

Modello gerarchico

Laura Balasso

10/24/2020

Hierarchical model for 3 italian regions

```
regions <- c('Lazio', 'Lombardia', 'Abruzzo')
hier_data <- get_hier_data(data_it, regions, initial_date = as.Date('2020-07-20'))

p_delay <- get_delay_distribution()

stan_data_hier <- list(J = length(regions),
                      N = nrow(hier_data$exposures),
                      N_nonzero = length(hier_data$nonzero_days),
                      nonzero_days = hier_data$nonzero_days,
                      conv_gt = get_gt_convolution(nrow(hier_data$exposures)),
                      length_delay = length(p_delay),
                      p_delay = p_delay,
                      exposures = hier_data$exposures,
                      nonzero_positives = hier_data$positives[hier_data$nonzero_days ,]
)

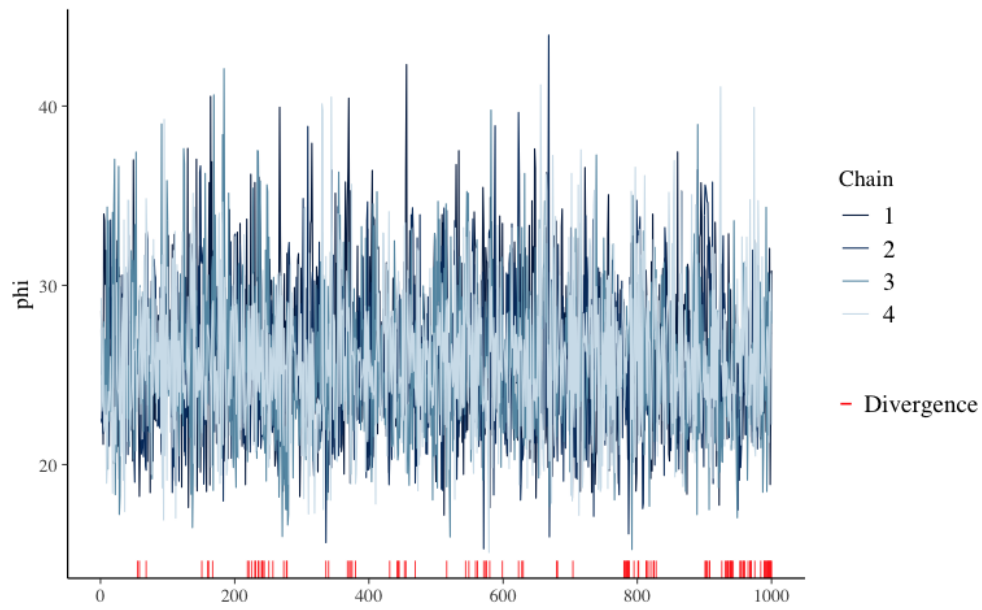
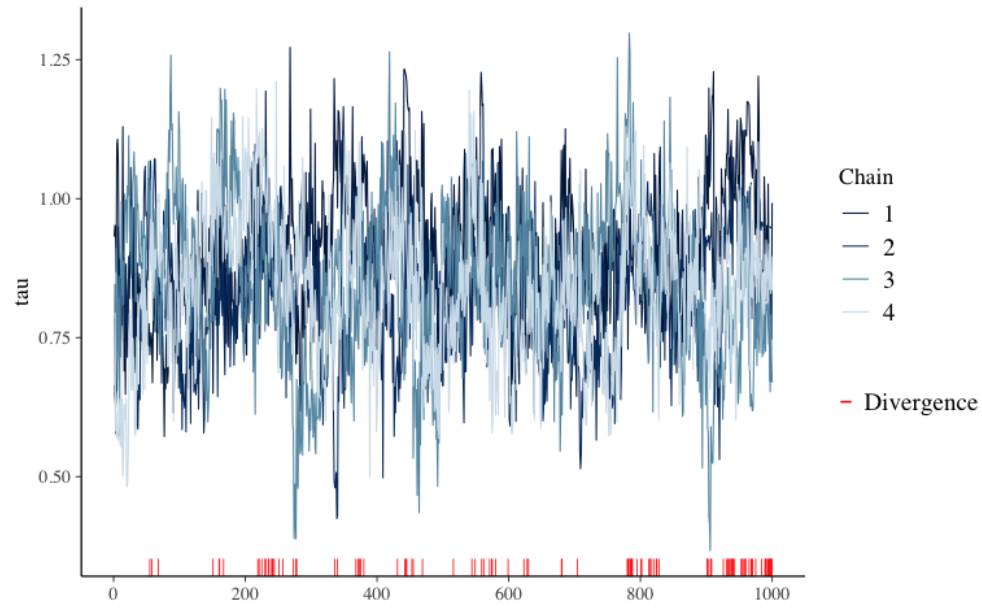
compiled_hier <- stan_model('hier_rt_model.stan')

fit_hier <- sampling(compiled_hier, data = stan_data_hier, iter= 2000)
```

