Problem Set #7

Nicole Wood (in group NJ with Jade Levandofsky and Jessika Viveros)

November 11, 2023

Grade: /33

Overview

In this problem set, you will practice creating visualizations using the `ggplot2` library, including labeling axes, customizing colors, and making these plots presentation-ready. The data you will be working with is a California housing data set based on the 1990 Census, where each observation is a California district.

Part I: Explore data and create simple graphs

1. Load the following packages in the code chunk below: `tidyverse`, `ggplot2`, `scales`, and `RColorBrewer`. Note that `ggplot2` is part of tidyverse, and you do not need to load it separately, but we will do so for this problem set.

/3

2. Use `load()` and `url()` to load the `housing_df` dataframe from `https://github.com/anyone-can-cook/rclass1/raw/`

This dataset was downloaded from Kaggle and contains data on California housing prices. Each observation in the dataset is a California district. Take some time to read about the data and the variables it contains. Kaggle California Housing Prices.

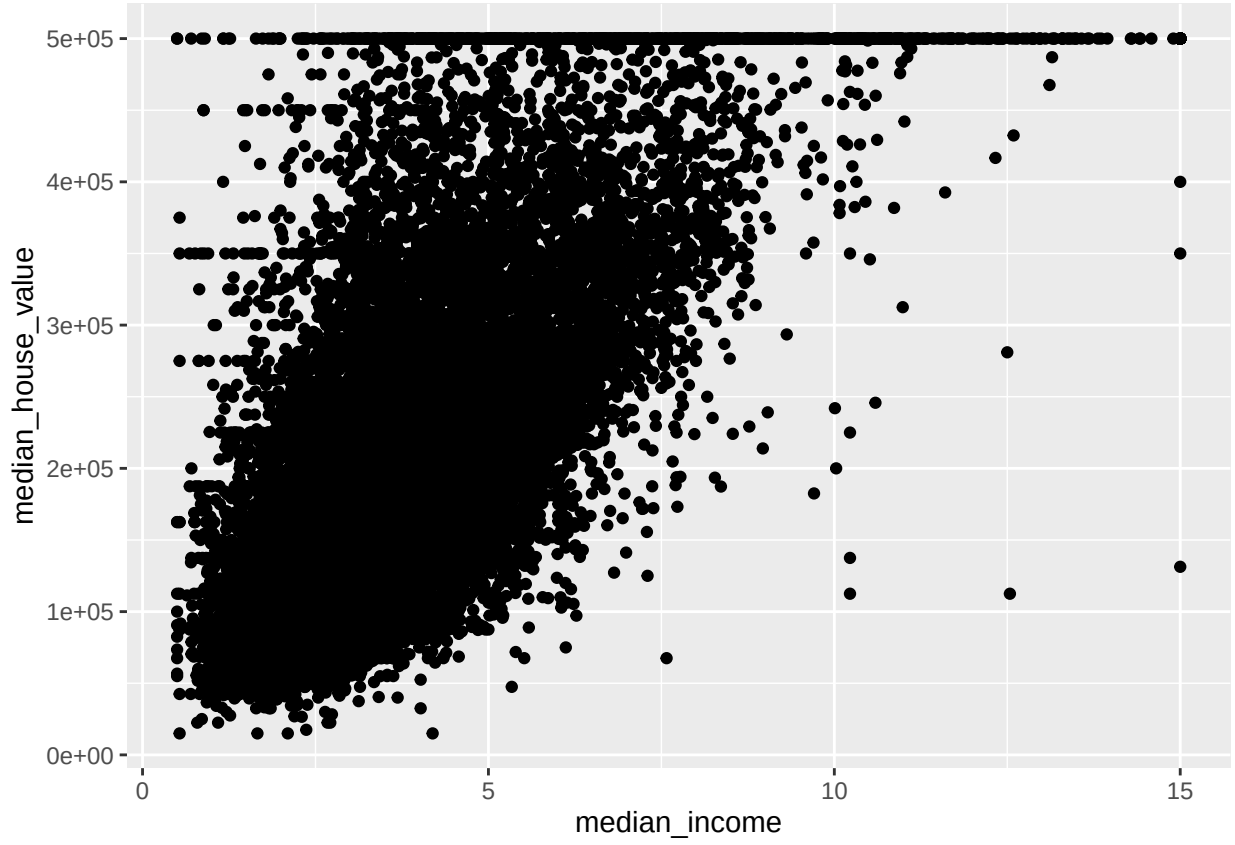
3. Let's investigate the `housing_df` dataframe. First, use `head()` and `glimpse()` to preview the data.
    - How many observations (rows) and variables (columns) are there?
    - **ANSWER**: There are 6 observations and 10 variables.

```
## # A tibble: 6 x 10
##   longitude latitude housing_median_age total_rooms total_bedrooms population
##       <dbl>    <dbl>              <dbl>       <dbl>          <dbl>      <dbl>
## 1     -122.     37.9                 41         880            129        322
## 2     -122.     37.9                 21        7099           1106       2401
## 3     -122.     37.8                 52        1467            190        496
## 4     -122.     37.8                 52        1274            235        558
## 5     -122.     37.8                 52        1627            280        565
## 6     -122.     37.8                 52         919            213        413
## # i 4 more variables: households <dbl>, median_income <dbl>,
## #   median_house_value <dbl>, ocean_proximity <chr>
```

```
## Rows: 20,640
## Columns: 10
## $ longitude          <dbl> -122.23, -122.22, -122.24, -122.25, -122.25, -122.2~
## $ latitude           <dbl> 37.88, 37.86, 37.85, 37.85, 37.85, 37.85, 37.84, 37~
## $ housing_median_age <dbl> 41, 21, 52, 52, 52, 52, 52, 52, 42, 52, 52, 52, 52,~
## $ total_rooms        <dbl> 880, 7099, 1467, 1274, 1627, 919, 2535, 3104, 2555,~
## $ total_bedrooms     <dbl> 129, 1106, 190, 235, 280, 213, 489, 687, 665, 707, ~
## $ population         <dbl> 322, 2401, 496, 558, 565, 413, 1094, 1157, 1206, 15~
## $ households         <dbl> 126, 1138, 177, 219, 259, 193, 514, 647, 595, 714, ~
## $ median_income      <dbl> 8.3252, 8.3014, 7.2574, 5.6431, 3.8462, 4.0368, 3.6~
## $ median_house_value <dbl> 452600, 358500, 352100, 341300, 342200, 269700, 299~
## $ ocean_proximity    <chr> "NEAR BAY", "NEAR BAY", "NEAR BAY", "NEAR BAY", "NE~
```

