

---

# **Regression Analysis**

## **ECE 341: Random Processes**

### **Lecture #21**

note: All lecture notes, homework sets, and solutions are posted on [www.BisonAcademy.com](http://www.BisonAcademy.com)

---

---

## Linear Estimation of Y given X:

Problem: Given measurement Y, estimate X.

- You want to know something that is difficult to measure. You estimate this based upon something that is easier to measure.
  - Fan speed  $\approx$  thrust for a jet engine (GE)
  - Pressure drop  $\approx$  thrust (Pratt & Whitney)

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

$B$  Basis matrix: functions 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

$$Y = BA$$

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

$$B^T Y = B^T B \cdot A$$

---

---

$B^T B$  is square and is usually invertable.

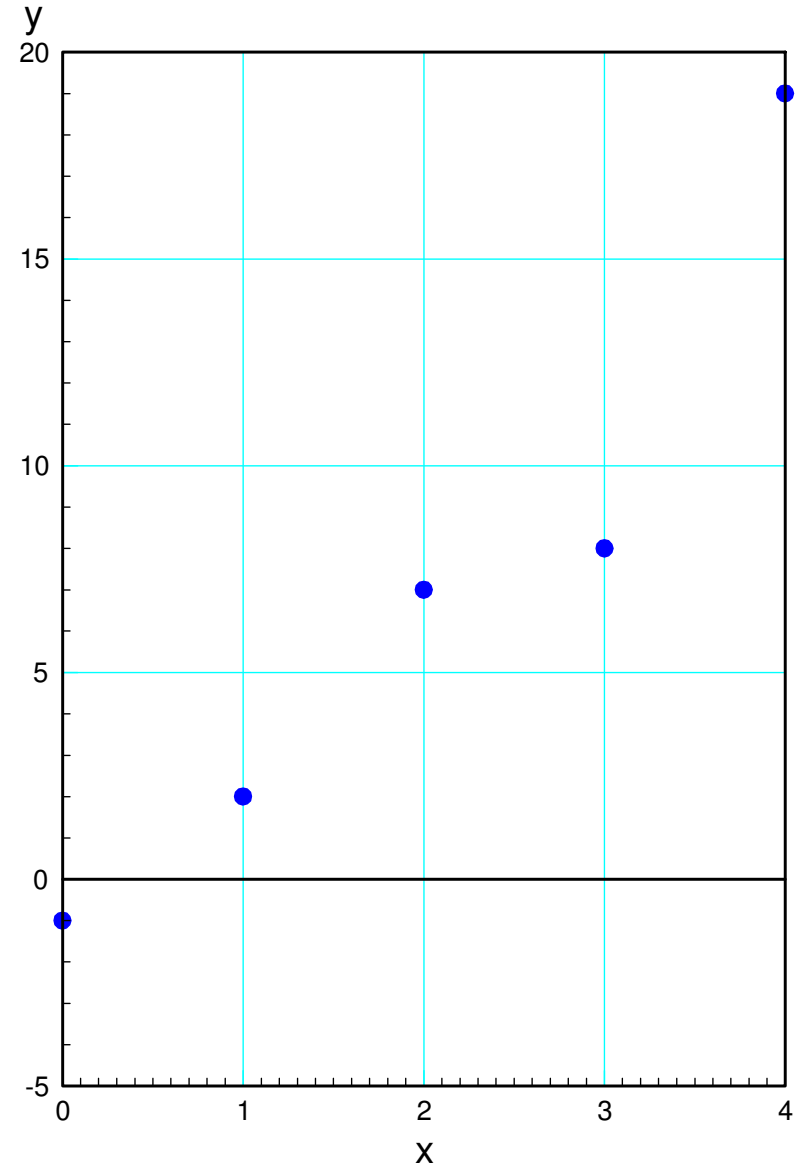
Solve for A:

$$A = \left(B^T B\right)^{-1} B^T Y$$

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 B that defines your basis functions:

```
B = [x, x.^0]
     0.    1.
     1.    1.
     2.    1.
     3.    1.
     4.    1.
```

Determine 'a' and 'b'

```
A = inv(B'*B)*B'*y
     4.6      times x
    - 2.2      times 1
```

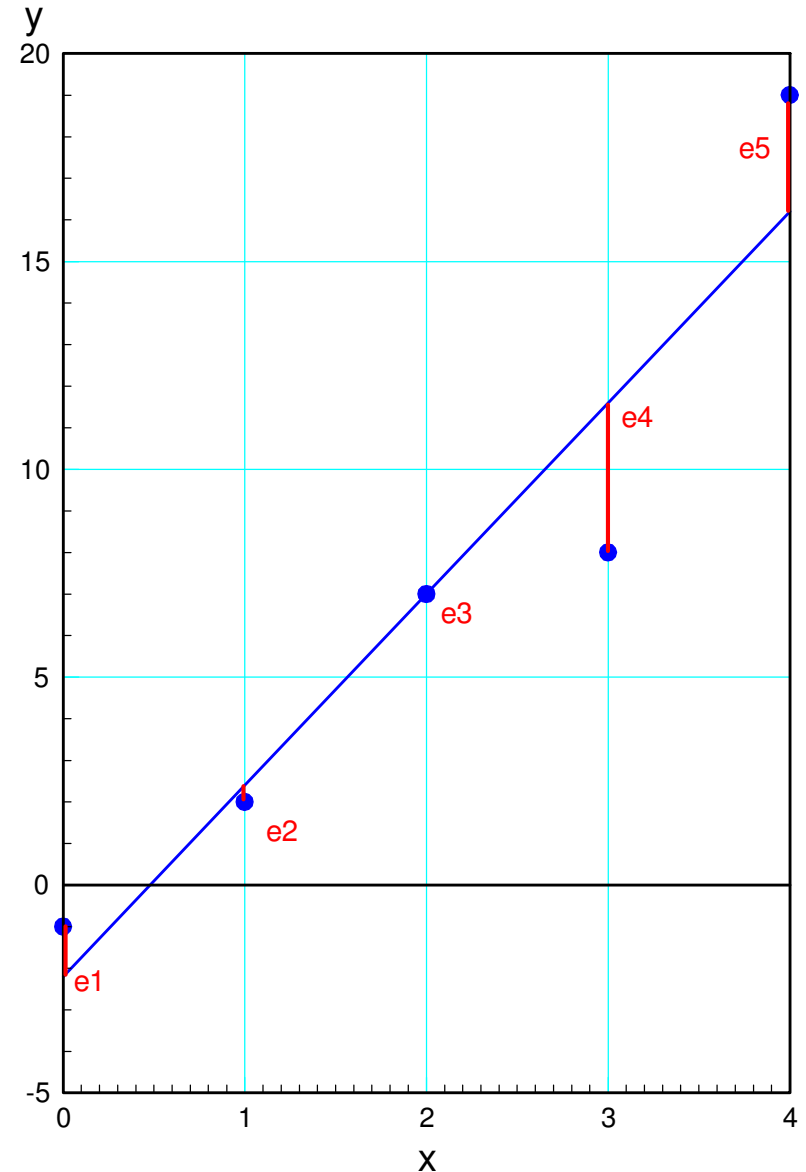
```
plot(x, y, 'b.', x, y, 'r-');
```

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

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

This minimizes the sum-squared error

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



---

## 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])
```

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:

$$Y = B A$$

Multiply by Q

$$QY = QB A$$

Multiply by X transpose

$$B^T QY = B^T QB A$$

Invert

$$(B^T QB)^{-1} B^T QY = A$$

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:

