Kobe Shot Selection

KOBE BRYANT

#10 USA BASKETBALL

Content

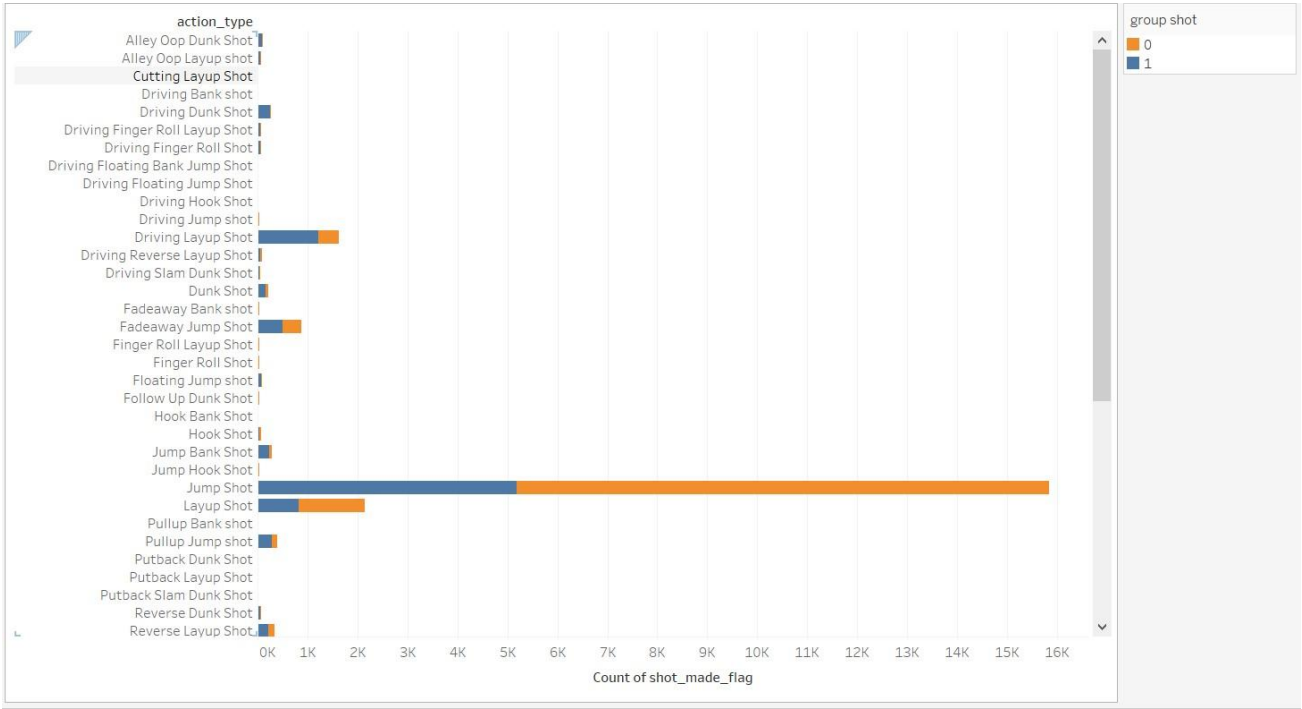
- Data exploration
- Data preparation
- Data modeling
- Model evaluation
- Result

