Regression Analysis

Linear Estimation of Y given X:

Problem: Given measurement Y, estimate X.

Why? You want to know something that is difficult to measure, such as engine thrust. You estimate this based upon something that is easier to measure. For example, for a jet engine, you want to know thrust. That's what keeps the airplane in the air. Thrust is hard to measure, however. Instead, measure something that's closely related to thrust but is easy to measure, such as fan speed or the pressure difference across the engine.

Since the estimate is different from the 'true' value, denote

$\hat{x}$	The estimate of x
x	The 'true' value of x
$\bar{x}$	The mean of x
X	A matrix that's a function of x

Form an estimate based upon Y using a linear curve fit:

$$\hat{y} = ax + b$$

Least Squares

Procedure to find the parameters 'a' and 'b' given n data points:

Step 1) Write this in matrix form:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \end{bmatrix} = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \\ \vdots & \vdots \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

or

$$Y = XA$$

You can't invert matrix Y since it's not square. To make it square, multiply by Y transpose:

$$X^T Y = X^T X A$$

YTY is square and is usually invertable. Solve for A:

$$A = (X^T X)^{-1} X^T X$$

This is the least squares solution for a and b.

Example: Find the least squares curve fit for the following data points (x,y)

```
 x     y
0.   - 1.
1.     2.
```

```
       2.     7.
       3.     8.
       4.     19.
```

Solution: Create matrix X:

```
     -->X = [x, x.^0]
         0.     1.
         1.     1.
         2.     1.
         3.     1.
         4.     1.
```

Create matrix Y (the true value)

```
     -->Y = y;
```

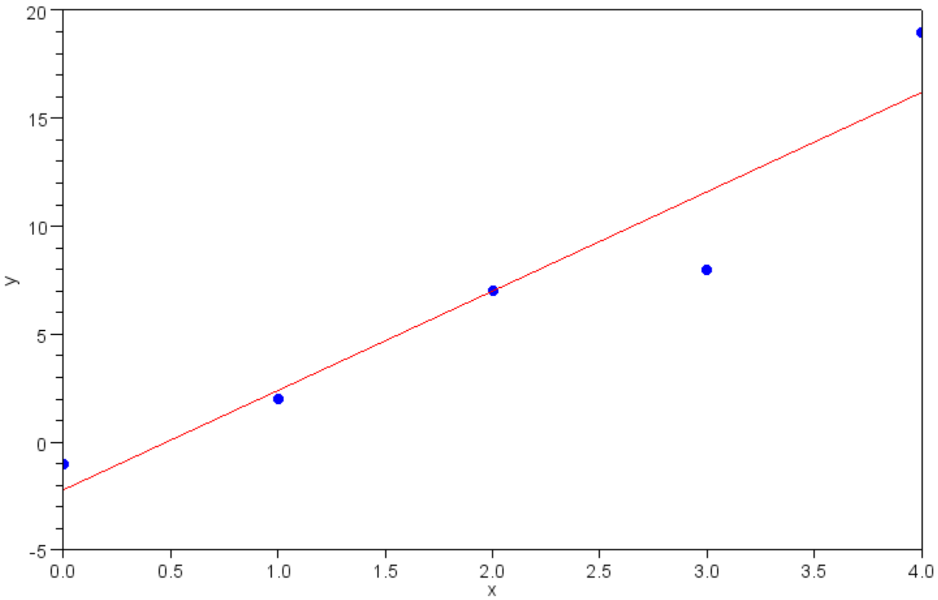
Determine 'a' and 'b'

```
     -->A = inv(X'*X)*X'*Y
         4.6
       - 2.2
```

So, the least squares estimate for y(x) is:

$$\hat{y} \approx 4.6x - 2.2$$

Plotting this function vs your data looks like the following:



Least squares solution for y = ax + b given equal weighting for all data.

This function minimizes the sum squared error, J:

$$J = \sum (y_i - \hat{y}_i)^2$$

Hence the name least squares.