```
-->Q = diag([1,1,1,10.6,1])  
-->A = inv(B'*Q*B)*B'*Q*Y
```

```
3.7092784  
- 2.2
```

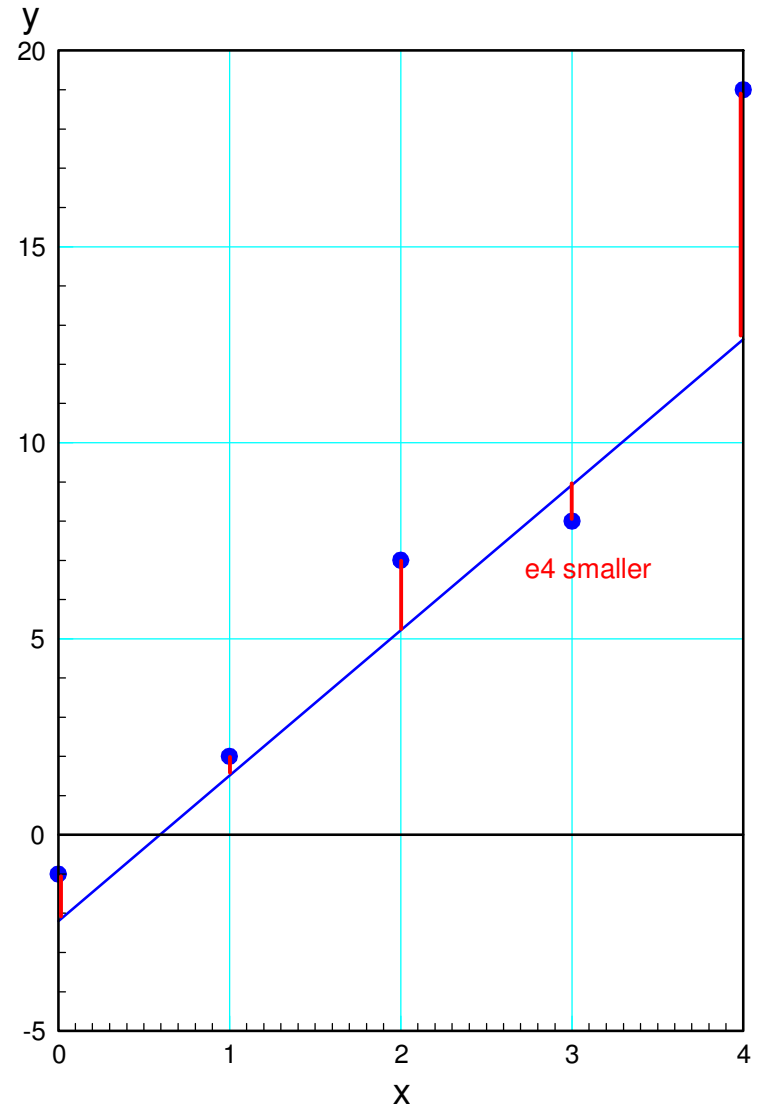
so now the estimate for y should be:

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

Checking by plotting this vs. your data:

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

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





## Example: Arctic Sea Ice

- Source: National Sea and Ice Data Center

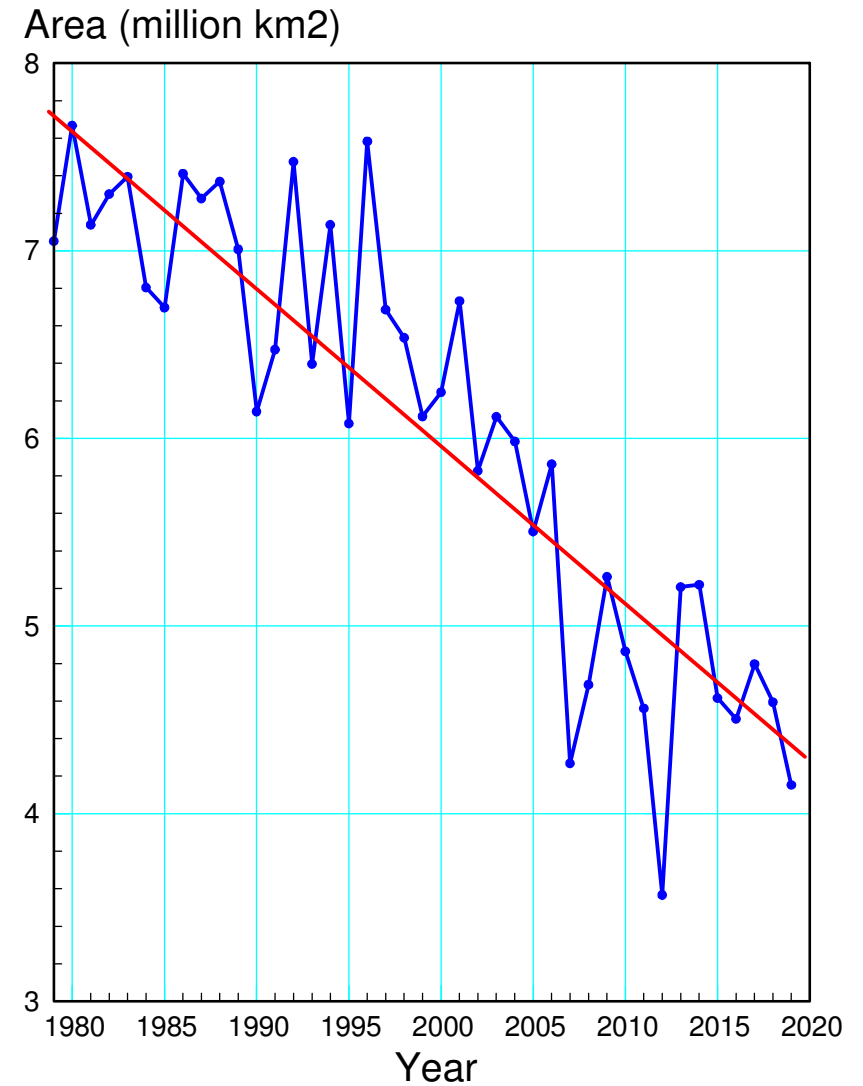
```
B = [year, year.^0];  
Y = [ice];
```

```
A = inv(B'*B)*B'*Y
```

```
- 0.0844726  
 174.68702
```

```
plot(y,a,'b.-',y,X*A,'r')
```

$$\text{Area} \approx -0.0844 \cdot \text{year} + 174.68$$



# Data Analysis

When will the Arctic be ice free?

- First time in 5 million years
- Find the zero crossing

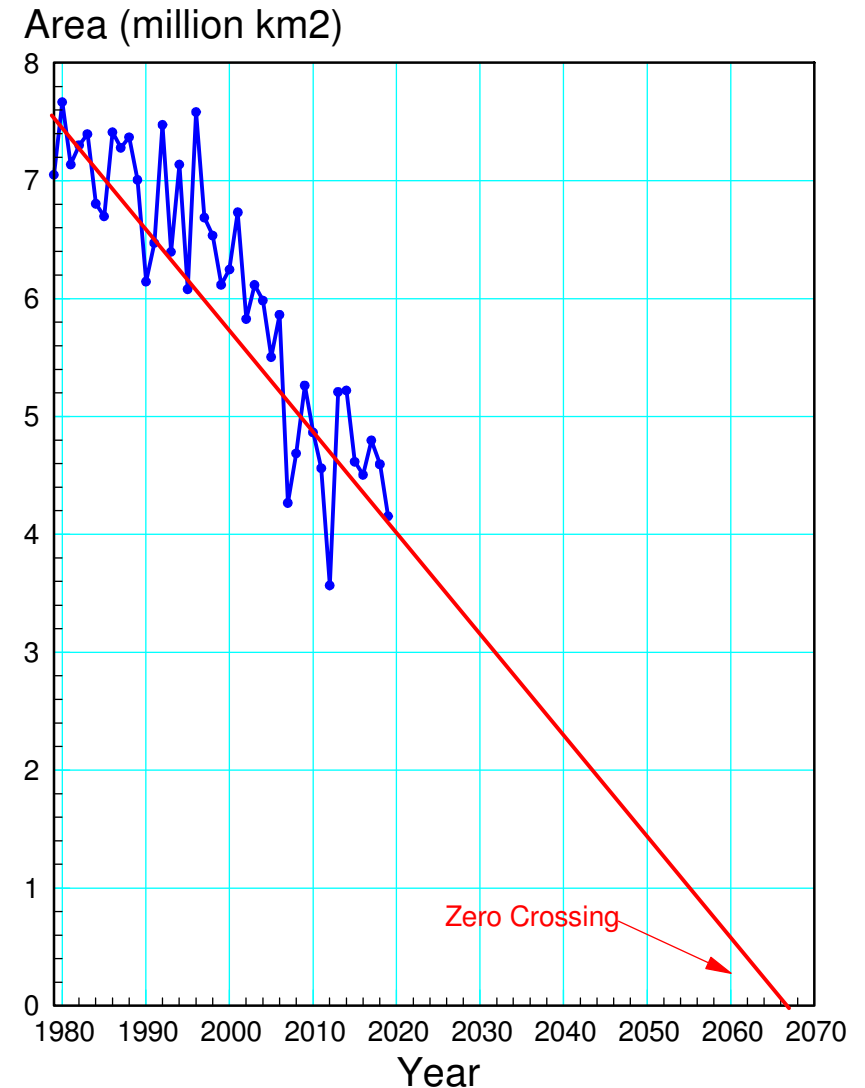
$$\text{Area} \approx 0 = -0.0844 \cdot \text{year} + 174.68$$

$$\text{year} = \left( \frac{174.68}{0.0844} \right) = 2067.97$$

`roots()` also works

