

# Stem density by size class

Tim White

## Packages, data, and functions

```
library(car)
library(tidyverse)

data_stems <- read_csv("data/data_stems.csv") %>%
  mutate(plot_id = as.factor(plot_id),
         size_class = as.factor(size_class),
         harvested = as.factor(harvested),
         vegetation_type = as.factor(vegetation_type),
         milpa = as.factor(milpa))
data_plots <- read_csv("data/data_plots.csv") %>%
  mutate(plot_id = as.factor(plot_id),
         harvested = as.factor(harvested),
         vegetation_type = as.factor(vegetation_type),
         milpa = as.factor(milpa))

compute_bootstrap_interval <- function(var, statistic = mean, num_iters = 50000,
                                       percentiles = c(0.025, 0.975)) {
  boot_samples <- numeric(num_iters)

  for (i in 1:num_iters) {
    boot_samples[i] <- statistic(sample(var, size = length(var), replace = TRUE))
  }

  return(quantile(boot_samples, percentiles))
}

plot_theme <- theme(plot.background = element_rect(fill = "white"),
                    plot.title = element_blank(),
                    plot.subtitle = element_text(family="sans", face="plain"),
                    axis.title.x = element_text(family="sans", face="bold"),
                    axis.title.y = element_text(family="sans", face="bold"),
                    axis.text.x = element_text(family="sans", face="plain"),
                    axis.text.y = element_text(family="sans", face="plain"),
                    panel.background = element_rect(fill="white"),
                    panel.grid.major.x = element_blank(),
                    panel.grid.major.y = element_line(color="gainsboro"),
                    panel.grid.minor = element_blank(),
                    axis.ticks = element_blank(),
                    legend.background = element_rect(color="black", fill = "white"),
                    legend.position = c(0.9, 0.85),
```

```
legend.title = element_text(color = "black", face = "bold", hjust = 0.5),
legend.text = element_text(color = "black"))
```

## Figures

### Average stem density by size class

Compute bootstrap intervals and organize data

```
harvested <- data_plots %>%
  filter(harvested == "yes") %>%
  select(stemden_seedlings, stemden_saplings,
         starts_with("stemden_trees"))
unharvested <- data_plots %>%
  filter(harvested == "no") %>%
  select(stemden_seedlings, stemden_saplings,
         starts_with("stemden_trees"))

set.seed(1)
boot_intervals <- bind_rows(compute_bootstrap_interval(unharvested$stemden_seedlings),
                             compute_bootstrap_interval(unharvested$stemden_saplings),
                             compute_bootstrap_interval(unharvested$stemden_trees05to09),
                             compute_bootstrap_interval(unharvested$stemden_trees10to14),
                             compute_bootstrap_interval(unharvested$stemden_trees15plus),
                             compute_bootstrap_interval(harvested$stemden_seedlings),
                             compute_bootstrap_interval(harvested$stemden_saplings),
                             compute_bootstrap_interval(harvested$stemden_trees05to09),
                             compute_bootstrap_interval(harvested$stemden_trees10to14),
                             compute_bootstrap_interval(harvested$stemden_trees15plus))

stemden_mean_data <- data_plots %>%
  select(harvested, stemden_seedlings,
         stemden_saplings, starts_with("stemden_trees")) %>%
  select(-contains("_count")) %>%
  group_by(harvested) %>%
  summarize_all(mean) %>%
  pivot_longer(cols = starts_with("stemden"), names_prefix = "stemden_",
               names_to = "size_class", values_to = "mean") %>%
  bind_cols(boot_intervals) %>%
  select(harvested, size_class, lower = `2.5%`, mean, upper = `97.5%`) %>%
  mutate(size_class = as.factor(size_class)) %>%
  mutate(harvested = fct_recode(harvested,
                                "Harvested" = "yes", "Unharvested" = "no"))

stemden_mean_data_noseedlings <- stemden_mean_data %>%
  filter(size_class != "seedlings") %>%
  mutate(size_class =
         fct_recode(size_class,
                     "Saplings (0-4 cm DBH)" = "saplings",
                     "Trees (5-9 cm DBH)" = "trees05to09",
                     "Trees (10-14 cm DBH)" = "trees10to14",
```

```

                                "Trees (15+ cm DBH)" = "trees15plus"))
stemden_mean_data_seedlings <- stemden_mean_data %>%
  filter(size_class == "seedlings") %>%
  mutate(size_class = fct_recode(size_class,
                                "Seedlings (0 cm DBH)" = "seedlings"))

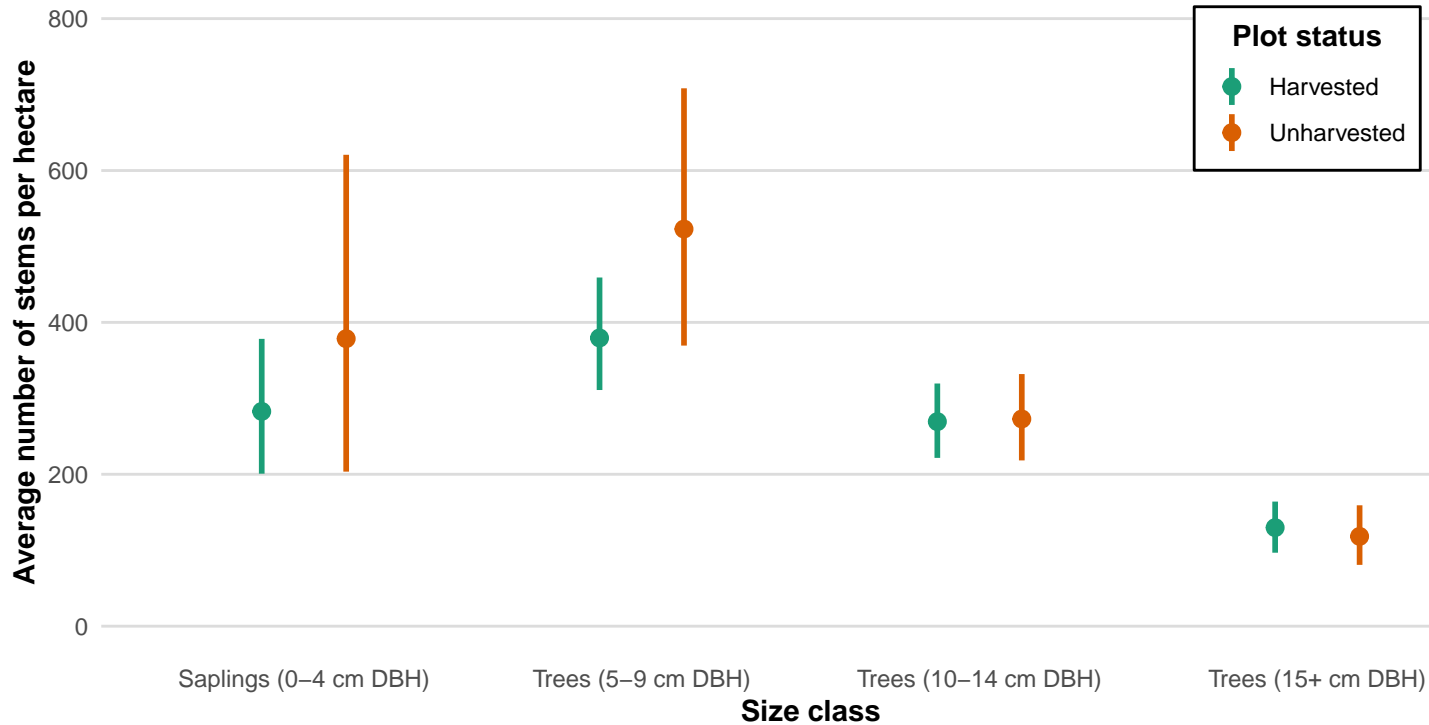
```

## Pointrange plots

```

# All size classes except seedlings
stemden_sizeclass_fig1_noseedlings <- stemden_mean_data_noseedlings %>%
  ggplot() +
    geom_pointrange(aes(x = size_class, col = fct_relevel(harvested, "Harvested", "Unharvested"),
                        ymin = lower, y = mean, ymax = upper),
                    position = position_dodge(width = 0.5), size = 0.5, linewidth = 1) +
    plot_theme + theme(legend.position.inside = c(0.85, 0.85),
                      legend.title = element_text(hjust = 0.5)) +
    ylim(0, 800) +
    scale_color_brewer(palette = "Dark2") +
    labs(x = "Size class", y = "Average number of stems per hectare") +
    guides(col=guide_legend(title = "Plot status"))
ggsave("figures/stemden_sizeclass_fig1_noseedlings.png", height = 4, width = 8)
stemden_sizeclass_fig1_noseedlings

```



```

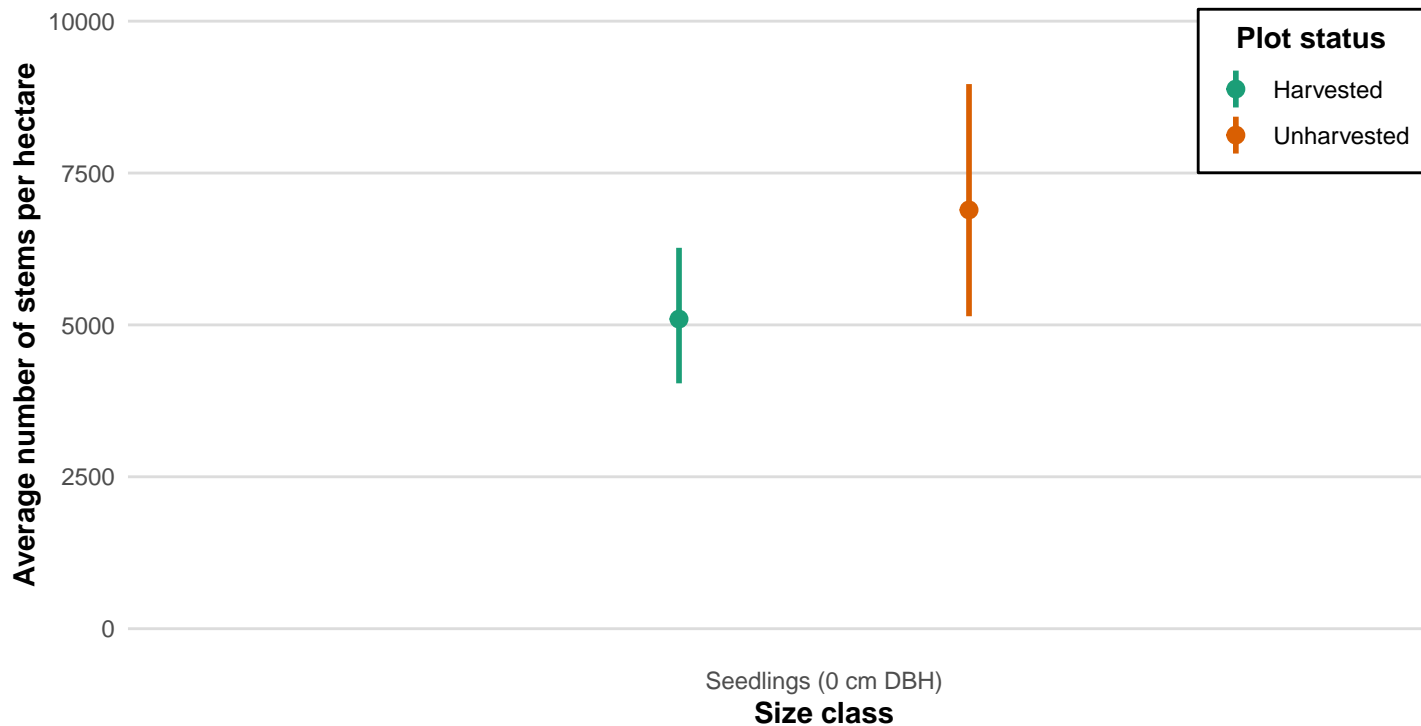
# Seedlings only
stemden_sizeclass_fig1_seedlings <- stemden_mean_data_seedlings %>%
  ggplot() +

```

```

geom_pointrange(aes(x = size_class, col = fct_relevel(harvested, "Harvested", "Unharvested"),
  ymin = lower, y = mean, ymax = upper),
  position = position_dodge(width = 0.5), size = 0.5, linewidth = 1) +
plot_theme + theme(legend.position.inside = c(0.85, 0.85),
  legend.title = element_text(hjust = 0.5)) +
ylim(0, 10000) +
scale_color_brewer(palette = "Dark2") +
labs(x = "Size class", y = "Average number of stems per hectare") +
guides(col=guide_legend(title = "Plot status"))
ggsave("figures/stemden_sizeclass_fig1_seedlings.png", height = 4, width = 8)
stemden_sizeclass_fig1_seedlings

```



## Bar plots

```

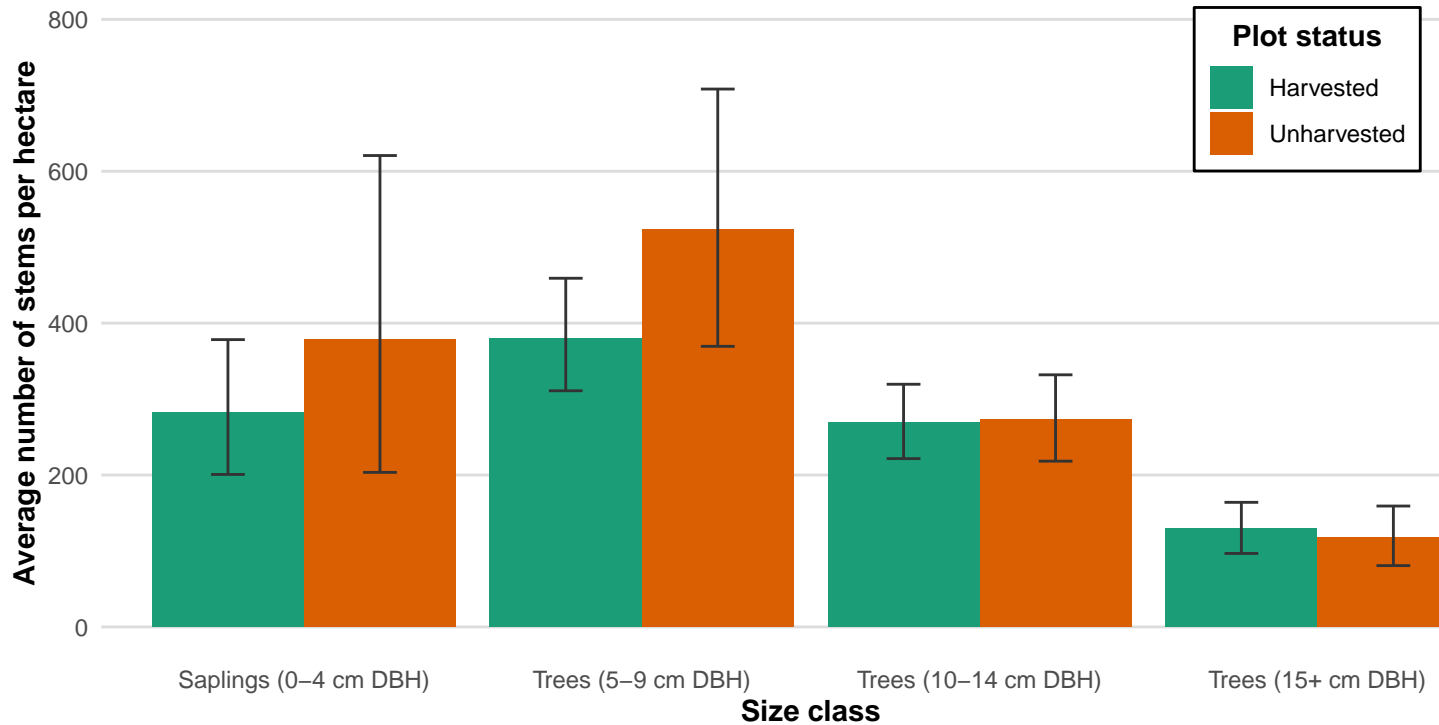
# All size classes except seedlings
stemden_sizeclass_fig2_noseedlings <- stemden_mean_data_noseedlings %>%
  ggplot(aes(x = size_class,
    fill = fct_relevel(harvested, "Harvested", "Unharvested"))) +
  geom_col(aes(y = mean),
    position = "dodge") +
  geom_errorbar(aes(x = size_class, ymin = lower, ymax = upper), col = "gray20",
    position = position_dodge(width = 0.9), width = 0.2, show.legend = FALSE) +
plot_theme + theme(legend.position.inside = c(0.85, 0.85),
  legend.title = element_text(hjust = 0.5)) +
ylim(0, 800) +
scale_fill_brewer(palette = "Dark2") +
labs(x = "Size class", y = "Average number of stems per hectare") +

