GLMS & STAN

BST228

Speaker notes

No notes on this slide.

RECAP

- Likelihood for linear regression.
- Gibbs sampler for regression coefficients β and precision τ.
- Strong posterior correlation for features with non-zero mean; funnels in $\beta - \tau$ pair plots.
- Improved posterior sampling using de-meaned features.

Speaker notes

No notes on this slide.

TODAY

- Heteroskedastic regression with unequal observation precision.
- Using Stan to draw posterior samples.
- Generalized linear models.

Speaker notes

No notes on this slide.

HETEROSKEDASTIC REGRESSION

We have the likelihood

$$\mathbf{y} \sim \text{Normal}\left(\mathbf{X}\beta, \boldsymbol{\tau}^{-1}\right),$$

where precision τ may depend on the observation, i.e.,

$$\boldsymbol{\tau} = \begin{pmatrix} \tau_1 & 0 & \dots \\ 0 & \tau_2 & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}.$$

Speaker notes

- Heteroskedastic regression allows precision to vary by observation, e.g., we may have collected data using different instruments with different precisions.
- Precision matrices can be generalized to include off-diagonal elements for correlated observations or the precision may depend on features $\mathbf{X}$ parametrically.
- These more general precision matrices are often best formulated in terms of the covariance $\boldsymbol{\tau}^{-1}$ because it is easier to think about priors.

EXAMPLE: INCUMBENCY

We have the following election data.

year	incumbent	democrat_share	...
1912	0	0.507	...
1912	-1	0.469	...
1912	1	0.518	...
1912	1	0.489	...
...	...	...	...

Speaker notes

- Data are from Bayesian Data Analysis and available here.
- `incumbent` : 0 if open, 1 if Democrat, -1 if Republican.

HETEROSKEDASTIC INCUMBENCY MODEL

$$y_i \sim \text{Normal}\left(\mathbf{x}_i^\intercal \boldsymbol{\beta}, \tau_{|\texttt{incumbent}|}^{-1}\right)$$

Speaker notes

- We use a regression model where the precision depends on if the incumbent seeks re-election.
- The outcome y_i is the Democrat vote share.
- The features $\mathbf{x}_i$ include intercept, previous election's Democrat vote share, signed incumbency indicator, interaction term.
- We de-mean all features to improve sampling efficiency and simplify interpretation of parameters.
- We could still obtain the posterior for this model either in closed form or using a custom Gibbs sampler. But we are spending most of our time writing samplers, not gaining insights from our data.

STAN

Stan is a state-of-the-art platform for statistical modeling and high-performance statistical computation.

Speaker notes

- This is how Stan describes itself.
- The main contribution of Stan (and other inference packages like numpyro) is to separate model building from inference.
- Stan is (largely) a declarative language, i.e., we write models just like we would on paper. This differs from R or Python which are procedural languages.

ANATOMY OF A STAN PROGRAM

```
data {
    ...
}

parameters {
    ...
}

model {
    ...
}
```

Speaker notes

- Each program has three core blocks: `data` declares the model *inputs*, not just the data; `parameters` declares the model parameters, including constraints; `model` contains sampling statements.

```
1 data {
2     int<lower=1> n, p;
3     matrix [n, p] X;
4     vector [n] y;
5     vector<lower=0, upper=1> [n] incumbent;
6 }
```

Speaker notes

- Line #2 declares number of observations `n` and features `p`.
- #3-4 declare features and responses.
- #5 declares incumbency indicator. This is technically redundant but convenient for declaring the model.

```
1 parameters {
2     vector [p] coef;
3     real<lower=0> scale0, scale1;
4 }
```

Speaker notes

- Line #2 declares a vector of regression coefficients matching the number of features `p`.
- #3 declares *two* observation noise scales: `scale0` if there is no incumbent, `scale` if there is an incumbent.
- We previously used precision instead of scales. That choice was motivated by convenience for algebraic manipulation. In Stan, we use scales as they are more intuitive.

```
1  model {
2      scale0 ~ cauchy(0, 1);
3      scale1 ~ cauchy(0, 1);
4      coef ~ normal(0, 2);
5      y ~ normal(
6          X * coef,
7          (1 - incumbent) * scale0
8          + incumbent * scale1
9      );
10 }
```

