

A. U.S. presidents dataset

Hayden (2005) studies this dataset (and provides a fascinating educational perspective too).

President	Days
Washington	2864
Adams	1460
Jefferson	2921
Madison	2921
Monroe	2921
Adams	1460
Jackson	2921
VanBuren	1460
Harrison	31
Tyler	1427
Polk	1460
Taylor	491
Filmore	967
Pierce	1460
Buchanan	1460
Lincoln	1503
Johnson	1418
Grant	2921
Hayes	1460
Garfield	199
Arthur	1260
Cleveland	1460
Harrison	1460
Cleveland	1460
McKinley	1655
Roosevelt	2727
Taft	1460
Wilson	2921
Harding	881
Coolidge	2039
Hoover	1460
Roosevelt	4452
Truman	2810
Eisenhower	2922
Kennedy	1036
Johnson	1886
Nixon	2027
Ford	895
Carter	1461
Reagan	2922
Bush	1461
Clinton	2922
Bush	1110

Table 6. Presidents of the United States of America: days in office.

B. Proof of Lemma 1

The widely applicable posterior dispersion index (WAPDI) is defined as

$$\text{WAPDI}(n) = \frac{\mathbb{V}_{\boldsymbol{\theta}|\mathbf{x}}[\log p(x_n | \boldsymbol{\theta})]}{\log \mathbb{E}_{\boldsymbol{\theta}|\mathbf{x}}[p(x_n | \boldsymbol{\theta})]}.$$

Suppress the conditioning on $\mathbf{x}$; it is notational baggage. Expand the numerator as

$$\mathbb{V}_{\boldsymbol{\theta}}[\log p(x_n | \boldsymbol{\theta})] = \mathbb{E}_{\boldsymbol{\theta}}[(\log p(x_n | \boldsymbol{\theta}))^2] - \mathbb{E}_{\boldsymbol{\theta}}[\log p(x_n | \boldsymbol{\theta})]^2.$$

Consider a Taylor expansion of the first term

$$\begin{aligned} \mathbb{E}_{\boldsymbol{\theta}}[(\log p(x_n | \boldsymbol{\theta}))^2] &= \log p^2(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]) + \frac{\mathbb{V}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]}{2} (\log p^2(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]))'' + \dots \\ &= \log p^2(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]) + \frac{\mathbb{V}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]}{2} \left\{ 2(\log p'(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]))^2 \right. \\ &\quad \left. + 2 \log p(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]) (\log p(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]))'' \right\} + \dots. \end{aligned}$$

Next, consider a Taylor expansion of the second term

$$\mathbb{E}_{\boldsymbol{\theta}}[\log p(x_n | \boldsymbol{\theta})]^2 = \left(\log p(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]) + \frac{\mathbb{V}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]}{2} (\log p(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]))'' + \dots \right)^2.$$

Assume that $\mathbb{V}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]^2 \ll \mathbb{V}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]$.

Then the second-order Taylor approximation of the numerator is

$$\mathbb{V}_{\boldsymbol{\theta}}[\log p(x_n | \boldsymbol{\theta})] \approx (\log p'(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]))^2 \mathbb{V}_{\boldsymbol{\theta}}[\boldsymbol{\theta}],$$

which gives the second-order Taylor approximation to WAPDI as

$$\text{WAPDI}(n) \approx \frac{(\log p'(x_n | \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]))^2 \mathbb{V}_{\boldsymbol{\theta}}[\boldsymbol{\theta}]}{\log \mathbb{E}_{\boldsymbol{\theta}}[p(x_n | \boldsymbol{\theta})]}.$$

C. Hierarchical logistic regression model

Hierarchical logistic regression models classification tasks in an intuitive way. We study three variants of [Gelman & Hill \(2006\)](#)'s model for the 1988 CBS United States election survey dataset. Stan code follows the equations.

