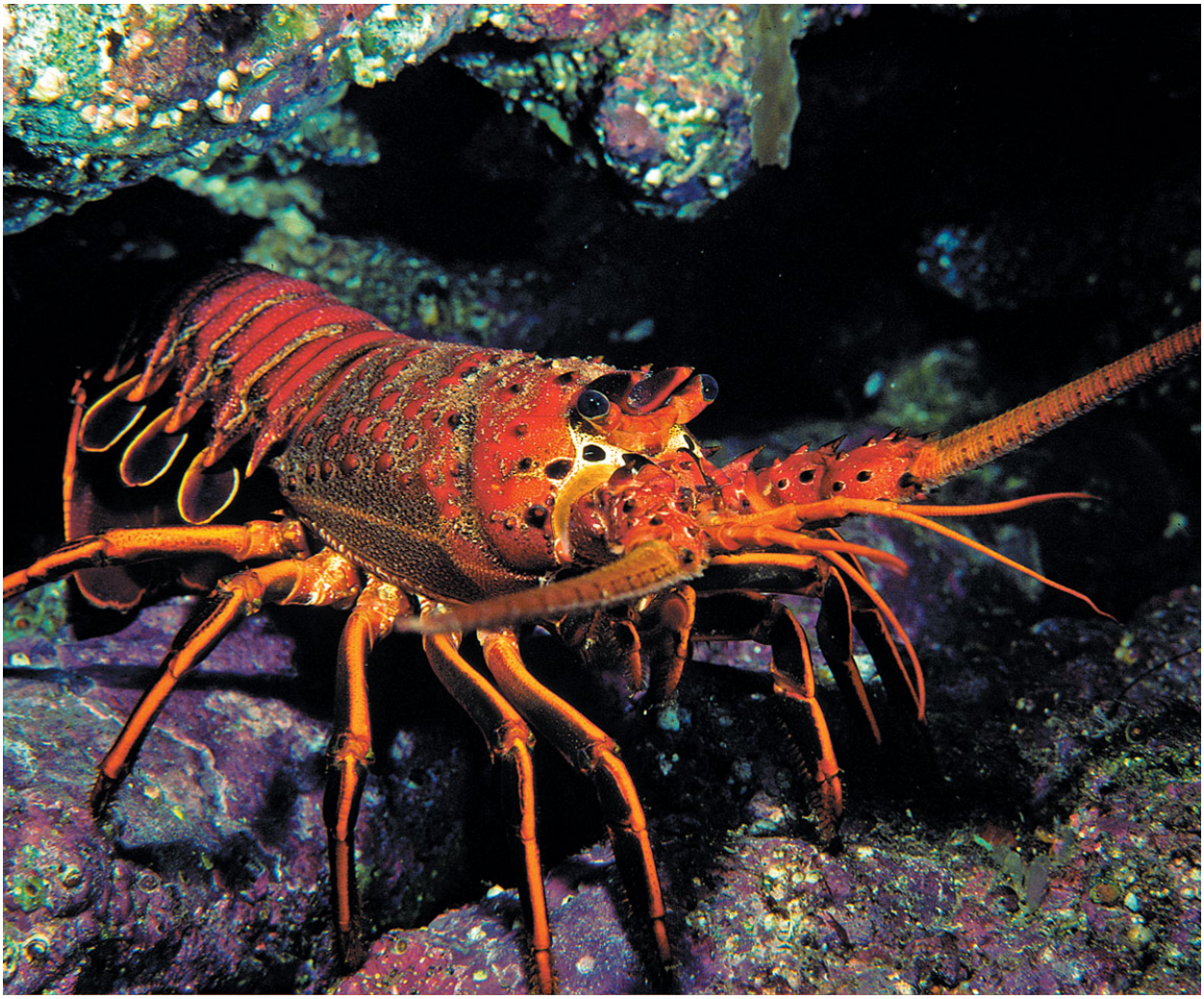
Assignment 1

California Spiny Lobster (*Panulirus Interruptus*): Assessing the Impact of Marine Protected Areas (MPAs) at 5 Reef Sites in Santa Barbara County

EDS 241 / ESM 244 (**Due: 1/17**)

1/8/26



Assignment Instructions:

- Working with partners to troubleshoot code and concepts is encouraged! If you work with a partner, please list their name next to yours at the top of your assignment so Annie and I can easily see who

collaborated.

- All written responses must be written independently (**in your own words**).
- Please follow the question prompts carefully and include only the information each question asks in your submitted responses.
- Submit both your knitted document and the associated `RMarkdown` or `Quarto` file.
- Your knitted presentation should meet the quality you'd submit to research colleagues or feel confident sharing publicly. Refer to the rubric for details about presentation standards.

Assignment submission (YOUR NAME): _____Ava Robillard_____

```
library(tidyverse)
library(here)
library(janitor)
library(estimatr)
library(performance)
library(jtools)
library(gt)
library(gtsummary)
library(interactions)
library(ggridges)
library(ggbeeswarm)
library(gghighlight)
```

DATA SOURCE:

Reed D. 2019. SBC LTER: Reef: Abundance, size and fishing effort for California Spiny Lobster (Panulirus interruptus), ongoing since 2012. Environmental Data Initiative. Data accessed 11/17/2019.

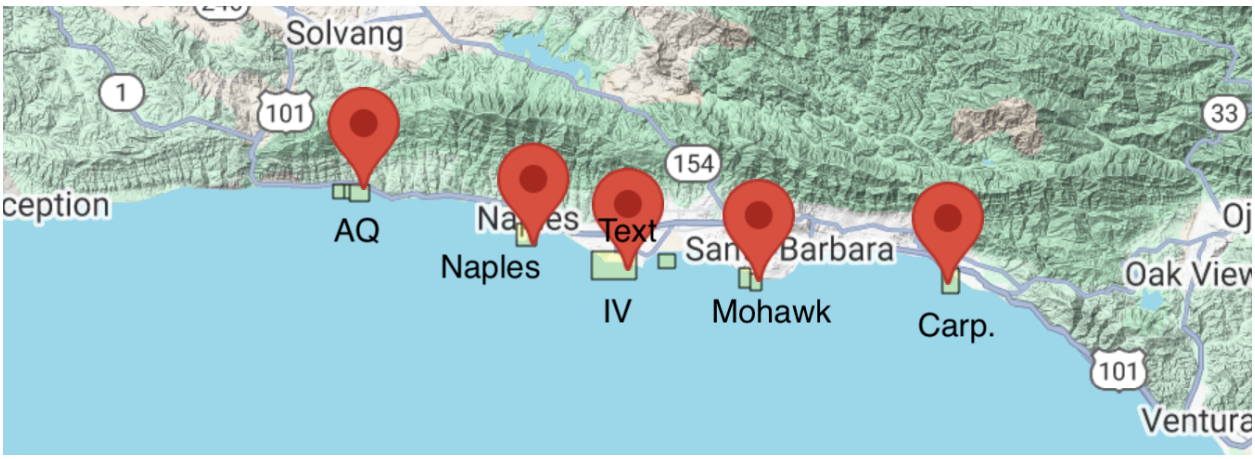
Introduction

You're about to dive into some deep data collected from five reef sites in Santa Barbara County, all about the abundance of California spiny lobsters! Data was gathered by divers annually from 2012 to 2018 across Naples, Mohawk, Isla Vista, Carpinteria, and Arroyo Quemado reefs.

Why lobsters? Well, this sample provides an opportunity to evaluate the impact of Marine Protected Areas (MPAs) established on January 1, 2012 (Reed, 2019). Of these five reefs, Naples, and Isla Vista are MPAs, while the other three are not protected (non-MPAs). Comparing lobster health between these protected and non-protected areas gives us the chance to study how commercial and recreational fishing might impact these ecosystems.

We will consider the MPA sites the `treatment` group and use regression methods to explore whether protecting these reefs really makes a difference compared to non-MPA sites (our control group). In this assignment, we'll think deeply about which causal inference assumptions hold up under the research design and identify where they fall short.

Let's break it down step by step and see what the data reveals!



Step 1: Anticipating potential sources of selection bias **a.** Do the control sites (Arroyo Quemado, Carpenteria, and Mohawk) provide a strong counterfactual for our treatment sites (Naples, Isla Vista)? Write a paragraph making a case for why this comparison is ceteris paribus or whether selection bias is likely (be specific!).

The control sites likely provide the best available counterfactual for the treatment sites for our purpose, which is understanding how commercial and recreational fishing impacts ecosystems based on Lobster health. These sites are close enough in proximity to have similar weather experiences, ocean temperatures, overall climate, and species. However, there are still some differences to take into account that could introduce selection bias, such as level of human disturbance and the location of especially diverse ecosystems in need of protection that might have been factors in determining where the MPAs were placed in 2012, making it not necessarily a ceteris paribus comparison.

