In [3]:
```
# Building Regression Models
#
# Anuar Konkashbaev
# School of Technology & Engineering, National University
# Course code: 3602869492
# Professor: Dr. Mohamed Nabeel
# March 20th, 2024
#
```

In [4]:
```
import numpy as np
import pandas as pd
import matplotlib as plt
import seaborn as sns
import statsmodels.api as sm
from statsmodels.formula.api import ols
```

In [5]:
```
# Question 3
```

In [6]:
```
income = 50+(20*4.0)+(0.07*110)+35*(1)+(0.01*4.0*110)-10*(4.0*1)
print("Income is", income)
```

```
Income is 137.1
```

In [7]:
```
# Question #10: Applied part
```

In [8]:
```
df = pd.read_csv('Carseats.csv')
```

In [9]:
```
# Fitting a model as generalised linear regression.
# Since both Urban and US are dichotomous variables, I decided to use same strategy
# In this exercise I am using "formula = Sales ~ Price + Urban + US"
```

In [10]:
```
formula="Sales ~ Price + Urban + US"
mod=ols(formula,data=df)
res=mod.fit()
print(res.summary())
mod2=ols(formula,data=df).fit()
table=sm.stats.anova_lm(mod2)
print(table)
print("\n\nConfidence intervals")
print(mod2.conf_int())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Sales   R-squared:                       0.239
Model:                            OLS   Adj. R-squared:                  0.234
Method:                 Least Squares   F-statistic:                     41.52
Date:                Mon, 31 Mar 2025   Prob (F-statistic):           2.39e-23
Time:                        12:20:08   Log-Likelihood:                -927.66
No. Observations:                 400   AIC:                             1863.
Df Residuals:                     396   BIC:                             1879.
Df Model:                           3
Covariance Type:            nonrobust
================================================================================
                   coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
Intercept       13.0435      0.651     20.036      0.000      11.764      14.323
Urban[T.Yes]    -0.0219      0.272     -0.081      0.936      -0.556       0.512
US[T.Yes]        1.2006      0.259      4.635      0.000       0.691       1.710
Price           -0.0545      0.005    -10.389      0.000      -0.065      -0.044
==============================================================================
Omnibus:                        0.676   Durbin-Watson:                   1.912
Prob(Omnibus):                  0.713   Jarque-Bera (JB):                0.758
Skew:                           0.093   Prob(JB):                        0.684
Kurtosis:                       2.897   Cond. No.                         628.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly spe
cified.
             df       sum_sq     mean_sq           F        PR(>F)
Urban       1.0     0.756615    0.756615    0.123767  7.251713e-01
US          1.0   100.846148  100.846148   16.496407  5.877444e-05
Price       1.0   659.837263  659.837263  107.936143  1.609917e-22
Residual  396.0  2420.834671    6.113219         NaN           NaN


Confidence intervals
                      0          1
Intercept     11.763597  14.323341
Urban[T.Yes]  -0.555973   0.512141
US[T.Yes]      0.691304   1.709841
Price         -0.064764  -0.044154
```

In [11]:
```
# Coefficient for the "Price" is -0.0545 and indicates that smaller price leads to
# Coefficient for "US" is 1.2006 and indicates that it is main contributing factor
# Coefficient for "Urban" is -0.0219, which indicates that Urban leads to lower sal
# "Sales" and "Price" are numerical variables, whereas "Urban" and "US" are dichoto
# Residuals degree of freedom is 396, model degree of freedom is 3, for a total of
# R-squared is 0.239 and adjusted R-squared 0.234, which is low. F-statistic is 41.
# Overall impression is that a better model is needed in this case.
# If the beta for a variable is 0, variable is not contributing to the outcome. Nei
# However, we cannot reject the null hypothesis for Urban, which has a non-signific
```

In [12]:
```
# I will use "US" and "Price" in a smaller model with formula "Sales ~ Price + US"
```

