Explonatory Data Analysis

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggplot2)
library(latex2exp)
library(svglite)

set.seed(1101)

.save_and_display <- function(g, main, file, width = 960, height = 480, ...) {
    g <- g +
        ggtitle(main) +
        theme(
            text = element_text(size = 12),
            plot.title = element_text(hjust = 0.5, size = 16),
            strip.text = element_text(size = 12),
            legend.position="bottom"
        )

    ggsave(file, plot = g, units = 'px', width = width, height = height, dpi = 100, ...)
    g
}

.gg.correct.colouring <- function(g) {
    g + scale_fill_manual(
        breaks = c(0, 1),
        values=c('#0D92F4', '#F95454')
    ) +
    scale_color_manual(
        breaks = c(0, 1),
        values=c('#0D92F4', '#F95454')
    )
}
```

Input dataframe

```
CATEGORICAL_VARIABLES <- c(
    'sex',
    'chest.pain',
    'fbs',
    'rest.ecg',
    'angina',
    'blood.disorder'
)

RESPONSE <- 'disease'

NUMERICAL_VARIABLES <- c(
    'age',
    'bp',
    'chol',
    'heart.rate',
    'st.depression',
    'vessels'                # 'vessels' is a discrete small variable from raning from 0-4
)

df <- read.csv('heart-disease-dsa1101.csv') %>%
    mutate_at(all_of(c(CATEGORICAL_VARIABLES, RESPONSE)), as.factor) %>%
    filter(blood.disorder != 0) %>%
    mutate(st.depression = st.depression + 0.01) %>%
    mutate(vessels = vessels + 0.01)
```

```
## Warning: Using `all_of()` outside of a selecting function was deprecated in tidyselect
## 1.2.0.
## i See details at
##   <https://tidyselect.r-lib.org/reference/faq-selection-context.html>
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
df_variables <- df %>% sapply(class) %>% data.frame %>% rownames_to_column %>%
    rename(feature = 1, type = 2) %>%
    mutate(type = factor(type, levels = c('factor', 'integer', 'numeric'))) %>%
    arrange(type)


# df_variables %>%
#     knitr::kable(format = 'latex') %>%
#     writeLines()
```

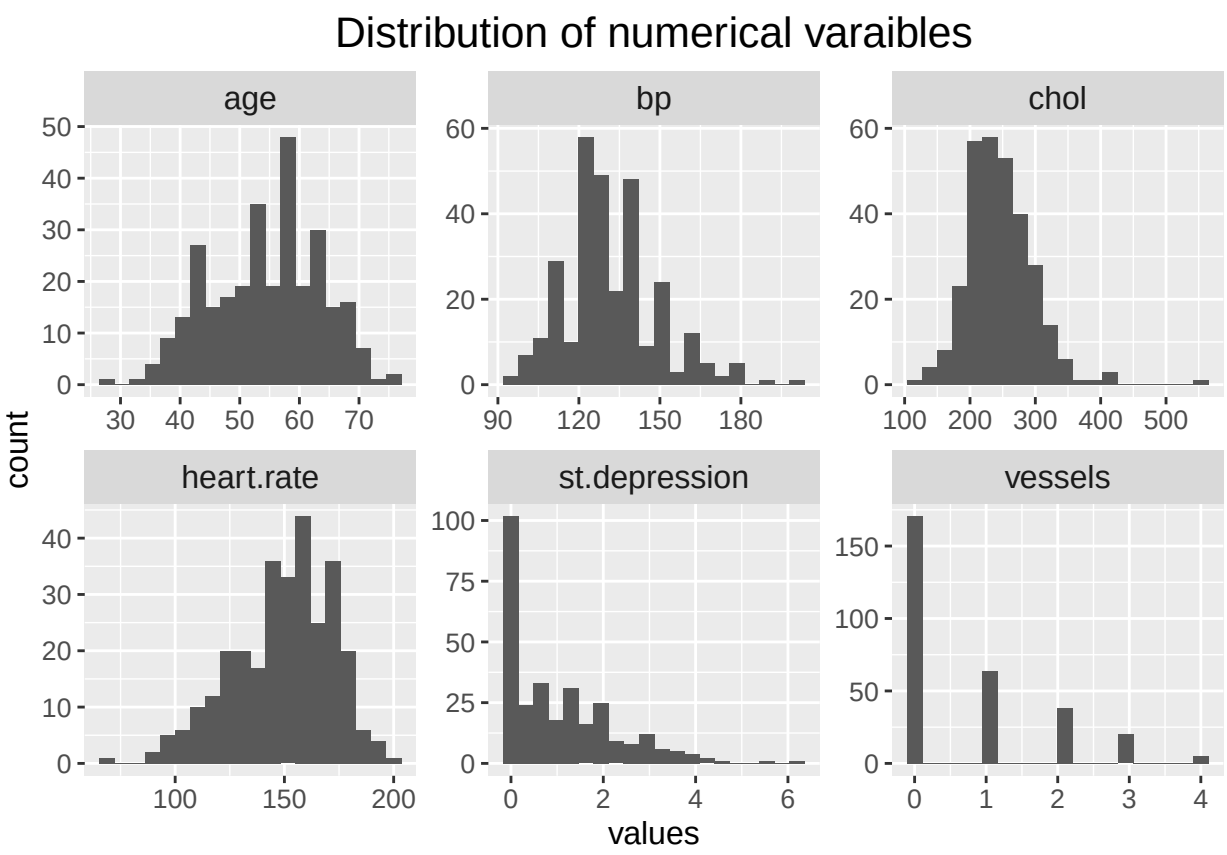
Numerical variables

Normality test

