Artificial Intelligence Opinion Survey

DATA 490 Independent Study

Mentor: Dr. Nicholas Dietrich
Researchers: Aadarsha Gopala Reddy, Darren Lo,
Ethan Love, Jose Mancilla, Santo Sumo

April 20, 2023

Contents

1. Load Data

```r
library(tidyverse)
library(ggplot2)


# Load data. Top row is column name.
edu <- read.csv("prolific_edu.csv")
health <- read.csv("prolific_health.csv")
retail <- read.csv("prolific_retail.csv")
tech <- read.csv("prolific_tech.csv")
qualtrics <- read.csv("qualtrics_data.csv")
```

2. Data Cleaning

```r
# Combine data into one data frame after mutating Age to be one data type
edu <- edu %>% mutate(Age = as.character(Age))
health <- health %>% mutate(Age = as.character(Age))
retail <- retail %>% mutate(Age = as.character(Age))
```

```r
tech <- tech %>% mutate(Age = as.character(Age))

combined <- bind_rows(edu, health, retail, tech)
# export combined data to csv
# write.csv(combined, "combined_non_qualtrics.csv")

# combine qualtrics and combined data using qualtrics data's
#     ProlificID column and combined data's Participant id
combined <- left_join(qualtrics, combined,
  by = c("ProlificID" = "Participant.id")
)
# rename "Duration..in.seconds." column to "Duration"
colnames(combined)[
  colnames(combined) == "Duration..in.seconds."
] <- "Duration"
# rename "Ethnicity.simplified" column to "Ethnicity"
colnames(combined)[
  colnames(combined) == "Ethnicity.simplified"
] <- "Ethnicity"
# remove Age.x and keep only Age.y column and rename Age.y to Age
combined <- combined %>%
  select(-Age.x) %>%
  rename(Age = Age.y)
# remove Status.x and Status.y columns
combined <- combined %>%
  select(-Status.x) %>%
  select(-Status.y)
# remove Finished, Progress, UserLanguage, DistributionChannel,
# Nationality, and Consent columns
combined <- combined %>%
  select(-Finished) %>%
  select(-Progress) %>%
  select(-UserLanguage) %>%
  select(-DistributionChannel) %>%
  select(-Nationality) %>%
  select(-Consent)


# remove rows where Submission.id is NA
combined <- combined %>%
  filter(!is.na(Submission.id))
# Keep only rows which say "United States" in "Country.of.residence" column
combined <- combined %>%
  filter(combined$Country.of.residence == "United States")

# Replace all cells that say "Information Technology" and
# "Science, Technology, Engineering & Mathematics" to "STEM/IT"
# in Employment.sector column
combined$Employment.sector[
  combined$Employment.sector == "Information Technology"
] <- "STEM/IT"
combined$Employment.sector[
  combined$Employment.sector == "Science, Technology, Engineering & Mathematics"
] <- "STEM/IT"
```

```r
# Replace all cells in column EnhanceHurt that say
# "AI will neither enhance nor detract from my work" to "neither",
# "AI will enhance my work" to "enhance", and
# "AI will detract from my work" to "detract"
combined$EnhanceHurt[
  combined$EnhanceHurt == "AI will neither enhance nor detract from my work"
] <- "neither"
combined$EnhanceHurt[
  combined$EnhanceHurt == "AI will enhance my work"
] <- "enhance"
combined$EnhanceHurt[
  combined$EnhanceHurt == "AI will detract from my work"
] <- "detract"

# Replace all cells in column TimeEnergy that say
# "Save a lot of time" to "lot of time",
# "Save little time" to "little time",
# "Save no time" to "no time",
# "Save some time" to "some time", and
# "Neutral" to "neutral"
combined$TimeEnergy[
  combined$TimeEnergy == "Save a lot of time"
] <- "lot of time"
combined$TimeEnergy[
  combined$TimeEnergy == "Save little time"
] <- "little time"
combined$TimeEnergy[
  combined$TimeEnergy == "Save no time"
] <- "no time"
combined$TimeEnergy[
  combined$TimeEnergy == "Save some time"
] <- "some time"
combined$TimeEnergy[
  combined$TimeEnergy == "Neutral"
] <- "neutral"

# export data to csv
# write.csv(combined, "combined_qualtrics.csv")
```

```r
# Keep only rows which say "Compose an email" in "Attention" column
combined <- combined %>% filter(combined$Attention == "Compose an email")
# remove Attention column
combined <- combined %>% select(-Attention)
# export data to csv
# write.csv(combined, "combined_qualtrics_attentive.csv")
```

