Computer Assignment 4

Gabriel Lindqvist, Jiahui Li

2024-12-21

In this exercise you will carry out several analyses of The ICU Study data (Hosmer Lemeshow (1989): *Applied Logistic Regression*). The data consists of records on 200 patients admitted to an Intensive Care Unit (ICU) during a certain period and the overall task is to use multiple logistic regression and decision tree models to identify factors that affect the survival of such patients.

Exercise 4:1 (Multiple logistic regression)

Which variables affect the survival of a patient admitted to ICU? And how do they affect the survival? To answer that, you should create a multiple logistic regression model for the probability to not survive (i.e.the probability to die). For variable selection, stepwise methods can be used. We will assume that there are no interactions between the variables (you are free to investigate whether this assumption is reasonable).

Data prepration

We start by reading and preparing the data. The preparations includes combining categories for two variables, with `v5` (Ethnicity) was already given by the instructions. The new encoding are `0=other` and `1=white`

```
##
##   1   2   3
## 175  15  10
```

```
##
##   0   1
##  25 175
```

After further investigation we found that `v21` (Consciousness level) had low counts for the levels 1=unconscious (5 obs.) and 2=coma (10 obs.). Since coma is a more specific type of unconsciousness, we can combine the two categories. The new encoding are `0=awake` and `1=unconscious`.

```
##
##   0   1   2
## 185   5  10
```

```
##
##   0   1
## 185  15
```

Task 1

Report the model selection process in short. Based on your chosen model, which factors affect the probability to not survive? Report odds ratio with confidence interval for the most important variables/factors and interpret them. Use the variable names in the table above when you report this (not V3,V4,…)

1.1 Fitting Multiple Logistic Regression Models

We start by fitting the empty model and the full model (excluding the first variable: Patient). These will be used as our boundaries when applying the stepwise model selection.

```r
m_empty <- glm(v2~1, family=binomial,data=data4)
m_full <- glm(v2~ .-v1, family=binomial,data=data4)
```

1.2 Stepwise Model Selection

In this stepwise regression process, both **AIC** (Akaike Information Criterion) and **BIC** (Bayesian Information Criterion) were used to select the best model. The basic approach for both is the same, with variables being added or removed step by step to optimize the model. However, **AIC** focuses more on model fit, while **BIC** penalizes model complexity more strongly, generally favoring simpler models.

AIC Stepwise Regression: The AIC stepwise regression started with an empty model (only the intercept), and variables were added step by step, selecting those that reduced the AIC value.

The best model (based on AIC) included the predictors:

- Consciousness level `v21` (Binary)
- Type of Admission `v14` (Binary)
- Age `v3` (Continuous)
- Cancer `v7` (Binary)
- Blood pressure `v11` (Continuous)
- Blood carbon dioxide `v18` (Binary)
- Blood pH `v17` (Binary)

`v2 ~ v21 + v14 + v3 + v7 + v11 + v18 + v17`

This model has an AIC value of **144.44**, including more variables, as AIC tends to favor models with better fit, even if they are more complex.

BIC Stepwise Regression: The BIC stepwise model selection also started with an empty model, but in selecting variables, BIC penalizes model complexity more strongly. Therefore, it tends to select models with fewer variables.

The best model (based on BIC) included the predictors:

- Consciousness level `v21` (Binary)
- Type of Admission `v14` (Binary)

- Age `v3` (Continuous)
- Cancer `v7` (Binary)

This model has an AIC value of **165.63**, and a smaller BIC, which is why it includes fewer variables compared to the AIC model.

1.3 Final Model Choice

Based on the results from both AIC and BIC:

- **If the goal is to achieve better model fit and the complexity is not a primary concern**, the **AIC model** should be chosen, as it has a lower AIC value (144.44) and includes more explanatory variables.
- **If the goal is to avoid potential overfitting and prefer a simpler model**, the **BIC model** should be selected, as it includes fewer variables due to the stronger penalty for complexity.

