

---

# Generating Synthetic Microdata from Marginal Tables and Confidentialised Files

---

Alan Lee

Department of Statistics

University of Auckland

---

# Today's agenda

1. The confidentiality/access tension
2. Approaches to resolving the tension
3. Synthetic data – review
4. Contingency tables and probability models
5. Methods and algorithms
6. Results using the 2001 census
7. Summary and conclusions

---

# Acknowledgement

- Thanks are due to Statistics New Zealand for financial support under the Official Statistics Research program

---

# The confidentiality dilemma

- The tension between researchers (who want microdata access) and statistical agencies (who want to preserve confidentiality) is well known
- The (old) NSO approach: Publish a few low dimensional tables
- The researchers demand: Open slather!

---

# The researcher's point of view

*Open access to official statistics provides the citizen with more than a picture of society. It offers a window on the work and performance of government itself, showing the scale of government activity in every area of public policy and allowing the impact of public policy to be assessed.*

*UK 1993 white paper on Open Government in the United Kingdom*

*The rush to ensure complete levels of privacy in the research context paradoxically results in less social benefit, rather than more.*

*... people will recognise that while they surely have a right to privacy, they may also come to the realisation that they have a duty to share information, if the common good is to be furthered.*

*Peter Madsen, NSF Workshop on Confidentiality Research, 2003.*

---

# The other side of the coin

- Individual data collected by statistical agencies for statistical compilation... are to be strictly confidential and used exclusively for statistical purposes

*Sixth UN principle of official statistics*

---

# The modern approach

- Manage the data risk, attempt to balance access with confidentiality
- It is appropriate for microdata collected for official purposes to be used to support research as long as confidentiality is protected
- Microdata should only be made available for statistical purposes
- Provision of microdata should be consistent with national legal arrangements

---

# Managing disclosure risk -options

- Open slather: Rely on sanctions, pass onus onto research community, appropriate retribution for confidentiality breaches, education, instill ethical behaviour
- Keep microdata in secure facility (data lab), control interrogation
- Perturb (confidentialise) data and release
- Release low –dimensional tables (data cubes)

---

# Confidentialised files

Light  
confidentialising  
Restricted access



Heavy  
confidentialising  
Liberal access

## Uses

- Software development for data lab
- Tutorial use
- Inference?

---

# An approach to confidentialising: Synthetic data

Basic idea:

- Use available data (from Table Builder, CURF, microdata) to fit a statistical model
- Model could be a log-linear model, mixture model, other....
- Having fitted a model, generate artificial data from the distribution defined by the model

---

# A toy example

Consider the following toy example from Alan Agresti's book, *Categorical Data Analysis*, on alcohol, cigarette and marijuana use among US high school students

3 variables:

*A: Alcohol ever used (Y/N)*

*C: Cigarettes ever used (Y/N)*

*M: Marijuana ever used (Y/N)*

# Contingency table

	Marijuana =Yes		Marijuana =No	
	Cig = Yes	Cig = No	Cig = Yes	Cig = No
Alcohol = Yes	911	44	538	456
Alcohol = No	3	2	43	279

---

# Example – the Agresti data again

In R, data frame ACM is

	<b>counts</b>	<b>C</b>	<b>A</b>	<b>M</b>
<b>1</b>	<b>911</b>	<b>Y</b>	<b>Y</b>	<b>Y</b>
<b>2</b>	<b>44</b>	<b>N</b>	<b>Y</b>	<b>Y</b>
<b>3</b>	<b>3</b>	<b>Y</b>	<b>N</b>	<b>Y</b>
<b>4</b>	<b>2</b>	<b>N</b>	<b>N</b>	<b>Y</b>
<b>5</b>	<b>538</b>	<b>Y</b>	<b>Y</b>	<b>N</b>
<b>6</b>	<b>456</b>	<b>N</b>	<b>Y</b>	<b>N</b>
<b>7</b>	<b>43</b>	<b>Y</b>	<b>N</b>	<b>N</b>
<b>8</b>	<b>279</b>	<b>N</b>	<b>N</b>	<b>N</b>

# Fitting a model: R code

Cheating here!!

```
stuff<-glm(counts~C*A + C*M + A*M,  
  data=ACM, family=poisson)  
fitted.means<-predict(stuff,  
  type="response")  
N=sum(ACM$counts)  
fitted.probs<-fitted.means/N
```