```
roots(A)  
2067.9729
```

Using a linear curve fit, the data predicts that the Arctic will be ice free for the first time in 5 million years in the year 2067.



# Example: Fargo Temperatures

Source: Hector Airport

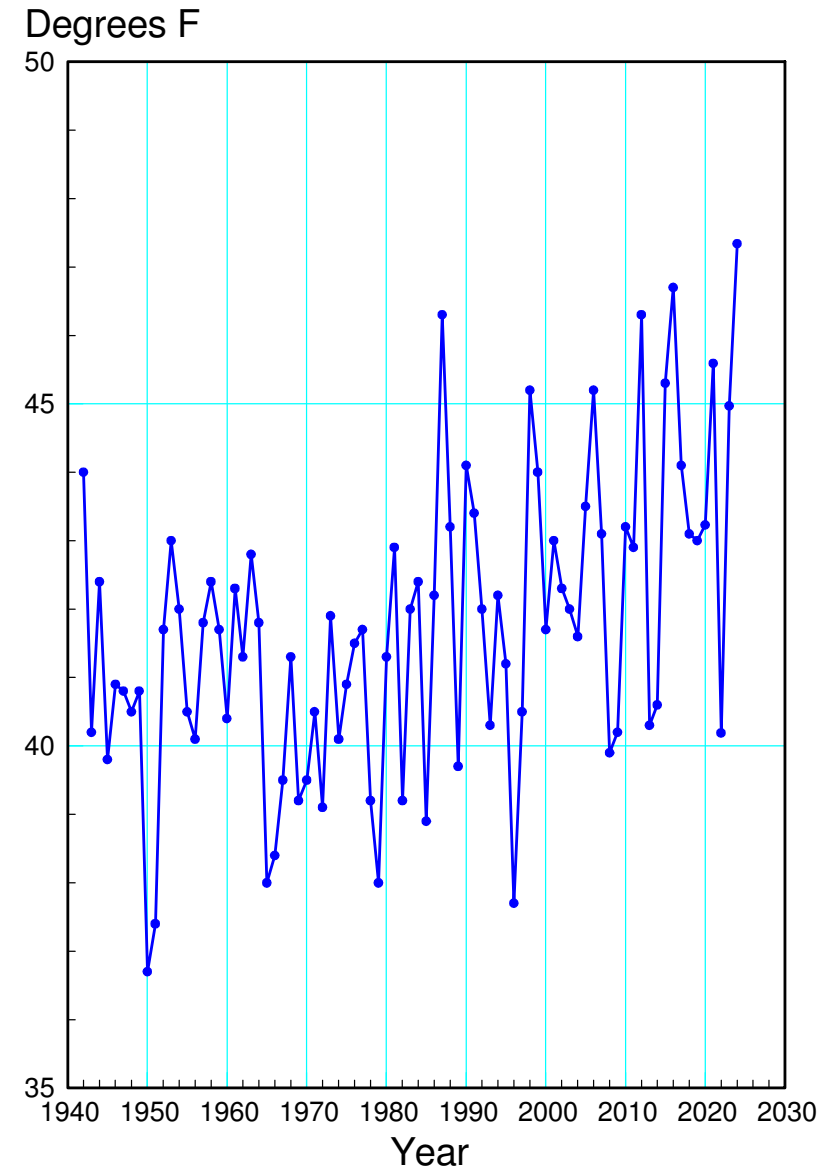
- Mean Temperature in Fargo
- Is there a trend?

Express this in the form of

$$F = ay + b$$

where

- F is the mean temperature and
- y is the year.

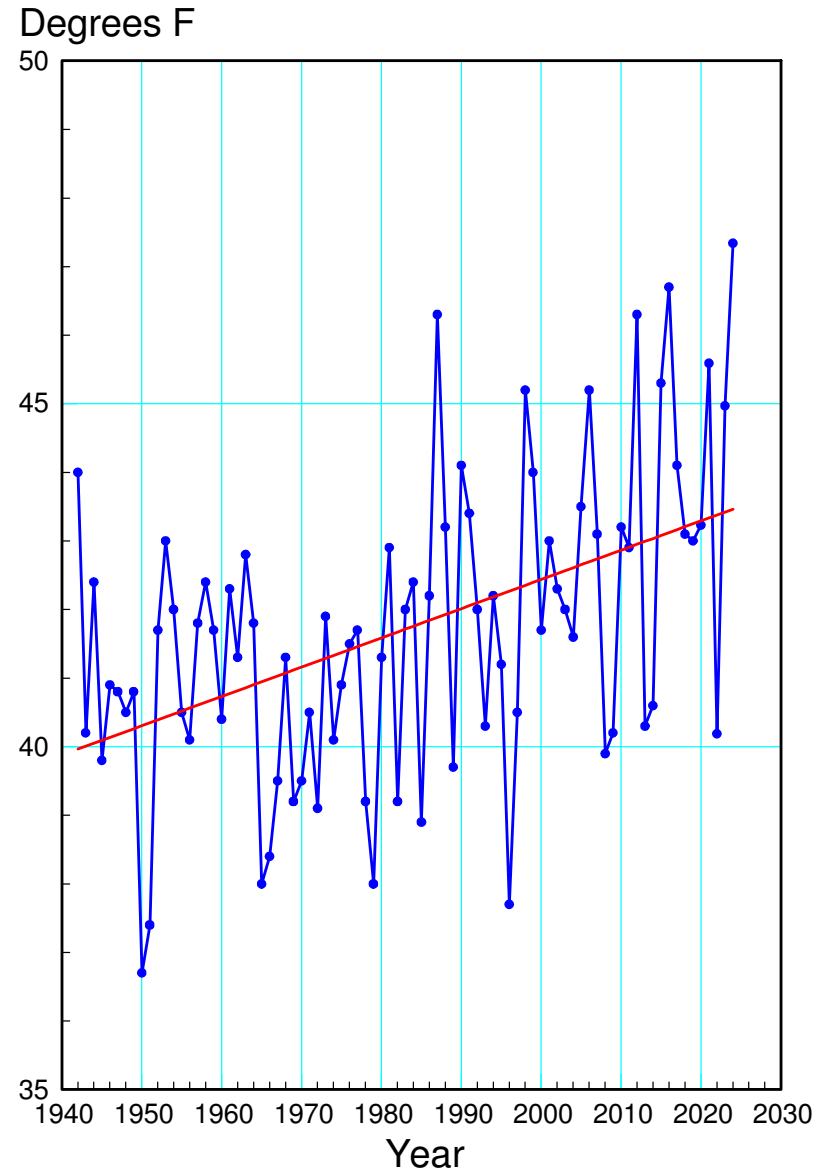


## In Matlab:

```
DATA = [  
    control V (paste the data)  
];  
y = DATA(:,1);  
F = DATA(:,14);  
plot(y,F,'.-')  
  
B = [y, y.^0];  
A = inv(B'*B)*B'*F  
  
    0.042649  
   -42.861  
  
plot(y,F,'.-',y,B*A,'r')
```

## Meaning

- Fargo is warming 0.0426F per year
- +3.419F over 80 years



# Atmospheric CO2 Levels

Source: NOAA Mauna Loa Observatory

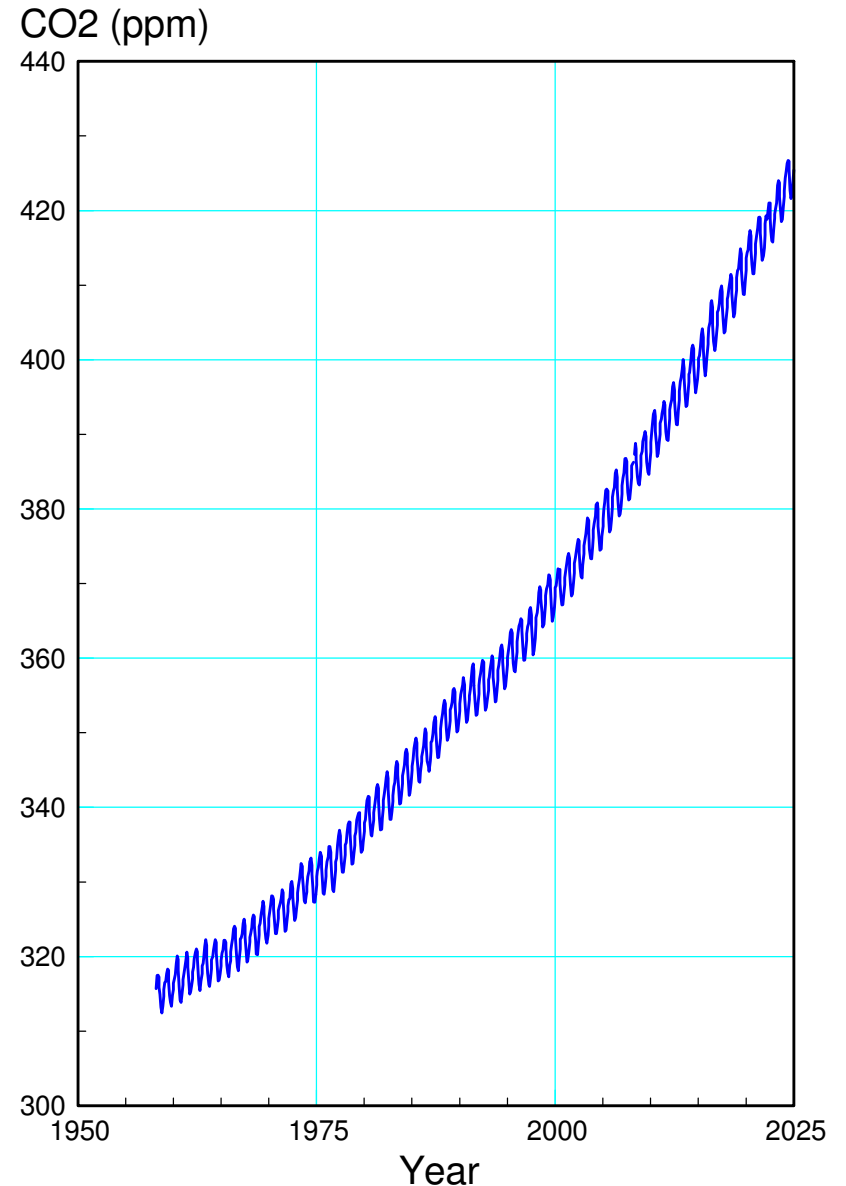
Measured since 1959

$$CO_2 = ay^2 + by + c$$

Determine a parabolic curve fit

Estimate when CO2 levels will reach 2000ppm

- Same as what triggered the Permian extinction
- 251 million years ago
- Nearly wiped out all life



# Least Squares Curve Fit

Use a parabolic curve fit:

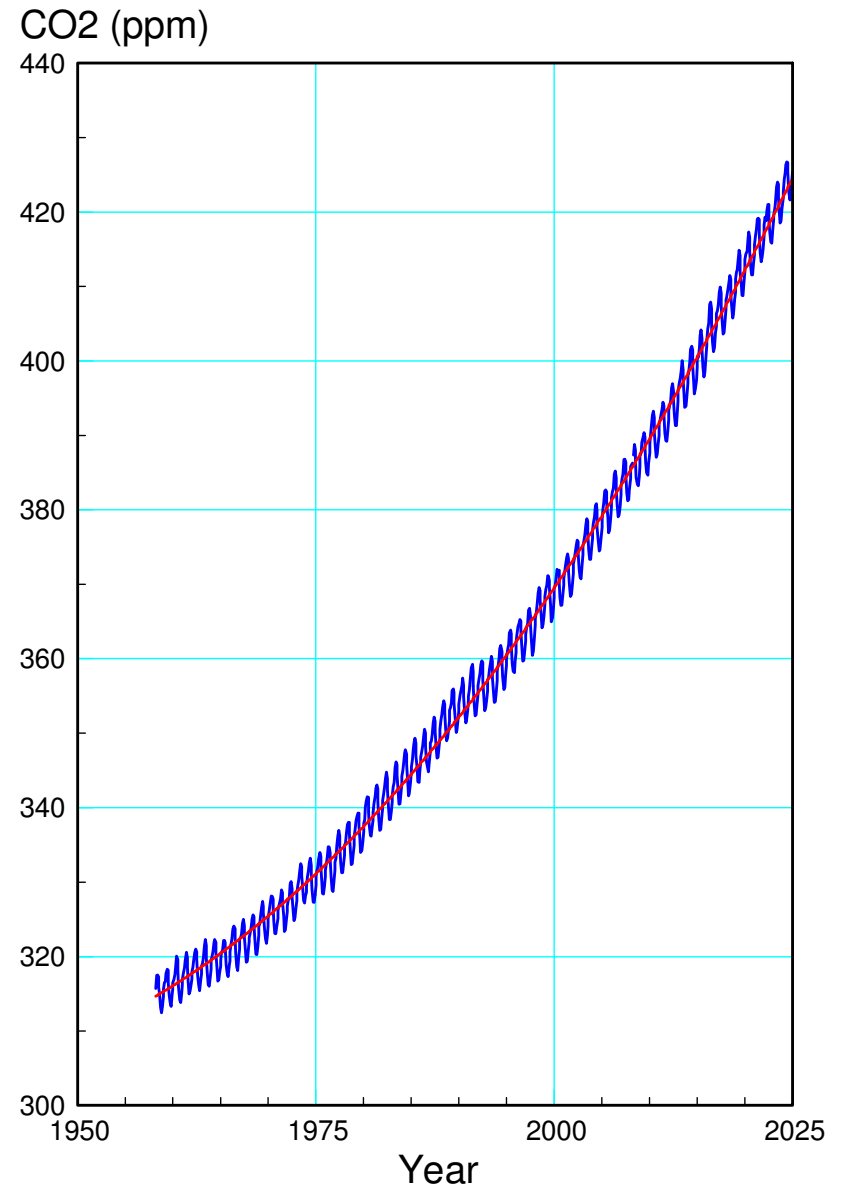
$$CO2 = ay^2 + by + c$$

