CISC271
Fall 2006
Homework for week 12
in preparation for quiz 5
Solutions

The following questions are from Recktenwald chapter 9.

Questions 1, 7, 8

9-1 I used Matlab more or less as a calculator to perfom the calculations as follows:

```
x = [1  2 4 5 ]
y = [1 2 2 3]
sx = sum(x)
sy = sum(y)
sxy = sum(x.*y)
sxs = sum(x.^2)
alpha = (sx*sy - 4*sxy) / (sx^2 - 4*sxs) = 0.4000
beta = (sx*sxy - sxs*sy)/(sx^2 - 4*sxs) =  0.8000
```

Observe that this yields the same solution as the normal equations.

```
o = [1     1     1     1];
A = [x' o'];
A =
     1     1
     2     1
     4     1
     5     1
```

```
 A'*A\A'*y'
ans =
    0.4000
    0.8000
```

9-7 Recall that for fitting a polynomial curve to minimize the least square error we set some derivatives to zero and solve for the polynomial coefficients. The general form for this derivative is:

$$\frac{\partial}{\partial a_k}(\sum_{i=1}^{m}[y_i - \sum_{j=0}^{n} a_j x_i^j]^2) = 2\sum_{i=1}^{m}(\sum_{j=0}^{n} a_j x_i^j - y_i)(x_i)^k$$

Where the data points are indexed $1...m$ and the polynomial coefficients from $0..n$. Our goal is to set the equation to zero and solve for the a_i values for $i = 0 \ldots n$. Furthermore letting:

$$g_{ik} = \sum_{i=1}^{m}(x_i)^{j+k} \text{ and } p_k = \sum_{i=1}^{m} y_i(x_i)^k$$

we can write the equations we solve as:

$$\sum_{j=0}^{n} a_j g_{jk} = p_k.$$

In each of the questions we solve a special case of the previous equation.

a) $y = cx$. Here we have a polynomial of degree 1, where the a_0 coefficient is equal to zero. We just need to solve the equation

$$a_1 g_{11} = p_1$$

for a_1 or in this case it is called c.

We get $c = \frac{p_1}{g_{11}}$.

The matlab one liner is:

```
>> c = sum(y.*x) / sum(x.^2)
```

b)$y = cx^2$ Using the same reasoning as above we get $c = \frac{p_2}{g_{22}}$. The corresponding matlab one liner is:

```
>>c = sum(y.*x.^2) / sum(x.^4)
```

c)$y = x^c$. Take the natural log of both sides of the equation to get $\ln y = c \ln x$. Now change variables so that $v = \ln y$ and $u = \ln x$, leaving us with $v = cu$. This is a problem we solved in part a). Putting it all together we get the matlab one liner:

```
>>c = sum(log(y).*log(x))/sum(log(x).^2)
```

9 - 8 Here's my m-file.

```
function c = expfitDR(x,y)
ly = log(y);
ctemp = linefit(x,ly);
c(1) = exp(ctemp(2));
c(2) = ctemp(1);
```

I plotted the curve I got with the given data. It seems to fit exactly! Here is the plot.