# Product Recommendations using Linear Predictive Modeling

Rajendra Banjade and Suraj Maharjan
Department of Research and Development
Verisk Information Technologies Pvt. Ltd. (Subsidiary of Verisk Analytics Inc.)
429 Hattisar, Kathmandu, Nepal
{rbanjade, sjmaharjan}@veriskhealth.com

*Abstract*

**Recommendation systems apply statistical and knowledge discovery techniques to the problem of making product recommendations and they are achieving widespread success in E-Commerce these days. A successful recommendation system fulfils several purposes and the choice of the methodology significantly influences the quality of recommendations and other aspects including scalability. As the volume of data in the e-commerce is growing massively, the system should also be able to address the need to provide the recommendations either by in-memory calculations or offline calculations, both demanding the high performance. For a large number of customers and products, the linear regression with a proper model selection can provide significantly better results and performance. Recommendations engines are increasingly becoming a popular choice for solving the problem of content discovery enabling the user to find personally relevant content that they might not have known was available. In this paper, we consider linear regression technique for analyzing large-scale dataset for the purpose of useful recommendations to e-commerce customers by offline calculations of model results.**

*Keywords*: e-commerce, regression analysis, modeling, tuning, scalability

## I. Introduction

The e-commerce sector has been growing year over year throughout the world offering thousands of products. Choosing a suitable product from among so many options is challenging for customers. With recommendations technologies, customers are automatically exposed to various products that they might like, without the need to search or browse for specific items. Recommendation systems generally infer these decisions by analyzing historical data (interaction between customers and products through sales transaction databases or real-time system capturing the action of customers on the websites, customer demographics, product characteristics etc) captured by the system. There are several different recommendation approaches categorized as Memory-based and Model-based. Model-based approaches are based on an offline pre-preprocessing and only the learned model is used to make predictions whereas, the Memory-based approaches are based on calculations at real-time. However, they have their own limitations [1][2].

Efficient recommendation systems not only benefit the customers but also the marketers and product companies by boosting their products sales among wide range of potential customers.

In this paper, we present a method of recommendation by using multivariate linear regression, a statistical analysis method, which is flexible and scalable.

## II. Problem Statement

Due to rapid surge in the e-commerce data, online merchants place a great deal of attention on empowering their e-commerce solutions with the scalable high performance techniques. As some of the techniques for product recommendation such as traditional collaborative filtering are impractical when the data size is huge, and many performance tuning methods are being applied but that has unfavorable effect on the accuracy in the recommendations [3]. Some of the issues seen in many of the recommendation systems are:

- Algorithms are not scalable to process the ever growing data in a reasonable time, especially they scale nonlinearly.
- Generated models are too complicated for human to comprehend.
- Unable to handle noisy and outliers.

In this paper, we explore the possibilities of using a multivariate linear regression with tuning whose complexity grows linearly and is a transparent method.

## III. Recommendation Algorithms

In this section, we introduce some of the prediction algorithms that are in use.

Collaborative filtering methods are based on collecting and analyzing a large amount of information on users' behavior, activity or preferences and predicting what users will like based on their similarity to other users. Traditional collaborative filtering does little or no offline computation, and its online computation scales with the number of customers and catalog items.

### *A. Apriori Algorithm*

Apriori algorithm is a classical algorithm for learning the association rules using a "bottom up" approach where frequent subsets are extended one item at a time (candidate generation). The algorithm terminates when there are not any large item sets. Generally association rules are helpful when we have a rough idea of what we are looking. Apriori algorithm is not suitable to mine a large data set for long patterns [6]. The candidate generation is the inherent cost of the Apriori algorithms, no matter what implementation technique is applied.

### *B. K-nearest Neighborhood*

K-nearest neighborhood is a very intuitive method that classifies unlabeled examples based on their similarity with examples in the training set. The K-nearest neighborhood is a lazy learning algorithm as it defers data processing until it receives a request to classify an unlabelled example. The K-nearest neighborhood requires a large storage and is highly susceptible to the curse of dimensionality [7].

## IV. Methodology

### *A. Data Preparation*

The predictive models generally require large data-set which is fed into the training algorithms in order to calculate the values of the necessary parameters backing the algorithms. Typically the efficiency of the prediction depends on the nature of data used in the model development. Hence, the collection, cleansing, and formatting the data play crucial role in the data mining processes. Generally the domain experts, having fair amount of knowledge about meta-data, perform this task of identification of data sources, collection of data, pre-processing, handling the noisy, missing data and other. During this process, the data may be tweaked multiple times in no prescribed order but the original meaning of data is never lost. Typically the following information is required for the model calibration:

- Demographic
- Products details
- Transaction history
- Customer activities – search, navigation, product ratings etc.

