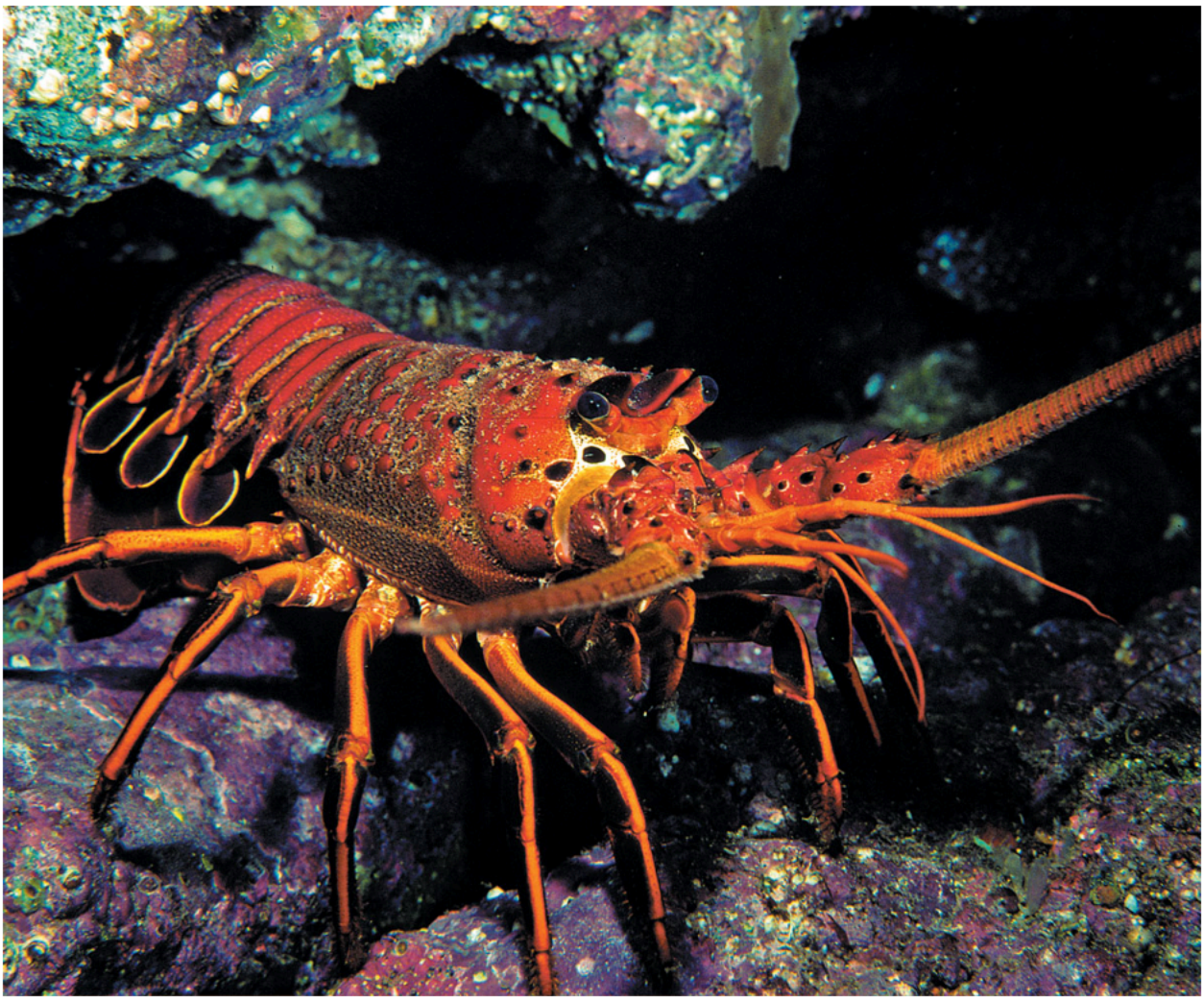
Assignment 1

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

Aakriti Poudel

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 (Aakriti Poudel): _____

```
# Load all required libraries
library(here)
library(janitor)
library(estimatr)
library(performance)
library(jtools)
library(gt)
library(gtsummary)
library(interactions)
library(ggridges)
library(ggbeeswarm)
library(see)
library(MASS)
library(tidyverse)
```

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

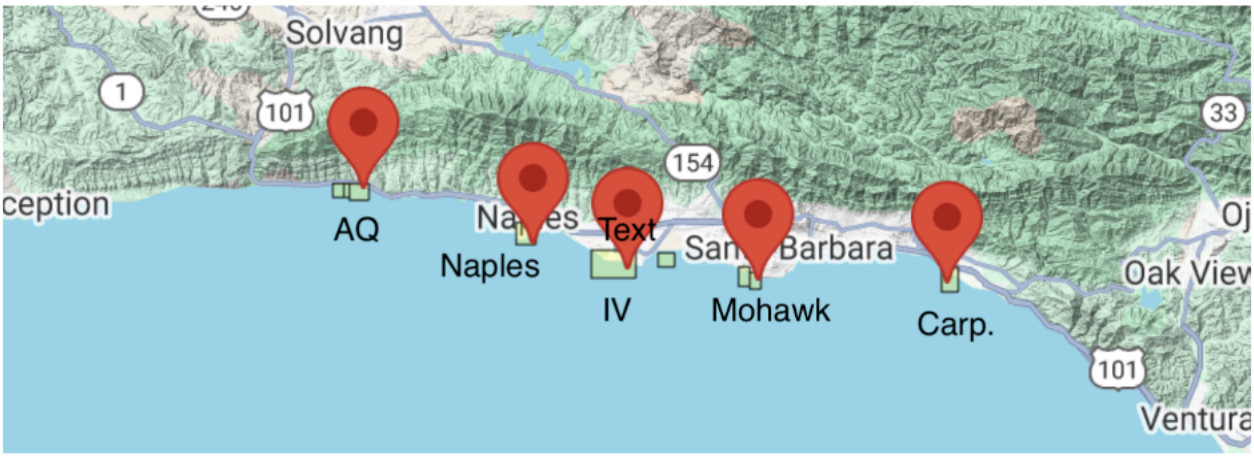
You're about to dive into some deep data collected from five reef sites in Santa Barbara County, all about the abundance of California spiny lobsters! Data was gathered by divers annually from 2012 to 2018 across Naples, Mohawk, Isla Vista, Carpinteria, and Arroyo Quemado reefs.

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!).

Answer: No, the control sites (Arroyo Quemado, Carpinteria, and Mohawk) do not provide a strong counterfactual for the treatment sites (Naples and Isla Vista). For the control sites to provide a strong counterfactual, they must represent what would have happened to the treatment sites had they not been designated as marine protected areas (MPAs), holding all other factors constant. However, MPAs are not randomly assigned. Naples and Isla Vista were likely selected for protection due to factors such as ecological importance, marine habitat quality, biodiversity value, or historical fishing pressure. These same factors can directly influence lobster abundance and size, independent of MPA status. As a result, observed differences between MPA and non-MPA reefs may reflect pre-existing site characteristics rather than the causal effect of protection.

Although all five reefs are located in Santa Barbara County and were surveyed using consistent methods over the same period of time, the non-MPA reefs cannot fully replicate the ecological conditions and socioeconomic pressure of the MPA reefs. Therefore, the selection bias is likely and weakens the validity of the control sites as a counterfactual, meaning these sites do not provide a sufficient comparison under ceteris paribus.

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"`)

