AB Testing

February 4, 2020

[4]:
```
import pandas as pd
import numpy as np
import statsmodels.api as sm
import statsmodels.formula.api as smf
from statsmodels.formula.api import ols
import patsy
from plotnine import *
from scipy.stats import ttest_ind
```

0.0.1 Exercise One: Importing Data

[7]:
```
# Control and experiment dataframes
control_df = pd.read_csv('control_data.csv')
experiment_df = pd.read_csv('experiment_data.csv')
```

0.0.2 Exercise Two: Exploring Data

[37]:
```
# Looking at the variables between the two data sets
control_df.head(1)
```

[37]:
```
          Date  Pageviews  Clicks  Enrollments  Payments  experimental
0  Sat, Oct 11       7723     687        134.0      70.0             0
```

[9]:
```
experiment_df.head(1)
```

[9]:
```
          Date  Pageviews  Clicks  Enrollments  Payments
0  Sat, Oct 11       7716     686        105.0      34.0
```

Payments is the response variable we are interested in observing.

0.0.3 Exercise Three: Stacking the dataframes

[10]:
```
# Adding column for which group they are in
experiment_df['experimental'] = 1
control_df['experimental'] = 0
```

[13]:
```
# Combining dataframes
appended_df =control_df.append(experiment_df, ignore_index = True)
appended_df.sample(10) # Looks like it stacked correctly
# Notcied we had NaNs here
```

[13]:
```
           Date  Pageviews  Clicks  Enrollments  Payments  experimental
35  Sat, Nov 15       8630     743          NaN       NaN             0
48  Wed, Oct 22       9737     801        128.0      70.0             1
68  Tue, Nov 11       9931     831          NaN       NaN             1
52  Sun, Oct 26       8881     693        153.0     101.0             1
39  Mon, Oct 13      10480     884        145.0      79.0             1
31  Tue, Nov 11       9880     830          NaN       NaN             0
73  Sun, Nov 16       8988     710          NaN       NaN             1
20  Fri, Oct 31       8890     706        174.0     101.0             0
56  Thu, Oct 30       9308     728        207.0      67.0             1
64   Fri, Nov 7       9272     767          NaN       NaN             1
```

0.0.4 Exercise Four: What are the Y Variables

We want to measure payments and enrollments. It might be best to use payments per click and enrollments per click in order to account for clicks.

0.0.5 Excercise Five: Validating the Data

[14]:
```
appended_df = appended_df.dropna()
```

[39]:
```
appended_df.shape
```

[39]:
```
(46, 8)
```

[20]:
```
# Seeing if means are similar
Mean_con_pageviews = appended_df[appended_df['experimental'] == 0].Pageviews.
 ↪mean()
Mean_exp_pageviews = appended_df[appended_df['experimental'] == 1].Pageviews.
 ↪mean()

print(f'Mean page views for the control group is {Mean_con_pageviews:.2f} and␣
 ↪mean page views for the treated group is {Mean_exp_pageviews:.2f}.')
```

```
Mean page views for the control group is 9224.48 and mean page views for the
treated group is 9189.65.
```

0.0.6 Excercise Six: T-test

[22]:
```
# Prep for t-test
Control_pageviews = appended_df[appended_df['experimental'] == 0].Pageviews.
 ↪values
```

```
Experimental_pageviews = appended_df[appended_df['experimental'] == 1].
 ↪Pageviews.values
```

```
[24]: # No significant difference
      ttest_ind(Control_pageviews, Experimental_pageviews)
```

```
[24]: Ttest_indResult(statistic=0.14289030617246118, pvalue=0.8870291739409888)
```

There is no significant difference between page views for the treated and untreated group.

0.0.7 Excercise Seven/Eight

```
[27]: # Checking if pre-treatment variable clicks is also balanced
      Clicks_con = appended_df[appended_df['experimental'] == 0].Clicks.mean()
      Clicks_exp = appended_df[appended_df['experimental'] == 1].Clicks.mean()
      print(f'Mean clicks for the control group is {Clicks_con:.2f} and mean clicks␣
       ↪for the treated group is {Clicks_exp:.2f}.')
```

```
Mean clicks for the control group is 751.87 and mean clicks for the treated
group is 750.43.
```

```
[28]: # Prep for t-test
      Control_clicks = appended_df[appended_df['experimental'] == 0].Clicks.values
      Experimental_clicks = appended_df[appended_df['experimental'] == 1].Clicks.
       ↪values
```

```
[29]: ttest_ind(Control_clicks, Experimental_clicks)
```

```
[29]: Ttest_indResult(statistic=0.06598153974790888, pvalue=0.9476914165204888)
```

