Task 1

Use your data analysis knowledge to date, to conduct an Exploratory Data Analysis (EDA) for fitting Logistic Regression models to predict the PURCHASE decision.   Some suggestions for things that you could do are:

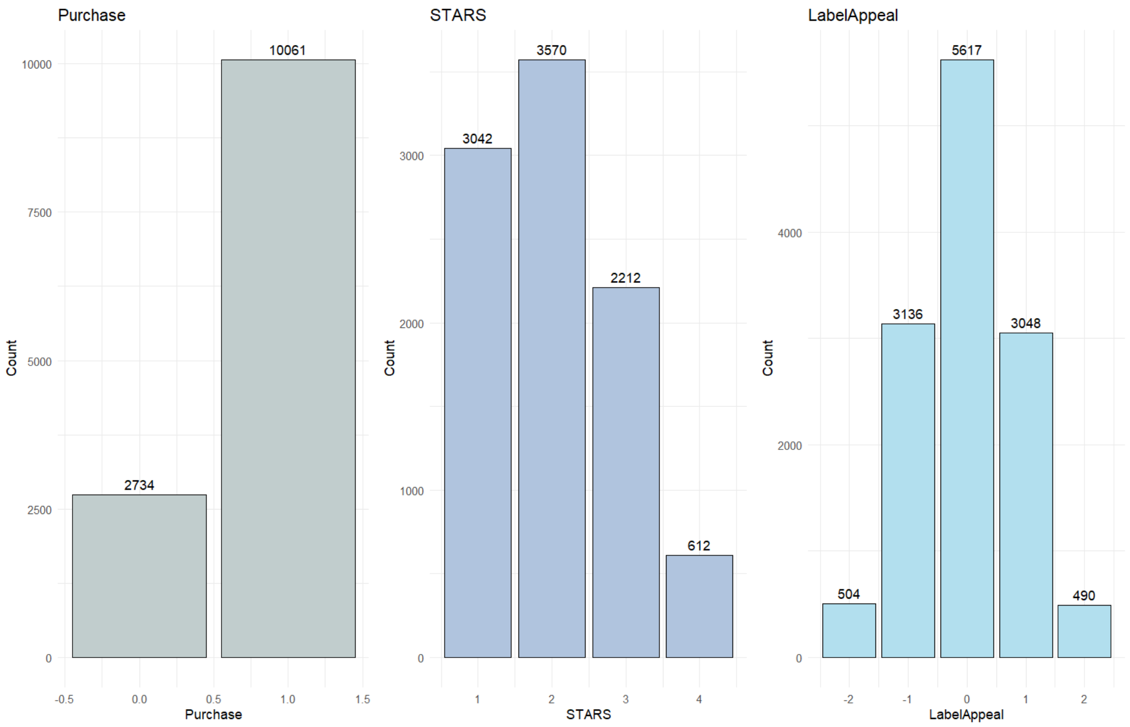
- Histograms for each continuous variable.
- Means, standard deviations, minimum, maximum, median for all continuous variables.

The wines dataframe has 17 columns and 12795 rows. The first columns in INDEX, which cannot be used in building models, it has been dropped.

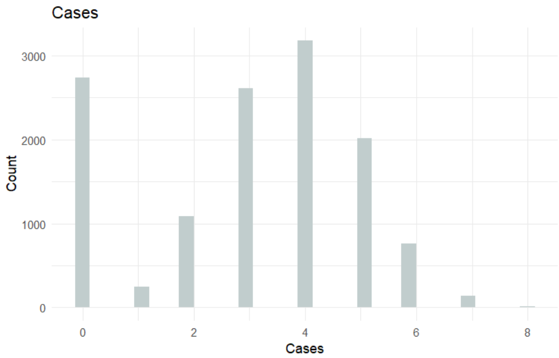
We assume that if the unique number of values in a numeric column is above a certain threshold (greater than 5 levels), consider it continuous. Otherwise, it's discrete.

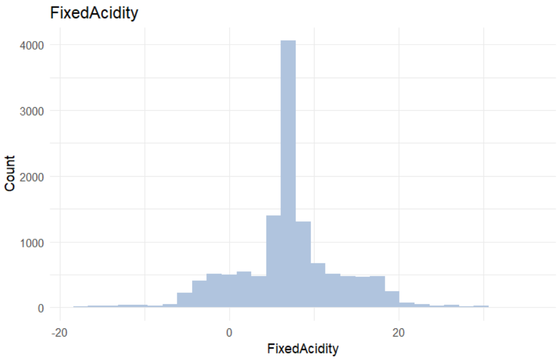
```
> # Print the continuous and discrete variables
> print("Continuous Variables:")
[1] "Continuous Variables:"
> print(continuous_vars)
 [1] "Cases"              "FixedAcidity"       "VolatileAcidity"    "CitricAcid"
 [5] "ResidualSugar"      "Chlorides"          "FreeSulfurDioxide"  "TotalSulfurDioxide"
 [9] "Density"            "pH"                 "Sulphates"          "Alcohol"
[13] "AcidIndex"
>
> print("Discrete Variables:")
[1] "Discrete Variables:"
> print(discrete_vars)
[1] "Purchase"    "STARS"       "LabelAppeal"
```

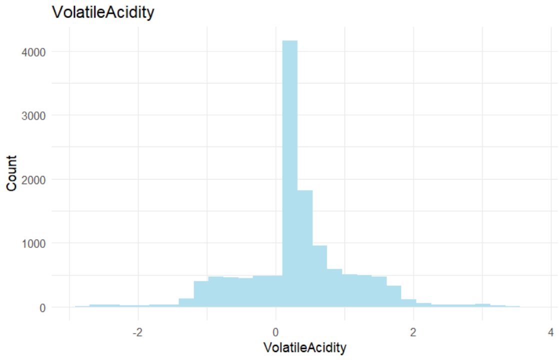
Number of records for each level of the discrete variables:

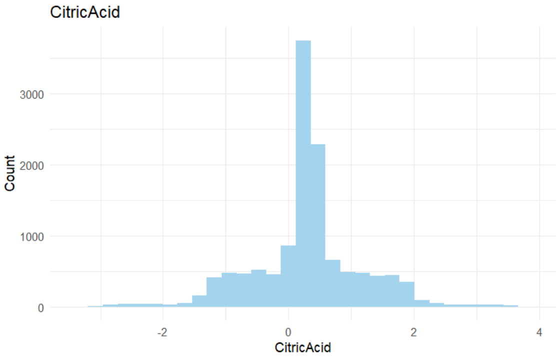


The histograms show distributions for Purchase, STARS, and LabelAppeal. Purchase is binary, with non-purchases (2,734) being less frequent than purchases (10,061). STARS ranges from 1 to 4, with 1 and 2 being the most common ratings. LabelAppeal, almost normally distributed, varies from -2 to 2, with 0 being the most frequent, suggesting neutral appeal is common.

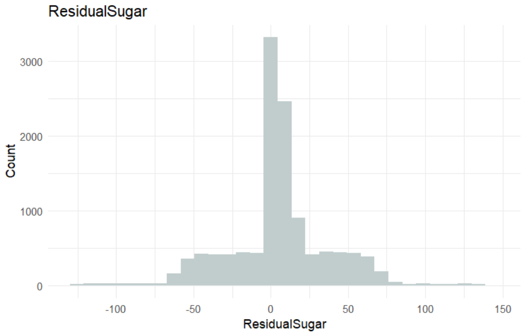


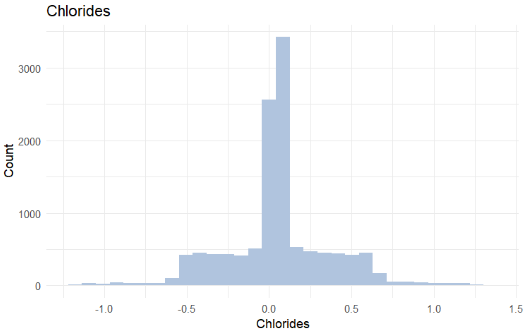


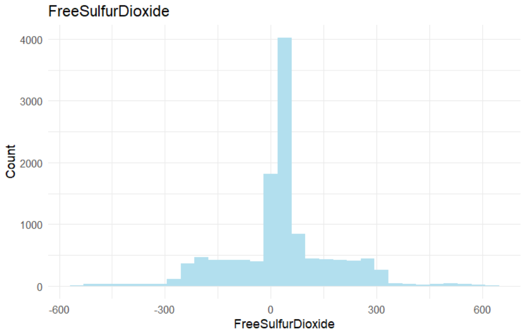


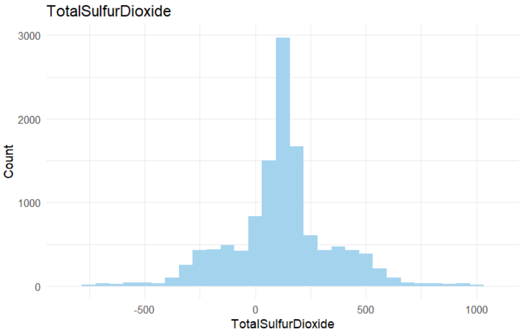


```
Table: Summary of Continuous Variables

|        |    Cases     | FixedAcidity   |VolatileAcidity | CitricAcid     |
|:-------|:-------------|:---------------|:---------------|:---------------|
|        |Min.   :0.000 |Min.   :-18.100 |Min.   :-2.7900 |Min.   :-3.2400 |
|        |1st Qu.:2.000 |1st Qu.:  5.200 |1st Qu.: 0.1300 |1st Qu.: 0.0300 |
|        |Median :3.000 |Median :  6.900 |Median : 0.2800 |Median : 0.3100 |
|        |Mean   :3.029 |Mean   :  7.076 |Mean   : 0.3241 |Mean   : 0.3084 |
|        |3rd Qu.:4.000 |3rd Qu.:  9.500 |3rd Qu.: 0.6400 |3rd Qu.: 0.5800 |
|        |Max.   :8.000 |Max.   : 34.400 |Max.   : 3.6800 |Max.   : 3.8600 |
|Std.Dev |1.9264        |6.3176          |0.784           |0.8621          |
```

