Progress Report
"Bet on Better Forecasts"
Leveraging Customer-Level Insights for Enhanced Financial Prediction for Racing and Wagering Western Australia

Reiko Takizuka, Duc Nguyen Hong, Chandana Mudunuri, Dayoon Sug, Michael Sullivan

Abstract

Customer Value is a key metric for businesses to gain insights into the effectiveness of their products, services and overall customer experience. Predicting customer value helps businesses estimate the total revenue they can expect from a customer. It allows for prioritization of valuable customer segments, tailor marketing efforts, and allocating resources effectively. This project is predicting the Lifetime Value of the TAB's individual wagering customers using machine learning techniques.

Company background

Racing and Wagering Western Australia (RWWA) is the controlling authority for horse and greyhound in the state of Western Australia. It is a governmental body charged with maintaining the long-term sustainability of the racing industry in the region.

Part of the RWWA's remit is to control off-course betting activities for the racing industry, which they do through the Western Australia TAB, a state-owned company that runs over 300 betting retail outlets and an online betting platform known as "TABtouch". WA TAB allows customers to wager on racing (horse or greyhound) and other sports (both Australian and international). Each year customers place bets with WA TAB of around AUS $2bn, and WA TAB pays out around AUS $1.7bn to winning bets. Though large, these amounts make WA TAB a relatively small player in the context of the gambling industry.

TABtouch logo

Problem Statement

TABTouch's current approach to predicting future turnover relies primarily on time series models built on aggregated data. While these models can be effective, we believe they might not capture all dynamics of TABTouch's customer behavior, nor do they allow for detailed exploration of evolving patterns among different customer segments. We believe that leveraging customer-level transaction data could allow us to provide more accurate cash flow predictions for the company and better understand which customer segments are driving business value.

Business Justification and Objective

The data pattern allows accurate turnover forecasting for the whole business using time series model, which is fantastic for financial planning and budgeting or risk management. However, we recognize the potential benefits of customer-level prediction that can be

achieved through different supervised machine learning models (since we are not able to apply the same time series model type to predict future customer's spending as they are much more inconsistent compared to the aggregated turnover). We will attempt both approaches in this project but with our individual prediction models as our primary objective.

In the context of this project, we define Customer Value as the total turnover spent by each individual customer. Predicting Customer Value will allow TABTouch to:

- Focus on retaining the customers with the highest long-term value.
- Make better decisions around marketing spending.
- Improve segmentation of customers based on long-term values.
- Improve financial forecasting by understanding the value of existing Customers.
- Help the company with development of new wagering products.

Data Source

Our primary data source is (anonymized) internal wagering data from TAB: we have data that aggregates daily betting totals for each individual customer across 2021 and 2022.

Each row of this dataset represents the bets made by one customer on a given day. There is a unique customer identifying number, customer attributes and then a breakdown of the customers betting pattern for that day. Because of the sheer number of bets made on the platform, we are using data that has been aggregated daily rather than including one separate row for each bet made by each customer.

Column	Description
DATE_DIM	Date of the transaction
DAY_OF_WEEK	Day name of the week of the transaction
BET_ACCOUNT_NUM_HASH	Customer unique identifier
AGE	Customer's age as of Wager date
AGE_BAND	Customer's age band as of Wager date
GENDER	Customer's gender (M, F, U)
TENURE_IN_DAYS	Number of days since Customer opened account as of Wager date
RESIDENTIAL_STATE	Residential state where the customer resides
FOB_RACING_TURNOVER	Total Bet amount spent on Fixed-odds Racing events
FOB_SPORT_TURNOVER	Total Bet amount spent on Fixed-odds Sports events
PARI_RACING_TURNOVER	Total Bet amount spent on Pari-mutuel Racing betting
PARI_SPORT_TURNOVER	Total Bet amount spent on Pari-mutuel Sports betting
TOTAL_TURNOVER	Total Bet Amount spent on the day
DIVIDENDS_PAID	Total Dividend Amount won by the customer
GROSS_MARGIN	Gross Margin for the Wagering provider (Turnover – Dividends - Costs)
TICKETS	Total tickets bought by the customer on the day

Methodologies

We shall follow a typical process pipeline of a data science project, including Data Cleaning -> Data Exploration -> Data Modeling -> Model evaluation.

