Analise Exploratoria do setor aquaviario

Melquisadec

2024-07-03

```
if (!require("pacman")) install.packages("pacman")
```

```
## Carregando pacotes exigidos: pacman
```

```
p_load(tidyverse, xtable,extRemes,ggplot2,quantmod,Quandl,ecostats,latex2exp,readxl,openxlsx, data.table
```

Contextualização

O presente trabalho tem como objetivo avaliar os dados do setor aquaviário, uma simples estatística descritiva tem um poder de nos apontar diversas informações. Neste sentido temos um data.frame com as seguintes colunas:

"IDAtracacao,NRAno,CDTrigrama,CDTUP,TPOperacao,TEsperaAtracacao,TEsperaInicioOp,TOperacao,TEsperaDesatracaca
IDAtracacao é o identificador da embarcação, *NRAno* são os anos que avaliaremos apesar de ter dados desde 2010 optamos por avaliar os ultimos 5 anos pois houve mudanças de sistemas que alimenta esse banco. As variáveis *CDTrigrama* é um código que identifica o porto organizado, *CDTUP* são terminais de uso privado, ou seja, uma empresa constroi um espaço de atividades portuária e pede autorização para Agência nacional de transporte aquaviário(ANTAQ) para realizar operações, *TPOperacoes* é o tipo de operação que cada embarcação realiza pode ser transporte de carga, transporte de passageiro, misto entre outras, e partir desta variáveis são os tempos que estas embarcações levam para cada etapa do processo como *TEsperaAtracacao* que é quanto tempo a embarcação fica atracada na costa (fundeio) *TEsperaInicioOp*, ou seja, quando a embarcação encosta no "berço" quanto tempo leva para iniciar as operações. Após o inicio da operção é avaliado o *TOperacao* que é o quanto tempo a embarcação leva para ser carregada, logo após tem o *TEsperaDesatracacao* que é o tempo que leva para a embarcação desatracar do berço e por fim os dois ultimos tempos que são *TAtracado* que é quanto tempo a embarcação ficou atracada e *TEstadia* que é o tempo total que a embarcação ficou parada naquele porto ou terminal autorizado.

Para importar os dados vamos utilizar a função fread do pacote data.table, pois o conjunto de dados possui aproximadamente 12 milhoes de observações, e esta função importa mais rápido do que funções convencionais como a import.csv ou read.table. Esta análise descritiva é muito importante nosso objetivo aqui é avaliar a qualidade dos dados, se há marcações dos tempos corretamente se tem muito valor descrepante oriundos de erro de digitação, se há tempos incoerentes por exemplo tempos zero, ou até mesmo avaliar a moda, ou seja, o agente responsável por anotar esses tempos se está repetindo anotações padrões.

```
DF <- fread('DF2.csv', header = TRUE,  sep = "," )
```

vamos avaliar a estrutura do data.frame com a função str nativa do R

```
str(DF)
```

```
## Classes 'data.table' and 'data.frame':   1116757 obs. of  11 variables:
##  $ IDAtracacao        : int  184298 241738 241739 241740 241741 241765 241774 241775 241776 241777 .
##  $ NRAno              : int  2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...
##  $ CDTrigrama         : chr  "REC" "" "" "" ...
##  $ CDTUP              : chr  "" "BRSE001" "BRSE001" "BRSE001" ...
##  $ TPOperacao         : int  5 2 3 3 2 2 3 2 2 2 ...
```

```
##  $ TEsperaAtracacao   : chr  "0,66666666674428" "0,083333333430346" "0,083333333255723" "0,083333334
##  $ TEsperaInicioOp    : chr  "" "" "" "" ...
##  $ TOperacao          : chr  "" "" "" "" ...
##  $ TEsperaDesatracacao: chr  "" "" "" "" ...
##  $ TAtracado          : chr  "13,333333333314" "13,249999999884" "22,5" "5,9166666665697" ...
##  $ TEstadia           : chr  "14,000000000058" "13,333333333314" "22,583333333256" "6" ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

É muito importante avaliar a estrutura dos dados como boa prática de programação note o software esta reconhecendo variáveis de tempos como caracteres na verdade não são elas deveriam ser numéricas para evitar de importar esse conjunto vamos abordar uma alternativa de transformar nossos tempos em numéricos. Os tempos aqui estudados estão em Horas.

