# Introduction to Linear Models

## Linear Models with R Chapter 1

### Diabetes

```
library(faraway)
library(ggplot2)
```

```
## Warning in register(): Can't find generic 'scale_type' in package ggplot2 to
## register S3 method.
```

```
data(pima, package = 'faraway')
head(pima)
```

```
##   pregnant glucose diastolic triceps insulin  bmi diabetes age test
## 1        6     148        72      35       0 33.6    0.627  50    1
## 2        1      85        66      29       0 26.6    0.351  31    0
## 3        8     183        64       0       0 23.3    0.672  32    1
## 4        1      89        66      23      94 28.1    0.167  21    0
## 5        0     137        40      35     168 43.1    2.288  33    1
## 6        5     116        74       0       0 25.6    0.201  30    0
```

```
summary(pima)
```

```
##     pregnant         glucose        diastolic         triceps
##  Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00
##  Median : 3.000   Median :117.0   Median : 72.00   Median :23.00
##  Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54
##  3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00
##  Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00
##     insulin           bmi           diabetes           age
##  Min.   :  0.0   Min.   : 0.00   Min.   :0.0780   Min.   :21.00
##  1st Qu.:  0.0   1st Qu.:27.30   1st Qu.:0.2437   1st Qu.:24.00
##  Median : 30.5   Median :32.00   Median :0.3725   Median :29.00
##  Mean   : 79.8   Mean   :31.99   Mean   :0.4719   Mean   :33.24
##  3rd Qu.:127.2   3rd Qu.:36.60   3rd Qu.:0.6262   3rd Qu.:41.00
##  Max.   :846.0   Max.   :67.10   Max.   :2.4200   Max.   :81.00
##       test
##  Min.   :0.000
##  1st Qu.:0.000
##  Median :0.000
##  Mean   :0.349
##  3rd Qu.:1.000
##  Max.   :1.000
```

We can see that this data has zeros in inappropriate places. These are actually erroneous values...

```
head(sort(pima$diastolic), 50)
```

```
##  [1]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## [26]  0  0  0  0  0  0  0  0  0  0 24 30 30 38 40 44 44 44 44 46 46 48 48 48 48
```

```
pima$diastolic[pima$diastolic == 0] <- NA
pima$glucose[pima$glucose == 0] <- NA
pima$triceps[pima$triceps == 0] <- NA
pima$insulin[pima$insulin == 0] <- NA
pima$bmi[pima$bmi == 0] <- NA
```

