Independent t-test and Hypothesis Testing

Mike Nguyen

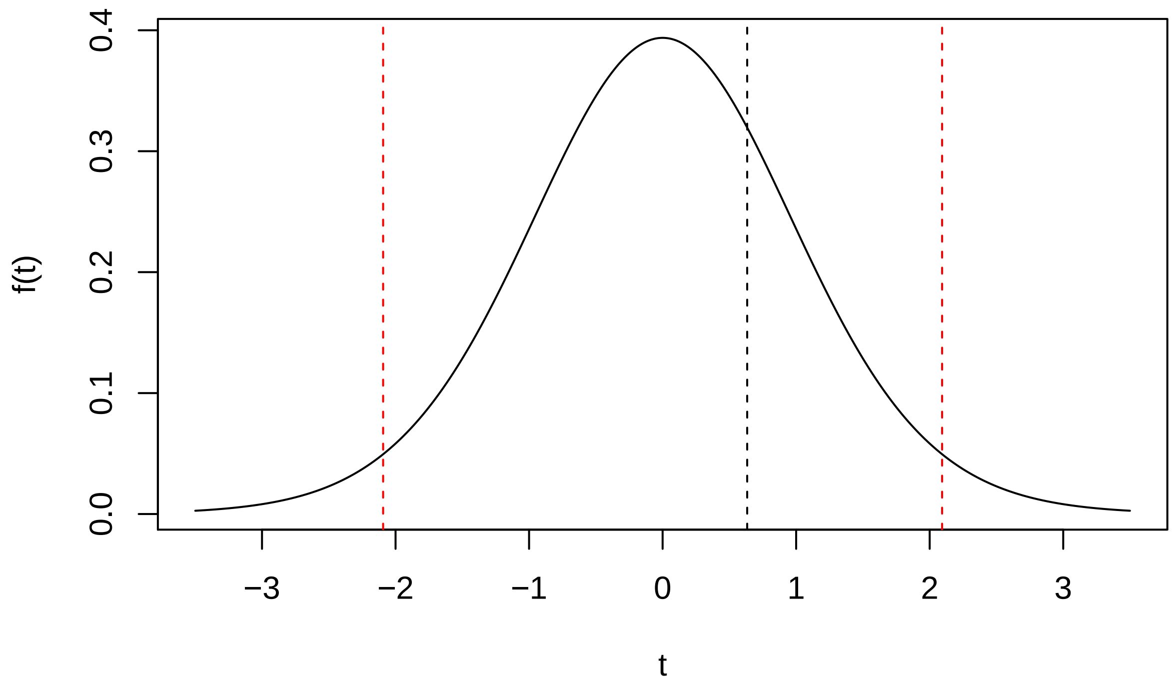
Hypothesis Test

Visualization 1

```
N = 20 #just chosen arbitrarily (20 responses)

samp = rnorm(N, mean = 0, sd = 1)

myTest = t.test(samp, mu = 0, alternative = "two.sided")
myTest
```

```
##
##  One Sample t-test
##
## data:  samp
## t = 0.63345, df = 19, p-value = 0.534
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.3000022  0.5604045
## sample estimates:
## mean of x
## 0.1302011
```

```
# get t-critical value (just like when you look up t-critical value)
tcrit = qt(0.025, df = (N - 1)) # alpha = 0.025

dum=seq(-3.5, 3.5, length=10^4) # For the plot


plot(dum, dt(dum, df=(N-1)), type='l', xlab='t', ylab='f(t)')
abline(v=myTest$statistic, lty=2)
abline(v=tcrit, col='red', lty=2)
abline(v=-tcrit, col='red', lty=2)
```

