Inference, aggregation and graphics for top-*k* rank lists

Michael G. Schimek[1] Eva Budinská[2] Shili Lin[3]
Alena Myšičková[4]

[1]Medical University of Graz and Danube University Krems, Austria

[2]Swiss Institute of Bioinformatics, Lausanne, Switzerland

[3]Ohio State University, Columbus, USA

[4]Humboldt University, Berlin, Germany

useR! 2009, Rennes, France, July 8-10, 2009

Motivation

- In various fields of application we are confronted with lists of distinct objects in rank order
- The ordering might be due to a measure of strength of evidence or to an assessment based on expert knowledge or a technical device
- The ranking might also represent some measurement taken on the objects which might not be comparable across the lists, for instance, because of different assessment technologies or levels of measurement error

Our aim is

- to **consolidate such lists of common objects**
- to **provide computationally tractable solutions**, hence appropriate algorithms and graphs
- to develop an `R` **package** named **`TopkLists`**

General assumptions

- Let us assume ℓ assessors or laboratories ($j = 1, 2, \ldots, \ell$) assigning rank positions to the same set of N distinct objects
- Assessment of N distinct objects according to the extent to which a particular attribute is present
- All assessors, independently of each other, rank the same objects between 1 and *N* on the basis of relative performance
- The ranking is from 1 to *N*, without ties
- Missing assessments are allowed
- The ℓ assessors produce ℓ ranked lists τ_j
- There are $(\ell^2 - \ell)/2$ possible pairs of such lists τ_j

The problem

Our overall goal is to identify a subset of objects that is characterized by high conformity across the lists

- It is implied that there is similarity between the rankings which can be evaluated by a distance measure d (a permutation metric)
- Such measures are
  - **Kendall's τ**
  - **Spearman's footrule**
- In practice we have **truncated lists and incomplete rankings** of objects in some or all of the lists caused by missing assignments
- Because of that **penalized distance measures are required**

The problem continued

- In most applications, especially for large or huge numbers N of objects, it is unlikely that consensus prevails
- As result only the top-ranked objects matter (the remainder ones show random ordering)
- Quite often we observe a general decrease, not necessarily monotone, of the probability for consensus rankings with increasing distance from the top rank position

Typically there is reasonable **conformity in the rankings for the first, say k, elements of the lists**

This motivates the **notion of *top-k rank lists*** as known from information retrieval literature

Important application field: **Integration and meta analysis of gene expression data (microarray experiments)**

Computational aspects and algorithms

List aggregation by means of brute force is limited to the situation where

- N is very small
- ℓ is very small
- the k's are equal and a priori known

Our purpose is to solve this computational problem for a realistic setting

There are **3 subtasks respectively algorithms**:

1. Selection of the $\hat{k}$'s for all possible pairs of lists τ_j
2. Integration of partial information from the pairs of lists via a graphical tool
3. Calculation of a set of objects characterized by rankings of high conformity across the lists up to some global index $\bar{k}$

Selection of the $\hat{k}$'s

Moderate deviation-based inference for random degeneration in paired rank lists (Hall and Schimek, 2009)

- For the estimation of the point of degeneration j_0 into noise independent **Bernoulli random variables** are assumed
- A general **decrease of the probability** p_j (need not be monotone) for concordance of rankings with increasing distance j from the top rank is assumed
- Several **tuning parameters** (δ, ν, ...) are required to account for the **closeness of the assessors' rankings and the degree of randomness in the assignments**
- The **algorithm** represents a simplified mathematical model;
- It is embedded in an **iterative scheme** to account for irregular rankings

