15.071 - Analytics Edge – Final Course Project

Blockchain Digital Asset Price Prediction

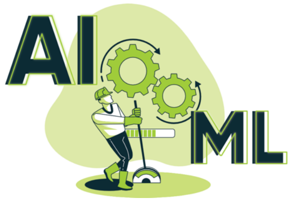
Team Members: Kazuaki Takeda, Yoshihide Iijima and Amiya Ranjan

Project Background

Problem

Digital assets are volatile, and their price movements are affected by variety of internal and external signals.

Solution & Models



Our model will predict the price of NFT(or any digital asset on a blockchain) using available predictors. We will choose amongst Linear Regression, Non-Linear Regression, CART, Boosting & XGBoost methods to select the best model.

Business Value

May be used by individuals, companies, and institutions who invest in NFTs or any other digital assets. Will help to maximize return on investment.

Dataset

Project Scope/Data Collected: We have downloaded NFT historical sales data from Kaggle (https://www.kaggle.com/datasets/francescofalleni/nft-historical-sales). We used this dataset of more than 100K rows to build our understanding of predictors for Digital Assets and to gain knowledge of trends in NFT sales. Further, we downloaded a smaller dataset which has the relevant predictors and price information. Using this dataset, we have built our model to predict NFT prices. This dataset has 4983 observations with 4 columns.

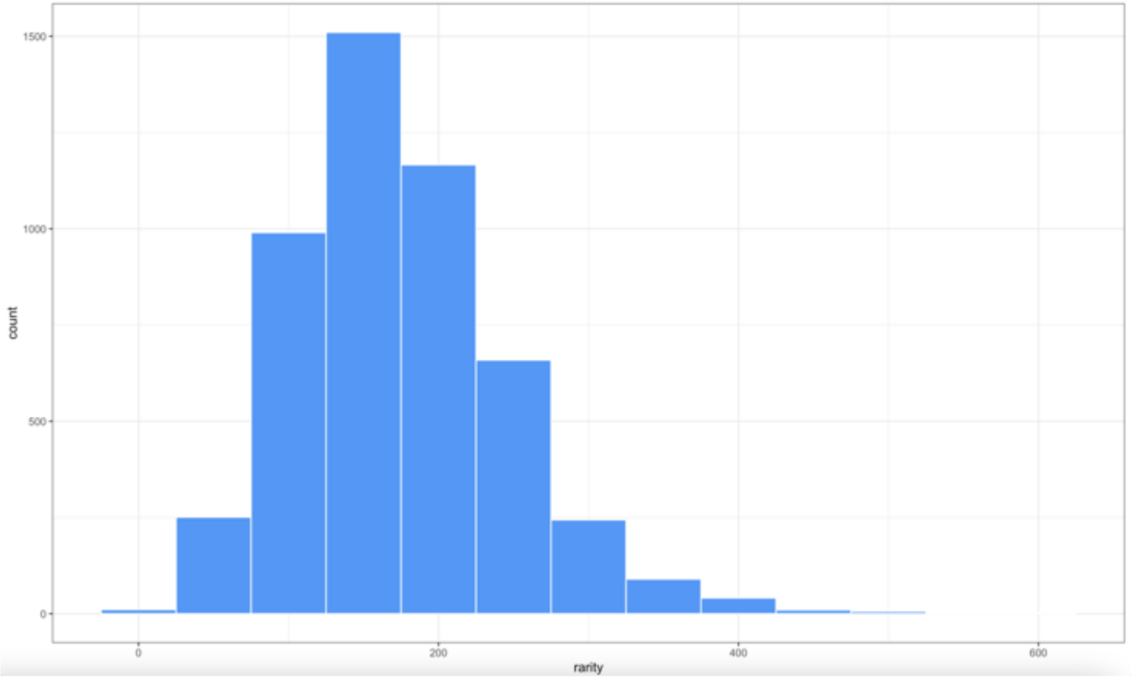
```
> str(nftnewData)
'data.frame':	4982 obs. of  5 variables:
 $ rarity         : num  57 244 187 184 186 ...
 $ last_sale_price: num  3.88 2.8 0.4 4.5 2.99 1.7 4.5 2.25 9 2.8 ...
 $ sale_count     : num  6 5 3 1 2 3 2 2 3 1 ...
 $ predicted_price: num  3.04 3.46 2.6 2.73 2.91 ...
 $ rarity_class   : chr  "LOW" "MEDIUM" "MEDIUM" "MEDIUM" ...
> qqplot(nftnewData
```