/2

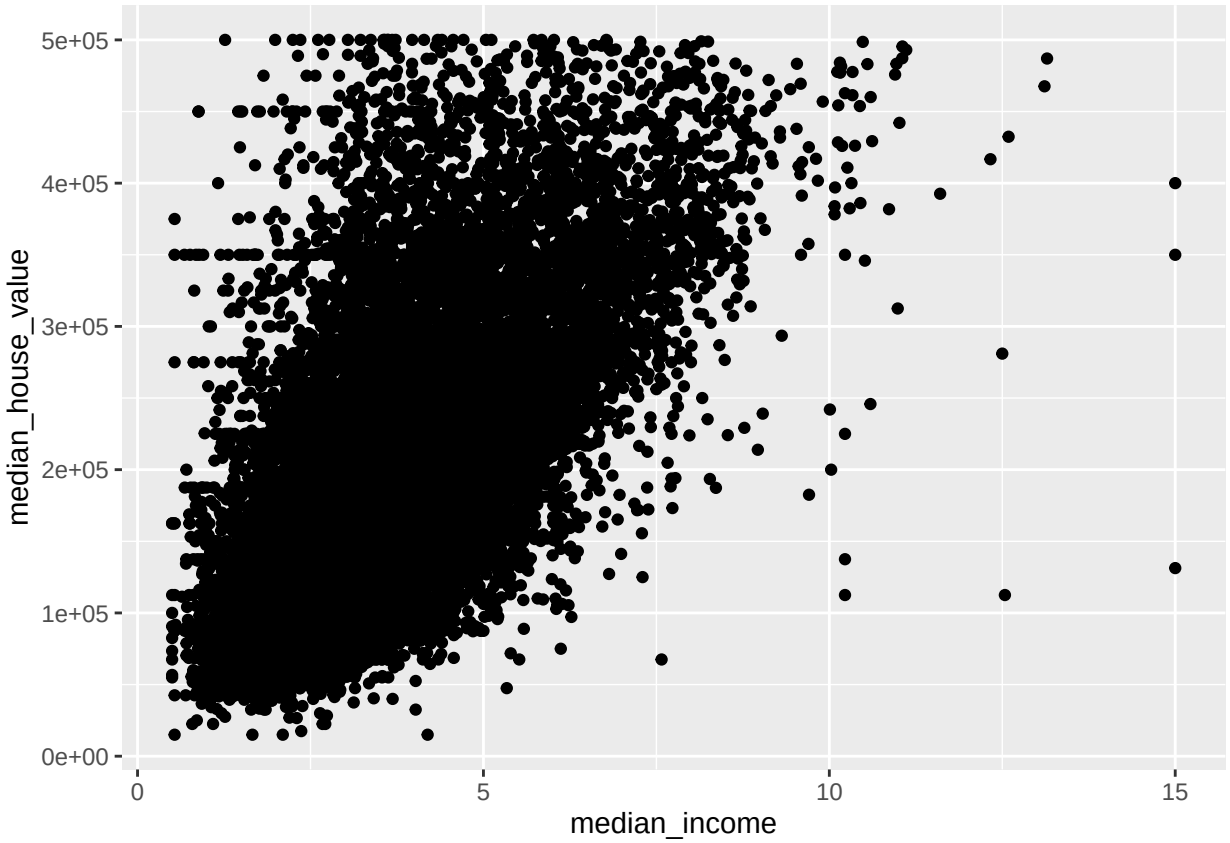
4. Use `ggplot` to create a simple scatterplot showing `median_income` on the x-axis and `median_house_value` on the y-axis.



/1

5. In the plot, you should notice there is a line of points spread out along `y = 500000`. (You may also notice the number on the axis being displayed as `5e+05` instead of `500000` – do not worry about this for now.) If you inspect `housing_df`, you'll see there are many points with `median_house_value` of `500001`, which suggests that observations containing `median_house_value` above `500000` may not be reliable.

- Filter the dataframe to only contain observations with `median_house_value` of `500000` or less, and resassign this back to the `housing_df` dataframe. If you try running your code from Question 4 again, you should see that the line of points is gone.



/2

6. Next, take some time to investigate the `ocean_proximity` variable in the dataframe (e.g., variable type, class, descriptive stats like count). You may comment out these lines of code after you're done.