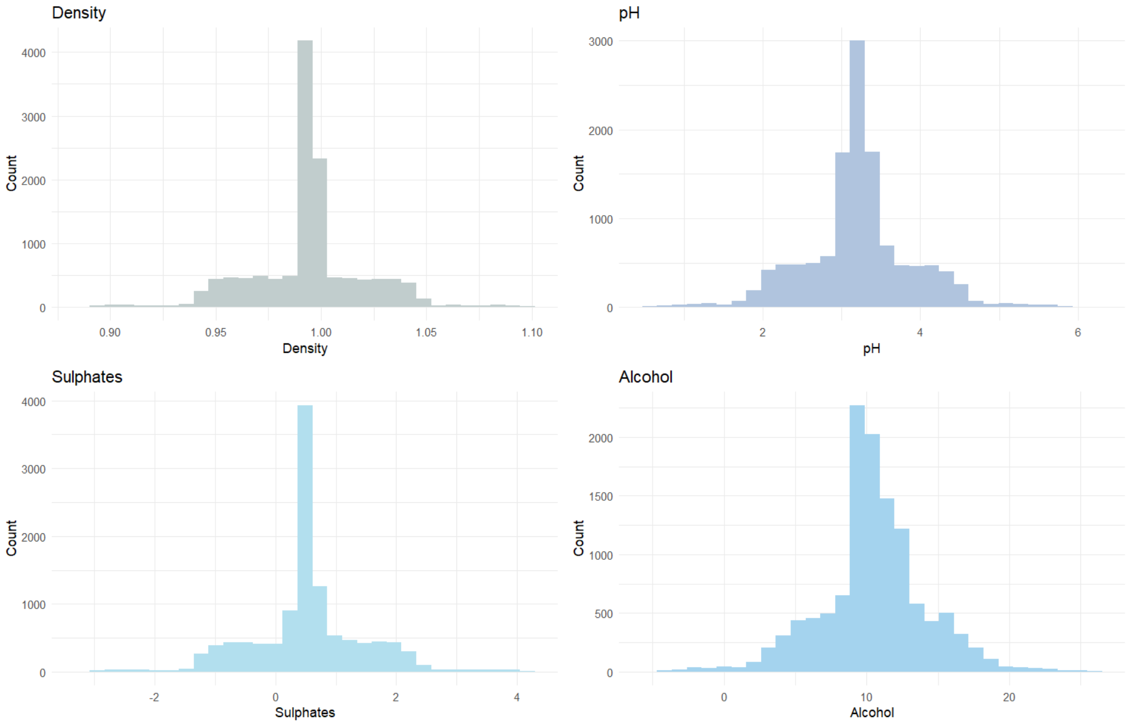




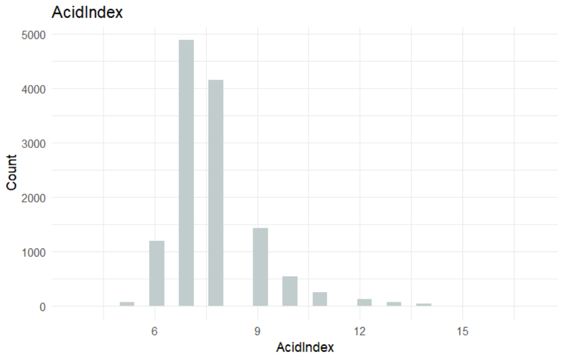




```
Table: Summary of Continuous Variables

|        |ResidualSugar    |  Chlorides     |FreeSulfurDioxide |TotalSulfurDioxide |
|:-------|:----------------|:---------------|:-----------------|:------------------|
|        |Min.   :-127.800 |Min.   :-1.1710 |Min.   :-555.00   |Min.   :-823.0     |
|        |1st Qu.:  -2.000 |1st Qu.:-0.0310 |1st Qu.:   0.00   |1st Qu.:  27.0     |
|        |Median :   3.900 |Median : 0.0460 |Median :  30.00   |Median : 123.0     |
|        |Mean   :   5.419 |Mean   : 0.0548 |Mean   :  30.85   |Mean   : 120.7     |
|        |3rd Qu.:  15.900 |3rd Qu.: 0.1530 |3rd Qu.:  70.00   |3rd Qu.: 208.0     |
|        |Max.   : 141.150 |Max.   : 1.3510 |Max.   : 623.00   |Max.   :1057.0     |
|        |NA's   :616      |NA's   :638     |NA's   :647       |NA's   :682        |
|Std.Dev |33.7494          |0.3185          |148.7146          |231.9132           |
```



```
Table: Summary of Continuous Variables

|        |   Density      |      pH      |  Sulphates      |   Alcohol    |
|:-------|:--------------|:-------------|:---------------|:-------------|
|        |Min.   :0.8881 |Min.   :0.480 |Min.   :-3.1300 |Min.   :-4.70 |
|        |1st Qu.:0.9877 |1st Qu.:2.960 |1st Qu.: 0.2800 |1st Qu.: 9.00 |
|        |Median :0.9945 |Median :3.200 |Median : 0.5000 |Median :10.40 |
|        |Mean   :0.9942 |Mean   :3.208 |Mean   : 0.5271 |Mean   :10.49 |
|        |3rd Qu.:1.0005 |3rd Qu.:3.470 |3rd Qu.: 0.8600 |3rd Qu.:12.40 |
|        |Max.   :1.0992 |Max.   :6.130 |Max.   : 4.2400 |Max.   :26.50 |
|        |NA             |NA's   :395   |NA's   :1210    |NA's   :653   |
|Std.Dev |0.0265         |0.6797        |0.9321          |3.7278        |
```

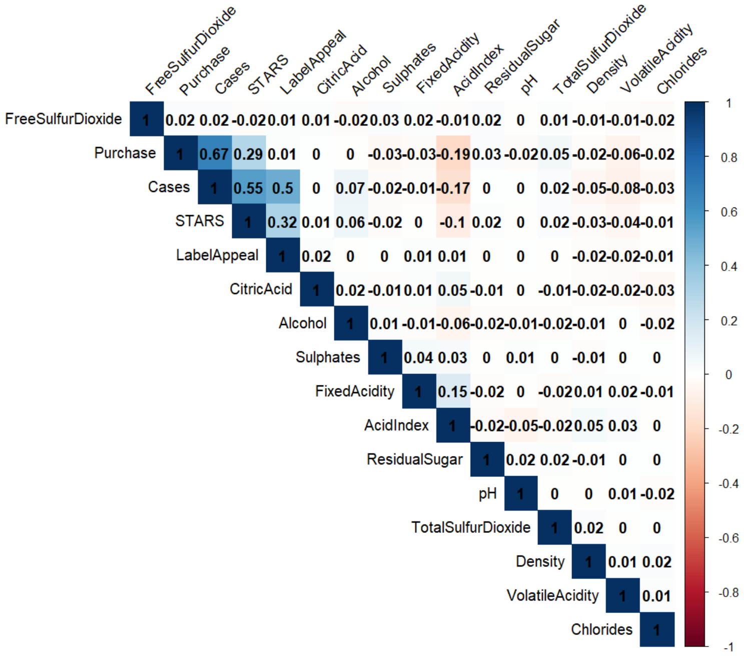


```
Table: Summary of Continuous Variables

|        |  AcidIndex     |
|:-------|:--------------|
|        |Min.   : 4.000 |
|        |1st Qu.: 7.000 |
|        |Median : 8.000 |
|        |Mean   : 7.773 |
|        |3rd Qu.: 8.000 |
|        |Max.   :17.000 |
|Std.Dev |1.3239         |
```

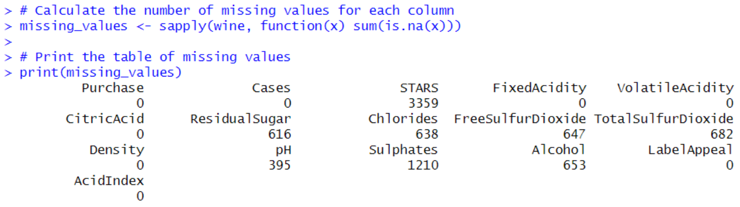
- Are variables correlated to the target variable (TARGET_WINS) or to other possible explanatory variables?

The dataset does not include a variable called 'TARGET_WINS'.

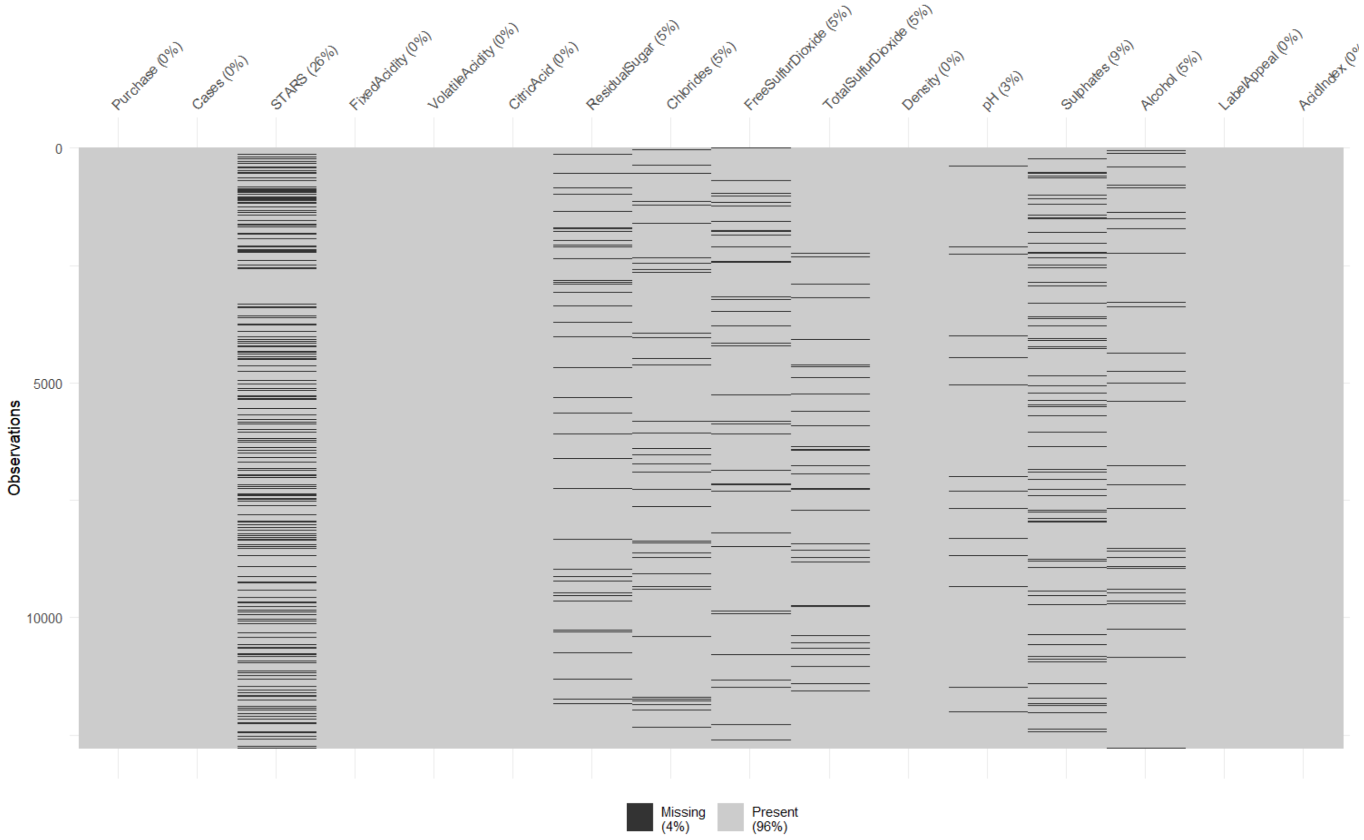
- Purchase has a strong positive correlation (0.67) with Cases and a moderate positive correlation with STARS (0.29).
- Cases and STARS are positively correlated with each other (0.55).
- Cases and LabelAppeal are positively correlated with each other (0.5).
- STARS and LabelAppeal are positively correlated with each other (0.32).
- FixedAcidity has a notable positive correlation with CitricAcid (0.15).
- Most variables have weak correlations with each other, as indicated by the color's low intensity.



- Are any of the variables with missing values that need to be imputed or “fixed”? Fix missing values (maybe with a Mean or Median value or use a decision tree). Are there variables with so many missing values that the entire variable should be eliminated from the analysis?



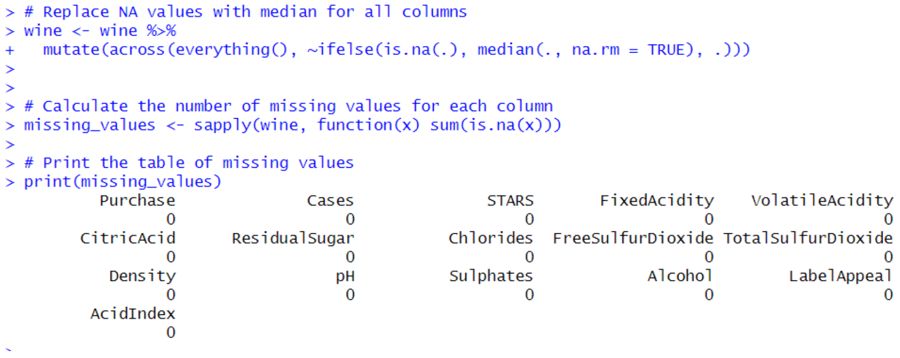
```
> # Calculate the number of missing values for each column
> missing_values <- sapply(wine, function(x) sum(is.na(x)))
> 
> # Print the table of missing values
> print(missing_values)
          Purchase              Cases              STARS       FixedAcidity    VolatileAcidity 
                 0                  0               3359                  0                  0 
        CitricAcid      ResidualSugar          Chlorides  FreeSulfurDioxide TotalSulfurDioxide 
                 0                616                638                647                682 
           Density                 pH          Sulphates            Alcohol        LabelAppeal 
                 0                395               1210                653                  0 
         AcidIndex 
                 0 
```



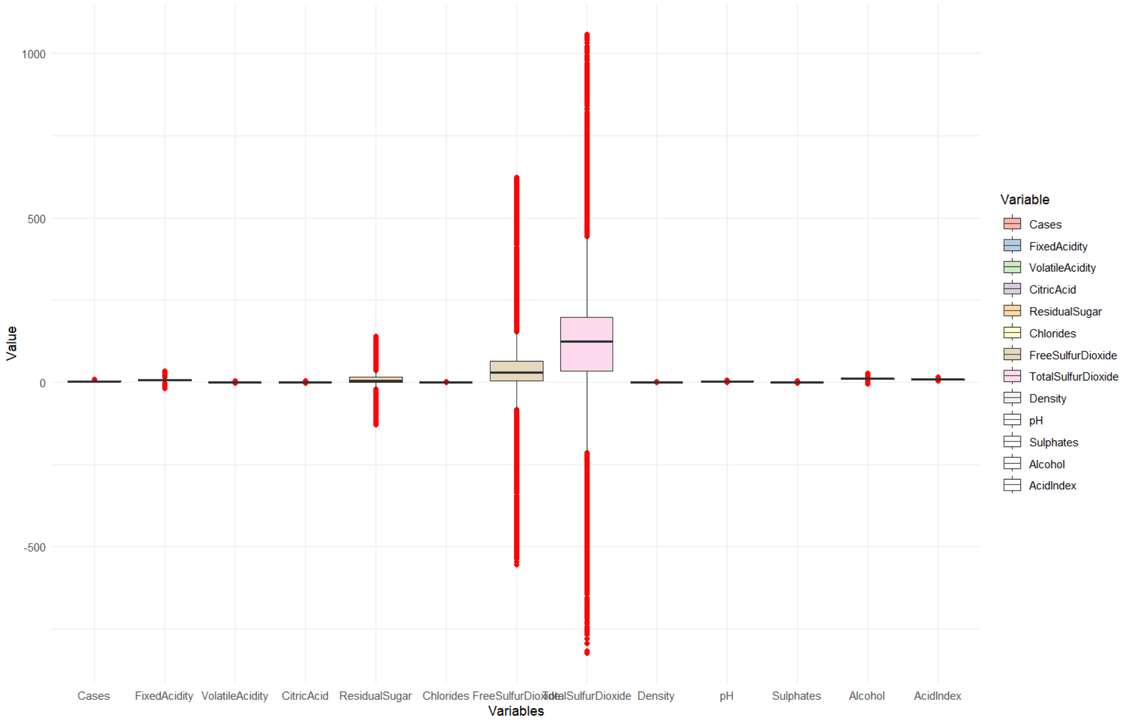
The general rule of thumb for eliminating a variable entirely from the analysis is if it is missing a significant amount of data, often more than 50-60%. Since none of the variables in this plot appear to be missing to that extent, it may not be necessary to eliminate any of the entire variables solely based on the amount of missing data.

