MSI Statistical Training Series

Analysis of Variance (ANOVA)

2023-04-01

Table of contents

## Background

The MSI Statistical Training Series offers tutorials on a range of topics that are commonly applied in data analysis. The objective is to i) refresh readers on the derivation and application of methods from a first statistics course, ii) provide the underlying theory that goes beyond an introductory statistics course, iii) introduce readers to new methods that would not be covered in an introductory statistics course, and iv) provide practical, hands-on coding examples to enable users of statistical analysis software to gain proficiency in coding skills not just to run statistical procedures, but also to derive statistical procedures through the underlying theory.

Exploration of each topic should include the underlying theory, a demonstration using toy or fake data, and a real-world application in data that MSI has generated for a client deliverable.

## Introduction

We use the F-distribution, which is constructed from the ratio of two chi-square statistics, to generate the probability of observing the data under the assumption that the null hypothesis is true. A high ratio of between variance to within variance will generate a high F-statistic, which will generate a low p-value.

Introductory statistics texts provide a summary table to generate the F-statistic.

# create anova table  
  
aov\_ex <- data.frame(errors=c("Between","Within","Total"),  
 squares=c("SSB","SSW","SST"),  
 df=c("g-1","N-g","N-1"),  
 mn\_sq = c("SSB/DFB", "SSW/DFW", ""),  
 F=c("MSB/MSW","","")) %>%  
 flextable() %>%  
 set\_header\_labels(values=list(  
 errors="Error type",  
 squares="Designation",  
 df="Degrees of Freedom",  
 mn\_sq="Mean square"  
 )) %>%  
 autofit()  
  
aov\_ex

| Error type | Designation | Degrees of Freedom | Mean square | F |
| --- | --- | --- | --- | --- |
| Between | SSB | g-1 | SSB/DFB | MSB/MSW |
| Within | SSW | N-g | SSW/DFW |  |
| Total | SST | N-1 |  |  |

## Underlying theory

The analysis of variance theorem derives from the properties of conditional expectations – see (Wooldridge 2010), pages 31-32 for a proof. The ANOVA theorem states that

This may be paraphrased as saying the variance of a random variably Y, whose elements comprise several groups denoted as the random variable X, can be decomposed into the average variance of the sub-groups of X plus the variance of the means of each sub-group of X.

The ANOVA theorem may be used to test whether the sub-groups of X are statistically distinguishable from the overall distribution of Y. To get some intuition behind this, consider a null hypothesis in which all sub-group means of X are equal and share the same variance:

Note that we construct the data model as , where , is the grand (overall) mean of the sample, and the are deviations from the grand mean. So, if all means are the same the deviations are zero.

Using the ANOVA theorem, if the are zero, then their variances are zero, so the first term of the ANOVA theorem drops out. Furthermore, by assumption, there is only a single variance for all sub-groups. So the second term simplifies to .

The alternative hypothesis would be that the group means differ, still on the assumption of equal variance across sub-groups of X.

The key here is the variance term. If the sub-group means differ, their means will have a variance around the grand mean. If this ‘between’ variance of the sub-group means around the grand mean is large relative to the ‘within’ variance of group observations around the group mean (which by assumption would just be for each sub-group), then we have evidence that the group means do in fact differ from the grand mean.

We will now demonstrate the use of the ANOVA theorem to test for the equality of sub-groups, using constructed data.

## Demonstration

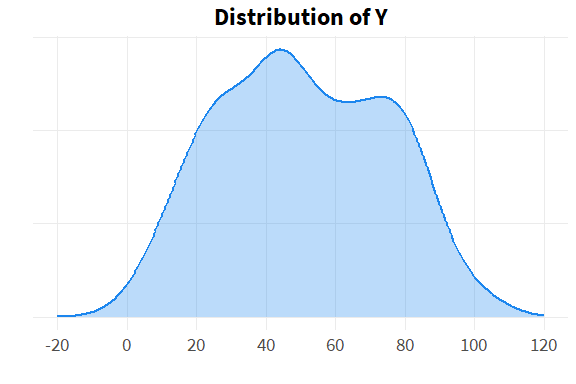
We construct a variable Y with sub-groups X with different means and standard deviations.

