

Lecture 19: ANOVA II

Pratheepa Jeganathan

05/15/2019

Recall

- ▶ One sample sign test, Wilcoxon signed rank test, large-sample approximation, median, Hodges-Lehman estimator, distribution-free confidence interval.
- ▶ Jackknife for bias and standard error of an estimator.
- ▶ Bootstrap samples, bootstrap replicates.
- ▶ Bootstrap standard error of an estimator.
- ▶ Bootstrap percentile confidence interval.
- ▶ Hypothesis testing with the bootstrap (one-sample problem.)
- ▶ Assessing the error in bootstrap estimates.
- ▶ Example: inference on ratio of heart attack rates in the aspirin-intake group to the placebo group.
- ▶ The exhaustive bootstrap distribution.

- ▶ Discrete data problems (one-sample, two-sample proportion tests, test of homogeneity, test of independence).
- ▶ Two-sample problems (location problem - equal variance, unequal variance, exact test or Monte Carlo, large-sample approximation, H-L estimator, dispersion problem, general distribution).
- ▶ Permutation tests (permutation test for continuous data, different test statistic, accuracy of permutation tests).
- ▶ Permutation tests (discrete data problems, exchangeability.)
- ▶ Rank-based correlation analysis (Kendall and Spearman correlation coefficients.)
- ▶ Rank-based regression (straight line, multiple linear regression, statistical inference about the unknown parameters, nonparametric procedures - does not depend on the distribution of error term.)
- ▶ Smoothing (density estimation, bias-variance trade-off, curse of dimensionality)
- ▶ Nonparametric regression (Local averaging, local regression, kernel smoothing, local polynomial, penalized regression)

- ▶ Cross-validation, Variance Estimation, Confidence Bands, Bootstrap Confidence Bands.
- ▶ Wavelets (wavelet representation of a function, coefficient estimation using Discrete wavelet transformation, thresholding - VishuShrink and SureShrink).
- ▶ One-way layout (general alternative (KW test), ordered alternatives)
- ▶ Two-way layout (complete block design (Friedman test))

ANOVA (Analysis of variance)

The Two-Way Layout

The Two-Way Layout

- ▶ Two factors A (treatment factor) and B (blocking factor).
 - ▶ A has k levels.
 - ▶ B has n levels.
 - ▶ primary interest is on factor A .
- ▶ Primary interest is about the relative locations (medians) of three or more populations (k populations) within each of the homogeneous groups.
- ▶ Data: Randomized complete block design (RCBD), Randomized block design (RBD), no observations in some blocks (Balanced incomplete block design), at least one observations in each cell.

The Two-Way Layout

Blocks	Treatments			
	1	2	...	k
1	X_{111}	X_{121}	...	X_{1k1}
	\vdots	\vdots		\vdots
2	$X_{11c_{11}}$	$X_{12c_{12}}$...	$X_{1kc_{1k}}$
	X_{211}	X_{221}	...	X_{2k1}
	\vdots	\vdots		\vdots
	$X_{21c_{21}}$	$X_{22c_{22}}$...	$X_{2kc_{2k}}$
	\vdots	\vdots		\vdots
n	X_{n11}	X_{n21}	...	X_{nk1}
	\vdots	\vdots		\vdots
	$X_{n1c_{n1}}$	$X_{n2c_{n2}}$...	$X_{nkc_{nk}}$

The Two-Way Layout

- ▶ $N = \sum_{i=1}^n \sum_{j=1}^k c_{ij}$ observations.
- ▶ c_{ij} - observations from the combination of the i -th block with the j -th treatment.
- ▶ $X_{ijc_{ij}}$ - c_{ij} -th response from the combination of the i -th block with the j -th treatment.
- ▶ Assumptions:
 - ▶ N random variables are mutually independent.
 - ▶ c_{ij} random variables are a random sample from a continuous distribution F_{ij} .
 - ▶ $F_{ij}(u) = F(u - \beta_i - \tau_j)$

The Two-Way Layout

- ▶ The two-way layout model is

$$X_{ijt} = \theta + \beta_i + \tau_j + e_{ijt}, i = 1, \dots, n, j = 1, \dots, k, t = 1, \dots, c_{ij}.$$

Randomized complete block design (HWC 7.1 -7.5)

- ▶ Single observation for every treatment–block combination.
- ▶ Could be the actual data or summarized data for each cell.
- ▶ $c_{ij} = 1, \forall i, j$.