```
DATA = [  
    paste in the data you just copied  
];
```

```
y = DATA(:,1);  
CO2 = DATA(:,2);  
B = [y.^2, y, y.^0];  
A = inv(B'*B)*B'*CO2
```

```
1.3300e-002  
-5.1333e+001  
4.9835e+004
```

```
plot(y,CO2,'b.-',y,B*A,'r')  
xlabel('Year');  
ylabel('CO2 ppm');
```



# Data Analysis

When will CO2 levels reach 2000 ppm?

$$ay^2 + by + c = 2000$$

Rewrite as

$$ay^2 + by + c - 2000 = 0$$

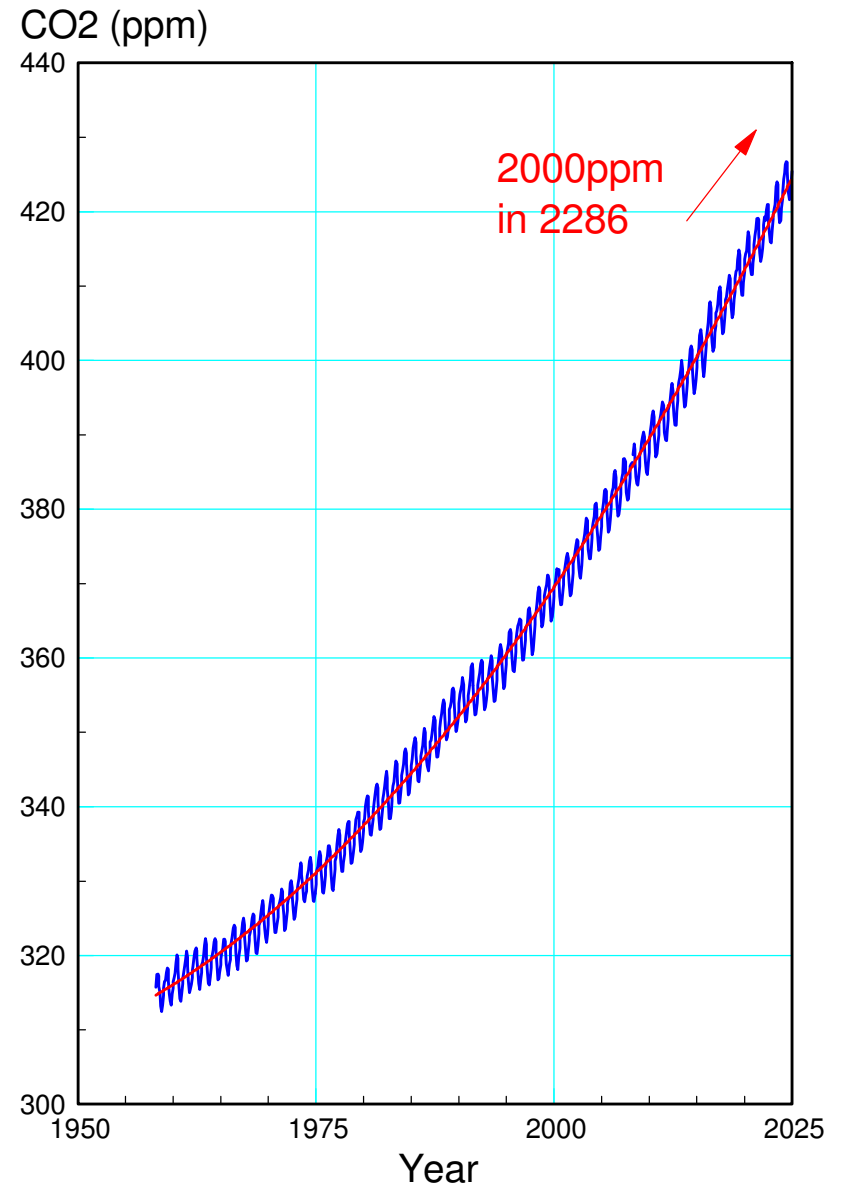
$$\text{roots}\left(\begin{bmatrix} a \\ b \\ c \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 2000 \end{bmatrix}\right)$$

`roots(A - [0;0;2000])`

**2286.9**

1564.3

If nothing changes, we should hit 2000ppm of CO2 in the year 2286.9



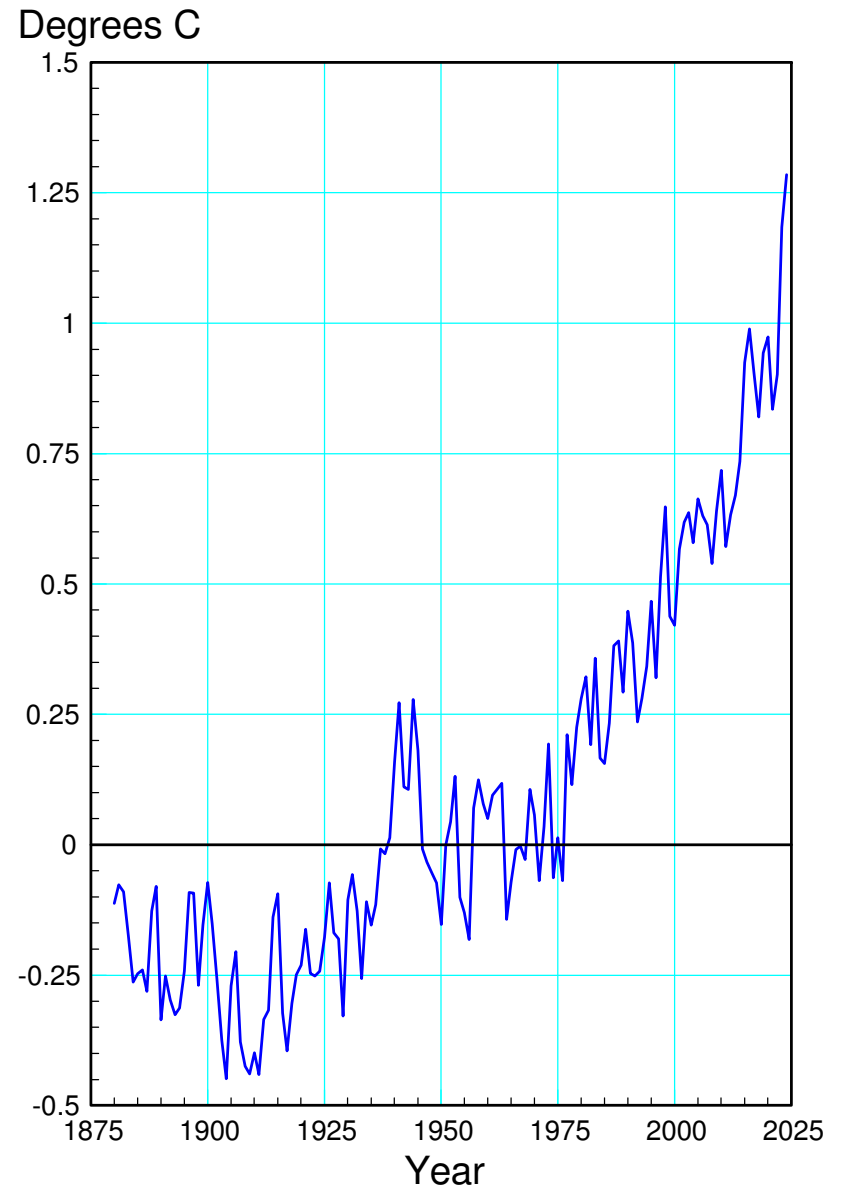
# Global Temperatures

- National Oceanic & Atmospheric Administration

NOAA has been monitoring global temperatures since 1880

Based upon the trend from 1975, what will

- What will be the temperature be in 2050?
- In 2100?





# Global Temperatures

Start with a linear curve fit

$$y = ax + b$$

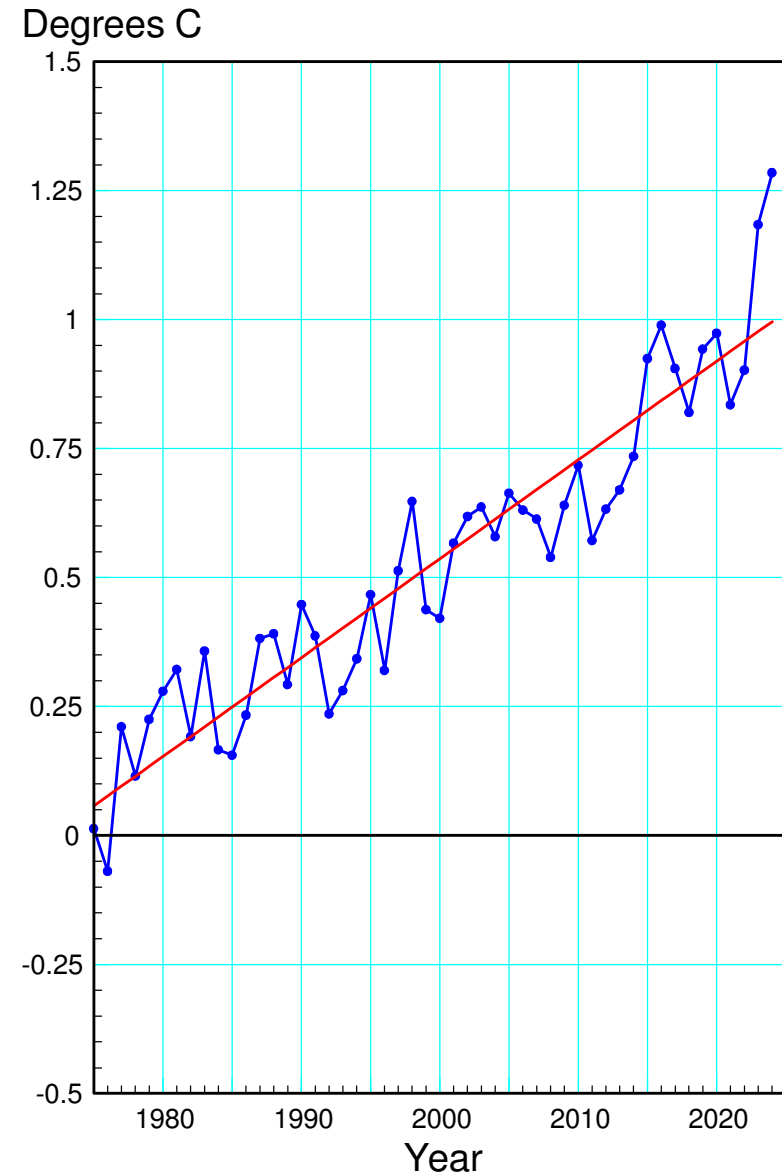
In Matlab