Step 2: Read & wrangle data **a.** Read in the raw data from the "data" folder named `spiny_abundance_sb_18.csv`. Name the data.frame `rawdata`

b. Use the function `clean_names()` from the `janitor` package

```
# HINT: check for coding of missing values (`na = "-99999"`)

# Read in data
rawdata <- read_csv(here("data", "spiny_abundance_sb_18.csv")) %>%
    # Clean column names
    clean_names() %>%
    # Replace -99999 with NA
    mutate(size_mm = na_if(size_mm, -99999))
```

c. Create a new `df` named `tidyata`. Using the variable `site` (reef location) create a new variable `reef` as a `factor` and add the following labels in the order listed (i.e., re-order the `levels`):

```
"Arroyo Quemado", "Carpenteria", "Mohawk", "Isla Vista",  "Naples"
```

```
# Create ordered reef column based on site
tidydata <- rawdata %>%
    mutate(reef = factor(site,
                         levels = c("AQUE", "CARP", "MOHK", "IVEE", "NAPL"),
                         labels = c("Arroyo Quemado", "Carpenteria", "Mohawk", "Isla Vista",  "Naples")))
```

Create new `df` named `spiny_counts`

d. Create a new variable `counts` to allow for an analysis of lobster counts where the unit-level of observation is the total number of observed lobsters per `site`, `year` and `transect`.

- Create a variable `mean_size` from the variable `size_mm`
- NOTE: The variable `counts` should have values which are integers (whole numbers).
- Make sure to account for missing cases (`na`)!

e. Create a new variable `mpa` with levels `MPA` and `non_MPA`. For our regression analysis create a numerical variable `treat` where MPA sites are coded `1` and non_MPA sites are coded `0`

```
#HINT(d): Use `group_by()` & `summarize()` to provide the total number of lobsters observed at each sit

#HINT(e): Use `case_when()` to create the 3 new variable columns

# Create data frame of lobster counts by site, year, and transect
spiny_counts <- tidydata %>%
    group_by(reef, year, transect) %>%
    # Create columns for mean size and count
    summarize(counts = sum(count, na.rm = TRUE),
              mean_size = mean(size_mm, na.rm = TRUE)) %>%
    # Create column for MPA status
    mutate(mpa = case_when(
        reef == "Isla Vista" ~ "MPA",
        reef == "Naples" ~ "MPA",
        reef == "Arroyo Quemado" ~ "non_MPA",
        reef == "Carpenteria" ~ "non_MPA",
        reef == "Mohawk" ~ "non_MPA"
    )) %>%
    # Encode MPA as treatment (1)
    mutate(treat = case_when(
        mpa == "MPA" ~ 1,
        mpa == "non_MPA" ~ 0
    ))
```

NOTE: This step is crucial to the analysis. Check with a friend or come to TA/instructor office hours to make sure the counts are coded correctly!

Step 3: Explore & visualize data **a.** Take a look at the data! Get familiar with the data in each `df` format (`tidydata`, `spiny_counts`)

b. We will focus on the variables `count`, `year`, `site`, and `treat`(`mpa`) to model lobster abundance. Create the following 4 plots using a different method each time from the 6 options provided. Add a layer (`geom`) to each of the plots including informative descriptive statistics (you choose; e.g., mean, median, SD, quartiles, range). Make sure each plot dimension is clearly labeled (e.g., axes, groups).

- Density plot
- Ridge plot
- Jitter plot
- Violin plot
- Histogram
- Beeswarm

Create plots displaying the distribution of lobster **counts**:

1) grouped by reef site

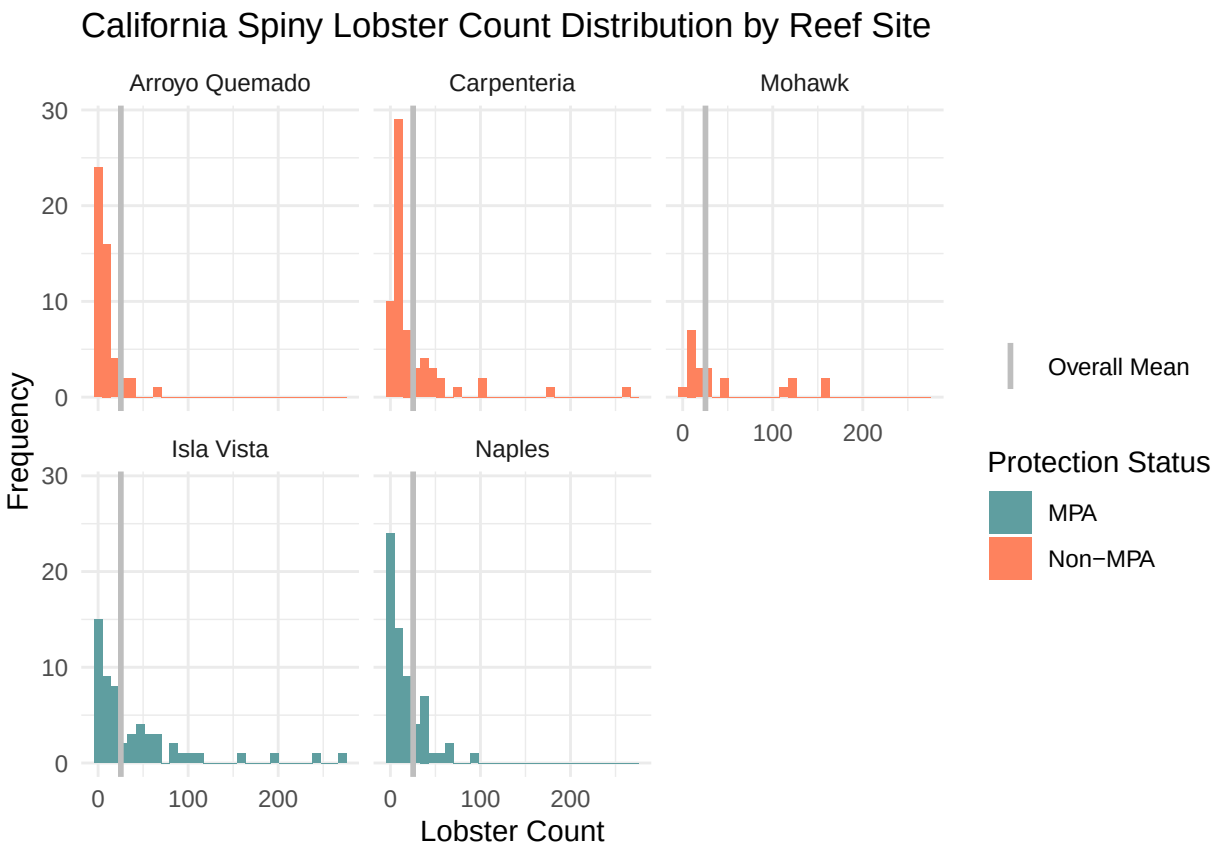
2) grouped by MPA status
3) grouped by year

Create a plot of lobster **size** :

4) You choose the grouping variable(s)!

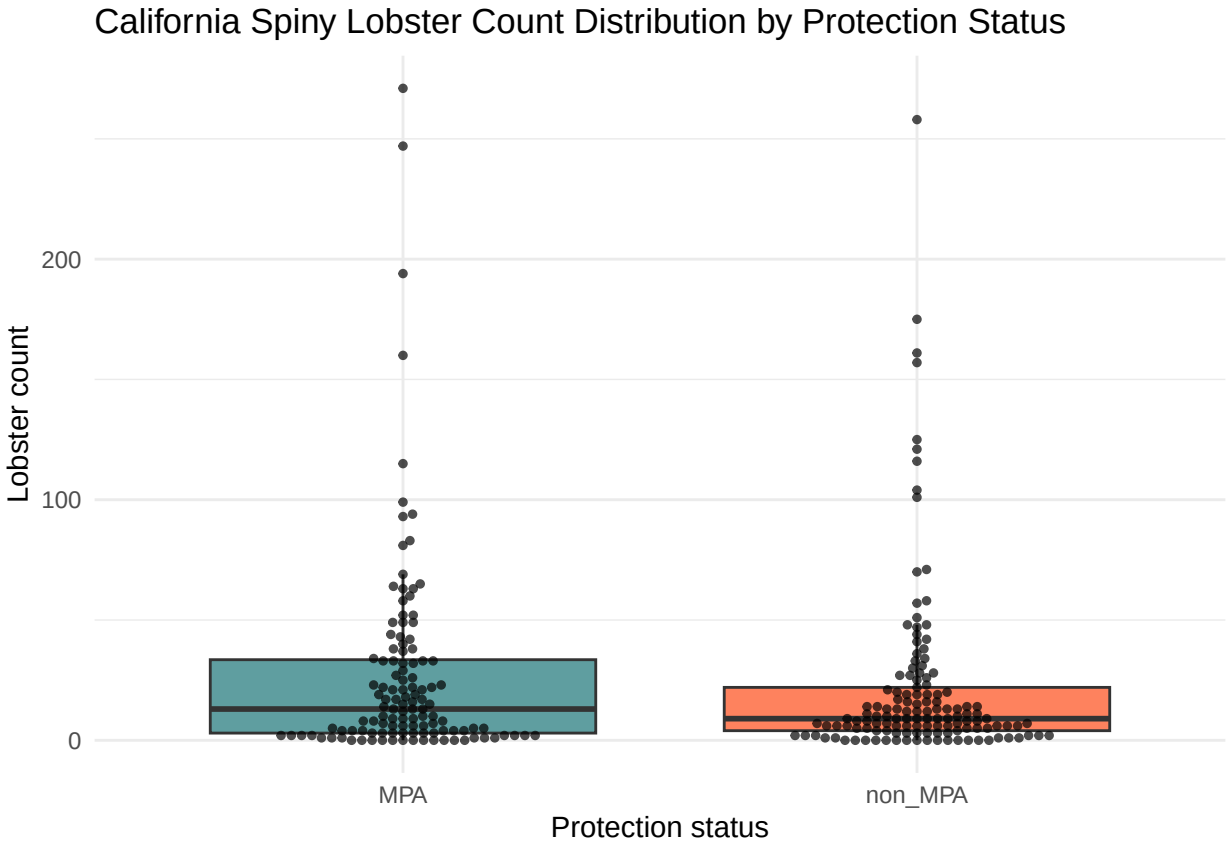
```
# Plot 1:

# Lobster count by reef site
spiny_counts %>%
ggplot(aes(x = counts, fill = mpa)) +
    geom_histogram() +
    facet_wrap(~reef) +
    scale_fill_manual(values = c("MPA" = "#5F9EA0", "non_MPA" = "#fe825e"),
        labels = c("MPA", "Non-MPA"),
        name = "Protection Status"
    ) +
    scale_color_manual(values = c("Overall Mean" = "grey"),
        name = ""
    ) +
    geom_vline(aes(xintercept = mean(spiny_counts$counts, na.rm = TRUE),
               color = "Overall Mean"),
               linewidth = 1,
               linetype = "solid") +
    labs(x = "Lobster Count",
         y = "Frequency",
         title = "California Spiny Lobster Count Distribution by Reef Site") +
    theme_minimal()
```