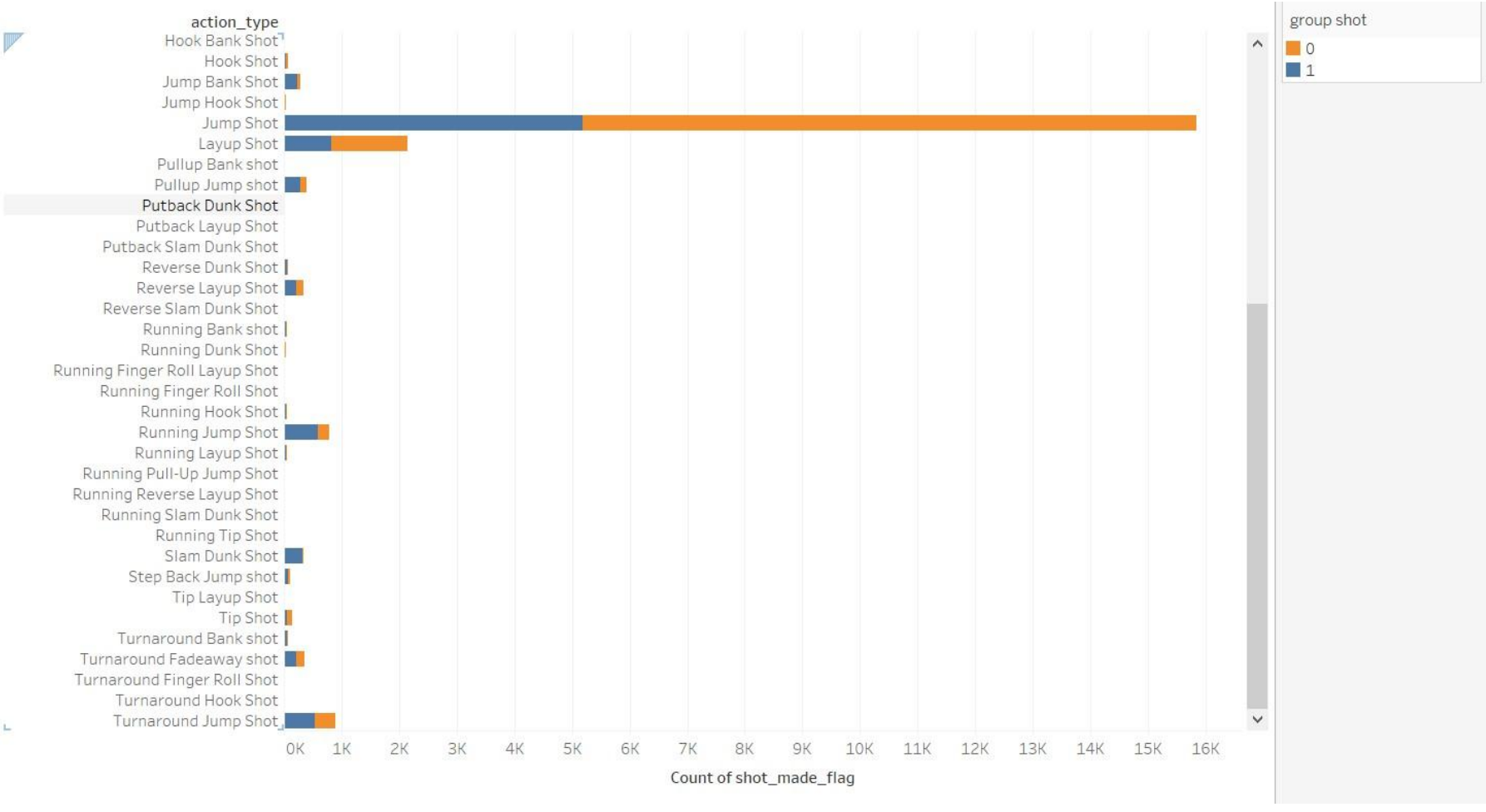
Data Exploration

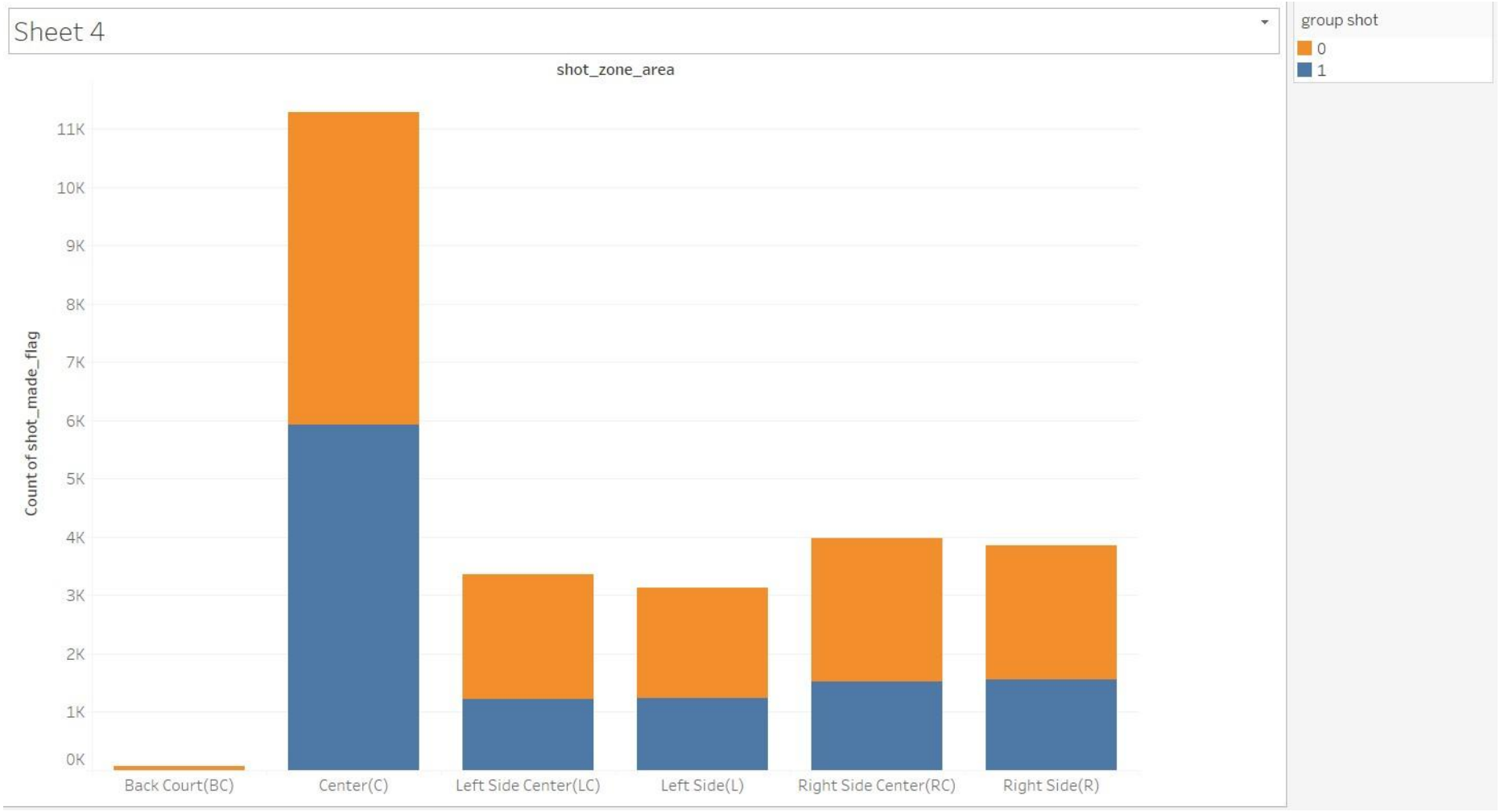
- We are going to explore the data of the selection data set.
- We list out the feature that we will use for the prediction model.

```
##  [1] "action_type"       "combined_shot_type" "game_event_id"
##  [4] "game_id"           "lat"                "loc_x"
##  [7] "loc_y"             "lon"                "minutes_remaining"
## [10] "period"            "playoffs"           "season"
## [13] "seconds_remaining" "shot_distance"      "shot_made_flag"
## [16] "shot_type"         "shot_zone_area"     "shot_zone_basic"
## [19] "shot_zone_range"   "team_id"            "team_name"
## [22] "game_date"         "matchup"            "opponent"
## [25] "shot_id"
```

