned Spend', '11-Watches Spend', '12-Misc Merchandise Spend',
'15-Store Events Spend', '16-Single Stone Jewelry Spend',
'MDAYREV_L1Y', 'MDAYREV_L2Y', 'MDAYREV_L3Y', 'MDAYREV_L4Y',
'MDAYQTY_L1Y', 'MDAYQTY_L2Y', 'MDAYQTY_L3Y', 'MDAYQTY_L4Y',
'MDAYSHOPPER_L1Y', 'MDAYSHOPPER_L2Y', 'MDAYSHOPPER_L3Y',
'MDAYSHOPPER_L4Y']
Categorical Columns: ['STATE', 'PRIVATELABELTENDERFLAG', 'MOAIC']
```

List the columns with duplicate a)Remove them b)Print before and after

```
print("Dataset Before Removing Duplicate Columns:")
print(data.head())

duplicate_columns = data.columns[data.columns.duplicated()].tolist()
print("\nDuplicate Columns Identified:")
print(duplicate_columns)
```

```
data_cleaned = data.loc[:, ~data.columns.duplicated()]
```

```
print("\nDataset After Removing Duplicate Columns:")
print(data_cleaned.head())
```

Dataset Before Removing Duplicate Columns:

	CUSTOMERID	STATE	LCPCOUNT	PRIVATELABELTENDERFLAG	TENURE_IN_MONTHS
0	5001	TX	1	N	-9.0
1	5002	OH	0	Y	9.0

2	5003	TX	0	N	12.0
3	5004	TN	0	N	-1.0
4	5005	TX	0	N	16.0

	CLOSESTSTOREDISTANCE	FEMALE	AGE	HS_DIPLOMA	SOME_COLLEGE	...	\
0	NaN	0	NaN	0	0	...	
1	8.728943	0	NaN	0	0	...	
2	NaN	0	NaN	0	0	...	
3	NaN	0	NaN	0	0	...	
4	NaN	0	NaN	0	0	...	

	MDAYREV_L3Y	MDAYREV_L4Y	MDAYQTY_L1Y	MDAYQTY_L2Y	MDAYQTY_L3Y	\
0	0.0	0.0	0	0	0	
1	0.0	0.0	0	0	0	
2	0.0	0.0	0	0	0	
3	0.0	0.0	0	0	0	
4	0.0	0.0	0	0	0	

	MDAYQTY_L4Y	MDAYSHOPPER_L1Y	MDAYSHOPPER_L2Y	MDAYSHOPPER_L3Y	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	

	MDAYSHOPPER_L4Y
0	0
1	0
2	0
3	0
4	0

[5 rows x 117 columns]

Duplicate Columns Identified:
[]

Dataset After Removing Duplicate Columns:

	CUSTOMERID	STATE	LCPCOUNT	PRIVATELABELTENDERFLAG	TENURE_IN_MONTHS
