CPE 213 Data Models

Game Sales Prediction

Member

1.นาย ปวริศ ร้านชิตวงศ์ 61070501034

2.นาย พีรภัทร เขมะชิต 61070501039

3.นาย ยศกร นุ่นปาน 61070501043

Introduction to the problem

Analytic objective

เพื่อสร้างโมเดลศึกษาความคุ้มค่าในการลงทุนผลิตเกมประเภทต่างๆของแต่ละบริษัท

Data descriptive

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
1	1	Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
2	2	Super Mario Bros.	NES	1985	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
3	3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
4	4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
5	5	Pokemon Red/Pokemon Blue	GB	1996	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37
6	6	Tetris	GB	1989	Puzzle	Nintendo	23.20	2.26	4.22	0.58	30.26
7	7	New Super Mario Bros.	DS	2006	Platform	Nintendo	11.38	9.23	6.50	2.90	30.01
8	8	Wii Play	Wii	2006	Misc	Nintendo	14.03	9.20	2.93	2.85	29.02
9	9	New Super Mario Bros. Wii	Wii	2009	Platform	Nintendo	14.59	7.06	4.70	2.26	28.62
10	10	Duck Hunt	NES	1984	Shooter	Nintendo	26.93	0.63	0.28	0.47	28.31
11	11	Nintendogs	DS	2005	Simulation	Nintendo	9.07	11.00	1.93	2.75	24.76
12	12	Mario Kart DS	DS	2005	Racing	Nintendo	9.81	7.57	4.13	1.92	23.42
13	13	Pokemon Gold/Pokemon Silver	GB	1999	Role-Playing	Nintendo	9.00	6.18	7.20	0.71	23.10
14	14	Wii Fit	Wii	2007	Sports	Nintendo	8.94	8.03	3.60	2.15	22.72
15	15	Wii Fit Plus	Wii	2009	Sports	Nintendo	9.09	8.59	2.53	1.79	22.00

Data preparation

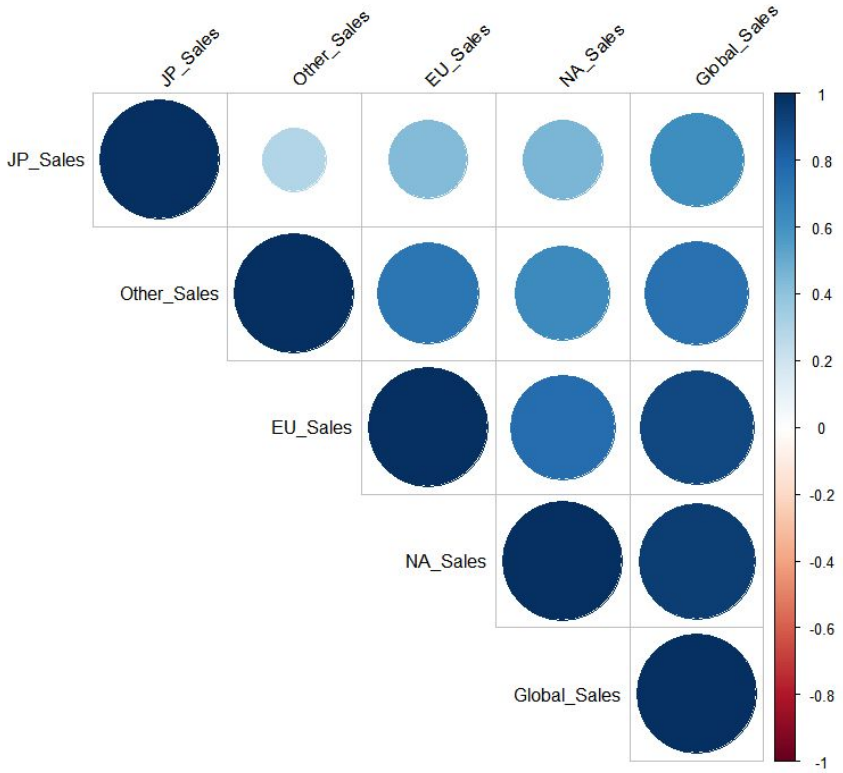
```
game %>% select(-Rank) %>% filter(Year != "N/A" ) %>% filter(Year != "2020") %>% filter(Year != "2017") %>%
  filter(Publisher!= "N/A") -> FilterGame

FilterGame$Year <- as.numeric(as.character(FilterGame$Year))

FilterGame %>% group_by(Publisher) %>% summarise(n=n()) %>% filter(n>100) -> nopublish

FilterGame %>% filter(Publisher %in% nopublish$Publisher) ->FilterGame
```

