Final Project

Zelalem Abahana and Jessica Ayers

2022-12-11

Introduction

A test of homogeneity is used to compare categorical variables and their distributions. The idea of homogeneity is derived from the null hypothesis, which states that we have equal cell probabilities, and as a result our data would come from the same multinomial distribution. Our question of interest is: do the two (or more) multinomial distributions differ? The test will be derived using the likelihood ratio test and the decision of homogeneity will be based on the resulting comparison of the test statistic and the corresponding Chi-Square critical value.

Derive the homogeneity test

For the homogeneity test, we will derive the corresponding likelihood ratio test. Our hypotheses are as follows

$$H_0 : \pi_{11} = \pi_{12} = ... = \pi_{1J}, \pi_{21} = \pi_{22} = ... = \pi_{2J}, \pi_{I1} = \pi_{I2} = ... = \pi_{IJ}$$
$$H_A : \text{At least some probabilities differ}$$

We will have J independent multinomial distributions each with I cases. The likelihood function of our multinomial functions is as follows:

$$L(\pi_{ij}'s) = \prod_{j=1}^{J}\prod_{i=1}^{I}\pi_{ij}^{n_{ij}}$$

for $i = 1, ..., I$ and $j = 1, ..., J$. The n_{ij} represents the observed counts and the π_{ij} is defined as the proportion of observed counts. Next, we will look at our likelihood function under the null hypothesis. The expected cell count under the null hypothesis is $n_{.j}n_{i.}/n$. It follows that our estimated proportion under the null is

$$\hat{\pi_{ij,null}} = \frac{n_{.j}n_{i.}/n}{n_j}$$

The counts under the alternative are n_{ij} and the estimated proportion follows as

$$\hat{\pi_{ij,alt}} = \frac{n_{ij}}{n_j}$$

We can then begin to construct the likelihood ratio by using these estimates. We have

$$\Lambda = \frac{\prod_{j=1}^{J}\prod_{i=1}^{I}\hat{\pi_{ij,null}}^{n_{ij}}}{\prod_{j=1}^{J}\prod_{i=1}^{I}\hat{\pi_{ij,alt}}^{n_{ij}}}$$

as our ratio. With substitution of the above equations and simplification, it follows

$$\begin{aligned}
\Lambda &= \prod_{j=1}^{J}\prod_{i=1}^{I}\frac{(\frac{n_{.j}n_{i.}/n}{n_j})^{n_{ij}}}{(\frac{n_{ij}}{n_j})^{n_{ij}}} \\
&= \prod_{j=1}^{J}\prod_{i=1}^{I}\frac{(\frac{n_{.j}n_{i.}}{n*n_j})^{n_{ij}}}{(\frac{n_{ij}}{n_j})^{n_{ij}}} \\
&= \prod_{j=1}^{J}\prod_{i=1}^{I}(\frac{n_{.j}n_{i.}}{n*n_j}\frac{n_j}{n_{ij}})^{n_{ij}} \\
&= \prod_{j=1}^{J}\prod_{i=1}^{I}(\frac{n_{.j}n_{i.}}{n*n_{ij}})^{n_{ij}}
\end{aligned}$$

Using this simplified form of our likelihood ratio, we will put this into the form of $-2ln(\Lambda)$ and find the large sample approximate distribution,

$$\begin{aligned}
-2ln(\Lambda) &= -2ln(\prod_{j=1}^{J}\prod_{i=1}^{I}(\frac{n_{.j}n_{i.}}{n*n_{ij}})^{n_{ij}}) \\
&= -2\sum_{j=1}^{J}\sum_{i=1}^{I}n_{ij}ln(\frac{n_{.j}n_{i.}}{n*n_{ij}}) \\
&= 2\sum_{j=1}^{J}\sum_{i=1}^{I}n_{ij}ln(\frac{n*n_{ij}}{n_{.j}n_{i.}}) \\
&= 2\sum_{j=1}^{J}\sum_{i=1}^{I}n_{ij}ln(\frac{n_{ij}}{n_{.j}n_{i.}/n}) \\
&= 2\sum_{j=1}^{J}\sum_{i=1}^{I}\text{obs}_{ij}ln(\frac{\text{obs}_{ij}}{\text{exp}_{ij}})
\end{aligned}$$

This derived test statistic will have an approximate Chi-Square distribution with degrees of freedom = (I-1)(J-1) as shown below,

$$-2ln(\Lambda) \text{ approximately distributed with } X^2_{(I-1)(J-1)}$$

We will see the application of this test with our example data analysis.

