Final Project: Cholangitis Analysis

Emelia Sprott

Introduction

Primary biliary cholangitis (PBC) is a chronic disease that gradually destroys the bile ducts connecting the liver and intestines. Though the exact cause is uncertain, PBC occurs when the body's immune system mistakenly attacks itself. When bile ducts are damaged, bile builds up in the liver causing progressively worsening scarring and inflammation. Eventually the scarring causes liver cirrhosis, and the liver is unable to function effectively. There is currently no cure for the disease, but some treatments can help manage symptoms and slow the progression of the disease. PBC primarily affects women and arises in individuals between the ages of 30 and 60 years old. The disease is most prevalent in northern Europe and North America, and a family history of PBC also increases the risk of developing it. In people predisposed to the disease, environmental factors including exposure to chemicals, smoking, and infections may trigger or aggravate the disease. The proportion of individuals with PBC ranges from 1.91 to 40.2 per 100,000 people, and the rate of its diagnosis ranges from 0.33 to 5.8 per 100,000 people per year.

Survival analysis is a statistical method used to analyze and predict the time until a specific event occurs, emphasizing the duration until the event of interest occurs. Widely applied in medical and health research, survival analysis is helpful for studies where the timing of an event is the focus. Another unique aspect of survival analysis is it allows for multiple different events to serve simultaneously as the event of interest. Anytime analysis is focused on the time to an event, survival analysis may be appropriate. Its most obvious application is in clinical trials, studying the length of time a patient survives with a certain medication. It could also be used to model the time it takes to find a new job after being fired, life expectancy with a particular disease, shipping and transportation time, or the lifespan of a product. Survival analysis with the cholangitis dataset will produce a model of the life expectancy of individuals with PBC, taking into account many different factors which might contribute to their health outcome.

Data and Packages for Reproducibility

```r
library(tidyverse)
library(lubridate)
library(broom)
library(GGally)
library(ggalluvial)
library(patchwork)
library(ggbeeswarm)
library(RColorBrewer)
library(leaps)
library(rpart)
library(rpart.plot)
library(randomForest)
library(knitr)
library(formatR)
```

```r
opts_chunk$set(tidy.opts = list(width.cutoff = 60), tidy = TRUE)
```

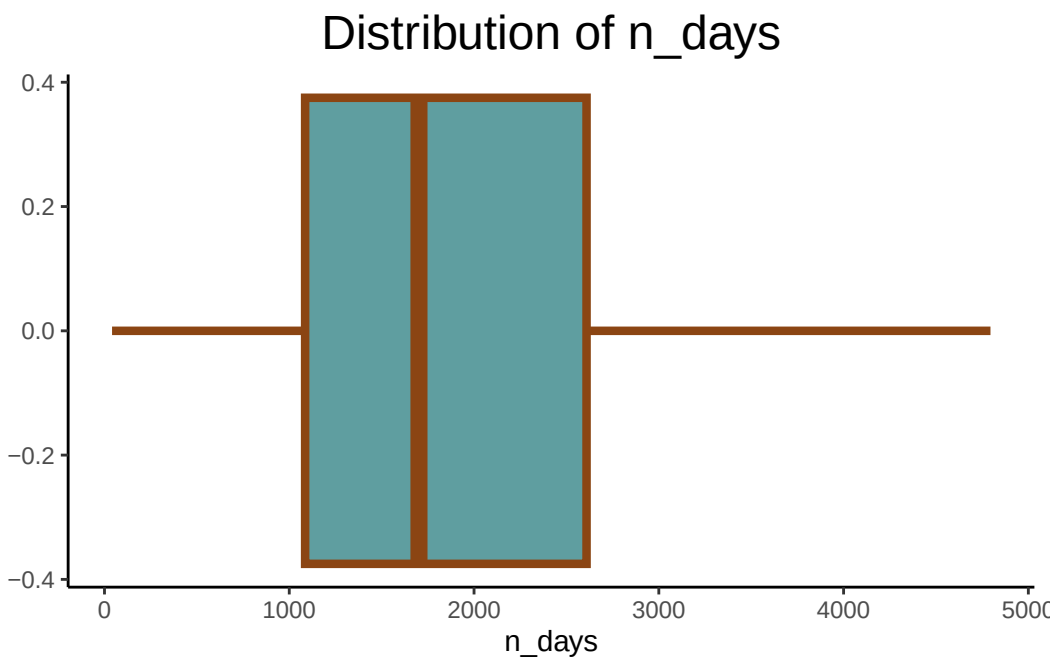
```r
# Load data
cholangitis_data <- read.csv("cholangitis.csv")
```

```r
# Convert ordinal categorical variables to factors
cholangitis_df <- cholangitis_data %>%
    mutate(drug = ifelse(is.na(drug), "None", drug)) %>%
    mutate(status = as.factor(status), drug = factor(drug, levels = c("Placebo",
        "D-penicillamine", "None")), ascites = as.factor(ascites),
        hepatomegaly = as.factor(hepatomegaly), spiders = as.factor(spiders),
        edema = as.factor(edema), stage = factor(stage, levels = rev(1L:4L)))
```

```r
# filter NA values other than drug
cholangitis_NA <- cholangitis_df %>%
    filter_all(any_vars(is.na(.)))

cholangitis <- cholangitis_df %>%
    anti_join(cholangitis_NA, by = "id") %>%
    select(-id)

# table of numeric variables
cholangitis_numeric <- cholangitis %>%
    select_if(is.numeric)
```

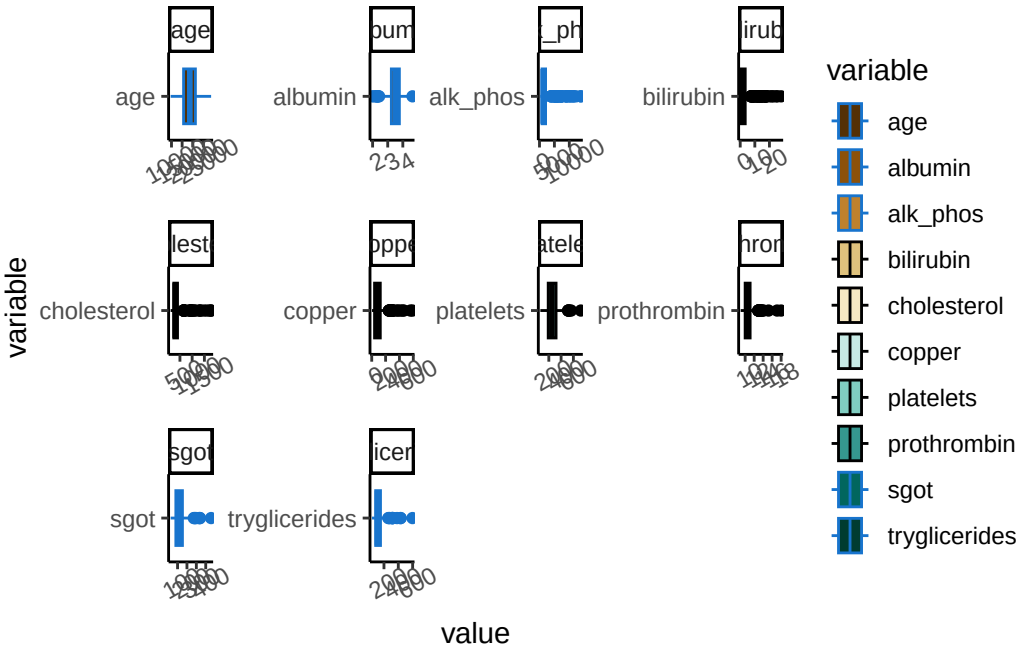
Data Structure

```
# check for outliers in response variable
cholangitis %>%
    ggplot(aes(x = n_days)) + geom_boxplot(fill = "cadetblue",
    color = "chocolate4", linewidth = 1.5) + labs(title = "Distribution of n_days") +
    theme_classic() + theme(plot.title = element_text(size = 18,
    hjust = 0.5, color = "black"))
```



I found no obvious outliers in the response variable, n_days, and its distribution is slightly right-skewed.

