jupyter Recommender ML algorithms Last Checkpoint: a few seconds ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help | Python [default]

CellToolbar

Building recommender system using machine learning algorithms.

author: bhavesh patel

There are many different models. Here are the few we will work on.

First and simplest: popularity model. This is where you see most popular items. It is not personalized.

Second: Recommendation based on past purchase history and other info like demographics etc.

Third: People who bought also bought. This is collaborative filtering model. This is where we build co-occurrence matrix

Fourth: Matrix factorization to find predict hidden values (e.g. people never bought that item). This works as long as somebody else bought the item.

Measuring performance of different models.

recall = number of items liked and recommended / Total number of items liked

Precision = number of items liked and recommended / total number of items recommended

Ideally, you want both recall and precision to be 1 (100%)

In reality, precision goes down as you try to recall more items as you have fewer data points.

We can compare recommender system by finding out AUC (Area under curve). More the better.

In [1]:
```
import graphlab
```

In [2]:
```
# Limit number of worker processes. This preserves system memory, which prevents hosted notebooks from crashing.
graphlab.set_runtime_config('GRAPHLAB_DEFAULT_NUM_PYLAMBDA_WORKERS', 4)
```

```
[INFO] graphlab.cython.cy_server: GraphLab Create v2.1 started. Logging: /tmp/graphlab_server_1479956553.log
This non-commercial license of GraphLab Create for academic use is assigned to bhaveshhk8@gmail.com and will expire o
n October 17, 2017.
```

In [5]:
```
#Load the data.

song_data = graphlab.SFrame('song_data.gl/')
```

In [7]:
```
# review data.

song_data.head()
```

Out[7]:

user_id	song_id	listen_count	title	artist
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SOAKIMP12A8C130995	1	The Cove	Jack Johnson
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SOBBMDR12A8C13253B	2	Entre Dos Aguas	Paco De Lucia
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SOBXHDL12A81C204C0	1	Stronger	Kanye West
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SOBYHAJ12A6701BF1D	1	Constellations	Jack Johnson
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SODACBL12A8C13C273	1	Learn To Fly	Foo Fighters
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SODDNQT12A6D4F5F7E	5	Apuesta Por El Rock 'N' Roll ...	Héroes del Silencio
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SODXRTY12AB0180F3B	1	Paper Gangsta	Lady GaGa
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SOFGUAY12AB017B0A8	1	Stacked Actors	Foo Fighters
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SOFRQTD12A81C233C0	1	Sehr kosmisch	Harmonia
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	SOHQWYZ12A6D4FA701	1	Heaven's gonna burn your eyes ...	Thievery Corporation feat. Emiliana Torrini ...

song
The Cove - Jack Johnson

Entre Dos Aguas - Paco De Lucia ...
Stronger - Kanye West
Constellations - Jack Johnson ...
Learn To Fly - Foo Fighters ...
Apuesta Por El Rock 'N' Roll - Héroes del ...
Paper Gangsta - Lady GaGa
Stacked Actors - Foo Fighters ...
Sehr kosmisch - Harmonia
Heaven's gonna burn your eyes - Thievery ...

[10 rows x 6 columns]

In [8]:
```
# make graph local.
graphlab.canvas.set_target('ipynb')
```

In [9]:
```
song_data.show()
```

user_id

dtype:	str
num_unique (est.):	66,019
num_undefined:	0

frequent items:

No values appear with ≥ 0.01% occurrence.

song_id

dtype:	str
num_unique (est.):	9,971
num_undefined:	0

frequent items:

SOFRQTD12A81C233C0
SOAUWYT12A81C206F1
SOBONKR12A58A7A7E0
SOAXGDH12A8C13F8A1
SOSXLTC12AF72A7F54
SOEGIYH12A6D4FC0E3
SONYKOW12AB01849...
SOFLJQZ12A6D4FADA6
SOLFXKT12AB017E3E0
SODJWHY12A8C142C...
SOUVTSM12AC468F6A7
SOUSMXX12AB0185C24

listen_count

dtype:	int
num_unique (est.):	276
num_undefined:	0
min:	1
max:	920
median:	1
mean:	3.291
std:	7.203

distribution of values:

title

dtype:	str
num_unique (est.):	9,540
num_undefined:	0

frequent items:

Sehr kosmisch
Undo
You're The One
Dog Days Are Over ...
Revelry
Horn Concerto No. ...
Secrets
Tive Sim
Fireflies
Hey_ Soul Sister
Drop The World
OMG

artist

dtype:	str
num_unique (est.):	3,371
num_undefined:	0

frequent items:

Coldplay
Florence + The ...
Kings Of Leon
Justin Bieber
The Black Keys
Jack Johnson
Train
Eminem
OneRepublic
Radiohead
Muse
Daft Punk