# Load raw spiny lobster abundance data
rawdata <- read_csv(here("data", "spiny_abundance_sb_18.csv")) %>%
  clean_names() %>%
  mutate(size_mm = na_if(size_mm, -99999)) # Replace -99999 with NA in the size_mm column
```

c. Create a new **df** named **tidydata**. 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 `tidydata` with all reef location
tidydata <- rawdata %>%
    mutate(reef = case_when(site == "AQUE" ~ "Arroyo Quemado",
                            site == "CARP" ~ "Carpenteria",
                            site == "MOHK" ~ "Mohawk",
                            site == "IVEE" ~ "Isla Vista",
                            site == "NAPL" ~ "Naples")) %>%
  # Convert reef to a factor and add the labels
  mutate(reef = factor(reef,
                       levels = 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 dataframe with lobster counts summarized by site, year and transect
spiny_counts <- tidydata %>%
  group_by(site, year, transect) %>%
  summarize(counts = sum(count, na.rm = TRUE),
            mean_size = mean(size_mm, na.rm = TRUE),
            .groups = "drop") %>%
    mutate(mpa = case_when(site %in% c("IVEE", "NAPL") ~ "MPA",
                           site %in% c("MOHK", "CARP", "AQUE") ~ "non_MPA"),
           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**:

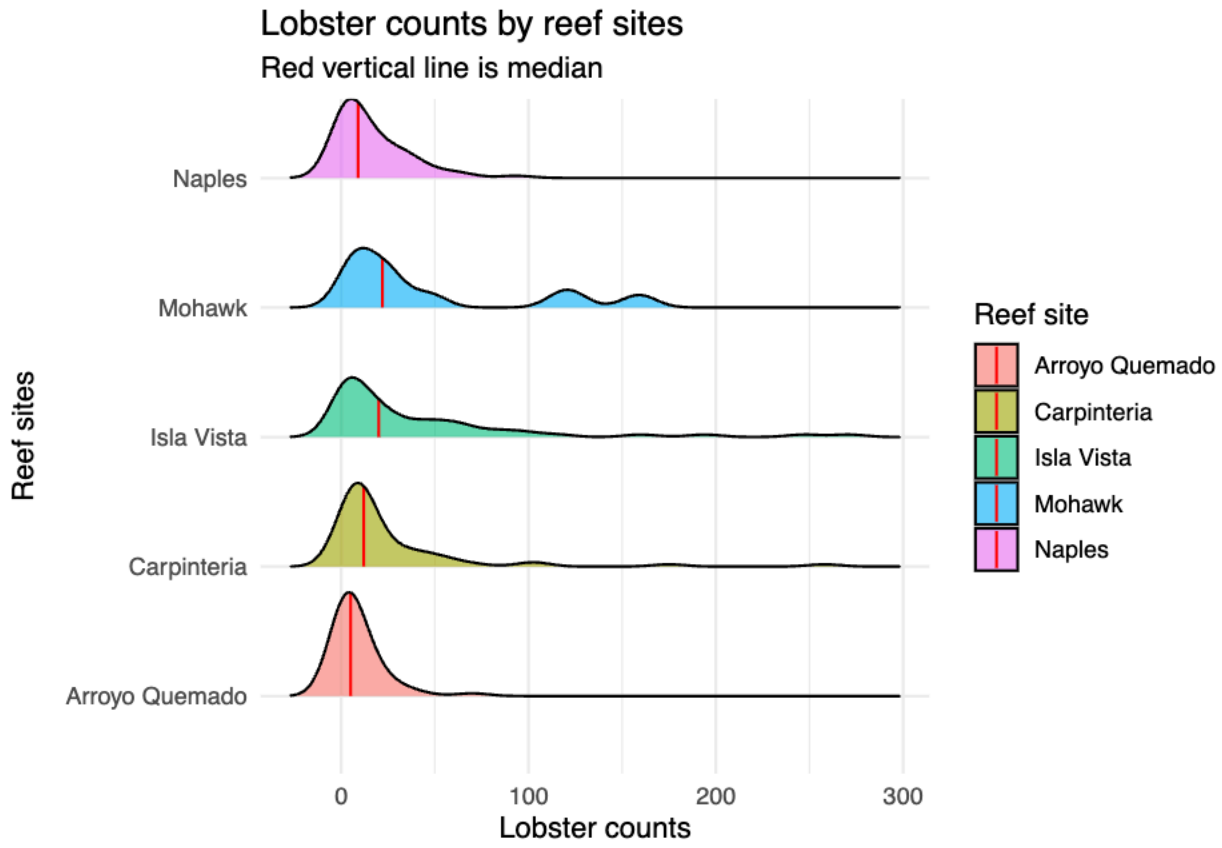
1) grouped by reef site
2) grouped by MPA status
3) grouped by year

Create a plot of lobster **size** :

4) You choose the grouping variable(s)!

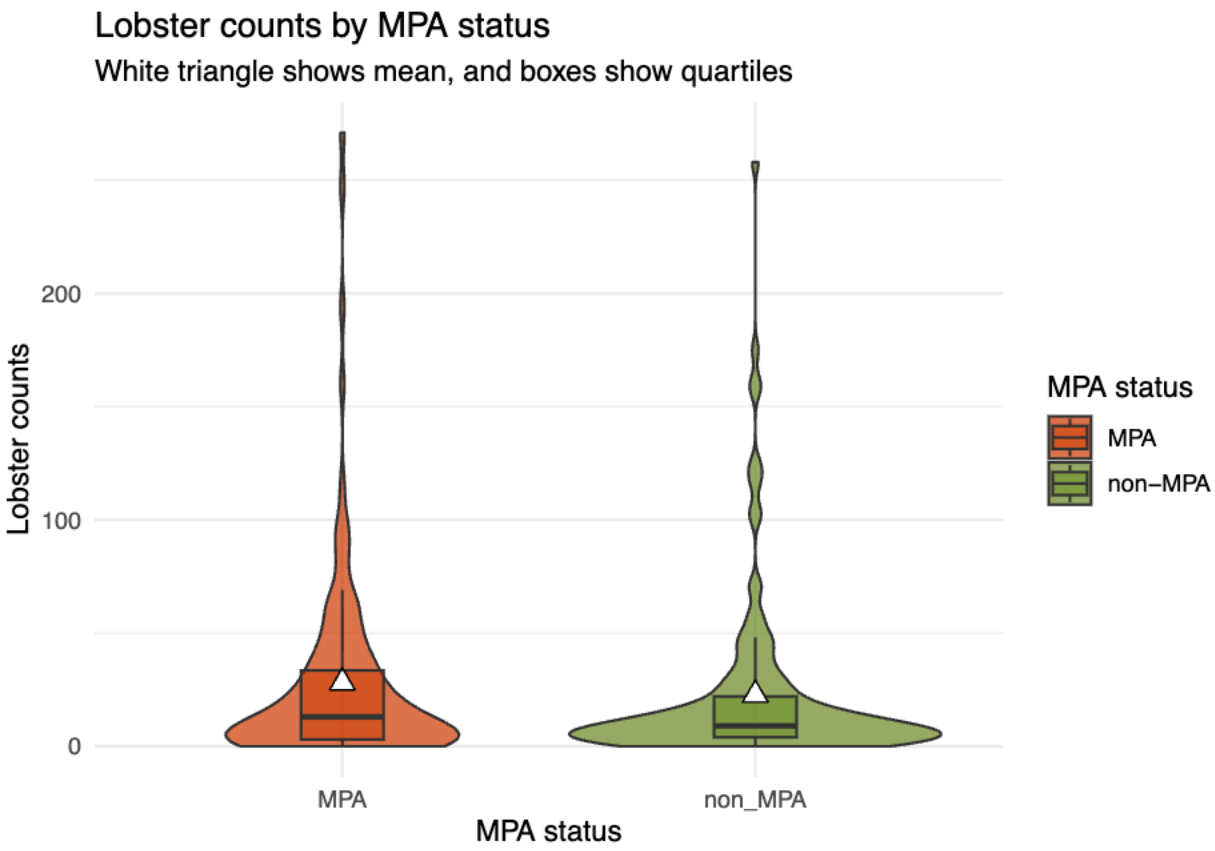
```
# Plot 1: Ridge plot - Lobster counts grouped by reef sites
# Recode site names
spiny_counts <- spiny_counts %>%
  mutate(site_label = case_when(
    site == "AQUE" ~ "Arroyo Quemado",
    site == "CARP" ~ "Carpinteria",
    site == "MOHK" ~ "Mohawk",
    site == "IVEE" ~ "Isla Vista",
    site == "NAPL" ~ "Naples",
    TRUE ~ site))

# Ridge plot with legend
spiny_counts %>% ggplot(aes(x = counts, y = site_label, fill = site_label)) +
    geom_density_ridges(alpha = 0.6,
                        quantile_lines = TRUE,
                        quantiles = 2,
                        quantile_fun = median,
                        vline_color = "red",
                        vline_size = 0.5,
                        scale = 0.8) +
    labs(title = "Lobster counts by reef sites",
         subtitle = "Red vertical line is median",
```

```
        x = "Lobster counts",
        y = "Reef sites",
        fill = "Reef sites",
        color = "Median") +
    theme_minimal() +
    theme(legend.position = "right",
          axis.title.y = element_text(margin = margin(r = 10))) +
    guides(fill = guide_legend(title = "Reef site"),
           color = guide_legend(override.aes = list(linetype = 1,
                                                    linewidth = 1,
                                                    color = "red")))
```

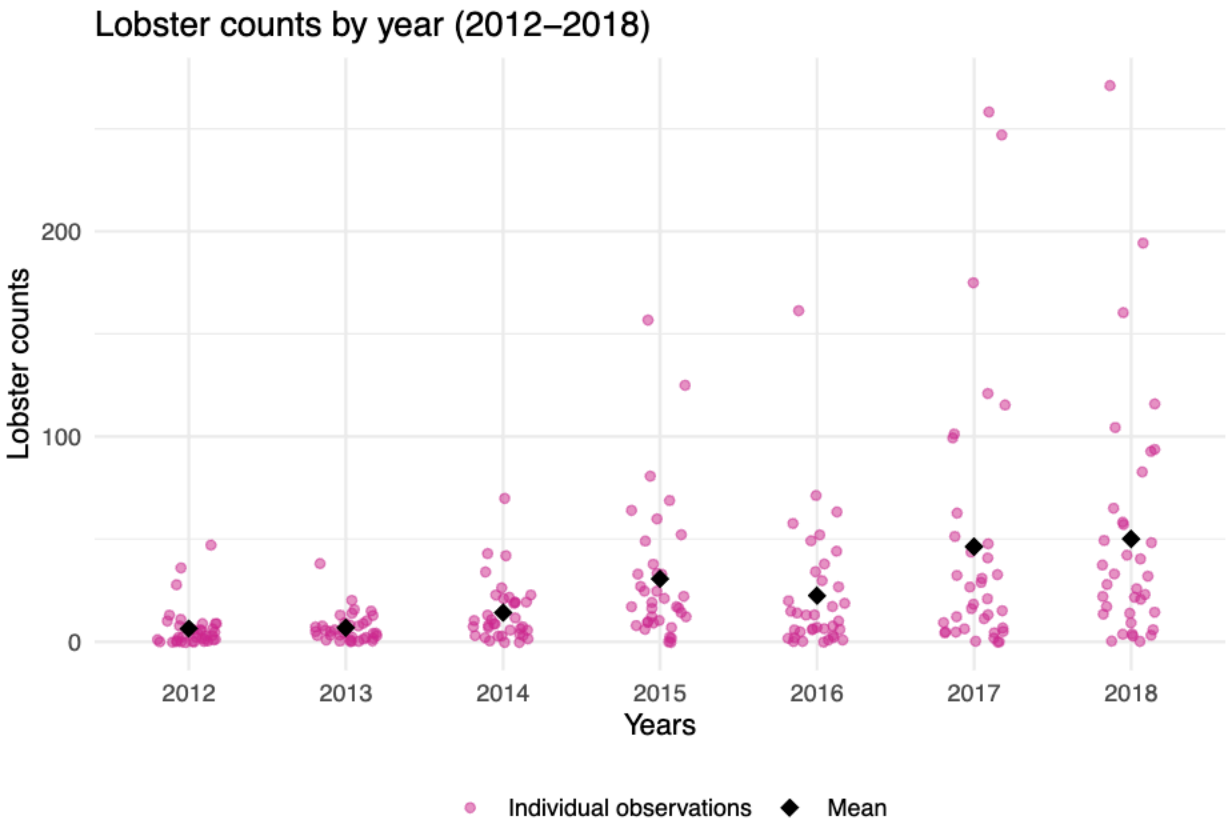



