Bacheloroppgave

NTNU
Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for matematiske fag

Eivind Hagemann Brataas

# Barnard's CSM Test

Bacheloroppgave i matematiske fag
Veileder: Øyvind Bakke
Mai 2020

NTNU
Kunnskap for en bedre verden

Eivind Hagemann Brataas

# Barnard's CSM Test

Bacheloroppgave i matematiske fag
Veileder: Øyvind Bakke
Mai 2020

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for matematiske fag

# Barnard's CSM test for two by two contingency tables

Eivind Hagemann Brataas

15 May 2020

### Abstract

In 1945 George Barnard published an article that described a new exact test for two by two contingency tables. He claimed that it was more powerful than Fisher's exact test, which resulted in a dispute over a series of articles. Because of its complexity and his disputes with Fisher himself, the CSM test has since been forgotten and is rarely used in science today. The aim of this thesis is to test whether or not the test has any merit by comparing it to some other, more popular methods. The conclusion is that, for low values of $n_1$ and $n_2$, Barnard's CSM test performs better than any of its competitors and should, therefore, see more usage when conducting research.

# Contents

# 1 Introduction

Assume that $X_1$ is binomially distributed with parameters $n_1$ and $p_1$ and that $X_2$ is binomially distributed with parameters $n_2$ and $p_2$. In addition, assume that $X_1$ and $X_2$ are independent. We want to test the null hypothesis $H_0\colon p_1 = p_2$ against the alternative $H_1\colon p_1 \neq p_2$ using the data $X_1 = x_1$ and $X_2 = x_2$.

In this thesis, I consider 4 methods for testing this type (2 by 2 contingency tables) of hypothesis.

- The asymptotic method
- Fisher's exact test
- The supremum method (often erroneously called Barnard's test)
- Barnard's CSM test

The first part of the thesis is about introducing each of these methods.

Both the asymptotic method and Fisher's exact test are commonly used in science today, whereas the supremum method and Barnard's test are not too popular. The latter two are quite computationally intensive, which may explain their low usage. However, with today's computing power we have no problem calculating the p-values for lower values of $n_1$ and $n_2$. Barnard claims that his test is more powerful than Fisher's. The question is whether or not this is true, and whether the more computationally heavy tests have merit than the more popular ones. This is explored in practice in the last part of the thesis.

# 2 Valid P-values

The p-value is often defined as the probability of obtaining test results at least as extreme as the observed result, given the null hypothesis being true. We reject the hypothesis if the p-value is less than the significance level $\alpha \in [0, 1]$.

I will use a more general definition: A p-value $p(X)$ is in its own a test statistic satisfying $0 \leq p(x) \leq 1$ for all sample points $x$. If $p(X)$ is small, there is a low probability of obtaining the observed results (or even more extreme) given that $H_0$ is true. So if $p(X)$ is small, $H_1$ is probably true.

Let $\Theta$ be the space of all possible values of $p_1$ and $p_2$ and let $\Theta_0 \subset \Theta$ be the subspace where $H_0$ is true. In general, we say that $p(X)$ is *valid* if, for every $\theta \in \Theta_0$ and every $0 \leq \alpha \leq 1$,

$$P_\theta(p(X) \leq \alpha) \leq \alpha$$

(Casella and Berger, 2008, p. 397). In other words, if the probability of rejecting $H_0$ on the basis of $p(X)$ is less than $\alpha$ given that $H_0$ is true, our p-value is valid.

If a test produces a valid p-value, we will only conclude that $H_0$ is true if the data warrant it.

# 3 Introduction of the Methods

## 3.1 The Asymptotic Method

To determine how extreme an observation is, we can use a test statistic. A test statistic often used by the asymptotic method is the $z$-pooled statistic,

$$T(x_1, x_2) = \frac{\frac{x_1}{n_1} - \frac{x_2}{n_2}}{\sqrt{(\frac{1}{n_1} + \frac{1}{n_2})(\frac{x_1+x_2}{n_1+n_2})(1 - \frac{x_1+x_2}{n_1+n_2})}}, \tag{1}$$

which by the central limit theorem and other convergence theorems tends towards the standard normal distribution when $n_1$ and $n_2$ increase. (Rejection of $H_0$ based on large values of $|T|$ is equivalent to rejecting based on large values of the chi-squared statistic commonly used for contingency tables.) However, this method does not produce a valid p-value. We will see that for lower $n_1$ and $n_2$, and when we go further out in the tails of the distribution (lower significance levels), the approximation is quite bad. In these cases, the most common method to use is Fisher's exact test.

## 3.2 Fisher's Exact Test

The most common method that produces a valid p-value is Fisher's exact test. The test is based on

$$\begin{aligned} P(X_1 = x_1 \mid X_1 + X_2 = c) &= \frac{P(X_1 = x_1 \cap X_1 + X_2 = c)}{P(X_1 + X_2 = c)} \\ &= \frac{P(X_1 = x_1 \cap X_2 = c - X_1)}{P(X_1 + X_2 = c)} \\ &= \frac{P(X_1 = x_1)P(X_2 = c - X_1)}{P(X_1 + X_2 = c)} \\ &= \frac{\binom{n_1}{x_1} p^{x_1}(1-p)^{n_1-x_1}\binom{n_2}{c-x_1} p^{c-x_1}(1-p)^{n_2-c+x_1}}{\binom{n_1+n_2}{c} p^c (1-p)^{n_1+n_2-c}} \\ &= \frac{\binom{n_1}{x_1}\binom{n_2}{c-x_1}}{\binom{n_1+n_2}{c}}, \end{aligned}$$

which is the pmf of a hypergeometric distribution with parameters $n_1+n_2$, $n_1$ and $c$. Note that the nuisance parameter $p$ disappeared when we conditioned on $c$, the total number of successes, showing that $X_1 + X_2$ is a sufficient statistic for $p = p_1 = p_2$ under $H_0$.

Let $T$ be the above test statistic. We define the p-value

$$\begin{aligned} p(x_1, x_2) &= P(T(X_1, X_2) \geq T(x_1, x_2) \mid X_1 + X_2 = x_1 + x_2) \\ &= \sum_{T(z, x_1+x_2-z) \geq T(x_1, x_2)} P(z \mid X_1 + X_2 = x_1 + x_2) \\ &= \sum_{T(z, x_1+x_2-z) \geq T(x_1, x_2)} f(z), \end{aligned}$$

where $f$ is the pmf of a hypergeometric distribution with parameters $n_1+n_2$, $n_1$ and $x_1 + x_2$. That is, we calculate $T(x_1, x_2) = t$ and sum over all the probabilities for the combinations that sum to $x_1 + x_2$ and that are at least as extreme with respect to our test statistic (Casella and Berger, 2008, p. 399).

The use of an external test statistic is rather unusual for Fisher's exact test. Normally, one simply uses $X_1$, where a more extreme observation is an observation where $X_1$ gets a higher value than $x_1$ (or lower, depending on the "sidedness" of the test). It is then somewhat unclear how to deal with two-sided tests, and several solutions exist. This is no problem in our case.