Blocks	Treatments			
	1	2	\cdots	k
1	X_{11}	X_{12}	\cdots	X_{1k}
2	X_{21}	X_{22}	\cdots	X_{2k}
	\vdots	\vdots		\vdots
n	X_{n1}	X_{n2}	\cdots	X_{nk}

- ▶ $H_0 : [\tau_1 = \tau_2 = \cdots = \tau_k]$ versus
 $H_A : [\tau_1, \tau_2, \cdots, \tau_k \text{ not all equal}]$.

Randomized complete block design (Friedman test)

- ▶ Friedman (1937) statistic S

$$S = \frac{12n}{k(k+1)} \sum_{j=1}^k \left(R_j - \frac{k+1}{2} \right)^2,$$

where $R_j = \frac{R_j}{n}$ and $R_j = \sum_{i=1}^n r_{ij}$, sum of the ranks in treatment j .

Randomized complete block design (large-sample test, ties)

- ▶ Large-sample test
 - ▶ As $n \rightarrow \infty$ (number of blocks increases),

$$S \sim \chi_{k-1}^2.$$

- ▶ Ties
 - ▶ When there are ties, an exact test is only approximate.
 - ▶ Modification needed for the test statistic in large-sample approximation.

Randomized complete block design (Example)

- ▶ Example 7.1 (HWC, page 293)
- ▶ Data from a study to determine which, if any, of three methods of rounding first base is best (round out, narrow angle, wide angle).
- ▶ This will give information on the average, the time to reach second base (minimizes).
- ▶ Each subject is a block.
 - ▶ Number of blocks $n = 22$.
 - ▶ We can use large-sample approximation.

Randomized complete block design (Example)

```
library(magrittr)
library(NSM3)
library(dplyr)
library(tidyr)
data("rounding.times")
df = rounding.times %>% data.frame
colnames(df) = paste0("Method_",
  seq(1,3))
df = mutate(df, block = seq(1, dim(df)[1]))
df = gather(df, key = "Method", value = "xij", c(1:3))
friedman.test(df$xij, df$Method, df$block)
```

```
##
```

```
## Friedman rank sum test
```

```
##
```

```
## data: df$xij, df$Method and df$block
```

```
## Friedman chi-squared = 11.143, df = 2, p-value = 0.00380
```


Randomized complete block design (Example)

- ▶ We reject H_0 based on the Friedman test.
- ▶ Large-sample approximation
 - ▶ $S = 11.14$.
 - ▶ degrees of freedom $k - 1 = 2$.

```
round(1 - pchisq (11.14, 2), digits = 3)
```

```
## [1] 0.004
```

- ▶ Using the large- sample procedure adjusted for ties, we reject H_0 .
- ▶ We reject the hypothesis that the methods are equivalent with respect to time to reach second base.

RCBD (Ordered alternative)

- ▶ HWC Chapter 7.2 (Page test statistic).
- ▶ Appropriate when there is an increasing treatment effects.
 - ▶ treatments corresponding to quality or quantity of materials,
 - ▶ severity of disease,
 - ▶ drug dosage levels,
 - ▶ intensity of stimulus.

Multiple comparison (RCBD)

- ▶ Two-sided all treatments.
- ▶ Approximately distribution-free multiple comparison procedures based on signed ranks.

```
library(BSDA)
p.value12 = SIGN.test(x = c(rounding.times[,1]-
    rounding.times[,2]),
    alternative = "two.sided")$p.value
p.value13 = SIGN.test(x = c(rounding.times[,1]-
    rounding.times[,3]),
    alternative = "two.sided")$p.value
p.value23 = SIGN.test(x = c(rounding.times[,2]-
    rounding.times[,3]),
    alternative = "two.sided")$p.value
```

Multiple comparison (RCBD)

```
round(c(p.value12, p.value13, p.value23), digits = 3)
```

```
## [1] 0.238 0.017 0.017
```

```
round(p.adjust(c(p.value12,  
  p.value13, p.value23), method = "holm"), digits = 3)
```

```
## [1] 0.238 0.051 0.051
```

At an approximate experiment-wise error rate of .05, we have reached the conclusion that the round out (τ_1) and wide angle (τ_3) and narrow angle (τ_2) and wide angle (τ_3) running methods yield significantly different median times to second base.