Data Cleaning

Given that the data was sourced directly from the company, minimal data cleaning was required. Nevertheless, various measures were taken to prepare the data for analysis:

- *Missing data*: There were NAs in 5 columns: AGE and 4 columns related to TURNOVER. The presence of null values in those 4 columns signifies zero transactions. We decided to impute AGE column by filling in the average age of the customers which is 44.

  Missing value on each column

```
DATE_DIM                       0
BET_ACCOUNT_NUM_HASH           0
AGE                         2669
AGE_BAND                       0
GENDER                         0
TENURE_IN_DAYS                 0
RESIDENTIAL_STATE              0
FOB_RACING_TURNOVER      3773146
FOB_SPORT_TURNOVER      10549527
PARI_RACING_TURNOVER     3800973
PARI_SPORT_TURNOVER     12305088
TOTAL_TURNOVER                 0
DIVIDENDS_PAID                 0
GROSS_MARGIN                   0
TICKETS                        0
```

- *Redundant columns and duplicate records*: DAY_OF_WEEK can be derived from DIM_DATE column, same with AGE and AGE_BAND. These columns are created for the purpose of reporting; therefore, they can be removed. There are no duplicate records in the datasets.
- *Transactions with TOTAL_TURNOVER equaling 0*: These transactions were complemented to other transactions in some regards. They can be deleted to increase prediction model accuracy.
-

Data Exploration

Regarding our primary research question, we emphasize on daily total turnover value and individual customer's turnover analysis.

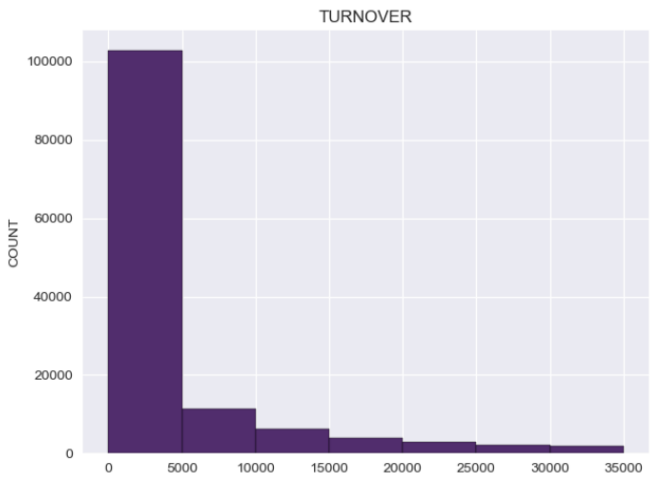
(1) Daily total turnover value (for time-series model):

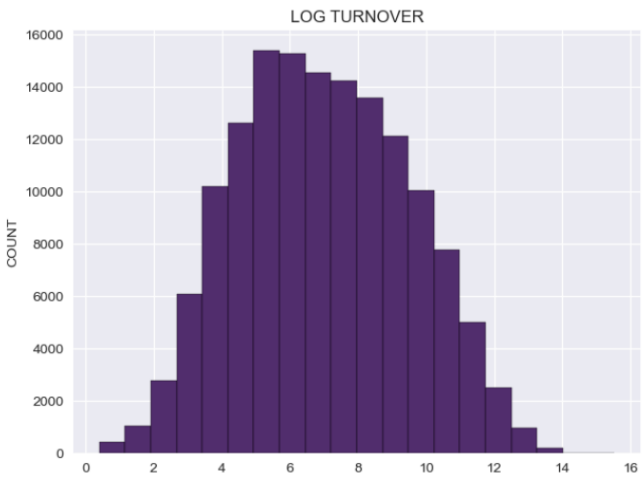
- *Seasonality:* The data demonstrate seasonality, with the wagering product showing variation throughout the week and differing slightly from month to month. The highest conversion rates happen on Saturdays and dip at the beginning of the week. This pattern stays consistent throughout both years.
- *Trend*: From the 7-day moving average line, there is no significant upward or downward trend throughout the year, except for two contextual outlier points.
- *Outliers*: There are two noteworthy outliers:
    - Melbourne Cup Day: an enormous racing event that takes place on the first Tuesday of November every year, garners significant attention. It attracts a substantial number of customers to place a wager on the event.
    - Christmas Day: Wagering options are limited on this day. Therefore, a significant drop is expected.

