EA Advertising

Roy Wang

12/27/2019

Consulting task

We are consulting for Electronic Arts or EA, the maker of popular video games. EA regularly advertises its new video games in mobile apps through a major mobile ad-network. The ad-network does behavioral tracking and engages with many different advertisers. EA wants to understand how consumer response to its own ads is influenced by the other ads shown earlier in a mobile app-session. A session is defined as a period of uninterrupted app usage by a user. For example, if a mobile user is playing Angry Birds for ten minutes on the mobile app and then takes a break for one hour to later come back and the spends six minutes on the Pinterest app, the ten minutes on Angry birds would count as one session, and the six minutes on Pinterest as another session. EA also wants to understand if it should focus more on user's behavior within a sessior or their behavior prior to a session, when it tries to predict their ad-response. In short, EA is interested in answers to the following questions related to previous ad exposure and behavioral history of the user, and the ROI from advertising: 1. First, what is the value of the past history of the user in the ability to predict clicks? Specifically, what is the relative value of session-level history (user behavior earlier in this session) vs. history prior to the session? 2. Second, does the variety of ads that a consumer has seen earlier in the session affect her probability of response to the ad currently being shown to her? For example, am I more likely to click when I have seen five different ads earlier in the session or when I have been seeing the same ad? 3. The ad-network does CPM pricing, i.e., charges EA for each impression. EA wants to understand – (1) What is ROI from the current advertising strategy where it buy all the impressions of these active users, and (2) What would be the ROI if EA instead restricted itself to approximately the top 25% of impressions (in terms of predicted CTR)? To answer these questions, EA ran a large-scale field experiment in conjunction with the mobile network to understand more about how consumers respond to its ads. The experiment was conducted on a sample of active users (as defined by the mobile ad-network). These are users who have exhibited a tendency to spend more time on their mobile apps and enagage with ads a lot more than the median user. Thus, we are working with a selected sample of user. For a randomly chosen mobile app session of these users, EA's ad was the only ad shown in the eigth impression of the session. That is, in the eighth impression of the chosen session, they always saw the same ad by EA. Further, EA (or the ad-network) showed a random set of ads in the previous seven impressions. So the earlier ads could have been EA's ad or one of other advertisers' who advertise through the ad network. Thus, there is variation in distribution of ads shown to the users and this variation is random. From this experiment, EA has data on over 80,000 impressions at the eighth impression of a session.

1. Descriptive analysis: Load the RData files into R

```
load("variety_train.RData")
load("variety_test.RData")
```

(a) What is the observed CTR in the data and the average of the users' past click through rate (ctruser)? Are these numbers as expected? Why or why not?

```
nrow(subset(variety_train,click==1))/nrow(variety_train)

## [1] 0.1134112

mean(variety_train$ctruser)

## [1] 0.1164985
```

The observed CTR in the data is 11.34%, and the average of the users' past click through rate is 11.65%, these numbers are higher than what I expected. They are close so that the data can represent the users' behavior.

(b) Plot the historgrams of in-session variety (variety), and pre-session variety (varietytotal). What do you infer from the plots?

```
hist(variety_train$variety)
```

```
hist(variety_train$varietytotal)
```

The number of in-session variety is less than 10, most of users have seen in-session ads 2-5 times. Before this session, most of users have seen 15-35 ads. These numbers makes sense becuase there seven sessions before this session.

(c) Run a correlation test between the two in-session variables (variety) and (rep)? What do you infer from the sign and the magnitude of the correlation?

```
cor.test(variety_train$variety,variety_train$rep)
```

```
##
##  Pearson's product-moment correlation
##
## data:  variety_train$variety and variety_train$rep
## t = -247.39, df = 63281, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7051143 -0.6971927
## sample estimates:
##        cor
## -0.7011752
```

the correlation between variety and rep is -0.7 which is close to -1. This means that they are highly negatively corrleated. They changed in inverse trends.

(d) Plot the average or mean CTR at each level of in-session variety.Now based on this graph, interpret the relationship between in-session variety and click? Are you more or less likely to click if you have seen a higher variety of ads previously?