```
df_numericals <- df %>%
    select(all_of(NUMERICAL_VARIABLES), RESPONSE) %>%
    pivot_longer(all_of(NUMERICAL_VARIABLES), names_to = "stats", values_to = "values")
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(RESPONSE)
##
##   # Now:
##   data %>% select(all_of(RESPONSE))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
(df_numericals %>%
    ggplot(aes(
        x = values
    )) +
        facet_wrap(stats ~ ., scales = "free") +
        geom_histogram(bins = 20)) %>%

        .save_and_display(
            'Distribution of numerical varaibles',
            '../figures/23.numerical.distribution.pdf'
        )
```

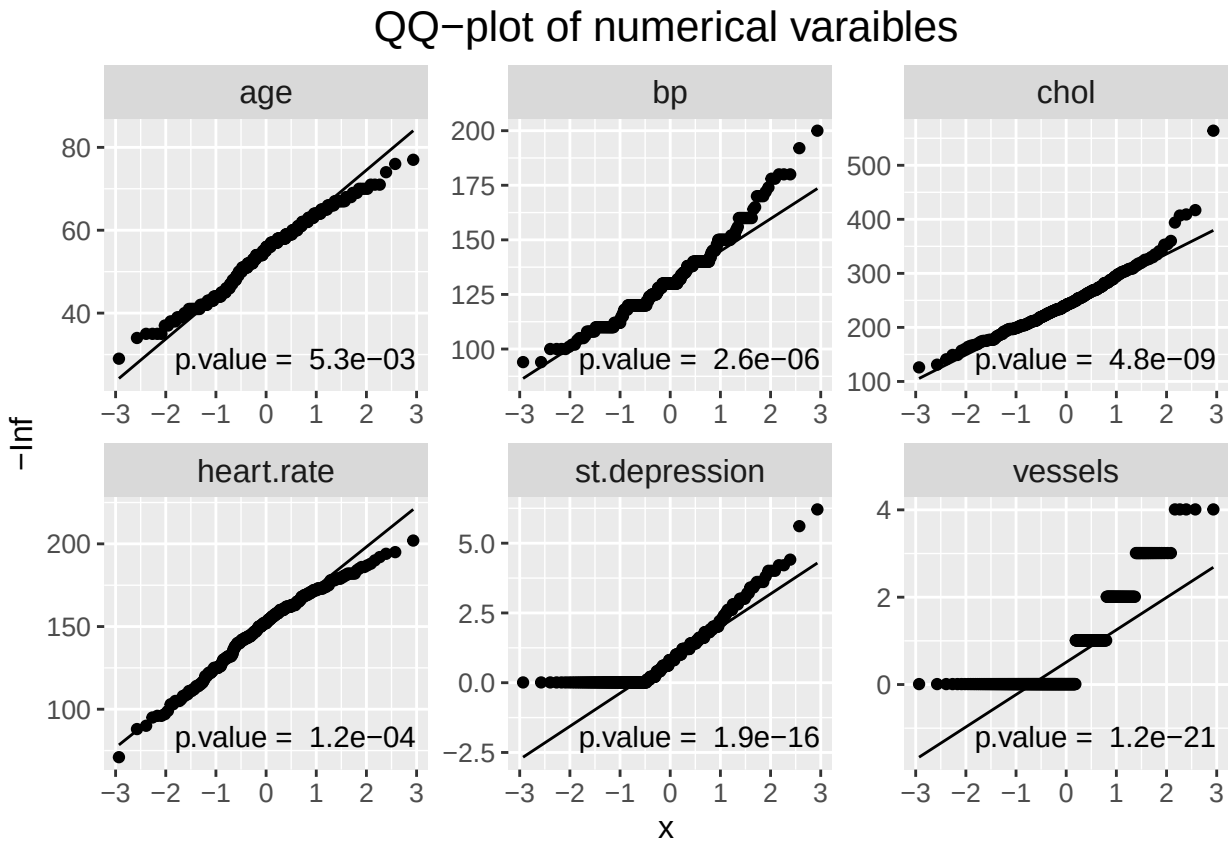



