Linear regression

Adi Sarid

We talked in the previous unit about clustering, which is part of **unsupervised learning**. In this unit we discuss linear regression which is a supervised learning model. This is a relatively simple supervised learning model, but will help us demonstrate some principles, starting with this one:

In a supervised learning model, we are looking for a function f which yields the best prediction considering:

$$Y = f(X) + \epsilon$$

Linear regression assumes that the underlying structure we are looking for is expressed by:

$$Y = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p \ + \epsilon$$

We solve a minimization problem that finds the coefficients $\{\beta_i\}_{i=0}^{p}$, which yield the minimum error, in this case the minimum sum of squares.

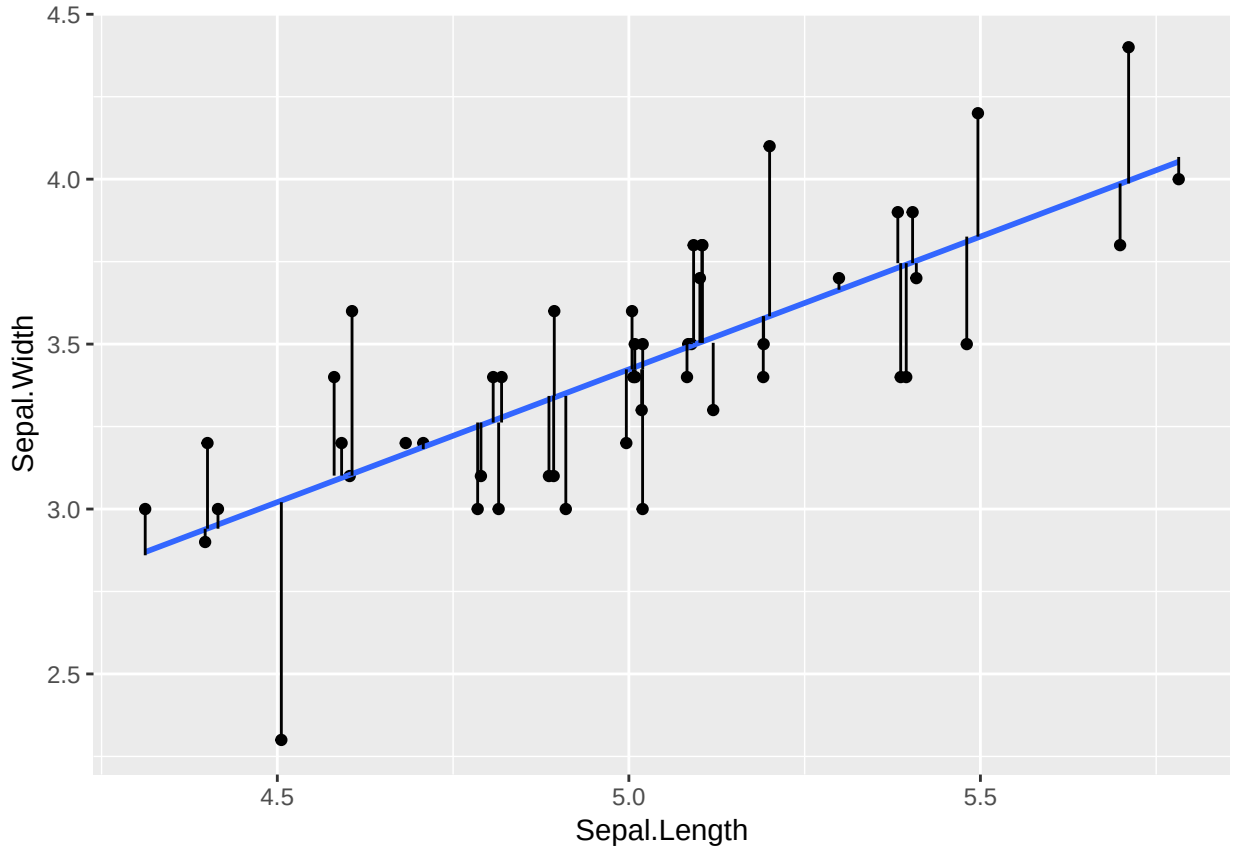
```
library(tidyverse)

iris_setosa <- iris %>%
  filter(Species == "setosa") %>%
  mutate(Sepal.Length = jitter(Sepal.Length))

setosa_lm = lm(data = iris_setosa,
               formula = Sepal.Width ~ Sepal.Length)

iris_lm_errors <- iris_setosa %>%
  mutate(Sepal.Width.pred = predict(setosa_lm,
                                    newdata = iris %>%
                                      filter(Species == "setosa")))

ggplot(iris_lm_errors,
       aes(x=Sepal.Length, y=Sepal.Width)) +
  geom_point() + stat_smooth(method = "lm", se = FALSE) +
  geom_segment(aes(x = Sepal.Length, xend = Sepal.Length, y = Sepal.Width, yend = Sepal.Width.pred))
```



The linear regression solution yields the linear fit which minimizes the observation distances from the line.

