

## Hierarchical Bayesian Modeling with tidymodels

```
# Load packages
library(tidyverse)
library(tidymodels)
library(multilevelmod) # install from Github

# Grab data and subset to the North Rocky Forest
df <- read_csv("../data/subsets/dat_small.csv")
m333 <- df %>%
  filter(province == "M333")
```

### Fitting a model

```
set.seed(37) # the best number

# HB engine, standard priors
hb_spec <- linear_reg() %>%
  set_engine("stan-glmer",
    prior_aux = rstanarm::exponential(rate = 1),
    prior = NULL,
    prior_intercept = NULL,
    prior_covariance = rstanarm::decov(shape = 2))

# Fit the model
hb_fit <-
  hb_spec %>%
  fit(BIOLIVE_TPA ~ 1 + forprob + (1 | subsection), # varying intercepts
    data = m333)

hb_fit
```

```
## parsnip model object
##
## Fit time: 56.4s
## stan_glmer
## family:      gaussian [identity]
## formula:      BIOLIVE_TPA ~ 1 + forprob + (1 | subsection)
## observations: 3003
## -----
##              Median MAD_SD
## (Intercept) -4.7      3.8
## forprob      48.8      2.9
##
## Auxiliary parameter(s):
##              Median MAD_SD
## sigma 35.2      0.5
##
## Error terms:
## Groups      Name      Std.Dev.
```

```
## subsection (Intercept) 12
## Residual 35
## Num. levels: subsection 20
##
## -----
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

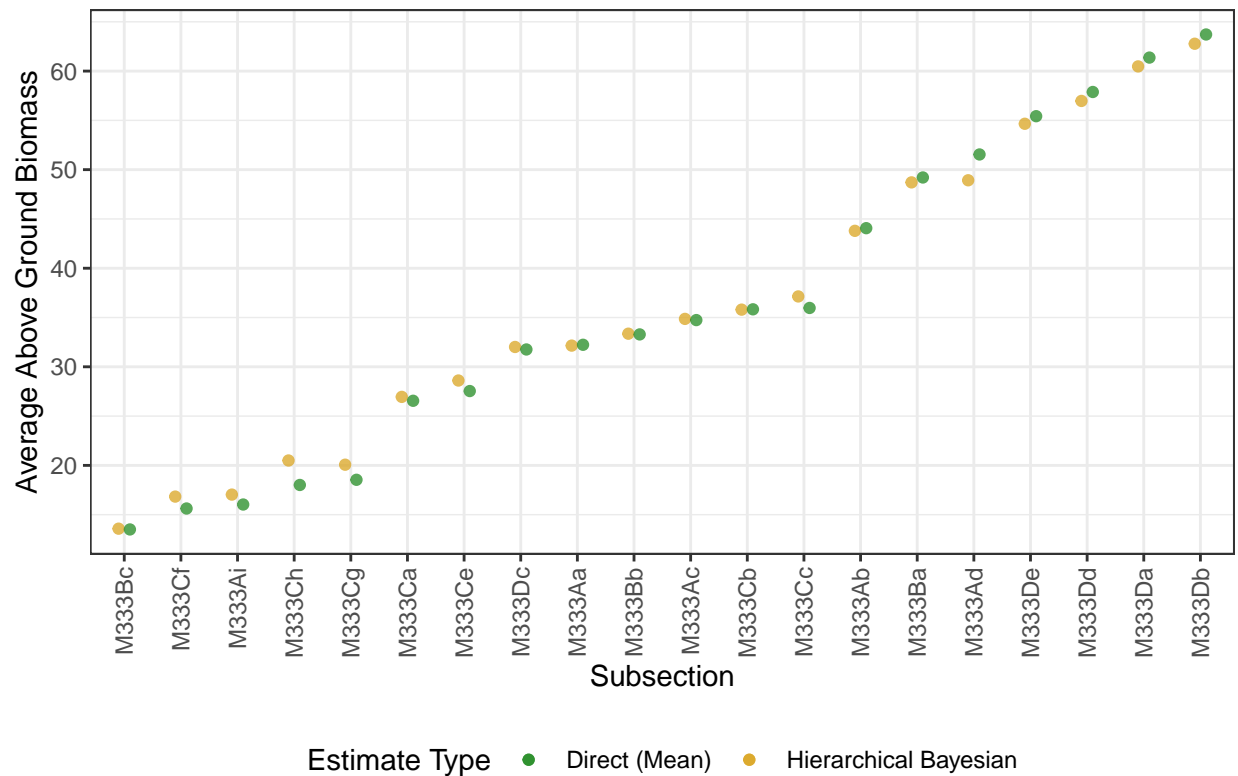
*# Let's plot the means*

```
hb_fit_df <- data.frame(
  fitted = hb_fit$fit$fitted.values,
  true = m333$BIOLIVE_TPA,
  subsection = m333$subsection
)
```

*# In this plot, I would like error bars on the HB estimate. however, since the model fits to the plot level and then I summarized these means, I am not sure what a true error bar would look like.*

```
hb_fit_df %>%
  group_by(subsection) %>%
  summarize(mean_fit = mean(fitted),
             mean_true = mean(true)) %>%
  mutate(subsection = fct_reorder(subsection, mean_true)) %>%
  ggplot(aes(x = subsection,
             y = mean_fit)) +
  geom_point(aes(color = "goldenrod"),
             alpha = 0.75,
             position = position_nudge(x = -0.1)) +
  geom_point(aes(y = mean_true, color = "forestgreen"),
             alpha = 0.75,
             position = position_nudge(x = 0.1)) +
  theme_bw() +
  theme(axis.text.x = element_text(
    angle = 90,
    vjust = 0.5,
    hjust = 1
  ),
        legend.position = "bottom") +
  labs(
    x = "Subsection",
    y = "Average Above Ground Biomass",
    title = "Above Ground Biomass Estimates in the North Rocky Forest"
  ) +
  scale_color_manual(name = 'Estimate Type',
                     values = c('goldenrod'='goldenrod',
                                'forestgreen'='forestgreen'),
                     labels = c('Direct (Mean)', 'Hierarchical Bayesian'),
                     guide = "legend")
```

## Above Ground Biomass Estimates in the North Rocky Forest



Fit freq model, split data, compare test MSEs

```
set.seed(1)
m333_test <- m333 %>%
  sample_frac(0.25)
m333_train <- m333 %>%
  anti_join(m333_test)

freq_spec <- linear_reg() %>%
  set_engine("lmer")

freq_fit <- freq_spec %>%
  fit(BIOLIVE_TPA ~ 1 + forprob + (1 | subsection), # varying intercepts
    data = m333_train)

hb_fit_train <-
  hb_spec %>%
  fit(BIOLIVE_TPA ~ 1 + forprob + (1 | subsection), # varying intercepts
    data = m333_train)

results_test <- hb_fit_train %>%
  predict(new_data = m333_test) %>%
  mutate(
```

```

    truth = m333_test$BIOLIVE_TPA,
    model = "hb"
  ) %>%
  bind_rows(
    freq_fit %>%
    predict(new_data = m333_test) %>%
    mutate(
      truth = m333_test$BIOLIVE_TPA,
      model = "freq"
    )
  )

results_test %>%
  group_by(model) %>%
  rmse(truth = truth,
       estimate = .pred)

```

```

## # A tibble: 2 x 4
##   model .metric .estimator .estimate
##   <chr> <chr>   <chr>         <dbl>
## 1 freq  rmse     standard      34.6
## 2 hb    rmse     standard      34.6

```