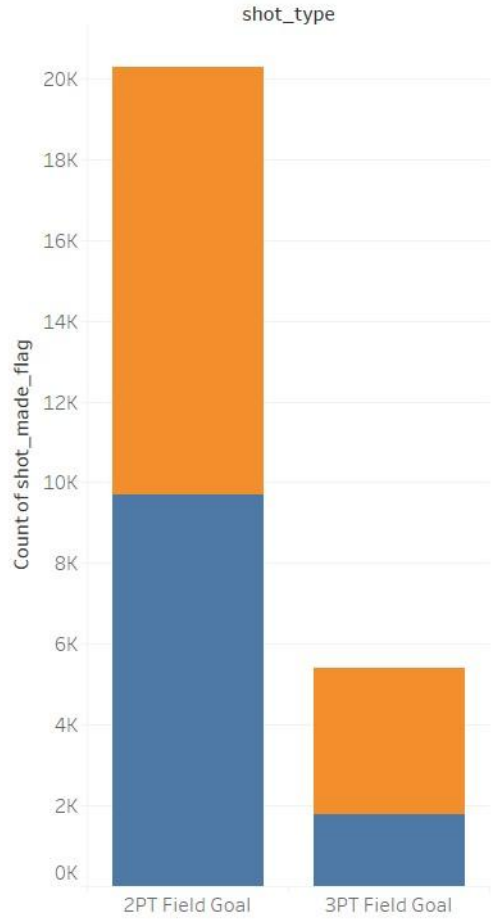
Let see kobe accuracy with each of this features by graph.

