

Click-Through Rate prediction: TOP-5 solution for the Avazu contest

Dmitry Efimov
Petrovac, Montenegro

June 04, 2015

Outline

Provided data

Likelihood features

FTRL-Proximal Batch algorithm

Factorization Machines

Final results

Competition

The image shows a browser window displaying a Kaggle competition page. The browser's address bar shows 'kaggle.com'. The page header includes the Kaggle logo, navigation links for 'Host', 'Competitions', 'Jobs', and 'Community', and the user's name 'Dmitry Efimov' with a 'Logout' button. The main content area features the Avazu logo, the competition title 'Click-Through Rate Prediction', and its status: 'Completed • \$15,000 • 1,604 teams'. The dates are 'Tue 18 Nov 2014 - Mon 9 Feb 2015 (53 days ago)'. A left sidebar contains a 'Dashboard' menu with options like 'Home', 'Data', 'Make a submission', 'Information', 'Forum', 'Leaderboard', 'My Team', and 'My Submissions'. The main content area has a breadcrumb trail 'Competition Details » Get the Data » Make a submission' and a large heading 'Predict whether a mobile ad will be clicked'. Below this is a paragraph explaining CTR: 'In online advertising, click-through rate (CTR) is a very important metric for evaluating ad performance. As a result, click prediction systems are essential and widely used for sponsored search and real-time bidding.' A large, stylized red and white graphic at the bottom right reads 'CTR prediction contest'.

kaggle.com

Apple iCloud AUS Home iLearn IT FAQ IT HelpDesk Library Password Station AUS Email Gmail AUS banner Мысль/тут/пан Twitter AUS TeleDir AUS Rooms AUS Files

kaggle

Host Competitions Jobs Community

Dmitry Efimov Logout

Avazu

Completed • \$15,000 • 1,604 teams

Click-Through Rate Prediction

Tue 18 Nov 2014 - Mon 9 Feb 2015 (53 days ago)

Dashboard

- Home
- Data
- Make a submission

Information

- Description
- Evaluation
- Rules
- Prizes
- Timeline

Forum

Leaderboard

- Public
- Private

My Team

- GitHub

My Submissions

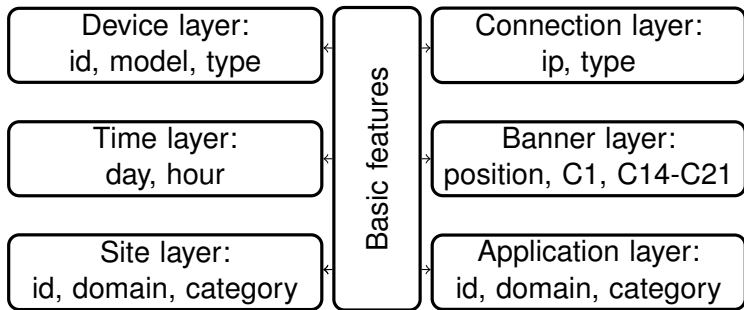
Competition Details » Get the Data » Make a submission

Predict whether a mobile ad will be clicked

In online advertising, click-through rate (CTR) is a very important metric for evaluating ad performance. As a result, click prediction systems are essential and widely used for sponsored search and real-time bidding.

CTR prediction contest

Provided data



Notations

X: $m \times n$ design matrix

$$m_{train} = 40\,428\,967$$

$$m_{test} = 4\,577\,464$$

$$n = 23$$

y: binary target vector of size m

x^j: column j of matrix X

x_i: row i of matrix X

$\sigma(\mathbf{z}) = \frac{\mathbf{1}}{\mathbf{1} + \mathbf{e}^{-\mathbf{z}}}$: sigmoid function

Evaluation

Logarithmic loss for $y_i \in \{0, 1\}$:

$$L = -\frac{1}{m} \sum_{i=1}^m (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i))$$

\hat{y}_i is a prediction for the sample i

Logarithmic loss for $y_i \in \{-1, 1\}$:

$$L = \frac{1}{m} \sum_{i=1}^m \log(1 + e^{-y_i p_i})$$

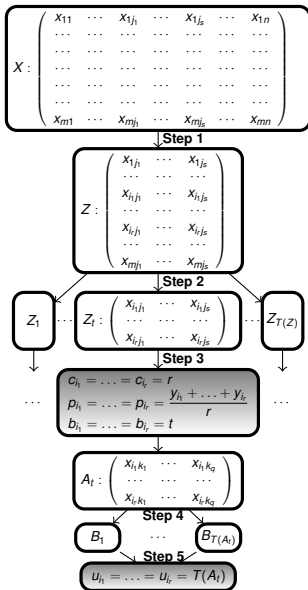
p_i is a raw score from the model

$\hat{y}_i = \sigma(p_i), \forall i \in \{1, \dots, m\}$

Feature engineering

- ▶ **Blocks:** combination of two or more features
- ▶ **Counts:** number of samples for different feature values
- ▶ **Counts unique:** number of different values of one feature for fixed value of another feature
- ▶ **Likelihoods:** $\min_{\theta_t} L$, where $\theta_t = P(y_i | x_{ij} = t)$
- ▶ **Others**

