SASによるインメモリ分散並列処理 レコメンドプロシジャ入門

庄子 楽
SAS Institute Japan 株式会社

Introduction: In-memory & Recommendation

Gaku Shoji
SAS Institute Japan Ltd.

要旨:

インメモリ解析環境 SAS In-memory Statistics(略称 IMSTAT)、および、レコメンド専用プロシジャついてご紹介いたします。

キーワード:

インメモリ、レコメンド、In-memory analytics, Recommendation

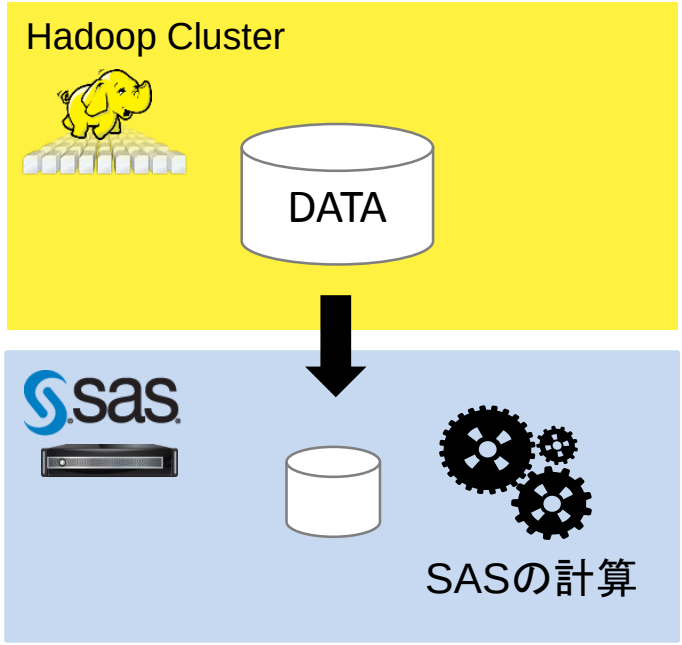
SAS® In-Memory Statistics

SASにおけるインメモリ解析環境 (略称 IMSTAT)

