StatsModels
Statistics in Python

Interactions and ANOVA

Link to Notebook GitHub

Note: This script is based heavily on Jonathan Taylor's class notes http://www.stanford.edu/class/stats191/interactions.html

Download and format data:

In [ ]:
```
from __future__ import print_function
from statsmodels.compat import urlopen
import numpy as np
np.set_printoptions(precision=4, suppress=True)
import statsmodels.api as sm
import pandas as pd
pd.set_option("display.width", 100)
import matplotlib.pyplot as plt
from statsmodels.formula.api import ols
from statsmodels.graphics.api import interaction_plot, abline_plot
from statsmodels.stats.anova import anova_lm

try:
    salary_table = pd.read_csv('salary.table')
except:  # recent pandas can read URL without urlopen
    url = 'http://stats191.stanford.edu/data/salary.table'
    fh = urlopen(url)
    salary_table = pd.read_table(fh)
    salary_table.to_csv('salary.table')

E = salary_table.E
M = salary_table.M
X = salary_table.X
S = salary_table.S
```

Take a look at the data:

In [ ]:
```
plt.figure(figsize=(6,6))
symbols = ['D', '^']
colors = ['r', 'g', 'blue']
factor_groups = salary_table.groupby(['E','M'])
for values, group in factor_groups:
    i,j = values
    plt.scatter(group['X'], group['S'], marker=symbols[j], color=colors[i-1],
               s=144)
plt.xlabel('Experience');
plt.ylabel('Salary');
```

Fit a linear model:

In [ ]:
```
formula = 'S ~ C(E) + C(M) + X'
lm = ols(formula, salary_table).fit()
print(lm.summary())
```

Have a look at the created design matrix:

In [ ]:
```
lm.model.exog[:5]
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      S   R-squared:                       0.957
Model:                            OLS   Adj. R-squared:                  0.953
Method:                 Least Squares   F-statistic:                     226.8
Date:                Mon, 20 Jul 2015   Prob (F-statistic):           2.23e-27
Time:                        17:43:41   Log-Likelihood:                -381.63
No. Observations:                  46   AIC:                             773.3
Df Residuals:                      41   BIC:                             782.4
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept   8035.5976    386.689     20.781      0.000      7254.663  8816.532
C(E)[T.2]   3144.0352    361.968      8.686      0.000      2413.025  3875.045
C(E)[T.3]   2996.2103    411.753      7.277      0.000      2164.659  3827.762
```

```
C(M)[T.1]     6883.5310    313.919     21.928      0.000      6249.559  7517.503
X              546.1840     30.519     17.896      0.000       484.549   607.819
==============================================================================
Omnibus:                        2.293   Durbin-Watson:                   2.237
Prob(Omnibus):                  0.318   Jarque-Bera (JB):                1.362
Skew:                          -0.077   Prob(JB):                        0.506
Kurtosis:                       2.171   Cond. No.                         33.5
==============================================================================

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

Or since we initially passed in a DataFrame, we have a DataFrame available in

In [ ]:
```
lm.model.data.orig_exog[:5]
```

We keep a reference to the original untouched data in

In [ ]:
```
lm.model.data.frame[:5]
```

Influence statistics

In [ ]:
```
infl = lm.get_influence()
print(infl.summary_table())
```

or get a dataframe

In [ ]:
```
df_infl = infl.summary_frame()
```