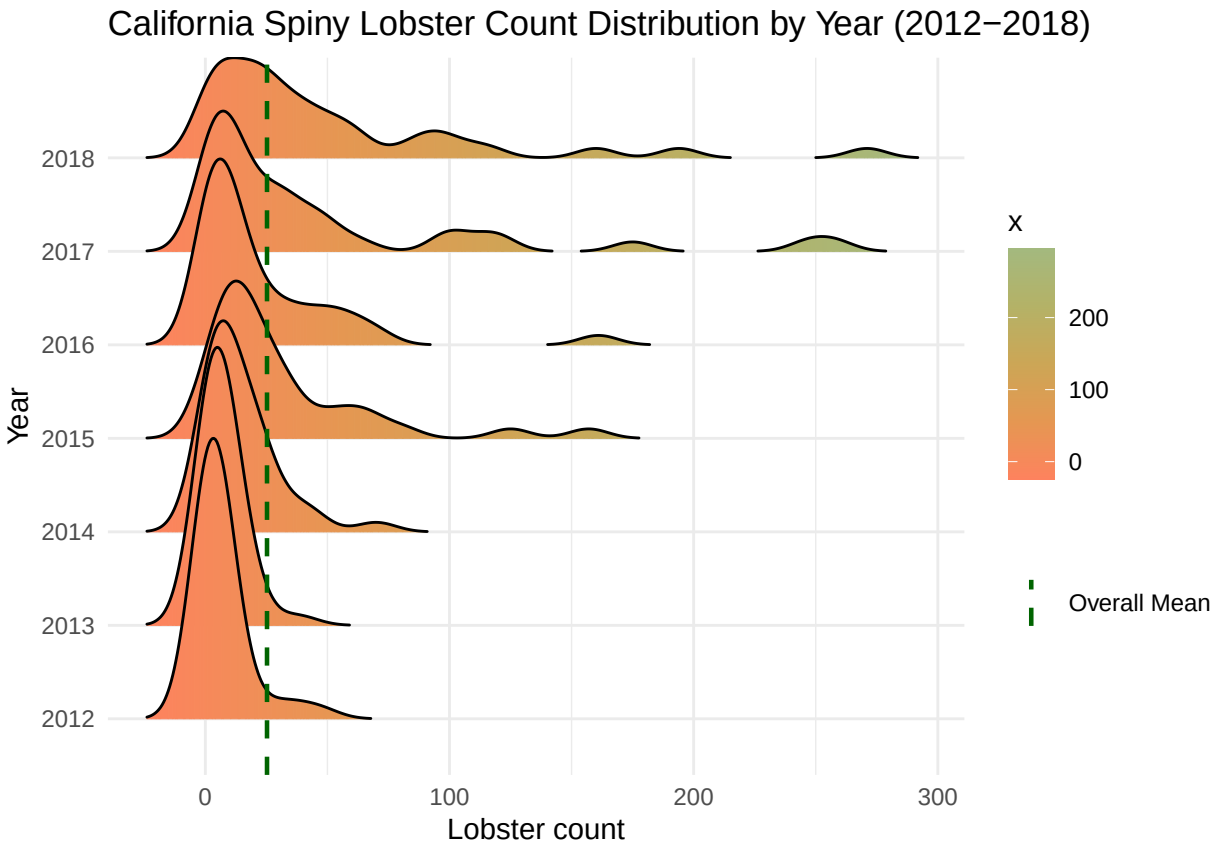

```
# Plot 2

# Lobster count by MPA treatment status
spiny_counts %>%
ggplot(aes(x = counts, y = mpa, fill = mpa)) +
    geom_boxplot(outlier.shape = NA) +
    geom_beeswarm(size = 1, alpha = 0.7) +
    scale_fill_manual(values = c("MPA" = "#5F9EA0", "non_MPA" = "#fe825e")) +
    labs(x = "Lobster count",
         y = "Protection status",
         title = "California Spiny Lobster Count Distribution by Protection Status") +
    coord_flip() +
    theme_minimal() +
    theme(legend.position = "none") # remove legend
```



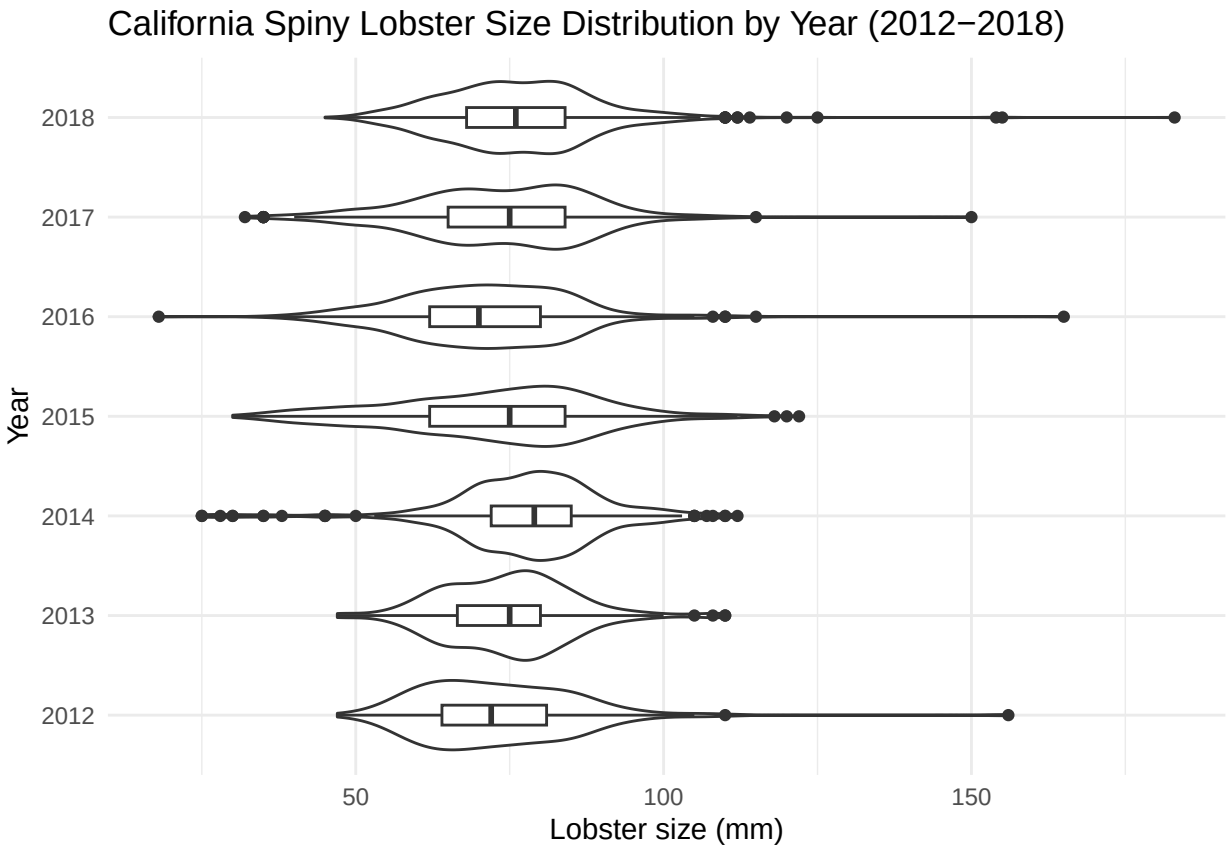

```
# Plot 3

# Lobster count by year
spiny_counts %>%
ggplot(aes(x = counts, y = as.factor(year), fill = after_stat(x))) +
    ggridges::geom_density_ridges_gradient(rel_min_height = 0.001, scale = 3) +
    scale_fill_gradientn(colors = c("#fe825e", "#e59752","#cca656","#b4b267","#a3b97f")) +
    geom_vline(aes(xintercept = mean(spiny_counts$counts, na.rm = TRUE),
               color = "Overall Mean"),
               linewidth = 0.8,
               linetype = "dashed") +
    scale_color_manual(values = c("Overall Mean" = "darkgreen"),
        name = ""
    ) +
    labs(x = "Lobster count",
         y = "Year",
         title = "California Spiny Lobster Count Distribution by Year (2012-2018)") +
    theme_minimal()
```




```
# Plot 4

# Lobster size by year
tidydata %>%
    ggplot(aes(x = as.factor(year), y = size_mm)) +
    geom_violin() +
    geom_boxplot(width = 0.2) +
    labs(y = "Lobster size (mm)",
         x = "Year",
         title = "California Spiny Lobster Size Distribution by Year (2012-2018)") +
    coord_flip() +
    theme_minimal()
```