Graphical integration of paired ranked lists

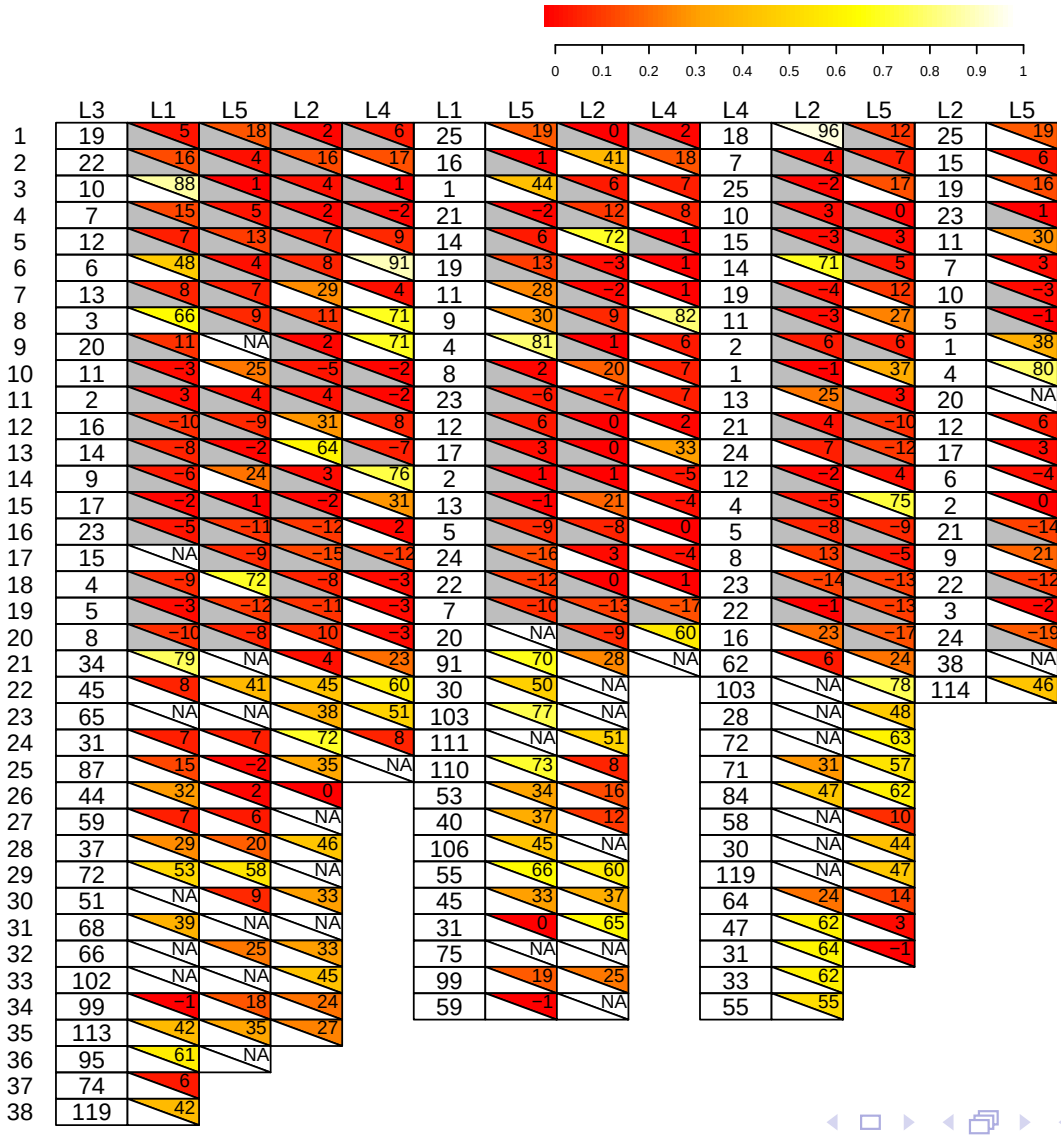
- Define a **partial reference list** L_1^0; anyone of the 2 lists with $\max_j(\hat{k}_j)$ objects among all pairwise comparisons
- L_1^0 gives the ordering of the objects O_i in the heatmap and defines the vertical axis
- Take L_1^0's highest ranking $\{\max_j(\hat{k}_j) + \delta\}$ objects O_i
- The **partial lists** $L_2, L_3, \ldots, L_\ell$ are ordered from highest to lowest by their individual k_j when compared to the reference list L_1^0 (one column per list)
- In each cell we represent: (1) **top-*k* membership**, '*yes*' is denoted by color 'grey' and '*no*' by 'white',
(2) **distance** of a current object $O_i \in L_1^0$ from its position in the other list, color scale from 'red' *identical* to 'yellow' *far distant* (integer value denotes distance with negative sign if to the left, and positive sign if to the right)