Between the two, **the AIC model** provides a better fit and includes more variables, while **the BIC model** is more parsimonious. Thus, **the final chosen model** should be the **AIC model with AIC=144.44**. This model is denoted as `mstep1_AIC`, which used forward selection.

1.4 Report odds ratio with confidence interval for the most important variables/factors and interpret them

Based on the `mstep1_AIC` model, we can report the **odds ratios (OR)** and their **95% confidence intervals (CI)** for the most significant variables in predicting survival.

```
##                Variable   Odds_Ratio     CI_Lower    CI_Upper
## (Intercept) (Intercept)  0.004861388 2.081749e-04   0.1135252
## v21                 v21 89.678346769 1.228637e+01 654.5632782
## v14                 v14 20.648619829 3.175787e+00 134.2550604
## v3                   v3  1.043608337 1.015944e+00   1.0720256
## v7                   v7 11.045449672 1.937500e+00  62.9687591
## v11                 v11  0.985566747 9.719798e-01   0.9993436
## v18                 v18  0.105060635 1.496223e-02   0.7377066
## v17                 v17  6.387161506 1.151334e+00  35.4335204
```

Model Results:

- **v21 (Consciousness Level)**: **Odds ratio = 89.68** (CI: 12.29 to 654.56). Being awake significantly increase the odds of survival by **89.68 times** compared to patients being unconscious, while other predictors are fixed.
- **v14 (Type of Admission)**:
  Odds ratio = 20.65 (CI: 3.18 to 134.26).
  Patients admitted through acute care have **20.65 times** higher odds of non-survival compared to those admitted through non-acute care, holding other predictors constant.
- **v3 (Age)**:
  Odds ratio = 1.04 (CI: 1.016 to 1.072).
  Each additional year of age increases the odds of non-survival by approximately **4%** (1.04 times), holding other predictors constant. This indicates a small but statistically significant negative effect of age on survival.

- **v7 (Cancer)**:
  Odds ratio = 11.05 (CI: 1.94 to 62.97).
  Patients with cancer have **11.05 times** higher odds of non-survival compared to those without cancer, holding other predictors constant. This is a significant risk factor for non-survival.
- **v11 (Blood Pressure)**:
  Odds ratio = 0.99 (CI: 0.972 to 0.999).
  Each unit increase in blood pressure slightly reduces the odds of non-survival by approximately **1%** (0.99 times), adjusting for other predictors. This protective effect is small but statistically significant.
- **v18 (Blood Carbon Dioxide)**:
  Odds ratio = 0.105 (CI: 0.015 to 0.738).
  Patients with blood carbon dioxide levels above 45 have significantly lower odds of non-survival, at only **10.5%** of the odds of patients with lower carbon dioxide levels, adjusting for other predictors. This suggests a strong protective effect of higher carbon dioxide levels on survival.
- **v17 (Blood pH)**:
  Odds ratio = 6.39 (CI: 1.15 to 35.43).
  Patients with higher blood pH levels (>7.25) have **6.39 times** higher odds of non-survival compared to patients with lower pH levels, after adjusting for other predictors. This indicates a significant negative impact of elevated pH levels on survival.

Summary:

1. **Consciousness level (v21)** and **type of admission (v14)** are key determinants of non-survival risk. Unconsciousness and acute admission significantly increase the odds of non-survival.
2. **Age (v3)** and **cancer status (v7)** also significantly increase the odds of non-survival, highlighting these as high-risk groups.
3. **Blood pressure (v11)** and **carbon dioxide levels (v18)** have protective effects on survival. Higher carbon dioxide levels, in particular, strongly reduce the odds of non-survival.
4. **Blood pH (v17)** is associated with a substantial increase in non-survival risk, suggesting that maintaining acid-base balance is a critical factor for survival.

