

Project 2: Investigating the t-test with simulated data

Probability and statistics for modelling 1

Gautham Ganesh

22 October 2021

Task 1. and 9.

```
s <- rnorm(10)
t <- rnorm(10)
print(t.test(s, t))
```

```
##
## Welch Two Sample t-test
##
## data: s and t
## t = -1.3253, df = 12.964, p-value = 0.208
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.107254 0.265422
## sample estimates:
## mean of x mean of y
## -0.40705726 0.01385875
```

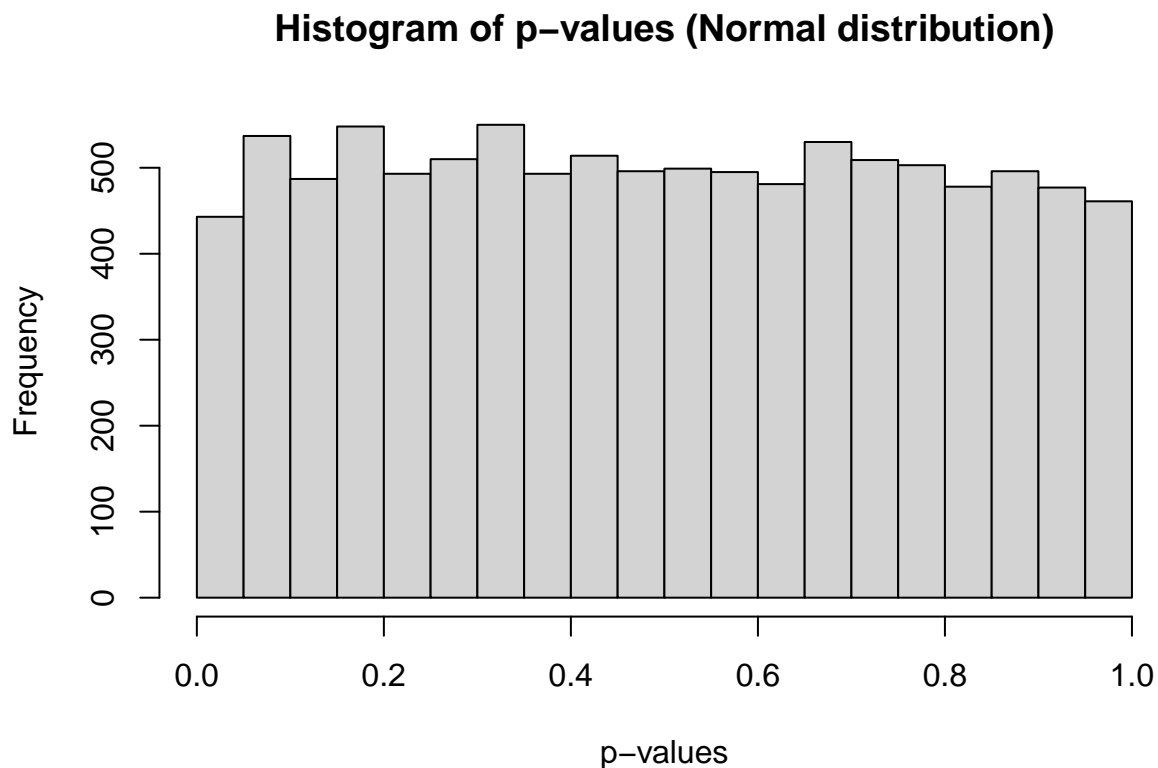
```
p <- rpois(10, 1)
q <- rpois(10, 1)
print(t.test(p, q))
```

```
##
## Welch Two Sample t-test
##
## data: p and q
## t = -0.42857, df = 17.413, p-value = 0.6735
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.5914033 0.3914033
## sample estimates:
## mean of x mean of y
## 0.9 1.0
```

```
p.large <- rpois(1000, 1)
q.large <- rpois(1000, 1)
print(t.test(p.large, q.large))
```