There is no significant difference between the treated group and the non-treated groups in amount of clicks per day

0.0.8 Exercise Nine: Average Treatment Effect

```
[45]: # Two metrics will be Payments per click and entrollments per click
      appended_df['PPC'] = appended_df['Payments']/appended_df['Clicks']
      appended_df['EPC'] = appended_df['Enrollments']/appended_df['Clicks']
      appended_df.tail(2)
```

```
[45]:          Date  Pageviews  Clicks  Enrollments  Payments  experimental  \
      58  Sat, Nov 1       8448     695        142.0     100.0             1
      59  Sun, Nov 2       8836     724        182.0     103.0             1

               PPC       EPC
      58  0.143885  0.204317
      59  0.142265  0.251381
```

[34]:
```
# PPC not significant: p-value 0.59
ttest_ind(appended_df[appended_df['experimental'] == 0].PPC.values,␣
 ↪appended_df[appended_df['experimental'] == 1].PPC.values)
```

[34]:
```
Ttest_indResult(statistic=0.5387777625331603, pvalue=0.5927558614268024)
```

[36]:
```
# EPC not significant: p-value 0.13
ttest_ind(appended_df[appended_df['experimental'] == 0].EPC.values,␣
 ↪appended_df[appended_df['experimental'] == 1].EPC.values)
```

[36]:
```
Ttest_indResult(statistic=1.5396752696188791, pvalue=0.13080105104323278)
```

[58]:
```
# Looking at PPC to see difference in the mean and sd
print(appended_df[appended_df['experimental'] == 0].PPC.mean(),␣
 ↪appended_df[appended_df['experimental'] == 1].PPC.mean())
print(appended_df[appended_df['experimental'] == 0].PPC.std(),␣
 ↪appended_df[appended_df['experimental'] == 1].PPC.std())
```

```
0.11826943663824471 0.11337257964843535
0.02940513659511976 0.0321759709762389
```

[59]:
```
# Looking at EPC to see difference in the mean and sd
print(appended_df[appended_df['experimental'] == 0].EPC.mean(),␣
 ↪appended_df[appended_df['experimental'] == 1].EPC.mean())
print(appended_df[appended_df['experimental'] == 0].EPC.std(),␣
 ↪appended_df[appended_df['experimental'] == 1].EPC.std())
```

```
0.2203509696998493 0.19956638767058335
0.044042538470278544 0.04745089805032958
```

[61]:
```
# Checking how much of each group we still have after dropna
print(appended_df[appended_df['experimental'] == 1].shape[0],␣
 ↪appended_df[appended_df['experimental'] == 0].shape[0])
```

```
23 23
```

Statistically, Udacity did not achieve their goal. However this might be due to a small sample size. The largest difference mean difference we saw was in enrollments per click, but this was not found to be significant. We only had 23 samples from each treated and untreated groups. Had Udacity had a larger sample size they may have been able to see something of statistical significance. However, it is worth nothing that EPC and PPC are lower for the experimental groups on average.

0.0.9 Exercise Ten: Regression Confirmation

[65]:
```
smf.ols('PPC ~ experimental', data = appended_df).fit().summary()
```

[65]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    PPC   R-squared:                       0.007
Model:                            OLS   Adj. R-squared:                 -0.016
Method:                 Least Squares   F-statistic:                    0.2903
Date:                Mon, 03 Feb 2020   Prob (F-statistic):              0.593
Time:                        21:15:39   Log-Likelihood:                 95.810
No. Observations:                  46   AIC:                            -187.6
Df Residuals:                      44   BIC:                            -184.0
Df Model:                           1
Covariance Type:            nonrobust
================================================================================
                   coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
Intercept        0.1183      0.006     18.403      0.000       0.105       0.131
experimental    -0.0049      0.009     -0.539      0.593      -0.023       0.013
==============================================================================
Omnibus:                        0.968   Durbin-Watson:                   1.092
Prob(Omnibus):                  0.616   Jarque-Bera (JB):                0.985
Skew:                           0.316   Prob(JB):                        0.611
Kurtosis:                       2.662   Cond. No.                         2.62
==============================================================================

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

[64]:

```python
smf.ols('EPC ~ experimental', data = appended_df).fit().summary()
```

