Communities discovered in a 3.7M node network of U.S. Patents

[Gopalan and Blei, PNAS 2013]
Topics found in 1.8M articles from the New York Times

[Hoffman, Blei, Wang, Paisley, JMLR 2013]
Population analysis of 2 billion genetic measurements

[Gopalan, Hao, Blei, Storey, submitted]
Neuroscience analysis of 220 million fMRI measurements

[Manning et al., PLOS ONE 2014]
Analysis of 1.7M taxi trajectories, in Stan

[Kucukelbir et al., 2016]
The probabilistic pipeline

- Customized data analysis is important to many fields.
- Pipeline separates assumptions, computation, application
- Eases collaborative solutions to statistics problems
The probabilistic pipeline

- **Inference** is the key algorithmic problem.
- Answers the question: What does this model say about this data?
- Our goal: **General** and **scalable** approaches to inference
The probabilistic pipeline

![Diagram of the probabilistic pipeline]

- **Make assumptions**
- **Discover patterns**
- **Predict & Explore**

- Variational methods: inference as optimization [Jordan et al., 1999]
- Scale up with **stochastic variational inference** (SVI)
- Generalize with **black box variational inference** (BBVI)
Figure S2: Population structure inferred from the TGP data set using the TeraStructure algorithm at three values for the number of populations $K$. The visualization of the $✓$'s in the Figure shows patterns consistent with the major geographical regions. Some of the clusters identify a specific region (e.g. red for Africa) while others represent admixture between regions (e.g. green for Europeans and Central/South Americans). The presence of clusters that are shared between different regions demonstrates the more continuous nature of the structure. The new cluster from $K=7$ to $K=8$ matches structure differentiating between American groups. For $K=9$, the new cluster is unpopulated.

“Stochastic variational inference” [Hoffman et al., 2013, JMLR]
Black box variational inference” [Ranganath et al., 2014, AISTATS]
Edward: A library for probabilistic modeling, inference, and criticism

github.com/blei-lab/edward

(lead by Dustin Tran)
Inference of Population Structure Using Multilocus Genotype Data

Jonathan K. Pritchard, Matthew Stephens and Peter Donnelly
Department of Statistics, University of Oxford, Oxford OX1 3TG, United Kingdom
Manuscript received September 23, 1999
Accepted for publication February 18, 2000

ABSTRACT
We describe a model-based clustering method for using multilocus genotype data to infer population structure and assign individuals to populations. We assume a model in which there are $K$ populations (where $K$ may be unknown), each of which is characterized by a set of allele frequencies at each locus. Individuals in the sample are assigned (probabilistically) to populations, or jointly to two or more populations if their genotypes indicate that they are admixed. Our model does not assume a particular mutation process, and it can be applied to most of the commonly used genetic markers, provided that they are not closely linked. Applications of our method include demonstrating the presence of population structure, assigning individuals to populations, studying hybrid zones, and identifying migrants and admixed individuals. We show that the method can produce highly accurate assignments using modest numbers of loci—e.g., seven microsatellite loci in an example using genotype data from an endangered bird species. The software used for this article is available from http://www.stats.ox.ac.uk/~pritch/home.html.

TeraStructure: Fast inference for the PSD model.
Figure 2: TeraStructure recovers the underlying per-individual population proportions on the simulated data sets generated via Scenario A. Each panel shows a visualization of the simulation parameters and the inferred expected population proportions for all individuals in a data set. The current state-of-the-art algorithms cannot complete their analyses of 100,000 and 1,000,000 individuals. TeraStructure is able to analyze data of this size and gives highly accurate estimates.
TeraStructure is fast.

