TECH DAYS

THE UNIVERSITY OF NEW MEXICO

Leveraging ChatGPT for R Code- Lightning Round

Heather Mechler & Kevin Mulcaire

Learning Objectives

1) Using pre-made prompts to help write code and navigate schemas

2) Transcribing code between different languages

3) Visualize Data with ChatGPT

How To Use A Pre-made Prompt

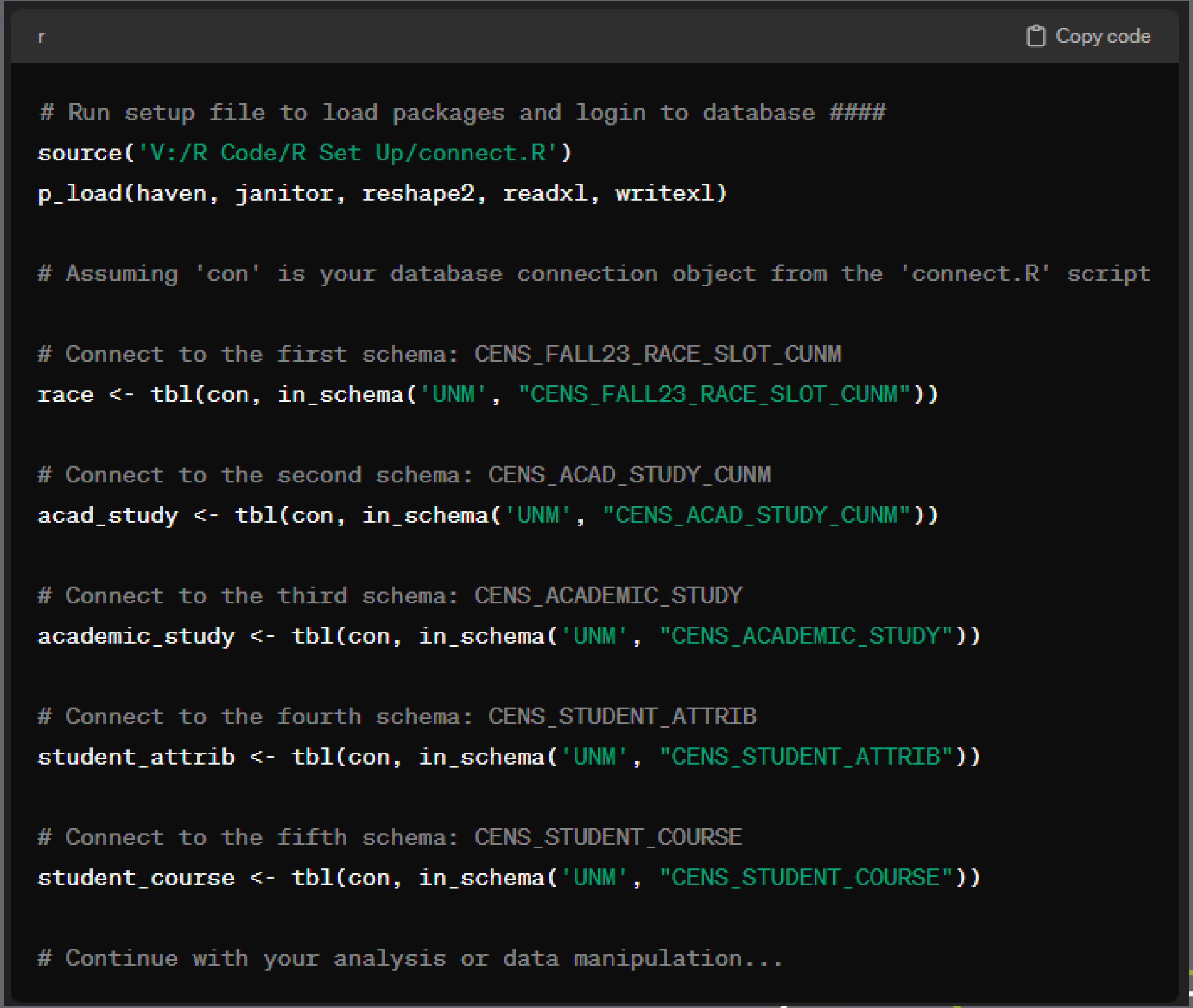
- V:_ Kevin Mulcaire\Chat Gpt projects\schema Prompt.docx
- Applications:
  - Automate the set up to ODBC in R
  - Connect to all or specific schema
  - Browse and select columns in large schema
  - Data architecture (finding shared columns among schema)

Example 1– Set Up

Prompt:

“Follow step one exactly as outlined for setting up connection in R then connect to all 5 schema”