In [10]:
```
# number of records.
len(song_data)
```

Out[10]: 1116609

In [11]:
```
# first create training and test data set.
train_data, test_data = song_data.random_split(0.8, seed=0)
```

First recommendation model: Populatiry model

In [12]:
```
popularity_model = graphlab.popularity_recommender.create(train_data,
                                                          user_id='user_id',
                                                          item_id='song')
```

```
Recsys training: model = popularity
Warning: Ignoring columns song_id, listen_count, title, artist;
    To use one of these as a target column, set target =
    and use a method that allows the use of a target.
Preparing data set.
    Data has 893580 observations with 66085 users and 9952 items.
    Data prepared in: 1.55758s
893580 observations to process; with 9952 unique items.
```

In [18]:
```
# let's check for the first user.
```

```
popularity_model.recommend(users=[song_data['user_id'][0]])
```

Out[18]:

user_id	song	score	rank
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Undo - Björk	4227.0	1
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	You're The One - Dwight Yoakam ...	3781.0	2
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Dog Days Are Over (Radio Edit) - Florence + The ...	3633.0	3
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Revelry - Kings Of Leon	3527.0	4
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Horn Concerto No. 4 in E flat K495: II. Romance ...	3161.0	5
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Secrets - OneRepublic	3148.0	6
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Hey_ Soul Sister - Train	2538.0	7
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Fireflies - Charttraxx Karaoke ...	2532.0	8
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Tive Sim - Cartola	2521.0	9
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Drop The World - Lil Wayne / Eminem ...	2053.0	10

[10 rows x 4 columns]

In [25]:
```
# now let's for 1000th user.

popularity_model.recommend(users=[song_data['user_id'][1000]])
```

Out[25]:

user_id	song	score	rank
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Sehr kosmisch - Harmonia	4754.0	1
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Undo - Björk	4227.0	2
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	You're The One - Dwight Yoakam ...	3781.0	3
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Dog Days Are Over (Radio Edit) - Florence + The ...	3633.0	4
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Revelry - Kings Of Leon	3527.0	5
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Horn Concerto No. 4 in E flat K495: II. Romance ...	3161.0	6
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Secrets - OneRepublic	3148.0	7
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Hey_ Soul Sister - Train	2538.0	8
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Fireflies - Charttraxx Karaoke ...	2532.0	9
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Tive Sim - Cartola	2521.0	10

[10 rows x 4 columns]

In [20]:
```
# as you can see above, it recommends same thing for both/any users.
# That's because it uses popularity model, not personalized model.
```

Second recommendation model: Based on past behaviour

In [26]:
```
past_behaviour_model = graphlab.item_similarity_recommender.create(train_data,
                                                                   user_id='user_id',
                                                                   item_id='song')
```

```
Recsys training: model = item_similarity

Warning: Ignoring columns song_id, listen_count, title, artist;

    To use one of these as a target column, set target =

    and use a method that allows the use of a target.

Preparing data set.

    Data has 893580 observations with 66085 users and 9952 items.

    Data prepared in: 1.36127s

Training model from provided data.

Gathering per-item and per-user statistics.
```

```
+--------------------------------+------------+
| Elapsed Time (Item Statistics) | % Complete |
+--------------------------------+------------+
| 1.586ms                        | 1.5        |
| 61.882ms                       | 100        |
+--------------------------------+------------+
Setting up lookup tables.
Processing data in one pass using dense lookup tables.
+-------------------------------------+------------------+-----------------+
| Elapsed Time (Constructing Lookups) | Total % Complete | Items Processed |
+-------------------------------------+------------------+-----------------+
| 390.942ms                           | 0                | 0               |
| 2.22s                               | 100              | 9952            |
+-------------------------------------+------------------+-----------------+
Finalizing lookup tables.
Generating candidate set for working with new users.
Finished training in 2.36233s
```

In [27]:

```
# now let's see how this fared against our previous model for the same users.
past_behaviour_model.recommend(users=[song_data['user_id'][0]])
```

Out[27]:

user_id	song	score	rank
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Meadowlarks - Fleet Foxes	0.0248072429707	1
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Quiet Houses - Fleet Foxes ...	0.0240329645182	2
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Heard Them Stirring - Fleet Foxes ...	0.0203885561542	3
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Tiger Mountain Peasant Song - Fleet Foxes ...	0.0199806752958	4
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Your Protector - Fleet Foxes ...	0.0193978893129	5
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Oliver James - Fleet Foxes ...	0.0190611293441	6
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Great Indoors - John Mayer ...	0.0149489750988	7
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Innocent Son - Fleet Foxes ...	0.0148925859677	8
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	White Winter Hymnal - Fleet Foxes ...	0.0148194040123	9
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	City Love - John Mayer	0.0138473055865	10