```
==================================================================================================
       obs      endog     fitted     Cook's   student.   hat diag    dffits   ext.stud.     dffits
                           value          d   residual              internal   residual
--------------------------------------------------------------------------------------------------
         0  13876.000  15465.313      0.104     -1.683      0.155     -0.722     -1.723     -0.739
         1  11608.000  11577.992      0.000      0.031      0.130      0.012      0.031      0.012
         2  18701.000  18461.523      0.001      0.247      0.109      0.086      0.244      0.085
         3  11283.000  11725.817      0.005     -0.458      0.113     -0.163     -0.453     -0.162
         4  11767.000  11577.992      0.001      0.197      0.130      0.076      0.195      0.075
         5  20872.000  19155.532      0.092      1.787      0.126      0.678      1.838      0.698
         6  11772.000  12272.001      0.006     -0.513      0.101     -0.172     -0.509     -0.170
         7  10535.000   9127.966      0.056      1.457      0.116      0.529      1.478      0.537
         8  12195.000  12124.176      0.000      0.074      0.123      0.028      0.073      0.027
         9  12313.000  12818.185      0.005     -0.516      0.091     -0.163     -0.511     -0.161
        10  14975.000  16557.681      0.084     -1.655      0.134     -0.650     -1.692     -0.664
        11  21371.000  19701.716      0.078      1.728      0.116      0.624      1.772      0.640
        12  19800.000  19553.891      0.001      0.252      0.096      0.082      0.249      0.081
        13  11417.000  10220.334      0.033      1.227      0.098      0.405      1.234      0.408
        14  20263.000  20100.075      0.001      0.166      0.093      0.053      0.165      0.053
        15  13231.000  13216.544      0.000      0.015      0.114      0.005      0.015      0.005
        16  12884.000  13364.369      0.004     -0.488      0.082     -0.146     -0.483     -0.145
        17  13245.000  13910.553      0.007     -0.674      0.075     -0.192     -0.669     -0.191
        18  13677.000  13762.728      0.000     -0.089      0.113     -0.032     -0.087     -0.031
        19  15965.000  17650.049      0.082     -1.747      0.119     -0.642     -1.794     -0.659
        20  12336.000  11312.702      0.021      1.043      0.087      0.323      1.044      0.323
        21  21352.000  21192.443      0.001      0.163      0.091      0.052      0.161      0.051
        22  13839.000  14456.737      0.006     -0.624      0.070     -0.171     -0.619     -0.170
        23  22884.000  21340.268      0.052      1.579      0.095      0.511      1.610      0.521
        24  16978.000  18742.417      0.083     -1.822      0.111     -0.644     -1.877     -0.664
        25  14803.000  15549.105      0.008     -0.751      0.065     -0.199     -0.747     -0.198
        26  17404.000  19288.601      0.093     -1.944      0.110     -0.684     -2.016     -0.709
        27  22184.000  22284.811      0.000     -0.103      0.096     -0.034     -0.102     -0.033
        28  13548.000  12405.070      0.025      1.162      0.083      0.350      1.167      0.352
        29  14467.000  13497.438      0.018      0.987      0.086      0.304      0.987      0.304
        30  15942.000  16641.473      0.007     -0.705      0.068     -0.190     -0.701     -0.189
        31  23174.000  23377.179      0.001     -0.209      0.108     -0.073     -0.207     -0.072
        32  23780.000  23525.004      0.001      0.260      0.092      0.083      0.257      0.082
        33  25410.000  24071.188      0.040      1.370      0.096      0.446      1.386      0.451
        34  14861.000  14043.622      0.014      0.834      0.091      0.263      0.831      0.262
        35  16882.000  17733.841      0.012     -0.863      0.077     -0.249     -0.860     -0.249
        36  24170.000  24469.547      0.003     -0.312      0.127     -0.119     -0.309     -0.118
        37  15990.000  15135.990      0.018      0.878      0.104      0.300      0.876      0.299
        38  26330.000  25163.556      0.035      1.202      0.109      0.420      1.209      0.422
        39  17949.000  18826.209      0.017     -0.897      0.093     -0.288     -0.895     -0.287
        40  25685.000  26108.099      0.008     -0.452      0.169     -0.204     -0.447     -0.202
        41  27837.000  26802.108      0.039      1.087      0.141      0.440      1.089      0.441
        42  18838.000  19918.577      0.033     -1.119      0.117     -0.407     -1.123     -0.408
        43  17483.000  16774.542      0.018      0.743      0.138      0.297      0.739      0.295
        44  19207.000  20464.761      0.052     -1.313      0.131     -0.511     -1.325     -0.515
        45  19346.000  18959.278      0.009      0.423      0.208      0.216      0.419      0.214
==================================================================================================
```

In [ ]:
```
df_infl[:5]
```

Now plot the reiduals within the groups separately:

In [ ]:
```
resid = lm.resid
plt.figure(figsize=(6,6));
for values, group in factor_groups:
    i,j = values
    group_num = i*2 + j - 1  # for plotting purposes
    x = [group_num] * len(group)
    plt.scatter(x, resid[group.index], marker=symbols[j], color=colors[i-1],
            s=144, edgecolors='black')
plt.xlabel('Group');
plt.ylabel('Residuals');
```

Now we will test some interactions using anova or f_test

In [ ]:
```
interX_lm = ols("S ~ C(E) * X + C(M)", salary_table).fit()
print(interX_lm.summary())
```

Do an ANOVA check

In [ ]:
```
from statsmodels.stats.api import anova_lm

