Importing the required packages

In [5]:
```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
import sklearn.metrics as met
import statsmodels.api as sm
import random as rd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [3]:
```
adults = pd.read_csv('C:/School/DSC-530/DataSets/Adult')
```

In [6]:
```
adults
```

Out[6]:

	age	workclass	demogweight	education	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	nat cou
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male	2174	0	40	Uni St
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	13	Uni St
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family	White	Male	0	0	40	Uni St
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	0	0	40	Uni St
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	Black	Female	0	0	40	(
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
24995	41	Private	112507	10th	6	Married-civ-spouse	Transport-moving	Husband	White	Male	0	0	60	Uni St
24996	19	Private	236940	HS-grad	9	Never-married	Farming-fishing	Own-child	White	Male	0	0	40	Uni St
24997	33	Private	278514	HS-grad	9	Divorced	Craft-repair	Own-child	White	Female	0	0	42	Uni St
24998	21	?	433330	Some-college	10	Never-married	?	Unmarried	White	Male	0	0	40	Uni St
24999	25	Private	258379	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife	White	Female	0	0	32	Uni St

25000 rows × 15 columns

1. Partition the data set into a training set and a test set, each containing about half of the records.

In [8]:
```
#Partitioning the data using the train_test_split() command
adult_train, adult_test =  train_test_split(adults, test_size = 0.50, random_state = 5)
```

In [10]:
```
adult_test
```

Out[10]:

	age	workclass	demogweight	education	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	nat cou
12653	23	Private	181820	Some-college	10	Never-married	Farming-fishing	Unmarried	White	Male	0	0	45	Uni St
22745	37	Self-emp-not-inc	154641	Assoc-acdm	12	Married-civ-spouse	Farming-fishing	Husband	White	Male	2105	0	50	Uni St
18675	30	Self-emp-not-inc	166961	Some-college	10	Married-civ-spouse	Adm-clerical	Wife	White	Female	0	0	20	Uni St
23488	40	Private	283174	Assoc-voc	11	Married-civ-spouse	Farming-fishing	Husband	White	Male	0	0	60	Uni St
5580	44	Private	112517	Masters	14	Married-civ-spouse	Tech-support	Husband	White	Male	0	0	20	Uni St
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
287	50	Private	176609	Some-college	10	Divorced	Other-service	Not-in-family	White	Male	0	0	45	Uni St
21269	29	Private	241667	Masters	14	Never-married	Prof-specialty	Not-in-family	White	Female	0	0	45	Uni St
17654	57	Private	159319	Masters	14	Married-civ-spouse	Exec-managerial	Husband	White	Male	0	0	50	Uni St
9991	55	State-gov	337599	Some-college	10	Divorced	Adm-clerical	Not-in-family	White	Male	0	0	40	Uni St
16173	19	?	129586	Some-college	10	Never-married	?	Own-child	White	Male	0	0	40	Uni St

12500 rows × 15 columns

In [11]:
```
adult_train
```

Out[11]:

	age	workclass	demogweight	education	education-num	marital-status	occupation	relationship	race	sex	capital-gain	capital-loss	hours-per-week	na cou
3493	35	Private	148903	HS-grad	9	Married-civ-spouse	Other-service	Wife	White	Female	0	0	16	Un S
21765	22	Private	326334	Some-college	10	Never-married	Craft-repair	Own-child	White	Male	0	0	35	Un S
23692	31	Private	37546	Bachelors	13	Never-married	Prof-specialty	Not-in-family	White	Female	0	0	40	Un S
600	32	Private	239824	Bachelors	13	Never-married	Tech-support	Not-in-family	White	Male	0	0	40	Un S
24820	29	Private	79481	Some-college	10	Never-married	Tech-support	Not-in-family	White	Female	0	0	40	Un S
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
23670	26	?	208994	Some-college	10	Never-married	?	Own-child	White	Male	0	0	12	Un S
3046	54	Private	220115	Some-college	10	Divorced	Adm-clerical	Not-in-family	White	Male	0	0	30	Un S
20463	85	Self-emp-not-inc	166027	HS-grad	9	Widowed	Sales	Not-in-family	White	Female	0	0	50	Un S
18638	36	Private	469056	HS-grad	9	Divorced	Sales	Unmarried	Black	Female	0	0	25	Un S
2915	26	Private	198163	Masters	14	Married-civ-spouse	Sales	Wife	White	Female	0	0	40	Un S

12500 rows × 15 columns

In [13]:
```
#The data of the adult_train and adult_test shows that the adult data have been partitioned into
#two equal halfs
```

2.Run a regression model to predict Hours per Week using Age and Education Num. Obtain a summary of the model. Are there any predictor variables that should not be in the model?

