km5188_ds4e_hw4

November 22, 2022

1 DS4E: Homework 4

[2]:
```
pip install pandas
```

```
Requirement already satisfied: pandas in /opt/conda/lib/python3.10/site-packages
(1.5.1)
Requirement already satisfied: numpy>=1.21.0 in /opt/conda/lib/python3.10/site-
packages (from pandas) (1.23.2)
Requirement already satisfied: pytz>=2020.1 in /opt/conda/lib/python3.10/site-
packages (from pandas) (2022.2.1)
Requirement already satisfied: python-dateutil>=2.8.1 in
/opt/conda/lib/python3.10/site-packages (from pandas) (2.8.2)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.10/site-
packages (from python-dateutil>=2.8.1->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

[3]:
```
pip install statsmodels
```

```
Requirement already satisfied: statsmodels in /opt/conda/lib/python3.10/site-
packages (0.13.5)
Requirement already satisfied: packaging>=21.3 in
/opt/conda/lib/python3.10/site-packages (from statsmodels) (21.3)
Requirement already satisfied: pandas>=0.25 in /opt/conda/lib/python3.10/site-
packages (from statsmodels) (1.5.1)
Requirement already satisfied: scipy>=1.3 in /opt/conda/lib/python3.10/site-
packages (from statsmodels) (1.9.0)
Requirement already satisfied: numpy>=1.17 in /opt/conda/lib/python3.10/site-
packages (from statsmodels) (1.23.2)
Requirement already satisfied: patsy>=0.5.2 in /opt/conda/lib/python3.10/site-
packages (from statsmodels) (0.5.3)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
/opt/conda/lib/python3.10/site-packages (from packaging>=21.3->statsmodels)
(3.0.9)
Requirement already satisfied: pytz>=2020.1 in /opt/conda/lib/python3.10/site-
packages (from pandas>=0.25->statsmodels) (2022.2.1)
Requirement already satisfied: python-dateutil>=2.8.1 in
/opt/conda/lib/python3.10/site-packages (from pandas>=0.25->statsmodels) (2.8.2)
Requirement already satisfied: six in /opt/conda/lib/python3.10/site-packages
```

```
(from patsy>=0.5.2->statsmodels) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

[4]:
```
# import libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import statsmodels.formula.api as smf
```

