Multilevel regression

ISTA 410 / INFO 510: Bayesian Modeling and Inference

U. of Arizona School of Information

April 7, 2021

Outline

Last time:

- Interactions
- Generalized linear models

Today:

- Multilevel regression / hierarchical linear models
- Varying intercepts and slopes

Hierarchical linear models

Recap: linear regression as a Bayesian model

Remember the basic framework we had for a linear model in the Bayesian setting:

$$
\begin{aligned}
y_i &\sim \text{Normal}(\mu_i, \sigma) \\
\mu_i &= \alpha + \beta \cdot \text{x}_\text{i} \\
\sigma &\sim \text{HalfCauchy}(\phi) \\
\beta_i &\sim \text{Normal}(0, \sigma_\beta) \\
\alpha &\sim \text{Normal}(0, \sigma_\alpha)
\end{aligned}
$$

(different prior choices possible, of course!)

Recap: hierarchical models

Recall the idea of a hierarchical model:

- Observations are grouped into clusters
- Model parameters for each group come from a prior distribution dependent on population-level *hyperparameters*
- Allows for "partial pooling"; clusters don't all have the same model parameters, but some information is shared across clusters
- Effect: shrinkage toward population parameters, especially for clusters with few observations

Example: ozone in Beijing

Data set: ~ 5 years of air quality monitoring data from Beijing

- Weather properties: temperature, pressure, dewpoint, rain, wind speed
- Pollutants: ozone, sulfur dioxide, nitrous oxide, carbon monoxide, particulates
- Measurements collected hourly at 12 monitoring stations

Simple modeling task: model ozone (in $\mu\text{g}/\text{m}^3$, about twice ppb) as a function of temperature

Preprocessing: group measurements by date/station

Basic model

Exploratory plotting suggests:

- the log maximum daily ozone reading is linearly associated with high temperature
- use max instead of average to prevent daily temporal associations from contributing

Preliminary model:

$$\log O_3 \sim \text{Normal}(\theta, \sigma)$$
$$\theta = \alpha + \beta_T T_{\max}$$
$$\alpha \sim \text{Normal}(0, 3)$$
$$\beta \sim \text{Normal}(0, 1)$$

Prior predictive reasonableness

Hang on, let's check those priors:

$$\alpha \sim \text{Normal}(0, 3)$$

$$\beta \sim \text{Normal}(0, 1)$$

Prior predictive reasonableness

Hang on, let's check those priors:

$$\alpha \sim \text{Normal}(0, 3)$$
$$\beta \sim \text{Normal}(0, 1)$$

If $\alpha = 3, \beta = 1$, then on a 30 degree day we get $\log O_3 \approx 33$; meaning about 100 trillion ppb.

A really bad ozone day might be a couple hundred ppb (about $\log O_3 \approx 6$ or 7). Let's rein in these priors.

Prior predictive reasonableness

With new priors:

$$\begin{aligned}
\log O_3 &\sim \text{Normal}(\theta, \sigma) \\
\theta &= \alpha + \beta_T T_{\max} \\
\alpha &\sim \text{Normal}(0, 2) \\
\beta &\sim \text{Normal}(0, 0.1)
\end{aligned}$$

Now a high estimate combined with a hot day gives us something more like $\log O_3 \approx 6$.

Let's proceed!

Model specification

In Python:

```
with pm.Model() as linear_model:
    alpha = pm.Normal('alpha', 0, 2)
    beta = pm.Normal('beta', 0, 0.1)

    # Model equation
    theta = alpha[dailies.dropna()['station_id']]
        + beta * dailies.dropna()['TEMP']

    # Likelihood
    y_ = pm.Normal('y', mu=theta, sigma = sigma,
         observed = dailies.dropna()['log_ozone'])
```