```
# Plot 2: Violin Plot - Lobster counts grouped by MPA status
spiny_counts %>%
    ggplot(aes(x = mpa, y = counts, fill = mpa)) +
    geom_violin(alpha = 0.7) +
    geom_boxplot(width = 0.2, alpha = 0.5, outlier.shape = NA) +
    stat_summary(fun = mean, geom = "point", shape = 24, size = 3, fill = "white") +
    labs(title = "Lobster counts by MPA status",
         subtitle = "White triangle shows mean, and boxes show quartiles",
         x = "MPA status",
         y = "Lobster counts",
         fill = "MPA status") +
    scale_fill_manual(values = c("MPA" = "orangered3", "non_MPA" = "olivedrab4"),
                      labels = c("MPA" = "MPA", "non_MPA" = "non-MPA")) +
```

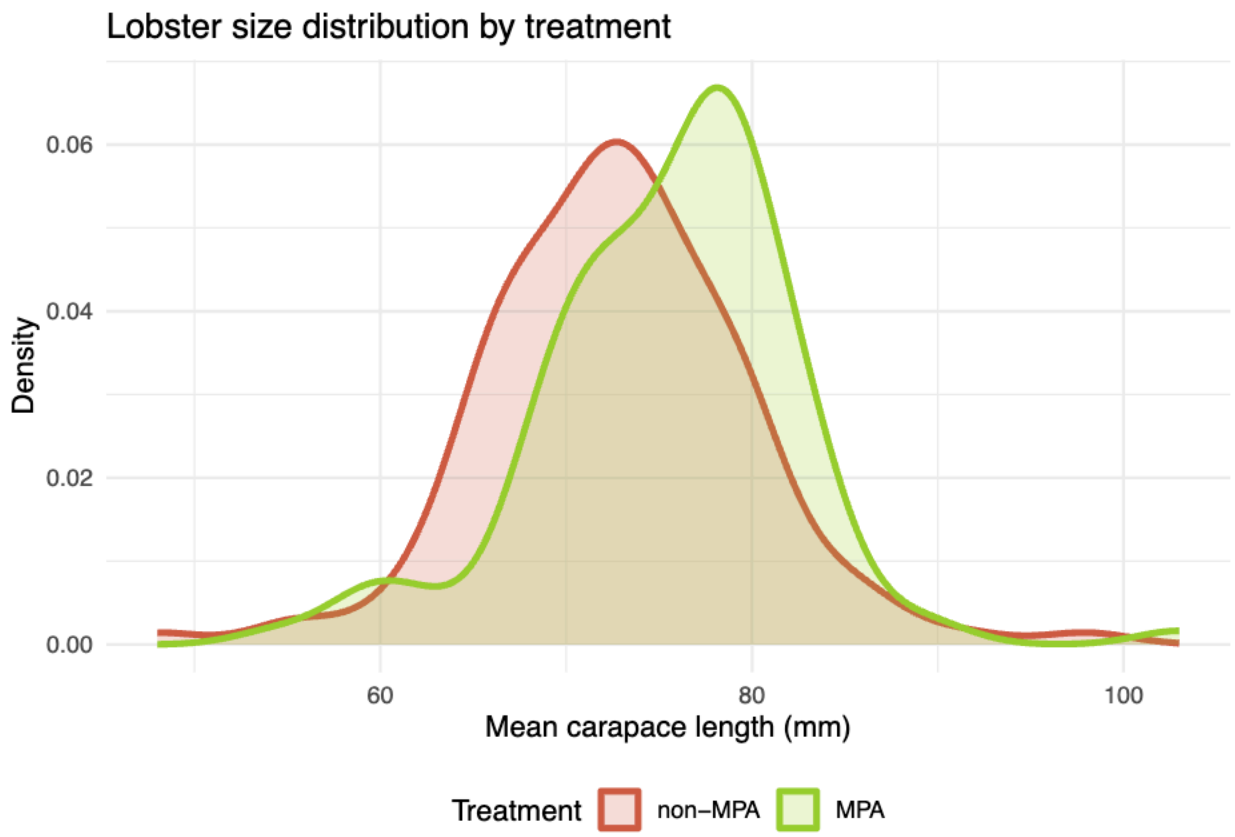
```
    theme_minimal()
```




```
# Plot 3: Jitter plot - Lobster counts grouped by year
spiny_counts %>%
    ggplot(aes(x = factor(year), y = counts)) +
    geom_jitter(aes(color = "Individual observations"),
                alpha = 0.5, width = 0.2, size = 1.4) +
    stat_summary(fun = mean, geom = "point",
                 aes(color = "Mean"),
                 size = 3, shape = 18) +
    labs(title = "Lobster counts by year (2012-2018)",
         x = "Years",
         y = "Lobster counts",
         color = "") +
    scale_color_manual(values = c("Individual observations" = "maroon3",
                                  "Mean" = "black")) +
    theme_minimal() +
    theme(legend.position = "bottom")
```




```r
# Plot 4: Density plot - Lobster size grouped by treatment
spiny_counts %>%
    filter(!is.na(mean_size)) %>%
    ggplot(aes(x = mean_size, color = factor(treat), fill = factor(treat))) +
    geom_density(alpha = 0.2, linewidth = 1.2) +
    labs(title = "Lobster size distribution by treatment",
         x = "Mean carapace length (mm)",
         y = "Density",
         color = "Treatment",
         fill = "Treatment") +
    scale_color_manual(values = c("0" = "coral3", "1" = "olivedrab3"),
                       labels = c("0" = "non-MPA", "1" = "MPA")) +
    scale_fill_manual(values = c("0" = "coral3", "1" = "olivedrab3"),
                      labels = c("0" = "non-MPA", "1" = "MPA")) +
    theme_minimal() +
    theme(legend.position = "bottom")
```



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

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

# Summary table: Compare means by treatment group using gtsummary::tbl_summary()
means_trt_group <- spiny_counts %>%
  gtsummary::tbl_summary(by = mpa,
                         include = c(counts, mean_size),
                         statistic = list(all_continuous() ~ "{mean} ({sd})"),
                         label = list(counts = "Lobster abundance",
                                      mean_size = "Mean size of lobster")) %>%
    add_p() %>%
    modify_header(label ~ "**Characteristics**") %>%
    modify_spanning_header(c("stat_1", "stat_2") ~ "**Reef sites**")

means_trt_group
```

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.

Characteristics	Reef sites		p-value[2]
	MPA N = 119[1]	non_MPA N = 133[1]	
Lobster abundance	28 (44)	23 (39)	0.3
Mean size of lobster	76 (7)	73 (7)	<0.001
Unknown	12	15	

[1] Mean (SD)
[2] Wilcoxon rank sum test

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

# OLS regression: Lobster counts regressed on treatment
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

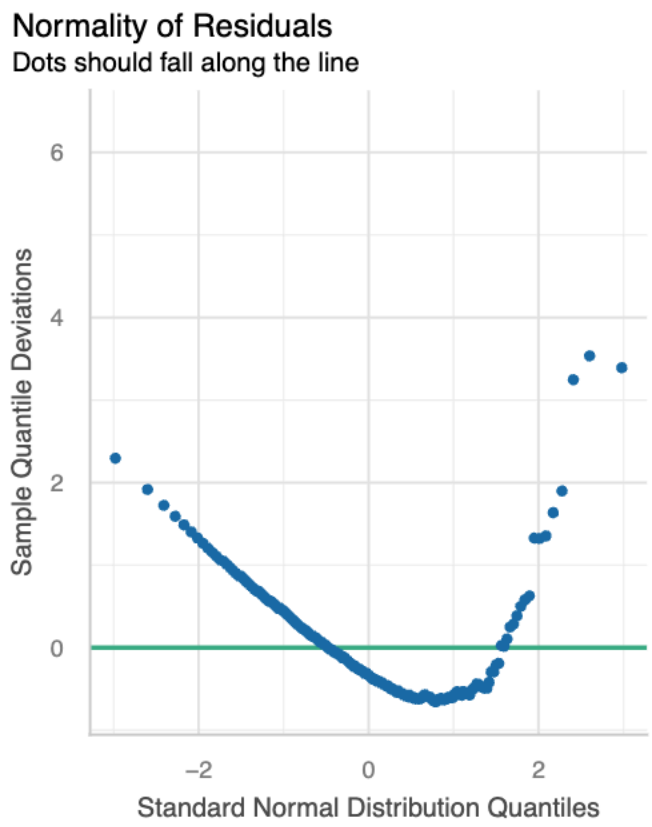
Standard errors: OLS

Interpretation: The model suggests that the estimated average number of spiny lobsters is about 28 on MPA reefs and about 23 on non-MPA reefs, a difference of roughly 5 lobsters. However, this difference is not statistically significant ($p = 0.30$), meaning we cannot be confident that the higher lobster abundance in MPAs is due to protection rather than random variation or other factors not captured in the model.

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

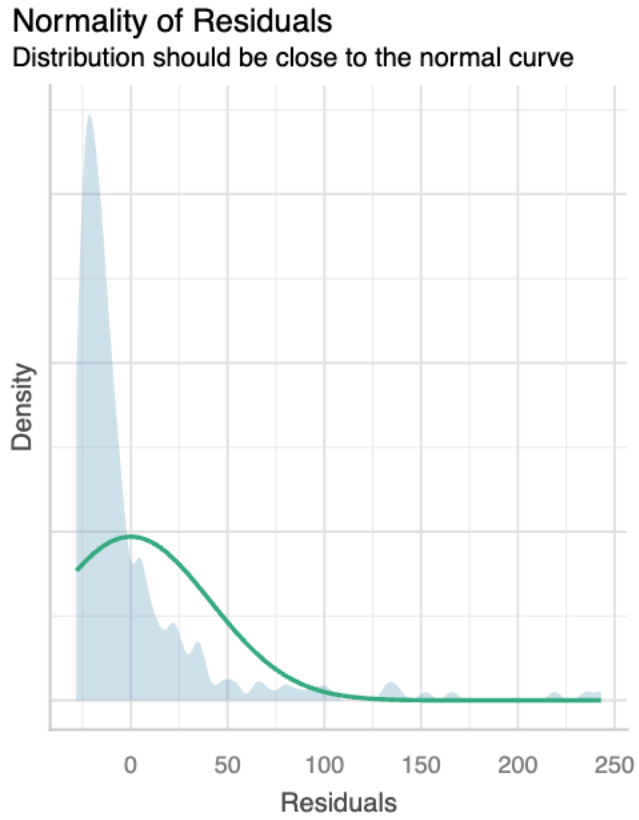
d. Explain the results of the 4 diagnostic plots. Why are we getting this result?