set.seed(4632)  
  
grp\_a <- rnorm(50, 25 ,12) %>%  
 round(0)  
  
grp\_b <- rnorm(50, 45, 20) %>%  
 round(0)  
  
grp\_c <- rnorm(50,55, 20) %>%  
 round(0)  
  
grp\_d <- rnorm(50, 75, 12) %>%  
 round(0)  
  
d <- data.frame(group=rep(letters[1:4], each=50),  
 y=c(grp\_a, grp\_b, grp\_c, grp\_d))  
flextable(d[c(1:2, 51:52, 101:102, 151:152),])

| group | y |
| --- | --- |
| a | 42 |
| a | 42 |
| b | 35 |
| b | 30 |
| c | 61 |
| c | 43 |
| d | 68 |
| d | 101 |

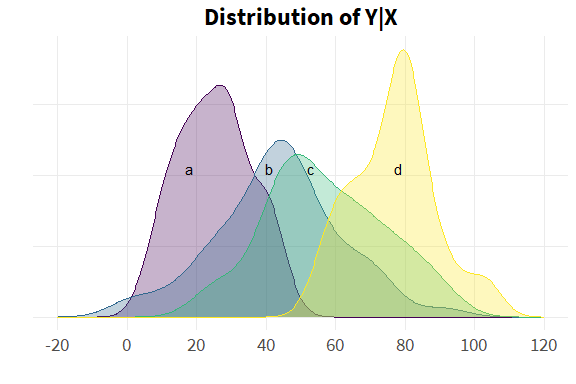
Let’s look at our data.

# distribution of overall data Y  
dens\_y <- ggplot(d, aes(x=y)) +   
 geom\_density(alpha=.3, fill="dodgerblue2", color="dodgerblue2", linewidth=1) +  
 scale\_x\_continuous(limits=c(-20,120),  
 breaks=seq(-20, 120, 20)) +  
 theme(axis.text.y=element\_blank()) +  
 labs(x="",  
 y="",  
 title="Distribution of Y")  
dens\_y



The data can be described as roughly normal, though there is clearly the beginning of a second mode at around 75, and the first hint of another mode at around 25. We know by construction that this is due to the presence of the sub-groups X. If we had knowledge only of Y but not of X, we might suspect the presence of hidden structure (sub-groups) in the data, but would not know.

# distribution of sub-groups X  
  
dens\_yx <- ggplot(d, aes(x=y, group=group, fill=group, color=group)) +   
 geom\_density(alpha=.3) +  
 scale\_x\_continuous(limits=c(-20,120),  
 breaks=seq(-20, 120, 20)) +  
 scale\_color\_viridis\_d() +  
 scale\_fill\_viridis\_d() +   
 annotate("text", x=c(18, 41, 53, 78), y=.021, label=letters[1:4]) +  
 theme(axis.text.y=element\_blank(),  
 legend.position="none") +  
 labs(x="",  
 y="",  
 title="Distribution of Y|X")  
  
dens\_yx



Let’s extract the properties of the data that we will use to construct our test statistic.

g <- 4 # number of groups  
g

[1] 4

n\_g <- length(grp\_a) # sample size of each group  
n\_g

[1] 50

N <- nrow(d) # overall sample size  
N

[1] 200

d\_grnd\_mn <- mean(d$y) # overall mean of Y   
d\_grnd\_mn

[1] 50.6

d\_mns <- d %>% # sub-group means  
 group\_by(group) %>%  
 summarise(grp\_mn=mean(y),  
 se=std.error(y))  
  
flextable(d\_mns)

| group | grp\_mn | se |
| --- | --- | --- |
| a | 25.1 | 1.48 |
| b | 43.5 | 2.59 |
| c | 57.0 | 2.39 |
| d | 76.7 | 1.68 |

First we need the between sum of squares. As a reminder, this is the variation of the sub-group means around their grand mean.

As the sub-group means do not vary within their own group, this reduces to the variance of the sub-group means around the grand mean, weighted by the sample size of each sub-group.