In [12]:
```
#Separating the data using data frame into predictors and target variables. The pred variable represent the x c
#independent variable while the target represent the target or dependent variable
pred = pd.DataFrame(adult_train[['age', 'education-num']])
target = pd.DataFrame(adult_train[['hours-per-week']])
```

In [13]:
```
pred
```

Out[13]:

	age	education-num
3493	35	9
21765	22	10
23692	31	13
600	32	13
24820	29	10
...	...	...
23670	26	10
3046	54	10
20463	85	9
18638	36	9
2915	26	14

12500 rows × 2 columns

In [14]:
```
target
```

Out[14]:

	hours-per-week
3493	16
21765	35
23692	40
600	40
24820	40
...	...
23670	12
3046	30
20463	50
18638	25
2915	40

12500 rows × 1 columns

In [15]:
```
#Adding a constant b0 to the regression model
pred = sm.add_constant(pred)
```

In [16]:
```
#Now running the multiple regression model
model01 = sm.OLS(target, pred).fit()
```

In [17]:
```
model01.summary()
```

Out[17]:

OLS Regression Results

Dep. Variable:	hours-per-week	R-squared:	0.025
Model:	OLS	Adj. R-squared:	0.025
Method:	Least Squares	F-statistic:	162.9
Date:	Tue, 07 Sep 2021	Prob (F-statistic):	1.46e-70
Time:	07:16:37	Log-Likelihood:	-48863.
No. Observations:	12500	AIC:	9.773e+04
Df Residuals:	12497	BIC:	9.776e+04
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	31.2676	0.524	59.716	0.000	30.241	32.294
age	0.0611	0.008	7.692	0.000	0.046	0.077
education-num	0.6746	0.042	15.998	0.000	0.592	0.757

Omnibus:	982.558	Durbin-Watson:	2.005
Prob(Omnibus):	0.000	Jarque-Bera (JB):	4966.320
Skew:	0.198	Prob(JB):	0.00
Kurtosis:	6.062	Cond. No.	204.

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

In [22]:
```
#The regression summary above shows that we have the regression coefficients for age
#and education-num as 0.0611 and 0.6746, respectively. The p-value for both the age and education-num
# is less than the cutoff value of 0.05, which allowed for retaining a variable
# in a regression model, so all the predictors used should be included in the model.
```

3. Validate the model from the previous exercise.

In [18]:
```
#Now validating the model using the test data. the pred_test represent the predictors on the test data
#and the target_test represent the dependent variables on the test data.
pred_test = pd.DataFrame(adult_test[['age', 'education-num']])
target_test = pd.DataFrame(adult_test[['hours-per-week']])
```

In [19]:
```
pred_test
```

Out[19]:

	age	education-num
12653	23	10
22745	37	12
18675	30	10
23488	40	11
5580	44	14
...	...	...
287	50	10
21269	29	14
17654	57	14
9991	55	10
16173	19	10

12500 rows × 2 columns

In [21]:
```
target_test
```

Out[21]:

	hours-per-week
12653	45
22745	50
18675	20
23488	60
5580	20
...	...
287	45
21269	45
17654	50
9991	40
16173	40

12500 rows × 1 columns

In [22]:
```
#Add the constants b0
pred_test = sm.add_constant(pred_test)
```

In [23]:
```
#Now running the regression model
model01_test = sm.OLS(target_test, pred_test).fit()
```

In [24]:
```
#Summarizing the model
model01_test.summary()
```

Out[24]:

OLS Regression Results

Dep. Variable:	hours-per-week	R-squared:	0.025
Model:	OLS	Adj. R-squared:	0.025
Method:	Least Squares	F-statistic:	163.4
Date:	Tue, 07 Sep 2021	Prob (F-statistic):	8.54e-71
Time:	07:17:13	Log-Likelihood:	-49025.
No. Observations:	12500	AIC:	9.806e+04
Df Residuals:	12497	BIC:	9.808e+04
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	31.1326	0.536	58.065	0.000	30.082	32.184
age	0.0547	0.008	6.890	0.000	0.039	0.070
education-num	0.7086	0.043	16.544	0.000	0.625	0.793

Omnibus:	1012.013	Durbin-Watson:	1.994
Prob(Omnibus):	0.000	Jarque-Bera (JB):	5142.734
Skew:	0.217	Prob(JB):	0.00
Kurtosis:	6.112	Cond. No.	208.

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

In [32]:
```
#The above result shows validated the model model01 is accurate because the p-value is less than
#the cutoff value, same as the model
```

4. Use the regression equation to complete this sentence: “The estimated hours per week equals....”

In [33]:
```
#The estimated hours per week equals $31.2676 plus $0.0611 times the age of the adults plus
#0.6746 of number of education obtained. or we say hours per week = 31.2676 + 0.0611(age) + 0.6746(education-nu
```

5. Interpret the coefficient for Age.

In [34]:
```
#The coefficient of the Age implies that hours per week for each average Age is estimated
#to increase to 0.0611 age.