[10 rows x 4 columns]

In [28]:

```
# for another users - 1000th user.
past_behaviour_model.recommend(users=[song_data['user_id'][1000]])
```

Out[28]:

user_id	song	score	rank
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Lights & Music - Cut Copy	0.00933748483658	1
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Oh! - Boys Noize	0.00898901266711	2
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Waters Of Nazareth (album version) - Justice ...	0.00894576098238	3
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Strangers In The Wind - Cut Copy ...	0.00881623370307	4
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Auto-Dub - Skream	0.00863063548292	5
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Thrills - LCD Soundsystem	0.00838310803686	6
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Clock - Simian Mobile Disco ...	0.00832954687732	7
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	On Repeat - LCD Soundsystem ...	0.00831711079393	8
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Lava Lava - Boys Noize	0.00791249317782	9

20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Shine Shine - Boys Noize	0.00781313436372	10

[10 rows x 4 columns]

Third recommendation model: People who bought also bought

In [32]:
```
also_bought_model=graphlab.recommender.ranking_factorization_recommender.create(train_data,
                                                     user_id='user_id',
                                                     item_id='song'
                                                     )
```

```
Recsys training: model = ranking_factorization_recommender

Preparing data set.

    Data has 893580 observations with 66085 users and 9952 items.

    Data prepared in: 2.88614s

Training ranking_factorization_recommender for recommendations.

+--------------------------------+--------------------------------------------------+----------+

| Parameter                      | Description                                      | Value    |

+--------------------------------+--------------------------------------------------+----------+

| num_factors                    | Factor Dimension                                 | 32       |

| regularization                 | L2 Regularization on Factors                     | 1e-09    |

| solver                         | Solver used for training                         | adagrad  |
```

In [33]:
```
# now let's see how this fared against our previous model for the same users.

also_bought_model.recommend(users=[song_data['user_id'][0]])
```

Out[33]:

user_id	song	score	rank
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Invalid - Tub Ring	0.182793350734	1
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Drop The World - Lil Wayne / Eminem ...	0.179032795042	2
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Undo - Björk	0.177632327762	3
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Can't Help But Wait (Album Version) - Trey ...	0.171274764003	4
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Horn Concerto No. 4 in E flat K495: II. Romance ...	0.170224094969	5
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Ain't Misbehavin - Sam Cooke ...	0.168632497922	6
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Catch You Baby (Steve Pitron & Max Sanna Radio ...	0.164428076157	7
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Paradise & Dreams - Darren Styles ...	0.161302347033	8
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Lucky (Album Version) - Jason Mraz & Colbie ...	0.159779645527	9
b80344d063b5ccb3212f76538 f3d9e43d87dca9e ...	Who Can Compare - Foolish Things ...	0.158943212951	10

[10 rows x 4 columns]

In [34]:
```
# for another users - 1000th user.

also_bought_model.recommend(users=[song_data['user_id'][1000]])
```

Out[34]:

user_id	song	score	rank
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Sehr kosmisch - Harmonia	0.175580286274	1
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Can't Help But Wait (Album Version) - Trey ...	0.172968190983	2
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	I'm On A Boat - The Lonely Island / T-Pain ...	0.170508792145	3
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Drop The World - Lil Wayne / Eminem ...	0.170295738144	4
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Paradise & Dreams - Darren Styles ...	0.16467718148	5
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Make Love To Your Mind - Bill Withers ...	0.16379823765	6
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Invalid - Tub Ring	0.162964197315	7
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Recado Falado (Metrô Da Saudade) - Alceu Vale ...	0.16259148151	8

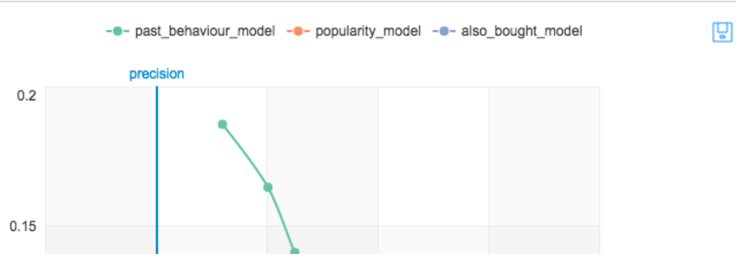
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Catch You Baby (Steve Pitron & Max Sanna Radio ...	0.161659921738	9
20d0638c7ada27ac12346b0ed 5ab99b39524291d ...	Undo - Björk	0.161299519582	10