Task 2

How well does your chosen model fits the data? In assignment 3, the deviance was used to assess model fit but since the data now is on individual level, deviance is not suitable and instead, the Hosmer-Lemeshow goodness-of-fit test (Agresti 5.2.5) should be used. Perform this test in R (see the R-instruction) and interpret the result.

```
## Warning: package 'ResourceSelection' was built under R version 4.2.3
```

```
## ResourceSelection 0.3-6   2023-06-27
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  data4$v2, predprob
## X-squared = 7.2202, df = 8, p-value = 0.5131
```

The **p-value** of **0.5131** from the **Hosmer-Lemeshow test** indicates that the model fits the data well. Since the p-value is greater than 0.05, we fail to reject the null hypothesis, meaning there is no significant difference between the observed and predicted values. This suggests the model's predictions align closely with the actual outcomes.

Task 3

Now we shift the focus to prediction.How well does your multiple logistic regression models predict the outcome?There exists several different metrics of prediction performance for binary responses and here we will use the following:‘Accuracy’, ’Sensitivity (True positive rate),’Specificity (True negative rate)and ’AUC (area under the ROC curve).To explain these measures we first notice that the model predictions (or fitted values)are probabilities to not survive.If we let patients be classified as ’Survive’or ’Not survive’when theirs predicted probability is below or above a certain cut-off value (threshold value),a confusion matrix (a 2x2 classification table)can be constructed,as shown below.

Actual	Predicted Survive	Predicted Not Survive	Total
Actual Survive	n11	n12	n1.
Actual Not Survive	n21	n22	n2.
Total			n..

Calculate the metrics:

- **Accuracy** = $\frac{n_{11}+n_{22}}{n}$
- **Sensitivity (True Positive Rate)** = $\frac{n_{11}}{n_{1.}}$
- **Specificity (True Negative Rate)** = $\frac{n_{22}}{n_{2.}}$

```
##              Predicted
## Actual        Survive Not-survive
##   Survive         157           3
##   Not-survive      23          17
```

```
##              Predicted
## Actual        Survive Not-survive Sum
##   Survive         157           3 160
##   Not-survive      23          17  40
##   Sum             180          20 200
```

```
## Accuracy:  0.87
```

```
## Sensitivity:  0.98125
```

```
## Specificity:  0.425
```

Interpretation of the Results:

1. **Confusion Matrix**:
   - **True Negatives (TN)** = 157: These are the cases where the model correctly predicted “Not Survive” (Actual = 0 and Predicted = 0).
   - **False Positives (FP)** = 3: These are the cases where the model incorrectly predicted “Survive” (Actual = 0 but Predicted = 1).
   - **False Negatives (FN)** = 23: These are the cases where the model incorrectly predicted “Not Survive” (Actual = 1 but Predicted = 0).

- **True Positives (TP)** = 17: These are the cases where the model correctly predicted "Survive" (Actual = 1 and Predicted = 1).

2. **Metrics**:

    - **Accuracy** = **0.87**: This means that 87% of the total predictions were correct. Accuracy is the overall proportion of correct predictions, including both "Survive" and "Not Survive" cases.
    - **Sensitivity (True Positive Rate)** = **0.98125**: This indicates that 98.13% of actual survivors (Actual = 1) were correctly predicted as "Survive" (Predicted = 1). A high sensitivity means the model is good at identifying survivors.
    - **Specificity (True Negative Rate)** = **0.425**: This indicates that only 42.5% of actual non-survivors (Actual = 0) were correctly predicted as "Not Survive" (Predicted = 0). A low specificity means that the model is not very good at identifying non-survivors, leading to a relatively high false positive rate.

