R Lab #11b- Linear regression

Contents

Lion's nose example

Chapter 17 starts with an example study of the association between the phenotype of a lion's nose and its age. The theory is that as a lion ages the proportion of its nose that is black increases. Thus, if you see a lion with a predominantly black nose you can predict that it is old. If this model is true then trophy hunters could preferentially cull old males from the population, which would reduce the impact of hunting on the community.

First import the data set:

```
lionData <- read.csv("chap17e1LionNoses.csv", stringsAsFactors = T)
str(lionData)
## 'data.frame':    32 obs. of  2 variables:
##  $ proportionBlack: num  0.21 0.14 0.11 0.13 0.12 0.13 0.12 0.18 0.23 0.22 ...
##  $ ageInYears     : num  1.1 1.5 1.9 2.2 2.6 3.2 3.2 2.9 2.4 2.1 ...
```

Notice that the data frame has two variables (proportionBlack and ageInYears), and both are numerical vectors.

Explore data with scatter plot

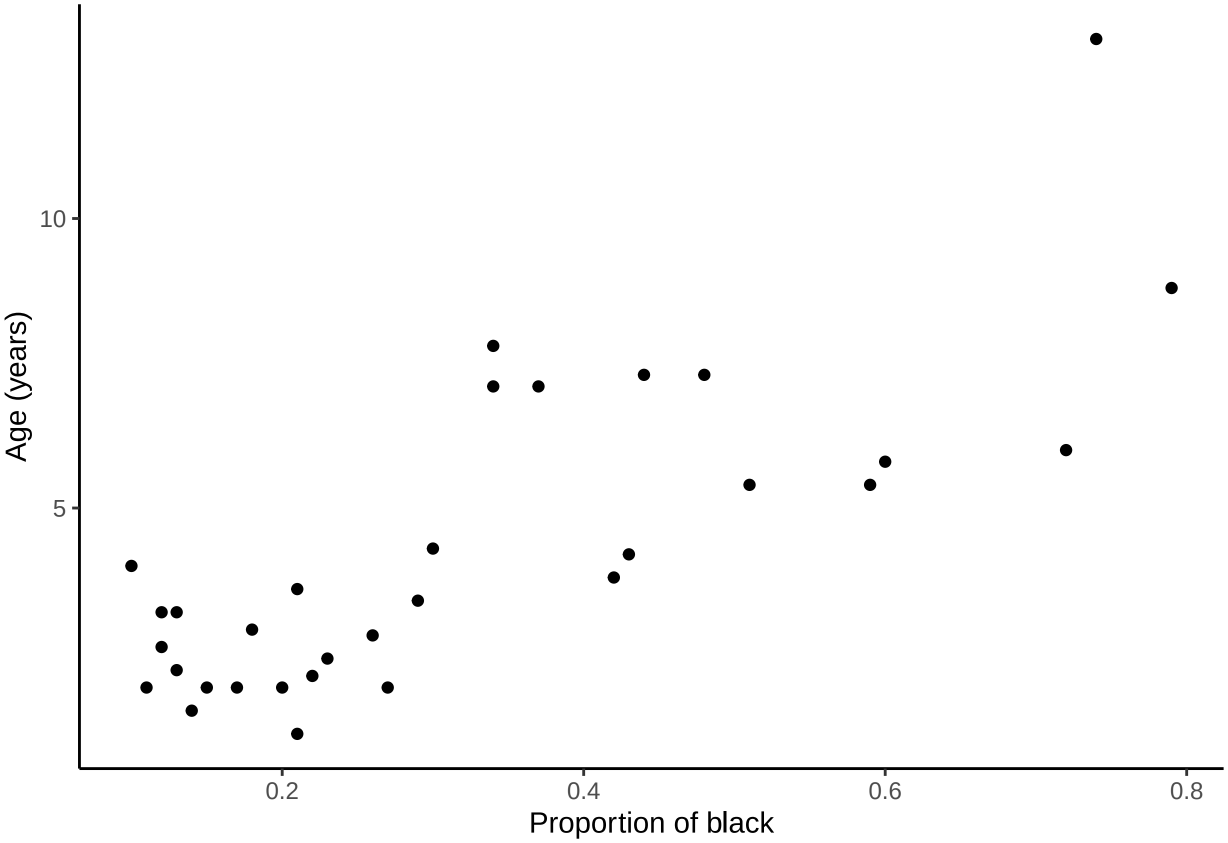
When conducting a correlation or regression analysis you should first visualize the data with an X-Y scatter plot.