table1 = anova_lm(lm, interX_lm)
print(table1)

interM_lm = ols("S ~ X + C(E)*C(M)", data=salary_table).fit()
print(interM_lm.summary())

table2 = anova_lm(lm, interM_lm)
print(table2)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      S   R-squared:                       0.961
Model:                            OLS   Adj. R-squared:                  0.955
Method:                 Least Squares   F-statistic:                     158.6
Date:                Mon, 20 Jul 2015   Prob (F-statistic):           8.23e-26
Time:                        17:43:41   Log-Likelihood:                -379.47
No. Observations:                  46   AIC:                             772.9
Df Residuals:                      39   BIC:                             785.7
Df Model:                           6
Covariance Type:            nonrobust
===============================================================================
                  coef    std err          t      P>|t|      [95.0% Conf. Int.]
-------------------------------------------------------------------------------
Intercept     7256.2800    549.494     13.205      0.000      6144.824  8367.736
C(E)[T.2]     4172.5045    674.966      6.182      0.000      2807.256  5537.753
C(E)[T.3]     3946.3649    686.693      5.747      0.000      2557.396  5335.333
C(M)[T.1]     7102.4539    333.442     21.300      0.000      6428.005  7776.903
X              632.2878     53.185     11.888      0.000       524.710   739.865
C(E)[T.2]:X  -125.5147     69.863     -1.797      0.080      -266.826    15.796
C(E)[T.3]:X  -141.2741     89.281     -1.582      0.122      -321.861    39.313
==============================================================================
Omnibus:                        0.432   Durbin-Watson:                   2.179
Prob(Omnibus):                  0.806   Jarque-Bera (JB):                0.590
Skew:                           0.144   Prob(JB):                        0.744
Kurtosis:                       2.526   Cond. No.                         69.7
==============================================================================

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

The design matrix as a DataFrame

In [ ]:
```
interM_lm.model.data.orig_exog[:5]
```

```
   df_resid             ssr  df_diff         ss_diff         F    Pr(>F)
0        41  43280719.492876        0             NaN       NaN       NaN
1        39  39410679.807560        2  3870039.685316  1.914856  0.160964
                            OLS Regression Results
==============================================================================
Dep. Variable:                      S   R-squared:                       0.999
Model:                            OLS   Adj. R-squared:                  0.999
Method:                 Least Squares   F-statistic:                     5517.
Date:                Mon, 20 Jul 2015   Prob (F-statistic):           1.67e-55
Time:                        17:43:41   Log-Likelihood:                -298.74
No. Observations:                  46   AIC:                             611.5
Df Residuals:                      39   BIC:                             624.3
Df Model:                           6
Covariance Type:            nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [95.0% Conf. Int.]
---------------------------------------------------------------------------------------
Intercept            9472.6854     80.344    117.902      0.000      9310.175  9635.196
C(E)[T.2]            1381.6706     77.319     17.870      0.000      1225.279  1538.063
C(E)[T.3]            1730.7483    105.334     16.431      0.000      1517.690  1943.806
C(M)[T.1]            3981.3769    101.175     39.351      0.000      3776.732  4186.022
C(E)[T.2]:C(M)[T.1]  4902.5231    131.359     37.322      0.000      4636.825  5168.222
```

```
C(E)[T.3]:C(M)[T.1]  3066.0351    149.330     20.532      0.000      2763.986  3368.084
X                     496.9870      5.566     89.283      0.000       485.728   508.246
==============================================================================
Omnibus:                       74.761   Durbin-Watson:                   2.244
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1037.873
Skew:                          -4.103   Prob(JB):                    4.25e-226
Kurtosis:                      24.776   Cond. No.                         79.0
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
   df_resid            ssr  df_diff          ss_diff           F        Pr(>F)
0        41  43280719.492876        0              NaN         NaN           NaN
1        39   1178167.864864        2  42102551.628012  696.844466  3.025504e-31
```