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



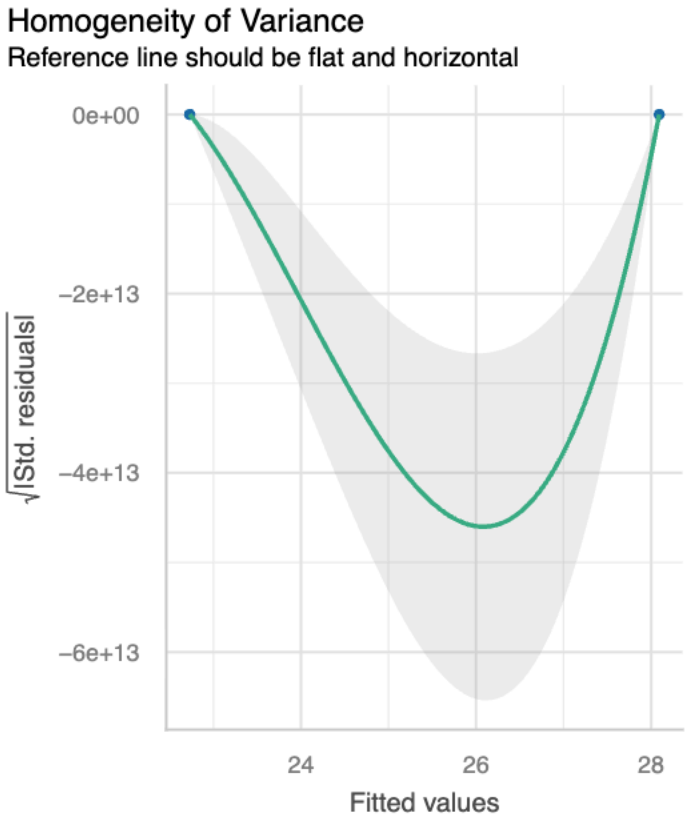
Explanation: This q–q (quantile-quantile) plot shows that the residuals do not follow a normal distribution. Instead of falling along a straight line, the points are curved. Residuals, which are the differences between the observed values and the model's fitted values, should be centered around zero. This pattern suggests skewness and the presence of possible outliers, indicating that the residuals do not follow a normal pattern.

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



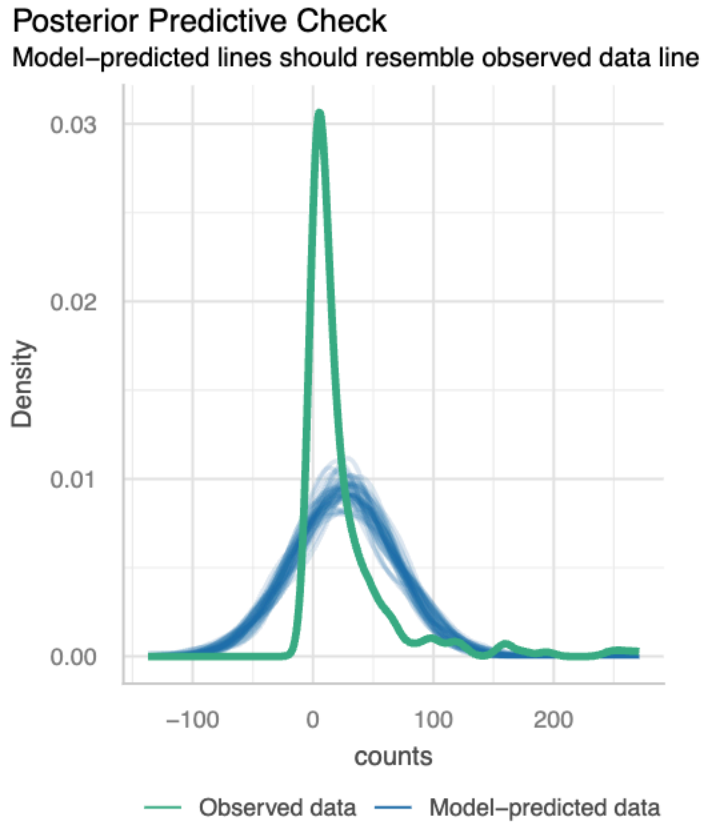
Explanation: The density distribution plot shows that the residuals do not follow a normal distribution. Instead of forming a symmetric, bell-shaped curve, the distribution is skewed to the right. Residuals should be centered around zero and spread evenly. The long tail and uneven shape suggest skewness and possible outliers, indicating that the normality assumption is not met.

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



Explanation: This plot shows that the homogeneity of variance assumption of OLS regression is violated. Instead of forming a straight, horizontal line, the residuals display a curved, non-linear pattern. This means the spread of the residuals changes across fitted values, indicating the presence of heteroscedasticity rather than constant variance.

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



Explanation: This posterior predictive check (PPC) shows a mismatch between the observed data and the model's predictions. The observed distribution is much more peaked and skewed, while the model-predicted distributions are wider and smoother. This indicates that the model does not fully capture the shape and variability of the observed data, suggesting a poor model fit.

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

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.

c. Explain the statistical concept of dispersion and overdispersion in the context of this model. write the assumptions, general description

Answer: Dispersion is the measure of how spread out the data points are. In a poisson model, the variance must equal the mean. Overdispersion occurs when the actual variability in lobster counts exceeds what the poisson model predicts (Variance > Mean). This can lead to misleading results and suggests that a more flexible model is needed to capture the true behavior.

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

```
#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 regression model
```

```r
m2_pois <- glm(counts ~ treat, data = spiny_counts,
               family = poisson(link = "log"))

# Interpret the predictor coefficient
summ(m2_pois, model.fit = FALSE)
```

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

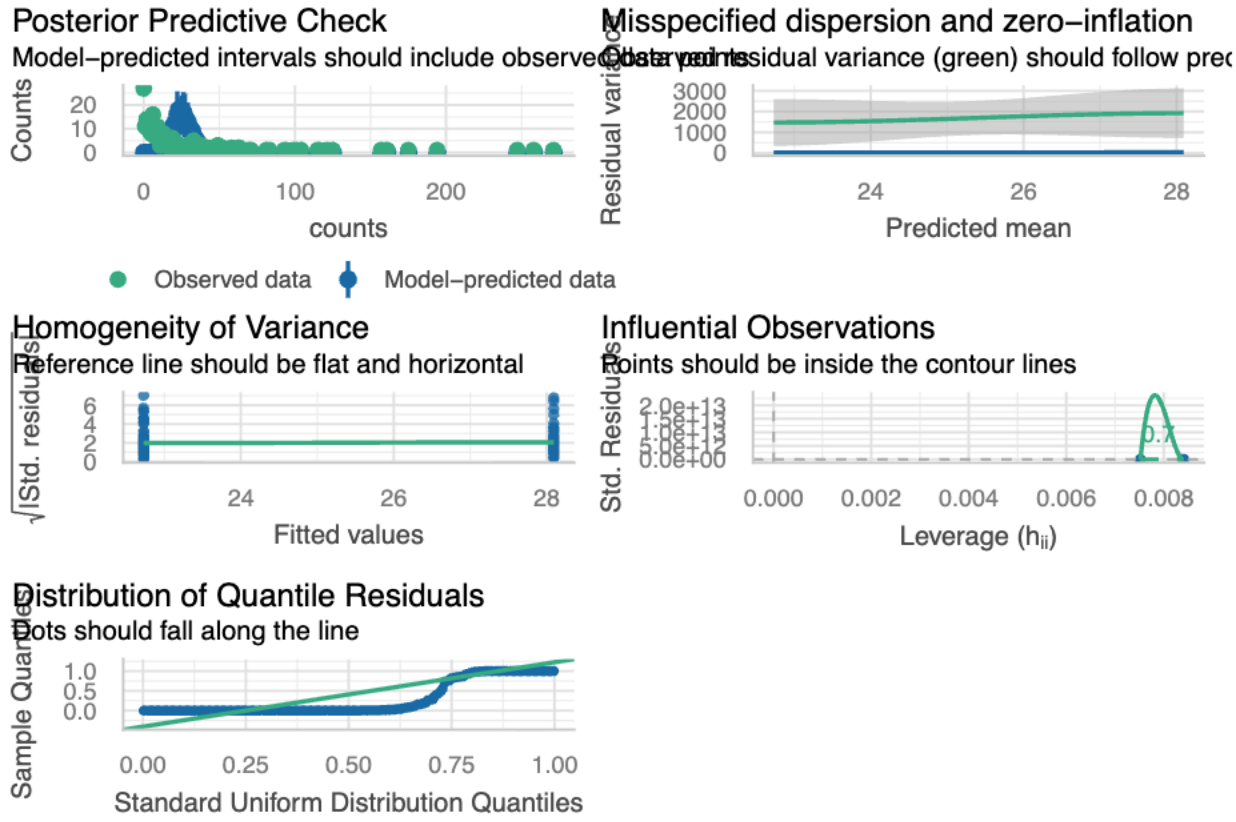
Standard errors: MLE

e. Check the model assumptions. Explain results.

Explanation: The poisson model uses a log link, so the coefficients must be exponentiated to interpret them as counts. The intercept implies an expected lobster count of about 23 lobsters in non-treated sites. The treatment effect indicates that MPAs have about 23% more lobsters than non-MPA sites. This effect is statistically significant. This suggests that protection may be associated with higher lobster numbers. Because lobster counts are whole numbers, a poisson model is more appropriate than a standard linear model. The poisson model is designed for count data and assumes that the average number of lobsters and the variation in those counts are similar. This property is called dispersion, and when the variance is roughly equal to the mean, the model gives reliable results. If the counts vary more than expected, this is called overdispersion, and the poisson model may underestimate uncertainty, making confidence intervals too narrow and statistical tests too optimistic. Since our model fits the Poisson assumptions reasonably well, we can conclude that MPAs are likely associated with higher lobster numbers, and the estimated 23% increase is meaningful.

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

```r
check_model(m2_pois)
```



Explanation: The posterior predictive check shows a mismatch between the model and reality. The observed data (green) spikes massively at zero, whereas the model-predicted distribution (blue) is centered to the right compared to the observed data. This indicates that the model is failing to account for zero-inflation, missing the most frequent observation in the dataset.

The misspecified dispersion and zero inflation plot highlights a critical failure in the model's error structure. The observed residual variance (green) is significantly higher than the predicted variance (blue), which remains near zero. This is a clear sign of overdispersion, meaning the data is far more spread out than the model assumes.

The homogeneity of variance shows a non-flat reference line and vertical clusters of residuals that suggest the variance is not constant across fitted values.

The distribution of quantile residuals plot indicates that the model is not fit for the data. Instead of the dots falling along the straight green diagonal line as they should in a well-fitted model, they follow a staircase pattern, flatlining at the bottom before jumping sharply upward. This suggests this model fails to handle a large number of zeros, confirming that the chosen probability distribution does not match the actual data generation process.

