Titanic Dataset

Winifred Sunday

2025-07-27

Using the passenger data we are to predict who are most likely to survive.

Loading the data

td <- read.csv("Titanic-Dataset.csv", stringsAsFactors = FALSE,na.strings = "")

Get a summary of the data structure and types

str(td)

## 'data.frame': 891 obs. of 12 variables:  
## $ PassengerId: int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Survived : int 0 1 1 1 0 0 0 0 1 1 ...  
## $ Pclass : int 3 1 3 1 3 3 1 3 3 2 ...  
## $ Name : chr "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)" "Heikkinen, Miss. Laina" "Futrelle, Mrs. Jacques Heath (Lily May Peel)" ...  
## $ Sex : chr "male" "female" "female" "female" ...  
## $ Age : num 22 38 26 35 35 NA 54 2 27 14 ...  
## $ SibSp : int 1 1 0 1 0 0 0 3 0 1 ...  
## $ Parch : int 0 0 0 0 0 0 0 1 2 0 ...  
## $ Ticket : chr "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803" ...  
## $ Fare : num 7.25 71.28 7.92 53.1 8.05 ...  
## $ Cabin : chr NA "C85" NA "C123" ...  
## $ Embarked : chr "S" "C" "S" "S" ...

Checking for missing values

any(is.na(td)) #Checking for missing values generally

## [1] TRUE

sapply(td, function(x) sum(is.na(x)))

## PassengerId Survived Pclass Name Sex Age   
## 0 0 0 0 0 177   
## SibSp Parch Ticket Fare Cabin Embarked   
## 0 0 0 0 687 2

Checking for duplicate values

anyDuplicated(td)

## [1] 0

Drop irrelevant columns

td <- td[ , !(names(td) %in% c("Ticket", "Cabin","Name","Parch")) ]  
str(td)

## 'data.frame': 891 obs. of 8 variables:  
## $ PassengerId: int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Survived : int 0 1 1 1 0 0 0 0 1 1 ...  
## $ Pclass : int 3 1 3 1 3 3 1 3 3 2 ...  
## $ Sex : chr "male" "female" "female" "female" ...  
## $ Age : num 22 38 26 35 35 NA 54 2 27 14 ...  
## $ SibSp : int 1 1 0 1 0 0 0 3 0 1 ...  
## $ Fare : num 7.25 71.28 7.92 53.1 8.05 ...  
## $ Embarked : chr "S" "C" "S" "S" ...

Replacing Missing values

numeric\_col <- names(td[sapply(td, is.numeric)])  
td[numeric\_col]<- data.frame(lapply(td[numeric\_col], function(x) ifelse(is.na(x), round(median(x, na.rm = T),0), x)))  
  
factor\_col <- names(td[sapply(td, is.character)])  
td[factor\_col]<- data.frame(lapply(td[factor\_col], function(x) ifelse(is.na(x) | x=="", names(which.max(table(x))), x)))  
  
td[factor\_col] <- data.frame(lapply(td[factor\_col], function(x) as.factor(x)))  
  
str(td)

## 'data.frame': 891 obs. of 8 variables:  
## $ PassengerId: int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Survived : int 0 1 1 1 0 0 0 0 1 1 ...  
## $ Pclass : int 3 1 3 1 3 3 1 3 3 2 ...  
## $ Sex : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...  
## $ Age : num 22 38 26 35 35 28 54 2 27 14 ...  
## $ SibSp : int 1 1 0 1 0 0 0 3 0 1 ...  
## $ Fare : num 7.25 71.28 7.92 53.1 8.05 ...  
## $ Embarked : Factor w/ 3 levels "C","Q","S": 3 1 3 3 3 2 3 3 3 1 ...

sapply(td, function(x) sum(is.na(x)))

## PassengerId Survived Pclass Sex Age SibSp   
## 0 0 0 0 0 0   
## Fare Embarked   
## 0 0

Summarizing numeric columns

numeric <- td[sapply(td, is.numeric)]  
lapply(numeric, function(x) summary(x))

## $PassengerId  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.0 223.5 446.0 446.0 668.5 891.0   
##   
## $Survived  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000 0.0000 0.0000 0.3838 1.0000 1.0000   
##   
## $Pclass  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.000 2.000 3.000 2.309 3.000 3.000   
##   
## $Age  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.42 22.00 28.00 29.36 35.00 80.00   
##   
## $SibSp  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.000 0.000 0.000 0.523 1.000 8.000   
##   
## $Fare  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.00 7.91 14.45 32.20 31.00 512.33

