Homework 3

April 6, 2023

1a) The researcher is conceptualizing political engagement of NYU students by likelihood of NYU students voting (voter turnout of nyu students).

1b) The researcher is operationalizing political engagement of NYU students by asking them to fill out a survey on if they are going to vote and collecting that data(with options: yes, no, or maybe)

1c) Filling out a yes instead of a no can be a random error in this study.

1d) This is due to response bias, which may occur due to social desiriabality, memory and other multitude of reasons. This bias will likely in the diection of more yeses as not voting is often frowned upon, and due to social desiriability, a person might vote yes.

1e) Only students from Data Science for Everyone are asked to fill out the survey for a study of voter turnout for all NYU students, this selection of a particular class causes a selection bias

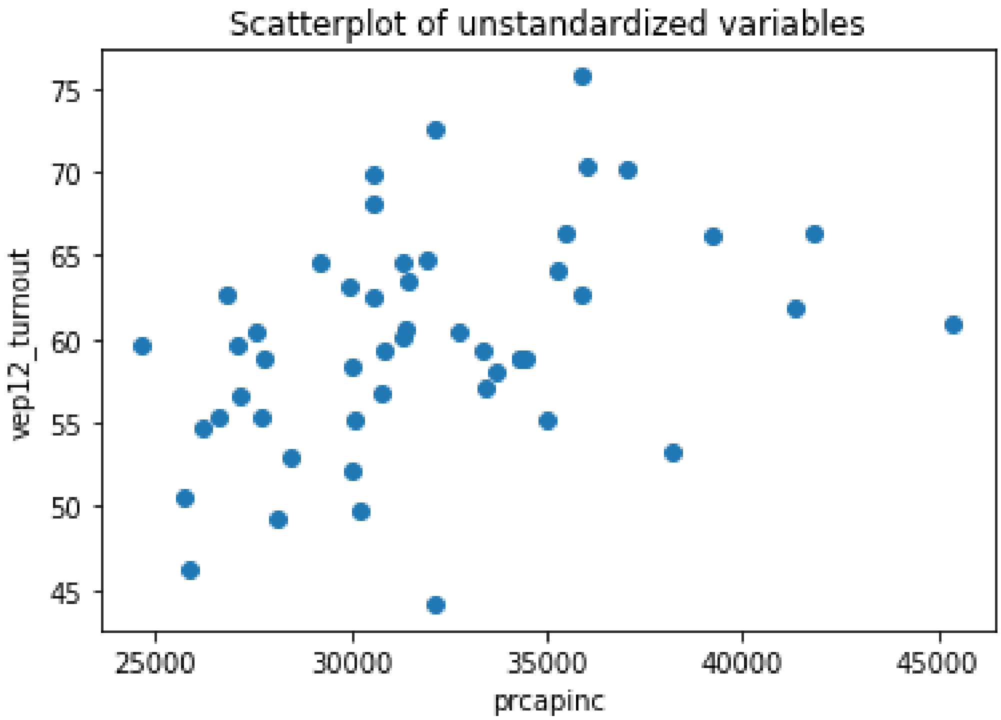
1f)Error of validity

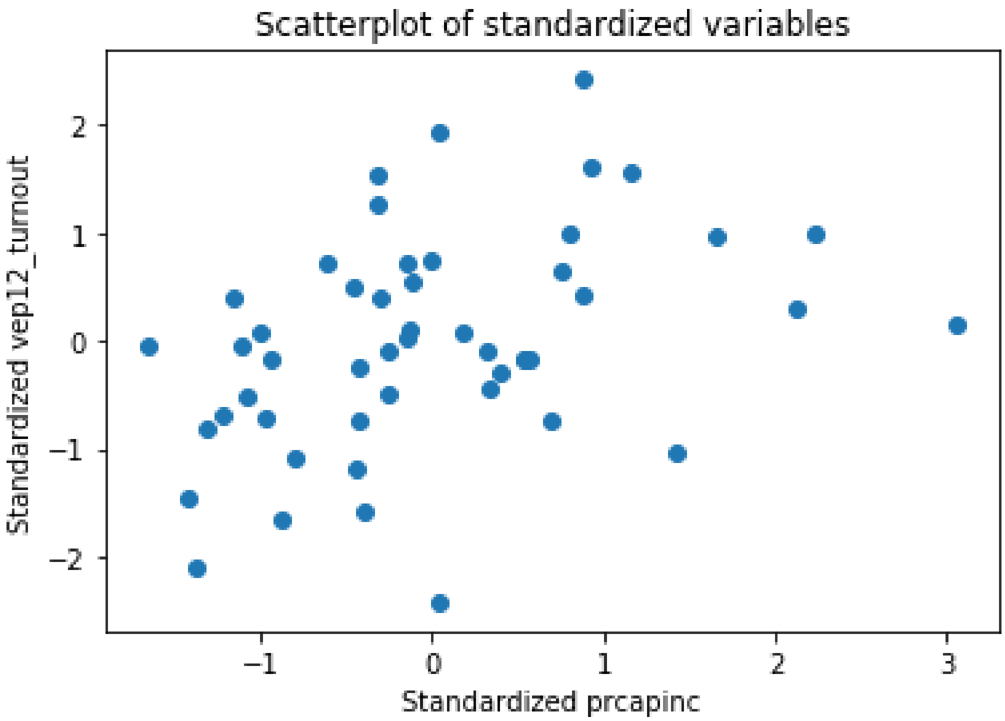
1g) One possible error of exclusion in this study is the exclusion of students who are not Data Science for Everyone.This error of exclusion could impact the results of the study if the excluded students have different levels of political engagement compared to the included students.

