MODUL 11

SAS® Viya® STEP-BY-STEP DATA SCIENCE PROJECT

THEME DESCRIPTION

Students understand and are able to implement science projects step by step with an outline of the workflow starting from provide the data to identifying business opportunities, creating Train and Test data sets and analyzing target variables and predictor variables in order to prepare data models.

WEEKLY LEARNING OUTCOMES (SUB-LESSONS)

CLO-3-Sub-CLO-11:
Able to implement the early stages of data science implementation including business opportunities as well as analysis and selection of effective data variables– C4.

Through the following learning steps**:**

1. Get the Data
2. Create Geographic Map
3. Exploring data files.
   a) Split data into TRAIN and TEST datasets at an 80/20 split
   b) Investigate the Target Variable
4. Explore the target variable:
   a) Eliminate outliers and create log transformed price variable
   b) Analyze the relationship between neighbourhood and zipcode variables
   c) Create global numeric variables
   d) Create global variables for character variables
5. Predictor Variable Analysis
6. Collinearity Analysis

PRACTICUM SUPPORT

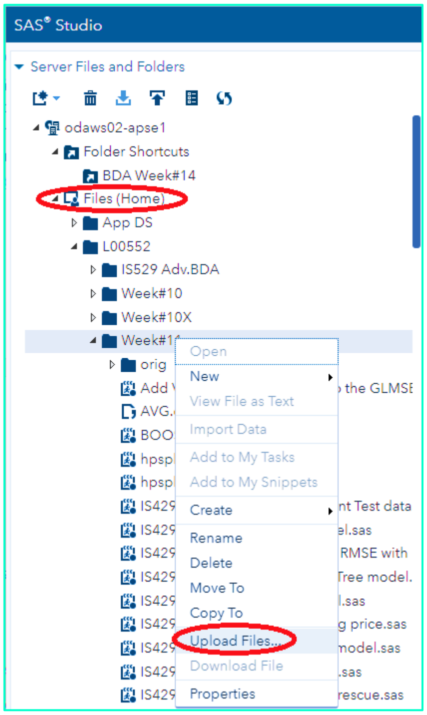
a. Windows Operating System
b. (any) Browser Application

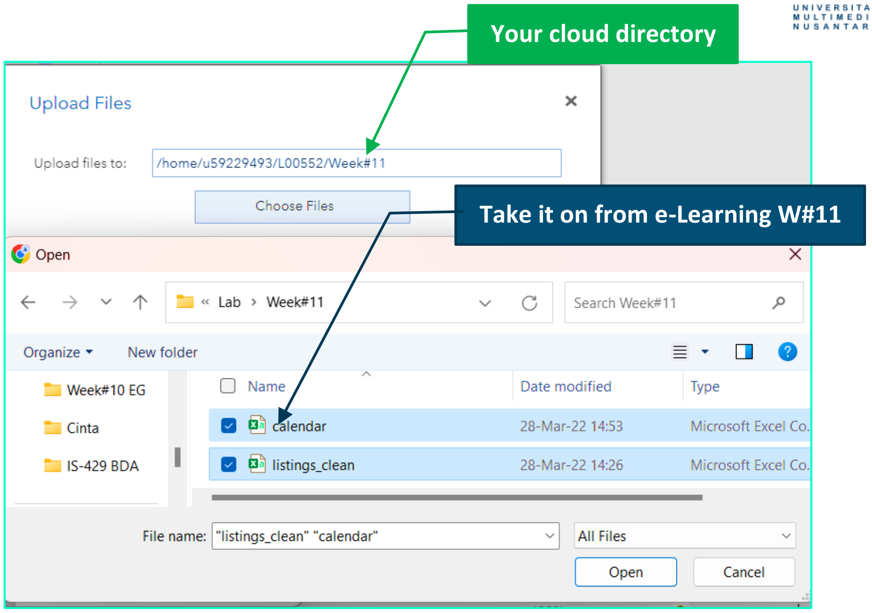
PRACTICUM STEPS

1. Get the Data

- The data provide of New York City Inside Airbnb data sets consisting of Listings and Calendar, have placed them in "/home/u59229493/L00552 shared-drive/u59229493/Week#11/" repository.
- These are the two main data sets that we will use for our analysis: listings_clean.csv and calendar.csv.
- If you fail to apply the L00552 shared-drive mentioned above then you can upload it directly from your local laptop/PC to your home directory using the following command:

a. Start, https://welcome.oda.sas.com





Your cloud directory

Upload Files

Upload files to: /home/u59229493/L00552/Week#11

Take it on from e-Learning W#11

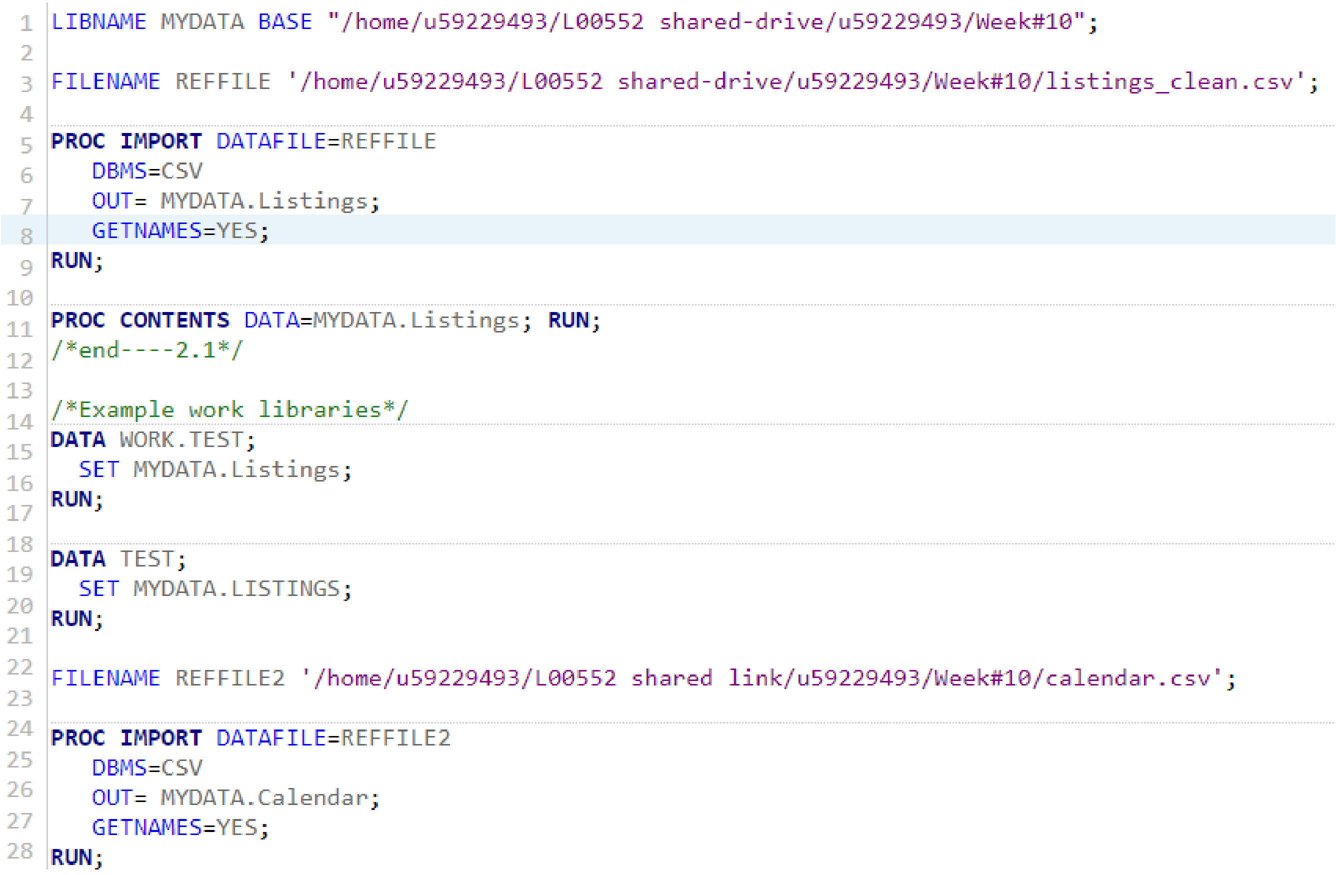
File name: "listings_clean" "calendar"

All Files

b. Open SAS®Studio, click options and click New SAS program (F4).

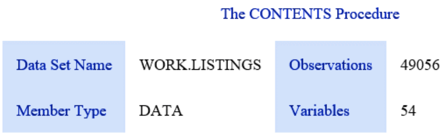
c. If you are not using the L00552 shared-drive, then change the folder according to the one you have!

d. Type it on these codes below:



```
LIBNAME MYDATA BASE "/home/u59229493/L00552 shared-drive/u59229493/Week#10";

FILENAME REFFILE '/home/u59229493/L00552 shared-drive/u59229493/Week#10/listings_clean.csv';

PROC IMPORT DATAFILE=REFFILE
   DBMS=CSV
   OUT= MYDATA.Listings;
   GETNAMES=YES;
RUN;

PROC CONTENTS DATA=MYDATA.Listings; RUN;
/*end----2.1*/

/*Example work libraries*/
DATA WORK.TEST;
  SET MYDATA.Listings;
RUN;

DATA TEST;
  SET MYDATA.LISTINGS;
RUN;

FILENAME REFFILE2 '/home/u59229493/L00552 shared link/u59229493/Week#10/calendar.csv';

PROC IMPORT DATAFILE=REFFILE2
   DBMS=CSV
   OUT= MYDATA.Calendar;
   GETNAMES=YES;
RUN;
```

e. After the above program is executed then you will get the data contents of Listings database as follows:



The CONTENTS Procedure

Data Set Name	WORK.LISTINGS	Observations	49056
Member Type	DATA	Variables	54

- The first two rows of the PROC CONTENTS output show that there are 49,056 observations with 54 variables in the Listings data set.
- The PROC CONTENTS procedure also produces a list of variables and attributes.
- The output of the above program produces several tables in the following libraries:
  1) MYDATA.Listings
  2) WORK.TEST
  3) MYDATA.LISTINGS

f. Save it on your program into "IS429 Lab Week#11_Get the Data.sas".

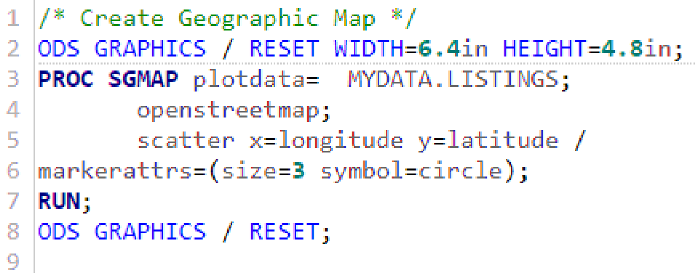
g. **Print your results as shown on Ouput figure A and save it on "IS-429 Lab Week#11A Get the Data NIMNAME".pdf**

2. Create Geographic Map

- SAS does an excellent job of creating picture-perfect maps with a few lines of code.

a. Click New SAS program (F4).

b. Type it on these codes below:



```
/* Create Geographic Map */
ODS GRAPHICS / RESET WIDTH=6.4in HEIGHT=4.8in;
PROC SGMAP plotdata=  MYDATA.LISTINGS;
       openstreetmap;
       scatter x=longitude y=latitude /
markerattrs=(size=3 symbol=circle);
RUN;
ODS GRAPHICS / RESET;

```

- They are highly customizable in Figure 10.1 with the ability to create choropleth maps and bubble maps along with lots of other customizable features.

