Mini Project #1: ALLERGENS IN RECIPES

Elijah Russell

10/12/25

Development Process

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.1.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Linking to ImageMagick 6.9.10.23
## Enabled features: fontconfig, freetype, fftw, lcms, pango, webp, x11
## Disabled features: cairo, ghostscript, heic, raw, rsvg
##
## Using 16 threads
```

Initial Exploration

To start I loaded the recommended libraries and additional columns, as well as some of my own. I then went to explore the dataset. My first thought was that it would be interesting to explore how other variables related to cuisine type, but that due to the high percentage of empty cells the resulting analysis would not be reflective of the entire dataset. With the new AI tools it might have been possible to have an AI assign a cuisine to each recipe, but I felt that that did not fit within the scope of this project.

Initial Testing:

To confirm that there were too many empty cells in the cuisine column, I ran a quick summarise on it.

```
all_recipes %>%
  summarise(Num_empty = sum(is.na(cuisine)),
            Num_full = sum(!is.na(cuisine)),
            Empty_ratio = sum(is.na(cuisine))/n())
```

```
## # A tibble: 1 x 3
##   Num_empty Num_full Empty_ratio
##       <int>    <int>       <dbl>
## 1     13375     2218       0.858
```

Initial Testing Results:

According to the results of the summarise function, about 86% of the cells are empty, which is way too many for any analysis to be reflective of the dataset.

Subsequent Explortation

My next thought was inspired by the recommended additional column "contains_eggs". I could create similar additional T/F columns for if a recipe contained some other common allergens. I could then create a plot to compare the amounts of recipes that contain or don't contain different combinations of common allergens. So I went ahead and set up those columns.

```
all_recipes <- all_recipes_raw %>%
  mutate( Dairy = str_detect(ingredients, "milk|butter|cheese|cream|chocolate"),
          Nuts = str_detect(ingredients,
                   "nut|almond|cashew|pecan|pistachio|marzipan|praline|macadamia"),
          Gluten = str_detect(ingredients, "bread|wheat|pasta|flour|cracker|beer"),
          Fish = str_detect(ingredients, "fish|crab|lobster|clam|oyster|shrimp|scallop"),
          Eggs = str_detect(ingredients," egg|custard|mayo"))

all_recipes <- all_recipes %>%
  mutate(across(c(Dairy, Nuts, Gluten, Fish, Eggs), ~replace_na(., FALSE)))
```

Column Setup:

To determine what common allergens besides egg to include I looked up common food allergens and their related ingredients. I then chose the categories of dairy, nuts (treenuts and peanuts combined), gluten, and fish (fish and shellfish combined). When deciding what keywords to use in the string detect function, I chose what I believed to be the most common allergen ingredients as well as any other allergens I noticed when observing the top few rows of the ingredients column.

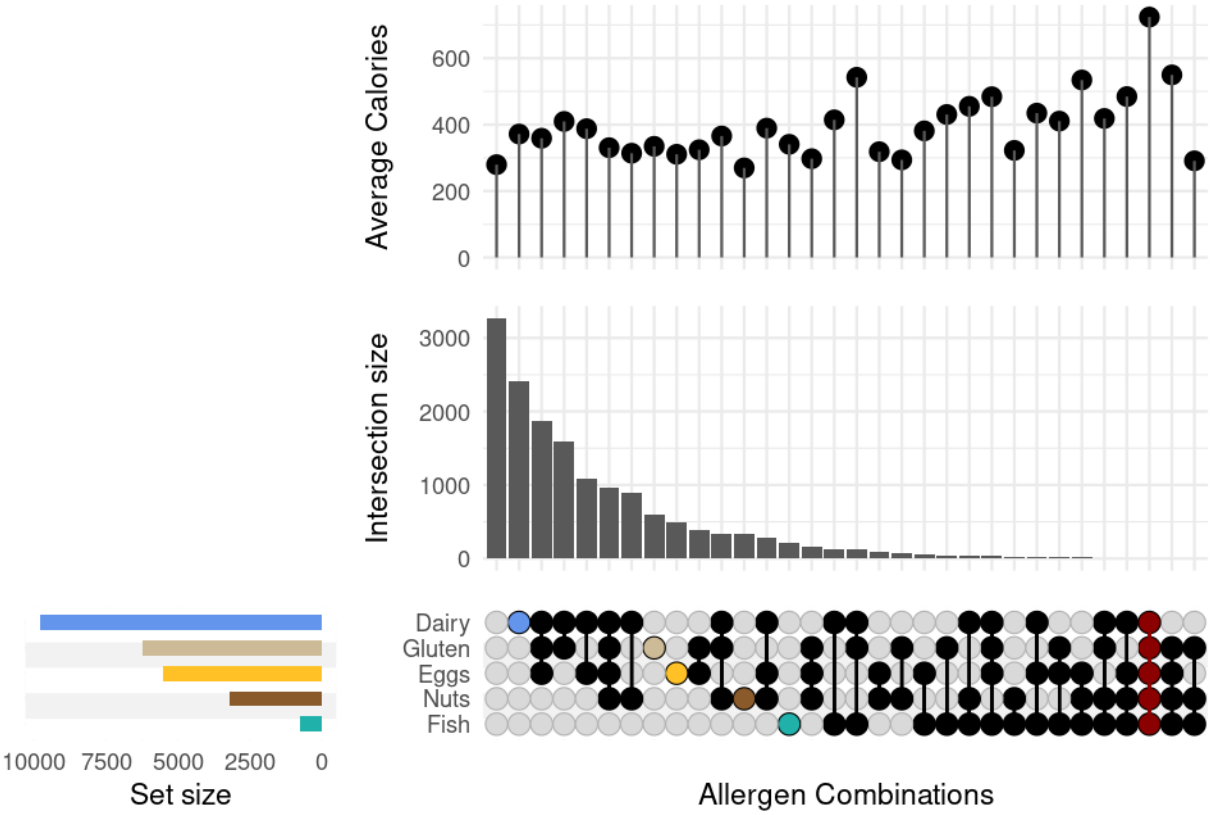
Plot Process

After deciding that I wanted to represent a comparison of the amounts of recipes that contain different combinations of common food allergens, I started to research what the best way to visualise that would be. My initial thought was some sort of venn diagram with size scaled to count of the allergen combination, but I quickly realised that that wouldn't work. I eventually came to the conclusion that an UpSet plot would be the best way to visualize this story, and lookedup the best UpSet plot library for R and added it to the beginning code. Next, I started about construction the code for the plot.