# Fitted probabilities

	Marijuana = Yes		Marijuana = No	
	Cig = Yes	Cig = No	Cig = Yes	Cig = No
Alcohol = Yes	$\pi_1$	$\pi_2$	$\pi_5$	$\pi_6$
Alcohol = No	$\pi_3$	$\pi_4$	$\pi_7$	$\pi_8$

---

# Generating a synthetic table: R code

```
x = rmultinom(1, N, fitted.probs)
```

# Cell counts

	Marijuana = Yes		Marijuana = No	
	Cig = Yes	Cig = No	Cig = Yes	Cig = No
Alcohol = Yes	<i>x1</i>	<i>x2</i>	<i>x5</i>	<i>x6</i>
Alcohol = No	<i>x3</i>	<i>x4</i>	<i>x7</i>	<i>x8</i>

# Results of 10 samplings

	Sample										
Actual	1	2	3	4	5	6	7	8	9	10	
x1	911	912	914	935	946	909	932	914	899	937	913
x2	44	65	55	37	49	54	33	32	40	39	44
x3	3	5	2	2	5	2	4	4	2	4	3
x4	2	1	1	1	2	0	2	1	2	2	0
x5	538	561	536	529	484	567	504	546	518	558	573
x6	456	447	454	447	465	443	450	456	468	435	432
x7	43	42	41	41	40	33	39	39	43	37	47
x8	279	243	273	284	285	268	312	284	304	264	264

---

# Other approaches to synthetic data generation

- Graham and Penny(2007,8) - Smooth empirical table by fitting a hierarchical model: generate repeated samples and estimate parameters using multiple imputation
- Divide variables into sensitive and private groups, predict private variables using regression techniques using public variables as covariates, release multiple data sets  
Woodcock and Benedetto(2007), Reiter(2005)

---

# Contingency tables

- Measure categorical variables  $A, B, C, \dots$  on each of  $N$  individuals
- Calculate counts  $n[i, j, k, \dots]$  = number in sample having  $A=i, B=j, C=k, \dots$  (*cell counts*)
- Arrange counts in a multiway table, a *contingency table*
- For several variables, tables have a very large number of *cells*, most of which are empty i.e with  $n[i, j, k, \dots] = 0$

---

# Models for tables

Two standard related models for contingency tables:

- ❑ First assumes each cell count has a Poisson distribution with mean  $m[i,j,k]$
- ❑ Second assumes that the cell counts have a multinomial distribution with total count  $N$  and cell probabilities  $\pi[i,j,k,..]$
- ❑ Second model obtained from the first by taking distribution of the cell counts, *conditional* on the grand total  $N$

# Log-linear model

- Split the log means up into main effects interactions etc
- Set certain interactions zero
- Assume a hierarchical model, so that e.g. if model contains AB interaction , it must contain A and B main effects
- Specified in R by

$$\text{Counts} \sim (\text{A} + \text{B} + \text{C})^2$$

- Specified algebraically by

$$\text{Log}(m[i, j, k]) = \text{const} + a_i + b_j + c_k + ab_{ij} + ac_{ik} + bc_{jk}$$

---

# Mixture model

- Model for cell probabilities
- A “mixture of independents”

$$\pi[i, j, k, \dots] = \sum_{t=1}^T \tau_t \alpha_{it} \beta_{jt} \gamma_{kt} \dots$$

- T “mixture components” or “latent classes”
- Easy to simulate from
- Easy to calculate margins

---

# Fitting the models

- Log-linear
  - Use Fisher scoring – limited by number of parameters
  - Use iterated proportional fitting – limited by numbers of cells
- Mixture
  - Use EM algorithm, equivalent to functional iteration

$$\theta_{n+1} = f(\theta_n)$$

# Fisher Scoring

- Log-linear model is  $\log m_i = x_i^T \beta$
- Parameter  $\beta$  updated using updating equation

$$\beta_{n+1} = \beta_n - I^{-1}(\beta_n) S(\beta_n)$$

$$S(\beta_n) = \sum_i (y_i - \exp(x_i^T \beta_n)) x_i$$

$$I(\beta_n) = \sum_i x_i x_i^T \exp(x_i^T \beta_n)$$

- Requires table of counts  $y_i$  to calculate score  $S(\beta)$  and information  $I(\beta)$ , but don't need to store whole table in computer memory

