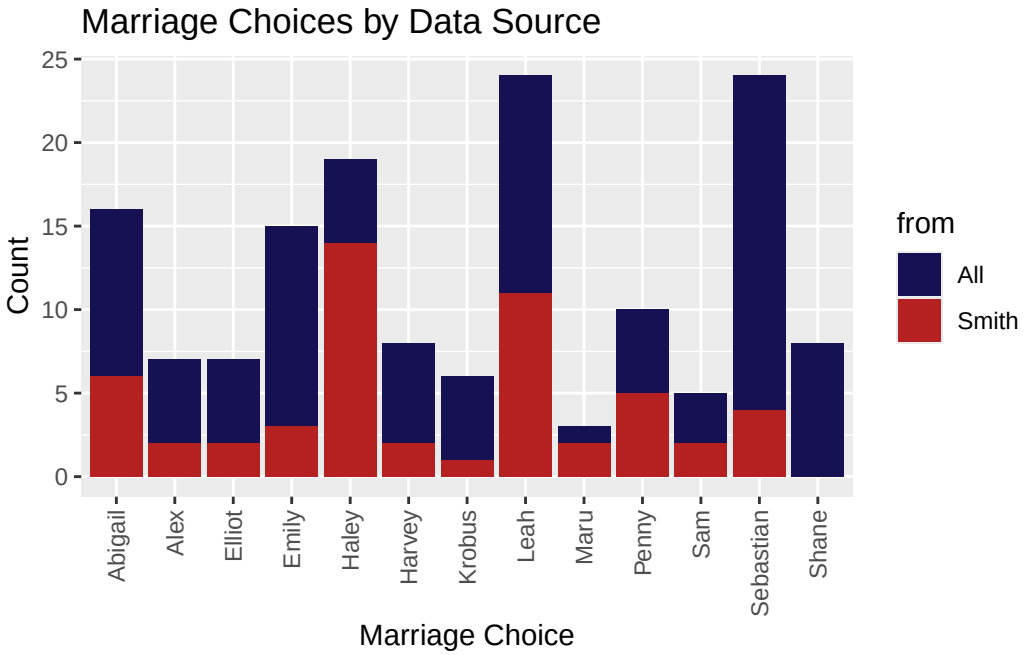
FP2

Kika Kovaleski, Molly Daniel, Emma Ruckle, Caroline Zouloumian, Sydney Weisberg

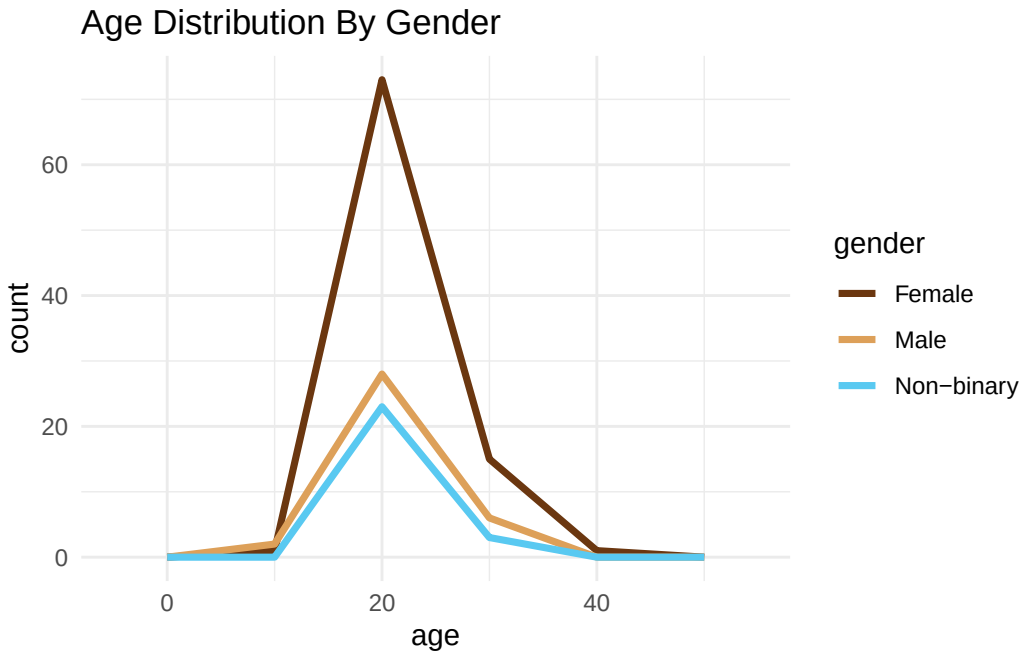
Loading and Cleaning the Data

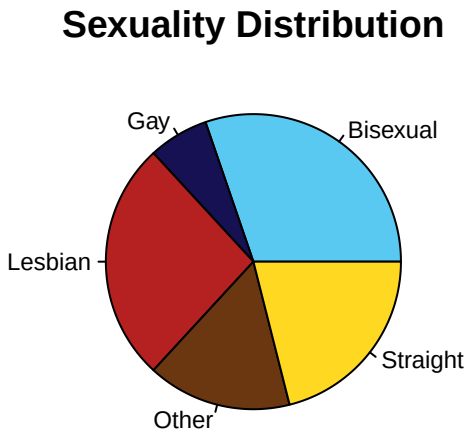
```
# reading in csvs and creating df
library(tidyverse)

stardew_all <- read.csv("stardew_all.csv") |> select(!Timestamp) |> mutate(from =
 ↪  "All")
stardew_smith <- read.csv("stardew_smith.csv") |> select(!Timestamp) |>
 ↪  mutate(from = "Smith")

stardew_data <- rbind(stardew_all, stardew_smith)

# filtering out observations where gender == "Other"
stardew_data <- stardew_data |> filter(gender != "Other")
```

Distribution of The Data



```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```





Splitting Up Data for Cross Validation

```
# separating the data into train and test data
```

```r
set.seed(1) # set seed

indices <- sample(seq(152), 106) # find 106 random indices (70% train data, 30%
 ↪  test data)

train <- stardew_data[indices,] # use those indices for training data

test_data <- stardew_data[-indices,] # use all the other indices for testing
test_marriage <- stardew_data$marriage_candidate[-indices] # get true marriage
 ↪  candidates for testing
```

Naive Bayes Model

```r
# Naive Bayes
library(e1071)
```

```
Warning: package 'e1071' was built under R version 4.4.3
```

```r
train <- indices
# fit Naive Bayes model
nb.fit <- naiveBayes(marriage_candidate ~ gender + sexuality + age, data =
 ↪  stardew_data, subset = train)
# display nb model summary
nb.fit
```

```
Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
    Abigail        Alex      Elliot       Emily       Haley      Harvey
0.075471698 0.037735849 0.066037736 0.084905660 0.113207547 0.047169811
     Krobus        Leah        Maru       Penny         Sam   Sebastian
0.047169811 0.216981132 0.009433962 0.056603774 0.047169811 0.141509434
      Shane
0.056603774

Conditional probabilities:
           gender
Y               Female       Male Non-binary
```

```
  Abigail   0.37500000 0.62500000 0.00000000
  Alex      0.25000000 0.75000000 0.00000000
  Elliot    0.85714286 0.00000000 0.14285714
  Emily     0.44444444 0.22222222 0.33333333
  Haley     0.58333333 0.16666667 0.25000000
  Harvey    1.00000000 0.00000000 0.00000000
  Krobus    0.60000000 0.00000000 0.40000000
  Leah      0.60869565 0.26086957 0.13043478
  Maru      1.00000000 0.00000000 0.00000000
  Penny     0.66666667 0.33333333 0.00000000
  Sam       0.40000000 0.60000000 0.00000000
  Sebastian 0.66666667 0.06666667 0.26666667
  Shane     0.33333333 0.33333333 0.33333333

          sexuality
Y            Bisexual        Gay    Lesbian      Other   Straight
  Abigail   0.25000000 0.00000000 0.12500000 0.00000000 0.62500000
  Alex      0.25000000 0.25000000 0.25000000 0.00000000 0.25000000
  Elliot    0.28571429 0.14285714 0.00000000 0.14285714 0.42857143
  Emily     0.11111111 0.00000000 0.55555556 0.11111111 0.22222222
  Haley     0.25000000 0.08333333 0.58333333 0.08333333 0.00000000
  Harvey    0.60000000 0.00000000 0.00000000 0.20000000 0.20000000
  Krobus    0.00000000 0.00000000 0.20000000 0.80000000 0.00000000
  Leah      0.08695652 0.00000000 0.56521739 0.08695652 0.26086957
  Maru      0.00000000 0.00000000 1.00000000 0.00000000 0.00000000
  Penny     0.16666667 0.00000000 0.33333333 0.16666667 0.33333333
  Sam       0.40000000 0.40000000 0.00000000 0.00000000 0.20000000
  Sebastian 0.40000000 0.06666667 0.06666667 0.33333333 0.13333333
  Shane     0.66666667 0.00000000 0.00000000 0.33333333 0.00000000

          age
Y              [,1]     [,2]
  Abigail   22.00000 4.000000
  Alex      20.12500 1.436141
  Elliot    21.28571 2.288689
  Emily     21.22222 3.032234
  Haley     20.16667 1.466804
  Harvey    25.20000 5.069517
  Krobus    25.60000 4.159327
  Leah      21.65217 4.923124
  Maru      20.00000       NA
  Penny     21.16667 3.868678
  Sam       20.60000 2.701851
  Sebastian 23.06667 4.431489
  Shane     22.83333 5.231316
```

```r
nb.class <- predict(nb.fit, test_data) # use nb model on test data to predict
 ↪  marriage candidate
table(nb.class, test_data$marriage_candidate) # nb model predictions are compared
 ↪  to real observations
```

nb.class	Abigail	Alex	Emily	Haley	Harvey	Krobus	Leah	Maru	Penny	Sebastian
Abigail	2	0	1	0	0	0	0	0	1	0
Alex	0	0	0	0	1	0	0	0	0	0
Elliot	0	0	0	0	0	0	0	0	0	2
Emily	0	0	0	0	0	0	0	0	0	0
Haley	3	0	0	5	1	0	0	1	1	2
Harvey	0	0	0	0	0	0	0	0	0	1
Krobus	0	1	1	0	0	0	0	0	0	0
Leah	0	0	1	1	0	0	1	0	0	0
Maru	0	0	0	0	0	0	0	0	0	0
Penny	0	0	0	0	0	0	0	0	0	0
Sam	1	2	0	0	0	0	0	0	0	0
Sebastian	2	0	2	1	1	1	0	1	2	4
Shane	0	0	1	0	0	0	0	0	0	0

nb.class	Shane
Abigail	1
Alex	0
Elliot	0
Emily	0
Haley	0
Harvey	0
Krobus	0
Leah	0
Maru	0
Penny	0
Sam	0
Sebastian	1
Shane	0

```r
mean(nb.class == test_data$marriage_candidate) # proportion of correct
 ↪  predictions is calculated
```

```
[1] 0.2608696
```

Our Naive Bayes model correctly predicts marriage candidate 26% of the time.