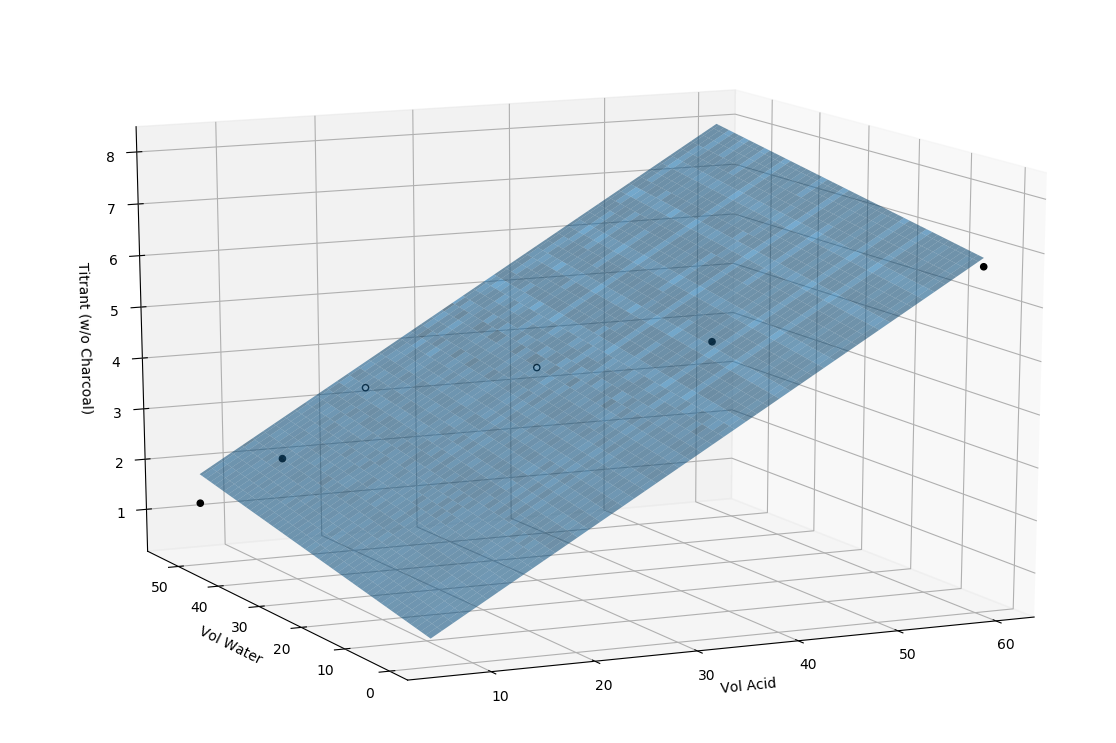
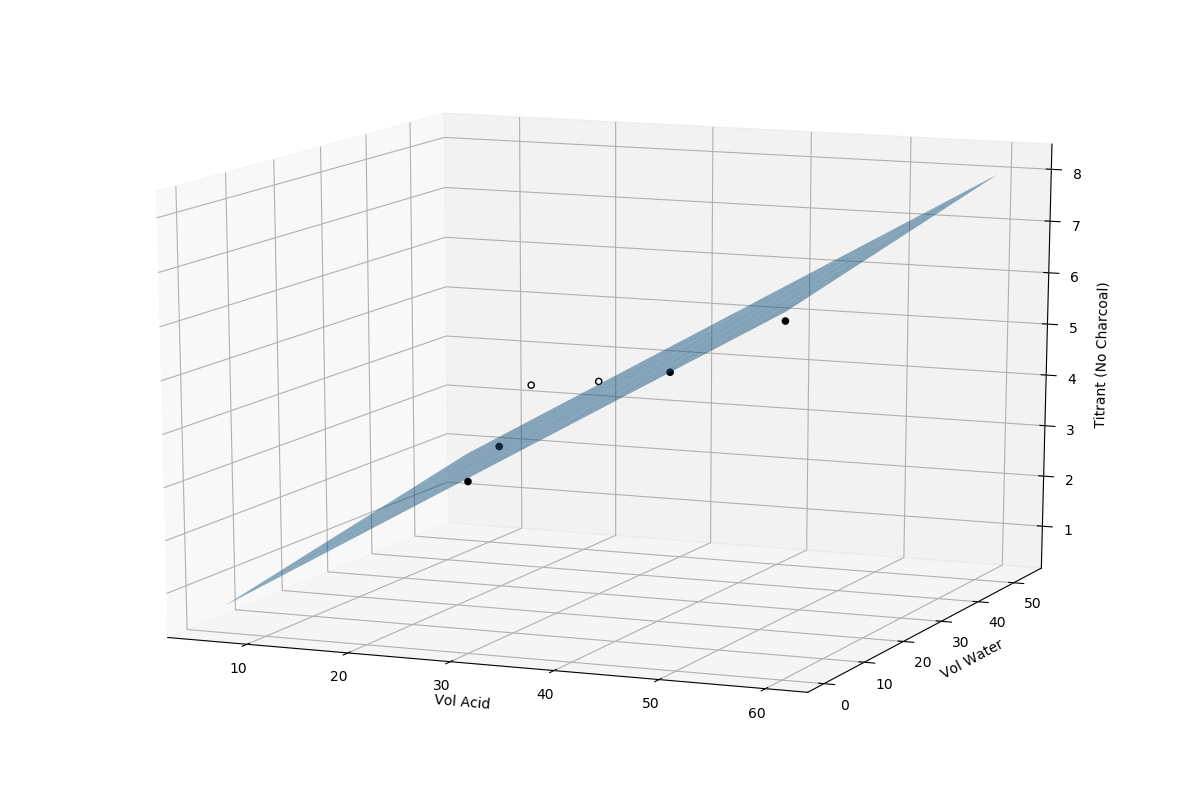