```
>> Data = [ <paste >];  
>> y = Data(:,1);  
>> dT = Data(:,2);  
>> B = [y, y.^0];  
>> A = inv(B'*B)*B'*dT  
    1.9159e-002  
   -3.7781e+001
```

$$dT \approx 0.019159y - 37.781$$

Temperature in years..

```
>> [2050, 1] * A  
ans = 1.4943e+000  
  
>> [2100, 1] * A  
ans = 2.4522e+000
```



# Global Temperatures

Try a parabolic curve fit

$$y = ax^2 + bx + c$$

In Matlab

```
>> B = [y.^2, y, y.^0];  
>> A = inv(B'*B)*B'*dT
```

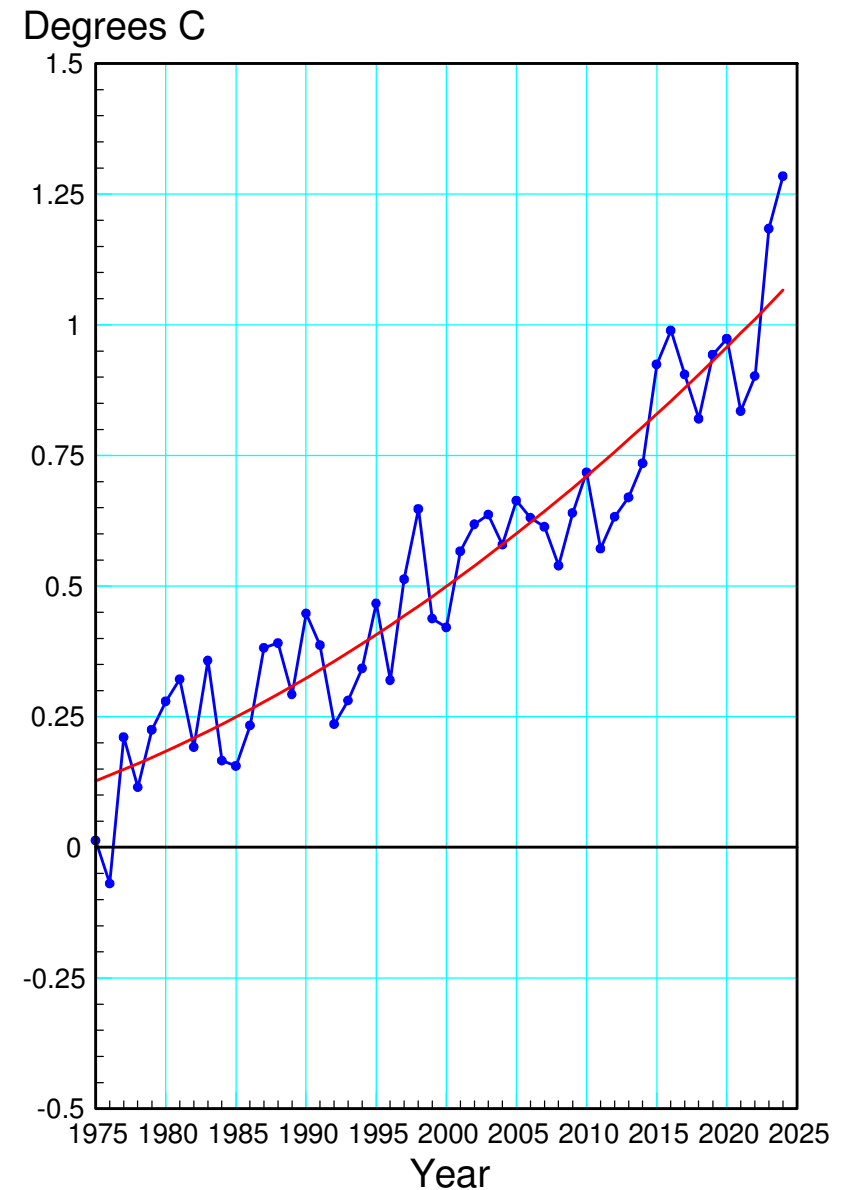
```
a 1.7844e-004  
b -6.9443e-001  
c 6.7559e+002
```