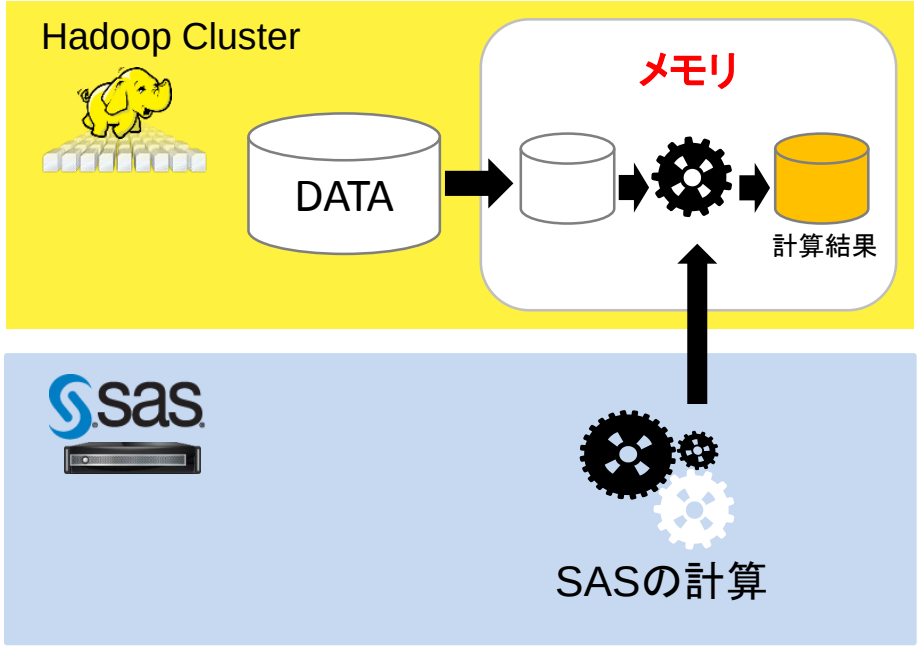
処理イメージ

Hadoop Cluster上のメモリへデータをロードした後、インメモリでのSASの統計解析・機械学習処理を可能とします。

データをSAS Serverに移動

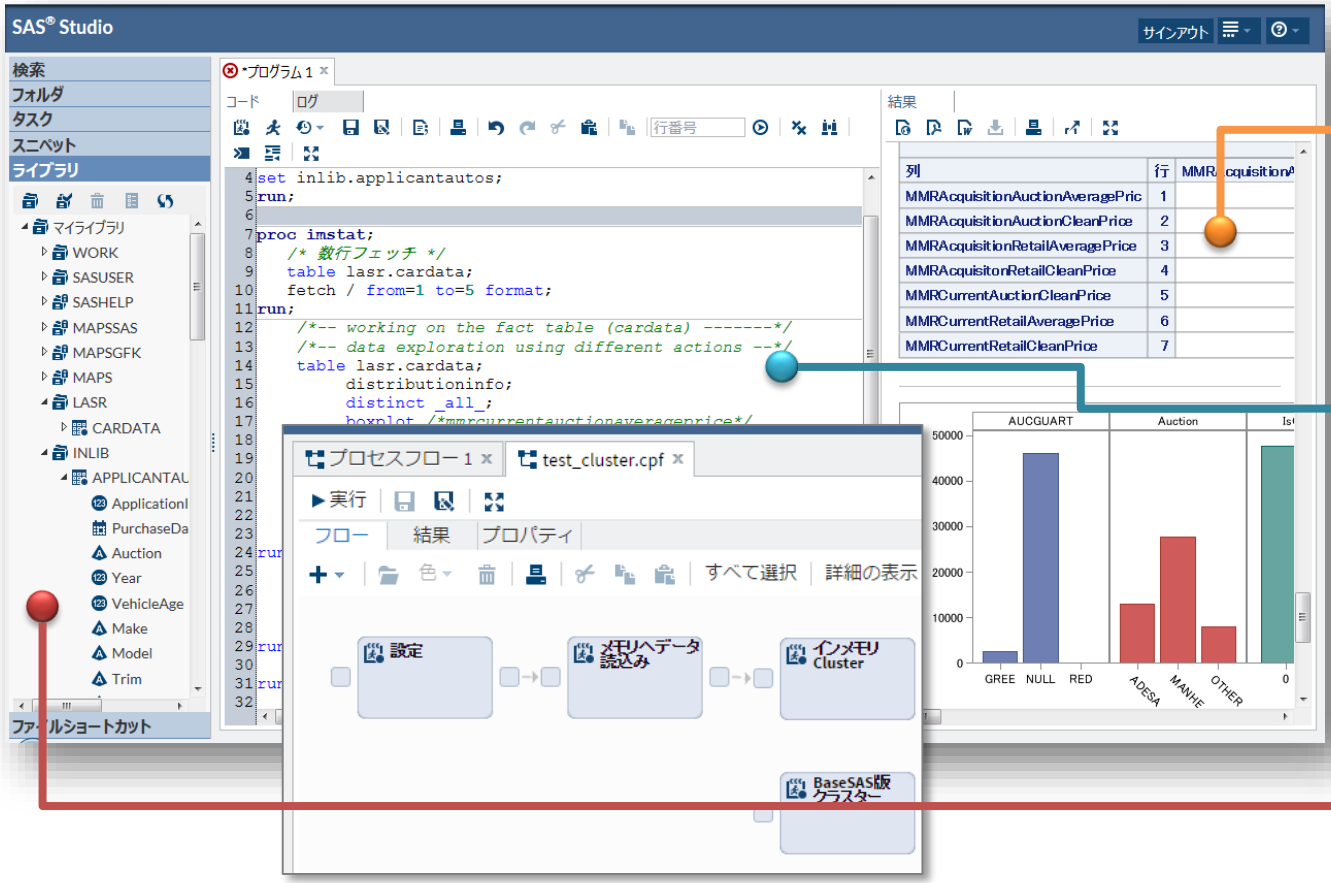


SAS® In-Memory Statistics



ユーザインタフェース

WEBブラウザで使う SAS® Studio と インメモリ用のSASコード



結果

結果のビジュアル化
表とグラフ
HTML, PDF, RTF形式

コード・エディタ

オンラインエディタ。
ビジュアルプログラミング。自動補完など。

アセット

データ一覧
共有コード一覧
分析テンプレート

SAS® In-Memory Statisticsで出来ること

データ加工

- 透過的なデータアクセス
- ライブラリ、データ定義
- Scheme
- Update、Append
- Set
- Filter
- Where句 処理
- Group By 処理
- Distinct
- Transform
  （ビン化、外れ値等）
  など

