Now available at

`http://dicook.github.io/Statistical_Thinking/tutorials/lab07/lab7help.pdf`

1 Question 1

Remember you can use the $(1 - \alpha)\%$ confidence interval for β to make a level α hypothesis test about that β without even looking at the t-critical value or p-value.

2 Question 2

boot needs four arguments to work for us:

- data = aus_ nomiss

- statistic = something that calculates our bootstrap value

- R = number of replications

- weights = aus_ nomiss$SENWGT_ STU

```
library(boot)
calc_stat <- function(d, i) {
  x <- d[i,]
  mod <- Create a GLM using the subset x of our data
  stat <- From that model, extract the parameter of interest
  return(stat)
}
stat <- boot(aus_nomiss, statistic=calc_stat, R=1000,
     weights=aus_nomiss$SENWGT_STU)
stat
sort(stat$t)[25]
sort(stat$t)[975]
```

3   Question 3

We need to make a prediction in this bootstrap. First, set up the X variables
for the prediction:

```
new_data <- data.frame(ST04Q01 = as.factor(1), ST06Q01 = 0, ST15Q01 = as.factor(1),
                       ST19Q01 = as.factor(1), ST26Q01 = as.factor(1),
                       ST26Q04 = as.factor(1), ST26Q06 = as.factor(1),
                       ST27Q02 = 3, ST28Q01 = 3, math_std = 0, SENWGT_STU = 0.1041)
```

Now our calc function has the same basic idea:

```
calc_pred <- function(d, i, newd){
  x <- d[i,]
  mod <- Create a GLM object with our data subset x
  pred <- Use that model and newd to make a prediction
  return(pred)
}
pred <- boot(aus_nomiss, statistic = calc_pred, R = 1000, weights = aus_nomiss$SENWGT_STU, n
```

4   Question 4

A prediction interval takes a prediction $\hat{y}$ then adds the 95% interval from the
residuals. So, make a new bootstrap calc function. It should fit the model on

a subset x of our data, then return the lowest and highest residual from that
model. Afterwards, run the boot function to get a bootstrap for both the lower

and upper residuals. Finally take your fitted value from predict(), and make a
95% interval using both the min and max bootstrap residuals as the lower and
upper bound of the prediction.