First model: no age or education. The likelihood of voting Republican is

$$\Pr(y_n = 1) = \text{sigmoid}\left(\beta^{\text{female}} \cdot \text{female}_n + \beta^{\text{black}} \cdot \text{black}_n + \alpha_{s[n]}^{\text{state}}\right),$$

with priors

$$\begin{aligned}\alpha_j^{\text{state}} &\sim \mathcal{N}(\mu_{\text{state}}, \sigma_{\text{state}}) \\ \mu_{\text{state}} &\sim \mathcal{N}(0, 10) \\ \sigma_{\text{state}} &\sim \mathcal{N}(0, 10) \\ \beta &\sim \mathcal{N}(0, 1).\end{aligned}$$

[Figure 6](#) presents Stan code that implements this model.

Second model: with age. The likelihood of voting Republican is

$$\Pr(y_n = 1) = \text{sigmoid}\left(\beta^{\text{female}} \cdot \text{female}_n + \beta^{\text{black}} \cdot \text{black}_n + \alpha_{s[n]}^{\text{state}} + \alpha_{a[n]}^{\text{age}}\right),$$

with priors

$$\begin{aligned}\alpha_j^{\text{state}} &\sim \mathcal{N}(\mu_{\text{state}}, \sigma_{\text{state}}) \\ \mu_{\text{state}} &\sim \mathcal{N}(0, 10) \\ \sigma_{\text{state}} &\sim \mathcal{N}(0, 10) \\ \alpha_j^{\text{age}} &\sim \mathcal{N}(\mu_{\text{age}}, \sigma_{\text{age}}) \\ \mu_{\text{age}} &\sim \mathcal{N}(0, 10) \\ \sigma_{\text{age}} &\sim \mathcal{N}(0, 10) \\ \beta &\sim \mathcal{N}(0, 1).\end{aligned}$$

[Figure 7](#) presents Stan code that implements this model.

Third model: with education. The likelihood of voting Republican is

$$\Pr(y_n = 1) = \text{sigmoid}\left(\beta^{\text{female}} \cdot \text{female}_n + \beta^{\text{black}} \cdot \text{black}_n + \alpha_{s[n]}^{\text{state}} + \alpha_{a[n]}^{\text{edu}}\right),$$

with priors

$$\begin{aligned}\alpha_j^{\text{state}} &\sim \mathcal{N}(\mu_{\text{state}}, \sigma_{\text{state}}) \\ \mu_{\text{state}} &\sim \mathcal{N}(0, 10) \\ \sigma_{\text{state}} &\sim \mathcal{N}(0, 10) \\ \alpha_j^{\text{edu}} &\sim \mathcal{N}(\mu_{\text{edu}}, \sigma_{\text{edu}}) \\ \mu_{\text{edu}} &\sim \mathcal{N}(0, 10) \\ \sigma_{\text{edu}} &\sim \mathcal{N}(0, 10) \\ \beta &\sim \mathcal{N}(0, 1).\end{aligned}$$

[Figure 8](#) presents Stan code that implements this model.

```
data {
  int<lower=0> N;
  int<lower=0> n_age;
  int<lower=0> n_age_edu;
  int<lower=0> n_edu;
  int<lower=0> n_region_full;
  int<lower=0> n_state;
  int<lower=0,upper=n_age> age[N];
  int<lower=0,upper=n_age_edu> age_edu[N];
  vector<lower=0,upper=1>[N] black;
  int<lower=0,upper=n_edu> edu[N];
  vector<lower=0,upper=1>[N] female;
  int<lower=0,upper=n_region_full> region_full[N];
  int<lower=0,upper=n_state> state[N];
  vector[N] v_prev_full;
  int<lower=0,upper=1> y[N];
}

parameters {
  vector[2] beta;
  vector[n_state] s;
  real<lower=0> sigma_s;
  real mu_s;
}

model {
  vector[N] y_hat;

  mu_s ~ normal(0, 10);
  sigma_s ~ lognormal(0, 10);
  s ~ normal(mu_s, sigma_s);

  beta ~ normal(0, 1);

  for (n in 1:N) {
    y_hat[n] = beta[1] * black[n]
              + beta[2] * female[n]
              + s[state[n]];
  }
  y ~ bernoulli_logit(y_hat);
}

generated quantities {
  vector[N] lp_users;
  for (n in 1:N) {
    lp_users[n] = bernoulli_logit_log(
      y[n],
      beta[1] * black[n]
      + beta[2] * female[n]
      + s[state[n]]);
  }
}
```