From recognizing the weekly pattern, an Exponential smoothing model can be applied at date granularity. However, because of the nature of the model, the further from the prediction is, the less accurate the result gets. Therefore, we decided to forecast 4 weeks from the prediction point.

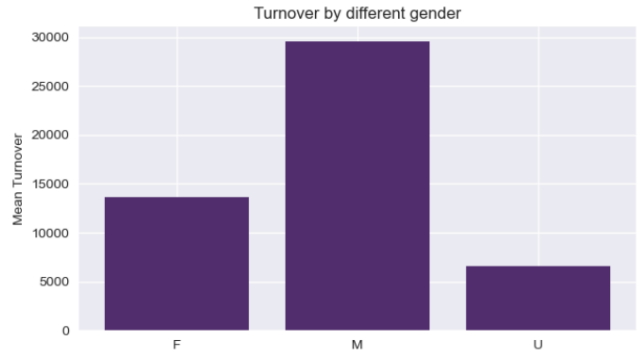
(2) Individual customer's turnover analysis:

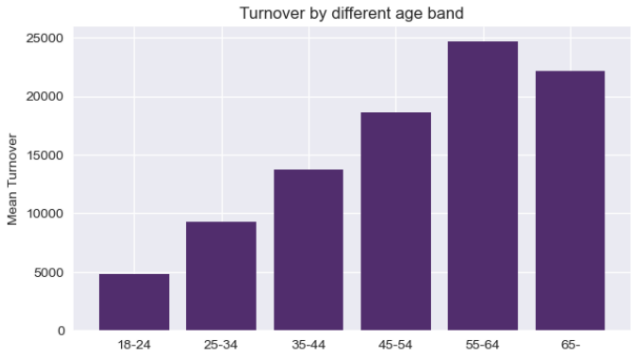
- *High variance*: In two years, 90% of customers spent under $5000 wagering. We spotted an exponential decay in the distribution. This could greatly affect the accuracy when it comes to large values. A log transformation could normalize the distribution.





- *Categorical factors*: Demographic factors can be significant to the prediction models for each categorical group has a different level of turnover.





- *Correlated numerical factors*: Since we are going to build regression model, it is important to collect linear correlated factors to use. Here is the correlation heat map:

Modeling Processes

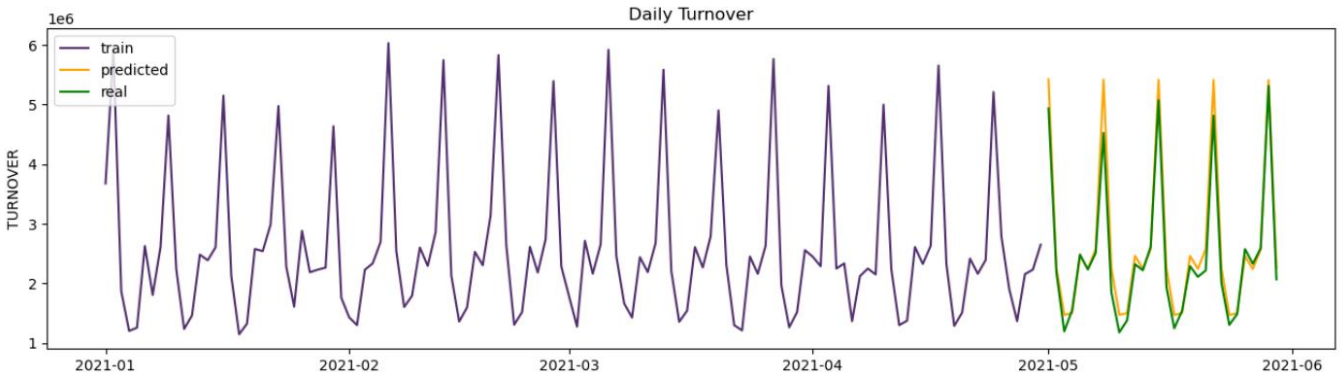
We plan to implement these model types to explore our research questions:

- Traditional Time Series Method: Exponential smoothing model.
- Customer-level prediction models: Regression models (linear & non-linear), Regression tree, Random Forest, K-nearest neighbor, Probabilistic Customer Lifetime Value (CLV).

Traditional Time Series Method: Exponential smoothing

Background: Exponential smoothing is a widely used forecasting technique that provides a flexible and effective approach for predicting future values based on historical data. It is particularly valuable in situations where there is a need to forecast time series data characterized by trends, seasonal patterns, and random fluctuations. The exponential smoothing model is based on the principle of assigning exponentially decreasing weights to past observations, with more recent data points given higher weights. This allows the model to adapt and respond quickly to changes in the underlying patterns of the data. We will use this model to predict the overall turnover of the next 28 days (about 4 weeks). Exponential smoothing algorithm can be implemented through statsmodels package in Python.

Modeling so far: We set the seasonal period to be 7 (as in 7 days in a week), additive trend factor and multiplicative seasonal factor. We used the estimated initialization method. Even though it required at least 4 weeks of data, we used five-month worth of data to train the model. ETSModel function from statsmodels package can return optimized smoothing factor, trend factor and seasonal factor.



Challenges:

- Contextual outliers: As mentioned in the exploration data analysis, there are two major outliers that are the Melbourne Cup final on the first Tuesday of November and Christmas day. The model needs customization to fit these outliers.
- Difficulty in detecting trend: Since the total turnover for a week stays consistent throughout the year, it's difficult to sense any upward or downward trend that could occur in the future.

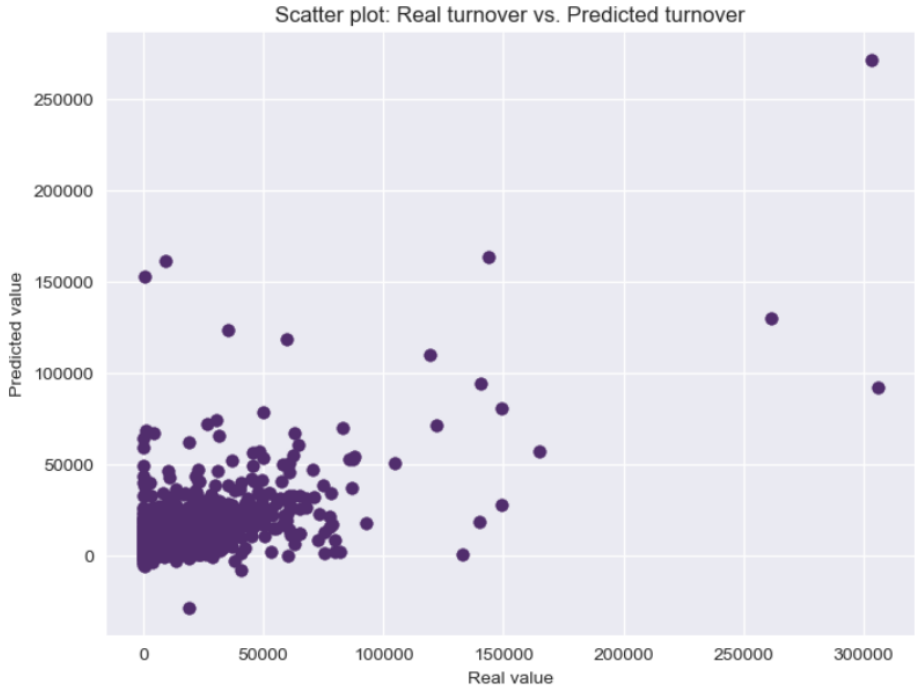
Next steps: Customize the model to factor in the outlier data. At the same time, tune the hyperparameter to deliver closer estimates.

