XI. Exact Methods

Lecture 11

4. Remove effect of unknown parameters:

7. 

1. Model: 

4. Model probability for item $j$ beating $k$ as 

$$\exp(\beta_j - \beta_k)/(1 + \exp(\beta_j - \beta_k))$$

5. Adding a constant to each of the $\beta$ keeps probabilities teh same 

a. Hence one of the items must be taken as baseline 

b. Note that $P[j$ beats $k] = 1 - P[k$ beats $j]$ as it should. 

7. $\beta_k$ represents strength of item $k$ 

8. Can add intercept for “home team advantage” 

9. Model fit is same as for quasi-symmetry model 

A: 5.4.2–5.4.3

XI. Exact Methods

A. Contingency tables: 

1. Model: 

a. $X_{ij} \sim \mathcal{P}(\lambda_{ij})$ 

b. $\log(\lambda_{ij}) = \alpha_i + \beta_j + \gamma_{ij}$ 

c. $\beta_j = 0$, $\gamma_{ij} = 0 \forall i, j$ 

2. $H_0 : \gamma_{ij} = 0 \forall i, j$ vs. $H_A : \gamma_{ij} \neq 0$ for some $i, j$ 

3. Test statistics: 

a. Score statistic is Pearson $\chi^2$: 

$$T = \sum_{i,j} (X_{ij} - X_i X_j / X.)^2 / (X_i X_j / X.)$$ 

b. LR statistic 

c. Fisher’s statistic $1/P[X]$ 

4. Remove effect of unknown parameters: 

a. Remove $\alpha_i$ by conditioning on $X_i$ 

- That is, note that for any common $\pi^0$ value, 

$$P_{\pi^0} \left[ \sup_{\pi \in [0,1]} P_{\pi} [Z \geq z] \right] \leq \sup_{\pi \in [0,1]} P_{\pi} [Z \geq z]$$ 

- Convexity condition: If test rejects for $(x_1, x_2)$, then test rejects for more extreme $(x_1, x_2 + 1)$ and $(x_1 + 1, x_2)$ 

- Extend to $p$-values: $p(x_1, x_2) \geq p(x_1, x_2 + 1)$ and $p(x_1, x_2) \geq p(x_1 + 1, x_2)$ 

- Implies that one need only look along boundary of alternative hypothesis for maximizer. 

- That is, $p$-value is the same whether we test $H_0 : \delta = \delta_0$ vs $H_A : \delta > \delta_0$ or $H_0 : \delta < \delta_0$ vs $H_A : \delta > \delta_0$ 

- Heuristically, because rejection region probabilities become less 

- Need to check for convexity: $Z$ statistics, Fisher’s exact test all work. 

- Can also phrase question in terms of relative risk $\pi_2/\pi_1$ 

6. Computation 

a. Either enumerate all tables, and calculate probabilities straight–forwardly, or 

b. (Pagano and Halvorsen, 1981) calculate recursively 

1. $P[X_{11} = x_{11} | X_1, X_j \forall i, j]$ 

$$= \frac{x_{11}!(x_{1} - x_{11})!(x_{.} - x_{11})!}{x_1!(x_{11} - x_{11})!(x_{1} - x_{11})!(x_{.} - x_{1})!}$$ 

- Probabilities do not depend on other aspects of conditioning event.
ii. \( P[V = v|U = u] = \frac{c(u, bv) \exp(v\tau)}{\sum_v c(u, bv) \exp(v\tau)} \)

So we need algorithm to generate list of \( v \) consistent with \( u \), and to calculate \( c(u, v) \) for these \( u \).

h. Let
   i. \( \Omega_i \) be sample space using observations 1, \ldots, \( i \), satisfying conditioning statement.
   ii. \( c_i \) be counts of \( X_1, \ldots, X_i \) ensembles giving \( t \)
   i. Note that
      i. \( \Omega_1 = \{0, z_1\} \)
      ii. \( c_1(0) = c_1(z_1) = 1 \).
      iii. \( \Omega_i = \Omega_{i-1} \cup (z_i + \Omega_{i-1}) \)
         • After removing duplicates
   iv. \( c_i(t) = c_{i-1}(t) + c_{i-1}(t - z_i) \)

j. Collects all possible \( t \)
   i. Excessive: we only need vectors consistent with conditioning event.
   ii. Algorithm more efficient if we can eliminate from \( \Omega_i \) many entries that can never satisfy conditioning event.
   iii. Easiest condition to implement: dump those if component gets too large or small.

This page intentionally left blank.

This page intentionally left blank.