HIERARCHICAL GAUSSIAN PROCESSES IN STAN

http://mc-stan.org

WHO AM I?

- Rob Trangucci
- Stan math library contributor
- Collaborator with Michael Betancourt, Andrew Gelman, and Aki Vehtari
- Funded, in part, by YouGov (Thank you, YouGov!)

KEY TAKEAWAYS FOR TODAY

- What Stan does and why you should use it
- Gaussian processes and when they're useful
- The importance of thinking about the scale of your hyper parameters
- Hierarchical Gaussian processes
- Spatio-temporal GPs in Stan

WHAT IS STAN?

STAN

1. Language

2. Inference algorithms

3. Interfaces

STAN LANGUAGE

- Statistical model specification language
  - specify: data, parameters, joint probability distribution
- Domain specific language

  not C++, R, Python, BUGS, JAGS…

INFERENCE ALGORITHMS

- Bayesian inference; Markov Chain Monte Carlo (MCMC)
  - **No-U-Turn Sampler (NUTS)**
  - Hamiltonian Monte Carlo (HMC)
- Approximate Bayesian inference
  - Automatic Differentiation Variational Inference (ADVI)
- Optimization
  - Limited memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS)

INTERFACES

- Tight integration
  - RStan
  - PyStan
- Process level integration
  - CmdStan
  - stan.jl, StataStan, MatlabStan, MathematicaStan

WHAT IS STAN FOR?

WHAT IS STAN FOR?

- You have data
- You specify a statistical model
  Stan program
- Want inference over the statistical model
  posterior distribution, approximate inference, optimization

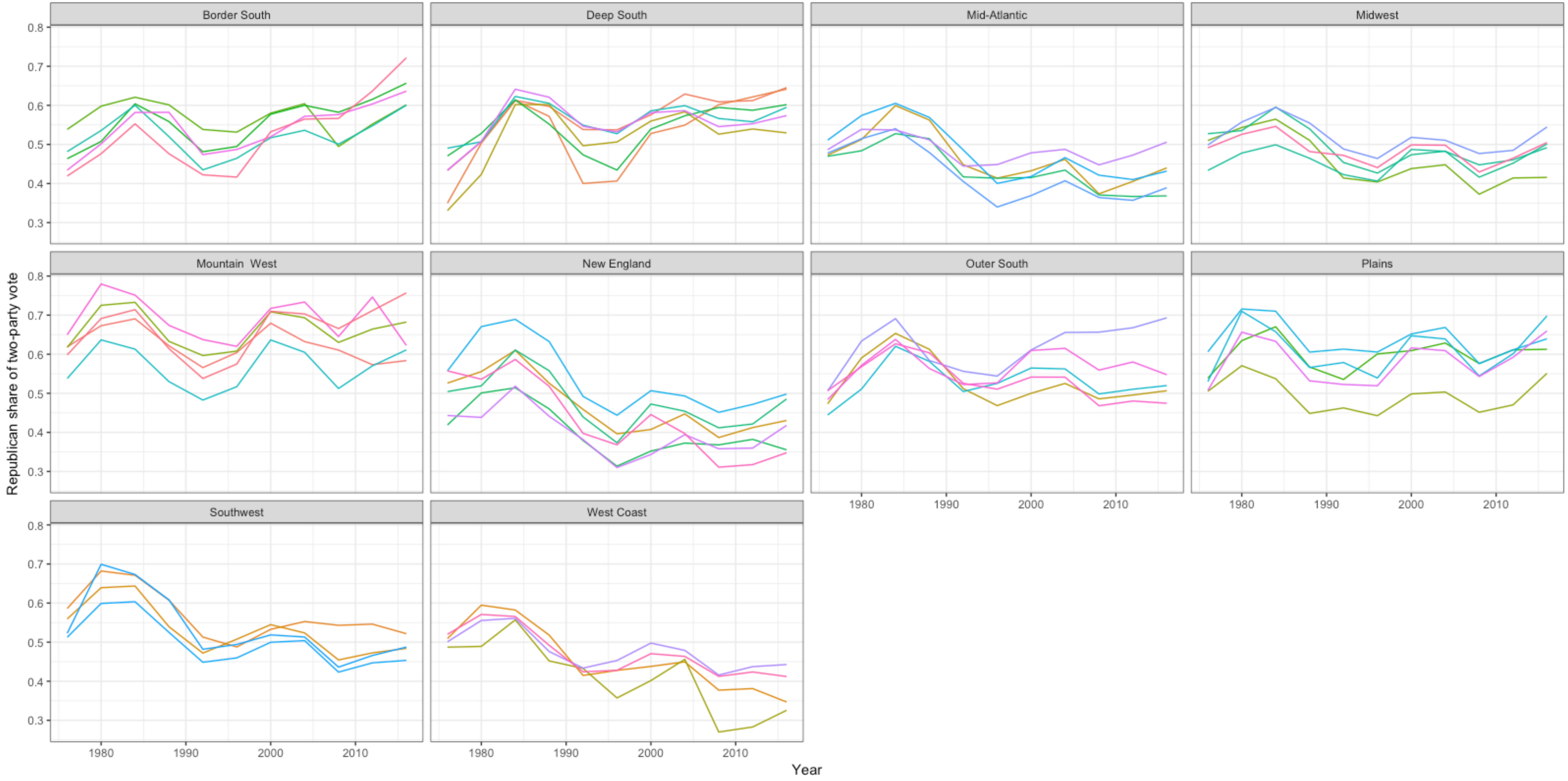
WHAT IS STAN FOR?

- You have data
- You specify a statistical model
  Stan program
- Want inference over the statistical model
  posterior distribution, approximate inference, optimization
- **You want to change the statistical model**
  new data, better model, some new information

FORECASTING US PRESIDENTIAL ELECTIONS 2020 - 2028

- Example from Andrew Gelman's blog just prior to the election:
  - Read more here: http://andrewgelman.com/2016/11/04/31935/
- You're interested in predicting what might happen in the 2020, 2024, and 2028 US presidential elections
- You have Republican share of the two party vote by year from 1976 through 2016 by state
- Aki Vehtari and Dan Simpson helped Andrew put the model together

FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028



BAYESIAN INFERENCE

$$p\left(\theta|\mathbf{D}\right)$$

STAN WORKFLOW

$$p\left(\theta \middle| \mathbf{D}\right)$$

$$p\left(\theta \middle| \mathbf{D}\right)$$

$$p\left(\theta \middle| \mathbf{D}\right)$$

$$p\left(\theta|\mathbf{D}\right)$$

1. Define probability model

$$y_i$$

$$i \in \{1, \ldots, N\}$$

```
data {
  int<lower=1> N;
  int<lower=0,upper=1> y[N];
}
```

1. Define probability model

$$y_i$$

$$i \in \{1, \ldots, N\}$$

```
data {
  int<lower=1> N;
  int<lower=0,upper=1> y[N];
}
parameters {
  real<lower=0,upper=1> theta;
}
```

1. Define probability model

$$\theta \sim \mathrm{Beta}(4, 4)$$

$$y_i \sim \mathrm{Bernoulli}(\theta)$$

$$i \in \{1, \ldots, N\}$$

```
data {
  int<lower=1> N;
  int<lower=0,upper=1> y[N];
}
parameters {
  real<lower=0,upper=1> theta;
}
model {
  theta ~ beta(4,4);
  y ~ bernoulli(theta);
}
```

STEPS

1. Define probability model

$$f(\theta|D) : (0,1) \to \mathbb{R}$$

$$f(\theta|D)\Big|_{\theta=v}$$

$$\frac{\partial}{\partial\theta} f(\theta|D)\Big|_{\theta=v}$$

```
data {
  int<lower=1> N;
  int<lower=0,upper=1> y[N];
}
parameters {
  real<lower=0,upper=1> theta;
}
model {
  theta ~ beta(4,4);
  y ~ bernoulli(theta);
}
```

1. Define probability model

$$\mathrm{f}(\theta|D) : (0, 1) \to \mathbb{R}$$

$$(4-1)\log\theta + (4-1)\log(1-\theta)$$

$$\sum_{i=1}^{N} y_i \log(\theta) + (1-y_i)\log(1-\theta)$$

```
data {
  int<lower=1> N;
  int<lower=0,upper=1> y[N];
}
parameters {
  real<lower=0,upper=1> theta;
}
model {
  theta ~ beta(4,4);
  y ~ bernoulli(theta);
}
```

GAUSSIAN PROCESSES

$$y_i \in \mathbb{R}, x_i \in \mathbb{R}^M \; i \in \{1, 2, \ldots\}$$

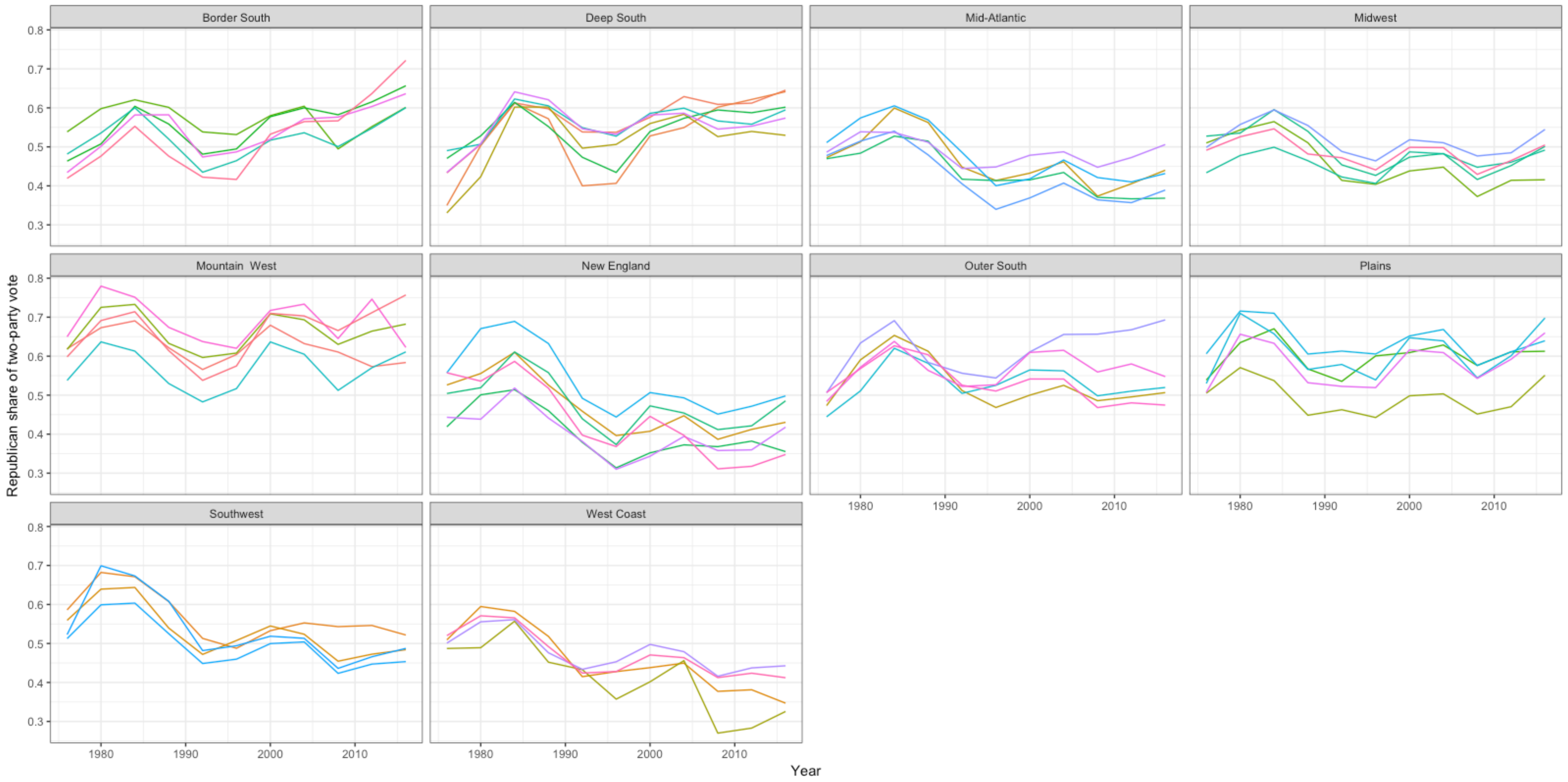
$$y_i \sim \mathrm{Normal}\left(f(x_i), \delta\right) \forall i$$

$$y_i \in \mathbb{R}, x_i \in \mathbb{R}^M \; i \in \{1, 2, \ldots\}$$

$$y_i \sim \mathrm{Normal}\left(f(x_i), \delta\right) \forall i$$

$$f(x) = x^T \beta$$

FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028



$$y_i \in \mathbb{R}, x_i \in \mathbb{R}^M \; i \in \{1, 2, \ldots\}$$

$$y_i \sim \mathrm{Normal}\left(f(x_i), \delta\right) \forall i$$

$$f(x) = \quad ?$$

$$y_i \in \mathbb{R}, x_i \in \mathbb{R}^M \; i \in \{1, 2, \ldots\}$$

$$y_i \sim \mathrm{Normal}\left(f(x_i), \delta\right) \forall i$$

$f(x)$ **CONTINUOUS**

$$y_i \in \mathbb{R}, x_i \in \mathbb{R}^M \; i \in \{1, 2, \ldots\}$$

$$y_i \sim \mathrm{Normal}\left(f(x_i), \delta\right) \forall i$$

$$f(x)$$

CONTINUOUS
SLOWLY VARYING IN X

$$y_i \in \mathbb{R}, x_i \in \mathbb{R}^M \; i \in \{1, 2, \ldots\}$$

$$y_i \sim \mathrm{Normal}\left(f(x_i), \delta\right) \forall i$$

$$f(x)$$

CONTINUOUS
SLOWLY VARYING IN X
REASONABLE BOUNDS

$$y_i \in \mathbb{R}, x_i \in \mathbb{R}^M \; i \in \{1, 2, \ldots\}$$

$$y_i \sim \mathrm{Normal}\left(f(x_i), \delta\right) \forall i$$

$$f(x) \sim \mathrm{GP}(0, K_\theta)$$

FINITE DIMENSIONAL DRAWS FROM A GP

$$y,\ f \in \mathbb{R}^N, X \in \mathbb{R}^{N \times M}$$

$$f \sim \mathrm{MultiNormal}\left(0, K_\theta(X)\right)$$

$$y_i \sim \mathrm{Normal}\left(f_i, \sigma\right) \forall i \in \{1, \ldots, N\}$$

$$y,\ f \in \mathbb{R}^T, \mathbf{T} = \{1, \ldots, T\}$$

$$f \sim \mathrm{MultiNormal}\left(0, K_\theta\left(\mathbf{T}\right)\right)$$

$$y_t \sim \mathrm{Normal}\left(f_t, \sigma\right) \forall\, t \in \{1, \ldots, T\}$$

$$K_\theta(X) = \mathrm{cov}(f)$$

COVARIANCE FUNCTION

$$K_\theta(X)_{i,j} = \operatorname{cov}\left(f(x_i), f(x_j)\right)$$

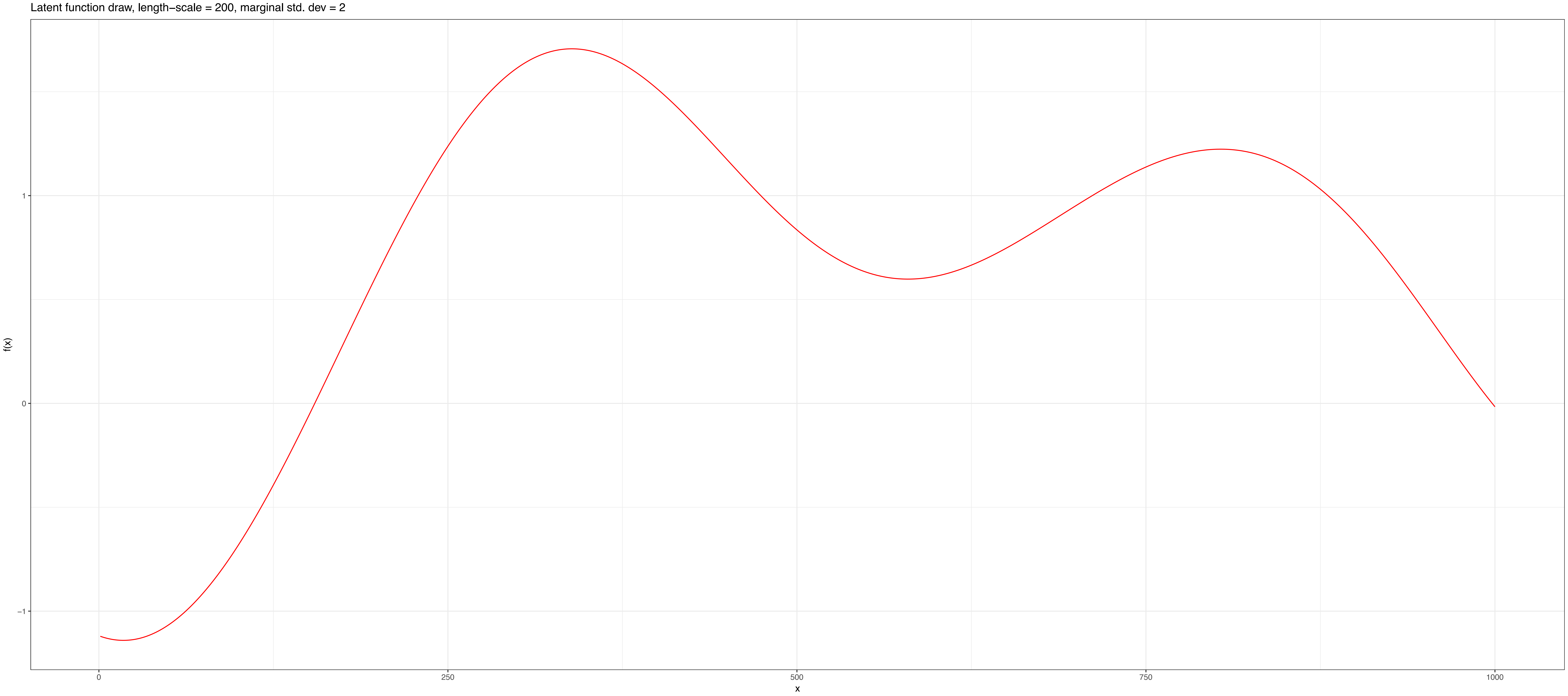
$$
\begin{aligned}
K_\theta(X)_{i,j} &= \operatorname{cov}\left(f(x_i), f(x_j)\right) \\
&= k\left(x_i, x_j \middle| \theta\right)
\end{aligned}
$$

$$
\begin{aligned}
K_\theta(X)_{i,j} &= \operatorname{cov}\left(f(x_i), f(x_j)\right) \\
&= k\left(x_i, x_j \middle| \theta\right) \\
&= \alpha^2 \exp\left(-\frac{1}{2\ell^2}(x_i - x_j)^2\right)
\end{aligned}
$$

