Correlations - death_180days

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
print(threshold)
```

```
## [1] 0.1
```

```
df[columns_list$outcome_columns] <- lapply(df[columns_list$outcome_columns], as.character)
df[columns_list$outcome_columns] <- lapply(df[columns_list$outcome_columns], as.integer)
```

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

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
Quantidade de exames de análises clínicas	Exames laboratoriais	1.00
Quantidade de exames de análises clínicas	Quantidade de exames diagnóstico por imagem	0.93
Biopsias	Quantidade de exames histopatológicos	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	813650	< 0.001
Culturas	1496546	< 0.001
Número de comorbidades	1711231	< 0.001
Ultrassom	1588128	< 0.001
Diuretico	1204987	< 0.001
Equipe Multiprofissional	1330010	< 0.001
Antagonista da Aldosterona	1324598	< 0.001
Exames laboratoriais	1275023	< 0.001
Quantidade de exames de análises clínicas	1275065	< 0.001
Quantidade de medicamentos de ação cardiovascular	1180048	< 0.001
Quantidade de classes medicamentosas de ação cardiovascular	714860	< 0.001
DVA	1328935	< 0.001
Quantidade de exames diagnóstico por imagem	1306972	< 0.001
ECG	1338140	< 0.001
Quantidade de exames por métodos gráficos	1338734	< 0.001
Radiografias	1372711	< 0.001
Insuficiência cardíaca	1367960	< 0.001
Tomografia	1637705	< 0.001
Vasodilator	1357169	< 0.001
Antiarritmicos	1429158	< 0.001

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

Variable	Statistic	p-value
Insulina	1538808	< 0.001
Ecocardiograma	1475620	< 0.001
Número da Admissão T0	2070530	< 0.001
Citologias	1840980	< 0.001
UTI durante a admissão T0	2133314	< 0.001
Núm. de hospitalizações pré-procedimento	2097110	< 0.001
Anticoagulantes orais	1584896	< 0.001
Diálise durante a admissão T0	2511317	< 0.001
Psicofármacos	1393170	< 0.001
Estatinas	1447921	< 0.001
Cintilografia	1779155	< 0.001
Ano do procedimento 1	2091440	< 0.001
Ano da admissão T0	2082784	< 0.001
Quantidade de exames histopatológicos	1842649	< 0.001
Idade no momento do primeiro procedimento	2113999	< 0.001
Idade no Procedimento 1	2113999	< 0.001
Antiplaquetario EV	1699338	< 0.001
Ressonancia magnetica	1758143	< 0.001
Interconsulta médica	1718058	< 0.001
Holter	1737130	< 0.001
Quantidade de antimicrobianos	1452036	< 0.001
Antibióticos	1454193	< 0.001
Quantidade de procedimentos invasivos	1701096	< 0.001
Transfusão de hemoderivados	1856042	< 0.001
Cateter venoso central	1837468	< 0.001
Cateterismo	1781807	< 0.001
Diárias no serviço de Emergência na admissão T0	1098082	< 0.001
Aortografia	1881387	< 0.001
Antifúngicos	1696812	0.002
Intervenção coronária percutânea	1867916	0.004
Suporte cardiocirculatório	1882274	0.005
Ventilação não invasiva	1882291	0.005
Bomba de infusão contínua	1684872	0.008
Outros procedimentos cirúrgicos	1830565	0.011
Digoxina	1685370	0.021
IECA/BRA	1607206	0.021
Arteriografia	1891894	0.034
Angiografia	1888209	0.065
Teste de esforço	1916587	0.081
Exames endoscópicos	1878285	0.09
Tilt Test	1888954	0.12
Betabloqueador	1690944	0.129
Anticonvulsivante	1712077	0.148
Flebografia	1878190	0.183
Cavografia	1885159	0.192
Polissonografia	1893236	0.289
Antihipertensivo	1718643	0.32
Angioplastia	1893384	0.321
Antiviral	1732967	0.324
Hipoglicemiante	1719112	0.35
Angio RM	1902283	0.379
Drenagem de tórax e punção pericárdica ou pleural	1891619	0.398
Eletrofisiologia	1879625	0.404

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

Variable	Statistic	p-value
Transplante cardíaco	1893979	0.448
Traqueostomia	1899303	0.581
PET-CT	1893277	0.587
Biopsias	1893288	0.588
Cirurgia Toracica	1894715	0.593
Intervenção cardiovascular em laboratório de hemodinâmica	1893423	0.603
Trombolitico	1740145	0.66
Antiretroviral	1740145	0.66
Angio TC	1889915	0.676
Cirurgia Cardiovascular	1891718	0.781
Espirometria / Ergoespirometria	1899344	0.788
Instalação de CEC	1900320	0.804
Marca-passo temporário	1718719	0.827
Número de procedimentos na admissão T0	2549579	0.87
Cardioversão/ Desfibrilação	1721262	0.984
Bloqueador do canal de calcio	1738924	0.996
Antiplaquetario VO	1738985	NaN
Hormonio tireoidiano	1738985	NaN
Broncodiltador	1738985	NaN
Stent	1897366	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
Escolaridade	34.73	< 0.001
Doença cardíaca	27.59	< 0.001
Classe funcional de IC	94.74	< 0.001
Hipertensão arterial	32.23	< 0.001
Infarto do miocárdio prévio / Doença arterial coronariana	40.04	< 0.001
Insuficiência cardíaca	69.69	< 0.001
Fibrilação / flutter atrial	24.89	< 0.001
Valvopatias/ Prótese valvares	41.55	< 0.001
Diabetes melittus	61.75	< 0.001
Insuficiência renal crônica	78.85	< 0.001
Acidente Vascular Cerebral/ Acidente isquêmico transitório prévios	24.44	< 0.001
Tipo de Procedimento 1	22.78	< 0.001
Tipo de Procedimento 1	24.27	< 0.001
Tipo de Dispositivo ao final do procedimento 1	77.13	< 0.001
Tipo de Dispositivo ao final do procedimento 1	72.44	< 0.001
Admissão em até 180 dias antes da T0	55.25	< 0.001
Hemodiálise	54.47	< 0.001
Doença cardíaca	32.54	0.001
Tipo de Reoperação 1	24.27	0.002
Sexo	9.00	0.003
Doença pulmonar obstrutiva crônica	7.13	0.011
Neoplasia em tratamento ou tratada recentemente	9.34	0.014
Parada cardíaca prévia/ Taquicardia ventricular instável	3.79	0.063
Estado de residência	27.20	0.394
Raça	4.93	0.441
Endocardite prévia	0.23	0.764
Transplante cardíaco prévio	0.26	> 0.999

```
dir.create(file.path("./auxiliar/significant_columns/"), showWarnings = FALSE)

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

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

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

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

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

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