Customer-level prediction models

(1) Regression models

Background: Regression models are powerful statistical techniques used for predictive analysis. They are widely used for prediction because they offer several advantages. First, they provide a systematic and quantitative approach to understanding and analyzing complex relationships between variables. Second, regression models can be used to make accurate predictions, allowing us to forecast future outcomes based on historical data and patterns. Third, regression models enable us to identify the most significant factors that influence the dependent variable, helping us gain insights into the underlying drivers of the phenomenon we are studying. We will use the regression model to predict a unique customer's total turnover for the next 28 days (about 4 weeks). Regression models can be implemented via statsmodels package in Python.

Modeling so far: We applied a simple linear regression model. We set the customers' total turnover of 28 days as the dependent variable. We also created new factors using historical data including total turnover, win-loss ratio, wagering frequency of the last 7 days, 28 days, 84 days along with the original demographic features as independent variables. Here is the result:



Challenges:

- Heteroskedasticity: the variance gets bigger as the value gets bigger. We can try different non-linear regression models to lessen this effect such as log-log model.
- Feature selection: This step needs to be applied in order to negate multicollinearity and insignificant variables. We can apply stepwise algorithm to implement this.
- Time series nature: Even though the weekly pattern stays consistent throughout the year, there are still seasonal trends that need to be considered in the model.

Next steps: Try other non linear regression models; optimize feature selection and feature engineering, tune the hyperparameter to deliver closer estimates.

(2) Probabilistic CLV Modelling

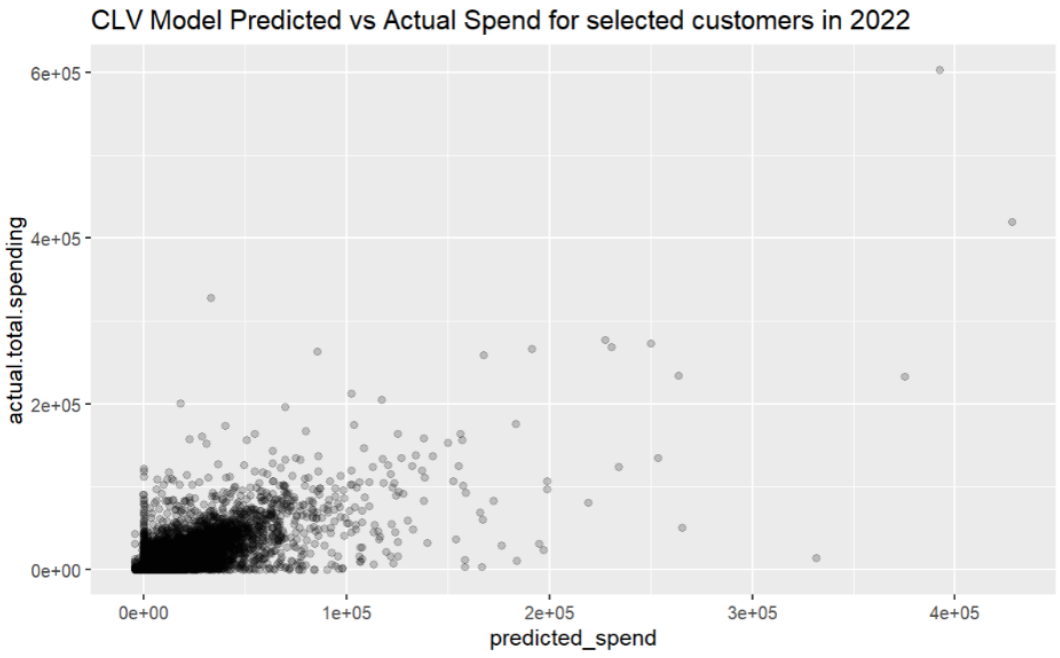
Background: Customer Lifetime Value (or CLV) is an estimate of the value that a customer will generate for the company across all their future interactions. Factors impacting CLV include[1] how much a customer spends and how frequently they purchase. CLV models can be very simple or use complex statistical techniques. Probabilistic CLV models use data on past transactions to assess whether a customer is still alive and predict their future transactions (and spend). We chose to use a Probabilistic CLV technique which combines a Pareto/NBD model (which estimates future transaction frequency) with a Gamma-Gamma model (which estimates future transaction value). We chose this method because it is relatively simpler to implement than some other probabilistic models (such as the Beta-Geometric/NBD). Specialized R packages allow for much easier implementation of these

