

PSTAT 220A Homework 2

Jaxon Stuhr

2022-10-19

Question 1

For each of Examples A-J (except C) in the file hw2scan.pdf, state which of the following tests you would apply and (briefly) why: paired t-test, signed rank test, two sample t-test (equal variances or unequal variances), Wilcoxon rank test, other. Conduct appropriate tests.

- Example A: Paired t-test, The dependent variable is interval and the observations are paired (twins)

```
honey = c(19, 12, 9, 17, 24, 24)
no_honey = c(14, 8, 4, 4, 11, 11)
t.test(honey, no_honey, paired = TRUE, alternative = "two.sided")

##
## Paired t-test
##
## data: honey and no_honey
## t = 4.7254, df = 5, p-value = 0.005218
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  4.028033 13.638633
## sample estimates:
## mean difference
##      8.833333
```

According to the two-sample t-test, the twins who received honey had statistically significant increases in hemoglobin compared to their counterparts who did not receive honey. The mean difference was 8.833 with a p-value of .005.

- Example B: Paired t-test: The dependent variable is interval and the observations are paired (two observations for one individual, before and after drug)

```
before = c(6,9,17,22,7,5,5,14,9,7,9,51)
after = c(5,2,0,0,2,1,0,0,0,0,13,0)
t.test(before, after, paired = TRUE, alternative = "two.sided")

##
## Paired t-test
##
## data: before and after
```

According to the two-sample t-test, individuals had statistically fewer premature heartbeats per minute after taking the drug. The mean difference was 11.5 with a p-value of .018.

- ```
oxygen_gel = c(10,15,6,10,11,3,8,8,3,13,10,9,8,9,8,4,10,15,11,5,14,7,8,8,2,13,6,2,7,3)
placebo_gel = c(5,6,4,3,3,5,6,4,4,2,0,7,0,3,2,2,3,6,0,3,-3,1,6,6,8,2,12,24,5,3,3,3,13,4)
wilcox.test(oxygen_gel, placebo_gel)
```

The individuals who took the oxygen gel had significantly higher Oral Hygiene Index, with  $p \ll .05$

- ```
pot_chlorate = c(1.42920, 1.42860, 1.42906, 1.42957, 1.42910, 1.42930, 1.42945)
electrolytic = c(1.42932, 1.42908, 1.42910, 1.42951, 1.42933, 1.42905, 1.42914, 1.42849, 1.42894, 1.42888)
t.test(pot_chlorate, electrolytic, paired = FALSE, alternative = "two.sided", var.equal = TRUE)
```

There was not significant difference between the samples and we cannot reject the null hypothesis that the true mean is not the true density of oxygen. P-value = .5013.

- 2

```
chem_compounds = c(2.30143, 2.29890, 2.29816, 2.30182, 2.29869, 2.29940, 2.29849, 2.29889)
air = c(2.31026, 2.31017, 2.30986, 2.31003, 2.31007, 2.31024, 2.31010, 2.31028)
t.test(chem_compounds, air, paired = FALSE, alternative = "two.sided", var.equal = FALSE)
```

```
##
## Welch Two Sample t-test
##
## data: chem_compounds and air
## t = -21.734, df = 7.1476, p-value = 8.594e-08
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.01180801 -0.00949949
## sample estimates:
## mean of x mean of y
## 2.299473 2.310126
```

There was significant difference between the two samples, and we can reject the null that the shared mean is the true density of nitrogen. P-value $\ll .05$.

- Example G: Two-Sample t-test, equal variances: The dependent variable is interval and the observations are from independent groups, with similar Standard Deviations

```
music = c(35.0, 36.8, 40.2, 46.6, 50.4, 64.2, 83.0, 87.6, 89.2)
no_music = c(28.2, 28.6, 33.0, 34.8, 45.4, 50.8, 52.6, 66.4, 67.8)
t.test(music, no_music, paired = FALSE, alternative = "two.sided", var.equal = TRUE)
```

```
##
## Two Sample t-test
##
## data: music and no_music
## t = 1.5457, df = 16, p-value = 0.1417
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -5.175961 33.042627
## sample estimates:
## mean of x mean of y
## 59.22222 45.28889
```

There was not significant difference between the samples (P-value = .1417), and we cannot reject the null that there is no affect of listening to music on productivity.

- Example H: Two-Sample t-test, equal variances: The dependent variable is interval and the observations are from independent groups, with similar Standard Deviations

```
males = c(40,76,89,106,120,130,150,155,382)
females = c(66,69,94,103,117,391)
t.test(males, females, paired = FALSE, alternative = "two.sided", var.equal = TRUE)
```

```
##
## Two Sample t-test
```

```
##
## data:  males and females
## t = -0.023186, df = 13, p-value = 0.9819
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -125.5648  122.8981
## sample estimates:
## mean of x mean of y
##  138.6667  140.0000
```

There is no significant difference in the number of trials it took male vs female kittens, and we cannot reject the null that they have similar pattern recognition capabilities.