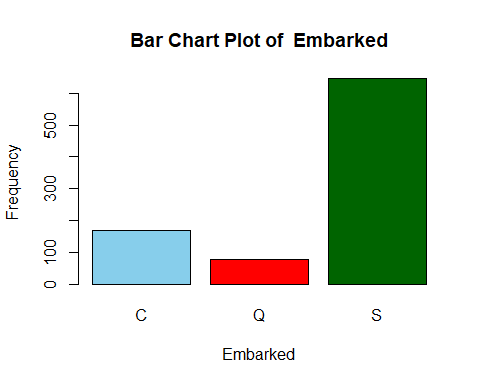
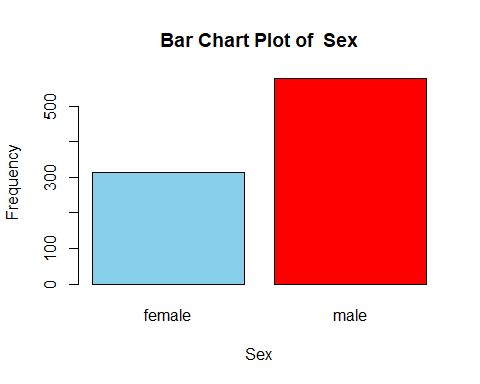
Summarizing non numeric columns

non\_numeric <- td[sapply(td, function(x) is.character(x) | is.factor(x))]  
lapply(non\_numeric,function(x) table(x))

## $Sex  
## x  
## female male   
## 314 577   
##   
## $Embarked  
## x  
## C Q S   
## 168 77 646

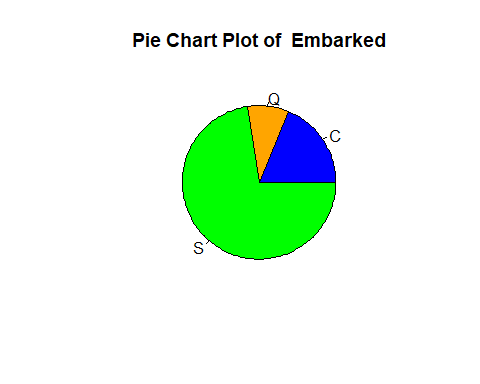
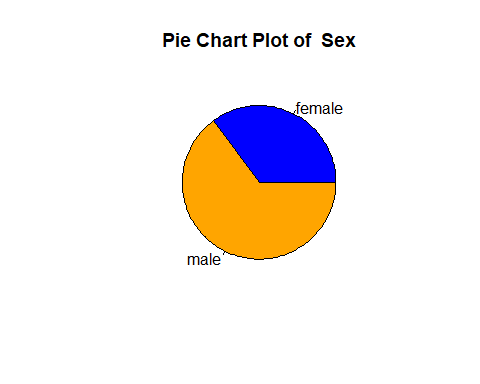
Visualizing non numeric columns

non\_numeric <- td[sapply(td, is.factor)]  
for(col in names(non\_numeric)){  
 barplot(table(td[col]), col = c("skyblue","red","darkgreen"), xlab = col, ylab = "Frequency", main = paste("Bar Chart Plot of ", col))  
}