---

# IPF algorithm

- To fit models need only certain margins
- e.g. to fit model  $A*B + A*C + B*C$  need only the AB, AC and BC margins
- Start with a complete table with all cells set to 1
- Successively adjust the margins of the complete table to match the supplied margins
- Algorithm converges to the ML estimates of the cells
- Cumbersome to implement if whole table can't be stored in computer memory, otherwise straightforward

---

# Tables of counts/tables of relative frequencies (Empirical probabilities)

- Form table of empirical probabilities
- Often table refers to population, not sample
- Can regard model fitting by ML as approximation, i.e. choosing the table of probabilities in a class (e.g. log-linear) that best approximates the empirical table in the Kullback-Leibler sense

$$KL(\pi^{(EMP)}, \pi^{(LL)}) = \sum_i \pi_i^{(EMP)} \log(\pi_i^{(EMP)} / \pi_i^{(LL)})$$

---

# Fitting model to an empirical table

- Fit model to available data (could be the whole census)
- Most cells in empirical table are zero, most non-zero cells unique
- Fitted model preserves empty cells
- Effectively equivalent to resampling the empirical table
- Not much confidentiality protection

# Solution: smooth empirical table

- Form the independence table by calculating the one dimensional margins i.e. for factors A, B, C

$\pi_i^{(A)}$  = proportion of sample having A=i

$\pi_j^{(B)}$  = proportion of sample having B=j

$\pi_k^{(C)}$  = proportion of sample having C=k

Independence table has cell probabilities

$$\pi_{ijk}^{(IND)} = \pi_i^{(A)} \times \pi_j^{(B)} \times \pi_k^{(C)}$$

- Smoothed table has cell probabilities

$$\pi_{ijk} = \tau \pi_{ijk}^{(EMP)} + (1 - \tau) \pi_{ijk}^{(IND)}$$

# Justification

- Assume a Dirichlet prior for  $\{\pi_i\}$ , parameters  $\{\beta_i\}$ , denote typical cell by  $i$
- Conditional distribution of table counts  $\{n_i\}$ , given  $\{\pi_i\}$ , is multinomial
- Posterior distribution of  $\{\pi_i\}$  is Dirichlet, with parameters  $\{n_i + \beta_i\}$ ,
- Posterior mean is of form

$$\pi_i = \tau \pi_i^{(EMP)} + (1 - \tau) \beta_i / \sum_i \beta_i, \quad t = N / (N + \sum_i \beta_i)$$

---

# Generating data: we use 3 methods

- From log-linear model
  - If number of cells is small, say  $< 8$  million, use inversion method
  - If number of cells is larger, and we have calculated the model parameters, use the Metropolis-Hastings method
- From the mixture model, method is simple

# Inversion method

		B	
		0	1
A	0	$\pi_1$	$\pi_2$
	1	$\pi_3$	$\pi_4$

- Draw a uniform  $[0, 1]$  variate  $U$
- If  $U \leq \pi_1$ , return  $(A=0, B=0)$
- If  $\pi_1 < U \leq \pi_1 + \pi_2$ , return  $(A=0, B=1)$
- If  $\pi_1 + \pi_2 < U \leq \pi_1 + \pi_2 + \pi_3$ , return  $(A=1, B=0)$
- If  $\pi_1 + \pi_2 + \pi_3 < U \leq 1$ , return  $(A=1, B=1)$

---

# Metropolis -Hastings

- Consider a Markov chain, states are cells
- Transition matrix  $P = (P_{ij})$
- $P_{ij} = \text{pr}[\text{skip to cell } j \mid \text{in cell } i]$
- Stationary distribution
$$\pi_i = \lim_{t \rightarrow \infty} \text{Pr}[\text{In cell } i \text{ after } t \text{ skips}]$$
- Choose  $P$  to get desired  $\pi_i$
- How?

---

# Metropolis-Hastings (cont)

- Suppose we are in cell  $i$ . Select a cell  $j$  using some distribution  $q_j$  (e.g. the independence distribution)
- Compute  $\alpha_{ij} = \min(1, \pi_i q_j / \pi_j q_i)$
- Generate  $U$ . If  $U < \alpha_{ij}$  skip to  $j$ , else stay in  $i$ .
- Works because ratio  $\pi_i / \pi_j$  is easy to compute.