### Proof that Fisher's Method Produces a Valid P-value

Let $X = (X_1, X_2)$ be the outcome of the experiment and let $C = X_1 + X_2$. For each sample point $x$, define

$$p(x) = P(T(X) \geq T(x) \mid C = c),$$

where $c$ denotes $x_1 + x_2$. As was shown in the last section, $p(x)$ is a valid p-value for the test when we condition on $C = c$ (no supremum needed since the nuisance parameter has disappeared).

Now we need to show that $p(x)$ is valid also when the statistical experiment is performed unconditionally, as in our situation:

$$\begin{aligned} P(p(X) \leq \alpha) &= \sum_c P(p(X) \leq \alpha \mid C = c) P(C = c) \\ &\leq \sum_c \alpha P(C = c) \\ &= \alpha \sum_c P(C = c) \\ &= \alpha. \end{aligned}$$

Thus, $p(X)$ is a valid p-value, which is what we wanted to show. However, as we will see, this method is very conservative, which means that it produces high p-values compared to other methods. In practice, this means that it will reject a lot fewer null hypotheses. This is good if $H_0$ is true. However, if $H_0$ is false, there will still be fewer null hypothesis rejected than the other methods. In a sense we want our rejections to occur right below $100\alpha\%$ of the time if $H_0$ is true, so that even when $p_1$ and $p_2$ are just slightly different, we get a power higher than the significance level.

## 3.3 The Supremum Method

For an outcome $(x_1, x_2)$, its p-value is given by

$$p(x_1, x_2) = \sup_p P_p(T(X_1, X_2) \geq T(x_1, x_2)),$$

where $p = p_1 = p_2$ is the common parameter under $H_0$, and we will take $T$ to be the same test statistic as in (1) (Casella and Berger, 2008, p. 397). Let $T(X_1, X_2) = T$ and $T(x_1, x_2) = t$. For a given $p$,

$$P_p(|T| \geq |t|) = \sum_{\substack{x_1, x_2 \\ \text{where} |T| \geq |t|}} P_p(X_1 = x_1 \cap X_2 = x_2),$$

where

$$P_p(X_1 = x_1 \cap X_2 = x_2) = \binom{n_1}{x_1} p^{x_1} (1-p)^{n_1 - x_1} \binom{n_2}{x_2} p^{x_2} (1-p)^{n_2 - x_2}.$$

So we maximize with respect to $p$ the sum of all the binomial probabilities for all the more "extreme" combinations of $x_1$ and $x_2$.

### Why the Supremum Method Produces a Valid P-value

$H_0\colon \theta \in \Theta_0$, $H_1\colon \theta \notin \Theta_0$. Let $X$ be the outcome of the experiment and let $T(X)$ be a test statistic such that large values of $T$ give evidence that the null hypothesis is false. Assume $\theta \in \Theta_0$. For a given realization $x$, define

$$p_\theta(x) = P_\theta(T(X) \geq T(x)).$$

From this definition, it follows that

$$p_\theta(x') \leq p_\theta(x) \iff T(x') \geq T(x).$$

This implies that

$$p_\theta(X) \leq p_\theta(x) \iff T(X) \geq T(x)$$

also when $X$ is a random variable. Hence

$$P_\theta(p_\theta(X) \leq p_\theta(x)) = P_\theta(T(X) \geq T(x)) = p_\theta(x).$$

Let $\alpha \in [0,1]$, and let $\alpha' = \sup\{p_\theta(x) \mid p_\theta(x) \leq \alpha\}$. (So, loosely, $\alpha'$ is the greatest $p_\theta(x)$ possible no larger than $\alpha$.) Then

$$P_\theta(p_\theta(X) \leq \alpha) = P_\theta(p_\theta(X) \leq \alpha') = \alpha' \leq \alpha$$

This holds for all $\theta \in \Theta_0$, so

$$\sup_{\theta \in \Theta_0} P_\theta(p_\theta(X) \leq \alpha) \leq \alpha$$

for all $\alpha$, which is what we wanted to show.

## 3.4 Barnard's CSM Test

The last test we will take a look at is Barnard's exact CSM test. Barnard calls his test progressive conservative, because of all the conservative tests, it is the least conservative (Barnard, 1947). Because of its complexity, it has seen a lot less usage than any of the previous tests we have looked at. However, for low $n_1$ and $n_2$, today's computers can definitely handle the computations.

Another reason for its low usage may come from Barnard's disputes with Fisher himself, the father of modern statistics. Fisher disagreed with Barnard's claim that his test was more powerful than his own (Barnard, 1945a). This resulted in a series of articles exchanged between the two, and a couple of years later, Barnard himself published an article where he explained how Fisher was right (Barnard, 1949). I will go through some of the arguments of this debate at the end of this section.

Until now, we have used the same test statistic to determine which observations are extreme and which are not. Barnard uses a more complex statistic. It involves sequentially adding up the joint probabilities for the most extreme observations when maximized with respect to $p$. The end result is that the p-values themselves serve as the primary test statistic (together with rank). Except for this bizarre test statistic, the test works in the same way as the supremum method.

Like the name of the test suggests, there are three conditions that have to be satisfied when making this test statistic. They all are meant to help determine the extremeness of an observation.

Like Barnard (1947) did, I will use a lattice diagram to help visualize these concepts (Figure 1).

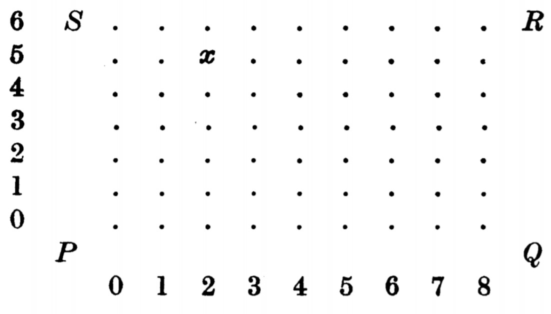

**Figure 1:** Lattice diagram copied from Barnard, 1947

*Convexity* (C): When considering the extremeness of an observation, the two points with respectively the same abscissa or the same ordinate as $(a, b)$, and which also lies closer to the diagonal PR, will both be less extreme observations than $(a, b)$ itself. For example, the points $(2, 4)$ and $(3, 5)$ are less extreme than $x = (2, 5)$ (Figure 1).

*Symmetry* (S): The points $(n_1 - x_1, n_2 - x_2)$ and $(x_1, x_2)$ have the same rank of extremeness. This means that we only have to consider the development of the statistic on one side of PR, since the other side will develop in the same way. This will simplify formulations later. For example, $(2, 5)$ and $(6, 1)$ have the same rank.

The convexity and symmetry requirements will always determine $(n_1, 0)$ and $(0, n_2)$ to be the most extreme points, and they are given rank 1.