0	5001	TX	1	N	-9.0
1	5002	OH	0	Y	9.0
2	5003	TX	0	N	12.0
3	5004	TN	0	N	-1.0

4	5005	TX	0		N	16.0	
	CLOSESTSTOREDISTANCE		FEMALE	AGE	HS_DIPLOMA	SOME_COLLEGE	...
0		NaN	0	NaN	0	0	...
1		8.728943	0	NaN	0	0	...
2		NaN	0	NaN	0	0	...
3		NaN	0	NaN	0	0	...
4		NaN	0	NaN	0	0	...
	MDAYREV_L3Y	MDAYREV_L4Y	MDAYQTY_L1Y	MDAYQTY_L2Y	MDAYQTY_L3Y		\
0	0.0	0.0	0	0	0		
1	0.0	0.0	0	0	0		
2	0.0	0.0	0	0	0		
3	0.0	0.0	0	0	0		
4	0.0	0.0	0	0	0		
	MDAYQTY_L4Y	MDAYSHOPPER_L1Y	MDAYSHOPPER_L2Y	MDAYSHOPPER_L3Y			\
0	0	0	0	0			
1	0	0	0	0			
2	0	0	0	0			
3	0	0	0	0			
4	0	0	0	0			
	MDAYSHOPPER_L4Y						
0	0						
1	0						
2	0						
3	0						
4	0						

[5 rows x 117 columns]

List the constants columns a)Remove them b)Print before and after

```
print("Dataset Before Removing constant Columns:")
print(data_cleaned.head())

constant_columns = [col for col in data.columns if data[col].nunique()
== 1]
print("Constant Columns:", constant_columns)

data.drop(columns=constant_columns, inplace=True)

print("\nDataset After Removing constants Columns:")
print(data_cleaned.head())
```

Dataset Before Removing constant Columns:

	CUSTOMERID	STATE	LCPCOUNT	PRIVATELABEL	TENDERFLAG	TENURE_IN_MONTHS
\						
0	5001	TX	1		N	-9.0
1	5002	OH	0		Y	9.0
2	5003	TX	0		N	12.0
3	5004	TN	0		N	-1.0
4	5005	TX	0		N	16.0

	CLOSESTSTORE	DISTANCE	FEMALE	AGE	HS_DIPLOMA	SOME_COLLEGE	...	\
0		NaN	0	NaN	0	0	...	
1		8.728943	0	NaN	0	0	...	
2		NaN	0	NaN	0	0	...	
3		NaN	0	NaN	0	0	...	
4		NaN	0	NaN	0	0	...	

	MDAYREV_L3Y	MDAYREV_L4Y	MDAYQTY_L1Y	MDAYQTY_L2Y	MDAYQTY_L3Y	\
0	0.0	0.0	0	0	0	
1	0.0	0.0	0	0	0	
2	0.0	0.0	0	0	0	
3	0.0	0.0	0	0	0	
4	0.0	0.0	0	0	0	

