Predicting First Day Returns for Japanese IPOs

Executive Summary

Goal: To predict the First Day returns on Japanese IPOs (based on first day closing price), using public information available prior to the offer

Purpose: To use the model to predict whether a new IPO coming on the market will make first day gains or not, and use the result to decide whether to invest in the IPO.

Type of Problem: Predictive

Data Gathering

We used a publically available data-set on Japanese IPO data from 1997-2009, from Kaneko and Pettway's Japanese IPO Database (KP-JIPO) : http://www.fbc.keio.ac.jp/~kaneko/KP-JIPO/top.htm

This dataset has 1561 records, for all IPOs in Japan from 1997 to 2009.

Data Cleaning

The raw data had multiple issues…

1) The "date" was not in a format recognized by excel, so we had to convert all the dates to excel-recognizable format.
2) Records from 1997-1999 were missing information about the Lead Manager's fees and Percentage of allocation to Lead Manager, which we considered to be important predictors. This led to removal of the 128 records within this time period.
3) Industry column had 2 spelling mistakes, which were removed.

Data Preparation

We created some columns (combining some of the columns from the data) which we thought would be a better predictor for first day returns…

1) Minimum bid size – The minimum bid a retail investor has to make to participate in the IPO. The rationale for using this was that retail investors would be more averse to investing in IPOs if the minimum upfront commitment required was high.
2) Secondary Offering %age – The percentage of secondary shares being offered in the IPO, i.e. the number of shares of existing shareholders that were exiting the company via the IPO. The rationale behind this variable was that IPOs in which all shares are new issues give all the money to the company, whereas if current shareholders are seen to be using the IPO as an exit route, then the public may see that as a negative signal about the quality of the IPO.

We also created binned variables from categorical data…

1) Industry – Of the 33 industries in the cleaned data, we binned them 4 categories of industries, representing the 4 types of patterns in the first day returns observed as a function of industry.
2) BRLMs – Out of 56 Lead Managers in the data, we binned them into 3 categories of Lead Managers, depending on their effect on the first day return.
3) Market – There were 11 types of markets over which the IPOs were listed, however we differentiated only between OTC (over the counter) and exchange traded IPOs, i.e. a dummy variable of whether the issue was OTC or not.

Post these steps, we partitioned the data into the standard 50-30-20 split for running prediction models.

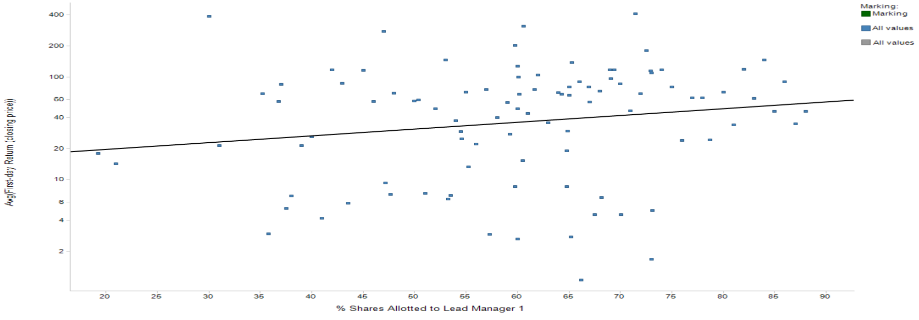
Predictors Used:

After data preparation and some exploration of the data, we narrowed down on the following 9 predictors:

