

Assignment 4

Bongani Mveng

11 July 2016

Hello Octocat

I love Octocat. She's the coolest cat in town.



Figure 1: Octocat

```
data(anscombe)
dim(anscombe)
```

```
## [1] 11 8
```

```
colnames(anscombe)
```

```
## [1] "x1" "x2" "x3" "x4" "y1" "y2" "y3" "y4"
```

```
head(anscombe)
```

```
##  x1 x2 x3 x4  y1  y2  y3  y4
## 1 10 10 10 8 8.04 9.14 7.46 6.58
## 2  8  8  8 8 6.95 8.14 6.77 5.76
## 3 13 13 13 8 7.58 8.74 12.74 7.71
## 4  9  9  9 8 8.81 8.77  7.11 8.84
## 5 11 11 11 8 8.33 9.26  7.81 8.47
## 6 14 14 14 8 9.96 8.10  8.84 7.04
```

```
tail(anscombe)
```

```
##  x1 x2 x3 x4  y1  y2  y3  y4
## 6 14 14 14 8 9.96 8.10 8.84 7.04
## 7  6  6  6 8 7.24 6.13 6.08 5.25
```

```
## 8  4  4  4 19  4.26 3.10 5.39 12.50
## 9 12 12 12  8 10.84 9.13 8.15  5.56
## 10 7  7  7  8  4.82 7.26 6.42  7.91
## 11 5  5  5  8  5.68 4.74 5.73  6.89
```

```
summary(anscombe)
```

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

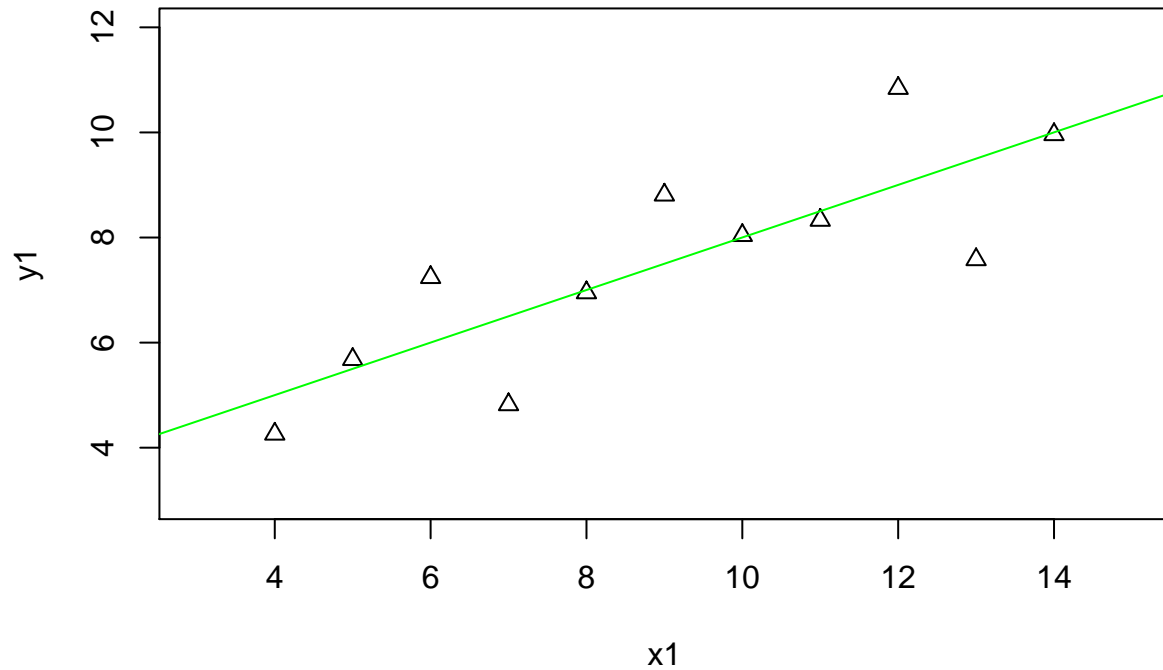
```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##   filter, lag
```

```
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
##    x1    y1
## 1  10  8.04
## 2   8  6.95
## 3  13  7.58
## 4   9  8.81
## 5  11  8.33
## 6  14  9.96
## 7   6  7.24
## 8   4  4.26
## 9  12 10.84
## 10  7  4.82
## 11  5  5.68
```