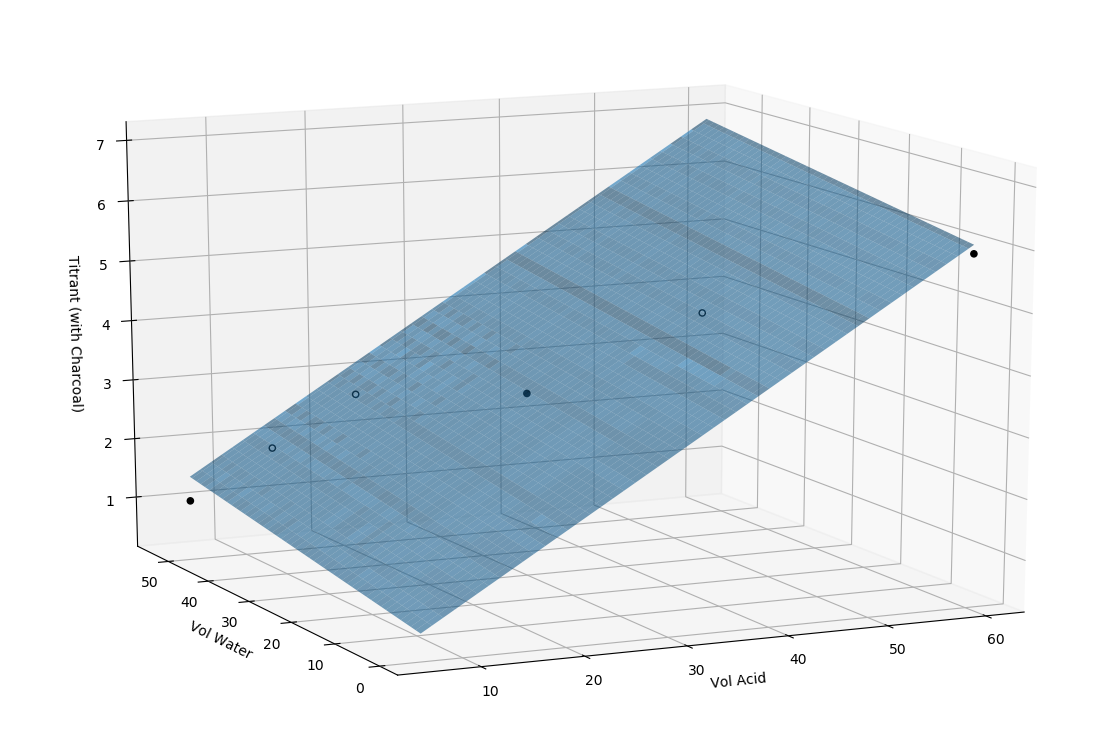
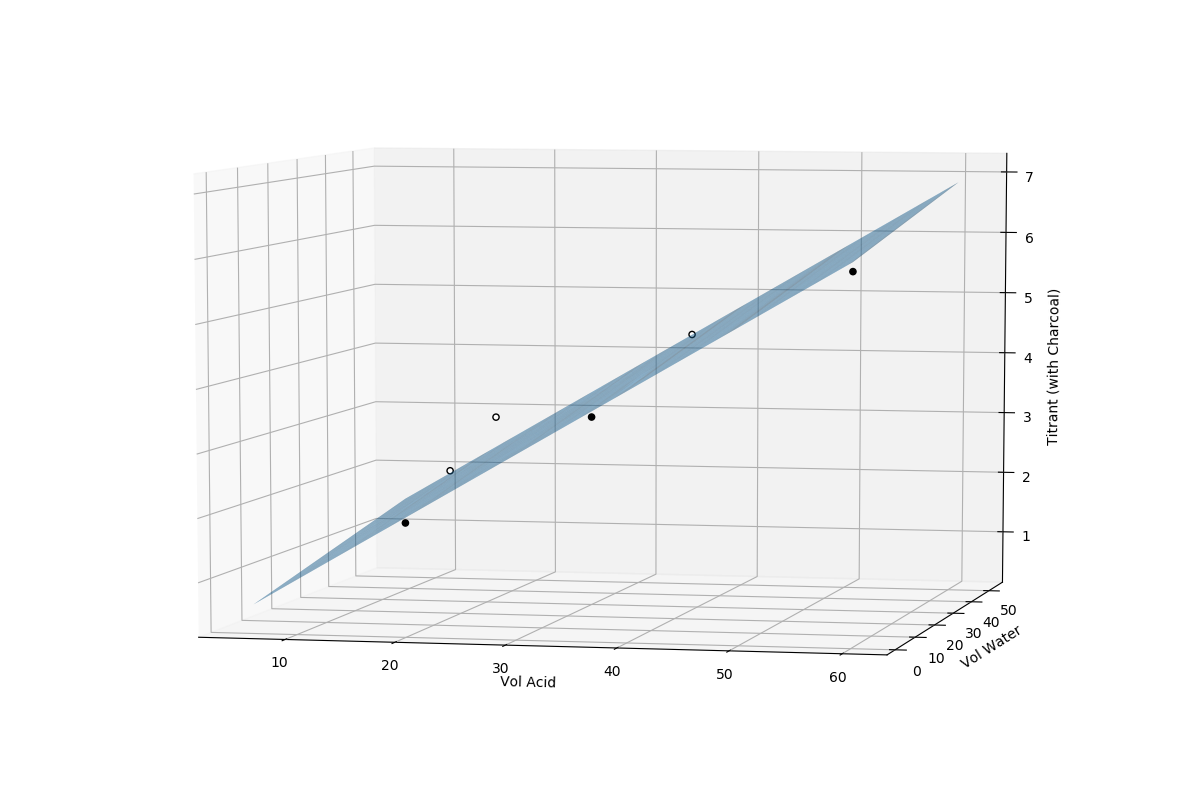
**Statistical Modeling using Python’s Linear Regression**