Model results

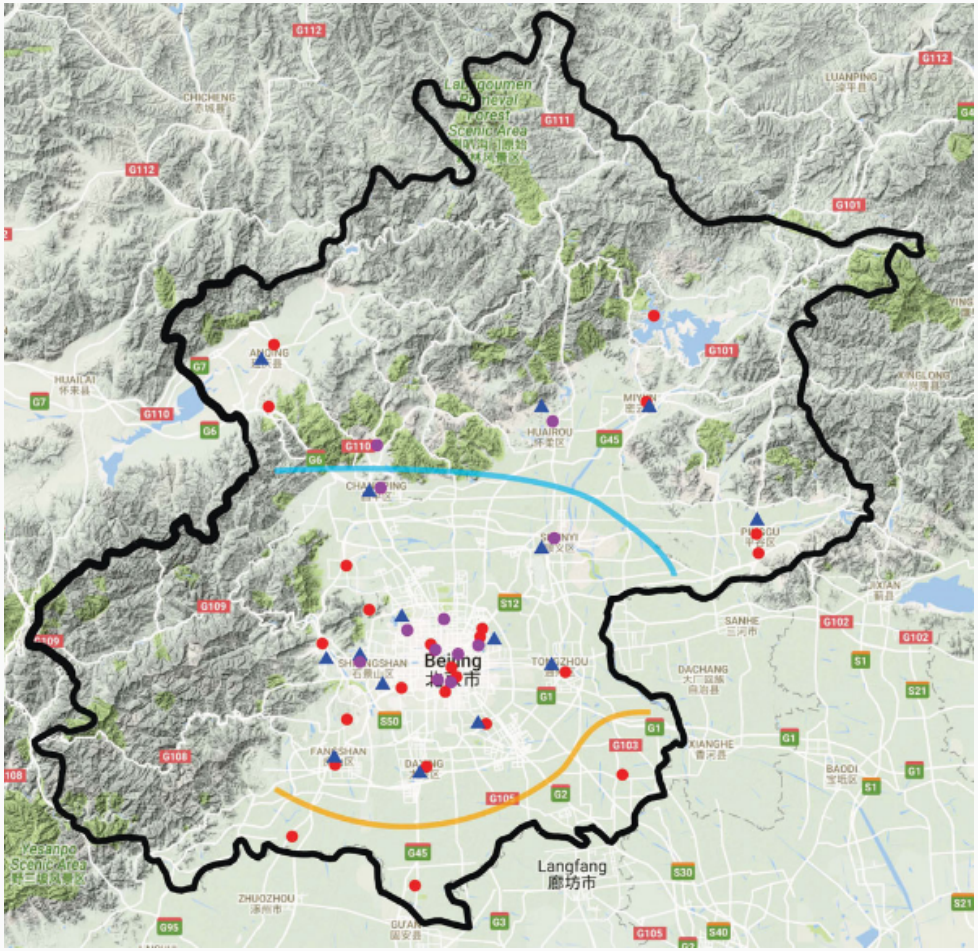
Here is a summary table for the preliminary model:

	mean	sd	hdi_3%	hdi_97%
alpha	3.454	0.009	3.436	3.471
beta	0.054	0.000	0.053	0.055
sigma	0.612	0.003	0.606	0.619

As expected:

- warmer days have more ozone
- specifically, change of 1 degree in high temperature associated with about a 5% increase in peak ozone concentration

Why multilevel model?



A map of Beijing. Purple dots are the 12 monitoring stations.

Why multilevel model?

