Model answer for assignment 3

Luis A. Apiolaza

11 August 2024

## Question 1

**As always, create a new folder for your work and also a new RStudio project in that folder. Download the aquatic\_toxicity.xlsx files in your folder.**

## Question 2

**Use the function read\_excel(), from the readxl package, to read the file and name the object toxic.**

library(tidyverse)

## Warning: package 'ggplot2' was built under R version 4.3.1

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.2 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.5.0 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.1   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(readxl)  
library(performance)  
library(corrplot)

## corrplot 0.92 loaded

library(GGally)

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

# I don't like the default theme. This assignment I am using  
theme\_set(theme\_minimal())  
  
toxic <- read\_xlsx("data/aquatic\_toxicity.xlsx")

## Question 3

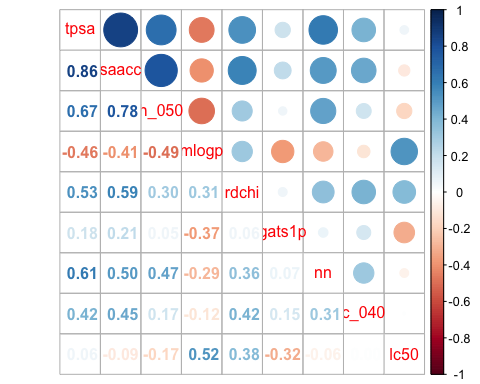
**You will use random selection of 500 rows of the dataset, which depends on your student ID. Name your sample my\_toxic.**

set.seed(20240911) # Replace this seed by your student ID  
  
my\_toxic <- toxic |>   
 sample\_n(500)

## Question 4 (1 point)

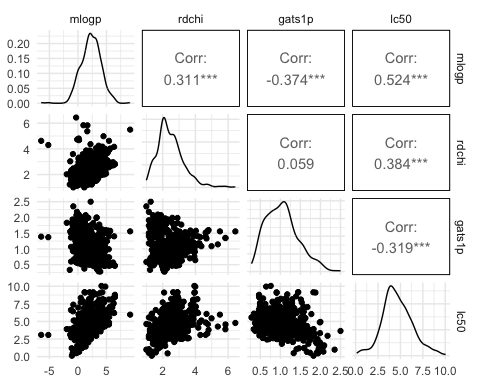
**Estimate the correlations between all variables (using my\_toxic %>% cor()) and, based on the correlations, choose three variables to predict lc50. Using ggpairs (from the GGally package) create a scatterplot matrix with your three predictors and lc50. Explain in 50 words the relationships you observe in those plots.**

# my\_toxic %>% cor() or  
my\_toxic |> cor() |> corrplot.mixed()



my\_toxic |> ggpairs(columns = c("mlogp", "rdchi", "gats1p", "lc50"))

## Warning in geom\_point(): All aesthetics have length 1, but the data has 16 rows.  
## ℹ Did you mean to use `annotate()`?



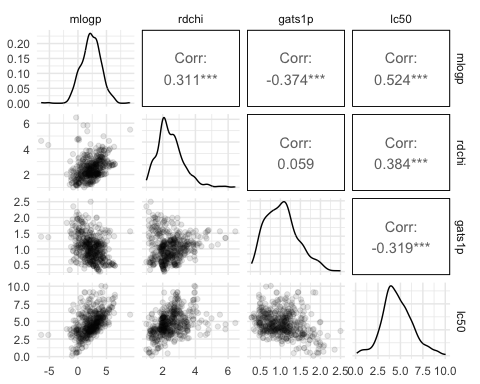
**Background:** We expect predictors to be correlated with the response. In my sample, the predictors with the highest correlations (in magnitude, independent of sign) with the response were: mlogp, rdchi, and gats1p. Based on that, I created scatterplots including those predictors and the response variable.

