Matrix completion:
via semi-supervised clustering

7 апреля, 2016

Ризванов Айдар

Содержание

Matrix completion

- Представим себе социальный опрос, результатом которого является некая **матрица**: строки – опрошенные люди, столбцы – вопросы. К сожалению, некоторые вопросы остались без ответа.
- Хотим восстановить матрицу **M (nxn)**, имеющую только p<<n (в общем случае n1 x n2).
- Важный вопрос: как восстановить матрицу меньше, чем за n^2 измерений.

Предпосылки

- Равномерная выборка наблюдаемых входных данных
- Должно быть проведено порядка $nrlogn$ (как минимум $2nr - r^2$ наблюдений , т.к количество степеней свободы = $2nr - r^2$).
- «Несвязность» сингулярных векторов.

Модель №0

$$min_x \ rank(X)$$
$$\text{subject to } X_{ij} = M_{ij}, \ i,j \in E$$

Данная задача является NP-трудной, поэтому введена следующая модель:

Модель №1

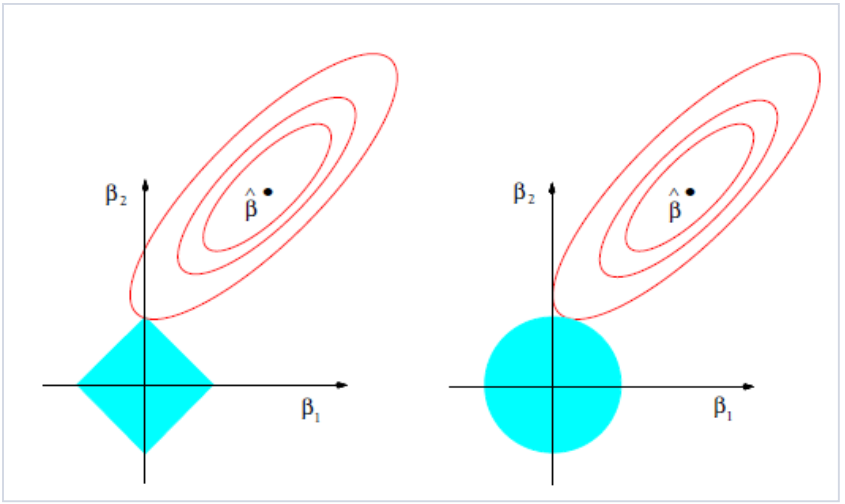
$$min_x \ \|X\|_*$$
$$\text{s.t. } X_{ij} = M_{ij}, \ i,j \in E,$$
$$\|X\|_* = \sum_{i=1}^{n} \sigma_k(X)$$

Главное её отличие в том, что теперь это задача полуопределённой оптимизации, т.к. на самом деле для **положительно полуопределённой** матрицы X мы можем заменить целевой функционал на tr(X)

Источники: Candes, Recht : Exact Matrix Completion via Conv. Opt., 2008

Регуляризация

Регуляризация модели необходима для штрафования полученных коэффициентов модели.



Изображение оценок lasso(слева) и Ridge(справа). Голубые зоны это ограничения: $|\beta_1| + |\beta_2| \leq t$ и $\beta_1^2 + \beta_2^2 \leq t$; красные эллипсы - оценки.

- **Ridge regression**.

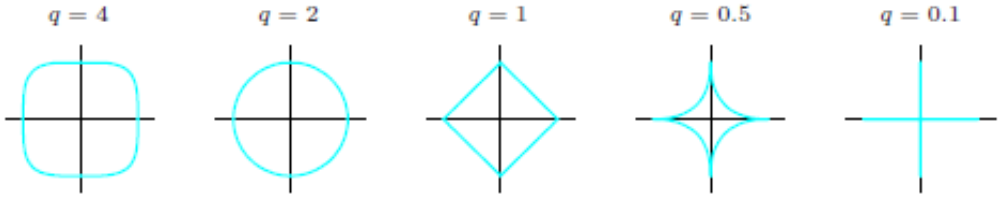
$$\hat{\beta}^{Ridge} = argmin_{\beta}||y - x\beta||_2^2 + \lambda||\beta||_2^2$$

, где λ – это параметр, отвечающий за величину «сжатия».

- **Lasso regression.**

$$\hat{\beta}^{lasso} = argmin_{\beta}||y - x\beta||_2^2 + \lambda||\beta||_1$$

- Именно Lasso позволяет проводить отбор переменных, значительно облегчая интерпретацию результатов.
- **Group Lasso**