Data preparation

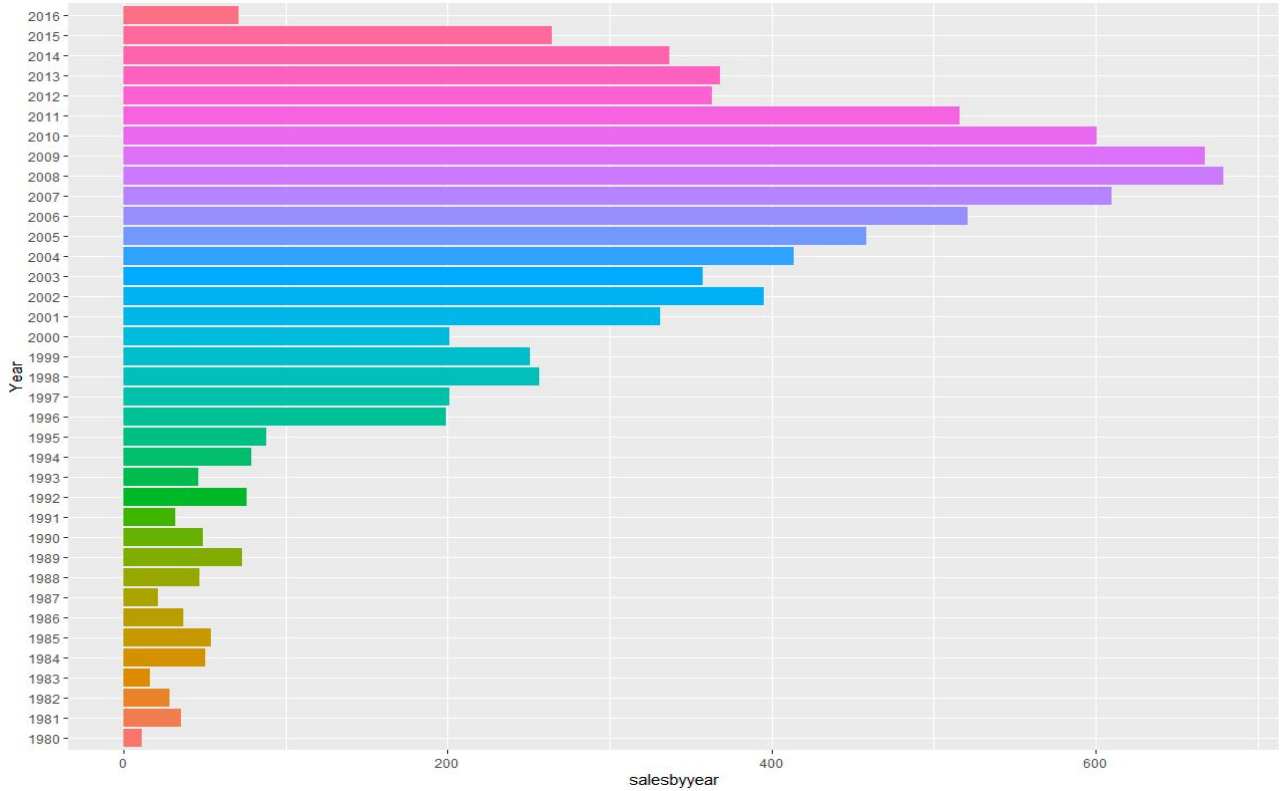
	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
1	Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
2	Super Mario Bros.	NES	1985	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
5	Pokemon Red/Pokemon Blue	GB	1996	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37
6	Tetris	GB	1989	Puzzle	Nintendo	23.20	2.26	4.22	0.58	30.26
7	New Super Mario Bros.	DS	2006	Platform	Nintendo	11.38	9.23	6.50	2.90	30.01
8	Wii Play	Wii	2006	Misc	Nintendo	14.03	9.20	2.93	2.85	29.02
9	New Super Mario Bros. Wii	Wii	2009	Platform	Nintendo	14.59	7.06	4.70	2.26	28.62
10	Duck Hunt	NES	1984	Shooter	Nintendo	26.93	0.63	0.28	0.47	28.31
11	Nintendogs	DS	2005	Simulation	Nintendo	9.07	11.00	1.93	2.75	24.76
12	Mario Kart DS	DS	2005	Racing	Nintendo	9.81	7.57	4.13	1.92	23.42
13	Pokemon Gold/Pokemon Silver	GB	1999	Role-Playing	Nintendo	9.00	6.18	7.20	0.71	23.10
14	Wii Fit	Wii	2007	Sports	Nintendo	8.94	8.03	3.60	2.15	22.72