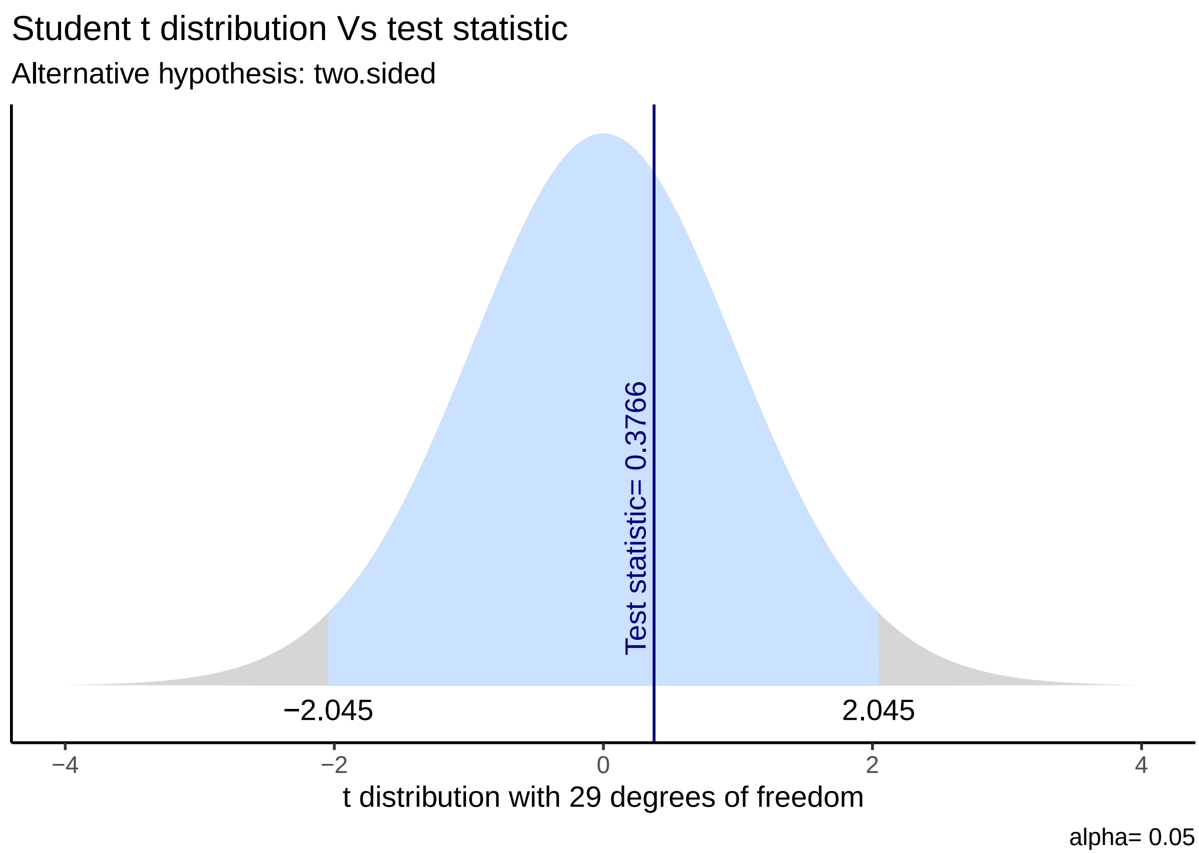


Visualization 2

```
library(MASS)
h=na.omit(survey$Height)
pop.mean=mean(h)
h.sample = sample(h,30)
t.test(h.sample,mu=pop.mean)
```

```
##
##  One Sample t-test
##
## data:  h.sample
## t = 0.37664, df = 29, p-value = 0.7092
## alternative hypothesis: true mean is not equal to 172.3809
## 95 percent confidence interval:
##  168.8907 177.4466
## sample estimates:
## mean of x
##  173.1687
```