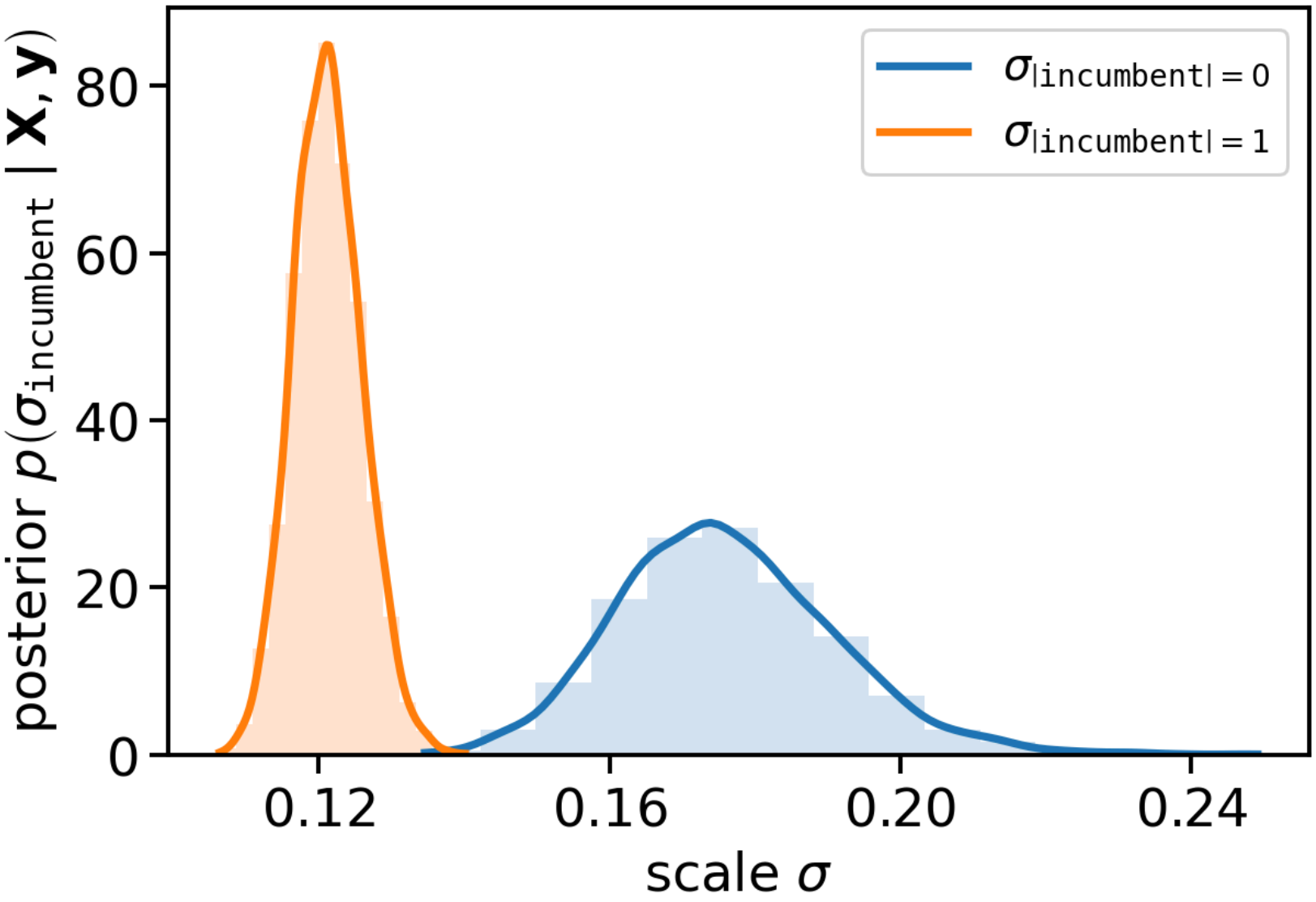
Speaker notes

- Lines #2-3 place a prior on scales. We are no longer restricted to conjugate priors and use a Cauchy prior which mildly regularizes the scales σ.
- #4 declares a mildly informative for regression coefficients. All features are of order 1, and responses vary on the order of 0.1. We expect coefficients to be relatively small.
- #6 evaluates the predictor (`*` represents matrix multiplication in Stan by default; use `.*` for elementwise multiplication).
- #7-#8 evaluates the scale depending on the `incumbent` indicator.

```
Checking sampler transitions treedepth.
Treedepth satisfactory for all transitions.

Checking sampler transitions for divergences.
No divergent transitions found.

Checking E-BFMI - sampler transitions HMC potential
  energy.
E-BFMI satisfactory.

Effective sample size satisfactory.

Split R-hat values satisfactory all parameters.

Processing complete, no problems detected.
```

