A2_DiTong

October 16, 2018

1 Assignment 2

MACS 30000, Dr. Evans
Di Tong

1. Imputing age and gender

(a) I will first use the SurveyIncome data to fit a simple linear regression model and a logistic regression model listed below:

$age_i = \beta_0 + \beta_1 * totinc_i + \beta_2 * wgt_i$
$log(P_f/P_m) = \beta_0 + \beta_1 * totinc_i + \beta_2 * wgt$
Then I will create a total income variable for the BestIncome data by adding up labor income and capital income:
best_tot_inc = lab_inc + cap_inc
Then, I will plug in the total income variable and the weight variable of the BestIncome dataset to the two regression models above in order to predict the values of age and gender. Finally, I will impute the predicted values of age and gender into the BestIncome dataset.

```
In [17]: # Import packages
         import numpy as np
         from sklearn import linear_model
         import pandas as pd
         import matplotlib.pyplot as plt
         import statsmodels.api as sm
```

```
In [29]: # Read in the data, name the variables
         BestInc = pd.read_csv('C:/Users/700S/persp-analysis_A18/Assignments/A2/BestIncome.txt
                               , names=['lab_inc', 'cap_inc', 'hgt', 'wgt'])
         SurvInc = pd.read_csv('C:/Users/700S/persp-analysis_A18/Assignments/A2/SurvIncome.txt
                               , names=['tot_inc', 'wgt', 'age', 'gender'])
```

(b) Here is where I'll use my proposed method from part (a) to impute variables.

```
In [19]: # Create new datasets and variables needed for fitting regression model
         survey_x = np.column_stack((SurvInc.tot_inc, SurvInc.wgt))
         age = SurvInc.age
         gender = SurvInc.gender
```

```python
# Simple linear regression of age on tot_inc and wgt
reg_age = linear_model.LinearRegression()
reg_age.fit(survey_x, age)

# Logistic regression of gender on tot_inc and wgt
reg_gender = linear_model.LogisticRegression()
reg_gender.fit(survey_x, gender)

# Create new variables and datasets needed for prediction and imputation
best_tot_inc = BestInc.lab_inc + BestInc.cap_inc
best_x = np.column_stack((best_tot_inc, BestInc.wgt))

#Impute age and gender using information from the SurvInc
imp_age = reg_age.predict(best_x)
imp_gender = reg_gender.predict(best_x)
imp_BestInc = np.column_stack((BestInc, imp_age, imp_gender))

# Convert numpy array into panda dataframe
new_BestInc = pd.DataFrame(imp_BestInc)
new_BestInc.columns = ['lab_inc', 'cap_inc', 'hgt', 'wgt', 'age', 'gender']
```

c) Here is where I'll report the descriptive statistics for my new imputed variables.

```python
In [20]: # Get descriptive statistics of age
         new_BestInc.age.describe()
```

```
Out[20]: count    10000.000000
         mean        44.890828
         std          0.219150
         min         43.976495
         25%         44.743776
         50%         44.886944
         75%         45.038991
         max         45.703819
         Name: age, dtype: float64
```

```python
In [21]: # Get descriptive statistics of gender
         new_BestInc.gender.describe()
```

```
Out[21]: count    10000.000000
         mean         0.471700
         std          0.499223
         min          0.000000
         25%          0.000000
         50%          0.000000
         75%          1.000000
         max          1.000000
         Name: gender, dtype: float64
```

(d) Correlation matrix for the now six variables

```
In [22]: corr = new_BestInc.corr()
         corr.style.background_gradient()
```

```
Out[22]: <pandas.io.formats.style.Styler at 0x1e72e98d128>
```