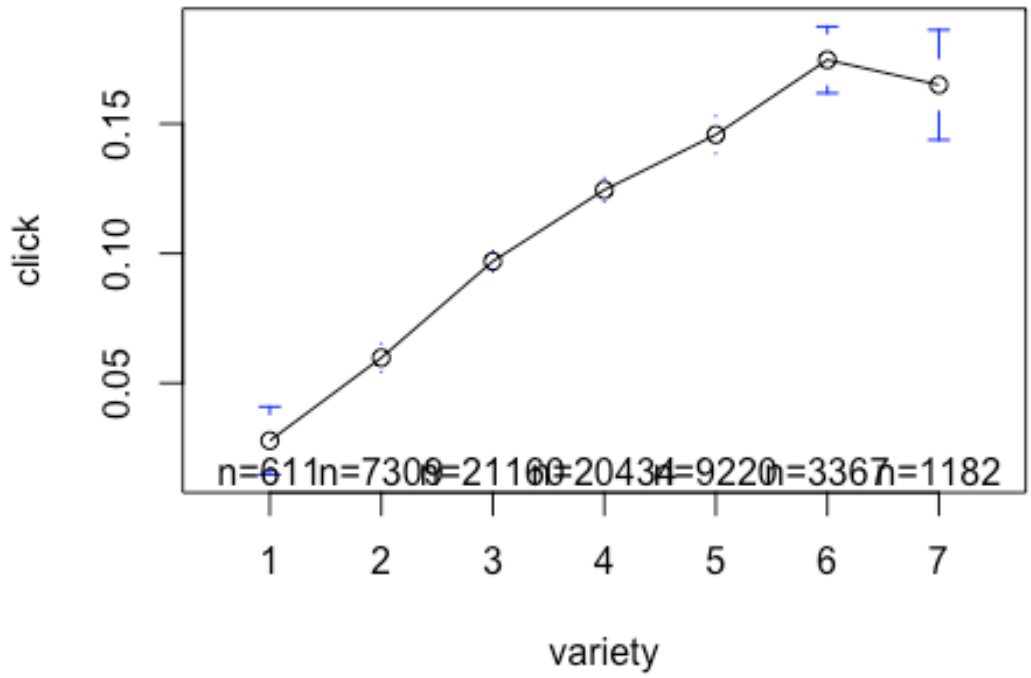
```
library(gplots)
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```
plotmeans(click ~ variety, data = variety_train)
```

```
## Warning in arrows(x, li, x, pmax(y - gap, li), col = barcol, lwd = lwd, :
zero-
## length arrow is of indeterminate angle and so skipped

## Warning in arrows(x, li, x, pmax(y - gap, li), col = barcol, lwd = lwd, :
zero-
## length arrow is of indeterminate angle and so skipped

## Warning in arrows(x, li, x, pmax(y - gap, li), col = barcol, lwd = lwd, :
zero-
## length arrow is of indeterminate angle and so skipped

## Warning in arrows(x, li, x, pmax(y - gap, li), col = barcol, lwd = lwd, :
zero-
## length arrow is of indeterminate angle and so skipped

## Warning in arrows(x, ui, x, pmin(y + gap, ui), col = barcol, lwd = lwd, :
zero-
## length arrow is of indeterminate angle and so skipped

## Warning in arrows(x, ui, x, pmin(y + gap, ui), col = barcol, lwd = lwd, :
zero-
## length arrow is of indeterminate angle and so skipped

## Warning in arrows(x, ui, x, pmin(y + gap, ui), col = barcol, lwd = lwd, :
zero-
## length arrow is of indeterminate angle and so skipped

## Warning in arrows(x, ui, x, pmin(y + gap, ui), col = barcol, lwd = lwd, :
zero-
## length arrow is of indeterminate angle and so skipped
```



Geneally, people are more likely to click if they have seen a higher variety of ads based on this graph.

e)Based on how the experiment was run, do you think this effect is causal? That is, is variety causing the changes in CTR that you see in the graph or is this simply a correlation between CTR and variety?

The effect is causal. The more variety, the less the probability that users remember the ads. Repeating ads will give the users higher extent of impression of ads which lead to click.

