# BtvGARCH-Ito Model for Volatility Analysis

AUTHOR
Pathairat Pastpipatkul & Htwe Ko

PUBLISHED
January 1, 2025

MODIFIED
June 1, 2025

## Load Packages

```r
library(ggplot2)
library(dplyr)
library(tidyr)
library(cmdstanr)
library(posterior)
```

## Fix Toolchain

```r
cat("Checking and fixing the CmdStan toolchain...\n")
```

```
Checking and fixing the CmdStan toolchain...
```

```r
cmdstanr::check_cmdstan_toolchain(fix = TRUE)
```

```
The C++ toolchain required for CmdStan is setup properly!
```

## Install CMDstanR

```r
cmdstan_path <- get0("CMDSTAN_HOME", envir = as.environment("package:cmdstanr"))
if (is.null(cmdstan_path)) {
  cat("CmdStan not found. Installing now...\n")
  install_cmdstan()
  cmdstan_path <- get0("CMDSTAN_HOME", envir = as.environment("package:cmdstanr"))
}
```

```
CmdStan not found. Installing now...

The C++ toolchain required for CmdStan is setup properly!

* Latest CmdStan release is v2.36.0

* Installing CmdStan v2.36.0 in C:\Users\User/.cmdstan/cmdstan-2.36.0

* Downloading cmdstan-2.36.0.tar.gz from GitHub...
```

```
Warning: An installation already exists at
C:\Users\User/.cmdstan/cmdstan-2.36.0. Please remove or rename the installation
folder or set overwrite=TRUE.
```

```r
cat("CmdStan is located at:", cmdstan_path, "\n")
```

```
CmdStan is located at:
```

## Define BtvGARCH-Ito Model

```r
model_code <- "
functions {
  real my_fmax(real x, real y) {
    return x > y ? x : y;
  }
}

data {
  int<lower=1> N;
  vector[N] RV;
  vector[N] JV;
  real lambda_omega_L;
}

parameters {
  real<lower=0, upper=1> omega_0;
  real<lower=0, upper=1> alpha_0;
  real<lower=0, upper=1> beta_0;
  real<lower=0, upper=1> gamma_0;

  real<lower=0> sigma_omega;
  real<lower=0> sigma_alpha;
  real<lower=0> sigma_beta;
  real<lower=0> sigma_gamma;
  real<lower=0> sigma_RV;

  vector[N] omega_tilde;
  vector[N] alpha_tilde;
  vector[N] beta_tilde;
  vector[N] gamma_tilde;
}

transformed parameters {
  vector[N] omega_raw;
  vector[N] alpha_raw;
  vector[N] beta_raw;
  vector[N] gamma_raw;
```

```
  omega_raw[1] = logit(omega_0) + 0.01 * omega_tilde[1];
  alpha_raw[1] = logit(alpha_0) + 0.01 * alpha_tilde[1];
  beta_raw[1]  = logit(beta_0)  + 0.01 * beta_tilde[1];
  gamma_raw[1] = logit(gamma_0) + 0.01 * gamma_tilde[1];

  for (t in 2:N) {
    omega_raw[t] = omega_raw[t-1] + sigma_omega * omega_tilde[t];
    alpha_raw[t] = alpha_raw[t-1] + sigma_alpha * alpha_tilde[t];
    beta_raw[t]  = beta_raw[t-1]  + sigma_beta  * beta_tilde[t];
    gamma_raw[t] = gamma_raw[t-1] + sigma_gamma * gamma_tilde[t];
  }

  vector<lower=0, upper=1>[N] omega_g = inv_logit(omega_raw);
  vector<lower=0, upper=1>[N] alpha_g = inv_logit(alpha_raw);
  vector<lower=0, upper=1>[N] beta_g  = inv_logit(beta_raw);
  vector<lower=0, upper=1>[N] gamma   = inv_logit(gamma_raw);

  vector[N] h;
  real eps = 1e-6;
  h[1] = (omega_g[1] + beta_g[1] * lambda_omega_L) / my_fmax(eps, 1 - alpha_g[1] - gamma[1]);

  for (t in 2:N) {
    h[t] = omega_g[t] + gamma[t] * h[t-1] + alpha_g[t] * RV[t-1] + beta_g[t] * JV[t-1];
  }
}

model {
  omega_0 ~ normal(0.3, 0.1);
  alpha_0 ~ normal(0.1, 0.05);
  beta_0  ~ normal(0.05, 0.02);
  gamma_0 ~ normal(0.6, 0.05);

  sigma_omega ~ cauchy(0, 0.1);
  sigma_alpha ~ cauchy(0, 0.1);
  sigma_beta  ~ cauchy(0, 0.1);
  sigma_gamma ~ cauchy(0, 0.1);
  sigma_RV    ~ cauchy(0, 0.2);

  omega_tilde ~ std_normal();
  alpha_tilde ~ std_normal();
  beta_tilde  ~ std_normal();
  gamma_tilde ~ std_normal();

  RV ~ normal(h, sigma_RV);
}

generated quantities {
  vector[N] omega_g_save = inv_logit(omega_raw);
  vector[N] alpha_g_save = inv_logit(alpha_raw);
  vector[N] beta_g_save  = inv_logit(beta_raw);
  vector[N] gamma_save   = inv_logit(gamma_raw);
```

```
}
"

write(model_code, "model_ncp.stan")
```

## Data Simulation

```r
set.seed(123)
n <- 100
t <- 1:n
JV <- runif(n, 0, 0.2)
lambda_omega_L <- median(JV)

omega_true <- 0.3 + 0.1 * sin(2 * pi * t / n)
alpha_true <- 0.1 + 0.05 * sin(4 * pi * t / n)
beta_true  <- 0.05 + 0.03 * sin(6 * pi * t / n)
gamma_true <- 0.6 + 0.05 * cos(2 * pi * t / n)

h_true <- numeric(n)
h_true[1] <- (omega_true[1] + beta_true[1] * lambda_omega_L) /
  max(1e-6, 1 - alpha_true[1] - gamma_true[1])

for (i in 2:n) {
  h_true[i] <- omega_true[i] + gamma_true[i] * h_true[i-1] +
    alpha_true[i] * h_true[i-1] + beta_true[i] * JV[i-1]
}

RV <- rnorm(n, mean = h_true, sd = 0.1)

data_list <- list(N = n, RV = RV, JV = JV, lambda_omega_L = lambda_omega_L)
```

## Compile the mode

```r
mod <- cmdstan_model("model_ncp.stan")
print(mod)
```

```
functions {
  real my_fmax(real x, real y) {
    return x > y ? x : y;
  }
}

data {
  int<lower=1> N;
  vector[N] RV;
```

```
  vector[N] JV;
  real lambda_omega_L;
}

parameters {
  real<lower=0, upper=1> omega_0;
  real<lower=0, upper=1> alpha_0;
  real<lower=0, upper=1> beta_0;
  real<lower=0, upper=1> gamma_0;

  real<lower=0> sigma_omega;
  real<lower=0> sigma_alpha;
  real<lower=0> sigma_beta;
  real<lower=0> sigma_gamma;
  real<lower=0> sigma_RV;

  vector[N] omega_tilde;
  vector[N] alpha_tilde;
  vector[N] beta_tilde;
  vector[N] gamma_tilde;
}

transformed parameters {
  vector[N] omega_raw;
  vector[N] alpha_raw;
  vector[N] beta_raw;
  vector[N] gamma_raw;

  omega_raw[1] = logit(omega_0) + 0.01 * omega_tilde[1];
  alpha_raw[1] = logit(alpha_0) + 0.01 * alpha_tilde[1];
  beta_raw[1]  = logit(beta_0)  + 0.01 * beta_tilde[1];
  gamma_raw[1] = logit(gamma_0) + 0.01 * gamma_tilde[1];

  for (t in 2:N) {
    omega_raw[t] = omega_raw[t-1] + sigma_omega * omega_tilde[t];
    alpha_raw[t] = alpha_raw[t-1] + sigma_alpha * alpha_tilde[t];
    beta_raw[t]  = beta_raw[t-1]  + sigma_beta  * beta_tilde[t];
    gamma_raw[t] = gamma_raw[t-1] + sigma_gamma * gamma_tilde[t];
  }

  vector<lower=0, upper=1>[N] omega_g = inv_logit(omega_raw);
  vector<lower=0, upper=1>[N] alpha_g = inv_logit(alpha_raw);
  vector<lower=0, upper=1>[N] beta_g  = inv_logit(beta_raw);
  vector<lower=0, upper=1>[N] gamma   = inv_logit(gamma_raw);

  vector[N] h;
  real eps = 1e-6;
  h[1] = (omega_g[1] + beta_g[1] * lambda_omega_L) / my_fmax(eps, 1 - alpha_g[1] - gamma[1]);

  for (t in 2:N) {
    h[t] = omega_g[t] + gamma[t] * h[t-1] + alpha_g[t] * RV[t-1] + beta_g[t] * JV[t-1];
```

```
  }
}

model {
  omega_0 ~ normal(0.3, 0.1);
  alpha_0 ~ normal(0.1, 0.05);
  beta_0  ~ normal(0.05, 0.02);
  gamma_0 ~ normal(0.6, 0.05);

  sigma_omega ~ cauchy(0, 0.1);
  sigma_alpha ~ cauchy(0, 0.1);
  sigma_beta  ~ cauchy(0, 0.1);
  sigma_gamma ~ cauchy(0, 0.1);
  sigma_RV    ~ cauchy(0, 0.2);

  omega_tilde ~ std_normal();
  alpha_tilde ~ std_normal();
  beta_tilde  ~ std_normal();
  gamma_tilde ~ std_normal();

  RV ~ normal(h, sigma_RV);
}

generated quantities {
  vector[N] omega_g_save = inv_logit(omega_raw);
  vector[N] alpha_g_save = inv_logit(alpha_raw);
  vector[N] beta_g_save  = inv_logit(beta_raw);
  vector[N] gamma_save   = inv_logit(gamma_raw);
}
```

