PS C236A/ Stat C239A
Section 4 Notes

1 Neyman vs Fisher

Fisher's hypothesis tests are about making inferences about particular treatment effects, on particular units, at a particular time. We make no reference to a larger population, nor how the units came to be a part of our experiment. Neyman's version of inference is different. Uncertainty is conceptualized as arising from repeatedly applying treatment, creating a population of treatment effects. Neyman's framework allows us to test the weak null of no average effect, while Fisher's method is more suited to the sharp null of no effect at all.

The advantage of the Fisherian hypotheses is that they are exact, while Neyman tests rely on the central limit theorem to form confidence intervals and p-values. In many settings, particularly in large experiments with standard methods of randomization, Neyman methods produce inferences that are valid. When an experiment is small or the data is unsual in some way, randomization inference will guarantee accurate results.

Some of the fiercest battles in the early years of statistics were over which type of inference was better. The Neyman framework won out, though of late, Fisherian inference seems to be resurging.

Cox highlights some of the issues by focusing on 2 questions, about outcomes (under two treatment conditions) $\mathbf{x}$ and $\mathbf{y}$:

1. Do $\mathbf{x}$ and $\mathbf{y}$ differ more than would be expected if they were samples from the same infinite population?
2. Do $\mathbf{x}$ and $\mathbf{y}$ differ more than would be expected if they had been formed by selecting random permutations of the outcomes?

First, when N is large, the answers to the two questions are nearly identical. The objection to 2 is that it involves reference to an infinite population which is an artificial construct, is not clearly defined, and which is in general certainly not a super- population of individuals to which one would like to apply the conclusions of the analysis.

What about the adequacy of the sharp null hypotheis? Cox concludes:

> ...if substantial variations in treatment effect from unit to unit do occur, one's understanding of the experimental situation will be very incomplete until the basis of this variation is discovered and any extension of the conclusions to a general set of experimental units will be hazardous. The mean treatment effect, averaged over all units in the experiment, or over the finite population of units from which they are randomly drawn, may in such cases not be too helpful. Particularly if appreciable systematic treatment-unit

interactions are suspected, the experiment should be set out so these may be detected and explained.

2 Basic Setup

There are N units divided into S strata or *blocks*, which are formed on the basis of pre-treatment characteristics. A unit is an opportunity to apply or withold the treatment. Note that this setup doesn't encompass group randomized experiments, but the notation can easily be expanded to address these types of designs. There are n_s units in stratum s for $s = 1, ..., S$, so $N = \sum n_s$.

Write $Z_{si} = 1$ if the ith unit in stratum s receives the treatment and write $Z_{si} = 0$ if this unit receives control. Write m_s for the number of treated units in stratum s, so $m_s = \sum_{i=0}^{n_s} Z_{si}$ and $0 \leq m_s \leq n_s$.

$$\mathbf{Z} = \begin{bmatrix} Z_{11} \\ Z_{12} \\ \vdots \\ Z_{1,n_1} \\ Z_{21} \\ \vdots \\ Z_{S,n_s} \end{bmatrix} = \begin{bmatrix} \mathbf{Z_1} \\ \vdots \\ \mathbf{Z_S} \end{bmatrix}$$

2.1 A Bit on Experimental Design

"Block what you can, randomize what you can't."

A disadvantage of complete randomization is that when variations among the experimental units are large, the treatment comparisons do not have good precision. Blocking is an effective way to reduce experimental error. The experimental units are divided into more homogeneous groups called blocks. Better precion can be achieved by comparing the treatments within blocks.

Treatment Assignment The most common assignment mechanism fixes the number of m_s in stratum s. Let Ω be the set containing $K = \Pi_{s=1}^{s} \binom{n_s}{m_s}$ possible treatment assigments $\mathbf{z}$. In the most common experiments, each of these K possible assignments is given the same probability, prob$(\mathbf{Z} = \mathbf{z}) = 1/K$ for all $\mathbf{z}$ in Ω.