2. Within-session level models: Build, visualize, and predict using CART and XGBoost models that take into account only the user's ad exposure earlier within the same session.

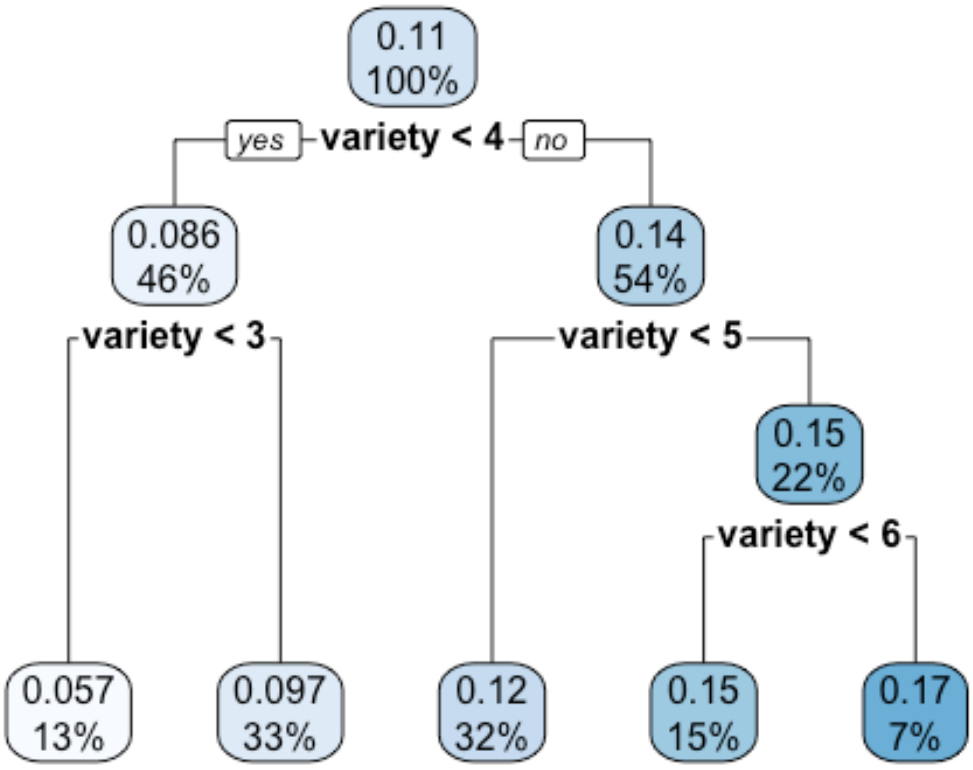
a)Estimate a CART model (to predict click) with the three within-session behavioral history variables on the training data. Use a complexity parameter of 0.00032.

```
install.packages("rpart.plot")

library('rpart')
library('rpart.plot')
```

```
behavioral.model <- click ~ variety+rep+adimpsession

behavioral.tree <- rpart(formula = behavioral.model,
                         data = variety_train, control = rpart.control(cp = 0
.00032))
```

b)Visualize this CART model and give a short overview of your findings. Discuss: how many leaves does this tree have, how does it split, which variables matter, and whether any variables are omitted.

```
rpart.plot(behavioral.tree)
```



There are 5 leaves in this tree. It splits by using only one variable- variety whic means that variety has much more power to predict click than rep and adimpsession. Users with variety >= 3 and variety < 4 have 9.7% probability to click the ads. They account for 33% of our data.