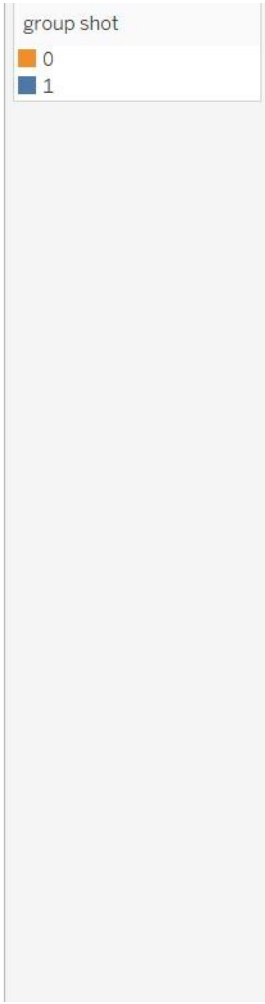


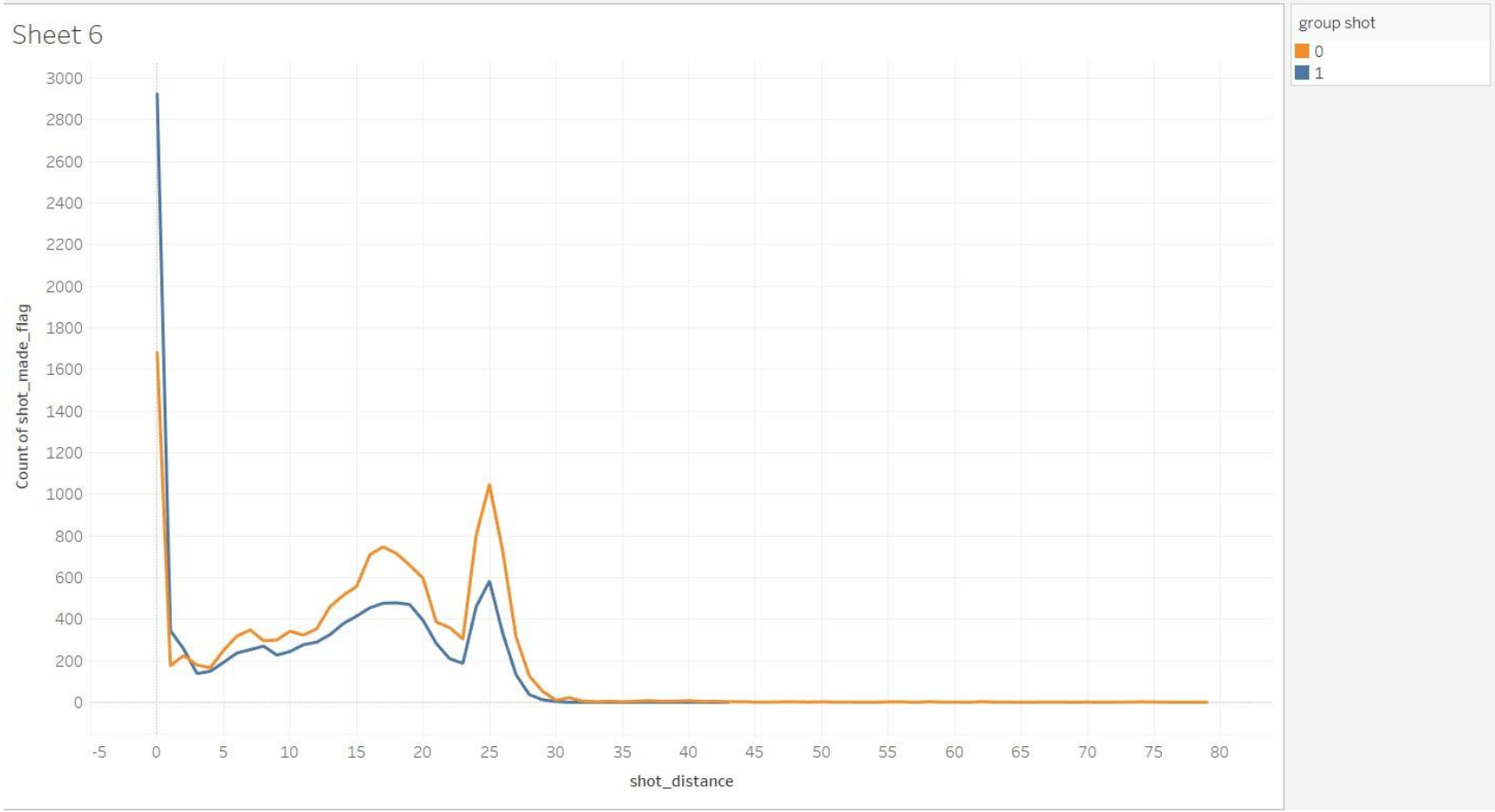

action_type
Hook Bank Shot
Hook Shot
Jump Bank Shot
Jump Hook Shot
Jump Shot
Layup Shot
Pullup Bank shot
Pullup Jump shot
Putback Dunk Shot
Putback Layup Shot
Putback Slam Dunk Shot
Reverse Dunk Shot
Reverse Layup Shot
Reverse Slam Dunk Shot
Running Bank shot
Running Dunk Shot
Running Finger Roll Layup Shot
Running Finger Roll Shot
Running Hook Shot
Running Jump Shot
Running Layup Shot
Running Pull-Up Jump Shot
Running Reverse Layup Shot
Running Slam Dunk Shot
Running Tip Shot
Slam Dunk Shot
Step Back Jump shot
Tip Layup Shot
Tip Shot
Turnaround Bank shot
Turnaround Fadeaway shot
Turnaround Finger Roll Shot
Turnaround Hook Shot
Turnaround Jump Shot
0K
1K
2K
3K
4K
5K
6K
7K
8K
9K
10K
11K
12K
13K
14K
15K
16K
Count of shot_made_flag
group shot
0
1


Sheet 4
shot_zone_area
group shot
0
1
Count of shot_made_flag
11K
10K
9K
8K
7K
6K
5K
4K
3K
2K
1K
0K
Back Court(BC)
Center(C)
Left Side Center(LC)
Left Side(L)
Right Side Center(RC)
Right Side(R)

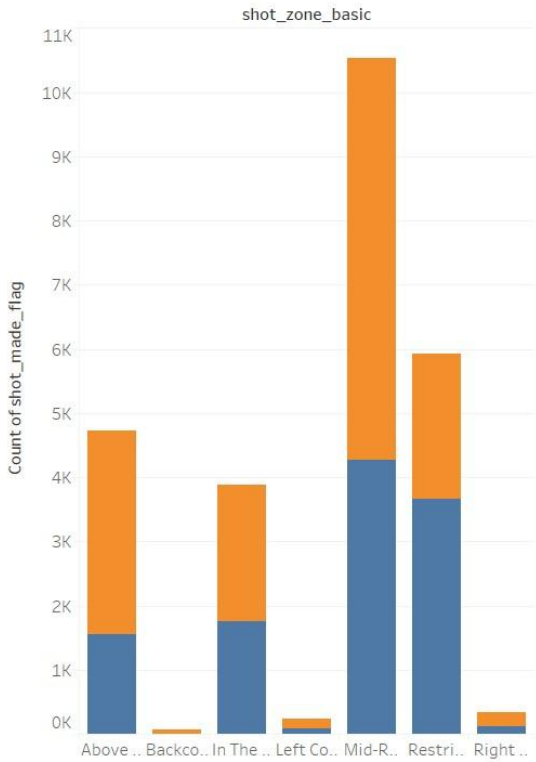
Sheet 5


shot_type
Count of shot_made_flag
20K
18K
16K
14K
12K
10K
8K
6K
4K
2K
0K
2PT Field Goal
3PT Field Goal