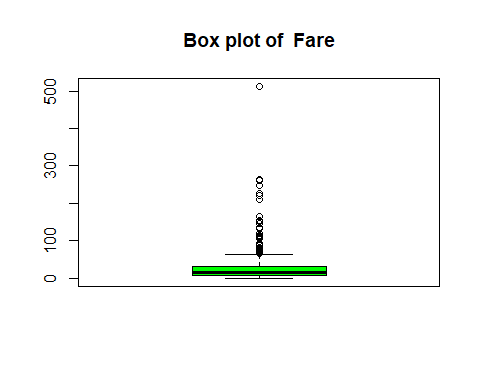
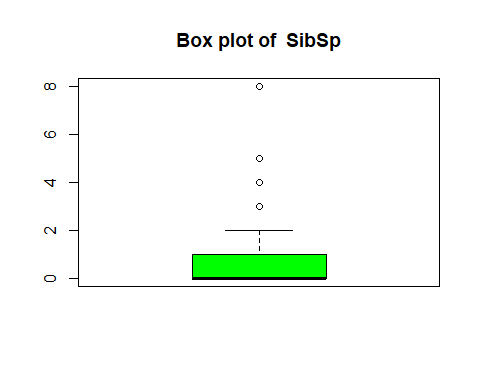
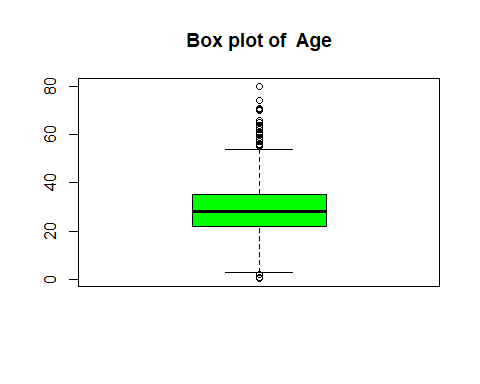
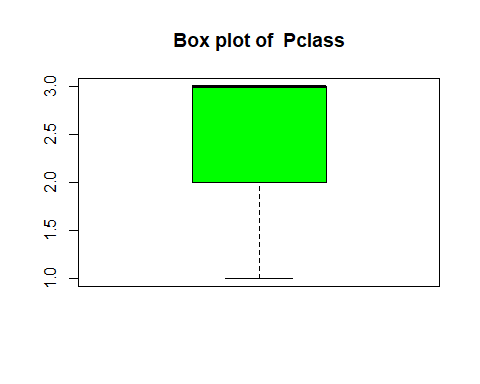
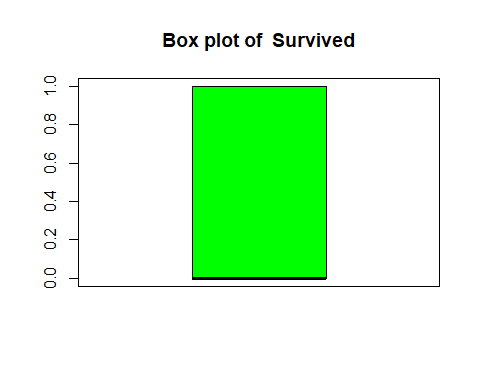
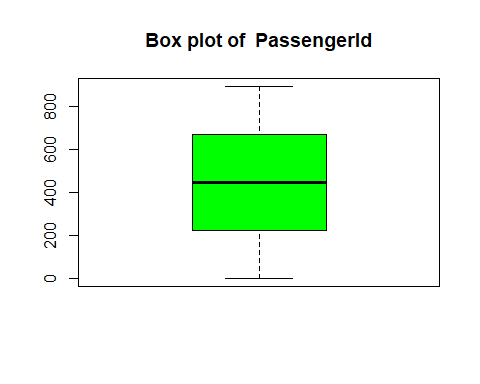
Pie Chart

non\_numeric <- td[sapply(td, is.factor)]  
for(col in names(non\_numeric)){  
 pie(table(td[col]), col = c("blue","orange","green"), main = paste("Pie Chart Plot of ", col))  
}

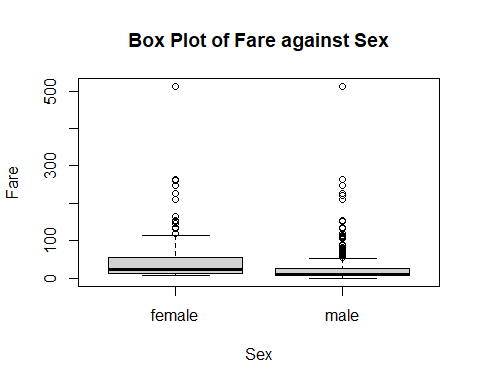
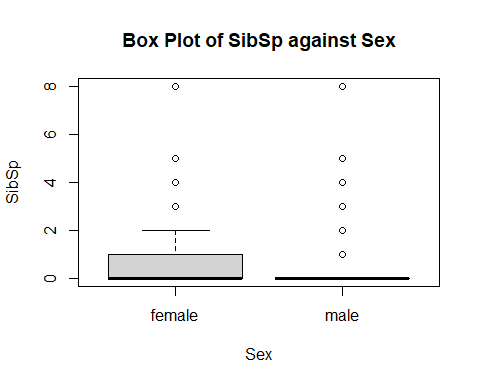
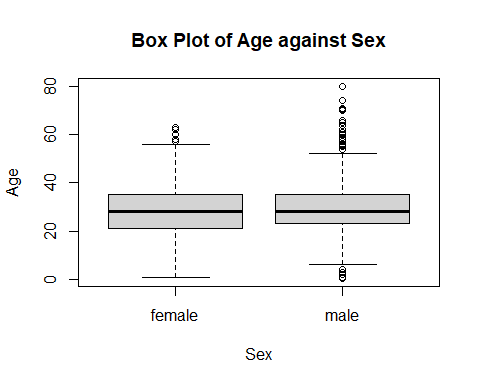
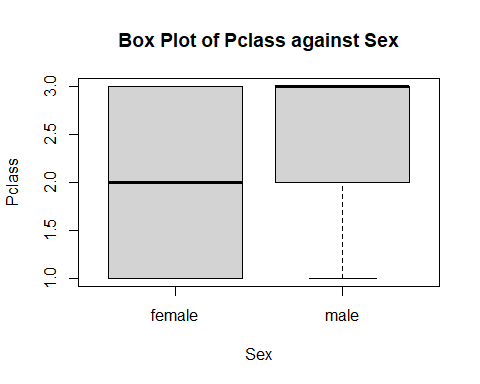
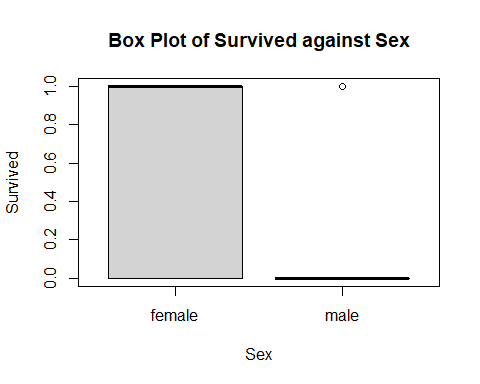
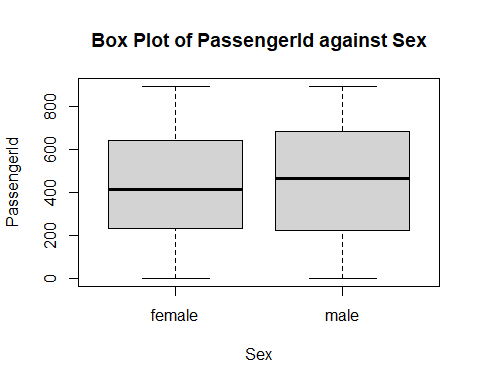