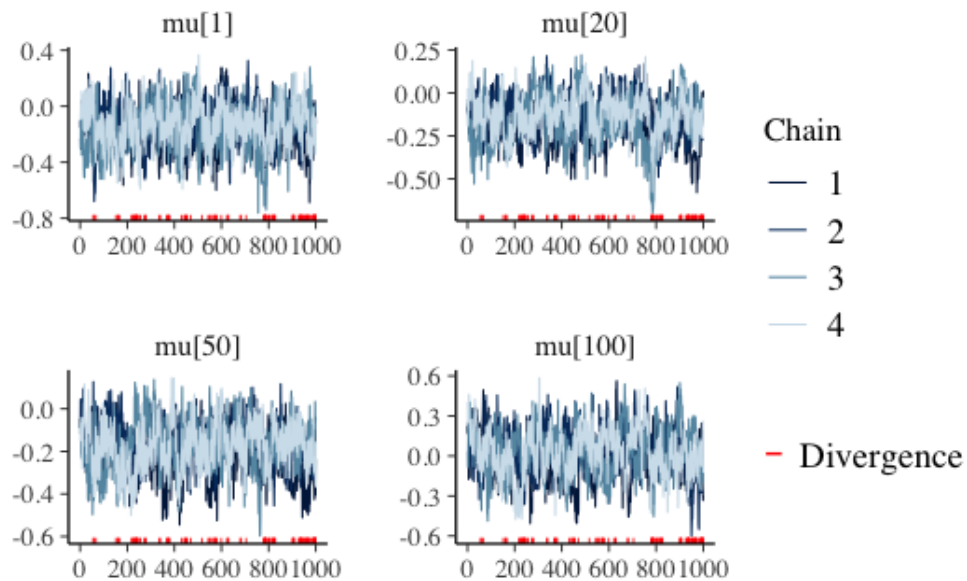
Trace plots

```
mcmc_trace(as.array(fit_hier, pars = c('tau')),
           np = nuts_params(fit_hier)
)

mcmc_trace(as.array(fit_hier, pars = c('phi')),
           np = nuts_params(fit_hier)
)
```

