Churn Modelling: A Perspective on Mixture-Based Clustering of Customer Behaviours

Victor (Sheng) Wang

University of Oxford

Supervisors: Andrew Mellor, Junaid Mubeen

A technical report for

InFoMM CDT Mini-Project 1

in partnership with

Whizz Education

Trinity 2018

Contents

1 Introduction

Retained customers in general create higher revenues than new customers do, and making a sell to a new customer can cost up to 5 times more depending on the business [2]. Therefore, many companies form the Customer Relationship Management (CRM) team with a focus on customer retention strategies. A crucial step is then to identify high risk customers who are intending to discontinue their usage of the services. This assessment is better known as *churn prediction*.

Our project aims to perform the churn prediction task based on investigation of customers's behavioural clusterings, and formulate the processes into a scalable pipeline which can be easily reused, updated and extended for many applications. In particular, we apply the pipeline to analyse pupil subscribers' data for Whizz Education (referred to as "Whizz"). Whizz provides online virtual tutorial service, Math-Whizz, which pupils can access by purchasing subscriptions. We have investigated whether behavioural-based data analytics can be used to help detect potential subscription cancellations.

The pipeline starts from representing pupils' behaviours by numerical data, also known as *feature* extraction. The structured features' data are fed into a mixture model that decodes features' distribution as a weighted combination of simple distributions. Since simple distribution is assumed to be generated by an unobserved *state*, we are interested in uncovering the states as well as their associated emitted feature distributions, also known as *clusters*. The mixture model is deemed to be effective if the identified clusters of pupils exhibit distinguishable proportions of churn, or *churn rates*. Assessing churn rate of the fitted mixture model results in a trained model that establishes a map between features to probability of churn. This enables the prediction of new coming pupils' churn probabilities by feeding their feature data into the trained model.

We show that clusters inferred from behavioural data are characterised by non-trivially different churn rate. Moreover, there is recognisable pattern observed in cluster sizes. This verifies the effectiveness of mixture model in uncovering what level of churn risk the pupils are at. We further investigate how features impact churn probability and examine that the mixture model can result in very minimal overfitting issues in prediction practices. We also infer the temporal transitional probabilities of states by studying on the dynamics of behavioural changes.

The report is structured as follows. We start by elaborating the modelling framework and pipeline in section 2. Next in section 3 we describe the feature data preparation and pre-processing techniques used to adapt Whizz's data for better modelling performance. Then we explain the details of clustering analysis and model evaluation in section 4. Finally, we carry out prediction task and also assess overfitting issues, as detailed in section 5. We conclude the project with deliverables and future directions in section 6.

2 Modelling Customer Behaviors

Markov State and Mixture Model

The goal is to assess the likelihood of customers canceling by examining their behaviors.

We model the behaviors of customers as the emission of Markov states, either hidden or observable. With this modelling assumption, we can also do temporal transitional analysis to gain insights of customers' dynamic behavioral change.

We can define states by some features.

We can also infer states from data through clustering task. To undertake the clustering task, we choose to use mixture model.

2.1 Representing Behaviours by Features

2.2 Customer Journeys and Markov Chains

Schematic plot showing the relations between states and behaviors.

Schematic plot showing temporal transition of states.

2.3 Clustering Using Mixture Model

Generative process

$$\mathbb{P}(\mathbf{y}, \mathbf{c}, \boldsymbol{\theta}) = \prod_{k=1}^{K} G_0(\theta_k) \prod_{n=1}^{N} F(y_n|\theta_{c_n}) P(c_n) \tag{1}$$

- $\mathbf{y} = \{y_1, y_2, \cdots, y_N\}$, observations
- $\mathbf{c} = \{c_1, c_2, \cdots, c_N\}$, cluster assignments
- $\boldsymbol{\theta} = \{\theta_1, \theta_2, \cdots, \theta_K\}$, cluster parameters
- $G_0(\theta)$, a prior over the cluster parameters
- $P(c)$, a prior over the mixing distribution
- $F(y_n|\theta_{c_n})$, a hypothetical distribution over the observations
- Assumptions: each observation is conditionally independent given its latent cluster assignments and the cluster parameters

Posterior probability of assignments

$$\mathbb{P}(\mathbf{c}|\mathbf{y}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{c})P(\mathbf{c})}{\sum_{\mathbf{c}} \mathbb{P}(\mathbf{y}|\mathbf{c})P(\mathbf{c})}, \tag{2}$$

where

$$\mathbb{P}(\mathbf{y}|\mathbf{c}) = \int_{\theta} \left[\prod_{n=1}^{N} F(y_n|\theta_{c_n}) \prod_{k=1}^{K} G_0(\theta_k) \right] \mathrm{d}\theta \tag{3}$$

BNP clustering assume that there is an infinite number of latent clusters (namely $K \to +\infty$), but that a finite number of them is used to generate the observed data. There are an infinite number of clusters, though a finite data set only exhibits a finite number of active clusters.

