Gender and authorship in Annals of Surgery: A fourteen year review

Marcello Chang, Jane W. Liang, Sharon Stein, and Arghavan Salles

2023-02-15

```
knitr::opts_chunk$set(message = FALSE, warning = FALSE)

# Data wrangling
library(tidyverse)
library(lubridate)

# Plotting
library(reshape2)
library(forcats)
library(ggrepel)
library(gridExtra)

# Robust SEs
library(sandwich)
library(lmtest)

# Marginal means
library(emmeans)

# Set background to be white for all ggplots
theme_set(theme_classic())
```

Data pre-processing

The data read in below is pulled from `annals_surg?figs_June_25_0_9.ipynb`. We performed some minor data pre-processing to get it R-ready.

```
# Read in data (includes grants)
load("../new_data/new_data.rData")
df_original = article_df

# Set blank genders to NA and convert to factor
df_original$first_author_gender[df_original$first_author_gender == ""] = NA
df_original$first_author_gender = factor(df_original$first_author_gender,
                                         levels = c("male", "female"),
                                         labels = c("Man", "Woman"))
df_original$last_author_gender[df_original$last_author_gender == ""] = NA
df_original$last_author_gender = factor(df_original$last_author_gender,
                                         levels = c("male", "female"),
                                         labels = c("Man", "Woman"))

# Create a variable to indicate that all articles were submitted
```

```r
df_original$submitted = 1
# Pandemic defined as on or after March 24, 2023
# Also create cutpoint for unknown speculated editorial process change that
# greatly reduces time to decision. Unclear exactly where it should be, but
# change may have occurred as early as 2013
df_original = df_original %>%
  mutate(time_periods = factor(case_when(
    # Pre-pandemic, before discontinuity
    first_receipt_date < as.Date("2014-01-01") ~ "1",
    # Pre-pandemic, after discontinuity
    first_receipt_date >= as.Date("2014-01-01") &
      first_receipt_date < as.Date("2020-03-24") ~ "2",
    # Pandemic
    first_receipt_date >= as.Date("2020-03-24") ~ "3")))
```

We used string processing on the author names to make it easier to match the same author to him/herself:

1. Trim white space
2. Attempt to remove degrees and suffixes at the end of the name (everything after the comma)
3. Attempt to remove middle initials by removing everything between the first name (characters that come after the first white space) and the last name (characters that come before the last white space)
4. Make everything lowercase, so that the names are not case-sensitive

```r
# Remove white space in author names and convert to factor
# Do some string processing (remove degrees, middle initials, case) to help
# match the same author by name
df_original$first_author =
  as.factor(tolower(sapply(strsplit(gsub("(.*),.*", "\\1",
                                         trimws(df_original$first_author)),
                                    split = " +"), function(x){
                                      paste(x[1], x[length(x)])
                                    })))
df_original$last_author =
  as.factor(tolower(sapply(strsplit(gsub("(.*),.*", "\\1",
                                         trimws(df_original$last_author)),
                                    split = " +"), function(x){
                                      paste(x[1], x[length(x)])
                                    })))
```

Authors with multiple manuscripts

One challenging aspect for analyzing this dataset is the fact that the same author can submit multiple manuscripts to the journal. Unique authors are identified based on name, so if the same person publishes under different names or multiple authors have the same name in this dataset, we may not pick up on that. The string processing in the last section should hopefully reduce inconsistencies in names.

Among articles where the first/last author's gender was assigned, about 22-30% of first and last authors submitted multiple manuscripts to the journal. About 55% of manuscripts were written by first authors with multiple manuscripts, and about 66% were written by last authors with multiple manuscripts. In general, fewer female authors appear to submit multiple articles and a lower percent of articles written by first/last authors with multiple articles have a female first/last author.

```r
# Tabulate articles by author
first_author_tab = df_original %>%
  drop_na(first_author_gender) %>%
```

```
  group_by(first_author) %>%
  summarize(first_author_gender = first(first_author_gender), n = n()) %>%
  ungroup()
last_author_tab = df_original %>%
  drop_na(last_author_gender) %>%
  group_by(last_author) %>%
  summarize(last_author_gender = first(last_author_gender), n = n()) %>%
  ungroup()

rbind("Percent of first authors with more than one article" =
        c(tapply(first_author_tab$n > 1,
                 first_author_tab$first_author_gender, mean),
          Overall = mean(first_author_tab$n > 1)),
      "Percent of last authors with more than one article" =
        c(tapply(last_author_tab$n > 1,
                 last_author_tab$last_author_gender, mean),
          Overall = mean(last_author_tab$n > 1))) %>% round(3)
```

```
##                                                    Man Woman Overall
## Percent of first authors with more than one article 0.291 0.217   0.271
## Percent of last authors with more than one article  0.316 0.228   0.299
```

```
rbind("Percent of articles written by first authors with more than one article" =
        c(first_author_tab %>%
            group_by(first_author_gender) %>%
            summarize(prop = sum(n[n>1]) / sum(n)) %>%
            pull(prop) %>% set_names(c("Man", "Woman")),
          Overall = sum(first_author_tab$n[first_author_tab$n > 1]) /
            sum(first_author_tab$n)),
      "Percent of articles written by last authors with more than one article" =
        c(last_author_tab %>%
            group_by(last_author_gender) %>%
            summarize(prop = sum(n[n>1]) / sum(n)) %>%
            pull(prop) %>% set_names(c("Man", "Woman")),
          Overall = sum(last_author_tab$n[last_author_tab$n > 1]) /
            sum(last_author_tab$n))) %>% round(3)
```

```
##                                                                    Man
## Percent of articles written by first authors with more than one article 0.584
## Percent of articles written by last authors with more than one article  0.685
##                                                                  Woman
## Percent of articles written by first authors with more than one article 0.443
## Percent of articles written by last authors with more than one article  0.542
##                                                                Overall
## Percent of articles written by first authors with more than one article   0.550
## Percent of articles written by last authors with more than one article    0.663
```

After dropping articles that were submitted by first or last authors with multiple articles, 6126 articles remain. This is about 23% of the total number of articles (restricted to those where the first and last authors' genders were assigned).

```
# Data frame without first or authors with multiple articles
df_unique_both = df_original[df_original$first_author %in%
                               first_author_tab$first_author[first_author_tab$n == 1] &
                               df_original$last_author %in%
```

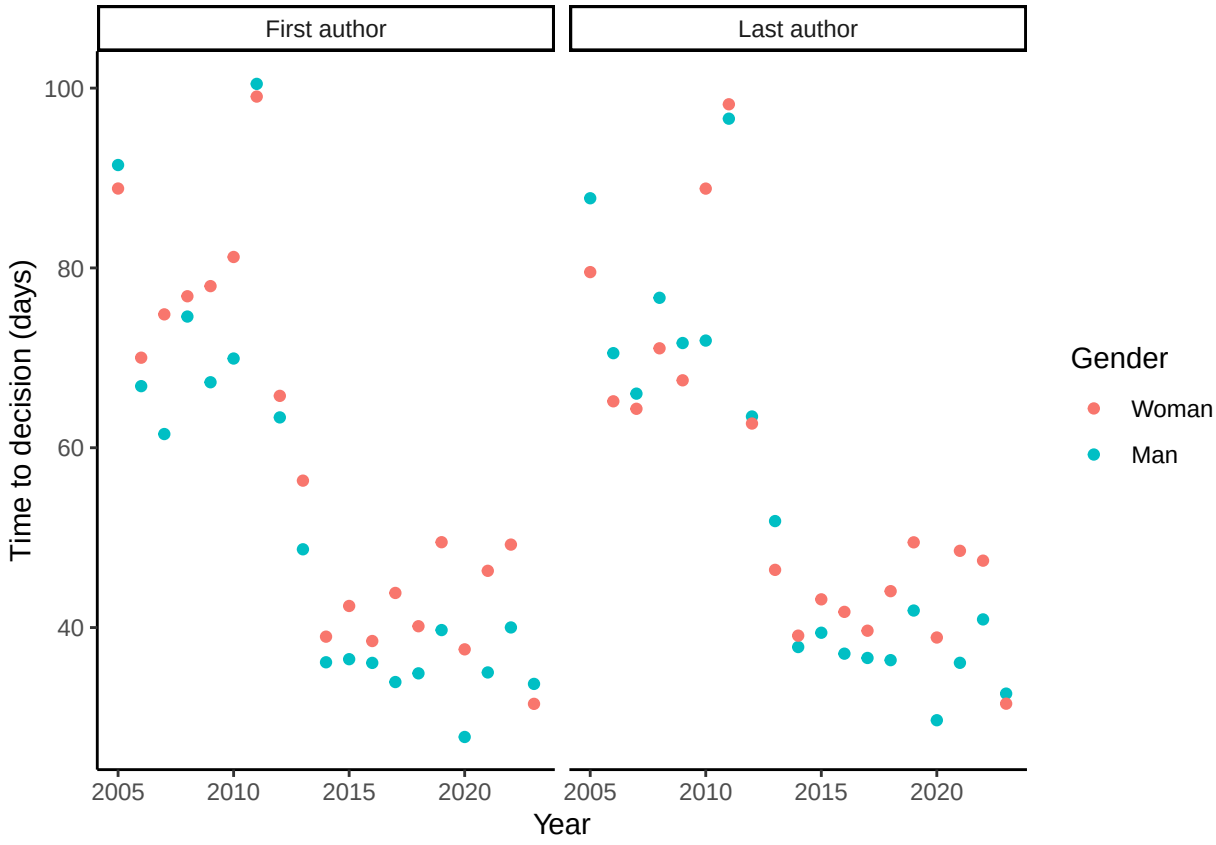
```
                                last_author_tab$last_author[last_author_tab$n == 1],]

print(paste("Percent of articles remaining:",
            round(nrow(df_unique_both) /
              nrow(df_original[!is.na(df_original$first_author_gender) &
                                 !is.na(df_original$last_author_gender),]), 3)))
```

```
## [1] "Percent of articles remaining: 0.227"
```

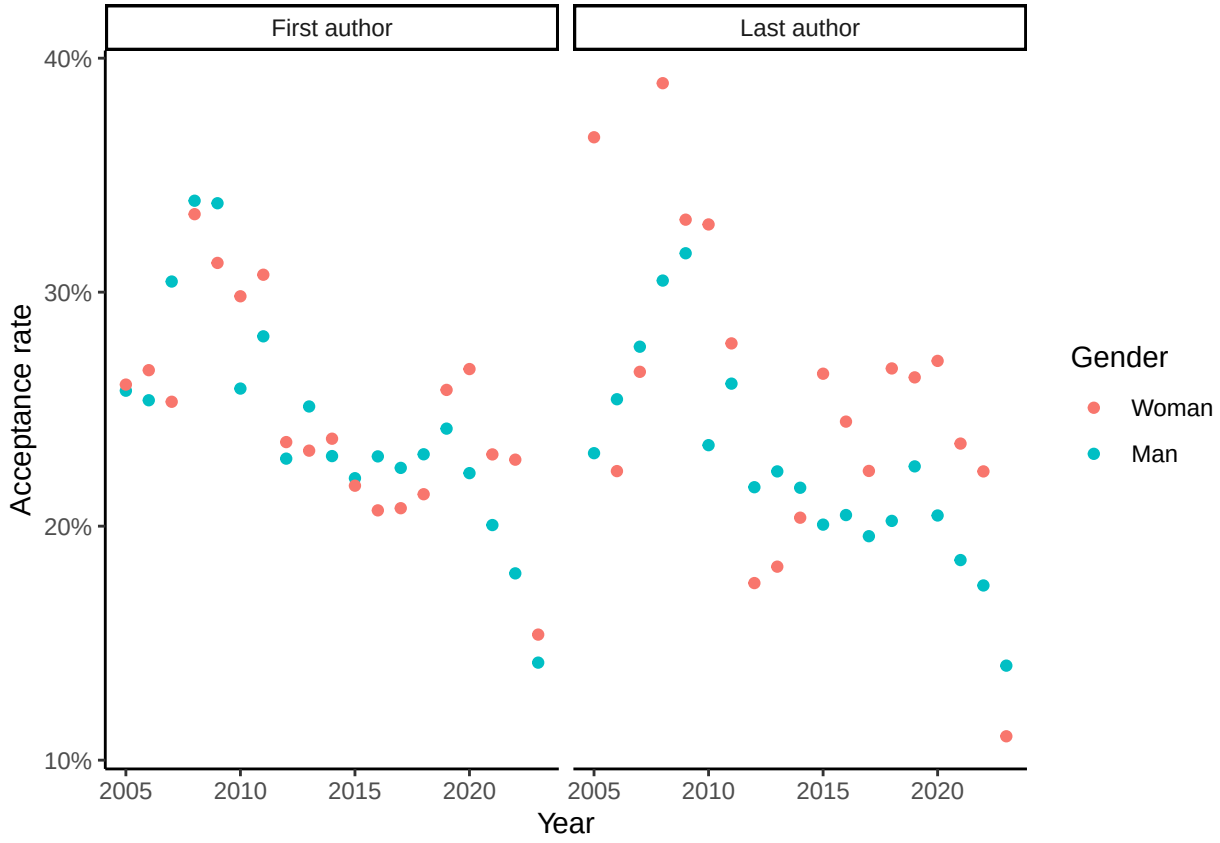
Plots of average timee to decision and acceptance rates over time, by gender.

```
df_original %>%
  pivot_longer(cols = c("first_author_gender", "last_author_gender"),
               names_to = "author_type",values_to = "gender",
               values_drop_na = TRUE) %>%
  group_by(receipt_year, author_type, gender) %>%
  summarize(ttd = mean(t_delta, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(gender = factor(gender, levels = c("Woman", "Man"))) %>%
  ggplot(aes(x = receipt_year, y = ttd, color = gender)) +
  geom_point() +
  facet_grid(~author_type,
             labeller = as_labeller(c(
               "first_author_gender" = "First author",
               "last_author_gender" = "Last author"))) +
  xlab("Year") + ylab("Time to decision (days)") +
  scale_color_discrete(name = "Gender")
```



```r
ggsave("ttd_over_time.png")

df_original %>%
  pivot_longer(cols = c("first_author_gender", "last_author_gender"),
               names_to = "author_type",values_to = "gender",
               values_drop_na = TRUE) %>%
  group_by(receipt_year, author_type, gender) %>%
  summarize(acc_rate = mean(accepted, na.rm = TRUE)) %>%
  ungroup() %>%
  mutate(gender = factor(gender, levels = c("Woman", "Man"))) %>%
  ggplot(aes(x = receipt_year, y = acc_rate, color = gender)) +
  geom_point() +
  facet_grid(~author_type,
             labeller = as_labeller(c(
               "first_author_gender" = "First author",
               "last_author_gender" = "Last author"))) +
  xlab("Year") + ylab("Acceptance rate") +
  scale_color_discrete(name = "Gender") +
  scale_y_continuous(labels = scales::percent)
```




```r
ggsave("acc_rate_over_time.png")
```