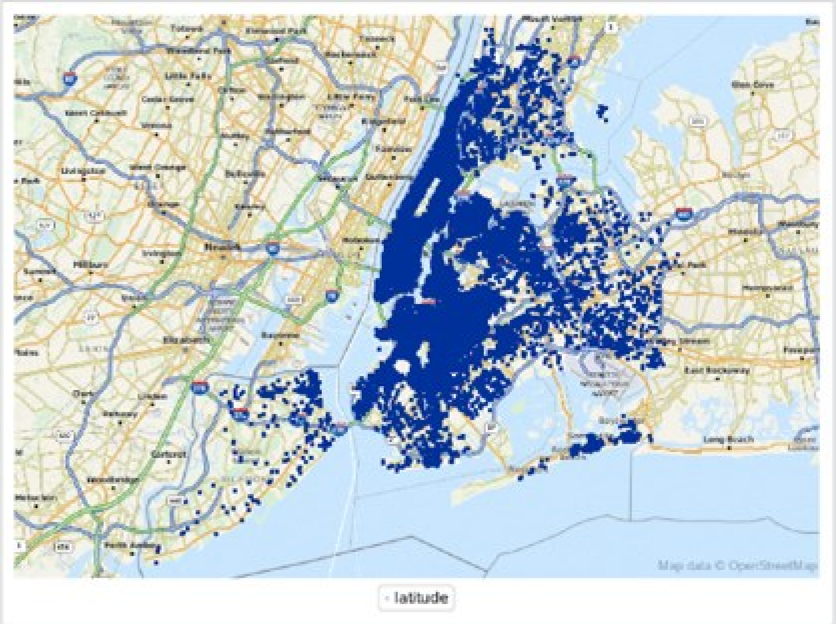
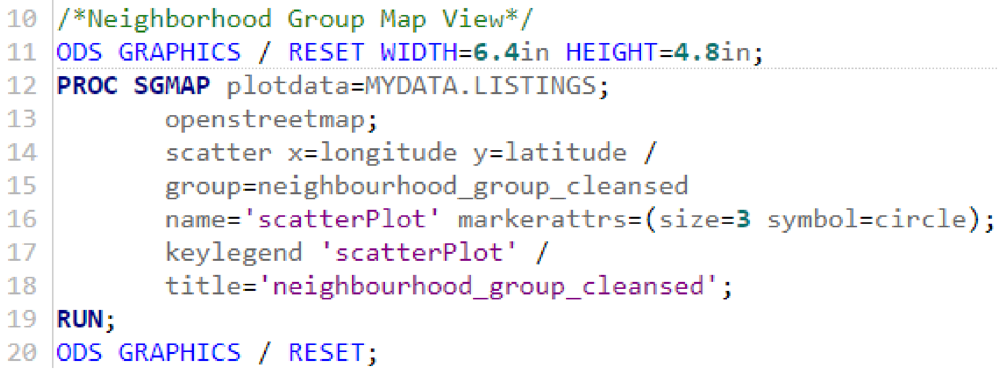


Figure 10.1 choropleth maps and bubble maps

- For additional information, you can group the data points by a categorical feature.
- The program code below shows the colors of the Airbnb listing-coded by the Neighborhood Group feature."

c. Type it on these codes below:



```
/*Neighborhood Group Map View*/
ODS GRAPHICS / RESET WIDTH=6.4in HEIGHT=4.8in;
PROC SGMAP plotdata=MYDATA.LISTINGS;
        openstreetmap;
        scatter x=longitude y=latitude /
        group=neighbourhood_group_cleansed
        name='scatterPlot' markerattrs=(size=3 symbol=circle);
        keylegend 'scatterPlot' /
        title='neighbourhood_group_cleansed';
RUN;
ODS GRAPHICS / RESET;
```

- The map view that is grouped by neighborhood in Figure 10.2 provides an easy visual snapshot of the data. We can see that Manhattan and Brooklyn appear to have a majority of the listings while The Bronx and Staten Island have relatively few listings.

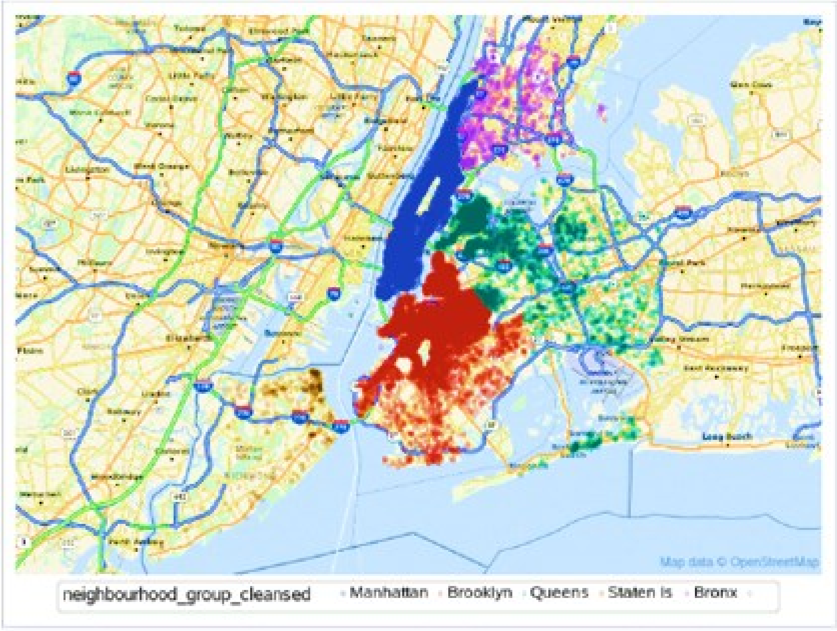


Figure 10.2 Map View by Neighborhood

d. Save it on your program into "IS429 Lab Week#11_Geographic Map.sas"

e. **Screenshot your results as shown on Ouput figure B and save it on "IS-429 Lab Week#11B Geographic Map-NIMNAME.JPG/PNG/pdf".**

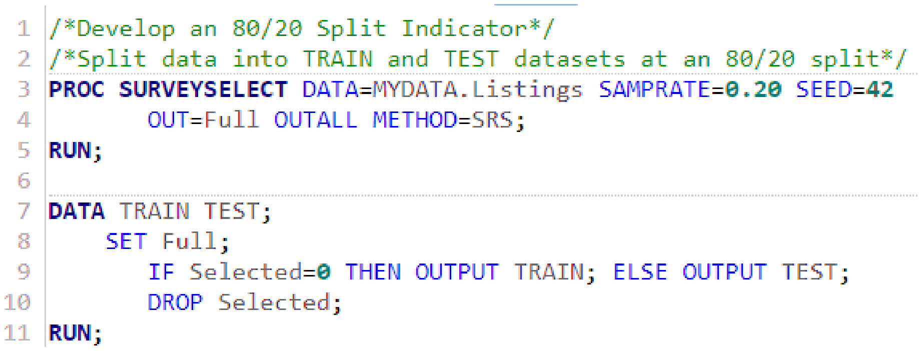
3. Exploring data files

- Before we start investigating the data and making any adjustments, we first need to split the data into TRAIN and TEST data sets.
- Create the TRAIN and TEST datasets using the **PROC SURVEY SELECT** procedure.
- This is a great tool to use which allows you **to set your sampling rate** (SAMPRATE). We use the 80/20 division for our TRAIN and TEST, with 80% of the data going to the TRAIN data set.

- For this Airbnb project, we applied a simple random sample without replacement. This sampling method is carried out by METHOD = SRS statement.
- Next is to build the seed. This determines the initial seed for random number generation. In the model development phase of the project, we want to make sure that we create the same data set every time we run our code. Seed formation is a great way to ensure that you can replicate your results.
- We use any number for the seed value. But for this Airbnb project we use 42 because **Douglas Adams** shows us that **42 is the "Answer to the Highest"** Questions about Life, the Universe, and Everything" in his book "**Hitchhiker's Guide Series for Galaxy**".

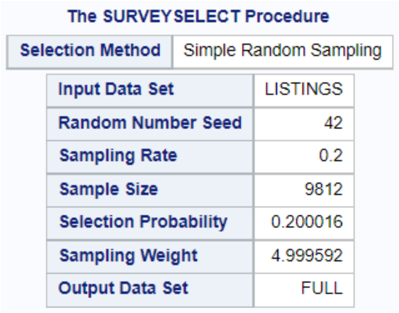
a. Still in the worksheet of SAS®Studio, click options and click New SAS program (F4).

b. Type it on the codes below and run.



```
/*Develop an 80/20 Split Indicator*/
/*Split data into TRAIN and TEST datasets at an 80/20 split*/
PROC SURVEYSELECT DATA=MYDATA.Listings SAMPRATE=0.20 SEED=42
       OUT=Full OUTALL METHOD=SRS;
RUN;

DATA TRAIN TEST;
    SET Full;
       IF Selected=0 THEN OUTPUT TRAIN; ELSE OUTPUT TEST;
       DROP Selected;
RUN;
```

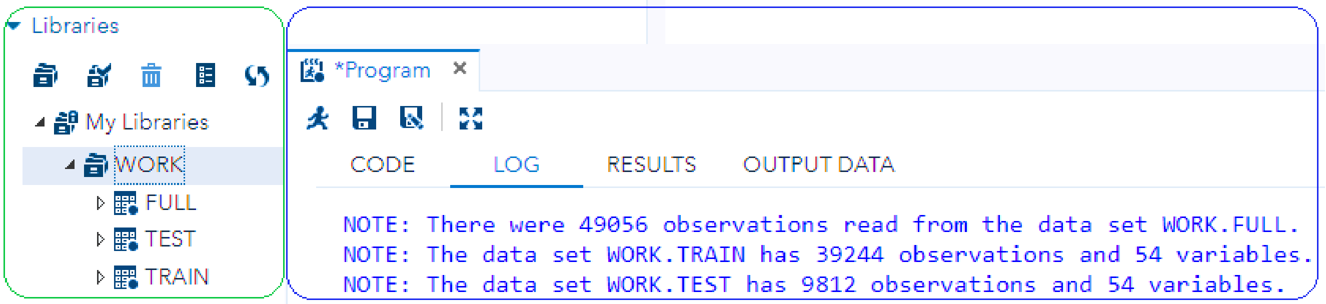
- The PROC SURVEYSELECT procedure simply adds a field to the existing MYDATA.Listings data set.
- The Selected field is a binary variable that is coded as a 1 or 0. The 20% of observations that are randomly selected will have an indicator of 1, while the remaining 80% will have an indicator of 0.
- The output of the PROC SURVEYSELECT procedure provides you with a summary statement of the chosen parameters. The Sample Size column shows the number of observations that were randomly selected by the algorithm.
- A quick confirmation shows above that 9812/4999592 = 20%.
- In addition to generating the sample size column above, the program generates three FULL TRAIN and TEST tables in your WORK repository.



