Visualisations

Code

load required packages

Hide

```
pacman::p_load(tidyr, dplyr, ggplot2, openxlsx, lubridate)
```

set working directory

import dataset

Hide

```
data <- read.csv("training_v2.csv")
# cln_data <- read.csv("cln_data.csv")
merge_data <- read.csv("merged_data_1.csv")
dic <- read.csv("WiDS Datathon 2020 Dictionary.csv")
```

columns

Hide

```
colnames(data)[4] <- "death"

death <- 4
age <- 5
ethnic <- 8
gender <- 9
```

get id

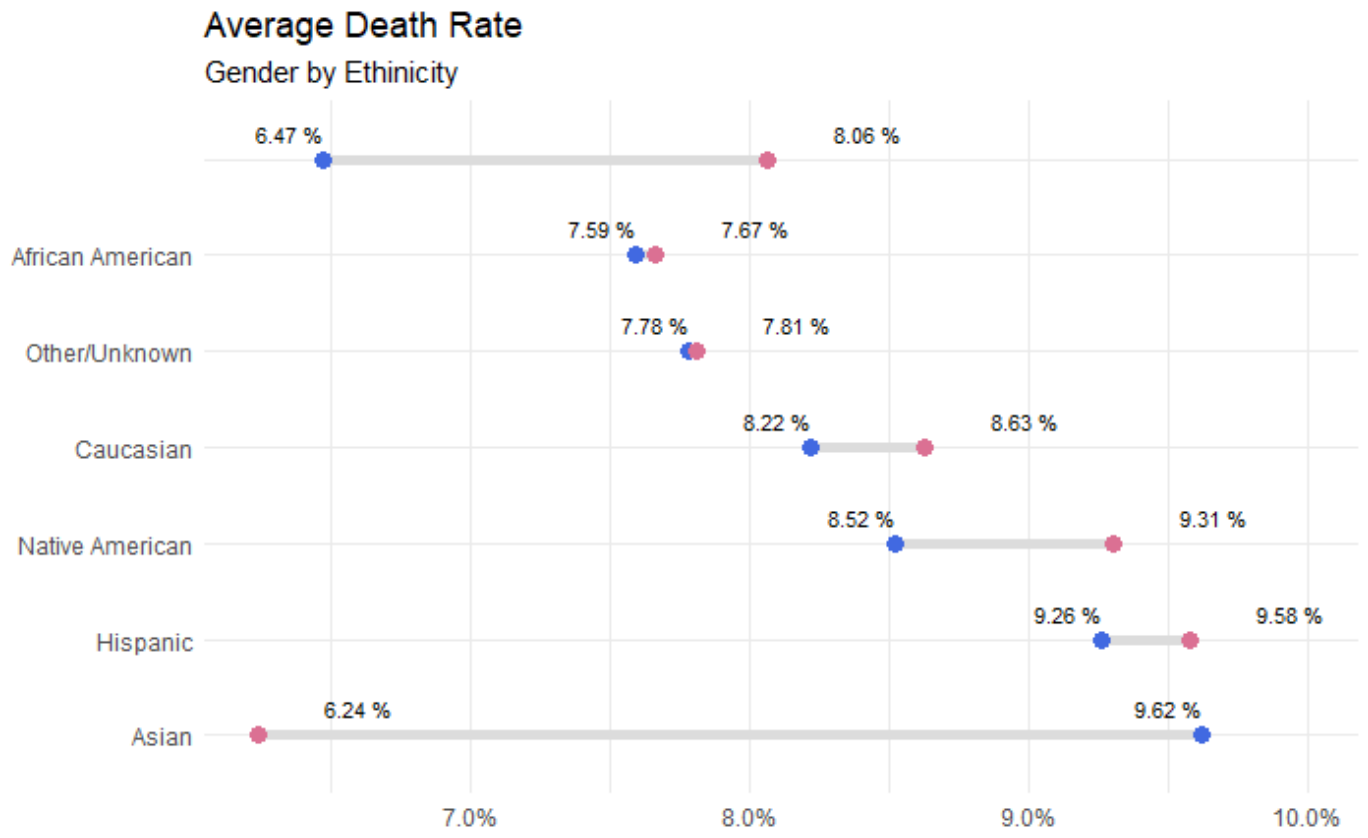
Hide

```
identifier <- c("encounter_id","hospital_id","patient_id")
```