## Fit the model

```r
fit <- mod$sample(
  data = data_list,
  seed = 123,
  chains = 4,
  parallel_chains = 4,
  iter_warmup = 500,
  iter_sampling = 1000,
  adapt_delta = 0.95,
  max_treedepth = 12
)
```

```
Running MCMC with 4 parallel chains...

Chain 1 Iteration:    1 / 1500 [  0%]  (Warmup)

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:
```

```
Chain 1 Exception: model_ncp_model_namespace::log_prob: alpha_g[12] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 52, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: alpha_g[15] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 52, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: alpha_g[9] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 52, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[13] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,
```

```
Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[11] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[11] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[11] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1
```

```
Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[8] is nan, but must be greater than
or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-3c0c653f46cb.stan',
line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[7] is nan, but must be greater than
or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-3c0c653f46cb.stan',
line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[7] is nan, but must be greater than
or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-3c0c653f46cb.stan',
line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[7] is nan, but must be greater than
or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-3c0c653f46cb.stan',
line 54, column 2 to column 61)
```

```
Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[7] is nan, but must be greater than
or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-3c0c653f46cb.stan',
line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[7] is nan, but must be greater than
or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-3c0c653f46cb.stan',
line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: alpha_g[11] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 52, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.
```

```
Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: alpha_g[11] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 52, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[11] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 1 Exception: model_ncp_model_namespace::log_prob: gamma[7] is nan, but must be greater than
or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-3c0c653f46cb.stan',
line 54, column 2 to column 61)

Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 1

Chain 2 Iteration:    1 / 1500 [  0%]  (Warmup)

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:
```

```
Chain 2 Exception: model_ncp_model_namespace::log_prob: beta_g[77] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 53, column 2 to column 60)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: beta_g[77] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 53, column 2 to column 60)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: omega_g[97] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 51, column 2 to column 61)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: omega_g[96] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 51, column 2 to column 61)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,
```

```
Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: omega_g[96] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 51, column 2 to column 61)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: omega_g[93] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 51, column 2 to column 61)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: omega_g[93] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 51, column 2 to column 61)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2
```

```
Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: omega_g[98] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 51, column 2 to column 61)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: omega_g[99] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 51, column 2 to column 61)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: omega_g[93] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 51, column 2 to column 61)

Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 2 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 2 Exception: model_ncp_model_namespace::log_prob: gamma[95] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 54, column 2 to column 61)
```

```
Chain 2 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 2 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 2

Chain 3 Iteration:    1 / 1500 [  0%]  (Warmup)

Chain 3 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 3 Exception: normal_lpdf: Scale parameter is 0, but must be positive! (in
'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-3c0c653f46cb.stan', line 82, column 2 to
column 27)

Chain 3 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 3 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 3

Chain 4 Iteration:    1 / 1500 [  0%]  (Warmup)

Chain 4 Informational Message: The current Metropolis proposal is about to be rejected because of
the following issue:

Chain 4 Exception: model_ncp_model_namespace::log_prob: omega_g[88] is nan, but must be greater
than or equal to 0.000000 (in 'C:/Users/User/AppData/Local/Temp/Rtmp06LtVH/model-
3c0c653f46cb.stan', line 51, column 2 to column 61)

Chain 4 If this warning occurs sporadically, such as for highly constrained variable types like
covariance matrices, then the sampler is fine,

Chain 4 but if this warning occurs often then your model may be either severely ill-conditioned
or misspecified.

Chain 4

Chain 1 Iteration:  100 / 1500 [  6%]  (Warmup)
Chain 1 Iteration:  200 / 1500 [ 13%]  (Warmup)
Chain 1 Iteration:  300 / 1500 [ 20%]  (Warmup)
Chain 3 Iteration:  100 / 1500 [  6%]  (Warmup)
Chain 1 Iteration:  400 / 1500 [ 26%]  (Warmup)
Chain 3 Iteration:  200 / 1500 [ 13%]  (Warmup)
Chain 1 Iteration:  500 / 1500 [ 33%]  (Warmup)
Chain 1 Iteration:  501 / 1500 [ 33%]  (Sampling)
Chain 2 Iteration:  100 / 1500 [  6%]  (Warmup)
```

```
Chain 3 Iteration:  300 / 1500 [ 20%]  (Warmup)
Chain 1 Iteration:  600 / 1500 [ 40%]  (Sampling)
Chain 3 Iteration:  400 / 1500 [ 26%]  (Warmup)
Chain 1 Iteration:  700 / 1500 [ 46%]  (Sampling)
Chain 3 Iteration:  500 / 1500 [ 33%]  (Warmup)
Chain 3 Iteration:  501 / 1500 [ 33%]  (Sampling)
Chain 2 Iteration:  200 / 1500 [ 13%]  (Warmup)
Chain 4 Iteration:  100 / 1500 [  6%]  (Warmup)
Chain 1 Iteration:  800 / 1500 [ 53%]  (Sampling)
Chain 2 Iteration:  300 / 1500 [ 20%]  (Warmup)
Chain 3 Iteration:  600 / 1500 [ 40%]  (Sampling)
Chain 1 Iteration:  900 / 1500 [ 60%]  (Sampling)
Chain 2 Iteration:  400 / 1500 [ 26%]  (Warmup)
Chain 4 Iteration:  200 / 1500 [ 13%]  (Warmup)
Chain 3 Iteration:  700 / 1500 [ 46%]  (Sampling)
Chain 1 Iteration: 1000 / 1500 [ 66%]  (Sampling)
Chain 2 Iteration:  500 / 1500 [ 33%]  (Warmup)
Chain 3 Iteration:  800 / 1500 [ 53%]  (Sampling)
Chain 2 Iteration:  501 / 1500 [ 33%]  (Sampling)
Chain 4 Iteration:  300 / 1500 [ 20%]  (Warmup)
Chain 1 Iteration: 1100 / 1500 [ 73%]  (Sampling)
Chain 2 Iteration:  600 / 1500 [ 40%]  (Sampling)
Chain 3 Iteration:  900 / 1500 [ 60%]  (Sampling)
Chain 4 Iteration:  400 / 1500 [ 26%]  (Warmup)
Chain 1 Iteration: 1200 / 1500 [ 80%]  (Sampling)
Chain 2 Iteration:  700 / 1500 [ 46%]  (Sampling)
Chain 3 Iteration: 1000 / 1500 [ 66%]  (Sampling)
Chain 4 Iteration:  500 / 1500 [ 33%]  (Warmup)
Chain 4 Iteration:  501 / 1500 [ 33%]  (Sampling)
Chain 1 Iteration: 1300 / 1500 [ 86%]  (Sampling)
Chain 2 Iteration:  800 / 1500 [ 53%]  (Sampling)
Chain 3 Iteration: 1100 / 1500 [ 73%]  (Sampling)
Chain 4 Iteration:  600 / 1500 [ 40%]  (Sampling)
Chain 1 Iteration: 1400 / 1500 [ 93%]  (Sampling)
Chain 2 Iteration:  900 / 1500 [ 60%]  (Sampling)
Chain 3 Iteration: 1200 / 1500 [ 80%]  (Sampling)
Chain 4 Iteration:  700 / 1500 [ 46%]  (Sampling)
Chain 1 Iteration: 1500 / 1500 [100%]  (Sampling)
Chain 1 finished in 40.8 seconds.
Chain 2 Iteration: 1000 / 1500 [ 66%]  (Sampling)
Chain 3 Iteration: 1300 / 1500 [ 86%]  (Sampling)
Chain 4 Iteration:  800 / 1500 [ 53%]  (Sampling)
Chain 2 Iteration: 1100 / 1500 [ 73%]  (Sampling)
Chain 3 Iteration: 1400 / 1500 [ 93%]  (Sampling)
Chain 4 Iteration:  900 / 1500 [ 60%]  (Sampling)
Chain 2 Iteration: 1200 / 1500 [ 80%]  (Sampling)
Chain 3 Iteration: 1500 / 1500 [100%]  (Sampling)
Chain 4 Iteration: 1000 / 1500 [ 66%]  (Sampling)
Chain 3 finished in 48.1 seconds.
Chain 2 Iteration: 1300 / 1500 [ 86%]  (Sampling)
Chain 4 Iteration: 1100 / 1500 [ 73%]  (Sampling)
```

```
Chain 2 Iteration: 1400 / 1500 [ 93%]  (Sampling)
Chain 4 Iteration: 1200 / 1500 [ 80%]  (Sampling)
Chain 2 Iteration: 1500 / 1500 [100%]  (Sampling)
Chain 2 finished in 55.8 seconds.
Chain 4 Iteration: 1300 / 1500 [ 86%]  (Sampling)
Chain 4 Iteration: 1400 / 1500 [ 93%]  (Sampling)
Chain 4 Iteration: 1500 / 1500 [100%]  (Sampling)
Chain 4 finished in 61.2 seconds.

All 4 chains finished successfully.
Mean chain execution time: 51.5 seconds.
Total execution time: 61.5 seconds.
Warning: 4 of 4000 (0.0%) transitions ended with a divergence.
See https://mc-stan.org/misc/warnings for details.
```