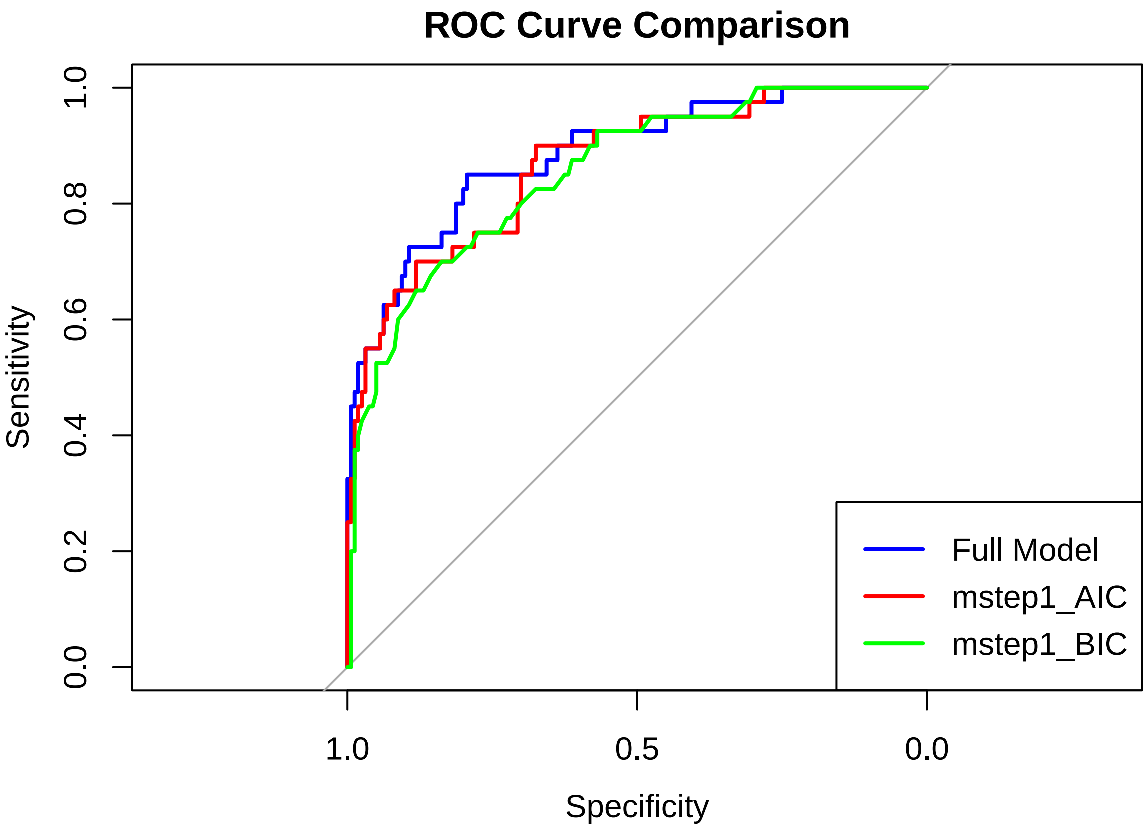
Conclusion:

- The model is performing well in terms of **sensitivity** (98.13%), meaning it is effective at identifying individuals who survive.
- However, the model has a **low specificity** (42.5%), meaning it struggles to correctly identify individuals who do not survive, as evidenced by a relatively high number of false positives.
- The **accuracy** of 87% suggests that the overall performance is reasonable, but the low specificity indicates that there might be room for improvement, especially in predicting non-survivors.

Task 4

As shown above, the sensitivity and specificity depend on the cut-off setting. A ROC-curve is a plot of sensitivity (true positive rate) against 1-specificity (false postive rate)a t each possible cut-off setting. AUC is the area under the ROC curve and can be used as a summary metsure of a binary classifier's performance. AUC=1 indicates perfect classification and AUC=0.5 indicates that the model's performance is no better than random guessing. Create plots of ROC curves and calculate the AUC for the full model and two more models of your choice (see the R-instruction) Which of the model performs best based on AUC?

```
## Warning: package 'pROC' was built under R version 4.2.3

## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```