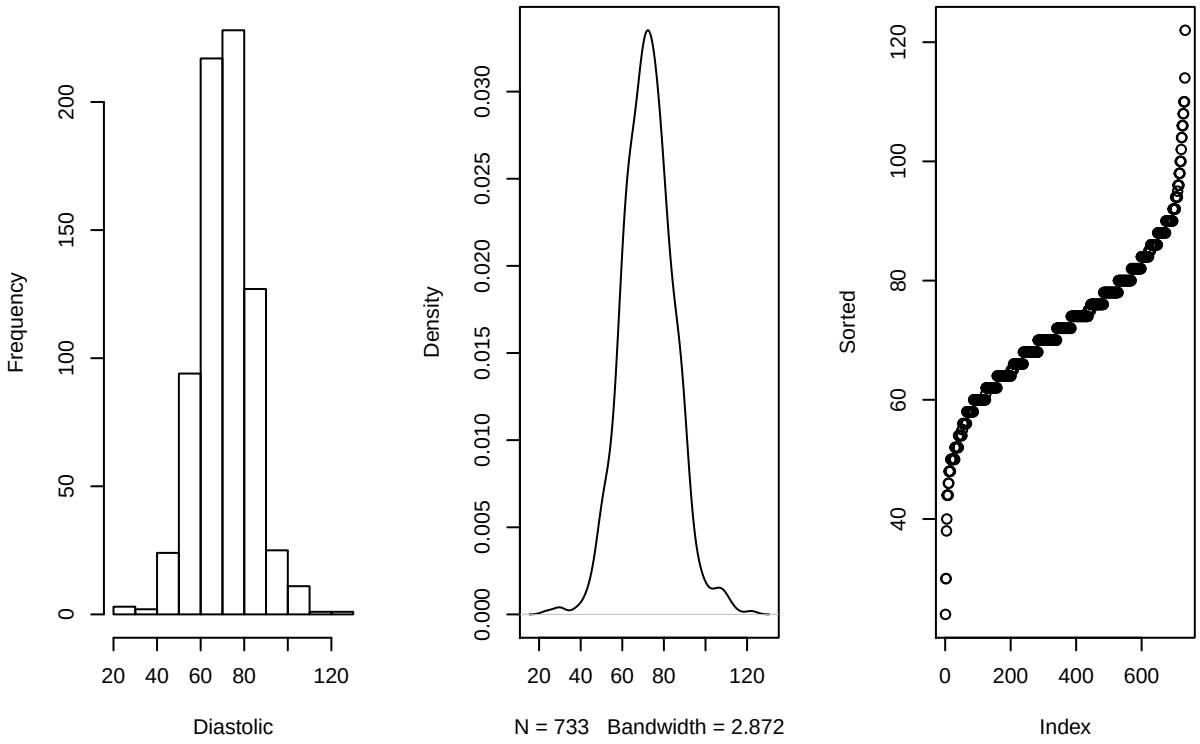
Change the boolean to a factor.

```
pima$test <- factor(pima$test)
levels(pima$test) <- c('negative', 'positive')
summary(pima)
```

```
##     pregnant         glucose        diastolic         triceps
##  Min.   : 0.000   Min.   : 44.0   Min.   : 24.00   Min.   : 7.00
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 64.00   1st Qu.:22.00
##  Median : 3.000   Median :117.0   Median : 72.00   Median :29.00
##  Mean   : 3.845   Mean   :121.7   Mean   : 72.41   Mean   :29.15
##  3rd Qu.: 6.000   3rd Qu.:141.0   3rd Qu.: 80.00   3rd Qu.:36.00
##  Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00
##                   NA's   :5       NA's   :35       NA's   :227
##     insulin            bmi           diabetes          age
##  Min.   : 14.00   Min.   :18.20   Min.   :0.0780   Min.   :21.00
##  1st Qu.: 76.25   1st Qu.:27.50   1st Qu.:0.2437   1st Qu.:24.00
##  Median :125.00   Median :32.30   Median :0.3725   Median :29.00
##  Mean   :155.55   Mean   :32.46   Mean   :0.4719   Mean   :33.24
##  3rd Qu.:190.00   3rd Qu.:36.60   3rd Qu.:0.6262   3rd Qu.:41.00
##  Max.   :846.00   Max.   :67.10   Max.   :2.4200   Max.   :81.00
##  NA's   :374      NA's   :11
##        test
##  negative:500
##  positive:268
##
##
##
##
##
```

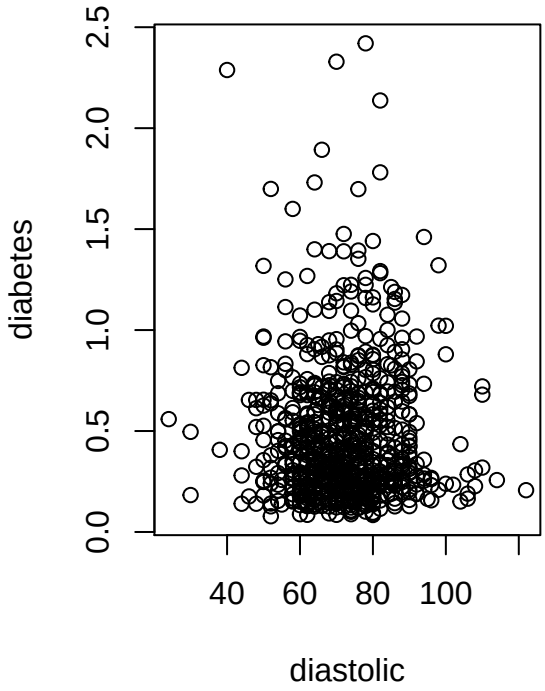
Some basic exploratory plots:

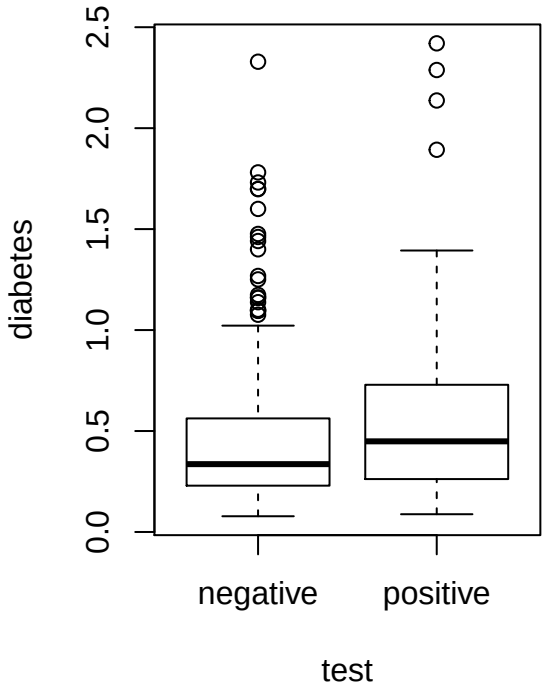
```
par(mfrow = c(1, 3))
hist(pima$diastolic, xlab = 'Diastolic', main = '')
plot(density(pima$diastolic, na.rm = TRUE), main = '')
plot(sort(pima$diastolic), ylab = 'Sorted')
```




Some basic bivariate plots:

```
par(mfrow = c(1, 2))
plot(diabetes ~ diastolic, pima)
plot(diabetes ~ test, pima)
```

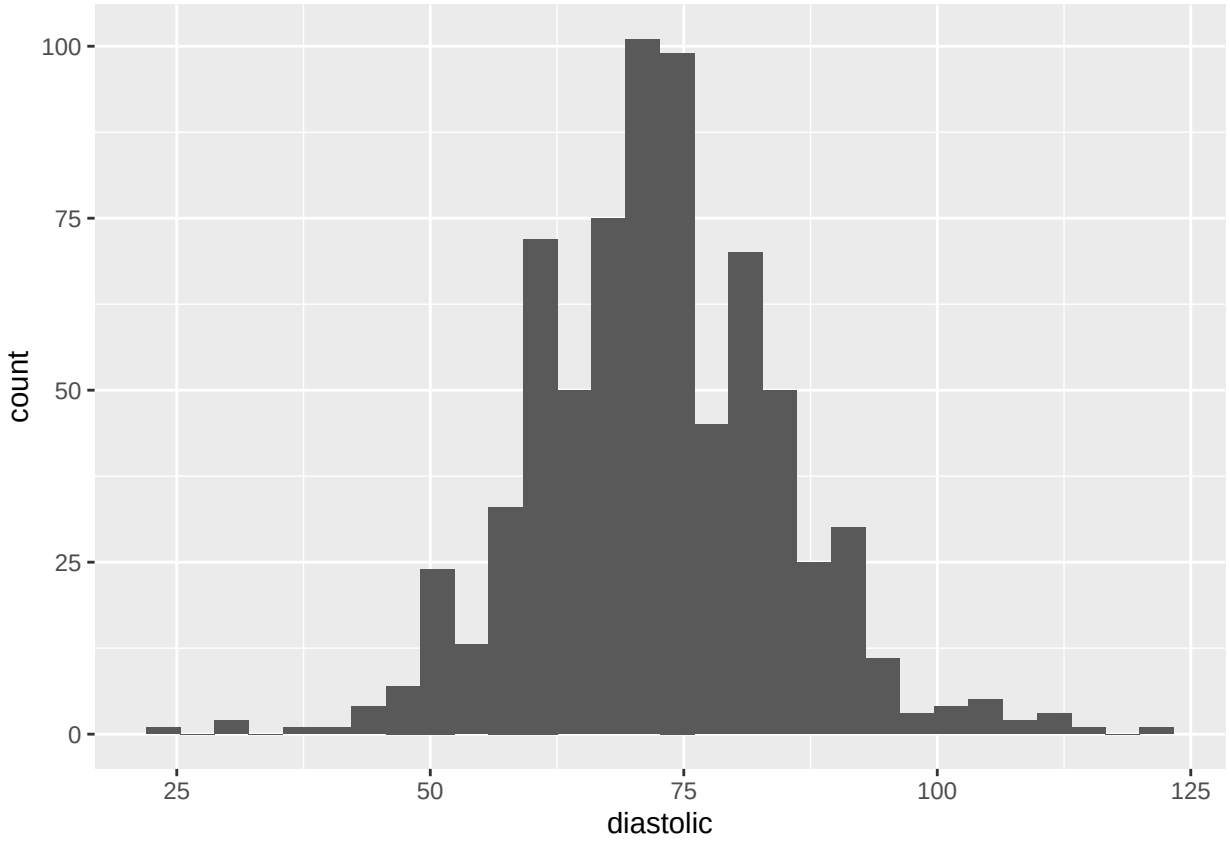



```
ggplot(pima, aes(x = diastolic)) + geom_histogram()
```

