# ETC5242Assignment

Sahinya Akila

9/4/2021

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.3     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
## Remove the line break in the file name!
churn_dat <- read_csv("https://raw.githubusercontent.com/square/pysurvival/master/pysurvival/datasets/cl
```

```
## Rows: 2000 Columns: 14

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (5): product_travel_expense, product_payroll, product_accounting, compan...
## dbl (9): product_data_storage, csat_score, articles_viewed, smartphone_notif...

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
churn_dat <- churn_dat %>% filter(months_active > 0) %>% select(c(months_active, churned, company_size))
```

```
km_model <- function(time, event){
  dataset <- data_frame(time, event)

  km_data <- dataset %>%
    group_by(time, event) %>%
    summarise(died = n()) %>%
    ungroup() %>%
    mutate(risk = nrow(dataset) - accumulate(died, `+`) + died) %>%
    filter(event == 1) %>%
```

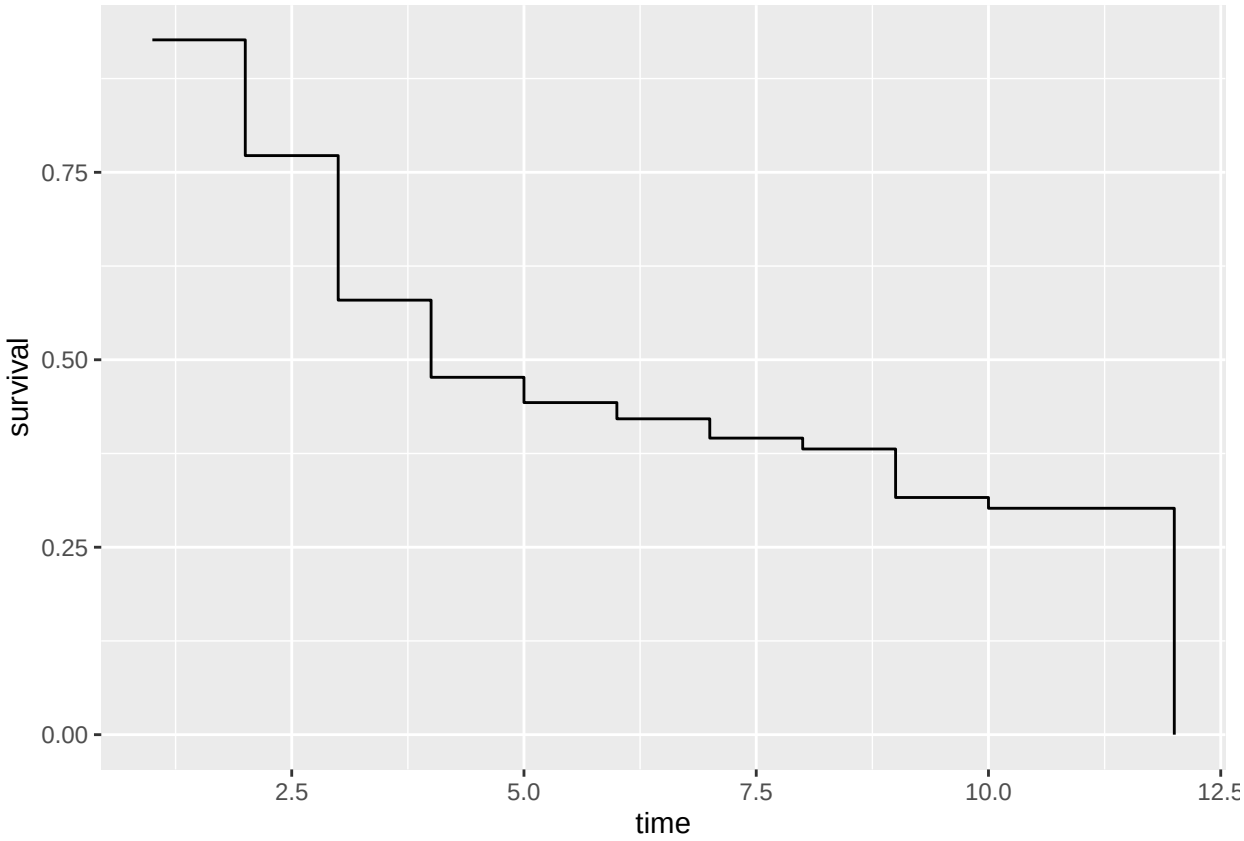
```
    mutate(probability = 1 - died/risk,
           survival = accumulate(probability, `*`))
  return(km_data %>% select(time, survival))
}

km_survive <- km_model(churn_dat$months_active, churn_dat$churned)
```

```
## Warning: ‘data_frame()‘ was deprecated in tibble 1.1.0.
## Please use ‘tibble()‘ instead.
```

```
## ‘summarise()‘ has grouped output by ’time’. You can override using the ‘.groups‘ argument.
```

```
km_survive %>%
  ggplot(aes(time, survival)) +
  geom_step()
```




```
company_km_model <- data.frame(time = double(), survival = double(), company_size = character())
for(size in unique(churn_dat$company_size)){
  filtered <- churn_dat %>% filter(company_size == size)
  final_model <- km_model(filtered$months_active, filtered$churned) %>% mutate(company_size = size)
  company_km_model <- rbind(company_km_model, final_model)
}
```

```
## ‘summarise()‘ has grouped output by ’time’. You can override using the ‘.groups‘ argument.
```

```
## `summarise()` has grouped output by 'time'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'time'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'time'. You can override using the `.groups` argument.
## `summarise()` has grouped output by 'time'. You can override using the `.groups` argument.
```

```
company_km_model %>%
  ggplot(aes(time, survival)) +
  geom_step() +
  facet_wrap(~company_size)
```