Stat 205: Introduction to Nonparametric Statistics

Lecture 04: Permutation Inference

Instructor David Donoho; TA: Yu Wang

Lady Tasting Tea, 1

"A lady declares that by tasting a cup of tea made with milk, she can discriminate whether the milk or the tea infusion was first added to the cup."

- Intro, Chapter 2: "The Design of Experiments" 1935 RA Fisher.
- Famous in Statistics Profession

Lady Tasting Tea, 2

Fisher proposed *randomization experiment*

- Prepare eight cups of tea with two 'treatments'
  - In four, add tea first
  - In the other four, add milk first
  - Hold constant everything else (cup size, cup temperature, etc.)
- Present eight cups in **random order**
- Lady tastes; predicts treatment
  She knows: the 8 cups include four 'milk first'; four 'tea first' treatments.

Lady Tasting Tea, 3

- Lady's Theory – H_1: she can predict treatments
- H_0 lady cannot taste difference
- Lady's prediction P_i: $i = 1, \ldots, 8$.
- Actual treatment T_i: $i = 1, \ldots, 8$.

```
      [,1]   [,2]   [,3]  [,4]   [,5]   [,6]   [,7]   [,8] 
truth "tea"  "milk" "tea" "tea"  "milk" "milk" "milk" "tea"
pred  "milk" "milk" "tea" "milk" "tea"  "tea"  "milk" "tea"
```

- Prediction Score: $S_i = 1_{\{P_i = T_i\}}$
- Test statistic: S, number of correct predictions

$$S = \sum_{i=1}^{8} S_i = \sum_{i=1}^{8} 1_{\{\text{Lady's } i\text{-th prediction correct}\}}$$

What is null distribution of S?, 1

Would-be Classical 'argument':

- Each prediction score S_i equally likely $\{0, 1\}$.
- Different prediction scores independent
- $S \sim Bin(n = 8, p = 1/2)$

Why it fails:

- There are *exactly 4* of each treatment
- Sequence of treatments *not independent*
- View Lady's predictions as fixed and nonrandom;
- Independence of prediction scores *fails*

  Example to think about:

  - Lady *each time* says "Milk First";
  - She *always* makes exactly 4 mistakes, deterministically;
  - *Not* binomial $bin(8, 1/2)$.

What is null distribution of S?, 2

Fisher's argument:

- Prediction score $S_i = S_i(T_i, P_i)$ function of Treatment T_i and Prediction P_i.
- Lady's prediction viewed as fixed.
- Under H_0
  - (T_i) formally independent of (P_i).
  - All orders of treatment $(T_i)_{i=1}^{8}$ are equally likely
  - There are $\binom{8}{4}$ binary strings of length 8 with 4 1's.

Null Distribution of S

- Depends on (P_i), which is viewed as *fixed* and *known*.
- If lady predicts 'Tea' 8 times, she must be making *exactly 4* mistakes.
- If lady predicts 'Tea' Four times and 'Milk' Four times, she can be making variable number of mistakes.

Example Null Distribution of S

- Suppose Lady predicts 'Tea' four times and 'Milk' four times.
- For the four cups where the tea was poured first, select s to say "tea" correctly, and $4 - s$ to say "milk" incorrectly.
- She is making $4 - s$ mistakes on the 'Tea First' cups, but also makes $4 - s$ mistakes on the 'MilkFirst' cups.

```
> m = 0:4
> probability = (choose(4,m)*choose(4,4-m))/choose(8,4)
> s = 2*m;
> data.frame(score=s,probability=round(probability,digits=3))
  score probability
1     0       0.014
2     2       0.229
3     4       0.514
4     6       0.229
5     8       0.014
```