```

```

    guides(fill=guide_legend(title = "Plot status"))
ggsave("figures/stemden_sizeclass_fig2_noseedlings.png", height = 4, width = 8)
stemden_sizeclass_fig2_noseedlings

```

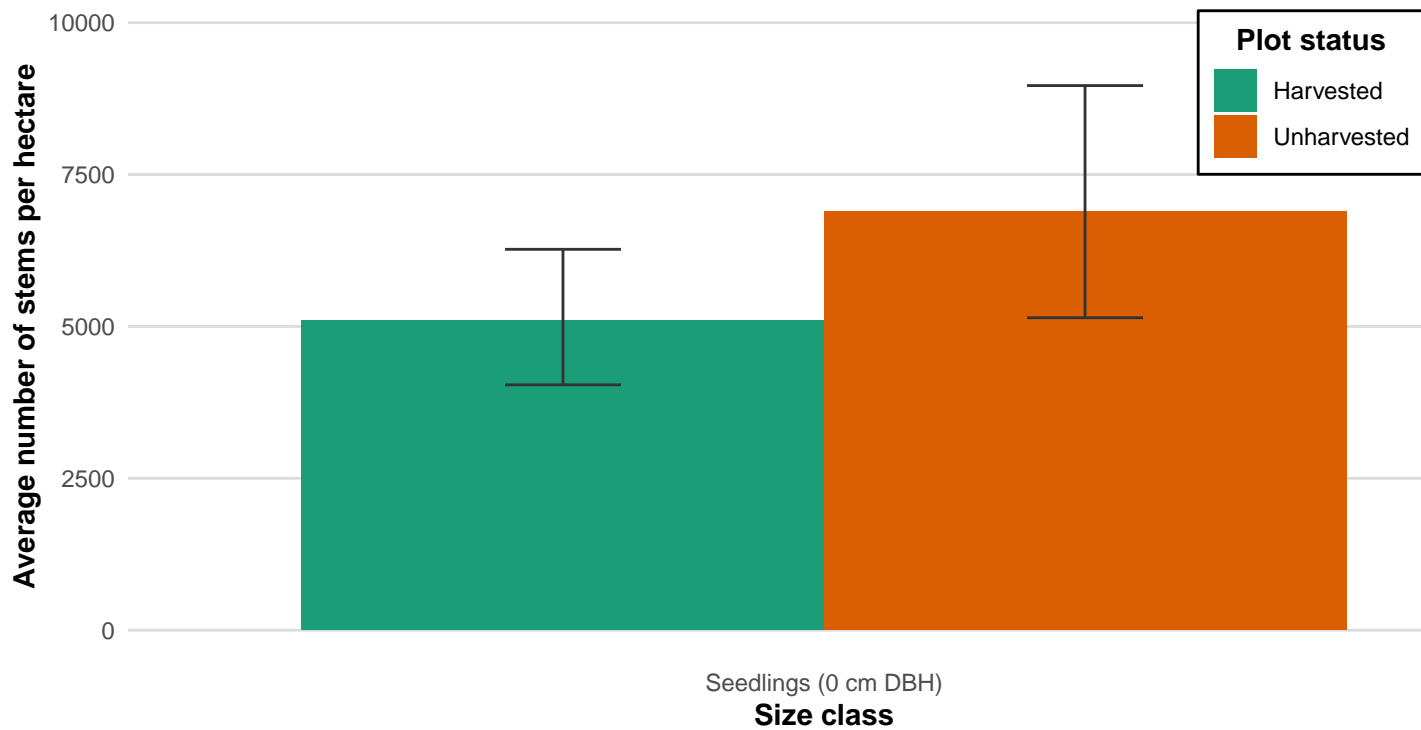


```

# Seedlings only
stemden_sizeclass_fig2_seedlings <- stemden_mean_data_seedlings %>%
  ggplot(aes(x = size_class,
             fill = fct_relevel(harvested, "Harvested", "Unharvested"))) +
  geom_col(aes(y = mean),
           position = "dodge") +
  geom_errorbar(aes(x = size_class, ymin = lower, ymax = upper), col = "gray20",
               position = position_dodge(width = 0.9), width = 0.2, show.legend = FALSE) +
  plot_theme + theme(legend.position.inside = c(0.85, 0.85),
                    legend.title = element_text(hjust = 0.5)) +

  ylim(0, 10000) +
  scale_fill_brewer(palette = "Dark2") +
  labs(x = "Size class", y = "Average number of stems per hectare") +
  guides(fill=guide_legend(title = "Plot status"))
ggsave("figures/stemden_sizeclass_fig2_seedlings.png", height = 4, width = 8)
stemden_sizeclass_fig2_seedlings

```



## Distribution of stem densities by size class

Organize data

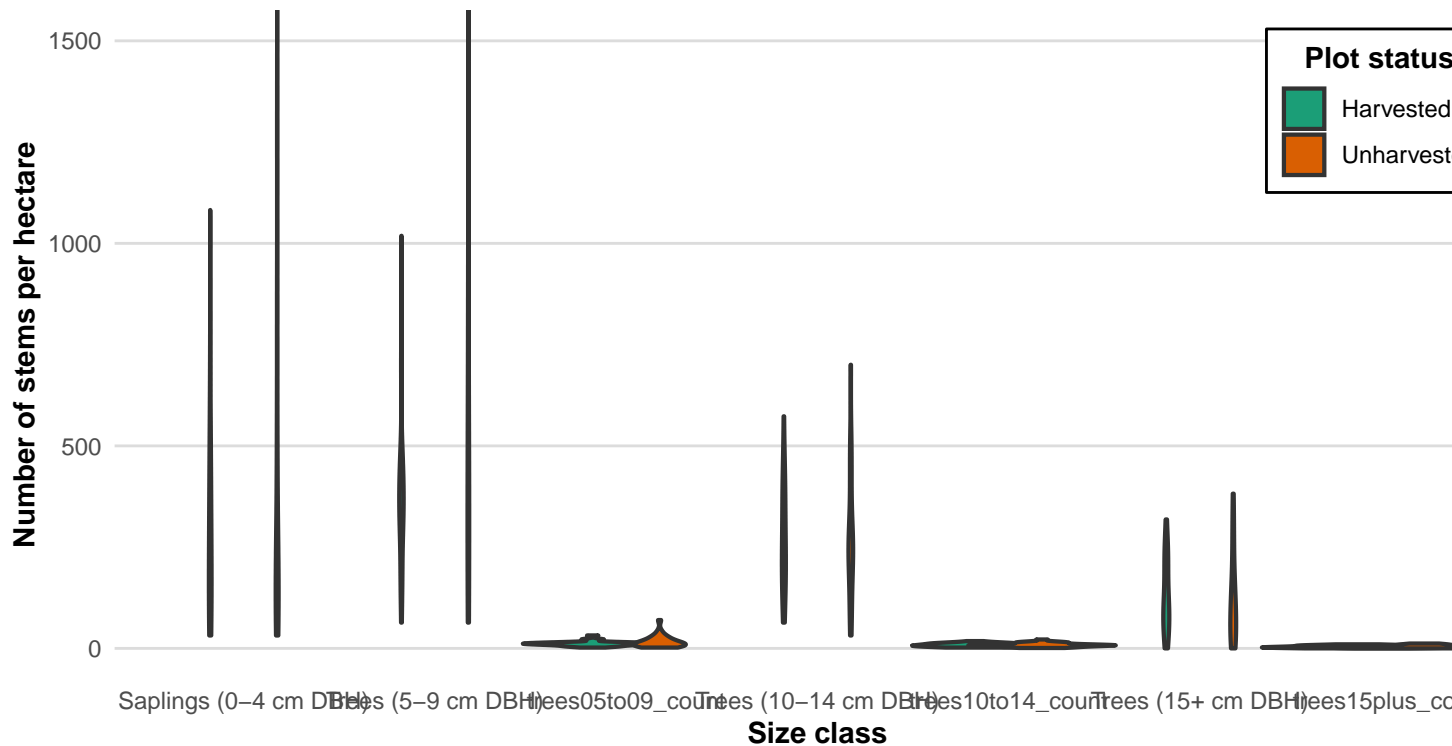
```
stemden_dist_data <- data_plots %>%
  select(harvested, stemden_seedlings,
         stemden_saplings, starts_with("stemden_trees")) %>%
  pivot_longer(cols = starts_with("stemden"),
               names_prefix = "stemden_",
               names_to = "size_class",
               values_to = "num_stems") %>%
  mutate(size_class = as.factor(size_class)) %>%
  mutate(harvested = fct_recode(harvested,
                                "Harvested" = "yes",
                                "Unharvested" = "no"))

stemden_dist_data_noseedlings <- stemden_dist_data %>%
  filter(size_class != "seedlings") %>%
  mutate(size_class =
    fct_recode(size_class,
               "Saplings (0-4 cm DBH)" = "saplings",
               "Trees (5-9 cm DBH)" = "trees05to09",
               "Trees (10-14 cm DBH)" = "trees10to14",
               "Trees (15+ cm DBH)" = "trees15plus"))

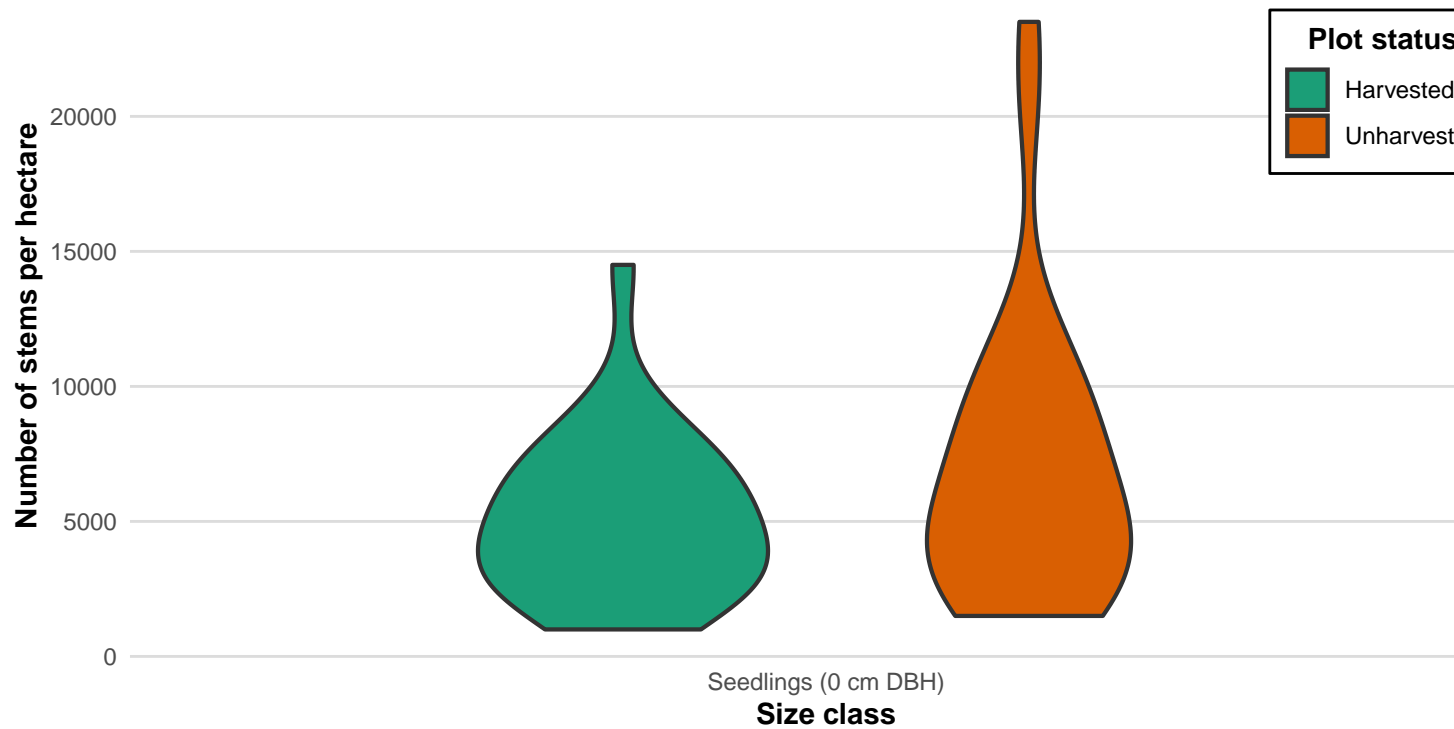
stemden_dist_data_seedlings <- stemden_dist_data %>%
  filter(size_class == "seedlings") %>%
  mutate(size_class = fct_recode(size_class,
                                "Seedlings (0 cm DBH)" = "seedlings"))
```

## Violin plots

```
# All size classes except seedlings
stemden_dist_data_noseedlings %>%
  ggplot() +
    geom_violin(aes(x = size_class, y = num_stems,
                    fill = fct_relevel(harvested, "Harvested", "Unharvested")),
               scale = "area", width = 2, position = position_dodge(width = 0.7),
               lwd = 0.75, kernel = "gaussian", adjust = 1.25) +
    plot_theme +
    scale_fill_brewer(palette = "Dark2") +
    coord_cartesian(ylim = c(0, 1500)) + # 2 outliers not shown
    labs(x = "Size class", y = "Number of stems per hectare") +
    guides(fill = guide_legend(title = "Plot status"))
```



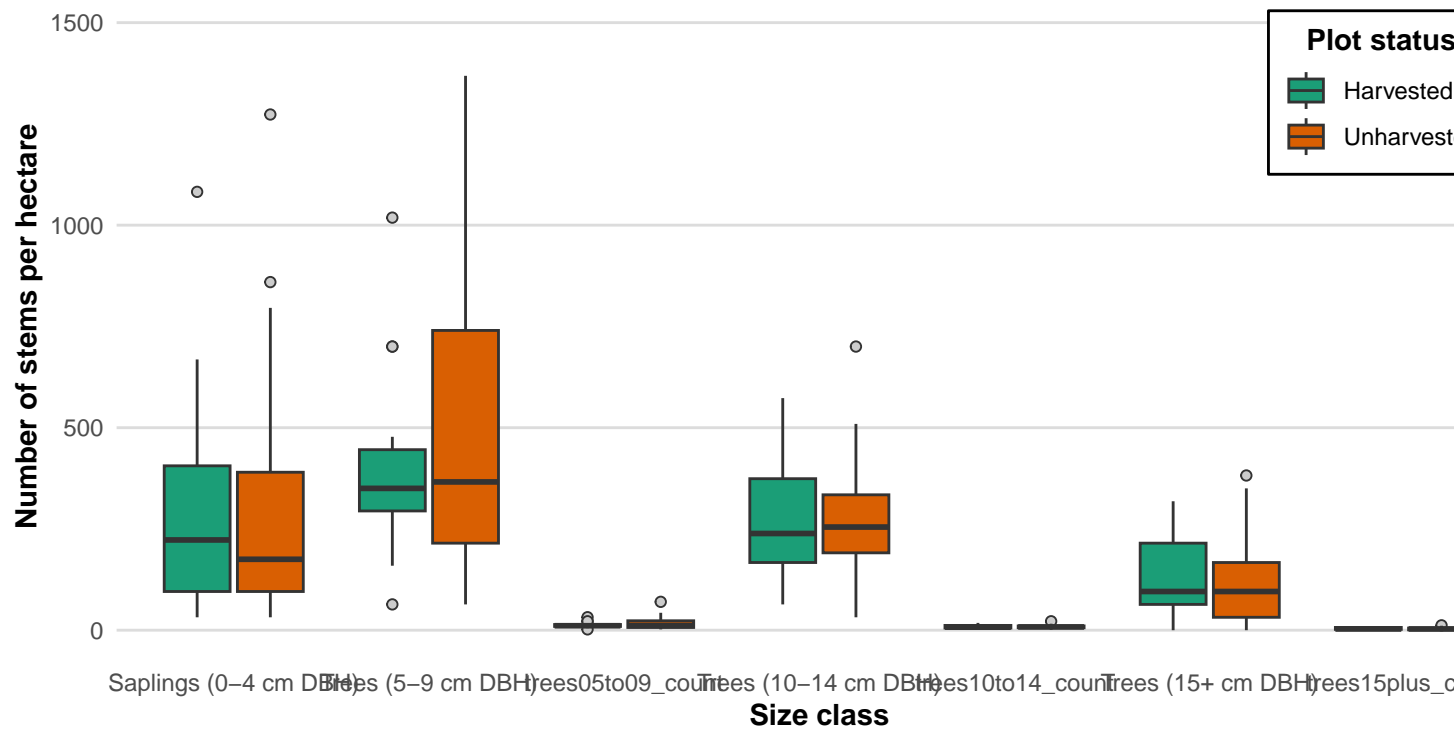
```
# Seedlings only
stemden_dist_data_seedlings %>%
  ggplot() +
    geom_violin(aes(x = size_class, y = num_stems,
                    fill = fct_relevel(harvested, "Harvested", "Unharvested")),
               scale = "area", width = 0.5, position = position_dodge(width = 0.7),
               lwd = 0.75, kernel = "gaussian", adjust = 1.25) +
    plot_theme +
    scale_fill_brewer(palette = "Dark2") +
    labs(x = "Size class", y = "Number of stems per hectare") +
    guides(fill = guide_legend(title = "Plot status"))
```