2.4 Modelling Pipeline

Feature extraction

Features distributional modelling

Fitting mixture model

Analytic (churn rate calculation, clusters interpretation)

3 Data Description and Pre-processing

3.1 Whizz Database

Imbalanced data - the reason we don't mak them balanced is because we do not want to twist numerical value of churn rate. For example, if we balance the data set perfectly, then the population churn rate will become 50%. After we do the clustering task, it is not straightforward to transform back the cluster churn rate for the balanced data set to that for the original imbalanced data set.

3.2 Feature Extraction

A table showing the definition of features.

3.3 Data Transformation

We denote the feature data by a matrix $\mathbf{X} = (x_{ij}) \in \mathbb{R}^{m \times n}$, which describes n observed values for each of the m features. In addition, we denote the i-th row of $\mathbf{X}$ by $\mathbf{x}_{i,\mathrm{R}} = [x_{i1}\ x_{i2}\ \cdots\ x_{in}]$, and the j-th column by $\mathbf{x}_j = [x_{1j}\ x_{2j}\ \cdots\ x_{mj}]^\top$.

In general, learning algorithms benefit from standardisation of the data set. We have employed 3 transformations in sequence to make our data more suitable for mixture model learning task. The transformations are performed for each feature separately, resulting in different sets of transformation parameters for different features. To be specific, for i-th feature we describe the transformations as following.

- Linear transformation
  The linear transformation is applied either to ensure all data to be positive

for eligibility of applying the following power transformation, or to adapt to distributional modelling choice. It has the format:

$$\mathbf{x}'_{i,\mathrm{R}} = a_i\mathbf{1} + b_i\mathbf{x}_{i,\mathrm{R}}, \tag{4}$$

where a_i and b_i are constants and $\mathbf{1} \in \mathbb{R}^{1\times n}$ is the row vector of all ones.

- Box-Cox power transformation
  The Box-Cox power transformation is used to modify the distributional shape of a set of data to be more normally distributed so that tests and confidence limits that require normality can be appropriately used. It has the format:

$$x'_{ij} = \begin{cases} \frac{x_{ij}^{\lambda_i}-1}{\lambda_i} & \text{if } \lambda_i \neq 0, \\ \ln x_{ij} & \text{if } \lambda_i = 0, \end{cases} \tag{5}$$

  for $j = 1,\ 2,\ \ldots, m$. In Box-Cox transformation, λ_i is estimated by maximizing the likelihood function [reference].

- Standardisation
  We standardise data for different features by scaling them into the same range to ensure the robustness to very small standard deviations of features. We choose to scale all features into range $[1, 100]$. If we denote the maximum and minimum values of observed feature i as $x_i^{\max}$ and $x_i^{\min}$ respectively, then the standardisation is a linear transformation such that,

$$\mathbf{x}'_{i,\mathrm{R}} = \mathbf{1} + \frac{100-1}{x_i^{\max} - x_i^{\min}}\left(\mathbf{x}_{i,\mathrm{R}} - x_i^{\min}\mathbf{1}\right). \tag{6}$$

Table of transformation parameters

Exhibitions of pre and post transformation

4 Clustering Analysis

4.1 Distributional Modelling of Features

empirical distributional properties require simply Gaussian mixtures

correlation analysis enables to separate a few features to be fitted independently

4.2 Fitting Baysian Gaussian Mixture Model

brief description of EM-algorithm and Baysian inference

4.3 Assessing Churn Probability

(start of the supervised learning, because of the usage of churn label)

compute cluster churn rate

interpret churn probability of individual pupil

4.4 Feature Impact

4.5 Markov States Temporal Transition Analysis

4.5.1 Defining Markov States

4.5.2 Transitional Analysis

calculate transition probability

visualise transition (matrix + Sankey plot) [1]

5 Churn Probability Prediction

5.1 Prediction Workflow

5.2 Evaluating Overfitting

6 Conclusion

References

[1] P. Rothenbuehler, J. Runge, F. Garcin, and B. Faltings. Hidden markov models for churn prediction. In *2015 SAI Intelligent Systems Conference (IntelliSys)*, pages 723–730, Nov 2015.

[2] S. F. Slater and J. C. Narver. Intelligence generation and superior customer value. *Journal of the Academy of Marketing Science*, 28(1):120, Dec 2000.