Example Data Analysis

We are given data on three different hospitals (A, B, and C), and the corresponding data that end up with infections from Surgical Sites, Pneumonia and Bloodstream. If we assume that for each hospital we have fixed number of trails, we can assume each individual hospital has it's own multinomial distribution. The Chi-Square test for homogeneity will help us analyze our given distributions in order to make a decision regarding if there is evidence that the distributions differ. The hospital data is given in the table below.

```
pdata <- matrix(c(41,27,51,36,3,40,169,106,109), nrow = 3, ncol = 3, byrow = TRUE,
dimnames = list(c("A", "B","C"), c("Surgical", "Pnuemonia", "BloodStream")))
pdata
```

```
##   Surgical Pnuemonia BloodStream
## A       41        27          51
## B       36         3          40
## C      169       106         109
```

We will begin our analysis of the hospital data by creating our expected values. This was done by first finding the row totals, column totals, and overall sample size. Next, the expected values were found by multiplying the corresponding row and column totals and dividing by the overall total.

```
col_sum <- colSums(pdata)
row_sum <- rowSums(pdata)
tot <- sum(row_sum)
#1st row exp
exp_1_1 <- (row_sum[1]*col_sum[1])/tot
exp_1_2 <- (row_sum[1]*col_sum[2])/tot
exp_1_3 <- (row_sum[1]*col_sum[3])/tot
#2nd row exp
exp_2_1 <- (row_sum[2]*col_sum[1])/tot
exp_2_2 <- (row_sum[2]*col_sum[2])/tot
exp_2_3 <- (row_sum[2]*col_sum[3])/tot
#3rd row exp
exp_3_1 <- (row_sum[3]*col_sum[1])/tot
exp_3_2 <- (row_sum[3]*col_sum[2])/tot
exp_3_3 <- (row_sum[3]*col_sum[3])/tot
expectedtable<-matrix(c(exp_1_1, exp_1_2, exp_1_3, exp_2_1, exp_2_2, exp_2_3,
exp_3_1,exp_3_2, exp_3_3), nrow = 3, ncol=3, byrow = TRUE,
dimnames = list(c("A", "B","C"), c("Surgical", "Pnuemonia", "BloodStream")))
expectedtable
```

```
##    Surgical Pnuemonia BloodStream
## A  50.29897  27.80756    40.89347
## B  33.39175  18.46048    27.14777
## C 162.30928  89.73196   131.95876
```

Each expected count in our table is greater than 5, so the Chi-Square large sample approximation will hold for our comparisons. After finding the expected values, the Pearson test statistic and the likelihood ratio test statistic were calculated.

```
Pearson_Chi <- (pdata[1]-exp_1_1)^2/exp_1_1 + (pdata[2]-exp_2_1)^2/exp_2_1 +
  (pdata[3]-exp_3_1)^2/exp_3_1 +
  + (pdata[4]-exp_1_2)^2/exp_1_2 + (pdata[5]-exp_2_2)^2/exp_2_2 +
  (pdata[6]-exp_3_2)^2/exp_3_2 +
  + (pdata[7]-exp_1_3)^2/exp_1_3 + (pdata[8]-exp_2_3)^2/exp_2_3 +
  (pdata[9]-exp_3_3)^2/exp_3_3
Pearson_Chi
```

```
##        A
## 30.69616
```

```
LRT_Chi <- pdata[1]*log(pdata[1]/exp_1_1) + pdata[2]*log(pdata[2]/exp_2_1) +
  pdata[3]*log(pdata[3]/exp_3_1) +pdata[4]*log(pdata[4]/exp_1_2) +
  pdata[5]/log(pdata[5]/exp_2_2) + pdata[6]*log(pdata[6]/exp_3_2) +
  pdata[7]*log(pdata[7]/exp_1_3) + pdata[8]*log(pdata[8]/exp_2_3) +
  pdata[9]*log(pdata[9]/exp_3_3)
LRT_Chi
```

```
##        A
## 22.30009
```

Both of these test statistics were then compared to a critical value from the Chi-Square distribution. To calculate the degrees of freedom, J= 1,. . . ,3 was defined for the number of hospitals and I = 1,. . . ,3 was defined for the different infections. As a result, the degrees of freedom = (I-1)(J-1) = (3-1)*(3-1) = 4. The alpha level was assumed to be 0.05.

```
critical <- qchisq(df=4,0.95)
critical
```

```
## [1] 9.487729
```

The critical value for this test is 9.487729. Finally, the p-values for both the Pearson and likelihood ratio test were found.

```
p_value_Pearson <- pchisq(q=Pearson_Chi, df=4, lower.tail=FALSE)
p_value_Pearson
```

```
##            A
## 3.530853e-06
```

```
p_value_LRT <- pchisq(q=LRT_Chi, df=4, lower.tail=FALSE)
p_value_LRT
```

```
##            A
## 0.0001746522
```

Regarding the Pearson Chi-Square test, our p-value of 3.5308531×10^{-6} is less than an alpha of 0.05 and the test statistic of 30.6961632 is greater than the critical value of 9.487729. Similarly, with the LRT test, our p-value of 1.7465221×10^{-4} is less than an alpha of 0.05 and the test statistic of 22.3000939 is greater than the critical value of 9.487729. Both the Pearson and LRT lead us to reject our null hypothesis and conclude that at least one of the hospitals has a different multinomial distribution.

Simulation Description and Implementation

For the simulation study, we will first analyze the type I error probability associated with different sample sizes with different multinomial probabilities. Secondly, we will explore the power of the each test with 3 different combinations of probabilities. In this Monte Carlo simulation, we will use the Pearson statistic to compute our alpha and power values for the two multinomial case.

