```
/*creating library for reference*/
libname STAT1 base "/home/u62159561/sasuser.v94/Naureen_Safoora/EPG1V2/data";

/*specifying excel file location*/
filename reffile '/home/u62159561/my_shared_file_links/u60594276/BSTAT 5325-002 Fall 2022/Course Project/Data/ames.xlsx';

/*used when importing excel files to get required format in SAS*/
options validvarname=v7;

/* gives graphs along with tables*/
ods graphics on;

/*loading the data from excel*/
proc import datafile=reffile dbms=xlsx out=STAT1.ames;
    getnames=yes;
run;

/*displaying contents of data*/
proc contents data=STAT1.ames;
run;


/* dropping ID columns as they are not of any statistical use*/
data STAT1.ames_drop;
set STAT1.ames;
drop PID;
run;

/* descriptive statistics*/
proc means data=STAT1.ames_drop N mean mode median std q1 p25 p50 q3 min max;
run;

/*MISSING VALUES HANDLING*/
title "Means Procedure to check for Missing Values";
proc means data=STAT1.ames_drop NMISS N;
run;

/* remove less missing*/
data STAT1.remove_missing;
    set STAT1.ames_drop;
    if Mas_Vnr_Area=. then delete;
    if BsmtFin_SF_1=. then delete;
    if BsmtFin_SF_2=. then delete;
    if Bsmt_Unf_SF=. then delete;
    if Total_Bsmt_SF=. then delete;
    if Bsmt_Full_Bath=. then delete;
    if Bsmt_Half_Bath=. then delete;
    if Garage_Cars=. then delete;
    if Garage_Area=. then delete;
    if Garage_Yr_Blt=. then delete;
run;

/*missing values*/
title "Means Procedure to check for Missing Values";
proc means data=STAT1.remove_missing NMISS N;  /* Lot_Frontage left */
run;

/* not much correlation, therefore MCAR*/
proc corr data=STAT1.remove_missing plots=matrix;
    with Lot_Frontage;
run;

/* checking Lot_Frontage dist*/
proc univariate data=STAT1.remove_missing;
    var Lot_Frontage;
    histogram Lot_Frontage / normal; /* we get a right skewed dist*/
run;

/* getting median to impute*/
proc means data=STAT1.remove_missing N median;
    var Lot_Frontage;
run;

/* imputing with median*/
data STAT1.impute_missing;
    set STAT1.remove_missing;
    if Lot_Frontage=. then
        Lot_Frontage=69;
run;

/*** Analyze categorical variables ***/
```

```
title "Frequencies for Categorical Variables";
proc freq data=STAT1.impute_missing;
    tables MS_SubClass MS_Zoning Street Alley Lot_Shape Land_Contour Utilities
        Lot_Config Land_Slope Neighborhood Condition_1 Condition_2 Bldg_Type
        House_Style Roof_Style Roof_Matl Exterior_1st Exterior_2nd Mas_Vnr_Type
        Exter_Qual Exter_Cond Foundation Bsmt_Qual Bsmt_Cond Bsmt_Exposure
        BsmtFin_Type_1 BsmtFin_Type_2 Heating Heating_QC Central_Air Electrical
        Kitchen_Qual Functional Fireplace_Qu Garage_Type Garage_Finish Garage_Qual
        Garage_Cond Paved_Drive Pool_QC Fence Misc_Feature Sale_Type Sale_Condition /
        plots=(freqplot);
run;

/* dropping cat. columns as >50% NA or >90% to other ratio */
data STAT1.cat_na_drop;
set STAT1.IMPUTE_MISSING;
drop Alley Street Utilities Land_Slope Condition_2 Roof_Matl Bsmt_Cond Heating Central_Air
 Electrical Functional Garage_Qual Garage_Cond Paved_Drive Pool_QC Fence Misc_Feature;
run;


/*  removing cat. rows with freq < 10 */
data STAT1.cat_less_rem;
    set STAT1.cat_na_drop;
    if MS_SubClass = 040 or MS_SubClass = 150 then delete;
    if MS_Zoning = 'A (agr)' or MS_Zoning = 'I (all)' then delete;