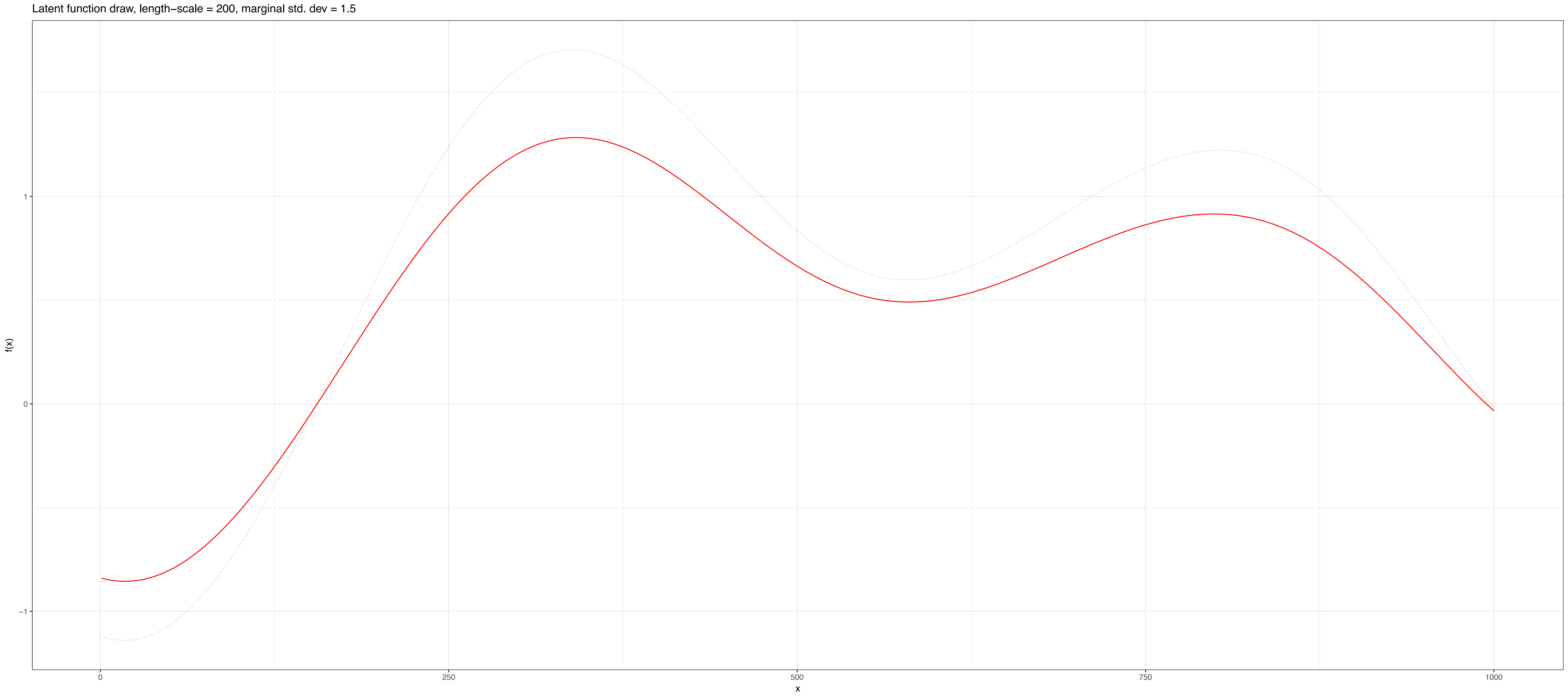
$$
\begin{aligned}
K_\theta(X)_{i,j} &= \operatorname{cov}\left(f(x_i), f(x_j)\right) \\
&= k\left(x_i, x_j \middle| \theta\right) \\
&= \alpha^2 \exp\left(-\frac{1}{2\ell^2}(x_i - x_j)^2\right)
\end{aligned}
$$

α **MARGINAL STANDARD DEVIATION**

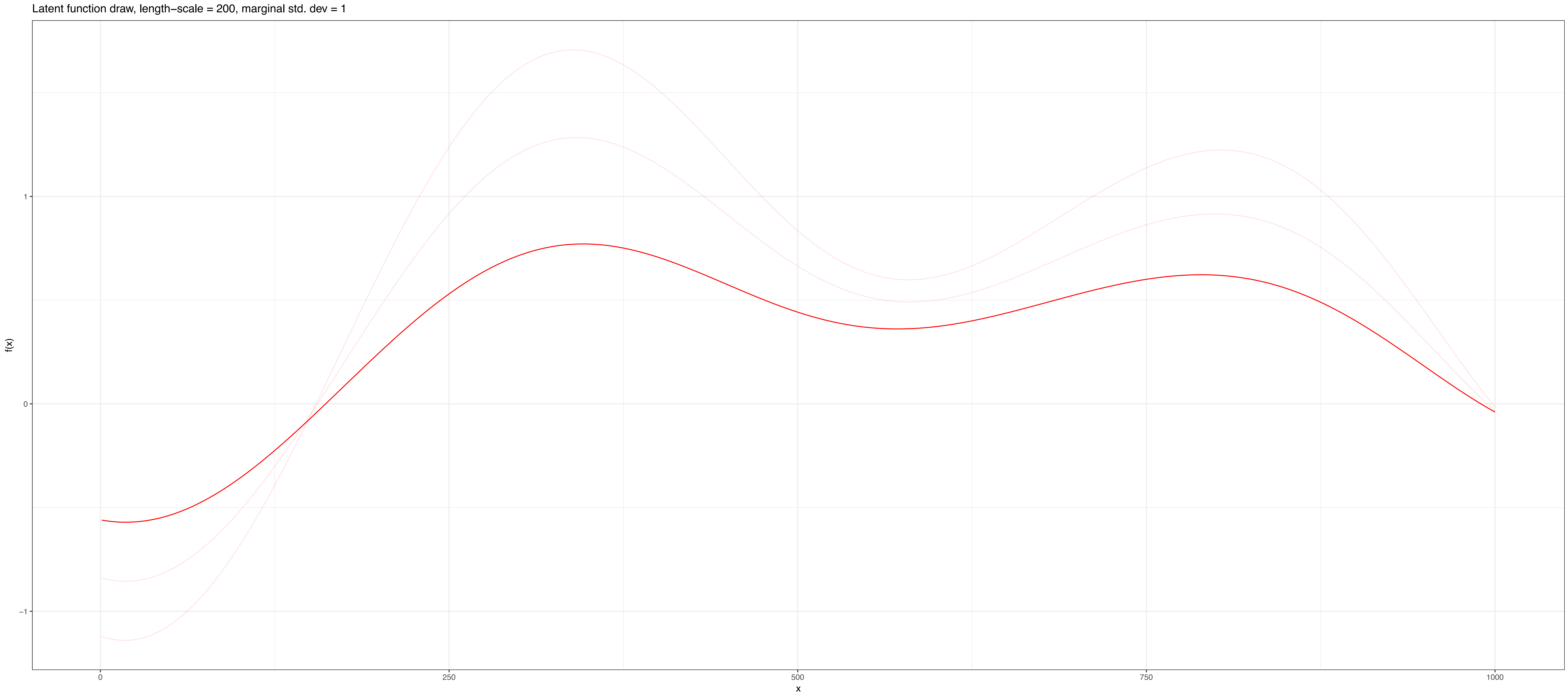
GENERATIVE MODEL – FINITE SAMPLE DRAW FROM GP



GENERATIVE MODEL – FINITE SAMPLE DRAW FROM GP



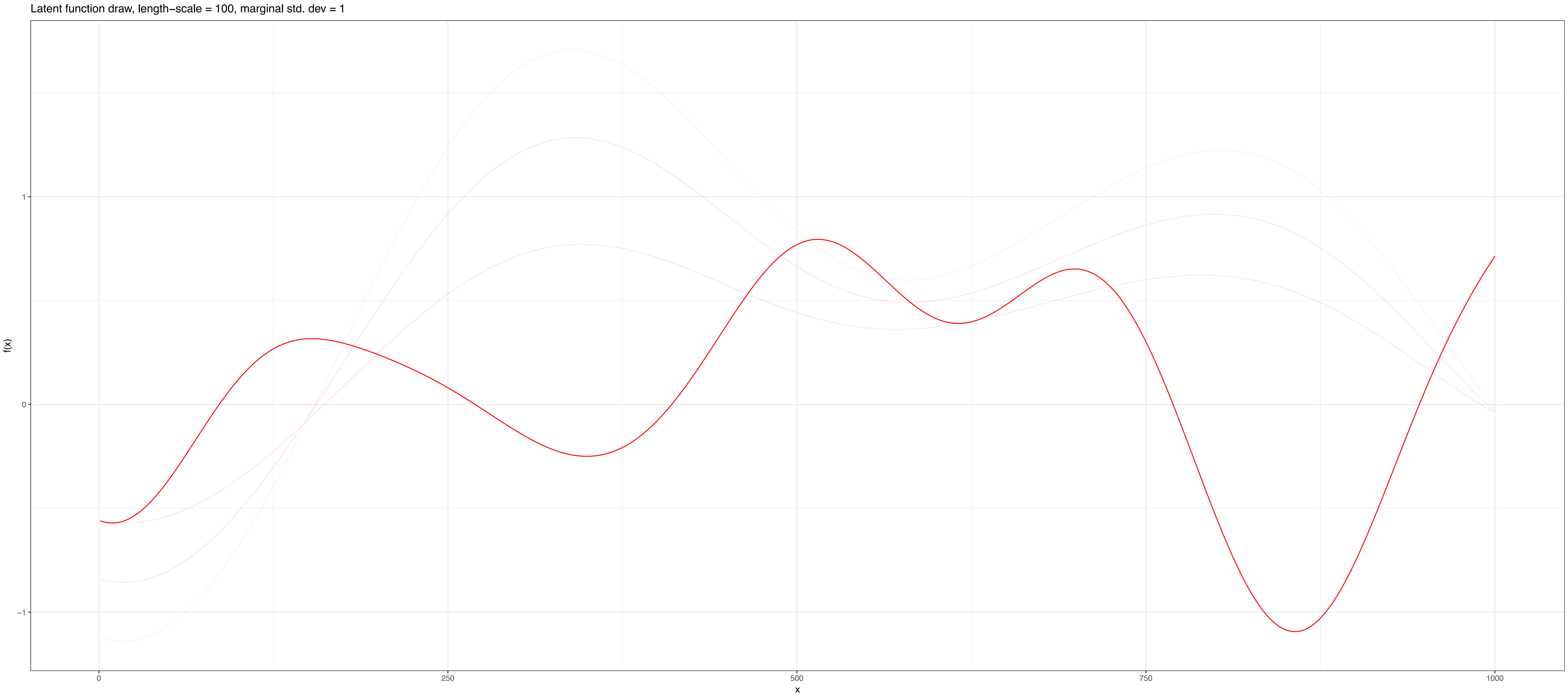
GENERATIVE MODEL – FINITE SAMPLE DRAW FROM GP



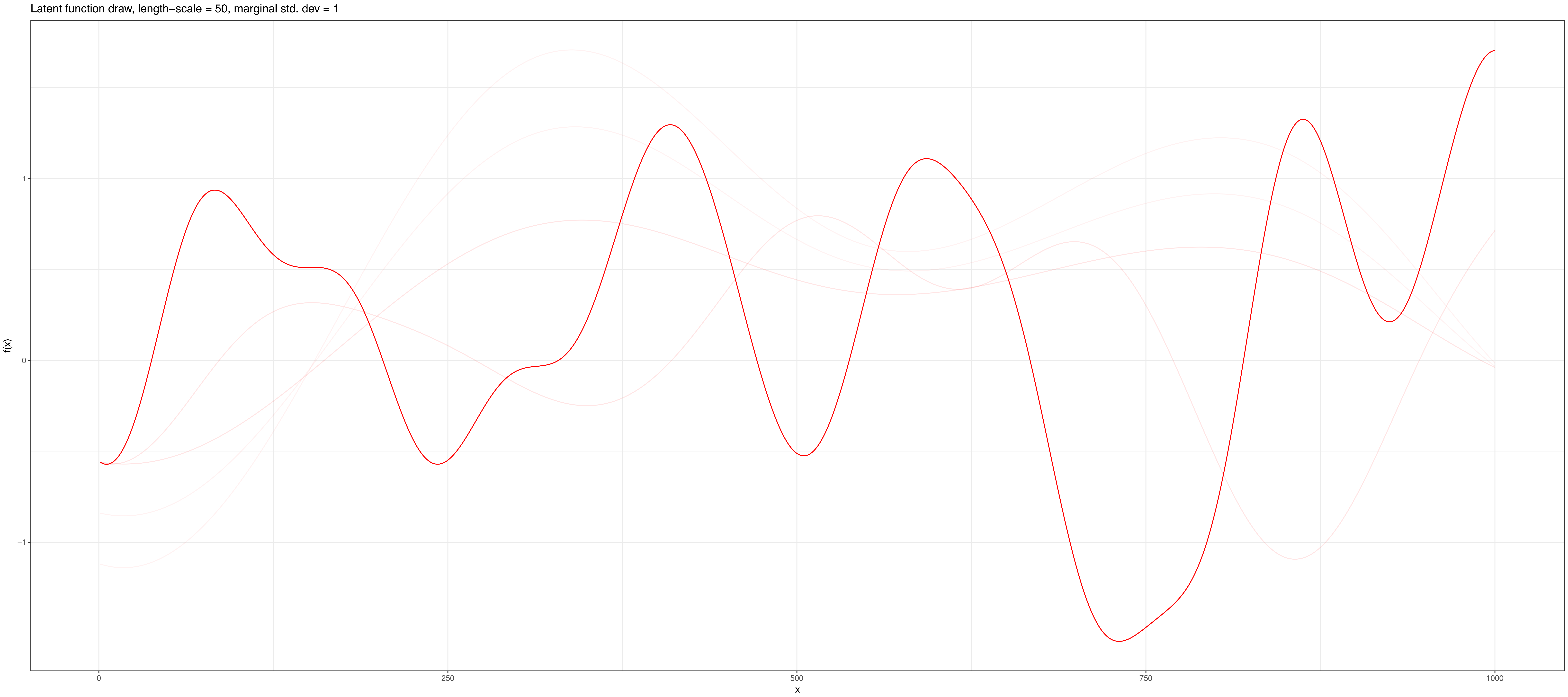
$$\begin{aligned} K_\theta(X)_{i,j} &= \operatorname{cov}\left(f(x_i), f(x_j)\right) \\ &= k\left(x_i, x_j \middle|\, \theta\right) \\ &= \alpha^2 \exp\left(-\frac{1}{2\ell^2}(x_i - x_j)^2\right) \end{aligned}$$

