Lab02 - t tests and confidence intervals for ANOVA

Your Name Here

Goals

The goal in this lab is to get some practice working with t-based inference for ANOVA models in R.

Loading packages

Here are some packages with functionality you may need for this lab. Run this code chunk now.

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(mosaic)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggformula
```

```
##
## New to ggformula?  Try the tutorials:
##  learnr::run_tutorial("introduction", package = "ggformula")
##  learnr::run_tutorial("refining", package = "ggformula")
```

```
## Loading required package: mosaicData
```

```
## Loading required package: Matrix
```

```
##
## The 'mosaic' package masks several functions from core packages in order to add
## additional features.  The original behavior of these functions should not be affected by this.
##
## Note: If you use the Matrix package, be sure to load it BEFORE loading mosaic.
```

```
##
## Attaching package: 'mosaic'

## The following object is masked from 'package:Matrix':
##
##     mean

## The following object is masked from 'package:ggplot2':
##
##     stat

## The following objects are masked from 'package:dplyr':
##
##     count, do, tally

## The following objects are masked from 'package:stats':
##
##     binom.test, cor, cor.test, cov, fivenum, IQR, median,
##     prop.test, quantile, sd, t.test, var

## The following objects are masked from 'package:base':
##
##     max, mean, min, prod, range, sample, sum
```

```
library(gmodels)

options("pillar.sigfig" = 10) # print 10 significant digits in summarize output
```

Reading in the Spock data

The following R code reads in the data set for the Spock Trial and takes a first look at the data. Run this code now; no need to modify it.

```
juries <- read_csv("http://www.evanlray.com/data/sleuth3/ex0502_women_jurors.csv")
```

```
## Parsed with column specification:
## cols(
##   Percent = col_double(),
##   Judge = col_character()
## )
```

```
dim(juries)
```

```
## [1] 46  2
```

```
head(juries)
```

```
## # A tibble: 6 x 2
##        Percent Judge
##          <dbl> <chr>
## 1  6.4         Spock's
## 2  8.700000000 Spock's
## 3 13.3         Spock's
## 4 13.6         Spock's
## 5 15           Spock's
## 6 15.2         Spock's
```

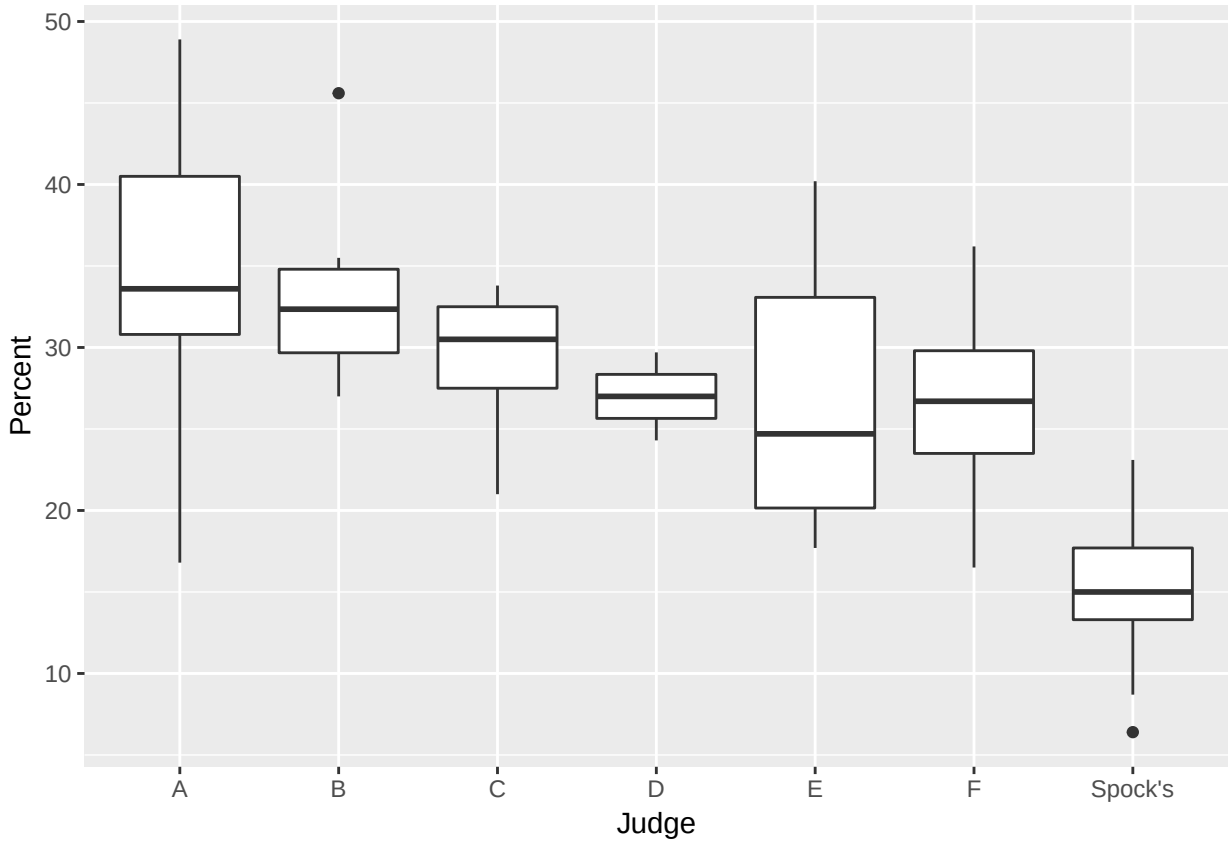
```
juries %>% count(Judge)
```