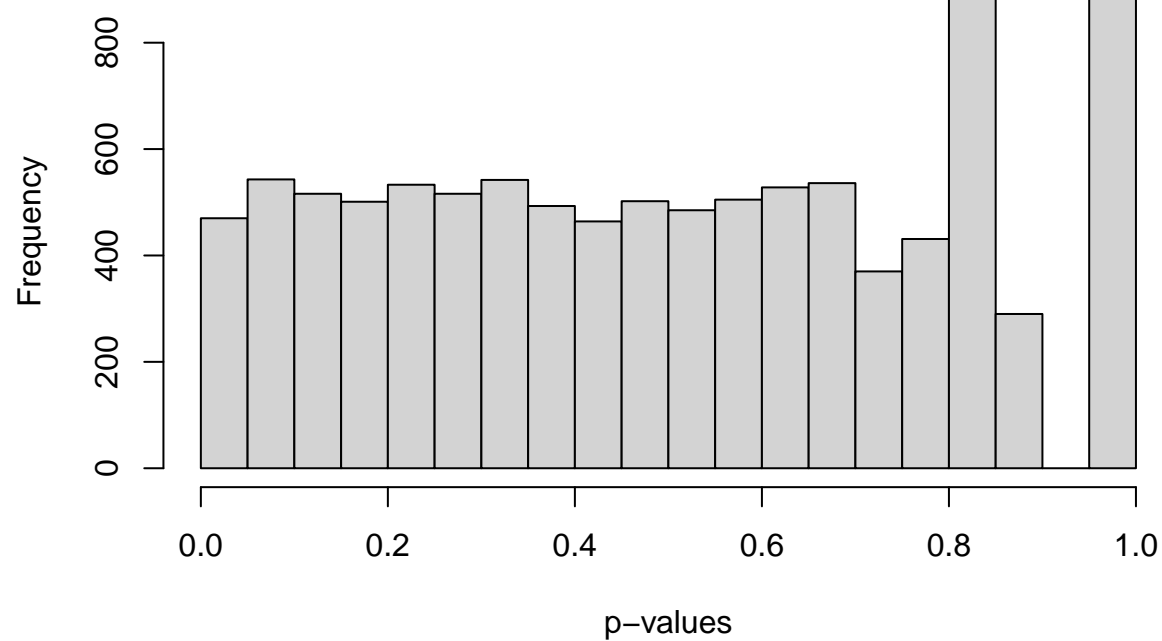
```
##
## Welch Two Sample t-test
##
## data: p.large and q.large
## t = 0.30921, df = 1992, p-value = 0.7572
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.07479401 0.10279401
## sample estimates:
## mean of x mean of y
## 1.008 0.994
```

```
results.normal <- replicate(10000, t.test(rnorm(10), rnorm(10))$p.value)
results.poisson <- replicate(10000, t.test(rpois(10, 1), rpois(10,
1))$p.value)
results.poisson.large <- replicate(10000, t.test(rpois(1000,
1), rpois(1000, 1))$p.value)
hist(results.normal, xlab = "p-values", ylab = "Frequency", main = "Histogram of p-values (Normal distr
```



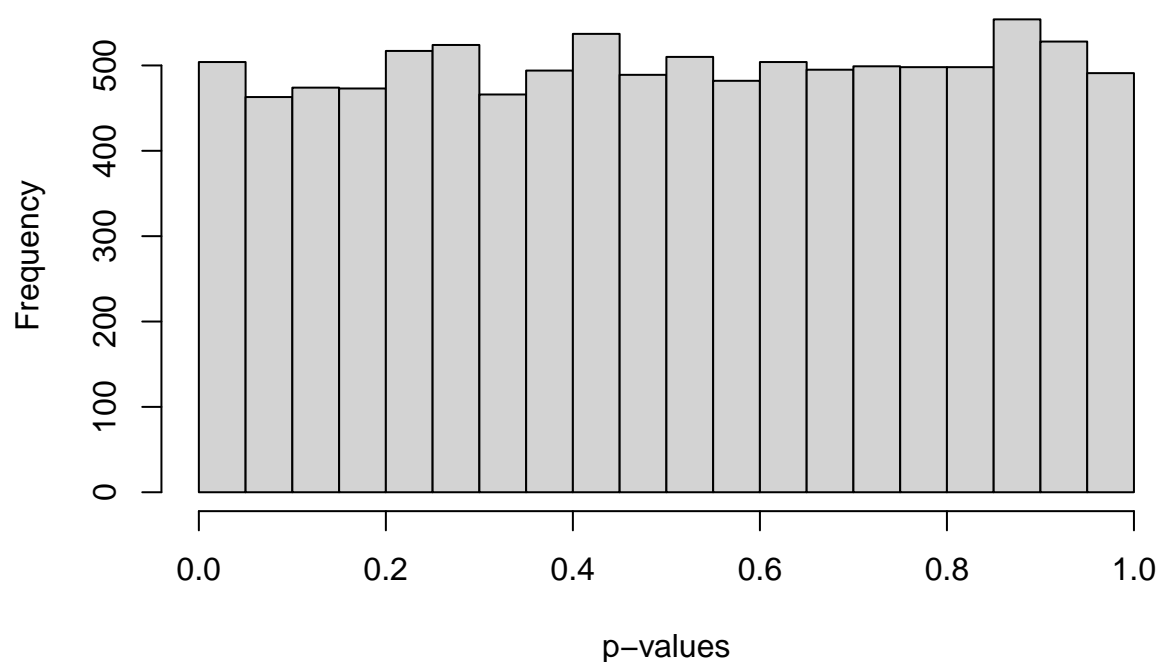
```
hist(results.poisson, xlab = "p-values", ylab = "Frequency",
main = "Histogram of p-values (Poisson distribution)")
```