There is a moderately positive relationship between mlogp and lc50, and between rdchi and lc50. There is also a moderately negative relationship between gats1p and lc50. The relationship between lc50 and rdchi shows some deviation from linearity. **Note:** the correlations shown in the upper triangle of the plot can help you figure out the strength of the association.

**More advanced:** It”s possible to change some of the parameters of the plot, to reduce overplotting in the scatterplots, helping interpretation. How does this work? We tell R to increase transparency for the points, only to the plots that deal with continuous variables in the lower triangle of plot of ggpairs().

my\_toxic |>   
 ggpairs(columns = c("mlogp", "rdchi", "gats1p", "lc50"),  
 lower = list(continuous = wrap("points", alpha = 0.1)))

## Warning in geom\_point(): All aesthetics have length 1, but the data has 16 rows.  
## ℹ Did you mean to use `annotate()`?



## Question 5 (0.5 point)

**Fit the model with your chosen predictors (call it m1) and write down the coefficients, the adjusted-r2 and residual standard error.**

m1 <- lm(lc50 ~ mlogp + rdchi + gats1p,   
 data = my\_toxic)  
summary(m1)

##   
## Call:  
## lm(formula = lc50 ~ mlogp + rdchi + gats1p, data = my\_toxic)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.6251 -0.9680 -0.1939 0.6096 5.7214   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.28865 0.25819 12.737 < 2e-16 \*\*\*  
## mlogp 0.34751 0.03988 8.714 < 2e-16 \*\*\*  
## rdchi 0.58692 0.07948 7.385 6.52e-13 \*\*\*  
## gats1p -0.83647 0.16452 -5.084 5.24e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.341 on 496 degrees of freedom  
## Multiple R-squared: 0.3625, Adjusted R-squared: 0.3586   
## F-statistic: 94.02 on 3 and 496 DF, p-value: < 2.2e-16

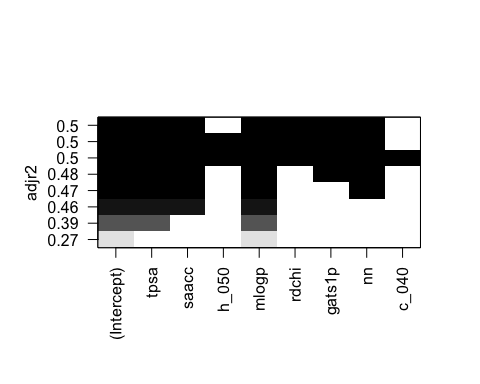
The model coefficients (intercept and slopes) are: 3.289, 0.348, 0.587, -0.836.

The model has a residual standard error of 1.341 and adjusted-R2 of 0.359.

## Question 6 (0.5 point)

**Now use the leaps package to fit all regression subsets. Plot the results of regression subsets, and explain which predictors are contained in the best model for each number of predictors.**

library(leaps)  
all\_mods <- regsubsets(lc50 ~ ., data = my\_toxic)  
plot(all\_mods, scale = "adjr2")



**Note:** you could have also used regsubsets(lc50 ~ tpsa + saacc + h\_050 + all the other predictors, data = my\_toxic), but that”s a lot of typing.

The best model with one predictor uses mlogp, with 2 predictors uses tpsa and mlogp, with 3 predictors uses tpsa, saacc and mlogp, with 4 predictors uses tpsa, saacc, mlogp and nn, etc.

## Question 7 (1 point)

**Now fit the best model (call it m2) and compare its adjusted-r2 and residual standard error with m1. Discuss in 50 words the similarities and differences between the results of the 2 models.**

m2 <- lm(lc50 ~ tpsa + saacc + mlogp + rdchi + gats1p + nn,  
 data = my\_toxic)  
  
summary(m2)

