Group 9 - Data-Driven Optimisation of Supply Chain Management

Model Comparison

The analysis Logistic Regression and Decision Tree, to compare their performance in terms of accuracy, precision, recall, F1 score, and AUC.

```r
# Load necessary libraries
library(readxl)
library(dplyr)
library(lubridate)
library(ggplot2)
library(reshape2)
library(corrplot)
library(reshape2)
library(rpart)
library(caret)
library(lattice)
library(randomForest)
library(forecast)
library(pROC)
library(tidyr)
library(rpart.plot)


#data
data <- read_excel("D:/data mining data/incom2024_delay_example_dataset1.x
lsx",
                   sheet = 'incom2024_delay_example_dataset')

#Data processing
data<- na.omit(data)
data <- unique(data)
data<- data[data$customer_state != '91732', ]

data <- data %>%
  mutate(label = ifelse(label %in% c(-1, 0), 0, label)) %>%
  mutate(label = case_when(
    label == 0 ~ "Not Delayed",
    label == 1 ~ "Delayed",
    TRUE ~ as.character(label)
  ))
data$label <- as.factor(data$label)

#Data processing
data <- data %>%mutate(across(where(is.character), as.factor))
data <- data %>% filter(customer_state != "OK")

# Split the data into training 、 testing sets and validation sets
set.seed(42)
train_index <- createDataPartition(data$label, p = 0.4, list = FALSE)
train_data <- data[train_index, ]
temp_data <- data[-train_index, ]
```

```r
valid_index <- createDataPartition(temp_data$label, p = 0.5, list = FALSE)
valid_data <- temp_data[valid_index, ]
test_data <- temp_data[-valid_index, ]

#Logistic regression model
logistic_model <- glm(label ~ shipping_mode + order_region + category_name
                                  + order_item_total_amount+ customer_sta
te
                                  + customer_segment + department_name
                                  + payment_type,
                      data = train_data, family = "binomial")
print(summary(logistic_model))

##
## Call:
## glm(formula = label ~ shipping_mode + order_region + category_name +
##     order_item_total_amount + customer_state + customer_segment +
##     department_name + payment_type, family = "binomial", data = train_d
ata)
##
## Coefficients:
##                                       Estimate Std. Error z value Pr(>|z
|)
## (Intercept)                      1.008e+01  8.827e+02   0.011   0.9909
## shipping_modeSame Day            4.484e+00  3.151e-01  14.230   <2e-16 ***
## shipping_modeSecond Class        3.178e+00  2.987e-01  10.639   <2e-16 ***
## shipping_modeStandard Class      4.793e+00  2.935e-01  16.328   <2e-16 ***
## order_regionCaribbean              -1.630e-01  4.950e-01  -0.329   0.7419

## order_regionCentral Africa         -1.140e-01  5.718e-01  -0.199   0.8420
## order_regionCentral America        -1.602e-01  4.812e-01  -0.333   0.7392
## order_regionCentral Asia            3.638e-01  1.323e+00   0.275   0.7833
## order_regionEast Africa             1.039e-01  5.321e-01   0.195   0.8452
## order_regionEast of USA            -1.419e-01  4.975e-01  -0.285   0.7754
## order_regionEastern Asia           -1.156e-01  4.966e-01  -0.233   0.8159
## order_regionEastern Europe          1.406e-01  5.233e-01   0.269   0.7881
## order_regionNorth Africa           -5.857e-01  5.285e-01  -1.108   0.2678
## order_regionNorthern Europe        -2.102e-02  4.903e-01  -0.043   0.9658
## order_regionOceania                 5.131e-02  4.907e-01   0.105   0.9167
## order_regionSouth America           2.887e-02  