    if Neighborhood = 'Greens' or Neighborhood = 'GrnHill' or Neighborhood = 'Landmrk' then delete;
    if Condition_1 = 'RRNe' or Condition_1 = 'RRNn' then delete;
    if House_Style = '2.5Fin' then delete;
    if Roof_Style = 'Mansard' or Roof_Style = 'Shed' then delete;
    if Exterior_1st = 'BrkComm' or Exterior_1st = 'CBlock' or Exterior_1st = 'ImStucc' or Exterior_1st = 'Stone' then del
    if Exterior_2nd = 'AsphShn' or Exterior_2nd = 'CBlock' or Exterior_2nd = 'Other' or Exterior_2nd = 'Stone' then delet
    if Mas_Vnr_Type = 'CBlock' then delete;
    if Exter_Cond = 'Po' then delete;
    if Foundation = 'Wood' then delete;
    if Bsmt_Qual = 'Po' then delete;
    if Heating_QC= 'Po' then delete;
    if Kitchen_Qual = 'Po' then delete;
    if Sale_Type = 'Con' or Sale_Type = 'ConLI' or Sale_Type = 'ConLw' or Sale_Type = 'Oth' or Sale_Type = 'VWD' then del
    if Sale_Condition = 'AdjLand' then delete;
run;


/* check again for encoding ordinal var */
proc freq data = stat1.cat_less_rem;
       tables MS_SubClass MS_Zoning Lot_Shape Land_Contour Lot_Config Neighborhood
        Condition_1 Bldg_Type House_Style Roof_Style Exterior_1st Exterior_2nd
        Mas_Vnr_Type Exter_Qual Exter_Cond Foundation Bsmt_Qual Bsmt_Exposure
        BsmtFin_Type_1 BsmtFin_Type_2 Heating_QC Kitchen_Qual Fireplace_Qu
        Garage_Type Garage_Finish Sale_Type Sale_Condition/
        plots=(freqplot);
run;

/* encoding ordinal cat var */
title "Transform Ordinal Cat. Data for Regression";
data stat1.cat_ord_encode;
    set stat1.cat_less_rem;
    if Lot_Shape = "Reg" then LotShape_new = 4;
    else if Lot_Shape="IR1" then LotShape_new  = 3;
    else if Lot_Shape = "IR2" then LotShape_new = 2;
    else if Lot_Shape = "IR3" then LotShape_new = 1;
    else LotShape_new = 0;

    if Exter_Qual="Ex"      then ExterQual_new = 4;
    else if Exter_Qual="Gd"     then ExterQual_new = 3;
    else if Exter_Qual="TA"     then ExterQual_new = 2;
    else if Exter_Qual="Fa"     then ExterQual_new = 1;
    else ExterQual_new = 0;

    if Exter_Cond="Ex"      then ExterCond_new = 4;
    else if Exter_Cond="Gd"     then ExterCond_new = 3;
    else if Exter_Cond="TA"     then ExterCond_new = 2;
    else if Exter_Cond="Fa"     then ExterCond_new = 1;
    else ExterCond_new = 0;

    if Bsmt_Qual="Ex"       then BsmtQual_new = 4;
    else if Bsmt_Qual="Gd"  then BsmtQual_new = 3;
    else if Bsmt_Qual="TA"  then BsmtQual_new = 2;
    else if Bsmt_Qual="Fa"  then BsmtQual_new = 1;
    else if Bsmt_Qual="NA"  then BsmtQual_new = 0;

    if Bsmt_Exposure="Gd"   then BsmtExposure_new = 4;
    else if Bsmt_Exposure="Av"  then BsmtExposure_new = 3;
    else if Bsmt_Exposure="Mn"  then BsmtExposure_new = 2;
    else if Bsmt_Exposure="No"  then BsmtExposure_new = 1;
    else BsmtExposure_new = 0;

    if BsmtFin_Type_1="ALQ" then BsmtFinType1_new = 5;
    else if BsmtFin_Type_1="BLQ" then BsmtFinType1_new = 4;
    else if BsmtFin_Type_1="GLQ" then BsmtFinType1_new = 6;
    else if BsmtFin_Type_1="LwQ" then BsmtFinType1_new = 2;
    else if BsmtFin_Type_1="Rec" then BsmtFinType1_new = 3;
    else if BsmtFin_Type_1="Unf" then BsmtFinType1_new = 1;
    else BsmtFinType1_new = 0;

    if BsmtFin_Type_2="ALQ" then BsmtFinType2_new = 5;
    else if BsmtFin_Type_2="BLQ" then BsmtFinType2_new = 4;
    else if BsmtFin_Type_2="GLQ" then BsmtFinType2_new = 6;