3 Example: An Experiment in the Republic of Georgia

In the lead up to the 2008 parliamentary elections in Georgia, there were efforts by the democracy promotion community to encourage citizen monitoring of elections. We conducted a program evaluation of one such effort: a simple information campaign to give voters the information necessary for filing a formal complaint with civil society groups or election officials if they witnessed problems on election day. The intervention consisted of sending canvassers to knock on doors and hand out fliers in randomly selected precincts.

Ths structure of randomization was as follows.

- 36 rural precincts were in blocks of 2, one treatment and one control. So for these precincts, $m_s = 1$ and $n_s = 2$.
- 48 urban precincts were in blocks of 4, two in treatment and and two in control ($m_s = 2$ and $n_s = 4$).

How big is Ω?

```
> choose(2,1)^18 * choose(4,2)^12
[1] 5.706304e+14
```

So every potential treatment assignment has a $\frac{1}{5.7\times10^{14}}$ probabilty of being assigned.

Let's create a function that will assign treatment repeatedly. Note that due to block randomization, we will randomize within blocks.

```
#Let's create a treatment assignment function
treat.assign <- function(treat,blocks=NA){
  if(length(unique(blocks))==1){
    treat.vector <- sample(treat)
  }
  else{
    #randomize within blocks using tapply
    treat.vector <- tapply(treat,blocks,sample)
  #tapply returns a list, to turn into a vector, use "unlist"
  treat.vector <- unlist(treat.vector)
  }
  return(treat.vector)
}
```

Let's create our distribution of treatment vectors. We could compute all 5.7×10^{14} treatment vectors, but to save on computing time, we can sample a large number of possible treatment vectors to get "close-to-exact" p-values. If our experiment were smaller, then exhaustive enumeration would be better.

Let's use the `replicate` function to assign treatment 5,000 times and generate our Ω:

```
omega <- replicate(5000,treat.assign(treat,blocks))
#We only want unique treatment assignments, so let's get rid of
duplicates.
#Specifying "margin=2" means  keep unique columns
omega <- unique(omega,MARGIN=2)
```

4 The Sharp Null

The most common hypothesis associated with randomization inference is the sharp null of no effect for all units. Under this null hypthesis, assigning treatment is like shuffling meaningless labels. A unit labeled as "treated" will have the exact same outcome as a unit labeled as "control". Under

the null, the units' responses are *fixed* and the only random element is the meaningless rotation of labels.

When testing the null hypothesis of no effect, the response of the *i*th unit in stratum s can be written r_{si} and the vector of responses is $\mathbf{r}$. A **test statistic** $t(\mathbf{Z}, r)$ is a quantity computed from the treatment assignment $\mathbf{Z}$ and the response r.

The most commonly used test-statistic is the point estimate for the average treatment effect. In a block randomized experiment, the differences within blocks are summed, and each block difference is weighted by the proportion of units in the block:

$$\sum_{s=1}^{S} \frac{n_s}{N} \sum_{i=1}^{n_s} \left\{ \frac{Z_{si} r_{si}}{m_s} - \frac{(1 - Z_{si}) r_{si}}{n_s - m_s} \right\}$$

To compute the significance level for any given test statistic, we simply calculate the proportion of treatment assignments $\mathbf{z}$ in Ω giving values of $t(\mathbf{z}, \mathbf{r})$ greater than or equal to the observed T, namely:

$$\text{prob}\{t(\mathbf{Z}, \mathbf{r} \geq T\} = \frac{|\{\mathbf{z} \in \Omega : t(\mathbf{z}, \mathbf{r}) \geq T\}|}{K}$$

The above p-value is for a one-tailed test. What about a two-tailed test? There is some disagreement in the literature about this, but Rosenbaum recommends simply doubling the one-tailed p-value.

4.1 Which Test Statistic?

The default test-statistic is the difference in means, but many other are possible. The difference in means test statistic will have low power in the presence of outliers, skewed distributions, or heavy tailed distributions. As a result, sometimes a more powerful test is desirable.