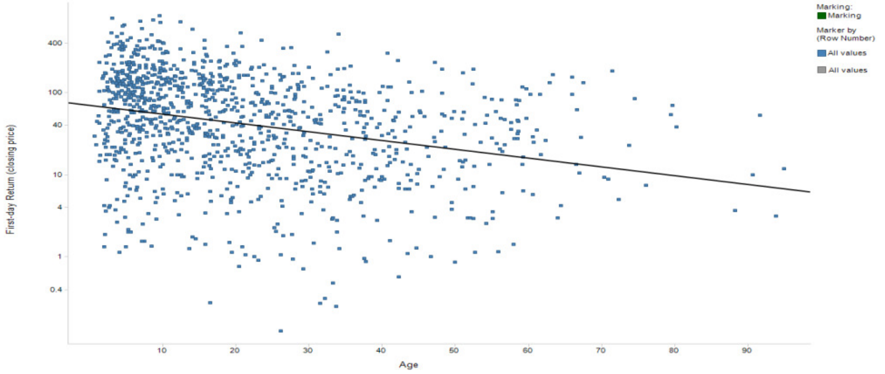
1) Age of company at time of IPO
2) Gross Proceeds (size of IPO)
3) Minimum Bid Amount
4) Underwriter’s Gross Spread (fees as %age of size of IPO)
5) Percentage shares allocated to Lead Manager 1
6) Secondary offering as %age of total
7) IS_OTC listing
8) Industry_Type (binned categorical variable – 4 categories)
9) Lead_Manager (binned categorical variable – 3 categories)

Exploration

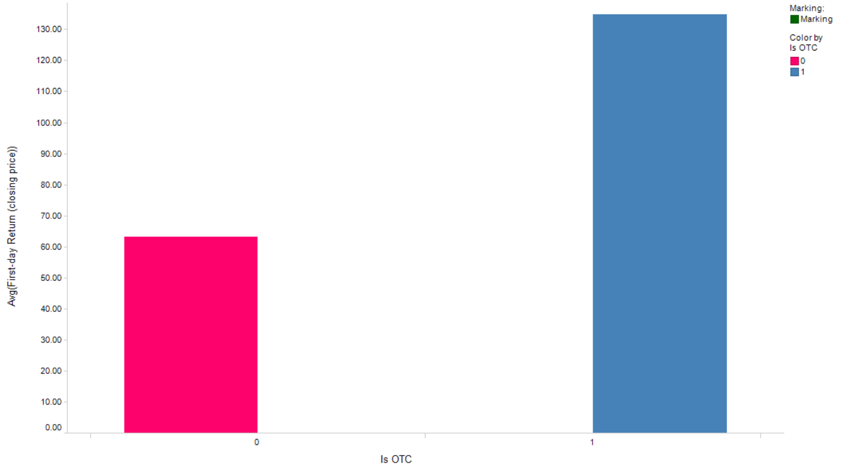
1) One of the things we noticed was that the y-variable, the first day IPO returns, was clustered around a small area with decreasing density away from the cluster. This led us to think that the y-variable may be better represented using a logarithmic relationship. So we converted our y-variable to log(y) and used that for all predictions.
2) We noticed that the first day return was increasing when the allocation to the lead manager was higher, which is somewhat intuitive as well that the lead manager would have an incentive to underprice the issue and ensure greater chance of full subscription if he had to bear a larger in case the issue wasn’t fully subscribed.



3) The underpricing of the IPO decreased further if the company was mature at the time of IPO, i.e. younger companies were much more likely to be underpriced than mature ones.



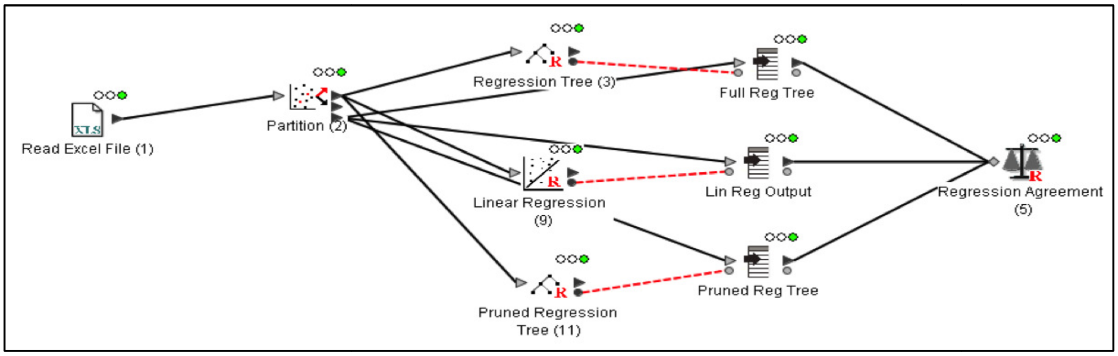
4) Our newly created variable, IS_OTC, was also helpful in assessing underpricing…



