Regression Analysis Application in R

```
library(car)
library(ggplot2)
library(mosaicData)
library(dplyr)
library(psych)
```

Simple Regression

The data set for the reported vs. measured weights is called `Davis` and it is contained in the `car` package.

```
dim(Davis) # dim() gives the number of rows and columns
```

```
## [1] 200   5
```

```
head(Davis) # default is to show first 6 rows
```

```
##   sex weight height repwt repht
## 1   M     77    182    77   180
## 2   F     58    161    51   159
## 3   F     53    161    54   158
## 4   M     68    177    70   175
## 5   F     59    157    59   155
## 6   M     76    170    76   165
```

```
head(Davis, 15)
```

```
##    sex weight height repwt repht
## 1    M     77    182    77   180
## 2    F     58    161    51   159
## 3    F     53    161    54   158
## 4    M     68    177    70   175
## 5    F     59    157    59   155
## 6    M     76    170    76   165
## 7    M     76    167    77   165
## 8    M     69    186    73   180
## 9    M     71    178    71   175
## 10   M     65    171    64   170
## 11   M     70    175    75   174
## 12   F    166     57    56   163
## 13   F     51    161    52   158
## 14   F     64    168    64   165
## 15   F     52    163    57   160
```

```
cDavis <- Davis
cDavis[12, c(2, 3)] <- Davis[12, c(3, 2)]  # correct the recording error
head(cDavis, 15)
```

```
##    sex weight height repwt repht
## 1    M     77    182    77   180
```

```
## 2    F      58     161     51    159
## 3    F      53     161     54    158
## 4    M      68     177     70    175
## 5    F      59     157     59    155
## 6    M      76     170     76    165
## 7    M      76     167     77    165
## 8    M      69     186     73    180
## 9    M      71     178     71    175
## 10   M      65     171     64    170
## 11   M      70     175     75    174
## 12   F      57     166     56    163
## 13   F      51     161     52    158
## 14   F      64     168     64    165
## 15   F      52     163     57    160
```

```
mod <- lm(weight ~ repwt, subset = sex == "F", data=cDavis)
#regression for women
```

Formulas look like `Y ~ X1`, which `lm()` will translate to a regression equation: $Y = b_0 + b_1 X_1 + \epsilon$.

How to get the results from the model object

```
mod
```

```
##
## Call:
## lm(formula = weight ~ repwt, data = cDavis, subset = sex == "F")
##
## Coefficients:
## (Intercept)        repwt
##      1.7775       0.9772
```

If you type the name of your model object, you get only the coefficients. You can also obtain them using `coef()`

```
coef(mod)
```

```
## (Intercept)       repwt
##   1.7775034   0.9772242
```

You get more information if you use `summary()`

```
summary(mod)
```

```
##
## Call:
## lm(formula = weight ~ repwt, data = cDavis, subset = sex == "F")
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5248 -0.7526 -0.3654  0.6118  6.3841
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.77750    1.74441   1.019    0.311
## repwt        0.97722    0.03053  32.009   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 2.057 on 99 degrees of freedom
##   (11 observations deleted due to missingness)
## Multiple R-squared:  0.9119, Adjusted R-squared:  0.911
## F-statistic:  1025 on 1 and 99 DF,  p-value: < 2.2e-16
```

For a 1 kg increase in reported weight, the measured weight increases by 0.977, which is statistically significantly different from 0, $t(99) = 32.009, p < .001$. In other words, there is nearly a 1-to-1 relationship. The intercept means that the predicted measured weight is 1.78 kg for those who report a weight of 0 kg. But no one reported a weight of 0 kg and even if they did, it would not make sense. Nevertheless, the intercept is not significantly different from 0, $t(99) = 1.019, p = 0.311$. Note that in the case of simple regression, the overall (or omnibus) F statistic is equal to t^2.

```
32.009^2
```

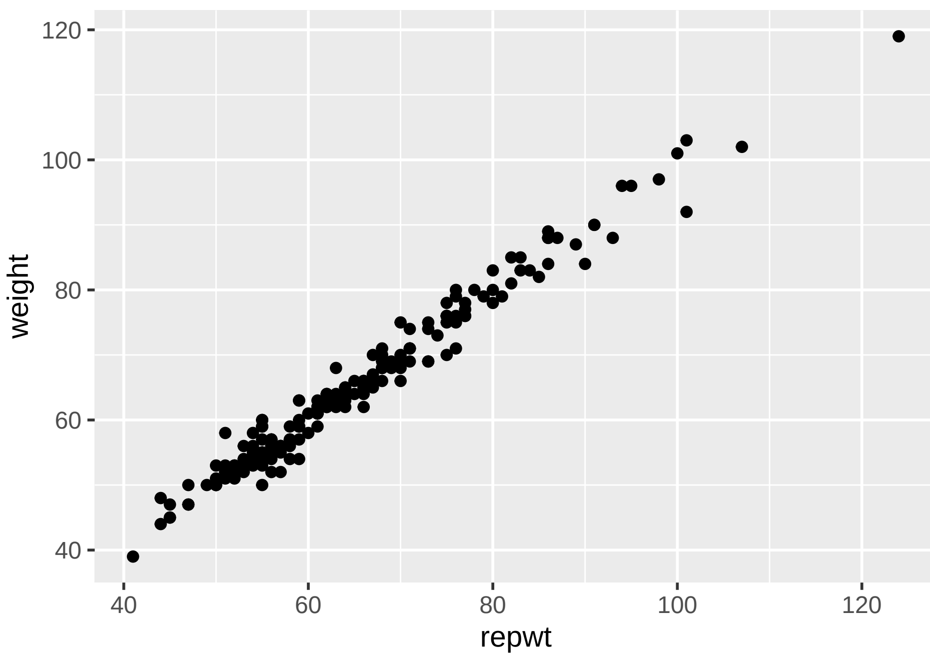
```
## [1] 1024.576
```

Finally, R^2 indicates that 91% of the variability in measured weight can be explained by reported weight.

