Data Collection and Preprocessing Phase

Date	19 April 2024
Team ID	738220
Project Title	Walmart Sales Analysis for Retail Industry with Machine Learning
Maximum Marks	6 Marks

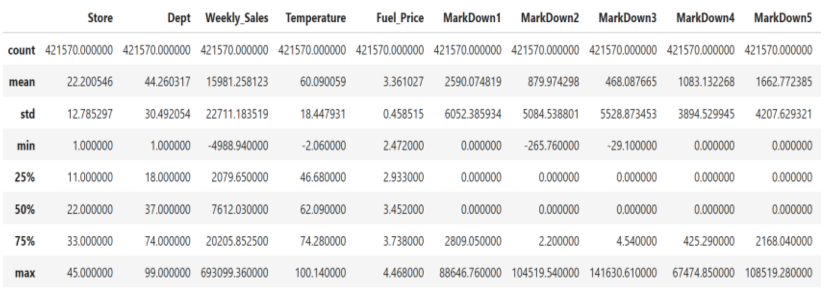
Data Exploration and Preprocessing

Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

<table>
<tr><th>Section</th><th>Description</th></tr>
<tr><td>Data Overview</td><td>

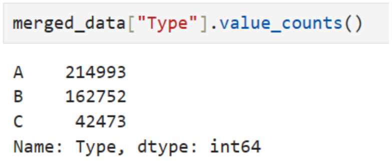
<u>Dimension :</u>
421570 rows × 17 columns

<u>Descriptive Statistics:</u>



<table>
<tr><th></th><th>Store</th><th>Dept</th><th>Weekly_Sales</th><th>Temperature</th><th>Fuel_Price</th><th>MarkDown1</th><th>MarkDown2</th><th>MarkDown3</th><th>MarkDown4</th><th>MarkDown5</th></tr>
<tr><td>count</td><td>421570.000000</td><td>421570.000000</td><td>421570.000000</td><td>421570.000000</td><td>421570.000000</td><td>421570.000000</td><td>421570.000000</td><td>421570.000000</td><td>421570.000000</td><td>421570.000000</td></tr>
<tr><td>mean</td><td>22.200546</td><td>44.260317</td><td>15981.258123</td><td>60.090059</td><td>3.361027</td><td>2590.074819</td><td>879.974298</td><td>468.087665</td><td>1083.132268</td><td>1662.772385</td></tr>
<tr><td>std</td><td>12.785297</td><td>30.492054</td><td>22711.183519</td><td>18.447931</td><td>0.458515</td><td>6052.385934</td><td>5084.538801</td><td>5528.873453</td><td>3894.529945</td><td>4207.629321</td></tr>
<tr><td>min</td><td>1.000000</td><td>1.000000</td><td>-4988.940000</td><td>-2.060000</td><td>2.472000</td><td>0.000000</td><td>-265.760000</td><td>-29.100000</td><td>0.000000</td><td>0.000000</td></tr>
<tr><td>25%</td><td>11.000000</td><td>18.000000</td><td>2079.650000</td><td>46.680000</td><td>2.933000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td></tr>
<tr><td>50%</td><td>22.000000</td><td>37.000000</td><td>7612.030000</td><td>62.090000</td><td>3.452000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td><td>0.000000</td></tr>
<tr><td>75%</td><td>33.000000</td><td>74.000000</td><td>20205.852500</td><td>74.280000</td><td>3.738000</td><td>2809.050000</td><td>2.200000</td><td>4.540000</td><td>425.290000</td><td>2168.040000</td></tr>
<tr><td>max</td><td>45.000000</td><td>99.000000</td><td>693099.360000</td><td>100.140000</td><td>4.468000</td><td>88646.760000</td><td>104519.540000</td><td>141630.610000</td><td>67474.850000</td><td>108519.280000</td></tr>
</table>