Boxplot and Histogram of numeric Variables

for(col in names(numeric)){  
 boxplot(td[col], col="green", main = paste("Box plot of ", col))  
}



numeric\_col <- names(td[sapply(td, is.numeric)])  
cols <- numeric\_col  
for (i in cols){  
 boxplot(td[[i]] ~td$Sex, main = paste("Box Plot of", i, "against Sex"), xlab = "Sex", ylab = i)  
}



factor\_cold <- td[, sapply(td, is.factor)]  
y <- factor\_cold  
lapply(y, function(x) {  
 table(td$Survived,x)  
})

## $Sex  
## x  
## female male  
## 0 81 468  
## 1 233 109  
##   
## $Embarked  
## x  
## C Q S  
## 0 75 47 427  
## 1 93 30 219

Divide the data into Train and test using 75 to 25 percent

set.seed(123)  
train\_sample <- sample(nrow(td), 0.75\*nrow(td))  
train\_td <- td[train\_sample,]  
test\_td <- td[-train\_sample,]  
  
str(train\_td)

## 'data.frame': 668 obs. of 8 variables:  
## $ PassengerId: int 415 463 179 526 195 818 118 299 229 244 ...  
## $ Survived : int 1 0 0 0 1 0 0 1 0 0 ...  
## $ Pclass : int 3 1 2 3 1 2 2 1 2 3 ...  
## $ Sex : Factor w/ 2 levels "female","male": 2 2 2 2 1 2 2 2 2 2 ...  
## $ Age : num 44 47 30 40.5 44 31 29 28 18 22 ...  
## $ SibSp : int 0 0 0 0 0 1 1 0 0 0 ...  
## $ Fare : num 7.92 38.5 13 7.75 27.72 ...  
## $ Embarked : Factor w/ 3 levels "C","Q","S": 3 3 3 2 1 1 3 3 3 3 ...

Divide the data into Train and test using 75 to 25 percent

logr\_model <- glm(Survived~., data = train\_td, family = "binomial")  
summary(logr\_model)

##   
## Call:  
## glm(formula = Survived ~ ., family = "binomial", data = train\_td)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 5.596e+00 6.853e-01 8.165 3.22e-16 \*\*\*  
## PassengerId -1.101e-05 4.087e-04 -0.027 0.97851   
## Pclass -1.117e+00 1.643e-01 -6.795 1.08e-11 \*\*\*  
## Sexmale -2.811e+00 2.303e-01 -12.207 < 2e-16 \*\*\*  
## Age -4.594e-02 9.456e-03 -4.858 1.19e-06 \*\*\*  
## SibSp -3.648e-01 1.258e-01 -2.898 0.00375 \*\*   
## Fare 1.949e-03 2.569e-03 0.758 0.44824   
## EmbarkedQ 1.974e-01 4.318e-01 0.457 0.64755   
## EmbarkedS -5.486e-01 2.723e-01 -2.015 0.04390 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 889.27 on 667 degrees of freedom  
## Residual deviance: 575.91 on 659 degrees of freedom  
## AIC: 593.91  
##   
## Number of Fisher Scoring iterations: 5

Survived\_Lr <- predict(logr\_model, test\_td, type = "response")  
Pred\_Survived\_Lr <- ifelse(Survived\_Lr< 0.5,0,1)  
library(Metrics)  
  
Accuracy\_Lr<- accuracy(test\_td$Survived,Pred\_Survived\_Lr)  
Accuracy\_Lr

## [1] 0.7578475

library(rpart)  
tree\_model <- rpart(Survived ~ ., data = train\_td, method = "class")  
summary(tree\_model)

