SDSS 2019 – Bellevue, Washington

CS32 – Statistical Methods for Analyzing Large Scale or Massive Data

An Application of Linear Programming to Computational Statistics

John M. Ennis, PhD

President
Aigora

William Russ

Computational Market Researcher
The Institute for Perception

AIGORA

The Institute for Perception

Meet Marlene

- Marlene is a data scientist who works at a major consumer packaged goods company
  - She is skilled in several statistical packages but uses R for most of her day-to-day work
  - Marlene is respected by her peers as someone who learns quickly and can be trusted to come through with solutions to difficult problems
- As we join Marlene today, we find her working on tables to communicate statistical differences between a large number of samples on a large number of attributes

Marlene’s Problem

- Marlene is working on a dataset of 31 frozen pizzas evaluated on 25 attributes
  - She seeks to produce tables reflecting multiple statistical comparisons between all pizzas and on all attributes
  - Her solution will be the basis for a tool to be used throughout her company
- Ideally, her solution will be:
  - Flexible enough to allow her to adjust the statistical testing without having to rewrite large portions of her code
  - Fast enough to handle a large number of samples and attributes in reasonable time

Marlene's First Attempt

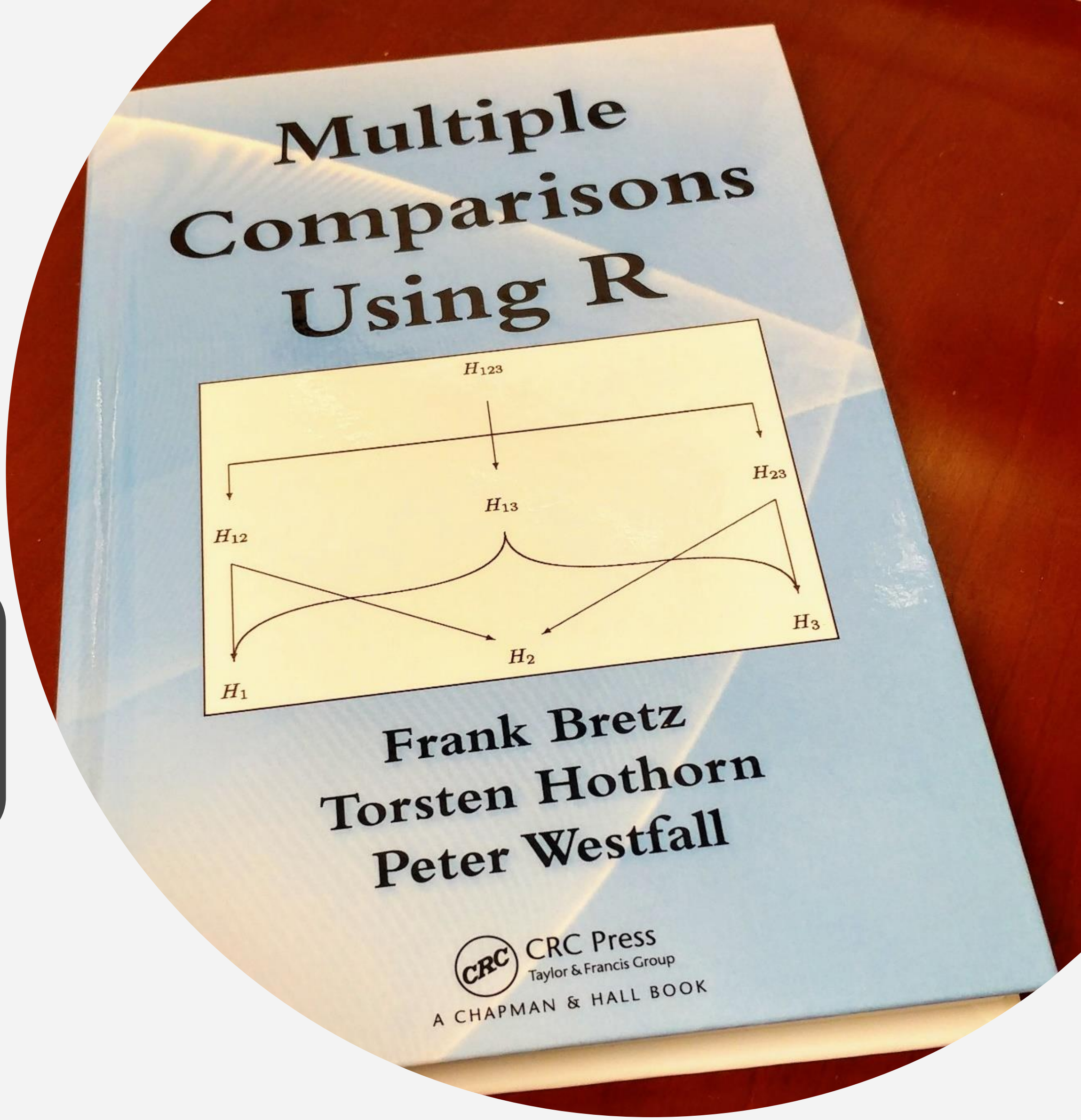
- Marlene begins by following the approach recommended by Bretz et al. (2016)
  - This excellent book recommends the use of compact letter displays to communicate the result of post-hoc Tukey tests:

```
ex_aov <- aov(scale_1 ~ prod_code, data = ex_data)

ex_mc <- ex_aov %>%
   multcomp::glht(linfct = mcp(code = "Tukey"))

cld_results <- multcomp::cld(ex_mc)
```