Histogram of p-values (Poisson distribution)



```
hist(results.poisson.large, xlab = "p-values", ylab = "Frequency",  
     main = "Histogram of p-values (Poisson distribution - large sample)")
```

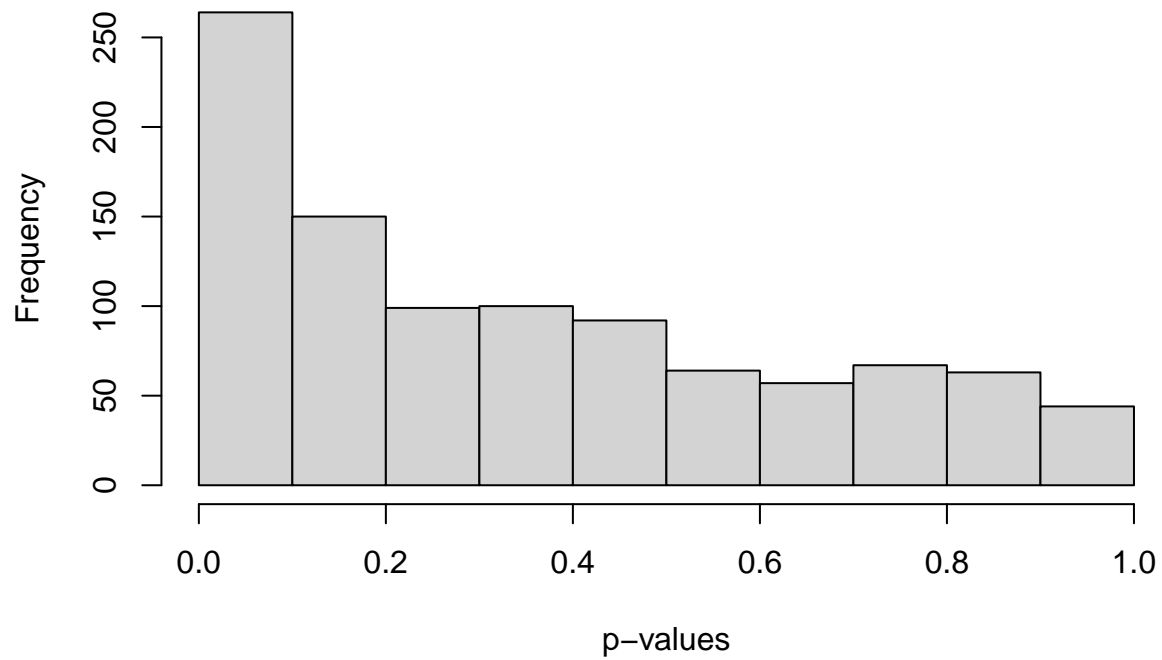
Histogram of p-values (Poisson distribution – large sample)



Task 3.