c) Predict on the test dataset with this CART model and store the predictions in a column named 'withinsession.CART.pred'.

```
withinsession.CART.pred <- predict(behavioral.tree, variety_test)
variety_test$withinsession.CART.pred <- withinsession.CART.pred
```

d)Estimate an XGBoost model (to predict click) with the three within-session behavioral history variables using the training dataset. Use the following hyper-parameters: eta = 0.1, max_depth = 6,nround = 100,subsample = 1,colsample_bytree = 1,num_class = 1,min_child_weight = 5, and gamma = 5.

```
library('xgboost')
col.behavioral = c(7,8,9)
xgb.behavioral<- xgboost(data = data.matrix(variety_train[,col.behavioral]),
                  label = variety_train[,1],
                  eta = 0.1,
                  max_depth = 6,
                  nround=100,
                  subsample = 1,
                  colsample_bytree = 1,
                  num_class = 1,
                  min_child_weight = 5,
                  gamma = 5,
                  nthread = 30,
                  eval_metric = "logloss",
                  objective = "binary:logistic",
                  verbose = 0
                  )
```

e)Predict on the test dataset with this XGBoost model and store the predictions in a column named 'presession.xgb.pred'.

```
variety_test$withinsession.xgb.pred <- predict(xgb.behavioral, data.matrix(va
riety_test[,col.behavioral]))
```

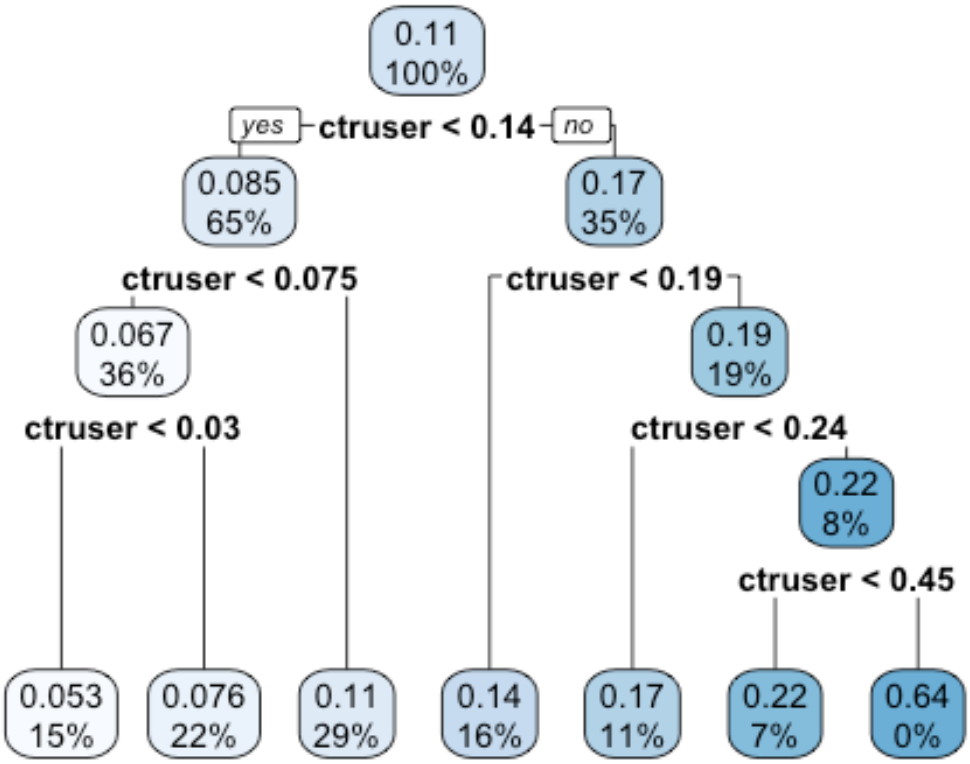
3)Pre-session level models: Build, visualize, and predict using CART and XGBoost models that only consider the user's ad exposure and behavior before the session.

a)Estimate a CART model (to predict click) with the four pre-session behavioral history variables on the training data. Use a complexity parameter of 0.00032.

```
presession.model <- click ~imptotal+ctruser+varietytotal+adimptotal

presession.tree <- rpart(formula = presession.model,
                         data = variety_train, control = rpart.control(cp = 0
.00032))
```

b)Visualize this CART model and give a short overview of your findings. Discuss: how many leaves does this tree have, how does it split, which variables matter, and whether any variables are omitted.

```
rpart.plot(presession.tree)
```



There are seven leaves in this tree. It splits by using only ctruser. The other 3 variables are omitted. Users with ctruser between 0.075 and 0.14 have 0.11 probability to click the ads. They account for 29% of our data.