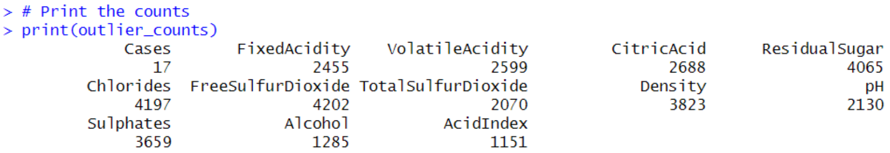
We are imputing the missing values with the medians, because:

- Outliers and Skewed Data: The median is robust to outliers and skewed distributions. If a dataset has extreme values or is not symmetrically distributed, the mean can be dragged in the direction of the skew or outliers, giving a non-representative value for central tendency. The median, being the middle value, remains unaffected by extreme scores and thus provides a better central value for imputation in such cases.
- Preservation of Distribution: Imputing with the median is less likely to affect the original distribution of the data. Mean imputation can artificially lower the variance of the data, especially when the missing data is substantial, because the mean is influenced by all data points and their values.
- Ordinal Data: For ordinal data (data that is categorical and ordered but not necessarily evenly spaced), the median can be more meaningful than the mean, which may not represent an actual category if the categories are not numerical.
- Intuitive for Categorical Data: When dealing with discrete or categorical variables, the median is more intuitive for imputation since it corresponds to an actual observed category.
- Non-Numeric Data: If dealing with ranks or other non-numeric data that can be ordered, the median is a more appropriate measure of central tendency than the mean.



```
> # Replace NA values with median for all columns
> wine <- wine %>%
+   mutate(across(everything(), ~ifelse(is.na(.), median(., na.rm = TRUE), .)))
>
>
> # Calculate the number of missing values for each column
> missing_values <- sapply(wine, function(x) sum(is.na(x)))
>
> # Print the table of missing values
> print(missing_values)
          Purchase              Cases              STARS       FixedAcidity    VolatileAcidity
                 0                  0                  0                  0                  0
        CitricAcid      ResidualSugar          Chlorides  FreeSulfurDioxide TotalSulfurDioxide
                 0                  0                  0                  0                  0
           Density                 pH          Sulphates            Alcohol        LabelAppeal
                 0                  0                  0                  0                  0
         AcidIndex
                 0
```





```
> # Print the counts
> print(outlier_counts)
           Cases        FixedAcidity     VolatileAcidity          CitricAcid       ResidualSugar 
              17                2455                2599                2688                4065 
       Chlorides  FreeSulfurDioxide TotalSulfurDioxide             Density                  pH 
            4197                4202                2070                3823                2130 
       Sulphates             Alcohol           AcidIndex 
            3659                1285                1151 
```

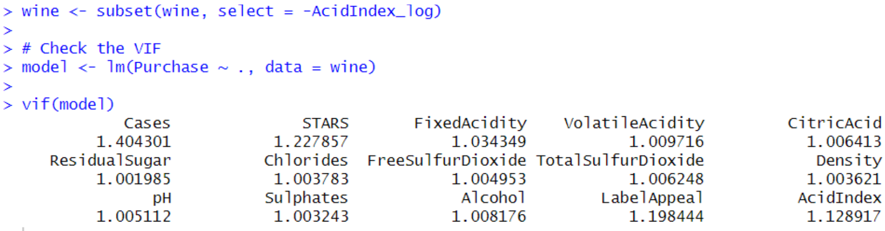
- The data shows a considerable number of outliers for many variables. This is particularly noticeable for FreeSulfurDioxide, Chlorides, Sulphates, and ResidualSugar, which show many data points beyond the upper whiskers.

In the absence of specialized domain knowledge, it's challenging to determine the significance of the extreme values in the dataset. Outliers can arise from various sources—some may be artifacts like data entry or measurement errors, which, if identified, should be rectified or excluded from the dataset. On the other hand, outliers may also represent genuine, albeit rare, phenomena that could be critical to the study. The distinction between error and valuable data point is not clear-cut without in-depth domain understanding. And, we lack the domain understanding.

Since we cannot conclusively ascertain whether the outliers in this case are the result of measurement errors, we must proceed with caution. Therefore, it is not possible to ascertain the optimal approach for addressing (fixing) these values.

- Do any of the variables need a mathematical transformation, such as log or square root? Create new variables with these transformations and add them to the end of the dataset.

Checking the VIF values:



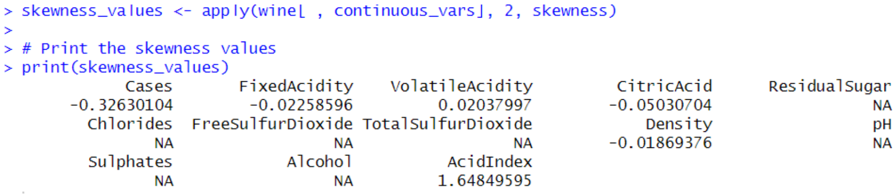
```
> wine <- subset(wine, select = -AcidIndex_log)
> 
> # Check the VIF
> model <- lm(Purchase ~ ., data = wine)
> 
> vif(model)
           Cases              STARS       FixedAcidity    VolatileAcidity         CitricAcid 
        1.404301           1.227857           1.034349           1.009716           1.006413 
   ResidualSugar          Chlorides  FreeSulfurDioxide TotalSulfurDioxide            Density 
        1.001985           1.003783           1.004953           1.006248           1.003621 
              pH          Sulphates            Alcohol        LabelAppeal          AcidIndex 
        1.005112           1.003243           1.008176           1.198444           1.128917 
```

Summary:

- Cases: VIF is 1.404301, which suggests a low to moderate level of correlation with other predictor variables.
- STARS: VIF is 1.227857, which also indicates a low to moderate correlation with other predictors.
- FixedAcidity: VIF is 1.034349, suggesting very little correlation.
- VolatileAcidity, Density, CitricAcid, ResidualSugar, Chlorides, FreeSulfurDioxide, TotalSulfurDioxide, pH, and Sulphates: All have VIFs close to 1, indicating no significant correlation with other predictors in the model.
- Alcohol: VIF is slightly higher at 1.988176, which could suggest a moderate correlation but still under the common threshold of concern.
- LabelAppeal: VIF is 1.198444, indicating a low to moderate correlation.
- AcidIndex: VIF is 1.128917, also indicating a low to moderate correlation.

In conclusion, there are no clear signs of multicollinearity based on these VIF values. All the variables have VIF values well below the threshold of concern (commonly considered to be 5 or 10), meaning that they could all reasonably be kept in the model without concern for inflating the variance of the estimated coefficients due to multicollinearity.

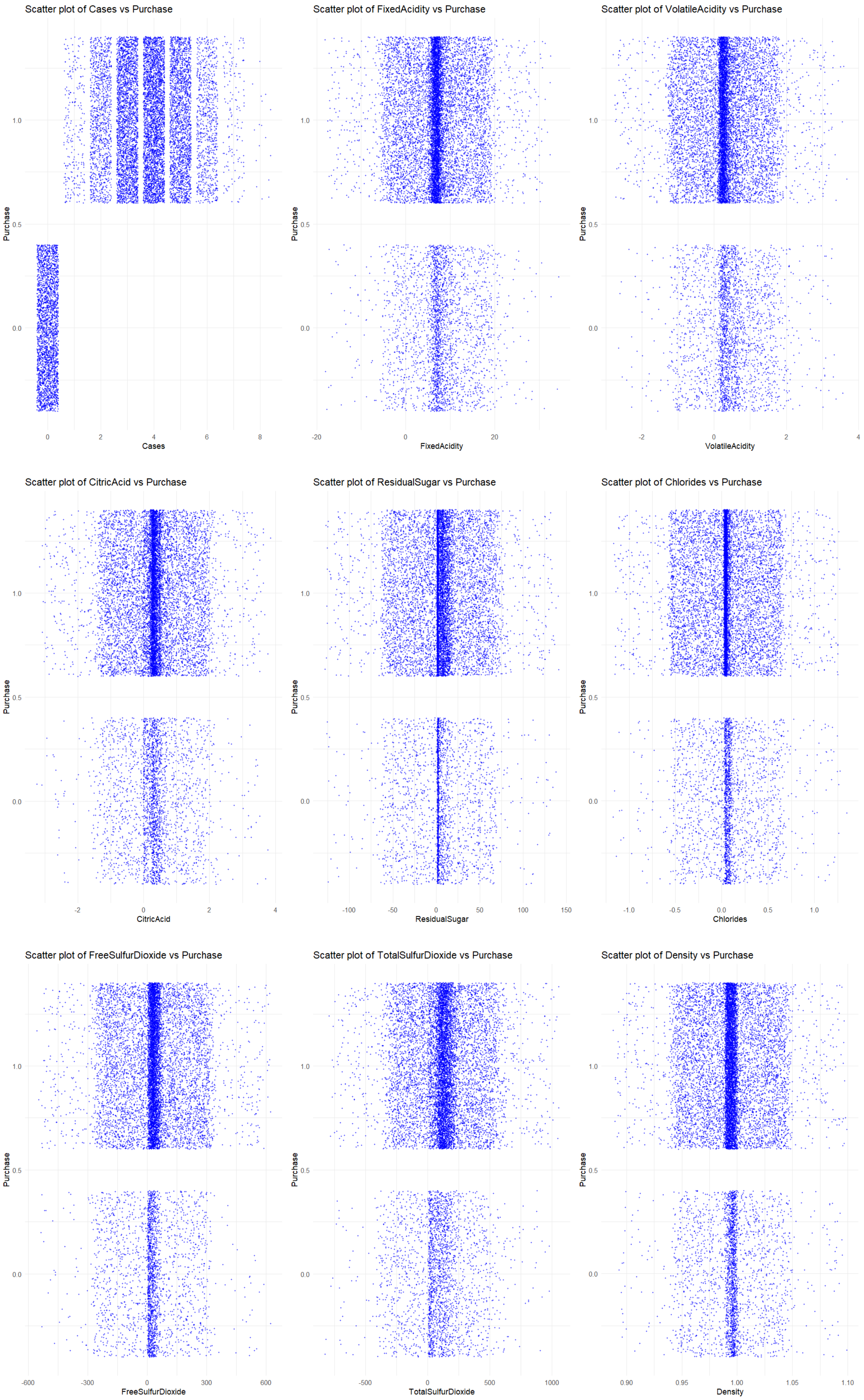
Checking the Skewness:

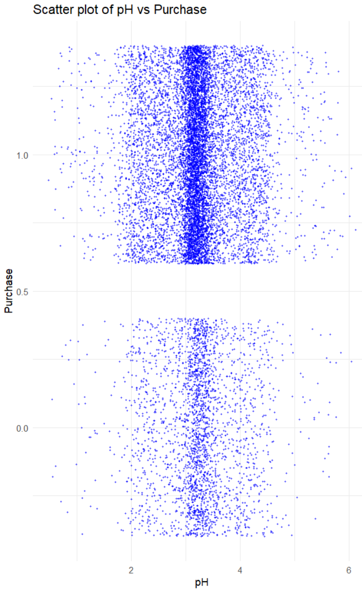


```
> skewness_values <- apply(wine[ , continuous_vars], 2, skewness)
> 
> # Print the skewness values
> print(skewness_values)
           Cases        FixedAcidity     VolatileAcidity          CitricAcid       ResidualSugar 
     -0.32630104         -0.02258596          0.02037997         -0.05030704                  NA 
       Chlorides  FreeSulfurDioxide TotalSulfurDioxide             Density                  pH 
              NA                  NA                  NA         -0.01869376                  NA 
       Sulphates             Alcohol           AcidIndex 
              NA                  NA          1.64849595 
```

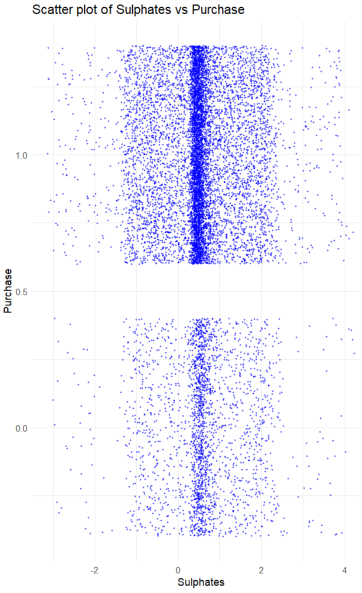
The skewness value for "AcidIndex" is positive and relatively high (greater than 1.0), indicating right-skewness.

Plotting Scatterplots with Jitters, for all continuous variables, with target variable Purchase.