4.877e-01   0.059   0.952
## order_regionSouth Asia              6.932e-02  4.999e-01   0.139   0.8897
## order_regionSouth of  USA           1.072e-01  5.196e-01   0.206   0.8365
## order_regionSoutheast Asia         -1.046e-01  4.922e-01  -0.212   0.8317
## order_regionSouthern Africa        -2.441e-01  6.545e-01  -0.373   0.7092
## order_regionSouthern Europe         1.943e-02  4.917e-01   0.040   0.9685
## order_regionUS Center              -1.132e-01  5.086e-01  -0.223   0.8239
## order_regionWest Africa             1.176e-01  5.220e-01   0.225   0.8217
## order_regionWest Asia               2.889e-01  5.004e-01   0.577   0.5637
## order_regionWest of USA            -1.505e-01  4.982e-01  -0.302   0.7626
## order_regionWestern Europe         -1.623e-02  4.812e-01  -0.034   0.9731
## category_nameAs Seen on  TV!       1.245e+00  2.769e+00   0.450   0.6529
## category_nameBaby                  -1.207e-01  8.383e-01  -0.144   0.8855
## category_nameBaseball & Softball    8.463e-02  9.207e-01   0.092   0.9268
## category_nameBasketball             2.841e-01  1.498e+00   0.190   0.8496
```

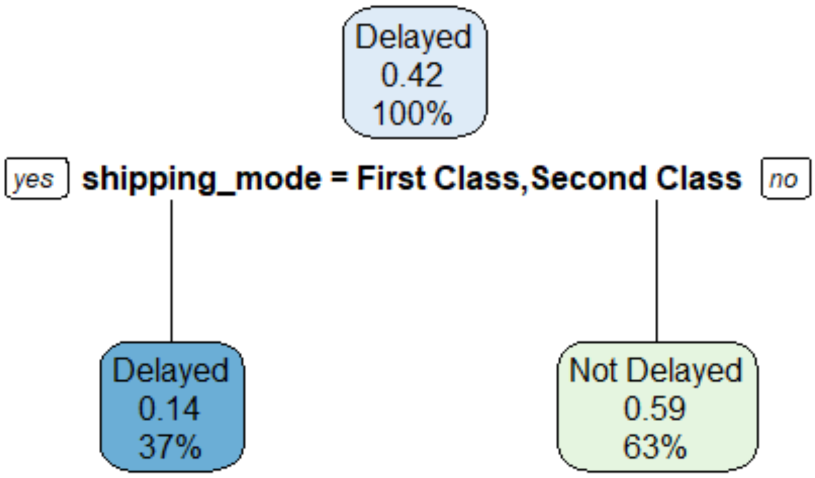
```
## category_nameBooks                -1.373e+01  4.839e+02  -0.028   0.9774
## category_nameBoxing & MMA         -1.016e+00  9.180e-01  -1.107   0.2683
## category_nameCameras               7.573e-02  8.928e-01   0.085   0.9324

## category_nameCamping & Hiking      6.549e-01  7.556e-01   0.867   0.3861
## category_nameCardio Equipment-     9.559e-01  5.718e-01  -1.672   0.0946
## category_nameCDs                  -2.846e-01  1.133e+00  -0.251   0.8017
## category_nameChildren's Clothing -1.383e-01  8.068e-01  -0.171   0.8639
## category_nameCleats               -1.603e-01  6.225e-01  -0.258   0.7968
## category_nameComputers             3.953e-01  1.158e+00   0.341   0.7328
## category_nameConsumer Electronics -4.864e-01  1.159e+00  -0.420   0.674
## category_nameCrafts               -6.450e-01  5.441e-01  -1.185   0.2358
## category_nameDVDs                  1.085e+00  1.085e+00   1.000   0.3172
## category_nameElectronics          -5.969e-01  4.557e-01  -1.310   0.1903
## category_nameFishing               5.402e-01  7.614e-01   0.709   0.4780
## category_nameFitness Accessories -7.674e-01  8.978e-01  -0.855   0.3927
## category_nameGarden               -1.007e-01  8.514e-01  -0.118   0.9059
## category_nameGirls' Apparel       -6.610e-01  9.370e-01  -0.705   0.4806
## category_nameGolf Apparel         -8.053e-01  1.041e+00  -0.774   0.4390
## category_nameGolf Balls           -1.029e+00  4.823e-01  -2.133   0.0330
## category_nameGolf Gloves          -3.627e-01  5.031e-01  -0.721   0.4709
## category_nameGolf Shoes           -1.731e+00  6.918e-01  -2.503   0.0123
## category_nameHealth and Beauty    -1.848e-01  1.763e+00  -0.105   0.9166
## category_nameHockey                3.046e-01  9.720e-01   0.313   0.7540
## category_nameHunting & Shooting    4.755e-01  9.189e-01   0.517   0.6048
## category_nameIndoor/Outdoor Games 5.616e-01  7.559e-01   0.743   0.4575
## category_nameKids' Golf Clubs     -1.078e+01  6.242e+02  -0.017   0.986
## category_nameLacrosse             -4.908e-01  9.782e-01  -0.502   0.6159
## category_nameMen's Clothing       -2.439e-01  1.170e+00  -0.208   0.8349
## category_nameMen's Footwear       -1.469e-01  6.201e-01  -0.237   0.8127
## category_nameMen's Golf Clubs     -1.440e+01  2.792e+02  -0.052   0.9589
## category_nameMusic                 1.327e-01  8.239e-01   0.161   0.8720
## category_namePet Supplies         -4.797e-01  9.516e-01  -0.504   0.6142
## category_nameShop By Sport        -2.401e-01  8.516e-01  -0.282   0.7780
## category_nameSoccer               -2.021e+00  8.974e-01  -2.252   0.0243
## category_nameSporting Goods        3.815e-01  1.296e+00   0.294   0.7684
## category_nameStrength Training.   -4.074e-01  1.424e+00  -0.286   0.7748
## category_nameTennis & Racquet     -8.551e-01  1.046e+00  -0.818   0.4135
## category_nameToys                  1.260e-01  9.270e-01   0.136   0.8918
## category_nameTrade-In             -4.739e-01  6.045e-01  -0.784   0.4331
## category_nameVideo Games           1.500e-01  1.027e+00   0.146   0.8838
## category_nameWater Sports          5.587e-01  7.566e-01   0.738   0.4602
## category_nameWomen's Apparel      -2.010e-01  8.497e-01  -0.237   0.8130
## category_nameWomen's Clothing      8.224e-02  7.623e-01   0.108   0.9141
## category_nameWomen's Golf Clubs    9.944e-01  1.328e+00   0.749   0.4540
## order_item_total_amount           -3.769e-04  4.320e-04  -0.873   0.3829
## customer_stateAZ                  -1.382e+01  8.827e+02  -0.016   0.9875
## customer_stateCA                  -1.433e+01  8.827e+02  -0.016   0.9871
## customer_stateCO                  -1.436e+01  8.827e+02  -0.016   0.9870
## customer_stateCT                  -1.508e+01  8.827e+02  -0.017   0.9864
## customer_stateDC                  -1.445e+01  8.827e+02  -0.016   0.9869
## customer_stateDE                  -1.382e+01  8.827e+02  -0.016   0.9875
## customer_stateFL                  -1.431e+01  8.827e+02  -0.016   0.9871
## customer_stateGA                  -1.442e+01  8.827e+02  -0.016   0.9870
```

```
## customer_stateHI                 -1.463e+01  8.827e+02  -0.017   0.9868
## customer_stateIA                  1.274e-01  1.248e+03   0.000   0.9999
## customer_stateID                 -1.428e+01  8.827e+02  -0.016   0.9871

## customer_stateIL                 -1.409e+01  8.827e+02  -0.016   0.9873
## customer_stateIN                 -1.439e+01  8.827e+02  -0.016   0.9870
## customer_stateKS                 -1.560e+01  8.827e+02  -0.018   0.9859
## customer_stateKY                 -1.438e+01  8.827e+02  -0.016   0.9870
## customer_stateLA                 -1.341e+01  8.827e+02  -0.015   0.9879
## customer_stateMA                 -1.357e+01  8.827e+02  -0.015   0.9877
## customer_stateMD                 -1.419e+01  8.827e+02  -0.016   0.9872
## customer_stateMI                 -1.438e+01  8.827e+02  -0.016   0.9870
## customer_stateMN                 -1.432e+01  8.827e+02  -0.016   0.9871
## customer_stateMO                 -1.466e+01  8.827e+02  -0.017   0.9867
## customer_stateMT                 -1.431e+01  8.827e+02  -0.016   0.9871
## customer_stateNC                 -1.444e+01  8.827e+02  -0.016   0.9869
## customer_stateND                 -1.298e+01  8.827e+02  -0.015   0.9883
## customer_stateNJ                 -1.443e+01  8.827e+02  -0.016   0.9870
## customer_stateNM                 -1.482e+01  8.827e+02  -0.017   0.9866
## customer_stateNV                 -1.431e+01  8.827e+02  -0.016   0.9871
## customer_stateNY                 -1.412e+01  8.827e+02  -0.016   0.9872
## customer_stateOH                 -1.425e+01  8.827e+02  -0.016   0.9871
## customer_stateOR                 -1.350e+01  8.827e+02  -0.015   0.9878
## customer_statePA                 -1.411e+01  8.827e+02  -0.016   