3. Data Exploration

The columns in the dataset are:

- *StartDate* - Date and time survey was started
- *EndDate* - Date and time survey was completed
- *IPAddress* - IP address of participant
- *Duration* - Duration of survey in seconds
- *RecordedDate* - Date and time survey was recorded
- *ResponseId* - Response ID
- *LocationLatitude* - Participant's location latitude
- *LocationLongitude* - Participant's location longitude
- *ProlificID* - Identification of the response on Prolific
- *Gender* - Gender of the participant
- *Education* - Education level of the participant
- *Salary* - Salary of the participant
- *AIKnowledge* - Knowledge of AI of the participant
- *UsedAI* - Whether the participant has used AI
- *TimeEnergy* - How much time and energy AI has saved the participant
- *SimilarTasks* - How much of the participant's tasks they think AI can do
- *EnhanceHurt* - Whether the participant thinks AI can enhance or hurt their work efficiency.
- *Comments* - Comments from the participant
- *Submission.id* - Submission ID
- *Started.at* - Date and time survey was started
- *Completed.at* - Date and time survey was completed
- *Reviewed.at* - Date and time survey was reviewed
- *Archived.at* - Date and time survey was archived
- *Time.taken* - Duration of survey in seconds
- *Completion.code* - Completion code
- *Total.approvals* - Total number of approvals
- *Employment.sector* - Employment sector
- *Age* - Age of the participant
- *Sex* - Sex of the participant
- *Ethnicity* - Ethnicity of the participant
- *Country.of.birth* - Country of birth of the participant
- *Country.of.residence* - Country of residence of the participant
- *Language* - Language of the participant
- *Student.status* - Whether the participant is a student

- *Employment.status* - Whether the participant is employed

4. Data Analysis

4.1. Average amount of time to complete survey

```r
# Create a new data frame with only the columns we need
avg_time <- combined %>% select(Duration)
# Remove rows where Duration is NA
avg_time <- avg_time %>% filter(!is.na(Duration))
# Calculate the average time taken to complete the survey
avg_time <- avg_time %>% summarise(avg_time = mean(Duration))
# Convert to <x> minutes and <y> seconds
avg_time$avg_time <- avg_time$avg_time / 60
avg_time$avg_time <- paste0(
  floor(avg_time$avg_time),
  " minutes and ",
  round((avg_time$avg_time - floor(avg_time$avg_time)) * 60),
  " seconds"
)
# Print the average time taken to complete the survey
avg_time
```

```
##                      avg_time
## 1 2 minutes and 13 seconds
```

4.2. EnhanceHurt vs. Sector

```r
# Create a new data frame with only the columns we need
enhancehurt_vs_sector <- combined %>%
  select(EnhanceHurt, Employment.sector)
```

```r
# For each sector, calculate the proportion of
# respondents who think AI can enhance their work efficiency
enhancehurt_vs_sector <- enhancehurt_vs_sector %>%
  group_by(Employment.sector, EnhanceHurt) %>%
  summarize(count = n()) %>%
  mutate(prop = count / sum(count))
# Visualize using different histogram for each sector.
# Show the number of respondents inside each bar.
# make legend bottom. Wrap x axis labels without changing plot size.
enhancehurt_vs_sector_plot <- ggplot(
  enhancehurt_vs_sector,
  aes(
    x = Employment.sector,
    y = prop, fill = factor(EnhanceHurt,
      levels = c("detract", "neither", "enhance")
    )
  ),
  ylim = c(0, 1)
) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = count),
    position = position_dodge(width = 1),
    vjust = 1.5
  ) +
  labs(
    x = "Employment Sector", y = "Proportion of Respondents",
    fill = "Enhance or Hurt"
  ) +
  ggtitle("Proportion of Respondents Who Think AI Can\n
  Enhance Their Work Efficiency by Employment Sector") +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "bottom"
  )

enhancehurt_vs_sector_plot
```