We should expect some variation among geographic sites:

- Ozone source density may vary
- Topography influences local airflow patterns
- Sensor calibration differences

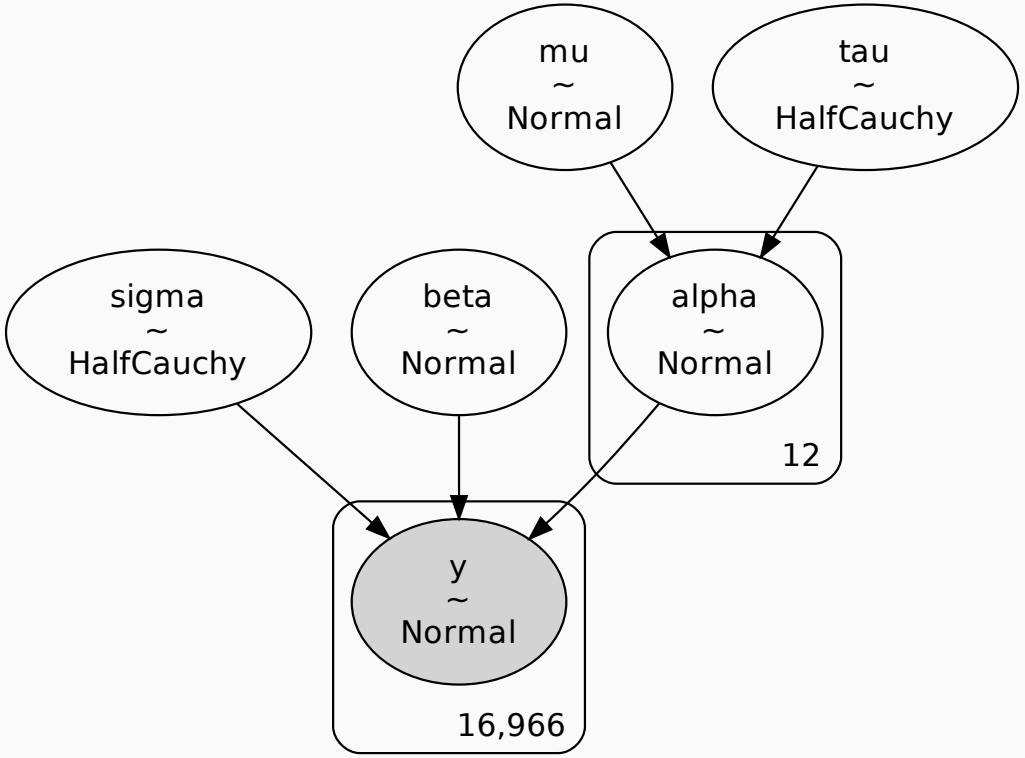
Simple extension of our model: allow varying intercepts across monitoring stations, to allow for each station to have a different "baseline" ozone level.

Varying-intercepts model

So let's extend the model:

$$
\begin{aligned}
\log O_3 &\sim \text{Normal}(\theta, \sigma) \\
\theta &= \alpha_j + \beta_T T_{\max} \\
\beta &\sim \text{Normal}(0, 1) \\
\alpha_j &\sim \text{Normal}(\mu, \tau) \\
\mu &\sim \text{Normal}(0, 3) \\
\tau &\sim \text{HalfCauchy}(1)
\end{aligned}
$$

Varying-intercepts model



Model specification

```
with pm.Model() as multilevel_model:
    # Hyperparameters
    mu = pm.Normal('mu', 0, 2)
    tau = pm.HalfCauchy('tau', 1)
    # Parameters
    sigma = pm.HalfCauchy('sigma', 1)
    alpha = pm.Normal('alpha', mu, tau, shape = 12)
    beta = pm.Normal('beta', 0, 0.1)
    # Model equation
    theta = alpha[dailies.dropna()['station_id']]
    + beta * dailies.dropna()['TEMP']
    # Likelihood
    y_ = pm.Normal('y', mu=theta, sigma = sigma,
        observed = dailies.dropna()['log_ozone'])
```