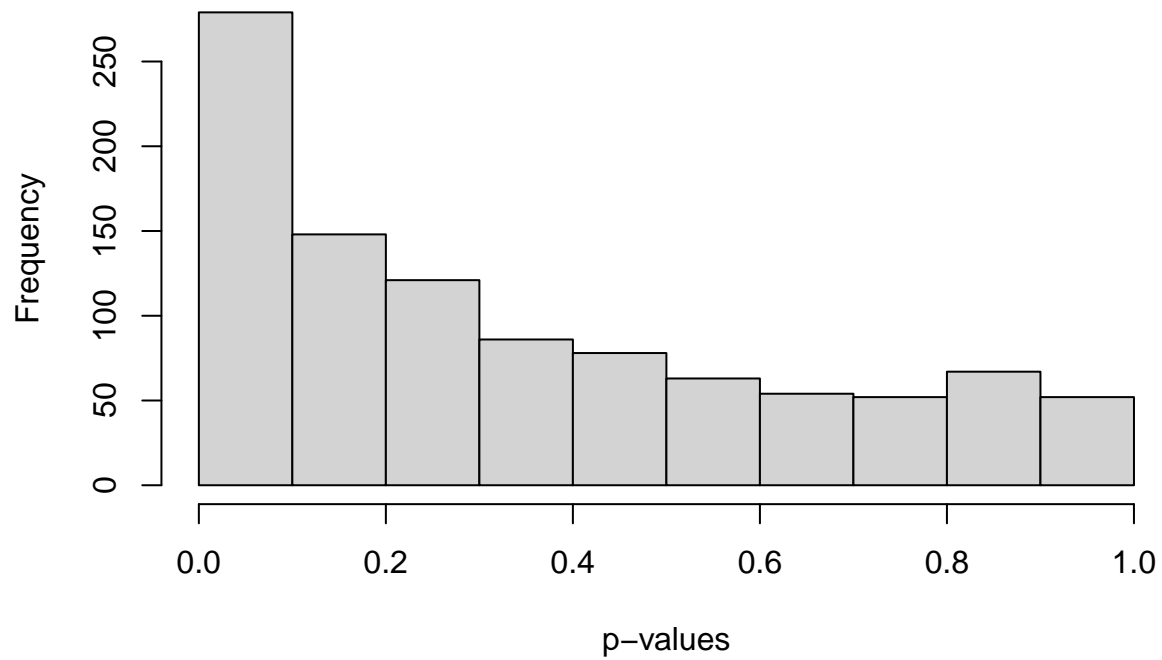
```
test.custom.means <- function(n, x, v, f) {  
  pvalues <- c()  
  for (i in 1:1000) {  
    if (f == 0) {  
      s <- rnorm(n, -x, v)  
      t <- rnorm(n, x, v)  
    } else {  
      s <- rpois(n, 1) - x  
      t <- rpois(n, 1) + x  
    }  
    pvalues <- append(pvalues, t.test(s, t)$p.value)  
  }  
  return(pvalues)  
}  
hist(test.custom.means(10, 0.25, 1, 0), xlab = "p-values", ylab = "Frequency",  
     main = "Histogram of p-values (Normal dist - diff means)")
```

Histogram of p-values (Normal dist – diff means)



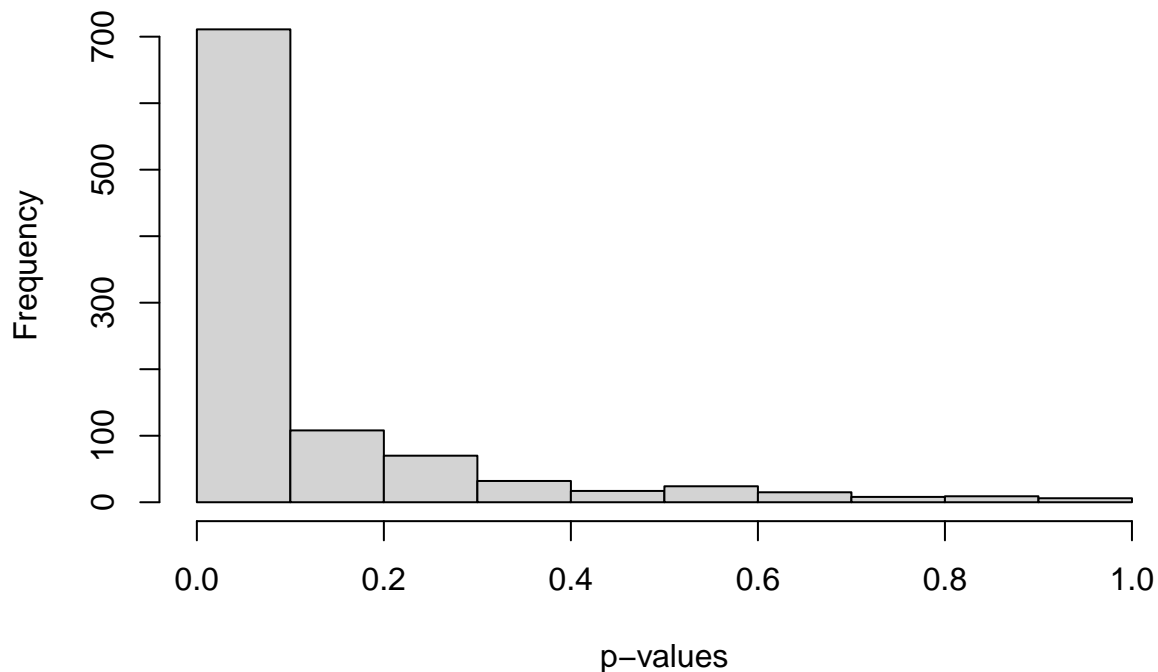
```
hist(test.custom.means(10, 0.25, 1, 1), xlab = "p-values", ylab = "Frequency",  
     main = "Histogram of p-values (Poisson dist - diff means)")
```