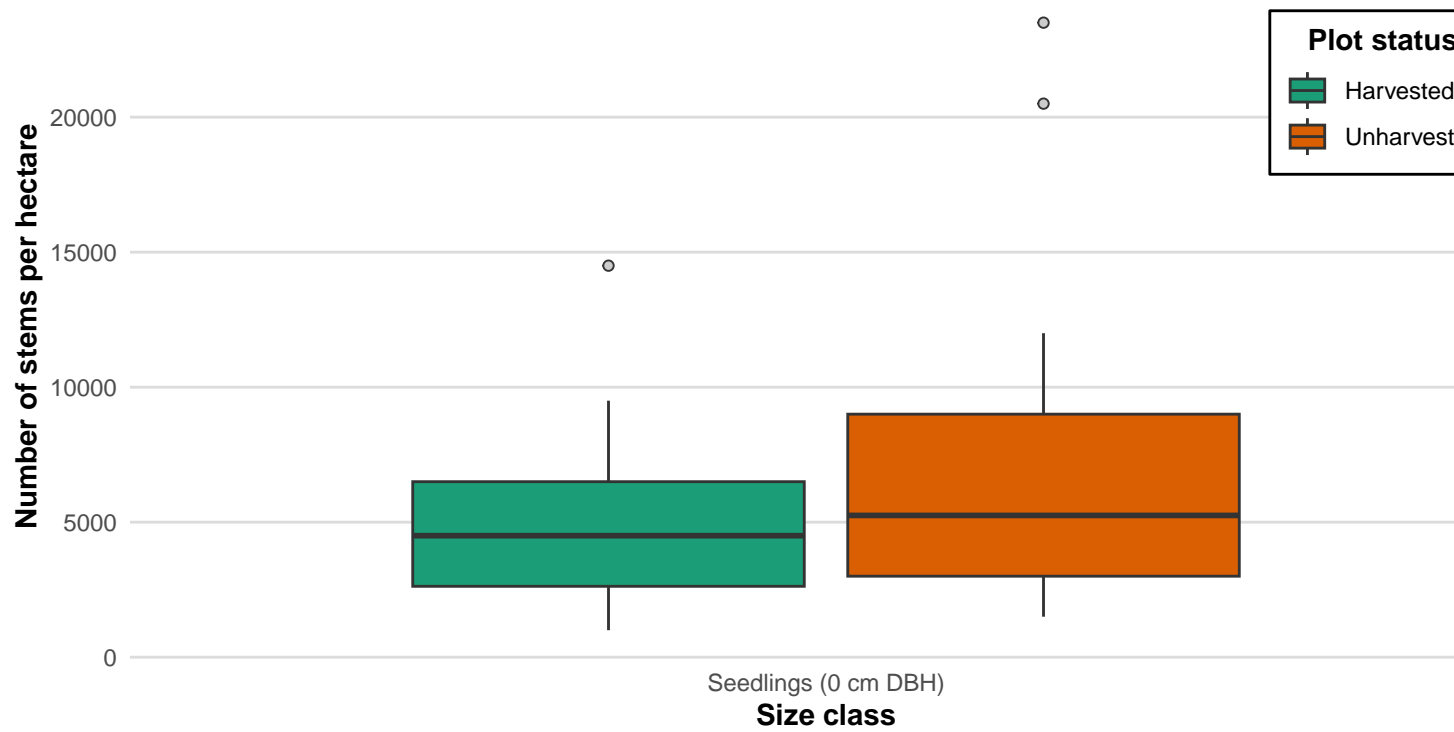
### Box plots

```
# All size classes except seedlings
stemden_sizeclass_fig3_noseedlings <- stemden_dist_data_noseedlings %>%
  ggplot() +
    geom_boxplot(aes(x = size_class, y = num_stems,
                     fill = fct_relevel(harvested, "Harvested", "Unharvested")),
                 outlier.shape = 21, outlier.color = "gray20",
                 outlier.fill = "gray80") +
    plot_theme +
    scale_fill_brewer(palette = "Dark2") +
    coord_cartesian(ylim = c(0, 1500)) + # 2 outliers not shown
    labs(x = "Size class", y = "Number of stems per hectare") +
    guides(fill = guide_legend(title = "Plot status"))
ggsave("figures/stemden_sizeclass_fig3_noseedlings.png", height = 4, width = 8)
stemden_sizeclass_fig3_noseedlings
```





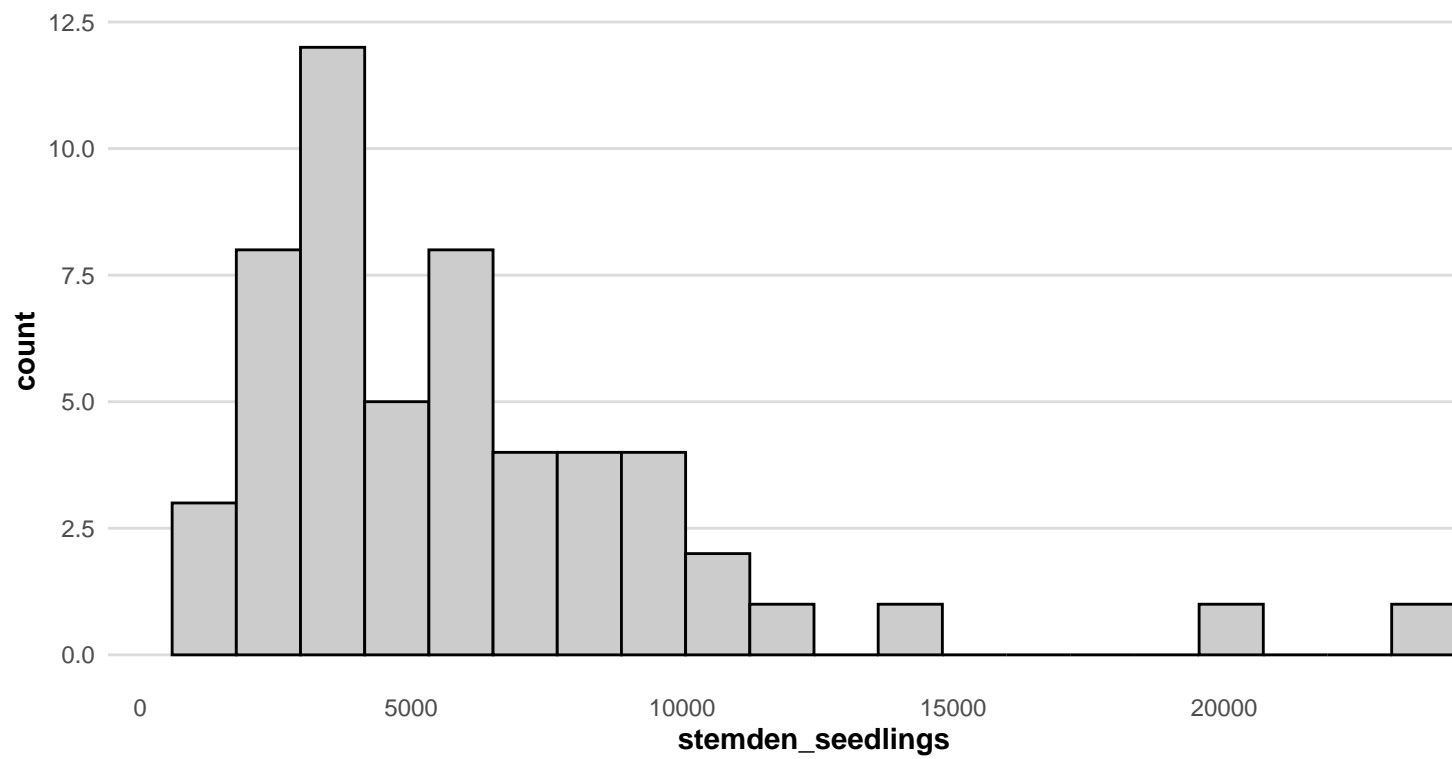
```
# Seedlings only
stemden_sizeclass_fig3_seedlings <- stemden_dist_data_seedlings %>%
  ggplot() +
  geom_boxplot(aes(x = size_class, y = num_stems,
                   fill = fct_relevel(harvested, "Harvested", "Unharvested")),
               outlier.shape = 21, outlier.color = "gray20",
               outlier.fill = "gray80") +
  plot_theme +
  scale_fill_brewer(palette = "Dark2") +
  labs(x = "Size class", y = "Number of stems per hectare") +
  guides(fill = guide_legend(title = "Plot status"))
ggsave("figures/stemden_sizeclass_fig3_seedlings.png", height = 4, width = 8)
stemden_sizeclass_fig3_seedlings
```



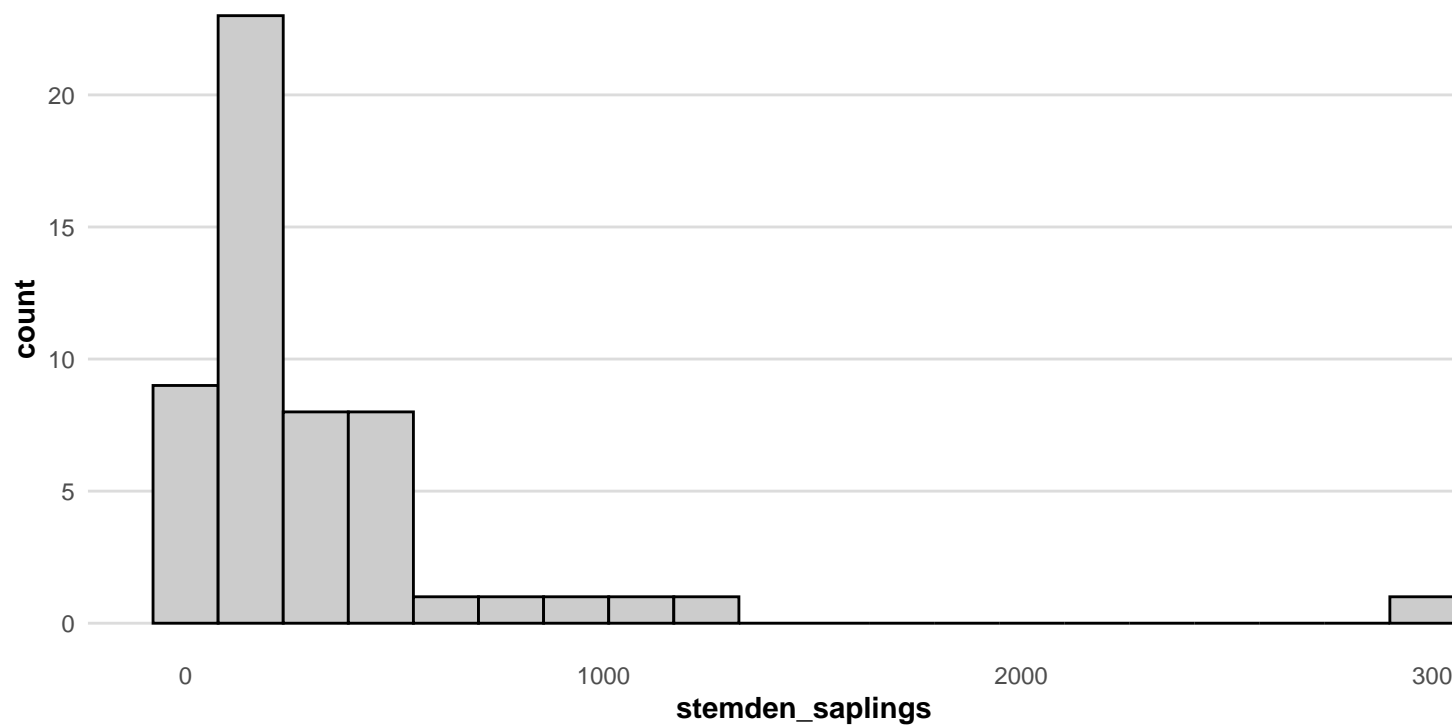
Stem density by size class for different covariates

Histograms, harvesting status

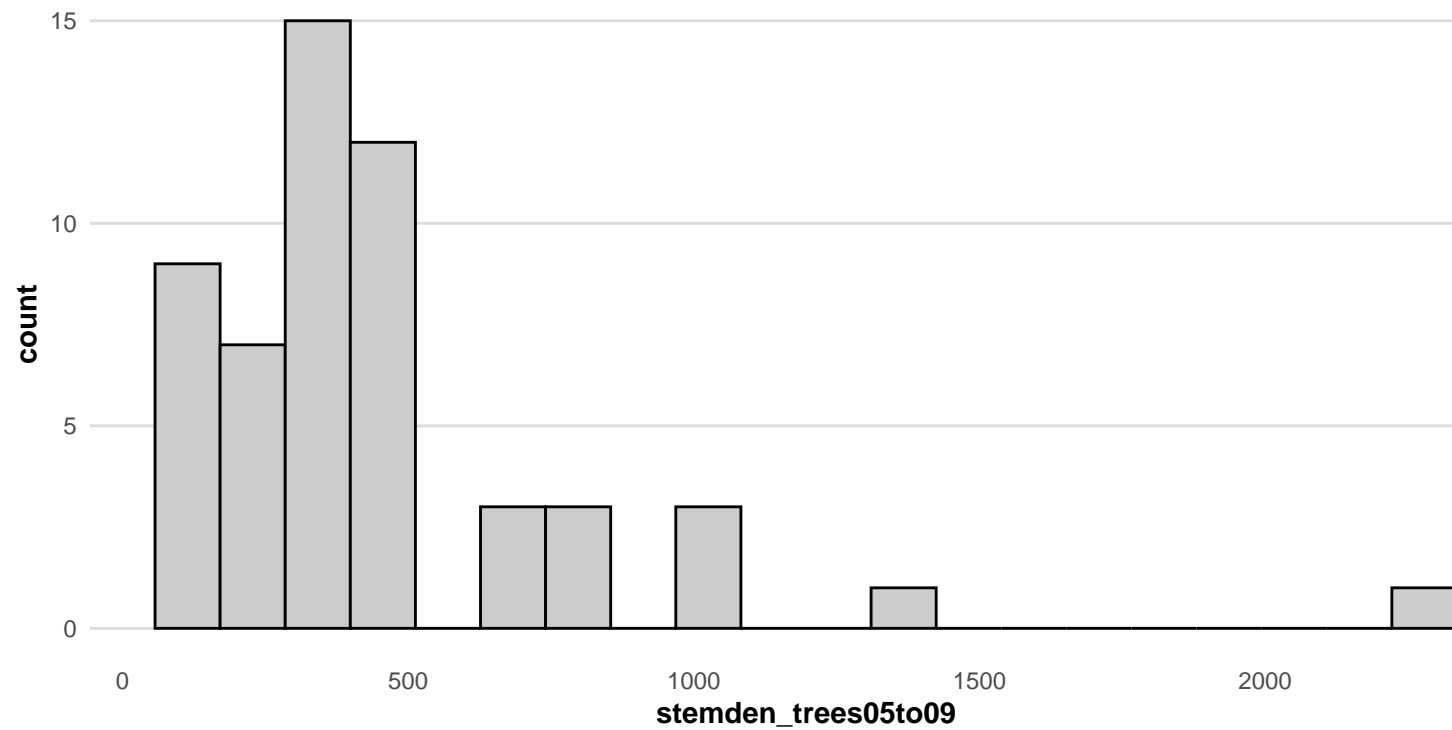
```
data_plots %>%
  ggplot() + geom_histogram(aes(x = stemden_seedlings), bins = 20,
                             col = "black", fill = "gray80") + plot_theme
```



