Machine-Learning-Mini-Lab-Project-1

February 1, 2021

[2]:
```
# Import required libraries
import numpy as np
import pandas as pd

# Display all columns in pandas
pd.set_option('display.max_columns', None)
```

[3]:
```
import os
path = r'C:\Users\srsid\Documents\GitHub\Machine-Learning-Mini-Lab-Project-1'
os.chdir(path)
os.listdir()
```

[3]:
```
['.git',
 'Machine-Learning-Mini-Lab-Project-1 -Code (Grade This).pdf',
 'Machine-Learning-Mini-Lab-Project-1 .ipynb',
 'Mini-Project 1(Grade This).pdf',
 'Mini-Project 1.docx',
 'PPHA_30545_MP01-Crosswalk.csv',
 'usa_00008.csv',
 '~$ni-Project 1.docx']
```

[4]:
```
acs_data = pd.read_csv('usa_00008.csv')
acs_data.head()
```

[4]:

	YEAR	SAMPLE	SERIAL	CBSERIAL	HHWT	CLUSTER	STRATA	GQ	\
0	2019	201901	2611	2019000016124	19504.8	2019000026111	80001	1	
1	2019	201901	3075	2019000047906	12074.4	2019000030751	250001	1	
2	2019	201901	3075	2019000047906	12074.4	2019000030751	250001	1	
3	2019	201901	3230	2019000058745	23065.2	2019000032301	30201	1	
4	2019	201901	3385	2019000070359	7275.6	2019000033851	60001	1	

	PERNUM	PERWT	NCHILD	NCHLT5	SEX	AGE	MARST	RACE	RACED	HISPAN	\
0	1	19659.6	1	1	1	29	1	1	100	0	
1	1	12074.4	1	0	2	45	4	1	100	0	
2	3	18111.6	0	0	2	30	4	1	100	0	
3	1	23065.2	0	0	2	49	4	2	200	0	
4	1	7275.6	0	0	2	23	6	1	100	0	

```
   HISPAND  EDUC  EDUCD  EMPSTAT  EMPSTATD  INCWAGE  VETSTAT  VETSTATD
0        0     8     81        1        10    59000        1        11
1        0     5     50        1        10    20000        1        11
2        0     6     63        1        10     2000        1        11
3        0    10    101        1        10      900        1        11
4        0     7     71        1        10    24300        1        11
```

[5]:
```
crosswalk = pd.read_csv('PPHA_30545_MP01-Crosswalk.csv')
crosswalk.head()
```

[5]:
```
   educd  educdc
0      2     0.0
1     10     0.0
2     11     2.0
3     12     0.0
4     13     2.5
```

[6]:
```
#Create continuous for EDUCD
acs_data = pd.merge(acs_data, crosswalk, left_on='EDUCD', right_on='educd')
acs_data
```

[6]:
```
      YEAR  SAMPLE   SERIAL       CBSERIAL     HHWT        CLUSTER   STRATA  \
0     2019  201901     2611  2019000016124  19504.8  2019000026111    80001
1     2019  201901     6016  2019000247422   1702.8  2019000060161   230001
2     2019  201901     6790  2019000301008   8668.8  2019000067901   190001
3     2019  201901     9577  2019000497180  42105.6  2019000095771    10001
4     2019  201901     9731  2019000507929   7430.4  2019000097311   270201
…      …      …        …              …        …              …       …
9049  2019  201901  1220732  2019000282619  26006.4  2019012207321   230548
9050  2019  201901  1222589  2019000308283  33746.4  2019012225891   680448
9051  2019  201901  1237295  2019000512472  20433.6  2019012372951   670248
9052  2019  201901  1302776  2019001404140  27554.4  2019013027761   461548
9053  2019  201901  1308039  2019000485767  13312.8  2019013080391  5700249

      GQ  PERNUM    PERWT  NCHILD  NCHLT5  SEX  AGE  MARST  RACE  RACED  \
0      1       1  19659.6       1       1    1   29      1     1    100
1      1       1   1702.8       3       1    1   41      1     1    100
2      1       3  20278.8       0       0    1   21      6     1    100
3      1       1  42105.6       0       0    1   20      6     1    100
4      1       1   7275.6       0       0    1   33      6     2    200
…    ..      …       …      …      …  …  …    …    …     …
9049   1       1  25851.6       2       1    1   47      1     1    100
9050   1       2  39628.8       0       0    1   43      6     1    100
9051   1       2  21207.6       2       0    2   59      1     1    100
9052   1       1  27554.4       0       0    2   49      3     7    700
9053   1       1  13312.8       2       0    1   61      1     1    100
```

```
      HISPAN  HISPAND  EDUC  EDUCD  EMPSTAT  EMPSTATD  INCWAGE  VETSTAT  \
0          0        0     8     81        1        10    59000        1
1          0        0     8     81        1        10    50000        2
2          0        0     8     81        1        10    10000        1
3          0        0     8     81        1        10      800        1
4          0        0     8     81        1        10    45000        1
…        …      …     …     …       …        …       …        …
9049       1      100     2     22        1        10    25000        1
9050       1      100     2     22        1        10    21600        1
9051       1      100     2     22        1        10    18000        1
9052       4      416     2     22        1        10    17000        1
9053       1      100     1     14        1        10    38000        1

      VETSTATD  educd  educdc
0           11     81    14.0
1           20     81    14.0
2           11     81    14.0
3           11     81    14.0
4           11     81    14.0
…         …    …      …
9049        11     22     5.0
9050        11     22     5.0
9051        11     22     5.0
9052        11     22     5.0
9053        11     14     1.0

[9054 rows x 28 columns]
```