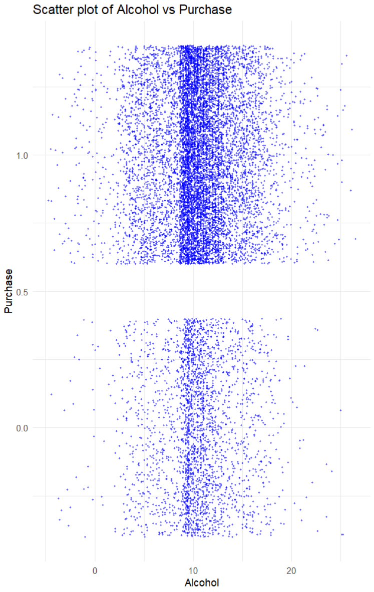




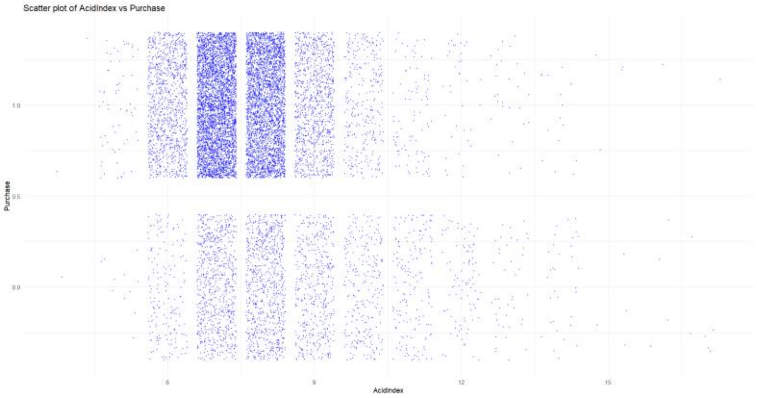
Scatter plot of pH vs Purchase



Scatter plot of Sulphates vs Purchase



Scatter plot of Alcohol vs Purchase



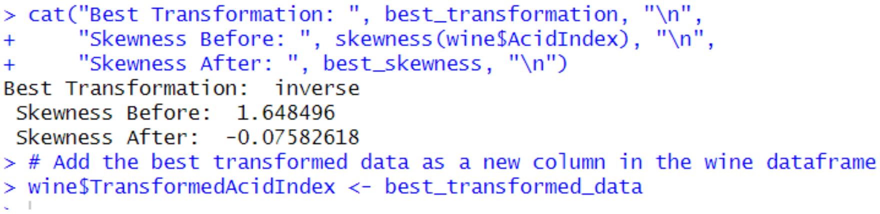
Scatter plot of AcidIndex vs Purchase

Looking at these plots, it is evident that Purchase = 1 only when Cases > 0, and Purchase = 0 when Cases = 0. For all other variables, Purchase values are spread across different values of the variables.

Given that Purchase is 0 when Cases is 0 and Purchase is 1 when Cases is 1, 2, 3, 4, 5, 6, 7, or 8, this relationship between the predictor (Cases) and the response (Purchase) is deterministic and simple. In this case, no statistical model, including logistic regression or any transformation of the predictor variable, is necessary or appropriate. Therefore, we will not consider Cases to build our model.

- Create any new variables that you are interested in.

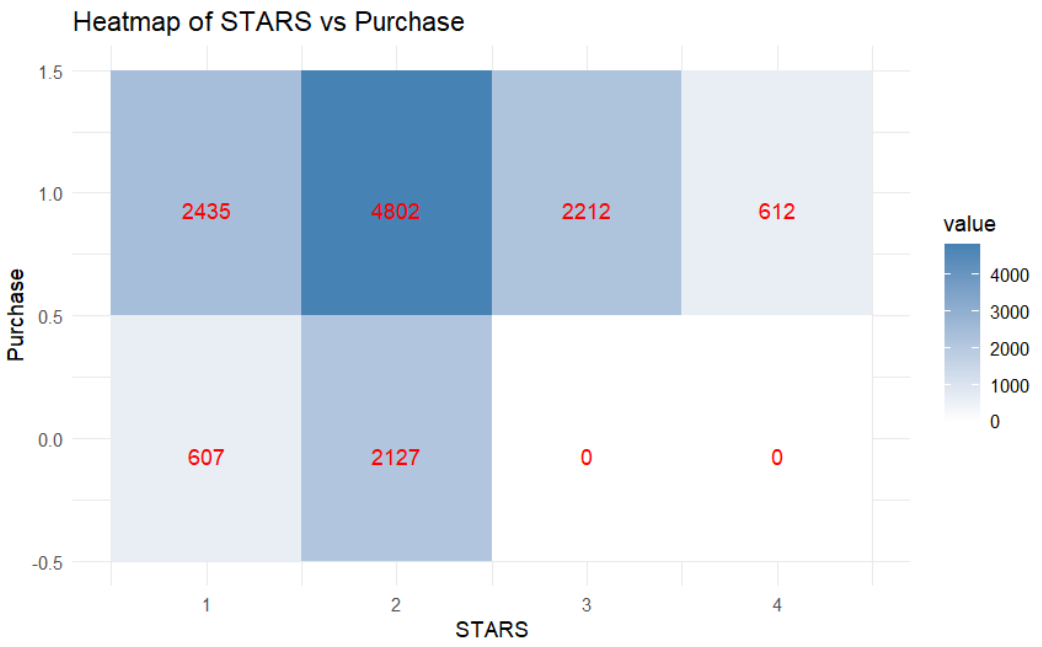
The Scatter Plot of AcidIndex also shows Right Skewness (1.64). To address this, we try out different transformations on AcidIndex, and include the best transformed AcidIndex in the dataframe.

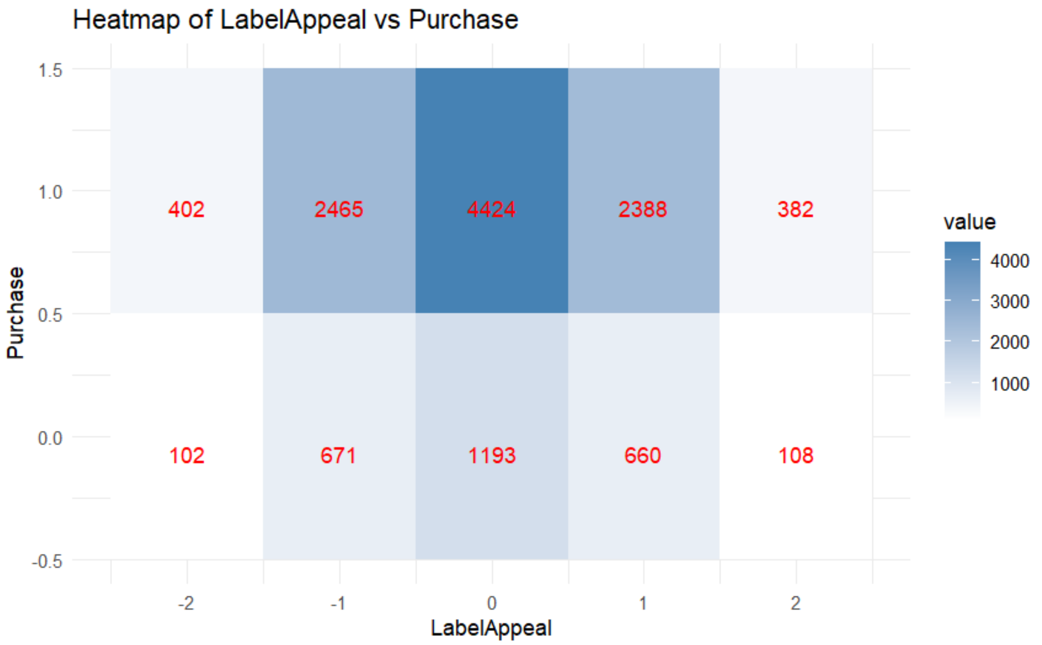


```
> cat("Best Transformation: ", best_transformation, "\n",
+     "Skewness Before: ", skewness(wine$AcidIndex), "\n",
+     "Skewness After: ", best_skewness, "\n")
Best Transformation:  inverse
 Skewness Before:  1.648496
 Skewness After:  -0.07582618
> # Add the best transformed data as a new column in the wine dataframe
> wine$TransformedAcidIndex <- best_transformed_data
```

Evaluating the Discrete Variables:

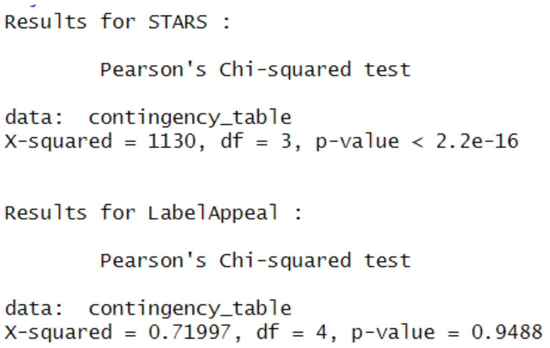
Printing the Contingency Heat Plots:





In the STARS heatmap, the frequency of purchases increases with the number of stars, peaking at 2 STARS, and then decreases. No Purchase = 0 are associated with 3 and 4 STARS, which could mean that well rated wines are always purchased, but their orders are lesser.

In the LabelAppeal heatmap, the middle categories (0 and 1) have the highest frequencies of purchases, with a noticeable drop for extreme negative and positive values. This could suggest that extremely polarized opinions on label appeal (both negative and positive) are less common among purchases or that most labels have a neutral to mildly positive appeal.



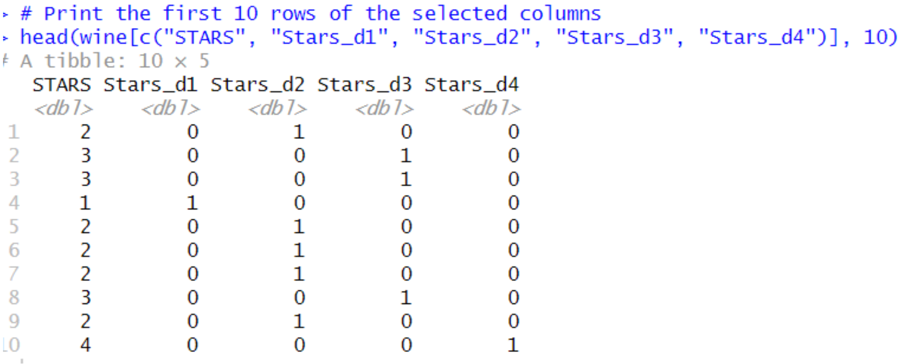
```
Results for STARS :

        Pearson's Chi-squared test

data:  contingency_table
X-squared = 1130, df = 3, p-value < 2.2e-16


Results for LabelAppeal :

        Pearson's Chi-squared test

data:  contingency_table
X-squared = 0.71997, df = 4, p-value = 0.9488
```

The Chi-squared test results for the STARS variable show a Chi-square statistic (X-squared) of 1130 with 3 degrees of freedom and a p-value less than 2.2e-16. This extremely small p-value indicates that there is a highly significant association between the STARS variable and the Purchase variable. In practical terms, the number of stars has a statistically significant impact on the purchase frequency.

On the other hand, the Chi-squared test results for the LabelAppeal variable show a Chi-square statistic of approximately 0.72 with 4 degrees of freedom and a p-value of 0.9488. This high p-value suggests that there is no significant association between LabelAppeal and Purchase; in other words, the variations in label appeal do not significantly affect the purchase decisions, at least not in a way that is detectable by this test.

In summary, STARS seems to have a significant relationship with purchases, while LabelAppeal does not, based on the data provided. Therefore, we will not be considering the LabelAppeal in our model.

We dummy code STARS and create 4 dummy variables.



```
> # Print the first 10 rows of the selected columns
> head(wine[c("STARS", "Stars_d1", "Stars_d2", "Stars_d3", "Stars_d4")], 10)
# A tibble: 10 × 5
   STARS Stars_d1 Stars_d2 Stars_d3 Stars_d4
   <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
 1     2        0        1        0        0
 2     3        0        0        1        0
 3     3        0        0        1        0
 4     1        1        0        0        0
 5     2        0        1        0        0
 6     2        0        1        0        0
 7     2        0        1        0        0
 8     3        0        0        1        0
 9     2        0        1        0        0
10     4        0        0        0        1
```

The variables that we will be using in creating our Logistic Model for Target Model Purchase are:

FixedAcidity, VolatileAcidity, CitricAcid, ResidualSugar, Chlorides, FreeSulfurDioxide, TotalSulfurDioxide, Density, pH, Sulphates, Alcohol, TransformedAcidIndex, Stars_d1, Stars_d2, Stars_d3, Stars_d4

We will create two models, one with AcidIndex and one with TransformedAcidIndex, and try to understand if the transformation of AcidIndex has significantly improved the model performance.

Task 2

There is not one perfectly correct way to approach model building. You are now charged with the task of producing your best predictive model for the PURCHASE (Y) decision. This is an open-ended modeling task. You may select the variables manually or use an automated approach such as Forward or Stepwise. You may use continuous or categorical variables as part of the explanatory variable set. You have enough data, so you should very seriously consider taking a validation approach to this modeling endeavor, though it is not required. You need to be sure you can interpret your models, have evidence on goodness of fit, and check on assumptions via diagnostics. What criteria are you going to use select your "best" model?

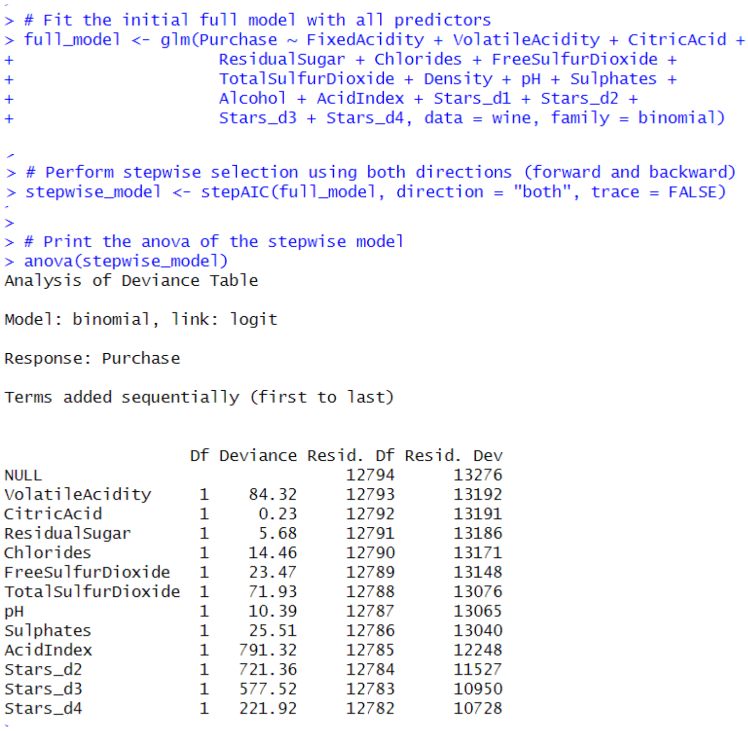
Write of description of the technique you used to decide on your final model. Write up your final model. Report the model. Discuss the coefficients in the model, do they make sense? Report on goodness of fit and model diagnostics.