Feature engineering (algorithm 1)



function SPLITBYROWS(M)

get partition of the matrix M into $\{M_t\}$ by rows such that each M_t has identical rows

Require:

$$J = \{j_1, \dots, j_s\} \subset \{1, \dots, n\}$$

$$K = \{k_1, \dots, k_q\} \subset \{1, \dots, n\} \setminus J$$

$$Z \leftarrow (x_{ij}) \subset X, i \in \{1, 2, \dots, m\}, j \in J$$

for $l = \{i_1, \dots, i_r\} \in \text{SPLITBYROWS}(Z)$

$$c_{i_1} = \dots = c_{i_r} = r$$

$$p_{i_1} = \dots = p_{i_r} = \frac{y_{i_1} + \dots + y_{i_r}}{r}$$

$$b_{i_1} = \dots = b_{i_r} = t$$

$$A_t = (x_{ik}) \subset X, i \in l, k \in K$$

$$T(A_t) = \text{size}(\text{SPLITBYROWS}(A_t))$$

$$u_{i_1} = \dots = u_{i_r} = T(A_t)$$

Feature engineering (algorithm 2)

function SPLITBYROWS(M)

get partition of the matrix M into $\{M_t\}$ by rows such that each M_t has identical rows

Require:

parameter $\alpha > 0$

$J \leftarrow (j_1, \dots, j_s) \subset \{1, \dots, n\}$

increasing sequence $V \leftarrow (v_1, \dots, v_l) \subset \{1, \dots, s\}, v_1 < s$

$f_i \leftarrow \frac{y_1 + \dots + y_m}{m}, \forall i \in \{1, \dots, m\}$

for $v \in V$ **do**

$J_v = \{j_1, \dots, j_v\}, Z = (x_{ij}) \subset X, i \in \{1, 2, \dots, m\}, j \in J_v$

for $l = \{i_1, \dots, i_r\} \in \text{SPLITBYROWS}(Z)$ **do**

$c_{i_1} = \dots = c_{i_r} = r$

$p_{i_1} = \dots = p_{i_r} = \frac{y_{i_1} + \dots + y_{i_r}}{r}$

$w = \sigma(-c + \alpha)$ - weight vector

$f_i = (1 - w_i) \cdot f_i + w_i \cdot p_i, \forall i \in \{1, \dots, m\}$

FTRL-Proximal model

Weight updates:

$$\begin{aligned}w_{i+1} &= \arg \min_w \left(\sum_{r=1}^i g_r \cdot w + \frac{1}{2} \sum_{r=1}^i \tau_r \|w - w_r\|_2^2 + \lambda_1 \|w\|_1 \right) = \\ &= \arg \min_w \left(w \cdot \sum_{r=1}^i (g_r - \tau_r w_r) + \frac{1}{2} \|w\|_2^2 \sum_{r=1}^i \tau_r + \lambda_1 \|w\|_1 + \text{const} \right),\end{aligned}$$

$$\text{where } \sum_{r=1}^i \tau_{rj} = \frac{\beta + \sqrt{\sum_{r=1}^i (g_{rj})^2}}{\alpha} + \lambda_2, \quad j \in \{1, \dots, N\},$$

$\lambda_1, \lambda_2, \alpha, \beta$ - parameters, $\tau_r = (\tau_{r1}, \dots, \tau_{rN})$ - learning rates,

g_r - gradient vector for the step r

FTRL-Proximal Batch model

Require: parameters $\alpha, \beta, \lambda_1, \lambda_2$

$z_j \leftarrow 0$ and $n_j \leftarrow 0, \forall j \in \{1, \dots, N\}$

for $i = 1$ to m **do**

receive sample vector x_i and let $J = \{j | x_{ij} \neq 0\}$

for $j \in J$ **do**

$$w_j = \begin{cases} 0 & \text{if } |z_j| \leq \lambda_1 \\ - \left(\frac{\beta + \sqrt{n_j}}{\alpha} + \lambda_2 \right)^{-1} (z_j - \text{sign}(z_j)\lambda_1) & \text{otherwise} \end{cases}$$

