

Acquire Valued Shoppers Challenge

By Marios Michailidis

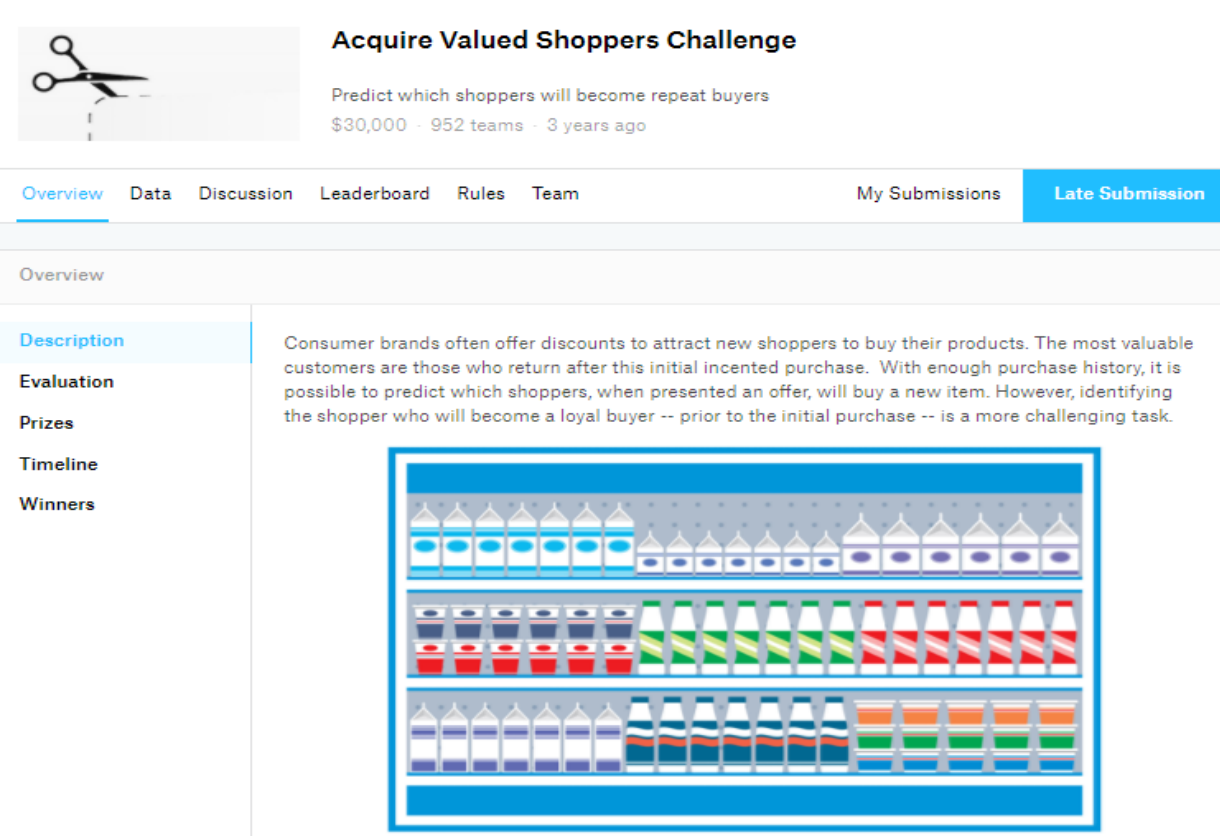


Background

- A recommenders' challenge
- Around 1,000 teams
- 1st place with Gert Jacobusse

First Place:

- Marios Michailidis - London, United Kingdom
- Gert Jacobusse - Goes, The Netherlands



Acquire Valued Shoppers Challenge

Predict which shoppers will become repeat buyers
\$30,000 · 952 teams · 3 years ago

[Overview](#) [Data](#) [Discussion](#) [Leaderboard](#) [Rules](#) [Team](#) [My Submissions](#) [Late Submission](#)

