Chiang_review

2025-03-21

Peer Review of `Viraaj_R_AssignmentUpdated.Rmd` by Lillian Chiang

Hi Vishnu! Overall, I think you did a great job on this assignment. For this review, I will discuss each section of the assignment (data inspection, data processing, and data visualization) and provide feedback.

Data Inspection

Both data sets (`fang` & `snp`) were described well with the code used. I believe all the information you gathered (data set dimensions, column names, variable types, etc.) are important for your data analysis and would be sufficient to complete the assignment goals.

One suggestion I have for this section would be rather than simply listing the output of the code (e.g. how many rows/columns, what the column names are, etc.), you could further expand on how this information will shape your code/analysis. For example, you describe the `fang` and `snp` data sets separately; however, they are associated to each other through the identical snp markers in both data sets (column names in `fang` and observations in `snp`). By noticing and describing this in your analysis, it will be clear to the readers why you later join the data sets using this key variable. Another example would be describing the variable data types, and why you may need to alter them later in your analysis (EX: changing `snp` "Chromosome" and "Position" column to numeric or factor type). In order to not cluster your Rmd with discussion points, you could add a brief description to your README.md where you already discuss what each function accomplishes.

Data Processing

Overall, your code ran and output the correct files. I think your inclusion of sub headers kept your data organized, and it was easy to follow along and know what each line of code accomplished.

I think one useful tool you could use in this section is the piping command (%>% or |>). I noticed later in your data visualization section, you used this symbol to plot your figures, but it is also useful when cleaning up data sets since it minimizes the amount of times you have to re-save a data frame with each new edit. The following code is one I used in my data processing workflow that utilizes the pipe which combines the filtering for group and exclusion of columns:

```
# filtering for maize (Group = ZMMIL, ZMMLR, and ZMMMR) & remove JG_OTU and Group column
maize <- genotypes |>
  filter(Group %in% c("ZMMIL", "ZMMLR", "ZMMMR")) |>
  select(!c(Group, JG_OTU))

# filtering for teosinte (Group = ZMPBA, ZMPIL, and ZMPJA)  remove JG_OTU and Group column
teosinte <- genotypes |>
  filter(Group %in% c("ZMPBA", "ZMPIL", "ZMPJA")) |>
  select(!c(Group, JG_OTU))
```

Another suggestion I have is the format of your missing data. I don't remember if the professor changed the rubric, but the missing data in the original data set is formatted like "?/?", which might be nice to maintain in the case where you would need to separate the nucleotides (could then run sep = "/"). I believe you're following the current rubric, but I distantly remember them discussing this topic in class.

I think your use of sapply to create the functions was well executed. The code itself was divided into multiple lines ensuring easy readability, and the viewable file after each sapply chunk allowed verification of accuracy. If you want to cut down on code for each output file type, creating a function that creates a directory, filters the Chromosomes, sorts by Position, creates a new data frame, and saves the frame into your computer directory would reduce the amount of code chunks you have, and potentially reduce the run time. I have provided the function I used for my analysis if you would like to see an example:

```
# ~~~ FUNCTION CREATION ~~~

process_chr_data <- function(data, chr_num, output_prefix, replace_na, sort_order = "asc", output_dir =
  # create file output directory (unless it already exists)
  if(!dir.exists(output_dir)) {
    dir.create(output_dir, recursive = FALSE)
  }

  # filter data
  filtered_data <- data |>
    filter(Chromosome == chr_num, !grepl("unknown|multiple", Chromosome)) |>
    mutate(
      across(everything(), ~ gsub("\\?/\\?", replace_na, .)),
      Position = as.numeric(as.character(Position))
      ) |>
    arrange(if (sort_order == "asc") Position else desc(Position))

  # write to file
  output_file <- file.path(output_dir, sprintf("%s_chr%d_%s.txt", output_prefix, chr_num, sort_order))
  write_tsv(filtered_data, output_file)
}

# ~~~ maize ascending ~~~
lapply(1:10, function(chr) { # chromosomes 2 - 10
  process_chr_data(maize_join, chr, "maize", "?/?", "asc", "maize_data")
})

# ~~~ maize descending ~~~
lapply(1:10, function(chr) { # chromosomes 2 - 10
  process_chr_data(maize_join, chr, "maize", "-/-", "desc", "maize_data")
})
```

Data Visualization

Your figures are neat and display the information outlined in the assignment instructions. I have a couple suggestions for figure aesthetics that I have listed below:

- "Single Nucleotide Polymorphism per Chromosome" - remove legend; the x-axis label clearly labels the chromosomes, the colors are all unique so no need for color legend, and the legend is out of order (Chr 10 before 2)

```
# your code
ggplot (data = SNP_Teosinte_Maize) + geom_bar(mapping = aes(x = as.factor(as.double(Chromosome)), fill

# addition
+ theme(legend.position = "none")
```

- "Single Nucleotide Polymorphism per Sample" - angle x axis labels so "multiple" and "unknown" don't overlap

```
# your code
by_group_plot <- ggplot (data = SNP_Teosinte_Maize_Groups) + geom_bar(mapping = aes(x = Chromosome, fil
  xlab(label = "Chromosome") + ylab(label = "SNPs") +
  ggtitle("Single Nucleotide Polymorphism per Sample")
by_group_plot

# addition
+ theme(axis.text.x = element_text(angle = 45))
```

I like your creative component figure! I think examining the specific sub groups within the maize and teosinte groups is important to determine if the generalizations about the maize data is accurate for all sub groups within a group, or if one sub group makes up a majority of the data (which is clearly ZMMLR & ZMPBA as seen in your figure).

Once again, I think you completed a thorough and organized analysis. Feel free to take any of my suggestions into account, but even without them, I think your analysis is excellent!