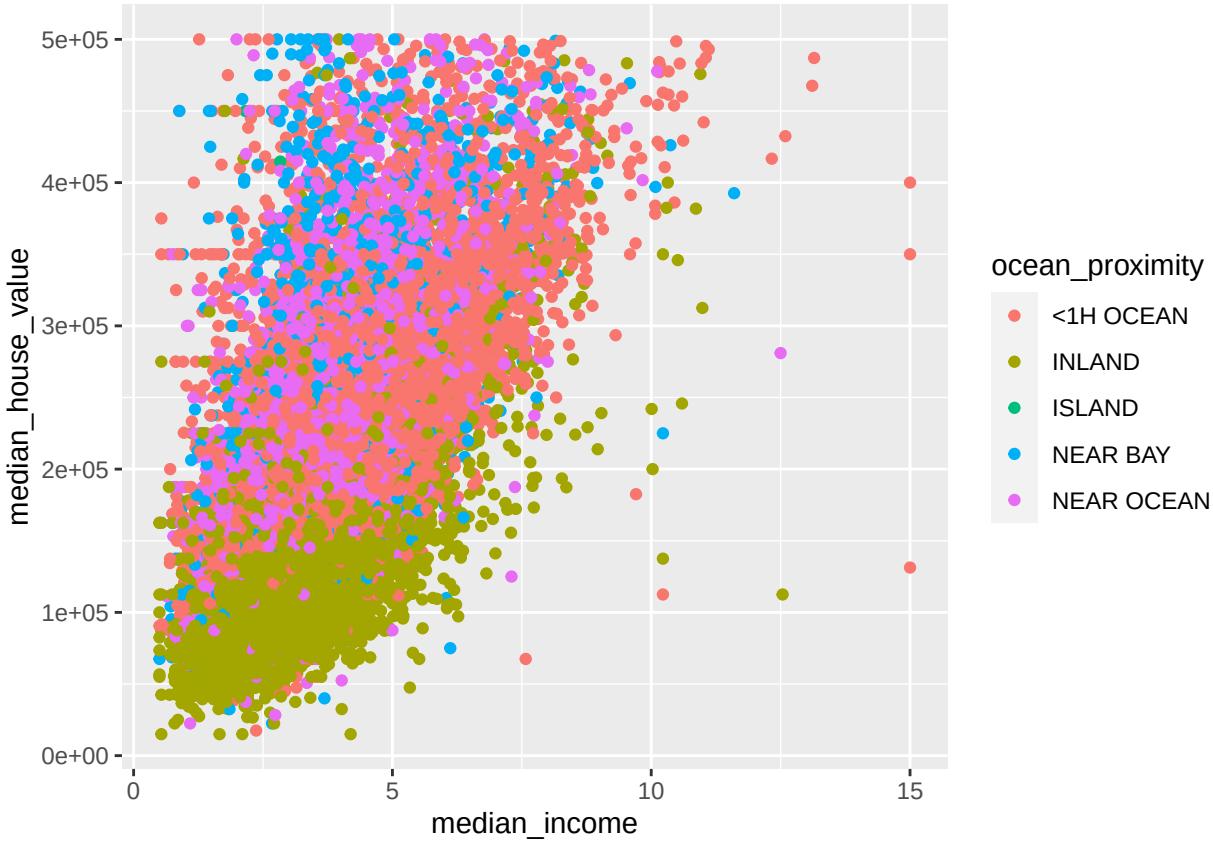
   Then, copy your code from Question 4 and update it to have the points from the scatterplot be colored by `ocean_proximity`.

```
## [1] "character"

##  chr [1:19675] "NEAR BAY" "NEAR BAY" "NEAR BAY" "NEAR BAY" "NEAR BAY" ...

## [1] 19675

## # A tibble: 5 x 2
## # Groups:   ocean_proximity [5]
##   ocean_proximity     n
##   <chr>           <int>
## 1 <1H OCEAN        8604
## 2 INLAND           6524
## 3 ISLAND              5
## 4 NEAR BAY         2096
## 5 NEAR OCEAN       2446
```



/2

7. In the legend, you should notice that the categorical values for `ocean_proximity` are ordered alphabetically by default. This might not be the most logical ordering, as it would make more sense to arrange them by location.

   Convert the `ocean_proximity` column into a factor with the levels in this order: 'ISLAND', 'NEAR BAY', 'NEAR OCEAN', '<1H OCEAN', 'INLAND'. (*Hint*: Use the `factor()` function to specify the levels and make sure to reassign it back to the original dataframe to retain the changes). Refer to section `1.2 Type/class of variables ggplot expect` from the lecture for an example on changing a variable to a factor with levels.

   If you try running your code from Question 6 again, you should see the legend values in the updated order.

```
##  chr [1:19675] "NEAR BAY" "NEAR BAY" "NEAR BAY" "NEAR BAY" "NEAR BAY" ...
```

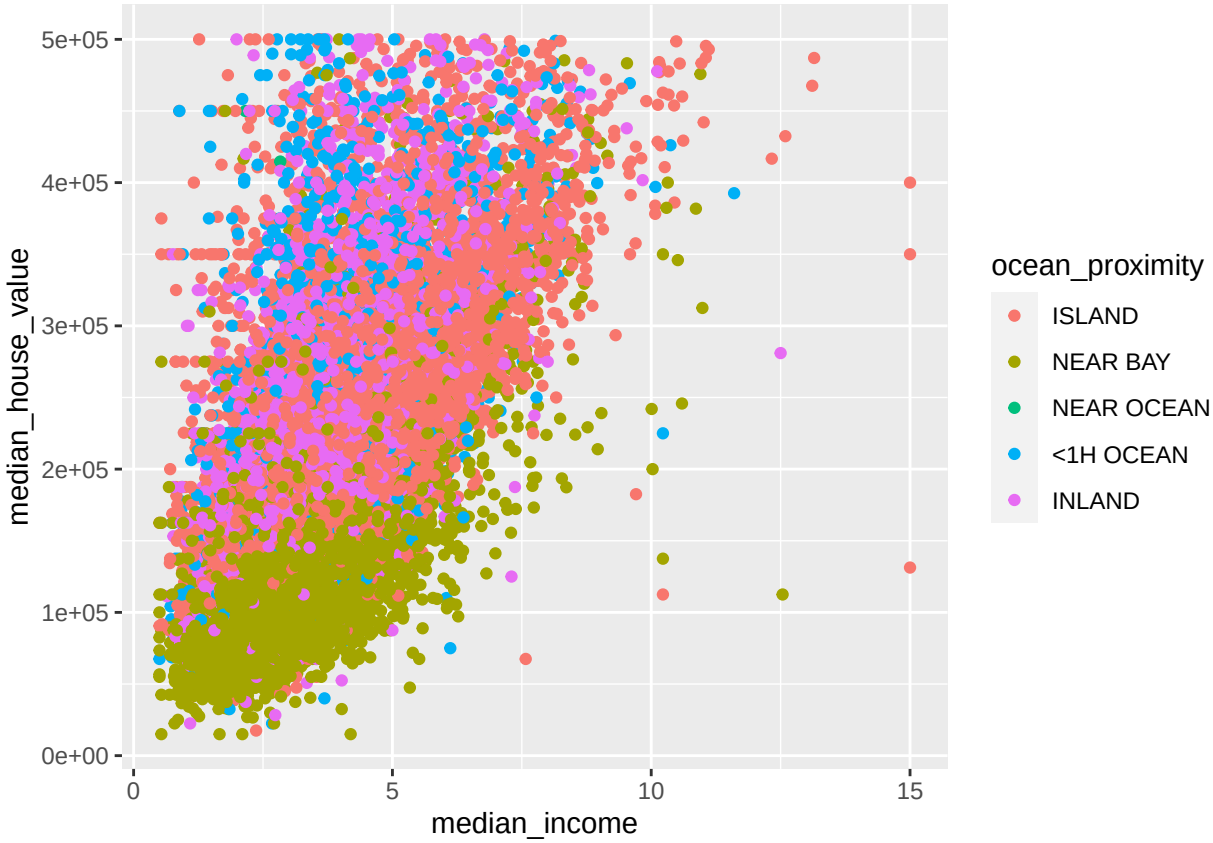
```
## [1] FALSE
```

```
## [1] TRUE
```

```
##  Factor w/ 5 levels "<1H OCEAN","INLAND",..: 4 4 4 4 4 4 4 4 4 4 ...
```

```
## [1] "<1H OCEAN"  "INLAND"     "ISLAND"     "NEAR BAY"   "NEAR OCEAN"
```

```
## [1] "ISLAND"     "NEAR BAY"   "NEAR OCEAN" "<1H OCEAN"  "INLAND"
```



Part II: Colors in ggplot2

Hue, Chroma, and Luminance are important color concepts, and are helpful specifically when working with colors in `ggplot2`.