	MDAYQTY_L4Y	MDAYSHOPPER_L1Y	MDAYSHOPPER_L2Y	MDAYSHOPPER_L3Y	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	

	MDAYSHOPPER_L4Y
0	0
1	0
2	0
3	0
4	0

[5 rows x 117 columns]

Constant Columns: ['MARRIED', 'CHILDREN', '7-Cost Only Spend']

Dataset After Removing constants Columns:

	CUSTOMERID	STATE	LCPCOUNT	PRIVATELABEL	TENDERFLAG	TENURE_IN_MONTHS
\						
0	5001	TX	1		N	-9.0
1	5002	OH	0		Y	9.0

2	5003	TX	0		N	12.0
3	5004	TN	0		N	-1.0
4	5005	TX	0		N	16.0

	CLOSESTSTOREDISTANCE	FEMALE	AGE	HS_DIPLOMA	SOME_COLLEGE	...	\
0	NaN	0	NaN	0	0	...	
1	8.728943	0	NaN	0	0	...	
2	NaN	0	NaN	0	0	...	
3	NaN	0	NaN	0	0	...	
4	NaN	0	NaN	0	0	...	

	MDAYREV_L3Y	MDAYREV_L4Y	MDAYQTY_L1Y	MDAYQTY_L2Y	MDAYQTY_L3Y	\
0	0.0	0.0	0	0	0	
1	0.0	0.0	0	0	0	
2	0.0	0.0	0	0	0	
3	0.0	0.0	0	0	0	
4	0.0	0.0	0	0	0	

	MDAYQTY_L4Y	MDAYSHOPPER_L1Y	MDAYSHOPPER_L2Y	MDAYSHOPPER_L3Y	\
0	0	0	0	0	
1	0	0	0	0	
2	0	0	0	0	
3	0	0	0	0	
4	0	0	0	0	

	MDAYSHOPPER_L4Y
0	0
1	0
2	0
3	0
4	0

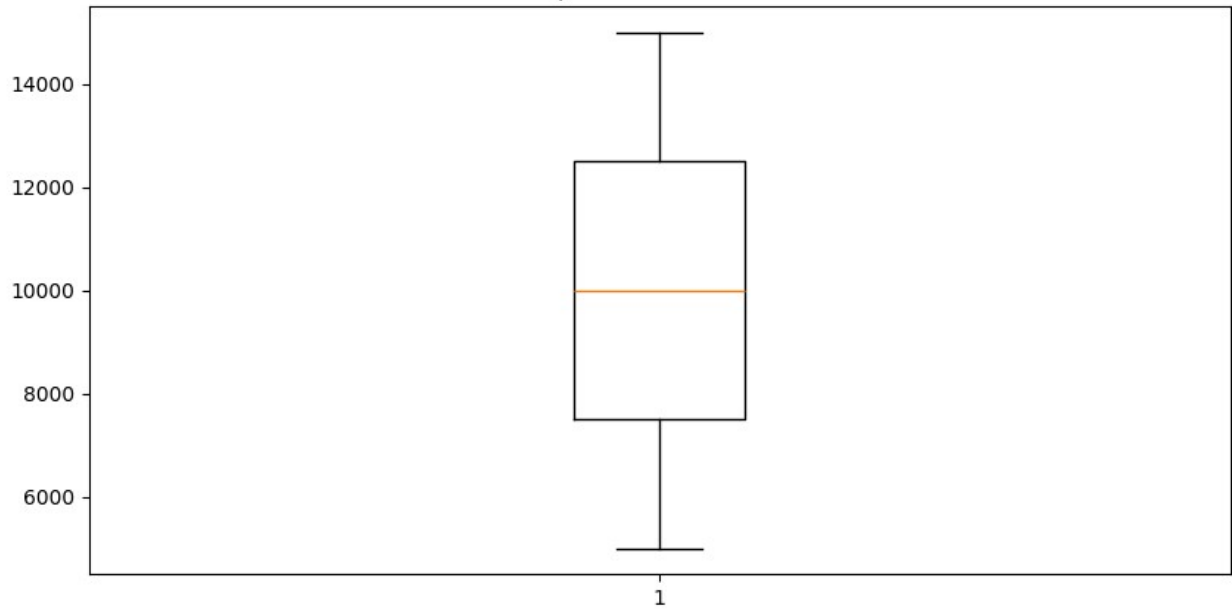
[5 rows x 117 columns]

boxplots for numeric columns

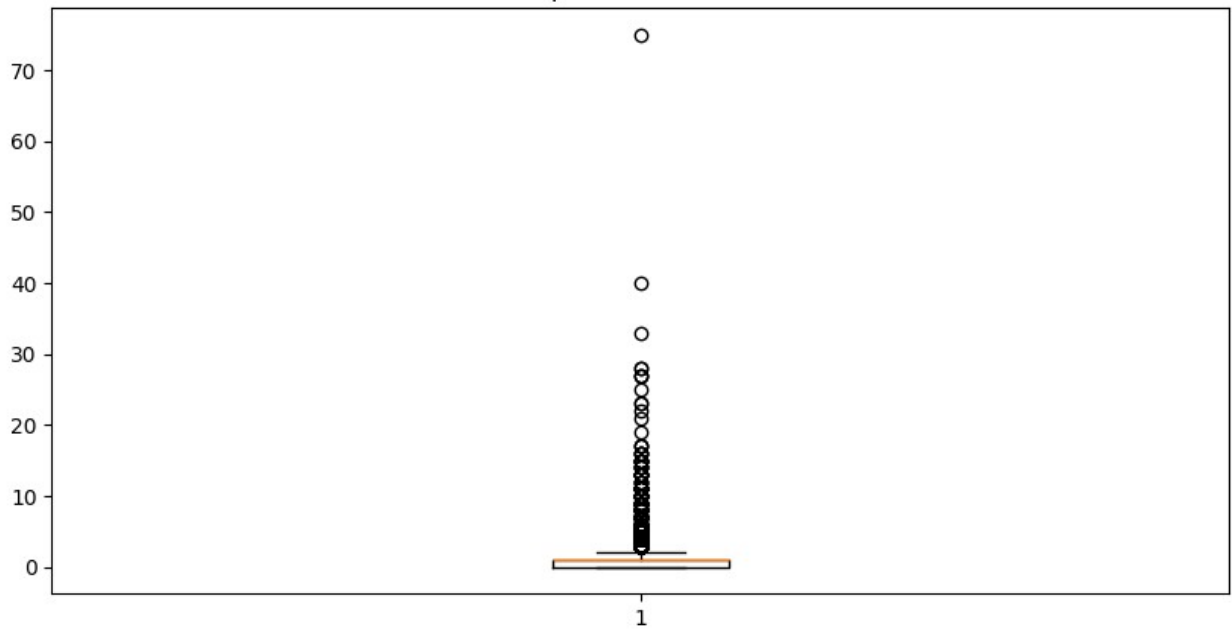
```
import matplotlib.pyplot as plt

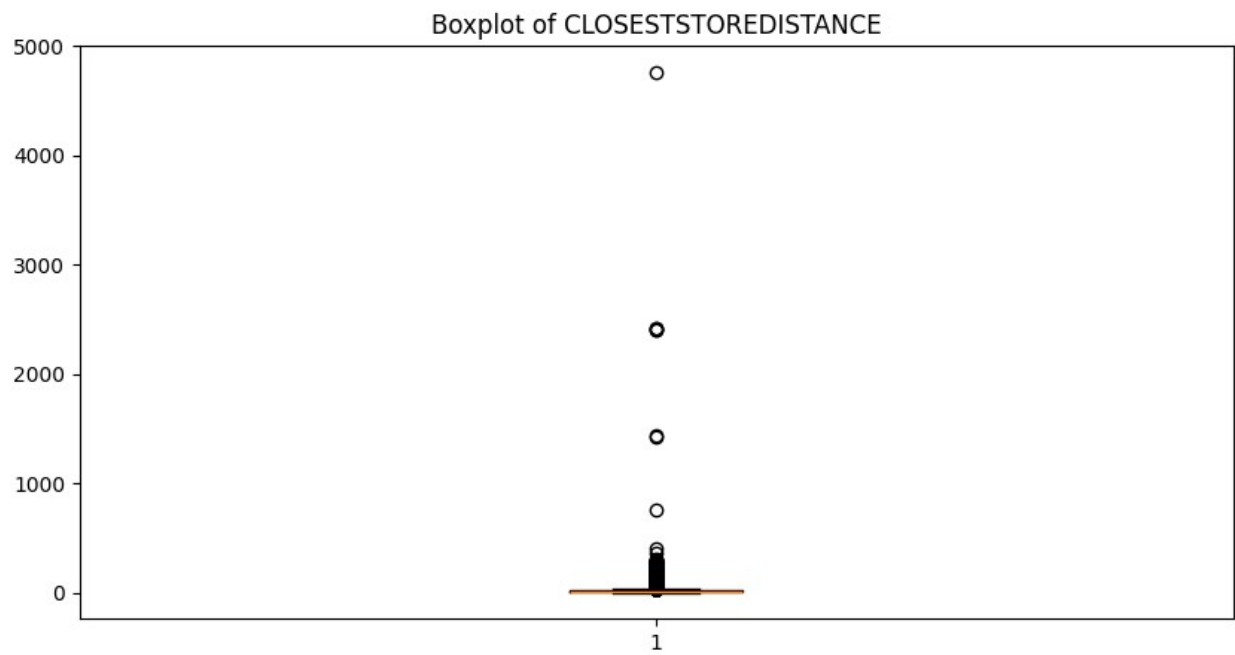
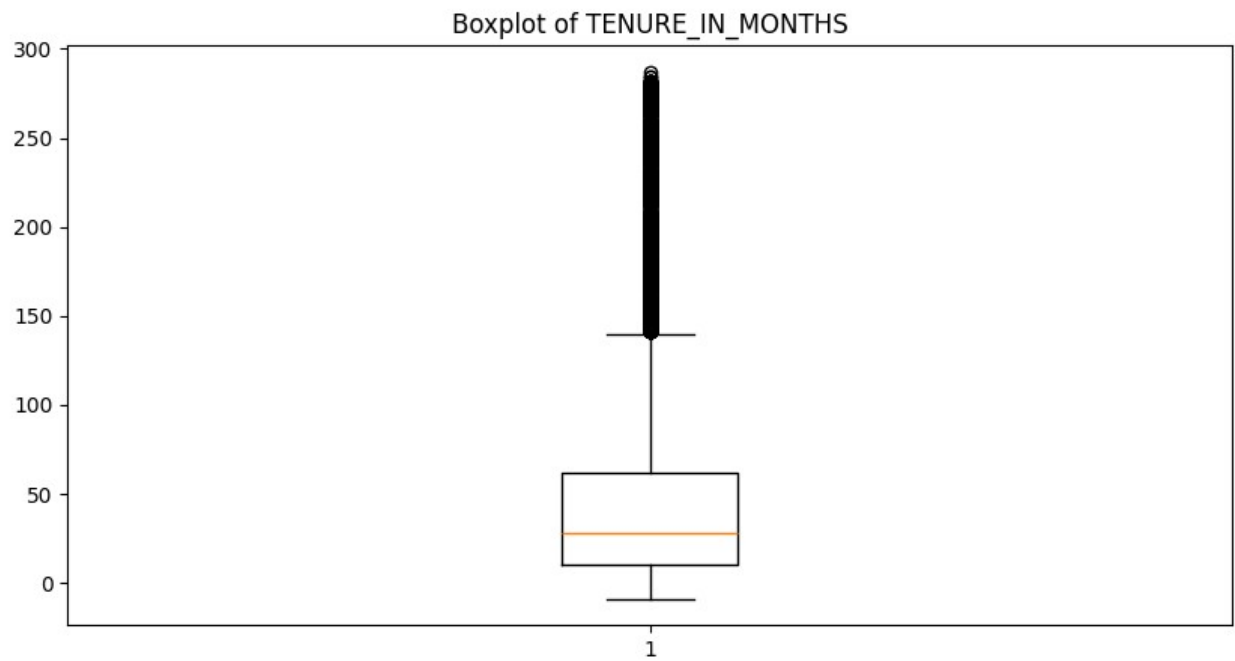
for col in numeric_columns:
    plt.figure(figsize=(10, 5))
    plt.boxplot(data_cleaned[col].dropna())
    plt.title(f"Boxplot of {col}")
    plt.show()
```

Boxplot of CUSTOMERID

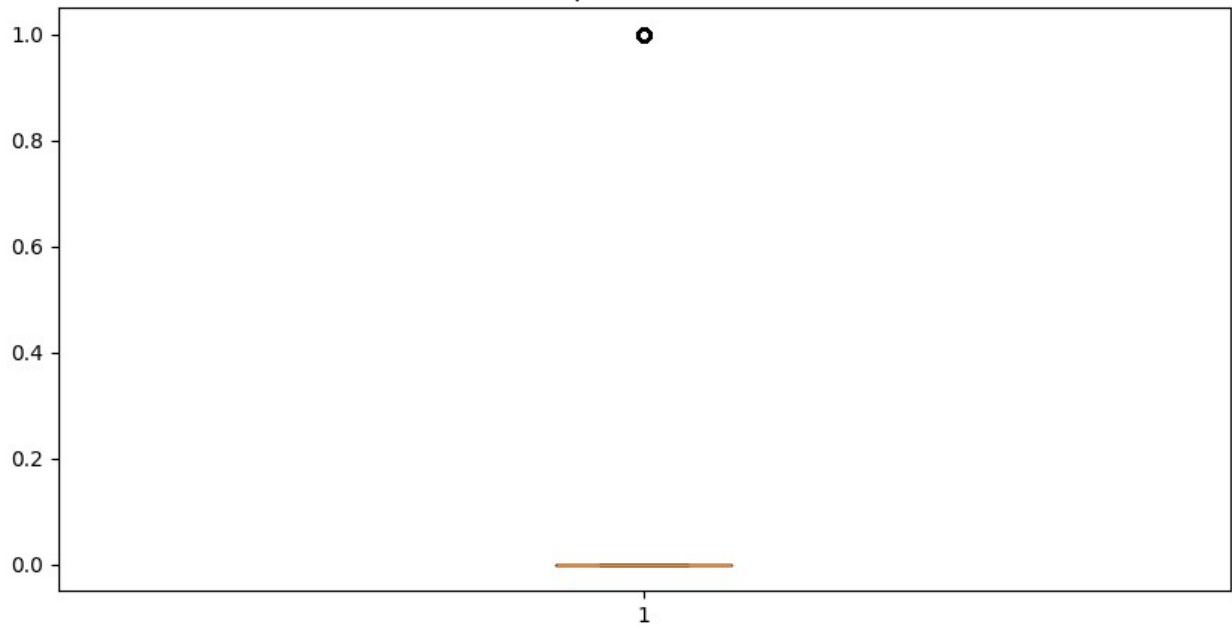


Boxplot of LCPCOUNT

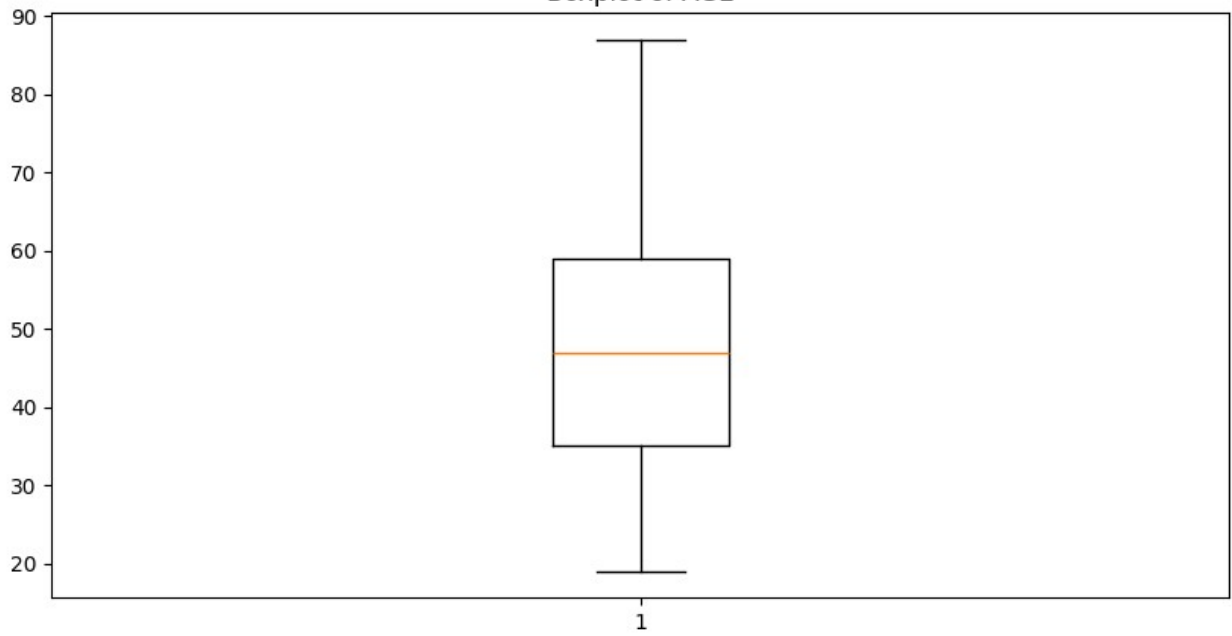


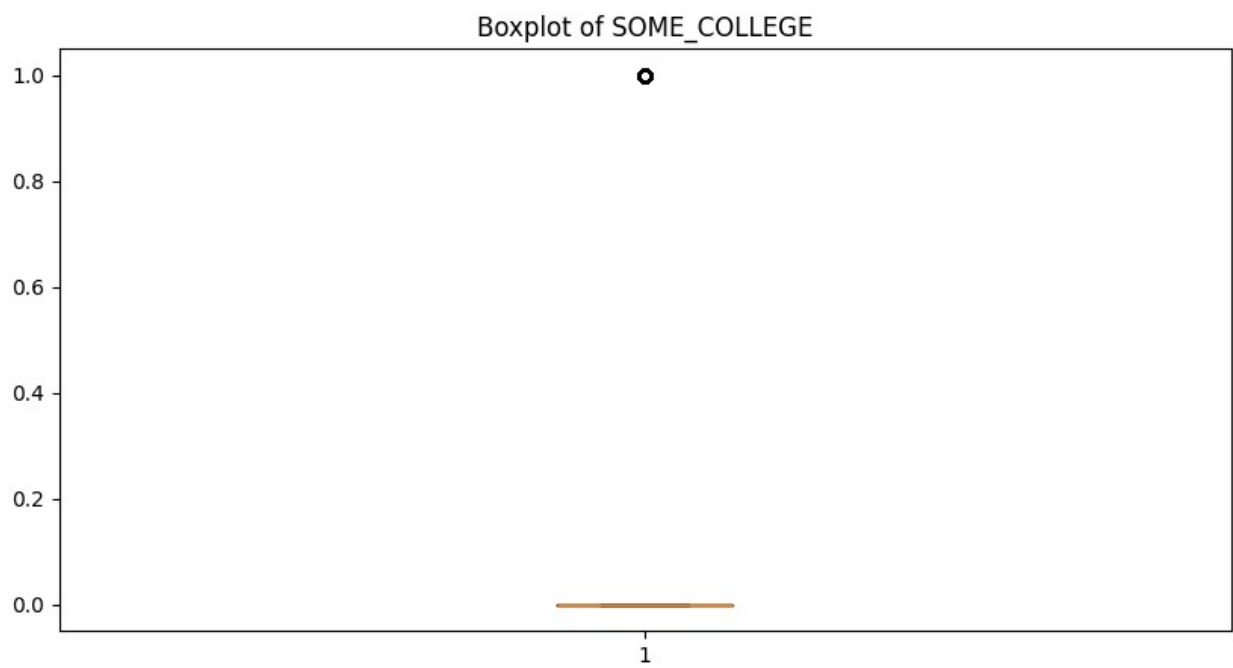
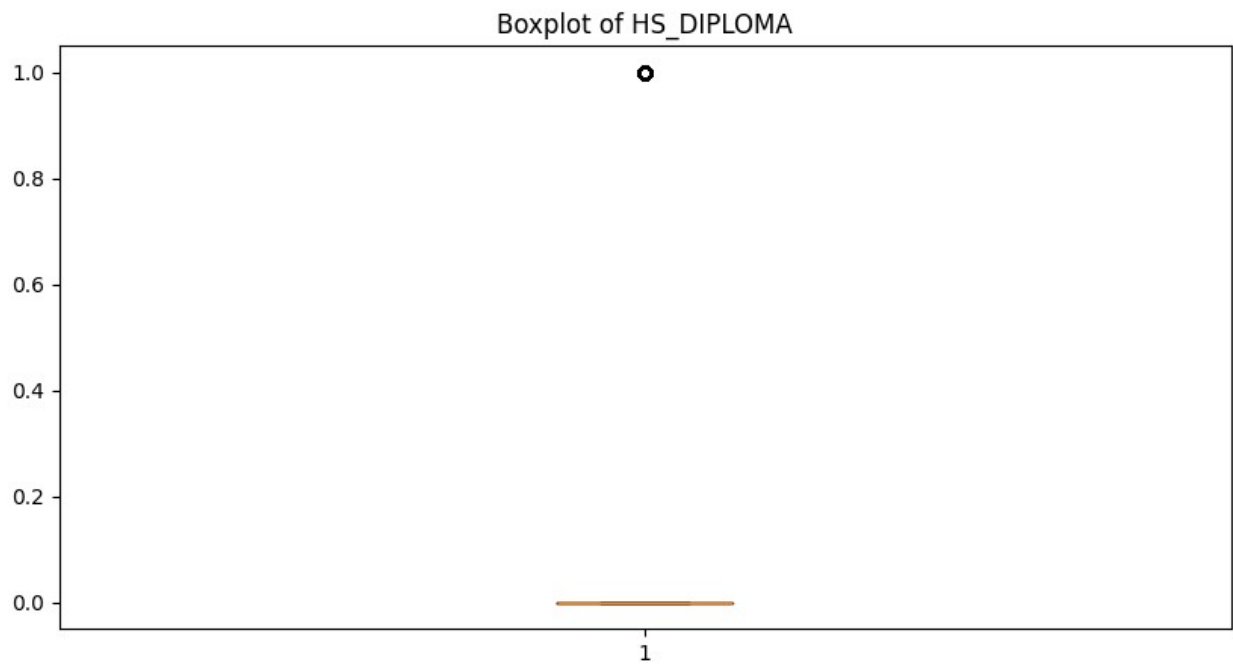


Boxplot of FEMALE

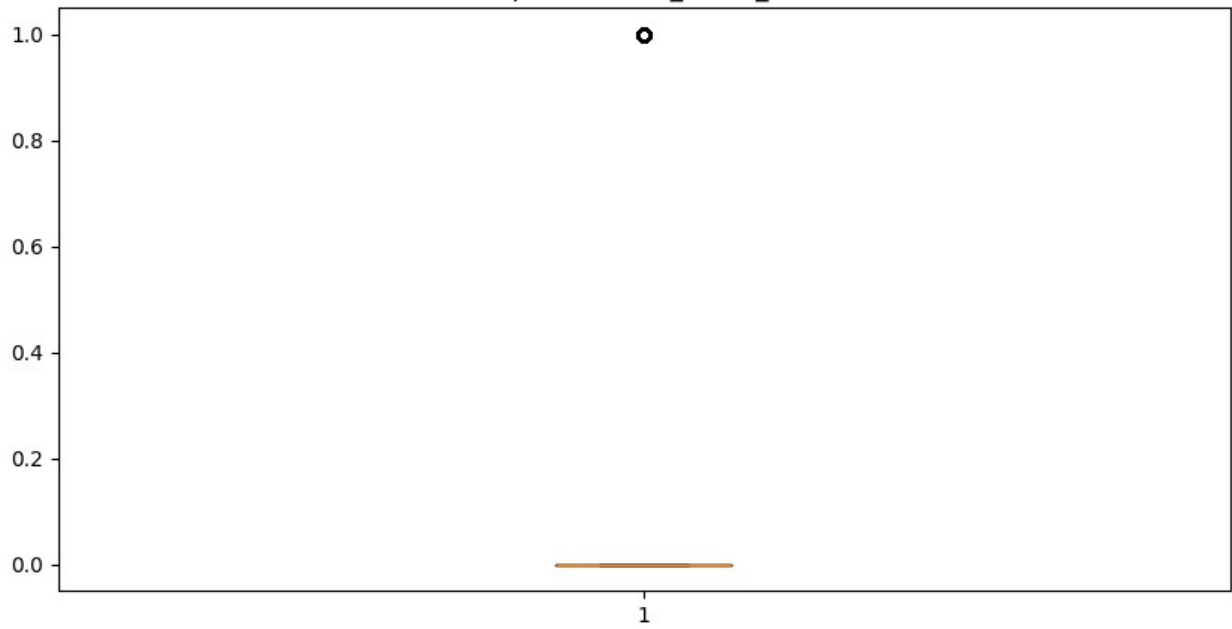


Boxplot of AGE

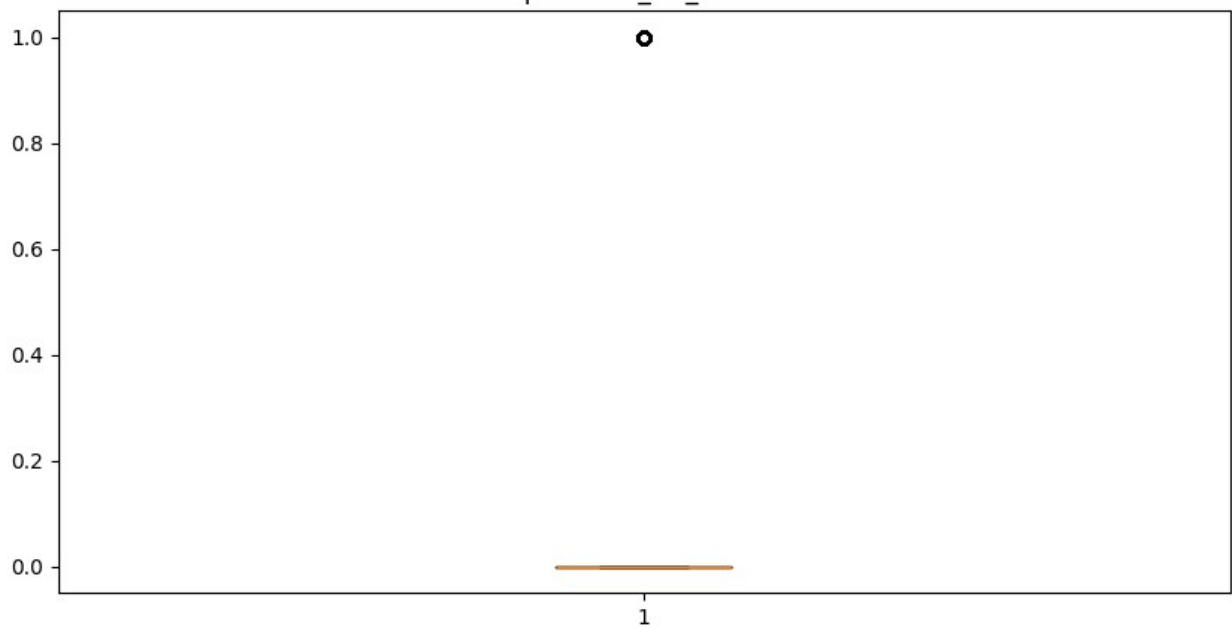


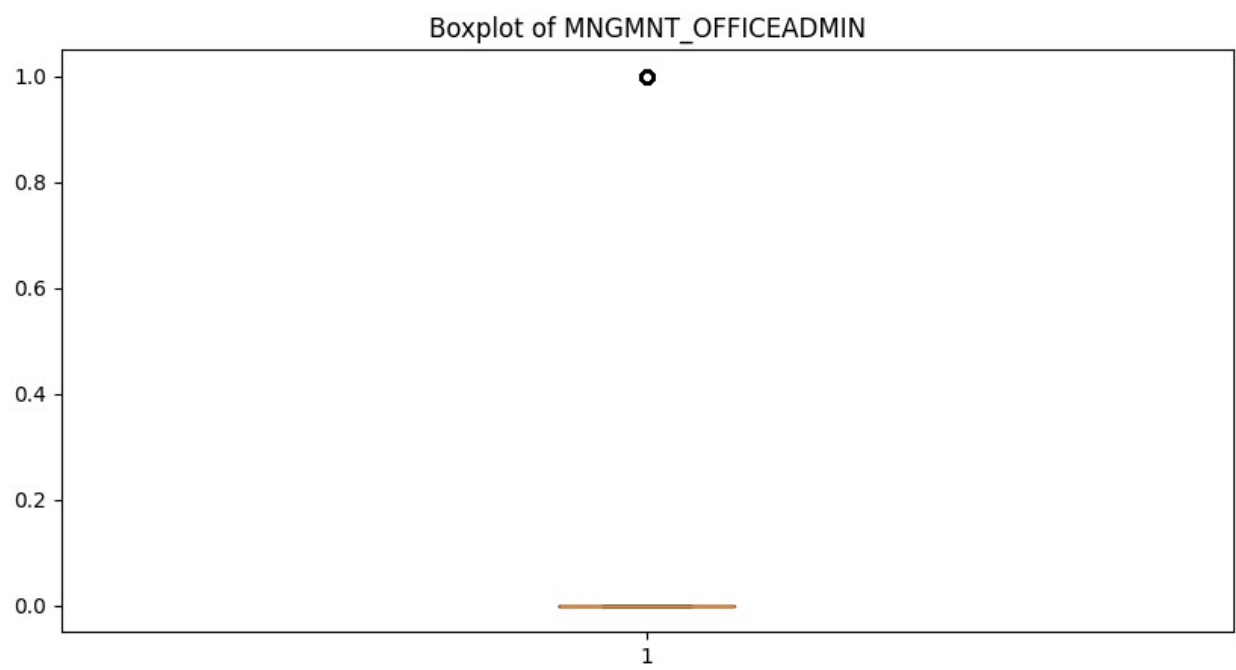