*Maximum* (M): For an observation $(a, b)$, let

$$P_p(a, b) = \binom{n_1}{a} p^a (1-p)^{n_1-a} \binom{n_2}{b} p^b (1-p)^{n_2-b}.$$

Then define recursively

$$P_{n,p}(a, b) = P_p(a, b) + P_p(n_1 - a, n_2 - b) + P_{n-1,p}(c, d), \tag{2}$$

where the point $(c, d)$ is an observation of rank $n-1$. When considering which of the observations $(a, b)$ and $(a', b')$ are the most extreme, the maximum condition says that if

$$\sup_{0<p<1} P_{n,p}(a,b) < \sup_{0<p<1} P_{n,p}(a',b')$$

then observation $(a, b)$ is the most extreme of the two, and vice versa (Barnard, 1947).

In an iteration of this algorithm, there will most of the time be two points that are equally as extreme (as is the case in (2)). However, both a single most extreme point and four most extreme points may occur. The first scenario happens when $(a, b)$ is on the diagonal, that is, if $(a, b) = (n_1 - a, n_2 - b)$. Then we would only include one of the two terms on the right-hand side of (2). The second scenario usually happens when $n_1 = n_2$, so $(a, b)$ and $(b, a)$ get the same rank. Here, we have to add all the joint probabilities for the four points (instead of the two points in (2)) and give them all the same rank (see Appendix for R-code). In testing, no other scenarios occurred. However, for large enough $n_1$ and $n_2$ one can probably observe other amounts of equally extreme points. Thus, a more general definition of $p_{n,p}(a, b)$ for a given amount of equally extreme points should be implemented:

$$P_{n,p} = \Bigg( \sum_{\substack{(x,y) \text{ has} \\ \text{rank n}}} P_p(x,y) \Bigg) + P_{n-1,p}(c,d)$$

The p-value of Barnard's CSM test is

$$p(x_1, x_2) = \sup_{0<p<1} P_{n,p}(x_1, x_2),$$

where $n$ is the rank of $(x_1, x_2)$.

## 3.5 Example of Barnard's CSM Test

With these three concepts, let us now build Barnard's test statistic for concrete values. We let $n_1 = 3$ and $n_2 = 2$. Condition C requires that the most extreme points will always be $(3, 0)$, and from S it then follows that $(0, 2)$ will be just as extreme. The p-value for both these observations will be

$$\sup_{0<p<1} P_{1,p}(3,0) = \sup_{0<p<1} \Big( P_p(3,0) + P_p(0,2) \Big) = 0.0625,$$

where the maximum was attained at $p = 0.5$.

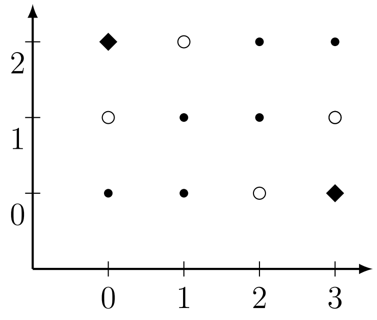

**Figure 2:** Diamond: p-value calculated, circle: points to consider next, the rest are dots

|   | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 2 | 0.0625 | – | – | – |
| 1 | – | – | – | – |
| 0 | – | – | – | 0.0625 |

**Table 1:** Development of table of p-values for Barnard's CSM test

The only points to consider next will, because of C, be $(2, 0)$ and $(3, 1)$. Since

$$\begin{aligned}
\sup_{0<p<1} P_{2,p}(2,0) &= \sup_{0<p<1} \Big( P_p(2,0) + P_p(1,2) + P_{1,p}(3,0) \Big) \\
&= 0.25 \\
&> 0.1998 \\
&= \sup_{0<p<1} \Big( P_p(3,1) + P_p(0,1) + P_{1,p}(3,0) \Big) \\
&= \sup_{0<p<1} P_{2,p}(3,1),
\end{aligned}$$

we choose $(3, 1)$ and $(0, 1)$ as the most extreme value of the remaining points and we plug in their p-values of 0.1998 in our table (Table 2). This time the maxima were attained at $p = 0.276$ and $p = 0.724$. We still consider $(2, 0)$ as a candidate for the next highest rank, but condition C also adds the point $(0, 0)$ (and therefore $(3, 2)$), to our consideration (Figure 3).

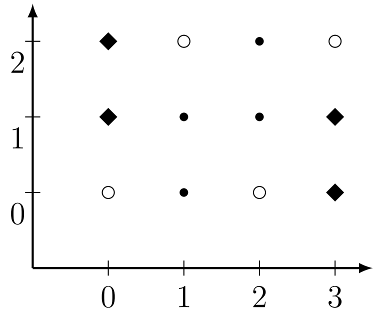

**Figure 3:** Diamond: p-value calculated, circle: points to consider next and the rest are dots

|   | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 2 | 0.0625 | – | – | – |
| 1 | 0.1998 | – | – | 0.1998 |
| 0 | – | – | – | 0.0625 |

**Table 2:** Development of table of p-values for Barnard's CSM test

We have that

$$\begin{aligned}\sup_{0<p<1} P_{3,p}(2,0) &= \sup_{0<p<1}\Big(P_p(2,0)+P_p(1,2)+P_{2,p}(3,0)\Big)\\ &= 0.375\end{aligned}$$

and

$$\begin{aligned}\sup_{0<p<1} P_{3,p}(0,0) &= \sup_{0<p<1}\Big(P_p(0,0)+P_p(3,2)+P_{2,p}(3,0)\Big)\\ &= 1.\end{aligned}$$

Hence, we set the p-value of the observations $(2,0)$ and $(1,2)$ to $0.375$ (maximum at $p = 0.5$) and add both $(1,0)$ and $(2,1)$ to consideration.

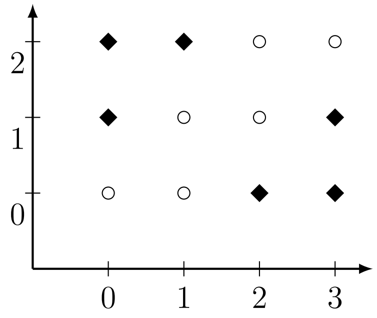

**Figure 4:** Diamond: p-value calculated, circle: points to consider next and the rest are dots

|   | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 2 | 0.0625 | 0.375 | – | – |
| 1 | 0.1998 | – | – | 0.1998 |
| 0 | – | – | 0.375 | 0.0625 |

**Table 3:** Development of table of p-values for Barnard's CSM test

Since

$$\begin{aligned}\sup_{0<p<1} P_{4,p}(1,0) &= \sup_{0<p<1}\Big(P_p(1,0)+P_p(2,2)+P_{3,p}(2,0)\Big)\\ &= 0.568,\end{aligned}$$

$$\begin{aligned}\sup_{0<p<1} P_{4,p}(2,1) &= \sup_{0<p<1}\Big(P_p(2,1)+P_p(1,1)+P_{3,p}(2,0)\Big)\\ &= 0.75,\end{aligned}$$

and

$$\begin{aligned}\sup_{0<p<1} P_{4,p}(0,0) &= \sup_{0<p<1}\Big(P_p(0,0)+P_p(3,2)+P_{3,p}(2,0)\Big)\\ &= 1,\end{aligned}$$

we choose $(1,0)$ with a corresponding p-value of 0.568 (Table 4) (where $p = 0.349$ and $p = 0.651$ maximized the expression).