Our dataset

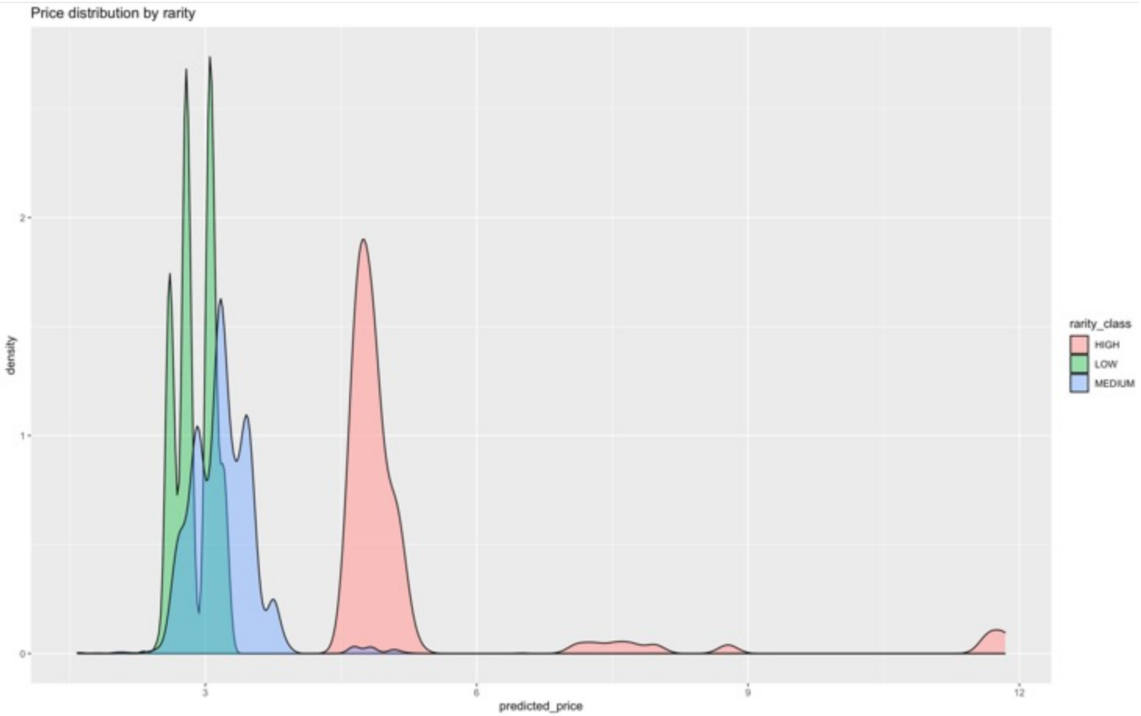
```
> str(nftSales)
'data.frame':	108147 obs. of  16 variables:
 $ X                     : int  0 1 2 3 4 5 6 7 8 9 ...
 $ collection_name       : chr  "Rarible" "Rarebit Bunnies" "Rarible" "Rarible" ...
 $ asset_id              : int  18214580 18276844 16911700 16986936 13382164 17408060 17139942 18197702 18050357 1675
1423 ...
 $ asset_name            : chr  "Daft Punk Never Die" "Rarebit #164 - Wax Off Bunny" "Meditation" "I´m OG" ...
 $ asset_contract_date   : chr  "5/27/20" "1/21/21" "5/27/20" "5/27/20" ...
 $ asset_contract_time   : chr  "16:53" "20:43" "16:53" "16:53" ...
 $ event_date            : chr  "2021-02-27" "2021-02-27" "2021-02-27" "2021-02-27" ...
 $ event_time            : chr  "23:59" "23:58" "23:58" "23:58" ...
 $ asset_age_time_of_sale: int  276 37 276 276 468 276 276 87 87 185 ...
 $ asset_age_tos_yrs     : num  0.76 0.1 0.76 0.76 1.28 0.76 0.76 0.24 0.24 0.51 ...
 $ event_auction_type    : chr  "dutch" "dutch" "dutch" "dutch" ...
 $ event_quantity        : num  1 1 1 1 1 1 1 1 1 1 ...
 $ event_payment_symbol  : chr  "ETH" "ETH" "ETH" "ETH" ...
 $ event_total_price     : num  0.07 0.15 0.001 0.000647 0.2 0.02 0.06 0.1 0.19 0.014 ...
 $ Conversion            : num  1200 1200 1200 1200 1200 1200 1200 1200 1200 1200 ...
 $ asset_unit_price_usd  : num  84 180 1.2 0.78 240 24 72 120 228 16.8 ...
```

