Exploração dos Dados

Bibliotecas

```
library(tidyverse)
library(data.table)
library(Hmisc)
library(knitr)
library(kableExtra)
library(ggcorrplot)
library(ggdendro)
library(dendextend)
library(biotools)
library(ggfortify)
library(factoextra)
library(yaml)
library(corrplot)
select = dplyr::select
```

Carregando os dados

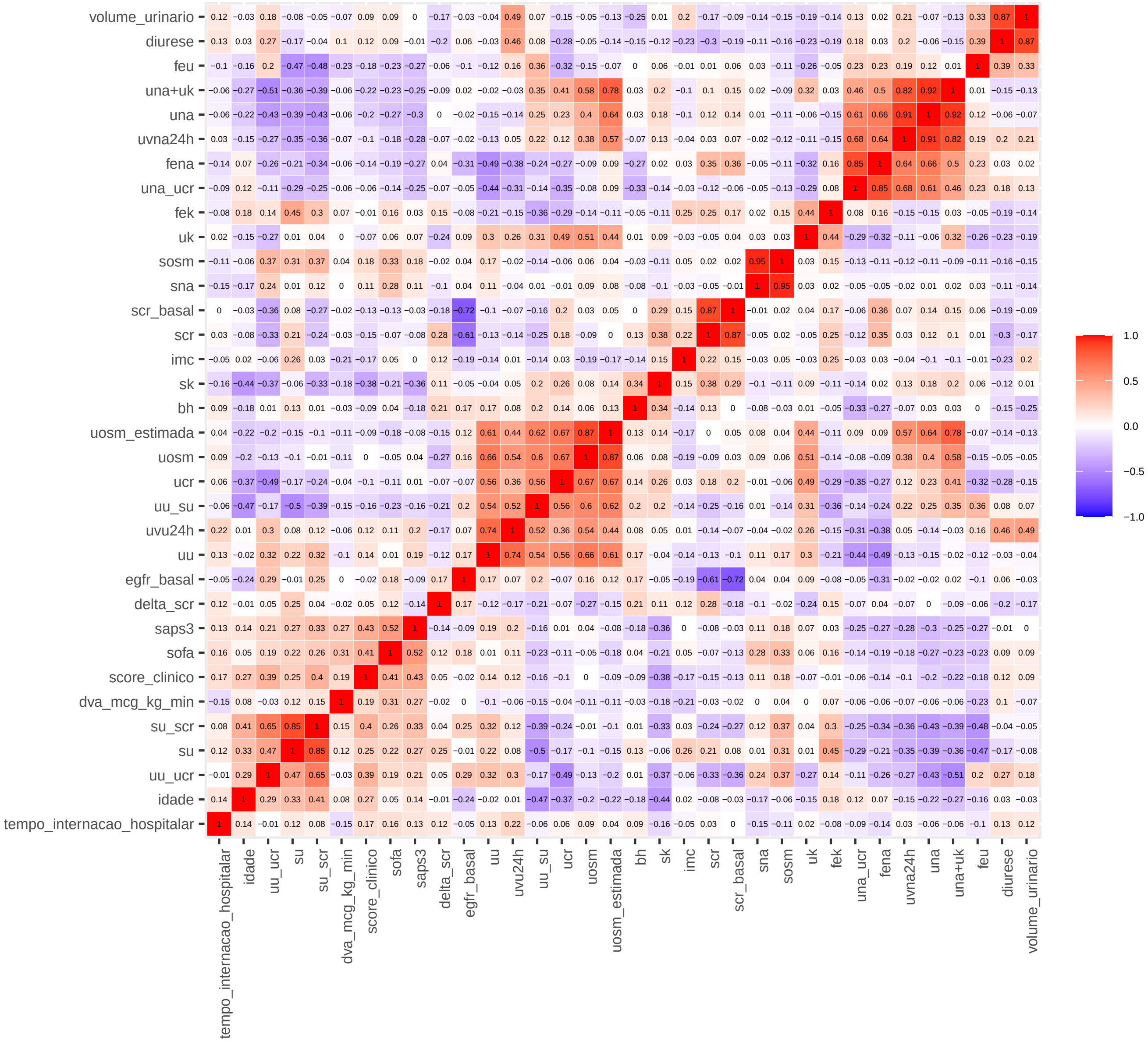
```
constants = yaml.load_file("constants.yaml")

categorical_columns = constants$categorical_columns

daily_columns = constants$daily_columns

numerical_columns = constants$numerical_columns

solutos_columns = constants$solutos_columns

all_columns = c(numerical_columns, categorical_columns)

original_columns = constants$original_columns

derived_columns = setdiff(daily_columns, original_columns)

camila_numerical_columns = constants$camila_columns
```