c. Compare means of the outcome by treatment group. Using the `tbl_summary()` function from the package `gt_summary`

```
# USE: gt_summary::tbl_summary()

# Create table of mean outcomes
mean_count <- spiny_counts %>%
  gtsummary::tbl_summary(
    # Group columns by treatment
    by = treat,
    # Include count and size outcome variable
    include = c(counts, mean_size),
    # Add variable labels
    label = list(
        counts ~ "Lobster Count",
        mean_size ~ "Mean Size (mm)"
    ),
    # Display the mean and standard deviation
    statistic = list(all_continuous() ~ "{mean} ({sd})")
  ) %>%
  # Add p value
    add_p() %>%

# Turn into gt table to customize
  as_gt() %>%
  # Add a Title and Subtitle
  tab_header(
```

California Spiny Lobster Health Outcomes
Using treatment and control reef sites

Variable	Treatment Groups		P-Value[2]
	non-MPA[1]	MPA[1]	
Lobster Count	23 (39)	28 (44)	0.3
Mean Size (mm)	73 (7)	76 (7)	<0.001
Unknown	15	12	

[1]Mean (SD)
[2]Wilcoxon rank sum test
Note: Data from SBC LTER: Reef: Abundance, size and fishing effort for California Spiny Lobster (Panulirus interruptus), ongoing since 2012. Environmental Data Initiative.

```
    title = "California Spiny Lobster Health Outcomes",
    subtitle = "Using treatment and control reef sites"
  ) %>%
  # Add a Spanner to group the data columns
  tab_spanner(
    label = "Treatment Groups",
    columns = c(stat_1, stat_2)
  ) %>%
  # Change column labels
  cols_label(
    label = "Variable",
    stat_1 = "non-MPA",
    stat_2 = "MPA",
    p.value = "P-Value"
  ) %>%
  # Add a source note at the bottom
  tab_source_note(
    source_note = "Note: Data from SBC LTER: Reef: Abundance, size and fishing effort for California Sp
  )
mean_count
```

Step 4: OLS regression- building intuition **a.** Start with a simple OLS estimator of lobster counts regressed on treatment. Use the function `summ()` from the `jtools` package to print the OLS output

b. Interpret the intercept & predictor coefficients *in your own words*. Use full sentences and write your interpretation of the regression results to be as clear as possible to a non-academic audience.

```
# NOTE: We will not evaluate/interpret model fit in this assignment (e.g., R-square)

m1_ols <- lm(
  counts ~ treat,
  data = spiny_counts
)

summ(m1_ols, model.fit = FALSE)
```

Observations	252
Dependent variable	counts
Type	OLS linear regression

	Est.	S.E.	t val.	p
(Intercept)	22.73	3.57	6.36	0.00
treat	5.36	5.20	1.03	0.30

Standard errors: OLS

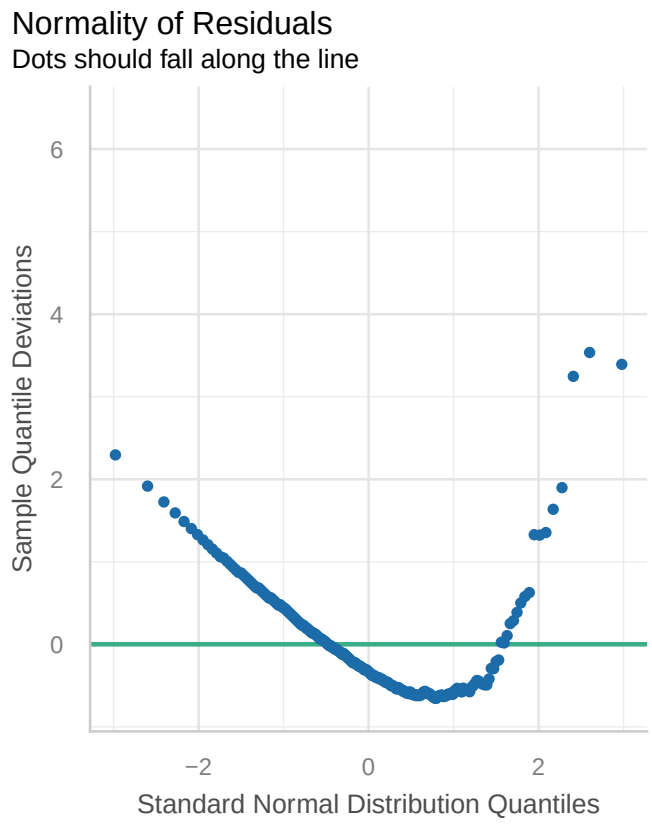
Intercept (22.73): When in an un-protected area (non-MPA), we expect the total number of observed Lobsters per unit (site, year, and transect combination) to be ~22 Lobsters on average.

Treat (5.36): Being within an MPA increases the expected total number of observed Lobsters per unit by ~5 Lobsters.

c. Check the model assumptions using the `check_model` function from the `performance` package

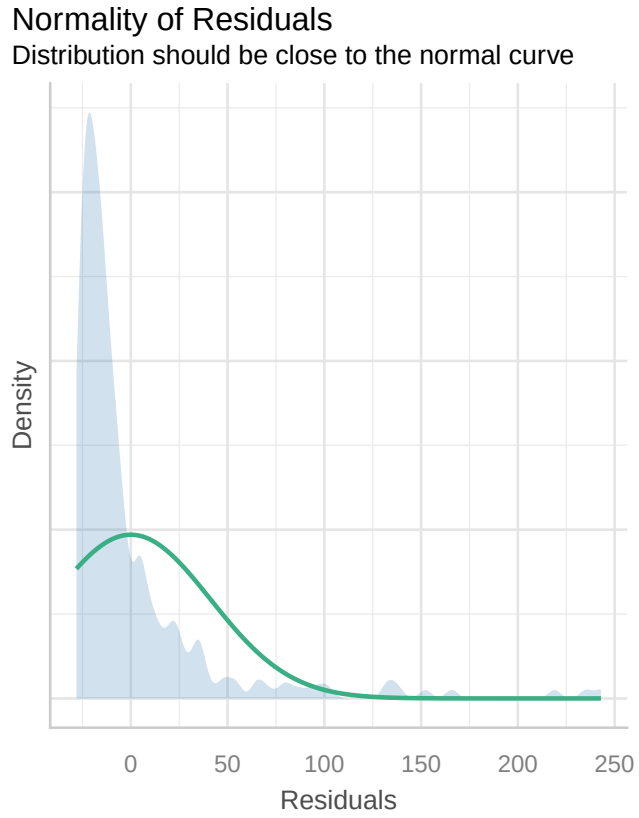
d. Explain the results of the 4 diagnostic plots. Why are we getting this result?

```
check_model(m1_ols,  check = "qq")
```