predict $\hat{y}_i = \sigma(x_i \cdot w)$ using the w_j and observe y_i

if $y_i \in \{0, 1\}$ **then**

for $j \in J$ **do**

$g_j = \hat{y}_i - y_i$ - gradient direction of loss w.r.t. w_j

$$\tau_j = \frac{1}{\alpha} \left(\sqrt{n_j + g_j^2} - \sqrt{n_j} \right)$$

$$z_j = z_j + g_j - \tau_j w_j$$

$$n_j = n_j + g_j^2$$

Performance

Description	Leaderboard score
dataset is sorted by <i>app id, site id, banner pos, count1, day, hour</i>	0.3844277
dataset is sorted by <i>app domain, site domain, count1, day, hour</i>	0.3835289
dataset is sorted by <i>person, day, hour</i>	0.3844345
dataset is sorted by <i>day, hour</i> with 1 iteration	0.3871982
dataset is sorted by <i>day, hour</i> with 2 iterations	0.3880423

Factorization Machine (FM)

Second-order polynomial regression:

$$\hat{y} = \sigma \left(\sum_{j=1}^{n-1} \sum_{k=j+1}^n w_{jk} x^j x^k \right)$$

Low rank approximation (FM):

$$\hat{y} = \sigma \left(\sum_{j=1}^{n-1} \sum_{k=j+1}^n (v_{j1}, \dots, v_{jH}) \cdot (v_{k1}, \dots, v_{kH}) x^j x^k \right)$$

H is a number of latent factors

Factorization Machine for categorical dataset

Assign set of latent factors for each pair level-feature:

$$\hat{y}_i = \sigma \left(\frac{2}{n} \sum_{j=1}^{n-1} \sum_{k=j+1}^n (w_{x_{ij}k1}, \dots, w_{x_{ij}kH}) \cdot (w_{x_{ik}j1}, \dots, w_{x_{ik}jH}) \right)$$

Add regularization term: $L_{reg} = L + \frac{1}{2} \lambda \|w\|^2$

The gradient direction:

$$g_{x_{ij}kh} = \frac{\partial L_{reg}}{\partial w_{x_{ij}kh}} = -\frac{2}{n} \cdot \frac{y_i e^{-y_i p_i}}{1 + e^{-y_i p_i}} \cdot w_{x_{ik}jh} + \lambda w_{x_{ij}kh}$$

Learning rate schedule: $\tau_{x_{ij}kh} = \tau_{x_{ij}kh} + (g_{x_{ij}kh})^2$

Weight update: $w_{x_{ij}kh} = w_{x_{ij}kh} - \alpha \cdot \sqrt{\tau_{x_{ij}kh}} \cdot g_{x_{ij}kh}$

Ensembling

Model	Description	Leaderboard score
ftrlb1	dataset is sorted by <i>app id, site id, banner pos, count1, day, hour</i>	0.3844277
ftrlb2	dataset is sorted by <i>app domain, site domain, count1, day, hour</i>	0.3835289
ftrlb3	dataset is sorted by <i>person, day, hour</i>	0.3844345
fm	factorization machine	0.3818004
ens	$fm^{0.6} \cdot ftrlb1^{0.1} \cdot ftrlb2^{0.2} \cdot ftrlb3^{0.1}$	0.3810447





Final results

Place	Team	Leaderboard score	Difference between the 1st place
1	4 Idiots	0.3791384	—
2	Owen	0.3803652	0.32%
3	Random Walker	0.3806351	0.40%
4	Julian de Wit	0.3810307	0.50%
5	Dmitry Efimov	0.3810447	0.50%
6	Marios and Abhishek	0.3828641	0.98%
7	Jose A. Guerrero	0.3829448	1.00%

Future work

- ▶ apply the batching idea to the Factorization Machine algorithm
- ▶ find a better sorting for the FTRL-Proximal Batch algorithm
- ▶ find an algorithm that can find better sorting without cross-validation procedure

References

-  H.Brendan McMahan et al. "Ad click prediction: a view from the trenches." *In KDD*, Chicago, Illinois, USA, August 2013.
-  Wei-Sheng Chin et al. "A learning-rate schedule for stochastic gradient methods to matrix factorization." *In PAKDD*, 2015.
-  Michael Jahrer et al. "Ensemble of collaborative filtering and feature engineered models for click through rate prediction." *In KDD Cup*, 2012
-  Steffen Rendle. "Social network and click-through prediction with factorization machines." *In KDD Cup*, 2012

Thank you! Questions???

Dmitry Efimov
defimov@aus.edu