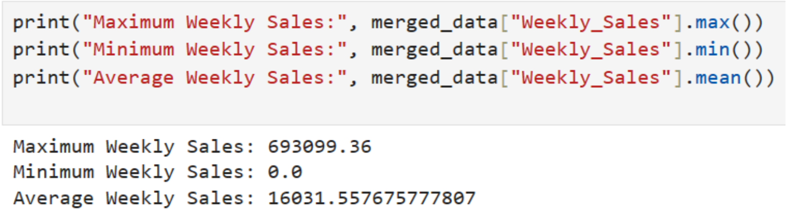
</td></tr>
<tr><td>Univariate Analysis</td><td>



```
merged_data["Type"].value_counts()
```

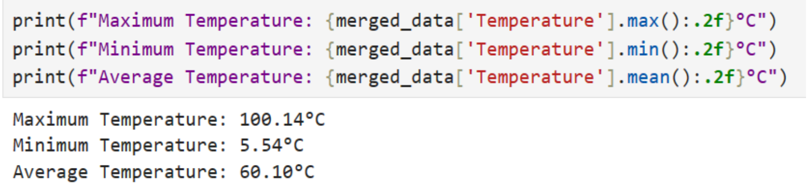
```
A    214993
B    162752
C     42473
Name: Type, dtype: int64
```

</td></tr>
</table>



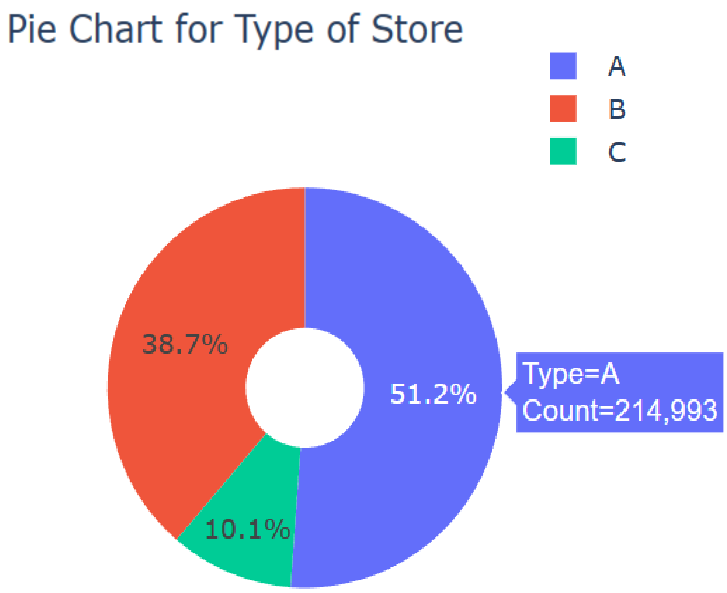
```
print("Maximum Weekly Sales:", merged_data["Weekly_Sales"].max())
print("Minimum Weekly Sales:", merged_data["Weekly_Sales"].min())
print("Average Weekly Sales:", merged_data["Weekly_Sales"].mean())
```

```
Maximum Weekly Sales: 693099.36
Minimum Weekly Sales: 0.0
Average Weekly Sales: 16031.557675777807
```