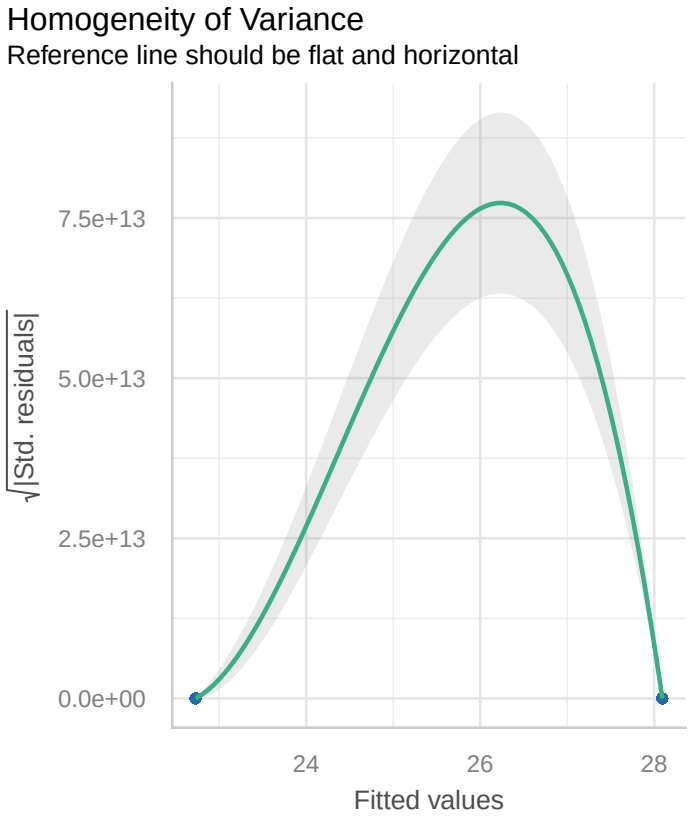
Based upon the QQ plot, the assumptions of an OLS regression are likely not met, as the residuals should be normally distributed. There is significant deviation from the green line, especially at the tails- indicating that the model does not predict the outcomes for data much beyond the most common values.

```
check_model(m1_ols, check = "normality")
```



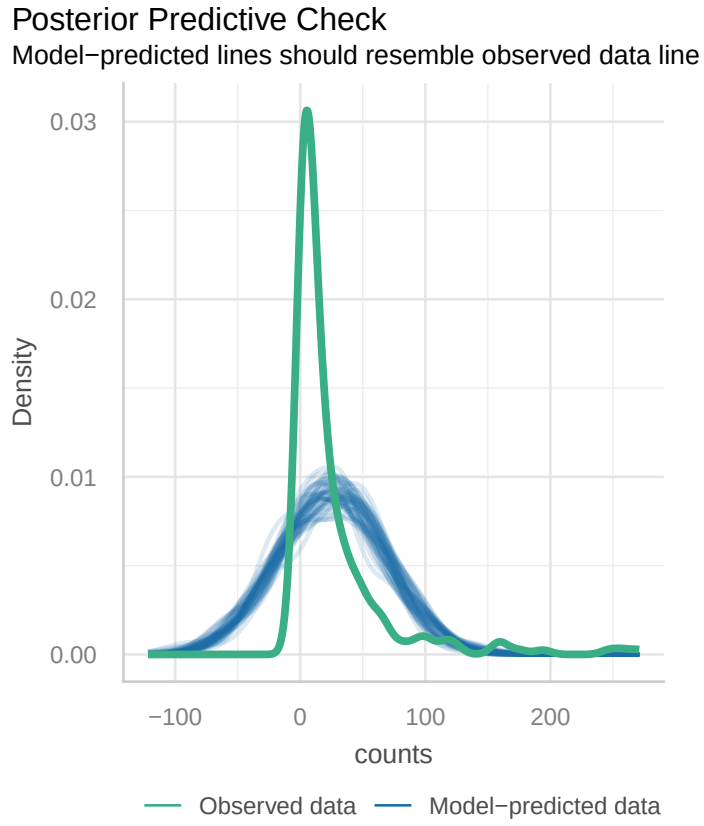
This check of the normality of residuals confirms that they do not follow a normal distribution as indicated by the green line with a sharp concentration to the left of the center. This contributes to the decision that we might not want to use a simple OLS estimator for our data.

```
check_model(m1_ols, check = "homogeneity")
```



The homogeneity of variance check shows that the variance of residuals across different values within our data is not constant, leading to a curved line instead of flat and horizontal.

```
check_model(m1_ols, check = "pp_check")
```



The posterior predictive check shows quite large differences between our observed data and the model-predicted data and does not seem to capture the distribution of the data well, so there are likely violations to the linear model assumptions.

Overall, our data does not seem to follow a normal distribution and likely needs a more complex regression model to capture the structure of our outcome of interest.

Step 5: Fitting GLMs **a.** Estimate a Poisson regression model using the `glm()` function

```
#HINT1: Incidence Ratio Rate (IRR): Exponentiation of beta returns coefficient which is interpreted as

#HINT2: For the second glm() argument `family` use the following specification option `family = poisson

# Fit Poisson model to data
m2_pois <- glm(counts ~ treat,
               family = poisson(link = "log"),
               data = spiny_counts)

# Check model output
summ(m2_pois, model.fit = FALSE)
```

b. Interpret the predictor coefficient in your own words. Use full sentences and write your interpretation of the results to be as clear as possible to a non-academic audience.

Without any protection status (non-MPA), the model estimates ~22 Lobsters per observational unit.

Observations	252
Dependent variable	counts
Type	Generalized linear model
Family	poisson
Link	log

	Est.	S.E.	z val.	p
(Intercept)	3.12	0.02	171.74	0.00
treat	0.21	0.03	8.44	0.00

Standard errors: MLE

Reef sites with MPA protection show an ~23.4% increase on average in the expected Lobster count per observational unit.

c. Explain the statistical concept of dispersion and overdispersion in the context of this model.

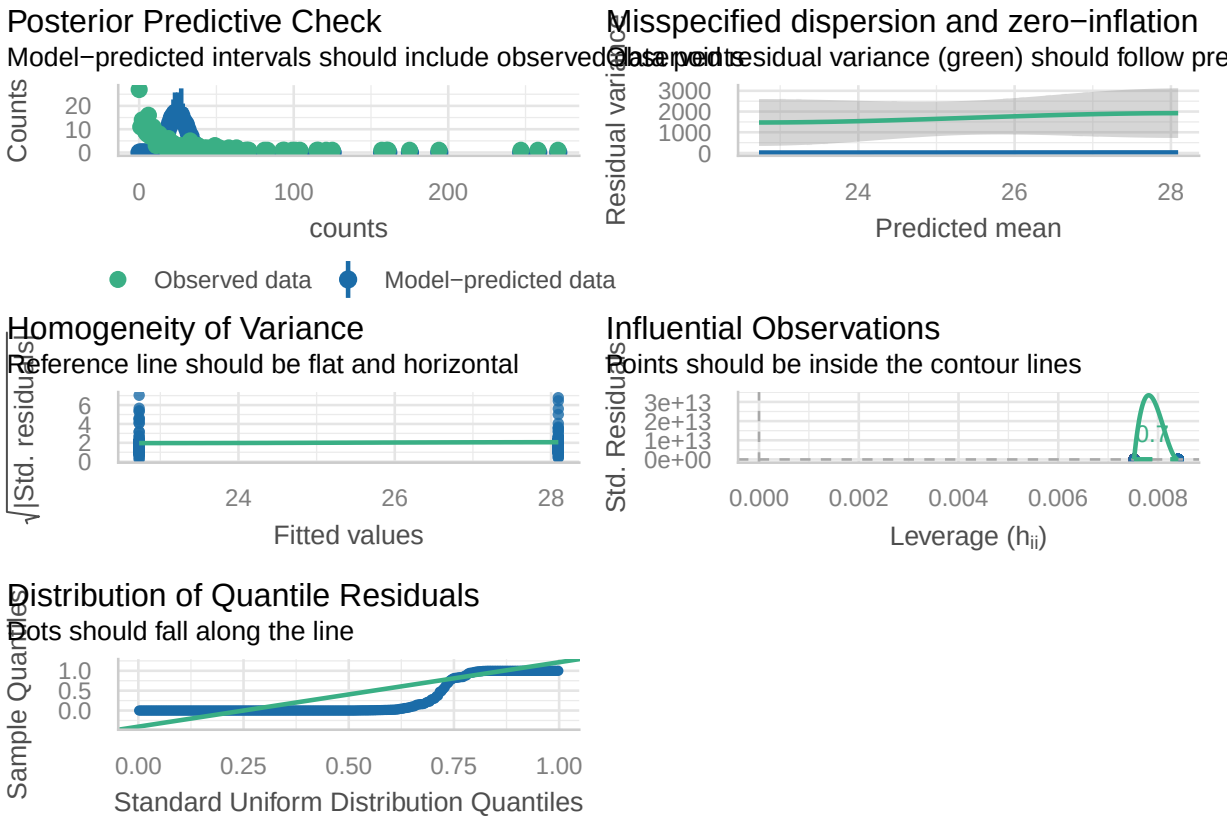
In the context of a Poisson model, dispersion refers to the mean (lambda) being equivalent to the variance, a key assumption. Overdispersion means that the variance is greater than the mean, or that the data shows more dipsersion than expected in a Poisson model.

d. Compare results with previous model, explain change in the significance of the treatment effect