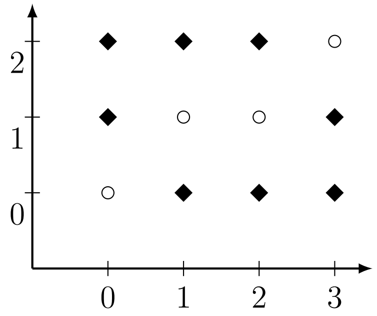

**Figure 5:** Diamond: p-value calculated, circle: points to consider next and the rest are dots

|   | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 2 | 0.0625 | 0.375 | 0.568 | – |
| 1 | 0.1998 | – | – | 0.1998 |
| 0 | – | 0.568 | 0.375 | 0.0625 |

**Table 4:** Development of table of p-values for Barnard's CSM test

There are no more points to add to our list of considered points, so the last comparison we have to make is between the points $(3,2)$ and $(2,1)$. They have the respective p-values of 1 (maximized at $p = 0.5$) and 0.9375 (maximized at $p = 0.115$ and $p = 0.885$). We have now obtained a table of p-values for all possible outcomes when $n_1 = 3$ and $n_2 = 2$ (Table 5).

|   | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 2 | 0.0625 | 0.375 | 0.568 | 1 |
| 1 | 0.1998 | 0.9375 | 0.9375 | 0.1998 |
| 0 | 1 | 0.568 | 0.375 | 0.0625 |

**Table 5:** Final table of p-values for Barnard's CSM test

### Proof that Barnard's CSM Test Produces a Valid P-value

By the way the p-values $p(x_1, x_2)$ are constructed for Barnard's CSM test,

$$\begin{aligned} p(x_1, x_2) &= \sup_{0<p<1} P_p(\text{rank of } (X_1, X_2) \leq \text{rank of } (x_1, x_2)) \\ &= \sup_{0<p<1} P_p(p(X_1, X_2) \leq p(x_1, x_2)), \end{aligned}$$

which is exactly the supremum method (with rejection of $H_0$ for small values of the statistic). This shows that the p-values are valid.

## 3.6 The Dispute Between Barnard and Fisher

In Barnard's first article on his new test, he claims that the CSM test is, in fact, more powerful than Fisher's exact test (Barnard, 1945a). Later that year, Fisher replied with his own article, where he (almost ideologically) argues why Barnard is wrong. To give the reader some context, they are talking about testing whether two groups of animals have the same probability of dying. He writes: "In my view the notion of defining the level of significance by repeated sampling of the same population is misleading in the theory small samples just because it allows of the uncritical inclusion in the denominator of material irrelevant to a critical judgment of what has been observed. In 2 of the 64 cases enumerated above, all animals die or all survive. The fact that such an unhelpful outcome as these might occur, or must occur with a certain probability, is surely no reason for enhancing our judgment of significance in cases where it has not occurred..." (Fisher, 1945). So Fisher argues that, for example, the case where all animals die is irrelevant to the null hypothesis and that such cases therefore, just inflates the denominator. Thus, the p-values gathered from the CSM test are too low.

Barnard later responds with introducing two examples of studies, one where $H_0$: "blue-eyed people are just as likely to catch colds as non-blue-eyed people", and one where the $H_0$: "taking a daily dose of XYZ does not affect the chance of having a cold" (Barnard, 1945b). In the first example, if all our subjects catch a cold, we learn nothing. However, in the second example, we learn that a daily dose of XYZ is unnecessary. This illustrates how, in some experiments, a result where all outcomes are observed in one category, still might be helpful. According to Barnard, their debate continued in privacy and, sadly, in a later article he acknowledged that Fisher was right and retracted his test (Barnard, 1949). Decades later, he elaborates: "Fisher finally persuaded me that before calculating the P-value one should classify the possible results of an experiment into sets which are equally informative about the point at issue" (Barnard, 1992). We will now compare the tests in practice.

# 4 Testing the Methods in Practice

In this section, we will look at several examples of the probability of rejecting the null hypothesis for the different tests. The way I did this started with implementing the methods in R to obtain p-values for all combinations of $x_1$ and $x_2$ in the region $[0, n_1] \times [0, n_2]$. I then used the following fact:

Let $\alpha$ be the chosen significance level. Then the probability of rejecting $H_0$ as a function of $p_1$ and $p_2$ is called the *power function*,

$$\begin{aligned}\gamma(p_1, p_2) &= P_{p_1,p_2}(\text{reject } H_0)\\ &= P_{p_1,p_2}(p(X_1, X_2) \leq \alpha))\\ &= \sum_{p(x_1,x_2)\leq\alpha} P_{p_1,p_2}(X_1 = x_1, X_2 = x_2).\end{aligned}$$

Hence, to compute power, we simply sum joint probabilities of outcomes having p-value less than or equal to $\alpha$. We will first look at the case where $H_0$ is true. Here we wish to see values lower than the significance level, since we hope for valid tests. Then we look at cases where $p_1 \neq p_2$, where higher values are better.

In this section, the supremum method and Barnard's CSM test will usually perform very similarly. Thus, for easier reading, when I write "the supremum methods", I am referring to both the supremum method and the CSM test.

When we now are going to compare the tests against each other, it will be interesting to see: How well the asymptotic method holds up against the valid methods, how different the performance of Barnard's tests is compared to the supremum method, and if Fisher's test is as conservative as people have it.

## 4.1 Comparisons of Validity

Since we are testing for methods that produce valid p-values, $p_1 = p_2 = p$ in all examples in this subsection. All results can be found in Table 6, which will be used for reference in this section. The calculations were done in R with all methods built from scratch in the same way as described in Section 3 (see Appendix for details).

In our first example we let $n_1 = n_2 = 10$, $\alpha = 0.05$ and $p = 0.5$. We see that the rate of rejection is the same for the asymptotic method, the supremum method, and Barnard's CSM test, but Fisher's test gets a much lower value. Note that all rejection rates are below $\alpha$, which is good. The approximation of the test statistic to a standard normal distribution, seems to be good at the $\alpha = 0.05$ level of significance. However, the approximation should get worse as we go further out in the tails of the distribution.

Let us try $\alpha = 0.01$ and keep everything else the same. This example shows that the asymptotic test is not valid (we get a rate of rejection larger than $\alpha$). The supremum methods are still equal. Fisher exact test is once again much more conservative than the two other valid methods. When decreasing $n_1$ to 5, and set $\alpha$ back to 0.05, we again see that the asymptotic method gets a probability of rejecting higher than alpha. This time, because the approximations get worse as $n$ decreases.

The last parameter we can adjust is $p$. Let's set it to 0.3. A lower value of $p$ results in more values being close to zero so that the data give less information. To keep the rejection rate below $\alpha$, the tests have to be more conservative when rejecting null hypothesis. We should therefore, see less null hypotheses rejected. As we can see from Table 6, this point becomes even more apparent when setting $p = 0.1$.

