DATA_PREPOROCESSING

April 21, 2021

[1]:
```
import pandas as pd
import seaborn as sns
import numpy as np

print("Libraries imported Sucessfully...!")
```

```
Libraries imported Sucessfully…!
```

[2]:
```
dataframes = pd.read_csv("middle_tn_schools.csv")
dataframes.head()
```

[2]:
```
                         name  school_rating  …  percent_asian
percent_hispanic
0  Allendale Elementary School            5.0  …            1.6
5.6
1          Anderson Elementary            2.0  …            1.0
4.9
2             Avoca Elementary            4.0  …            1.2
4.4
3                Bailey Middle            0.0  …            2.3
4.3
4          Barfield Elementary            4.0  …            7.1
6.0

[5 rows x 15 columns]
```

[3]:
```
dataframes.shape
```

[3]:
```
(347, 15)
```

[4]:
```
dataframes.dtypes
```

[4]:
```
name                   object
school_rating         float64
size                  float64
reduced_lunch         float64
state_percentile_16   float64
state_percentile_15   float64
```

```
stu_teach_ratio        float64
school_type             object
avg_score_15           float64
avg_score_16           float64
full_time_teachers     float64
percent_black          float64
percent_white          float64
percent_asian          float64
percent_hispanic       float64
dtype: object
```

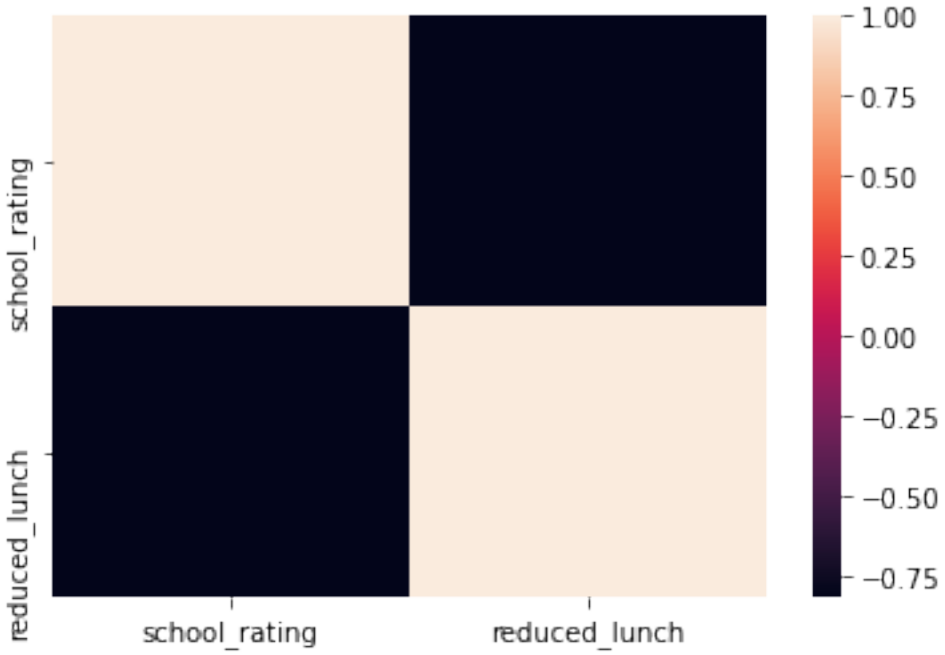
[5]:
```
dataframes.describe()
```

[5]:

	school_rating	size	…	percent_asian	percent_hispanic
count	347.000000	347.000000	…	347.000000	347.000000
mean	2.968300	699.472622	…	2.642651	11.164553
std	1.690377	400.598636	…	3.109629	12.030608
min	0.000000	53.000000	…	0.000000	0.000000
25%	2.000000	420.500000	…	0.750000	3.800000
50%	3.000000	595.000000	…	1.600000	6.400000
75%	4.000000	851.000000	…	3.100000	13.800000
max	5.000000	2314.000000	…	21.100000	65.200000

[8 rows x 13 columns]

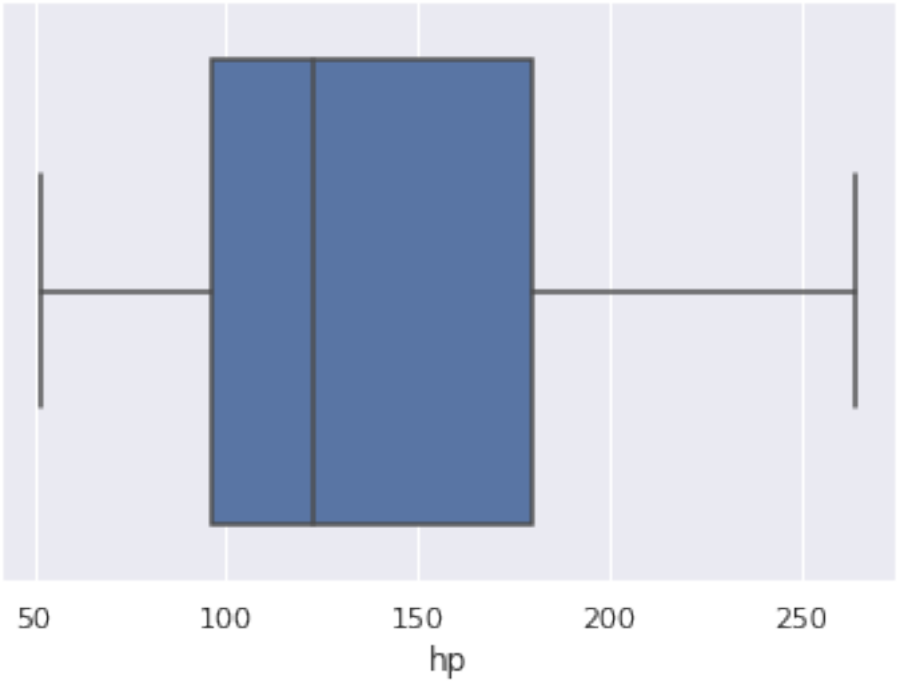
[8]:
```
sns.heatmap(dataframes[['school_rating','reduced_lunch']].corr());
```



