Correlations - readmission_60d

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
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	2120972	< 0.001
Quantidade de exames diagnóstico por imagem	3362978	< 0.001
Número da Admissão T0	5478487	< 0.001
Radiografias	3492325	< 0.001
Quantidade de medicamentos de ação cardiovascular	3084421	< 0.001
Quantidade de classes medicamentosas de ação cardiovascular	1819778	< 0.001
UTI durante a admissão T0	5669852	< 0.001
Quantidade de exames por métodos gráficos	3563561	< 0.001
Equipe Multiprofissional	3685459	< 0.001
ECG	3586949	< 0.001
Ecocardiograma	3781160	< 0.001
Ultrassom	4186558	< 0.001
Antiarritmicos	3517318	< 0.001
DVA	3436291	< 0.001
Exames laboratoriais	3670531	< 0.001
Quantidade de exames de análises clínicas	3670696	< 0.001
Diuretico	3309420	< 0.001
Antagonista da Aldosterona	3535375	< 0.001
Antifúngicos	3992410	< 0.001
Núm. de hospitalizações pré-procedimento	5769627	< 0.001
Quantidade de procedimentos invasivos	4129268	< 0.001
Culturas	4190934	< 0.001
Biopsias	4614213	< 0.001

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

Variable	Statistic	p-value
Quantidade de exames histopatológicos	4582131	< 0.001
Transplante cardíaco	4658038	< 0.001
Insuficiência cardíaca	3632772	< 0.001
Suporte cardiocirculatório	4648428	< 0.001
Exames endoscópicos	4577831	< 0.001
Vasodilator	3614722	< 0.001
Psicofármacos	3551656	< 0.001
Ressonancia magnetica	4395230	< 0.001
Cateterismo	4319608	< 0.001
Anticoagulantes orais	3919554	< 0.001
Antiviral	4113880	< 0.001
Número de comorbidades	5827715	< 0.001
Cateter venoso central	4529535	< 0.001
Tomografia	4364866	< 0.001
Holter	4371403	< 0.001
Digoxina	3958452	< 0.001
Cintilografia	4508877	< 0.001
Quantidade de antimicrobianos	3631497	< 0.001
Antibióticos	3635617	< 0.001
Betabloqueador	3886306	< 0.001
Diárias no serviço de Emergência na admissão T0	2428645	< 0.001
Bloqueador do canal de calcio	4069195	< 0.001
Estatinas	3752955	< 0.001
Diálise durante a admissão T0	6686844	< 0.001
Outros procedimentos cirúrgicos	4507898	< 0.001
Instalação de CEC	4623338	< 0.001
Eletrofisiologia	4553317	< 0.001
Bomba de infusão contínua	3999605	< 0.001
IECA/BRA	3764359	< 0.001
Citologias	4659576	< 0.001
Anticonvulsivante	4053610	< 0.001
Transfusão de hemoderivados	4642312	< 0.001
Insulina	4012238	< 0.001
Angio TC	4620171	< 0.001
Angio RM	4688730	0.003
Cirurgia Toracica	4692521	0.004
Idade no momento do primeiro procedimento	7098240	0.006
Idade no Procedimento 1	7098240	0.006
Antiplaquetario EV	4152606	0.006
Intervenção coronária percutânea	4672180	0.007
Espirometria / Ergoespirometria	4684920	0.009
Arteriografia	4704781	0.011
PET-CT	4686048	0.014
Tilt Test	4694466	0.016
Teste de esforço	4673478	0.016
Cardioversão/ Desfibrilação	4079249	0.023
Angioplastia	4702114	0.04
Interconsulta médica	4613458	0.071
Antihipertensivo	4136870	0.134
Ano da admissão T0	6896117	0.142
Flebografia	4683024	0.157
Ano do procedimento 1	6917498	0.162
Marca-passo temporário	4078806	0.164

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

Variable	Statistic	p-value
Ventilação não invasiva	4726659	0.17
Intervenção cardiovascular em laboratório de hemodinâmica	4699509	0.192
Polissonografia	4708239	0.262
Antiretroviral	4181114	0.428
Trombolitico	4187254	0.479
Número de procedimentos na admissão T0	6715335	0.537
Drenagem de tórax e punção pericárdica ou pleural	4720916	0.589
Aortografia	4711323	0.613
Angiografia	4712090	0.698
Traqueostomia	4713607	0.784
Cirurgia Cardiovascular	4722985	0.806
Cavografia	4711786	0.821
Hipoglicemiante	4182640	0.958
Antiplaquetario VO	4184358	NaN
Hormonio tireoidiano	4184358	NaN
Broncodiltador	4184358	NaN
Stent	4715124	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	27.42	< 0.001
Infarto do miocárdio prévio / Doença arterial coronariana	17.33	< 0.001
Insuficiência cardíaca	69.57	< 0.001
Tipo de Procedimento 1	81.71	< 0.001
Tipo de Reoperação 1	95.18	< 0.001
Tipo de Procedimento 1	95.18	< 0.001
Tipo de Dispositivo ao final do procedimento 1	91.08	< 0.001
Tipo de Dispositivo ao final do procedimento 1	41.49	< 0.001
Admissão em até 180 dias antes da T0	105.55	< 0.001
Doença cardíaca	16.29	< 0.001
Doença cardíaca	29.32	0.002
Classe funcional de IC	25.98	0.002
Fibrilação / flutter atrial	9.38	0.004
Transplante cardíaco prévio	16.87	0.004
Diabetes melittus	7.66	0.005
Parada cardíaca prévia/ Taquicardia ventricular instável	6.50	0.008
Valvopatias/ Prótese valvares	6.38	0.013
Hemodiálise	8.23	0.017
Endocardite prévia	2.66	0.132
Estado de residência	36.60	0.153
Insuficiência renal crônica	2.17	0.159
Neoplasia em tratamento ou tratada recentemente	1.03	0.32
Doença pulmonar obstrutiva crônica	0.67	0.462
Acidente Vascular Cerebral/ Acidente isquêmico transitório prévios	0.42	0.565
Hipertensão arterial	0.29	0.596
Raça	3.08	0.736
Sexo	0.02	0.919

```
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] 18
```

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

```
## [1] 71
```