Larger dataset

Basic visualization of dataset

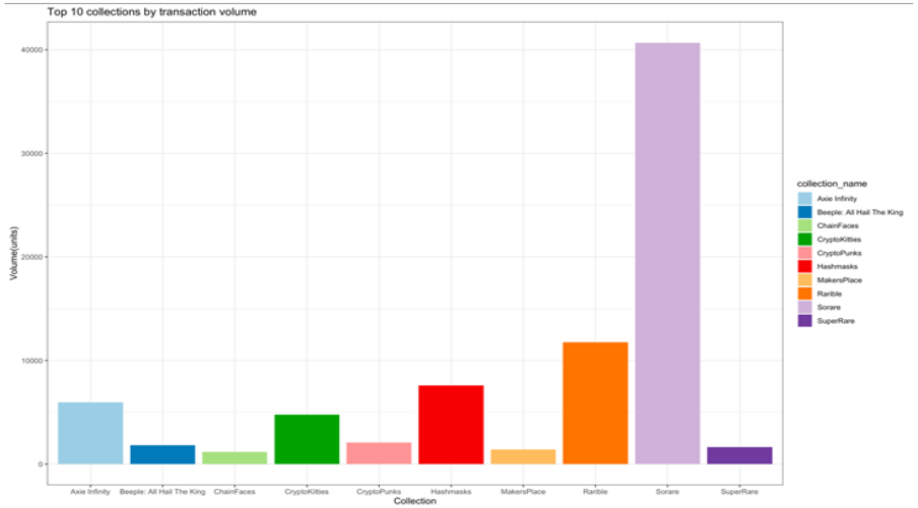


Histogram (Rarity)

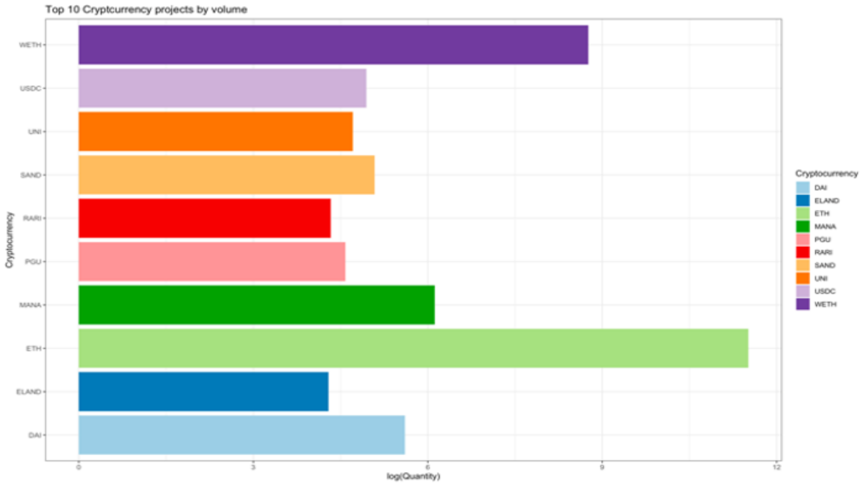


Grouped Kernel Density Plot (Rarity)