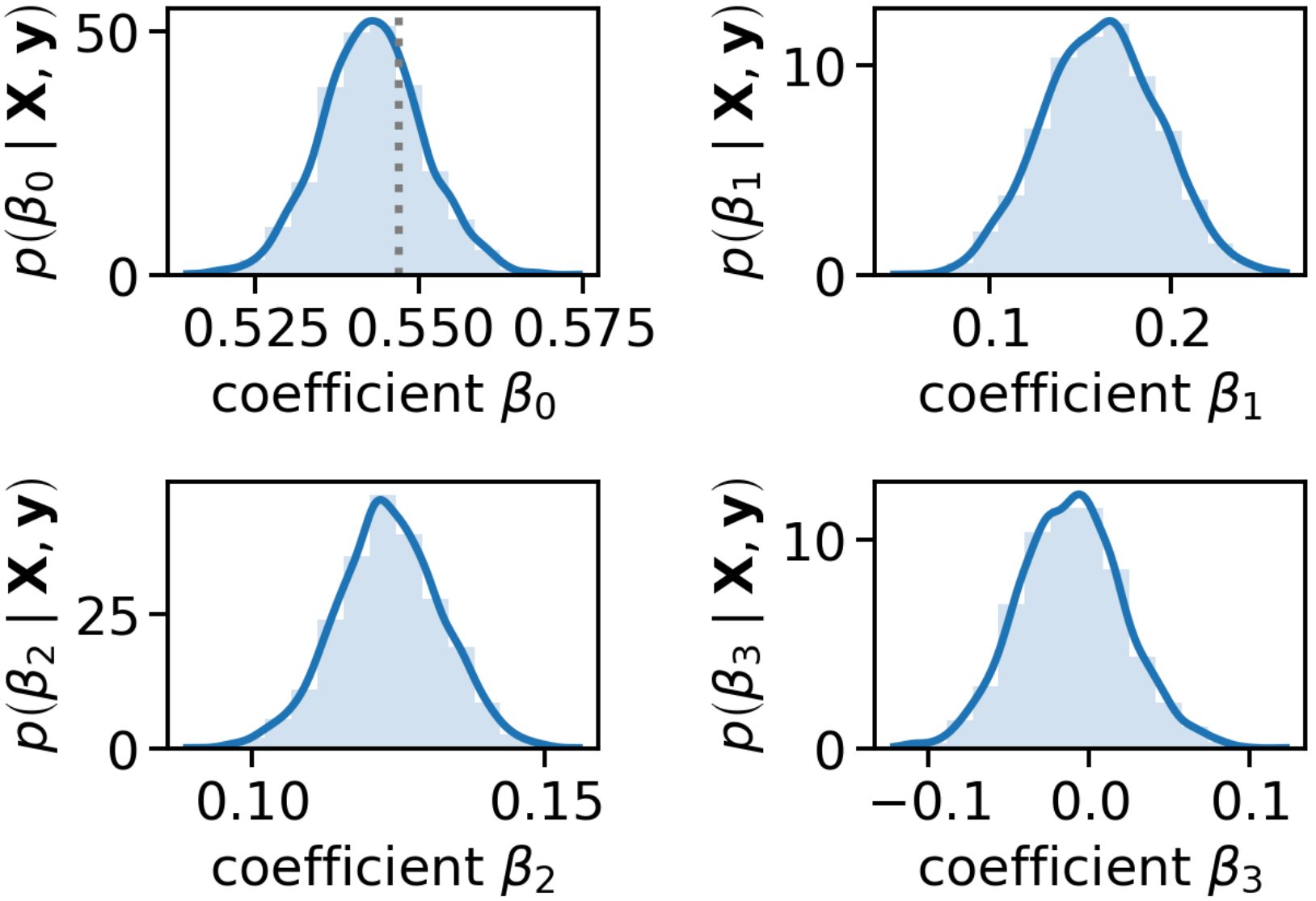
Speaker notes

- With Stan, we can simply hit "run" and get posterior samples (see `stan-intro.R` on Canvas for an example).
- But we still need to inspect posterior samples to validate the inference. There is on inference algorithm that can be trusted blindly.
- Lines #1-9 report diagnostics specific to Stan's sampling algorithm.
- #11 & 13 report diagnostics we are already familiar with.
- See arXiv:1903.08008 for details on the *split* R-hat diagnostic.
- These basic diagnostics look satisfactory, and we can further inspect the posterior.



Speaker notes

- The variability for elections where the incumbent seeks re-election is *much* smaller than if there is no incumbent. This is consistent with what we might expect: People know what they're getting if they vote for the incumbent (or against them).