c)Predict on the test dataset with this CART model and store the predictions in a column named 'presession.CART.pred'

```
pre.CART.prediction <- predict(presession.tree, variety_test)
variety_test$presession.CART.pred <-pre.CART.prediction
```

d)Estimate an XGBoost model (to predict click) with the four pre-session behavioral history variables using the training dataset. Use the following hyper-parameters: eta = 0.1, max_depth = 6,nround = 100,subsample = 1,colsample_bytree = 1,num_class = 1,min_child_weight = 5, and gamma = 5.

```
library('xgboost')
col.pre = c(3,4,5,6)
xgb.pre<- xgboost(data = data.matrix(variety_train[,col.pre]),
                  label = variety_train[,1],
                  eta = 0.1,
                  max_depth = 6,
                  nround=100,
                  subsample = 1,
                  colsample_bytree = 1,
```

```
                    num_class = 1,
                    min_child_weight = 5,
                    gamma = 5,
                    nthread = 30,
                    eval_metric = "logloss",
                    objective = "binary:logistic",
                    verbose = 0
                    )
```

e)Predict on the test dataset with this XGBoost model and store the predictions in a column named 'presession.xgb.pred'.

```
variety_test$presession.xgb.pred <- predict(xgb.pre, data.matrix(variety_test[,col.pre]))
```

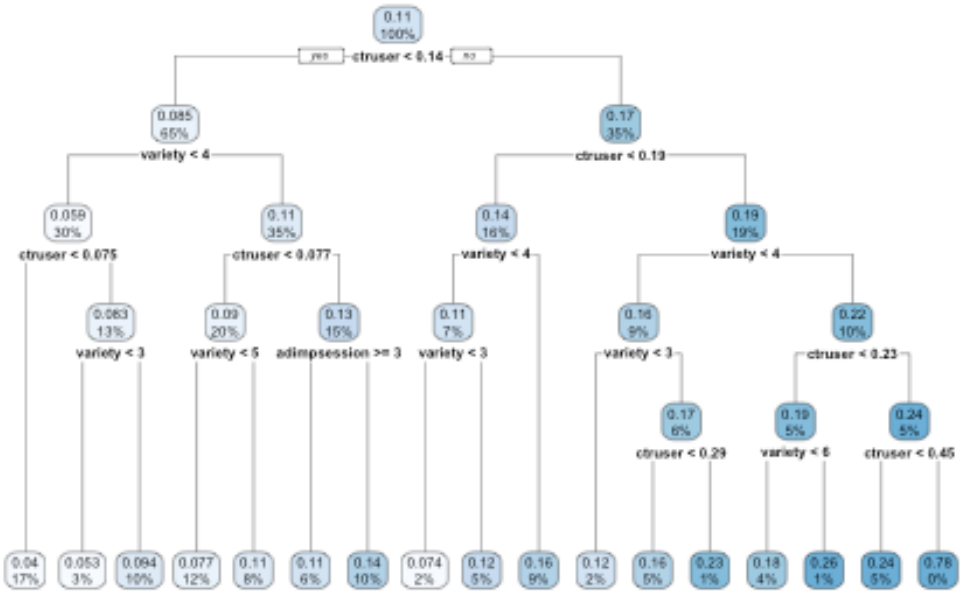
4. Full models: Build, visualize, and predict using CART and XGBoost models that use all the data available for each impression.

a)Estimate a CART model (to predict click) with all the impression-level variables in the training data. Use a complexity parameter of 0.00032.

```
full.model <- click ~imptotal+ctruser+varietytotal+adimptotal+variety+rep+adimpsession

full.tree <- rpart(formula = full.model,
                           data = variety_train, control = rpart.control(cp = 0.00032))
```

b)Visualize this CART model and give a short overview of your findings. Discuss: how many leaves does this tree have, how does it split, which variables matter, and whether any variables are omitted.

```
rpart.plot(full.tree)
```



There are 17 leaves in this tree. It splits using ctruser, variety, and adimpseesion. other variables are omitted due to the power of predicting. One example of explianing the leaves: users with ctruser< <0.075 and variety < 4 account for 17% of our data, and they have 4% probabilty to click the ads.