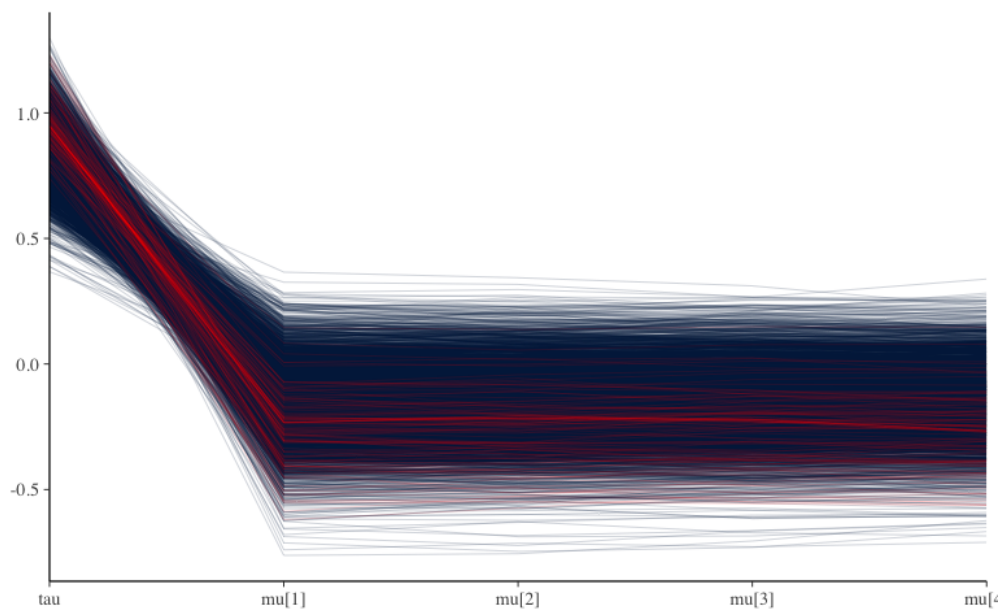


```
mcmc_trace(as.array(fit_hier, pars = c('mu[1]', 'mu[20]', 'mu[50]', 'mu[100]')),
            np = nuts_params(fit_hier)
)
```



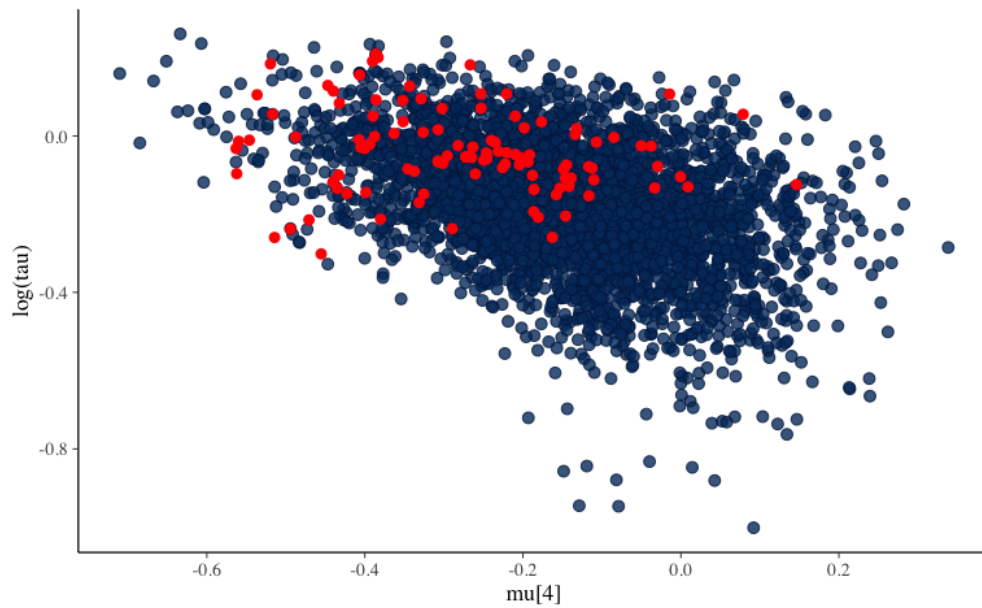
Divergent transition

```
parcoord_with_divs <- mcmc_parcoord(
  as.array(fit_hier, pars = c("tau", "mu[1]", "mu[2]", "mu[3]", "mu[4]")),
  np = nuts_params(fit_hier)
)
parcoord_with_divs
```