[14]:
```
import pandas as pd
import seaborn as sns
from sklearn.impute import SimpleImputer
import numpy as np
import matplotlib as plt
%matplotlib inline

mtcars = pd.read_csv("mtcars.csv")

print("Files Imported Sucessfully...!")

description = mtcars.describe()
#print(description)

data_types = mtcars.dtypes
#print(data_types)

missing_values = mtcars.isna().any()
#print(missing_values)

#sns.boxplot(mtcars['hp'])

filt = mtcars["hp"].values<300

mtcars_filt = mtcars[filt]

sns.boxplot(mtcars_filt['hp'])

#
```

```
Files Imported Sucessfully…!
```

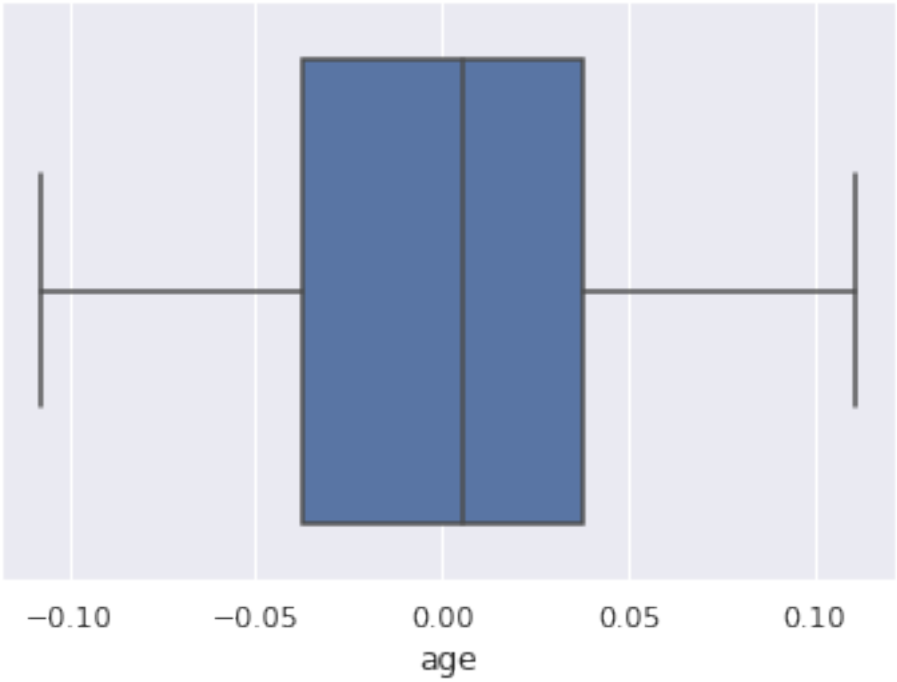
[14]:
```
<AxesSubplot:xlabel='hp'>
```

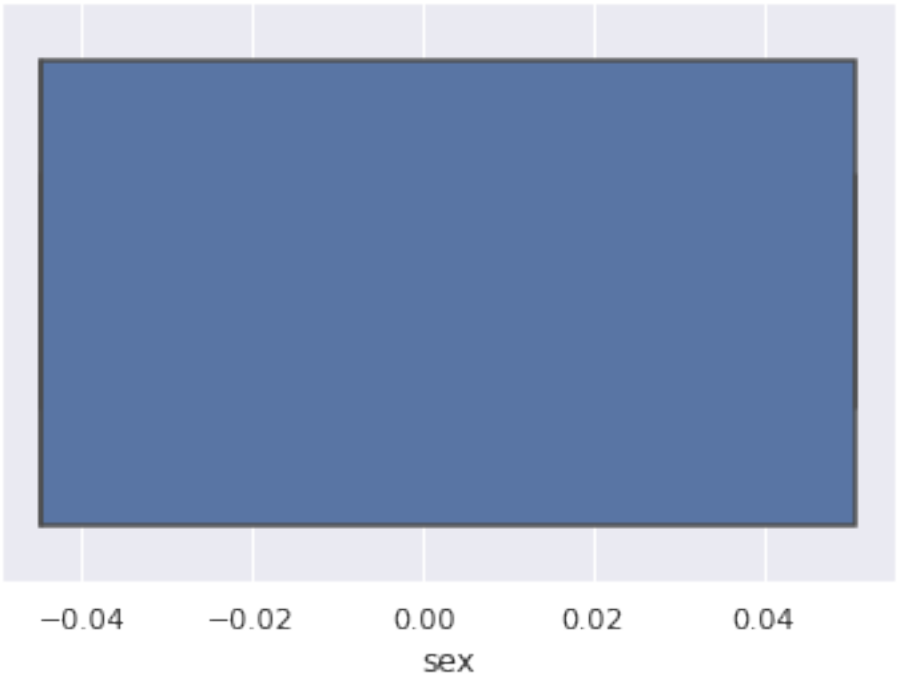


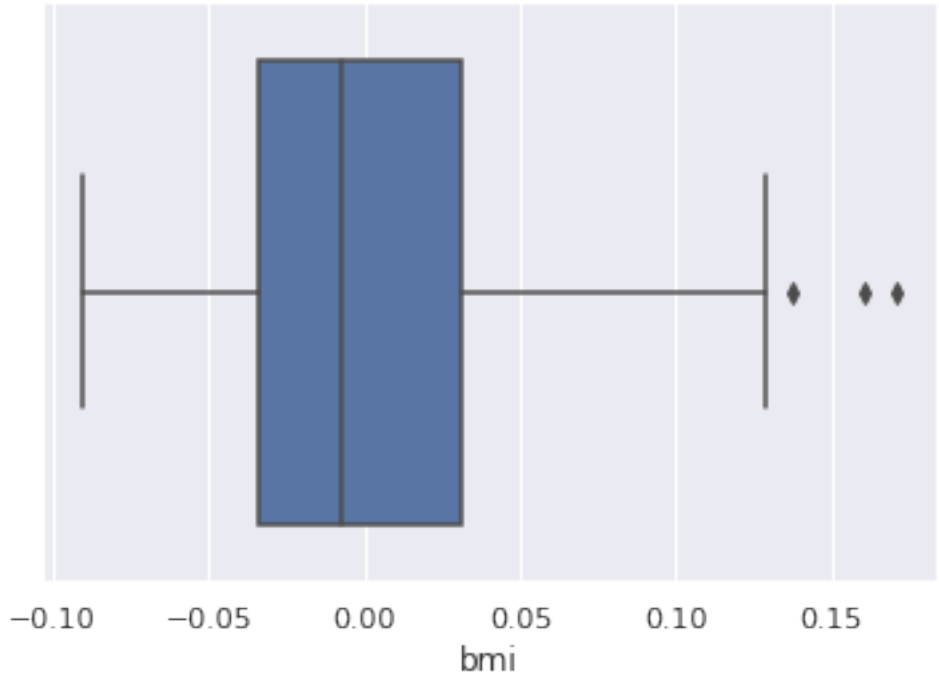
[15]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()
from sklearn.impute import SimpleImputer
import numpy as np
from sklearn.datasets import load_diabetes
%matplotlib inline
diabetes = load_diabetes()

print("Files Imported Sucessfully...!")

#print(diabetes.DESCR)
df=pd.DataFrame(data=diabetes.data,columns=diabetes.feature_names)

#df.head()

df.isna().any()