TECH DAYS



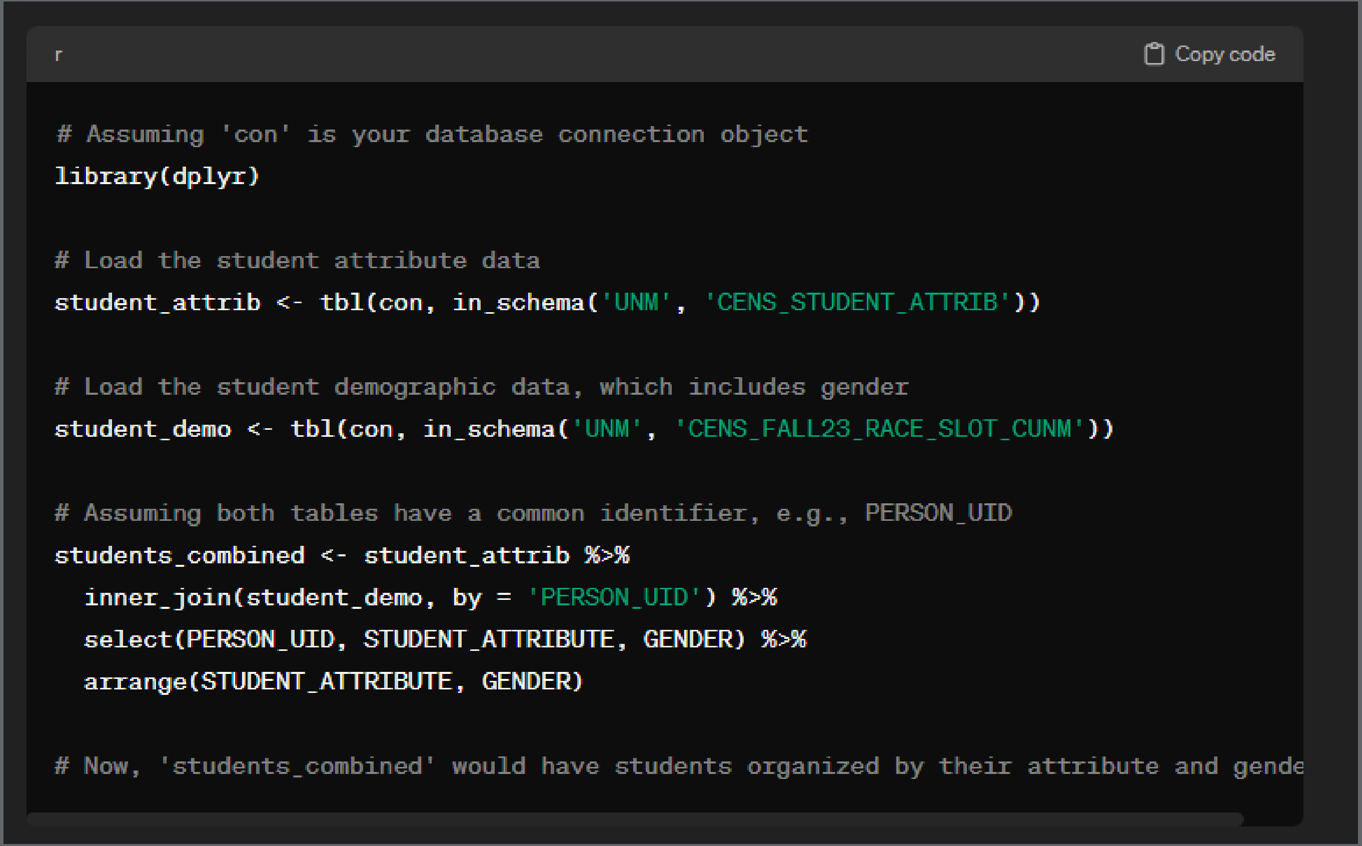
```r
# Run setup file to load packages and login to database ####
source('V:/R Code/R Set Up/connect.R')
p_load(haven, janitor, reshape2, readxl, writexl)

# Assuming 'con' is your database connection object from the 'connect.R' script

# Connect to the first schema: CENS_FALL23_RACE_SLOT_CUNM
race <- tbl(con, in_schema('UNM', "CENS_FALL23_RACE_SLOT_CUNM"))

# Connect to the second schema: CENS_ACAD_STUDY_CUNM
acad_study <- tbl(con, in_schema('UNM', "CENS_ACAD_STUDY_CUNM"))

# Connect to the third schema: CENS_ACADEMIC_STUDY
academic_study <- tbl(con, in_schema('UNM', "CENS_ACADEMIC_STUDY"))

# Connect to the fourth schema: CENS_STUDENT_ATTRIB
student_attrib <- tbl(con, in_schema('UNM', "CENS_STUDENT_ATTRIB"))

# Connect to the fifth schema: CENS_STUDENT_COURSE
student_course <- tbl(con, in_schema('UNM', "CENS_STUDENT_COURSE"))

# Continue with your analysis or data manipulation...
```

THE UNIVERSITY OF NEW MEXICO

Example 2 – Browse Schema

- **Prompt:**

“What schema can I use to organize students by attribute and gender?”

