Correlations - death_2year

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	1765579	< 0.001
Antagonista da Aldosterona	2818585	< 0.001
Número da Admissão T0	4337076	< 0.001
Quantidade de classes medicamentosas de ação cardiovascular	1490230	< 0.001
Insuficiência cardíaca	2809742	< 0.001
Diuretico	2632846	< 0.001
Quantidade de medicamentos de ação cardiovascular	2583233	< 0.001
DVA	2915914	< 0.001
Exames laboratoriais	2956961	< 0.001
Quantidade de exames de análises clínicas	2957943	< 0.001
Ultrassom	3584055	< 0.001
Quantidade de exames diagnóstico por imagem	2995273	< 0.001
Núm. de hospitalizações pré-procedimento	4485472	< 0.001
Número de comorbidades	4271006	< 0.001
Equipe Multiprofissional	3129297	< 0.001
Quantidade de exames por métodos gráficos	3036377	< 0.001
ECG	3037520	< 0.001
Antiarritmicos	3093068	< 0.001
Culturas	3519782	< 0.001
Radiografias	3133972	< 0.001

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

Variable	Statistic	p-value
UTI durante a admissão T0	4799814	< 0.001
Anticoagulantes orais	3423784	< 0.001
Ecocardiograma	3426193	< 0.001
Tomografia	3722006	< 0.001
Psicofármacos	3090389	< 0.001
Vasodilator	3158482	< 0.001
Cintilografia	3877875	< 0.001
Digoxina	3494683	< 0.001
Quantidade de antimicrobianos	3133525	< 0.001
Citologias	4051513	< 0.001
Antibióticos	3139336	< 0.001
Estatinas	3211433	< 0.001
Diálise durante a admissão T0	5612258	< 0.001
Ressonancia magnetica	3865829	< 0.001
Insulina	3495626	< 0.001
Quantidade de procedimentos invasivos	3735937	< 0.001
Holter	3835694	< 0.001
Bomba de infusão contínua	3590330	< 0.001
Cateterismo	3886397	< 0.001
IECA/BRA	3342299	< 0.001
Idade no momento do primeiro procedimento	5105899	< 0.001
Idade no Procedimento 1	5105899	< 0.001
Quantidade de exames histopatológicos	4063264	< 0.001
Antiplaquetario EV	3690094	< 0.001
Cateter venoso central	4033691	< 0.001
Ano do procedimento 1	5238938	< 0.001
Ano da admissão T0	5224401	< 0.001
Antifúngicos	3671557	0.001
Intervenção coronária percutânea	4081155	0.001
Exames endoscópicos	4076864	0.002
Outros procedimentos cirúrgicos	4010346	0.002
Transfusão de hemoderivados	4083693	0.003
Diárias no serviço de Emergência na admissão T0	2363355	0.007
Flebografia	4073539	0.008
Angioplastia	4115708	0.019
Teste de esforço	4166149	0.024
Angio TC	4074914	0.038
Suporte cardiocirculatório	4113508	0.041
Ventilação não invasiva	4113538	0.041
Tilt Test	4113868	0.049
Antiviral	3719906	0.06
Interconsulta médica	4039194	0.086
PET-CT	4110529	0.086
Aortografia	4118037	0.101
Cirurgia Toracica	4118698	0.137
Angiografia	4118714	0.138
Eletrofisiologia	4085758	0.162
Polissonografia	4121891	0.181
Anticonvulsivante	3702384	0.208
Arteriografia	4125399	0.272
Intervenção cardiovascular em laboratório de hemodinâmica	4117411	0.277
Traqueostomia	4133918	0.402
Trombolitico	3739329	0.507

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

Variable	Statistic	p-value
Antiretroviral	3739329	0.507
Espirometria / Ergoespirometria	4122554	0.517
Biopsias	4136598	0.527
Angio RM	4134074	0.585
Cirurgia Cardiovascular	4115258	0.632
Instalação de CEC	4121375	0.64
Antihipertensivo	3723745	0.664
Betabloqueador	3718816	0.699
Número de procedimentos na admissão T0	5658309	0.729
Drenagem de tórax e punção pericárdica ou pleural	4126362	0.749
Bloqueador do canal de calcio	3743073	0.749
Cardioversão/ Desfibrilação	3700778	0.859
Hipoglicemiante	3733034	0.905
Cavografia	4128058	0.912
Transplante cardíaco	4130074	0.94
Marca-passo temporário	3697771	0.956
Antiplaquetario VO	3736761	NaN
Hormonio tireoidiano	3736761	NaN
Broncodiltador	3736761	NaN
Stent	4129576	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	18.59	< 0.001
Escolaridade	63.65	< 0.001
Doença cardíaca	60.19	< 0.001
Doença cardíaca	34.92	< 0.001
Classe funcional de IC	119.59	< 0.001
Hipertensão arterial	12.99	< 0.001
Infarto do miocárdio prévio / Doença arterial coronariana	30.79	< 0.001
Insuficiência cardíaca	164.56	< 0.001
Fibrilação / flutter atrial	28.07	< 0.001
Valvopatias/ Prótese valvares	68.75	< 0.001
Diabetes melittus	38.93	< 0.001
Insuficiência renal crônica	78.97	< 0.001
Hemodiálise	29.94	< 0.001
Acidente Vascular Cerebral/ Acidente isquêmico transitório prévios	13.28	< 0.001
Tipo de Procedimento 1	38.15	< 0.001
Tipo de Reoperação 1	41.70	< 0.001
Tipo de Procedimento 1	41.70	< 0.001
Tipo de Dispositivo ao final do procedimento 1	169.58	< 0.001
Tipo de Dispositivo ao final do procedimento 1	124.16	< 0.001
Admissão em até 180 dias antes da T0	95.09	< 0.001
Doença pulmonar obstrutiva crônica	10.02	0.001
Parada cardíaca prévia/ Taquicardia ventricular instável	5.45	0.021
Neoplasia em tratamento ou tratada recentemente	2.52	0.113
Raça	11.19	0.122
Estado de residência	29.47	0.339
Transplante cardíaco prévio	0.60	0.665
Endocardite prévia	0.02	> 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] 22
```

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

```
## [1] 63
```