for col in df.columns:
    fil=df[col].values<-12
    df_new=df[fil]
    sns.boxplot(df[col])
```

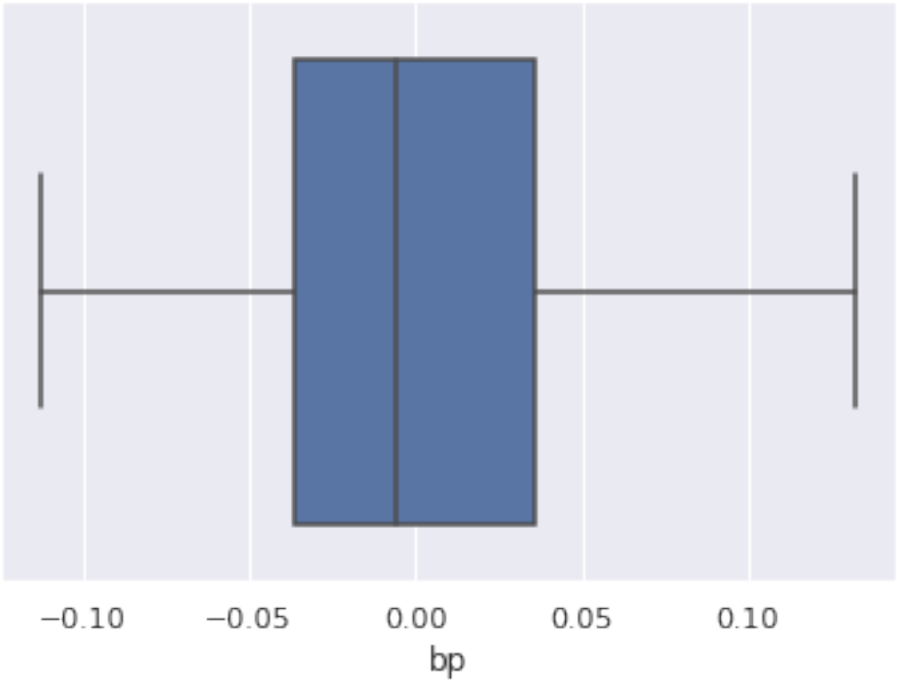
```
    plt.show()
```

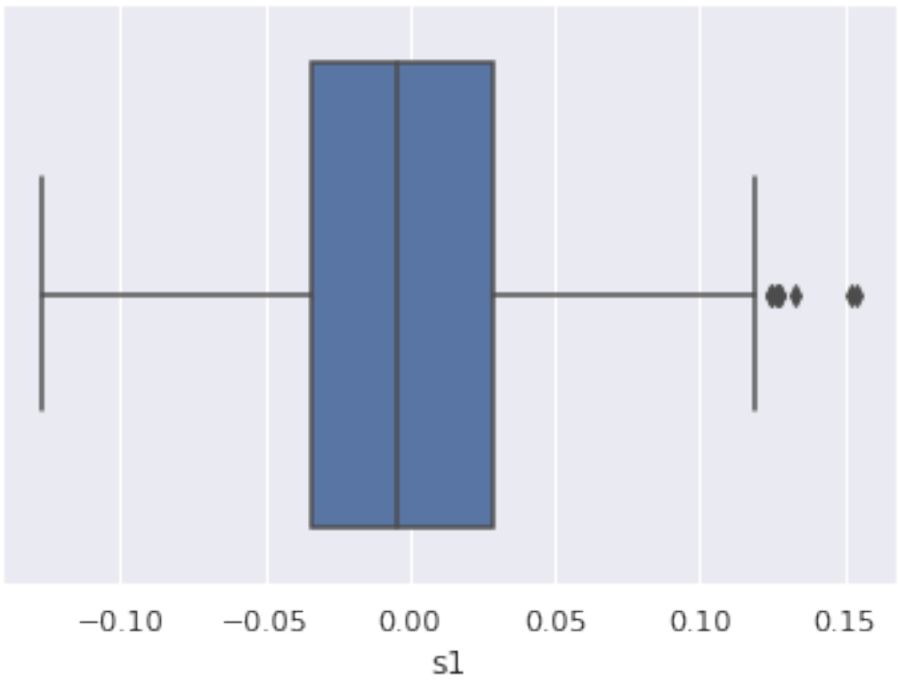
```
Files Imported Sucessfully…!
```

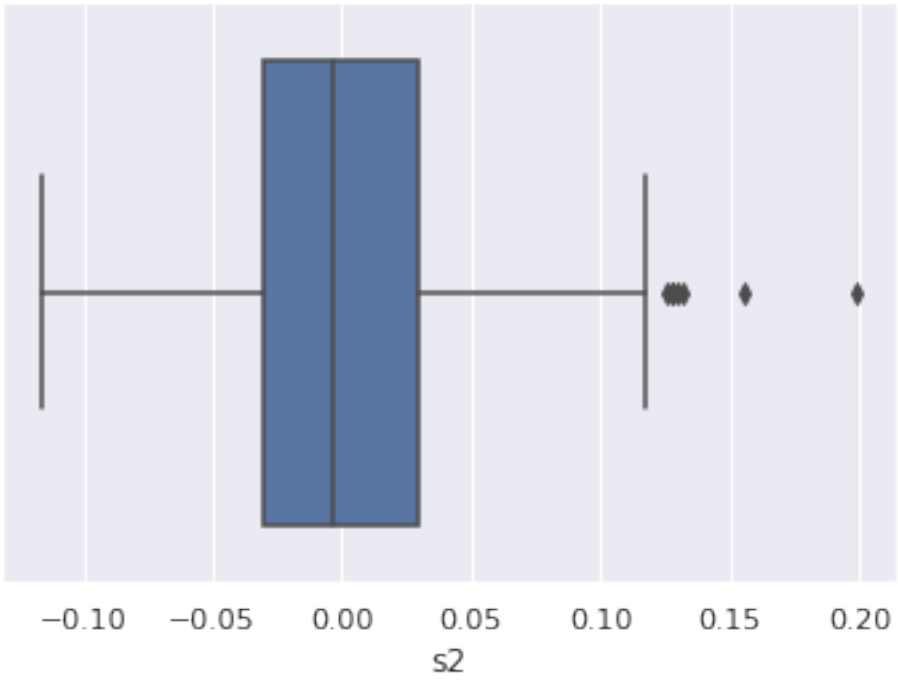