Histogram of p-values (Poisson dist – diff means)



```
hist(test.custom.means(1000, 0.05, 1, 1), xlab = "p-values",  
     ylab = "Frequency", main = "Histogram of p-values (Poisson dist - diff means & larger sample)")
```

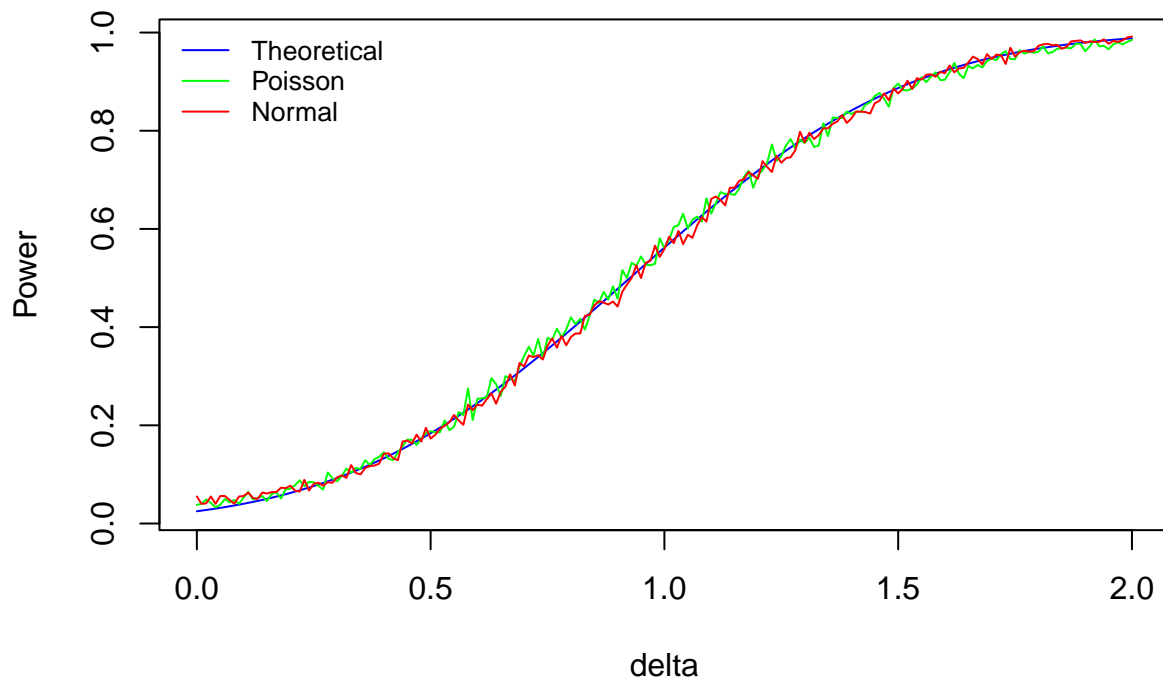
Histogram of p-values (Poisson dist – diff means & larger sample)



Task 4.

```
rpowers.normal <- c()
rpowers.poisson <- c()
tpowers <- c()
delta.list <- seq(0, 2, 0.01)
for (d in delta.list) {
  pvalues.normal <- replicate(1000, t.test(rnorm(10, -d/2),
    rnorm(10, d/2))$p.value)
  pvalues.poisson <- replicate(1000, t.test(rpois(10, 1) -
    d/2, rpois(10, 1) + d/2)$p.value)
  tpowers <- append(tpowers, power.t.test(10, delta = d)$power)
  rpowers.normal <- append(rpowers.normal, sum(pvalues.normal <
    0.05)/1000)
  rpowers.poisson <- append(rpowers.poisson, sum(pvalues.poisson <
    0.05)/1000)
}
plot(delta.list, tpowers, type = "l", col = "blue", xlab = "delta",
  ylab = "Power", main = "Power comparison between t-tests on different distributions")
lines(delta.list, rpowers.poisson, col = "green")
lines(delta.list, rpowers.normal, col = "red")
legend(x = "topleft", legend = c("Theoretical", "Poisson", "Normal"),
  cex = 0.8, lty = c(1, 1, 1), col = c("blue", "green", "red"),
  bty = "n")
```