Overview

Description

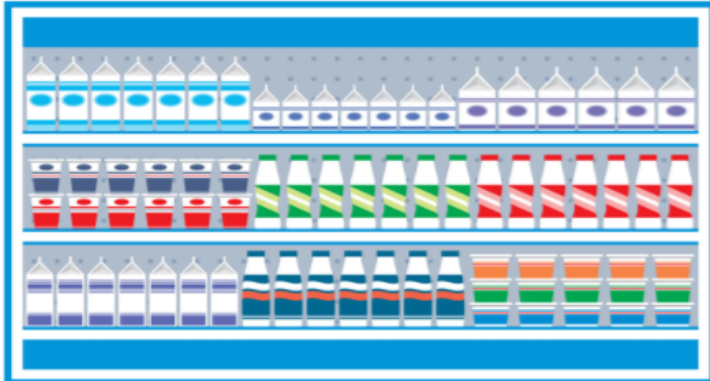
Evaluation

Prizes

Timeline

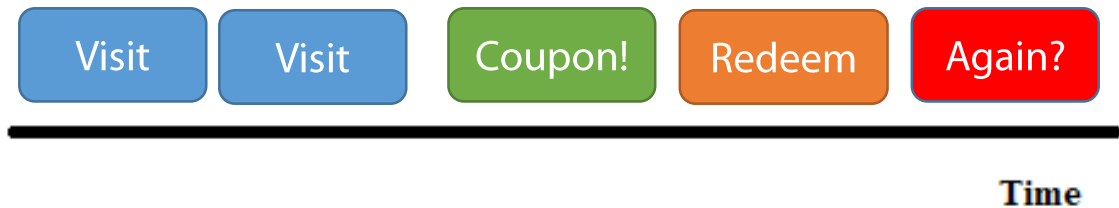
Winners

Consumer brands often offer discounts to attract new shoppers to buy their products. The most valuable customers are those who return after this initial incented purchase. With enough purchase history, it is possible to predict which shoppers, when presented an offer, will buy a new item. However, identifying the shopper who will become a loyal buyer -- prior to the initial purchase -- is a more challenging task.



Problem to solve

- 310,000 shoppers (160K in train and 150K in test)
- 350,000,000 transactions (for 1 year+ for each shopper)
- 37 offers
- No exactly products , but could infer a product is a combination of:
 - The **brand** a product belongs to
 - The **category**
 - The **company** that produced it
- Optimize AUC for whether the shopper will buy again