Multiple comparison (treatments versus control)

- ▶ Daly and Cooper (1967) considered the rate of stuttering adaptation under three conditions.
 - ▶ no electroshock
 - ▶ electroshock was administered during each moment of stuttering (Shock following)
 - ▶ electroshock was administered immediately following each stuttered word (Shock during)
- ▶ determine if either of the two treatments yield improved (larger) median adaptation scores

```
data("adaptation.scores")
df = adaptation.scores %>% data.frame
colnames(df) = c("no.electroshock",
  "Shock.following", "Shock.during")
df = mutate(df, block = seq(1, dim(df)[1]))
df = gather(df, key = "Conditions",
  value = "xij", c(1:3))
```

Multiple comparison (treatments versus control)

```
friedman.test(df$xij, df$Conditions, df$block)
```

```
##
```

```
## Friedman rank sum test
```

```
##
```

```
## data: df$xij, df$Conditions and df$block
```

```
## Friedman chi-squared = 1.1077, df = 2, p-value = 0.5747
```

- ▶ We do not have enough evidence to conclude that at least one of the conditions is different from no electroshock condition.
- ▶ We do not need to do the multiple comparison.

Multiple comparison (treatments versus control)

- ▶ Let's do the multiple comparison for the sake of illustration for the multiple comparison (one-sided treatments versus control).
- ▶ Approximately distribution-free multiple comparison procedures based on signed ranks.

```
p21 = SIGN.test(x = c(adaptation.scores[,2]-  
  adaptation.scores[,1]),  
  alternative = "greater")$p.value  
p31 = SIGN.test(x = c(adaptation.scores[,3]-  
  adaptation.scores[,1]),  
  alternative = "greater")$p.value  
c(p21, p31)
```

```
## [1] 0.2272491 0.4018097
```

Multiple comparison (treatments versus control)

```
p.adjust(c(p21, p31), method = "holm")
```

```
## [1] 0.4544983 0.4544983
```

- ▶ At an experiment-wise error rate of .05, we find no statistical evidence that either of the two electroshock treatments lead to an increase in median adaptation scores over the control setting.

Randomized balanced incomplete block design (BIBD)

- ▶ General alternative: HWC Chapter 7.6.
- ▶ Multiple comparison (two-sided): HWC Chapter 7.7.

Randomized block design (HWC 7.9 -7.10)

- ▶ $c_{ij} = c > 1 \forall i, j$. thus, $N = nkc$.
- ▶ $H_0 : [\tau_1 = \tau_2 = \cdots = \tau_k]$ versus $H_A : [\tau_1, \tau_2, \cdots, \tau_k \text{ not all equal}]$.
- ▶ Friedman ranks
 - ▶ Order the k observations from least to greatest separately within each of the n blocks.
- ▶ Mack - Skillings test (MS).
- ▶ The large-sample test
 - ▶ As $nc \rightarrow \infty$, $MS \sim \chi_{k-1}^2$.

Randomized block design (Example)

- ▶ Example 7.9: Determination of Niacin in Bran Flakes.
- ▶ Investigate the precision and homogeneity of a procedure for assessing the amount of niacin in bran flakes.
- ▶ In four different laboratories (number of blocks is 4), bran flakes enriched with 0, 4, or 8 mg niacin per 100 g of cereal.