The SURVEYSELECT Procedure

Selection Method	Simple Random Sampling

Input Data Set	LISTINGS
Random Number Seed	42
Sampling Rate	0.2
Sample Size	9812
Selection Probability	0.200016
Sampling Weight	4.999592
Output Data Set	FULL



```
NOTE: There were 49056 observations read from the data set WORK.FULL.
NOTE: The data set WORK.TRAIN has 39244 observations and 54 variables.
NOTE: The data set WORK.TEST has 9812 observations and 54 variables.
```

f. Screenshotted your log and WORK repository as shown in Output figure C and save it as **"IS-429 Lab Week#11C Exploring Data NIMNAME".JPG/PNG/pdf".**

g. Save it on your program into "IS429 Lab Week#11_Exploring Data Files.sas"

h. The output of the above program will produce the following analysis results:

1) The neighbourhood_group per zipcode using the FREQ procedure with the result that 3 missing frequencies were found

2) The neighbourhood_group per room_type using the FREQ procedure with the result that 3 missing frequencies were found

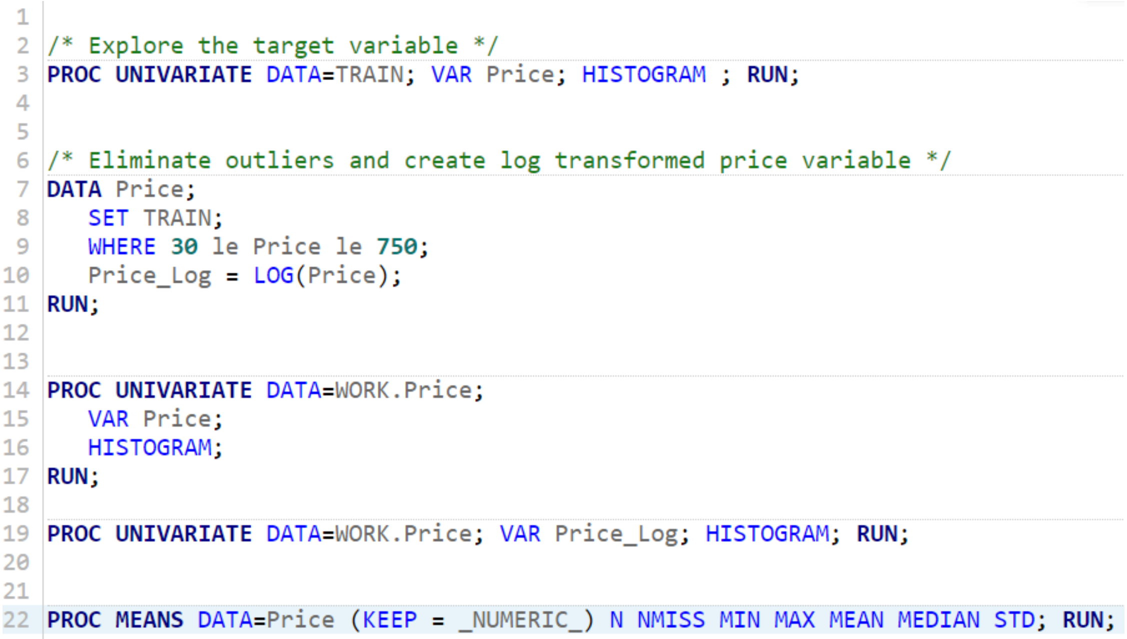
3) The output of the PROC SURVEYSELECT procedure provides you with a summary statement of the chosen parameters. The Sample Size column shows the number of observations that were randomly selected by the algorithm. A quick confirmation shows that 9812/49056 = 20%.

   You can check the log to confirm that the TRAIN and TEST data sets are the expected sizes and that there are no errors in the processing.

```
NOTE: There were 49056 observations read from the data set WORK.FULL.
NOTE: The data set WORK.TRAIN has 39244 observations and 54 variables.
NOTE: The data set WORK.TEST has 9812 observations and 54 variables.
```

4. Explore the target variable

- Type it on below codes and run:

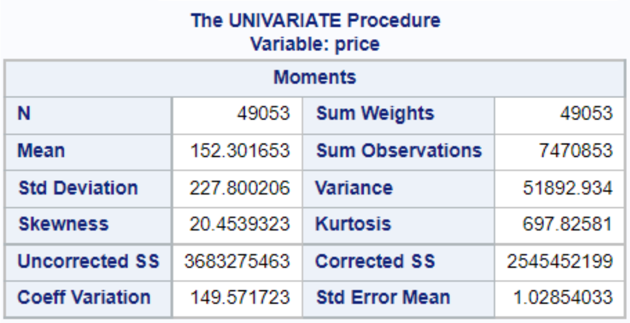


```

/* Explore the target variable */
PROC UNIVARIATE DATA=TRAIN; VAR Price; HISTOGRAM ; RUN;


/* Eliminate outliers and create log transformed price variable */
DATA Price;
   SET TRAIN;
   WHERE 30 le Price le 750;
   Price_Log = LOG(Price);
RUN;


PROC UNIVARIATE DATA=WORK.Price;
   VAR Price;
   HISTOGRAM;
RUN;

PROC UNIVARIATE DATA=WORK.Price; VAR Price_Log; HISTOGRAM; RUN;


PROC MEANS DATA=Price (KEEP = _NUMERIC_) N NMISS MIN MAX MEAN MEDIAN STD; RUN;
```

- Let's start by focusing on the cross-sectional data in the TRAIN data set. We are attempting to predict the per diem rate of Airbnb listings in New York City as of 12/06/2018.
- The data set contains a few different price variables for us to choose from:
  - ● Price
  - ● Weekly price
  - ● Monthly price
  - ● Security deposit
  - ● Cleaning fee
  - ●Extra guest fee
- The Price variable looks like the right one to use for our model. The other variables relating to cost are important pieces of information that we might use later, but for now, we will investigate our selected target variable.
- We have specified that we want to look at the Price variable. We have also requested that the output produce a histogram of the Price variable.

a. The distribution of Price using UNIVARIATE procedure

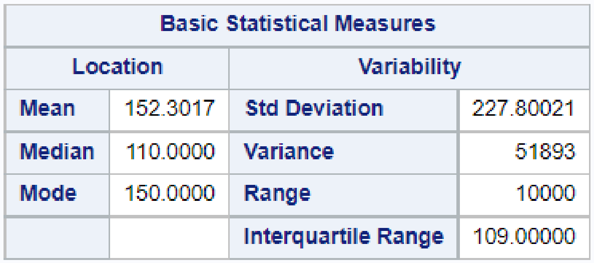
- The output of the PROC UNIVARIATE procedure contains a lot of information, so let's look at it one section at a time.
- The first section of Output shows that the data contained 49,053 observations and that the average value for price is $152.30.



The UNIVARIATE Procedure
Variable: price

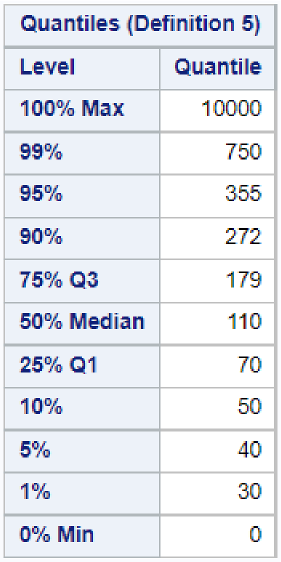
Moments			
N	49053	Sum Weights	49053
Mean	152.301653	Sum Observations	7470853
Std Deviation	227.800206	Variance	51892.934
Skewness	20.4539323	Kurtosis	697.82581
Uncorrected SS	3683275463	Corrected SS	2545452199
Coeff Variation	149.571723	Std Error Mean	1.02854033

- The next section of the PROC UNIVARIATE output shows the basic statistical measures.
- This section contains the average value of the Price variable (mean, median, and mode) along with measures of variability (std dev, variance, range, and interquartile range).

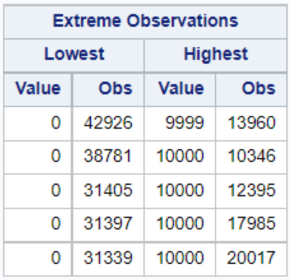


Basic Statistical Measures			
Location		Variability	
Mean	152.3017	Std Deviation	227.80021
Median	110.0000	Variance	51893
Mode	150.0000	Range	10000
		Interquartile Range	109.00000

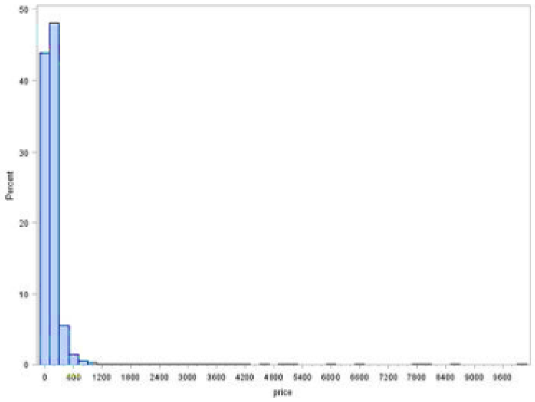
- There is a significant difference between the mean and median values. This difference is further evidence that outlier values are affecting the mean value.
- The range metric shows us the difference between the highest and lowest value. There is a $10,000 per diem range for price. Wow, that high end must be an awesome place to live.
- The final section of the PROC UNIVARIATE output below contains more information about the details of the variable distribution, along with outliers and missing values.