demographics - death by ethnicity/gender

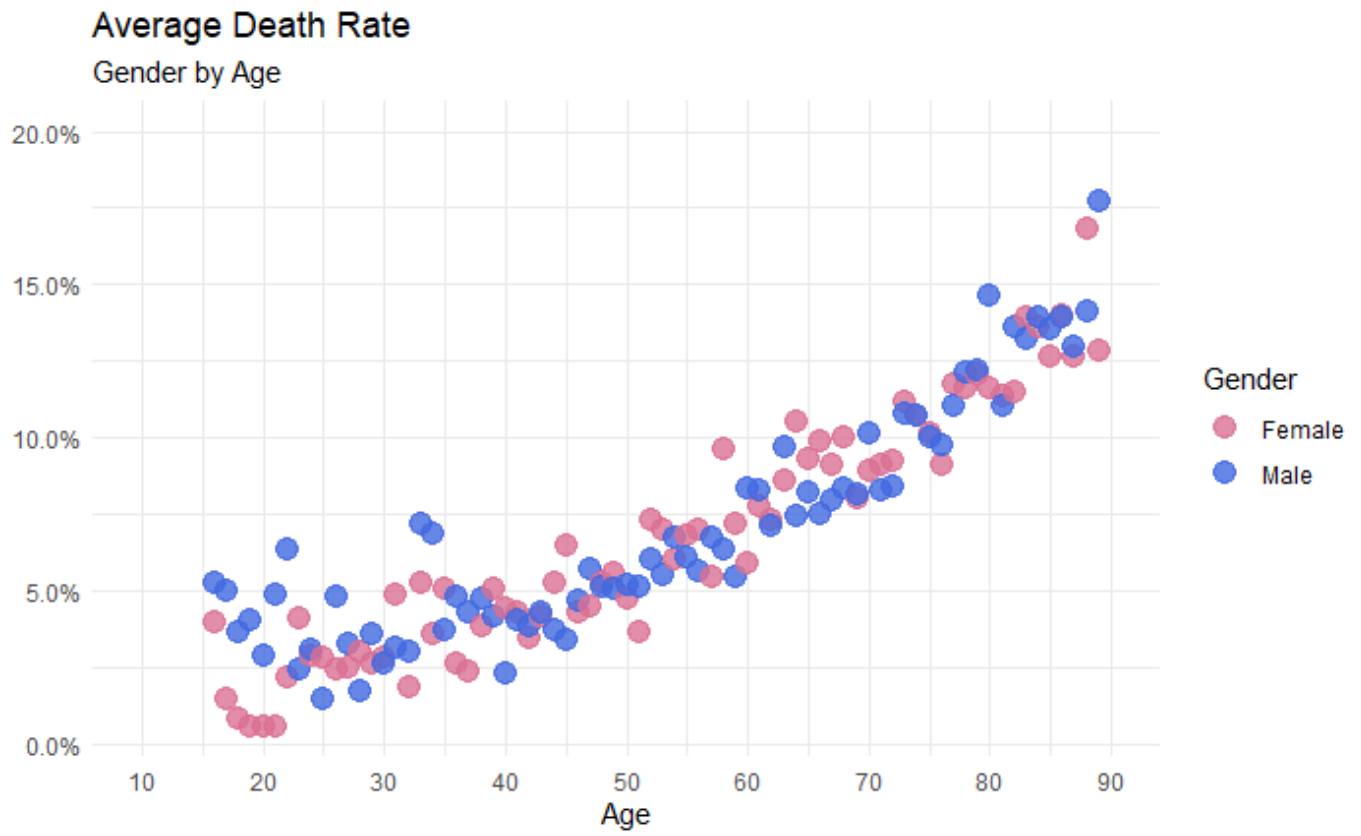
Hide

```
age_gender_eth <- data[, c(4, 5, 8, 9)]
age_gender_eth  <- na.omit(age_gender_eth)

p <- age_gender_eth %>%
  mutate(age = as.integer(age),
         gender = case_when(gender == "F" ~ "F",
                            gender == "M" ~ "M",
                            TRUE ~ "NA"),
         death = as.integer(death),
         ethnicity = factor(ethnicity)) %>%
  filter(gender != "NA")

p %>%
  group_by(gender, ethnicity) %>%
  summarise(mean = mean(death)) %>%
  ungroup() %>%
  spread(key = gender, value = mean) %>%
  ggplot(aes(x =M, xend = F, y = reorder(ethnicity,-M), group = ethnicity)) +
  ggalt::geom_dumbbell(colour = "#DCDCDC",
                       size = 2,
                       colour_x ="#4169E1",
                       colour_xend = "#DB7093",
                       size_x = 3,
                       size_xend = 3) +
  scale_x_continuous(label=scales::percent, limits = c(NA, 0.1)) +
  labs(x=NULL,
      y=NULL,
      title="Average Death Rate",
      subtitle="Gender by Ethinicity") +
     theme(plot.title = element_text(hjust=0.5, face="bold"),
              plot.background=element_rect(fill="#f7f7f7"),
              panel.background=element_rect(fill="#f7f7f7"),
              panel.grid.minor=element_blank(),
              panel.grid.major.y=element_blank(),
              panel.grid.major.x=element_line(),
              axis.ticks=element_blank(),
              legend.position="top",
              panel.border=element_blank())+
  geom_text(color="black", size=3, hjust=1,vjust = -1,
                  aes(x=M, label=paste(round(M*100,2), "%")))+
  geom_text(aes(x=F, label=paste(round(F*100,2),"%")),
                  color="black", size=3, hjust=-1, vjust = -1)+
  theme_minimal()
```