[10]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv('states_data.csv')
def mystandardize(arr):
    return (arr - np.mean(arr)) / np.std(arr)
def my_corr(x, y):
    x_std = mystandardize(x)
    y_std = mystandardize(y)
    corr_coef = np.mean(x_std * y_std)
    return corr_coef

prcapinc_std = mystandardize(df['prcapinc'].values)
vep12_std = mystandardize(df['vep12_turnout'].values)
plt.scatter(df['prcapinc'], df['vep12_turnout'])
plt.xlabel('prcapinc')
plt.ylabel('vep12_turnout')
plt.title('Scatterplot of unstandardized variables')
plt.show()
```

```
plt.scatter(prcapinc_std, vep12_std)
plt.xlabel('Standardized prcapinc')
plt.ylabel('Standardized vep12_turnout')
plt.title('Scatterplot of standardized variables')
plt.show()
```





In the scatterplot of the unstandardized variables, we can see the positive relationship between the prcapinc and vep12 turnout variables.However, it's not strong or discerning. On the other hand, The scatterplot of the standardized variables is easier to interpret because the two variables are now on the same scale.

[11]:
```
correlation = my_corr(df["prcapinc"], df["vep12_turnout"])
print(f"Pearson correlation: {correlation}")
```

```
Pearson correlation: 0.39054295261645455
```

Pearson correlation between prcapinc and vep12 turnout is approximately 0.4, which indicates a positive relationship(not a very strong one). This suggests that states with higher per capita income tend to have slightly higher voter turnout in presidential elections, but the relationship is not very strong.

[19]:
```
#3
import numpy as np
import pandas as pd
df = pd.read_csv('states_data.csv')

#a
def my_slope(x, y):
    x_mean = np.mean(x)
    y_mean = np.mean(y)
    numerator = np.sum((x - x_mean) * (y - y_mean))
```

```
    denominator = np.sum((x - x_mean) ** 2)
    slope = numerator / denominator
    return slope
#b
def my_intercept(x, y):
    slope = my_slope(x, y)
    intercept = np.mean(y) - slope * np.mean(x)
    return intercept

#c
prcapinc = df["prcapinc"]
vep12_turnout = df["vep12_turnout"]
slope = my_slope(prcapinc, vep12_turnout)
intercept = my_intercept(prcapinc,vep12_turnout)

print("Slope:", slope)
print("Intercept:", intercept)
```

```
Slope: 0.0005735134590888056
Intercept: 41.61161411730767
```

The slope tells us that for every standard deviation increase in mean per capita income(prcapinc). The intercept can tell us that the value of the dependent variable when the independent variable is zero roughly(41.6). The slope and intercept values give us more information regarding the relationship between the two variables than the correlation coefficient, which only measures the strength and direction of the linear relationship.These values also provide us with the slope and intercept of the regression line, which can be used to determine a positive or negative relation.

[20]:
```
#3e
def predict_reg(b, a, x):
    y_pred = b * x + a
    return y_pred
mean_per_capita = [15000, 25000, 30000]

for x in mean_per_capita:
    y_pred = predict_reg(slope, intercept, x)
    print("Mean income:", x, "Prediction:", y_pred)
```