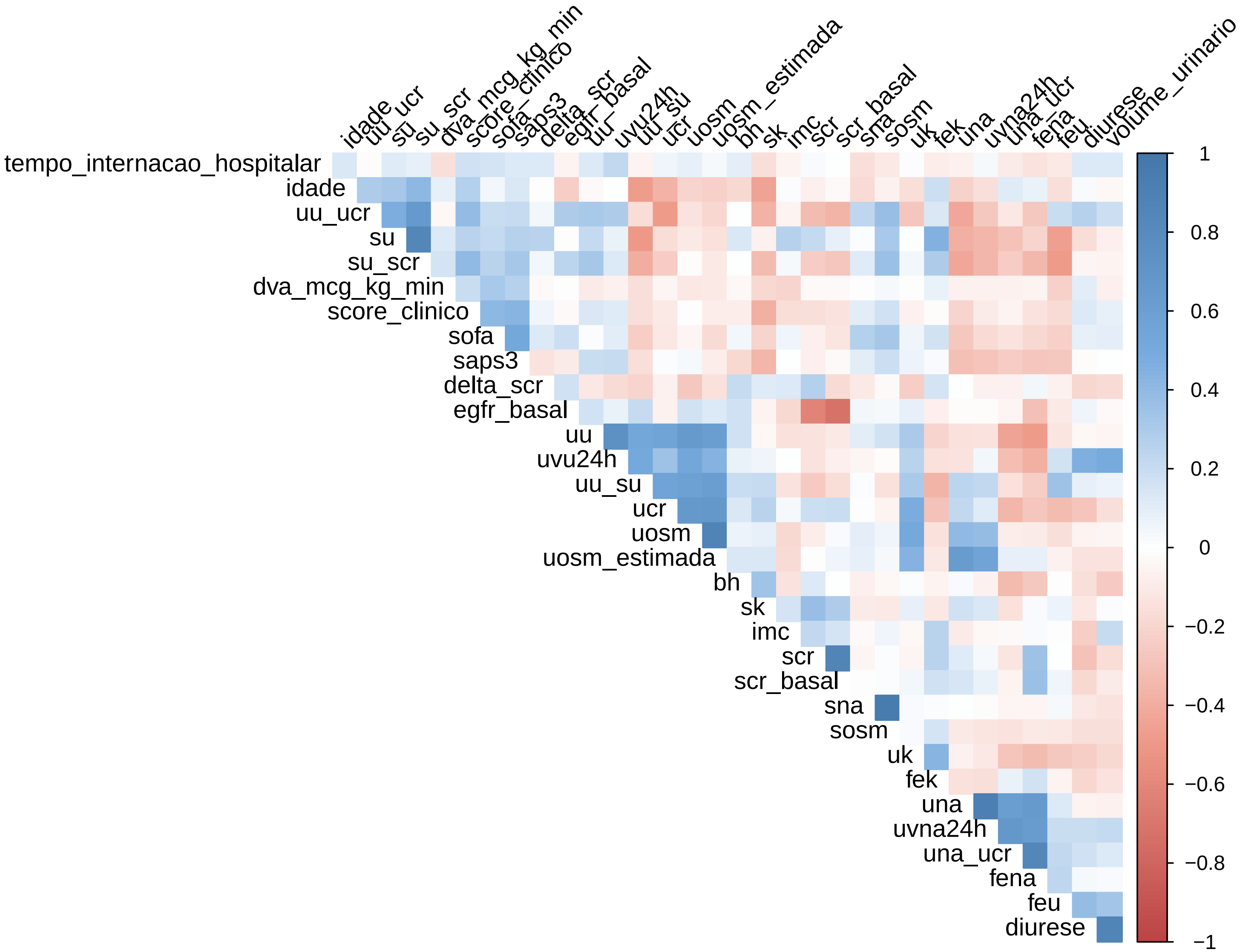
```
df = readRDS(’./data/dados_processados.rds’) %>%
    mutate_at(categorical_columns, list(~factor(.)))

df_names = readxl::read_excel(’./data/Nomes das variaveis.xlsx’) %>%
  mutate(variavel = tolower(variavel),
         nome = coalesce(nome, variavel)) %>%
  select(nome, variavel)
```