Hide

```
age_b <- seq(10, 90, by = 10)

p %>%
  group_by(age, gender) %>%
  summarise(mean = mean(death)) %>%
  ungroup() %>%
  filter(!is.na(gender)) %>%
  ggplot(aes(x = age, y = mean, colour = gender)) +
  geom_point(alpha = 0.8, size = 4) +
  scale_x_continuous(breaks = age_b, labels = age_b, limits = c(10,90)) +
  scale_colour_manual(name = "Gender",
                      labels = c("Female",
                                 "Male"),
                      values = c("F" = "#DB7093",
                                 "M" = "#4169E1")) +
  labs(x="Age",
      y=NULL,
      title="Average Death Rate",
      subtitle="Gender by Age") +
  scale_y_continuous(label=scales::percent, limits = c(NA, 0.2)) +
  theme_minimal()
```

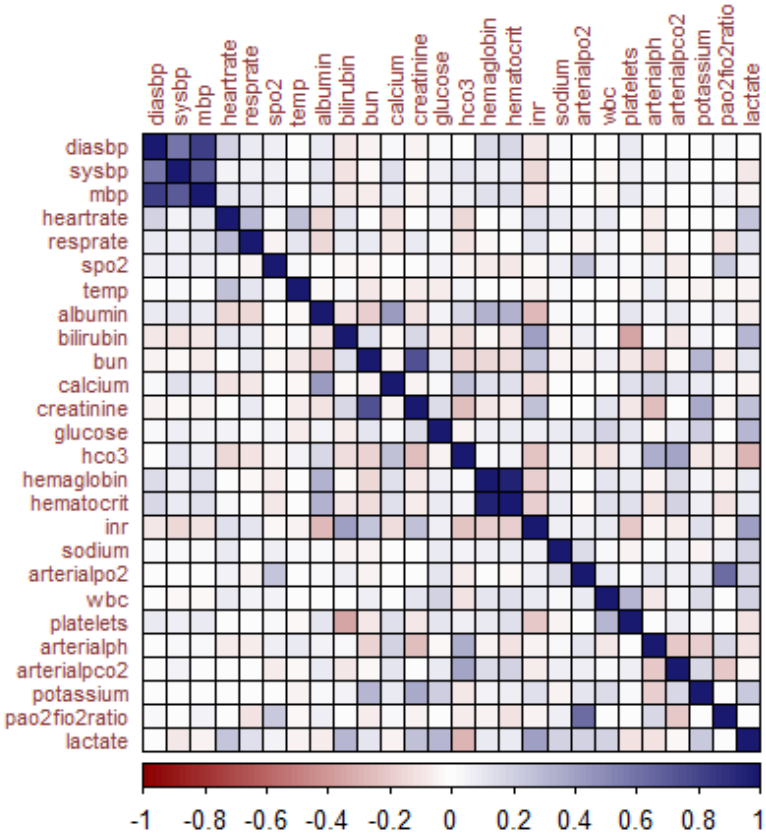


Hide

```
cln_data_1 <- merge_data %>%
  select(ends_with("max")) %>%
  sapply(as.numeric)
cln_data_1 <- as.data.frame(cln_data_1)

names(cln_data_1) <- gsub(pattern = "d1_*", replacement = "", x = names(cln_data_1))
names(cln_data_1) <- gsub(pattern = "*_max", replacement = "", x = names(cln_data_1))
```