```
allergens = c('Dairy','Nuts','Gluten','Fish','Eggs')

upset(
  all_recipes,
  allergens,
  annotations = list(
    'Average Calories' = ggplot(mapping = aes(x = intersection, y = calories)) +
  stat_summary(fun = mean, geom = "point", size = 3, na.rm = TRUE) +
  stat_summary(fun = mean, geom = "segment", aes(xend = intersection, yend = 0),
               color = "grey35", na.rm = TRUE) +
  labs(y = "Average Calories") + theme(legend.position = "none")
    ),
   base_annotations = list(
    'Intersection size' = (
      intersection_size(
        counts = FALSE) )
  ),
  queries=list(
        upset_query(
            set='Dairy',
```

```
            fill='cornflowerblue'
        ),
        upset_query(
            set='Gluten',
            fill='wheat3'
        ),
        upset_query(
            set='Eggs',
            fill='goldenrod1'
        ),
        upset_query(
            set='Fish',
            fill='lightseagreen'
        ),
        upset_query(
            set='Nuts',
            fill='tan4'
        ),
        upset_query(
            intersect=c('Dairy'),
            color='cornflowerblue',
            only_components=c('intersections_matrix')
        ),
        upset_query(
            intersect=c('Gluten'),
            color='wheat3',
            only_components=c('intersections_matrix')
        ),
        upset_query(
            intersect=c('Eggs'),
            color='goldenrod1',
            only_components=c('intersections_matrix')
        ),
        upset_query(
            intersect=c('Fish'),
            color='lightseagreen',
            only_components=c('intersections_matrix')
        ),
        upset_query(
            intersect=c('Nuts'),
            color='tan4',
            only_components=c('intersections_matrix')
        ),
        upset_query(
          intersect=c('Dairy','Gluten','Eggs','Fish','Nuts'),
            color='darkred',
            only_components=c('intersections_matrix')
        )
    ),
    ) +
  labs(x = "Allergen Combinations")
```



Visualization Evaluation

After a lot of trial and error this is the plot I arrived at. You can see what allergen combinations are the most common, and the combinations with either just one or all of the allergens are highlighted. I chose the colors for each allergen based on color associations to foods related to each allergen, and for the combination of all allergens I used dark red as a warning color.

From this visualization it can be seen that it is more likely for a recipe to have no allergens than any one combination of allergens, but more likely for a recipe to have any allergen combination than no allergens. It can also be seen that Diary is the most common allergen, followed by gluten, eggs, nuts, and fish respectively.

I also chose to add a graph comparing average calories across allergen combinations. Interestingly there is not much difference across allergen combinations, with the exception of recipes with every allergen having the most calories.

Conclusion:

Overall, I feel that this visualization does a good job at telling the story I wanted to see from the data. It highlights just how much and how frequently people with food allergies have to worry about double and triple checking their food for allergens. It also shows that by avoiding recipes that contain their allergens, people with food allergies are generally not losing out calorically.