## print the summary results

```r
if ("CmdStanMCMC" %in% class(fit)) {
  summary_fit <- fit$summary()
  print(head(summary_fit, 10))  # Show first 10 rows
} else {
  stop("Fit object is invalid! Check compilation/sampling errors.")
}
```

```
# A tibble: 10 × 10
   variable       mean   median      sd     mad       q5      q95  rhat ess_bulk
   <chr>         <dbl>    <dbl>   <dbl>   <dbl>    <dbl>    <dbl> <dbl>    <dbl>
 1 lp__       -44.3    -44.0    1.48e+1 1.44e+1 -6.88e+1 -20.5     1.00    1214.
 2 omega_0      0.353    0.353  6.79e-2 6.95e-2  2.41e-1   0.466   1.00    2373.
 3 alpha_0      0.106    0.105  4.37e-2 4.55e-2  3.28e-2   0.177   1.00    2114.
 4 beta_0       0.0480   0.0477 1.96e-2 2.02e-2  1.52e-2   0.0799  1.00    3617.
 5 gamma_0      0.624    0.623  4.34e-2 4.53e-2  5.54e-1   0.695   1.00    4715.
 6 sigma_ome…   0.0522   0.0485 3.32e-2 3.64e-2  5.50e-3   0.111   1.00     964.
 7 sigma_alp…   0.0954   0.0666 1.33e-1 6.00e-2  6.82e-3   0.252   1.00    1491.
 8 sigma_beta   0.249    0.0830 1.22e+0 8.91e-2  6.33e-3   0.838   1.00    4782.
 9 sigma_gam…   0.0544   0.0528 3.25e-2 3.38e-2  6.31e-3   0.111   1.00     837.
10 sigma_RV     0.0991   0.0989 8.26e-3 8.08e-3  8.66e-2   0.114   1.00    3684.
# ℹ 1 more variable: ess_tail <dbl>
```

## Check Diagnositcs

```r
fit$diagnostic_summary()
```

```
Warning: 4 of 4000 (0.0%) transitions ended with a divergence.
See https://mc-stan.org/misc/warnings for details.
```

```
$num_divergent
[1] 3 0 1 0

$num_max_treedepth
[1] 0 0 0 0

$ebfmi
[1] 0.9795583 1.0507203 0.9310035 1.0299116
```

```r
fit$cmdstan_diagnose()
```

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

Checking sampler transitions for divergences.
4 of 1000 (0.40%) transitions ended with a divergence.
These divergent transitions indicate that HMC is not fully able to explore the posterior
distribution.
Try increasing adapt delta closer to 1.
If this doesn't remove all divergences, try to reparameterize the model.

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

Rank-normalized split effective sample size satisfactory for all parameters.

Rank-normalized split R-hat values satisfactory for all parameters.

Processing complete.
```

# Extract only time-varying parameters

```r
params <- c("omega_g_save", "alpha_g_save", "beta_g_save", "gamma_save", "h")
draws_df <- as_draws_df(fit$draws(variables = params))
```

# Convert to long format with parameter name and time

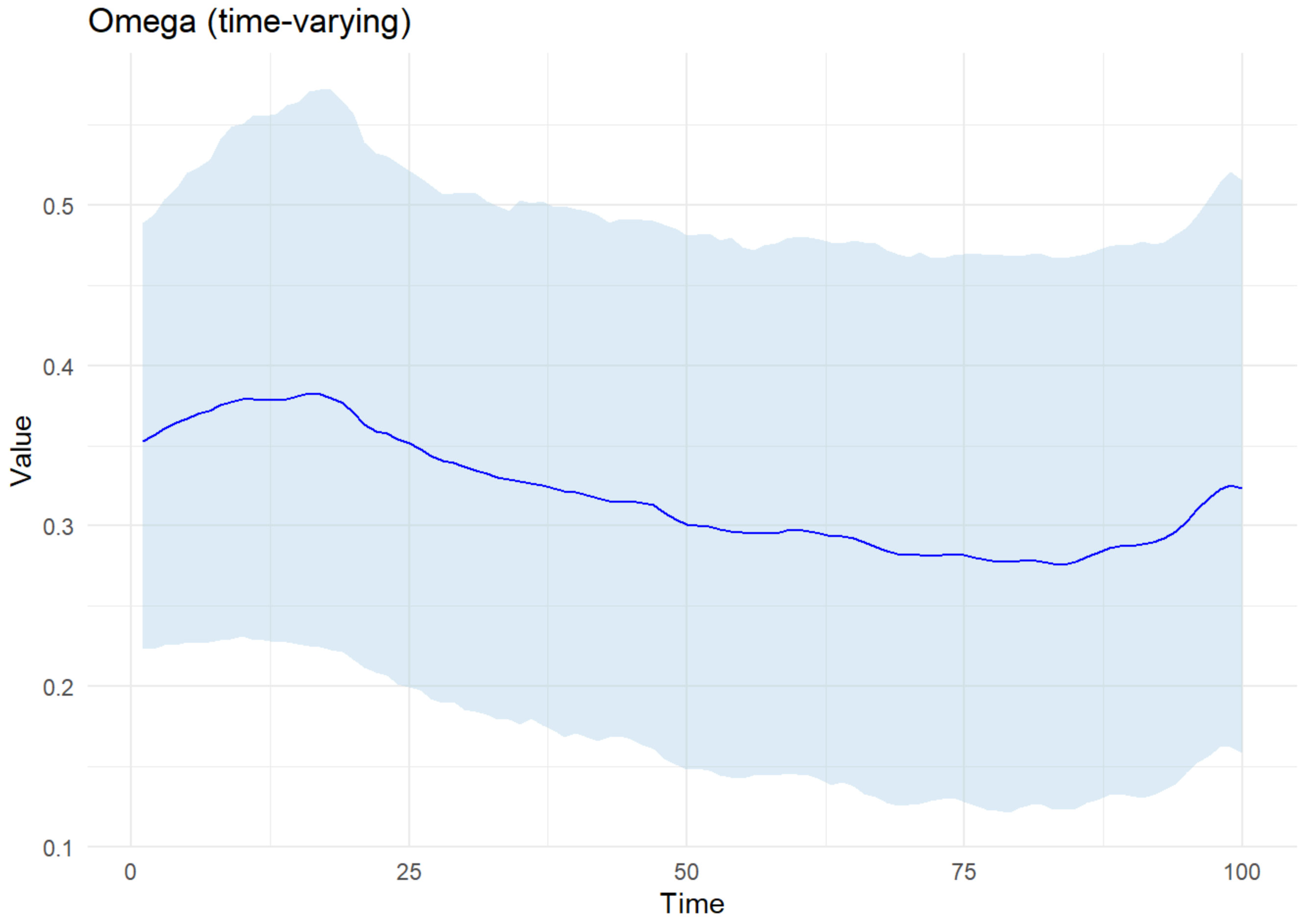
```r
draws_long <- draws_df %>%
  pivot_longer(cols = everything(), names_to = "param", values_to = "value") %>%
  mutate(
    time = as.integer(gsub(".*\\[|\\]", "", param)),
    name = sub("\\[.*", "", param)
  )
```