## Call:  
## rpart(formula = Survived ~ ., data = train\_td, method = "class")  
## n= 668   
##   
## CP nsplit rel error xerror xstd  
## 1 0.46093750 0 1.0000000 1.0000000 0.04908405  
## 2 0.03320312 1 0.5390625 0.5390625 0.04087420  
## 3 0.02734375 3 0.4726562 0.5078125 0.03997004  
## 4 0.01953125 4 0.4453125 0.4804688 0.03913115  
## 5 0.01562500 5 0.4257812 0.4804688 0.03913115  
## 6 0.01000000 7 0.3945312 0.4687500 0.03875723  
##   
## Variable importance  
## Sex Fare Age Pclass SibSp Embarked   
## 49 18 12 11 4 3   
## PassengerId   
## 3   
##   
## Node number 1: 668 observations, complexity param=0.4609375  
## predicted class=0 expected loss=0.3832335 P(node) =1  
## class counts: 412 256  
## probabilities: 0.617 0.383   
## left son=2 (436 obs) right son=3 (232 obs)  
## Primary splits:  
## Sex splits as RL, improve=97.889380, (0 missing)  
## Pclass < 2.5 to the right, improve=30.045850, (0 missing)  
## Fare < 10.48125 to the left, improve=24.587560, (0 missing)  
## Embarked splits as RLL, improve=11.254240, (0 missing)  
## Age < 5.5 to the right, improve= 7.567057, (0 missing)  
## Surrogate splits:  
## Fare < 77.6229 to the left, agree=0.678, adj=0.073, (0 split)  
## Age < 15.5 to the right, agree=0.656, adj=0.009, (0 split)  
## PassengerId < 29.5 to the right, agree=0.654, adj=0.004, (0 split)  
##   
## Node number 2: 436 observations, complexity param=0.02734375  
## predicted class=0 expected loss=0.1857798 P(node) =0.6526946  
## class counts: 355 81  
## probabilities: 0.814 0.186   
## left son=4 (417 obs) right son=5 (19 obs)  
## Primary splits:  
## Age < 6.5 to the right, improve=9.870601, (0 missing)  
## Pclass < 1.5 to the right, improve=6.981121, (0 missing)  
## Fare < 26.26875 to the left, improve=6.840184, (0 missing)  
## Embarked splits as RLL, improve=4.121528, (0 missing)  
## PassengerId < 182 to the left, improve=1.602857, (0 missing)  
##   
## Node number 3: 232 observations, complexity param=0.03320312  
## predicted class=1 expected loss=0.2456897 P(node) =0.3473054  
## class counts: 57 175  
## probabilities: 0.246 0.754   
## left son=6 (107 obs) right son=7 (125 obs)  
## Primary splits:  
## Pclass < 2.5 to the right, improve=21.184200, (0 missing)  
## Fare < 48.2 to the left, improve= 7.917180, (0 missing)  
## SibSp < 2.5 to the right, improve= 5.684160, (0 missing)  
## Embarked splits as RRL, improve= 2.930635, (0 missing)  
## Age < 12 to the left, improve= 2.120384, (0 missing)  
## Surrogate splits:  
## Fare < 25.69795 to the left, agree=0.776, adj=0.514, (0 split)  
## Age < 28.5 to the left, agree=0.681, adj=0.308, (0 split)  
## Embarked splits as RLR, agree=0.629, adj=0.196, (0 split)  
## PassengerId < 257 to the left, agree=0.608, adj=0.150, (0 split)  
## SibSp < 1.5 to the right, agree=0.586, adj=0.103, (0 split)  
##   
## Node number 4: 417 observations  
## predicted class=0 expected loss=0.1630695 P(node) =0.6242515  
## class counts: 349 68  
## probabilities: 0.837 0.163   
##   
## Node number 5: 19 observations  
## predicted class=1 expected loss=0.3157895 P(node) =0.02844311  
## class counts: 6 13  
## probabilities: 0.316 0.684   
##   
## Node number 6: 107 observations, complexity param=0.03320312  
## predicted class=1 expected loss=0.4766355 P(node) =0.1601796  
## class counts: 51 56  
## probabilities: 0.477 0.523   
## left son=12 (23 obs) right son=13 (84 obs)  
## Primary splits:  
## Fare < 23.7 to the right, improve=9.046739, (0 missing)  
## Embarked splits as RRL, improve=7.011085, (0 missing)  
## Age < 38.5 to the right, improve=5.383178, (0 