Randomized block design (Example)

```
library(Skillings.Mack)
data("niacin")
df = matrix(niacin, nrow=3,
  byrow=TRUE) %>% t %>% data.matrix()
colnames(df) = paste0("trt",
  c(0, 4, 8))
Ski.Mack.test = Ski.Mack(df) # df is a data matrix with columns

##
## Skillings-Mack Statistic = 22.679487 , p-value = 0.019
## Note: the p-value is based on the chi-squared distribution
##
## $Nblocks
## [1] 3
##
## $Ntreatments
## [1] 12
##
```

Randomized block design (Example)

- ▶ We can reject H_0 at the $\alpha = .05$ level, providing rather strong evidence that the studied process for assessing niacin content in bran flakes does not produce consistent results across a variety of laboratories and is therefore not reliable as an evaluative procedure.

Median Polish

- ▶ To find an additively-fit model for data in a two-way layout table - John Tukey reference here.
 - ▶ Overall median + row effect + column effect.
- ▶ Use rows and columns medians to compute the row and column effect.

Median polish procedure

- ▶ Step 1
 - ▶ Compute overall median.
 - ▶ Compute the residual table: observed - overall median.
 - ▶ Set row and column effect to zero.
- ▶ Step 2
 - ▶ Compute the row medians, including for column effect row.
- ▶ Step 3
 - ▶ Create residual table from row medians
 - ▶ Add row medians to row effect.
 - ▶ Subtract row median from last set of residuals.
- ▶ Step 4
 - ▶ Compute the column medians, including for the row effect.
- ▶ Step 5
 - ▶ Create a residual table from column medians.
 - ▶ Add column medians to column effect.
 - ▶ Subtract column median from last set of residuals.
- ▶ Repeat (1)-(5) until negligible change occur with row or column medians (effect).

Median polish procedure (Example)

- ▶ Infant mortality rates in the United States 1964-1966 by region and father's education.
- ▶ Cell entries are number of deaths (under one year old) per 1000 live births.

```
df = data.frame(ed8 = c(25.3, 32.1, 38.8, 25.4),  
  ed9to11 = c(25.3, 29, 31, 21),  
  ed12 = c(18.2, 18.8, 19.3, 20.3),  
  ed13to15 = c(18.3, 24.3, 15.7, 24),  
  ed16 = c(16.3, 19, 16.8, 17.5))  
rownames(df) = c("NE", "NC", "S", "W")  
df
```

	ed8	ed9to11	ed12	ed13to15	ed16
## NE	25.3	25.3	18.2	18.3	16.3
## NC	32.1	29.0	18.8	24.3	19.0
## S	38.8	31.0	19.3	15.7	16.8
## W	25.4	21.0	20.3	24.0	17.5

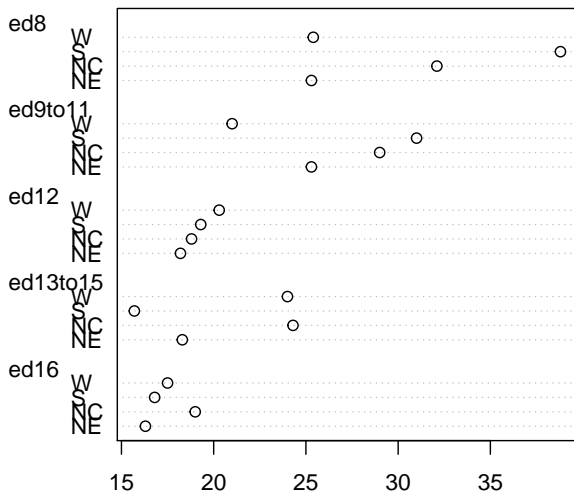
Cleveland Dot Plot

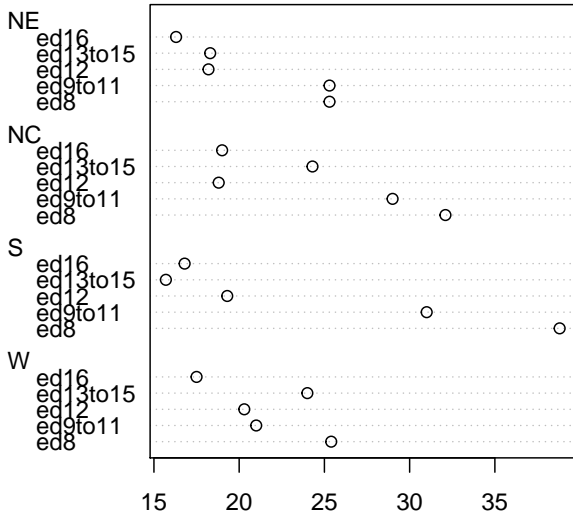
- ▶ A simple plot

```
df = df %>% as.matrix()  
dotchart(df)
```

