Exploratory Data Analysis

This project has two raw data files at different scales from a study of infants, children, and adults watching a series of 7 video clips. I wrote Steps 1 and 2 to import and merge the data, and kept them here for your reference. Skip down to Step 3 to work on EDA.

SOURCE DESCRIPTION

FILE 1: auc.csv

Columns:

- stim (stimulus video, levels/labels provided below)
- id (unique participant identifier)
- age (in days)
- AUC_sal (area-under-the-curve for a saliency model)
- AUC_dist (area-under-the-curve for a distance model)

AUC values indicate how well each model predicted where participants looked when watching a video. AUC values can range from 0-1 where .5 is chance and 1 is perfect prediction.

FILE 2: participants_info_full_headers.csv

Columns:

- id (unique participant identifier, matches auc.csv)
- age_group (a categorical age variable with levels:
- ".5-1 y" "1-1.5 y" "1.5-2 y" "2-4 y" "4-6 y" "8-10 y" "adult"
- precision (a quality measure of the eye data, smaller is better)
- 7 columns of "Seen X" the stimulus video before the study coded as SEEN (1), NOT SEEN (2), NOT SURE (3)

STEP 1: Read in the AUC data

Code stim as a factor.

```
auc <- read_csv(here("data_raw", "auc_bystim.csv"))
stim_levels <- 1:7
stim_labels <- c("Fallon","Feist","Pentatonix","Science","Rube","Plane","Dogs")
auc <- auc %>% mutate(stim = factor(stim, levels = stim_levels, labels = stim_labels))
```

STEP 2: Read in the participant info data

Wrangle the ppt info data so that you can merge it into the auc data #Drop any data where the AUC values are missing. In the final, merged data, make the "watched" variable is coded as a factor with levels "seen" (1), "not seen" (2), "not sure" (3). Write the cleaned file to data_cleaned/.

Read in the ppt data and rename columns to be easier to work with.

```
ppt <- read_csv(here("data_raw","participants_info_full_headers.csv")) %>%
  rename(id = `participant ID`,
         age_group = `Age group`,
         precision = "Precision")
```

Each question about watching each video is a column, so pivot_longer. Use separate to get just the video name into it's own column.

```
ppt_long <- ppt %>%
  pivot_longer(cols = starts_with("Seen"), names_to = "stim", values_to = "watched")

ppt_long <- ppt_long %>%
  separate(stim, into = c(NA, "stim"))
```

Code stim and watched as factors.

```
ppt_long <- ppt_long %>%
  mutate( stim = factor(stim, levels = stim_labels, labels = stim_labels),
          watched = factor(watched, levels = 1:3, labels = c("Yes","No","Not Sure")))
```

Join the ppt data to the AUC data (by id and by stim since each participant has observations for each stim).

```
ds <- left_join(auc, ppt_long, by = c("id", "stim"))
ds <- ds %>% drop_na(AUC_sal:AUC_dist) #Drop participants for whom we don't have data for the
```

Write the data to file.

```
ds %>% write_csv(here("data_cleaned","cleaned.csv"))
```