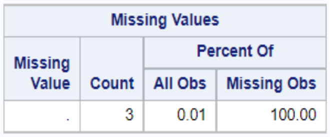


Quantiles (Definition 5)	
Level	Quantile
100% Max	10000
99%	750
95%	355
90%	272
75% Q3	179
50% Median	110
25% Q1	70
10%	50
5%	40
1%	30
0% Min	0



Extreme Observations			
Lowest		Highest	
Value	Obs	Value	Obs
0	42926	9999	13960
0	38781	10000	10346
0	31405	10000	12395
0	31397	10000	17985
0	31339	10000	20017





Missing Values			
Missing Value	Count	Percent Of	
		All Obs	Missing Obs
.	3	0.01	100.00

- The first section of the above Output shows the quantiles of the variable.
- This contains the minimum and maximum values as well as values at certain percentage cutoff points. The maximum value for the Price variable is $10,000 while the minimum is $0.
- At this point, we need to ask ourselves a question. Does it make any sense that someone would charge $0 per night to stay at their house?
- Maybe they are very lonely and just want some company. It is probably just the result of bad data entry.
- We also see that in the missing values section of the output, there are two observations with missing price data. This is actually good news. Usually, user-entered data is messier than this.
- These statistics show that less than 1% of the data would need to be either adjusted or deleted. We will make those decisions in the near future.
- On the high end of the distributions, we see that the 99% cutoff value is $750. There is a massive difference between the 99% and 100% values. That top 1% of observations are extreme outlier data points.
- The final piece of output that we can examine is the histogram of the Price variable. The distribution chart gives a visual demonstration of the statistics that we just reviewed.
- The histogram Output above shows a highly skewed distribution with a very long right-sided tail. The story that this chart tells us matches perfectly with the skewness value of 20.25 that we see in the statistics table above.
- All of this information tells us that the Price variable has significant outliers on the high end that we need to fix before we can do any modeling.
- So, we have investigated the target variable and determined that values with missing or $0 entries do not make sense. We have also investigated the high-end values, and we have determined that a cutoff point of $750 would eliminate extreme outliers. Let's code it up…

b.Save it on your program into "IS429 Lab Week#11_Explore the target variable"

Investigate the target variable:

- The data provide of New York City Inside Airbnb data sets consisting of Listings and Calendar, have placed them in "/home/u59229493/L00552 shared-drive/u59229493/Week#11/" repository or you can use as previous instructions to get in from your local laptop/ PC.
- These are the two main data sets that we will use for our analysis.

a. Eliminate outliers and create log transformed price variable

- The log shows us that the adjusted data set contains 38,527 observations. The original data set contained 39,244; therefore, we have lost only 717 observations.

```
NOTE: There were 38527 observations read from the data set WORK.TRAIN.
      WHERE (Price>=30 and Price<=750);
NOTE: The data set WORK.PRICE has 38527 observations and 55 variables
```

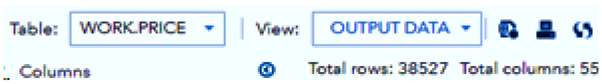
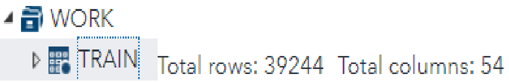


Table: WORK.PRICE View: OUTPUT DATA

Columns Total rows: 38527 Total columns: 55



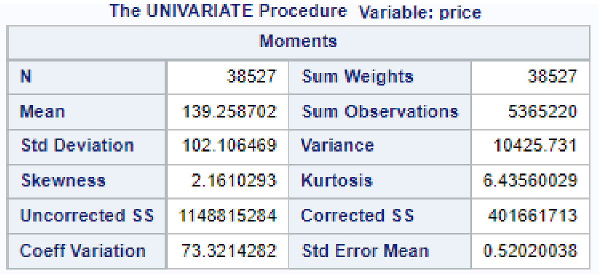
WORK

TRAIN Total rows: 39244 Total columns: 54

2.The majority of these excluded observations were for properties that have a $0 per diem rate.

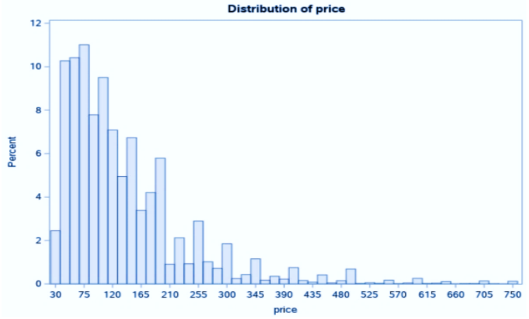
- Although our target variable is much cleaner now that we have established our upper and lower boundaries, we expect that the distribution of the data will still be skewed to the right.

- A PROC UNIVARIATE procedure developed on the Price variable shows a right tail skewness with a value of 2.16



The UNIVARIATE Procedure Variable: price

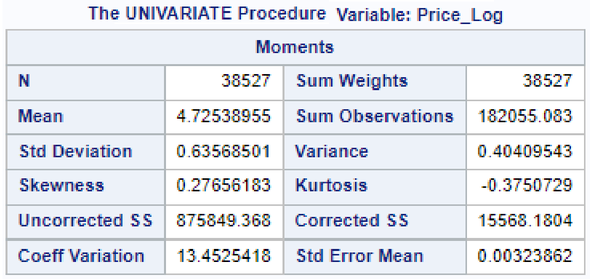
Moments			
N	38527	Sum Weights	38527
Mean	139.258702	Sum Observations	5365220
Std Deviation	102.106469	Variance	10425.731
Skewness	2.1610293	Kurtosis	6.43560029
Uncorrected SS	1148815284	Corrected SS	401661713
Coeff Variation	73.3214282	Std Error Mean	0.52020038



- In anticipation of this skewness, We created a log-transformed version of the Price variable in the previous DATA step:

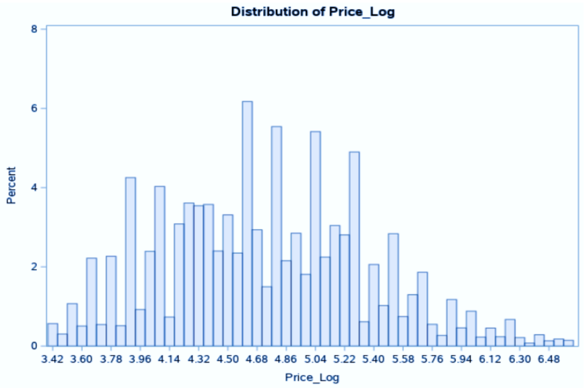
```
PROC MEANS DATA=Price (KEEP = _NUMERIC_) N NMISS MIN MAX MEAN MEDIAN STD;
RUN;
```

- This approach will transform the Price variable into a normally distributed variable. A PROC UNIVARIATE developed on the log-transformed Price variable shows a skewness value of 0.26



The UNIVARIATE Procedure Variable: Price_Log

Moments			
N	38527	Sum Weights	38527
Mean	4.72538955	Sum Observations	182055.083
Std Deviation	0.63568501	Variance	0.40409543
Skewness	0.27656183	Kurtosis	-0.3750729
Uncorrected SS	875849.368	Corrected SS	15568.1804
Coeff Variation	13.4525418	Std Error Mean	0.00323862

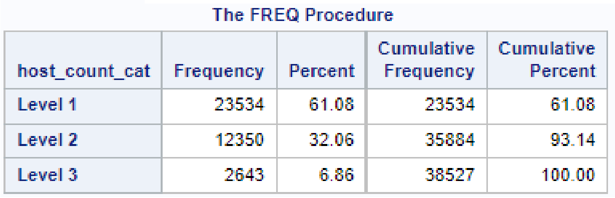


- Great! We now have a data set where we have eliminated extreme outliers and nonsensical values.
- We have also transformed our target variable into a normally distributed variable.
- Save it on your program into "IS429 Lab Week#11_Explore the target variable.sas"
- **Screenshotted your Price table and Price_log distribution chart as show on Output figure D.**
- **Save it as "IS-429 Lab Week#11D Target Variable NIM Name.JPG/PNG/pdf"**

5. Predictor Variable Analysis

- Due to the goal of the Airbnb project, We want to develop a tool that property owners could use to give them the optimal per diem price for their property on Airbnb.
- The predictor variables will be the values that property owners enter to describe the features of their property. You have the option of including additional features based on regional factors (tourist attractions, crime rate, events, and so on), but for now, we will focus on the data that we have already gathered.
- Under this scenario, we could make some assumptions:
  - ➲ We can expect that the traditional home features will be significant in the model:
    - ◦ Location
    - ◦ Property type
    - ◦ Number of people the property will accommodate
    - ◦ Number of beds, baths, rooms, and so on
  - ➲ Some data will not be available at the point of application:
    - ◦ User review scores
    - ◦ Review volume

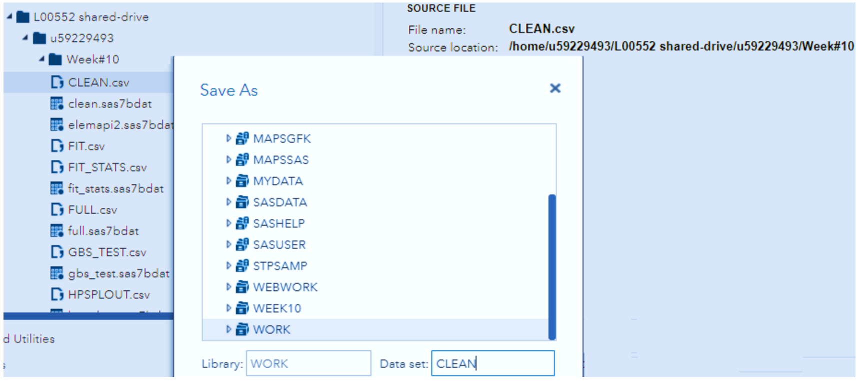
- Further analysis of this variable shows that there are differences between property owners who have a single property compared to those that have between two and ten properties and those who are major property owners who have more than ten properties.
- We have created by Clean table, a categorical variable for these levels and discarded the original numeric variable.



The FREQ Procedure

host_count_cat	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Level 1	23534	61.08	23534	61.08
Level 2	12350	32.06	35884	93.14
Level 3	2643	6.86	38527	100.00

a. Import Clean.csv from "/home/u59229493/L00552 shared-drive/u59229493/Week#11/clean.csv" or you can use as previous instructions to get in from your local laptop/ PC to your WORK space repository.