Rank Sum Test One common alternative to the difference in means statistic is the Wilcoxon *rank sum* test. In a completely randomized experiment, the responses are ranked from smallest to largest. If all N responses were different numbers, the ranks woud be the numbers $1, 2, ..., N$. If some of the responses were equal, then the average of their ranks would be used. Write q_i for the rank of r_i, and write $\mathbf{q} = (q_i, ..., q_N)^T$. The rank sum statistic is simply the sum of the ranks of the treated observations, i.e. $t(\mathbf{z}, \mathbf{r}) = \mathbf{Z^T q}$. The advantage of the rank transformation is that it improves the power of the test in the presence of skewed distributions, outliers and symmetric, heavy-tailed distributions.

Stratified Rank Sum Test For block randomized experiments, one easy extension of the rank sum test is to calculate the rank sum test separately in each strata and take the sum of these S rank sums as the test statistic.

Aligned Rank Statisic According to Hodges and Lehmann (1962), a more efficient rank test for block randomized experiment is the aligned rank statistic. For this statistic, subtract the mean of each stratum from the responses in that stratum, creating "aligned responses" . Rank the aligned

responses without regard to block. The aligned rank statistic is the sum of the aligned ranks in the treated group.

5 Example, Part 2

First, let's examine the effect of the intervention on the the main dependent variable, r_i which is a dummy variable for whether or not a complaint was filed on election day in precinct i. This variable is heavily skewed, as the base rate of complaints is very low. The vast majority of precincts are scored as 0.

Conventional Inference As a check, let's see what the standard methods would tell us. One can estimate the following logit model:

$$P(Y_i = 1|T, X) = \Phi(T_i\alpha + X_s\beta) \tag{1}$$

where Y_i is an indicator variable if a complaint is filed in precinct i, T_i is a treatment indicator variable, and X_s is a vector of dummy variables indicating block membership.

```
summary(glm(y~treat + blocks, family=binomial))

Call:
glm(formula = y ~ treat + blocks, family = binomial)

Deviance Residuals:
       Min          1Q      Median          3Q         Max
-1.177e+00  -1.768e-05  -2.107e-08  -2.107e-08   1.177e+00

Coefficients:
              Estimate Std. Error  z value Pr(>|z|)
(Intercept) -2.027e+01  1.249e+04   -0.002    0.999
treat        4.054e+01  1.249e+04    0.003    0.997
....
2: In glm.fit(x = X, y = Y, weights = weights, start = start, etastart = eta
  fitted probabilities numerically 0 or 1 occurred
```

Unfortunately, *likelihood maximization fails to converge because of quasi-complete separation.* Inference is not possible with the standard method.

Randomization Inference Let's start with the difference-in-means test statistic:

```
##First let's create a function that computes the difference-in-means
##(taking into account the block structure of the experiment)
diff.means <- function(y,treat,blocks){
  #use tapply to calculate the within block averages
  ave.treated <- tapply(y[treat==1],blocks[treat==1],mean)
```

```
  ave.control <- tapply(y[treat==0],blocks[treat==0],mean)
  wt <- tapply(y,blocks,length)/length(y)
  test.stat <- sum(wt*(ave.treated-ave.control))
  return(test.stat)
}
```

Our observed T is 0.12.

Now let's compute the entire distribution of $t(\mathbf{z}, \mathbf{r})$ under the null hypothesis. To do this, we'll loop over the 5000 $\mathbf{z}$ vectors and compute the test statistic repeatedly, creating a distribution of test statistics.

```
#Now let's compute the entire distribution of test-statistics under the null
#create a matrix to hold the test statistics
test.stat.dist <- matrix(nrow=ncol(omega))
#loop over omega, caculate the test stat every iteration
for (i in 1:ncol(omega)){
  treat.fake <- omega[,i]
  fake.test.stat <- diff.means(y,treat.fake,blocks)
  test.stat.dist[i] <- fake.test.stat
}
```

Let's look at the distribution of $t(\mathbf{z}, \mathbf{r})$ under the null hypothesis, in Figure 1.

```
#Let's plot the randomization distribution
plot(density(test.stat.dist),col="blue", main="Randomization Distribution")
#where does the true test statistic fall?
abline(v=true.test.stat,col="red",lwd=2)
```

What's our (close to) exact p-value?