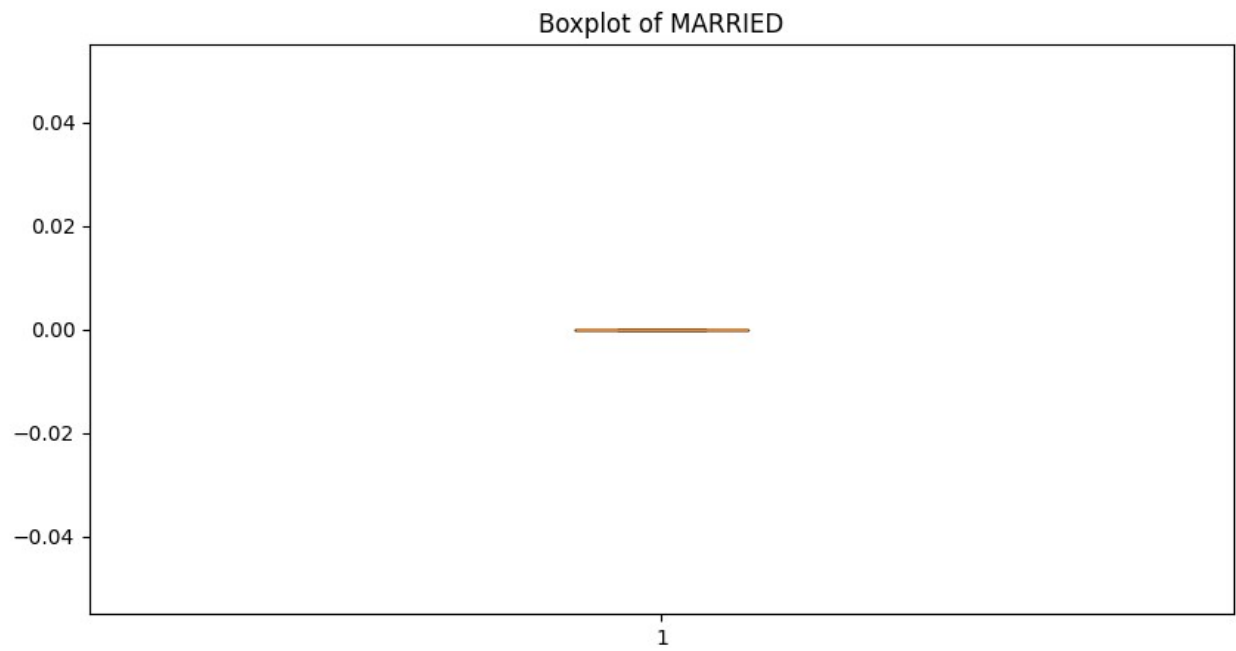


Boxplot of BACH_GRAD_DEG

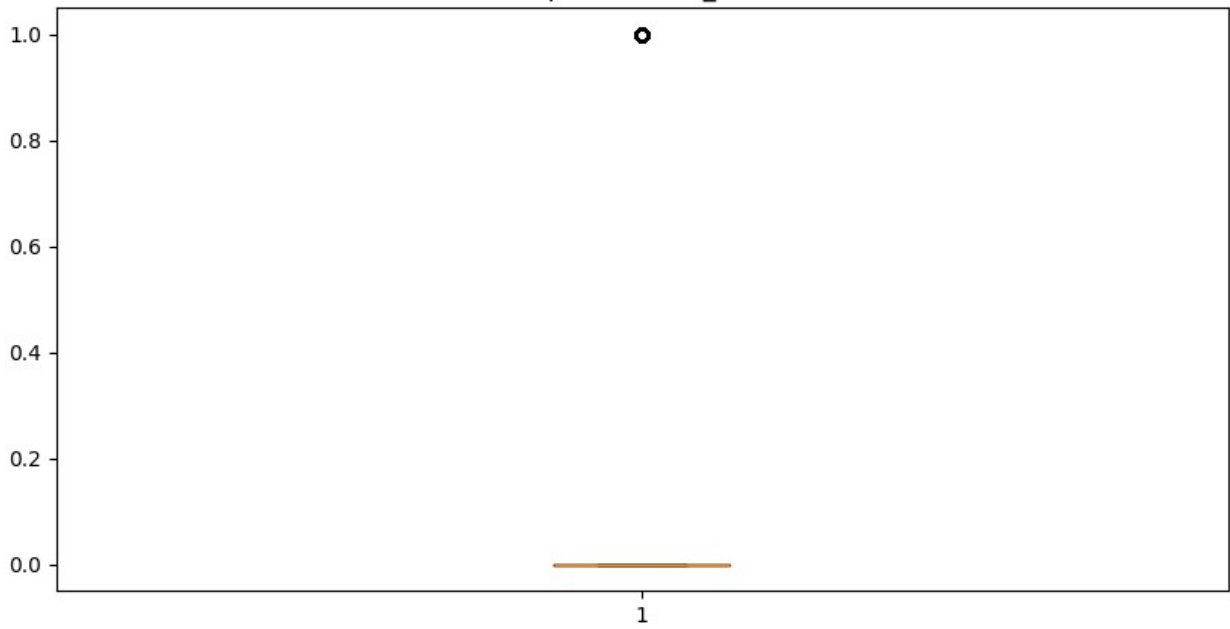


Boxplot of LT_HS_DIPLOMA

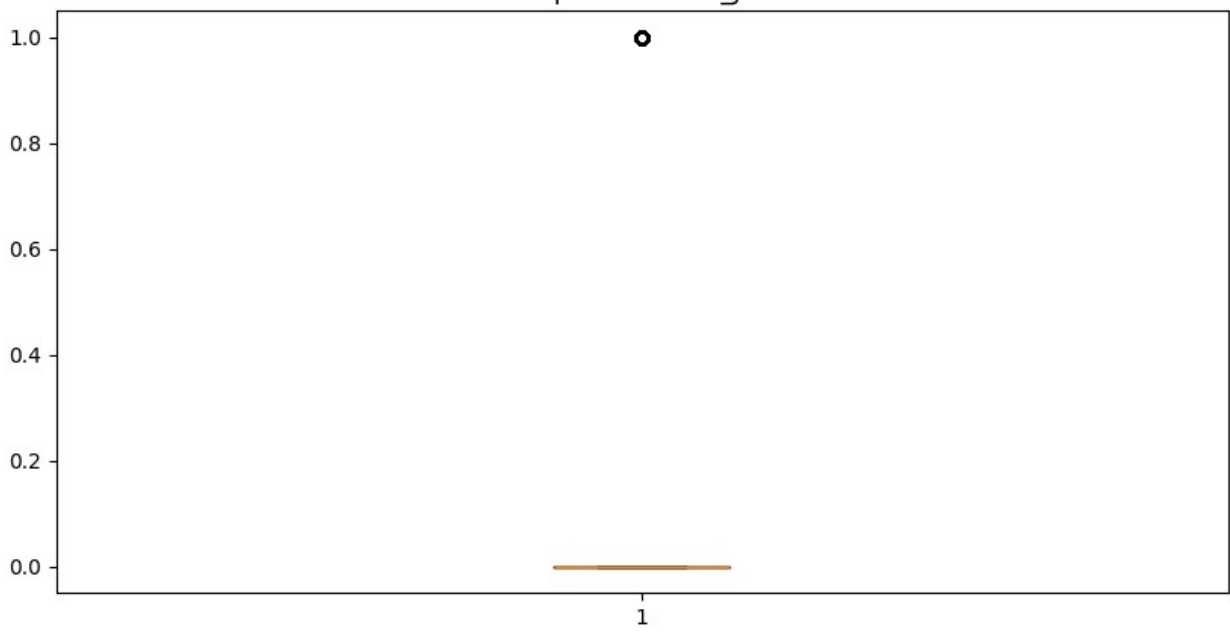




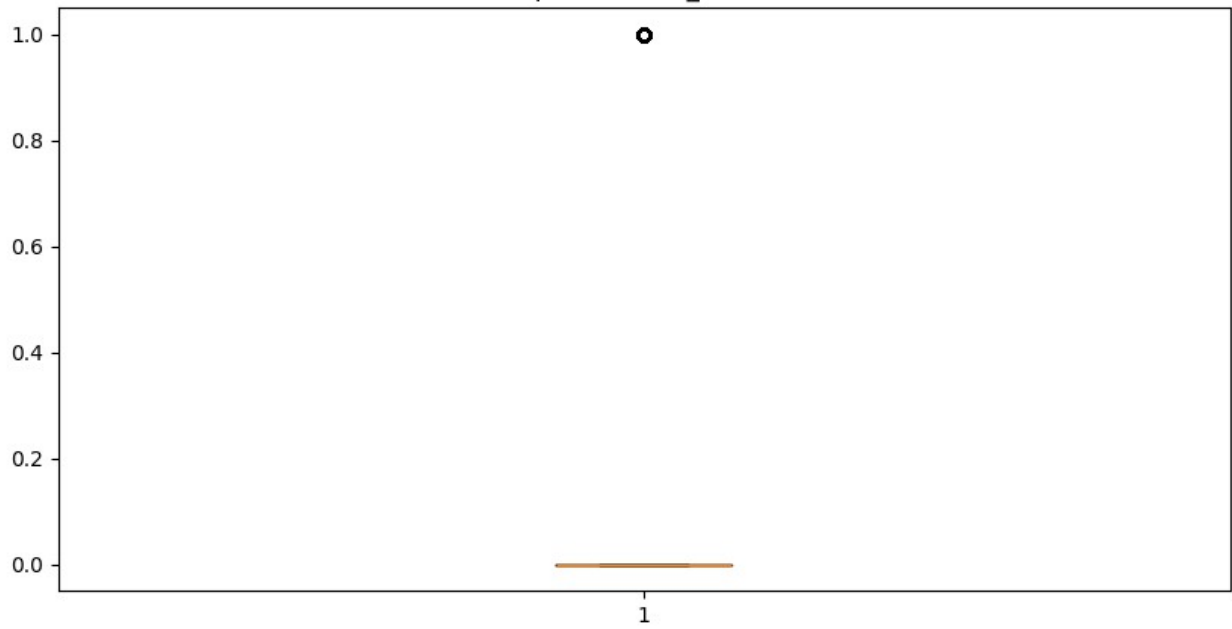
Boxplot of TECH_PROF



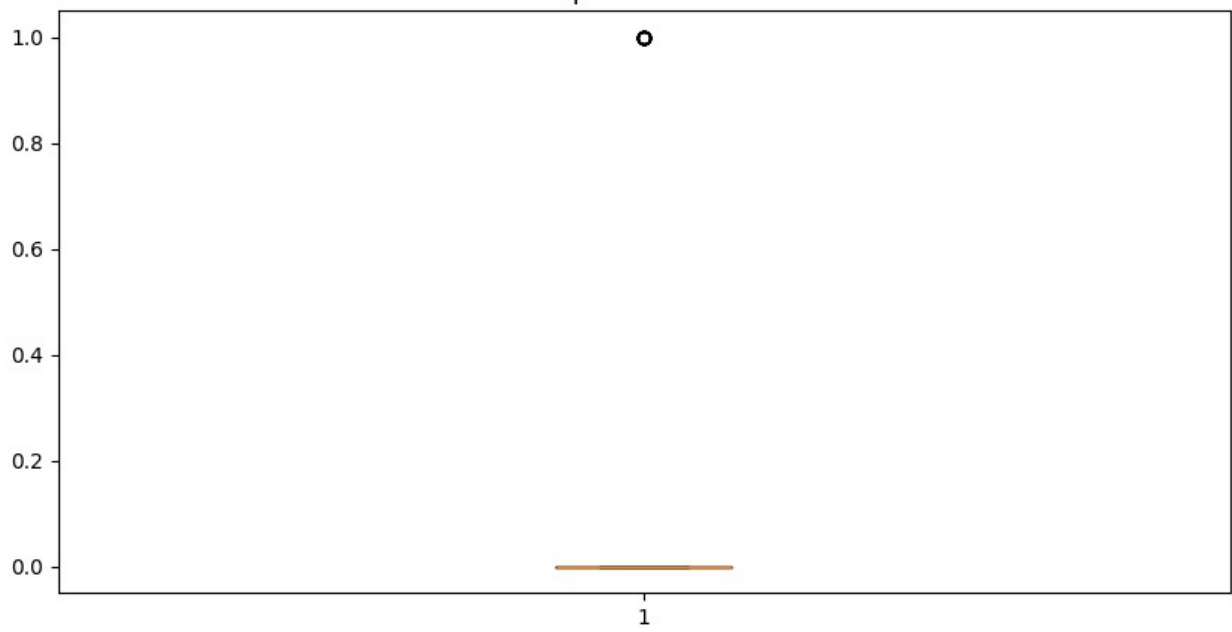
Boxplot of SALES_JOB

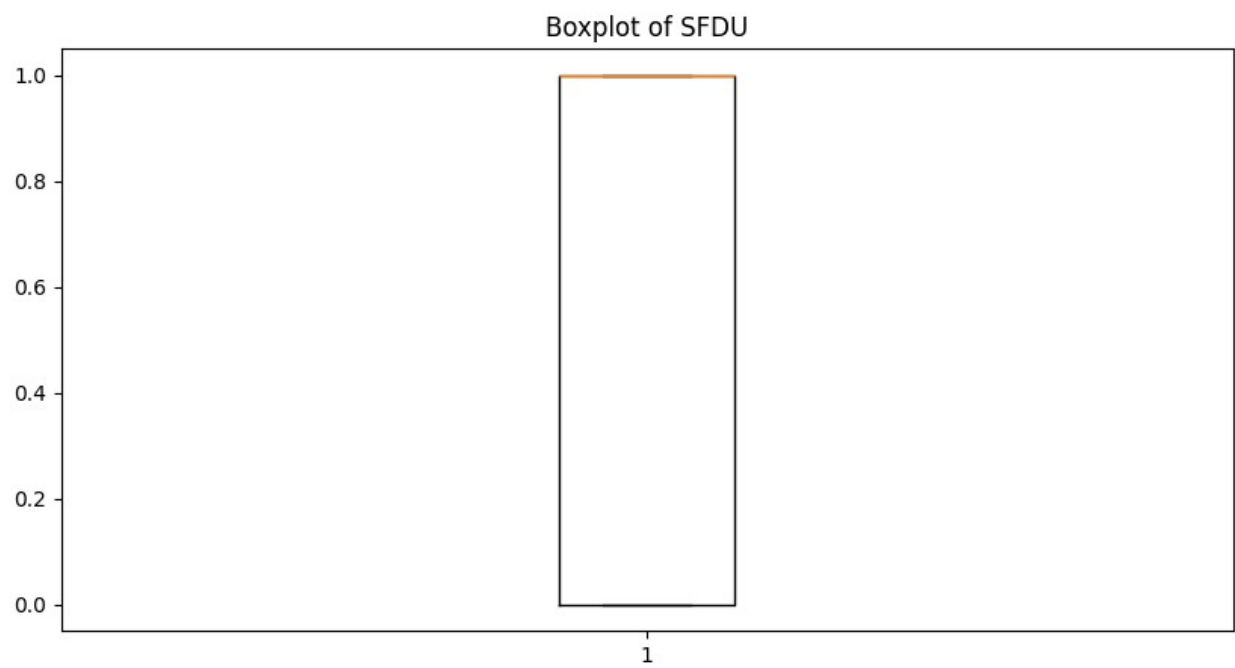
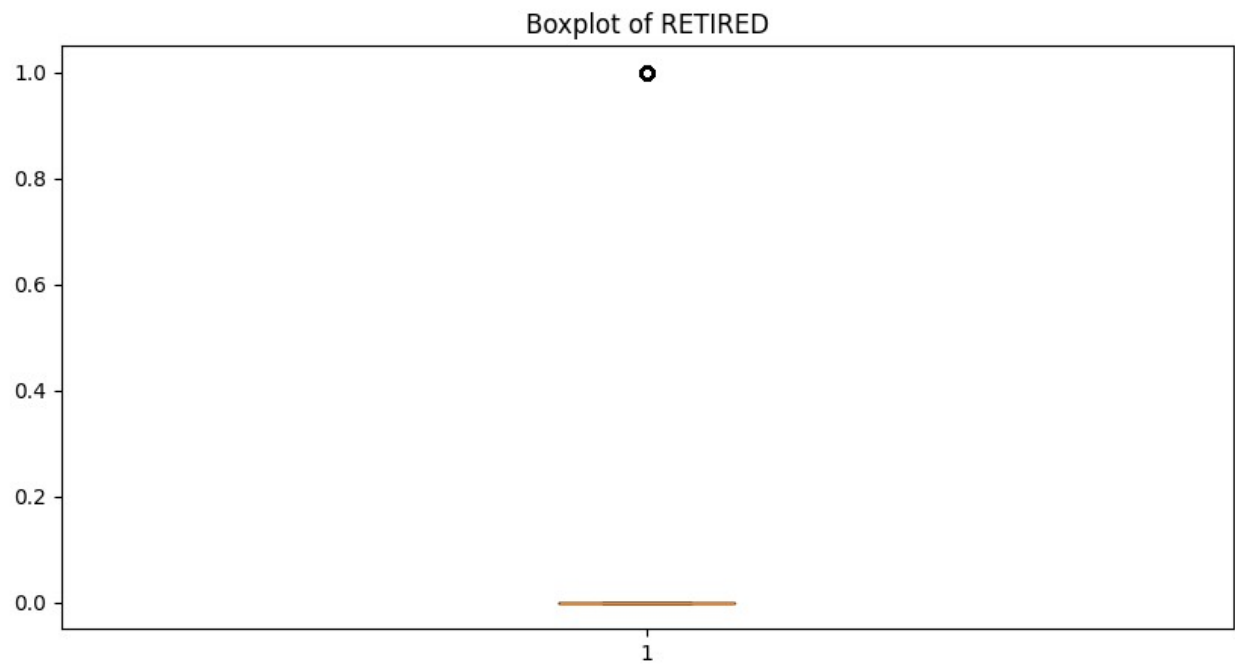


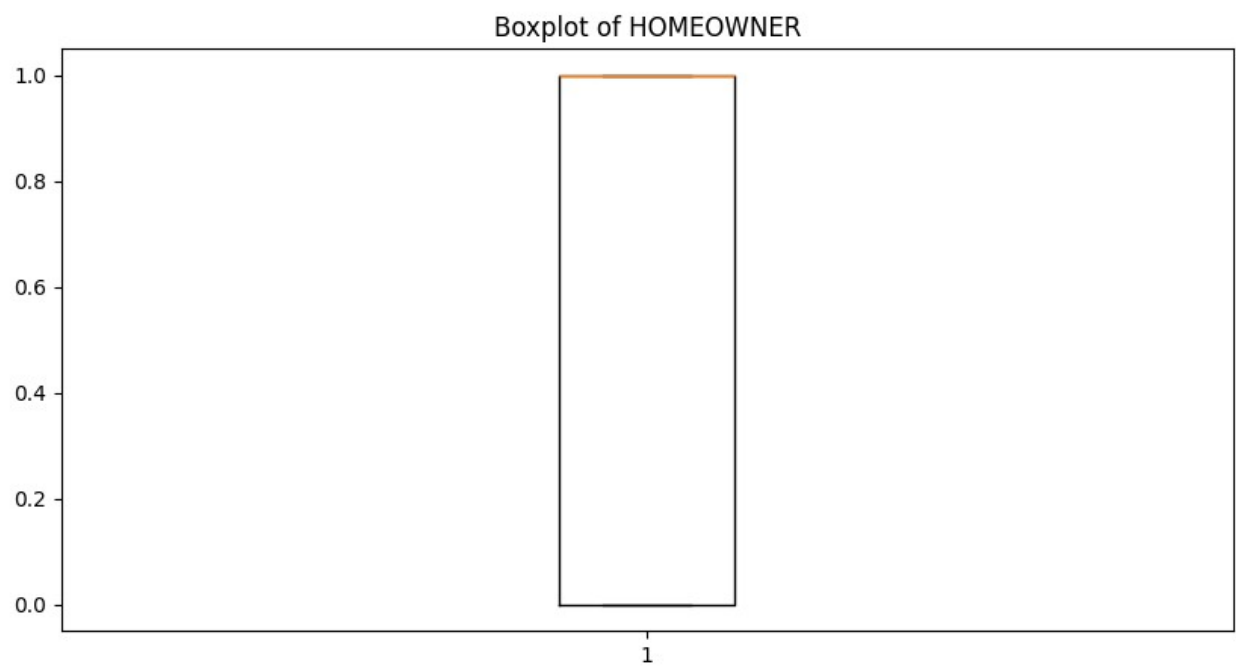
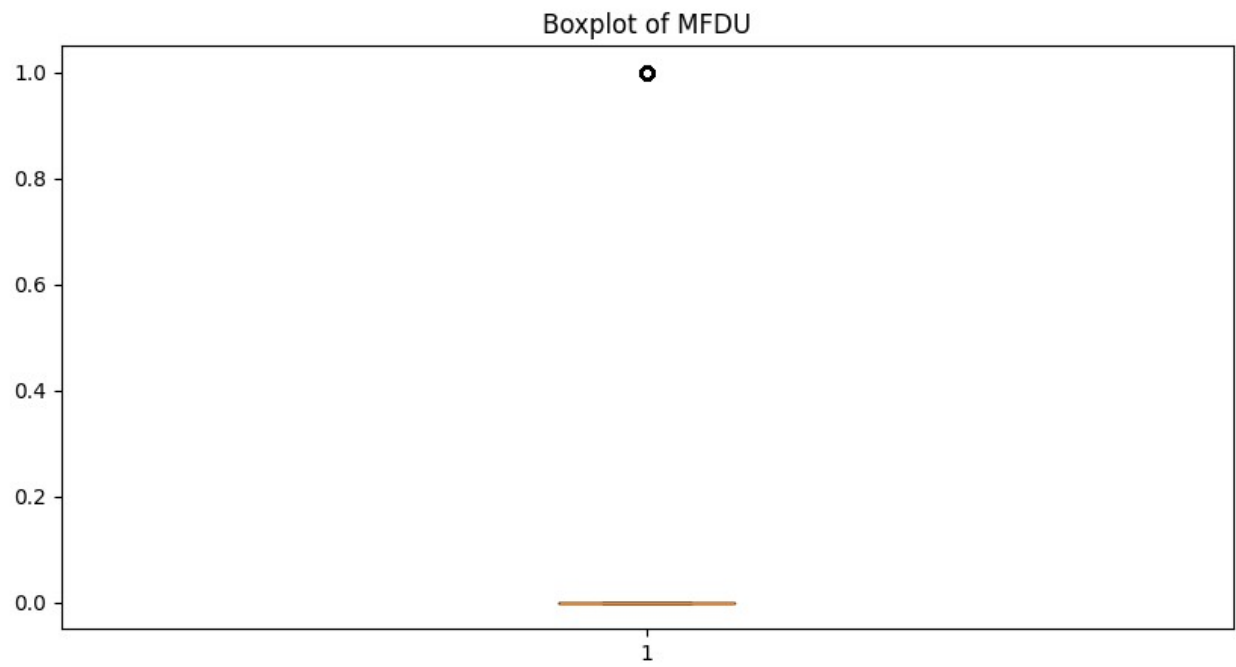
Boxplot of BLUE_COLLAR

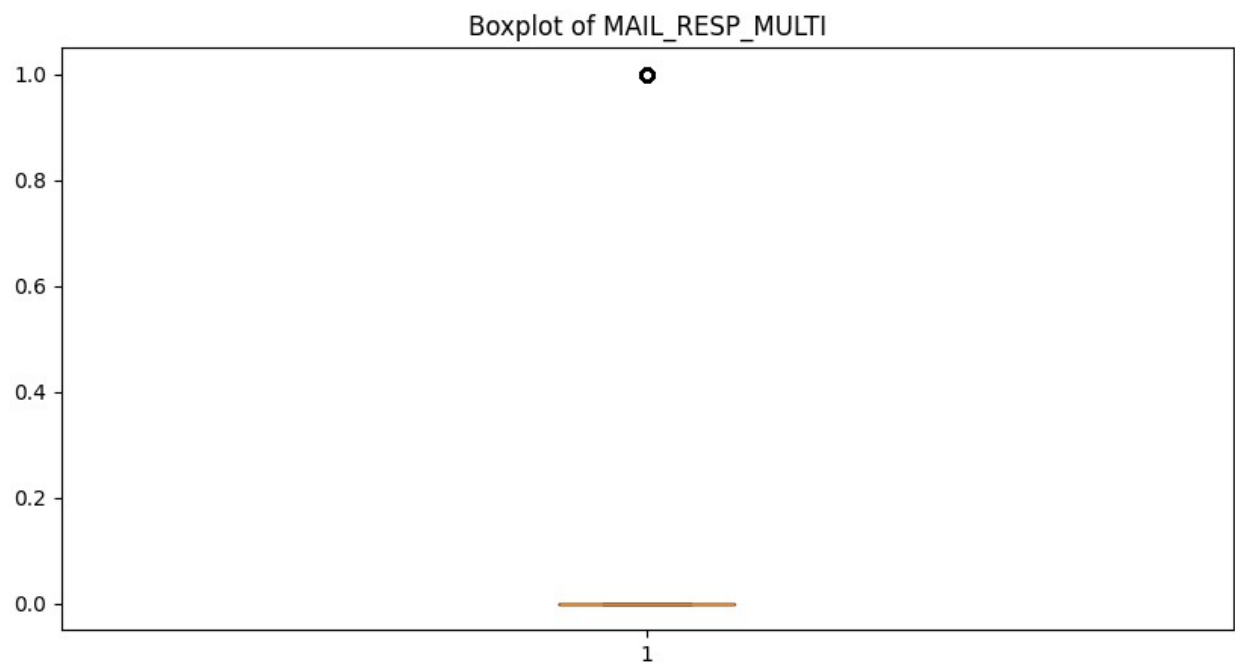
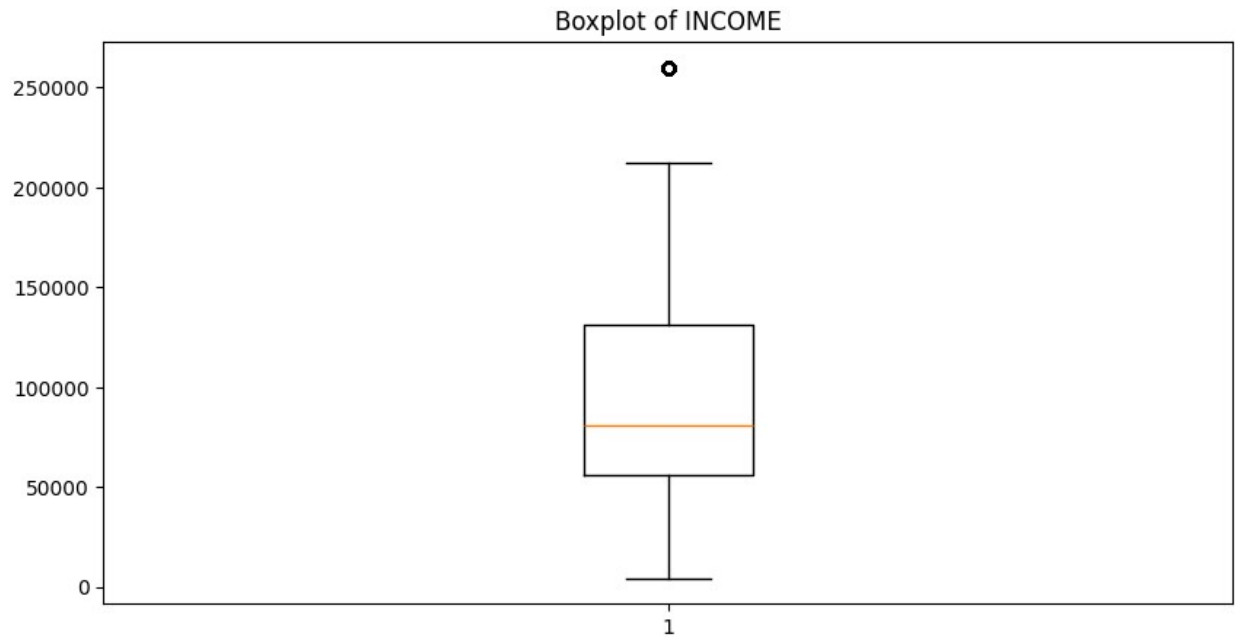


Boxplot of FARMER

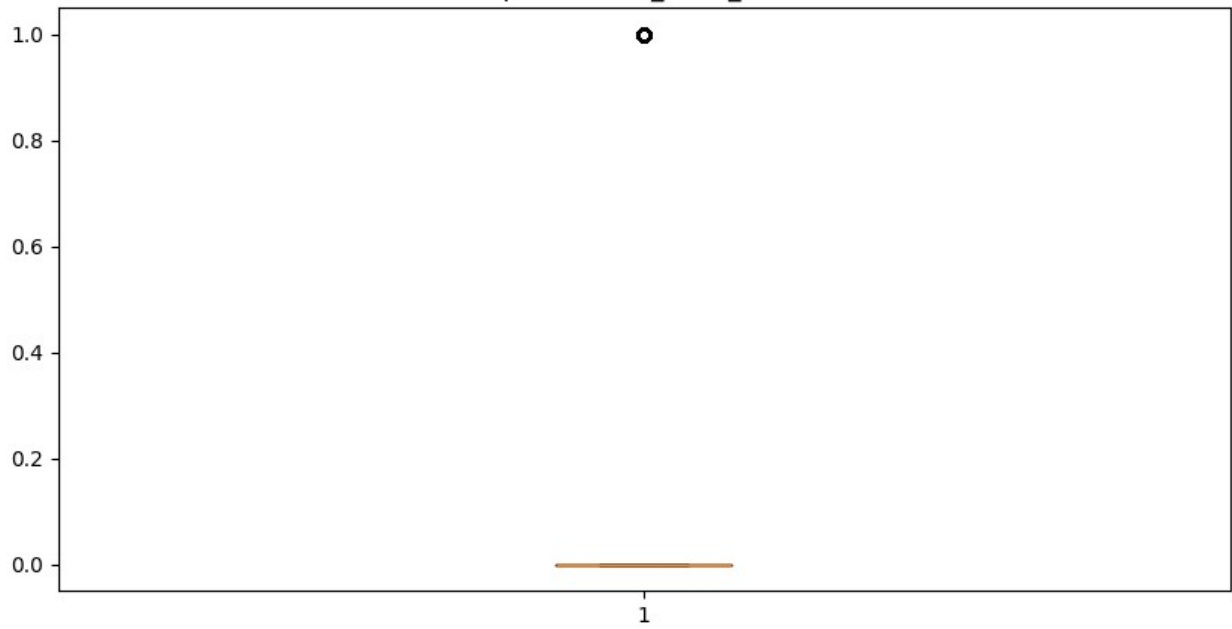




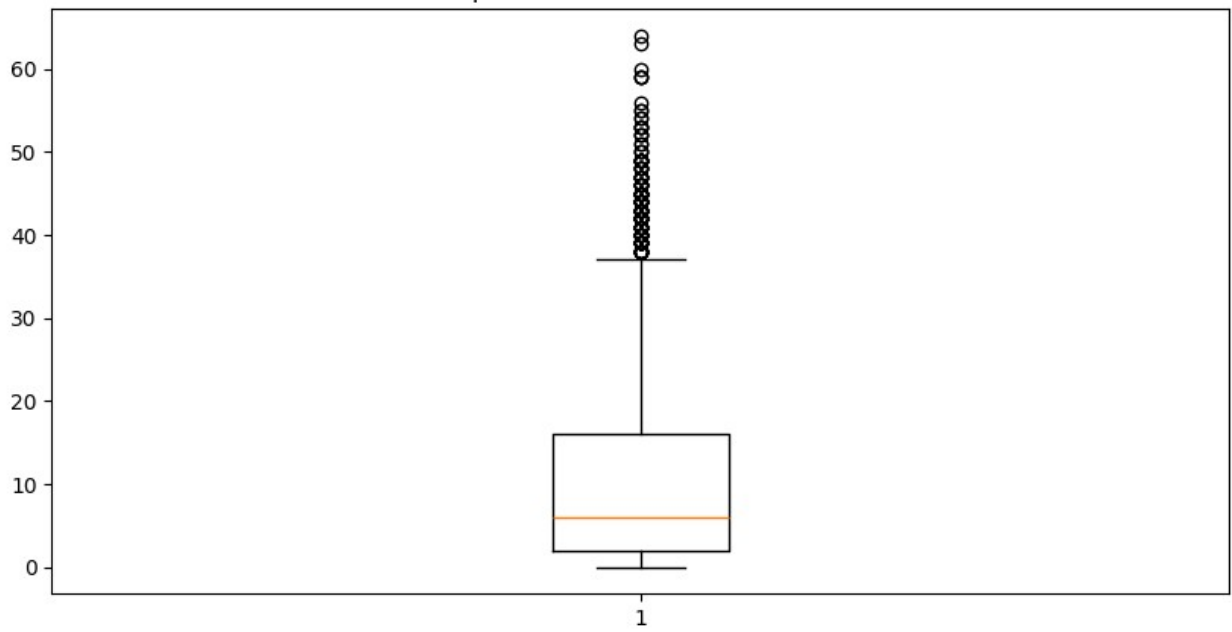




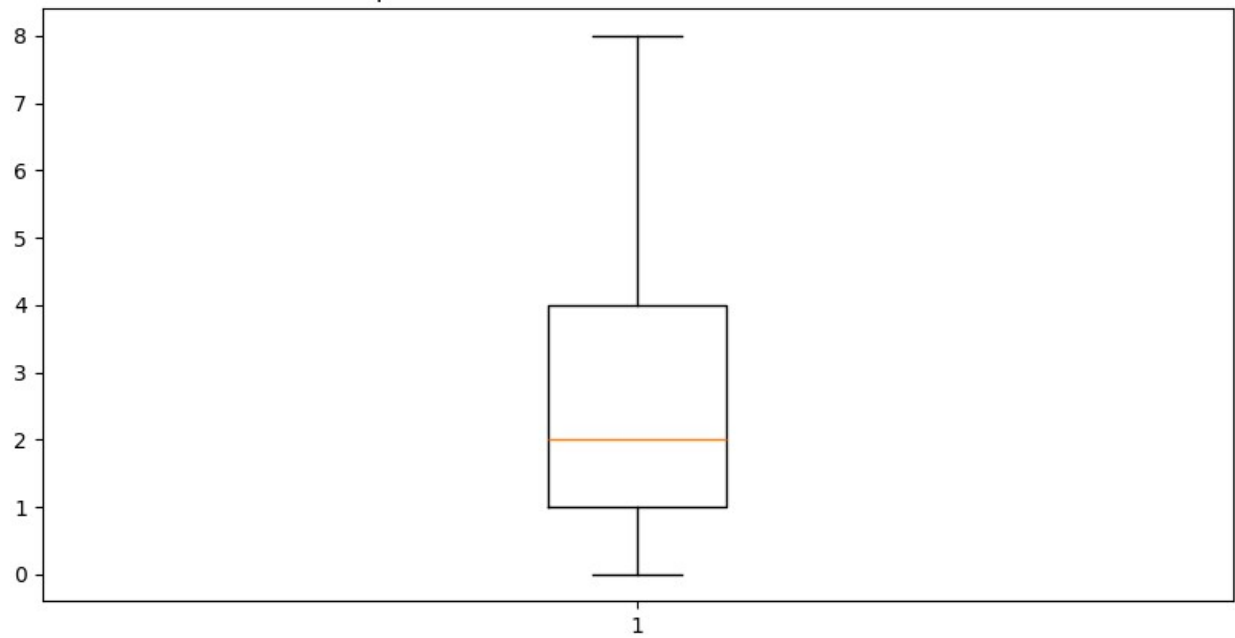
Boxplot of MAIL_RESP_SINGLE



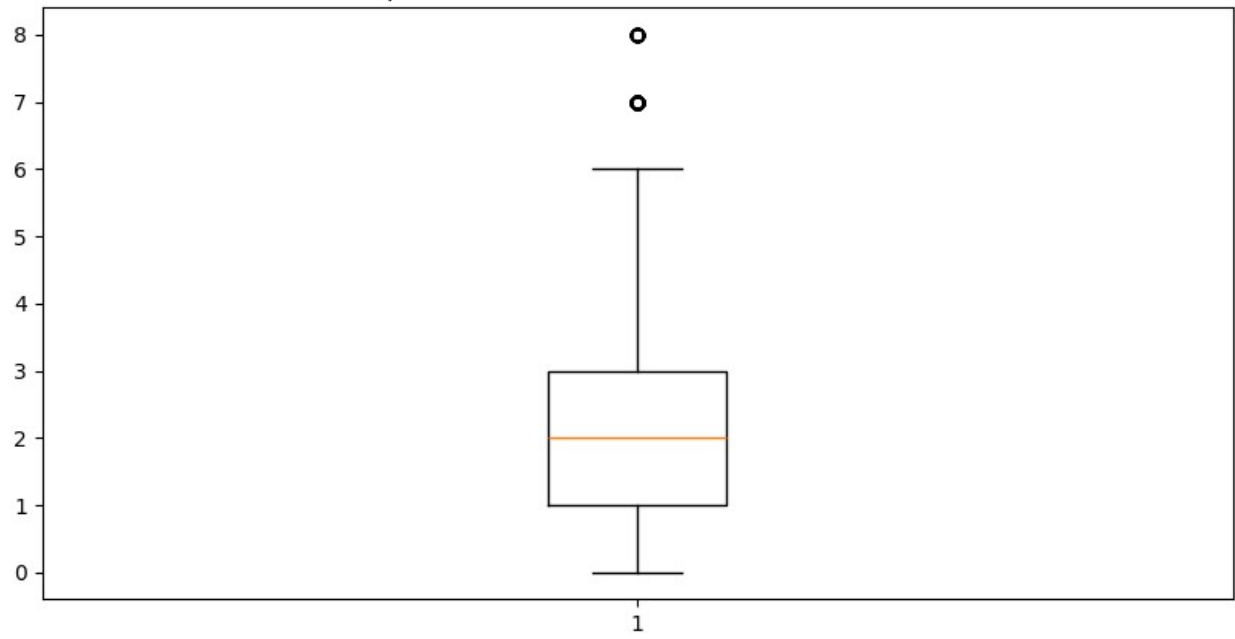
Boxplot of LENGTH OF RESIDENCE



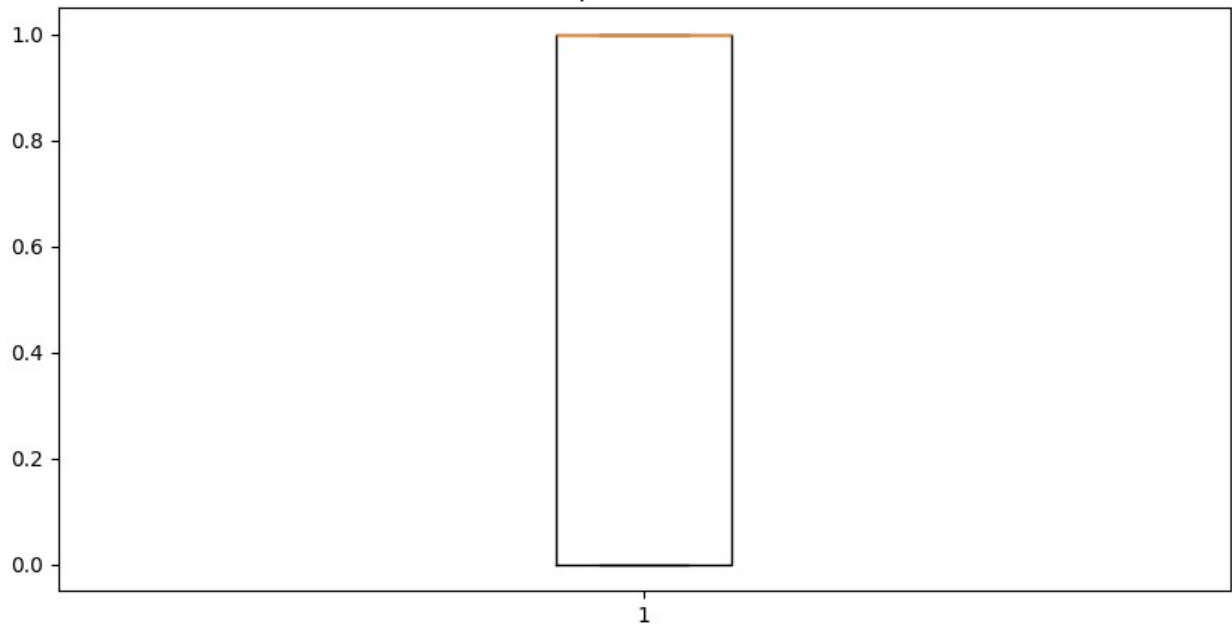
Boxplot of NUMBER OF PERSONS IN LIVING UNIT



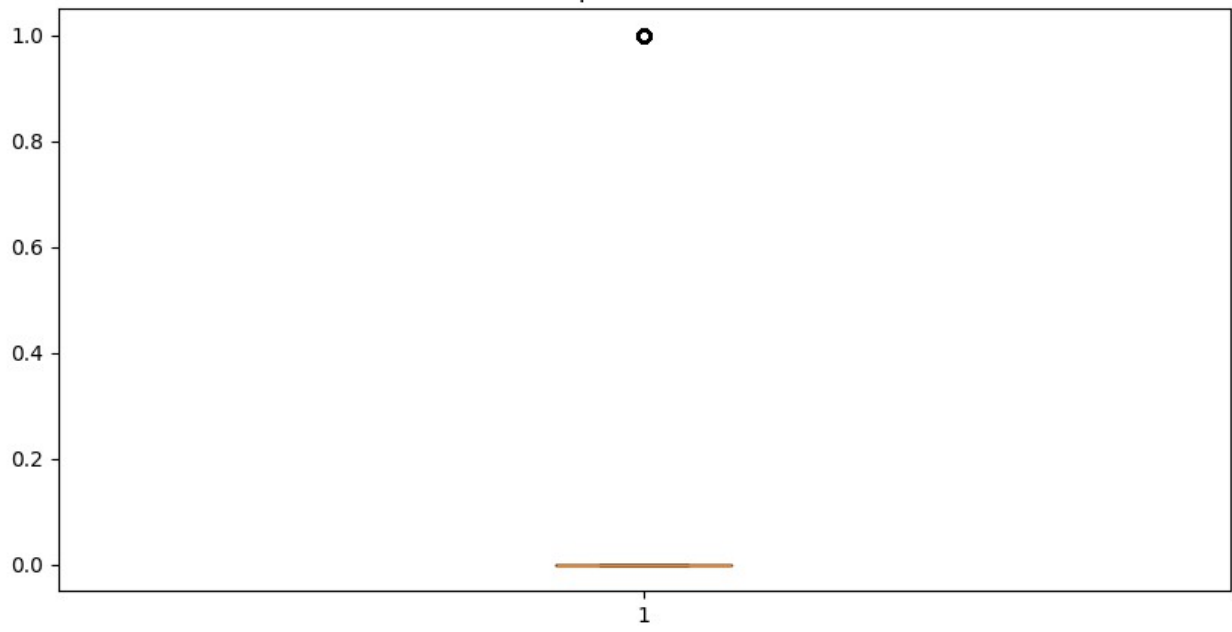
Boxplot of NUMBER OF ADULTS IN LIVING UNIT



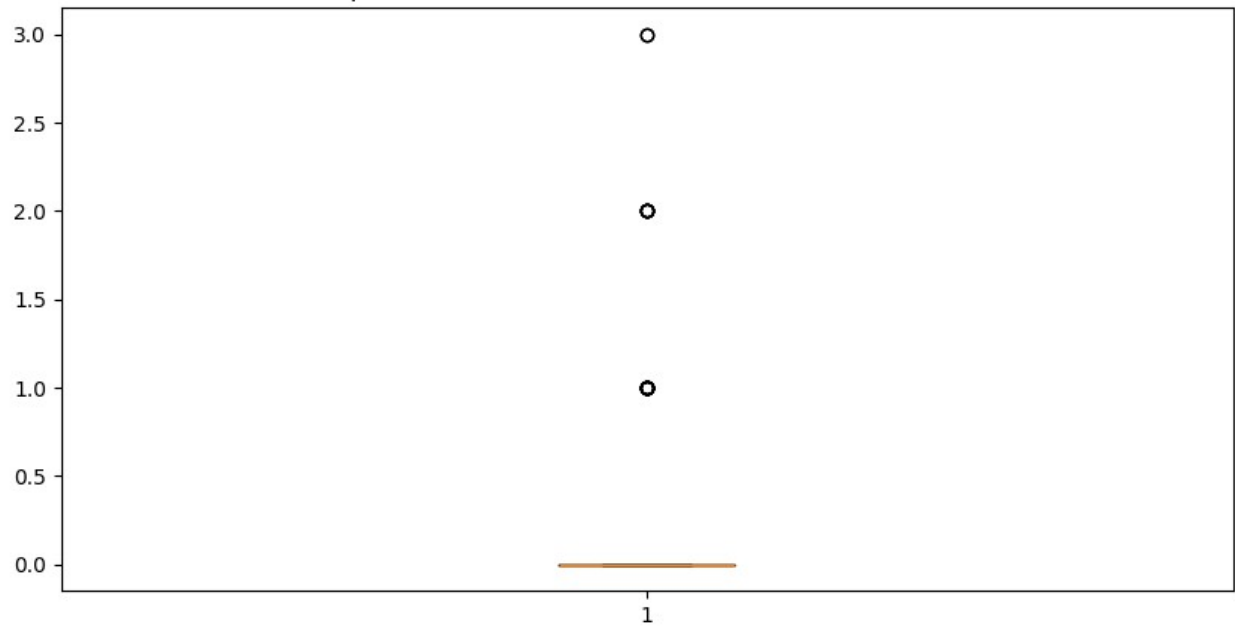
Boxplot of METRO



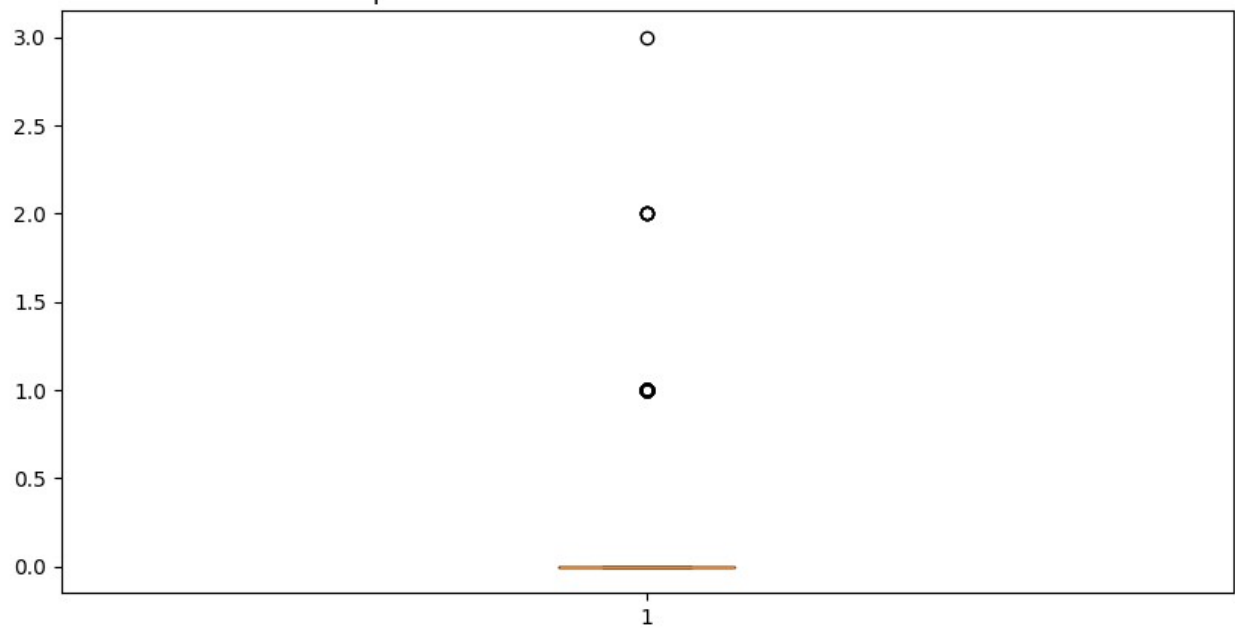
Boxplot of URBAN



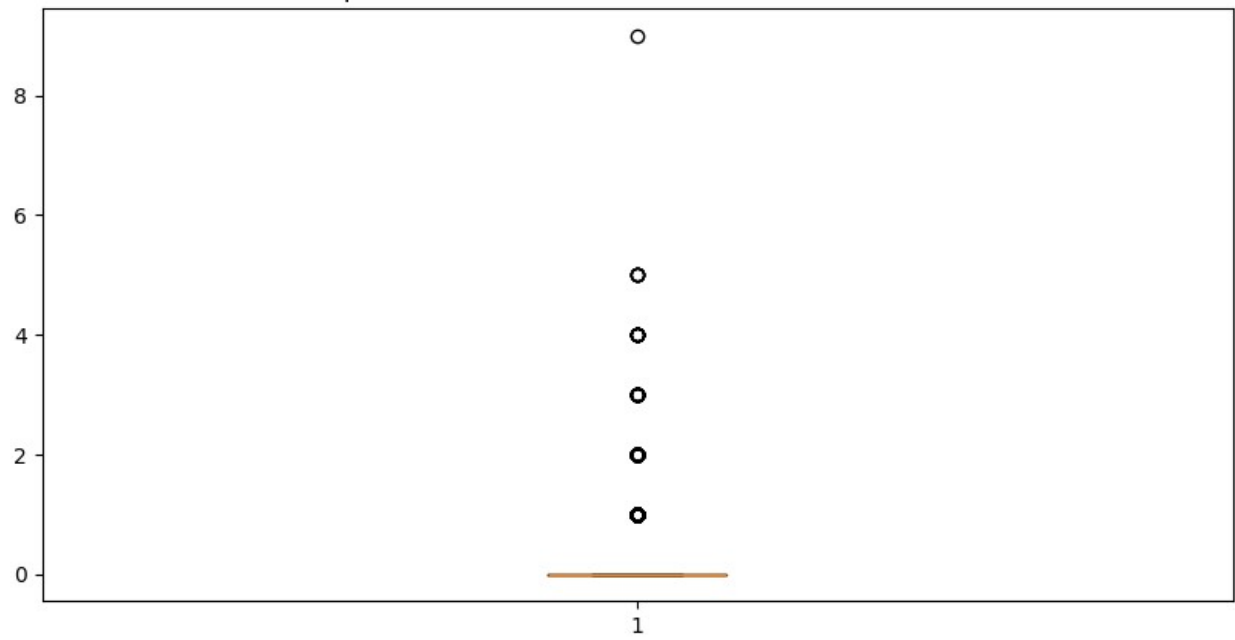
Boxplot of MOR BANK: UPSCALE MERCHANDISE BUYER



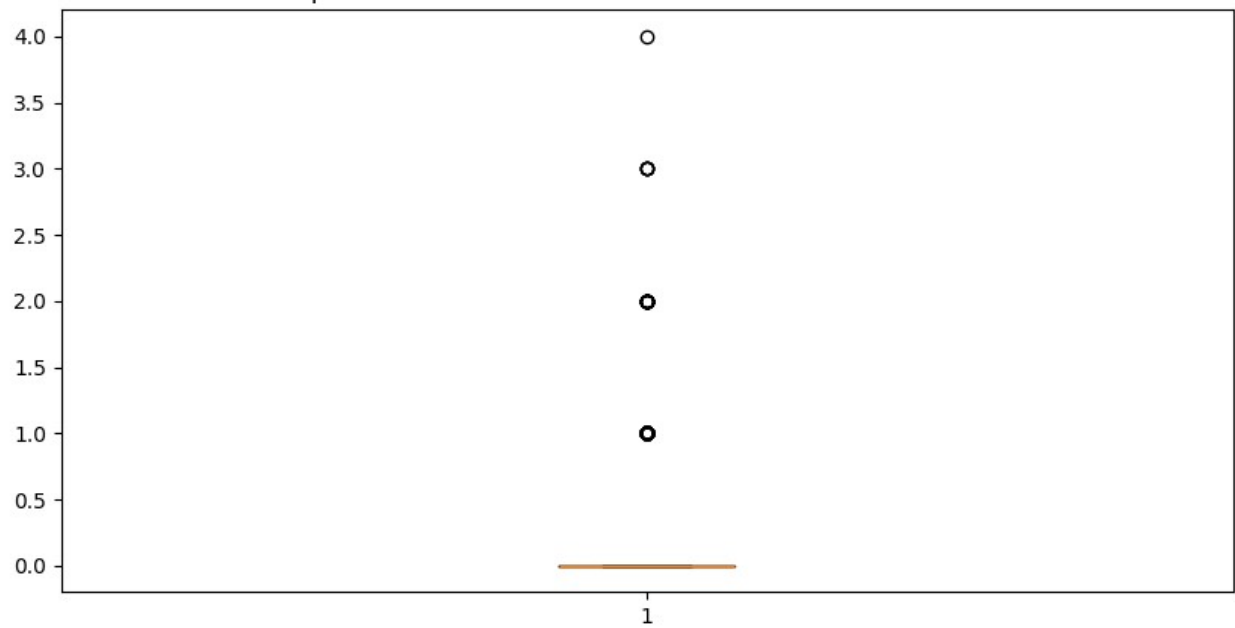
Boxplot of MOR BANK: MALE MERCHANDISE BUYER



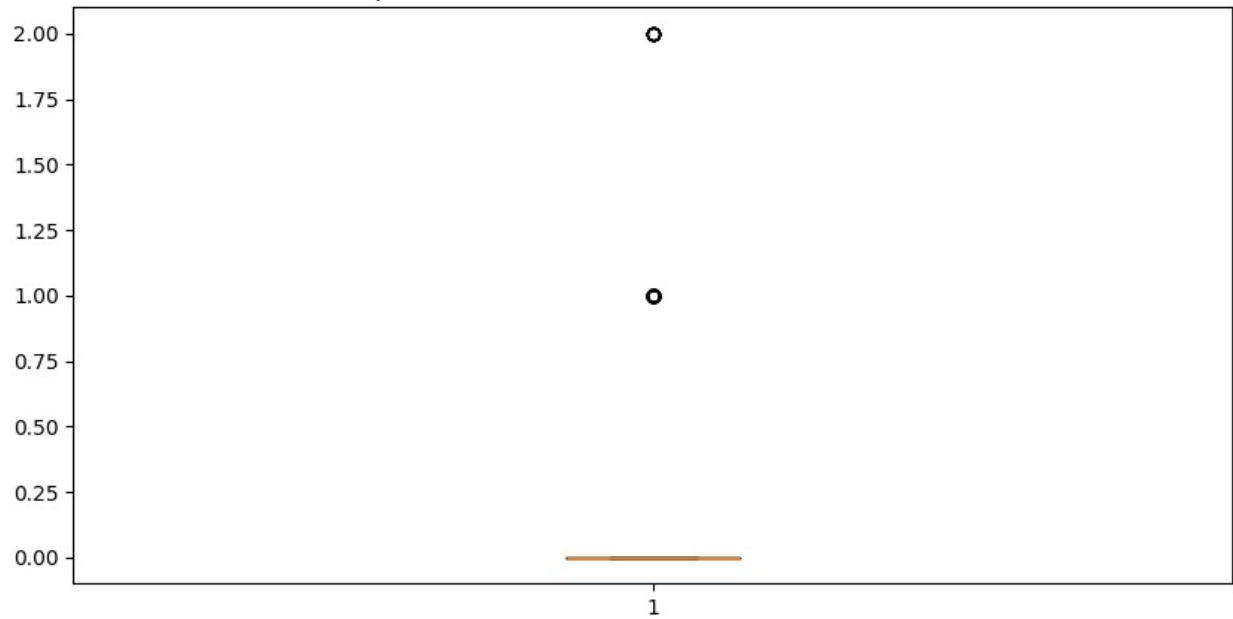
Boxplot of MOR BANK: FEMALE MERCHANDISE BUYER



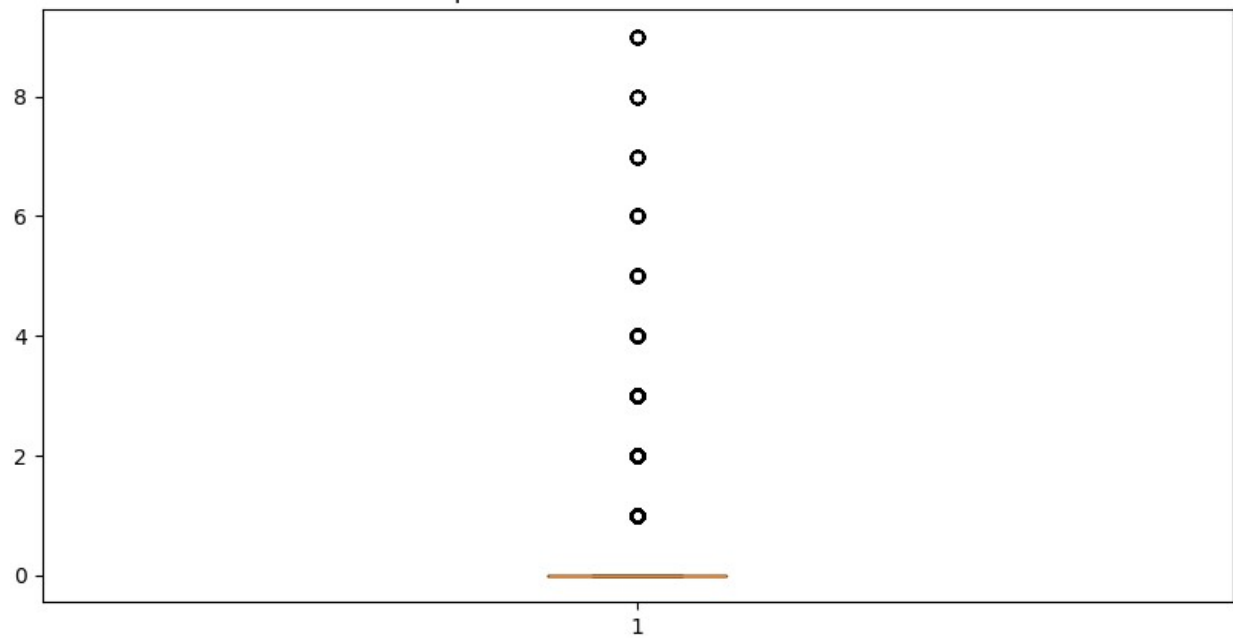
Boxplot of MOR BANK: CRAFTS-HOBBY MERCHANDISE BUYER



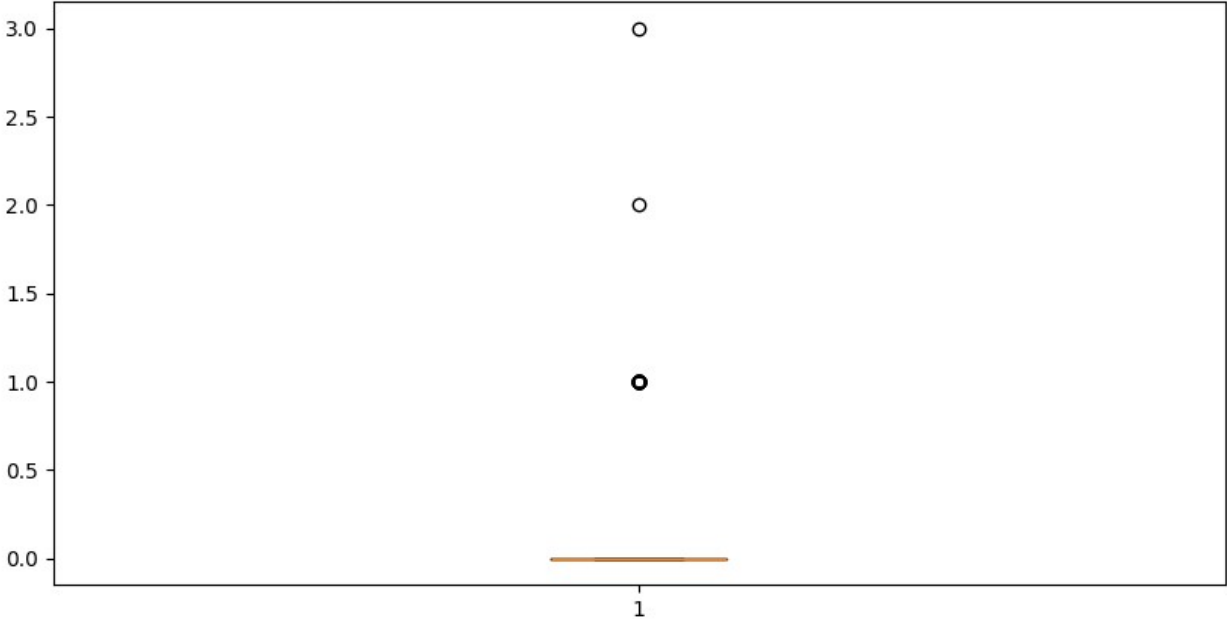
Boxplot of MOR BANK: GARDENING-FARMING BUYER



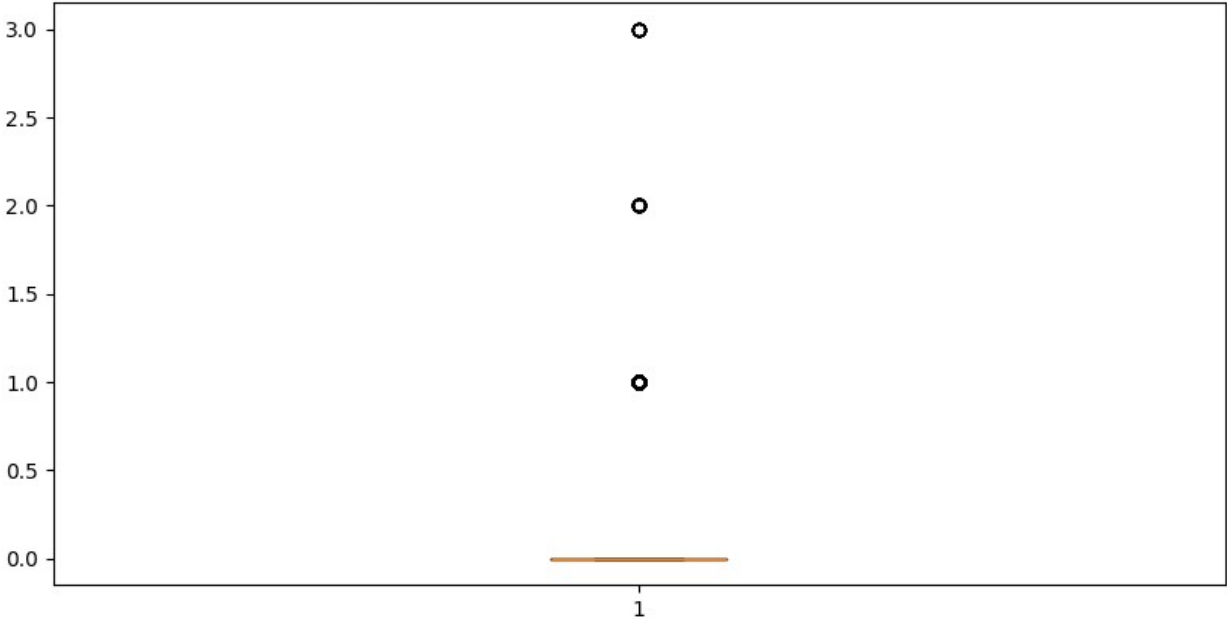
Boxplot of MOR BANK: BOOK BUYER