Proportion of Respondents Who Think AI Can

Enhance Their Work Efficiency by Employment Sector


```
# save to png
ggsave("enhancehurt_vs_sector_plot.png",
  width = 10, height = 6, units = "in", dpi = 300
)
```
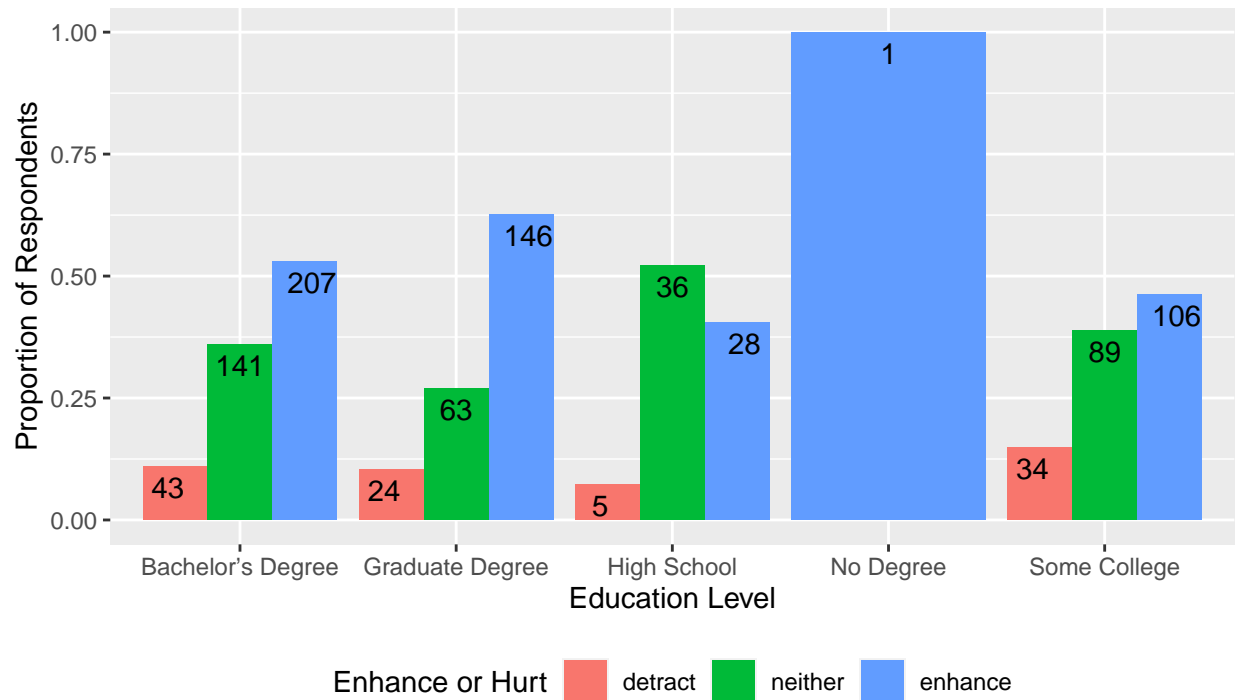
4.3. EnhanceHurt vs. Education

```r
# Create a new data frame with only the columns we need
enhancehurt_vs_education <- combined %>%
  select(EnhanceHurt, Education)
```

```r
# For each education level, calculate the proportion of
# respondents who think AI can enhance their work efficiency
enhancehurt_vs_education <- enhancehurt_vs_education %>%
  group_by(Education, EnhanceHurt) %>%
  summarize(count = n()) %>%
  mutate(prop = count / sum(count))
# Visualize using different histogram for each education level.
# Show the number of respondents on each bar.
enhancehurt_vs_education_plot <- ggplot(
  enhancehurt_vs_education,
  aes(
    x = Education, y = prop, fill = factor(EnhanceHurt,
      levels = c("detract", "neither", "enhance")
    )
  ),
  ylim = c(0, 1)
) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = count),
    position = position_dodge(width = 1),
    vjust = 1.5
  ) +
  labs(
    x = "Education Level", y = "Proportion of Respondents",
    fill = "Enhance or Hurt"
  ) +
  ggtitle("Proportion of Respondents Who Think AI Can\n
  Enhance Their Work Efficiency by Education Level") +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(legend.position = "bottom")

enhancehurt_vs_education_plot
```