c)Predict on the test dataset with this CART model and store the predictions in a column named 'full.CART.pred'.

```
full.CART.prediction <- predict(full.tree, variety_test)
variety_test$full.CART.pred <-full.CART.prediction
```

d)Estimate an XGBoost model (to predict click) with all variables using the training dataset. Use the following hyper-parameters: eta = 0.1,max_depth = 4,nround = 100,subsample = 1, colsample_bytree = 1, num_class = 1, min_child_weight = 5, and gamma = 5.

```
col.full = c(2:9)
xgb.full<- xgboost(data = data.matrix(variety_train[,col.full]),
                   label = variety_train[,1],
                   eta = 0.1,
                   max_depth = 6,
                   nround=100,
                   subsample = 1,
                   colsample_bytree = 1,
```

```
                  num_class = 1,
                  min_child_weight = 5,
                  gamma = 5,
                  nthread = 30,
                  eval_metric = "logloss",
                  objective = "binary:logistic",
                  verbose = 0
                  )
```

e)Predict on the test dataset with this XGBoost model and store the predictions in a column named 'full.xgb.pred'.

```
variety_test$full.xgb.pred <- predict(xgb.full, data.matrix(variety_test[,col
.full]))
```

5). Model evaluation: Evaluate the performance of all the six models you ran earlier on AUC and RIG.

(a) First, use Area Under the Curve (AUC) to evaluate the performance of the six models presented above. Present the results in a table. (You do not need to plot the ROC curves for each of the six models.)

```
library('pROC')

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

##
## Attaching package: 'pROC'

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

#auc of withinsession cart
auc.cart.withinsession = roc(variety_test$click, variety_test$withinsession.C
ART.pred)

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

## Setting direction: controls < cases

auc(auc.cart.withinsession)

## Area under the curve: 0.5763

#auc of withinsession xgboost
auc.xgb.withinsession = roc(variety_test$click, variety_test$withinsession.xg
b.pred)

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

```
auc(auc.xgb.withinsession)

## Area under the curve: 0.5834

#auc of presession cart
auc.cart.presession = roc(variety_test$click, variety_test$presession.CART.pr
ed)

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

auc(auc.cart.presession)

## Area under the curve: 0.6385

#auc of presession xgboost
auc.xgb.presession = roc(variety_test$click, variety_test$presession.xgb.pred
)

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

auc(auc.xgb.presession)

## Area under the curve: 0.6425

#auc of full cart
auc.cart.full = roc(variety_test$click, variety_test$full.CART.pred)

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

auc(auc.cart.full)

## Area under the curve: 0.6569

#auc of full xgboost
auc.xgb.full = roc(variety_test$click, variety_test$full.xgb.pred)

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

auc(auc.xgb.full)

## Area under the curve: 0.6624
```

We now tabulate AUC scores for all three models and two optimization methods:

Table of AUC comparisons

	Withinsession	Presession	Full
CART	0.5763	0.6385	0.6569
XGBoost	0.5834	0.6425	0.6624

(b) Next, use Relative Information Gain (RIG) to evaluate the performance of the six models presented above. Present the results in a table.

```
RIG <- function(pred,actual){
  mean.outcome = mean(actual)
  pred = pmin(pmax(pred, 0.0000001), 1-0.0000001)
  llpred = mean(-log(pred)*actual-log(1-pred)*(1-actual))
  llbase = mean(-log(mean.outcome)*actual-log(1-mean.outcome)*(1-actual))
  rig = (1- llpred/llbase)*100
  return(rig)
}
RIG(variety_test$withinsession.CART.pred, variety_test$click)

## [1] 1.21679

RIG(variety_test$withinsession.xgb.pred, variety_test$click)

## [1] 1.3396

RIG(variety_test$presession.CART.pred, variety_test$click)

## [1] 3.47516

RIG(variety_test$presession.xgb.pred, variety_test$click)

## [1] 3.535669

RIG(variety_test$full.CART.pred, variety_test$click)

## [1] 4.452958

RIG(variety_test$full.xgb.pred, variety_test$click)