[64]:

```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:                    EPC   R-squared:                       0.051
Model:                            OLS   Adj. R-squared:                  0.030
Method:                 Least Squares   F-statistic:                     2.371
Date:                Mon, 03 Feb 2020   Prob (F-statistic):              0.131
Time:                        21:15:34   Log-Likelihood:                 77.613
No. Observations:                  46   AIC:                            -151.2
Df Residuals:                      44   BIC:                            -147.6
Df Model:                           1
Covariance Type:            nonrobust
================================================================================
                   coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
Intercept        0.2204      0.010     23.084      0.000       0.201       0.240
```

```
experimental    -0.0208      0.013     -1.540      0.131      -0.048       0.006
==============================================================================
Omnibus:                        6.181   Durbin-Watson:                   1.130
Prob(Omnibus):                  0.045   Jarque-Bera (JB):                6.094
Skew:                           0.850   Prob(JB):                       0.0475
Kurtosis:                       2.460   Cond. No.                         2.62
==============================================================================

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

We see that PPC and EPC p-values and f-statistics are the same as seen in the t-test.

0.0.10 Exercise Eleven: Adding Date

[62]:
```
smf.ols('PPC ~ experimental + C(Date)', data = appended_df).fit().summary()
```

[62]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:                    PPC   R-squared:                       0.743
Model:                            OLS   Adj. R-squared:                  0.475
Method:                 Least Squares   F-statistic:                     2.770
Date:                Mon, 03 Feb 2020   Prob (F-statistic):            0.00991
Time:                        21:01:24   Log-Likelihood:                 126.94
No. Observations:                  46   AIC:                            -205.9
Df Residuals:                      22   BIC:                            -162.0
Df Model:                          23
Covariance Type:            nonrobust
================================================================================
==========
                            coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
----------
Intercept                 0.0815      0.016      5.090      0.000       0.048
0.115
C(Date)[T.Fri, Oct 24]    0.0791      0.022      3.569      0.002       0.033
0.125
C(Date)[T.Fri, Oct 31]    0.0777      0.022      3.507      0.002       0.032
0.124
C(Date)[T.Mon, Oct 13]    0.0179      0.022      0.809      0.427      -0.028
0.064
C(Date)[T.Mon, Oct 20]    0.0343      0.022      1.548      0.136      -0.012
```

```
0.080
C(Date)[T.Mon, Oct 27]      0.0799      0.022      3.604      0.002       0.034
0.126
C(Date)[T.Sat, Nov 1]       0.0612      0.022      2.763      0.011       0.015
0.107
C(Date)[T.Sat, Oct 11]     -0.0033      0.022     -0.148      0.884      -0.049
0.043
C(Date)[T.Sat, Oct 18]      0.0239      0.022      1.080      0.292      -0.022
0.070
C(Date)[T.Sat, Oct 25]      0.0742      0.022      3.347      0.003       0.028
0.120
C(Date)[T.Sun, Nov 2]       0.0405      0.022      1.826      0.081      -0.005
0.086
C(Date)[T.Sun, Oct 12]      0.0239      0.022      1.078      0.293      -0.022
0.070
C(Date)[T.Sun, Oct 19]      0.0196      0.022      0.887      0.385      -0.026
0.066
C(Date)[T.Sun, Oct 26]      0.0673      0.022      3.038      0.006       0.021
0.113
C(Date)[T.Thu, Oct 16]      0.0095      0.022      0.430      0.672      -0.036
0.055
C(Date)[T.Thu, Oct 23]      0.0161      0.022      0.725      0.476      -0.030
0.062
C(Date)[T.Thu, Oct 30]      0.0181      0.022      0.817      0.423      -0.028
0.064
C(Date)[T.Tue, Oct 14]      0.0394      0.022      1.779      0.089      -0.007
0.085
C(Date)[T.Tue, Oct 21]      0.0226      0.022      1.022      0.318      -0.023
0.069
C(Date)[T.Tue, Oct 28]      0.0643      0.022      2.904      0.008       0.018
0.110
C(Date)[T.Wed, Oct 15]      0.0157      0.022      0.709      0.486      -0.030
0.062
C(Date)[T.Wed, Oct 22]      0.0196      0.022      0.884      0.386      -0.026
0.066
C(Date)[T.Wed, Oct 29]      0.0452      0.022      2.040      0.054      -0.001
0.091
experimental               -0.0049      0.007     -0.750      0.461      -0.018
0.009
==============================================================================
Omnibus:                        4.114   Durbin-Watson:                   1.713
Prob(Omnibus):                  0.128   Jarque-Bera (JB):                1.796
Skew:                           0.000   Prob(JB):                        0.407
Kurtosis:                       2.032   Cond. No.                         27.3
==============================================================================

Warnings:
```