- Example I: Two-Sample t-test, unequal variances: The dependent variable is interval and the observations are from independent groups, with difference Standard Deviations

```
t.test.stats = function(m1, sd1, n1, m2, sd2, n2) {
  t = (m1-m2)/sqrt((sd1^2)/n1 + (sd2^2)/n2)
  df = n1 + n2 - 2
  p = pt(t, df)
  return(p)
}
```

```
high = c(20.6, 6, 200)
low = c(48.1, 14.2, 47)
p = t.test.stats(high[1], high[2], high[3], low[1], low[2], low[3])
print(paste("p-value = ", p))
```

```
## [1] "p-value = 4.62984568537978e-30"
```

There is significant difference in means and we can reject the null that there is no impact of activity levels on the lifetimes of flies. p-value « .05

- Example J: Two-Sample t-test, unequal variances: The dependent variable is interval and the observations are from independent groups, with difference Standard Deviations

```
perm = c(5.1, 2, 200)
temp = c(25.9, 8, 200)
control = c(17.6, 6, 200)
p_perm_temp = t.test.stats(perm[1], perm[2], perm[3], temp[1], temp[2], temp[3])
p_temp_control = t.test.stats(control[1], control[2], control[3], temp[1], temp[2], temp[3])
p_perm_control = t.test.stats(perm[1], perm[2], perm[3], control[1], control[2], control[3])
p_vals = c(p_perm_temp, p_temp_control, p_perm_control)
print(paste("p-values:", p_vals[1], p_vals[2], p_vals[3]))
```

```
## [1] "p-values: 2.46181728966564e-126 7.9603510728832e-28 3.28634024834041e-96"
```

There is significant difference between each sample and the other two, and we can reject the nulls that:

1. There is no impact of Permanent O2 exposure on fly-lifetime relative to a control
2. There is no impact of Temporary O2 exposure on fly-lifetime relative to a control
3. There is equal impact of Permanent and Temporary O2 exposure on fly-lifetime

Question 2

For Example C in the file hw2scan.pdf, you may apply one of the above simple tests to the combined data across experiments. Is this a reasonable approach for this particular data? Why or why not? Suppose this is not reasonable, can you suggest an alternative approach?

I will apply a Paired t-test as the dependent variable is interval and the observations are paired (pairs of rats from the same litter).

```
treatment = c(689, 656, 668, 660, 679, 663, 664, 647, 694, 633, 653, 707, 740, 745, 652, 649, 676, 699,
control = c(657, 623, 652, 654, 658, 646, 600, 640, 605, 635, 642, 669, 650, 651, 627, 656, 642, 698, 6
t.test(treatment, control, paired = TRUE, alternative = "two.sided")
```

```
##
## Paired t-test
##
## data: treatment and control
## t = 5.7695, df = 22, p-value = 8.365e-06
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
## 23.11535 49.05856
## sample estimates:
## mean difference
## 36.08696
```

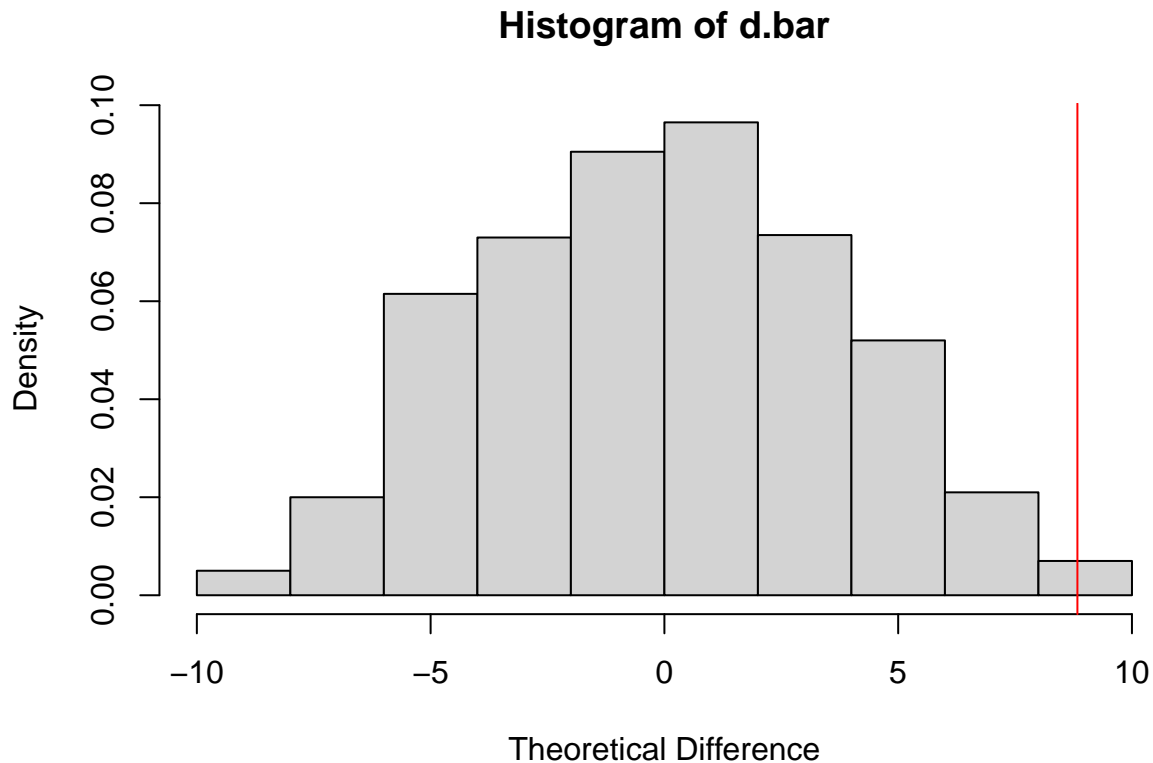
Based on a limited sample of the combined data, we can reject the null that the mean weights of isolated cortex and social cortex are the same. P-Value « .05