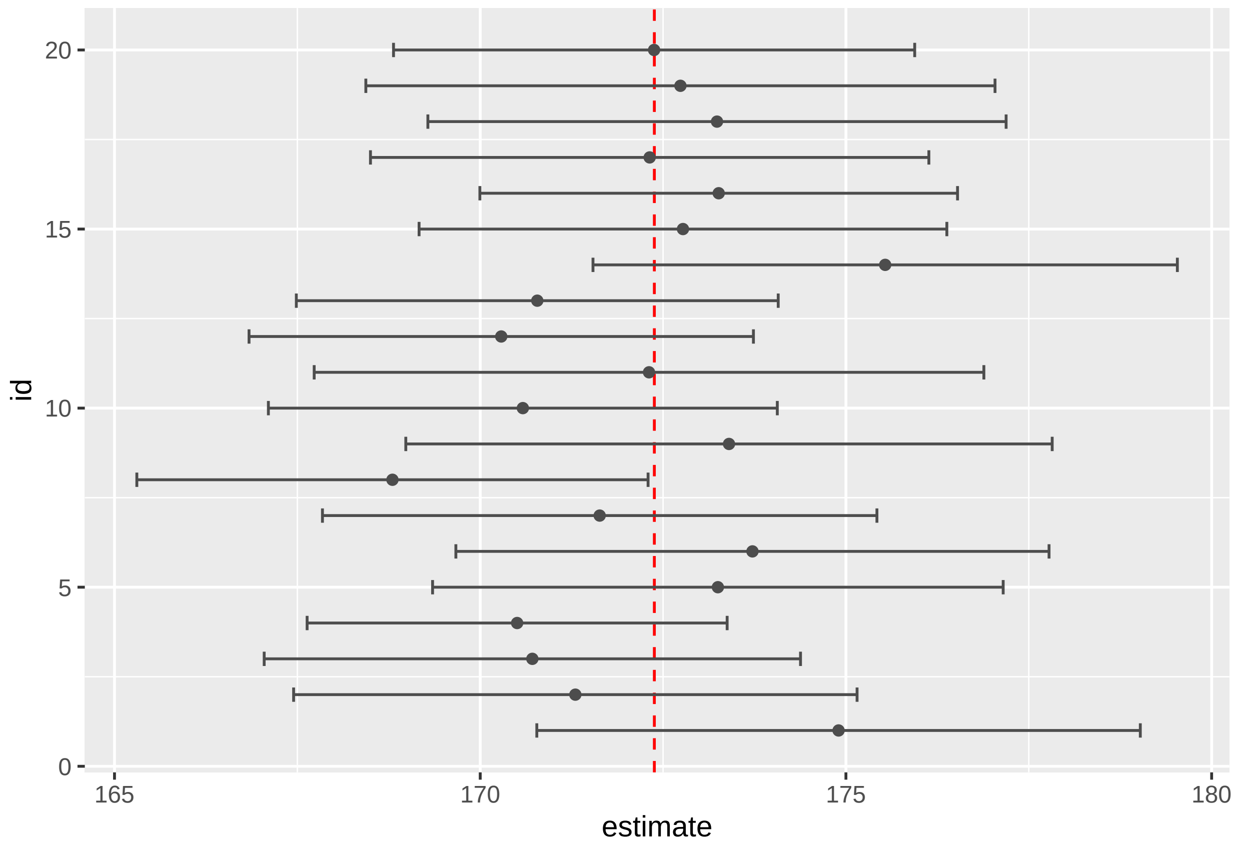
```
library(gginference)
ggttest(t.test(h.sample,mu=pop.mean))
```



Visualization 3

```
library(MASS)
library(ggplot2)

h = na.omit(survey$Height)
pop.mean = mean(h)

n_reps = 20
sample_size = 30
res_list = list()

for (i in 1:n_reps) {
  h.sample = sample(h, sample_size)
  res_list[[i]] = t.test(h.sample, mu = pop.mean)
}

dat = data.frame(
  id = seq(length(res_list)),
  estimate = sapply(res_list, function(x)
    x$estimate),
  conf_int_lower = sapply(res_list, function(x)
    x$conf.int[1]),
  conf_int_upper = sapply(res_list, function(x)
```

```
    x$conf.int[2])
)

p = ggplot(data = dat, aes(x = estimate, y = id)) +
  geom_vline(xintercept = pop.mean,
             color = "red",
             linetype = 2) +
  geom_point(color = "grey30") +
  geom_errorbarh(
    aes(xmin = conf_int_lower, xmax = conf_int_upper),
    color = "grey30",
    height = 0.4
  )
p
```