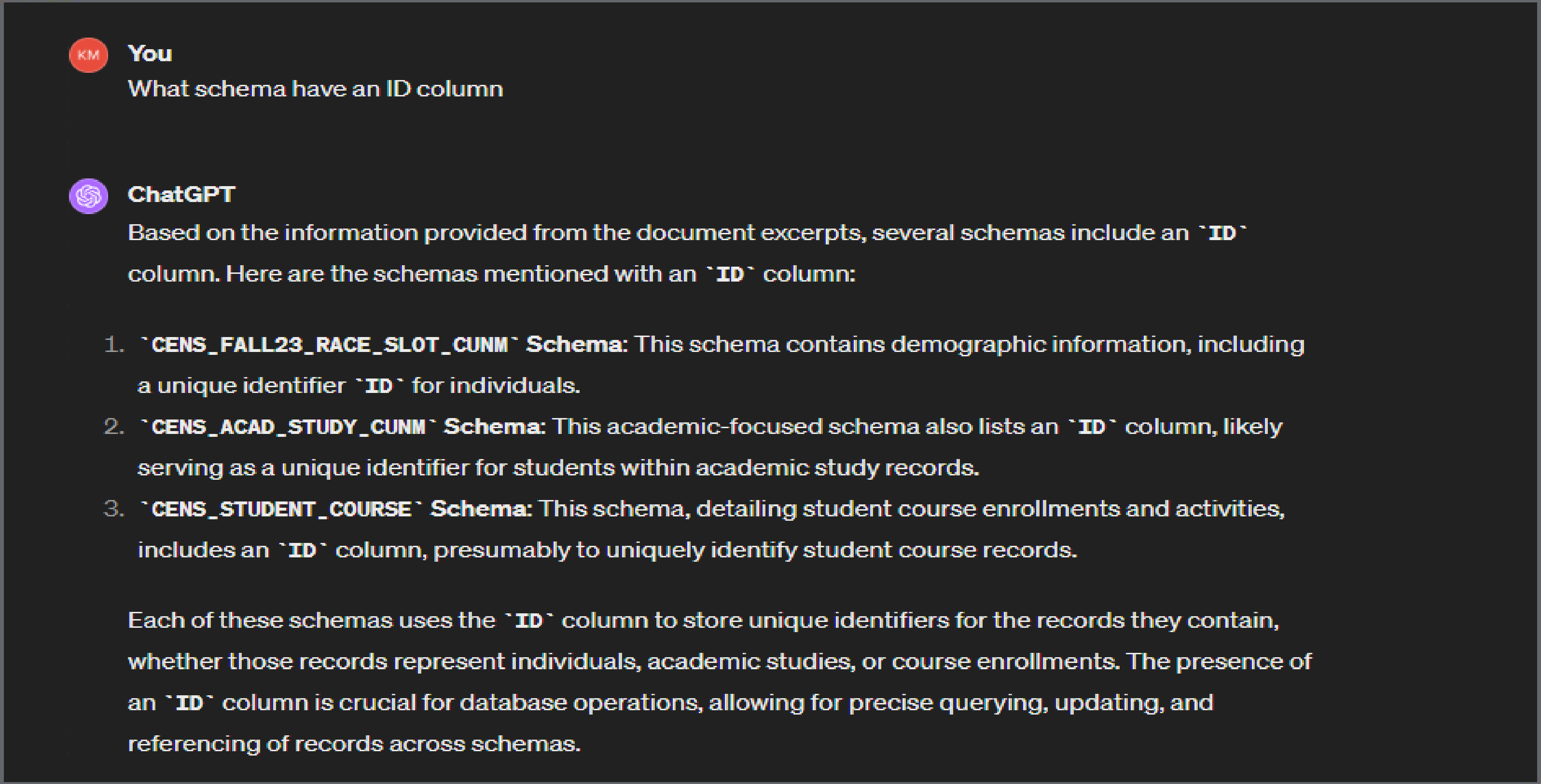


```r
# Assuming 'con' is your database connection object
library(dplyr)

# Load the student attribute data
student_attrib <- tbl(con, in_schema('UNM', 'CENS_STUDENT_ATTRIB'))

# Load the student demographic data, which includes gender
student_demo <- tbl(con, in_schema('UNM', 'CENS_FALL23_RACE_SLOT_CUNM'))

# Assuming both tables have a common identifier, e.g., PERSON_UID
students_combined <- student_attrib %>%
  inner_join(student_demo, by = 'PERSON_UID') %>%
  select(PERSON_UID, STUDENT_ATTRIBUTE, GENDER) %>%
  arrange(STUDENT_ATTRIBUTE, GENDER)

# Now, 'students_combined' would have students organized by their attribute and gende
```

TECH DAYS

Example 3 – Search For a Column

THE UNIVERSITY OF NEW MEXICO



KM **You**

What schema have an ID column

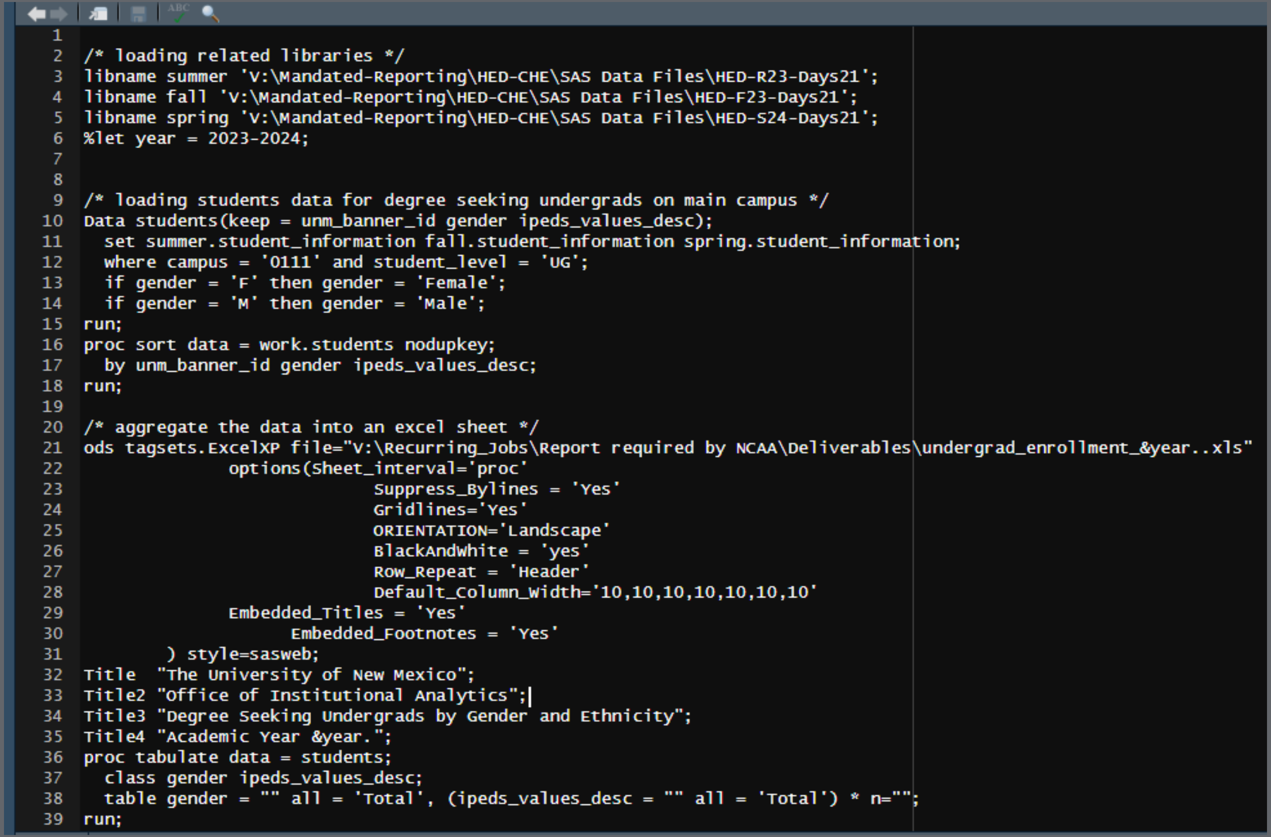
ChatGPT

Based on the information provided from the document excerpts, several schemas include an `ID` column. Here are the schemas mentioned with an `ID` column:

1. `CENS_FALL23_RACE_SLOT_CUNM` **Schema**: This schema contains demographic information, including a unique identifier `ID` for individuals.
2. `CENS_ACAD_STUDY_CUNM` **Schema**: This academic-focused schema also lists an `ID` column, likely serving as a unique identifier for students within academic study records.
3. `CENS_STUDENT_COURSE` **Schema**: This schema, detailing student course enrollments and activities, includes an `ID` column, presumably to uniquely identify student course records.

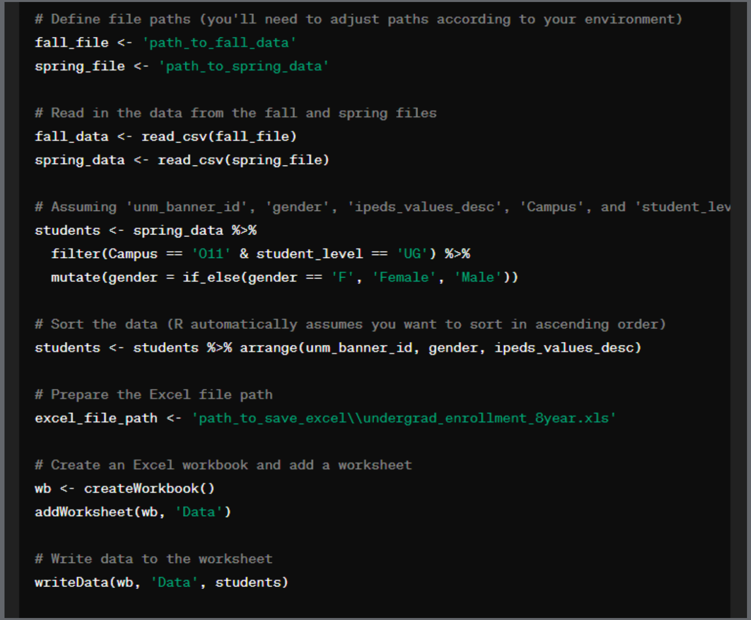
Each of these schemas uses the `ID` column to store unique identifiers for the records they contain, whether those records represent individuals, academic studies, or course enrollments. The presence of an `ID` column is crucial for database operations, allowing for precise querying, updating, and referencing of records across schemas.

Transcribing SAS to R



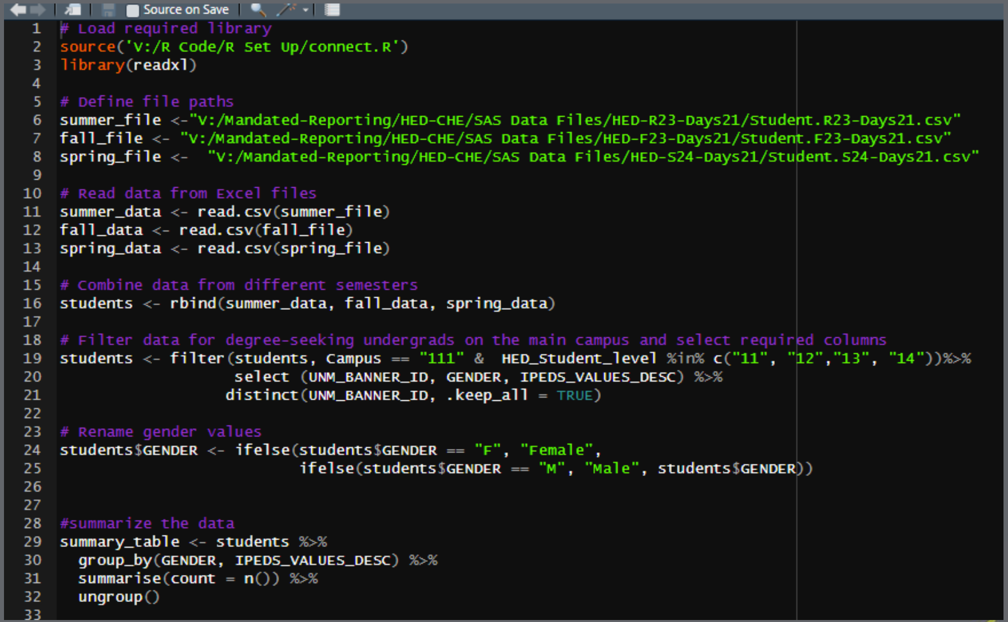
```

/* loading related libraries */
libname summer 'V:\Mandated-Reporting\HED-CHE\SAS Data Files\HED-R23-Days21';
libname fall 'V:\Mandated-Reporting\HED-CHE\SAS Data Files\HED-F23-Days21';
libname spring 'V:\Mandated-Reporting\HED-CHE\SAS Data Files\HED-S24-Days21';
%let year = 2023-2024;


/* loading students data for degree seeking undergrads on main campus */
Data students(keep = unm_banner_id gender ipeds_values_desc);
  set summer.student_information fall.student_information spring.student_information;
  where campus = '0111' and student_level = 'UG';
  if gender = 'F' then gender = 'Female';
  if gender = 'M' then gender = 'Male';
run;
proc sort data = work.students nodupkey;
  by unm_banner_id gender ipeds_values_desc;
run;

/* aggregate the data into an excel sheet */
ods tagsets.ExcelXP file="V:\Recurring_Jobs\Report required by NCAA\Deliverables\undergrad_enrollment_&year..xls"
              options(Sheet_interval='proc'
                            Suppress_Bylines = 'Yes'
                            Gridlines='Yes'
                            ORIENTATION='Landscape'
                            BlackAndWhite = 'yes'
                            Row_Repeat = 'Header'
                            Default_Column_Width='10,10,10,10,10,10,10'
              Embedded_Titles = 'Yes'
                    Embedded_Footnotes = 'Yes'
        ) style=sasweb;
Title  "The University of New Mexico";
Title2 "Office of Institutional Analytics";
Title3 "Degree Seeking Undergrads by Gender and Ethnicity";
Title4 "Academic Year &year.";
proc tabulate data = students;
  class gender ipeds_values_desc;
  table gender = "" all = 'Total', (ipeds_values_desc = "" all = 'Total') * n="";
run;
```