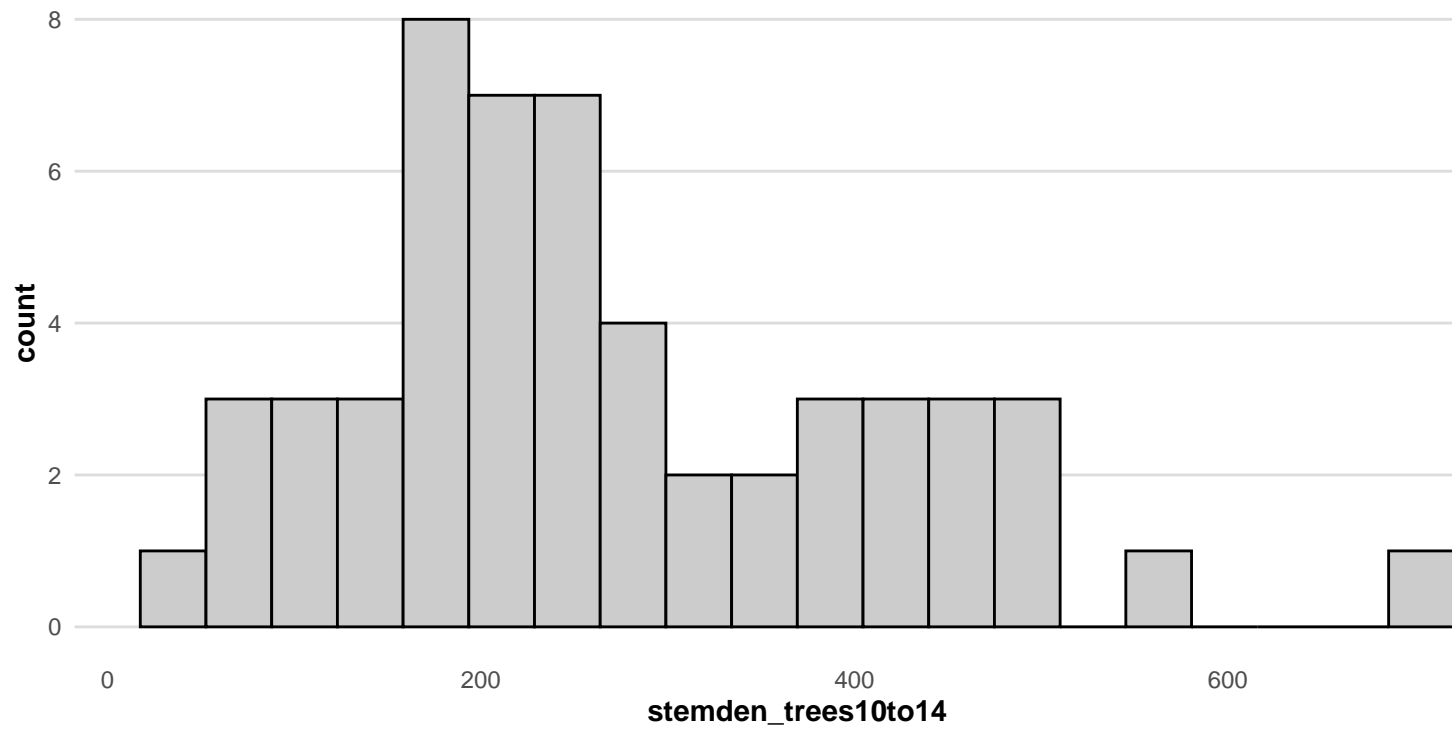
```
data_plots %>%
  ggplot() + geom_histogram(aes(x = stemden_saplings), bins = 20,
    col = "black", fill = "gray80") + plot_theme
```



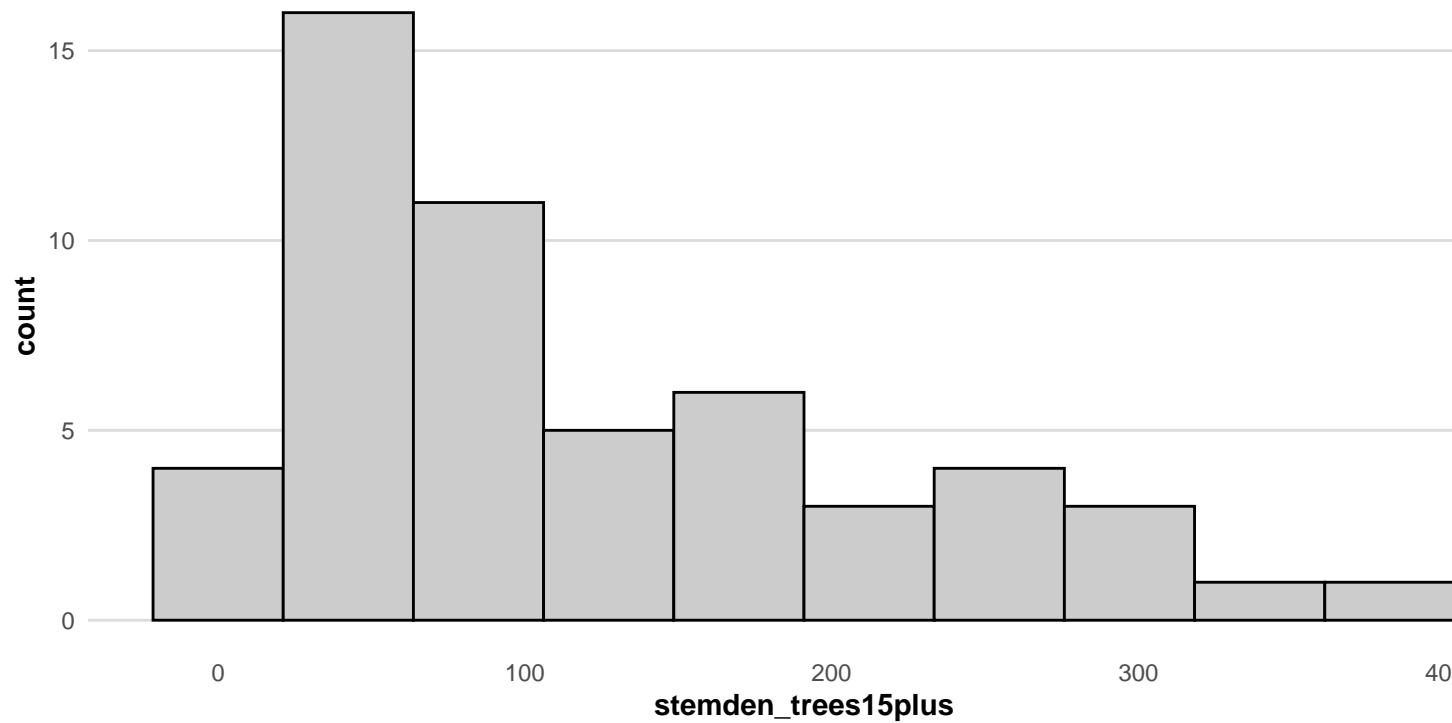
```
data_plots %>%
  ggplot() + geom_histogram(aes(x = stemden_trees05to09), bins = 20,
                             col = "black", fill = "gray80") + plot_theme
```



```
data_plots %>%
  ggplot() + geom_histogram(aes(x = stemden_trees10to14), bins = 20,
                             col = "black", fill = "gray80") + plot_theme
```



```
data_plots %>%
  ggplot() + geom_histogram(aes(x = stemden_trees15plus), bins = 10,
    col = "black", fill = "gray80") + plot_theme
```



## Five-number summaries, harvesting status

```
summary(data_plots$stemden_seedlings)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      1000   3000   4750   6028   7875   23500
```

```
data_plots %>% filter(harvested == "yes") %>% pull(stemden_seedlings) %>% summary()
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      1000   2625   4500   5096   6500   14500
```

```
data_plots %>% filter(harvested == "no") %>% pull(stemden_seedlings) %>% summary()
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      1500   3000   5250   6893   9000   23500
```

```
summary(data_plots$stemden_saplings)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      31.83   95.49  190.99  332.46  405.85 2992.11
```

```
data_plots %>% filter(harvested == "yes") %>% pull(stemden_saplings) %>% summary()
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      31.83   95.49  222.82  282.81  405.85 1082.25
```

```
data_plots %>% filter(harvested == "no") %>% pull(stemden_saplings) %>% summary()
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      31.83   95.49  175.07  378.56  389.93 2992.11
```

```
summary(data_plots$stemden_trees05to09)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      63.66  254.65  350.14  453.89  501.34 2228.17
```

```
data_plots %>% filter(harvested == "yes") %>% pull(stemden_trees05to09) %>% summary()
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      63.66  294.44  350.14  379.52  445.63 1018.59
```

```
data_plots %>% filter(harvested == "no") %>% pull(stemden_trees05to09) %>% summary()
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      63.66  214.86  366.06  522.94  740.07 2228.17
```

```
summary(data_plots$stemden_trees10to14)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    31.83  190.99  254.65  271.15  374.01  700.28
```

```
data_plots %>% filter(harvested == "yes") %>% pull(stemden_trees10to14) %>% summary()
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    63.66  167.11  238.73  269.34  374.01  572.96
```

```
data_plots %>% filter(harvested == "no") %>% pull(stemden_trees10to14) %>% summary()
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    31.83  190.99  254.65  272.84  334.23  700.28
```

```
summary(data_plots$stemden_trees15plus)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.00   31.83   95.49  123.79  190.99  381.97
```

```
data_plots %>% filter(harvested == "yes") %>% pull(stemden_trees15plus) %>% summary()
```

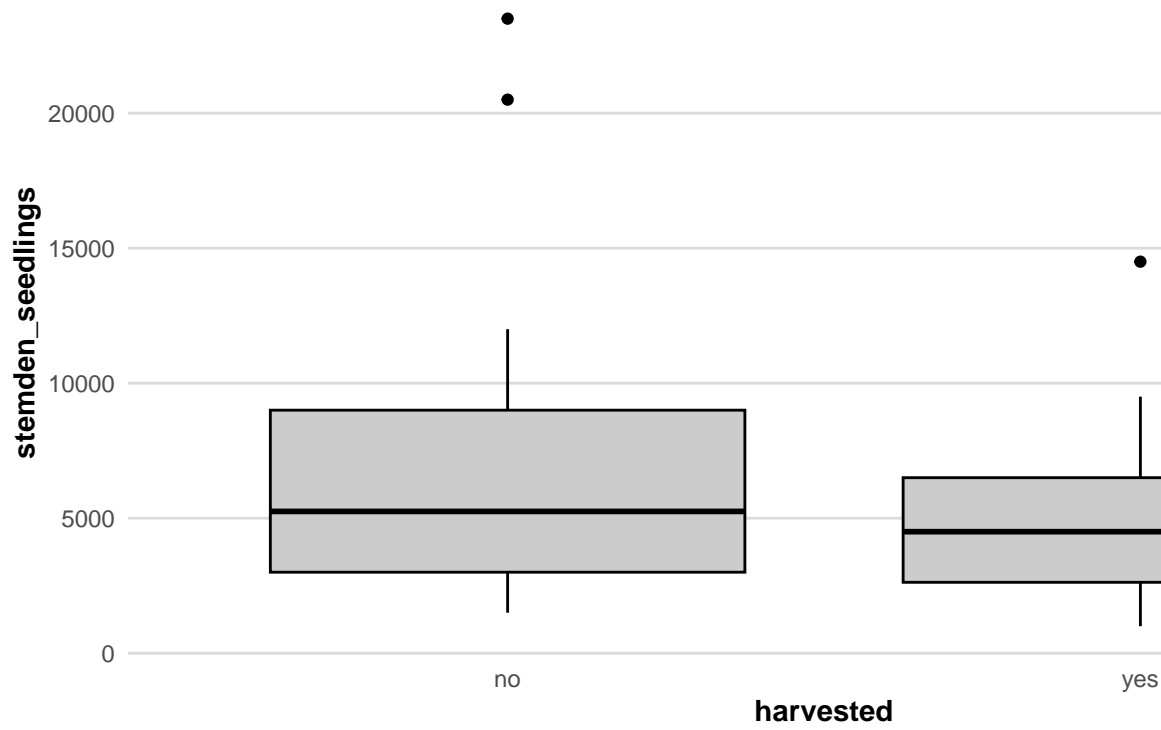
```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.00   63.66   95.49  129.77  214.86  318.31
```

```
data_plots %>% filter(harvested == "no") %>% pull(stemden_trees15plus) %>% summary()
```

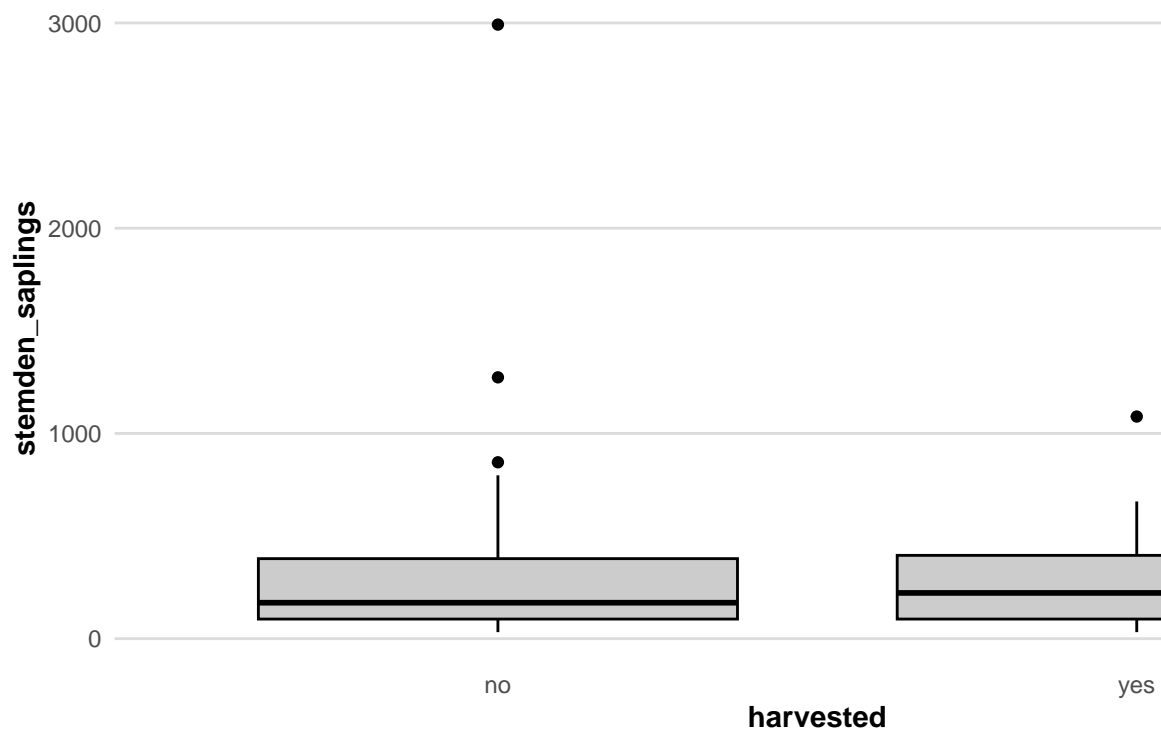
```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.00   31.83   95.49  118.23  167.11  381.97
```

Box plots, harvesting status

```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = harvested, y = stemden_seedlings),
    col = "black", fill = "gray80") + plot_theme
```