This approach is fairly reasonable given the visually consistent % change in cortex weight shown in figure (b). It is likely, however, that conditions were not consistent from one experiment to the next, making aggregating the data sub-optimal. An alternative would be

Question 3

Apply the permutation test with statistic $t = \bar{x} - \bar{y}$ to Example A, where x and y respectively refer to measurements with and without honey. Construct bootstrap confidence intervals for the difference of population means.

```
dif = honey - no_honey
t = mean(dif)
n = length(dif)
dif.perm = matrix(abs(dif), n, 1000)
dif.perm = dif.perm*sign(runif(1000*n)-.5)
d.bar = apply(dif.perm, 2, mean)
hist(d.bar, freq = F, xlab = "Theoretical Difference")
abline(v = mean(dif), col = "red")
```



```
p_val = sum(d.bar>=t)/length(d.bar)
print(p_val)
```

```
## [1] 0.014
```

Based on the above permutation test, we can see that the true mean of the difference is far from the mean of the theoretical distribution if there were no difference between samples. P-value = 0.012

```
honey.sample = matrix(dif, length(dif), 1000)
honey.sample = apply(honey.sample, 2, sample, r = T)
honey.md = apply(honey.sample, 2, median)
print(quantile(honey.md, c(0.025, 0.975)))
```

```
## 2.5% 97.5%
## 4.5 13.0
```

From the bootstrap method, we get a 95% confidence interval of (4.5, 13.0) which represent the difference in means of hemoglobin increases for subjects with vs. without honey treatment.