```
Warning: There was 1 warning in `mutate()`.
ℹ In argument: `time = as.integer(gsub(".*\\[|\\]", "", param))`.
```

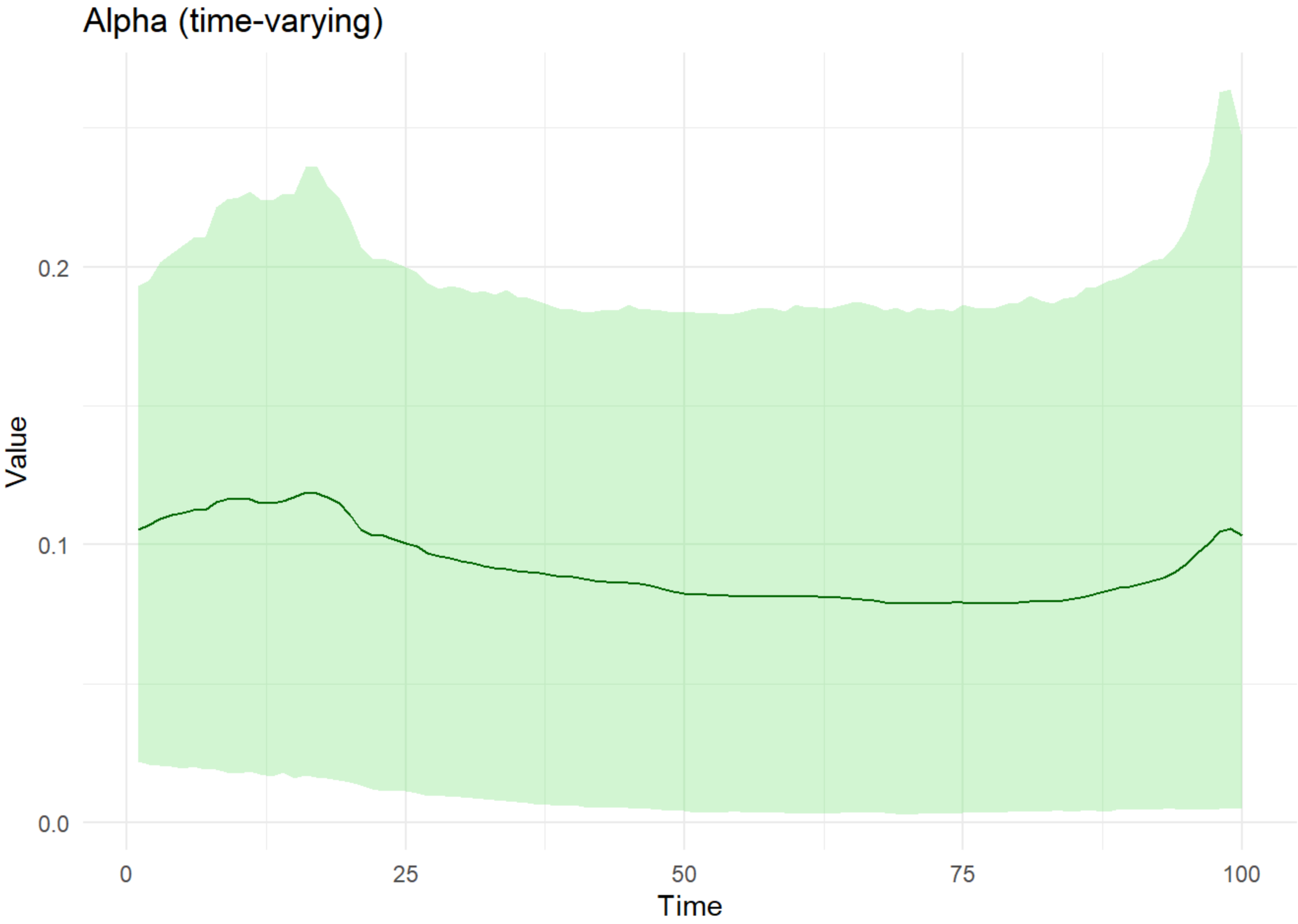
```
Caused by warning:
! NAs introduced by coercion
```

## Compute posterior means and 95% credible intervals

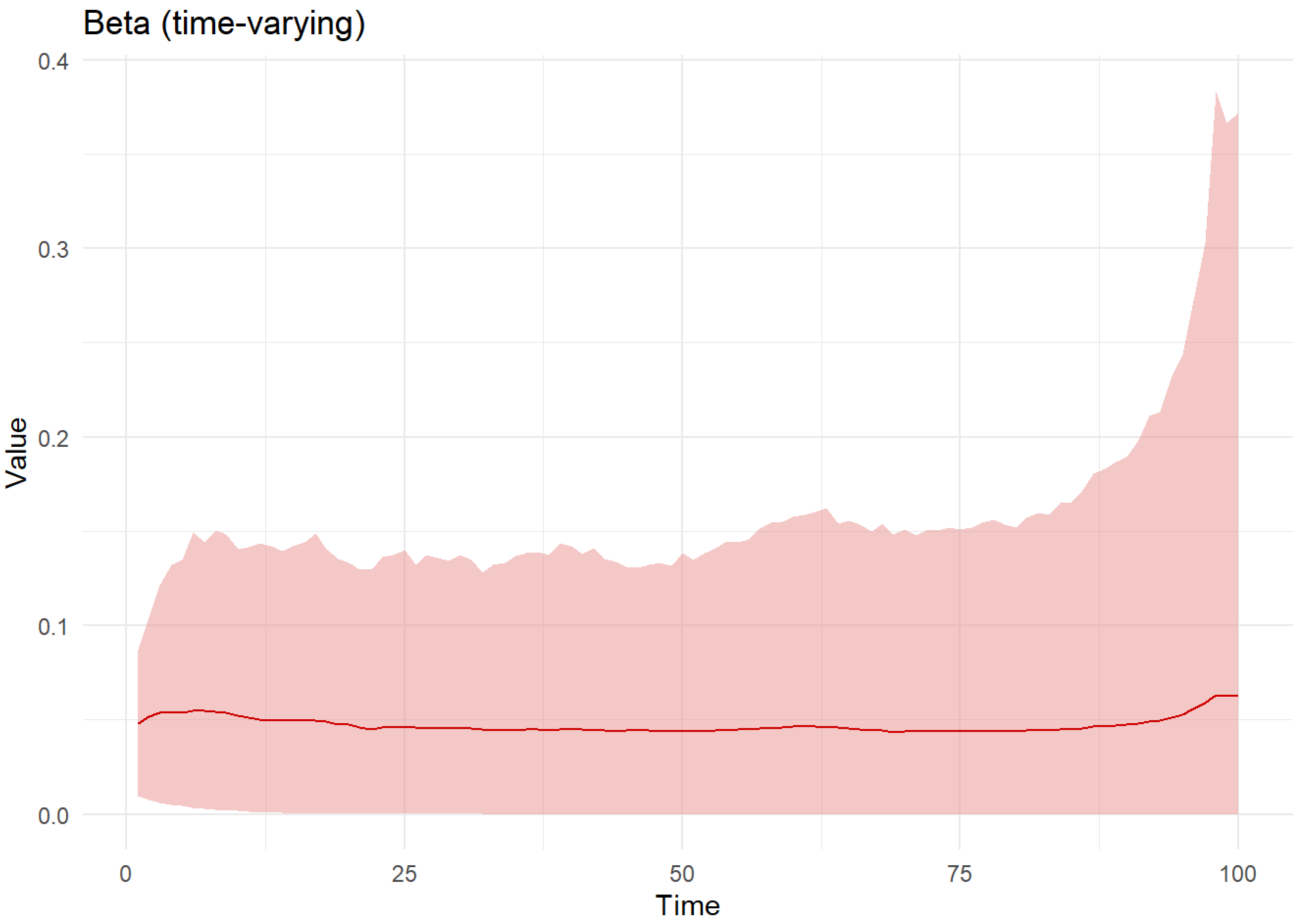
```
summary_stats <- draws_long %>%
  group_by(name, time) %>%
  summarise(
    mean = mean(value),
    lower = quantile(value, 0.025),
    upper = quantile(value, 0.975),
    .groups = "drop"
  )