ℓ **LENGTH-SCALE**

GENERATIVE MODEL – FINITE SAMPLE DRAW FROM GP



GENERATIVE MODEL – FINITE SAMPLE DRAW FROM GP



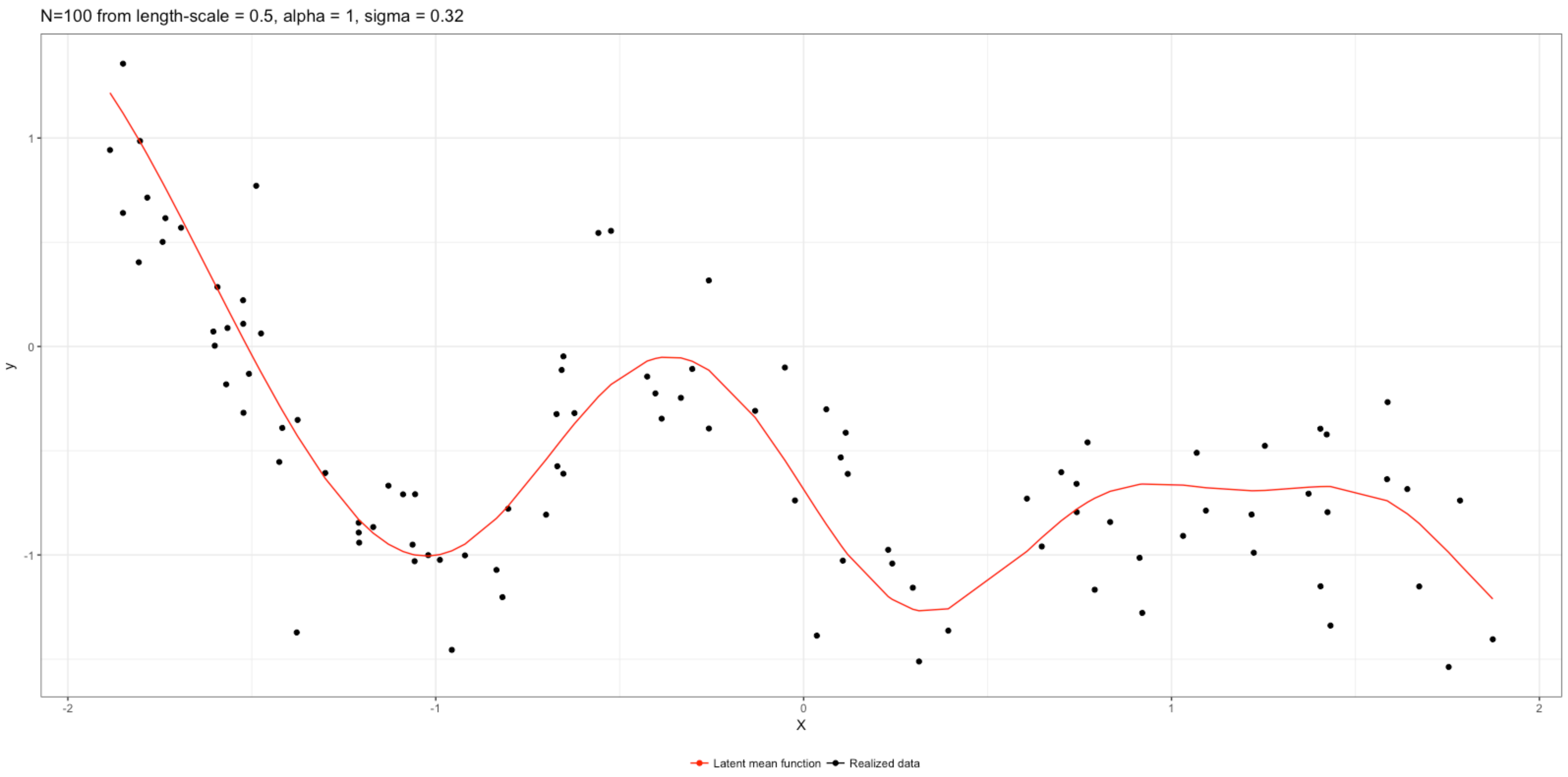
BUILDING GP MODELS IN STAN

GENERATIVE MODEL – FINITE SAMPLE DRAW FROM GP

$$
\begin{aligned}
\ell &\sim \mathrm{Gamma}(2, 2) \\
\alpha &\sim \text{Half-Normal}(0, 1) \\
\mathbf{f} &\sim \mathrm{MultiNormal}(0, K_{\ell,\alpha}(\mathbf{x}, \mathbf{x})) \\
\sigma &\sim \text{Half-Normal}(0, 1) \\
y_i &\sim \mathrm{Normal}(f_i, \sigma)
\end{aligned}
$$

$$\forall i \in \{1, \ldots, N\}, \mathbf{f}, \mathbf{x} \in R^N$$

GENERATIVE MODEL – FINITE SAMPLE DRAW FROM GP



DEFINE PROBABILITY MODEL – MARGINAL LIKELIHOOD

$$\ell \sim \text{Gamma}(2, 2)$$

$$\alpha \sim \text{Half-Normal}(0, 1)$$

$$\sigma \sim \text{Half-Normal}(0, 1)$$

$$\mathbf{y} \sim \text{MultiNormal}(0, K_{\ell,\alpha}(\mathbf{x}, \mathbf{x}) + \sigma^2 I_n)$$

$$\mathbf{y}, \mathbf{x} \in R^N$$

DEFINE PROBABILITY MODEL

```
data {
  int<lower=1> N;
  vector[N] y;
  real x[N];
  vector[N] zeros;
}
```

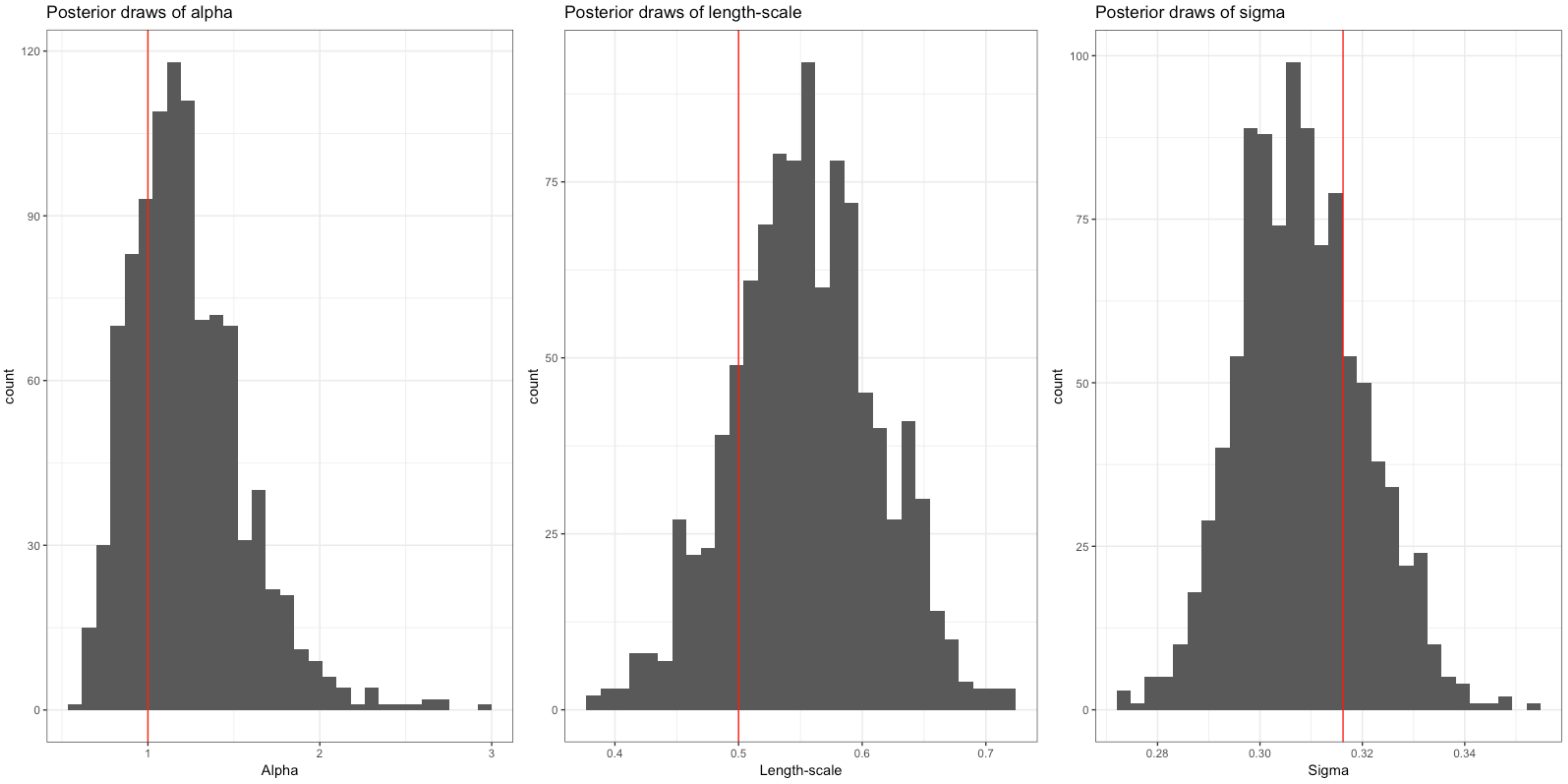
DEFINE PROBABILITY MODEL

```
parameters {
  real<lower=0> length_scale;
  real<lower=0> alpha;
  real<lower=0> sigma;
}
```

DEFINE PROBABILITY MODEL

```
parameters {
  real<lower=0> length_scale;
  real<lower=0> alpha;
  real<lower=0> sigma;
}
model {
  matrix[N, N] L_cov;
  {
    matrix[N, N] cov;
    cov = cov_exp_quad(x,
                              alpha, length_scale);
    for (n in 1:N)
      cov[n, n] = cov[n, n] + square(sigma);
    L_cov = cholesky_decompose(cov);
  }
  length_scale ~ gamma(2, 2);
  alpha ~ normal(0, 1);
  sigma ~ normal(0, 1);
  y ~ multi_normal_cholesky(zeros, L_cov);
}
```

RESULTS


Posterior draws of alpha
count
Alpha
0
30
60
90
120
1
2
3
Posterior draws of length-scale
count
Length-scale
0
25
50
75
0.4
0.5
0.6
0.7
Posterior draws of sigma
count
Sigma
0
25
50
75
100
0.28
0.30
0.32
0.34

DEFINE PROBABILITY MODEL - LATENT VARIABLE

$$
\begin{aligned}
\ell &\sim \mathrm{Gamma}(2, 2) \\
\alpha &\sim \mathrm{Half\text{-}Normal}(0, 1) \\
\mathbf{f} &\sim \mathrm{MultiNormal}(0, K_{\ell,\alpha}(\mathbf{x}, \mathbf{x})) \\
\sigma &\sim \mathrm{Half\text{-}Normal}(0, 1) \\
y_i &\sim \mathrm{Normal}(f_i, \sigma)
\end{aligned}
$$

$$
\forall i \in \{1, \ldots, N\}, \mathbf{f}, \mathbf{x} \in R^N
$$

DEFINE PROBABILITY MODEL - LATENT VARIABLE

$$L \times L^T = K_{\ell,\alpha}$$

DEFINE PROBABILITY MODEL - LATENT VARIABLE

$$L = \text{cholesky\_decompose}(K_{\ell,\alpha})$$

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

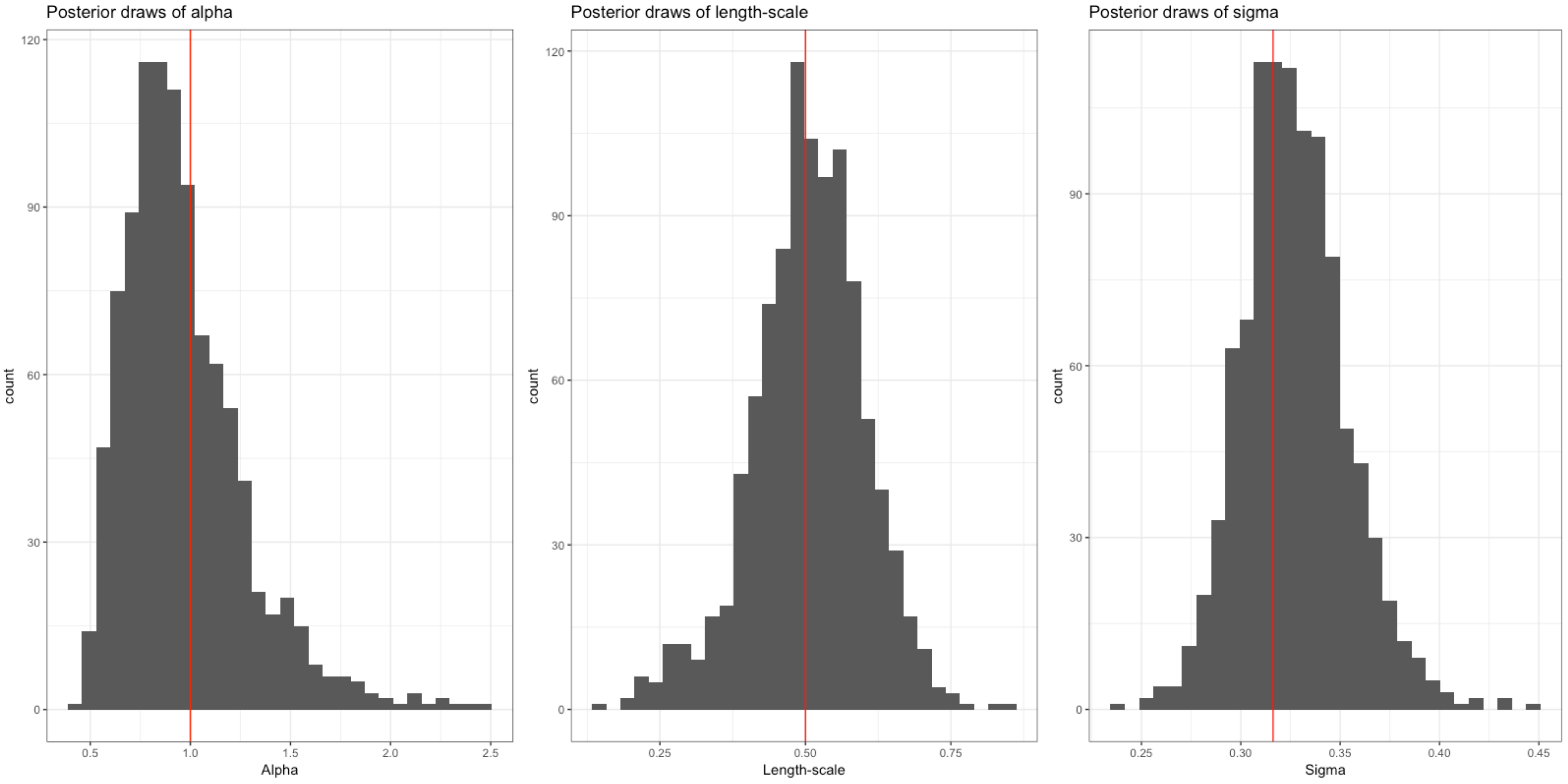
$$\mathbf{f} = L \times \boldsymbol{\eta}$$

$$\mathbf{f} \sim \text{MultiNormal}(0, K_{\ell,\alpha}(x, x))$$

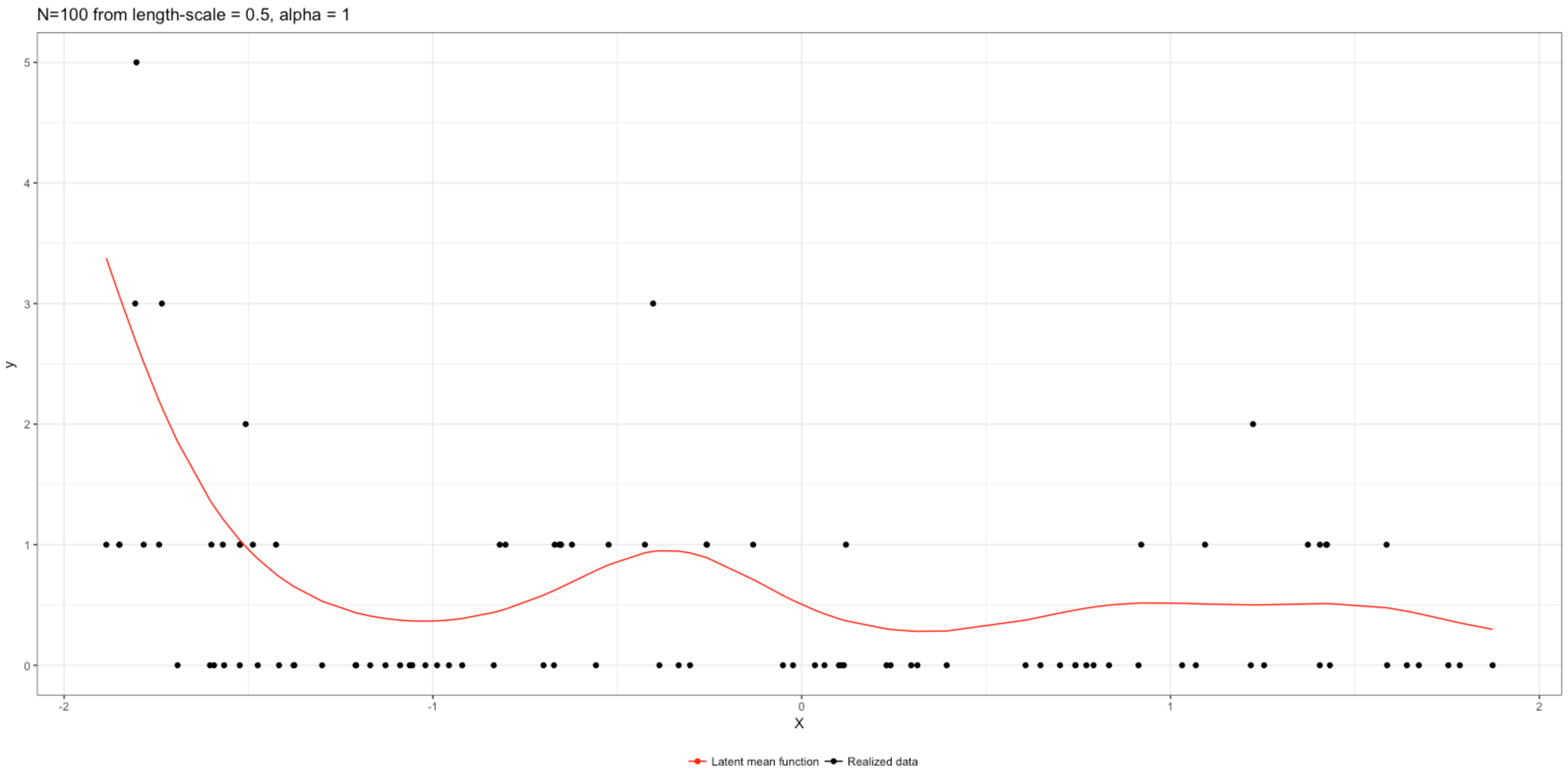
DEFINE PROBABILITY MODEL

```
parameters {
  real<lower=0> length_scale;
  real<lower=0> alpha;
  real<lower=0> sigma;
  vector[N] eta;
}
model {
  vector[N] f;
  {
    matrix[N, N] L_cov;
    matrix[N, N] cov;
    cov = cov_exp_quad(x, alpha, length_scale);
    for (n in 1:N)
      cov[n, n] = cov[n, n] + 1e-12;
    L_cov = cholesky_decompose(cov);
    f = L_cov * eta;
  }
  length_scale ~ gamma(2, 2);
  alpha ~ normal(0, 1);
  sigma ~ normal(0, 1);
  eta ~ normal(0, 1);
  y ~ normal(f, sigma);
}
```