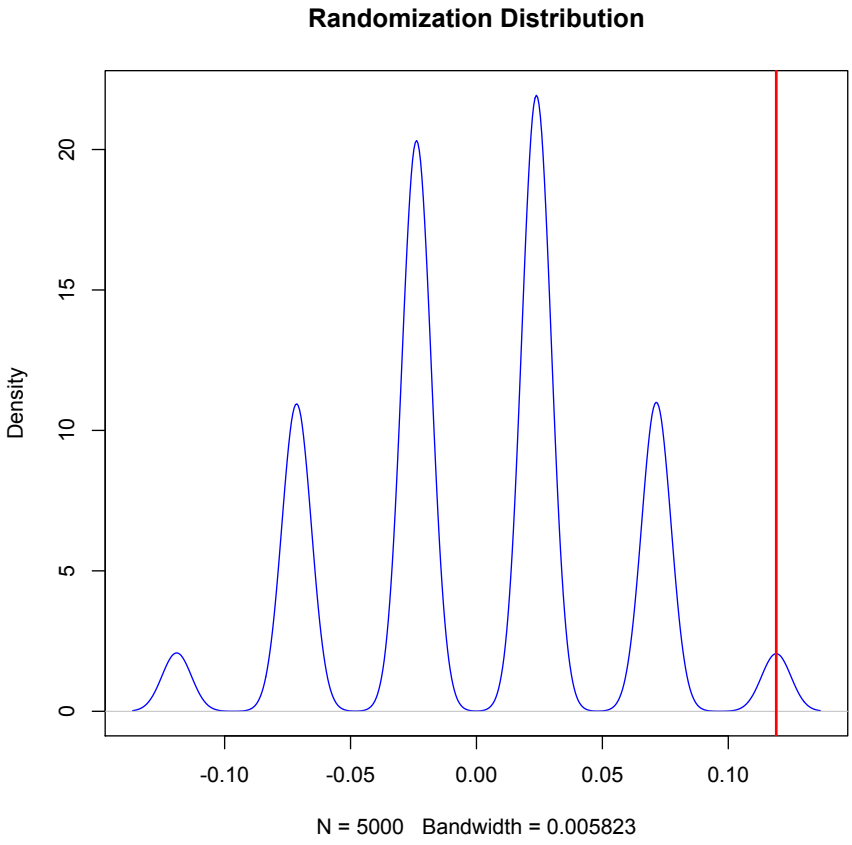
```
 sum(test.stat.dist>=true.test.stat)/ncol(omega)
[1] 0.03
```

Rank Sum Test To see how the rank sum test works, we'll look at a different dependent variable: voter turnout. Note that the point estimate for the effect of the treatment is -3.1 percent. Because this is a block randomized experiment, we'll use the stratified version of the rank sum test.

Like before, we'll create a function to compute our test statistic:

```
#create a rank sum function
strat.ranksum <- function(y,treat,blocks){
  #rank within strata
  ranks <- unlist(tapply(y,blocks,rank))
  #sum ranks of treated units
  ranksum <- sum(ranks[treat==1])
  return(ranksum)
}
```

Figure 1: The randomization distribution of the difference in means test statistic under the null hypothesis. The dependent variable is an indicator variable for whether or not a complaint was filed.



The observed value of our test statistic is 85. As before, we need to compare that number to distribution of possible test statistics under the null. The code to do this is virtually unchanged from what was printed above; just swap `diff.means` for `strat.ranksum`.

