# Correlations

Eduardo Yuki Yada

## Imports

```
library(tidyverse)
library(yaml)
library(kableExtra)
library(ggcorrplot)
```

## Loading data

```
load('../dataset/processed_data.RData')
load('../dataset/processed_dictionary.RData')

columns_list <- yaml.load_file("./auxiliar/columns_list.yaml")

outcome_column <- params$outcome_column
threshold <- params$threshold
```

## Functions

```
niceFormatting = function(df, caption="", digits = 2, font_size = NULL){
  df %>%
    kbl(booktabs = T, longtable = T, caption = caption, digits = digits, format = "latex") %>%
    kable_styling(font_size = font_size,
                  latex_options = c("striped", "HOLD_position", "repeat_header"))
}
```

## Correlation

```
na_eligible_columns <- df %>%
  summarise(across(everything(), ~ mean(is.na(.)))) %>%
  select_if(function(.) last(.) < 0.8) %>%
  names

unique_eligible_columns <- df %>%
  summarise(across(everything(), ~ length(unique(.)))) %>%
  select_if(function(.) last(.) > 1) %>%
  names

pre_columns = df_names %>%
  filter(momento.aquisicao == 'Admissão t0') %>%
  .$variable.name

weird_columns <- c('dieta_parenteral', 'dieta_enteral')

eligible_columns <- intersect(na_eligible_columns,
                              unique_eligible_columns) %>%
```

```
  intersect(pre_columns)

eligible_columns <- setdiff(eligible_columns, weird_columns)

corr <- df %>%
  select(all_of(intersect(columns_list$numerical_columns,
                          eligible_columns))) %>%
  drop_na %>%
  cor %>%
  as.matrix
```

```
## Warning in cor(.): the standard deviation is zero
```

```
corr_table <- corr %>%
  as.data.frame %>%
  tibble::rownames_to_column(var = 'row') %>%
  tidyr::pivot_longer(-row, names_to = 'column', values_to = 'correlation') %>%
  filter(row < column)

rename_column <- function(df, column_name){
  variable.name <- 'variable.name'
  df <- df %>%
    left_join(df_names %>% select(variable.name, abbrev.field.label),
              by = setNames(variable.name, column_name)) %>%
    select(-all_of(column_name)) %>%
    rename(!!sym(column_name) := abbrev.field.label) %>%
    relocate(!!sym(column_name))
}

corr_table %>%
  filter(correlation > 0.9) %>%
  rename_column('row') %>%
  rename_column('column') %>%
  select(row, column, correlation) %>%
  niceFormatting(caption = "Pearson Correlation", font_size = 9)
```

Table 1: Pearson Correlation

| row | column | correlation |
|---|---|---|
| Idade no momento do primeiro procedimento | Idade no Procedimento 1 | 1.00 |
| Núm. de hospitalizações pré-procedimento | Número da Admissão T0 | 0.98 |
| Ano da admissão T0 | Ano do procedimento 1 | 1.00 |
| Antibióticos | Quantidade de antimicrobianos | 1.00 |
| Quantidade de procedimentos invasivos | Suporte cardiocirculatório | 0.97 |
| ECG | Quantidade de exames por métodos gráficos | 1.00 |
| Exames laboratoriais | Radiografias | 0.90 |
| Quantidade de exames de análises clínicas | Exames laboratoriais | 1.00 |
| Quantidade de exames de análises clínicas | Radiografias | 0.90 |
| Quantidade de exames de análises clínicas | Quantidade de exames diagnóstico por imagem | 0.93 |
| Quantidade de exames diagnóstico por imagem | Exames laboratoriais | 0.93 |
| Quantidade de exames diagnóstico por imagem | Radiografias | 0.98 |
| Quantidade de classes medicamentosas de ação cardiovascular | Quantidade de classes medicamentosas utilizadas | 0.91 |