เมื่อทำการกรองข้อมูลเสร็จจะได้ข้อมูลจาก 16,598 แถวเหลือ 11,735 แถว

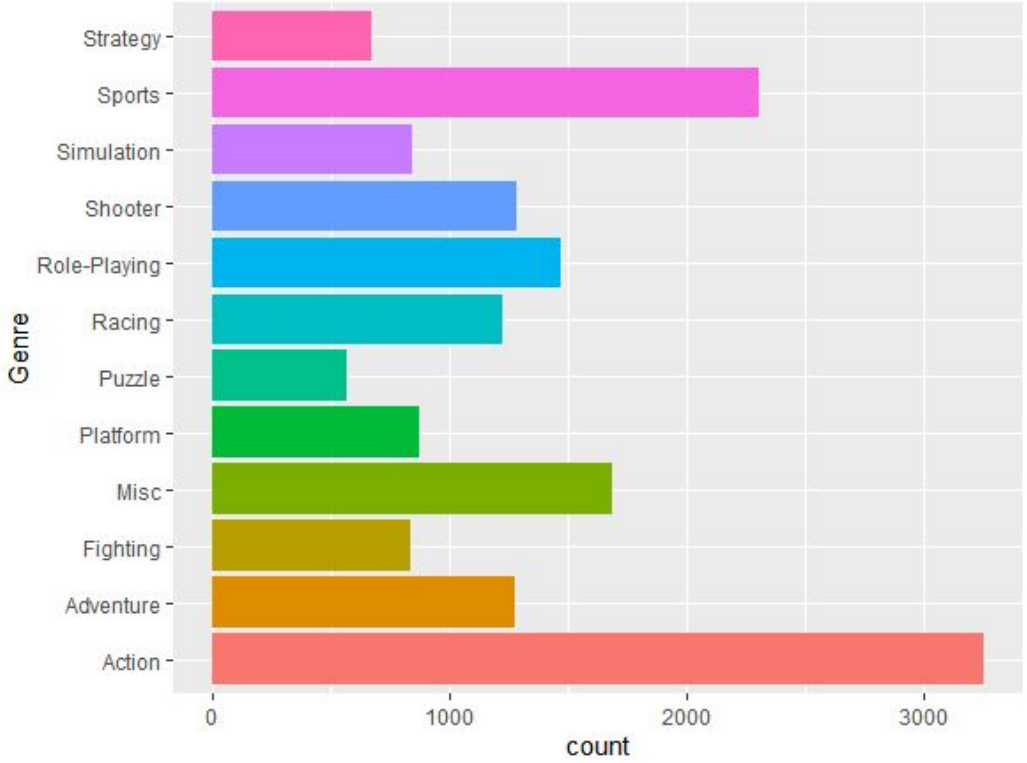
Data exploration and visualization



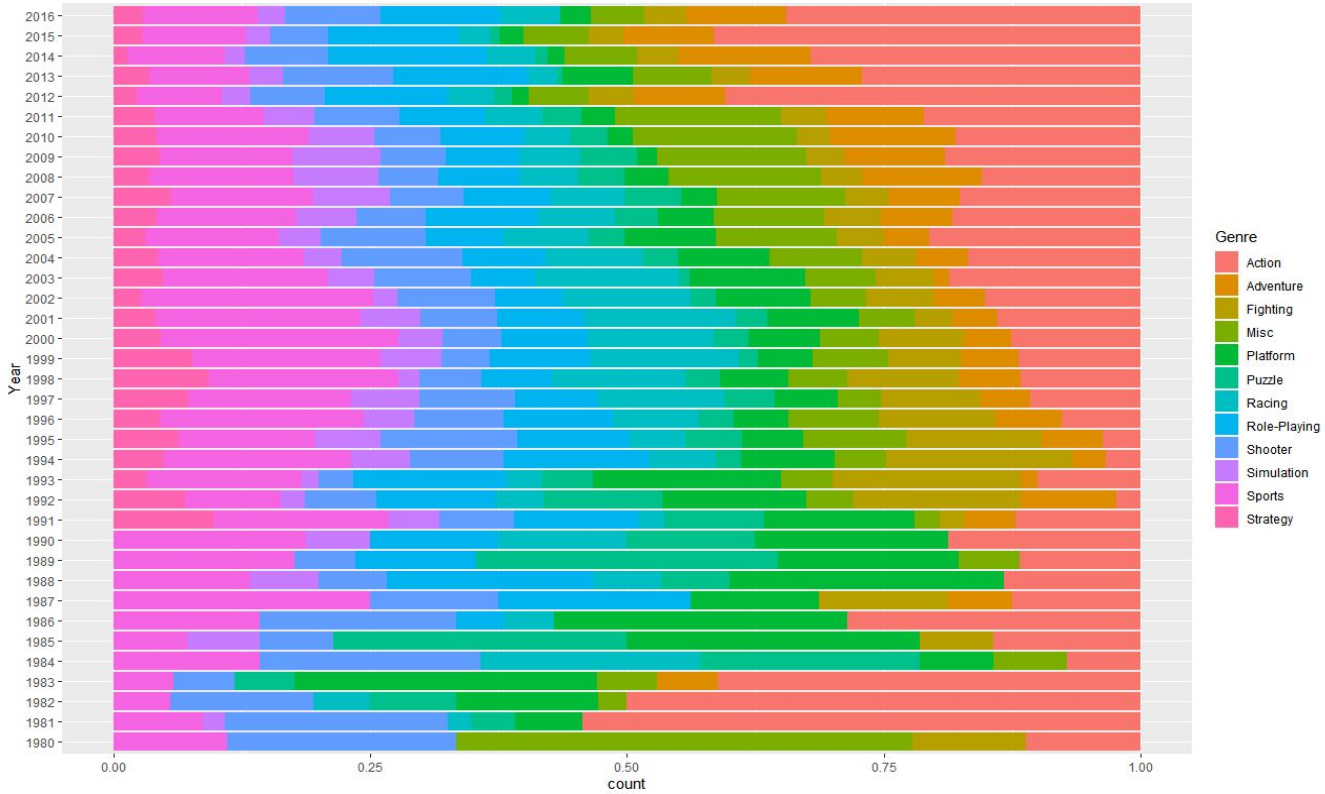
Data exploration and visualization



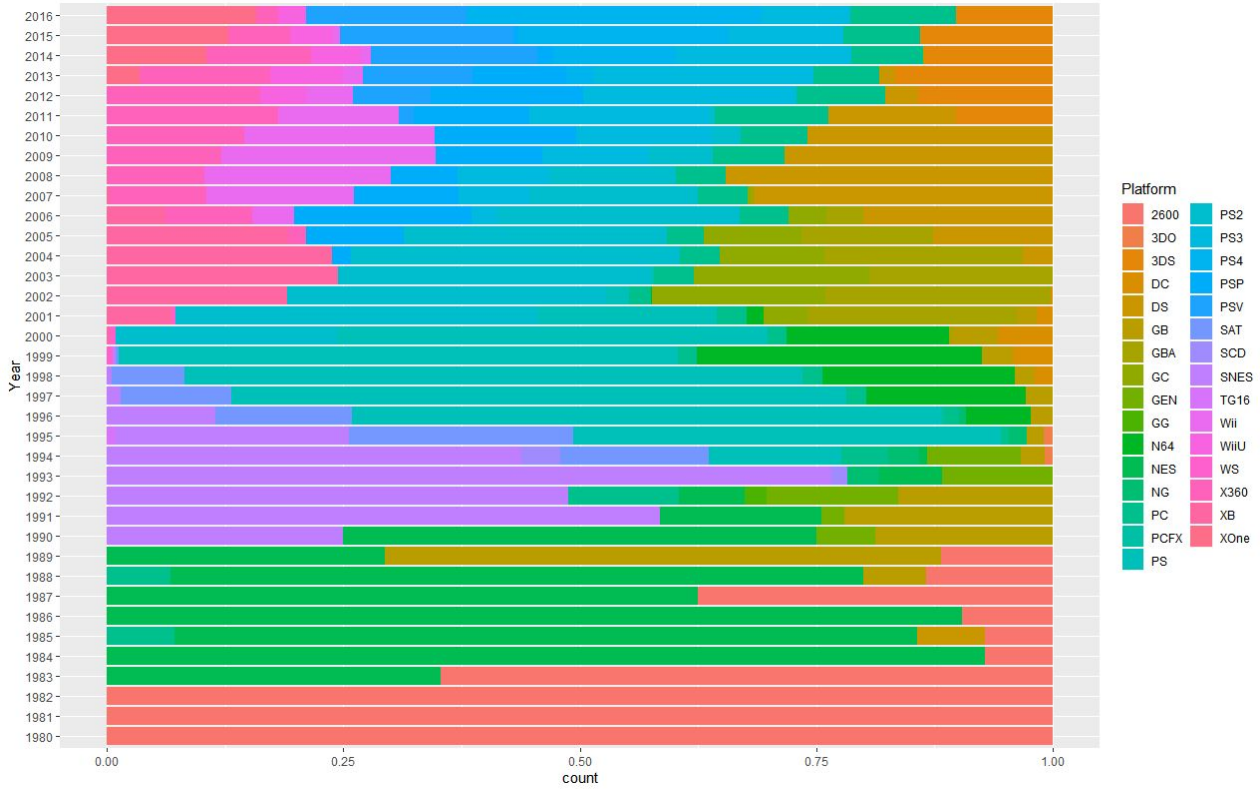
Data exploration and visualization



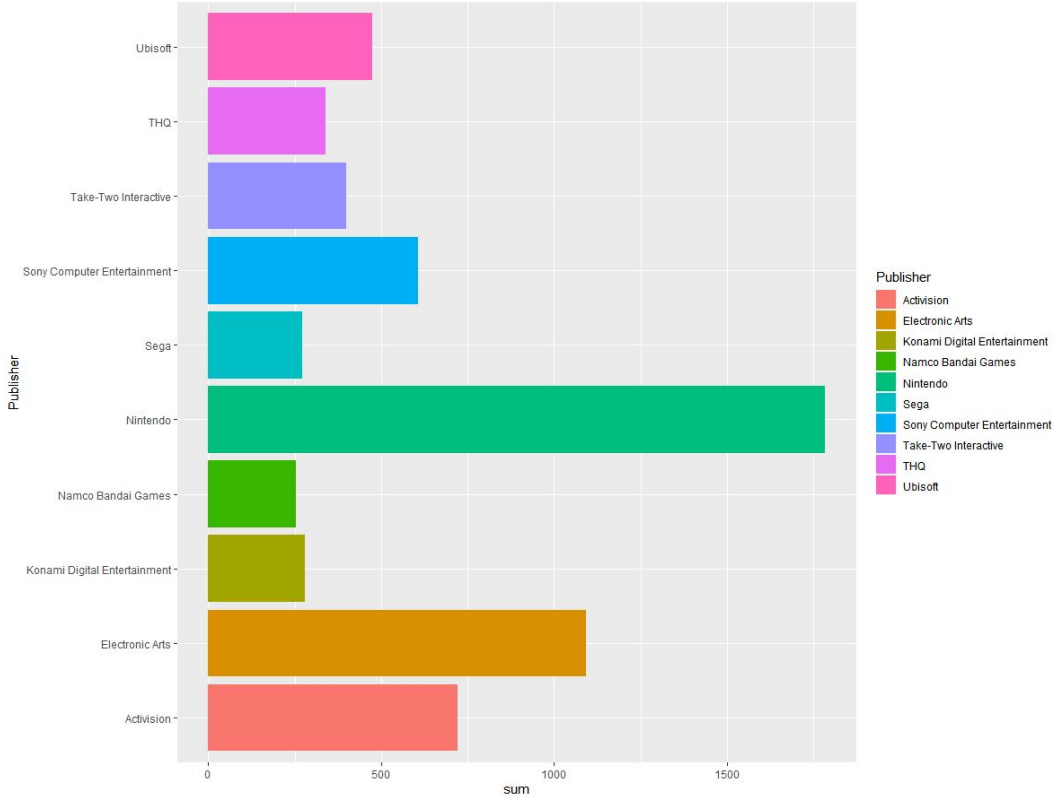
Data exploration and visualization