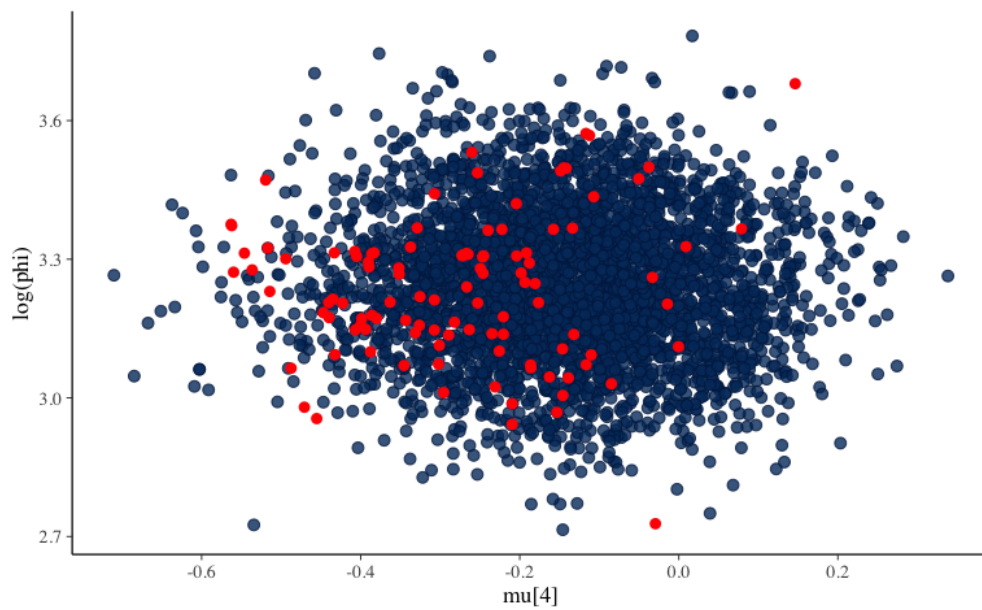


```
scatter_with_divs <- mcmc_scatter(
  as.array(fit_hier),
  pars = c("mu[4]", "tau"),
  transform = list('tau' = "log"),
  np = nuts_params(fit_hier)
)
```

```
)
scatter_with_divs
```

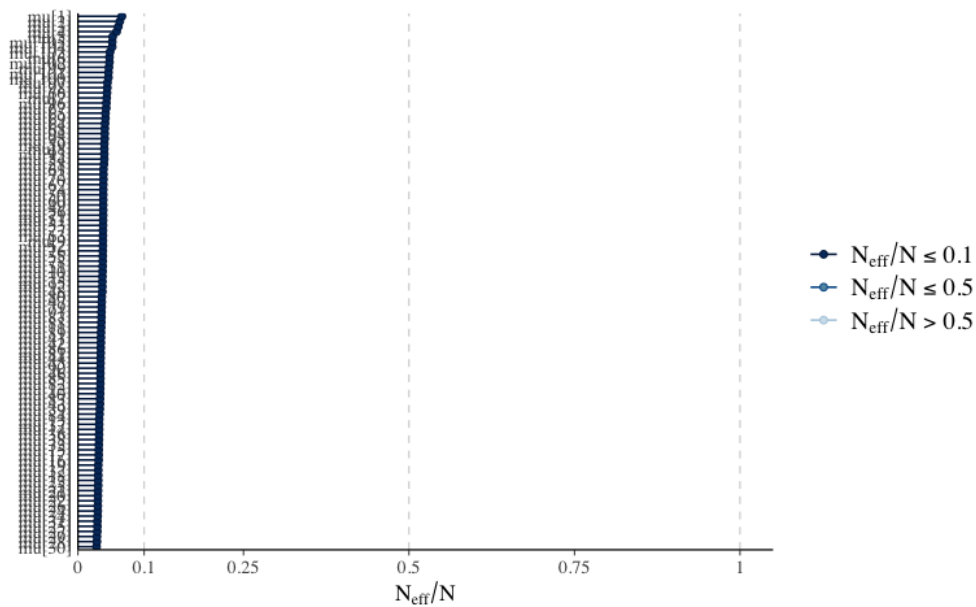


```
scatter_with_divs2 <- mcmc_scatter(
  as.array(fit_hier),
  pars = c("mu[4]", 'phi'),
  transform = list('phi' = "log"),
  np = nuts_params(fit_hier)
)
scatter_with_divs2
```



