The impact of double blind reviewing at EvoLang 11: statistics

Introduction

Note that the analysis here has changed since the original submission, updating three data points with incorrectly assigned genders and adding an extra analysis with deviation coding instead of dummy coding. The original script and results can be found in the file 'gendercheck_stats3_original.html'.

Data

This script uses two data files.

AllConferenceData.csv

- Score.Mean: Mean raw score given by reviewers (scaled between 0 and 1, hierh = better paper)
- FirstAuthorGender: Gender of first author
- conference: Which conference the paper was submitted to
- rank: scaled rank within each conference (higher = better paper)
- year: Year number (1-3, for convenience)
- review: Type of review (Single / Double blind)
- po: Number of conferences the author has previously submitted to

sameAuthorData_3conferences_Differences.csv

- X: arbitrary author number
- e10: best rank for first-authored paper by this author in EvoLang 10
- e11: best rank for first-authored paper by this author in EvoLang 11
- gender: gender of author
- student: whether the author was a student *in EvoLang 10*
- maletype: The strength of the male typing of the topic of the paper (see supplementary materials)
- diff10_11: The difference between e10 and e11, for convenience

This data is also available in a different format in "sameAuthorData_3conferences.csv", which also contains data from EvoLang 9. In the main analysis, it was decided that there were too few authors who applied to all three conferences to take advantage of this.

Loading data for first analysis

Load libraries.

```r
# Load data
library(lattice)
library(ggplot2)
library(gplots)
library(xtable)
library(party)
```

```r
# read data
allData = read.csv("AllConferenceData.csv",stringsAsFactors = F)
# relabel factor
allData$FirstAuthorGender = factor(allData$FirstAuthorGender,labels=c("Female","Male"))
allData$review = factor(allData$review,levels = c("Single","Double"))
allData$conference = factor(allData$conference,levels = c("EvoLang9","EvoLang10","EvoLang11"))
allData$student = factor(allData$student)
# get rid of unneed columns
allData = allData[,!names(allData) %in% c("AuthorCode",'maletype','X')]

allData$po = allData$po -1

allData$conference.name = c(EvoLang9="E9", EvoLang10="E10", EvoLang11="E11")[allData$conference]
allData$conference.name = factor(allData$conference.name,levels = c("E9","E10","E11"))
```

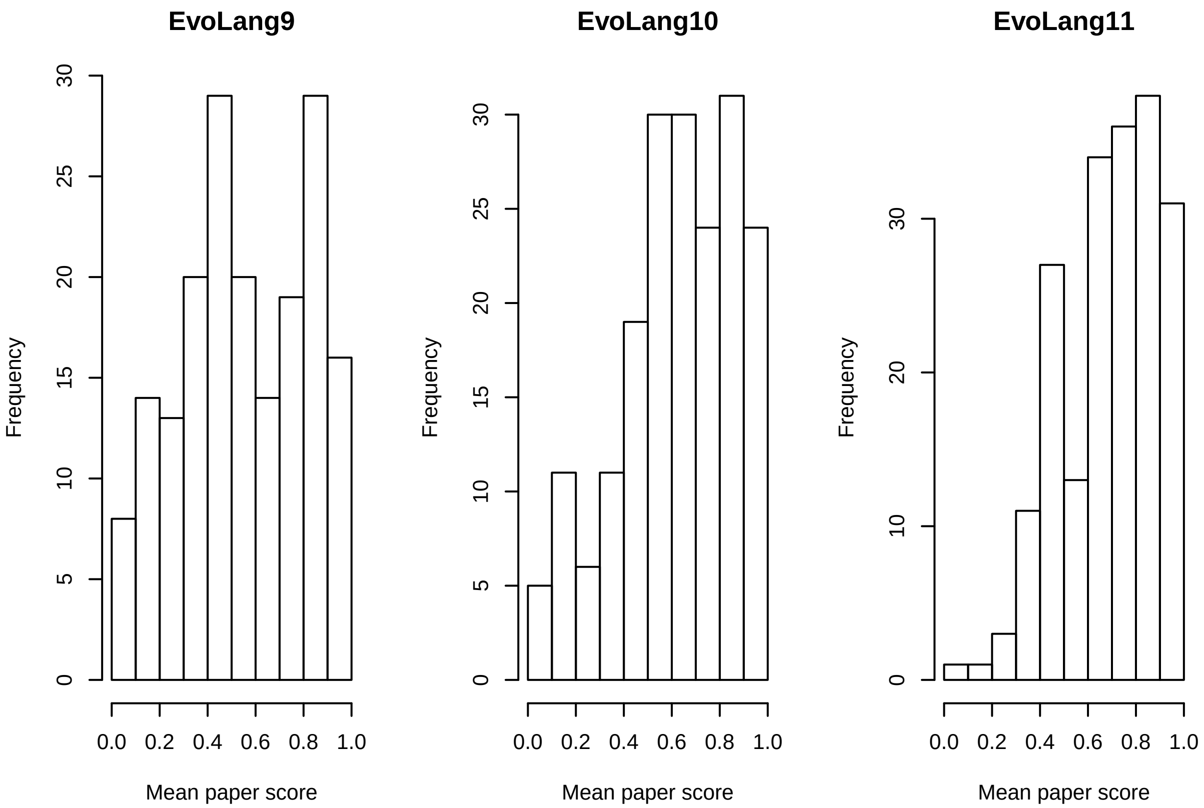
Look at the distribution of submissions:

```r
table(allData$conference, allData$FirstAuthorGender, allData$student)
```

```
## , ,  = Non-Student
## 
##            
##             Female Male
##   EvoLang9      34   85
##   EvoLang10     55   94
##   EvoLang11     40   78
## 
## , ,  = Student
## 
##            
##             Female Male
##   EvoLang9      18   45
##   EvoLang10     12   30
##   EvoLang11     35   42
```

Look at the distributions of raw scores by conference:

```r
# Plot data
# Histograms of raw score means by conference
#pdf("Hist_3conf.pdf", height=4, width=12)
par(mfrow=c(1,3))
for(i in levels(allData$conference)){
  hist(allData[allData$conference==i,]$Score.Mean, xlab="Mean paper score", main=i)
}
```



```
#dev.off()
par(mfrow=c(1,1))
```

The raw scores are shifted to the right for EvoLang 10 and 11:

```
summary(aov(Score.Mean ~ conference,data=allData))
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## conference    2   1.93  0.9645   16.25 1.37e-07 ***
## Residuals   565  33.53  0.0593
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

So we'll use normalised rank instead of raw score.

Review ranks by gender and student status

```
m = aov(rank~(FirstAuthorGender*conference*student), data = allData)
summary(m)
```

```
##                                          Df Sum Sq Mean Sq F value  Pr(>F)
## FirstAuthorGender                         1   0.48  0.4800   5.636 0.01793 *
## conference                                2   0.21  0.1040   1.222 0.29557
## student                                   1   0.09  0.0908   1.066 0.30225
## FirstAuthorGender:conference              2   0.98  0.4899   5.752 0.00337 **
## FirstAuthorGender:student                 1   0.08  0.0789   0.926 0.33632
## conference:student                        2   0.34  0.1699   1.995 0.13703
## FirstAuthorGender:conference:student      2   0.01  0.0037   0.043 0.95764
## Residuals                          556  47.36  0.0852
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Post hoc t-tests

Whole data:

```
t.test(rank~FirstAuthorGender, data=allData)
```

```
##
##  Welch Two Sample t-test
##
## data:  rank by FirstAuthorGender
## t = 2.3523, df = 390.29, p-value = 0.01915
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.01006582 0.11253854
## sample estimates:
## mean in group Female   mean in group Male
##            0.5412323            0.4799301
```

Compare genders within each conference:

```
t.test.string = function(tx){
  t = signif(tx$statistic,2)
  df = tx$parameter['df']
  p = signif(tx$p.value,3)
  est = signif(diff(tx$estimate),2)

  paste("(difference in means = ",est,", t = ",t,", p = ",p,")",sep = "")
}

# EvoLang 9
t.test.string(t.test(rank~FirstAuthorGender, data=allData[allData$conference=='EvoLang9',]))
```

```
## [1] "(difference in means = 0.043, t = -0.87, p = 0.386)"
```

```
# EvoLang 10
t.test.string(t.test(rank~FirstAuthorGender, data=allData[allData$conference=='EvoLang10',]))
```

```
## [1] "(difference in means = -0.036, t = 0.75, p = 0.454)"
```

```
# EvoLang 11
t.test.string(t.test(rank~FirstAuthorGender, data=allData[allData$conference=='EvoLang11',]))
```

```
## [1] "(difference in means = -0.17, t = 4.5, p = 1.55e-05)"
```

Number of times submitted

The variable *po* specifies the number of conferences that the author has submitted to, up to the date of the particular submission. (so 1= first submission year to EvoLang, 2 = submitting for the second conference, 3 = submitting for E11, and has also submitted for E10 and E9).

```r
m.first = aov(rank~(FirstAuthorGender*conference*student*po),
              data = allData)
summary(m.first)
```

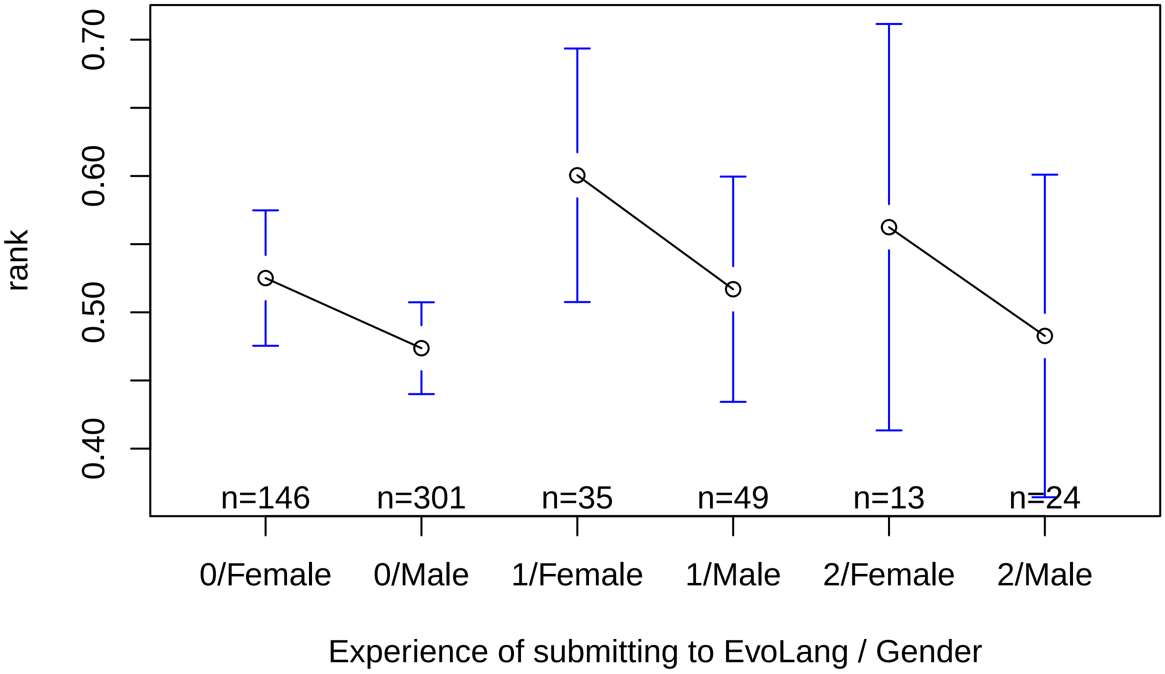
```
##                                           Df Sum Sq Mean Sq F value  Pr(>F)
## FirstAuthorGender                          1   0.48  0.4800   5.674 0.01755
## conference                                 2   0.21  0.1040   1.230 0.29315
## student                                    1   0.09  0.0908   1.073 0.30063
## po                                         1   0.28  0.2843   3.361 0.06732
## FirstAuthorGender:conference               2   0.96  0.4803   5.677 0.00363
## FirstAuthorGender:student                  1   0.10  0.0981   1.160 0.28191
## conference:student                         2   0.28  0.1419   1.677 0.18787
## FirstAuthorGender:po                       1   0.10  0.0999   1.181 0.27763
## conference:po                              1   0.02  0.0158   0.186 0.66628
## student:po                                 1   0.39  0.3889   4.598 0.03246
## FirstAuthorGender:conference:student       2   0.00  0.0015   0.017 0.98275
## FirstAuthorGender:conference:po            1   0.12  0.1195   1.413 0.23508
## FirstAuthorGender:student:po               1   0.10  0.1014   1.199 0.27410
## conference:student:po                      1   0.01  0.0052   0.061 0.80482
## FirstAuthorGender:conference:student:po    1   0.04  0.0418   0.494 0.48236
## Residuals                                548  46.36  0.0846
##
## FirstAuthorGender                         *
## conference
## student
## po                                        .
## FirstAuthorGender:conference              **
## FirstAuthorGender:student
## conference:student
## FirstAuthorGender:po
## conference:po
## student:po                                *
## FirstAuthorGender:conference:student
## FirstAuthorGender:conference:po
## FirstAuthorGender:student:po
## conference:student:po
## FirstAuthorGender:conference:student:po
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(lm(rank~(FirstAuthorGender*conference) + (student*po),data=allData))
```

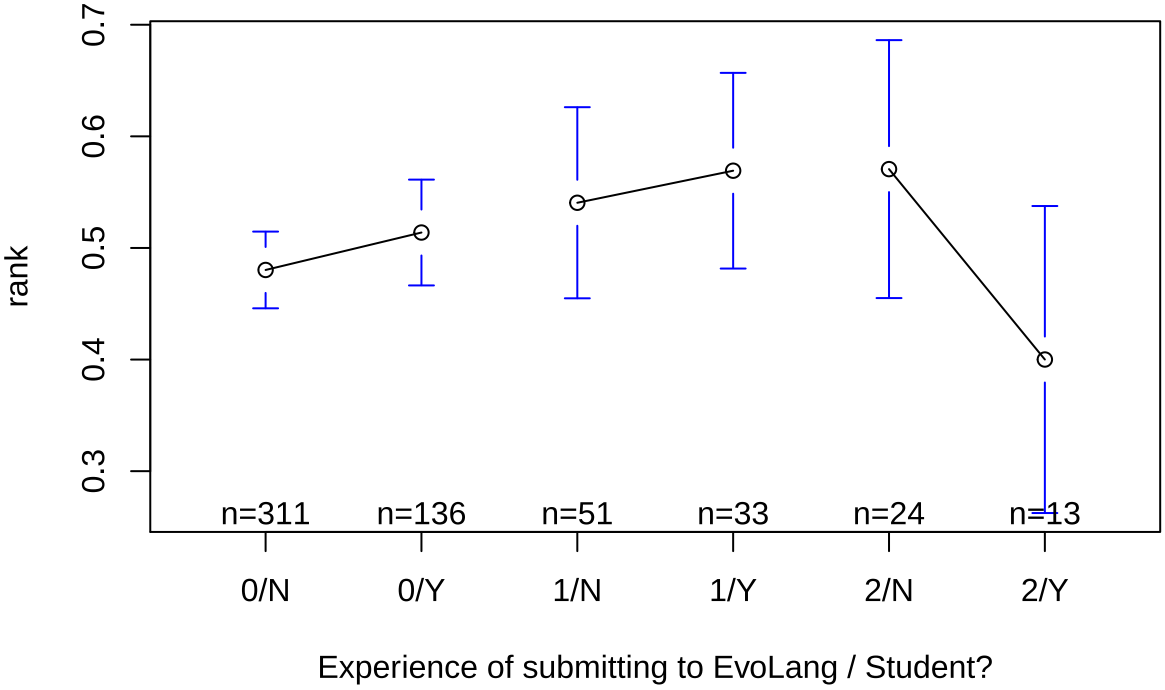
```
##
## Call:
## lm(formula = rank ~ (FirstAuthorGender * conference) + (student *
##     po), data = allData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.56324 -0.25322  0.00196  0.24481  0.52597
##
## Coefficients:
##                                             Estimate Std. Error t value
## (Intercept)                              0.45218    0.04166  10.855
## FirstAuthorGenderMale                    0.04320    0.04773   0.905
## conferenceEvoLang10                      0.08068    0.05419   1.489
## conferenceEvoLang11                      0.08578    0.05487   1.563
## studentStudent                           0.04376    0.03006   1.456
## po                                       0.06862    0.02891   2.373
## FirstAuthorGenderMale:conferenceEvoLang10 -0.08144    0.06501  -1.253
## FirstAuthorGenderMale:conferenceEvoLang11 -0.20896    0.06423  -3.253
## studentStudent:po                        -0.07244    0.04484  -1.616
##                                          Pr(>|t|)
## (Intercept)                               < 2e-16 ***
## FirstAuthorGenderMale                     0.36574
## conferenceEvoLang10                       0.13708
## conferenceEvoLang11                       0.11855
## studentStudent                            0.14597
## po                                        0.01797 *
## FirstAuthorGenderMale:conferenceEvoLang10  0.21080
## FirstAuthorGenderMale:conferenceEvoLang11  0.00121 **
## studentStudent:po                         0.10672
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2909 on 559 degrees of freedom
## Multiple R-squared:  0.04531,    Adjusted R-squared:  0.03165
## F-statistic: 3.316 on 8 and 559 DF,  p-value: 0.001026
```

We see that there is a weak effect of *po* on rank: more experienced submitters do better. There's also a weak interaction between student status and po: the effect of experience is greater for non-students.

```
plotmeans(rank~paste(po,FirstAuthorGender,sep='/'), data=allData,
          xlab='Experience of submitting to EvoLang / Gender',
          connect = list(1:2,3:4,5:6))
```



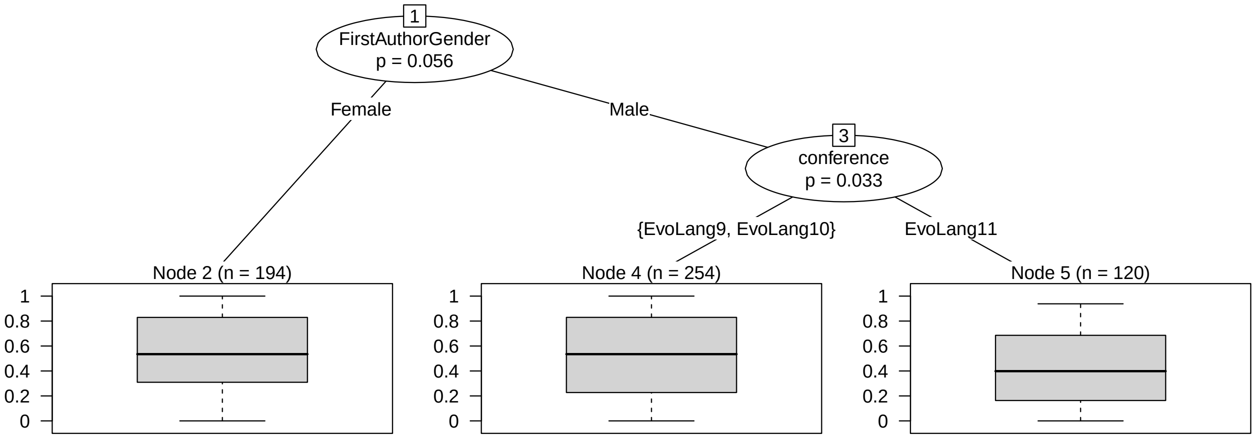
```
studentYN = c("N",'Y')[1 + as.numeric(allData$student=="Student")]
plotmeans(rank~paste(po,studentYN,sep='/'), data=allData,
          xlab='Experience of submitting to EvoLang / Student?',
          connect = list(1:2,3:4,5:6))
```



Decision tree

Use classification trees to explore data. We've changed the minimum criterion from 95% to 94% in order to show some of the structure.

```
x = ctree(rank~FirstAuthorGender+conference+student,data=allData, controls=ctree_control(mincriterion =
plot(x)
```



This is broadly in agreement with regressions above. It suggests that female first authored papers are generally higher, but ranking of male papers declines in EvoLang 11.

Data on same authors: differences between E10 and E11

```
sameAuthorData3 = read.csv("sameAuthorData_3conferences_Differences.csv", stringsAsFactors = T)
sameAuthorData3$po = sameAuthorData3$po - 2
```

The original results reported in the paper used a linear regression with dummy coding:

```
summary(lm(diff10_11~gender*student, data=sameAuthorData3))
```

```
## 
## Call:
## lm(formula = diff10_11 ~ gender * student, data = sameAuthorData3)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.80827 -0.17480 -0.00854  0.18701  0.80418 
## 
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)  
## (Intercept)                0.03849    0.06667   0.577   0.5656  
## genderMale                -0.19389    0.08820  -2.198   0.0313 *
## studentStudent            -0.20099    0.12693  -1.583   0.1179  
## genderMale:studentStudent  0.34827    0.15900   2.190   0.0319 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.3055 on 69 degrees of freedom
## Multiple R-squared:  0.08273,    Adjusted R-squared:  0.04284 
## F-statistic: 2.074 on 3 and 69 DF,  p-value: 0.1116
```

However, the data appears more like an interaction with no main effect. Indeed, when using deviation coding, this is the case:

```
summary(lm(diff10_11~gender*student,
           data=sameAuthorData3,
           contrasts=list(
             gender=contr.sum(2)/2,
             student=contr.sum(2)/2)
  ))
```

```
## 
## Call:
## lm(formula = diff10_11 ~ gender * student, data = sameAuthorData3, 
##     contrasts = list(gender = contr.sum(2)/2, student = contr.sum(2)/2))
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.80827 -0.17480 -0.00854  0.18701  0.80418 
## 
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)  
## (Intercept)      -0.07188    0.03975  -1.808   0.0749 .
## gender1           0.01976    0.07950   0.249   0.8044  
## student1          0.02685    0.07950   0.338   0.7365  
## gender1:student1  0.34827    0.15900   2.190   0.0319 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3055 on 69 degrees of freedom
## Multiple R-squared:  0.08273,    Adjusted R-squared:  0.04284
## F-statistic: 2.074 on 3 and 69 DF,  p-value: 0.1116
```

Check effect of po:

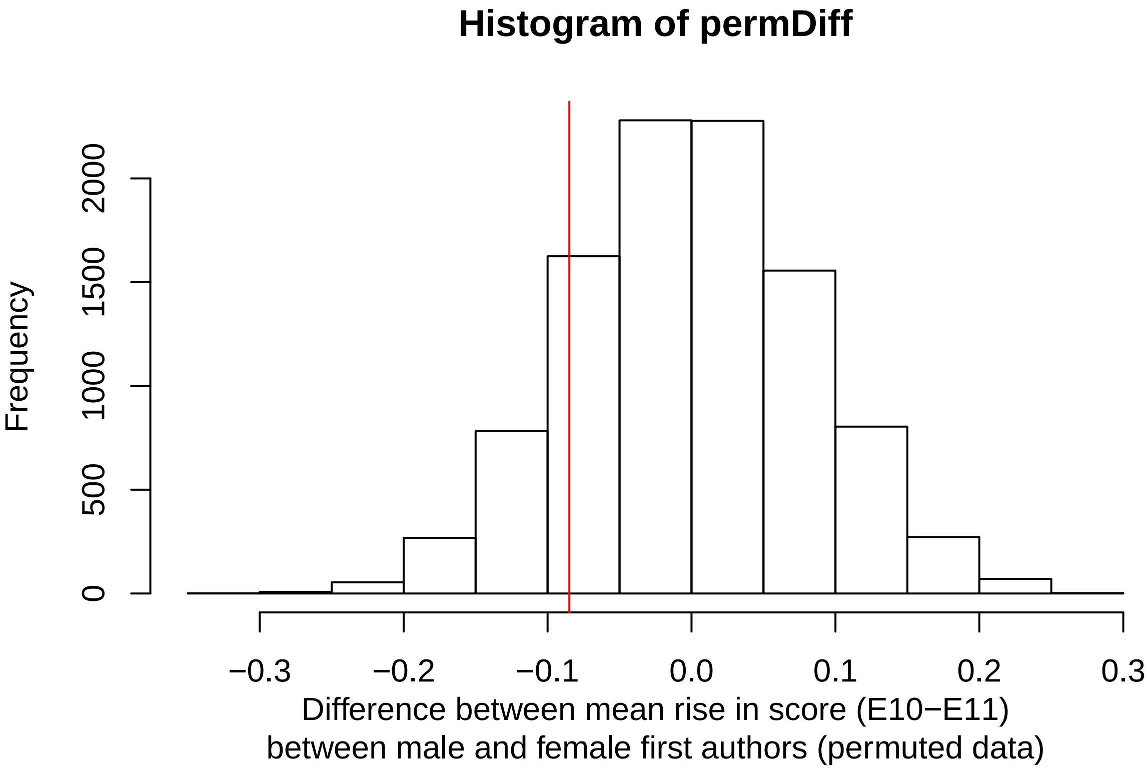
```r
summary(lm(diff10_11~gender*student + (po*gender),
           data=sameAuthorData3,
           contrasts=list(
             gender=contr.sum(2)/2,
             student=contr.sum(2)/2)
  ))
```

```
##
## Call:
## lm(formula = diff10_11 ~ gender * student + (po * gender), data = sameAuthorData3,
##     contrasts = list(gender = contr.sum(2)/2, student = contr.sum(2)/2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7896 -0.1600 -0.0197  0.2047  0.7777
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.10132    0.05201  -1.948   0.0556 .
## gender1           0.02088    0.10402   0.201   0.8415
## student1          0.04116    0.08217   0.501   0.6181
## po                0.06940    0.07930   0.875   0.3846
## gender1:student1  0.36066    0.16435   2.194   0.0317 *
## gender1:po        0.01771    0.15860   0.112   0.9114
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3082 on 67 degrees of freedom
## Multiple R-squared:  0.09343,    Adjusted R-squared:  0.02577
## F-statistic: 1.381 on 5 and 67 DF,  p-value: 0.2426
```

Permutation test

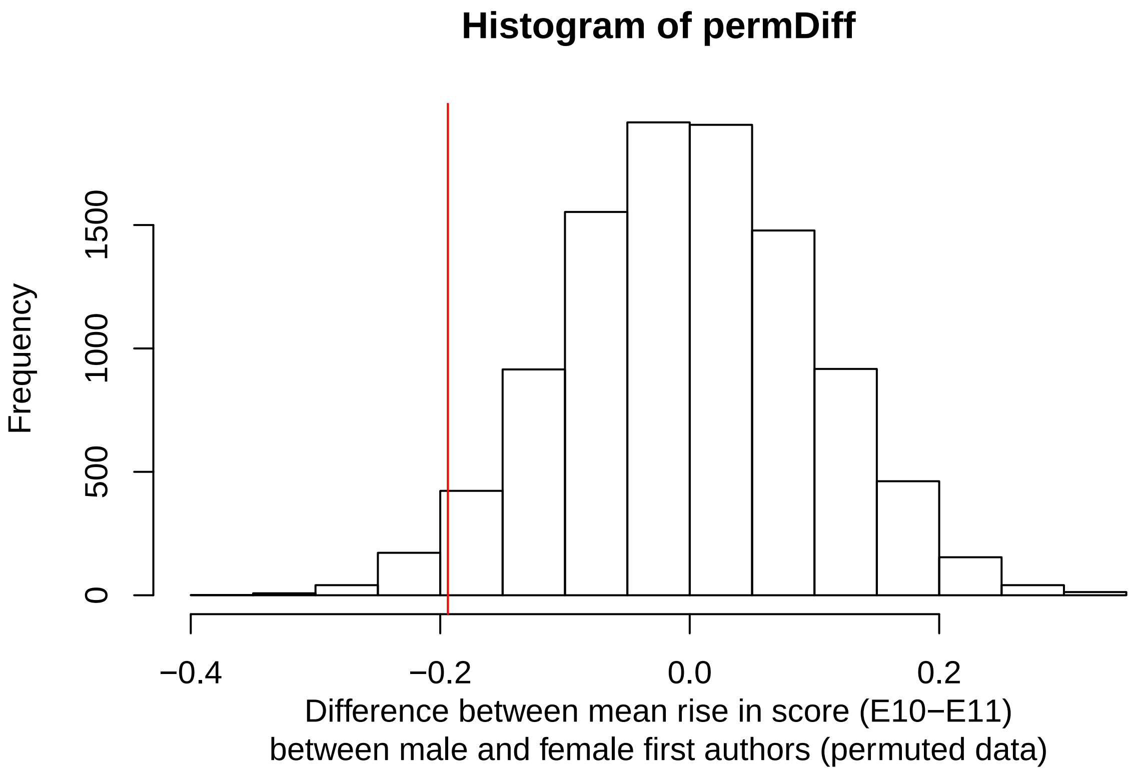
```r
#permutation test functions
perm = function(dx){
  # balance sample size
  min.n = min(table(dx$gender))
  diff(tapply(
    dx$diff10_11,
    sample(dx$gender),
    function(X){
      mean(sample(X,min.n))
    }))
}
perm.test = function(dx){
  trueDiff =diff(tapply(
    dx$diff10_11,
    dx$gender,
    mean))
  permDiff = replicate(10000, perm(dx))
  hist(permDiff, xlab='Difference between mean rise in score (E10-E11)\nbetween male and female first au
  abline(v=trueDiff,col=2)
  p = sum(permDiff < trueDiff) / length(permDiff)
  z = (trueDiff - mean(permDiff))/sd(permDiff)
  print(paste("Difference between true and permuted data: z = ",round(z,3),", p = ",round(p,3)))
}
```

```
par(mfrow=c(1,1))
set.seed(6789)
# Permutation test for whole data
perm.test(sameAuthorData3)
```



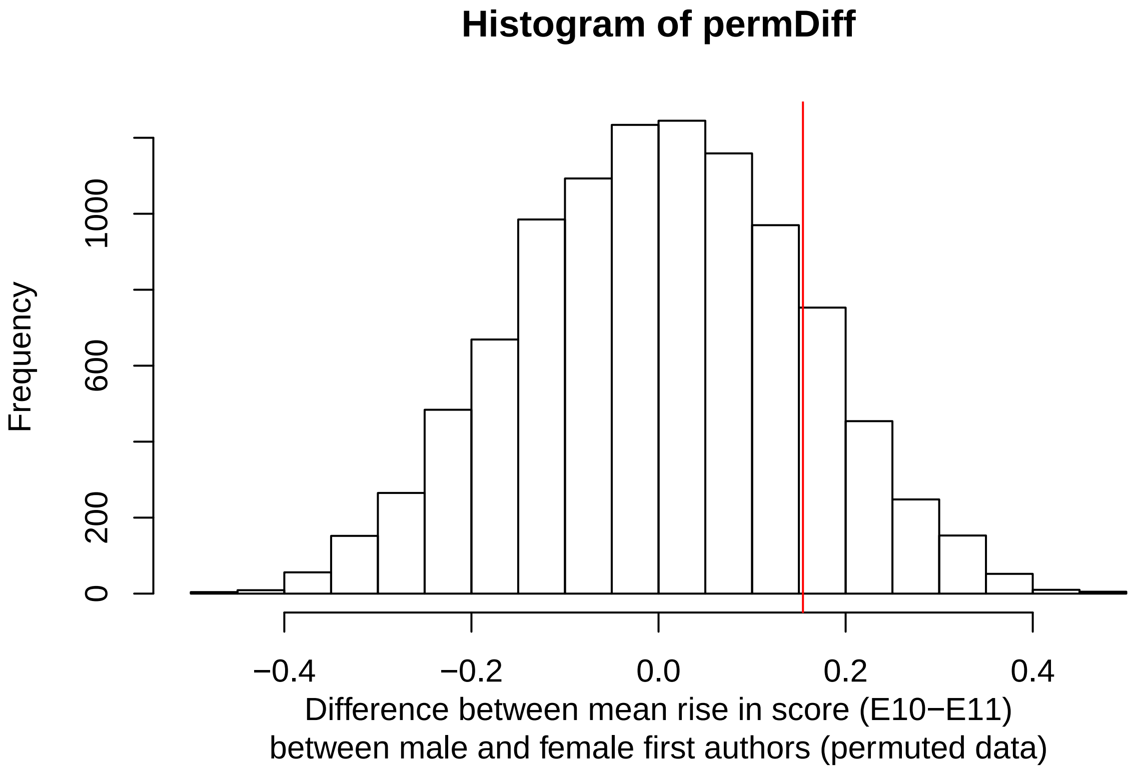
```
## [1] "Difference between true and permuted data: z =  -1.033 , p =  0.152"
```

```
# Only for non-students
perm.test(sameAuthorData3[sameAuthorData3$student=="Non-Student",])
```



```
## [1] "Difference between true and permuted data: z =  -1.952 , p =  0.025"
```

```
# Only for students
perm.test(sameAuthorData3[sameAuthorData3$student=="Student",])
```



```
## [1] "Difference between true and permuted data: z =  1.024 , p =  0.84"
```

Plots

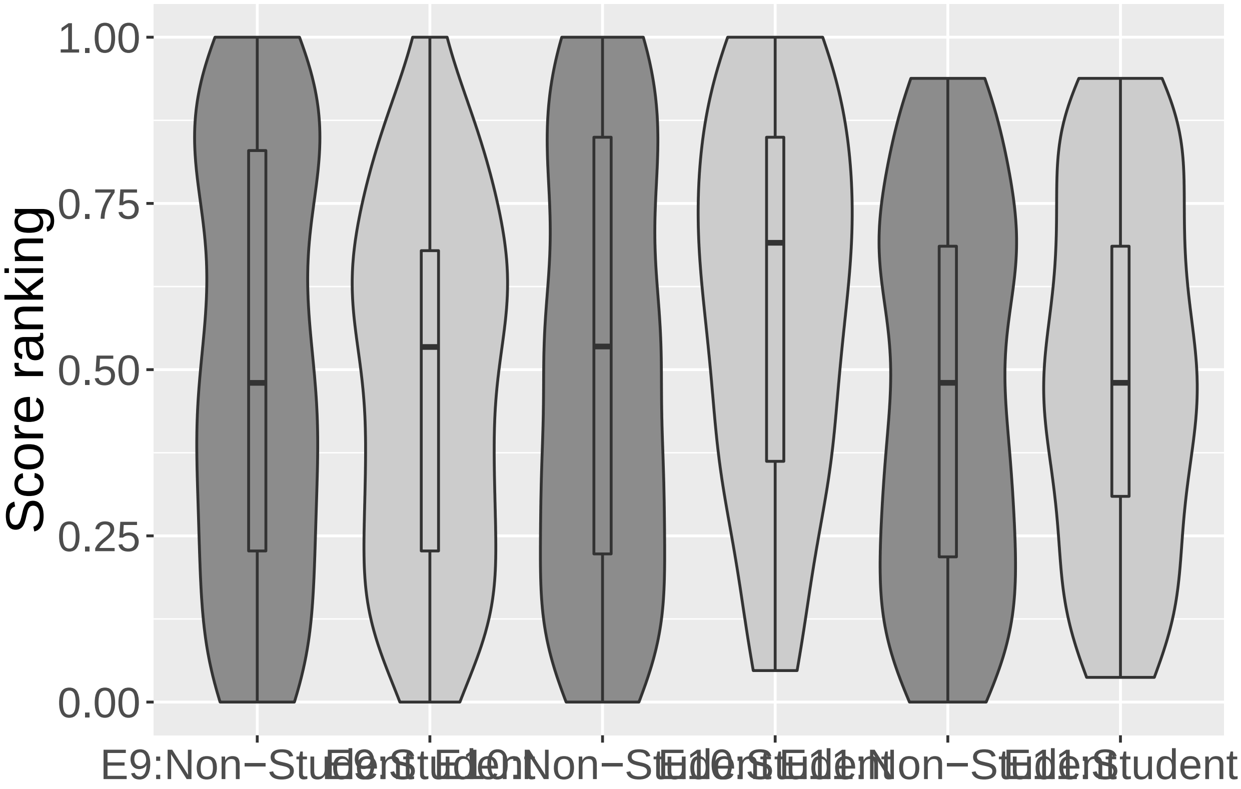
Plot data: Violin plots of rank by student status in each conference.

```
source("summarySE.r")



p <- ggplot(allData[complete.cases(allData),], aes(factor(conference.name):factor(student), rank, fill=

p <- p + geom_violin() + geom_boxplot(width=0.1) +
  theme(text=element_text(size=20), legend.position="none") +
  scale_y_continuous(name="Score ranking")+
  scale_x_discrete(name="")+
  scale_fill_grey(start = 0.55, end=0.8)
p
```



```
pdf("Results_Student_3conf.pdf", width = 12, height= 6)
p
dev.off()
```

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

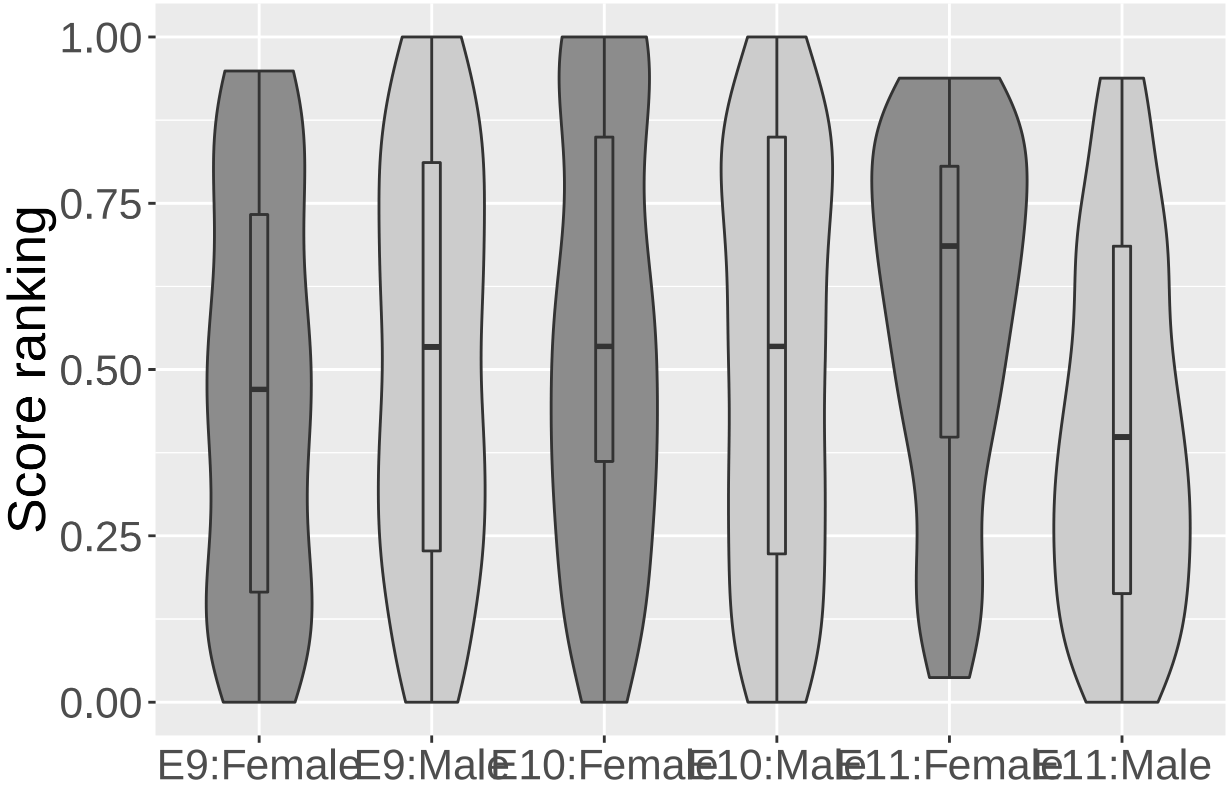
Rank by gender

```
# Plot data:  Violin plots of rank by gender in E10 and E11

p2 <- ggplot(allData[!is.na(allData$FirstAuthorGender),], aes(factor(conference.name):factor(FirstAutho

p2 <- p2 + geom_violin() + geom_boxplot(width=0.1) +
  theme(text=element_text(size=20), legend.position="none") +
  scale_y_continuous(name="Score ranking")+
  scale_x_discrete(name="")+
  scale_fill_grey(start = 0.55, end=0.8)

p2
```



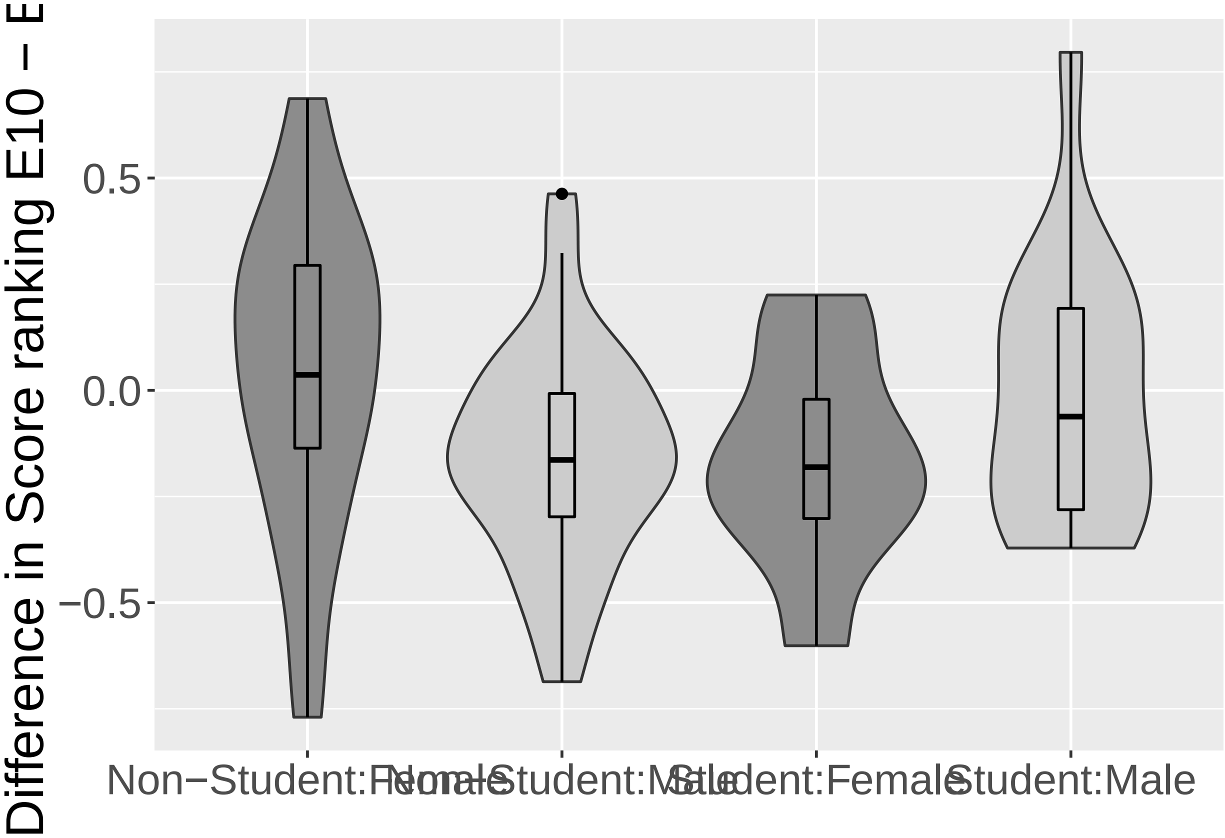
```
pdf("Results_Gender_3conf.pdf", width = 12, height= 6)
p2
dev.off()
```

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

Improvement between E10 and E11 for paired data.

```
p3 <- ggplot(sameAuthorData3, aes(factor(student):factor(gender), diff10_11, fill=gender))

p3 <- p3 + geom_violin() + geom_boxplot(width=0.1, col=1) +
  theme(text=element_text(size=20), legend.position="none") +
  scale_y_continuous(name="Difference in Score ranking E10 - E11")+
  scale_x_discrete(name="")+
  scale_fill_grey(start = 0.55, end=0.8)
p3
```



```
pdf("Improvement_3conf.pdf", width = 12, height= 6)
p3
dev.off()
```

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