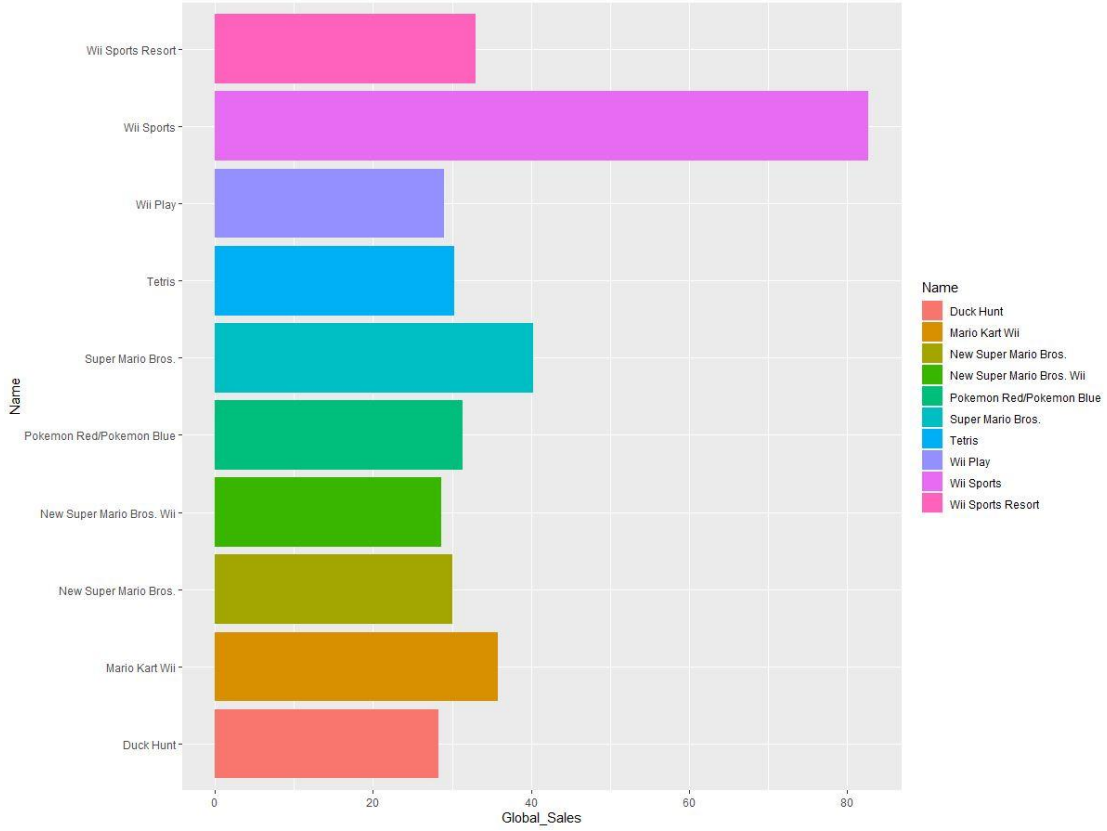
Data exploration and visualization



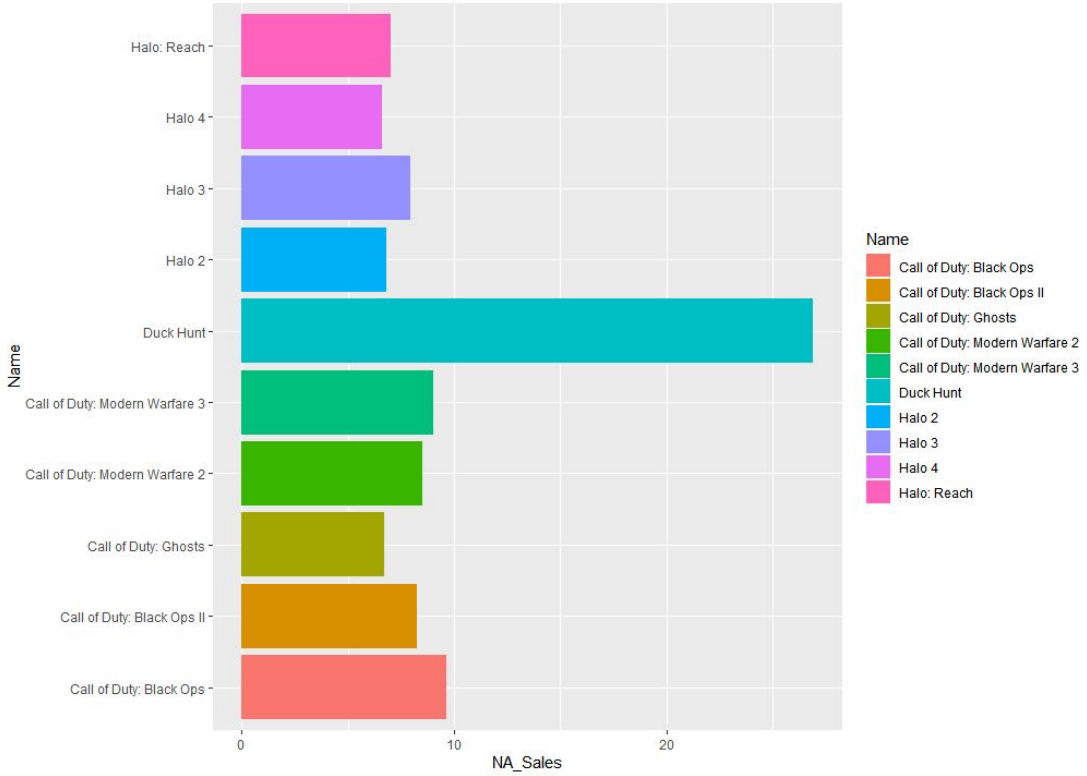
Data exploration and visualization



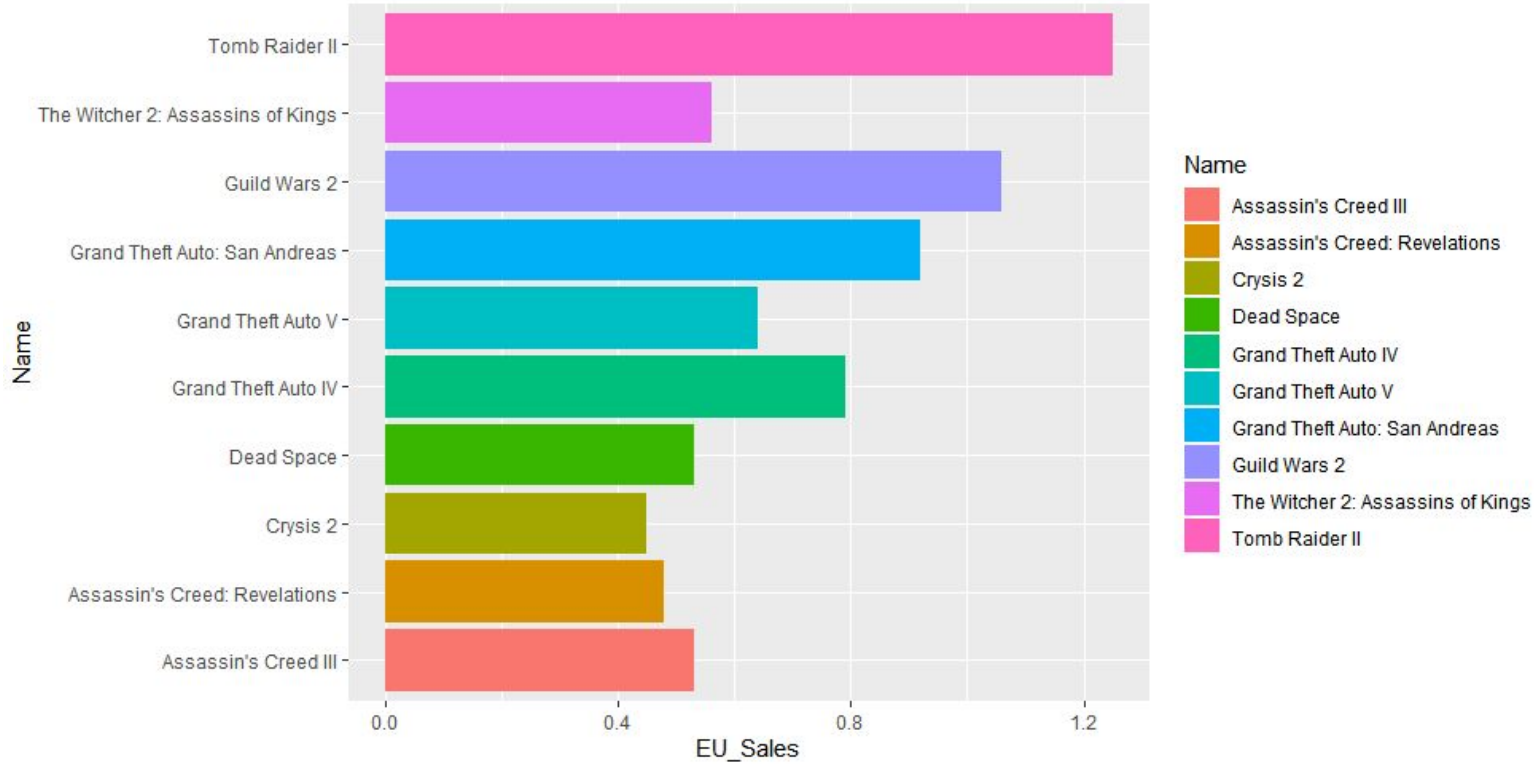
Data exploration and visualization



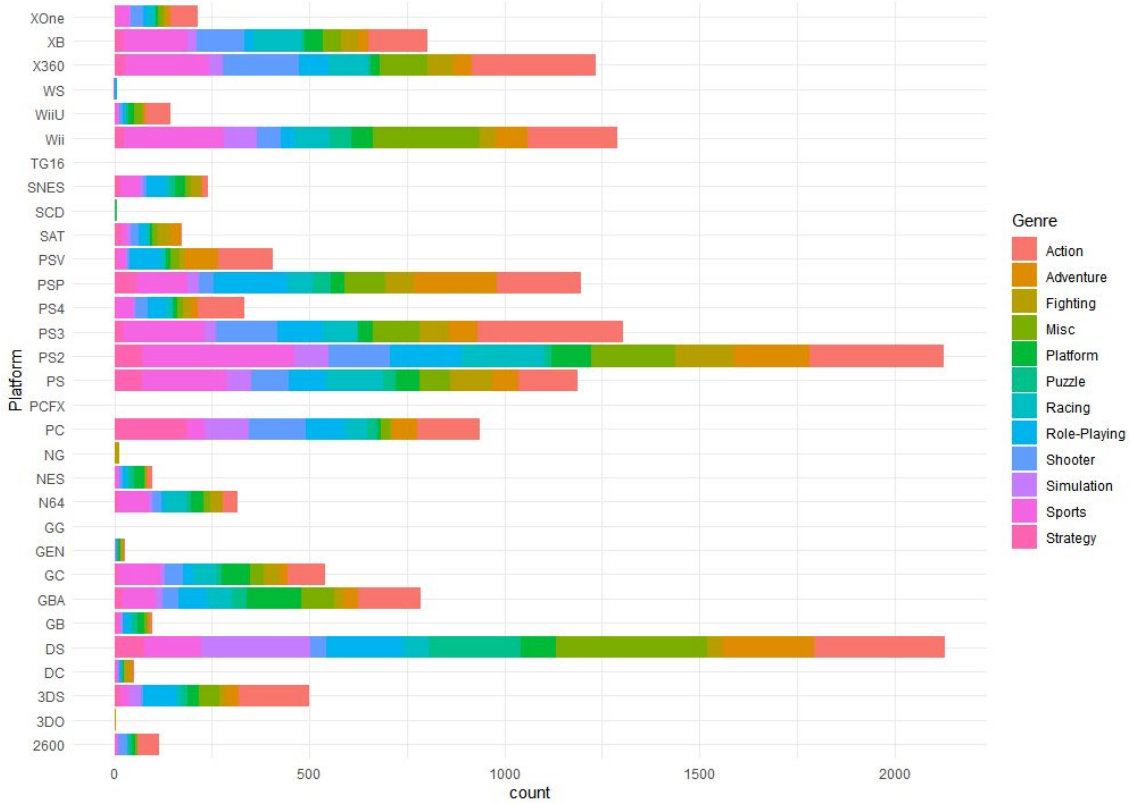
Data exploration and visualization



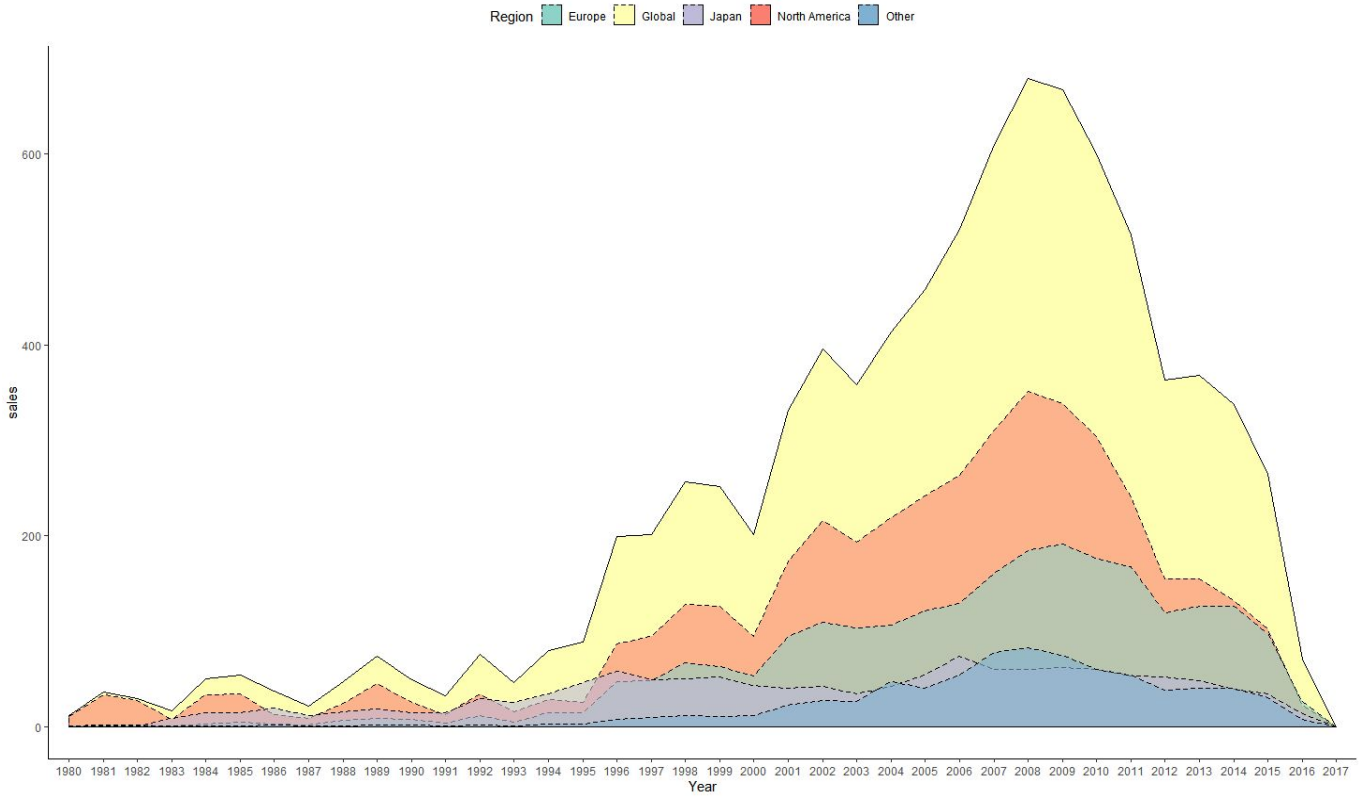
Data exploration and visualization



