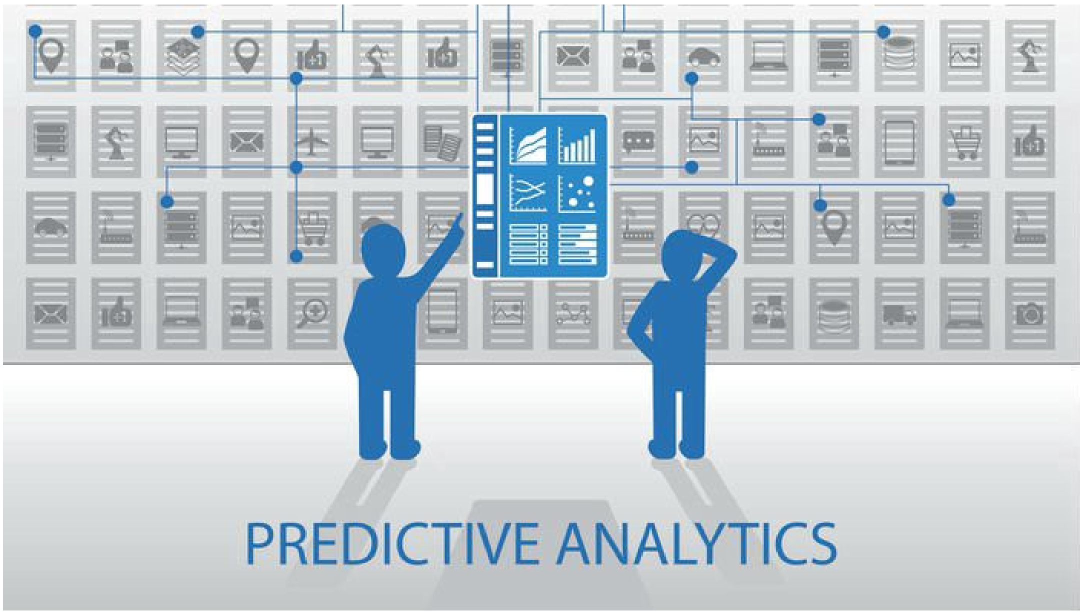
Customer Lifetime Value

Ryan Aminollahi
Feb 10, 2018 · 10 min read

Predictive Customer Analytics — part III



In business there are some golden sayings that everyone needs to appreciate and adopt.

> *Eighty percent of our business comes from 20% of our customers.*
>
> *It costs 10 times less to sell to an existing customer than to find a new customer.*

Different customers generate different levels of sales for a business. We need to identify and nurture these top customers to ensure a steady stream of revenue. But the main question is,

- How indicative of future revenue is the past revenue?
- Is our best customer in the past going to be our best customer in the future?
- Is this outcome in our hands?

We want to identify these customers who bring us significant future business and nurture them, and ensure that they are happy and satisfied with us.

We need a way to compute a customer's lifetime value. So, let's get back to our sample use case. Customer just bought a laptop from us. So, what kind of future revenues can we expect from him? Well, he just bought a laptop just now, so he might not need a laptop in the next three years, but wait. He might buy software upgrades from time to time, for every year or so. There could be recurring purchases like supplies based on his original purchase.

He has growing children. When they go to middle school or high school they might need a laptop. They then go to college and there will be more purchasing involved. We will need to know his demographics and family information to time these life events. In general, He might spend on gifts for others during Thanksgiving or Christmas, like most family men do. He might give a laptop or money goes to others. There could be potential annual purchases. If we can identify these events, these recurring needs and annual needs we would be able to estimate his total future business value.

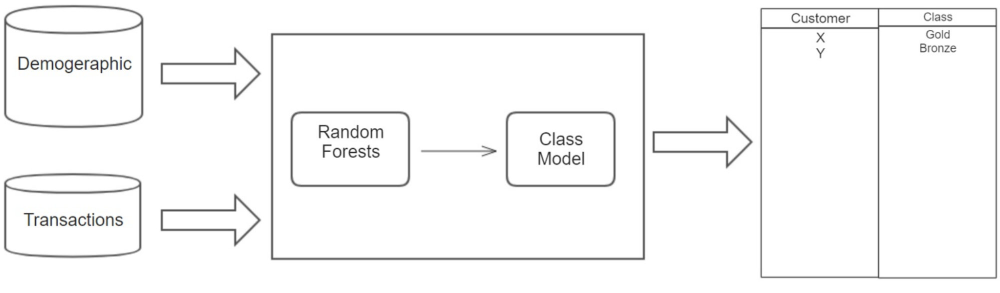
Then we might be able to schedule loyalty schemes, like discount coupons, package offers, to entice him to buy more. So, creating a customer's lifetime value and the timing of his future purchases will help us contact him at the right time with the right deals and generate more revenue.