The influential observations plot shows that a few extreme data points are having an oversized impact. These points, seen as the sharp green peak on the right, are significant outliers that pull the model towards them. Because the model is working too hard to fit these unusual cases, results may be biased and may not accurately reflect the actual trend.

```
check_overdispersion(m2_pois)
```

```
## # Overdispersion test
##
```

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

Explanation: The dispersion ratio of 67.033 indicates that the variation in data is 67 times higher than what the model expects. The ratio should be close to 1 for a perfect poisson model. The p value (< 0.001) confirms that this overdispersion is statistically significant and not just a result of random chance.

The pearson's chi-squared measures the total difference between the observed data and the model prediction. The extremely high value of 16758.289 shows that the model is failing to capture the true structure of the data. When combined with the high dispersion ratio, it proves that the poisson distribution is the wrong choice for this data set.

```
check_zeroinflation(m2_pois)
```

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

Explanation: The zero-inflation check indicates that there are 27 zeros in the actual data but the model predicts none. This gives a ratio of 0.00, showing that the model completely ignores the zero observations in data. This ratio should be close to 1.00. in well fitted model. This discrepancy confirms that data are zero-inflated, and because the model cannot handle these frequent zeros, it produces biased estimates and inaccurate predictions.

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

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

Explanation: We fitted the GLM model (negative binomial model) because it includes dispersion parameter that allows the variance to be different from the mean. This adds flexibility, and lets the model account for the overdispersion found in lobster data, which a poisson model is unable to do.

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

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

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

m3_nb <- glm.nb(counts ~ treat, data = spiny_counts)

summ(m3_nb, model.fit = "none")
```

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

Explanation: Both models predict an average of 23 lobsters in non-MPA reefs and a similar increase in abundance (23–24%) for MPA reefs. However, their conclusions differ significantly when we consider p value.

	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

The poisson model shows a significant effect ($p < 0.01$), while the negative binomial model shows the effect is actually non-significant ($p = 0.22$). The negative binomial model account for the extra variability in the data, and indicates that the significance in the poisson model was likely an error caused by overdispersion.

```
check_overdispersion(m3_nb)
```

```
## # Overdispersion test
##
##  dispersion ratio = 1.398
##           p-value = 0.088
```

Explanation: The dispersion ratio for the poisson regression model was very high (67.033), indicating substantial overdispersion and a poor fit to the data. The negative binomial model has a dispersion ratio of 1.398, suggesting that the observed variance is only slightly higher than what the model expects. Since this ratio is close to 1, it indicates that the negative binomial model adequately accounts for the overdispersion present in the data.

The p-value of 0.088 indicates that there is insufficient statistical evidence to reject the null hypothesis at the 5% significance level. Therefore, overdispersion is not statistically significant in the negative binomial model, supporting its appropriateness over the poisson model for this analysis.

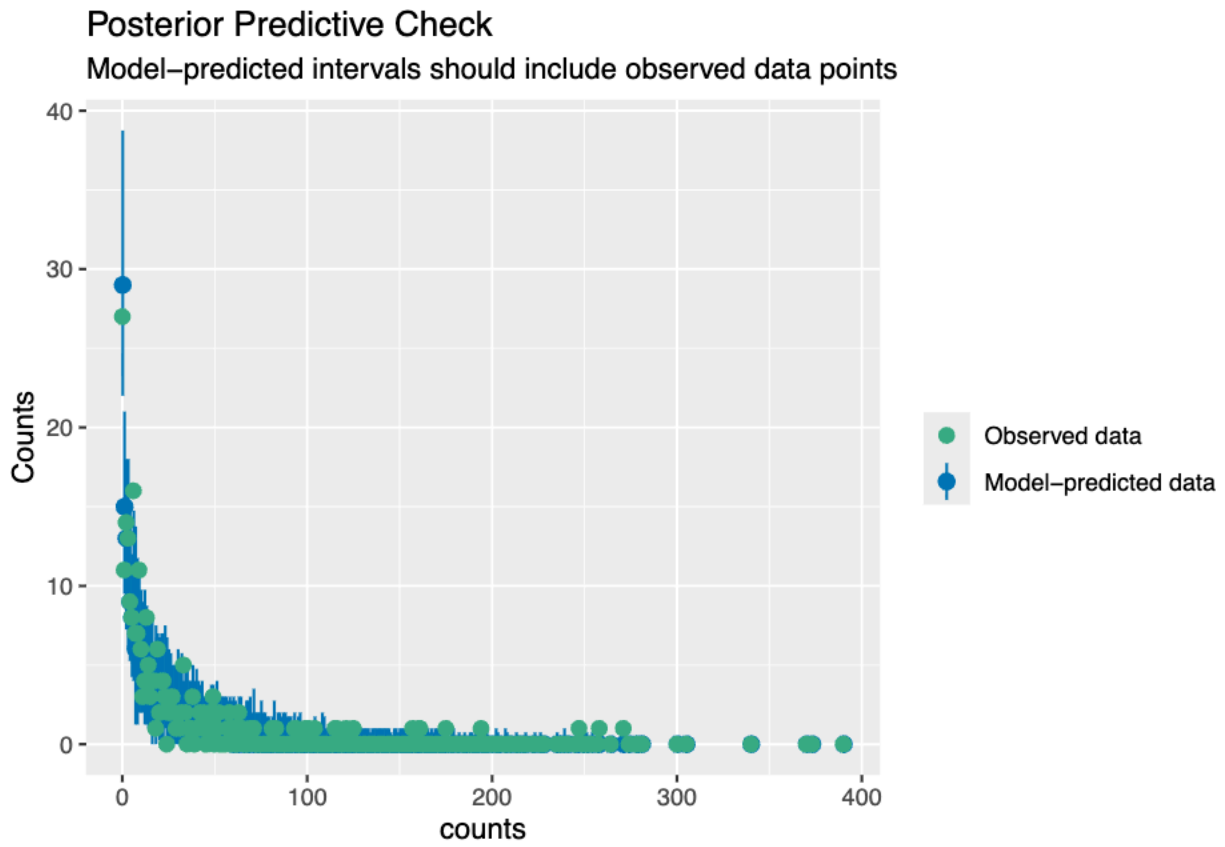
```
check_zeroinflation(m3_nb)
```

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

Explanation: The zero-inflation check indicates that the observed data contain 27 zero counts, and the negative binomial model predicts 30 zeros, resulting in a ratio of 1.12. This ratio is close to 1, suggesting that the model captures the frequency of zero observations and that there is no substantial zero-inflation remaining.

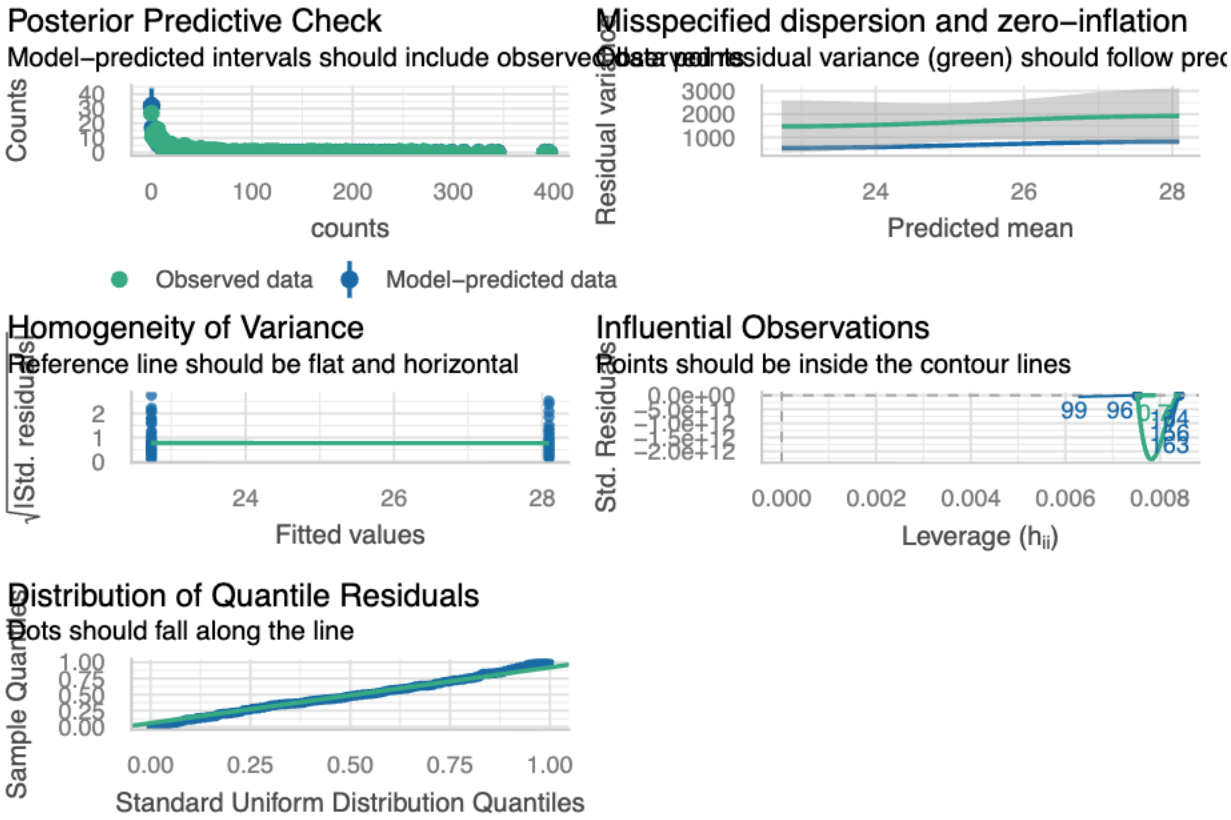
In contrast, the poisson regression model under-predicted the number of zero counts relative to the observed data, indicating that it did not adequately account for excess zeros. This suggests potential model misspecification under the poisson assumption, whereas the negative binomial model provides a better fit to the data.

```
check_predictions(m3_nb)
```



Explanation: The posterior predictive check indicates that the negative binomial model fits the data better than both OLS and poisson models. Unlike OLS, which fails to capture the skewed and heteroskedastic nature of the count data, and the poisson model, which underestimates variability and zero counts, the negative binomial model closely aligns with the observed distribution. Most observed values fall within the model-predicted intervals, suggesting that the negative binomial model captures overdispersion and provides the most appropriate representation of the data.