/2

1. In your own words, define Hue, Chroma, and Luminance and use these concepts to describe the `RColorBrewer` "Oranges" palette.

   - **ANSWER**: Hue refers to the three primary colors (red, yellow, blue) and the three secondary colors (orange, green, purple); it does not include black, gray, or white. Chroma refers to the purity of a color; adding black, gray, or white to a hue reduces both its purity and chroma. Luminance is a measure of percieved brightness. The hue of the RColorBrewer pallete is orange, and it has varying levels of luminance ranging from light to dark. According to the Hue-Chroma-Luminance color scheme on the ggplot lecture notes, the pallete does not appear to show variations in chroma.

/3

2. There are three color palette scales to choose from in the `RColorBrewer` library: sequential, qualitative, and diverging, best suited for representing different types of data (e.g., categorical, ordered). Run the code below to graph the relationship between ocean proximity and population by California district.

   Comment out the code below and add a color palette from `RColorBrewer` that is best suited for the data. Also, remove the `eval=FALSE` from the code chunk.

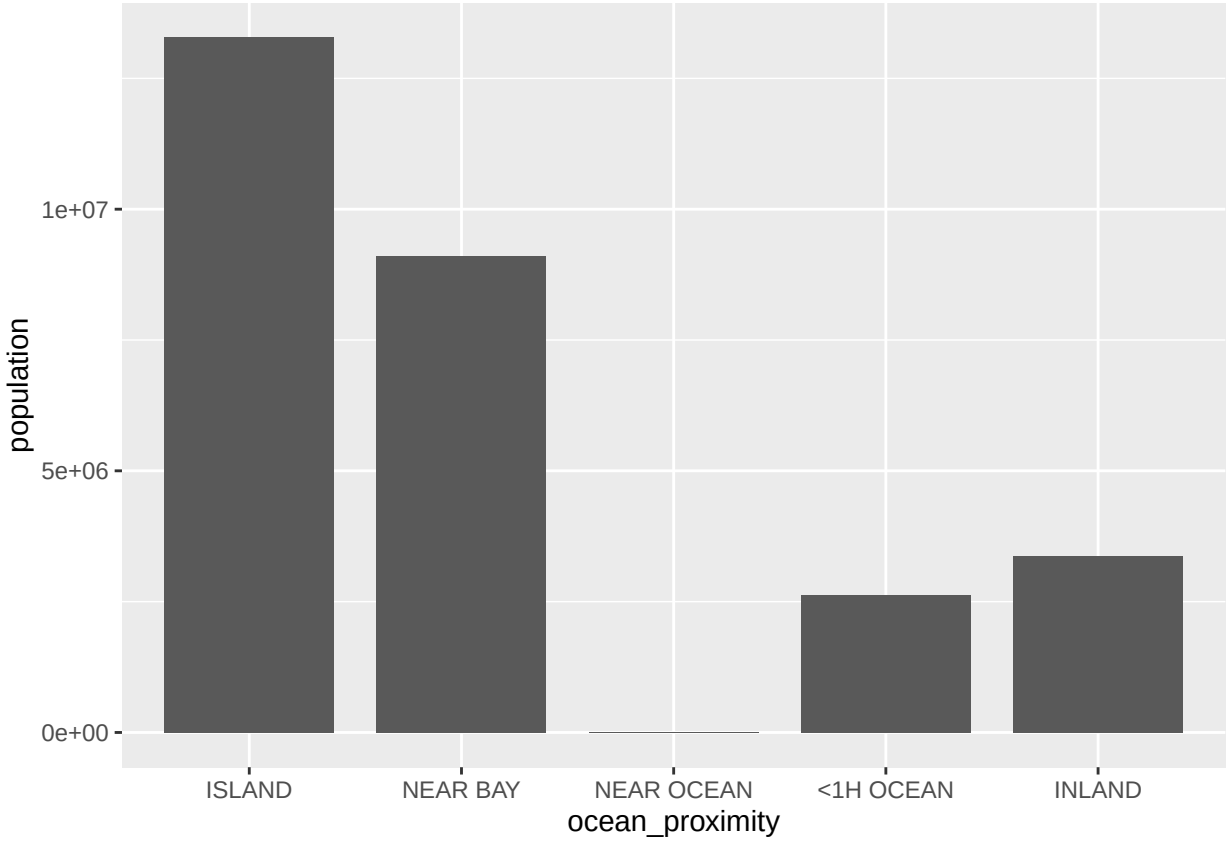
- Explain why you chose this palette and why other palettes might not work as well.