##   
## Call:  
## lm(formula = lc50 ~ tpsa + saacc + mlogp + rdchi + gats1p + nn,   
## data = my\_toxic)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.7792 -0.7600 -0.1029 0.5865 4.9888   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.789470 0.235920 11.824 < 2e-16 \*\*\*  
## tpsa 0.028804 0.002664 10.812 < 2e-16 \*\*\*  
## saacc -0.015434 0.001715 -9.000 < 2e-16 \*\*\*  
## mlogp 0.431210 0.062212 6.931 1.31e-11 \*\*\*  
## rdchi 0.533073 0.138603 3.846 0.000136 \*\*\*  
## gats1p -0.675823 0.152074 -4.444 1.09e-05 \*\*\*  
## nn -0.221816 0.047985 -4.623 4.85e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.189 on 493 degrees of freedom  
## Multiple R-squared: 0.502, Adjusted R-squared: 0.4959   
## F-statistic: 82.82 on 6 and 493 DF, p-value: < 2.2e-16

The best model is the one with the highest adjusted-R2 (at the top of the plot), and also darker squares for the predictors. In my sample, the best model contains tpsa, saacc, mlogp, rdchi, gats1p and nn. The best model for your sample may have a slightly different number of predictors.

Model m2 is a moderate improvement compared to m1, with a slight reduction of residual standard error (now 1.189 and increase of adjusted-R2 of 0.496. We are now explaining around 14% more of the total variability compared to m1.

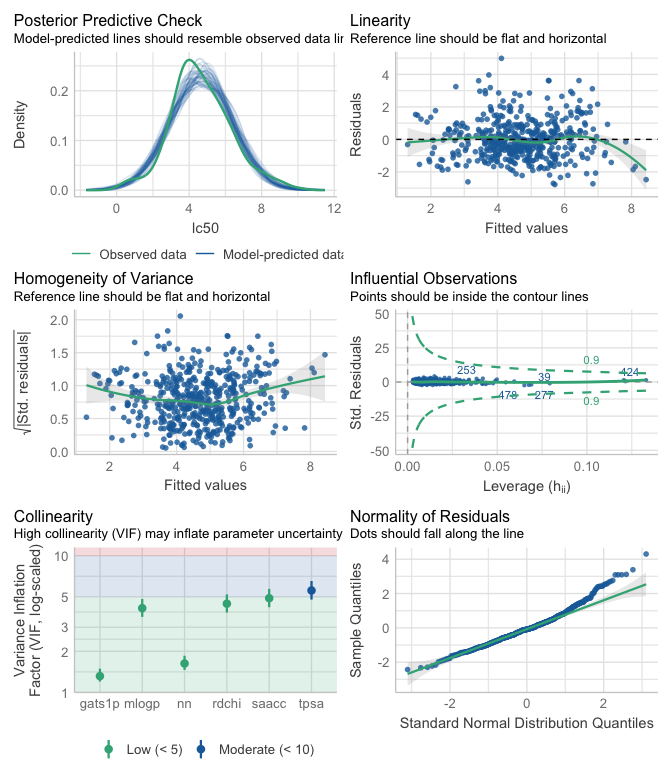
The magnitud of the slopes for the predictors in model m1 may have changed slightly to account for the fact that we now have more predictors doing the job.

**Note:** while the relationship between some predictors and lc50 is weak, they still have a small contribution to prediction.

## Question 8 (0.5 point)

**Create diagnostic plots for the residuals of the best model; use check\_model(m2). Check the model for assumptions for the residuals. Explain in no more than 70 words if there is anything unusual or wrong.**

check\_model(m2)



There is no evidence for a deviation from Linearity (Residuals vs Fitted), and there is an slight increase of residual variance for larger predicted values (Homogeneity of variance). There is a small deviation from normality, mostly driven by the highest lc50 observations (Normality Q-Q). There are no highly Influencial observations (Residuals vs Leverage, passing the lines for 0.5 Cook”s Distance).