Weighted Least squares:

If you 'trust' some data points more than others, you can weight the data. For example, suppose you weight (trust) the 4th data point 10.6 times more than the rest.

x	y	q (weight)
0.	- 1.	1
1.	2.	1
2.	7.	1
3.	8.	10.6
4.	19.	1

Create a diagonal matrix, Q, which has the weight for each element:

```
-->Q = diag([1,1,1,10.6,1])
 Q  =

    1.    0.    0.    0.      0.
    0.    1.    0.    0.      0.
    0.    0.    1.    0.      0.
    0.    0.    0.    10.6    0.
    0.    0.    0.    0.      1.
```

Return to the equation for X and Y in matrix form:

$$Y = X\,A$$

Multiply by Q

$$QY = QX\,A$$

Multiply by X transpose

$$X^T\,QY = X^TQX\,A$$

Invert

$$(X^TQX)^{-1}\,X^T\,QY = A$$

The results is the least squares solution with weighting Q:

$$J = \sum q_i (y_i - \hat{y}_i)^2$$

Going back to our example:

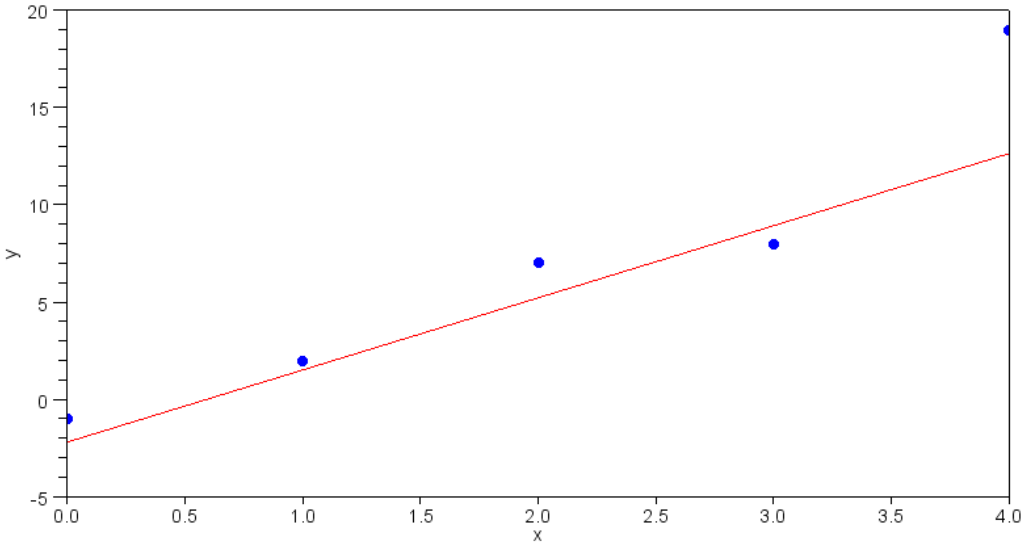
```
-->A = inv(X'*Q*X)*X'*Q*Y
 A  =

    3.7092784
  - 2.2
```

so now the estimate for y should be:

$$\hat{y} = 3.70927x - 2.2$$

Checking by plotting this vs. your data:

```
-->y1 = 3.7092784*x1 - 2.2;
-->plot(x,y,'.',x1,y1,'-r')

-->xlabel('x')
-->ylabel('y')
```



Weighted Least squares solution for y = ax + b with the 4th data point having a weight of 10.6

Note that the line is drawn closer to the 4th data point (3,8) due to assigning it a weight of 10.6.

Covariance and Correlation Coefficient

The correlation between X and Y tells you how closely the two are related

- Correlation of zero means they are independent
- Correlation of +1.000 means that as X increases, Y increases.
- Correlation of -1.000 means that as X increases, Y decreases.

Correlation doesn't care about cause and affect: it just tells you whether the two behave the same way. This is useful in Wall Street: if you can find something that has a strong correlation with stock prices one month or one year in the future, you can use it as a buy / sell indicator. That's one of the ways stock brokers make their money.

To determine the correlation coefficient, you first need to determine the covariance between X and Y.

Covariance:

The covariance between X and Y is defined as

$$Cov[X,Y] = E[(x-\bar{x})(y-\bar{y})]$$

Doing some algebra

$$Cov[X,Y] = E[(x-\bar{x})(y-\bar{y})]$$
$$= E[xy] - \bar{x}\cdot\bar{y}$$

The correlation coefficient is defined as

$$\rho_{X,Y} = \frac{Cov[X,Y]}{\sqrt{\sigma_X^2\cdot\sigma_Y^2}}$$

The correlation coefficient is a measure of how certain you are of you're estimate of y.