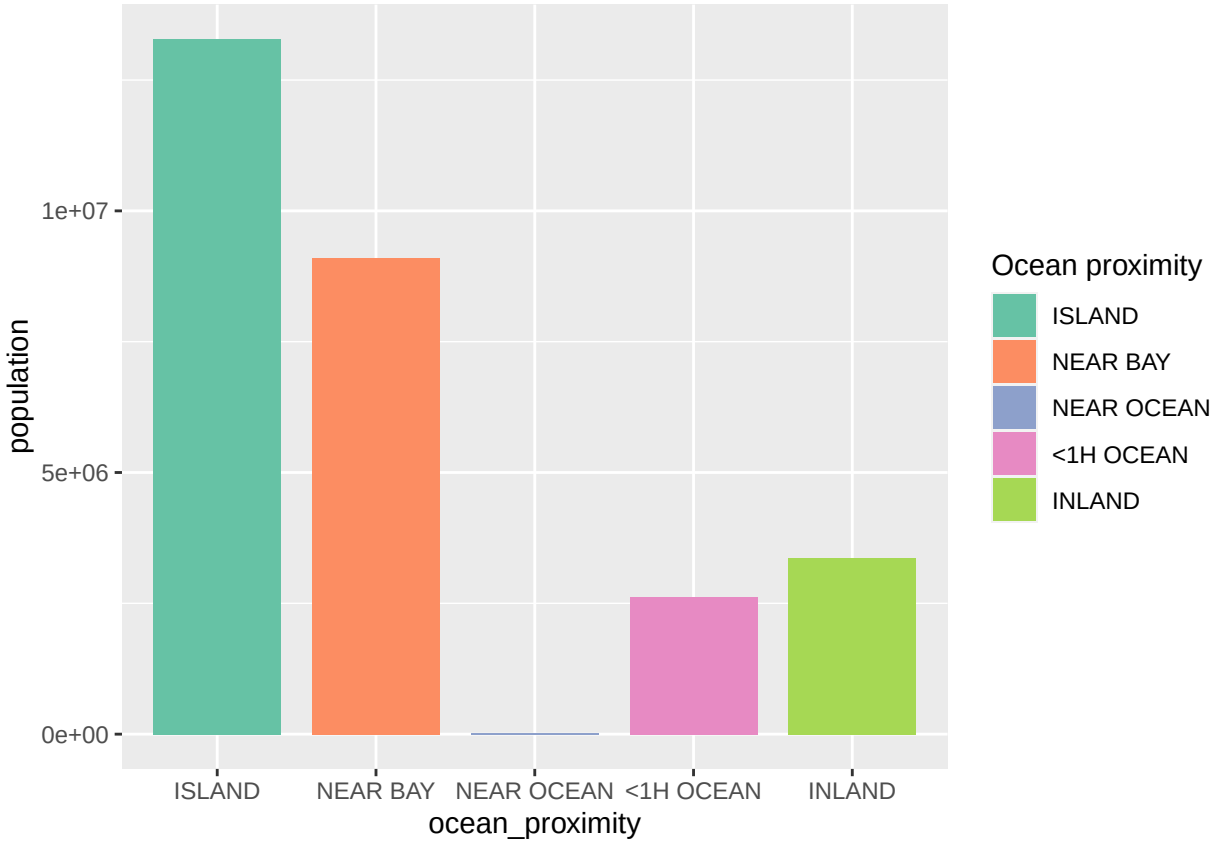
ANSWER: Because the ocean_proximity variables are categorical, the qualitative palettes are best suited to represent the data. Sequential and diverging are less suited, because there is no particular ordering to the categories.

```
## [1] 3
```

```
## [1] 35682
```

```
## [1] 1440.812
```



