

# worksheet01\_walker

Reuben Walker

15 April 2024

## t-test

```
#Sample Vector
num_samples <- 10
#Take a sample of random continues values between [-1,1] with .001 steps (number of steps 2000+1 for -1:1)
cont_distribution <- sample(seq(-1, 1, length.out = 2001), size = num_samples)

#Take a sample of a student-t distribution with parameter nu = 1
#To do this, we use the rt function (pseudo-random numbers from t-distribution) with nu (degrees of freedom) = 1
t_distribution <- rt(num_samples, df=1)

#Take a sample of random discrete values either -1 or 1, replace=TRUE so we can keep sampling
disc_distribution <- sample(c(-1,1), size=num_samples, replace=TRUE)

#Let's just do it for n=5 and plot the histogram of the p values
df = data.frame(continuous=double(),
                t=double(),
                discrete=double()
                )

for (j in 1:10000) {
  num_samples = 5
  cont_distribution_1 <- sample(seq(-1, 1, length.out = 2001), size = num_samples)
  cont_distribution_2 <- sample(seq(-1, 1, length.out = 2001), size = num_samples)
  #Pull the p-value from the t-test
  p_cont <- t.test(cont_distribution_1, cont_distribution_2)$p.value

  t_distribution_1 <- rt(num_samples, df=1)
  t_distribution_2 <- rt(num_samples, df=1)
  p_t <- t.test(t_distribution_1, t_distribution_2)$p.value

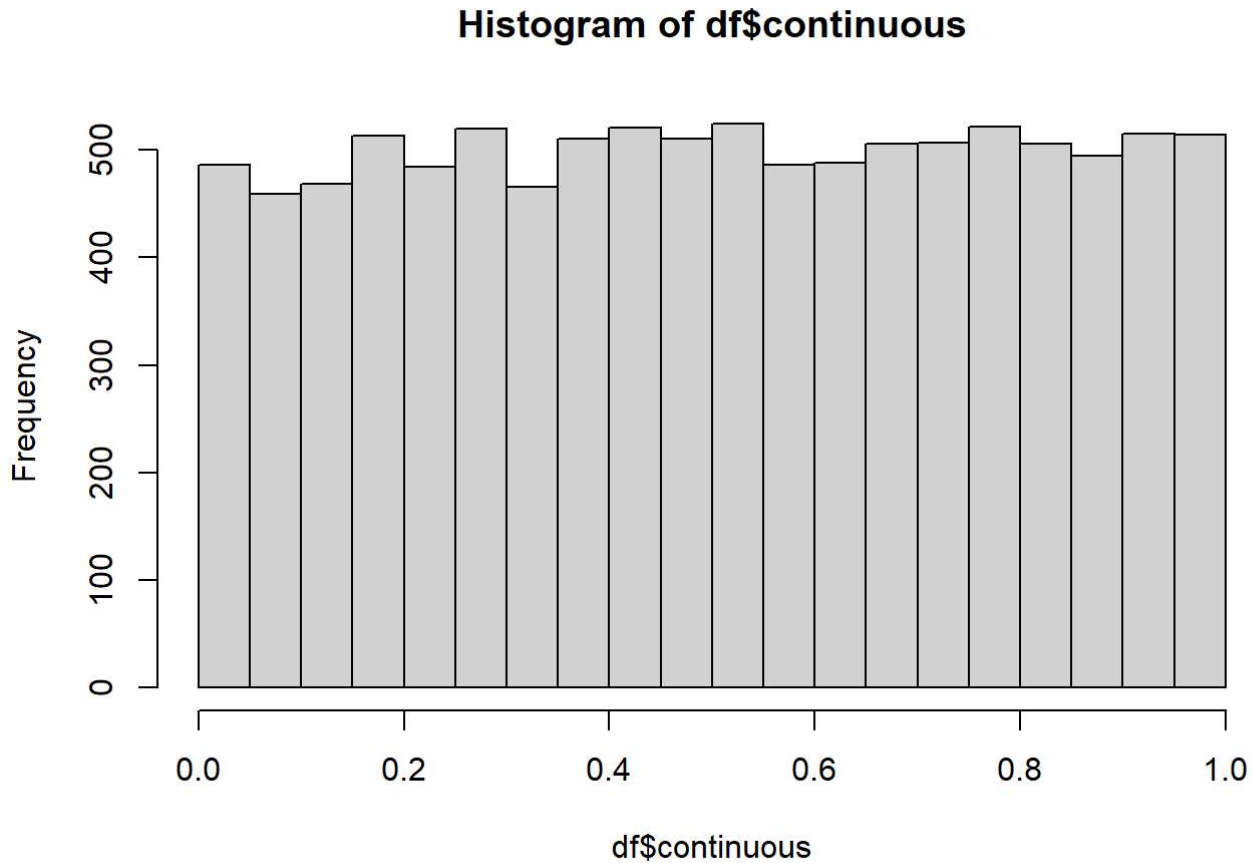
  disc_distribution_1 <- sample(c(-1,1), size=num_samples, replace=TRUE)
  disc_distribution_2 <- sample(c(-1,1), size=num_samples, replace=TRUE)
  #The t.test returns an error when the two values are constant
  #This is because there is a variance term in the denominator
  #Check for null variance
  null_variance <- ((var(disc_distribution_1) == 0) & (var(disc_distribution_2) == 0))
  p_disc <- ifelse(null_variance, NaN, t.test(disc_distribution_1, disc_distribution_2)$p.value)

  #Append a row in the dataframe
  df <- rbind(df, data.frame(continuous=p_cont, t=p_t, discrete=p_disc))
}
```