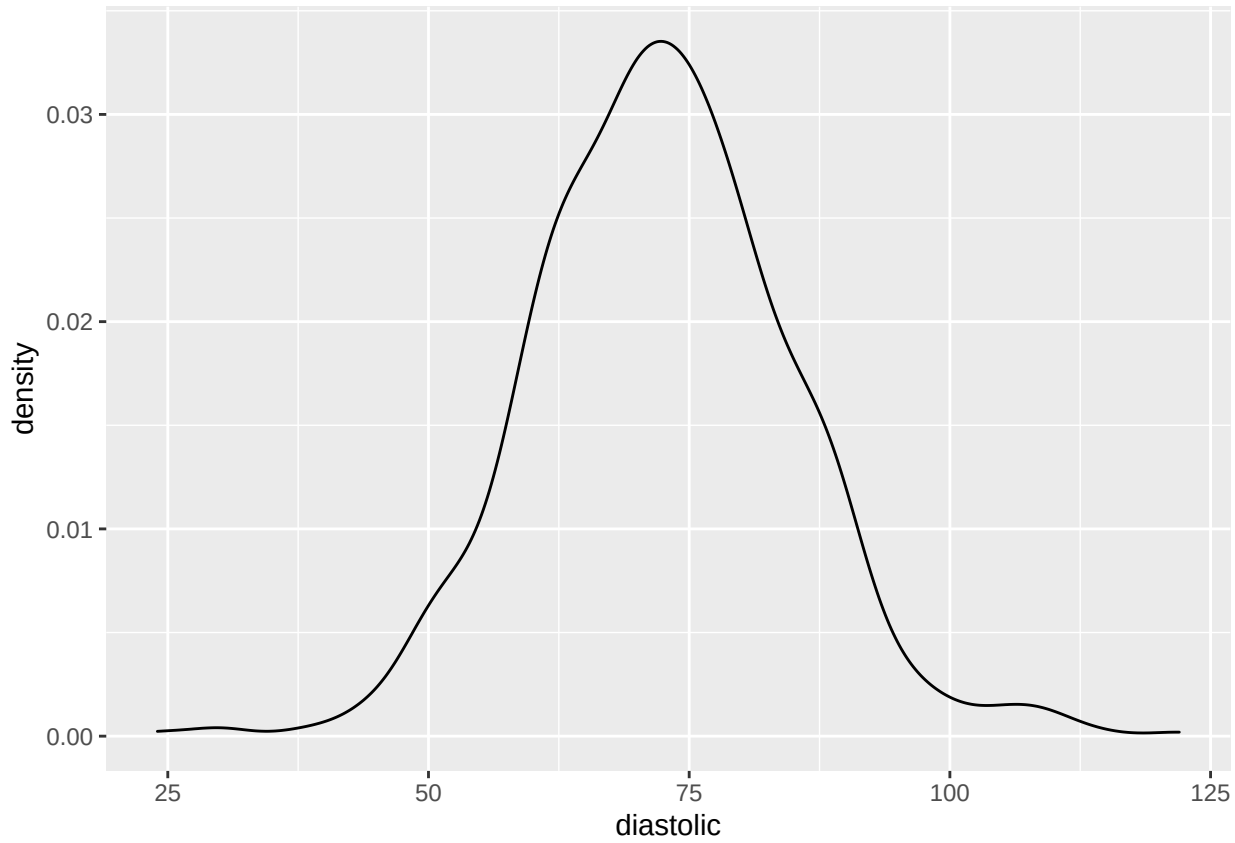
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 35 rows containing non-finite values (stat_bin).
```



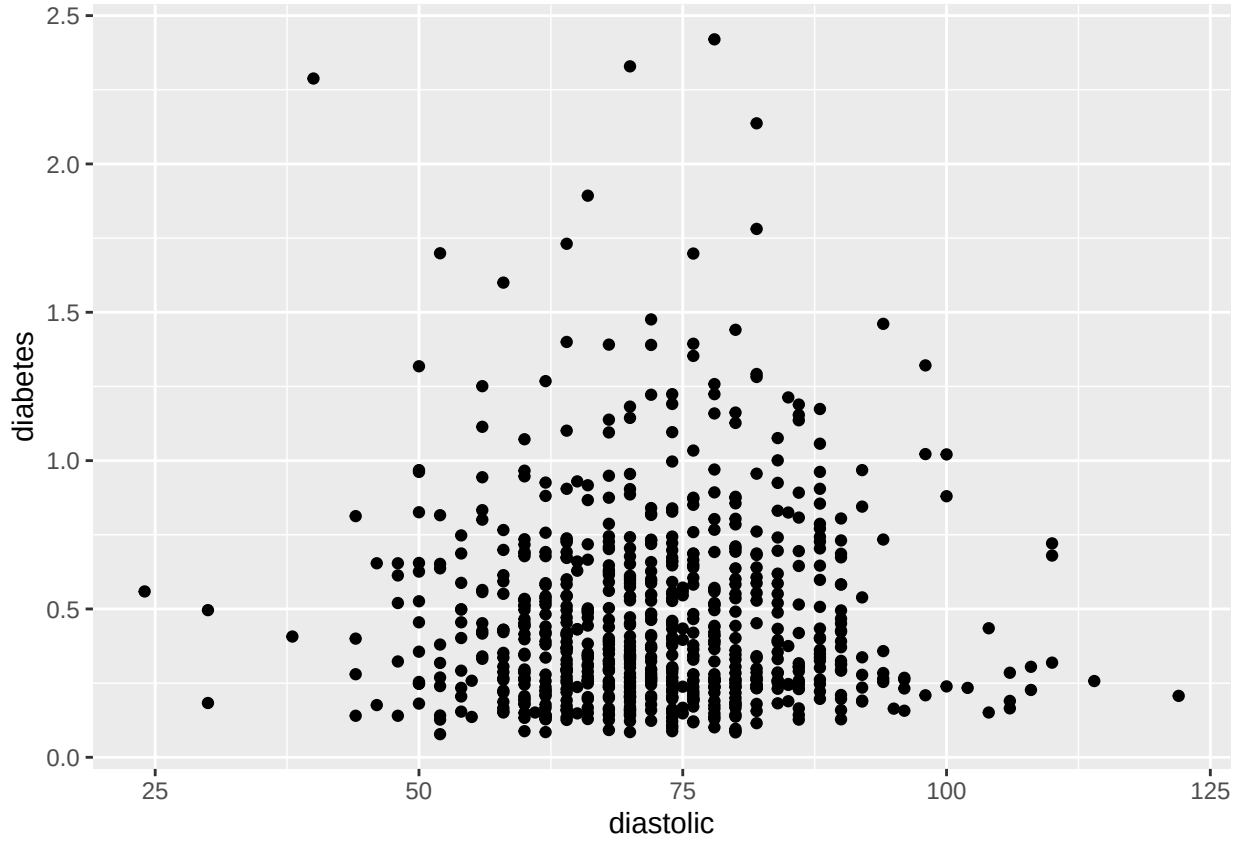