- $\rho = \pm 1$ x and y are 100% correlated. If you know x you know y with no error.
- $\rho = 0$ x and y have no correlation. Knowing x tells you nothing about y.

Some other useful relationships are

1st moment (m1)

$$m_1 = mean(x)$$

2nd moment (m2)

$$m_2 = mean(x^2)$$

Variance

$$\sigma^2 = m_2 - m_1^2$$

Covariance:

$$Cov(X,Y) = mean(xy) - mean(x)\ mean(y)$$

Correlation coefficient

$$\rho_{X,Y} = \left(\frac{Cov(X,Y)}{\sqrt{\sigma_x^2\sigma_y^2}}\right)$$

Examples: Let's generate a random variable which is 100% determined by x:

y0 = 2x + 3

and one which is pure noise with a similar standard deviation as y0:

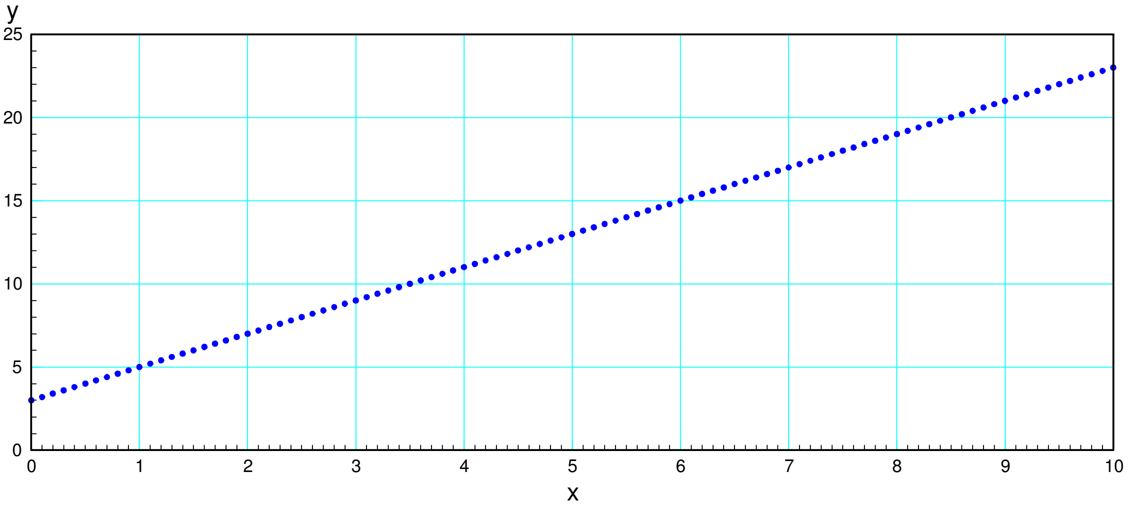
$$y_1 \sim N(0,6)$$

Create y from y0 and y1:

$$y = \alpha\cdot y_0 + (1-\alpha)\cdot y_1$$

Determine how the correlation coefficient varies with alpha.

a) $\alpha = 1$ (no noise)

