Stat 6021 R Tutorial: More with Linear Regression

In the last tutorial, we will look at a few more things we can do in R for linear regression. To start, read the "purity.txt" dataset into R. Suppose we regress *purity* against *hydro*.

```
data<-read.table("purity.txt", header=TRUE ,sep="")
attach(data)
result<-lm(purity~hydro)
```

1. Obtaining quantiles from t distributions. For confidence intervals and hypothesis testing, we can use R to obtain the appropriate quantiles and critical region. To obtain the corresponding multiplier for a 95% confidence interval for β_1, type `qt(0.975, 18)`. The appropriate percentile and degree of freedom need to be provided. This multiplier is also the cut-off for the critical region when conducting a two-sided hypothesis test for β_1. Use `qnorm()` for the normal distribution and `qf()` for the F distribution. Suppose you want to conduct a one-sided hypothesis test for β_1. What would you type?

2. Obtaining p-values. To obtain the p-value for a two-sided hypothesis test for β_1, type `2*(1-pt(3.386,18))`. You need to specify the value of the test statistic and the degree of freedom in the `pt()` function.

3. Confidence interval for β_0 and β_1. The `confint()` function obtains the confidence intervals for the regression parameters. For example

   ```
   confint(result,level = 0.95)
   ```

4. Confidence interval for mean response, prediction interval for the response of a future observation. The `predict.lm()` function helps with obtaining fitted values for given values of the predictor, and the corresponding confidence and prediction intervals. Suppose for *hydro*= 1.2, we want the corresponding CI for mean *purity*. The following code can be used:

   ```
   newdata<-data.frame(hydro=1.2)
   predict.lm(result, newdata, level=0.95, interval="confidence")
   ```

5. Using R help. To access the help that is available in R, type `?predict.lm`. What do you type if you want a prediction interval instead?

6. Extracting components from `lm()`. Using `summary(result)` produces quite a bit of information. Sometimes, you may want to extract some of these results and store them into your own defined variables. What appears when you type `names(result)`? What appears when you type `result$residuals`?