Funções

```
flattenCorrMatrix <- function(cormat, pmat) {
  ut <- upper.tri(cormat)
  data.frame(
    row = rownames(cormat)[row(cormat)[ut]],
    column = rownames(cormat)[col(cormat)[ut]],
    rho = (cormat)[ut]
    )
}

highlyCorrelated = function(df, threshold = 0.8){
  df_cor <- df %>%
    as.matrix() %>%
    rcorr(type = "pearson")

  df_flatten = flattenCorrMatrix(df_cor$r, df_cor$P) %>%
    mutate(abs_rho = abs(rho))

  return(df_flatten %>% filter(abs_rho > threshold))
}

highlyCorrelatedByGroup = function(df, group_col, threshold=0.8){
  group_levels = levels(df[[group_col]])
  concat_df = tibble()

  for (level in group_levels){
    corr_df = df %>%
      filter(!!sym(col) == level) %>%
      select(-all_of(group_col)) %>%
      highlyCorrelated(threshold = threshold) %>%
      mutate(group = col,
             group_level = level,
             size = dim(df %>% filter(!!sym(col) == level))[1])

    concat_df = bind_rows(concat_df, corr_df)
  }
  return(concat_df)
}

niceFormatting = function(df, caption=""){
  df %>%
    kbl(booktabs = T, longtable = T, caption = caption, digits = 2) %>%
    kable_styling(latex_options = c("striped", "HOLD_position", "repeat_header"))
}

rename_matrix = function(df){
  rownames(df) = tibble(variavel = cbind(rownames(df))) %>%
    inner_join(df_names, by='variavel') %>%
    .$nome

  colnames(df) = tibble(variavel = cbind(colnames(df))) %>%
    inner_join(df_names, by='variavel') %>%
    .$nome

  return(df)
}
```

Correlação


volume_urinario
diurese
feu
una+uk
una
uvna24h
fena
una_ucr
fek
uk
sosm
sna
scr_basal
scr
imc
sk
bh
uosm_estimada
uosm
ucr
uu_su
uvu24h
uu
egfr_basal
delta_scr
saps3
sofa
score_clinico
dva_mcg_kg_min
su_scr
su
uu_ucr
idade
tempo_internacao_hospitalar
tempo_internacao_hospitalar
idade
uu_ucr
su
su_scr
dva_mcg_kg_min
score_clinico
sofa
saps3
delta_scr
egfr_basal
uu
uvu24h
uu_su
ucr
uosm
uosm_estimada
bh
sk
imc
scr
scr_basal
sna
sosm
uk
fek
una_ucr
fena
uvna24h
una
una+uk
feu
diurese
volume_urinario
1.0
0.5
0.0
−0.5
−1.0


idade
uu_ucr
su
su_scr
dva_mcg_kg_min
score_clinico
sofa
saps3
delta_scr
egfr_basal
uu
uvu24h
uu_su
ucr
uosm
uosm_estimada
bh
sk
imc
scr
scr_basal
sna
sosm
uk
fek
una
uvna24h
una_ucr
fena
feu
diurese
volume_urinario
tempo_internacao_hospitalar
idade
uu_ucr
su
su_scr
dva_mcg_kg_min
score_clinico
sofa
saps3
delta_scr
egfr_basal
uu
uvu24h
uu_su
ucr
uosm
uosm_estimada
bh
sk
imc
scr
scr_basal
sna
sosm
uk
fek
una
uvna24h
una_ucr
fena
feu
diurese
1
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−1