Calculation of a set of highly conforming objects

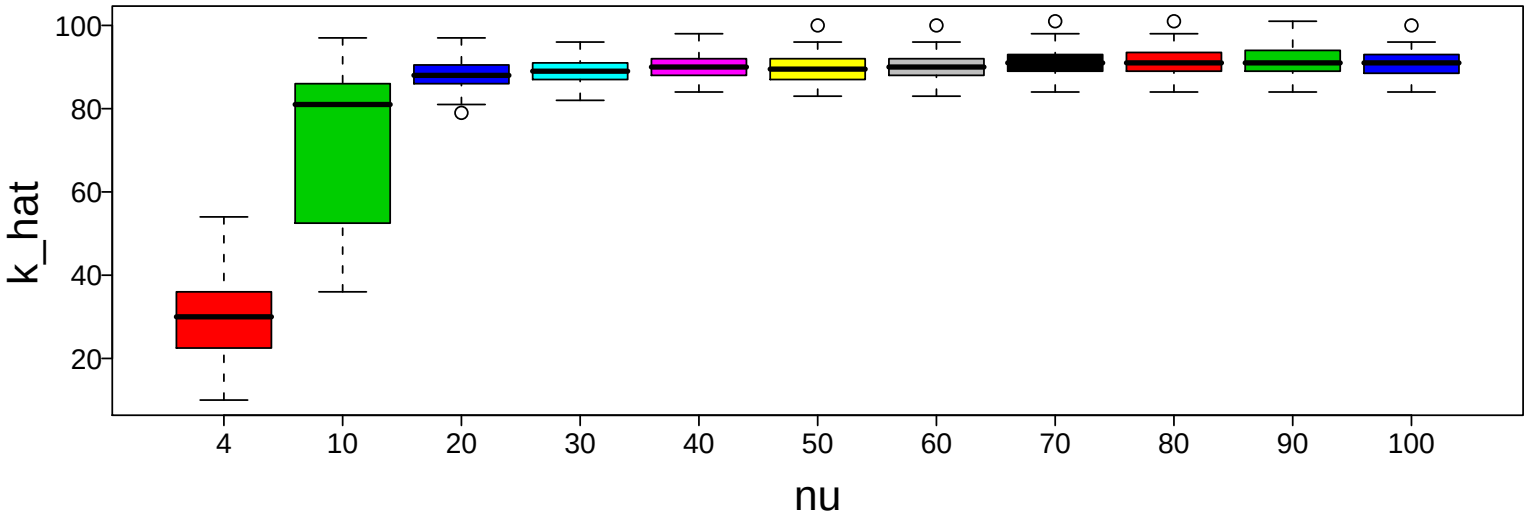
Cross-entropy Monte Carlo (CEMC) for consolidation of top-k objects (Lin and Ding, 2009)

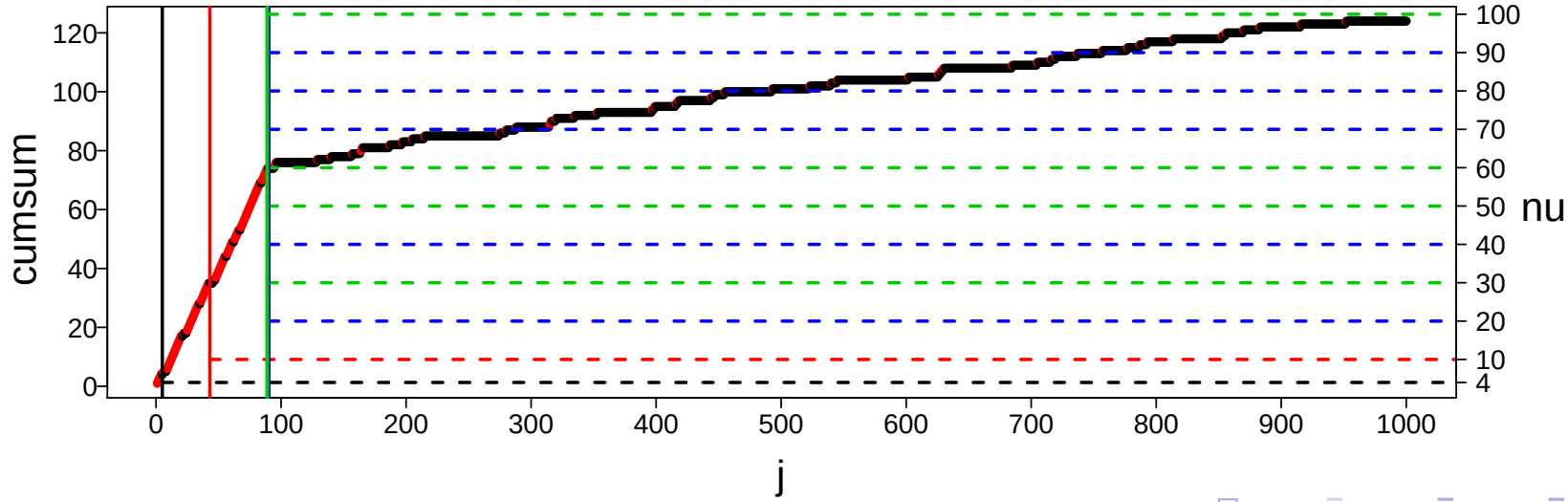
- Assume a **random matrix X** and a corresponding **probability matrix p**
- Given the **probability mass function** $P_{\mathbf{v}}(x)$, any realization x of **X** uniquely determines the corresponding top-k candidate list without reference to the probability matrix **p**
- **Stochastic search** to find an ordering x^* that corresponds to an optimal τ^* satisfying the minimization criterion
- **Iterative CEMC algorithm in two steps**: (i) simulation step in which random samples from $P_{\mathbf{v}}(x)$ are drawn, (ii) update step for improved samples increasingly concentrating around an x^* (correspond to optimal τ^*)