```
## # A tibble: 7 x 2
##   Judge       n
##   <chr>   <int>
## 1 A           5
## 2 B           6
## 3 C           9
## 4 D           2
## 5 E           6
## 6 F           9
## 7 Spock's     9
```

Make some plots

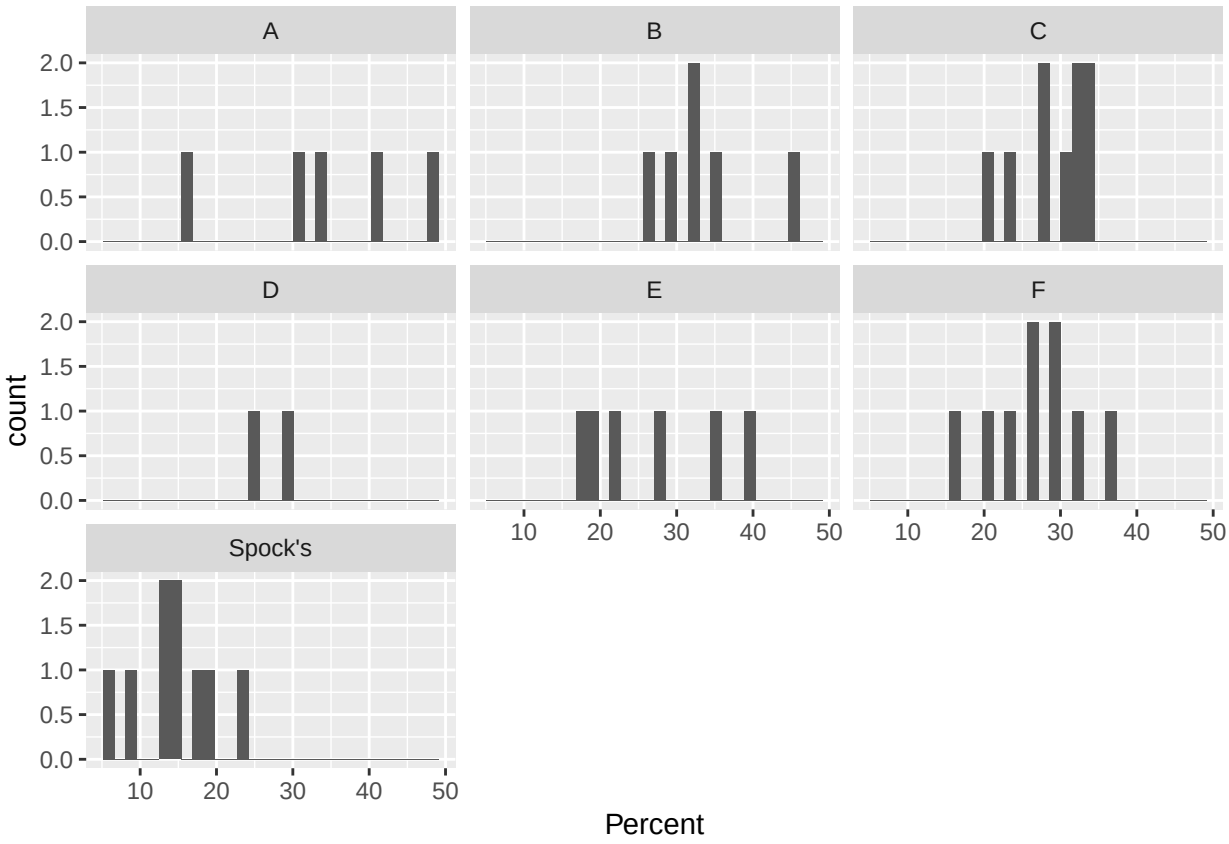
Use this space to make some plots of the Spock trial data.