Power comparison between t-tests on different distributions



Task 5

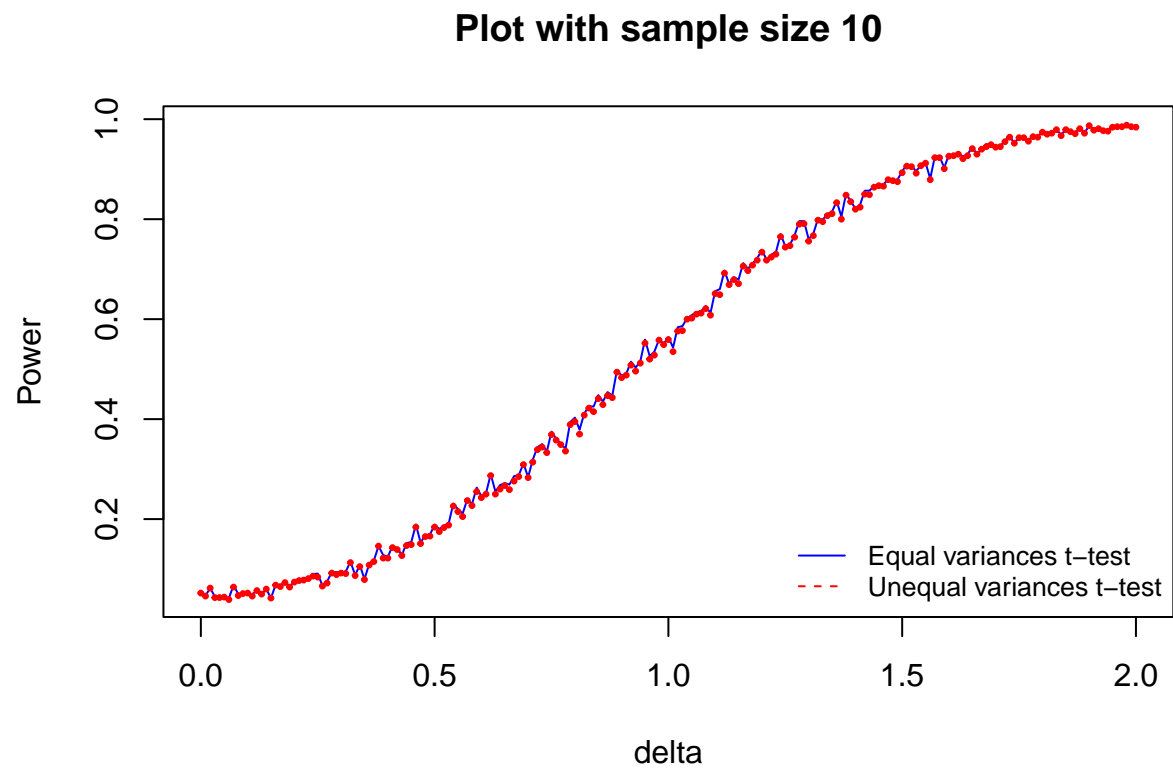
```
for (sample.size in c(10, 100, 1000)) {  
  v.equal.power <- c()  
  v.unequal.power <- c()  
  for (d in delta.list) {  
    pvalues.equal <- c()  
    pvalues.unequal <- c()  
    for (i in 1:1000) {  
      s <- rnorm(sample.size, -d/2)  
      t <- rnorm(sample.size, d/2)  
      pvalues.equal <- append(pvalues.equal, t.test(s,  
        t, var.equal = TRUE)$p.value)  
      pvalues.unequal <- append(pvalues.unequal, t.test(s,  
        t)$p.value)  
    }  
    v.equal.power <- append(v.equal.power, sum(pvalues.equal <  
      0.05)/1000)  
    v.unequal.power <- append(v.unequal.power, sum(pvalues.unequal <  
      0.05)/1000)  
  }  
  plot(delta.list, v.equal.power, type = "l", col = "blue",  
    xlab = "delta", ylab = "Power", main = paste0("Plot with sample size ",  
      sample.size))  
  points(delta.list, v.unequal.power, col = "red", pch = 20,  
    cex = 0.5)  
  legend(x = "bottomright", legend = c("Equal variances t-test",
```