[7]:
```
acs_data.columns
```

[7]:
```
Index(['YEAR', 'SAMPLE', 'SERIAL', 'CBSERIAL', 'HHWT', 'CLUSTER', 'STRATA',
       'GQ', 'PERNUM', 'PERWT', 'NCHILD', 'NCHLT5', 'SEX', 'AGE', 'MARST',
       'RACE', 'RACED', 'HISPAN', 'HISPAND', 'EDUC', 'EDUCD', 'EMPSTAT',
       'EMPSTATD', 'INCWAGE', 'VETSTAT', 'VETSTATD', 'educd', 'educdc'],
      dtype='object')
```

[8]:
```
acs_data['EDUCD'].unique()
```

[8]:
```
array([ 81,  50,  63, 101,  71,  65,  30, 114,  64,  25,  61, 116,  40,
       115,   2,  23,  15,  26,  12,  17,  11,  16,  22,  14], dtype=int64)
```

[9]:
```
#Get Dummies
acs_data['hsdip'] = np.where(acs_data['EDUCD'] == 63, 1, 0)
acs_data['coldip'] = np.where(acs_data['EDUCD'] == 101,  1, 0)
acs_data['white'] = np.where(acs_data['RACE'] == 1,  1, 0)
acs_data['black'] = np.where(acs_data['RACE'] == 2,  1, 0)
acs_data['hispanic'] = np.where(acs_data['RACE'] != 0,  1, 0)
```

```
acs_data['married'] = np.where(acs_data['MARST'] == 1,  1, 0)
acs_data['female'] = np.where(acs_data['SEX'] == 2,  1, 0)
acs_data['vet'] = np.where(acs_data['VETSTAT'] == 2,  1, 0)
acs_data.head()
```

[9]:

	YEAR	SAMPLE	SERIAL	CBSERIAL	HHWT	CLUSTER	STRATA	GQ
0	2019	201901	2611	2019000016124	19504.8	2019000026111	80001	1
1	2019	201901	6016	2019000247422	1702.8	2019000060161	230001	1
2	2019	201901	6790	2019000301008	8668.8	2019000067901	190001	1
3	2019	201901	9577	2019000497180	42105.6	2019000095771	10001	1
4	2019	201901	9731	2019000507929	7430.4	2019000097311	270201	1

	PERNUM	PERWT	NCHILD	NCHLT5	SEX	AGE	MARST	RACE	RACED	HISPAN
0	1	19659.6	1	1	1	29	1	1	100	0
1	1	1702.8	3	1	1	41	1	1	100	0
2	3	20278.8	0	0	1	21	6	1	100	0
3	1	42105.6	0	0	1	20	6	1	100	0
4	1	7275.6	0	0	1	33	6	2	200	0

	HISPAND	EDUC	EDUCD	EMPSTAT	EMPSTATD	INCWAGE	VETSTAT	VETSTATD	educd
0	0	8	81	1	10	59000	1	11	81
1	0	8	81	1	10	50000	2	20	81
2	0	8	81	1	10	10000	1	11	81
3	0	8	81	1	10	800	1	11	81
4	0	8	81	1	10	45000	1	11	81

	educdc	hsdip	coldip	white	black	hispanic	married	female	vet
0	14.0	0	0	1	0	1	1	0	0
1	14.0	0	0	1	0	1	1	0	1
2	14.0	0	0	1	0	1	0	0	0
3	14.0	0	0	1	0	1	0	0	0
4	14.0	0	0	0	1	1	0	0	0

[10]:
```
#Interaction
acs_data['EDUC:educdc'] = acs_data['EDUC'].mul(acs_data['educdc'])
acs_data.head()
```

[10]:

	YEAR	SAMPLE	SERIAL	CBSERIAL	HHWT	CLUSTER	STRATA	GQ
0	2019	201901	2611	2019000016124	19504.8	2019000026111	80001	1
1	2019	201901	6016	2019000247422	1702.8	2019000060161	230001	1
2	2019	201901	6790	2019000301008	8668.8	2019000067901	190001	1
3	2019	201901	9577	2019000497180	42105.6	2019000095771	10001	1
4	2019	201901	9731	2019000507929	7430.4	2019000097311	270201	1

	PERNUM	PERWT	NCHILD	NCHLT5	SEX	AGE	MARST	RACE	RACED	HISPAN
0	1	19659.6	1	1	1	29	1	1	100	0
1	1	1702.8	3	1	1	41	1	1	100	0

```
2       3  20278.8       0       0    1   21      6     1    100       0
3       1  42105.6       0       0    1   20      6     1    100       0
4       1   7275.6       0       0    1   33      6     2    200       0

   HISPAND  EDUC  EDUCD  EMPSTAT  EMPSTATD  INCWAGE  VETSTAT  VETSTATD  educd  \
0        0     8     81        1        10    59000        1        11     81
1        0     8     81        1        10    50000        2        20     81
2        0     8     81        1        10    10000        1        11     81
3        0     8     81        1        10      800        1        11     81
4        0     8     81        1        10    45000        1        11     81

   educdc  hsdip  coldip  white  black  hispanic  married  female  vet  \
0    14.0      0       0      1      0         1        1       0    0
1    14.0      0       0      1      0         1        1       0    1
2    14.0      0       0      1      0         1        0       0    0
3    14.0      0       0      1      0         1        0       0    0
4    14.0      0       0      0      1         1        0       0    0

   EDUC:educdc
0        112.0
1        112.0
2        112.0
3        112.0
4        112.0
```