```
shapiro.test.p.value <- df_numericals %>% group_by(stats) %>%
    summarise(p.value = shapiro.test(values)$p.value)
```

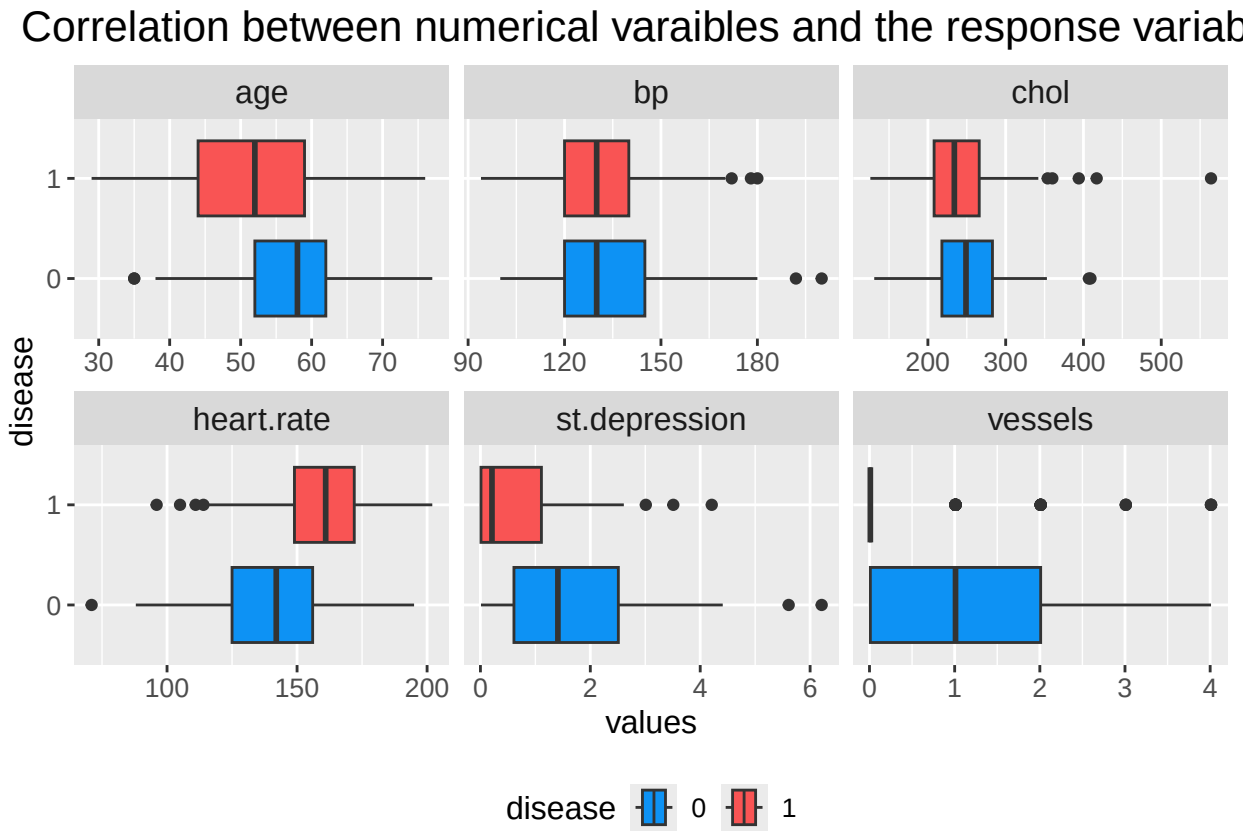
```
shapiro.test.p.value
```

```
## # A tibble: 6 x 2
##   stats          p.value
##   <chr>            <dbl>
## 1 age           5.32e- 3
## 2 bp            2.64e- 6
## 3 chol          4.84e- 9
## 4 heart.rate    1.21e- 4
## 5 st.depression 1.86e-16
## 6 vessels       1.22e-21
```

```
(df_numericals %>%
    ggplot() +
        facet_wrap(stats ~ ., scales = "free") +
        geom_qq(aes(sample = values)) +
        geom_qq_line(aes(sample = values)) +
        geom_text(
            data = shapiro.test.p.value,
            mapping = aes(
                x = Inf,
                y = -Inf,
                label = paste('p.value = ', format(p.value, trim = T, digits = 2)),
                hjust = 1.05,
                vjust = -1.05
            )
        )
) %>%

        .save_and_display(
            'QQ-plot of numerical varaibles',
            '../figures/23.numerical.qq.pdf'
        )
```




```
(df_numericals %>%
    ggplot(aes(
        x = values, y = disease, fill = disease
    )) +
        facet_wrap(stats ~ ., scales = 'free_x') +
        geom_boxplot()) %>%

        .gg.correct.colouring %>%
        .save_and_display(
            'Correlation between numerical varaibles and the response variable',
            '../figures/23.numerical.correlation.to.response.pdf'
        )
```



Box-cox Transformation

```
.boxcox <- function(xs, LAMBDA = 0) {
    sapply(xs, function(x) {
        if (LAMBDA == 0)
            return (log(x))
        return (x ** LAMBDA - 1) / LAMBDA
    })
}

.boxcox.p.values <- function(xs, lambdas) {
    sapply(lambdas, function(k) {
        box.cox <- .boxcox(xs, k)
        if (var(box.cox) == 0) { return (0) }
        shapiro.test(box.cox)$p.value
    })
}

.boxcox.best.lambda <- function(xs, lambdas) {
    y <- .boxcox.p.values(xs, lambdas)
    idx <- which(y == max(y))
    lambdas[idx]
}

.boxcox.plot <- function(xs, lambdas) {
    y <- .boxcox.p.values(xs, lambdas)
```

