Gender balance between major and minor characters

Introduction

This report explores whether the gender bias in the data as a whole is also observed when looking at either only major characters or only minor characters. Major characters may be playable characters, party characters, characters more central to the plot, or characters that speak more. One concern is that dialogue data for major characters is more complete and that gender is easier to code. This might bias estimates of the proportion of dialogue by female characters.

The analyses below demonstrate that conclusions about the proportion of female dialogue are unlikely to be affected by such concerns in our data.

Gender bias in the dialogue of major and minor characters

We first look at major and minor characters as discrete groups.

Load libraries:

```
library(rjson)
library(ggplot2)
library(ggstance)
library(mgcv)
library(knitr)
library(betareg)
```

Load data (for games with coded main player characters):

```
folders = list.dirs("../data", recursive = T)
folders = folders[sapply(folders,function(X){
    "stats_by_character.csv" %in% list.files(X)
  })]


allGames = NULL
for(folder in folders){
  shortName = tail(strsplit(folder,"/")[[1]],1)
  js = fromJSON(file = paste0(folder,"/meta.json"))
  alternativeMeasure = FALSE
  if(!is.null(js$alternativeMeasure)){
    alternativeMeasure = js$alternativeMeasure
  }
  if(!alternativeMeasure){
    statsByChar = read.csv(paste0(folder,"/stats_by_character.csv"),stringsAsFactors = F)
    statsByChar = statsByChar[!is.na(statsByChar$words),]
    statsByChar = statsByChar[statsByChar$words>0,]

    if(nrow(statsByChar)>0 && !is.null(js$mainPlayerCharacters)){
      majc = statsByChar$charName %in% js$mainPlayerCharacters
      minc = (!statsByChar$charName %in% js$mainPlayerCharacters) &
        (statsByChar$group %in% c("male","female"))
      majc.Female = statsByChar$group=="female" & majc
      majc.Male = statsByChar$group=="male" & majc
      minc.Female =  statsByChar$group=="female" & minc
```

```
      minc.Male =  statsByChar$group=="male" & minc

      # Only include games with coded major characters
      if((sum(majc.Female) + sum(majc.Male))> 0 ){
        print(folder)
        majc.Female.words = sum(statsByChar[majc.Female,]$words)
        propFemaleDialogue.mainChar = 0
        if(majc.Female.words>0){
         propFemaleDialogue.mainChar =
           sum(statsByChar[majc.Female,]$words) /
             (sum(statsByChar[majc.Female,]$words) +
              sum(statsByChar[majc.Male,]$words))
        }
        propFemaleDialogue.minorChar =
          sum(statsByChar[minc.Female,]$words) /
            (sum(statsByChar[minc.Female,]$words) +
             sum(statsByChar[minc.Male,]$words))

        ret = data.frame(
          folder = folder,
          game = js$game,
          shortName = shortName,
          group = c("major","minor"),
          numFemaleWords = c(
            sum(statsByChar[majc.Female,]$words),
            sum(statsByChar[minc.Female,]$words)),
          numMaleWords = c(
            sum(statsByChar[majc.Male,]$words),
            sum(statsByChar[minc.Male,]$words)),
          propFemaleDialogue = c(
            propFemaleDialogue.mainChar,
            propFemaleDialogue.minorChar
          ),
          numFemaleCharacters = c(sum(majc.Female),sum(minc.Female)),
          numMaleCharacters = c(sum(majc.Male),sum(minc.Male))
        )
        allGames = rbind(allGames, ret)
      }
    }
  }
}
```

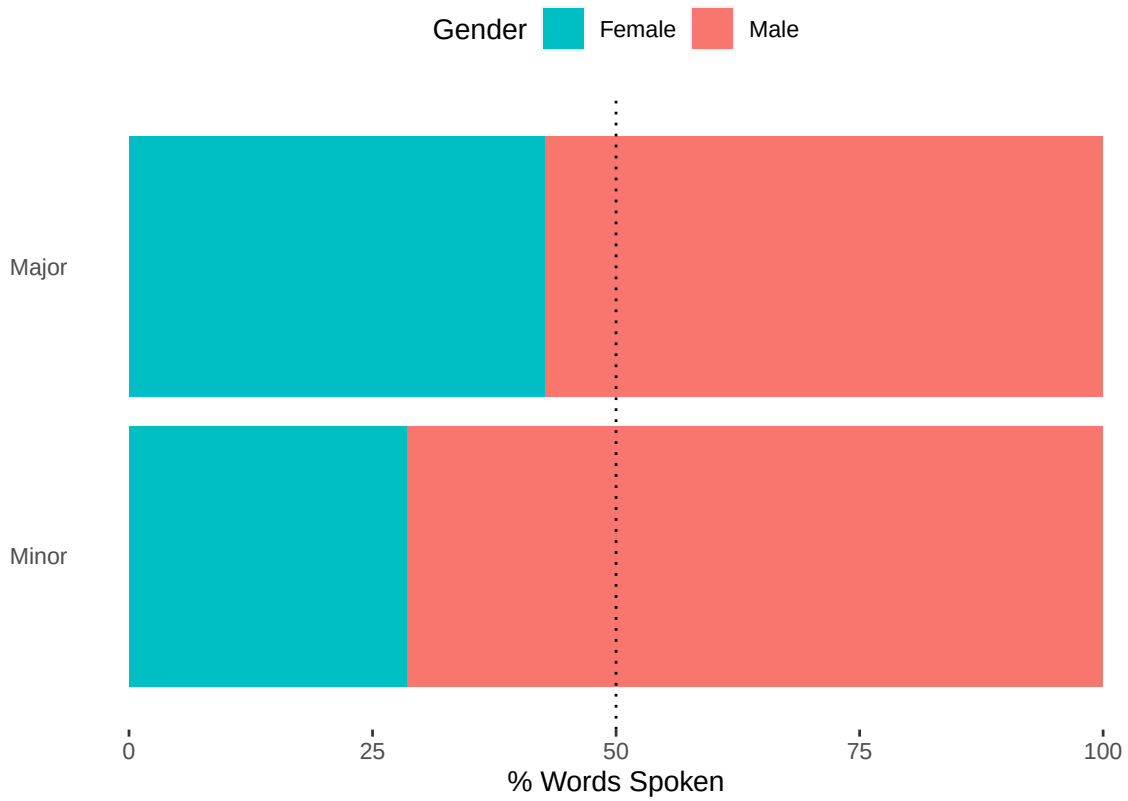
```
## [1] "../data/ChronoTrigger/ChronoTrigger"
## [1] "../data/DragonAge/DragonAgeOrigins_B"
## [1] "../data/FinalFantasy/FFII"
## [1] "../data/FinalFantasy/FFIV_DS"
## [1] "../data/FinalFantasy/FFIX_B"
## [1] "../data/FinalFantasy/FFV"
## [1] "../data/FinalFantasy/FFVI"
## [1] "../data/FinalFantasy/FFVII"
## [1] "../data/FinalFantasy/FFVII_Remake"
## [1] "../data/FinalFantasy/FFVIII"
## [1] "../data/FinalFantasy/FFX_B"
## [1] "../data/FinalFantasy/FFX2"
## [1] "../data/FinalFantasy/FFXII_B"
## [1] "../data/FinalFantasy/FFXIII"
## [1] "../data/FinalFantasy/FFXIII-2"
## [1] "../data/FinalFantasy/FFXIII-LR"
```

```
## [1] "../data/FinalFantasy/FFXV"
## [1] "../data/Horizon/HorizonZeroDawn"
## [1] "../data/KingdomHearts/KingdomHearts_B"
## [1] "../data/KingdomHearts/KingdomHearts2"
## [1] "../data/KingdomHearts/KingdomHearts3"
## [1] "../data/KingdomHearts/KingdomHearts3D"
## [1] "../data/KingsQuest/KingsQuest1"
## [1] "../data/KingsQuest/KingsQuest2"
## [1] "../data/KingsQuest/KingsQuest3"
## [1] "../data/KingsQuest/KingsQuest4"
## [1] "../data/KingsQuest/KingsQuest5"
## [1] "../data/KingsQuest/KingsQuest6"
## [1] "../data/KingsQuest/KingsQuest7"
## [1] "../data/KingsQuest/KingsQuest8"
## [1] "../data/KingsQuest/KingsQuestChapters"
## [1] "../data/MassEffect/MassEffect1B"
## [1] "../data/MassEffect/MassEffect2"
## [1] "../data/MassEffect/MassEffect3C"
## [1] "../data/MonkeyIsland/MonkeyIsland2"
## [1] "../data/MonkeyIsland/TheCurseOfMonkeyIsland"
## [1] "../data/MonkeyIsland/TheSecretOfMonkeyIsland"
## [1] "../data/Persona/Persona3"
## [1] "../data/Persona/Persona4"
## [1] "../data/Persona/Persona5B"
## [1] "../data/StarWarsKOTOR/StarWarsKOTOR"
## [1] "../data/SuperMarioRPG/SuperMarioRPG"
```

Visualise total amount of female vs. male dialogue for major and minor characters:

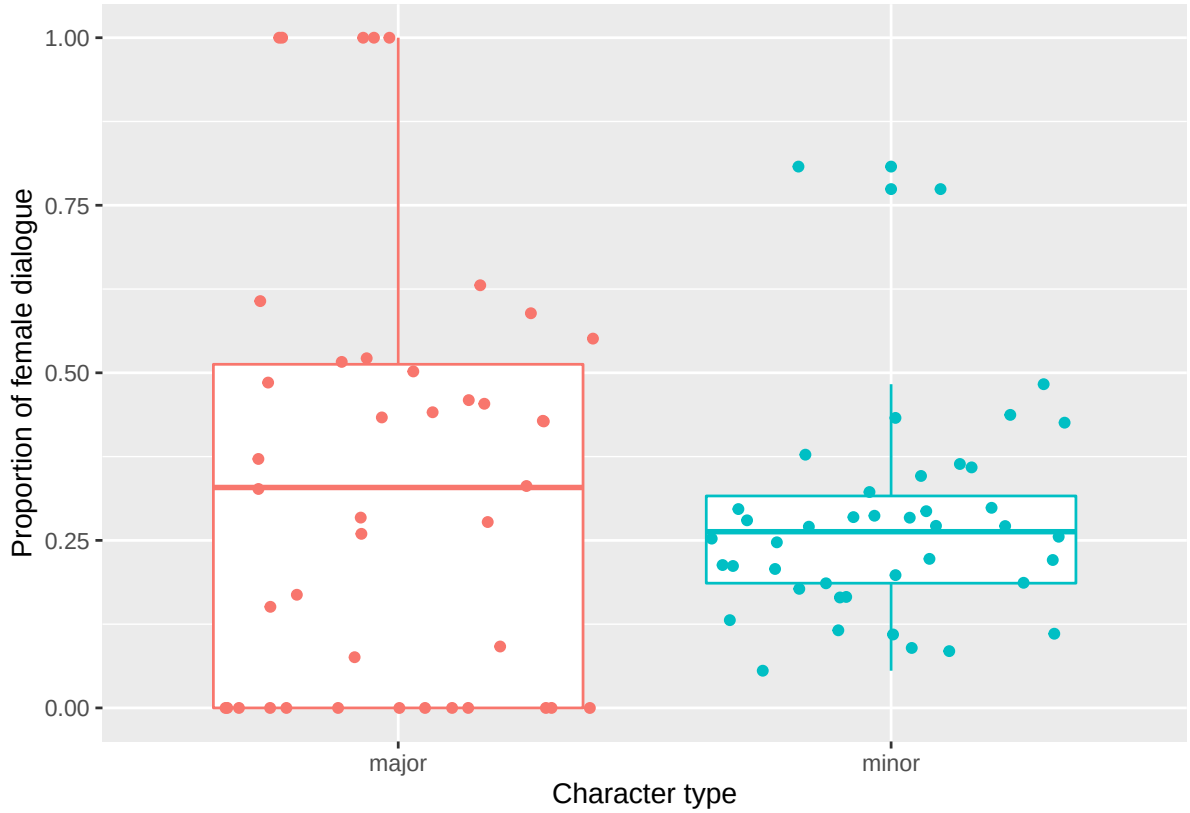
```
allGames.Maj.PercentFemale =
  100 * (
  sum(allGames[allGames$group=="major",]$numFemaleWords) /
   (sum(allGames[allGames$group=="major",]$numFemaleWords) +
      sum(allGames[allGames$group=="major",]$numMaleWords)))
allGames.Maj.PercentMale = 100 - allGames.Maj.PercentFemale

allGames.Min.PercentFemale =
  100 * (
  sum(allGames[allGames$group=="minor",]$numFemaleWords) /
   (sum(allGames[allGames$group=="minor",]$numFemaleWords) +
      sum(allGames[allGames$group=="minor",]$numMaleWords)))
allGames.Min.PercentMale = 100 - allGames.Min.PercentFemale

allGames$propFemaleCharacters=
  allGames$numFemaleCharacters /
    (allGames$numFemaleCharacters + allGames$numMaleCharacters)

dx = data.frame(
  Gender=factor(c("Male","Female","Male","Female"),
                levels=c("Male","Female")),
  Group = factor(c("Major","Major","Minor","Minor"),
                 levels=c("Minor","Major")),
  percentageWords=
    c(allGames.Maj.PercentMale,allGames.Maj.PercentFemale,
      allGames.Min.PercentMale,allGames.Min.PercentFemale))

ggplot(dx,aes(x=Group,y=percentageWords,fill=Gender))+ geom_bar(stat='identity')+
  geom_hline(yintercept=50,linetype="dotted") +
  coord_flip(ylim = c(0,100)) +
```

```
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.ticks.y = element_blank(),
        legend.position = "top") +
  scale_fill_discrete(breaks=c("Female","Male"),name="Gender")+
  ylab("% Words Spoken") +
  xlab("")
```



Below is a boxplot of the distribution of individual games. It's clear that the minor character group has a higher mean proportion of female dialogue, though the range for major characters is higher.

```
ggplot(allGames, aes(y=propFemaleDialogue,x=group,colour=group)) +
  geom_boxplot() +
  geom_jitter() +
  theme(legend.position = "none")+
  xlab("Character type") +
  ylab("Proportion of female dialogue")
```



Statistical test of average proportion of female dialogue in each game, comparing major and minor characters.

```
t.test(allGames$propFemaleDialogue~allGames$group)
```

```
##
##  Welch Two Sample t-test
##
## data:  allGames$propFemaleDialogue by allGames$group
## t = 1.2057, df = 58.644, p-value = 0.2328
## alternative hypothesis: true difference in means between group major and group minor is not equal
## 95 percent confidence interval:
##  -0.04416159  0.17801532
## sample estimates:
## mean in group major mean in group minor
##           0.3424845           0.2755577
```

The test is significant, suggesting that the female dialogue is lower in major characters than in minor characters.

However, we also know that the proportion of female dialogue is predicted by the proportion of female characters. The regression below tests whether the character groups (major or minor) predict the proportion of female dialogue over and above the proportion of female characters within the group:

```
mInt = lm(propFemaleDialogue~ group * propFemaleCharacters, data = allGames)
summary(mInt)
```

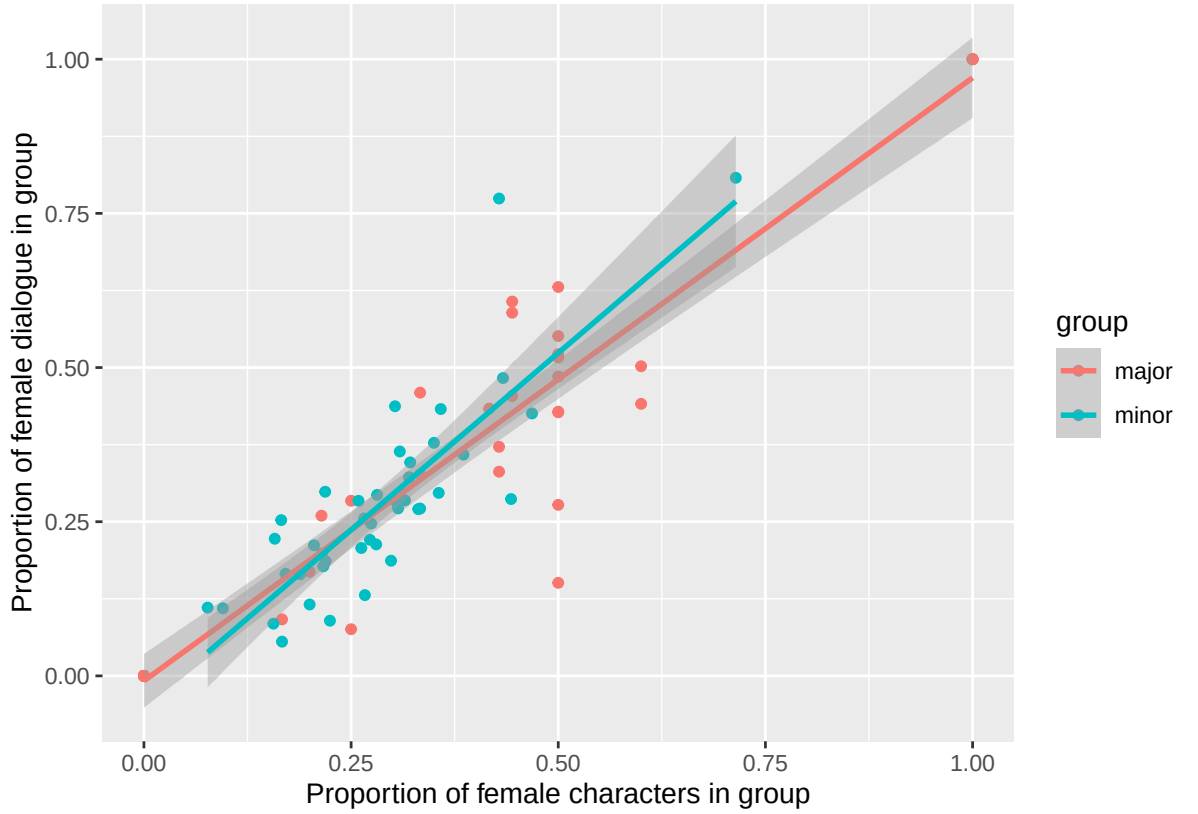
```
##
## Call:
## lm(formula = propFemaleDialogue ~ group * propFemaleCharacters,
##     data = allGames)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -0.33000 -0.04380  0.00805  0.03433  0.33245
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     -0.008052   0.020819  -0.387    0.700
## groupminor                      -0.042076   0.043018  -0.978    0.331
## propFemaleCharacters             0.977932   0.043627  22.416   <2e-16 ***
## groupminor:propFemaleCharacters  0.169571   0.130958   1.295    0.199
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08907 on 80 degrees of freedom
## Multiple R-squared:  0.8825, Adjusted R-squared:  0.8781
## F-statistic: 200.2 on 3 and 80 DF,  p-value: < 2.2e-16
```

The effect of the proportion of female characters is significant, but the effect of group is not, nor is the interaction.

```
ggplot(allGames,
       aes(x=propFemaleCharacters,
           y=propFemaleDialogue,
           colour=group)) +
  geom_point() +
  stat_smooth(method='lm') +
  xlab("Proportion of female characters in group") +
  ylab("Proportion of female dialogue in group")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



In summary, the gender bias against female dialogue is evident for both major and minor characters. This bias is exaggerated for major characters. However, this can be mostly explained by the number of female characters, rather than a systematic difference between major and minor character groups.

The full list of results:

```
kable(allGames[,c("game","group","propFemaleDialogue")])
```

game	group	propFemaleDialogue
Chrono Trigger	major	0.6306403
Chrono Trigger	minor	0.2526138
Dragon Age: Origins	major	0.4538605
Dragon Age: Origins	minor	0.2554227
Final Fantasy II	major	0.2839721
Final Fantasy II	minor	0.2986326
Final Fantasy IV	major	0.0916418
Final Fantasy IV	minor	0.1309963
Final Fantasy IX	major	0.3311388
Final Fantasy IX	minor	0.2073560
Final Fantasy V	major	0.4411320
Final Fantasy V	minor	0.0555346
Final Fantasy VI	major	0.2598039
Final Fantasy VI	minor	0.0847276
Final Fantasy VII	major	0.3266936
Final Fantasy VII	minor	0.1981802
Final Fantasy VII Remake	major	0.5020579
Final Fantasy VII Remake	minor	0.2838857
Final Fantasy VIII	major	0.4283841
Final Fantasy VIII	minor	0.1859402
Final Fantasy X	major	0.3715065
Final Fantasy X	minor	0.2967687
Final Fantasy X-2	major	1.0000000
Final Fantasy X-2	minor	0.2117596
Final Fantasy XII	major	0.2773916
Final Fantasy XII	minor	0.2716304
Final Fantasy XIII	major	0.4854317
Final Fantasy XIII	minor	0.2207313
Final Fantasy XIII-2	major	0.4593027
Final Fantasy XIII-2	minor	0.3778930
Lightning Returns: Final Fantasy XIII	major	1.0000000
Lightning Returns: Final Fantasy XIII	minor	0.3588774
Final Fantasy XV	major	0.0000000
Final Fantasy XV	minor	0.4326998
Horizon Zero Dawn	major	1.0000000
Horizon Zero Dawn	minor	0.2704224
Kingdom Hearts	major	0.0000000
Kingdom Hearts	minor	0.2846654
Kingdom Hearts II	major	0.0000000
Kingdom Hearts II	minor	0.2131808
Kingdom Hearts III	major	0.0756906
Kingdom Hearts III	minor	0.1777299
Kingdom Hearts 3D: Dream Drop Distance	major	0.0000000
Kingdom Hearts 3D: Dream Drop Distance	minor	0.0895143
King's Quest I: Quest for the Crown	major	0.0000000
King's Quest I: Quest for the Crown	minor	0.1158129
King's Quest II: Romancing the Throne	major	0.0000000
King's Quest II: Romancing the Throne	minor	0.8076923
King's Quest III: To Heir Is Human	major	0.0000000
King's Quest III: To Heir Is Human	minor	0.2713463
King's Quest IV: The Perils of Rosella	major	1.0000000
King's Quest IV: The Perils of Rosella	minor	0.7741100
King's Quest V	major	0.0000000

game	group	propFemaleDialogue
King's Quest V	minor	0.3461666
King's Quest VI	major	0.0000000
King's Quest VI	minor	0.2471298
King's Quest VII: The Princeless Bride	major	1.0000000
King's Quest VII: The Princeless Bride	minor	0.1867603
King's Quest VIII	major	0.0000000
King's Quest VIII	minor	0.2799134
King's Quest Chapters	major	0.1509165
King's Quest Chapters	minor	0.4830259
Mass Effect	major	0.5510669
Mass Effect	minor	0.2935248
Mass Effect 2	major	0.4334063
Mass Effect 2	minor	0.4372264
Mass Effect 3	major	0.5162871
Mass Effect 3	minor	0.3221426
Monkey Island 2: LeChuck's Revenge	major	0.0000000
Monkey Island 2: LeChuck's Revenge	minor	0.2224024
The Curse of Monkey Island	major	0.0000000
The Curse of Monkey Island	minor	0.1095802
The Secret of Monkey Island	major	0.0000000
The Secret of Monkey Island	minor	0.1107168
Persona 3	major	0.5889274
Persona 3	minor	0.2867931
Persona 4	major	0.5217268
Persona 4	minor	0.4256015
Persona 5	major	0.4274823
Persona 5	minor	0.3639303
Star Wars: Knights of the Old Republic	major	0.6070153
Star Wars: Knights of the Old Republic	minor	0.1655871
Super Mario RPG: Legend of the Seven Stars	major	0.1688733
Super Mario RPG: Legend of the Seven Stars	minor	0.1647965

Bias in different quantiles of character dialgoue

Does the gender bias differ for characters that speak a lot compared to characters that don't?

For each game, we divide characters into four groups based on the amount of dialogue they speak. Characters are ranked by the proportion of dialogue that they speak within the game. Then the characters are split into four groups of even number (the 'quantiles' of the dialogue proportions). Across all games, the total number of words is calculated for each gender for each quantile.