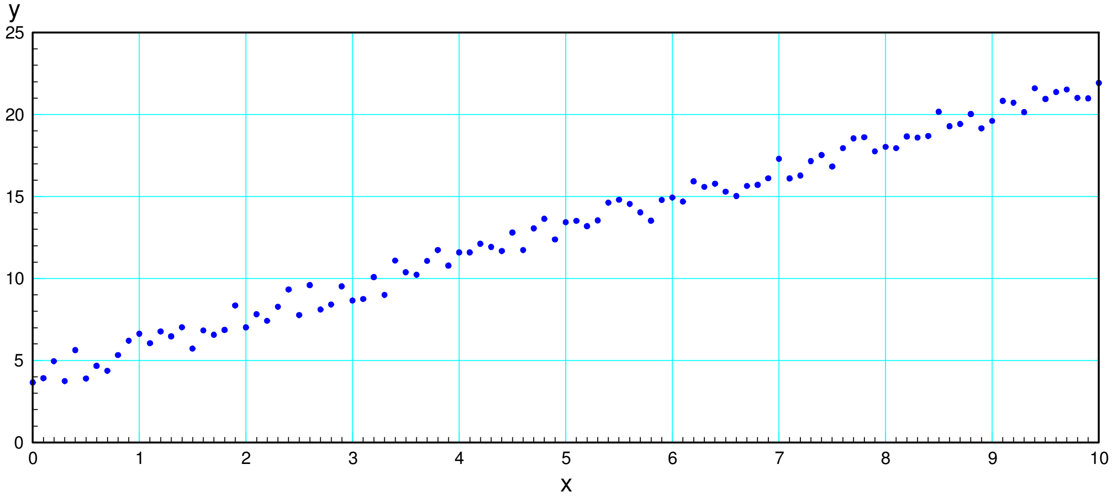


```
x = [0:0.1:10]';
n = 10*rand(size(x0));

P = zeros(100,1);

x0 = 1.0*x + 0.0*n;

y = 2*x0 + 3;
s2x = mean(x.^2) - mean(x)^2;
s2y = mean(y.^2) - mean(y)^2;
Cov = mean(x.*y) - mean(x)*mean(y)
p2 = Cov / sqrt(s2x*s2y)

Cov =   17.0000

p2 =    1.0000
```

There is no noise so the correlation coefficient is 1

90% data, 10% noise

Next, suppose Y is 90% from y0 and 10% from y1. This data looks like the following:

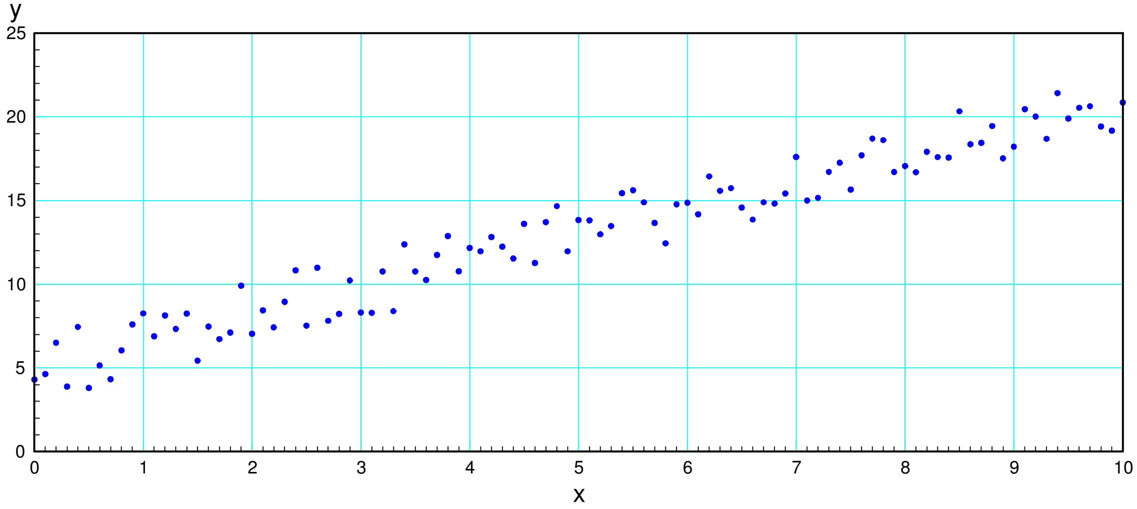


Find correlation coefficient:

```
x = [0:0.1:10]';
n = 10*rand(size(x0));

P = zeros(100,1);

x0 = 0.9*x + 0.1*n;

y = 2*x0 + 3;
s2x = mean(x.^2) - mean(x)^2;
s2y = mean(y.^2) - mean(y)^2;
Cov = mean(x.*y) - mean(x)*mean(y)
p2 = Cov / sqrt(s2x*s2y)

Cov =    15.5010

p2 =    0.9943
```