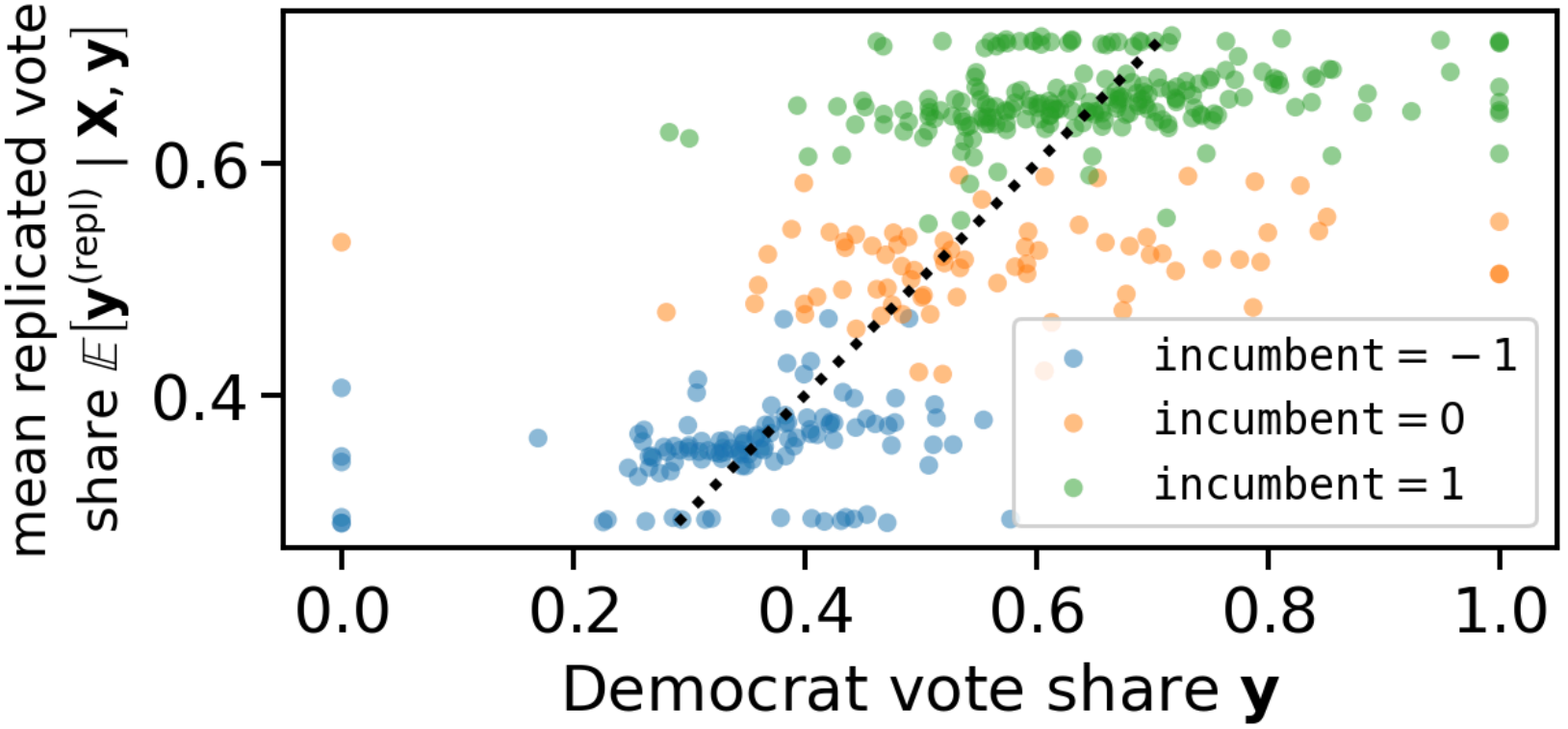
Speaker notes

- The sample mean of Democrat vote share (dotted gray line in first panel) matches the intercept. This inference agrees with the results of the 1992 House election.
- Posterior on β_1 implies a 10% point increase in Democrat vote share from previous election translates to ~1.5% point increase for this election.
- Posterior on β_2 implies a ~1.2% point incumbency advantage.
- No obvious interaction from the posterior on β_3.
- Remember that interpreting coefficients, especially their uncertainties, directly can be misleading.

```
generated quantities {
    array [n] real y_repl = normal_rng(
        X * coef,
        (1 - incumbent) * scale0
        + incumbent * scale1
    );
}
```

Speaker notes

- We can add another block to the Stan program: `generated quantities`. This block can be used to generate replications of the data, posterior predictions, or any other quantity of interest.
- Here, we replicate the data to serve as a sanity check. If we cannot replicate the data, there is something wrong with the model. We will further explore data replication in future lectures on model comparison.