Proportion of Respondents Who Think AI Can

Enhance Their Work Efficiency by Education Level


```
# save to png
ggsave("enhancehurt_vs_education_plot.png",
  width = 10, height = 6, units = "in", dpi = 300
)
```
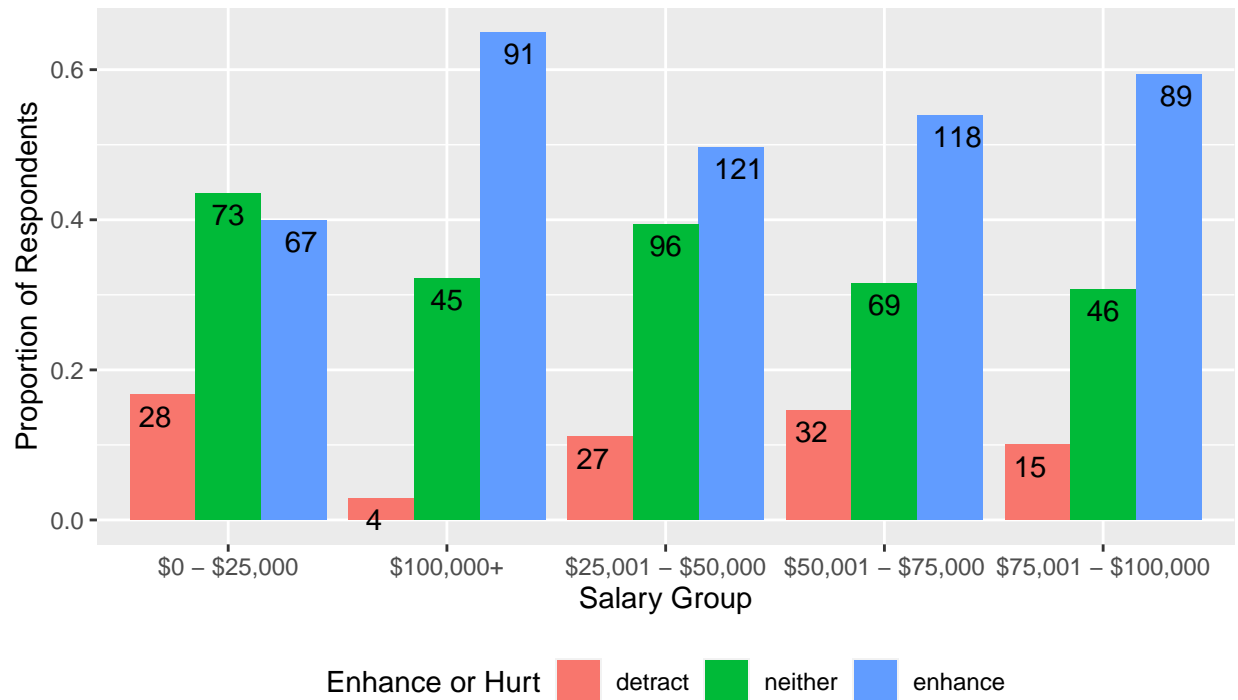
4.4. EnhanceHurt vs. Salary

```r
# Create a new data frame with only the columns we need
enhancehurt_vs_salary <- combined %>% select(EnhanceHurt, Salary)
```

```r
# Create a new dataframe with the proportion of respondents who
# think AI can enhance their work efficiency for each salary group
enhancehurt_vs_salary <- enhancehurt_vs_salary %>%
  group_by(Salary, EnhanceHurt) %>%
  summarize(count = n()) %>%
  mutate(prop = count / sum(count))
# Visualize using different histogram for each salary group.
# Show the number of respondents on each bar.
enhancehurt_vs_salary_plot <- ggplot(
  enhancehurt_vs_salary,
  aes(
    x = Salary, y = prop, fill = factor(EnhanceHurt,
      levels = c("detract", "neither", "enhance")
    )
  ),
  ylim = c(0, 1)
) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = count),
    position = position_dodge(width = 1),
    vjust = 1.5
  ) +
  labs(
    x = "Salary Group",
    y = "Proportion of Respondents",
    fill = "Enhance or Hurt"
  ) +
  ggtitle("Proportion of Respondents Who Think AI Can\n
  Enhance Their Work Efficiency by Salary Group") +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(legend.position = "bottom")

enhancehurt_vs_salary_plot
```