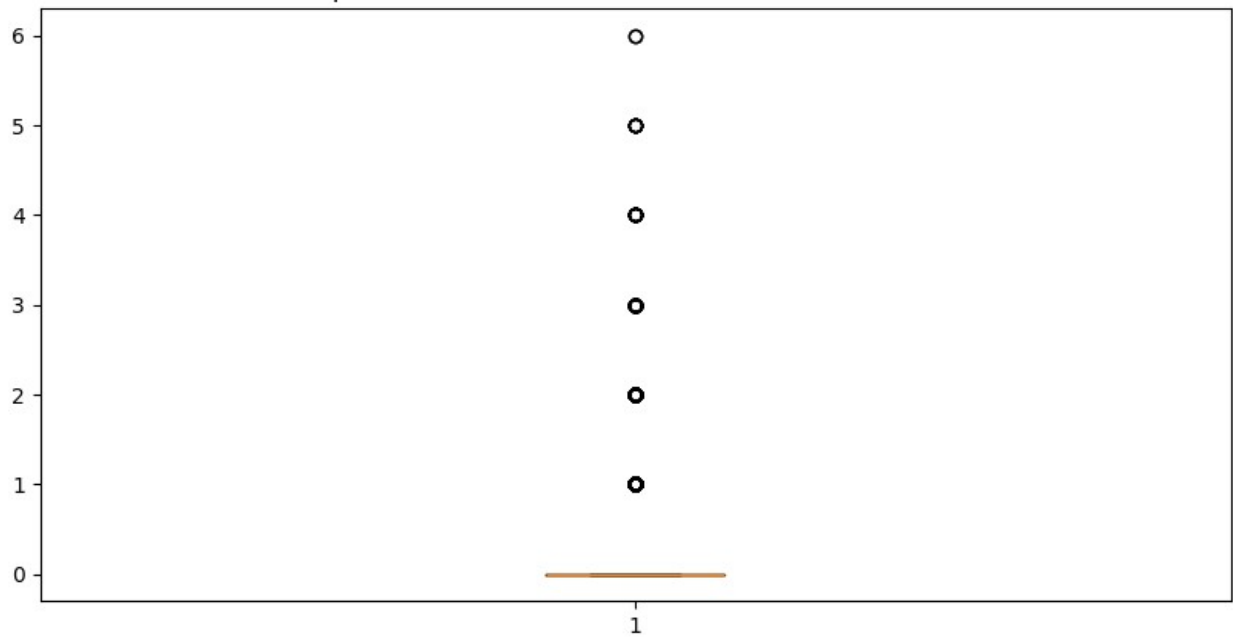
Boxplot of MOR BANK: COLLECT-SPECIAL FOODS BUYER



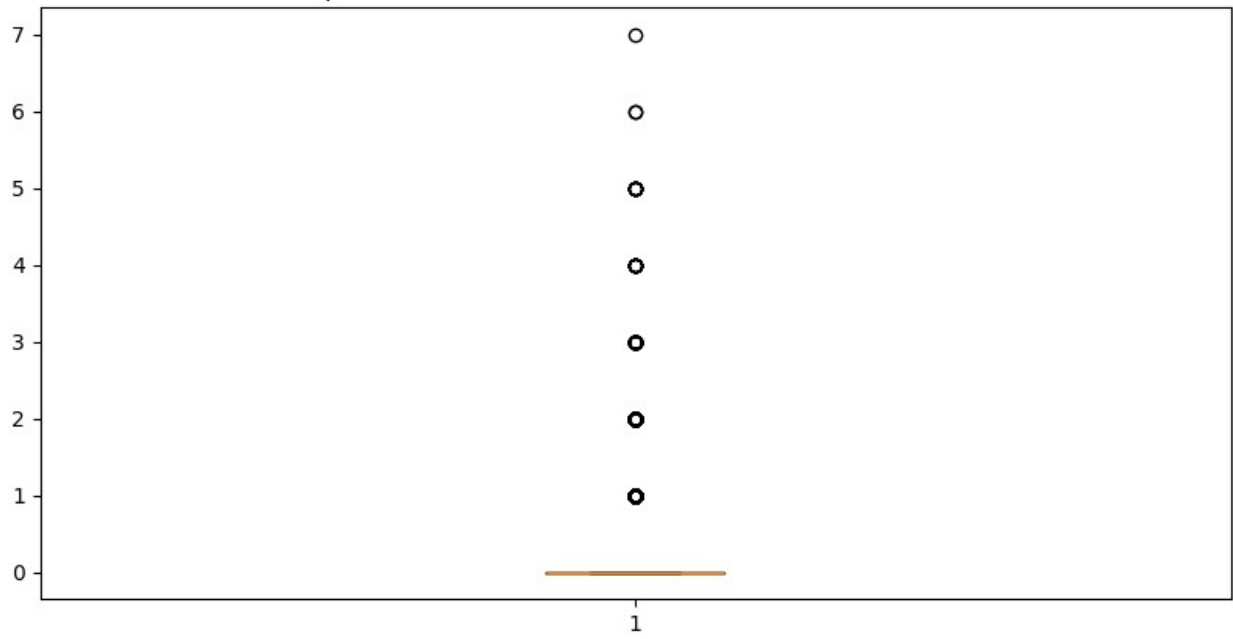
Boxplot of MOR BANK: GIFTS AND GADGETS BUYER



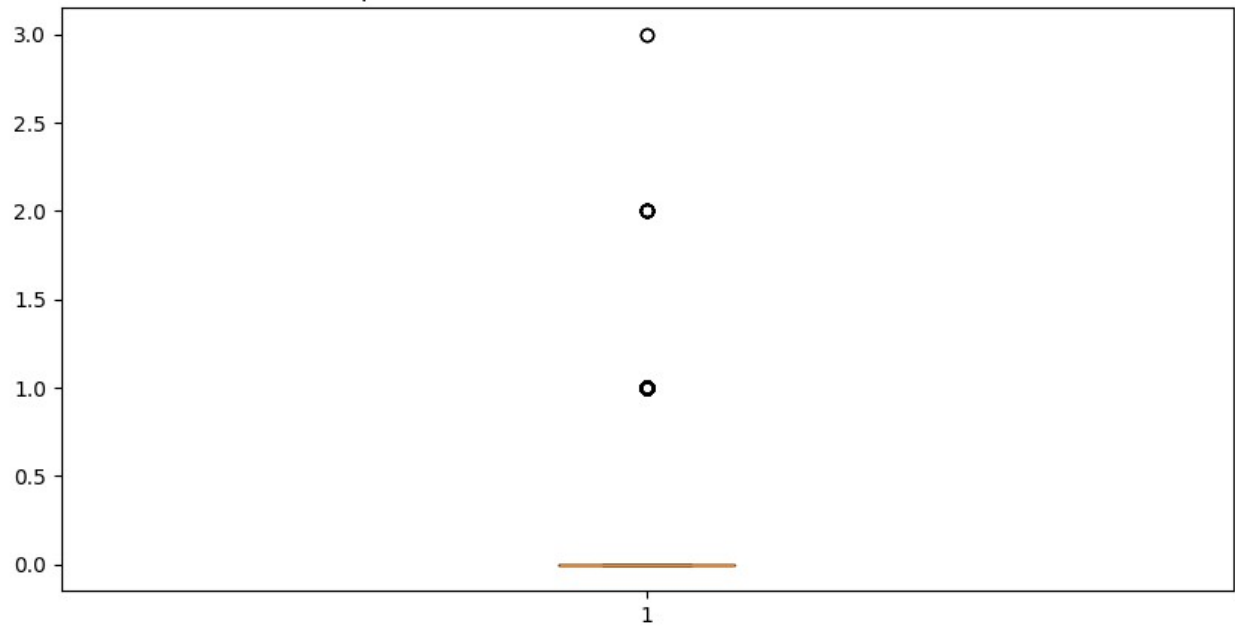
Boxplot of MOR BANK: GENERAL MERCHANDISE BUYER



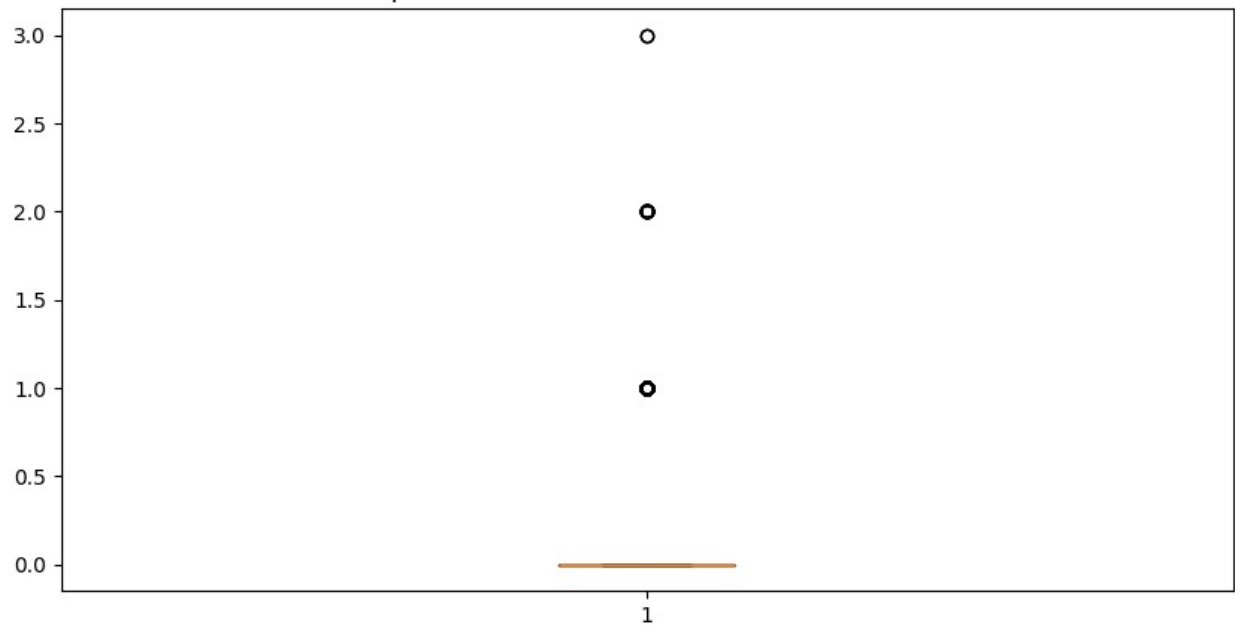
Boxplot of MOR BANK: FAMILY AND GENERAL MAGAZINE



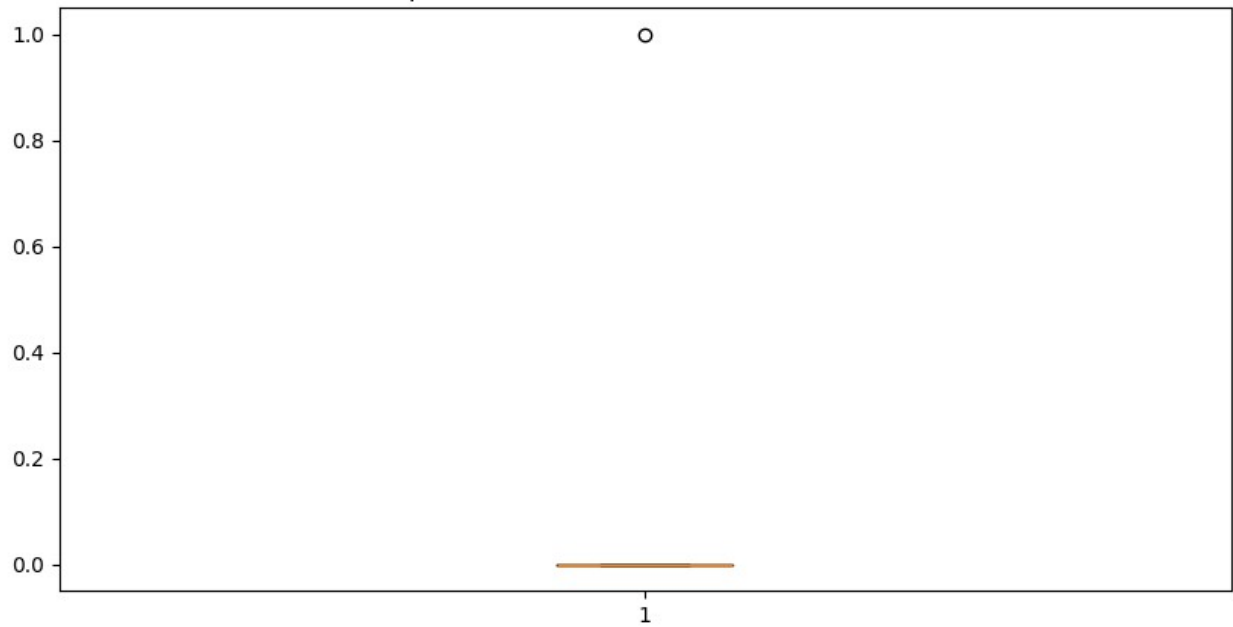
Boxplot of MOR BANK: FEMALE ORIENTED MAGAZINE



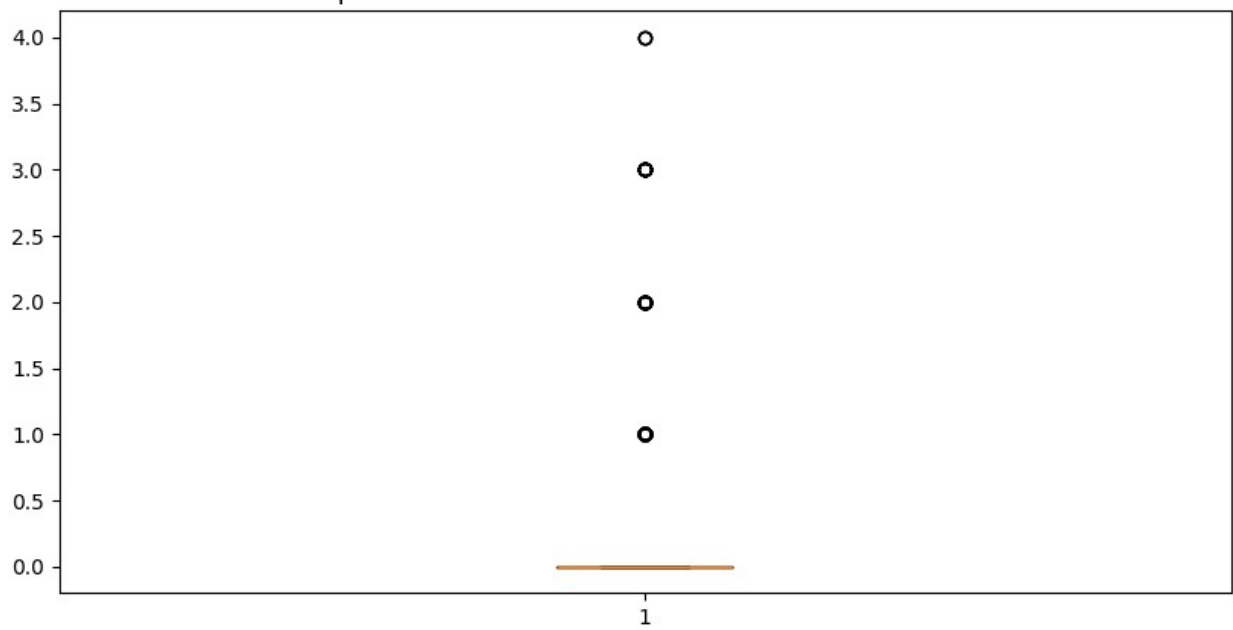
Boxplot of MOR BANK: MALE SPORTS MAGAZINE



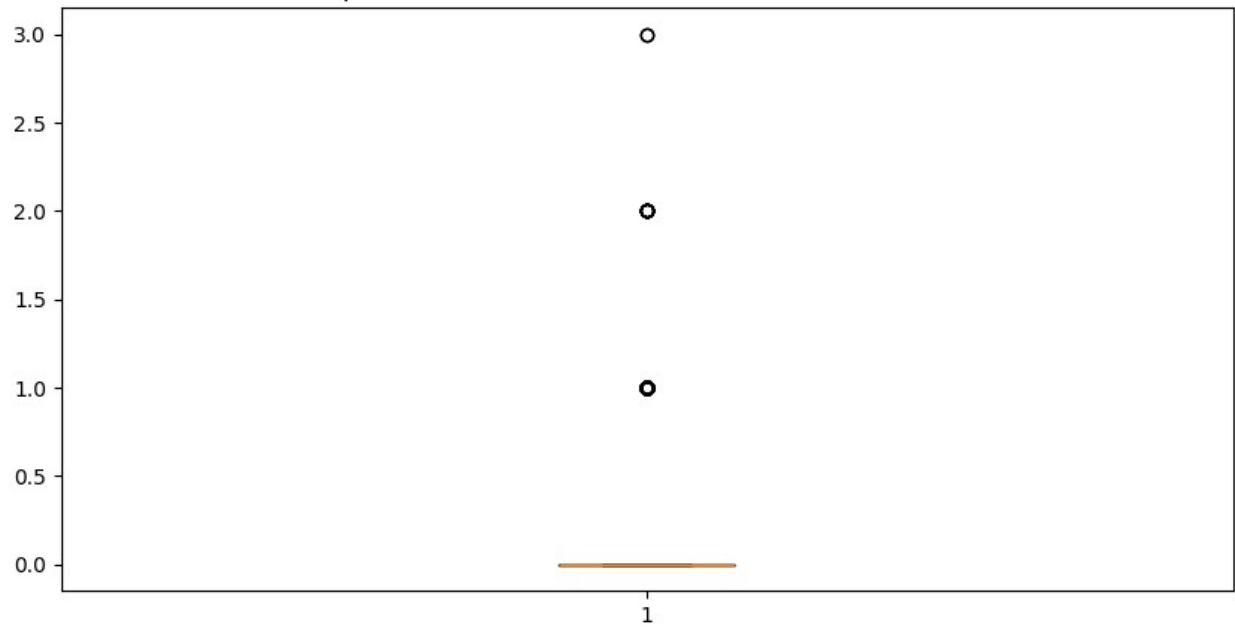
Boxplot of MOR BANK: RELIGIOUS MAGAZINE



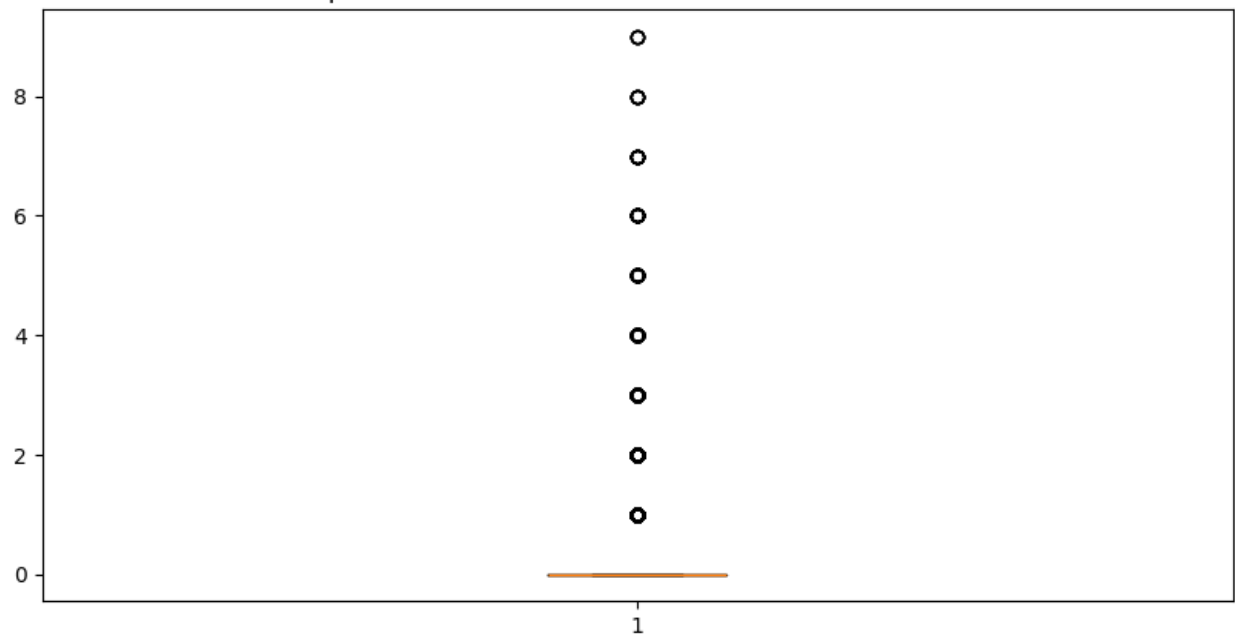
Boxplot of MOR BANK: GARDENING-FARMING MAGAZINE



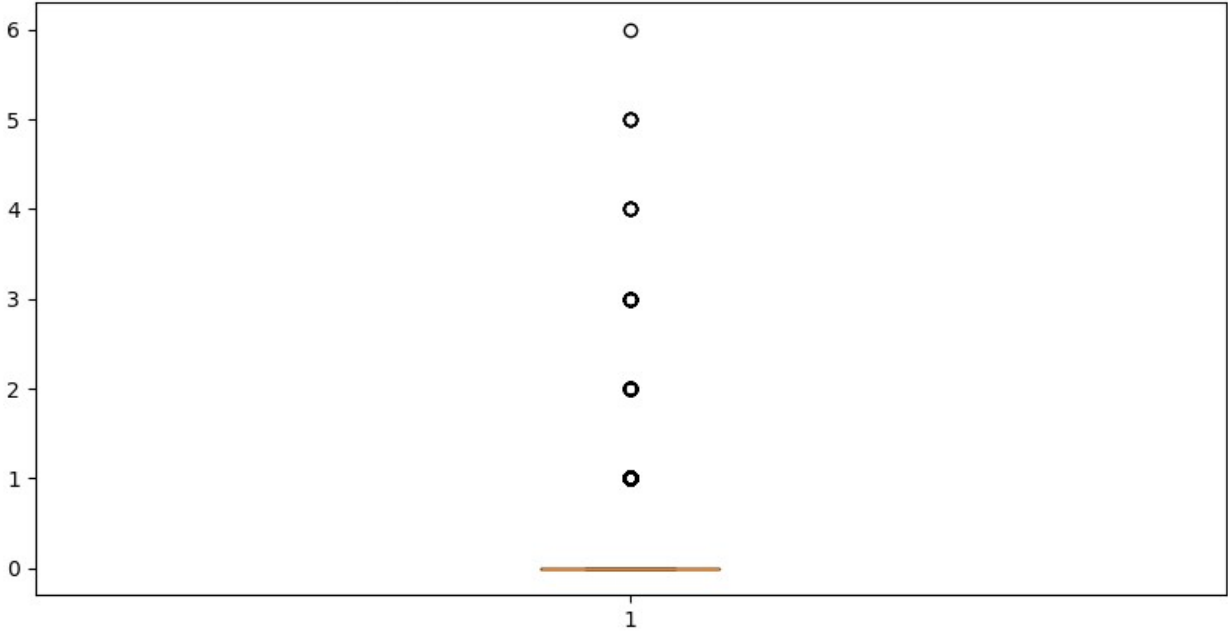
Boxplot of MOR BANK: CULINARY INTERESTS MAGAZINE



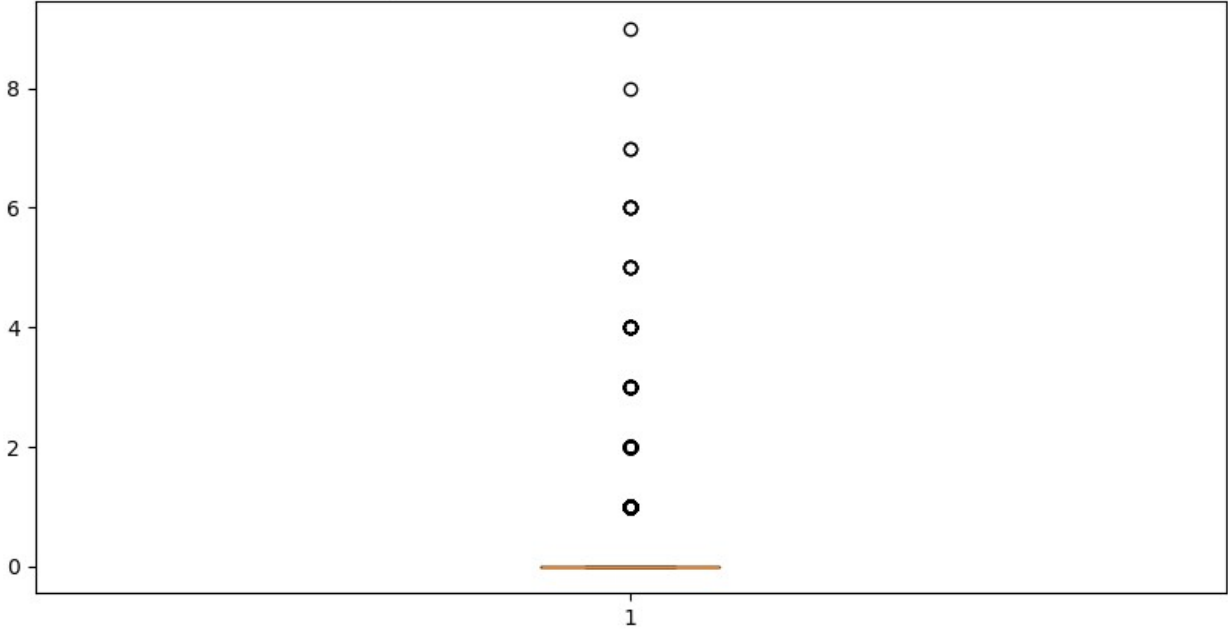
Boxplot of MOR BANK: HEALTH AND FITNESS MAGAZINE



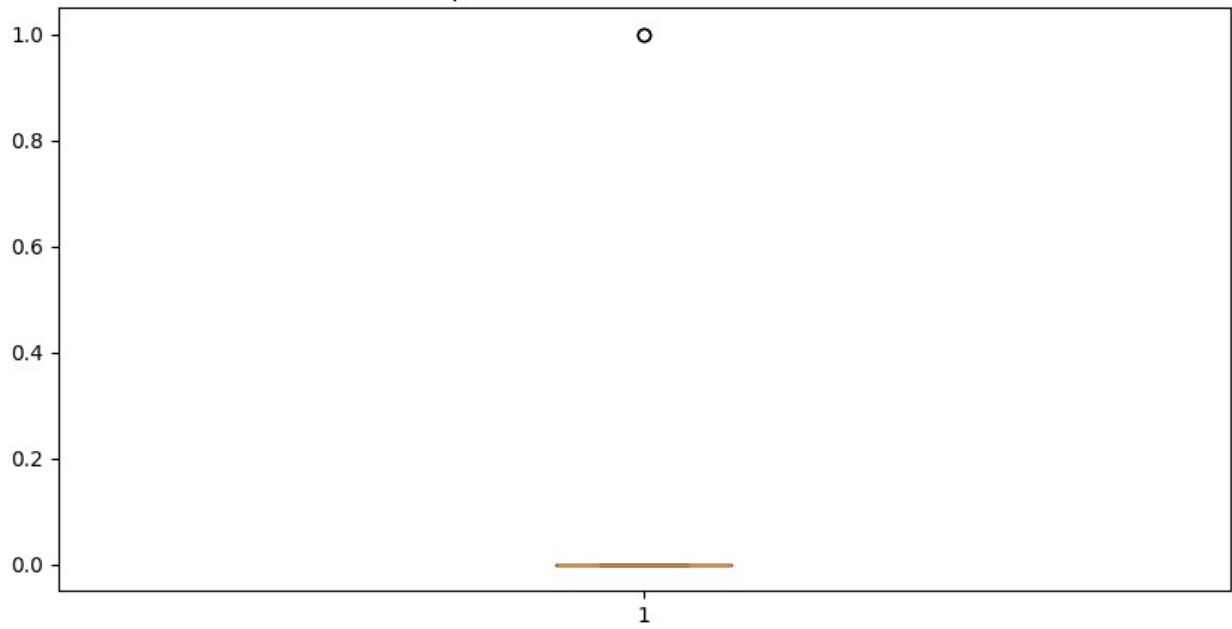
Boxplot of MOR BANK: DO-IT-YOURSELFERS



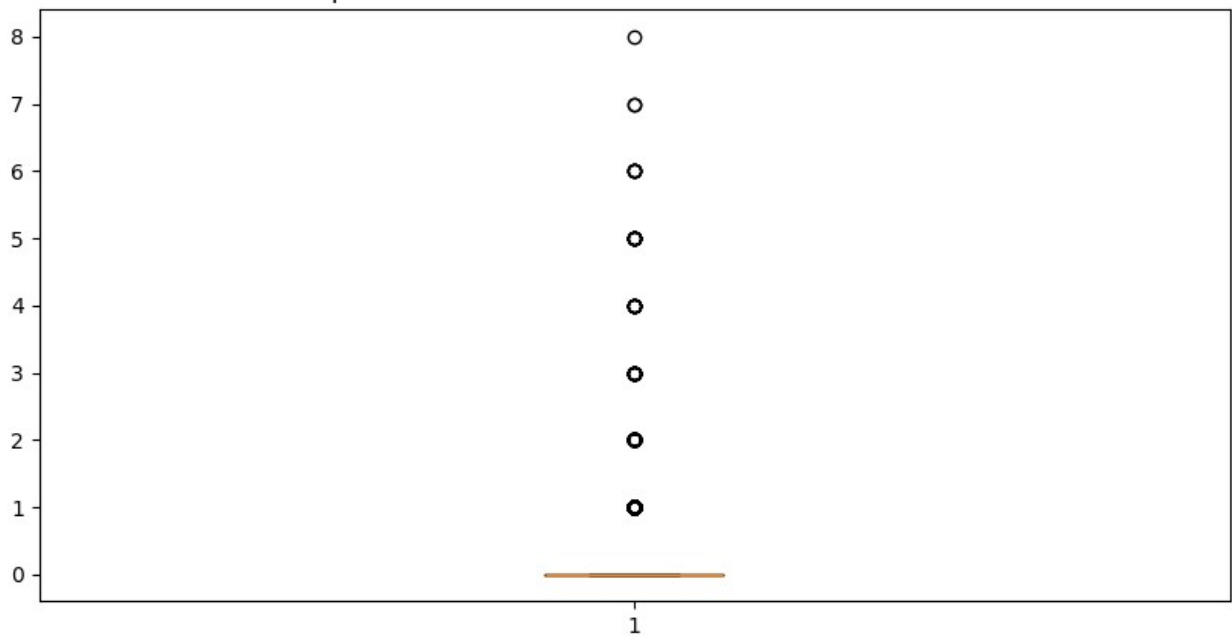
Boxplot of MOR BANK: NEWS AND FINANCIAL



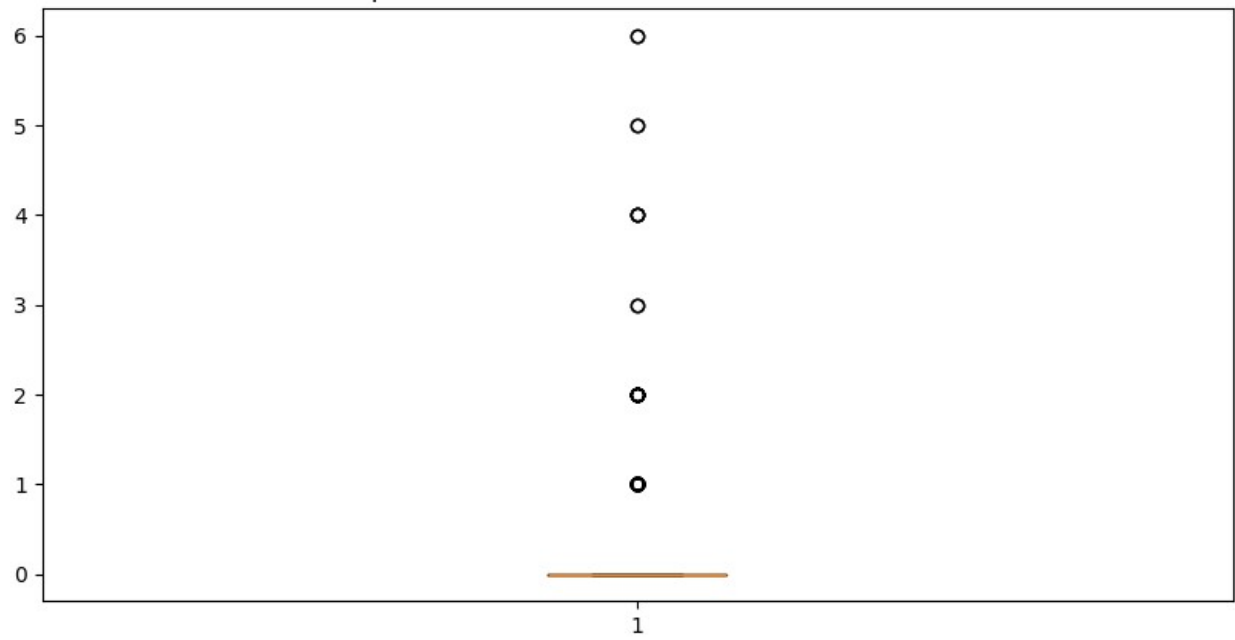
Boxplot of MOR BANK: PHOTOGRAPHY



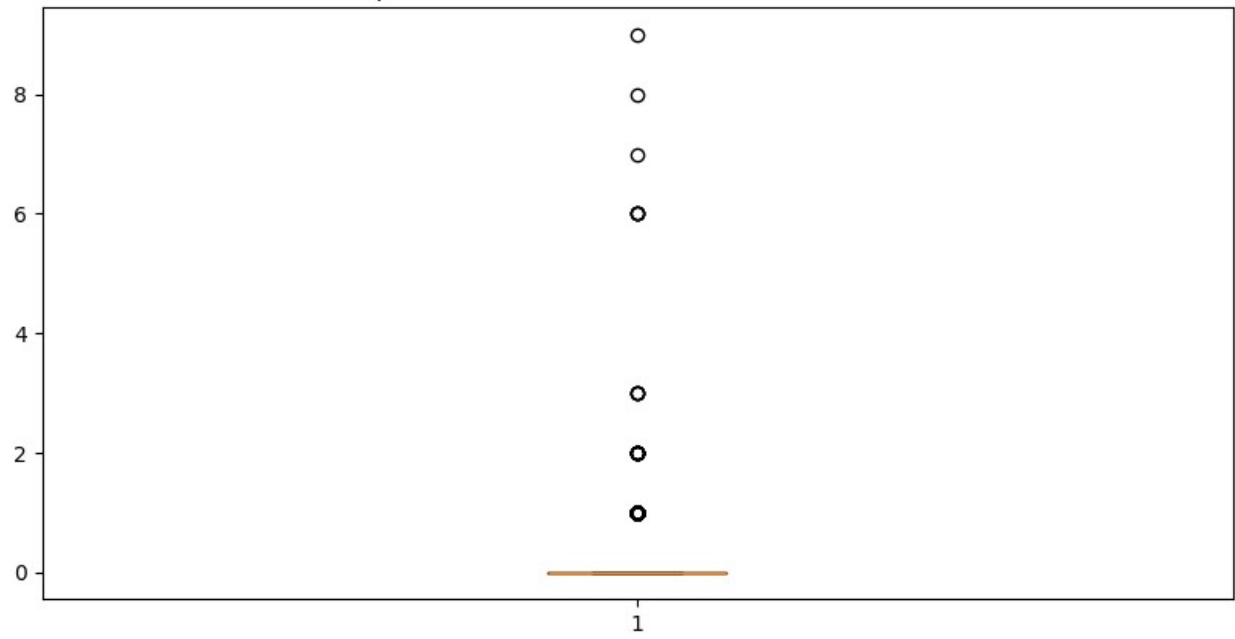
Boxplot of MOR BANK: OPPORTUNITY SEEKERS AND CE



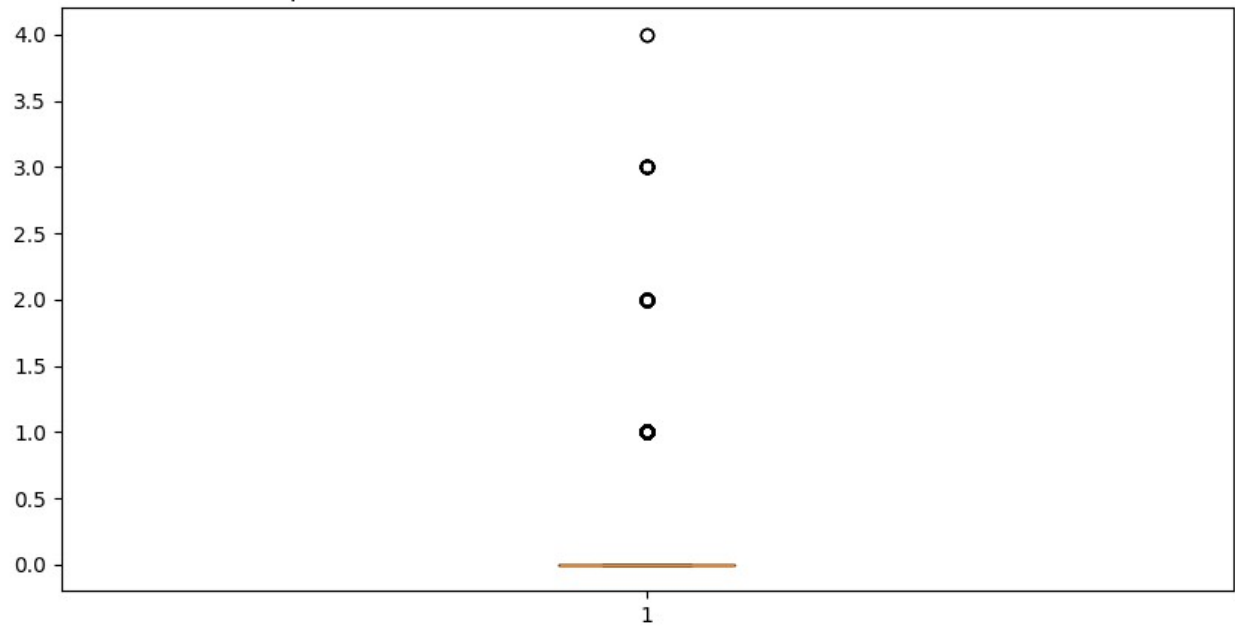
Boxplot of MOR BANK: RELIGIOUS CONTRIBUTOR



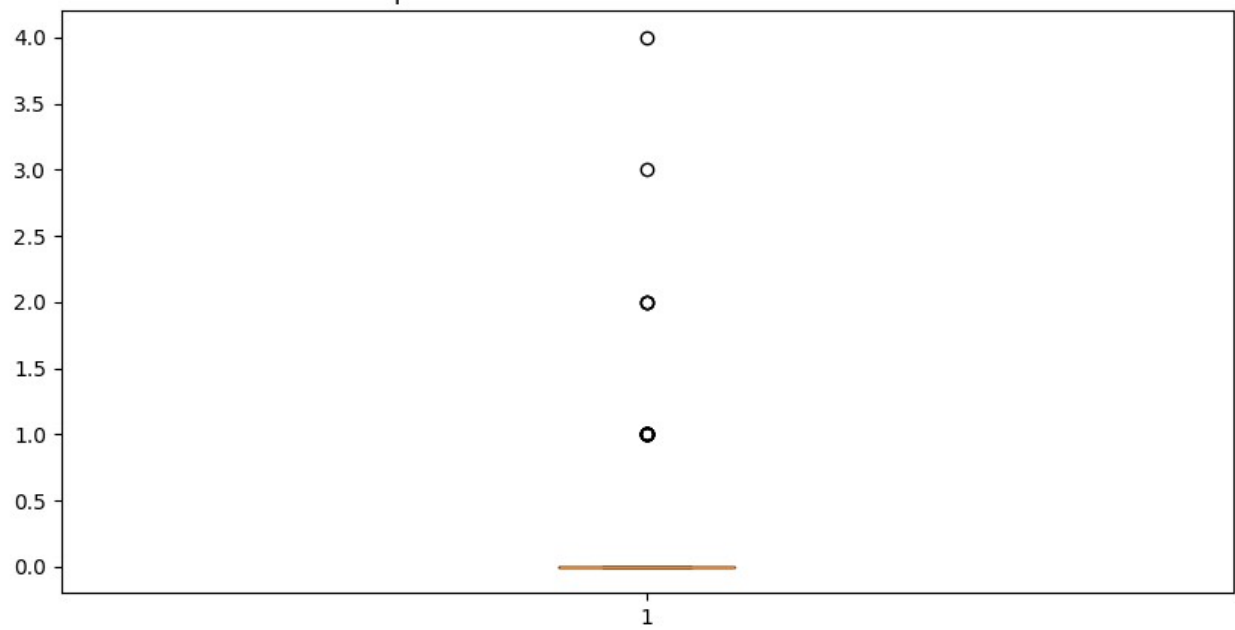
Boxplot of MOR BANK: POLITICAL CONTRIBUTOR



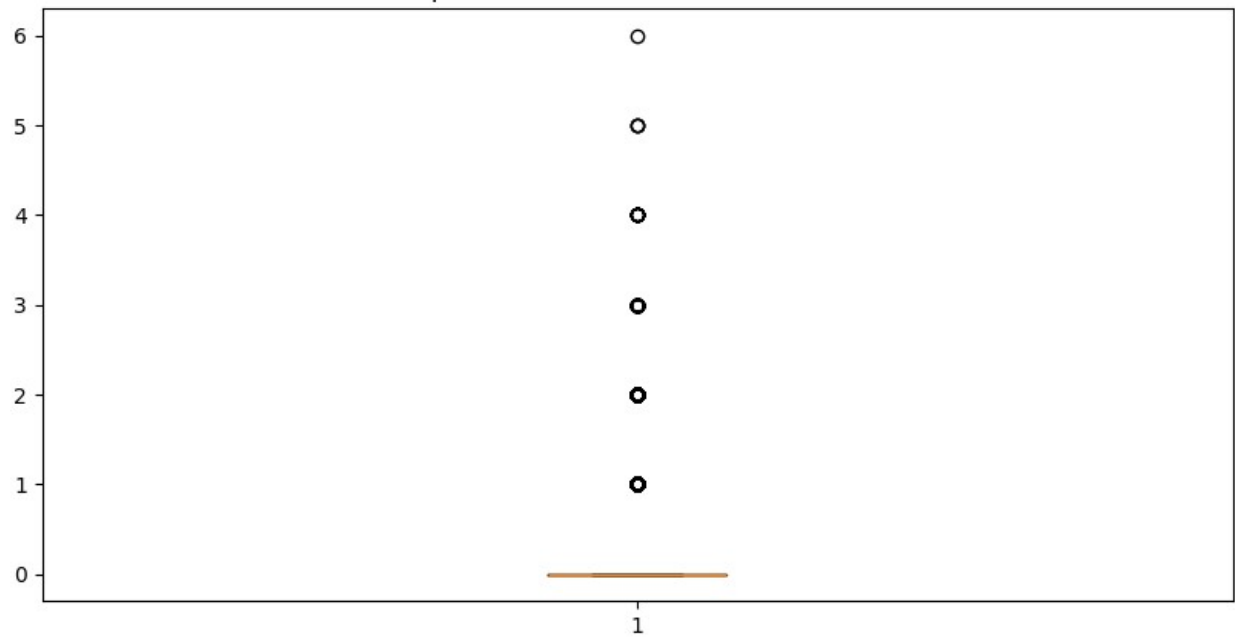
Boxplot of MOR BANK: HEALTH AND INSTITUTION CONTRIBUTOR



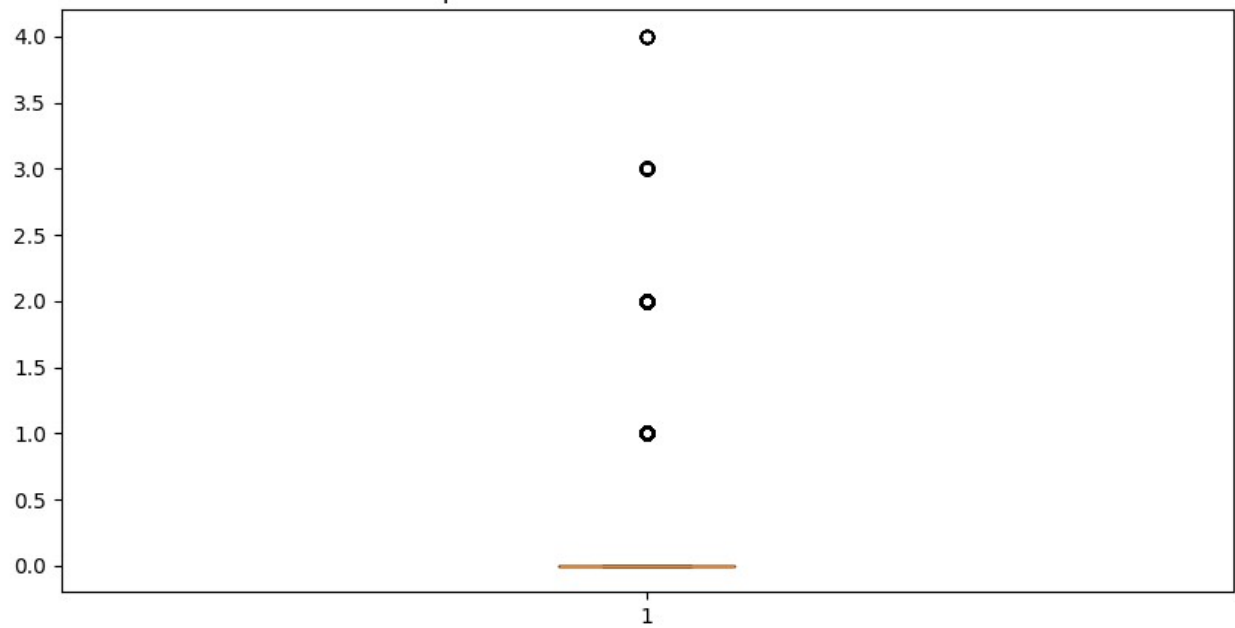
Boxplot of MOR BANK: GENERAL CONTRIBUTOR



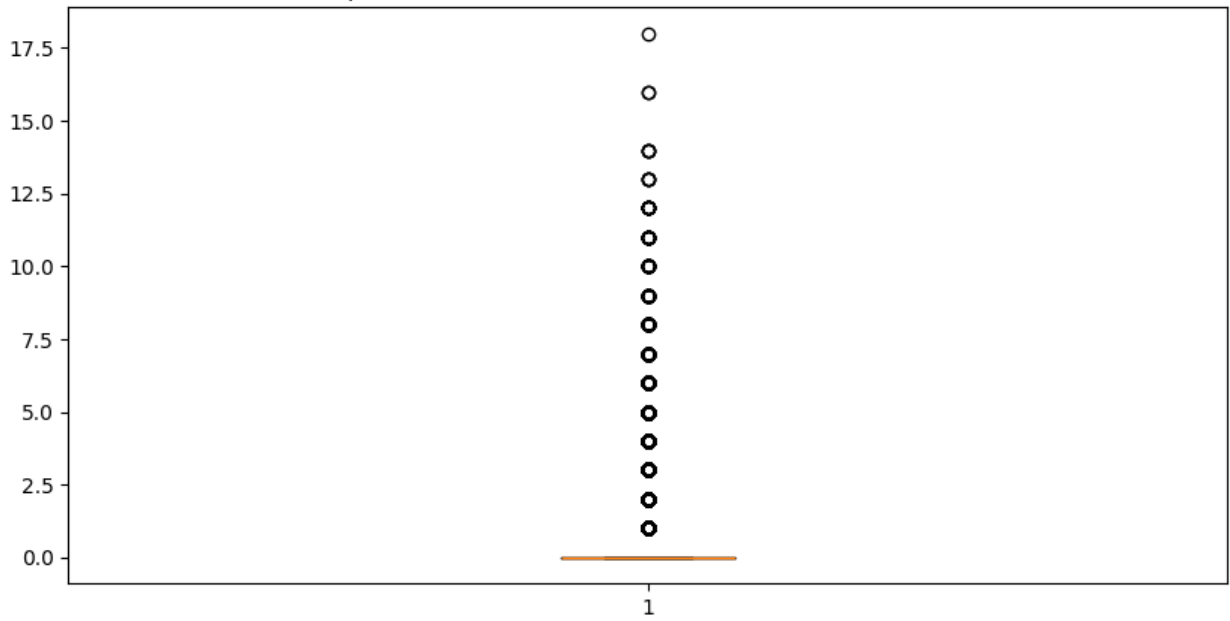
Boxplot of MOR BANK: MISCELLANEOUS