SOURCE FILE

File name: CLEAN.csv

Source location: /home/u59229493/L00552 shared-drive/u59229493/Week#10

Library: WORK Data set: CLEAN

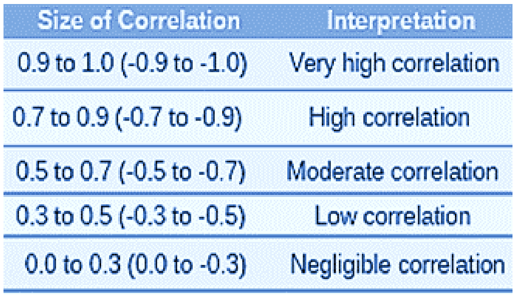
b. Type it on these codes below and run them all:

```
/*Import Clean.csv from /home/u59229493/L00552 shared-drive/u59229493/Week#10/clean.csv to your WORK repository*/
/* proc univariate data=clean; var bedrooms bathrooms beds; run; */
PROC FREQ DATA=Clean; TABLES host_count_cat; RUN;

PROC MEANS DATA=Clean;
   CLASS host_count_cat;
   VAR Price Price_Log;
RUN;

/* Create global numeric variables */
PROC CONTENTS NOPRINT DATA=Clean (KEEP=_NUMERIC_ DROP=id host_id
latitude longitude Price Price_Log) OUT=var1 (KEEP=name);
RUN;

PROC SQL NOPRINT;
   SELECT name INTO:varx separated by " " FROM var1;
QUIT;

%PUT &varx;

/* Create correlation analysis */
PROC CORR DATA=Clean;
   VAR &varx.;
RUN;
```

6. Collinearity Analysis

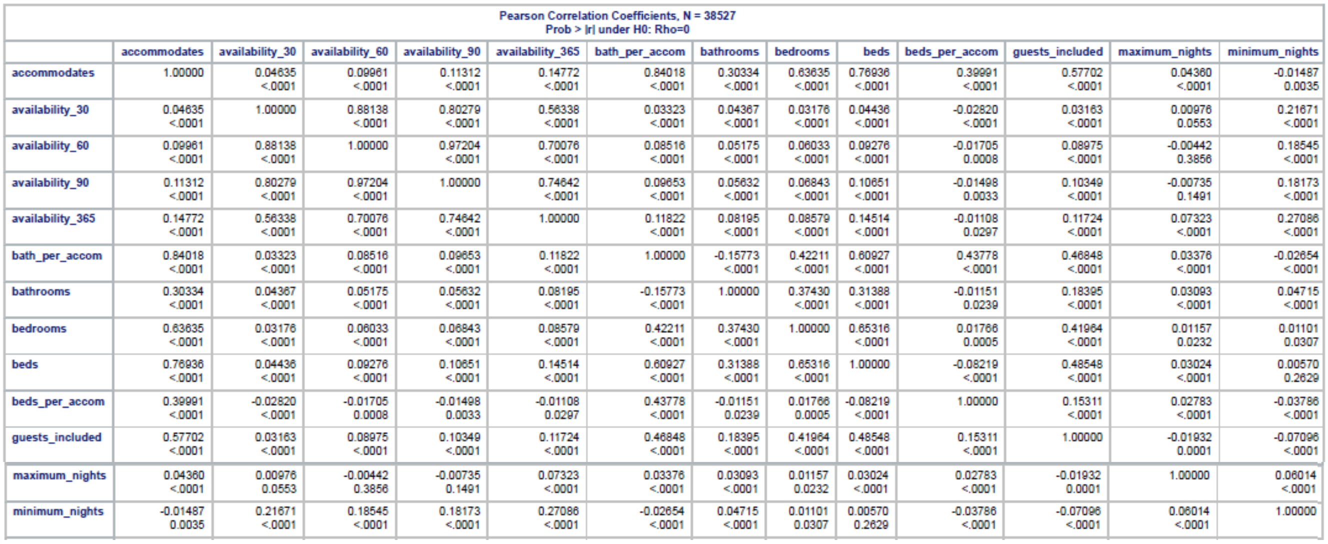
- One of the founding assumptions of regression models is that the predictor variables need to be independent (not correlated with one another).
- This assumption is important because if two or more variables are closely related, it is difficult to separate the individual effects of those variables on the response variable. In the end, collinearity reduces the accuracy of the estimates of the regression coefficients because it inflates the standard error of those coefficients.

- The table below provides a good rule of thumb for interpreting correlation coefficients. Any correlation coefficient above 0.7 or below -0.7 is considered highly correlated.



Size of Correlation	Interpretation
0.9 to 1.0 (-0.9 to -1.0)	Very high correlation
0.7 to 0.9 (-0.7 to -0.9)	High correlation
0.5 to 0.7 (-0.5 to -0.7)	Moderate correlation
0.3 to 0.5 (-0.3 to -0.5)	Low correlation
0.0 to 0.3 (0.0 to -0.3)	Negligible correlation

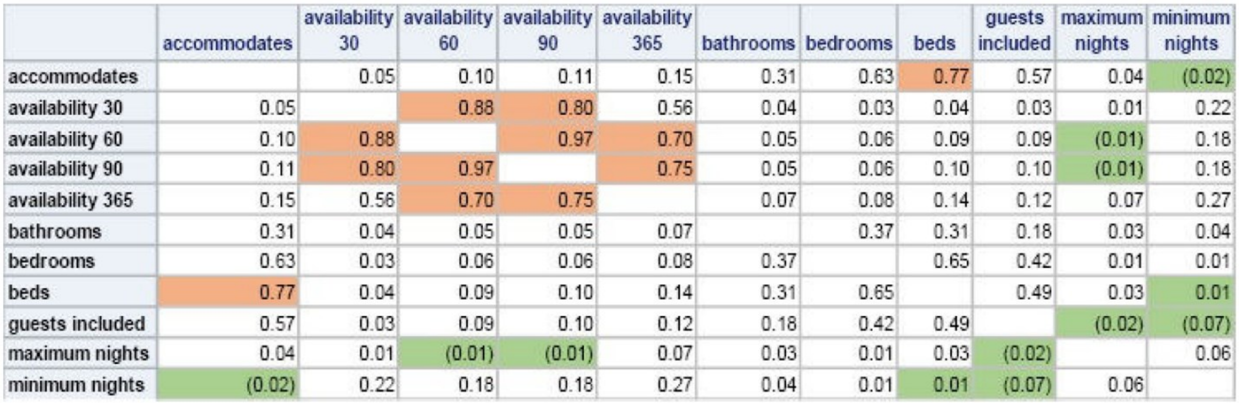
- SAS provides two great ways to identify multicollinearity through the CORR and REG procedures.
- PROC CORR creates a correlation matrix that contains the correlation coefficient for each variable combination.



Pearson Correlation Coefficients, N = 38527
Prob > |r| under H0: Rho=0

	accommodates	availability_30	availability_60	availability_90	availability_365	bath_per_accom	bathrooms	bedrooms	beds	beds_per_accom	guests_included	maximum_nights	minimum_nights
accommodates	1.00000	0.04635 <.0001	0.09961 <.0001	0.11312 <.0001	0.14772 <.0001	0.84018 <.0001	0.30334 <.0001	0.63635 <.0001	0.76936 <.0001	0.39991 <.0001	0.57702 <.0001	0.04360 <.0001	-0.01487 0.0035
availability_30	0.04635 <.0001	1.00000	0.88138 <.0001	0.80279 <.0001	0.56338 <.0001	0.03323 <.0001	0.04367 <.0001	0.03176 <.0001	0.04436 <.0001	-0.02820 <.0001	0.03163 <.0001	0.00976 0.0553	0.21671 <.0001
availability_60	0.09961 <.0001	0.88138 <.0001	1.00000	0.97204 <.0001	0.70076 <.0001	0.08516 <.0001	0.05175 <.0001	0.06033 <.0001	0.09276 <.0001	-0.01705 0.0008	0.08975 <.0001	-0.00442 0.3856	0.18545 <.0001
availability_90	0.11312 <.0001	0.80279 <.0001	0.97204 <.0001	1.00000	0.74642 <.0001	0.09653 <.0001	0.05632 <.0001	0.06843 <.0001	0.10651 <.0001	-0.01498 0.0033	0.10349 <.0001	-0.00735 0.1491	0.18173 <.0001
availability_365	0.14772 <.0001	0.56338 <.0001	0.70076 <.0001	0.74642 <.0001	1.00000	0.11822 <.0001	0.08195 <.0001	0.08579 <.0001	0.14514 <.0001	-0.01108 0.0297	0.11724 <.0001	0.07323 <.0001	0.27086 <.0001
bath_per_accom	0.84018 <.0001	0.03323 <.0001	0.08516 <.0001	0.09653 <.0001	0.11822 <.0001	1.00000	-0.15773 <.0001	0.42211 <.0001	0.60927 <.0001	0.43778 <.0001	0.46848 <.0001	0.03376 <.0001	-0.02654 <.0001
bathrooms	0.30334 <.0001	0.04367 <.0001	0.05175 <.0001	0.05632 <.0001	0.08195 <.0001	-0.15773 <.0001	1.00000	0.37430 <.0001	0.31388 <.0001	-0.01151 0.0239	0.18395 <.0001	0.03093 <.0001	0.04715 <.0001
bedrooms	0.63635 <.0001	0.03176 <.0001	0.06033 <.0001	0.06843 <.0001	0.08579 <.0001	0.42211 <.0001	0.37430 <.0001	1.00000	0.65316 <.0001	0.01766 0.0005	0.41964 <.0001	0.01157 0.0232	0.01101 0.0307
beds	0.76936 <.0001	0.04436 <.0001	0.09276 <.0001	0.10651 <.0001	0.14514 <.0001	0.60927 <.0001	0.31388 <.0001	0.65316 <.0001	1.00000	-0.08219 <.0001	0.48548 <.0001	0.03024 <.0001	0.00570 0.2629
beds_per_accom	0.39991 <.0001	-0.02820 <.0001	-0.01705 0.0008	-0.01498 0.0033	-0.01108 0.0297	0.43778 <.0001	-0.01151 0.0239	0.01766 0.0005	-0.08219 <.0001	1.00000	0.15311 <.0001	0.02783 <.0001	-0.03786 <.0001
guests_included	0.57702 <.0001	0.03163 <.0001	0.08975 <.0001	0.10349 <.0001	0.11724 <.0001	0.46848 <.0001	0.18395 <.0001	0.41964 <.0001	0.48548 <.0001	0.15311 <.0001	1.00000	-0.01932 0.0001	-0.07096 <.0001
maximum_nights	0.04360 <.0001	0.00976 0.0553	-0.00442 0.3856	-0.00735 0.1491	0.07323 <.0001	0.03376 <.0001	0.03093 <.0001	0.01157 0.0232	0.03024 <.0001	0.02783 <.0001	-0.01932 0.0001	1.00000	0.06014 <.0001
minimum_nights	-0.01487 0.0035	0.21671 <.0001	0.18545 <.0001	0.18173 <.0001	0.27086 <.0001	-0.02654 <.0001	0.04715 <.0001	0.01101 0.0307	0.00570 0.2629	-0.03786 <.0001	-0.07096 <.0001	0.06014 <.0001	1.00000

- The default correlation statistic is the Pearson's r. The results of the procedure are below. We have color-coded the top 10% of high and low correlation values.



	accommodates	availability 30	availability 60	availability 90	availability 365	bathrooms	bedrooms	beds	guests included	maximum nights	minimum nights
accommodates		0.05	0.10	0.11	0.15	0.31	0.63	0.77	0.57	0.04	(0.02)
availability 30	0.05		0.88	0.80	0.56	0.04	0.03	0.04	0.03	0.01	0.22
availability 60	0.10	0.88		0.97	0.70	0.05	0.06	0.09	0.09	(0.01)	0.18
availability 90	0.11	0.80	0.97		0.75	0.05	0.06	0.10	0.10	(0.01)	0.18
availability 365	0.15	0.56	0.70	0.75		0.07	0.08	0.14	0.12	0.07	0.27
bathrooms	0.31	0.04	0.05	0.05	0.07		0.37	0.31	0.18	0.03	0.04
bedrooms	0.63	0.03	0.06	0.06	0.08	0.37		0.65	0.42	0.01	0.01
beds	0.77	0.04	0.09	0.10	0.14	0.31	0.65		0.49	0.03	0.01
guests included	0.57	0.03	0.09	0.10	0.12	0.18	0.42	0.49		(0.02)	(0.07)
maximum nights	0.04	0.01	(0.01)	(0.01)	0.07	0.03	0.01	0.03	(0.02)		0.06
minimum nights	(0.02)	0.22	0.18	0.18	0.27	0.04	0.01	0.01	(0.07)	0.06	

- Variables are considered highly correlated if the correlation coefficient is above 0.7 or below -0.7.
- The bottom 10% of correlation values are highlighted in green. We can see that negative correlation is not a problem since the strongest negative correlation is -0.07.