Proportion of Respondents Who Think AI Can

Enhance Their Work Efficiency by Salary Group


```
# save to png
ggsave("enhancehurt_vs_salary_plot.png",
  width = 10, height = 6, units = "in", dpi = 300
)
```
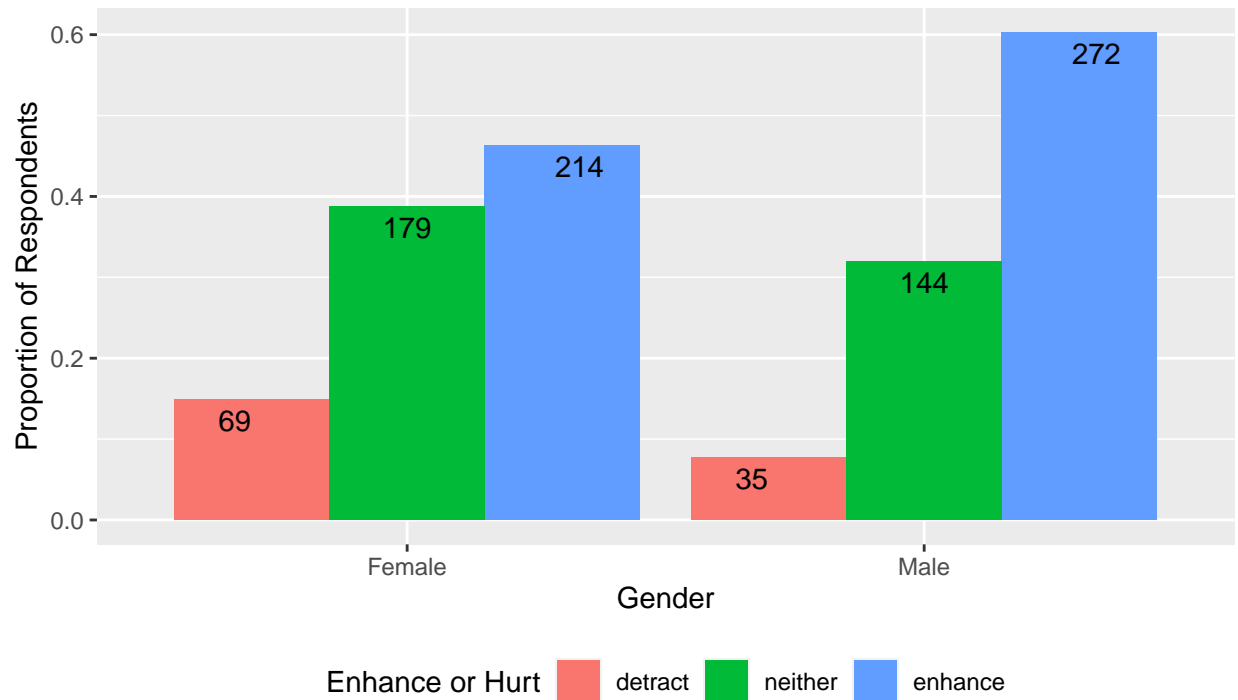
4.5. EnhanceHurt vs. Gender

```r
# Create a new data frame with only the columns we need
enhancehurt_vs_gender <- combined %>% select(EnhanceHurt, Gender)

# Create a new dataframe with the proportion of respondents who
# think AI can enhance their work efficiency for each gender
enhancehurt_vs_gender <- enhancehurt_vs_gender %>%
  group_by(Gender, EnhanceHurt) %>%
  summarize(count = n()) %>%
  mutate(prop = count / sum(count))
# Visualize the proportion of respondents who think AI can
# enhance or hurt, based on gender, using a bar chart.
# Show the number of respondents inside each bar.
# make legend bottom. Wrap x axis labels without changing plot size.
enhancehurt_vs_gender_plot <- ggplot(
  enhancehurt_vs_gender,
  aes(
    x = Gender,
    y = prop, fill = factor(EnhanceHurt,
      levels = c("detract", "neither", "enhance")
    )
  ),
  ylim = c(0, 1)
) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = count),
    position = position_dodge(width = 1),
    vjust = 1.5
  ) +
  labs(
    x = "Gender", y = "Proportion of Respondents",
    fill = "Enhance or Hurt"
  ) +
  ggtitle("Proportion of Respondents Who Think AI Can\n
  Enhance Their Work Efficiency by Gender") +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "bottom"
  )

enhancehurt_vs_gender_plot
```