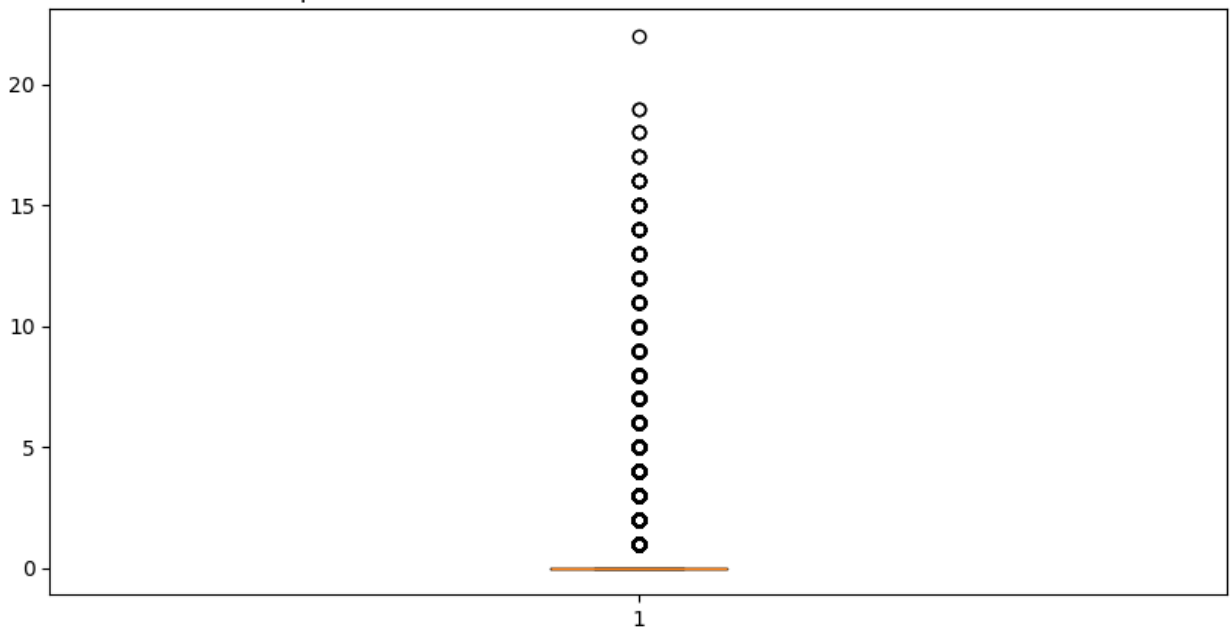
Boxplot of MOR BANK: ODDS AND ENDS



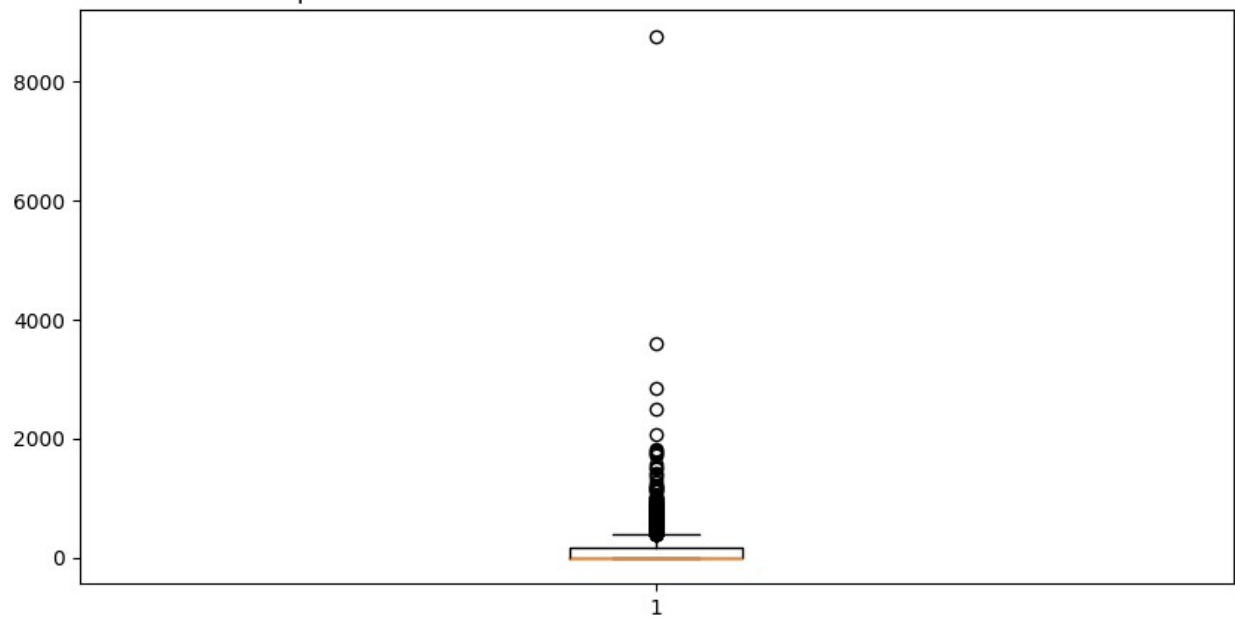
Boxplot of MOR BANK: DEDUPED CATEGORY HIT COUNT



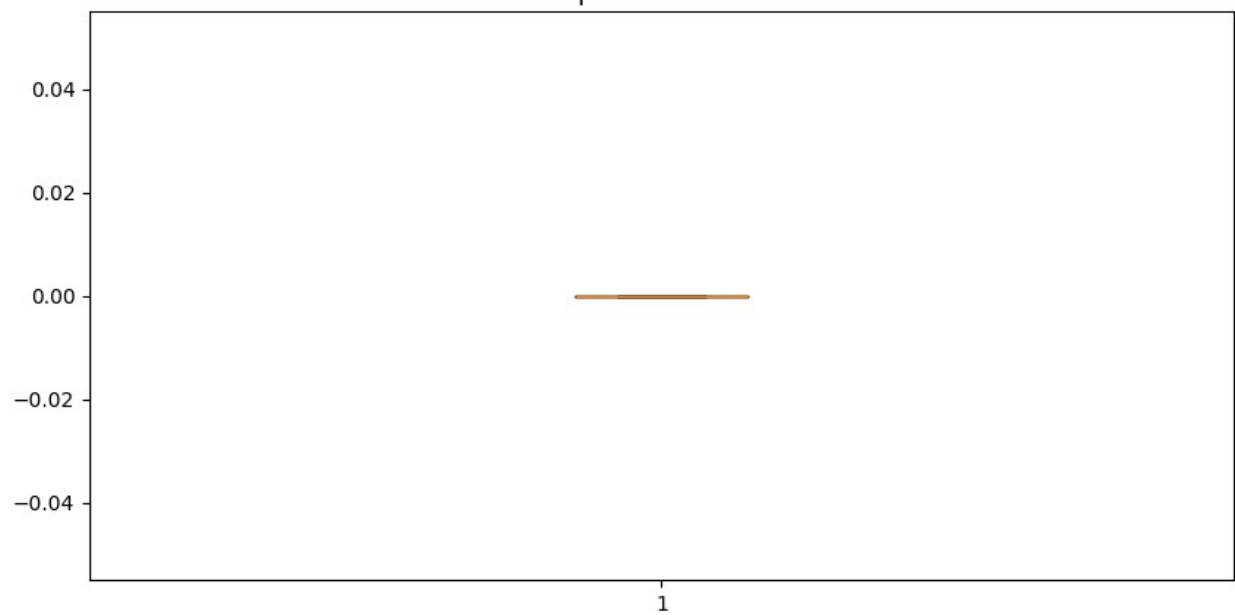
Boxplot of MOR BANK: NON-DEDUPED CATEGORY HIT COUNT



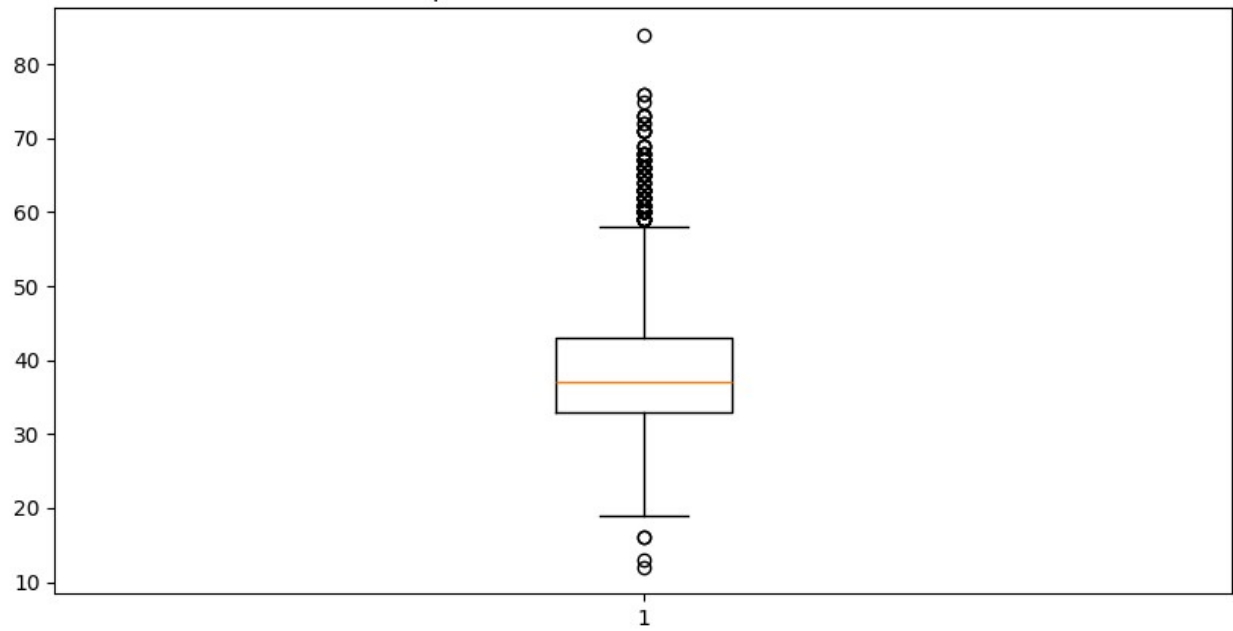
Boxplot of MORTGAGE-HOME PURCHASE: HOME PURCHASE PRICE



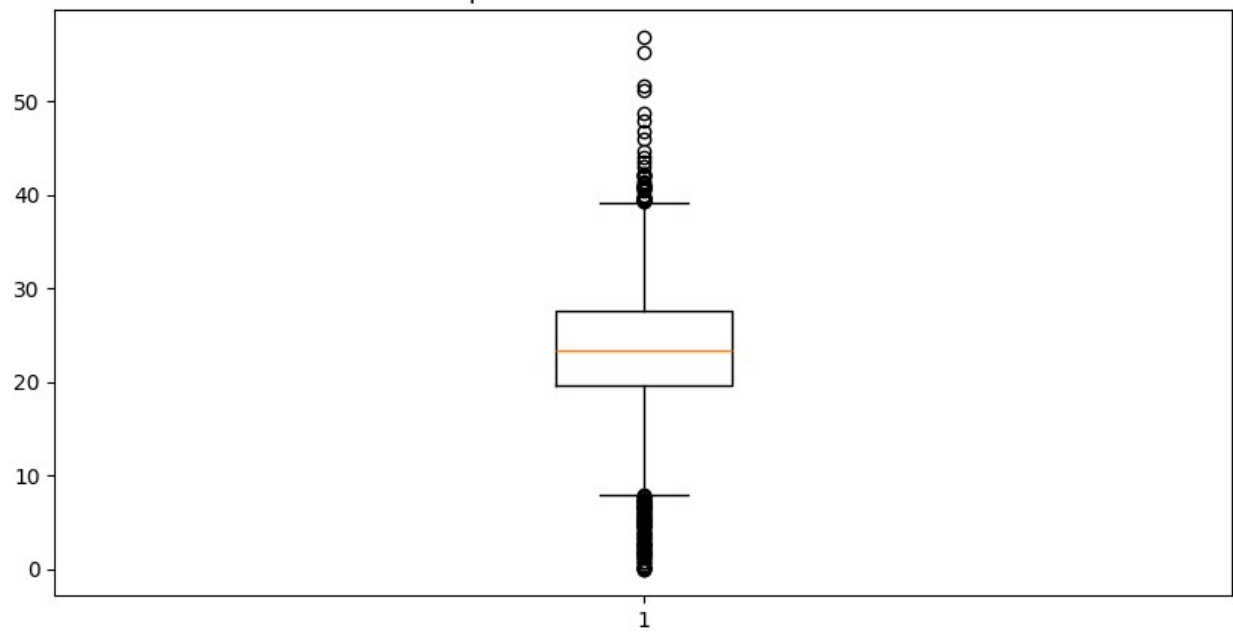
Boxplot of CHILDREN



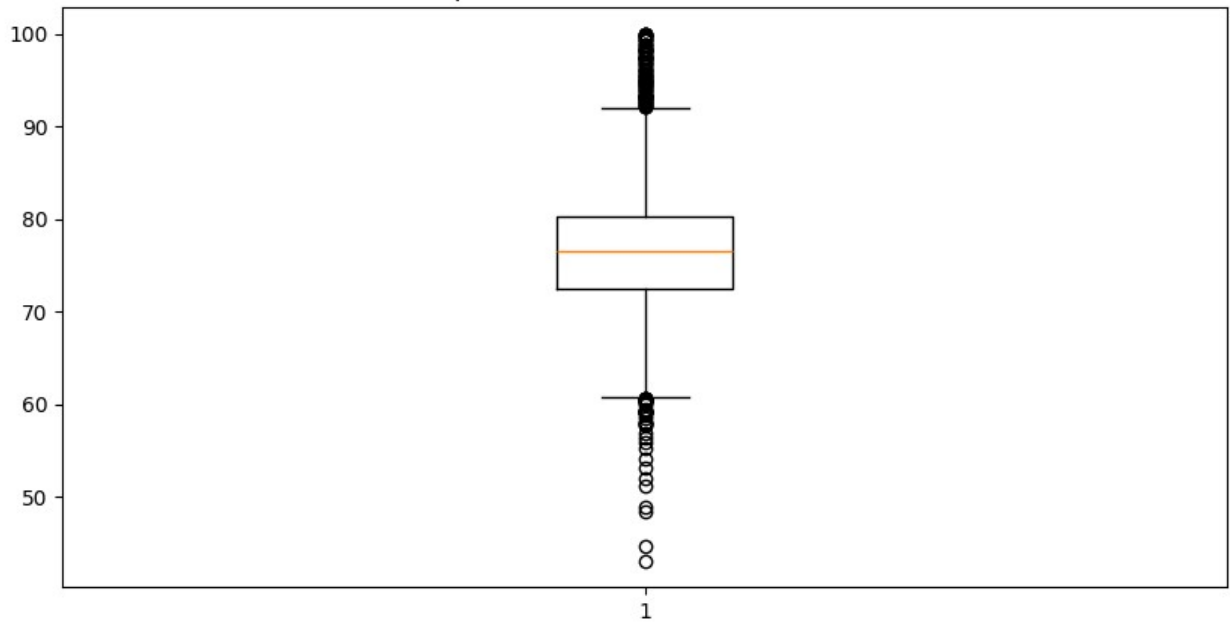
Boxplot of CAPE: AGE: POP: MEDIAN AGE



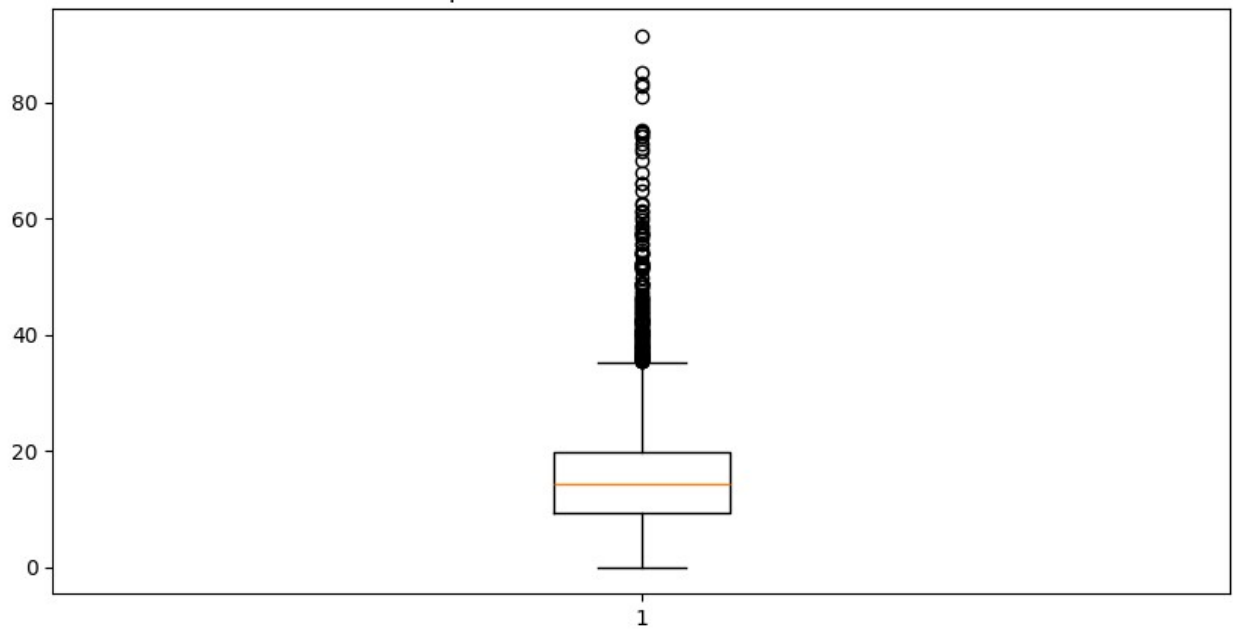
Boxplot of CAPE: AGE: POP: % 0-17



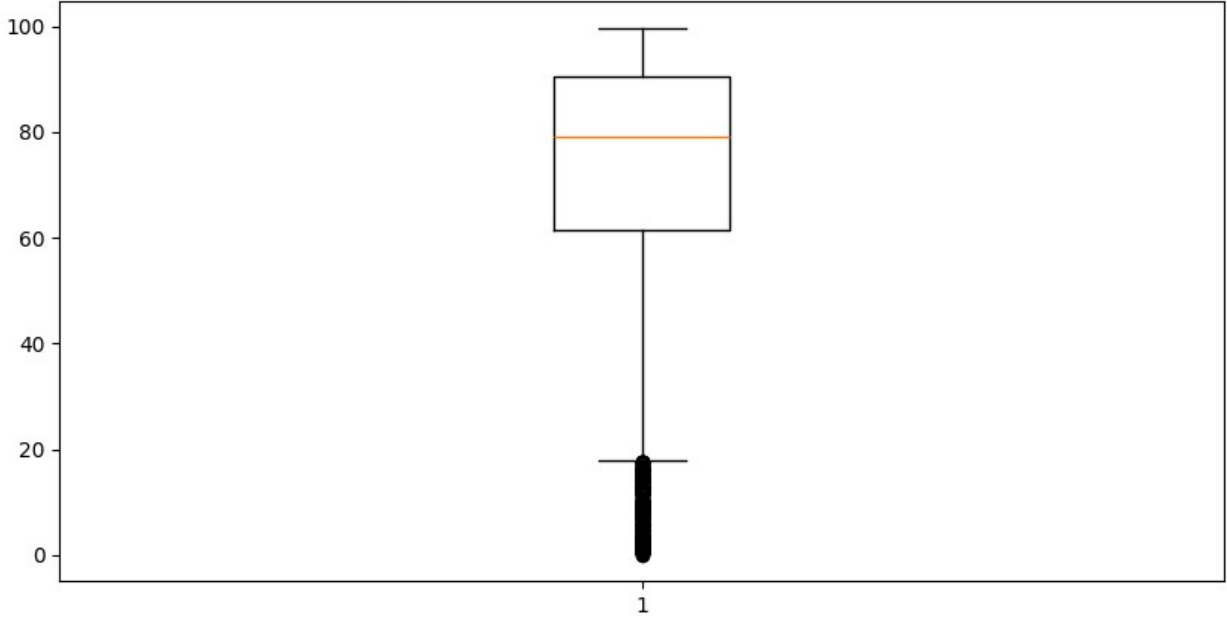
Boxplot of CAPE: AGE: POP: % 18-99+



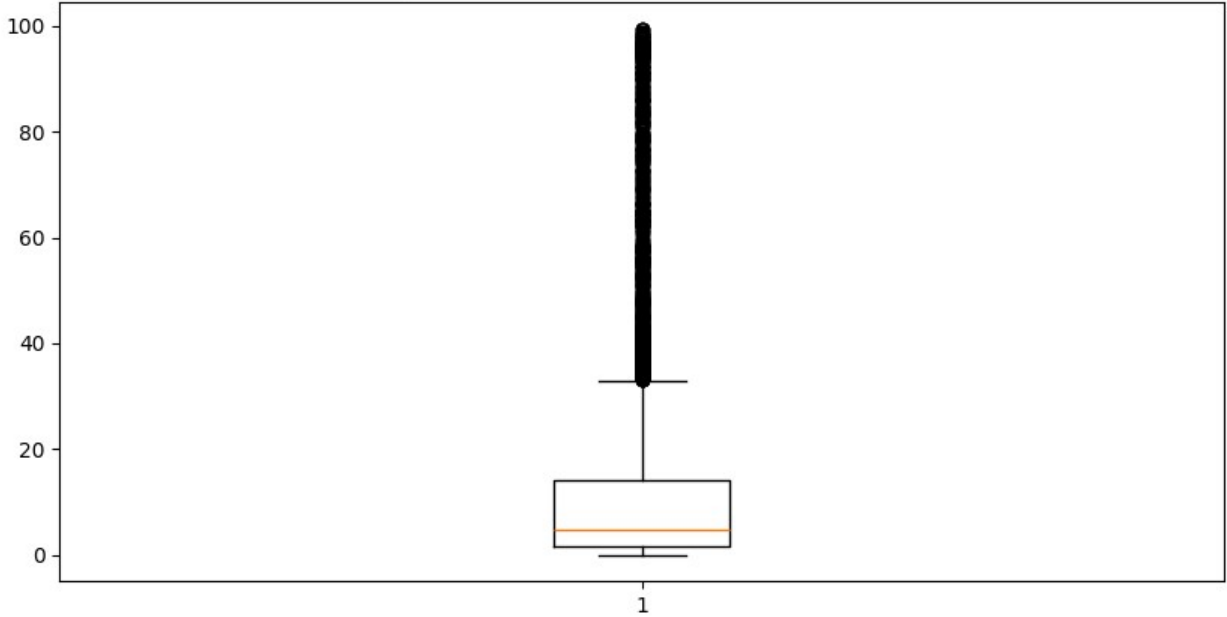
Boxplot of CAPE: AGE: POP: % 65-99+



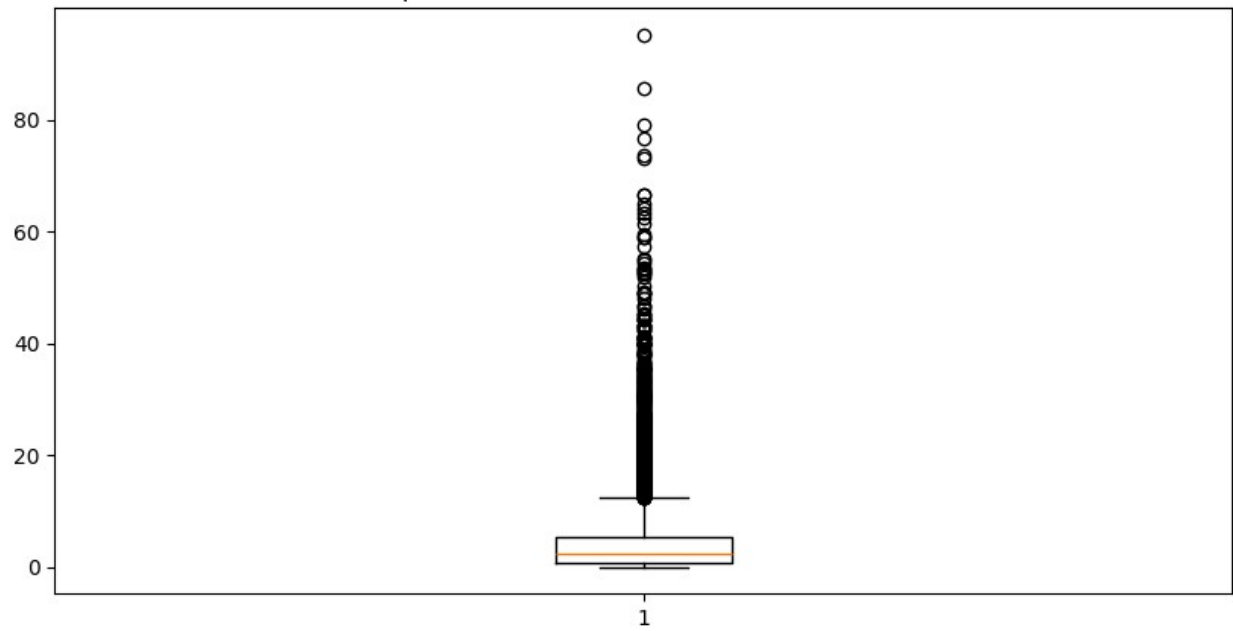
Boxplot of CAPE: ETHNIC: POP: % WHITE ONLY



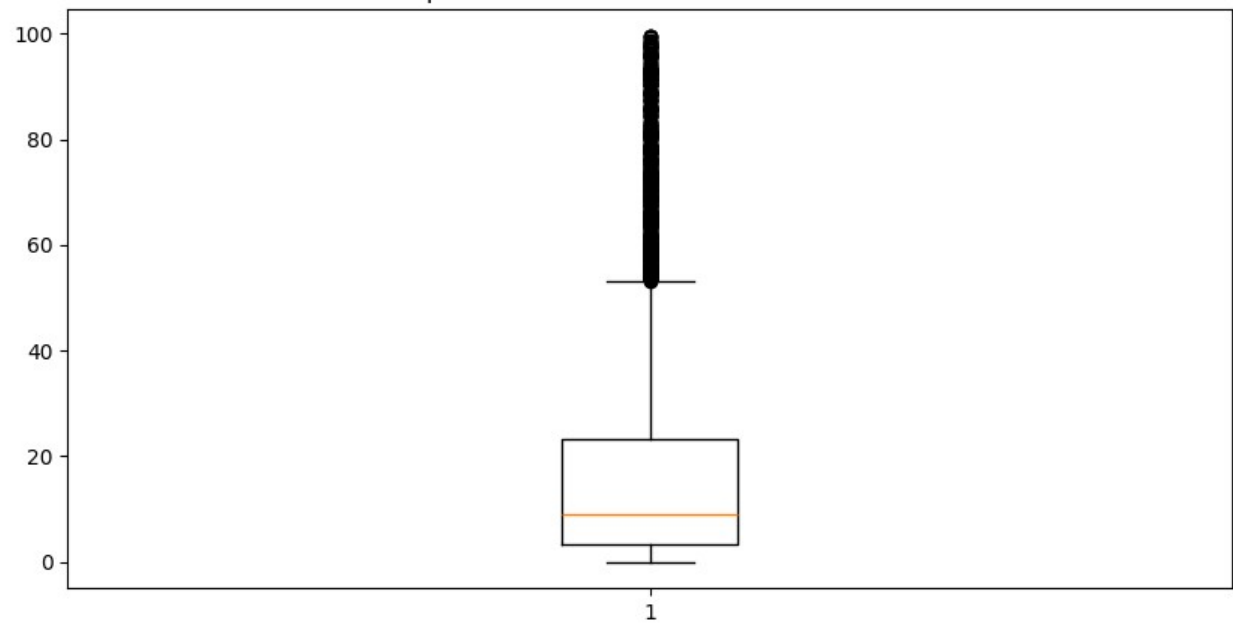
Boxplot of CAPE: ETHNIC: POP: % BLACK ONLY



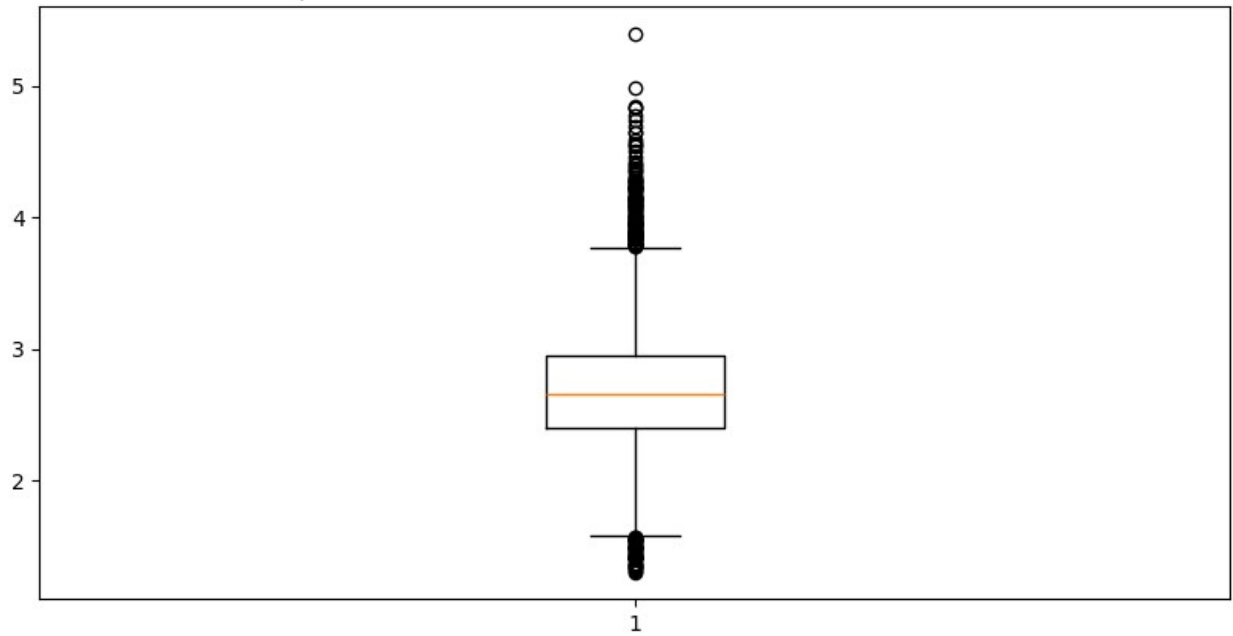
Boxplot of CAPE: ETHNIC: POP: % ASIAN ONLY



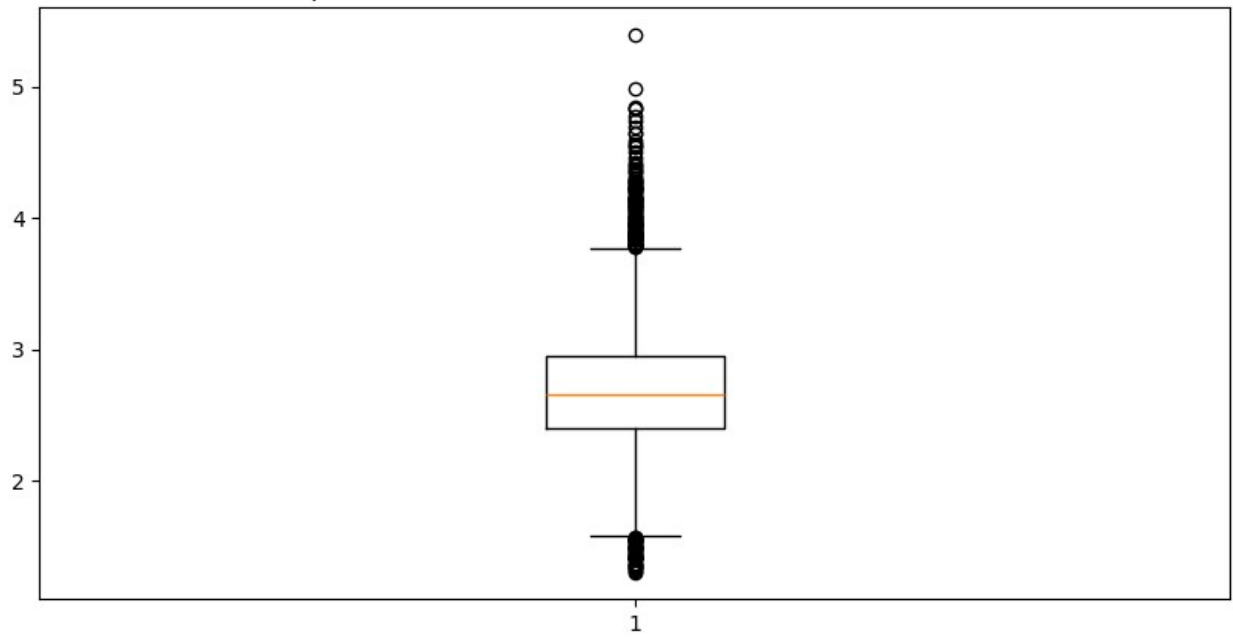
Boxplot of CAPE: ETHNIC: POP: % HISPANIC



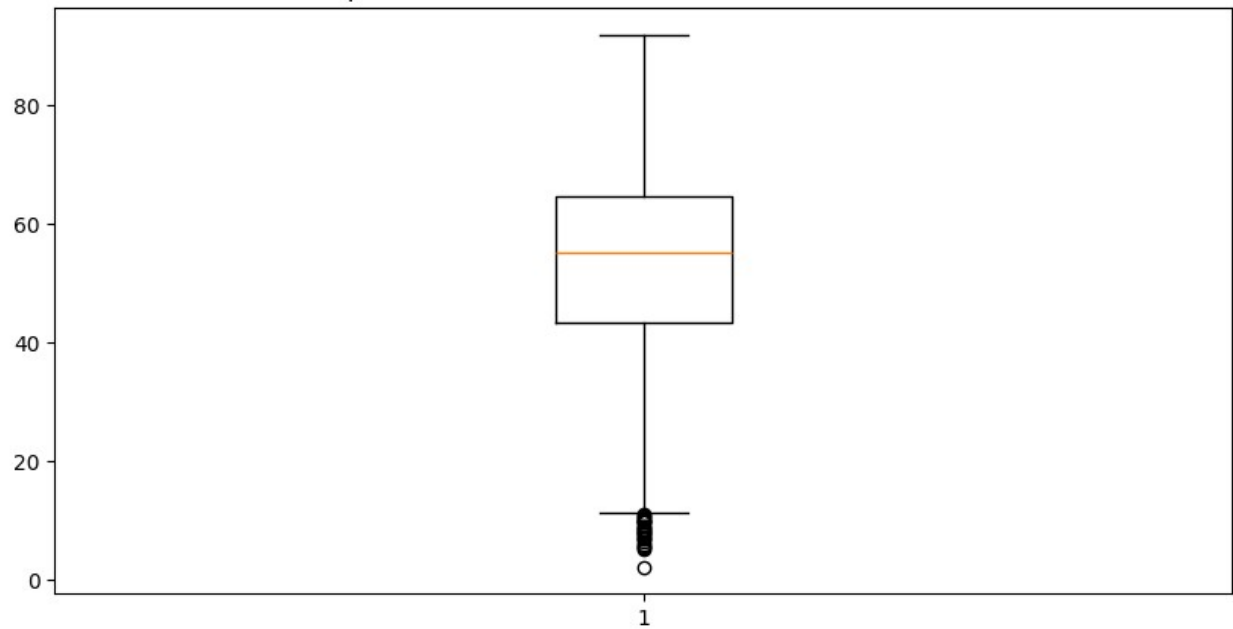
Boxplot of CAPE: DENSITY: PERSONS PER HH FOR POP IN HH



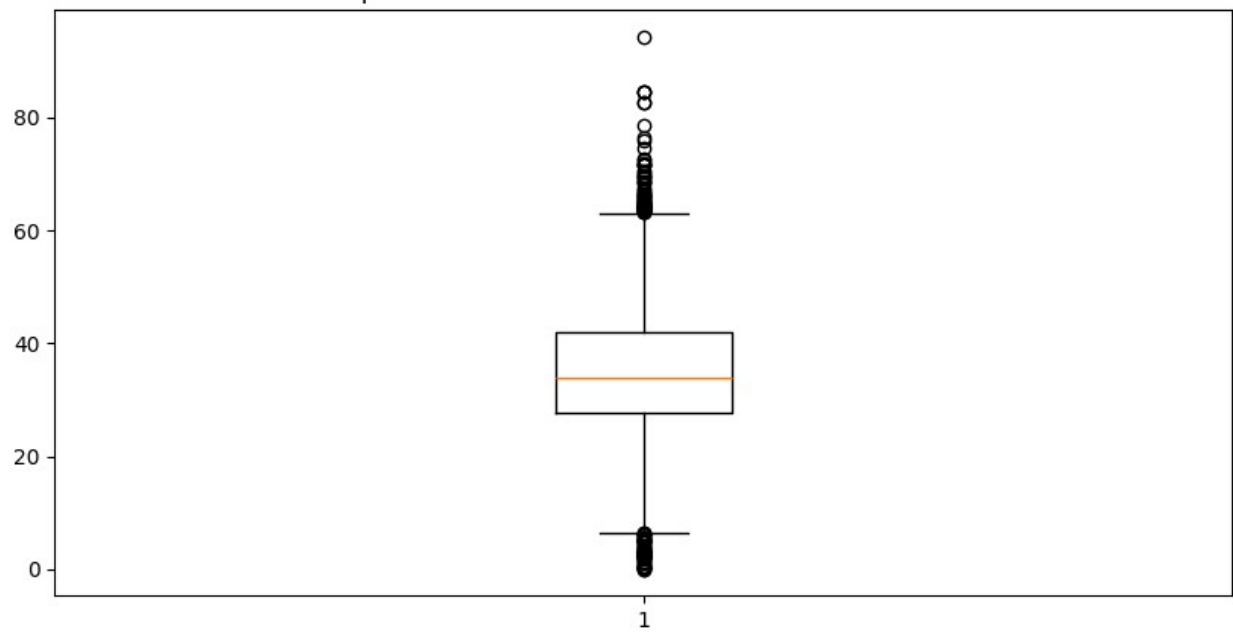
Boxplot of CAPE: HHSIZE: HH: AVERAGE HOUSEHOLD SIZE



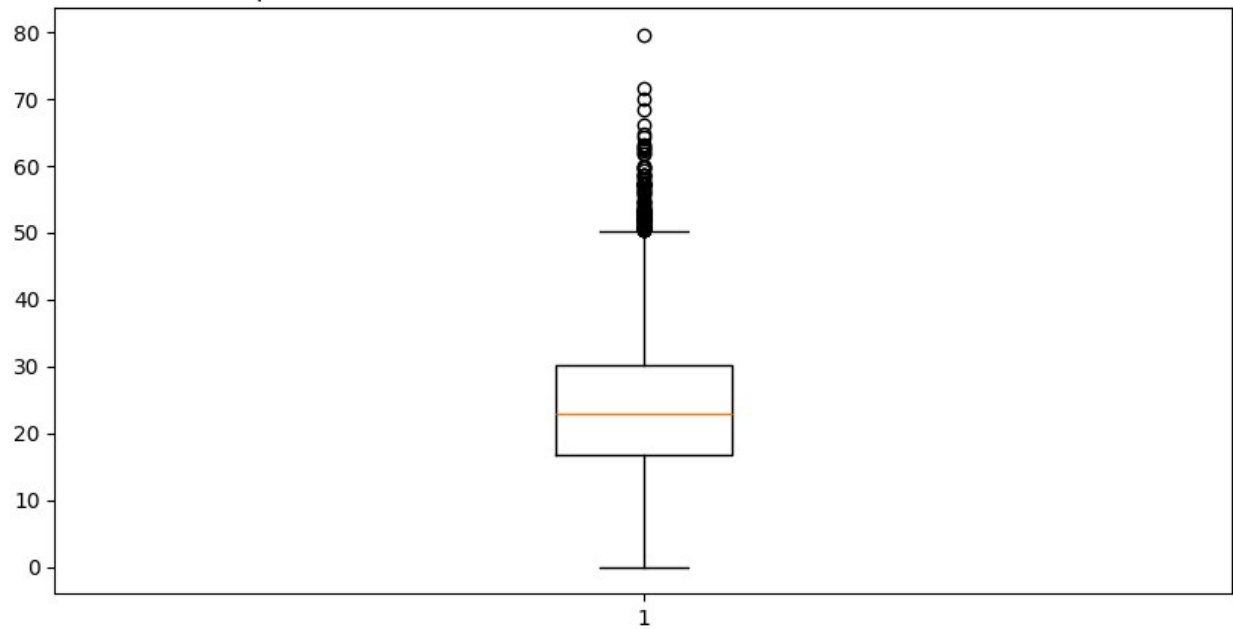
Boxplot of CAPE: TYP: HH: % MARRIED COUPLE FAMILY



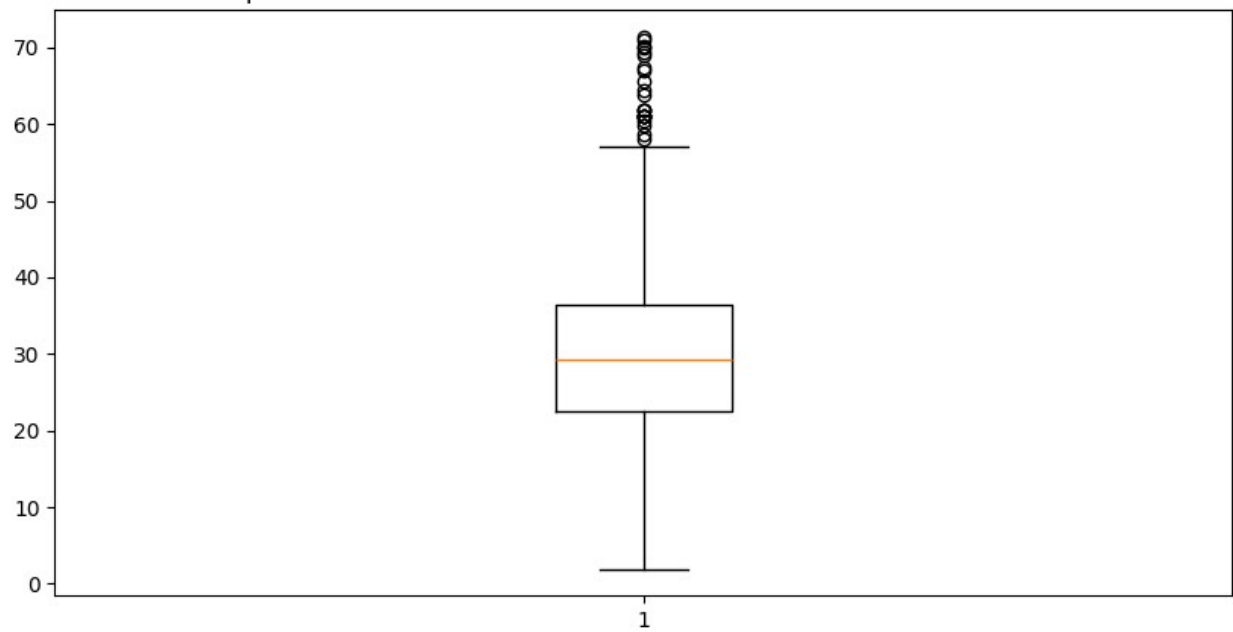
Boxplot of CAPE: CHILD: HH: % WITH PERSONS LT18



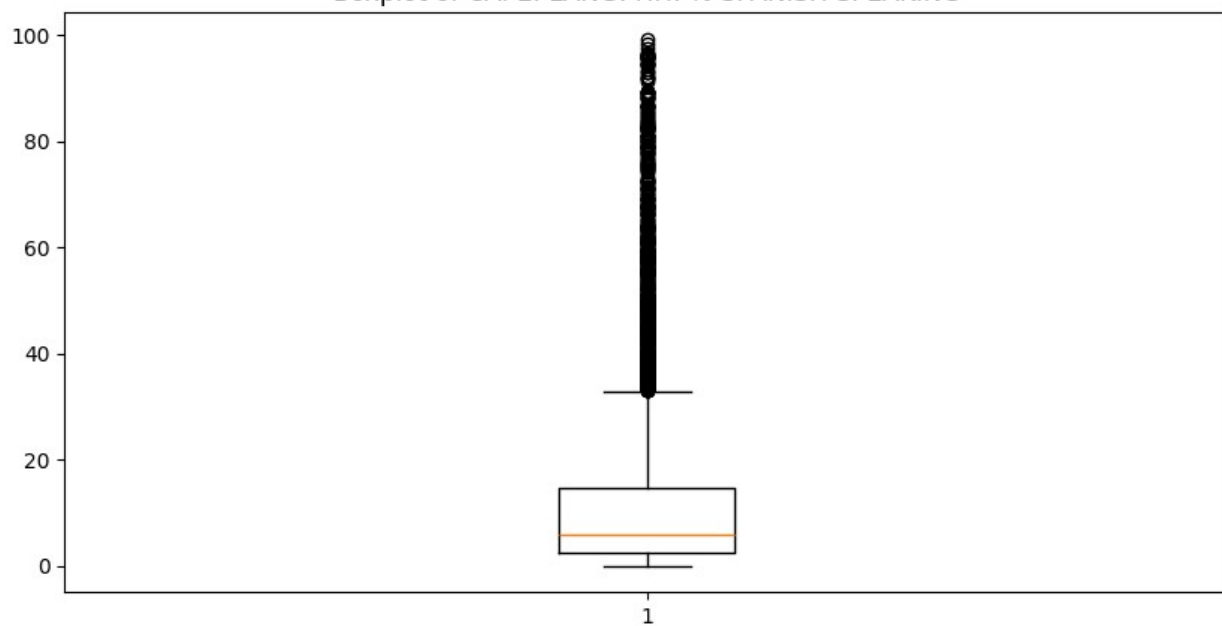
Boxplot of CAPE: CHILD: HH: % MARR COUPLE FAMW- PERSONS LT18



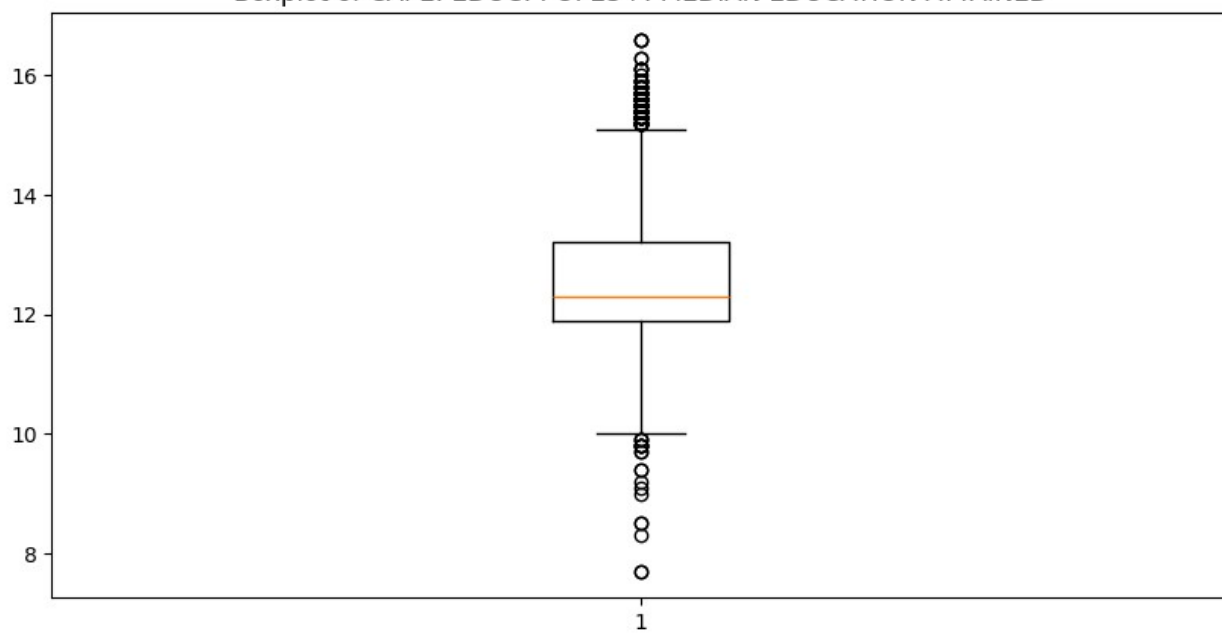
Boxplot of CAPE: CHILD: HH: % MARR COUPLE FAMW-O PERSONS LT18