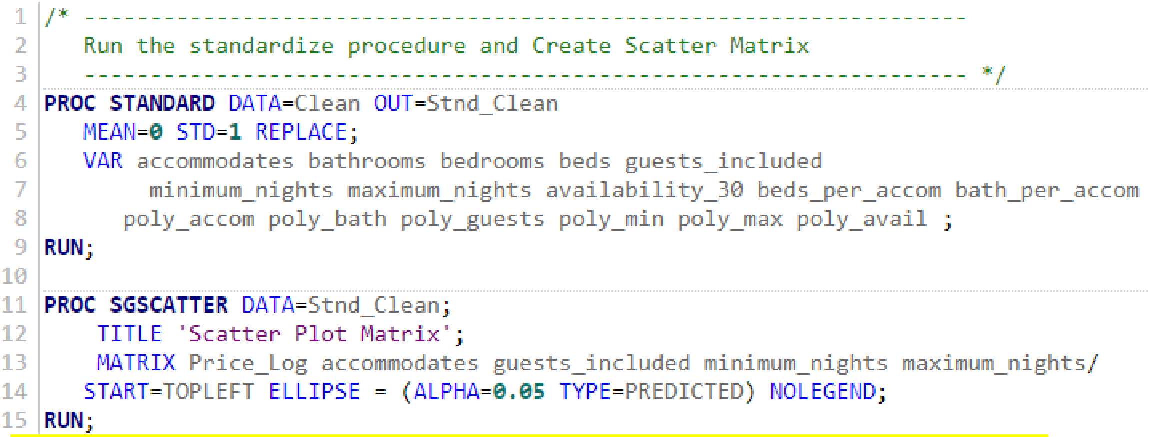
- For the positively correlated variables, we see that there are several cases where the correlation coefficient is above 0.7. It makes sense that the availability variables are highly correlated. If a property is available within 30 days, it would also be available within the next 60, 90, and 365 days. Also, the correlation between accommodates and beds makes sense. If a property has a high number of beds, it can accommodate more people.
- Save it on your program into "IS429 Lab Week#11_Predictor Variable Analysis.sas"

c. **Print your results as shown in Output figure E into "IS-429 Lab Week#11E Predictor Variable AnalysisNIMNAME".pdf**

b. Scatter Matrix

- An additional method of analyzing our numeric variables is to produce scatter plots of each variable's relationship with the target variable. SAS provides a great visual demonstration of these relationships with the scatter matrix graph.
- This graph contains a series of scatter plots and allows the researcher to easily see the relationships that each of the numeric predictor variables has with the target variable. We can also see the relationship that the numeric predictors have with one another.
- In the Output Figure E, we can already see the strong positive relationship between the price and accommodates variables, as well as between price and guests included.

a. Still on SAS®Studio, click options and click New SAS program (F4), type in below codes and run it.



```
/* ------------------------------------------------------------------
   Run the standardize procedure and Create Scatter Matrix
   ------------------------------------------------------------------ */
PROC STANDARD DATA=Clean OUT=Stnd_Clean
   MEAN=0 STD=1 REPLACE;
   VAR accommodates bathrooms bedrooms beds guests_included
        minimum_nights maximum_nights availability_30 beds_per_accom bath_per_accom
      poly_accom poly_bath poly_guests poly_min poly_max poly_avail ;
RUN;

PROC SGSCATTER DATA=Stnd_Clean;
    TITLE 'Scatter Plot Matrix';
    MATRIX Price_Log accommodates guests_included minimum_nights maximum_nights/
   START=TOPLEFT ELLIPSE = (ALPHA=0.05 TYPE=PREDICTED) NOLEGEND;
RUN;
```

b. **Save your Scatter Matrix as "IS-429 Lab Week#11F ScatterMatrix NIMNAME.pdf".**

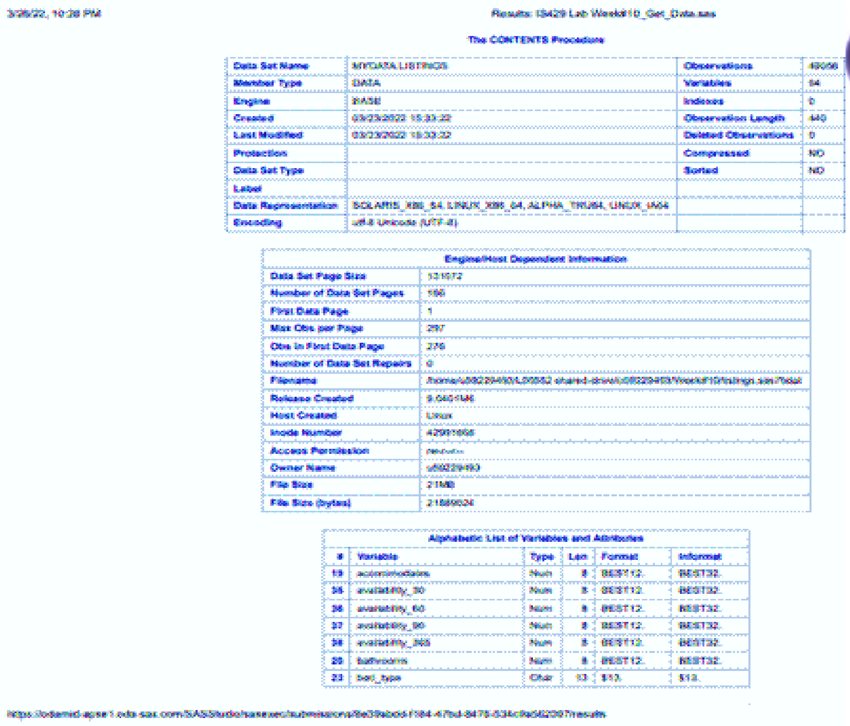
c. Save your program into "IS429 Lab Week#11_Scatter Matrix.sas".

➲ Finally, today's practicum is over..

➲ **Zipped your screenshots and pdf files and submit immediately today to e-Learning IS-429 Practicum Week#11 with the naming format IS-429 BDA Lab Week#11 NIM yourName.zip**.

RESULTS/ OUTPUT

A. Get the Data



3/28/22, 10:28 PM

Results: IS429 Lab Week#10_Get_Data.sas

The CONTENTS Procedure

Data Set Name	MYDATA.LISTINGS	Observations	49056
Member Type	DATA	Variables	54
Engine	BASE	Indexes	0
Created	03/23/2022 15:33:22	Observation Length	440
Last Modified	03/23/2022 15:33:22	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	SOLARIS_X86_64, LINUX_X86_64, ALPHA_TRU64, LINUX_IA64		
Encoding	utf-8 Unicode (UTF-8)		

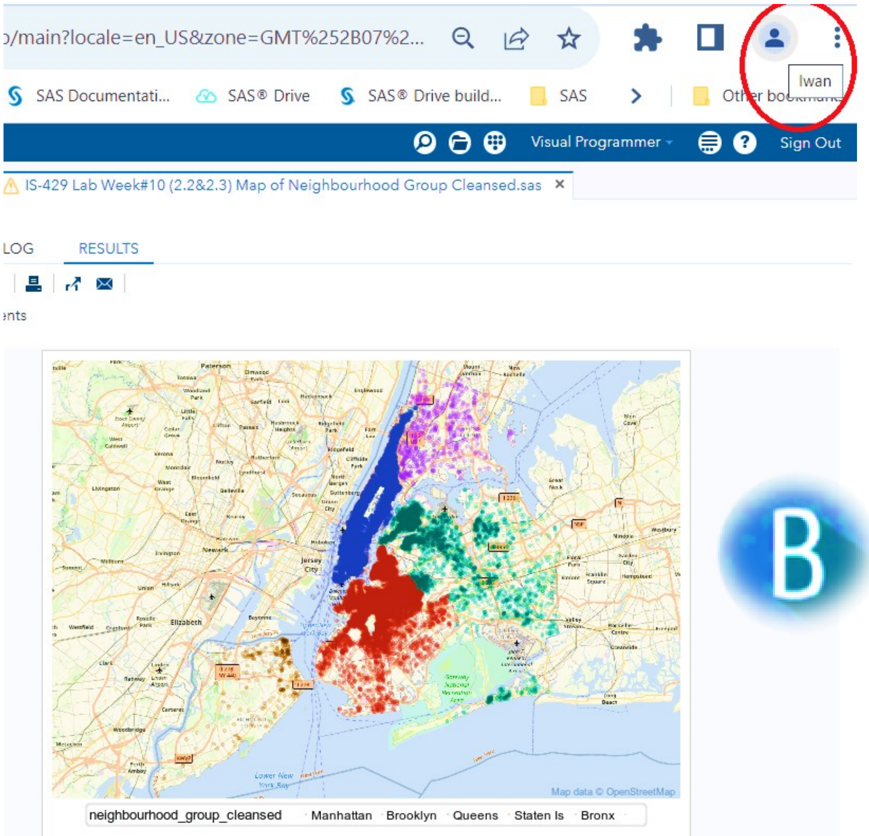
Engine/Host Dependent Information	
Data Set Page Size	131072
Number of Data Set Pages	166
First Data Page	1
Max Obs per Page	297
Obs in First Data Page	276
Number of Data Set Repairs	0
Filename	/home/u58229493/LS0652 shared-drive/u58229493/Week#10/listings.sas7bdat
Release Created	9.0401M6
Host Created	Linux
Inode Number	42991668
Access Permission	rw-r--r--
Owner Name	u58229493
File Size	21MB
File Size (bytes)	21889024

Alphabetic List of Variables and Attributes

#	Variable	Type	Len	Format	Informat
19	accommodates	Num	8	BEST12.	BEST32.
35	availability_30	Num	8	BEST12.	BEST32.
36	availability_60	Num	8	BEST12.	BEST32.
37	availability_90	Num	8	BEST12.	BEST32.
38	availability_365	Num	8	BEST12.	BEST32.
20	bathrooms	Num	8	BEST12.	BEST32.
23	bed_type	Char	13	$13.	$13.