```

6. Interpret the coefficient for Education Num

In [35]:
```
#The coefficient for the Education num implies that the hour per week for each number of
#of education is estimated to increase to 0.6746
```

7. Find and interpret the value of s

In [25]:
```
#To get the s which is the prediction error, we will take the squares of the model
np.sqrt(model01.scale)
```

Out[25]:
```
12.06437437928219
```

In [37]:
```
#This shows that the size of the model prediction error is 12.06 hours per week
```

8. Find and interpret Radj 2

In [38]:
```
#On the model summary, it shows that our regression model have Radj 2 as 0.025, which indicates the
#The age and the education-num account for 2.4 percent of the variability in hours per week.
#This small amount of variation is expected since there could still be other factors like
#marital status, occupation, relationship, race, etc.
```

9. Find MAEBaseline and MAERegression, and determine whether the regression model outperformed its baseline model.

In [26]:
```
#Finding the MAE of the regression, to get this we need the predicted and actual value for the target

#Obtaining the predicted target values
ypred = model01.predict(pred)
```

In [27]:
```
ypred
```

Out[27]:
```
3493     39.477563
21765    39.358006
23692    41.931761
600      41.992853
24820    39.785651
           ...
23670    39.602375
3046     41.312958
20463    42.532177
18638    39.538656
2915     42.300941
Length: 12500, dtype: float64
```

In [28]:
```
#Obtaining the actual target values
ytrue = adult_train[['hours-per-week']]
```

In [29]:
```
#Calculating the MAE regression values
met.mean_absolute_error(y_true= ytrue, y_pred=ypred)
```

Out[29]:
```
7.703084213268585
```

In [49]:
```
#The result above shows that he mean absolute error of the model is 7.70.
```

In [30]:
```
#Now finding the baseline MAE

#Obtaining the predicted target values for the test data
ypred_test = model01_test.predict(pred_test)
```

In [31]:
```
#Obtaining the actual target values
ytrue_test = adult_test[['hours-per-week']]
```

In [32]:
```
#Calculating the MAE baseling values
met.mean_absolute_error(y_true= ytrue_test, y_pred=ypred_test)
```

Out[32]:
```
7.821581622406873
```

In [53]:
```
#The result above shows that the MAE regression is 7.70 while the MAE baseline is 7.82
#this indicates that the MAE regression < MAE baseline, therefore making the model outperforms the baseline moc
```

In [54]:
```
#Reference
#Larose, C. D., Larose, D. T., & Larose, Chantal D., Author. (2019). Data science using python and r. John Wile
```

Part 2 Question 1: Referring to your notes from DSC-520, show that the regression model

created in Part 1 is mathematically consistent with the least-squares methodology for linear regression.

In [ ]:
```
#The least-squares method is the most common method for fitting a regression line.
#By reducing the sum of the squares of the vertical deviations from each data point to the line,
#this method calculates the best-fitting line for the observed data
#(if a point lies on the fitted line exactly, then its vertical deviation is 0).
#There are no cancellations between positive and negative numbers since the variations are
#squared first and then summed. After a regression line has been constructed for a set of data,
#an outlier is a point that is far from the line.
#These points could suggest erroneous data or a poorly fitting regression line.
#An influential observation is separated from the other data in the horizontal direction;
#linear regression models the relationship between two variables by fitting a linear equation
#to observed data. One variable is regarded as an explanatory variable, while the other is
#regarded as a dependent variable. Our model is consistent with the least-square methodology as
#it models the comparison of two variables to get the best-fitted line.
```

Part 2 Question 2: Does the regression model's correlation coefficient r, and the coefficient of

determination r squared, demonstrate a strong, medium, or weak relationship between the predictor and criterion variables? Is the regression result statistically significant given the p-value?

Assume a 0.05 cutoff value.

In [35]:
```
#The coefficient of determination is a metric for determining how much variability in one
#component can be attributed to its relationship with another.
#The "goodness of fit," or correlation, is expressed as a number between 0.0 and 1.0.
#A result of 1.0 implies a perfect fit and consequently a very dependable model for future
#forecasts, whereas a value of 0.0 shows that the calculation failed to describe the data
#effectively. However, a value of 0.20 indicates that the independent variable predicts 20%
#of the dependent variable. In contrast, a value of 0.50 indicates that the independent variable
#predicts 50% of the dependent variable, and so on. Our model shows that 2.5% of the independent
#(age and education-num) is used to predict the dependent variable (hours-per-week).
#This percentage is expected because there are other factors in our dataset.
#There the coefficient has a medium relationship with the predictor.
#The regression is statistically significant because its p-value is less than the cutoff value
#of 0.05
```

In [ ]:
```
#Reference
#How the coefficient of determination works. (n.d.). Investopedia.
#     https://www.investopedia.com/terms/c/coefficient-of-determination.asp
#Larose, C. D., Larose, D. T., & Larose, Chantal D., Author. (2019). Data science using python and r.
#     John Wiley & Sons,inc,.
```