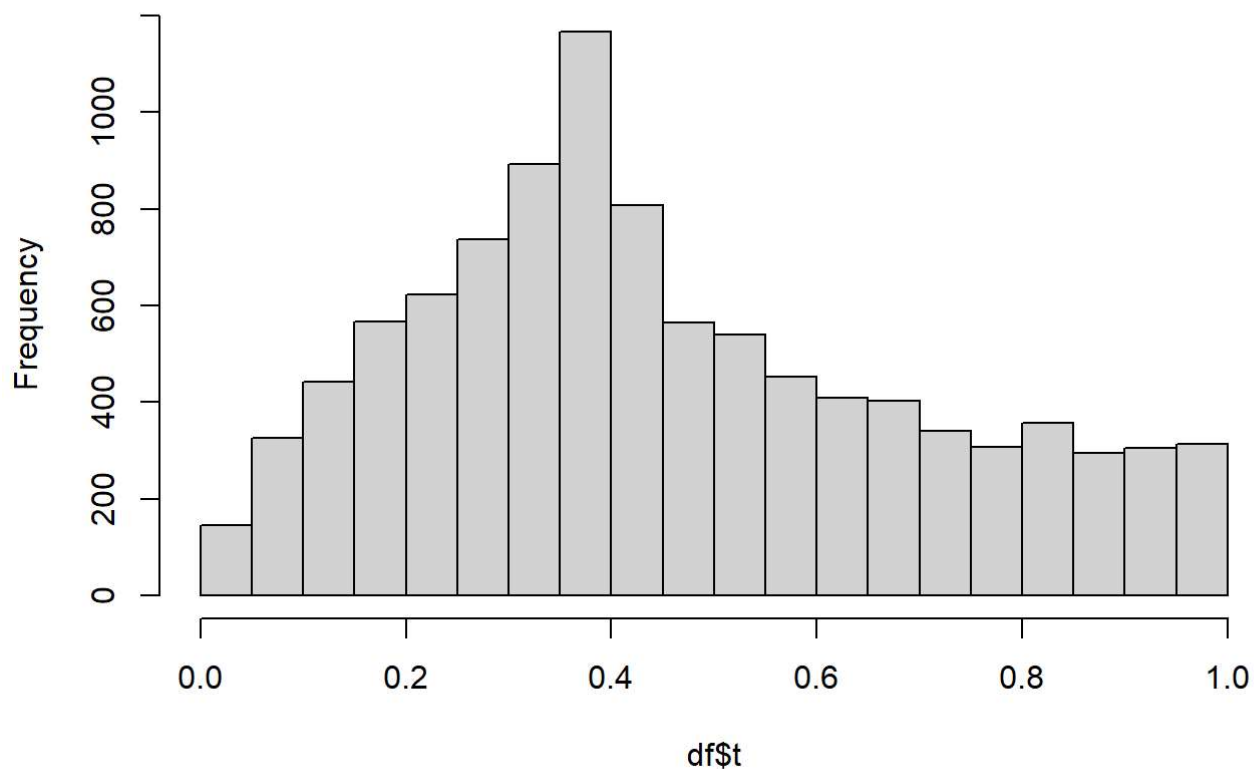
# Histograms for continuous, student-t, and discrete distributions

