

```

libname mydata '/home/u49985123/sasuser.v94';

proc import datafile = '/home/u49985123/stores.csv'
  dbms = csv
  out = mydata.stores
  replace;
run;

proc import datafile = '/home/u49985123/sales.csv'
  dbms = csv
  out = mydata.sales
  replace;
run;

proc import datafile = '/home/u49985123/features.csv'
  dbms = csv
  out = mydata.features
  replace;
run;

/* find missing values*/
proc format;
  value missing low-high = 'Value'
              other = 'Missing';
  value $missing ' ' = 'Missing'
              other = 'Value';
run;

proc freq data=mydata.features;
  tables _all_/missing;
  format _character_ $missing. _numeric_ missing.;
  title 'Quick QC Check';
run;

/* check*/
proc freq data= mydata.features_correct;
  tables Store Date Temperature Fuel_price MarkDown1 MarkDown2 MarkDown3 MarkDown4 MarkDown5 CPI Unemployment IsHoliday;
run;

proc print data=mydata.features;
run;

/* delete missing values*/
data mydata.features_no_null;
  set mydata.features;
  if CPI=" " then delete;
  if Unemployment = " " then delete;
run;

/*look for table*/
proc print data=mydata.features_no_null;
run;

/* join*/
proc sql;
create table myData.join as
select
  s.*,
  f.Temperature, f.Fuel_price, f.MarkDown1, f.MarkDown2, f.MarkDown3, f.MarkDown4, f.MarkDown5, f.CPI, f.Unemployment,
  st.Type, st.Size
from myData.sales s
  left join myData.features_no_null f on s.Store = f.Store
  left join myData.stores st on s.Store= st.Store;
quit;
/* from this point couldn't check if code runs well due to reached Quotas in SAS Studio*/
/* 1.sales by store*/
proc sql;
create table mydata.sales_by_store as
select
  Store, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit
from myData.join
group by Store;
quit; /* shows avg profits by store, also largest and minimal profits*/

/*2. make profit groups*/
data mydata.profit_groups;
  set mydata.join;
  if Weekle_Sales >100000 then Profit = "Low";

```

```

else if 10000<= Weekly_Sales< 20000 then Profit = "Good";
else Profit = "Very good";
run;
/*3.sales by temperature */
proc sql;
create table mydata.sales_by_Temperature as
select
    Profit, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit, avg(Temperature)
from myData.profit_groups
group by Profit;
quit; /* shows avg temperature and profit in Low, Good and Very good profit groups. Also largest and minimal profits in those groups */
/*4. sales by fuel prices*/
proc sql;
create table mydata.sales_by_Fuel_price as
select
    Profit, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit, avg(Fuel_Price)
from myData.profit_groups
group by Profit;
quit; /* shows avg fuel price and profit in Low, Good and Very good profit groups. Also largest and minimal profits in those groups */
/* 5.sales by CPI*/
proc sql;
create table mydata.sales_by_CPI as
select
    Profit, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit, avg(CPI)
from myData.profit_groups
group by Profit;
quit; /* shows avg CPI and profit in Low, Good and Very good profit groups. Also largest and minimal profits in those groups */
/*6. sales by Unemployment*/
proc sql;
create table mydata.sales_by_Unemployment as
select
    Profit, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit, avg(Unemployment)
from myData.profit_groups
group by Profit;
quit; /* shows avg unemployment and profit in Low, Good and Very good profit groups. Also largest and minimal profits in those groups */
/*7. sales by store type*/
proc sql;
create table mydata.sales_by_store as
select
    Type, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit
from myData.join
group by Type
order by avg.profit;
quit; /* shows avg profits by store type, also largest and minimal profits */
/* 8.sales by item storage*/
proc sql;
create table mydata.sales_by_Storage as
select
    Profit, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit, avg(Size)
from myData.profit_groups
group by Profit;
quit; /* shows avg number of items store can hold and profit in Low, Good and Very good profit groups. Also largest and minimal profits in those groups */
/* 9. sales depending on holidays*/
proc sql;
create table mydata.sales_by_Holidays as
select
    IsHoliday, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit
from myData.join
group by IsHoliday;
quit; /* shows avg profits during the holiday and not holiday week, also largest and minimal profits */
/* sales before promotion of discounts*/
/*10.delete rows with date when discounts are promoted*/
data mydata.before_discounts;
set myData.join;
if Date > 01/11/2010 then delete;
run;

/* repeat steps 1-9 with new table mydata.before_discounts*/
/*examples*/
/* 1.sales by store*/
proc sql;
create table mydata.salesNoDisc_by_store as
select
    Store, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit

```

```

from mydata.before_discounts
group by Store;
quit; /* shows avg profits by store before discounts, also largest and minimal profits in sores*/

/*2. make profit groups*/
data mydata.profit_groups2;
set mydata.before_discounts;
if Weekle_Sales >100000 then Profit = "Low";
else if 100000<= Weekle_Sales< 20000 then Profit = "Good";
else Profit = "Very good";
run;

/*3.sales by temperature */
proc sql;
create table mydata.salesNoDisc_by_Temperature as
select
    Profit, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit, avg(Temperature) as avg.temperature
from mydata.profit_groups2
group by Profit;
quit; /* shows avg, max and min temperature by profit group before discounts*/

/* sales during discounts*/
/*11. keep data only with promoted discounts*/
data mydata.during_discounts;
set mydata.join;
where MarkDown1 > 0 or MarkDown2>0 or MarkDown3>0 or MarkDown4>0 or MarkDown5>0;
run;
/* repeat steps 1-9 with new table mydata.during_discounts*/
/*examples*/
/* 1.sales by store*/
proc sql;
create table mydata.salesDisc_by_store as
select
    Store, avg(Weekly_Sales)as avg.profit, max(Weekly_Sales) as max.profit, min(Weekly_Sales) as min.profit
from mydata.during_discounts
group by Store;
quit; /* shows avg profits by store during discounts, also largest and minimal profits in sores*/

/*2. make profit groups*/
data mydata.profit_groups3;
set mydata.during_discounts;
if Weekle_Sales >100000 then Profit = "Low";
else if 100000<= Weekle_Sales< 20000 then Profit = "Good";
else Profit = "Very good";
run;

/*3.sales by temperature */
proc sql;
create table mydata.salesDisc_by_Temperature as
select

```