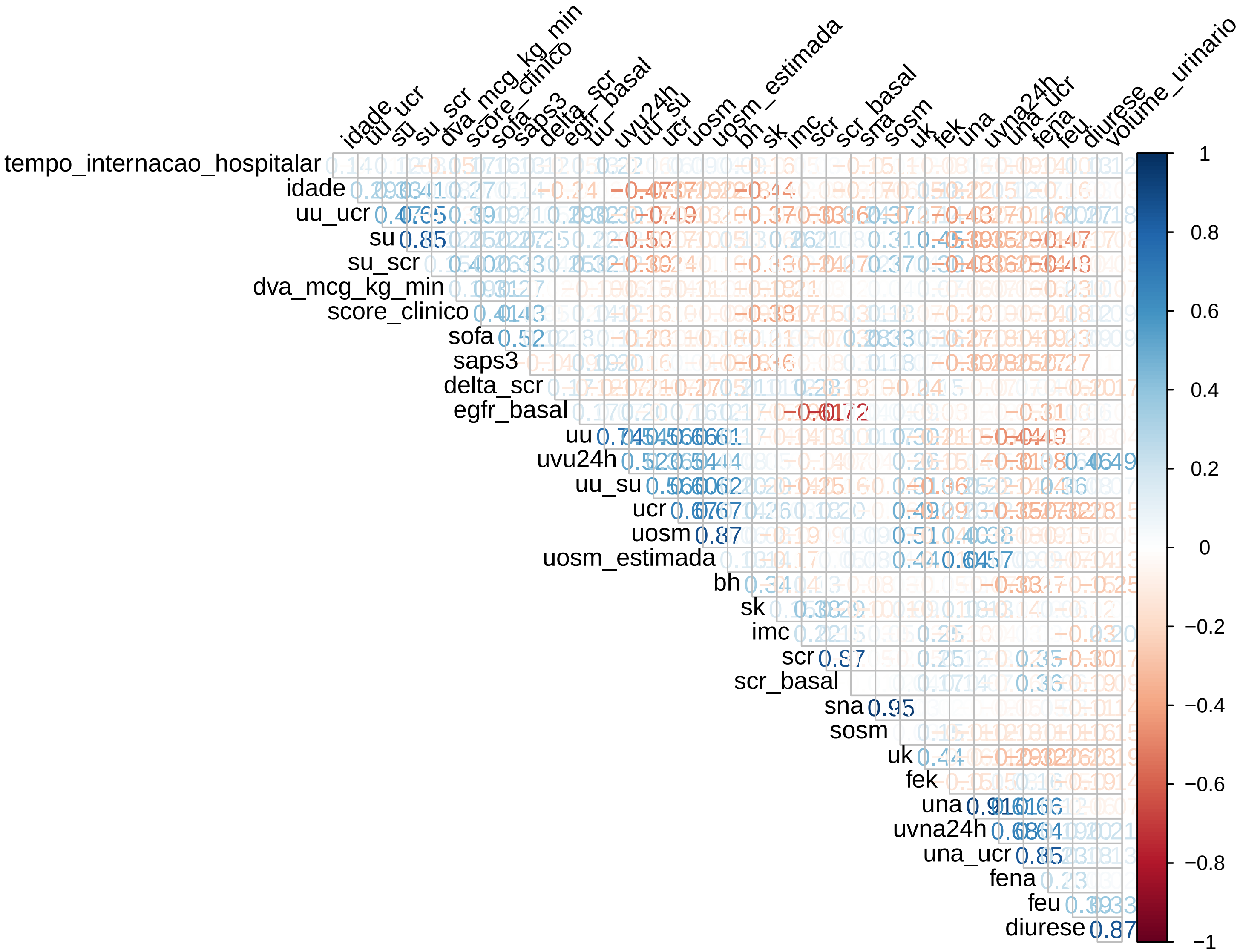