Monte-Carlo Simulation of Null Distribution,1

- Repeatedly:
  - Choose a would-be arrangement uniformly at random
  - Compute the score the Lady's predictions would receive under the would-be arrangement.
- Calculate empirical distribution

```
> arrangements <- NULL
> for(i in 1:10){
+   arrangements <- rbind(arrangements,sample(base,replace=FALSE));
+ }
> arrangements
      [,1]   [,2]   [,3]   [,4]   [,5]   [,6]   [,7]   [,8]  
 [1,] "milk" "tea"  "tea"  "milk" "milk" "tea"  "milk" "tea" 
 [2,] "tea"  "milk" "milk" "milk" "tea"  "tea"  "milk" "tea" 
 [3,] "milk" "milk" "tea"  "tea"  "milk" "tea"  "tea"  "milk"
 [4,] "tea"  "milk" "tea"  "tea"  "milk" "milk" "milk" "tea" 
 [5,] "milk" "tea"  "milk" "tea"  "tea"  "tea"  "milk" "milk"
 [6,] "tea"  "tea"  "milk" "milk" "tea"  "milk" "tea"  "milk"
 [7,] "tea"  "tea"  "milk" "tea"  "tea"  "milk" "milk" "milk"
 [8,] "milk" "tea"  "tea"  "milk" "tea"  "tea"  "milk" "milk"
 [9,] "tea"  "milk" "tea"  "tea"  "milk" "milk" "tea"  "milk"
[10,] "tea"  "milk" "tea"  "tea"  "milk" "milk" "tea"  "milk"
```

Monte-Carlo Simulation of Null Distribution, 2

```
> base <- c(rep("tea",4),rep("milk",4))
> base
[1] "tea"  "tea"  "tea"  "tea"  "milk" "milk" "milk" "milk"
> truth <- sample(base,replace=FALSE)
> pred  <- sample(base,replace=FALSE)
> rbind(truth=truth,pred=pred)
      [,1]   [,2]   [,3]  [,4]   [,5]   [,6]   [,7]   [,8]
truth "tea"  "milk" "tea" "tea"  "milk" "milk" "milk" "tea"
pred  "milk" "milk" "tea" "milk" "tea"  "tea"  "milk" "tea"
>  succ=rep(0,9);
> for(iMC in 1:1000){
+   mightbe <- sample(base,replace=FALSE);
+   S <- sum(mightbe==pred);
+   succ[S+1]= succ[S+1]+1
+ }
> succ
[1]  13   0 242   0 512   0 219   0  14
> print(succ/sum(succ))
[1] 0.013 0.000 0.242 0.000 0.512 0.000 0.219 0.000 0.014
>
```

Note: These numbers agree with the earlier exact calculation

General Permutation Inference

- Specify statistic in functional terms, i.e. a function $T(\cdot)$ (say) that, given a dataset $\mathcal{D}$ evaluates $T(\mathcal{D})$.
- Specify how permutations operate on data $\mathcal{D}_\pi$
- Enumerate all permutations π, and corresponding statistic $t_\pi = T(\mathcal{D}_\pi)$.
- Compute null distribution:

$$P_0\{T = t|X\} = \frac{\#\{\pi : t_\pi = t\}}{n!}.$$

- Suppose t_{obs} is the observed statistic and large values indicate departure from null.
- p-value $= P_0\{T \geq t_{obs}|X\}$
- Reject if p unusually small.

Example of Permutation Inference, 1
Two Sample Problem

- Two Samples $X = (X_i)_{i=1}^n$, $Y = (Y_j)_{j=1}^m$.
- Statistic $T = \bar{X} - \bar{Y}$.
- Dataset $= \mathcal{D} = [I \ Z]$, where
  - $Z = (Z_k)_{k=1}^{n+m} = (X_1, \dots, X_n, Y_1, \dots, Y_m)$
  - $I = (I_k)_{k=1}^{n+m} = (0, ..., 0, 1, ..., 1)$ are *group labels*;
  - Statistic written $T(\mathcal{D}) = Ave_{\{k:I_k=1\}}[Z_k] - Ave_{\{k:I_k=0\}}[Z_k]$.
- Permutations π act on $n+m$-vectors
- Permuted Dataset $\mathcal{D}_\pi = [I_\pi Z]$; "shuffle labels"

Example of Permutation Inference, 2 Wilcoxon Rank-Sum Two-Sample

Something we already have discussed!