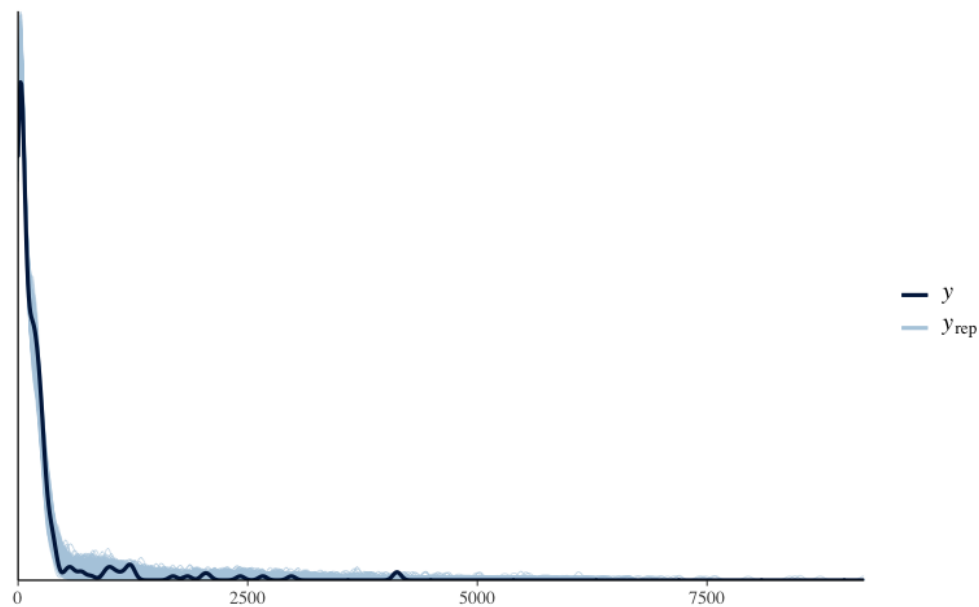
Effective sample size

```
ratios1 <- neff_ratio(fit_hier, pars = c('mu'))  
mcmc_neff(ratios1) + yaxis_text(hjust = 1)
```

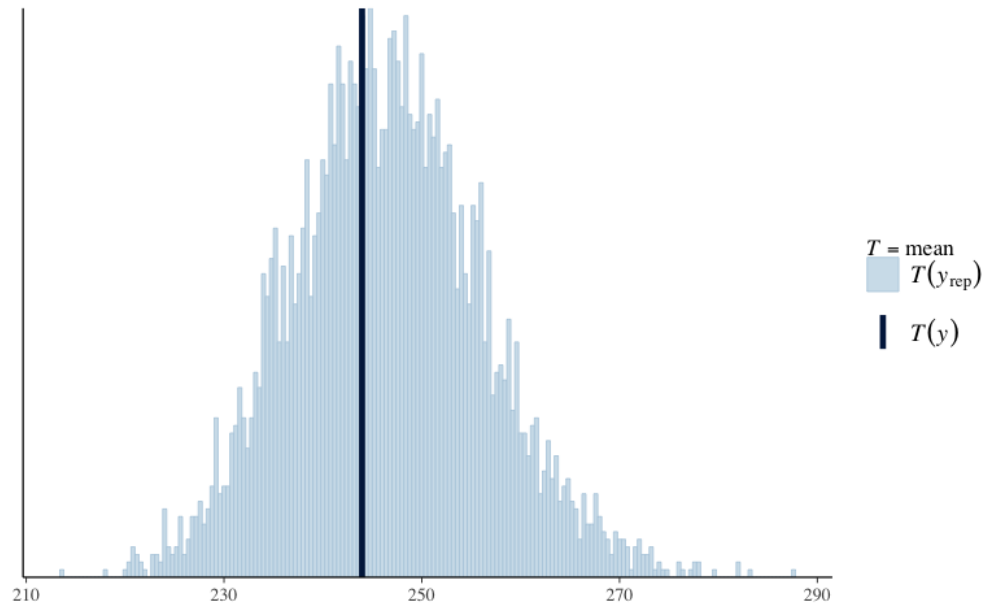


Posterior predictive check

```
y_rep <- as.matrix(fit_hier, pars = "y_rep")  
ppc_dens_overlay(y = as.vector(stan_data_hier$nonzero_positives), y_rep[1:1000, ])
```