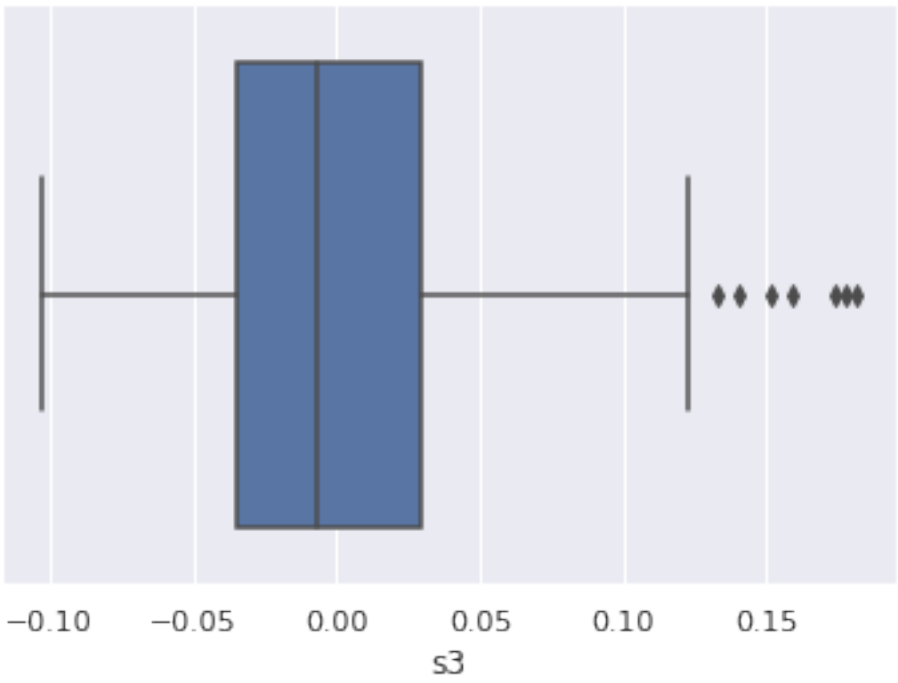


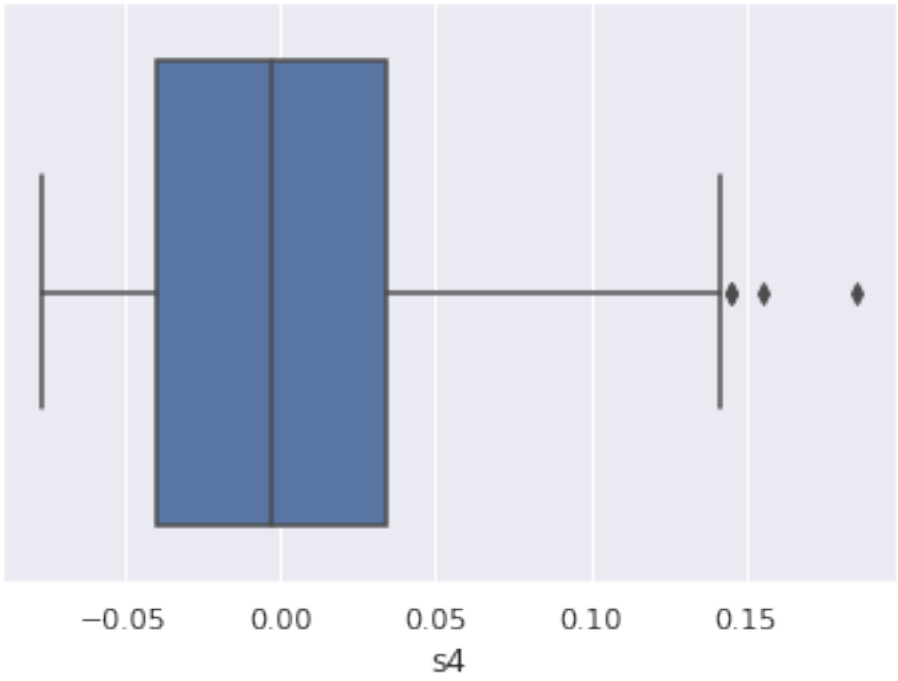

−0.10
−0.05
0.00
0.05
0.10
0.15
bmi

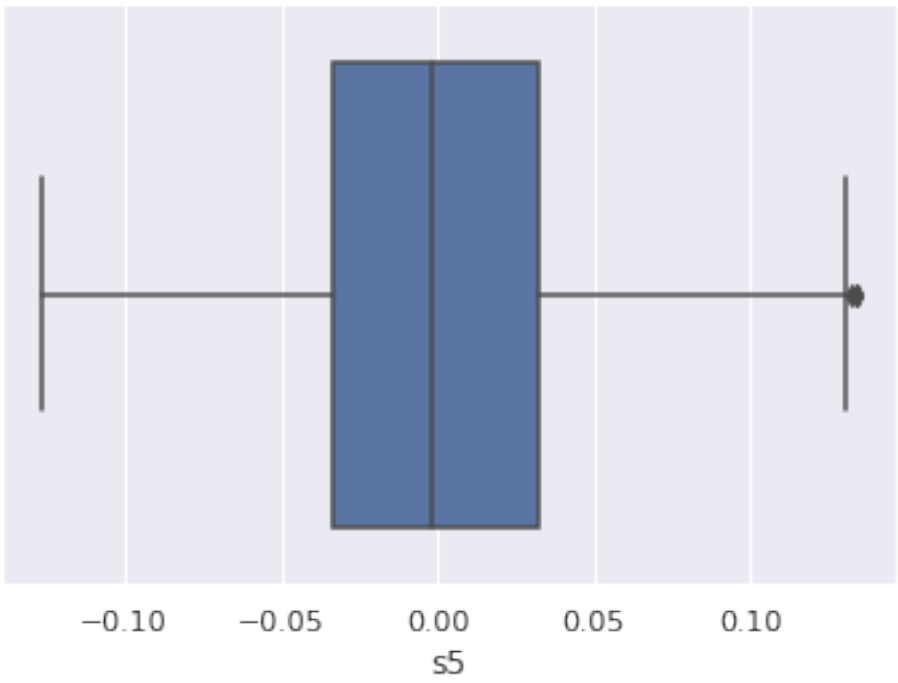

−0.10
−0.05
0.00
0.05
0.10
bp

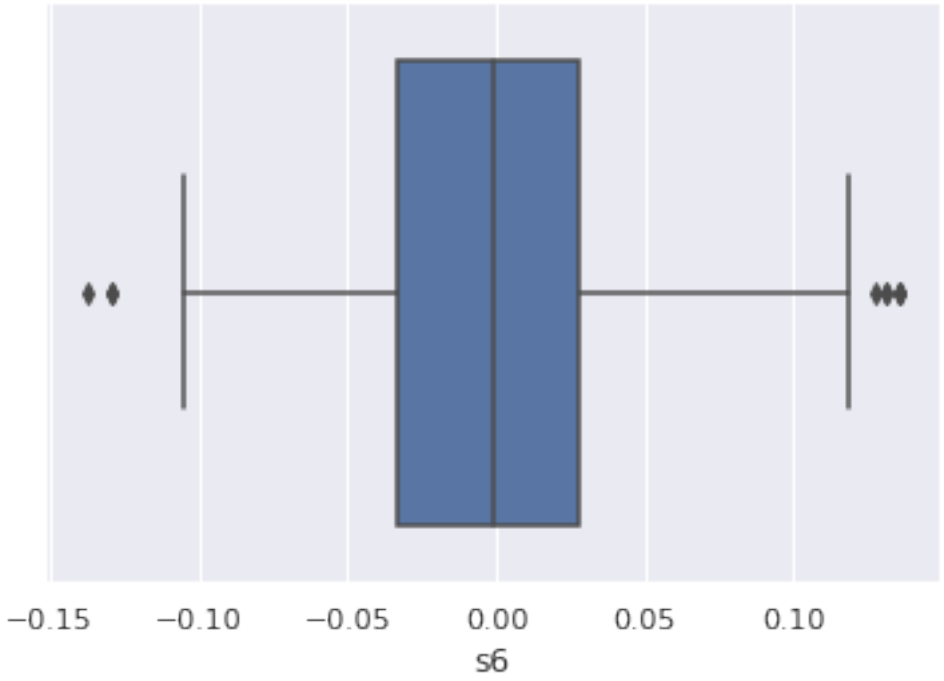

−0.10
−0.05
0.00
0.05
0.10
0.15
s1


−0.10
−0.05
0.00
0.05
0.10
0.15
0.20
s2


−0.10
−0.05
0.00
0.05
0.10
0.15
s3


−0.05
0.00
0.05
0.10
0.15
s4


−0.10
−0.05
0.00
0.05
0.10
s5


−0.15
−0.10
−0.05
0.00
0.05
0.10
s6

[17]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib as plt
%matplotlib inline
import sklearn as sk


school_data = pd.read_csv("school_data.csv")
school_data.head()
```