Data exploration and visualization



Data exploration and visualization



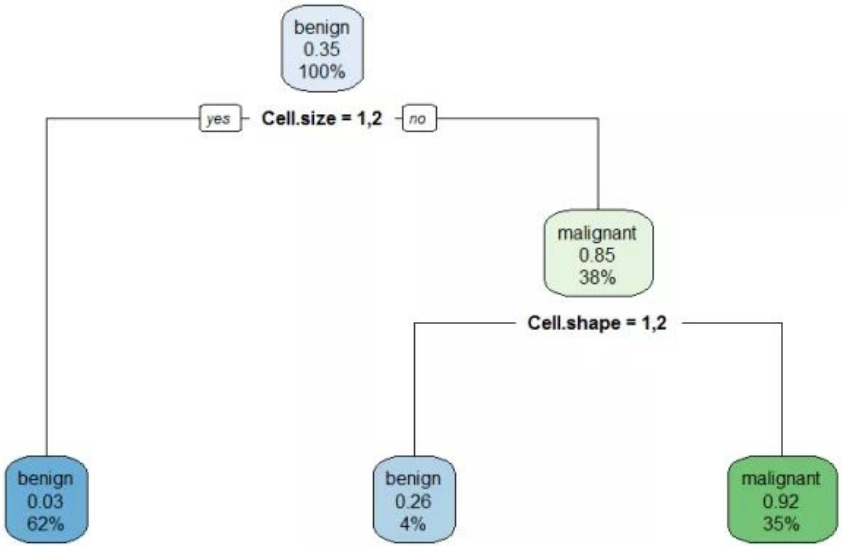
Model Explanation

Platform	Genre	Publisher
Wii	Sports	Nintendo
NES	Platform	Nintendo
Wii	Racing	Nintendo
Wii	Sports	Nintendo
GB	Role-Playing	Nintendo
GB	Puzzle	Nintendo
DS	Platform	Nintendo
Wii	Misc	Nintendo
Wii	Platform	Nintendo
NES	Shooter	Nintendo
DS	Simulation	Nintendo
DS	Racing	Nintendo
GB	Role-Playing	Nintendo
Wii	Sports	Nintendo
Wii	Sports	Nintendo
X360	Misc	Microsoft Game Studios

Model Explanation

Decision tree model

EX.



Modeling Implementation

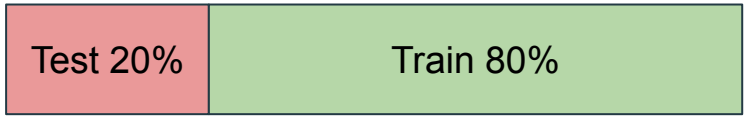
Decision Tree model

```
FilterGame %>% mutate(success = cut(Global_Sales,
                                    breaks = c(0,0.2,Inf),
                                    labels = c("n","y"))) -> FilterGame
```

Modeling Implementation

Decision Tree model

```
FilterGame %>% select(-NA_Sales,-Name,-EU_Sales,-JP_Sales,-Other_Sales,-Global_Sales) -> GameData

set.seed(555)
test_ind <- sample(nrow(GameData),
                   0.2*nrow(GameData))
Data_training <- GameData[-test_ind,]
Data_testing <- GameData[test_ind,]
```