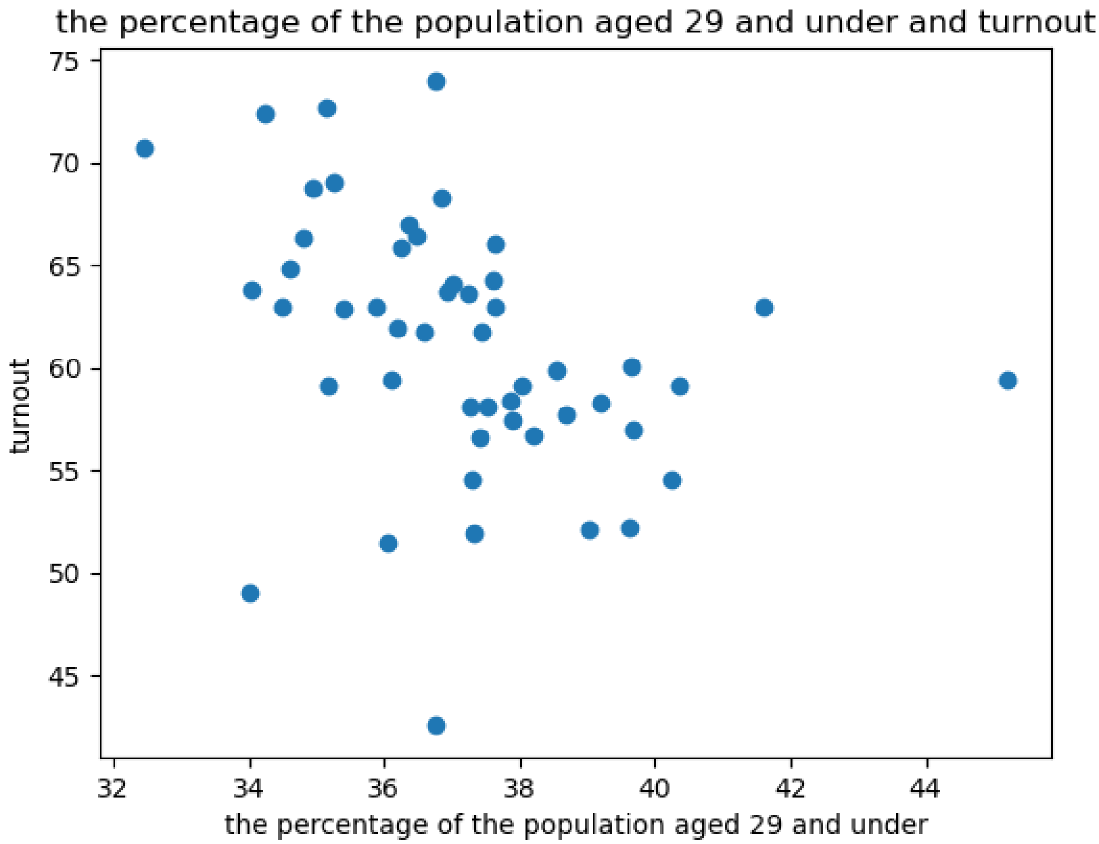
1.1 Question 1

1(a)

[5]:
```
df = pd.read_csv("election_2016.csv")
# df
df.head()
```

[5]:
```
        state stateid      cvap    turnout  age29andunder_pct  \
0     Alabama      AL   3639505  58.342192          37.864079
1     Arizona      AZ   4613575  56.978720          39.687833
2    Arkansas      AR   2175330  51.941361          37.333503
3  California      CA  24582600  57.689565          38.681232
4    Colorado      CO   3824445  72.696038          35.154120

   age65andolder_pct  median_hh_inc  lesscollege_pct
0          16.930066      38.834925        83.080870
1          18.951752      44.166533        80.589436
2          18.258998      37.503720        84.499622
3          15.962776      58.091241        73.988558
4          17.294236      52.243594        69.555890
```

[6]:
```
plt.scatter(df['age29andunder_pct'], df['turnout'])
plt.xlabel('the percentage of the population aged 29 and under')
plt.ylabel('turnout')
plt.title('the percentage of the population aged 29 and under and turnout')
plt.show()
```

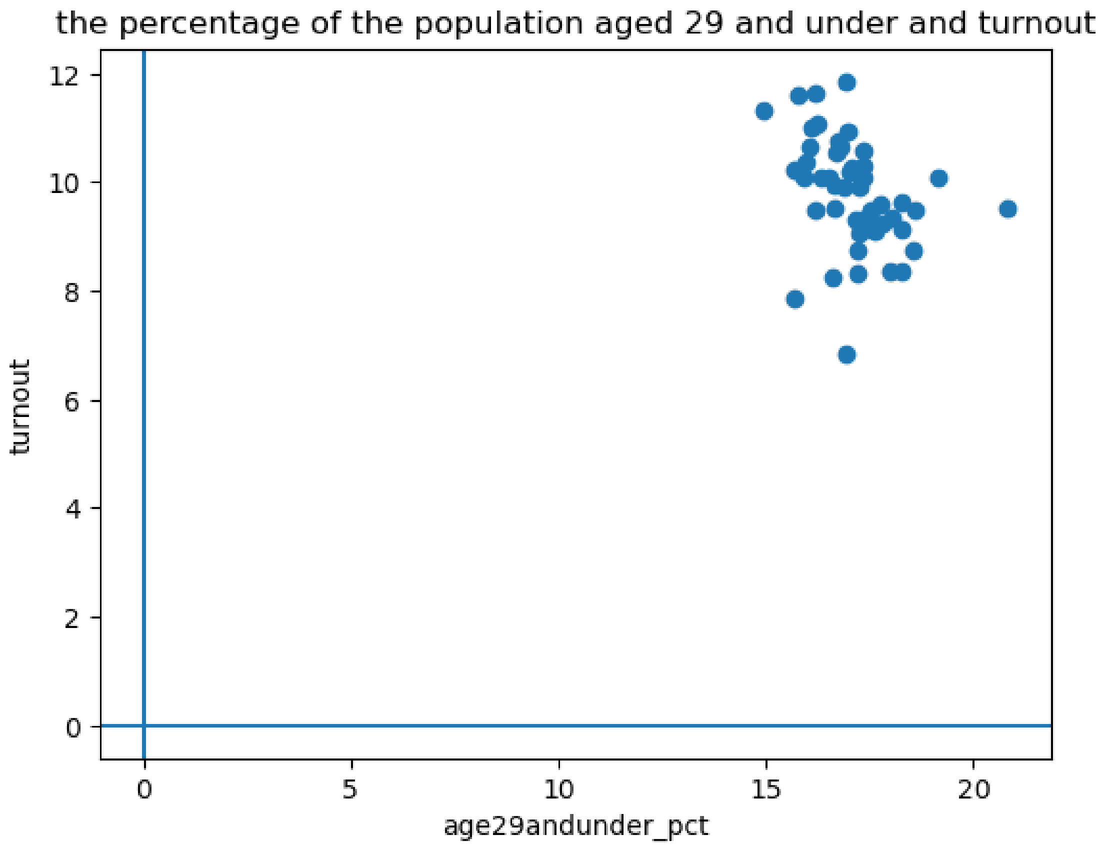


1(b)

