Correlations - death_1year

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	1310355	< 0.001
Número de comorbidades	2694469	< 0.001
Diuretico	1860245	< 0.001
Ultrassom	2466201	< 0.001
Quantidade de classes medicamentosas de ação cardiovascular	1087647	< 0.001
Antagonista da Aldosterona	2068043	< 0.001
Exames laboratoriais	2014378	< 0.001
Quantidade de exames de análises clínicas	2014909	< 0.001
Equipe Multiprofissional	2102404	< 0.001
DVA	2047891	< 0.001
Culturas	2394866	< 0.001
Quantidade de exames diagnóstico por imagem	2033993	< 0.001
Quantidade de medicamentos de ação cardiovascular	1859987	< 0.001
Insuficiência cardíaca	2060259	< 0.001
ECG	2075347	< 0.001
Quantidade de exames por métodos gráficos	2078633	< 0.001
Radiografias	2125051	< 0.001
Número da Admissão T0	3177315	< 0.001
Antiarritmicos	2231242	< 0.001
Tomografia	2556040	< 0.001

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

Variable	Statistic	p-value
Núm. de hospitalizações pré-procedimento	3246222	< 0.001
Ecocardiograma	2357373	< 0.001
UTI durante a admissão T0	3349810	< 0.001
Citologias	2844288	< 0.001
Anticoagulantes orais	2454060	< 0.001
Vasodilator	2218700	< 0.001
Diálise durante a admissão T0	3892624	< 0.001
Psicofármacos	2189857	< 0.001
Cintilografia	2745807	< 0.001
Insulina	2463913	< 0.001
Quantidade de antimicrobianos	2231678	< 0.001
Antibióticos	2232209	< 0.001
Quantidade de exames histopatológicos	2850533	< 0.001
Ano do procedimento 1	3379070	< 0.001
Holter	2700613	< 0.001
Ano da admissão T0	3364789	< 0.001
Ressonancia magnetica	2739513	< 0.001
Quantidade de procedimentos invasivos	2642383	< 0.001
Idade no momento do primeiro procedimento	3401352	< 0.001
Idade no Procedimento 1	3401352	< 0.001
Estatinas	2340098	< 0.001
Digoxina	2524166	< 0.001
Bomba de infusão contínua	2575151	< 0.001
Cateter venoso central	2841158	< 0.001
Antiplaquetario EV	2639424	< 0.001
Antifúngicos	2612605	< 0.001
Cateterismo	2780256	< 0.001
IECA/BRA	2437997	< 0.001
Ventilação não invasiva	2902744	0.004
Exames endoscópicos	2881564	0.004
Outros procedimentos cirúrgicos	2828105	0.004
Interconsulta médica	2795885	0.004
Transfusão de hemoderivados	2890189	0.015
Aortografia	2907852	0.018
Teste de esforço	2951820	0.028
Cirurgia Toracica	2908310	0.028
Angiografia	2908326	0.028
Intervenção coronária percutânea	2894976	0.033
Diárias no serviço de Emergência na admissão T0	1749002	0.045
PET-CT	2903205	0.046
Antiviral	2658992	0.05
Suporte cardiocirculatório	2909241	0.057
Flebografia	2888784	0.064
Tilt Test	2909489	0.066
Angio TC	2884272	0.09
Arteriografia	2916940	0.126
Antihipertensivo	2644722	0.251
Anticonvulsivante	2649409	0.29
Drenagem de tórax e punção pericárdica ou pleural	2913640	0.332
Cavografia	2910870	0.345
Traqueostomia	2924850	0.488
Espirometria / Ergoespirometria	2915535	0.49
Hipoglicemiante	2655628	0.491

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

Variable	Statistic	p-value
Betabloqueador	2648332	0.516
Polissonografia	2919033	0.563
Trombolitico	2675614	0.58
Antiretroviral	2675614	0.58
Angioplastia	2919265	0.607
Número de procedimentos na admissão T0	3932480	0.641
Eletrofisiologia	2912647	0.728
Cardioversão/ Desfibrilação	2646914	0.742
Cirurgia Cardiovascular	2914319	0.765
Transplante cardíaco	2920194	0.768
Instalação de CEC	2924360	0.864
Angio RM	2922976	0.869
Intervenção cardiovascular em laboratório de hemodinâmica	2922978	0.903
Biopsias	2922759	0.921
Bloqueador do canal de calcio	2672357	0.931
Marca-passo temporário	2649768	0.946
Antiplaquetario VO	2673806	NaN
Hormonio tireoidiano	2673806	NaN
Broncodiltador	2673806	NaN
Stent	2921828	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	35.31	< 0.001
Doença cardíaca	49.16	< 0.001
Doença cardíaca	37.31	< 0.001
Classe funcional de IC	120.36	< 0.001
Hipertensão arterial	34.46	< 0.001
Infarto do miocárdio prévio / Doença arterial coronariana	39.80	< 0.001
Insuficiência cardíaca	119.18	< 0.001
Fibrilação / flutter atrial	29.98	< 0.001
Valvopatias/ Prótese valvares	81.55	< 0.001
Diabetes melittus	60.86	< 0.001
Insuficiência renal crônica	96.38	< 0.001
Doença pulmonar obstrutiva crônica	15.05	< 0.001
Tipo de Procedimento 1	28.57	< 0.001
Tipo de Reoperação 1	32.13	< 0.001
Tipo de Procedimento 1	32.13	< 0.001
Tipo de Dispositivo ao final do procedimento 1	112.03	< 0.001
Tipo de Dispositivo ao final do procedimento 1	94.17	< 0.001
Admissão em até 180 dias antes da T0	50.36	< 0.001
Sexo	16.44	< 0.001
Hemodiálise	48.03	< 0.001
Acidente Vascular Cerebral/ Acidente isquêmico transitório prévios	13.98	0.003
Parada cardíaca prévia/ Taquicardia ventricular instável	6.78	0.012
Neoplasia em tratamento ou tratada recentemente	5.18	0.035
Raça	6.36	0.322
Endocardite prévia	0.44	0.614
Estado de residência	20.58	0.662
Transplante cardíaco prévio	0.41	> 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] 65
```