## [1] 4.927713
```

To summarize the RIG results for all six models, we make the following table:
Table of RIG (in percent) comparisons

+————————+————————+————————+————————+ | | WithinSession | Presession | Full | +==========+==========+==========+==========+ | CART | 1.2168 | 3.4752 | 4.4530 | +————————+————————+————————+————————+ | XGBoost | 1.3396 | 3.5357 | 4.9277 | +————————+————————+————————+————————+ ### (c) Compare the performance of different models and summarize your findings on the relative predictive ability of the six models. What is the best model among these six?

The qualitative results from this table are exactly the same as that from the AUC table. Overall, this suggests that, irrespective of the evaluation metric used, the XGBoost model that uses all the targeting information is the best predictive model. Hence, for all the business purposes and to develop targeting policies, we should use this model.

6. Summarize your findings on the two main substantive questions of interest:

(a) What is the relative value of within-session user history vs. pre-session user history?

From the model evaulation I did in previous questions, the pre-session user history has higher value than the within-session history because the predictive models using pre-session user history has higher performance on predicting users clicks.

(b) What is the effect (positive or negative) of within-session variety on users' ad response?

The within-session variety has positive effect on user's ad response.

7. Business implications: EA now buys all the impressions in the test data. Going forward, EA would like to identify and only buy the top 5000 impressions which yield the highest CTR. To help them with this objective:

(a) Identify the top 5000 of impressions with the highest predicted CTR (based on the best model that you identified in the previous question) and store these impressions in a separate dataframe.

```
top5000=variety_test[order(variety_test$full.xgb.pred,decreasing = T)[1:5000]
,]
```

(b) What is the average CTR for these 5000 impressions? What is the average predicted CTR of these impressions based on your best model. Is your model-predicted average CTR close or similar to the true CTR observed in this subset of the data?

```
mean(top5000$click)

## [1] 0.1914

mean(top5000$full.xgb.pred)

## [1] 0.1905556
```

The average predicted CTR of these impressions is 0.1914, the model-predicted averaged CTR is 0.190. They are similar to each other.

###(c) ROI calculation on test data: Assume that each of these impressions costs EA $0.05 and each click is worth $2. ROI is defined: (Marginal gain - Marketing spend)/Marketing spend.

i. Baseline ROI – First, calculate the Baseline ROI in the sitution where EA buys all the impressions in the test data.

```
(mean(variety_test$click) * 2 - 0.05) / 0.05
```

```
## [1] 3.505333
```

It's 3.50

ii. New ROI – Next, calculate the ROI if EA only buys the top 5000 impressions. How does this ROI compare to the baseline?

```
(mean(top5000$click) * 2 - 0.05)/ 0.05
```

```
## [1] 6.656
```

It's 6.67, this ROI is higher than the baseline.

(d) Assuming that there is another marketing activity (price promotions) which has an ROI of 5. Suppose EA has a total of $1000 to invest in price promotions and advertising. How should EA distribute this money between advertising and price promotions. Specifically, how many of the top impressions should EA buy (consider only multiples of 500, e.g., 500 impressions, 1000 impressions and so on), and what is the revenue and cost of this advertising spend? And how much should EA invest in price promotions?

We want to find the target number when the ROI goes below 5. Which must larger than 5000

```
top8000=variety_test[order(variety_test$full.xgb.pred,decreasing = T)[1:7000]
,]
(mean(top8000$click) * 2 - 0.05)/ 0.05
```

```
## [1] 6.114286
```

```
top15000=top8000=variety_test[order(variety_test$full.xgb.pred,decreasing = T
)[1:15000],]
(mean(top15000$click) * 2 - 0.05)/ 0.05
```

```
## [1] 4.557333
```

```
top12000=top8000=variety_test[order(variety_test$full.xgb.pred,decreasing = T
)[1:12000],]
(mean(top12000$click) * 2 - 0.05)/ 0.05
```

```
## [1] 5.066667
```

we found that when the target number equals to 12000, the ROI will goes below 5. So, it should target 12000 top impressions

12000*0.05=600. So EA should invest 600 on ads and 400 on promotions.