The design matrix as an ndarray

In [ ]:
```
interM_lm.model.exog
interM_lm.model.exog_names
```

In [ ]:
```
infl = interM_lm.get_influence()
resid = infl.resid_studentized_internal
plt.figure(figsize=(6,6))
for values, group in factor_groups:
    i,j = values
    idx = group.index
    plt.scatter(X[idx], resid[idx], marker=symbols[j], color=colors[i-1],
            s=144, edgecolors='black')
plt.xlabel('X');
plt.ylabel('standardized resids');
```

Looks like one observation is an outlier.

In [ ]:
```
drop_idx = abs(resid).argmax()
print(drop_idx)  # zero-based index
idx = salary_table.index.drop(drop_idx)

lm32 = ols('S ~ C(E) + X + C(M)', data=salary_table, subset=idx).fit()

print(lm32.summary())
print('\n')

interX_lm32 = ols('S ~ C(E) * X + C(M)', data=salary_table, subset=idx).fit()

print(interX_lm32.summary())
print('\n')


table3 = anova_lm(lm32, interX_lm32)
print(table3)
print('\n')


interM_lm32 = ols('S ~ X + C(E) * C(M)', data=salary_table, subset=idx).fit()

table4 = anova_lm(lm32, interM_lm32)
print(table4)
print('\n')
```