**Figure 1:** 3D-Surface Plot of our statistical model for untreated assay (view 1)

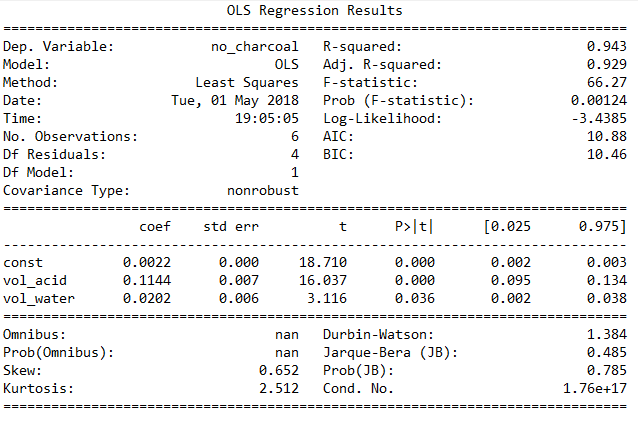


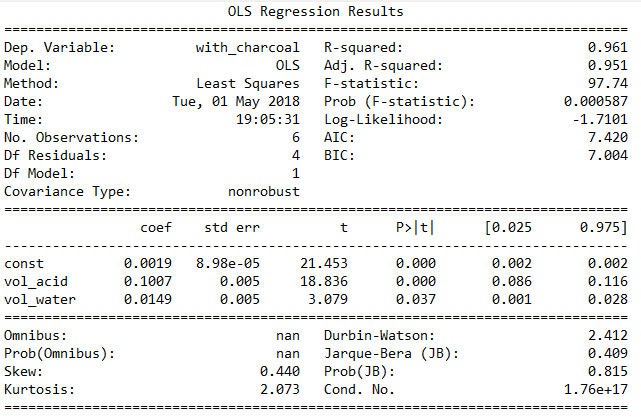
**Figure 2:** 3D-Surface Plot of our statistical model for untreated assay (view 2)



**Figure 3:** 3D-Surface Plot of our statistical model for charcoal-treated assay (view 1) **Figure 4:** 3D-Surface Plot of our statistical model for charcoal-treated assay (view 2)

**Statistical Model Regression Summary**

**Figure 3:** Regression Summary for untreated assay

****

**Figure 3:** Regression Summary for charcoal-treated assay

**Statistical Model Python Code for Untreated Assay**

import pandas as pd

from sklearn import preprocessing, svm, model\_selection

from sklearn.linear\_model import LinearRegression

from matplotlib import pyplot as plt

import numpy as np

from matplotlib import cm

from mpl\_toolkits.mplot3d import axes3d, Axes3D

from random import random, seed

import pickle

import statsmodels.api as sm

df = pd.read\_csv('titration\_data.csv')

x = df[['vol\_acid', 'vol\_water']]

y = df[['no\_charcoal','with\_charcoal']]

x\_train, x\_test, y\_train, y\_test = model\_selection.train\_test\_split(x,y,test\_size=0.2)

*#model = svm.SVR(kernel='linear')*

model = LinearRegression()

model.fit(x\_train, y\_train)

print(model.intercept\_[0])

print(model.coef\_[0])

print(model.coef\_[1])

model\_filename = 'titration.model'

pickle.dump(model, open(model\_filename, 'wb'))

*# Plot*

X = df[['vol\_acid', 'vol\_water']]

y = df['no\_charcoal']

X = sm.add\_constant(X)

est = sm.OLS(y,X).fit()

xx1, xx2 = np.meshgrid(np.linspace(X.vol\_acid.min(), X.vol\_acid.max(), 100),

np.linspace(X.vol\_water.min(), X.vol\_water.max(), 100))