In [13]:
```
formula="Sales ~ Price + US"
mod=ols(formula,data=df)
res=mod.fit()
print(res.summary())
mod2=ols(formula,data=df).fit()
table=sm.stats.anova_lm(mod2)
print(table)
print("\n\nConfidence intervals")
print(mod2.conf_int())
```

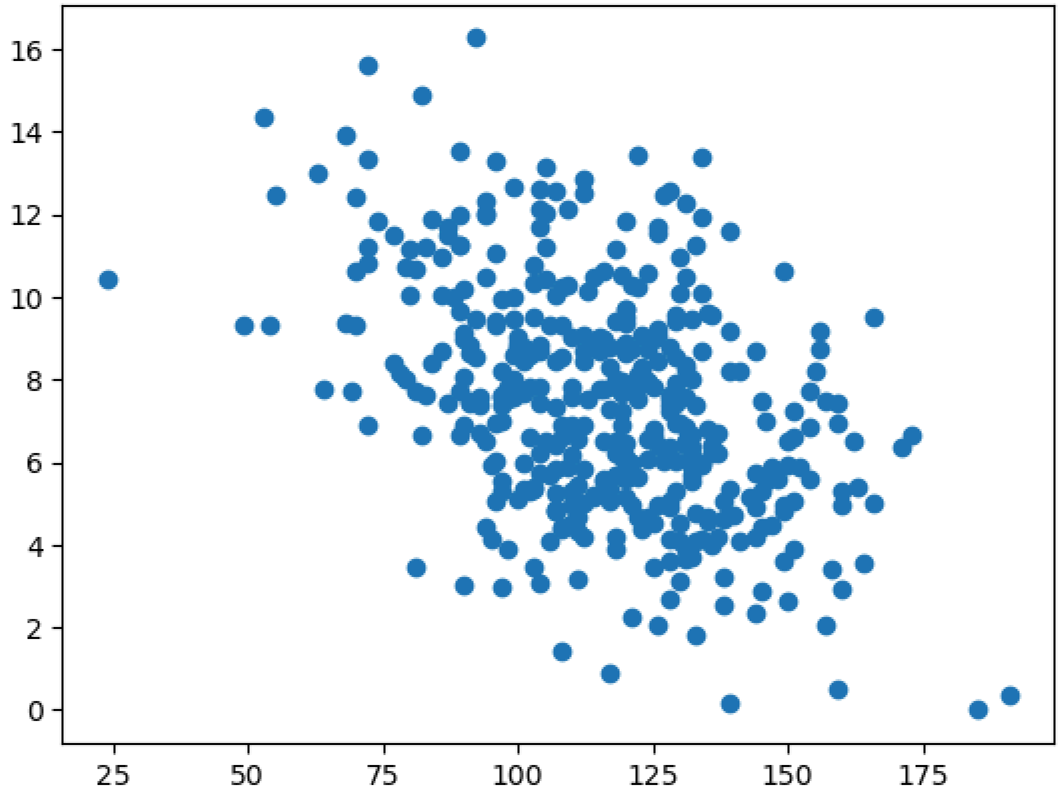
```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Sales   R-squared:                       0.239
Model:                            OLS   Adj. R-squared:                  0.235
Method:                 Least Squares   F-statistic:                     62.43
Date:                Mon, 31 Mar 2025   Prob (F-statistic):           2.66e-24
Time:                        12:20:15   Log-Likelihood:                -927.66
No. Observations:                 400   AIC:                             1861.
Df Residuals:                     397   BIC:                             1873.
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     13.0308      0.631     20.652      0.000      11.790      14.271
US[T.Yes]      1.1996      0.258      4.641      0.000       0.692       1.708
Price         -0.0545      0.005    -10.416      0.000      -0.065      -0.044
==============================================================================
Omnibus:                        0.666   Durbin-Watson:                   1.912
Prob(Omnibus):                  0.717   Jarque-Bera (JB):                0.749
Skew:                           0.092   Prob(JB):                        0.688
Kurtosis:                       2.895   Cond. No.                         607.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly spe
cified.
             df       sum_sq     mean_sq           F        PR(>F)
US          1.0    99.802580   99.802580   16.366658  6.273751e-05
Price       1.0   661.597655  661.597655  108.495617  1.272157e-22
Residual  397.0  2420.874462    6.097921         NaN           NaN


Confidence intervals
                  0          1
Intercept  11.79032  14.271265
US[T.Yes]   0.69152   1.707766
Price      -0.06476  -0.044195
```

