Kaiser sample data analysis

Steve Linberg / DACSS 601 (Online)

2021-01-16

This project explores data from a late 2020 study by the Kaiser Family Foundation. The study and its data may be found at:

https://ropercenter.cornell.edu/ipoll/study/31118130

Citation text:

Henry J. Kaiser Family Foundation. Kaiser Family Foundation Poll: December 2020 Kaiser Health Tracking Poll/COVID-19 Vaccine Monitor, 2020 [Dataset]. Roper #31118130, Version 2. SSRS [producer]. Cornell University, Ithaca, NY: Roper Center for Public Opinion Research [distributor]. doi:10.25940/ROPER-31118130

It is submitted as homeworks 3 and 4 for DACCS 601, showing an import/cleaup process and a line of inquiry into the possible correlation of gender and age among the respondents.

```
raw_data <- read_csv("31118130.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_character(),
##   end_state1 = col_double(),
##   acarot = col_number(),
##   endq1 = col_double(),
##   q2rot = col_logical(),
##   endq2 = col_double(),
##   endq10 = col_double(),
##   q17rot = col_number(),
##   endq22 = col_double(),
##   partyrot = col_number(),
##   length = col_double(),
##   reprostat = col_logical(),
##   qctr = col_double(),
##   xctr = col_double(),
##   wt1 = col_double(),
##   weight = col_double(),
##   standwt = col_double(),
##   weight_ssrs = col_double(),
##   cdc2013 = col_double()
## )
## i Use `spec()` for the full column specifications.
```

```
raw_data
```

```
## # A tibble: 1,676 x 168
##    id    d1    end_state1 hispanic race  racethn nativity racethn2 acarot aca
```

```
##    <chr> <chr>      <dbl> <chr>    <chr> <chr>   <chr>    <chr>     <dbl> <chr>
##  1 0000~ Male         789 No       White WHITE,~ <NA>     WHITE, ~     41 Some~
##  2 0000~ Male        1148 No       White WHITE,~ <NA>     WHITE, ~     14 Some~
##  3 0000~ Male         901 Yes      White HISPAN~ U.S.     HISPANI~     14 Very~
##  4 0000~ Male         685 No       White WHITE,~ <NA>     WHITE, ~     41 Some~
##  5 0000~ Male         656 No       White WHITE,~ <NA>     WHITE, ~     41 Some~
##  6 0000~ Male         704 No       Othe~ OTHER,~ <NA>     OTHER, ~     41 Very~
##  7 0000~ Male         749 No       White WHITE,~ <NA>     WHITE, ~     41 Very~
##  8 0000~ Fema~       1001 Yes      Othe~ HISPAN~ Another~ HISPANI~     14 Some~
##  9 0000~ Male         737 No       White WHITE,~ <NA>     WHITE, ~     41 Very~
## 10 0000~ Male         933 No       White WHITE,~ <NA>     WHITE, ~     14 Very~
## # ... with 1,666 more rows, and 158 more variables: q1rot <chr>, q1 <chr>,
## #   endq1 <dbl>, q2rot <lgl>, q2rot2 <chr>, q2a <chr>, q2b <chr>, q2c <chr>,
## #   q2d <chr>, q2e <chr>, q2f <chr>, q2g <chr>, endq2 <dbl>, q3 <chr>,
## #   q27 <chr>, q4rot <chr>, q4 <chr>, q5rot <chr>, q5 <chr>, q6 <chr>,
## #   q7rot <chr>, q7rot2 <chr>, q7a <chr>, q7b <chr>, q8 <chr>, q9rot <chr>,
## #   q9 <chr>, q10rot <chr>, q10a <chr>, q10b <chr>, q10c <chr>, endq10 <dbl>,
## #   q16 <chr>, q17rot <dbl>, q17 <chr>, q18rot <chr>, q18a <chr>, q18b <chr>,
## #   q18c <chr>, q18e <chr>, q18f <chr>, q18g <chr>, q18h <chr>, q18i <chr>,
## #   q18j <chr>, q11rot <chr>, q11 <chr>, q12rot <chr>, q12a <chr>, q12b <chr>,
## #   q12c <chr>, q12d <chr>, q12e <chr>, q12f <chr>, q12g <chr>, q12h <chr>,
## #   q12i <chr>, q13rot <chr>, q13 <chr>, q14rot <chr>, q14 <chr>, q15 <chr>,
## #   q19_q20rot <chr>, q19 <chr>, q20 <chr>, q21 <chr>, q22 <chr>, endq22 <dbl>,
## #   age <chr>, age2 <chr>, recage <chr>, recage2 <chr>, recage3 <chr>,
## #   recage4 <chr>, recage5 <chr>, child <chr>, marital <chr>, rvote <chr>,
## #   voted <chr>, voted2rot <chr>, voted2 <chr>, voted2ot <chr>,
## #   inclosstotal <chr>, employ <chr>, recemploy <chr>, essential <chr>,
## #   hcworker2 <chr>, hcworker3 <chr>, coverage <chr>, agecov <chr>,
## #   covtype <chr>, agecovtype <chr>, covselfother <chr>, q23 <chr>,
## #   q23ot1 <chr>, q23ot2 <chr>, q23ot3 <chr>, q23ot4 <chr>, rsex <chr>,
## #   gendervar <chr>, ...
```