[17]:

	Serial No	School Name	Student Name	Subject	Score	Result
0	1	ABCD INTL School	Jack	Mathematics	56	Pass
1	2	XYZ Public School	Jim	Science	32	Fail
2	3	ABCD INTL School	Jolie	Science	67	Pass
3	4	JK Secondary School	Jim	Mathematics	75	Pass
4	5	ABCD INTL School	Kavita	Social	55	Pass

[18]:
```
school_data[school_data["Student Name"]=='Jim'].groupby(['School␣
↪Name','Subject']).max()
```

[18]:

		Serial No	Student Name	Score	Result
School Name	Subject				
ABCD INTL School	Science	7	Jim	31	Fail
JK Secondary School	Mathematics	4	Jim	75	Pass
XYZ Public School	Mathematics	6	Jim	66	Pass
	Science	2	Jim	32	Fail
	Social	10	Jim	18	Fail

[19]:
```
new_df_1 = pd.DataFrame({'Country Name' :␣
↪['US','UK','INDIA','EGYPT'],'Currencies':
↪['Dollar','Pounds','Rupees','Riyals']})
new_df_1
```

[19]:

	Country Name	Currencies
0	US	Dollar
1	UK	Pounds
2	INDIA	Rupees
3	EGYPT	Riyals

[20]:
```
new_df_2 = pd.DataFrame({'Country Name':␣
↪['Bangladesh','Singapore','India'],'Currencies':['Taka','Dollars','Rupees']})
new_df_2
```

[20]:

	Country Name	Currencies
0	Bangladesh	Taka
1	Singapore	Dollars

```
2        India     Rupees
```

[21]:
```
pd.concat([new_df_1,new_df_2])
```

[21]:
```
  Country Name Currencies
0           US     Dollar
1           UK     Pounds
2        INDIA     Rupees
3        EGYPT     Riyals
0   Bangladesh       Taka
1    Singapore    Dollars
2        India     Rupees
```