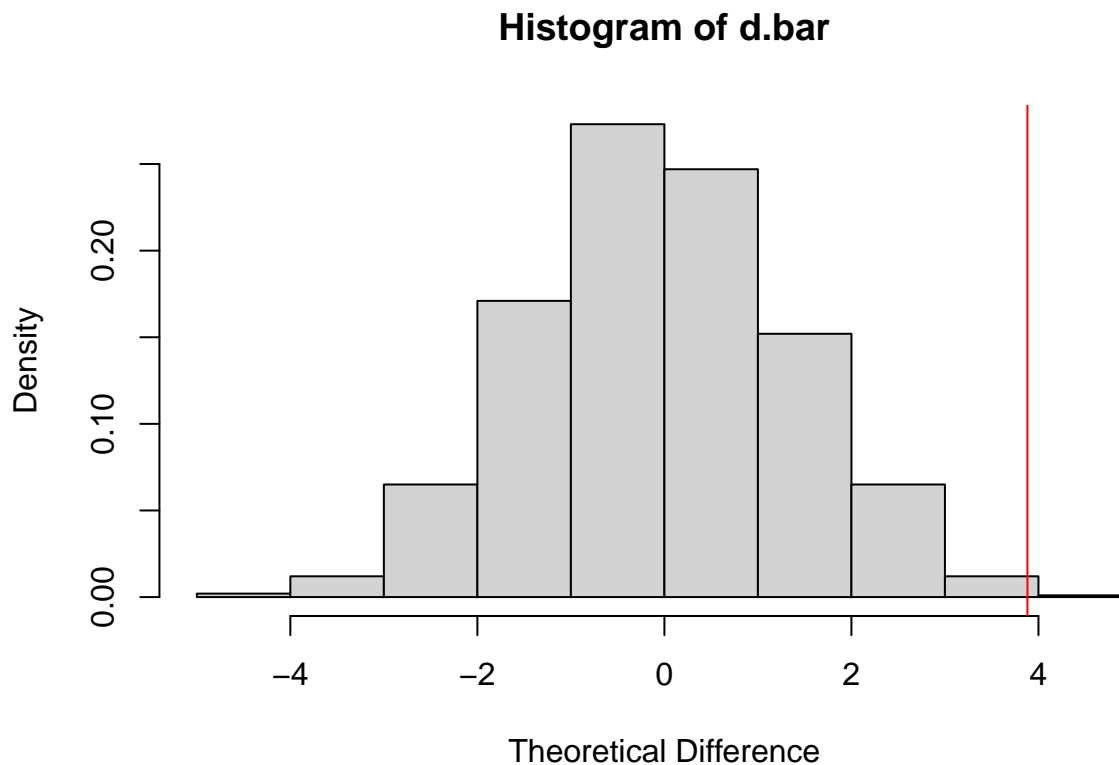
Question 4

Apply the permutation test with statistic $t = \bar{x} - \bar{y}$ to Example D, where x and y respectively refer to measurements for the oxygen gel and placebo gel groups. Construct bootstrap confidence intervals for the difference of population means.

```

dif = oxygen_gel - placebo_gel
t = mean(dif)
n = length(dif)
dif.perm = matrix(abs(dif), n, 1000)
dif.perm = dif.perm*sign(runif(1000*n)-.5)
d.bar = apply(dif.perm, 2, mean)
hist(d.bar, freq = F, xlab = "Theoretical Difference")
abline(v = mean(dif), col = "red")

```



```

p_val = sum(d.bar>=t)/length(d.bar)
print(p_val)

```

```
## [1] 0.001
```

Based on the above permutation test, we can see that the true mean of the difference is far from the mean of the theoretical distribution if there were no difference between samples. P-value = .001

```

oxygen.sample = matrix(dif, length(dif), 1000)
oxygen.sample = apply(oxygen.sample, 2, sample, r = T)
oxygen.md = apply(oxygen.sample, 2, median)
print(quantile(oxygen.md, c(0.025, 0.975)))

```

```
## 2.5% 97.5%
## 2 7
```

From the bootstrap method, we get a 95% confidence interval of (2, 7) which represent the difference in means of oral hygiene index for subjects who took oxygen gels vs those who did not.

Question 5

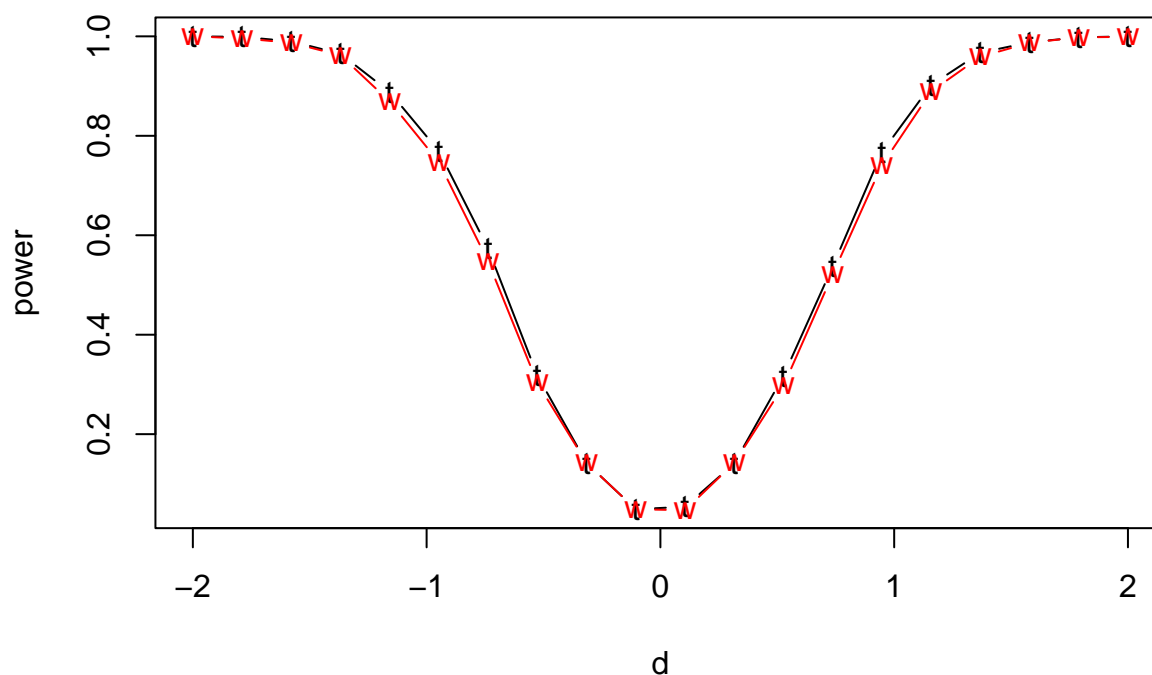
Discuss pros and cons between t and Wilcoxon tests. Conduct simulations to check your claims. Specifically, modify the simulation code i used in class to check the robustness of t test to departure from the normality assumption by generating sample from non-Gaussian distributions and compare powers of the t and Wilcoxon tests in these situations.

