hw3

September 29, 2015

1 Juan Carlos Apitz

2 STAT510 Homework 3

2.1 Part 1

2.1.1 a.

This is a model without intercept, i.e. $\hat{Y} = b_1 X$. To find the least squares solution b_1, we solve the minimization:

$$\underset{b_1}{min} \|y - b_1 x\|^2$$

$$= \underset{b_1}{min} \left\{ \|y\|^2 - 2b_1 \langle x, y \rangle + b_1^2 \|x\|^2 \right\}$$

Here x and y are n by 1 vectors, $\|.\|$ the L_2 norm, and $\langle ., . \rangle$ the dot product.
We use the first and second derivative tests with respect to b_1 to find the minimum:

$$-2 \langle x, y \rangle + 2b_1 \|x\|^2 \stackrel{set}{=} 0$$

$$b_1 = \frac{\langle x, y \rangle}{\|x\|^2}$$

The second derivative test tells us that b_1 is indeed a minimum:

$$2\|x\|^2 > 0$$

2.1.2 b.

For the variance we take advantage of the fact the $\langle x, x \rangle = \|x\|^2$ and simply take the variance of b_1:

$$V(b_1) = V\left(\frac{\langle x, y \rangle}{\|x\|^2}\right)$$

$$= V(y) \frac{\langle x, x \rangle}{(\|x\|^2)^2}$$

$$= \frac{\sigma^2}{\|x\|^2}$$

since $V(y) = \sigma^2$.

2.1.3 c.

```
In [1]: import numpy as np

        x = np.array([1.,2.,3.])
        y = np.array([1.,4.,3.])

        b1 = np.sum(x*y)/np.sum(x**2)

        print 'the estimate for beta 1 is b1 = %.3f.' %b1
```

```
the estimate for beta 1 is b1 = 1.286.
```

```
In [2]: yhat = b1*x

        n = len(y)

        MSE = np.sum((y-yhat)**2)/(n-1)

        std_err_b1 = MSE/np.sum(x**2)

        print 'the standard error for b1 is %.3f.' %std_err_b1
```

```
the standard error for b1 is 0.102.
```

```
In [3]: from scipy.stats import t

        tcrit = abs(t.ppf(0.025, n-1, loc=0, scale=1))

        print 'the critical t-value for the 95 percent confidence interval is %.3f.' %tcrit
```

```
the critical t-value for the 95 percent confidence interval is 4.303.
```

```
In [4]: low_val = b1-tcrit*std_err_b1

        high_val = b1+tcrit*std_err_b1

        print 'the 95 percent confidence interval for b1 is (%.3f, %.3f).' %(low_val, high_val)
```

```
the 95 percent confidence interval for b1 is (0.847, 1.725).
```

2.2 Part 2.

2.2.1 Problem 5.3

(1) in matrix notation $\hat{Y} = Xb$, which represents the equation:

$$\hat{Y} = \begin{pmatrix} \hat{Y}_1 \\ \hat{Y}_2 \\ \hat{Y}_3 \\ \hat{Y}_4 \end{pmatrix} = \begin{pmatrix} 1 & X_1 \\ 1 & X_2 \\ 1 & X_3 \\ 1 & X_4 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \end{pmatrix}$$

Additionally,

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{pmatrix}$$

then:

$$e = Y - Xb = \begin{pmatrix} Y_1 - b_0 - b_1 X_1 \\ Y_2 - b_0 - b_1 X_2 \\ Y_3 - b_0 - b_1 X_3 \\ Y_4 - b_0 - b_1 X_4 \end{pmatrix} = \begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{pmatrix} - \begin{pmatrix} 1 & X_1 \\ 1 & X_2 \\ 1 & X_3 \\ 1 & X_4 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \end{pmatrix}$$

Furthermore, we know that b is the solution to the normal equations given by:

$$b = \left(X^T X\right)^{-1} X^T Y$$

Then:

$$\hat{Y} = X \left(X^T X\right)^{-1} X^T Y$$

Define the hat matrix:

$$H = X \left(X^T X\right)^{-1} X^T$$

Then we can write the residuals as:

$$e = Y - HY = (I - H)\, Y$$

(2) The sum

$$\sum_{i=1}^{4} X_i e_i = 0$$

can be found by finding the product $X^T e$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ X_1 & X_2 & X_3 & X_4 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{4} e_i \\ \sum_{i=1}^{4} X_i e_i \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Hence in matrix form:

$$X^T e = 0$$

2.3 Problem 5.14

2.3.1 a.