```
>> [2050^2, 2050, 1] * A
```

```
ans = 1.9122
```

```
>> [2100^2, 2100, 1] * A
```

```
ans = 4.2174
```



---

## Comment:

One of the great things about being an engineer is we're really good at analyzing data

Data is available on the internet

- Free of charge
- Multiple sources

You don't have to rely on experts to interpret the data

- You can analyze it yourself
- Matlab is a great tool for analyzing data

I

If you don't like what the data is telling you, you can

- Pretend the data doesn't exist, or
- Do something about it

Wind, Solar, Electric Vehicles are some of the ways of doing something about this

- And provide jobs for ECE majors
-

---

# 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.

- Useful in jet engines: measure something highly correlated with thrust
- Useful in Wall Street: measure something that is highly correlated with stock prices 1 year in the future.

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

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

Doing some algebra

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

The correlation coefficient is defined as

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

- $\rho = \pm 1$     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)

$$m_1 = \text{mean}(x)$$

2nd moment (m2)

$$m_2 = \text{mean}(x^2)$$

Variance

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

Covariance:

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

Correlation coefficient

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

---

---

Examples: Let

- $x_0$  be a variable in the range of  $(0,10)$
- $n$  be noise: random variable with a uniform distribution over  $(0,10)$

Let  $x$  be 0% to 100% noise

$$x = \alpha x_0 + (1 - \alpha)\eta$$

Let  $y$  be related to  $x$  as

$$y = 2x + 3$$

Determine how the correlation coefficient varies with  $\alpha$ .

# No Noise

- $\rho^2 = 1.000$

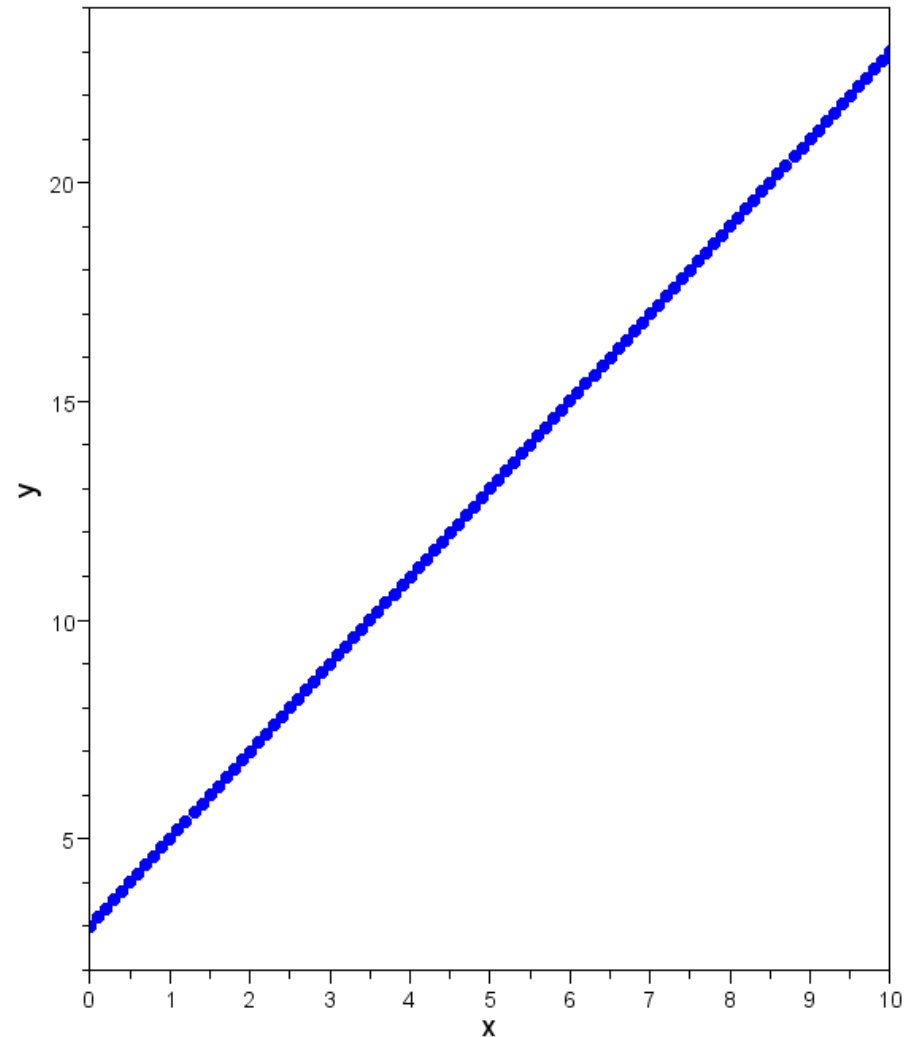
```
x = [0:0.1:10]';  
n = 10*rand(length(x),1);  
  
x0 = 1.0*x + 0.0*n;  
y = 2*x0 + 3;  
plot(x, y, 'b.');
```

```
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
```

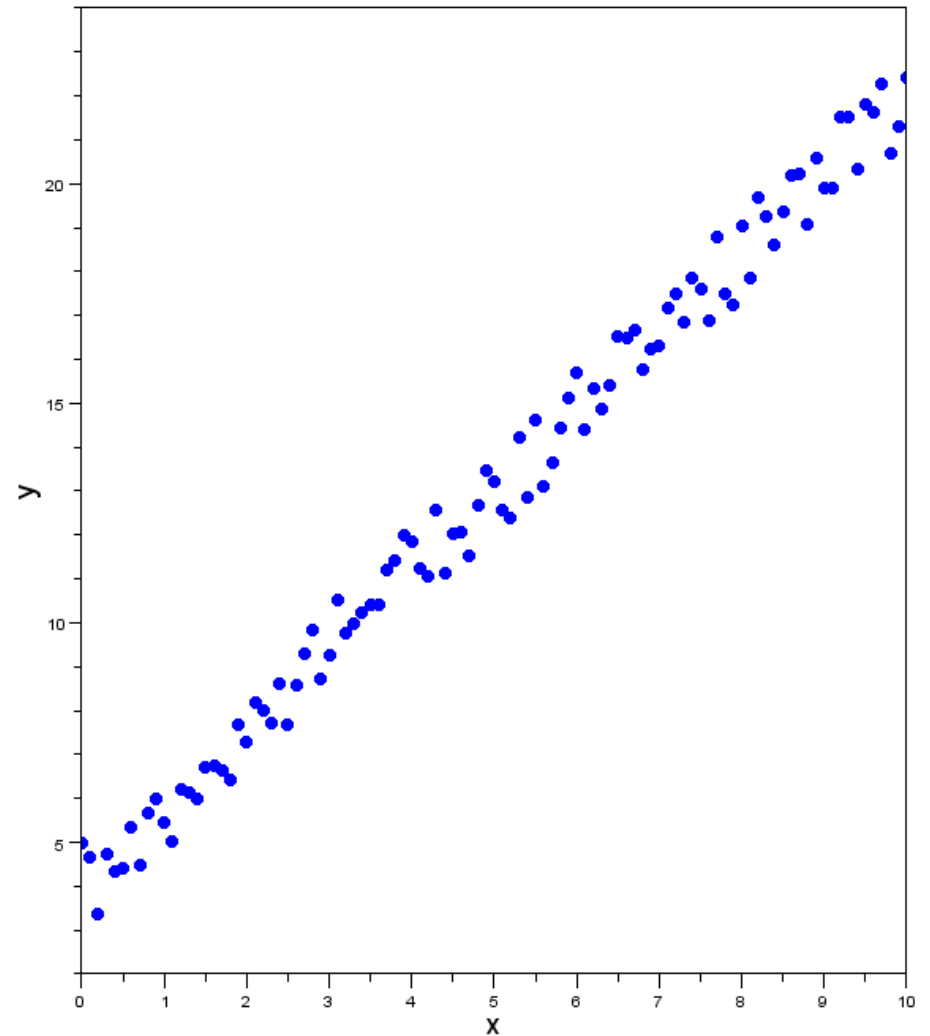




## 90% data, 10% noise

- $\rho^2 = 0.9943$

```
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
```



## 80% data / 20% noise

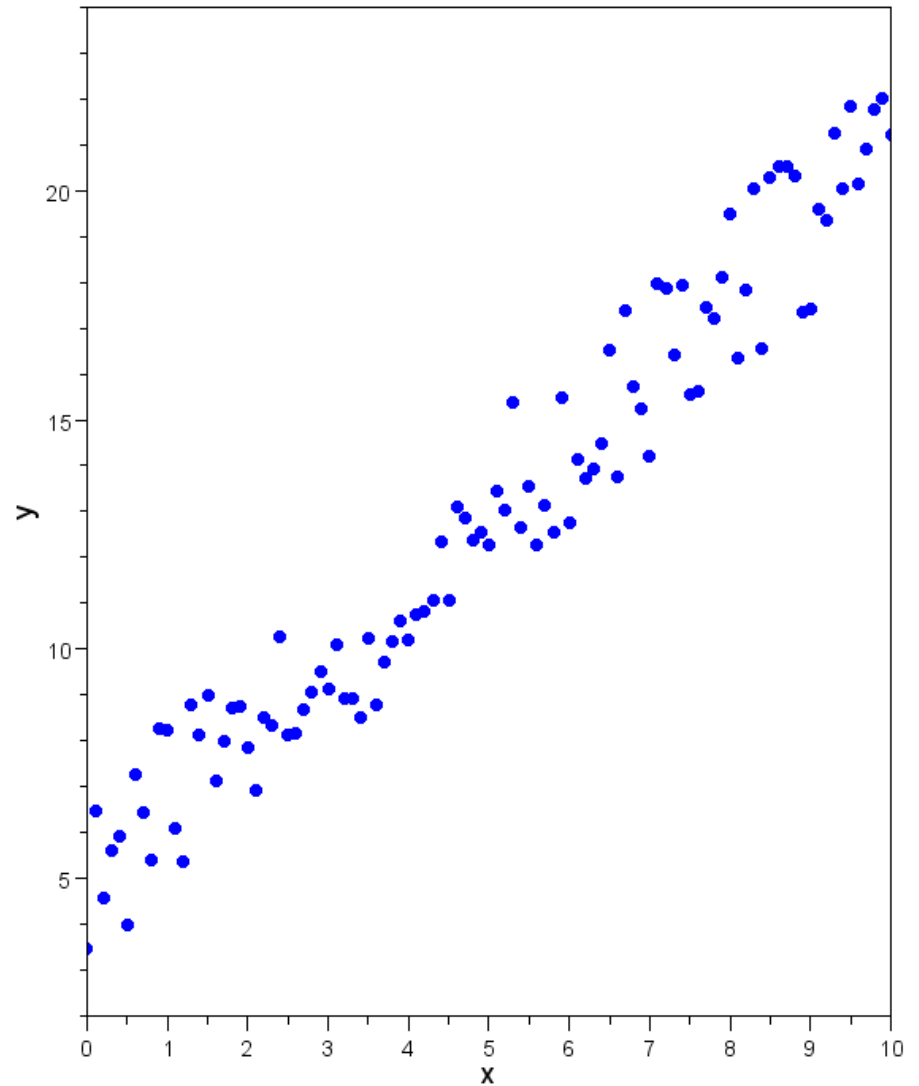
- $\rho^2 = 0.9700$

