1 Fitting a linear model in R

Now, we shall see how to fit a linear model in our spring example. It is simple to do so in R, as follows :

```
fit=lm(length w̃eight,dat)
fit
model.matrix(fit)
plot(lengthw̃eight,dat)
names(fit)
fit$coef
fit$res
fit$fit
fit$rank
abline(fit$coef)
```

Let us analyse and understand each command.

1.1 fit=lm(length w̃eight, dat)

This command fits a least squares regression model where *length* is the response variable, *weight* is the input variable and *dat* is the data frame in which our information about length and weight for each trial is stored. R calculates the values of the unknown parameters using an algorithm and stores them in the object *fit*.

1.2 fit

```
Call:
lm(formula = length w̃eight, data = dat)

Coefficients:
(Intercept) weight
3.283 2.010
```

This command displays the calculated values of the unknown parameters of our linear model. In our spring example we have two unknown parameters, whose values are stored in the object *fit*. Hence, $\beta_1 = 3.283, and, \beta_2 = 2.010$.

1.3 model.matrix(fit)

This command prints the design matrix which, in our example, has 6 rows (corresponding to 6 springs) and 2 columns :

```
(Intercept) weight
1 1.0
1 1.5
1 2.0
1 2.5
1 3.0
1 3.5
```

1.4 names(fit)

```
[1] "coefficients" "residuals" "effects" "rank" "fitted.values" "assign" "qr"
[8] "df.residual" "xlevels" "call" "terms" "model"
```

This command displays the various components present in the object *fit*:

coefficients : the vector of unknown parameters
residuals : the residuals, i.e, response minus fitted values.
fitted.values : the fitted mean values.
rank : the rank of the design matrix.
df.residual : the residual degrees of freedom.

1.5 components in *fit*

```
fit$coef
(Intercept) weight
3.283143 2.009714
fit$res
1 2 3 4 5 6
-0.002857143 0.012285714 -0.022571429 0.022571429 -0.012285714 0.002857143
fit$fit
1 2 3 4 5 6
5.292857 6.297714 7.302571 8.307429 9.312286 10.317143
fit$rank
[1] 2
```

1.6 plot(length ˜weight,dat)

1.7 abline(fit$coef)

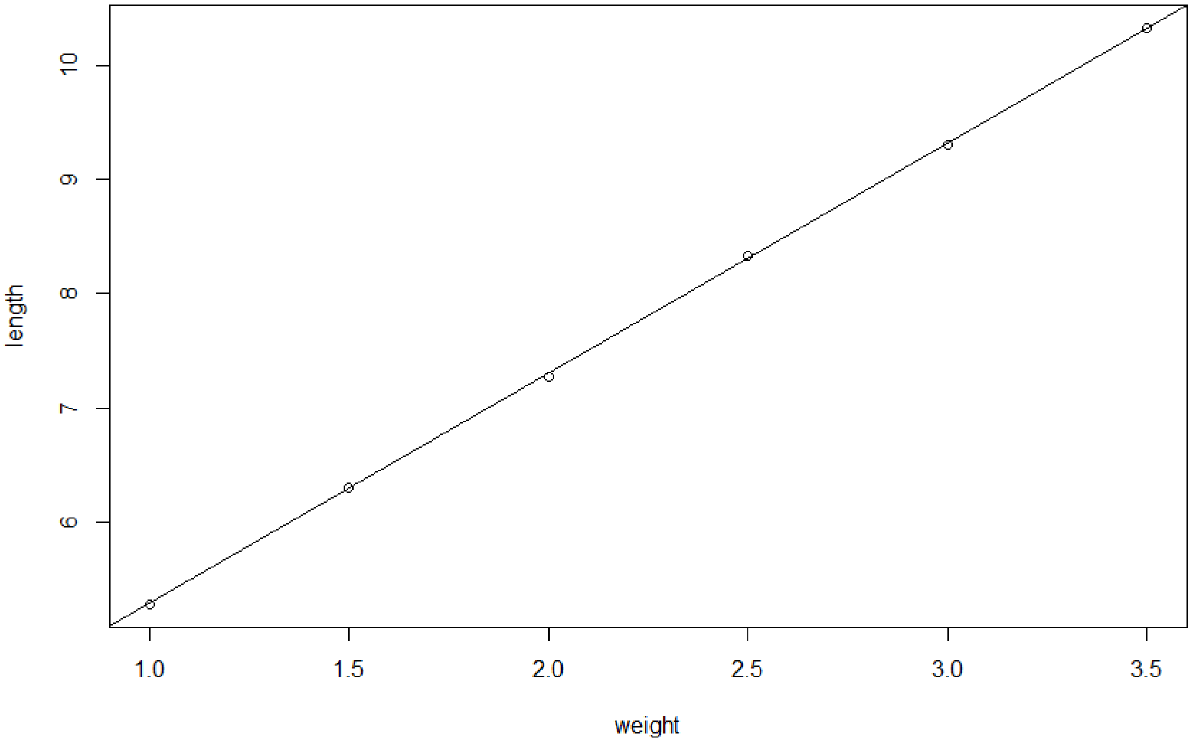

Figure 1: Fitted linear model in R.

This command displays a graph of length versus weight. The *abline()* command gives a straight line with intercept and slope as the calculated coefficients. Here we see that the 6 points are almost lying on a straight line, with hardly any random variation. Hence, a linear model would be a very good fit for our particular example.