[22]:
```
#how , merge, concade

new_df_1.merge(new_df_2,how='right')
```

[22]:
```
  Country Name Currencies
0   Bangladesh       Taka
1    Singapore    Dollars
2        India     Rupees
```

[23]:
```
import pandas as pd
north_america = pd.read_csv("north_america_2000_2010.csv")
north_america
```

[23]:
```
  Country    2000    2001    2002    2003  …    2006    2007  2008    2009
2010
0  Canada  1779.0  1771.0  1754.0  1740.0  …  1745.0  1741.0  1735  1701.0
1703.0
1  Mexico  2311.2  2285.2  2271.2  2276.5  …  2280.6  2261.4  2258  2250.2
2242.4
2     USA  1836.0  1814.0  1810.0  1800.0  …  1800.0  1798.0  1792  1767.0
1778.0

[3 rows x 12 columns]
```

[24]:
```
south_america = pd.read_csv("south_america_2000_2010.csv")
south_america
```

[24]:
```
  Country  2000  2001  2002  2003  2004  2005  2006  2007  2008  2009    2010
0   Chile  2263  2242  2250  2235  2232  2157  2165  2128  2095  2074  2069.6
```

[25]:
```
data=pd.concat([north_america, south_america])
data.mean()
```

[25]:
```
2000    2047.300
2001    2028.050
2002    2021.300
2003    2012.875
2004    2016.150
2005    1996.000
2006    1997.650
2007    1982.100
2008    1970.000
2009    1948.050
2010    1948.250
dtype: float64
```

[ ]:

[ ]:

[26]:
```
salary = pd.read_csv("Salaries.csv",low_memory=False)
df=pd.DataFrame(salary)

# Total salary cost has increased from year 2011 to 2014

df.info()
df.head()
#df['Year'].unique()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 148648 entries, 0 to 148647
Data columns (total 13 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   Id                148648 non-null  int64
 1   EmployeeName      148648 non-null  object
 2   JobTitle          148648 non-null  object
 3   BasePay           148043 non-null  float64
 4   OvertimePay       148648 non-null  float64
 5   OtherPay          148648 non-null  float64
 6   Benefits          112490 non-null  float64
 7   TotalPay          148648 non-null  float64
 8   TotalPayBenefits  148648 non-null  float64
 9   Year              148648 non-null  int64
 10  Notes             0 non-null       float64
 11  Agency            148648 non-null  object
 12  Status            38119 non-null   object
dtypes: float64(7), int64(2), object(4)
memory usage: 14.7+ MB
```

[26]:

	Id	EmployeeName	…	Agency	Status
0	1	NATHANIEL FORD	…	San Francisco	NaN
1	2	GARY JIMENEZ	…	San Francisco	NaN
2	3	ALBERT PARDINI	…	San Francisco	NaN
3	4	CHRISTOPHER CHONG	…	San Francisco	NaN
4	5	PATRICK GARDNER	…	San Francisco	NaN

[5 rows x 13 columns]

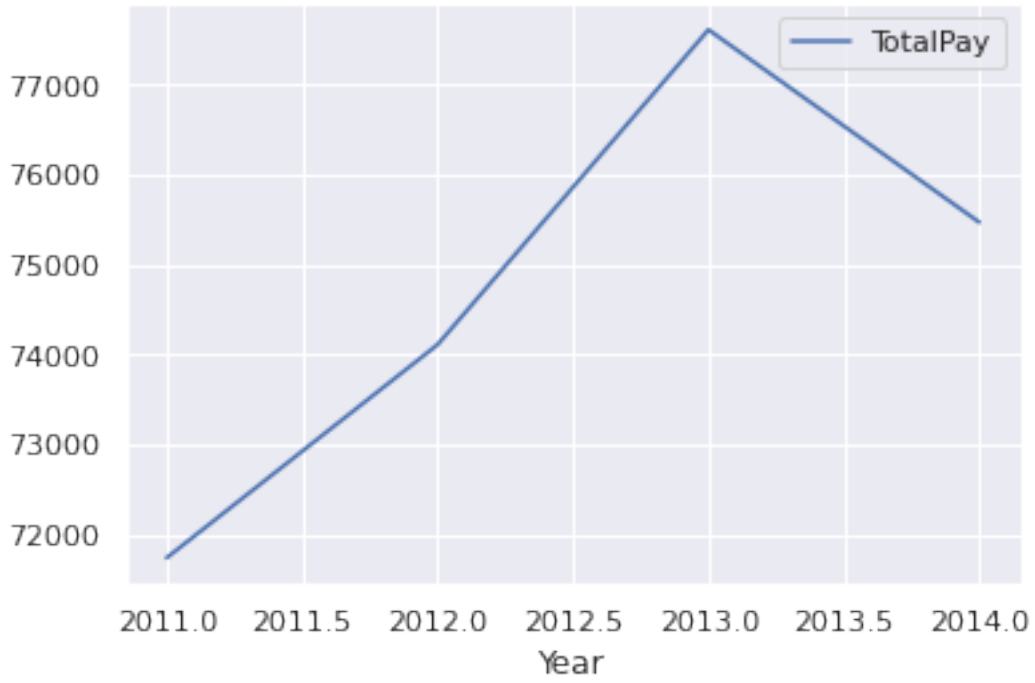
[29]:
```
feature=df[['Year','TotalPay']]
feature

salary_mean = df.groupby('Year').mean()[['TotalPay']]
print(salary_mean)

salary_dif = salary_mean.loc[2014]-salary_mean.loc[2011]
salary_dif
```

```
          TotalPay
Year
2011  71743.819645
2012  74112.234931
2013  77611.443142
2014  75471.836912
```