The first step is to differentiate the explanatory (x-axis) and response (y-axis) variables. Here the design of the study is to use the proportion of black to predict age. Thus, proportion of black is the explanatory variable and age is the response variable.

Use ggplot() and geom_point() to generate a plot. We'll also add the xlab() and ylab() functions to make proper axis labels.

Note that we do **NOT** have to lower the axis limits to zero. In a *t*-test or ANOVA framework you are trying to visualize differences among groups, and thus need the y-axis to go to zero (or the lowest biologically meaningful number) so that the differences are not visually exaggerated. In a correlation or regression we are looking for *an association* between variables rather than differences, and thus can use the default axis limits.

```
library(ggplot2)
ggplot(lionData, aes(x = proportionBlack, y = ageInYears)) +
    geom_point() + xlab("Proportion of black") + ylab("Age (years)") +
    theme_classic()
```



Qualitative check of assumptions using scatter plot

A regression analysis assumes that at each value of X there is a population of possible Y-values with a normal distribution whose mean lies on the true regression line.

You should thus inspect the plot for outliers or a nonlinear relationship between X and Y.

Here, the plot shows a positive correlation between the variables that is generally linear. There doesn't appear to be any major outliers.

Linear regression model

Now we can set up a least squares linear regression model. This will take the form of $Y = \mathrm{a} + \mathrm{b}X$, where a is the y-intercept and b is the slope of the least squares line.

This is done in R using the function lm(). It is best to save the model to a variable name and use the syntax to use is lm(response~explanatory,data=dataframe). Thus:

```
lionRegression <- lm(ageInYears ~ proportionBlack, data = lionData)
```

Now the model is saved in memory as lionRegression. To see the coefficients we can simply call lionRegression.

```
lionRegression
##
```

```
## Call:
## lm(formula = ageInYears ~ proportionBlack, data = lionData)
##
## Coefficients:
##     (Intercept)  proportionBlack
##           0.879           10.647
```

Here we can see that the slope of the least squares line is 10.647. This means that for every increase of 1 unit in the X variable there is an increase of 10.647 units of the Y variable. Here, the X variable can only range from 0 to 1. So another way to interpret the slope is that for every increase of 0.1 proportion of black there is a 1.0647 increase in years.

We see that the Y-intercept is 0.879. This is the mean value of Y when X is zero. Here we can interpret this as lions with no black on their nose being on average 0.879 years old.

Hypothesis tests of coefficients

The coefficients from the model are a=0.879 and b=10.647. How else can we analyze these values? We can test two null hypotheses.

First, we can test the null hypothesis that a=0. That is, that when X=0 the mean predicted value of Y is also 0.

Second, we can test the null hypothesis that the b=0. This is generally the test of interest, as it is used to test if there is a significant correlation between X and Y. Even though the slope seems much higher than 0, there is uncertainty in the estimate (note that the larger the residuals the greater the uncertainty).

Output for these tests are shown simply by using the summary() function with the regression as the argument.

```
summary(lionRegression)
##
## Call:
## lm(formula = ageInYears ~ proportionBlack, data = lionData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.5449 -1.1117 -0.5285  0.9635  4.3421
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.8790     0.5688   1.545    0.133
## proportionBlack  10.6471     1.5095   7.053 7.68e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.669 on 30 degrees of freedom
## Multiple R-squared:  0.6238, Adjusted R-squared:  0.6113
## F-statistic: 49.75 on 1 and 30 DF,  p-value: 7.677e-08
```

In both cases a t-test is used to test the null. The first row gives the results for the test of a=0. The P-value of 0.133, and thus we **fail to reject** the null.

The second row (proportionBlack) gives the results for the test of b=0. The P-value of 7.68e-08 is well below 0.05 and thus we can **reject the null**. This means that there is a significant association between proportion black on nose and age. Furthermore, since the slope is positive the correlation between these variables is positive (i.e., as proportion black increases age increases). Note that this is the same P-value that you would get with cor.test().

R^2 of regression

The R^2 statistic in a linear regression calculates the fraction of the variation in Y that is “explained” by X. This is a measure of the strength of the correlation. The values range from 0 (no variation explained; true cloud of points) to 1 (all variation explained; all points would be on a straight line).