# Sampling from a mixture model

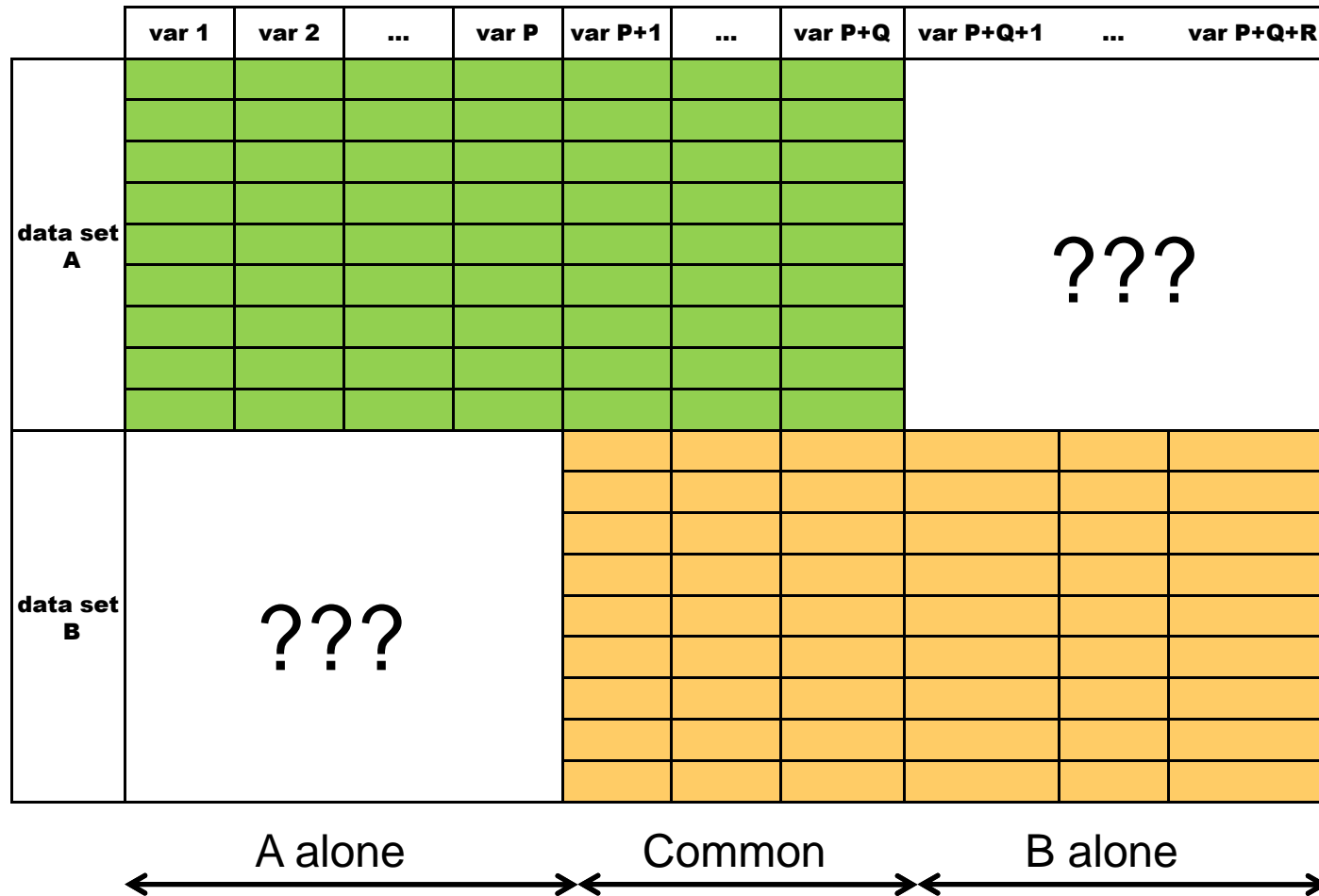
- Mixture model is

$$p[i, j, k, \dots] = \Pr(A=i, B=j, C=k, \dots)$$

$$\pi[i, j, k, \dots] = \sum_{t=1}^T \tau_t \alpha_{it} \beta_{jt} \gamma_{kt} \dots$$

- Select a latent class  $t$  using  $\tau_t$
- Then select A, B, C, ... independently using  $\{\alpha_{it}\}, \{\beta_{jt}\}, \{\gamma_{kt}\}$

# Fitting mixtures using separate data sets



---

# 2001 Census data

- CURF has 34 variables
  - Table Builder has many 3-dimensional margins
  - Ethnicity and religion not compatible between CURF and Table Builder, so eliminate
  - Many empty cells (structural zeroes)
  - Consider 3 nested populations
    - Usually resident
    - Usually resident 15+
    - Employed usually resident 15+
-