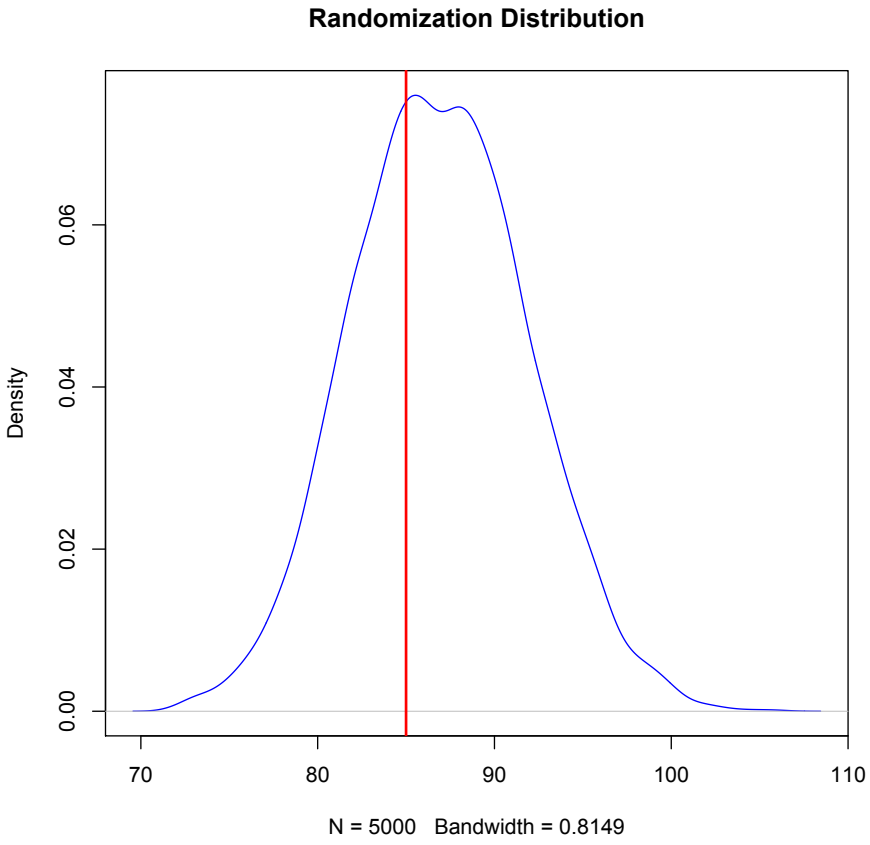
The distribution, along with the actual test statistic, is displayed in figure 2. The (close to) exact p value is .4. We cannot reject the null at conventional levels.

6 Covariate Adjustment

Write $\tilde{\epsilon}(\cdot)$ for a function that creates residuals ($\tilde{\epsilon}(\mathbf{r}) = \mathbf{e}$) from $\mathbf{r}$, which are the oucomes under the null hypothesis, and $\mathbf{X}$, which is a matrix of covariates. $\tilde{\epsilon}(\cdot)$ can be a simple linear model, some non-parametric smoother such as lowess, a robust linear model, etc. The point of adjustment is to reduce dispersion in $\mathbf{r}$, so choose $\tilde{\epsilon}(\cdot)$ with that goal in mind.

Remember that under the null hypothesis, nothing is stochastic except for the shuffling of treatment assignment labels. As a result $\mathbf{e}$ is a fixed quantity, not a random variable or a by-product of estimation. $\mathbf{e}$, however, may be less dispersed than $\mathbf{r}$ because some of the variation in $\mathbf{r}$ will have been captured by $\mathbf{X}$.

Figure 2: The randomization distribution of the stratified rank sum test statistic under the null hypothesis. The dependent variable is voter turnout (percent).



Under the null hypothesis, since $\mathbf{r_T} = \mathbf{r_c}$, $\mathbf{e_T} = \mathbf{e_C}$. So once can simply use the test statistic $t(\mathbf{z}, \mathbf{e})$ instead of $t(\mathbf{z}, \mathbf{r})$. With $\mathbf{e}$ in hand, just proceed as you would with $\mathbf{r}$.