```
check_model(m3_nb)
```



Explanation: The above plots indicate that the negative binomial model provides an overall adequate fit to the data compared to the OLS and poisson model.

The posterior predictive check effectively captures the long-tail nature of the data, with the blue prediction intervals generally covering the observed green data points.

The posterior predictive check shows that most observed counts fall within the model-predicted intervals, suggesting good agreement between observed and predicted values. The residual variance generally follows the predicted variance, indicating that overdispersion is largely accounted for, while the quantile residuals align closely with the reference line, supporting the distributional assumptions of the model. Although a few observations appear influential, there is no strong evidence of systematic misspecification, making the negative binomial model an appropriate choice for these data.

In the misspecified dispersion and zero-inflation,the observed variance (green line) is significantly higher than the predicted variance (blue line) and sits outside the shaded confidence area. This indicates overdispersion which can lead to overconfident p-values, in case of negative binomial model.

The homogeneity of variance suggests that the variance remains relatively constant across different fitted values. The vertical clustering of data points indicates the model is likely using categorical predictors or a limited range of discrete values to make its predictions.

The influential observations reveals a critical issue such as several data points have extremely high residuals. These are highly influential outliers that are pulling the model estimates away from the true trend, likely causing the dispersion issues seen in other plots.

The distribution of quantile residuals shows that the residuals do not follow the expected uniform distribution, particularly at the tails of the data. This confirms that the model is failing to properly account for extreme values, often a byproduct of the influential outliers identified.

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.

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

	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.

Explanation: All three models show a positive treatment effect, meaning MPAs have more lobsters. However, the effect is only statistically significant in the poisson model (p value), not in the OLS or Negative Binomial models. This means the direction of the effect is consistent, but the statistical support varies across models.

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`)

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)

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

```
ff_counts <- spiny_counts %>%
    mutate(year=as_factor(year))

m5_fixedeffs <- glm.nb(
    counts ~
        treat +
        year +
        treat*year,
    data = ff_counts)

summ(m5_fixedeffs, model.fit = FALSE)
```

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

	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

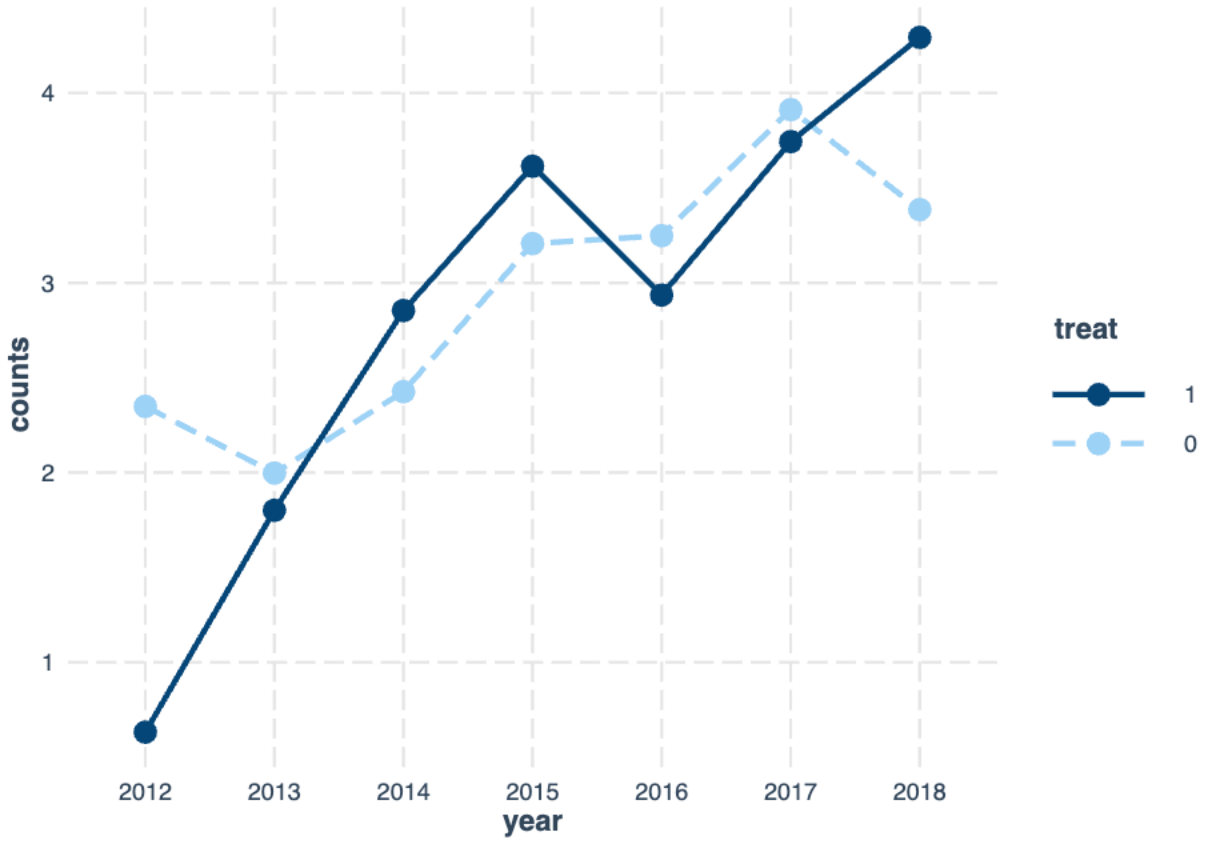
Standard errors: MLE

Explanation: The negative effect for the treatment (-1.72) represents the difference between the MPA reefs and control reefs at the beginning of the study. This indicates that the reefs selected for MPA initially started with significantly fewer lobsters than the control reefs. This makes sense because it establishes the starting point before the MPA protections had time to work. The positive interaction terms (such as treat:year2018) show that while the MPA reefs started lower, they experienced higher growth rates over time compared to the control reefs.

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.

```
interact_plot(m5_fixedeffs, pred = year, modx = treat,
              outcome.scale = "link") # NOTE: y-axis on log-scale
```



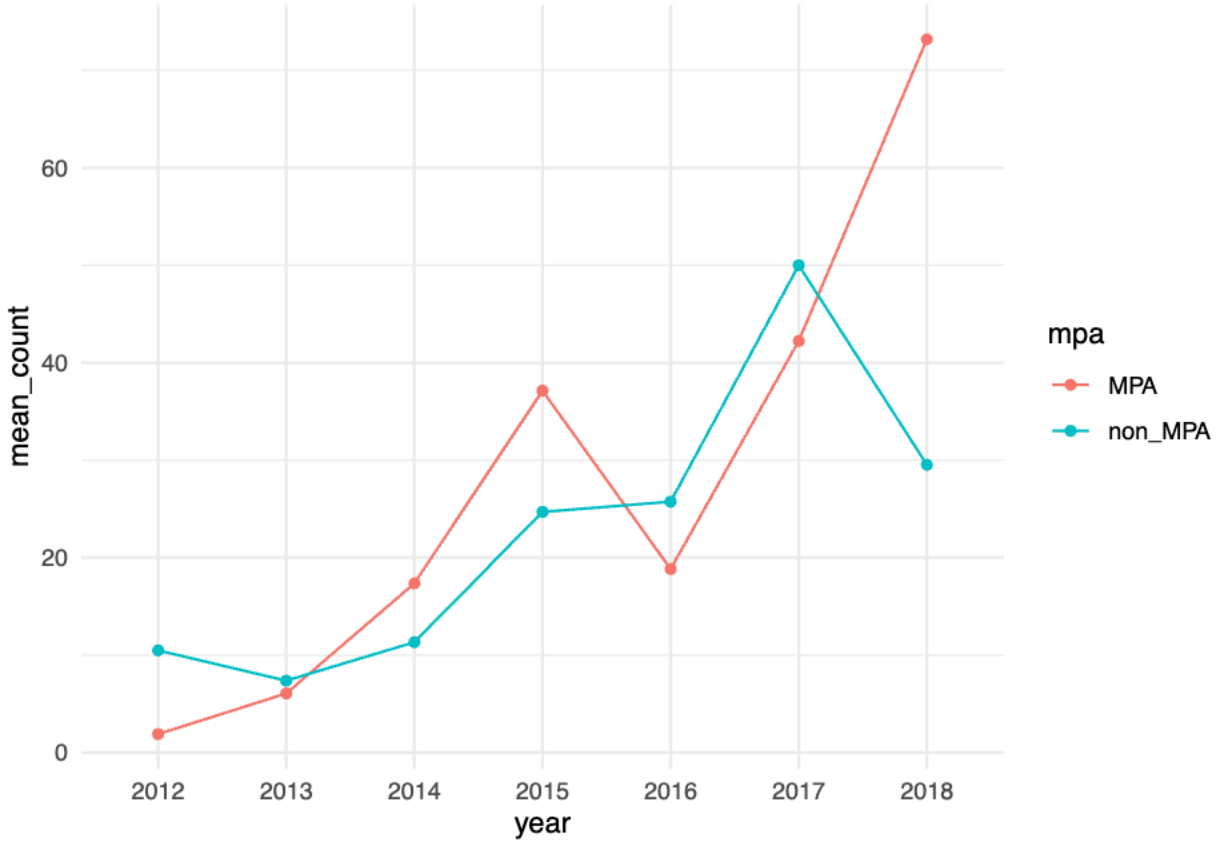

```
# HINT: Change `outcome.scale` to "response" to convert y-axis scale to counts
```

Explanation: After plotting the model predictions, lobster abundance in MPAs increased from 2012 to 2015, declined in 2016, and then again increased through 2018. This pattern may seem unusual, but it likely reflects the external influences rather than a failure of the MPA. Possible explanations include spillover effects such as lobsters move beyond protected boundaries, selection bias from having more control sites than MPA sites, differences in sampling effort, and environmental factors such as changes in ocean currents or local food availability.

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

plot_counts <- ff_counts |>
    group_by(year, mpa) |>
    summarise(mean_count = mean(counts, na.rm = TRUE))

plot_counts %>% ggplot(aes(year, mean_count, color = mpa, group = mpa)) +
    geom_point() +
    geom_line() +
    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)

Answer: Spillover effects are likely in this research context. As lobster populations grow within MPAs, limited space, food availability, or predator pressure can lead individuals to move into nearby non-MPA areas. The predictions plot supports this pattern, showing a decline in MPA counts alongside an increase in non-MPA counts between 2015 and 2016. This suggests movement out of protected areas rather than a loss of population. Later, the decline in non-MPAs may be due to the fishing pressure, while MPAs continue to support population growth until spillover occurs again.

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

Answer: Spillover is an issue for identifying causal effects because it violates the assumption of that the treatment and control groups are independent of each other. If lobsters move from MPAs to non-MPAs, the control sites are indirectly affected by the treatment. This makes differences between MPAs and non-MPAs smaller than the true effect of protection, leading to biased estimates of the causal impact of MPAs.

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

Answer: Spillover relates to impactin this research because it spreads the benefits of MPAs beyond their boundaries. While this indicates a positive ecological outcome, it makes the measured impact within MPAs appear smaller, since some of the benefits are also observed in non-MPA sites rather than only in protected areas.

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

SUTVA means that the treatment of one site should not affect other sites. In this study, it may not hold true because lobsters can move from MPAs to non-MPAs. This movement means that outcomes in control sites can be influenced by the treatment, so SUTVA is not fully met.

2) Exogeneity Assumption

Exogeneity means that the choice of MPA sites is independent of lobster populations. This is also unlikely here because MPAs are usually chosen based on factors like good habitat, ecological importance, or existing lobster abundance. These factors affect outcomes, so the assumption may not be entirely true.

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