```
    plot( lambdas, y, xlab = TeX('\\lambda'), ylab = 'p-value' )
}
```

```
ALPHA = 0.01
lambdas <- seq(-5.0, 5.0, .005)

df_boxcox_param <- df %>%
    select(all_of(NUMERICAL_VARIABLES)) %>%
    reframe(
        lambdas = lambdas,
        across(all_of(NUMERICAL_VARIABLES), .boxcox.p.values, lambdas)
    ) %>%
    pivot_longer(-lambdas, names_to = 'stats', values_to = 'p.value')
```

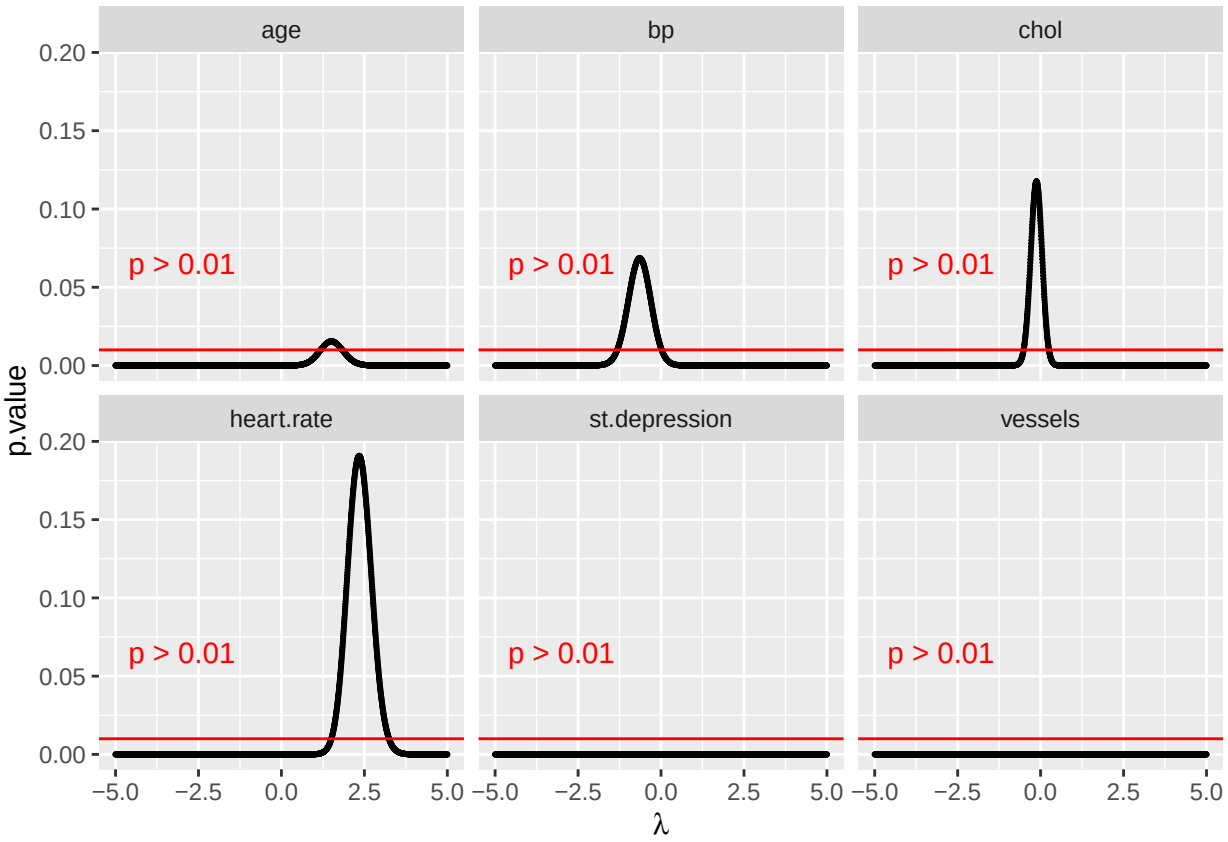
```
## Warning: There was 1 warning in `reframe()`.
## i In argument: `across(all_of(NUMERICAL_VARIABLES), .boxcox.p.values,
##   lambdas)`.
## Caused by warning:
## ! The `...` argument of `across()` is deprecated as of dplyr 1.1.0.
## Supply arguments directly to `.fns` through an anonymous function instead.
##
##   # Previously
##   across(a:b, mean, na.rm = TRUE)
##
##   # Now
##   across(a:b, \(x) mean(x, na.rm = TRUE))
```

```
df_boxcox_param_best <- df_boxcox_param %>%
    group_by(stats) %>%
    slice_max(p.value, n = 1) %>%
    mutate(significant = p.value > ALPHA)

df_boxcox_param_default <- df_boxcox_param %>%
    filter(lambdas == 1.000)

df_boxcox_param_summary <- df_boxcox_param_best %>%
    inner_join(df_boxcox_param_default, by = 'stats', suffix = c('', '.old'))

df_boxcox_param_summary
```

```
## # A tibble: 6 x 6
## # Groups:   stats [6]
##   lambdas stats          p.value significant lambdas.old p.value.old
##     <dbl> <chr>            <dbl> <lgl>             <dbl>       <dbl>
## 1   1.50  age           1.54e- 2 TRUE                  1    5.32e- 3
## 2  -0.645 bp            6.86e- 2 TRUE                  1    2.64e- 6
## 3  -0.125 chol          1.18e- 1 TRUE                  1    4.84e- 9
## 4   2.34  heart.rate     1.91e- 1 TRUE                  1    1.21e- 4
## 5   0.565 st.depression 7.06e-13 FALSE                 1    1.86e-16
## 6   0.675 vessels       6.21e-21 FALSE                 1    1.22e-21
```