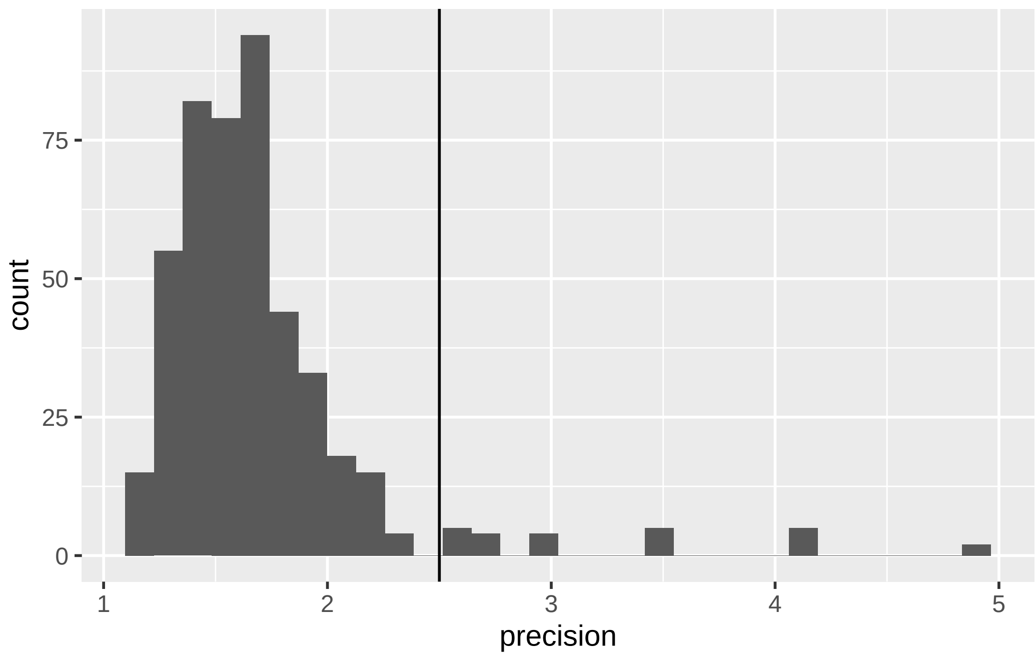
STEP 3: Exploratory Data Analysis

3A Precision

Is the precision acceptable (< 2.5) for each participant?

Visualize the distribution of precision to see if there are values above 2.5

```
ds %>% ggplot(aes(x = precision)) + geom_histogram() + geom_vline(xintercept = 2.5)
```



Create a summary to figure out which participants would we need to exclude if > 2.5 meant the data are unuseable?

```
ds %>%
  group_by(id, age_group) %>%
  summarize(precision = mean(precision, na.RM = T)) %>%
  filter(precision > 2.5)
```

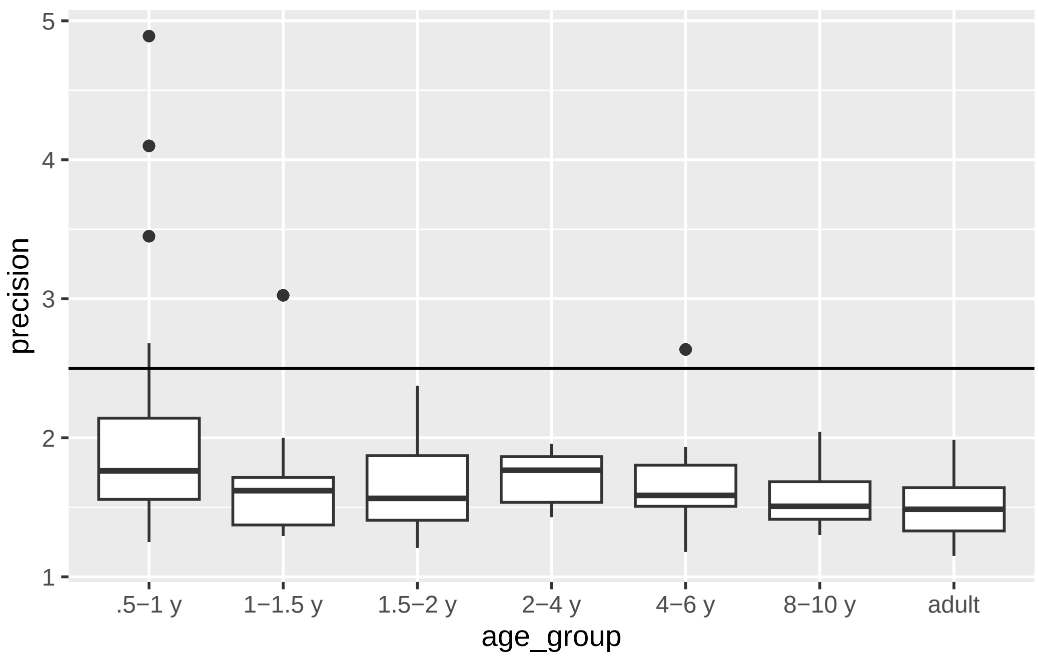
```
# A tibble: 6 x 3
# Groups:   id [6]
     id age_group precision
  <dbl> <chr>         <dbl>
1    52 1-1.5 y        3.02
2    78 .5-1 y         3.45
3    79 .5-1 y         4.89
4    81 .5-1 y         2.68
5    84 4-6 y          2.64
6   108 .5-1 y         4.1
```

Use a summary table and plots to investigate whether data equally precise for participants of different ages

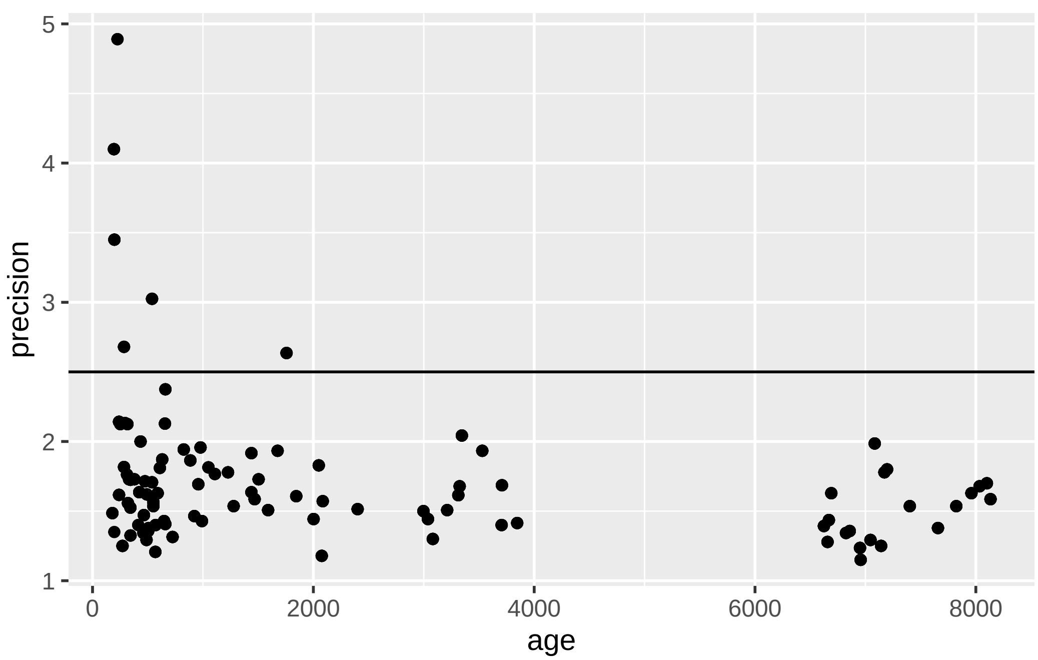
```
ds %>% group_by(age_group) %>% summarize(across(precision, list(M = mean, MIN = min, MAX = ma
```

```
# A tibble: 7 x 4
  age_group precision_M precision_MIN precision_MAX
  <chr>           <dbl>         <dbl>         <dbl>
1 .5-1 y           2.11          1.25          4.89
2 1-1.5 y          1.66          1.29          3.02
3 1.5-2 y          1.64          1.21          2.38
4 2-4 y            1.72          1.43          1.96
5 4-6 y            1.68          1.18          2.64
6 8-10 y           1.59          1.3           2.04
7 adult            1.50          1.15          1.99
```

```
ds %>% ggplot(aes(x = age_group, y = precision)) + geom_boxplot() + geom_hline(yintercept = 
```




```
ds %>% ggplot(aes(x = age, y = precision)) + geom_point() + geom_hline(yintercept = 2.5)
```



3B AGE:

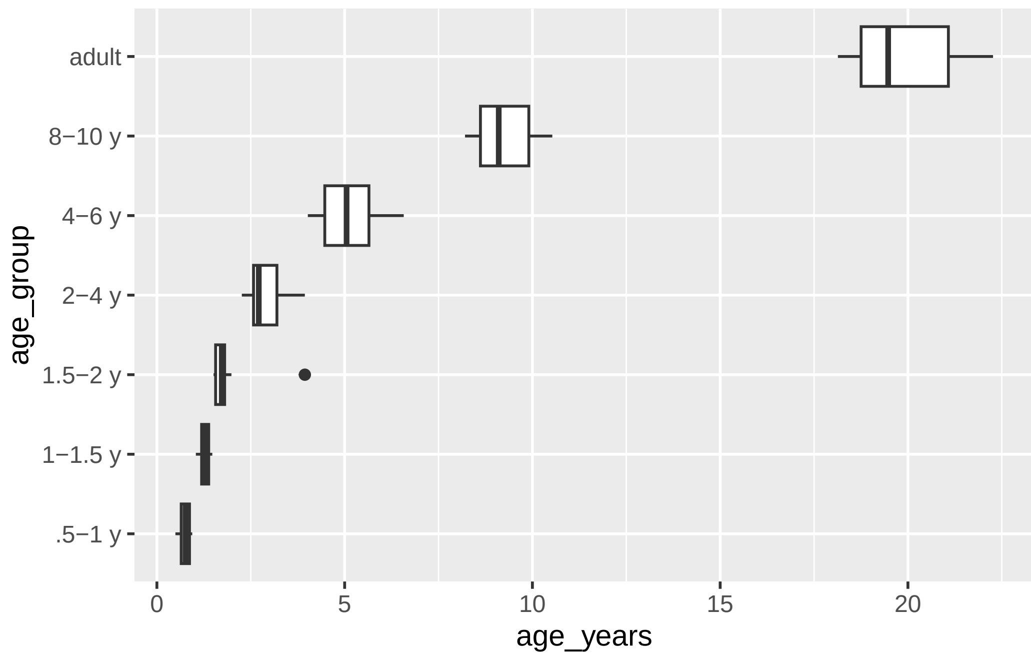
Are there any errors in age?

Convert age to years so that it can be more easily compared to age_group

```
ds <- ds %>% mutate(age_years = age/365.25)
```

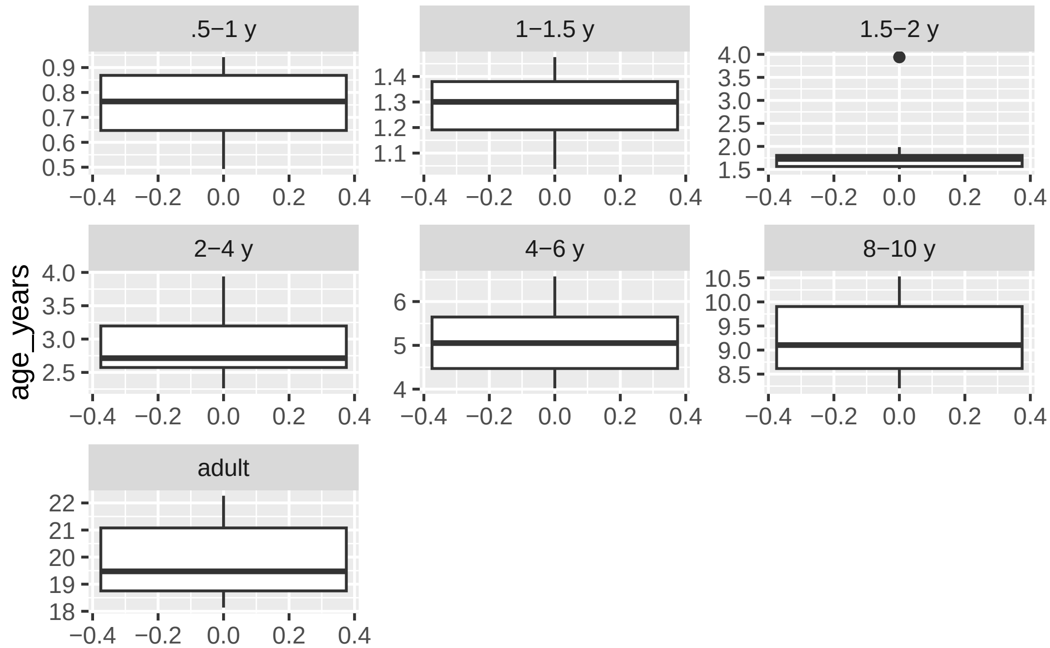
Visualize age in years by age_group to see whether participants are the correct age for their group

```
ds %>% group_by(id, age_group) %>%
  summarize(age_years = mean(age_years)) %>%
  ggplot(aes(y = age_group, x = age_years)) + geom_boxplot()
```



Another option would be to facet by age group and to let the scales be "free" to get a better look

```
ds %>% group_by(id, age_group) %>%
  summarize(age_years = mean(age_years)) %>%
  ggplot(aes(y = age_years)) +
  geom_boxplot() +
  facet_wrap("age_group", scales = "free")
```



Make a summary table of age in years by age group to check whether all participants' ages are correct

```
ds %>% group_by(age_group) %>% summarize(min_age = min(age_years), max_age = max(age_years))
```

```
# A tibble: 7 x 3
  age_group min_age max_age
  <chr>       <dbl>   <dbl>
1 .5-1 y      0.493   0.942
2 1-1.5 y     1.04    1.48
3 1.5-2 y     1.51    3.94
4 2-4 y       2.26    3.94
5 4-6 y       4.02    6.57
6 8-10 y      8.21   10.5
7 adult      18.1    22.3
```