記述統計

- 要約統計量 (summary)
- 集計表
- クロス集計表
- 相関係数
- 箱ひげ図(Box Plot)
- ヒストグラム
- パーセンタイル
  など

SAS® In-Memory Statisticsで出来ること

解析手法

- 回帰
- ロジスティック回帰
- 一般化線形モデル
- ディシジョンツリー
- ランダムフォレスト
- ニューラルネットワーク
- クラスタリング（K-means, DBSCAN）
- 特異値分解（SVD）
- 時系列予測
- コミュニティ検出
- 非線形最適化

レコメンド

- 協調フィルタリング (KNN)
- SVD
- アソシエーションルール
- Slopeone
- クラスタリング
- コールドスタート対策
- アンサンブル

テキスト解析*

- 形態素解析
- クレンジング（ステミング、辞書）
- 語、文書の頻度
- 特異値分解（SVD）
- エンティティ抽出、トピック生成

*日本語未対応（対応を検討中）

インメモリ用SASコード　例) メモリへロード

```
libname LSARLIB sasiola port=番号 host=ホスト名;
```

インメモリライブラリ

```
data LSARLIB. PRDSALE;
    set sashelp.prdsale;
run;
```

メモリにロード

※インメモリライブラリのイメージを掴みやすいので、例として記載しましたが、
通常は、proc lasr add、proc hpds2、といった適したプロシジャを利用します。
※LSARLIB部分は任意の文字列でOK。

インメモリ用SASコード 例) 集計

```
proc imstat;
  table      LSARLIB.PRDSALE;
  summary    ACTUAL / groupby = PRODUCT;
run;
quit;
```

インメモリ処理を記述

例) PRODUCT毎にACTUALの基本統計量を算出。

インメモリ用SASコード　例) クラスタリング

```
proc imstat;
  table     LSARLIB.IRIS;
  cluster    SepalLength SepalWidth ・・・
              / numclus=3;
run;
quit;
```

インメモリ処理を記述

例) IRISの観測データを、k-meansで3つのクラスターに分類。

例) クラスタリング結果 (画面のイメージだけ)

*test_for_UserForum.cpf

test_for_UserForum > インメモリ Cluster

コード ログ ノード

行番号

```
1
2 /* インメモリのクラスタリング処理 */
3 proc imstat;
4     table    LSARLIB.IRIS;
5     cluster  petal: sepal:
6              / numclus=3 temptable freq = Species;
7
8 run;
9 quit;
10
11
```

行 11, 列 1

結果

IMSTAT プロシジャ

K-Means クラスターの情報: 4 変数 (テーブル HPS.IRIS)

クラスター ID	OBS 数	STD の平均平方根	シードからオブザベーションまでの最大距離	シードからオブザベーションまでの最小距離	クラスター内の平方和	最も近いクラスター	クラスター重心間の距離
0	50	2.7803	12.4803	0.6618	1515.10	2	33.4949
1	39	4.0890	15.5156	2.3945	2541.38	2	17.8842
2	61	3.9943	16.4680	2.3571	3829.08	1	17.8842

変数 Species (テーブル HPS.IRIS) の度数クラスターの情報

クラスター ID	水準	フォーマット済み	度数
0	0	Setosa	50
0	1	Versicolor	0
0	2	Virginica	0
1	0	Setosa	0
1	1	Versicolor	3
1	2	Virginica	36
2	0	Setosa	0
2	1	Versicolor	47
2	2	Virginica	14

一時テーブルの情報: テーブル HPS.IRIS

Proc Recommend

レコメンド計算専用プロシジャ
(SAS® In-Memory Statistics専用)

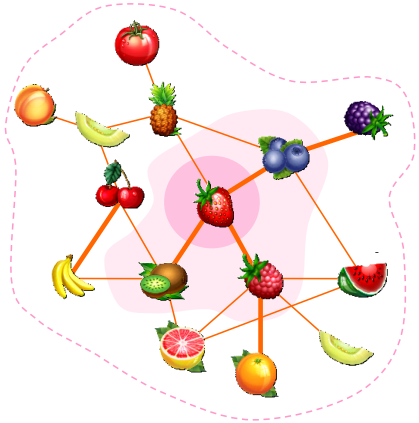
Proc Recommend

データ準備～計算～計算後の整備が、1プロシジャに包含。

1.データの準備
データを指定するだけ

- 履歴データ
- 顧客マスタ
- 商品マスタ

2.おすすめ計算
計算メソッドを選ぶ



3.計算後の整備
おすすめの出力&既知排除

顧客ID	お薦め	スコア
001	item66	2.0
001	item80	1.8
001	item120	1.2
001	item2	0.8

1.データの準備 “3つ用意して指定するだけ”