ggplot(filter(summary_stats, name == "omega_g_save"), aes(x = time, y = mean)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "lightblue", alpha = 0.4) +
  geom_line(color = "blue") +
  labs(title = "Omega (time-varying)", x = "Time", y = "Value") +
  theme_minimal()
```



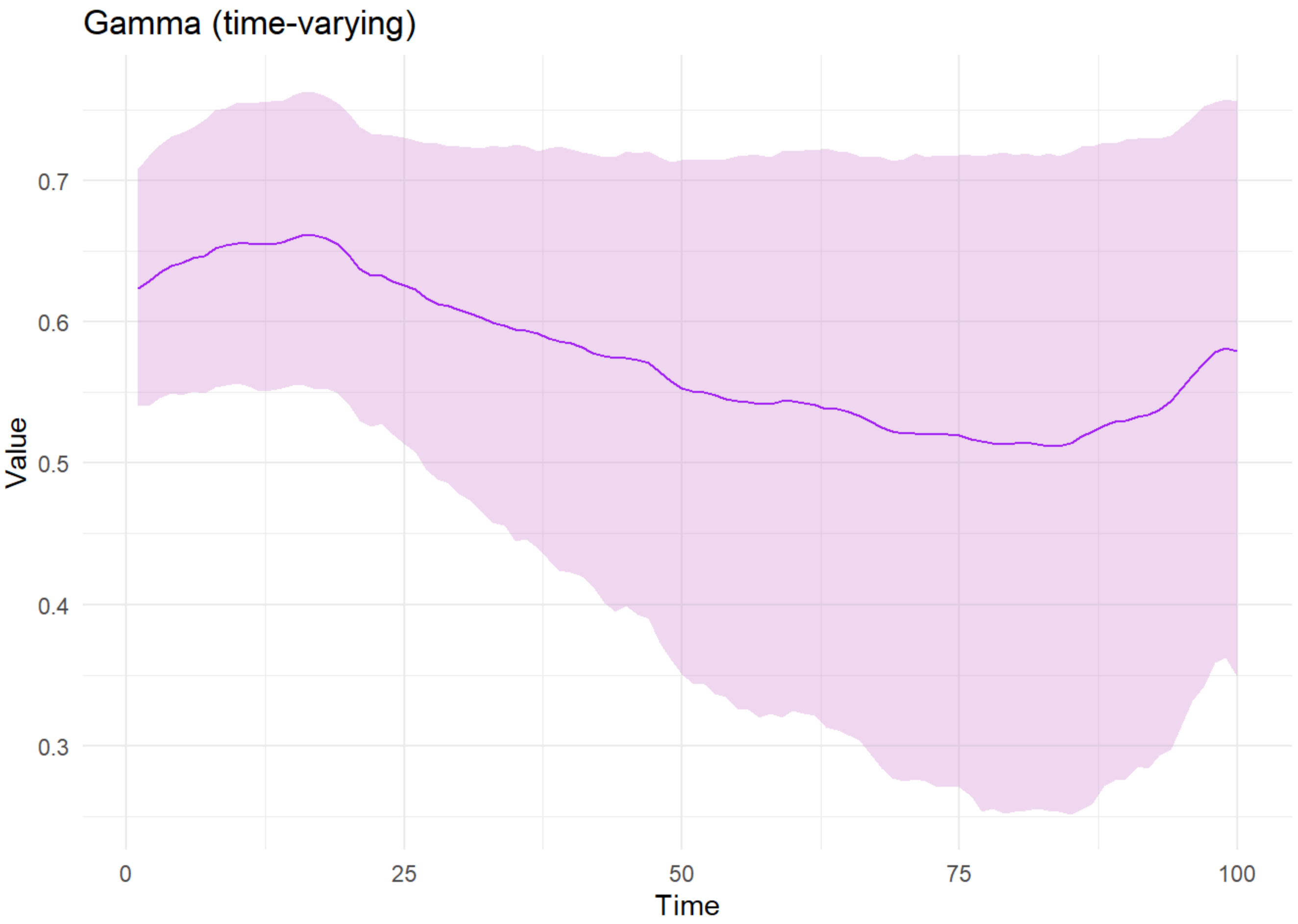
```
ggplot(filter(summary_stats, name == "alpha_g_save"), aes(x = time, y = mean)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "lightgreen", alpha = 0.4) +
  geom_line(color = "darkgreen") +
  labs(title = "Alpha (time-varying)", x = "Time", y = "Value") +
  theme_minimal()
```



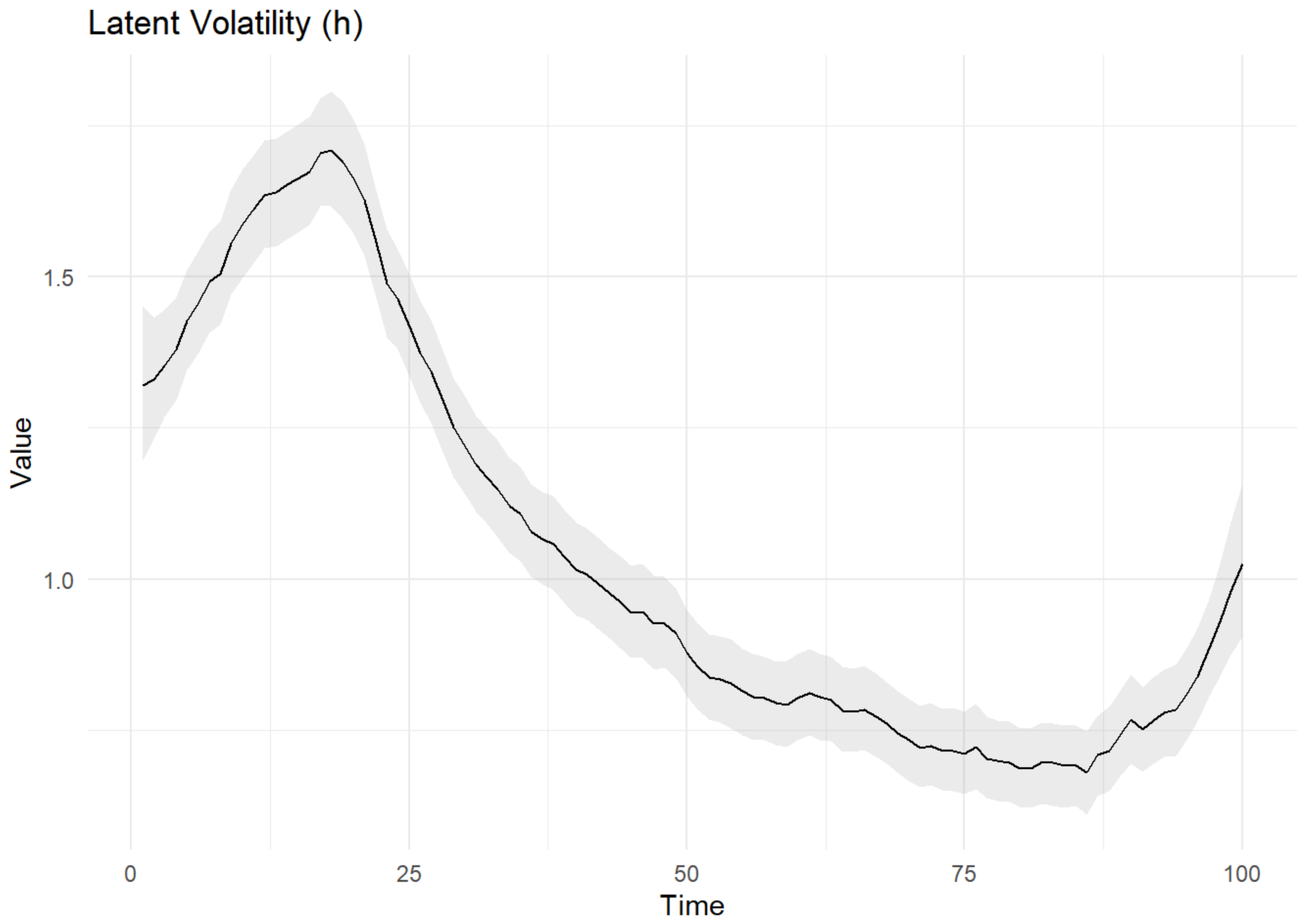
```
ggplot(filter(summary_stats, name == "beta_g_save"), aes(x = time, y = mean)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "lightcoral", alpha = 0.4) +
  geom_line(color = "red3") +
  labs(title = "Beta (time-varying)", x = "Time", y = "Value") +
  theme_minimal()
```