Anscombe Scatterplot



```
library(readr)
df <- read.csv("analgesic.csv")
```

```
dim(df)
```

```
## [1] 40  5
```

```
colnames(df)
```

```
## [1] "ID"          "Group"       "Measurement_1" "Measurement_2"
## [5] "Measurement_3"
```

```
head(df)
```

```
##   ID    Group Measurement_1 Measurement_2 Measurement_3
## 1  1 Analgesic           26           26           21
## 2  2 Analgesic           29           26           23
## 3  3 Analgesic           24           28           22
## 4  4 Analgesic           25           22           24
## 5  5 Analgesic           24           28           23
## 6  6 Analgesic           22           23           26
```

```
tail(df)
```

```
##      ID   Group Measurement_1 Measurement_2 Measurement_3
## 35 35 Placebo           17           21           15
## 36 36 Placebo           19           17           15
## 37 37 Placebo           14           19           13
## 38 38 Placebo           17           19           13
## 39 39 Placebo           11           20           18
## 40 40 Placebo           15           18           12
```

```
summary(df)
```

```
##           ID           Group Measurement_1 Measurement_2
## Min.      : 1.00   Analgesic:20   Min.      :10.00   Min.      : 8.0
## 1st Qu.:10.75   Placebo  :20   1st Qu.:17.00   1st Qu.:17.0
## Median :20.50                        Median :20.00   Median :20.0
## Mean    :20.50                        Mean    :20.12   Mean    :20.7
## 3rd Qu.:30.25                        3rd Qu.:24.00   3rd Qu.:25.0
## Max.     :40.00                        Max.     :30.00   Max.     :32.0
## Measurement_3
## Min.      :12.00
## 1st Qu.:16.00
## Median :20.50
## Mean     :20.52
## 3rd Qu.:24.25
## Max.     :30.00
```

```
library(tidyr)
library(dplyr)
# Tidy the data from a wide to long format
df.new <- gather(df, Replicate_reading, Measurement, Measurement_1:Measurement_3)

# Group by the 'Group' column ("Analgesic", "Placebo")
grouped <- group_by(df.new, Group)
grouped
```

```
## Source: local data frame [120 x 4]
## Groups: Group [2]
##
##      ID   Group Replicate_reading Measurement
##   <int> <fctr>          <chr>          <int>
## 1     1 Analgesic    Measurement_1          26
## 2     2 Analgesic    Measurement_1          29
## 3     3 Analgesic    Measurement_1          24
## 4     4 Analgesic    Measurement_1          25
## 5     5 Analgesic    Measurement_1          24
## 6     6 Analgesic    Measurement_1          22
## 7     7 Analgesic    Measurement_1          25
## 8     8 Analgesic    Measurement_1          28
## 9     9 Analgesic    Measurement_1          22
## 10    10 Analgesic    Measurement_1          18
## .. ... ..
## .. ... ..
```

```
# Group by the 'ID' column
```

```
grouped.2 <- group_by(grouped, ID)
grouped.2
```

```
## Source: local data frame [120 x 4]
```

```
## Groups: ID [40]
```

```
##
##      ID      Group Replicate_reading Measurement
##   <int>    <fctr>          <chr>          <int>
## 1      1 Analgesic    Measurement_1         26
## 2      2 Analgesic    Measurement_1         29
## 3      3 Analgesic    Measurement_1         24
## 4      4 Analgesic    Measurement_1         25
## 5      5 Analgesic    Measurement_1         24
## 6      6 Analgesic    Measurement_1         22
## 7      7 Analgesic    Measurement_1         25
## 8      8 Analgesic    Measurement_1         28
## 9      9 Analgesic    Measurement_1         22
## 10     10 Analgesic    Measurement_1         18
## .. ...
```

```
# Get the mean for every individual's ("ID") measurements
```

```
sum <- summarize(grouped.2, mean(Measurement))
```

```
# Print the final dataframe
```

```
sum
```

```
## Source: local data frame [40 x 2]
```

```
##
##      ID mean(Measurement)
##   <int>          <dbl>
## 1      1      24.33333
## 2      2      26.00000
## 3      3      24.66667
## 4      4      23.66667
## 5      5      25.00000
## 6      6      23.66667
## 7      7      26.66667
## 8      8      23.33333
## 9      9      22.66667
## 10     10      24.00000
## .. ...
```