Beginning with simulating the alpha values that correspond to every combination of sample sizes with n1 = 20,30,50, 100 and n2 = 20,30,50,100, we will find the test statistic with each probability value for the equal, mixed1, and mixed2 multinomials. These probabilities are:

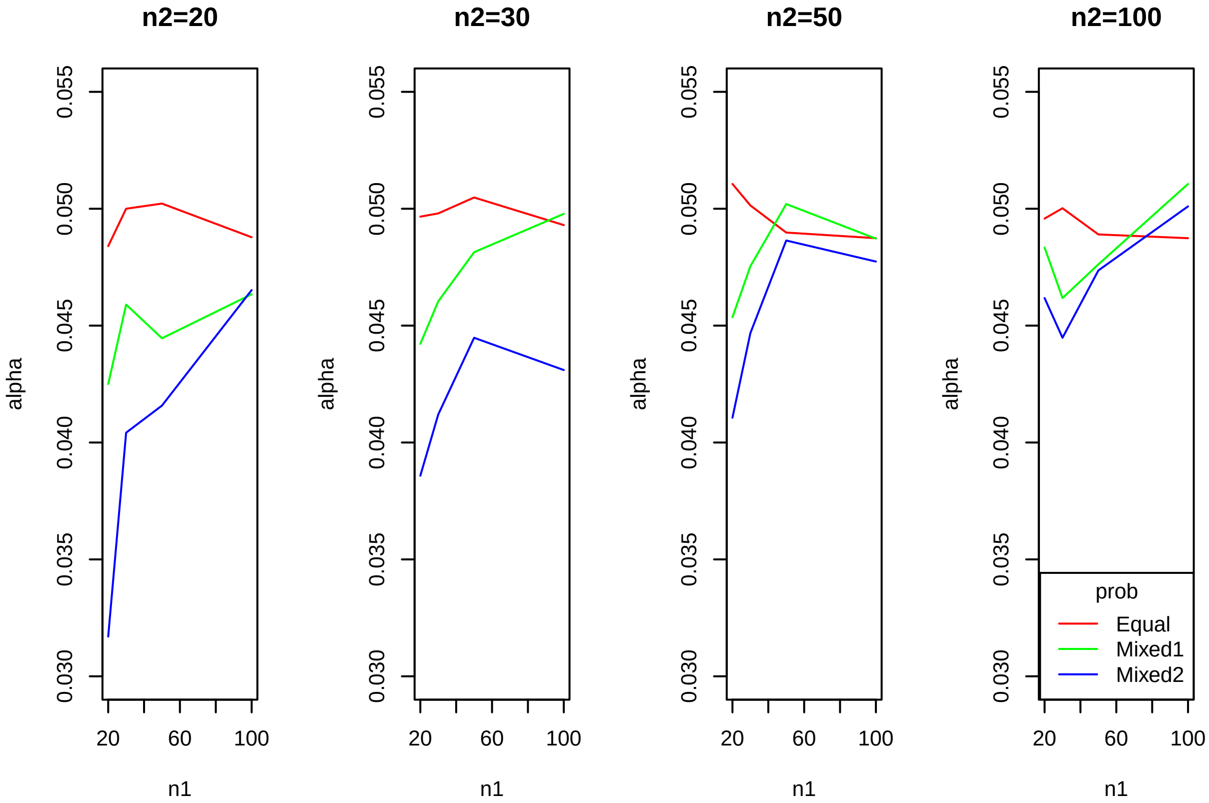
$$
\begin{aligned}
p1 &= 1/3, 1/3, 1/3 \\
p2 &= 1/10, 3/10, 6/10 \\
p3 &= 1/10, 1/10, 8/10
\end{aligned}
$$