Before looking into potential questions relating to gender in this dataset, it is worth looking at the distributions of age within gender among the survey's respondents to see if there is any correlation; if gender is skewed by age at all, it would complicate any observations of other columns using gender as a dependent variable, because it could include hidden bias involving age as well as gender.

As this dataset contains 1676 observations of 168 variables, we'll filter the raw data down to what is minimally necessary to examine age and gender. The `d1` column contains gender, and `age` (column 79) contains a textual representation of the respondent's age.

A quick frequency table of the `d1` fields shows two clean values:

```
d1_freq = table(raw_data$d1)
d1_freq
```

```
##
## Female   Male
##    763    913
```

There are 763 observations for "Female", and 913 for "Male". The `d1` gender data is well-formed and does not need any further work (beyond renaming).

`Age` is slightly more complicated, however; although it looks like an integer, it is actually a `<chr>` variable type, and it has some non-numeric values:

```
summary(raw_data$age)
```

```
##    Length     Class      Mode
##      1676 character character
```

```
sort(unique(raw_data[["age"]]))
```

```
##  [1] "18"          "19"          "20"          "21"          "22"
##  [6] "23"          "24"          "25"          "26"          "27"
## [11] "28"          "29"          "30"          "31"          "32"
## [16] "33"          "34"          "35"          "36"          "37"
## [21] "38"          "39"          "40"          "41"          "42"
## [26] "43"          "44"          "45"          "46"          "47"
## [31] "48"          "49"          "50"          "51"          "52"
## [36] "53"          "54"          "55"          "56"          "57"
## [41] "58"          "59"          "60"          "61"          "62"
## [46] "63"          "64"          "65"          "66"          "67"
## [51] "68"          "69"          "70"          "71"          "72"
## [56] "73"          "74"          "75"          "76"          "77"
## [61] "78"          "79"          "80"          "81"          "82"
## [66] "83"          "84"          "85"          "86"          "87"
## [71] "88"          "89"          "90"          "91"          "92"
## [76] "93"          "97 or older" "Refused"
```

So before we can work with age as a numeric quantity, we have to first strip out the values `97 or older` and `Refused`, and then convert the remaining string values to numeric.

Rather than enumerate the specific values to remove, it is cleaner to simply filter out any value that cannot be converted to numeric. `as.numeric` is a good engine for this process; any input that can't be converted to numeric returns `NA`, which can then be filtered. For efficiency, we make a new `data.frame` consisting of only the information we want (gender and age), named in sensible ways.

```
age_data <- raw_data %>%
  # create a new field with the numeric value of "age"
  mutate(numeric_age = suppressWarnings(as.numeric(age))) %>%
  # filter rows whose numeric conversion of "age" is "NA"
  filter(!is.na(numeric_age)) %>%
  # rename "d1" as "gender"
  rename(gender = d1) %>%
  # include only gender and age in the new frame.
  select(gender,numeric_age)

# See how many rows we removed.
removed_rows <- nrow(raw_data) - nrow(age_data)
removed_rows_pct = round((removed_rows / nrow(raw_data)) * 100, 2)
```

We removed 33 rows, or 1.97%, of the original data, because their age values could not be converted to numeric. The question of whether this is an acceptable threshold is not addressed here, except to observe that we are only testing here to see if age and gender correlate in the original data set; we have not removed any data from the original set, and will continue our investigation with it, not with this extract.