```
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
```



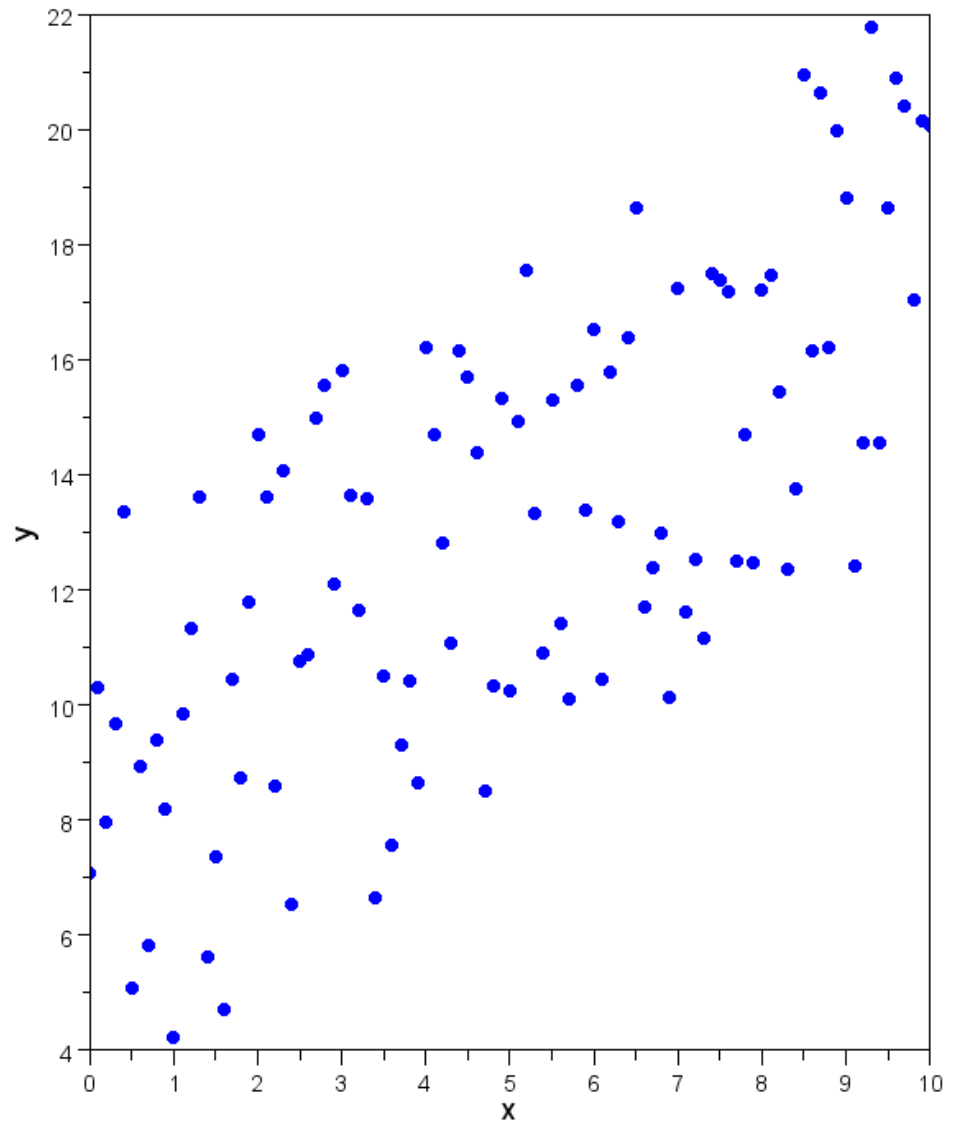
## 50% data / 50% noise

- $\rho^2 = 0.7074$

```
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



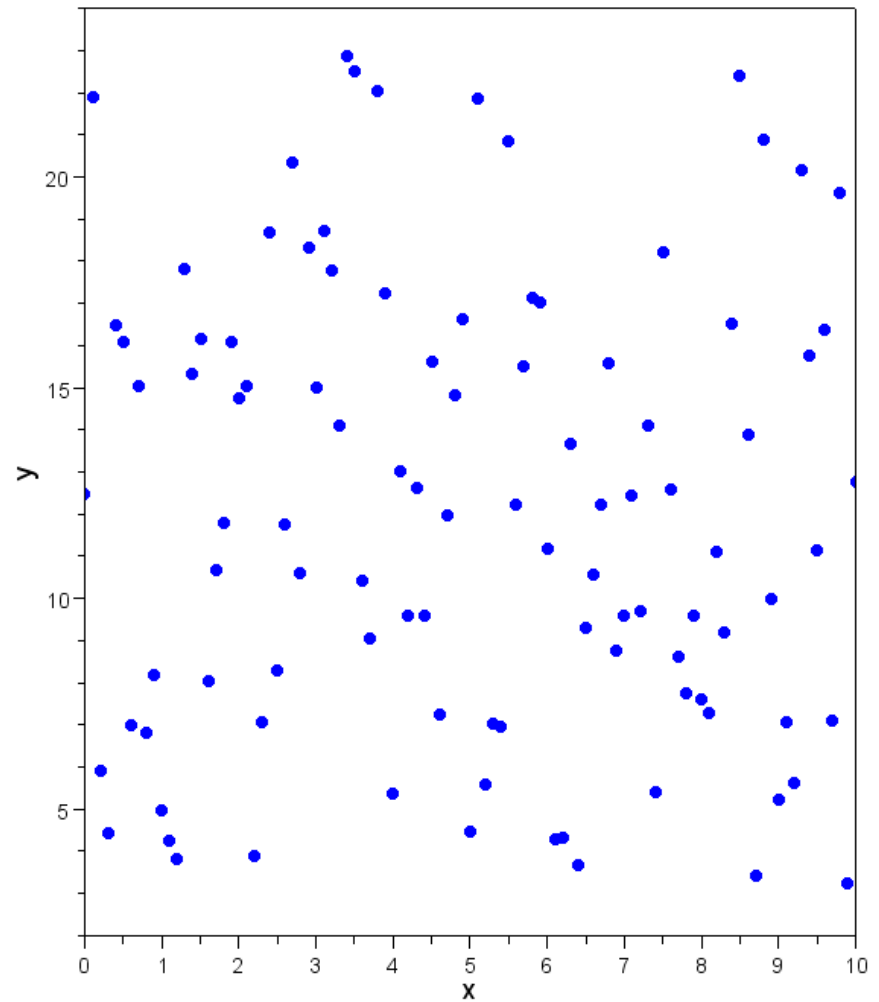
## 0% Data, 100% Noise

- $\rho^2 = 0.2429$

```
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

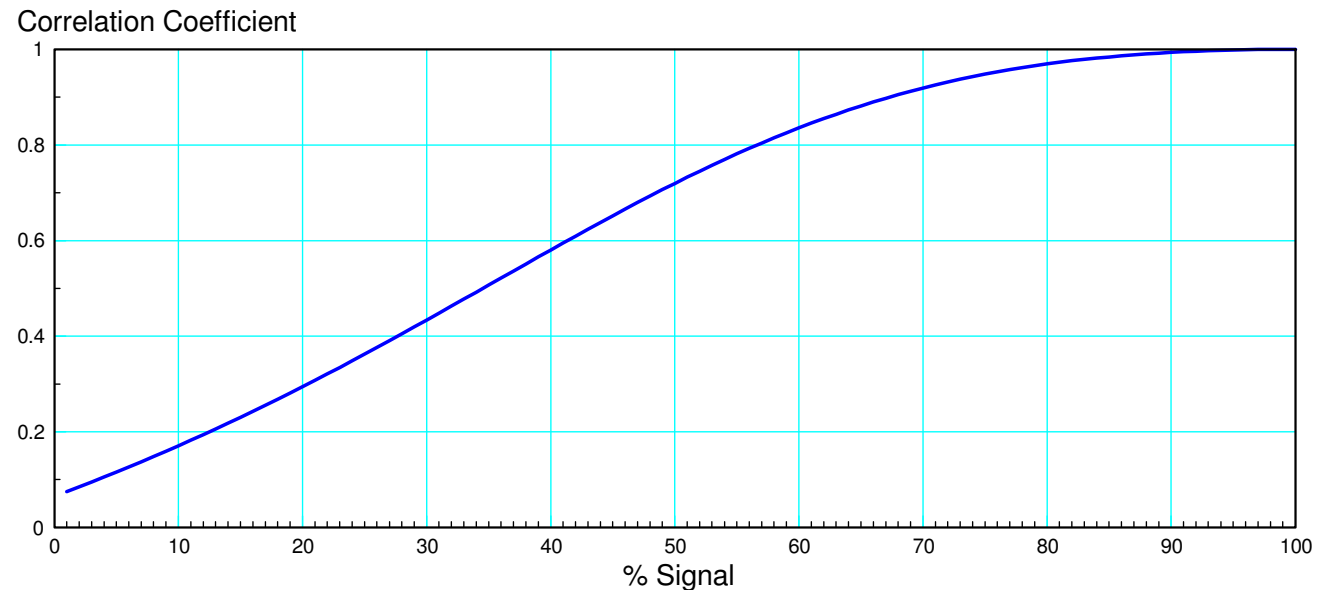
p2 = 0.2494



---

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
```