Now that we have seen the effects on the probability of rejection, the different parameters have by themselves let us try to combine them. First, when $p = 0.5$, $\alpha = 0.01$ and $n_1 = 3$, $n_2 = 6$, we unsurprisingly see that the asymptotic method does not hold up. Also, note that there are no rejections made by Fisher's test. In the next line, when $p = 0.5$, $\alpha = 0.01$ and $n_1 = 3$, $n_2 = 12$, we finally get an example where the supremum method gave a different result than Barnard's test. Barnard's power is closer to 0.01. This may indicate better powers for the CSM test in the next subsection. The asymptotic method especially seems to be bad when there is a big relative difference between $n_1$ and $n_2$.

But is the asymptotic method okay if $n_1 = n_2 = 20$ at $\alpha = 0.01$? As we see, this is not the case; the probability of rejection is still way above $\alpha$.

| Row | $p$ | $\alpha$ | $n_1$ | $n_2$ | Asymptotic | Fisher | Supremum | Barnard |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.5 | 0.05 | 10 | 10 | 0.04219 | 0.01278 | 0.04219 | 0.04219 |
| 2 | 0.5 | 0.01 | 10 | 10 | 0.01278 | 0.002577 | 0.007956 | 0.007956 |
| 3 | 0.5 | 0.05 | 5 | 10 | 0.1575 | 0.01471 | 0.04797 | 0.04797 |
| 4 | 0.3 | 0.05 | 10 | 10 | 0.03711 | 0.01188 | 0.03711 | 0.03711 |
| 5 | 0.1 | 0.05 | 10 | 10 | 0.009040 | 0.001147 | 0.009040 | 0.009040 |
| 6 | 0.5 | 0.01 | 3 | 6 | 0.01367 | 0 | 0.003906 | 0.003906 |
| 7 | 0.5 | 0.01 | 3 | 12 | 0.3278 | 0.0007935 | 0.0009766 | 0.004822 |
| 8 | 0.5 | 0.01 | 20 | 20 | 0.2246 | 0.004253 | 0.007273 | 0.007273 |

**Table 6:** Probability of rejection for each method for different combinations of parameters

## 4.2 Comparisons of Power

In this section, we want to compare the power of the different methods. By definition, this is the probability of rejecting the null hypothesis when $H_0$ is false, i.e., when $p_1 \neq p_2$. The numbers in this subsection can be found in Table 7.

In our first example we let $p_1 = 0.3$, $p_2 = 0.7$, $\alpha = 0.05$ and $n_1 = n_2 = 10$.

Fisher's test is the least powerful, and all the others gave the same power (Table 7). The result of changing the significance level to 0.01 can be seen in the second line. As could be expected, now the asymptotic method is by far the most powerful. Fisher's test still has the lowest power, and the supremum methods are equal.

In the last subsection, we learned that deviations from $p = 0.5$ increased the probability of rejecting $H_0$. We can expect this to be the case when offsetting the mean of $p_1$ and $p_2$. In row 3, we see that this drastically affected the asymptotic method, somewhat Fisher's test, but there is almost no decrease for the supremum tests.

From row 4 to 5, we see that increasing the relative difference between $n_1$ and $n_2$ to 0.6 separates Fisher from the supremum methods. However, it does not separate the latter two. In row 6, we decrease the significance level to $\alpha = 0.01$. Here, Barnard's method becomes more powerful than the other valid methods. Offsetting $p_1$ and $p_2$, the CSM test does even better compared to the supremum test (row 7, Table 7).

Let us see if similar results can be obtained when increasing $n_1$ and $n_2$ while keeping the relative difference the same (row 8–11). We now see that even at the $\alpha = 0.05$ level of significance, the CSM test is slightly more powerful than the supremum test. Also note that since $n$ is higher than before, Fisher is not too far behind. When lowering the level of significance, Fisher actually is now more powerful than the supremum method. This time, offsetting $p_1$ and $p_2$ did now increase the difference in power between the supremum methods. When we increased $n$ from row 7 to row 8, we see that the advantage in the power of Barnard's test increased. Doing the same three steps again, only with $n_1 = n_2 = 20$, Barnard still gives the best results (of the tests that give valid p-values).

In the next nine rows (14–22), this whole cycle is repeated, only with a bigger gap between $p_1$ and $p_2$. This did not change much of what we already knew. Barnard beats the other valid methods in all rows, except for the rows where they are equal. In some of the rows, we also see that Fisher is more powerful than the supremum method.

It was difficult to find combinations where the supremum method was more powerful than the CSM test, but in rows 23 and 25, we see that it is at least possible.

| Row | $p_1$ | $p_2$ | $\alpha$ | $n_1$ | $n_2$ | Asymptotic | Fisher | Supremum | Barnard |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.3 | 0.7 | 0.05 | 10 | 10 | 0.4185 | 0.2433 | 0.4185 | 0.4185 |
| 2 | 0.3 | 0.7 | 0.01 | 10 | 10 | 0.2433 | 0.1071 | 0.1830 | 0.1830 |
| 3 | 0.2 | 0.6 | 0.01 | 10 | 10 | 0.2527 | 0.1012 | 0.1945 | 0.1945 |
| 4 | 0.3 | 0.7 | 0.05 | 3 | 9 | 0.7489 | 0.1766 | 0.1766 | 0.1766 |
| 5 | 0.3 | 0.7 | 0.05 | 3 | 12 | 0.9526 | 0.1751 | 0.2065 | 0.2065 |
| 6 | 0.3 | 0.7 | 0.01 | 3 | 12 | 0.8788 | 0.02916 | 0.03527 | 0.08672 |
| 7 | 0.2 | 0.6 | 0.01 | 3 | 12 | 0.7820 | 0.01003 | 0.01087 | 0.04275 |
| 8 | 0.3 | 0.7 | 0.05 | 5 | 20 | 0.9886 | 0.3211 | 0.3441 | 0.3493 |
| 9 | 0.3 | 0.7 | 0.01 | 5 | 20 | 0.9873 | 0.1152 | 0.08512 | 0.1707 |
| 10 | 0.2 | 0.6 | 0.01 | 5 | 20 | 0.9888 | 0.08341 | 0.04274 | 0.1429 |
| 11 | 0.3 | 0.7 | 0.05 | 20 | 20 | 0.5435 | 0.5994 | 0.7052 | 0.7114 |
| 12 | 0.3 | 0.7 | 0.01 | 20 | 20 | 0.5357 | 0.3722 | 0.4519 | 0.4519 |
| 13 | 0.2 | 0.6 | 0.01 | 20 | 20 | 0.8311 | 0.4130 | 0.4939 | 0.4939 |
| 14 | 0.2 | 0.8 | 0.05 | 3 | 12 | 0.9960 | 0.4332 | 0.5124 | 0.5124 |
| 15 | 0.2 | 0.8 | 0.01 | 3 | 12 | 0.9836 | 0.1407 | 0.1671 | 0.2859 |
| 16 | 0.1 | 0.7 | 0.01 | 3 | 12 | 0.9565 | 0.06198 | 0.06535 | 0.1843 |
| 17 | 0.2 | 0.8 | 0.05 | 5 | 20 | 0.9677 | 0.6990 | 0.7440 | 0.7447 |
| 18 | 0.2 | 0.8 | 0.01 | 5 | 20 | 0.9677 | 0.3860 | 0.3621 | 0.4998 |
| 19 | 0.1 | 0.7 | 0.01 | 5 | 20 | 0.9986 | 0.3707 | 0.2581 | 0.4917 |
| 20 | 0.2 | 0.8 | 0.05 | 20 | 20 | 0.5785 | 0.9610 | 0.9808 | 0.9815 |
| 21 | 0.2 | 0.8 | 0.01 | 20 | 20 | 0.5782 | 0.8745 | 0.9162 | 0.9162 |
| 22 | 0.1 | 0.7 | 0.01 | 20 | 20 | 0.9644 | 0.9156 | 0.9456 | 0.9456 |
| 23 | 0.3 | 0.7 | 0.1 | 10 | 15 | 0.9573 | 0.5605 | 0.6168 | 0.6055 |
| 24 | 0.3 | 0.7 | 0.01 | 10 | 15 | 0.8434 | 0.1999 | 0.2251 | 0.2261 |
| 25 | 0.3 | 0.7 | 0.001 | 10 | 15 | 0.6076 | 0.06000 | 0.08815 | 0.07492 |