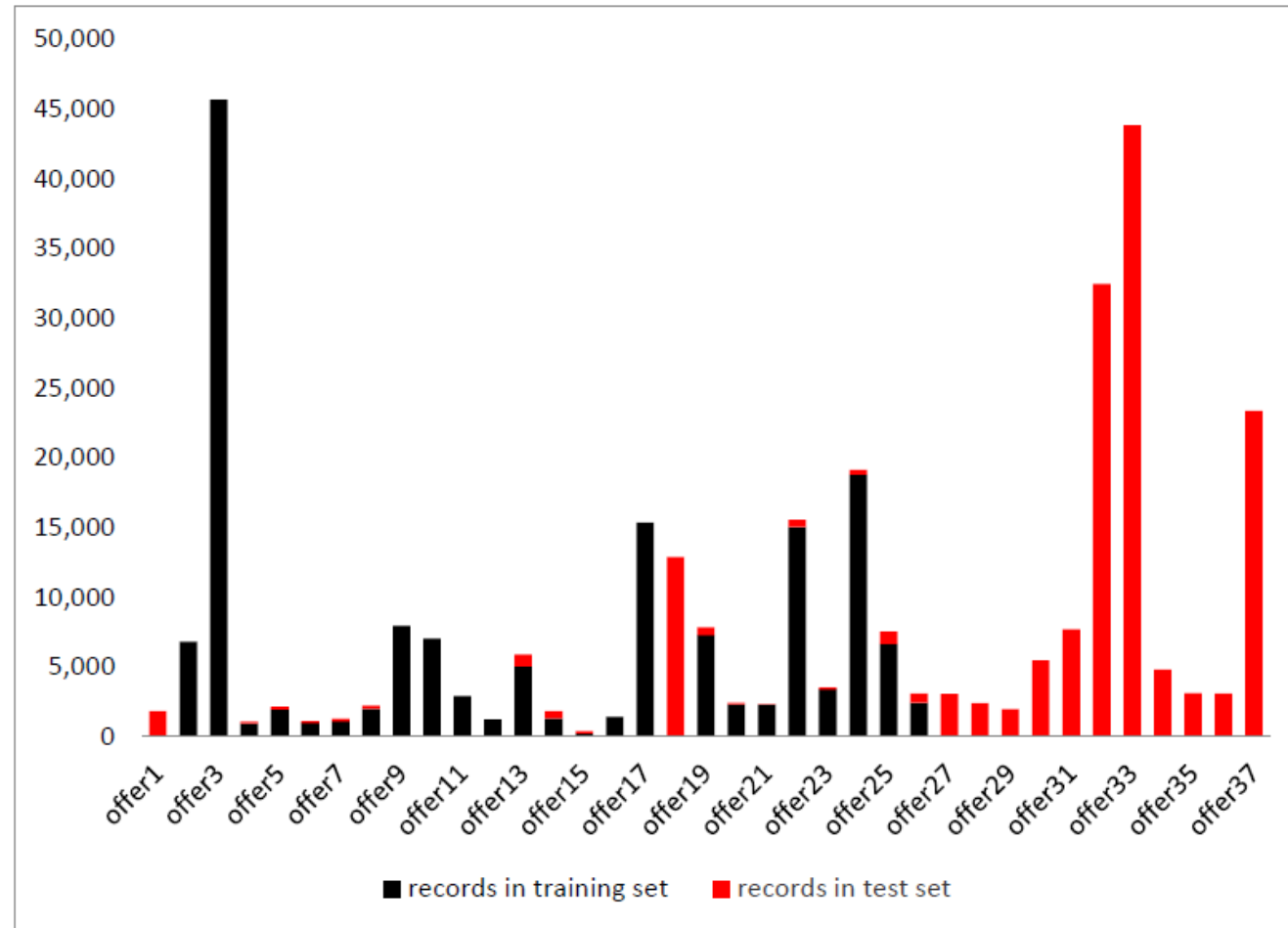
“Challenges” of this challenge

- Datasets were big
- You had to create the features yourself.
- Irregular testing environment because:
 - Train and test data had different customers
 - Train and test data had in general different offers



“Challenges” of this challenge

- Datasets w
- You had to
- Irregular te
 - Train and
 - Train and



“Challenges” of this challenge

- Datasets were big
- You had to create the features yourself.
- Irregular testing environment because:
 - Train and test data had different customers
 - Train and test data had in general different offers
 - Test data may be well in the future
 - Focused on acquisition. Limited history of customer and offer.
 - Offers' propensity to buy varies significantly



“Challenges” of this challenge

- Data
- You
- Irre

Offer	Propensity to buy
offer2	0.507
offer24	0.434
offer17	0.424
offer25	0.378
offer26	0.341
offer22	0.321
offer23	0.305
offer19	0.285
offer15	0.230
offer5	0.214
offer4	0.210



Handling Big data...

- With indexing
- Dataset already sorted by customer
- Create different file for every customer
- Create different file for every category, brand, company.
- All sorted by time



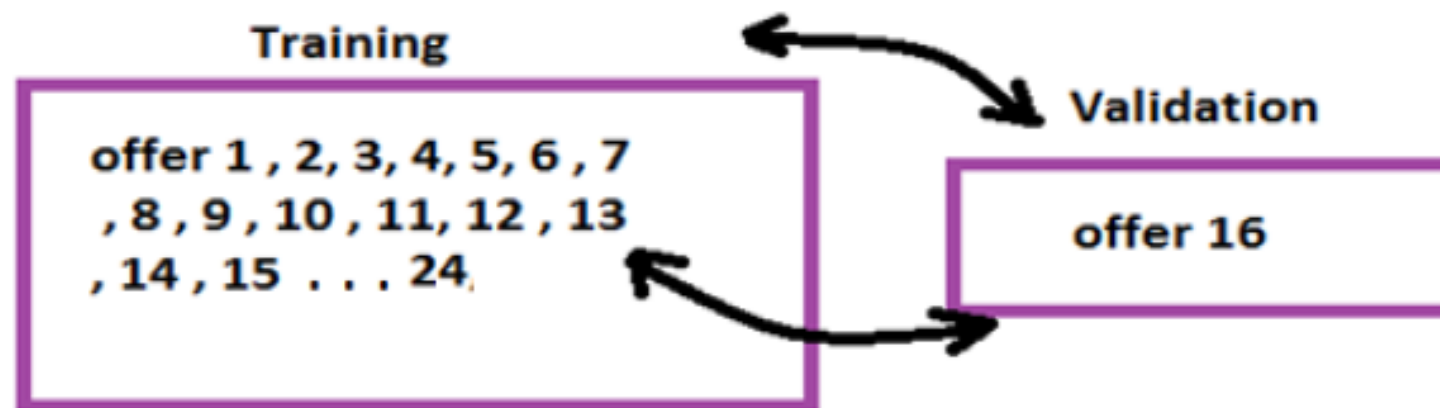
Handling irregularity

- Explored different cross validation.
- Stratified Kfold based on offer
- Leave-one-offer-out