Now that the remaining age data is numeric, we can look at means and standard deviations of age by gender:

```
age_data %>%
  group_by(gender) %>%
  summarize(mean_age = mean(numeric_age), sd_age = sd(numeric_age))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 3
##   gender mean_age sd_age
##   <chr>     <dbl>  <dbl>
## 1 Female     52.6   18.8
## 2 Male       50.3   18.4
```

We have a similar SD for the genders, but the mean age for women is approximately 2.25 years higher than for men.

Running a T-test on gender and age compares the standard error of the means to see if there are statistically significant differences. Our null hypothesis H_0 would be that the means are equal, meaning there is no statistically significant difference between the age means by gender.

```
age_data %>%
    t.test(numeric_age ~ gender, data = .)
```

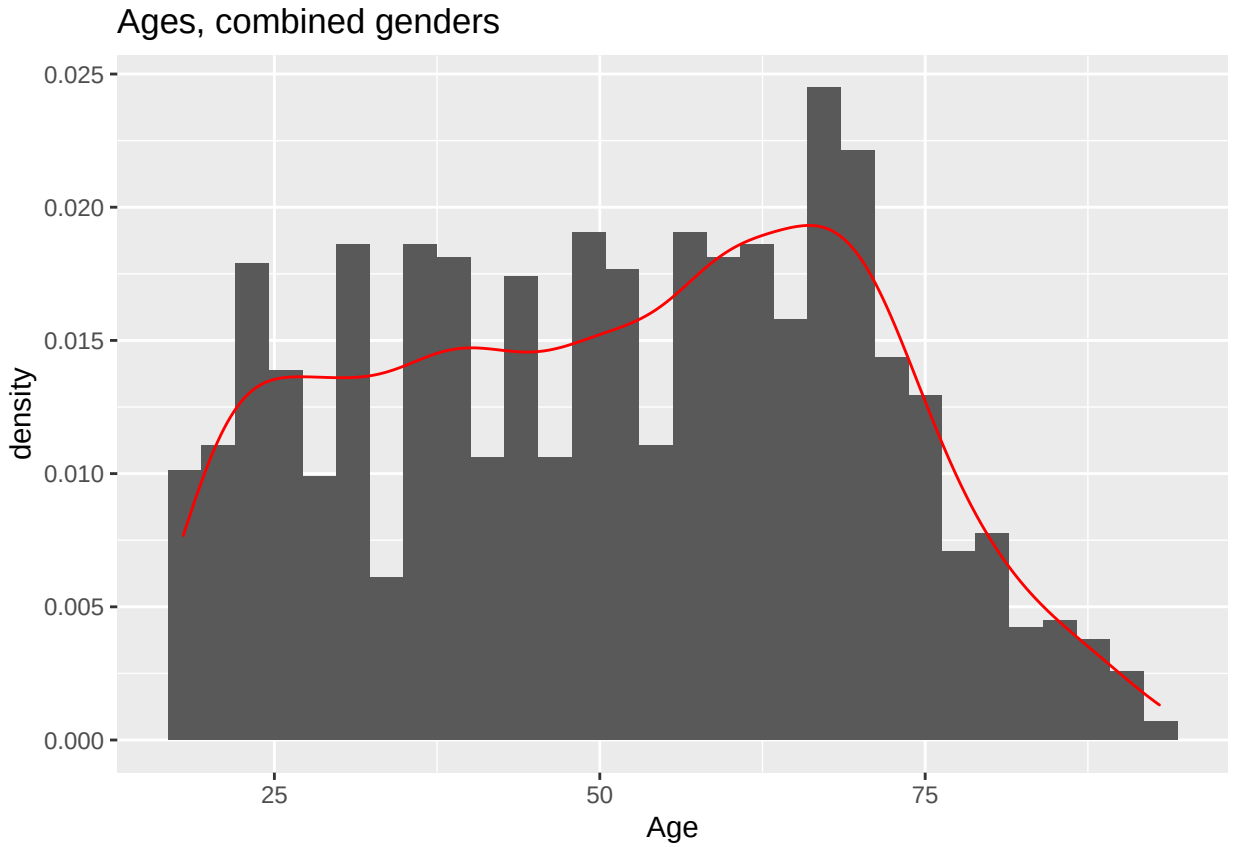
```
##
##  Welch Two Sample t-test
##
## data:  numeric_age by gender
## t = 2.4485, df = 1580, p-value = 0.01445
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.4498965 4.0738273
## sample estimates:
## mean in group Female   mean in group Male
##             52.56533             50.30347
```