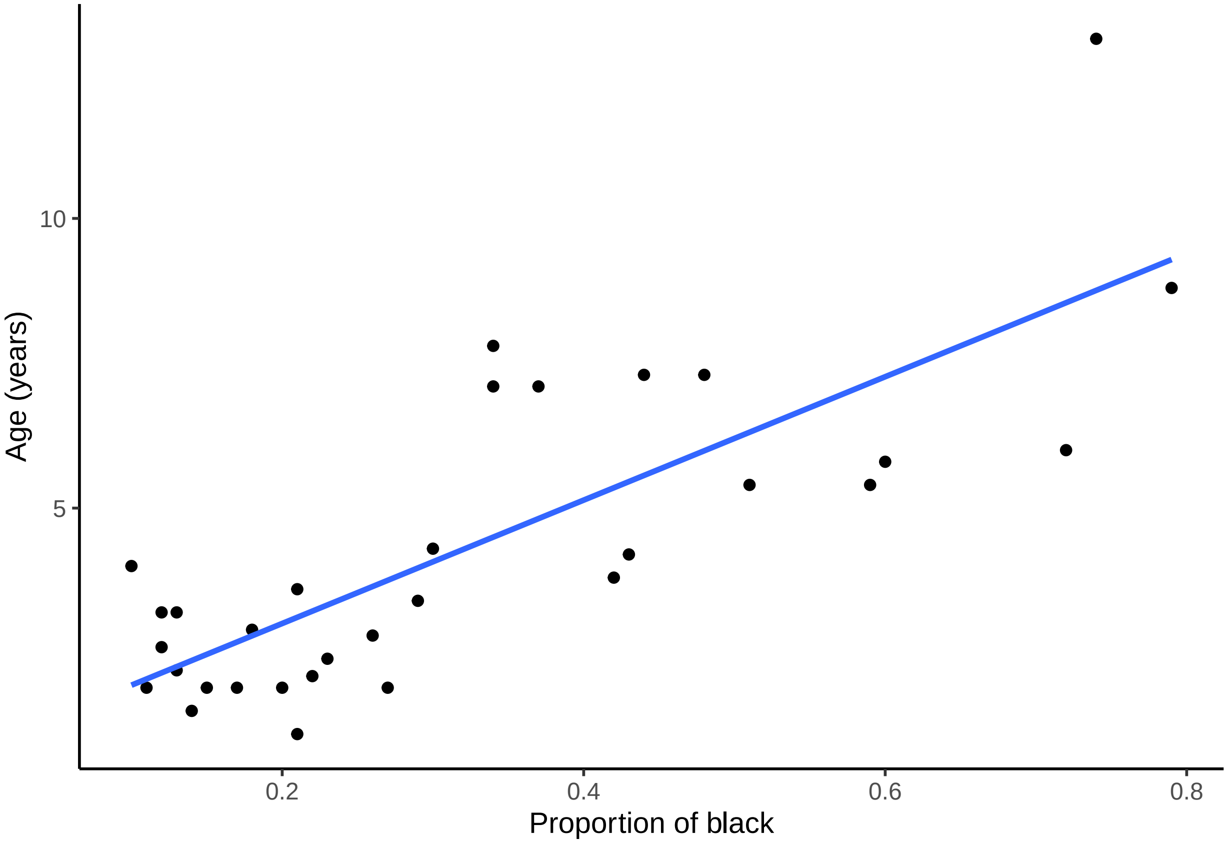
The R^2 is also reported using summary(). Note that two calculations are shown, but the one using the formula from the textbook is labeled “Multiple R-squared.”

Here the R^2 of 0.6238 means that 62.38% of the variance in age can be explained by the proportion of black on the nose.