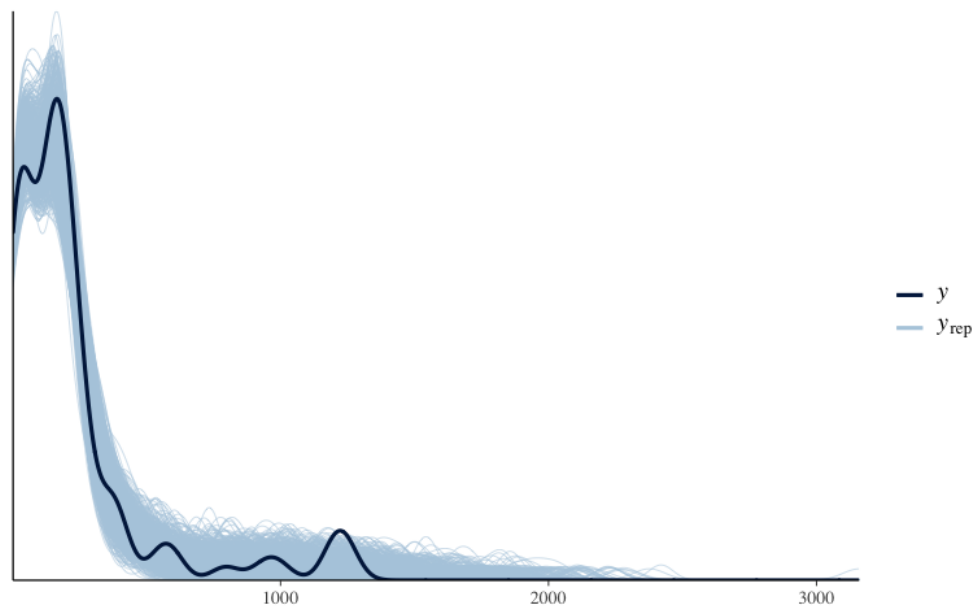
```
ppc_stat(y=as.vector(stan_data_hier$nonzero_positives), yrep = y_rep, stat="mean", binwidth = 0.4)
```



Posterior predictive check by region

```
regional_yrep_idx <- function(region, regions_vector, nonzero_days){
  region_idx <- which(regions_vector == region)
  yrep_idx <- (region_idx-1)*length(nonzero_days) + 1
  range <- yrep_idx : (yrep_idx + length(nonzero_days)-1)
  return(range)
}
```

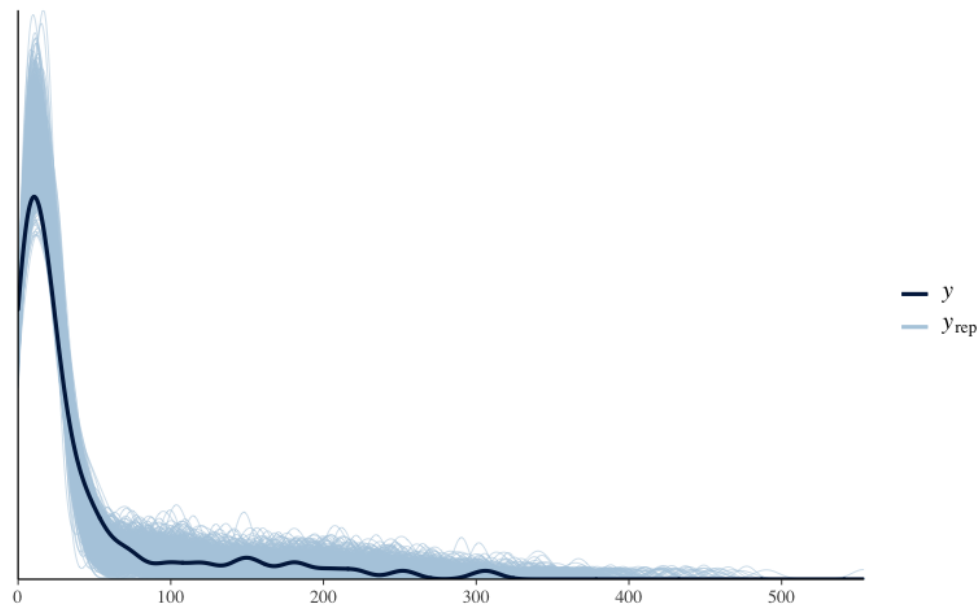
```
ppc_dens_overlay(y = stan_data_hier$nonzero_positives[, which(regions == 'Lazio')], y_rep[1:1000, regional_yrep_idx])
```



```
ppc_dens_overlay(y = stan_data_hier$nonzero_positives[, which(regions == 'Lombardia')], y_rep[1:1000,regions == 'Lombardia'])
```