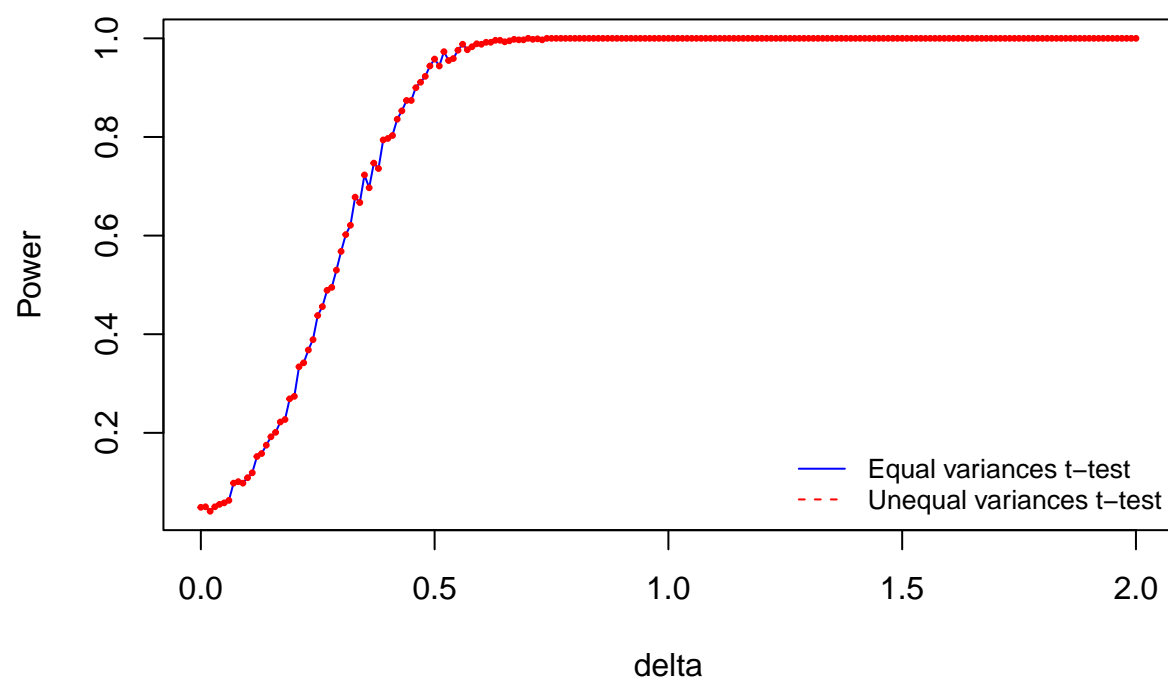
```

    "Unequal variances t-test"), cex = 0.8, lty = c(1, 2),
    col = c("blue", "red"), bty = "n")
}

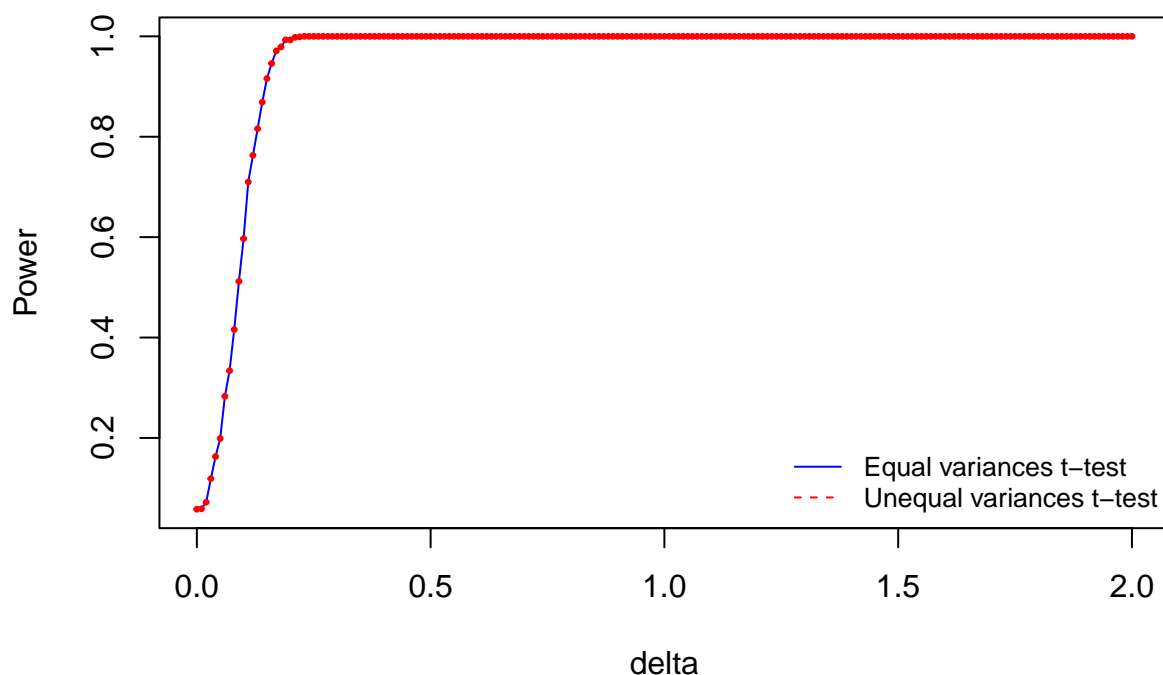
```



