simulation-study

Tony Ni

1/20/2021

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts ------------------------------------------------------------------------------------ tidy
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(Metrics) #package to help calculate mse
```

```
## Warning: package 'Metrics' was built under R version 4.0.3
```

```
library(NADA) #package with implementation of many methods
```

```
## Loading required package: survival
```

```
##
## Attaching package: 'NADA'
```

```
## The following object is masked from 'package:stats':
##
##     cor
```

First, a function to help us generate our desired dataset, where users can specify the mean, sd, sample size (of each run), and censoring rate:

What do we want our code to do?

We first need to generate simulated data sets with the following combinations of censoring rates and sample sizes: (censoring rates: 10, 30, 50, sample sizes: 10, 100, 1000)

For each censoring/sample size pair, we create a simulated dataset with those specifications and run the substitution method with a specified number of iterations (`iterations`). For each sample, we will obtain the true mean (when considering all values to be uncensored), the estimated mean (when considering the uncensored values and the imputed values), alongside the MSE of each dataset

the average mse is ultimately what we are interested in

```
knitr::kable(df.final)
```

combinations	substitution	mle	km	ros
0.1, 10	0.0008124	0.0001906	0.0000408	0.0002694
0.1, 100	0.0001925	0.0000378	0.0001260	0.0000250
0.1, 1000	0.0002385	0.0000013	0.0000757	0.0000024
0.3, 10	0.0043468	0.0013788	0.0026348	0.0023965
0.3, 100	0.0032445	0.0006639	0.0026627	0.0000521
0.3, 1000	0.0029305	0.0005835	0.0024473	0.0000384
0.5, 10	0.0105395	0.0088390	0.0281452	0.0131503
0.5, 100	0.0095159	0.0087439	0.0149960	0.0001720
0.5, 1000	0.0088103	0.0089044	0.0165289	0.0002318