```
ggplot(pima, aes(x = diastolic)) + geom_density()
```

```
## Warning: Removed 35 rows containing non-finite values (stat_density).
```



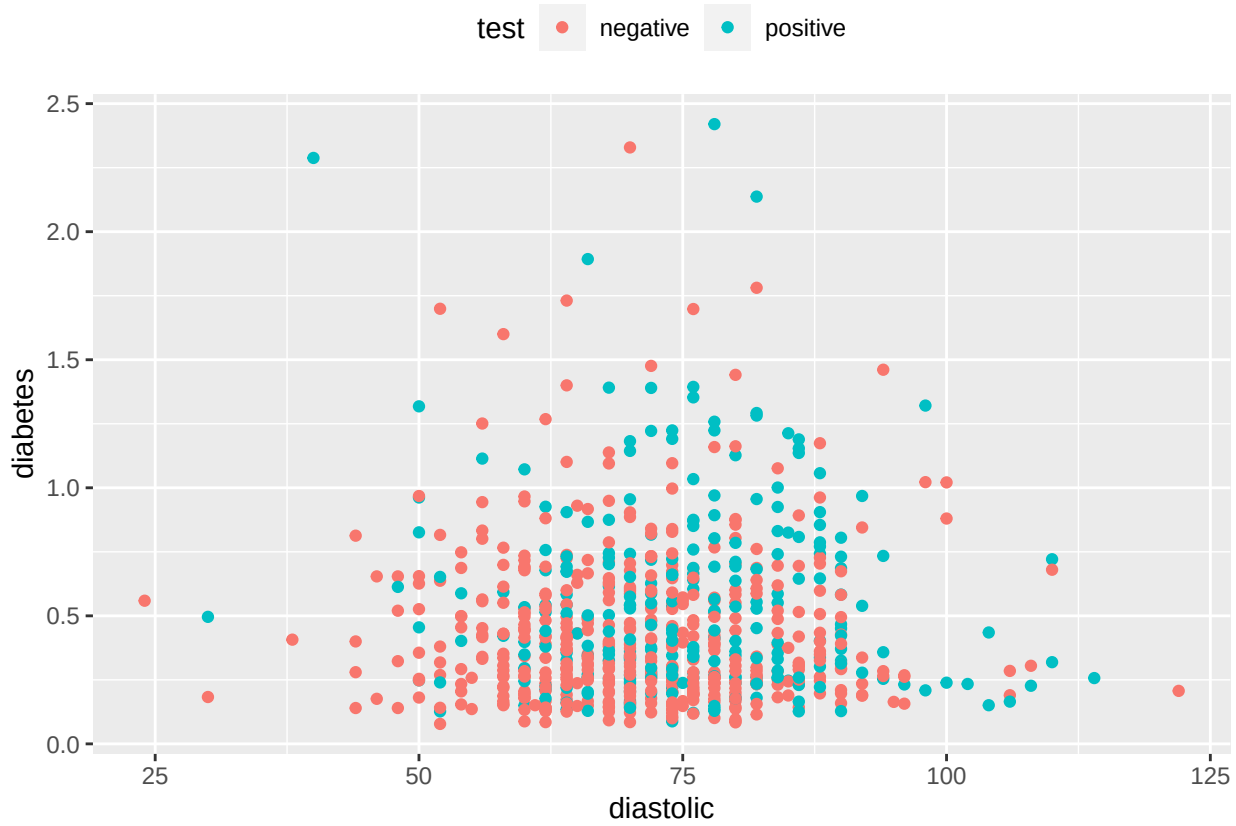

```