2. Stationarity and data drift

(a) Estimate by OLS and report coefficients and standard errors on those coefficients.

```
In [23]: # Read in the data, name the variables
         Inc_Int = pd.read_csv(
             'C:/Users/700S/persp-analysis_A18/Assignments/A2/IncomeIntel.txt', \
             names=['grad_year', 'gre_qnt', 'salary_p4'])

         #Define Outcome and Independent Variables
         outcome  = 'salary_p4'
         features = ['gre_qnt']

         x, y = Inc_Int[features], Inc_Int[outcome]

         # Simple linear regression of salary_p4 on gre_qnt
         x = sm.add_constant(x, prepend=False)
         x.head()

         m = sm.OLS(y, x)

         res = m.fit()

         # Report coefficients and SE's
         print(res.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              salary_p4   R-squared:                       0.263
Model:                            OLS   Adj. R-squared:                  0.262
Method:                 Least Squares   F-statistic:                     356.3
Date:                Tue, 16 Oct 2018   Prob (F-statistic):           3.43e-68
Time:                        23:31:00   Log-Likelihood:                -10673.
No. Observations:                1000   AIC:                         2.135e+04
Df Residuals:                     998   BIC:                         2.136e+04
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
gre_qnt      -25.7632      1.365    -18.875      0.000     -28.442     -23.085
const       8.954e+04    878.764    101.895      0.000    8.78e+04    9.13e+04
```

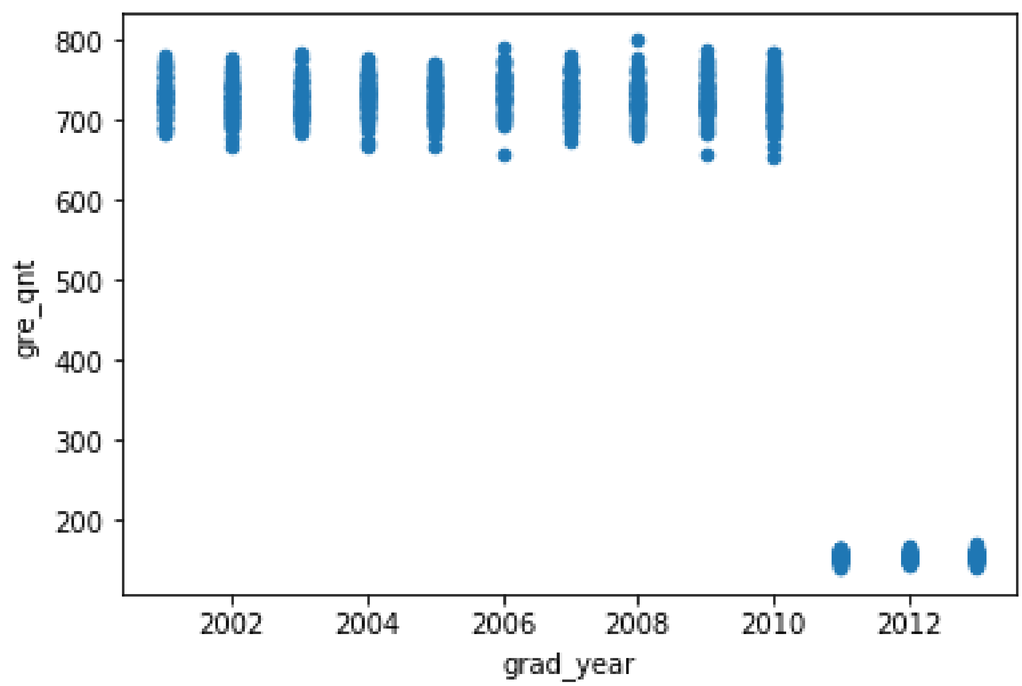
```
==============================================================================
Omnibus:                        9.118   Durbin-Watson:                   1.424
Prob(Omnibus):                  0.010   Jarque-Bera (JB):                9.100
Skew:                           0.230   Prob(JB):                       0.0106
Kurtosis:                       3.077   Cond. No.                     1.71e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.71e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

(b) Create a scatterplot of GRE score and graduation year.

```
In [24]: # Make scatterplot of GRE quantitative score on the y-axis
         # and graduation year on the x-axis
         # Simple scatterplot using matplotlib