[11]:
```
#Age Squared
acs_data['AGE^2'] = np.power(acs_data['AGE'],2)
acs_data.head()
```

[11]:
```
   YEAR  SAMPLE  SERIAL       CBSERIAL     HHWT        CLUSTER  STRATA  GQ  \
0  2019  201901    2611  2019000016124  19504.8  2019000026111   80001   1
1  2019  201901    6016  2019000247422   1702.8  2019000060161  230001   1
2  2019  201901    6790  2019000301008   8668.8  2019000067901  190001   1
3  2019  201901    9577  2019000497180  42105.6  2019000095771   10001   1
4  2019  201901    9731  2019000507929   7430.4  2019000097311  270201   1

   PERNUM    PERWT  NCHILD  NCHLT5  SEX  AGE  MARST  RACE  RACED  HISPAN  \
0       1  19659.6       1       1    1   29      1     1    100       0
1       1   1702.8       3       1    1   41      1     1    100       0
2       3  20278.8       0       0    1   21      6     1    100       0
3       1  42105.6       0       0    1   20      6     1    100       0
4       1   7275.6       0       0    1   33      6     2    200       0

   HISPAND  EDUC  EDUCD  EMPSTAT  EMPSTATD  INCWAGE  VETSTAT  VETSTATD  educd  \
0        0     8     81        1        10    59000        1        11     81
1        0     8     81        1        10    50000        2        20     81
2        0     8     81        1        10    10000        1        11     81
```

```
3        0     8     81        1        10      800        1        11     81
4        0     8     81        1        10    45000        1        11     81

   educdc  hsdip  coldip  white  black  hispanic  married  female  vet  \
0    14.0      0       0      1      0         1        1       0    0
1    14.0      0       0      1      0         1        1       0    1
2    14.0      0       0      1      0         1        0       0    0
3    14.0      0       0      1      0         1        0       0    0
4    14.0      0       0      0      1         1        0       0    0

   EDUC:educdc  AGE^2
0        112.0    841
1        112.0   1681
2        112.0    441
3        112.0    400
4        112.0   1089
```

[12]:
```
#log of Wage
acs_data = acs_data[acs_data['INCWAGE'] != 0] #Only one row had a 0 so needed␣
 ↪to remove it
acs_data['LNINCWAGE'] = np.log(acs_data['INCWAGE'])
acs_data.head()
```

```
C:\Users\srsid\anaconda3\lib\site-packages\ipykernel_launcher.py:3:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  This is separate from the ipykernel package so we can avoid doing imports
until
```

[12]:
```
   YEAR  SAMPLE  SERIAL       CBSERIAL     HHWT        CLUSTER  STRATA  GQ  \
0  2019  201901    2611  2019000016124  19504.8  2019000026111   80001   1
1  2019  201901    6016  2019000247422   1702.8  2019000060161  230001   1
2  2019  201901    6790  2019000301008   8668.8  2019000067901  190001   1
3  2019  201901    9577  2019000497180  42105.6  2019000095771   10001   1
4  2019  201901    9731  2019000507929   7430.4  2019000097311  270201   1

   PERNUM    PERWT  NCHILD  NCHLT5  SEX  AGE  MARST  RACE  RACED  HISPAN  \
0       1  19659.6       1       1    1   29      1     1    100       0
1       1   1702.8       3       1    1   41      1     1    100       0
2       3  20278.8       0       0    1   21      6     1    100       0
3       1  42105.6       0       0    1   20      6     1    100       0
4       1   7275.6       0       0    1   33      6     2    200       0

   HISPAND  EDUC  EDUCD  EMPSTAT  EMPSTATD  INCWAGE  VETSTAT  VETSTATD  educd  \
```

```
0        0     8     81        1        10    59000        1        11     81
1        0     8     81        1        10    50000        2        20     81
2        0     8     81        1        10    10000        1        11     81
3        0     8     81        1        10      800        1        11     81
4        0     8     81        1        10    45000        1        11     81

   educdc  hsdip  coldip  white  black  hispanic  married  female  vet  \
0    14.0      0       0      1      0         1        1       0    0
1    14.0      0       0      1      0         1        1       0    1
2    14.0      0       0      1      0         1        0       0    0
3    14.0      0       0      1      0         1        0       0    0
4    14.0      0       0      0      1         1        0       0    0

   EDUC:educdc  AGE^2  LNINCWAGE
0        112.0    841  10.985293
1        112.0   1681  10.819778
2        112.0    441   9.210340
3        112.0    400   6.684612
4        112.0   1089  10.714418
```

