Week One In-class Exercise

Anscombe Data Set

The Anscombe data set is indroduced in the *NIST Engineering and Statistics Handbook* to demonstrate the value of Exporatory Data Analysis (EDA)

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 3.2.5
```

```
Anscombe <- read_excel("~/Google Drive/UU PMST/MST 6600 - Advanced Statistical Techniques/NIST Engineer
```

```
## Warning in strptime(x, format, tz = tz): unknown timezone 'zone/tz/2017c.
## 1.0/zoneinfo/America/Denver'
```

Although this looks like a lot of code, it was generated using the **Import Dataset** dropdown box.

We can view the data in the concole, "viewer," or in the table below:

```
Anscombe
```

```
## # A tibble: 11 x 8
##       X1    Y1    X2    Y2    X3    Y3    X4    Y4
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1    10  8.04    10  9.14    10  7.46     8  6.58
##  2     8  6.95     8  8.14     8  6.77     8  5.76
##  3    13  7.58    13  8.74    13 12.74     8  7.71
##  4     9  8.81     9  8.77     9  7.11     8  8.84
##  5    11  8.33    11  9.26    11  7.81     8  8.47
##  6    14  9.96    14  8.10    14  8.84     8  7.04
##  7     6  7.24     6  6.13     6  6.08     8  5.25
##  8     4  4.26     4  3.10     4  5.39    19 12.50
##  9    12 10.84    12  9.13    12  8.15     8  5.56
## 10     7  4.82     7  7.26     7  6.42     8  7.91
## 11     5  5.68     5  4.74     5  5.73     8  6.89
```

What's intersting about this data set?

Given we have a set of X and Y pairs, we may assume that data should be grouped together; a normal, *non-graphical* analysis would be to preform a linear regression on the data.

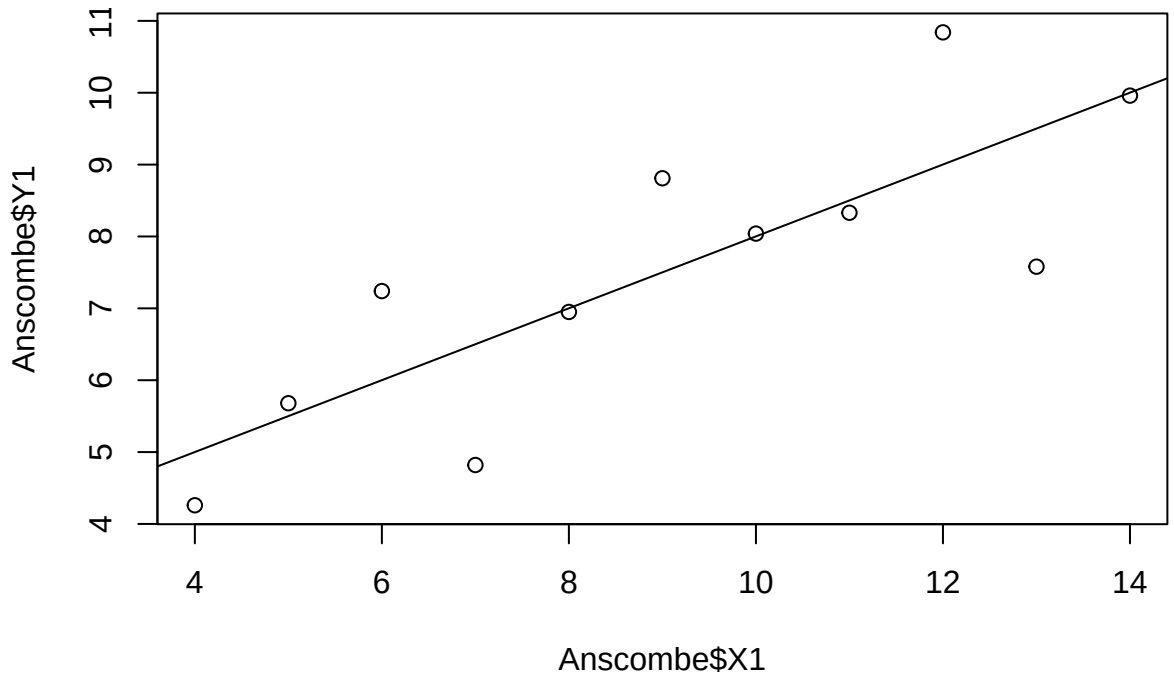
```
xy1.linear.model <- lm(Anscombe$Y1 ~ Anscombe$X1)
summary(xy1.linear.model)
```

```
##
## Call:
## lm(formula = Anscombe$Y1 ~ Anscombe$X1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.92127 -0.45577 -0.04136  0.70941  1.83882
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0001     1.1247   2.667  0.02573 *
## Anscombe$X1   0.5001     0.1179   4.241  0.00217 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6665, Adjusted R-squared:  0.6295
## F-statistic: 17.99 on 1 and 9 DF,  p-value: 0.00217
```

I might even plot the data (adding the linear regression line from above):

```
plot(Anscombe$X1, Anscombe$Y1)
abline(xy1.linear.model)
```



Create summaries of the other XY pairs

```
xy2.linear.model <- lm(Anscombe$Y2 ~ Anscombe$X2)
xy3.linear.model <- lm(Anscombe$Y3 ~ Anscombe$X3)
xy4.linear.model <- lm(Anscombe$Y4 ~ Anscombe$X4)
```

```
summary(xy2.linear.model)
```

```
##
## Call:
## lm(formula = Anscombe$Y2 ~ Anscombe$X2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9009 -0.7609  0.1291  0.9491  1.2691
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.001      1.125   2.667  0.02576 *
## Anscombe$X2    0.500      0.118   4.239  0.00218 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6662, Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002179
```

```
summary(xy3.linear.model)
```

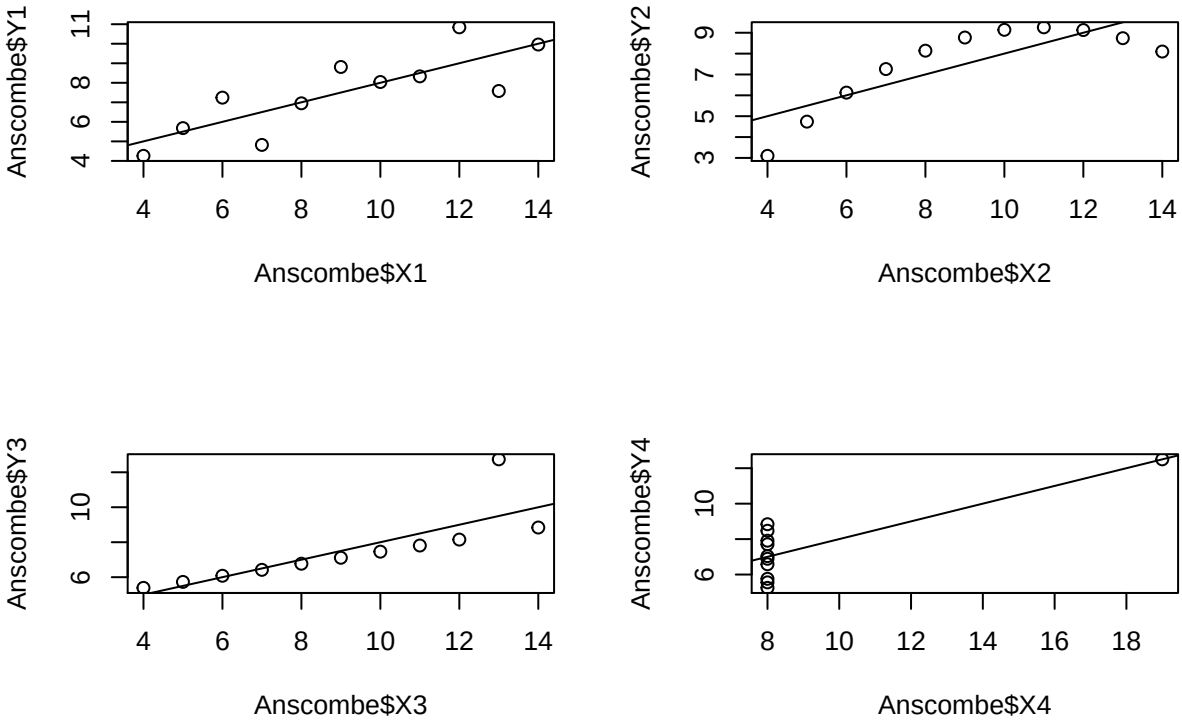
```
##
## Call:
## lm(formula = Anscombe$Y3 ~ Anscombe$X3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1586 -0.6146 -0.2303  0.1540  3.2411
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0025     1.1245   2.670  0.02562 *
## Anscombe$X3   0.4997     0.1179   4.239  0.00218 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6663, Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002176
```

```
summary(xy4.linear.model)
```

```
##
## Call:
## lm(formula = Anscombe$Y4 ~ Anscombe$X4)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.751 -0.831  0.000  0.809  1.839
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0017     1.1239   2.671  0.02559 *
## Anscombe$X4   0.4999     0.1178   4.243  0.00216 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6667, Adjusted R-squared:  0.6297
## F-statistic:    18 on 1 and 9 DF,  p-value: 0.002165
```

Let's look at all the data graphically

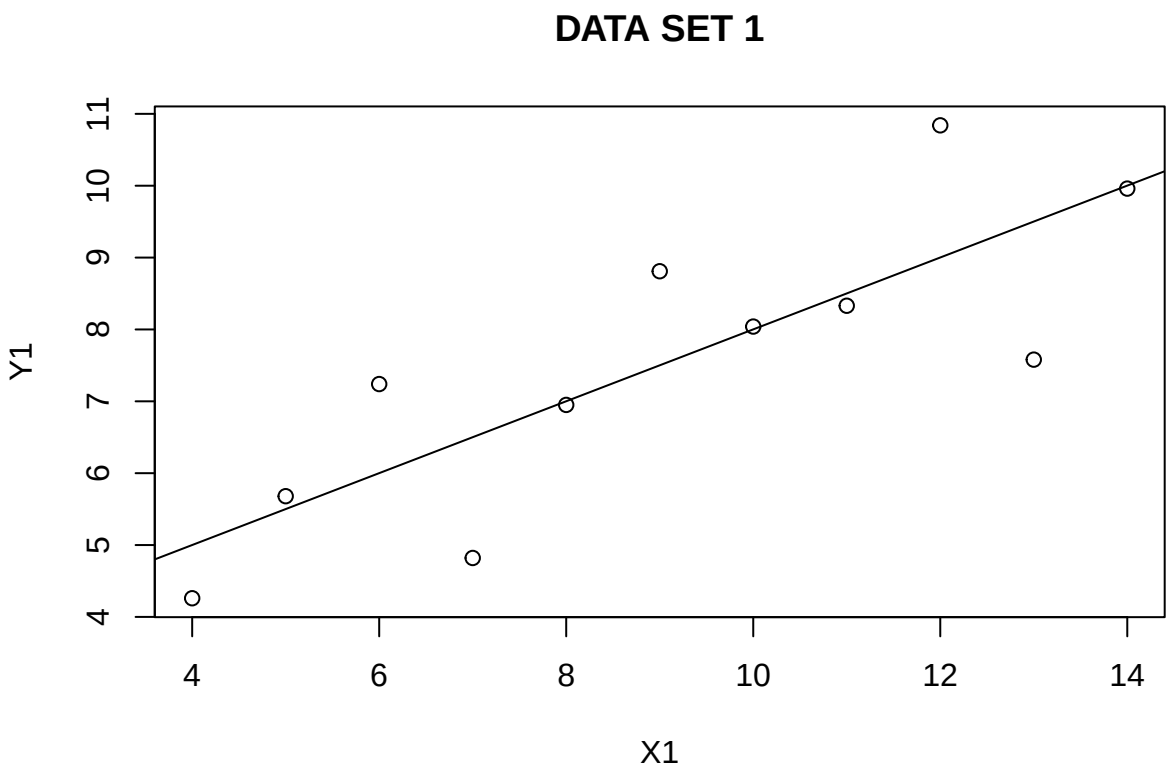
```
# create a single page of all values
par(mfrow=c(2,2)) # Change the panel layout to 2 x 2
plot(Anscombe$X1, Anscombe$Y1)
abline(xy1.linear.model)

plot(Anscombe$X2, Anscombe$Y2)
abline(xy2.linear.model)

plot(Anscombe$X3, Anscombe$Y3)
abline(xy3.linear.model)

plot(Anscombe$X4, Anscombe$Y4)
abline(xy4.linear.model)
```




```
par(mfrow=c(1,1)) # Change back to 1 x 1
```

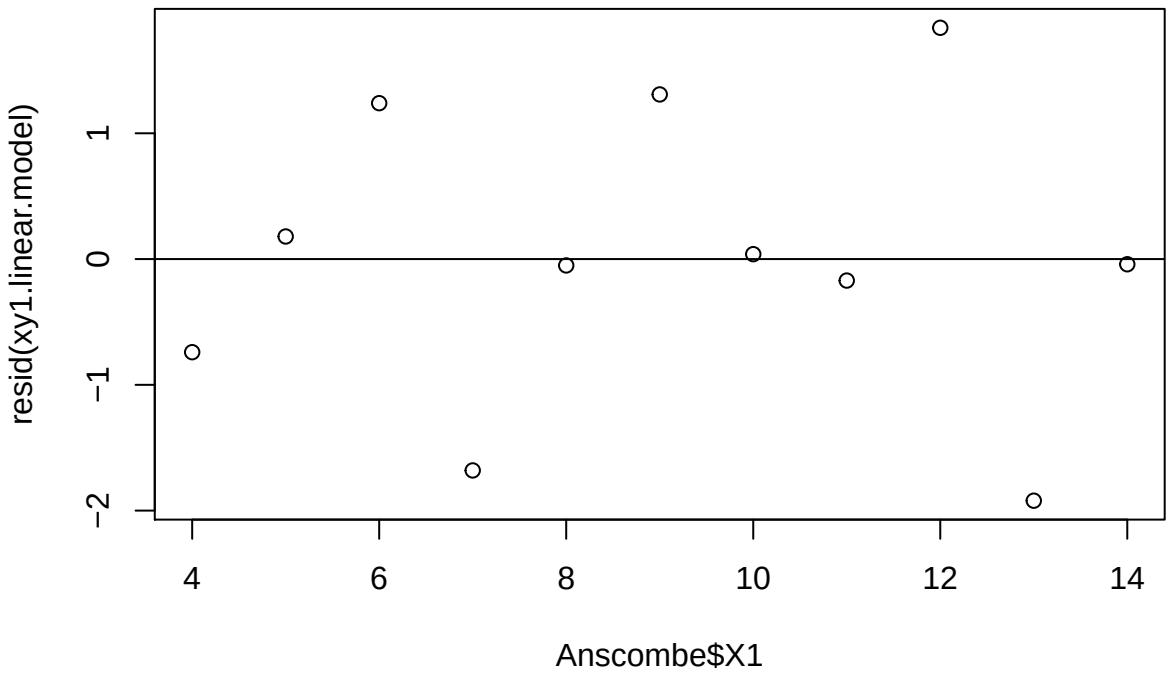
Let's make one of the graphs "pretty"

```
# Making changes to graphs
# create a simple scatter plot of X1 vs Y1 with some lables
plot(Anscombe$X1, Anscombe$Y1, main="DATA SET 1", xlab="X1", ylab="Y1")
abline(xy1.linear.model)
```

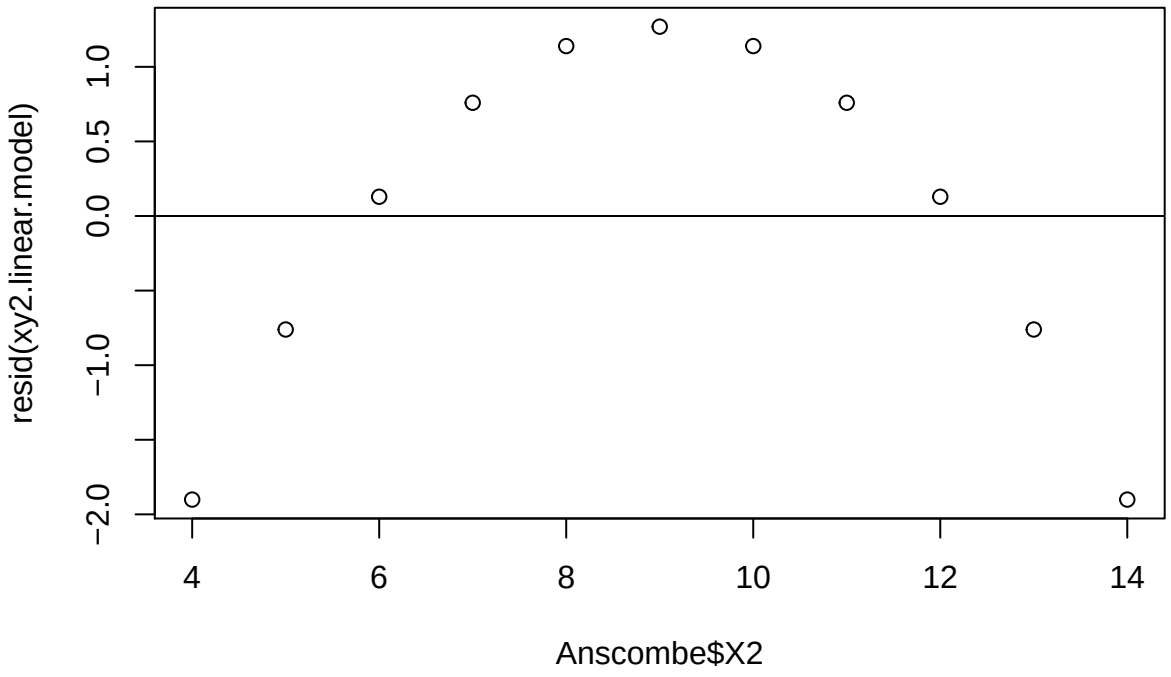


What else do I get with R?

```
plot(Anscombe$X1, resid(xy1.linear.model))
abline(h = 0)
```




```
plot(Anscombe$X2, resid(xy2.linear.model))
abline(h = 0)
```



***

A Look Ahead

The code above uses *base R* commands. While they can be used to create clean, accurate graphs, it requires the user to **know** many different commands and structures.

Over the last 10-years, several new R packages have been created to make analysis and graphics easier. This course will primarily exploit two of those packages: *tidyverse* and *ggplot2*.

Example 2. The Anscombe dataset with ggplot2

I need to tidy up the data:

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.2.5
```

```
## Warning: replacing previous import by 'tidyr::%>%' when loading 'broom'
```

```
## Warning: replacing previous import by 'tidyr::gather' when loading 'broom'
```

```
## Warning: replacing previous import by 'tidyr::spread' when loading 'broom'
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr
```

```
## Warning: package 'ggplot2' was built under R version 3.2.5
```

```
## Warning: package 'tibble' was built under R version 3.2.5
```

```
## Warning: package 'tidyr' was built under R version 3.2.5
```

```
## Warning: package 'readr' was built under R version 3.2.5
## Warning: package 'purrr' was built under R version 3.2.5
## Warning: package 'dplyr' was built under R version 3.2.5
## Conflicts with tidy packages ----------------------------------------------
## filter(): dplyr, stats
## lag():    dplyr, stats
```

```
Anscombe.tidy.xy1 <- Anscombe %>% select("X1", "Y1") %>%
  mutate(dataset = "DataSet1") %>%
  rename(X = X1, Y = Y1)

Anscombe.tidy.xy2 <- Anscombe %>% select("X2", "Y2") %>%
  mutate(dataset = "DataSet2") %>%
  rename(X = X2, Y = Y2)

Anscombe.tidy.xy3 <- Anscombe %>% select("X3", "Y3") %>%
  mutate(dataset = "DataSet3") %>%
  rename(X = X3, Y = Y3)

Anscombe.tidy.xy4 <- Anscombe %>% select("X4", "Y4") %>%
  mutate(dataset = "DataSet4") %>%
  rename(X = X4, Y = Y4)
```

We can look at the output of the first data set:

```
Anscombe.tidy.xy1
```

```
## # A tibble: 11 x 3
##        X     Y  dataset
##    <dbl> <dbl>    <chr>
##  1    10  8.04 DataSet1
##  2     8  6.95 DataSet1
##  3    13  7.58 DataSet1
##  4     9  8.81 DataSet1
##  5    11  8.33 DataSet1
##  6    14  9.96 DataSet1
##  7     6  7.24 DataSet1
##  8     4  4.26 DataSet1
##  9    12 10.84 DataSet1
## 10     7  4.82 DataSet1
## 11     5  5.68 DataSet1
```

In order to fully complete the analysis, we need to put all for *datasets* together:

```
Anscombe.tidy.data <-
  bind_rows(Anscombe.tidy.xy1, Anscombe.tidy.xy2, Anscombe.tidy.xy3, Anscombe.tidy.xy4)
```

The new dataset is "tidy."

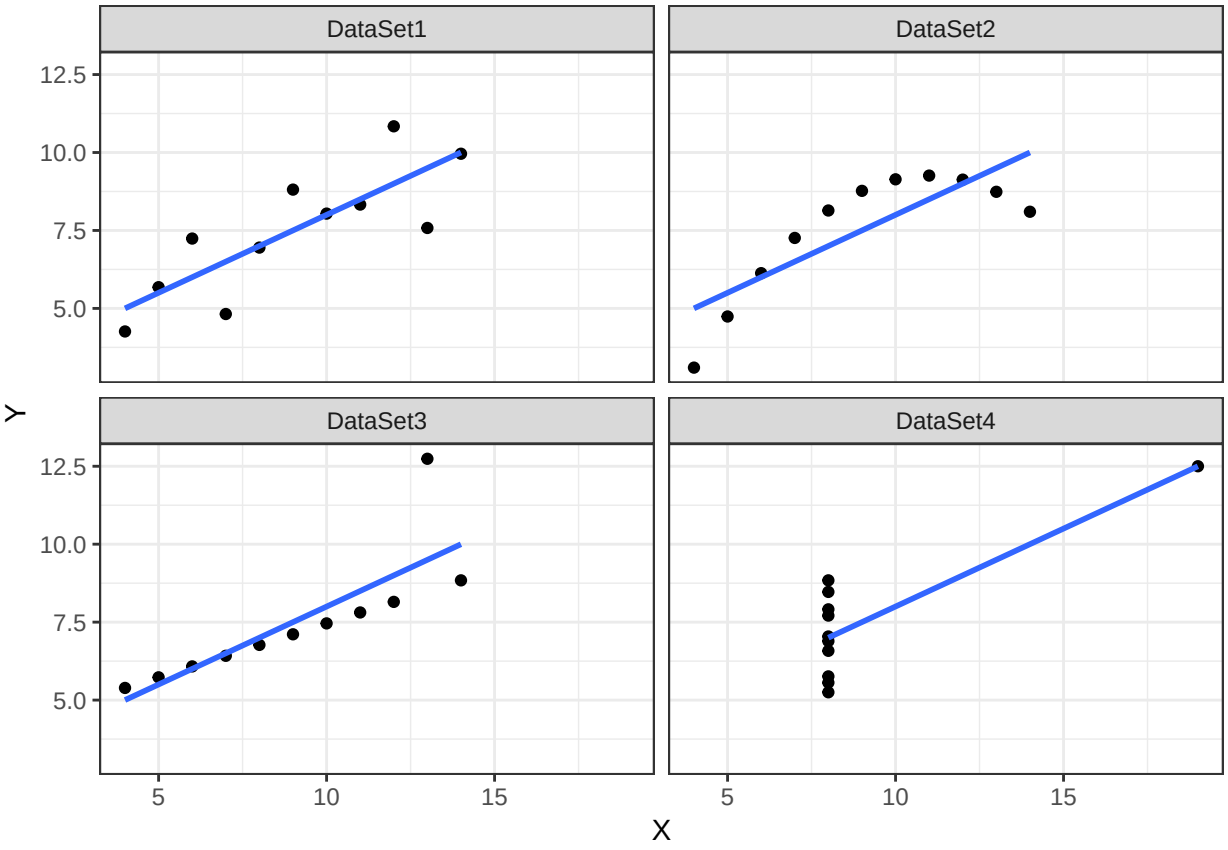
```
Anscombe.tidy.data
```

```
## # A tibble: 44 x 3
##        X     Y  dataset
##    <dbl> <dbl>    <chr>
##  1    10  8.04 DataSet1
##  2     8  6.95 DataSet1
```

```
##  3    13  7.58 DataSet1
##  4     9  8.81 DataSet1
##  5    11  8.33 DataSet1
##  6    14  9.96 DataSet1
##  7     6  7.24 DataSet1
##  8     4  4.26 DataSet1
##  9    12 10.84 DataSet1
## 10     7  4.82 DataSet1
## # ... with 34 more rows
```

The POWER of ggplot2

```
Anscombe.tidy.data %>%
  ggplot(aes(X, Y)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  facet_wrap(~ dataset) +
  theme_bw()
```



Modeling is easier as well...

```
# modeling the data for DataSet1 through DataSet4
xy1.lm <- Anscombe.tidy.data %>%
  filter(dataset == "DataSet1") %>% lm(Y ~ X, .)
```

```
## Warning: package 'bindrcpp' was built under R version 3.2.5
```

```
#plot(xy1.lm)
summary(xy1.lm)
```

```
##
## Call:
## lm(formula = Y ~ X, data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.92127 -0.45577 -0.04136  0.70941  1.83882
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0001     1.1247   2.667  0.02573 *
## X             0.5001     0.1179   4.241  0.00217 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6665, Adjusted R-squared:  0.6295
## F-statistic: 17.99 on 1 and 9 DF,  p-value: 0.00217
```

```
xy2.lm <- Anscombe.tidy.data %>%
  filter(dataset == "DataSet2") %>% lm(Y ~ X, .)
#plot(xy2.lm)
summary(xy2.lm)
```

```
##
## Call:
## lm(formula = Y ~ X, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9009 -0.7609  0.1291  0.9491  1.2691
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.001      1.125   2.667  0.02576 *
## X              0.500      0.118   4.239  0.00218 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.237 on 9 degrees of freedom
## Multiple R-squared:  0.6662, Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002179
```

```
xy3.lm <- Anscombe.tidy.data %>%
  filter(dataset == "DataSet3") %>% lm(Y ~ X, .)
#plot(xy3.lm)
summary(xy3.lm)
```

```
##
## Call:
## lm(formula = Y ~ X, data = .)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1586 -0.6146 -0.2303  0.1540  3.2411
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0025     1.1245   2.670  0.02562 *
## X             0.4997     0.1179   4.239  0.00218 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6663, Adjusted R-squared:  0.6292
## F-statistic: 17.97 on 1 and 9 DF,  p-value: 0.002176
```

```
xy4.lm <- Anscombe.tidy.data %>%
  filter(dataset == "DataSet4") %>% lm(Y ~ X, .)
#plot(xy4.lm)
summary(xy4.lm)
```

```
##
## Call:
## lm(formula = Y ~ X, data = .)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.751 -0.831  0.000  0.809  1.839
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0017     1.1239   2.671  0.02559 *
## X             0.4999     0.1178   4.243  0.00216 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.236 on 9 degrees of freedom
## Multiple R-squared:  0.6667, Adjusted R-squared:  0.6297
## F-statistic:    18 on 1 and 9 DF,  p-value: 0.002165
```