<table>
<thead>
<tr>
<th>Data set</th>
<th>$N$</th>
<th>$L$</th>
<th>$S$</th>
<th>TeraStructure</th>
<th>ADMIXTURE</th>
<th>fastSTRUCTURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>HGDP</td>
<td>940</td>
<td>644,258</td>
<td>0.9</td>
<td>&lt; 1</td>
<td>&lt; 1</td>
<td>12</td>
</tr>
<tr>
<td>TGP</td>
<td>1718</td>
<td>1,854,622</td>
<td>0.5</td>
<td>3</td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>Scenario A (10K)</td>
<td>10,000</td>
<td>1,000,000</td>
<td>1.0</td>
<td>9</td>
<td>28</td>
<td>216</td>
</tr>
<tr>
<td>Scenario A (100K)</td>
<td>100,000</td>
<td>1,000,000</td>
<td>0.7</td>
<td>158</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scenario A (1M)</td>
<td>1,000,000</td>
<td>1,000,000</td>
<td>0.5</td>
<td>509</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Scenario B</td>
<td>10,000</td>
<td>100,000</td>
<td>1.0</td>
<td>6.9</td>
<td>31</td>
<td>140</td>
</tr>
<tr>
<td>Scenario B</td>
<td>10,000</td>
<td>1,000,000</td>
<td>0.5</td>
<td>9.3</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>
TeraStructure is accurate.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Replication</th>
<th>( N )</th>
<th>( L )</th>
<th>Median per-individual KL divergence</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TeraStructure</td>
</tr>
<tr>
<td>Scenario A (10K)</td>
<td>1</td>
<td>10,000</td>
<td>1,000,000</td>
<td>0.016</td>
</tr>
<tr>
<td>Scenario A (10K)</td>
<td>2</td>
<td>10,000</td>
<td>1,000,000</td>
<td>0.009</td>
</tr>
<tr>
<td>Scenario A (10K)</td>
<td>3</td>
<td>10,000</td>
<td>1,000,000</td>
<td>0.020</td>
</tr>
<tr>
<td>Scenario A (100K)</td>
<td>1</td>
<td>100,000</td>
<td>1,000,000</td>
<td>0.006</td>
</tr>
<tr>
<td>Scenario A (100K)</td>
<td>2</td>
<td>100,000</td>
<td>1,000,000</td>
<td>0.013</td>
</tr>
<tr>
<td>Scenario A (100K)</td>
<td>3</td>
<td>100,000</td>
<td>1,000,000</td>
<td>0.009</td>
</tr>
<tr>
<td>Scenario A (1M)</td>
<td>1</td>
<td>1,000,000</td>
<td>1,000,000</td>
<td>0.015</td>
</tr>
<tr>
<td>Scenario B</td>
<td>1</td>
<td>10,000</td>
<td>100,000</td>
<td>0.21</td>
</tr>
<tr>
<td>Scenario B</td>
<td>2</td>
<td>10,000</td>
<td>100,000</td>
<td>0.27</td>
</tr>
<tr>
<td>Scenario B</td>
<td>3</td>
<td>10,000</td>
<td>100,000</td>
<td>0.26</td>
</tr>
<tr>
<td>Scenario B</td>
<td>1</td>
<td>10,000</td>
<td>1,000,000</td>
<td>0.16</td>
</tr>
<tr>
<td>Scenario B</td>
<td>2</td>
<td>10,000</td>
<td>1,000,000</td>
<td>0.23</td>
</tr>
<tr>
<td>Scenario B</td>
<td>3</td>
<td>10,000</td>
<td>1,000,000</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Table S2: The accuracy of the algorithms on simulated data generated via Scenario A. TeraStructure is the only algorithm that was able to complete its analysis on the simulated data sets with \( N = 100,000 \) individuals and \( N = 1,000,000 \) individuals. On these massive data sets, TeraStructure found a highly accurate fit to the data (see also Figure 2). On smaller simulated data, TeraStructure finds a fit to the data that is closer to the simulation model than either of the other methods. The number of ancestral populations is set to the number of ancestral populations used in the simulation: \( K = 6 \).
Randomly initialize population proportions

Subsample a SNP

Infer ancestral population frequencies for the SNP

Update population proportions

Check for convergence

Massive genotype data

Fitted population proportions

TeraStructure
Why does TeraStructure work?
1. Introduction to variational inference
2. Coordinate ascent variational inference (CAVI)
3. CAVI for the PSD model
4. Scalable inference with stochastic variational inference (SVI)
5. SVI for the PSD model
6. Black box variational inference