PHYS 250 Homework 10

John Dulin

April 15, 2013

Problem 1

```
def linear_leastsq(x, y, sigma = None):
    """ Returns a1, a2, should include sigma_a1, sigma_a2 """

    if sigma is None:
        sigma = np.ones_like(y)
    else:
        sigma = y.copy.fill(sigma)

    N     = length(x)

    Sx    = sum(x)
    Sy    = sum(y)
    Sxy   = sum(x*y)
    Sxx   = sum(x**2)
    Delta = N * Sxx - Sx**2
    a1    = (Sxx * Sy - Sx * Sxy) / Delta
    a2    = (N * Sxy - Sx * Sy) / Delta

    return [a1, a2]
```

Problem 2

Our previous function can return the a_1 and a_2 values for line fit.

We define a function to calculate chi-squared,

```
def chisq(model, x, y, sigma = None):
    if sigma == None:
```

```
        sigma = np.ones_like(y)
    else:
        sigma = y.copy.fill(sigma)

    return sum(((y - model(x , *params) / sigma )**2)
```

We calculate the degrees of freedom (DoF) as the difference between the number of parameters and number of data points, then divide the chi-squared by the DoF to determine the reduced chi-squared. So the output of the attached code for problem 2 (i) is:

```
Parameter a1 is -4
Parameter a2 is 0
Chi-squared: 15
Reduced Chi-squared: 5
```

* Obviously wrong because the y-intercept is -3....

We would want to remove the outlier (-1,-7) and relaunch the same code to solve for (iii). (Once we found out bug.)

Problem 3

Each bin has a width of $\frac{8}{51} = 0.15686$. We'll shift the bins to the right by half of this width to center them (moving all of the x values to the right by about 0.8).

See attached code for problem outline.