Handling irregularity

- Explored different cross validation
- Stratified
- Leave-one



Handling irregularity

- Explored different cross validation.
- Stratified Kfold based on offer
- Leave-one-offer-out
- + concatenation



Handling irregularity

- Explored different cross validation.
- Stratified Kfold based on offer

prediction_offer_2	Target
0.91	1
0.81	1
0.77	1
0.65	1
0.52	1
0.46	0
0.34	0
0.23	0
0.2	0
0.05	0



Handling irregularity

- Explored different cross validation.
- Stratified Kfold based on offer

prediction_offer_2	Target
0.91	1
0.81	1
0.77	1
0.65	1
0.52	1
0.46	0
0.34	0
0.23	0
0.2	0
0.05	0

prediction_offer_4	Target
0.4	1
0.35	1
0.28	1
0.24	1
0.18	1
0.17	0
0.15	0
0.14	0
0.1	0
0.07	0



Handling irregularity

- Explored different
- Stratified Kfold ba
- Leave-one-offer-
- + concatenation

prediction_offer_2,4	Target
0.91	1
0.81	1
0.77	1
0.65	1
0.52	1
0.46	0
0.4	1
0.35	1
0.34	0
0.28	1
0.24	1
0.23	0
0.2	0
0.18	1
0.17	0
0.15	0
0.14	0
0.1	0
0.07	0
0.05	0



Handling irregularity

- Explored different cross validation.
- Stratified Kfold based on offer
- Leave-one-offer-out
- + concatenation

Validation schemas	AUC_CV	AUC_TEST
Stratified K-Fold (K=5)	0.683	0.579
Stratified K-Fold (K=10)	0.699	0.578
Stratified K-Fold (K=15)	0.712	0.576
Leave-one-offer validation	0.655	0.588
Leave-one-offer + concatenation	0.632	0.601