    else if BsmtFin_Type_2="LwQ" then BsmtFinType2_new = 2;
    else if BsmtFin_Type_2="Rec" then BsmtFinType2_new = 3;
    else if BsmtFin_Type_2="Unf" then BsmtFinType2_new = 1;
    else BsmtFinType2_new = 0 ;

    if Heating_QC="Ex"      then HeatingQC_new = 4;
    else if Heating_QC="Gd"     then HeatingQC_new = 3;
    else if Heating_QC="TA"     then HeatingQC_new = 2;
    else if Heating_QC="Fa"     then HeatingQC_new = 1;
    else HeatingQC_new = 0;

    if Kitchen_Qual="Ex"        then KitchenQual_new = 4;
    else if Kitchen_Qual="Gd"   then KitchenQual_new = 3;
    else if Kitchen_Qual="TA"   then KitchenQual_new = 2;
    else if Kitchen_Qual="Fa"   then KitchenQual_new = 1;
    else KitchenQual_new = 0;

    if Fireplace_Qu="Ex"        then FireplaceQu_new = 5;
    else if Fireplace_Qu="Gd"   then FireplaceQu_new = 4;
    else if Fireplace_Qu="TA"   then FireplaceQu_new = 3;
    else if Fireplace_Qu="Fa"   then FireplaceQu_new = 2;
    else if Fireplace_Qu="Po"   then FireplaceQu_new = 1;
    else FireplaceQu_new = 0;

    if Garage_Finish="RFn" then GarageFinish_new = 2;
    else if Garage_Finish="Fin" then GarageFinish_new = 3;
    else if Garage_Finish="Unf" then GarageFinish_new = 1;
    else GarageFinish_new = 0;
run;

/* dropping old encoded cat. columns */
data STAT1.cat_ord_encode;
set STAT1.cat_ord_encode;
drop Lot_Shape Exter_Qual Exter_Cond Bsmt_Qual Bsmt_Exposure BsmtFin_Type_1 BsmtFin_Type_2
Heating_QC Kitchen_Qual Fireplace_Qu Garage_Finish;
run;


/*simpsons paradox*/
ods graphics / reset width=6.4in height=4.8in imagemap;
proc sgplot data=STAT1.cat_ord_encode;
    vbar Neighborhood / response=SalePrice stat=mean datalabel;
    yaxis grid;
run;
ods graphics / reset;


/* dist edwards noridge*/
ods graphics / reset width=6.4in height=4.8in imagemap;
proc sort data=STAT1.cat_ord_encode out=_BarChartTaskData;
    by House_Style;
run;
proc sgplot data=_BarChartTaskData;
    by House_Style;
    vbar Neighborhood / response=SalePrice fillattrs=(color=CX6818b3) datalabel
        stat=mean;
    yaxis grid;
run;
ods graphics / reset;
proc datasets library=WORK noprint;
    delete _BarChartTaskData;
    run;

/* imbalanced data NAmes*/
ods graphics / reset width=6.4in height=4.8in imagemap;
proc sgplot data=STAT1.cat_ord_encode;
    vbar Neighborhood / group=House_Style groupdisplay=stack datalabel
```

```
        stat=percent;
    yaxis grid;
    keylegend / location=inside;
run;
ods graphics / reset;


/* getting corelation heatmap of all var*/
%paint(values=-1 to 1 by 0.01, macro=setstyle,
colors=CXFF6767 magenta CX6767FF cyan white white white
white white white cyan CX6767FF magenta CXFF6767)
proc template;
delete Base.Corr.StackedMatrix / store=sasuser.templat;
edit Base.Corr.StackedMatrix;
column (RowName RowLabel) (Matrix);
header 'Pearson Correlation Coefficients';
edit matrix;
format=5.2;
%setstyle(backgroundcolor)
end;
end;
quit;
proc corr data=STAT1.cat_ord_encode noprob;
ods select PearsonCorr;
run;  /* >0.7 correlation with SalePrice -> Overall_Qual Gr_Liv_Area */

/* other >0.8 var Garage_Yr_Blt and Year_Built, BsmtFinType2_new and BsmtFin_SF_2, Total_Bsmt_SF and _1st_Flr_SF,
TotRms_AbvGrd and Gr_Liv_Area, FireplaceQu_new and Fireplaces, Garage_Area and Garage_Cars */
/* as these have high, take only one from set with highest corr with SalePrice to avoid multicollinearity*/
proc corr data=STAT1.cat_ord_encode plots=matrix;