[29]:
```
TotalPay    3728.017267
dtype: float64
```

[30]:
```
import seaborn as sns
import matplotlib as plt
%matplotlib inline
sns.lineplot(data=salary_mean)
```

[30]:
```
<AxesSubplot:xlabel='Year'>
```



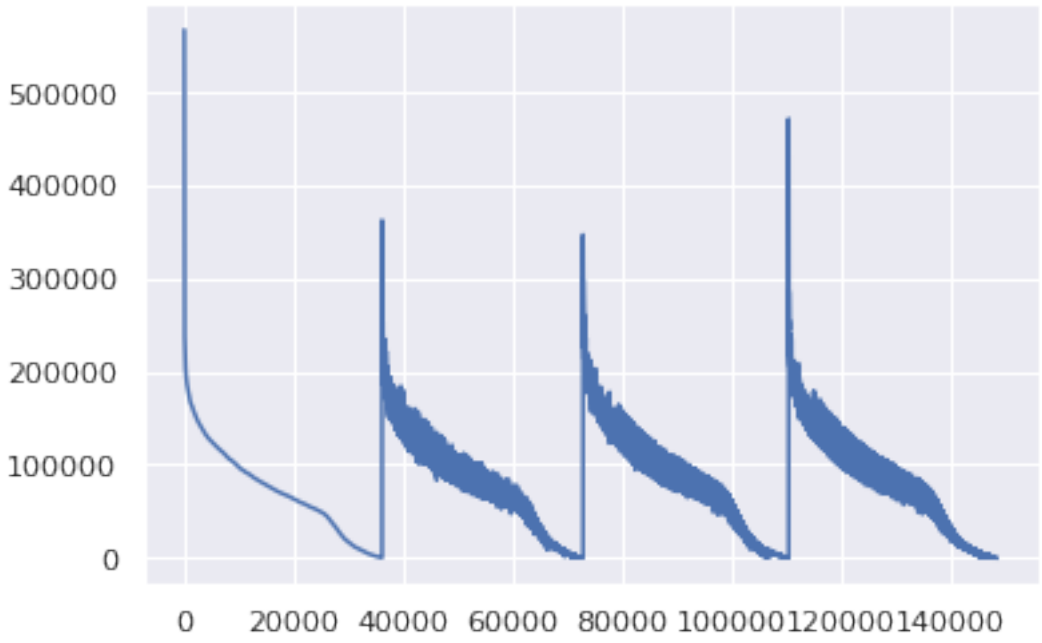
[31]:
```
emp_mean = df.groupby('Year').max()[['TotalPay','EmployeeName']]
emp_mean
```

[31]:

Year	TotalPay	EmployeeName
2011	567595.43	ZURI JONES
2012	362844.66	Zuri Jones
2013	347102.32	Zuri Jones
2014	471952.64	Zuri Jones

[32]:
```
import seaborn as sns
sns.lineplot(data=df['TotalPay'])
```

[32]:
```
<AxesSubplot:>
```



[ ]:

[33]:
```
data = pd.DataFrame({'first_name': ['Jason', 'Molly', 'Tina', 'Jake', 'Amy'],
                     'last_name':['Miller', 'Jacobson', ".", 'Milner', 'Cooze'],
                     'age': [42, 52, 36, 24, 73],
                     'preTestScore': [4, 24, 31, ".", "."],
                     'postTestScore': ["25,000", "94,000", 57, 62, 70]})
data
```

[33]:

	first_name	last_name	age	preTestScore	postTestScore
0	Jason	Miller	42	4	25,000
1	Molly	Jacobson	52	24	94,000
2	Tina	.	36	31	57
3	Jake	Milner	24	.	62
4	Amy	Cooze	73	.	70

[ ]:

[34]:
```
# 1. save dataframe into csv file

data.to_csv("project.csv")
print("Data Exported Sucessfully as 'project.csv' ")
```

```
Data Exported Sucessfully as 'project.csv'
```

[36]:
```
# 2. Read project.csv and print the dataframe

raw = pd.read_csv("project.csv")
print(pd.DataFrame(raw))

print("\nproject.csv printed Sucessfully as 'Dataframe' ")
```

```
   Unnamed: 0 first_name last_name  age preTestScore postTestScore
0           0       Jason    Miller   42            4        25,000
1           1       Molly  Jacobson   52           24        94,000
2           2        Tina         .   36           31            57
3           3        Jake    Milner   24            .            62
4           4         Amy     Cooze   73            .            70

project.csv printed Sucessfully as 'Dataframe'
```

[ ]:
```
raw
```

[ ]:
```

```