```r
ggplot(filter(summary_stats, name == "gamma_save"), aes(x = time, y = mean)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "plum", alpha = 0.4) +
  geom_line(color = "purple") +
  labs(title = "Gamma (time-varying)", x = "Time", y = "Value") +
  theme_minimal()
```



```
ggplot(filter(summary_stats, name == "h"), aes(x = time, y = mean)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "gray80", alpha = 0.4) +
  geom_line(color = "black") +
  labs(title = "Latent Volatility (h)", x = "Time", y = "Value") +
  theme_minimal()
```



# add true parameters

```r
true_df <- data.frame(
  time = 1:n,
  omega_true = omega_true,
  alpha_true = alpha_true,
  beta_true  = beta_true,
  gamma_true = gamma_true,
  h_true     = h_true
)
```

# Pivot longer to match summary format

```r
true_long <- pivot_longer(true_df, -time, names_to = "name", values_to = "true")
```

# Match naming with summary_stats

```
true_long$name <- recode(true_long$name,
                         omega_true = "omega_g_save",
                         alpha_true = "alpha_g_save",
                         beta_true  = "beta_g_save",
                         gamma_true = "gamma_save",
                         h_true     = "h"
)
```

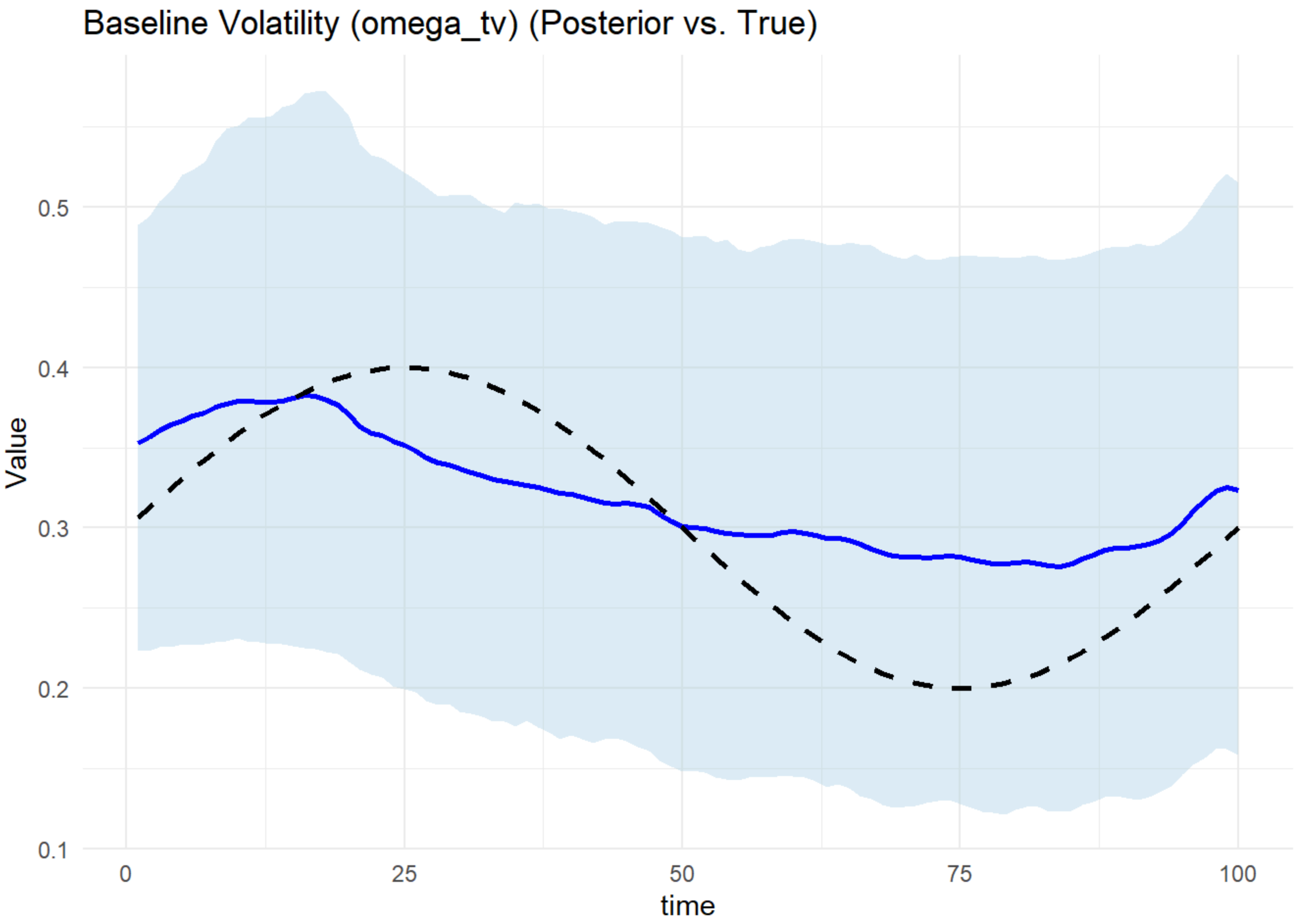
## Merge summary with true values

```
summary_stats_with_true <- left_join(summary_stats, true_long, by = c("name", "time"))
```

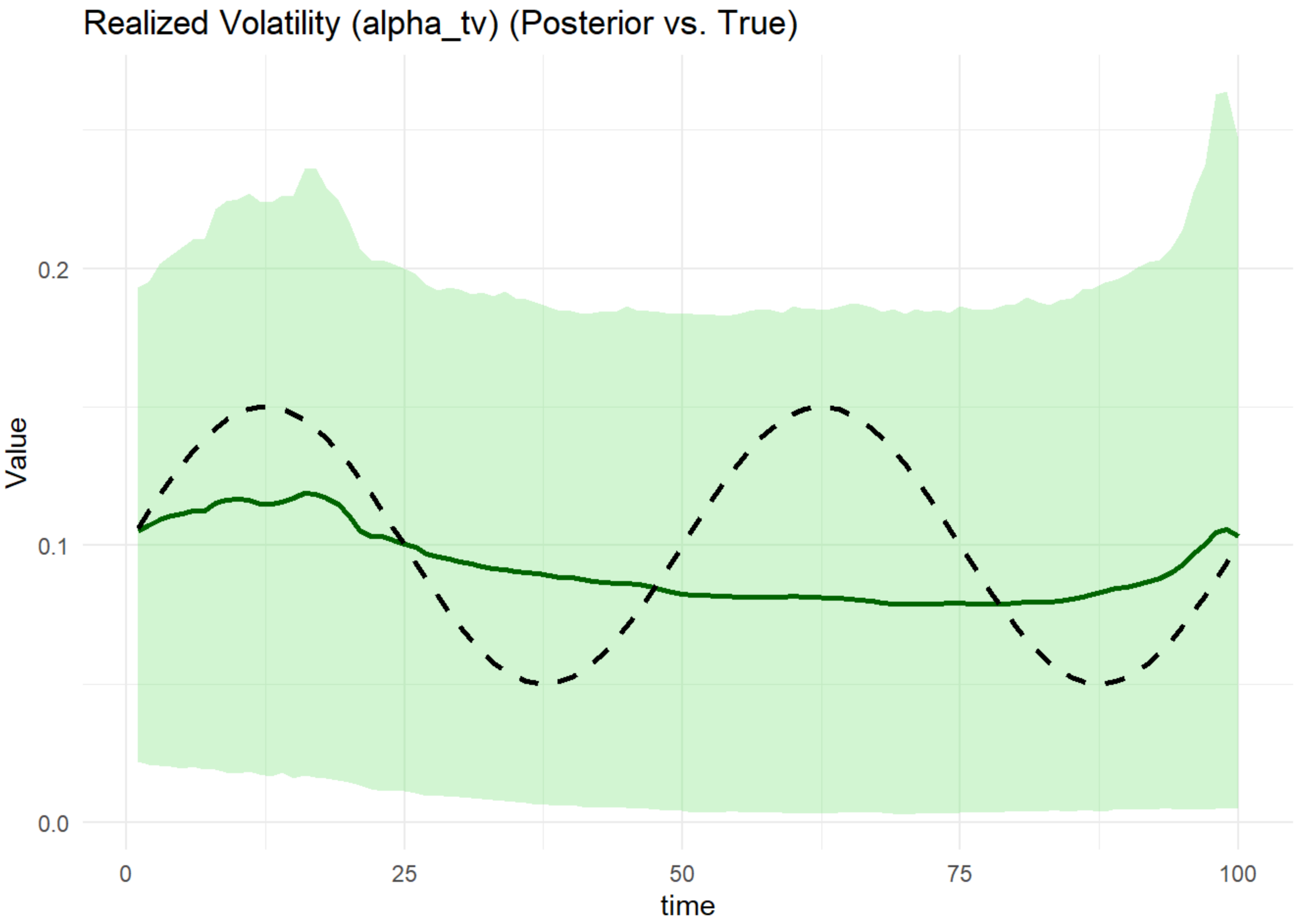
## True vs. BtvGARCH-Ito estimated values Plots