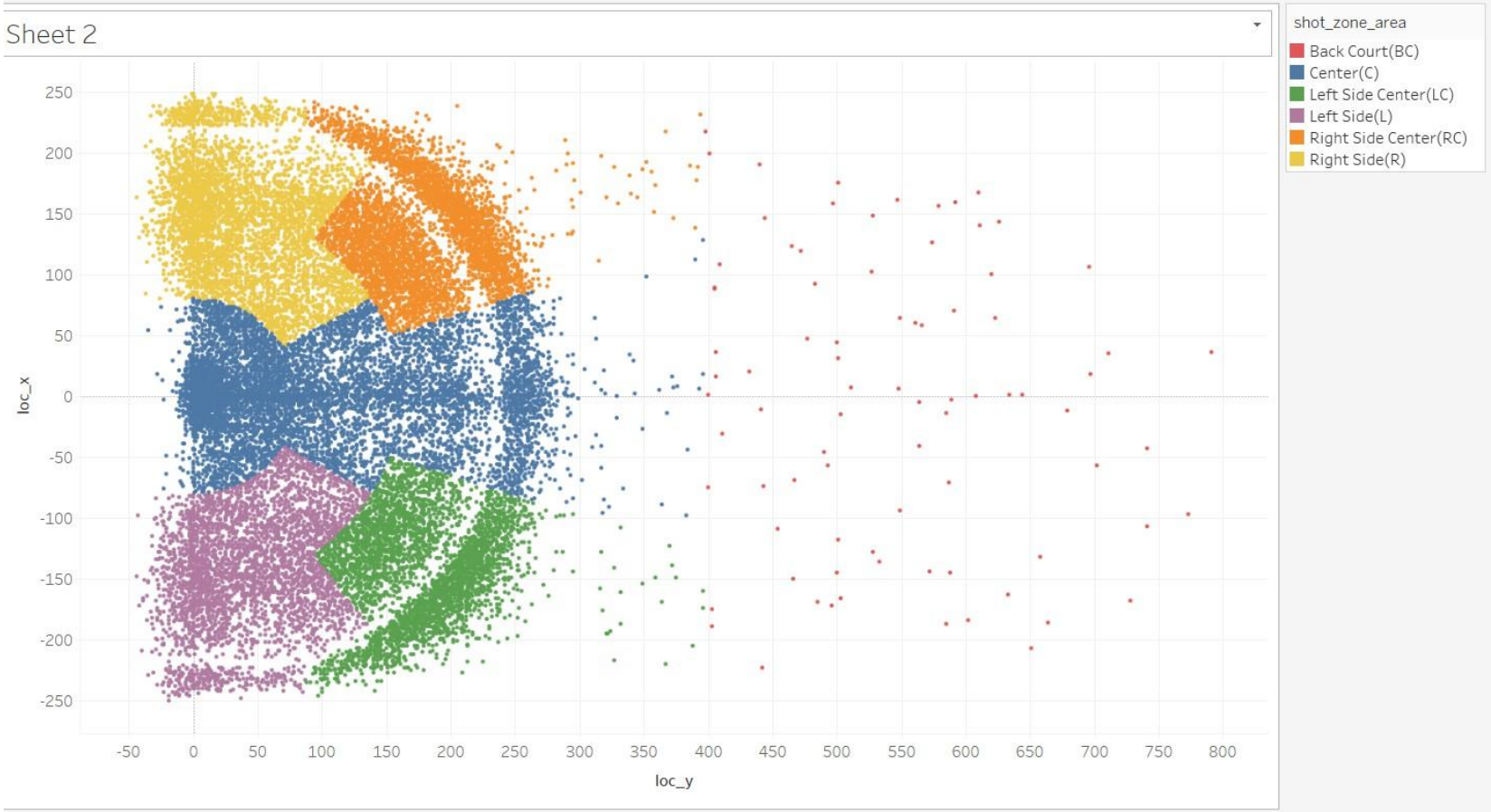

group shot
0
1

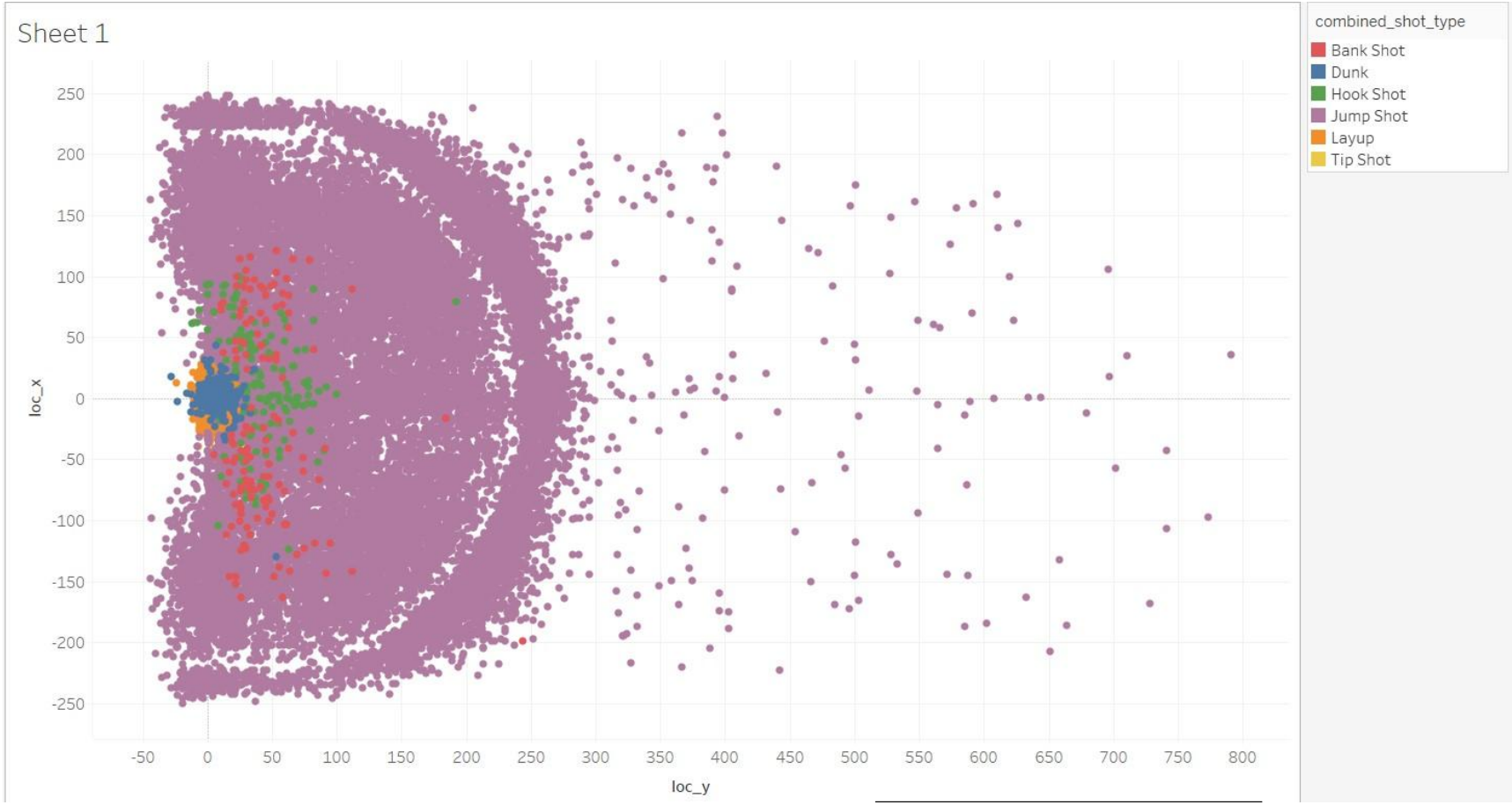

Sheet 6
group shot
0
1
Count of shot_made_flag
3000
2800
2600
2400
2200
2000
1800
1600
1400
1200
1000
800
600
400
200
0
-5
0
5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
shot_distance

Sheet 7



group shot
0
1

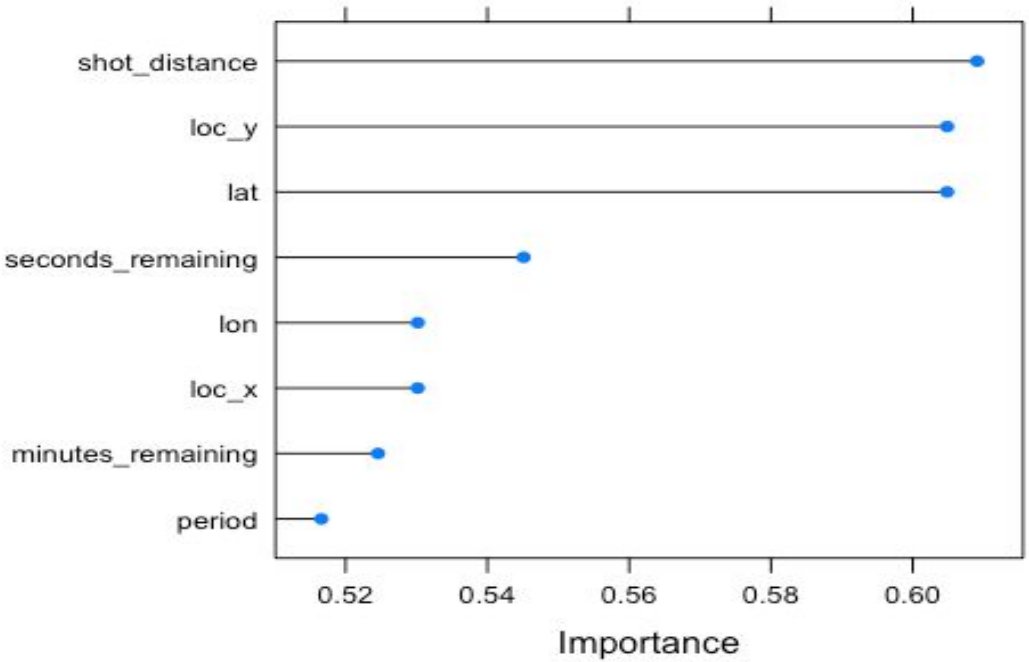

Sheet 2
shot_zone_area
Back Court(BC)
Center(C)
Left Side Center(LC)
Left Side(L)
Right Side Center(RC)
Right Side(R)
loc_x
250
200
150
100
50
0
-50
-100
-150
-200
-250
loc_y
-50
0
50
100
150
200
250
300
350
400
450
500
550
600
650
700
750
800


Sheet 1
combined_shot_type
Bank Shot
Dunk
Hook Shot
Jump Shot
Layup
Tip Shot
loc_x
250
200
150
100
50
0
-50
-100
-150
-200
-250
loc_y
-50
0
50
100
150
200
250
300
350
400
450
500
550
600
650
700
750
800

Data preparation

- We are going to prepare the data of the selection data set for modeling.
- There are some method for the preparation of the selected prediction model.
- We also use Learning Vector Quantization model to help us in finding suitable features for the prediction model.

Learning Vector Quantization



Method to preparation.

- The first step we convert the date(yyyy-mm-dd) to weekday, month, and year.

```
11  # Convert date to weekday and other
12  install.packages("lubridate") #remove comment here if you didn't install this package yet
13  library(lubridate)
14  train$week_day <- wday(train$game_date)
15  train$month <- month(train$game_date)
16  train$year <- year(train$game_date)
17
```

Method to preparation.

- The next step we will prepare the data by normalize the data into range 0:1.

```r
# Function : normalize data into range 0:1
normalize <- function(data) {
  data <- (data - min(data)) / (max(data) - min(data))
}
```

```r
# Normalize data
train$loc_x <- normalize(train$loc_x)
train$loc_y <- normalize(train$loc_y)
train$shot_distance <- normalize(train$shot_distance)
train$time_remaining <- (train$minutes_remaining*60) + train$seconds_remaining
train$time_remaining <- normalize(train$time_remaining)
train$period <- normalize(train$period)
train$matchup <- ifelse( grepl("@",train$matchup), train$matchup <- "AWAY", train$matchup <- "HOME" )
train$week_day <- normalize(train$week_day)
train$month <- normalize(train$month)
train$year <- normalize(train$year)
```

Method to preparation.