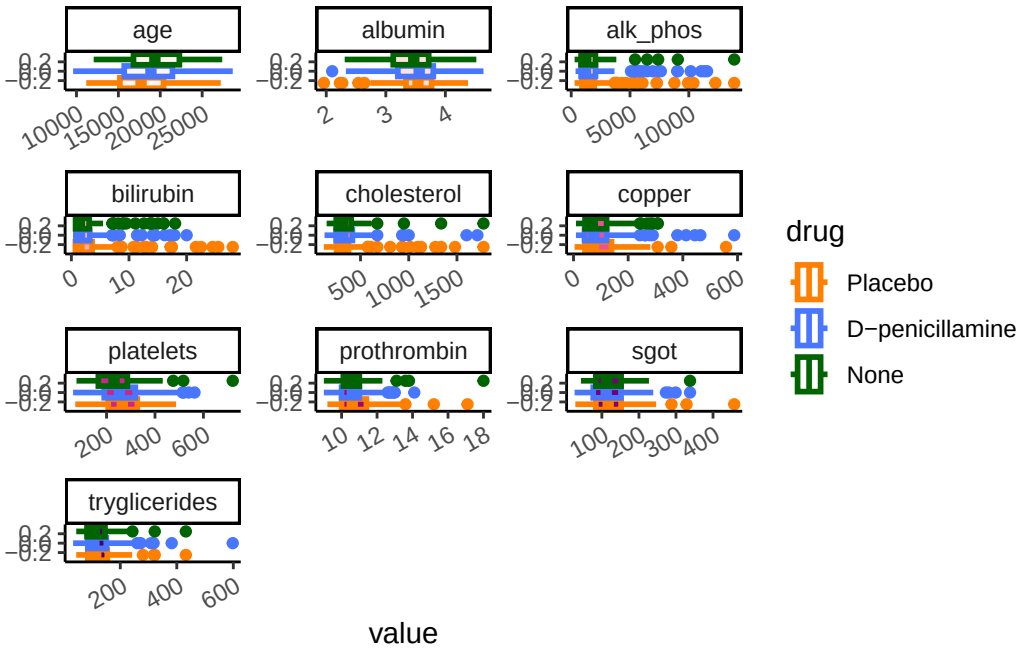
```
# outliers and distributions of numeric variables
cholangitis_numeric %>%
    select(-n_days) %>%
    gather(key = "variable", value = "value") %>%
    ggplot(aes(x = value, y = variable, fill = variable, color = variable)) +
    geom_boxplot() + scale_fill_brewer(palette = "BrBG") + scale_color_manual(values = c(r
    3), rep("black", 5), rep("dodgerblue3", 3))) + facet_wrap(~variable,
    scales = "free") + theme_classic() + theme(axis.text.x = element_text(angle = 30))
```



Age, albumin, and platelets are all roughly symmetrical, with few outliers. The other numeric explanatory variables are all right-skewed, with many outliers.

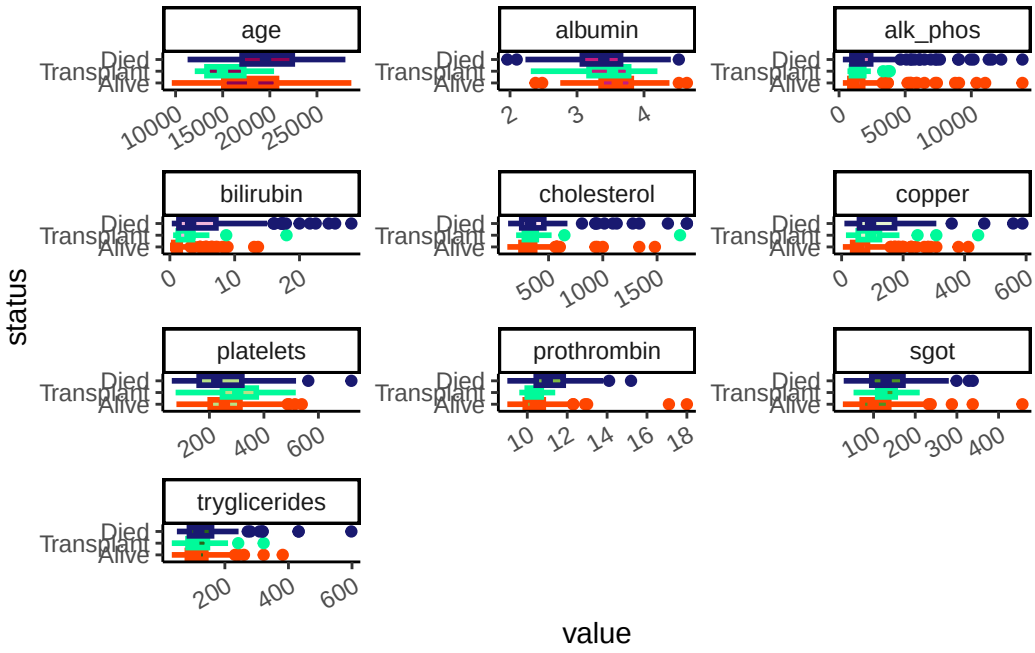
I will construct the same plot again, this time grouping by drug type.

```
cholangitis %>%
    select(-n_days) %>%
    select(drug, where(is.numeric)) %>%
    gather(key = "variable", value = "value", -drug) %>%
    ggplot(aes(x = value, fill = variable, color = drug)) + geom_boxplot(linewidth = 1) +
    scale_fill_manual(values = colorRampPalette(brewer.pal(9,
        "RdPu"))(10)) + scale_color_manual(values = c("darkorange1",
    "royalblue1", "darkgreen")) + facet_wrap(~variable, scales = "free",
    ncol = 3) + theme_classic() + theme(axis.text.x = element_text(angle = 30,
    hjust = 1)) + guides(fill = "none")
```



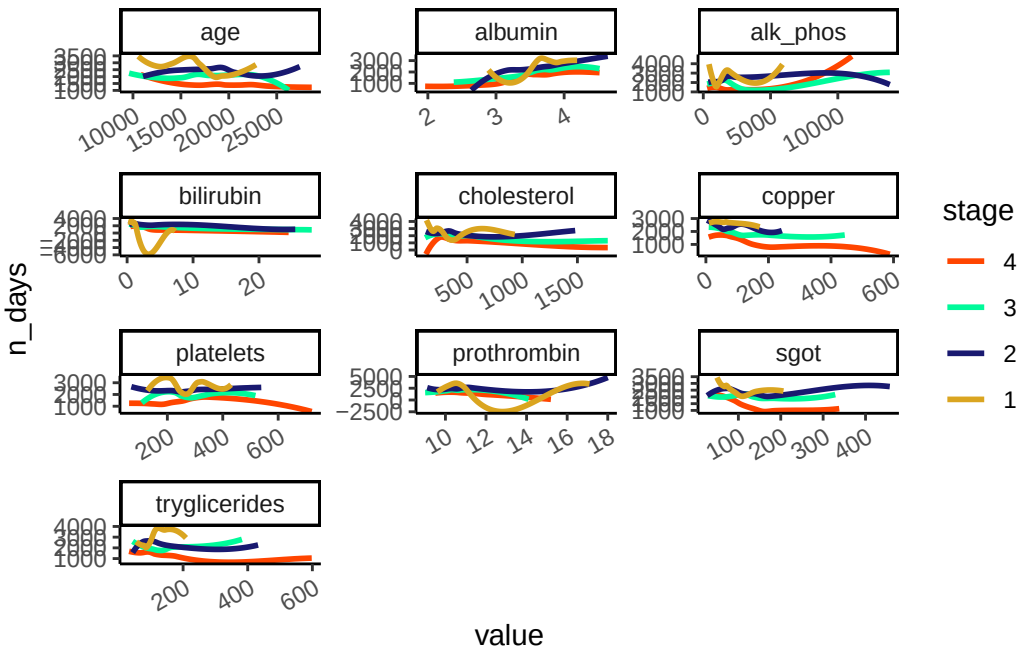
The boxplots for each numerical variable between drug type are all roughly the same shapes, but there are some observable difference between the groups. The placebo group shows more outliers at the high end of the distribution for bilirubin and cholesterol. The drug group shows a slightly lower IQR for prothrombin and copper, and a higher IQR for albumin and cholesterol.

```
# same plot, by status
cholangitis %>%
    select(-n_days) %>%
    select(status, where(is.numeric)) %>%
    gather(key = "variable", value = "value", -status) %>%
    ggplot(aes(x = value, y = status, fill = variable, color = status)) +
    geom_boxplot(linewidth = 1) + scale_fill_brewer(palette = "PiYG") +
    scale_color_manual(values = c("orangered", "mediumspringgreen",
        "midnightblue")) + scale_y_discrete(labels = list(D = "Died",
    CL = "Transplant", C = "Alive")) + facet_wrap(~variable,
    scales = "free", ncol = 3) + theme_classic() + theme(axis.text.x = element_text(angle
    hjust = 1)) + guides(fill = "none", color = "none")
```



Using the same plot, but this time grouping by status, the boxplot distributions show much more variance among levels than with the drug grouping. The most notable differences in distributions are seen in age, albumin, bilirubin, copper, prothrombin, and sgot.