1 Data Analysis

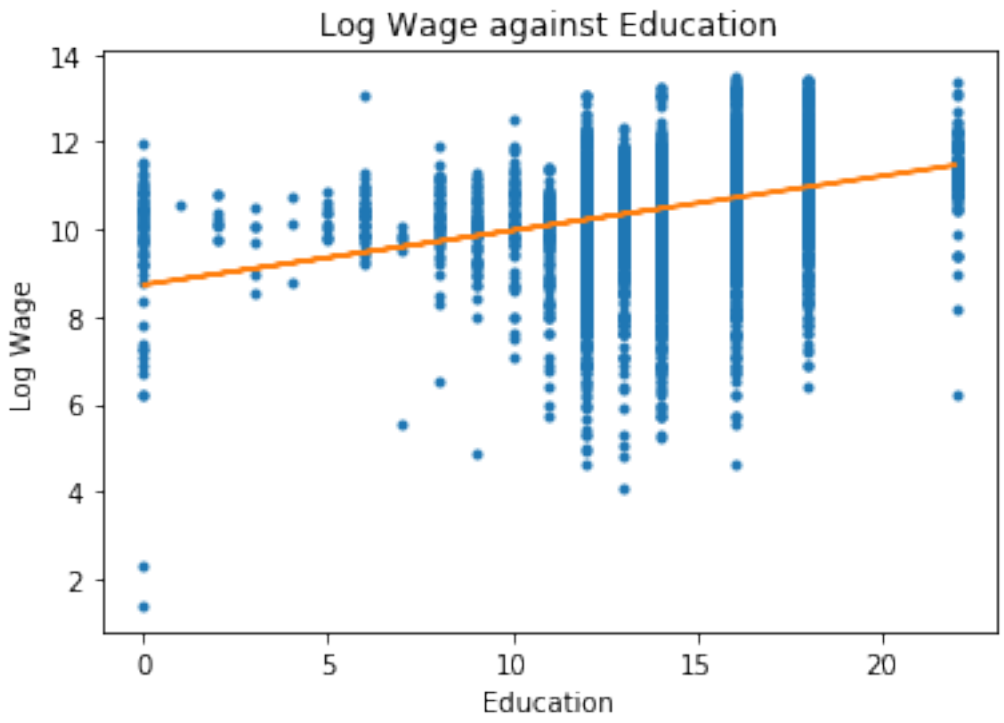
[13]:
```
#Question 1
acs_data[['YEAR','INCWAGE','LNINCWAGE','educdc','female','AGE','AGE^2','white','black','hispan
 ↪educdc']].describe()
```

[13]:
```
          YEAR        INCWAGE    LNINCWAGE       educdc       female  \
count   8606.0    8606.000000  8606.000000  8606.000000  8606.000000
mean    2019.0   58420.063212    10.496826    14.187427     0.486521
std        0.0   68115.268196     1.097734     2.857626     0.499847
min     2019.0       4.000000     1.386294     0.000000     0.000000
25%     2019.0   22725.000000    10.031219    12.000000     0.000000
50%     2019.0   41300.000000    10.628615    14.000000     0.000000
75%     2019.0   70750.000000    11.166889    16.000000     1.000000
max     2019.0  717000.000000    13.482831    22.000000     1.000000

                AGE        AGE^2        white        black  hispanic  \
count  8606.000000  8606.000000  8606.000000  8606.000000    8606.0
mean     41.849059  1931.513479     0.773298     0.090286       1.0
std      13.423513  1126.920775     0.418723     0.286607       0.0
min      18.000000   324.000000     0.000000     0.000000       1.0
25%      30.000000   900.000000     1.000000     0.000000       1.0
50%      42.000000  1764.000000     1.000000     0.000000       1.0
75%      54.000000  2916.000000     1.000000     0.000000       1.0
max      65.000000  4225.000000     1.000000     1.000000       1.0
```

```
           married       NCHILD          vet        hsdip       coldip  \
count  8606.000000  8606.000000  8606.000000  8606.000000  8606.000000
mean      0.525912     0.784801     0.049733     0.210783     0.236812
std       0.499357     1.100708     0.217405     0.407888     0.425150
min       0.000000     0.000000     0.000000     0.000000     0.000000
25%       0.000000     0.000000     0.000000     0.000000     0.000000
50%       1.000000     0.000000     0.000000     0.000000     0.000000
75%       1.000000     1.000000     0.000000     0.000000     0.000000
max       1.000000     9.000000     1.000000     1.000000     1.000000

       EDUC:educdc
count  8606.000000
mean    117.493028
std      51.649895
min       0.000000
25%      72.000000
50%      98.000000
75%     160.000000
max     242.000000
```

[14]:
```
import matplotlib.pyplot as plt
from numpy.polynomial.polynomial import polyfit
```

[16]:
```
#Question 2
x = acs_data['educdc']
y = acs_data['LNINCWAGE']
b, m = polyfit(x, y, 1)
plt.plot(x, y, '.')
plt.plot(x, b + m * x, '-')
plt.xlabel('Education')
plt.ylabel('Log Wage')
plt.title('Log Wage against Education')
plt.show()

#Source: https://stackoverflow.com/questions/19068862/
 ↪how-to-overplot-a-line-on-a-scatter-plot-in-python
```