Next we need the within sum of squares. As a reminder, this is the variation of a sub-group observation around its sub-group mean.

It can be shown that this expression is the same as the overall variance times the sample size.

This also matches the intuition that, under the null of equal variances, the variation of a sub-group observation around its sub-group mean is no different than the variation of any observation around the grand mean.

Let’s try to illustrate the different type of variation graphically.

set.seed(422)  
  
grp2\_a <- sample(1:4, 4, T)  
grp2\_a

[1] 3 4 2 4

grp2\_b <- sample(5:8, 4, T)  
grp2\_b

[1] 5 6 6 8

grp2\_c <- sample(9:12, 4, T)  
grp2\_c

[1] 10 10 12 10

var\_ex <- data.frame(id=1:12,  
 Group=rep(letters[1:3], each=4),  
 y=c(grp2\_a, grp2\_b, grp2\_c)) %>%  
 arrange(y)   
  
var\_ex

|  |  |  |
| --- | --- | --- |
| **id** | **Group** | **y** |
| 3 | a | 2 |
| 1 | a | 3 |
| 2 | a | 4 |
| 4 | a | 4 |
| 5 | b | 5 |
| 6 | b | 6 |
| 7 | b | 6 |
| 8 | b | 8 |
| 9 | c | 10 |
| 10 | c | 10 |
| 12 | c | 10 |
| 11 | c | 12 |

describe(var\_ex)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **vars** | **n** | **mean** | **sd** | **median** | **trimmed** | **mad** | **min** | **max** | **range** | **skew** | **kurtosis** | **se** |
| 1 | 12 | 6.5 | 3.61 | 6.5 | 6.5 | 4.45 | 1 | 12 | 11 | 0 | -1.5 | 1.04 |
| 2 | 12 | 2 | 0.853 | 2 | 2 | 1.48 | 1 | 3 | 2 | 0 | -1.74 | 0.246 |
| 3 | 12 | 6.67 | 3.26 | 6 | 6.6 | 3.71 | 2 | 12 | 10 | 0.172 | -1.56 | 0.94 |

var\_ex\_grnd\_mn <- mean(var\_ex$y)  
var\_ex\_grnd\_mn

[1] 6.67

var\_ex\_mns <- var\_ex %>%  
 group\_by(Group) %>%  
 summarise(grp\_mn=mean(y))  
  
var\_ex\_mns

|  |  |
| --- | --- |
| **Group** | **grp\_mn** |
| a | 3.25 |
| b | 6.25 |
| c | 10.5 |

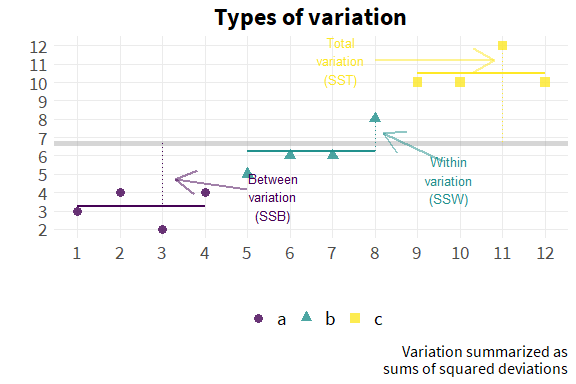
mns2 <- unlist(var\_ex\_mns$grp\_mn)  
mns2

[1] 3.25 6.25 10.50

var\_ex\_mns[1,2]

|  |
| --- |
| **grp\_mn** |
| 3.25 |

cols <- viridis(3)  
  