0.9872
## customer_statePR                 -1.418e+01  8.827e+02  -0.016   0.9872
## customer_stateRI                 -1.454e+01  8.827e+02  -0.016   0.9869
## customer_stateSC                 -1.437e+01  8.827e+02  -0.016   0.9870
## customer_stateTN                 -1.427e+01  8.827e+02  -0.016   0.9871
## customer_stateTX                 -1.434e+01  8.827e+02  -0.016   0.9870
## customer_stateUT                 -1.338e+01  8.827e+02  -0.015   0.9879
## customer_stateVA                 -1.454e+01  8.827e+02  -0.016   0.9869
## customer_stateWA                 -1.379e+01  8.827e+02  -0.016   0.9875
## customer_stateWI                 -1.388e+01  8.827e+02  -0.016   0.9875
## customer_stateWV                 -1.489e+01  8.827e+02  -0.017   0.9865
## customer_segmentCorporate         2.643e-02  6.755e-02   0.391   0.6956
## customer_segmentHome Office       2.073e-02  8.350e-02   0.248   0.8039
## department_nameBook Shop          1.341e+01  4.839e+02   0.028   0.9779
## department_nameDiscs Shop         6.970e-03  8.985e-01   0.008   0.9938
## department_nameFan Shop          -6.456e-01  6.646e-01  -0.971   0.3313
## department_nameFitness           -2.146e-01  8.647e-01  -0.248   0.8040
## department_nameFootwear           7.507e-01  6.054e-01   1.240   0.2150
## department_nameGolf               1.140e-02  8.561e-01   0.013   0.9894
## department_nameHealth and Beauty -3.364e-02  1.776e+00  -0.019   0.9849
## department_nameOutdoors           3.413e-01  5.191e-01   0.658   0.5108
## department_namePet Shop           5.359e-01  1.179e+00   0.455   0.6494
## department_nameTechnology        -1.821e-01  1.032e+00  -0.177   0.8599
## payment_typeDEBIT                 1.810e-02  1.003e-01   0.180   0.8568
## payment_typePAYMENT              -2.319e-02  1.065e-01  -0.218   0.8276
## payment_typeTRANSFER             -4.988e-02  1.065e-01  -0.468   0.6396

## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 8472.3  on 6218  degrees of freedom
## Residual deviance: 6781.6  on 6088  degrees of freedom
## AIC: 7043.6
##
## Number of Fisher Scoring iterations: 13
```

```
logistic_predictions <- predict(logistic_model, test_data, type = "respons
e")
logistic_predictions <- ifelse(logistic_predictions > 0.5, "Delayed", "Not
 Delayed")
conf_matrix <- confusionMatrix(as.factor(logistic_predictions), as.factor
(test_data$label))

# valid
valid_predictions <- predict(logistic_model, newdata = valid_data, type =
"response")
valid_pred_labels <- ifelse(valid_predictions > 0.5, "Delayed", "Not Delay
ed")

# test
test_predictions <- predict(logistic_model, newdata = test_data, type = "r
esponse")
test_pred_labels <- ifelse(test_predictions > 0.5, "Delayed", "Not Delayed
")

# caculate valid metrics
valid_conf_matrix <- confusionMatrix(as.factor(valid_pred_labels), as.fact
or(valid_data$label))
valid_accuracy <- valid_conf_matrix$overall["Accuracy"]
valid_precision <- valid_conf_matrix$byClass["Precision"]
valid_recall <- valid_conf_matrix$byClass["Recall"]
valid_F1 <- valid_conf_matrix$byClass["F1"]
valid_auc <- auc(valid_data$label, valid_predictions)

# print valid results
cat("Accuracy: ", valid_accuracy, "\n")
## Accuracy:  0.3262594
cat("Precision: ", valid_precision, "\n")
## Precision:  0.4183007
cat("Recall: ", valid_recall, "\n")
## Recall:  0.4277757
cat("F1 Score: ", valid_F1, "\n")
## F1 Score:  0.4229851
cat("AUC: ", valid_auc, "\n")