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

```
[67]: smf.ols('EPC ~ experimental + C(Date)', data = appended_df).fit().summary()
```

```
[67]: <class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:                    EPC   R-squared:                       0.918
Model:                            OLS   Adj. R-squared:                  0.832
Method:                 Least Squares   F-statistic:                     10.67
Date:                Mon, 03 Feb 2020   Prob (F-statistic):           2.59e-07
Time:                        21:19:46   Log-Likelihood:                 133.84
No. Observations:                  46   AIC:                            -219.7
Df Residuals:                      22   BIC:                            -175.8
Df Model:                          23
Covariance Type:            nonrobust
================================================================================
==========
                             coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
----------
Intercept                  0.1894      0.014     13.752      0.000       0.161
0.218
C(Date)[T.Fri, Oct 24]     0.1236      0.019      6.483      0.000       0.084
0.163
C(Date)[T.Fri, Oct 31]     0.0703      0.019      3.685      0.001       0.031
0.110
C(Date)[T.Mon, Oct 13]    -0.0051      0.019     -0.269      0.790      -0.045
0.034
C(Date)[T.Mon, Oct 20]     0.0058      0.019      0.303      0.765      -0.034
0.045
C(Date)[T.Mon, Oct 27]     0.1126      0.019      5.907      0.000       0.073
0.152
C(Date)[T.Sat, Nov 1]      0.0377      0.019      1.977      0.061      -0.002
0.077
C(Date)[T.Sat, Oct 11]    -0.0049      0.019     -0.259      0.798      -0.044
0.035
C(Date)[T.Sat, Oct 18]    -0.0199      0.019     -1.043      0.308      -0.059
0.020
C(Date)[T.Sat, Oct 25]     0.0433      0.019      2.269      0.033       0.004
0.083
C(Date)[T.Sun, Nov 2]      0.0953      0.019      4.999      0.000       0.056
0.135
```

```
C(Date)[T.Sun, Oct 12]    -0.0108      0.019     -0.565      0.578      -0.050
0.029
C(Date)[T.Sun, Oct 19]     0.0019      0.019      0.098      0.923      -0.038
0.041
C(Date)[T.Sun, Oct 26]     0.0451      0.019      2.365      0.027       0.006
0.085
C(Date)[T.Thu, Oct 16]    -0.0133      0.019     -0.698      0.492      -0.053
0.026
C(Date)[T.Thu, Oct 23]     0.0115      0.019      0.603      0.553      -0.028
0.051
C(Date)[T.Thu, Oct 30]     0.0769      0.019      4.035      0.001       0.037
0.116
C(Date)[T.Tue, Oct 14]    -0.0023      0.019     -0.119      0.906      -0.042
0.037
C(Date)[T.Tue, Oct 21]     0.0168      0.019      0.880      0.388      -0.023
0.056
C(Date)[T.Tue, Oct 28]     0.0357      0.019      1.871      0.075      -0.004
0.075
C(Date)[T.Wed, Oct 15]     0.0025      0.019      0.131      0.897      -0.037
0.042
C(Date)[T.Wed, Oct 22]    -0.0024      0.019     -0.128      0.899      -0.042
0.037
C(Date)[T.Wed, Oct 29]     0.0918      0.019      4.817      0.000       0.052
0.131
experimental              -0.0208      0.006     -3.697      0.001      -0.032
-0.009
==============================================================================
Omnibus:                        3.257   Durbin-Watson:                   1.334
Prob(Omnibus):                  0.196   Jarque-Bera (JB):                2.788
Skew:                           0.000   Prob(JB):                        0.248
Kurtosis:                       4.206   Cond. No.                         27.3
==============================================================================

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

The standard error for treatment is lower at -0.0049 for PPC and 0.006 for EPC. This is because of matching between the two groups.

0.0.11 Excercise Twelve

We would tell them that the model we looking at the two treatment groups as a whole we do not see any statistical significance between payments per click (PPC) and payments per enrollments (EPC). Their sample size is too small and a larger one is required to determine if there is an effect. However, the treated group did have a lower mean for both PPC and EPC. When looking at treatment and date compared to PPC and EPC (paired groups of treated and not treated), we did find significant evidence that EPC and PPC were lower for the treated group than the untreated group.

0.0.12 Excercise Thirteen

[77]:
```
# For every day of the week
appended_df['Day']  = appended_df['Date'].str[:3]
appended_df.head(3)
```

[77]:
```
          Date  Pageviews  Clicks  Enrollments  Payments  experimental  \
0  Sat, Oct 11       7723     687        134.0      70.0             0
1  Sun, Oct 12       9102     779        147.0      70.0             0
2  Mon, Oct 13      10511     909        167.0      95.0             0

