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
Equipe Multiprofissional	275871.0	< 0.001
Culturas	331815.5	< 0.001
Ultrassom	354812.5	< 0.001
Angiografia	422937.0	< 0.001
Ecocardiograma	301060.5	< 0.001
Insulina	343993.5	< 0.001
Quantidade de exames diagnóstico por imagem	291931.5	< 0.001
UTI durante a admissão T0	440267.5	< 0.001
Tomografia	364321.0	< 0.001
Radiografias	307029.0	< 0.001
Holter	369189.0	< 0.001
Quantidade de classes medicamentosas utilizadas	185209.0	< 0.001
Psicofármacos	296937.5	< 0.001
Ventilação não invasiva	423376.0	< 0.001
Núm. de hospitalizações pré-procedimento	441027.5	< 0.001
Quantidade de exames por métodos gráficos	312927.0	< 0.001
ECG	314485.5	< 0.001
Citologias	418309.5	< 0.001
Exames laboratoriais	319671.0	< 0.001
Quantidade de exames de análises clínicas	319700.5	< 0.001
Número de comorbidades	430119.5	< 0.001
Número da Admissão T0	460534.0	< 0.001
Diuretico	308750.5	< 0.001

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

Variable	Statistic	p-value
Cateter venoso central	405946.5	< 0.001
Quantidade de classes medicamentosas de ação cardiovascular	156665.0	0.001
Idade no momento do primeiro procedimento	443070.0	0.002
Idade no Procedimento 1	443070.0	0.002
DVA	337382.5	0.002
Intervenção cardiovascular em laboratório de hemodinâmica	424123.5	0.002
Interconsulta médica	382917.5	0.003
Vasodilator	337982.0	0.005
Transfusão de hemoderivados	420135.5	0.006
Quantidade de procedimentos invasivos	380003.0	0.007
Antiplaquetario EV	394138.5	0.007
Quantidade de exames histopatológicos	420333.0	0.007
Quantidade de medicamentos de ação cardiovascular	328906.0	0.009
Bomba de infusão contínua	377918.5	0.01
Ressonancia magnetica	403426.5	0.014
Antagonista da Aldosterona	354384.5	0.015
Ano do procedimento 1	473118.0	0.016
Ano da admissão T0	473612.5	0.018
Cintilografia	411678.0	0.021
Antiarritmicos	359952.5	0.021
Antihipertensivo	383430.0	0.037
Antifúngicos	388844.0	0.044
Quantidade de antimicrobianos	349816.5	0.054
Antibióticos	351751.0	0.063
Flebografia	423028.5	0.079
Betabloqueador	378112.5	0.09
Angio TC	420782.5	0.097
Anticoagulantes orais	387776.5	0.141
Antiviral	399723.5	0.153
Diárias no serviço de Emergência na admissão T0	229421.5	0.164
Cateterismo	413144.5	0.174
PET-CT	430953.0	0.253
Cirurgia Cardiovascular	424060.5	0.259
Estatinas	379025.5	0.31
Número de procedimentos na admissão T0	558320.0	0.315
Eletrofisiologia	425655.5	0.347
Outros procedimentos cirúrgicos	423586.0	0.372
Teste de esforço	439593.0	0.42
Intervenção coronária percutânea	439164.0	0.444
Cavografia	432007.0	0.474
IECA/BRA	422164.0	0.521
Insuficiência cardíaca	390425.5	0.525
Biopsias	437514.0	0.565
Espirometria / Ergoespirometria	437217.0	0.593
Drenagem de tórax e punção pericárdica ou pleural	437184.0	0.596
Marca-passo temporário	392837.5	0.6
Cardioversão/ Desfibrilação	396864.0	0.625
Hipoglicemiante	409292.0	0.627
Suporte cardiocirculatório	436623.0	0.658
Diálise durante a admissão T0	567751.5	0.673
Angio RM	436458.0	0.68
Cirurgia Toracica	436425.0	0.684
Anticonvulsivante	400458.5	0.687

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

Variable	Statistic	p-value
Tilt Test	436359.0	0.694
Aortografia	436194.0	0.718
Transplante cardíaco	436062.0	0.74
Angioplastia	435930.0	0.764
Polissonografia	435930.0	0.764
Traqueostomia	435831.0	0.784
Digoxina	407317.5	0.79
Trombolitico	404657.5	0.811
Antiretroviral	404592.5	0.829
Arteriografia	435600.0	0.842
Instalação de CEC	434219.5	0.852
Exames endoscópicos	434448.0	0.884
Bloqueador do canal de calcio	405098.0	0.904
Stent	435369.0	0.944
Antiplaquetario VO	404300.0	NaN
Hormonio tireoidiano	404300.0	NaN
Broncodiltador	404300.0	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
Insuficiência renal crônica	30.23	< 0.001
Escolaridade	25.02	< 0.001
Tipo de Dispositivo ao final do procedimento 1	12.29	< 0.001
Admissão em até 180 dias antes da T0	17.93	0.001
Hipertensão arterial	8.97	0.002
Doença cardíaca	14.23	0.003
Diabetes melittus	11.68	0.003
Tipo de Procedimento 1	9.11	0.003
Tipo de Dispositivo ao final do procedimento 1	15.07	0.003
Insuficiência cardíaca	9.14	0.004
Valvopatias/ Prótese valvares	8.77	0.006
Infarto do miocárdio prévio / Doença arterial coronariana	7.03	0.016
Tipo de Procedimento 1	10.45	0.02
Tipo de Reoperação 1	10.45	0.021
Doença cardíaca	19.68	0.033
Classe funcional de IC	19.85	0.041
Fibrilação / flutter atrial	4.38	0.043
Sexo	3.21	0.072
Hemodiálise	8.40	0.077
Neoplasia em tratamento ou tratada recentemente	4.41	0.092
Estado de residência	50.54	0.155
Desfecho principal da admissão T0	1.17	0.416
Acidente Vascular Cerebral/ Acidente isquêmico transitório prévios	0.27	0.718
Parada cardíaca prévia/ Taquicardia ventricular instável	0.04	0.865
Raça	1.08	0.967
Transplante cardíaco prévio	0.06	> 0.999
Endocardite prévia	0.25	> 0.999
Doença pulmonar obstrutiva crônica	0.00	> 0.999
Óbito intraoperatório 1	0.03	> 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] 20
```

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

```
## [1] 50
```