**Table 7:** Power of each method for different combinations of parameters

# 5 Discussing the Results

We have seen that of the three valid methods, Fisher produced the lowest power. The asymptotic method was the most powerful. However, since it is not valid, we have to be cautious when choosing to use this method, especially for low $n_1$ and $n_2$ and for low significance levels. In most situations, the supremum methods do equally good. However, for lower significance levels, and when the difference between $n_1$ and $n_2$, and between $p_1$ and $p_2$ was big, Barnard's test seemed to do slightly better. I found no cases where Fisher's test was more powerful than Barnard's. However, there were rare cases where it outperformed the supremum method. It is also important to note that these power differences become minuscule when $n_1 \times n_1$ reaches levels of $\sim 100$.

Based on these results, both the CSM test and (to a lesser extent) the supremum method should see an increase in usage when conducting experiments on two by two contingency tables for low $n_1$ and $n_2$.

# 6 Conclusion

In this thesis, we have reviewed some common and some not so common tests for two by two contingency tables. Of the valid methods, the two least used methods turned out to be most powerful. It seems, however, that Fisher and later also Barnard did not value power in itself, stating that including unhelpful outcomes would automatically increase the power. To me, this is almost an absurd stance since it takes the objectivity out of statistics by introducing feelings on what are helpful and what are not helpful observations. Hence, despite even Barnard himself disagreeing, I hope to see more people using these two tests in scientific experiments.

# 7 References

- Barnard, G. A. (1992). Statistics and OR–Some Needed Interactions. *The Journal of the Operational Research Society*, Vol. 43, p. 787–795.

- Barnard, G. A. (1949). Statistical Inference. *The Journal of the Operational Research Society*, Vol. 11, p. 115–149.

- Barnard, G. A. (1947). Significance Tests for 2 × 2 Tables. *Biometrika*, Vol. 34. p. 123–138.

- Barnard, G. A. (1945a). A New Test for 2 × 2 Tables. *Nature*, Vol. 156, p. 177.

- Barnard, G. A. (1945b). A New Test for 2 × 2 Tables. *Nature*, Vol. 156, p. 783.

- Casella G. and Berger, R. (2008). Chapter 8: Hypothesis testing. *Statistical Inference: Second Edition*, p. 397–399. USA: Duxbury.

- Fisher R. A. (1945). A New Test for 2 × 2 Tables. *Nature*, Vol. 156, p. 388.