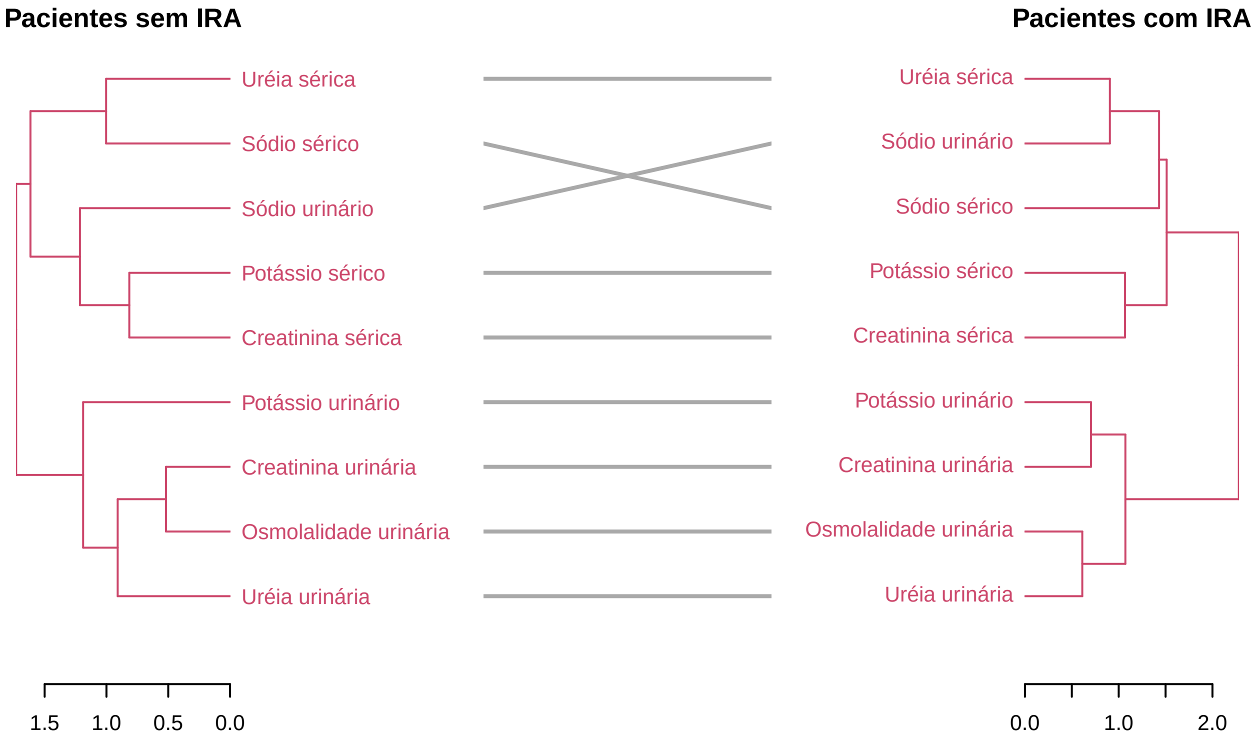