[7]:
```
def standardize(var):
    mean = np.mean(var)
    sqr_sum = 0
    for row_val in var:
        sqr_sum += (row_val-mean)**2
    N = var.count()
    std = (sqr_sum/N)**(1/2)
    var = var/std
    return var
df['turnout']=standardize(df['turnout'])
df['age29andunder_pct']=standardize(df['age29andunder_pct'])
# df.head()
```

[12]:
```
plt.scatter(df['age29andunder_pct'], df['turnout'])
plt.xlabel('age29andunder_pct')
plt.ylabel('turnout')
plt.title('the percentage of the population aged 29 and under and turnout')
plt.axvline(x=0)
```

```
plt.axhline(y=0)
plt.show()
```



After standardization, there are no changes either in the direction or in the strength of the relationship, which is expected, since standardization does not affect correlation.If we remove the lines x = 0 and y=0, the graph will be the same as the previous one for non-standardized values.

1(c)

[13]:
```
def correlation(x,y):
    n = x.count()
    denom = n*sum(x*y)-sum(x)*sum(y)
    nom = ((n*sum(x**2)-sum(x)**2)*(n*sum(y**2)-sum(y)**2))**(1/2)

    r = denom/nom
    return r

correlation(df['age29andunder_pct'],df['turnout'])
```

[13]: -0.35687306231858257

1(d)

[15]:
```
df2=df[['age29andunder_pct','turnout']]
df2.corr()
```

[15]:

	age29andunder_pct	turnout
age29andunder_pct	1.000000	-0.356873
turnout	-0.356873	1.000000

The numbers on the diagonal represent the correlation of the variable with itself. Since each variable is perfectly positively correlated with itself, we have 1.0 diagonally.

1(e)

Correlation fallacy. Making conclusions about the causal effect based solely on the relationship between variables. Association does not imply causation.

1.2 Question 2

2(a)

[16]:
```
election = pd.read_csv("election_2016.csv")
```

[22]:
```
plt.figure(figsize = (20,10))
stateid = election['stateid']
turnout = election['turnout']
under29 = election['age29andunder_pct']

for i, typep in enumerate(stateid):
    y = turnout[i]
    x = under29[i]
    plt.scatter(x, y, marker='.', color='purple', s=100)
    plt.text(x,y,typep,fontsize = 15)

plt.grid()
plt.xlabel('the percentage of the population aged 29 and under', fontsize=18)
plt.ylabel('turnout', fontsize=18)
plt.title('the percentage of the population aged 29 and under and turnout')
plt.show()
```

2(b)

i. UT
ii. MN
iii. ME
iv. HI

2(c)

[18]:
```
results = smf.ols('turnout ~ age29andunder_pct', data = election).fit()
results.summary()
```

[18]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:                turnout   R-squared:                       0.127
Model:                            OLS   Adj. R-squared:                  0.109
Method:                 Least Squares   F-statistic:                     7.005
Date:                Tue, 22 Nov 2022   Prob (F-statistic):             0.0110
Time:                        17:36:01   Log-Likelihood:                -159.09
No. Observations:                  50   AIC:                             322.2
Df Residuals:                      48   BIC:                             326.0
Df Model:                           1
Covariance Type:            nonrobust
================================================================================
=====
                        coef    std err          t      P>|t|      [0.025
0.975]
```

```
--------------------------------------------------------------------------------
-----
Intercept            99.2005     14.422      6.879      0.000      70.204
128.197
age29andunder_pct    -1.0259      0.388     -2.647      0.011      -1.805
-0.247
==============================================================================
Omnibus:                        9.802   Durbin-Watson:                   2.127
Prob(Omnibus):                  0.007   Jarque-Bera (JB):                9.875
Skew:                          -0.813   Prob(JB):                      0.00717
Kurtosis:                       4.447   Cond. No.                         638.
==============================================================================

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

2(d)

Intercept is 99.2005, coefficient is -1.0259

2(e)

The intercept tells us the value of DV when IV is zero. A one-unit increase in the DV is associated with a -1.0259 increase in GDP per capita.

2(f)

P(abs(t)>-1.0259) = 0.011 P(t>-1.0259) = 0.011/2 = 0.0055 At 0.05 level, our p value is statistically significant, since our p value is less than the level of significance. So we can reject the null hypothesis, and say that there is a sufficient evidence to conclude there is a relationship between the percentage of population aged 29 and under and turnout.

