Assignment 3 (11 January 2021)

Due Date: 14 January 2021

(Each question is worth 20 point; Total Points= 100)

Ensure that you submit the file as pdf along with the output or ipynb upload which can be executed.

Q1. Upload the PopulationEstimates.xls and save it as a dataframe and do the following:

A. Subset the dataframe or select all values in column "State" where the county or State is "TX".
B. Save the subset dataframe in variable named "dfTX".
C. Print the output subset dataframe.

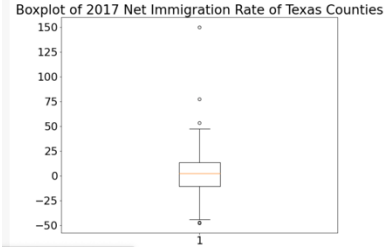
Your dfTX dataframe should look like this:

	FIPS	State	Area_Name	Rural-urban_Continuum Code_2003	Rural-urban_Continuum Code_2013	Urban_Influence_Code_2003	Urban_Influence_Code_2013	Economic_typology_2015	CENSUS
2567	48000	TX	Texas	NaN	NaN	NaN	NaN	NaN	
2568	48001	TX	Anderson County	5.0	7.0	8.0	8.0	4.0	
2569	48003	TX	Andrews County	6.0	6.0	5.0	5.0	2.0	
2570	48005	TX	Angelina County	5.0	5.0	8.0	8.0	0.0	
2571	48007	TX	Aransas County	2.0	2.0	2.0	2.0	5.0	
...	...	...	...	...	...	...	...	...	
2817	48499	TX	Wood County	6.0	6.0	6.0	6.0	0.0	
2818	48501	TX	Yoakum County	7.0	7.0	9.0	9.0	2.0	
2819	48503	TX	Young County	6.0	7.0	6.0	11.0	2.0	
2820	48505	TX	Zapata County	6.0	6.0	6.0	5.0	2.0	
2821	48507	TX	Zavala County	7.0	7.0	9.0	9.0	0.0	

255 rows × 149 columns

Q2. Create a Boxplot using the subset dataframe in Question 1 for Texas counties' net immigration rate of 2017 (column name 'R_NET_MIG_2017'). Add title "Boxplot of 2017 Net Immigration Rate of Texas Counties" to the graph and set the figure size to (10,8).(Matplotlib has to be used)

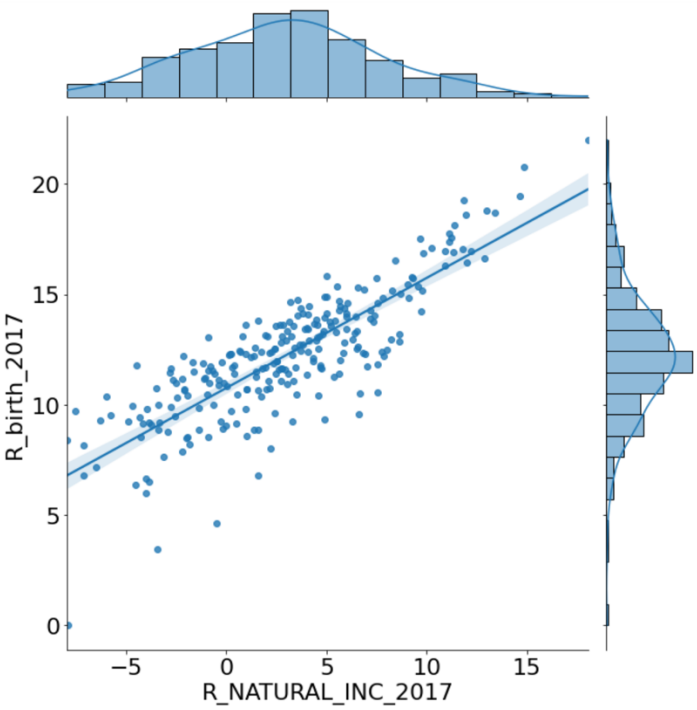
Ideal output:



Q3. Explain the distribution of the column titled 'R_NET_MIG_2017' from question 2. (Note that you are still using the subset dataframe; hint: use distfit(). Visualize the plot's distribution summary using plot_summary().

Q4. Using "dfTX" dataframe create a jointplot between "R_NATURAL_INC_2017" (in x axis) and "R_birth_2017" (in y axis). Set the kind parameter as "reg" and height=10. (Seaborn has to be used)

Ideal output:



Q5. Load the tips dataset using the below syntax.

```
import seaborn as sns
tips = sns.load_dataset("tips")
tips.head()
```

Create a box plot using seaborn for "day" in x-axis and "total_bill" in y-axis.

Ideal output: