Assignment 3

1. Chose the persp() part

```r
# Persp
x <- seq(-10, 10, length= 30) # create the x sequence
y <- x # let y = x
f <- function(x,y) { r <- sqrt(x^2+y^2); 10 * sin(r)/r } # define the function f(x,y)
z <- outer(x, y, f) # compute z using outer product
z[is.na(z)] <- 1 # replace NA into 1
# 0.5 to include z axis label
par(mar=c(0, 0.5, 0, 0), lwd=0.5) # Adjust margins and line width
persp(x, y, z, theta = 30, phi = 30, # using persp() with viewing angles theta and phi to create 3d surface
      expand = 0.5) #compress the height of z by half
# theta = rotate angle around z-axis
# phi = vertical tilt
par(mar=c(5.1, 4.1, 4.1, 2.1), lwd=1)  # Reset default margins and line width for next plot
```

2. a.

   According to the summary statistics and regression results, the four groups of data look very similar at first glance. Therefore, we might expect them to have a similar distribution.

```
> summary(anscombe)
       x1             x2             x3             x4           y1              y2              y3
 Min.   : 4.0   Min.   : 4.0   Min.   : 4.0   Min.   : 8   Min.   : 4.260   Min.   :3.100   Min.   : 5.39
 1st Qu.: 6.5   1st Qu.: 6.5   1st Qu.: 6.5   1st Qu.: 8   1st Qu.: 6.315   1st Qu.:6.695   1st Qu.: 6.25
 Median : 9.0   Median : 9.0   Median : 9.0   Median : 8   Median : 7.580   Median :8.140   Median : 7.11
 Mean   : 9.0   Mean   : 9.0   Mean   : 9.0   Mean   : 9   Mean   : 7.501   Mean   :7.501   Mean   : 7.50
 3rd Qu.:11.5   3rd Qu.:11.5   3rd Qu.:11.5   3rd Qu.: 8   3rd Qu.: 8.570   3rd Qu.:8.950   3rd Qu.: 7.98
 Max.   :14.0   Max.   :14.0   Max.   :14.0   Max.   :19   Max.   :10.840   Max.   :9.260   Max.   :12.74
       y4
 Min.   : 5.250
 1st Qu.: 6.170
 Median : 7.040
 Mean   : 7.501
 3rd Qu.: 8.190
 Max.   :12.500
```

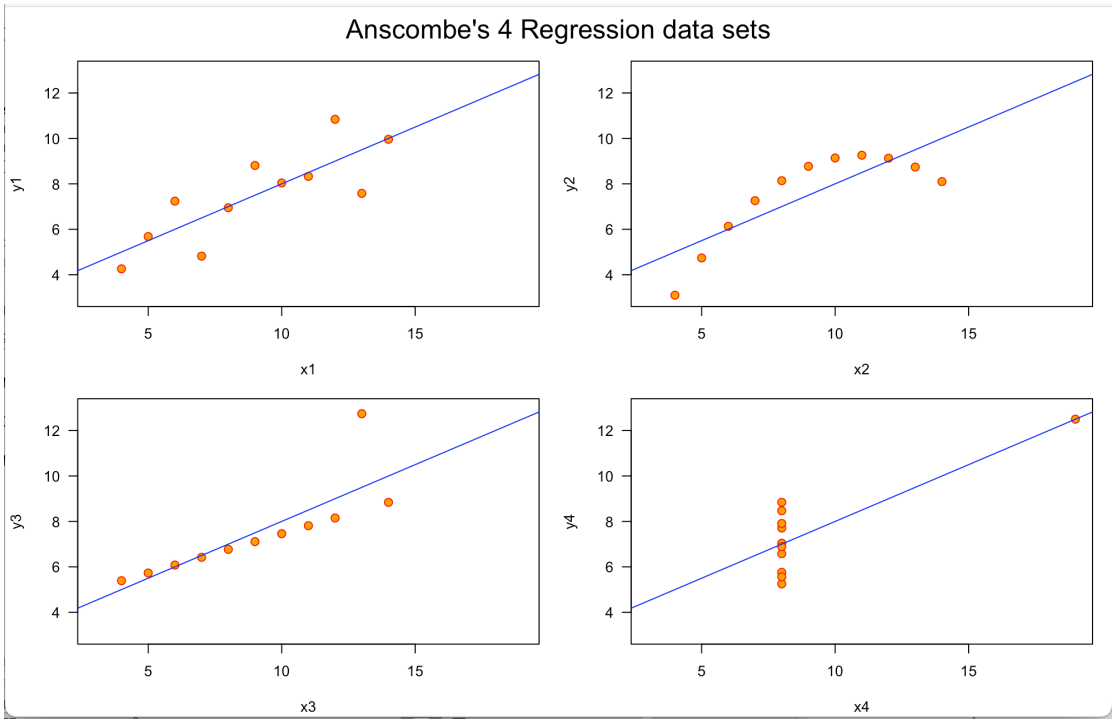
```
> lm1 <- lm(y1 ~ x1, data=anscombe)
> summary(lm1)

Call:
lm(formula = y1 ~ x1, data = anscombe)

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 *
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
```

```
> lm2 <- lm(y2 ~ x2, data=anscombe)
> summary(lm2)

Call:
lm(formula = y2 ~ x2, data = anscombe)

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 *
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
```

```
> lm3 <- lm(y3 ~ x3, data=anscombe)
> summary(lm3)

Call:
lm(formula = y3 ~ x3, data = anscombe)

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 *
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
```

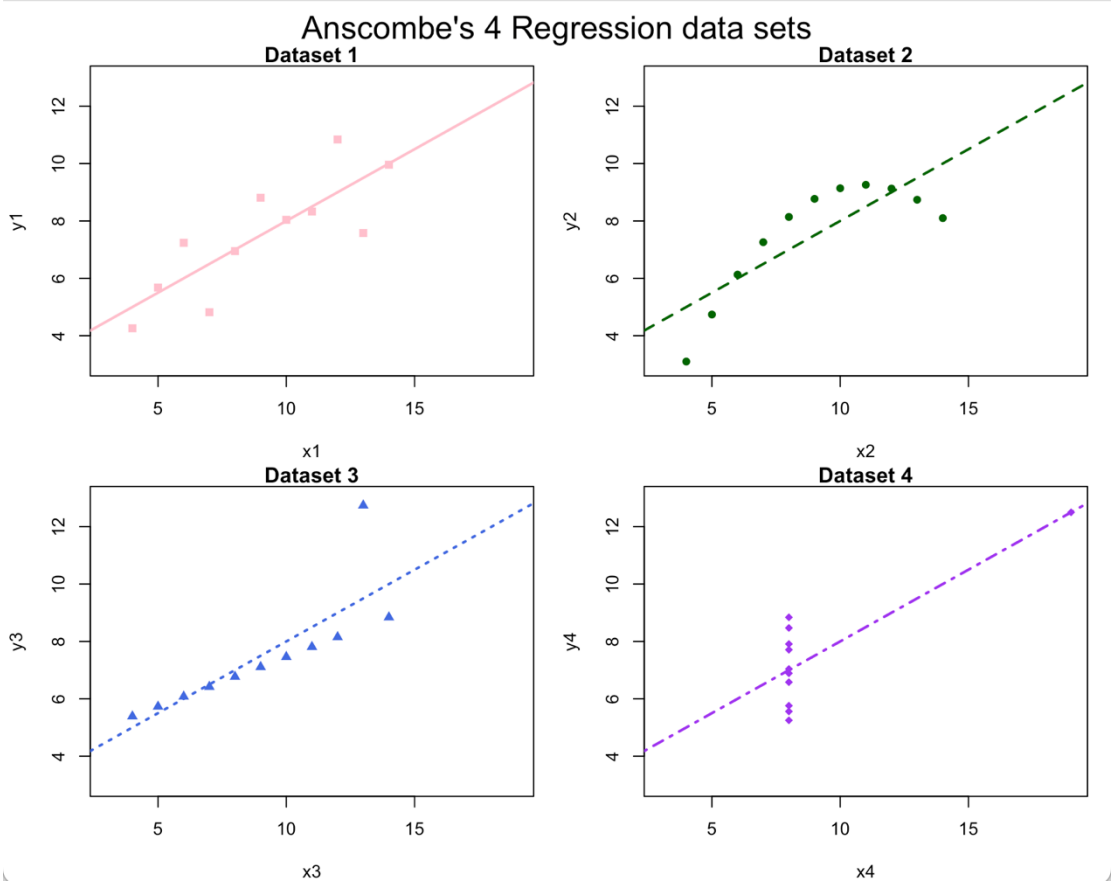
```
> lm4 <- lm(y4 ~ x4, data=anscombe)
> summary(lm4)

Call:
lm(formula = y4 ~ x4, data = anscombe)

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 *
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
```

However, once we plot the graphs of each group, it becomes clear that their distributions are actually very different.



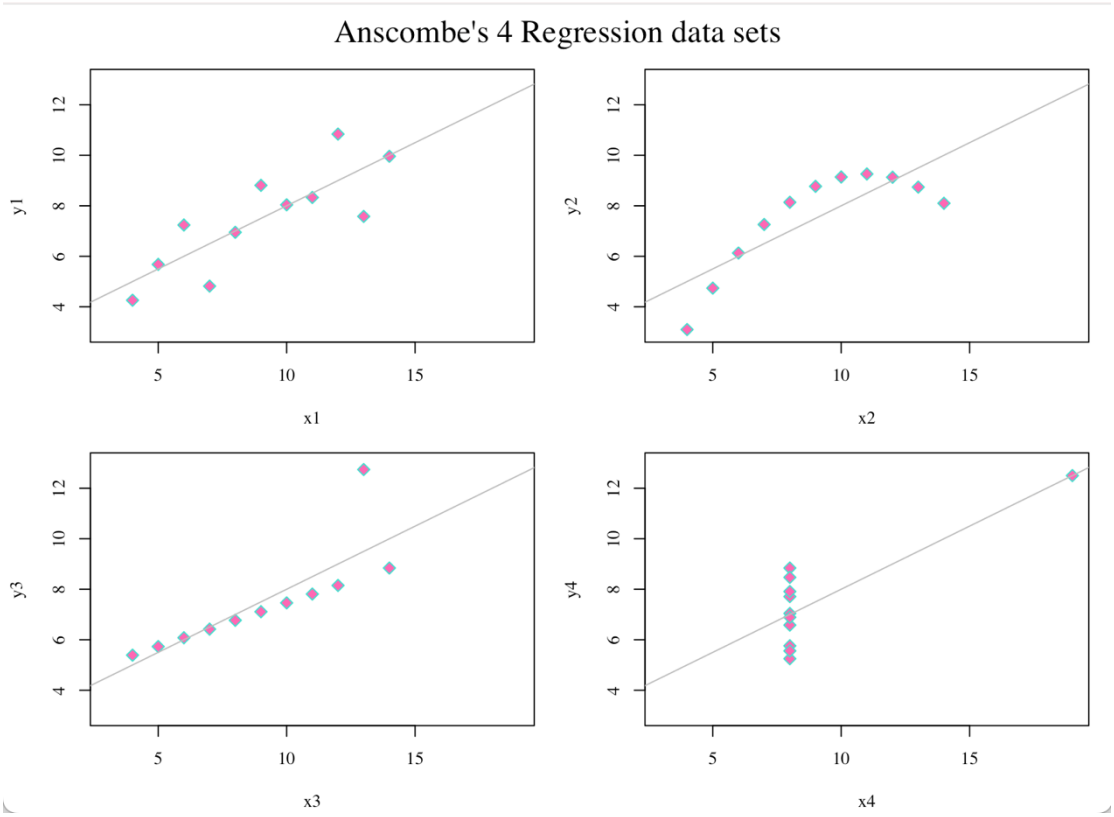
b. different ways to create the plot

```r
# Different styles for each subplot
colors <- c("pink", "darkgreen", "royalblue", "purple")
symbols <- c(15, 16, 17, 18)
linetypes <- c(1, 2, 3, 4)

# Preparing for the plots
op <- par(mfrow = c(2, 2), mar = 0.1+c(4,4,1,1), oma =  c(0, 0, 2, 0))

# Plot charts using for loop
for(i in 1:4) {
  ff[2:3] <- lapply(paste0(c("y","x"), i), as.name)
  plot(ff, data = anscombe,
       pch = symbols[i],
       col = colors[i],
       bg = adjustcolor(colors[i], alpha.f = 0.5),
       xlim = c(3, 19), ylim = c(3, 13),
       main = paste("Dataset", i),
       xlab = paste0("x", i), ylab = paste0("y", i))
  abline(mods[[i]], col = colors[i], lty = linetypes[i], lwd = 2)
}
mtext("Anscombe's 4 Regression data sets", outer = TRUE, cex = 1.5)
par(op)
```