         grad_year = Inc_Int['grad_year']
         gre_qnt = Inc_Int['gre_qnt']
         Inc_Int.plot(x='grad_year', y='gre_qnt', kind='scatter')
         plt.show()
```



From the scatterplot above, we can see that there is a huge gap between the pre-2011 data and the post-2011 data. It is obvious that the the change in the GRE score scale rather than a real drastic "intellengence drop" of GRE takers accounts for this gap. Hence, we will get biased coefficients using this problematic data in the regression that tests our hyphothesis. My solution to this problem is to convert the pre-2011 GRE scores according to the post-2011 scale using this equation: g_qnt_converted = 130.0 + (g_qnt - 200) * (170 - 130) / (800-200)

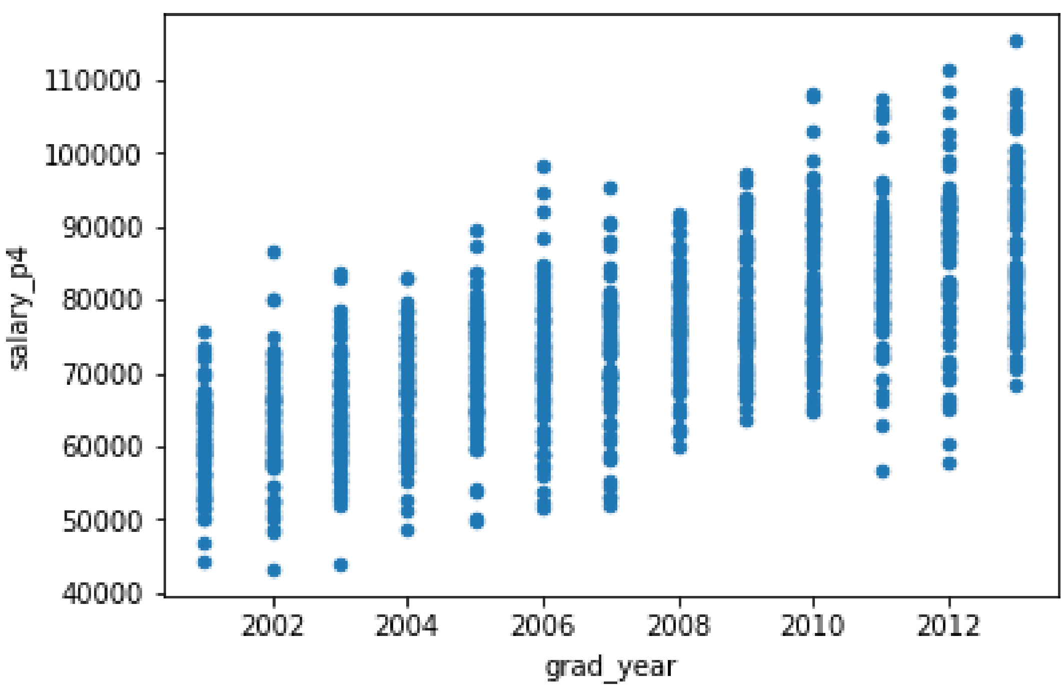
In [25]:
```
# Convert GRE quantitative score before 2011
# according to the scale adopted after 2011

for i in range(len(grad_year)):
    if grad_year[i] < 2011.0:
        gre_qnt[i] = 130.0 + (gre_qnt[i] - 200.0) * (170.0 - 130.0) / (800.0 - 200.0)
```

(c) Create a scatterplot of income and graduation year

In [26]:
```
# Make scatterplot of income 4 years after graduation on the y-axis
# and graduation year on the x-axis
# Simple scatterplot using matplotlib

grad_year = Inc_Int['grad_year']
salary_p4 = Inc_Int['salary_p4']
Inc_Int.plot(x='grad_year', y='salary_p4', kind='scatter')
plt.show()
```



From the scatterplot above, we can see that the salary data is non-stationary, namely, there is a time trend in salary. Hence, we will get biased coefficients using this problematic data in the regression that tests our hyphothesis. My proposed solution is to detrend the salary data by first calculate the average growth rate in salaries across all 13 years using the mean salary of each year and then divide each salary by (1 + avg_growth_rate) ** (grad_year - 2001). Here is the equation for calculating the growth rate for each year: growth rate = (mean salary - mean salary of the previous year) / mean salary of the previous year

In [27]:
```
# Calculate the mean salary each year
avg_inc_by_year = Inc_Int['salary_p4'].groupby(Inc_Int['grad_year']).mean().values

# Calculate the average growth rate in salaries across all 13 years
avg_growth_rate = ((avg_inc_by_year[1:] - \
                    avg_inc_by_year[:-1]) / avg_inc_by_year[:-1]).mean()
avg_growth_rate

# Detrend salary
for i in range(len(salary_p4)):
    salary_p4[i] = salary_p4[i] / ((1 + avg_growth_rate) ** (grad_year[i] - 2001.0))
```

(d) Re-estimate by OLS with updated variables and report coefficients and standard errors on those coefficients.

In [28]:
```
#Define Outcome and Independent Variables
outcome  = 'salary_p4'
features = ['gre_qnt']

x, y = Inc_Int[features], Inc_Int[outcome]

