

week-9-analysis

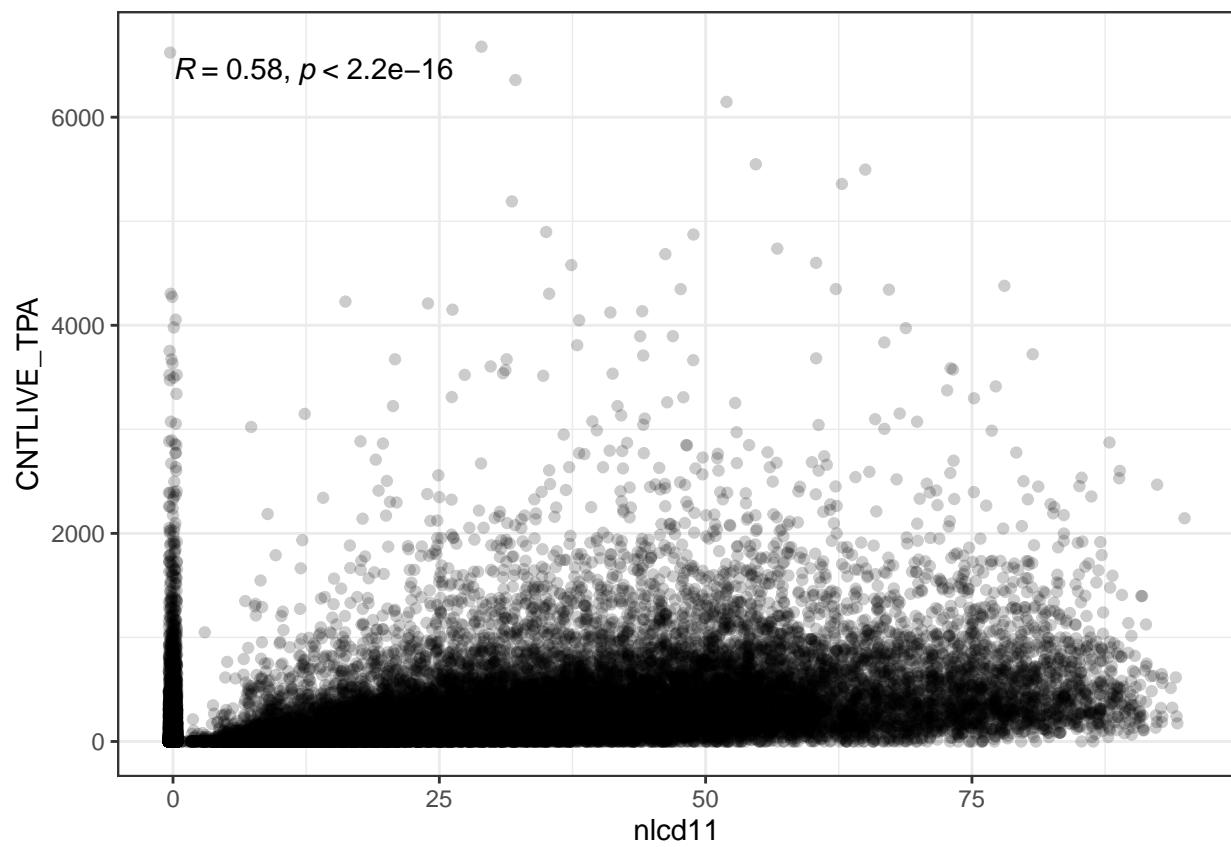
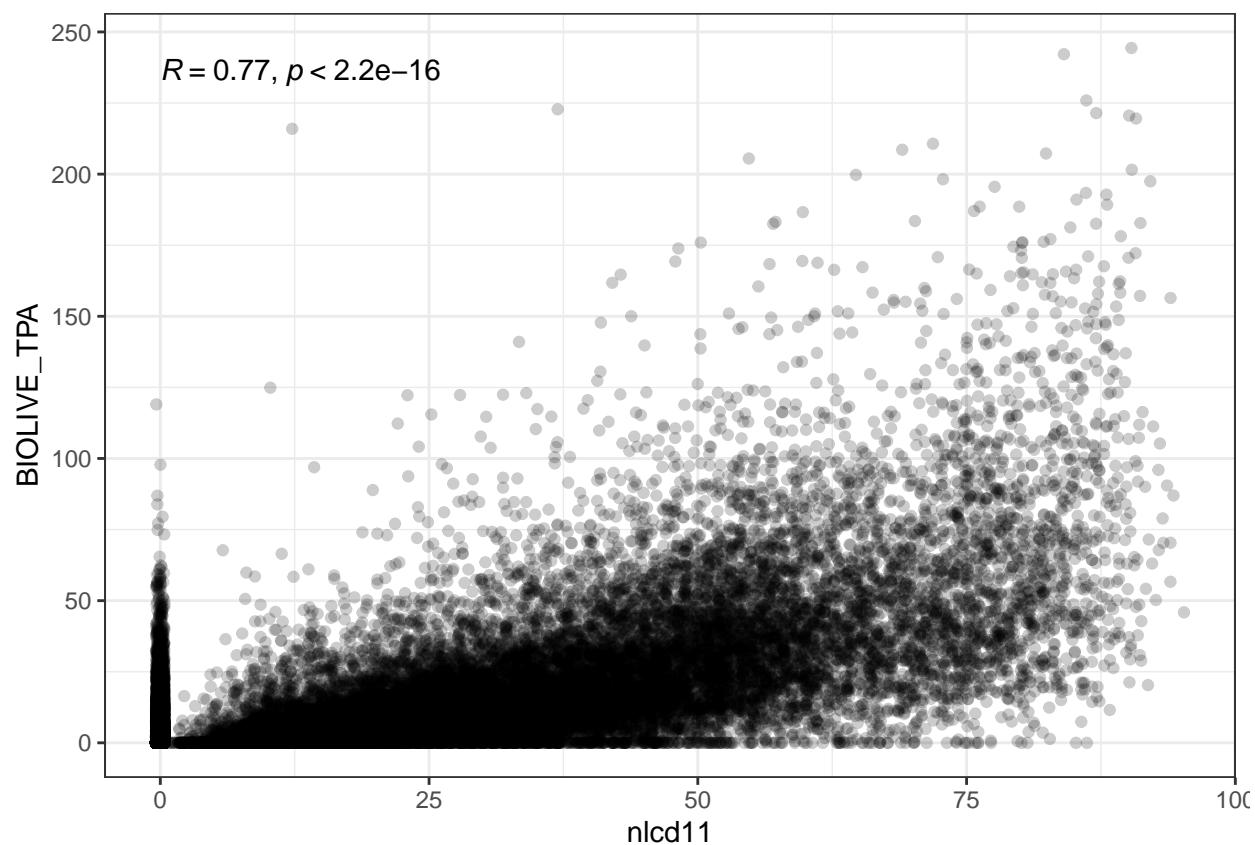
Grayson White

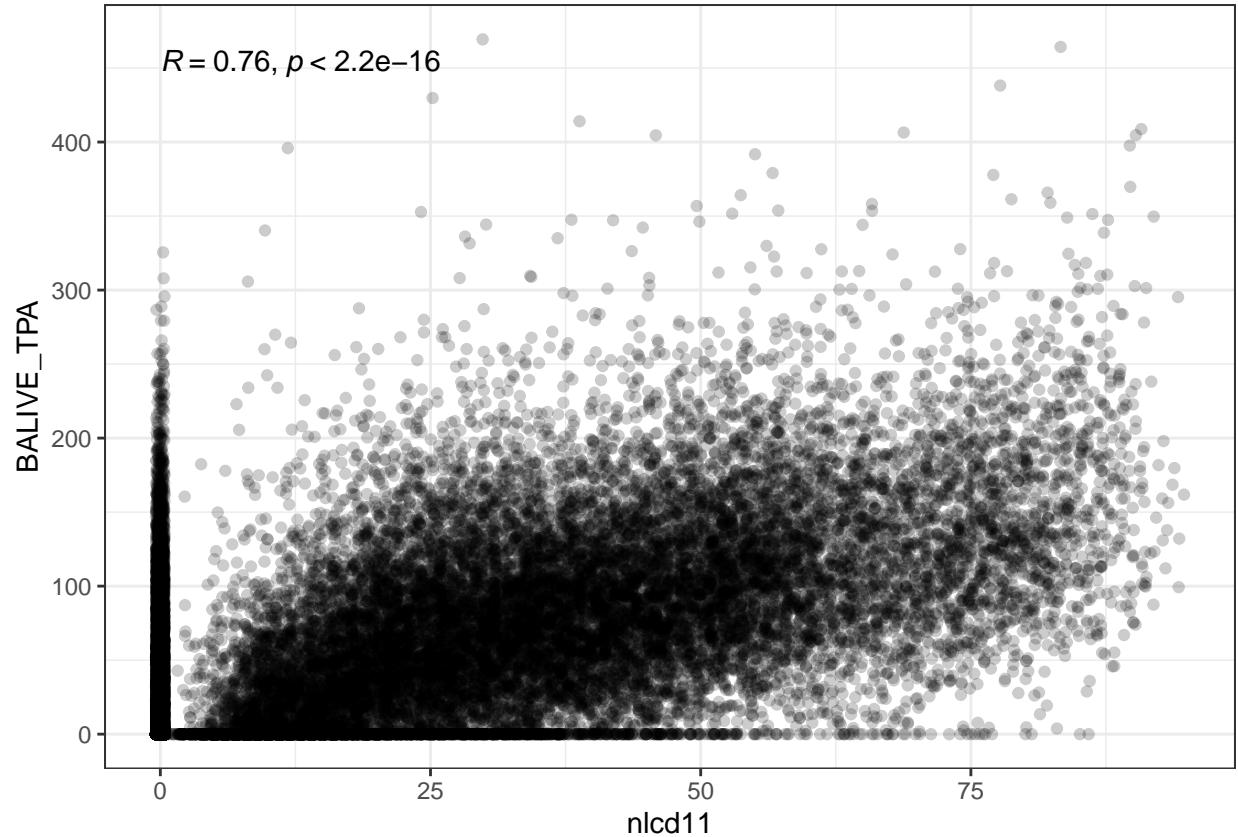
10/26/2020