Hide

```
library(RColorBrewer)
corr_mat=cor(cln_data_1,method="s")
corrplot::corrplot(corr_mat, method="color", outline = TRUE, addrect = 4, rect.col = "black",
rect.lwd = 5, cl.pos = "b", tl.col = "indianred4",
                   tl.cex = 0.7, number.digits = 2, number.cex = 0.75, col = colorRampPalette
(c("darkred","white","midnightblue"))(100))
```



Hide

```
cln_data_2 <- merge_data %>%
  sapply(as.numeric) %>%
  as.data.frame()


cln_data_2[,3:53] <- sapply(cln_data_2[,3:53], scale)
cln_data_2 <- as.data.frame(cln_data_2)

names(cln_data_2) <- gsub(pattern = "d1_*", replacement = "", x = names(cln_data_2))

gat <- cln_data_2 %>%
  gather(key = key, value = value, diasbp_max:lactate_min) %>%
  separate(key, into = c("key" , "type"), sep = "_")
```

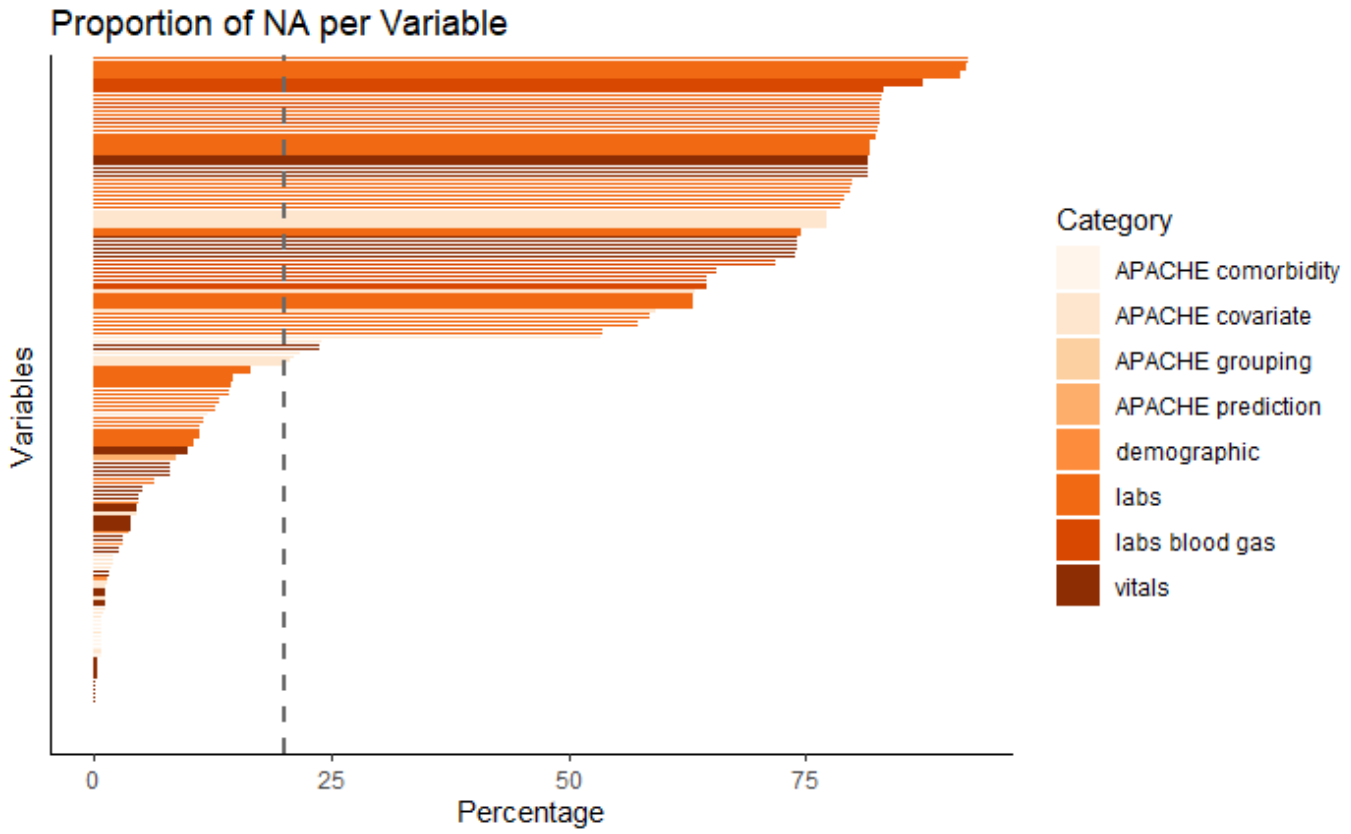
Hide

```
death <- list("1" = "Died",
              "0" = "Survived")

inv_labeller <- function(variable,value){
  return(death[value])
}


gat %>%
  ggplot(aes(x=key, y=value, colour = type))+
  geom_point(alpha = 0.5)  +
  labs(x = NULL,
       y = NULL,
       title="Min-max Variable Impact on Survival",
       subtitle="Normalized variable values") +
  scale_colour_manual(name = "Type",
                      labels = c("Max",
                                 "Min"),
                      values = c("max" = "#00008B",
                                 "min" = "#A52A2A")) +
  coord_flip() +
  theme_minimal() +
  facet_wrap(~ hospital_death, labeller = inv_labeller)
```