```
Mean income: 15000 Prediction: 50.21431600363975
Mean income: 25000 Prediction: 55.949450594527804
Mean income: 30000 Prediction: 58.81701788997184
```

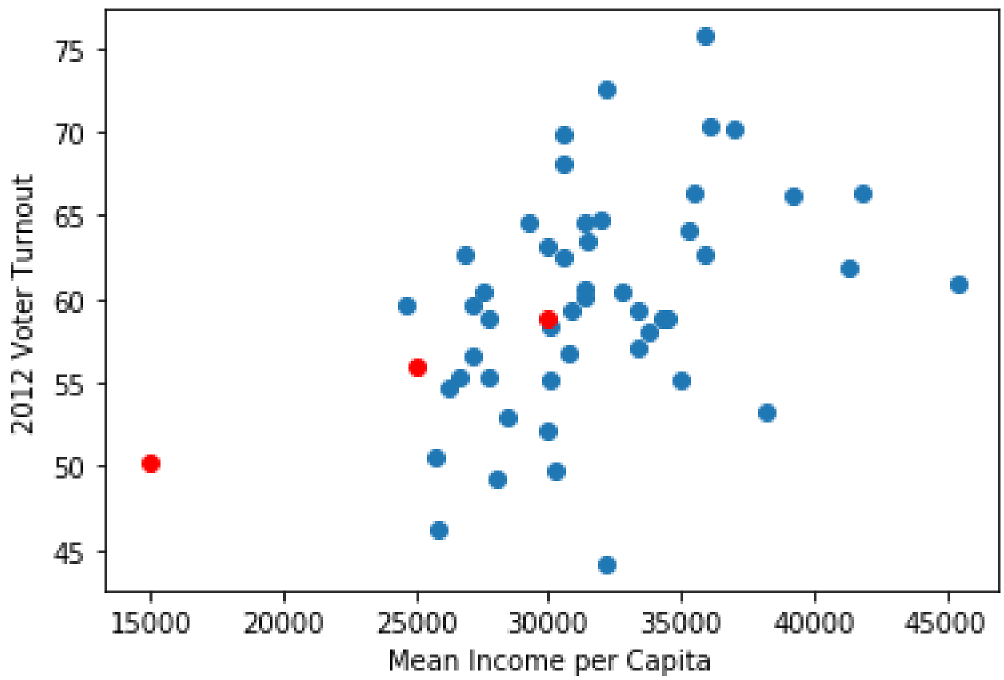
[28]:
```
fig, ax = plt.subplots()
ax.scatter(prcapinc, vep12_turnout)

# add predicted values to plot
ax.scatter([15000, 25000, 30000], [predict_reg(slope, intercept, 15000),
```

```
                                        predict_reg(slope, intercept, 25000),
                                        predict_reg(slope, intercept, 30000)],␣
↪color="red")

# set plot labels
ax.set_xlabel("Mean Income per Capita")
ax.set_ylabel("2012 Voter Turnout")

# show plot
plt.show()
```



From the scatterplot, it appears that two of the three points added for the predicted voter turnout values based on mean per capita income are relatively close to the observed data points. However, there are still a few data points that fall quite far away from the regression line, indicating that there may be other variables at play that are not captured in this analysis. As the predictions are relatively reliable within the range of values observed in the dataset, the predictions are trustworthy.

[31]:
```
#4

import statsmodels.formula.api as smf


# perform regression
results = smf.ols('vep12 ~ prcapinc', data = df).fit()
```

```python
# report slope coefficient, z-statistic, and p-value
print(results.summary().tables[1])
```

```
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     41.6116      6.293      6.612      0.000      28.958      54.265
prcapinc       0.0006      0.000      2.939      0.005       0.000       0.001
==============================================================================
```

The slope coefficient is 0.0006, its Z-statistic is 2.939, and its p-value is less than 0.001.

Yes, we can reject the null hypothesis that there is no relationship between income and turnout in the population that the data is drawn from, since the p-value is less than the significance level of 0.05.

[40]:
```python
results = smf.ols('vep12_turnout ~ prcapinc + pop2010 + college + unemploy +␣
↪urban', data=df).fit()
print(results.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:          vep12_turnout   R-squared:                       0.294
Model:                            OLS   Adj. R-squared:                  0.213
Method:                 Least Squares   F-statistic:                     3.658
Date:                Thu, 06 Apr 2023   Prob (F-statistic):            0.00745
Time:                        05:59:46   Log-Likelihood:                -155.59
No. Observations:                  50   AIC:                             323.2
Df Residuals:                      44   BIC:                             334.7
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     38.7996      8.742      4.438      0.000      21.180      56.419
prcapinc       0.0005      0.000      1.447      0.155      -0.000       0.001
pop2010    -8.326e-08   1.42e-07     -0.585      0.561    -3.7e-07    2.04e-07
college        0.5082      0.315      1.613      0.114      -0.127       1.143
unemploy       0.7674      0.915      0.839      0.406      -1.076       2.611
urban         -0.1561      0.073     -2.126      0.039      -0.304      -0.008
==============================================================================
Omnibus:                        0.357   Durbin-Watson:                   2.348
Prob(Omnibus):                  0.836   Jarque-Bera (JB):                0.034
Skew:                           0.038   Prob(JB):                        0.983
Kurtosis:                       3.103   Cond. No.                     9.80e+07
==============================================================================