ggplot(var\_ex, aes(id, y, group=Group, color=Group, shape=Group)) +   
 geom\_hline(yintercept=var\_ex\_grnd\_mn, color="grey60", linewidth=2, alpha=.4) +  
 geom\_point(size=3, alpha=.8) +  
# scale\_color\_viridis\_d() +  
 scale\_color\_manual(values=cols) +  
 scale\_x\_continuous(breaks=1:15) +  
 scale\_y\_continuous(breaks=1:30) +  
 geom\_segment(aes(x=1, xend=4, y=mns2[1], yend=mns2[1]),   
 color=cols[1],  
 size=1,  
 alpha=.4) +  
 geom\_segment(aes(x=5, xend=8, y=mns2[2], yend=mns2[2]),   
 color=cols[2],  
 size=1,  
 alpha=.4) +  
 geom\_segment(aes(x=9, xend=12, y=mns2[3], yend=mns2[3]),   
 color=cols[3],  
 size=1,  
 alpha=.4) +  
 geom\_segment(aes(x=3, xend=3, y=mns2[1]+.1, yend=var\_ex\_grnd\_mn),  
 color=cols[1], linewidth=.7, linetype="dotted", alpha=.5) +  
 geom\_segment(aes(x=8, xend=8, y=7.8, yend=mns2[2]),  
 color=cols[2], linewidth=.7, linetype="dotted", alpha=.5) +  
# geom\_segment(aes(x=5, xend=5, y=5.2, yend=grnd\_mn),  
# color=col[2], size=1, linetype="dotted", alpha=.5) +  
 geom\_segment(aes(x=11, xend=11, y=11.8, yend=var\_ex\_grnd\_mn),  
 color=cols[3], linewidth=.7, linetype="dotted", alpha=.5) +  
 annotate("segment", x=5, xend=3.3, y=4.2, yend=4.7, color=cols[1], size=.8, arrow=arrow(), alpha=.5) +  
 annotate("text", x=5.6, y=3.8, label="Between\nvariation\n(SSB)", size=3.4, color=cols[1]) +  
 annotate("segment", x=9.6, xend=8.2, y=5.7, yend=7.2, color=cols[2], size=.8, arrow=arrow(), alpha=.5) +  
 annotate("text", x=9.75, y=4.7, label="Within\nvariation\n(SSW)", size=3.4, color=cols[2]) +  
# geom\_label(aes(x=5.8, y=4.1, label="Between\nvariation\n(SSB)"),   
# color=cols[1])  
 annotate("segment", x=8, xend=10.8, y=11.2, yend=11.2, color=cols[3], size=.8, arrow=arrow(), alpha=.5) +  
 annotate("text", x=7.2, y=11.2, label="Total\nvariation\n(SST)", size=3.4, color=cols[3]) +  
 theme(legend.position="bottom",  
 legend.title=element\_blank()) +  
 labs(x="",  
 y="",  
 title="Types of variation",  
 caption="Variation summarized as\nsums of squared deviations")



We can see the squared deviations explicitly by constructing new variables in our data set.

# decompose variance across groups  
  
d <- d %>%  
 left\_join(d\_mns[,1:2]) %>%  
 group\_by(group) %>%  
 mutate(w\_dev=y-mean(y),  
 sq\_w\_dev=w\_dev^2) %>%  
 ungroup() %>%  
 mutate(bet\_dev = grp\_mn-d\_grnd\_mn,  
 sq\_bet\_dev = bet\_dev^2, # squared between deviation  
 grnd\_mn = d\_grnd\_mn,  
 grnd\_dev=d\_grnd\_mn-y,  
 sq\_grnd\_dev=grnd\_dev^2) # squared within deviation  
  
#flextable(d)  
  
flextable(d[c(1:2, 51:52, 101:102, 151:152),])

| group | y | grp\_mn | w\_dev | sq\_w\_dev | bet\_dev | sq\_bet\_dev | grnd\_mn | grnd\_dev | sq\_grnd\_dev |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a | 42 | 25.1 | 16.88 | 284.9 | -25.46 | 648.0 | 50.6 | 8.58 | 73.5 |
| a | 42 | 25.1 | 16.88 | 284.9 | -25.46 | 648.0 | 50.6 | 8.58 | 73.5 |
| b | 35 | 43.5 | -8.46 | 71.6 | -7.12 | 50.6 | 50.6 | 15.58 | 242.6 |
| b | 30 | 43.5 | -13.46 | 181.2 | -7.12 | 50.6 | 50.6 | 20.58 | 423.3 |
| c | 61 | 57.0 | 4.02 | 16.2 | 6.40 | 41.0 | 50.6 | -10.42 | 108.7 |
| c | 43 | 57.0 | -13.98 | 195.4 | 6.40 | 41.0 | 50.6 | 7.58 | 57.4 |
| d | 68 | 76.7 | -8.74 | 76.4 | 26.16 | 684.6 | 50.6 | -17.42 | 303.6 |
| d | 101 | 76.7 | 24.26 | 588.5 | 26.16 | 684.6 | 50.6 | -50.42 | 2,542.7 |

And then we would just take the sum of the relevant variable. We can also use the shortcut equations as a check.