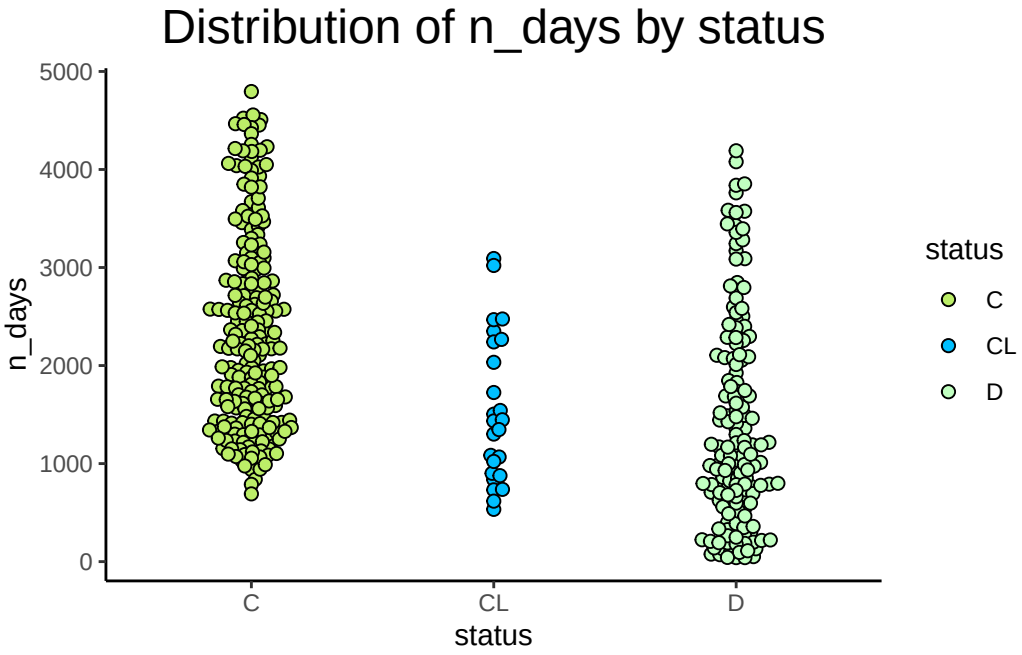
```
cholangitis %>%
    select(stage, where(is.numeric)) %>%
    gather(key = "variable", value = "value", -stage, -n_days) %>%
    ggplot(aes(x = value, y = n_days, color = stage)) + geom_smooth(formula = "y~x",
    method = "loess", se = FALSE) + scale_color_manual(values = c("orangered",
    "mediumspringgreen", "midnightblue", "goldenrod")) + facet_wrap(~variable,
    scales = "free", ncol = 3) + theme_classic() + theme(axis.text.x = element_text(angle
    hjust = 1)) + guides(fill = "none")
```



One final comparison of the relationship between n_days and the numerical variables, by stage. Notably, Stage 1 exhibits a distinct distribution from the other stages in many of the variables.

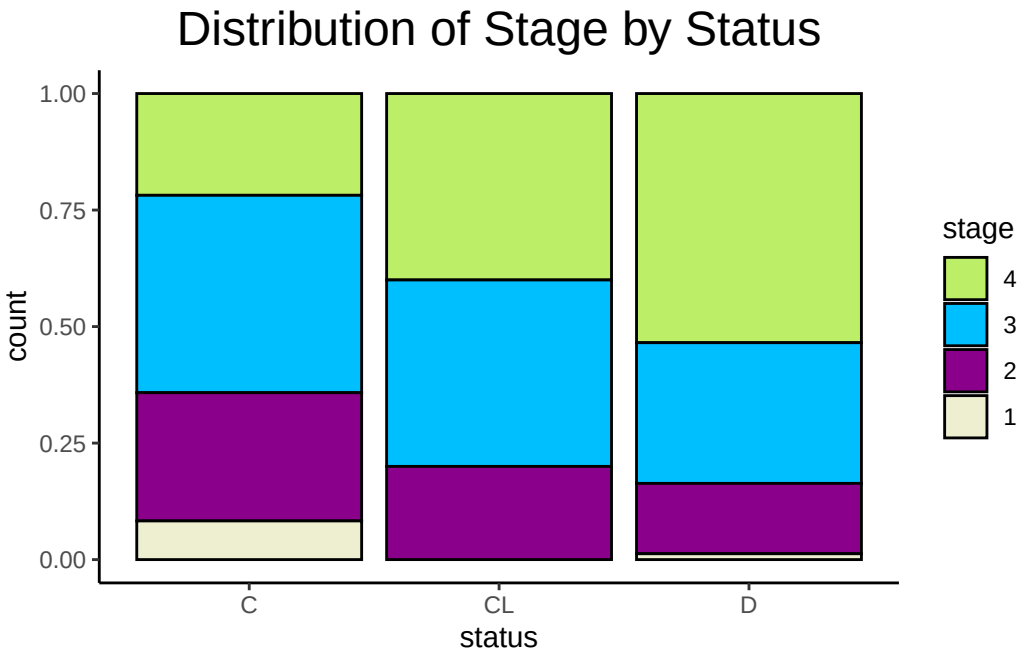
The way the study was conducted implies an inherent relationship between n_days and status, so I would like to examine their relationship first.

```
# n_days and status
cholangitis %>%
    ggplot(aes(x = status, y = n_days)) + geom_beeswarm(aes(fill = status),
    color = "black", size = 2, cex = 1.2, shape = 21) + scale_fill_manual(values = c("dark
    "deepskyblue", "darkseagreen1")) + labs(title = "Distribution of n_days by status") +
    theme_classic() + theme(plot.title = element_text(size = 18,
    hjust = 0.5))
```



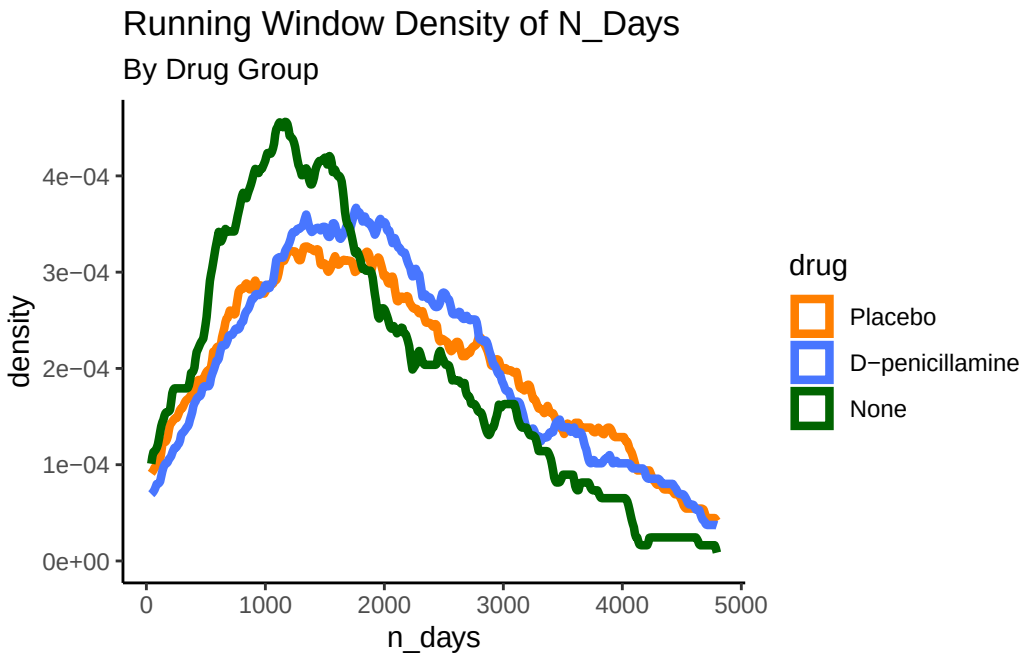
There is an observable pattern in the relationship between n_days and status level. Patients with a status of "Not Dead" tend to have higher values of n_days than patients with a status of "Dead". There are also very few observations with a status of "received liver transplant".

```
# n_days, status, and stage
cholangitis %>%
    ggplot(aes(x = status, fill = stage)) + geom_bar(color = "black",
    position = "fill") + scale_fill_manual(values = c("darkolivegreen2",
    "deepskyblue", "darkmagenta", "lightyellow2")) + labs(title = "Distribution of Stage b
    theme_classic() + theme(plot.title = element_text(size = 18,
    hjust = 0.5))
```



The standardized proportions of the stages of patients by their status at the end of the study shows that the proportion of patients at stage 4 increases as status declines. For example, of the patients who died, more than half were stage 4.

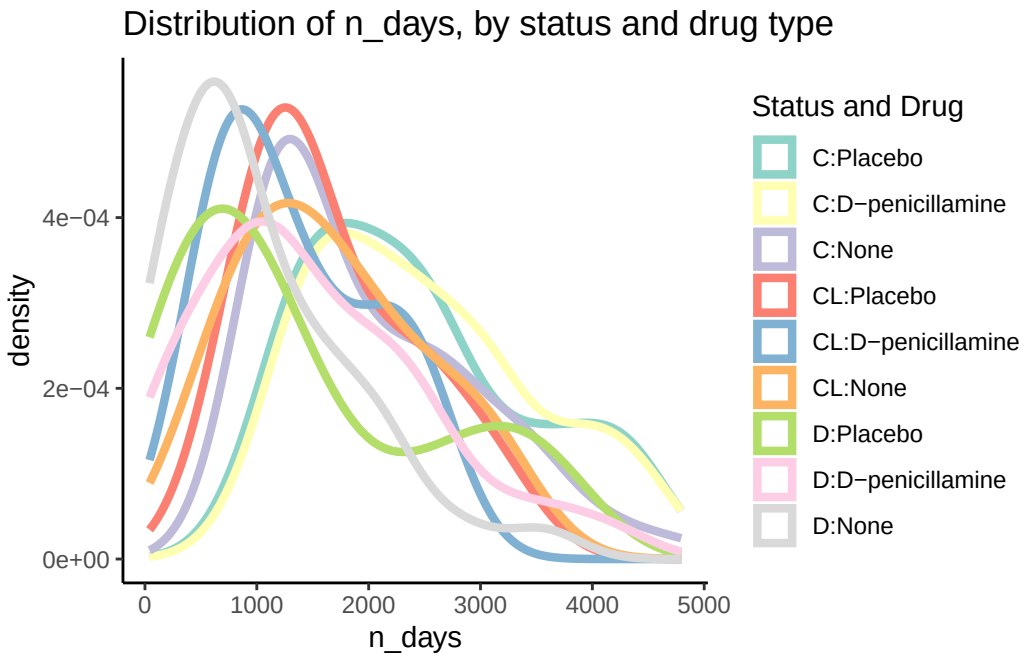
```
# response variable and drug type
cholangitis %>%
    ggplot(aes(x = n_days, color = drug)) + geom_density(linewidth = 1.5,
    kernel = "rectangular") + scale_color_manual(values = c("darkorange1",
    "royalblue1", "darkgreen")) + theme_classic() + labs(title = "Running Window Density o
    subtitle = "By Drug Group")
```



The density curves for n_days show similar shapes, although the patients who received no drug show the earliest peak and the drug group shows the latest peak.