RESULTS


Posterior draws of alpha
count
Alpha
120
90
60
30
0
0.5
1.0
1.5
2.0
2.5
Posterior draws of length-scale
count
Length-scale
120
90
60
30
0
0.25
0.50
0.75
Posterior draws of sigma
count
Sigma
90
60
30
0
0.25
0.30
0.35
0.40
0.45

COUNT DATA


N=100 from length-scale = 0.5, alpha = 1
y
0
1
2
3
4
5
x
-2
-1
0
1
2
Latent mean function
Realized data

DEFINE PROBABILITY MODEL – LATENT VARIABLE POISSON

$$\ell \sim \text{Gamma}(2, 2)$$

$$\alpha \sim \text{Half-Normal}(0, 1)$$

$$\mathbf{f} \sim \text{MultiNormal}(0, K_{\ell,\alpha}(\mathbf{x}, \mathbf{x}))$$

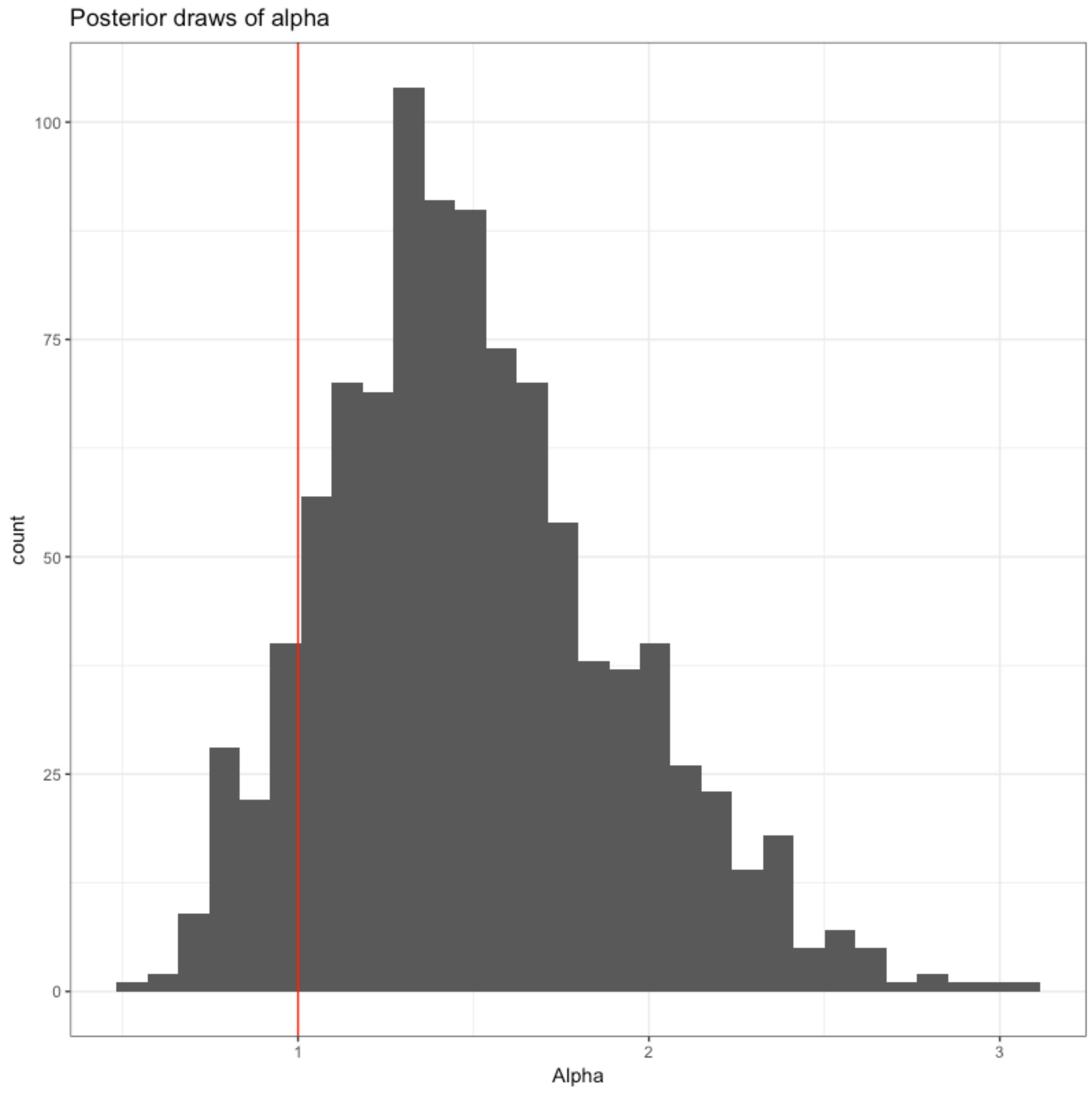
$$y_i \sim \text{Poisson}(\exp(f_i))$$

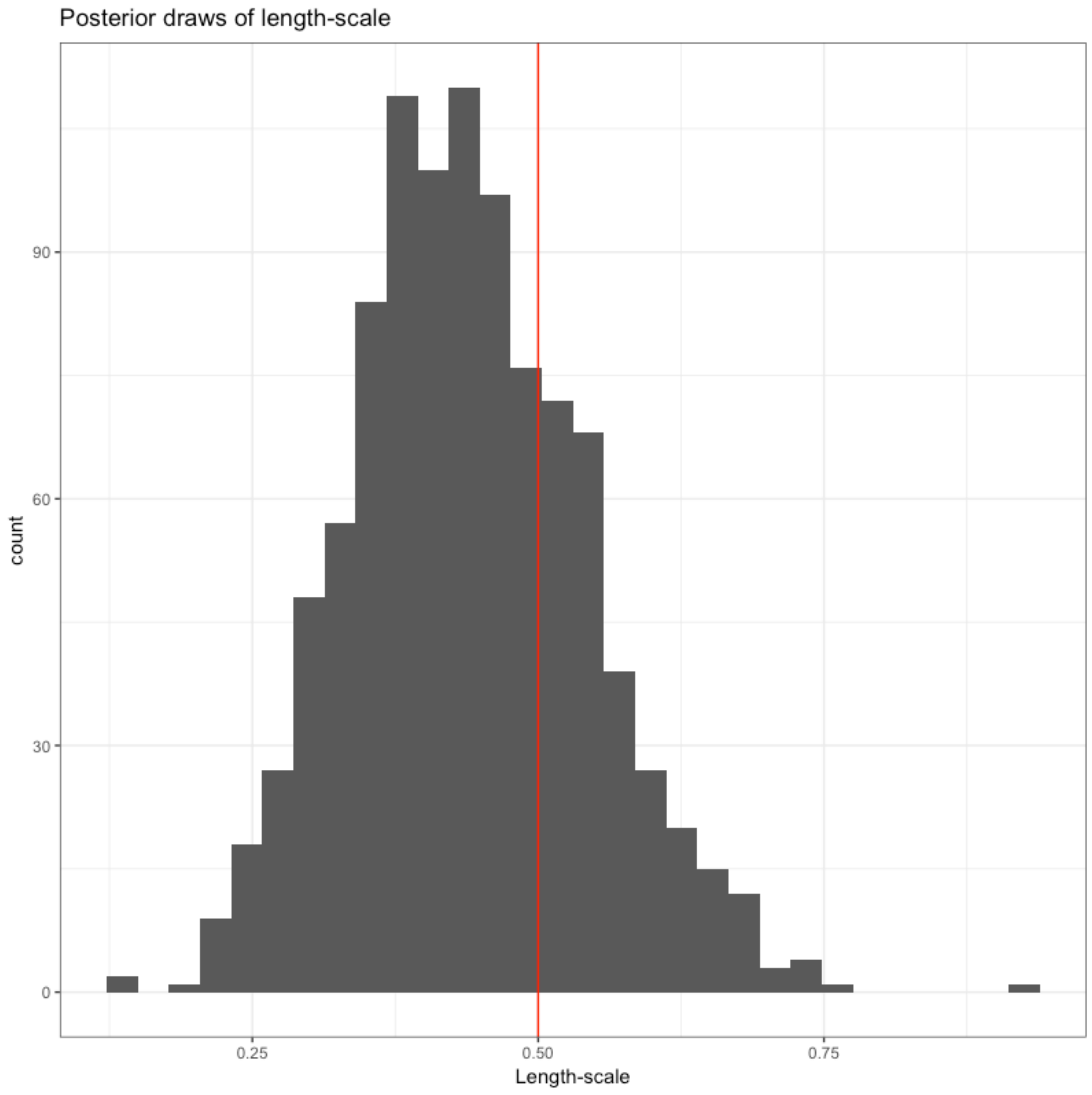
$$\forall i \in \{1, \ldots, N\}, \mathbf{f} \in R^N$$

DEFINE PROBABILITY MODEL

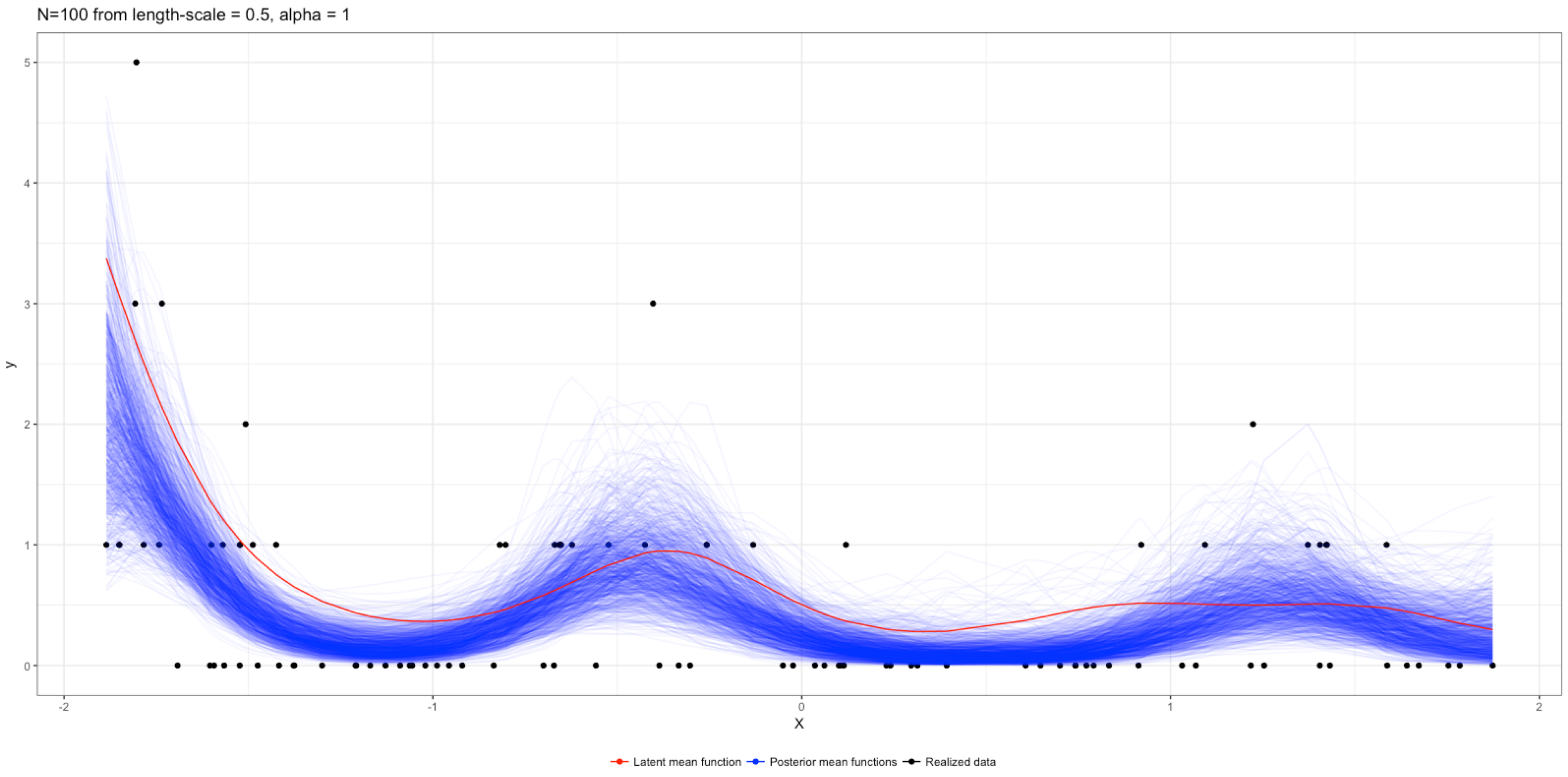
```
data {
  ...same as above...
  int y[N];
  ...same as above...
}
parameters {
  real<lower=0> length_scale;
  real<lower=0> alpha;
  vector[N] eta;
}
model {
  ...same as above...
  y ~ poisson_log(f);
}
```