missing)  
## PassengerId < 396 to the right, improve=3.858265, (0 missing)  
## SibSp < 2.5 to the right, improve=3.439318, (0 missing)  
## Surrogate splits:  
## SibSp < 2.5 to the right, agree=0.879, adj=0.435, (0 split)  
## Age < 37.5 to the right, agree=0.822, adj=0.174, (0 split)  
##   
## Node number 7: 125 observations  
## predicted class=1 expected loss=0.048 P(node) =0.1871257  
## class counts: 6 119  
## probabilities: 0.048 0.952   
##   
## Node number 12: 23 observations  
## predicted class=0 expected loss=0.1304348 P(node) =0.03443114  
## class counts: 20 3  
## probabilities: 0.870 0.130   
##   
## Node number 13: 84 observations, complexity param=0.01953125  
## predicted class=1 expected loss=0.3690476 P(node) =0.1257485  
## class counts: 31 53  
## probabilities: 0.369 0.631   
## left son=26 (15 obs) right son=27 (69 obs)  
## Primary splits:  
## Age < 28.5 to the right, improve=3.2349900, (0 missing)  
## Embarked splits as LRL, improve=3.2275520, (0 missing)  
## Fare < 8.0396 to the right, improve=3.0414010, (0 missing)  
## PassengerId < 396 to the right, improve=1.3412700, (0 missing)  
## SibSp < 0.5 to the right, improve=0.1773548, (0 missing)  
##   
## Node number 26: 15 observations  
## predicted class=0 expected loss=0.3333333 P(node) =0.02245509  
## class counts: 10 5  
## probabilities: 0.667 0.333   
##   
## Node number 27: 69 observations, complexity param=0.015625  
## predicted class=1 expected loss=0.3043478 P(node) =0.1032934  
## class counts: 21 48  
## probabilities: 0.304 0.696   
## left son=54 (38 obs) right son=55 (31 obs)  
## Primary splits:  
## Fare < 8.0396 to the right, improve=3.4601760, (0 missing)  
## Embarked splits as LRL, improve=2.6400100, (0 missing)  
## PassengerId < 399 to the right, improve=1.1984260, (0 missing)  
## Age < 6.5 to the right, improve=0.5821454, (0 missing)  
## SibSp < 0.5 to the right, improve=0.3673913, (0 missing)  
## Surrogate splits:  
## SibSp < 0.5 to the right, agree=0.710, adj=0.355, (0 split)  
## Embarked splits as LRL, agree=0.710, adj=0.355, (0 split)  
## PassengerId < 97.5 to the right, agree=0.623, adj=0.161, (0 split)  
## Age < 14.75 to the left, agree=0.594, adj=0.097, (0 split)  
##   
## Node number 54: 38 observations, complexity param=0.015625  
## predicted class=1 expected loss=0.4473684 P(node) =0.05688623  
## class counts: 17 21  
## probabilities: 0.447 0.553   
## left son=108 (22 obs) right son=109 (16 obs)  
## Primary splits:  
## Fare < 15.3729 to the left, improve=5.7440190, (0 missing)  
## Age < 6.5 to the right, improve=2.1061400, (0 missing)  
## PassengerId < 399 to the right, improve=0.4736842, (0 missing)  
## SibSp < 0.5 to the left, improve=0.4141235, (0 missing)  
## Embarked splits as LRR, improve=0.1228070, (0 missing)  
## Surrogate splits:  
## SibSp < 1.5 to the left, agree=0.684, adj=0.250, (0 split)  
## PassengerId < 399 to the right, agree=0.658, adj=0.188, (0 split)  
## Embarked splits as LRL, agree=0.658, adj=0.188, (0 split)  
## Age < 0.875 to the right, agree=0.632, adj=0.125, (0 split)  
##   
## Node number 55: 31 observations  
## predicted class=1 expected loss=0.1290323 P(node) =0.04640719  
## class counts: 4 27  
## probabilities: 0.129 0.871   
##   
## Node number 108: 22 observations  
## predicted class=0 expected loss=0.3181818 P(node) =0.03293413  
## class counts: 15 7  
## probabilities: 0.682 0.318   
##   
## Node number 109: 16 observations  
## predicted class=1 expected loss=0.125 P(node) =0.0239521  
## class counts: 2 14  
## probabilities: 0.125 0.875