- ▶ Within fathers educated up to 6th grade category, the largest infant mortality is observed in the Southern USA.
- ▶ The group with fathers educated 16 years or more has the least infant mortality.

Cleveland Dot Plot

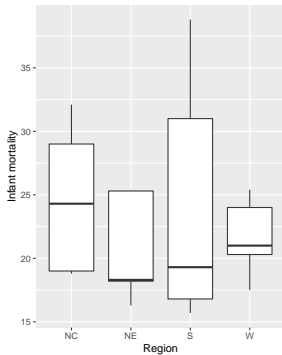
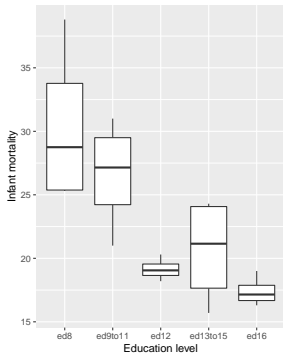




Boxplot

```
library(ggplot2)
df = data.frame(ed8 = c(25.3, 32.1, 38.8, 25.4),
  ed9to11 = c(25.3, 29, 31, 21),
  ed12 = c(18.2,18.8,19.3,20.3),
  ed13to15 = c(18.3,24.3,15.7,24),
  ed16 = c(16.3, 19, 16.8, 17.5))
rownames(df) = c("NE", "NC", "S", "W")
df.edu = gather(df, key = "Education_level")
df.edu = mutate(df.edu,
  Education_level = factor(Education_level,
    levels = c("ed8", "ed9to11", "ed12", "ed13to15", "ed16"))
df.region = df %>% as.matrix() %>%
  t %>% data.frame %>% gather(key = "Region")
df.region = mutate(df.region,
  Region = factor(Region))
```

Boxplot



Median polish

- ▶ We can represent the infant mortality rate using 4 components: overall median, median between education level, median between region, difference between each mortality and the difference.
- ▶ We can compare infant mortality rate between education level, region and both education and region.
- ▶ Additive model

$$X_{ij} = \theta + \beta_i + \tau_j + e_{ij}, i = 1, \dots, n, j = 1, \dots, k.$$

- ▶ θ : overall median.
- ▶ β_i : row effect (region).
- ▶ τ_j - column effect (fathers' number of years of education).
- ▶ e_{ij} - residuals

df

##		ed8	ed9to11	ed12	ed13to15	ed16
##	NE	25.3	25.3	18.2	18.3	16.3
##	NC	32.1	29.0	18.8	24.3	19.0
##	S	38.8	31.0	19.3	15.7	16.8
##	W	25.4	21.0	20.3	24.0	17.5

- ▶ Step 1
 - ▶ Compute overall median.
 - ▶ Compute the residual table: observed - overall median.
 - ▶ Set row and column effect to zero.

```
overall.median = median(as.matrix(df)); overall.median
```

```
## [1] 20.65
```

```
df = df - overall.median
df = cbind(rowEff = c(rep(0,4)), df)
df = rbind(c(rep(0,4)), df); rownames(df)[1] = "colEff"
df[1,1] = overall.median; df
```

##	rowEff	ed8	ed9to11	ed12	ed13to15	ed16
## colEff	20.65	0.00	0.00	0.00	0.00	0.00
## NE	0.00	4.65	4.65	-2.45	-2.35	-4.35
## NC	0.00	11.45	8.35	-1.85	3.65	-1.65
## S	0.00	18.15	10.35	-1.35	-4.95	-3.85
## W	0.00	4.75	0.35	-0.35	3.35	-3.15