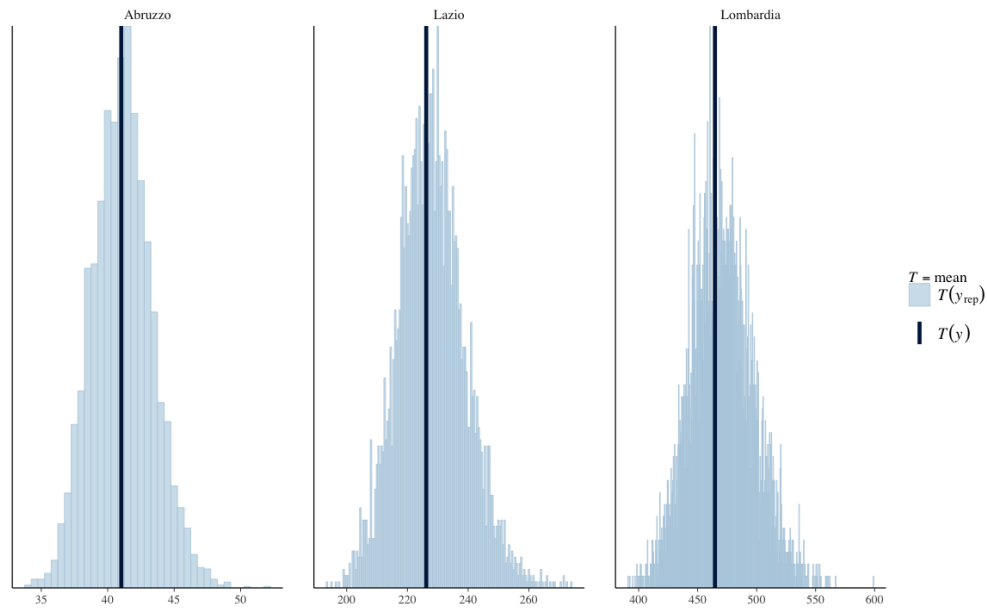


```
ppc_dens_overlay(y = stan_data_hier$nonzero_positives[, which(regions == 'Abruzzo')], y_rep[1:1000,regions == 'Abruzzo'])
```