# Simple linear regression of salary_p4 on gre_qnt
x = sm.add_constant(x, prepend=False)
x.head()

m = sm.OLS(y, x)

res = m.fit()

# Report coefficients and SE's
print(res.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:              salary_p4   R-squared:                       0.000
Model:                            OLS   Adj. R-squared:                 -0.001
Method:                 Least Squares   F-statistic:                   0.05257
Date:                Tue, 16 Oct 2018   Prob (F-statistic):              0.819
Time:                        23:31:04   Log-Likelihood:                -10291.
No. Observations:                1000   AIC:                         2.059e+04
Df Residuals:                     998   BIC:                         2.060e+04
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
```

	coef	std err	t	P>|t|	[0.025	0.975]
gre_qnt	-9.9681	43.474	-0.229	0.819	-95.279	75.343
const	6.304e+04	7083.395	8.900	0.000	4.91e+04	7.69e+04

Omnibus:	0.757	Durbin-Watson:	2.026
Prob(Omnibus):	0.685	Jarque-Bera (JB):	0.668
Skew:	0.059	Prob(JB):	0.716
Kurtosis:	3.048	Cond. No.	5.11e+03

```
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 5.11e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

The coefficient of gre_qnt changes from -25.7632 to -9.9681, and the coefficient for the constant changes from 8.954e+04 to 6.304e+0.

In the regression in (a), GRE quantitative scores before 2011 are much higher than that after 2011, while the salary presents an increasing trend over the years. Hence, the problematic decreasing pattern of the GRE scores and increasing pattern of the salary might have magnified the negative association between them. After we adjust all GRE scores into the same range and detrend the salary data, the variance of both variables get smaller, and therefore, the negative association between them is lessened.

Since the P value for the coefficient of gre_qnt is very large, we do not have sufficient evidence to reject the null hypothesis that the GRE quantitative score has no effect on salary. Therefore, the results provide evidence against our hypothesis that higher intelligence is associated with higher income. But this regression model itself is problematic, as it does not control for a range of other variables that could affect salary. Hence, the results might not reflect the real situation in the first place.

3. Assessment of Kossinets and Watts.

See next page.

The question of Kossinets and Watts (2009)'s research is: what is the origin of homophily? The more precise version in their own words is that, "on what grounds do individuals selectively make or break some ties over others, and how do these choices shed light on the observation that similar people are more likely to become acquainted than dissimilar people? " (p. 406)

The authors use a network data set comprising interaction, affiliation, and attribute-type longitudinal data of 30,396 students, faculty, and staff in a university community during one academic year.

The data is constructed on the basis of three data sources: "(1) the logs of e-mail interactions within the university over one academic year, (2) a database of individual attributes (status, gender, age, department, number of years in the community, etc.), and (3) records of course registration, in which courses were recorded separately for each semester. " (Kossinets and Watts 2009, p. 410)

For the final data set used for analysis, there are 30,396 observations who are selected as active e-mail users from the original sample of 43,553 individuals who used university e-mail to both send and receive messages during the academic year.

While the article reports analysis of only one academic year's worth of data, the full data set spans two calendar years.

A precise descriptions and definitions of all variables can be found in Appendix A.

According to the note on data cleansing and missing values in Appendix B, the error-correction strategies for errors and missing values only yield marginal improvement for the variable "field". Hence, the problem of errors and missing values in the variable field is largely unsolved. Since field is one of the measurements of similarity, this problem might diminish the authors' ability to effectively and accurately test the effect of similarity on homophily.

Due to privacy considerations, the email message content is unavailable to the researchers. Yet, email contact and its frequency per se do not necessarily represent the existence and intensity of social ties and can not fully measure all dimensions of the concept social tie. Without message content, it is extremely difficult to interpret the meaning of any given pattern of relations.

The authors address this weakness by proposing possible supplementary approaches to construct validity. The first approach is text analysis and validation of inferred network ties through selective surveying of e-mail users. Alternatively, they can also obtain message content data for analysis through informed consent procedures under which users would be willing to provide content in exchange for benefits as well as assurances on the use of the content.