ggplot(pima, aes(x = diastolic, y = diabetes)) + geom_point()
```

```
## Warning: Removed 35 rows containing missing values (geom_point).
```



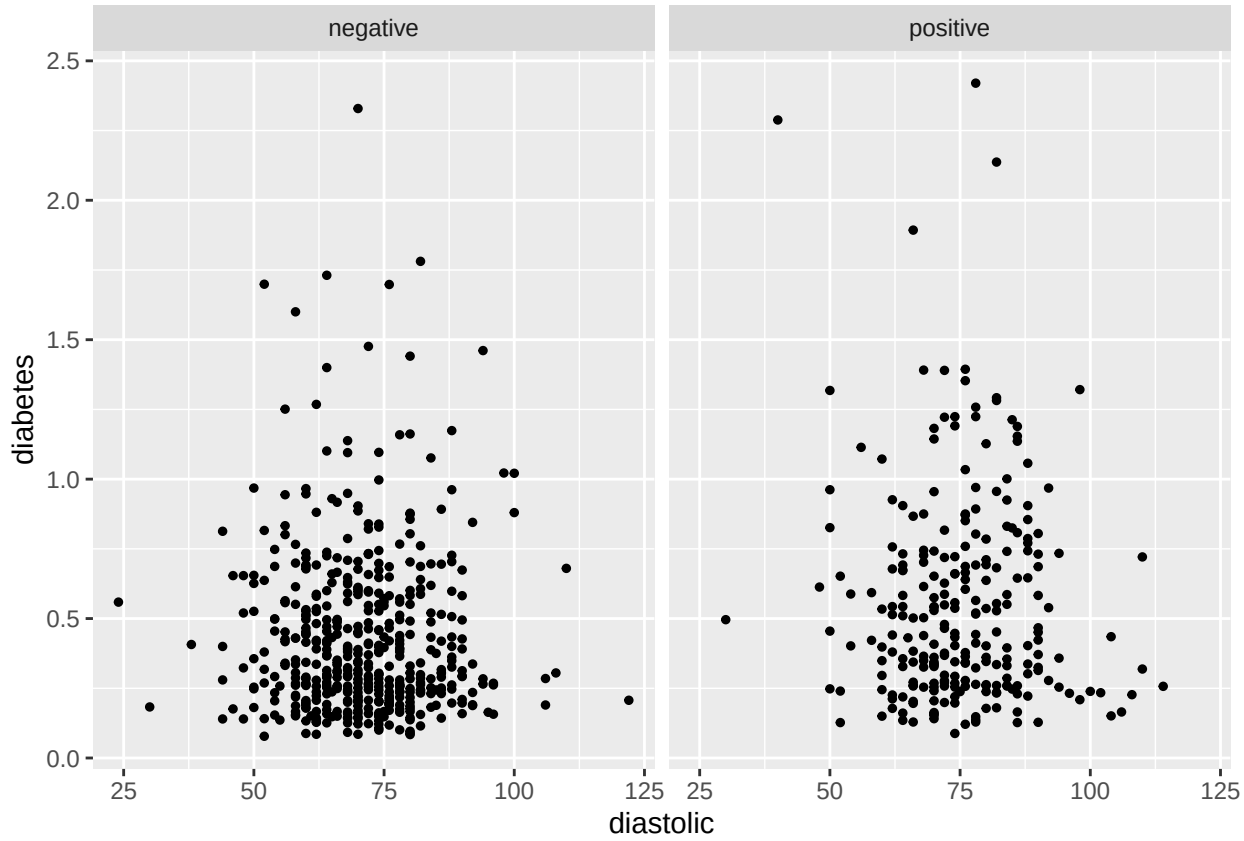