履歴データ

顧客ID	アイテムID	Rating(※)
000001	0014	5
000001	0012	2
000001	0002	1
000002	0043	3
000002	0027	1

顧客マスタ

顧客ID	gender	age
000001	M	25
000002	M	34
000003	F	46
000004	F	50
000005	M	26

商品マスタ

アイテムID	item_name
0043	ベルト
0044	マフラー
0045	下着
0046	革靴
0047	高級時計

※顧客からの評価データ（Ratingなど）がある場合は利用し、
評価データが無い場合は、回数や金額、TF-IDF値等様々な指標での代用を推奨します。

2.おすすめ計算

Content Based

アイテムの属性情報（ジャンル、俳優、監督、紹介文章等）から関連するものを判定し、お薦めを決定する。
例）閲覧した作品と、同じ俳優の作品をお薦めするなど

➡ IMSTATのデータ加工、SVD、テキスト解析等で

Community Based （協調フィルタリング）

ユーザのアイテムへの評価データを解析し、ユーザ間やアイテム間の類似性を測り、似たユーザの評価に基づきお薦めを決定。
※ユーザベース / アイテムベース 協調フィルタリング

➡ IMSTAT / Proc Recommend で

2.おすすめ計算 (Aさんへのおすすめを単純化して考えると)

item	種類	監督	紹介文章
1	Action	AAA	カーチェイス・・・
3	SF	BBB	2100年 カー・・・
4	Action	AAA	カンフー・・・

Content Based

Aが見たitem1に似たitemは？
item4が監督が同じ！
item3は文章が似てる！

顧客	Item1	item2	item3	item4
A	5	2	-	-
B	4	3	4	1
C	-	-	2	5

Community Based

Aと似た利用者は誰か？
Bが似てる！
Bで評価のよいItem3は？

2.おすすめ計算　6手法 + Ensemble

KNN (K-Nearest-Neighbor)	ユーザ or アイテムベース協調フィルタリング (類似度指標は、Pearson、調整済みCosine類似度等)
Slope one	アイテムベース協調フィルタリングの一種 シンプルな回帰ベース
SVD (Singular value decomposition) (ALS / L-BFGS)	ユーザ - アイテムMatrixの特異値分解に基づき、 ユーザ - アイテム間の近接度を測る
Association Rule	アイテムの併買に基づく
Average	コールドスタート対策用 ※データが少ないユーザ向けへのお薦め計算
Clustering (K-means, DBSCAN)	ユーザ or アイテムをプロファイル情報でクラスタリングし、クラスタ毎にKNNの計算を実施
Ensemble (Linear combination)	各種手法の結果を合成するタイプ

3.計算後の整備

個々人へのおすすめが計算された後、以下のような整備も行われます。

既知アイテムの除去

出力されるおすすめアイテムには、既知のアイテムが除かれて出力されます。

顧客	おすすめ	スコア
~~A~~	~~Item1~~	既知
A	Item66	2.0
A	Item80	1.8

おすすめ不足時の補完

1人あたり5個のおすすめを出したい時、おすすめアイテムが5個に満たないケースも生じますが、補完されて出力されます。

顧客	おすすめ	スコア
A	Item66	2.0
A	Item66	1.8
A	Item80	1.2

補完

顧客	おすすめ	スコア
B	Item 5	2.0
B	Item 10	1.8
B	Item 20	1.2
B	Item 22	1.0
B	Item 60	0.9

Proc Recommend 構文例

データ指定

```
Proc recommend     port  = 10031 recom = LASR.RecoDemo1;
add        LASR.RecoDemo1          / item= itemid  user = userid  rating = _SUM_ ;
addtable LASR.TRANSACTION          / recom = LASR.RecoDemo1 type = rating;
addtable LASR.MASTER_ITEM          / recom = LASR.RecoDemo1 type = item;
addtable LASR.MASTER_USER          / recom = LASR.RecoDemo1 type = user;
```

計算

```
/* 計算メソッドを指定   KNN  SVD */
method knn / label = "knn" k = 20 positive similarity = pc seed = 1234;
method svd / factors = 100 label = "svd" technique = als;
/* 計算メソッドを指定   KNNとSVDのアンサンブル */
method ensemble / methods =("svd","knn") label = "ensemble";
```

出力

```
/* おすすめ出力   顧客“A”に対するおすすめ5つ */
predict / method =  ensemble  label="ensemble " Num = 5 userlist  = (“A");
run;
```

