Data Analysis using R

******

2023-11-15

Loading required libraries

```
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
```

Question 1: Reading the dataset into R

```
cdrc_data <- read.csv("hh_churn_lsoa11_2023.csv")
str(cdrc_data)
```

```
## ’data.frame':    42619 obs. of  28 variables:
##  $ area   : chr  "95AA01S1" "95AA01S2" "95AA01S3" "95AA02W1" ...
##  $ chn1997: num  1 0.663 0.7 0.591 0.602 0.633 0.658 0.685 0.675 0.539 ...
##  $ chn1998: num  1 0.651 0.688 0.576 0.591 0.618 0.64 0.663 0.644 0.522 ...
##  $ chn1999: num  1 0.636 0.671 0.56 0.586 0.601 0.61 0.638 0.607 0.498 ...
##  $ chn2000: num  1 0.629 0.663 0.549 0.584 0.593 0.591 0.624 0.584 0.483 ...
##  $ chn2001: num  1 0.618 0.653 0.542 0.581 0.583 0.577 0.615 0.569 0.473 ...
##  $ chn2002: num  1 0.603 0.633 0.537 0.569 0.56 0.559 0.596 0.55 0.458 ...
##  $ chn2003: num  1 0.59 0.611 0.534 0.554 0.535 0.537 0.566 0.523 0.443 ...
##  $ chn2004: num  1 0.569 0.591 0.529 0.541 0.514 0.512 0.538 0.485 0.427 ...
##  $ chn2005: num  1 0.541 0.57 0.524 0.524 0.494 0.492 0.521 0.444 0.408 ...
##  $ chn2006: num  1 0.517 0.555 0.518 0.51 0.478 0.48 0.506 0.412 0.391 ...
##  $ chn2007: num  1 0.506 0.548 0.517 0.506 0.467 0.475 0.489 0.395 0.378 ...
##  $ chn2008: num  1 0.495 0.538 0.506 0.502 0.46 0.465 0.47 0.374 0.366 ...
##  $ chn2009: num  1 0.48 0.528 0.49 0.494 0.439 0.445 0.44 0.343 0.35 ...
##  $ chn2010: num  1 0.47 0.522 0.48 0.487 0.415 0.43 0.417 0.322 0.332 ...
##  $ chn2011: num  1 0.462 0.515 0.47 0.478 0.402 0.42 0.406 0.311 0.321 ...
##  $ chn2012: num  1 0.446 0.495 0.448 0.454 0.387 0.402 0.385 0.292 0.307 ...
##  $ chn2013: num  1 0.429 0.469 0.411 0.419 0.369 0.379 0.362 0.274 0.292 ...
##  $ chn2014: num  1 0.416 0.453 0.387 0.398 0.353 0.364 0.351 0.268 0.284 ...
##  $ chn2015: num  0.969 0.397 0.435 0.362 0.371 0.335 0.354 0.333 0.256 0.276 ...
##  $ chn2016: num  0.928 0.373 0.412 0.331 0.337 0.314 0.344 0.308 0.239 0.267 ...
##  $ chn2017: num  0.914 0.336 0.367 0.297 0.304 0.281 0.317 0.273 0.217 0.25 ...
##  $ chn2018: num  0.879 0.287 0.31 0.262 0.27 0.239 0.282 0.232 0.188 0.227 ...
##  $ chn2019: num  0.799 0.256 0.279 0.243 0.252 0.216 0.264 0.207 0.169 0.21 ...
##  $ chn2020: num  0.723 0.244 0.266 0.233 0.244 0.205 0.257 0.192 0.16 0.199 ...
##  $ chn2021: num  0.537 0.186 0.182 0.172 0.177 0.145 0.189 0.132 0.11 0.145 ...
##  $ chn2022: num  0.29 0.102 0.072 0.087 0.081 0.066 0.09 0.055 0.044 0.071 ...
##  $ chn2023: int  0 0 0 0 0 0 0 0 0 0 ...
```

Question 2: Examining the LSOA code W01000092 Menai (Bangor), and identifying the index which W01000092 is greater than 0.5 and less than 0.8.

```
cdrc_data_menai <- filter(cdrc_data, area == "W01000092")
cdrc_data_menai_W01000092 <- unlist(cdrc_data_menai[1, -1])

cdrc_data_menai_sub <- cdrc_data_menai_W01000092[cdrc_data_menai_W01000092 > 0.5 & cdrc_data_menai_W0100
print(cdrc_data_menai_sub)
```

```
## chn2002 chn2003 chn2004 chn2005 chn2006 chn2007 chn2008 chn2009 chn2010 chn2011
##   0.791   0.783   0.776   0.771   0.765   0.758   0.752   0.747   0.743   0.729
## chn2012 chn2013 chn2014 chn2015 chn2016 chn2017
##   0.701   0.664   0.634   0.609   0.579   0.529
```

```
print(names(cdrc_data_menai_sub))
```

```
##  [1] "chn2002" "chn2003" "chn2004" "chn2005" "chn2006" "chn2007" "chn2008"
##  [8] "chn2009" "chn2010" "chn2011" "chn2012" "chn2013" "chn2014" "chn2015"
## [15] "chn2016" "chn2017"
```