Figure 6. Stan code for the first hierarchical logistic regression model.

```

data {
  int<lower=0> N;
  int<lower=0> n_age;
  int<lower=0> n_age_edu;
  int<lower=0> n_edu;
  int<lower=0> n_region_full;
  int<lower=0> n_state;
  int<lower=0,upper=n_age> age[N];
  int<lower=0,upper=n_age_edu> age_edu[N];
  vector<lower=0,upper=1>[N] black;
  int<lower=0,upper=n_edu> edu[N];
  vector<lower=0,upper=1>[N] female;
  int<lower=0,upper=n_region_full> region_full[N];
  int<lower=0,upper=n_state> state[N];
  vector[N] v_prev_full;
  int<lower=0,upper=1> y[N];
}

parameters {
  vector[2] beta;
  vector[n_state] s;
  real<lower=0> sigma_s;
  real mu_s;
  vector[n_age] a;
  real<lower=0> sigma_a;
  real mu_a;
}

model {
  vector[N] y_hat;

  mu_s ~ normal(0, 10);
  sigma_s ~ lognormal(0, 10);
  s ~ normal(mu_s, sigma_s);

  mu_a ~ normal(0, 10);
  sigma_a ~ lognormal(0, 10);
  a ~ normal(mu_a, sigma_a);

  beta ~ normal(0, 1);

  for (n in 1:N) {
    y_hat[n] = beta[1] * black[n]
              + beta[2] * female[n]
              + s[state[n]]
              + a[age[n]];
  }
  y ~ bernoulli_logit(y_hat);
}

generated quantities {
  vector[N] lp_users;
  for (n in 1:N) {
    lp_users[n] = bernoulli_logit_log(
      y[n],
      beta[1] * black[n]
      + beta[2] * female[n]
      + s[state[n]]
      + a[age[n]]);
  }
}

```

Figure 7. Stan code for the second hierarchical logistic regression model.

```

data {
  int<lower=0> N;
  int<lower=0> n_age;
  int<lower=0> n_age_edu;
  int<lower=0> n_edu;
  int<lower=0> n_region_full;
  int<lower=0> n_state;
  int<lower=0,upper=n_age> age[N];
  int<lower=0,upper=n_age_edu> age_edu[N];
  vector<lower=0,upper=1>[N] black;
  int<lower=0,upper=n_edu> edu[N];
  vector<lower=0,upper=1>[N] female;
  int<lower=0,upper=n_region_full> region_full[N];
  int<lower=0,upper=n_state> state[N];
  vector[N] v_prev_full;
  int<lower=0,upper=1> y[N];
}

parameters {
  vector[2] beta;
  vector[n_state] s;
  real<lower=0> sigma_s;
  real mu_s;
  vector[n_edu] e;
  real<lower=0> sigma_e;
  real mu_e;
}

model {
  vector[N] y_hat;

  mu_s ~ normal(0, 10);
  sigma_s ~ lognormal(0, 10);
  s ~ normal(mu_s, sigma_s);

  mu_e ~ normal(0, 10);
  sigma_e ~ lognormal(0, 10);
  e ~ normal(mu_e, sigma_e);

  beta ~ normal(0, 1);

  for (n in 1:N) {
    y_hat[n] = beta[1] * black[n]
              + beta[2] * female[n]
              + s[state[n]]
              + e[edu[n]];
  }
  y ~ bernoulli_logit(y_hat);
}

generated quantities {
  vector[N] lp_users;
  for (n in 1:N) {
    lp_users[n] = bernoulli_logit_log(
      y[n],
      beta[1] * black[n]
      + beta[2] * female[n]
      + s[state[n]]
      + e[edu[n]]);
  }
}

```

Figure 8. Stan code for the second hierarchical logistic regression model.

D. Hierarchical Poisson factorization model

Hierarchical Poisson factorization models a matrix of counts as a low-dimensional inner product. (Gopalan et al., 2015) present the model in detail along with an efficient variational inference algorithm for inference. We summarize the model below.