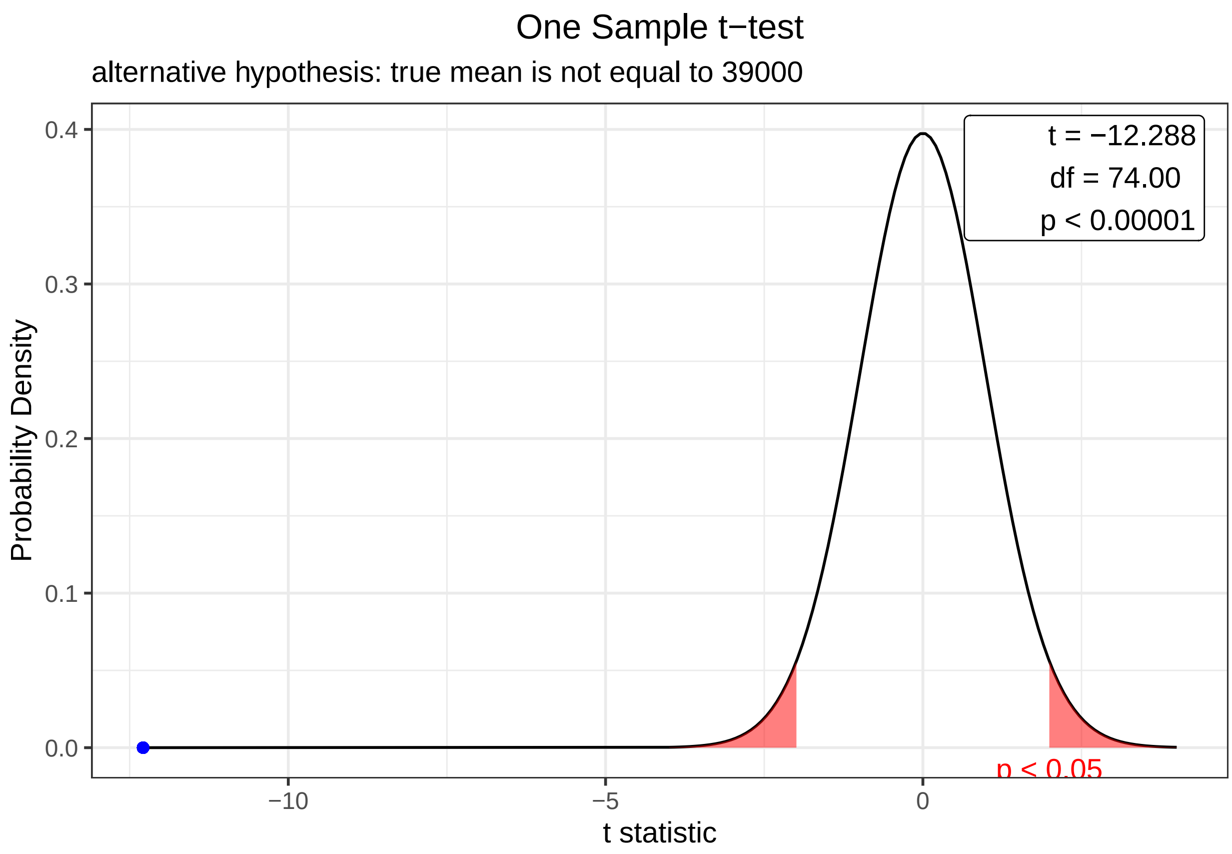

```
# ggsave("CI_plot.png", plot=p, height=4, width=6, units="in", dpi=150)
```

One sample t-test

```
library(webr)
library(moonBook)

set.seed(0)
treeVolume <- c(rnorm(75, mean = 36500, sd = 2000))
t.test(treeVolume, mu = 39000) # Ho: mu = 39000
```

```
##
##  One Sample t-test
##
## data:  treeVolume
## t = -12.288, df = 74, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 39000
## 95 percent confidence interval:
##  36033.60 36861.38
## sample estimates:
## mean of x
##  36447.49
```

```
plot(t.test(treeVolume, mu = 39000))
```



Generate Data

```
set.seed(0)

ClevelandSpending <- rnorm(50, mean = 250, sd = 75)
NYSpending <- rnorm(50, mean = 300, sd = 80)

spending <- c(ClevelandSpending, NYSpending)
city <- c(rep("Cleveland", 50), rep("New York", 50))
```