Analysis

We used the Multi-Linear Regression, K-Nearest Neighbors and Regression Tree algorithms to attempt to predict the first day return (on log scale). We note that using the Naïve rule, the average of first day returns was 67%, with a standard error of 106%.

We used Spotfire Miner for Liner Regression, Regression Trees (full and pruned) and XLMiner for K-Nearest Neighbours. The Spotfire Miner setup for our procedure is shown below.



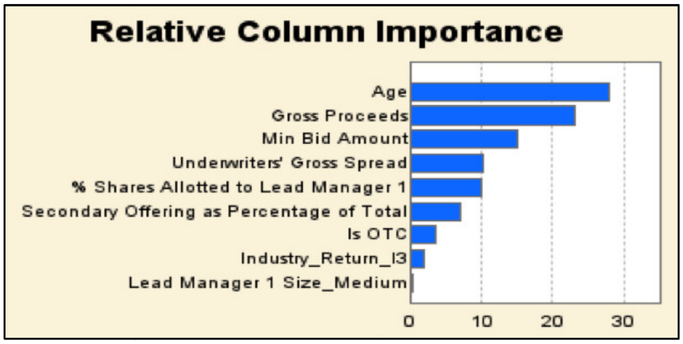
The plots for the residuals using the different methods are presented as (***Exhibit 1)*** at the end of the report.

1) Multiple Linear Regression: Multiple linear regression gave poor results, with high RMSE of 110% and Mean absolute error of 67%

Multiple Linear Regression	
Mean Squared Error	12095.23
RMSE	109.9783
Mean Absolute Error	67.27547

2) Full Regression Tree: (***Exhibit 2)***

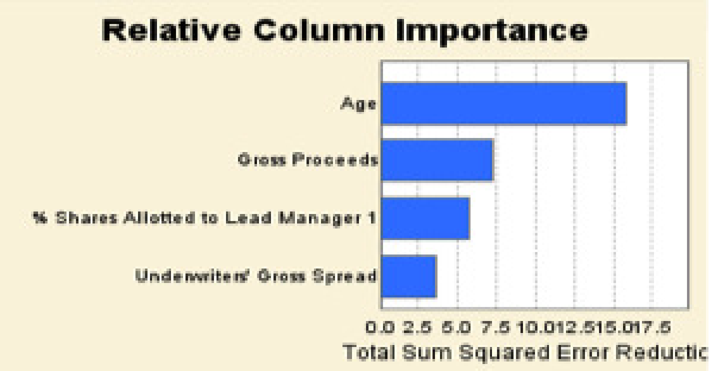
The relative importance of the predictors was as follows…

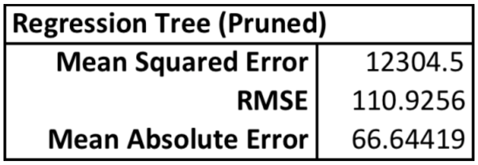


Age, Gross proceeds (size of issue) and Minimum bid amount ranked as the top 3 factors affecting first day IPO returns. The Regression tree however, did not improve much on the prediction.

Regression Tree (Full)	
Mean Squared Error	14813.77
RMSE	121.7118
Mean Absolute Error	79.19805

3) Regression: Pruned Tree

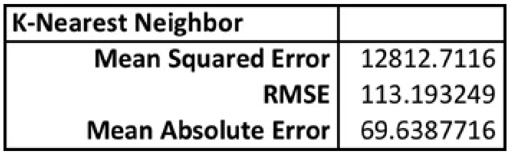




Regression Tree (Pruned)	
Mean Squared Error	12304.5
RMSE	110.9256
Mean Absolute Error	66.64419

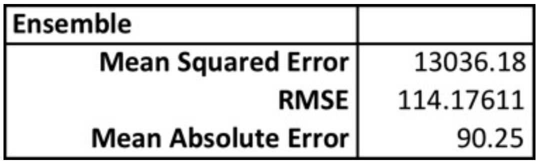
The pruned tree gave marginally better results than the full tree.