Question 3: appending additional columns to the data

```
cdrc_data$ARMI <- round(rowMeans(cdrc_data[, -c(1, 28)]), 3)
cdrc_data$ARMIctgrs <- cut(cdrc_data$ARMI, breaks = c(0, 0.2, 0.5, 1), labels = c("Low", "Medium", "Hig
str(cdrc_data)
```

```
## ’data.frame’:    42619 obs. of  30 variables:
##  $ area     : chr  "95AA01S1" "95AA01S2" "95AA01S3" "95AA02W1" ...
##  $ chn1997  : num  1 0.663 0.7 0.591 0.602 0.633 0.658 0.685 0.675 0.539 ...
##  $ chn1998  : num  1 0.651 0.688 0.576 0.591 0.618 0.64 0.663 0.644 0.522 ...
##  $ chn1999  : num  1 0.636 0.671 0.56 0.586 0.601 0.61 0.638 0.607 0.498 ...
##  $ chn2000  : num  1 0.629 0.663 0.549 0.584 0.593 0.591 0.624 0.584 0.483 ...
##  $ chn2001  : num  1 0.618 0.653 0.542 0.581 0.583 0.577 0.615 0.569 0.473 ...
##  $ chn2002  : num  1 0.603 0.633 0.537 0.569 0.56 0.559 0.596 0.55 0.458 ...
##  $ chn2003  : num  1 0.59 0.611 0.534 0.554 0.535 0.537 0.566 0.523 0.443 ...
##  $ chn2004  : num  1 0.569 0.591 0.529 0.541 0.514 0.512 0.538 0.485 0.427 ...
##  $ chn2005  : num  1 0.541 0.57 0.524 0.524 0.494 0.492 0.521 0.444 0.408 ...
##  $ chn2006  : num  1 0.517 0.555 0.518 0.51 0.478 0.48 0.506 0.412 0.391 ...
##  $ chn2007  : num  1 0.506 0.548 0.517 0.506 0.467 0.475 0.489 0.395 0.378 ...
##  $ chn2008  : num  1 0.495 0.538 0.506 0.502 0.46 0.465 0.47 0.374 0.366 ...
##  $ chn2009  : num  1 0.48 0.528 0.49 0.494 0.439 0.445 0.44 0.343 0.35 ...
##  $ chn2010  : num  1 0.47 0.522 0.48 0.487 0.415 0.43 0.417 0.322 0.332 ...
##  $ chn2011  : num  1 0.462 0.515 0.47 0.478 0.402 0.42 0.406 0.311 0.321 ...
##  $ chn2012  : num  1 0.446 0.495 0.448 0.454 0.387 0.402 0.385 0.292 0.307 ...
##  $ chn2013  : num  1 0.429 0.469 0.411 0.419 0.369 0.379 0.362 0.274 0.292 ...
##  $ chn2014  : num  1 0.416 0.453 0.387 0.398 0.353 0.364 0.351 0.268 0.284 ...
##  $ chn2015  : num  0.969 0.397 0.435 0.362 0.371 0.335 0.354 0.333 0.256 0.276 ...
##  $ chn2016  : num  0.928 0.373 0.412 0.331 0.337 0.314 0.344 0.308 0.239 0.267 ...
##  $ chn2017  : num  0.914 0.336 0.367 0.297 0.304 0.281 0.317 0.273 0.217 0.25 ...
##  $ chn2018  : num  0.879 0.287 0.31 0.262 0.27 0.239 0.282 0.232 0.188 0.227 ...
##  $ chn2019  : num  0.799 0.256 0.279 0.243 0.252 0.216 0.264 0.207 0.169 0.21 ...
##  $ chn2020  : num  0.723 0.244 0.266 0.233 0.244 0.205 0.257 0.192 0.16 0.199 ...
##  $ chn2021  : num  0.537 0.186 0.182 0.172 0.177 0.145 0.189 0.132 0.11 0.145 ...
##  $ chn2022  : num  0.29 0.102 0.072 0.087 0.081 0.066 0.09 0.055 0.044 0.071 ...
##  $ chn2023  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ ARMI     : num  0.925 0.458 0.489 0.429 0.439 0.412 0.428 0.423 0.364 0.343 ...
##  $ ARMIctgrs: Factor w/ 3 levels "Low","Medium",..: 3 2 2 2 2 2 2 2 2 2 ...
```