Proportion of Respondents Who Think AI Can

Enhance Their Work Efficiency by Gender


```
# save to png
ggsave("enhancehurt_vs_gender_plot.png",
  width = 10, height = 6, units = "in", dpi = 300
)
```
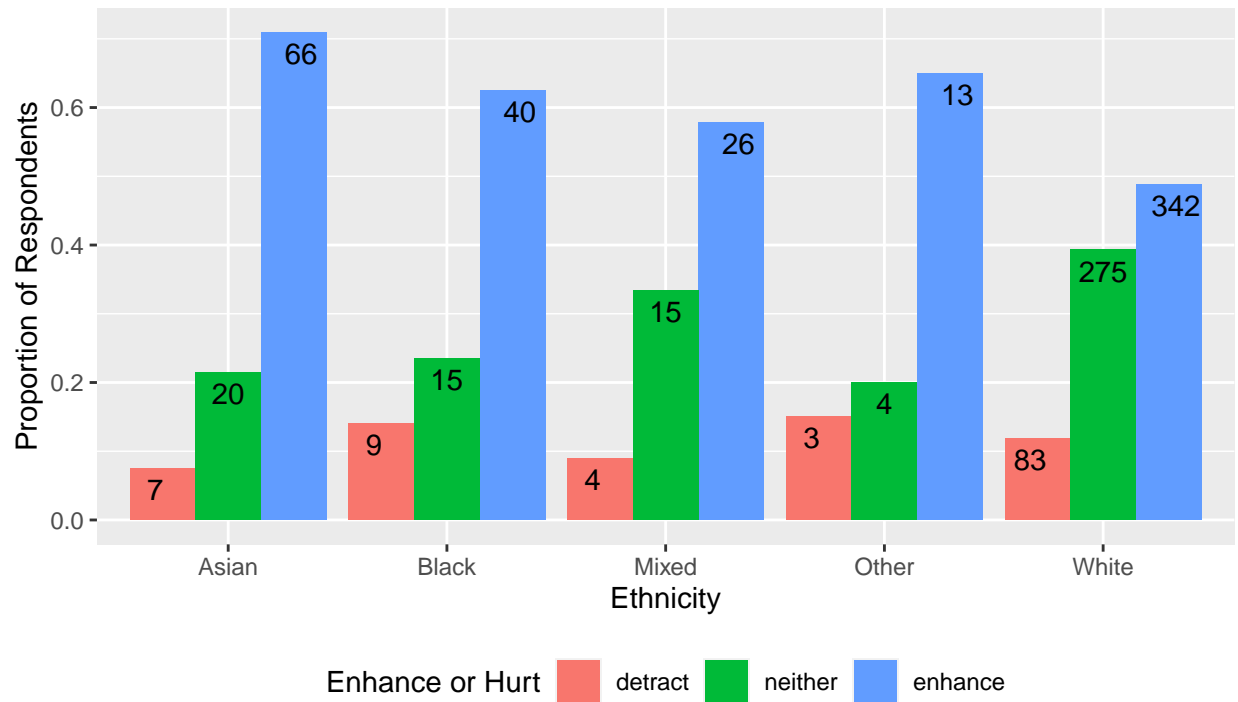
4.6. EnhanceHurt vs. Ethnicity

```r
# Create a new data frame with only the columns we need
enhancehurt_vs_ethnicity <- combined %>% select(EnhanceHurt, Ethnicity)
```

```r
# Create a new dataframe with the proportion of respondents who
# think AI can enhance their work efficiency for each country
enhancehurt_vs_ethnicity <- enhancehurt_vs_ethnicity %>%
  group_by(Ethnicity, EnhanceHurt) %>%
  summarize(count = n()) %>%
  mutate(prop = count / sum(count))
# Visualize the proportion of respondents who think AI can
# enhance or hurt, based on country, using a bar chart.
# Show the number of respondents inside each bar.
# make legend bottom. Wrap x axis labels without changing plot size.
enhancehurt_vs_ethnicity_plot <- ggplot(
  enhancehurt_vs_ethnicity,
  aes(
    x = Ethnicity,
    y = prop, fill = factor(EnhanceHurt,
      levels = c("detract", "neither", "enhance")
    )
  ),
  ylim = c(0, 1)
) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = count),
    position = position_dodge(width = 1),
    vjust = 1.5
  ) +
  labs(
    x = "Ethnicity",
    y = "Proportion of Respondents",
    fill = "Enhance or Hurt"
  ) +
  ggtitle("Proportion of Respondents Who Think AI Can\n
  Enhance Their Work Efficiency by Ethnicity") +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "bottom"
  )

enhancehurt_vs_ethnicity_plot
```

Proportion of Respondents Who Think AI Can

Enhance Their Work Efficiency by Ethnicity


```
# save to png
ggsave("enhancehurt_vs_ethnicity_plot.png",
  width = 10, height = 6, units = "in", dpi = 300
)
```
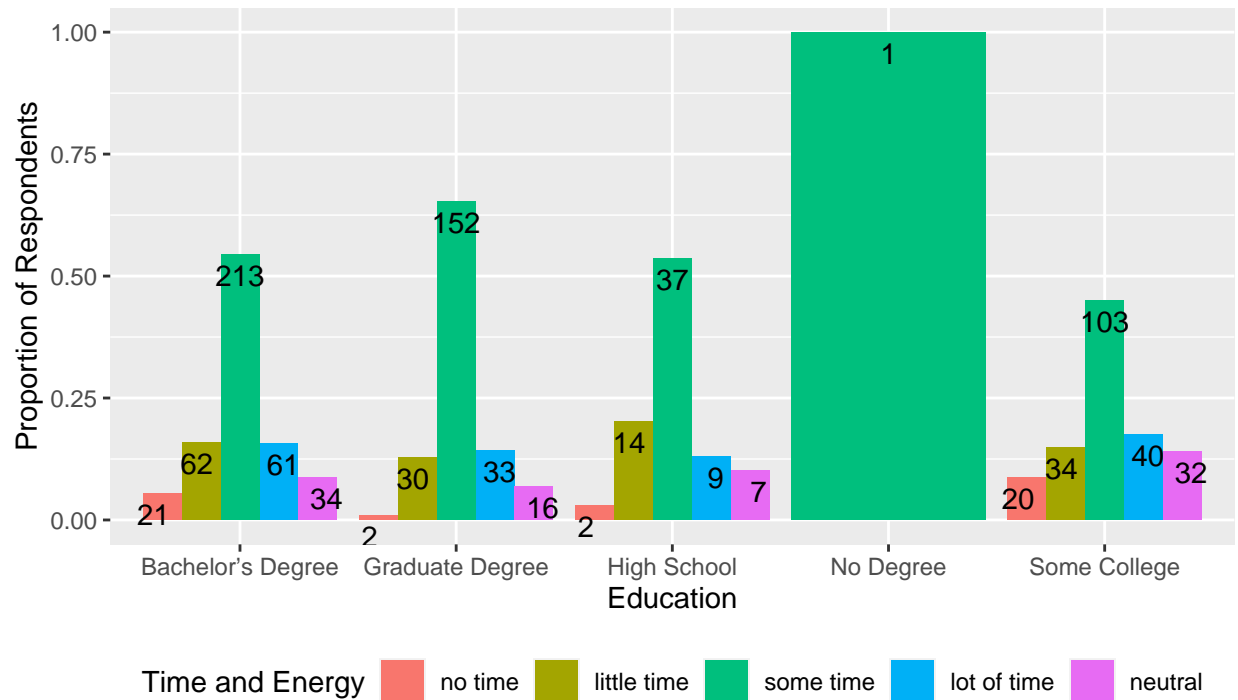
4.7. TimeEnergy vs. Education

```r
# Create a new data frame with only the columns we need
timeenergy_vs_education <- combined %>% select(TimeEnergy, Education)
```

```r
# Create a new dataframe with the proportion of respondents who
# think AI can save their time and energy for each education level
timeenergy_vs_education <- timeenergy_vs_education %>%
  group_by(Education, TimeEnergy) %>%
  summarize(count = n()) %>%
  mutate(prop = count / sum(count))
# Visualize the proportion of respondents who think AI can
# save their time and energy, based on education level, using a bar chart.
# Show the number of respondents inside each bar.
# make legend bottom. Wrap x axis labels without changing plot size.
# order the bars as "no time", "little time", "some time", "lot of time", and "neutral".
timeenergy_vs_education_plot <- ggplot(
  timeenergy_vs_education,
  aes(
    x = Education,
    y = prop, fill = factor(TimeEnergy,
      levels = c("no time", "little time", "some time", "lot of time", "neutral")
    )
  ),
  ylim = c(0, 1)
) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = count),
    position = position_dodge(width = 1),
    vjust = 1.5
  ) +
  labs(
    x = "Education", y = "Proportion of Respondents",
    fill = "Time and Energy"
  ) +
  ggtitle("Proportion of Respondents Who Think AI Can\n
  Save Their Time and Energy by Education Level") +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "bottom"
  )

timeenergy_vs_education_plot
```