A function was created to first create a table from two multinomial distributions with the first probability vector (equal). Similarly to the example data analysis with the hospital data, the Pearson Chi-Square statistic was calculated. By definition, alpha is the type 1 error probability. This is the probability that we reject our null hypothesis, under the conditions that the null hypothesis is true. We will compare the Pearson test statistic to the critical value with degrees of freedom, df = (I-1)(J-1) = (2-1)*(3-1) = 2. Our rejection region is when our test statistic is greater than the corresponding critical value. We will then simulate 50000 true or false values from the comparison, take the mean, and as a result find our alpha value. This was repeated for each probability vector with each combination of sample sizes, In total, there were 48 alpha values created. A plot was created to visualize the differing alpha values.

```r
n1<-c(20,30,50,100)
n2<-c(20,30,50,100)
p1<-c(1/3,1/3,1/3)
p2<-c(1/10,3/10,6/10)
p3<-c(1/10,1/10,8/10)
sim<-function(n1,n2,p1){
  table<-t(cbind(rmultinom(1,n1,p1),rmultinom(1,n2,p1)))
  row_sum<-rowSums(table)
  col_sum<-colSums(table)
  totalsample<-sum(row_sum)
  exp11<-(col_sum[1]*row_sum[1])/totalsample
  exp12<-(col_sum[2]*row_sum[1])/totalsample
  exp13<-(col_sum[3]*row_sum[1])/totalsample
  exp21<-(col_sum[1]*row_sum[2])/totalsample
  exp22<-(col_sum[2]*row_sum[2])/totalsample
  exp23<-(col_sum[3]*row_sum[2])/totalsample
  if(exp11 == 0){
    exp11 = 0.5
  } else{
    exp11 = exp11
  }
  if(exp12 == 0){
   exp12 = 0.5
  } else{
   exp12 = exp12
  }
  if(exp13 == 0){
    exp13 == 0.5
  } else{
   exp13 = exp13
  }
  if(exp21 == 0){
    exp21 = 0.5
  } else{
    exp21 = exp21
  }
  if(exp22 == 0){
    exp22 = 0.5
  } else{
    exp22 = exp22
  }
  if(exp23 == 0){
   exp23 = 0.5
  } else{
    exp23 = exp23
  }
Pearson<-((table[1]-exp11)^2/exp11) + ((table[2]-exp21)^2/exp21) +
  ((table[3]-exp12)^2/exp12) + ((table[4]-exp22)^2/exp22) +
  ((table[5]-exp13)^2/exp13) + ((table[6]-exp23)^2/exp23)
criticalval<-qchisq(df = 2, 0.95)
comp<-Pearson>criticalval
pval<-pchisq(Pearson, df = 2, lower.tail = FALSE)
return(comp)
}
rep1<-replicate(50000,sim(n1[1],n2[1],p1))
alpha1_20<-mean(rep1)
rep2<-replicate(50000,sim(n1[1],n2[2],p1))
alpha2_30<-mean(rep2)
rep3<-replicate(50000,sim(n1[1],n2[3],p1))
alpha3_50<-mean(rep3)
rep4<-replicate(50000,sim(n1[1], n2[4],p1))
alpha4_100<-mean(rep4)
rep5<-replicate(50000, sim(n1[2], n2[1], p1))
alpha5_20<-mean(rep5)
rep6<-replicate(50000, sim(n1[2], n2[2], p1))
alpha6_30<-mean(rep6)
rep7<-replicate(50000, sim(n1[2], n2[3], p1))
alpha7_50<-mean(rep7)
rep8<-replicate(50000, sim(n1[2], n2[4], p1))
alpha8_100<-mean(rep8)
rep9<-replicate(50000, sim(n1[3], n2[1], p1))
alpha9_20<-mean(rep9)
rep10<-replicate(50000, sim(n1[3], n2[2], p1))
alpha10_30<-mean(rep10)
rep11<-replicate(50000, sim(n1[3], n2[3], p1))
alpha11_50<-mean(rep11)
rep12<-replicate(50000, sim(n1[3], n2[4], p1))
alpha12_100<-mean(rep12)
rep13<-replicate(50000, sim(n1[4], n2[1], p1))
alpha13_20<-mean(rep13)
rep14<-replicate(50000, sim(n1[4], n2[2], p1))
alpha14_30<-mean(rep14)
rep15<-replicate(50000, sim(n1[4], n2[3], p1))
alpha15_50<-mean(rep15)
rep16<-replicate(50000, sim(n1[4], n2[4], p1))
alpha16_100<-mean(rep16)
alpha_p1_20<-cbind(alpha1_20, alpha5_20,alpha9_20,alpha13_20)
alpha_p1_30<-cbind(alpha2_30,alpha6_30,alpha10_30,alpha14_30)
alpha_p1_50<-cbind(alpha3_50,alpha7_50,alpha11_50,alpha15_50)
alpha_p1_100<-cbind(alpha4_100,alpha8_100,alpha12_100,alpha16_100)
rep17<-replicate(50000,sim(n1[1],n2[1],p2))