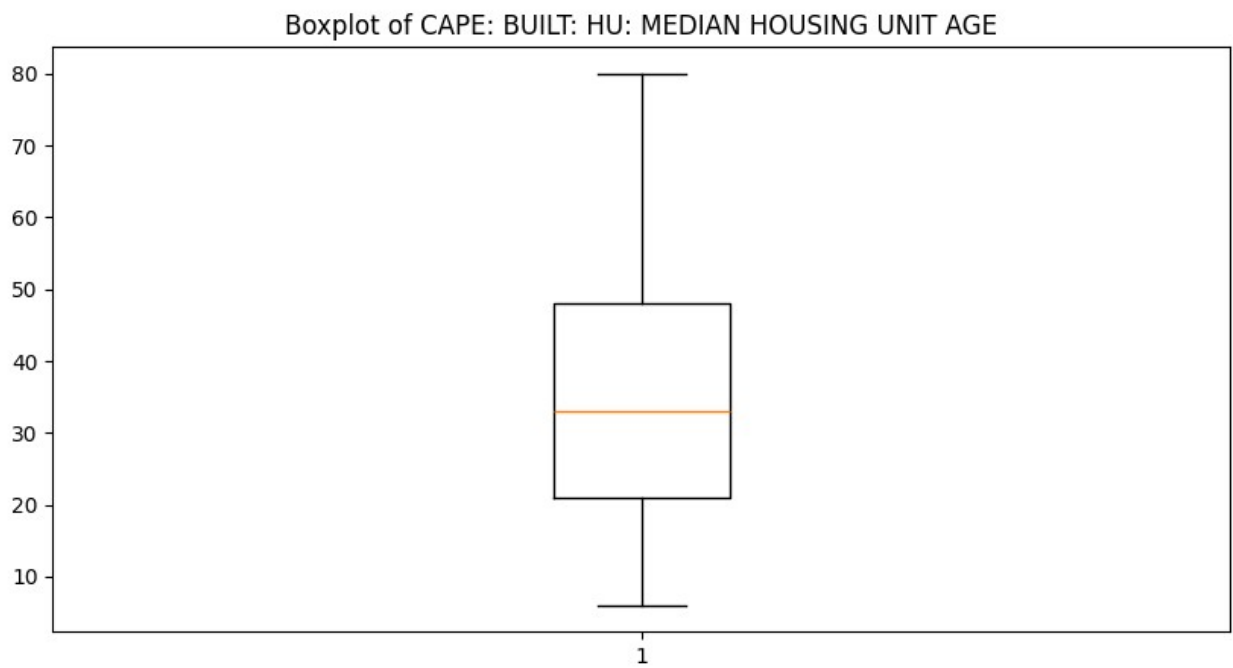
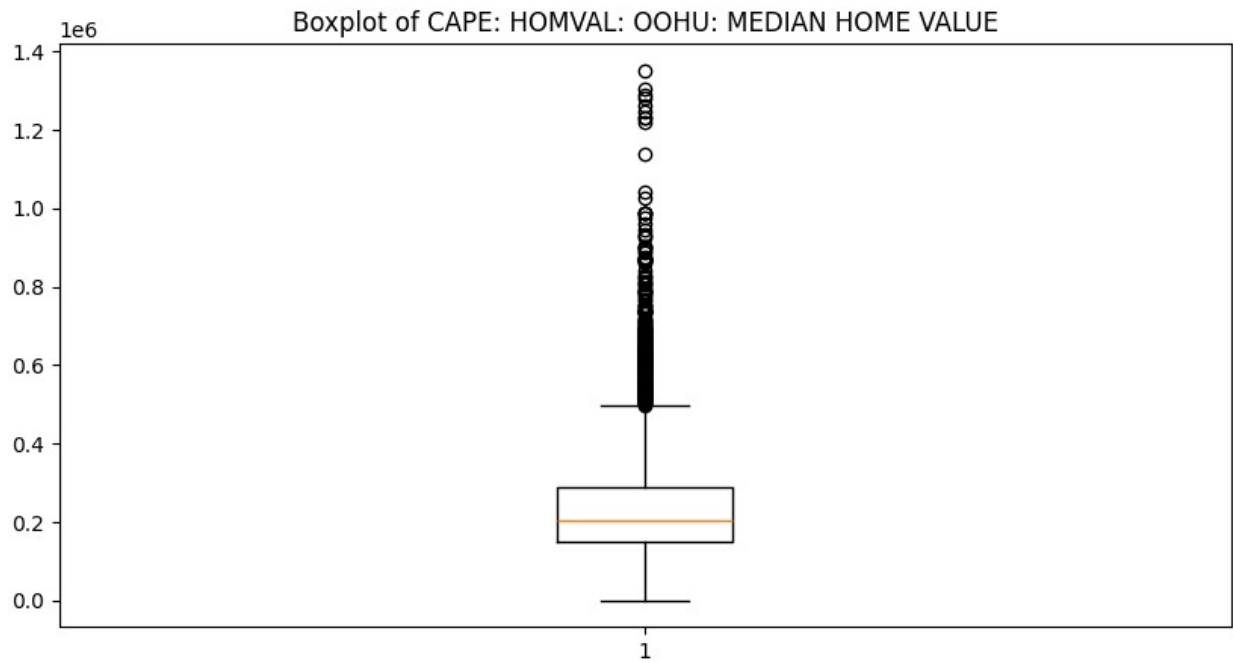


Boxplot of CAPE: LANG: HH: % SPANISH SPEAKING

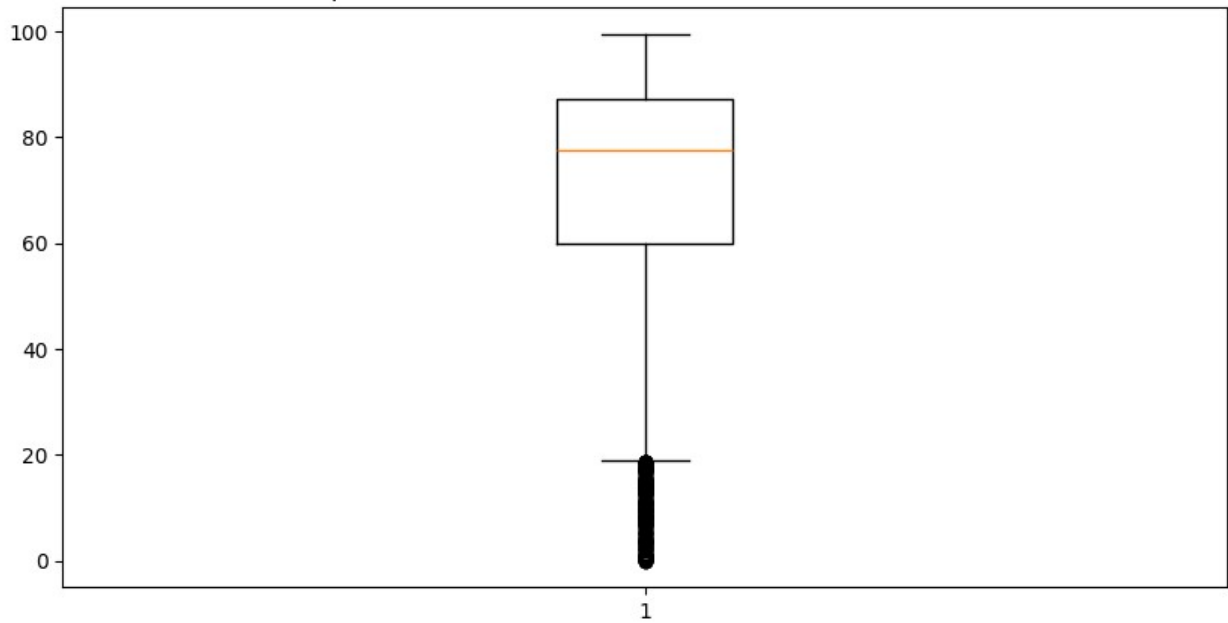


Boxplot of CAPE: EDUC: POP25+: MEDIAN EDUCATION ATTAINED

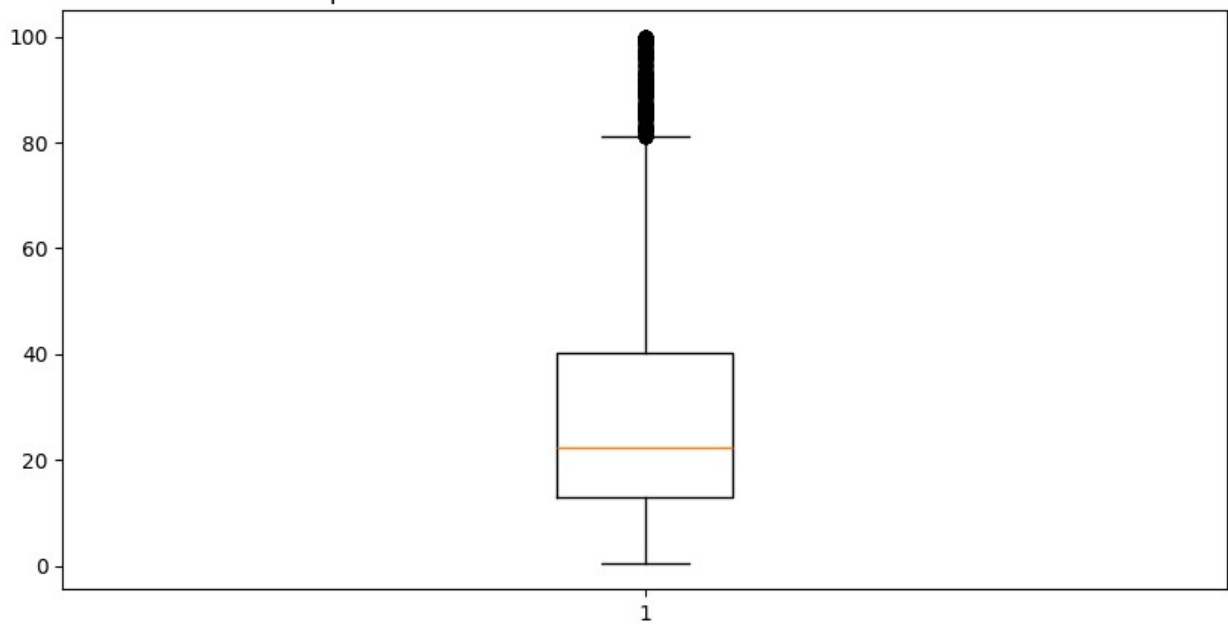




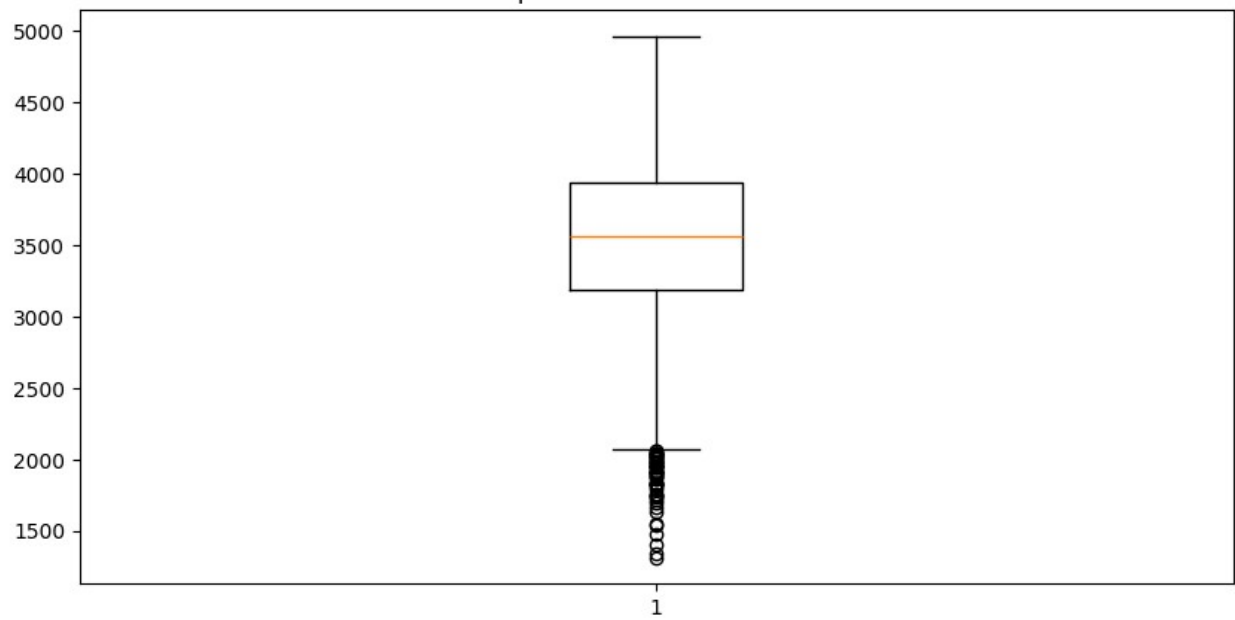
Boxplot of CAPE: TENANCY: OCCHU: % OWNER OCCUPIED



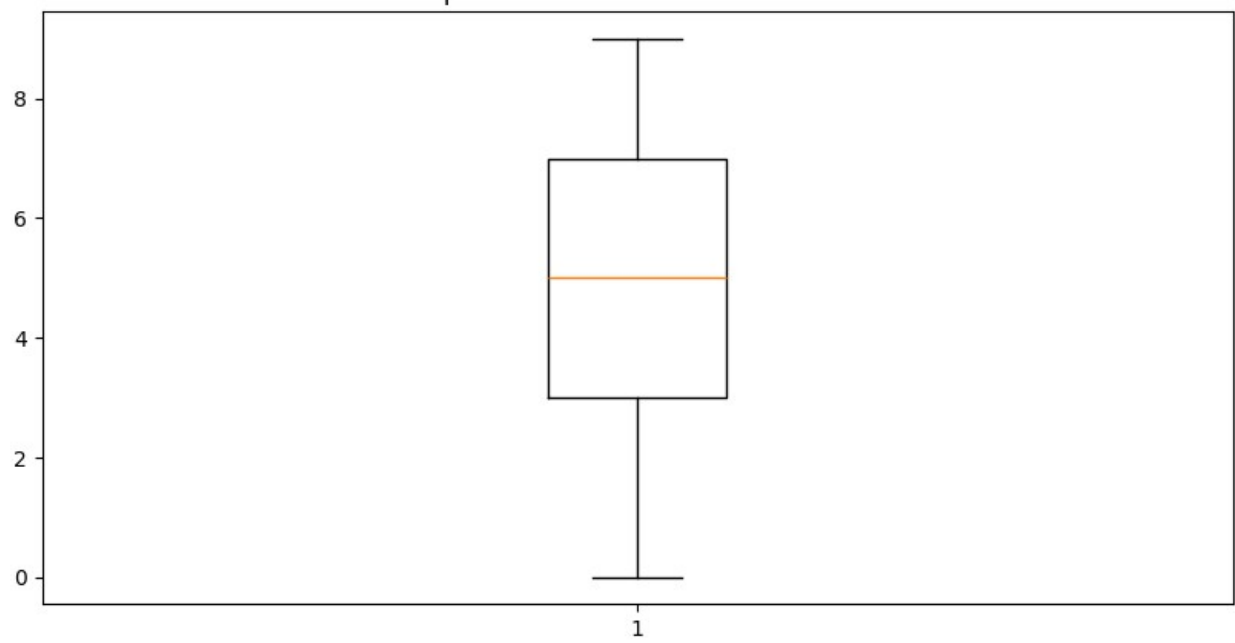
Boxplot of CAPE: TENANCY: OCCHU: % RENTER OCCUPIED



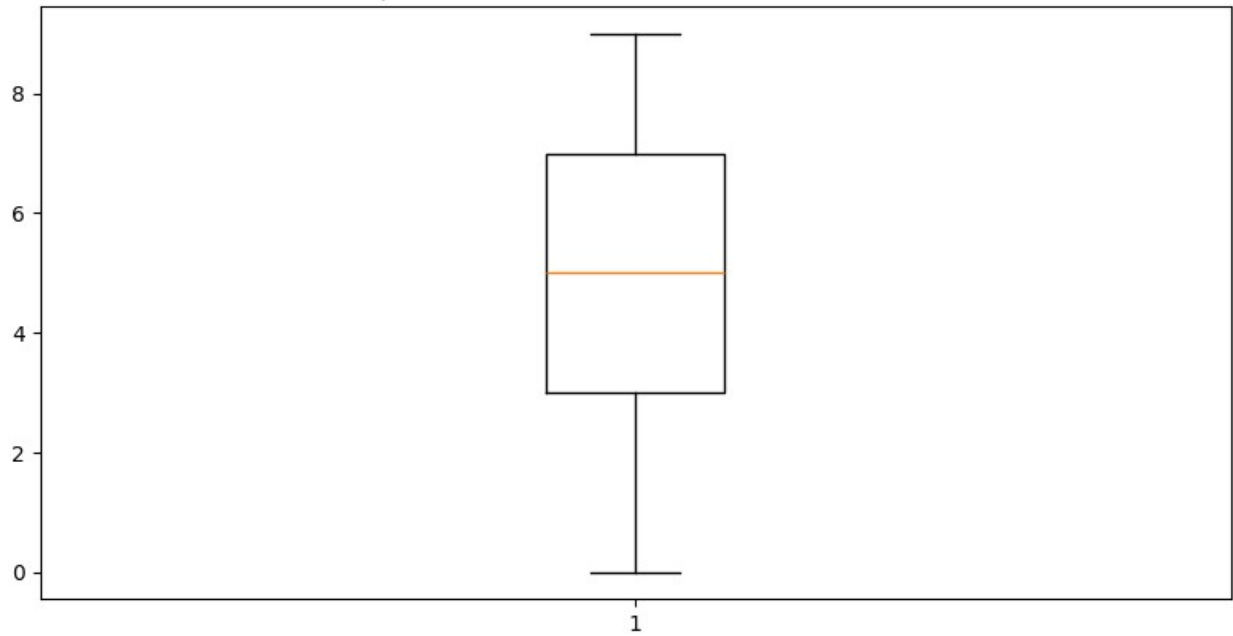
Boxplot of CAPE: EDUC: ISPSA



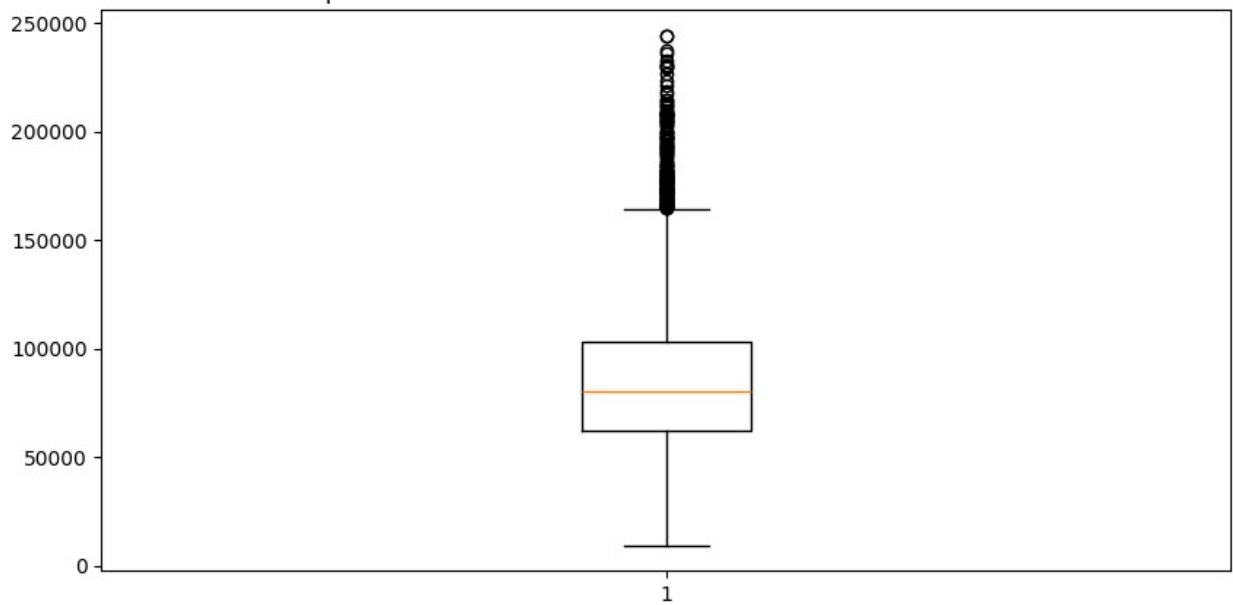
Boxplot of CAPE: EDUC: ISPSA DECILE



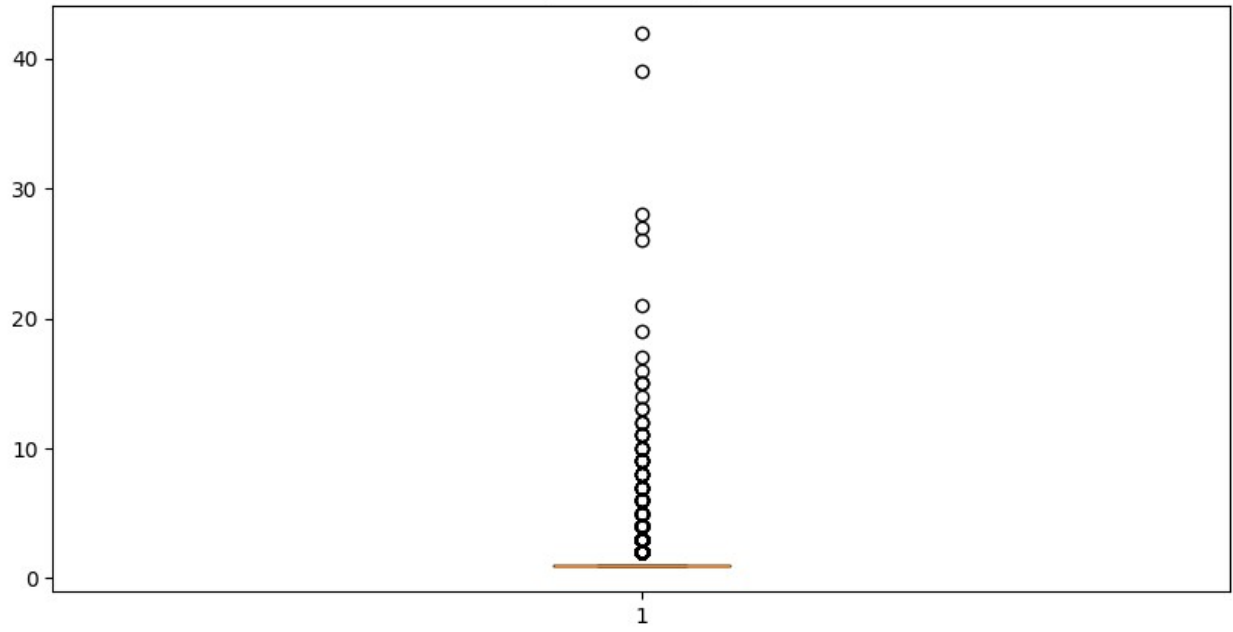
Boxplot of CAPE: INC: FAMILY INC STATE DECILE



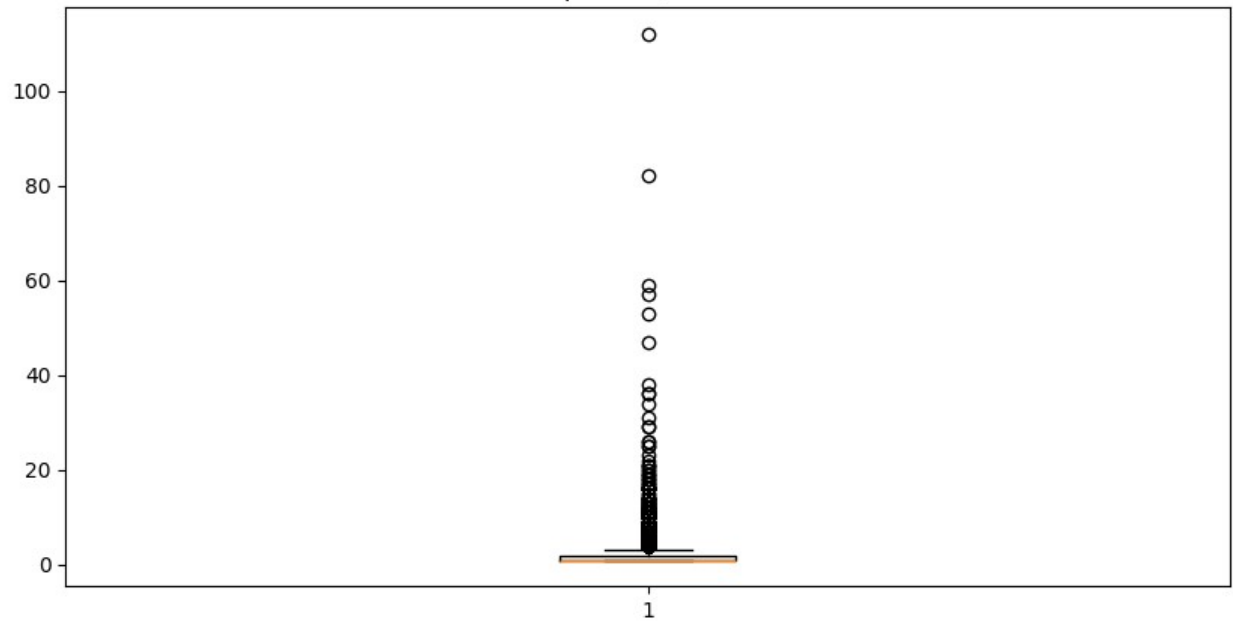
Boxplot of CAPE: INC: HH: MEDIAN FAMILY HOUSEHOLD INCOME

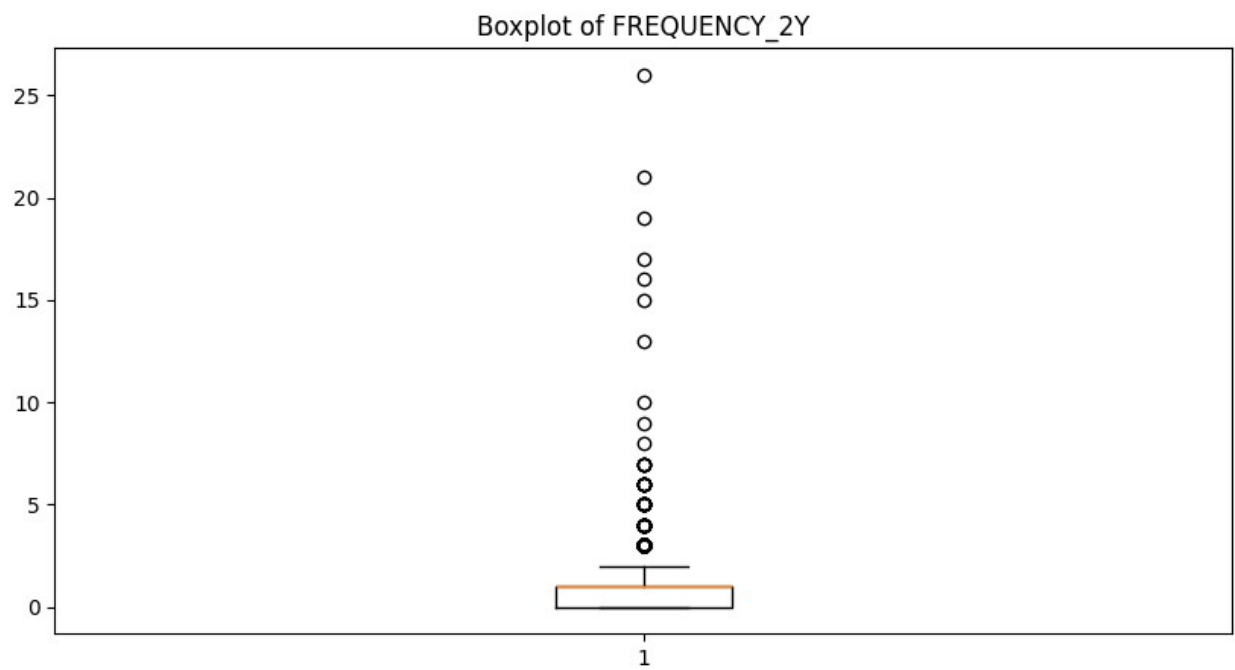
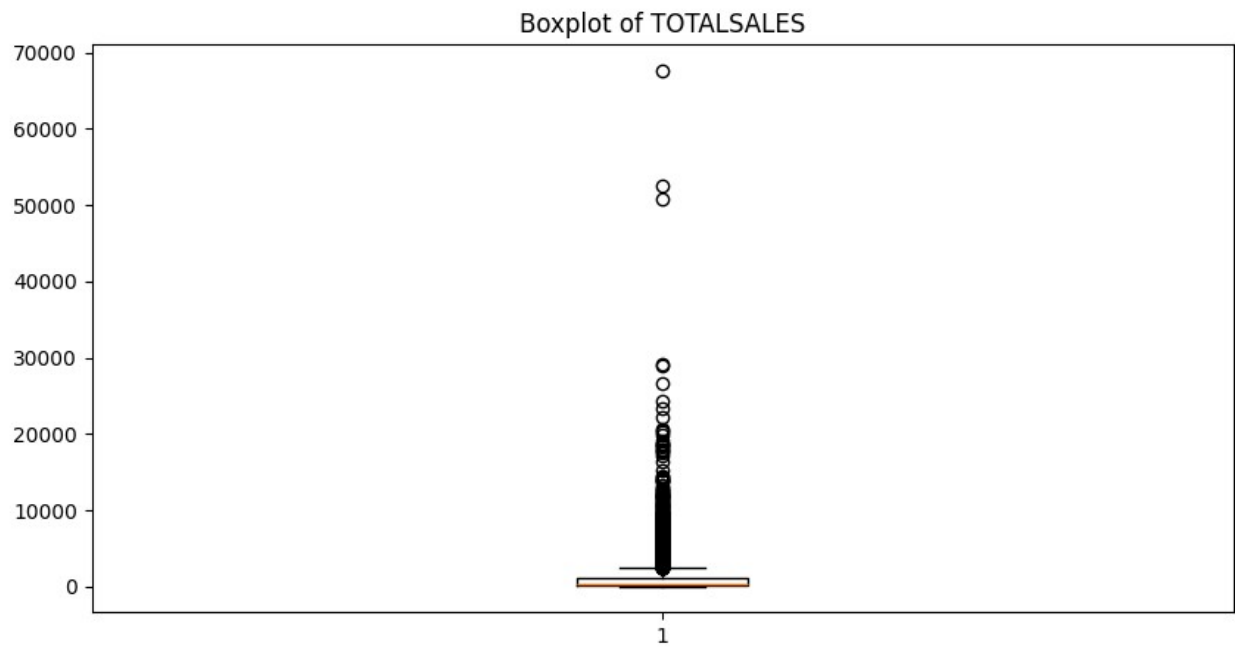


Boxplot of FREQUENCY

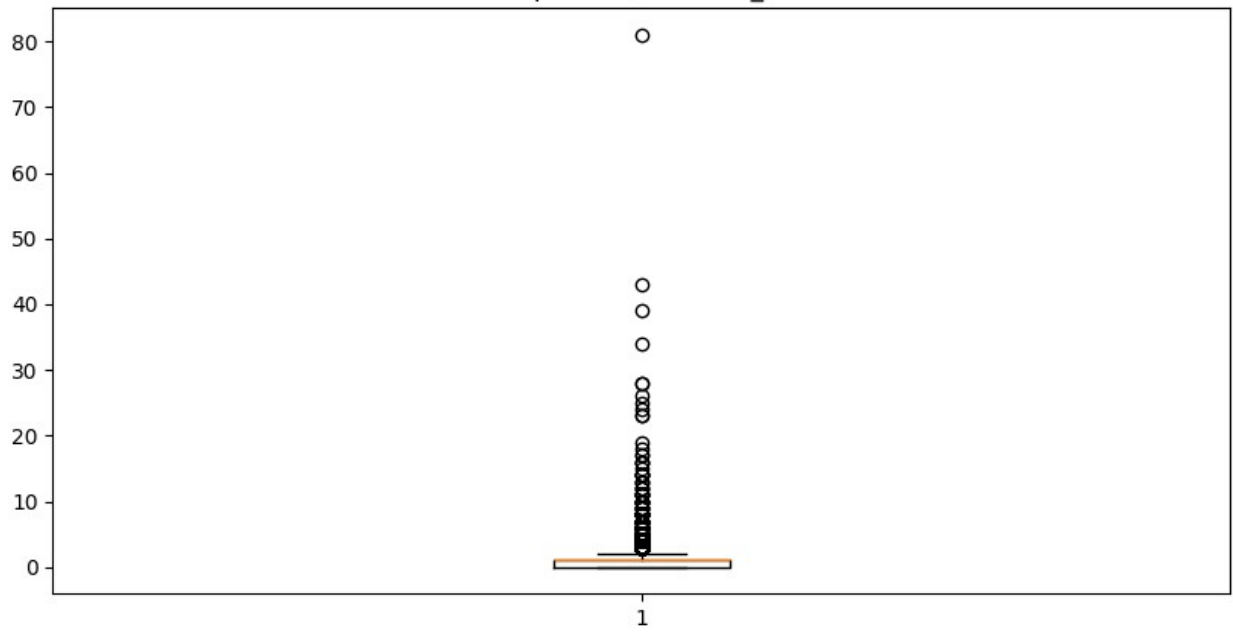


Boxplot of QUANTITY

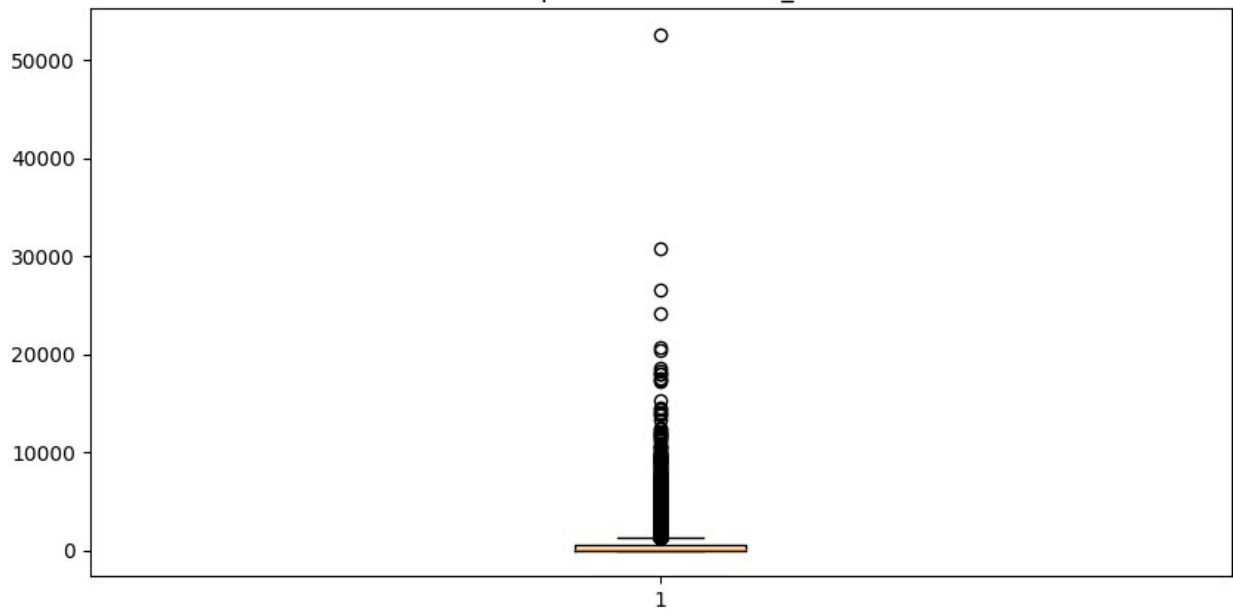




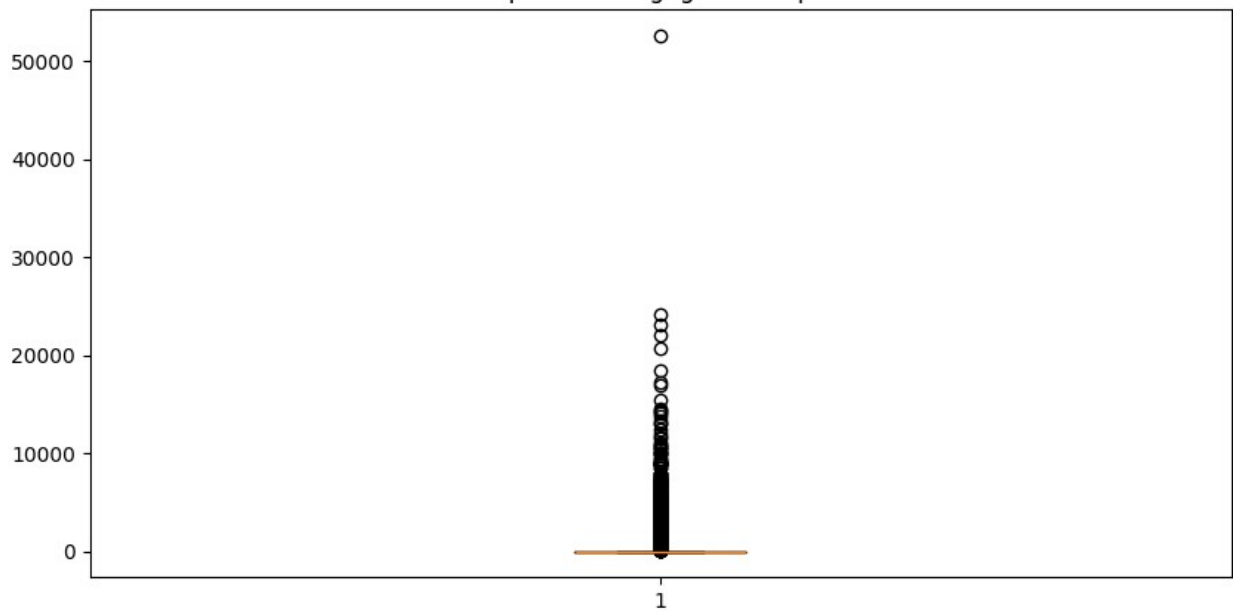
Boxplot of QUANTITY_2Y



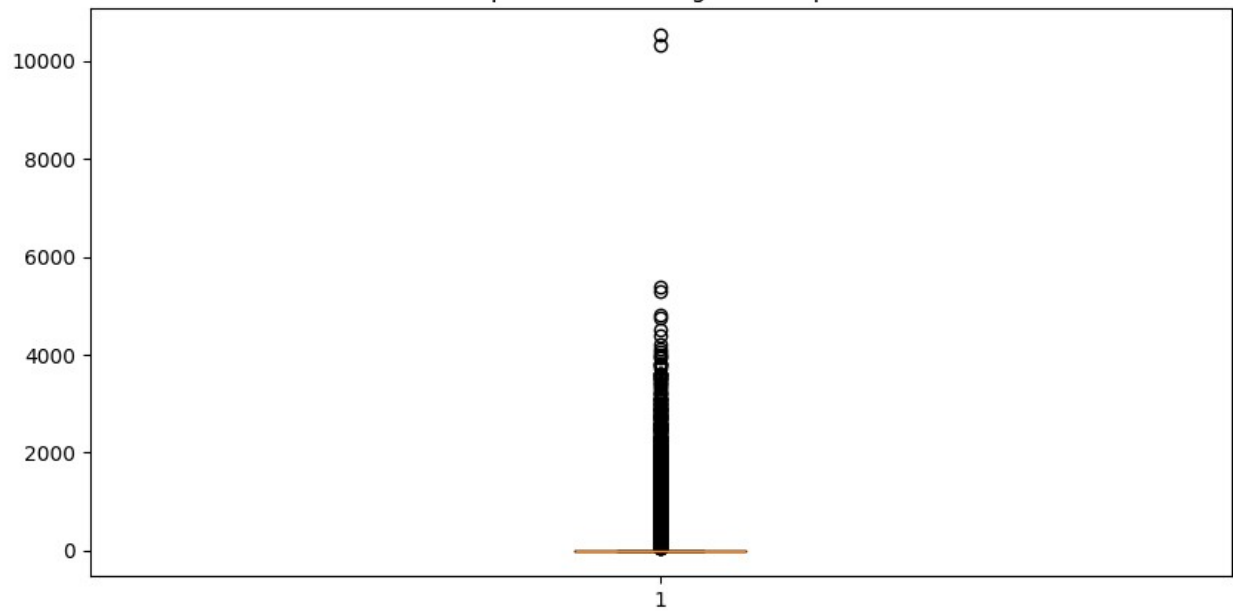
Boxplot of TOTALSALES_2Y



Boxplot of 1-Engagement Spend

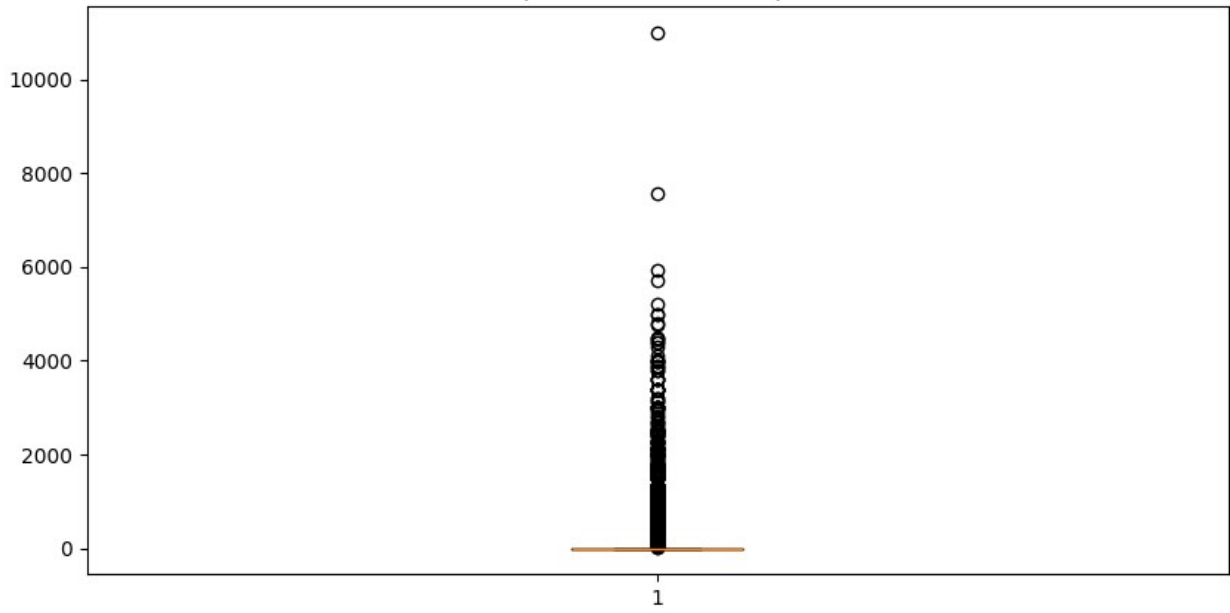


Boxplot of 2-Wedding Bands Spend

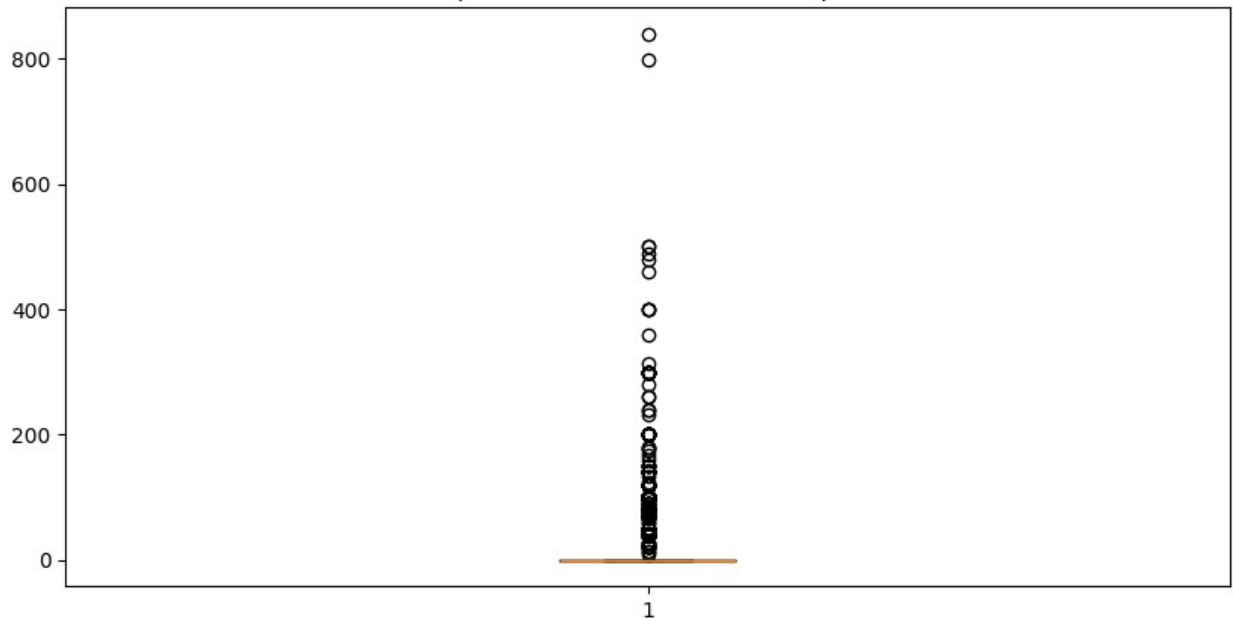


A box plot showing the distribution of the number of children per family. The x-axis is labeled '1' and the y-axis ranges from 0 to 25,000. The plot shows a single box at 0 with a long whisker extending to approximately 10,000 and many outliers up to 25,000.

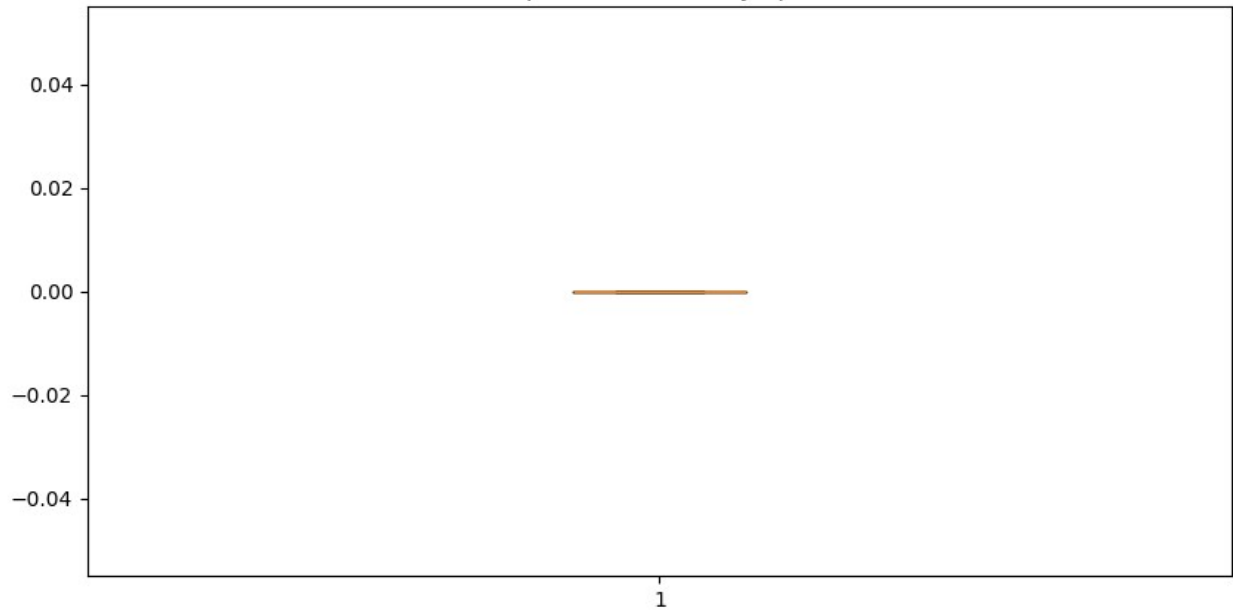
Boxplot of 5-Close Out Spend