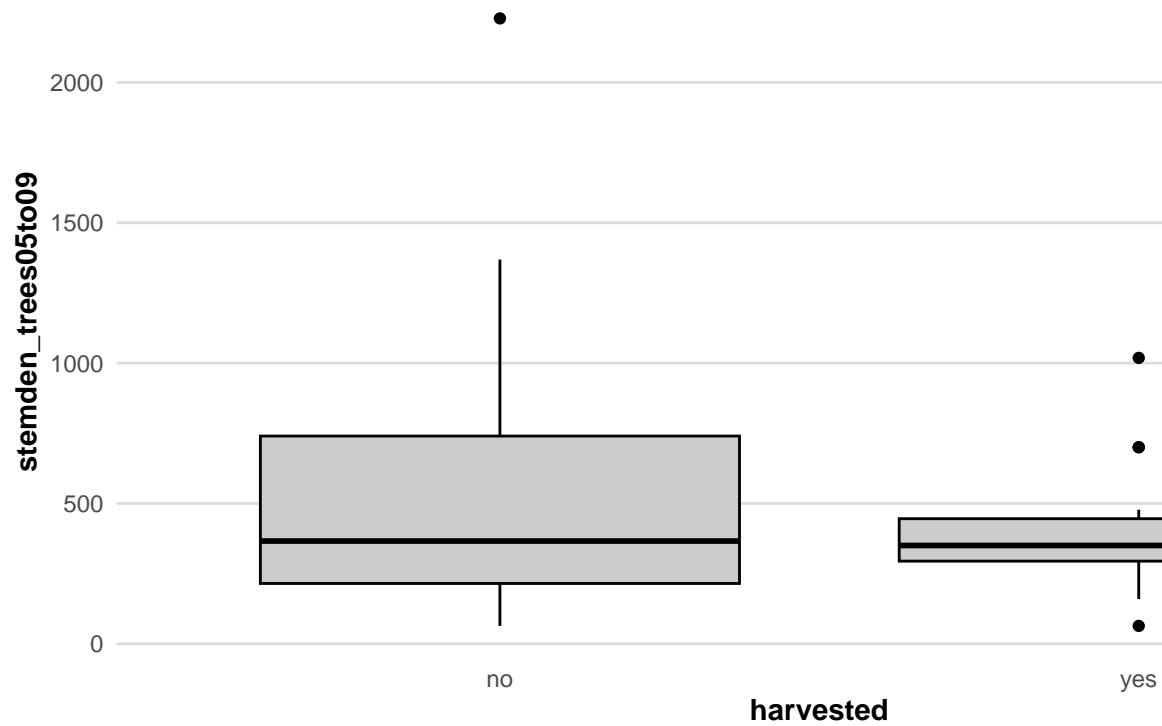


```
data_plots %>% ggplot() +  
  geom_boxplot(aes(x = harvested, y = stemden_saplings),  
    col = "black", fill = "gray80") + plot_theme
```

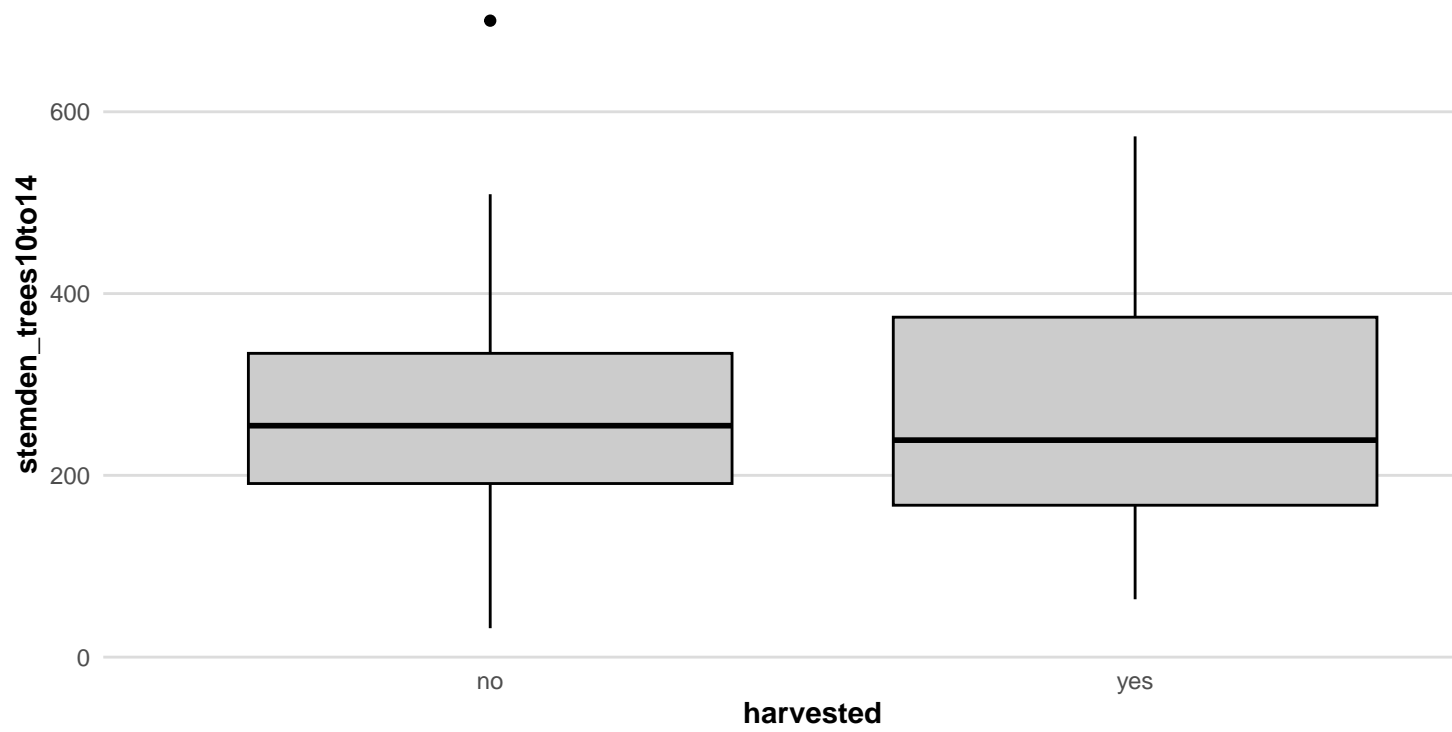




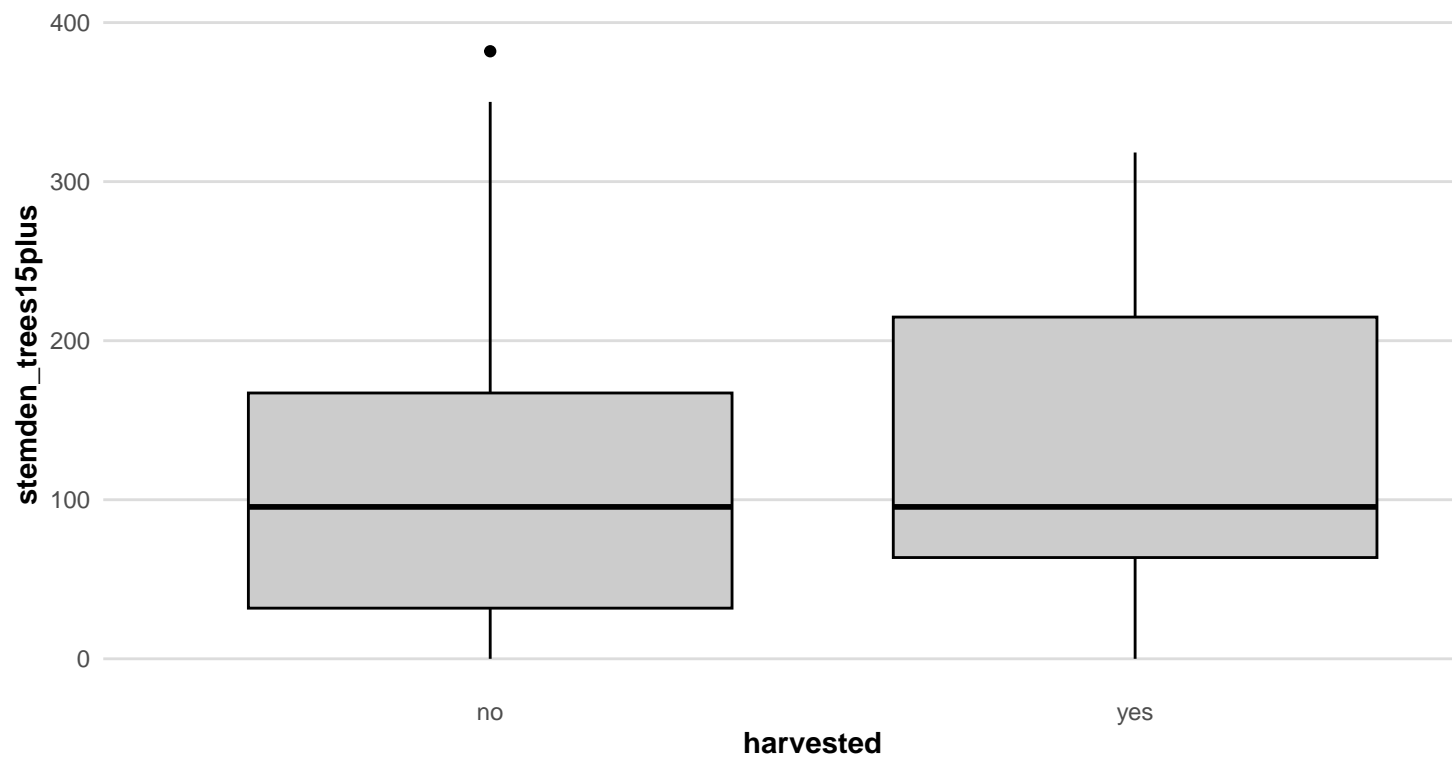
```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = harvested, y = stemden_trees05to09),
    col = "black", fill = "gray80") + plot_theme
```



```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = harvested, y = stemden_trees10to14),
    col = "black", fill = "gray80") + plot_theme
```

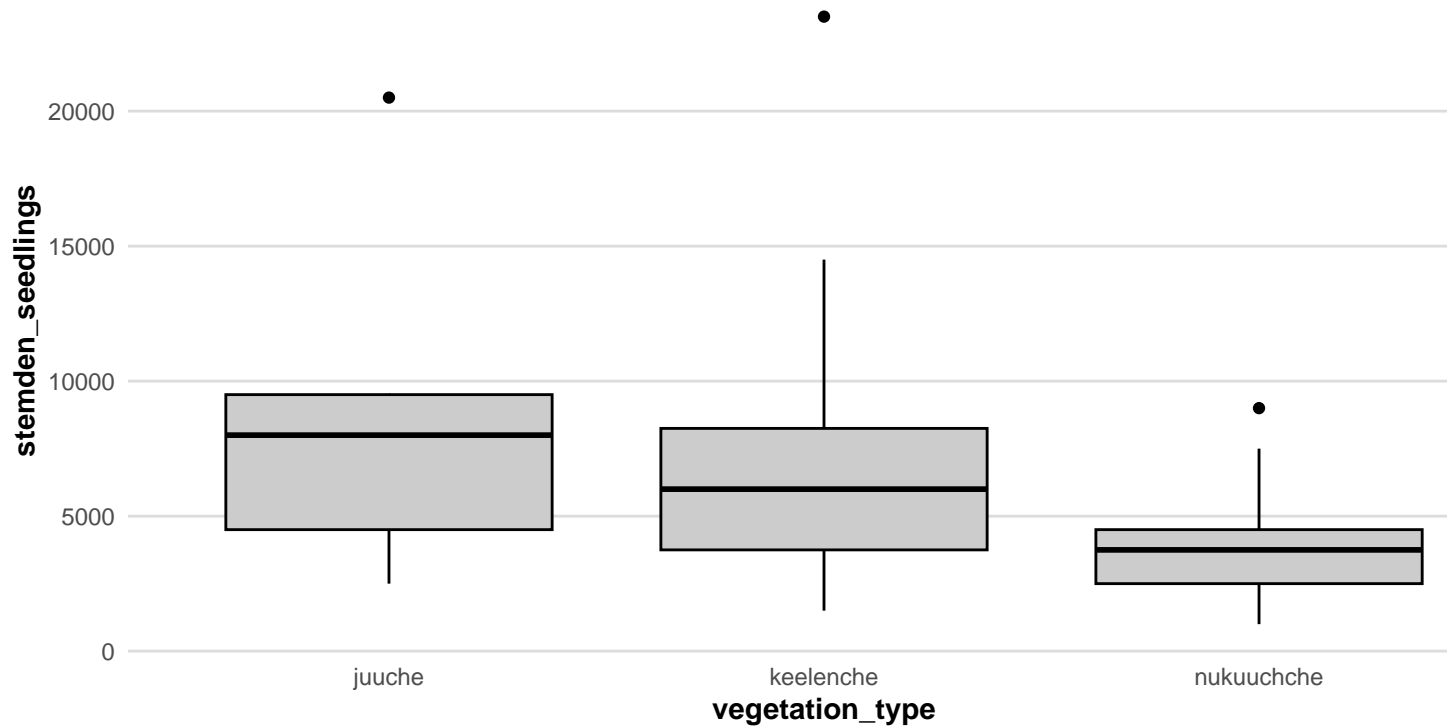


```
data_plots %>% ggplot() +  
  geom_boxplot(aes(x = harvested, y = stemden_trees15plus),  
    col = "black", fill = "gray80") + plot_theme
```