結果例

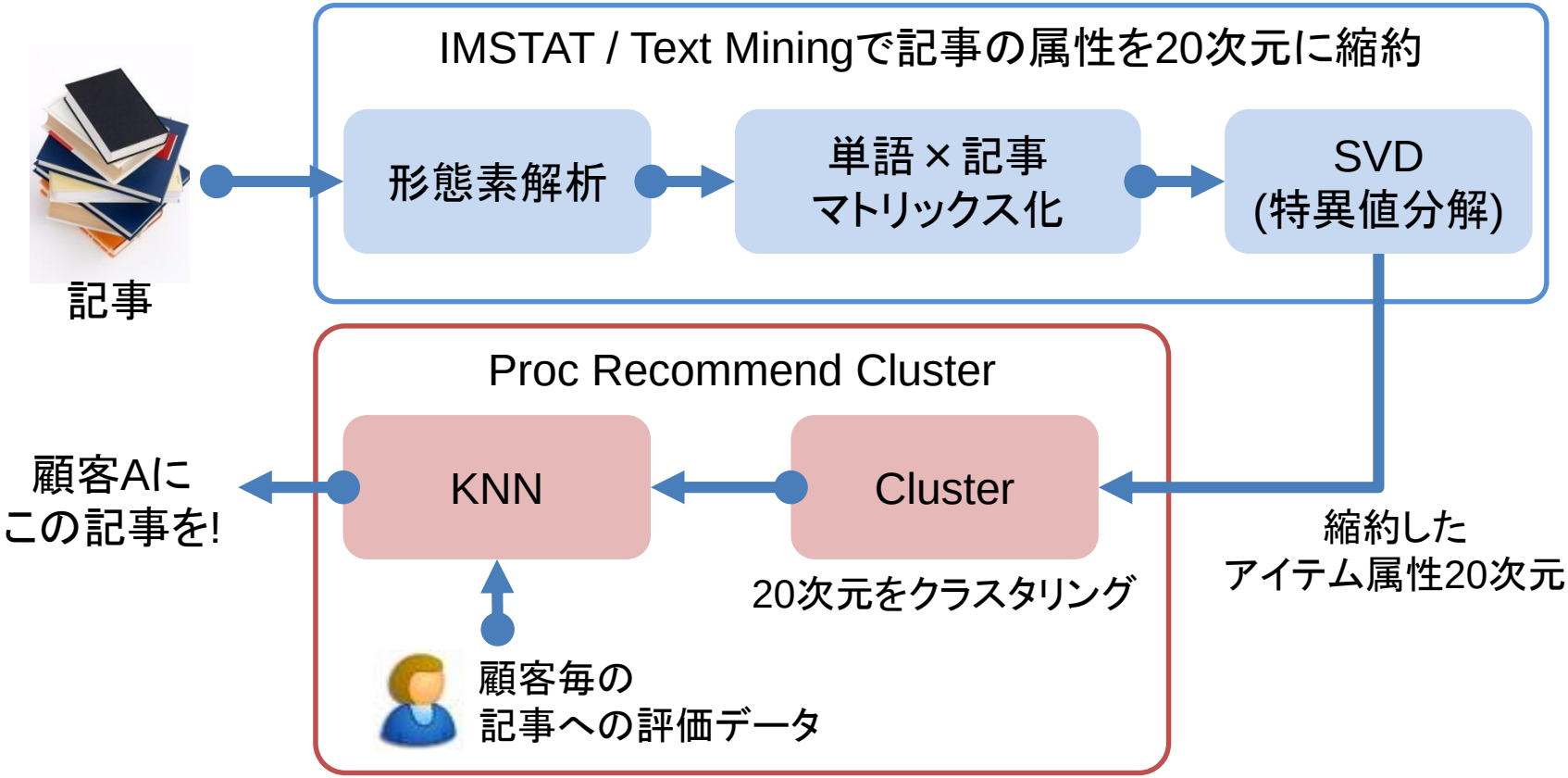
ユーザ毎に5アイテムのおすすめ算出結果イメージ。

Prediction from Recommender System RS.MOVIELENS						
User	Rank	Rating	itemID	year	title	category
1	1	5.3542	557.000000	1962.000000	Mamma Roma	Drama
1	2	5.0897	2503.000000	1998.000000	Apple, The (Sib)	Drama
1	3	5.0719	1178.000000	1957.000000	Paths of Glory	Drama|War
1	4	5.0651	2360.000000	1998.000000	Celebration, The (Festen)	Drama
1	5	5.0172	3245.000000	1964.000000	I Am Cuba (Soy Cuba/Ya Kuba)	Drama
33	1	4.6101	2905.000000	1962.000000	Sanjuro	Action|Adventure
33	2	4.5831	3897.000000	2000.000000	Almost Famous	Comedy|Drama
33	3	4.5613	2503.000000	1998.000000	Apple, The (Sib)	Drama
33	4	4.5612	53.000000	1994.000000	Lamerica	Drama
33	5	4.5317	457.000000	1993.000000	Fugitive, The	Action|Thriller