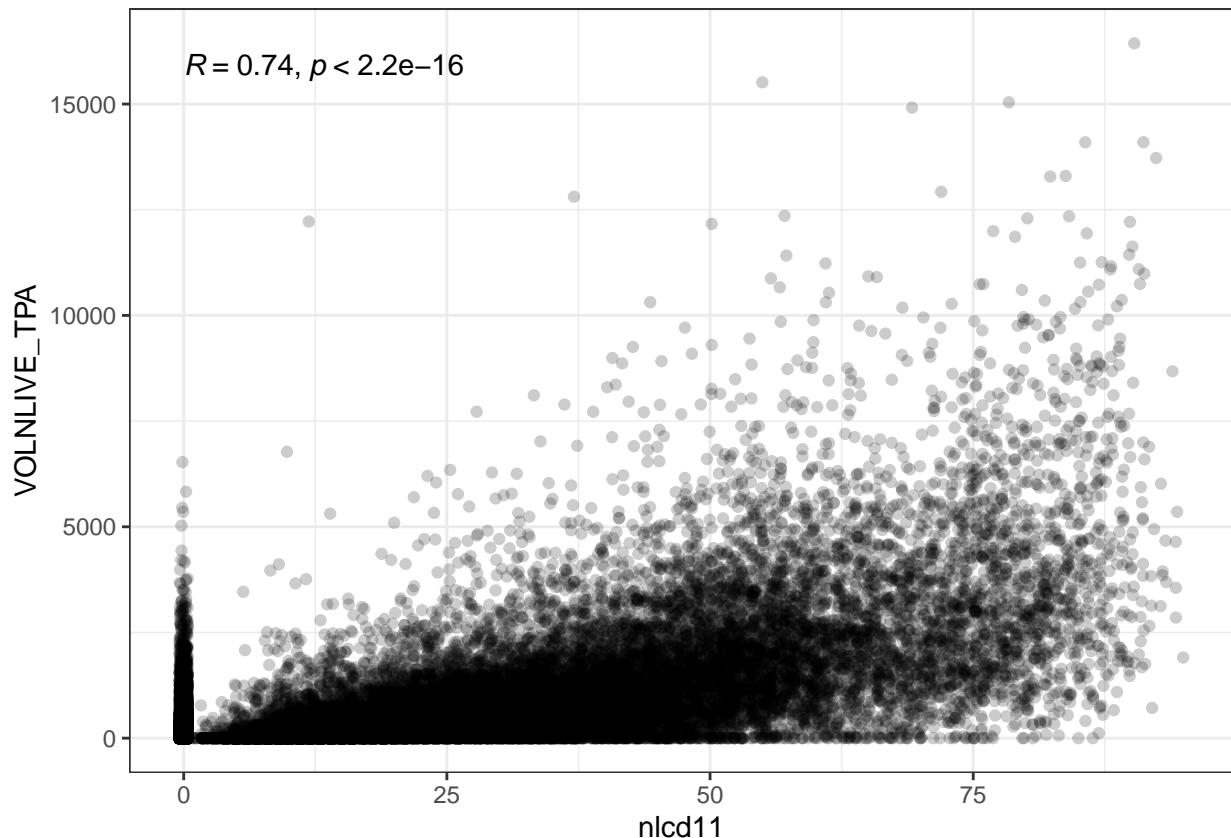
Goal 1: Scatterplots: nlcld11 by Y variables at the unit and area level

```
# Load data
dat_small <- read_csv("../data/subsets/dat_small.csv")
```