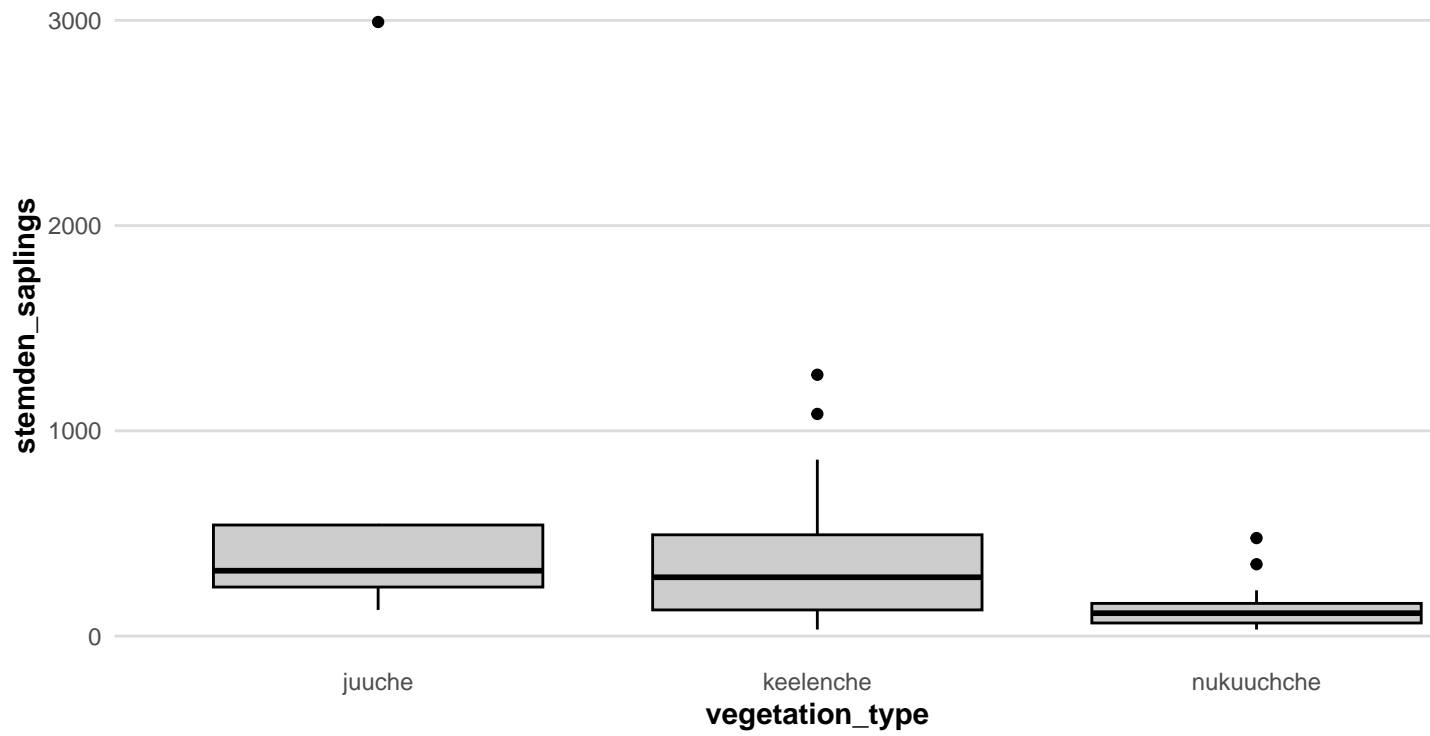


## Box plots, vegetation type

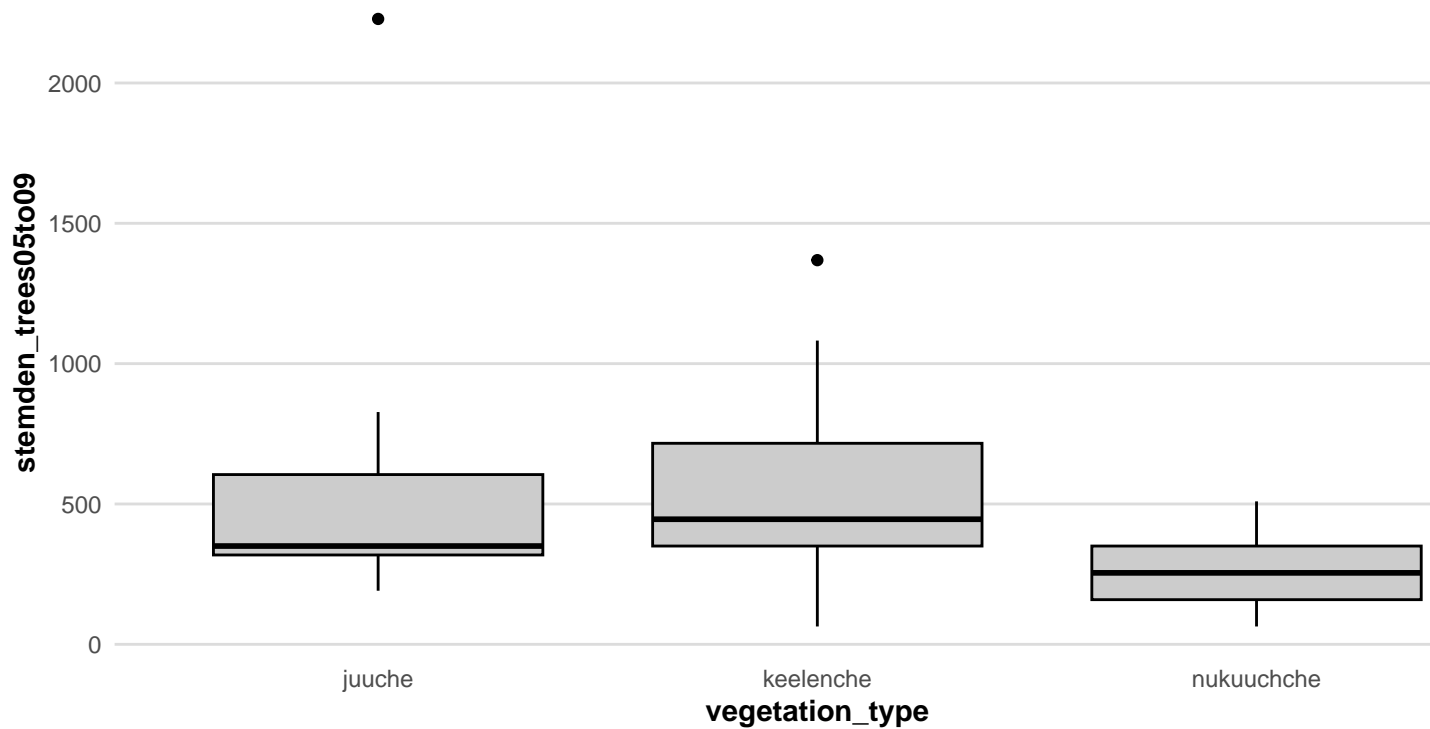
```
data_plots %>% ggplot() +  
  geom_boxplot(aes(x = vegetation_type, y = stemden_seedlings),  
    col = "black", fill = "gray80") + plot_theme
```



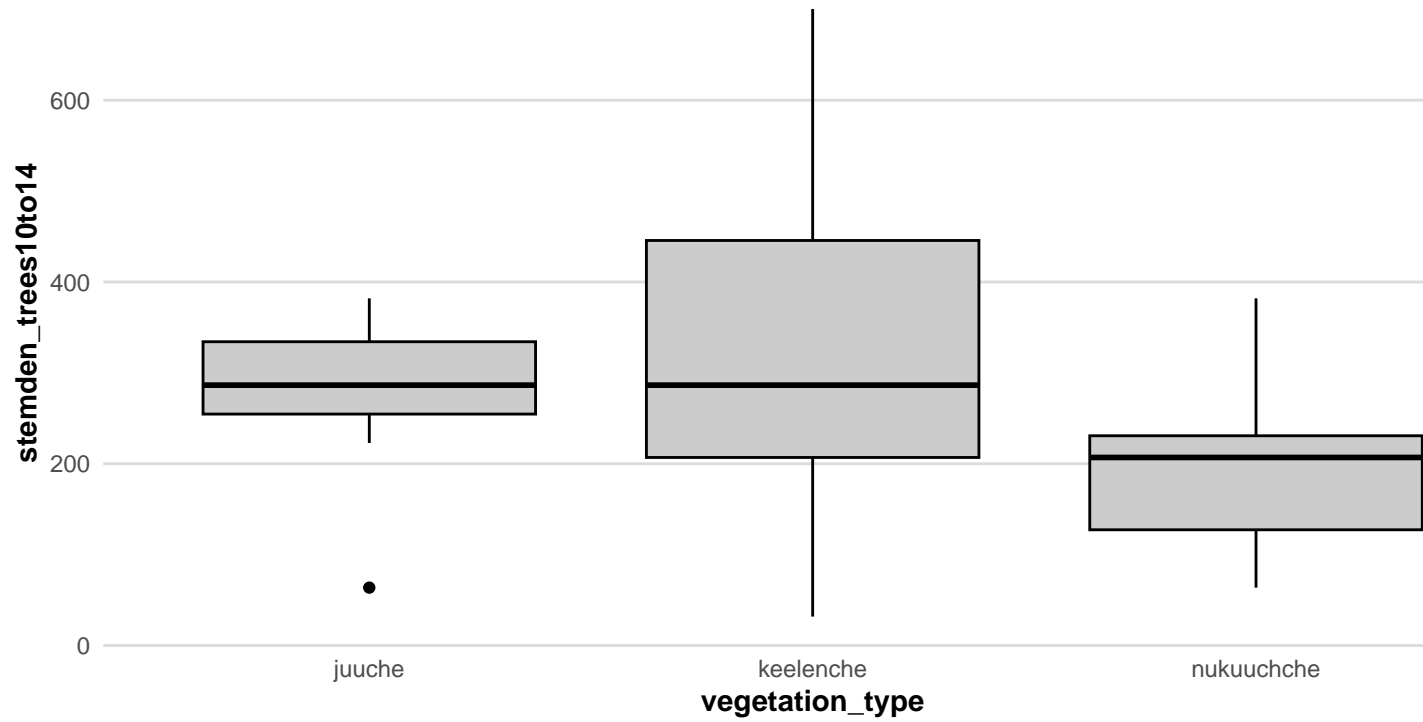
```
data_plots %>% ggplot() +  
  geom_boxplot(aes(x = vegetation_type, y = stemden_saplings),  
    col = "black", fill = "gray80") + plot_theme
```



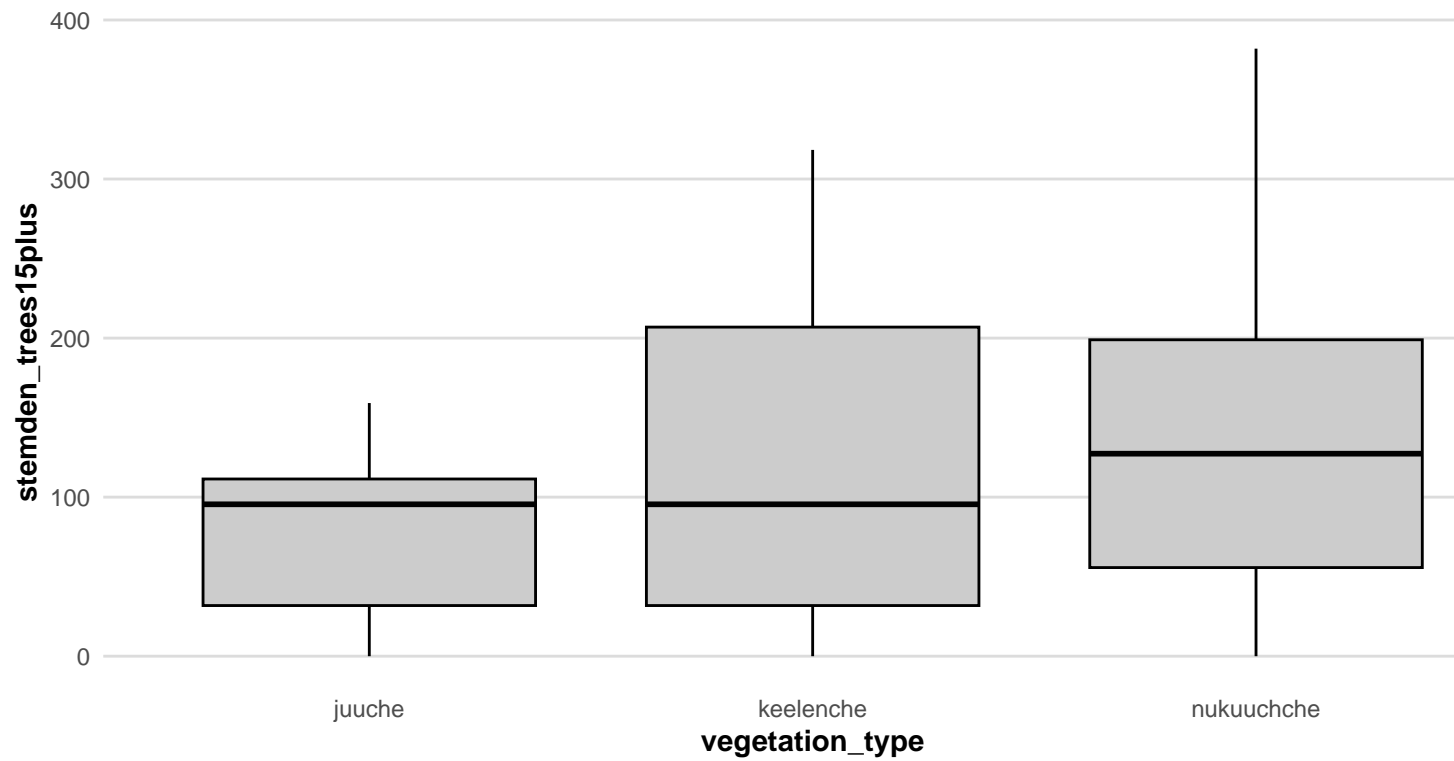
```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = vegetation_type, y = stemden_trees05to09),
    col = "black", fill = "gray80") + plot_theme
```



```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = vegetation_type, y = stemden_trees10to14),
    col = "black", fill = "gray80") + plot_theme
```

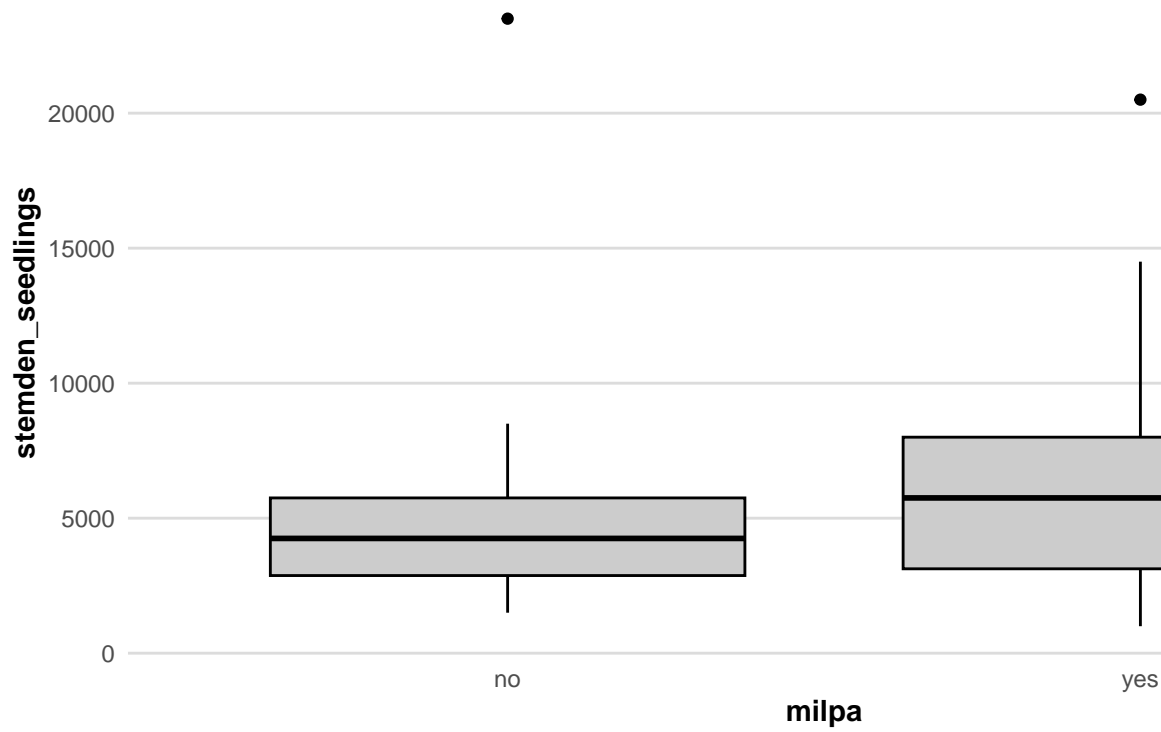


```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = vegetation_type, y = stemden_trees15plus),
    col = "black", fill = "gray80") + plot_theme
```