```
ggplot(filter(summary_stats_with_true, name == "omega_g_save"), aes(x = time)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "lightblue", alpha = 0.4) +
  geom_line(aes(y = mean), color = "blue", size = 1) +
  geom_line(aes(y = true), color = "black", linetype = "dashed", size = 0.9) +
  labs(title = "Baseline Volatility (omega_tv) (Posterior vs. True)", y = "Value") +
  theme_minimal()
```

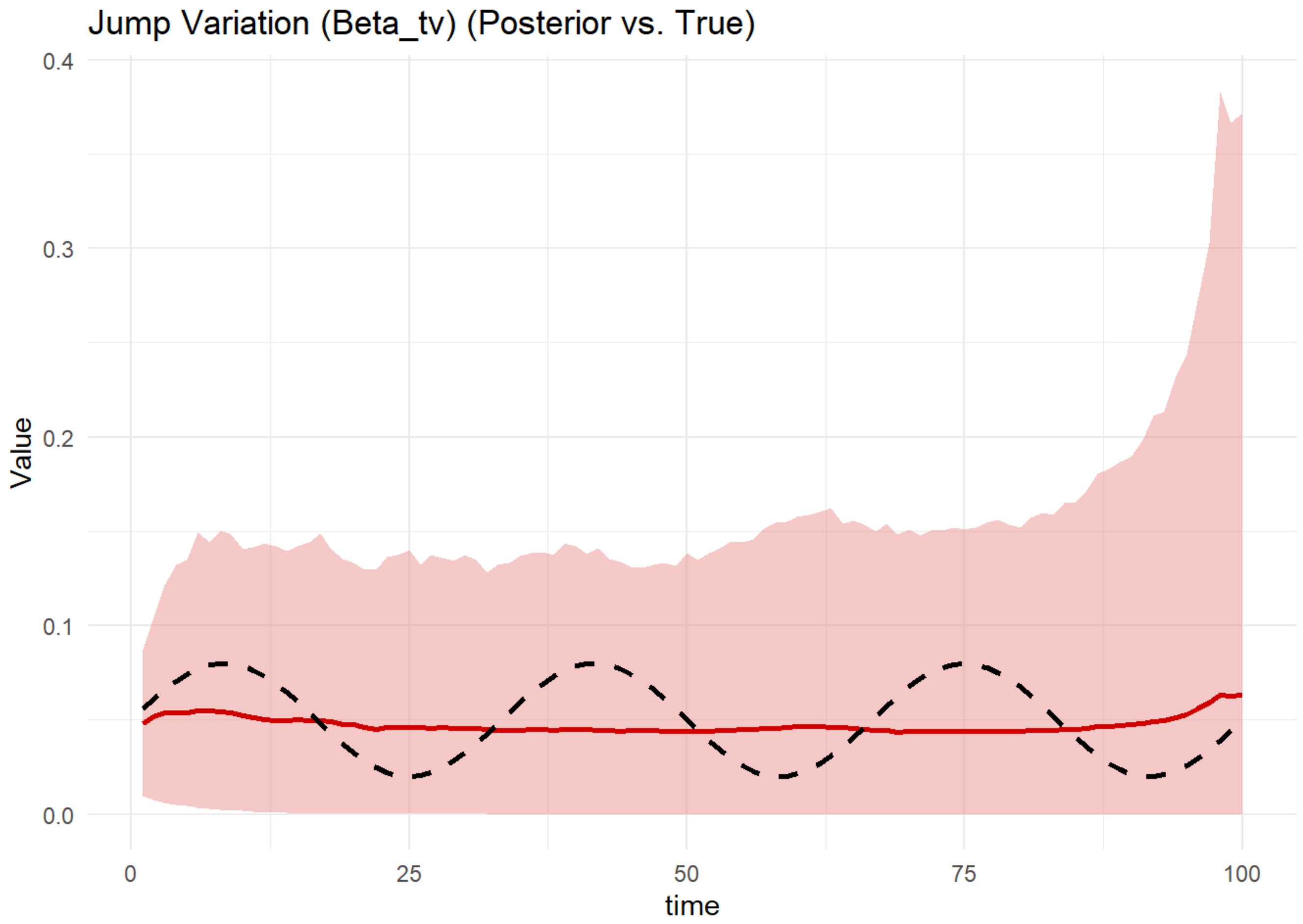
```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead.
```



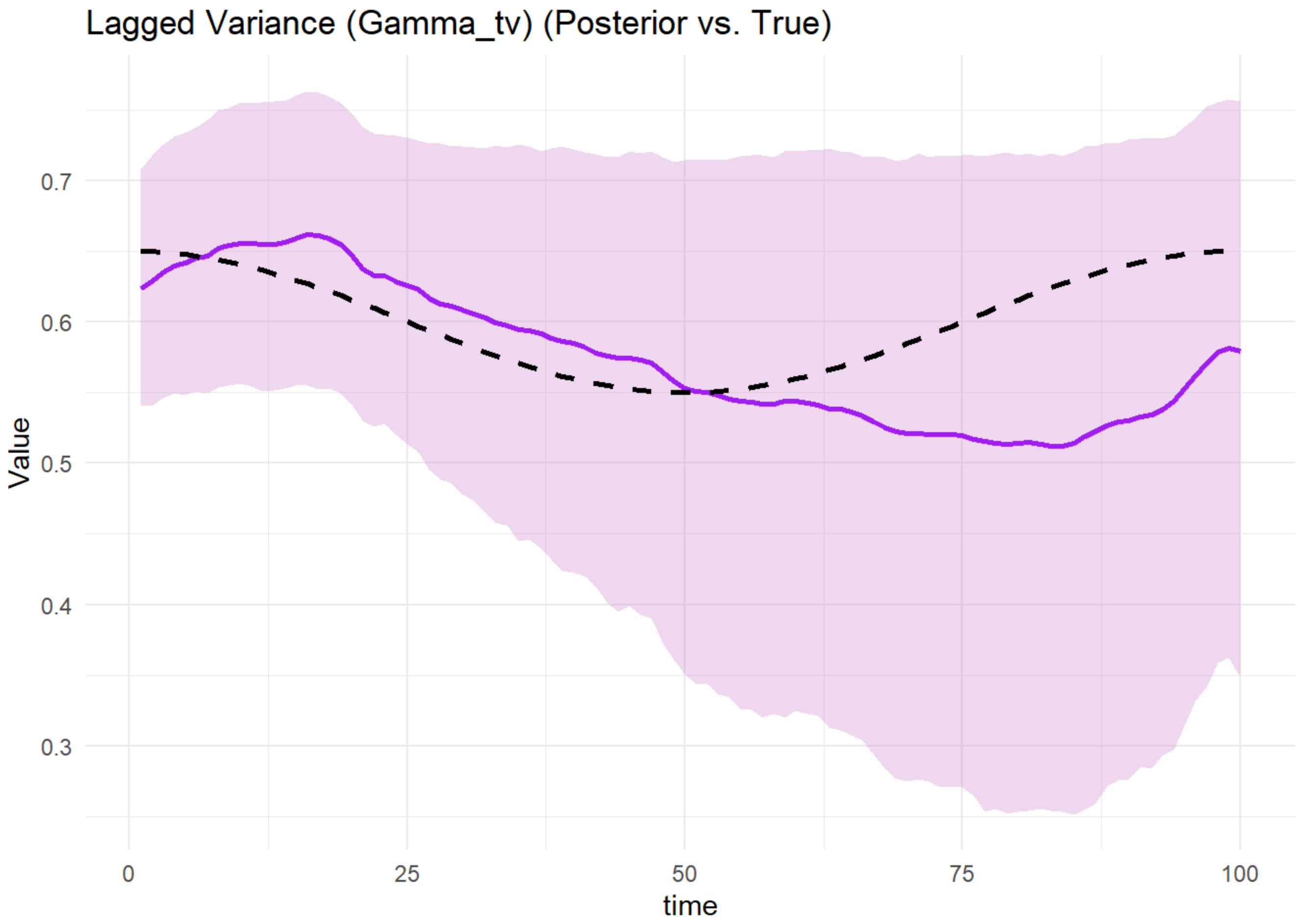
```
ggplot(filter(summary_stats_with_true, name == "alpha_g_save"), aes(x = time)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "lightgreen", alpha = 0.4) +
  geom_line(aes(y = mean), color = "darkgreen", size = 1) +
  geom_line(aes(y = true), color = "black", linetype = "dashed", size = 0.9) +
  labs(title = "Realized Volatility (alpha_tv) (Posterior vs. True)", y = "Value") +
  theme_minimal()
```



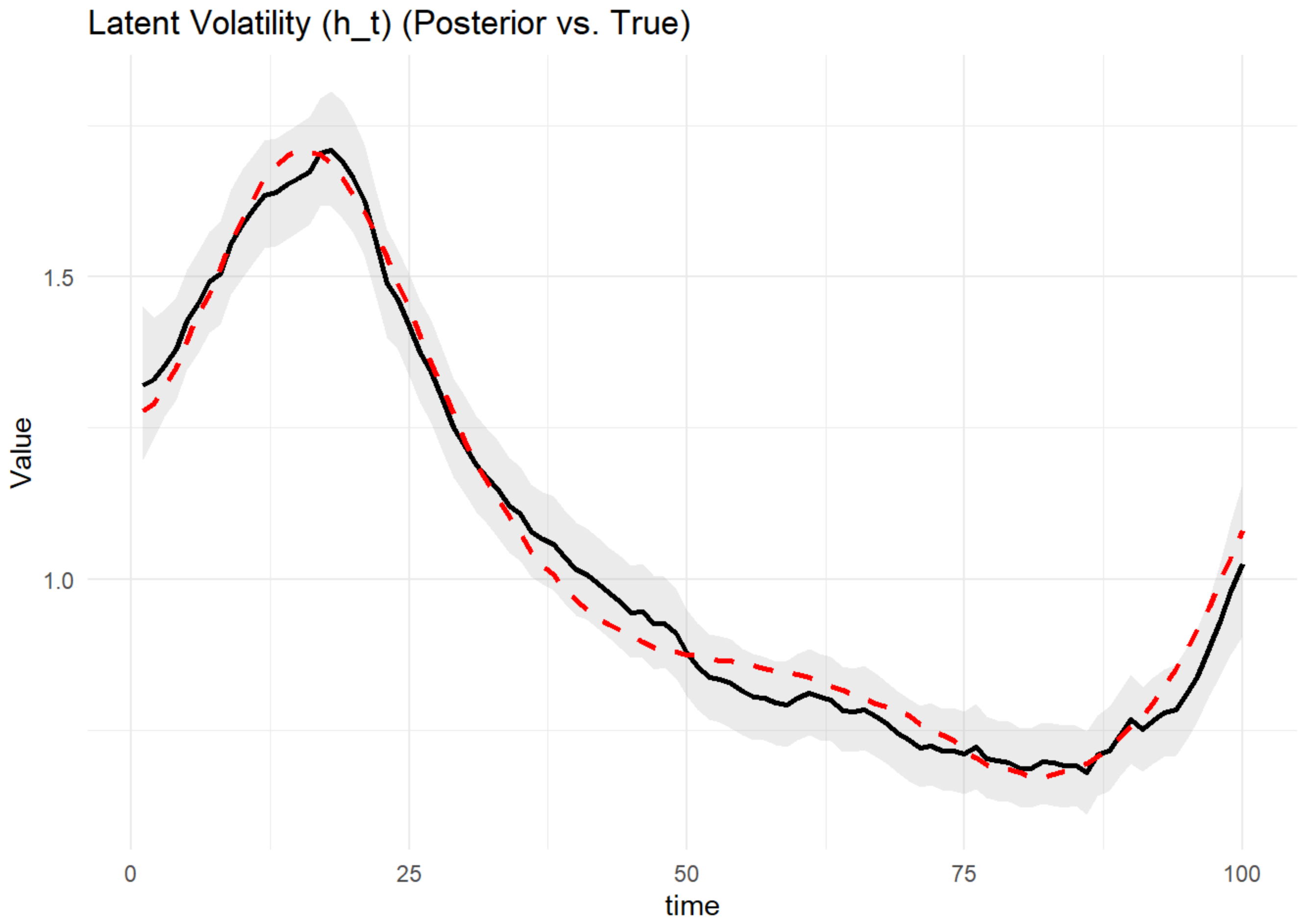
```
ggplot(filter(summary_stats_with_true, name == "beta_g_save"), aes(x = time)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "lightcoral", alpha = 0.4) +
  geom_line(aes(y = mean), color = "red3", size = 1) +
  geom_line(aes(y = true), color = "black", linetype = "dashed", size = 0.9) +
  labs(title = "Jump Variation (Beta_tv) (Posterior vs. True)", y = "Value") +
  theme_minimal()
```