Other visualizations on larger dataset

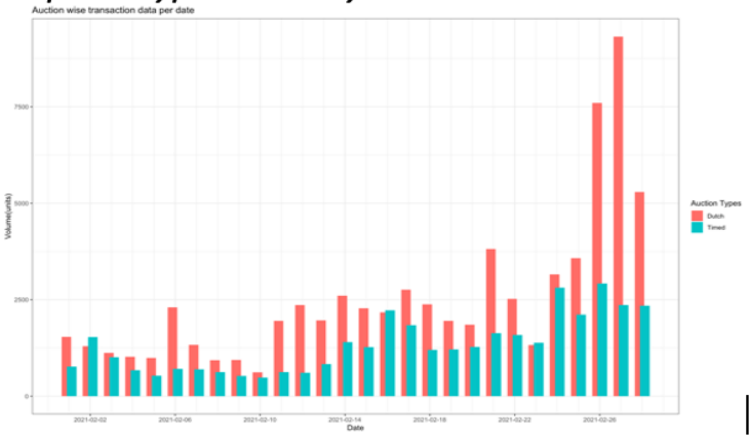


Top 10 collection



Top 10 cryptocurrency

BTC/ETH/Market price movement



Auction types

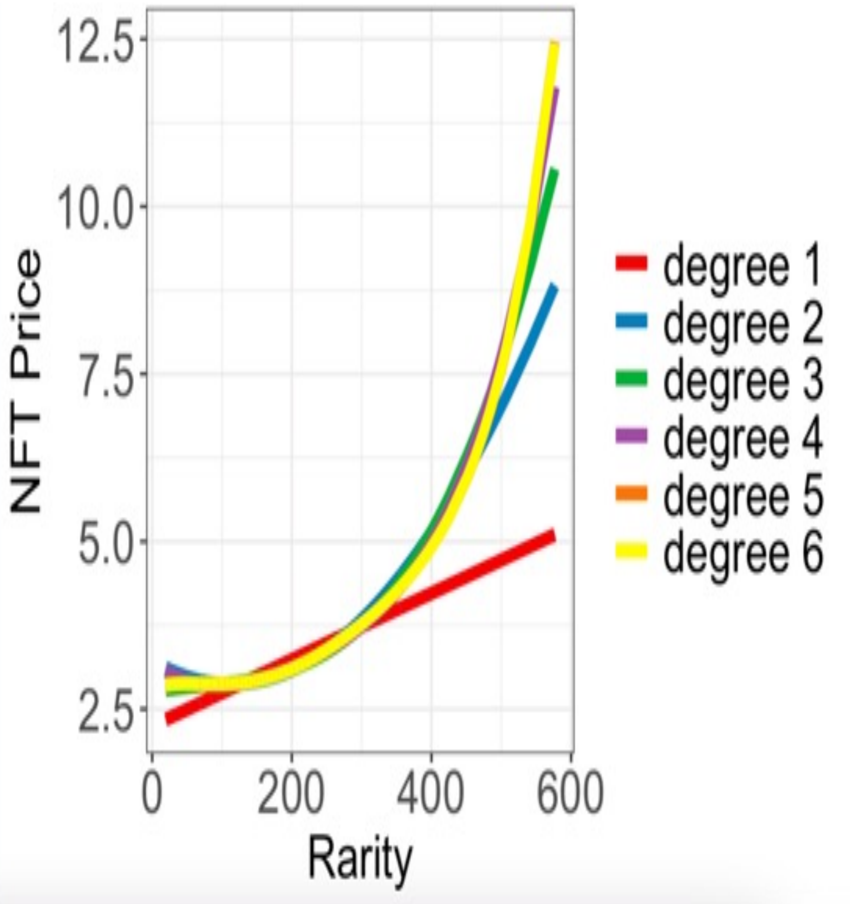
Analytical Methods: Linear Regression

```
Call:
lm(formula = predicted_price ~ ., data = train)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.3635 -0.1291 -0.0261  0.0870  5.9132 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)    1.889e+00  1.559e-02  121.13   <2e-16 ***
rarity         4.983e-03  7.263e-05   68.61   <2e-16 ***
last_sale_price 2.417e-02  1.293e-03   18.69   <2e-16 ***
sale_count     6.975e-02  2.104e-03   33.16   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2781 on 3483 degrees of freedom
Multiple R-squared:  0.6719,	Adjusted R-squared:  0.6716 
F-statistic:  2377 on 3 and 3483 DF,  p-value: < 2.2e-16

> preds <- predict(pricePredictionModel, newdata=test)
> summary(preds)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  2.164   2.832   3.075   3.106   3.348   6.650 
> lrosr2<-calc.OSR2(test$predicted_price, preds, mean(train$predicted_price))
> lrosr2
[1] 0.6129668
```