- Marlene runs into difficulty almost immediately
  - Her analysis of the first attribute requires more than 15 minutes to complete!
- Marlene decides she needs to review the literature on compact letter displays



Letter Displays

Letter Displays

- Letter displays enable representation of multiple statistical comparisons in a single table
  - For each attribute, samples with at least one letter in common are not significantly different

Sample	**Attribute 1**	**Attribute 2**	**Attribute 3**	…
1	3.73 a	2.96 ***e***f	3.01 b***c***	…
2	3.57 b	3.57 b	2.73 efghi	…
3	3.46 ***c***	3.16 d	2.79 efg	…
4	3.33 ***c***d	3.77 a	2.96 ***c***d	…
5	3.30 d	2.92 ***e***fg	2.72 fghi	…
6	3.29 d	3.28 cd	2.49 mn	…
7	3.26 d	3.47 b	3.10 b	…
8	3.01 e	3.26 cd	3.64 a	…
9	2.99 e	2.99 ***e***	3.09 b***c***	…
10	2.98 e	3.31 c	3.08 b***c***	…
⋮	⋮	⋮	⋮	

Example of Letter Display Reduction

Sample	Attribute	Sample	Attribute
1	6.25 a	12	5.53 bcdefgh
2	5.95 ab	13	5.52 bcdefghi
3	5.92 abc	14	5.50 bcdefghi
4	5.79 abc	15	5.42 bcdefghi
5	5.75 abcd	16	5.41 cdefghi
6	5.71 bcde	17	5.24 defghi
7	5.68 bcde	18	5.20 efghi
8	5.65 bcdef	19	5.13 fghi
9	5.62 bcdef	20	5.06 ghi
10	5.59 bcdefg	21	5.00 ij
11	5.54 bcdefgh	22	5.00 hij

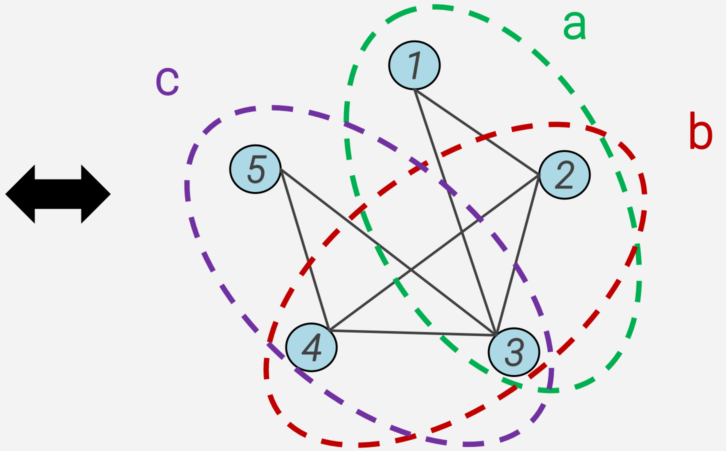
Sample	Attribute	Sample	Attribute
1	6.25 a	12	5.53 bfh
2	5.95 ab	13	5.52 bi
3	5.92 abc	14	5.50 bi
4	5.79 abc	15	5.42 bi
5	5.75 abcd	16	5.41 cefi
6	5.71 be	17	5.24 defi
7	5.68 be	18	5.20 efi
8	5.65 bf	19	5.13 fi
9	5.62 bf	20	5.06 ghi
10	5.59 bfg	21	5.00 ij
11	5.54 bfh	22	5.00 hij

48 letter assignments of 103 total can be removed, leaving just 55

Letter Displays and Graph Theory

- Each matrix of non-significant differences corresponds to a bidirectional graph
  - Samples correspond to vertices, which are connected when the samples are not significantly different