2(g)

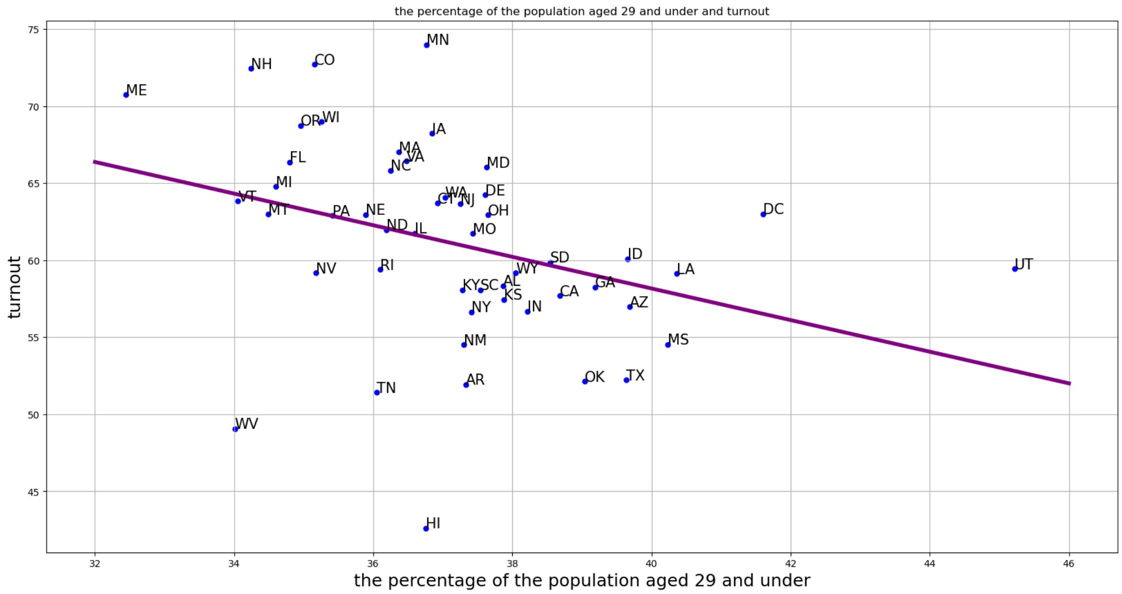
R^2 from the regression model, which is 0.127

1.3 Question 3

3(a)

[19]:
```python
def standard_units(any_numbers):
    "Convert any array of numbers to standard units."
    return (any_numbers - np.mean(any_numbers)) / np.std(any_numbers)


def correlation(t, x, y):
    return np.mean(standard_units(t[x]) * standard_units(t[y]))
```

```python
def slope(t, label_x, label_y):
    r = correlation(t, label_x, label_y)
    return r * np.std(t[label_y]) / np.std(t[label_x])


def intercept(t, label_x, label_y):
    return np.mean(t[label_y]) - slope(t, label_x, label_y) * np.
 ↪mean(t[label_x])


intercept = intercept(election, 'age29andunder_pct', 'turnout')
slope = slope(election, 'age29andunder_pct', 'turnout')

xx = ([32, 46])
yy = ([(intercept + (slope * 32)), (intercept + (slope * 46))])

plt.figure(figsize=(20, 10))
stateid = election['stateid']
turnout = election['turnout']
under29 = election['age29andunder_pct']

for i, typep in enumerate(stateid):
    y = turnout[i]
    x = under29[i]
    plt.scatter(x, y, marker='.', color='blue', s=100)
    plt.text(x, y, typep, fontsize=15)

plt.grid()
plt.xlabel('the percentage of the population aged 29 and under', fontsize=18)
plt.ylabel('turnout', fontsize=18)
plt.title('the percentage of the population aged 29 and under and turnout')

plt.plot(xx, yy, color='purple', linewidth=4)

plt.show()
```



3(b)

```
[16]: turnout = intercept + (slope * 40)
      turnout
```

```
[16]: 58.16264428378027
```

3(c)

```
[18]: # i didn't know if the input should be the state name or the percentage
      def turnout_predict_by_state(state):
          state_29pct = election.loc[election['stateid']==state]['age29andunder_pct']
          turnout = intercept + (slope * float(state_29pct))
          return turnout

      def turnout_predict_by_pct(pct):
          turnout = intercept + (slope * float(pct))
          return turnout

      print("New York turnout prediction:", turnout_predict_by_state('NY'))
      print("Texas turnout prediction:", turnout_predict_by_state('TX'))
      print("West Virginia turnout prediction:", turnout_predict_by_state('WV'))
```

```
New York turnout prediction: 60.823071758642
Texas turnout prediction: 58.537510986808215
West Virginia turnout prediction: 64.31005226668046
```

