Computational Cognitive Science

Lecture 6: Aggregation and Individual Differences

Chris Lucas

School of Informatics

University of Edinburgh

October 3, 2019

Readings

- Chapter 5 of F&L

Recommended:

- "Modeling individual differences using Dirichlet processes" [link] by Navarro et al.

The Effect of Averaging

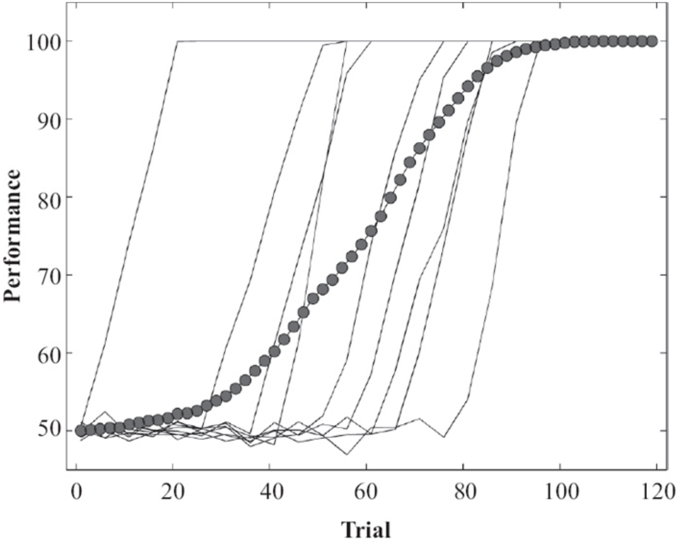
Admission data from UC Berkeley:

	Men		Women	
Department	Applied	Admitted	Applied	Admitted
A	825	511 (62%)	108	89 (82%)
B	560	353 (63%)	25	17 (68%)
C	325	120 (37%)	593	225 (38%)
D	191	53 (28%)	393	114 (29%)
Total	1901	1037 (55%)	1119	445 (40%)

Data aggregation (e.g., averaging) can substantially alter the interpretation of the data (and of modeling results).

The Effect of Averaging

Averaging of learning curves generated by a model:



Modeling and Data Aggregation

When building models, we need to decide whether to aggregate the data. We can assess and fit models with:

1. summary statistics (like averages)
2. data merged across individuals
3. individual-level data (independent)
4. groups or clusters of individuals
5. individual-level data (non-independent/hierarchical)

Why aggregate?

- Less work for you
- Less computationally expensive
- Usually implies simpler models
  - Less risk of overfitting with small data sets
  - Easier to communicate
- Sometimes the only realistic option
  - E.g., few points per participant

Why avoid aggregating?

- People aren't the same
  - Different strategies
  - Different expectations
  - Motivation, memory, ...
- Pretending they are the same can:
  - Mask interesting patterns
  - Lead to spurious conclusions

Groups: Splitting the difference

- Accommodate differences
- Not as data-intensive as separate individual analyses
- Evidence for clusters may be scientifically interesting

Fitting aggregate data: Reaction times

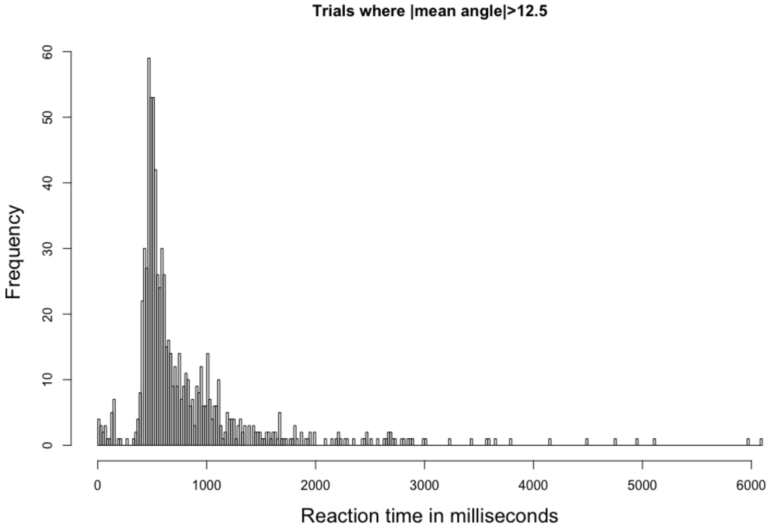
1. Fitting summary statistics of aggregate data
   - Typically easiest, with greatest downsides
   - Loses a great deal of information
2. Pretend everyone is the same
   - Common model, parameters, etc.
   - If assuming data are already conditionally independent, just lump them together

Example: Reaction times

Suppose we want to characterize people using a shifted Weibull distribution.