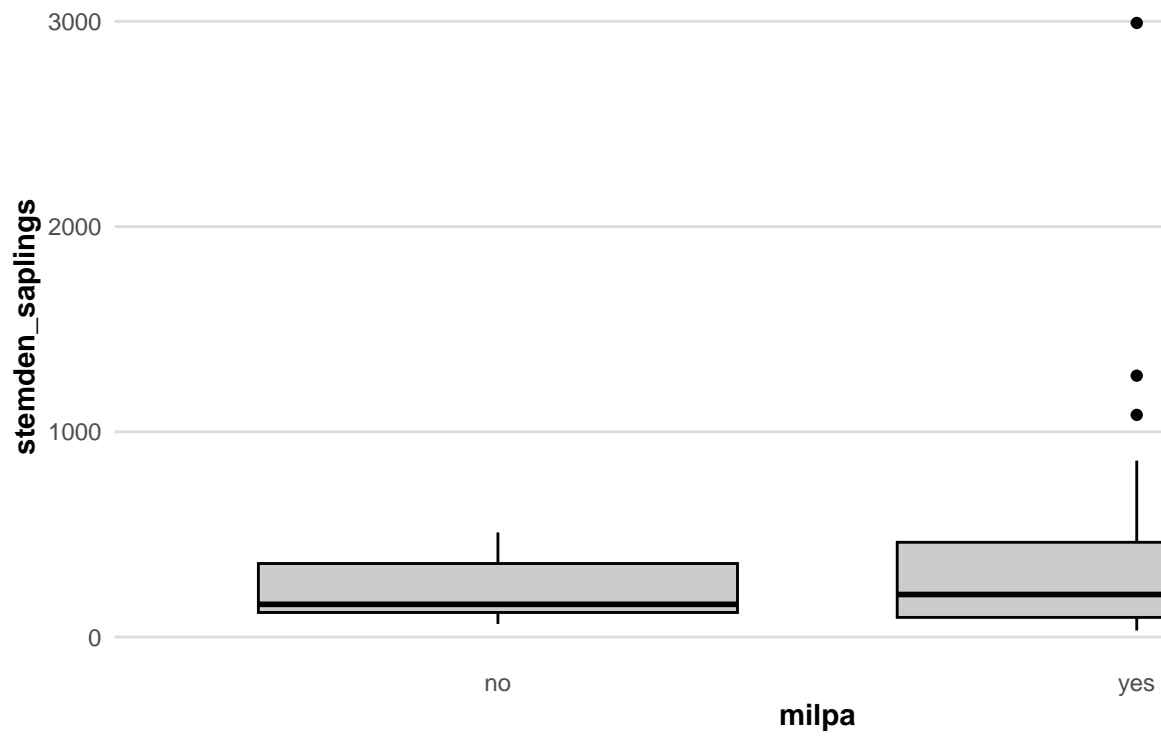


Box plots, milpa exposure

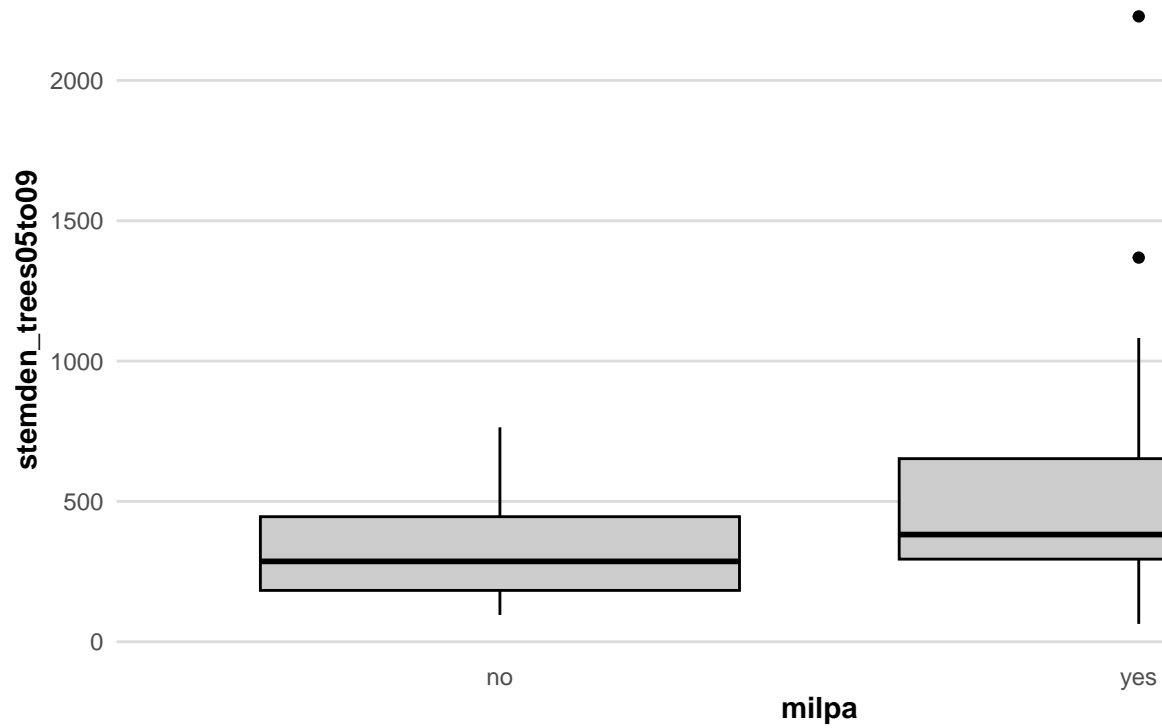
```
data_plots %>% ggplot() +  
  geom_boxplot(aes(x = milpa, y = stemden_seedlings),  
    col = "black", fill = "gray80") + plot_theme
```



```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = milpa, y = stemden_saplings),
    col = "black", fill = "gray80") + plot_theme
```

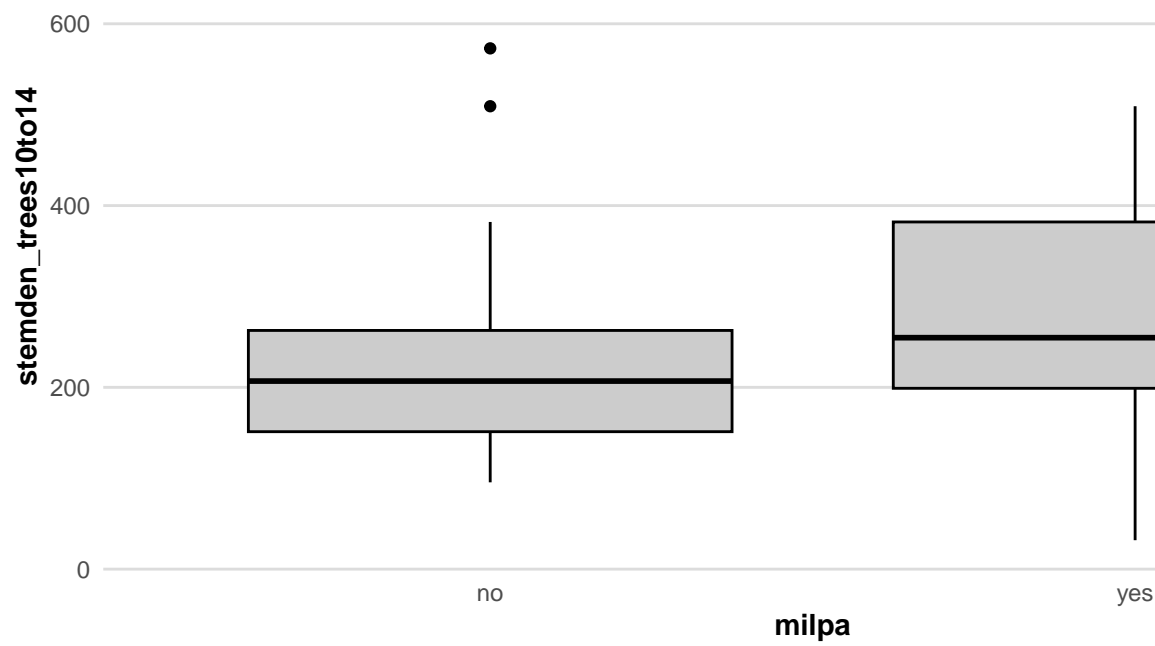


```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = milpa, y = stemden_trees05to09),
    col = "black", fill = "gray80") + plot_theme
```

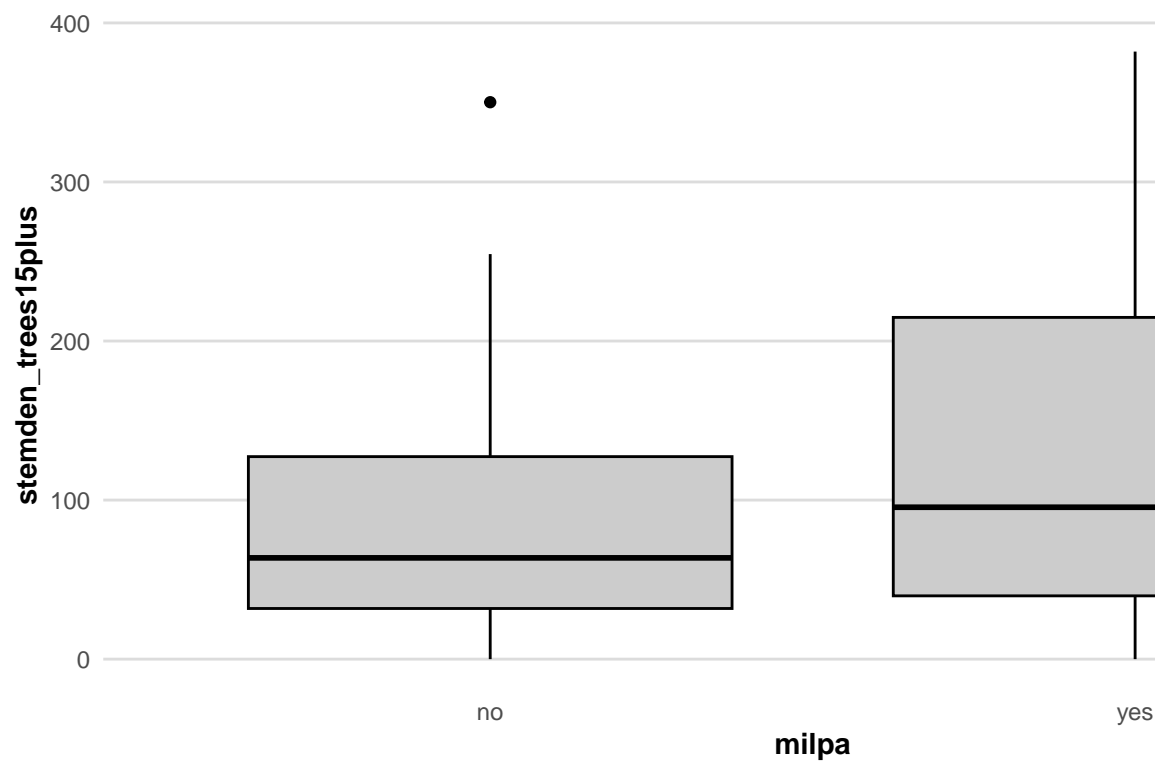


```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = milpa, y = stemden_trees10to14),
    col = "black", fill = "gray80") + plot_theme
```





```
data_plots %>% ggplot() +
  geom_boxplot(aes(x = milpa, y = stemden_trees15plus),
    col = "black", fill = "gray80") + plot_theme
```



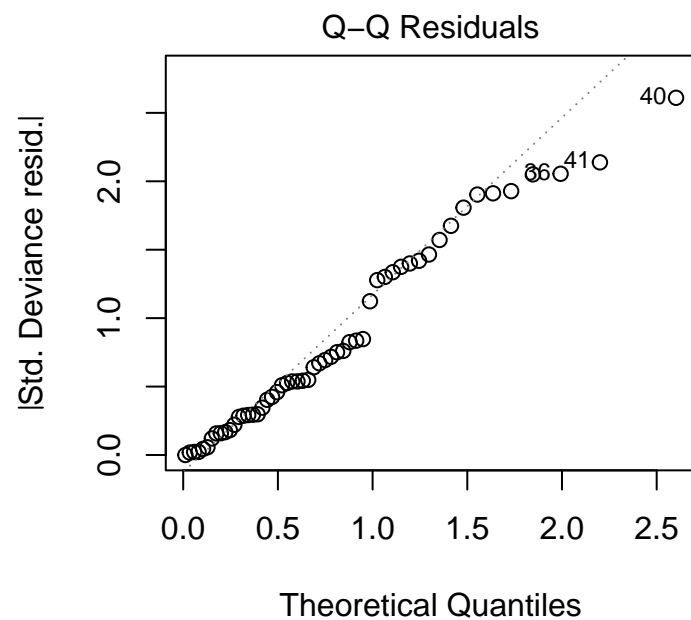
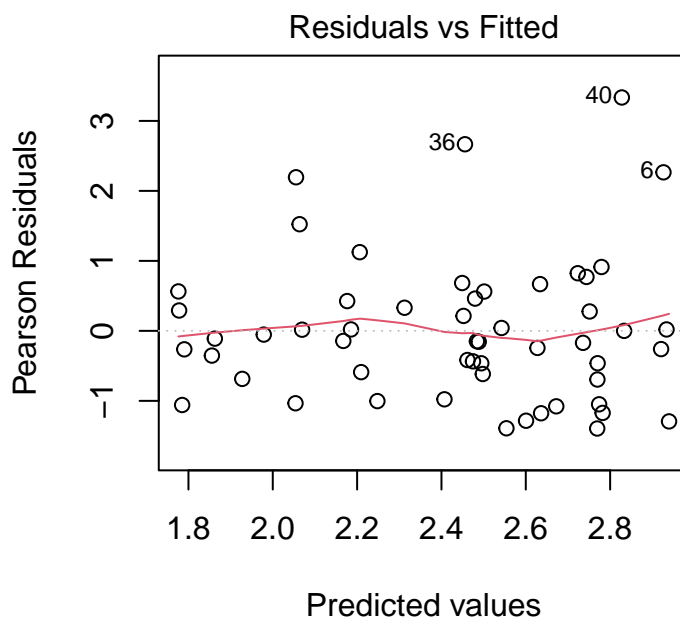
# Models

## Seedlings

Fit negative binomial model and check residual plots

```
mod_seedlings <- MASS::glm.nb(stemden_seedlings_count ~ harvested + vegetation_type + milpa +  
                             latitude + longitude + offset(log(plots_per_ha)), data = data_plots)
```

```
par(mfrow = c(1,2))  
plot(mod_seedlings, which = 1:2)
```



Summarize the model

```
summary(mod_seedlings)
```

```
##  
## Call:  
## MASS::glm.nb(formula = stemden_seedlings_count ~ harvested +  
##   vegetation_type + milpa + latitude + longitude + offset(log(plots_per_ha)),  
##   data = data_plots, init.theta = 4.431556184, link = log)  
##  
## Coefficients:  
##               Estimate Std. Error z value Pr(>|z|)  
## (Intercept)      6.40554    0.32882  19.480  <2e-16 ***  
## harvestedyes     -0.28202    0.15359  -1.836   0.0663 .  
## vegetation_typekeelenche -0.16222    0.24717  -0.656   0.5116  
## vegetation_typenukuuchche -0.66862    0.31346  -2.133   0.0329 *
```

```
## milpayes          -0.06720    0.22546   -0.298    0.7656
## latitude          0.04888    0.08333    0.587    0.5575
## longitude         -0.09125    0.09999   -0.913    0.3614
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(4.4316) family taken to be 1)
##
## Null deviance: 72.647  on 53  degrees of freedom
## Residual deviance: 53.460  on 47  degrees of freedom
## AIC: 359.05
##
## Number of Fisher Scoring iterations: 1
##
##
##          Theta:  4.43
##        Std. Err.:  1.14
##
## 2 x log-likelihood:  -343.049
```

## Construct analysis of deviance table

```
# Use type 2 SS since data are unbalanced
Anova(mod_seedlings, type = 2, test.statistic = "LR")
```

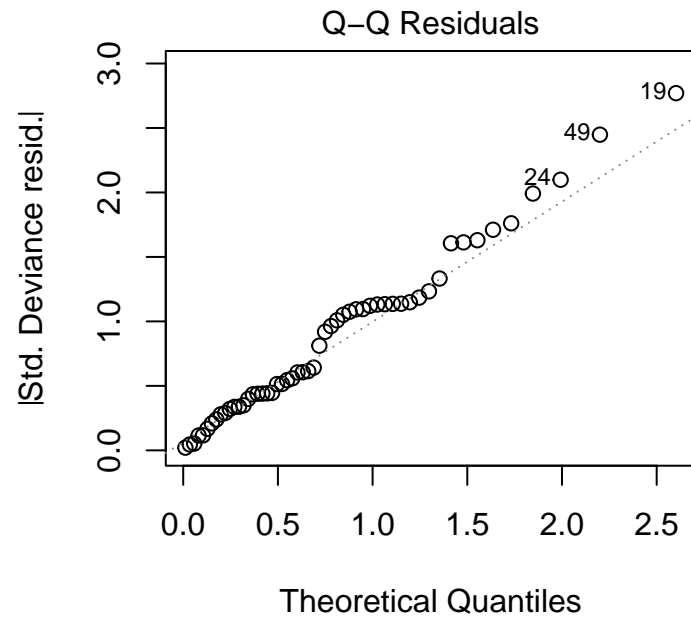
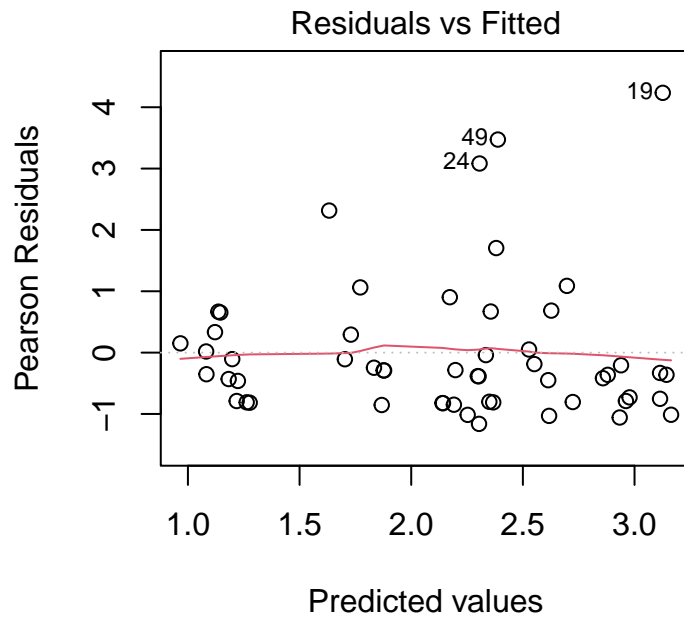
```
## Analysis of Deviance Table (Type II tests)
##
## Response: stemden_seedlings_count
##          LR Chisq Df Pr(>Chisq)
## harvested      3.3154  1  0.06863 .
## vegetation_type  5.9763  2  0.05038 .
## milpa          0.0958  1  0.75693
## latitude       0.3055  1  0.58047
## longitude      0.8437  1  0.35835
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## Saplings (0-4 cm DBH)

Fit negative binomial model and check residual plots