[18]:
```
import statsmodels.api as sm
```

[19]:
```
#Question 3
X =␣
 ↪acs_data[['educdc','female','AGE','AGE^2','white','black','hispanic','married','NCHILD','ve
y = acs_data['LNINCWAGE']
Regression = sm.OLS(y ,X).fit()
print(Regression.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              LNINCWAGE   R-squared:                       0.309
Model:                            OLS   Adj. R-squared:                  0.308
Method:                 Least Squares   F-statistic:                     426.4
Date:                Mon, 01 Feb 2021   Prob (F-statistic):               0.00
Time:                        18:15:04   Log-Likelihood:                -11425.
No. Observations:                8606   AIC:                         2.287e+04
Df Residuals:                    8596   BIC:                         2.294e+04
Df Model:                           9
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
educdc         0.1112      0.004     31.564      0.000       0.104       0.118
```

```
female        -0.4338      0.020    -21.577      0.000      -0.473      -0.394
AGE            0.1567      0.006     27.646      0.000       0.146       0.168
AGE^2         -0.0016    6.7e-05    -24.270      0.000      -0.002      -0.001
white         -0.0297      0.029     -1.024      0.306      -0.087       0.027
black         -0.1875      0.043     -4.410      0.000      -0.271      -0.104
hispanic       5.6555      0.114     49.594      0.000       5.432       5.879
married        0.1955      0.023      8.450      0.000       0.150       0.241
NCHILD        -0.0063      0.010     -0.616      0.538      -0.026       0.014
vet           -0.0396      0.046     -0.856      0.392      -0.130       0.051
==============================================================================
Omnibus:                     2437.163   Durbin-Watson:                   1.867
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            10300.166
Skew:                          -1.335   Prob(JB):                         0.00
Kurtosis:                       7.647   Cond. No.                     2.60e+04
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The condition number is large, 2.6e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

[20]:
```
import statsmodels.formula.api as smf
result = smf.ols('LNINCWAGE ~ C(AGE)', data = acs_data).fit()
print(result.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              LNINCWAGE   R-squared:                       0.209
Model:                            OLS   Adj. R-squared:                  0.205
Method:                 Least Squares   F-statistic:                     48.16
Date:                Mon, 01 Feb 2021   Prob (F-statistic):               0.00
Time:                        18:15:04   Log-Likelihood:                -12004.
No. Observations:                8606   AIC:                         2.410e+04
Df Residuals:                    8558   BIC:                         2.444e+04
Df Model:                          47
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      8.5613      0.088     96.999      0.000       8.388       8.734
C(AGE)[T.19]   0.1403      0.118      1.192      0.233      -0.090       0.371
C(AGE)[T.20]   0.5995      0.119      5.020      0.000       0.365       0.834
C(AGE)[T.21]   0.8163      0.120      6.803      0.000       0.581       1.052
C(AGE)[T.22]   0.8900      0.113      7.889      0.000       0.669       1.111
C(AGE)[T.23]   1.2769      0.117     10.936      0.000       1.048       1.506
C(AGE)[T.24]   1.5024      0.116     12.901      0.000       1.274       1.731
C(AGE)[T.25]   1.5190      0.115     13.157      0.000       1.293       1.745
```

C(AGE)[T.26]	1.7094	0.115	14.859	0.000	1.484	1.935
C(AGE)[T.27]	1.8766	0.118	15.876	0.000	1.645	2.108
C(AGE)[T.28]	1.8940	0.114	16.649	0.000	1.671	2.117
C(AGE)[T.29]	1.7146	0.113	15.213	0.000	1.494	1.936
C(AGE)[T.30]	2.0496	0.111	18.489	0.000	1.832	2.267
C(AGE)[T.31]	1.9967	0.114	17.552	0.000	1.774	2.220
C(AGE)[T.32]	2.0790	0.115	18.134	0.000	1.854	2.304
C(AGE)[T.33]	1.9255	0.116	16.658	0.000	1.699	2.152
C(AGE)[T.34]	2.1251	0.112	18.928	0.000	1.905	2.345
C(AGE)[T.35]	1.9479	0.115	16.892	0.000	1.722	2.174
C(AGE)[T.36]	2.0875	0.112	18.593	0.000	1.867	2.308
C(AGE)[T.37]	2.2044	0.117	18.855	0.000	1.975	2.434
C(AGE)[T.38]	2.1583	0.114	18.868	0.000	1.934	2.382
C(AGE)[T.39]	2.3228	0.112	20.689	0.000	2.103	2.543
C(AGE)[T.40]	2.1019	0.115	18.313	0.000	1.877	2.327
C(AGE)[T.41]	2.1998	0.114	19.295	0.000	1.976	2.423
C(AGE)[T.42]	2.1307	0.114	18.770	0.000	1.908	2.353
C(AGE)[T.43]	2.1793	0.115	18.988	0.000	1.954	2.404
C(AGE)[T.44]	2.0735	0.115	17.960	0.000	1.847	2.300
C(AGE)[T.45]	2.2876	0.115	19.931	0.000	2.063	2.513
C(AGE)[T.46]	2.2064	0.115	19.134	0.000	1.980	2.432
C(AGE)[T.47]	2.1709	0.115	18.870	0.000	1.945	2.396
C(AGE)[T.48]	2.2629	0.115	19.738	0.000	2.038	2.488
C(AGE)[T.49]	2.3234	0.115	20.196	0.000	2.098	2.549
C(AGE)[T.50]	2.3618	0.113	20.975	0.000	2.141	2.583
C(AGE)[T.51]	2.2763	0.112	20.389	0.000	2.057	2.495
C(AGE)[T.52]	2.2442	0.112	19.989	0.000	2.024	2.464
C(AGE)[T.53]	2.1435	0.112	19.217	0.000	1.925	2.362
C(AGE)[T.54]	2.2303	0.111	20.085	0.000	2.013	2.448
C(AGE)[T.55]	2.1614	0.114	19.040	0.000	1.939	2.384
C(AGE)[T.56]	2.1321	0.108	19.776	0.000	1.921	2.343
C(AGE)[T.57]	2.3128	0.114	20.373	0.000	2.090	2.535
C(AGE)[T.58]	2.1994	0.111	19.736	0.000	1.981	2.418
C(AGE)[T.59]	2.1471	0.114	18.792	0.000	1.923	2.371
C(AGE)[T.60]	2.1243	0.116	18.355	0.000	1.897	2.351
C(AGE)[T.61]	2.0855	0.113	18.448	0.000	1.864	2.307
C(AGE)[T.62]	2.1106	0.116	18.192	0.000	1.883	2.338
C(AGE)[T.63]	1.9634	0.120	16.336	0.000	1.728	2.199
C(AGE)[T.64]	1.9695	0.120	16.439	0.000	1.735	2.204
C(AGE)[T.65]	2.0175	0.126	16.063	0.000	1.771	2.264

Omnibus:	2106.747	Durbin-Watson:	1.748
Prob(Omnibus):	0.000	Jarque-Bera (JB):	8851.648
Skew:	-1.147	Prob(JB):	0.00
Kurtosis:	7.407	Cond. No.	59.0

Warnings:

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

[21]:
```
d = {'AGE': [50]}
df = pd.DataFrame(data=d)
predictions = result.get_prediction(df)
predictions.summary_frame(alpha=0.05)
```

[21]:
```
       mean  mean_se  mean_ci_lower  mean_ci_upper  obs_ci_lower  \
0  10.923088  0.06992      10.786029      11.060148       8.99937