- Two Samples $X = (X_i)_{i=1}^n$, $Y = (Y_j)_{j=1}^m$.
- Null Hypothesis: Generating Distributions of X, Y the same.
- Ranks in combined sample: $R(X_i|X \cup Y)$, $R(Y_j|X \cup Y)$
- Statistic $W^+ = \sum_{j=1}^m R(Y_j|X \cup Y)$.
- Dataset $= \mathcal{D} = [I\ Z]$, where
  - $Z = (Z_k)_{k=1}^{n+m} = (R(X_1|X \cup Y), \ldots, R(X_n|X \cup Y), R(Y_1|X \cup Y), \ldots, R(Y_n|X \cup Y))$
  - $I = (I_k)_{k=1}^{n+m} = (0, \ldots, 0, 1, \ldots, 1)$ are *group labels*;
  - Statistic written $T(\mathcal{D}) = \sum_{\{k:I_k=1\}}[Z_k]$.
- Permutations π act on $n+m$-vectors
- Permuted Dataset $\mathcal{D}_\pi = [I_\pi\ Z]$; "shuffle labels"
- Permutation null: $P_0\{W^+ \leq w|X, Y\} = \frac{\#\{\pi : W^+(\mathcal{D}_\pi) \leq w\}}{(n+m)!}$.

The permutation null distribution of W^+ as defined in this lecture, is *precisely* the null distribution we discussed earlier.

Example of Permutation Inference, 3
Wilcoxon Signed-Rank One-Sample

Something we already have discussed!

- One Sample $X = (X_i)_{i=1}^n$.
- Hypothesis: Symmetry of generative distribution $P\{X_i \le -x\} = P\{X_i \ge x\}$, $\forall x$.
- Ranks of absolute values: $R|X_i| \equiv R(|X_i||(|X_i|)_{i=1}^n)$
- Signed-Rank statistic $W = \sum_{i=1}^n \mathrm{sign}(X_i) R|X_i|$.
- Dataset $= \mathcal{D} = [I\ Z]$, where
  - $Z = (Z_k)_{k=1}^n = (R|X_1|, \ldots, R|X_n|)$
  - $I = (I_k)_{k=1}^n = (I_i)$ are signs $I_i = \mathrm{sign}(X_i)$;
  - Statistic written $T(\mathcal{D}) = \sum_i I_i \cdot Z_i$.
- Permutations π act on n-vectors
- Permuted Dataset $\mathcal{D}_\pi = [I_\pi\ Z]$; "shuffle signs"

The permutation null distribution of the signed-rank W as defined in this lecture, is *precisely* the null distribution we discussed earlier.

Advantages of Permutation Inference

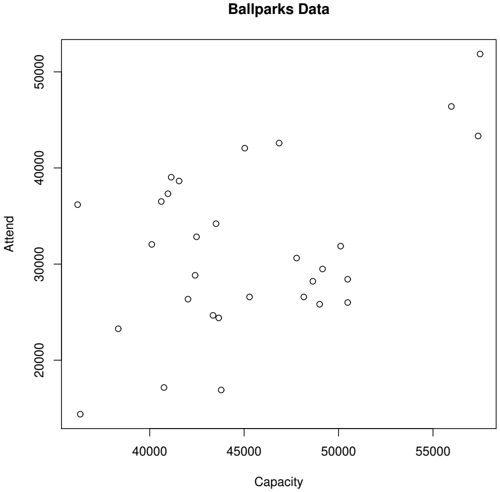
- Works with (pretty much) *any statistic*
- Nonparametric: no need to specify assumed distribution
- Exact: under distributional assumptions *weaker than independence*, and *far broader than normality*, has correct α-level.