Strategies for recommenders

- Content-based
- Collaborative filtering
- Hybrid



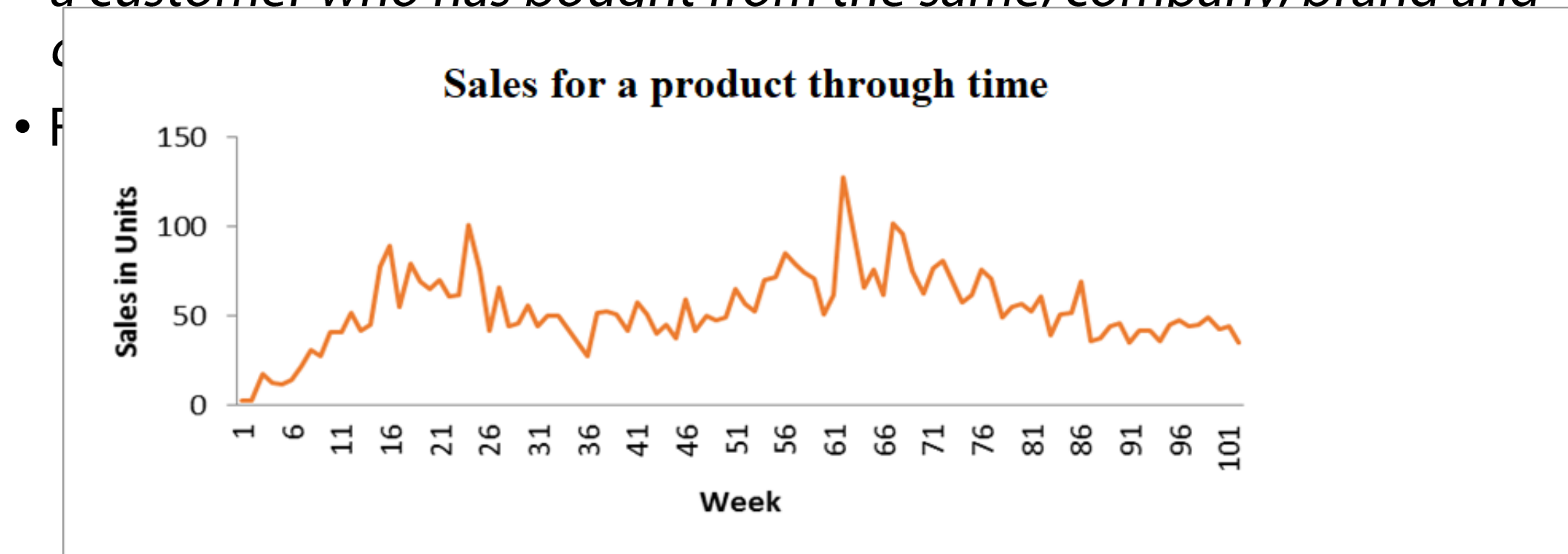
Content-based

- *a customer who has bought from the same, company, brand and category will have higher chance to buy the products once offered*
- Features exploiting **product hierarchy** versus **customer** and **time**



Content-based

- *a customer who has bought from the same, company, brand and*



Content-based

- *a customer who has bought from the same, company, brand and category will have higher chance to buy the products once offered*
- Features exploiting **product hierarchy** versus **customer** and **time**
- Time intervals were last 30,60,90,120,180 and 360 days
- Features selected through forward cross validation



Content-based

category_brand_30	Times bought the same category&brand in last 30 days
category_brand_60	Times bought the same category&brand in last 30 to 60 days
category_brand_90	Times bought the same category&brand in last 60 to 90 days
category_company_30	Times bought the same category&company in last 30 days
category_company_60	Times bought the same category&company in last 30 to 60 days
brand_company_30	Times bought the same brand&company in last 30 days
brand_company_60	Times bought the same brand&company in last 30 to 60 days
brand_company_90	Times bought the same brand&company in last 60 to 90 days
brand_company_120	Times bought the same brand&company in last 90 to 120 days
brand_company_180	Times bought the same brand&company in last 120 to 180 days
brand_company_360	Times bought the same brand&company in last 180 to 360 days
brand_company_over360	Times bought the same brand&company in more than 360 days
category_brand_company_30	Times bought the same category&brand&company in last 30 days
category_brand_company_60	Times bought the same category&brand&company in last 30 to 60 days



Content-based

- *a customer who has bought from the same, company, brand and category will have higher chance to buy the products once offered*
- Features exploiting **product hierarchy** versus **customer** and **time**
- Time intervals were last 30,60,90,120,180 and 360 days
- Features selected through forward cross validation
- Big values capped – missing values replaced with -1
- Modelling using **Ridge regression** on the actual repeat counts
- Scored **0.610+** in the test data



Collaborative Filtering

- *Would the customers have bought the product, had they not received the offer?*
- A model for every offer in train and test
- Target variable natural logarithm of the times a customer bought the product 90 days BEFORE receiving the coupon.



Collaborative Filtering

- *Would the customers have bought the product, had they not received the offer?*
- A model for every offer in train and test
- Target variable natural logarithm of the times a customer bought the product 90 days BEFORE receiving the coupon.
- Features based on users' activity
 - Counts of popular categories, brands, companies
 - **R**estricted **B**oltzmann **M**achines to summarize purchase activity on least popular.
 - Average amount purchase, total visits ,distinct brands/categories/companies
 - Total discounts/returns, visits in weekends, spend in weekends
- GBM (form sklearn) on the log of counts
- Scored 0.616+ on the test