Table 1

Below are the number of first/last author genders that were successfully assigned by genderize.io.

```r
with(df_original, {
  rbind(First = c(table(first_author_gender),
                  Total = sum(!is.na(first_author_gender))),
        Last = c(table(last_author_gender),
                 Total = sum(!is.na(last_author_gender))))
})
```

```
##         Man Woman Total
## First 24661  7874 32535
## Last  26462  4810 31272
```

We create a "Table 1" with the number of accepted manuscripts, number of submitted manuscripts, and percent accepted, by year and author gender.

```r
tab1 = with(df_original, {
  rbind(
    cbind(
      # First author
      tapply(accepted, list(receipt_year, first_author_gender), sum),
      tapply(submitted, list(receipt_year, first_author_gender), sum),
      round(tapply(accepted, list(receipt_year, first_author_gender), mean), 4)*100,
      # Last author
      tapply(accepted, list(receipt_year, last_author_gender), sum),
      tapply(submitted, list(receipt_year, last_author_gender), sum),
      round(tapply(accepted, list(receipt_year, last_author_gender), mean), 4)*100,
      # Overall
      tapply(accepted, receipt_year, sum),
      tapply(submitted, receipt_year, sum),
      round(tapply(accepted, receipt_year, mean), 4)*100),

    # Total
    Total = c(# First author
      tapply(accepted, first_author_gender, sum),
      tapply(submitted, first_author_gender, sum),
      round(tapply(accepted, first_author_gender, mean), 4)*100,
      # Last author
      tapply(accepted, last_author_gender, sum),
      tapply(submitted, last_author_gender, sum),
      round(tapply(accepted, last_author_gender, mean), 4)*100,
      # Overall
      sum(accepted),
      sum(submitted),
      round(mean(accepted), 3)*100)
    )
  })

tab1 = rbind(c("First author", "", "", "", "", "",
               "Last author", "", "", "", "", "",
               "Overall", "", ""),
             c(rep(c("Accepted (n)", "", "Submitted (n)", "",
                     "Accepted (%)", ""), 2),
               c("Accepted (n)", "Submitted (n)", "Accepted (%)")),
             c(rep(c("Man", "Woman"), 6), "", "", ""),
             tab1)
write.csv(tab1, file = "tab1.csv")
```

The proportion of female authors goes up for both first authors and last authors after dropping articles written by authors with multiple articles. It appears that men are more likely to be authors of multiple submissions.

```
round(rbind(original = df_original %>%
              summarize(first_author = mean(first_author_gender=="Woman", na.rm = TRUE),
                        last_author = mean(last_author_gender=="Woman", na.rm = TRUE)),
            unique = df_unique_both %>%
              summarize(first_author = mean(first_author_gender=="Woman"),
                        last_author = mean(last_author_gender=="Woman"))), 3)
```

```
## # A tibble: 2 x 2
##   first_author last_author
##          <dbl>       <dbl>
## 1        0.242       0.154
## 2        0.287       0.21
```

This trend also holds among accepted manuscripts.

```
round(rbind(original = df_original %>% subset(accepted == TRUE) %>%
              summarize(first_author = mean(first_author_gender=="Woman", na.rm = TRUE),
                        last_author = mean(last_author_gender=="Woman", na.rm = TRUE)),
            unique = df_unique_both %>% subset(accepted == TRUE) %>%
              summarize(first_author = mean(first_author_gender=="Woman"),
                        last_author = mean(last_author_gender=="Woman"))), 3)
```

```
## # A tibble: 2 x 2
##   first_author last_author
##          <dbl>       <dbl>
## 1        0.243       0.17
## 2        0.306       0.244
```

Overview of approaches

For each of the analyses in the paper, we tried up to three modeling approaches.

1. When feasible, we applied robust standard errors to linear or logistic regression models. We typically used clustered standard errors where the clusters are defined by the first and last authors.
2. As far as we are aware, robust standard errors are not available for contingency tables. In such cases (where a contingency table tests cannot be coerced into a regression model), we used the usual chi-squared test. The issue with this approach is that it treats each article as an independent unit, even though this independence assumption is likely violated because authors can submit multiple articles.
3. When robust standard errors cannot be run for contingency tables, we attempted a secondary analysis that excludes articles written by first/last authors with multiple articles. The assumption here is that the first and last authors create the dependency between articles (and that we don't have to worry about the middle authors or relationships between authors). So if we drop articles that were submitted by first or last authors with multiple articles, we can uphold the independence assumption. However, this results in a large amount of data loss, which may lead to underpowering. Moreover, authors with multiple outcomes may be different from authors with only one article in other ways: for example, more men are authors of multiple articles.

We collected all of the p-values from the analyses in this document into a single vector. At the end, we used the Benjamini-Hochberg method to adjust these p-values for multiple comparisons.

```
all_pvals = c()
```

Number of submissions

Rate of change in number of submissions

This analysis looks at the interaction between year and first and/or last author gender in a linear regression model predicting the number of manuscripts submitted. An indicator for years 2020 and beyond (coinciding with the COVID-19 pandemic) is also included as an interaction. We used robust standard errors, but not clustered standard errors in this case, since the observations fed into the regression model are summarized by year/author gender and cannot be clustered by author. We can repeat this exercise for last authors and first and last author gender pairs.

```
# General function to fit linear models of the form
# outcome ~ author_gender * year * (year >= 2020)
change_over_time = function(df, outcome, author_gender) {
  outcome_tab = tapply(df[[outcome]],
                       list(df$receipt_year, df[[author_gender]]), sum)
  mod_df = data.frame(
    author_gender = relevel(as.factor(rep(c("Man", "Woman"),
                                          each = nrow(outcome_tab))),
                            ref = "Man"),
    year = rep(as.numeric(rownames(outcome_tab)), 2),
    outcome = as.numeric(outcome_tab))
  mod_df$year_post2020 = mod_df$year >= 2020
  mod_df = mod_df[mod_df$year != 2023,]
  fit_lm = lm(outcome ~ author_gender*year*year_post2020,
              data = mod_df)
  return(list(fit = fit_lm, df = outcome_tab))
}

# General function to fit linear models of the form
# outcome ~ first_author_gender * last_author_gender * year * (year >= 2020)
change_over_time_author_interaction = function(df, outcome,
                                               ref_first = "Man",
                                               ref_last = "Man") {
  outcome_tab = tapply(df[[outcome]],
                       list(df$receipt_year,
                            interaction(df$first_author_gender,
                                        df$last_author_gender)), sum)
  mod_df = data.frame(
    first_author_gender =
      relevel(as.factor(rep(rep(c("Man", "Woman"),
                                each = nrow(outcome_tab)), 2)),
              ref = ref_first),
    last_author_gender =
      relevel(as.factor(rep(c("Man", "Woman"),
                            each = 2*nrow(outcome_tab))),
              ref = ref_last),
    year = rep(as.numeric(rownames(outcome_tab)), 4),
    outcome = as.numeric(outcome_tab))
  mod_df$author_gender_pair = interaction(mod_df$first_author_gender,
                                          mod_df$last_author_gender, sep = " ")
  mod_df$year_post2020 = mod_df$year >= 2020
  mod_df = mod_df[mod_df$year != 2023,]
  fit_lm = lm(outcome ~ author_gender_pair*year*year_post2020, data = mod_df)
  return(list(fit = fit_lm, df = outcome_tab))
```

```
}

# First authors
sub_rate_first_lm = change_over_time(df_original,
                                     "submitted", "first_author_gender")
sub_rate_first_robust =
  emtrends(sub_rate_first_lm$fit,
           pairwise ~ author_gender | year_post2020,
           vcov = vcovHC, var = "year",
           adjust = "none")

# Last authors
sub_rate_last_lm = change_over_time(df_original,
                                    "submitted", "last_author_gender")
sub_rate_last_robust =
  emtrends(sub_rate_last_lm$fit,
           pairwise ~ author_gender | year_post2020,
           vcov = vcovHC, var = "year",
           adjust = "none")

# First-last author pairs
sub_rate_pair_lm = change_over_time_author_interaction(df_original,
                                                       "submitted")
sub_rate_pair_robust =
  emtrends(sub_rate_pair_lm$fit,
           pairwise ~ author_gender_pair | year_post2020,
           vcov = vcovHC, var = "year",
           adjust = "none")

# Extract relevant contrasts
sub_rate_pair_tab = data.frame(sub_rate_pair_robust$contrasts)
rownames(sub_rate_pair_tab) =
  paste(rep(c("(2005-2019)", "(2020-2022)"), each = 6),
        sub_rate_pair_tab$contrast)
sub_rate_pair_pvals = sub_rate_pair_tab[,"p.value"]
names(sub_rate_pair_pvals) = paste("sub_rate_pair", rownames(sub_rate_pair_tab))

# Collect p-values
all_pvals = c(
  all_pvals,
  setNames(data.frame(sub_rate_first_robust$contrasts)[,"p.value"],
           paste("sub_rate_first", c("pre", "post"))),
  setNames(data.frame(sub_rate_last_robust$contrasts)[,"p.value"],
           paste("sub_rate_last", c("pre", "post"))),
  sub_rate_pair_pvals
  )
```

These are the trend etsimates for first authors, last authors, and author pairs, with robust standard errors.

```
# First authors
sub_rate_first_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman     31.9   4.85 28   6.583  <.0001
```

```
##
## year_post2020 =  TRUE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman   -425.5 333.55 28  -1.276  0.2125
```

```
sub_rate_first_robust$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man                 67.4   4.59 28     58.0     76.9
##  Woman               35.5   1.54 28     32.4     38.7
##
## year_post2020 =  TRUE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man               -636.0 330.93 28  -1313.9     41.9
##  Woman             -210.5  41.72 28   -296.0   -125.0
##
## Confidence level used: 0.95
```

```
# Last authors
sub_rate_last_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman     59.1   4.94 28  11.972  <.0001
##
## year_post2020 =  TRUE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman   -580.0 432.76 28  -1.340  0.1909
```

```
sub_rate_last_robust$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man                 80.6   4.83 28     70.7     90.5
##  Woman               21.5   1.02 28     19.4     23.5
##
## year_post2020 =  TRUE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man               -695.0 432.75 28  -1581.4    191.4
##  Woman             -115.0   2.83 28   -120.8   -109.2
##
## Confidence level used: 0.95
```

```
# First-last author pairs
sub_rate_pair_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast                estimate     SE df t.ratio p.value
##  Man Man - Woman Man         21.8   4.02 56   5.439  <.0001
##  Man Man - Man Woman         33.5   3.65 56   9.185  <.0001
##  Man Man - Woman Woman       35.4   3.74 56   9.489  <.0001
##  Woman Man - Man Woman       11.7   1.89 56   6.195  <.0001
##  Woman Man - Woman Woman     13.6   2.04 56   6.652  <.0001
##  Man Woman - Woman Woman      1.9   1.18 56   1.614  0.1122
##
## year_post2020 =  TRUE:
```

```
##  contrast                estimate     SE df t.ratio p.value
##  Man Man - Woman Man       -319.5 334.79 56  -0.954  0.3440
##  Man Man - Man Woman       -391.0 330.22 56  -1.184  0.2414
##  Man Man - Woman Woman     -409.0 330.24 56  -1.239  0.2207
##  Woman Man - Man Woman      -71.5  55.16 56  -1.296  0.2002
##  Woman Man - Woman Woman    -89.5  55.27 56  -1.619  0.1110
##  Man Woman - Woman Woman    -18.0   3.61 56  -4.992  <.0001
```

```
sub_rate_pair_robust$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender_pair year.trend      SE df lower.CL upper.CL
##  Man Man                 43.80   3.599 56    36.59     51.0
##  Woman Man               21.96   1.783 56    18.39     25.5
##  Man Woman               10.26   0.622 56     9.02     11.5
##  Woman Woman              8.36   1.000 56     6.36     10.4
##
## year_post2020 =  TRUE:
##  author_gender_pair year.trend      SE df lower.CL upper.CL
##  Man Man               -455.50 330.219 56 -1117.01    206.0
##  Woman Man             -136.00  55.154 56  -246.49    -25.5
##  Man Woman              -64.50   0.707 56   -65.92    -63.1
##  Woman Woman            -46.50   3.535 56   -53.58    -39.4
##
## Confidence level used: 0.95
```

Here, we restrict to unique first, last, and first-last author pairs for each year's submissions.

```
# Unique authors
# First authors
sub_rate_first_lm2 =
  change_over_time(df_original %>%
                     select(submitted, first_author_gender,
                            first_author, receipt_year) %>%
                     unique(), "submitted", "first_author_gender")
sub_rate_first_robust2 =
    emtrends(sub_rate_first_lm2$fit,
          pairwise ~ author_gender | year_post2020,
          vcov = vcovHC, var = "year",
          adjust = "none")

# Last authors
sub_rate_last_lm2 =
  change_over_time(df_original %>%
                     select(submitted, last_author_gender,
                            last_author, receipt_year) %>%
                     unique(), "submitted", "last_author_gender")
sub_rate_last_robust2 =
    emtrends(sub_rate_last_lm2$fit,
          pairwise ~ author_gender | year_post2020,
          vcov = vcovHC, var = "year",
          adjust = "none")

# First-last author pairs
sub_rate_pair_lm2 = change_over_time_author_interaction(
  df_original %>% select(submitted, first_author_gender, first_author,
```

```
                        last_author_gender, last_author, receipt_year) %>%
                    unique(), "submitted")
sub_rate_pair_robust2 =
  emtrends(sub_rate_pair_lm2$fit,
           pairwise ~ author_gender_pair | year_post2020,
           vcov = vcovHC, var = "year",
           adjust = "none")

# Extract relevant contrasts
sub_rate_pair_tab2 = data.frame(sub_rate_pair_robust2$contrasts)
rownames(sub_rate_pair_tab2) =
  paste(rep(c("(2005-2019)", "(2020-2022)"), each = 6),
        sub_rate_pair_tab2$contrast)
sub_rate_pair_pvals2 = sub_rate_pair_tab2[,"p.value"]
names(sub_rate_pair_pvals2) = paste("sub_rate_pair2", rownames(sub_rate_pair_tab2))

# Collect p-values
all_pvals = c(
  all_pvals,
  setNames(data.frame(sub_rate_first_robust2$contrasts)[,"p.value"],
           paste("sub_rate_first2", c("pre", "post"))),
  setNames(data.frame(sub_rate_last_robust2$contrasts)[,"p.value"],
           paste("sub_rate_last2", c("pre", "post"))),
  sub_rate_pair_pvals2
  )

# First authors
sub_rate_first_robust2$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman     23.6   3.68 28   6.401  <.0001
##
## year_post2020 =  TRUE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman   -328.5 268.73 28  -1.222  0.2317
```

```
sub_rate_first_robust2$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man                55.7   3.46 28     48.6     62.8
##  Woman              32.2   1.26 28     29.6     34.7
##
## year_post2020 =  TRUE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man              -511.5 266.58 28  -1057.6     34.6
##  Woman            -183.0  33.94 28   -252.5   -113.5
##
## Confidence level used: 0.95
```

```
# Last authors
sub_rate_last_robust2$contrasts
```

```
## year_post2020 = FALSE:
```

```
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman     38.3   3.65 28  10.480  <.0001
##
## year_post2020 =  TRUE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman   -396.0 303.39 28  -1.305  0.2024
```

```
sub_rate_last_robust2$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend      SE df lower.CL upper.CL
##  Man                55.6   3.590 28     48.2     62.9
##  Woman              17.3   0.664 28     15.9     18.7
##
## year_post2020 =  TRUE:
##  author_gender year.trend      SE df lower.CL upper.CL
##  Man              -494.5 303.349 28  -1115.9    126.9
##  Woman             -98.5   4.950 28   -108.6    -88.4
##
## Confidence level used: 0.95
```

```
# First-last author pairs
sub_rate_pair_robust2$contrasts
```

```
## year_post2020 = FALSE:
##  contrast                estimate     SE df t.ratio p.value
##  Man Man - Woman Man        19.29   3.55 56   5.436  <.0001
##  Man Man - Man Woman        30.40   3.27 56   9.298  <.0001
##  Man Man - Woman Woman      31.84   3.37 56   9.455  <.0001
##  Woman Man - Man Woman      11.10   1.60 56   6.927  <.0001
##  Woman Man - Woman Woman    12.55   1.80 56   6.989  <.0001
##  Man Woman - Woman Woman     1.45   1.15 56   1.262  0.2123
##
## year_post2020 =  TRUE:
##  contrast                estimate     SE df t.ratio p.value
##  Man Man - Woman Man      -285.00 295.85 56  -0.963  0.3395
##  Man Man - Man Woman      -348.50 292.31 56  -1.192  0.2382
##  Man Man - Woman Woman    -364.50 292.04 56  -1.248  0.2172
##  Woman Man - Man Woman     -63.50  49.06 56  -1.294  0.2008
##  Woman Man - Woman Woman   -79.50  47.40 56  -1.677  0.0991
##  Man Woman - Woman Woman   -16.00  12.81 56  -1.249  0.2167
```

```
sub_rate_pair_robust2$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender_pair year.trend      SE df lower.CL upper.CL
##  Man Man                39.87   3.218 56    33.42     46.3
##  Woman Man              20.57   1.497 56    17.58     23.6
##  Man Woman               9.47   0.574 56     8.32     10.6
##  Woman Woman             8.03   0.992 56     6.04     10.0
##
## year_post2020 =  TRUE:
##  author_gender_pair year.trend      SE df lower.CL upper.CL
##  Man Man              -408.50 292.035 56  -993.52    176.5
##  Woman Man            -123.50  47.376 56  -218.41    -28.6
##  Man Woman             -60.00  12.728 56   -85.50    -34.5
```

```
##  Woman Woman            -44.00   1.414 56   -46.83    -41.2
##
## Confidence level used: 0.95
```

We re-run the analysis for first and last author gender, but normalize the outcome (number of submissions by gender) by the total number of board-certified surgeons by gender for that year.

```
# Read in number of certified surgeons each year, by gender
gen_surg_df = read.csv("../CERTIFIED SURGEONS BY GENDER.csv", skip = 1)
# Some data processing
colnames(gen_surg_df) = c("year", "author_gender", "gen_surg")
gen_surg_df = gen_surg_df %>%
  mutate(author_gender = relevel(as.factor(
    ifelse(author_gender == "Female", "Woman", "Man")), ref = "Man")) %>%
  filter(year >=2005 & year <= 2022)


# General function to fit linear models of the form
# outcome/num_surgeons ~ author_gender * year
change_over_time_gen_surg = function(df, outcome, author_gender,
                                     gs_df = gen_surg_df) {
  outcome_tab = tapply(df[[outcome]],
                       list(df$receipt_year, df[[author_gender]]), sum)
  mod_df = data.frame(
    author_gender = relevel(as.factor(rep(c("Man", "Woman"),
                                          each = nrow(outcome_tab))),
                            ref = "Man"),
    year = rep(as.numeric(rownames(outcome_tab)), 2),
    outcome = as.numeric(outcome_tab))
  mod_df$year_post2020 = mod_df$year >= 2020
  mod_df = mod_df[mod_df$year != 2023,]
  mod_df = merge(mod_df, gs_df, by = c("year", "author_gender"))
  fit_lm = lm(outcome/gen_surg ~ author_gender*year*year_post2020,
              data = mod_df)

  outcome_tab = outcome_tab /
    (pivot_wider(gs_df, names_from = author_gender, values_from = gen_surg) %>%
       column_to_rownames("year") %>%
       select(Man, Woman))
  return(list(fit = fit_lm, df = as.matrix(outcome_tab)))
}

# First authors
sub_rate_gs_first_lm = change_over_time_gen_surg(df_original,
                                                 "submitted",
                                                 "first_author_gender")
sub_rate_gs_first_robust =
    emtrends(sub_rate_gs_first_lm$fit,
          pairwise ~ author_gender | year_post2020,
          vcov = vcovHC, var = "year",
          adjust = "none")

# Last authors
sub_rate_gs_last_lm = change_over_time_gen_surg(df_original,
                                                "submitted",
```

```
                                                "last_author_gender")
sub_rate_gs_last_robust =
    emtrends(sub_rate_gs_last_lm$fit,
          pairwise ~ author_gender | year_post2020,
          vcov = vcovHC, var = "year",
          adjust = "none")

# Collect p-values
all_pvals = c(
  all_pvals,
  setNames(data.frame(sub_rate_gs_first_robust$contrasts)[,"p.value"],
           paste("sub_rate_gs_first", c("pre", "post"))),
  setNames(data.frame(sub_rate_gs_last_robust$contrasts)[,"p.value"],
           paste("sub_rate_gs_last", c("pre", "post")))
  )


# First authors
sub_rate_gs_first_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman   0.0525 0.0159 28   3.297  0.0027
##
## year_post2020 =  TRUE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman  -0.8736 2.2069 28  -0.396  0.6952
```

```
sub_rate_gs_first_robust$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man               0.1081 0.0120 28   0.0835   0.1326
##  Woman             0.0556 0.0105 28   0.0340   0.0771
##
## year_post2020 =  TRUE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man              -1.7074 1.9947 28  -5.7934   2.3786
##  Woman            -0.8338 0.9441 28  -2.7678   1.1002
##
## Confidence level used: 0.95
```

```
# Last authors
sub_rate_gs_last_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman   0.0941 0.0134 28   7.014  <.0001
##
## year_post2020 =  TRUE:
##  contrast    estimate     SE df t.ratio p.value
##  Man - Woman  -1.3989 2.3662 28  -0.591  0.5591
```

```
sub_rate_gs_last_robust$emtrends
```

```
## year_post2020 = FALSE:
```

```
##  author_gender year.trend      SE df lower.CL upper.CL
##  Man               0.1272 0.01267 28    0.101   0.1532
##  Woman             0.0331 0.00442 28    0.024   0.0421
##
## year_post2020 =  TRUE:
##  author_gender year.trend      SE df lower.CL upper.CL
##  Man              -1.8880 2.30263 28   -6.605   2.8287
##  Woman            -0.4891 0.54491 28   -1.605   0.6271
##
## Confidence level used: 0.95
```

Time to decision

We fit linear regression models to assess the relationship between first or last author gender and time to decision, with cluster robust errors. We include an interaction effect for whether the submission was received prior to January 1, 2014 (pre-pandemic, before discontinuity), between January 1, 2014 and March 24, 2020 (pre-pandemic, after discontinuity), or after to March 24, 2020 (pandemic).

```
# First author
time_first_prepost_lm = lm(t_delta ~ first_author_gender*time_periods,
                           data = df_original)
time_first_prepost_robust =
  emmeans(time_first_prepost_lm,
          pairwise ~ first_author_gender | time_periods,
          vcov = sandwich::vcovCL(time_first_prepost_lm,
                                  cluster = ~first_author*last_author),
          adjust = "none")

# Last author
time_last_prepost_lm = lm(t_delta ~ last_author_gender*time_periods,
                          data = df_original)
time_last_prepost_robust =
  emmeans(time_last_prepost_lm,
          pairwise ~ last_author_gender | time_periods,
          vcov = sandwich::vcovCL(time_last_prepost_lm,
                                  cluster = ~first_author*last_author),
          adjust = "none")

# Collect p-values
all_pvals = c(
  all_pvals,
  setNames(data.frame(time_first_prepost_robust$contrasts)[,"p.value"],
           paste("time_first_prepost", 1:3)),
  setNames(data.frame(time_last_prepost_robust$contrasts)[,"p.value"],
           paste("time_last_prepost", 1:3))
)

# First authors
time_first_prepost_robust$contrasts
```

```
## time_periods = 1:
##  contrast    estimate   SE    df t.ratio p.value
##  Man - Woman    -4.26 2.20 32487  -1.933  0.0533
##
## time_periods = 2:
```

```
##  contrast    estimate   SE    df t.ratio p.value
##  Man - Woman    -6.59 1.15 32487  -5.746  <.0001
##
## time_periods = 3:
##  contrast    estimate   SE    df t.ratio p.value
##  Man - Woman    -8.46 1.38 32487  -6.148  <.0001
```

```
time_first_prepost_robust$emmeans
```

```
## time_periods = 1:
##  first_author_gender emmean   SE    df lower.CL upper.CL
##  Man                  71.0 1.34 32487     68.3     73.6
##  Woman                75.2 2.11 32487     71.1     79.4
##
## time_periods = 2:
##  first_author_gender emmean   SE    df lower.CL upper.CL
##  Man                  36.3 1.22 32487     33.9     38.7
##  Woman                42.9 1.11 32487     40.7     45.1
##
## time_periods = 3:
##  first_author_gender emmean   SE    df lower.CL upper.CL
##  Man                  32.6 1.02 32487     30.6     34.6
##  Woman                41.0 1.37 32487     38.3     43.7
##
## Confidence level used: 0.95
```

```
# Last authors
time_last_prepost_robust$contrasts
```

```
## time_periods = 1:
##  contrast    estimate   SE    df t.ratio p.value
##  Man - Woman     1.60 2.80 31233   0.571  0.5679
##
## time_periods = 2:
##  contrast    estimate   SE    df t.ratio p.value
##  Man - Woman    -4.85 1.41 31233  -3.434  0.0006
##
## time_periods = 3:
##  contrast    estimate   SE    df t.ratio p.value
##  Man - Woman    -8.98 1.84 31233  -4.882  <.0001
```

```
time_last_prepost_robust$emmeans
```

```
## time_periods = 1:
##  last_author_gender emmean    SE    df lower.CL upper.CL
##  Man                 72.1 1.065 31233     70.0     74.2
##  Woman               70.5 2.606 31233     65.4     75.6
##
## time_periods = 2:
##  last_author_gender emmean    SE    df lower.CL upper.CL
##  Man                 38.3 0.557 31233     37.2     39.4
##  Woman               43.2 1.307 31233     40.6     45.7
##
## time_periods = 3:
##  last_author_gender emmean    SE    df lower.CL upper.CL
##  Man                 33.6 0.678 31233     32.3     34.9
```

```
##  Woman                42.6 1.726 31233     39.2     45.9
##
## Confidence level used: 0.95
```

Author pairs

Now, we consider all pairwise comparisons between author pairs for time to decision.

```
time_pairs_prepost_lm =
  lm(t_delta ~ first_author_gender*last_author_gender*time_periods,
     data = df_original)
# Gender pairs within each time period
time_pairs_prepost_gender_robust =
  emmeans(time_pairs_prepost_lm,
          pairwise ~ first_author_gender*last_author_gender | time_periods,
          vcov = sandwich::vcovCL(time_pairs_prepost_lm,
                                  cluster = ~first_author*last_author),
          adjust = "none")

# Time to decision p-values
# Gender pairs within each time period
time_pairs_prepost_gender_pvals =
  summary(time_pairs_prepost_gender_robust$contrasts)[,"p.value"]
names(time_pairs_prepost_gender_pvals) =
  paste(rep(c("time (pre-pandemic, before discontinuity)",
              "time (pre-pandemic, after discontinuity)",
              "time (pandemic)"), each = 6),
        as.character(time_pairs_prepost_gender_robust$contrasts@levels$contrast))

# Collect p-values
all_pvals = c(
  all_pvals,
  time_pairs_prepost_gender_pvals
)

# Gender pairs within each time period
time_pairs_prepost_gender_robust$contrasts
```

```
## time_periods = 1:
##  contrast                estimate   SE    df t.ratio p.value
##  Man Man - Woman Man      -5.1220 2.72 26974  -1.882  0.0598
##  Man Man - Man Woman       0.0367 3.69 26974   0.010  0.9921
##  Man Man - Woman Woman    -1.0445 4.46 26974  -0.234  0.8147
##  Woman Man - Man Woman     5.1587 4.25 26974   1.213  0.2252
##  Woman Man - Woman Woman   4.0776 4.92 26974   0.829  0.4072
##  Man Woman - Woman Woman  -1.0812 5.44 26974  -0.199  0.8424
##
## time_periods = 2:
##  contrast                estimate   SE    df t.ratio p.value
##  Man Man - Woman Man      -5.1041 1.24 26974  -4.132  <.0001
##  Man Man - Man Woman      -5.2446 1.88 26974  -2.791  0.0053
##  Man Man - Woman Woman    -7.4838 2.01 26974  -3.730  0.0002
##  Woman Man - Man Woman    -0.1405 2.12 26974  -0.066  0.9470
##  Woman Man - Woman Woman  -2.3797 2.18 26974  -1.092  0.2748
##  Man Woman - Woman Woman  -2.2392 2.48 26974  -0.905  0.3656
```

```
##
## time_periods = 3:
##  contrast                estimate   SE    df t.ratio p.value
##  Man Man - Woman Man      -4.8068 1.47 26974  -3.280  0.0010
##  Man Man - Man Woman      -6.5320 2.11 26974  -3.095  0.0020
##  Man Man - Woman Woman   -17.8889 3.18 26974  -5.622  <.0001
##  Woman Man - Man Woman    -1.7252 2.34 26974  -0.738  0.4607
##  Woman Man - Woman Woman -13.0822 3.29 26974  -3.975  0.0001
##  Man Woman - Woman Woman -11.3570 3.53 26974  -3.221  0.0013
```

```
time_pairs_prepost_gender_robust$emmeans
```

```
## time_periods = 1:
##  first_author_gender last_author_gender emmean    SE    df lower.CL upper.CL
##  Man                 Man                  72.2 1.263 26974     69.7     74.6
##  Woman               Man                  77.3 2.441 26974     72.5     82.1
##  Man                 Woman                72.1 3.484 26974     65.3     79.0
##  Woman               Woman                73.2 4.273 26974     64.8     81.6
##
## time_periods = 2:
##  first_author_gender last_author_gender emmean    SE    df lower.CL upper.CL
##  Man                 Man                  38.0 0.625 26974     36.8     39.3
##  Woman               Man                  43.1 1.144 26974     40.9     45.4
##  Man                 Woman                43.3 1.780 26974     39.8     46.8
##  Woman               Woman                45.5 1.907 26974     41.8     49.3
##
## time_periods = 3:
##  first_author_gender last_author_gender emmean    SE    df lower.CL upper.CL
##  Man                 Man                  33.5 0.846 26974     31.8     35.2
##  Woman               Man                  38.3 1.278 26974     35.8     40.8
##  Man                 Woman                40.0 1.959 26974     36.2     43.9
##  Woman               Woman                51.4 3.067 26974     45.4     57.4
##
## Confidence level used: 0.95
```

```
png("ttd_pairs_pre_post.png", width = 1000, height = 750, res = 140)
rbind(data.frame(time_pairs_prepost_gender_robust$emmeans)) %>%
  mutate(group = interaction(first_author_gender, last_author_gender, sep = ", "),
         time_periods = factor(case_when(
           time_periods == "1" ~ "Pre-pandemic, before discontinuity",
           time_periods == "2" ~ "Pre-pandemic, after discontinuity",
           time_periods == "3" ~ "Pandemic"),
           levels = c("Pre-pandemic, before discontinuity",
                      "Pre-pandemic, after discontinuity", "Pandemic"))) %>%
  ggplot(aes(x = emmean, y = group, color = time_periods)) +
  geom_pointrange(aes(xmin = lower.CL, xmax = upper.CL), size = 0.25,
                  position = position_jitterdodge(jitter.width = 0.01)) +
  xlab("Time to Decision (Days)") +
  ylab("First Author Gender, Last Author Gender") +
  ggtitle("Time to Decision for Author Pairs") +
  guides(color = guide_legend(ncol = 1)) +
  theme(legend.position = "bottom", legend.title = element_blank())
dev.off()
```

```
## pdf
```

```
##   2
```

Number of accepted manuscripts

Rate of change in number of accepted manuscripts

This is the same as the analysis for the rate of change in number of submitted manuscripts, except that now we subset the data to only include accepted manuscripts.

```
# First authors
acc_rate_first_lm = change_over_time(df_original,
                                     "accepted", "first_author_gender")
acc_rate_first_robust =
  emtrends(acc_rate_first_lm$fit,
           pairwise ~ author_gender | year_post2020,
           vcov = vcovHC, var = "year",
           adjust = "none")

# Last authors
acc_rate_last_lm = change_over_time(df_original,
                                    "accepted", "last_author_gender")
acc_rate_last_robust =
  emtrends(acc_rate_last_lm$fit,
           pairwise ~ author_gender | year_post2020,
           vcov = vcovHC, var = "year",
           adjust = "none")

# First-last author pairs
acc_rate_pair_lm = change_over_time_author_interaction(df_original,
                                                       "accepted")
acc_rate_pair_robust =
  emtrends(acc_rate_pair_lm$fit,
           pairwise ~ author_gender_pair | year_post2020,
           vcov = vcovHC, var = "year",
           adjust = "none")

# Extract relevant contrasts
acc_rate_pair_tab = data.frame(acc_rate_pair_robust$contrasts)
rownames(acc_rate_pair_tab) =
  paste(rep(c("(2005-2019)", "(2020-2022)"), each = 6),
        acc_rate_pair_tab$contrast)
acc_rate_pair_pvals = acc_rate_pair_tab[,"p.value"]
names(acc_rate_pair_pvals) = paste("acc_rate_pair", rownames(acc_rate_pair_tab))

# Collect p-values
all_pvals = c(
  all_pvals,
  setNames(data.frame(acc_rate_first_robust$contrasts)[,"p.value"],
           paste("acc_rate_first", c("pre", "post"))),
  setNames(data.frame(acc_rate_last_robust$contrasts)[,"p.value"],
           paste("acc_rate_last", c("pre", "post"))),
  acc_rate_pair_pvals
  )
```

These are the trend etsimates for first authors, last authors, and author pairs, with robust standard errors.

```r
# First authors
acc_rate_first_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate    SE df t.ratio p.value
##  Man - Woman     3.65  2.78 28   1.313  0.1999
##
## year_post2020 =  TRUE:
##  contrast    estimate    SE df t.ratio p.value
##  Man - Woman  -102.00 94.02 28  -1.085  0.2872
```

```r
acc_rate_first_robust$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man                10.66  2.679 28     5.17    16.15
##  Woman               7.01  0.754 28     5.46     8.55
##
## year_post2020 =  TRUE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man              -169.00 87.681 28  -348.61    10.61
##  Woman             -67.00 33.941 28  -136.53     2.53
##
## Confidence level used: 0.95
```

```r
# Last authors
acc_rate_last_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate    SE df t.ratio p.value
##  Man - Woman     7.03   2.6 28   2.702  0.0116
##
## year_post2020 =  TRUE:
##  contrast    estimate    SE df t.ratio p.value
##  Man - Woman  -124.00 108.1 28  -1.147  0.2609
```

```r
acc_rate_last_robust$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend      SE df lower.CL upper.CL
##  Man                11.7   2.502 28     6.60    16.85
##  Woman               4.7   0.705 28     3.26     6.14
##
## year_post2020 =  TRUE:
##  author_gender year.trend      SE df lower.CL upper.CL
##  Man               -164.0 107.480 28  -384.16    56.16
##  Woman              -40.0  11.314 28   -63.18   -16.82
##
## Confidence level used: 0.95
```

```r
# First-last author pairs
acc_rate_pair_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast                estimate     SE df t.ratio p.value
##  Man Man - Woman Man        2.339  2.100 56   1.114  0.2701
##  Man Man - Man Woman        3.957  2.087 56   1.896  0.0631
```

```
##  Man Man - Woman Woman      4.632  2.064 56   2.244  0.0288
##  Woman Man - Man Woman      1.618  0.680 56   2.380  0.0207
##  Woman Man - Woman Woman    2.293  0.606 56   3.787  0.0004
##  Man Woman - Woman Woman    0.675  0.559 56   1.208  0.2321
##
## year_post2020 =  TRUE:
##  contrast                estimate     SE df t.ratio p.value
##  Man Man - Woman Man      -74.500 85.103 56  -0.875  0.3851
##  Man Man - Man Woman      -91.500 84.253 56  -1.086  0.2821
##  Man Man - Woman Woman    -87.500 84.726 56  -1.033  0.3062
##  Woman Man - Man Woman    -17.000 13.416 56  -1.267  0.2104
##  Woman Man - Woman Woman  -13.000 16.125 56  -0.806  0.4235
##  Man Woman - Woman Woman    4.000 10.770 56   0.371  0.7117
```

```
acc_rate_pair_robust$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender_pair year.trend     SE df lower.CL upper.CL
##  Man Man                  6.48  2.038 56     2.40    10.56
##  Woman Man                4.14  0.508 56     3.12     5.16
##  Man Woman                2.52  0.451 56     1.62     3.43
##  Woman Woman              1.85  0.329 56     1.19     2.51
##
## year_post2020 =  TRUE:
##  author_gender_pair year.trend     SE df lower.CL upper.CL
##  Man Man               -108.50 84.146 56  -277.06    60.06
##  Woman Man              -34.00 12.728 56   -59.50    -8.50
##  Man Woman              -17.00  4.243 56   -25.50    -8.50
##  Woman Woman            -21.00  9.899 56   -40.83    -1.17
##
## Confidence level used: 0.95
```

Here, we restrict to unique first, last, and first-last author pairs for each year's acceptances.

```
# Unique authors
# First authors
acc_rate_first_lm2 =
  change_over_time(df_original %>%
                     select(accepted, first_author_gender,
                            first_author, receipt_year) %>%
                     unique(), "accepted", "first_author_gender")
acc_rate_first_robust2 =
    emtrends(acc_rate_first_lm2$fit,
          pairwise ~ author_gender | year_post2020,
          vcov = vcovHC, var = "year",
          adjust = "none")

# Last authors
acc_rate_last_lm2 =
  change_over_time(df_original %>%
                     select(accepted, last_author_gender,
                            last_author, receipt_year) %>%
                     unique(), "accepted", "last_author_gender")
acc_rate_last_robust2 =
    emtrends(acc_rate_last_lm2$fit,
          pairwise ~ author_gender | year_post2020,
```

```
          vcov = vcovHC, var = "year",
          adjust = "none")

# First-last author pairs
acc_rate_pair_lm2 = change_over_time_author_interaction(
  df_original %>% select(accepted, first_author_gender, first_author,
                         last_author_gender, last_author, receipt_year) %>%
                    unique(), "accepted")
acc_rate_pair_robust2 =
  emtrends(acc_rate_pair_lm2$fit,
          pairwise ~ author_gender_pair | year_post2020,
          vcov = vcovHC, var = "year",
          adjust = "none")

# Extract relevant contrasts
acc_rate_pair_tab2 = data.frame(acc_rate_pair_robust2$contrasts)
rownames(acc_rate_pair_tab2) =
  paste(rep(c("(2005-2019)", "(2020-2022)"), each = 6),
        acc_rate_pair_tab2$contrast)
acc_rate_pair_pvals2 = acc_rate_pair_tab2[,"p.value"]
names(acc_rate_pair_pvals2) = paste("acc_rate_pair2", rownames(acc_rate_pair_tab2))

# Collect p-values
all_pvals = c(
  all_pvals,
  setNames(data.frame(acc_rate_first_robust2$contrasts)[,"p.value"],
           paste("acc_rate_first2", c("pre", "post"))),
  setNames(data.frame(acc_rate_last_robust2$contrasts)[,"p.value"],
           paste("acc_rate_last2", c("pre", "post"))),
  acc_rate_pair_pvals2
  )

# First authors
acc_rate_first_robust2$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate    SE df t.ratio p.value
##  Man - Woman     2.39  2.57 28   0.928  0.3613
##
## year_post2020 =  TRUE:
##  contrast    estimate    SE df t.ratio p.value
##  Man - Woman   -88.00 82.76 28  -1.063  0.2968
```

```
acc_rate_first_robust2$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man                 8.94  2.470 28     3.88  13.9955
##  Woman               6.55  0.725 28     5.06   8.0313
##
## year_post2020 =  TRUE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man              -146.00 77.782 28  -305.33  13.3287
##  Woman             -58.00 28.284 28  -115.94  -0.0623
##
```

```
## Confidence level used: 0.95
```

```
# Last authors
acc_rate_last_robust2$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate    SE df t.ratio p.value
##  Man - Woman      4.6  2.57 28   1.788  0.0845
##
## year_post2020 =  TRUE:
##  contrast    estimate    SE df t.ratio p.value
##  Man - Woman    -99.5 91.92 28  -1.083  0.2883
```

```
acc_rate_last_robust2$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man                8.55  2.513 28     3.40    13.70
##  Woman              3.95  0.538 28     2.85     5.06
##
## year_post2020 =  TRUE:
##  author_gender year.trend     SE df lower.CL upper.CL
##  Man              -127.50 91.217 28  -314.35    59.35
##  Woman             -28.00 11.314 28   -51.18    -4.82
##
## Confidence level used: 0.95
```

```
# First-last author pairs
acc_rate_pair_robust2$contrasts
```

```
## year_post2020 = FALSE:
##  contrast                estimate     SE df t.ratio p.value
##  Man Man - Woman Man        1.918  2.060 56   0.931  0.3559
##  Man Man - Man Woman        3.554  2.054 56   1.730  0.0891
##  Man Man - Woman Woman      4.175  2.023 56   2.063  0.0437
##  Woman Man - Man Woman      1.636  0.679 56   2.408  0.0194
##  Woman Man - Woman Woman    2.257  0.581 56   3.882  0.0003
##  Man Woman - Woman Woman    0.621  0.557 56   1.116  0.2694
##
## year_post2020 =  TRUE:
##  contrast                estimate     SE df t.ratio p.value
##  Man Man - Woman Man      -69.500 83.501 56  -0.832  0.4088
##  Man Man - Man Woman      -85.500 83.165 56  -1.028  0.3083
##  Man Man - Woman Woman    -83.500 83.501 56  -1.000  0.3216
##  Woman Man - Man Woman    -16.000 14.142 56  -1.131  0.2627
##  Woman Man - Woman Woman  -14.000 16.000 56  -0.875  0.3853
##  Man Woman - Woman Woman    2.000 14.142 56   0.141  0.8880
```

```
acc_rate_pair_robust2$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender_pair year.trend     SE df lower.CL upper.CL
##  Man Man                 5.91  2.000 56     1.90    9.914
##  Woman Man               3.99  0.494 56     3.00    4.980
##  Man Woman               2.35  0.466 56     1.42    3.286
##  Woman Woman             1.73  0.306 56     1.12    2.345
##
```

```
## year_post2020 =  TRUE:
##  author_gender_pair year.trend     SE df lower.CL upper.CL
##  Man Man               -101.50 82.731 56  -267.23   64.231
##  Woman Man              -32.00 11.314 56   -54.66   -9.336
##  Man Woman              -16.00  8.485 56   -33.00    0.998
##  Woman Woman            -18.00 11.314 56   -40.66    4.664
##
## Confidence level used: 0.95
```

We re-run the analysis for first and last author gender, but normalize the outcome (number of acceptances by gender) by the total number of board-certified surgeons by gender for that year.

```
# First authors
acc_rate_gs_first_lm = change_over_time_gen_surg(df_original,
                                                 "accepted",
                                                 "first_author_gender")
acc_rate_gs_first_robust =
    emtrends(acc_rate_gs_first_lm$fit,
          pairwise ~ author_gender | year_post2020,
          vcov = vcovHC, var = "year",
          adjust = "none")

# Last authors
acc_rate_gs_last_lm = change_over_time_gen_surg(df_original,
                                                "accepted",
                                                "last_author_gender")
acc_rate_gs_last_robust =
    emtrends(acc_rate_gs_last_lm$fit,
          pairwise ~ author_gender | year_post2020,
          vcov = vcovHC, var = "year",
          adjust = "none")

# Collect p-values
all_pvals = c(
  all_pvals,
  setNames(data.frame(acc_rate_gs_first_robust$contrasts)[,"p.value"],
           paste("acc_rate_gs_first", c("pre", "post"))),
  setNames(data.frame(acc_rate_gs_last_robust$contrasts)[,"p.value"],
           paste("acc_rate_gs_last", c("pre", "post")))
  )


# First authors
acc_rate_gs_first_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate      SE df t.ratio p.value
##  Man - Woman   0.0099 0.00635 28   1.558  0.1304
##
## year_post2020 =  TRUE:
##  contrast    estimate      SE df t.ratio p.value
##  Man - Woman  -0.1724 0.53708 28  -0.321  0.7505
```

```
acc_rate_gs_first_robust$emtrends
```

```
## year_post2020 = FALSE:
```

```
##  author_gender year.trend      SE df  lower.CL upper.CL
##  Man              0.01825 0.00474 28  0.008531    0.028
##  Woman            0.00835 0.00423 28 -0.000308    0.017
##
## year_post2020 =  TRUE:
##  author_gender year.trend      SE df  lower.CL upper.CL
##  Man             -0.41458 0.45520 28 -1.347015    0.518
##  Woman           -0.24214 0.28505 28 -0.826034    0.342
##
## Confidence level used: 0.95
```

```
# Last authors
acc_rate_gs_last_robust$contrasts
```

```
## year_post2020 = FALSE:
##  contrast    estimate      SE df t.ratio p.value
##  Man - Woman   0.0141 0.00491 28   2.864  0.0078
##
## year_post2020 =  TRUE:
##  contrast    estimate      SE df t.ratio p.value
##  Man - Woman  -0.2653 0.51542 28  -0.515  0.6108
```

```
acc_rate_gs_last_robust$emtrends
```

```
## year_post2020 = FALSE:
##  author_gender year.trend      SE df  lower.CL upper.CL
##  Man              0.0196 0.00432 28  0.010733   0.0284
##  Woman            0.0055 0.00235 28  0.000696   0.0103
##
## year_post2020 =  TRUE:
##  author_gender year.trend      SE df  lower.CL upper.CL
##  Man             -0.4136 0.48868 28 -1.414651   0.5874
##  Woman           -0.1484 0.16389 28 -0.484071   0.1873
##
## Confidence level used: 0.95
```

This figure plots the linear regression lines for the number of submissions and acceptances by year and first, last, and first/last author gender. The raw data counts are shown as points.

```
plot_change_over_time =
  function(obj, paired, outcome, author_gender, fig_lab,
           my_pal = c("#0072B2", "#D55E00", "#E69F00", "#009E73")) {

  # Helper function for generating labels
  lm_lab = function(group, m, b) {
    paste0(group, "\ny = " , round(m), "x", sign(b), round(abs(b)))
  }
  # Extract coefficients from lm
  coefs = coef(obj$fit)

  # Remove 2023 (incomplete year)
  obj$df = obj$df[rownames(obj$df) != 2023,]

  # Set up data frames for plotting
  if (paired == FALSE) { # First or last author, not paired
    plot_df = melt(obj$df) %>%
```

```
      mutate(post2020 = Var1 >= 2020,
             label = "")
    plot_df$label[plot_df$Var1 == 2020] =
      c("Man", "Woman")
      # c(lm_lab("Men", coefs[3], coefs[1]),
      #   lm_lab("Women", sum(coefs[3:4]), sum(coefs[1:2])))
  } else { # first/last author pairs
    plot_df = melt(obj$df) %>%
      within(Var2 <- factor(Var2, levels = c("Man.Man", "Woman.Woman",
                                             "Man.Woman", "Woman.Man"))) %>%
      mutate(post2020 = Var1 >= 2020,
             label = "")
    plot_df$label[plot_df$Var1 == 2020] =
      c("Man, Man", "Woman, Man", "Man, Woman", "Woman, Woman")
      # c(lm_lab("Man, Man", coefs[5], coefs[1]),
      #   lm_lab("Woman, Man", sum(coefs[c(5:6)]), sum(coefs[c(1:2)])),
      #   lm_lab("Man, Woman", sum(coefs[c(5,7)]), sum(coefs[c(1,3)])),
      #   lm_lab("Woman, Woman", sum(coefs[c(5,8)]), sum(coefs[c(1,4)])))
  }

  # Plot
   p1 = plot_df %>%
    ggplot(aes(x = Var1, y = value, color = Var2)) +
    geom_point(size = 1) +
    geom_smooth(aes(group = interaction(Var2, post2020)),
                method = "lm", se = FALSE, linewidth = 0.7) +
    scale_color_manual(values = my_pal, guide = "none") +
    geom_label_repel(aes(label = label),
                     seed = 1, hjust = 0, size = 4, force = 45,
                     segment.color = NA, fill = alpha(c("white"), 0.5),
                     xlim = c(min(plot_df$Var1), max(plot_df$Var1)),
                     ylim = c(min(plot_df$value), max(plot_df$value))) +
    xlab("Year of Receipt") +
    ylab("Manuscript Count") +
    ggtitle(paste(fig_lab, "Number of", outcome, "\nby", author_gender)) +
    theme(plot.title = element_text(size = 13, hjust = 0.5),
          axis.title = element_text(size = 13),
          axis.text = element_text(size = 11))

  return(p1)
}

p1 = plot_change_over_time(sub_rate_first_lm, FALSE,
                           "Submissions", "First Author Gender", "(a)")
p2 = plot_change_over_time(sub_rate_last_lm, FALSE,
                           "Submissions", "Last Author Gender", "(b)")
p3 = plot_change_over_time(sub_rate_pair_lm, TRUE,
                           "Submissions",
                           "First, Last Author Gender Pairs", "(c)")
p4 = plot_change_over_time(acc_rate_first_lm, FALSE,
                           "Manuscripts Accepted", "First Author Gender", "(d)")
p5 = plot_change_over_time(acc_rate_last_lm, FALSE,
                           "Manuscripts Accepted", "Last Author Gender", "(e)")
```

```
p6 = plot_change_over_time(acc_rate_pair_lm, TRUE,
                           "Manuscripts Accepted",
                           "First, Last Author Gender Pairs", "(f)")

png("regress_over_time.png", width = 1500, height = 1050, res = 140)
grid.arrange(p1, p2, p3, p4, p5, p6, nrow = 2, ncol = 3)
dev.off()
```

```
## pdf
##   2
```

This figure reproduces the linear regression plots, but using the results based on unique authors.

```
p1 = plot_change_over_time(sub_rate_first_lm2, FALSE,
                           "Submissions", "First Author Gender", "(a)")
p2 = plot_change_over_time(sub_rate_last_lm2, FALSE,
                           "Submissions", "Last Author Gender", "(b)")
p3 = plot_change_over_time(sub_rate_pair_lm2, TRUE,
                           "Submissions",
                           "First, Last Author Gender Pairs", "(c)")
p4 = plot_change_over_time(acc_rate_first_lm2, FALSE,
                           "Manuscripts Accepted", "First Author Gender", "(d)")
p5 = plot_change_over_time(acc_rate_last_lm2, FALSE,
                           "Manuscripts Accepted", "Last Author Gender", "(e)")
p6 = plot_change_over_time(acc_rate_pair_lm2, TRUE,
                           "Manuscripts Accepted",
                           "First, Last Author Gender Pairs", "(f)")

png("regress_over_time2.png", width = 1500, height = 1050, res = 140)
grid.arrange(p1, p2, p3, p4, p5, p6, nrow = 2, ncol = 3)
dev.off()
```

```
## pdf
##   2
```

Chance of acceptance

We fit logistic regression models to assess the relationship between first or last author gender and acceptance rate, with cluster robust errors. We include an interaction effect for whether the submission was received prior to January 1, 2014 (pre-pandemic, before discontinuity), between January 1, 2014 and March 24, 2020 (pre-pandemic, after discontinuity), or after to March 24, 2020 (pandemic).

```
# First author
acc_chance_first_prepost_glm =
  glm(accepted ~ first_author_gender*time_periods,
      data = df_original, family = binomial)
acc_chance_first_prepost_robust =
  emmeans(acc_chance_first_prepost_glm,
          pairwise ~ first_author_gender | time_periods,
          vcov = sandwich::vcovCL(acc_chance_first_prepost_glm,
                                  cluster = ~first_author*last_author),
          type = "response", adjust = "none")

# Last author
acc_chance_last_prepost_glm =
  glm(accepted ~ last_author_gender*time_periods,
```

```
      data = df_original, family = binomial)
acc_chance_last_prepost_robust =
  emmeans(acc_chance_last_prepost_glm,
          pairwise ~ last_author_gender | time_periods,
          vcov = sandwich::vcovCL(acc_chance_last_prepost_glm,
                                  cluster = ~first_author*last_author),
          type = "response", adjust = "none")

# Collect p-values
all_pvals = c(
  all_pvals,
  setNames(data.frame(acc_chance_first_prepost_robust$contrasts)[,"p.value"],
           paste("acc_chance_first_prepost", 1:3)),
  setNames(data.frame(acc_chance_last_prepost_robust$contrasts)[,"p.value"],
           paste("acc_chance_last_prepost", 1:3))
)

# First authors
acc_chance_first_prepost_robust$contrasts
```

```
## time_periods = 1:
##  contrast    odds.ratio     SE  df null z.ratio p.value
##  Man / Woman      1.017 0.0838 Inf    1   0.206  0.8371
##
## time_periods = 2:
##  contrast    odds.ratio     SE  df null z.ratio p.value
##  Man / Woman      1.034 0.0678 Inf    1   0.508  0.6115
##
## time_periods = 3:
##  contrast    odds.ratio     SE  df null z.ratio p.value
##  Man / Woman      0.795 0.0534 Inf    1  -3.413  0.0006
##
## Tests are performed on the log odds ratio scale
```

```
acc_chance_first_prepost_robust$emmeans
```

```
## time_periods = 1:
##  first_author_gender  prob     SE  df asymp.LCL asymp.UCL
##  Man                 0.278 0.0204 Inf     0.240     0.319
##  Woman               0.274 0.0138 Inf     0.248     0.302
##
## time_periods = 2:
##  first_author_gender  prob     SE  df asymp.LCL asymp.UCL
##  Man                 0.232 0.0140 Inf     0.206     0.260
##  Woman               0.226 0.0108 Inf     0.206     0.248
##
## time_periods = 3:
##  first_author_gender  prob     SE  df asymp.LCL asymp.UCL
##  Man                 0.192 0.0118 Inf     0.169     0.216
##  Woman               0.229 0.0138 Inf     0.204     0.258
##
## Confidence level used: 0.95
## Intervals are back-transformed from the logit scale
```

```
# Last authors
acc_chance_last_prepost_robust$contrasts
```

```
## time_periods = 1:
##  contrast    odds.ratio     SE  df null z.ratio p.value
##  Man / Woman      0.934 0.0918 Inf    1  -0.693  0.4881
##
## time_periods = 2:
##  contrast    odds.ratio     SE  df null z.ratio p.value
##  Man / Woman      0.814 0.0647 Inf    1  -2.587  0.0097
##
## time_periods = 3:
##  contrast    odds.ratio     SE  df null z.ratio p.value
##  Man / Woman      0.743 0.0661 Inf    1  -3.340  0.0008
##
## Tests are performed on the log odds ratio scale
```

```
acc_chance_last_prepost_robust$emmeans
```

```
## time_periods = 1:
##  last_author_gender  prob      SE  df asymp.LCL asymp.UCL
##  Man                0.255 0.00625 Inf     0.243     0.268
##  Woman              0.268 0.01839 Inf     0.234     0.306
##
## time_periods = 2:
##  last_author_gender  prob      SE  df asymp.LCL asymp.UCL
##  Man                0.210 0.00613 Inf     0.198     0.222
##  Woman              0.246 0.01326 Inf     0.221     0.273
##
## time_periods = 3:
##  last_author_gender  prob      SE  df asymp.LCL asymp.UCL
##  Man                0.180 0.00629 Inf     0.168     0.192
##  Woman              0.228 0.01404 Inf     0.201     0.256
##
## Confidence level used: 0.95
## Intervals are back-transformed from the logit scale
```

Author pairs

These are the log odds ratios from comparing marginal means for each pairwise comparison of authorship pairs, with robust standard errors.

```
acc_chance_pairs_prepost_glm =
  glm(accepted ~ first_author_gender*last_author_gender*time_periods,
           data = df_original, family = "binomial")
# Gender pairs within each time period
acc_chance_pairs_prepost_gender_robust =
  emmeans(acc_chance_pairs_prepost_glm,
          pairwise ~ first_author_gender*last_author_gender | time_periods,
          vcov = sandwich::vcovCL(acc_chance_pairs_prepost_glm,
                                  cluster = ~first_author*last_author),
          type = "response", adjust = "none")

# Chance of acceptance p-values
# Gender pairs within each time period
```

```r
acc_chance_pairs_prepost_gender_pvals =
  summary(acc_chance_pairs_prepost_gender_robust$contrasts)[,"p.value"]
names(acc_chance_pairs_prepost_gender_pvals) =
  paste(rep(c("acc_chance (pre-pandemic, before discontinuity)",
              "acc_chance (pre-pandemic, after discontinuity)",
              "acc_chance (pandemic)"), each = 6),
        as.character(acc_chance_pairs_prepost_gender_robust$contrasts@levels$contrast))

# Collect p-values
all_pvals = c(
  all_pvals,
  acc_chance_pairs_prepost_gender_pvals
)

# Gender pairs within each time period
acc_chance_pairs_prepost_gender_robust$contrasts
```

```
## time_periods = 1:
##  contrast                odds.ratio     SE  df null z.ratio p.value
##  Man Man / Woman Man          0.965 0.0711 Inf    1  -0.484  0.6287
##  Man Man / Man Woman          0.859 0.0963 Inf    1  -1.351  0.1765
##  Man Man / Woman Woman        0.998 0.1458 Inf    1  -0.010  0.9918
##  Woman Man / Man Woman        0.891 0.1130 Inf    1  -0.914  0.3609
##  Woman Man / Woman Woman      1.035 0.1588 Inf    1   0.222  0.8241
##  Man Woman / Woman Woman      1.162 0.1830 Inf    1   0.952  0.3409
##
## time_periods = 2:
##  contrast                odds.ratio     SE  df null z.ratio p.value
##  Man Man / Woman Man          1.022 0.0688 Inf    1   0.328  0.7429
##  Man Man / Man Woman          0.777 0.0731 Inf    1  -2.682  0.0073
##  Man Man / Woman Woman        0.906 0.1081 Inf    1  -0.827  0.4084
##  Woman Man / Man Woman        0.760 0.0796 Inf    1  -2.620  0.0088
##  Woman Man / Woman Woman      0.886 0.1105 Inf    1  -0.968  0.3328
##  Man Woman / Woman Woman      1.166 0.1489 Inf    1   1.202  0.2294
##
## time_periods = 3:
##  contrast                odds.ratio     SE  df null z.ratio p.value
##  Man Man / Woman Man          0.892 0.0737 Inf    1  -1.378  0.1683
##  Man Man / Man Woman          0.781 0.0861 Inf    1  -2.243  0.0249
##  Man Man / Woman Woman        0.602 0.0762 Inf    1  -4.006  0.0001
##  Woman Man / Man Woman        0.875 0.1088 Inf    1  -1.075  0.2826
##  Woman Man / Woman Woman      0.675 0.0907 Inf    1  -2.927  0.0034
##  Man Woman / Woman Woman      0.771 0.1125 Inf    1  -1.780  0.0751
##
## Tests are performed on the log odds ratio scale
```

```r
acc_chance_pairs_prepost_gender_robust$emmeans
```

```
## time_periods = 1:
##  first_author_gender last_author_gender  prob      SE  df asymp.LCL asymp.UCL
##  Man                 Man                0.259 0.00711 Inf     0.246     0.273
##  Woman               Man                0.266 0.01329 Inf     0.241     0.293
##  Man                 Woman              0.289 0.02202 Inf     0.248     0.334
##  Woman               Woman              0.260 0.02715 Inf     0.210     0.316
```

```
##
## time_periods = 2:
##  first_author_gender last_author_gender  prob      SE  df asymp.LCL asymp.UCL
##  Man                 Man                 0.221 0.00739 Inf     0.207     0.236
##  Woman               Man                 0.218 0.01039 Inf     0.198     0.239
##  Man                 Woman               0.268 0.01668 Inf     0.236     0.302
##  Woman               Woman               0.239 0.02025 Inf     0.201     0.281
##
## time_periods = 3:
##  first_author_gender last_author_gender  prob      SE  df asymp.LCL asymp.UCL
##  Man                 Man                 0.185 0.00761 Inf     0.170     0.200
##  Woman               Man                 0.203 0.01163 Inf     0.181     0.226
##  Man                 Woman               0.225 0.01768 Inf     0.192     0.261
##  Woman               Woman               0.273 0.02306 Inf     0.231     0.321
##
## Confidence level used: 0.95
## Intervals are back-transformed from the logit scale
```

```
png("acc_rate_pairs_pre_post.png", width = 1000, height = 750, res = 140)
rbind(data.frame(acc_chance_pairs_prepost_gender_robust$emmeans)) %>%
  mutate(group = interaction(first_author_gender, last_author_gender, sep = ", "),
         time_periods = factor(case_when(
           time_periods == "1" ~ "Pre-pandemic, before discontinuity",
           time_periods == "2" ~ "Pre-pandemic, after discontinuity",
           time_periods == "3" ~ "Pandemic"),
           levels = c("Pre-pandemic, before discontinuity",
                      "Pre-pandemic, after discontinuity", "Pandemic"))) %>%
  ggplot(aes(x = prob, y = group, color = time_periods)) +
  geom_pointrange(aes(xmin = asymp.LCL, xmax = asymp.UCL), size = 0.25,
                  position = position_jitterdodge(jitter.width = 0.01)) +
  xlab("Probability of Acceptance") +
  ylab("First Author Gender, Last Author Gender") +
  ggtitle("Probability of Acceptance for Author Pairs") +
  guides(color = guide_legend(ncol = 1)) +
  theme(legend.position = "bottom", legend.title = element_blank()) +
  scale_x_continuous(labels = scales::percent)
dev.off()
```

```
## pdf
##   2
```

Authorship pairs

We used chi-squared contingency table tests to assess if first authors and last authors of each gender are more likely to work with each other than would be expected by chance. We considered authorship patterns for all submissions and when restricted to acceptances, and included a sensitivity analysis that drops articles written by first/last authors with multiple articles.

```
# Run proportions test for each authorship pair
author_pair_prop = function(df) {
  # Expected proportion of female authors
  expected_first = df %>%
    summarize(Woman = mean(first_author_gender == "Woman", na.rm = TRUE),
              Man = mean(first_author_gender == "Man", na.rm = TRUE)) %>%
    unlist()
```

```
  expected_last = df %>%
    summarize(Woman = mean(last_author_gender == "Woman", na.rm = TRUE),
              Man = mean(last_author_gender == "Man", na.rm = TRUE)) %>%
    unlist()

  # Observed counts of each authorship pair
  observed = with(df, {
    table(first_author_gender, last_author_gender)
  })
  n = with(df, {
    sum(!is.na(first_author_gender) & !is.na(last_author_gender))
  })

  out = data.frame(
    do.call(rbind,
            lapply(c("Woman", "Man"), function(first) {
              t(sapply(c("Woman", "Man"), function(last) {
                res = prop.test(observed[first, last], n,
                                expected_first[first] * expected_last[last])
                return(data.frame("First" = first, "Last" = last,
                                  Observed = observed[first, last] / n,
                                  Expected = expected_first[first] * expected_last[last],
                                  "X-squared" = res$statistic,
                                  df = res$parameter,
                                  pvalue = res$p.value))
              }))
            })
    )
  )
  rownames(out) = NULL
  return(out)
}
```

Pre-pandemic, before discontinuity

Restricting to pre-pandemic period before discontinuity.

```
# Submissions
# Original
sub_author_by_pair_original_pre =
  author_pair_prop(df_original %>% filter(time_periods == "1"))
sub_author_by_pair_original_pre
```

```
##   First  Last   Observed   Expected X.squared df       pvalue
## 1 Woman Woman 0.03560176 0.02292508   67.7478  1 1.858036e-16
## 2 Woman   Man  0.1637261  0.1703006  2.866499  1   0.09044139
## 3   Man Woman  0.0805503 0.09571899  25.13682  1  5.34035e-07
## 4   Man   Man  0.7201218  0.7110553  3.765689  1   0.05231431
```

```
# Unique
sub_author_by_pair_unique_pre =
  author_pair_prop(df_unique_both %>% filter(time_periods == "1"))
sub_author_by_pair_unique_pre
```

```
##   First  Last   Observed   Expected X.squared df       pvalue
```

```
## 1 Woman Woman 0.06039076 0.04143544  19.89773  1 8.169717e-06
## 2 Woman   Man  0.1758437   0.194799  5.038553  1   0.02478919
## 3   Man Woman  0.1150089  0.1339642  6.811963  1  0.009054916
## 4   Man   Man  0.6487567  0.6298013  3.389678  1   0.06560575
```

```
# Acceptances
# Original
acc_author_by_pair_original_pre =
  author_pair_prop(df_original %>% filter(accepted == 1, time_periods == "1"))
acc_author_by_pair_original_pre
```

```
##   First  Last   Observed   Expected  X.squared df       pvalue
## 1 Woman Woman 0.03515781 0.02372434   13.63768  1 0.0002216917
## 2 Woman   Man   0.165801  0.1675965 0.04568586  1    0.8307479
## 3   Man Woman 0.08869357  0.1002786   3.596071  1    0.0579163
## 4   Man   Man  0.7103476  0.7084006 0.03699197  1    0.8474814
```

```
# Unique
acc_author_by_pair_unique_pre =
  author_pair_prop(df_unique_both %>% filter(accepted == 1, time_periods == "1"))
acc_author_by_pair_unique_pre
```

```
##   First  Last   Observed   Expected X.squared df     pvalue
## 1 Woman Woman 0.09206349 0.05520786  7.511785  1 0.00612966
## 2 Woman   Man  0.1714286  0.2082842  2.376045  1  0.1232091
## 3   Man Woman  0.1174603  0.1543159   3.00235  1 0.08314381
## 4   Man   Man  0.6190476   0.582192  1.610784  1  0.2043816
```

```
# Author pair p-values
sub_author_by_pair_original_pre_pvals =
    unlist(sub_author_by_pair_original_pre$pvalue)
names(sub_author_by_pair_original_pre_pvals) =
  paste("sub_author_pair_original_pre",
        sub_author_by_pair_original_pre$First,
        sub_author_by_pair_original_pre$Last)

sub_author_by_pair_unique_pre_pvals =
    unlist(sub_author_by_pair_unique_pre$pvalue)
names(sub_author_by_pair_unique_pre_pvals) =
  paste("sub_author_pair_unique_pre",
        sub_author_by_pair_unique_pre$First,
        sub_author_by_pair_unique_pre$Last)

acc_author_by_pair_original_pre_pvals =
    unlist(acc_author_by_pair_original_pre$pvalue)
names(acc_author_by_pair_original_pre_pvals) =
  paste("acc_author_pair_original_pre",
        acc_author_by_pair_original_pre$First,
        acc_author_by_pair_original_pre$Last)

acc_author_by_pair_unique_pre_pvals =
    unlist(acc_author_by_pair_unique_pre$pvalue)
names(acc_author_by_pair_unique_pre_pvals) =
  paste("acc_author_pair_unique_pre",
        acc_author_by_pair_unique_pre$First,
        acc_author_by_pair_unique_pre$Last)
```

```r
# Collect p-values
all_pvals = c(
  all_pvals,
  sub_author_by_pair_original_pre_pvals, sub_author_by_pair_unique_pre_pvals,
  acc_author_by_pair_original_pre_pvals, acc_author_by_pair_unique_pre_pvals
)
```

Pre-pandemic, after discontinuity

Restricting to pre-pandemic period before discontinuity.

```r
# Submissions
# Original
sub_author_by_pair_original_pre2 =
  author_pair_prop(df_original %>% filter(time_periods == "2"))
sub_author_by_pair_original_pre2
```

```
##   First  Last   Observed   Expected X.squared df       pvalue
## 1 Woman Woman 0.05969014 0.03904343  118.9886  1 1.053317e-27
## 2 Woman   Man  0.1987454  0.2100899  8.090953  1  0.004448676
## 3   Man Woman 0.09970535  0.1176736  32.54315  1 1.165751e-08
## 4   Man   Man  0.6418591  0.6331931   3.36472  1   0.06660689
```

```r
# Unique
sub_author_by_pair_unique_pre2 =
  author_pair_prop(df_unique_both %>% filter(time_periods == "2"))
sub_author_by_pair_unique_pre2
```

```
##   First  Last   Observed   Expected X.squared df       pvalue
## 1 Woman Woman 0.08887833 0.06430491  20.70893  1 5.346611e-06
## 2 Woman   Man  0.2124525  0.2370259  6.890179  1  0.008667057
## 3   Man Woman  0.1245247  0.1490981   9.82165  1  0.001724696
## 4   Man   Man  0.5741445  0.5495711  5.033684  1   0.02485896
```

```r
# Acceptances
# Original
acc_author_by_pair_original_pre2 =
  author_pair_prop(df_original %>% filter(accepted == 1, time_periods == "2"))
acc_author_by_pair_original_pre2
```

```
##   First  Last   Observed   Expected X.squared df       pvalue
## 1 Woman Woman 0.06299874 0.04370532  20.74655  1 5.242602e-06
## 2 Woman   Man  0.1910962  0.2006556  1.297604  1    0.2546513
## 3   Man Woman  0.1180176  0.1351503  5.833638  1   0.01572258
## 4   Man   Man  0.6278874  0.6204888 0.5225116  1    0.4697723
```

```r
# Unique
acc_author_by_pair_unique_pre2 =
  author_pair_prop(df_unique_both %>% filter(accepted == 1, time_periods == "2"))
acc_author_by_pair_unique_pre2
```

```
##   First  Last  Observed   Expected X.squared df    pvalue
## 1 Woman Woman 0.1047619 0.08052154  1.355322  1 0.2443498
## 2 Woman   Man 0.2142857  0.2385261 0.5524657  1  0.457312
## 3   Man Woman  0.147619  0.1718594 0.7050482  1 0.4010926
## 4   Man   Man 0.5333333   0.509093 0.4015132  1 0.5263088
```

```r
# Author pair p-values
sub_author_by_pair_original_pre2_pvals =
    unlist(sub_author_by_pair_original_pre2$pvalue)
names(sub_author_by_pair_original_pre2_pvals) =
  paste("sub_author_pair_original_pre2",
        sub_author_by_pair_original_pre2$First,
        sub_author_by_pair_original_pre2$Last)

sub_author_by_pair_unique_pre2_pvals =
    unlist(sub_author_by_pair_unique_pre2$pvalue)
names(sub_author_by_pair_unique_pre2_pvals) =
  paste("sub_author_pair_unique_pre2",
        sub_author_by_pair_unique_pre2$First,
        sub_author_by_pair_unique_pre2$Last)

acc_author_by_pair_original_pre2_pvals =
    unlist(acc_author_by_pair_original_pre2$pvalue)
names(acc_author_by_pair_original_pre2_pvals) =
  paste("acc_author_pair_original_pre2",
        acc_author_by_pair_original_pre2$First,
        acc_author_by_pair_original_pre2$Last)

acc_author_by_pair_unique_pre2_pvals =
    unlist(acc_author_by_pair_unique_pre2$pvalue)
names(acc_author_by_pair_unique_pre2_pvals) =
  paste("acc_author_pair_unique_pre2",
        acc_author_by_pair_unique_pre2$First,
        acc_author_by_pair_unique_pre2$Last)

# Collect p-values
all_pvals = c(
  all_pvals,
  sub_author_by_pair_original_pre2_pvals, sub_author_by_pair_unique_pre2_pvals,
  acc_author_by_pair_original_pre2_pvals, acc_author_by_pair_unique_pre2_pvals
)
```

Pandemic

Restricting to pandemic period.

```r
# Submissions
# Original
sub_author_by_pair_original_post =
  author_pair_prop(df_original %>% filter(time_periods == "3"))
sub_author_by_pair_original_post
```

```
##   First  Last   Observed   Expected X.squared df       pvalue
## 1 Woman Woman 0.09179575 0.05796677  145.4945  1 1.674356e-33
## 2 Woman   Man  0.2160069  0.2391801  20.44714  1 6.130124e-06
## 3   Man Woman  0.1045611   0.137111  62.16052  1 3.165714e-15
## 4   Man   Man  0.5876363   0.565742  13.51451  1 0.0002367261
```

```r
# Unique
sub_author_by_pair_unique_post =
  author_pair_prop(df_unique_both %>% filter(time_periods == "3"))
```

```
sub_author_by_pair_unique_post
```

```
##   First  Last  Observed   Expected X.squared df       pvalue
## 1 Woman Woman 0.1188455 0.08309577  29.17272  1 6.620487e-08
## 2 Woman   Man 0.2156197  0.2513694  11.81136  1 0.0005887054
## 3   Man Woman 0.1295982  0.1653479  16.10555  1 5.990821e-05
## 4   Man   Man 0.5359366  0.5001869  8.890779  1  0.002866144
```

```
# Acceptances
# Original
acc_author_by_pair_original_post =
  author_pair_prop(df_original %>% filter(accepted == 1, time_periods == "3"))
acc_author_by_pair_original_post
```

```
##   First  Last  Observed   Expected X.squared df       pvalue
## 1 Woman Woman 0.1249108 0.07901712  39.91994  1 2.645881e-10
## 2 Woman   Man 0.2177016  0.2572351  11.25405  1 0.0007944955
## 3   Man Woman 0.1170592  0.1559765  15.82365  1 6.952794e-05
## 4   Man   Man 0.5403283  0.5077713  5.811904  1   0.01591805
```

```
# Unique
acc_author_by_pair_unique_post =
  author_pair_prop(df_unique_both %>% filter(accepted == 1, time_periods == "3"))
acc_author_by_pair_unique_post
```

```
##   First  Last  Observed  Expected X.squared df     pvalue
## 1 Woman Woman 0.1657143 0.1086694  5.305109  1 0.02126297
## 2 Woman   Man       0.2 0.2570449  2.690722  1  0.1009341
## 3   Man Woman 0.1314286 0.1884735  3.359599  1 0.06681434
## 4   Man   Man 0.5028571 0.4458122  2.079847  1  0.1492551
```

```
# Author pair p-values
sub_author_by_pair_original_post_pvals =
    unlist(sub_author_by_pair_original_post$pvalue)
names(sub_author_by_pair_original_post_pvals) =
  paste("sub_author_pair_original_post",
        sub_author_by_pair_original_post$First,
        sub_author_by_pair_original_post$Last)

sub_author_by_pair_unique_post_pvals =
    unlist(sub_author_by_pair_unique_post$pvalue)
names(sub_author_by_pair_unique_post_pvals) =
  paste("sub_author_pair_unique_post",
        sub_author_by_pair_unique_post$First,
        sub_author_by_pair_unique_post$Last)

acc_author_by_pair_original_post_pvals =
    unlist(acc_author_by_pair_original_post$pvalue)
names(acc_author_by_pair_original_post_pvals) =
  paste("acc_author_pair_original_post",
        acc_author_by_pair_original_post$First,
        acc_author_by_pair_original_post$Last)

acc_author_by_pair_unique_post_pvals =
    unlist(acc_author_by_pair_unique_post$pvalue)
names(acc_author_by_pair_unique_post_pvals) =
```

```
  paste("acc_author_pair_unique_post",
        acc_author_by_pair_unique_post$First,
        acc_author_by_pair_unique_post$Last)

# Collect p-values
all_pvals = c(
  all_pvals,
  sub_author_by_pair_original_post_pvals, sub_author_by_pair_unique_post_pvals,
  acc_author_by_pair_original_post_pvals, acc_author_by_pair_unique_post_pvals
)
```

Multiple comparisons adjustment

We applied a Benjamini-Hochberg multiple comparisons adjustment to control the FDR.

```
all_pvals_adj = data.frame(pvals = all_pvals,
                           adj_pvals = p.adjust(all_pvals, method = "BH"))
rownames(all_pvals_adj) = names(all_pvals)
all_pvals_adj$significant = ifelse(all_pvals_adj$adj_pvals < 0.05, "*", "")

write.csv(all_pvals_adj, file = "all_pvals_adj.csv")
```

Additional tables

These tables summarize the regression output for the rate of change in submitted and accepted manuscripts, with an extra column for the adjusted p-values.

```
# Submitted manuscripts
# Pre-pandemic
sub_over_time_pairs_pre_tab = data.frame(
  Dataset = c("All", rep("", 5), "Unique", rep("", 5)),
  rbind(sub_rate_pair_tab[1:6,], sub_rate_pair_tab2[1:6,]),
  adj.p.value = all_pvals_adj[c(names(sub_rate_pair_pvals)[1:6],
                                names(sub_rate_pair_pvals2)[1:6]),
                              "adj_pvals"])
rownames(sub_over_time_pairs_pre_tab) = NULL
sub_over_time_pairs_pre_tab$year_post2020 = NULL
sub_over_time_pairs_pre_tab$df = NULL
sub_over_time_pairs_pre_tab$estimate = round(sub_over_time_pairs_pre_tab$estimate, 2)
sub_over_time_pairs_pre_tab$SE = round(sub_over_time_pairs_pre_tab$SE, 2)
sub_over_time_pairs_pre_tab$t.ratio = round(sub_over_time_pairs_pre_tab$t.ratio, 2)

colnames(sub_over_time_pairs_pre_tab) =
  c("Dataset", "Author Gender Pair Contrast",
    "Estimate", "Std. Error", "t-Statistic", "p-value", "Adj. p-value")

write.csv(sub_over_time_pairs_pre_tab,
          file = "sub_over_time_pairs_pre_tab.csv", row.names = FALSE)


# Pandemic
sub_over_time_pairs_post_tab = data.frame(
  Dataset = c("All", rep("", 5), "Unique", rep("", 5)),
```

```
  rbind(sub_rate_pair_tab[7:12,], sub_rate_pair_tab2[7:12,]),
  adj.p.value = all_pvals_adj[c(names(sub_rate_pair_pvals)[7:12],
                                names(sub_rate_pair_pvals2)[7:12]),
                              "adj_pvals"])
rownames(sub_over_time_pairs_post_tab) = NULL
sub_over_time_pairs_post_tab$year_post2020 = NULL
sub_over_time_pairs_post_tab$df = NULL
sub_over_time_pairs_post_tab$estimate = round(sub_over_time_pairs_post_tab$estimate, 2)
sub_over_time_pairs_post_tab$SE = round(sub_over_time_pairs_post_tab$SE, 2)
sub_over_time_pairs_post_tab$t.ratio = round(sub_over_time_pairs_post_tab$t.ratio, 2)

colnames(sub_over_time_pairs_post_tab) =
  c("Dataset", "Author Gender Pair Contrast",
    "Estimate", "Std. Error", "t-Statistic", "p-value", "Adj. p-value")

write.csv(sub_over_time_pairs_post_tab,
          file = "sub_over_time_pairs_post_tab.csv", row.names = FALSE)


# Accepted manuscripts
# Pre-pandemic
acc_over_time_pairs_pre_tab = data.frame(
  Dataset = c("All", rep("", 5), "Unique", rep("", 5)),
  rbind(acc_rate_pair_tab[1:6,], acc_rate_pair_tab2[1:6,]),
  adj.p.value = all_pvals_adj[c(names(acc_rate_pair_pvals)[1:6],
                                names(acc_rate_pair_pvals2)[1:6]),
                              "adj_pvals"])
rownames(acc_over_time_pairs_pre_tab) = NULL
acc_over_time_pairs_pre_tab$year_post2020 = NULL
acc_over_time_pairs_pre_tab$df = NULL
acc_over_time_pairs_pre_tab$estimate = round(acc_over_time_pairs_pre_tab$estimate, 2)
acc_over_time_pairs_pre_tab$SE = round(acc_over_time_pairs_pre_tab$SE, 2)
acc_over_time_pairs_pre_tab$t.ratio = round(acc_over_time_pairs_pre_tab$t.ratio, 2)

colnames(acc_over_time_pairs_pre_tab) =
  c("Dataset", "Author Gender Pair Contrast",
    "Estimate", "Std. Error", "t-Statistic", "p-value", "Adj. p-value")

write.csv(acc_over_time_pairs_pre_tab,
          file = "acc_over_time_pairs_pre_tab.csv", row.names = FALSE)


# Pandemic
acc_over_time_pairs_post_tab = data.frame(
  Dataset = c("All", rep("", 5), "Unique", rep("", 5)),
  rbind(acc_rate_pair_tab[7:12,], acc_rate_pair_tab2[7:12,]),
  adj.p.value = all_pvals_adj[c(names(acc_rate_pair_pvals)[7:12],
                                names(acc_rate_pair_pvals2)[7:12]),
                              "adj_pvals"])
rownames(acc_over_time_pairs_post_tab) = NULL
acc_over_time_pairs_post_tab$year_post2020 = NULL
acc_over_time_pairs_post_tab$df = NULL
acc_over_time_pairs_post_tab$estimate = round(acc_over_time_pairs_post_tab$estimate, 2)
```

```
acc_over_time_pairs_post_tab$SE = round(acc_over_time_pairs_post_tab$SE, 2)
acc_over_time_pairs_post_tab$t.ratio = round(acc_over_time_pairs_post_tab$t.ratio, 2)

colnames(acc_over_time_pairs_post_tab) =
  c("Dataset", "Author Gender Pair Contrast",
    "Estimate", "Std. Error", "t-Statistic", "p-value", "Adj. p-value")

write.csv(acc_over_time_pairs_post_tab,
          file = "acc_over_time_pairs_post_tab.csv", row.names = FALSE)
```

These tables summarize the pairwise results for time to decision and acceptance rates, with an extra column for the adjusted p-values. Separate tables are created for pairwise results within each time period and for evaluating the effect of time period on the pairs.

```
# Time to decision
# Gender pairs within each time period
time_pairs_prepost_gender_tab = cbind(
  data.frame(time_pairs_prepost_gender_robust$contrasts),
  adj.p.value = all_pvals_adj[names(time_pairs_prepost_gender_pvals), "adj_pvals"])
time_pairs_prepost_gender_tab$df = NULL
time_pairs_prepost_gender_tab = time_pairs_prepost_gender_tab %>%
  mutate(time_periods = factor(case_when(
    time_periods == "1" ~ "Pre-pandemic, before discontinuity",
    time_periods == "2" ~ "Pre-pandemic, after discontinuity",
    time_periods == "3" ~ "Pandemic"),
    levels = c("Pre-pandemic, before discontinuity",
               "Pre-pandemic, after discontinuity", "Pandemic"))
  ) %>%
  arrange(time_periods)
time_pairs_prepost_gender_tab[,3:5] = round(time_pairs_prepost_gender_tab[,3:5], 2)
colnames(time_pairs_prepost_gender_tab) =
  c("Author Gender Pair Contrast", "Time Period", "Difference",
    "Std. Error", "t-Statistic", "p-value", "Adj. p-value")
write.csv(time_pairs_prepost_gender_tab,
          file = "time_pairs_prepost_gender_tab.csv", row.names = FALSE)

# Acceptance rate
# Gender pairs within each time period
acc_chance_pairs_prepost_gender_tab = cbind(
  data.frame(acc_chance_pairs_prepost_gender_robust$contrasts),
  adj.p.value = all_pvals_adj[names(acc_chance_pairs_prepost_gender_pvals), "adj_pvals"])
acc_chance_pairs_prepost_gender_tab$df = NULL
acc_chance_pairs_prepost_gender_tab$null = NULL
acc_chance_pairs_prepost_gender_tab = acc_chance_pairs_prepost_gender_tab %>%
  mutate(time_periods = factor(case_when(
    time_periods == "1" ~ "Pre-pandemic, before discontinuity",
    time_periods == "2" ~ "Pre-pandemic, after discontinuity",
    time_periods == "3" ~ "Pandemic"),
    levels = c("Pre-pandemic, before discontinuity",
               "Pre-pandemic, after discontinuity", "Pandemic"))
  ) %>%
  arrange(time_periods)
acc_chance_pairs_prepost_gender_tab[,3:5] = round(acc_chance_pairs_prepost_gender_tab[,3:5], 2)
colnames(acc_chance_pairs_prepost_gender_tab) =
```

```
  c("Author Gender Pair Contrast", "Time Period", "Odds Ratio",
    "Std. Error", "z-Statistic", "p-value", "Adj. p-value")
write.csv(acc_chance_pairs_prepost_gender_tab,
          file = "acc_chance_pairs_prepost_gender_tab.csv", row.names = FALSE)
```

These tables summarize the results from the authorship pair contingency tables, with adjusted p-values.

```
# Pre-pandemic, before discontinuity
authorship_pairs_pre_tab = data.frame(
  Manuscripts = c("Submitted", rep("", 7), "Accepted", rep("", 7)),
  Dataset = rep(c("All", rep("", 3), "Unique", rep("", 3)), 2),
  rbind(
    # Original submissions
    sub_author_by_pair_original_pre,
    # Unique submissions
    sub_author_by_pair_unique_pre,
    # Original acceptances
    acc_author_by_pair_original_pre,
    # Unique acceptances
    acc_author_by_pair_unique_pre),
  adj.p.value = all_pvals_adj[c(
    # Original submissions
    names(sub_author_by_pair_original_pre_pvals),
    # Unique submissions
    names(sub_author_by_pair_unique_pre_pvals),
    # Original acceptances
    names(acc_author_by_pair_original_pre_pvals),
    # Unique acceptances
    names(acc_author_by_pair_unique_pre_pvals)),
    "adj_pvals"])
authorship_pairs_pre_tab$df = NULL
authorship_pairs_pre_tab$First = unlist(authorship_pairs_pre_tab$First)
authorship_pairs_pre_tab$Last = unlist(authorship_pairs_pre_tab$Last)
authorship_pairs_pre_tab$Observed = paste(
  round(as.numeric(authorship_pairs_pre_tab$Observed)*100, 2), "%")
authorship_pairs_pre_tab$Expected = paste(
  round(as.numeric(authorship_pairs_pre_tab$Expected)*100, 2), "%")
authorship_pairs_pre_tab$X.squared = round(as.numeric(authorship_pairs_pre_tab$X.squared), 2)
authorship_pairs_pre_tab$pvalue = as.numeric(authorship_pairs_pre_tab$pvalue)

colnames(authorship_pairs_pre_tab) =
  c("Manuscripts", "Dataset", "First Author", "Last Author",
    "Observed", "Expected", "X2 Statistic", "p-value", "Adj. p-value")

write.csv(authorship_pairs_pre_tab,
          file = "authorship_pairs_pre_tab.csv", row.names = FALSE)


# Pre-pandemic, after discontinuity
authorship_pairs_pre2_tab = data.frame(
  Manuscripts = c("Submitted", rep("", 7), "Accepted", rep("", 7)),
  Dataset = rep(c("All", rep("", 3), "Unique", rep("", 3)), 2),
  rbind(
    # Original submissions
```

```
    sub_author_by_pair_original_pre2,
    # Unique submissions
    sub_author_by_pair_unique_pre2,
    # Original acceptances
    acc_author_by_pair_original_pre2,
    # Unique acceptances
    acc_author_by_pair_unique_pre2),
  adj.p.value = all_pvals_adj[c(
    # Original submissions
    names(sub_author_by_pair_original_pre2_pvals),
    # Unique submissions
    names(sub_author_by_pair_unique_pre2_pvals),
    # Original acceptances
    names(acc_author_by_pair_original_pre2_pvals),
    # Unique acceptances
    names(acc_author_by_pair_unique_pre2_pvals)),
    "adj_pvals"])
authorship_pairs_pre2_tab$df = NULL
authorship_pairs_pre2_tab$First = unlist(authorship_pairs_pre2_tab$First)
authorship_pairs_pre2_tab$Last = unlist(authorship_pairs_pre2_tab$Last)
authorship_pairs_pre2_tab$Observed = paste(
  round(as.numeric(authorship_pairs_pre2_tab$Observed)*100, 2), "%")
authorship_pairs_pre2_tab$Expected = paste(
  round(as.numeric(authorship_pairs_pre2_tab$Expected)*100, 2), "%")
authorship_pairs_pre2_tab$X.squared = round(as.numeric(authorship_pairs_pre2_tab$X.squared), 2)
authorship_pairs_pre2_tab$pvalue = as.numeric(authorship_pairs_pre2_tab$pvalue)

colnames(authorship_pairs_pre2_tab) =
  c("Manuscripts", "Dataset", "First Author", "Last Author",
    "Observed", "Expected", "X2 Statistic", "p-value", "Adj. p-value")

write.csv(authorship_pairs_pre2_tab,
          file = "authorship_pairs_pre2_tab.csv", row.names = FALSE)


# Pandemic
authorship_pairs_post_tab = data.frame(
  Manuscripts = c("Submitted", rep("", 7), "Accepted", rep("", 7)),
  Dataset = rep(c("All", rep("", 3), "Unique", rep("", 3)), 2),
  rbind(
    # Original submissions
    sub_author_by_pair_original_post,
    # Unique submissions
    sub_author_by_pair_unique_post,
    # Original acceptances
    acc_author_by_pair_original_post,
    # Unique acceptances
    acc_author_by_pair_unique_post),
  adj.p.value = all_pvals_adj[c(
    # Original submissions
    names(sub_author_by_pair_original_post_pvals),
    # Unique submissions
    names(sub_author_by_pair_unique_post_pvals),
```

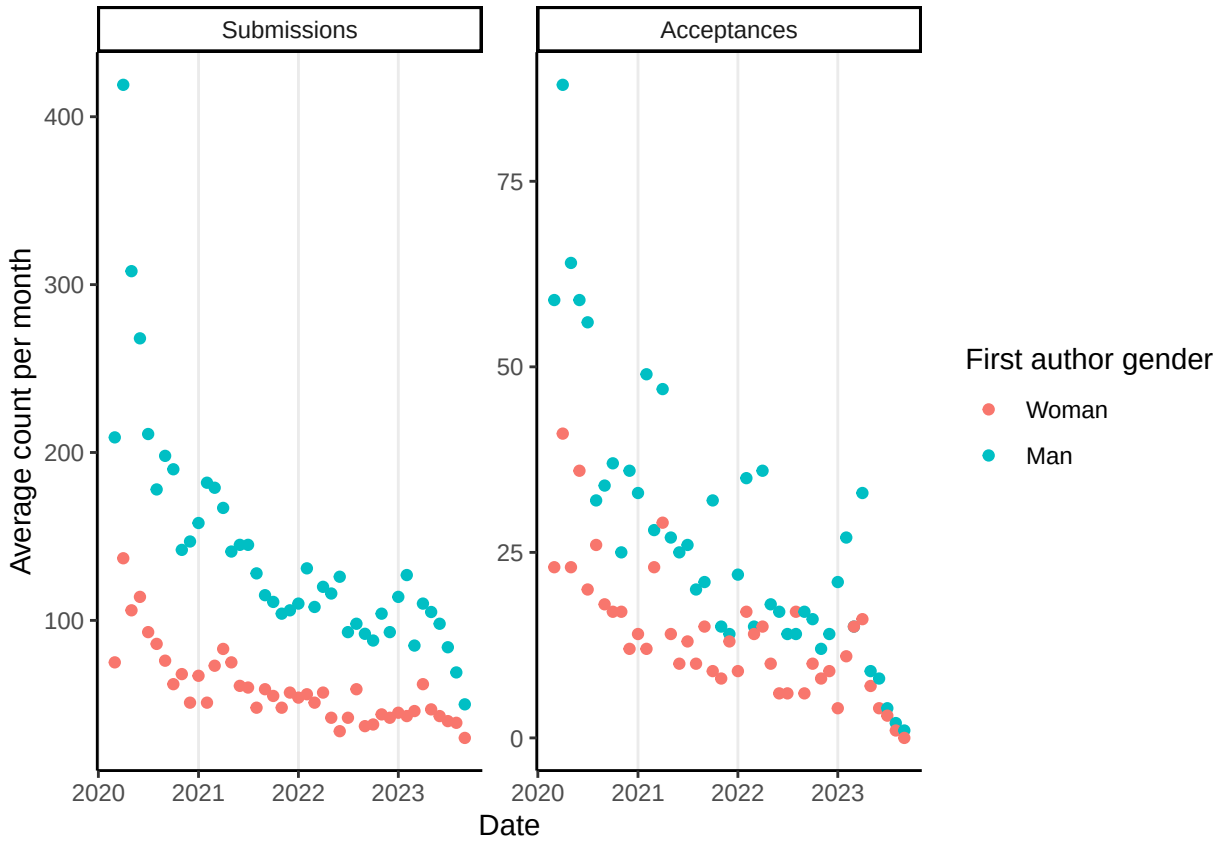
```
    # Original acceptances
    names(acc_author_by_pair_original_post_pvals),
    # Unique acceptances
    names(acc_author_by_pair_unique_post_pvals)),
    "adj_pvals"])
authorship_pairs_post_tab$df = NULL
authorship_pairs_post_tab$First = unlist(authorship_pairs_post_tab$First)
authorship_pairs_post_tab$Last = unlist(authorship_pairs_post_tab$Last)
authorship_pairs_post_tab$Observed = paste(
  round(as.numeric(authorship_pairs_post_tab$Observed)*100, 2), "%")
authorship_pairs_post_tab$Expected = paste(
  round(as.numeric(authorship_pairs_post_tab$Expected)*100, 2), "%")
authorship_pairs_post_tab$X.squared = round(as.numeric(authorship_pairs_post_tab$X.squared), 2)
authorship_pairs_post_tab$pvalue = as.numeric(authorship_pairs_post_tab$pvalue)

colnames(authorship_pairs_post_tab) =
  c("Manuscripts", "Dataset", "First Author", "Last Author",
    "Observed", "Expected", "X2 Statistic", "p-value", "Adj. p-value")

write.csv(authorship_pairs_post_tab,
          file = "authorship_pairs_post_tab.csv", row.names = FALSE)
```

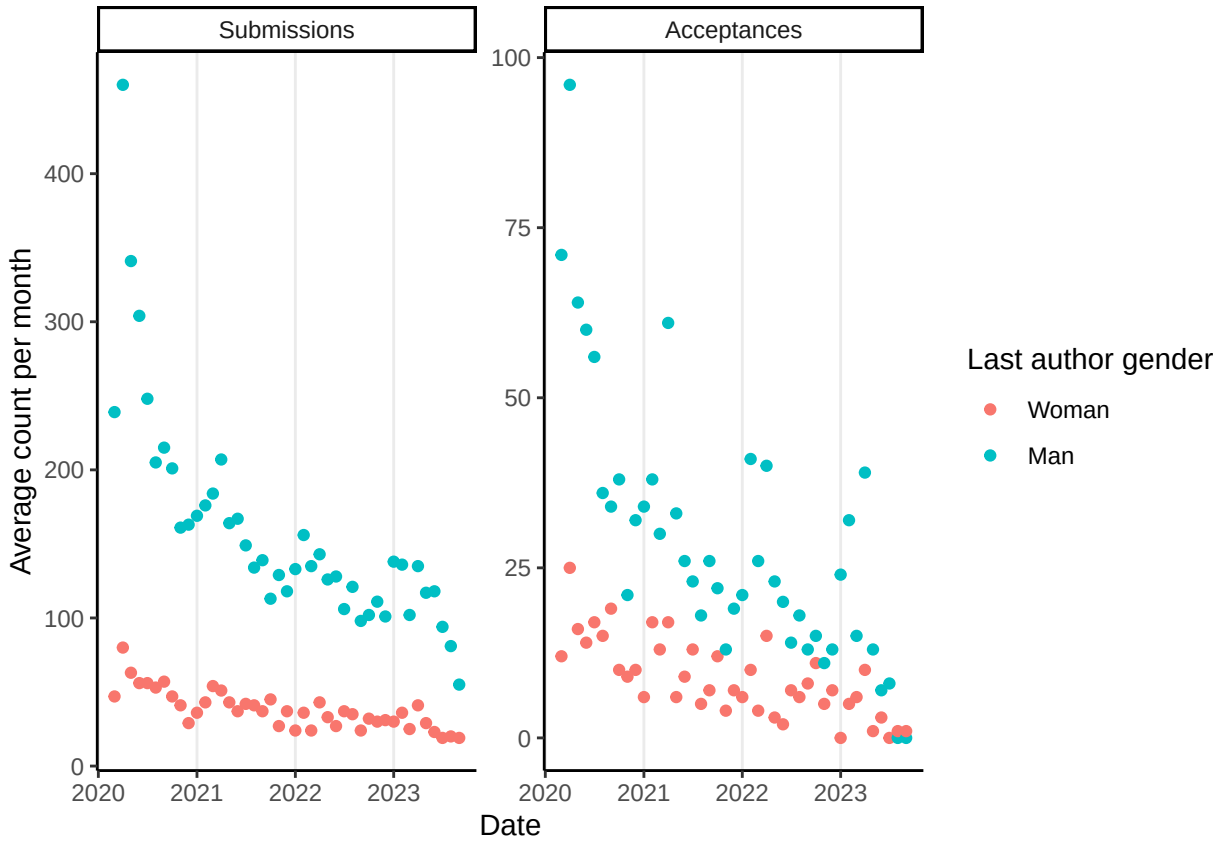
Pandemic-era submissions and acceptances by month

```
# First authors
df_original %>%
  mutate(month_date = lubridate::floor_date(as_date(first_receipt_date), "month")) %>%
  drop_na(first_author_gender) %>%
  filter(first_receipt_date >= as.Date("2020-03-01") &
           first_receipt_date < as.Date("2023-10-01")) %>%
  group_by(month_date, first_author_gender) %>%
  summarise(submitted = sum(submitted), accepted = sum(accepted)) %>%
  ungroup() %>%
  pivot_longer(cols = -c("month_date", "first_author_gender")) %>%
  mutate(name = factor(ifelse(name == "submitted", "Submissions", "Acceptances"),
                       levels = c("Submissions", "Acceptances")),
         `First author gender` = factor(first_author_gender,
                                        levels = c("Woman", "Man"))) %>%
  ggplot(aes(x = month_date, y = value, col = `First author gender`)) +
  geom_point() +
  facet_wrap(~name, scales = "free_y") +
  scale_x_date(minor_breaks = "1 year") +
  theme(panel.grid.major.x = element_line(),
        panel.grid.minor.x = element_line()) +
  xlab("Date") + ylab("Average count per month")
```




```r
ggsave("pandemic_monthly_rates_first.png")

# Last authors
df_original %>%
  mutate(month_date = lubridate::floor_date(as_date(first_receipt_date), "month")) %>%
  drop_na(last_author_gender) %>%
  filter(first_receipt_date >= as.Date("2020-03-01") &
           first_receipt_date < as.Date("2023-10-01")) %>%
  group_by(month_date, last_author_gender) %>%
  summarise(submitted = sum(submitted), accepted = sum(accepted)) %>%
  ungroup() %>%
  pivot_longer(cols = -c("month_date", "last_author_gender")) %>%
  mutate(name = factor(ifelse(name == "submitted", "Submissions", "Acceptances"),
                       levels = c("Submissions", "Acceptances")),
         `Last author gender` = factor(last_author_gender,
                                       levels = c("Woman", "Man"))) %>%
  ggplot(aes(x = month_date, y = value, col = `Last author gender`)) +
  geom_point() +
  facet_wrap(~name, scales = "free_y") +
  scale_x_date(minor_breaks = "1 year") +
  theme(panel.grid.major.x = element_line(),
        panel.grid.minor.x = element_line()) +
  xlab("Date") + ylab("Average count per month")
```




```
ggsave("pandemic_monthly_rates_last.png")
```