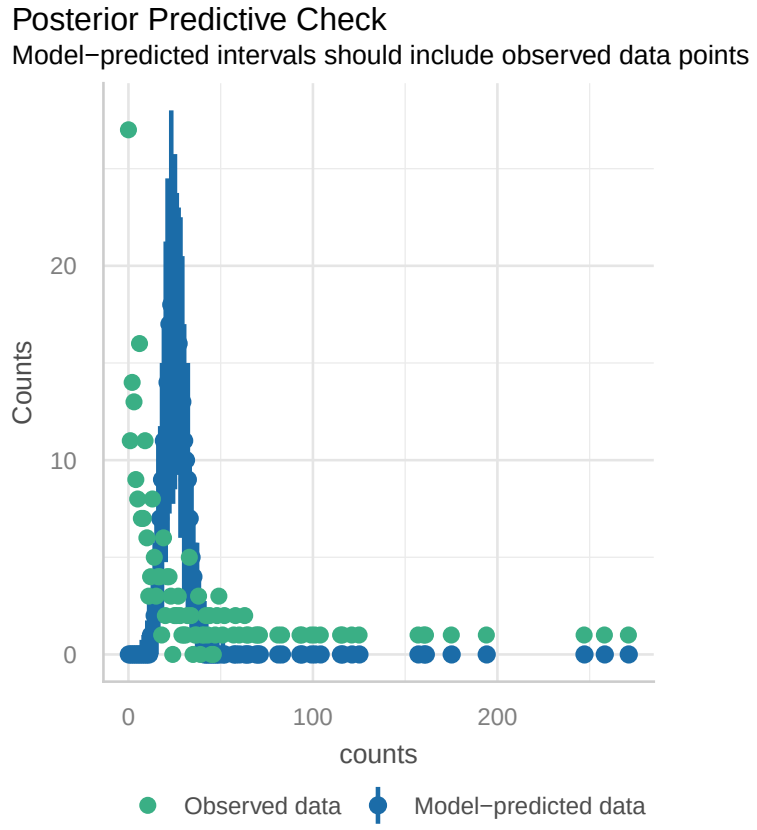
The treatment effect was more significant in the Poisson model with a p-value very close to 0, while the OLS had a p-value of 0.3 for the treatment effect. This increase in significance is likely because the Poisson model is better suited for count data like our lobster counts, where these values are positive and have a multiplicative effect.

e. Check the model assumptions. Explain results.

```
check_model(m2_pois)
```




```
check_model(m2_pois, check = "pp_check")
```



The model assumptions do not appear to be met for the Poisson model. The observed data deviates from the distribution of the model-predicted data slightly as seen in the post-predictive check, and the observed residual variance does not follow the predicted mean line in the misspecified dispersion and zero-inflation plot. These do not align with the assumption of proportional dispersion for a Poisson model.

f. Conduct tests for over-dispersion & zero-inflation. Explain results.

```
check_overdispersion(m2_pois)
```

```
## # Overdispersion test
##
##        dispersion ratio =    67.033
##   Pearson's Chi-Squared = 16758.289
##                 p-value =   < 0.001
```

If our dispersion ratio was equal to 1, this would mean our mean is equal to our variance. Since our dispersion ratio is much higher at 67.033, we can see that our data has significant overdispersion.

```
check_zeroinflation(m2_pois)
```

```
## # Check for zero-inflation
##
##    Observed zeros: 27
##   Predicted zeros: 0
##             Ratio: 0.00
```

The model has 27 observed zeros compared to none predicted by the model, so this is likely an underestimation of zeros. Overall, a Poisson model is likely not a good fit for our data based on the model checks and clear overdispersion.

g. Fit a negative binomial model using the function glm.nb() from the package `MASS` and check model diagnostics

```
library(MASS) ## NOTE: The `select()` function is masked. Use: `dplyr::select()` ##
```

```
# NOTE: The `glm.nb()` function does not require a `family` argument

# Fit negative binomial model
m3_nb <- glm.nb(counts ~ treat,
                data = spiny_counts)

# Check model output
summ(m3_nb, model.fit = FALSE)
```

Observations	252
Dependent variable	counts
Type	Generalized linear model
Family	Negative Binomial(0.55)
Link	log

	Est.	S.E.	z val.	p
(Intercept)	3.12	0.12	26.40	0.00
treat	0.21	0.17	1.23	0.22

Standard errors: MLE

h. In 1-2 sentences explain rationale for fitting this GLM model.

A negative binomial model includes a term to account for overdispersion, which allows for the violation of the Poisson model assumption that the mean is equal to the variance.

i. Interpret the treatment estimate result in your own words. Compare with results from the previous model.

Without any protection status (non-MPA), the model estimates ~22 Lobsters per observational unit.

There was an ~23.4% increase in the expected Lobster count in reef sites that are protected (MPA).

These results are almost identical to the previous estimates, with larger p-value for the treatment effect (0.22 compared to 0.00). The standard error for these estimates were also larger compared to the Poisson model, likely due to accounting for overdispersion.

```
check_overdispersion(m3_nb)
```

```
## # Overdispersion test
##
##  dispersion ratio = 1.404
##           p-value = 0.072
```

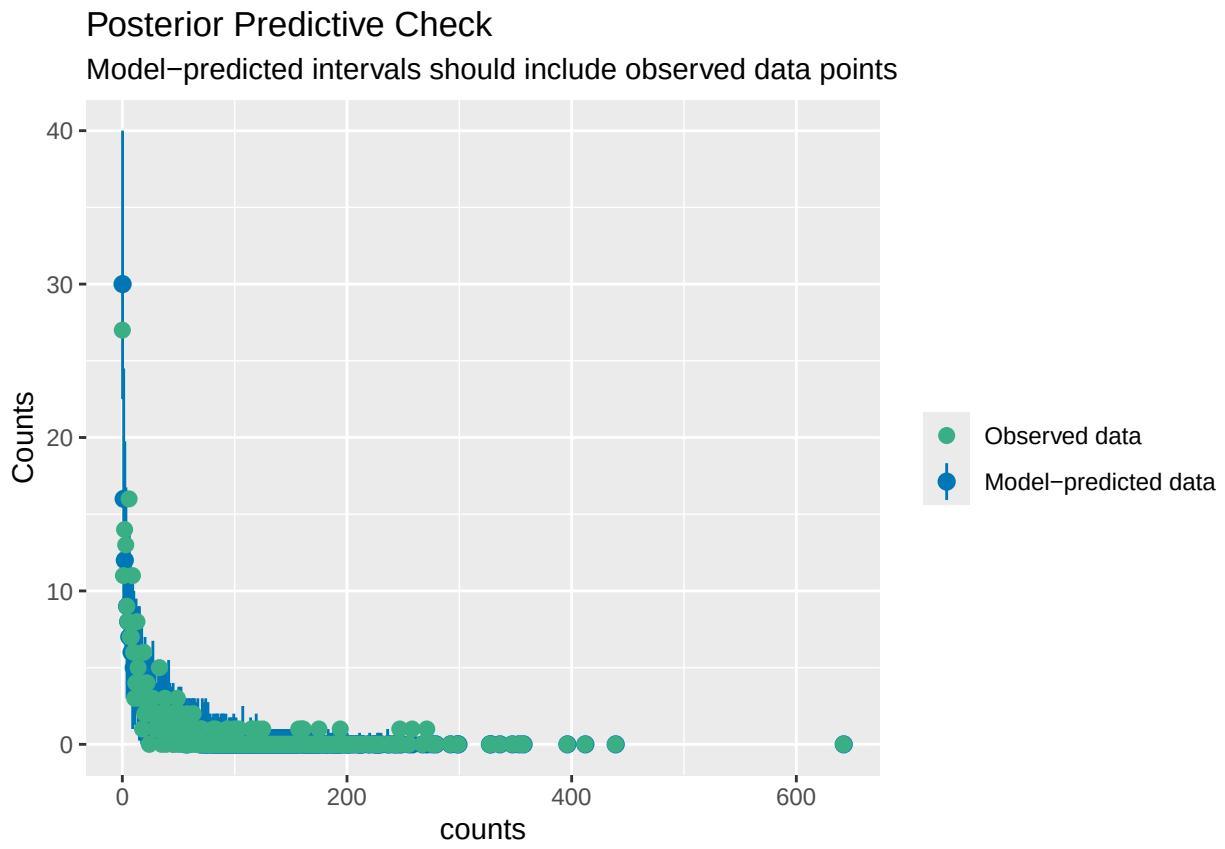
The overdispersion test results show that the overdispersion was addressed with a dispersion ratio now close to 1.

```
check_zeroinflation(m3_nb)
```

```
## # Check for zero-inflation
##
##    Observed zeros: 27
##   Predicted zeros: 30
##             Ratio: 1.12
```