What ChatGPT Produced



```
# Define file paths (you'll need to adjust paths according to your environment)
fall_file <- 'path_to_fall_data'
spring_file <- 'path_to_spring_data'

# Read in the data from the fall and spring files
fall_data <- read_csv(fall_file)
spring_data <- read_csv(spring_file)

# Assuming 'unm_banner_id', 'gender', 'ipeds_values_desc', 'Campus', and 'student_lev
students <- spring_data %>%
  filter(Campus == '011' & student_level == 'UG') %>%
  mutate(gender = if_else(gender == 'F', 'Female', 'Male'))

# Sort the data (R automatically assumes you want to sort in ascending order)
students <- students %>% arrange(unm_banner_id, gender, ipeds_values_desc)

# Prepare the Excel file path
excel_file_path <- 'path_to_save_excel\\undergrad_enrollment_8year.xls'

# Create an Excel workbook and add a worksheet
wb <- createWorkbook()
addWorksheet(wb, 'Data')

# Write data to the worksheet
writeData(wb, 'Data', students)
```

What It Should Have Looked Like



```
# Load required library
source('V:/R Code/R Set Up/connect.R')
library(readxl)

# Define file paths
summer_file <-"V:/Mandated-Reporting/HED-CHE/SAS Data Files/HED-R23-Days21/Student.R23-Days21.csv"
fall_file <- "V:/Mandated-Reporting/HED-CHE/SAS Data Files/HED-F23-Days21/Student.F23-Days21.csv"
spring_file <-  "V:/Mandated-Reporting/HED-CHE/SAS Data Files/HED-S24-Days21/Student.S24-Days21.csv"

# Read data from Excel files
summer_data <- read.csv(summer_file)
fall_data <- read.csv(fall_file)
spring_data <- read.csv(spring_file)

# Combine data from different semesters
students <- rbind(summer_data, fall_data, spring_data)

# Filter data for degree-seeking undergrads on the main campus and select required columns
students <- filter(students, Campus == "111" &  HED_Student_level %in% c("11", "12","13", "14"))%>%
                   select (UNM_BANNER_ID, GENDER, IPEDS_VALUES_DESC) %>%
                  distinct(UNM_BANNER_ID, .keep_all = TRUE)

# Rename gender values
students$GENDER <- ifelse(students$GENDER == "F", "Female",
                          ifelse(students$GENDER == "M", "Male", students$GENDER))


#summarize the data
summary_table <- students %>%
  group_by(GENDER, IPEDS_VALUES_DESC) %>%
  summarise(count = n()) %>%
  ungroup()

```

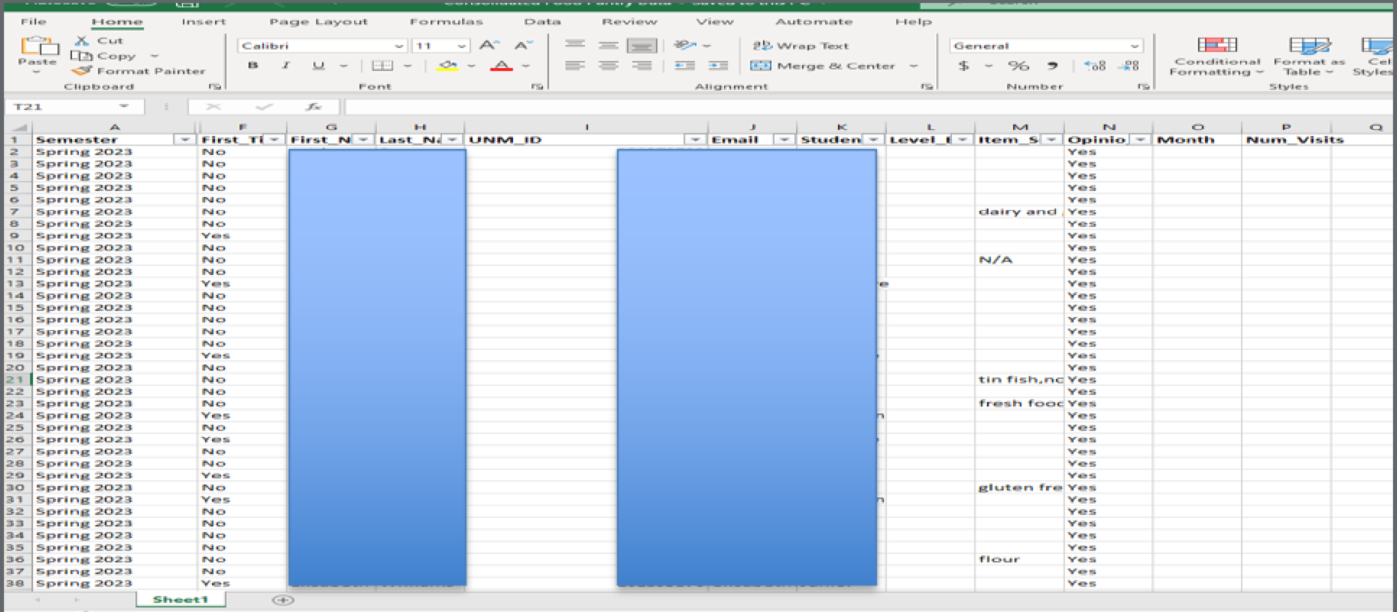
Visualization

- **Raw Data**
  - Only works when the data is local and reasonably small
- **Summarized Data**
  - R summary tables