Create customer value classes

One of the most early and basic customer management processes is to classify customers into various buckets, like bronze and gold, based on their past business and perceived future business. It then helps the business to target them differentially. When a new or potential customer is identified, classifying him or her as early as possible in the business cycle helps a business to focus more resources on those customers with significant future value.

Given customer's(called X) demographics, he might qualify as a gold customer since he would buy premium electronics for him and his growing family. Another customer(called Y), on the other hand, might be budget focused and might not spend a lot on electronics in the next five years. She would be a bronze customer. Given that, we want to focus our marketing resources on X to generate most bang for the buck. The goal of this use case is to build a prediction model that can classify new customers as silver, gold, or platinum.

Given that we are focused on new customers, the data set should be the one that is easily available for new customers. So that would be customer demographics information and second, their first purchase information like the product, the amount, timing, returns, warranty, etcetera. The training data will also have a customer class that tells what class the older customers belong to. This is a classic classification problem, so any previous algorithm variance will do the job, like Random Forests.



We will build a model that will predict the customer class based on the customer attributes as well as the transactions in the first purchase, and this would provide individual classes for each of our customers. The call to action areas that when a customer makes a new purchase, we will use this algorithm to classify them into one or more classes. Then these classes can then be used by our marketing or customer management teams to target him or her differentially in generating more revenue during subsequent visits.

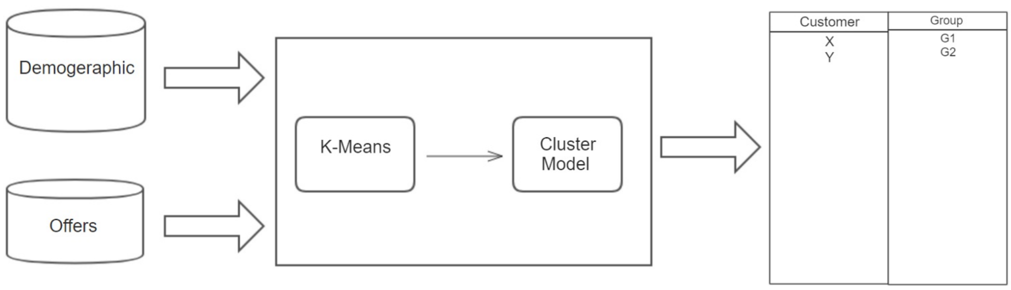
This classification can also be used by our support team to provide premium support. Knowing our best revenue sources helps nurture them and generate more revenue from them.

Discover response patterns

When the customer buys something from our business and then randomly switches to another business for the same service, it hurts. We want to keep our customers interested in buying more products and services from us. So, we reach out to our customers with coupons, offers, deals, packages, discounts, etcetera, however we can. Do all customers respond to these offers in the same way? No. There is a good chance that X would be more interested in coupons that he gets during Christmas so that he can buy gifts for his family.

Identifying how customers respond to our offers helps target our marketing dollars on customers who are most likely to respond. The goal of this use case is to identify distinct patterns in which our customers respond to offers. We group our customers into clusters. Then, identify patterns common to these clusters or groups, and then devise marketing schemes that generate better revenues for each of these individual clusters or groups. The data that will be used for this use case would be customer demographics, including the customer’s purchase history.

It also includes data about the offers made, like what event the offer was made, the product for which the offer was made, the discount offered, the channel in which the offer was made, whether the particular customers checked the offer, and would they really use the offer or not? The algorithm used will be K-Means clustering or any derivatives of this algorithm. We would use demographics and offer data to build the clusters and groups and then use it for our analysis.