Adding regression line to plot

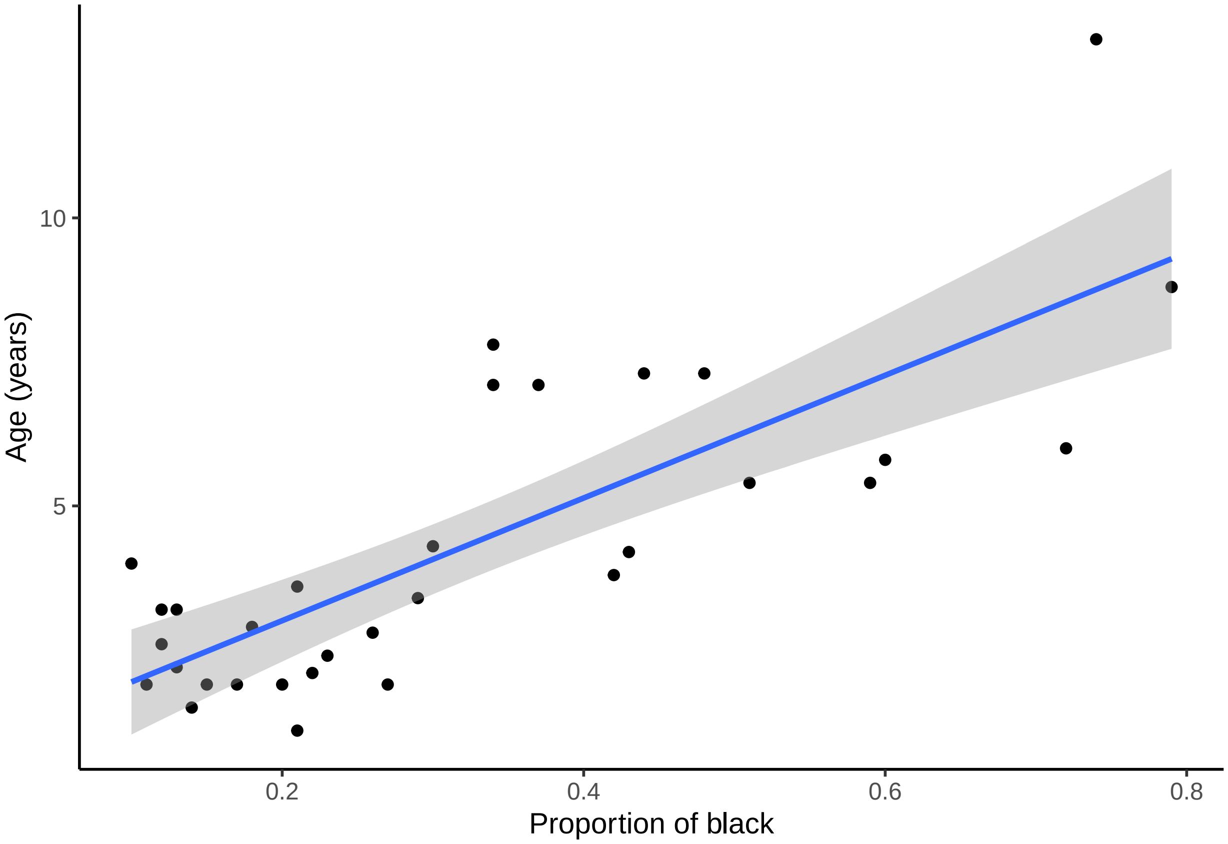
Now we can go back to the scatter plot and add the least squares line. This can be done by adding the geom_smooth() function to the plot command above. By using the method=“lm” argument a least squares line will be added. If you only want the line then also add the argument se=F.

```
ggplot(lionData, aes(x = proportionBlack, y = ageInYears)) +
    geom_point() + xlab("Proportion of black") + ylab("Age (years)") +
    theme_classic() + geom_smooth(method = "lm", se = F)
## `geom_smooth()` using formula = 'y ~ x'
```



The “se” argument specifies whether or not you want to show the confidence bands for predictions. That is, the 95% confidence intervals for predictions of mean Y for any value of X. The default se is TRUE, so you can either remove the argument or switch the F to T.

```
ggplot(lionData, aes(x = proportionBlack, y = ageInYears)) +
    geom_point() + xlab("Proportion of black") + ylab("Age (years)") +
    theme_classic() + geom_smooth(method = "lm")
## `geom_smooth()` using formula = 'y ~ x'
```



Predicting Y-values

Now that we have fit a least squares regression model, it can be used to predict values. That is, for any value of X we can predict the mean value of Y using the model formula $Y = \mathrm{a} + \mathrm{b}X$.

This can be done in R with the function predict(), which takes as arguments the model and the X-value of interest, but the latter has a specific syntax. Note that the predict() function is going to be more precise then calculating it “by hand” using R output because there is no rounding involved. For example, if we want to predict the mean age for a proportion of black of exactly 0.7 then we use this command:

```
predict(lionRegression, data.frame(proportionBlack = 0.7))
##       1
## 8.33199
```