The `p-value` of `0.014` shows that the null hypothesis - that the two distributions are equal in means, and that therefore age and gender are not in any way correlated in the data - should be rejected. The `t` value of `2.45` shows a significant difference in the error of the means.

Since we are seeing *some* correlation between age and gender in the survey data, let's get a look at it to help us understand it better. First, we can plot the distributions of ages, for both genders together and each individually, with a density line overlaid for clarity:

```
ggplot(age_data, aes(`numeric_age`)) +
  labs(title = "Ages, combined genders", x = "Age") +
  geom_histogram(aes(y = ..density..)) + geom_density(color="red")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

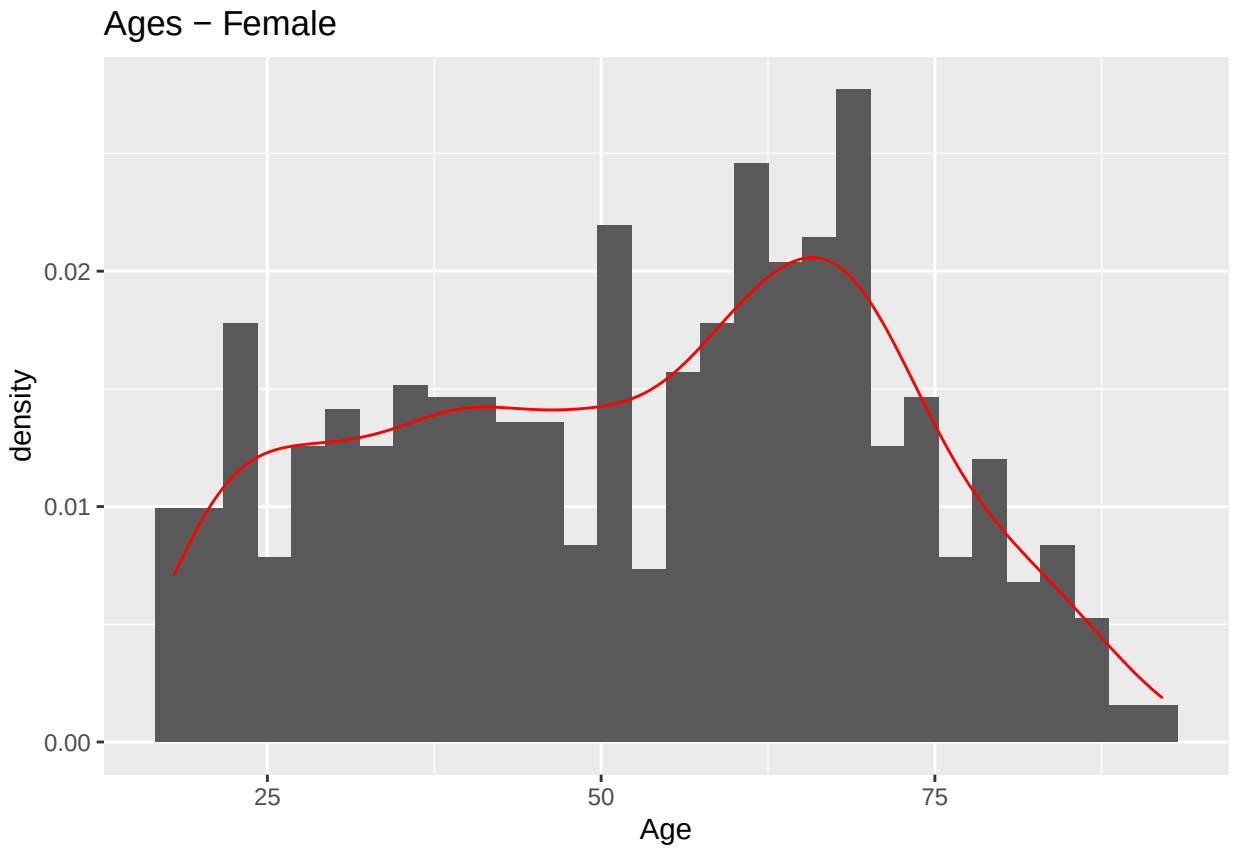



```
ggplot(filter(age_data, gender == "Male"), aes(`numeric_age`)) +
  labs(title = "Ages - Male", x = "Age") +
  geom_histogram(aes(y = ..density..)) + geom_density(color="red")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

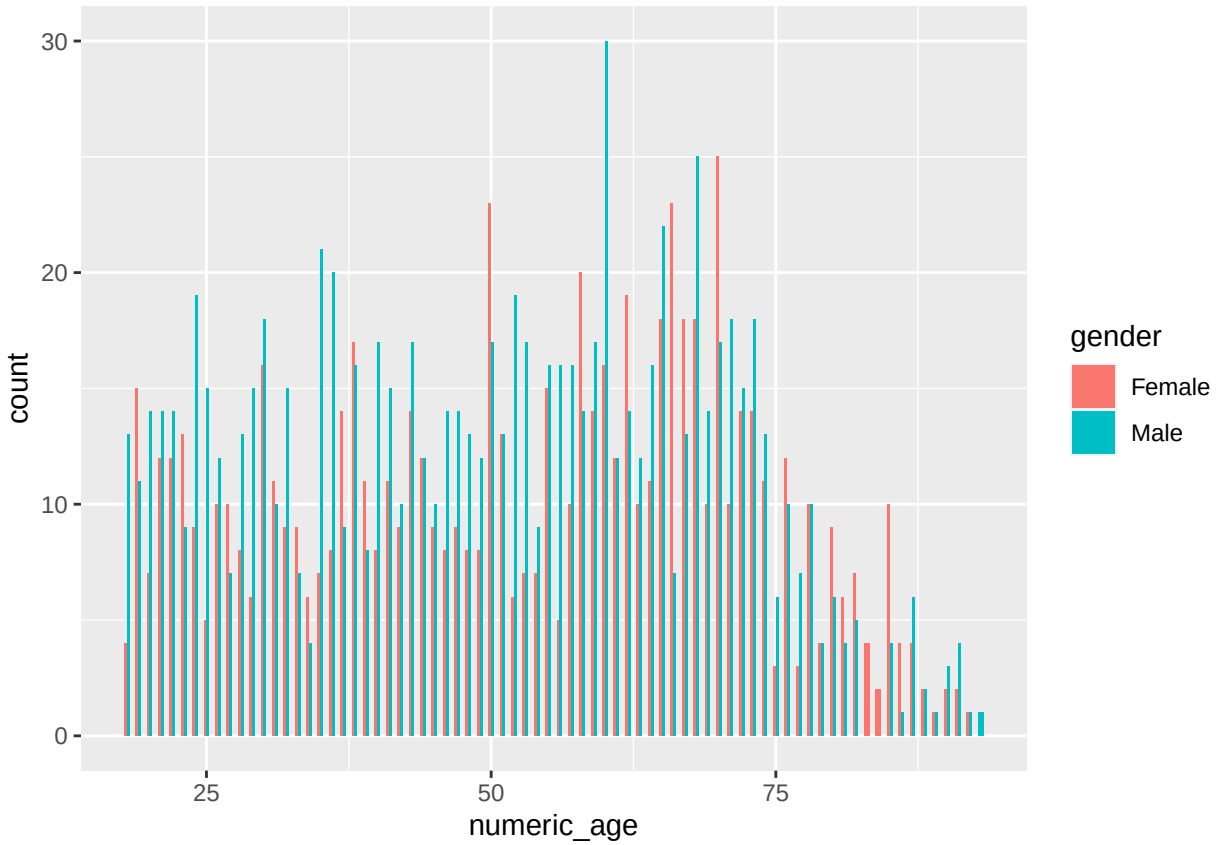

```
ggplot(filter(age_data, gender == "Female"), aes(`numeric_age`)) +
  labs(title = "Ages - Female", x = "Age") +
  geom_histogram(aes(y = ..density..)) + geom_density(color="red")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



From these plots, we can see that the age distribution does not appear to be normal, and that there is a visible difference in the distribution shapes, but the granularity of the results is a bit blocky and the results are hard to compare. It might be easier to get a sense of gender distribution by adding color and combining the genders into a single graph. The `dodge` parameter puts the colored bars for each age adjacent to each other:

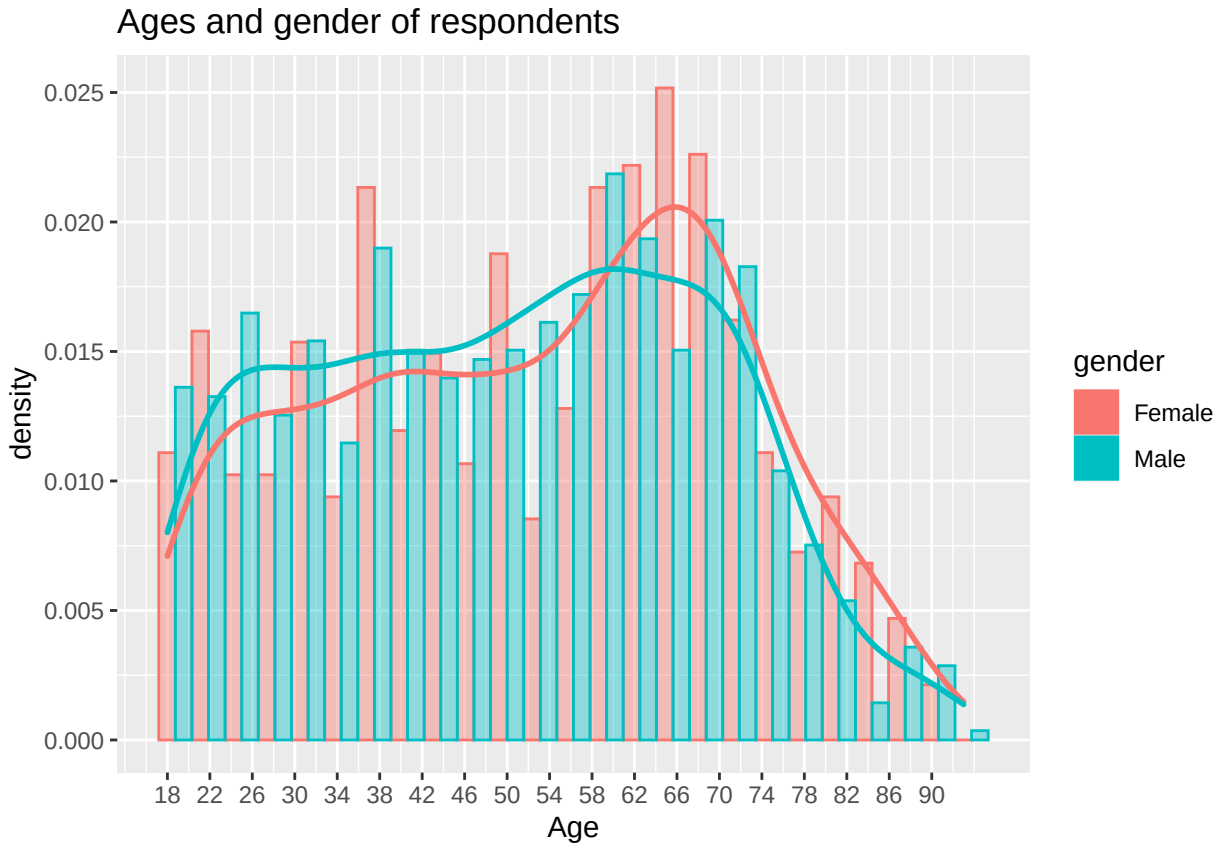
```
age_data %>%
  ggplot(aes(numeric_age, fill = gender), xlab="gender") +
  geom_bar(stat = "count", position = "dodge", width = 0.5)
```



This works, but is visually noisy, as the bars are very narrow due to their number; we are essentially treating `age` as a categorical, or possibly ordinal, variable in this case, which could be a matter of interpretation, but probably isn't its best representation here.

A density plot with fewer increments in the underlying histogram gives a much clearer view:

```
age_data %>%
  ggplot(aes(`numeric_age`)) +
  labs(title = "Ages and gender of respondents", x = "Age") +
  scale_x_continuous(breaks = round(seq(min(age_data$numeric_age), max(age_data$numeric_age), by = 4),1
  geom_histogram(aes(y = ..density.., color = gender, fill = gender),
                 alpha = 0.4, position = "dodge", bins=25) +
  geom_density(aes(color = gender), size =1)
```



From this visualization, we can draw the following conclusions:

1. The proportion of males in the survey is slightly higher than that of females until approximately age 60;
2. There is a significant spike in the representation of women beginning at approximately age 58, and the percentage of females in the 60-72 age range is considerably higher than that of males;
3. Women represent a consistently higher percentage of respondents for the remainder of the age range, to its maximum;
4. The ages of respondents, both combined and grouped by gender, appear not to follow normal distributions.

All of this is to be expected, to some degree; women in general live longer then men, so it's not surprising to see larger percentages of women at older ages, and depending on the study's selection methods (not discussed here), perfectly normal age distributions would be surprising to find.

Before deciding what to do about this skew in the data, it's worth investigating the question of whether our t.test was valid in the first place. Most of the literature on t.test (Wasserman, wikipedia) says that it is preferred for data that follows a normal distribution, and is less robust in non-normal distributions, notably with small samples. Considering the surprising amount of apparent academic disagreement on this subject, it's helpful to run a couple of additional tests to shed as much light as possible on the situation.

First, the observation that the data does not appear to follow a normal distribution is just that, an observation. We want to be certain that the distribution is not normal, and to not rely solely on a visual inspection. The Shapiro-Wilks test will calculate the probability of a distribution being normal; we can run it on the combined data, and on each gender's data individually:

```
shapiro.test(age_data$numeric_age)
```

```
##
##  Shapiro-Wilk normality test
##
```

```
## data:  age_data$numeric_age
## W = 0.96936, p-value < 2.2e-16
```

```
shapiro.test(filter(age_data, gender == "Male")$numeric_age)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  filter(age_data, gender == "Male")$numeric_age
## W = 0.97067, p-value = 1.986e-12
```

```
shapiro.test(filter(age_data, gender == "Female")$numeric_age)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  filter(age_data, gender == "Female")$numeric_age
## W = 0.96634, p-value = 4.082e-12
```

The `p-values` in the Shapiro-Wilk test show that both the combined distributions of `numeric_age`, and those grouped by gender, have vanishingly small chances of being normal. This confirms what we observed optically.

Second, in this case, we have 1643 observations: well above the 20-100 thresholds variously suggested as a minimum N for the t-test's validity, but another test, the Wilcoxon-Mann-Whitney U test, can be used in place of the t-test and is robust with non-normalized data. Once source suggests that it's a preferable drop-in replacement for t-tests in nearly all cases, except where N is very small (<6), or in certain circumstances where computational power might be limited.

```
wilcox.test(numeric_age ~ gender, data=age_data)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  numeric_age by gender
## W = 358628, p-value = 0.01314
## alternative hypothesis: true location shift is not equal to 0
```

As with the t-test, the Wincoxon test's `p-value` less than 0.05 shows that the means of the distributions are very unlikely to be equivalent, and that therefore we cannot be certain that gender-based inquiries in this data set will not contain hidden age biases. It's noteworthy that the `p-value`s for the two tests are very close: 0.01445 for the t-test, and 0.01314 for the Wincoxon test, suggesting that the t-test gave a robust result in this case even with the skews present in the data.

This does not solve the problem of what to do about the skew. The dataset being used here does contain a weighting factor, and this will be explored in the final paper.

References

Wasserman, L. (2004). *All of Statistics: A Concise Course in Statistical Inference.* New York: Springer. ISBN: 978-1-4419-2322-6, p.170