```
df_boxcox_param %>%
    group_by(stats) %>%
    ggplot(aes(x = lambdas, y = p.value)) +
        labs(x = TeX('\\lambda')) +
```

```
        facet_wrap(~ stats) +
        geom_point(size = 0.4) +
        geom_hline(yintercept = ALPHA, colour = 'red') +
        annotate(
            'text',
            label = paste0('p > ', ALPHA),
            x = -3, y = 0.065,
            colour = 'red'
        )
```



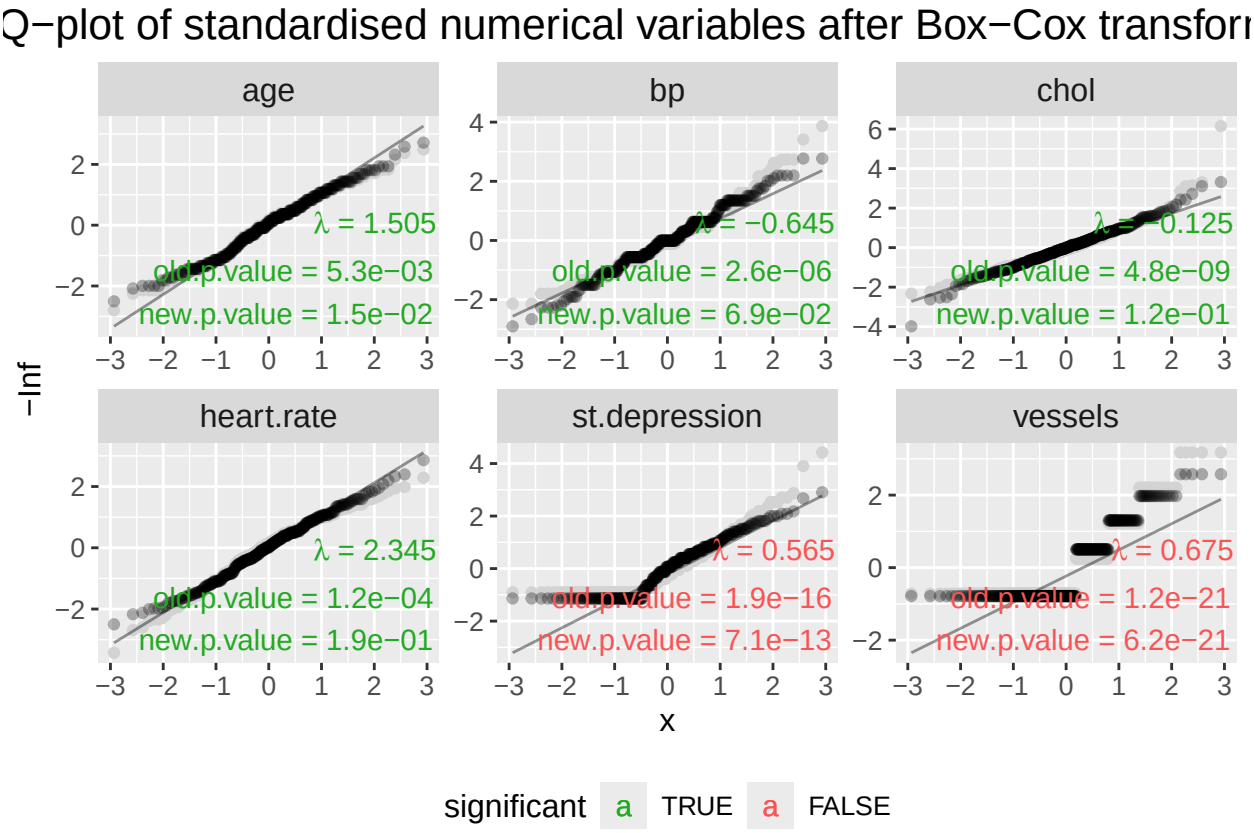

```
# df_numericals_transformed <- df %>%
#     mutate(age            = .boxcox(age,                  1.505)) %>%
#     mutate(bp             = .boxcox(bp,                  -0.645)) %>%
#     mutate(chol           = .boxcox(chol,                -0.125)) %>%
#     mutate(heart.rate     = .boxcox(heart.rate,           2.345)) %>%
#     mutate(st.depression  = .boxcox(st.depression + 0.05, 0.525)) %>%
#     mutate(vessels        = .boxcox(vessels + 0.05,       0.640)) %>%
#
#     select(all_of(NUMERICAL_VARIABLES), RESPONSE) %>%
#     pivot_longer(all_of(NUMERICAL_VARIABLES), names_to = "stats", values_to = "values")

df_numericals_transformed <- df_numericals %>%
    inner_join(select(df_boxcox_param_summary, stats, lambdas)) %>%
    group_by(stats) %>%
    mutate(
        values.scaled = scale(values),
        values.boxcox.scaled = scale(.boxcox(values, lambdas[1]))
```