Sample	Attribute
1	3.73 a
2	3.57 ab
3	3.46 abc
4	3.33 bc
5	3.30 c

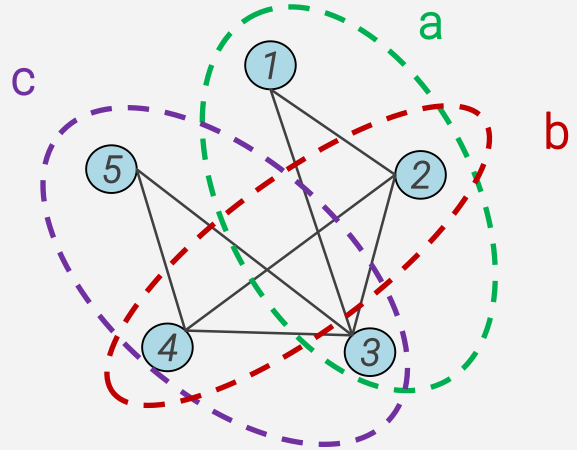


- Each letter of the display corresponds to a clique in the corresponding graph
  - An accurate letter display corresponds to a clique covering of the graph's edges
  - Removing a letter assignment from the display corresponds to removing a vertex from a clique
- As long as the graph's edges remain covered, the corresponding letter display is still accurate

Letter Displays and Graph Theory

- Each matrix of non-significant differences corresponds to a bidirectional graph
  - Samples correspond to vertices, which are connected when the samples are not significantly different

Sample	Attribute
1	3.73 **a**
2	3.57 **ab**
3	3.46 **ac**
4	3.33 **bc**
5	3.30 **c**



- Each letter of the display corresponds to a clique in the corresponding graph
  - An accurate letter display corresponds to a clique covering of the graph's edges
  - Removing a letter assignment from the display corresponds to removing a vertex from a clique
- As long as the graph's edges remain covered, the corresponding letter display is still accurate

AIGORA The Institute for Perception

Searching for Minimal Displays

- To find minimal displays in reasonable time for moderate size problems (Ennis et al., 2012; fewer than 50 samples):
  - Start with maximal display
  - Identify all individually removable letter assignments
  - Search for all simultaneously removable assignments
  - Select largest combination of simultaneously removable assignments
- To determine whether assignments are simultaneously removable, Ennis et al. (2012) proposed a characterization of accurate displays:
  - Let a_{ik} indicate with 0 or 1 whether item *i* is assigned letter *k*
  - A matrix of non-significant differences $M = [m_{ij}]$ is accurately described if $m_{ij} = 1$ exactly when $\sum_k a_{ik} a_{jk} \geq 1$
- This condition can be checked quickly as the search proceeds

We next improve speed using linear programming

Linear Programming

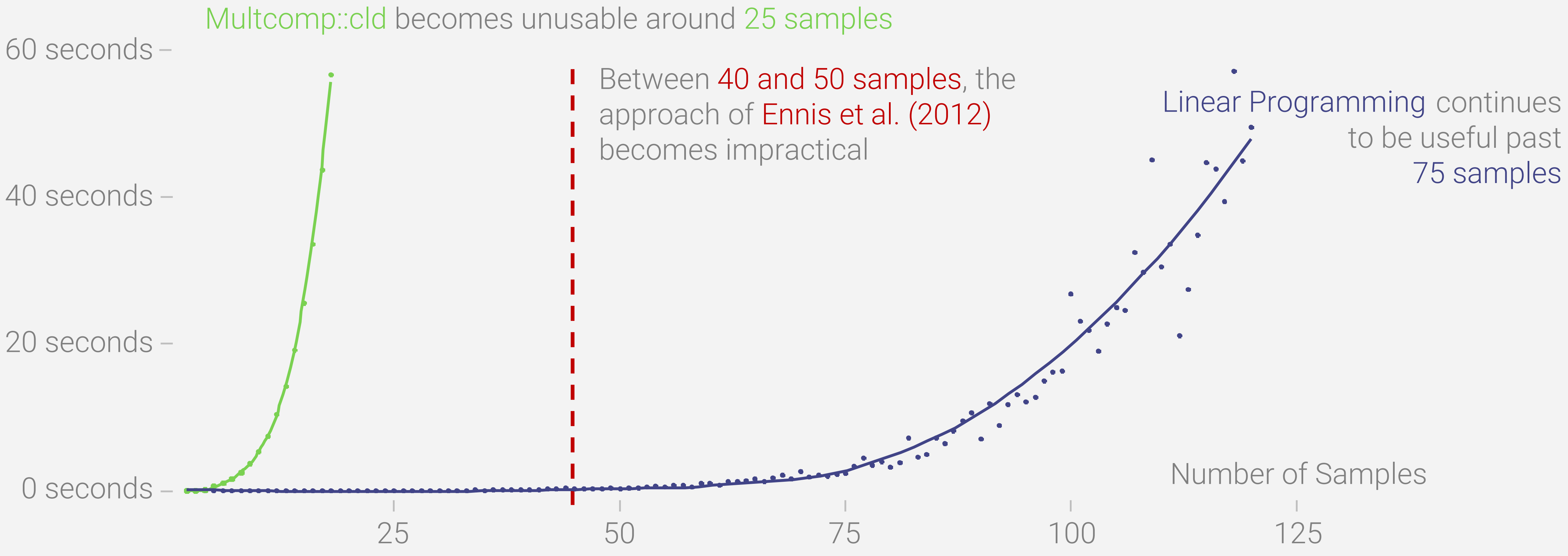
Minimizing Assignments with Linear Programming