Let:

$$A = \begin{pmatrix} 4 & 7 \\ 2 & 3 \end{pmatrix},\ y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \text{ and } b = \begin{pmatrix} 25 \\ 12 \end{pmatrix}$$

Then in matrix form the system is:

$$Ay = b$$

2.3.2 b.

To solve the system, we use the standard method of Gaussian elimination and reduce the augmented matrix to row-echelon form:

$$\begin{pmatrix} 4 & 7 & 25 \\ 2 & 3 & 12 \end{pmatrix}$$

$$\sim \begin{pmatrix} 1 & \frac{3}{2} & 6 \\ 4 & 7 & 25 \end{pmatrix}$$

$$\sim \begin{pmatrix} 1 & \frac{3}{2} & 6 \\ 0 & 1 & 1 \end{pmatrix}$$

After back substitution the solution is:

$$y = \begin{pmatrix} \frac{9}{2} \\ 1 \end{pmatrix}$$

2.4 Problem 5.17

2.4.1 a.

In matrix form:

$$W = AY$$

or:

$$\begin{pmatrix} W_1 \\ W_2 \\ W_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \end{pmatrix} = \begin{pmatrix} Y_1 + Y_2 + Y_3 \\ Y_1 - Y_2 \\ Y_1 - Y_2 - Y_3 \end{pmatrix}$$

2.4.2 b.

In matrix form:

$$E(W) = AE(Y) = A\mu$$

Where:

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix}$$

So μ_i is the expected value Y_i.
Then we can also write:

$$\begin{pmatrix} E(W_1) \\ E(W_2) \\ E(W_3) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} = \begin{pmatrix} \mu_1 + \mu_2 + \mu_3 \\ \mu_1 - \mu_2 \\ \mu_1 - \mu_2 - \mu_3 \end{pmatrix}$$

2.4.3 c.

In matrix form:

$$Cov\,(W) = Cov\,(AY) = ACov\,(Y)\,A^T$$

Variance-covarince matrix of Y is:

$$Cov\,(Y) = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{pmatrix}$$

Then the variance-covariance matrix of W is:

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & 0 \\ 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & -1 \\ 1 & 0 & -1 \end{pmatrix}$$

Where σ_{ii} is the variance of Y_i.

2.5 Problem 6.5 (b)

In [5]:
```
#PANDAS
import pandas as pd
from pandas import DataFrame, Series

#NUMPY
import numpy as np

#SCIPY t and F distributions
from scipy.stats import t
from scipy.stats import f

#STATMODELS
import statsmodels.formula.api as sm
#import statsmodels.api as sm

#SEABORN plotting
import seaborn as sns

#MATPLOTLIB plotting
import matplotlib.pyplot as plt
%matplotlib inline
```

In [6]:
```
filename = '~/Documents/LinearRegression/STAT510/Kutner/CH6DS/CH06PR05.txt'

df = pd.read_table(filename, delim_whitespace=True, names=['brand_liking','moisture','sweetness
```