```
ggplot(pima, aes(x = diastolic, y = diabetes, col = test)) +
  geom_point() +
  theme(legend.position = 'top', legend.direction = 'horizontal')
```

```
## Warning: Removed 35 rows containing missing values (geom_point).
```




```
ggplot(pima,aes(x=diastolic,y=diabetes)) +
  geom_point(size=1) +
  facet_grid(~ test)
```

```
## Warning: Removed 35 rows containing missing values (geom_point).
```




## Manilius

Mayer describes the motion of the moon (libration) using the Manilius crater:

```
#\[\text{arc} = \beta + \alpha \text{sinang} + \gamma \text{cosang}\]
```

He wished to obtain values for the three unknowns $\alpha, \beta,$ and$\gamma$. The variables arc, sinang and cosang can be observed using a telescope.

Since there are three unknowns, we need only three distinct observations of the set of three variables to find a unique solution for $\alpha, \beta,$ and$\gamma$. Embarassingly for Mayer, there were 27 sets of observations available. Astronomical measurements were naturally subject to some variation and so there was no solution that fit all 27 observations.

```
data(manilius, package = 'faraway')
head(manilius)
```

```
##        arc sinang  cosang group
## 1 13.16667 0.8836 -0.4682     1
## 2 13.13333 0.9996 -0.0282     1
## 3 13.20000 0.9899  0.1421     1
## 4 14.25000 0.2221  0.9750     3
## 5 14.70000 0.0006  1.0000     3
## 6 13.01667 0.9308 -0.3654     1
```

Mayer's solution was to divide the data into three groups so that observations within each group were similar in some respect. He then computed the sum of the variables within each group. We can also do this:

```
(moon3 <- aggregate(manilius[,1:3], list(manilius$group), sum))
```

```
##   Group.1      arc  sinang  cosang
## 1       1 118.1333  8.4987 -0.7932
## 2       2 140.2833 -6.1404  1.7443
## 3       3 127.5333  2.9777  7.9649
```

3 equations, 3 unknowns... Solved with matrices.

```
solve(cbind(9, moon3$sinang, moon3$cosang), moon3$arc)
```

```
## [1] 14.5445859 -1.4898221  0.1341264
```

Suppose it isn't exact, and we add an error term...

```
#\[\text{arc}_i = \beta + \alpha \text{sinang}_i + \gamma \text{cosang}_i + \epsilon_i\]
```

Don't forget that $i = 1, 2, ..., 27$. We can find the $\alpha, \beta, \gamma$ which minimize the sum of the squared errors: $\sum \epsilon^2$.