1. Start with matrix $M = [m_{ij}]$ of non-significant differences
2. Create an binary indicator variable for every possible letter assignment a_{ik}
   - Goal: minimize the total number of letter assignments $T = \sum_{i,k} a_{ik}$
3. Find the maximal display by computing the complete set of maximal cliques
   - For example, using igraph::maximal.cliques (Csardi & Nepusz, 2006)
4. Define constraints:
   a) Set a_{ik} = 0 if assignment a_{ik} doesn't appear in the maximal display
   b) Set a_{ik}= 1 if assignments a_{ik} can't be removed individually
   c) For every *i* and *j* for which $m_{ij} = 1$, require $\sum_k a_{ik} a_{jk} \geq 1$
5. To implement step 4b, a trick is required because $\sum_k a_{ik} a_{jk}$ is quadratic
   a. For every i and j for which $m_{ij} = 1$, create an intermediate variable $e_{ijk} = a_{ik} a_{jk}$
   b. Require $e_{ijk} \leq a_{ik}, e_{ijk} \leq a_{jk}$, and $e_{ijk} \geq a_{ik} + a_{jk} - 1$ for every *i, j, k*

This problem can be solved using lpsolve::lp (Berkelaar, 2007)

Timing Comparison

- Data simulated according to an approach specified in Gramm et al. (2007)
  - 100 replications for each fixed number of samples
  - 90% quantiles for time to complete shown for each case



Back to Marlene

Back to Our Story

- Marlene writes scripts to implement a linear programming solution to her multiple statistical comparisons problem
- Her analysis of all 25 attributes runs in under 15 seconds
  - Recall: Her original analysis of just one attribute required more than 15 minutes
  - The median time for each attribute is less than half a second
  - The median number of letter assignments removed per attribute is 17
- Marlene's fame within her company grows
  - Her scientific colleagues look forward to using her scripts
  - Her colleagues in packaging and marketing insights invite her to discuss other problems that can be solved using linear programming

But, most important, **Marlene had fun**!

AIGORA

The Institute for Perception

Thank you for attending!

References

References (1/2)

- Berkelaar, M. (2007). The lpsolve package. *URL: http://www. lpsolve. sourceforge. net*.
- Bretz, F., Westfall, P., & Hothorn, T. (2016). *Multiple comparisons using R*. Chapman and Hall/CRC.
- Csardi, G., & Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal, Complex Systems*, *1695*(5), 1-9.
- Ennis, J. M., Fayle, C. M., & Ennis, D. M. (2012). Assignment-minimum clique coverings. *Journal of Experimental Algorithmics (JEA)*, *17*, 1-5.
- Gramm, J., Guo, J., Hüffner, F., & Niedermeier, R. (2006). Data reduction, exact, and heuristic algorithms for clique cover. In *Proceedings of the Meeting on Algorithm Engineering & Expermiments* (pp. 86-94). Society for Industrial and Applied Mathematics.
- Gramm, J., Guo, J., Hüffner, F., Niedermeier, R., Piepho, H. P., & Schmid, R. (2007). Algorithms for compact letter displays: Comparison and evaluation. *Computational Statistics & Data Analysis*, *52*(2), 725-736.

References (2/2)

- Gramm, J., Guo, J., Hüffner, F., & Niedermeier, R. (2009). Data reduction and exact algorithms for clique cover. *Journal of Experimental Algorithmics (JEA)*, *13*, 2.
- Hothorn, T., Bretz, F., Westfall, P., Heiberger, R. M., Schuetzenmeister, A., Scheibe, S., & Hothorn, M. T. (2017). Package 'multcomp'. *URL: http://cran.Statsfu.ca/web/packages/multcomp/multcomp*.
- Piepho, H.-P. (2000). Multiple treatment comparisons in linear models when the standard error of a difference is not constant. *Biometrical J. 42(7)*, 823–835.
- Piepho, H.-P. (2004). An algorithm for a letter-based representation of all-pairwise comparisons. *J. Comput. Graph. Stat. 13(2)*, 456–466.
- Sultan, A. (2014). *Linear programming: an introduction with applications*. Elsevier.

SDSS 2019 – Bellevue, Washington
CS32 – Statistical Methods for Analyzing Large Scale or Massive Data

An Application of Linear Programming to Computational Statistics

John M. Ennis, PhD
President
Aigora

William Russ
Computational Market Researcher
The Institute for Perception

AIGORA

The Institute for Perception