WilliamsFaculty

Yiheng Zhang 18'

February 8, 2016

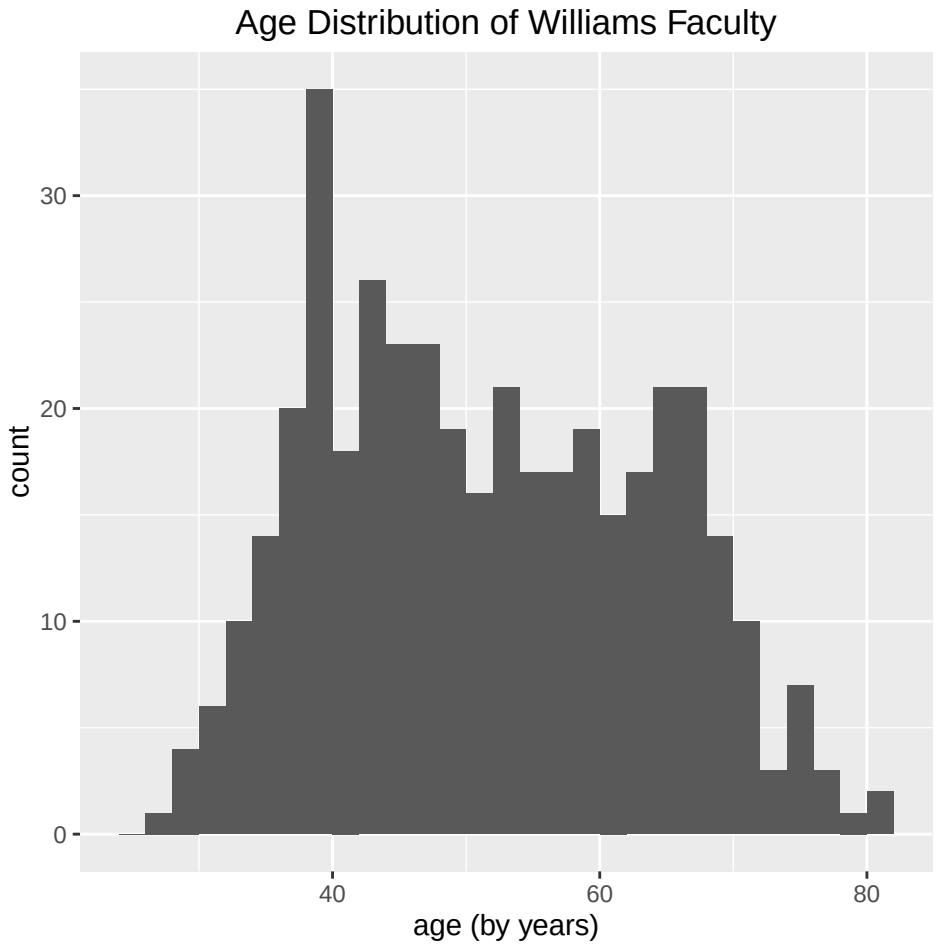
1 Introduction

This package explores the age data of all Williams faculty, with special attention paid to two factors: department and gender. The package provides functions that generate summary tables and interesting visual presentations of the data. Some functions generate different summaries and plots depending on the user's input, setting the focus to department or gender. In addition, the package also contains a function that performs statistical analysis on the data. The original question that this package seeks to answer is: **What is the average age of Williams faculty?** This package contains a function `average_age` that answers that question directly, offering a numerical value. The other functions explore visually and statistically the relationship between age and gender/department. The package relies on ggplot2 for interesting plots and DT for interactive data tables.

2 Data

The data frame used for this package is internal and comes with the package itself. The original data is retrieved from `http://web.williams.edu/admin/registrar//catalog/archive.html` under the 2013-14 school year. The professor names, departments, genders, and ages are compiled by hand into an excel file which can be found under `/inst/extdata`, which is then changed into an RData file through the use of the "xlsx" package. The names and departments are copied directly from the Williams archive, but the gender is guessed from the professor's first name, and the ages are estimated using the year the professor received his/her BA, assuming that the professor is 22 years old at that time. A few visiting professors/lecturers whose education records were not locatable have been omitted from the dataset. Some departments with very few professors have been combined with other departments in the same field to form categories, such as `"Physics/Astronomy"` and `"Theatre/Dance"`. The following is a histagram of all the ages generated by one of the functions:

```
plot_age()
```



The distribution is slightly right-skewed. More characteristics of the distribution will be included in a summary table in the next section.

3 Use WilliamsFaculty

First and foremost, to find out the average age of the Williams faculty, simply use the function `average_age`, which returns a numerical value (all numerical values are limited to 2 decimal digits). To find out more about the age data in general, the user can use the function `plot_age` to generate the generic histogram as shown above, and `age_summary` to return a summary table like this:

```
age_summary()
```

```
##             attributes     attribute_data
## 1 Number of Professors                403
## 2        Range of Ages                 53
## 3              Oldest Donald deB. Beaver
## 4            Youngest Sarah A. Mirseyedi
## 5         Average Age              50.99
## 6   Standard Deviation              12.18
## 7             Variance             148.35
```

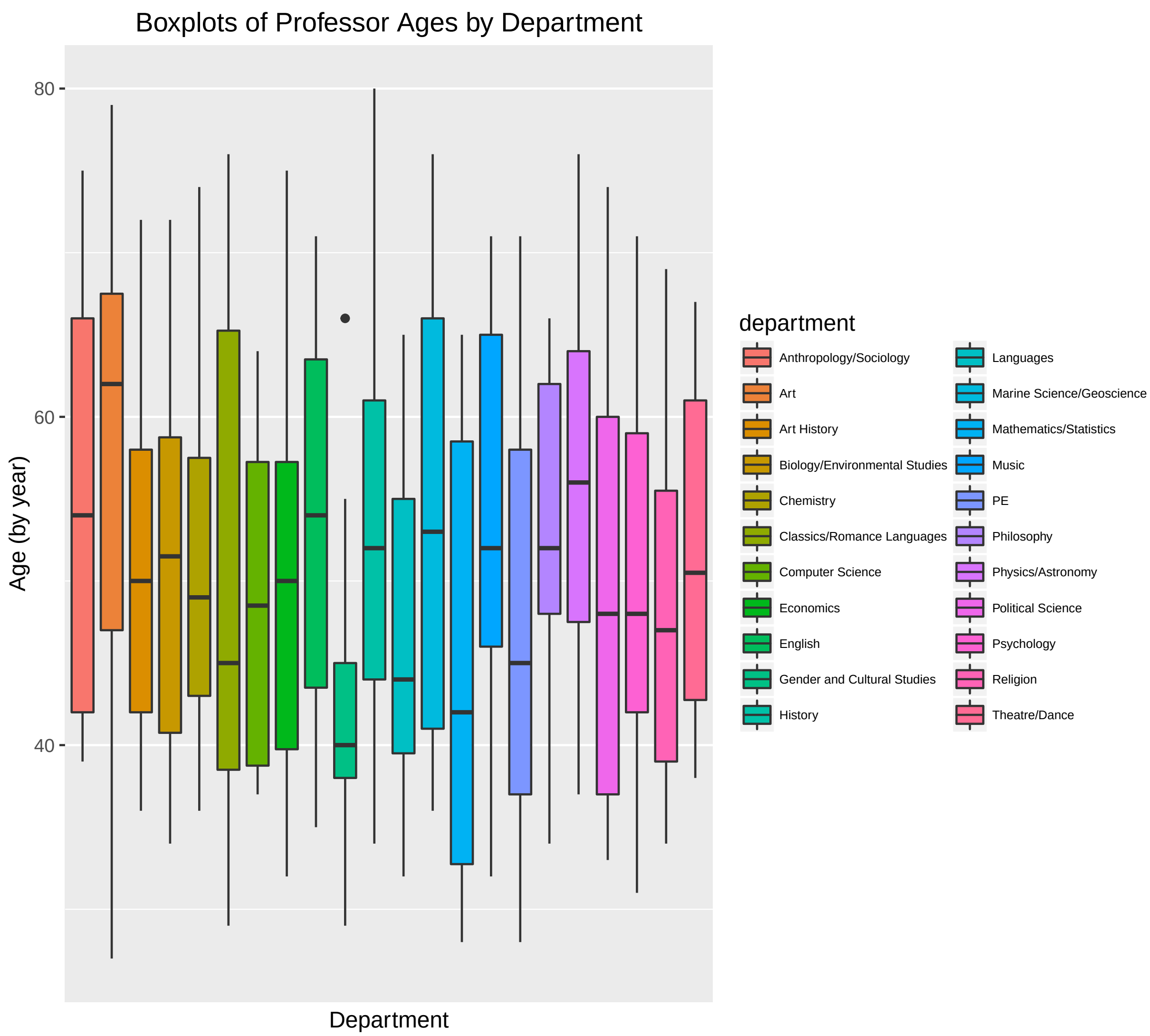
The user can also set `dpmt=TRUE` to generate a DT table with data focusing on departments:

```r
age_summary(dpmt=TRUE)
```

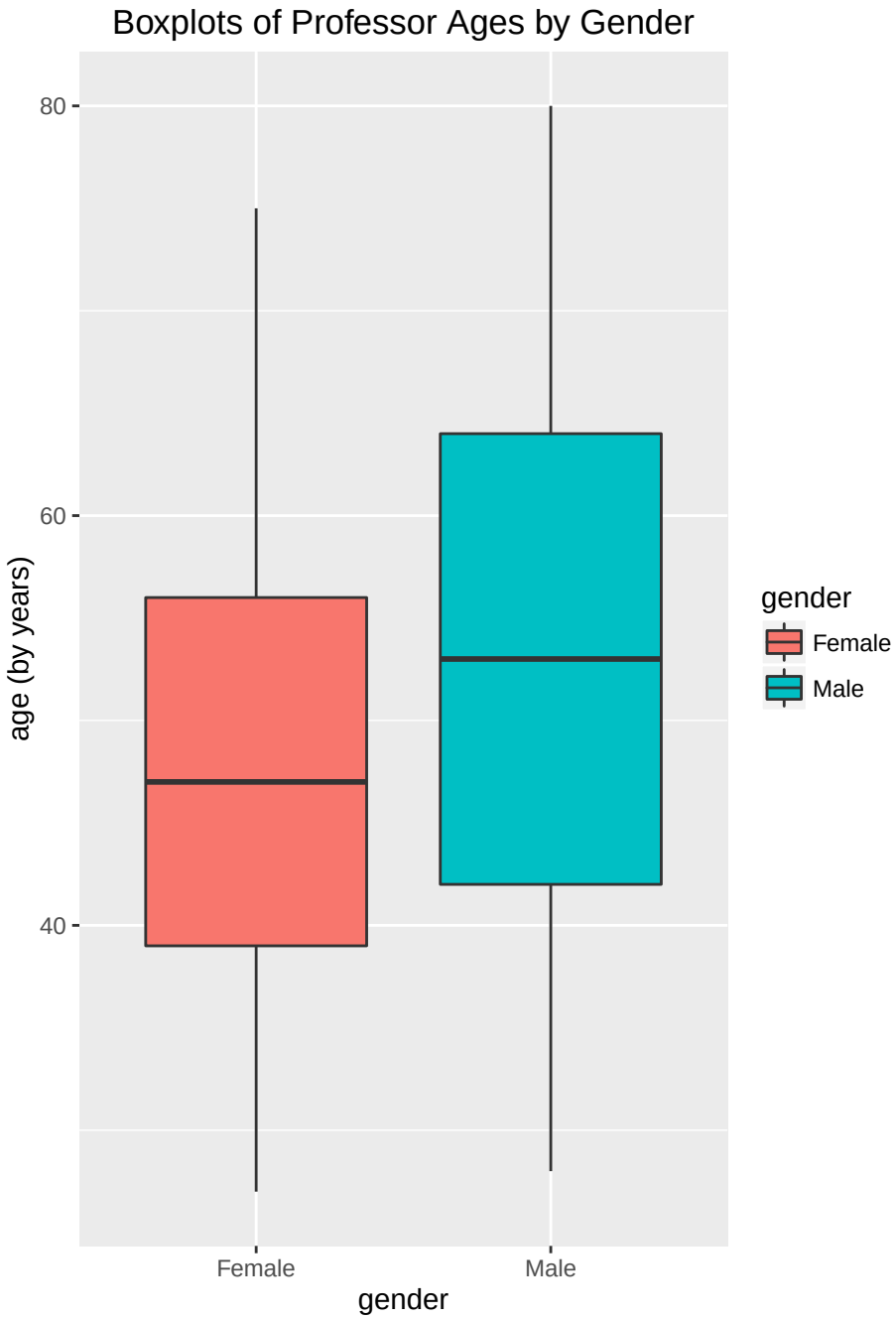
<!–html_preserve– – >< divid = ”htmlwidget – 7033”style = ”width : 100 < scripttype = ”application/json”data – for = ”htmlwidget – 7033” > ”x” : ”data” : [[”1”,”2”,”3”,”4”,”5”,”6”,”7”,”8”,”9”,”10”,”11”,”12”,”13”,”14”,”15”,”16”,” /script ><! – – /html_preserve – – >

To visualize the department and gender data, the user can use function `plot_by_dpmt`, `plot_by_gender`,`color_by_dpmt`, and `color_by_gender`. The first two functions generates boxplots of ages of each department and gender:

```r
plot_by_dpmt()
```

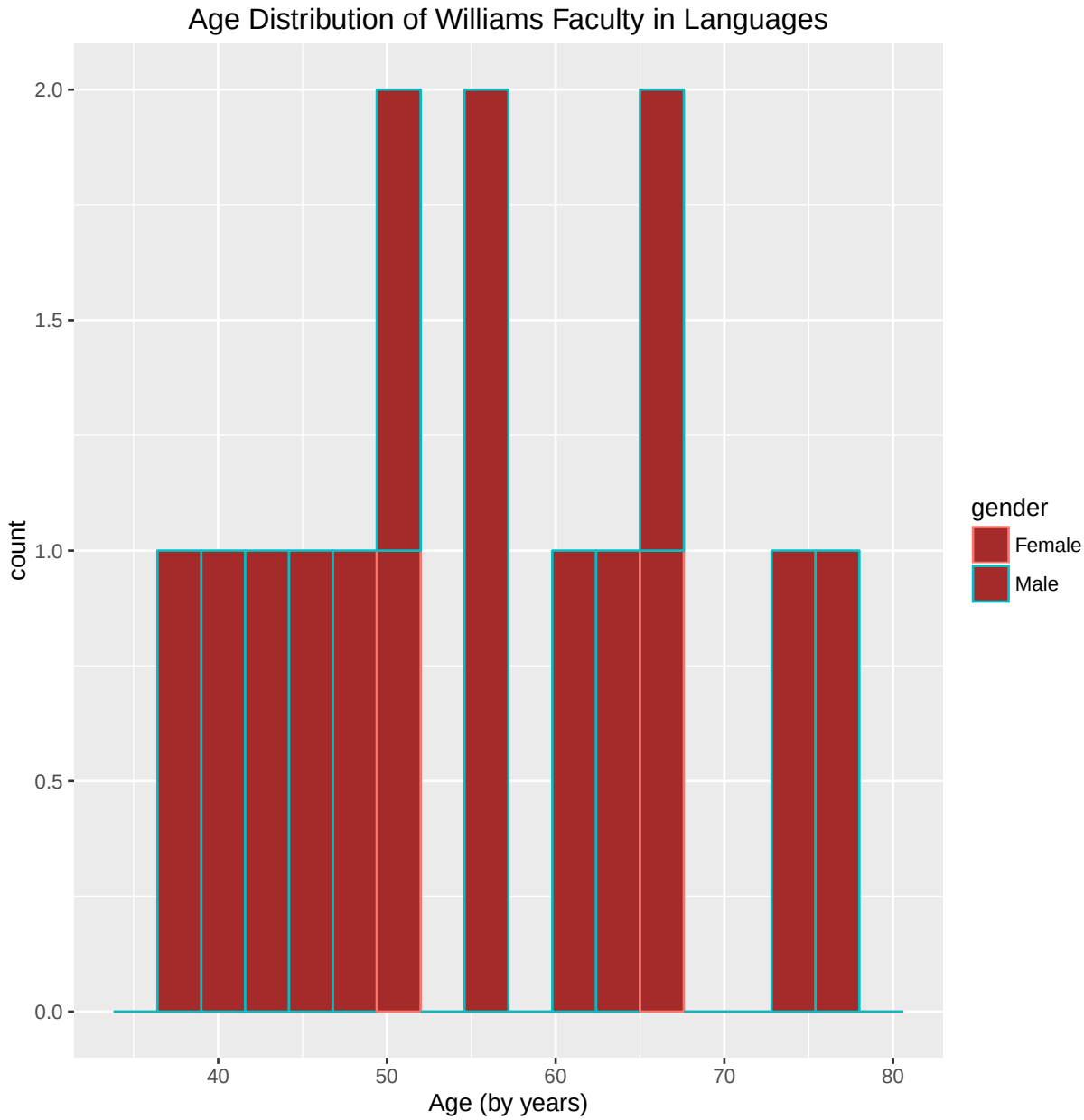



```r
plot_by_gender()
```



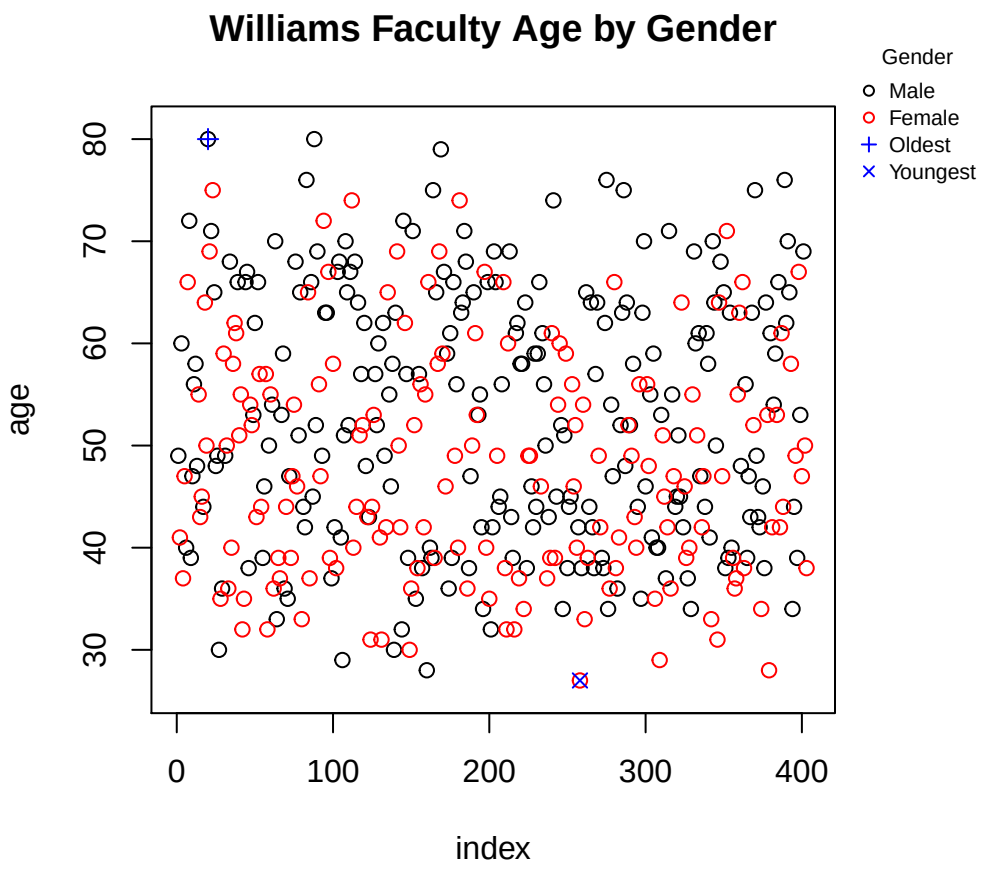
The function `plot_by_dpmt` can also take in input. If the user wishes to see the distribution of ages within a specific department, the user can simply input the department like this: `plot_by_dpmt`*"Languages"*.

```
plot_by_dpmt("Languages")
```

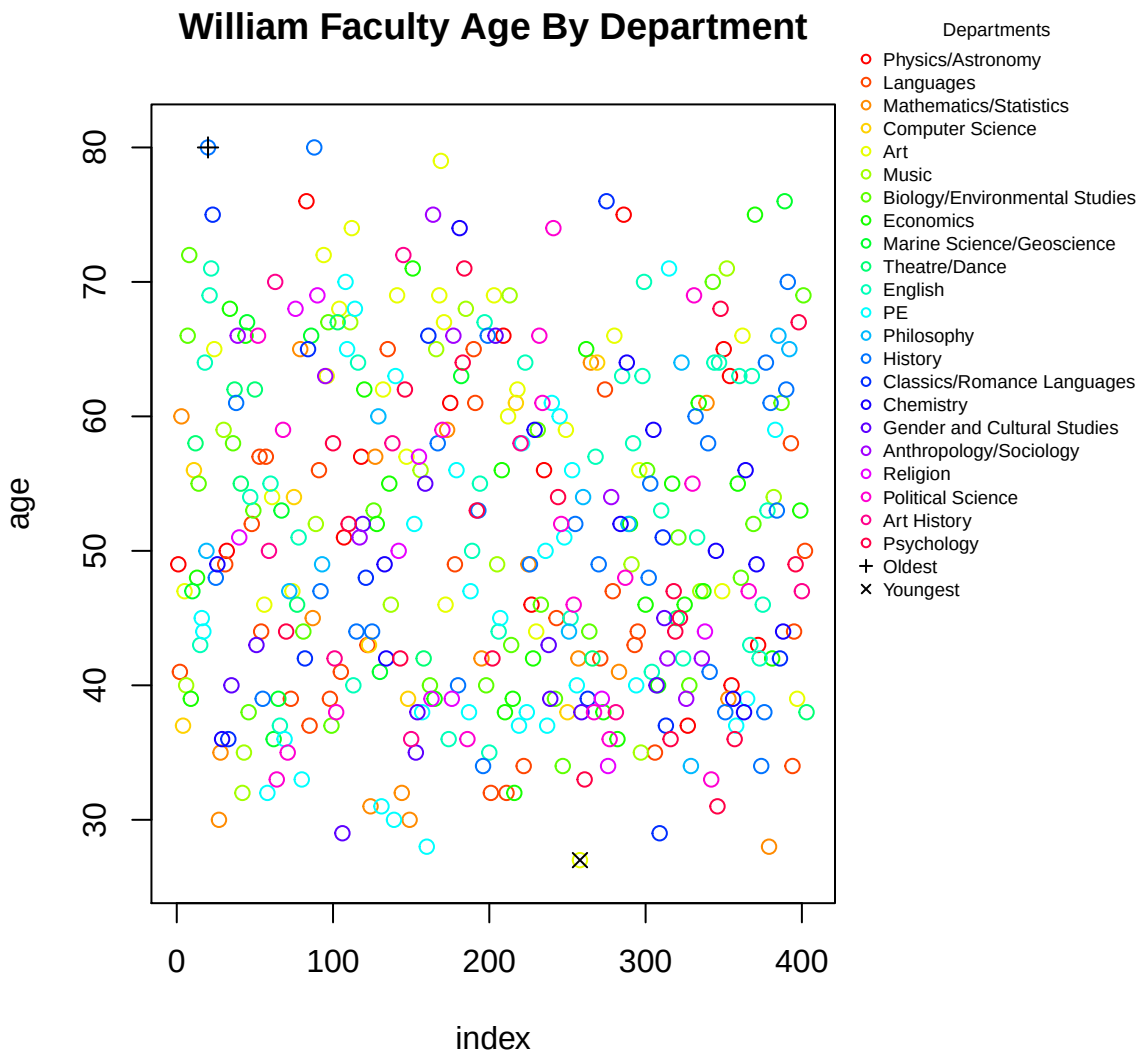


Note that the histograms are also outlined to give a sense of the gender division within that department. The latter two functions generate colored scatter plots that present each professor as a point in the scatter plot. The `color_by_gender` function separates the male and female professors by color, while the `color_by_dpmt` function separates professors in each department by color:

```
color_by_gender()
```

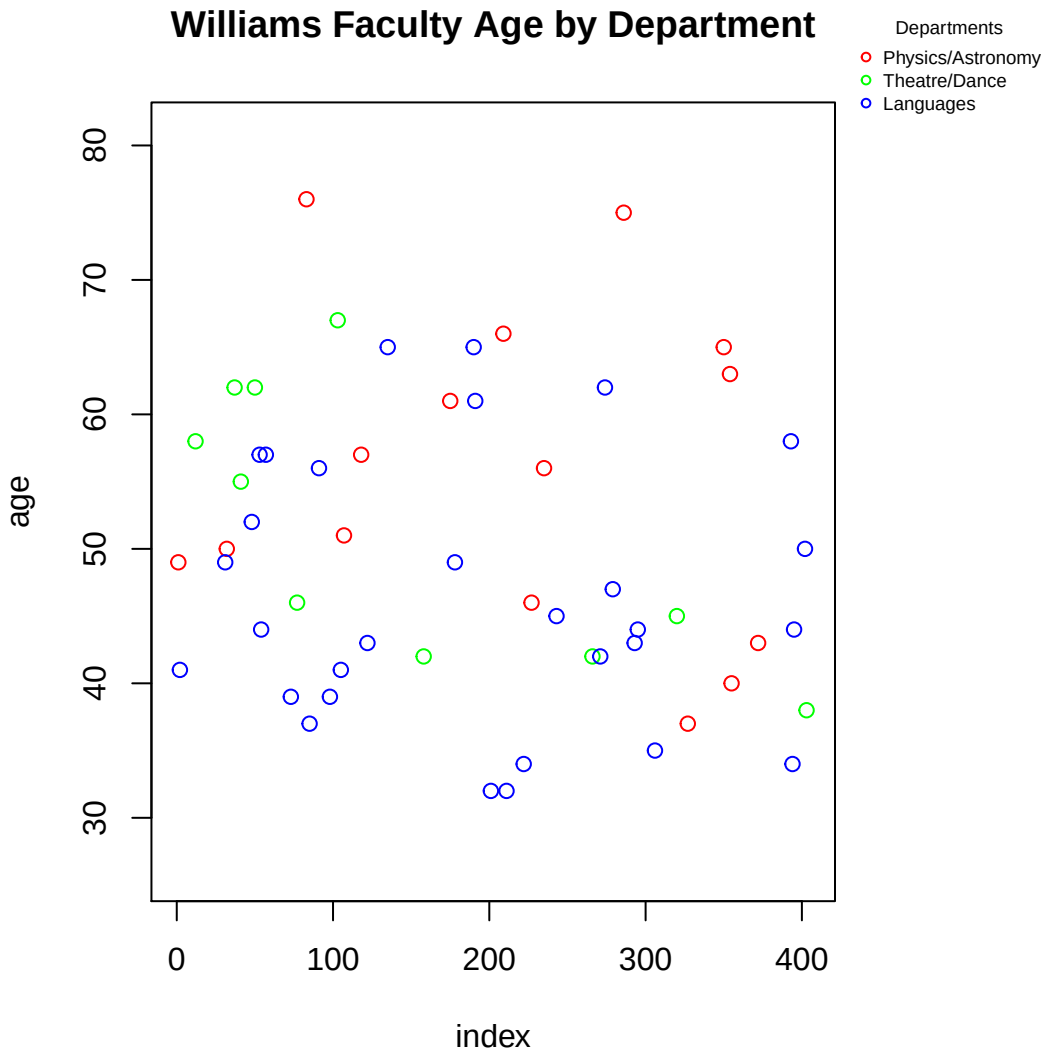



```
color_by_dpmt()
```



Note that the youngest and oldest professors are marked. The index (x-axis) is the professor's index in the original data, and since the data is in alphabetical order, the index gives a sense of what the professor's last name starts with. Again, the user can see data about any specific department(s) with `color_by_dpmt` if the user gives an input.

```
color_by_dpmt(c("Physics/Astronomy","Theatre/Dance","Languages"))
```



4 Results

Some important results include the mean age, and the results from the statistical analysis performed on the data with regards to department and gender. The mean age is about 51, calculated with `average_age`.

```
average_age()

## [1] 50.99
```

To explore the relationship between gender and age, the user can use `test_age`$gender = TRUE, dpmt = FALSE$, which first performs a binomial test on the number of female professors, and then performs a two-sample t-test on the ages of professors of each gender.

```
test_age(gender=TRUE, dpmt=FALSE)

##
##  Exact binomial test
```

```
##
## data:  number_of_female_professors and number_of_professors
## number of successes = 170, number of trials = 403, p-value = 0.0009876
## alternative hypothesis: true probability of success is less than 0.5
## 95 percent confidence interval:
##  0.0000000 0.4638723
## sample estimates:
## probability of success
##              0.4218362
##
## For the t-test, the p-value is 1.202e-05, null hypothesis rejected at 0.05 significance level.
## So we reject the null hypothesis that the mean ages of male and female professors are the same.
##
##  Welch Two Sample t-test
##
## data:  male_ages and female_ages
## t = 4.2762, df = 382.45, p-value = 1.202e-05
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  3.115053      Inf
## sample estimates:
## mean of x mean of y
##  53.12876  48.05882
```

To explore the relationship between department and age, the user can use `test_age`$gender = FALSE, dpmt = TRUE$, which performs an analysis of variance on the number of professors of each department.

```
test_age(gender=FALSE, dpmt=TRUE)

## The p-value is 0.0244, null hypothesis rejected at 0.05 significance level.
## So we reject the null hypothesis that all departments have the same mean age.
##              Df Sum Sq Mean Sq F value Pr(>F)
## department   21   5193   247.3   1.731 0.0244 *
## Residuals   381  54442   142.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

To explore whether department and gender are independent, the user can use `test_age`, which performs a chi-squared test of independence on the number of professors of each gender and in each department.

```
test_age()

## Null hypothesis is not rejected at 0.05 significance level. So we do not reject the null
## hypothesis that gender is independent from department with regards to professor ages
##
##  Pearson's Chi-squared test with simulated p-value (based on 2000 replicates)
##
## data:  tbl
## X-squared = 30.486, df = NA, p-value = 0.08696
```

The results of the analysis seem to suggest that:

- There are more male professors than female professors.
- Male professors may be older than female professors on average.

- The department that the professors are in may be related with the ages of the professors.
- There seem not to be any relation between the professor's gender and the department that he/she is in.

5 Conclusion

The **WilliamsFaculty** package calculates the average age of Williams College faculty, and presents the data in an interesting and interactive way. In addition, the package analyzes the relationships between age and department and between age and gender, providing functions to perform statistical analysis and also visualization of the relationships.

The package contains an internal data frame, called **data**, and the package's contents only work with that particular data frame. A helpful future modification would be to allow users to import their own data frames or raw data files.