```r
ggplot(filter(summary_stats_with_true, name == "gamma_save"), aes(x = time)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "plum", alpha = 0.4) +
  geom_line(aes(y = mean), color = "purple", size = 1) +
  geom_line(aes(y = true), color = "black", linetype = "dashed", size = 0.9) +
  labs(title = "Lagged Variance (Gamma_tv) (Posterior vs. True)", y = "Value") +
  theme_minimal()
```



```r
ggplot(filter(summary_stats_with_true, name == "h"), aes(x = time)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), fill = "gray80", alpha = 0.4) +
  geom_line(aes(y = mean), color = "black", size = 1) +
  geom_line(aes(y = true), color = "red", linetype = "dashed", size = 0.9) +
  labs(title = "Latent Volatility (h_t) (Posterior vs. True)", y = "Value") +
  theme_minimal()
```



## Compute error metrics

```
error_metrics <- summary_stats_with_true %>%
  group_by(name) %>%
  summarise(
    MSE = mean((mean - true)^2),
    RMSE = sqrt(MSE),
    MAE = mean(abs(mean - true)),
    .groups = "drop"
  )
print(error_metrics)
```

```
# A tibble: 8 × 4
  name                 MSE     RMSE      MAE
  <chr>              <dbl>    <dbl>    <dbl>
1 .chain        NA         NA       NA
2 .draw         NA         NA       NA
3 .iteration    NA         NA       NA
4 alpha_g_save   0.00115    0.0340   0.0290
5 beta_g_save    0.000480   0.0219   0.0198
6 gamma_save     0.00324    0.0569   0.0440
```

```
7 h             0.00123   0.0350  0.0301
8 omega_g_save  0.00240   0.0490  0.0432
```

## Trace plot of Omega

```
df_draws <- fit$draws(format = "df")
param_vars_df <- grep("^omega_raw", colnames(df_draws), value = TRUE)
df_sub <- df_draws[, param_vars_df[1:5]] # Subset data.frame
```

```
Warning: Dropping 'draws_df' class as required metadata was removed.
```

```
mat_draws <- as.matrix(df_sub) # Convert to matrix for bayesplot
bayesplot::mcmc_trace(mat_draws) + ggtitle("Traceplots for omega_raw (first 5 time points)")
```