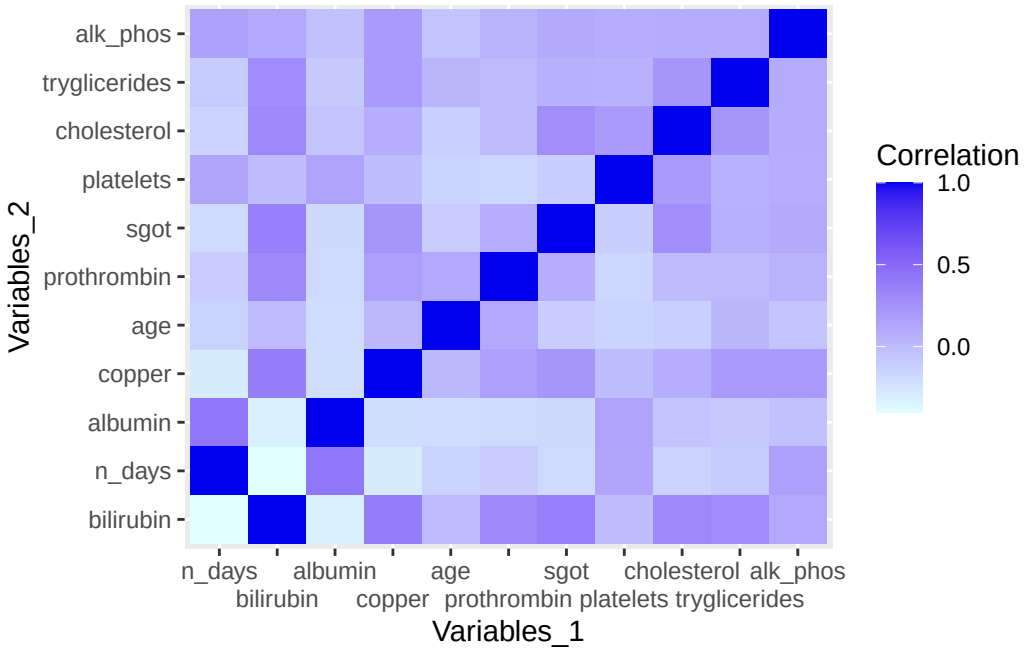
The relationship between n_days and drug type is not as clear as the relationship between n_days and status. The density curves for n_days show very similar shapes. In the context of the study, I am most interested in understanding the relationships of drug and status with n_days before I fit the model. The way n_days is determined is inherently related to the patient's status at n_days, as the patient's time in the study concludes as soon as one of the status levels occurs. This would make sense in the context of the study. Patients in the drug group might exhibit less variation in status level. I would like to examine the interaction between status and drug, and their combined relationship with n_days.

```
cholangitis %>%
    ggplot(aes(x = n_days, color = (status:drug))) + geom_density(alpha = 1,
    linewidth = 1.5) + scale_color_brewer(palette = "Set3") +
    labs(title = "Distribution of n_days, by status and drug type",
        color = "Status and Drug") + theme_classic()
```



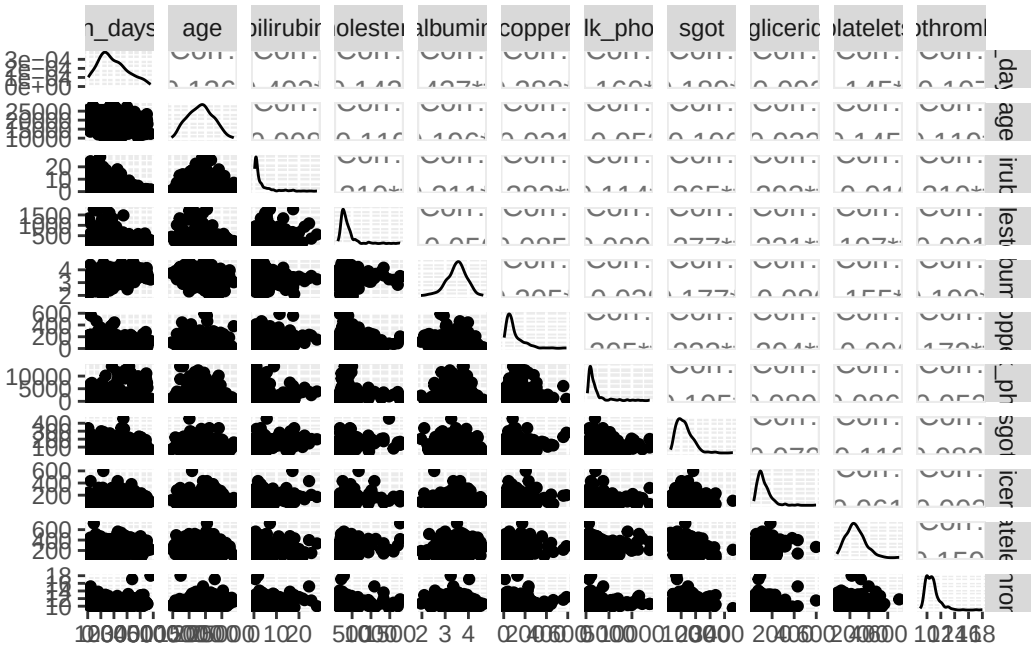
The density curves for each status type and drug type show slightly different shapes. Among the patients who died, the drug group tend to have higher n_days. The patients who received a liver transplant also show varying distributions between drug groups, although because there are few observations in the category this may be coincidental.

```
as.data.frame(cor(cholangitis_numeric)) %>%
    rownames_to_column("Variables_1") %>%
    pivot_longer(-c(Variables_1), names_to = "Variables_2", values_to = "Correlation") %>%
    arrange(Correlation) %>%
    mutate(Variables_1 = factor(Variables_1, levels = unique(Variables_1)),
        Variables_2 = factor(Variables_2, levels = unique(Variables_2))) %>%
    ggplot(mapping = aes(x = Variables_1, y = Variables_2)) +
    geom_tile(aes(fill = Correlation)) + scale_fill_gradient(low = "lightcyan",
    high = "blue2") + scale_x_discrete(guide = guide_axis(n.dodge = 2))
```



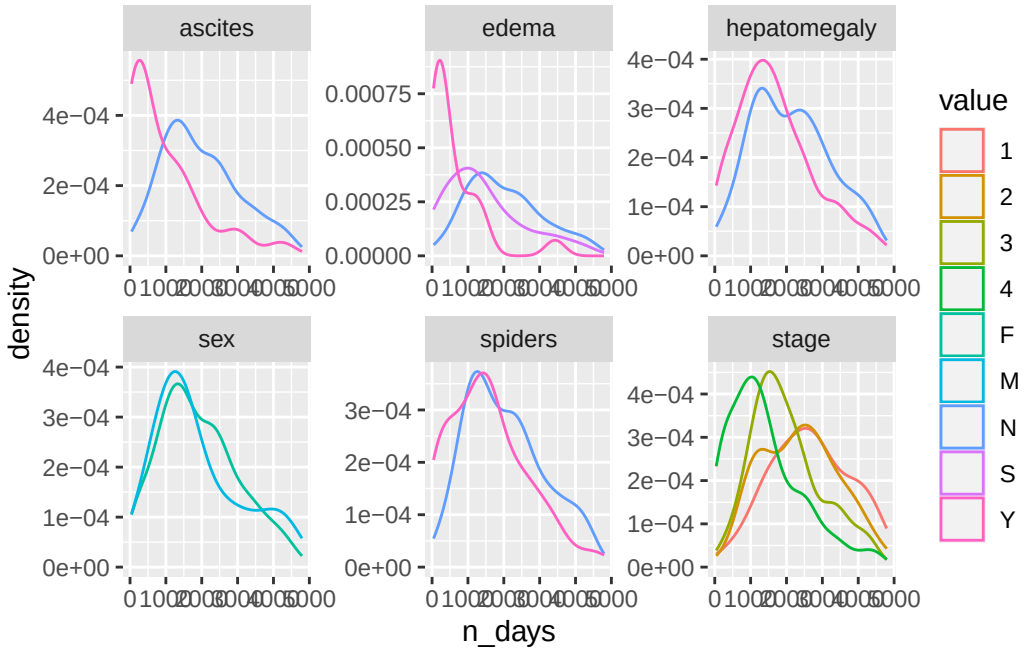
N_days has the highest level of correlation with platelets and alk_phos. It also shows a high negative correlation with albumin, bilirubin, sgot, and copper.

```
# pairs plot of numerical variables
cholangitis_numeric %>%
    ggpairs()
```



Of all the numerical variables, alburnin, copper, and alk_phos exhibit correlations of the highest magnitude. I noticed in the pairs plot n_days plotted against each of these variables creates a curved shape in the graph, which means some of these variables may need to be log transformed in the model.