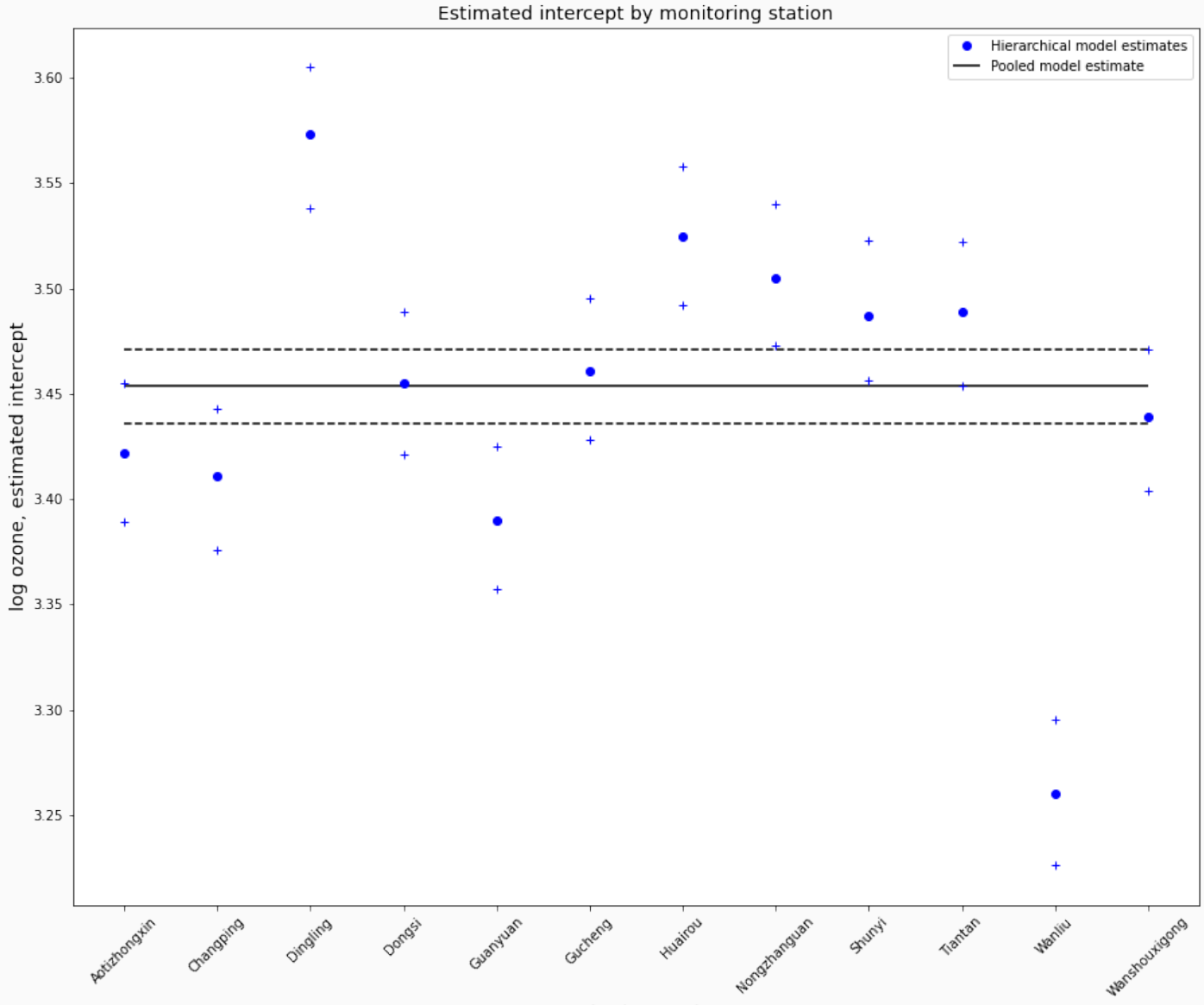
Model fitting

Results:

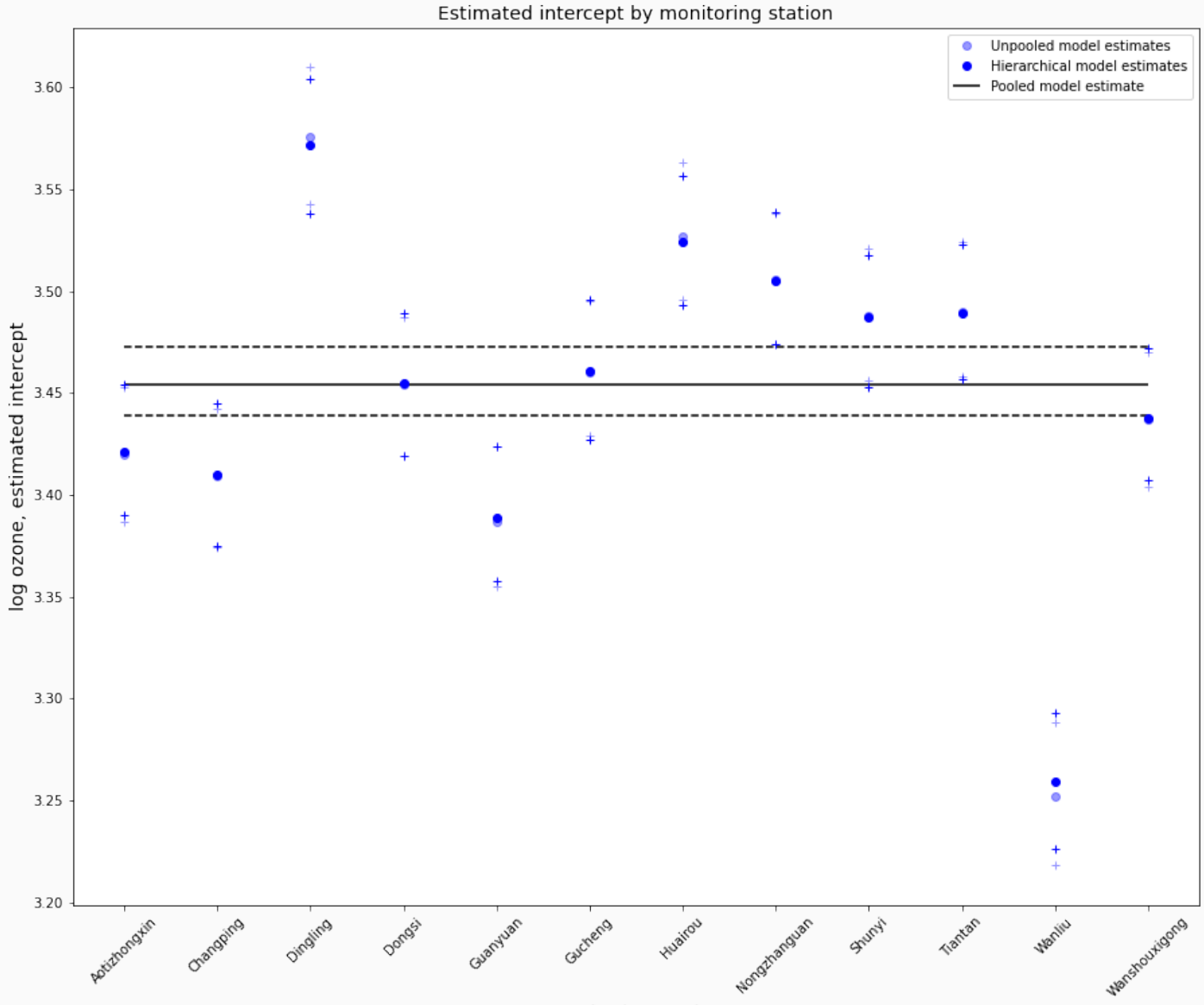
	mean	sd	hdi_3%	hdi_97%
mu	3.451	0.029	3.396	3.504
alpha[0]	3.422	0.018	3.389	3.455
alpha[1]	3.411	0.018	3.376	3.443
alpha[2]	3.573	0.018	3.538	3.605
alpha[3]	3.455	0.018	3.421	3.489
alpha[4]	3.390	0.018	3.357	3.425