3(d)

[19]:
```python
def observed_turnout(state):
    observed = election.loc[election['stateid']==state]['turnout']
    return float(observed)
print('NY:',turnout_predict_by_state('NY')-observed_turnout('NY'))
print('TX:',turnout_predict_by_state('TX')-observed_turnout('TX'))
print('WV:',turnout_predict_by_state('WV')-observed_turnout('WV'))
```

```
NY: 4.175672532353026
TX: 6.322908498221324
WV: 15.242746096202687
```

3(e)

[20]:
```python
def pct_predict_by_turnout(turnout):
    pct = (turnout - intercept) / slope
    return pct
pct_predict_by_turnout(80)
```

[20]: 18.714895038525775

1.4 Question 4

4(a)

[21]:
```python
residuals = []
for row in election['turnout']:
    observed = row
    pct  = float(election.loc[election['turnout']==row]['age29andunder_pct'])
    predicted = turnout_predict_by_pct(pct)
    residual = observed - predicted
    residuals.append(residual)
election['residuals'] = residuals
election.head()
```

[21]:

	state	stateid	cvap	turnout	age29andunder_pct	\
0	Alabama	AL	3639505	58.342192	37.864079	
1	Arizona	AZ	4613575	56.978720	39.687833	
2	Arkansas	AR	2175330	51.941361	37.333503	
3	California	CA	24582600	57.689565	38.681232	
4	Colorado	CO	3824445	72.696038	35.154120	

	age65andolder_pct	median_hh_inc	lesscollege_pct	residuals
0	16.930066	38.834925	83.080870	-2.011791
1	18.951752	44.166533	80.589436	-1.504190
2	18.258998	37.503720	84.499622	-8.956964
3	15.962776	58.091241	73.988558	-1.826063
4	17.294236	52.243594	69.555890	9.561785

[22]:
```python
plt.figure(figsize=(20, 10))
stateid = election['stateid']
residuals = election['residuals']
under29 = election['age29andunder_pct']

for i, typep in enumerate(stateid):
    y = residuals[i]
    x = under29[i]
    plt.scatter(x, y, marker='.', color='red', s=100)
    plt.text(x, y, typep, fontsize=15)

plt.grid()
plt.axhline(y=0)
plt.xlabel('the percentage of the population aged 29 and under', fontsize=18)
plt.ylabel('residuals', fontsize=18)
plt.title('plot of the model residuals')
plt.show()
```

Except some outliers, such as DC and UT, in general, as the percentage of the population aged 29 and under increases, the absolute value of the residuals decrease. This means that as the proportion of the population aged 29 and younger increases, our model predicts values more accurately. Thus, for a smaller percentage of people under the age of 29, the model both underestimates and overestimates the value, while for larger values it generally becomes more accurate.

4(b)

[25]:
```python
print("PA residual:", float(election.
↪loc[election['stateid']=='PA']['residuals']))
```

```
PA residual: 0.011406884789757044
```

4(c)

```
[36]: max_res= max(election['residuals'])
min_res = min(election['residuals'])
stateid_max = election.loc[election['residuals']==max_res]['stateid']
stateid_min = election.loc[election['residuals']==min_res]['stateid']
print(stateid_max)
print(stateid_min)
```

```
22    MN
Name: stateid, dtype: object
10    HI
Name: stateid, dtype: object
```

largest positive residual - Minnesota

largest negative residual - Hawaii

1.5 Question 5

5(a)

In this study, "users will not be informed that an experiment is being conducted," which can be problematic because the subjects must give informed consent to the study. In addition, the study could be treated sexist, in the sense that vaccination gives people more freedom to work and travel. Reminding only men of the need for vaccination can affect the dynamics of gender power in society (regardless of how large or small it is).

5(b)

At least this short text doesn't say that a data scientist is trying to increase vaccination rates. What if vaccination reminders reduce vaccination rates? In this case, the researcher harms people for the sake of his experiment. In addition, there is a certain proportion of the population for whom vaccination can have a negative impact on their health. Therefore, the researcher should take into account such nuances so as not to violate the principles of charity.

5(c)

Once again, assuming that the proposed experiment will benefit society, since its benefits are unfairly distributed among men/women, this may lead to a violation of the principle of justice.

5(d)

It is unclear in this text how the researcher collects the data. In addition, since they do not ask the user's permission to collect and use their data. This may violate the principle of respect for the law and public interests.