Ballparks Dataset

- 30 Major League Baseball Teams
- Capacity - nominal stadium capacity
- Attend - Average Annual Attendance 2006.

```
> head(bp)
         Team Capacity Attend
1  NY Yankees    57492  51858
2  LA Dodgers    55971  46400
3     NY Mets    57387  43327
4   St. Louis    46851  42588
5   LA Angels    45031  42059
6 Chicago Cubs   41138  39040
> tail(bp)
         Team Capacity Attend
25   Cleveland    43351  24667
26     Oakland    43653  24402
27  Pittsburgh    38334  23269
28 Kansas City    40755  17158
29   Tampa Bay    43785  16901
30     Florida    36323  14384
```

Apparent Correlation



(This had better be true:
build larger stadiums where crows are larger!)

Permutation Inference for Correlation

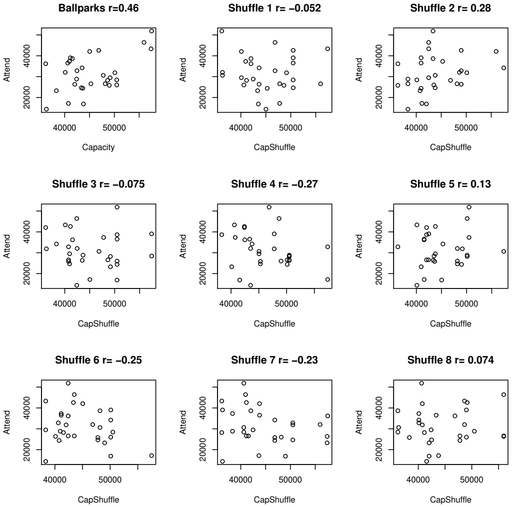
- Dataset $\mathcal{D} = [XY]$ has two columns of data
- Action of permutation π applied *only to* first column:

$$\mathcal{D}_\pi = [X_\pi Y]$$

- In this 'randomly shuffled' dataset:
  columns *stochastically independent*, hence *uncorrelated*:
  this is our null distribution

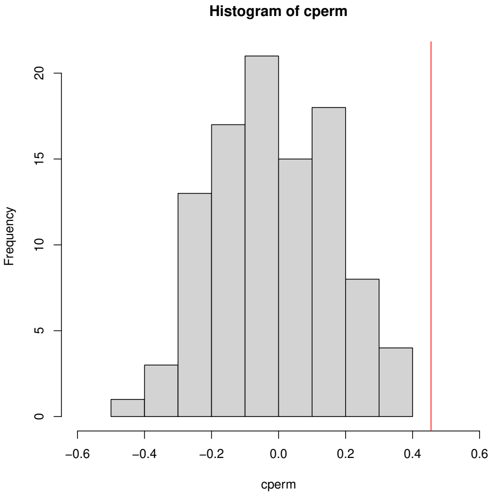
```
shuffle = function(X){X[sample(1:length(X),replace=T)]}
corr.shfl = cor(shuffle(Capacity),Attend)
```

Original and Shuffled Datasets



Shuffled datasets can have negative correlation.

Permutation Distribution



Observed Correlation of unshuffled data is $r = 0.46$ (red)

Implementation of permutation p-value in Correlation Case

```
corr.obs = cor(bp$Capacity[perm],bp$Attend)
corr.shuffle = function(perm){cor(bp$Capacity[perm],bp$Attend)}
makeperms = function(m,n){replicate(m,sample(1:n,replace=T))}
cperm = apply(makeperms(100,nrow(bp)),2,corr.shuffle)
p.value = mean( cperm > corr.obs)
```

The Bible Code Affair, 1

Statistical Science
1994, Vol. 9, No, 3, 429–438

Equidistant Letter Sequences in the Book of Genesis

Doron Witztum, Eliyahu Rips and Yoav Rosenberg

Abstract. It has been noted that when the Book of Genesis is written as two-dimensional arrays, equidistant letter sequences spelling words with related meanings often appear in close proximity. Quantitative tools for measuring this phenomenon are developed. Randomization analysis shows that the effect is significant at the level of 0.00002.