Hide

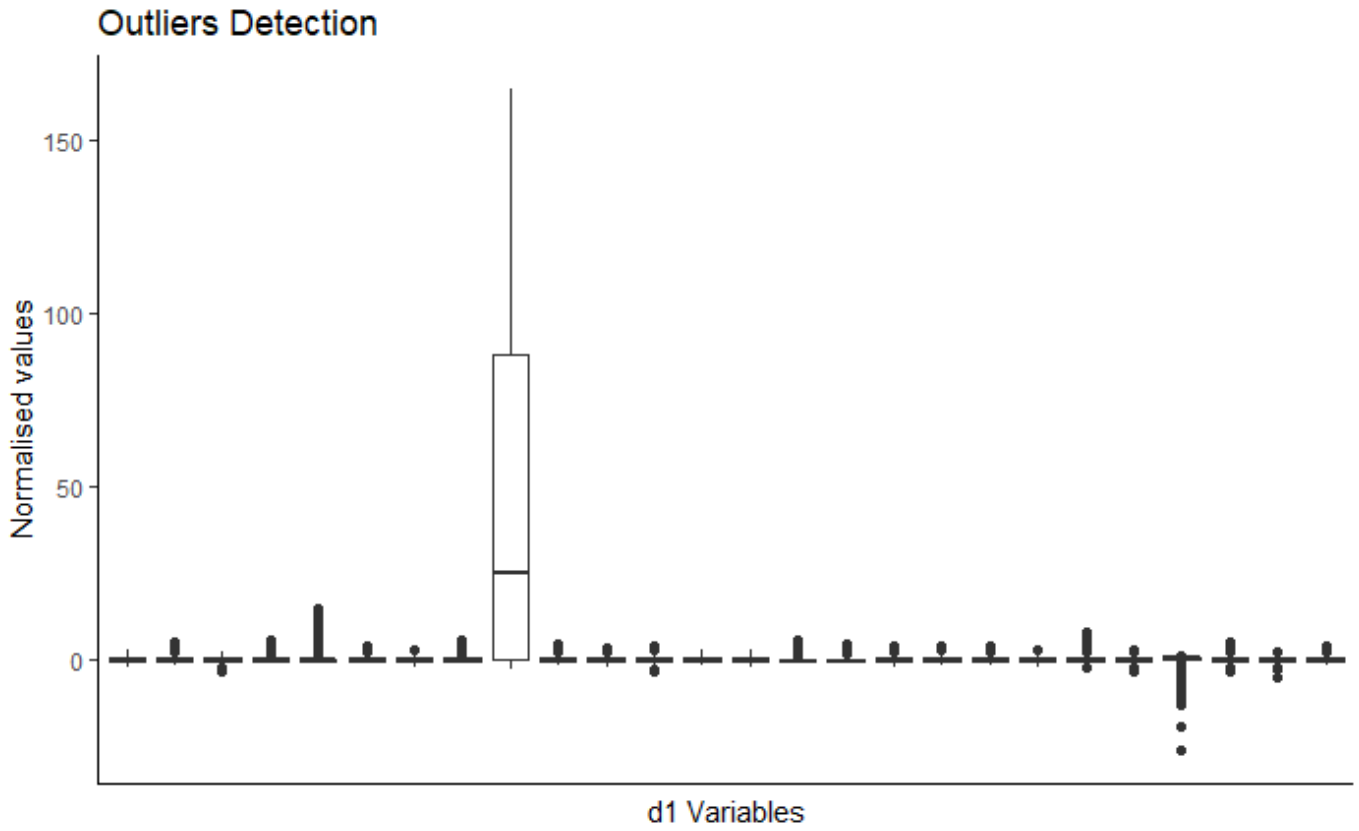
```
data %>%
  summarise_each(funs(100*mean(is.na(.)))) %>%
  select(-identifier) %>%
  gather(key= key, value = value) %>%
  arrange(desc(value)) %>%
  inner_join(dic, by = c("key" = "Variable.Name")) %>%
  ggplot(aes(x = reorder(key, value), y = value, fill = Category)) +
  geom_bar(stat="identity",  width=0.7) +
  geom_hline(yintercept = 20, linetype="dashed", size = 1, colour = "#696969") +
  labs(x = "Variables",
       y = "Percentage",
       title = "Proportion of NA per Variable") +
  scale_fill_brewer(palette="Oranges") +
  theme_classic() +
  theme(axis.title.y=element_text(),
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank()) +
  coord_flip()
```

```
Note: Using an external vector in selections is ambiguous.
□[34mi□[39m Use `all_of(identifier)` instead of `identifier` to silence this message.
□[34mi□[39m See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
□[90mThis message is displayed once per session.□[39m
```