# Hypothesis Tests

```
df_wilcox <- tibble()

for (variable in intersect(columns_list$numerical_columns,
                           eligible_columns)){
  if (mean(is.na(df[[variable]])) > 0.95) next
```

```
  x <- filter(df, !!sym(outcome_column) == 0)[[variable]]
  y <- filter(df, !!sym(outcome_column) == 1)[[variable]]

  test = tryCatch(wilcox.test(x, y, alternative = "two.sided", exact = FALSE),
                  error=function(cond) {
                    message("Can't calculate Wilcox test for variable ", variable)
                    message(cond)
                    return(list(statistic = NaN, p.value = NaN))
                  })

  df_wilcox = bind_rows(df_wilcox,
                        list("Variable" = variable,
                             "Statistic" = test$statistic,
                             "p-value" = test$p.value))
}

significant_num_cols <- df_wilcox %>%
  filter(`p-value` <= threshold) %>%
  select(Variable) %>%
  pull

df_wilcox <- df_wilcox %>%
  arrange(`p-value`) %>%
  mutate(`Statistic`  = round(`Statistic`, 3)) %>%
  rename_column('Variable')

df_wilcox %>%
  mutate(`p-value` = case_when(`p-value` == 1 ~ sprintf('> 0%s999', getOption("OutDec")),
                               `p-value` < 0.001 ~ sprintf('< 0%s001', getOption("OutDec")),
                               TRUE ~ as.character(round(`p-value`, 3)))) %>%
  niceFormatting(caption = "Mann-Whitney Test")
```

Table 2: Mann-Whitney Test

| Variable | Statistic | p-value |
|---|---|---|
| Quantidade de classes medicamentosas utilizadas | 2124087 | $< 0.001$ |
| Número da Admissão T0 | 5193100 | $< 0.001$ |
| Antagonista da Aldosterona | 3430622 | $< 0.001$ |
| Quantidade de classes medicamentosas de ação cardiovascular | 1796884 | $< 0.001$ |
| Insuficiência cardíaca | 3429206 | $< 0.001$ |
| Diuretico | 3264322 | $< 0.001$ |
| Núm. de hospitalizações pré-procedimento | 5400085 | $< 0.001$ |
| Quantidade de medicamentos de ação cardiovascular | 3202352 | $< 0.001$ |
| Antiarritmicos | 3752732 | $< 0.001$ |
| Exames laboratoriais | 3795664 | $< 0.001$ |
| Quantidade de exames de análises clínicas | 3797133 | $< 0.001$ |
| DVA | 3667521 | $< 0.001$ |
| Número de comorbidades | 5398826 | $< 0.001$ |
| Quantidade de exames por métodos gráficos | 3853718 | $< 0.001$ |
| ECG | 3860234 | $< 0.001$ |
| Quantidade de exames diagnóstico por imagem | 3863653 | $< 0.001$ |
| Ultrassom | 4455366 | $< 0.001$ |
| Equipe Multiprofissional | 3988348 | $< 0.001$ |
| Radiografias | 4001001 | $< 0.001$ |
| UTI durante a admissão T0 | 5984057 | $< 0.001$ |
| Anticoagulantes orais | 4157110 | $< 0.001$ |
| Digoxina | 4171725 | $< 0.001$ |
| Culturas | 4446598 | $< 0.001$ |

