

# Working with Missing Data: Takeaways

by Dataquest Labs, Inc. - All rights reserved © 2021

## Syntax

- Summing a subset of dataframe over rows:

```
df %>% mutate( new_column_name = rowSums(.[1:3]) )
```

- Selecting a subset of a dataframe which variable names match with a string:

```
df %>% select( contains( string ) )  
df %>% select( starts_with( string ) )  
df %>% select( ends_with( string ) )
```

- Replacing matching values with a single value:

```
df %>% mutate( colname = if_else(condition, val_if_true, val_if_false) )
```

- Replacing matching values with corresponding values from a vector:

```
df %>% mutate( colname = if_else(condition, list_of_values_if_true,  
list_of_values_if_false) )
```

- Preparing data for heatmaps visualization:

```
df_na <- map_df(df, function(x) as.numeric(is.na(x)))  
df_na_heat <- df_na %>%  
  pivot_longer(cols = everything(),  
    names_to = "x") %>%  
  group_by(x) %>%  
  mutate(y = row_number())
```

- Creating a function to plot NA matrix as a heatmap:

```
plot_na_matrix <- function(df) {  
  # Preparing the dataframe for heatmaps  
  df_heat <- df %>%  
    pivot_longer(cols = everything(),  
      names_to = "x") %>%  
    group_by(x) %>%  
    mutate(y = row_number())  
  
  # Ensuring the order of columns is kept as it is  
  df_heat <- df_heat %>%  
    ungroup() %>%  
    mutate(x = factor(x, levels = colnames(df)))  
  
  # Plotting data  
  g <- ggplot(data = df_heat, aes(x=x, y=y, fill=value)) +  
    geom_tile() +  
    theme(legend.position = "none",  
      axis.title.y=element_blank(),  
      axis.text.y =element_blank(),  
      axis.ticks.y=element_blank(),  
      axis.title.x=element_blank(),
```

```

        axis.text.x = element_text(angle = 90, hjust = 1))

# Returning the plot
g
}

```

- Computing the correlation matrix with `cor()` function:

```
missing_corr <- cor(df_na)
```

- Creating a function to plot NA correlation matrix as a heatmap:

```

plot_na_correlation <- function(df) {
  # Taking the lower triangle of the correlation matrix
  missing_corr_up <- df
  missing_corr_up[lower.tri(missing_corr_up)] <- NA
  missing_corr_up <- data.frame(missing_corr_up)
  # Preparing the dataframe for heatmaps
  col_names <- colnames(missing_corr_up)
  missing_corr_up_heat <- missing_corr_up %>%
    pivot_longer(cols = everything(),
                 names_to = "x") %>%
    group_by(x) %>%
    mutate(y = col_names[row_number()]) %>%
    na.omit

  # Ordering triangle
  ordered_cols_asc <- col_names[order(colSums(is.na(missing_corr_up)))]
  ordered_cols_desc <- col_names[order(-colSums(is.na(missing_corr_up)))]
  missing_corr_up_heat <- missing_corr_up_heat %>%
    ungroup() %>%
    mutate(x = factor(x, levels = ordered_cols_asc)) %>%
    mutate(y = factor(y, levels = ordered_cols_desc))

  # Plotting heatmaps
  g <- ggplot(data = missing_corr_up_heat, aes(x=x, y=y, fill=value)) +
    geom_tile() +
    geom_text(aes(label=value)) +
    theme_minimal() +
    scale_fill_gradientn(colours = c("white", "yellow", "red"), values = c(-1,0,1)) +
    theme(legend.position = "none",
          axis.title.y=element_blank(),
          axis.title.x=element_blank(),
          axis.text.x = element_text(angle = 90, hjust = 1))

  # Returning the plot
  g
}

```

## Concepts

- Imputation is the process of replacing missing values with other values.
- Imputing can be a better option than simply dropping values because you retain more of your original data.

- You might find values for imputation by:
  - Deriving the value from related columns.
  - Using the most common non-NA value from a column.
  - Using an placeholder for missing values.
  - Augmenting factual data (e.g. location data) using an external resource.
- Using plots can help identify patterns in missing values which can help with imputation.

## Resources

- [ggplot2](#) [cheat sheet](#)
- [dplyr](#) [package](#)

Takeaways by Dataquest Labs, Inc. - All rights reserved © 2021