7 Example, Part 3

Recall that the p-value on voter turnout, using the rank sum test, was .4, which is much larger than conventional levels of statistical significance. Let's see if reducing the considerable dispersion in voter turnout will increase the sensitivity of our test.

To keep things simple, we will use a simple linear regression to generate our residuals. The goal is to reduce dispersion, so I'll include a number of covariates: voter turnout in the previous parliamentary election, a dummy variable indicating rural or urban, the number of registered voters, percent supporting the president's party in the previous election, and block dummies.

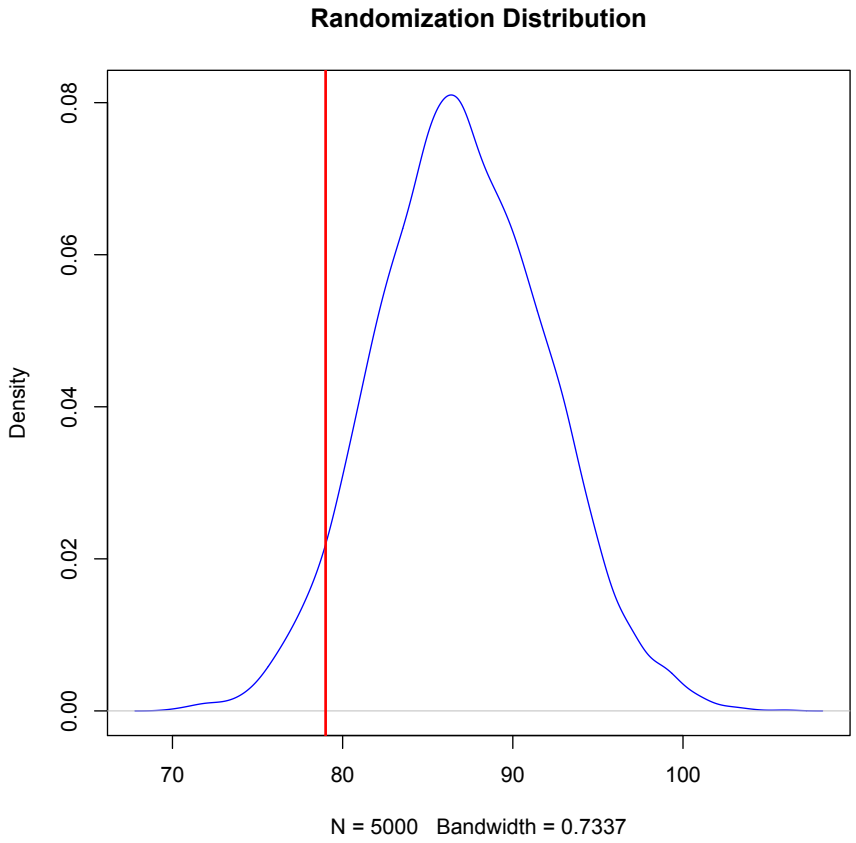
```
> #what's the dispersion in Y before adjustment
> sd(y)
[1] 15.2738
```

```
>
> #grab the residuals
> y <- lm(turnout08~turnout2006+urban+num.reg.voters
+pctPresParty2006+block,data=data)$residuals
> #what's the dispersion before adjustment?
> sd(y)
[1] 8.692608
```

Our $\tilde{\epsilon}(\cdot)$ succeeded in reducing $\mathbf{r}$ by almost half!

The observed value of our test statistic $t(\mathbf{z}, \mathbf{e})$ is 79. As before, we need to compare that number to distribution of possible test statistics under the null. We use the code from section 5.

Figure 3: The randomization distribution of the stratified rank sum test statistic under the null hypothesis. The dependent variable is residualized voter turnout (percent).



The distribution, along with the actual test statistic, is displayed in figure 3. The (close to) exact p value is .06, which is much smaller than our previous result. Thus, we see some evidence that the intervention reduced turnout.