RESULTS


Posterior draws of alpha
count
100
75
50
25
0
1
2
3
Alpha

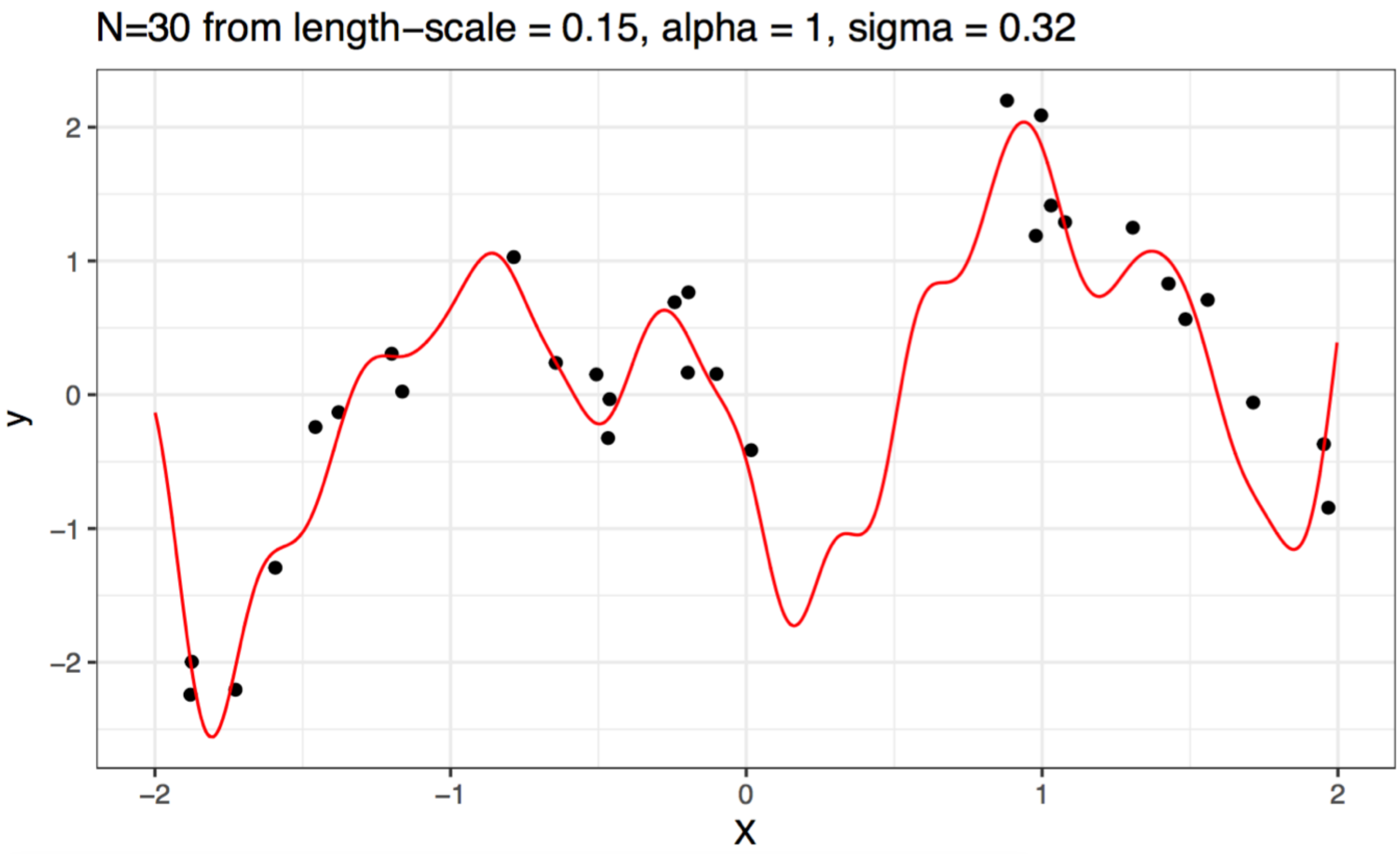

Posterior draws of length-scale
count
90
60
30
0
0.25
0.50
0.75
Length-scale

RESULTS

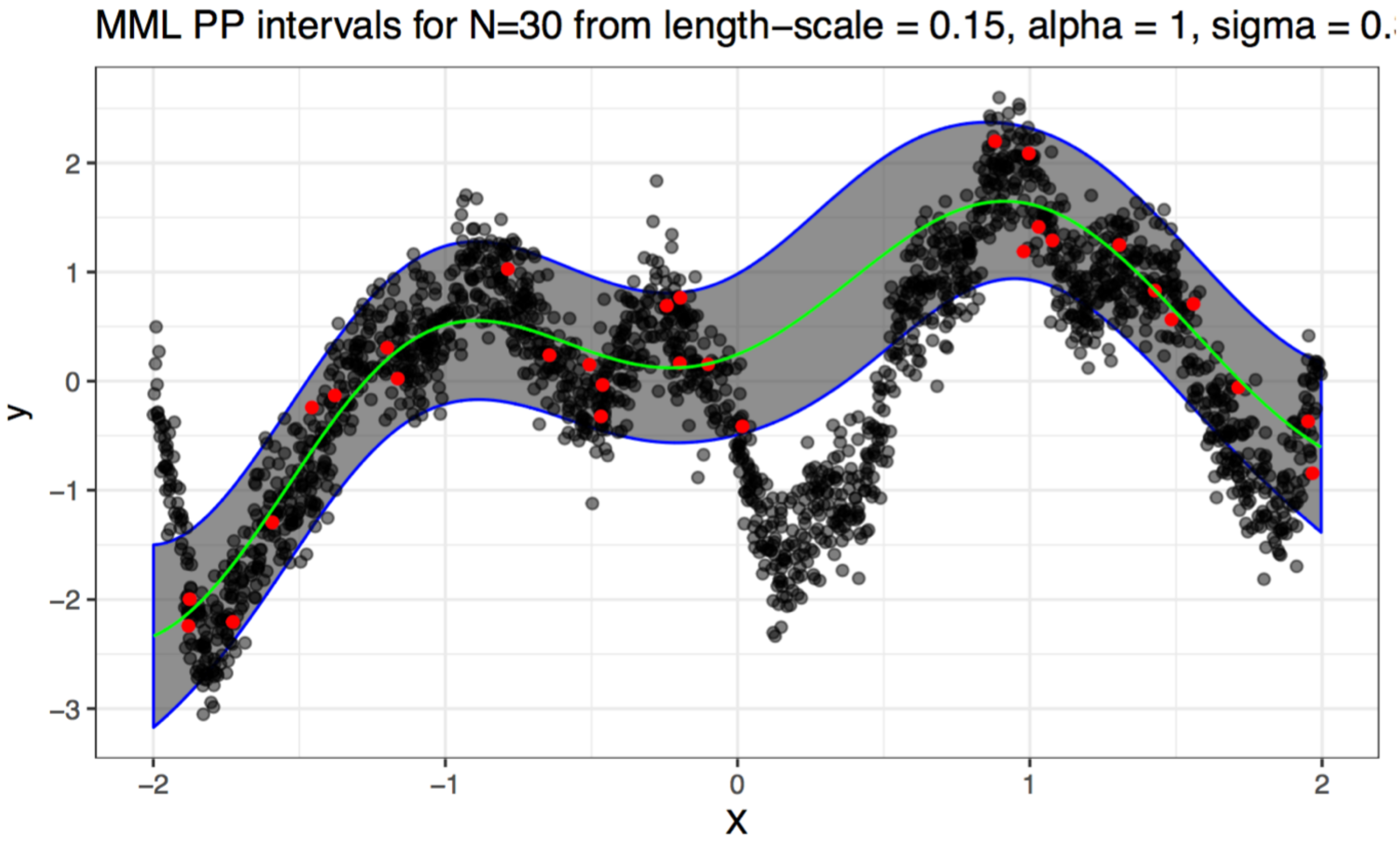

N=100 from length-scale = 0.5, alpha = 1
y
X
5
4
3
2
1
0
-2
-1
0
1
2
Latent mean function
Posterior mean functions
Realized data

PITFALLS OF ESTIMATING GPS

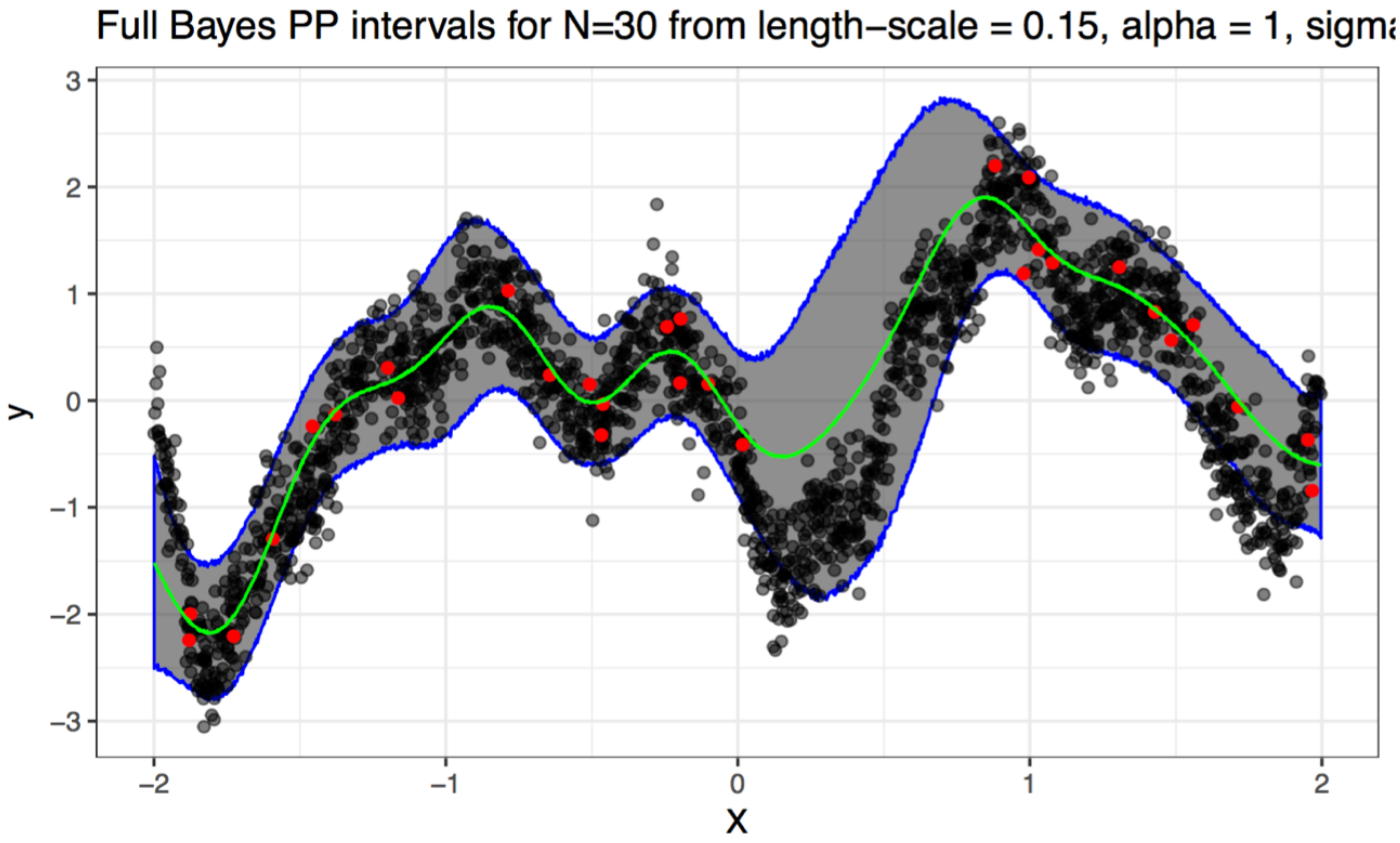
GENERATIVE MODEL – FINITE SAMPLE DRAW FROM GP