```
    )
```

```
## Joining with `by = join_by(stats)`
```

```
(ggplot() +
    facet_wrap(stats ~ ., scales = "free") +
    geom_qq(
        data = df_numericals_transformed,
        mapping = aes(sample = values.scaled),
        color = 'lightgray'
    ) +
    geom_qq(
        data = df_numericals_transformed,
        mapping = aes(sample = values.boxcox.scaled),
        color = 'black',
        alpha = 0.3
    ) +
    geom_qq_line(
        data = df_numericals_transformed,
        mapping = aes(sample = values.scaled),
        alpha = 0.4
    ) +
    geom_text(
        data = df_boxcox_param_summary,
        mapping = aes(
            x = Inf,
            y = -Inf,
            label = paste0(
                TeX(
                    paste0('$\\lambda$ = ', format(lambdas, trim = T, digits = 3)),
                    output = 'character'
                )
            ),
            fontface = 'bold.italic',
            color = significant,
            hjust = 1.02,
            vjust = -4.6
        ),
        parse = TRUE
    ) +
    geom_text(
        data = df_boxcox_param_summary,
        mapping = aes(
            x = Inf,
            y = -Inf,
            label = paste0(
                paste0('old.p.value = ', format(p.value.old, trim = T, digits = 2)),
                paste0('\n'),
                paste0('new.p.value = ', format(p.value,     trim = T, digits = 2))
            ),
            color = significant,
            hjust = 1.02,
            vjust = -0.2
        )
```

```
    ) +
        scale_color_manual(
            breaks = c(TRUE, FALSE),
            values=c('#23AA23', '#F95454')
        )
) %>%
        .save_and_display(
            'QQ-plot of standardised numerical variables after Box-Cox transformation',
            '../figures/23.numerical.boxcox.qq.pdf'
        )
```



Categorical variables

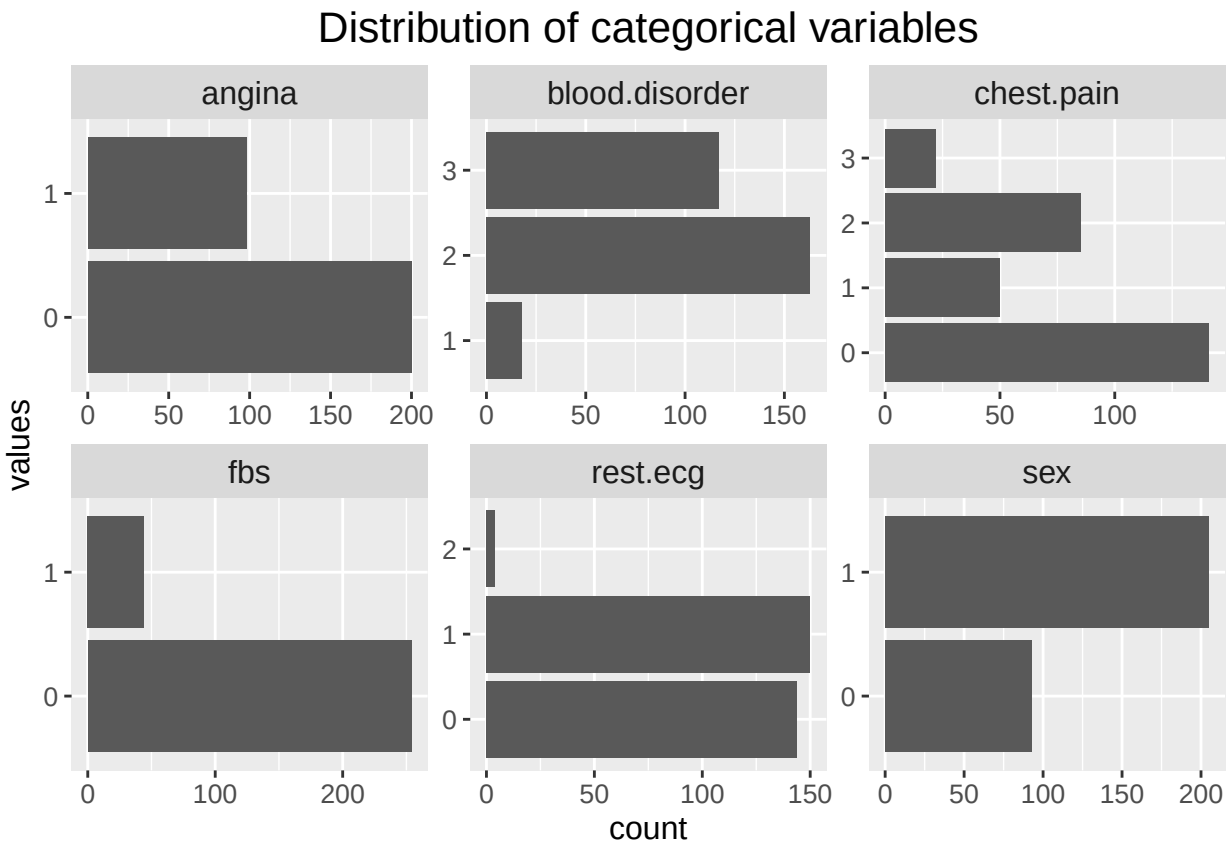
```
df %>% select(all_of(RESPONSE)) %>% pull %>% table
```