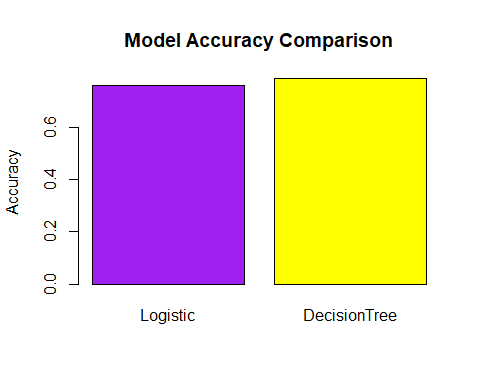
library(rpart)  
tree\_model <- rpart(Survived ~ ., data = train\_td, method = "class")  
summary(tree\_model)

## Call:  
## rpart(formula = Survived ~ ., data = train\_td, method = "class")  
## n= 668   
##   
## CP nsplit rel error xerror xstd  
## 1 0.46093750 0 1.0000000 1.0000000 0.04908405  
## 2 0.03320312 1 0.5390625 0.5390625 0.04087420  
## 3 0.02734375 3 0.4726562 0.5429688 0.04098331  
## 4 0.01953125 4 0.4453125 0.5000000 0.03973504  
## 5 0.01562500 5 0.4257812 0.5078125 0.03997004  
## 6 0.01000000 7 0.3945312 0.5000000 0.03973504  
##   
## Variable importance  
## Sex Fare Age Pclass SibSp Embarked   
## 49 18 12 11 4 3   
## PassengerId   
## 3   
##   
## Node number 1: 668 observations, complexity param=0.4609375  
## predicted class=0 expected loss=0.3832335 P(node) =1  
## class counts: 412 256  
## probabilities: 0.617 0.383   
## left son=2 (436 obs) right son=3 (232 obs)  
## Primary splits:  
## Sex splits as RL, improve=97.889380, (0 missing)  
## Pclass < 2.5 to the right, improve=30.045850, (0 missing)  
## Fare < 10.48125 to the left, improve=24.587560, (0 missing)  
## Embarked splits as RLL, improve=11.254240, (0 missing)  
## Age < 5.5 to the right, improve= 7.567057, (0 missing)  
## Surrogate splits:  
## Fare < 77.6229 to the left, agree=0.678, adj=0.073, (0 split)  
## Age < 15.5 to the right, agree=0.656, adj=0.009, (0 split)  
## PassengerId < 29.5 to the right, agree=0.654, adj=0.004, (0 split)  
##   
## Node number 2: 436 observations, complexity param=0.02734375  
## predicted class=0 expected loss=0.1857798 P(node) =0.6526946  
## class counts: 355 81  
## probabilities: 0.814 0.186   
## left son=4 (417 obs) right son=5 (19 obs)  
## Primary splits:  
## Age < 6.5 to the right, improve=9.870601, (0 missing)  
## Pclass < 1.5 to the right, improve=6.981121, (0 missing)  
## Fare < 26.26875 to the left, improve=6.840184, (0 missing)  
## Embarked splits as RLL, improve=4.121528, (0 missing)  
## PassengerId < 182 to the left, improve=1.602857, (0 missing)  
##   
## Node number 3: 232 observations, complexity param=0.03320312  
## predicted class=1 expected loss=0.2456897 P(node) =0.3473054  
## class counts: 57 175  
## probabilities: 0.246 0.754   
## left son=6 (107 obs) right son=7 (125 obs)  
## Primary splits:  
## Pclass < 2.5 to the right, improve=21.184200, (0 missing)  
## Fare < 48.2 to the left, improve= 7.917180, (0 missing)  
## SibSp < 2.5 to the right, improve= 5.684160, (0 missing)  
## Embarked splits as RRL, improve= 2.930635, (0 missing)  
## Age < 12 to the left, improve= 2.120384, (0 missing)  
## Surrogate splits:  
## Fare < 25.69795 to the left, agree=0.776, adj=0.514, (0 split)  
## Age < 28.5 to the left, agree=0.681, adj=0.308, (0 split)  
## Embarked splits as RLR, agree=0.629, adj=0.196, (0 split)  
## PassengerId < 257 to the left, agree=0.608, adj=0.150, (0 split)  
## SibSp < 1.5 to the right, agree=0.586, adj=0.103, (0 split)  
##   
## Node number 4: 417 observations  
## predicted class=0 expected loss=0.1630695 P(node) =0.6242515  
## class counts: 349 68  
## probabilities: 0.837 0.163   
##   
## Node number 5: 19 observations  
## predicted class=1 expected loss=0.3157895 P(node) =0.02844311  
## class counts: 6 13  
## probabilities: 0.316 0.684   
##   
## Node number 6: 107 observations, complexity param=0.03320312  
## predicted class=1 expected loss=0.4766355 P(node) =0.1601796  
## class counts: 51 56  
## probabilities: 0.477 0.523   
## left son=12 (23 obs) right son=13 (84 obs)  
## Primary splits:  
## Fare < 23.7 to the right, improve=9.046739, (0 missing)  
## Embarked splits as RRL, improve=7.011085, (0 missing)  
## Age < 38.5 to the right, improve=5.383178, (0 missing)  
## PassengerId < 396 to the right, improve=3.858265, (0 missing)  
## SibSp < 2.5 to the right, improve=3.439318, (0 missing)  
## Surrogate splits:  
## SibSp < 2.5 to the right, agree=0.879, adj=0.435, (0 split)  
## Age < 37.5 to the