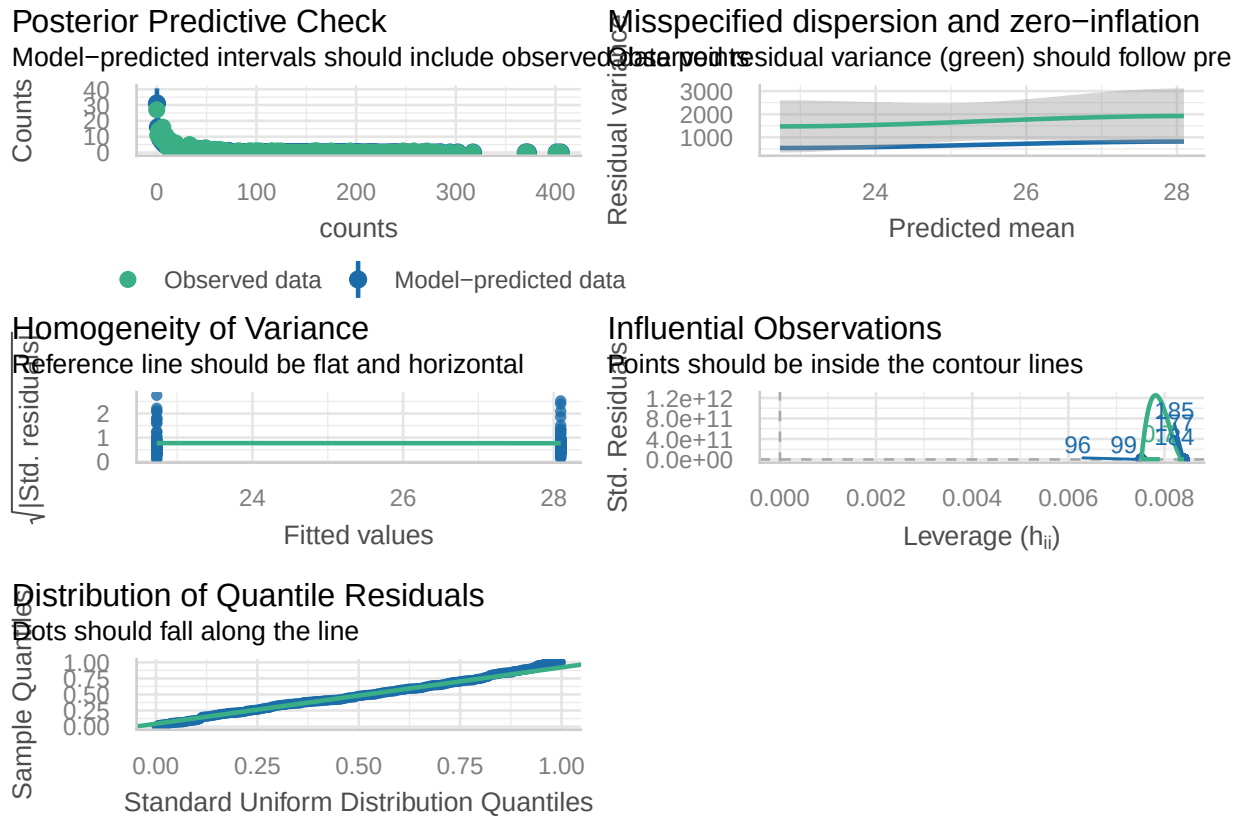
The predicted number of zeros is now closer to the observed number, with a difference of 3. This is an improvement from the Poisson model, showing no extreme zero-inflation.

```
check_predictions(m3_nb)
```



The observed data fits the model-predicted data significantly better than that of the Poisson model.

```
check_model(m3_nb)
```



Based on the model checks, the most noticable changes are that the distribution of residuals appears quite normal and along the line, and the observed residual variance more closely follows the predicted mean line in the misspecified dispersion and zero-inflation plot. These aspects support the use of a negative binomal model for our spiny lobster count data.

Step 6: Compare models **a.** Use the `export_summ()` function from the `jtools` package to look at the three regression models you fit side-by-side.

c. Write a short paragraph comparing the results. Is the treatment effect `robust` or stable across the model specifications.

```
# Compare model outputs
export_summs(m1_ols, m2_pois, m3_nb,
             model.names = c("OLS","Poisson", "NB"),
             statistics = "none")
```

```
m1_change = (5.36/22.73) * 100 # Change in OLS 37.45
m2_change = (exp(0.21) - 1) * 100 # Change in POIS 37.71
m3_change = (exp(0.21) - 1) * 100 # Change in NB 37.71
```

We need to convert these values into percent change to have a consistent comparison of the treatment effect.

When comparing treatments, the treatment effect appears to be robust and stable across the model specifications, averaging at about a 23% increase in expected lobster count between the control and treatment sites. The Poisson and Negative Binomial models predicted a slightly lower treatment effect at 23.37% compared to 23.58% in the OLS model. Only the Poisson model treatment effect is significant, supporting the positive effect of MPAs.

	OLS	Poisson	NB
(Intercept)	22.73 ***	3.12 ***	3.12 ***
	(3.57)	(0.02)	(0.12)
treat	5.36	0.21 ***	0.21
	(5.20)	(0.03)	(0.17)

*** p < 0.001; ** p < 0.01; * p < 0.05.

Step 7: Building intuition - fixed effects **a.** Create new `df` with the `year` variable converted to a factor

b. Run the following negative binomial model using `glm.nb()`

- Add fixed effects for `year` (i.e., dummy coefficients)
- Include an interaction term between variables `treat` & `year` (`treat*year`)

```
# Create new df with year as factor
ff_counts <- spiny_counts %>%
    mutate(year=as_factor(year))

# Run negative binomial model with fixed effects
m5_fixedeffs <- glm.nb(
    counts ~
        treat +
        year +
        treat*year,
    data = ff_counts)

# Check model summary
summ(m5_fixedeffs, model.fit = FALSE)
```

Observations	252
Dependent variable	counts
Type	Generalized linear model
Family	Negative Binomial(0.8129)
Link	log

c. Take a look at the regression output. Each coefficient provides a comparison or the difference in means for a specific sub-group in the data. Informally, describe the what the model has estimated at a conceptual level (NOTE: you do not have to interpret coefficients individually)

The model estimates how the expected lobster count differs over time between MPA and non-MPA reef sites over time through including a fixed effect for year and an interaction effect between treatment and year. The year coefficients control for the effects of year on the control sites, while the interaction effects allow the MPA treatment effect to vary by year. In general, the MPA treatment showed an increase in the predicted lobster count in most years beginning after 2013.

d. Explain why the main effect for treatment is negative? *Does this result make sense?

The main effect for treatment is likely negative because this is referring to the effect of an MPA on expected lobster count in 2012 (the reference level), which might be too early to have a noticeable positive difference.

	Est.	S.E.	z val.	p
(Intercept)	2.35	0.26	8.89	0.00
treat	-1.72	0.42	-4.12	0.00
year2013	-0.35	0.38	-0.93	0.35
year2014	0.08	0.37	0.21	0.84
year2015	0.86	0.37	2.32	0.02
year2016	0.90	0.37	2.43	0.01
year2017	1.56	0.37	4.25	0.00
year2018	1.04	0.37	2.81	0.00
treat:year2013	1.52	0.57	2.66	0.01
treat:year2014	2.14	0.56	3.80	0.00
treat:year2015	2.12	0.56	3.79	0.00
treat:year2016	1.40	0.56	2.50	0.01
treat:year2017	1.55	0.56	2.77	0.01
treat:year2018	2.62	0.56	4.69	0.00