        PPC       EPC  Day
0  0.101892  0.195051  Sat
1  0.089859  0.188703  Sun
2  0.104510  0.183718  Mon
```

[80]:
```
# EPC and PPC are not significant
smf.ols('EPC ~ experimental + C(Day)', data = appended_df).fit().summary()
```

[80]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:                    EPC   R-squared:                       0.163
Model:                            OLS   Adj. R-squared:                  0.008
Method:                 Least Squares   F-statistic:                     1.053
Date:                Mon, 03 Feb 2020   Prob (F-statistic):              0.412
Time:                        21:29:08   Log-Likelihood:                 80.485
No. Observations:                  46   AIC:                            -145.0
Df Residuals:                      38   BIC:                            -130.3
Df Model:                           7
Covariance Type:            nonrobust
================================================================================
=
                    coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
-
Intercept         0.2540      0.020     12.646      0.000       0.213
0.295
C(Day)[T.Mon]    -0.0269      0.027     -1.006      0.321      -0.081
0.027
C(Day)[T.Sat]    -0.0506      0.025     -2.024      0.050      -0.101
-5.8e-08
```

```
C(Day)[T.Sun]    -0.0317      0.025     -1.270      0.212      -0.082
0.019
C(Day)[T.Thu]    -0.0396      0.027     -1.482      0.147      -0.094
0.015
C(Day)[T.Tue]    -0.0479      0.027     -1.793      0.081      -0.102
0.006
C(Day)[T.Wed]    -0.0340      0.027     -1.272      0.211      -0.088
0.020
experimental     -0.0208      0.014     -1.523      0.136      -0.048
0.007
==============================================================================
Omnibus:                        4.493   Durbin-Watson:                   0.869
Prob(Omnibus):                  0.106   Jarque-Bera (JB):                3.644
Skew:                           0.576   Prob(JB):                        0.162
Kurtosis:                       2.242   Cond. No.                         9.27
==============================================================================

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

[79]:
```
smf.ols('PPC ~ experimental + C(Day)', data = appended_df).fit().summary()
```

[79]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:                    PPC   R-squared:                       0.138
Model:                            OLS   Adj. R-squared:                 -0.021
Method:                 Least Squares   F-statistic:                    0.8687
Date:                Mon, 03 Feb 2020   Prob (F-statistic):              0.540
Time:                        21:28:01   Log-Likelihood:                 99.073
No. Observations:                  46   AIC:                            -182.1
Df Residuals:                      38   BIC:                            -167.5
Df Model:                           7
Covariance Type:            nonrobust
================================================================================
=
                    coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
-
Intercept         0.1337      0.013      9.971      0.000       0.107
0.161
C(Day)[T.Mon]    -0.0082      0.018     -0.461      0.647      -0.044
0.028
```

```
C(Day)[T.Sat]     -0.0133      0.017     -0.794      0.432      -0.047
0.021
C(Day)[T.Sun]     -0.0144      0.017     -0.865      0.393      -0.048
0.019
C(Day)[T.Thu]     -0.0377      0.018     -2.113      0.041      -0.074
-0.002
C(Day)[T.Tue]     -0.0101      0.018     -0.568      0.574      -0.046
0.026
C(Day)[T.Wed]     -0.0254      0.018     -1.425      0.162      -0.062
0.011
experimental      -0.0049      0.009     -0.538      0.594      -0.023
0.014
==============================================================================
Omnibus:                        0.703   Durbin-Watson:                   0.986
Prob(Omnibus):                  0.704   Jarque-Bera (JB):                0.237
Skew:                          -0.155   Prob(JB):                        0.888
Kurtosis:                       3.168   Cond. No.                         9.27
==============================================================================

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

[ ]: