Data_Cleaning

Caleb Neale

3/23/2021

Importing Data

```
fac_names = read.csv("fac_names_depts.csv")
salaries = read.csv("2020_cav_daily_data.csv")
```

Applying gender funciton to fac_names

Documentation and methodology found here, see page 3-4: https://cran.r-project.org/web/packages/gender/gender.pdf

```
library(gender)
```

```
## Warning: package 'gender' was built under R version 4.0.4
```

```
## PLEASE NOTE: The method provided by this package must be used cautiously
## and responsibly. Please be sure to see the guidelines and warnings about
## usage in the README or the package documentation.
```

```
library(tidyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# separate first names
fac_names %>% separate(full_name, c("first_name", "other_names"), sep=" ", extra="merge") -> fac_names
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 9 rows [1, 485,
## 734, 995, 2399, 2537, 2545, 2596, 3179].
```

```
# assume oldest professor is 75, youngest is 35

fac_names$first_name %>% gender(years=c(1946, 1986)) %>% select("name", "gender") -> name_guesses

# merge gender data back into fac_names
fac_names %>% left_join(name_guesses, by=c("first_name" = "name")) -> fac_names_gender

# remove duplicates generated by merge
fac_names_gender %>% distinct() -> fac_names_gender

# check counts
fac_names_gender %>% count(gender)
```

```
##   gender    n
## 1 female 1242
## 2   male 1747
## 3   <NA>  318
```

```
# save as csv
fac_names_gender %>% write.csv("fac_names_gender.csv")
```

Applying gender funtion to salaries data

```
# assume oldest professor is 75, youngest is 35

salaries$First.Name %>% gender(years=c(1946, 1986)) %>% select("name", "gender") -> name_guesses_salari

# merge gender data back into salaries
salaries %>% left_join(name_guesses_salaries, by=c("First.Name" = "name")) -> salaries_gender

# remove duplicates generated by merge
salaries_gender %>% distinct() -> salaries_gender

# check counts
salaries_gender %>% count(gender)
```

```
##   gender     n
## 1 female 10470
## 2   male  6731
## 3   <NA>  1470
```

```
# save as csv
salaries_gender %>% write.csv("fac_names_gender.csv")
```