

```

/* Import the Dataset */

FILENAME REFFILE '/folders/myfolders/Walmart_Store_sales.csv';

PROC IMPORT DATAFILE=REFFILE
    DBMS=CSV
    OUT=WORK.walmart replace;
    GETNAMES=YES;
RUN;

proc print data=walmart;
run;

/* Check the content of the data */

proc contents data= work.walmart;
run;

/* Check the missing value */

proc means data=work.walmart nmiss;
run;

/* Which store has maximum sales */

proc means data=work.walmart max;
    by store;
    var weekly_sales;
run;

/* OR */

proc sort data=work.walmart;
    by descending weekly_sales;
run;

/* Which store has maximum standard deviation */

proc summary data=work.walmart;
    class store;
    output out= walmart_standard(drop= _type_ _freq_) std(weekly_sales)=sd_max;
run;

proc print data=work.walmart_standard;
run;

proc sort data=work.walmart_standard;
    by descending sd_max;
run;

proc print data=work.walmart_standard;
run;

/* Find out the coefficient of mean to standard deviation */

proc means data=work.walmart nonobs cv;
class store;
var weekly_sales;
run;

/* OR */

proc means data=work.walmart nonobs cv;
run;

/* Which store/s has good quarterly growth rate in Q3'2012 */

/* Filter year(2012) */

data date_12;
    set work.walmart;

```

```

    where year(date)=2012;
run;

proc print data= date_12;
run;

/* Calculate growth rate */

data growth;
    format growth_rate percent8.2;
    set work.date_12;
    by store date weekly_sales;
    lag_sales = ifn(first.store,0,lag(weekly_sales));
    growth_rate = (weekly_sales/lag_sales)-1;
    drop lag_sales;
run;

proc print data=growth;
run;

/* Convert the normat data into timeseries data */

proc timeseries data= growth out= good_growth;
    by store;
    id date interval= qtr accumulate=total;
    var growth_rate;
run;

proc print data= good_growth;
run;

/* From timeseries data filterd only Q3 observations */

data good_growth_rate;
    set good_growth;
    where qtr(date)= 3;
run;

proc print data=good_growth_rate;
run;

/* Now Sort the data to see the good growth rate store wise */

proc sort data= good_growth_rate;
    by descending growth_rate;
run;

proc print data= good_growth_rate;
run;

/* Some holidays have a negative impact on sales.
Find out holidays which have higher sales than
the mean sales in non-holiday season for all stores together */

/* Separate the holiday dates from main dataset's date */

data holiday;
    set work.walmart;
    where holiday_flag=1;
run;

proc print data=holiday;
run;

/* Separate the non-holiday dates from main dataset's date */

data non_holiday;
    set work.walmart;
    where holiday_flag=0;
run;

```

```

proc print data=non_holiday;
run;

/* Calculate the mean weekly_sales of the non-holiday data */

proc means data= non_holiday mean nonobs;
  output out= mean_sales;
  var weekly_sales;
run;

/* Compare the mean weekly_sales of the non-holiday data with weekly_sales of the holiday data */

proc sql;
  create table holiday_sales as
  select store, weekly_sales, date, holiday_flag as holiday,
  case
  when weekly_sales > 1041256.38 then 'Higher'
  when weekly_sales < 1041256.38 then 'Lower'
  end
  as higher_sales
  from holiday;
quit;

proc print data= holiday_sales;
run;

/* Finally found out holidays which have higher sales than
the mean sales in non-holiday season for all stores together */

data higher_holiday_sales;
  set work.holiday_sales;
  where higher_sales = 'Higher';
  drop higher_sales;
  title 'Higher Sales during Holidays';
run;

proc print data= higher_holiday_sales;
run;

/* Provide a monthly and semester view of sales in units and give insights */

/* Monthly view of sales in units */

/* Convert walmart data into timeseries data */

proc timeseries data= work.walmart
  out= monthly_sales;
  by store;
  id date interval=month accumulate=total;
  var weekly_sales holiday_flag temperature fuel_price cpi unemployment;
run;

proc print data=work.monthly_sales;
  format weekly_sales dollar16.2;
run;

/* Giving insights */

/* Checking the correlation */

proc corr data= work.monthly_sales;
run;

/* 1. Doing Comparison */

/* a) Bar Chart */

proc sgplot data = work.monthly_sales;
  hbar store/response = weekly_sales stat= sum
  datalabel datalabelattrs=(weight=bold);
  title 'Total Views by Store';
run;