3. serif font +non-default colors+own plotting colors

```r
data(anscombe)
lm1 <- lm(y1 ~ x1, data=anscombe)
lm2 <- lm(y2 ~ x2, data=anscombe)
lm3 <- lm(y3 ~ x3, data=anscombe)
lm4 <- lm(y4 ~ x4, data=anscombe)


ff <- y ~ x
mods <- setNames(as.list(1:4), paste0("lm", 1:4))
# Plot using for loop
for(i in 1:4) {
  ff[2:3] <- lapply(paste0(c("y","x"), i), as.name)
  ## or   ff[[2]] <- as.name(paste0("y", i))
  ##      ff[[3]] <- as.name(paste0("x", i))
  mods[[i]] <- lmi <- lm(ff, data = anscombe)
  #print(anova(lmi))
}


# Preparing for the plots
op <- par(mfrow = c(2, 2), mar = 0.1+c(4,4,1,1), oma =  c(0, 0, 2, 0), family = "serif") #family

# Plot charts using for loop
for(i in 1:4) {
  ff[2:3] <- lapply(paste0(c("y","x"), i), as.name)
  plot(ff, data = anscombe, col = "turquoise", pch = 23, bg = "hotpink", cex = 1.2,
       xlim = c(3, 19), ylim = c(3, 13))
  abline(mods[[i]], col = "gray")
}
mtext("Anscombe's 4 Regression data sets", outer = TRUE, cex = 1.5)
par(op)
```



4. ggplot

```r
library(tidyverse)
library(gridExtra)

p1 = ggplot(anscombe, aes(x = x1, y = y1)) +
  geom_point(
    shape = 23,
    color = "deeppink",
    fill = "turquoise",
    size = 3 ) +
  geom_smooth(method = "lm", se = FALSE, color = "darkgray") +
  labs(
    title = "Anscombe's Quartet - Dataset 1 (ggplot2)"
  ) +
  theme_minimal(base_family = "serif")

p2 = ggplot(anscombe, aes(x = x2, y = y2)) +
  geom_point(
    shape = 23,
    color = "deeppink",
    fill = "turquoise",
    size = 3 ) +
  geom_smooth(method = "lm", se = FALSE, color = "darkgray") +
  labs(
    title = "Anscombe's Quartet - Dataset 2 (ggplot2)"
  ) +
  theme_minimal(base_family = "serif")

p3 = ggplot(anscombe, aes(x = x3, y = y3)) +
  geom_point(
    shape = 23,
    color = "deeppink",
    fill = "turquoise",
    size = 3 ) +
  geom_smooth(method = "lm", se = FALSE, color = "darkgray") +
  labs(
    title = "Anscombe's Quartet - Dataset 3 (ggplot2)"
  ) +
  theme_minimal(base_family = "serif")

p4 = ggplot(anscombe, aes(x = x4, y = y4)) +
  geom_point(
    shape = 23,
    color = "deeppink",
    fill = "turquoise",
    size = 3 ) +
  geom_smooth(method = "lm", se = FALSE, color = "darkgray") +
  labs(
    title = "Anscombe's Quartet - Dataset 4 (ggplot2)"
  ) +
  theme_minimal(base_family = "serif")


grid.arrange(p1, p2, p3, p4, ncol = 2)
```