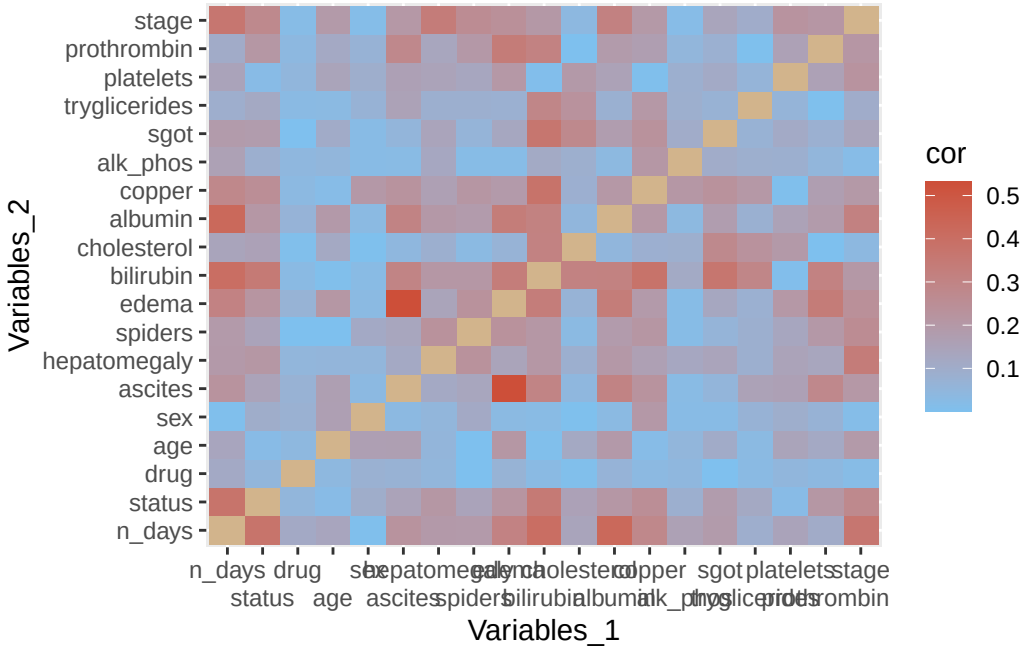
```
# other categorical variables
cholangitis %>%
    select(n_days, sex, ascites, hepatomegaly, spiders, edema,
        stage) %>%
    gather(key = "category", value = "value", -n_days) %>%
    ggplot(aes(x = n_days, color = value)) + geom_density(aes(group = interaction(category
    value))) + facet_wrap(~category, scales = "free")
```



I have already examined the relationships between drug and status with n_days, but the other categorical variables exhibit a relationship as well. The variables for edema, stage, sex, ascites, and hepatomegaly exhibit very different distributions of n_days among levels.

```
subbed_categorical <- cholangitis %>%
    mutate(status = ifelse(status == "C", 0, ifelse(status ==
        "D", 1, 2)), drug = ifelse(drug == "None", 0, ifelse(drug ==
        "Placebo", 1, 2)), sex = ifelse(sex == "M", 0, 1), ascites = ifelse(ascites ==
        "N", 0, 1), hepatomegaly = ifelse(hepatomegaly == "N",
        0, 1), spiders = ifelse(spiders == "N", 0, 1), edema = ifelse(edema ==
        "N", 0, ifelse(edema == "S", 1, 2)), stage = as.numeric(stage))

as.data.frame(cor(subbed_categorical)) %>%
    rownames_to_column("Variables_1") %>%
    pivot_longer(-c(Variables_1), names_to = "Variables_2", values_to = "Correlation") %>%
    mutate(cor = abs(Correlation)) %>%
    mutate(Variables_1 = factor(Variables_1, levels = unique(Variables_1)),
        Variables_2 = factor(Variables_2, levels = unique(Variables_2))) %>%
    arrange(desc(cor)) %>%
    mutate(cor = ifelse(Variables_1 == Variables_2, NA, cor)) %>%
    ggplot(mapping = aes(x = Variables_1, y = Variables_2)) +
    geom_tile(aes(fill = cor)) + scale_fill_gradient(low = "skyblue2",
```

```
    high = "tomato3", na.value = "tan") + scale_x_discrete(guide = guide_axis(n.dodge = 2)
```



Using encoding for categorical variables, I wanted to examine the correlation between all variables. I removed the panels for the correlation between each variable and itself. Edema and ascites show the highest correlation. N_days shows high correlations with status, bilirubin, albumin, and stage.

Modeling n_days with linear regression

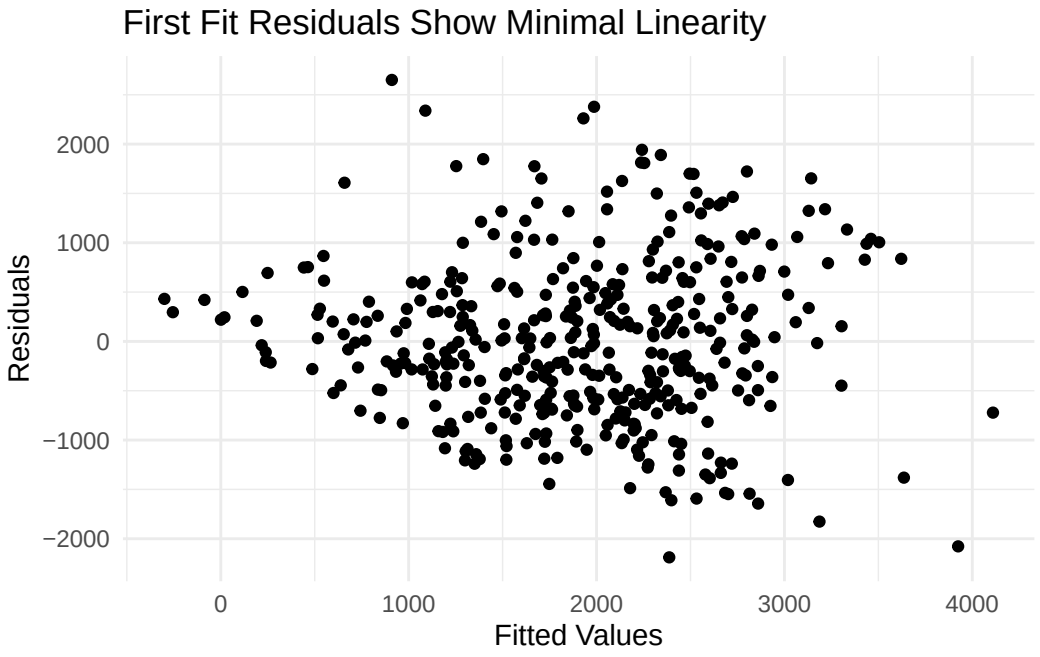
To model n_days using all explanatory variables, I first fit an MLR model without any transformations.

```
# first attempt at model
model_raw <- lm(n_days ~ ., data = cholangitis)
model_raw %>%
    tidy() %>%
    head(n = 6)
```

```
# A tibble: 6 x 5
  term                  estimate std.error statistic      p.value
```

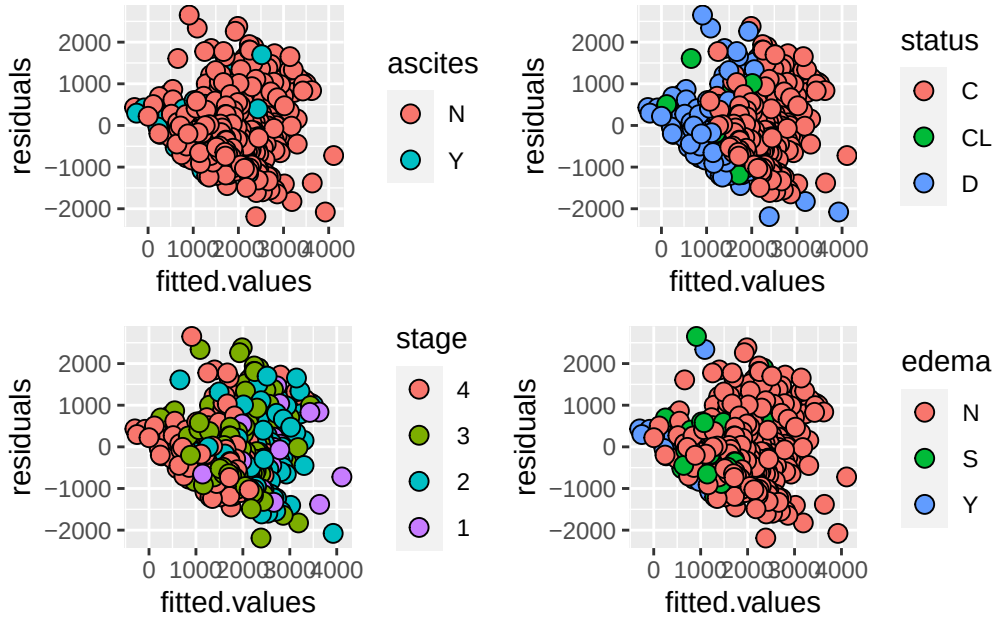
```
  <chr>                    <dbl>     <dbl>     <dbl>        <dbl>
1 (Intercept)          -1850.       745.       -2.48   0.0134
2 statusCL              -463.       188.       -2.47   0.0141
3 statusD               -610.       107.       -5.69   0.0000000255
4 drugD-penicillamine     -5.39      99.0      -0.0545 0.957
5 drugNone              -348.       111.       -3.13   0.00185
6 age                      0.00319    0.0126    0.252  0.801
```

```
residuals_1 <- cholangitis %>%
    mutate(residuals = model_raw$residuals, fitted.values = model_raw$fitted.values)

ggplot(residuals_1, aes(x = fitted.values, y = residuals)) +
    geom_point() + labs(title = "First Fit Residuals Show Minimal Linearity",
    x = "Fitted Values", y = "Residuals") + theme_minimal()
```




```
# Plotting the first fit residuals, with categorical
# groupings
ascites_resid <- ggplot(residuals_1, aes(x = fitted.values, y = residuals)) +
    geom_point(aes(fill = ascites), color = "black", shape = 21,
        size = 3)
status_resid <- ggplot(residuals_1, aes(x = fitted.values, y = residuals)) +
```

```
    geom_point(aes(fill = status), color = "black", shape = 21,
        size = 3)
stage_resid <- ggplot(residuals_1, aes(x = fitted.values, y = residuals)) +
    geom_point(aes(fill = stage), color = "black", shape = 21,
        size = 3)
edema_resid <- ggplot(residuals_1, aes(x = fitted.values, y = residuals)) +
    geom_point(aes(fill = edema), color = "black", shape = 21,
        size = 3)
(ascites_resid + status_resid)/(stage_resid + edema_resid)
```