Plot with sample size 100



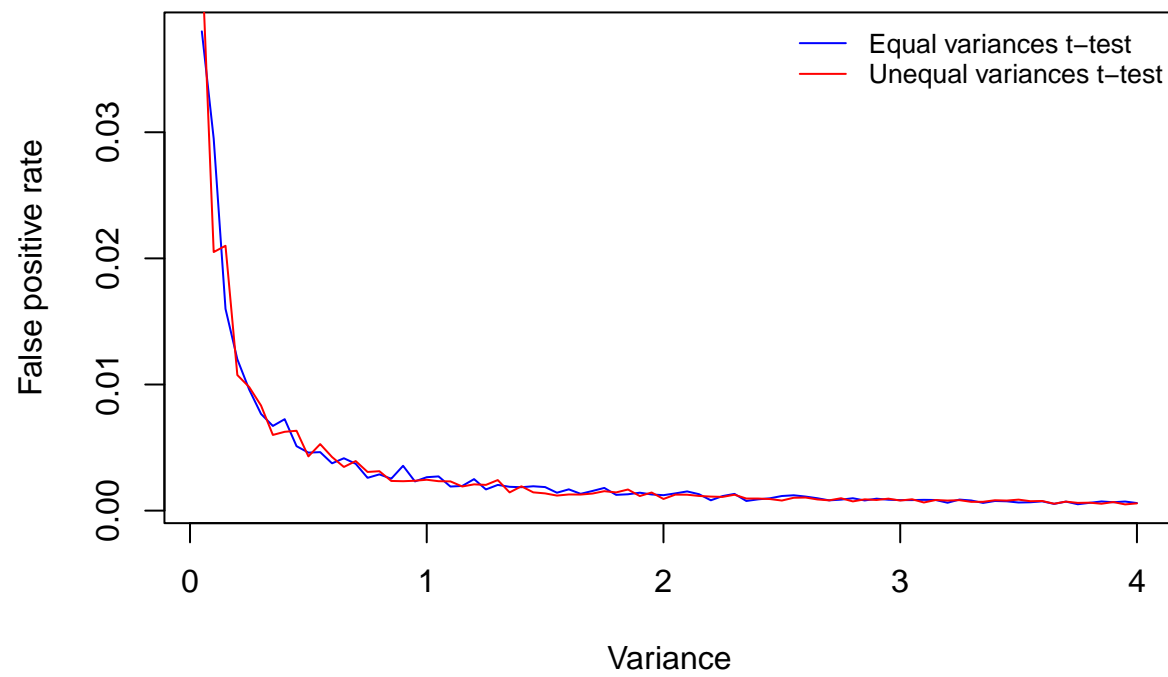
Plot with sample size 1000



Task 6.

```
v.seq <- seq(0.05, 4, 0.05)
f.pos.eq.list <- c()
f.pos.uneq.list <- c()
for (v in v.seq) {
  f.pos.equal <- replicate(1000, t.test(rnorm(100, 0, 0.5),
    rnorm(100, 0, v), var.equal = TRUE)$p.value) < 0.05
  f.pos.unequal <- replicate(1000, t.test(rnorm(100, 0, 0.5),
    rnorm(100, 0, v))$p.value) < 0.05
  f.pos.eq.list <- append(f.pos.eq.list, mean(f.pos.equal))
  f.pos.uneq.list <- append(f.pos.uneq.list, mean(f.pos.unequal))
}
f.pos.eq.rate <- f.pos.eq.list/seq_along(v.seq)
f.pos.uneq.rate <- f.pos.uneq.list/seq_along(v.seq)
plot(v.seq, f.pos.eq.rate, type = "l", col = "blue", xlab = "Variance",
  ylab = "False positive rate", main = "Evolution of false positive rate for a normally-distributed s
lines(v.seq, f.pos.uneq.rate, col = "red")
legend(x = "topright", c("Equal variances t-test", "Unequal variances t-test"),
  cex = 0.8, lty = c(1, 1), col = c("blue", "red"), bty = "n")
```

Evolution of false positive rate for a normally-distributed sample



Task 7.