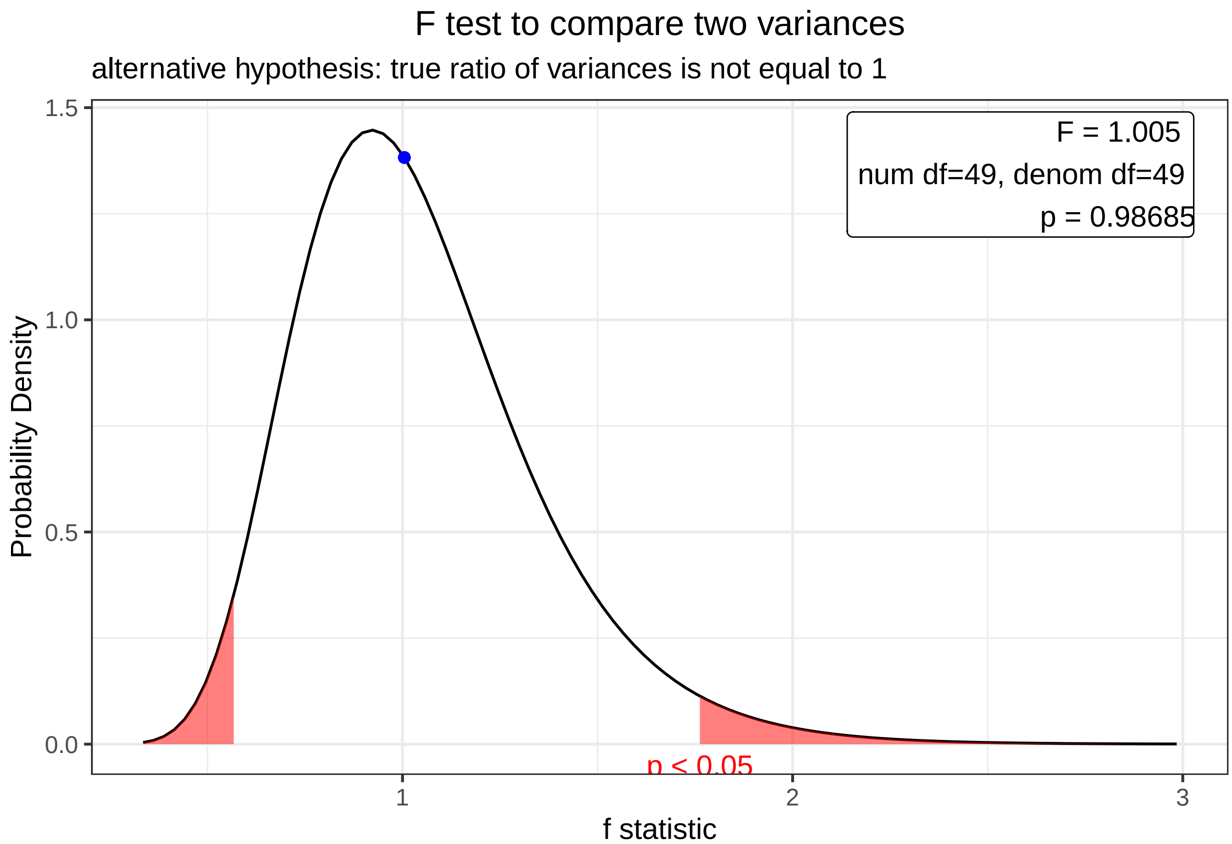
```r
#Sample data
data = data.frame(spending = spending,
                  city = city)

# export data to excel
# rio::export(
#   cbind(ClevelandSpending, NYSpending),
#   file.path(getwd(), "lectures", "10", "independent_t.xlsx")
# )
```

F-test for 2 variances

```r
var.test(ClevelandSpending, NYSpending)
```

```
##
##  F test to compare two variances
##
## data:  ClevelandSpending and NYSpending
## F = 1.0047, num df = 49, denom df = 49, p-value = 0.9869
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.5701676 1.7705463
## sample estimates:
## ratio of variances
##           1.004743
```

```r
# Alternatively
F_test = var.test(spending ~ city, data = data)
plot(F_test)
```



Two Sample t-test

Equal Variances

```
t.test(ClevelandSpending, NYSpending, var.equal = TRUE)
```

```
##
##  Two Sample t-test
##
## data:  ClevelandSpending and NYSpending
## t = -3.6361, df = 98, p-value = 0.0004433
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -77.1608 -22.6745
## sample estimates:
## mean of x mean of y
##  251.7948  301.7125
```