- Hierarchical mean parameter similar to pooled intercept, but station intercepts vary

Model fitting



Model fitting



Model comparison

Comparing the models using PSIS-LOO:

	rank	loo	p_loo	d_loo	weight	se	dse	warning	loo_scale
multilevel	0	-15629.400655	17.039457	0.000000	0.969016	189.518973	0.00000	False	log
simple	1	-15757.707232	6.678447	128.306577	0.030984	190.474747	16.99114	False	log

- Multilevel model: better predictive score
- Also allows us to estimate which locations have elevated (Dingling) or depressed (Wanliu) baseline ozone

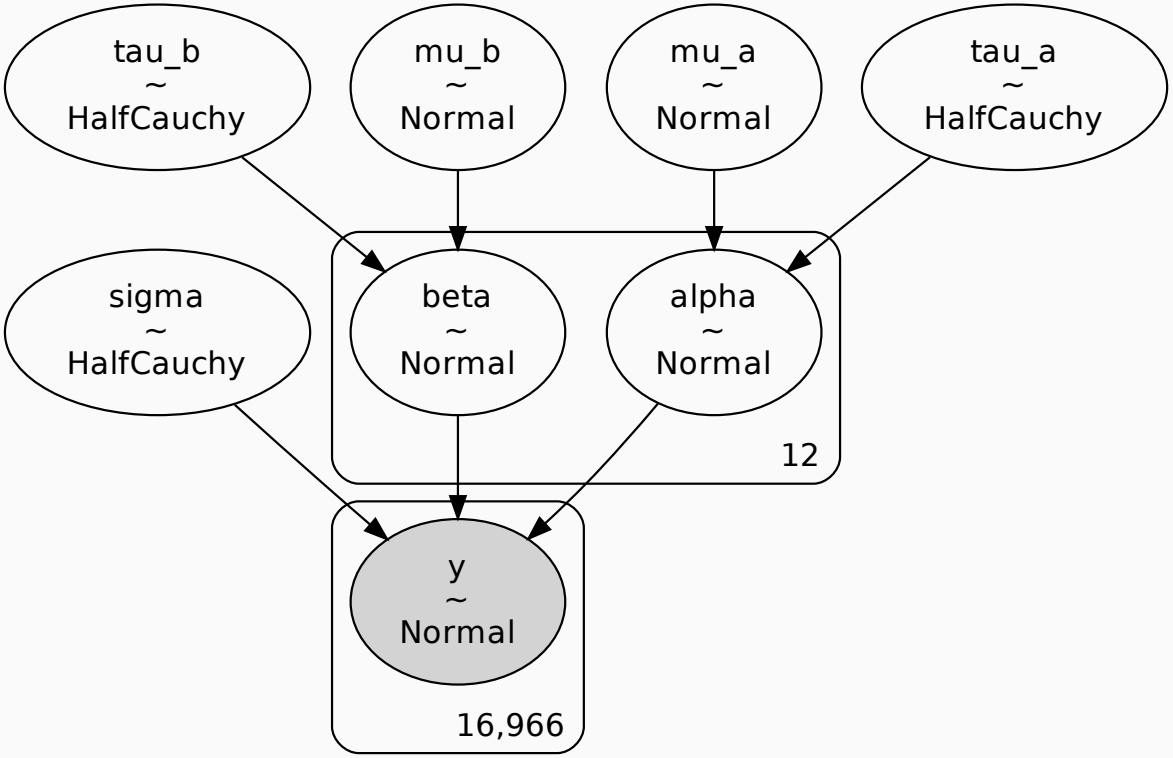
Varying slopes

Of course, once we have *varying intercepts*, it seems natural to also want *varying slopes*.