Internet is the prime source for collecting the information about the customers. Majority of the e-commerce sites keep track of their customers' interests by capturing the information entered by their customers in their database system. Moreover, they frequently use polling techniques to determine the best products or may ask their customers to sort the list of products in order of their preference. The record about the items purchased, viewed, viewing times, navigation of sites, query strings used in searching the products etc by the customers also play vital role in determining the customer's interests. The social networking sites can also be used to determine the likes and dislikes of their customers. Data also come from surveys of small and medium logistics enterprises, including the distribution center, processing, warehousing, packaging and information service providers.

### *B. Classification*

Developing models on all possible values of different variables (i.e. product attributes, customer attributes etc) adds complexity as their possible values might present thinly even in the fairly large development data set. Classifying data into different coherent groups of interests creates group-level predictive results and helps analyze and develop models. For instance, rather than analyzing the customer of every age and gender, creating different groups based on age band and gender is much easier to analyze without reducing the predictive power. The granularity is fixed based on the nature of data.

For example, customers are categorized based on their age and gender into some categories (in Level-1) as,

| Gender | Age-range | Level-1 |
|---|---|---|
| Male | 20-25 | M1 |
| Female | 20-25 | F1 |
| Male | 26-30 | M2 |
| … | | |

And products can be categorized (for example, books on programming languages can be grouped as programming books in the first level, and in the second level, different first level items are grouped into a single category) as shown in the example below,

| Product | Level-1 | Level-2 |
|---|---|---|
| C programming book | Programming | Computer |
| Java beginners' guide | Programming | Computer |
| PC Bible | Hardware | Computer |
| … | … | |

*C. Model Calibration*

In mathematics, regression analysis is done to understand how the typical value of dependent variable changes when any one of the independent variables is varied, while the other variables are held fixed. Moreover, the regression analysis is also used to determine the relationship between the independent and the dependent variables. We use Multivariate Linear Regression Analysis, with Ordinary Least Squares (OLS) estimation, on the development dataset for the model calibration.

The general multivariate linear regression model is written in the equation as,

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots\dots + \beta_k x_k + \varepsilon. \qquad (1)$$

Where,

$\beta_0$ is the intercept

$\beta_1$ is the parameter associated with $x_1$ (slope parameter)

k represents the number of independent variables

$x_1$-$x_k$ independent variables

y dependent variable

$\varepsilon$ error term or disturbance.

If there are 'n' training data set then applying these data in (1), we get,

$$y_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \dots\dots\dots + \beta_k x_{1k} + \varepsilon_1$$
$$y_2 = \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \dots\dots\dots + \beta_k x_{2k} + \varepsilon_2$$
$$\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots$$
$$y_n = \beta_0 + \beta_1 x_{n1} + \beta_2 x_{n2} + \dots\dots\dots + \beta_k x_{nk} + \varepsilon_n.$$

This can be represented in the matrix form as,

$$Y = X\beta + \varepsilon. \qquad (2)$$

Where,

$$Y = [y_1\, y_2 \dots\dots y_n]'$$

$$X = \begin{bmatrix} 1 & x_{11} & .. & x_{1k} \\ 1 & x_{21} & .. & x_{2k} \\ .. & .. & .. & .. \\ 1 & x_{n1} & .. & x_{nk} \end{bmatrix}$$

$$\beta = [\beta_0\ \beta_1\ \beta_2 \dots\dots\dots\dots \beta_k]'$$
$$\varepsilon = [\varepsilon_1\ \varepsilon_2 \dots\dots\dots\dots\dots\dots \varepsilon_n]'$$

The parameter represented by β is calculated as

$$\beta = (X'X)^{-1} XY \qquad (3)$$

Where,

$X'$ is the transpose of $X$

*Goodness of fit of a model*: Generally, the R-Square ($R^2$) coefficient of determination is used to get the information about the goodness of fit of a model. In regression, the $R^2$ coefficient of determination is a statistical measure of how well the regression line approximates the real data points. An $R^2$ of 1.0 indicates that the regression line perfectly fits the data.

The $R^2$ coefficient of determination is calculated by simply squaring the correlation coefficient between the observed and modeled (predicted) data values.

