

```
## hw 1 stat 587
```

```
## Jennifer Weichenrieder
```

```
## February 20, 2019
```

```
# q 1
```

```
treata <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04, 79.97, 80.05, 80.03, 80.02, 80.00, 80.02)
```

```
treatb <- c(80.02, 79.94, 79.98, 79.97, 79.97, 80.03, 79.95, 79.97)
```

```
print(paste("Sample mean of method A:", mean(treata), sep = " "))
```

```
print(paste("Sample mean of method B:", mean(treatb), sep = " "))
```

```
# equal variance?
```

```
print(paste("Sample variance of method A:", var(treata), sep = " "))
```

```
print(paste("Sample variance of method B:", var(treatb), sep = " "))
```

```
# t test can help
```

```
# by default, R does not assume equal variance
```

```
t.test(treata, treatb, alternative = "two.sided", conf.level = 0.95, var.equal = FALSE)
```

```
t.test(treata, treatb, alternative = "two.sided", conf.level = 0.95, var.equal = TRUE)
```

```
# some plots
```

```
boxplot(treata, treatb, main = "Box Plot Comparison of Method A and Method B")
```

```
hist(treata, main = "Histogram of Method A")
```

```
hist(treatb, main = "Histogram of Method B")
```

```
qqnorm(treata, main = "Q-Q Plot for Treatment A")
```

```
qqline(treata, main = "Q-Q Plot for Treatment A")
```

```
qqnorm(treatb, main = "Q-Q Plot for Treatment B")
```

```
qqline(treatb, main = "Q-Q Plot for Treatment B")
```

```
# pooling data and creating kernel density function
```

```
treatpool <- c(treata, treatb)
```

```
density(treatpool, bw = 0.015, kernel = "triangular")
```

```
plot(density(treatpool, bw = 0.015, kernel = "triangular"),  
     main = "Kernel Density Estimation for Combined Data")
```

```
mydensity <- density(treatpool, bw = 0.015, kernel = "triangular")
```

```
# approximating the density at x= 80
```

```
approx(mydensity$x, mydensity$y, xout = 80)
```

```
# q 2
```

```
#
```

```
# entering cake brand data
```

```
# y is rating, x1 is moisture content, x2 is sweetness
```

```
x1 <- c(4, 4, 4, 4, 6, 6, 6, 6, 8, 8, 8, 8, 10, 10, 10, 10)
```

```
x2 <- c(2, 4, 2, 4, 2, 4, 2, 4, 2, 4, 2, 4, 2, 4, 2, 4)
```

```
y <- c(64, 73, 61, 76, 72, 80, 71, 83, 83, 89, 86, 93, 88, 95, 94, 100)
```

```
cake <- data.frame(y, x1, x2)
```

```
# model with interaction term
```

```

interact = lm(y ~ x1 + x2 + (x1*x2), data=cake)
summary(interact)
# model without interaction term
nointeract = lm(y ~ x1 + x2, data=cake)
summary(nointeract)
# looking at residuals
boxplot(interact$residuals, main = "Box plot of residuals, full model")
print(paste("Mean: ", mean(interact$residuals)))
print(paste("Standard deviation: ", sqrt(var(interact$residuals))))
par(mfrow=c(2,2))
plot(interact$fitted.values, interact$residuals, main = "Fitted values vs. residuals")
plot(interact$model$x1, interact$residuals, main = "Values of X1 vs. residuals")
plot(interact$model$x2, interact$residuals, main = "Values of X2 vs. residuals")
plot(interact$model$x1*interact$model$x2, interact$residuals, main = "Values of X1X2 vs.
residuals")
# ANOVA
anova(nointeract)

```

# q 3

```

# looking at hat matrix
print(hatvalues(nointeract))
# other measures of finding influential points
dif <- dffits(nointeract)
print(paste("DFFITS for Case 14: ", dif[14]))
beta <- dfbeta(nointeract)
print(paste("DFBETAS for Case 14: ", beta[14,]))
cook <- cooks.distance(nointeract)
print(paste("Cook's Distance for Case 14: ", cook[14]))
# what happens to the model when we remove the most influential point?
cake$oldfit <- nointeract$fitted.values
without <- cake[c(1:13, 15, 16),]
nointeract2 <- lm(y ~ x1 + x2, data=without)
without$newfit <- nointeract2$fitted.values
plot(without$newfit, without$oldfit)
# percent difference
without$pct <- abs(((without$newfit-without$oldfit)/(without$oldfit))*100)
print(paste("Average absolute percent difference: ",mean(without$pct)))
# Cook's Distance for all points
cooks.distance(nointeract)

```