Unit level correlations with Y variables



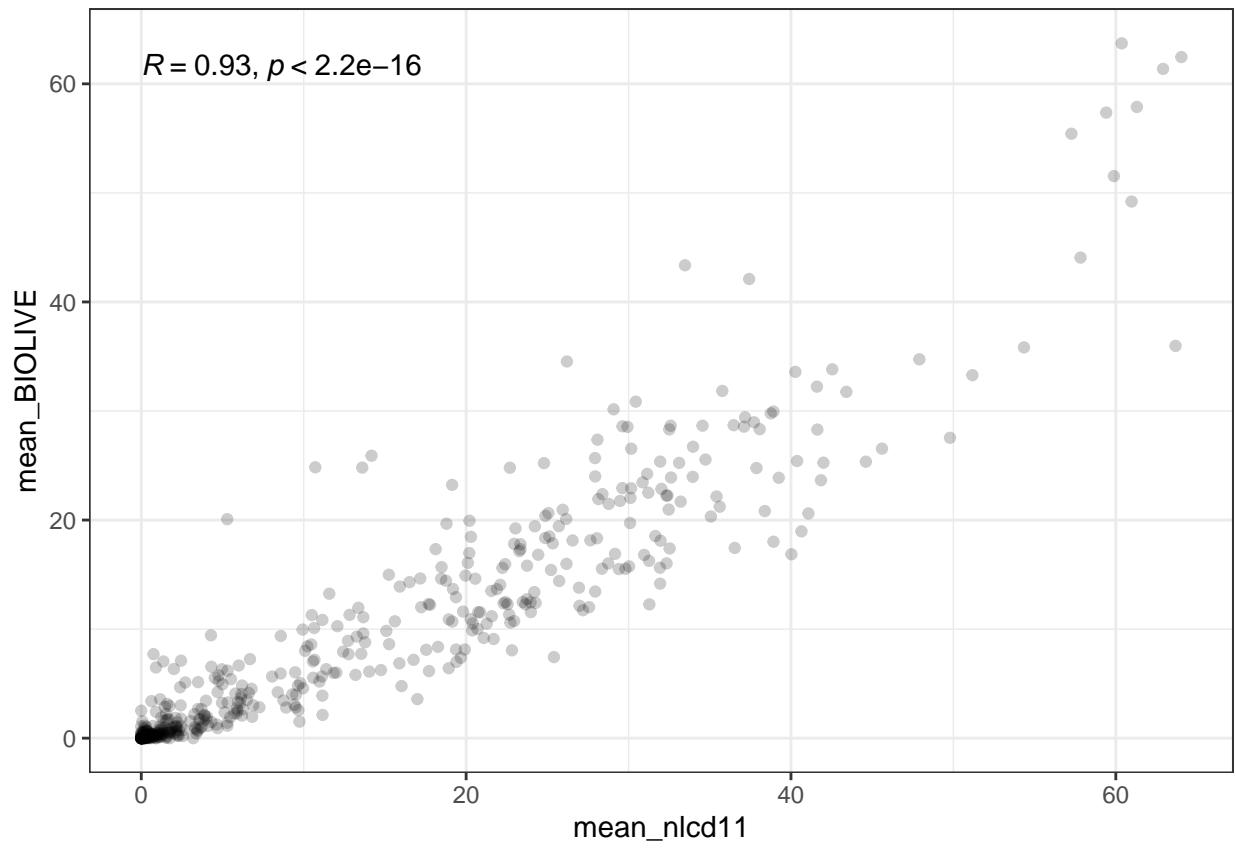


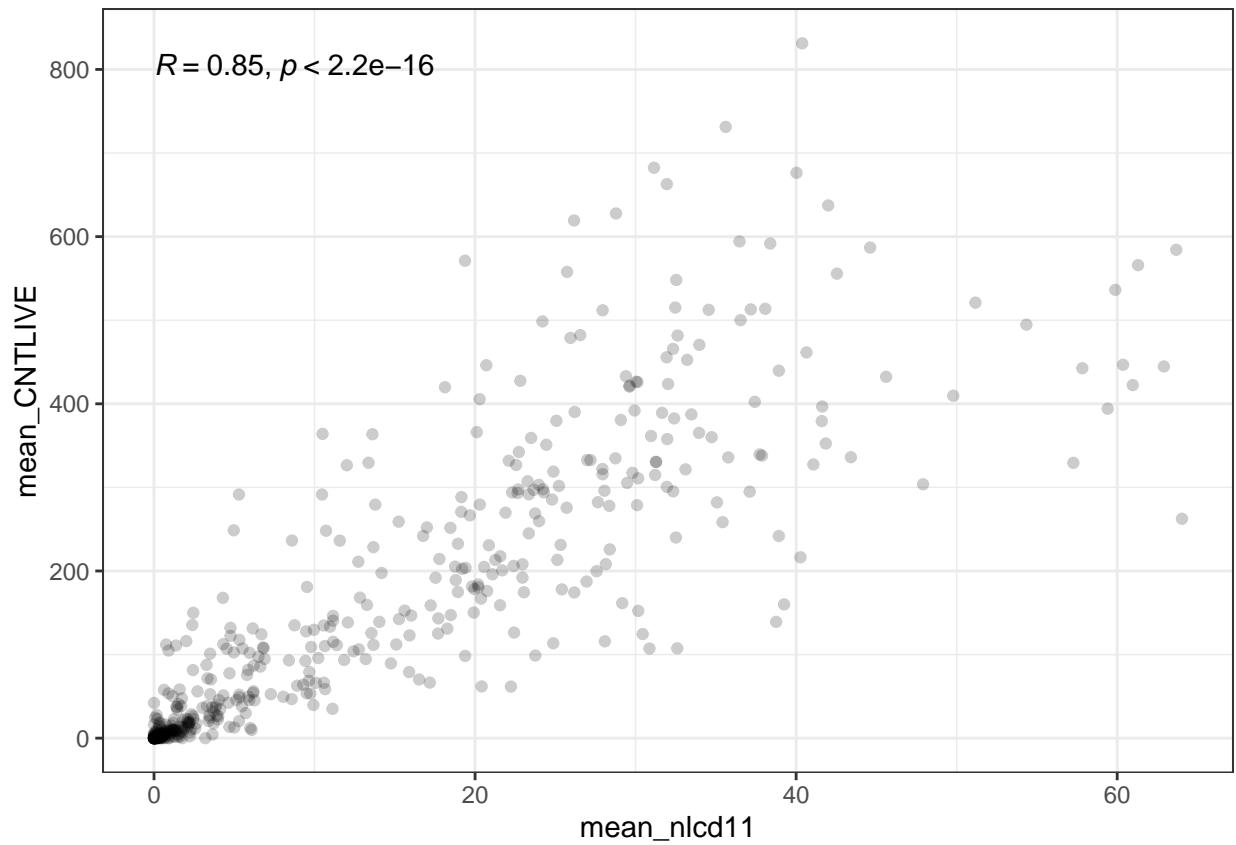


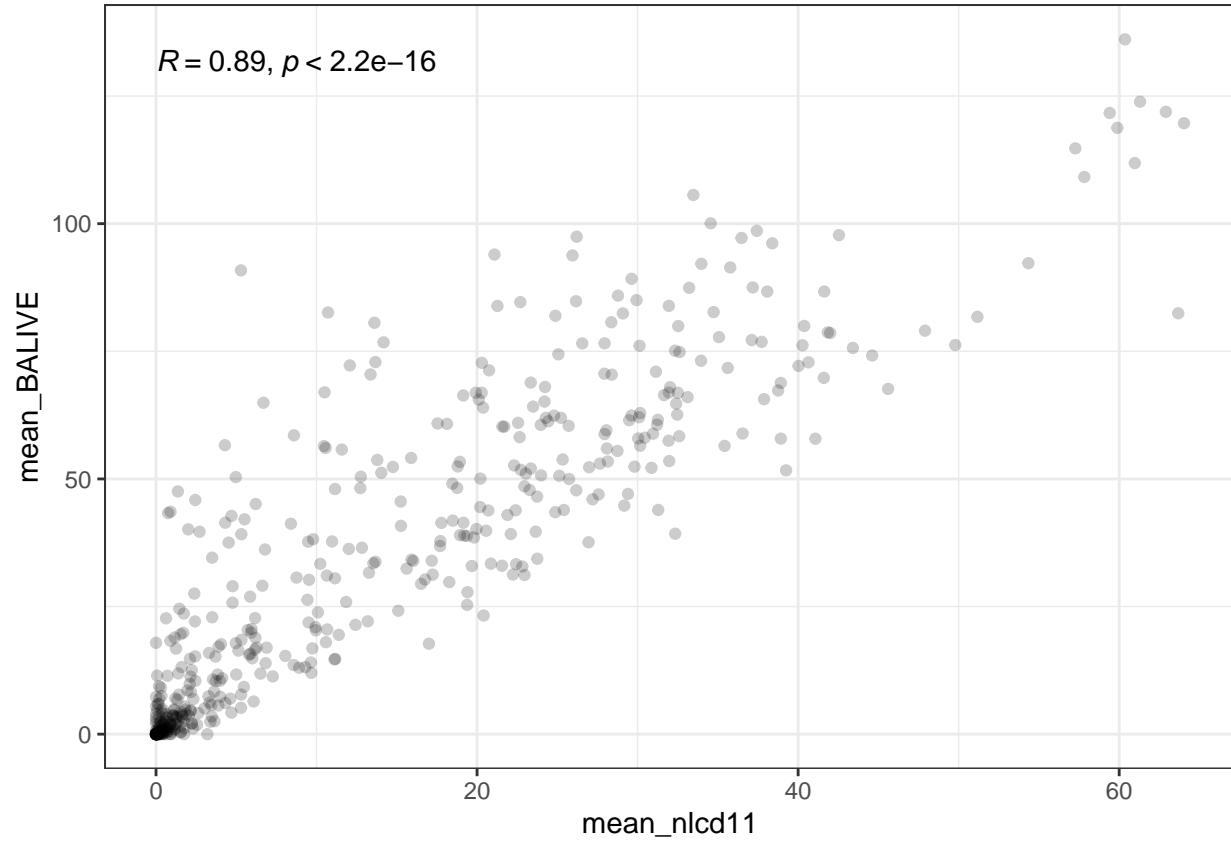
Area level correlation with Y variables