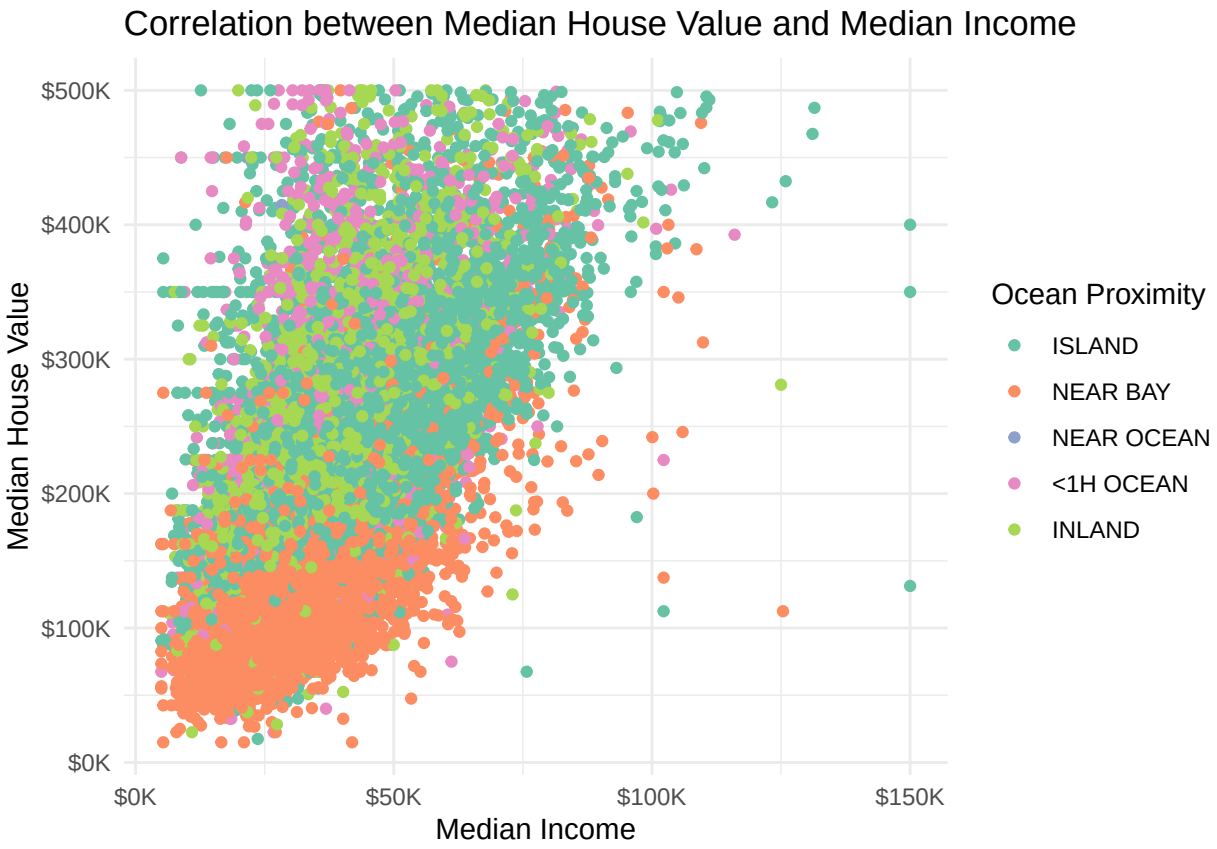


Part III: Creating and customizing graphs

/4

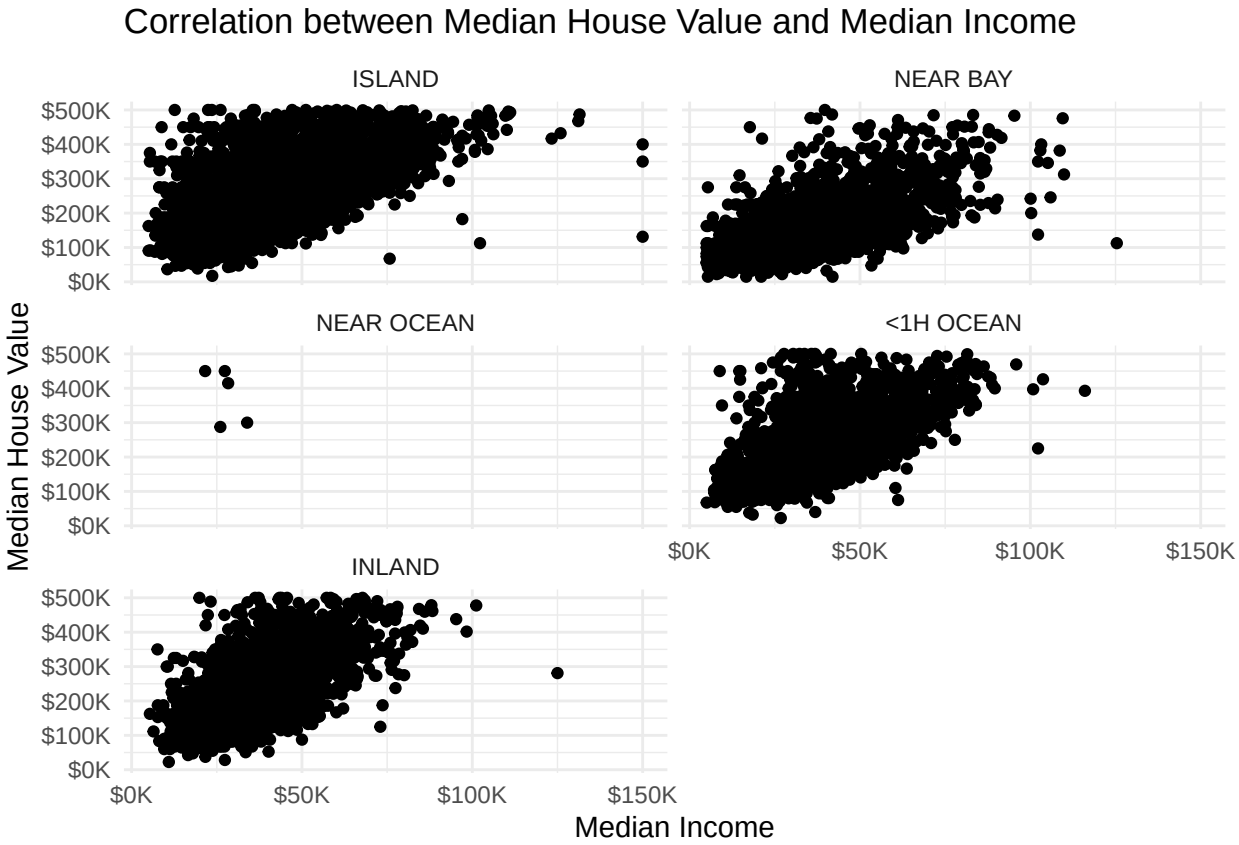
1. Building from Question 7 in Part I, add the following to customize your plot:

   - Use `ggtitle()` to give the plot a title
   - Use `xlab()` and `ylab()` to label the axes
   - Use `scale_color_brewer()` to set the color palette and legend title
   - Use `scale_x_continuous()` and `scale_y_continuous()`, along with the `label_number()` function from the `scales` library, to customize the scale display so they display the dollars in hundreds of thousands (e.g., `$100K`, `$200K`, etc.)
     - *Hint*: According to the variable descriptions here, the median income is reported in tens of thousands – make sure to display this accordingly
   - Use `theme_minimal()` or a custom theme to add further customizations to your plot



/3

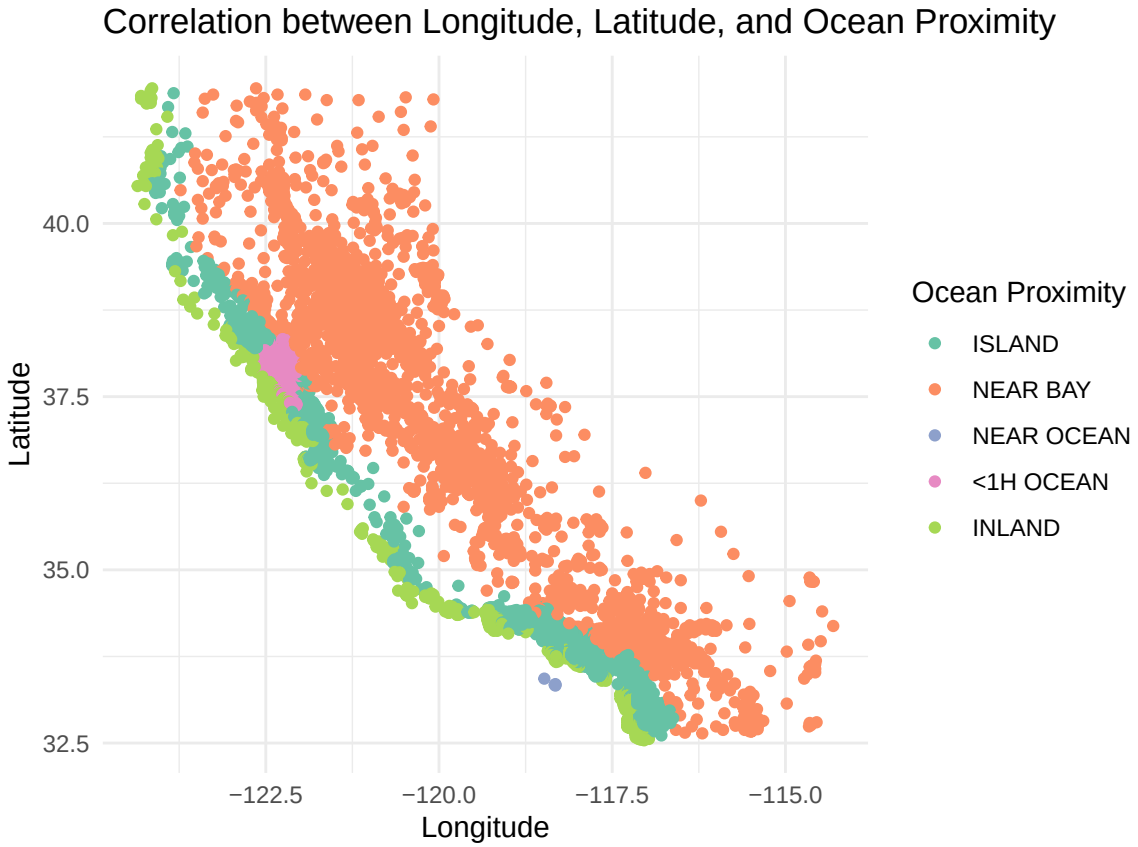
2. Create a second graph, the same scatterplot as the previous question showing `median_income` on the x-axis and `median_house_value` on the y-axis, but with separate subplots (i.e., small multiple) for each value of `ocean_proximity`. Make sure to remove the `color` aesthetic from the previous scatterplot.