SSW:

# ssw sum of squares within groups (distance between observation and its group mean)  
  
ssw <- sum(d$sq\_w\_dev)  
ssw # 42642

[1] 42642

ssw\_check <- d %>%  
 group\_by(group) %>%  
 summarise(ss = var(y) \* (n\_g-1) ) %>%  
 summarise(sum(ss))  
  
flextable(ssw\_check) # 42642

| sum(ss) |
| --- |
| 42,642 |

SSB:

ssb <- sum(d$sq\_bet\_dev)  
ssb # 71211

[1] 71211

ssb\_check <- d\_mns %>%  
 mutate(bet\_dev=grp\_mn-d\_grnd\_mn,  
 sq\_bet\_dev=bet\_dev^2) %>%  
 summarise(ssb=sum(sq\_bet\_dev\*n\_g))  
  
flextable(ssb\_check) # 71211

| ssb |
| --- |
| 71,211 |

Total sum of squares (SST):

# sst total sum of squares (observations from grand mean)  
  
d\_mns <- d\_mns %>%  
 mutate(grnd\_dev=grp\_mn-d\_grnd\_mn,  
 sq\_grnd\_dev=grnd\_dev^2)  
  
flextable(d\_mns)

| group | grp\_mn | se | grnd\_dev | sq\_grnd\_dev |
| --- | --- | --- | --- | --- |
| a | 25.1 | 1.48 | -25.46 | 648.0 |
| b | 43.5 | 2.59 | -7.12 | 50.6 |
| c | 57.0 | 2.39 | 6.40 | 41.0 |
| d | 76.7 | 1.68 | 26.16 | 684.6 |

sst <- sum(d$sq\_grnd\_dev)  
sst # 113853

[1] 113853

ssw + ssb # 113853

[1] 113853

Now we can create our ANOVA table with actual values.

# create anova table  
  
aov\_tab <- data.frame(errors=c("Between (SSB)","Within (SSW)","Total (SST)"),  
 #squares=c("SSB","SSW","SST"),  
 ss\_act=c(ssb, ssw, sst),  
 #df=c("g-1","N-g","N-1"),  
 df\_act = c(g-1, N-g, N-1)) %>%  
 #mn\_sq = c("SSB/DFB", "SSW/DFW", "")) %>%  
 #F=c("MSB/MSW","","")) %>%  
 mutate(mn\_sq\_act=ss\_act/df\_act,  
 F\_act=c(mn\_sq\_act[1]/mn\_sq\_act[2], NA, NA))  
  
aov\_tab <- aov\_tab %>%  
 flextable() %>%  
 set\_header\_labels(values=list(  
 errors="Error type",  
 ss\_act="Sum of squares",  
 df\_act="Degrees of Freedom",  
 mn\_sq\_act="Mean square",  
 F\_act="F"  
 )) %>%  
 autofit()  
  
aov\_tab

| Error type | Sum of squares | Degrees of Freedom | Mean square | F |
| --- | --- | --- | --- | --- |
| Between (SSB) | 71,211 | 3 | 23,737 | 109 |
| Within (SSW) | 42,642 | 196 | 218 |  |
| Total (SST) | 113,853 | 199 | 572 |  |

We can use R’s built in functions to generate the critical value of the F distribution, as well as the probability (p-value) of observing the actual value under the null hypothesis.