出展：SAS IMSTAT online doc より MovieLesn データの利用例。 http://support.sas.com/documentation/onlinedoc/lasrserver/index.html

応用例 “Text Mining + Proc Recommend”

記事をテキストマイニングし、記事の属性情報を生成。レコメンドに利用する。

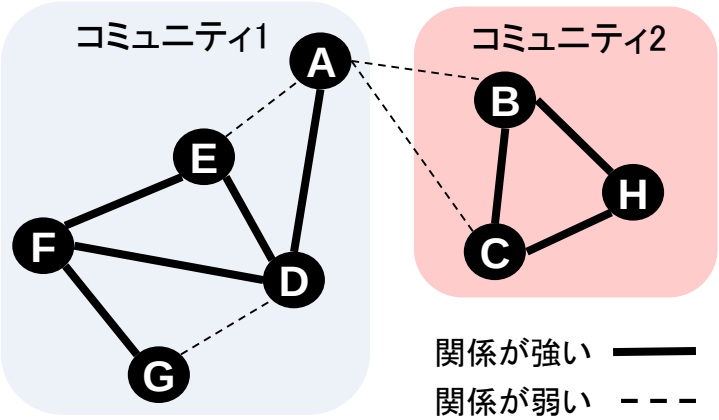


応用例 “Community Detection + Proc Recommend”

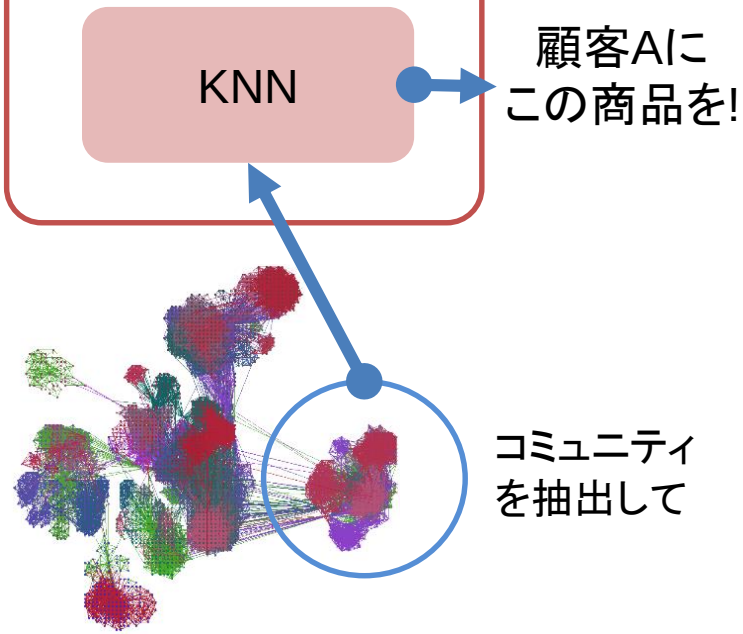
SNS等ソーシャルネットワーク情報からヒトのコミュニティを判定。
コミュニティ内部での流行りを意識したコミュニティ内でのレコメンド計算。

IMSTAT / Hypergroups

Community Detection (コミュニティ検出)
つながりの強弱によって、高密度でつながるグループ(コミュニティ)に分解する。



Proc Recommend



For more Information

SAS® In-Memory Statistics

1) http://www.sas.com/ja_jp/software/analytics/in-memory-statistics.html

2) http://www.sas.com/ja_jp/insights/articles/big-data/recommendation-systems.html

SAS Global Forum 2015 発表資料

3) http://support.sas.com/resources/papers/proceedings15/

SASによるFacebook Community Detectionの実施例

4) http://support.sas.com/resources/papers/proceedings15/SAS4648-2015.pdf

KDD 2014 Netflix

5) http://www.slideshare.net/xamat/kdd-2014-tutorial-the-recommender-problem-revisited