Replot the residuals

In [ ]:
```
try:
    resid = interM_lm32.get_influence().summary_frame()['standard_resid']
except:
    resid = interM_lm32.get_influence().summary_frame()['standard_resid']

plt.figure(figsize=(6,6))
for values, group in factor_groups:
    i,j = values
    idx = group.index
    plt.scatter(X[idx], resid[idx], marker=symbols[j], color=colors[i-1],
            s=144, edgecolors='black')
plt.xlabel('X[~[32]]');
plt.ylabel('standardized resids');
```

```
32
                            OLS Regression Results
==============================================================================
Dep. Variable:                      S   R-squared:                       0.955
Model:                            OLS   Adj. R-squared:                  0.950
Method:                 Least Squares   F-statistic:                     211.7
Date:                Mon, 20 Jul 2015   Prob (F-statistic):           2.45e-26
Time:                        17:43:42   Log-Likelihood:                -373.79
```

```
No. Observations:                  45   AIC:                             757.6
Df Residuals:                      40   BIC:                             766.6
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept   8044.7518    392.781     20.482      0.000      7250.911  8838.592
C(E)[T.2]   3129.5286    370.470      8.447      0.000      2380.780  3878.277
C(E)[T.3]   2999.4451    416.712      7.198      0.000      2157.238  3841.652
C(M)[T.1]   6866.9856    323.991     21.195      0.000      6212.175  7521.796
X            545.7855     30.912     17.656      0.000       483.311   608.260
==============================================================================
Omnibus:                        2.511   Durbin-Watson:                   2.265
Prob(Omnibus):                  0.285   Jarque-Bera (JB):                1.400
Skew:                          -0.044   Prob(JB):                        0.496
Kurtosis:                       2.140   Cond. No.                         33.1
==============================================================================

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

                            OLS Regression Results
==============================================================================
Dep. Variable:                      S   R-squared:                       0.959
Model:                            OLS   Adj. R-squared:                  0.952
Method:                 Least Squares   F-statistic:                     147.7
Date:                Mon, 20 Jul 2015   Prob (F-statistic):           8.97e-25
Time:                        17:43:42   Log-Likelihood:                -371.70
No. Observations:                  45   AIC:                             757.4
Df Residuals:                      38   BIC:                             770.0
Df Model:                           6
Covariance Type:            nonrobust
===============================================================================
                  coef    std err          t      P>|t|      [95.0% Conf. Int.]
-------------------------------------------------------------------------------
Intercept    7266.0887    558.872     13.001      0.000      6134.711  8397.466
C(E)[T.2]    4162.0846    685.728      6.070      0.000      2773.900  5550.269
C(E)[T.3]    3940.4359    696.067      5.661      0.000      2531.322  5349.549
C(M)[T.1]    7088.6387    345.587     20.512      0.000      6389.035  7788.243
X             631.6892     53.950     11.709      0.000       522.473   740.905
C(E)[T.2]:X  -125.5009     70.744     -1.774      0.084      -268.714    17.712
C(E)[T.3]:X  -139.8410     90.728     -1.541      0.132      -323.511    43.829
==============================================================================
Omnibus:                        0.617   Durbin-Watson:                   2.194
Prob(Omnibus):                  0.734   Jarque-Bera (JB):                0.728
Skew:                           0.162   Prob(JB):                        0.695
Kurtosis:                       2.468   Cond. No.                         68.7
==============================================================================

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

   df_resid             ssr  df_diff         ss_diff         F    Pr(>F)
0        40  43209096.482552        0             NaN       NaN       NaN
1        38  39374237.269069        2  3834859.213483  1.850508  0.171042

   df_resid             ssr  df_diff          ss_diff            F        Pr(>F)
0        40  43209096.482552        0              NaN          NaN           NaN
1        38    171188.119937        2  43037908.362615  4776.734853  2.291239e-46
```

Plot the fitted values

In [ ]:

```
lm_final = ols('S ~ X + C(E)*C(M)', data = salary_table.drop([drop_idx])).fit()
mf = lm_final.model.data.orig_exog
lstyle = ['-','--']

plt.figure(figsize=(6,6))
for values, group in factor_groups:
    i,j = values
    idx = group.index
    plt.scatter(X[idx], S[idx], marker=symbols[j], color=colors[i-1],
            s=144, edgecolors='black')
    # drop NA because there is no idx 32 in the final model
    plt.plot(mf.X[idx].dropna(), lm_final.fittedvalues[idx].dropna(),
            ls=lstyle[j], color=colors[i-1])
plt.xlabel('Experience');
plt.ylabel('Salary');
```

From our first look at the data, the difference between Master's and PhD in the management group is different than in the non-management group. This is an interaction between the two qualitative variables management,M and education,E. We can visualize this by first removing the effect of experience, then plotting the means within each of the 6 groups using interaction.plot.

In [ ]:

```
U = S - X * interX_lm32.params['X']
```

```
plt.figure(figsize=(6,6))
interaction_plot(E, M, U, colors=['red','blue'], markers=['^','D'],
        markersize=10, ax=plt.gca())
```

Minority Employment Data

In [ ]:
```
try:
    minority_table = pd.read_table('minority.table')
except:  # don't have data already
    url = 'http://stats191.stanford.edu/data/minority.table'
    minority_table = pd.read_table(url)

factor_group = minority_table.groupby(['ETHN'])

fig, ax = plt.subplots(figsize=(6,6))
colors = ['purple', 'green']
markers = ['o', 'v']
for factor, group in factor_group:
    ax.scatter(group['TEST'], group['JPERF'], color=colors[factor],
                marker=markers[factor], s=12**2)
ax.set_xlabel('TEST');
ax.set_ylabel('JPERF');
```