We will be using Stepwise Selection to identify the variables that have the most impact on our Logistic Regression Model, because it:

Combines Forward and Backward: Stepwise selection is a combination of both forward and backward selection. It starts like forward selection by adding predictors one at a time, but after adding each new variable, it checks if any of the previously included variables have become insignificant and should be removed. This allows for a more nuanced model building process.

Provides flexibility: It provides a balance between the purely sequential addition of variables (forward) and the elimination of variables (backward). This can lead to a more optimized model, as it considers both inclusion and exclusion of variables throughout the process.
Prevents Overfitting: By evaluating at each step whether any variables should be removed, stepwise selection can help prevent overfitting, which is a common issue in model building.

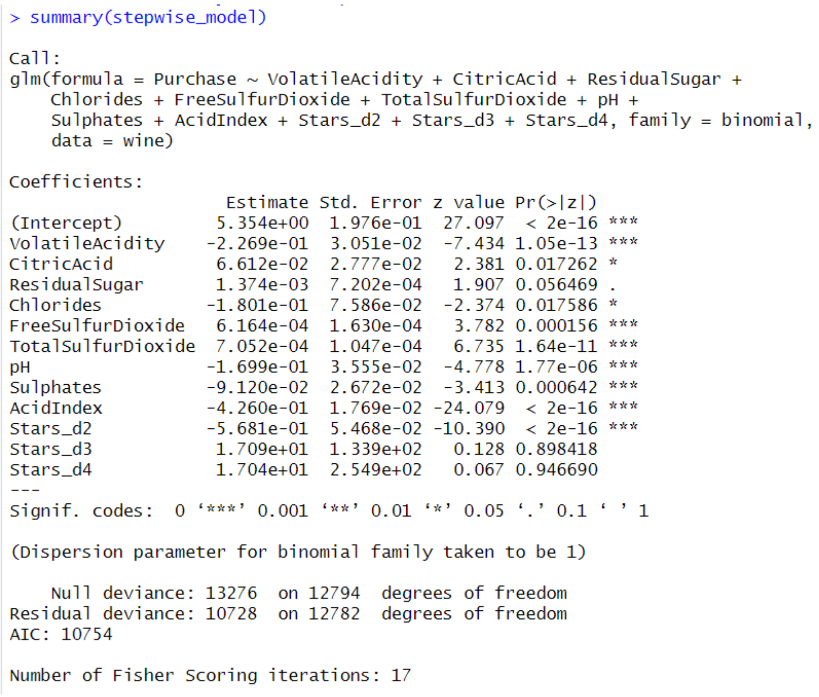
This is the first model that has been created without transforming AcidIndex:



```
> # Fit the initial full model with all predictors
> full_model <- glm(Purchase ~ FixedAcidity + VolatileAcidity + CitricAcid +
+                     ResidualSugar + Chlorides + FreeSulfurDioxide +
+                     TotalSulfurDioxide + Density + pH + Sulphates +
+                     Alcohol + AcidIndex + Stars_d1 + Stars_d2 +
+                     Stars_d3 + Stars_d4, data = wine, family = binomial)

> # Perform stepwise selection using both directions (forward and backward)
> stepwise_model <- stepAIC(full_model, direction = "both", trace = FALSE)
>
> # Print the anova of the stepwise model
> anova(stepwise_model)
Analysis of Deviance Table

Model: binomial, link: logit

Response: Purchase

Terms added sequentially (first to last)


                   Df Deviance Resid. Df Resid. Dev
NULL                              12794      13276
VolatileAcidity     1    84.32    12793      13192
CitricAcid          1     0.23    12792      13191
ResidualSugar       1     5.68    12791      13186
Chlorides           1    14.46    12790      13171
FreeSulfurDioxide   1    23.47    12789      13148
TotalSulfurDioxide  1    71.93    12788      13076
pH                  1    10.39    12787      13065
Sulphates           1    25.51    12786      13040
AcidIndex           1   791.32    12785      12248
Stars_d2            1   721.36    12784      11527
Stars_d3            1   577.52    12783      10950
Stars_d4            1   221.92    12782      10728
```

Summary:

- For each variable added in the stepwise model, the table shows the reduction in deviance, indicating the contribution of each variable to the model's explanatory power.
- The "NULL" model's deviance represents the model with no predictors, just an intercept. As variables are added, the residual deviance decreases, showing improvement in the model.
- Variables like AcidIndex, Stars_d2, Stars_d3, and Stars_d4 are key contributors to the model, based on the substantial decrease in deviance they caused when added.



```
> summary(stepwise_model)

Call:
glm(formula = Purchase ~ VolatileAcidity + CitricAcid + ResidualSugar +
    Chlorides + FreeSulfurDioxide + TotalSulfurDioxide + pH +
    Sulphates + AcidIndex + Stars_d2 + Stars_d3 + Stars_d4, family = binomial,
    data = wine)

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)         5.354e+00  1.976e-01  27.097  < 2e-16 ***
VolatileAcidity    -2.269e-01  3.051e-02  -7.434 1.05e-13 ***
CitricAcid          6.612e-02  2.777e-02   2.381 0.017262 *
ResidualSugar       1.374e-03  7.202e-04   1.907 0.056469 .
Chlorides          -1.801e-01  7.586e-02  -2.374 0.017586 *
FreeSulfurDioxide   6.164e-04  1.630e-04   3.782 0.000156 ***
TotalSulfurDioxide  7.052e-04  1.047e-04   6.735 1.64e-11 ***
pH                 -1.699e-01  3.555e-02  -4.778 1.77e-06 ***
Sulphates          -9.120e-02  2.672e-02  -3.413 0.000642 ***
AcidIndex          -4.260e-01  1.769e-02 -24.079  < 2e-16 ***
Stars_d2           -5.681e-01  5.468e-02 -10.390  < 2e-16 ***
Stars_d3            1.709e+01  1.339e+02   0.128 0.898418
Stars_d4            1.704e+01  2.549e+02   0.067 0.946690
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13276  on 12794  degrees of freedom
Residual deviance: 10728  on 12782  degrees of freedom
AIC: 10754

Number of Fisher Scoring iterations: 17
```

Summary:

Model Formula: The model predicts Purchase using a set of 12 predictors: VolatileAcidity, CitricAcid, ResidualSugar, Chlorides, FreeSulfurDioxide, TotalSulfurDioxide, pH, Sulphates, AcidIndex, Stars_d2, Stars_d3, and Stars_d4.

Coefficients:

CitricAcid, TotalSulfurDioxide and FreeSulfurDioxide: All three have positive coefficients and are significant, suggesting their increase is associated with an increased likelihood of purchase.

Sulphates, AcidIndex: Significant negative coefficients, indicating their increase is associated with a decreased likelihood of purchase.

Stars_d2: Negative and highly significant, implying that this category has a lower likelihood of purchase compared to the baseline category.

Stars_d3 and Stars_d4: Not significant (high p-values), suggesting these variables do not have a statistically significant effect on the likelihood of purchase in the presence of other variables.

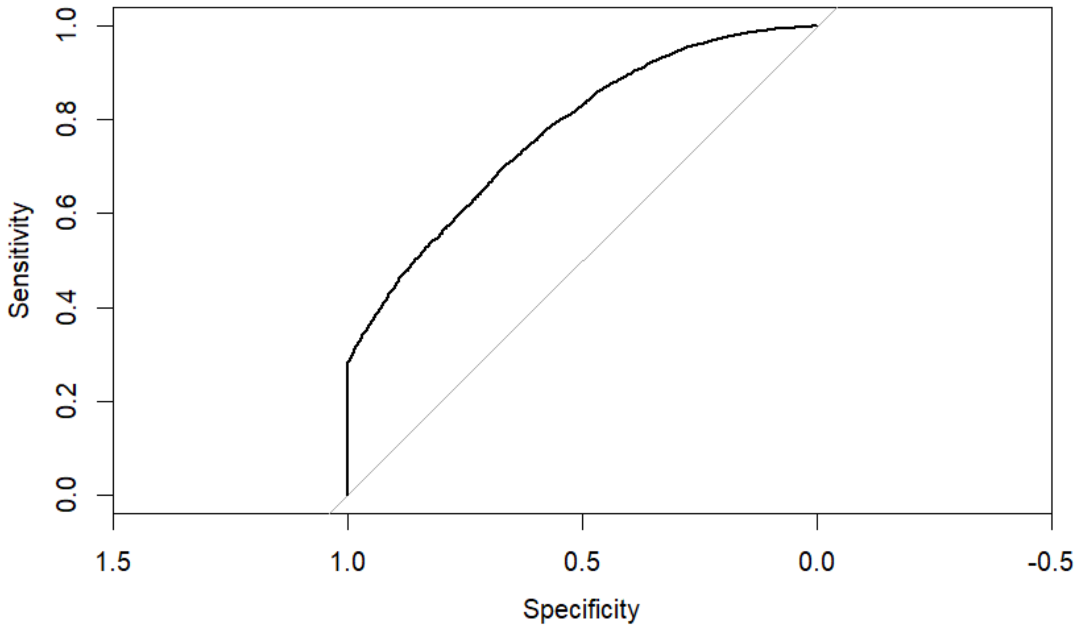
Model Fit and Diagnostics

Deviance: The null deviance and residual deviance indicate how well the model fits the data compared to a null model. A substantial drop from null to residual deviance suggests the model fits the data well.

AIC (Akaike Information Criterion: 10754): A measure of the relative quality of the model for the given set of data. Lower AIC values indicate a better fit.

Conclusion

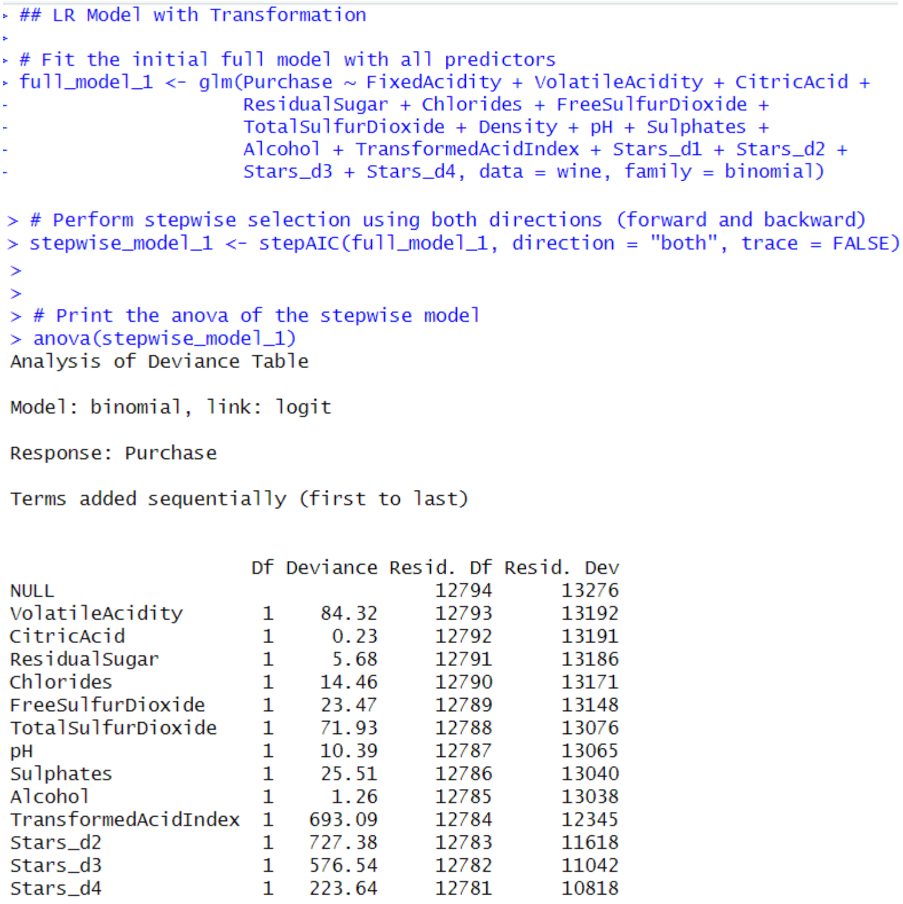
The stepwise_model seems to fit the data well, as indicated by the significant reduction in deviance from the null model. Most predictors are statistically significant, with their respective signs indicating whether they increase or decrease the likelihood of wine purchase.