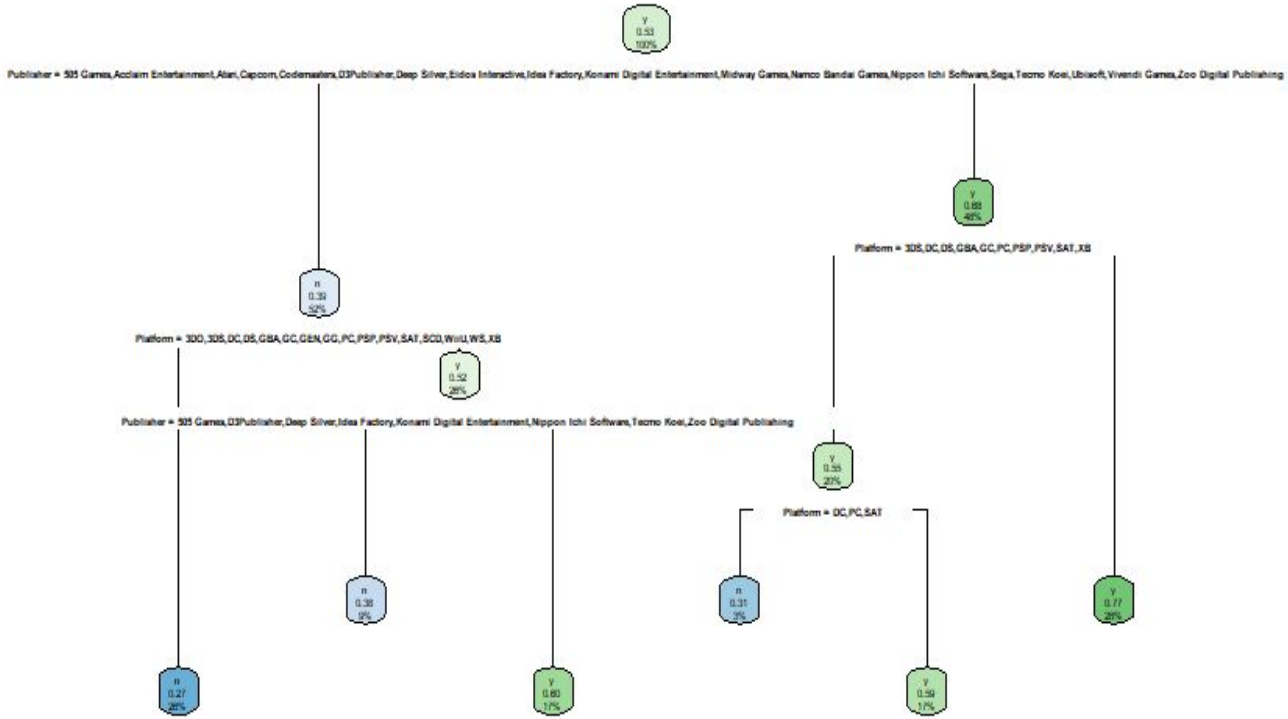
Modeling Implementation

Decision Tree model

```
tree <- rpart(success ~ ., data = Data_training)
rpart.plot(tree)
```

Modeling Implementation

Decision Tree model



Modeling Implementation

Variable Importance

```
> tree$variable.importance
Publisher  Platform     Genre      Year
462.52501 341.52441  67.69058  19.82499
```

Modeling Implementation

สร้าง Confusion Matrix

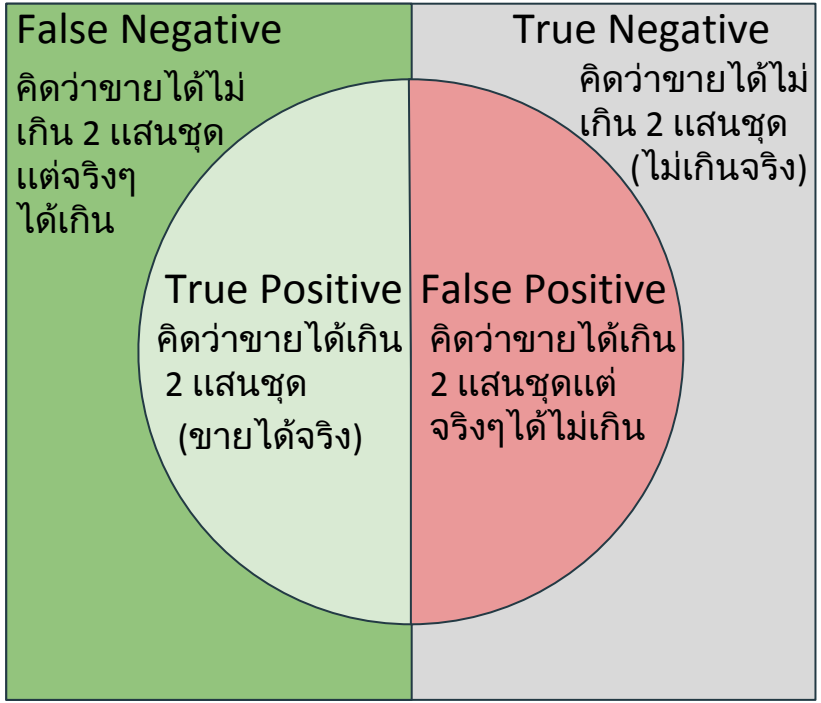
```
res<-predict(tree,Data_testing,type = "class")
```

```
confusionMatrix(res,
                Data_testing$success,
                positive = "y",
                mode = "prec_recall")
```

Confusion Matrix

```
Confusion Matrix and Statistics

          Reference
Prediction   n   y
         n 621 247
         y 493 986

               Accuracy : 0.6847
                 95% CI : (0.6655, 0.7035)
    No Information Rate : 0.5254
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.361

 Mcnemar's Test P-Value : < 2.2e-16

              Precision : 0.6667
                 Recall : 0.7997
                     F1 : 0.7271
             Prevalence : 0.5254
         Detection Rate : 0.4201
   Detection Prevalence : 0.6302
      Balanced Accuracy : 0.6786

       'Positive' Class : y
```

Evaluation




```
Confusion Matrix and Statistics

          Reference
Prediction   n   y
         n 621 247
         y 493 986

               Accuracy : 0.6847
                 95% CI : (0.6655, 0.7035)
    No Information Rate : 0.5254
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.361

 Mcnemar's Test P-Value : < 2.2e-16

              Precision : 0.6667
                 Recall : 0.7997
                     F1 : 0.7271
             Prevalence : 0.5254
         Detection Rate : 0.4201
   Detection Prevalence : 0.6302
      Balanced Accuracy : 0.6786

       'Positive' Class : y
```

Discussion and Conclusion

```
Confusion Matrix and Statistics

          Reference
Prediction   n   y
         n 621 247
         y 493 986

               Accuracy : 0.6847
                 95% CI : (0.6655, 0.7035)
    No Information Rate : 0.5254
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.361

 Mcnemar's Test P-Value : < 2.2e-16

              Precision : 0.6667
                 Recall : 0.7997
                     F1 : 0.7271
             Prevalence : 0.5254
         Detection Rate : 0.4201
   Detection Prevalence : 0.6302
      Balanced Accuracy : 0.6786

       'Positive' Class : y
```


Thank you!