Hide

```
gat %>%
  ggplot(aes(x = key, y = value)) +
  geom_boxplot()+
  labs(x = "d1 Variables",
      y = "Normalised values",
      title = "Outliers Detection") +
  theme_classic() +
  theme(axis.title.x=element_text(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```



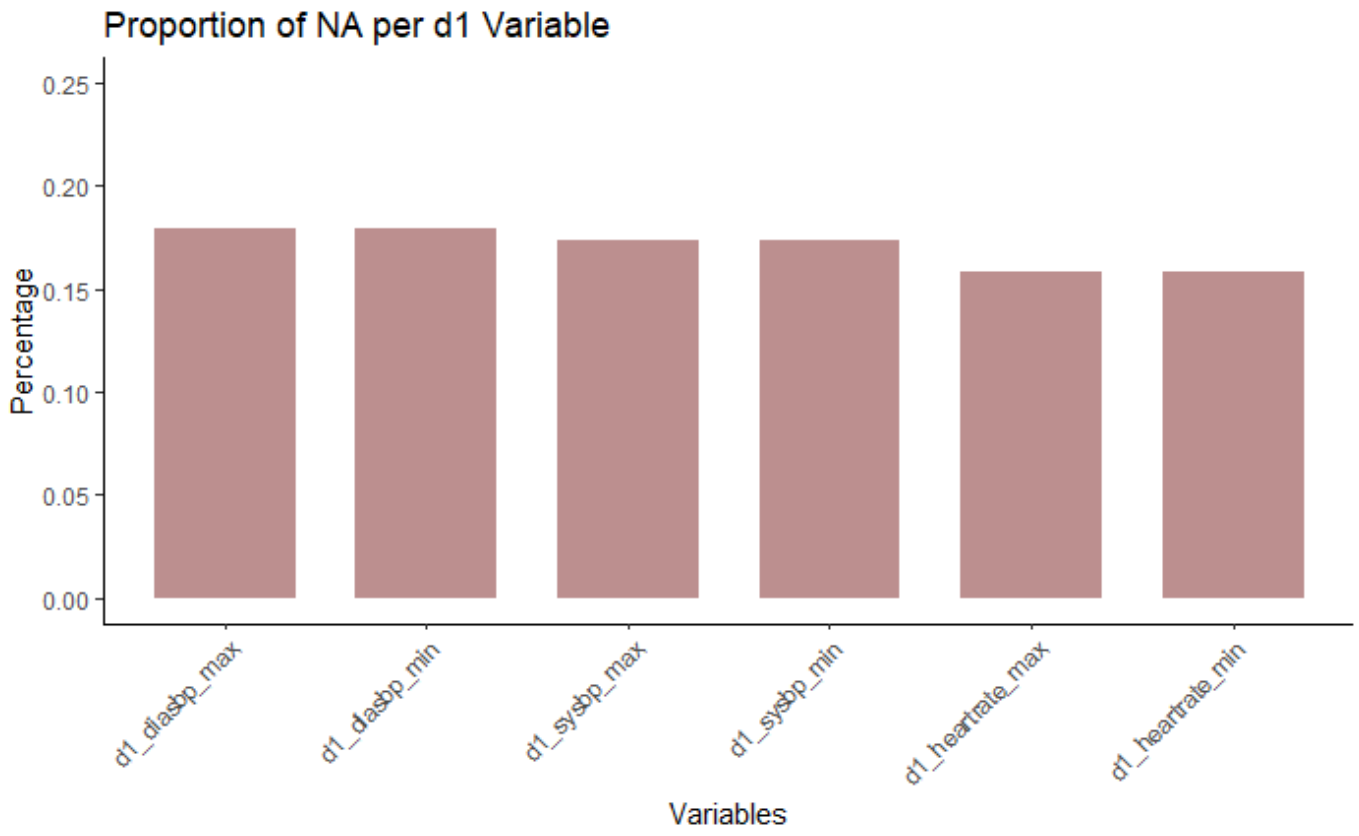
Hide