```

```
/* b) Clustered Bar Chart / Column Chart */
```

```
data date;  
  set work.monthly_sales;  
  month = month(Date);  
  month_name=PUT(Date,monname.);  
  put month_name= @;  
run;
```

```
proc print data= date;  
run;
```

```
proc sgplot data= date;  
  vbar store/ response= weekly_sales group=month_name groupdisplay=cluster  
        datalabel datalabelattrs = (weight = bold) dataskin=gloss; yaxis grid;  
  title 'Total View by monthly wise';  
run;
```

```
/* 2) Studying relationship */
```

```
/* a) Bubble Chart */
```

```
proc sgplot data = work.monthly_sales;  
  bubble X=weekly_sales Y=store size= weekly_sales  
  /fillattrs=(color = teal) datalabel = store;  
run;
```

```
/* b) Scatter Plot for Relationship */
```

```
proc sgplot data = work.monthly_sales;  
  title 'Relationship of Store with Weekly_sales';  
  scatter X= weekly_sales Y = store/  
          markerattrs=(symbol=circlefilled size=15);  
run;
```

```
/* 3. Studying Distribution */
```

```
/* a) Histogram */
```

```
proc sgplot data = work.monthly_sales;  
  histogram weekly_sales/fillattrs=(color = steel)scale = proportion;  
  density weekly_sales;  
run;
```

```
/* b) Scatter Plot */
```

```
proc sgplot data = work.monthly_sales;  
  scatter X= date Y = weekly_sales/group= store groupdisplay=cluster  
          markerattrs=(symbol=circlefilled size=15);  
run;
```

```
/* 4) Composition */
```

```
/* a) Stacked Column Chart: */
```

```
proc sgplot data= work.monthly_sales;  
  title 'Weekly_sales by Store and date';  
  vbar date / response= weekly_sales group= store stat=percent datalabel;  
  xaxis display=(nolabel);  
  yaxis grid label='Weekly_sales';  
run;
```

```
/* Semester view of sales in units */
```

```
/* Convert walmart data into timeseries data */
```

```
proc timeseries data= work.walmart  
  out= semester_sales;  
  by store;  
  id date interval= semiyear accumulate= total;  
  var weekly_sales holiday_flag temperature fuel_price cpi unemployment;  
run;
```

```

proc print data= work.semester_sales;
run;

/* Giving insights */

/* Checking the correlation */

proc corr data= work.semester_sales;
run;

/* 1. Doing Comparison */

/* a) Bar Chart */

proc sgplot data = work.semester_sales;
    hbar store/response = weekly_sales stat= sum
        datalabel datalabelattrs=(weight=bold);
    title 'Total Views by Store';
run;

/* b) Clustered Bar Chart / Column Chart */

data date;
    set work.semester_sales;
    month = month(Date);
    month_name=PUT(Date,monname.);
    put month_name= @;
run;

proc print data= date1;
run;

proc sgplot data= date;
    vbar store/ response= weekly_sales group=month_name groupdisplay=cluster
        datalabel datalabelattrs = (weight = bold) dataskin=gloss; yaxis grid;
    title 'Total View by monthly wise';
run;

/* 2) Studying relationship */

/* a) Bubble Chart */

proc sgplot data = work.semester_sales;
    bubble X=weekly_sales Y=store size= weekly_sales
        /fillattrs=(color = teal) datalabel = store;
run;

/* b) Scatter Plot for Relationship */

proc sgplot data = work.semester_sales;
    title 'Relationship of Store with Weekly_sales';
    scatter X= weekly_sales Y = store/
        markerattrs=(symbol=circlefilled size=15);
run;

/* 3. Studying Distribution */

/* a) Histogram */

proc sgplot data = work.semester_sales;
    histogram weekly_sales/fillattrs=(color = steel)scale = proportion;
    density weekly_sales;
run;

/* b) Scatter Plot */

proc sgplot data = work.semester_sales;
    scatter X= date Y = weekly_sales/group= store groupdisplay=cluster
        markerattrs=(symbol=circlefilled size=15);
run;

/* 4) Composition */

/* a) Stacked Column Chart: */

```

```

proc sgplot data= work.semester_sales;
    title 'Weekly_sales by Store and date';
    vbar date / response= weekly_sales group= store stat=percent datalabel;
    xaxis display=(nolabel);
    yaxis grid label='Weekly_sales';
run;

/* For Store 1 - Build prediction models to forecast demand */

/* Store-1 data */

data store1;
    set work.walmart;
    where store = 1;
run;

proc print data= store1;
run;

/* Convert store-1 data into timeseries data */

proc timeseries data= store1
    out= store_1;
    by store;
    id date interval= month accumulate= total;
    var weekly_sales holiday_flag temperature fuel_price cpi unemployment;
run;

proc print data= work.store_1;
run;

/* Build Model */

ods noproctitle;
ods graphics / imagemap=on;

proc arima data=WORK.STORE_1 plots
    (only)=(series(corr crosscorr) residual(corr normal)
        forecast(forecast forecastonly) );
    identify var=Weekly_Sales(1);
    estimate p=(1 2 3) q=(1) method=ML;
    forecast lead=4 back=0 alpha=0.05;
    outlier;
run;

```