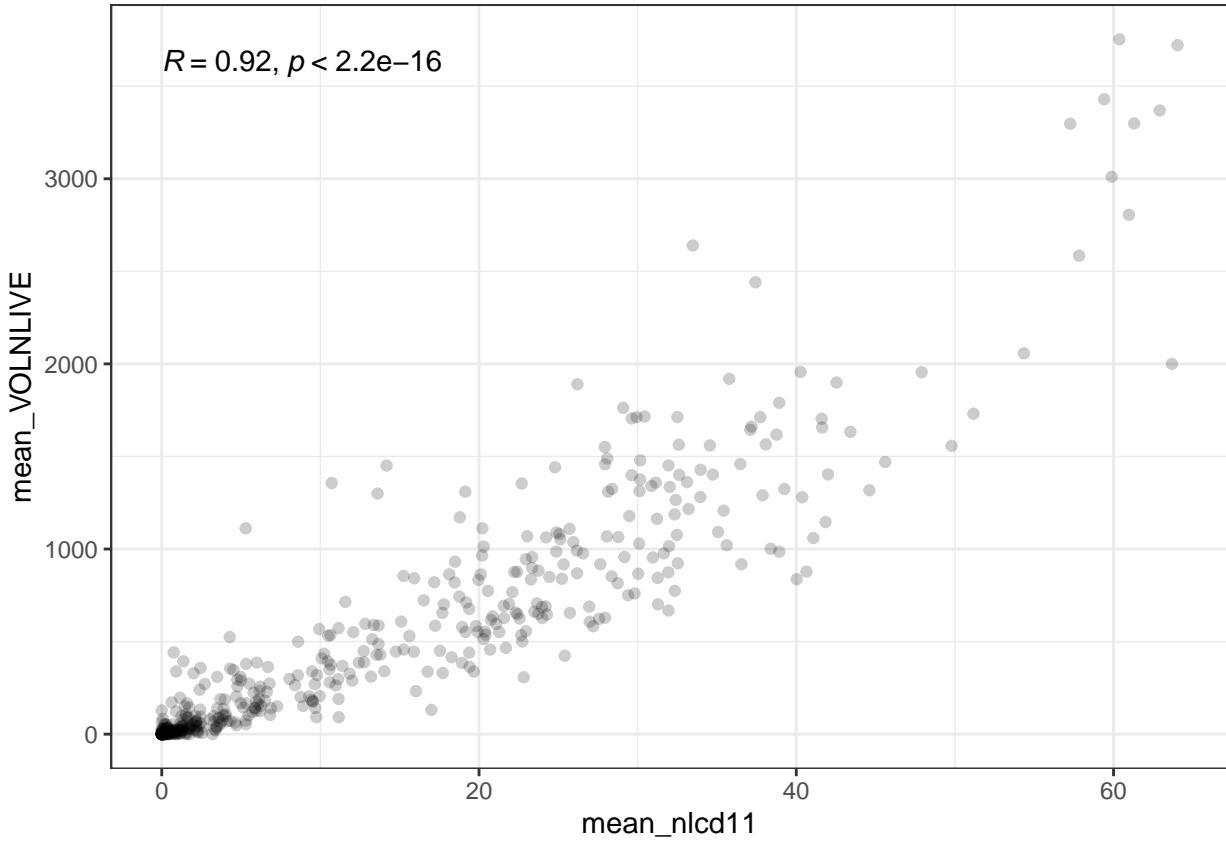
```
# Data wrangling
dat_small_area <- dat_small %>%
  group_by(subsection, section, province) %>%
  summarize(mean_BIOLIVE = mean(BIOLIVE_TPA),
            mean_BALIVE = mean(BALIVE_TPA),
            mean_CNTLIVE = mean(CNTLIVE_TPA),
            mean_VOLNLIVE = mean(VOLNLIVE_TPA),
            var_BIOLIVE = var(BIOLIVE_TPA),
            var_BALIVE = var(BALIVE_TPA),
            var_CNTLIVE = var(CNTLIVE_TPA),
            var_VOLNLIVE = var(VOLNLIVE_TPA),
            mean_nlcd11 = mean(nlcd11),
            var_nlcd11 = var(nlcd11))
```

```
## `summarise()` regrouping output by 'subsection', 'section' (override with '.groups' argument)
```









These correlations are much, much better!