Table 2: Mann-Whitney Test *(continued)*

| Variable | Statistic | p-value |
|---|---|---|
| Psicofármacos | 3812408 | $< 0.001$ |
| Ecocardiograma | 4306753 | $< 0.001$ |
| Vasodilator | 3895966 | $< 0.001$ |
| Cintilografia | 4754255 | $< 0.001$ |
| Ressonancia magnetica | 4706668 | $< 0.001$ |
| Estatinas | 3944091 | $< 0.001$ |
| Tomografia | 4644480 | $< 0.001$ |
| Quantidade de antimicrobianos | 3901273 | $< 0.001$ |
| Antibióticos | 3905344 | $< 0.001$ |
| Quantidade de procedimentos invasivos | 4576315 | $< 0.001$ |
| Holter | 4680985 | $< 0.001$ |
| Insulina | 4268199 | $< 0.001$ |
| IECA/BRA | 3974307 | $< 0.001$ |
| Bomba de infusão contínua | 4334959 | $< 0.001$ |
| Citologias | 4946743 | $< 0.001$ |
| Cateterismo | 4756373 | $< 0.001$ |
| Diálise durante a admissão T0 | 6844335 | $< 0.001$ |
| Idade no momento do primeiro procedimento | 6315301 | $< 0.001$ |
| Idade no Procedimento 1 | 6315301 | $< 0.001$ |
| Cateter venoso central | 4909718 | $< 0.001$ |
| Outros procedimentos cirúrgicos | 4869813 | $< 0.001$ |
| Antiplaquetario EV | 4476558 | 0.001 |
| Quantidade de exames histopatológicos | 4957532 | 0.001 |
| Diárias no serviço de Emergência na admissão T0 | 2794775 | 0.002 |
| Intervenção coronária percutânea | 4971950 | 0.007 |
| Transfusão de hemoderivados | 4969854 | 0.01 |
| Eletrofisiologia | 4930069 | 0.012 |
| Flebografia | 4958410 | 0.012 |
| Angio TC | 4945575 | 0.015 |
| Tilt Test | 4997507 | 0.019 |
| Ano do procedimento 1 | 6604189 | 0.03 |
| Ano da admissão T0 | 6589326 | 0.035 |
| Exames endoscópicos | 4975078 | 0.037 |
| Angiografia | 5002120 | 0.047 |
| PET-CT | 4992544 | 0.049 |
| Angioplastia | 5005526 | 0.06 |
| Teste de esforço | 5050265 | 0.073 |
| Antifúngicos | 4483094 | 0.174 |
| Cardioversão/ Desfibrilação | 4455569 | 0.187 |
| Intervenção cardiovascular em laboratório de hemodinâmica | 5003422 | 0.236 |
| Aortografia | 5008739 | 0.243 |
| Suporte cardiocirculatório | 5007327 | 0.273 |
| Ventilação não invasiva | 5007375 | 0.275 |
| Anticonvulsivante | 4483403 | 0.278 |
| Antiviral | 4506403 | 0.28 |
| Polissonografia | 5012155 | 0.372 |
| Espirometria / Ergoespirometria | 5007987 | 0.394 |
| Trombolitico | 4522210 | 0.395 |
| Interconsulta médica | 4969448 | 0.409 |
| Hipoglicemiante | 4489960 | 0.414 |
| Antiretroviral | 4521440 | 0.442 |
| Arteriografia | 5014752 | 0.447 |
| Marca-passo temporário | 4456277 | 0.456 |