Proportion of Respondents Who Think AI Can

Save Their Time and Energy by Education Level


```r
# save to png
ggsave("timeenergy_vs_education_plot.png",
  width = 10, height = 6, units = "in", dpi = 300
)
```
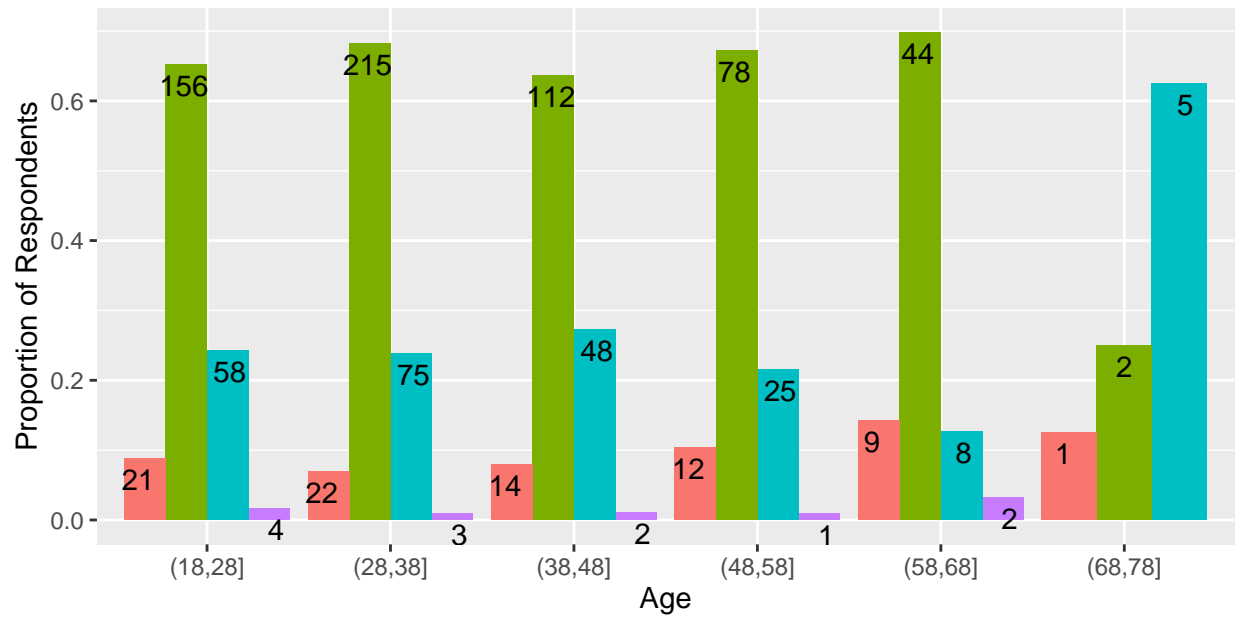
4.8. SimilarTasks vs. Age

```r
# Create a new data frame with only the columns we need
similartasks_vs_age <- combined %>% select(SimilarTasks, Age)
```

```r
# Visualize the proportion of respondents who think AI can
# do similar tasks, based on age group, using a bar chart.
# Show the number of respondents inside each bar.
# make legend bottom. Wrap x axis labels without changing plot size.
# order the bars as "Cannot perform any tasks", "Can perform only a few tasks",
# "Can perform many tasks", "Can perform all tasks".
# force the y axis to start at 0 and end at 1.
similartasks_vs_age_plot <- ggplot(
  similartasks_vs_age,
  aes(
    x = Age,
    y = prop, fill = factor(SimilarTasks,
      levels = c("Cannot perform any tasks", "Can perform only a few tasks",
      "Can perform many tasks", "Can perform all tasks")
    )
  ),
  ylim = c(0, 1)
) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = count),
    position = position_dodge(width = 1),
    vjust = 1.5
  ) +
  labs(
    x = "Age", y = "Proportion of Respondents",
    fill = "Similar Tasks"
  ) +
  ggtitle("Proportion of Respondents Who Think AI Can\n
  Do Similar Tasks by Age Group") +
  theme(
    plot.title = element_text(hjust = 0.5),
    legend.position = "bottom"
  )

similartasks_vs_age_plot
```

Proportion of Respondents Who Think AI Can

Do Similar Tasks by Age Group


```
# save to png
ggsave("similartasks_vs_age_plot.png",
  width = 10, height = 6, units = "in", dpi = 300
)
```