Let's create a scatterplot.

```
ggplot(data=cDavis, aes(x=repwt, y=weight)) +
  geom_point()
```

```
## Warning: Removed 17 rows containing missing values (`geom_point()`).
```



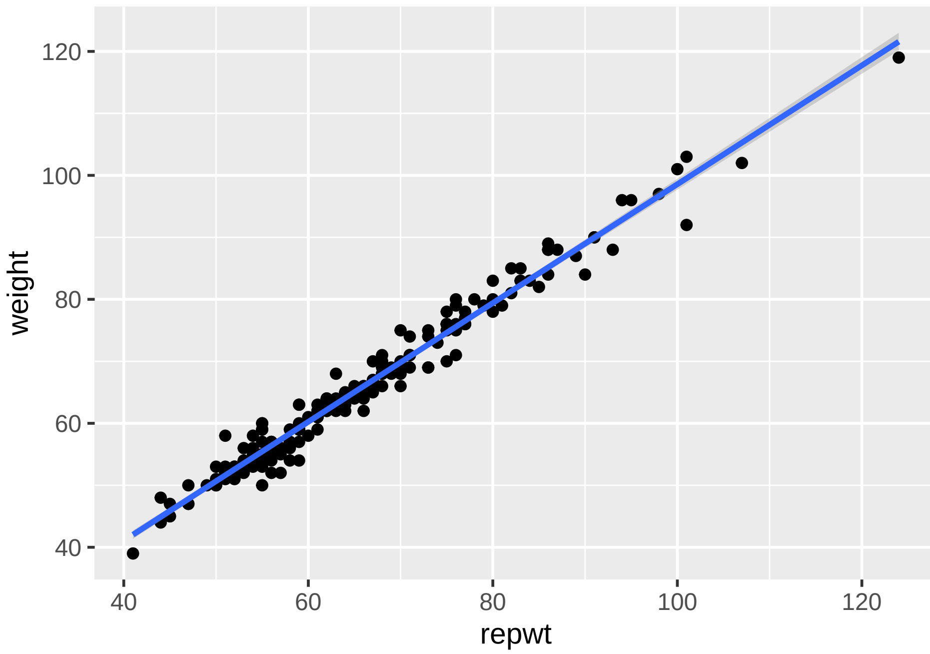
You can add the regression line to the plot.

```
ggplot(data=cDavis, aes(x=repwt, y=weight)) +
  geom_point() +
  geom_smooth(method="lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 17 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 17 rows containing missing values (`geom_point()`).
```



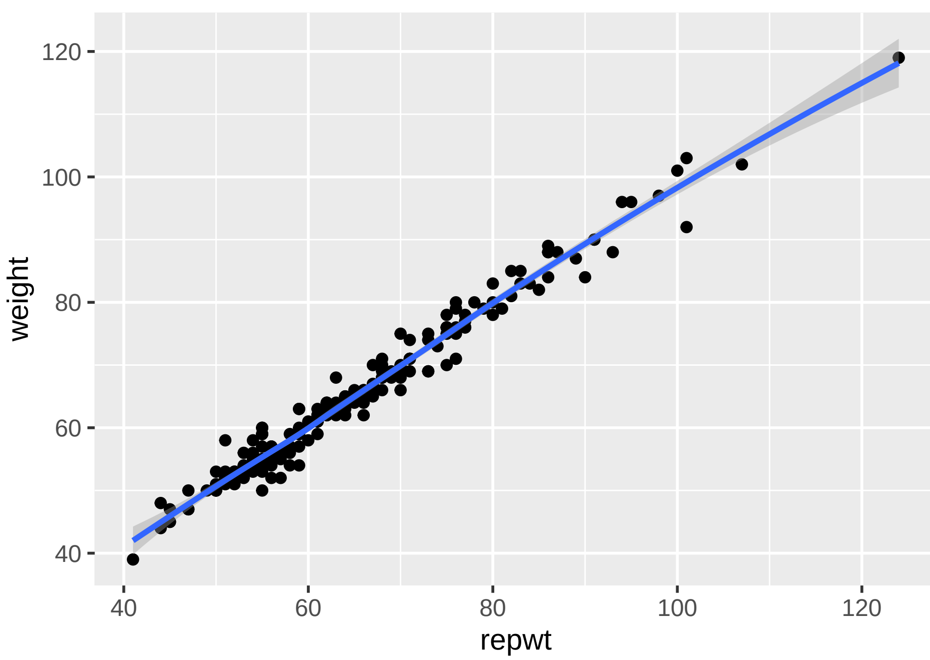
There are other options for `geom_smooth()`, for example `method="loess"`, which gives a local linear regression smoother.

```
ggplot(data=cDavis, aes(x=repwt, y=weight)) +
  geom_point() +
  geom_smooth(method="loess")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 17 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 17 rows containing missing values (`geom_point()`).
```



How about with the original "uncorrected" data:

```r
ggplot(data=Davis, aes(x=repwt, y=weight)) +
  geom_point() +
  geom_smooth(method="loess")
```

```
## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 17 rows containing non-finite values (`stat_smooth()`).

## Warning: Removed 17 rows containing missing values (`geom_point()`).
```