```
lmod <- lm(arc ~ sinang + cosang, manilius)
coef(lmod)
```

```
## (Intercept)      sinang      cosang
## 14.56162351 -1.50458123  0.09136504
```

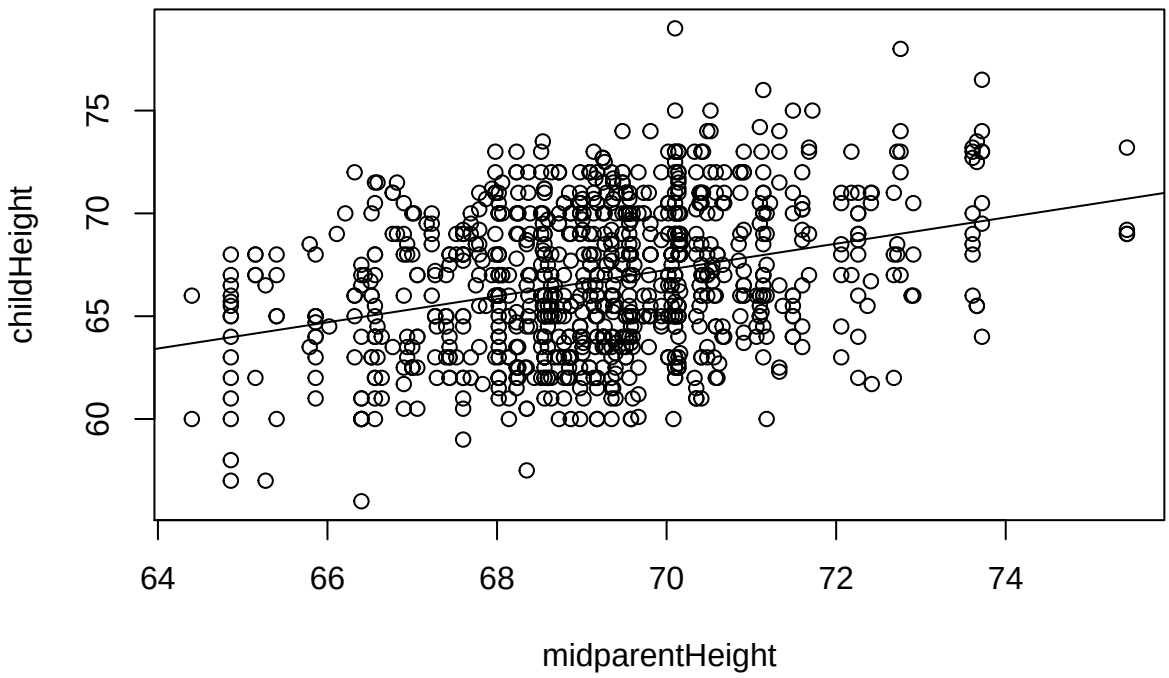
---

## Galton families

```
# \[\text{childHeight} = \alpha + \beta \text{}midparentHeight + \epsilon\]

data(GaltonFamilies, package = 'HistData')
plot(childHeight ~ midparentHeight, GaltonFamilies)

lmod <- lm(childHeight ~ midparentHeight, GaltonFamilies)

coef(lmod)
```

```
##     (Intercept) midparentHeight
##      22.6362405       0.6373609
```

```
abline(lmod)
```




$$\frac{y - \bar{y}}{SD_y} = r \frac{x - \bar{x}}{SD_x}$$

$r$ is the correlation between x and y.

```
(beta <- with(GaltonFamilies,
              cor(midparentHeight,
                  childHeight) * sd(childHeight) / sd(midparentHeight)))
```

```
## [1] 0.6373609
```

```
(alpha <- with(GaltonFamilies,
               mean(childHeight) - beta * mean(midparentHeight)))
```

```
## [1] 22.63624
```

Set $r = 1$ and we can compute a better line...

```
(beta1 <- with(GaltonFamilies,
               sd(childHeight) / sd(midparentHeight)))
```

```
## [1] 1.985858
```

```
(alpha1 <- with(GaltonFamilies,
               mean(childHeight) - beta1 * mean(midparentHeight)))
```

```
## [1] -70.68889
```

```
plot(childHeight ~ midparentHeight, GaltonFamilies)
abline(lmod)
abline(alpha1, beta1, lty=2)
```