```
# Aplicar a substituicao e conversao para as 6 ultimas colunas numericas
DF <- DF %>%
  mutate_at(vars(tail(names(DF), 6)), ~ as.numeric(gsub(",", ".", .)))

str(DF)
```

```
## Classes 'data.table' and 'data.frame':   1116757 obs. of  11 variables:
##  $ IDAtracacao        : int  184298 241738 241739 241740 241741 241765 241774 241775 241776 241777 .
##  $ NRAno              : int  2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...
##  $ CDTrigrama         : chr  "REC" "" "" "" ...
##  $ CDTUP              : chr  "" "BRSE001" "BRSE001" "BRSE001" ...
##  $ TPOperacao         : int  5 2 3 3 2 2 3 2 2 2 ...
##  $ TEsperaAtracacao   : num  0.6667 0.0833 0.0833 0.0833 0.0833 ...
##  $ TEsperaInicioOp    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ TOperacao          : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ TEsperaDesatracacao: num  NA NA NA NA NA NA NA NA NA NA ...
##  $ TAtracado          : num  13.33 13.25 22.5 5.92 0.4 ...
##  $ TEstadia           : num  14 13.333 22.583 6 0.483 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

A função acima necessita que tenha instalado a biblioteca tidyverse no R. Note que ele ja converteu e apareceu uns NA, nas análises é so utilizar funções com na.rm e realizar as análises normalmente.

Dentro da varivel tipo de operação temos 8 caracteristica porém ela esta enumerada de 1 a 8 temos um dicionário que indica o que é cada uma dela. O tipo 1 é moventação de carga e uma das mais importante(Uma análise posterior será realizada afunilando ainda mais os filtros pelo tipo das cargas) e o tipo 7 é a misto vamos avaliar esse dois perfis neste primeiro momento podemos atribuir todos os perfis com a seguinte função:

```
DF <- DF %>%
  mutate(TPOperacao = recode(TPOperacao,
                             `1` = "Movimentacao de Carga",
                             `2` = "Passageiro",
                             `3` = "Apoio",
                             `4` = "Marinha",
                             `5` = "Abastecimento",
                             `6` = "Reparo/Manutencao",
                             `7` = "Misto",
                             `8` = "Retirada de Residuos"))
```

Vamos guardar no obtjeto TP o tipo de operação movimentação de carga e misto podemos fazer isso usando a função abaixo note que no argumento da função filter eu peço para ele pegar a movimentação de carga ou (indicado pelo operador |) misto.

```r
TP <- filter(DF, TPOperacao  == "Movimentacao de Carga"|TPOperacao  == "Misto")
```

Após separar os tipos movimentação de carga e Misto vamos agrupar pelos anos de 2019 até 2024, poderia ter feito em um único passo, mas além da análise o objetivo desse R markdow e que iniciante possam aprender a utilizar ferramentas de tratamento de dados.

```r
dados<-TP %>%
  filter(CDTrigrama !="" | CDTUP !="",
         NRAno %in% c("2019", "2020", "2021", "2022", "2023", "2024")) %>% select(-IDAtracacao)
```

Na função acima temos algo importante as operações são feitas tanto no em portos organizados que é o CDTrigrama quanto em terminais então vamos pegar as operações neste e verificar so os anos de interesse já o ID da atracação podemos retirar da análise então o select está fazendo isso removendo essa coluna.