We would use customer demographics data to build customer groups. Once the groups are identified, our marketing department can further analyse individual groups to understand their response patterns and what makes them unique. It can be demographics, products purchased, seasonality, or any other. Then, our marketing folks can make the call on how to better target those specific customer groups. The more we know about our customer behaviour, the better we can design our products and offers and generate more revenue from our customers.

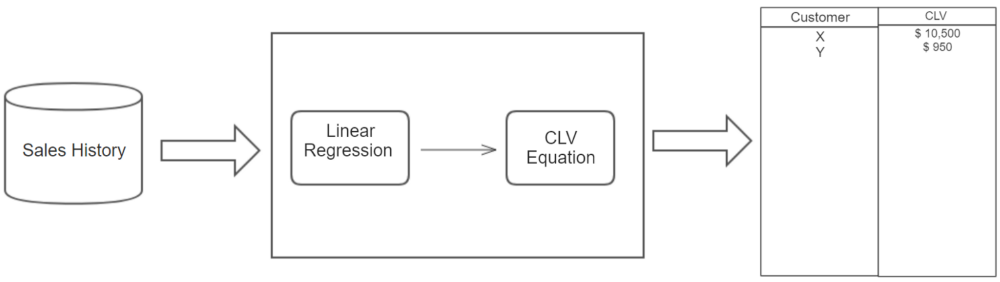
Predict customer lifetime value (CLV)

Customer Lifetime Value is a monetary value that represents the amount of revenue a customer will provide the business over the lifetime of the relationship. If we know ahead of time that X would bring in 10 k in revenue in the next five years, as opposed to Y, who will only generate $500, We can spend more marketing dollars on X and harvest his full potential budget. Tagging each customer with the Customer Lifetime Value helps a business focus on those customers who can bring in the most revenue in the future.

Several numerical techniques exist in computing Customer Lifetime Value, but we can compute CLV reliably only for customers who have a significant purchase history with the company. But how can we compute this for a recent customer? Well, predictive analytics is here to help. The goal of this use case to build a regression model that can predict the Customer Lifetime Value for a new or recent customer, based on his or her recent buying patterns.

It will also help us to recompute CLV more accurately, as more data becomes available. Customer Lifetime Value relies on data which is recent and shows frequency of purchases. So we would create a data set that provides the monetary value of purchases for each of the first six months of a customer for all old customers. This is the future set. The target attribute is the customer's actual CLV over the lifetime of the relationship.

CLV can be measured over a finite time, like two years or so, based on our business and the kind of historical data that we have. The algorithm used here would be linear regression. The first six months of data would become predictor variables and the CLV is the target variable. The analysis will provide an equation that can be used to compute CLV for new customers. We can actually generate different portions of this equation by using subsets of this data, say only the first month, then the first two months, etc.



More data, meaning more accuracy, as the lifetime of the customer grows. Once a new customer is captured in our system, we compute the CLV based on the first month's data alone. Then as more months pass by, we can refine the CLV by including more and more months. The CLV value can then be used by our sales, our service, and our marketing folks to identify customers with high CLV and provide them differentiated services and offers.

To summarize, this use case shows, how to predict CLV for a new customer based on CLVs of similar old customers.

Use case Predict CLV

We're going to see how we can predict the customer lifetime value for new customers based on models that we will base on existing customer data.

We have the first six months of revenue generated by these customers, month one to month six, how much revenue they generated, and the customer's lifetime value. Which may be possibly, is like a 3 year overall revenue that they gave. That is something we can decide based on the length to which our customers stay with our business. So this is the data that we're going to use, and we're going to use this to build a linear regression model that can then be used to predict the customer lifetime value.

We start out by importing a set of Python, Pandas, and Skylearn libraries. We load up the history.csv file. And we just look at the file to make sure that all the data elements have been loaded as integers, because the elements would require them to be integers. We do a little head here on the top filer cards to see if they have all loaded up properly. We then move on to correlation analysis between the first six months of data and the CLV.