Speaker notes

- We are not great at replicating the data, but we capture the dominant factors: incumbency and slight additional correlation between data $\mathbf{y}$ and posterior mean replications $\mathbb{E}\left[\mathbf{y}^{(\text{repl})} \mid \mathbf{X}, \mathbf{y}\right]$ due to previous vote share.
- However, if we evaluate the minimum and maximum of replications, we find $\min \mathbf{y}^{(\text{repl})} = -0.34$ and $\max \mathbf{y}^{(\text{repl})} = 1.39$. This is obviously wrong because vote shares must belong to the unit interval.
- We have used the wrong model in the sense that the support of the likelihood is not the same as the support of the data.

GENERALIZED LINEAR MODELS

$$\mathbf{y} \sim \text{Appropriate Distribution} \left(g^{-1} \left(\mathbf{X}\boldsymbol{\beta} \right), \ldots \right)$$

Speaker notes

- Generalized linear models (GLMs) use a distribution suitable for the data at hand (such as a beta distribution for vote shares) and a link function g such that $\mathbb{E}\left[\mathbf{y} \mid \mathbf{X}, \boldsymbol{\beta}\right] = g^{-1}\left(\mathbf{X}\boldsymbol{\beta}\right)$. The use of g^{-1} instead of g is convention and not material. The linear predictor passed to the link function is often denoted $\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta}$.
- Here, $\ldots$ denotes other parameters of the likelihood, such as overdispersion parameters.
- GLMs cannot in general be treated analytically, but we fortunately have access to a flexible probabilistic programming language: Stan.

GLM FOR VOTE SHARE DATA

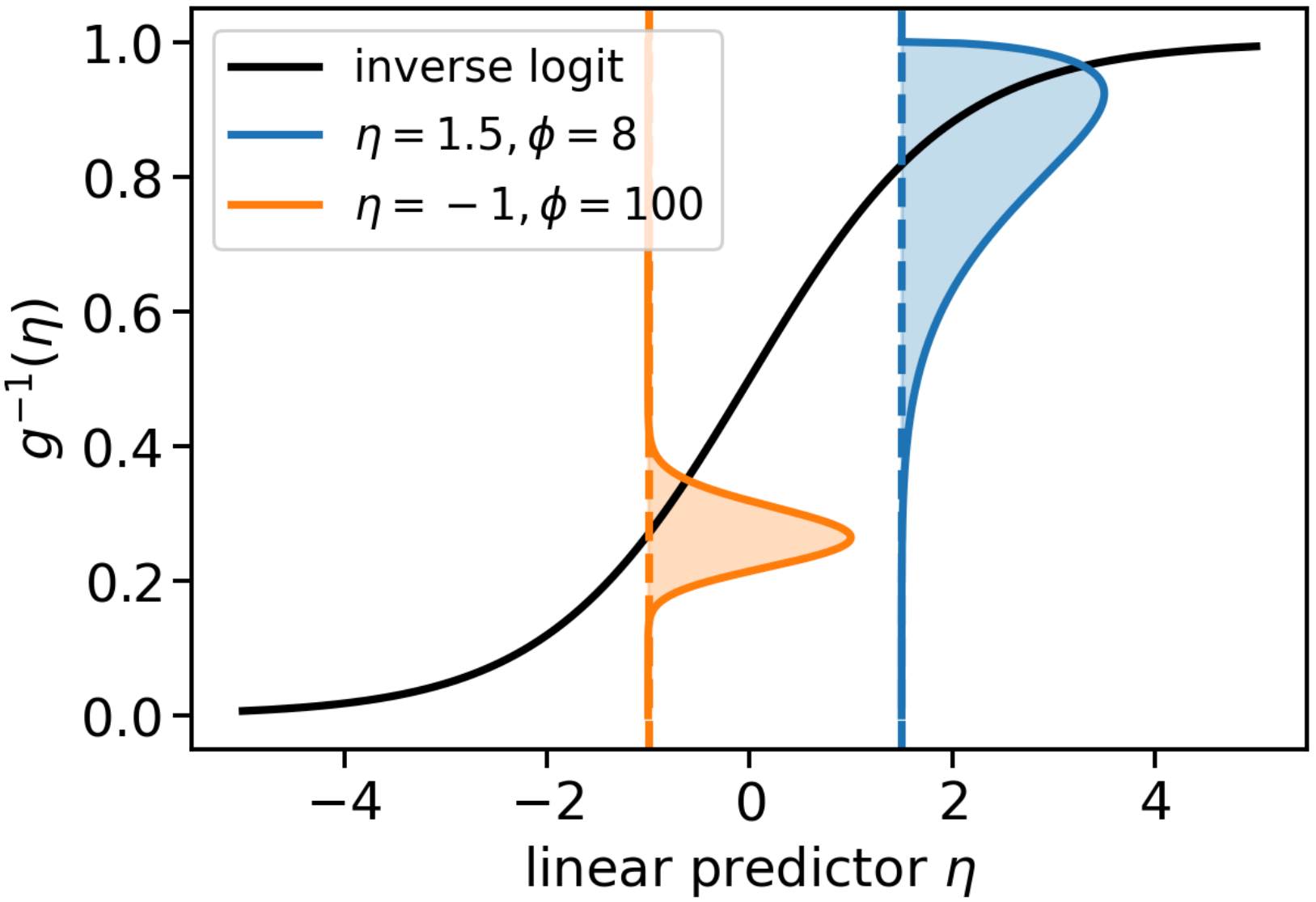
$$\mathbf{y} \sim \text{Beta}\left(\phi_{|\texttt{incumbent}|} g^{-1}\left(\boldsymbol{\eta}\right), \phi_{|\texttt{incumbent}|} g^{-1}\left(-\boldsymbol{\eta}\right)\right)$$