```
## AUC for Full Model:  0.8846875

## AUC for mstep1_AIC:  0.8678125

## AUC for mstep1_BIC:  0.8532812
```

- **Full Model** performs the best with the highest AUC, indicating the strongest predictive ability.
- **mstep1__AIC** follows closely behind, but with slightly lower performance.
- **mstep1__BIC** has the lowest performance in terms of AUC.

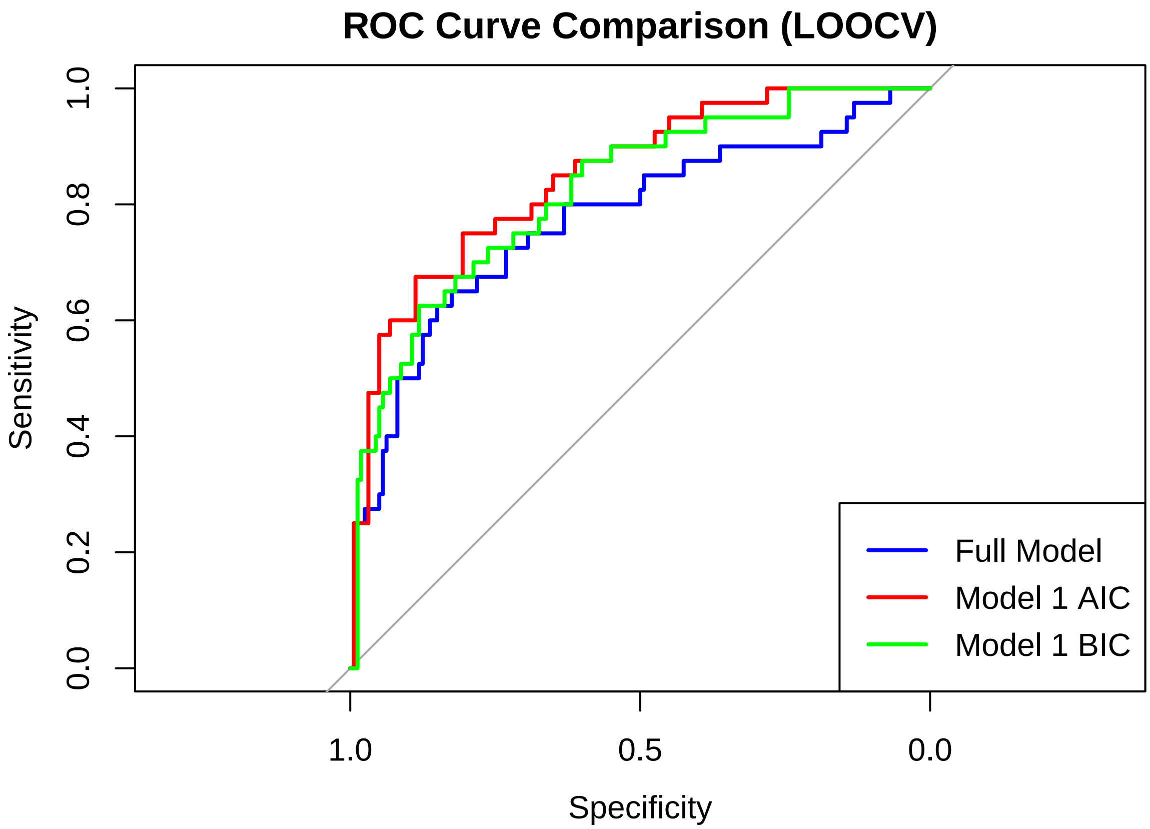
The full model has the highest AUC but these three models' AUC are very close.

Task 5

The AUC-values obtained in the previous exercise may be subject to optimistic bias, as they were computed for the same dataset that was used for model fitting. This might result in overfitting, where the model become well adapted to the training data but may not perform as good to new data. To get a more reliable AUC-value, a validation method can be used. We will use the Leave-One-Out-Cross-Validation (LOOCV). For LOOCV, a model is fitted (or trained) using all observations except one, which is held out for validation. The model is then used to make a prediction on the held-out observation. This process is repeated for each observation, resulting in LOOCV predicted probabilities for each observation (see the R-instruction). These LOOCV predicted probabilities can then be used to calculate AUC.Do this for the full model and two more models of your choice (the same models as in the previous exercise). Which of the model performs best based on LOOCV-adjusted AUC? Compare with the result in the previous exercise.

Note: The stepwise procedure should not be performed within the cross-validation 1oop.