*D. Tuning*

The model output is tuned for better predictions using splining. Spline is a piecewise-polynomial real function defined as,

$S : [a, b] \rightarrow R$ on the interval $[a, b]$ composed of k ordered disjoint sub intervals $[t_{i-1}, ti]$ with

$$a = t_0 < t_1 < \dots\dots\dots\dots < t_{k-1} < t_k = b.$$

The restriction of S to an interval 'i' is a polynomial

$$P_i : \quad [t_{i-1}, t_i] \rightarrow R,$$

So that

$$S(t) = P_1(t), \quad t_0 <= t < t_1,$$
$$S(t) = P_2(t), \quad t_1 <= t < t_2,$$
$$\dots$$
$$S(t) = P_k(t), \quad t_{k-1} <= t <= t_k.$$

The highest of the polynomials $P_i(t)$ is said to be the order of the spline S.

The main purpose of splinig is to choose the polynomials in a way that guarantees sufficient smoothness of S.

*E. Implementation*

For any given product, the selected model is run for every customer. The input to the model are the cleaned and classified information of customers' details including demographics, transaction history and activities in the website (if any), and the details of a product at-hand.

Based on the relative score produced after model run, the top selected members are recommended for that product.

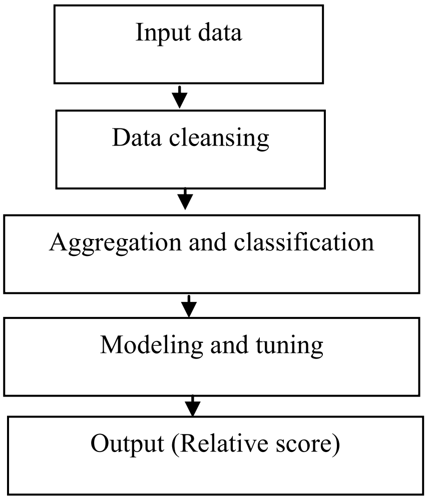

Fig. 1. Implementation process

Fig.1 shows the implementation process. Actually, the score produced in the output is relative to the development sample.

For performance evaluation, the number of independent variables, the attributes in the input data and their size such as the average number of records per customer are varied and observed.

## V. CONCLUSION and FUTURE WORK

In this paper we have presented a model-based product recommendation algorithm using multivariate linear regression analysis. The purposed scalable framework allows the speedy model development and flexible integration of various parameters preserving the quality.

The major challenge for model development and validation is the requirement of up-to-date, realistic, and high quality data set that should be sufficiently large, especially for the regression analysis applied in our system. The classification, model design, and tuning of model output can be improved after some experiments and observations. And a hybrid system combining this technique with other algorithm(s) may improve the quality of recommendations. More experimental validations using the real world dataset is needed to recognize this technique. In the future, we will go along validating and amending the process that we have presented here.

## REFERENCES

[1] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl, "Analysis of Recommendation Algorithms for E-Commerce", *EC'00*, October 17-20, 2000, Minneapolis, Minnesota. Available: http://www.uop.edu.jo/download/research/members/ec00.pdf

[2] Zan Huang, Deniel Zeng, and Hsinchun Chen, "A Comparative Study of Recommendation Algorithms in E-Commerce Applications". Available: http://ai.arizona.edu/intranet/papers/comparative.ieeeis.pdf

[3] Greg Linden, Brent Smith, and Jeremy York. Amazon.com Recommendations, Item-to-Item Collaborative Filtering. Available: http://www.cs.umd.edu/~samir/498/Amazon-Recommendations.pdf

[4] Andreas Mild and Martin Natter. Collaborative filtering or Regression models for internet recommendation systems. Available: http://www.marketing.uni-frankfurt.de/fileadmin/Publikationen/Journal_of_Targeting_Collaborative_Filtering.pdf

[5] Dietmar Jannach (2011). Tutorial: Recommender Systems. In *International Joint Conference on Artificial Intelligence, Barcelona*. Available: http://ls13-www.cs.uni-dortmund.de/homepage/publications/jannach/tutorial-ijcai2011.pdf

[6] Sixue Bai, Xinxi Dai (2007). An Efficiency apriori Algorithm: P_Matrix Algorithm. In *Proceeding ISDPE '07 Proceedings of the The First International Symposium on Data, Privacy, and E-Commerce,* IEEE Computer Society Washington, DC, USA. http://portal.acm.org/citation.cfm?id=1338004

[7] Ricardo Gutierrez-Osuna, "Introduction to Pattern Recognition", Wright State University. Available: http://courses.cs.tamu.edu/rgutier/cs790_w02/l8.pdf