T-Test assumes interval data and a normal distribution and equal variances between groups. On the other hand, Wilcoxon tests use ranks instead of raw data and don't require distribution assumptions. The Wilcoxon test is more flexible, but less powerful when the assumptions of the T-Test are met.

```
### this is the sample code from class with a couple of small modifications
n <- 10 # number of point in a sample simulated dataset
nsim <- 1000 # number of simulations per mean
d <- seq(-2,2,len=20) # sdquence of means to simulate
pt <- pw <- matrix(NA, length(d), nsim) # NA Matrix to fill in with p-values of each of 1000 tests for
for (j in 1:length(d)) { # for each mean in d
  for (i in 1:nsim) { # run nsim simulations
    y <- rnorm(n, mean=d[j], sd=1) # generate a random normal dataset of n points with a mean of d[j]
    pt[j,i] <- t.test(y)$p.value # add p-val of t-test for dataset to matrix
    pw[j,i] <- wilcox.test(y)$p.value # add p-val of wilcox-test for dataset to matrix
  }
}
powerw <- apply(pw<.05,1,mean) # calculate power of wilcox-tests
powerw <- apply(pw<.05,1,mean) # calculate power of wilcox-tests
print(rbind(powerw,powerw))
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
## powerw  1 0.999 0.990 0.965 0.888 0.769 0.573 0.318 0.142 0.049 0.054 0.141
## powerw  1 0.996 0.987 0.962 0.869 0.747 0.547 0.303 0.142 0.049 0.047 0.142
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20]
## powerw 0.317 0.537 0.767 0.902 0.967 0.988 0.998 1.000
## powerw 0.297 0.521 0.741 0.889 0.959 0.987 0.998 0.999
```

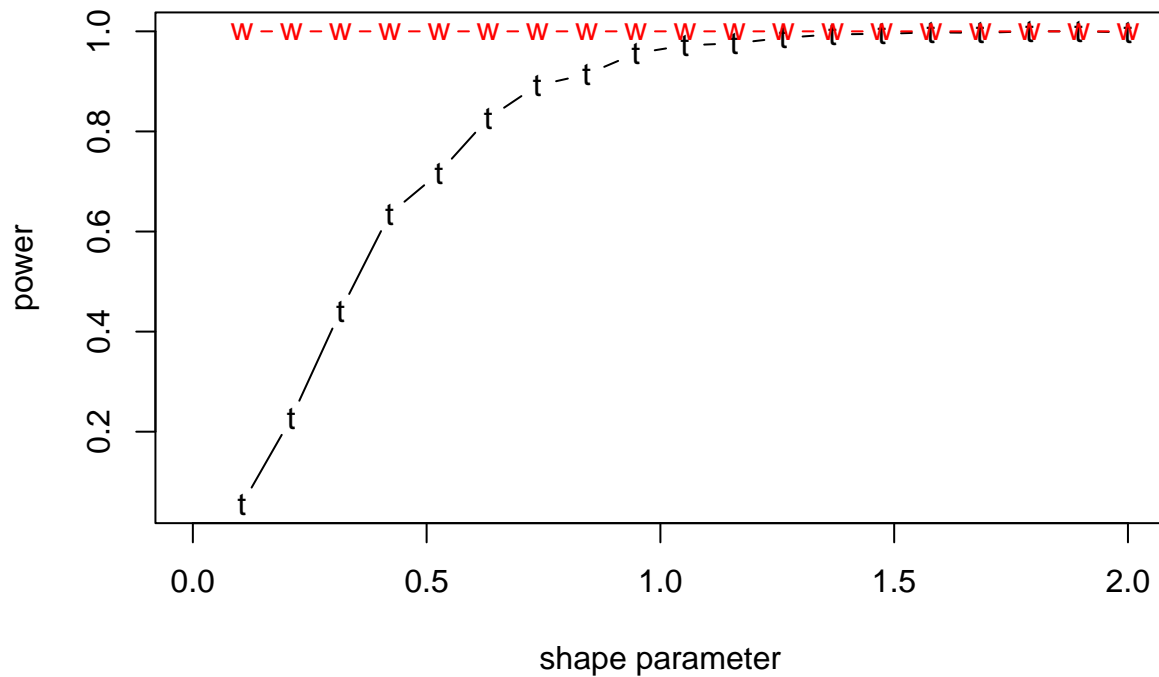
```
plot(d, powerw, type="b", pch="t", xlab="d", ylab="power", title = "Normal Distribution Powers")
points(d, powerw, type="b", pch="w", col="red")
```

```
### this switches the sample code to use a gamma distribution with varied shape parameters and a fixed n
n <- 10 # number of point in a sample simulated dataset
nsim <- 1000 # number of simulations per mean
shapes <- seq(0,2,len=20) # sdquence of means to simulate
pt <- pw <- matrix(NA, length(shapes), nsim) # NA Matrix to fill in with p-values of each of 1000 test
for (j in 1:length(shapes)) { # for each mean in d
  for (i in 1:nsim) { # run nsim simulations
    y <- rgamma(n, shape = shapes[j]) # generate a random gamma distribution dataset of n points with
    pt[j,i] <- t.test(y)$p.value # add p-val of t-test for dataset to matrix
    pw[j,i] <- wilcox.test(y)$p.value # add p-val of wilcox-test for dataset to matrix
  }
}
power <- apply(pt<.05,1,mean) # calculate power of t-tests
powerw <- apply(pw<.05,1,mean) # calculate power of wilcox-tests
print(rbind(power,powerw))
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
## power  NA 0.055 0.227 0.441 0.635 0.717 0.826 0.892 0.914 0.956 0.972 0.976
## powerw NA 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
##      [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20]
## power  0.987 0.994 0.995 0.998 0.997      1      1 0.998
## powerw 1.000 1.000 1.000 1.000 1.000      1      1 1.000
```