In [ ]:
```
min_lm = ols('JPERF ~ TEST', data=minority_table).fit()
print(min_lm.summary())
```

In [ ]:
```
fig, ax = plt.subplots(figsize=(6,6));
for factor, group in factor_group:
    ax.scatter(group['TEST'], group['JPERF'], color=colors[factor],
                marker=markers[factor], s=12**2)

ax.set_xlabel('TEST')
ax.set_ylabel('JPERF')
fig = abline_plot(model_results = min_lm, ax=ax)
```

In [ ]:
```
min_lm2 = ols('JPERF ~ TEST + TEST:ETHN',
        data=minority_table).fit()

print(min_lm2.summary())
```

In [ ]:
```
fig, ax = plt.subplots(figsize=(6,6));
for factor, group in factor_group:
    ax.scatter(group['TEST'], group['JPERF'], color=colors[factor],
                marker=markers[factor], s=12**2)

fig = abline_plot(intercept = min_lm2.params['Intercept'],
                 slope = min_lm2.params['TEST'], ax=ax, color='purple');
fig = abline_plot(intercept = min_lm2.params['Intercept'],
        slope = min_lm2.params['TEST'] + min_lm2.params['TEST:ETHN'],
        ax=ax, color='green');
```

In [ ]:
```
min_lm3 = ols('JPERF ~ TEST + ETHN', data = minority_table).fit()
print(min_lm3.summary())
```

In [ ]:
```
fig, ax = plt.subplots(figsize=(6,6));
for factor, group in factor_group:
    ax.scatter(group['TEST'], group['JPERF'], color=colors[factor],
                marker=markers[factor], s=12**2)

fig = abline_plot(intercept = min_lm3.params['Intercept'],
                 slope = min_lm3.params['TEST'], ax=ax, color='purple');
fig = abline_plot(intercept = min_lm3.params['Intercept'] + min_lm3.params['ETHN'],
        slope = min_lm3.params['TEST'], ax=ax, color='green');
```

In [ ]:
```
min_lm4 = ols('JPERF ~ TEST * ETHN', data = minority_table).fit()
print(min_lm4.summary())
```

In [ ]:
```
fig, ax = plt.subplots(figsize=(8,6));
for factor, group in factor_group:
    ax.scatter(group['TEST'], group['JPERF'], color=colors[factor],
                marker=markers[factor], s=12**2)

fig = abline_plot(intercept = min_lm4.params['Intercept'],
                 slope = min_lm4.params['TEST'], ax=ax, color='purple');
fig = abline_plot(intercept = min_lm4.params['Intercept'] + min_lm4.params['ETHN'],
        slope = min_lm4.params['TEST'] + min_lm4.params['TEST:ETHN'],
        ax=ax, color='green');
```

In [ ]:
```
# is there any effect of ETHN on slope or intercept?
table5 = anova_lm(min_lm, min_lm4)
print(table5)
```

In [ ]:
```
# is there any effect of ETHN on intercept
```

```
table6 = anova_lm(min_lm, min_lm3)
print(table6)
```

In [ ]:
```
# is there any effect of ETHN on slope
table7 = anova_lm(min_lm, min_lm2)
print(table7)
```

In [ ]:
```
# is it just the slope or both?
table8 = anova_lm(min_lm2, min_lm4)
print(table8)
```

One-way ANOVA

In [ ]:
```
try:
    rehab_table = pd.read_csv('rehab.table')
except:
    url = 'http://stats191.stanford.edu/data/rehab.csv'
    rehab_table = pd.read_table(url, delimiter=",")
    rehab_table.to_csv('rehab.table')

fig, ax = plt.subplots(figsize=(8,6))
fig = rehab_table.boxplot('Time', 'Fitness', ax=ax, grid=False)
```