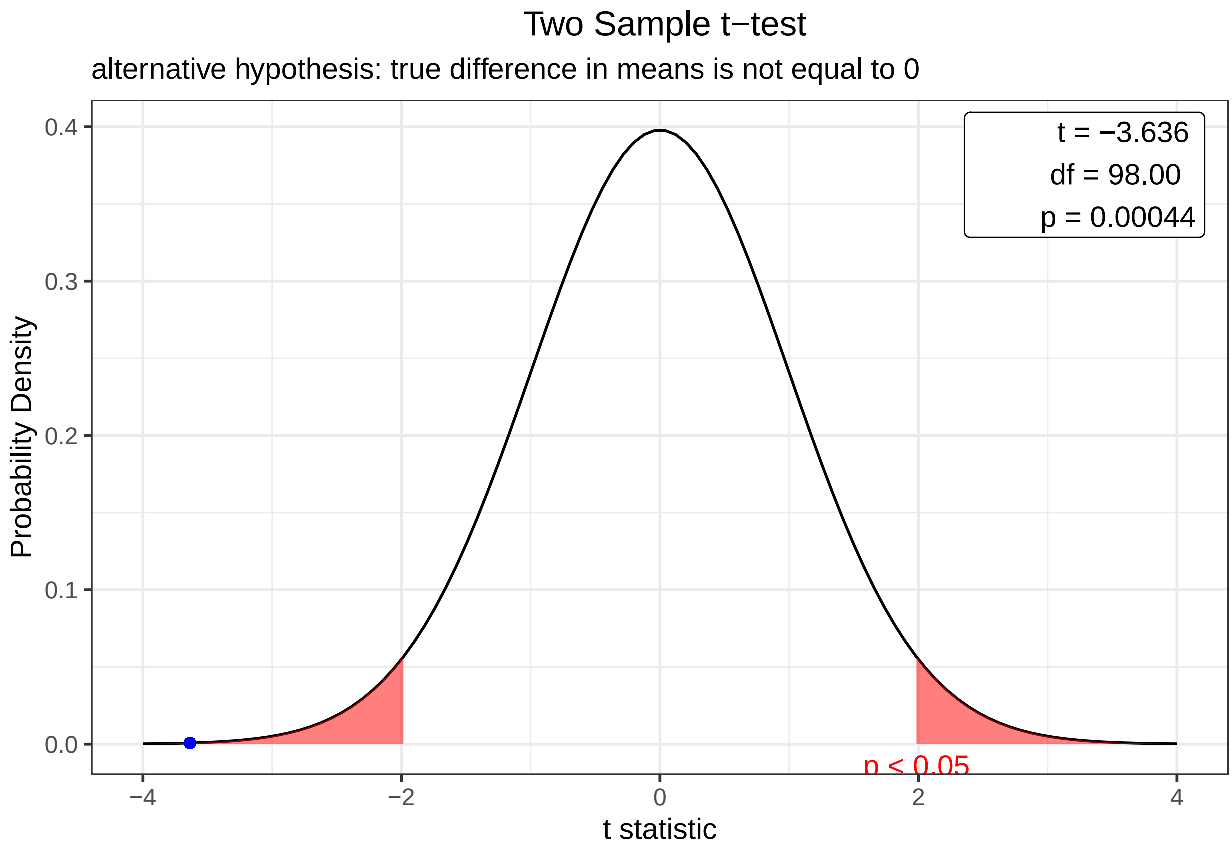
Equivalently,

```
t.test(spending ~ city, var.equal = TRUE)
```

```
##
##  Two Sample t-test
##
## data:  spending by city
```

```
## t = -3.6361, df = 98, p-value = 0.0004433
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -77.1608 -22.6745
## sample estimates:
## mean in group Cleveland  mean in group New York
##                251.7948                301.7125
```

```
plot(t.test(spending ~ city, var.equal = TRUE))
```



Unequal Variances

what if we have different variance for the two variables

```
t.test(ClevelandSpending, NYSpending, var.equal = FALSE)
```

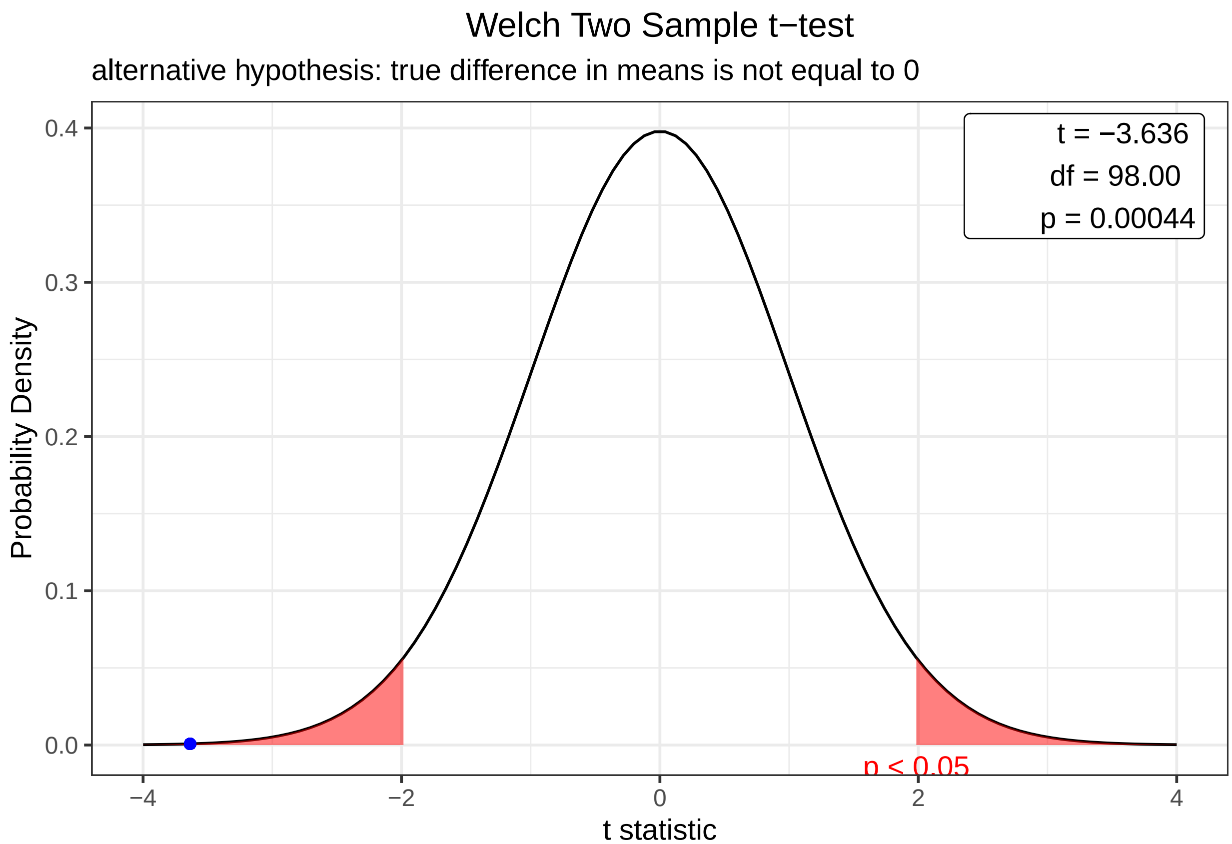
```
##
##  Welch Two Sample t-test
##
## data:  ClevelandSpending and NYSpending
## t = -3.6361, df = 97.999, p-value = 0.0004433
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -77.1608 -22.6745
## sample estimates:
## mean of x mean of y
##  251.7948  301.7125
```