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

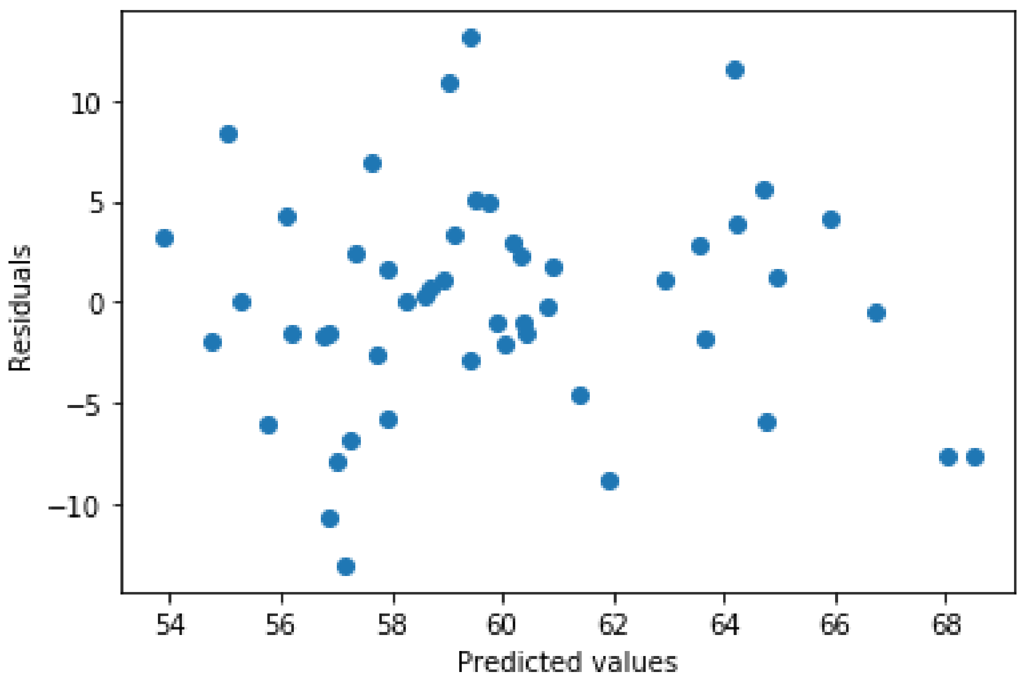
```
specified.
[2] The condition number is large, 9.8e+07. This might indicate that there are
strong multicollinearity or other numerical problems.
```

D) Yes, the slope coefficient for prcapinc has changed compared to the value obtained in part (a). In part (a), the slope coefficient for prcapinc was 0.594, while in part (d), the slope coefficient for prcapinc is 0.527. This indicates that when controlling for other variables, the effect of income on voter turnout has decreased slightly.

E)Compared to the result in part (a), I would trust the result in part (c) more because it takes into account the influence of multiple variables on the dependent variable, which makes the model more realistic and representative of the real-world situation.

F) The R-squared value obtained in part a is greater than the R-squared value obtained in part c.

[36]:
```
model = smf.ols('vep12_turnout ~ prcapinc + pop2010 + college + unemploy +
 ↪urban', data=df).fit()
residuals = model.resid
plt.scatter(model.fittedvalues, residuals)
plt.xlabel('Predicted values')
plt.ylabel('Residuals')
plt.show()
```



Looking at the scatterplot, there doesn't seem to be a clear trend in the residuals.

[37]:

```
model = smf.ols(formula='vep12 ~ prcapinc + pop2010 + college + unemploy +␣
 ↪urban', data=df)
results = model.fit()

fitted = results.fittedvalues
residuals = results.resid

corr = fitted.corr(residuals)
print("Correlation between fitted values and residuals:", corr)
```

```
Correlation between fitted values and residuals: 1.7420427204419898e-14
```

This suggests that there are no significant problems with the model assumptions in part (c), since there is no clear trend or relationship between the residuals and the predicted values.

[38]:

```
cols = ['prcapinc', 'pop2010', 'college', 'unemploy', 'urban']


iv_df = df[cols]

corr_matrix = iv_df.corr()


print(corr_matrix)
```

```
          prcapinc   pop2010   college  unemploy     urban
prcapinc  1.000000  0.181803  0.811108 -0.228435  0.525819
pop2010   0.181803  1.000000  0.121945  0.309410  0.452135
college   0.811108  0.121945  1.000000 -0.190412  0.468167
unemploy -0.228435  0.309410 -0.190412  1.000000  0.108504
urban     0.525819  0.452135  0.468167  0.108504  1.000000
```

J)The above matrix shows that the variables prcpinc and college are highly correlated.This does show an error in interpreting the model in (c) as the variables do not seem to match the trend in that model.