Note: You have quite a bit of noise but still report $\rho^2 = 0.9943$

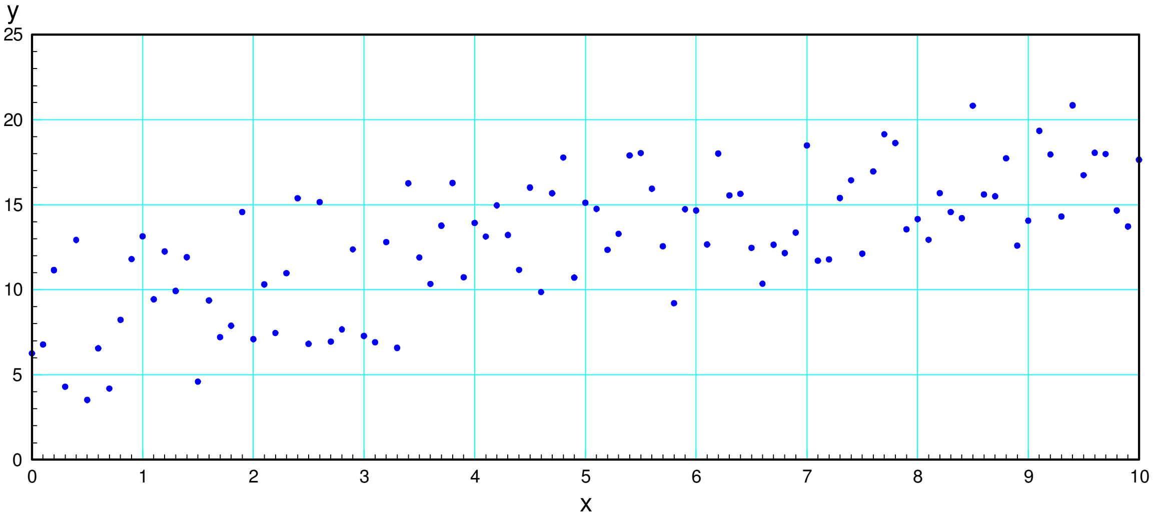
80% data / 20% noise



```
x = [0:0.1:10]';
n = 10*rand(size(x0));

x0 = 0.8*x + 0.2*n;

y = 2*x0 + 3;
s2x = mean(x.^2) - mean(x)^2;
s2y = mean(y.^2) - mean(y)^2;
Cov = mean(x.*y) - mean(x)*mean(y)
p2 = Cov / sqrt(s2x*s2y)

plot(x,y,'.')

Cov =   13.8326

p2 =    0.9700
```

With 20% noise, you report $\rho^2 = 0.9700$

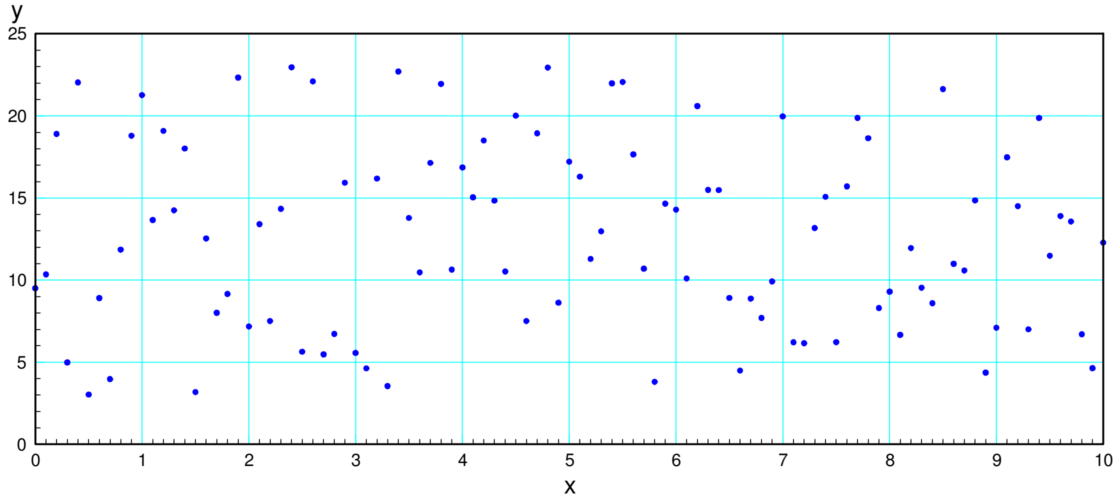
50% data / 50% noise



```
x = [0:0.1:10]';
n = 10*rand(size(x0));

x0 = 0.5*x + 0.5*n;

y = 2*x0 + 3;
s2x = mean(x.^2) - mean(x)^2;
s2y = mean(y.^2) - mean(y)^2;
Cov = mean(x.*y) - mean(x)*mean(y)
p2 = Cov / sqrt(s2x*s2y)

plot(x,y,'.')

Cov =     8.3611

p2 =    0.7074
```