- The next step we will remove unused features.

```
# Remove unused attributes (Useless for data modeling such as _id)
train$seconds_remaining <- NULL  # use time_remaining instead
train$minutes_remaining <- NULL  # use time_remaining instead
train$game_id <- NULL
train$game_event_id <- NULL
train$game_date <- NULL          # use weekday,month,year
train$team_id <- NULL
train$team_name <- NULL          # Kobe NEVER play for other teams
train$season <- NULL
train$shot_id <- NULL
train$lat <- NULL
train$lon <- NULL
```

Method to preparation.

- The next step we used the library(dummies)
- We used this library to convert categorical data to binary matrix so it is useable in the model.

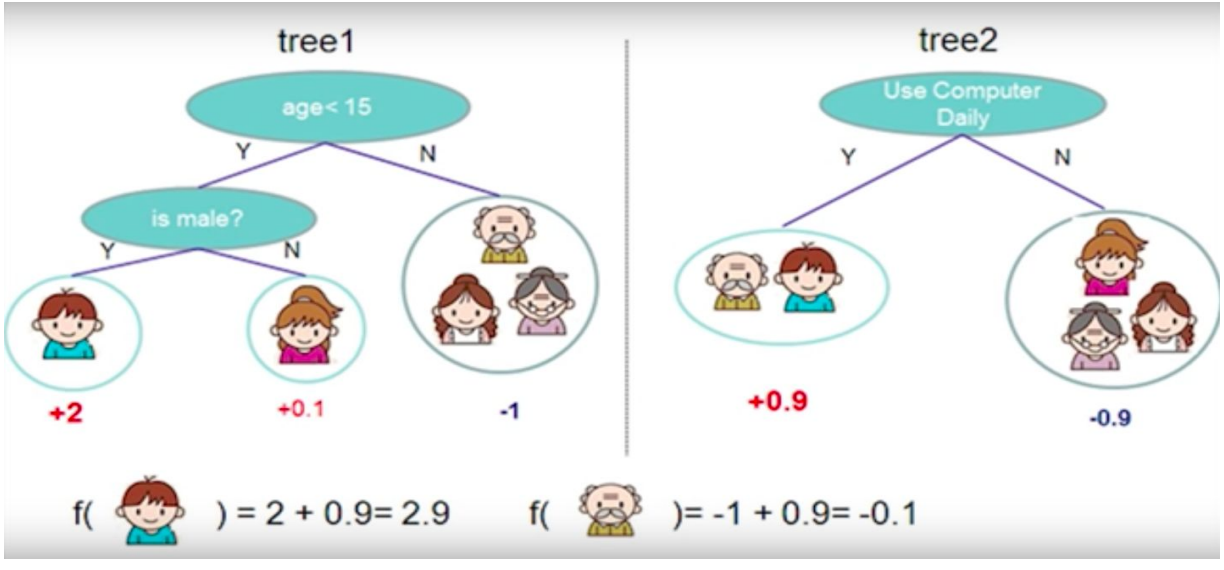
```
43  # Prepare for model
44  install.packages("dummies") #remove comment here if you didn't install this package yet
45  library(dummies)
46  df<- dummy.data.frame(train , names = c('action_type','combined_shot_type' , 'shot_type' ,
47  'shot_zone_area' ,'playoffs' , 'shot_zone_basic' , 'shot_zone_range' , 'matchup' , 'opponent') , sep='_')
48  X_train <- df[(!is.na(df$shot_made_flag)),]
49  X_test <- df[(is.na(df$shot_made_flag)),]
50  y <- X_train$shot_made_flag
51
```

Data Modeling

25,697 records as train dataset and 5,000 as test dataset

Xgboost - execute quickly and get good model. Based on **Decision tree**

We tune the objective parameter to **binary:logistic (logistic regression)** because our problem is classification. Whether the shot success or not (1 or 0)



Model Evaluation

We test our model accuracy by sampling data from train dataset 75% remains as model and 25% for testing the model

Confusion matrix	Actual Positive (shot_made_ flag = 1)	Actual Negative (shot_made_ flag = 0)
Model say YES (shot_made_ flag >= 0.5)	TP = 1371	FP = 507
Model say No (shot_made_ flag < 0.5)	FN = 1543	TN = 3004

Accuracy ~ 70% Precision ~73% Recall ~ 50%

What is the problem ? - Prediction in range ~ 0.35-0.65

Another evaluation

However, we got **34th** rank in the Kaggle leaderboard with score 0.5998

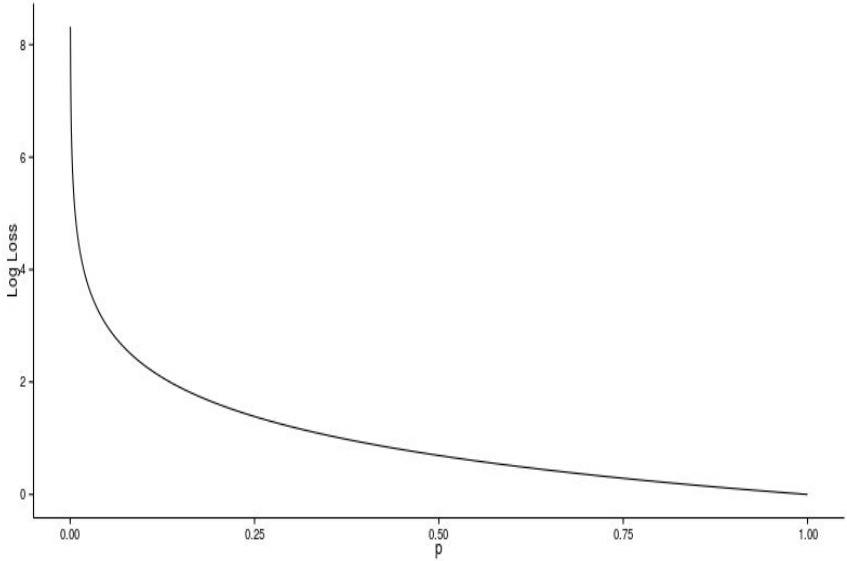
Name	Submitted	Wait time	Execution time	Score
kobe_prediction_result.csv	9 hours ago	0 seconds	0 seconds	0.59986