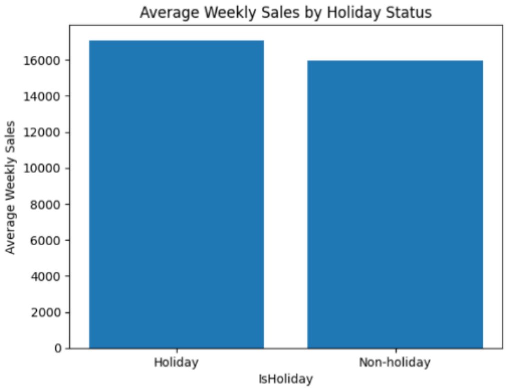


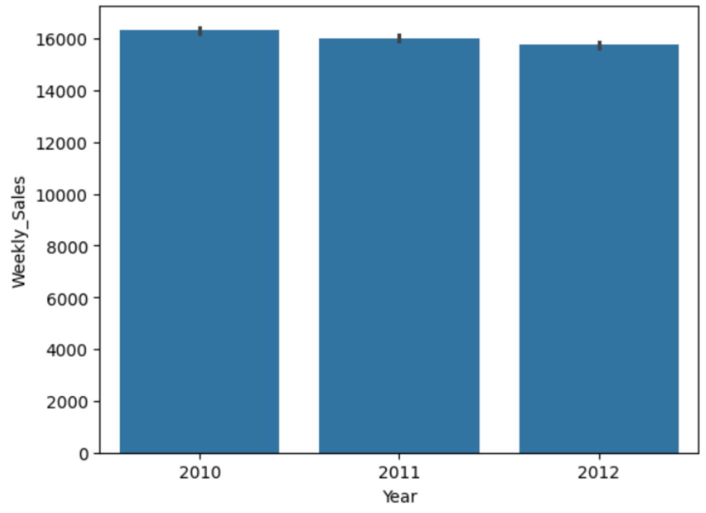
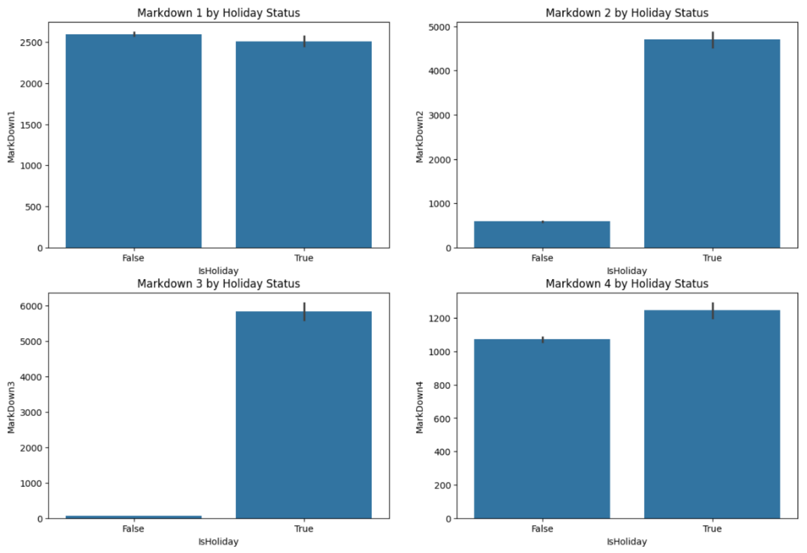
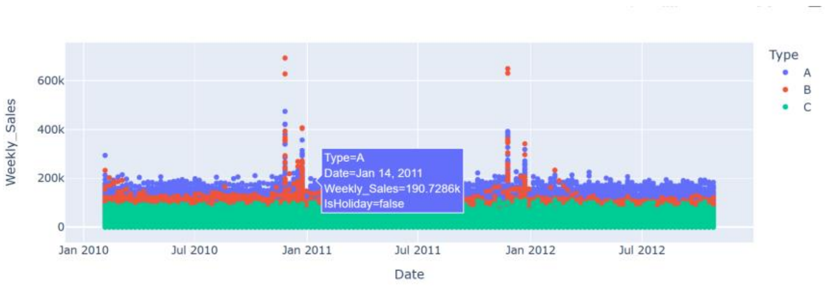
```
print(f"Maximum Temperature: {merged_data['Temperature'].max():.2f}°C")
print(f"Minimum Temperature: {merged_data['Temperature'].min():.2f}°C")
print(f"Average Temperature: {merged_data['Temperature'].mean():.2f}°C")
```