```
# Read in updated data
sblobster <- read_csv(here("data", "extracredit_sblobstrs24.csv")) %>%
  clean_names() %>%
  mutate(size_mm = na_if(size_mm, -99999))

# Clean and prepare data for analysis
sblobster_tidy <- sblobster %>%
  mutate(reef = case_when(
    site == "IVEE" ~ "Isla Vista",
    site == "NAPL" ~ "Naples",
    site == "MOHK" ~ "Mohawk",
    site == "CARP" ~ "Carpenteria",
    site == "AQUE" ~ "Arroyo Quemado"
  )) %>%
  mutate(reef = factor(reef,
                       levels = c("Arroyo Quemado", "Carpenteria",
                                  "Mohawk", "Isla Vista", "Naples")))

# Summarize counts by site, year, and transect
sblobster_count <- sblobster_tidy %>%
  group_by(site, year, transect) %>%
```

```
  summarize(counts = sum(count, na.rm = TRUE),
            mean_size = mean(size_mm, na.rm = TRUE),
            .groups = "drop") %>%
  mutate(treat = case_when(site %in% c("IVEE", "NAPL") ~ "MPA",
                           site %in% c("MOHK", "CARP", "AQUE") ~ "non_MPA"),
         treat_binary = case_when(treat == "MPA" ~ 1,
                                  treat == "non_MPA" ~ 0))
```

b. Run at least 3 regression models & assess model diagnostics

```
# Fit a OLS regression model
model_ols <- lm(counts ~ treat, data = sblobster_count)

# Interpret the coefficient
summ(model_ols, model.fit = FALSE)
```

Observations	466
Dependent variable	counts
Type	OLS linear regression

	Est.	S.E.	t val.	p
(Intercept)	34.99	2.84	12.31	0.00
treatnon_MPA	-7.72	3.91	-1.97	0.05

Standard errors: OLS

Interpretation: In OLS model, MPAs have an average of about 35 lobsters. Non-MPA have around 8 fewer lobsters than MPAs. The effect is barely statistically significant, showing some evidence that MPAs have more lobsters, but the result is not very strong.

```
# Fit a poisson regression model
model_pois <- glm(counts ~ treat, data = sblobster_count,
                  family = poisson(link = "log"))

# Interpret the coefficient
summ(model_pois, model.fit = FALSE)
```

Observations	466
Dependent variable	counts
Type	Generalized linear model
Family	poisson
Link	log

	Est.	S.E.	z val.	p
(Intercept)	3.55	0.01	311.89	0.00
treatnon_MPA	-0.25	0.02	-14.92	0.00

Standard errors: MLE

Interpretation: In the poisson model, the intercept of 3.55 means the expected lobster count in MPAs is about (e^{3.55}) ~35 lobsters. The treatnon_MPA coefficient of -0.25 means non-MPA sites have about 22% fewer lobsters than MPAs (e^{-0.25}) ~ 0.78. Both results are statistically significant.

```
# Fit a negative binomial regression model
model_nb <- glm.nb(counts ~ treat, data = sblobster_count)

# Interpret the coefficient
summ(model_nb, model.fit = FALSE)
```

Observations	466
Dependent variable	counts
Type	Generalized linear model
Family	Negative Binomial(0.5769)
Link	log

	Est.	S.E.	z val.	p
(Intercept)	3.55	0.09	39.72	0.00
treatnon_MPA	-0.25	0.12	-2.02	0.04

Standard errors: MLE

Interpretation: In the negative binomial model, MPAs have about 35 lobsters on average. Non-MPA sites have around 22% fewer lobsters. The effect is statistically significant.

c. Compare and contrast results with the analysis from the 2012-2018 data sample (~ 2 paragraphs)

Export summary of the "OLS", "Poisson" and "Negative Binomial" models

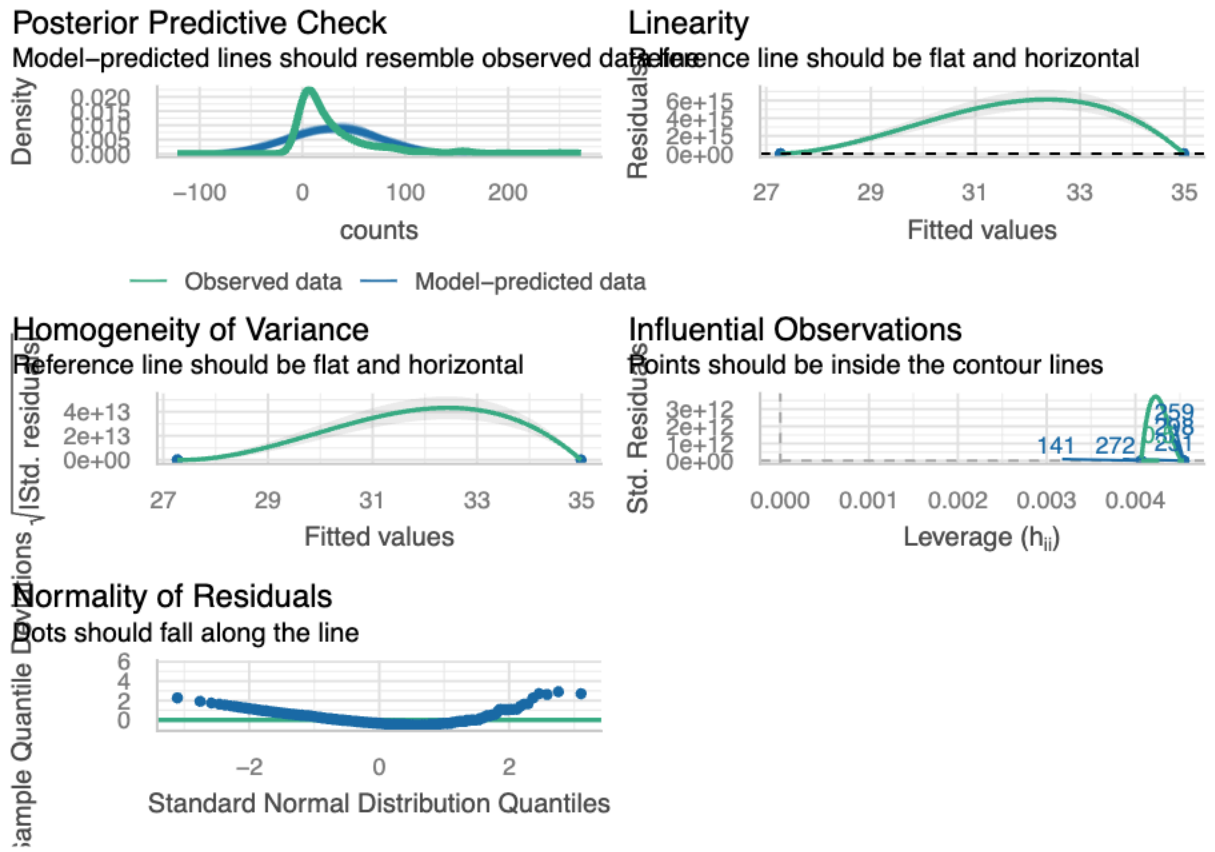
```
# Export summary of the models
export_summs(model_ols, model_pois, model_nb,
             model.names = c("OLS","Poisson", "NB"),
             statistics = "none")
```

	OLS	Poisson	NB
(Intercept)	34.99 ***	3.55 ***	3.55 ***
	(2.84)	(0.01)	(0.09)
treatnon_MPA	-7.72 *	-0.25 ***	-0.25 *
	(3.91)	(0.02)	(0.12)

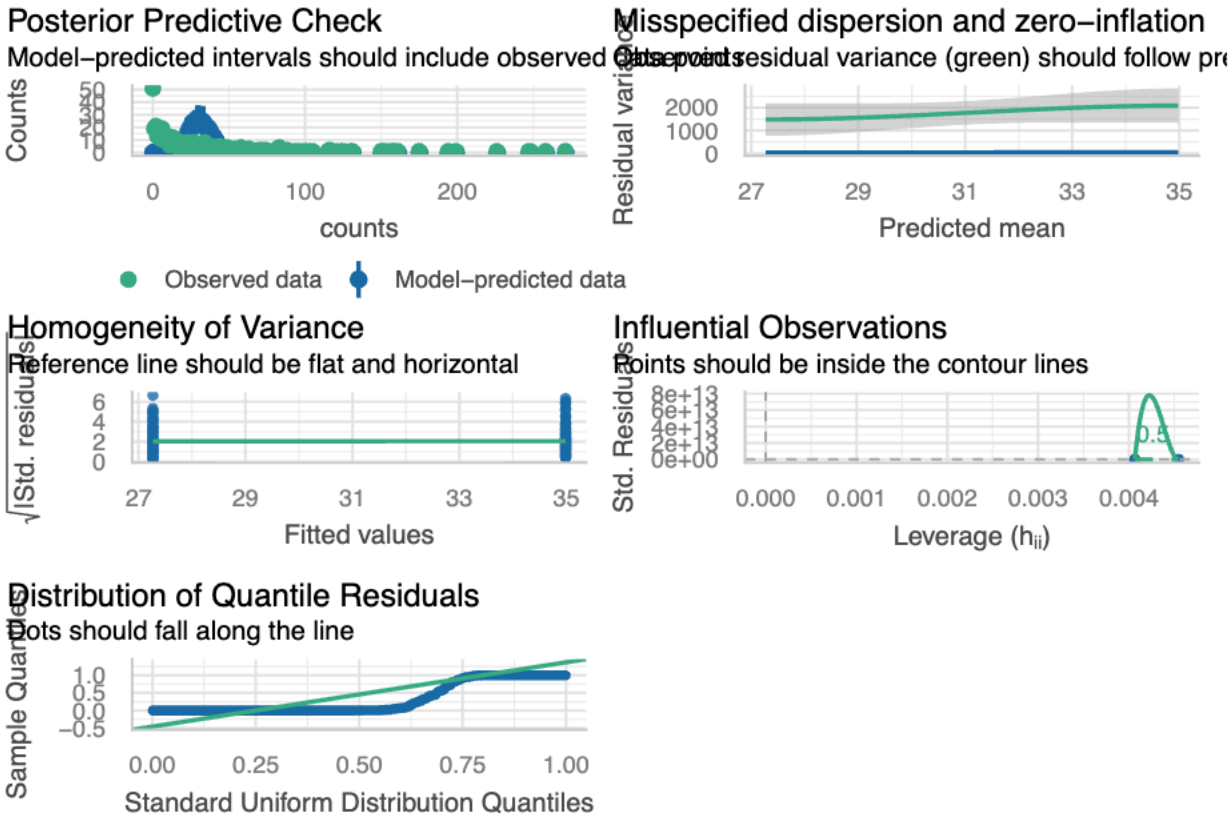
*** p < 0.001; ** p < 0.01; * p < 0.05.

Explanation: All the three models have negative treatment effect. This indicates that non-MPA sites have lower lobster counts than MPAs. The effect is statistically significant in all models, though the level of significance varies. This suggests that both the direction and the evidence for the treatment effect are fairly stable across model specifications.

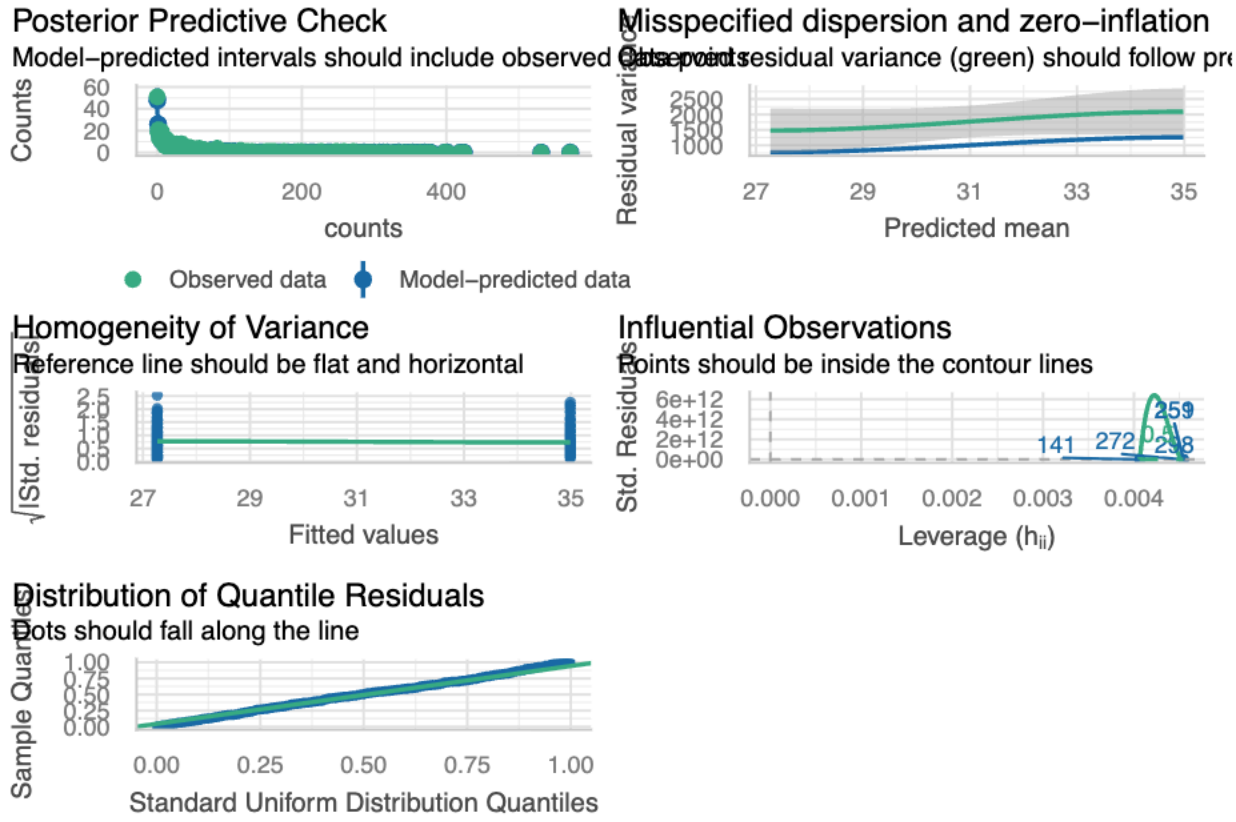
```
# Check model diagnostics
check_model(model_ols)
```