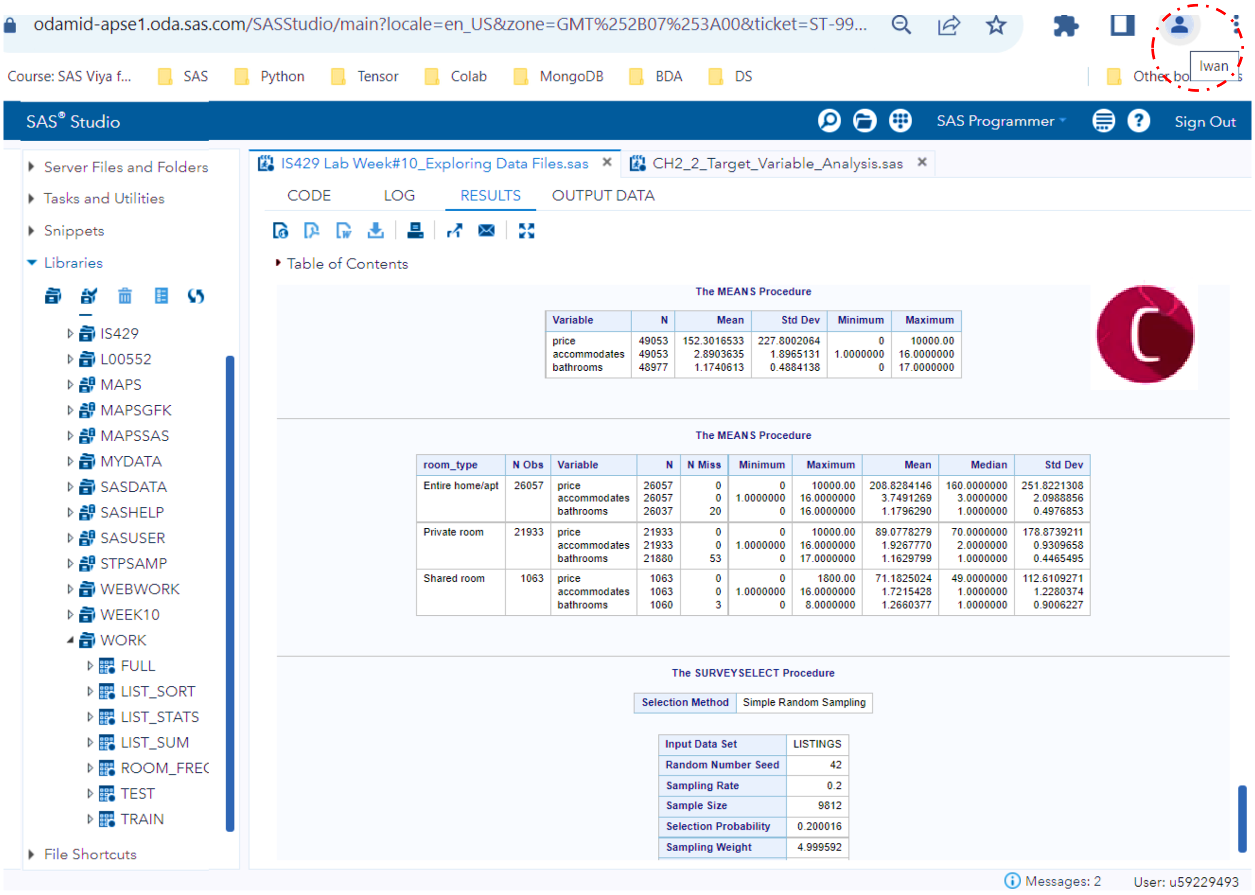
https://odamid-apse1.oda.sas.com/SASStudio/sasexec/submissions/8e39eb06-f384-47bd-8476-534c8a5d6230/results

B. Create Geographic Map



IS-429 Lab Week#10 (2.2&2.3) Map of Neighbourhood Group Cleansed.sas

C. Exploring data files.



The MEANS Procedure

Variable	N	Mean	Std Dev	Minimum	Maximum
price	49053	152.3016533	227.8002064	0	10000.00
accommodates	49053	2.8903635	1.8965131	1.0000000	16.0000000
bathrooms	48977	1.1740613	0.4884138	0	17.0000000

The MEANS Procedure

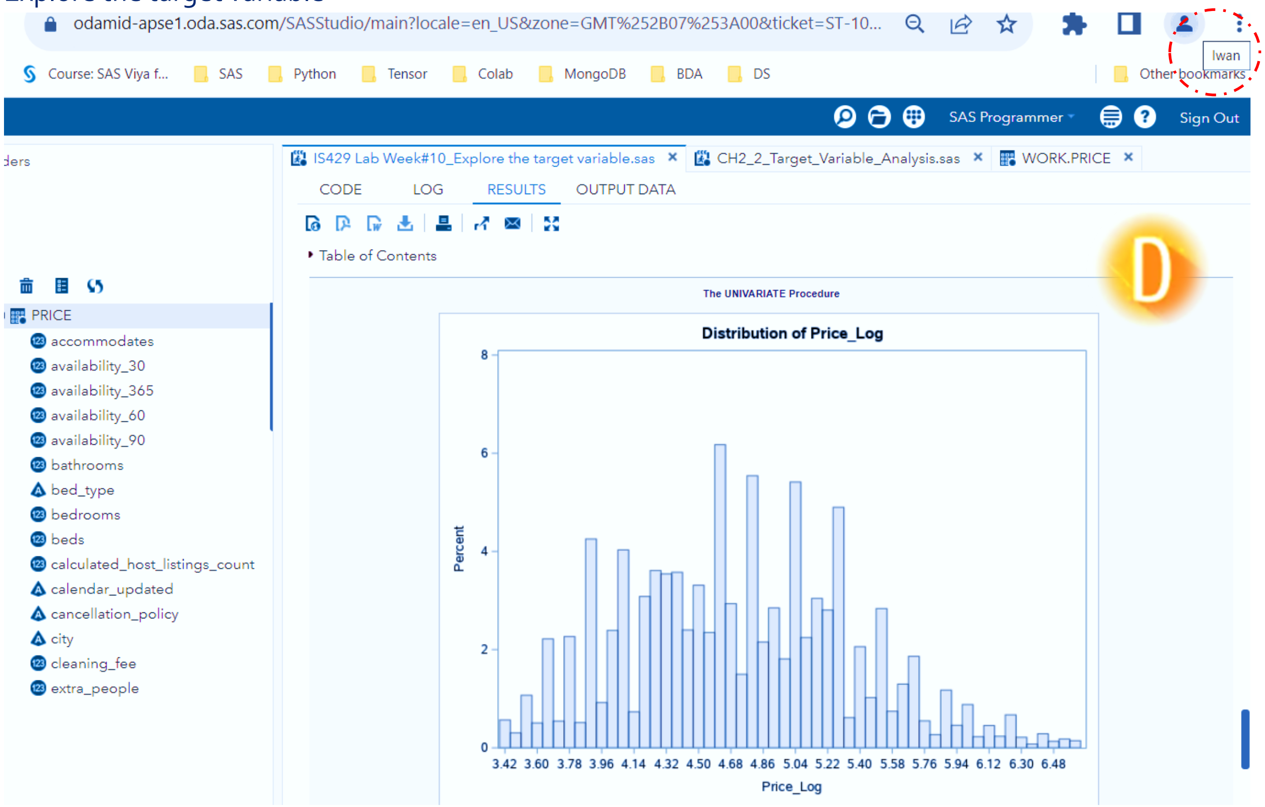
room_type	N Obs	Variable	N	N Miss	Minimum	Maximum	Mean	Median	Std Dev
Entire home/apt	26057	price	26057	0	0	10000.00	208.8284146	160.0000000	251.8221308
		accommodates	26057	0	1.0000000	16.0000000	3.7491269	3.0000000	2.0988856
		bathrooms	26037	20	0	16.0000000	1.1796290	1.0000000	0.4976853
Private room	21933	price	21933	0	0	10000.00	89.0778279	70.0000000	178.8739211
		accommodates	21933	0	1.0000000	16.0000000	1.9267770	2.0000000	0.9309658
		bathrooms	21880	53	0	17.0000000	1.1629799	1.0000000	0.4465495
Shared room	1063	price	1063	0	0	1800.00	71.1825024	49.0000000	112.6109271
		accommodates	1063	0	1.0000000	16.0000000	1.7215428	1.0000000	1.2280374
		bathrooms	1060	3	0	8.0000000	1.2660377	1.0000000	0.9006227

The SURVEYSELECT Procedure

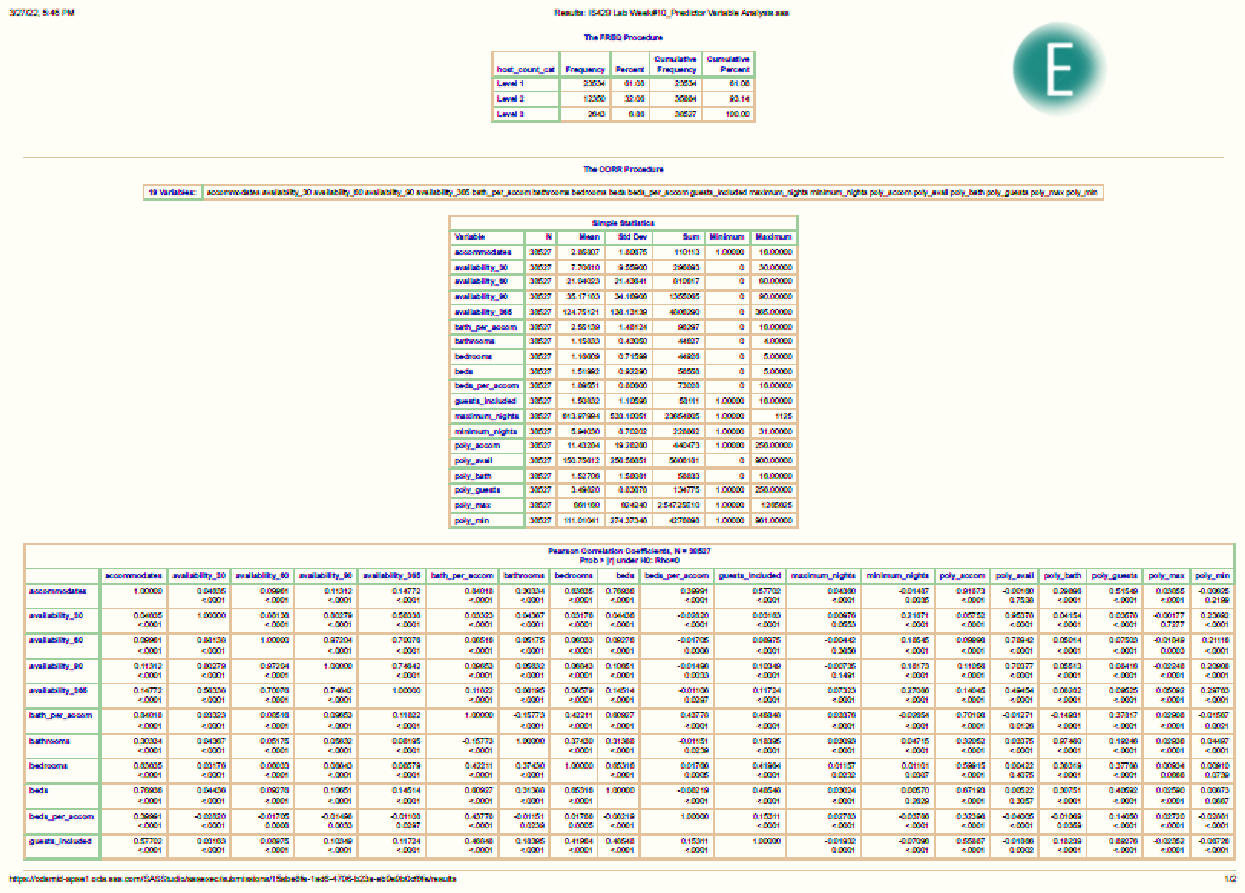
Selection Method	Simple Random Sampling

Input Data Set	LISTINGS
Random Number Seed	42
Sampling Rate	0.2
Sample Size	9812
Selection Probability	0.200016
Sampling Weight	4.999592