Alternatively,

```
t.test(spending ~ city, var.equal = FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  spending by city
## t = -3.6361, df = 97.999, p-value = 0.0004433
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -77.1608 -22.6745
## sample estimates:
## mean in group Cleveland  mean in group New York
##                251.7948                301.7125
```

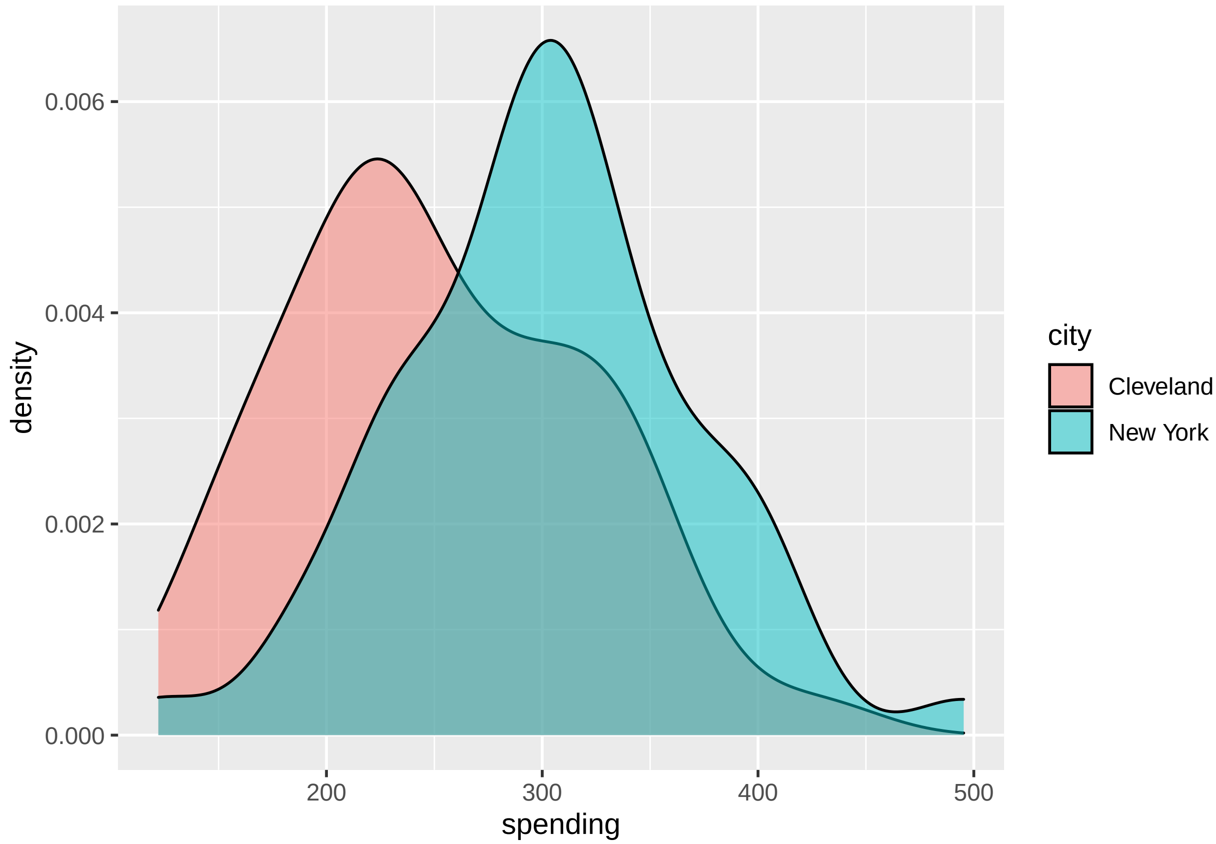
```
plot(t.test(spending ~ city, var.equal = FALSE))
```