Area under the curve: 0.7718

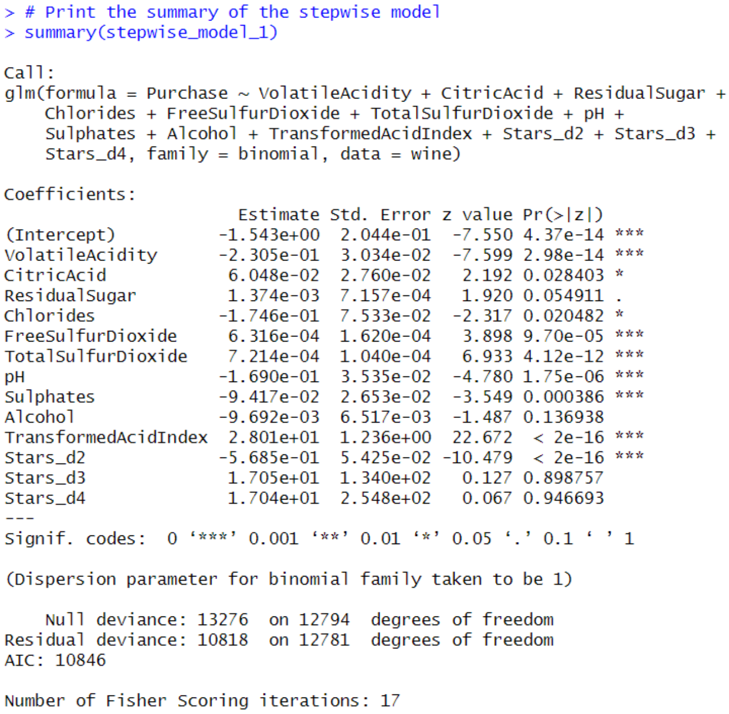
This level of AUC indicates that when presented with a pair consisting of one instance where a purchase was made and one where it was not, the model can correctly identify the purchase instance over 77% of the time.

This is the second model; it has been created after transforming AcidIndex:



```
> ## LR Model with Transformation
>
> # Fit the initial full model with all predictors
> full_model_1 <- glm(Purchase ~ FixedAcidity + VolatileAcidity + CitricAcid +
+                      ResidualSugar + Chlorides + FreeSulfurDioxide +
+                      TotalSulfurDioxide + Density + pH + Sulphates +
+                      Alcohol + TransformedAcidIndex + Stars_d1 + Stars_d2 +
+                      Stars_d3 + Stars_d4, data = wine, family = binomial)

> # Perform stepwise selection using both directions (forward and backward)
> stepwise_model_1 <- stepAIC(full_model_1, direction = "both", trace = FALSE)
>
>
> # Print the anova of the stepwise model
> anova(stepwise_model_1)
Analysis of Deviance Table

Model: binomial, link: logit

Response: Purchase

Terms added sequentially (first to last)


                     Df Deviance Resid. Df Resid. Dev
NULL                                 12794      13276
VolatileAcidity       1    84.32     12793      13192
CitricAcid            1     0.23     12792      13191
ResidualSugar         1     5.68     12791      13186
Chlorides             1    14.46     12790      13171
FreeSulfurDioxide     1    23.47     12789      13148
TotalSulfurDioxide    1    71.93     12788      13076
pH                    1    10.39     12787      13065
Sulphates             1    25.51     12786      13040
Alcohol               1     1.26     12785      13038
TransformedAcidIndex  1   693.09     12784      12345
Stars_d2              1   727.38     12783      11618
Stars_d3              1   576.54     12782      11042
Stars_d4              1   223.64     12781      10818
```

Key Contributors: Variables such as VolatileAcidity, TotalSulfurDioxide, TransformedAcidIndex, Stars_d2, Stars_d3, and Stars_d4 show significant reductions in deviance, indicating strong associations with the purchase decision.
TransformedAcidIndex: Notably, TransformedAcidIndex leads to a large decrease in deviance, suggesting it has a substantial effect on the model.
Residual Deviance: The "Resid. Dev" shows the unexplained variation by the model; a lower value indicates a better fit. The stepwise model has a residual deviance of 10818 on 12781 degrees of freedom, which is an improvement from the null model's deviance of 13276 on 12794 degrees of freedom.



```
> # Print the summary of the stepwise model
> summary(stepwise_model_1)

Call:
glm(formula = Purchase ~ VolatileAcidity + CitricAcid + ResidualSugar +
    Chlorides + FreeSulfurDioxide + TotalSulfurDioxide + pH +
    Sulphates + Alcohol + TransformedAcidIndex + Stars_d2 + Stars_d3 +
    Stars_d4, family = binomial, data = wine)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -1.543e+00  2.044e-01  -7.550 4.37e-14 ***
VolatileAcidity      -2.305e-01  3.034e-02  -7.599 2.98e-14 ***
CitricAcid            6.048e-02  2.760e-02   2.192 0.028403 *
ResidualSugar         1.374e-03  7.157e-04   1.920 0.054911 .
Chlorides            -1.746e-01  7.533e-02  -2.317 0.020482 *
FreeSulfurDioxide     6.316e-04  1.620e-04   3.898 9.70e-05 ***
TotalSulfurDioxide    7.214e-04  1.040e-04   6.933 4.12e-12 ***
pH                   -1.690e-01  3.535e-02  -4.780 1.75e-06 ***
Sulphates            -9.417e-02  2.653e-02  -3.549 0.000386 ***
Alcohol              -9.692e-03  6.517e-03  -1.487 0.136938
TransformedAcidIndex  2.801e+01  1.236e+00  22.672  < 2e-16 ***
Stars_d2             -5.685e-01  5.425e-02 -10.479  < 2e-16 ***
Stars_d3              1.705e+01  1.340e+02   0.127 0.898757
Stars_d4              1.704e+01  2.548e+02   0.067 0.946693
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13276  on 12794  degrees of freedom
Residual deviance: 10818  on 12781  degrees of freedom
AIC: 10846

Number of Fisher Scoring iterations: 17
```

Summary:

Model Formula: The model predicts Purchase using a set of 13 predictors: VolatileAcidity, CitricAcid, ResidualSugar, Chlorides, FreeSulfurDioxide, TotalSulfurDioxide, pH, Sulphates, Alcohol, TransformedAcidIndex, Stars_d2, Stars_d3, and Stars_d4.

Coefficients:
CitricAcid, TotalSulfurDioxide and FreeSulfurDioxide: All three have positive coefficients and are significant, suggesting their increase is associated with an increased likelihood of purchase.

Sulphates, Alcohol: Significant negative coefficients, indicating their increase is associated with a decreased likelihood of purchase.

Stars_d2: Negative and highly significant, implying that this category has a lower likelihood of purchase compared to the baseline category.

TransformedAcidIndex: Has a significant positive coefficient, indicating its increase is associated with a decreased likelihood of purchase.

Stars_d3 and Stars_d4: Not significant (high p-values), suggesting these variables do not have a statistically significant effect on the likelihood of purchase in the presence of other variables.

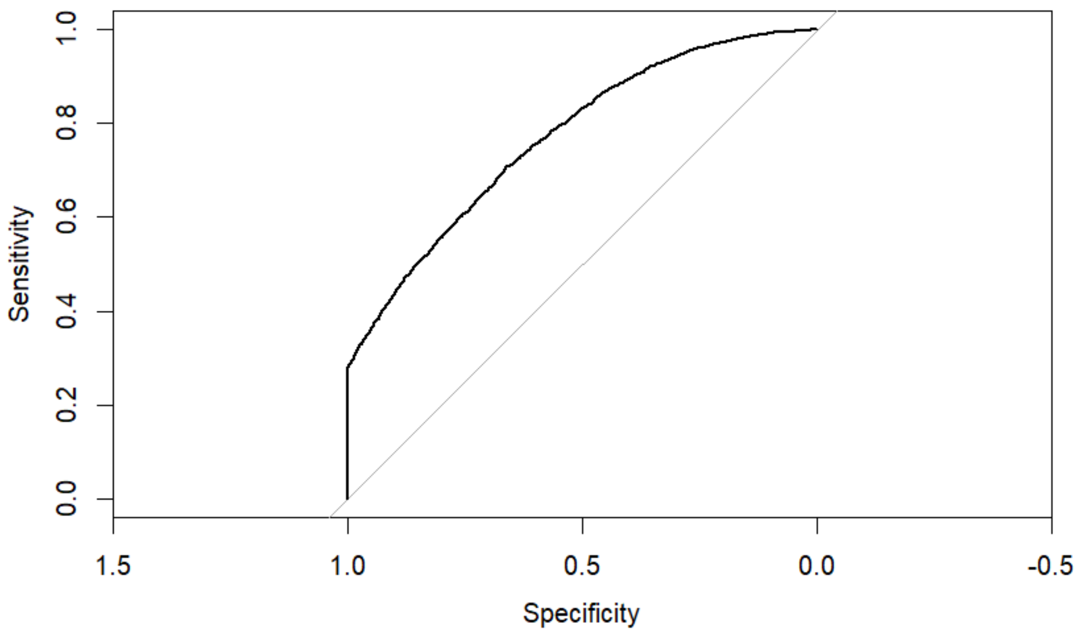
Model Fit and Diagnostics

Deviance: The null deviance and residual deviance indicate how well the model fits the data compared to a null model. A substantial drop from null to residual deviance suggests the model fits the data well.

AIC (Akaike Information Criterion: 10846): A measure of the relative quality of the model for the given set of data. Lower AIC values indicate a better fit.

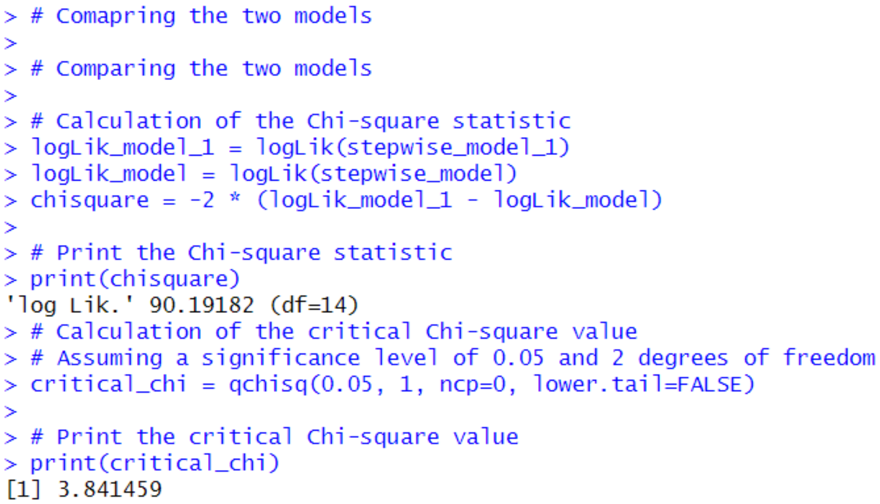
Conclusion

The stepwise_model_1 seems to fit the data well, as indicated by the significant reduction in deviance from the null model. Most predictors are statistically significant, with their respective signs indicating whether they increase or decrease the likelihood of wine purchase.



Area under the curve: 0.7696

This level of AUC indicates that when presented with a pair consisting of one instance where a purchase was made and one where it was not, the model can correctly identify the purchase instance over 77% of the time.



```
> # Comapring the two models
>
> # Comparing the two models
>
> # Calculation of the Chi-square statistic
> logLik_model_1 = logLik(stepwise_model_1)
> logLik_model = logLik(stepwise_model)
> chisquare = -2 * (logLik_model_1 - logLik_model)
>
> # Print the Chi-square statistic
> print(chisquare)
'log Lik.' 90.19182 (df=14)
> # Calculation of the critical Chi-square value
> # Assuming a significance level of 0.05 and 2 degrees of freedom
> critical_chi = qchisq(0.05, 1, ncp=0, lower.tail=FALSE)
>
> # Print the critical Chi-square value
> print(critical_chi)
[1] 3.841459
```

This test is the likelihood ratio test, which is used to compare the goodness of fit between two models. Here's the interpretation of the results:

- logLik_model_1 and logLik_model are capturing the log-likelihoods of stepwise_model_1 and stepwise_model, respectively.
- chisquare: The Chi-square statistic calculated as -2 * (logLik_model_1 - logLik_model) is 90.19182. This value is a measure of the difference in fit between the two models, with a higher value indicating a greater difference.
- critical_chi: The critical value from the Chi-square distribution for 1 degree of freedom (number of variables in stepwise_model_1 - number of variables in stepwise_model = 1) at the 5% significance level is 3.841459. This value is the cutoff above which we would reject the null hypothesis that the two models fit the data equally well.
- The output [1] 3.841459 is the critical Chi-square value for this test.
- Since 90.19182 is substantially greater than 3.841459, it suggests that the difference between the two models is statistically significant at the 0.05 level.

Comparison of Models:

AUC Values: The AUC value is a measure of the model's ability to distinguish between different outcomes (e.g., positive and negative classes in classification problems). A higher AUC indicates a better model in terms of predictive power.

- stepwise_model has an AUC of 0.7718.
- stepwise_model_1 has an AUC of 0.7696.

Number of Variables: A model with fewer variables is generally preferred if it maintains similar predictive performance because it is simpler and potentially less prone to overfitting. This concept is known as the principle of parsimony or Occam's Razor.

- stepwise_model uses 12 variables.
- stepwise_model_1 uses 13 variables.