GENERATIVE MODEL – UNIFORM PRIORS ON HYPERPARAMETERS

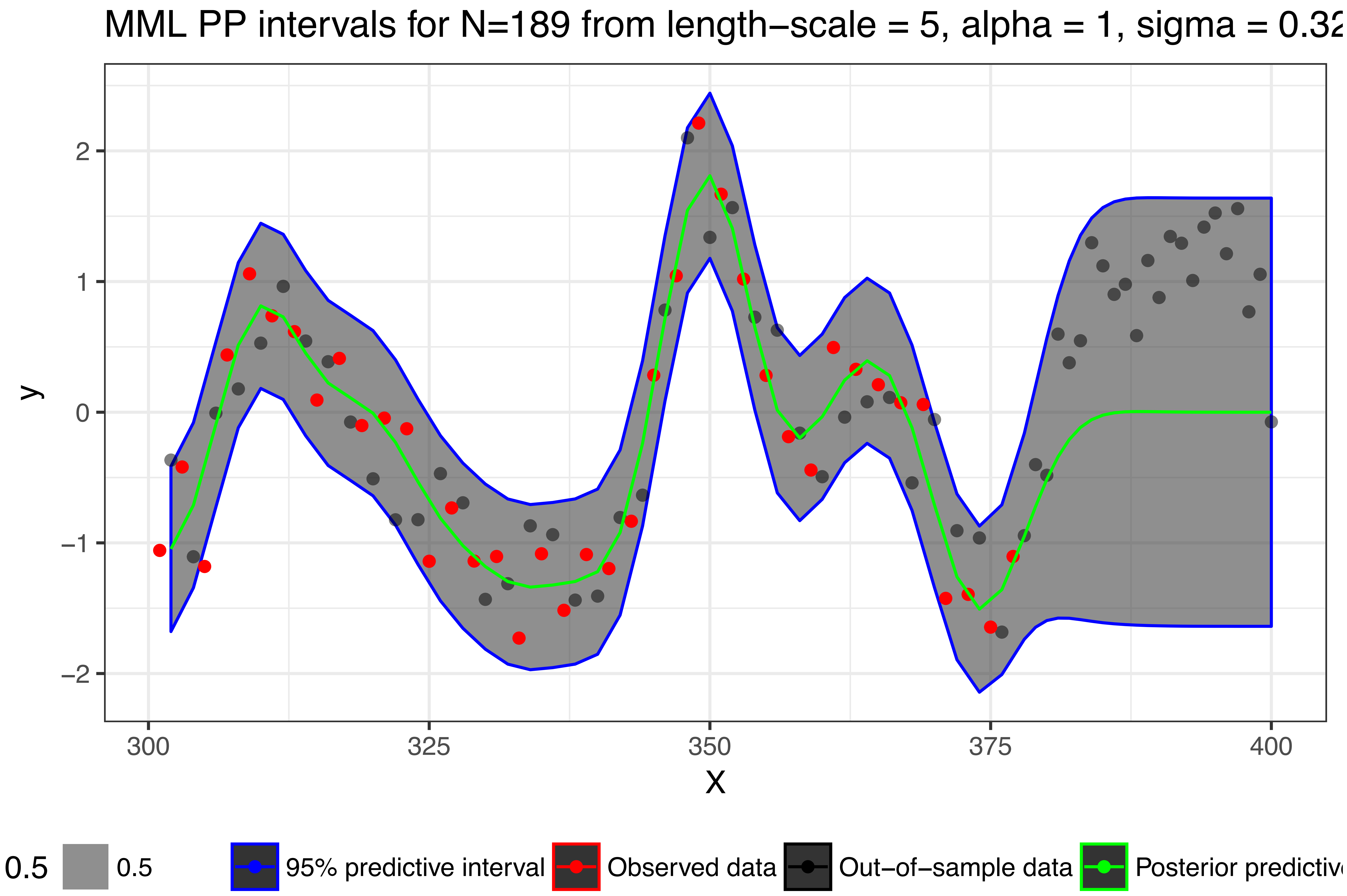


GENERATIVE MODEL – PROPER PRIORS ON HYPERPARAMETERS

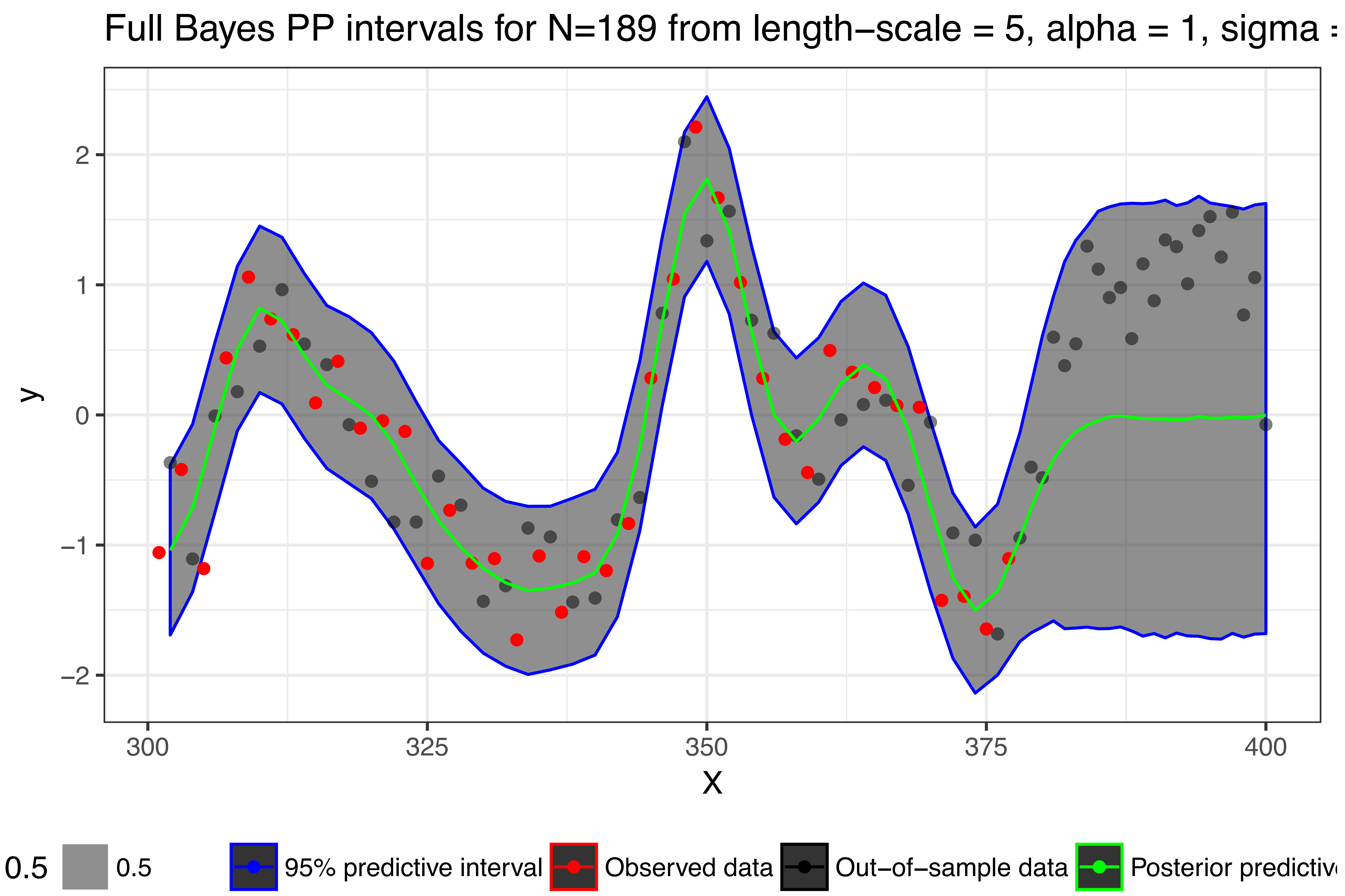


PITFALLS OF ESTIMATING GPS TIME SERIES

GENERATIVE MODEL – UNIFORM PRIORS ON HYPERPARAMETERS

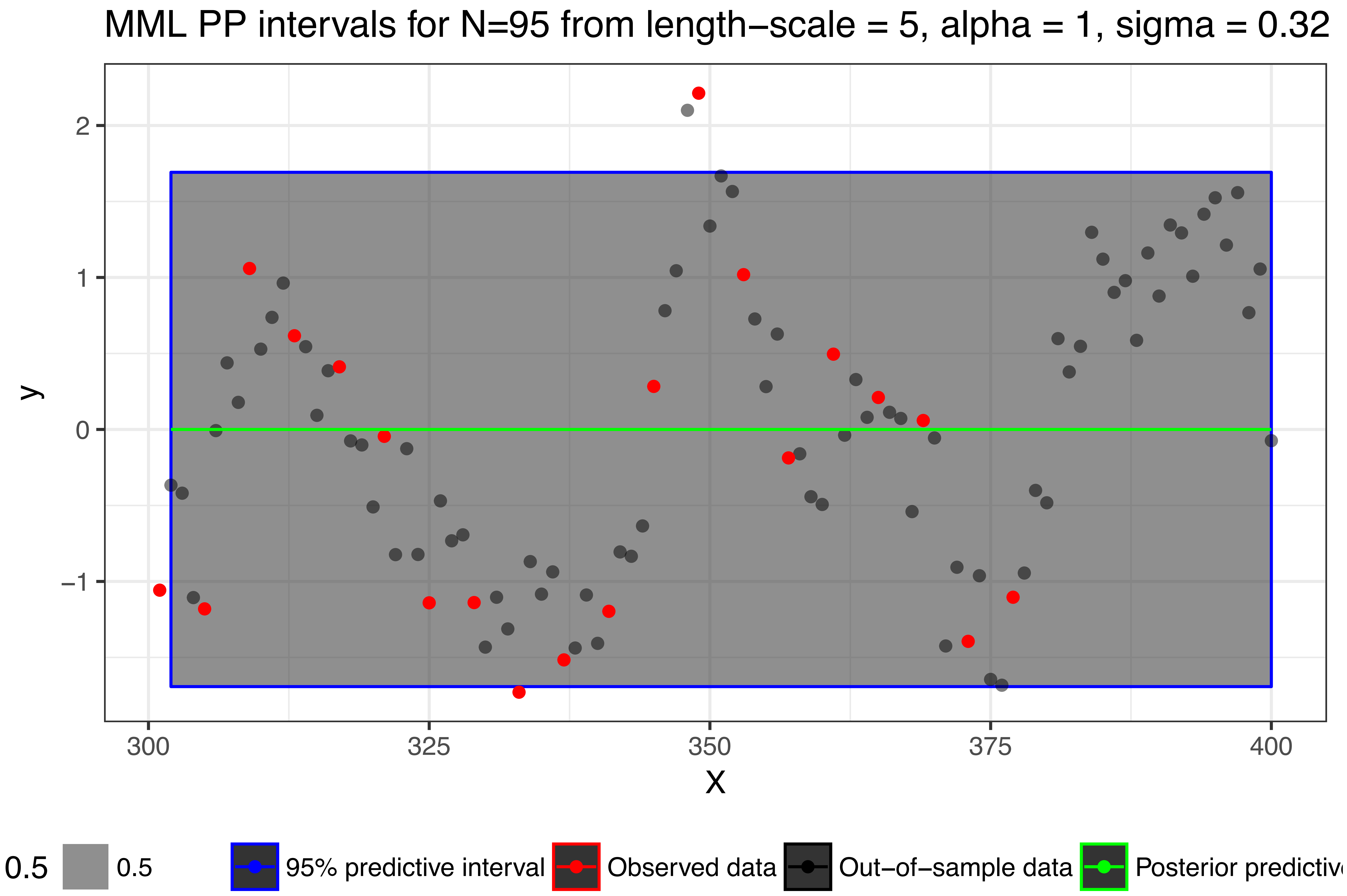


GENERATIVE MODEL - PROPER PRIORS ON HYPERPARAMETERS

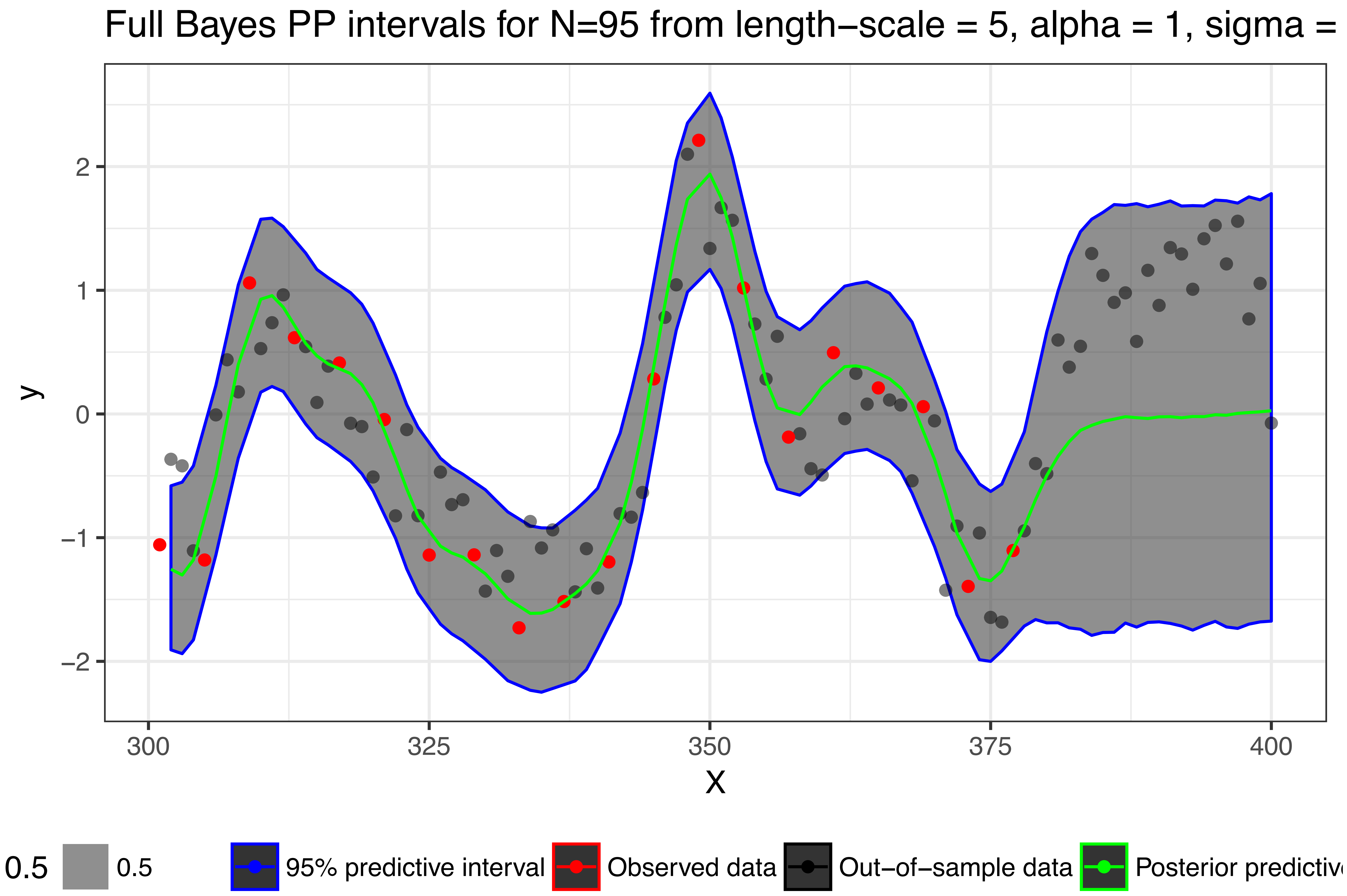


DELETE EVERY OTHER OBSERVATION

GENERATIVE MODEL – UNIFORM PRIORS ON HYPERPARAMETERS

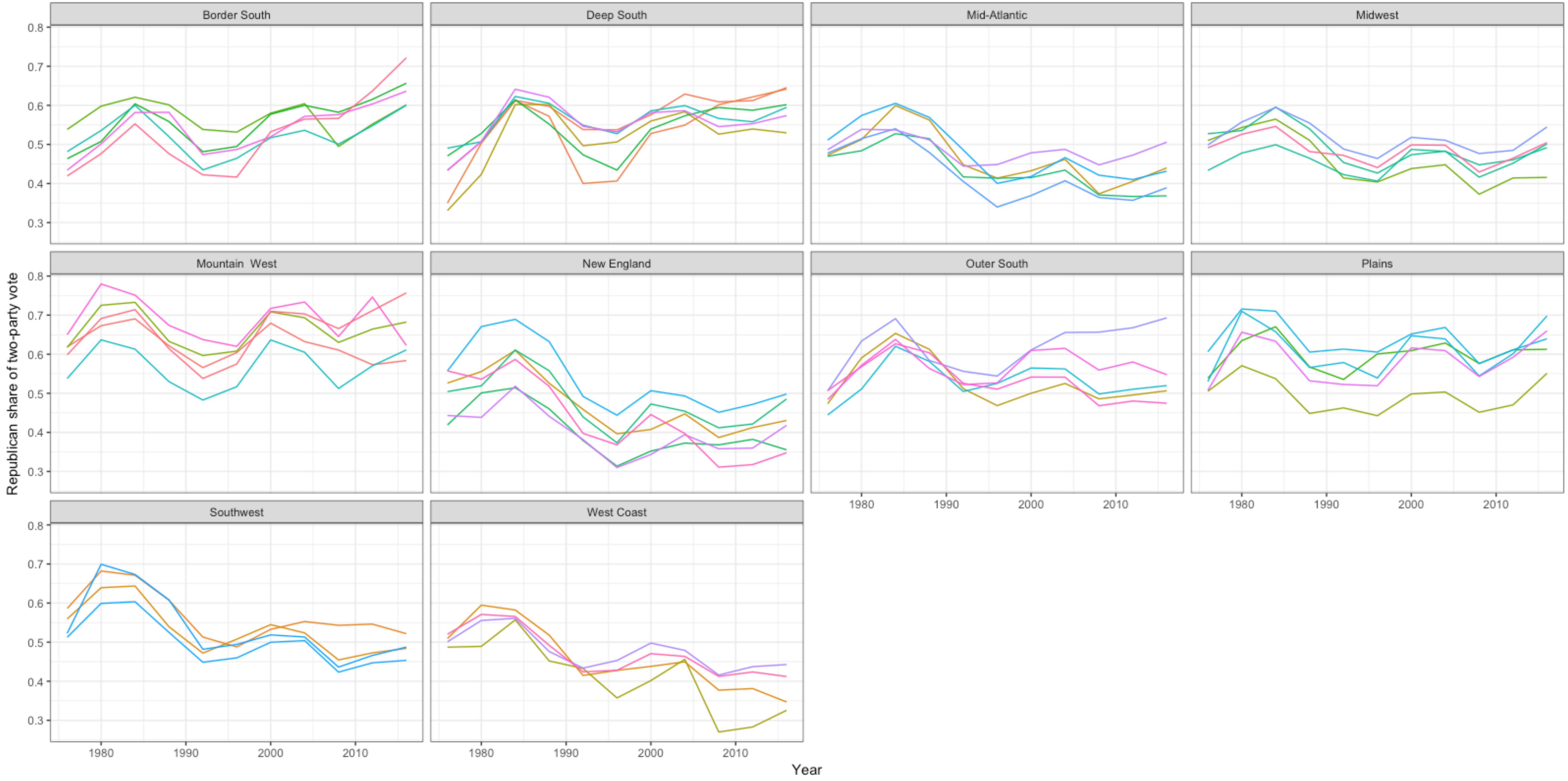


GENERATIVE MODEL - PROPER PRIORS ON HYPERPARAMETERS



PRESIDENTIAL FORECASTING

FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028



FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028

$$y_{t,j} \sim \mathrm{Beta}(\mu_{t,j}\,\nu, (1 - \mu_{t,j})\,\nu)$$

FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028

$$y_{t,j} \sim \mathrm{Beta}(\mu_{t,j}\,\nu, (1 - \mu_{t,j})\,\nu)$$

$$\operatorname{logit} \mu_{t,j} = \theta_t^{\text{year}} + \theta_j^{\text{state}} + \theta_{k[j]}^{\text{region}} + \gamma_{t,j} + \delta_{t,k[j]}$$

FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028

$$
\begin{aligned}
y_{t,j} &\sim \mathrm{Beta}(\mu_{t,j}\,\nu, (1-\mu_{t,j})\,\nu) \\
\operatorname{logit}\mu_{t,j} &= \theta^{\text{year}}_{t} + \theta^{\text{state}}_{j} + \theta^{\text{region}}_{k[j]} \\
&\quad + \gamma_{t,j} + \delta_{t,k[j]} \\
\boldsymbol{\gamma_j} &\sim \mathrm{MultiNormal}(0, K_{\ell^{\gamma}_1,\alpha^{\gamma}_1} + K_{\ell^{\gamma}_2,\alpha^{\gamma}_2}) \\
\boldsymbol{\delta_k} &\sim \mathrm{MultiNormal}(0, K_{\ell^{\delta}_1,\alpha^{\delta}_1} + K_{\ell^{\delta}_2,\alpha^{\delta}_2})
\end{aligned}
$$

FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028

$$y_{t,j} \sim \mathrm{Beta}(\mu_{t,j}\,\nu, (1-\mu_{t,j})\,\nu)$$

$$\mathrm{logit}\,\mu_{t,j} = \theta_t^{\mathrm{year}} + \theta_j^{\mathrm{state}} + \theta_{k[j]}^{\mathrm{region}} + \gamma_{t,j} + \delta_{t,k[j]}$$

Short-term wiggle

Long-term trend

State $\boldsymbol{\gamma_j} \sim \mathrm{MultiNormal}(0, K_{\ell_1^\gamma, \alpha_1^\gamma} + K_{\ell_2^\gamma, \alpha_2^\gamma})$

Region $\boldsymbol{\delta_k} \sim \mathrm{MultiNormal}(0, K_{\ell_1^\delta, \alpha_1^\delta} + K_{\ell_2^\delta, \alpha_2^\delta})$

FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028

$$\boldsymbol{\eta} \in R^{T,J}, L \in R^{T,T}$$

$$L = \text{cholesky\_decompose}(K_{\ell,\alpha})$$

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

$$\boldsymbol{\gamma} = L \times \boldsymbol{\eta}$$

$$\boldsymbol{\gamma}_{[,j]} \sim \text{MultiNormal}(0, K_{\ell,\alpha}(x, x))$$

FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028

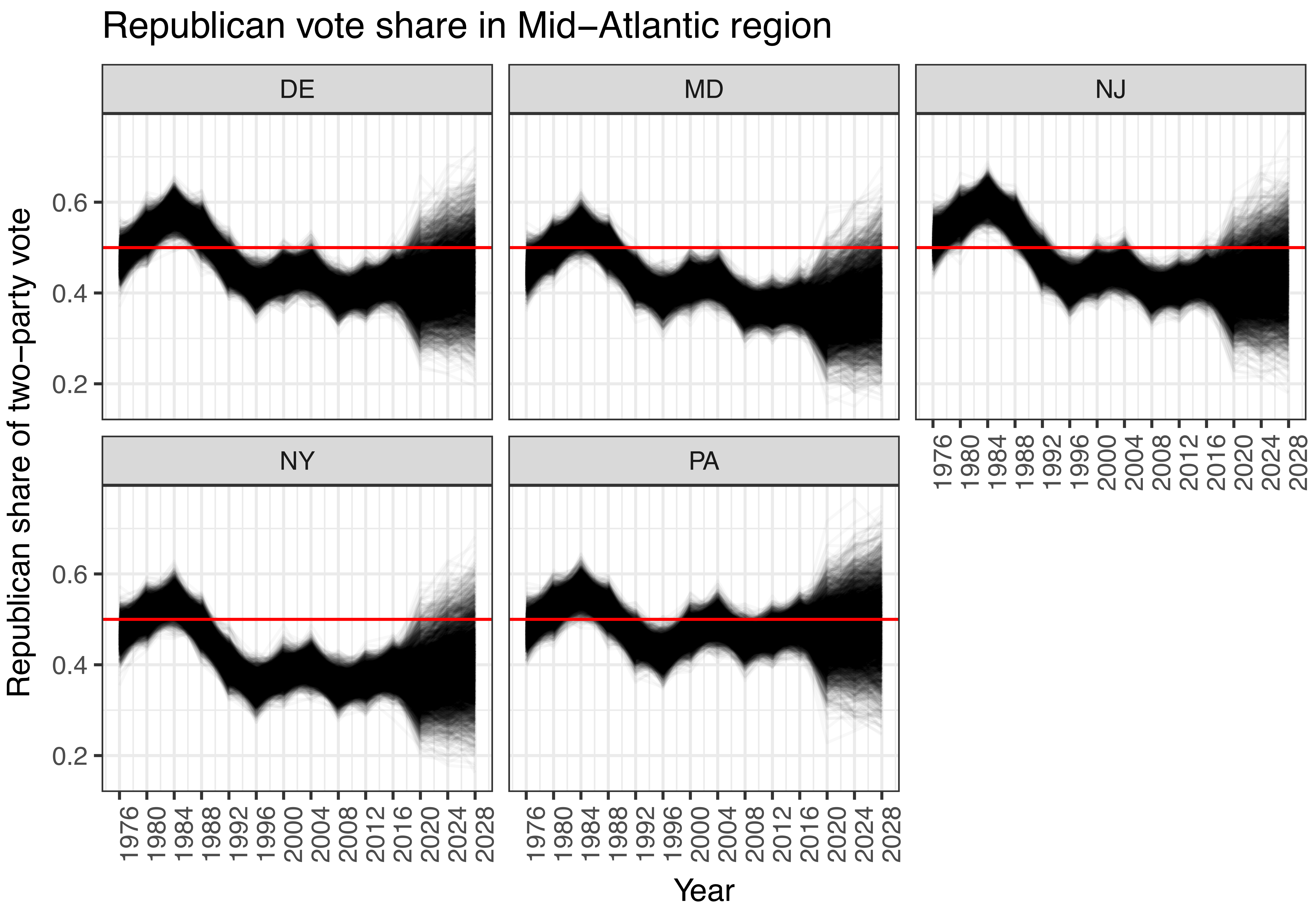
$$\ell_1^\gamma \sim \mathrm{Weibull}(30, 8)$$