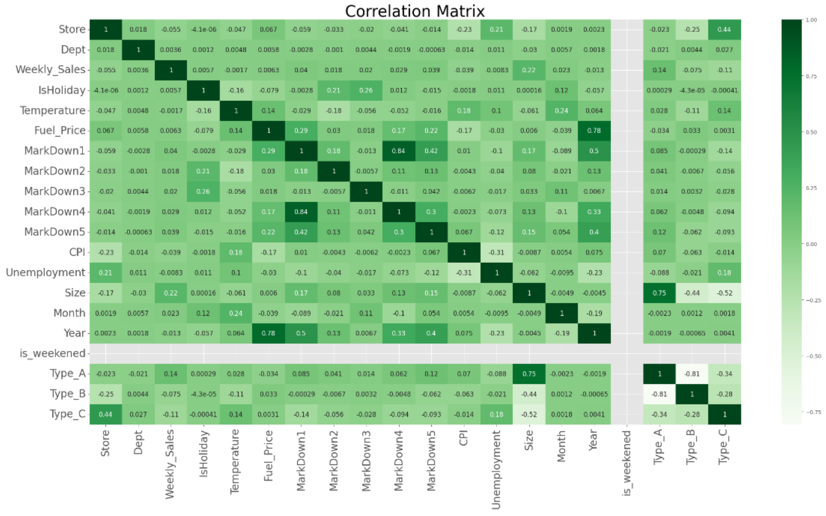
```
Maximum Temperature: 100.14°C
Minimum Temperature: 5.54°C
Average Temperature: 60.10°C
```