Sub-goal: Model area level with `tidymodels`

```
m333_area <- dat_small_area %>%
  filter(province == "M333")
m333 <- dat_small %>%
  filter(province == "M333")

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 + nlcd11 + (1 | subsection),
       data = m333)
```

```
hb_fit
```

```
m333_mean <- m333 %>%
  group_by(subsection) %>%
  summarize(mean_BIOLIVE = mean(BIOLIVE_TPA))
m333 <- m333 %>%
  left_join(m333_mean)

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(mean_BIOLIVE ~ 1 + nlcd11 + (1 | subsection),
       data = m333)
```

```
hb_fit
```

```
# Let's plot the means
m333 <- dat_small %>%
  filter(province == "M333")
set.seed(13)
m333_nested <- m333 %>%
  mutate(id = subsection) %>%
  group_by(subsection) %>%
  nest()
means <- data.frame()
for (j in 1:length(unique(m333$subsection))) {
  for (i in 1:1000) {
    means[j, i] <- sample_n(
      m333_nested[[2]][[j]],
      size = length(m333_nested[[2]][[j]]$BIOLIVE_TPA),
      replace = TRUE
    ) %>%
      summarize(mean = mean(BIOLIVE_TPA)) %>%
      select(mean)
  }
}
library(matrixStats)
sds <- means %>%
  as.matrix() %>%
  rowSds()
```

```
hb_fit_df <- data.frame(
```

```

fitted = hb_fit$fit$fitted.values,
se = predict(object = hb_fit, new_data = m333_area, type = "pred_int"),
true = m333_area$mean_BIOLIVE,
subsection = m333_area$subsection,
mean_lower = m333_area$mean_BIOLIVE - 1.96*sds,
mean_upper = m333_area$mean_BIOLIVE + 1.96*sds
)

# Load sds
# conf <- read_csv("sd_df.csv")

# 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)) %>%
  # left_join(conf, by = c("subsection" = "id")) %>%
  mutate(subsection = fct_reorder(subsection, true)) %>%
  ggplot(aes(x = subsection,
             y = fitted)) +
  geom_point(aes(color = "goldenrod"),
             alpha = 0.75,
             position = position_nudge(x = -0.1)) +
  geom_point(
    aes(y = true, color = "forestgreen"),
    alpha = 0.75,
    position = position_nudge(x = 0.1)
  ) +
  geom_errorbar(
    mapping = aes(
      ymin = se..pred_lower,
      ymax = se..pred_upper
    ),
    position = position_nudge(x = -0.1),
    color = "goldenrod"
  ) +
  geom_errorbar(
    mapping = aes(
      ymin = mean_lower,
      ymax = mean_upper
    ),
    position = position_nudge(x = 0.1),
    color = "forestgreen"
  ) +
  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")
```