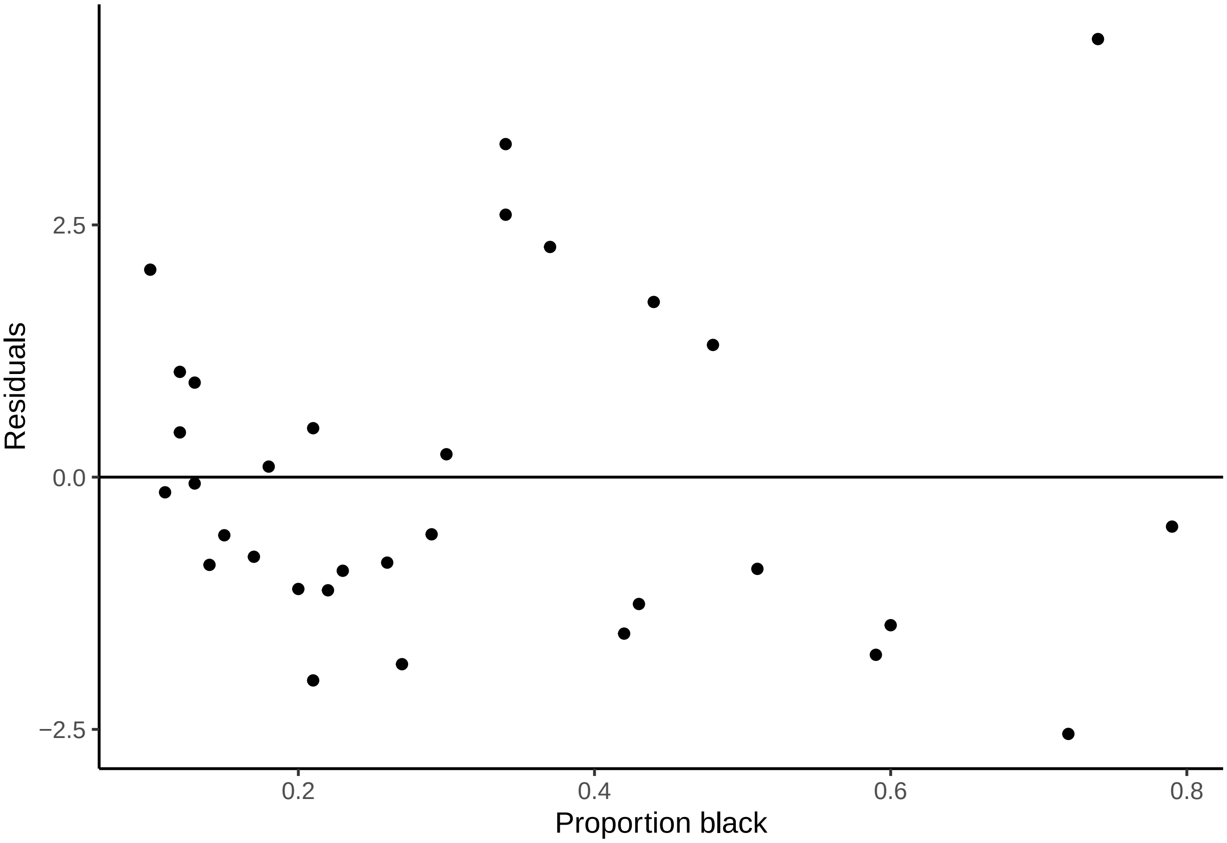
Qualitative check of assumptions using plot of residuals

Another assumption of a linear regression is that the variance of Y-values is the same at all values of X.

If this assumption is met, then a plot of the residuals (i.e., vertical distances between each point and the least squares line) should be a roughly symmetric cloud of points above and below the horizontal line at zero, with a higher density of points close to the line than away from the line and little noticeable curvature as we move from left to right.

The residuals are stored in the linear regression model (lionRegression). Typically, ggplot needs a data frame to work with, but it is also possible to give ggplot a linear model. However, instead of the x and y asthetics being variable names they will be attributes of the model. In an R the residuals are saved as .resid. So you can plot the residuals with the following command. Note that a horizontal line at Y=0 is added because if the data meet assumptions then we expect a cloud of Y points about equally spread above and below zero.

```
ggplot(lionRegression, aes(x = proportionBlack, y = .resid)) +
    geom_point() + xlab("Proportion black") + ylab("Residuals") +
    geom_hline(yintercept = 0) + theme_classic()
```



Assessing these plots is qualitative. Here the residuals are spread a bit more widely to the right compared to the left, but there is not a terrible "funnel" pattern.

R commands summary

- **Linear model**
  - lm(y~x,data=dataframe)
- **Add least squares line to ggplot**
  - geom_smooth(method="lm",se=F)
- **Add least squares line and confidence bands to ggplot**
  - geom_smooth(method="lm")
- **Add horizontal line to ggplot**
  - geom_hline(yintercept = Y)