Extend the model again:

$$
\begin{aligned}
\log O_3 &\sim \text{Normal}(\mu, \sigma) \\
\mu &= \alpha_j + \beta_{T,j} T_{\max} \\
\beta_{T,j} &\sim \text{Normal}(\mu_\beta, \tau_\beta) \\
\alpha_j &\sim \text{Normal}(\mu_\alpha, \tau_\beta) \\
\mu_\alpha &\sim \text{Normal}(0, 2) \\
\tau_\alpha &\sim \text{HalfCauchy}(1) \\
\mu_\beta &\sim \text{Normal}(0, 0.1) \\
\tau_\beta &\sim \text{HalfCauchy}(1)
\end{aligned}
$$

Varying slopes



Model comparison

Finally, we add the varying-slopes model to the model comparison:

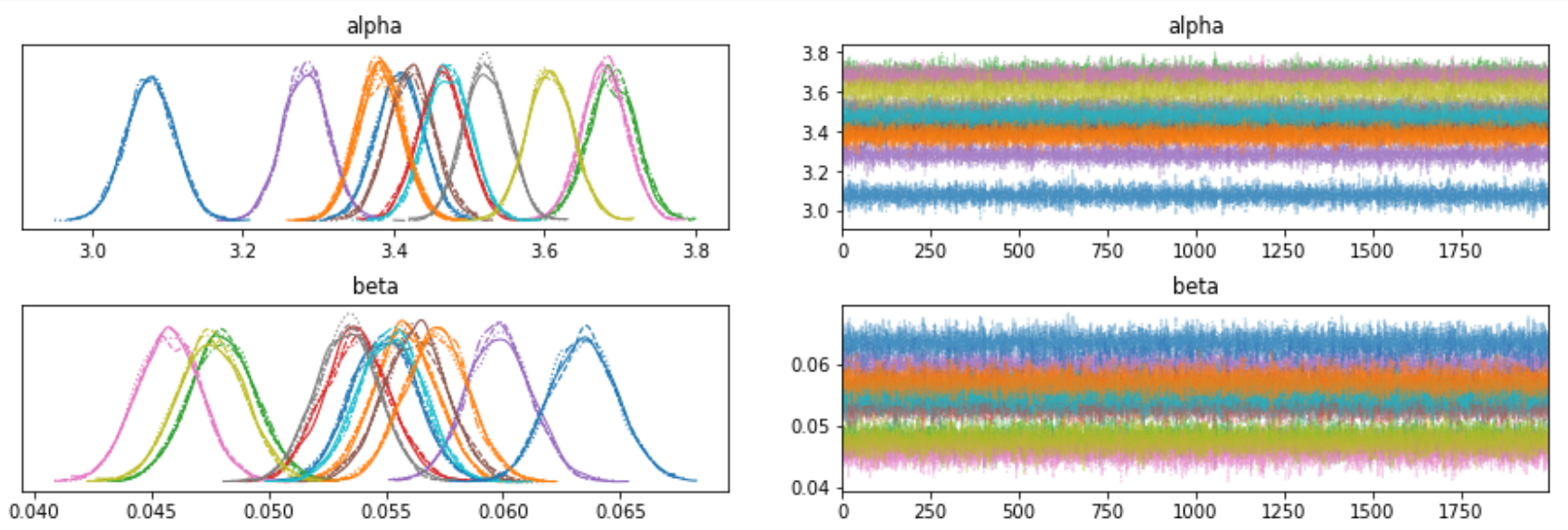
	rank	loo	p_loo	d_loo	weight	se	dse
varying slopes	0	-15557.545038	29.396342	0.000000	8.172186e-01	190.105625	0.000000
varying intercepts	1	-15629.400655	17.039457	71.855617	5.087248e-12	189.518973	13.373548
simple	2	-15757.707232	6.678447	200.162195	1.827814e-01	190.474747	26.581303