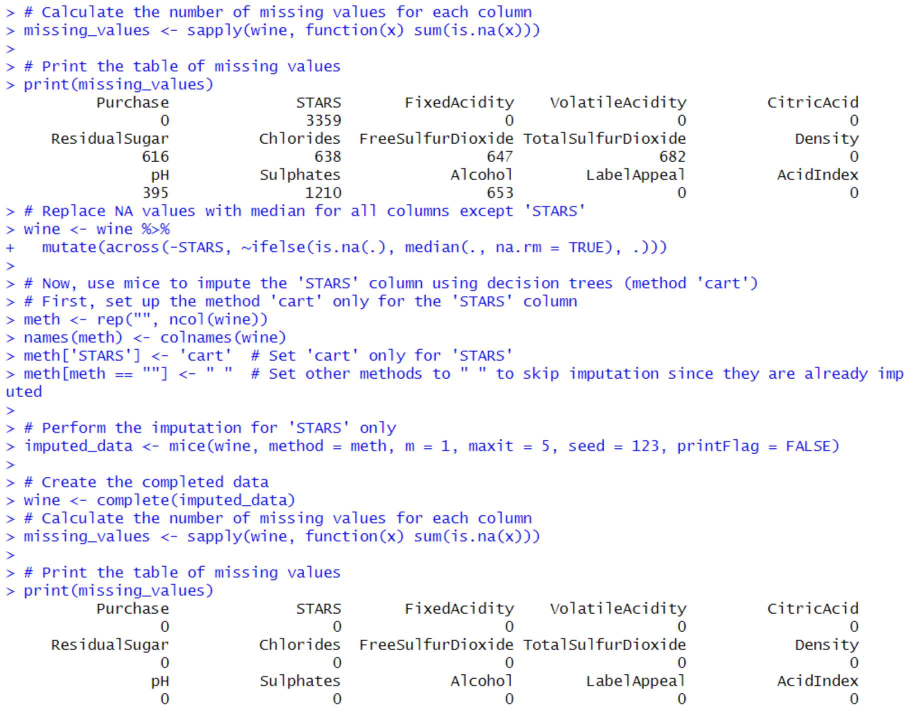
Comparison and Decision:

- While stepwise_model has a slightly higher AUC, indicating marginally better predictive performance, it also uses one lesser variable than stepwise_model_1.
- The difference in AUC is very small (0.7718 vs. 0.7696). Such a minor difference might not be practically significant, especially considering the trade-off with model complexity.

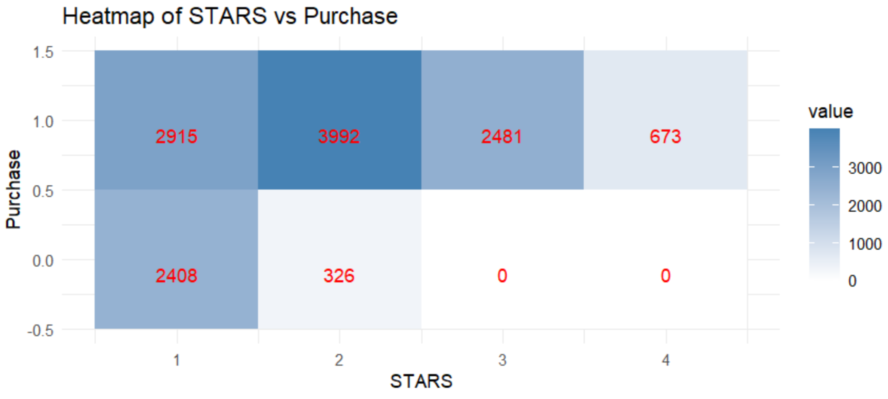
Decision Factors:

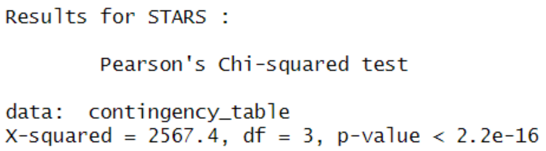
- If predictive accuracy is the sole criterion, stepwise_model is marginally better.
- If one values simplicity and a reduced risk of overfitting, stepwise_model is still the preferred choice.

Considering that "STARS" is a discrete variable with approximately 25% of its values missing, we are adapting our modeling strategy. For our third model iteration, we will employ Decision Tree imputation specifically for the "STARS" variable. This method is particularly suited for handling the discrete nature of "STARS" and can adeptly manage the significant proportion of missing data. Decision Trees are adept at capturing complex patterns and are robust to outliers, which is beneficial for a variable that is not continuously scaled. For the other variables, which display far lesser missing data, we will continue to implement median imputation, thereby maintaining the distributional integrity of these more complete variables. This tailored approach to imputation aims to refine the accuracy of our model and capitalize on the strengths of both imputation methods given the nature of the data.



```
> # Calculate the number of missing values for each column
> missing_values <- sapply(wine, function(x) sum(is.na(x)))
>
> # Print the table of missing values
> print(missing_values)
        Purchase            STARS       FixedAcidity    VolatileAcidity          CitricAcid
               0             3359                  0                  0                   0
   ResidualSugar        Chlorides  FreeSulfurDioxide TotalSulfurDioxide             Density
             616              638                647                682                   0
              pH        Sulphates            Alcohol        LabelAppeal           AcidIndex
             395             1210                653                  0                   0
> # Replace NA values with median for all columns except 'STARS'
> wine <- wine %>%
+   mutate(across(-STARS, ~ifelse(is.na(.), median(., na.rm = TRUE), .)))
>
> # Now, use mice to impute the 'STARS' column using decision trees (method 'cart')
> # First, set up the method 'cart' only for the 'STARS' column
> meth <- rep("", ncol(wine))
> names(meth) <- colnames(wine)
> meth['STARS'] <- 'cart'  # Set 'cart' only for 'STARS'
> meth[meth == ""] <- " "  # Set other methods to " " to skip imputation since they are already imp
uted
>
> # Perform the imputation for 'STARS' only
> imputed_data <- mice(wine, method = meth, m = 1, maxit = 5, seed = 123, printFlag = FALSE)
>
> # Create the completed data
> wine <- complete(imputed_data)
> # Calculate the number of missing values for each column
> missing_values <- sapply(wine, function(x) sum(is.na(x)))
>
> # Print the table of missing values
> print(missing_values)
        Purchase            STARS       FixedAcidity    VolatileAcidity          CitricAcid
               0                0                  0                  0                   0
   ResidualSugar        Chlorides  FreeSulfurDioxide TotalSulfurDioxide             Density
               0                0                  0                  0                   0
              pH        Sulphates            Alcohol        LabelAppeal           AcidIndex
               0                0                  0                  0                   0
```

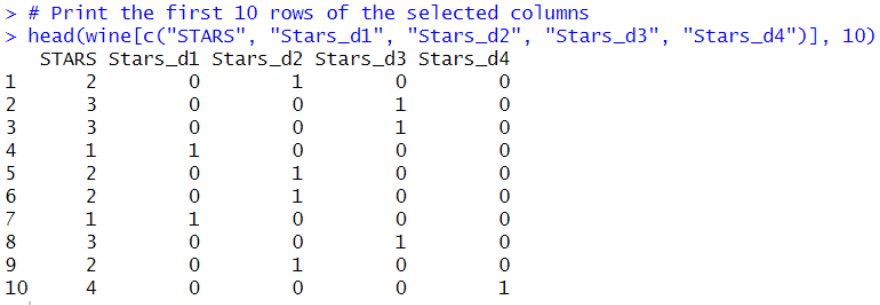




```
Results for STARS :

        Pearson's Chi-squared test

data:  contingency_table
X-squared = 2567.4, df = 3, p-value < 2.2e-16
```

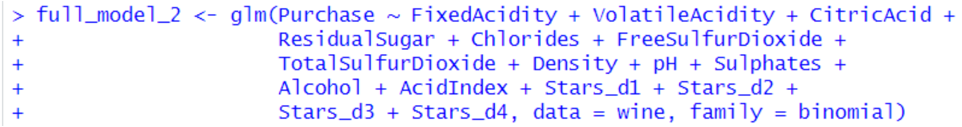
The Chi-squared test results for the STARS variable show a Chi-square statistic (X-squared) of 2567 with 3 degrees of freedom and a p-value less than 2.2e-16. This extremely small p-value indicates that there is a highly significant association between the STARS variable and the Purchase variable. In practical terms, the number of stars has a statistically significant impact on the purchase frequency. Previously, the Chi-squared test results for the STARS variable show a Chi-square statistic (X-squared) of 1130 with 3 degrees of freedom and a p-value less than 2.2e-16.

Dummy Code STARS

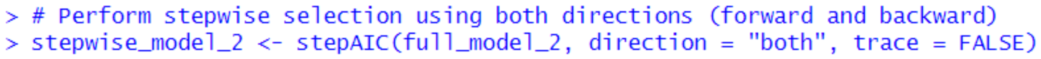


```
> # Print the first 10 rows of the selected columns
> head(wine[c("STARS", "Stars_d1", "Stars_d2", "Stars_d3", "Stars_d4")], 10)
   STARS Stars_d1 Stars_d2 Stars_d3 Stars_d4
1      2        0        1        0        0
2      3        0        0        1        0
3      3        0        0        1        0
4      1        1        0        0        0
5      2        0        1        0        0
6      2        0        1        0        0
7      1        1        0        0        0
8      3        0        0        1        0
9      2        0        1        0        0
10     4        0        0        0        1
```

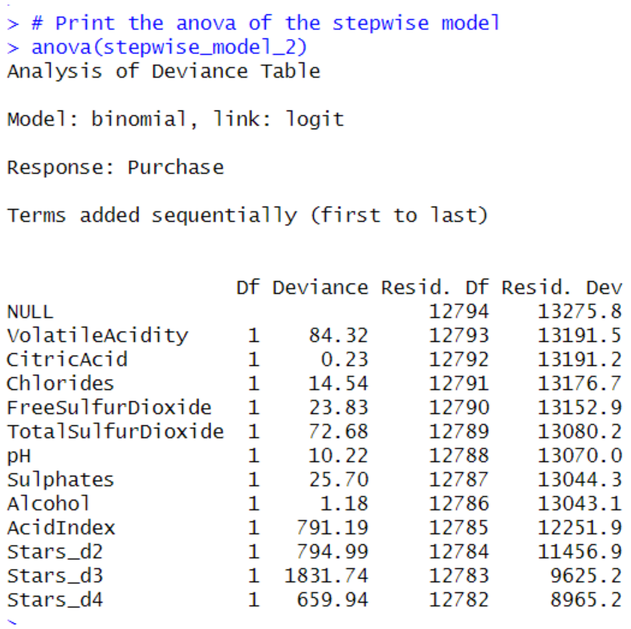
Creating the Linear Regression Model full_model_2, followed by creating stepwise_model_2:



```
> full_model_2 <- glm(Purchase ~ FixedAcidity + VolatileAcidity + CitricAcid +
+                     ResidualSugar + Chlorides + FreeSulfurDioxide +
+                     TotalSulfurDioxide + Density + pH + Sulphates +
+                     Alcohol + AcidIndex + Stars_d1 + Stars_d2 +
+                     Stars_d3 + Stars_d4, data = wine, family = binomial)
```



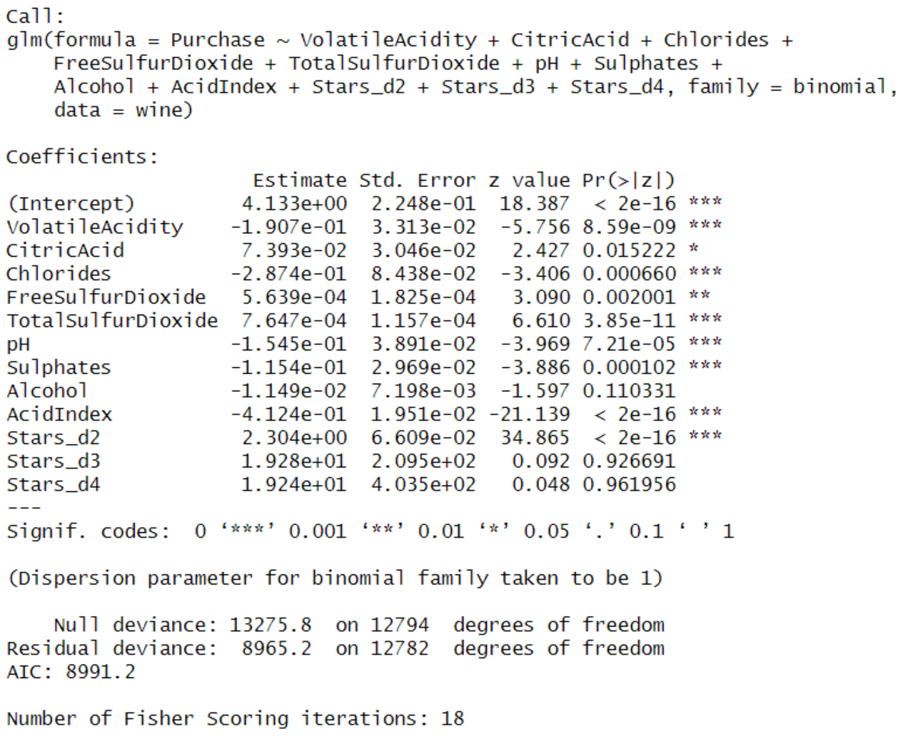
```
> # Perform stepwise selection using both directions (forward and backward)
> stepwise_model_2 <- stepAIC(full_model_2, direction = "both", trace = FALSE)
```



```
> # Print the anova of the stepwise model
> anova(stepwise_model_2)
Analysis of Deviance Table

Model: binomial, link: logit

Response: Purchase

Terms added sequentially (first to last)


                   Df Deviance Resid. Df Resid. Dev
NULL                               12794    13275.8
VolatileAcidity     1    84.32     12793    13191.5
CitricAcid          1     0.23     12792    13191.2
Chlorides           1    14.54     12791    13176.7
FreeSulfurDioxide   1    23.83     12790    13152.9
TotalSulfurDioxide  1    72.68     12789    13080.2
pH                  1    10.22     12788    13070.0
Sulphates           1    25.70     12787    13044.3
Alcohol             1     1.18     12786    13043.1
AcidIndex           1   791.19     12785    12251.9
Stars_d2            1   794.99     12784    11456.9
Stars_d3            1  1831.74     12783     9625.2
Stars_d4            1   659.94     12782     8965.2
```