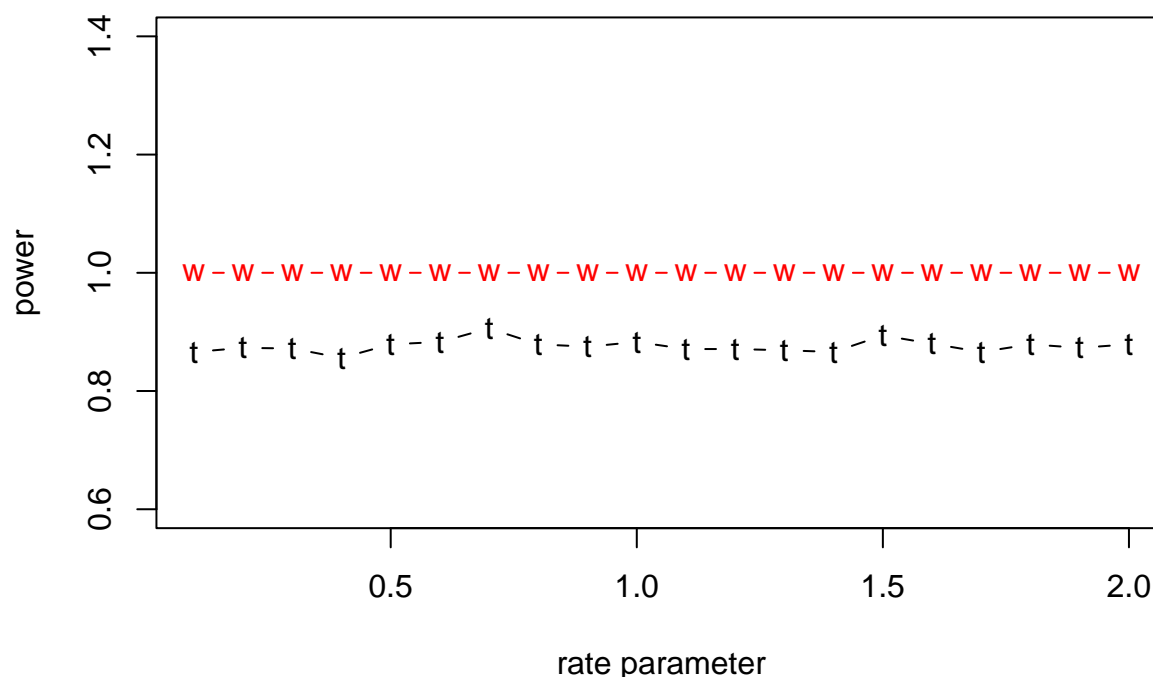
```
plot(shapes, power, type="b", pch="t", xlab="shape parameter", ylab="power", title = "Gamma Distribut.
points(shapes, powerw, type="b", pch="w", col="red")
```



```
### this switches the sample code to use a gamma distribution with a varied rate and a fixed shape of s
n <- 8 # number of point in a sample simulated dataset
nsim <- 1000 # number of simulations per mean
rates <- seq(0.1,2,len=20) # sdquence of means to simulate
pt <- pw <- matrix(NA, length(rates), nsim) # NA Matrix to fill in with p-values of each of 1000 tests
for (j in 1:length(rates)) { # for each mean in d
  for (i in 1:nsim) { # run nsim simulations
    y <- rexp(n, rate = rates[j]) # generate a random gamma distribution dataset of n points with a sh
    pt[j,i] <- t.test(y)$p.value # add p-val of t-test for dataset to matrix
    pw[j,i] <- wilcox.test(y)$p.value # add p-val of wilcox-test for dataset to matrix
  }
}
powerw <- apply(pw<.05,1,mean) # calculate power of t-tests
powerw <- apply(pw<.05,1,mean) # calculate power of wilcox-tests
print(rbind(powerw,powerw))
```

```
##           [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
## powerw 0.866 0.873 0.872 0.856 0.879 0.883 0.906 0.879 0.875 0.883 0.871 0.871
## powerw 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000
##           [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20]
## powerw 0.869 0.866 0.894 0.88 0.866 0.879 0.873 0.879
## powerw 1.000 1.000 1.000 1.00 1.000 1.000 1.000 1.000
```

```
plot(rates, powerw, type="b", pch="w", col = "red", xlab="rate parameter", ylab="power", title = "Expor
points(rates, powerw, type="b", pch="t", col="black")
```



Above I conducted simulations of the performance of t-test and wilcoxon-tests for a normal distribution as well as gamma distributions (with varied shape paramters) and exponential distributions (with varied rate parameters).