    var SalePrice Garage_Yr_Blt Year_Built BsmtFinType2_new BsmtFin_SF_2 Total_Bsmt_SF _1st_Flr_SF
    TotRms_AbvGrd  Gr_Liv_Area FireplaceQu_new Fireplaces Garage_Area Garage_Cars ;
    with SalePrice;
run;

/* dropping highly correlated independent var. */
data STAT1.drop_corr;
set STAT1.cat_ord_encode;
drop Garage_Yr_Blt BsmtFinType2_new _1st_Flr_SF TotRms_AbvGrd Fireplaces Garage_Area;
run;

/* kurtosis and skewness */
ods noproctitle;
ods graphics / imagemap=on;
proc univariate data=STAT1.drop_corr;
    ods select Histogram;
    var SalePrice;
    histogram SalePrice / normal kernel;
    inset n skewness kurtosis / position=ne;
run;

/* checking outliers in SalePrice with CooksD */
proc reg data=STAT1.drop_corr  plots(only) = (CooksD(label));
   model SalePrice=Gr_Liv_Area Overall_Qual;
   id Order;
   output out=RegOut pred=Pred cookd=CooksD;
run; quit; /* r sq is 0.73*/

/* remove order = 1499,2181,2182 as it is outlier */
data STAT1.outlier_drop;
    set STAT1.drop_corr;
    if Order=1499 or Order=2181 or Order = 2182 then delete;

/* checking outliers in SalePrice with CooksD */
proc reg data=STAT1.outlier_drop  plots(only) = (CooksD(label));
   model SalePrice=Gr_Liv_Area Overall_Qual;
   id Order;
   output out=RegOut pred=Pred cookd=CooksD;
run; quit; /* r sq is 0.76*/


/* hyp testing*/
proc ttest data = stat1.outlier_drop h0=170000
    plot(only shownull) = interval;
var SalePrice;
title 'Testing if mean saleprice is 170000';
run; /* p <0.0001 , reject null hyp, therefore, mean <> 170k */


/*H0: There is no difference between the mean SalePrice of homes sold in different months.*/
/*Ha: There is a difference between mean SalePrice of homes sold in different months.*/
PROC ANOVA DATA= stat1.outlier_drop ;
```

```
CLASS Mo_Sold;
MODEL SalePrice= Mo_Sold;
MEANS Mo_Sold;
RUN;
QUIT;

/* which month should i sell my house, buy house */
ods graphics / reset width=6.4in height=4.8in imagemap;
proc sgplot data=STAT1.OUTLIER_DROP;
    vbar Mo_Sold / response=SalePrice stat=mean datalabel;
    yaxis grid;
run;
ods graphics / reset;


/* stepwise selection */
proc reg data=stat1.outlier_drop plots(only)=(adjrsq);
    Stepwise: model SalePrice=Order Lot_Frontage Lot_Area Overall_Qual Overall_Cond Year_Built
        Year_Remod_Add Mas_Vnr_Area BsmtFin_SF_1 BsmtFin_SF_2 Bsmt_Unf_SF
        Total_Bsmt_SF _2nd_Flr_SF Low_Qual_Fin_SF Gr_Liv_Area Bsmt_Full_Bath
        Bsmt_Half_Bath Full_Bath Half_Bath Bedroom_AbvGr Kitchen_AbvGr Garage_Cars
        Wood_Deck_SF Open_Porch_SF Enclosed_Porch _3Ssn_Porch Screen_Porch Pool_Area
        Misc_Val Mo_Sold Yr_Sold LotShape_new ExterQual_new ExterCond_new
        BsmtQual_new BsmtExposure_new BsmtFinType1_new HeatingQC_new KitchenQual_new
        FireplaceQu_new GarageFinish_new /selection=stepwise
        slentry=0.01 slstay=0.01 details;
    run;

title "Final Regression Analysis";
proc reg data=stat1.outlier_drop ;
    No_var_selection: model SalePrice=  Overall_Qual  Gr_Liv_Area    BsmtFin_SF_1
    ExterQual_new Total_Bsmt_SF Lot_Area KitchenQual_new Mas_Vnr_Area BsmtExposure_new
    Kitchen_AbvGr Garage_Cars Overall_Cond Year_Built Lot_Frontage  Bedroom_AbvGr   ;
    run;  /* adj r sq 0.8860 */


/*Normalize all variables to perform PCA on them*/

title "Normalize all Predictor varibles for PCA";
proc standard data=stat1.outlier_drop mean =0 std=1