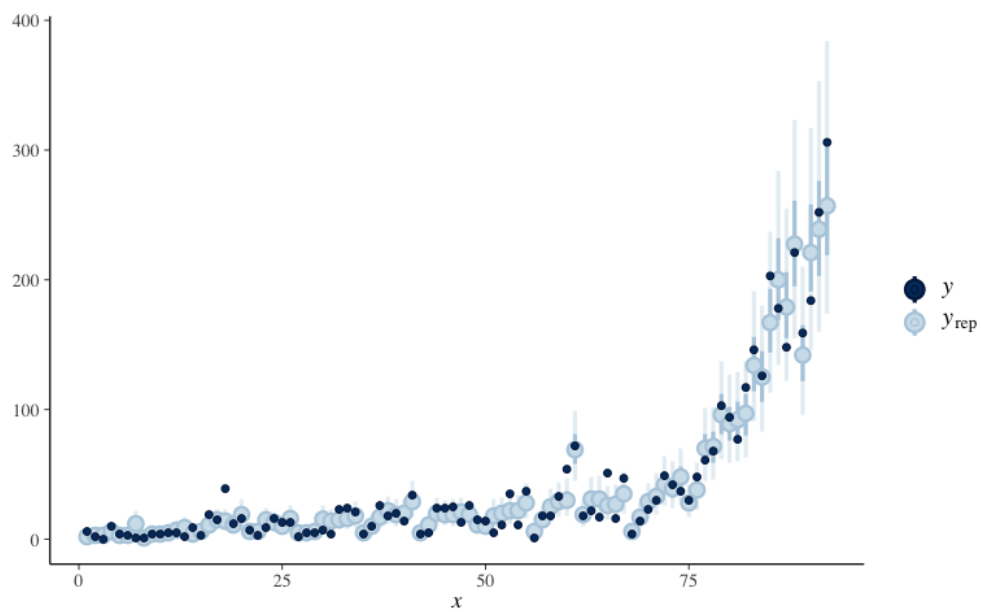


```
groups <- function(regions, nonzero_days){
  group <- rep(regions[1], length(nonzero_days))
  for(r in 2:length(regions))
    group <- c(group, rep(regions[r], length(nonzero_days)))

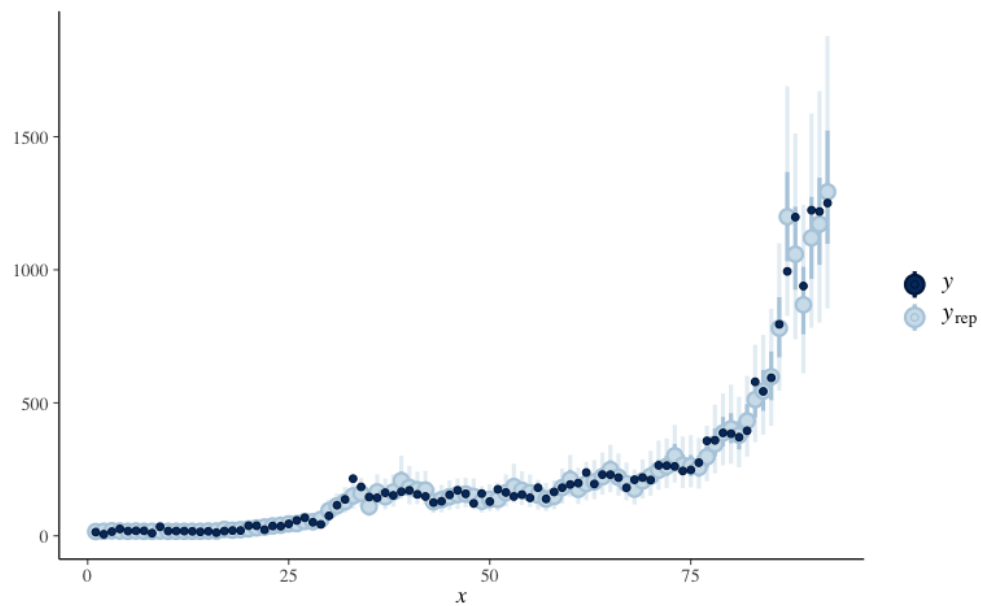
  return(group)
}
ppc_stat_grouped(y=as.vector(stan_data_hier$nonzero_positives), yrep=y_rep, group = groups(regions, stan_data_hier$nonzero_days))
```



```
ppc_intervals(y = stan_data_hier$nonzero_positives[, which(regions == 'Abruzzo')], y_rep[1:1000, regional_y])
```



```
ppc_intervals(y = stan_data_hier$nonzero_positives[, which(regions == 'Lazio')], y_rep[1:1000, regional_y])
```

```
ppc_intervals(y = stan_data_hier$nonzero_positives[, which(regions == 'Lombardia')], y_rep[1:1000,regions == 'Lombardia'])
```

