Predicting PicCollage users’ first purchase for targeted promotions

Team 2
Reggie Escobar . Eduardo Salazar
Uni Ang . Lynn Pan
Cardinal Blue, Inc.

Founded in 2011

100m installs

$2.3m Seed funding (2013)

In-app purchases - backgrounds; stickers & watermark removal
About PicCollage

**WATERMARK**

**STICKER**

**BACKGROUND**
Problem:
Limited user info data hinders user specific targeted promotions

Business Goal:
Target users likely to make a first purchase
Send personalized promotions

Stakeholder:
PicCollage
Data Mining Goal

Ranking the user’s with high probability of making a first purchase when they create their first collage

Supervised. Forward-looking

Categorical: Binary for first purchase (Y/N)
One Month New user (2017/9) data from firebase

- Structure: User info + Events info by session
  - First open
  - First Collage Save
  - First Purchase
  - First open time
  - Continent / Country
  - Device category
  - Login
  - create_collage_empty
  - Create_Collage: Empty / Grid / Remix
  - Remix_category
  - Add Photos: type & avg number
  - Add photo from web
  - Per Collage: Sticker / ...
  - Font type : 10 type
  - Share Collage : type + number
  - Background pick : search / URL / library
  - Doodle per added
  - Sum of Frame try
  - Sum of Clip
  - Avg Collage in Library
  - Num of sticker preview
  - Export collage : sticker / background/ .....

Data Source

Share Collage : type + number
Export collage : sticker / background/ .....
### Data Description and Preparation

- **Training data**
  - # record: 10,000
  - % purchase: 28%
  - # record: 9344
  - % purchase: 50%

- **Validation data**
  - # record: 11,405
  - % purchase: 28%

- **Test data**
  - # record: 11,405
  - % purchase: 28%

### Filter

- By User
  - Create derived variables from events
  - Filter events before first purchase/first collage save

### Missing value

- Country
  - device language

---

**Pico Extractor**

This project is a parser to get the predictors.
Methods & Performance Evaluation

- Task: Ranking
- Benchmark: naive (all class “0”)
- Method
  - Naive Bayes (Binned variables)
  - Classification tree (single)
  - Random Forest
  - Boosted Tree
  - Logistic Regression
- Performance measure
  - Lift Chart
  - Decile lift chart
  - Sensitivity
  - Specificity
Method: Random Forest

Test Data scoring - Summary Report

<table>
<thead>
<tr>
<th>Confusion Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual Class</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>Overall</td>
</tr>
</tbody>
</table>

Performance:

- Success Class: 1
- Precision: 1
- Recall (Sensitivity): 0.020685
- Specificity: 1
- F1-Score: 0.040532

Lift chart (test dataset)

Decile-wise lift chart (test dataset)
Method: Single Tree

oversample / Full Tree / terminal 934
Method: Random Forest oversampling

Test Data scoring - Summary Report

Cutoff probability value for success (UPDATABLE): 0.5

Confusion Matrix

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1494</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>842</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2501</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>3365</td>
</tr>
</tbody>
</table>

Error Report

<table>
<thead>
<tr>
<th>Class</th>
<th># Cases</th>
<th># Errors</th>
<th>% Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2336</td>
<td>842</td>
<td>36.04452</td>
</tr>
<tr>
<td>0</td>
<td>5866</td>
<td>2501</td>
<td>42.63553</td>
</tr>
<tr>
<td>Overall</td>
<td>8202</td>
<td>3343</td>
<td>40.75835</td>
</tr>
</tbody>
</table>

Performance

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Success Class</td>
<td>1</td>
</tr>
<tr>
<td>Precision</td>
<td>0.373967</td>
</tr>
<tr>
<td>Recall (Sensitivity)</td>
<td>0.639555</td>
</tr>
<tr>
<td>Specificity</td>
<td>0.573645</td>
</tr>
<tr>
<td>F1-Score</td>
<td>0.471963</td>
</tr>
</tbody>
</table>

Lift chart (test dataset)

Decile-wise lift chart (test dataset)
Method: Boosted Tree

Test Data scoring - Summary Report

<table>
<thead>
<tr>
<th>Confusion Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual Class</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>Overall</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Success Class</td>
</tr>
<tr>
<td>Precision</td>
</tr>
<tr>
<td>Recall (Sensitivity)</td>
</tr>
<tr>
<td>Specificity</td>
</tr>
<tr>
<td>F1-Score</td>
</tr>
</tbody>
</table>

[Images of lift charts and performance metrics]
Method: Logistic Regression

Variables selection—Stepwise

- Num_events
- Create_collage_empty
- Num_background_try
- Num_frame_try
- Avg_of_image_export
- Avg_photo_facebook
- remix_cat_Back_to_School
- remix_cat_Congrats
- remix_cat_Just_for_Fun
- remix_cat_Labor_Day_Weekend
- font_Roboto_BlackItalic
- Create_collage_grid
- Login

Confusion Matrix and Statistics

<table>
<thead>
<tr>
<th>Reference</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prediction</td>
<td>1</td>
<td>104</td>
</tr>
<tr>
<td>0</td>
<td>2745</td>
<td>6359</td>
</tr>
</tbody>
</table>

- Accuracy: 0.6566
- Sensitivity: 0.03650
- Specificity: 0.90921
Boosted Tree and Random Forest are top two best model.
Recommendations

• How to use this model for marketing promotion?
  Offering bundles/discount to users that have a high probability of making a first purchase.

• Model recommendation
  – Due to the unbalanced dataset and ranking goal, we suggest to adopt over-sampling

• Date recommendation
  – The data we are using now is missing the October purchase.
  – Collect events data per user for their 30 days full history.

• Variables recommendation
  – Getting user information might help to predict first purchase earlier.