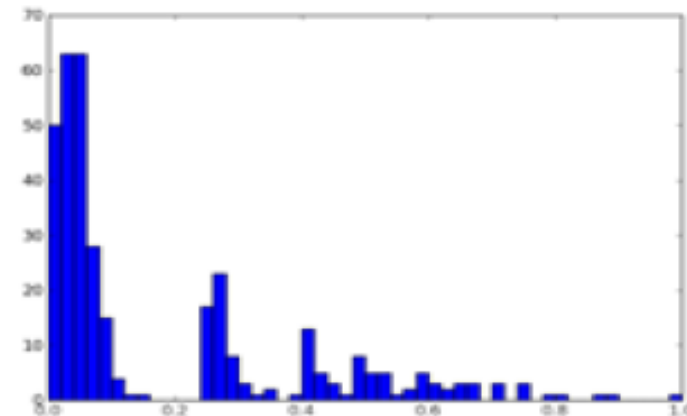
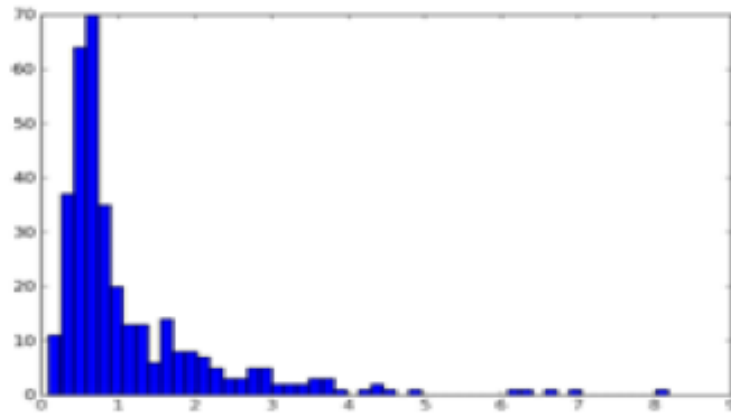
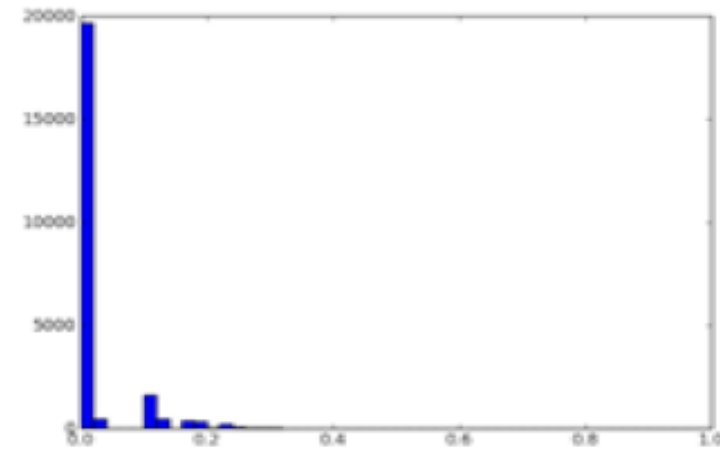
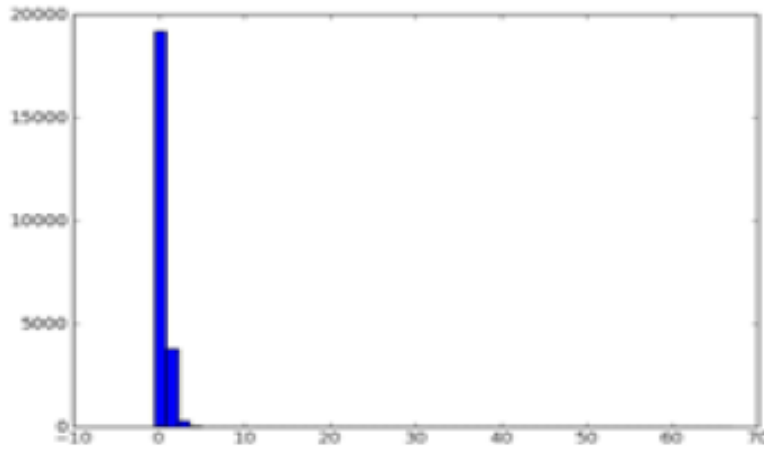
Combination

- Both models trained on different targets – tough to combine normally
- However irrespective of scores, distributions looked similar



Combination

- Both models trained on different targets – tough to combine normally
- However



Combination

- Both models trained on different targets – tough to combine normally
- However irrespective of scores, distributions looked similar
- Converted to ranks and take average

Strategies	AUC_TEST
Strategy 1: Content-based	0.610
Strategy 2: Collaborative filtering	0.616
$\text{strategy1}_{\text{rank}} \times 0.5 + \text{strategy2}_{\text{rank}} \times 0.5$	0.626