Após essa etapa vamos agrupar operações por ano pelos portos e pelos terminais autorizados e ja de imediato faremos as descritivas. Para este trabalho vamos avaliar média, moda e mediana. Para além disse vamos definir intervalo de percentis vamos avaliar o percentil 99 e 0.01 por cento, e posteriormente vamos avaliar se há tempos acima deste percentil e vamos tratar eles com outliers. Apesar do procedimento padrão do outlier ser via boxplot e tem uma fórmula específica para tratar os limites superior e inferior aqui descidimos ser menos criterioso e utilizar o percentil para obter uma margem maiores para valores outilers. Alem do mais optamos por avaliar somente aquelas operações com mais de 30 registros anuais. para calcular a moda não temos funções nativas implementadas então a função criada abaixo realiza o calculo da moda

```r
calc_moda <- function(v) {
  v <- v[!is.na(v)] # Remove NA
  if (length(v) == 0) return(NA_real_)
  uniq_v <- unique(v)
  uniq_v[which.max(tabulate(match(v, uniq_v)))]
}
```

```r
df2 <- dados %>%
  group_by(NRAno, CDTrigrama, CDTUP, TPOperacao) %>%
  summarise(across(
    .cols = c(1:6),
    .fns = list(
      mean = ~ if_else(n() >= 30, mean(.x, na.rm = TRUE), NA_real_),
      mediana = ~ if_else(n() >= 30, median(.x, na.rm = TRUE), NA_real_),
      moda = ~ if_else(n() >= 30, round(calc_moda(.x) * 24 * 60, 2), NA_real_),
      LI = ~ if_else(n() >= 30, quantile(.x, probs = 0.01, na.rm = TRUE), NA_real_),
      LS = ~ if_else(n() >= 30, quantile(.x, probs = 0.99, na.rm = TRUE), NA_real_)
    ),
    .names = "{col}_{fn}"
  ), .groups = 'drop')
```

com as funções acima ja conseguimos calcular toda a descritiva e esta salva no objeto df2 agora retiraremos as linhas que possui NA, mas atenção vamos colocar uma condição se a linha inteira contem NA ai sim retiramos caso contrario deixaremos o valor faltante indicado. Caso fosse realizar procedimentos de modelagem ai sim esses valores teriam outro tratamento como interpolação substituição por médias entre outros tratamento de NA, mas para descritiva podemo deixar em branco.

```r
df3 <- df2 %>%
  filter(if_any(6:34, ~ !is.na(.)))
```

Vamos visualizar agora o cabeçalho dos dados utilizando a função head do R

```r
head(df3)
```

```
## # A tibble: 6 x 34
```

```
##   NRAno CDTrigrama CDTUP TPOperacao TEsperaAtracacao_mean TEsperaAtracacao_med~1
##   <int> <chr>      <chr> <chr>                       <dbl>                   <dbl>
## 1  2019 ""         BRAL~ Movimenta~                53.3                    15.5
## 2  2019 ""         BRAM~ Movimenta~                 0.0167                  0.0167
## 3  2019 ""         BRAM~ Movimenta~                 0.542                   0.683
## 4  2019 ""         BRAM~ Movimenta~                52.8                     0.867
## 5  2019 ""         BRAM~ Movimenta~                 0.0167                  0.0167
## 6  2019 ""         BRAM~ Movimenta~                 0.0833                  0.0833
## # i abbreviated name: 1: TEsperaAtracacao_mediana
## # i 28 more variables: TEsperaAtracacao_moda <dbl>, TEsperaAtracacao_LI <dbl>,
## #   TEsperaAtracacao_LS <dbl>, TEsperaInicioOp_mean <dbl>,
## #   TEsperaInicioOp_mediana <dbl>, TEsperaInicioOp_moda <dbl>,
## #   TEsperaInicioOp_LI <dbl>, TEsperaInicioOp_LS <dbl>, TOperacao_mean <dbl>,
## #   TOperacao_mediana <dbl>, TOperacao_moda <dbl>, TOperacao_LI <dbl>,
## #   TOperacao_LS <dbl>, TEsperaDesatracacao_mean <dbl>, ...
```

Após a tabela pronta vamos exportar-la o restante desta análise foi feita na ferramenta click sense, por lá percebemos diversos problemas como por exemplo diversos agentes portuários estão marcando o mesmo valor (30 minutos para uma operação as vezes 25) e esta marcação não condiz com a realidade, imagine você em no seu trajeto de trabalho é a mesma distancia que percorre todos os dias nem por isso você gasta exatamente o mesmo tempo para chegada, logo o mesmo raciocínio se aplica aqui. Estes tempos também impacta na avaliação portuária, principalmente no indicador prancha média que não foi avaliado aqui mais esta diretamente ligada com o tempo de operação, esse tempo mede o desmpenho portuário no sentido de sua capacidade de carregamento em toneladas por hora.

Este é um projeto que está em andamento tem muita análise estatística que da pra fazer com estes dados visando melhoria nas atividades portuárias e deixo como trabalho futuro, para quem quiser contribuir vou deixar o arquivo com os dados.

```
write.xlsx(df3, file = "19-06-2024-outliers-anos-2019-a-2024-movimentacao-carga-e-misto.xlsx")
```