   obs_ci_upper
0     12.846806
```

[22]:
```
import statsmodels.formula.api as smf
result = smf.ols('LNINCWAGE ~ C(SEX)', data = acs_data).fit()
print(result.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              LNINCWAGE   R-squared:                       0.034
Model:                            OLS   Adj. R-squared:                  0.034
Method:                 Least Squares   F-statistic:                     300.0
Date:                Mon, 01 Feb 2021   Prob (F-statistic):           4.38e-66
Time:                        18:15:09   Log-Likelihood:                -12866.
No. Observations:                8606   AIC:                         2.574e+04
Df Residuals:                    8604   BIC:                         2.575e+04
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      10.6929      0.016    658.686      0.000      10.661      10.725
C(SEX)[T.2]    -0.4031      0.023    -17.320      0.000      -0.449      -0.357
==============================================================================
Omnibus:                     1958.332   Durbin-Watson:                   1.575
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             6348.921
Skew:                          -1.148   Prob(JB):                         0.00
Kurtosis:                       6.527   Cond. No.                         2.59
==============================================================================

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

[23]:
```
d = {'SEX': [1]}
df = pd.DataFrame(data=d)
predictions = result.get_prediction(df)
```

```
predictions.summary_frame(alpha=0.05)
```

[23]:
```
        mean   mean_se  mean_ci_lower  mean_ci_upper  obs_ci_lower  \
0  10.69294  0.016234      10.661118      10.724762      8.577313

   obs_ci_upper
0     12.808567
```

[24]:
```
d = {'SEX': [2]}
df = pd.DataFrame(data=d)
predictions = result.get_prediction(df)
predictions.summary_frame(alpha=0.05)
```

[24]:
```
        mean   mean_se  mean_ci_lower  mean_ci_upper  obs_ci_lower  \
0  10.289845  0.016677      10.257153      10.322536      8.174204

   obs_ci_upper
0     12.405485
```