When the residuals of the first attempted model are plotted against its fitted values, it's clear this model is not appropriate for the data. The residuals show the most variance for middle values of n_days. Each of the categorical variable groupings adds a semi-distinguishable pattern to the residuals plot, with the most distinct pattern coming from status.

In my EDA, I noticed:
* age, albumin, bilirubin, copper, prothrombin, and sgot have varying distributions of among status levels
* Ascites and edema have a strong positive correlation
* Ascites shows a difference in density distributions of n_days among its groups
* The "Yes" edema group shows a different density distribution of n_days than the other groups

* Stage 1 and Stage 2 shows different density distributions of n_days than Stage 3 and Stage 4.

Encoding the model will make it easier to add specific interaction terms, as some combinations of categorical variables do not exist in the dataset. I will also log transform the input variables that showed the most skew in EDA.

```
encoded_chol <- cholangitis %>%
    mutate(ascitesY = ifelse(ascites == "Y", 1, 0), statusCL = ifelse(status ==
        "CL", 1, 0), statusD = ifelse(status == "D", 1, 0), stage3 = ifelse(stage ==
        3L, 1, 0), stage2 = ifelse(stage == 2L, 1, 0), stage1 = ifelse(stage ==
        1L, 1, 0), edemaS = ifelse(edema == "S", 1, 0), edemaY = ifelse(edema ==
        "Y", 1, 0), sexM = ifelse(sex == "M", 1, 0), drugPlacebo = ifelse(drug ==
        "Placebo", 1, 0), drugD_penicillamine = ifelse(drug ==
        "D-penicillamine", 1, 0), hepatomegalyY = ifelse(hepatomegaly ==
        "Y", 1, 0), spidersY = ifelse(spiders == "Y", 1, 0)) %>%
    select(-c(ascites, status, stage, edema, sex, drug, hepatomegaly,
        spiders)) %>%
    mutate_at(vars(cholesterol, alk_phos, copper, age), log)
```

Based on the estimates and p-values in this helper model, I determined the most relevant interaction terms to include.

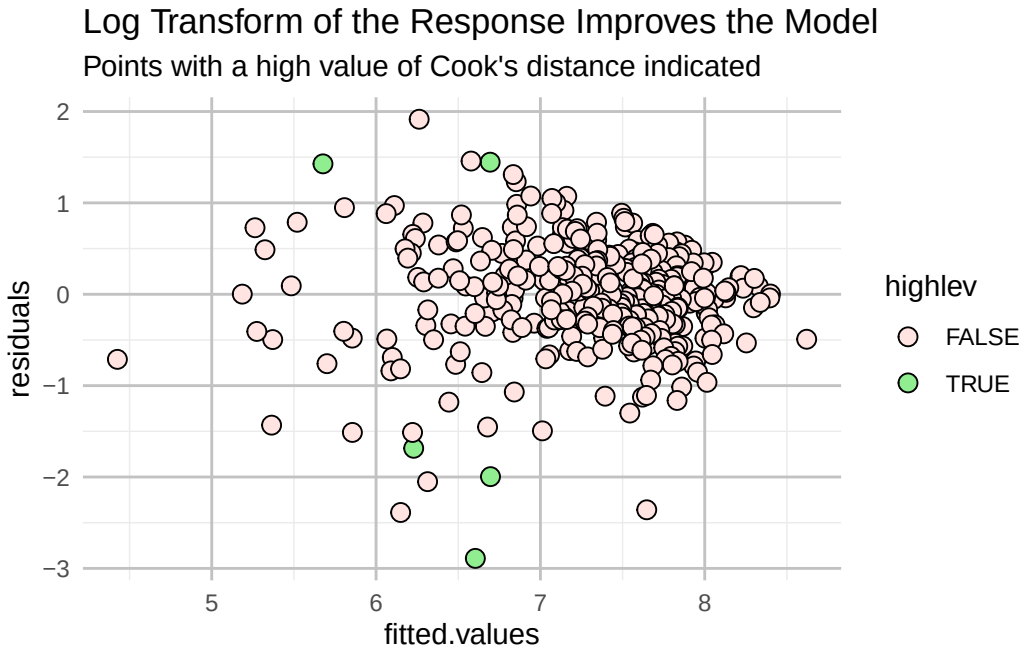
```
lm(n_days ~ .:ascites + .:status + .:stage + .:edema + ., data = cholangitis)
```

```
interaction_model <- lm(log(n_days) ~ edemaY:albumin + statusD:bilirubin +
    statusD:alk_phos + statusD:prothrombin + statusD:stage3 +
    statusD:stage1 + age:stage1 + spidersY:stage2 + spidersY:stage1 +
    sexM:edemaS + hepatomegalyY:edemaS + ., data = encoded_chol)

interaction_model %>%
    tidy() %>%
    head(n = 6)
```

```
# A tibble: 6 x 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)  2.73       1.99       1.37  0.172
2 age          0.141      0.169      0.837 0.403
3 bilirubin   -0.00491    0.0189    -0.260 0.795
4 cholesterol -0.0347     0.0834    -0.416 0.678
5 albumin      0.316      0.0850     3.72  0.000230
```

```
6 copper       -0.0919     0.0440    -2.09  0.0372
```

```
residuals_2 <- encoded_chol %>%
    mutate(residuals = residuals(interaction_model), fitted.values = fitted.values(interac
        cooks.distance = cooks.distance(interaction_model), highlev = ifelse(cooks.distanc
            0.04, TRUE, FALSE))

ggplot(residuals_2, aes(x = fitted.values, y = residuals)) +
    geom_point(aes(fill = highlev), size = 3, color = "black",
        shape = 21) + labs(title = "Log Transform of the Response Improves the Model",
    subtitle = "Points with a high value of Cook's distance indicated",
    color = "High Cook's Distance Value") + scale_fill_manual(values = c(`TRUE` = "lightgr
    `FALSE` = "mistyrose1")) + theme_minimal() + theme(panel.grid.major = element_line(col
    linewidth = 0.5))
```



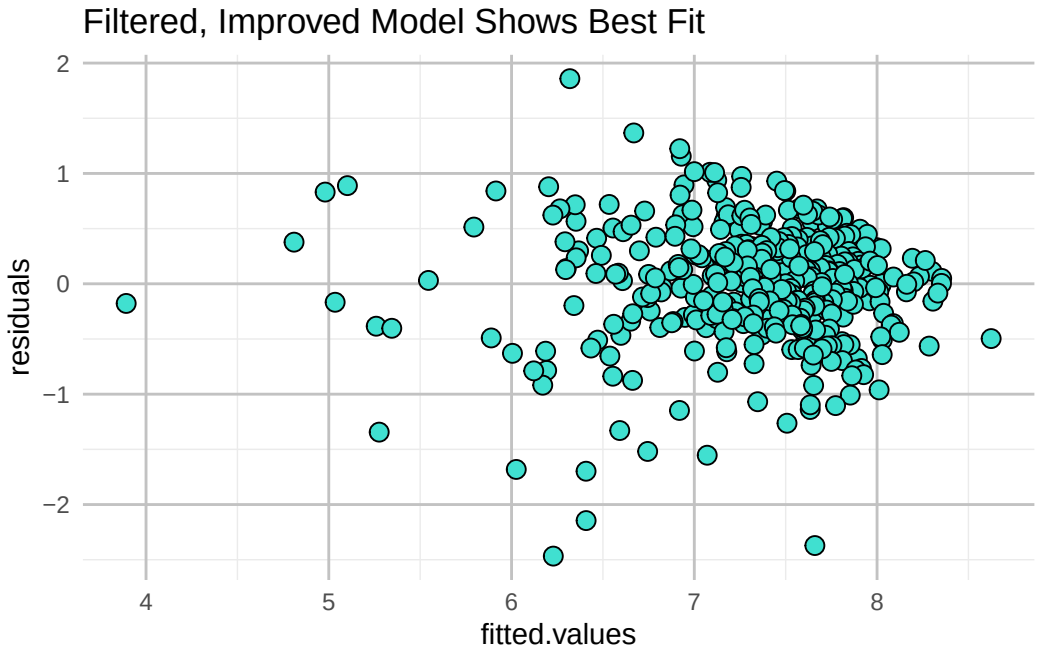
The model was improved with the interaction terms and log transformations, although the linearity and homoskedasticity of the residuals are still not ideal. I will filter the points with a high value of Cook's distance, then refit the model to see if this improves the model.

```
filtered_encoded_chol <- encoded_chol %>%
    mutate(highlev = ifelse(cooks.distance(interaction_model) >
```

```
        0.04, TRUE, FALSE)) %>%
    filter(highlev == FALSE) %>%
    select(-highlev)

filtered_model <- lm(log(n_days) ~ edemaY:albumin + statusD:bilirubin +
    statusD:alk_phos + statusD:prothrombin + statusD:stage3 +
    stage1:bilirubin + age:stage1 + spidersY:stage2 + spidersY:stage1 +
    sexM:edemaS + hepatomegalyY:edemaS + ., data = filtered_encoded_chol)

summary_filtered_model <- summary(filtered_model)
summary_filtered_model$adj.r.squared
```

```
[1] 0.5300354
```