For the Normal Distribution, we observe that for means close to zero, the t-test is slightly more powerful than the wilcoxon tests. As the $|\text{mean}|$ increases, the two tests both approach a power of 1.

For the gamma distribution with a varied shape parameters, we observe that for shape parameters close to zero, the t-test performs quite poorly while the wilcoxon test performs very well. As the shape parameter increases, the distribution appears more normal, and the power of the t-test approaches 1 as well.

For the exponential distribution with varied rate parameter, the t-test performs fairly well with power $\sim .9$, but the wilcoxon test performs better with a power of ~ 1 .

These observations support the discussion in the beggining of this question, as the t-test performed better when it's assumptions were met, but the wilcoxon test was much more flexible.

Question 6

Write a R function for Z-test that compares the proportions from two independent groups. Include an option so that the user can specify one-sided or two-sided alternative hypothesis. Also include an option so that the user can specify the confidence level for confidence intervals. Output Zstatistic, p-value, estimates of proportion for each group, estimate of the difference and confidence intervals. Apply this function to Florida death penalty data.

```

z.stat = function(p1, n1, p2, n2, alternative = c("two.sided", "less", "greater"), confidence.lvl = .95)

  alternative <- match.arg(alternative)

  p = (p1+p2)/(n1+n2)
  z_stat = (p1/n1 - p2/n2)/sqrt(p*(1-p)*(1/n1 + 1/n2))

  if (alternative == "less") {
    c_int = c(-Inf, z_stat + qnorm(confidence.lvl, mean = 0, sd = 1))
    p_val = pnorm(z_stat, mean = 0, sd = 1)
  }
  else if (alternative == "greater") {
    c_int = c(z_stat - qnorm(confidence.lvl, mean = 0, sd = 1), Inf)
    p_val = pnorm(z_stat, mean = 0, sd = 1, lower.tail = FALSE)
  }
  else {
    alpha = (1-confidence.lvl)
    c_int = qnorm(1-alpha/2, mean = 0, sd = 1)
    c_int = z_stat+c(-c_int, c_int)
    p_val = 2*pnorm(-abs(z_stat), mean = 0, sd = 1)
  }

  return_vals = list(z_stat, p_val, c_int)

  return(return_vals)
}

```

```

sample_z_stat = z.stat(19, 160, 17, 166, alternative = "two.sided")

```

Applying our Z-test to the Florida death-penalty data, we cannot reject the null that there is no correlation between race and death penalty. (Z-Stat = 0.471, P-Value = .639, CI = (-1.489, 2.430))

Question 7

Write a R function for goodness-of-fit tests when category probabilities are completely specified. Applies your function to the linkage study of the tomato and the hour of birth examples.

```

gof_test = function(x, p, lower.tail = TRUE) {
  test_stat = 0
  for (i in 1:length(x)) {
    test_stat = test_stat + ((x[i] - p[i]*sum(x))^2)/(p[i]*sum(x))
  }
  df = length(x) - 1
  if (lower.tail == FALSE) {
    p.val = pchisq(test_stat, df, lower.tail = FALSE)
  } else {
    p.val = pchisq(test_stat, df, lower.tail = TRUE)
  }
  return(c(test_stat, df, p.val))
}

```

Test on Tomato Data:

```
observed_counts = c(926, 288, 293, 104)
theoretical_probs = c(9,3,3,1)/16
gof_test(observed_counts, theoretical_probs, lower.tail = FALSE)
```

```
## [1] 1.4687220 3.0000000 0.6895079
```

From the goodness-of-fit test we get a p-val = .689 and cannot reject the null that the tomatoes are inconsistent with Mendel's laws.

Test on hour-of-birth data:

```
observed_counts = c(52,73,89,88,68,47,58,47,48,53,47,34,21,31,
40,24,37,31,47,34,36,44,78,59)
theoretical_probs = rep(1,24)/24
gof_test(observed_counts, theoretical_probs, lower.tail = FALSE)
```

```
## [1] 1.627757e+02 2.300000e+01 4.990700e-23
```

From the goodness-of-fit test we get a p-val « .05 and can reject the null that all birth-hours have equal probability.

Question 8

Each individual in a random sample of high school and college students was cross-classified with respect to both political views and marijuana usage. Does the data support the hypothesis that political views and marijuana usage level are independent within the population?

