SPRINT I

Universidad del Desarrollo
Universidad de Excelencia

Modelo Predictivo de Fuga de Clientes en Servicio de Streaming
UDD MAGÍSTER DE DATA SCIENCE

José Pedro Cordero / Hernán Rivera

Desafío

- **KKBOX** es el servicio de transmisión de música líder en Asia, y cuenta con la biblioteca de música Asia-Pop más completa del mundo con más de 30 millones de canciones.
- El desafío consiste en desarrollar un algoritmo que prediga si un usuario de suscripción dejará de usar el servicio, esto, con un conjunto de testeo (Marzo 2017) de la empresa **KKBOX**.
- Para un negocio de suscripción, predecir con precisión el abandono es fundamental para el éxito a largo plazo. Incluso pequeñas variaciones en la rotación pueden afectar drásticamente las ganancias.
- La métrica de evaluación es **Log Loss**, donde **N** es el número de observaciones, **Log** es el logaritmo natural, **yi** es el objetivo binario y **pi** es la probabilidad de predecir que **yi** es igual a 1

$$logloss = -\frac{1}{N}\sum_{i=1}^{N}(y_i \log(p_i) + (1 - y_i)\log(1 - p_i))$$

Definición

- **Churn (Wikipedia)**: La tasa de abandono, cuando se aplica a una base de clientes, se refiere a la proporción de clientes contractuales o suscriptores que dejan un proveedor durante un período de tiempo determinado.
- En contexto, la definición de **abandono/renovación** puede ser complicada debido al modelo de suscripción de KKBox. Dado que la mayoría de la duración de la suscripción de KKBox es de 30 días, muchos usuarios se vuelven a suscribir cada mes. Los campos clave para determinar la **rotación/renovación** son la fecha de la transacción, la fecha de vencimiento de la membresía y el campo **is_cancel**.
- El campo **is_cancel** indica si un usuario cancela activamente una suscripción. La cancelación de la suscripción no implica que el usuario haya cancelado el servicio. Un usuario puede cancelar la suscripción del servicio debido a cambios en los planes de servicio u otros motivos. **El criterio de "abandono"** no es una nueva suscripción de servicio válida dentro de los 30 días posteriores a la expiración de la membresía actual.

Dataset

El conjunto de training y testing corresponde a los usuarios cuya membresía en el mes de febrero de 2017, y los datos de testing corresponde a los usuarios cuya suscripción expiraba en el mes de marzo de 2017.

Datasets	Descripción	Número de Registros
members_v3.csv	Información de miembros	~ 5m de registros
train.csv train_v2.csv	Dataset de entrenamiento. Dataset de entrenamiento actualizado.	~ 22 m de registros
transactions.csv transactions_v2.csv	Transacción de usuarios. Transacción de usuarios actualizada.	
user_logs.csv user_logs_v2.csv	Log diario que describe el comportamiento de los usuarios. Log diario de usuarios actualizado	~ 400m de registros
sample_submission_zero.csv sample_submission_v2.csv	Muestra que contiene el user id Muestra actualizada	

Modelo de Datos

El modelo de datos de la competencia KKBOX se divide en dos tipos de tablas, las que contienen el total de registros y las que contienen una sampleo del total de registros para efectos de poder procesarlas con poco poder de cómputo, estas tablas finalizan en _v2 .

train
msno
is_churn

train_v2
msno
is_churn

sample_submission_v2
msno
is_churn

transactions
msno
payment_method_id
payment_plan_days
plan_list_price
actual_amount_paid
is_auto_renew
transaction_date
membership_expire_date
is_cancel

transactions_v2
msno
payment_method_id
payment_plan_days
plan_list_price
actual_amount_paid
is_auto_renew
transaction_date
membership_expire_date
is_cancel

user_logs
msno
date
num_25
num_50
num_75
num_985
num_100
num_unq
total_secs

user_logs_v2
msno
date
num_25
num_50
num_75
num_985
num_100
num_unq
total_secs

members_v3
msno
city
bd
gender
registered_via
registration_init_time

Dataset

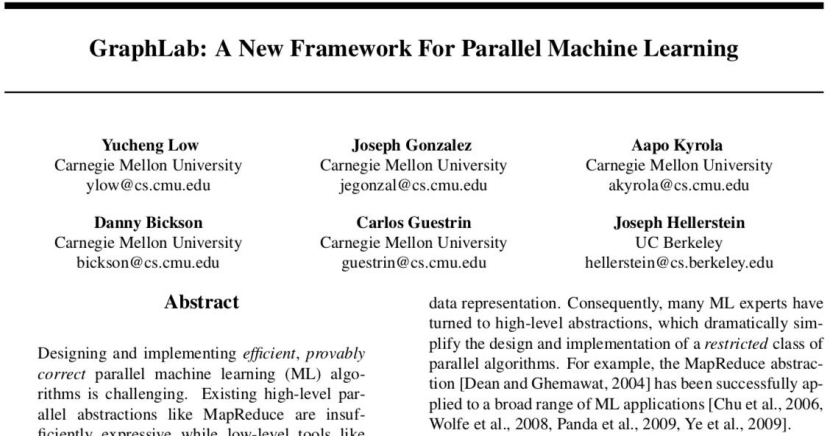
A continuación describiremos las columnas de los datasets.

Datasets	Variables
members_v3.csv	user_id, city, age, gender, registered_method, registration_init_time
train.csv train_v2.csv	user_id, is_churn
transactions.csv transactions_v2.csv	user_id, payment_method_id, payment_plan_days, plan_list_price, actual_amount_paid, is_auto_renew, transaction_date, membership_expire date, is_cancel
user_logs.csv user_logs_v2.csv	user_id, #of songs played less than 25/50/75/98.5/100 % of song length, num_unq, total_secs
sample_submission_zero.csv sample_submission_v2.csv	user_id, is_churn

Implementación

Para abordar el desafío de este proyecto, utilizaremos los siguientes elementos:

- Notebook Mac Air 8 GB RAM
- Disco Duro SATA 512 GB
- Framework Python Graphlab de Turi



GraphLab: A New Framework For Parallel Machine Learning

Yucheng Low
Carnegie Mellon University
ylow@cs.cmu.edu

Joseph Gonzalez
Carnegie Mellon University
jegonzal@cs.cmu.edu

Aapo Kyrola
Carnegie Mellon University
akyrola@cs.cmu.edu

Danny Bickson
Carnegie Mellon University
bickson@cs.cmu.edu

Carlos Guestrin
Carnegie Mellon University
guestrin@cs.cmu.edu

Joseph Hellerstein
UC Berkeley
hellerstein@cs.berkeley.edu

Abstract

Designing and implementing *efficient*, *provably correct* parallel machine learning (ML) algorithms is challenging. Existing high-level parallel abstractions like MapReduce are insufficiently expressive while low-level tools like data representation. Consequently, many ML experts have turned to high-level abstractions, which dramatically simplify the design and implementation of a *restricted* class of parallel algorithms. For example, the MapReduce abstraction [Dean and Ghemawat, 2004] has been successfully applied to a broad range of ML applications [Chu et al., 2006, Wolfe et al., 2008, Panda et al., 2009, Ye et al., 2009].

- **Notebook**: https://github.com/hriverap/machine-learning/tree/master/churn-model-kkbox