Inference with R: Part One

```
> # Inference with R, part One
> sat =
read.table("http://www.utstat.utoronto.ca/~brunner/data/legal/openSAT.data.txt")
> head(sat)
  VERBAL MATH  GPA
1    578  567 2.68
2    474  653 2.51
3    546  657 1.95
4    664  686 2.81
5    600  619 2.79
6    488  738 2.36
> lm(GPA ~ VERBAL+MATH, data=sat)

Call:
lm(formula = GPA ~ VERBAL + MATH, data = sat)

Coefficients:
(Intercept)       VERBAL         MATH
  0.6080747    0.0023070    0.0009974

> model1 = lm(GPA ~ VERBAL+MATH, data=sat); summary(model1)

Call:
lm(formula = GPA ~ VERBAL + MATH, data = sat)

Residuals:
     Min       1Q   Median       3Q      Max
-1.70296 -0.36750  0.02644  0.38869  1.24830

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.6080747  0.4413074   1.378    0.170
VERBAL      0.0023070  0.0005521   4.178 4.41e-05 ***
MATH        0.0009974  0.0006095   1.636    0.103
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.5484 on 197 degrees of freedom
Multiple R-squared: 0.116,  Adjusted R-squared: 0.107
F-statistic: 12.93 on 2 and 197 DF,  p-value: 5.305e-06
```

```
> summary(lm(GPA~MATH,data=sat)) # Math not controlling for Verbal

Call:
lm(formula = GPA ~ MATH, data = sat)

Residuals:
     Min       1Q   Median       3Q      Max
-1.92390 -0.38854 -0.00325  0.38448  1.43043

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.5272240  0.3981544   3.836 0.000168 ***
MATH        0.0016979  0.0006098   2.784 0.005885 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5707 on 198 degrees of freedom
Multiple R-squared: 0.03768,     Adjusted R-squared: 0.03282
F-statistic: 7.752 on 1 and 198 DF,  p-value: 0.005885

> # model1 is a list of lists -- see help(lm)
> model1$coefficients
 (Intercept)       VERBAL         MATH
0.6080747411 0.0023070007 0.0009973607
> model1$residuals[1:8] # Just the first 8 -- There are 200
          1           2           3           4           5           6           7
8
 0.17297532  0.15713038 -0.57296312 -0.01411267  0.18035855 -0.10994329 -0.65948194
0.22646511
> # Summary object is a list of lists too.
> summary(model1)[1] # First element
$call
lm(formula = GPA ~ VERBAL + MATH, data = sat)

> summary(model1)[6] # 6th element
$sigma
[1] 0.548354

> sqrt( sum((model1$residuals^2))/197 ) # It's the square root of MSE
[1] 0.548354
> summary(model1)[10] # 10th element
$fstatistic
   value    numdf    dendf
 12.9276   2.0000 197.0000

> is.list( summary(model1)[10] )  # It's a list with one element
[1] TRUE
> summary(model1)[[10]] # 10th element as a numeric vector
   value    numdf    dendf
 12.9276   2.0000 197.0000
> summary(model1)[[10]][1] # Just the F statistic – first element
  value
12.9276
```

```
> V = vcov(model1); V    # Estimated cov(betahat) = sigma-squared X'X-inverse
             (Intercept)        VERBAL          MATH
(Intercept)  0.1947522360 -1.214588e-04 -1.861318e-04
VERBAL      -0.0001214588  3.048530e-07 -9.257555e-08
MATH        -0.0001861318 -9.257555e-08  3.714564e-07
> # Get it another way
> model1 = lm(GPA ~ VERBAL+MATH, x=T, data=sat) # model1 now has the X matrix
> X = model1$x
> XpXinv = solve(t(X) %*% X)  # X-Prime-X Inverse
> s2 = summary(model1)$sigma^2  # MSE
> s2 * XpXinv
             (Intercept)        VERBAL          MATH
(Intercept)  0.1947522360 -1.214588e-04 -1.861318e-04
VERBAL      -0.0001214588  3.048530e-07 -9.257555e-08
MATH        -0.0001861318 -9.257555e-08  3.714564e-07
```

$$T = \frac{\mathbf{a}'\hat{\boldsymbol{\beta}} - \mathbf{a}'\boldsymbol{\beta}}{\sqrt{MSE\,\mathbf{a}'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{a}}}$$

```
> # Confidence interval for betahat1
> betahat1 = model1$coefficients[2]; betahat1
    VERBAL
0.002307001
> dfe = summary(model1)[[10]][3]; dfe
dendf
  197
> tcrit = qt(0.975,dfe); tcrit  # t-sub-alpha/2
[1] 1.972079
> se_1 = sqrt(V[2,2]); se_1 # Standard error of beta-hat1
[1] 0.000552135
> low1 = betahat1 - se_1*tcrit; up1 = betahat1 + se_1*tcrit
> c(low1, up1)
    VERBAL     VERBAL
0.001218147 0.003395855
```

$$T = \frac{\mathbf{a}'\widehat{\boldsymbol{\beta}} - \mathbf{a}'\boldsymbol{\beta}}{\sqrt{MSE\,\mathbf{a}'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{a}}}$$

```
> # Test equal regression coefficients for  Math and Verbal
> a = rbind(0,1,-1); a
     [,1]
[1,]    0
[2,]    1
[3,]   -1
> sediff = sqrt(t(a) %*% V %*% a )
> T = t(a) %*% model1$coefficients / sediff;   T
         [,1]
[1,] 1.411023
> T = as.numeric(T); T    # Make T numeric rather than a
[1] 1.411023
> T > tcrit
[1] FALSE
> 2 * (1-pt(T,dfe)) # 2-sided p-value
[1] 0.1598149
```