Summary:

- For each variable added in the stepwise model, the table shows the reduction in deviance, indicating the contribution of each variable to the model's explanatory power.
- The "NULL" model's deviance represents the model with no predictors, just an intercept. As variables are added, the residual deviance decreases, showing improvement in the model.
- Variables like AcidIndex, Stars_d2, Stars_d3, and Stars_d4 are key contributors to the model, based on the substantial decrease in deviance they caused when added.



```
Call:
glm(formula = Purchase ~ VolatileAcidity + CitricAcid + Chlorides +
    FreeSulfurDioxide + TotalSulfurDioxide + pH + Sulphates +
    Alcohol + AcidIndex + Stars_d2 + Stars_d3 + Stars_d4, family = binomial,
    data = wine)

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)         4.133e+00  2.248e-01  18.387  < 2e-16 ***
VolatileAcidity    -1.907e-01  3.313e-02  -5.756 8.59e-09 ***
CitricAcid          7.393e-02  3.046e-02   2.427 0.015222 *
Chlorides          -2.874e-01  8.438e-02  -3.406 0.000660 ***
FreeSulfurDioxide   5.639e-04  1.825e-04   3.090 0.002001 **
TotalSulfurDioxide  7.647e-04  1.157e-04   6.610 3.85e-11 ***
pH                 -1.545e-01  3.891e-02  -3.969 7.21e-05 ***
Sulphates          -1.154e-01  2.969e-02  -3.886 0.000102 ***
Alcohol            -1.149e-02  7.198e-03  -1.597 0.110331
AcidIndex          -4.124e-01  1.951e-02 -21.139  < 2e-16 ***
Stars_d2            2.304e+00  6.609e-02  34.865  < 2e-16 ***
Stars_d3            1.928e+01  2.095e+02   0.092 0.926691
Stars_d4            1.924e+01  4.035e+02   0.048 0.961956
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13275.8  on 12794  degrees of freedom
Residual deviance:  8965.2  on 12782  degrees of freedom
AIC: 8991.2

Number of Fisher Scoring iterations: 18
```

CitricAcid, TotalSulfurDioxide and FreeSulfurDioxide: All three have positive coefficients and are significant, suggesting their increase is associated with an increased likelihood of purchase.

Sulphates, AcidIndex: Significant negative coefficients, indicating their increase is associated with a decreased likelihood of purchase.

Stars_d2: Negative and highly significant, implying that this category has a lower likelihood of purchase compared to the baseline category.

Stars_d3 and Stars_d4: Not significant (high p-values), suggesting these variables do not have a statistically significant effect on the likelihood of purchase in the presence of other variables. However, these will have to be included in the model because we have categorical data where the 'STARS' variable is represented as a series of dummy variables (one for each category). Omitting any of these dummy variables could lead to an incomplete representation of the categorical effect on the response variable. Moreover, their inclusion is essential for the interpretive completeness of the 'STARS' categorical predictor, ensuring that the model fully captures the potential influence of each category within 'STARS' on the outcome.

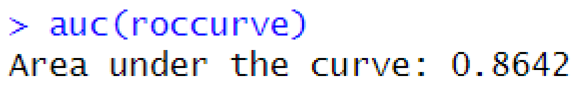
Model Fit and Diagnostics:

Deviance: The null deviance and residual deviance indicate how well the model fits the data compared to a null model. A substantial drop from null to residual deviance suggests the model fits the data well.

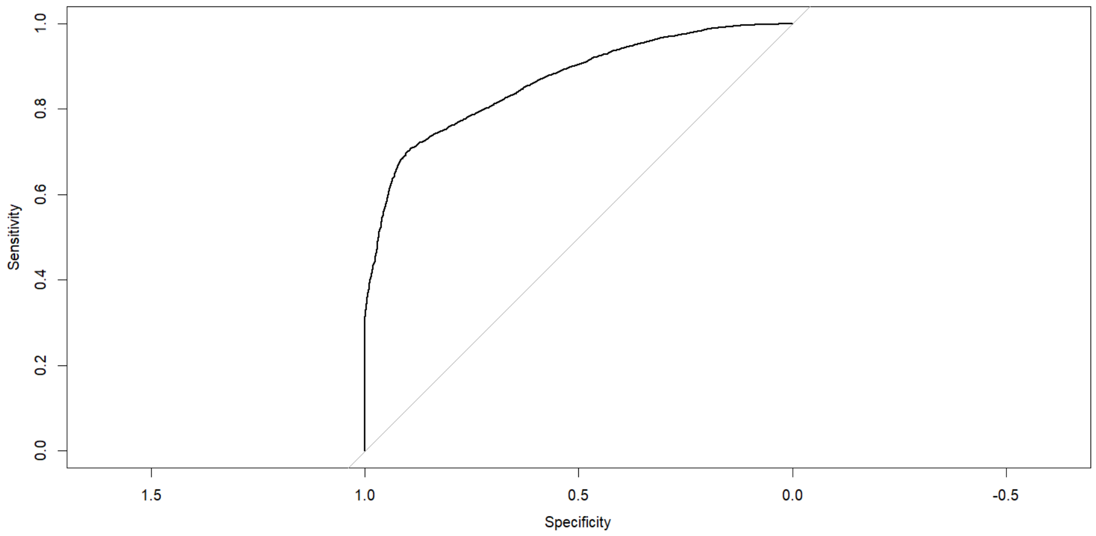
AIC (Akaike Information Criterion: 8991.2): A measure of the relative quality of the model for the given set of data. Lower AIC values indicate a better fit.

Conclusion

The stepwise_model_2 seems to fit the data well, as indicated by the significant reduction in deviance from the null model. Most predictors are statistically significant, with their respective signs indicating whether they increase or decrease the likelihood of wine purchase.



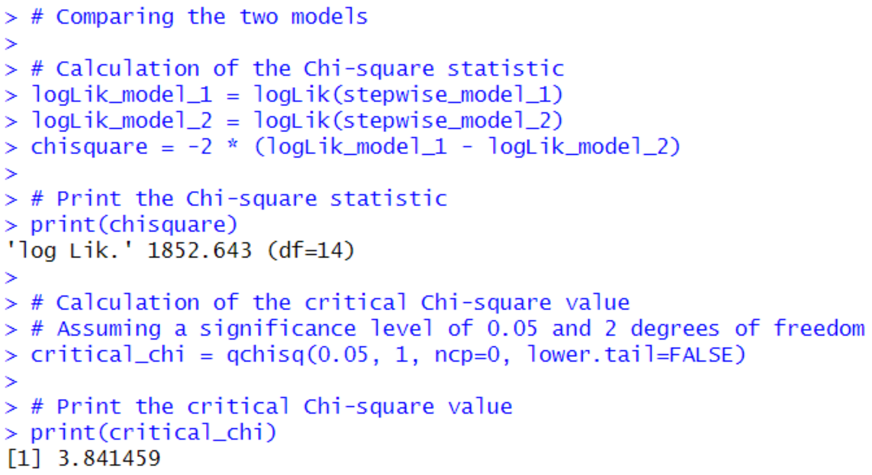
```
> auc(roccurve)
Area under the curve: 0.8642
```



Area under the curve: 0.8642

This level of AUC indicates that when presented with a pair consisting of one instance where a purchase was made and one where it was not, the model can correctly identify the purchase instance over 86.42% of the time.

Comparison of stepwise_model_1 and stepwise_model_2:



```
> # Comparing the two models
>
> # Calculation of the Chi-square statistic
> logLik_model_1 = logLik(stepwise_model_1)
> logLik_model_2 = logLik(stepwise_model_2)
> chisquare = -2 * (logLik_model_1 - logLik_model_2)
>
> # Print the Chi-square statistic
> print(chisquare)
'log Lik.' 1852.643 (df=14)
>
> # Calculation of the critical Chi-square value
> # Assuming a significance level of 0.05 and 2 degrees of freedom
> critical_chi = qchisq(0.05, 1, ncp=0, lower.tail=FALSE)
>
> # Print the critical Chi-square value
> print(critical_chi)
[1] 3.841459
```

The Chi-square statistic of 1852.643, being significantly larger than the critical value of 3.841459, indicates that stepwise_model_2 fits the data better than stepwise_model_1 at a 0.05 significance level. Stepwise_model_2, with a higher AUC of 0.8642 compared to 0.7696 for stepwise_model_1, demonstrates superior predictive performance. Additionally, it achieves this with fewer variables (12 compared to 13), aligning with the principle of parsimony.

In summary, stepwise_model_2 is not only more efficient in terms of variables used but also more effective in predictive accuracy, making it the preferable model.

Comparison of stepwise_model and stepwise_model_2:

Given that stepwise_model and stepwise_model_2 have an equal count of variables, and consequently the same degrees of freedom, the use of a Chi-square test for their comparison is inappropriate. The reason lies in the fact that the degrees of freedom for such a test, determined by the difference in the number of parameters between the two models, would be zero (df = number of parameters in stepwise_model - number of parameters in stepwise_model_2). A zero degree of freedom renders the Chi-square statistic uncomputable, thus making the test unsuitable for contrasting stepwise_model and stepwise_model_2.

However, considering other metrics, stepwise_model_2 shows superiority. It boasts an AUC of 0.8642, surpassing the 0.7718 AUC of stepwise_model. Additionally, the AIC of stepwise_model_2 stands at 8991.2, markedly lower than the 10754 AIC of stepwise_model. These figures indicate that stepwise_model_2 has a better balance of model fit and complexity.

In conclusion, stepwise_model_2 emerges as the most effective among the three models.

- What conclusions do you draw from having conducted this analysis? What did you learn about the wine world through your modeling endeavor? What actions can you recommend to anyone involved in this field? How did your perspective on modeling change? Discuss anything else you wish to discuss.

The analysis conducted using logistic regression models in the wine industry has yielded valuable insights and led to several important conclusions:

1. Effectiveness of Model Simplification:
   - Altering the AcidIndex in `stepwise_model` to TransformedAcidIndex in `stepwise_model_1` did not yield a more efficient model. This adjustment led to an increase in the number of predictors without improving the Area Under the Curve (AUC) score, which remained similar to that of the original `stepwise_model`. This highlights the importance of achieving a balance between a model's predictive accuracy and its simplicity. A simpler model is often more interpretable and less prone to overfitting, making it more applicable in real-world scenarios.
   - The application of decision-tree-based imputation for the STARS variable has significantly enhanced `stepwise_model_2` over `stepwise_model_1`. This approach significantly improved the model's performance, indicating the value of strategic imputation techniques in predictive modeling.
2. Learnings About the Wine Market:
   - The variables selected for inclusion in our predictive models underscore the significant role that the chemical composition of wine plays in swaying consumer purchase decisions. Attributes such as Volatile Acidity, Chlorides, and Sulphates are not merely incidental; they are key determinants that drive consumer preferences.
   - Adjustments to the AcidIndex in our modeling suggest that there are complex, possibly non-linear, interactions between the acid content of wines and the likelihood of consumer purchases. This complexity points to a sophisticated consumer palate that discerns and evaluates the nuanced profiles of acidity in wines.
   - Importantly, the STAR ratings, reflecting direct consumer feedback, emerge as a critical factor within our models. The deliberate strategy to impute missing STAR values, which constituted a substantial 25% of the dataset, using Decision Trees has been instrumental in `stepwise_model_2`'s significant enhancement over its predecessor. This sophisticated imputation not only compensates for missing data but also captures the essence of consumer sentiment more accurately. By doing so, it substantially improves the model's ability to predict purchase behavior based on customer ratings, affirming the pivotal influence that perceived quality, as encapsulated by STAR ratings, exerts on consumer preferences. The improvement is a testament to the importance of consumer-perceived quality in influencing purchasing decisions, and it validates the effort invested in accurately imputing these pivotal STAR ratings.
3. Implications for Stakeholders:
   - Wine producers and marketers should prioritize understanding and leveraging these key chemical attributes and consumer ratings. Educating consumers about these aspects and using data analytics for product development can lead to more successful outcomes.
   - Simplifying complex wine attributes for consumers and focusing on sustainable practices can be beneficial. Additionally, keeping abreast of global market trends and building strong networks can provide a comprehensive market understanding.
4. Evolution in Modeling Perspective:
   - The analysis emphasizes the value of parsimony in models. A simpler model with similar predictive power but greater ease of use and interpretation can often be more valuable than a more complex one.

- The distinction between statistical and practical significance is crucial. It's essential to interpret models within the context of their real-world application.
- The dynamic nature of consumer preferences in the wine market necessitates continuous adaptation and updating of models to reflect changing trends.

5. Broader Insights and Future Directions:
   - Integration of modeling with modern technologies like AI can offer deeper insights into consumer behavior.
   - Ethical and sustainable practices are increasingly important in the wine industry, appealing to environmentally conscious consumers and ensuring long-term sustainability.
   - Understanding global trends and engaging in collaboration and networking are key for a holistic understanding of the wine market.

In summary, this modeling exercise in the wine industry has provided a multifaceted understanding of consumer behavior and preferences. It highlights the importance of data-driven decision-making, the need for balance between complexity and usability in models, and the dynamic nature of market trends. This analysis not only informs specific strategies for those involved in the wine industry but also offers broader lessons on the application and interpretation of statistical models in market analysis.