Graphics tool example: top-k integration of 5 gene expression lists ($N = 120, \hat{k}_j \in [20, 38]$)

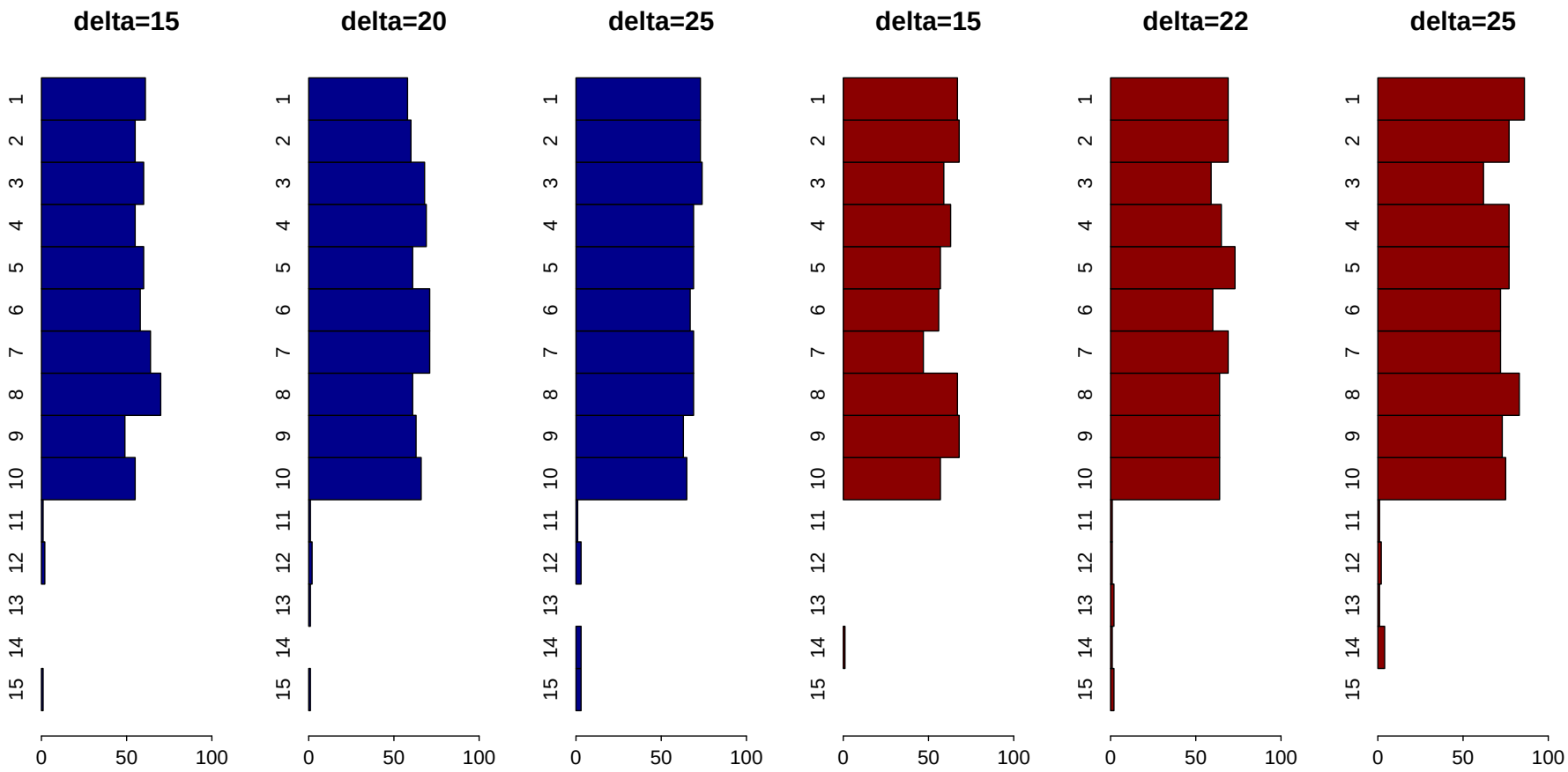


Simulation with 2 lists and k=100, δ=40, $N = 1000$: Estimation of $\widehat{k}$ for different ν





Simulation with 5 lists and k=10, Spearman's footrule (blue) vs. Kendall's τ (red), $N = 100$: Top selected genes (objects)



Summary

The TopkLists Package

- is implemented in `R`, applying the `grid` package
- is a computationally tractable and efficient approach to the top-k rank list problem
- **implements the Hall & Schimek-algorithm**
- **implements the Lin & Ding-algorithm**
- **implements graphical procedures for information integration**
- allows the user to interact with the data and to **select an overall top-k set of objects**
- allows to **monitor the aggregation process**
- allows to **evaluate tuning parameter choice**