50% noise still reports $\rho^2 = 0.7074$

0% Data, 100% Noise

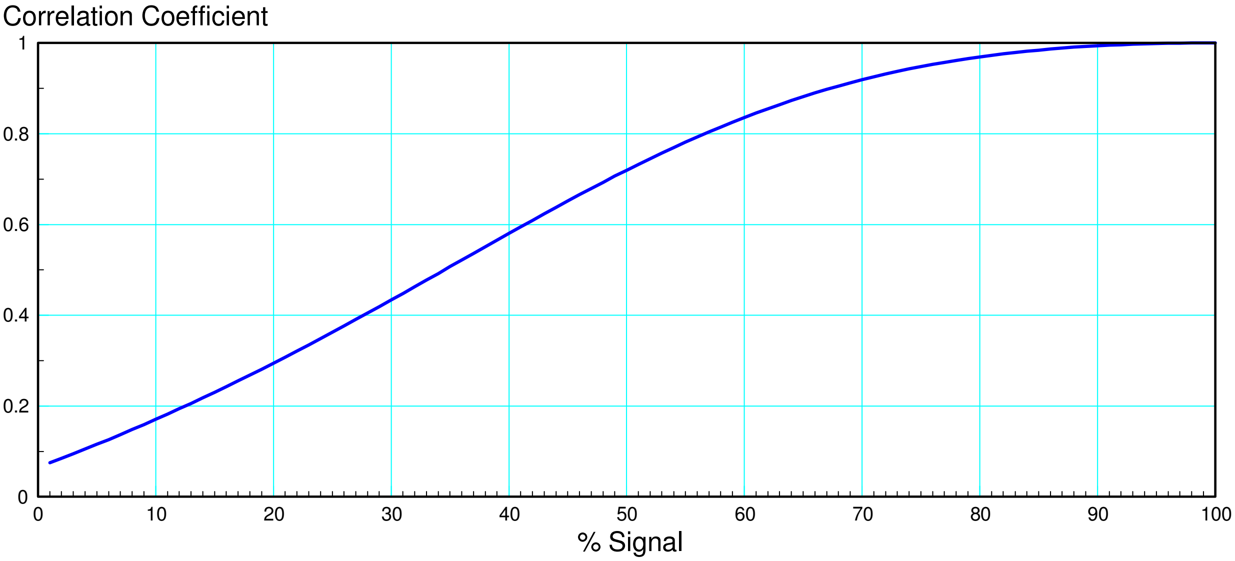


```
x = [0:0.1:10]';
n = 10*rand(size(x0));

x0 = 0.0*x + 1.0*n;

y = 2*x0 + 3;
s2x = mean(x.^2) - mean(x)^2;
s2y = mean(y.^2) - mean(y)^2;
Cov = mean(x.*y) - mean(x)*mean(y)
p2 = Cov / sqrt(s2x*s2y)

plot(x,y,'.')

Cov =    4.0005

p2 =   0.2494
```

100% noise reports 24.9% correlation.

Question: What's the relationship between the correlation coefficient and the contribution of the signal to your measurement?

```
x = [0:0.1:10]';
n = 10*rand(size(x));

P = zeros(100,1);

for i=1:100
   a = i/100;
   x0 = a*x + (1-a)*n;

   y = 2*x0 + 3;
   s2x = mean(x.^2) - mean(x)^2;
   s2y = mean(y.^2) - mean(y)^2;
   Cov = mean(x.*y) - mean(x)*mean(y)
   p2 = Cov / sqrt(s2x*s2y)
   P(i) = p2;
   end

plot(P)
```



Moral: Don't be impressed with correlation coefficients less than 0.8