[25]:
```
result = smf.ols('LNINCWAGE ~ RACE', data = acs_data).fit()
print(result.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              LNINCWAGE   R-squared:                       0.001
Model:                            OLS   Adj. R-squared:                  0.001
Method:                 Least Squares   F-statistic:                     12.56
Date:                Mon, 01 Feb 2021   Prob (F-statistic):           0.000397
Time:                        18:15:11   Log-Likelihood:                -13007.
No. Observations:                8606   AIC:                         2.602e+04
Df Residuals:                    8604   BIC:                         2.603e+04
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     10.5374      0.016    639.840      0.000      10.505      10.570
RACE          -0.0225      0.006     -3.543      0.000      -0.035      -0.010
==============================================================================
Omnibus:                     1902.558   Durbin-Watson:                   1.627
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             6042.310
Skew:                          -1.122   Prob(JB):                         0.00
Kurtosis:                       6.437   Cond. No.                         3.89
==============================================================================

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

[26]:
```
import statsmodels.formula.api as smf
result = smf.ols('LNINCWAGE ~ C(educdc)', data = acs_data).fit()
print(result.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              LNINCWAGE   R-squared:                       0.135
Model:                            OLS   Adj. R-squared:                  0.133
Method:                 Least Squares   F-statistic:                     78.82
Date:                Mon, 01 Feb 2021   Prob (F-statistic):          1.59e-254
Time:                        18:15:12   Log-Likelihood:                -12389.
No. Observations:                8606   AIC:                         2.481e+04
Df Residuals:                    8588   BIC:                         2.494e+04
Df Model:                          17
Covariance Type:            nonrobust
================================================================================
=====
                     coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
-----
Intercept              9.6034      0.120     80.286      0.000       9.369
9.838
C(educdc)[T.1.0]       0.9420      1.029      0.915      0.360      -1.075
2.959
C(educdc)[T.2.0]       0.6267      0.361      1.736      0.083      -0.081
1.334
C(educdc)[T.3.0]       0.0355      0.434      0.082      0.935      -0.815
0.886
C(educdc)[T.4.0]       0.2870      0.602      0.477      0.634      -0.893
1.467
C(educdc)[T.5.0]       0.5971      0.331      1.806      0.071      -0.051
1.245
C(educdc)[T.6.0]       0.7069      0.194      3.650      0.000       0.327
1.087
C(educdc)[T.7.0]      -0.5038      0.434     -1.161      0.246      -1.355
0.347
C(educdc)[T.8.0]       0.6059      0.184      3.285      0.001       0.244
0.967
C(educdc)[T.9.0]       0.2727      0.177      1.538      0.124      -0.075
0.620
C(educdc)[T.10.0]      0.5957      0.166      3.579      0.000       0.269
0.922
C(educdc)[T.11.0]     -0.0094      0.155     -0.061      0.951      -0.314
0.295
C(educdc)[T.12.0]      0.5666      0.122      4.661      0.000       0.328
0.805
C(educdc)[T.13.0]      0.5977      0.127      4.721      0.000       0.349
```

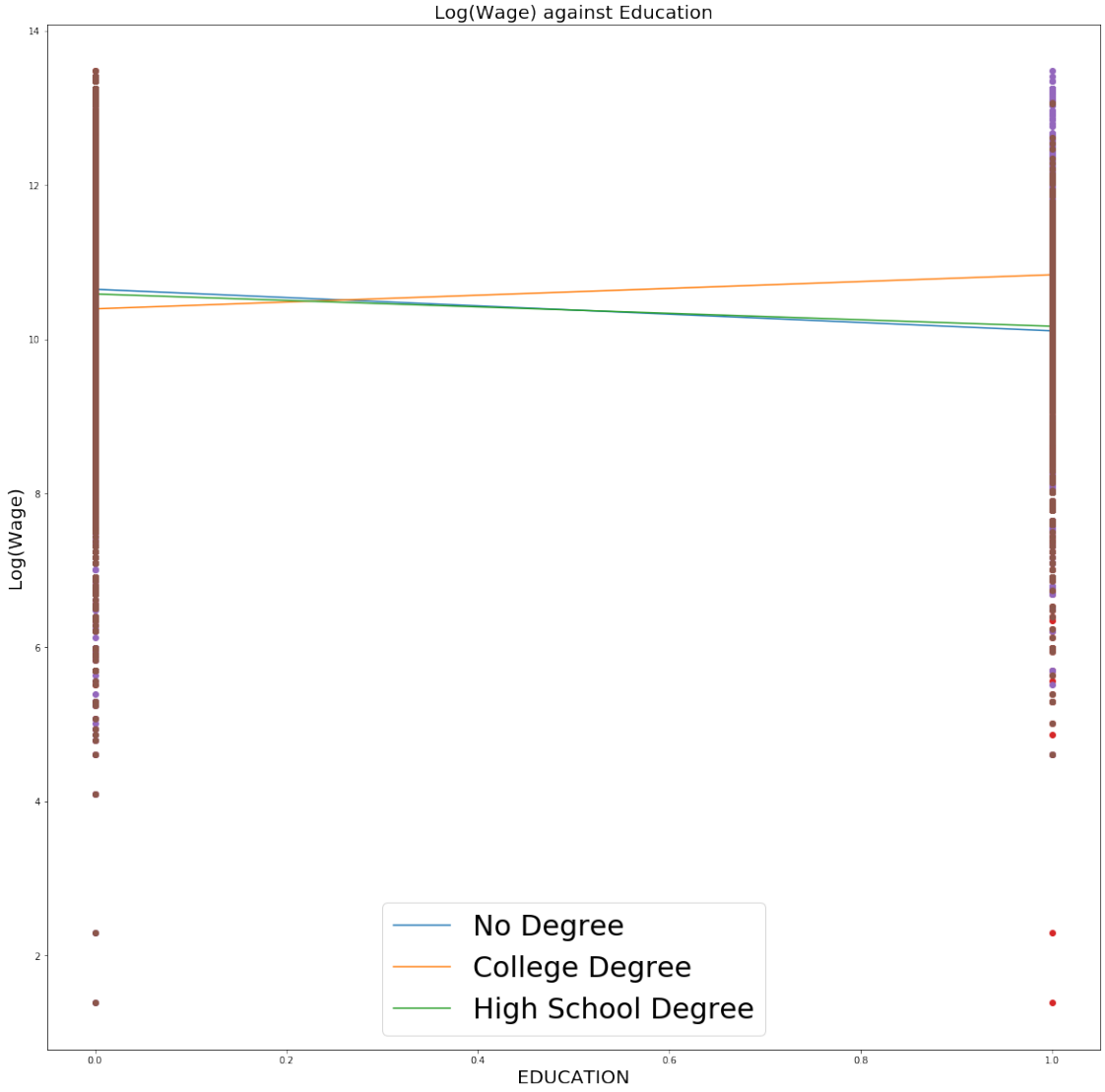
```
0.846
C(educdc)[T.14.0]      0.7022      0.122      5.767      0.000       0.463
0.941
C(educdc)[T.16.0]      1.2316      0.122     10.117      0.000       0.993
1.470
C(educdc)[T.18.0]      1.5652      0.123     12.684      0.000       1.323
1.807
C(educdc)[T.22.0]      1.7243      0.150     11.520      0.000       1.431
2.018
==============================================================================
Omnibus:                     2264.267   Durbin-Watson:                   1.880
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             7813.239
Skew:                          -1.306   Prob(JB):                         0.00
Kurtosis:                       6.869   Cond. No.                         103.
==============================================================================

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

[27]:
```
import seaborn as sns
```

[164]:
```
#Question 4


x = acs_data['EDUCD'] <=63
x1 = acs_data['coldip']
x2 = acs_data['hsdip']
y= acs_data['LNINCWAGE']

plt.figure(figsize=(20, 20))
plt.plot(np.unique(x), np.poly1d(np.polyfit(x, y, 1))(np.unique(x)))
plt.plot(np.unique(x1), np.poly1d(np.polyfit(x1, y, 1))(np.unique(x1)))
plt.plot(np.unique(x2), np.poly1d(np.polyfit(x2, y, 1))(np.unique(x2)))

m, b = np.polyfit(x, y, 1)
m1, b1 = np.polyfit(x1, y, 1)
m2, b2 = np.polyfit(x2, y, 1)
plt.plot(x, y, 'o')
plt.plot(x1, y, 'o')
plt.plot(x2, y, 'o')


plt.legend(labels=['No Degree','College Degree', 'High School␣
 ↪Degree'],prop={'size': 30})
plt.xlabel('EDUCATION', size = 20)
plt.ylabel('Log(Wage)', size = 20)
```

```
plt.title("Log(Wage) against Education", size = 20)

#Source
#https://stackoverflow.com/questions/22239691/
 ↪code-for-best-fit-straight-line-of-a-scatter-plot-in-python
#https://towardsdatascience.com/
 ↪simple-and-multiple-linear-regression-with-python-c9ab422ec29c
#https://stackoverflow.com/questions/7125009/
 ↪how-to-change-legend-size-with-matplotlib-pyplot
```