idade
uu_ucr
su
su_scr
dva_mcg_kg_min
score_clinico
sofa
saps3
delta_scr
egfr_basal
uu
uvu24h
uu_su
ucr
uosm
uosm_estimada
bh
sk
imc
scr
scr_basal
sna
sosm
uk
fek
una
uvna24h
una_ucr
fena
feu
diurese
volume_urinario
tempo_internacao_hospitalar
idade
uu_ucr
su
su_scr
dva_mcg_kg_min
score_clinico
sofa
saps3
delta_scr
egfr_basal
uu
uvu24h
uu_su
ucr
uosm
uosm_estimada
bh
sk
imc
scr
scr_basal
sna
sosm
uk
fek
una
uvna24h
una_ucr
fena
feu
diurese
1
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−1

Table 1:

	diurese	bh	sofa	su	scr	sna	sk	sosm	uu	ucr	una	uk	volume__urinario	uvu24h	feu	uu__ucr	una__ucr	uvna24h	uosm	uosm__estimada
diurese	1.00	-0.12	0.06	-0.17	-0.28	-0.11	-0.05	-0.15	-0.05	-0.30	-0.03	-0.22	0.86	0.45	0.43	0.26	0.20	0.24	-0.09	-0.12
bh	-0.12	1.00	0.03	0.13	0.13	-0.08	0.35	-0.03	0.16	0.13	0.04	0.02	-0.24	0.08	0.03	0.01	-0.32	-0.04	0.04	0.13
sofa	0.06	0.03	1.00	0.22	-0.08	0.27	-0.22	0.33	0.02	-0.10	-0.27	0.05	0.08	0.10	-0.25	0.19	-0.15	-0.19	-0.03	-0.18
su	-0.17	0.13	0.22	1.00	0.21	0.01	-0.07	0.31	0.22	-0.16	-0.39	0.01	-0.08	0.08	-0.46	0.47	-0.30	-0.35	-0.10	-0.15
scr	-0.28	0.13	-0.08	0.21	1.00	-0.05	0.38	0.02	-0.13	0.18	0.12	-0.05	-0.16	-0.14	0.01	-0.33	-0.12	0.04	-0.09	0.00
sna	-0.11	-0.08	0.27	0.01	-0.05	1.00	-0.09	0.95	0.11	-0.01	0.01	0.03	-0.14	-0.04	0.03	0.24	-0.05	-0.02	0.09	0.08
sk	-0.05	0.35	-0.22	-0.07	0.38	-0.09	1.00	-0.11	-0.05	0.23	0.19	0.09	0.04	0.06	0.12	-0.36	-0.11	0.17	0.05	0.14
sosm	-0.15	-0.03	0.33	0.31	0.02	0.95	-0.11	1.00	0.17	-0.06	-0.11	0.03	-0.15	-0.02	-0.11	0.37	-0.13	-0.12	0.06	0.04
uu	-0.05	0.16	0.02	0.22	-0.13	0.11	-0.05	0.17	1.00	0.57	-0.15	0.30	-0.05	0.74	-0.14	0.32	-0.45	-0.14	0.66	0.61
ucr	-0.30	0.13	-0.10	-0.16	0.18	-0.01	0.23	-0.06	0.57	1.00	0.21	0.48	-0.16	0.36	-0.34	-0.49	-0.36	0.10	0.67	0.67
una	-0.03	0.04	-0.27	-0.39	0.12	0.01	0.19	-0.11	-0.15	0.21	1.00	-0.06	-0.06	-0.14	0.14	-0.43	0.62	0.91	0.38	0.64
uk	-0.22	0.02	0.05	0.01	-0.05	0.03	0.09	0.03	0.30	0.48	-0.06	1.00	-0.19	0.26	-0.25	-0.27	-0.29	-0.11	0.50	0.44
volume	0.86	-0.24	0.08	-0.08	-0.16	-0.14	0.04	-0.15	-0.05	-0.16	-0.06	-0.19	1.00	0.49	0.35	0.18	0.14	0.23	-0.06	-0.13
uvu24h	0.45	0.08	0.10	0.08	-0.14	-0.04	0.06	-0.02	0.74	0.36	-0.14	0.26	0.49	1.00	0.16	0.30	-0.31	0.05	0.53	0.44
feu	0.43	0.03	-0.25	-0.46	0.01	0.03	0.12	-0.11	-0.14	-0.34	0.14	-0.25	0.35	0.16	1.00	0.19	0.25	0.23	-0.18	-0.05
uu__ucr	0.26	0.01	0.19	0.47	-0.33	0.24	-0.36	0.37	0.32	-0.49	-0.43	-0.27	0.18	0.30	0.19	1.00	-0.11	-0.26	-0.13	-0.20
una__u	0.20	-0.32	-0.15	-0.30	-0.12	-0.05	-0.11	-0.13	-0.45	-0.36	0.62	-0.29	0.14	-0.31	0.25	-0.11	1.00	0.68	-0.10	0.09
uvna24h	0.24	-0.04	-0.19	-0.35	0.04	-0.02	0.17	-0.12	-0.14	0.10	0.91	-0.11	0.23	0.05	0.23	-0.26	0.68	1.00	0.34	0.56
uosm	-0.09	0.04	-0.03	-0.10	-0.09	0.09	0.05	0.06	0.66	0.67	0.38	0.50	-0.06	0.53	-0.18	-0.13	-0.10	0.34	1.00	0.86
uosm__estimada	-0.12	0.13	-0.18	-0.15	0.00	0.08	0.14	0.04	0.61	0.67	0.64	0.44	-0.13	0.44	-0.05	-0.20	0.09	0.56	0.86	1.00