alpha17_20<-mean(rep17)
rep18<-replicate(50000,sim(n1[1],n2[2],p2))
alpha18_30<-mean(rep18)
rep19<-replicate(50000,sim(n1[1],n2[3],p2))
alpha19_50<-mean(rep19)
rep20<-replicate(50000,sim(n1[1], n2[4],p2))
```

```
alpha20_100<-mean(rep20)
rep21<-replicate(50000, sim(n1[2], n2[1], p2))
alpha21_20<-mean(rep21)
rep22<-replicate(50000, sim(n1[2], n2[2], p2))
alpha22_30<-mean(rep22)
rep23<-replicate(50000, sim(n1[2], n2[3], p2))
alpha23_50<-mean(rep23)
rep24<-replicate(50000, sim(n1[2], n2[4], p2))
alpha24_100<-mean(rep24)
rep25<-replicate(50000, sim(n1[3], n2[1], p2))
alpha25_20<-mean(rep25)
rep26<-replicate(50000, sim(n1[3], n2[2], p2))
alpha26_30<-mean(rep26)
rep27<-replicate(50000, sim(n1[3], n2[3], p2))
alpha27_50<-mean(rep27)
rep28<-replicate(50000, sim(n1[3], n2[4], p2))
alpha28_100<-mean(rep28)
rep29<-replicate(50000, sim(n1[4], n2[1], p2))
alpha29_20<-mean(rep29)
rep30<-replicate(50000, sim(n1[4], n2[2], p2))
alpha30_30<-mean(rep30)
rep31<-replicate(50000, sim(n1[4], n2[3], p2))
alpha31_50<-mean(rep31)
rep32<-replicate(50000, sim(n1[4], n2[4], p2))
alpha32_100<-mean(rep32)
alpha_p2_20<-cbind(alpha17_20,alpha21_20,alpha25_20,alpha29_20)
alpha_p2_30<-cbind(alpha18_30,alpha22_30,alpha26_30,alpha30_30)
alpha_p2_50<-cbind(alpha19_50,alpha23_50,alpha27_50,alpha31_50)
alpha_p2_100<-cbind(alpha20_100,alpha24_100,alpha28_100,alpha32_100)
rep33<-replicate(50000,sim(n1[1],n2[1],p3))
alpha33_20<-mean(rep33)
rep34<-replicate(50000,sim(n1[1],n2[2],p3))
alpha34_30<-mean(rep34)
rep35<-replicate(50000,sim(n1[1],n2[3],p3))
alpha35_50<-mean(rep35)
rep36<-replicate(50000,sim(n1[1], n2[4],p3))
alpha36_100<-mean(rep36)
rep37<-replicate(50000, sim(n1[2], n2[1], p3))
alpha37_20<-mean(rep37)
rep38<-replicate(50000, sim(n1[2], n2[2], p3))
alpha38_30<-mean(rep38)
rep39<-replicate(50000, sim(n1[2], n2[3], p3))
alpha39_50<-mean(rep39)
rep40<-replicate(50000, sim(n1[2], n2[4], p3))
alpha40_100<-mean(rep40)
rep41<-replicate(50000, sim(n1[3], n2[1], p3))
alpha41_20<-mean(rep41)
rep42<-replicate(50000, sim(n1[3], n2[2], p3))
alpha42_30<-mean(rep42)
rep43<-replicate(50000, sim(n1[3], n2[3], p3))
alpha43_50<-mean(rep43)
rep44<-replicate(50000, sim(n1[3], n2[4], p3))
alpha44_100<-mean(rep44)
```

```
rep45<-replicate(50000, sim(n1[4], n2[1], p3))
alpha45_20<-mean(rep45)
rep46<-replicate(50000, sim(n1[4], n2[2], p3))
alpha46_30<-mean(rep46)
rep47<-replicate(50000, sim(n1[4], n2[3], p3))
alpha47_50<-mean(rep47)
rep48<-replicate(50000, sim(n1[4], n2[4], p3))
alpha48_100<-mean(rep48)
alpha_p3_20<-cbind(alpha33_20,alpha37_20,alpha41_20,alpha45_20)
alpha_p3_30<-cbind(alpha34_30,alpha38_30,alpha42_30,alpha46_30)
alpha_p3_50<-cbind(alpha35_50,alpha39_50,alpha43_50,alpha47_50)
alpha_p3_100<-cbind(alpha36_100,alpha40_100,alpha44_100,alpha48_100)
par(mfrow=c(1,4))
plot(n1, alpha_p1_20, ylim = c(0.030, 0.055), type = "l", col = "red",
     main="n2=20", ylab= "alpha")
lines(n1,alpha_p2_20, col = "green")
lines(n1,alpha_p3_20, col = "blue")
plot(n1, alpha_p1_30, ylim=c(0.030,0.055), type = "l", col = "red",
     main= "n2=30", ylab = "alpha")
lines(n1,alpha_p2_30, col="green")
lines(n1,alpha_p3_30, col="blue")
plot(n1,alpha_p1_50, ylim=c(0.030,0.055),type="l",col="red",
     main = "n2=50", ylab = "alpha")
lines(n1,alpha_p2_50, col = "green")
lines(n1,alpha_p3_50, col= "blue")
plot(n1,alpha_p1_100, ylim=c(0.030,0.055),type="l",col="red",
     main = "n2=100", ylab = "alpha")
lines(n1,alpha_p2_100, col = "green")
lines(n1,alpha_p3_100, col= "blue")
legend("bottomright", c("Equal", "Mixed1","Mixed2"), lty = 1,
       col = c("red", "green", "blue"), title= "prob")
```