/3

3. Create a third graph, a scatterplot showing `longitude` on the x-axis and `latitude` on the y-axis, with the points colored by `ocean_proximity`. Make sure to include the following:

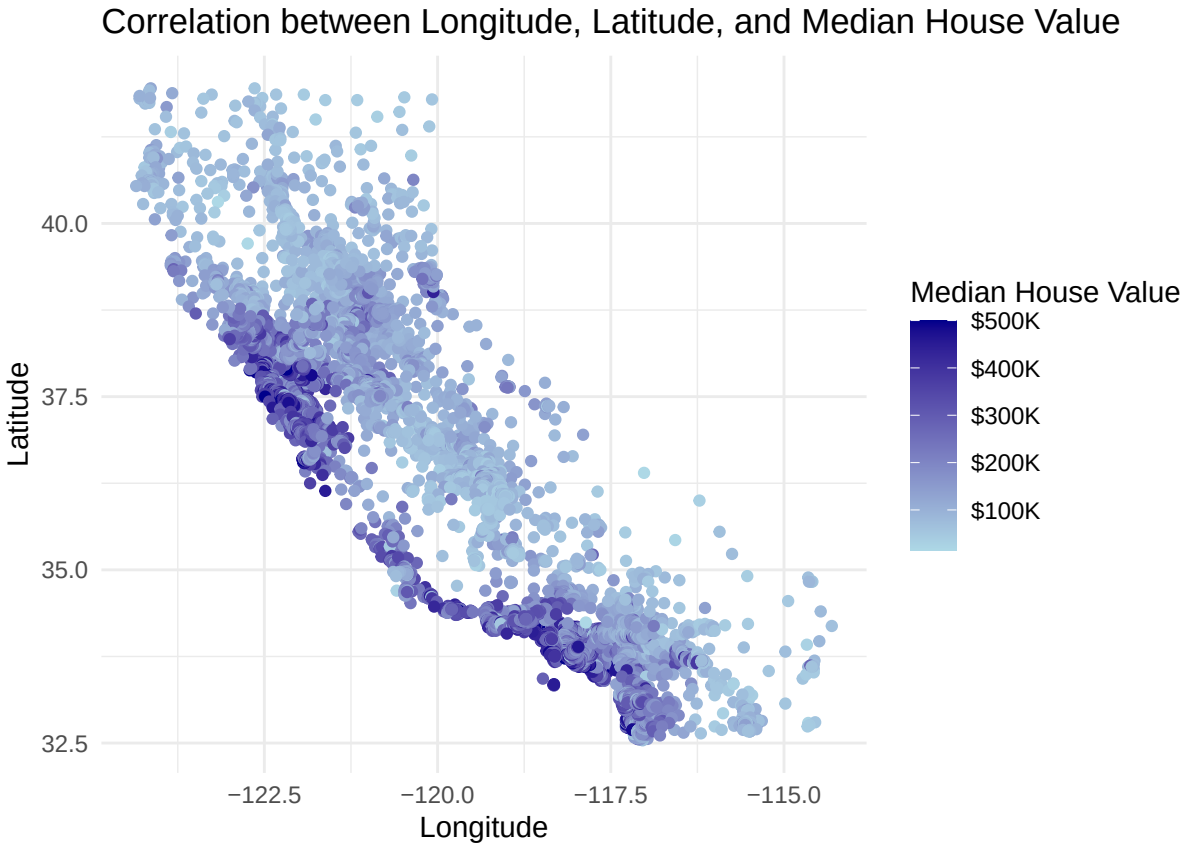
    - Plot title
    - Appropriate axis labels
    - Legend with an approriate title and your choice of color palette
    - Use `coord_fixed()` to fix the coordinate scaling
    - Any other theme or style customizations



/3

4. Create a fourth graph, a scatterplot showing `longitude` on the x-axis and `latitude` on the y-axis, with the points colored by `median_house_value`. Make sure to include the following:

   - Plot title
   - Appropriate axis labels
   - Legend with an approriate title, value labels, and your choice of color palette
     - *Hint*: Use `scale_color_gradient()` along with the `label_number()` to customize the gradient scale display
   - Use `coord_fixed()` to fix the coordinate scaling
   - Any other theme or style customizations