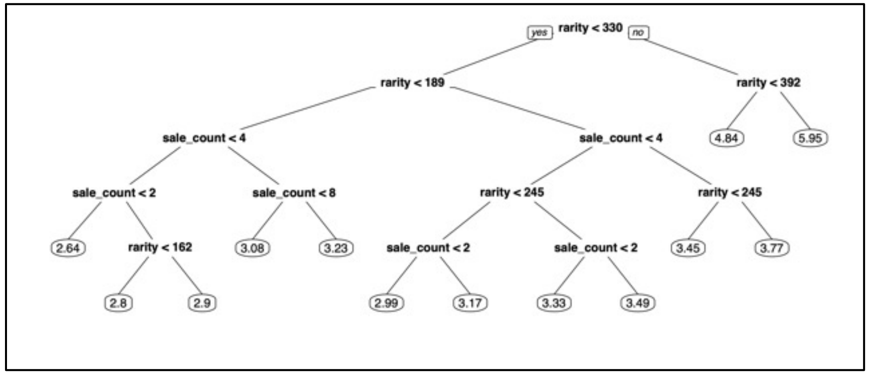
Observations:
All variables are significant
Signs of coefficients make sense
R2 is not very low but can be improved

Analytical Methods: Polynomial Regression

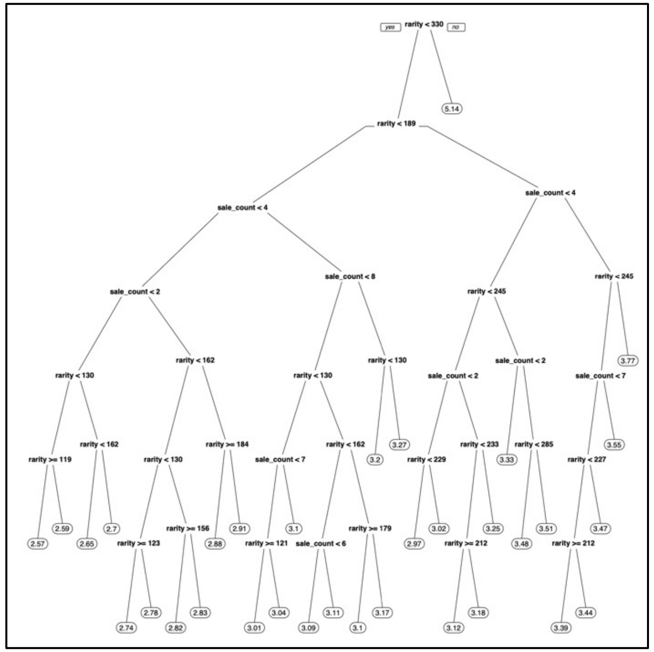
```
R 4.2.1 · ~/Documents/MIT-Home/Fall/Analytics Edge/R CodeBase/Project_AE/
#################################Poly on Rarity#########################################
degrees <- 1:10
n <- length(degrees)
summary.model.fit <- train
R2.all <- rep(NA,n)
OSR2.all <- rep(NA,n)
for (i in 1:n){
  degree <- degrees[i]
  model=lm(predicted_price~poly(rarity, degree),data=train)
  summary.model.fit[[paste0('poly', degree)]] <- predict(model,train)
  R2.all[i] <- calc.OSR2(train$predicted_price, predict(model,train), mean(train$predicted_price))
  OSR2.all[i] <- calc.OSR2(test$predicted_price, predict(model,test), mean(train$predicted_price))
}
R2.all
[1] 0.5065580 0.7205213 0.7392244 0.7468654 0.7495095 0.7495143 0.7505930 0.7571224 0.7707473 0.7712486
OSR2.all
[1] 0.4613435 0.7123235 0.7556615 0.7710305 0.7725822 0.7726318 0.7688556 0.7819189 0.7949796 0.7958758
ggplot(data=summary.model.fit,aes(x=rarity)) +
  geom_line(aes(y=poly1,col='1'),lwd=2) +
  geom_line(aes(y=poly2,col='2'),lwd=2) +
```