```
NA
NA
```

Hide

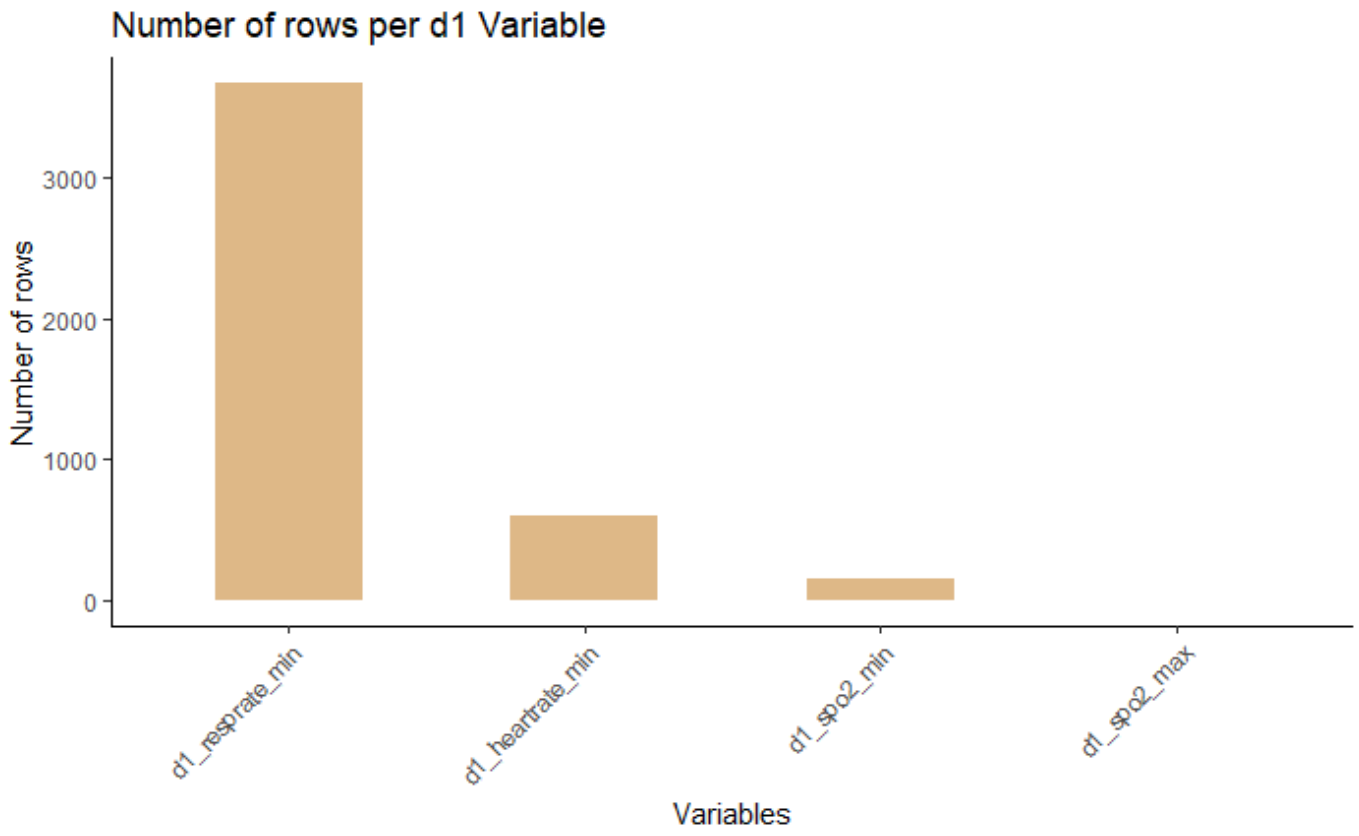
```
data %>%
  summarise_each(funs(100*mean(is.na(.)))) %>%
  select(starts_with("d1")) %>%
  gather(key= key, value = value) %>%
  arrange(desc(value)) %>%
  filter(value < 0.2) %>%
  inner_join(dic, by = c("key" = "Variable.Name")) %>%
  ggplot(aes(x = reorder(key, -value), y = value)) +
  geom_bar(fill = "#BC8F8F", stat="identity",  width=0.7) +
  ylim(0,0.25) +
  labs(x = "Variables",
       y = "Percentage",
       title = "Proportion of NA per d1 Variable") +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
Column `key`/`Variable.Name` joining character vector and factor, coercing into character vec
tor
```



Hide