```
folders = list.dirs("../data", recursive = T)
folders = folders[sapply(folders,function(X){
    "stats_by_character.csv" %in% list.files(X)
  })]


allChars = NULL
for(folder in folders){
  shortName = tail(strsplit(folder,"/")[[1]],1)
  js = fromJSON(file = paste0(folder,"/meta.json"))
  alternativeMeasure = FALSE
  if(!is.null(js$alternativeMeasure)){
    alternativeMeasure = js$alternativeMeasure
  }
  if(!alternativeMeasure){
    statsByChar = read.csv(paste0(folder,"/stats_by_character.csv"),stringsAsFactors = F)
    statsByChar = statsByChar[!is.na(statsByChar$words),]
    statsByChar = statsByChar[statsByChar$words>0,]
    if(nrow(statsByChar)>0){
        statsByChar = statsByChar[statsByChar$group %in% c("male","female"),]
        statsByChar$dialogProp = statsByChar$words/sum(statsByChar$words)
        allChars = rbind(allChars,statsByChar)
      }
    }
}
```

```
numQuantiles = 4
allChars$Quantile = cut(allChars$dialogProp,
      breaks= quantile(allChars$dialogProp,
          probs = seq(0,1,length.out=numQuantiles+1)))
q = data.frame(
  Quantile = 1:numQuantiles,
  femaleWords = tapply(
    allChars[allChars$group=="female",]$words,
    allChars[allChars$group=="female",]$Quantile,sum),
  maleWords = tapply(
    allChars[allChars$group=="male",]$words,
    allChars[allChars$group=="male",]$Quantile,sum),
  femaleChars = tapply(
    allChars[allChars$group=="female",]$words,
    allChars[allChars$group=="female",]$Quantile,length),
  maleChars = tapply(
    allChars[allChars$group=="male",]$words,
    allChars[allChars$group=="male",]$Quantile,length)
)
q$propFemale = q$femaleWords/ (q$femaleWords+q$maleWords)
q$propMale = q$maleWords/ (q$femaleWords+q$maleWords)
q$propFemaleChar = q$femaleChar/ (q$femaleChar+q$maleChar)
q$propMaleChar = q$maleChar/ (q$femaleChar+q$maleChar)
```

There are small but significant differences in the proportion of words in each of the four quantiles:

```
q[,c("femaleWords","maleWords","propFemale","propMale")]
```

```
##                     femaleWords maleWords propFemale  propMale
## (1.29e-06,0.000151]       27980     60779  0.3152356 0.6847644
## (0.000151,0.00051]        79830    190099  0.2957444 0.7042556
## (0.00051,0.00188]        221795    520444  0.2988188 0.7011812
## (0.00188,0.62]          1667455   2910938  0.3642009 0.6357991
```

```
chisq = chisq.test(q[,c("femaleWords","maleWords")])
chisq
```

```
##
##  Pearson's Chi-squared test
##
## data:  q[, c("femaleWords", "maleWords")]
## X-squared = 16467, df = 3, p-value < 2.2e-16
```

```
pv = chisq$p.value
if(pv < 0.0001){
  pv = paste0("p = 0.0001")
  } else{
  pv = paste0("p = ",round(pv,3))
}
chisqOut = paste0("$\\chi^2$ = ", round(chisq$statistic,2),", ",pv)
cat(chisqOut,file="../results/latexStats/quantileWordsChiSq.tex")
# compare just first and last quantile
chisq.test(q[c(1,4),c("femaleWords","maleWords")])
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  q[c(1, 4), c("femaleWords", "maleWords")]
## X-squared = 902.33, df = 1, p-value < 2.2e-16
```

However, the proportion of characters is not significantly different (the p-value is marginal at the 0.05 level, but very different from the result above).

```
q[,c("femaleChars","maleChars","propFemaleChar","propMaleChar")]
```

```
##                     femaleChars maleChars propFemaleChar propMaleChar
## (1.29e-06,0.000151]         871      2176      0.2858549    0.7141451
## (0.000151,0.00051]          902      2141      0.2964180    0.7035820
## (0.00051,0.00188]           926      2118      0.3042050    0.6957950
## (0.00188,0.62]              874      2171      0.2870279    0.7129721
```

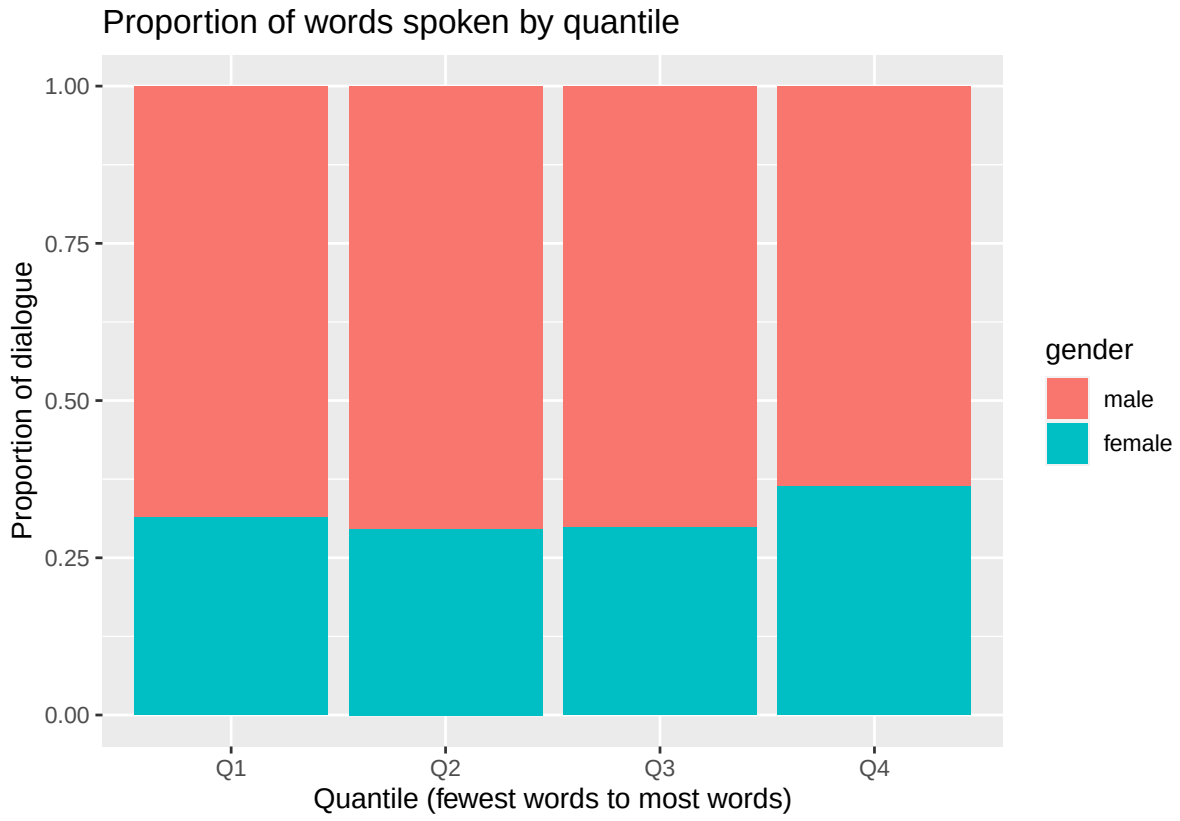
```
chisq = chisq.test(q[,c("femaleChars","maleChars")])
pv = chisq$p.value
if(pv < 0.0001){
  pv = paste0("p = 0.0001")
  } else{
  pv = paste0("p = ",round(pv,3))
}
chisqOut = paste0("$\\chi^2$ = ", round(chisq$statistic,2),", ",pv)
cat(chisqOut,file="../results/latexStats/quantileCharChiSq.tex")
# compare just first and last quantile
chisq.test(q[c(1,4),c("femaleChars","maleChars")])
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  q[c(1, 4), c("femaleChars", "maleChars")]
```

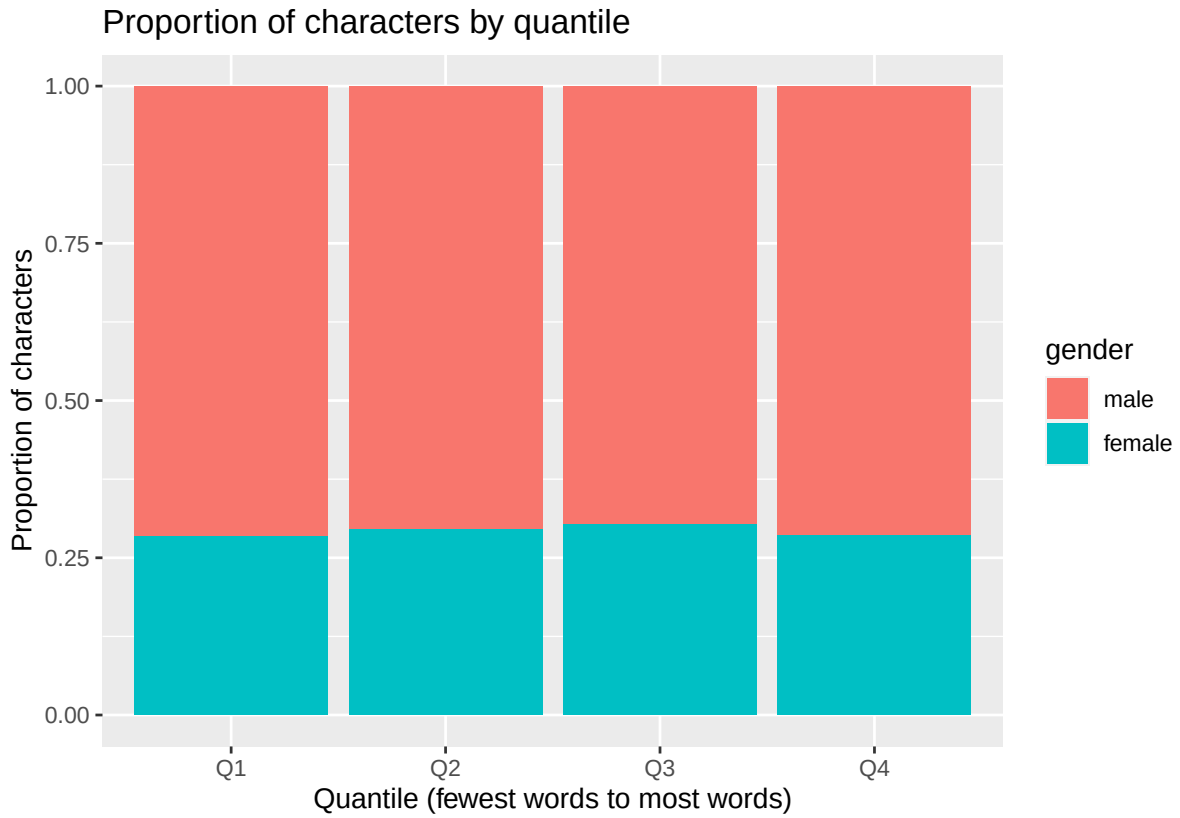
```
## X-squared = 0.0053164, df = 1, p-value = 0.9419
```

Plot results:

```
q2 = data.frame(
  Quantile = paste0("Q",rep(1:numQuantiles,2)),
  gender = factor(rep(c("female","male"),each=numQuantiles),
                  levels=c("male","female")),
  prop = c(q$propFemale,q$propMale),
  propChar =c(q$propFemaleChar,q$propMaleChar)
)

ggplot(q2,aes(x=Quantile,fill=gender,y=prop)) +
  geom_bar(stat = "identity",position="stack") +
  ylab("Proportion of dialogue") +
  xlab("Quantile (fewest words to most words)") +
  ggtitle("Proportion of words spoken by quantile")
```




```
ggplot(q2,aes(x=Quantile,fill=gender,y=propChar)) +
  geom_bar(stat = "identity",position="stack") +
  ylab("Proportion of characters") +
  xlab("Quantile (fewest words to most words)")+
  ggtitle("Proportion of characters by quantile")
```



In summary, there are small differences in the proportion of female dialogue for talkative and less talkative characters. But the overall

Other estimates of bias

One concern is that the coding for main characters may be more complete or more accurate than coding for minor characters, because minor characters are harder to find in videos, have less documentation on wikis and less direct linguistic cues.

Method 1: Complete vs. incomplete scripts

Each game is coded for whether it's complete, partial or a sample of the dialogue in the full game. If the estimate was biased by the completeness of the coding, then we might expect to see a difference in the estimated proportion of female dialogue between these types.

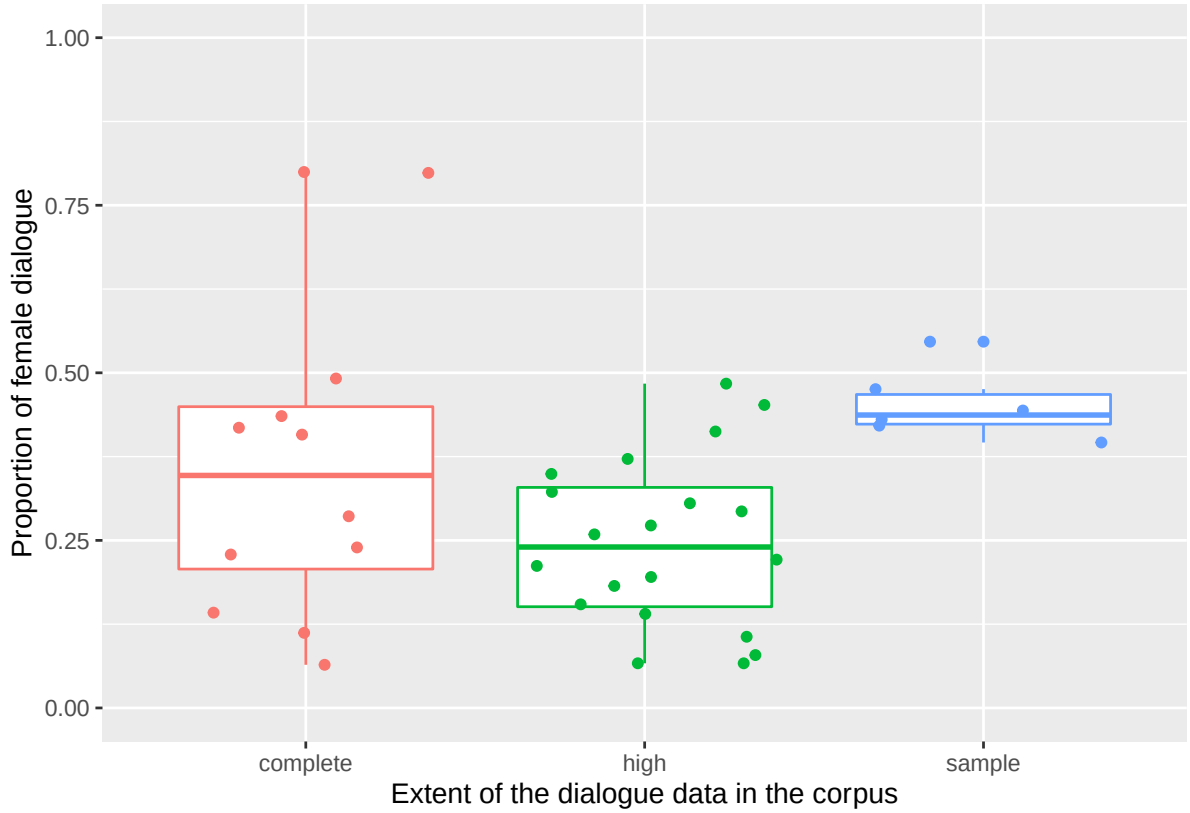
Below is a boxplot showing how the proportion of female dialogue differs between the game types.

First, we load the data:

```
allGames.completeness = NULL
for(folder in folders){
  js = fromJSON(file = paste0(folder,"/meta.json"))
  completeness = NA
  if(!is.null(js$sourceFeatures)){
    if(!is.null(js$sourceFeatures$completeness)){
      completeness = js$sourceFeatures$completeness
    }
  }
  alternativeMeasure = FALSE
  if(!is.null(js$alternativeMeasure)){
    alternativeMeasure = js$alternativeMeasure
  }
  if(!alternativeMeasure){
    stats = read.csv(paste0(folder,"/stats_by_character.csv"),stringsAsFactors = F)
    propFemaleDialogue = sum(stats[stats$group=="female",]$words) /
      sum(stats[stats$group %in% c("male","female"),]$words)
    propFemaleCharacters = sum(stats$group=="female")/
      sum(stats$group %in% c("male","female"))
    ret = data.frame(
      folder = folder,
      game = js$game,
      completeness = completeness,
      propFemaleDialogue = propFemaleDialogue,
      propFemaleCharacters = propFemaleCharacters,
      year = js$year
    )
    allGames.completeness = rbind(allGames.completeness, ret)
  }
}
allGames.completeness = allGames.completeness[
  is.finite(allGames.completeness$propFemaleDialogue),]
```

Below we plot the proportion of female dialogue according to the extent of the data coded:

```
ggplot(allGames.completeness[!is.na(allGames.completeness$completeness),],
       aes(x=completeness, y =propFemaleDialogue, colour=completeness)) +
  geom_boxplot() +
  geom_jitter() +
  theme(legend.position = "none")+
  xlab("Extent of the dialogue data in the corpus") +
  ylab("Proportion of female dialogue") +
  coord_cartesian(ylim=c(0,1))
```



A t-test comparing proportion of female dialogue in 'complete' and 'high' sources:

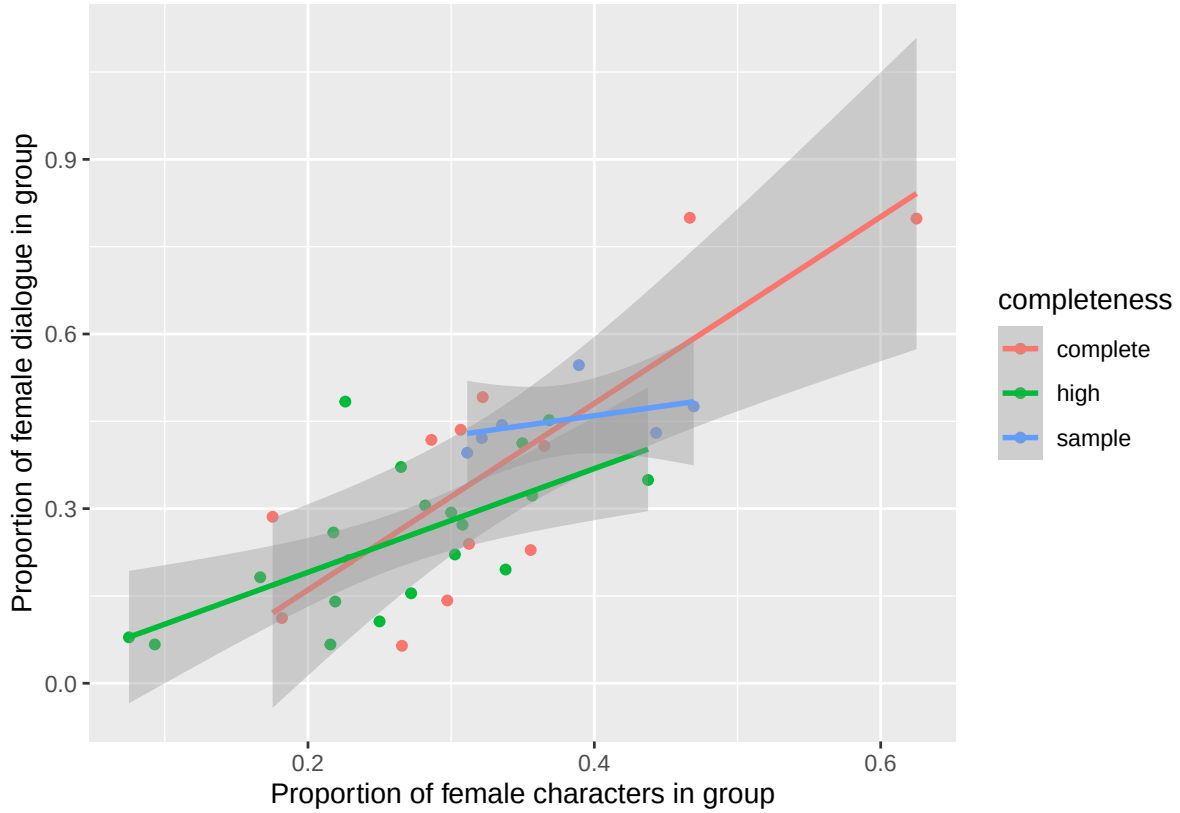
```
t.test(propFemaleDialogue~completeness,
       data = allGames.completeness[
         allGames.completeness$completeness %in%
           c("complete", "high"),])
```

```
##
##  Welch Two Sample t-test
##
## data:  propFemaleDialogue by completeness
## t = 1.6076, df = 14.676, p-value = 0.1292
## alternative hypothesis: true difference in means between group complete and group high is not equ
## 95 percent confidence interval:
##  -0.03986572  0.28264781
## sample estimates:
## mean in group complete     mean in group high
##              0.3686546              0.2472635
```

The proportion of female dialogue is slightly higher for more complete data. This might suggest that the estimate of female dialogue is underestimated in the corpus. However, we also know that the proportion of female dialogue is predicted by the proportion of female characters:

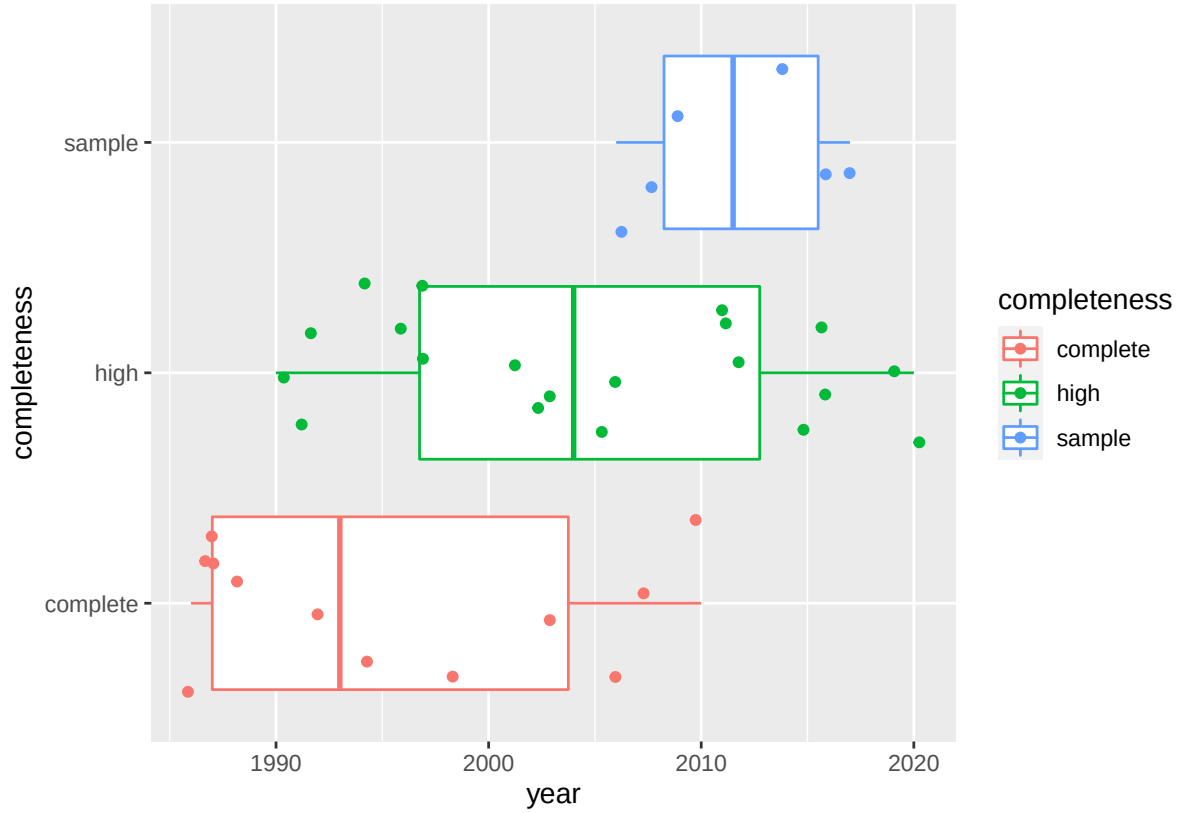
```
ggplot(allGames.completeness[!is.na(allGames.completeness$completeness),],
       aes(x=propFemaleCharacters,
           y =propFemaleDialogue,
           colour=completeness)) +
  geom_point() +
  stat_smooth(method='lm') +
  xlab("Proportion of female characters in group") +
  ylab("Proportion of female dialogue in group")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



It could also be the case that the year of publication is related to how complete the script is (since earlier games have less content, and are more likely to have accessible data). This appears to be the case:

```
ggplot(allGames.completeness[!is.na(allGames.completeness$completeness),],
       aes(x=year, y=completeness,colour=completeness)) +
  geom_boxplot() +
  geom_jitter()
```



This makes it seem like the the relationship with completeness may be partially driven by the proportion of female characters and/or by the date of release. Here is a regression, predicting the proportion of female dialogue by completeness and by the proportion of female characters:

```
mComp = lm(propFemaleDialogue ~ completeness + propFemaleCharacters + year,
   data = allGames.completeness)
summary(mComp)
```

```
##
## Call:
## lm(formula = propFemaleDialogue ~ completeness + propFemaleCharacters +
##     year, data = allGames.completeness)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.22375 -0.08847  0.01843  0.07518  0.28310
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -2.574439   4.555489  -0.565    0.576
## completenesshigh      -0.054719   0.051601  -1.060    0.297
## completenesssample     0.005606   0.070947   0.079    0.937
## propFemaleCharacters   1.182882   0.210203   5.627 2.89e-06 ***
## year                   0.001279   0.002287   0.559    0.580
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1207 on 33 degrees of freedom
## Multiple R-squared:  0.6009, Adjusted R-squared:  0.5525
## F-statistic: 12.42 on 4 and 33 DF,  p-value: 2.858e-06
```

It seems there is no effect of completeness once the proportion of female characters is taken into account.

Method 2: Cumulative gender balance

We can try to estimate how our measuring of gender balance is affected by the coding of minor characters. This can be done by looking at how the estimate varies as we observe more and more minor characters. That is, what would the estimate be like if we had only coded the top 10% of most prolific characters (those that speak most), or the top 20%, or top 30% etc. At some point, the estimate of gender balance would converge on the final estimate for all coded characters.

Example

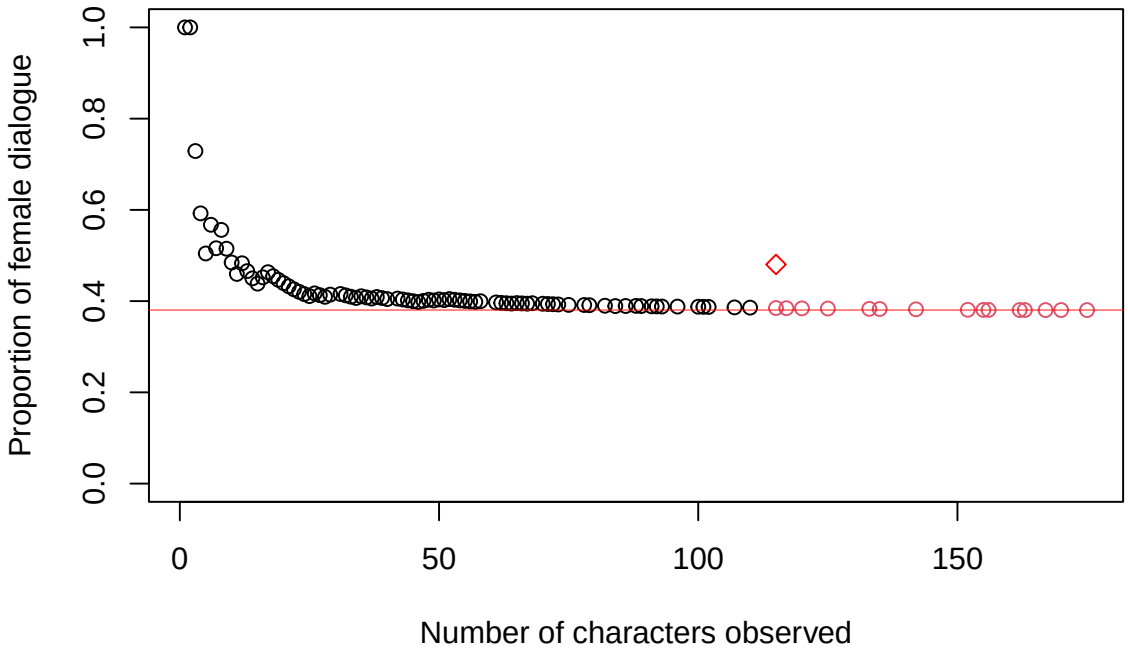
Here's a function that works out the gender balance in dialogue over the range of characters. It ranks all characters from most words to least words, then works out the gender balance taking into account just the top character, the top two characters, the top three characters etc.

```
getBalanceOverCumulativeCharacterRange = function(statsByChar){
  # Remove other groups except male and female
  statsByChar = statsByChar[statsByChar$group %in% c("male","female"),]
  statsByChar = statsByChar[!is.na(statsByChar$words),]
  statsByChar = statsByChar[statsByChar$words>0,]

  # Sorted list of number of words for each character (most to least)
  sortedUniqueNumOfWords = sort(unique(statsByChar$words),decreasing = TRUE)

  # Table of total number of words observed for each gender
  #  as the number of characters observed increases from
  #  character with most dialogue to least dialogue
  wordsByGender.Cumulative =
    sapply(sortedUniqueNumOfWords,
      function(minNumWords){
        x = statsByChar[statsByChar$words>=minNumWords,]
        femaleWords = sum(x[x$group=="female",]$words)
        maleWords = sum(x[x$group=="male",]$words)
        femaleProp = femaleWords / (femaleWords+maleWords)
        return(c(femaleWords,maleWords))
      })
  # Convert to proportion of female dialogue
  femalePropCumulative = wordsByGender.Cumulative[1,] /
    colSums(wordsByGender.Cumulative)
  totalEstimate = femalePropCumulative[length(femalePropCumulative)]
  # Binomial test at each point: is it significantly different
  #  from the total?
  sigDifferentFromTotal.p = apply(wordsByGender.Cumulative,2,
    function(wbg){
      x = binom.test(wbg,p = totalEstimate)$p.value
  })

  sigDifferentFromTotal = sigDifferentFromTotal.p>0.05
  firstNonSignif = which(sigDifferentFromTotal)[1]
  # stable non-signif (point after last significant result)
  firstStableNonSignif = length(sigDifferentFromTotal) - which(!rev(sigDifferentFromTotal))[1] + 2

  numCharCumulative = sapply(sortedUniqueNumOfWords,
      function(minNumWords){
        sum(statsByChar$words>=minNumWords)
      })

  numCharBeforeEstimateFirstNotSigDiff = numCharCumulative[firstNonSignif]
  numCharBeforeEstimateStabilises = numCharCumulative[firstStableNonSignif]
```

```
  return(list(femalePropCumulative = femalePropCumulative,
      numCharCumulative = numCharCumulative,
      p = sigDifferentFromTotal.p,
      numCharBeforeEstimateFirstNotSigDiff = numCharBeforeEstimateFirstNotSigDiff,
      numCharBeforeEstimateStabilises = numCharBeforeEstimateStabilises
              ))
}

# Function for visualising the information
plotCumulativeCharRange = function(stats){
  plot(stats$femalePropCumulative~
       stats$numCharCumulative,
     xlab = "Number of characters observed",
     ylab = "Proportion of female dialogue",
     ylim=c(0,1),
     col = 1 + (stats$p>0.05))
abline(h=tail(stats$femalePropCumulative,n=1),
       col=rgb(1,0,0,0.5))
points(stats$numCharBeforeEstimateStabilises,
       tail(stats$femalePropCumulative,n=1) + 0.1,
       pch=5, col="red")
}
```

We can apply this to Chrono Trigger:

```
chrono =  read.csv("../data/ChronoTrigger/ChronoTrigger/stats_by_character.csv",
                   stringsAsFactors = F)
chrono = chrono[!is.na(chrono$words),]
chrono = chrono[chrono$words>0,]
chrono.stats = getBalanceOverCumulativeCharacterRange(chrono)
plotCumulativeCharRange(chrono.stats)
```

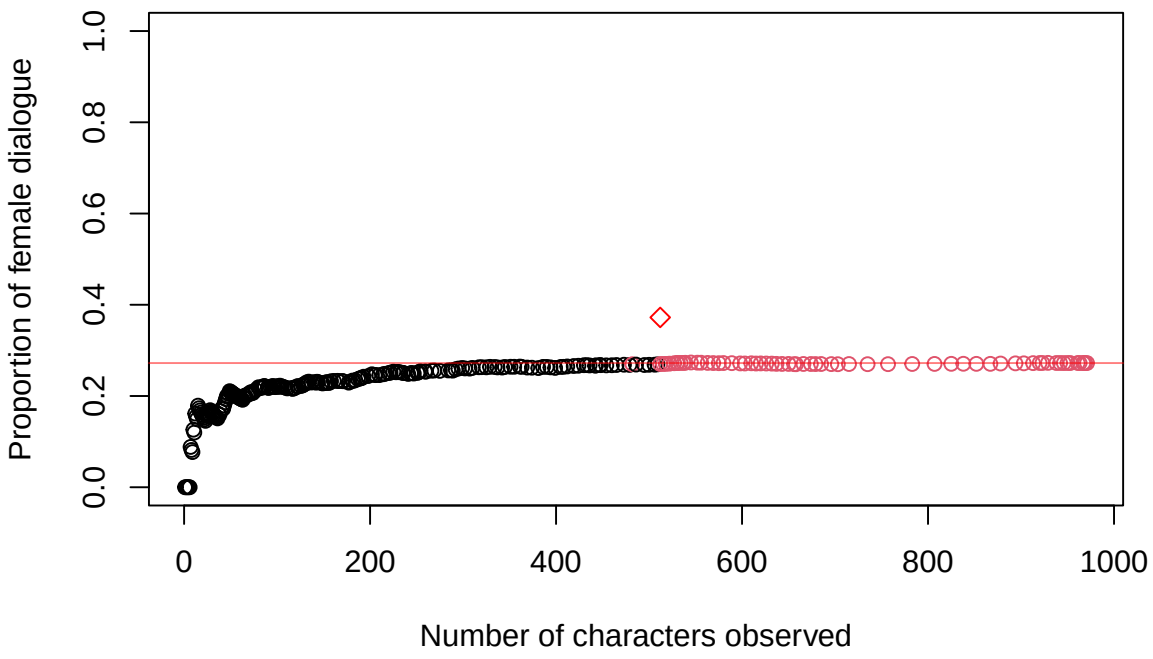


The points in the plot above shows the gender balance when taking into account various numbers of characters. Note that there is not an estimate for every possible number of characters, since several characters are tied in the number of words they speak. The red line shows the gender balance in the full game. The points are coloured red if they are not significantly different from the gender balance in the full game.

It's clear that the proportion of female dialogue is higher for characters with more dialogue. The first time that the estimate is not significantly different from the total estimate is after seeing 115 characters, or after seeing 66% of the characters. After this, the estimate does not change significantly. This is indicated with a red diamond on the plot. This also happens to be the same as the point at which the estimate stabilises (the point after which the estimate is never again significantly different from the total estimate).

Things are a little different for Final Fantasy XII, which has many more characters:

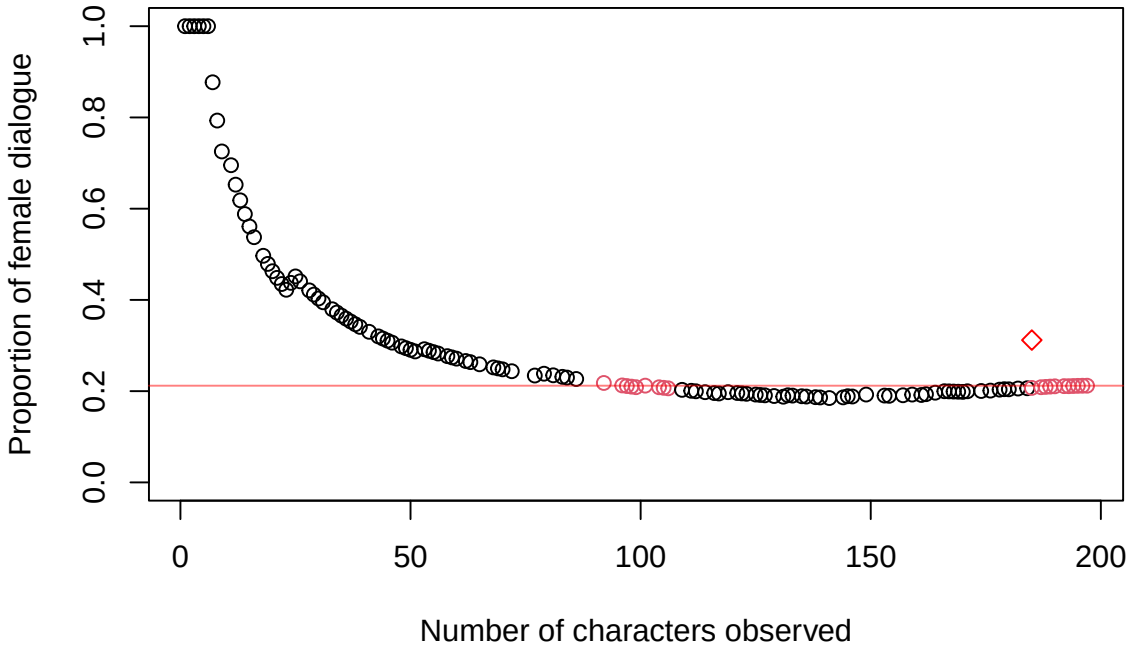
```
FFXII =  read.csv("../data/FinalFantasy/FFXII_B/stats_by_character.csv",
                  stringsAsFactors = F)
FFXII.stats = getBalanceOverCumulativeCharacterRange(FFXII)
plotCumulativeCharRange(FFXII.stats)
```



Here, it's clear that the proportion of female dialogue is *lower* for characters with more dialogue. The first time that the estimate is not significantly different from the total estimate is after seeing 482 characters, or after seeing 50% of the characters.

Things are different again for Daggerfall:

```
Daggerfall =  read.csv("../data/ElderScrolls/Daggerfall/stats_by_character.csv",
                  stringsAsFactors = F)
Daggerfall.stats = getBalanceOverCumulativeCharacterRange(Daggerfall)
plotCumulativeCharRange(Daggerfall.stats)
```



Here, the female dialogue is over-estimated for characters with lots of dialogue, but then under-estimated for mid-range characters. It is only after seeing 47% of characters that the estimate stabilises.

Cumulative gender balance over all games

We can now estimate the statistics above for all games:

```
folders = list.dirs("../data", recursive = T)
folders = folders[sapply(folders,function(X){
  "stats_by_character.csv" %in% list.files(X)
})]


allGames.Cum = NULL
for(folder in folders){
  shortName = tail(strsplit(folder,"/")[[1]],1)
  js = fromJSON(file = paste0(folder,"/meta.json"))
  alternativeMeasure = FALSE
  if(!is.null(js$alternativeMeasure)){
    alternativeMeasure = js$alternativeMeasure
  }
  if(!alternativeMeasure){
    print(folder)
    statsByChar = read.csv(paste0(folder,"/stats_by_character.csv"),stringsAsFactors = F)
    statsByChar = statsByChar[!is.na(statsByChar$words),]
    statsByChar = statsByChar[statsByChar$words>0,]

    if(nrow(statsByChar)>0){
      gameStats = getBalanceOverCumulativeCharacterRange(statsByChar)
      totalNumChar = tail(gameStats$numCharCumulative,n=1)
      percentCharBeforeEstimateStabilises=
        100 * (gameStats$numCharBeforeEstimateStabilises / totalNumChar)
      ret = data.frame(
        folder = folder,
        game = js$game,
        shortName = shortName,
        totalNumChar = totalNumChar,
        femalePercentCumulative = 100*gameStats$femalePropCumulative,
        percentCharCumulative = 100*(gameStats$numCharCumulative/totalNumChar),
        percentCharBeforeEstimateStabilises = percentCharBeforeEstimateStabilises
      )
      ret$femalePercentTotal = tail(ret$femalePercentCumulative,n=1)
      allGames.Cum = rbind(allGames.Cum,ret)
    }
  }
}
```

```
## [1] "../data/ChronoTrigger/ChronoTrigger"
## [1] "../data/DragonAge/DragonAge2"
## [1] "../data/DragonAge/DragonAgeOrigins_B"
## [1] "../data/ElderScrolls/Daggerfall"
## [1] "../data/ElderScrolls/Morrowind"
## [1] "../data/ElderScrolls/Oblivion"
## [1] "../data/ElderScrolls/Skyrim"
## [1] "../data/FinalFantasy/FFI"
## [1] "../data/FinalFantasy/FFII"
## [1] "../data/FinalFantasy/FFIV_DS"
## [1] "../data/FinalFantasy/FFIX_B"
## [1] "../data/FinalFantasy/FFV"
## [1] "../data/FinalFantasy/FFVI"
## [1] "../data/FinalFantasy/FFVII"
## [1] "../data/FinalFantasy/FFVII_Remake"
```

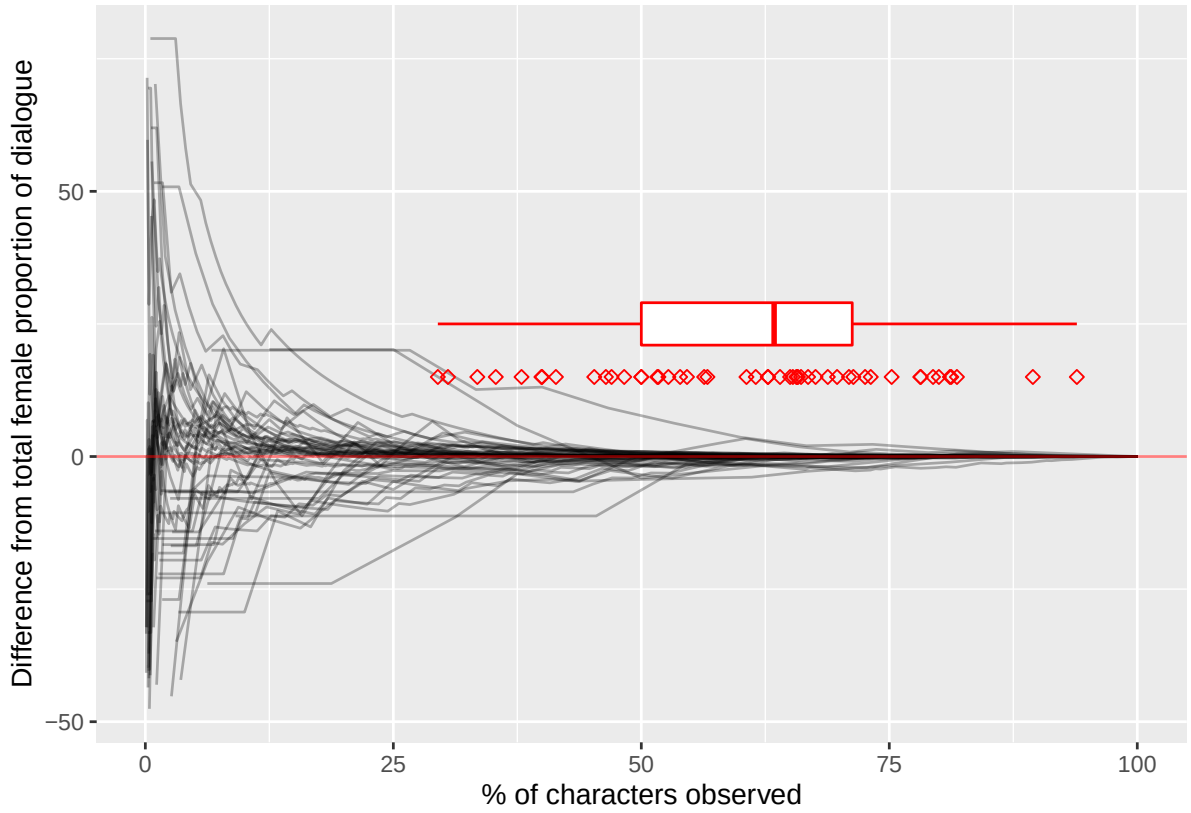
```
## [1] "../data/FinalFantasy/FFVIII"
## [1] "../data/FinalFantasy/FFX_B"
## [1] "../data/FinalFantasy/FFX2"
## [1] "../data/FinalFantasy/FFXII_B"
## [1] "../data/FinalFantasy/FFXIII"
## [1] "../data/FinalFantasy/FFXIII-2"
## [1] "../data/FinalFantasy/FFXIII-LR"
## [1] "../data/FinalFantasy/FFXIV"
## [1] "../data/FinalFantasy/FFXV"
## [1] "../data/Horizon/HorizonZeroDawn"
## [1] "../data/KingdomHearts/KingdomHearts_B"
## [1] "../data/KingdomHearts/KingdomHearts2"
## [1] "../data/KingdomHearts/KingdomHearts3"
## [1] "../data/KingdomHearts/KingdomHearts3D"
## [1] "../data/KingsQuest/KingsQuest1"
## [1] "../data/KingsQuest/KingsQuest2"
## [1] "../data/KingsQuest/KingsQuest3"
## [1] "../data/KingsQuest/KingsQuest4"
## [1] "../data/KingsQuest/KingsQuest5"
## [1] "../data/KingsQuest/KingsQuest6"
## [1] "../data/KingsQuest/KingsQuest7"
## [1] "../data/KingsQuest/KingsQuest8"
## [1] "../data/KingsQuest/KingsQuestChapters"
## [1] "../data/MassEffect/MassEffect1B"
## [1] "../data/MassEffect/MassEffect2"
## [1] "../data/MassEffect/MassEffect3C"
## [1] "../data/MonkeyIsland/MonkeyIsland2"
## [1] "../data/MonkeyIsland/TheCurseOfMonkeyIsland"
## [1] "../data/MonkeyIsland/TheSecretOfMonkeyIsland"
## [1] "../data/Persona/Persona3"
## [1] "../data/Persona/Persona4"
## [1] "../data/Persona/Persona5B"
## [1] "../data/StardewValley/StardewValley"
## [1] "../data/StarWarsKOTOR/StarWarsKOTOR"
## [1] "../data/SuperMarioRPG/SuperMarioRPG"
```

```
allGames.Cum$diffFromTotalEstimate =
  allGames.Cum$femalePercentCumulative - allGames.Cum$femalePercentTotal
```

The graph below visualises the data for all games. The vertical axis plots the distance from the final total estimate for the specific game, so that all games converge on zero (no difference from the total estimate). The horizontal axis shows the percentage of characters seen, so that each game is normalised for the number of characters. Red diamonds show the points at which the estimate for each game stabilises. The boxplot shows the distribution of the stabilisation points over the horizontal axis.

```
ggplot(allGames.Cum,
       aes(x=percentCharCumulative,
           y=diffFromTotalEstimate,
           group=folder)) +
  geom_line(alpha=0.3) +
geom_hline(yintercept = 0, alpha=0.5,colour='red') +
  geom_point(data=allGames.Cum[!duplicated(allGames.Cum$folder),],
    aes(x=percentCharBeforeEstimateStabilises,
        y=15),
    colour="red",shape =5) +
  geom_boxploth(data=allGames.Cum[!duplicated(allGames.Cum$folder),],
                aes(x=percentCharBeforeEstimateStabilises,
                    y=25,group=NULL), colour="red", width=8) +
  ylab("Difference from total female proportion of dialogue") +
```

```
  xlab("% of characters observed")
```



Estimates for all games stabilised before seeing 95% of their characters, suggesting that the estimates for the specific games in the corpus would not change much with the addition of data from more minor characters.

The mean stabilisation point is after seeing 60.7437992% of characters (95% quantile = [31.17468 ,87.7511962].

Is the estimate biased?

The graphs above show that, when considering only the most prolific characters, there are games which both over- and under- estimate the proportion of female dialogue. Additionally, we want to formally test whether the estimate of the estimate of the proportion of female dialogue is biased *in the corpus as a whole*. We can do this by looking at the data after seeing 10% of the characters in a game. The t-test below tests whether there is a bias one way or the other:

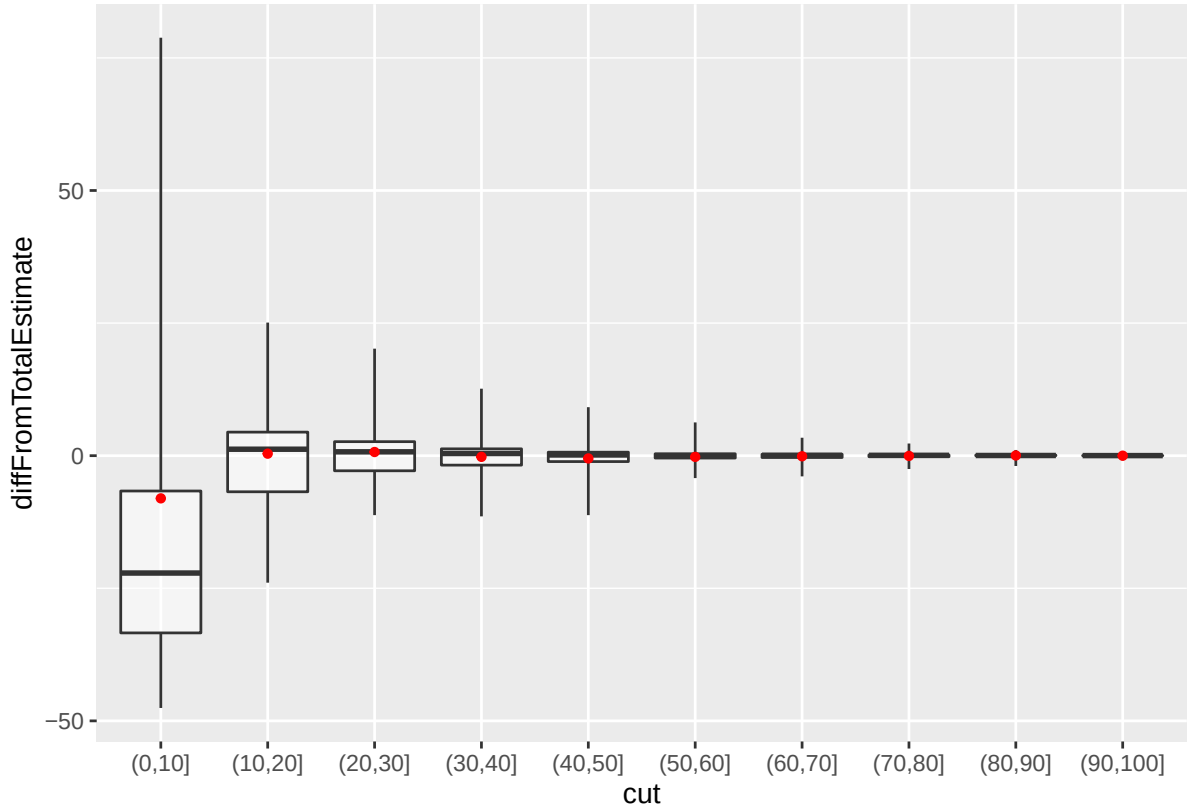
```
firstTenPercent = allGames.Cum[allGames.Cum$percentCharCumulative>=10.00,]
firstTenPercent = firstTenPercent[!duplicated(firstTenPercent$folder),]
tBias = t.test(firstTenPercent$diffFromTotalEstimate)
tBias
```

```
##
##  One Sample t-test
##
## data:  firstTenPercent$diffFromTotalEstimate
## t = -0.008225, df = 49, p-value = 0.9935
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -2.967659  2.943466
## sample estimates:
##  mean of x
## -0.0120968
```

The mean difference from the final estimate is -0.0120968, which indicates that the estimate based on prolific characters may be under-estimating the proportion of female speech on average. However, this is not significantly different from 0.

A similar point is made when drawing boxplots for the mean estimate of female dialogue % for each 10% of the range of characters seen (the whiskers show the full range of the data). Seeing only 10% of the characters biases the estimates (the proportion of female dialogue is under-estimated, though this isn't significant as shown by the t-test above). After seeing 10% of the characters, the estimates are very close to the final estimates after seeing all the characters.

```
allGames.Cum$cut = cut(allGames.Cum$percentCharCumulative,
                       breaks = seq(0,100,by=10))
ggplot(allGames.Cum[!duplicated(paste(allGames.Cum$folder, allGames.Cum$cut)),],
       aes(y=diffFromTotalEstimate,
           x = cut)) +
  geom_boxplot(alpha=0.5, coef = Inf) +
  stat_summary(fun=mean, geom="point", shape=20, size=2, color="red", fill="red")
```



However, cutting the data in 10% chunks is arbitrary, and the data within each chunk are not necessarily normally distributed.

A more generalised answer can be give by fitting a general additive model (GAM), predicting the difference from the total estimate based on the percentage of characters seen, with random effects to capture the dependency of datapoints that belong to the same game.

```
mBeta = betareg(I(0.3 + (diffFromTotalEstimate/100)) ~
                percentCharCumulative | percentCharCumulative,
                data = allGames.Cum[allGames.Cum$percentCharCumulative>5,])

summary(mBeta)
```

```
##
## Call:
## betareg(formula = I(0.3 + (diffFromTotalEstimate/100)) ~ percentCharCumulative |
```

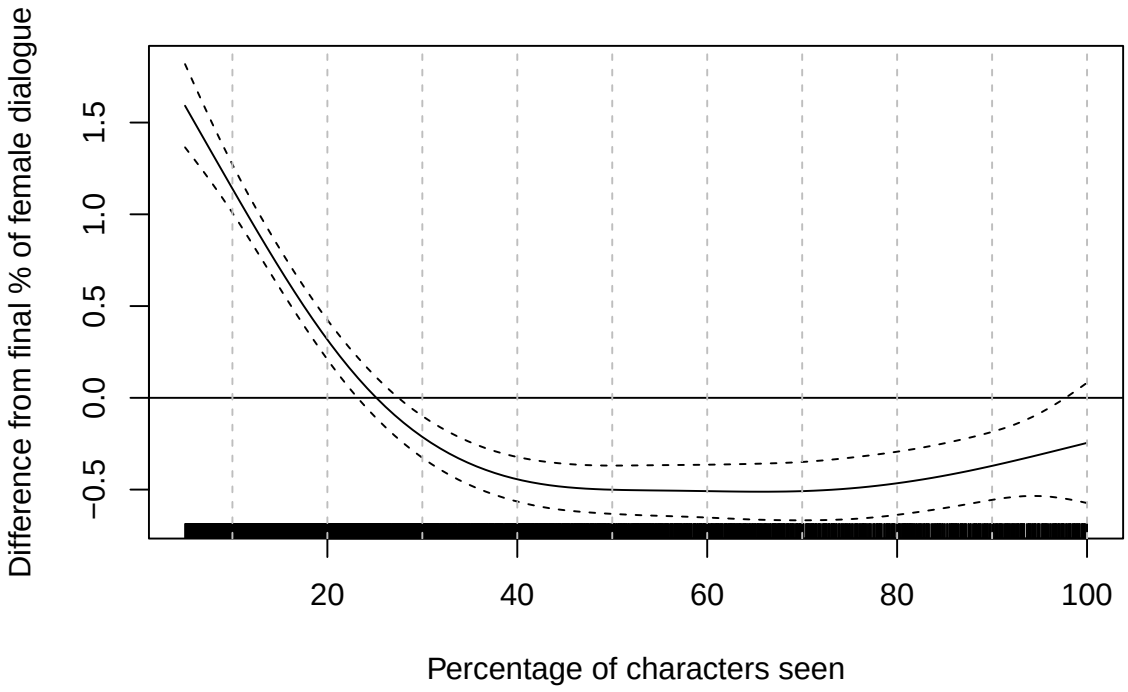
```
##     percentCharCumulative, data = allGames.Cum[allGames.Cum$percentCharCumulative >
##     5, ])
##
## Standardized weighted residuals 2:
##      Min       1Q   Median       3Q      Max
## -14.2673  -0.0195   0.1715   0.3686   7.3104
##
## Coefficients (mean model with logit link):
##                          Estimate Std. Error  z value Pr(>|z|)
## (Intercept)           -8.401e-01  1.743e-03 -482.060  < 2e-16 ***
## percentCharCumulative -8.464e-05  1.949e-05   -4.344  1.4e-05 ***
##
## Phi coefficients (precision model with log link):
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)           3.1811550  0.0330767   96.17   <2e-16 ***
## percentCharCumulative 0.0873846  0.0006701  130.41   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Type of estimator: ML (maximum likelihood)
## Log-likelihood: 1.741e+04 on 4 Df
## Pseudo R-squared: 0.009122
## Number of iterations: 17 (BFGS) + 2 (Fisher scoring)
```

```
px = predict(mBeta, newdata = data.frame(
  percentCharCumulative = seq(0,100,by=5)
))
px.var = predict(mBeta, type="variance", newdata = data.frame(
  percentCharCumulative = seq(0,100,by=5)
))

allGames.Cum$folder.factor = factor(allGames.Cum$folder)
mGAM = gam(diffFromTotalEstimate~
             s(percentCharCumulative) +
             s(folder.factor, bs = 're'),
           data = allGames.Cum[allGames.Cum$percentCharCumulative>5,])
summary(mGAM)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## diffFromTotalEstimate ~ s(percentCharCumulative) + s(folder.factor,
##     bs = "re")
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.1051     0.6181    0.17    0.865
##
## Approximate significance of smooth terms:
##                             edf Ref.df     F p-value
## s(percentCharCumulative)  4.354  5.355 60.05  <2e-16 ***
## s(folder.factor)         48.616 49.000 62.21  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.347   Deviance explained = 35.3%
## GCV = 7.6505  Scale est. = 7.5847    n = 6276
```

The model suggests that there is a significant relationship overall. We can visualise the curve to get an idea of the trends:

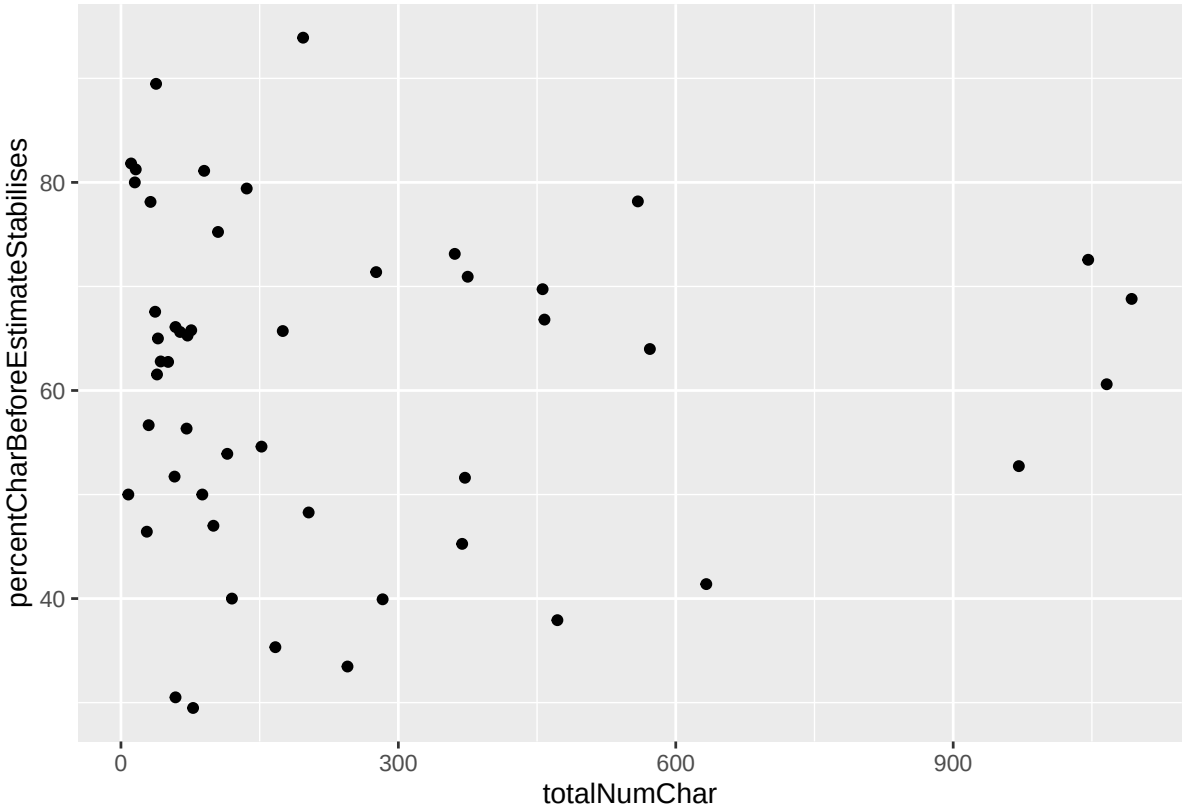
```
plot.gam(mGAM, xlab="Percentage of characters seen",
     ylab = "Difference from final % of female dialogue",
     select=1)
abline(h=0,col=1)
abline(v=seq(0,100,by=10), lty=2,col="gray")
```



In contrast to the boxplot, the visualisation suggests that female dialogue tends to be *over-estimated* for prolific characters, then slightly under-estimated after seeing about between 20-50% of characters. After about 60% of characters, the estimate stabilises. Note that the range is very small - predicting biases of around 1% at most.

The plot below shows how the stabilisation point varies with the total number of characters in the game. There's no strong relationship between the two.

```
ggplot(data=allGames.Cum[!duplicated(allGames.Cum$folder),],
       aes(y=percentCharBeforeEstimateStabilises,
           x =totalNumChar)) +
  geom_point()
```




```
cor.test(allGames.Cum[!duplicated(allGames.Cum$folder),]$percentCharBeforeEstimateStabilises,
         allGames.Cum[!duplicated(allGames.Cum$folder),]$totalNumChar)
```

```
##
##  Pearson's product-moment correlation
##
## data:  allGames.Cum[!duplicated(allGames.Cum$folder), ]$percentCharBeforeEstimateStabilises and a
## t = -0.091829, df = 48, p-value = 0.9272
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2905291  0.2660761
## sample estimates:
##         cor
## -0.01325316
```

Conclusion

The analyses above suggest that the gender bias in video game dialogue is present in major and minor characters. Although the bias was stronger for major characters, this can be mostly explained by the low number of female characters in this group, rather than a systematic difference between major and minor character groups *per-se*.

Furthermore, the estimate of the proportion of female characters is not systematically biased across games due to being able to get more complete or accurate data on more prolific characters. Individual games were biased in different directions, but with the majority converging after seeing 90% of the characters. The estimate of female dialogue across all games was not different from the final estimate after seeing the top 60% prolific characters in each game. For some ranges of the data, the average estimate of female dialogue across all games was an *under-estimate*, rather than an over-estimate. In any case, the bias in the estimate after seeing 20% of the characters are in the range of a few percentage points. This is much smaller than the overall gender bias in the corpus.

In conclusion, it's unlikely that the gender biases reported in the main data are affected by a tendency to have more complete or accurate data on more central characters.