#Bin width of 0.05, so the furthest left bar is p-value less than 0.05

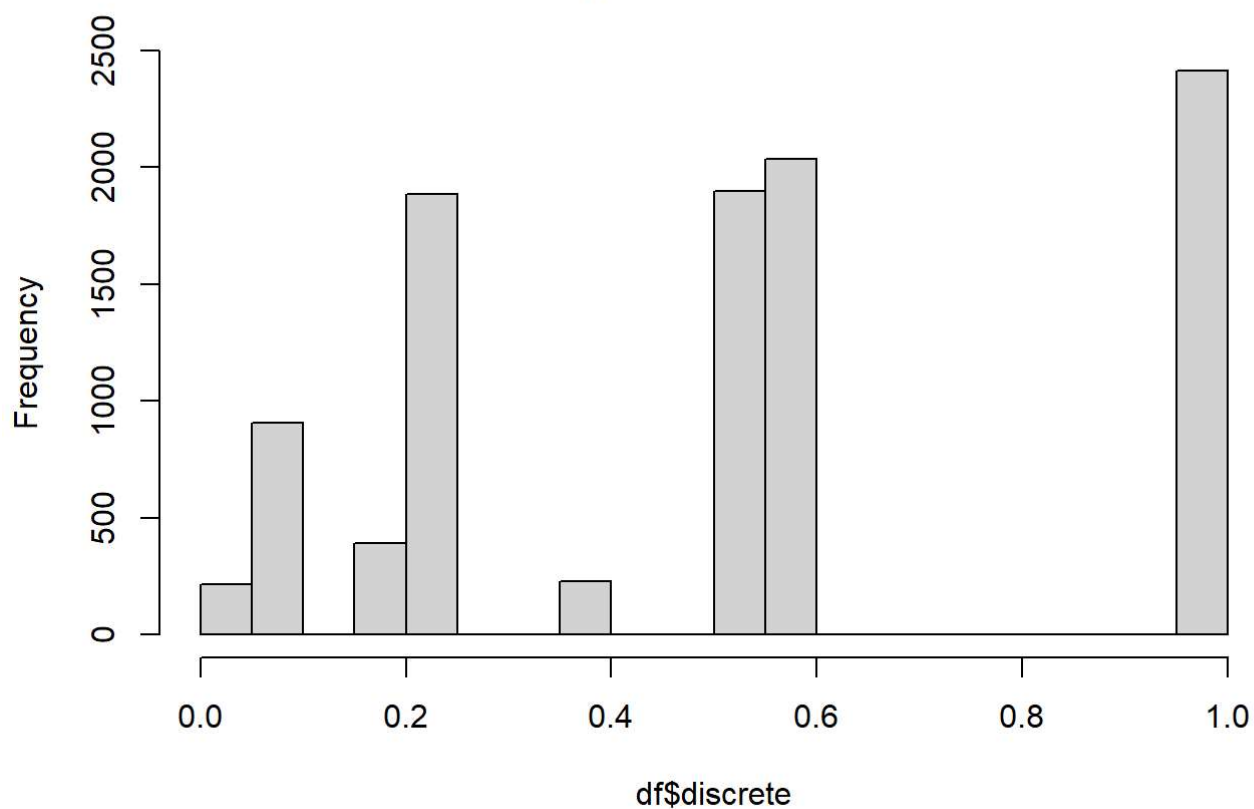
```
hist(df$continuous)#, breaks = seq(from=0, to=1, by=0.05))
```



```
hist(df$t)#, breaks = seq(from=0, to=1, by=0.05))
```

**Histogram of df\$t**

```
hist(df$discrete)#, breaks = seq(from=0, to=1, by=0.05))
```

**Histogram of df\$discrete**

```
#How many is that for each?  
cont_result_0 <- sum(df$continuous < 0.05)/length(df$continuous)  
cont_result_0
```

```
## [1] 0.0486
```

```
#0.04
```

```
t_result_0 <- sum(df$t < 0.05)/length(df$t)  
t_result_0
```

```
## [1] 0.0145
```

```
#0.01
```

```
#Non-NaN subset  
disc_result_0 <- sum(na.omit(df$discrete) < 0.05)/length(na.omit(df$discrete))  
disc_result_0
```

```
## [1] 0.02146655
```

#0.02

*#Now Let's repeat is for all sample sizes*

sampleNumbers &lt;- c(5,10,20,50,100)

cont\_result &lt;- c()

t\_result &lt;- c()

disc\_result &lt;- c()

for (x in sampleNumbers) {

df = data.frame(continuous=double(),

t=double(),

discrete=double()