```
data %>%
  select(starts_with("d1")) %>%
  gather(key= key, value = value) %>%
  filter(value == "0") %>%
  group_by(key) %>%
  summarise(count = n()) %>%
  ggplot(aes(x = reorder(key, -count), y = count)) +
  geom_bar(fill = "#DEB887", stat="identity",  width=0.5) +
  labs(x = "Variables",
       y = "Number of rows",
       title = "Number of rows per d1 Variable") +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Hide

```
library(ggthemes)

gat %>%
  group_by(key, type) %>%
  summarise(sum = sum(value)) %>%
  mutate(sum = ifelse(type == "max" & sum < 0, sum * -1, sum),
         sum = ifelse(type == "min" & sum > 0, sum * -1, sum)) %>%
  ggplot(aes(x = reorder(key, -sum), y = sum/100, fill = type)) +
  geom_bar(stat="identity",  width=0.7) +
  labs(x = "Variables",
       y = "Normalized values",
       title = "Max vs Min Values") +
  scale_colour_manual(name = "Type",
                      labels = c("Max",
                                 "Min")) +
  theme_tufte() +
  coord_flip() +
  theme(plot.title = element_text(hjust = .5)) +
  scale_fill_brewer(palette = "Dark2") +
   theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```

Hide

```
com <- data[,c(4,177:184)]

com %>%
  rename(AIDS = aids,
         Cirrhosis = cirrhosis,
         Diabetes = diabetes_mellitus,
         Hepatic_failure = hepatic_failure,
         Immunosuppression = immunosuppression,
         Leukemia = leukemia,
         Lymphona = lymphoma,
         Metastatic_cancer = solid_tumor_with_metastasis) %>%
  gather(key = com, value = value, AIDS:Metastatic_cancer) %>%
  filter(value > 0 ) %>%
  mutate(com = ifelse(value == 0, "no disease", com)) %>%
  group_by(com, death) %>%
  summarise(npax = n()) %>%
  mutate(death = ifelse(death == 1, "died", "survived")) %>%
  spread(key = death, value = npax) %>%
  mutate(deathrate = died / (survived + died)) %>%
  ggplot(aes(x = com, y = deathrate)) +
  geom_point(aes(size =died), colour = "#3CB371") +
  labs(x = NULL,
       y = NULL,
       title = "Death Rate by Comorbidity") +
  scale_y_continuous(label=scales::percent, limits = c(NA, 0.2)) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  theme(legend.position = "none")
```