```
## .
##   0   1
## 137 161
```

```
df_categoricals <- df %>%
    select(all_of(CATEGORICAL_VARIABLES)) %>%
    pivot_longer(everything(), names_to = "stats", values_to = "values")


(df_categoricals %>%
    ggplot(aes(
```

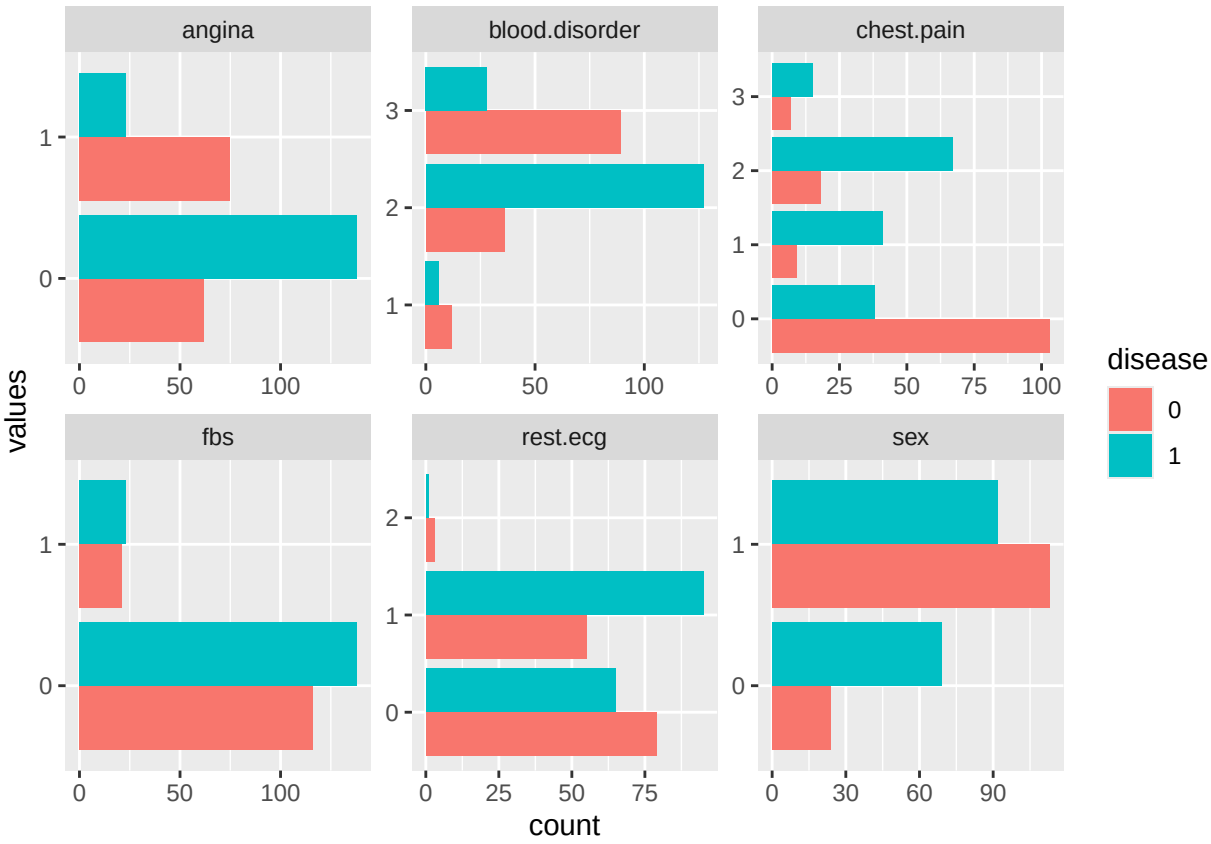
```
        x = values
    )) +
        facet_wrap(stats ~ ., nrow = 2, scales = "free") +
        geom_histogram(stat='count') +
        coord_flip()) %>%

        .save_and_display(
            'Distribution of categorical variables',
            '../figures/22.categorical.distribution.pdf'
        )
```