Z = est.params[0] + est.params[1] \* xx1 + est.params[2] \* xx2

fig = plt.figure(figsize=(12, 8))

ax = Axes3D(fig, azim=-115, elev=15)

surf = ax.plot\_surface(xx1, xx2, Z, alpha=0.6, linewidth=0)

resid = y - est.predict(X)

ax.scatter(X[resid >= 0].vol\_acid, X[resid >= 0].vol\_water, y[resid >= 0], color='black', alpha=1.0, facecolor='white')

ax.scatter(X[resid < 0].vol\_acid, X[resid < 0].vol\_water, y[resid < 0], color='black', alpha=1.0)

*# set axis labels*

ax.set\_xlabel('Vol Acid')

ax.set\_ylabel('Vol Water')

ax.set\_zlabel('Titrant (No Charcoal)')

print(est.summary())

plt.show()

***--- end of code (filename: model\_no\_charcoal.py) written May 5, 2018---***

**Statistical Model Python Code for Charcoal-Treated Assay**

import pandas as pd

from sklearn import preprocessing, svm, model\_selection

from sklearn.linear\_model import LinearRegression

from matplotlib import pyplot as plt

import numpy as np

from matplotlib import cm

from mpl\_toolkits.mplot3d import axes3d, Axes3D

from random import random, seed

import pickle

import statsmodels.api as sm

df = pd.read\_csv('titration\_data.csv')

x = df[['vol\_acid', 'vol\_water']]

y = df[['no\_charcoal','with\_charcoal']]

x\_train, x\_test, y\_train, y\_test = model\_selection.train\_test\_split(x,y,test\_size=0.2)

*#model = svm.SVR(kernel='linear')*

model = LinearRegression()

model.fit(x\_train, y\_train)

print(model.intercept\_[0])

print(model.coef\_[0])

print(model.coef\_[1])

model\_filename = 'titration.model'

pickle.dump(model, open(model\_filename, 'wb'))

*# Plot*

X = df[['vol\_acid', 'vol\_water']]

y = df['with\_charcoal']

X = sm.add\_constant(X)

est = sm.OLS(y,X).fit()

xx1, xx2 = np.meshgrid(np.linspace(X.vol\_acid.min(), X.vol\_acid.max(), 100),

np.linspace(X.vol\_water.min(), X.vol\_water.max(), 100))

Z = est.params[0] + est.params[1] \* xx1 + est.params[2] \* xx2

fig = plt.figure(figsize=(12, 8))

ax = Axes3D(fig, azim=-115, elev=15)

surf = ax.plot\_surface(xx1, xx2, Z, alpha=0.6, linewidth=0)

resid = y - est.predict(X)

ax.scatter(X[resid >= 0].vol\_acid, X[resid >= 0].vol\_water, y[resid >= 0], color='black', alpha=1.0, facecolor='white')

ax.scatter(X[resid < 0].vol\_acid, X[resid < 0].vol\_water, y[resid < 0], color='black', alpha=1.0)

*# set axis labels*

ax.set\_xlabel('Vol Acid')

ax.set\_ylabel('Vol Water')

ax.set\_zlabel('Titrant (with Charcoal)')

print(est.summary())

plt.show()

***--- end of code (filename: model\_with\_charcoal.py) written May 5, 2018---***

**Statistical Model Experimental Training Data**

*Format: CSV, Filename: titration\_data.csv*

"vol\_acid","vol\_water","no\_charcoal","with\_charcoal"

6,54,1.2,1

12,48,2.2,2

18,42,3.7,3

30,30,4.3,3.2

42,18,5,4.7

60,0,6.7,5.9

**Statistical Model Equation**

**Not treated with charcoal:**

y = 0.04706704\**vol\_acetic\_acid* + 0.04241155\**vol\_water* + 4.116201117318435

**Treated with charcoal:**

y = 0.04982993\**vol\_acetic\_acid* + 0.04345238\**vol\_water* + 3.898979591836734

**NOTE:** This statistical model for our experiment is uploaded on my github account

**GitHub Account:** http://github.com/kristianespina

**Statistical Model:** https://github.com/kristianespina/Acetic-Acid-Adsorption-Charcoal-Statistical-Model