```
mod_saplings <- MASS::glm.nb(stemden_saplings_count ~ harvested + vegetation_type + milpa +
                             latitude + longitude + offset(log(plots_per_ha)), data = data_plots)
```

```
par(mfrow = c(1,2))
plot(mod_saplings, which = 1:2)
```



Summarize the model

```
summary(mod_saplings)
```

```
##
## Call:
## MASS::glm.nb(formula = stemden_saplings_count ~ harvested + vegetation_type +
##     milpa + latitude + longitude + offset(log(plots_per_ha)),
##     data = data_plots, init.theta = 1.998067315, link = log)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      7.19054    0.45806  15.698 < 2e-16 ***
## harvestedyes      -0.06765    0.21698  -0.312  0.7552
## vegetation_typekeelenche -0.57495    0.34255  -1.678  0.0933 .
## vegetation_typenukuuchche -2.16024    0.45084  -4.792 1.65e-06 ***
## milpayes         -0.64989    0.31479  -2.065  0.0390 *
## latitude          0.04118    0.11729   0.351  0.7255
## longitude         0.24265    0.15164   1.600  0.1095
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(1.9981) family taken to be 1)
##
## Null deviance: 86.822 on 53 degrees of freedom
## Residual deviance: 53.194 on 47 degrees of freedom
## AIC: 354.3
##
## Number of Fisher Scoring iterations: 1
##
##
```

```
##           Theta: 1.998
##         Std. Err.: 0.439
##
## 2 x log-likelihood: -338.301
```

## Construct analysis of deviance table

```
# Use type 2 SS since data are unbalanced
Anova(mod_saplings, type = 2, test.statistic = "LR")
```

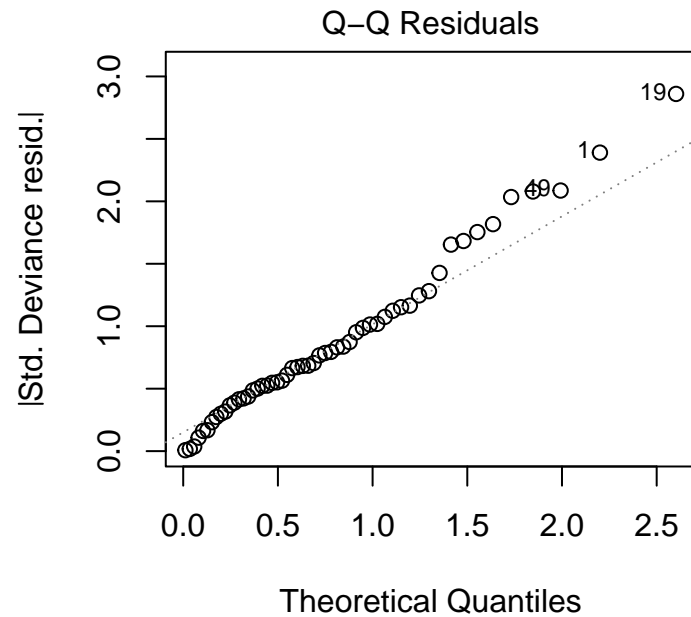
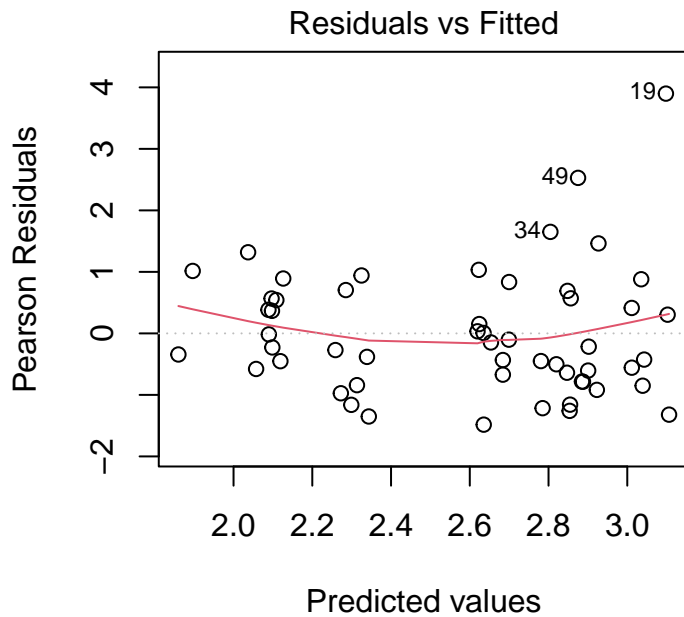
```
## Analysis of Deviance Table (Type II tests)
##
## Response: stemden_saplings_count
##           LR Chisq Df Pr(>Chisq)
## harvested      0.0919 1    0.76176
## vegetation_type 23.6667 2 7.258e-06 ***
## milpa          3.6939 1    0.05461 .
## latitude       0.1477 1    0.70071
## longitude      2.3977 1    0.12151
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## Trees (5-9 cm DBH)

Fit negative binomial model and check residual plots

```
mod_trees05to09 <- MASS::glm.nb(stemden_trees05to09_count ~ harvested + vegetation_type + milpa +
                                latitude + longitude + offset(log(plots_per_ha)), data = data_plots)
```

```
par(mfrow = c(1,2))
plot(mod_trees05to09, which = 1:2)
```



Summarize the model

```
summary(mod_trees05to09)
```

```
##
## Call:
## MASS::glm.nb(formula = stemden_trees05to09_count ~ harvested +
##   vegetation_type + milpa + latitude + longitude + offset(log(plots_per_ha)),
##   data = data_plots, init.theta = 3.821729251, link = log)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)      6.48187   0.34212  18.946  <2e-16 ***
## harvestedyes     -0.22461   0.15912  -1.412   0.1581
## vegetation_typekeelenche -0.10869   0.25708  -0.423   0.6724
## vegetation_typenukuuchche -0.80323   0.32864  -2.444   0.0145 *
## milpayes         0.04076   0.23328   0.175   0.8613
## latitude         0.03014   0.08622   0.350   0.7266
## longitude        0.06194   0.10753   0.576   0.5646
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(3.8217) family taken to be 1)
##
##   Null deviance: 75.917  on 53  degrees of freedom
## Residual deviance: 54.526  on 47  degrees of freedom
## AIC: 379.14
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  3.822
```

```
##          Std. Err.:  0.918
##
## 2 x log-likelihood: -363.145
```

## Construct analysis of deviance table

```
# Use type 2 SS since data are unbalanced
Anova(mod_trees05to09, type = 2, test.statistic = "LR")

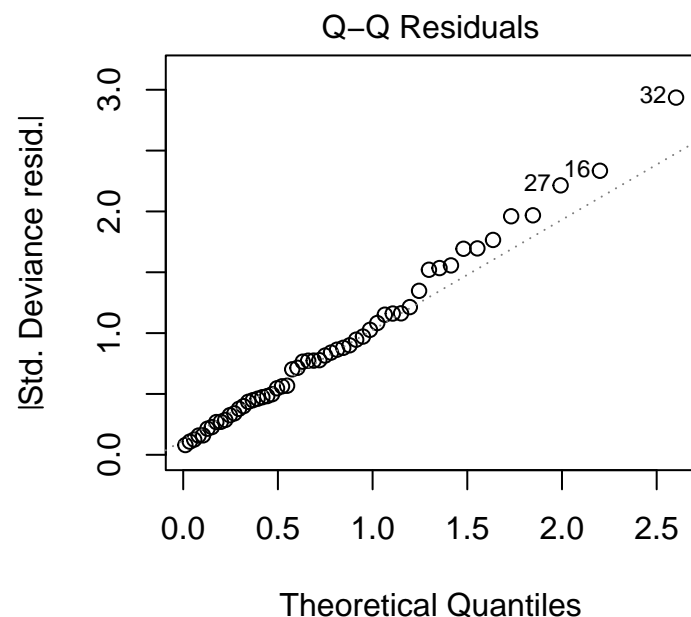
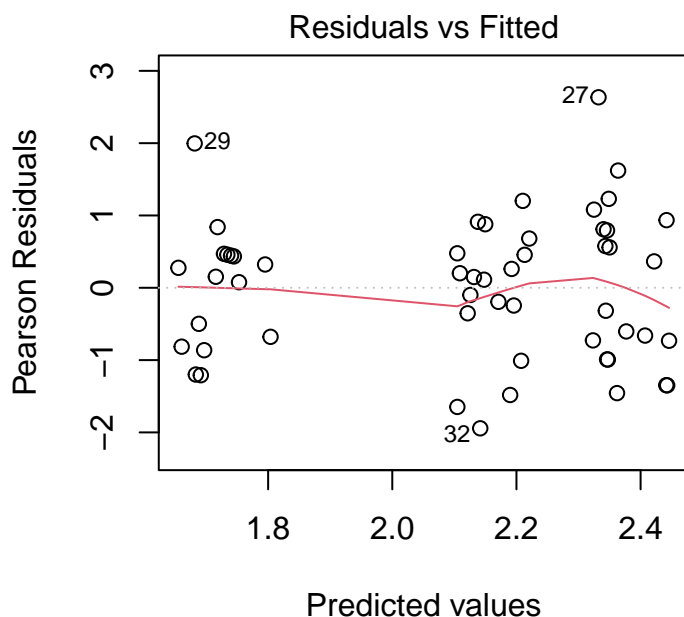
## Analysis of Deviance Table (Type II tests)
##
## Response: stemden_trees05to09_count
##          LR Chisq Df Pr(>Chisq)
## harvested      1.9290  1  0.16487
## vegetation_type  8.6431  2  0.01328 *
## milpa           0.0291  1  0.86458
## latitude        0.1400  1  0.70823
## longitude       0.3381  1  0.56090
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## Trees (10-14 cm DBH)

Fit negative binomial model and check residual plots

```
mod_trees10to14 <- MASS::glm.nb(stemden_trees10to14_count ~ harvested + vegetation_type + milpa +
                                latitude + longitude + offset(log(plots_per_ha)), data = data_plots)

par(mfrow = c(1,2))
plot(mod_trees10to14, which = 1:2)
```



## Summarize the model

```
summary(mod_trees10to14)
```

```
##
## Call:
## MASS::glm.nb(formula = stemden_trees10to14_count ~ harvested +
##   vegetation_type + milpa + latitude + longitude + offset(log(plots_per_ha)),
##   data = data_plots, init.theta = 11.26446796, link = log)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)      5.594083   0.272227  20.549   <2e-16 ***
## harvestedyes      0.006911   0.124847   0.055   0.9559
## vegetation_typekeelenche 0.098029   0.206609   0.474   0.6352
## vegetation_typenukuuchche -0.258169   0.258446  -0.999   0.3178
## milpayes         0.024663   0.184516   0.134   0.8937
## latitude        -0.023941   0.066723  -0.359   0.7197
## longitude       -0.147160   0.080787  -1.822   0.0685 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(11.2645) family taken to be 1)
##
##   Null deviance: 73.237  on 53  degrees of freedom
## Residual deviance: 55.910  on 47  degrees of freedom
## AIC: 309.13
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  11.26
##             Std. Err.:  5.19
##
## 2 x log-likelihood:  -293.128
```

## Construct analysis of deviance table

```
# Use type 2 SS since data are unbalanced
Anova(mod_trees10to14, type = 2, test.statistic = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: stemden_trees10to14_count
##               LR Chisq Df Pr(>Chisq)
## harvested      0.0031  1   0.95591
## vegetation_type 3.5769  2   0.16722
## milpa          0.0182  1   0.89260
## latitude       0.1277  1   0.72083
## longitude      3.2234  1   0.07259 .
```



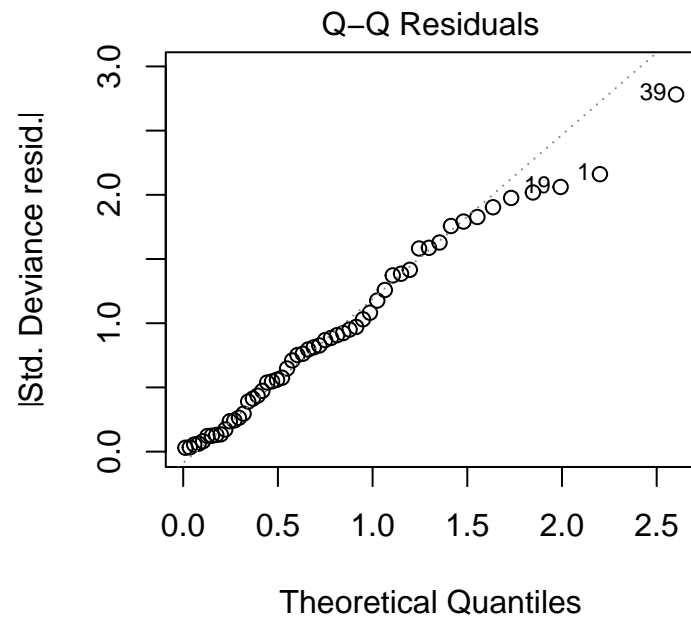
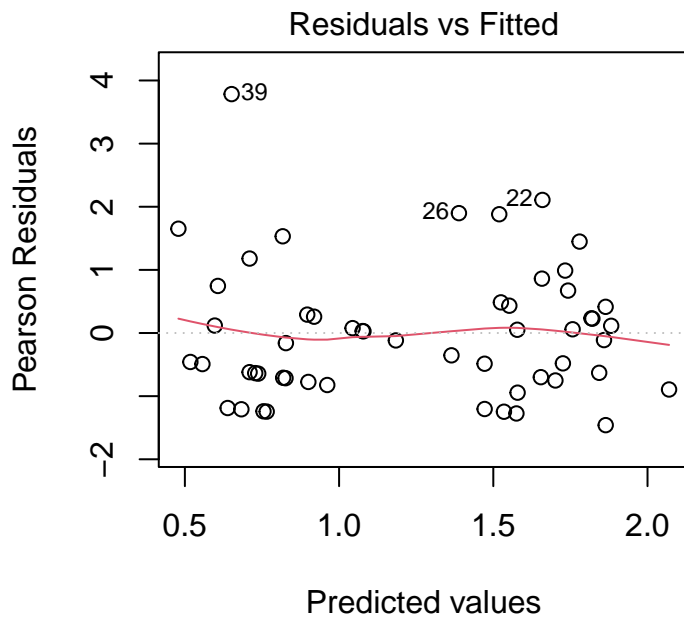
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## Trees (15+ cm DBH)

Fit negative binomial model and check residual plots

```
mod_trees15plus <- MASS::glm.nb(stemden_trees15plus_count ~ harvested + vegetation_type + milpa +
                                latitude + longitude + offset(log(plots_per_ha)), data = data_plots)
```

```
par(mfrow = c(1,2))
plot(mod_trees15plus, which = 1:2)
```



Summarize the model

```
summary(mod_trees15plus)
```

```
##
## Call:
## MASS::glm.nb(formula = stemden_trees15plus_count ~ harvested +
##   vegetation_type + milpa + latitude + longitude + offset(log(plots_per_ha)),
##   data = data_plots, init.theta = 5.543400709, link = log)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)      3.3386     0.4550   7.337 2.18e-13 ***
## harvestedyes       0.1417     0.1847   0.767 0.442860
## vegetation_typekeelenche 0.2565     0.3453   0.743 0.457603
```

```
## vegetation_typedukuuchche 1.1883 0.4014 2.960 0.003076 **
## milpayes 1.0442 0.3042 3.432 0.000598 ***
## latitude -0.1336 0.1033 -1.294 0.195764
## longitude -0.3950 0.1144 -3.454 0.000553 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(5.5434) family taken to be 1)
##
## Null deviance: 84.723 on 53 degrees of freedom
## Residual deviance: 59.409 on 47 degrees of freedom
## AIC: 254.68
##
## Number of Fisher Scoring iterations: 1
##
##
## Theta: 5.54
## Std. Err.: 2.82
##
## 2 x log-likelihood: -238.682
```

## Construct analysis of deviance table

```
# Use type 2 SS since data are unbalanced
Anova(mod_trees15plus, type = 2, test.statistic = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: stemden_trees15plus_count
## LR Chisq Df Pr(>Chisq)
## harvested 0.5871 1 0.4435505
## vegetation_type 14.0366 2 0.0008953 ***
## milpa 13.5045 1 0.0002380 ***
## latitude 1.6021 1 0.2056015
## longitude 11.8395 1 0.0005799 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```