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta}$$

$$g^{-1}\left(\boldsymbol{\eta}\right) = \frac{1}{1 + \exp\left(-\boldsymbol{\eta}\right)}$$

Speaker notes

- We use a beta likelihood because the data are proportions, $\boldsymbol{\beta}$ is the linear predictor, and the inverse link function g^{-1} is the logistic sigmoid (equivalently, the link function g evaluates the log odds).
- ϕ is an overall concentration parameter fulfilling the same role as σ in the previous example with normal likelihood.
- The above model has the desired GLM property $\mathbb{E}\left[\mathbf{y}\right] = g^{-1}\left(\boldsymbol{\eta}\right)$. Verify in your own time, using the mean of the beta distribution and noting that $g^{-1}\left(-\boldsymbol{\eta}\right) = 1 - g^{-1}\left(\boldsymbol{\eta}\right)$.



Speaker notes

- Black line is the the inverse logit function g^{-1}, also known as expit.
- Two colored lines show the density of an observation for given linear predictor η and different concentrations ϕ. Larger ϕ results in more precise likelihoods (just like the precision of a normal likelihood).
- These likelihoods have the right support for vote shares: the unit interval.

```
1 data {
2     int<lower=1> n, p;
3     matrix [n, p] X;
4     vector<lower=0, upper=1> [n] y;
5     vector<lower=0, upper=1> [n] incumbent;
6 }
```

Speaker notes

- Data are exactly the same, except we added a constraint `<lower=0, upper=1>` to the data in line #4.

```
1 parameters {
2     vector [p] coef;
3     real<lower=0> phi0, phi1;
4 }
```

Speaker notes

- Coefficients in line #2 remain unchanged.
- We replaced the observation noise scales in #3 with concentration parameters for the beta distribution.

```
1 transformed parameters {
2     vector [n] eta = X * coef;
3     vector [n] phi = (1 - incumbent) * phi0
4         + incumbent * phi1;
5 }
```

Speaker notes

- We added a new block `transformed parameters` which declares, unsurprisingly, transformed parameters.
- We use the block to declare the predictor `eta` in line #2 and the convex combination of precisions for races with and without incumbents seeking re-election (`phi1` and `phi0` , respectively). This is not strictly required but reduces the amount of typing later in the model.

```
1 model {
2     phi0 ~ cauchy(0, 100);
3     phi1 ~ cauchy(0, 100);
4     coef ~ normal(0, 2);
5     y ~ beta(phi .* inv_logit(eta),
6              phi .* inv_logit(-eta));
7 }
```