```
ggplot(data = juries, mapping = aes(x = Judge, y = Percent)) +
  geom_boxplot()
```




```
ggplot(data = juries, mapping = aes(x = Percent)) +
  geom_histogram() +
  facet_wrap( ~ Judge)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



t tests and confidence intervals

Use this space to conduct hypothesis tests and find confidence intervals.

```
fit <- lm(Percent ~ Judge, data = juries)
```

Below is code relevant to part 2 (a) on the handout from 2019-02-04

```
library(gmodels)
fit.contrast(fit, "Judge", c(1, 0, 0, 0, 0, 0, -1), conf.int = 0.95)
```

```
##                           Estimate Std. Error  t value     Pr(>|t|)
## Judge c=( 1 0 0 0 0 0 -1 ) 19.49778   3.856562 5.055741 1.050245e-05
##                           lower CI upper CI
## Judge c=( 1 0 0 0 0 0 -1 ) 11.69715 27.29841
```

```
juries %>%
  group_by(Judge) %>%
  summarize(
    mean_pct_women = mean(Percent)
  )
```

```
## # A tibble: 7 x 2
##   Judge   mean_pct_women
##   <chr>            <dbl>
## 1 A          34.120000
## 2 B          33.61666667
```

```
## 3 C          29.1000000
## 4 D          27
## 5 E          26.96666667
## 6 F          26.8
## 7 Spock's    14.62222222
```

```r
34.12 - 14.62
```

```
## [1] 19.5
```

Below is code relevant to part 2 (b) on the handout from 2019-02-04

```r
library(gmodels)
fit.contrast(fit, "Judge", c(1/6, 1/6, 1/6, 1/6, 1/6, 1/6, -1), conf.int = 0.95)
```

```
##
## Judge c=( 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 (
##
## Judge c=( 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 (
##
## Judge c=( 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 (
##
## Judge c=( 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 (
##
## Judge c=( 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 (
##
## Judge c=( 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 0.166666666666667 (
```

The formatting of the output above is not good. Below I'm dropping that extra junk to make the results easier to read.

```r
temp <- fit.contrast(fit, "Judge", c(1/6, 1/6, 1/6, 1/6, 1/6, 1/6, -1), conf.int = 0.95)
rownames(temp) <- NULL
temp
```

```
##      Estimate Std. Error  t value     Pr(>|t|) lower CI upper CI
## [1,] 14.97833   2.641804 5.669737 1.489078e-06 9.634781 20.32189
```