---

# Usually resident population

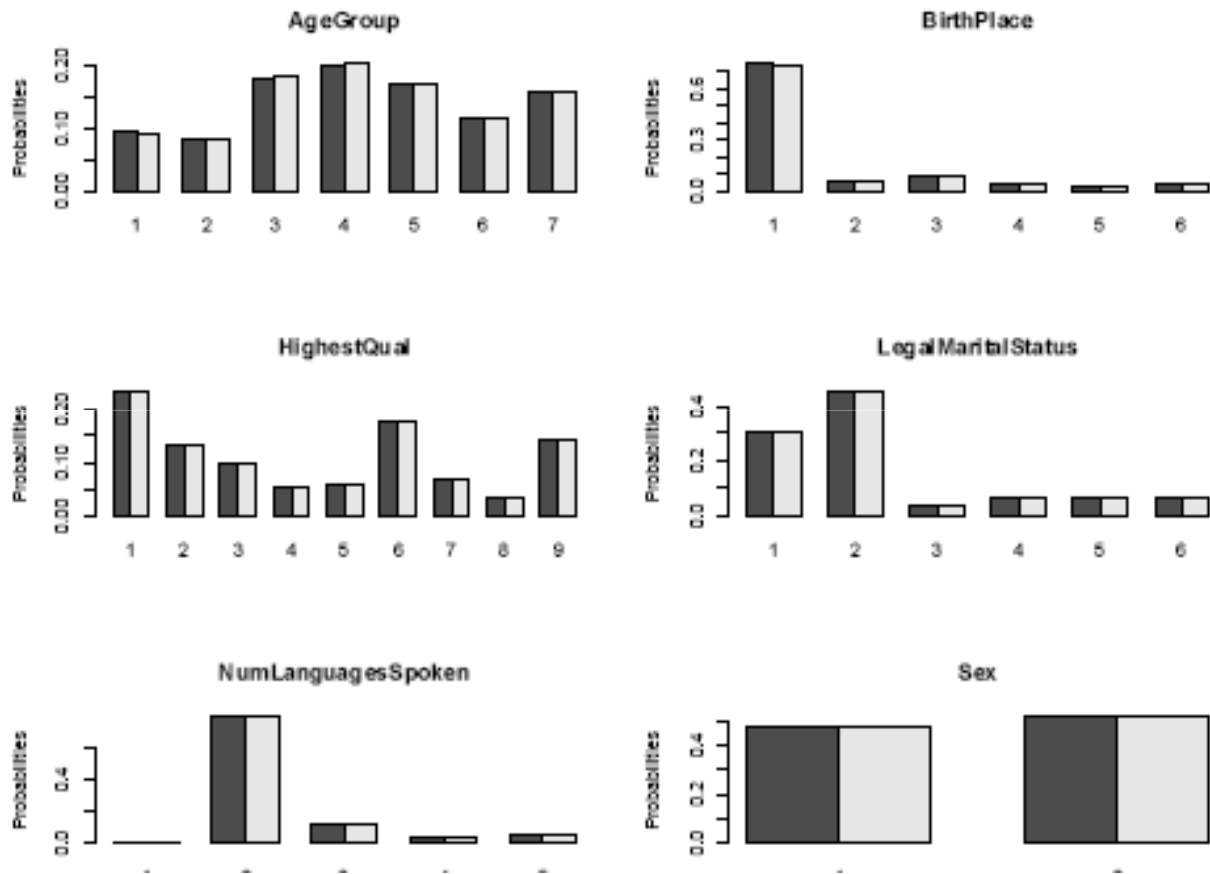
- 5 variables, 7200 cells, CURF has 74,767 records
- Strategies
  - Generate data from smoothed table using inversion method
  - Fit log-linear model (all 3fi) to smoothed table, generate data using inversion method
  - Fit mixture model, generate data from mixture

---

# Usually resident population 15+

- 10 variables, 48,988,800 cells, CURF has 35882 rows
- Split variables up into 2 overlapping groups
  - First 8 (816,480 cells)
  - Last 8 (1,166,400 cells)
- Fit log-linear model to first 8, last 8 using IPF
- Use separate tables to fit a mixture model