- Models an accumulator like the random walk
- Difference: “race” approach – first accumulator past the post
- Parameters:
  - Shift
  - Scale
  - Shape

Aggregate reaction times



Aggregate reaction times

- F&L (C5): Average quantiles by participant, minimize RMSE for these
  - Resembles the empirical plot
  - Produces a distribution that assigns zero probability to real judgments
- Alternately: MLE

Fitting individual participants

- Can directly maximize MLE for each person separately
- Unlikely to work well for sparse* data:
  - few observations per parameter
  - MLE not trustworthy (or unique)
- What if we could
  - Use a well-informed per-person prior?
  - Determine which people are similar; combine?

Fitting subgroups

- People aren't all the same
- People aren't all different
- Cluster people who are similar
  - W/raw data or descriptive features
  - Model-based clustering

Mixture models

Sometimes a distribution is a mixture of multiple latent distributions

- An experiment could recruit a mix of performance and speed-optimizing participants
- An individual's judgments or reaction times might be a mixture
- A sensor could have broken/non-broken modes

Mixture models

A probabilistic approach:

$$p(\mathbf{y}_i|\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k p(\mathbf{y}_i|\boldsymbol{\theta}_k)$$

- π_k is the weight of the k^{th} component
- $\boldsymbol{\theta}$ is now an ensemble of K different sets of parameters, one per group.

Expectation-maximization

Suppose we want to compute an MLE for:

1. The probability the each person belongs to each group $P(\mathbf{z})$
2. The parameters for each group $\boldsymbol{\theta}_k$?

- If we know who is in what group, we can get (2)
- If we know the parameters for each group, we can get (1)

We have neither.

Expectation-maximization

Full joint inference may be intractable.

What if we pretend we know the parameter MLEs, and get MLE group membership probabilities? (E)

What if we pretend we know $\mathbf{z}_{MLE}$, and MLE parameters? (M)

Better than nothing...

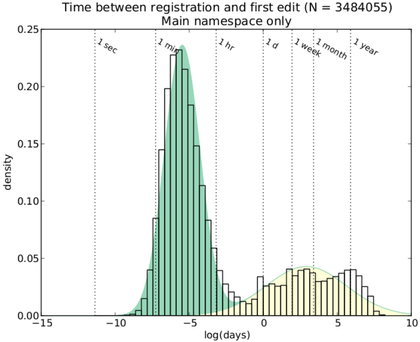
- What if we alternate between the two?

This provably converges to a locally optimal MLE for $\mathbf{z}$ and $\boldsymbol{\theta}$.

Mixtures of Gaussians

If we are using Gaussians, we have closed-form MLEs for both steps.

- Listing 5.3 in F&L Chapter 5.
- Very popular, even when data aren't Gaussian (e.g., proportions)
  - Not always correct, but often good enough



(Wikimedia commons, by Junkie.Dolphin)

Overfitting in MLE strikes again!

MLE with Gaussian mixture models suffers from a overfitting / degeneracy problem:

1. A cluster converges to a single point
2. MLE for standard deviation is zero
3. Error

As dimensionality increases, this problem becomes worse.

- MAP estimates under conjugate priors can get around this
- Can be an issue for other continuous mixtures as well

K-means as (kind of) a special case

- Hard assignments
- Equal and spherical covariance
- Not really a mixture model

Non-conjugate mixture models

- Mixture models are very generally useful
- However, standard EM doesn't work well in
  - high-dimensional cases
  - situations with non-conjugate priors
- There exist general methods for Bayesian inference in these settings, adoption is limited

Other uses for mixture models

Not just about individual differences under a model:

- can account for error
- multiple within-participant strategies
- less-arbitrary outlier detection

How many groups?

Standard approaches:

1. Model selection! E.g., BIC. More later
2. Nonparametric models

Nonparametric models

E.g., “stick-breaking models” like Dirichlet process mixture models.

Pros:

- Bayesian!
- No need to worry about group sizes
- Compatible with many probabilistic models

Cons:

- Inference can be expensive and/or tricky
- Harder to interpret distributions over clusters
  - Expected number of clusters can be misleading
  - Point estimates are easier to talk about

Hierarchical models

What if we could have it both ways?

- Group-level *and* individual parameters
- Robustness to over-fitting
- Inferences about individuals where supported by data
- Compatible with groups