Lab 15: Prediction in SLR; practice problems

STAT218

This activity has three objectives:

1. Review basic SLR analysis
2. Demonstrate how to compute and visualize confidence intervals for the mean response and prediction intervals
3. Provide an opportunity to practice using SLR

## SLR Review

Previously, we covered how to do the following:

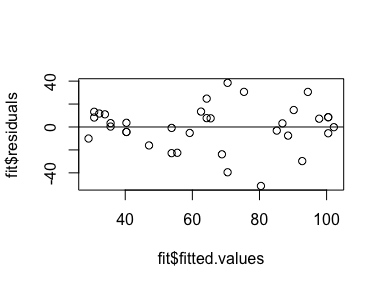
* fit a model in R
* check residual diagnostics
* perform inference for model parameters

These steps are illustrated below

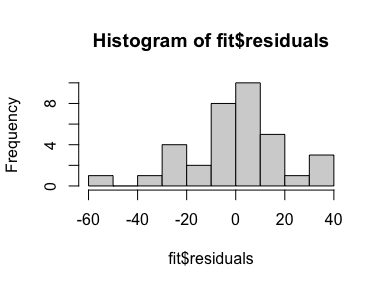
# fit model and check summary  
fit <- lm(RFFT ~ Age, data = prevend)  
summary(fit)

Call:  
lm(formula = RFFT ~ Age, data = prevend)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-51.437 -6.502 3.162 9.785 38.501   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 162.9731 13.7266 11.87 1.86e-13 \*\*\*  
Age -1.6844 0.2295 -7.34 1.99e-08 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 19.4 on 33 degrees of freedom  
Multiple R-squared: 0.6201, Adjusted R-squared: 0.6086   
F-statistic: 53.87 on 1 and 33 DF, p-value: 1.995e-08

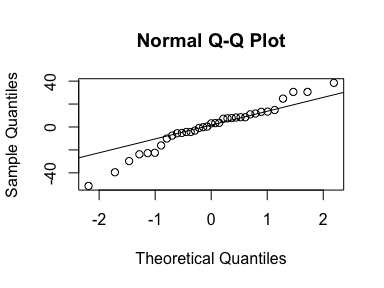
# residual diagnostics: residual-fit plot  
plot(fit$fitted.values, fit$residuals)  
abline(h = 0)



# residual diagnostics: residual histogram  
hist(fit$residuals)



# residual diagnostics: qq plot  
qqnorm(fit$residuals)  
qqline(fit$residuals)



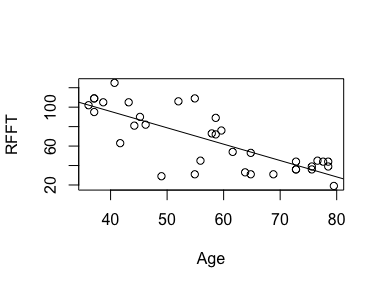
# significance tests  
summary(fit)$coef

Estimate Std. Error t value Pr(>|t|)  
(Intercept) 162.973062 13.7265565 11.872829 1.862539e-13  
Age -1.684404 0.2294928 -7.339681 1.994774e-08

# confidence intervals for parameter estimates  
confint(fit, level = 0.95)

2.5 % 97.5 %  
(Intercept) 135.046173 190.899951  
Age -2.151311 -1.217498

# model visualization  
coefs <- coef(fit)  
plot(prevend)  
abline(a = coefs[1], b = coefs[2])



We would interpret the output as indicating:

* age is significantly associated with mean RFFT (*p = 0.0000000199*)
* with 95% confidence, each year of age is associated with a decline in mean RFFT estimated to be between 1.22 and 2.15 points
* age explains 62% of variability in RFFT scores ( = 0.6201)

## Predictions and interval estimates

While predictions *could* be computed manually from the fitted model equation, the predict() function will do the job given only the fitted model object and new data, formatted as a data frame. To add interval estimates, simply include either:

* interval = 'confidence' for an interval for the mean response
* interval = 'prediction' for an interval for a specific response

For example, to predict the RFFT score for a 55 year old:

# point prediction  
predict(fit, newdata = data.frame(Age = 55))

1   
70.33082

# prediction interval for an observation  
predict(fit, newdata = data.frame(Age = 55), interval = 'prediction')

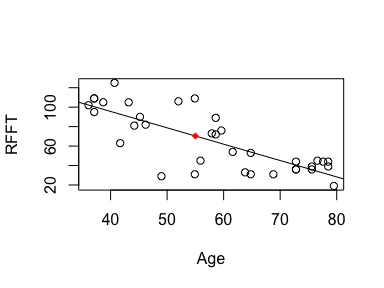
fit lwr upr  
1 70.33082 30.26724 110.3944

# confidence interval for the mean  
predict(fit, newdata = data.frame(Age = 55), interval = 'confidence')

fit lwr upr  
1 70.33082 63.50466 77.15698

To visualize a specific prediction, one can simply add the prediction as a point on the plot:

# store prediction  
pred <- predict(fit, newdata = data.frame(Age = 55))  
  
# plot data, add line, add point  
plot(prevend)  
abline(a = coefs[1], b = coefs[2])  
points(x = 55, y = pred, pch = 18, col = 'red')

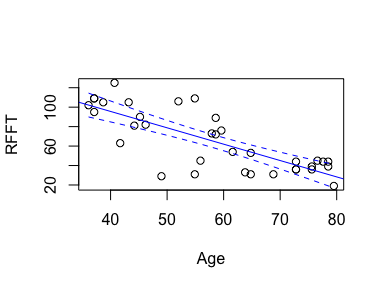


Visualizing intervals is done by:

1. generate a grid of evenly-spaced points along the range of the explanatory variable
2. compute predictions with uncertainty quantification for each point on the grid
3. plotting a path through the upper and lower bounds

For example, to include 95% confidence intervals in the plot:

# generate a grid of 100 points spanning the range of ages  
x <- seq\_range(prevend$Age, n = 100)  
  
# compute predictions  
preds <- predict(fit, newdata = data.frame(Age = x),   
 interval = 'confidence', level = 0.95)  
  
# add to plot  
plot(prevend)  
abline(a = coefs[1], b = coefs[2], col = 'blue')  
lines(x = x, y = preds[, 2], lty = 2, col = 'blue')  
lines(x = x, y = preds[, 3], lty = 2, col = 'blue')



|  |
| --- |
| Your turn |
| Modify the example above to…   * visualize 99% intervals for the mean * visualize 90% prediction intervals   # generate a grid of 100 points spanning the range of ages x <- seq\_range(prevend$Age, n = 100)  # compute predictions preds <- predict(fit, newdata = data.frame(Age = x),   interval = 'confidence', level = 0.95)  # add to plot plot(prevend) abline(a = coefs[1], b = coefs[2], col = 'blue') lines(x = x, y = preds[, 2], lty = 2, col = 'blue') lines(x = x, y = preds[, 3], lty = 2, col = 'blue') |

## Practice problems

### Crab claws

Estimate the association between claw height and closing force. Predict the closing force for an individual crab with claw height 12.2; provide both point and interval estimates.

# claw data  
head(ex0722)  
  
# fit model of closing force (response) against height (explanatory)  
  
# check diagnostics  
  
# model summary  
  
# interval estimate for association  
  
# prediction for claw height 12.2

### Galapagos Islands

Estimate the relationship between island area and the number of observed species and indicate how well your model fits the data. Provide a 95% confidence interval for the model parameter that captures the relationship and interpret the interval in context.

# galapagos data  
head(ex1220)  
  
# plot total species count against area  
  
# same, but log-transformed  
  
# fit a model of log-total count against log-area  
  
# check diagnostics  
  
# check model summary for quality of fit  
  
# compute and interpret CI for association parameter