TECH DAYS

THE UNIVERSITY OF NEW MEXICO

Lobo Food Pantry Data (raw data)

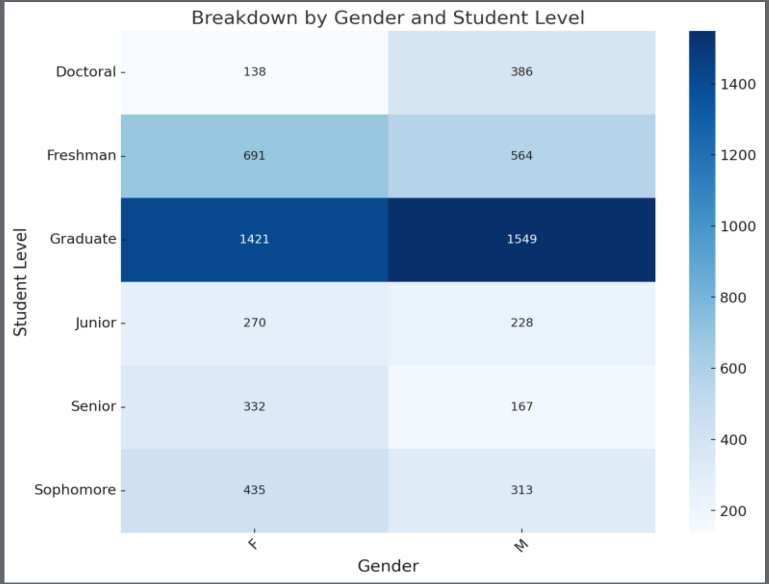


	A	F	G	H	I	J	K	L	M	N	O	P
1	Semester	First_Ti	First_N	Last_Na	UNM_ID	Email	Studen	Level_I	Item_S	Opinio	Month	Num_Visits
2	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
3	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
4	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
5	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
6	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
7	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]		dairy and	Yes		
8	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
9	Spring 2023	Yes	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
10	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
11	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]		N/A	Yes		
12	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
13	Spring 2023	Yes	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
14	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
15	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
16	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
17	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
18	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
19	Spring 2023	Yes	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
20	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
21	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]		tin fish,nc	Yes		
22	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
23	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]		fresh fooc	Yes		
24	Spring 2023	Yes	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
25	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
26	Spring 2023	Yes	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
27	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
28	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
29	Spring 2023	Yes	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
30	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]		gluten fre	Yes		
31	Spring 2023	Yes	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
32	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
33	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
34	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
35	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
36	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]		flour	Yes		
37	Spring 2023	No	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		
38	Spring 2023	Yes	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]			Yes		

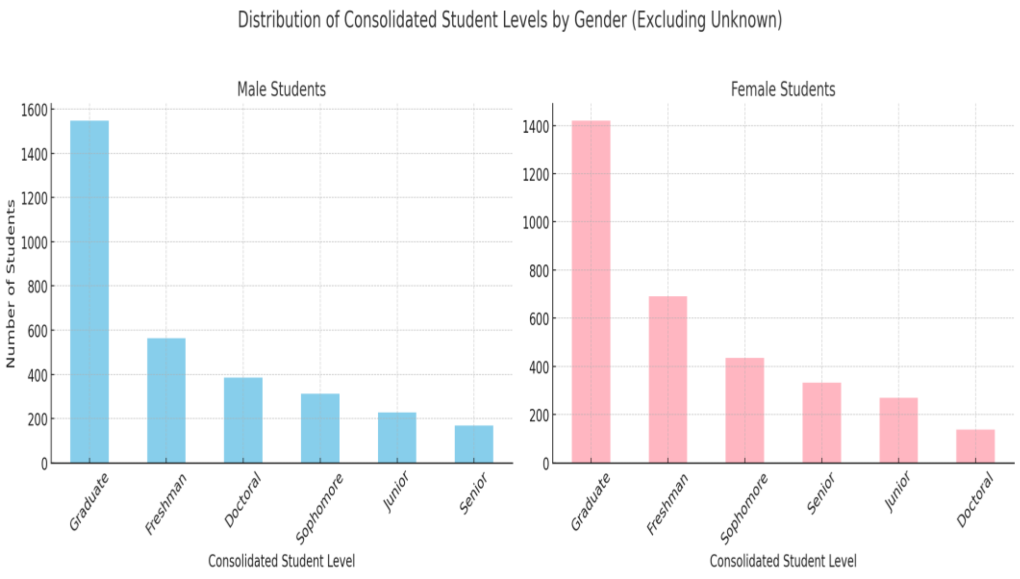
TECH DAYS

THE UNIVERSITY OF NEW MEXICO

"Create a visual that shows a break down by gender and student level. Do not allow for duplicate groups. Make it clear to read"