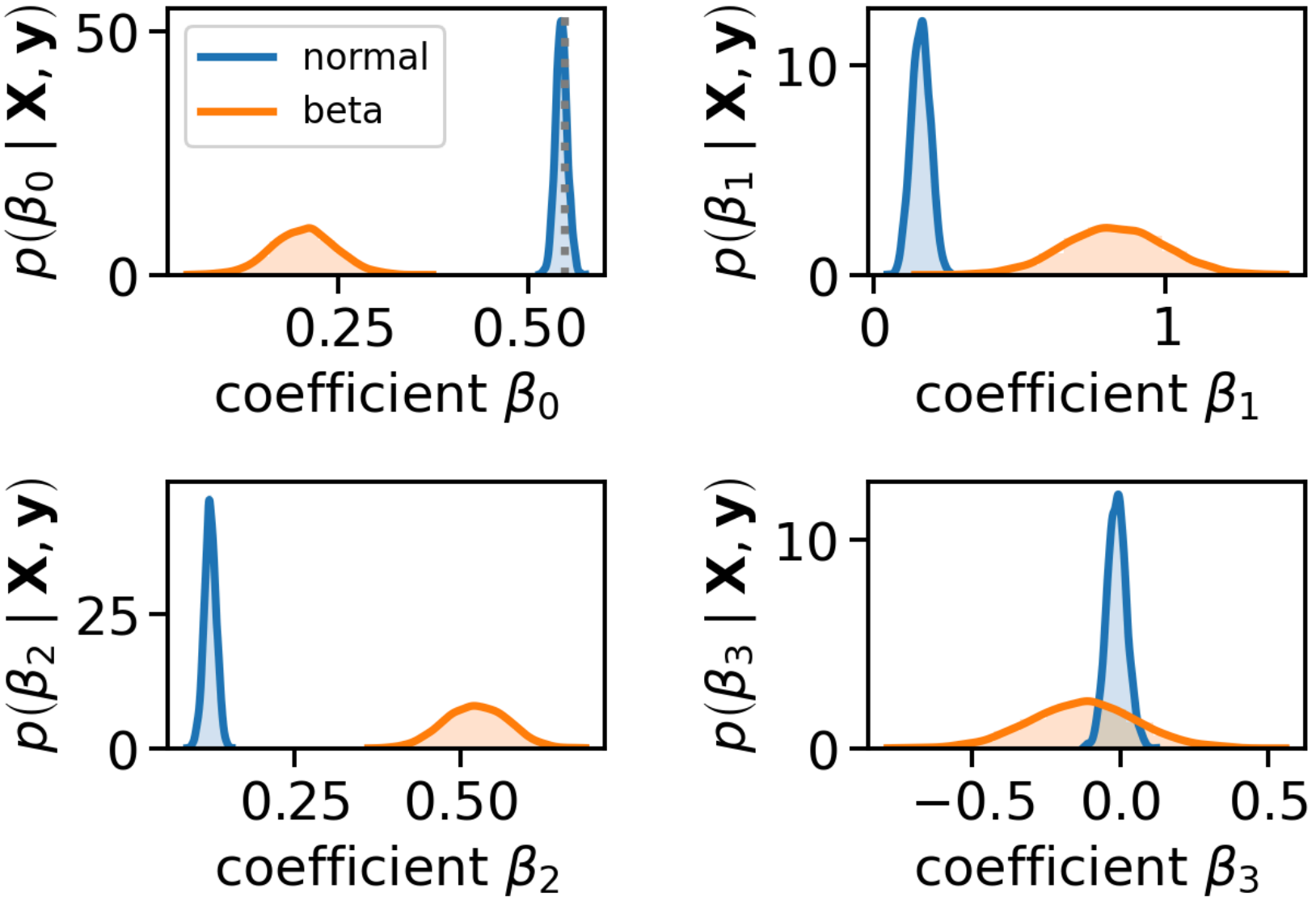
Speaker notes

- Lines #2-3 declare priors for the concentration parameters which are not unlikely precisions we encountered for normal models. The variance of a beta distribution is on the order of `1 / phi`. We need a relatively broad prior to reproduce the small variances in the data.
- The coefficient prior in #4 remains unchanged.
- #5-6 declare the likelihood.

```stan
generated quantities {
    array [n] real<lower=0, upper=1> y_repl =
    beta_rng(
        phi .* inv_logit(eta),
        phi .* inv_logit(-eta)
    );
}
```

