AsiaYo!

體驗亞洲，住宿悠遊

Identifying bookings with a high-risk of host rejection to improve customer service and customer satisfaction

Arturo Heyner Cano Bejar, Tonny Kuo, Nick Danks, Kellan Nguyen
Team 1

Executive Summary

AsiaYo! operates in a highly competitive industry serving as intermediary and agent between accommodation providers (hosts) and accommodation seekers (guests). High customer service is a critical business goal for AsiaYo!. Rejection of bookings by hosts can cause customer dissatisfaction and potentially loss of customers. Currently, AsiaYo! experiences a host rejection rate of 15% of orders. Rejection of a booking by the host triggers a reaction from the AsiaYo! customer service team in which they contact the guest, offer alternatives and provide solutions to resolve the rejection and convert it into an alternative booking. This is a re-active process and only occurs after some time when the host has rejected the booking.

The goal of this analysis is to provide a tool for the AsiaYo! customer service team to proactively intervene on bookings with a high risk of rejection at the time of the booking. This will be achieved by providing a probability of rejection for every booking made. High rejection risk bookings can then trigger a customer service team intervention allowing for fast, efficient, and proactive customer service response.

The data used in the predictive task is generated from the booking transaction and includes the number of guests, the number of nights, the number of rooms, the amount paid, the day of week for check-in date, the day of week for booking date, and the days-in-advance period from booking date to check-in date.

The key challenge of this analysis was the correct prediction of rejected orders – that is predicting a transaction as highly likely of rejection and it indeed then resulted in rejection. To this ended we used several classification algorithms and statistical analysis to arrive to a best possible recommendation. Results from our analysis are promising. However, their interpretation must be handled with care. Our performance metrics, sensitivity (0.69) and accuracy (0.53), lead use to best results when using the Naïve Bayes. Recommendations are to implement the predictive algorithm right after a booking is done, update the algorithm periodically to compensate for time, and input more variables that potentially lead to better performance such as behavioral data.

Problem description

Business goal:

AsiaYo!'s business model allows high flexibility to hosts when deciding whether a booking is accepted or rejected. Consequently, one of the biggest challenges is reducing the booking rejection rate – a task which is generally performed by the customer service team who will respond to a rejected booking by contacting the customer and resolving the problem. Being informed of bookings with high risk of rejection, the customer service team can intervene early and proactively, and direct their efforts at the high-risk bookings.

Analytics goal:

The team's analytics objective is to classify the probability of a transaction being rejected by the host, with the classification occurring at the time of booking.

Data description

The data was provided by AsiaYo! and the main dataset is historical booking transaction data in which there are more than 65,000 rows and 24 columns originally. Each row is one transaction made, and the unit of analysis of this project.

	X	order_num	user_id	host_id	accom_id	room_id	guests	nights	ack_status	check_in	check_out	rooms	amount_paid	paid_status	order_status	created_at	payment	acquiring_bank	platform
1	44	2.01409e+11	1541	416	411	1392	1	1	4	9/17/2014	9/18/2014	1	3800	1	N	9/9/2014 13:06	NA	NA	pc
2	45	2.01409e+11	1547	426	420	1432	2	2	5	12/31/2014	1/2/2015	1	3400	1	N	9/9/2014 21:23	NA	NA	pc
3	46	2.01409e+11	1553	329	335	1136	4	1	4	9/22/2014	9/23/2014	1	3200	1	N	9/10/2014 12:08	NA	NA	pc
4	47	2.01409e+11	1556	80	106	331	2	1	4	10/7/2014	10/8/2014	1	1380	1	N	9/10/2014 13:57	NA	NA	pc
5	48	2.01409e+11	1572	426	420	1432	2	1	5	9/13/2014	9/14/2014	1	1700	1	N	9/11/2014 15:24	NA	NA	pc

Figure 1: *AsiaYo! Raw Data Example.*

Brief data preparation details

Initially, we had to perform some data cleaning activities. We found that the database contained many illegal rows – primarily due to testing – and these had to be removed. Additionally, several input variables contained over 90% NA values and were not informative to prediction and therefore were dropped. The outcome variable was not explicitly provided by the data, however the variable "ack_status" provided us with the raw material with which to derive the outcome. This categorical variable had 3 values (4,5 or 9). We determined that value 5 and 9 related to a rejected booking while 4 related to an accepted booking – we thus derived the outcome variable is.rejected from this raw variable.

We conducted exploratory data analysis by comparing correlation tables (see Appendix Figure 1), considering which of the data fields were legal and available at the time of the booking and contained valid predictive power. Subsequently, we found 7 useful predictors, namely, the number of guests (guests), the number of nights (nights), the number of rooms (rooms), the amount paid (amount_paid), the day of week for check-in date (DOW_ci), the day of week for booking date (DOW_created_at), and the advance period from booking date to check-in date (advancebook). Note that DOW_ci, DOW_created_at, and advancebook are derived from variables in the original data, but are easily generated and pose no implementation challenge.

	guests	nights	rooms	amount_paid	DOW.ci	DOW.created.at	advancebook	is.rejected
1	1	1	1	3800	Wednesday	Tuesday	8	0
2	2	2	1	3400	Wednesday	Tuesday	113	1
3	4	1	1	3200	Monday	Wednesday	12	0
4	2	1	1	1380	Tuesday	Wednesday	27	0
5	2	1	1	1700	Saturday	Thursday	2	1
6	2	1	1	2800	Saturday	Friday	1	1
7	4	1	1	2680	Tuesday	Friday	18	0
8	1	1	1	1980	Wednesday	Sunday	108	1
9	4	1	1	4340	Friday	Sunday	26	1
10	2	1	1	1380	Saturday	Tuesday	4	0

Figure 2: *AsiaYo! Clean Data Example.*

Datamining solution:

In the initial data exploration phase, many datamining models were considered and applied on this data, such as naïve classification, Naïve Bayes, KNN, Trees, Discriminant Analysis and Logistic Regressions in order to compare their predictive performance. Given that the outcome of interest, being rejected bookings (is.rejected = 1), is relatively rare at only 15% of the cases we also conducted all the above models on both the raw data and over-sampled data where the training set was resampled to contain 50% rejected bookings and 50% accepted bookings.