[1] https://online.wharton.upenn.edu/blog/why-customer-lifetime-value-matters/

techniques: we chose to use the package CLVTools because of its ease of use and because it allows for the introduction of covariates (which other packages like BTYD do not).

Modeling so far: We have already implemented a basic version of our Pareto/NBD and Gamma-Gamma model. The approximate steps we took to create this model with[2] are as follows. We took our cleaned customer transactions dataset and converted this into a custom data object that can work with the CLVTools package (as part of this we need to set the 'time' unit and split the data into train/vs holdout periods – we set the time to weeks and held out half the data). We then took this custom data object and created our Pareto/NBD model using CLVTools inbuilt pnbd() function. Finally, we took our trained model and used it to predict future transactions (frequency and projected spend) for our holdout period. The CLVTools package uses a Gamma-Gamma model to help with the prediction of future transaction value.

These modeling steps allow us to predict the frequency of transactions and predicted mean spending at the individual customer level for every single customer. This gives a very detailed dataset which we can either: aggregate to make predictions about segments or all customers; use to make observations about individual customers. The chart below shows a plot of each customer's actual spend during 2022 (the holdout period) vs the customer's predicted spend from our model (which was calculated by timing the predicted frequency of transactions with mean spend per transaction). Please note that due to some model limitations, this model and the graph represent a subset of transactions/customers (see the section "Challenges" for details of exclusions).



[2] A fuller walkthrough of the package can be found here: https://cran.r-project.org/web/packages/CLVTools/vignettes/CLVTools.pdf

Challenges: The nature of the Pareto/NBD has led to some challenges which we are discussing how to tackle currently.

- Negative Transactions: Pareto/BND with Gamma-Gamma models are not able to model transactions with a negative value (which wouldn't occur in most situations but can with gambling data). In our exploratory modeling phase, we excluded all transactions with negative values – so lost a lot of data. To fix this we will explore predicting "turnover" instead of "margin"; and try splitting out the negative transactions, reversing their sign and modeling them separately.
- Holdout Period Only Customers: Pareto/NBD models can only make predictions for customers within the training data period. In our case (when we split our data in half) we had to exclude customers who only bet in 2022. We will try to decrease the size of the holdout period so we can train with more customers; but once this model is in production this is less of an issue as ultimately, we will want to train on all data available anyway (and then predict the unknown future).

Next Steps: The next steps in our probabilistic CLV modeling will be to fix the "negative transactions" issue and then begin implementing some covariates to see if we can improve the accuracy of our predictions (CLVTools allows us to implement 'static' covariates to account for customer features and 'time variable' covariates to account for seasonality). Once we are getting better customer-level predictions from our model, we will begin applying them to our business questions in various ways.

Project Next Steps

- Tune our models by trying different combinations of hyperparameters; try solving existing problems with the models.
- Implement other prediction models aforementioned; compare the result and see which one is the most accurate.
- Work on the final document and video presentation.
- Finalize the codes on GitHub repository.