$$\ell_2^\gamma \sim \mathrm{Weibull}(30, 3)$$

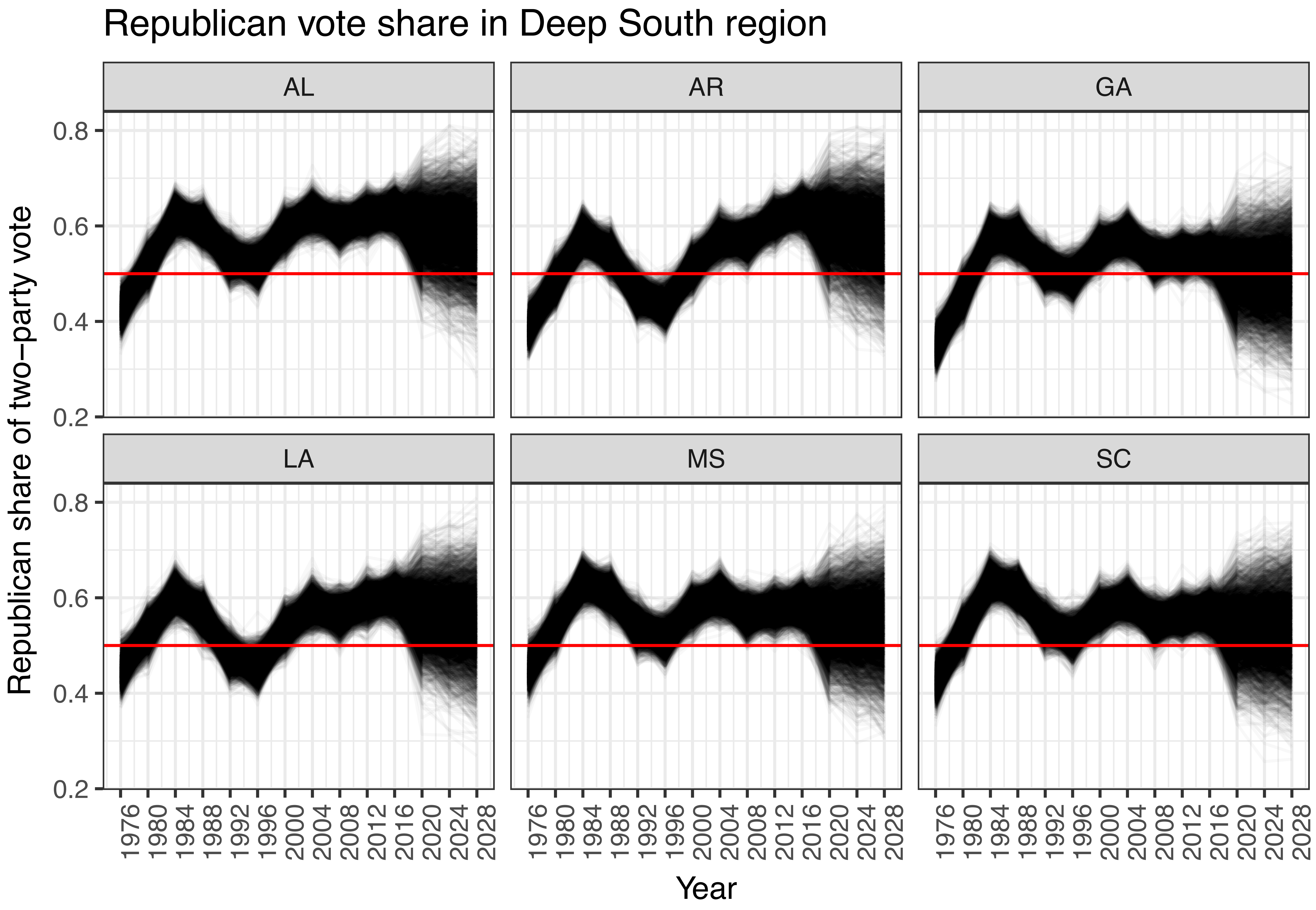
$$\ell_1^\delta \sim \mathrm{Weibull}(30, 8)$$

$$\ell_2^\delta \sim \mathrm{Weibull}(30, 3)$$

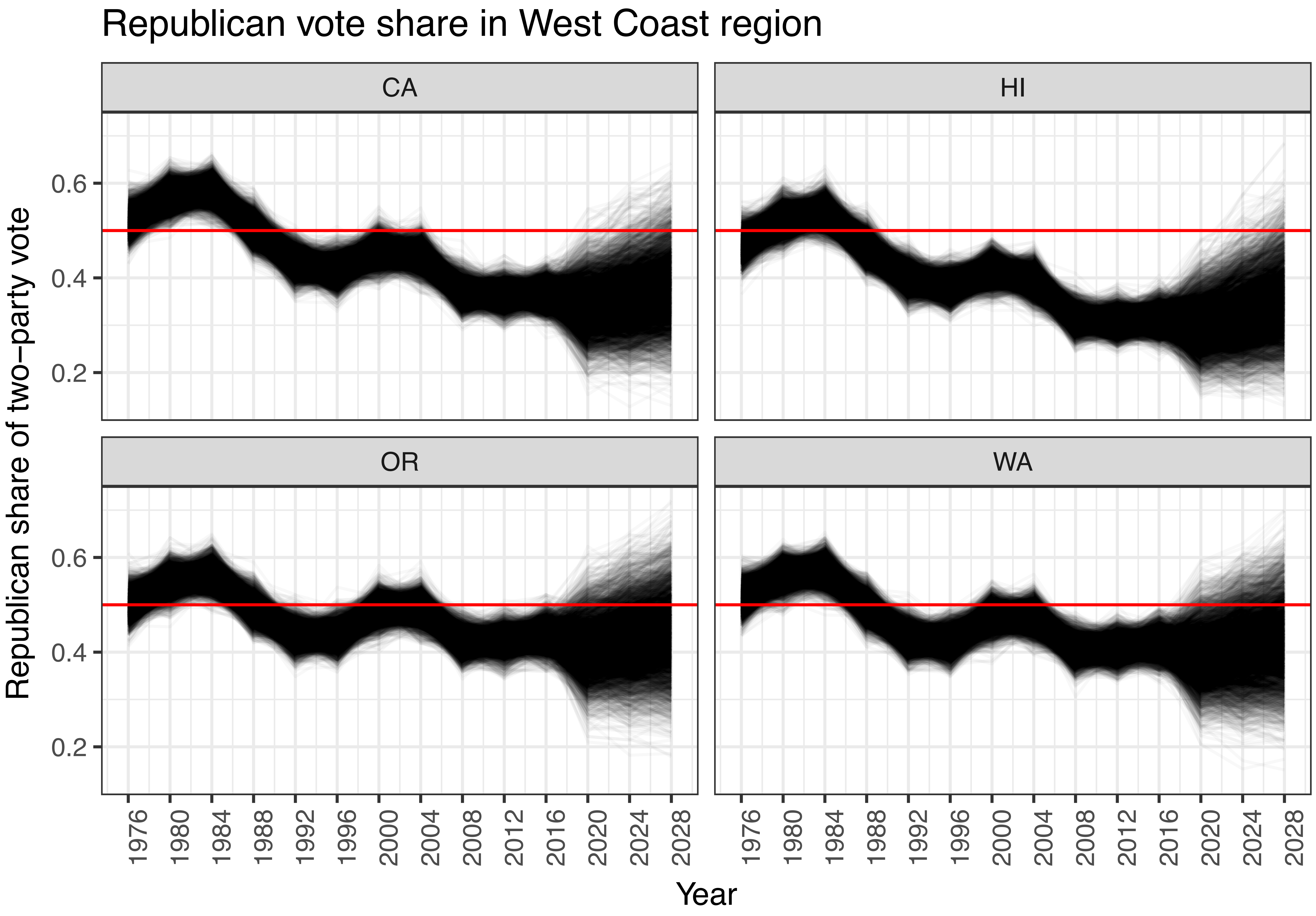
FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028



FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028



FORECASTING US PRESIDENTIAL ELECTIONS 2020 – 2028



SPATIOTEMPORAL MODELING

DATA STRUCTURE

$y_{1,1}$	$y_{1,2}$	$y_{1,3}$	$y_{1,4}$	$y_{1,5}$	$y_{1,6}$	$y_{1,7}$	$y_{1,8}$	$y_{1,9}$	$y_{1,10}$	$y_{1,11}$	$y_{1,12}$

DATA STRUCTURE

X2-direction

$y_{1,1}$	$y_{1,2}$	$y_{1,3}$	$y_{1,4}$	$y_{1,5}$	$y_{1,6}$	$y_{1,7}$	$y_{1,8}$	$y_{1,9}$	$y_{1,10}$	$y_{1,11}$	$y_{1,12}$

DATA STRUCTURE

x_2 dimension →

x_1 dimension ↓

$y_{1,1}$	$y_{1,2}$	$y_{1,3}$	$y_{1,4}$	$y_{1,5}$	$y_{1,6}$	$y_{1,7}$	$y_{1,8}$	$y_{1,9}$	$y_{1,10}$	$y_{1,11}$	$y_{1,12}$

DATA STRUCTURE

x_2 dimension

x_1 dimension

$y_{1,1}$	$y_{1,2}$	$y_{1,3}$	$y_{1,4}$	$y_{1,5}$	$y_{1,6}$	$y_{1,7}$	$y_{1,8}$	$y_{1,9}$	$y_{1,10}$	$y_{1,11}$	$y_{1,12}$
$y_{2,1}$	$y_{2,2}$	$y_{2,3}$	$y_{2,4}$	$y_{2,5}$	$y_{2,6}$	$y_{2,7}$	$y_{2,8}$	$y_{2,9}$	$y_{2,10}$	$y_{2,11}$	$y_{2,12}$

DATA STRUCTURE

x_2 dimension

x_1 dimension

$y_{1,1}$	$y_{1,2}$	$y_{1,3}$	$y_{1,4}$	$y_{1,5}$	$y_{1,6}$	$y_{1,7}$	$y_{1,8}$	$y_{1,9}$	$y_{1,10}$	$y_{1,11}$	$y_{1,12}$
$y_{2,1}$	$y_{2,2}$	$y_{2,3}$	$y_{2,4}$	$y_{2,5}$	$y_{2,6}$	$y_{2,7}$	$y_{2,8}$	$y_{2,9}$	$y_{2,10}$	$y_{2,11}$	$y_{2,12}$
$y_{3,1}$	$y_{3,2}$	$y_{3,3}$	$y_{3,4}$	$y_{3,5}$	$y_{3,6}$	$y_{3,7}$	$y_{3,8}$	$y_{3,9}$	$y_{3,10}$	$y_{3,11}$	$y_{3,12}$
$y_{4,1}$	$y_{4,2}$	$y_{4,3}$	$y_{4,4}$	$y_{4,5}$	$y_{4,6}$	$y_{4,7}$	$y_{4,8}$	$y_{4,9}$	$y_{4,10}$	$y_{4,11}$	$y_{4,12}$
$y_{5,1}$	$y_{5,2}$	$y_{5,3}$	$y_{5,4}$	$y_{5,5}$	$y_{5,6}$	$y_{5,7}$	$y_{5,8}$	$y_{5,9}$	$y_{5,10}$	$y_{5,11}$	$y_{5,12}$
$y_{6,1}$	$y_{6,2}$	$y_{6,3}$	$y_{6,4}$	$y_{6,5}$	$y_{6,6}$	$y_{6,7}$	$y_{6,8}$	$y_{6,9}$	$y_{6,10}$	$y_{6,11}$	$y_{6,12}$
$y_{7,1}$	$y_{7,2}$	$y_{7,3}$	$y_{7,4}$	$y_{7,5}$	$y_{7,6}$	$y_{7,7}$	$y_{7,8}$	$y_{7,9}$	$y_{7,10}$	$y_{7,11}$	$y_{7,12}$
$y_{8,1}$	$y_{8,2}$	$y_{8,3}$	$y_{8,4}$	$y_{8,5}$	$y_{8,6}$	$y_{8,7}$	$y_{8,8}$	$y_{8,9}$	$y_{8,10}$	$y_{8,11}$	$y_{8,12}$

DATA STRUCTURE

x_2 dimension →

x_1 dimension ↓

$\mu_{1,1}$	$\mu_{1,2}$	$\mu_{1,3}$	$\mu_{1,4}$	$\mu_{1,5}$	$\mu_{1,6}$	$\mu_{1,7}$	$\mu_{1,8}$	$\mu_{1,9}$	$\mu_{1,10}$	$\mu_{1,11}$	$\mu_{1,12}$
$\mu_{2,1}$	$\mu_{2,2}$	$\mu_{2,3}$	$\mu_{2,4}$	$\mu_{2,5}$	$\mu_{2,6}$	$\mu_{2,7}$	$\mu_{2,8}$	$\mu_{2,9}$	$\mu_{2,10}$	$\mu_{2,11}$	$\mu_{2,12}$
$\mu_{3,1}$	$\mu_{3,2}$	$\mu_{3,3}$	$\mu_{3,4}$	$\mu_{3,5}$	$\mu_{3,6}$	$\mu_{3,7}$	$\mu_{3,8}$	$\mu_{3,9}$	$\mu_{3,10}$	$\mu_{3,11}$	$\mu_{3,12}$
$\mu_{4,1}$	$\mu_{4,2}$	$\mu_{4,3}$	$\mu_{4,4}$	$\mu_{4,5}$	$\mu_{4,6}$	$\mu_{4,7}$	$\mu_{4,8}$	$\mu_{4,9}$	$\mu_{4,10}$	$\mu_{4,11}$	$\mu_{4,12}$
$\mu_{5,1}$	$\mu_{5,2}$	$\mu_{5,3}$	$\mu_{5,4}$	$\mu_{5,5}$	$\mu_{5,6}$	$\mu_{5,7}$	$\mu_{5,8}$	$\mu_{5,9}$	$\mu_{5,10}$	$\mu_{5,11}$	$\mu_{5,12}$
$\mu_{6,1}$	$\mu_{6,2}$	$\mu_{6,3}$	$\mu_{6,4}$	$\mu_{6,5}$	$\mu_{6,6}$	$\mu_{6,7}$	$\mu_{6,8}$	$\mu_{6,9}$	$\mu_{6,10}$	$\mu_{6,11}$	$\mu_{6,12}$
$\mu_{7,1}$	$\mu_{7,2}$	$\mu_{7,3}$	$\mu_{7,4}$	$\mu_{7,5}$	$\mu_{7,6}$	$\mu_{7,7}$	$\mu_{7,8}$	$\mu_{7,9}$	$\mu_{7,10}$	$\mu_{7,11}$	$\mu_{7,12}$
$\mu_{8,1}$	$\mu_{8,2}$	$\mu_{8,3}$	$\mu_{8,4}$	$\mu_{8,5}$	$\mu_{8,6}$	$\mu_{8,7}$	$\mu_{8,8}$	$\mu_{8,9}$	$\mu_{8,10}$	$\mu_{8,11}$	$\mu_{8,12}$

$$x_1 \in \mathbb{R}^{D_{x_1}}, i - \text{th element of } x_1 = x_{1,i}$$

$$x_2 \in \mathbb{R}^{D_{x_2}}, i - \text{th element of } x_2 = x_{2,i}$$

$$y,\ \mu \in \mathbb{R}^{D_{x_1} \times D_{x_2}}$$

DEFINE PROBABILITY MODEL

$$y_{q,m} \sim \mathrm{Poisson}(\mu_{q,m})$$

$$\log \mu \sim \mathrm{MatrixNormal}(0, K_{\theta_1}(x_1), K_{\theta_2}(x_2))$$

$$x_1 \in \mathbb{R}^{D_{x_1}},\ x_2 \in \mathbb{R}^{D_{x_2}},\ \log \mu \in \mathbb{R}^{D_{x_1} \times D_{x_2}}$$