fstat <- round(( ssb/(g-1) ) / ( ssw/(N-g) ), 2)   
  
fstat # 109

[1] 109

crit <- qf(p=.05,  
 df1=g-1,  
 df2=N-g,  
 lower.tail=F)  
  
crit

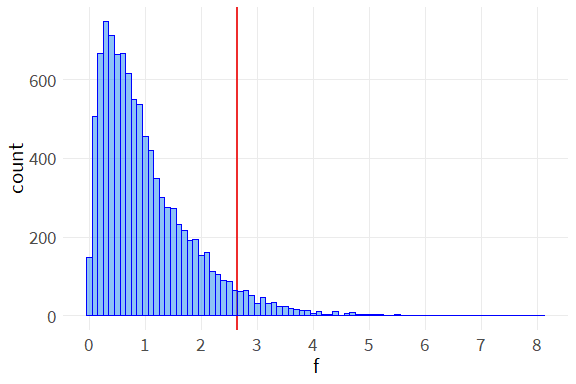
[1] 2.65

p <- pf(fstat, g-1, N-g, lower.tail=F)  
p

[1] 1.42e-41

For explication purposes, let’s look at the distribution of F-statistics that are generated under our null hypothesis.

y\_rf <- data.frame(f=rf(1e4, g-1, N-g))  
  
ggplot(y\_rf, aes(f)) +  
 geom\_vline(xintercept=crit, color="firebrick2",  
 linewidth=1) +  
 geom\_histogram(#data=filter(y\_rf, x<crit),   
 color="blue",  
 fill="dodgerblue2",  
 bins=40,  
 binwidth=.1,  
 alpha=.5) +  
 scale\_x\_continuous(breaks=0:8)



Our observed F-statistic of 109 is far above the critical value of 2.7, indicating the inequality of sub-group means.

We have completed our ANOVA test for the equality of means, and we’ve done it mostly by hand. There are several ways to apply ANOVA tests using existing functions within R. In base R, we call the function ‘aov’ on a formula, or we call ‘anova’ on a linear model.

aov:

av <- aov(y~group, d)  
summary(av)

Df Sum Sq Mean Sq F value Pr(>F)   
group 3 71211 23737 109 <2e-16 \*\*\*  
Residuals 196 42642 218   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova:

rg <- lm(y~group, d)   
  
anova(rg) %>%  
 flextable() %>%  
 colformat\_double(j=5, digits=3)

| Df | Sum Sq | Mean Sq | F value | Pr(>F) |
| --- | --- | --- | --- | --- |
| 3 | 71,211 | 23,737 | 109 | 0.000 |
| 196 | 42,642 | 218 |  |  |

We can also compare the sub-groups explicitly in a regression framework.

rg\_flx <- rg %>%  
 tidy() %>%  
 flextable() %>%  
 colformat\_double(j=5, digits=3)  
  
rg\_flx

| term | estimate | std.error | statistic | p.value |
| --- | --- | --- | --- | --- |
| (Intercept) | 25.1 | 2.09 | 12.04 | 0.000 |
| groupb | 18.3 | 2.95 | 6.22 | 0.000 |
| groupc | 31.9 | 2.95 | 10.80 | 0.000 |
| groupd | 51.6 | 2.95 | 17.50 | 0.000 |

Note that the intercept is not centered to represent the grand mean, and the t-tests associated with each individual group mean are nonsensical (statistically different from zero).

Returning to the ANOVA output, the evidence suggests that the group means differ from the grand mean, but we don’t necessarily know which groups may differ and which groups may not. There are a number of post hoc significance tests to address this.

TukeyHSD(av)

Tukey multiple comparisons of means  
 95% family-wise confidence level  
  
Fit: aov(formula = y ~ group, data = d)  
  
$group  
 diff lwr upr p adj  
b-a 18.3 10.70 26.0 0  
c-a 31.9 24.22 39.5 0  
d-a 51.6 43.98 59.3 0  
c-b 13.5 5.88 21.2 0  
d-b 33.3 25.64 40.9 0  
d-c 19.8 12.12 27.4 0

ScheffeTest(av)