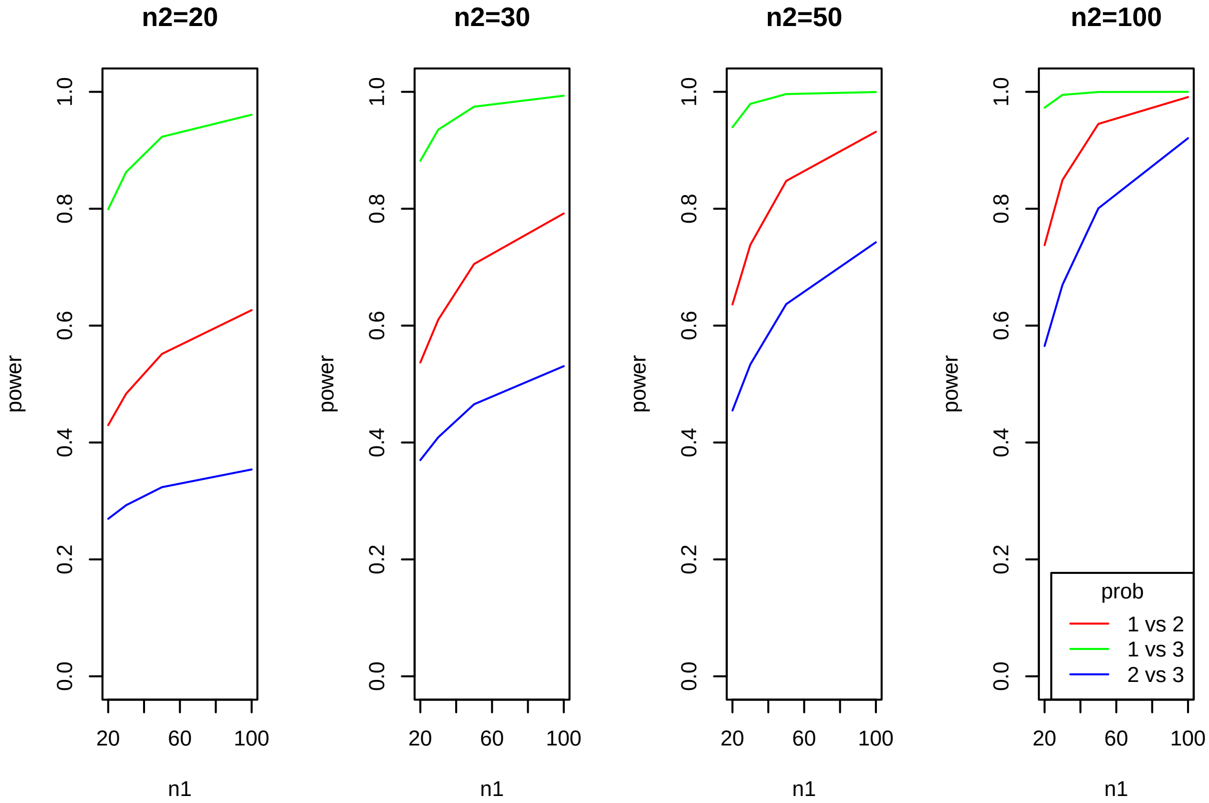
We can see in the above plots that mixed 2 probabilities consistently generated the lowest type I error values. The equal probabilities gave the highest alpha values overall. This supports the idea that equal probabilities would provide evidence against the alternative and thus have a higher type I error rate. It also seems that as our sample sizes increase, the alpha values also increase. We would suggest the mixed 2 probabilities performs the best out of the probability combinations, especially at lower sample sizes.