In [ ]:
```
rehab_lm = ols('Time ~ C(Fitness)', data=rehab_table).fit()
table9 = anova_lm(rehab_lm)
print(table9)

print(rehab_lm.model.data.orig_exog)
```

```
            df  sum_sq     mean_sq          F    PR(>F)
C(Fitness)   2     672  336.000000  16.961538  0.000041
Residual    21     416   19.809524        NaN       NaN
    Intercept  C(Fitness)[T.2]  C(Fitness)[T.3]
0           1                0                0
1           1                0                0
2           1                0                0
3           1                0                0
4           1                0                0
5           1                0                0
6           1                0                0
7           1                0                0
8           1                1                0
9           1                1                0
10          1                1                0
11          1                1                0
12          1                1                0
13          1                1                0
14          1                1                0
15          1                1                0
16          1                1                0
17          1                1                0
18          1                0                1
19          1                0                1
20          1                0                1
21          1                0                1
22          1                0                1
23          1                0                1
```

In [ ]:
```
print(rehab_lm.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                   Time   R-squared:                       0.618
Model:                            OLS   Adj. R-squared:                  0.581
Method:                 Least Squares   F-statistic:                     16.96
Date:                Mon, 20 Jul 2015   Prob (F-statistic):           4.13e-05
Time:                        17:43:46   Log-Likelihood:                -68.286
No. Observations:                  24   AIC:                             142.6
Df Residuals:                      21   BIC:                             146.1
Df Model:                           2
Covariance Type:            nonrobust
===================================================================================
                      coef    std err          t      P>|t|      [95.0% Conf. Int.]
-----------------------------------------------------------------------------------
Intercept          38.0000      1.574     24.149      0.000        34.728    41.272
C(Fitness)[T.2]    -6.0000      2.111     -2.842      0.010       -10.390    -1.610
C(Fitness)[T.3]   -14.0000      2.404     -5.824      0.000       -18.999    -9.001
==============================================================================
Omnibus:                        0.163   Durbin-Watson:                   2.209
Prob(Omnibus):                  0.922   Jarque-Bera (JB):                0.211
Skew:                          -0.163   Prob(JB):                        0.900
Kurtosis:                       2.675   Cond. No.                         3.80
==============================================================================

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

Two-way ANOVA

In [ ]:
```
try:
    kidney_table = pd.read_table('./kidney.table')
except:
    url = 'http://stats191.stanford.edu/data/kidney.table'
    kidney_table = pd.read_table(url, delimiter=" *")
```

```
/Users/tom.augspurger/Envs/py3/lib/python3.4/site-packages/pandas/io/parsers.py:648: ParserWarning: Falling back t
  ParserWarning)
```

Explore the dataset

In [ ]:
```
kidney_table.groupby(['Weight', 'Duration']).size()
```

Balanced panel

In [ ]:
```
kt = kidney_table
plt.figure(figsize=(8,6))
fig = interaction_plot(kt['Weight'], kt['Duration'], np.log(kt['Days']+1),
        colors=['red', 'blue'], markers=['D','^'], ms=10, ax=plt.gca())
```

You have things available in the calling namespace available in the formula evaluation namespace

In [ ]:
```
kidney_lm = ols('np.log(Days+1) ~ C(Duration) * C(Weight)', data=kt).fit()

table10 = anova_lm(kidney_lm)

print(anova_lm(ols('np.log(Days+1) ~ C(Duration) + C(Weight)',
                data=kt).fit(), kidney_lm))
print(anova_lm(ols('np.log(Days+1) ~ C(Duration)', data=kt).fit(),
               ols('np.log(Days+1) ~ C(Duration) + C(Weight, Sum)',
                   data=kt).fit()))
print(anova_lm(ols('np.log(Days+1) ~ C(Weight)', data=kt).fit(),
               ols('np.log(Days+1) ~ C(Duration) + C(Weight, Sum)',
                   data=kt).fit()))
```