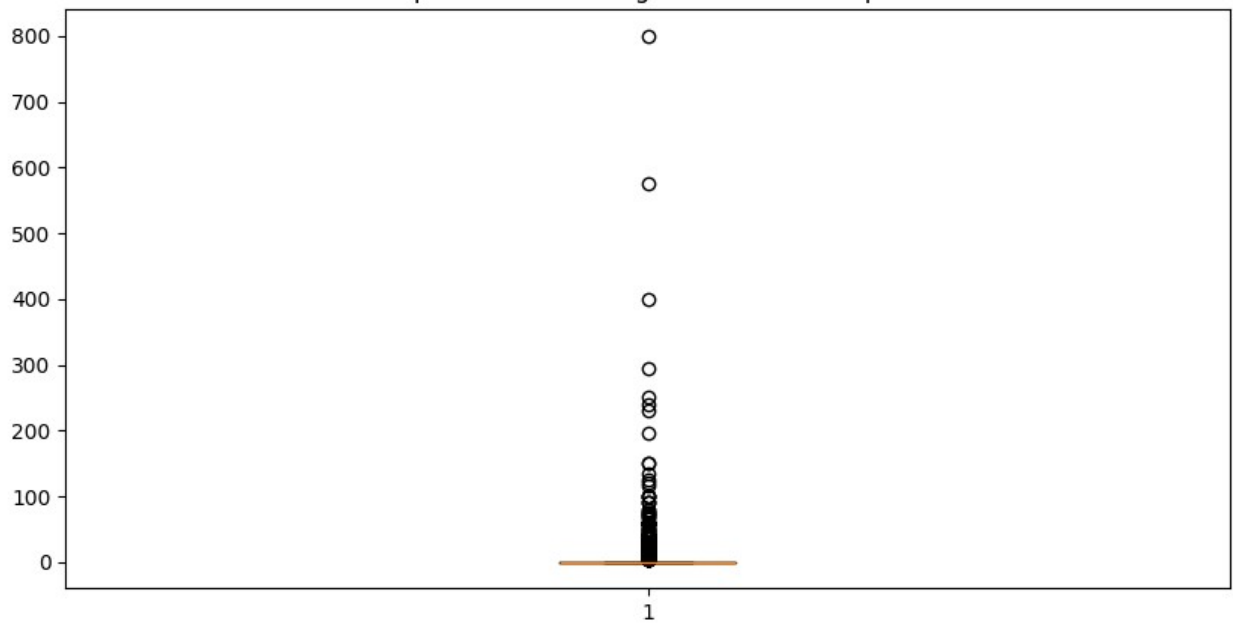
Boxplot of 6-Promotional Items Spend



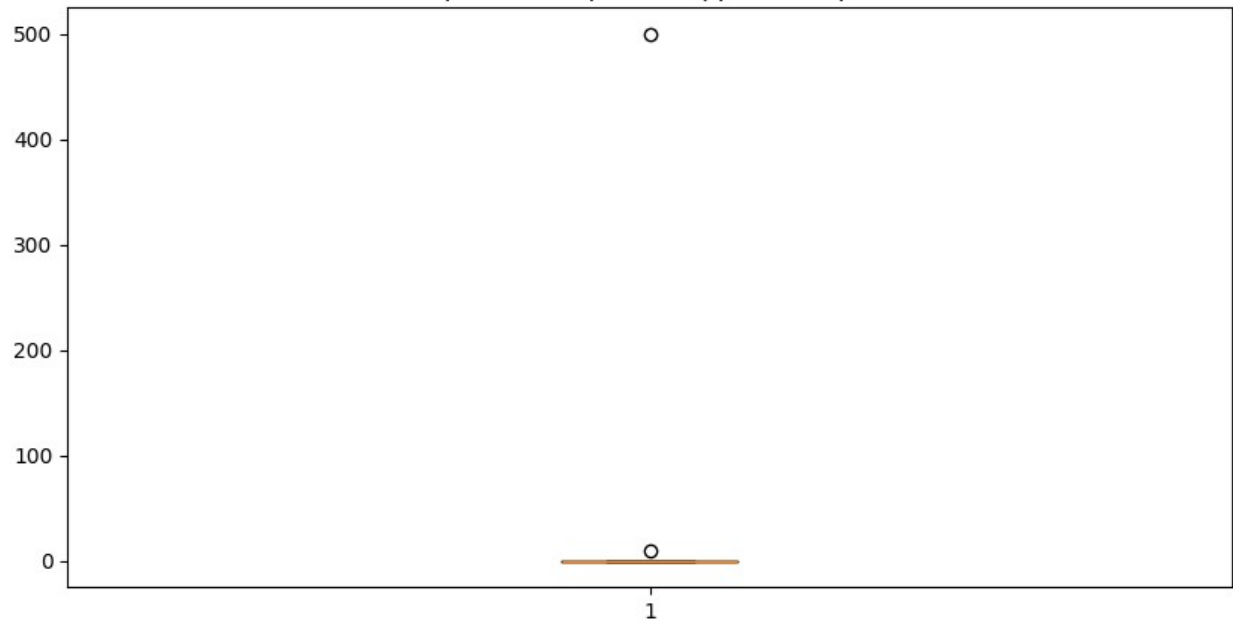
Boxplot of 7-Cost Only Spend



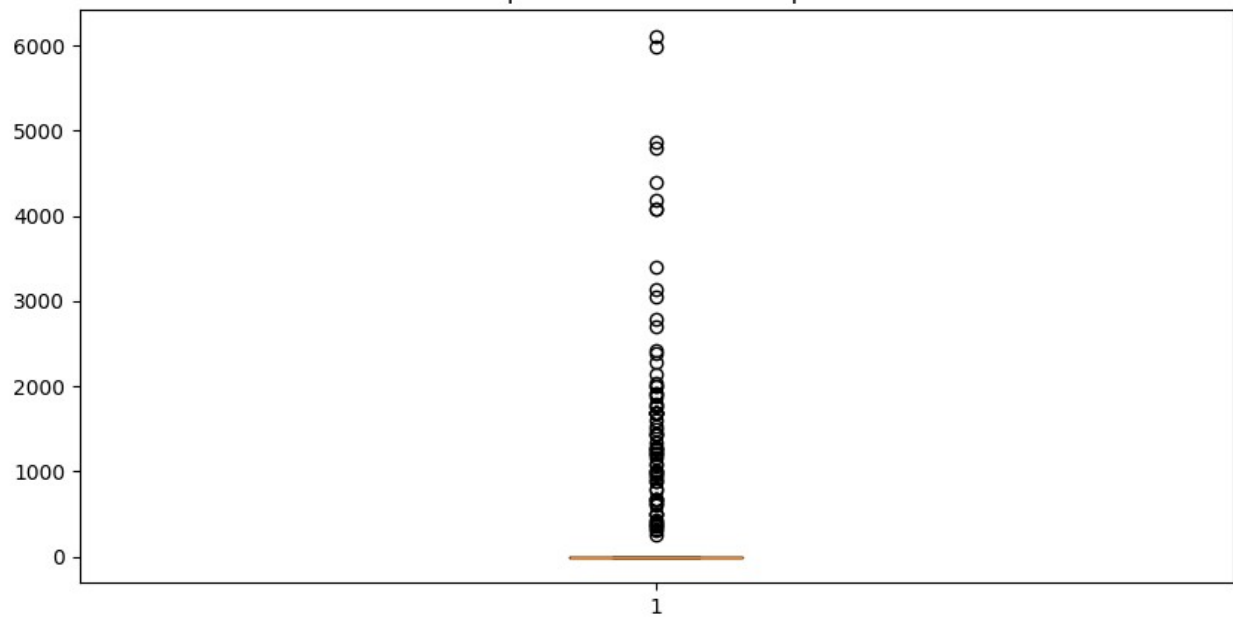
Boxplot of 8-Marketing Premium SKUs Spend



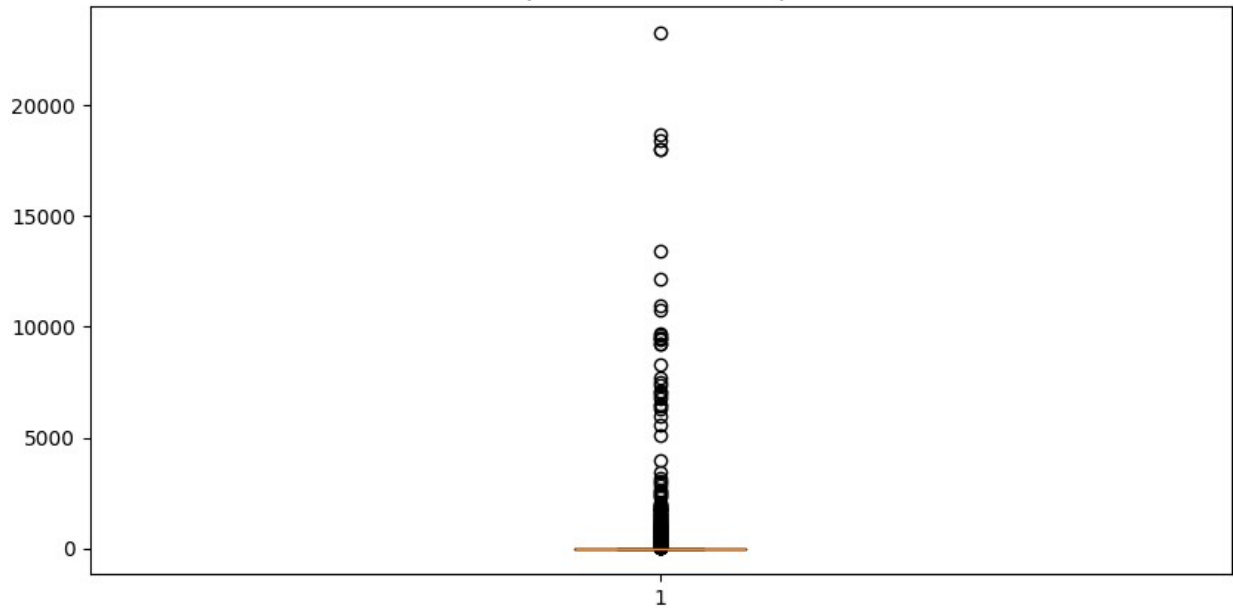
Boxplot of 9-Repairs & Appraisals Spend



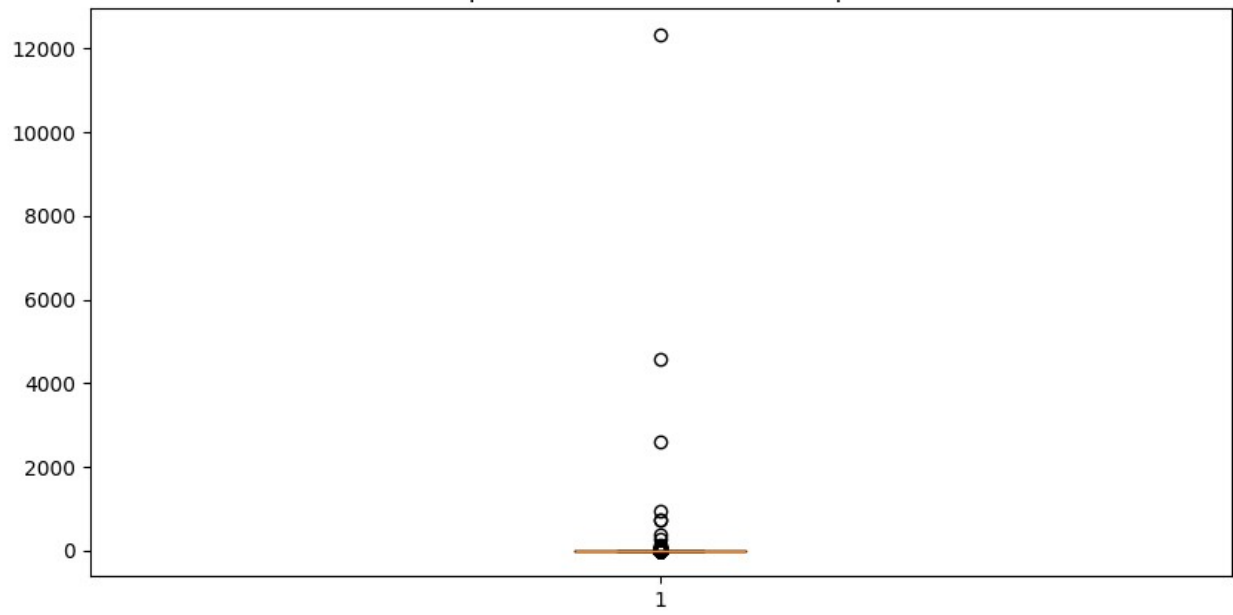
Boxplot of 10-Pre Owned Spend



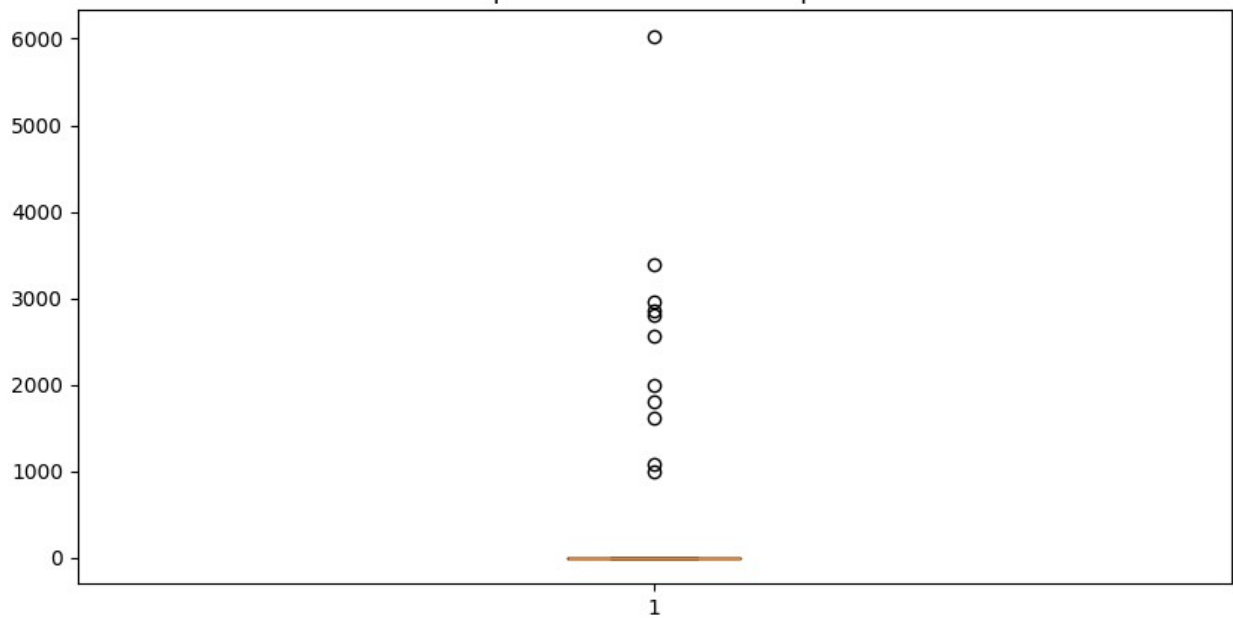
Boxplot of 11-Watches Spend



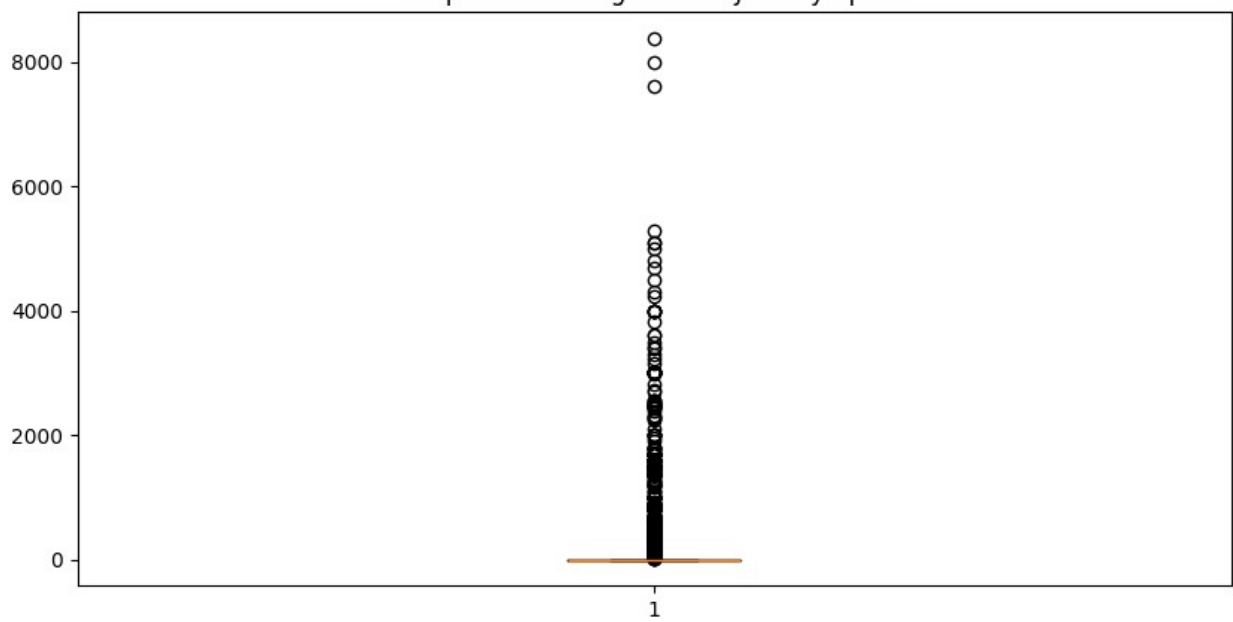
Boxplot of 12-Misc Merchandise Spend



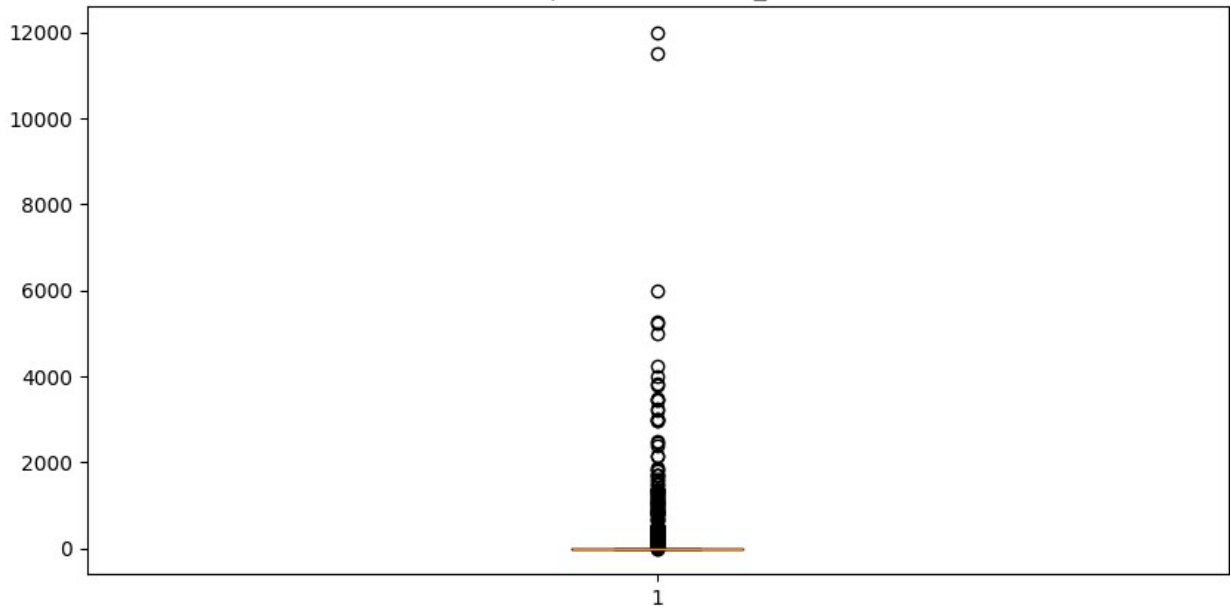
Boxplot of 15-Store Events Spend



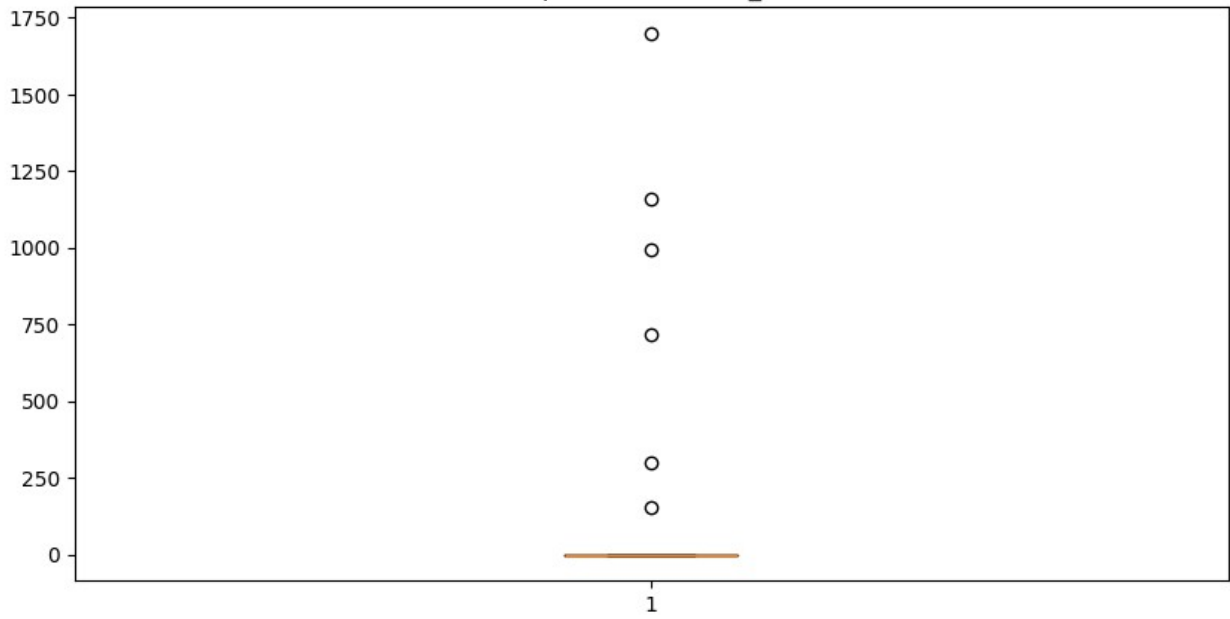
Boxplot of 16-Single Stone Jewelry Spend



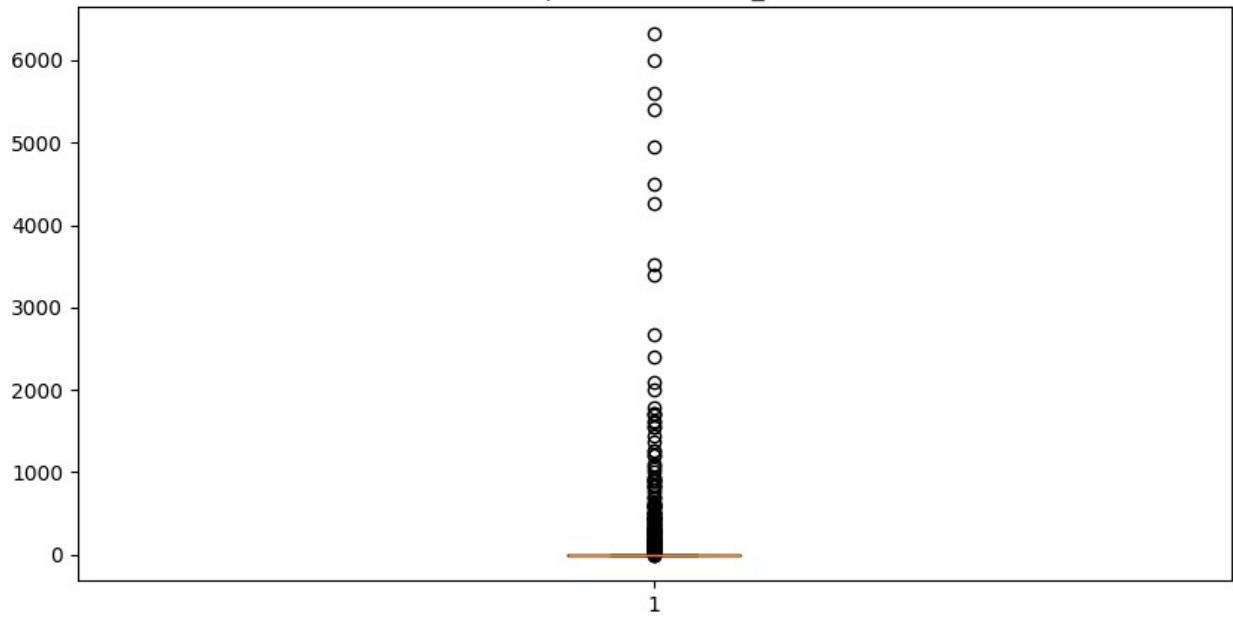
Boxplot of MDAYREV_L1Y



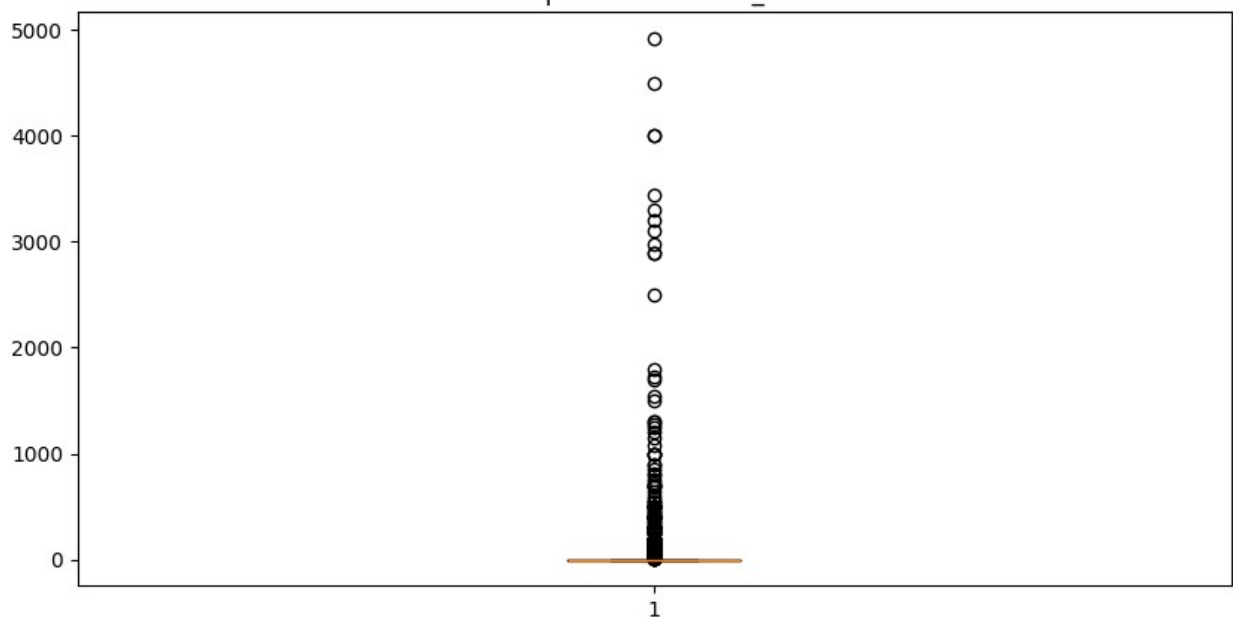
Boxplot of MDAYREV_L2Y



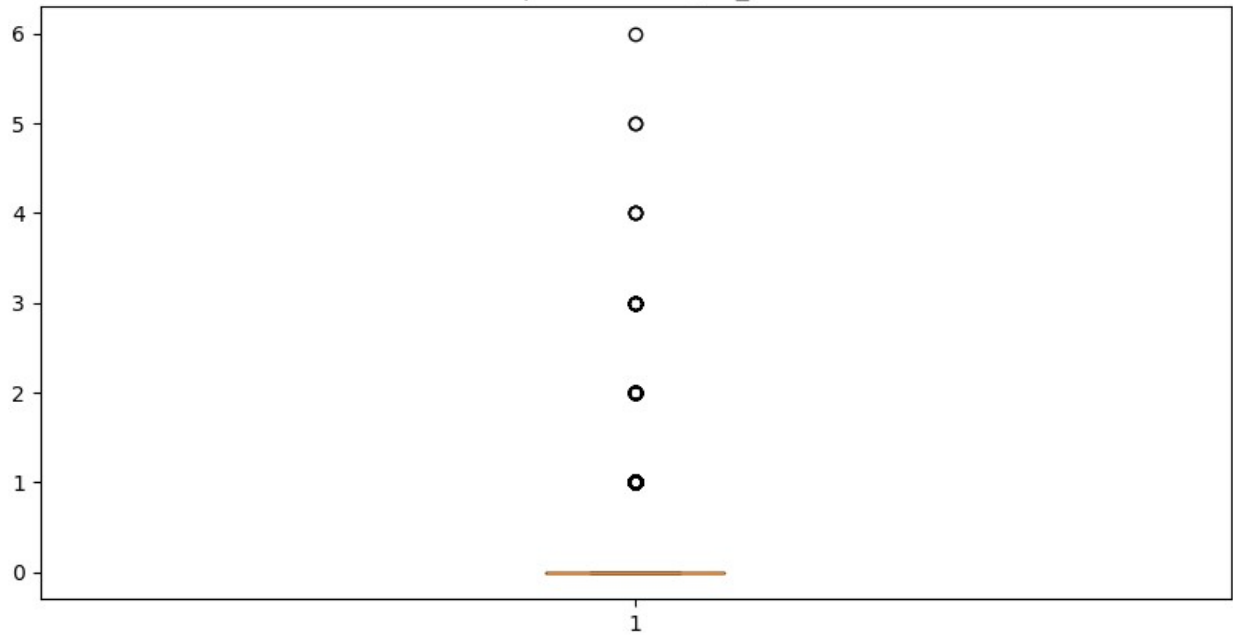
Boxplot of MDAYREV_L3Y



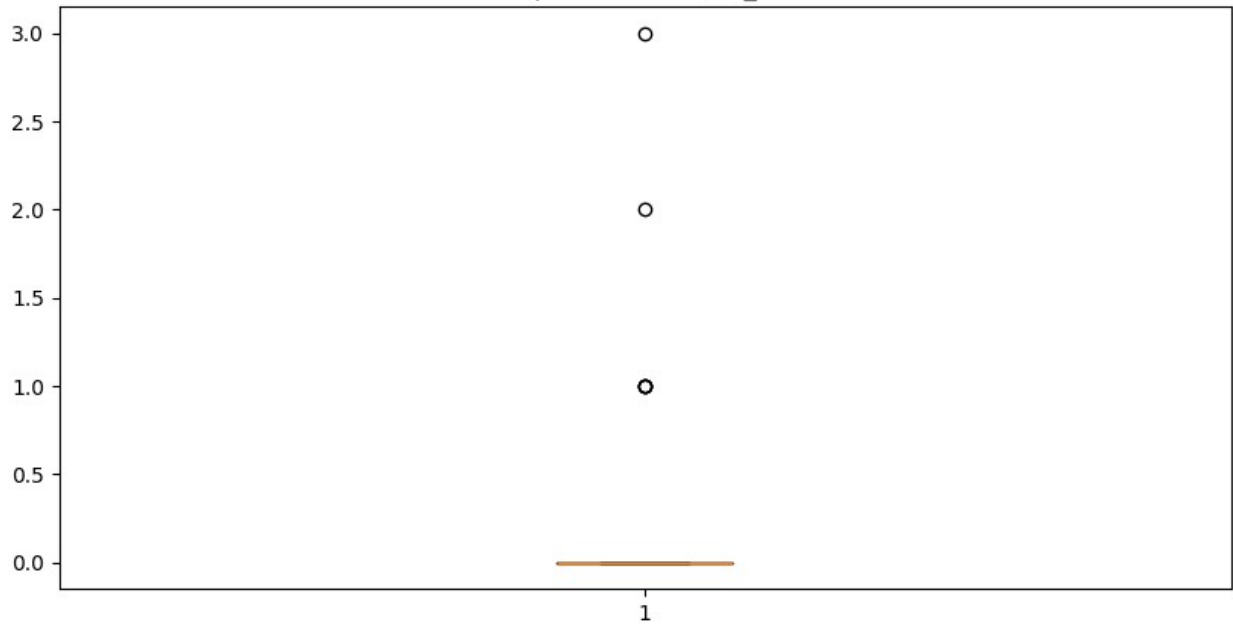
Boxplot of MDAYREV_L4Y



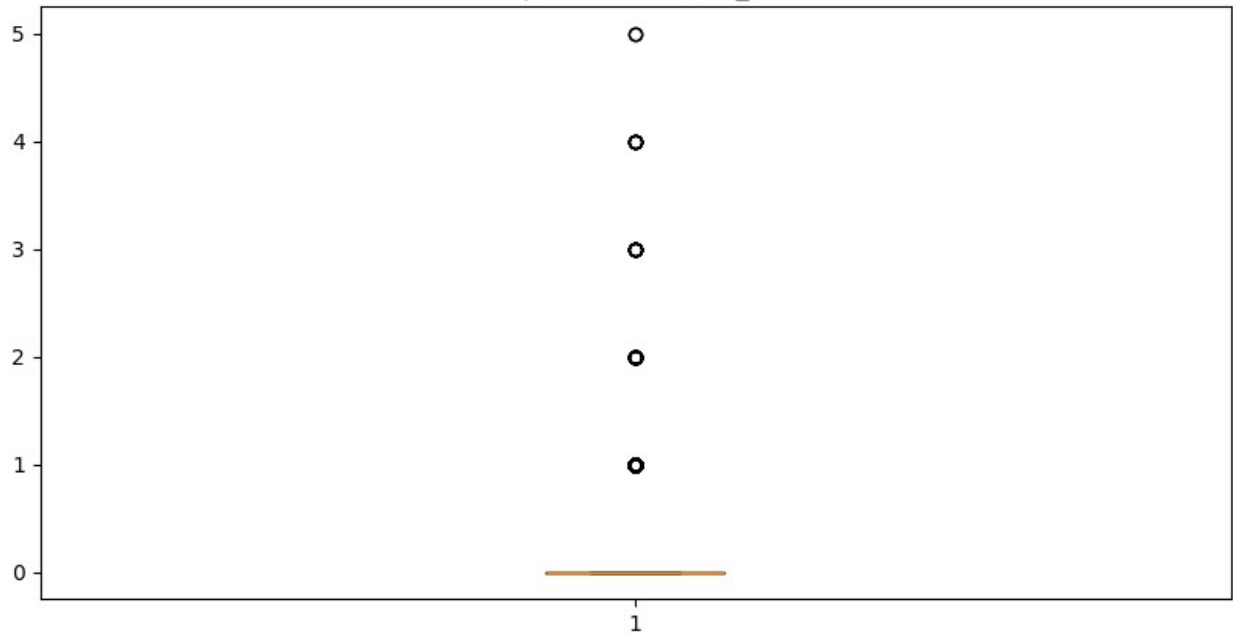
Boxplot of MDAYQTY_L1Y



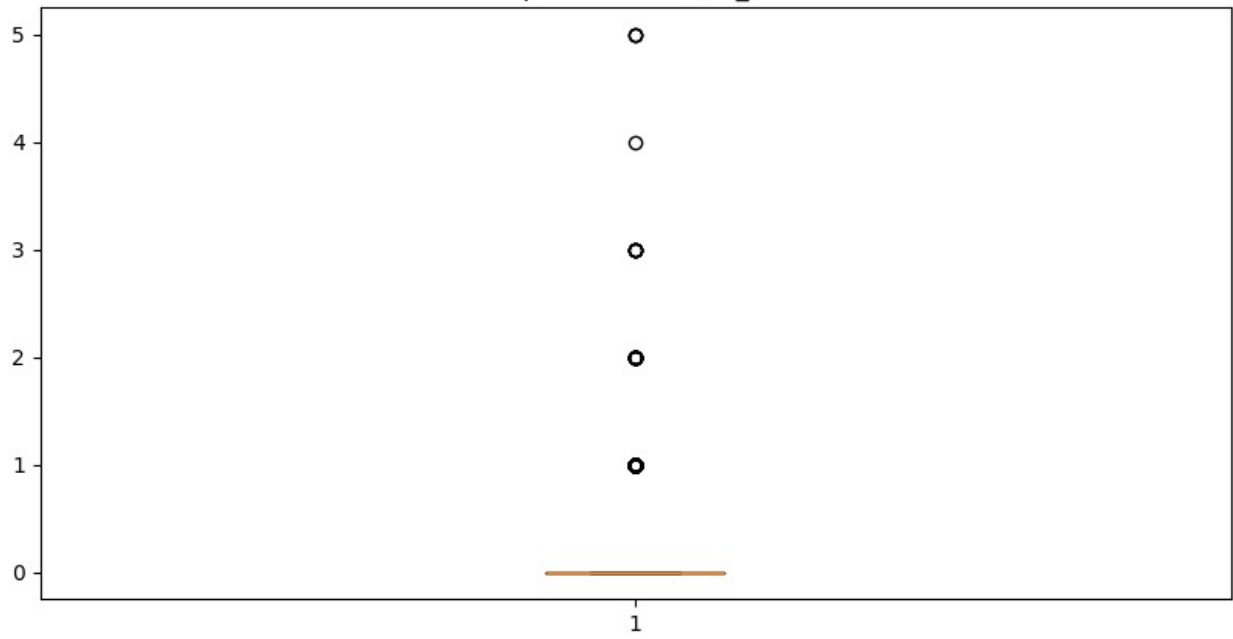
Boxplot of MDAYQTY_L2Y



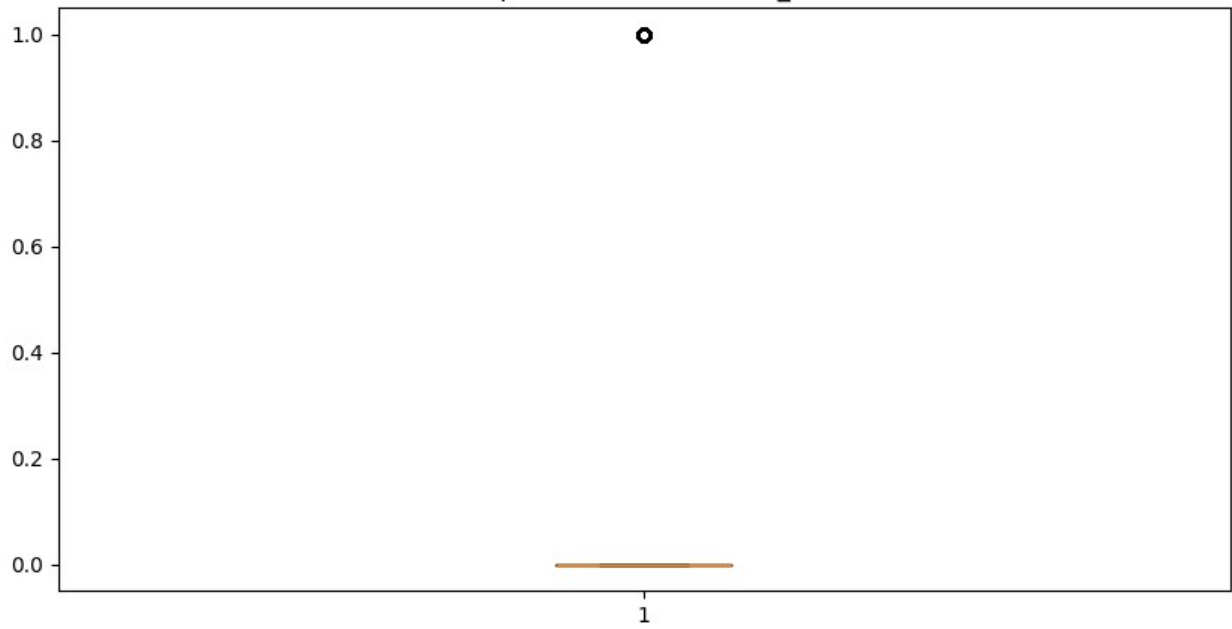
Boxplot of MDAYQTY_L3Y



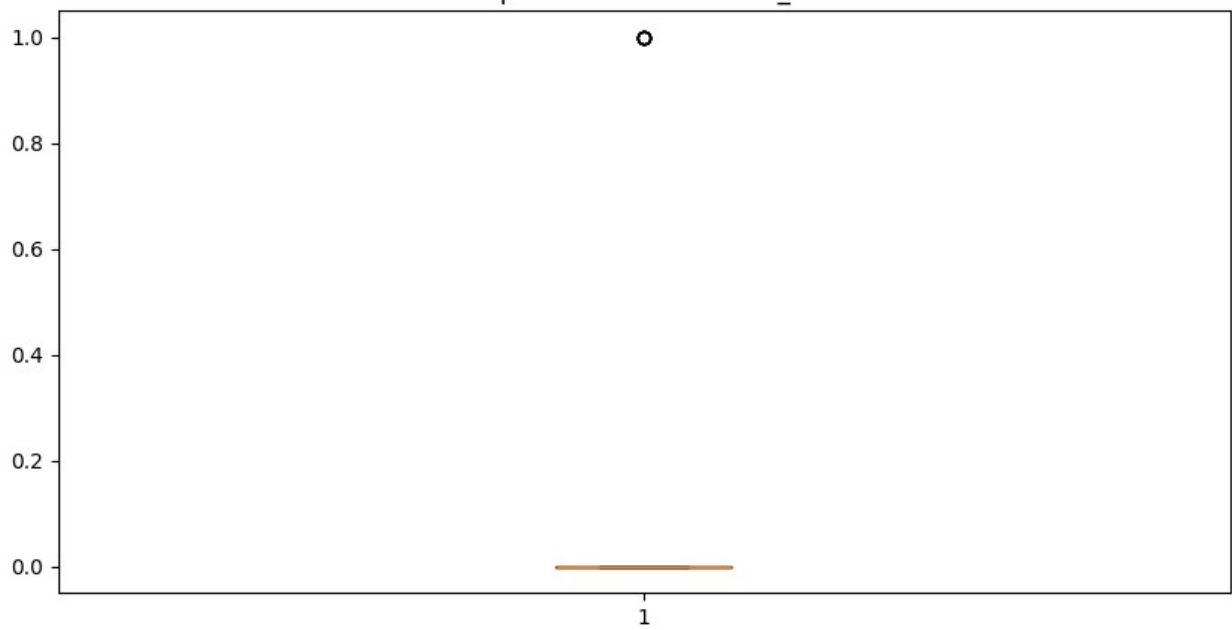
Boxplot of MDAYQTY_L4Y

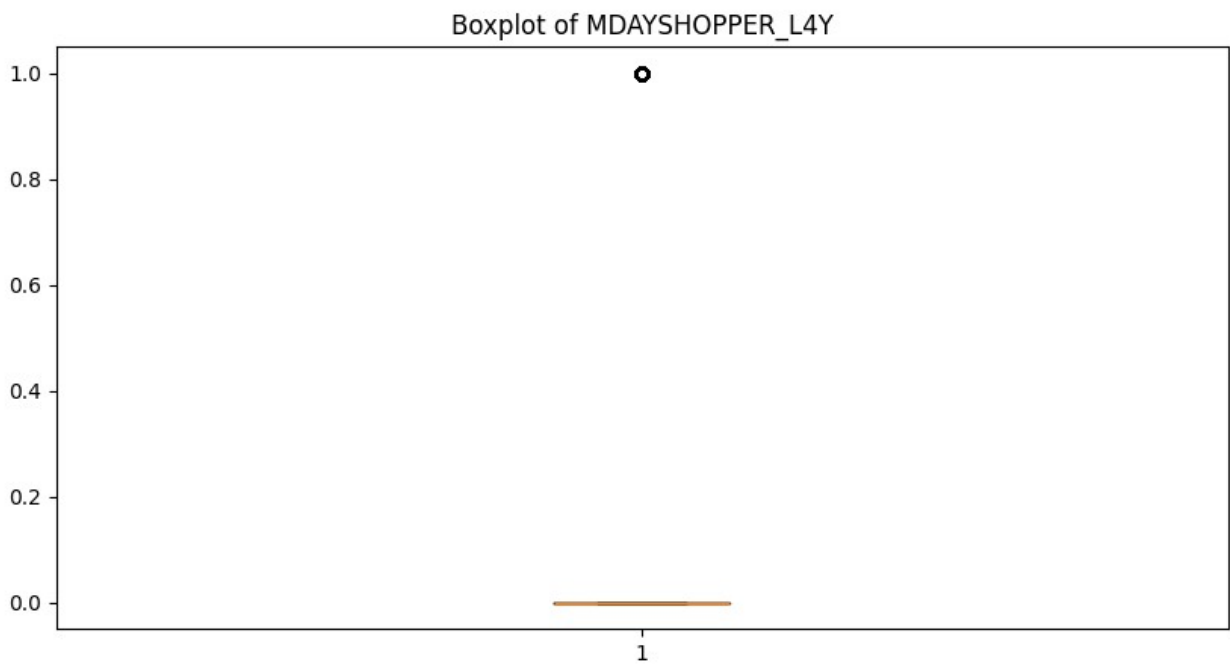
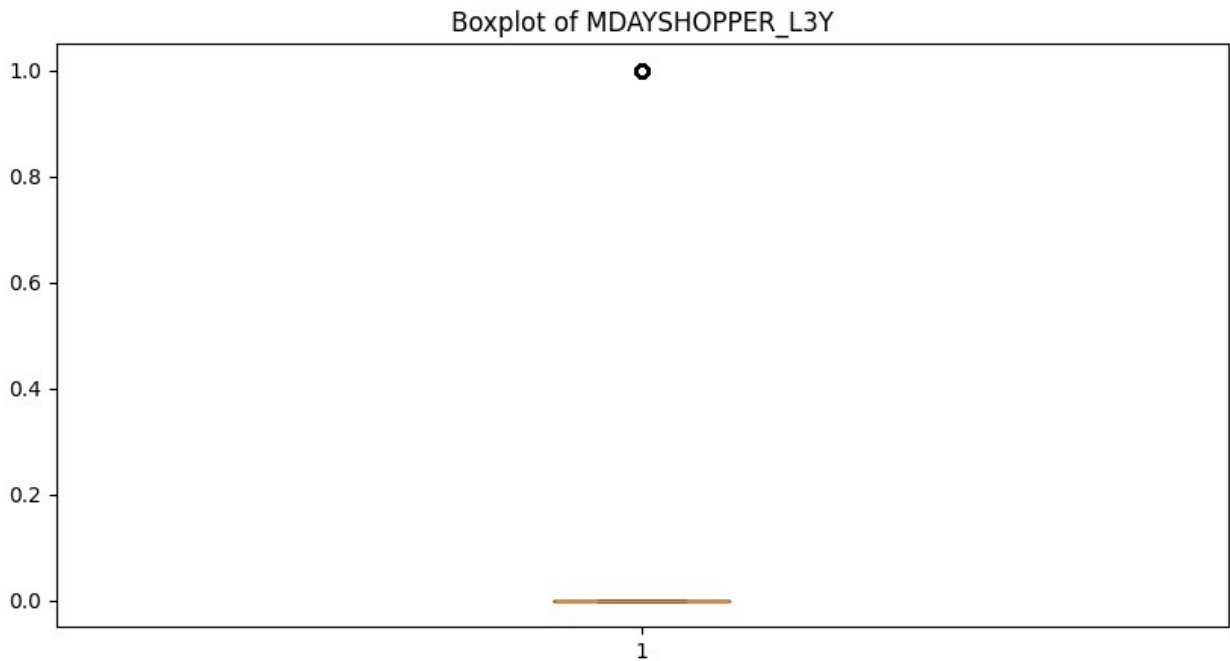


Boxplot of MDAYSHOPPER_L1Y



Boxplot of MDAYSHOPPER_L2Y





Plot distribution for 6 columns

```
columns_to_plot = numeric_columns[:6]
for col in columns_to_plot:
    plt.figure(figsize=(10, 5))
    data_cleaned[col].dropna().hist(bins=20)
    plt.title(f"Distribution of {col}")
    plt.xlabel(col)
```



```
plt.ylabel("Frequency")  
plt.show()
```