```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```



```
## AUC for Full Model:  0.7746875
```

```
## AUC for Model 1 (AIC):  0.8495312
```

```
## AUC for Model 1 (BIC):  0.824375
```

- The **AIC model** (AUC = 0.8495312) performs the best, with the strongest discriminative power.
- The **BIC model** (AUC = 0.824375) performs well but is slightly weaker than the AIC model.
- The **Full model** (AUC = 0.7746875) performs relatively poorly, indicating that its predictive ability is not as strong as the AIC and BIC models.

Based on the **LOOCV-adjusted AUC**, the **AIC model** is the best choice.

Exercise 4:2 Decision Tree

Task 1

We start off by reporting the changes in the decision trees while adjusting the `cp`, for the two splitting methods. This was done in the following way:

1. Use the formula `v2 ~. - v1` and set `cp = 0.1` as our starting value.
2. Generate and plot the two decision trees for each splitting method.
3. Write down the observations.
4. Lower the `cp` value and go to step 2.

Observations and Choosing Tree:

- `cp = 0.1`: Both splitting methods generate the same tree and the height of the tree is 1, with `v21` (Consciousness level) as the root.
- `cp = 0.05`: Both splitting methods generate the same tree and the height of the tree is now 2, where the left node has two children `v11 >= 88` (blood pressure).
- `cp = 0.005`: Information splitting method still has the same tree as previously. Gini splitting method has grown larger, including the predictors: `v3 < 75` (age), `v12 < 78` (heart rate) and `v11 >= 137` as a leaf.
- `cp = 0.001`: Information splitting method has generated the largest tree and has now included the predictor `v14` (Type of Admission). The tree generated by using gini splitting method remains the same.
- `cp = 0.0005`: No notable changes.
- `cp = 0.0001`: Again, no notable changes, indicating that no further reduction in complexity parameter will let the tree grow larger.

```
library(rpart, quietly = TRUE)
```

```
## Warning: package 'rpart' was built under R version 4.2.3
```

```
library(rpart.plot, quietly = TRUE)
```

```
## Warning: package 'rpart.plot' was built under R version 4.2.3
```

```
# Chosen decision tree
tm_info <- rpart(
  v2 ~ . - v1,
  method = "class",
  data = data4,
  parms = list(split = "information"),
  cp = 0.05
  )

# Largest decision tree generated
tml_info <- rpart(
  v2 ~ . - v1,
  method = "class",
  data = data4,
  parms = list(split = "information"),
  cp = 0.001
  )
```

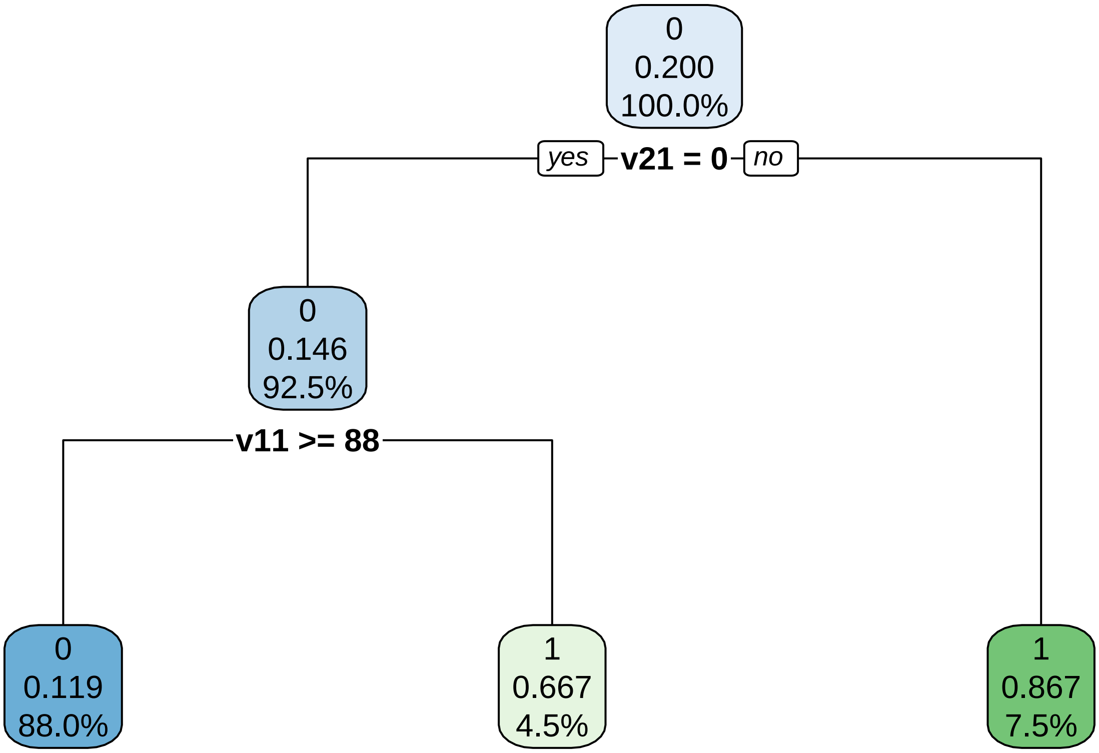
To decide on a "good" decision tree (without computing AUC or LOOCV), intuitively, we compare the predictors selected in the decision tree with the logistic regression chosen in the previous exercise. We also weigh in interpretation, with simpler decision trees are in favor.