	A	B	C	D	E	F	G	H
1	CUST_ID	MONTH_1	MONTH_2	MONTH_3	MONTH_4	MONTH_5	MONTH_6	CLV
2	1001	150	75	200	100	175	75	13125
3	1002	25	50	150	200	175	200	9375
4	1003	75	150	0	25	75	25	5156
5	1004	200	200	25	100	75	150	11756
6	1005	200	200	125	75	175	200	15525
7	1006	25	200	125	150	50	25	7950
8	1007	100	175	150	100	75	50	8813
9	1008	150	50	50	50	75	200	9375
10	1009	100	125	150	25	125	75	8063
11	1010	100	50	175	25	125	200	8313
12	1011	75	50	0	150	75	175	6825
13	1012	200	125	50	175	125	100	13425
14	1013	50	200	50	200	75	0	6813
15	1014	150	200	25	100	75	200	10381
16	1015	200	200	150	150	200	200	17100
17	1016	175	175	100	50	175	50	11619
18	1017	100	100	125	175	200	125	9750
19	1018	175	150	175	125	175	175	12813
20	1019	125	200	0	200	25	25	9675
21	1020	100	50	0	25	50	0	4950

from pandas import Series, DataFrame

import pandas as pd

import numpy as np

import os

import matplotlib.pylab as plt

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LinearRegression

import sklearn.metrics

raw_data = pd.read_csv("history.csv")

raw_data.dtypes

```
CUST_ID    int64
MONTH_1    int64
MONTH_2    int64
MONTH_3    int64
MONTH_4    int64
MONTH_5    int64
MONTH_6    int64
CLV        int64
dtype: object
```

raw_data.head()

Out[4]:

	CUST_ID	MONTH_1	MONTH_2	MONTH_3	MONTH_4	MONTH_5	MONTH_6	CLV
0	1001	150	75	200	100	175	75	13125
1	1002	25	50	150	200	175	200	9375
2	1003	75	150	0	25	75	25	5156
3	1004	200	200	25	100	75	150	11756
4	1005	200	200	125	75	175	200	15525

Do Correlation Analysis

cleaned_data = raw_data.drop("CUST_ID",axis=1)

cleaned_data .corr()['CLV']

```
Out[5]: MONTH_1     0.734122
        MONTH_2     0.250397
        MONTH_3     0.371742
        MONTH_4     0.297408
        MONTH_5     0.376775
        MONTH_6     0.327064
        CLV         1.000000
        Name: CLV, dtype: float64
```

We drop the customer ID column because it is not required for our model building purposes. So the correlation shows some really good correlation across different months. So I think now we can go forward and build a model. We start off by doing the training and testing split by using the train test split material available in the Skylearn library in the ratio 90:10. We print out the size to make sure that everything looks okay.

predictors = cleaned_data.drop(“CLV”,axis=1)

targets = cleaned_data.CLV

pred_train, pred_test, tar_train, tar_test = train_test_split(predictors, targets, test_size=.1)

print(“Predictor — Training : “, pred_train.shape, “Predictor — Testing : “, pred_test.shape)

```
Predictor - Training :  (90, 6) Predictor - Testing :  (10, 6)
```

We then go on to build a model. We start off with a linear regression model. We do a fit to build a model, print out the coefficients and the intercept. This gives us the actual equation, the linear regression equation. Then we can test on the testing data that we created by creating the predictions. And then we also look at the auto score for the regression model, which tells us how to create the model list. And that is turning out to have a 91% accuracy, which is really good.

#Build model on training data

model = LinearRegression()

model.fit(pred_train,tar_train)

print(“Coefficients: \n”, model.coef_)

print(“Intercept:”, model.intercept_)

#Test on testing data

predictions = model.predict(pred_test)

predictions

sklearn.metrics.r2_score(tar_test, predictions)

```
Coefficients:
 [ 34.59195048  11.53796271  15.17878598  11.72051702   8.60555913
   5.44443443]
Intercept: -199.535985333
```

Out[7]:

```
0.91592106093124581
```

It shows a 91% accuracy. This is an excellent model for predicting CLV. Now how do I predict for new customers? Suppose there is a new customer who has been with us for

3 months. We take out the first 3 months of revenue that he has given us, and based on that we build this array of all the values. We have the first 3 month values, and the next 3 months are going to be zeros. And we use that to print the CLV.

new_data = np.array([100,0,50,0,0,0]).reshape(1, -1)

new_pred=model.predict(new_data)

print(“The CLV for the new customer is : $”,new_pred[0])

```
The CLV for the new customer is : $ 4018.59836236
```

So this is how we can build a linear regression model for CLV and be able to predict our CLV for our new customers.