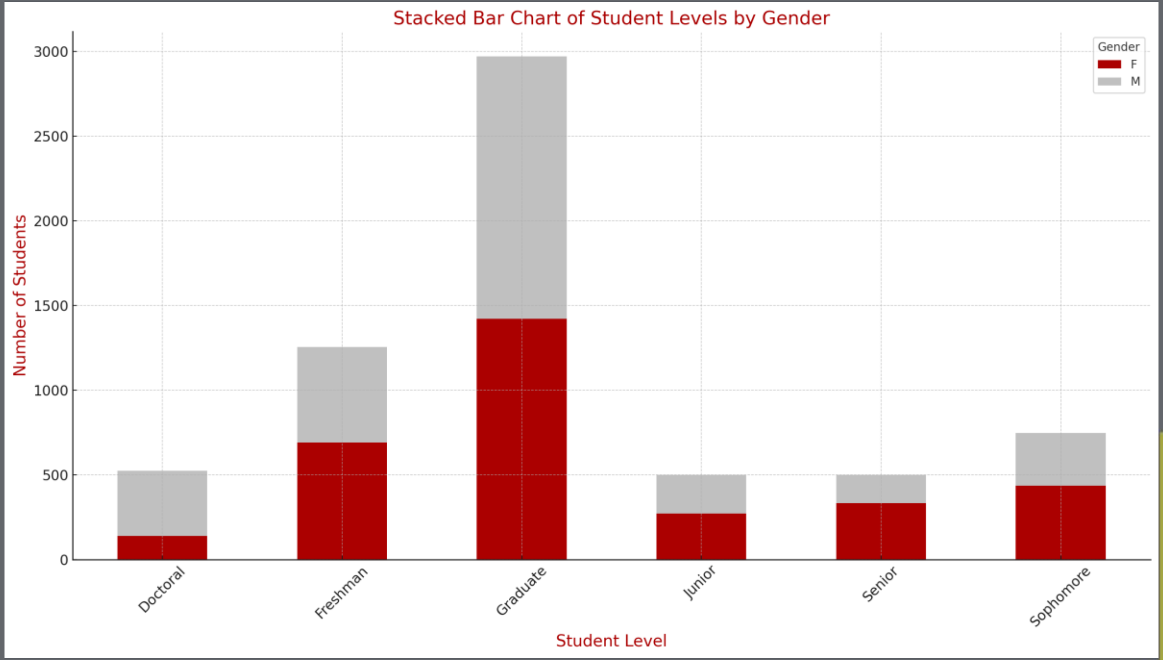

Breakdown by Gender and Student Level

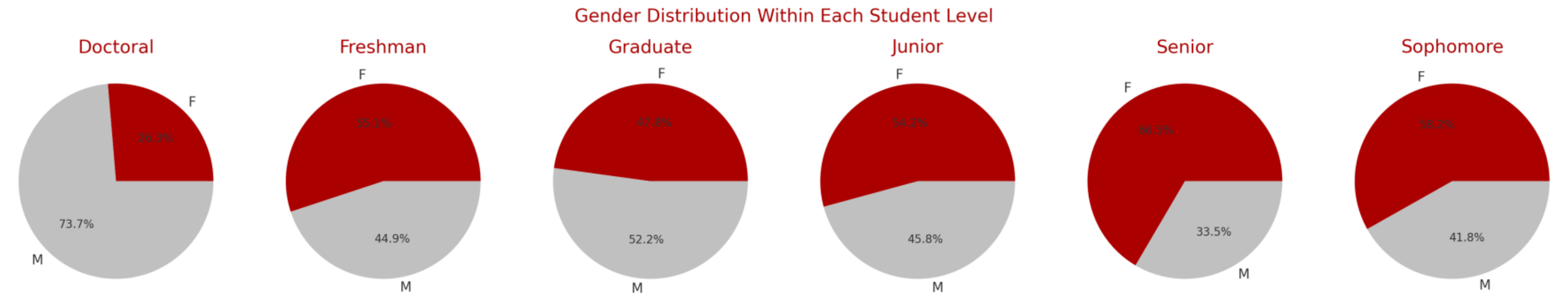
Student Level	F	M
Doctoral	138	386
Freshman	691	564
Graduate	1421	1549
Junior	270	228
Senior	332	167
Sophomore	435	313


Distribution of Consolidated Student Levels by Gender (Excluding Unknown)

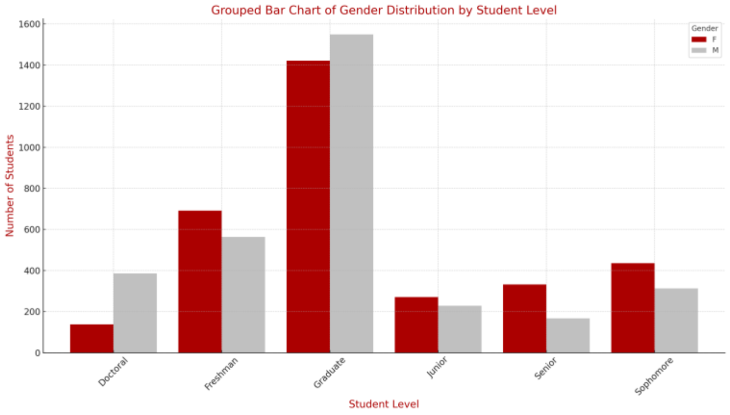
Just For Fun

- “Create 4 more different visuals to display this data in different ways using the UNM colors”