## Question 9 (0.5 point)

**Obtain the predicted lc50 (95% prediction and 95% confidence intervals) for new observations with the following characteristics:**

tpsa saacc h\_050 mlogp rdchi gats1p nn c\_040 9.23 11 0 2.27 2.15 1.75 0 0 0 0 0 3.37 2.08 1.20 0 0

This requires:

1. Creating a table with one row and all the predictors for each observation
2. Using predict with the coefficients of m2 and the new data.

new\_obs <- tibble(tpsa = c(69.97, 3.24),   
 saacc = c(97.43, 3.12),   
 h\_050 = c(0, 0),  
 mlogp = c(3.12, 9.15),  
 rdchi = c(3.72, 5.49),  
 gats1p = c(1.26, 1.56),  
 nn = c(0, 1),  
 c\_040 = c(2, 0))  
  
predict(m2, newdata = new\_obs, interval = "confidence")

## fit lwr upr  
## 1 5.777978 5.524025 6.031932  
## 2 8.430685 7.824111 9.037258

predict(m2, newdata = new\_obs, interval = "prediction")

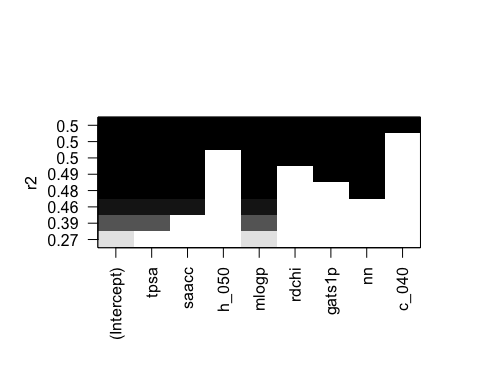
## fit lwr upr  
## 1 5.777978 3.427724 8.128232  
## 2 8.430685 6.016739 10.844630

**Note:** Notice that the confidence interval—for where the **average response** is located—is narrower than the prediction interval—for where **any response** is located.

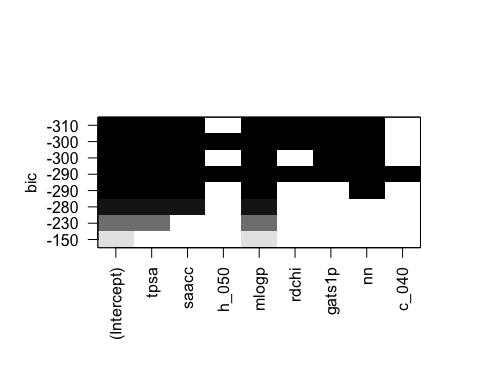
## Question 10 (0.5 point)

**Let’s have a look again at all\_mods (from question 6). Plot it again using scale = "r2" and scale = "bic". Compare the 3 plots and explain how/why they differ in 50 words.**

plot(all\_mods, scale = "r2")



plot(all\_mods, scale = "bic")



The main point to understand here is that R2 **does not** penalize for number of predictors (unlike adj-R2 and BIC) causing that the “best” model will always have all predictors. Adjusted- and BIC apply different trade offs, so they may choose different *best* models.

## Question 11 (0.5 point)

**We will revisit the formula $(X` X)^{-1} X`y$ to obtain the vector of regression coefficients. Use R’s matrix algebra capabilities to reproduce the coefficients estimated in m2.**

X <- model.matrix(m2)  
y <- my\_toxic %>% pull(lc50)  
  
b <- solve(t(X) %\*% X) %\*% t(X) %\*% y  
b

## [,1]  
## (Intercept) 2.78947033  
## tpsa 0.02880373  
## saacc -0.01543426  
## mlogp 0.43121018  
## rdchi 0.53307320  
## gats1p -0.67582303  
## nn -0.22181585

This should produce the same results as m2. Some variations in code:

* I am using pull(name) to extract a single column as a vector. This is tidyverse for mytoxic$lc50.
* One could produce the X matrix by selecting the predictors, convert them to a matrix and cbind() a columns of 1s before all the predictors.
* One could surround (b <- solve(t(X) %\*% X) %\*% t(X) %\*% y) with parentheses to print the contents of b without typing the final b.