4) K-Nearest Neighbours: We ran KNN using a k of 5, since when we allowed k to vary to large numbers the algorithm kept on choosing the largest number possible (went upto 20, the max that XLMiner can handle), with marginal improvements in error rates with each incremental increase in k ***(Exhibit 4)***. So we decided to choose the k at a point from where onwards the improvements seemed to taper off.



K-Nearest Neighbor	
Mean Squared Error	12812.7116
RMSE	113.193249
Mean Absolute Error	69.6387716

5) Ensemble : We took the average prediction of the above 4 methods to predict the value for a new record, however the results from the ensemble are not significantly better than any of our individual methods. However the dispersion of the residuals ***(Exhibit 1)*** seems to be lesser than any of the methods individually.



Ensemble	
Mean Squared Error	13036.18
RMSE	114.17611
Mean Absolute Error	90.25

Conclusion

On average the prediction algorithms do not seem to do significantly better than the naïve rule, with standard errors bordering near 100%. With an average expected return of 67%, and standard error of about 100% is not good enough to use for investment purposes. Ideally we would want the *mean – 1 standard deviation* of the first day returns to be higher than 0 for us to have any confidence in the model.

Submitted by:

Vivek Kumar	**61210021**
Rohan Mahadar	**61210055**
Gaurav Jain	**61210585**
Tejas Pahlajani	**61210606**