Dendograma

```r
d0 = df %>%
  filter(ira == 0) %>%
  select(all_of(original_columns)) %>%
  drop_na %>%
  cor %>%
  abs %>%
  rename_matrix %>%
  dist %>%
  hclust(method = ’complete’) %>%
  as.dendrogram

d1 = df %>%
  filter(ira == 1) %>%
  select(all_of(original_columns)) %>%
  drop_na %>%
  cor %>%
  abs %>%
  rename_matrix %>%
  dist %>%
  hclust(method = ’ward.D2’) %>%
  as.dendrogram

dl <- dendlist(
  d0 %>%
    set("labels_col",
        value = c("skyblue", "orange", "grey"), k=3) %>%
    set("branches_lty", 1) %>%
    set("branches_k_color",
        value = c("skyblue", "orange", "grey"), k = 3),
  d1 %>%
    set("labels_col",
        value = c("skyblue", "orange", "grey"), k=3) %>%
    set("branches_lty", 1) %>%
    set("branches_k_color",
        value = c("skyblue", "orange", "grey"), k = 3)
)

dl %>%
  untangle(method = "step2") %>%
  tanglegram(common_subtrees_color_lines = FALSE,
             highlight_distinct_edges = TRUE,
             highlight_branches_lwd = FALSE,
             margin_inner = 10,
             main_left = ’Pacientes sem IRA’,
             main_right = ’Pacientes com IRA’,
             lwd = 2,
             k_branches = 1,
             k_labels = 1,
             cex_main = 1.2,
             columns_width = c(5, 3, 5),
             margin_outer = 0.5)
```


Pacientes sem IRA
Pacientes com IRA
Uréia sérica
Sódio sérico
Sódio urinário
Potássio sérico
Creatinina sérica
Potássio urinário
Creatinina urinária
Osmolalidade urinária
Uréia urinária
Uréia sérica
Sódio urinário
Sódio sérico
Potássio sérico
Creatinina sérica
Potássio urinário
Creatinina urinária
Osmolalidade urinária
Uréia urinária
1.5
1.0
0.5
0.0
0.0
1.0
2.0

Testing difference of means

```
df_wilcox = tibble()

for (variable in numerical_columns){
  x = filter(df, ira == 0)[[variable]]
  y = filter(df, ira == 1)[[variable]]

  test = wilcox.test(x, y, alternative = "two.sided", exact = FALSE)

  df_wilcox = bind_rows(df_wilcox,
                        list("variavel" = variable, "Estatística" = test$statistic, "p-valor" = test$p.v
}

df_wilcox = df_wilcox %>%
  arrange(‘p-valor‘) %>%
  mutate(‘Estatística‘  = round(‘Estatística‘, 3)) %>%
  inner_join(df_names, by=’variavel’) %>%
  rename(‘Variável‘ = nome)

significant_numerical_columns = df_wilcox %>%
  filter(‘p-valor‘ <= 0.25) %>%
  select(variavel) %>%
  pull

df_wilcox %>%
  select(‘Variável‘, ‘Estatística‘, ‘p-valor‘) %>%
  mutate(‘p-valor‘ = case_when(‘p-valor‘ == 1 ~ ’> 0.999’,
                               ‘p-valor‘ < 0.001 ~ ’< 0.001’,
                               TRUE ~ as.character(round(‘p-valor‘, 3)))) %>%
  niceFormatting(caption = "Teste Mann-Whitney")
```