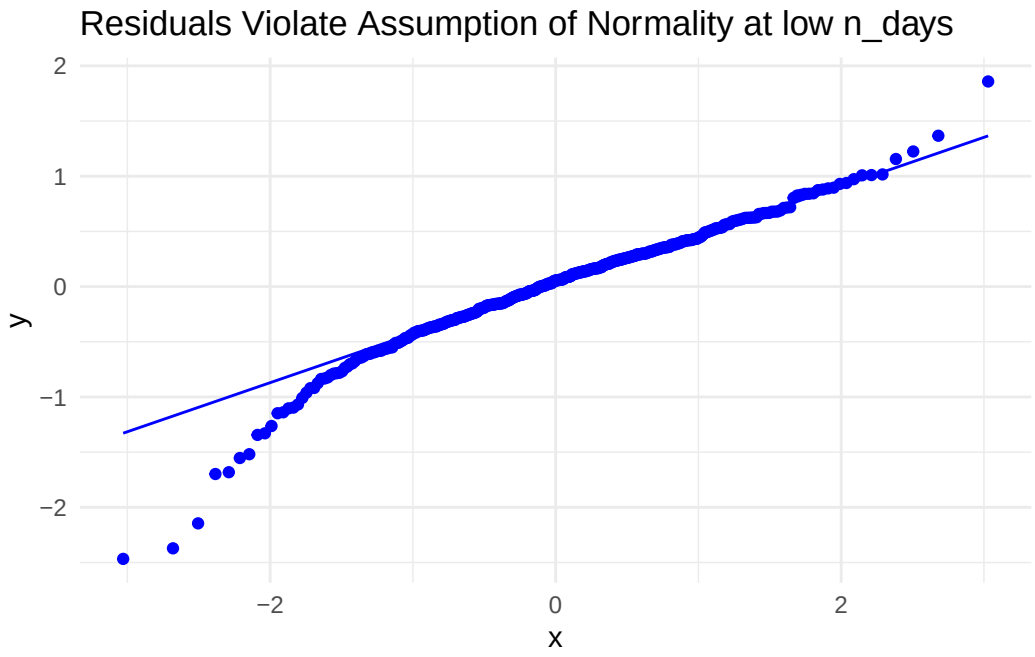
```
data.frame(list(residuals = residuals(filtered_model), fitted.values = fitted.values(filte
    ggplot(aes(x = fitted.values, y = residuals)) + geom_point(fill = "turquoise",
    color = "black", shape = 21, size = 3) + labs(title = "Filtered, Improved Model Shows
    theme_minimal() + theme(panel.grid.major = element_line(color = "gray76",
    linewidth = 0.5))
```



The model shows a better fit for the data, with a much higher R-squared value. The model is

still not ideal, but it is the best representation of the data using all of the variables. The tails in the middle values are softened by the filtering and the transformation, without eliminating excessive data points.

```
filtered_encoded_chol %>%
    mutate(residuals = residuals(filtered_model)) %>%
    ggplot(aes(sample = residuals)) + stat_qq(color = "blue") +
    stat_qq_line(color = "blue") + labs(title = "Residuals Violate Assumption of Normality
    theme_minimal()
```



The assumption of normality of the residuals is not met for the lowest values of n_days, but that makes sense as the data has less of those values.

```
drugD_raw <- model_raw %>%
    tidy() %>%
    select(term, estimate, p.value) %>%
    filter(term == "drugD-penicillamine")
drugD_interaction <- interaction_model %>%
    tidy() %>%
    select(term, estimate, p.value) %>%
    filter(term == "drugD_penicillamine")
```

```r
drugD_filtered <- filtered_model %>%
    tidy() %>%
    select(term, estimate, p.value) %>%
    filter(term == "drugD_penicillamine")
rbind(drugD_raw, drugD_interaction, drugD_filtered)
```

```
# A tibble: 3 x 3
  term                estimate  p.value
  <chr>                  <dbl>    <dbl>
1 drugD-penicillamine   -5.39  0.957
2 drugD_penicillamine    0.306 0.000215
3 drugD_penicillamine    0.260 0.000746
```

To evaluate the drug's relationship to n_days, I compare the three t-tests performed when creating each model. Both refined models show a positive coefficient estimate for drugD-penicillamine.

Under the null hypothesis of the t-test, the value of the drugD-penicillamine coefficient is 0. With two p-values very close to 0, this null hypothesis is rejected. Additionally, in the model that does not reject the null hypothesis, the coefficient given to drugD-penicillamine is negative. The coefficient of drugD-penicillamine is thus non-zero. Therefore, we can conclude that the drug is relevant to this model.

Model Accuracy with Cross Validation

With so many variables, as well as a relatively small dataset, I split the data into a testing and training set using 5-fold cross-validation. Many interactions of variables are not shown frequently in the data, so a minimal number of folds will make sure the testing set is large enough to be modeled.

```r
# create 5 folds in the data set
set.seed(131)

k <- 5
fold_vector <- cut(1:nrow(filtered_encoded_chol), breaks = k,
    labels = FALSE)

random_folds <- sample(x = fold_vector, size = nrow(filtered_encoded_chol),
    replace = FALSE)

filtered_chol <- filtered_encoded_chol %>%
```

```
    mutate(folds = random_folds)
```

Stepwise variable selection

```
# model on all data to obtain formulas for iteration
subset_model <- regsubsets(log(n_days) ~ edemaY:albumin + statusD:bilirubin +
    statusD:alk_phos + statusD:prothrombin + statusD:stage3 +
    stage1:bilirubin + age:stage1 + spidersY:stage2 + spidersY:stage1 +
    sexM:edemaS + hepatomegalyY:edemaS + ., data = filtered_encoded_chol,
    method = "forward", nvmax = 34)

subset_summary <- summary(subset_model)
```

```
# obtain all terms included in model
coef_mat <- subset_summary$which
# get formulas from helper model
formulas <- list()
for (i in 1:29) {
    terms <- data.frame(list(variable = names(coef_mat[i, ]),
        included = coef_mat[i, ]), row.names = NULL)
    variables <- terms %>%
        filter(variable != "(Intercept)", included == TRUE) %>%
        pull(variable)
    input <- paste(variables, collapse = " + ")
    formula <- paste0("log(n_days) ~ ", input)
    formulas[i] <- formula
}
```

```
set.seed(131)
# function to perform cross validation on one fold with one
# selected formula
ONE_CV_FOLD <- function(fold_number, formula, model_number) {
    chol_train <- filtered_chol %>%
        filter(folds != fold_number)

    chol_test <- filtered_chol %>%
        filter(folds == fold_number)

    cv_linear_model <- do.call(what = "lm", args = list(formula = as.formula(formula[[mode
        data = quote(chol_train)))
```

```
    cv_predictions <- predict(object = cv_linear_model, newdata = chol_test)

    observations <- chol_test %>%
        select(n_days) %>%
        pull()

    RMSE <- sqrt(mean((cv_predictions - log(observations))^2))

    return(RMSE)
}
```

```
rmses <- list()
for (i in 1:29) {
    # iterate through each formula and each fold
    rmse <- mean(sapply(unique(fold_vector), FUN = ONE_CV_FOLD,
        model_number = i, formula = formulas))
    # calculate average RMSE for each formula
    rmses[i] <- rmse
}

print(formulas[[which.min(rmses)]])
```

```
[1] "log(n_days) ~ albumin + copper + alk_phos + prothrombin + statusCL + stage2 + edemaY + c
```

```
which.min(rmses)
```

```
[1] 15
```

```
rmses[which.min(rmses)]
```

```
[[1]]
[1] 0.5656368
```

Using stepwise variable selection, the best model is the one that includes 15 variables and follows the returned formula.