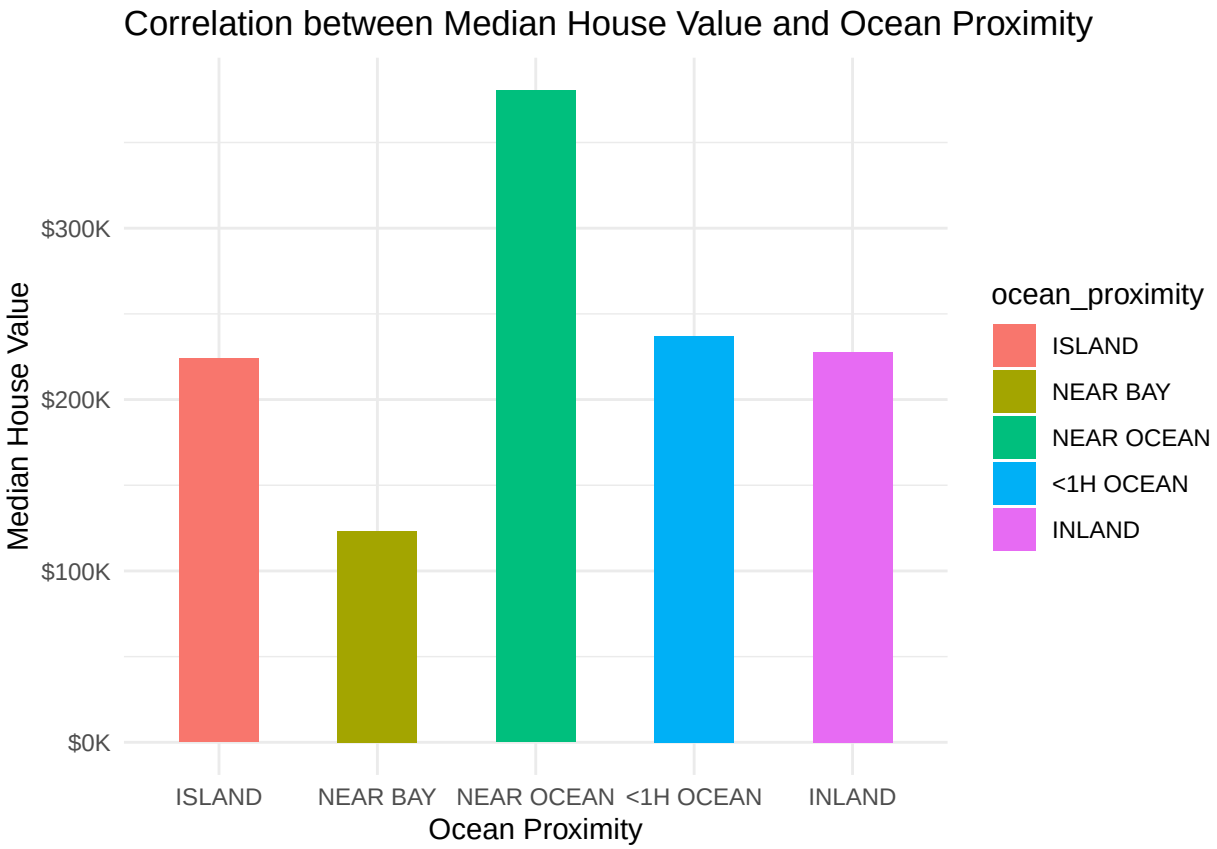


/3

5. Create a barplot showing the average house value by `ocean_proximity`, with each category of `ocean_proximity` along the x-axis and the house value on the y-axis.

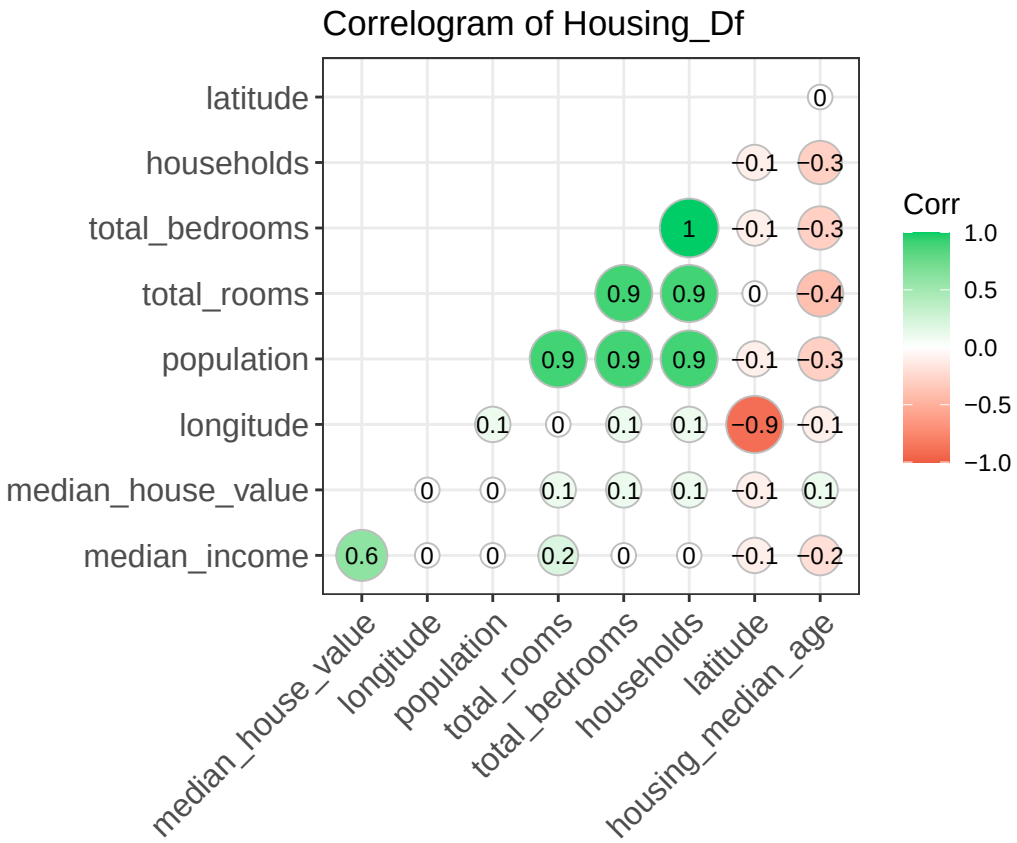
   First group `housing_df` by `ocean_proximity` and calculate the average house value per group, then create the graph with the following features:

   - Plot title
   - Appropriate axis labels
   - Appropriate scale display on axis (e.g., `$100K`, `$200K`, etc.)
   - Any other theme or style customizations (e.g., bar width)



Part IV: Bonus (up to 10% extra credit)

1. Create a graph using any variables you'd like from the housing dataset. However, as you've done in the previous questions, this cannot be a scatterplot or barplot you have already done. Make sure to title and label the plot appropriately and customize it how you'd like. Add a color palette from `Rcolorbrewer` or curate your own color palette. Then, write some text describing your findings or observations.



ANSWER: This correlogram reveals expected results. There is a strong correlation between the number of households and number of rooms, number of bedrooms, and population. There is a weak correlation between the median age of houses and the population, number of households, total rooms, and median_income.

Create a GitHub issue

/2

- Go to the class repository and create a new issue.
- Refer to rclass1 student issues readme for instructions on how to post questions or reflections.
- You are also required to respond to at least one issue posted by another student.
- Paste the url to your issue here: https://github.com/anyone-can-cook/rclass1_student_issues_f23/issues/832
- Paste the url to the issue you responded to here: https://github.com/anyone-can-cook/rclass1_student_issues_f23/issues/830

Knit to pdf and submit problem set

Knit to pdf by clicking the "Knit" button near the top of your RStudio window (icon with blue yarn ball) or drop-down and select "Knit to PDF."

- Go to the class website and under the "Readings & Assignments" » "Week 7" tab, click on the "Problem set 7 submission link"
- Submit both .Rmd and pdf files
- Use this naming convention "lastname_firstname_ps#" for your .Rmd and pdf files (e.g. jaquette_ozan_ps7.Rmd & jaquette_ozan_ps7.pdf)