```
check_model(model_pois)
```




```
check_model(model_nb)
```




```
# Check overdispersion
check_overdispersion(model_pois)
```

```
## # Overdispersion test
##
##        dispersion ratio =    57.103
##   Pearson's Chi-Squared = 26496.023
##                 p-value =   < 0.001
```

```
check_overdispersion(model_nb)
```

```
## # Overdispersion test
##
##  dispersion ratio = 1.035
##           p-value = 0.808
```

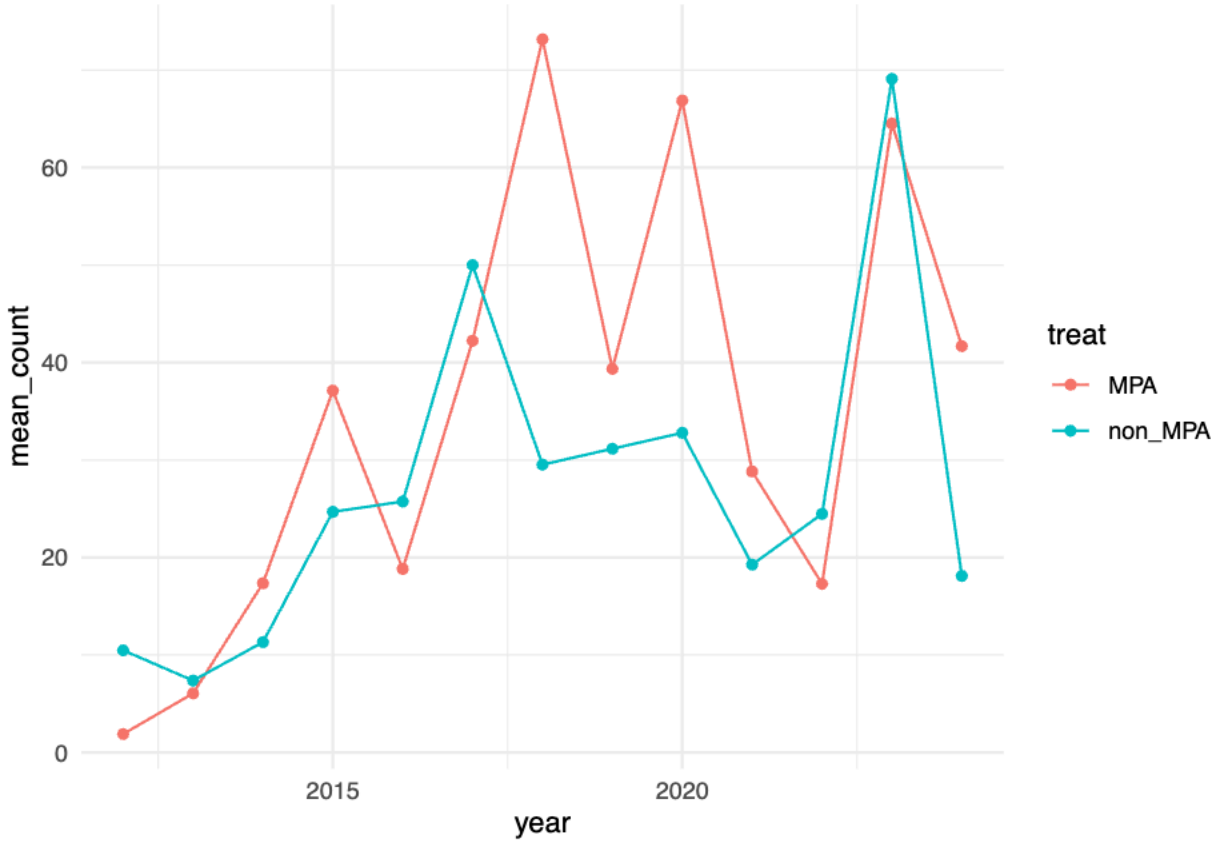
```
# Check zeroinflation
check_zeroinflation(model_pois)
```

```
## # Check for zero-inflation
##
##    Observed zeros: 51
##   Predicted zeros: 0
##             Ratio: 0.00
```

```r
check_zeroinflation(model_nb)
```

```
## # Check for zero-inflation
##
##    Observed zeros: 51
##   Predicted zeros: 47
##             Ratio: 0.91
```

```r
# Plot predictions
sblobster_count %>%
  dplyr:::group_by(year, treat) %>%
  summarise(mean_count = mean(counts, na.rm = TRUE)) |>
  ggplot(aes(year, mean_count, group = treat, color = treat)) +
  geom_point() +
  geom_line() +
  theme_minimal()
```



The OLS model shows MPAs have about 35 lobsters on average, with non-MPAs having ~8 fewer, though the effect is weakly significant. The poisson model predicts ~35 lobsters in MPAs and ~22% fewer in non-MPAs, but it shows severe overdispersion (dispersion ratio = 57.1) and extreme zero-inflation (observed zeros = 51, predicted = 0), indicating poor fit. In contrast, the negative binomial model predicts similar counts (~35 in MPAs, ~22% fewer in non-MPAs) and handles overdispersion (ratio ~1.04) and zero-inflation (predicted zeros = 47, observed = 51) well, making it the most appropriate model.

Here, the 12 year results are similar to 6 year results, showing similar lobster abundance trends across OLS, poisson and negative binomial models. OLS models still shows overdispersion and zero-inflation, while

negative binomial models handle these well. The poisson model is no longer overdispersed and shows little zero-inflation with the added data. Treatment effects are still significant and slightly stronger in the 12 year dataset and model coefficients remains fairly consistent. Short-term changes in MPA and non-MPA lobster trends may reflect natural ecological changes or potential spillover effects.