Complete

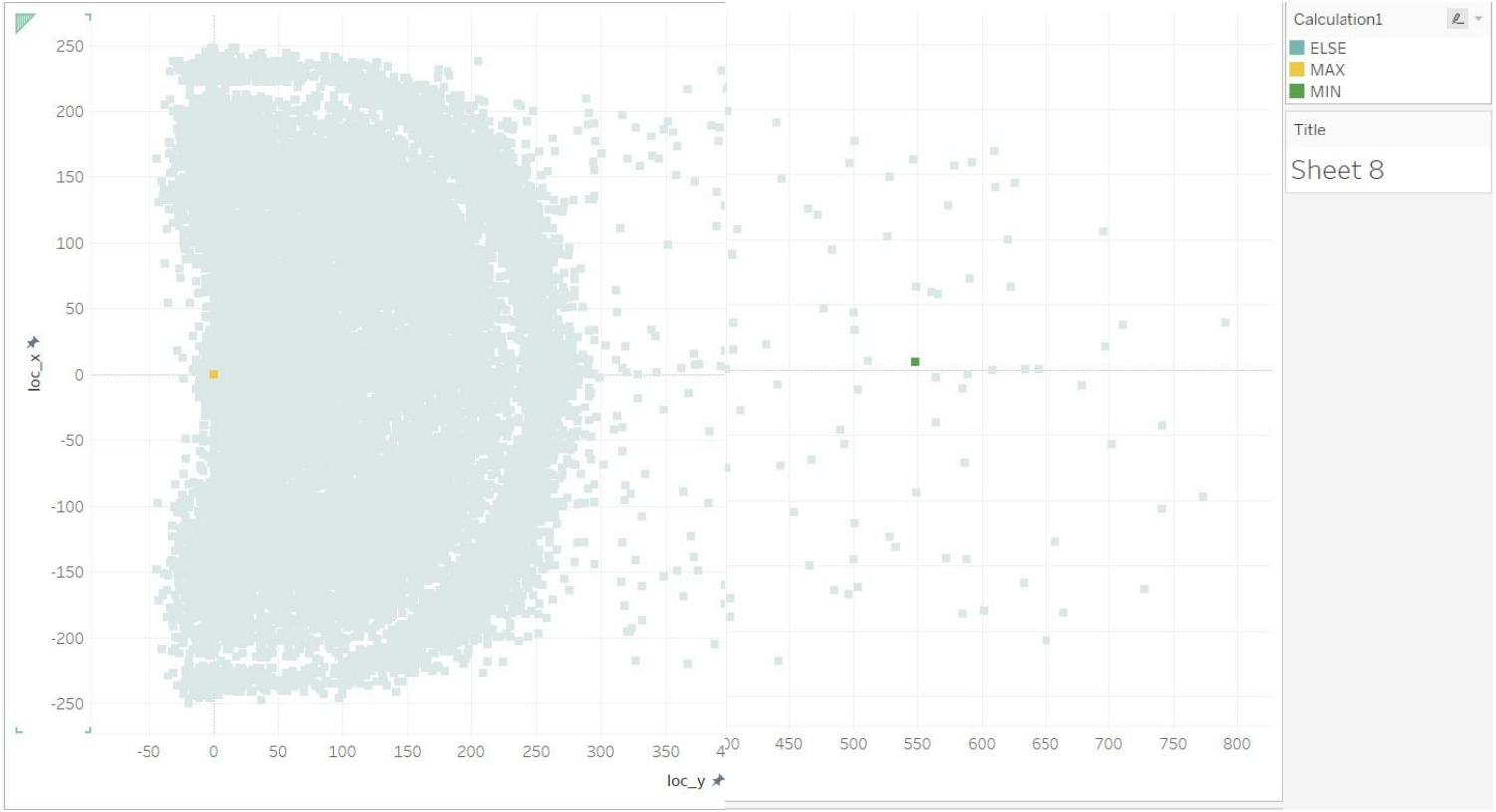
Jump to your position on the leaderboard

1	—	Humberto Brandão	0.56529	50	1y
2	—	y130038	0.56977	15	1y
3	—	The Black Mamba Team	0.57222		
4	—	kshain	0.57414		
5	—	Pavel Shashkin	0.57509		
6	—	LeBronLearnsML	0.58713		
7	—	Jean Souquet	0.58943		
8	—	derekatwood	0.58949		
9	—	Juan Aguilera	0.59134		
10	—	Gaspard Jallat	0.59741		

Log Loss metric



Result


Calculation1
ELSE
MAX
MIN
Title
Sheet 8
loc_x
loc_y
250
200
150
100
50
0
-50
-100
-150
-200
-250
-50
0
50
100
150
200
250
300
350
450
500
550
600
650
700
750
800