A key difference from the observations is that the gini splitting method has included **Heart rate**, while excluded in the chosen logistic regression model. This does not necessarily imply that the predictor has no influence in patient survival, due to AIC-based model selection. Furthermore, **Heart rate** is treated as binary, while its continuous in the logistic regression model.

In contrast, the information splitting method is consistent with its inclusion of predictors (relative to the chosen logistic regression model). That is, all predictors found in the largest tree are also found in the chosen logistic regression model. However, this tree is rather complex with a height of 7. Thus, we decided to go with the simpler tree that only includes **Conciousness level** and **Blood pressure**.

Interpretation of Chosen Tree Model

First step is to plot the chosen decision tree.



In order to interpret the illustrated decision tree, we need to understand the structure of each node.

1. **Top Value (Binary Code)**: This is the so called *majority class*. That is, the decision tree assigns a class to each node based on the most frequent (majority) class among the observations in that node. If the majority of observations in a node have `v2 = 0` (survived), then the top value will be `0`. Conversely, if the majority of observations in a node have `v2 = 1` (did not survive), then the top value will be `1`.

2. **Middle Value (Proportion)**: Reflects the proportion of observations in the node that belong to class `1` (did not survive). The majority class is determined based on the proportion. For example, the right child of the root has a middle value of `0.867`, meaning 86.7% did not survive. Thus the top value is `1`

3. **Bottom Percentage**: Represents the proportion of the entire dataset that ends up at that specific node. This percentage is distributed between siblings. For example, `92.5%` of the total observations fall into the left child of the root, while `7.5%` fall into the right child of the root.

With that in mind, we can deduce the following:

- **Consciousness level**: This factor serves as the root in the decision tree, meaning it is the most important factor. Being unconscious (represented by the right child of the root), is strongly associated with non-survival.