# Results

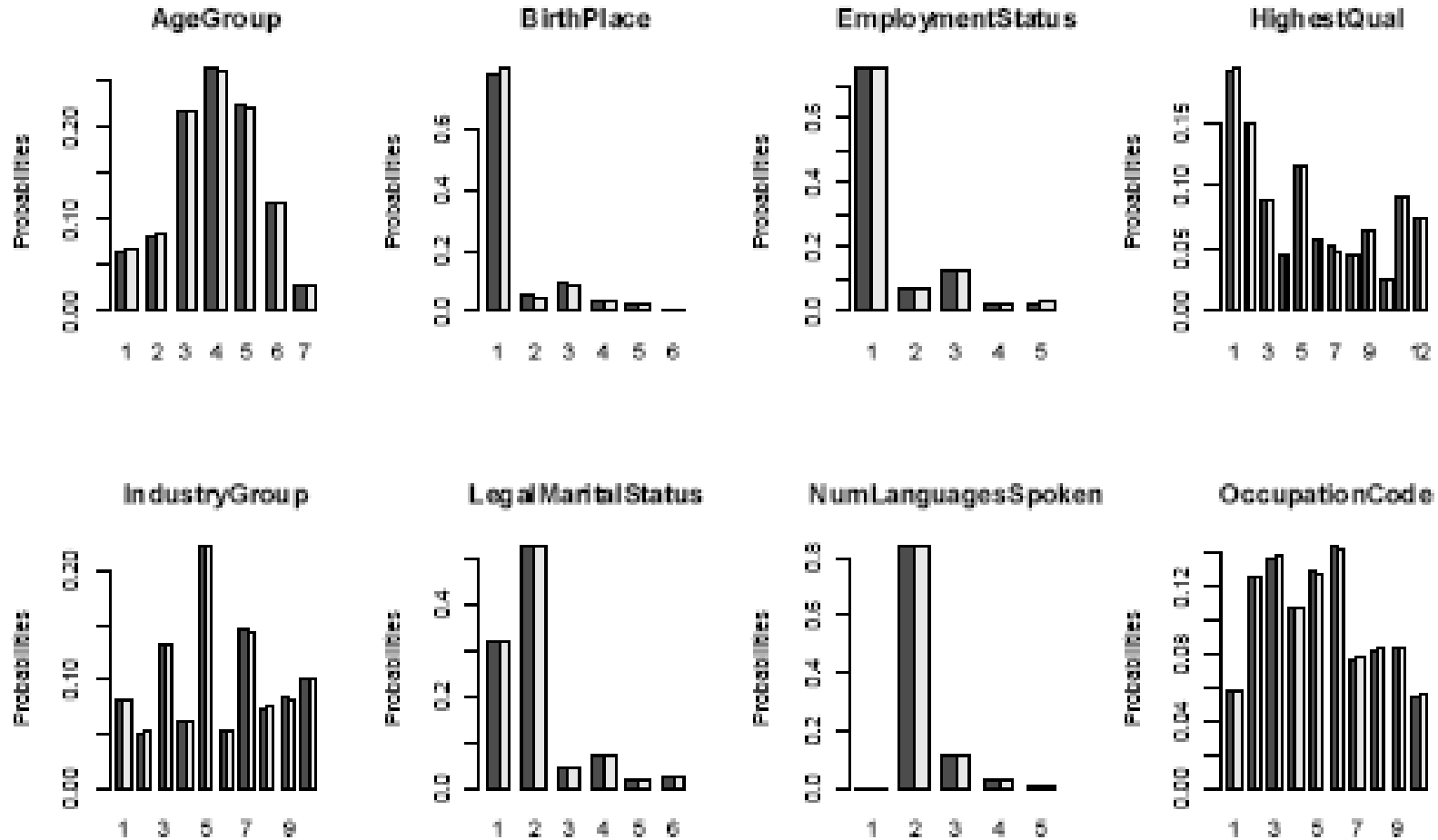


---

# Employed Usually resident population 15+

- 16 variables, 2,821,754,880,000 cells, CURF has 34,492 rows
- Split variables up into 5 overlapping groups, ranging from 373,248 to 777,600 cells
- Fit log-linear model to each group using IPF, all  $3f_i$
- Use 5 separate tables to fit a mixture model

# Results



---

# Conclusions

- For small tables, fit log-linear model or mixture model to smoothed tables
- For bigger tables, split variables into overlapping groups, fit log-linear models to separate groups as above, then fit an overall mixture model
- Takes a while, but gets there in the end!

---

# Questions

- How good does the approximation of the empirical table have to be before we can use synthetic data for “inference”?
- How to best split the variables into groups?
- How many latent classes to use?
- How to speed up the mixture model fitter?