Next, the power values were simulated in a similar way for three comparisons: equal vs mixed 1 (p1 to p2), equal vs mixed 2 (p1 to p3), and mixed 1 to mixed 2 (p2 to p3). The difference in simulation was the change in the probability values when creating our random sample. In addition, power is defined as the probability we reject the null, given the alternative is true. This can be translated into creating our test statistic under the alternative hypothesis and comparing it to our critical value. 48 power values were simulated and a plot was created to visualize our findings. Sim2 was a function created to compare equal to mixed 1, sim3 for equal to mixed 2, and sim4 for mixed1 to mixed2.

```r
sim2<-function(n1,n2,p1,p2){
  table<-t(cbind(rmultinom(1,n1,p1),rmultinom(1,n2,p2)))
  row_sum<-rowSums(table)
  col_sum<-colSums(table)
  totalsample<-sum(row_sum)
  exp11<-(col_sum[1]*row_sum[1])/totalsample
  exp12<-(col_sum[2]*row_sum[1])/totalsample
  exp13<-(col_sum[3]*row_sum[1])/totalsample
  exp21<-(col_sum[1]*row_sum[2])/totalsample
  exp22<-(col_sum[2]*row_sum[2])/totalsample
  exp23<-(col_sum[3]*row_sum[2])/totalsample
  if(exp11 == 0){
    exp11 = 0.5
  } else{
    exp11 = exp11
  }
  if(exp12 == 0){
    exp12 = 0.5
  } else{
    exp12 = exp12
  }
  if(exp13 == 0){
    exp13 == 0.5
  } else{
    exp13 = exp13
  }
  if(exp21 == 0){
    exp21 = 0.5
  } else{
    exp21 = exp21
  }
  if(exp22 == 0){
    exp22 = 0.5
  } else{
    exp22 = exp22
  }
  if(exp23 == 0){
    exp23 = 0.5
  } else{
    exp23 = exp23
  }
  Pearson_HA<-((table[1]-exp11)^2/exp11) + ((table[2]-exp21)^2/exp21) +
    ((table[3]-exp12)^2/exp12) + ((table[4]-exp22)^2/exp22) +
    ((table[5]-exp13)^2/exp13) + ((table[6]-exp23)^2/exp23)
  critical<-qchisq(0.95,df=2)
  power<-Pearson_HA>critical
  return(power)
}
newrep1<-replicate(50000,sim2(n1[1],n2[1],p1,p2))
power1_20<-mean(newrep1)
newrep2<-replicate(50000,sim2(n1[1],n2[2],p1,p2))
power2_30<-mean(newrep2)
newrep3<-replicate(50000,sim2(n1[1],n2[3],p1,p2))
power3_50<-mean(newrep3)
newrep4<-replicate(50000,sim2(n1[1],n2[4],p1,p2))
power4_100<-mean(newrep4)
newrep5<-replicate(50000,sim2(n1[2],n2[1],p1,p2))
power5_20<-mean(newrep5)
newrep6<-replicate(50000,sim2(n1[2],n2[2],p1,p2))
power6_30<-mean(newrep6)
newrep7<-replicate(50000,sim2(n1[2],n2[3],p1,p2))
power7_50<-mean(newrep7)
newrep8<-replicate(50000,sim2(n1[2],n2[4],p1,p2))
power8_100<-mean(newrep8)
newrep9<-replicate(50000,sim2(n1[3],n2[1],p1,p2))
power9_20<-mean(newrep9)
```

```
newrep10<-replicate(50000,sim2(n1[3],n2[2],p1,p2))
power10_30<-mean(newrep10)
newrep11<-replicate(50000,sim2(n1[3],n2[3],p1,p2))
power11_50<-mean(newrep11)
newrep12<-replicate(50000,sim2(n1[3],n2[4],p1,p2))
power12_100<-mean(newrep12)
newrep13<-replicate(50000,sim2(n1[4],n2[1],p1,p2))
power13_20<-mean(newrep13)
newrep14<-replicate(50000,sim2(n1[4],n2[2],p1,p2))
power14_30<-mean(newrep14)
newrep15<-replicate(50000,sim2(n1[4],n2[3],p1,p2))
power15_50<-mean(newrep15)
newrep16<-replicate(50000,sim2(n1[4],n2[4],p1,p2))
power16_100<-mean(newrep16)
power_p1_p2_20<-cbind(power1_20,power5_20,power9_20,power13_20)
power_p1_p2_30<-cbind(power2_30,power6_30,power10_30,power14_30)
power_p1_p2_50<-cbind(power3_50,power7_50,power11_50,power15_50)
power_p1_p2_100<-cbind(power4_100,power8_100,power12_100,power16_100)
sim3<-function(n1,n2,p1,p3){
  table<-t(cbind(rmultinom(1,n1,p1),rmultinom(1,n2,p3)))
  row_sum<-rowSums(table)
  col_sum<-colSums(table)
  totalsample<-sum(row_sum)
  exp11<-(col_sum[1]*row_sum[1])/totalsample
  exp12<-(col_sum[2]*row_sum[1])/totalsample
  exp13<-(col_sum[3]*row_sum[1])/totalsample
  exp21<-(col_sum[1]*row_sum[2])/totalsample
  exp22<-(col_sum[2]*row_sum[2])/totalsample
  exp23<-(col_sum[3]*row_sum[2])/totalsample
  if(exp11 == 0){
    exp11 = 0.5
  } else{
    exp11 = exp11
  }
  if(exp12 == 0){
    exp12 = 0.5
  } else{
    exp12 = exp12
  }
  if(exp13 == 0){
    exp13 == 0.5
  } else{
    exp13 = exp13
  }
  if(exp21 == 0){
    exp21 = 0.5
  } else{
    exp21 = exp21
  }
  if(exp22 == 0){
    exp22 = 0.5
  } else{
    exp22 = exp22
  }
  if(exp23 == 0){
    exp23 = 0.5
  } else{
    exp23 = exp23
  }
  Pearson_HA<-((table[1]-exp11)^2/exp11) + ((table[2]-exp21)^2/exp21) +
    ((table[3]-exp12)^2/exp12) + ((table[4]-exp22)^2/exp22) +
    ((table[5]-exp13)^2/exp13) + ((table[6]-exp23)^2/exp23)
  critical<-qchisq(0.95,df=2)
  power<-Pearson_HA>critical
  return(power)
}
newrep17<-replicate(50000,sim3(n1[1],n2[1],p1,p3))
power17_20<-mean(newrep17)
newrep18<-replicate(50000,sim3(n1[1],n2[2],p1,p3))
power18_30<-mean(newrep18)
newrep19<-replicate(50000,sim3(n1[1],n2[3],p1,p3))
power19_50<-mean(newrep19)
newrep20<-replicate(50000,sim3(n1[1],n2[4],p1,p3))
power20_100<-mean(newrep20)
newrep21<-replicate(50000,sim3(n1[2],n2[1],p1,p3))
power21_20<-mean(newrep21)
newrep22<-replicate(50000,sim3(n1[2],n2[2],p1,p3))
power22_30<-mean(newrep22)