In [14]:
```
# Both full and smaller models are practically the same; note, for example, the adj
# I excluded the "Urban" vavriable because it does not substantially comtribute add
# I also calculated the confidence intervals for the coefficients (see above).
```

In [15]:
```
# Both "Urban" and "US" are dichotomous, only "Price" variable is numerical
# Visualizing prediction
```

```
plt.pyplot.scatter(df["Price"],df["Sales"], label='Original model')
```

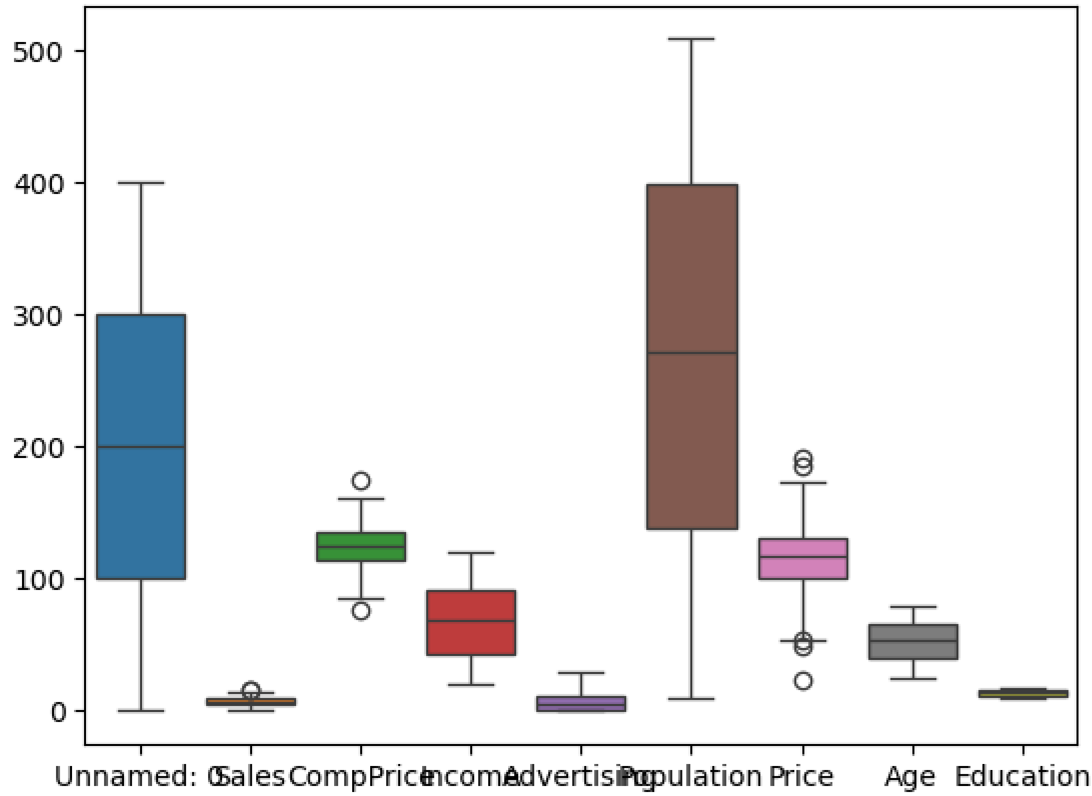
Out[15]: `<matplotlib.collections.PathCollection at 0x203d0099d90>`



From the scatter plot, it does not appear that the dataset has any outliers. But we need to have a better meassurement.

In [19]:
```
# Detecting outliers using boxplot:
sns.boxplot(data=df)
```

Out[19]: `<Axes: >`



In [ ]:
```
# Now we can see that both Sales and Price do have outliers.
```