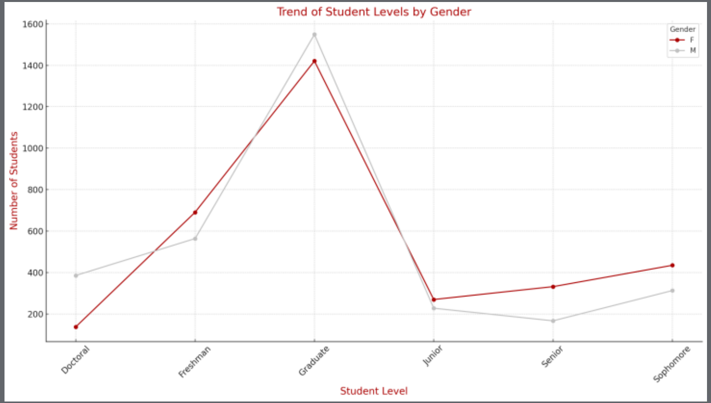




Gender Distribution Within Each Student Level



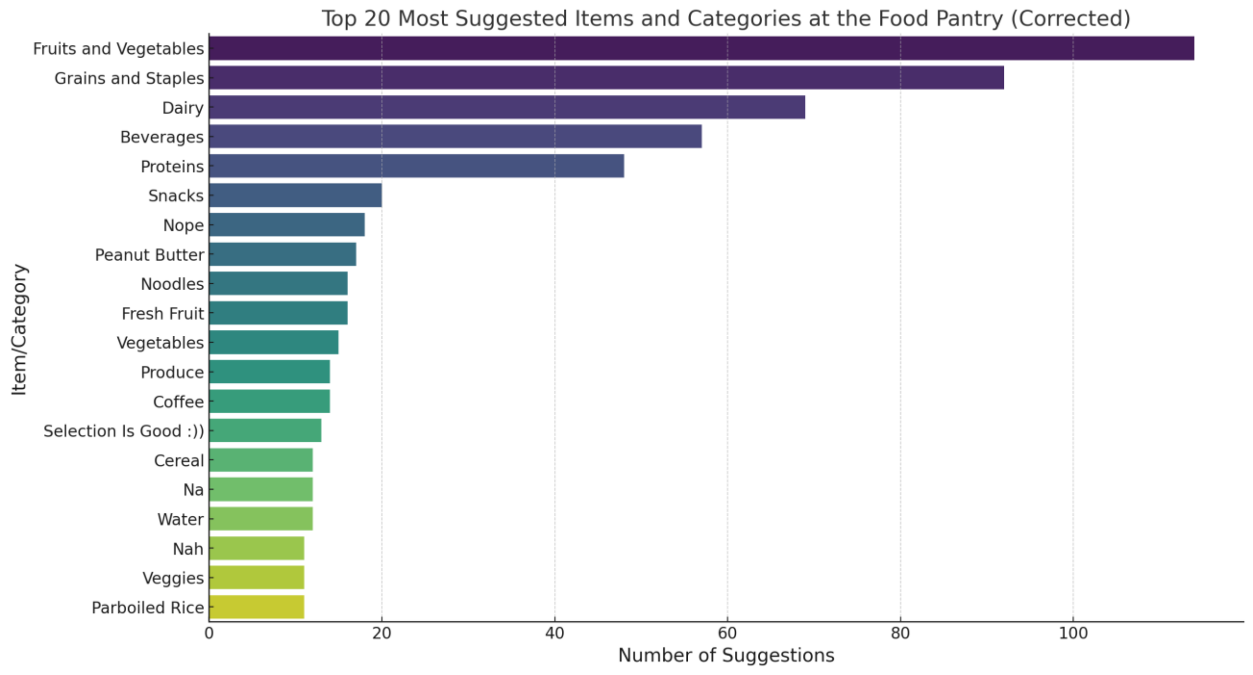
Grouped Bar Chart of Gender Distribution by Student Level



Trend of Student Levels by Gender

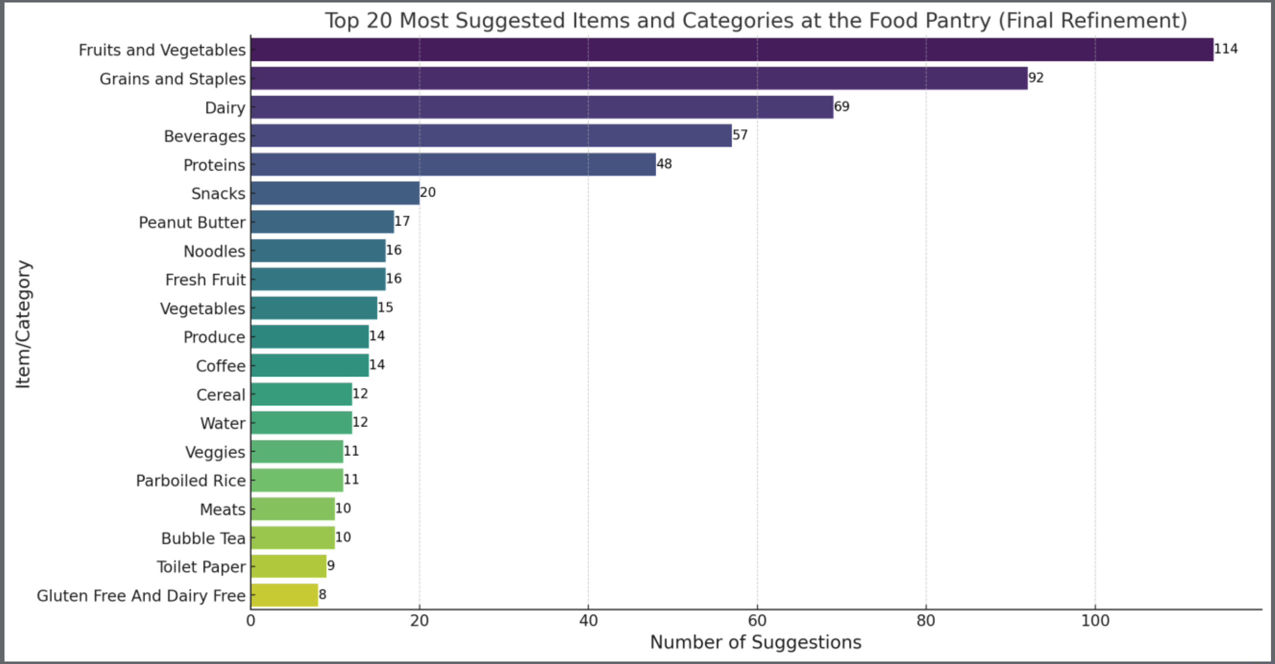
Organizing Open Response Data

"Create a chart of the top 20 most suggested food items and group similar items into one category"



Natural language for Adjusting

"Remove the non-food items then add count labels to each bar"



Summary

- Chat GPT is good at:

- Interoperating natural language
- Generating unique and creative ideas
- Automating repetitive tasks

- Not so good at:

- Predicting what you mean without a lot of guidance
- Following instructions over multiple prompts

Thank You

Questions?