right, agree=0.822, adj=0.174, (0 split)  
##   
## Node number 7: 125 observations  
## predicted class=1 expected loss=0.048 P(node) =0.1871257  
## class counts: 6 119  
## probabilities: 0.048 0.952   
##   
## Node number 12: 23 observations  
## predicted class=0 expected loss=0.1304348 P(node) =0.03443114  
## class counts: 20 3  
## probabilities: 0.870 0.130   
##   
## Node number 13: 84 observations, complexity param=0.01953125  
## predicted class=1 expected loss=0.3690476 P(node) =0.1257485  
## class counts: 31 53  
## probabilities: 0.369 0.631   
## left son=26 (15 obs) right son=27 (69 obs)  
## Primary splits:  
## Age < 28.5 to the right, improve=3.2349900, (0 missing)  
## Embarked splits as LRL, improve=3.2275520, (0 missing)  
## Fare < 8.0396 to the right, improve=3.0414010, (0 missing)  
## PassengerId < 396 to the right, improve=1.3412700, (0 missing)  
## SibSp < 0.5 to the right, improve=0.1773548, (0 missing)  
##   
## Node number 26: 15 observations  
## predicted class=0 expected loss=0.3333333 P(node) =0.02245509  
## class counts: 10 5  
## probabilities: 0.667 0.333   
##   
## Node number 27: 69 observations, complexity param=0.015625  
## predicted class=1 expected loss=0.3043478 P(node) =0.1032934  
## class counts: 21 48  
## probabilities: 0.304 0.696   
## left son=54 (38 obs) right son=55 (31 obs)  
## Primary splits:  
## Fare < 8.0396 to the right, improve=3.4601760, (0 missing)  
## Embarked splits as LRL, improve=2.6400100, (0 missing)  
## PassengerId < 399 to the right, improve=1.1984260, (0 missing)  
## Age < 6.5 to the right, improve=0.5821454, (0 missing)  
## SibSp < 0.5 to the right, improve=0.3673913, (0 missing)  
## Surrogate splits:  
## SibSp < 0.5 to the right, agree=0.710, adj=0.355, (0 split)  
## Embarked splits as LRL, agree=0.710, adj=0.355, (0 split)  
## PassengerId < 97.5 to the right, agree=0.623, adj=0.161, (0 split)  
## Age < 14.75 to the left, agree=0.594, adj=0.097, (0 split)  
##   
## Node number 54: 38 observations, complexity param=0.015625  
## predicted class=1 expected loss=0.4473684 P(node) =0.05688623  
## class counts: 17 21  
## probabilities: 0.447 0.553   
## left son=108 (22 obs) right son=109 (16 obs)  
## Primary splits:  
## Fare < 15.3729 to the left, improve=5.7440190, (0 missing)  
## Age < 6.5 to the right, improve=2.1061400, (0 missing)  
## PassengerId < 399 to the right, improve=0.4736842, (0 missing)  
## SibSp < 0.5 to the left, improve=0.4141235, (0 missing)  
## Embarked splits as LRR, improve=0.1228070, (0 missing)  
## Surrogate splits:  
## SibSp < 1.5 to the left, agree=0.684, adj=0.250, (0 split)  
## PassengerId < 399 to the right, agree=0.658, adj=0.188, (0 split)  
## Embarked splits as LRL, agree=0.658, adj=0.188, (0 split)  
## Age < 0.875 to the right, agree=0.632, adj=0.125, (0 split)  
##   
## Node number 55: 31 observations  
## predicted class=1 expected loss=0.1290323 P(node) =0.04640719  
## class counts: 4 27  
## probabilities: 0.129 0.871   
##   
## Node number 108: 22 observations  
## predicted class=0 expected loss=0.3181818 P(node) =0.03293413  
## class counts: 15 7  
## probabilities: 0.682 0.318   
##   
## Node number 109: 16 observations  
## predicted class=1 expected loss=0.125 P(node) =0.0239521  
## class counts: 2 14  
## probabilities: 0.125 0.875

Survived\_tree <- predict(tree\_model, test\_td, type = "class")  
Survived\_tree\_probs <- predict(tree\_model, test\_td, type = "prob")  
library(Metrics)  
Accuracy\_tree <- accuracy(test\_td$Survived, Survived\_tree)  
print(Accuracy\_tree)

## [1] 0.7847534

barplot(  
 c(Logistic = Accuracy\_Lr,DecisionTree = Accuracy\_tree),  
 col = c("purple", "yellow", "green"),  
 main = "Model Accuracy Comparison",  
 ylab = "Accuracy"  
)



if(Accuracy\_Lr > Accuracy\_tree) cat("The best model is Logistic Regression with accuracy of", round(Accuracy\_Lr,2)) else cat("The best model is Regression Tree with rmse of",Accuracy\_tree)

## The best model is Regression Tree with rmse of 0.7847534