► Step 2

- Compute the row medians, including for column effect row.

```
rowMed = apply(df[c(1:5), c(2:6)], 1, median); rowMed
```

```
## colEff      NE      NC      S      W  
##    0.00   -2.35    3.65   -1.35    0.35
```

► Step 3

- Create residual table from row medians
 - Add row medians to row effect.
 - Subtract row median from last set of residuals.

```
df[, 1] = df[, 1] + rowMed
df[c(2:5), c(2:6)] = df[c(2:5), c(2:6)] - rowMed[-1]; df
```

##	rowEff	ed8	ed9to11	ed12	ed13to15	ed16
## colEff	20.65	0.0	0.0	0.0	0.0	0.0
## NE	-2.35	7.0	7.0	-0.1	0.0	-2.0
## NC	3.65	7.8	4.7	-5.5	0.0	-5.3
## S	-1.35	19.5	11.7	0.0	-3.6	-2.5
## W	0.35	4.4	0.0	-0.7	3.0	-3.5

► Step 4

- Compute the column medians, including for the row effect.

```
colMed = apply(df[c(2:5), c(1:6)], 2, median); colMed
```

##	rowEff	ed8	ed9to11	ed12	ed13to15	ed16
##	-0.50	7.40	5.85	-0.40	0.00	-3.00

► Step 5

- Create a residual table from column medians.
 - Add column medians to column effect.
 - Subtract column median from last set of residuals.

```
df[1, ] = df[1, ] + colMed  
df[c(2:5), ] = sweep(df[c(2:5), ], 2, colMed);df
```

##	rowEff	ed8	ed9to11	ed12	ed13to15	ed16
## colEff	20.15	7.4	5.85	-0.4	0.0	-3.0
## NE	-1.85	-0.4	1.15	0.3	0.0	1.0
## NC	4.15	0.4	-1.15	-5.1	0.0	-2.3
## S	-0.85	12.1	5.85	0.4	-3.6	0.5
## W	0.85	-3.0	-5.85	-0.3	3.0	-0.5

- ▶ That completes the first smoothing iteration.
 - ▶ We can continue the iteration until the row median and column median are zero or relatively small change to the previous iteration.

► Let us do the second iteration:

```
df = data.frame(ed8 = c(25.3, 32.1, 38.8, 25.4),  
  ed9to11 = c(25.3, 29, 31, 21),  
  ed12 = c(18.2,18.8,19.3,20.3),  
  ed13to15 = c(18.3,24.3,15.7,24),  
  ed16 = c(16.3, 19, 16.8, 17.5))  
rownames(df) = c("NE", "NC", "S", "W")  
  
med.polish2 = medpolish(df , maxiter = 2)
```

```
## 1: 46.9
```

```
## 2: 43.05
```

```
med.polish2$overall
```

```
## [1] 20.15
```

```
med.polish2$row
```

```
##      NE      NC      S      W  
## -1.55  3.00 -0.35  0.35
```

```
med.polish2$col
```

```
##      ed8  ed9to11  ed12 ed13to15  ed16  
##  7.825   6.275  -0.450   0.425  -3.000
```

```
med.polish2$residuals
```

```
##      ed8 ed9to11  ed12 ed13to15  ed16  
## NE -1.125   0.425  0.05  -0.725  0.70  
## NC  1.125  -0.425 -3.90   0.725 -1.15  
## S  11.175   4.925 -0.05  -4.525  0.00  
## W  -2.925  -5.775  0.25   3.075  0.00
```

- ▶ row and column effects are not really close to zero.
- ▶ Let's do the third iteration.

```
med.polish3 = medpolish(df , maxiter = 3)
```

```
## 1: 46.9
```

```
## 2: 43.05
```

```
## 3: 42.575
```



```
med.polish3$overall
```

```
## [1] 20.575
```

```
med.polish3$row
```

```
##      NE      NC      S      W  
## -1.500  2.575 -0.350  0.350
```

```
med.polish3$col
```

```
##      ed8  ed9to11  ed12 ed13to15  ed16  
##  7.5875  6.0375 -0.9000  0.1875 -3.4250
```

```
med.polish3$residuals
```

```
##      ed8 ed9to11  ed12 ed13to15  ed16  
## NE -1.3625  0.1875  0.025 -0.9625  0.650  
## NC  1.3625 -0.1875 -3.450  0.9625 -0.725  
## S  10.9875  4.7375 -0.025 -4.7125  0.000  
## W  -3.1125 -5.9625  0.275  2.8875  0.000
```