# A Appendix: R-Code

```r
##################### Asymptotic Method ######################

calc_t <- function(x1, x2, n1 = 10, n2 = 10){
  t <- (x1/n1 - x2/n2)/sqrt((1/n1 + 1/n2)*((x1+x2)/
        (n1 + n2))*(1-(x1+x2)/(n1 + n2)))
  if (is.nan(t)){ t <- 0 }
  t
}

asympt_test <- function(x1, x2){
  t <- -abs(calc_t(x1, x2))
  2*pnorm(t)
}
```

```r
##################### Fisher's Exact Test ######################

fisher_test <- function(x1, x2, n1 = 10, n2 = 10){
  p_value <- 0
  c <- x1 + x2
  table <- make_t_value_table(n1, n2)
  t_0 <- table[x1+1, x2+1]

  for (i in 0:n1){
    for (j in 0:n2){
      if (i + j == c){
        t <- table[i+1, j+1]
        if (t_0 - eps < t){
          p_value <- p_value + dhyper(i, n1, n2, c)
        }
      }
    }
  }
  p_value
}

####################### Supremum Method ########################

calc_prob<- function(x1, x2, p, n1 = 10, n2 = 10){
  dbinom(x1, n1, p)*dbinom(x2, n2, p)
}

eps <- 1e-8

make_t_value_table <- function(n1 = 10, n2 = 10){
  table <- matrix(0, n1+1, n2+1)

  for (x1 in 1:(n1+1)){
    for (x2 in 1:(n2+1)){
      t <- abs(calc_t(x1-1, x2-1, n1, n2))
      table[x1, x2] <- t
    }
  }
  table
}

sum_probs <- function(x1, x2, p, n1 = 10, n2 = 10, table){
  t_0 <- abs(calc_t(x1, x2, n1, n2))
  inds <- which(table > t_0 - eps, arr.ind = TRUE)
        #indexes of extreme observations
  sum <- 0
  for (i in 1:nrow(inds)){
    sum <- sum + calc_prob(inds[i,1] - 1, inds[i,2] - 1, p
            , n1, n2)
  }
  sum
}

supremum_test <- function(x1, x2, n1 = 10, n2 = 10){
  table <- make_t_value_table(n1, n2)
  p_vec <- seq(0, 1, 1/1000)
  p_max <- 0

  for (p in p_vec){
    p_val <- sum_probs(x1, x2, p, n1, n2, table = table)
    p_max <- max(p_max, p_val)
  }
  p_max
}

##################### Barnard's CSM Test #####################

barnard_prob_calc <- function(x1, x2, n1, n2, x11 = 0, x22=0,
        en_like = FALSE, to_like = FALSE, prev = rep(0, 1001)){
  #prev contains sum of all the more extreme joint
  #probabilities, one sum for each value of p
  y1 <- n1 -x1
  y2 <- n2 - x2
  p_vec <- seq(0, 1, 1/1000)
  p_vals <- rep(0, 1001)

  if(en_like){
    for (i in 1:length(p_vec)){
      p_vals[i] <- calc_prob(x1, x2, p_vec[i], n1, n2) + prev[i]
    }
    i <- which(p_vals == max(p_vals), arr.ind = TRUE)[1]
    return(list(p_vals, p_vec[i]))
  }
  if (to_like){

    y11 <- n1 - x11
    y22 <- n2 - x22
    for (i in 1:length(p_vec)){
      p_vals[i] <- calc_prob(x1, x2, p_vec[i], n1, n2) +
        calc_prob(y1, y2, p_vec[i], n1, n2) +
        calc_prob(x11, x22, p_vec[i], n1, n2) +
        calc_prob(y11, y22, p_vec[i], n1, n2) + prev[i]
    }
    i <- which(p_vals == max(p_vals), arr.ind = TRUE)[1]
    return(list(p_vals, p_vec[i]))
  }
  else{
    for (i in 1:length(p_vec)){
      p_vals[i] <- calc_prob(x1, x2, p_vec[i], n1, n2) +
        calc_prob(y1, y2, p_vec[i], n1, n2) + prev[i]
    }
    i <- which(p_vals == max(p_vals), arr.ind = TRUE)
    return(list(p_vals, p_vec[i]))
  }
}

barnard_matrix <- function(n1, n2){
  p_matrix <- matrix(0, n1+1, n2+1)
  p_matrix[1, n2+1] <- max(barnard_prob_calc(0, n2, n1,
        n2)[[1]])
  p_matrix[n1+1, 1] <- max(barnard_prob_calc(n1, 0, n1,
        n2)[[1]])

  prev <- barnard_prob_calc(0, n2, n1, n2)[[1]]

  queue_matrix <- matrix(0, n1+1, n2+1)
  queue_matrix[1, n2+1] <- 2
  queue_matrix[n1+1, 1] <- 2
  queue_matrix[1, n2] <- 1
  queue_matrix[2, n2+1] <- 1
  queue_matrix[n1, 1] <- 1
  queue_matrix[n1+1, 2] <- 1

  while (min(queue_matrix) != 2){
    calc_matrix <- matrix(100, n1+1, n2+1)
    for (i in 0:n1+1){
      for (j in 0:n2+1){
        if (queue_matrix[i, j] == 1){
          calc_matrix[i, j] <- 1
        }
      }
    }

    for (i in 0:n1+1){
      for (j in 0:n2+1){
        if (calc_matrix[i, j] == 1){
          calc_matrix[i, j] <- max(barnard_prob_calc(i-1, j-1,
                n1, n2, prev =prev)[[1]])
        }
      }
    }

    #matrix with indices for minimums (could be multiple)
    min_mat <- which(calc_matrix == min(calc_matrix),
            arr.ind = TRUE)
    if (nrow(min_mat) == 1){
      x1 <- as.numeric(which(calc_matrix == min(calc_matrix),
            arr.ind = TRUE)[1, 1])
      x2 <- as.numeric(which(calc_matrix == min(calc_matrix),
            arr.ind = TRUE)[1, 2])
      p_matrix[x1, x2] <- max(barnard_prob_calc(x1-1, x2-1, n1,
            n2, en_like = TRUE, prev =prev)[[1]])
      queue_matrix[x1, x2] <- 2

      prev <- barnard_prob_calc(x1-1, x2-1, n1, n2,
            en_like = TRUE, prev =prev)[[1]]

    }
    if (nrow(min_mat) == 2){
      x1 <- as.numeric(which(calc_matrix == min(calc_matrix),
            arr.ind = TRUE)[1, 1])
      x2 <- as.numeric(which(calc_matrix == min(calc_matrix),
            arr.ind = TRUE)[1, 2])
      y1 <- n1+1 - (x1-1)
      y2 <- n2+1 - (x2-1)

      p_matrix[x1, x2] <- max(barnard_prob_calc(x1-1, x2-1, n1,
            n2, prev =prev)[[1]])
      p_matrix[y1, y2] <- max(barnard_prob_calc(y1-1, y2-1, n1,
            n2, prev=prev)[[1]])
      queue_matrix[x1, x2] <- 2
      queue_matrix[y1, y2] <- 2

      if (x2 != n2+1){
        if (queue_matrix[x1, x2+1] == 0){
          if (x1 == n1+1){
            queue_matrix[x1, x2+1] <- 1
            queue_matrix[y1, y2-1] <- 1
          }

          else if (queue_matrix[x1+1, x2+1] != 1){
            queue_matrix[x1, x2+1] <- 1
            queue_matrix[y1, y2-1] <- 1
          }
        }
      }

      if (x1 != n1+1){
        if (queue_matrix[x1+1, x2] == 0){
          if (x2 == n2+1){
            queue_matrix[x1+1, x2] <- 1
            queue_matrix[y1-1, y2] <- 1
          }

          else if (queue_matrix[x1+1, x2+1] != 1){
            queue_matrix[x1+1, x2] <- 1
            queue_matrix[y1-1, y2] <- 1
          }
        }
      }

      if (x2 != 1){
        if (queue_matrix[x1, x2-1] == 0){
          if (x1 == 1){
            queue_matrix[x1, x2-1] <- 1
            queue_matrix[y1, y2+1] <- 1
          }

          else if (queue_matrix[x1-1, x2+1] != 1){
            queue_matrix[x1, x2-1] <- 1
            queue_matrix[y1, y2+1] <- 1
          }
        }
      }

      if (x1 != 1){
        if (queue_matrix[x1-1, x2] == 0){
          if (x2 == 1){
            queue_matrix[x1-1, x2] <- 1
            queue_matrix[y1+1, y2] <- 1