Key words and phrases: Genesis, equidistant letter sequences, cylindrical representations, statistical analysis.

Best selling issue of *Statistical Science*, ever.

The Bible Code Affair, 2

Bible Code

- Equsipaced Letter Sequences in Genesis
- Measure Closeness of Pairs of phrases
  - Phrase 1. Name of Rabbi
  - Phrase 2. Birth Date of Rabbi
- Statistical Significance is claimed!
- *Genesis* written millennia before Rabbis born!
- Published by Referees at *Statistical Science*

The Bible Code Affair, 3

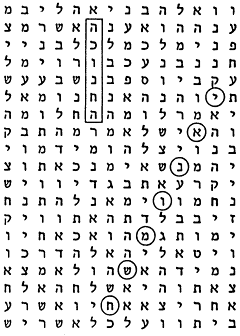


FIG. 3.

Text (L_i) where each L_i letter in Hebrew alphabet
Text presents right-to-left, starting upper right
Wraps at line feeds
Spaces and punctuation removed
ELS is a linear subsequence $(L_{a+bi})_{i=0}^{m}$ of letters
First ELS above spells Rabbi name
Second ELS is birthdate of that Rabbi

Should that be surprising?

The Bible Code Affair, 4

Table 1
The first list of personalities

Personality	Name	Date
1. The Ra'avad of Posquieres	רבי אברהם, הראב"ד	כ"ו כסלו, בכ"ו כסלו, כ"ו בכסלו
2. Rabbi Avraham, son of the Rambam	רבי אברהם	י"ח כסלו, בי"ח כסלו, י"ח בכסלו
3. Rabbi Avraham Ibn-Ezra	רבי אברהם, אבן עזרא, בן עזרא, הראב"ע	א' אדר א', בא' אדר א', א' באדר א'
4. Rabbi Eliyahu Bahur	רבי אליהו, הבחור, בעל הבחור	בו' שבט, ו' בשבט
5. Rabbi Eliyahu of Vilna	רבי אליהו, הגאון	ט"ו ניסן, בט"ו ניסן, ט"ו בניסן י"ה ניסן, בי"ה ניסן, י"ה בניסן י"ט תשרי, בי"ט תשרי, י"ט בתשרי
6. Rabbi Gershon Ashkenazi	רבי גרשון, הגרשני	י' אדר ב', בי' אדר ב', י' באדר ב'
7. Rabbi David Ganz	רבי דוד, דוד גנז, דוד גאנז, צמח דוד	ה' אלול, בה' אלול, ה' באלול
8. The Taz	רבי דוד, דוד הלוי, בעל הט"ז	כ"ו שבט, בכ"ו שבט, כ"ו בשבט
9. Rabbi Haim Ibn-Attar	רבי חיים, בן עטר, אבן עטר, אור החיים	ט"ו תמוז, בט"ו תמוז, ט"ו בתמוז י"ה תמוז, בי"ה תמוז, י"ה בתמוז
10. Rabbi Yehudah, son of the Rosh	רבי יהודה	י"ז תמוז, בי"ז תמוז, י"ז בתמוז
11. Rabbi Yehudah He-Hasid	רבי יהודה	י"ג אדר, בי"ג אדר, י"ג באדר

The Bible Code Affair, 5

The data mining procedure the authors carried out:

- List $\mathcal{P}$ of 50 pairs
  $\mathcal{P} = \{(w_1, w_2)\} = \{(Name_\ell, Birthdate_\ell), \ell = 1, \ldots, 50\}$
- For Name/Birthdate pair $p = (w_1, w_2)$ in $\mathcal{P}$ we have string lengths $\ell(w_1) =$ length of name, $\ell(w_2) =$ length of birthdate.
- Text $\mathcal{L} = (L_i)$
- Family of arithmetic progressions of length ℓ:
  $\mathcal{A}_\ell = \{a = (b_0 + b_1 i)_{i=0}^{\ell-1}\}$.
- ELS $\mathcal{L}|a$ where $a \in \mathcal{A}_\ell$, $\mathcal{L}|a = (L_{b_0+b_1 i})_{i=0}^{\ell-1}$.
- Metric of proximity $M(a_1, a_2)$ of two progressions.
- Data Mining Algorithm: Proximity measure of two ELS.

$$P = \min_{p \in \mathcal{P}} \min_{\mathcal{L}|a_1 = w_1} \min_{\mathcal{L}|a_2 = w_2} M(a_1.a_2)$$

here $\min_{\mathcal{L}|a_1=w_1}$ is short for $\min_{a_1 \in \mathcal{A}_{\ell(w_1)} \& \mathcal{L}|a_1 = w_1}$, etc.

The Bible Code Affair, 6

Role for Permutation Inference

- No parametric model for actual Hebrew Text in Genesis
- Metrics for closeness of pairs of equidistant letter sequences not well studied
- Prospect of evaluating 'statistical significance' by mathematical analysis seems hopeless.
- Authors propose we can evaluate 'statistical significance' by permutation analysis
- We create many pseudo-Genesis replications by permuting original text of Genesis randomly.
- Scan each one the same way the original was scanned
- Shuffled text maintains the same letter frequencies as original prior to permuting

The Bible Code Affair, 7

Permutation inference procedure the authors carried out:

- View data-mining outcome P from last slide as a *function of the text* $(L_i)_{i=1}^{I}$.
- Hence $P = P(\mathcal{D})$ where $\mathcal{D} = (L_i)$
- Permutation π acts on vectors of length I.
- Act on dataset $\mathcal{D}$ by shuffling letter order $\mathcal{D}_\pi = (L_{\pi(i)})_{i=1}^{I}$.
- $P_\pi = P(D_\pi)$.
- Permutation null distribution:

$$P_0\{P \leq q\} = \frac{\#\{\pi : P_\pi \leq q\}}{I!}.$$

- Monte-Carlo estimation of permutation p-value.

Shuffling of Genesis: the results, 1

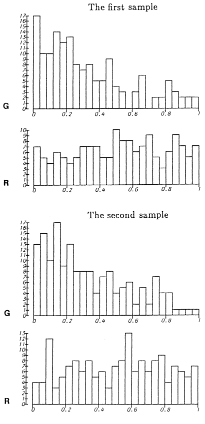


FIG. 4. *The distribution of value of $c(w, w')$ in the interval* $[0, 1]$.

G = (Name,Date) proximity in Genesis
R = (Name,Date) proximity in Randomly-Permuted Genesis text
First Sample = First list of 50 rabbis

Second Sample = Second list of 50 rabbis

Shuffling of Genesis: the results, 2

TABLE 3
Rank order of P_i among one million P_i^π

	P_1	P_2	P_3	P_4
G	453	5	570	4
R	619,140	681,451	364,859	573,861
T	748,183	363,481	580,307	277,103
I	899,830	932,868	929,840	946,261
W	883,770	516,098	900,642	630,269
U	321,071	275,741	488,949	491,116
V	211,777	519,115	410,746	591,503

P_i, $i = 1, \ldots, 4$ are measures of overall pair closeness across full text
G :Genesis, R random permutation of Genesis; T,I,W,U,V other texts
Monte-Carlo estimate of p-value of Genesis entry in column 4 is 4×10^{-6}.

MC p-values for other texts all look much larger, often $1/2$ or larger.

Properties of Permutation Inference

- Works with pretty much *any statistic*
- Nonparametric: no need to specify assumed distribution
- Computable by Monte-Carlo experiment
- Slogan: "Shuffle the data and apply statistic, many times"