## AUC:  0.7255645

# caculate test metrics
test_conf_matrix <- confusionMatrix(as.factor(test_pred_labels), as.factor
(test_data$label))
test_accuracy <- test_conf_matrix$overall["Accuracy"]
test_precision <- test_conf_matrix$byClass["Precision"]
test_recall <- test_conf_matrix$byClass["Recall"]
```

```
test_F1 <- test_conf_matrix$byClass["F1"]
test_auc <- auc(test_data$label, test_predictions)

## Setting levels: control = Delayed, case = Not Delayed
## Setting direction: controls < cases

# print test results
cat("Accuracy: ", test_accuracy, "\n")
## Accuracy:  0.3158911
cat("Precision: ", test_precision, "\n")
## Precision:  0.4097172
cat("Recall: ", test_recall, "\n")
## Recall:  0.4197623
cat("F1 Score: ", test_F1, "\n")
## F1 Score:  0.4146789
cat("AUC: ", test_auc, "\n")
## AUC:  0.733879

# decision tree model
decision_tree_model <- rpart(label ~ shipping_mode + order_region + catego
ry_name
                                + order_item_total_amount + customer_state

                                + customer_segment + department_name
                                + payment_type,
                             data = train_data, method = "class")

print(decision_tree_model)

## n= 6219
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
## 1) root 6219 2629 Delayed (0.5772632 0.4227368)
##   2) shipping_mode=First Class,Second Class 2303  324 Delayed (0.859313
9 0.1406861) *
##   3) shipping_mode=Same Day,Standard Class 3916 1611 Not Delayed (0.411
3892 0.5886108) *

# plot the decision tree
rpart.plot(decision_tree_model)
```




```
# predict on train set
train_predictions <- predict(decision_tree_model, newdata = train_data, ty
pe = "class")

# predict on validation set
test_predictions <- predict(decision_tree_model, newdata = test_data, type
 = "class")

# caculate train metrics
train_conf_matrix <- confusionMatrix(factor(train_predictions), factor(tra
in_data$label))

train_accuracy <- train_conf_matrix$overall["Accuracy"]
train_precision <- train_conf_matrix$byClass["Precision"]
train_recall <- train_conf_matrix$byClass["Recall"]
train_F1 <- train_conf_matrix$byClass["F1"]

# caculate train AUC
train_roc <- roc(train_data$label, as.numeric(train_predictions))

## Setting levels: control = Delayed, case = Not Delayed

## Setting direction: controls < cases

train_AUC <- train_roc$auc

# print train results
cat("Accuracy: ", train_accuracy, "\n")
## Accuracy:  0.6888567
cat("Precision: ", train_precision, "\n")
## Precision:  0.8593139
```

```r
cat("Recall: ", train_recall, "\n")
## Recall:  0.5512535
cat("F1 Score: ", train_F1, "\n")
## F1 Score:  0.6716443
cat("AUC: ", train_AUC, "\n")
## AUC:  0.7140064

# caculate test metrics
test_conf_matrix <- confusionMatrix(factor(test_predictions), factor(test_
data$label))

test_accuracy <- test_conf_matrix$overall["Accuracy"]
test_precision <- test_conf_matrix$byClass["Precision"]
test_recall <- test_conf_matrix$byClass["Recall"]
test_F1 <- test_conf_matrix$byClass["F1"]

# caculate test AUC
test_roc <- roc(test_data$label, as.numeric(test_predictions))

## Setting levels: control = Delayed, case = Not Delayed
## Setting direction: controls < cases

test_AUC <- test_roc$auc

# print test results
cat("Accuracy: ", test_accuracy, "\n")
## Accuracy:  0.6849668
cat("Precision: ", test_precision, "\n")
## Precision:  0.8611931
cat("Recall: ", test_recall, "\n")
## Recall:  0.5416048
cat("F1 Score: ", test_F1, "\n")
## F1 Score:  0.6649943
cat("AUC: ", test_AUC, "\n")
## AUC:  0.711188
```