Visualization

Density Plot

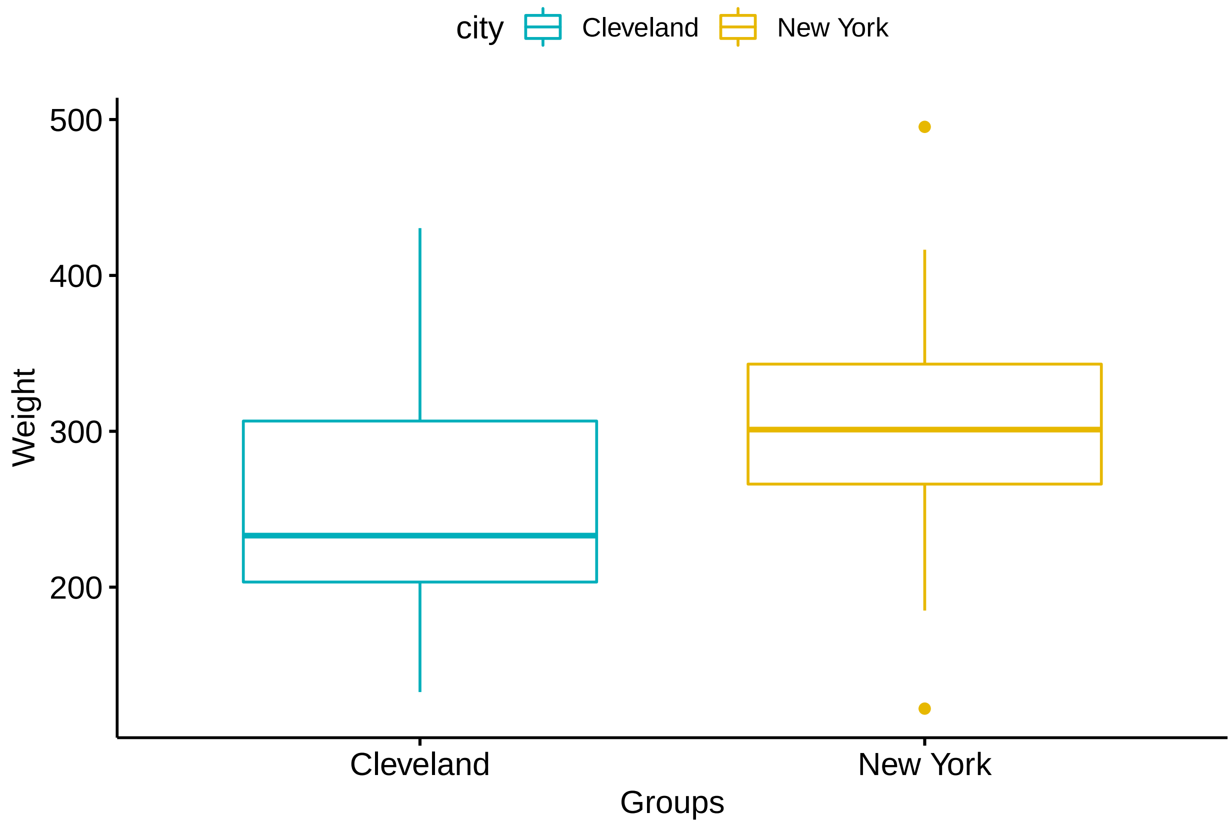
```
library(ggplot2)
#Plot.
ggplot(data, aes(x = spending, fill = city)) + geom_density(alpha = 0.5)
```



Boxplot

Plot weight by group and color by group

```
library("ggpubr")
ggboxplot(
  data,
  x = "city",
  y = "spending",
  color = "city",
  palette = c("#00AFBB", "#E7B800"),
  ylab = "Weight",
  xlab = "Groups"
)
```


city
Cleveland
New York
500
400
300
200
Weight
Cleveland
New York
Groups