[164]: Text(0.5, 1.0, 'Log(Wage) against Education')



[109]:
```
#Question 6
import statsmodels.formula.api as smf
X =␣
 ↪acs_data[['female','AGE','AGE^2','white','black','hispanic','married','NCHILD','vet']]
predictors = ' + '.join(X)
result = smf.ols('LNINCWAGE ~ {} + (EDUCD==61)+hsdip+coldip'.
 ↪format(predictors), data = acs_data).fit()
print(result.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              LNINCWAGE   R-squared:                       0.205
Model:                            OLS   Adj. R-squared:                  0.204
Method:                 Least Squares   F-statistic:                     202.0
Date:                Mon, 01 Feb 2021   Prob (F-statistic):               0.00
Time:                        19:08:06   Log-Likelihood:                -12024.
No. Observations:                8606   AIC:                         2.407e+04
Df Residuals:                    8594   BIC:                         2.416e+04
Df Model:                          11
Covariance Type:            nonrobust
==============================================================================
=======
                           coef    std err          t      P>|t|      [0.025
0.975]
------------------------------------------------------------------------------
-------
Intercept                4.8096      0.023    210.864      0.000       4.765
4.854
EDUCD == 61[T.True]     -0.5766      0.091     -6.339      0.000      -0.755
-0.398
female                  -0.4154      0.022    -19.289      0.000      -0.458
-0.373
AGE                      0.0156      0.005      2.882      0.004       0.005
0.026
AGE ^ 2                  0.0052      0.005      0.990      0.322      -0.005
0.016
white                    0.0090      0.031      0.288      0.773      -0.052
0.070
black                   -0.1334      0.046     -2.925      0.003      -0.223
-0.044
hispanic                 4.8096      0.023    210.864      0.000       4.765
4.854
married                  0.2953      0.025     11.983      0.000       0.247
0.344
NCHILD                   0.0600      0.010      5.824      0.000       0.040
0.080
vet                     -0.0192      0.050     -0.388      0.698      -0.117
0.078
```

```
hsdip                -0.3009      0.027    -11.038      0.000      -0.354
-0.247
coldip                0.3486      0.026     13.318      0.000       0.297
0.400
==============================================================================
Omnibus:                     2375.306   Durbin-Watson:                   1.724
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            10218.708
Skew:                          -1.293   Prob(JB):                         0.00
Kurtosis:                       7.670   Cond. No.                     2.88e+17
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 4.03e-28. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

[162]:
```
import statsmodels.formula.api as smf

d = {'EDUCD':0,'female':1,'AGE':22,'AGE^2':22, 'white':0, 'black':0,'hispanic':
 ↪0, 'married':0,'NCHILD':0,'vet':0,
    'hsdip':1,'coldip':0}
df = pd.DataFrame([d])
predictions = result.get_prediction(df)
predictions.summary_frame(alpha=0.05)
```

[162]:
```
       mean   mean_se  mean_ci_lower  mean_ci_upper  obs_ci_lower  \
0  4.541411  0.034663       4.473464       4.609358      2.620911

   obs_ci_upper
0      6.461911
```

[112]:
```
#Question 7
d = {'EDUCD':0,'female':1,'AGE':22,'AGE^2':22, 'white':0, 'black':0,'hispanic':
 ↪0, 'married':0,'NCHILD':0,'vet':0,
     'hsdip':0,'coldip':1}
df = pd.DataFrame([d])
predictions = result.get_prediction(df)
predictions.summary_frame(alpha=0.05)
```

[112]:
```
       mean   mean_se  mean_ci_lower  mean_ci_upper  obs_ci_lower  \
0  5.190984  0.033371       5.125568         5.2564      3.270572

   obs_ci_upper
0      7.111395
```

[163]:
```
#Question 7
import statsmodels.formula.api as smf

d = {'EDUCD':0,'female':0,'AGE':22,'AGE^2':22, 'white':0, 'black':0,'hispanic':
 ↪0, 'married':0,'NCHILD':0,'vet':0,
     'hsdip':1,'coldip':0}
df = pd.DataFrame([d])
predictions = result.get_prediction(df)
predictions.summary_frame(alpha=0.05)
```

[163]:

	mean	mean_se	mean_ci_lower	mean_ci_upper	obs_ci_lower	obs_ci_upper
0	4.956786	0.030338	4.897315	5.016256	3.036567	6.877004

[113]:
```
#Question 7
d = {'EDUCD':0,'female':0,'AGE':22,'AGE^2':22, 'white':0, 'black':0,'hispanic':
 ↪0, 'married':0,'NCHILD':0,'vet':0,
     'hsdip':0,'coldip':1}
df = pd.DataFrame([d])
predictions = result.get_prediction(df)
predictions.summary_frame(alpha=0.05)
```

[113]:

	mean	mean_se	mean_ci_lower	mean_ci_upper	obs_ci_lower	obs_ci_upper
0	5.606358	0.030472	5.546626	5.666091	3.686132	7.526585