Table 2: Teste Mann-Whitney

Variável	Estatística	p-valor
Índice SOFA	140.5	< 0.001
Uréia sérica	176.5	0.004
Creatinina sérica	211.0	0.025
SAPS3	212.5	0.026
Razão entre sódio urinário e creatinina urinária	448.0	0.036
Fração de excreção de potássio	230.0	0.058
Razão entre uréia urinária e uréia sérica	436.0	0.06
Sódio urinário	433.0	0.068
Creatinina sérica basal	244.0	0.101
Osmolalidade sérica	246.0	0.109
Excreção de sódio em 24h	421.0	0.109
Fração de excreção de sódio	392.0	0.188
Delta creatinina sérica	264.0	0.203
Razão entre uréia sérica e creatinina sérica	264.0	0.204
Tempo de internação hospitalar	265.5	0.213
Escore Clínico	272.5	0.219
Dose de noradrenalina	273.0	0.224

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

Variável	Estatística	p-valor
Potássio urinário	270.0	0.246
Fração de excreção de uréia	394.0	0.269
Sódio sérico	276.5	0.297
Balanço hídrico	282.0	0.347
Sódio urinário + Potássio urinário	383.0	0.367
Diurese	377.0	0.428
Osmolalidade urinária estimada	372.0	0.484
Volume urinário	360.0	0.631
IMC	313.5	0.719
Creatinina urinária	351.0	0.754
Osmolalidade urinária	349.0	0.782
Razão entre uréia urinária e creatinina urinária	320.0	0.811
Excreção de uréia em 24h	328.0	0.927
Potássio sérico	329.5	0.948
Ritmo de filtração glomerular	337.0	0.956
Idade	330.5	0.963
Uréia urinária	332.0	0.985

```
df_chisq = tibble()
selected_categorical_columns = categorical_columns[!categorical_columns %in%
                                                    c(’d_ira’, ’kdigo’)]

for (variable in selected_categorical_columns){
  if (length(unique(df[[variable]])) > 1){
    test = chisq.test(df$ira, df[[variable]],
                      simulate.p.value = TRUE)

    df_chisq = bind_rows(df_chisq,
                         list("variavel" = variable,
                              "Estatística" = test$statistic,
                              "p-valor" = test$p.value))
  }
}

df_chisq %>%
  arrange(‘p-valor‘) %>%
  mutate(‘p-valor‘ = case_when(‘p-valor‘ == 1 ~ ’> 0.999’,
                               ‘p-valor‘ < 0.001 ~ ’< 0.001’,
                               TRUE ~ as.character(round(‘p-valor‘, 3))),
         ‘Estatística‘  = round(‘Estatística‘, 3)) %>%
  inner_join(df_names, by=’variavel’) %>%
  rename(‘Variável‘ = nome) %>%
  select(‘Variável‘, ‘Estatística‘, ‘p-valor‘) %>%
  niceFormatting(caption = "Teste Chi-quadrado")
```

Table 3: Teste Chi-quadrado

Variável	Estatística	p-valor
Uso de ventilação mecânica	3.02	0.149
Asma ou DPOC	1.65	0.521
Câncer ativo	0.65	0.632
Cor	0.92	0.713
Diabetes mellitus	0.17	0.758
Hipertensão arterial sistêmica	0.12	0.791
Fator de risco	0.56	0.843
Insuficiência cardíaca congestiva	0.06	> 0.999
Acidente vascular cerebral	0.09	> 0.999
Doença vascular periférica	0.15	> 0.999
Uso de diuréticos	0.15	> 0.999
Uso de vasopressina	0.03	> 0.999
Uso de iECA ou espironolactona	0.02	> 0.999
Pós operatório	0.01	> 0.999

Selected variables

```
length(significant_numerical_columns)

drop = c(’uvna24h’, ’una_ucr’, ’fena’, ’su_scr’, ’saps3’)
```

```r
selected_numerical_columns = setdiff(significant_numerical_columns, drop)

length(selected_numerical_columns)

dput(selected_numerical_columns)
```