- medpolish can determine the number of iteration until convergence. For our example, convergence is reached at the fourth iteration.

```
library(reshape2)
med.polish = medpolish(df)
```

```
## 1: 46.9
## 2: 43.05
## 3: 42.575
## Final: 42.3625
```

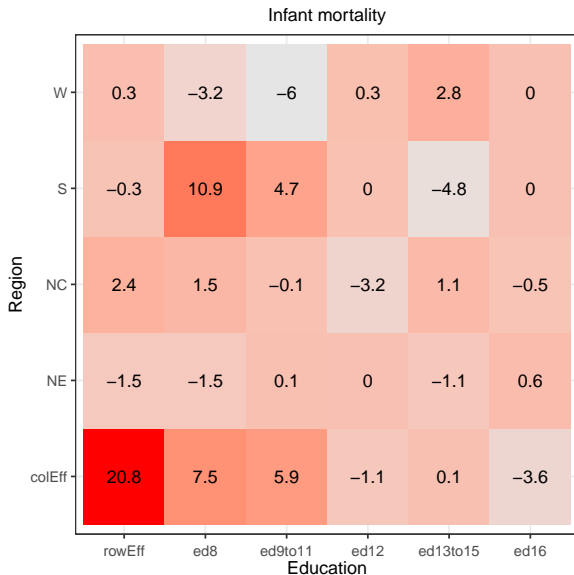
```
med.polish.res = med.polish$residuals
# add row effect and column effect to the residual table
med.polish.res = cbind(rowEff = med.polish$row,
  med.polish.res)
med.polish.res = rbind(c(med.polish$overall,
  med.polish$col),
  med.polish.res)
rownames(med.polish.res)[1] = "colEff"
colnames(med.polish.res)[1] = "rowEff"
```

```
med.polish.res
```

```
##           rowEff           ed8  ed9to11      ed12 ed13to15      ed1
## colEff 20.7625   7.48125   5.93125  -1.1000   0.08125  -3.612
## NE      -1.4750  -1.46875   0.08125   0.0125  -1.06875   0.625
## NC       2.3875   1.46875  -0.08125  -3.2500   1.06875  -0.537
## S       -0.3500  10.90625   4.65625  -0.0125  -4.79375   0.000
## W        0.3500  -3.19375  -6.04375   0.2875   2.80625   0.000
```

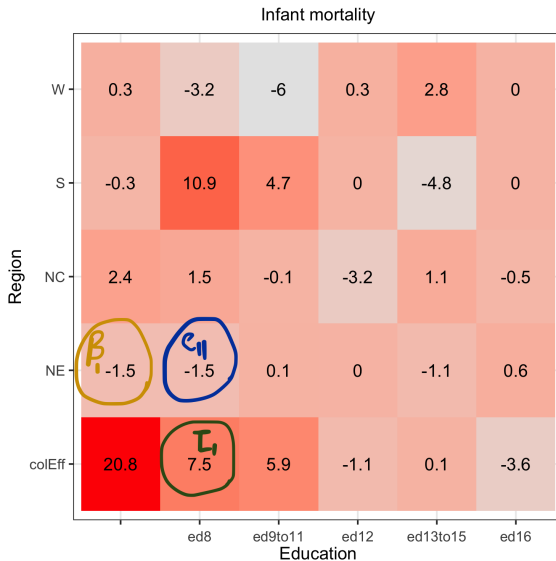
```
med.polish.res.long = melt(med.polish.res)
```

```
p = ggplot(med.polish.res.long,  
  aes(x = Var2, y = Var1)) +  
  geom_tile(aes(fill = value, label = value)) +  
  geom_text(aes(label = round(value, 1))) +  
  scale_fill_gradient(low="grey90", high="red") +  
  labs(x="Education", y="Region",  
    title="Infant mortality") + theme_bw() +  
  theme(axis.text.x=element_text(size=9,  
    angle=0, vjust=0.3),  
    axis.text.y=element_text(size=9),  
    plot.title=element_text(size=11,  
      hjust = 0.5), legend.position="none")
```



- ▶ Overall median contributing to the largest effect.
- ▶ Abnormal large residual in Southern region with 8th grade education of father.