out = House_priceZ;
var Order Lot_Frontage Lot_Area Overall_Qual Overall_Cond Year_Built
        Year_Remod_Add Mas_Vnr_Area BsmtFin_SF_1 BsmtFin_SF_2 Bsmt_Unf_SF
        Total_Bsmt_SF _2nd_Flr_SF Low_Qual_Fin_SF Gr_Liv_Area Bsmt_Full_Bath
        Bsmt_Half_Bath Full_Bath Half_Bath Bedroom_AbvGr Kitchen_AbvGr Garage_Cars
        Wood_Deck_SF Open_Porch_SF Enclosed_Porch _3Ssn_Porch Screen_Porch Pool_Area
        Misc_Val Mo_Sold Yr_Sold LotShape_new ExterQual_new ExterCond_new
        BsmtQual_new BsmtExposure_new BsmtFinType1_new HeatingQC_new KitchenQual_new
        FireplaceQu_new GarageFinish_new ;
run;


title "Summary of Normalized Data";
proc means data=House_priceZ;
var Order Lot_Frontage Lot_Area Overall_Qual Overall_Cond Year_Built
        Year_Remod_Add Mas_Vnr_Area BsmtFin_SF_1 BsmtFin_SF_2 Bsmt_Unf_SF
        Total_Bsmt_SF _2nd_Flr_SF Low_Qual_Fin_SF Gr_Liv_Area Bsmt_Full_Bath
        Bsmt_Half_Bath Full_Bath Half_Bath Bedroom_AbvGr Kitchen_AbvGr Garage_Cars
        Wood_Deck_SF Open_Porch_SF Enclosed_Porch _3Ssn_Porch Screen_Porch Pool_Area
        Misc_Val Mo_Sold Yr_Sold LotShape_new ExterQual_new ExterCond_new
        BsmtQual_new BsmtExposure_new BsmtFinType1_new HeatingQC_new KitchenQual_new
        FireplaceQu_new GarageFinish_new    ;
run;


title "PCA for all Normalized Variables";
proc princomp data=House_priceZ
plots=all;
var Order Lot_Frontage Lot_Area Overall_Qual Overall_Cond Year_Built
        Year_Remod_Add Mas_Vnr_Area BsmtFin_SF_1 BsmtFin_SF_2 Bsmt_Unf_SF
        Total_Bsmt_SF _2nd_Flr_SF Low_Qual_Fin_SF Gr_Liv_Area Bsmt_Full_Bath
        Bsmt_Half_Bath Full_Bath Half_Bath Bedroom_AbvGr Kitchen_AbvGr Garage_Cars
        Wood_Deck_SF Open_Porch_SF Enclosed_Porch _3Ssn_Porch Screen_Porch Pool_Area
        Misc_Val Mo_Sold Yr_Sold LotShape_new ExterQual_new ExterCond_new
        BsmtQual_new BsmtExposure_new BsmtFinType1_new HeatingQC_new KitchenQual_new
        FireplaceQu_new GarageFinish_new        ;
run;

/*Only 1-12 Prin Comp Eigenvalue > 1*/
title "PCA for 12 components with all normalized Variables";
proc princomp data=House_priceZ
n=12
```

```
plots=all
out=pca_House_price;
var         Order Lot_Frontage Lot_Area Overall_Qual Overall_Cond Year_Built
        Year_Remod_Add Mas_Vnr_Area BsmtFin_SF_1 BsmtFin_SF_2 Bsmt_Unf_SF
        Total_Bsmt_SF _2nd_Flr_SF Low_Qual_Fin_SF Gr_Liv_Area Bsmt_Full_Bath
        Bsmt_Half_Bath Full_Bath Half_Bath Bedroom_AbvGr Kitchen_AbvGr Garage_Cars
        Wood_Deck_SF Open_Porch_SF Enclosed_Porch _3Ssn_Porch Screen_Porch Pool_Area
        Misc_Val Mo_Sold Yr_Sold LotShape_new ExterQual_new ExterCond_new
        BsmtQual_new BsmtExposure_new BsmtFinType1_new HeatingQC_new KitchenQual_new
        FireplaceQu_new GarageFinish_new        ;
run;


title "Multiple Linear Regression Anlysis on PCs";
proc reg data=pca_House_price outest=est_House_price2 plots=all;
    model   SalePrice = Prin1 - Prin12
    /   stb dwProb pcorr1 VIF ss1 ss2 selection=MAXR;
quit;

title "Multiple Regression Analysis on PC with significant variables";
proc reg data=pca_House_price outest=est_House_price3 plots(label)=all;
    model   SalePrice = Prin1 Prin3 Prin5- Prin10;
quit;


/*clustering*/
ods graphics on;
proc cluster data=stat1.outlier_drop method=centroid ccc print=15 pseudo out= tree;
var Year_Built SalePrice;
run;
ods graphics off;
```