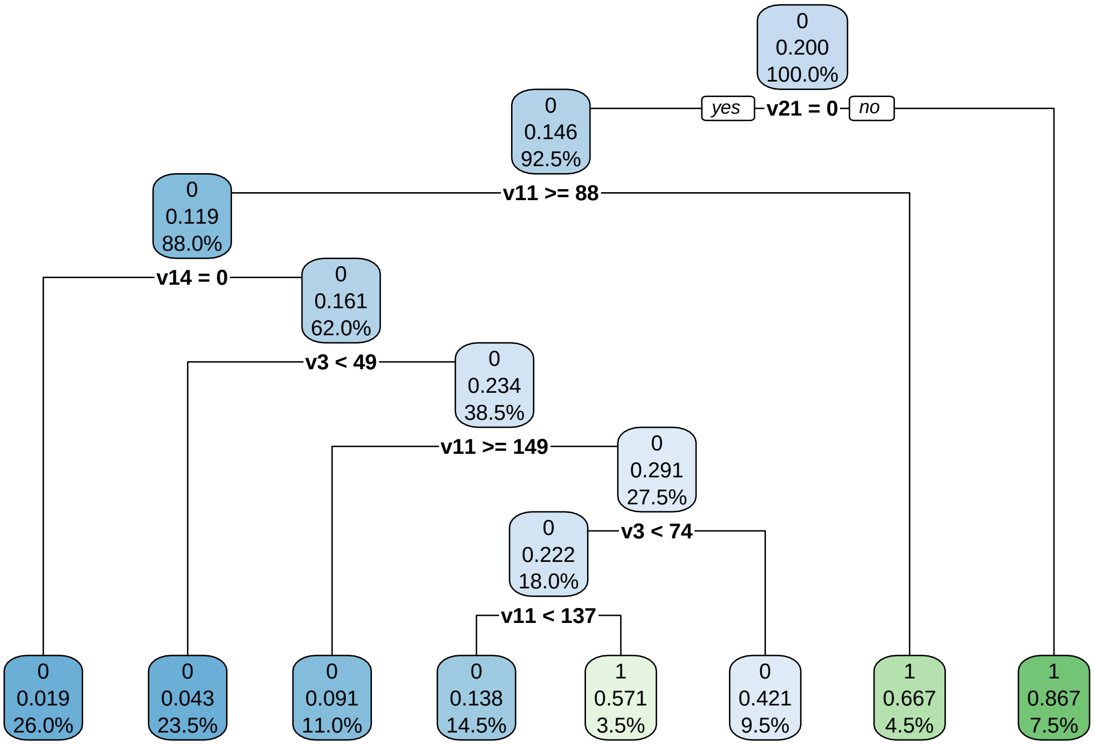
- **Blood Pressure**: Represented as the left child of the root. For patients that are conscious (`v21 = 0`), blood pressure is a key predictor for survival outcome. High blood pressure (≥ 88 mm Hg) is associated with better outcomes, while low blood pressure (< 88 mm Hg) indicates a higher risk of non-survival.

Task 2

In this task, we want to investigate the predictive power of our chosen tree model (AUC without cross validation) and compare the results with that for the multiple logistic regression models in the previous exercise.

For curiosity, we calculate the AUC for the largest decision tree model that information splitting method generated (see displayed plot below for a visualization of this decision tree)

```r
rpart.plot(tml_info, digits = 3)
```



```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## [1] "AUC for Chosen Tree Model: 0.724"

## [1] "AUC for Largest Tree Model: 0.8891"
```

Comparing AUC between the chosen tree model and the logistic regression models, the chosen tree model ranks at the bottom regarding predictive power. However, the largest tree model edges all three multiple logistic regression models, with the highest calculated AUC. Thus, the largest tree model generated with information splitting method performs best based on AUC.

Thoughts Regarding The Results

Based on intuition, the results are not surprising. The chosen tree model is simplistic, including only two predictors, which limits its ability to capture the full complexity of the relationship found in the data. The largest tree model includes all key predictors identified as significant in the AIC-based logistic regression model. In the sense that the decision tree has access to the same information as the AIC-based logistic regression model.

Task 3

In this task, we want to calculate the LOOCV-corrected AUC for our chosen tree model (Again, we include the other generated tree model).

```
## Setting levels: control = 0, case = 1

## Setting direction: controls > cases

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

## [1] "AUC for Chosen Tree Model: 0.5356"

## [1] "AUC for Largest Tree Model: 0.7469"
```

The cross-validated AUC for the largest tree model and the chosen tree model demonstrate reduced predictive accuracy compared to the logistic regression models in Exercise 4:1 Task 5.The reduced performance of the chosen tree model is expected, given its simplicity and inclusion of only two predictors, which limits its ability to capture the full complexity of the data.

In contrast, the poor performance of the largest tree model, ranking second to last in cross-validated AUC, is somewhat surprising. The results might suggest potential overfitting, since the model performs well on the training data, but generalizes poorly on unseen data.