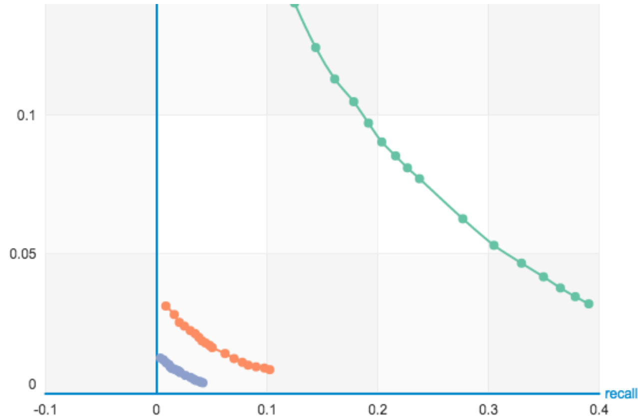
[10 rows x 4 columns]

In [35]:
```
# now let's compare the models.

model_performance = graphlab.compare(test_data, [popularity_model, past_behaviour_model, also_bought_model], user_sampl
```

```
compare_models: using 2931 users to estimate model performance
PROGRESS: Evaluate model M0

recommendations finished on 1000/2931 queries. users per second: 10861.5

recommendations finished on 2000/2931 queries. users per second: 12495.1


Precision and recall summary statistics by cutoff
+--------+-----------------+------------------+
| cutoff |  mean_precision |   mean_recall    |
+--------+-----------------+------------------+
|   1    | 0.0310474240873 | 0.00864231725286 |
|   2    | 0.0279767997271 |  0.016270962727  |
|   3    | 0.0251336290231 | 0.0208536161325  |
|   4    | 0.0237973387922 | 0.0254220609246  |
|   5    |  0.022176731491 | 0.0307244388985  |
|   6    | 0.0210963266235 | 0.0351415114578  |
|   7    | 0.0197397280304 | 0.0384487959319  |
|   8    | 0.0184663937223 | 0.0412952159739  |
|   9    |  0.01770347625  | 0.0445078634803  |
|   10   | 0.0168543159331 | 0.0480589959976  |
+--------+-----------------+------------------+
[10 rows x 3 columns]

PROGRESS: Evaluate model M1

recommendations finished on 1000/2931 queries. users per second: 9017.78

recommendations finished on 2000/2931 queries. users per second: 10091.7


Precision and recall summary statistics by cutoff
+--------+-----------------+-----------------+
| cutoff |  mean_precision |   mean_recall   |
+--------+-----------------+-----------------+
|   1    |  0.186625725009 | 0.0596278086938 |
|   2    |  0.163937222791 |  0.100715287499 |
|   3    |  0.140452632776 |  0.124928678358 |
|   4    |  0.124360286592 |  0.144016652346 |
|   5    |  0.112998976459 |  0.161185270383 |
|   6    |  0.104799272148 |  0.178368400002 |
|   7    | 0.0970902178681 |  0.191659779005 |
|   8    | 0.0902848857045 |  0.203655789836 |
|   9    | 0.0852572121764 |  0.216180337392 |
|   10   | 0.0809280109178 |  0.226882091095 |
+--------+-----------------+-----------------+
[10 rows x 3 columns]

PROGRESS: Evaluate model M2

recommendations finished on 1000/2931 queries. users per second: 1444.17

recommendations finished on 2000/2931 queries. users per second: 1472.62


Precision and recall summary statistics by cutoff
+--------+------------------+------------------+
| cutoff |  mean_precision  |   mean_recall    |
+--------+------------------+------------------+
|   1    | 0.0122824974411  | 0.00356601300818 |
|   2    | 0.0116001364722  | 0.00653617867026 |
|   3    | 0.0105765950188  | 0.00902808240321 |
|   4    | 0.00989423404981 | 0.0115177522163  |
|   5    | 0.00900716479017 | 0.0127300052198  |
|   6    | 0.00847264869783 |  0.014323940872  |
|   7    | 0.00828581176585 | 0.0161457278269  |
|   8    | 0.00793244626407 | 0.0181685760882  |
|   9    | 0.0077334243148  | 0.0196126360277  |
|   10   | 0.00736949846469 | 0.0209222546096  |
+--------+------------------+------------------+
[10 rows x 3 columns]

Model compare metric: precision_recall
```

In [36]:
```
graphlab.show_comparison(model_performance,[popularity_model, past_behaviour_model, also_bought_model])
```





In [37]:
```
# Conclusion:

# As expected, populariy model did not perform well compare to personalized model based on past behaviour.
# In this case of songs, past behaviour is a better predictor than watching other users with also bought.
# Also bought is more useful when you have missing data.  For example, when we want to recommend new items
# to a user.
```