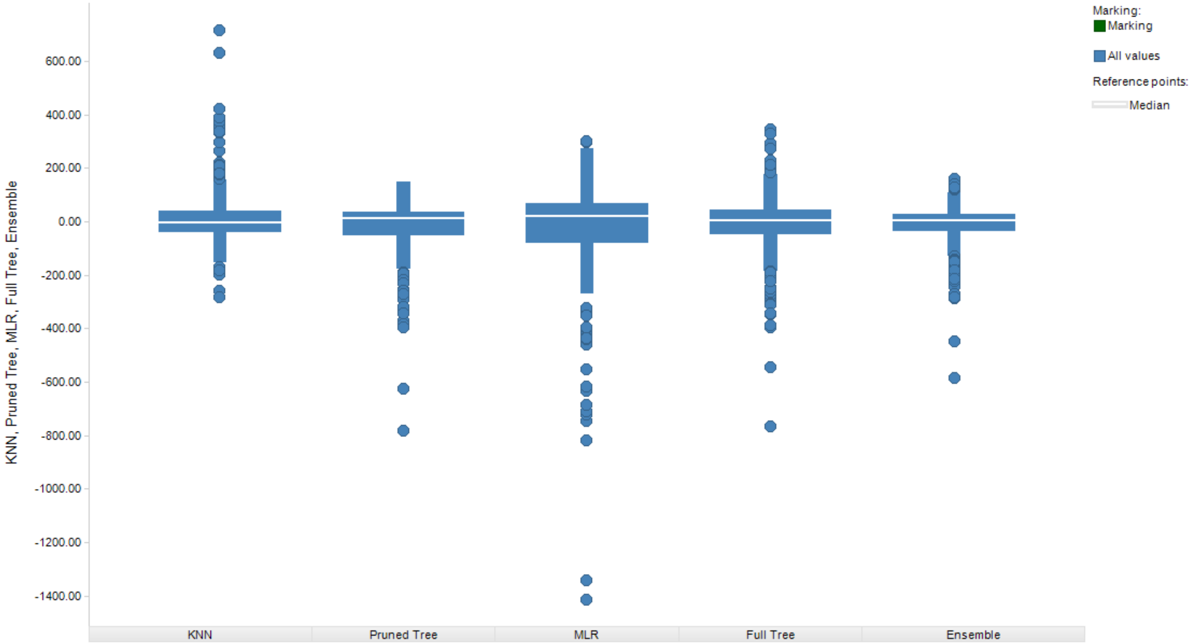


Exhibit 1: Residuals using the different methods

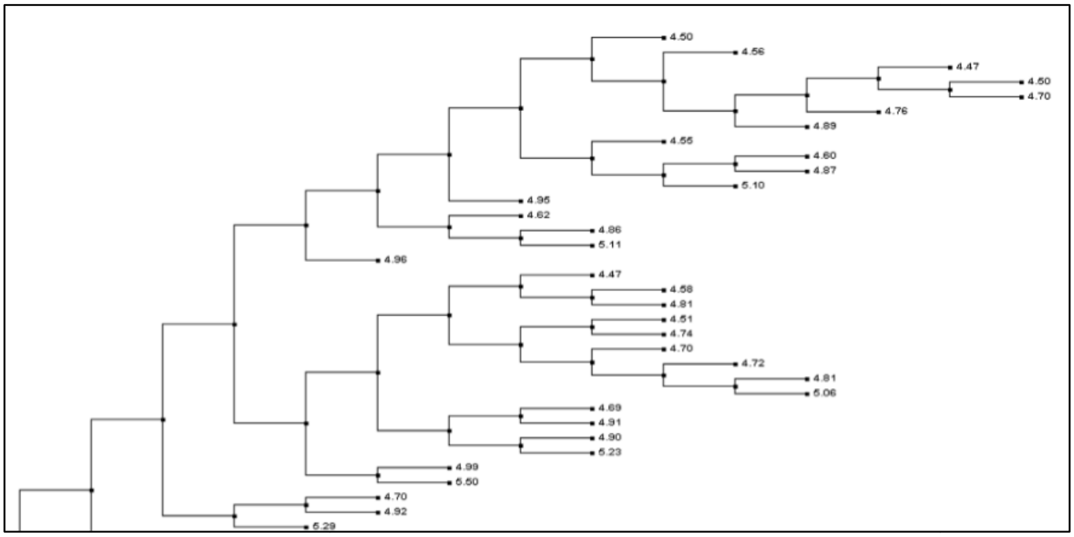


Exhibit 2: Full Regression Tree

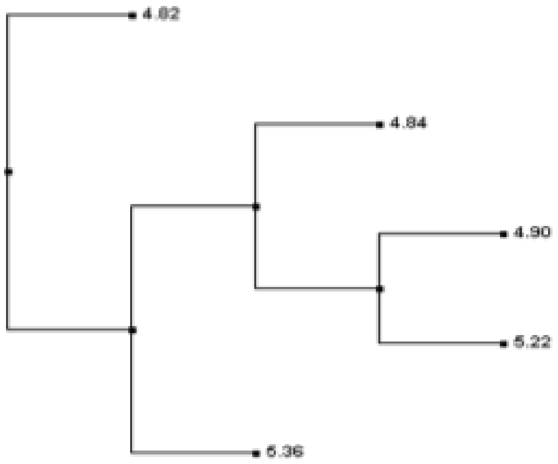


Exhibit 3: Pruned Tree

Value of k	Training RMS Error	Validation RMS Error	
1	0	0.722727291	
2	0	0.650958645	
3	0	0.622172251	
4	0	0.617716972	
5	0	0.612118456	
6	0	0.608404501	
7	0	0.607777356	
8	0	0.606935697	
9	0	0.605849004	
10	0	0.604014927	
11	0	0.603725082	
12	0	0.603623558	
13	0	0.602380018	
14	0	0.602627935	
15	0	0.601277304	
16	0	0.600809471	
17	0	0.601014934	
18	0	0.600638226	
19	0	0.600669147	
20	0	0.600285097	<--- Best k

Exhibit 4: Choice of (k) for KNN