```
   df_resid        ssr  df_diff   ss_diff        F    Pr(>F)
0        56  29.624856        0       NaN      NaN       NaN
1        54  28.989198        2  0.635658  0.59204  0.556748
   df_resid        ssr  df_diff    ss_diff          F    Pr(>F)
0        58  46.596147        0        NaN        NaN       NaN
1        56  29.624856        2  16.971291  16.040454  0.000003
   df_resid        ssr  df_diff   ss_diff         F   Pr(>F)
0        57  31.964549        0       NaN       NaN      NaN
1        56  29.624856        1  2.339693  4.422732  0.03997
```

Sum of squares

Illustrates the use of different types of sums of squares (I,II,II) and how the Sum contrast can be used to produce the same output between the 3.

Types I and II are equivalent under a balanced design.

Don't use Type III with non-orthogonal contrast - ie., Treatment

In [ ]:
```
sum_lm = ols('np.log(Days+1) ~ C(Duration, Sum) * C(Weight, Sum)',
            data=kt).fit()

print(anova_lm(sum_lm))
print(anova_lm(sum_lm, typ=2))
print(anova_lm(sum_lm, typ=3))
```

```
                                 df     sum_sq   mean_sq          F    PR(>F)
C(Duration, Sum)                  1   2.339693  2.339693   4.358293  0.041562
C(Weight, Sum)                    2  16.971291  8.485645  15.806745  0.000004
C(Duration, Sum):C(Weight, Sum)   2   0.635658  0.317829   0.592040  0.556748
Residual                         54  28.989198  0.536837        NaN       NaN
                                    sum_sq  df          F    PR(>F)
C(Duration, Sum)                  2.339693   1   4.358293  0.041562
C(Weight, Sum)                   16.971291   2  15.806745  0.000004
C(Duration, Sum):C(Weight, Sum)   0.635658   2   0.592040  0.556748
Residual                         28.989198  54        NaN       NaN
                                     sum_sq  df           F        PR(>F)
Intercept                        156.301830   1  291.153237  2.077589e-23
C(Duration, Sum)                   2.339693   1    4.358293  4.156170e-02
C(Weight, Sum)                    16.971291   2   15.806745  3.944502e-06
C(Duration, Sum):C(Weight, Sum)    0.635658   2    0.592040  5.567479e-01
Residual                          28.989198  54         NaN           NaN
```

In [ ]:
```
nosum_lm = ols('np.log(Days+1) ~ C(Duration, Treatment) * C(Weight, Treatment)',
            data=kt).fit()
```

```
print(anova_lm(nosum_lm))
print(anova_lm(nosum_lm, typ=2))
print(anova_lm(nosum_lm, typ=3))
```

```
                                            df     sum_sq   mean_sq          F    PR(>F)
C(Duration, Treatment)                       1   2.339693  2.339693   4.358293  0.041562
C(Weight, Treatment)                         2  16.971291  8.485645  15.806745  0.000004
C(Duration, Treatment):C(Weight, Treatment)  2   0.635658  0.317829   0.592040  0.556748
Residual                                    54  28.989198  0.536837        NaN       NaN
                                               sum_sq  df          F    PR(>F)
C(Duration, Treatment)                       2.339693   1   4.358293  0.041562
C(Weight, Treatment)                        16.971291   2  15.806745  0.000004
C(Duration, Treatment):C(Weight, Treatment)  0.635658   2   0.592040  0.556748
Residual                                    28.989198  54        NaN       NaN
                                               sum_sq  df          F    PR(>F)
Intercept                                   10.427596   1  19.424139  0.000050
C(Duration, Treatment)                       0.054293   1   0.101134  0.751699
C(Weight, Treatment)                        11.703387   2  10.900317  0.000106
C(Duration, Treatment):C(Weight, Treatment)  0.635658   2   0.592040  0.556748
Residual                                    28.989198  54        NaN       NaN
```