Model. Consider a $U \times I$ dataset, with non-negative integer elements x_{ui} . It helps to think of u as an index over “users” and i as an index over “items”.

The likelihood for each measurements is

$$p(x_{ui}) = \text{Poisson}(x_{ui} ; \theta_u^\top \beta_i)$$

where θ is a $U \times K$ matrix of non-negative real-valued latent variables; it represents “user preferences”. Similarly, β is a $K \times I$ matrix of non-negative real-valued latent variables; it represents “item attributes”.

The priors for these variables are

$$p(\theta_{uk}) = \text{Gam}(\theta_{uk} ; a, \xi_u)$$

$$p(\beta_{ki}) = \text{Gam}(\beta_{ki} ; c, \eta_i)$$

where ξ is a U vector of non-negative real-valued latent variables; it represents “user activity”. Similarly, η is a I vector of non-negative real-valued latent variables; it represents “item popularity”.

The prior for these hierarchical latent variable are

$$p(\xi_u) = \text{Gam}(\xi_u ; a', a'/b')$$

$$p(\eta_i) = \text{Gam}(\eta_i ; c', c'/d').$$

Parameters for IRi analysis. The parameters we used were

$$K = 20$$

$$a = 0.3/\sqrt{K}$$

$$c = 0.3/\sqrt{K}$$

$$a' = 1.5$$

$$b' = 0.3$$

$$c' = 1.5$$

$$d' = 0.3.$$

Inference. We used the coordinate ascent variational inference algorithm provided by the authors of (Gopalan et al., 2015).

E. Population genetics data and model

Population genetics studies ancestral trends of genomic mutations. Consider N individuals and L locations on the genome. For each location, we measure whether each individual reveals a mutation. This gives an $(N \times L)$ dataset $\mathbf{x}$ where $x_{nl} \in 0, 1, 2, 3$. (We assume two specific forms of mutation; 3 encodes a missing observation.)

Model. Pritchard et al. (2000) propose a probabilistic model for this kind of data. Represent K ancestral populations with a latent variable ϕ . This is a $(K \times L)$ matrix of mutation probabilities. Place a beta prior for each probability. Each individual mixes these populations. Denote this with another latent variable θ . This is a $(N \times K)$ matrix of mixture proportions. Place a Dirichlet prior for each individual. The likelihood of each mutation is a K -mixture of categorical distributions.

Data. We study a subset of the Hapmap 3 dataset from (Mimno et al., 2015). This includes $N = 324$ individuals and $L = 13\,928$ locations on the genome. Four geographic regions are represented: 49 ASW, 112 CEU, 50 MEX, and 113 YRI. Mimno et al. (2015) expect this data to exhibit at most $K = 3$ ancestral populations; the full Hapmap 3 dataset exhibits $K = 6$ populations.

In more detail, this data studies single-nucleotide polymorphisms (SNPs). The data is pre-processed from the raw genetic observations such that:

- non-SNPs are removed (i.e. genes with more than one location changed),
- SNPs with low entropy compared to the dominant mutation at each location are removed,
- SNPs that are too close to each other on the genome are removed.

Inference. We use the fastSTRUCTURE software suite (Raj et al., 2014) to perform variational inference. We use the default parameters and only specify $K = 3$.

References

- Gelman, Andrew and Hill, Jennifer. *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press, 2006.
- Gopalan, Prem, Hofman, Jake M, and Blei, David M. Scalable recommendation with hierarchical Poisson factorization. *UAI*, 2015.
- Hayden, Robert W. A dataset that is 44% outliers. *J Stat Educ*, 13(1), 2005.
- Mimno, David, Blei, David M, and Engelhardt, Barbara E. Posterior predictive checks to quantify lack-of-fit in admixture models of latent population structure. *Proceedings of the National Academy of Sciences*, 112(26), 2015.
- Pritchard, Jonathan K, Stephens, Matthew, and Donnelly, Peter. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959, 2000.
- Raj, Anil, Stephens, Matthew, and Pritchard, Jonathan K. fastSTRUCTURE: variational inference of population structure in large SNP data sets. *Genetics*, 197(2):573–589, 2014.