```
## Warning in geom_histogram(stat = "count"): Ignoring unknown parameters:
## `binwidth`, `bins`, and `pad`
```

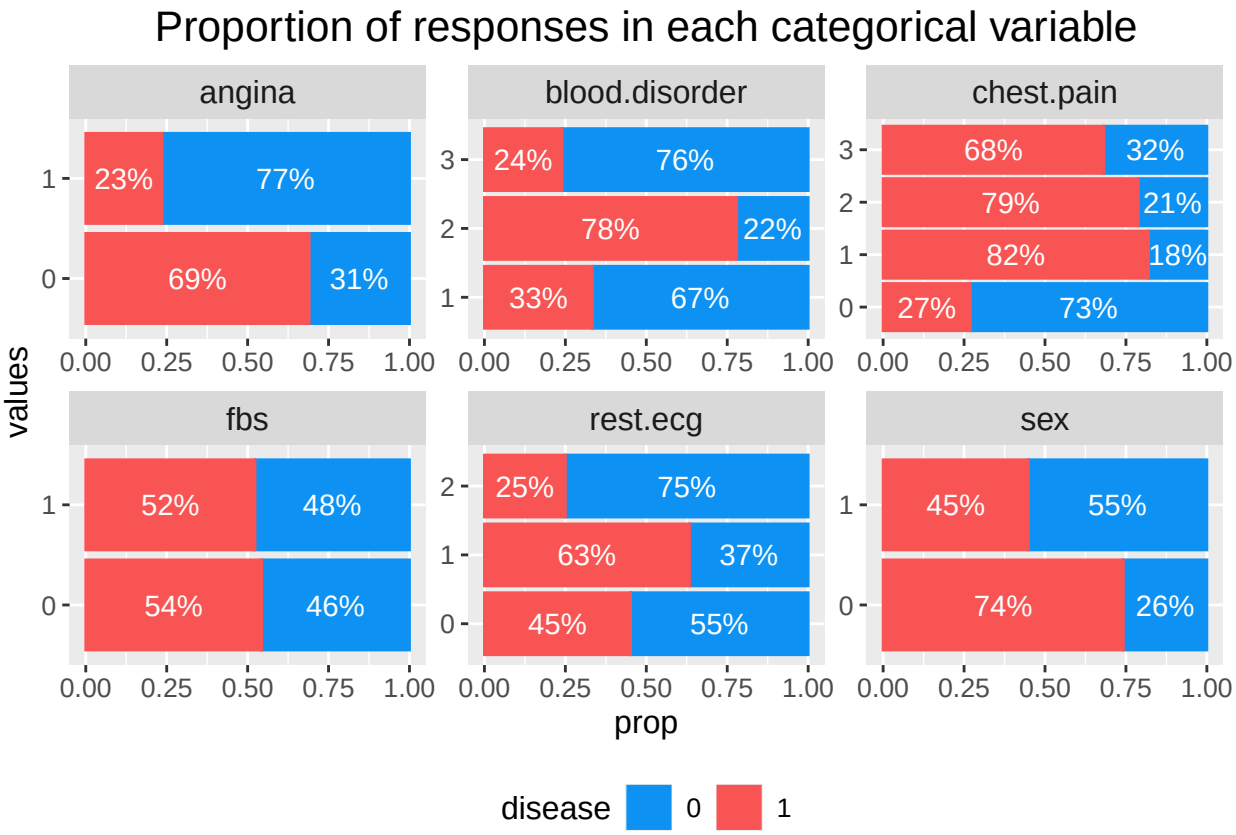



```
df_categoricals_disease <- df %>%
    select(all_of(CATEGORICAL_VARIABLES), RESPONSE) %>%
    pivot_longer(-all_of(RESPONSE), names_to = "stats", values_to = "values")

df_categoricals_disease %>%
    ggplot(aes(
        x = values, fill = disease
    )) +
        facet_wrap(stats ~ ., scales = "free") +
        geom_histogram(stat='count', position='dodge') +
        coord_flip()
```

```
## Warning in geom_histogram(stat = "count", position = "dodge"): Ignoring unknown
## parameters: `binwidth`, `bins`, and `pad`
```




```
(df_categoricals_disease %>%
    group_by(disease, stats, values) %>%
    count() %>%
    group_by(values, stats) %>%
    mutate(prop = n / sum(n)) %>%

    ggplot(aes(
        x = prop, y = values, fill = disease, color = disease
    )) +
        facet_wrap(stats ~ ., nrow = 2, scales = "free") +
        geom_col() +
        geom_text(
            aes(label = paste0(100 * round(prop, 2), '%')),
            colour = 'white',
            alpha = 1,
            position = position_stack(vjust = .5)
        )
    ) %>%

        .gg.correct.colouring %>%
        .save_and_display(
            'Proportion of responses in each categorical variable',
            '../figures/22.categorical.correlation.to.response.pdf'
        )
```




```
df_categoricals %>%
    group_by(stats) %>%
    summarise(
        fisher.p.value = fisher.test(values, df$disease)$p.value,
        chisq.p.value  =  chisq.test(values, df$disease)$p.value
    )
```

```
## Warning: There was 1 warning in `summarise()`.
## i In argument: `chisq.p.value = chisq.test(values, df$disease)$p.value`.
## i In group 5: `stats = "rest.ecg"`.
## Caused by warning in `chisq.test()`:
## ! Chi-squared approximation may be incorrect
```

```
## # A tibble: 6 x 3
##   stats          fisher.p.value chisq.p.value
##   <chr>                   <dbl>         <dbl>
## 1 angina               8.47e-14      3.21e-13
## 2 blood.disorder       5.07e-20      8.52e-19
## 3 chest.pain           2.50e-18      2.80e-17
## 4 fbs                  8.70e- 1      9.29e- 1
## 5 rest.ecg             1.89e- 3      3.76e- 3
## 6 sex                  2.79e- 6      4.66e- 6
```

```
df_categoricals %>%
    group_by(stats) %>%
    summarise(fisher.p.value = fisher.test(values, df$disease)$p.value %>% round(4))
```

```
## # A tibble: 6 x 2
##   stats          fisher.p.value
```

```
##   <chr>                  <dbl>
## 1 angina                 0
## 2 blood.disorder         0
## 3 chest.pain             0
## 4 fbs                    0.870
## 5 rest.ecg               0.0019
## 6 sex                    0
```

```
# %>%
#     knitr::kable(format = 'latex') %>%
#     writeLines()
```