In [7]:
```
df.head()
```

Out[7]:

	brand_liking	moisture	sweetness
0	64	4	2
1	73	4	4
2	61	4	2
3	76	4	4
4	72	6	2

```
In [8]: model = sm.ols(formula="brand_liking ~ moisture + sweetness", data=df).fit()

        b0 = model.params[0]
        b1 = model.params[1]
        b2 = model.params[2]
        n = len(df)
```

```
In [9]: print model.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable:           brand_liking   R-squared:                       0.952
Model:                            OLS   Adj. R-squared:                  0.945
Method:                 Least Squares   F-statistic:                     129.1
Date:                Tue, 29 Sep 2015   Prob (F-statistic):           2.66e-09
Time:                        13:54:32   Log-Likelihood:                -36.894
No. Observations:                  16   AIC:                             79.79
Df Residuals:                      13   BIC:                             82.11
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept     37.6500      2.996     12.566      0.000        31.177    44.123
moisture       4.4250      0.301     14.695      0.000         3.774     5.076
sweetness      4.3750      0.673      6.498      0.000         2.920     5.830
==============================================================================
Omnibus:                        0.766   Durbin-Watson:                   2.313
Prob(Omnibus):                  0.682   Jarque-Bera (JB):                0.647
Skew:                           0.049   Prob(JB):                        0.724
Kurtosis:                       2.020   Cond. No.                         35.9
==============================================================================

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

/home/jcapitz/anaconda/lib/python2.7/site-packages/scipy/stats/stats.py:1277: UserWarning: kurtosistest
  "anyway, n=%i" % int(n))
```

The estimated regression function is: $\hat{y} = 37.650 + 4.425x_1 + 4.375x_2$

Where $\hat{y}$ represents the estimated degree of brand liking, x_1 represents moisture content, and x_2 represents the sweetness level.

In this case $b_1 = 4.425$ can be interpreted as the change in the estimated degree brand liking per unit change in moisture content. In other words, if moisture content changes by one unit, we should expect that brand liking will change by 4.425 units in the same direction.

2.6 Problem 6.15 (c)

```
In [10]: filename = '~/Documents/LinearRegression/STAT510/Kutner/CH6DS/CH06PR15.txt'

         df = pd.read_table(filename, delim_whitespace=True, names=['satisfaction','age','severity','an
```

```
In [11]: df.head()
```

```
Out[11]:    satisfaction  age  severity  anxiety
         0            48   50        51      2.3
```

```
1              57   36         46      2.3
2              66   40         48      2.2
3              70   41         44      1.8
4              89   28         43      1.8
```

```
In [12]: model = sm.ols(formula="satisfaction ~ age + severity + anxiety", data=df).fit()

         b0 = model.params[0]
         b1 = model.params[1]
         b2 = model.params[2]
         b3 = model.params[3]
         n = len(df)
```

```
In [14]: print model.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable:           satisfaction   R-squared:                       0.682
Model:                            OLS   Adj. R-squared:                  0.659
Method:                 Least Squares   F-statistic:                     30.05
Date:                Tue, 29 Sep 2015   Prob (F-statistic):           1.54e-10
Time:                        14:18:02   Log-Likelihood:                -169.36
No. Observations:                  46   AIC:                             346.7
Df Residuals:                      42   BIC:                             354.0
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept    158.4913     18.126      8.744      0.000       121.912   195.071
age           -1.1416      0.215     -5.315      0.000        -1.575    -0.708
severity      -0.4420      0.492     -0.898      0.374        -1.435     0.551
anxiety      -13.4702      7.100     -1.897      0.065       -27.798     0.858
==============================================================================
Omnibus:                        5.219   Durbin-Watson:                   2.183
Prob(Omnibus):                  0.074   Jarque-Bera (JB):                2.074
Skew:                          -0.098   Prob(JB):                        0.354
Kurtosis:                       1.978   Cond. No.                         782.
==============================================================================

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

The estimated regression function is: $\hat{y} = 158.4913 - 1.1416x_1 - 0.4420x_2 - 13.4702x_3$

Where $\hat{y}$ represents the estimated patients' satisfaction, x_1 represents the patients' age in years, x_2 represents the severity of illnes index, and x_3 represents the anxiety level.

In this case $b_2 = -0.442$ can be interpreted as the change in the estimated degree of patients' satisfaction per unit change in the index of severity of illness. In other words, if the index of severity of ilness changes by one unit, we should expect that the degree of patients' satisfaction will change by 0.4420 units in the opposite direction.

```
In [ ]:
```