newrep23<-replicate(50000,sim3(n1[2],n2[3],p1,p3))
power23_50<-mean(newrep23)
newrep24<-replicate(50000,sim3(n1[2],n2[4],p1,p3))
power24_100<-mean(newrep24)
newrep25<-replicate(50000,sim3(n1[3],n2[1],p1,p3))
power25_20<-mean(newrep25)
newrep26<-replicate(50000,sim3(n1[3],n2[2],p1,p3))
power26_30<-mean(newrep26)
newrep27<-replicate(50000,sim3(n1[3],n2[3],p1,p3))
power27_50<-mean(newrep27)
newrep28<-replicate(50000,sim3(n1[3],n2[4],p1,p3))
power28_100<-mean(newrep28)
newrep29<-replicate(50000,sim3(n1[4],n2[1],p1,p3))
power29_20<-mean(newrep29)
newrep30<-replicate(50000,sim3(n1[4],n2[2],p1,p3))
power30_30<-mean(newrep30)
newrep31<-replicate(50000,sim3(n1[4],n2[3],p1,p3))
power31_50<-mean(newrep31)
newrep32<-replicate(50000,sim3(n1[4],n2[4],p1,p3))
power32_100<-mean(newrep32)
power_p1_p3_20<-cbind(power17_20,power21_20,power25_20,power29_20)
power_p1_p3_30<-cbind(power18_30,power22_30,power26_30,power30_30)
power_p1_p3_50<-cbind(power19_50,power23_50,power27_50,power31_50)
power_p1_p3_100<-cbind(power20_100,power24_100,power28_100,power32_100)
sim4<-function(n1,n2,p2,p3){
  table<-t(cbind(rmultinom(1,n1,p2),rmultinom(1,n2,p3)))
  row_sum<-rowSums(table)
  col_sum<-colSums(table)
  totalsample<-sum(row_sum)
  exp11<-(col_sum[1]*row_sum[1])/totalsample
  exp12<-(col_sum[2]*row_sum[1])/totalsample
  exp13<-(col_sum[3]*row_sum[1])/totalsample
  exp21<-(col_sum[1]*row_sum[2])/totalsample
  exp22<-(col_sum[2]*row_sum[2])/totalsample
  exp23<-(col_sum[3]*row_sum[2])/totalsample
  if(exp11 == 0){
    exp11 = 0.5
  } else{
    exp11 = exp11
  }
  if(exp12 == 0){
    exp12 = 0.5
  } else{
    exp12 = exp12
  }
  if(exp13 == 0){
    exp13 == 0.5
  } else{
    exp13 = exp13
  }
  if(exp21 == 0){
    exp21 = 0.5
  } else{
    exp21 = exp21
  }
  if(exp22 == 0){
    exp22 = 0.5
  } else{
    exp22 = exp22
  }
  if(exp23 == 0){
    exp23 = 0.5
  } else{
    exp23 = exp23
  }
  Pearson_HA<-((table[1]-exp11)^2/exp11) + ((table[2]-exp21)^2/exp21) +
    ((table[3]-exp12)^2/exp12) + ((table[4]-exp22)^2/exp22) +
    ((table[5]-exp13)^2/exp13) + ((table[6]-exp23)^2/exp23)
  critical<-qchisq(0.95,df=2)
  power<-Pearson_HA>critical
  return(power)
}
newrep33<-replicate(50000,sim4(n1[1],n2[1],p2,p3))
power33_20<-mean(newrep33)
newrep34<-replicate(50000,sim4(n1[1],n2[2],p2,p3))
power34_30<-mean(newrep34)
newrep35<-replicate(50000,sim4(n1[1],n2[3],p2,p3))
power35_50<-mean(newrep35)
newrep36<-replicate(50000,sim4(n1[1],n2[4],p2,p3))
power36_100<-mean(newrep36)
newrep37<-replicate(50000,sim4(n1[2],n2[1],p2,p3))
```

```
power37_20<-mean(newrep37)
newrep38<-replicate(50000,sim4(n1[2],n2[2],p2,p3))
power38_30<-mean(newrep38)
newrep39<-replicate(50000,sim4(n1[2],n2[3],p2,p3))
power39_50<-mean(newrep39)
newrep40<-replicate(50000,sim4(n1[2],n2[4],p2,p3))
power40_100<-mean(newrep40)
newrep41<-replicate(50000,sim4(n1[3],n2[1],p2,p3))
power41_20<-mean(newrep41)
newrep42<-replicate(50000,sim4(n1[3],n2[2],p2,p3))
power42_30<-mean(newrep42)
newrep43<-replicate(50000,sim4(n1[3],n2[3],p2,p3))
power43_50<-mean(newrep43)
newrep44<-replicate(50000,sim4(n1[3],n2[4],p2,p3))
power44_100<-mean(newrep44)
newrep45<-replicate(50000,sim4(n1[4],n2[1],p2,p3))
power45_20<-mean(newrep45)
newrep46<-replicate(50000,sim4(n1[4],n2[2],p2,p3))
power46_30<-mean(newrep46)
newrep47<-replicate(50000,sim4(n1[4],n2[3],p2,p3))
power47_50<-mean(newrep47)
newrep48<-replicate(50000,sim4(n1[4],n2[4],p2,p3))
power48_100<-mean(newrep48)
power_p2_p3_20<-cbind(power33_20,power37_20,power41_20,power45_20)
power_p2_p3_30<-cbind(power34_30,power38_30,power42_30,power46_30)
power_p2_p3_50<-cbind(power35_50,power39_50,power43_50,power47_50)
power_p2_p3_100<-cbind(power36_100,power40_100, power44_100, power48_100)
par(mfrow = c(1,4))
plot(n1, power_p1_p2_20, type = "l", main = "n2=20", col= "red",
     ylim=c(0,1), ylab="power")
lines(n1, power_p1_p3_20, col="green")
lines(n1,power_p2_p3_20, col="blue")
plot(n1, power_p1_p2_30, type = "l", main= "n2=30", col="red",
     ylim=c(0,1), ylab= "power")
lines(n1,power_p1_p3_30, col="green")
lines(n1,power_p2_p3_30, col= "blue")
plot(n1,power_p1_p2_50, type = "l", main = "n2=50", col= "red",
     ylim=c(0,1), ylab= "power")
lines(n1,power_p1_p3_50, col= "green")
lines(n1,power_p2_p3_50, col = "blue")
plot(n1, power_p1_p2_100, col = "red", type = "l", ylim = c(0,1),
     ylab= "power", main = "n2=100")
lines(n1,power_p1_p3_100,col = "green")
lines(n1, power_p2_p3_100, col = "blue")
legend("bottomright", title = "prob", c("1 vs 2", "1 vs 3", "2 vs 3"),
       lty = 1, col = c("red", "green", "blue"))
```



Based off the plot above, the test consisting of comparing our equal probabilities to our mixed 2 probabilities consistently had the highest power. For each test, the power also increased as our sample sizes increased The test comparing mixed 1 to mixed 2 provided us with the lowest power. This means we would have a large type II error probability for this test.