Analytical Methods: CART



Initial tree



Final Tree

Analytical Methods: Boosting & XGBoost

```
> train.boost$bestTune
    n.trees interaction.depth shrinkage n.minobsinnode
315   35000                12     0.001             10
>
```

```
> params.winner = list(max_depth = 9, eta = .001, subsample = .5, min_child_weight = 1, gamma = 0, colsample_bytree=
1, alpha=.5, lambda = 0)
> round.winner = 14993
> winner = which.min(RMSE.cv)
> params.winner = list(max_depth = 9, eta = .001, subsample = .5, min_child_weight = 1, gamma = 0, colsample_bytree=
1, alpha=.5, lambda = 0)
> round.winner = 14993
> mod.xgboost <- xgboost(data = as.matrix(x.train), label = y.train, params = params.winner, nrounds = round.winner,
 verbose = F)
> pred.xgboost <- predict(mod.xgboost, newdata=as.matrix(x.test), rounds=rounds.winner)
> SST = sum((y.test - mean(y.train))^2)
> OSR2.xgboost <- 1 - sum((pred.xgboost - y.test)^2)/SST
> OSR2.xgboost
[1] 0.9421044
> #[1] 0.5324156
> RMSE.xgboost <- sqrt(mean((pred.xgboost - y.test)^2))
> RMSE.xgboost
[1] 0.1275812
> #[1] 0.05225879
> MAE.xgboost <- mean(abs(pred.xgboost-y.test))
> MAE.xgboost
[1] 0.01594445
```

Results & Recommendations

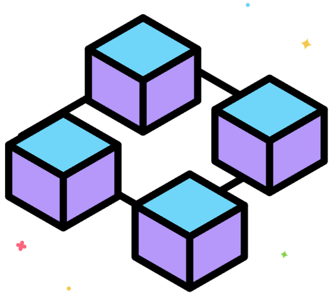
S.No.	Analytical Model	R2	OSR2
1	Linear Regression	.6718	.6129
2	Polynomial Regression(Rarity degree = 6)	.749	.772
3	CART (cp .001)	.817	.703
4	CART(cp .00001)	.822	.706
5	XGBoost		.938

1. For Rarity analysis, maybe we can use models that filter visual traits and assign ranks to NFTs.
2. Can help if historic data is available for longer duration.

3. We may have to give comparison data of predicted vs actual to show how our model is better in terms of business value.

Thank You!!

Problem



Project Idea: Meet Bill. A 20-year old MIT CS undergrad who keeps himself updated about latest technological trends. For last 2 years Bill has been investing and following something called crypto (broadly known as digital assets and may include cryptocurrency coins and Non-fungible Tokens (NFTs)). These assets whose transactions are recorded on a permanent, immutable ledger called blockchain (public blockchain such as Ethereum) are volatile and their price movements are affected by variety of internal and external signals. For Bill to make profits on his investment, he needs to have sound information on which assets are going to appreciate.

For our project, we decided to build a model to help Bill (and many others like him). Our model will predict the price of NFT(or any digital asset on a blockchain) using available information. Currently, there are not many credible sources which can reliably predict the price for a NFT or Digital Asset. As transactions in Digital Assets may increase in next few years, building a model that can help in predicting price for these assets can be helpful. We hope to eventually be able to provide a credible, independent, 3rd party rating for each NFT collection which can then be used by individuals, companies, and institutions to invest in NFTs or any other digital assets.