          }
      else if (queue_matrix[x1-1, x2-1] != 1){
        queue_matrix[x1-1, x2] <- 1
        queue_matrix[y1+1, y2] <- 1
      }
    }
  }
  prev <- barnard_prob_calc(x1-1, x2-1, n1, n2,
        prev =prev)[[1]]
}
if (nrow(min_mat) == 4){
  x1 <- as.numeric(which(calc_matrix == min(calc_matrix),
        arr.ind = TRUE)[1, 1])
  x2 <- as.numeric(which(calc_matrix == min(calc_matrix),
        arr.ind = TRUE)[1, 2])
  y1 <- n1+1 - (x1-1)
  y2 <- n2+1 - (x2-1)

  x11 <- as.numeric(which(calc_matrix == min(calc_matrix),
        arr.ind = TRUE)[2, 1])
  x22 <- as.numeric(which(calc_matrix == min(calc_matrix),
        arr.ind = TRUE)[2, 2])
  y11 <- n1+1 - (x11-1)
  y22 <- n2+1 - (x22-1)

  p_matrix[x1, x2] <- max(barnard_prob_calc(x1-1, x2-1, n1,
        n2, x11 = x11-1, x22 = x22-1, to_like = TRUE,
        prev =prev)[[1]])
  p_matrix[y1, y2] <- max(barnard_prob_calc(y1-1, y2-1, n1,
        n2, x11 = y11-1, x22 = y22-1, to_like = TRUE,
        prev=prev)[[1]])
  queue_matrix[x1, x2] <- 2
  queue_matrix[y1, y2] <- 2

  p_matrix[x11, x22] <- max(barnard_prob_calc(x1-1, x2-1,
        n1, n2, x11 = x11-1, x22 = x22-1, to_like = TRUE,
        prev=prev)[[1]])
  p_matrix[y11, y22] <- max(barnard_prob_calc(y1-1, y2-1,
            n1, n2, x11 = y11-1, x22 = y22-1, to_like = TRUE,
            prev=prev)[[1]])
      queue_matrix[x11, x22] <- 2
      queue_matrix[y11, y22] <- 2

      if (x2 != n2+1){
        if (queue_matrix[x1, x2+1] == 0){
          if (x1 == n1+1){
            queue_matrix[x1, x2+1] <- 1
            queue_matrix[y1, y2-1] <- 1
          }

          else if (queue_matrix[x1+1, x2+1] != 1){
            queue_matrix[x1, x2+1] <- 1
            queue_matrix[y1, y2-1] <- 1
          }
        }
      }

      if (x1 != 1){
        if (queue_matrix[x1-1, x2] == 0){
          if (x2 == 1){
            queue_matrix[x1-1, x2] <- 1
            queue_matrix[y1+1, y2] <- 1
          }

          else if (queue_matrix[x1-1, x2-1] != 1){
            queue_matrix[x1-1, x2] <- 1
            queue_matrix[y1+1, y2] <- 1
          }
        }
      }


      if (x22 != n2+1){
        if (queue_matrix[x11, x22+1] == 0){
          if (x11 == n1+1){
            queue_matrix[x11, x22+1] <- 1
            queue_matrix[y11, y22-1] <- 1
          }

          else if (queue_matrix[x11+1, x22+1] != 1){
            queue_matrix[x11, x22+1] <- 1
            queue_matrix[y11, y22-1] <- 1
          }
        }
      }

      if (x22 != 1){
        if (queue_matrix[x11, x22-1] == 0){
          if (x11 == 1){
            queue_matrix[x11, x22-1] <- 1
            queue_matrix[y11, y22+1] <- 1
          }
          else if (queue_matrix[x11, x22-1] != 1){
            queue_matrix[x11, x22-1] <- 1
            queue_matrix[y11, y22+1] <- 1
          }
        }
      }

      if (x11 != 1){
        if (queue_matrix[x11-1, x22] == 0){
          if (x22 == 1){
            queue_matrix[x11-1, x22] <- 1
            queue_matrix[y11+1, y22] <- 1
          }

          else if (queue_matrix[x11-1, x22-1] != 1){
            queue_matrix[x11-1, x22] <- 1
            queue_matrix[y11+1, y22] <- 1
          }
        }
      }
      if (x11 != n1+1){
        if (queue_matrix[x11+1, x22] == 0){
          if (x22 == n2+1){
            queue_matrix[x11+1, x22] <- 1
            queue_matrix[y11-1, y22] <- 1
          }

          else if (queue_matrix[x11+1, x22+1] != 1){
            queue_matrix[x11+1, x22] <- 1
            queue_matrix[y11-1, y22] <- 1
          }
        }
      }
      prev <- barnard_prob_calc(x1-1, x2-1, n1, n2, x11 = x11-1,
            x22 = x22-1, to_like = TRUE, prev = prev)[[1]]
    }
  }
  for (i in nrow(p_matrix)){
    for (j in ncol(p_matrix)){
      if (p_matrix[i, j] == 0){
        p_matrix[i, j] = 1
      }
    }
  }
  p_matrix
}

######################### Comparisons ##########################

asympt_matrix <- function(n1, n2){
  asympt <- matrix(0, n1+1, n2+1)

  for (i in 1:(n1+1)){
    for (j in 1:(n2+1)){
      asympt[i, j] <- asympt_test(i-1, j-1)
    }
  }
```

```r
  asympt
}

fisher_matrix <- function(n1, n2){
  fish <- matrix(0, n1+1, n2+1)

  for (i in 1:(n1+1)){
    for (j in 1:(n2+1)){
      fish[i, j] <- fisher_test(i-1, j-1, n1, n2)
    }
  }
  fish
}

supremum_matrix <- function(n1, n2){
  sup <- matrix(0, n1+1, n2+1)

  for (i in 1:(n1+1)){
    for (j in 1:(n2+1)){
      sup[i, j] <- supremum_test(i-1, j-1, n1, n2)
    }
  }
  sup
}

power_asympt <- function(p1, p2, alpha = 0.05, n1=10, n2=10){
  asympt <- asympt_matrix(n1, n2)
  sum <- 0
  for (x1 in 1:(n1+1)){
    for (x2 in 1:(n2+1)){
      if (asympt[x1, x2] <= alpha){
        sum <- sum + dbinom(x1-1, n1, p1)*dbinom(x2-1, n2, p2)
      }
    }
  }
  sum
}
```

```
power_fish <- function(p1, p2, alpha = 0.05, n1 = 10, n2 = 10){
  fish <- fisher_matrix(n1, n2)
  sum <- 0
  for (x1 in 1:(n1+1)){
    for (x2 in 1:(n2+1)){
      if (fish[x1, x2] <= alpha){
        sum <- sum + dbinom(x1-1, n1, p1)*dbinom(x2-1, n2, p2)
      }
    }
  }
  sum
}

power_sup <- function(p1, p2, alpha = 0.05, n1 = 10, n2 = 10){
  sup <- supremum_matrix(n1, n2)
  sum <- 0
  for (x1 in 1:(n1+1)){
    for (x2 in 1:(n2+1)){
      if (sup[x1, x2] <= alpha){
        sum <- sum + dbinom(x1-1, n1, p1)*dbinom(x2-1, n2, p2)
      }
    }
  }
  sum
}

power_barnard <- function(p1, p2, alpha = 0.05, n1=10, n2=10){
  barn <- barnard_matrix(n1, n2)
  sum <- 0
  for (x1 in 1:(n1+1)){
    for (x2 in 1:(n2+1)){
      if (barn[x1, x2] <= alpha){
        sum <- sum + dbinom(x1-1, n1, p1)*dbinom(x2-1, n2, p2)
      }
    }
  }
  sum
}

comparison_of_power <- function(p1, p2, alpha = 0.05,
        n1 = 10, n2 = 10){
  asympt <- power_asympt(p1, p2, alpha, n1, n2)
  fish <- power_fish(p1, p2, alpha, n1, n2)
  sup <- power_sup(p1, p2, alpha, n1, n2)
  barn <- power_barnard(p1, p2, alpha, n1, n2)

  cat("Asymptotic: ", asympt, "\n")
  cat("Fisher: ", fish, "\n")
  cat("Supremum: ", sup, "\n")
  cat("Barnard: ", barn, "\n")
}
```

In the example of Barnard's CSM test I simply ran the inside of the while-loop in "Barnard-matrix" multiple times to monitor each iteration of the different matrices and used "prev[[2]]" to access the value of $p$ that maximized the expression in each iteration.

NTNU

Kunnskap for en bedre verden