To evaluate the model, we first established our key metrics with which to measure performance. The key goal of this analysis is the successful classification of rejected bookings. We will thus use a classification matrix. We are primarily interested in True Positive classifications and so our primary interest is sensitivity of the classification matrix. However, a large proportion of False Positives will also lead to increased workload on the customer service team and thus we balance our evaluation with consideration of the overall accuracy.

It was not easy to set a benchmark with which to compare predictive power of the various models as the naïve classification completely disregarded the classification of interest (is.rejected = 1) and provided sensitivity of 0%. As such we will directly compare the performance of the models and select the model with the highest sensitivity and accuracy.

Table 1: *Performance across all models considered with and without oversampling.*

Non-oversampling							
Methods	**Sensitivity**	**Accuracy**	**Specificity**	**TP**	**FP**	**TN**	**FN**
Logistics	0.00	0.85	1.00	4	1833	10013	4
KNN	0.00	0.85	1.00	0	1837	10017	0
Naive Bayes	0.03	0.84	0.99	60	1777	9899	118
Discriminant Analysis	0.00	0.84	1.00	2	1835	10013	4
SVM	0.00	0.85	1.00	0	1837	10017	0
Classification Tree	0.01	0.83	0.98	15	1822	9865	152
Random Forest	0.13	0.84	0.97	231	1606	9711	306
Boosted Tree	0.20	0.81	0.92	364	1473	9227	790
Oversampling							
Methods	**Sensitivity**	**Accuracy**	**Specificity**	**TP**	**FP**	**TN**	**FN**
Logistics	0.65	0.55	0.53	1705	923	8023	7129
KNN	0.00	0.85	1.00	0	2628	15152	0
Naive Bayes	0.69	0.53	0.51	1801	827	7694	7458
Discriminant Analysis	0.65	0.52	0.50	1721	907	7603	7549
SVM	0.58	0.63	0.64	1530	1098	9633	5519
Classification Tree	0.39	0.57	0.60	1034	1594	9084	6068
Random Forest	0.63	0.62	0.61	1664	964	9317	5835
Boosted Tree	0.60	0.60	0.60	1581	1047	9096	6056

Conclusions

The present project aimed at classifying future customer bookings into high and low probability of rejection by the host, therefore, performance metrics such as sensitivity and accuracy were compared.

Generally, non-sampling models show higher accuracy rate while oversampled models reached slightly higher than 50% accuracy rate. Higher sensitivity is present in oversampled models; therefore, oversampled models were chosen. Specifically, the Naïve Bayes method happened to be the most adequate. In sum, classifying new bookings using the Naïve Bayes algorithm gives the most suitable results. That said, the accuracy rate needs to be interpreted carefully.

Advantages and Limitations

By classifying bookings "after booking" as high or low propensity of rejection AsiaYo! could get deeper insights on customers since more data input is expected. The Naïve Bayes model is a simple algorithm to implement allowing for addition of several many other predictors without much increase in computational power. However, one of the challenges leading our model to such performance is the lack of behavioral data, which in our opinion could leverage the results.

Operational Recommendations

Some recommendations as follows:

1.- We suggest to use the Naïve Bayes model to classify bookings as potential rejections.

2.- The implementation of the algorithm is designed to be done right after booking occurs.

3.- Other variables could also be included such as month of booking, month of check in an so since these variables are more general in nature and may potentially reveal patterns.

4. Properly coded behavioral data might also increase performance if handled with consideration of the privacy issues.

5.- We suggest to update the algorithm constantly to account for possible changes in performance.

Appendix

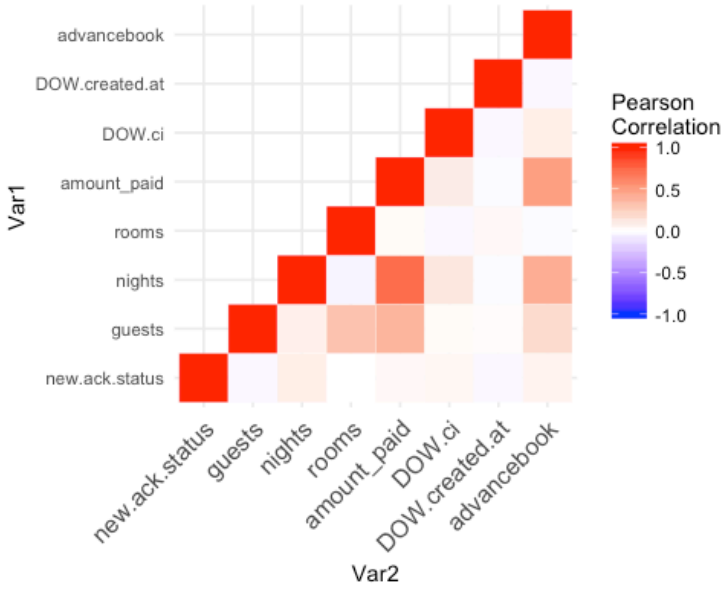

Figure 1: Correlation table of variables