A summary of a linear model will look like this:

```
iris_lm_complete <- lm(data = iris,
                       formula = Sepal.Width ~ Sepal.Length + Petal.Width + Petal.Length)

summary(iris_lm_complete)
```

```
##
## Call:
## lm(formula = Sepal.Width ~ Sepal.Length + Petal.Width + Petal.Length,
##     data = iris)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.88045 -0.20945  0.01426  0.17942  0.78125
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.04309    0.27058   3.855 0.000173 ***
## Sepal.Length  0.60707    0.06217   9.765  < 2e-16 ***
## Petal.Width   0.55803    0.12256   4.553  1.1e-05 ***
## Petal.Length -0.58603    0.06214  -9.431  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.3038 on 146 degrees of freedom
## Multiple R-squared:  0.524,  Adjusted R-squared:  0.5142
## F-statistic: 53.58 on 3 and 146 DF,  p-value: < 2.2e-16
```

The estimate reflects the coefficients of the model, i.e.:

$$\text{Sepal.Width} \approx 1.04 + 0.61 \cdot \text{Sepal.Length} + 0.56 \cdot \text{Petal.Width} - 0.58 \cdot \text{Petal.Length}$$

The next column reflects the statistical (standard) error of the coefficients, the next column is a statistic relating to the parameter, which is utilized for the test in the last column (t value is used to compute the p-value in the last column).

Stars to the right of each row reflect significant coefficients (statistically we can say that the are non-zeros, with a 95% confidence interval).

In the buttom we can see the multiple R-sqared and Adjusted R-squared. The range in 0-1 and values closed to 1 reflect a stronger linear relationship.

The RSE (residual standard error) in the buttom of the summary is given by:

$$\text{RSE} = \sqrt{\frac{\text{RSS}}{n-2}} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}$$

The formula for R^2 by:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \bar{y}_i\right)^2} = 1 - \frac{\text{RSS}}{\text{TSS}}$$

We will not dive into the theory of linear regression further, but to the interested, you can read:

- Gareth J., Witten D., Hastie T., and Tibshirani R., An Introduction to Statistical Learning with Applications in R, Springer, 7th printing, 2017. Online access: www.statlearning.com, (feteched October 2018).

In practice, a linear regression model can be used to predict a continuous outcome, but in some cases can also be used for classification or ordinal (ordered factor) outcome.

Category variables (factors) in linear regression

In the last example we discussed continuous variables, but sometimes we want to incorporate factors. For example how gender or education level (both are factors) influence income (continuous). In the case of gender, we can use:

$$\text{Salary} = \beta_0 + \beta_\text{f} \cdot X_\text{female}$$

The nominal value will be β_0 and an addition (or subtraction) of β_f will apply if the observation is female.

Quiz

1. How would we deal with...?
    a. An ordinal variable? (e.g. income levels)
    b. A factor? (e.g. family status)

Generalization to a non-linear model

To generalize to a non-linear form while remaining in the same additive framework, we can transform some of the variables. In the next example we incorporate the interaction of Petal.Length and Petal.Width.

```
iris_nonlm <- lm(data = iris %>% mutate(sqaured.Length = Sepal.Length^2),
                 formula =
                   Sepal.Width ~
                   Sepal.Length + Petal.Length + Petal.Width +
                   sqaured.Length + Petal.Length*Petal.Width +
                   factor(Species))
summary(iris_nonlm)
```

```
##
## Call:
## lm(formula = Sepal.Width ~ Sepal.Length + Petal.Length + Petal.Width +
##     sqaured.Length + Petal.Length * Petal.Width + factor(Species),
##     data = iris %>% mutate(sqaured.Length = Sepal.Length^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.85025 -0.15336 -0.00016  0.15714  0.77920
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -2.55343    1.27616  -2.001 0.047312 *
## Sepal.Length                1.82593    0.45677   3.997 0.000103 ***
## Petal.Length               -0.12474    0.12068  -1.034 0.303026
## Petal.Width                 0.39122    0.34471   1.135 0.258321
## sqaured.Length             -0.12249    0.03834  -3.194 0.001727 **
## factor(Species)versicolor -1.33290    0.30149  -4.421 1.94e-05 ***
## factor(Species)virginica  -1.58903    0.34753  -4.572 1.04e-05 ***
## Petal.Length:Petal.Width   0.03115    0.06563   0.475 0.635800
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2589 on 142 degrees of freedom
## Multiple R-squared:  0.6638, Adjusted R-squared:  0.6473
## F-statistic: 40.06 on 7 and 142 DF,  p-value: < 2.2e-16
```

Pitfalls of linear regression - outliers

- Heteroschedastity
- Correlations between variables makes you see misleading things
- Outliers

```
iris_setosa_outlier <- iris_setosa %>%
  add_row(Sepal.Length = 5.1, Sepal.Width = 35, Petal.Length = 1.4, Petal.Width = 0.2, Species = "setosa

ggplot(iris_setosa_outlier, aes(x = Sepal.Length, y = Sepal.Width)) +
  geom_point() + stat_smooth(method = "lm", se = FALSE,
                             linetype = "dashed") +
  stat_smooth(inherit.aes = FALSE,
              method = "lm",
              data = iris_setosa, aes(x = Sepal.Length, y = Sepal.Width),
              se = FALSE) +
  coord_cartesian(ylim = c(2, 4.5)) +
  ggtitle("The influence of a single outlier on the regression model\nChart is zoomed-in, i.e., the outl
```