D. Explore the target variable



E. Predictor Variable Analysis



3/27/22, 5:45 PM

Results: IS429 Lab Week#10_Predictor Variable Analysis.sas

The FREQ Procedure

host_count_cat	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Level 1	23534	61.08	23534	61.08
Level 2	12350	32.06	35884	93.14
Level 3	2643	6.86	38527	100.00

The CORR Procedure

19 Variables:	accommodates availability_30 availability_60 availability_90 availability_365 bath_per_accom bathrooms bedrooms beds beds_per_accom guests_included maximum_nights minimum_nights poly_accom poly_avail poly_bath poly_guests poly_max poly_min

Simple Statistics

Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
accommodates	38527	2.85807	1.80675	110113	1.00000	16.00000
availability_30	38527	7.70610	9.55900	296893	0	30.00000
availability_60	38527	21.04023	21.43641	810617	0	60.00000
availability_90	38527	35.17103	34.10900	1355065	0	90.00000
availability_365	38527	124.75121	130.13139	4806290	0	365.00000
bath_per_accom	38527	2.55139	1.48124	98297	0	16.00000
bathrooms	38527	1.15933	0.43050	44667	0	4.00000
bedrooms	38527	1.16809	0.71589	44928	0	5.00000
beds	38527	1.51992	0.92290	58558	0	5.00000
beds_per_accom	38527	1.89551	0.80900	73028	0	16.00000
guests_included	38527	1.50832	1.10596	58111	1.00000	16.00000
maximum_nights	38527	613.87984	530.10051	23651805	1.00000	1125
minimum_nights	38527	5.94030	8.70232	228862	1.00000	31.00000
poly_accom	38527	11.43204	19.38280	440473	1.00000	256.00000
poly_avail	38527	150.75812	258.56851	5808131	0	900.00000
poly_bath	38527	1.52706	1.58081	58833	0	16.00000
poly_guests	38527	3.49820	8.83676	134775	1.00000	256.00000
poly_max	38527	661160	624240	2.5472358E10	1.00000	1265625
poly_min	38527	111.01041	274.37346	4276898	1.00000	961.00000

Pearson Correlation Coefficients, N = 38527
Prob > \|r\| under H0: Rho=0

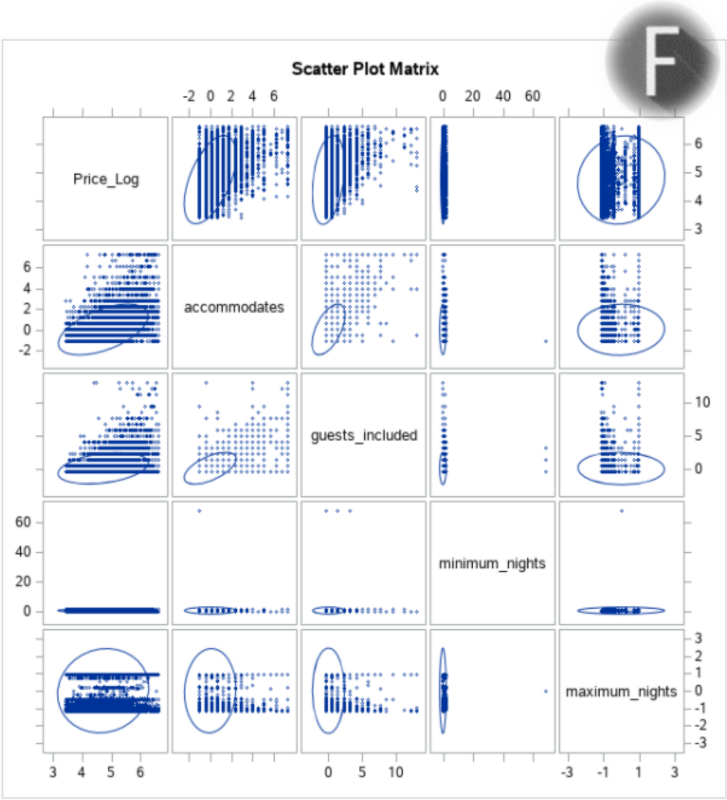
	accommodates	availability_30	availability_60	availability_90	availability_365	bath_per_accom	bathrooms	bedrooms	beds	beds_per_accom
accommodates	1.00000	0.04035 <.0001	0.09961 <.0001	0.11312 <.0001	0.14772 <.0001	0.84018 <.0001	0.30334 <.0001	0.83035 <.0001	0.76936 <.0001	0.39991 <.0001
availability_30	0.04035 <.0001	1.00000	0.88130 <.0001	0.80279 <.0001	0.58038 <.0001	0.03323 <.0001	0.04367 <.0001	0.03176 <.0001	0.04436 <.0001	-0.02820 <.0001
availability_60	0.09961 <.0001	0.88130 <.0001	1.00000	0.97204 <.0001	0.70078 <.0001	0.06516 <.0001	0.05175 <.0001	0.06033 <.0001	0.09278 <.0001	-0.01705 0.0008
availability_90	0.11312 <.0001	0.80279 <.0001	0.97204 <.0001	1.00000	0.74642 <.0001	0.09653 <.0001	0.05632 <.0001	0.06643 <.0001	0.10651 <.0001	-0.01496 0.0033
availability_365	0.14772 <.0001	0.58038 <.0001	0.70078 <.0001	0.74642 <.0001	1.00000	0.11622 <.0001	0.08195 <.0001	0.08579 <.0001	0.14514 <.0001	-0.01108 0.0297
bath_per_accom	0.84018 <.0001	0.03323 <.0001	0.06516 <.0001	0.09653 <.0001	0.11622 <.0001	1.00000	-0.15773 <.0001	0.42211 <.0001	0.80927 <.0001	0.43778 <.0001
bathrooms	0.30334 <.0001	0.04367 <.0001	0.05175 <.0001	0.05632 <.0001	0.08195 <.0001	-0.15773 <.0001	1.00000	0.37430 <.0001	0.31368 <.0001	-0.01151 0.0239
bedrooms	0.83035 <.0001	0.03176 <.0001	0.06033 <.0001	0.06643 <.0001	0.08579 <.0001	0.42211 <.0001	0.37430 <.0001	1.00000	0.85316 <.0001	0.01786 0.0005
beds	0.76936 <.0001	0.04436 <.0001	0.09278 <.0001	0.10651 <.0001	0.14514 <.0001	0.80927 <.0001	0.31368 <.0001	0.85316 <.0001	1.00000	-0.06219 <.0001
beds_per_accom	0.39991 <.0001	-0.02820 <.0001	-0.01705 0.0008	-0.01496 0.0033	-0.01108 0.0297	0.43778 <.0001	-0.01151 0.0239	0.01786 0.0005	-0.06219 <.0001	1.00000
guests_included	0.57702 <.0001	0.03163 <.0001	0.06975 <.0001	0.10349 <.0001	0.11724 <.0001	0.46646 <.0001	0.18395 <.0001	0.41964 <.0001	0.48548 <.0001	0.15311 <.0001

	guests_included	maximum_nights	minimum_nights	poly_accom	poly_avail	poly_bath	poly_guests	poly_max	poly_min
accommodates	0.57702 <.0001	0.04080 <.0001	-0.01467 0.0039	0.91873 <.0001	-0.00160 0.7536	0.39696 <.0001	0.51549 <.0001	0.03655 <.0001	-0.00625 0.2188
availability_30	0.03163 <.0001	0.00976 0.0553	0.21871 <.0001	0.05752 <.0001	0.95370 <.0001	0.04154 <.0001	0.03570 <.0001	-0.00177 0.7277	0.23692 <.0001
availability_60	0.06975 <.0001	-0.00442 0.3856	0.18545 <.0001	0.09996 <.0001	0.78942 <.0001	0.05014 <.0001	0.07503 <.0001	-0.01849 0.0003	0.21116 <.0001
availability_90	0.10349 <.0001	-0.00735 0.1491	0.18173 <.0001	0.11058 <.0001	0.70377 <.0001	0.05513 <.0001	0.06410 <.0001	-0.02248 <.0001	0.20906 <.0001
availability_365	0.11724 <.0001	0.07323 <.0001	0.27086 <.0001	0.14045 <.0001	0.49454 <.0001	0.08282 <.0001	0.09525 <.0001	0.05092 <.0001	0.29703 <.0001
bath_per_accom	0.46646 <.0001	0.03376 <.0001	-0.02854 <.0001	0.70106 <.0001	-0.01271 0.0126	-0.14801 <.0001	0.37817 <.0001	0.02906 <.0001	-0.01567 0.0021
bathrooms	0.18395 <.0001	0.03093 <.0001	0.04715 <.0001	0.32262 <.0001	0.03375 <.0001	0.87480 <.0001	0.18246 <.0001	0.02936 <.0001	0.04487 <.0001
bedrooms	0.41964 <.0001	0.01157 0.0232	0.01101 0.0307	0.59615 <.0001	0.00422 0.4075	0.36319 <.0001	0.37786 <.0001	0.00934 0.0666	0.00910 0.0739
beds	0.48548 <.0001	0.03024 <.0001	0.00570 0.2629	0.67193 <.0001	0.00522 0.3057	0.30751 <.0001	0.40592 <.0001	0.02590 <.0001	0.00873 0.0867
beds_per_accom	0.15311 <.0001	0.03780 <.0001	-0.03796 <.0001	0.32396 <.0001	-0.04005 <.0001	-0.01089 0.0325	0.14050 <.0001	0.02720 <.0001	-0.02901 <.0001
guests_included	1.00000	-0.01932 0.0001	-0.07095 <.0001	0.55667 <.0001	-0.01886 0.0002	0.18239 <.0001	0.89276 <.0001	-0.02353 <.0001	-0.06726 <.0001

https://odamid-apse1.oda.sas.com/SASStudio/sasexec/submissions/15ebe8fe-1ed6-47D6-b23e-eb0e0b0cf8fe/results

1/2

F. Scatter Matirx



3. SAS® Support | Documentation

The
End

REFERENCE

1. Gearhart, James. 2020. End-to-End Data Science with SAS®: A Hands-On Programming Guide. Cary, NC: SAS Institute Inc.
2. SAS Institute Inc. 2020. SAS® Viya® Programming: Getting Started. SAS Institute Inc. Cary, NC, USA.
3. SAS® Support | Documentation
4. Other additional references are excerpts from various Online Learning/websites.