)

for (j in 1:10000) {

num\_samples = x

cont\_distribution\_1 &lt;- sample(seq(-1, 1, length.out = 20001), size = num\_samples)

cont\_distribution\_2 &lt;- sample(seq(-1, 1, length.out = 20001), size = num\_samples)

*#Pull the p-value from the t-test*

p\_cont = t.test(cont\_distribution\_1, cont\_distribution\_2)\$p.value

t\_distribution\_1 &lt;- rt(num\_samples, df=1)

t\_distribution\_2 &lt;- rt(num\_samples, df=1)

p\_t = t.test(t\_distribution\_1, t\_distribution\_2)\$p.value

disc\_distribution\_1 &lt;- sample(c(-1,1), size=num\_samples, replace=TRUE)

disc\_distribution\_2 &lt;- sample(c(-1,1), size=num\_samples, replace=TRUE)

*#The Lack of variation is much less likely with increasing sample size*

null\_variance &lt;- ((var(disc\_distribution\_1) == 0) &amp; (var(disc\_distribution\_2) == 0))

p\_disc &lt;- ifelse(null\_variance, NaN, t.test(disc\_distribution\_1, disc\_distribution\_2)\$p.value)

*#Append a row in the dataframe*

df &lt;- rbind(df, data.frame(continuous=p\_cont, t=p\_t, discrete=p\_disc))

}

cont\_perc &lt;- sum(df\$continuous &lt; 0.05)/length(df\$continuous)

cont\_result &lt;- append(cont\_result, cont\_perc)

t\_perc &lt;- sum(df\$t &lt; 0.05)/length(df\$t)

t\_result &lt;- append(t\_result, t\_perc)

disc\_perc &lt;- sum(na.omit(df\$discrete) &lt; 0.05)/length(na.omit(df\$discrete))

disc\_result &lt;- append(disc\_result, disc\_perc)

}

*#So more or less regardless of sample size, they're staying the same?*

library(ggplot2)

library(tidyverse)

```
## — Attaching core tidyverse packages — tidyverse 2.0.0 —
## ✓ dplyr      1.1.4      ✓ readr      2.1.5
## ✓ forcats    1.0.0      ✓ stringr    1.5.1
## ✓ lubridate  1.9.3      ✓ tibble     3.2.1
## ✓ purrr      1.0.2      ✓ tidyr      1.3.1
## — Conflicts — tidyverse_conflicts() —
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to be
come errors
```