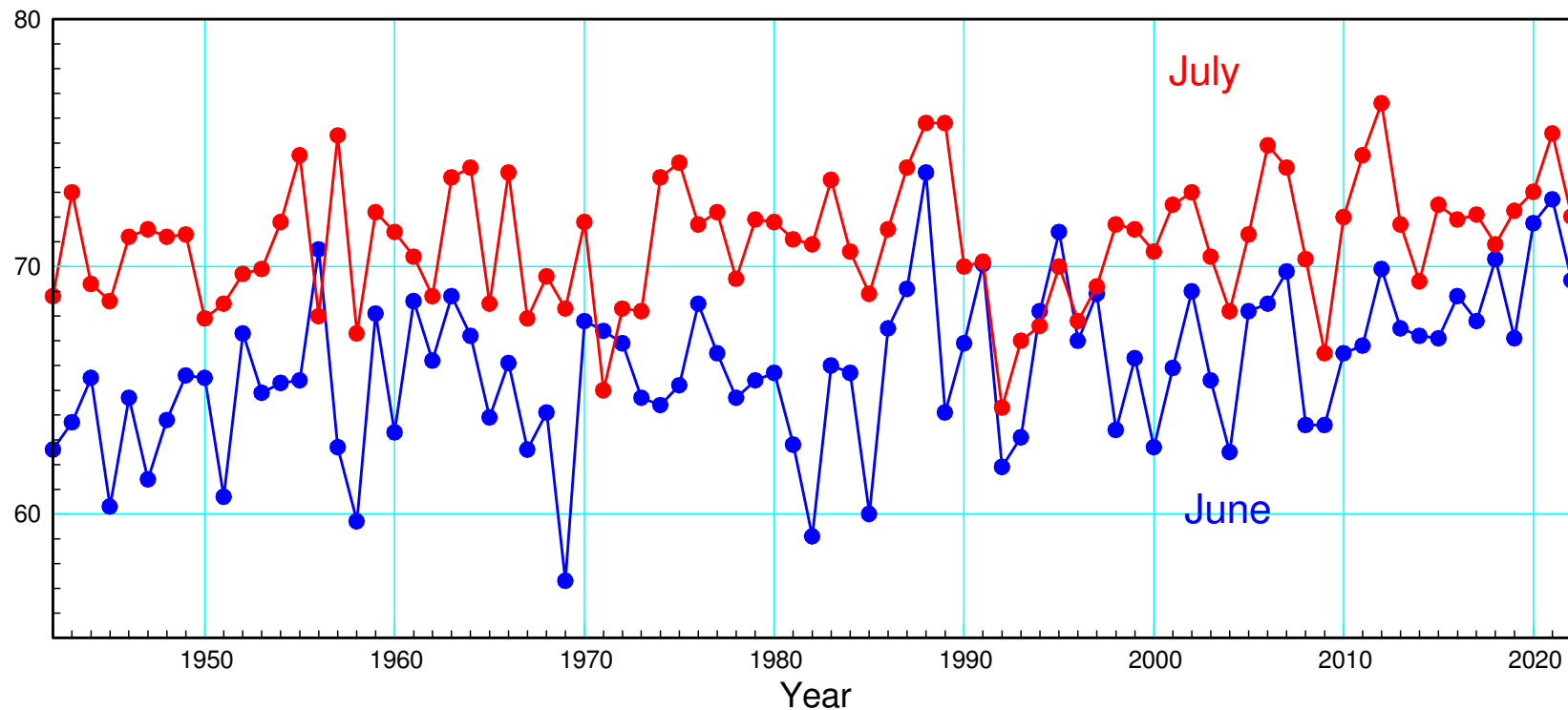


---

# Fun with Correlation Coefficients

If June is a hot month, what's the chance that July will also be hot?

- What's the correlation between the temperature in June and July?



---

The average temperature in June in Fargo, ND is available at

hector international airport

[http://www.bisonacademy.com/ECE111/Code/Fargo\\_Weather\\_Monthly\\_Avg.txt](http://www.bisonacademy.com/ECE111/Code/Fargo_Weather_Monthly_Avg.txt)

```
>> July = DATA(:,8);
```

```
>> June = DATA(:,7);
```

```
>> Cov = mean(June .* July) - mean(June) * mean(July)
```

```
Cov =      2.8057
```

```
>> correlation = Cov / ( std(June) * std(July) )
```

```
correlation =      0.3624
```

There is a 36% correlation between July and June

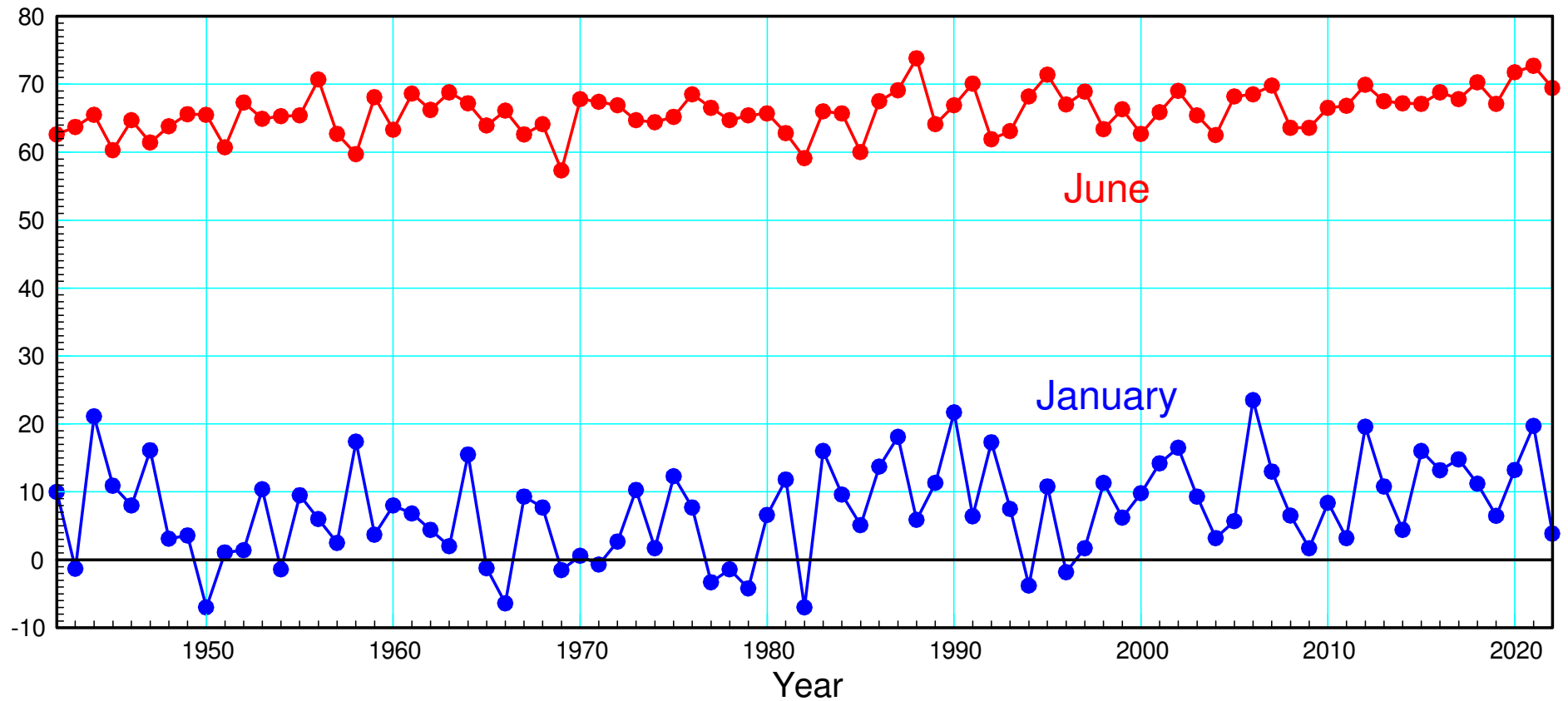
- If it's a hot June, there chance that July will also be hot is 36% higher than average

---

---

# January vs. June

If January is warm, will June be warm?





---

## Load data from Hector International Airport

[http://www.bisonacademy.com/ECE111/Code/Fargo\\_Weather\\_Monthly\\_Avg.txt](http://www.bisonacademy.com/ECE111/Code/Fargo_Weather_Monthly_Avg.txt)

## Find the correlation coefficient

```
>> June = DATA(:,7);  
>> January = DATA(:,2);  
>> Cov = mean(June .* January) - mean(June) * mean(January)
```

```
Cov =      3.5356
```

```
>> correlation = Cov / ( std(June) * std(January) )
```

```
correlation =      0.1640
```

There is a 16.4% correlation between January and June

- If January is cold, the chance that June will also be cold is 16% higher than average

---

## Summary

Regression analysis is basically curve fitting

Using least-squares, you can approximate data with polynomials

Correlation coefficients tell you how closely two variables change with each other

- Used in industry
- If you want to know something that is difficult to measure,
  - Find something that you can measure
  - That is highly correlated with what you want to measure