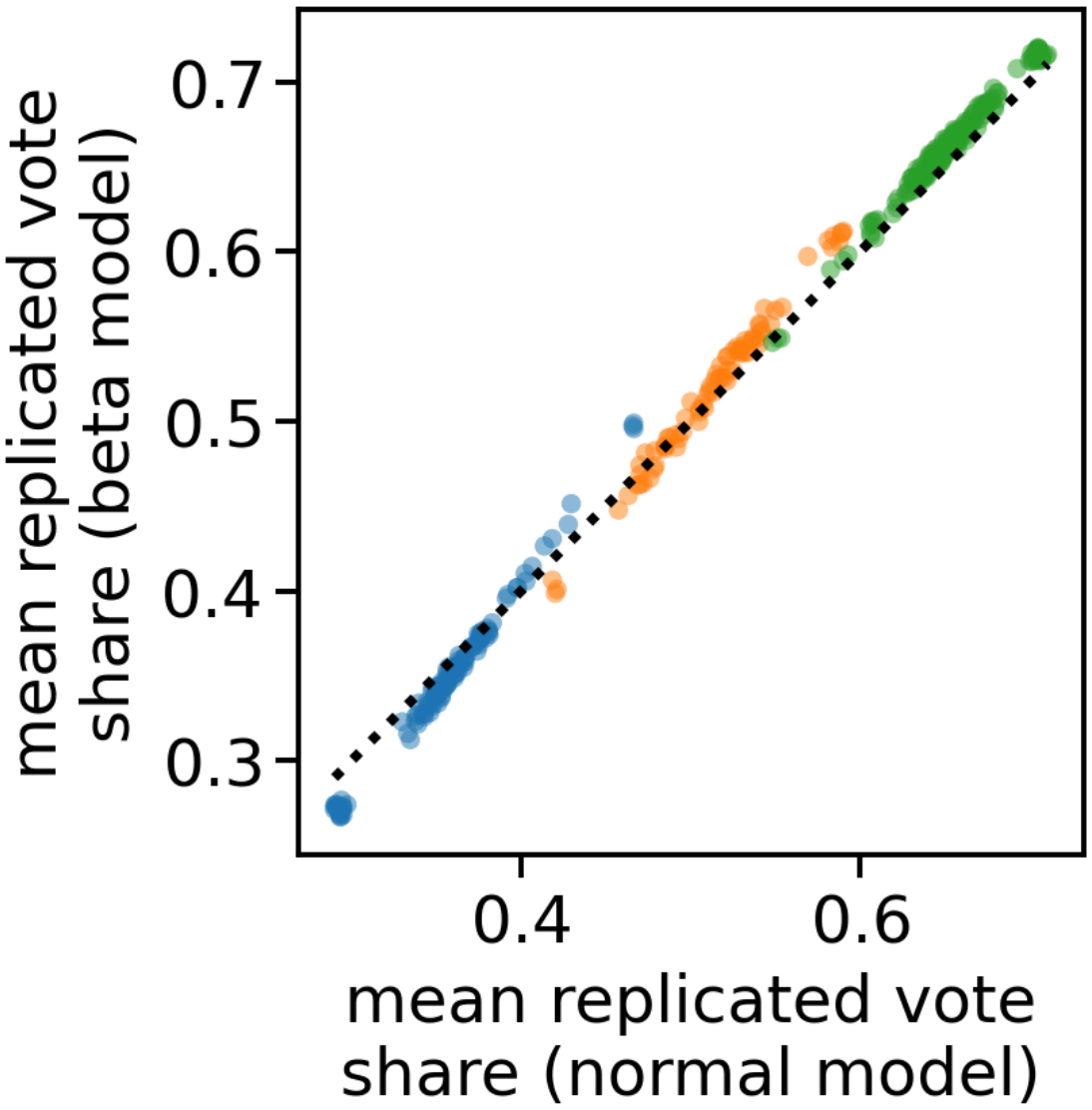
Speaker notes

- Generated quantities are updated to use the right distribution for replication.
- We add an constraint to `y_repl` for explicitness although it is not required for the model to work.



Speaker notes

- Comparing coefficients between the two models is not trivial because they are on different scales. The normal model operates on the percentage point scale, the beta GLM operates on the log odds scale.
- Comparing the two models using counterfactual predictions would be more interpretable here. For example, how would the national vote share change this election if the Democrat vote share in the previous election was 5% points higher.
- Note: I clipped the data $\mathbf{y}$ below at 0.01 and above at 0.99 because, for $\phi > 2$, the likelihood at $y = 0$ or $y = 1$ is zero, and the model fails to fit. What other model could we use? Hint: What are the raw data used to evaluate the vote share?



Speaker notes

- We scatter the mean replicated vote share for each district for the two models. As we might expect, they yield very similar predictions because they are fit to the same data and the normal model is not a terrible approximation.

RECAP

- Heteroskedastic regression
- Posterior sampling with Stan
- Generalized linear models

Speaker notes

No notes on this slide.

HOMEWORK HINT

Sometimes link functions are equivalent to latent variable models that can be fit using Gibbs samplers. See Hoff section 12.1.1 and BDA3 section 16.2 for a discussion.

Speaker notes

No notes on this slide.

NEXT

- Hierarchical models

Speaker notes

No notes on this slide.