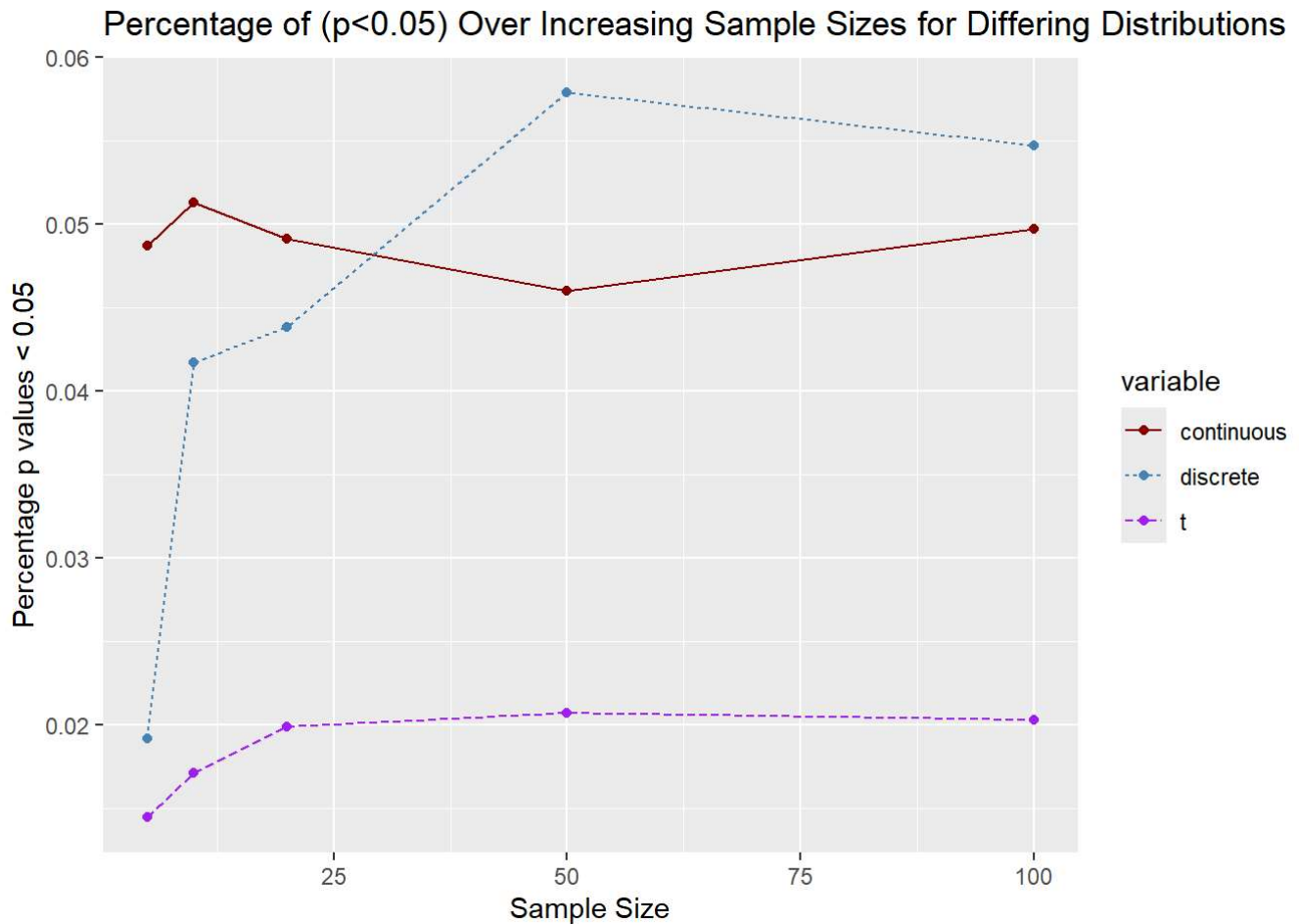
```
#plot(sampleNumbers, cont_result, col='green', ylim=c(0,0.1), main='Probability of p<0.05 (co
ntinuous)')#, type='l'
#plot(sampleNumbers, t_result, type='l', col='blue', ylim=c(0,0.1))
#plot(sampleNumbers, disc_result, type='l', col='red', ylim=c(0,1))

#And all together?
plot_df <- data.frame(samplesize=sampleNumbers,
                      continuous=cont_result,
                      t=t_result,
                      discrete=disc_result)

library(tidyverse)
plot_df2 <- plot_df %>%
  select(samplesize, continuous, t, discrete) %>%
  gather(key = "variable", value = "value", -samplesize)
head(plot_df2)
```

```
##  samplesize  variable  value
## 1           5 continuous 0.0487
## 2          10 continuous 0.0513
## 3          20 continuous 0.0491
## 4          50 continuous 0.0460
## 5         100 continuous 0.0497
## 6           5          t 0.0145
```

```
plot_f <- ggplot(plot_df2, aes(x = samplesize, y = value)) +
  geom_line(aes(color = variable, linetype = variable)) +
  geom_point(aes(color = variable)) +
  scale_color_manual(values = c("darkred", "steelblue", "purple"))
plot_f + ggtitle("Percentage of (p<0.05) Over Increasing Sample Sizes for Differing Distribut
ions") + xlab("Sample Size") + ylab("Percentage p values < 0.05")
```



The continuous and discrete distributions still produce “significant” results in around 5% of cases, even with increasing sample size. The t-distribution produces “significant” results in around 2% of cases.

#3 FALSE: p-values are used to calculate the probability of the null hypothesis given the data. Why: p-values are a measure of surprise and calculated under the assumption that the null hypothesis is true.

TRUE: The significance level  $\alpha$  is the probability of rejecting the null hypothesis when it is true.

FALSE: The Central Limit Theorem only holds if the population from which we are sampling is normally distributed. Why: The Central Limit Theorem states that for large sample sizes, the sample mean is approximately normally distributed, regardless of the distribution.

FALSE: As the sample size gets larger, the standard error of the sampling distribution of the sample mean gets larger as well. Why: The calculation of the standard error includes the root of sample size in the denominator and should decrease with increased sample size.

FALSE: The statistical power of a hypothesis test is the probability of not rejecting the null when  $H_1$  is true. Why: Statistical power is the probability that “one will correctly reject the null hypothesis” if the alternative hypothesis is true. (The alternative hypothesis being that the null hypothesis is false).

FALSE: The statistical power of a hypothesis test is the probability of rejecting  $H_1$  when  $H_1$  is true. Why: Statistical power is the probability that “one will correctly reject the null hypothesis” if the alternative hypothesis is true. (The alternative hypothesis being that the null hypothesis is false).