Week 11: Correlation and regression with paired data

Stat 201: Statistics I

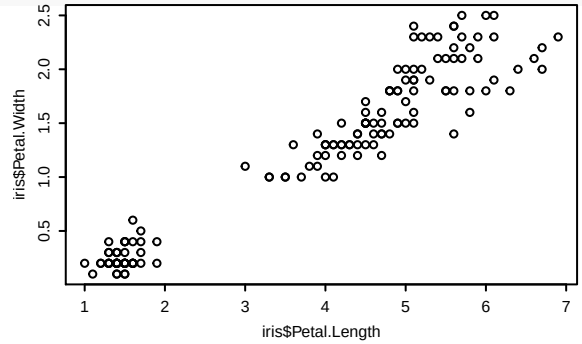
April 21, 2019

Plotting paired data

Paired data is often first examined with a scatterplot. This can be easily done by passing to the `plot` function a pair of equal length vectors.

- Plot the relationship between petal length and petal width in the built-in dataset `iris`.

```
plot(iris$Petal.Length, iris$Petal.Width)
```



Correlation

Correlation between two numeric vectors can be found with the `cor` function.

- Find the correlation between petal length and petal width.

```
cor(iris$Petal.Length, iris$Petal.Width)
```

```
## [1] 0.9628654
```

Correlation tests

A test of the population correlation ρ can be conducted with the `cor.test` function.

- Test whether the population correlation between petal length and petal width is zero or not.

```
cor.test(iris$Petal.Length, iris$Petal.Width)
```

```
##
##  Pearson's product-moment correlation
##
## data:  iris$Petal.Length and iris$Petal.Width
## t = 43.387, df = 148, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9490525 0.9729853
## sample estimates:
##       cor
## 0.9628654
```

Linear models

Regression analysis is conducted by defining a model with the `lm` ("linear model") function. A model is defined by a **formula** in the form:

$$y \sim x$$

where y is a vector of values for the response variable and x is an equal length vector of values of the predictor variable. Note the use of a tilde ("~").

- Define a linear model of the relationship between petal length and petal width, with petal width as the response variable.

```
petal.lm <- lm(iris$Petal.Width ~ iris$Petal.Length)
```

or

```
petal.lm <- lm(Petal.Width ~ Petal.Length, data=iris)
```

Regression coefficients

The estimated regression coefficients are found in the output of the `lm` function.

- Find the estimated regression equation for the relationship between petal length and petal width.

```
petal.lm <- lm(Petal.Width ~ Petal.Length, data=iris)
petal.lm
```

```
##
## Call:
## lm(formula = Petal.Width ~ Petal.Length, data = iris)
##
## Coefficients:
##  (Intercept)  Petal.Length
##      -0.3631        0.4158
```

  - The regression equation is: $\hat{y} = -0.3631 + 0.4158x$

More regression results

Often, we want more information about a regression model than just the equation. By passing the linear model to the `summary` function, we get fuller results.

- The first part of the summary output lists the formula used to define the model and information about residuals (errors) which we do not concern ourselves with in theis class.

```
summary(petal.lm)
```

```
##
## Call:
## lm(formula = Petal.Width ~ Petal.Length, data = iris)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.56515 -0.12358 -0.01898  0.13288  0.64272
```

More regression results, cont.

- The second part of the summary output gives information about the estimated coeffiecients, including the coefficients themselves and the t-score and p-value for hypothesis tests on whether the parameters are zero or not.

```
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.363076   0.039762  -9.131  4.7e-16 ***
## Petal.Length  0.415755   0.009582  43.387  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

More regression results, cont.

- The final part of the summary output has information about the overall fit of the regression model. In particular, the coefficient of determination R^2 is designated by `Multiple R-squared`.

```
##
## Residual standard error: 0.2065 on 148 degrees of freedom
## Multiple R-squared:  0.9271, Adjusted R-squared:  0.9266
## F-statistic:  1882 on 1 and 148 DF,  p-value: < 2.2e-16
```

Adding regression line to scatterplot

A regression line can be added to a scatterplot by passing the linear model to the function `abline`. Note: `abline` can only be used after a `plot` function call.

- Add regression line to iris petal plot.

```
plot(iris$Petal.Length, iris$Petal.Width)
abline(petal.lm)
```