Two categorical variables, will use chi-squared to check for correlation.

```
liberal = c(479, 173, 119)
conservative = c(214, 47, 15)
other = c(172, 45, 85)
pot_and_politics = matrix(data = c(liberal, conservative, other), nrow = 3, ncol = 3)
chisq.test(pot_and_politics)
```

```
##
## Pearson's Chi-squared test
##
## data: pot_and_politics
## X-squared = 64.654, df = 4, p-value = 3.043e-13
```

From the Chi-Squared test, we see that there is a strong correlation between political identity and marijuana use. (P-Value « .05)

Question 9

A certain type of flashlight is sold with the four batteries included. A random sample of 150 flashlights is obtained and the number of defective batteries in each is determined. Let X be the number of defective batteries in a randomly selected flashlight. Test the hypothesis that the distribution of X is Binomial.

```

num_defective = c(0, 1, 2, 3, 4)
freq = c(26, 51, 47, 16, 10)
building_array = array(0, dim = length(freq))
all_obs = array(0, dim = sum(freq))
all_obs[1:building_array[1]] = num_defective[1]

for (i in 1:5) {
  building_array[i] = sum(freq[1:i])
}

for (i in 2:5) {
  all_obs[(building_array[i-1] + 1):building_array[i]] = num_defective[i]
}

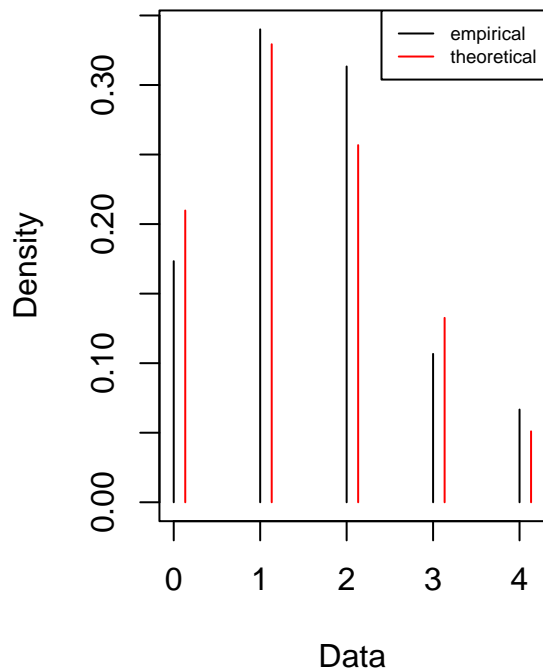
fit.binom = fitdist(data = c(all_obs), dist="binom",
                    fix.arg=list(size = sum(freq)),
                    start=list(prob = 0.2))

theoretical_freq = c(0,0,0,0,0)
for (i in 1:5) {
  theoretical_freq[i] = dbinom(num_defective[i], 150, fit.binom$estimate)
}

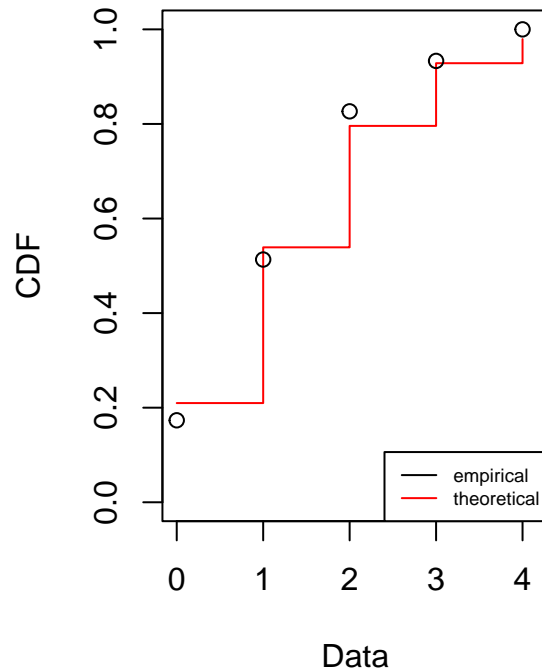
plot(fit.binom)

```

Emp. and theo. distr.



Emp. and theo. CDFs



```
gof_test(freq, theoretical_freq)
```

```
## [1] 4.3520236 4.0000000 0.6395441
```

The empirical data appears to fit a binomial distribution quite well with a parameter estimate of $\theta = 0.0103$. Comparing the theoretical proportions to the observed failure frequencies with a goodness-of-fit chi-squared test, we get a p-value = .639, and we cannot reject the null hypothesis that the observed data was drawn from a binomial distribution.

Question 10

Each of 325 individuals participating in a certain drug program was categorized both with respect to the presence or absence of hypoglycemia and with respect to mean daily dosage of insulin. Does the data support the claim that the presence/absence of hypoglycemia is independent of insulin dosage?

```
present = c(4, 21, 28, 15, 12)
absent = c(40, 74, 59, 26, 46)
total = present + absent
theoretical_absent = (sum(absent)/sum(total)) * total

m_absent <- matrix(0, nrow = 5, ncol = 2)
m_absent[1:5,1] = theoretical_absent
m_absent[1:5,2] = absent

chisq.test(m_absent)
```

```
##
## Pearson's Chi-squared test
##
## data:  m_absent
## X-squared = 1.5058, df = 4, p-value = 0.8256
```

We can test if hypoglycemia is independent of insulin dosage by projecting theoretical hypoglycemia counts based on the total proportion of individuals with hypoglycemia to those without, and multiplying this proportion by the total counts of each insulin dose level. We can then compare these theoretical hypoglycemia counts with our observed counts via a chi-squared test. We get a p-value of .8256 and cannot reject the null that hypoglycemia is independent of insulin dosage.