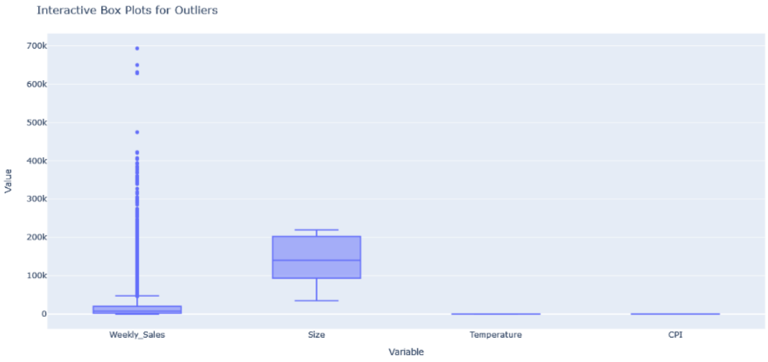
Bivariate Analysis



	 
Multivariate Analysis	

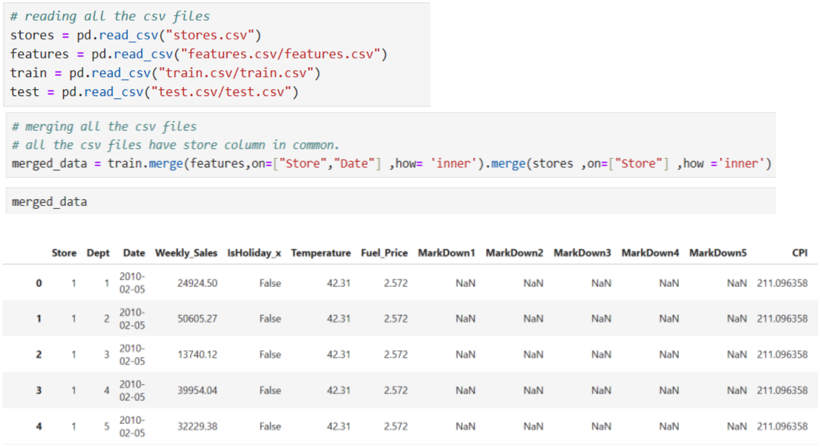


Outliers and Anomalies



Data Preprocessing Code Screenshots

Loading Data

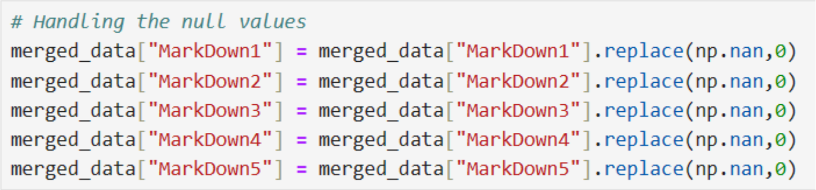


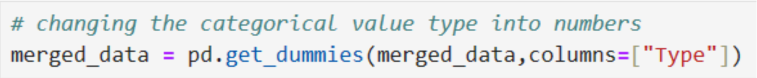
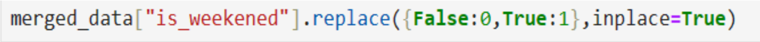
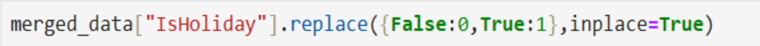
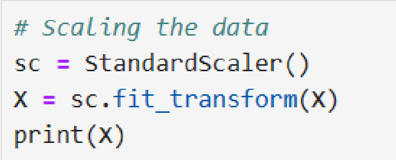
```
# reading all the csv files
stores = pd.read_csv("stores.csv")
features = pd.read_csv("features.csv/features.csv")
train = pd.read_csv("train.csv/train.csv")
test = pd.read_csv("test.csv/test.csv")
```

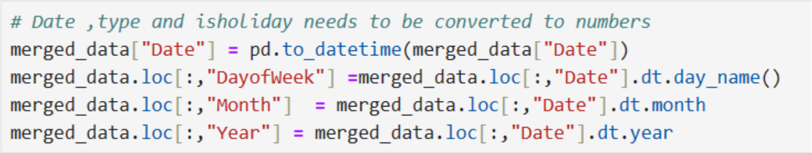
```
# merging all the csv files
# all the csv files have store column in common.
merged_data = train.merge(features,on=["Store","Date"] ,how= 'inner').merge(stores ,on=["Store"] ,how ='inner')
```

```
merged_data
```

	Store	Dept	Date	Weekly_Sales	IsHoliday_x	Temperature	Fuel_Price	MarkDown1	MarkDown2	MarkDown3	MarkDown4	MarkDown5	CPI
0	1	1	2010-02-05	24924.50	False	42.31	2.572	NaN	NaN	NaN	NaN	NaN	211.096358
1	1	2	2010-02-05	50605.27	False	42.31	2.572	NaN	NaN	NaN	NaN	NaN	211.096358
2	1	3	2010-02-05	13740.12	False	42.31	2.572	NaN	NaN	NaN	NaN	NaN	211.096358
3	1	4	2010-02-05	39954.04	False	42.31	2.572	NaN	NaN	NaN	NaN	NaN	211.096358
4	1	5	2010-02-05	32229.38	False	42.31	2.572	NaN	NaN	NaN	NaN	NaN	211.096358

Handling Missing Data	
	`# Handling the null values`
	`merged_data["MarkDown1"] = merged_data["MarkDown1"].replace(np.nan,0)`
	`merged_data["MarkDown2"] = merged_data["MarkDown2"].replace(np.nan,0)`
	`merged_data["MarkDown3"] = merged_data["MarkDown3"].replace(np.nan,0)`
	`merged_data["MarkDown4"] = merged_data["MarkDown4"].replace(np.nan,0)`
	`merged_data["MarkDown5"] = merged_data["MarkDown5"].replace(np.nan,0)`

Data Transformation	
	`# changing the categorical value type into numbers`
	`merged_data = pd.get_dummies(merged_data,columns=["Type"])`
	
	`merged_data["is_weekened"].replace({False:0,True:1},inplace=True)`
	
	`merged_data["IsHoliday"].replace({False:0,True:1},inplace=True)`
	
	`# Scaling the data`
	`sc = StandardScaler()`
	`X = sc.fit_transform(X)`
	`print(X)`

Feature Engineering	
	`# Date ,type and isholiday needs to be converted to numbers`
	`merged_data["Date"] = pd.to_datetime(merged_data["Date"])`
	`merged_data.loc[:,"DayofWeek"] =merged_data.loc[:,"Date"].dt.day_name()`
	`merged_data.loc[:,"Month"]  = merged_data.loc[:,"Date"].dt.month`
	`merged_data.loc[:,"Year"] = merged_data.loc[:,"Date"].dt.year`

Save Processed Data	-