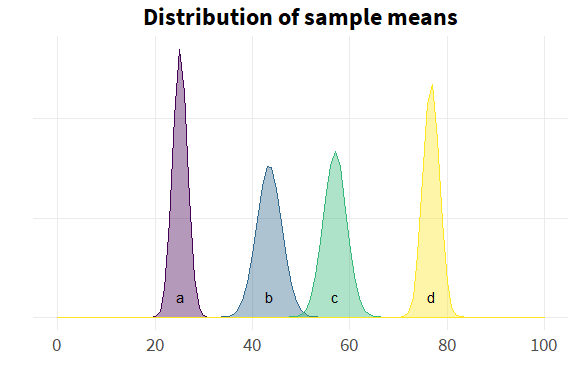
Posthoc multiple comparisons of means: Scheffe Test   
 95% family-wise confidence level  
  
$group  
 diff lwr.ci upr.ci pval   
b-a 18.3 10.0 26.7 0.0000001020 \*\*\*  
c-a 31.9 23.5 40.2 < 2e-16 \*\*\*  
d-a 51.6 43.3 59.9 < 2e-16 \*\*\*  
c-b 13.5 5.2 21.8 0.00017 \*\*\*  
d-b 33.3 25.0 41.6 < 2e-16 \*\*\*  
d-c 19.8 11.4 28.1 0.0000000083 \*\*\*  
  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

pairwise.t.test(d$y, d$group, p.adj="holm")

Pairwise comparisons using t tests with pooled SD   
  
data: d$y and d$group   
  
 a b c   
b 0.0000000060 - -   
c < 2e-16 0.0000081476 -   
d < 2e-16 < 2e-16 0.0000000007  
  
P value adjustment method: holm

A previous examination of the distribution of the sub-group data may have suggested that groups B and C would not differ. However, even after adjusting for multiple comparisons, the post hoc tests suggest that all sub-groups differ. It should be remembered that tests for differences in means may be statistically significant, even when there is extensive overlap in their distributions. The differences in means may be seen more clearly by examining the distribution of sub-group sample means, rather than the distribution of the sub-group data.

# distribution of sample means ----   
  
ggplot() +  
 stat\_function(fun = dnorm,   
 args=list(mean=d\_mns$grp\_mn[1],  
 sd=d\_mns$se[1]),  
 geom = "polygon",  
 color = col[1],   
 fill = col[1],   
 alpha = 0.4) +  
 stat\_function(fun = dnorm,   
 args=list(mean=d\_mns$grp\_mn[2],  
 sd=d\_mns$se[2]),  
 geom = "polygon",  
 color = col[2],   
 fill = col[2],   
 alpha = 0.4) +  
 stat\_function(fun = dnorm,   
 args=list(mean=d\_mns$grp\_mn[3],  
 sd=d\_mns$se[3]),  
 geom = "polygon",  
 color = col[3],   
 fill = col[3],   
 alpha = 0.4) +  
 stat\_function(fun = dnorm,   
 args=list(mean=d\_mns$grp\_mn[4],  
 sd=d\_mns$se[4]),  
 geom = "polygon",  
 color = col[4],   
 fill = col[4],   
 alpha = 0.4) +  
 scale\_x\_continuous(limits=c(0,100),  
 breaks=seq(0,100,20)) +  
 theme(axis.text.y=element\_blank()) +  
 labs(x="",  
 y="",  
 title="Distribution of sample means") +  
 annotate("text", x=c(25.1, 43.5, 57, 76.7), y=.021, label=letters[1:4])



Now the inequality of sub-group means is much clearer.

## Application

We will now show an application of ANOVA to real-world data that MSI collected, whose results were included in a client deliverable.

## Resources

This tutorial relied on the following resources:

* Course notes for the Statistics pre-requisite in the Johns Hopkins Applied Economics program
* Applied Statistics with R, a text used in Stat 420, Methods of Applied Statistics at the University of Illinois (Dalpiaz, n.d.)
* Common statistical tests are linear models, chapter 7 (Doogue, n.d.)

Dalpiaz, David. n.d. *Applied Statistics with r*. <https://book.stat420.org/>.

Doogue, Steve. n.d. *Common Statistical Tests Are Linear Models: A Work Through*. <https://steverxd.github.io/Stat_tests/>.

Wooldridge, Jeff. 2010. “Econometric Analysis of Cross Section and Panel Data.” <https://mitpress.mit.edu/9780262232586/econometric-analysis-of-cross-section-and-panel-data/>.