Standard errors: MLE

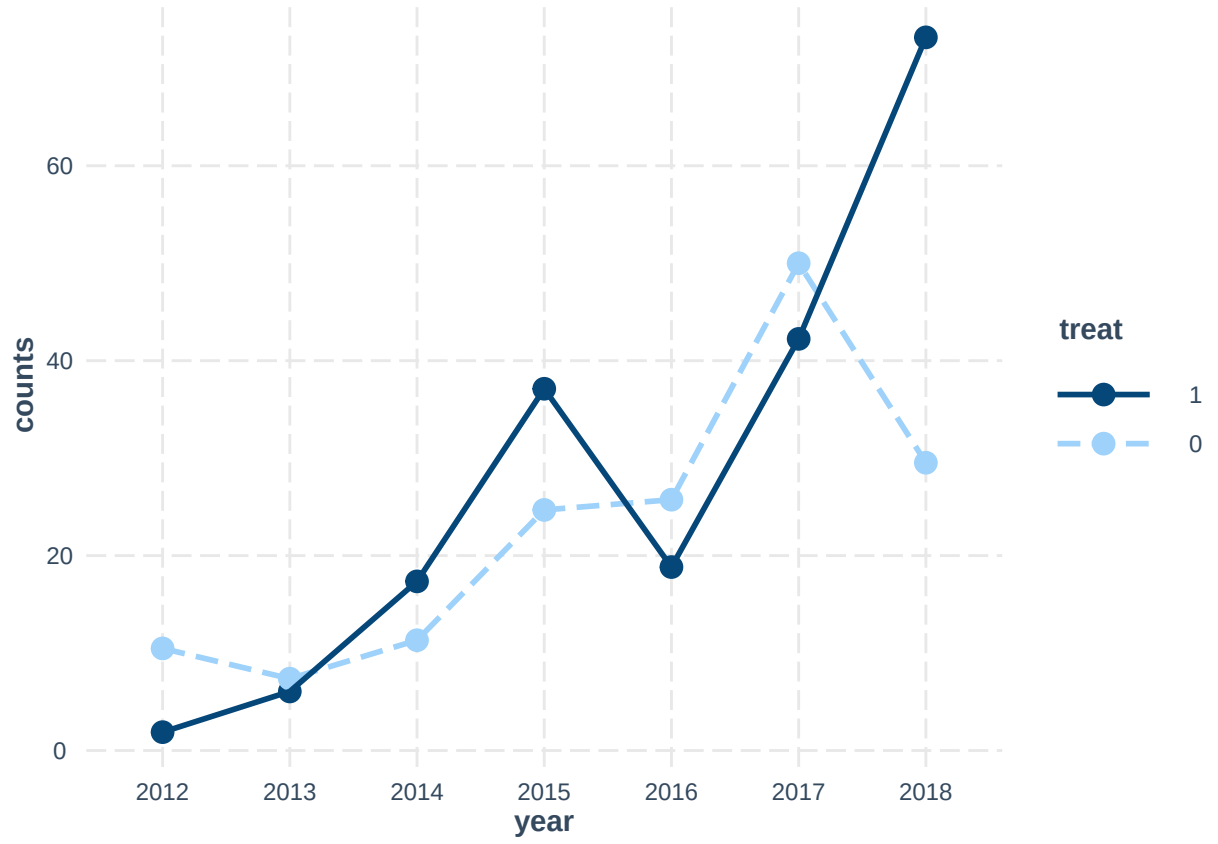
When taking into account the interaction terms that allow for a change in treatment effect, the expected count increases in later years with the MPA treatment.

e. Look at the model predictions: Use the `interact_plot()` function from package `interactions` to plot mean predictions by year and treatment status.

f. Re-evaluate your responses (c) and (b) above.

The effect of MPA treatment on expected lobster count varies throughout time, with the treatment sometimes having less of a positive effect on lobster populations than the control groups within a singular year but maintaining an overall more positive trend over the full time period. The `treat` term being negative still makes sense, as between 2012 and 2013 the expected lobster count became the same value for the treatment and control reef sites.

```
# Plot mean predictions by year and treatment
interact_plot(m5_fixedeffs, pred = year, modx = treat,
              outcome.scale = "response") # NOTE: y-axis on log-scale
```




```
# HINT: Change `outcome.scale` to "response" to convert y-axis scale to counts
```

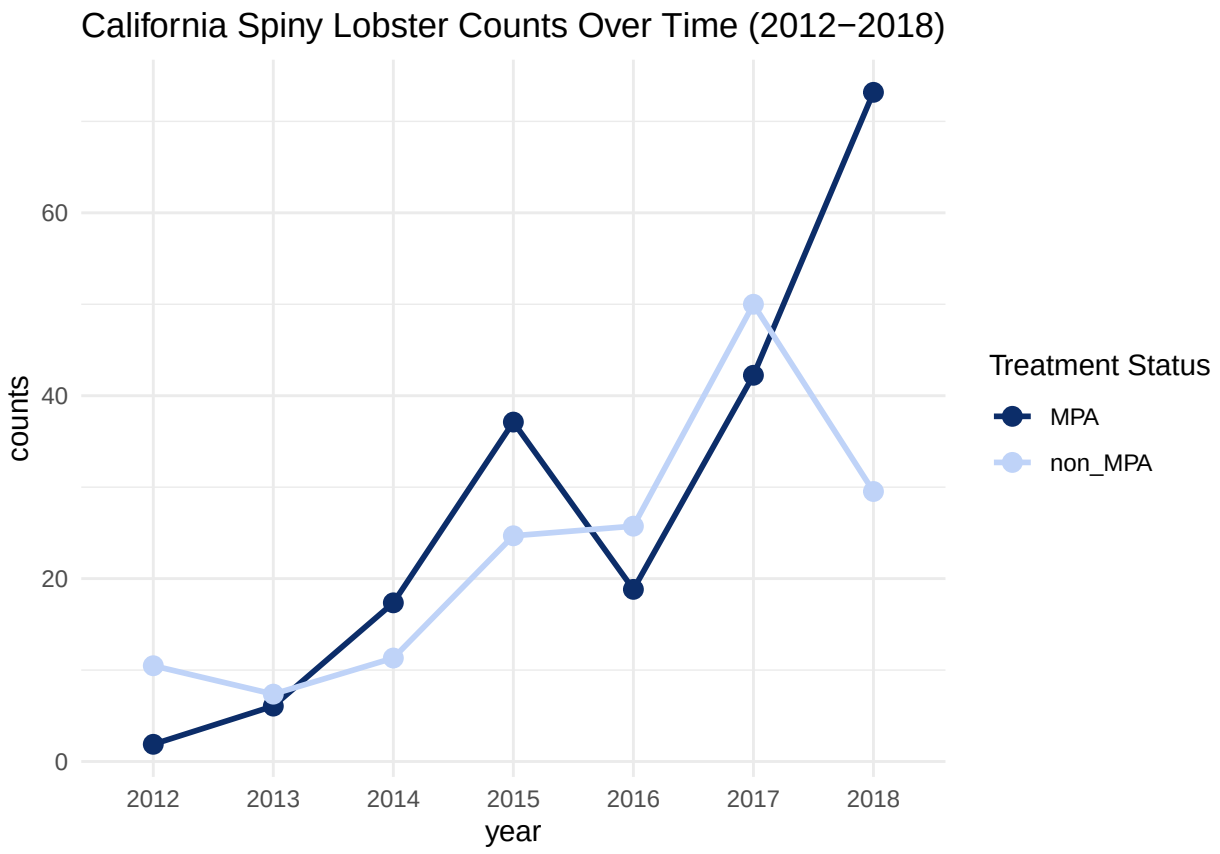
g. Using `ggplot()` create a plot in same style as the previous `interaction plot`, but displaying the original scale of the outcome variable (lobster counts). This type of plot is commonly used to show how the treatment effect changes across discrete time points (i.e., panel data).

The plot should have... - `year` on the x-axis - `counts` on the y-axis - `mpa` as the grouping variable

```
# Hint 1: Group counts by `year` and `mpa` and calculate the `mean_count`
# Hint 2: Convert variable `year` to a factor

# Create data frame with mean counts
plot_counts <- spiny_counts %>%
    mutate(year=as_factor(year)) %>%
    group_by(year, mpa) %>%
    summarize(mean_count = mean(counts, na.rm = TRUE))

# Plot treatment effect across time
plot_counts %>% ggplot(aes(x = year, y = mean_count, color = mpa, group = mpa)) +
    geom_line(linewidth = 1) +
    geom_point(size = 3) +
    labs(x = "year",
         y = "counts",
         title = "California Spiny Lobster Counts Over Time (2012-2018)",
         color = "Treatment Status") +
    scale_color_manual(values = c("#0C2D69", "#BFD3F8")) +
    theme_minimal()
```



Step 8: Reconsider causal identification assumptions

a. Discuss whether you think `spillover effects` are likely in this research context (see Glossary of terms; https://docs.google.com/document/d/1RIudsVcYhWGpqC-Uftk9UTz3PIq6stVyEpT44EPNgpE/edit?usp=sharing)

b. Explain why spillover is an issue for the identification of causal effects

c. How does spillover relate to impact in this research setting?

d. Discuss the following causal inference assumptions in the context of the MPA treatment effect estimator. Evaluate if each of the assumption are reasonable:

   1) SUTVA: Stable Unit Treatment Value assumption
   2) Exogeneity assumption

I think that spillover effects are very likely in this research context, because these reef sites are not far enough apart to exclude the possibility of the ecological benefits such as lobster abundance caused by an MPA spreading to a control site. Spillover is an issue for the identification of causal effects because it lessens the difference between the count outcomes for the treatment and control groups, blurring the source of improvement in the response variable. In this research setting, spillover could make MPAs look less effective in comparison to control areas, when these MPAs potentially have a far-reaching positive effect. In terms of causal inference assumptions, the Stable Unit Treatment Value assumption (SUTVA) is likely violated because the researchers cannot guarantee that the MPA treatment does not affect the lobster count in the control sites. The exogeneity assumption may not hold in this setting, as treatment was not randomized. Instead, treatment sites were compared with control sites serving as counterfactuals. There could therefore

still be a link between the treatment and unobserved factors, such as the amount of human activity in the MPA, habitat quality, and biodiversity differences.

EXTRA CREDIT

Use the recent lobster abundance data with observations collected up until 2024 (`extracredit_sblobstrs24.csv`) to run an analysis evaluating the effect of MPA status on lobster counts using the same focal variables.

a. Create a new script for the analysis on the updated data
b. Run at least 3 regression models & assess model diagnostics
c. Compare and contrast results with the analysis from the 2012-2018 data sample (~ 2 paragraphs)