Interpretation of median polish results



Interpretation of median polish results

- ▶ Additive model

$$X_{ij} = \theta + \beta_i + \tau_j + e_{ij}, i = 1, \dots, n, j = 1, \dots, k.$$

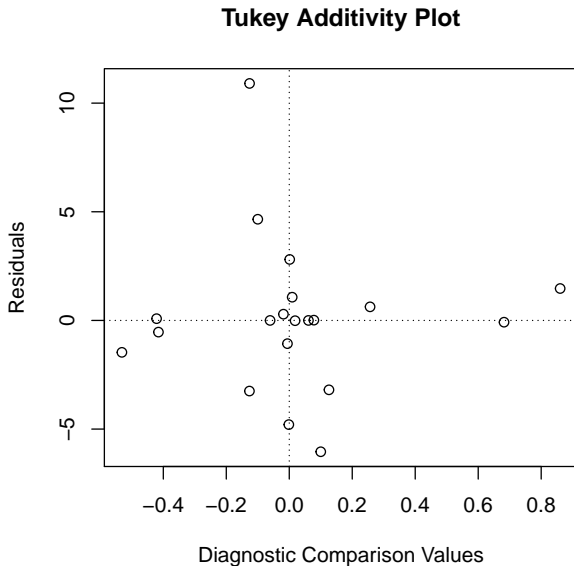
- ▶ For example, $X_{11} = 25.3$, $\theta = 20.8$, $\beta_1 = -1.5$, $\tau_1 = 7.5$, $e_{11} = -1.5$.
 - ▶ That gives, $25.3 \approx 20.8 - 1.5 + 7.5 - 1.5$.
- ▶ The largest infant mortality rate is observed for fathers who did not attain more than 8 years of school.
- ▶ The least infant mortality rate is observed for fathers who has completed more than 16 years of school.
- ▶ Note for median polish:
 - ▶ We can use mean instead of median.
 - ▶ If we use mean, then there is no iterations.
 - ▶ The row and column effects are regression coefficients.

Tukey additivity plot

- ▶ Check whether additive model is appropriate.
- ▶ Plot residuals versus the comparison value, cv_{ij} , defined as $\frac{\beta_i \tau_j}{\theta}$.
 - ▶ If there is no pattern in the plot, then the additive model is appropriate.

Tukey additivity plot

```
plot(med.polish)
```

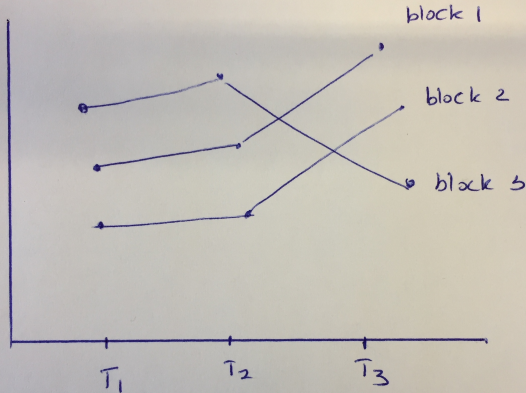


Tukey additivity plot

- ▶ If there is a trend observed, it's slope could be used to define the transformation of the data: $X_{ij}^{1-\text{slope}}$.
- ▶ If the transformation is not appropriate, then add cv_{ij} to the model.

Profile plots

- Identifying interaction between treatment and blocking factors.



Block & Treatment interaction.

References for this lecture

HWC Chapter 7

CS 2016 Lecture notes on introduction to nonparametric.

Manny Gimond 2019 Lecture notes on Median polish and Tukey additivity plot.