Table 2: Mann-Whitney Test *(continued)*

| Variable | Statistic | p-value |
|---|---|---|
| Cirurgia Toracica | 5011551 | 0.467 |
| Bloqueador do canal de calcio | 4502318 | 0.479 |
| Cirurgia Cardiovascular | 4995975 | 0.513 |
| Biopsias | 5024903 | 0.599 |
| Cavografia | 5025311 | 0.65 |
| Betabloqueador | 4502397 | 0.763 |
| Transplante cardíaco | 5020393 | 0.763 |
| Antihipertensivo | 4528064 | 0.763 |
| Número de procedimentos na admissão T0 | 6905961 | 0.78 |
| Instalação de CEC | 5012555 | 0.783 |
| Stent | 5018559 | 0.799 |
| Drenagem de tórax e punção pericárdica ou pleural | 5020804 | 0.823 |
| Traqueostomia | 5017574 | 0.924 |
| Angio RM | 5018599 | 0.962 |
| Antiplaquetario VO | 4517975 | NaN |
| Hormonio tireoidiano | 4517975 | NaN |
| Broncodiltador | 4517975 | NaN |

```
df_chisq <- tibble()

for (variable in intersect(columns_list$categorical_columns,
                           eligible_columns)){
  if (length(unique(df[[variable]])) > 1){
    test <- tryCatch(chisq.test(df[[outcome_column]],
                        df[[variable]] %>% replace_na('NA'), # counting NA as cat
                        simulate.p.value = TRUE),
                     error = function (cond) {
                        message("Can't calculate Chi Squared test for variable ", variable)
                        message(cond)
                        return(list(statistic = NaN, p.value = NaN))
                     })

    df_chisq <- bind_rows(df_chisq,
                          list("Variable" = variable,
                               "Statistic" = test$statistic,
                               "p-value" = test$p.value))
  }
}

significant_cat_cols <- df_chisq %>%
  filter(`p-value` <= threshold) %>%
  select(Variable) %>%
  pull

df_chisq <- df_chisq %>%
  arrange(`p-value`) %>%
  mutate(`Statistic`  = round(`Statistic`, 3)) %>%
  rename_column('Variable')

df_chisq %>%
  mutate(`p-value` = case_when(`p-value` == 1 ~ sprintf('> 0%s999', getOption("OutDec")),
                               `p-value` < 0.001 ~ sprintf('< 0%s001', getOption("OutDec")),
                               TRUE ~ as.character(round(`p-value`, 3)))) %>%
  niceFormatting(caption = "Chi-squared test")
```

Table 3: Chi-squared test

| Variable | Statistic | p-value |
|---|---|---|
| Sexo | 23.48 | < 0.001 |
| Escolaridade | 82.51 | < 0.001 |
| Doença cardíaca | 76.64 | < 0.001 |
| Doença cardíaca | 38.17 | < 0.001 |
| Classe funcional de IC | 119.31 | < 0.001 |
| Infarto do miocárdio prévio / Doença arterial coronariana | 31.68 | < 0.001 |
| Insuficiência cardíaca | 184.20 | < 0.001 |
| Fibrilação / flutter atrial | 21.20 | < 0.001 |
| Valvopatias/ Prótese valvares | 66.07 | < 0.001 |
| Diabetes melittus | 34.37 | < 0.001 |
| Insuficiência renal crônica | 65.83 | < 0.001 |
| Tipo de Procedimento 1 | 32.38 | < 0.001 |
| Tipo de Reoperação 1 | 34.85 | < 0.001 |
| Tipo de Procedimento 1 | 34.85 | < 0.001 |
| Tipo de Dispositivo ao final do procedimento 1 | 189.26 | < 0.001 |
| Tipo de Dispositivo ao final do procedimento 1 | 128.21 | < 0.001 |
| Admissão em até 180 dias antes da T0 | 122.02 | < 0.001 |
| Desfecho principal da admissão T0 | 15.84 | < 0.001 |
| Hemodiálise | 19.09 | < 0.001 |
| Acidente Vascular Cerebral/ Acidente isquêmico transitório prévios | 11.20 | 0.002 |
| Doença pulmonar obstrutiva crônica | 7.81 | 0.007 |
| Hipertensão arterial | 5.96 | 0.014 |
| Parada cardíaca prévia/ Taquicardia ventricular instável | 4.23 | 0.044 |
| Neoplasia em tratamento ou tratada recentemente | 3.22 | 0.102 |
| Raça | 10.71 | 0.123 |
| Estado de residência | 26.18 | 0.468 |
| Endocardite prévia | 0.10 | 0.853 |
| Transplante cardíaco prévio | 0.10 | > 0.999 |
| Óbito intraoperatório 1 | 0.42 | > 0.999 |

```
saveRDS(significant_cat_cols,
        file = sprintf("../EDA/auxiliar/significant_columns/categorical_%s.rds", outcome_column))

saveRDS(significant_num_cols,
        file = sprintf("../EDA/auxiliar/significant_columns/numerical_%s.rds", outcome_column))
```

```
## [1] 78
```

```
## [1] 23
```

```
## [1] 144
```

```
## [1] 60
```