DEFINE PROBABILITY MODEL

$$K_{\theta_1}(x_1)$$

Intra-column covariance

$$K_{\theta_2}(x_2)$$

Intra-row covariance

DEFINE PROBABILITY MODEL

$$
\begin{aligned}
L_{x_1} &= \text{cholesky\_decompose}(K_{\theta_1}(x_1)) \\
L_{x_2} &= \text{cholesky\_decompose}(K_{\theta_2}(x_2)) \\
\eta_{q,m} &\sim \text{Normal}(0, 1)\,\forall q, m \\
\log \mu &= L_{x_1} \eta L'_{x_2} \\
\log \mu &\sim \text{MatrixNormal}(0, K_{\theta_1}(x_1), K_{\theta_1}(x_2))
\end{aligned}
$$

$$\eta \in \mathbb{R}^{D_{x_1} \times D_{x_2}}$$

DEFINE PROBABILITY MODEL

```
functions {
  matrix gp_exp_quad_chol(real[] x, real alpha,
                          real len, real jitter) {
    int dim_x = size(x);
    matrix[dim_x, dim_x] L_K_x;
    matrix[dim_x, dim_x] K_x = cov_exp_quad(x, alpha, len);
    for (n in 1:dim_x)
      K_x[n,n] = K_x[n,n] + jitter;
    L_K_x = cholesky_decompose(K_x);
    return L_K_x;
  }
}
```

DEFINE PROBABILITY MODEL

```
data {
  // Model is for outcomes that have been
  // observed on a 2 dimensional grid
  int<lower=1> dim_x_1; // size of grid dimension 1
  int<lower=1> dim_x_2; // size of grid dimension 2
  int y[dim_x_2,dim_x_1]; // Outcome
  real X_1[dim_x_1]; // locations - 1st dimension
  real X_2[dim_x_2]; // locations - 2nd dimension
}
```

DEFINE PROBABILITY MODEL

```
parameters {
  real<lower=0> len_scale_x_1; // length-scale - 1st dimension
  real<lower=0> len_scale_x_2; // length-scale - 2nd dimension
  real<lower=0> alpha; // Scale of outcomes
 // Standardized latent GP
  matrix[dim_x_1, dim_x_2] eta;
}
```

DEFINE PROBABILITY MODEL

```
model {
  matrix[dim_x_1, dim_x_2] latent_gp;
  {
    matrix[dim_x_1, dim_x_1] L_K_x_1 =
gp_exp_quad_chol(x_1, 1.0, len_scale_x_1, 1e-12);
    matrix[dim_x_2, dim_x_2] L_K_x_2 =
gp_exp_quad_chol(x_2, alpha, len_scale_x_2, 1e-12);

    // latent_gp is matrix-normal with within-
column covariance K_x_1
    // within-row covariance K_x_2

    latent_gp = L_K_x_1 * eta * L_K_x_2';
  }
  // priors
  len_scale_x_1 ~ gamma(8, 2);
  len_scale_x_2 ~ gamma(8, 2);
  alpha ~ normal(0, 1);
  to_vector(eta) ~ normal(0, 1);

  // likelihood
  to_array_1d(y) ~
poisson_log(to_vector(latent_gp));
}
```

DATA STRUCTURE

x_2 dimension

x_1 dimension

$y_{1,1,1}$	$y_{1,2,1}$	$y_{1,3,1}$	$y_{1,4,1}$	$y_{1,5,1}$	$y_{1,6,1}$	$y_{1,7,1}$	$y_{1,8,1}$	$y_{1,9,1}$	$y_{1,10,1}$	$y_{1,11,1}$	$y_{1,12,1}$
$y_{2,1,1}$	$y_{2,2,1}$	$y_{2,3,1}$	$y_{2,4,1}$	$y_{2,5,1}$	$y_{2,6,1}$	$y_{2,7,1}$	$y_{2,8,1}$	$y_{2,9,1}$	$y_{2,10,1}$	$y_{2,11,1}$	$y_{2,12,1}$
$y_{3,1,1}$	$y_{3,2,1}$	$y_{3,3,1}$	$y_{3,4,1}$	$y_{3,5,1}$	$y_{3,6,1}$	$y_{3,7,1}$	$y_{3,8,1}$	$y_{3,9,1}$	$y_{3,10,1}$	$y_{3,11,1}$	$y_{3,12,1}$
$y_{4,1,1}$	$y_{4,2,1}$	$y_{4,3,1}$	$y_{4,4,1}$	$y_{4,5,1}$	$y_{4,6,1}$	$y_{4,7,1}$	$y_{4,8,1}$	$y_{4,9,1}$	$y_{4,10,1}$	$y_{4,11,1}$	$y_{4,12,1}$
$y_{5,1,1}$	$y_{5,2,1}$	$y_{5,3,1}$	$y_{5,4,1}$	$y_{5,5,1}$	$y_{5,6,1}$	$y_{5,7,1}$	$y_{5,8,1}$	$y_{5,9,1}$	$y_{5,10,1}$	$y_{5,11,1}$	$y_{5,12,1}$
$y_{6,1,1}$	$y_{6,2,1}$	$y_{6,3,1}$	$y_{6,4,1}$	$y_{6,5,1}$	$y_{6,6,1}$	$y_{6,7,1}$	$y_{6,8,1}$	$y_{6,9,1}$	$y_{6,10,1}$	$y_{6,11,1}$	$y_{6,12,1}$
$y_{7,1,1}$	$y_{7,2,1}$	$y_{7,3,1}$	$y_{7,4,1}$	$y_{7,5,1}$	$y_{7,6,1}$	$y_{7,7,1}$	$y_{7,8,1}$	$y_{7,9,1}$	$y_{7,10,1}$	$y_{7,11,1}$	$y_{7,12,1}$
$y_{8,1,1}$	$y_{8,2,1}$	$y_{8,3,1}$	$y_{8,4,1}$	$y_{8,5,1}$	$y_{8,6,1}$	$y_{8,7,1}$	$y_{8,8,1}$	$y_{8,9,1}$	$y_{8,10,1}$	$y_{8,11,1}$	$y_{8,12,1}$

DATA STRUCTURE

x_2 dimension

x_1 dimension

$y_{1,1,1}$	$y_{1,2,1}$	$y_{1,3,1}$	$y_{1,4,1}$	$y_{1,5,1}$	$y_{1,6,1}$	$y_{1,7,1}$	$y_{1,8,1}$	$y_{1,9,1}$	$y_{1,10,1}$	$y_{1,11,1}$	$y_{1,12,1}$
$y_{2,1,1}$	$y_{2,2,1}$	$y_{2,3,1}$	$y_{2,4,1}$	$y_{2,5,1}$	$y_{2,6,1}$	$y_{2,7,1}$	$y_{2,8,1}$	$y_{2,9,1}$	$y_{2,10,1}$	$y_{2,11,1}$	$y_{2,12,1}$
$y_{3,1,1}$	$y_{3,2,1}$	$y_{3,3,}$	$y_{3,4,1}$	$y_{3,5,1}$	$y_{3,6,1}$	$y_{3,7,1}$	$y_{3,8,1}$	$y_{3,9,1}$	$y_{3,10,1}$	$y_{3,11,1}$	$y_{3,12,1}$
$y_{4,1,1}$	$y_{4,2,1}$	$y_{4,3,1}$	$y_{4,4,1}$	$y_{4,5,1}$	$y_{4,6,1}$	$y_{4,7,1}$	$y_{4,8,1}$	$y_{4,9,1}$	$y_{4,10,1}$	$y_{4,11,1}$	$y_{4,12,1}$
$y_{5,1,1}$	$y_{5,2,1}$	$y_{5,3,1}$	$y_{5,4,1}$	$y_{5,5,1}$	$y_{5,6,1}$	$y_{5,7,1}$	$y_{5,8,1}$	$y_{5,9,1}$	$y_{5,10,1}$	$y_{5,11,1}$	$y_{5,12,1}$
$y_{6,1,1}$	$y_{6,2,1}$	$y_{6,3,1}$	$y_{6,4,1}$	$y_{6,5,}$	[illegible]	[illegible]	$y_{6,8,1}$	$y_{6,9,1}$	$y_{6,10,1}$	$y_{6,11,1}$	$y_{6,12,1}$
$y_{7,1,1}$	$y_{7,2,1}$	$y_{7,3,1}$	$y_{7,4,1}$	$y_{7,5,1}$	$y_{7,6,1}$	$y_{7,7,1}$	$y_{7,8,1}$	$y_{7,9,1}$	$y_{7,10,1}$	$y_{7,11,1}$	$y_{7,12,1}$
$y_{8,1,1}$	$y_{8,2,1}$	$y_{8,3,1}$	$y_{8,4,1}$	$y_{8,5,1}$	$y_{8,6,1}$	$y_{8,7,1}$	$y_{8,8,1}$	$y_{8,9,1}$	$y_{8,10,1}$	$y_{8,11,1}$	$y_{8,12,1}$

T-direction

DATA STRUCTURE

x_2 dimension

x_1 dimension

$y_{1,1,1}$	$y_{1,2,1}$	$y_{1,3,1}$	$y_{1,4,1}$	$y_{1,5,1}$	$y_{1,6,1}$	$y_{1,7,1}$	$y_{1,8,1}$	$y_{1,9,1}$	$y_{1,10,1}$	$y_{1,11,1}$	$y_{1,12,1}$
$y_{2,1,1}$											
$y_{3,1,1}$											
$y_{4,1,1}$											
$y_{5,1,1}$											
$y_{6,1,1}$											
$y_{7,1,1}$											
$y_{8,1,1}$											

$y_{1,1,2}$	$y_{1,2,2}$	$y_{1,3,2}$	$y_{1,4,2}$	$y_{1,5,2}$	$y_{1,6,2}$	$y_{1,7,2}$	$y_{1,8,2}$	$y_{1,9,2}$	$y_{1,10,2}$	$y_{1,11,2}$	$y_{1,12,2}$
$y_{2,1,2}$	$y_{2,2,2}$	$y_{2,3,2}$	$y_{2,4,2}$	$y_{2,5,2}$	$y_{2,6,2}$	$y_{2,7,2}$	$y_{2,8,2}$	$y_{2,9,2}$	$y_{2,10,2}$	$y_{2,11,2}$	$y_{2,12,2}$
$y_{3,1,2}$	$y_{3,2,2}$	$y_{3,3,2}$	$y_{3,4,2}$	$y_{3,5,2}$	$y_{3,6,2}$	$y_{3,7,2}$	$y_{3,8,2}$	$y_{3,9,2}$	$y_{3,10,2}$	$y_{3,11,2}$	$y_{3,12,2}$
$y_{4,1,2}$	$y_{4,2,2}$	$y_{4,3,2}$	$y_{4,4,2}$	$y_{4,5,2}$	$y_{4,6,2}$	$y_{4,7,2}$	$y_{4,8,2}$	$y_{4,9,2}$	$y_{4,10,2}$	$y_{4,11,2}$	$y_{4,12,2}$
$y_{5,1,2}$	$y_{5,2,2}$	$y_{5,3,2}$	$y_{5,4,2}$	$y_{5,5,2}$	$y_{5,6,2}$	$y_{5,7,2}$	$y_{5,8,2}$	$y_{5,9,2}$	$y_{5,10,2}$	$y_{5,11,2}$	$y_{5,12,2}$
$y_{6,1,2}$	$y_{6,2,2}$	$y_{6,3,2}$	$y_{6,4,2}$	[illegible]	[illegible]	$y_{6,7,2}$	$y_{6,8,2}$	$y_{6,9,2}$	$y_{6,10,2}$	$y_{6,11,2}$	$y_{6,12,2}$
$y_{7,1,2}$	$y_{7,2,2}$	$y_{7,3,2}$	$y_{7,4,2}$	$y_{7,5,2}$	$y_{7,6,2}$	$y_{7,7,2}$	$y_{7,8,2}$	$y_{7,9,2}$	$y_{7,10,2}$	$y_{7,11,2}$	$y_{7,12,2}$
$y_{8,1,2}$	$y_{8,2,2}$	$y_{8,3,2}$	$y_{8,4,2}$	$y_{8,5,2}$	$y_{8,6,2}$	$y_{8,7,2}$	$y_{8,8,2}$	$y_{8,9,2}$	$y_{8,10,2}$	$y_{8,11,2}$	$y_{8,12,2}$

T-direction

DATA STRUCTURE

DEFINE PROBABILITY MODEL

$$y_{q,m,t} \sim \mathrm{Poisson}(\mu_{q,m,t})$$

$$\log \mu \sim \mathrm{ArrayNormal}(0, K_{\theta_1}(x_1), K_{\theta_2}(x_2), K_{\theta_T}(T))$$

$$x_1 \in \mathbb{R}^{D_{x_1}},\ x_2 \in \mathbb{R}^{D_{x_2}},\ T \in \mathbb{R}^{D_T},\ \log \mu \in \mathbb{R}^{D_{x_1} \times D_{x_2} \times D_T}$$

DEFINE PROBABILITY MODEL

$$
\begin{aligned}
L_{x_1} &= \text{cholesky\_decompose}(K_{\theta_1}(x_1)) \\
L_{x_2} &= \text{cholesky\_decompose}(K_{\theta_2}(x_2)) \\
L_T &= \text{cholesky\_decompose}(K_{\theta_T}(T)) \\
\eta_{q,m,t} &\sim \text{Normal}(0,1)\;\forall q,m,t \\
\log\mu[,m,] &= \eta[,j,]\,L_T'\;\forall m \\
\log\mu[,,t] &= L_{x_1}\;\log\mu[,,t]\,L_{x_2}'\;\forall t \\
\log\mu &\sim \text{ArrayNormal}(0, K_{\theta_1}(x_1), K_{\theta_2}(x_2), K_{\theta_T}(T)) \\
\eta &\in \mathbb{R}^{D_{x_1}\times D_{x_2}\times D_T}
\end{aligned}
$$

DEFINE PROBABILITY MODEL

```
data {
  // Model is for outcomes that have been
  // observed on a 3 dimensional grid
  int<lower=1> dim_x_1; // size of grid dimension 1
  int<lower=1> dim_x_2; // size of grid dimension 2
  int<lower=1> dim_T; // size of grid dimension 3
  int y[dim_x_1, dim_x_2, dim_T]; // Outcome
  real x_1[dim_x_1]; // locations - 1st dimension
  real x_2[dim_x_2]; // locations - 2nd dimension
  real T[dim_T]; // locations - 3rd dimension
}
```

DEFINE PROBABILITY MODEL

```
parameters {
  real<lower=0> len_scale_x_1; // length-scale - 1st dimension
  real<lower=0> len_scale_x_2; // length-scale - 2nd dimension
  real<lower=0> len_scale_T; // length-scale - 3rd dimension
  real<lower=0> alpha_x_1_x2; // Scale of outcomes x1 x2 plane
  real<lower=0> alpha_T; // Scale of outcomes on T
  // Standardized latent GP
  real eta[dim_x_1, dim_x_2, dim_T];
}
```

DEFINE PROBABILITY MODEL

```
model {
  real latent_gp[dim_x_1, dim_x_2, dim_T];
  {
    matrix[dim_x_1, dim_x_1] L_K_x_1 = gp_exp_quad_chol(x_1,
alpha_x_1_x_2, len_scale_x_1, 1e-12);
    matrix[dim_x_2, dim_x_2] t_L_K_x_2 = gp_exp_quad_chol(x_2,
1.0, len_scale_x_2, 1e-12)';
    matrix[dim_T, dim_T] t_L_K_T = gp_exp_quad_chol(T, alpha_T,
len_scale_T, 1e-12)';

    // latent_gp is matrix-normal with among-column covariance
K_x_1
    // among-row covariance K_x_2
    for (m in 1:dim_x_2)
      latent_gp[,m,] = to_array_2d(to_matrix(eta[,m,]) * t_L_K_T);

    for (t in 1:dim_T)
      latent_gp[,,t] = to_array_2d(L_K_x_1 *
to_matrix(latent_gp[,,t]) * t_L_K_x_2);
  }
  // priors
  len_scale_x_1 ~ gamma(8, 2);
  len_scale_x_2 ~ gamma(8, 2);
  len_scale_T ~ gamma(8, 2);
  alpha_T ~ normal(0, 1);
  alpha_x_1_x_2 ~ normal(0, 1);

  // likelihood
  to_array_1d(eta) ~ normal(0, 1);
  to_array_1d(y) ~ poisson_log(to_array_1d(latent_gp));
}
```

KEY TAKEAWAYS FOR TODAY

- What Stan does and why you should use it
- Gaussian processes and when they're useful
- The importance of thinking about the scale of your hyper parameters
- Hierarchical Gaussian processes
- Spatio-temporal GPs in Stan

FUTURE OF GAUSSIAN PROCESSES IN STAN

MORE VARIETY!

- More kernels!
- Fast decompositions for structured matrices
- Sparse matrix support

THANKS

- To Stan dev-team: http://mc-stan.org/team/
- Especially Jonah Gabry and Daniel Lee who supplied some slides for this talk and Michael Betancourt and Aki Vehtari who've been working with me on GPs
- YouGov