Regression tree

The interaction terms will create an error in the regression tree, so I will use a simplified version of the model while fitting it.

```
set.seed(456)

k <- 5

cholangitis_tree <- cholangitis %>%
    mutate_at(vars(cholesterol, alk_phos, copper, age), log)

fold_vector_tree <- cut(1:nrow(cholangitis_tree), breaks = k,
    labels = FALSE)

random_folds_tree <- sample(x = fold_vector_tree, size = nrow(cholangitis_tree),
    replace = FALSE)

filtered_cholangitis <- cholangitis_tree %>%
    mutate(folds = random_folds_tree)
```

```
set.seed(131)

avg_rmse <- numeric()

# iterate through different alpha values
for (alpha in seq(0, 0.08, by = 0.001)) {

    rmse <- numeric()

    for (fold in 1:5) {
        # training and testing sets for the current fold
        training_data <- filtered_cholangitis %>%
            filter(folds != fold) %>%
            select(-folds)
        testing_data <- filtered_cholangitis %>%
            filter(folds == fold) %>%
            select(-folds)

        # fit a tree to the training set
        tree_model <- rpart(log(n_days) ~ ., data = training_data,
            control = list(minsplit = 5), cp = 0)
```

```
        # prune to the current alpha level
        pruned_tree <- prune(tree_model, cp = alpha)

        # use the subtree to predict on test set
        predictions <- predict(pruned_tree, testing_data)

        # calculate rmse
        rmse[fold] <- sqrt(mean((log(testing_data$n_days) - predictions)^2))
    }

    # average rmse for all folds at current alpha level
    avg_rmse <- c(avg_rmse, mean(rmse))
}

# alpha level that minimizes average rmse
optimal_alpha <- seq(0, 0.08, by = 0.001)[which.min(avg_rmse)]

optimal_alpha
```

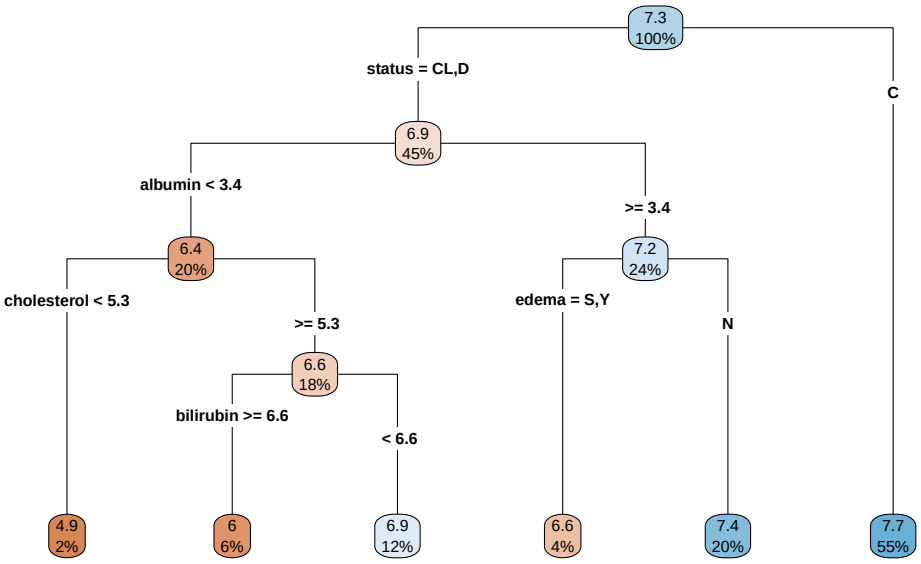
```
[1] 0.018
```

```
avg_rmse[which.min(avg_rmse)]
```

```
[1] 0.7198657
```

Using 5-fold cross validation, the best pruned subtree has an alpha of 0.018 and an average RMSE of 0.7198657.

```
# examine the tree with all of the data
large_tree <- rpart(log(n_days) ~ ., data = cholangitis_tree,
    cp = optimal_alpha)

rpart.plot(large_tree, type = 4, box.palette = "BnBu")
```




```
rpart.rules(large_tree)
```

```
log(n_days)
        4.9 when status is CL or D & albumin <  3.4 & cholesterol <  5.3
        6.0 when status is CL or D & albumin <  3.4 & cholesterol >= 5.3 & bilirubin >= 6.6
        6.6 when status is CL or D & albumin >= 3.4
        6.9 when status is CL or D & albumin <  3.4 & cholesterol >= 5.3 & bilirubin <  6.6
        7.4 when status is CL or D & albumin >= 3.4
        7.7 when status is       C
```

The regression tree, at the optimal alpha level, considers status, albumin, cholesterol, bilirubin, and edema. It groups status by "C", and not "C", and groups edema by "N" and not "N".

Random Forest

```
set.seed(131)
avg_rmse_rf <- numeric()

# define specific values for mtry
mtry_values <- seq(2, 10, by = 1)
```

```
# iterate over different mtry values
for (mtry in mtry_values) {

    rmse_rf <- c()

    for (fold in 1:5) {
        training_data <- filtered_cholangitis %>%
            filter(folds != fold) %>%
            select(-folds)
        testing_data <- filtered_cholangitis %>%
            filter(folds == fold) %>%
            select(-folds)

        rf_model <- randomForest(log(n_days) ~ ., data = training_data,
            ntree = 200, mtry = mtry)

        predictions_rf <- predict(object = rf_model, newdata = testing_data)

        rmse_rf[fold] <- sqrt(mean((log(testing_data$n_days) -
            predictions_rf)^2))
    }

    avg_rmse_rf <- c(avg_rmse_rf, mean(rmse_rf))
}

optimal_mtry_rf <- mtry_values[which.min(avg_rmse_rf)]
optimal_mtry_rf
```

```
[1] 5
```

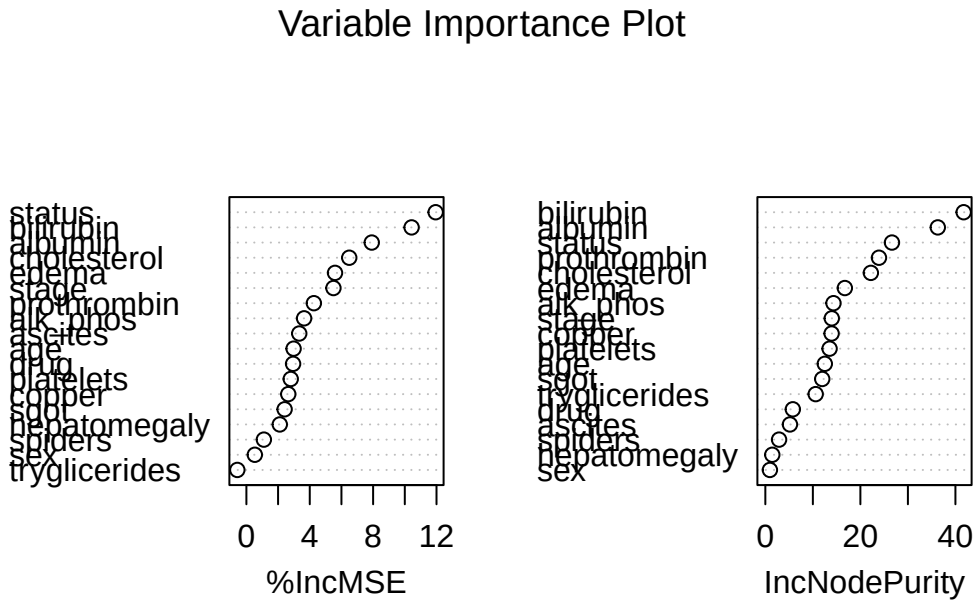
```
avg_rmse_rf[which.min(avg_rmse_rf)]
```

```
[1] 0.617292
```

Using 5-fold cross-validation, the optimal number of variables to consider at each split is 5. This random forest has a cross-validated average RMSE of 0.617292 when mtry = 5.

```
# fit forest to all data to examine
large_rf <- randomForest(log(n_days) ~ ., data = cholangitis_tree,
    ntree = 200, mtry = 5, importance = TRUE)
```

```r
varImpPlot(large_rf, main = "Variable Importance Plot", pch = 21)
```



Model comparison

Model	Linear Regression	Regression Tree	Random Forest
5-fold average RMSE	0.565636801286259	0.7198657	0.617292

Final Comments

Sources:

Clark, T. G., Bradburn, M. J., Love, S. B., & Altman, D. G. (2003). Survival analysis part I: Basic concepts and first analyses. British Journal of Cancer, 89(2), 232–238. https://doi.org/10.1038/sj.bjc.6601118

Hepatobiliary & Pancreatic Surgery - Primary Biliary Cirrhosis. (n.d.). Hpbsurgery.ucsf.edu. Retrieved December 3, 2023, from https://hpbsurgery.ucsf.edu/conditions–procedures/primary-biliary-cirrhosis.aspx

(n.d.). Primary Biliary Cholangitis [Review of Primary Biliary Cholangitis]. Orphanet. https://www.orpha.net/consor/cgi-bin/OC_Exp.php?Lng=GB&Expert=186#:~:text=Primary%20biliary%20

Primary Biliary Cirrhosis | Conditions and Treatments | Center for Liver Disease & Transplantation | Columbia University Department of Surgery. (n.d.). Columbiasurgery.org. https://columbiasurgery.org/conditions-and-treatments/primary-biliary-cirrhosis