See a further improvement in model fit from varying slopes

Covarying parameters

Slopes and intercepts

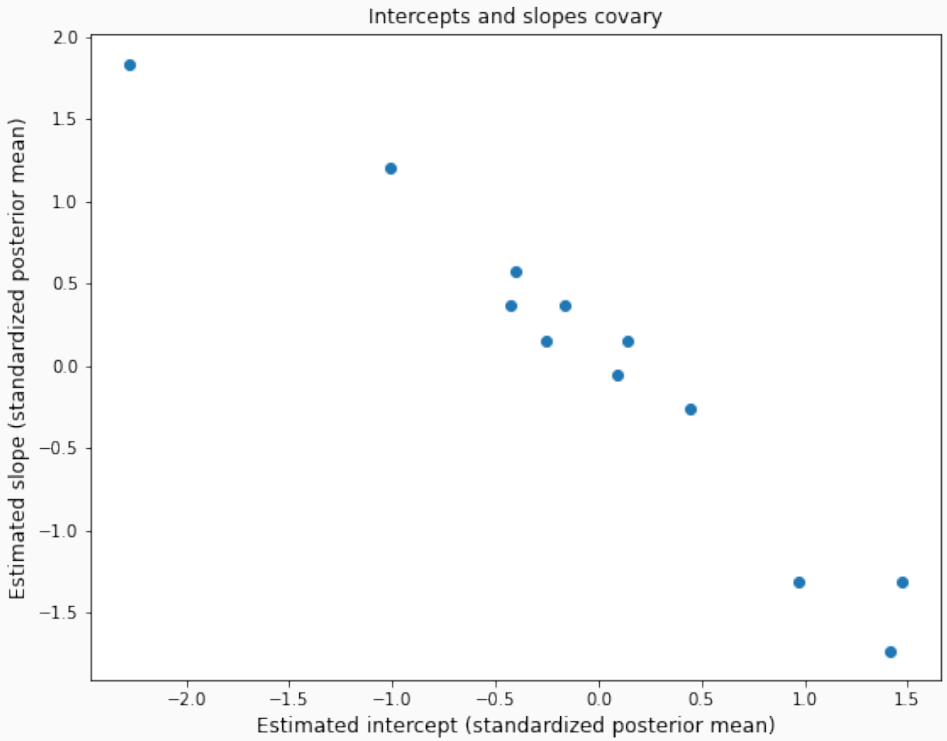
Here is part of the traceplot from the varying-slopes model:



Colors are the same. What do we see?

Covariance between parameters

- Intercepts vary across stations
- Slopes vary across stations
- Intercepts and slopes are correlated



A station with a high baseline ozone sees a smaller relative effect with temperature

Covariance between parameters

Right now:

- slopes and intercepts are *a priori* independent
- we can see the negative association in the posterior, but the model cannot learn that pattern
- if we add a new monitoring station, it restarts with the same independence assumption:
  - the model knows something about typical intercepts
  - the model knows something about typical slopes
  - the model knows nothing about the relationship between them

Covariance between parameters

In order to capture this effect, we need to explicitly include correlations in the model:

- Instead of thinking of 12 different intercepts and 12 different slopes, think of 12 intercept-slope pairs
- Right now, α_i, β_i are *a priori* normally distributed
- Replace the normal priors on α, β with multivariate normal

Multivariate normal distribution

Multivariate normal distribution:

- Generalizes normal distribution to produce vectors instead of scalars
- Specified by a vector of means and a covariance matrix
  - mean vector: contains mean of each component
  - covariance matrix: contains variance of each component, and covariance of each pair of components

Multivariate normal distribution

2×2 covariance matrix:

$$\Sigma = \begin{pmatrix} \sigma_a^2 & \sigma_a \sigma_b \rho_{ab} \\ \sigma_a \sigma_b \rho_{ab} & \sigma_b^2 \end{pmatrix}$$

- σ_a^2 – variance of a
- σ_b^2 – variance of b
- ρ_{ab} – correlation of a and b

Next time: how to pull these apart and set priors on them

Summary

Today:

- Hierarchical linear regression models
    - Varying intercepts
    - Varying slopes

Next week:

- Multivariate normal mechanics
- Priors for covariance matrices
- Gaussian processes