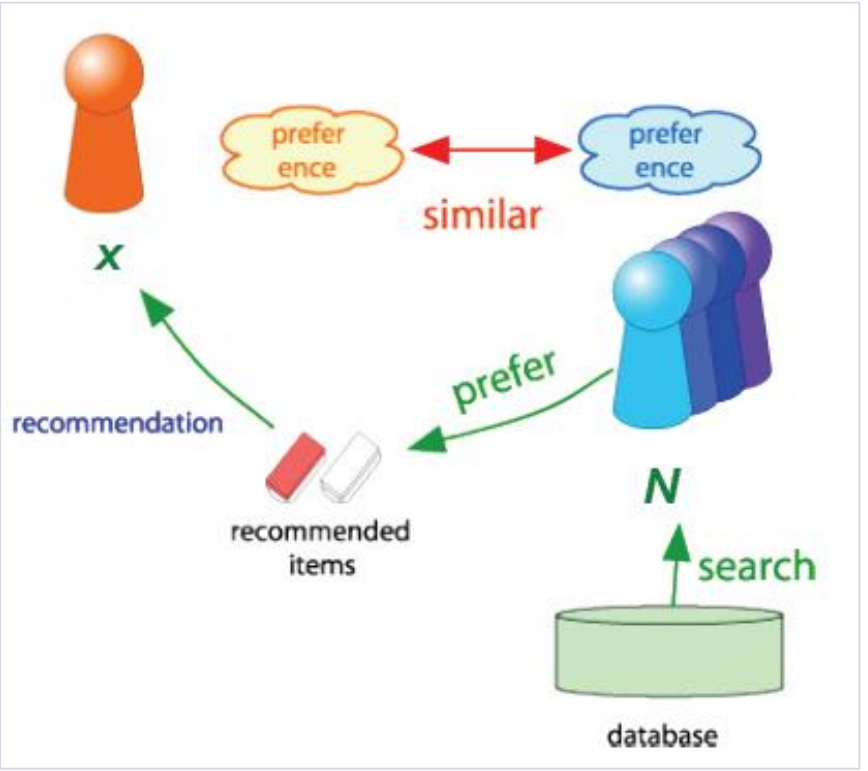


Контуры ограничений $\sum_j |\beta_j|^q$ для разных q

Источники: The Elements of Statistical Learning

Рекомендательные системы

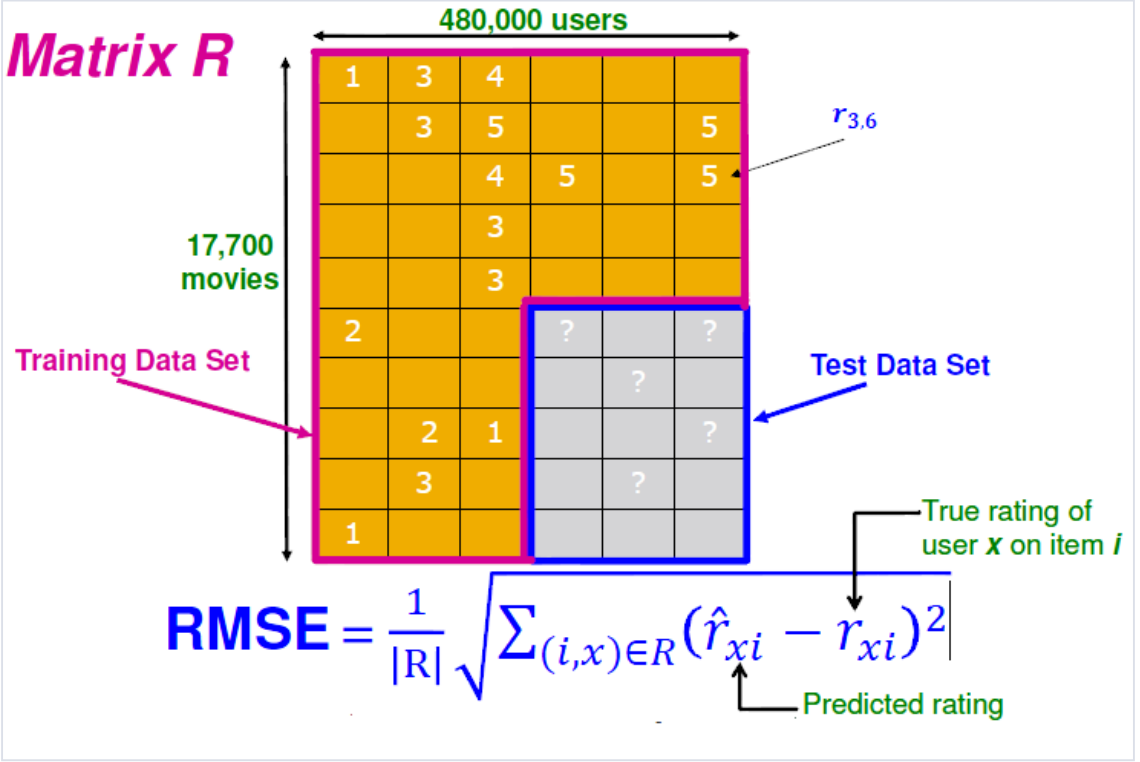
- Попробуем предсказать оценки некоторым элементам (например, фильмам) пользователя **X.**
- Найдём набор пользователей **N**, чьи рейтинги схожи с **X**, и попробуем через них оценить будущие рейтинги **X.**
- Мерой схожести может являться коэффициент Жакара, Пирсона и др. При заранее известной матрице ответов используют **RMSE.**
- Обычные методы **плохо** работают с уникальными пользователями и без накопленных данных.
- Можно (и нужно) подойти и со стороны рекомендуемых элементов.



Источники: Yuan. et al, Model selection and estimation in regression with grouped variables, 2006

Пример 1

- **The Netflix prize**. Известное соревнование на создание лучшего алгоритма предсказания рейтинга фильму, на основе данных о предыдущих выставленных оценках.
- Дано: 480000 пользователей, 17700 фильмов, 6 лет наблюдений
- Тестирование на 2.8 миллионах рейтингов
- Для оценки результата используется **RMSE** (Root mean squared error). 0.8567- победители.



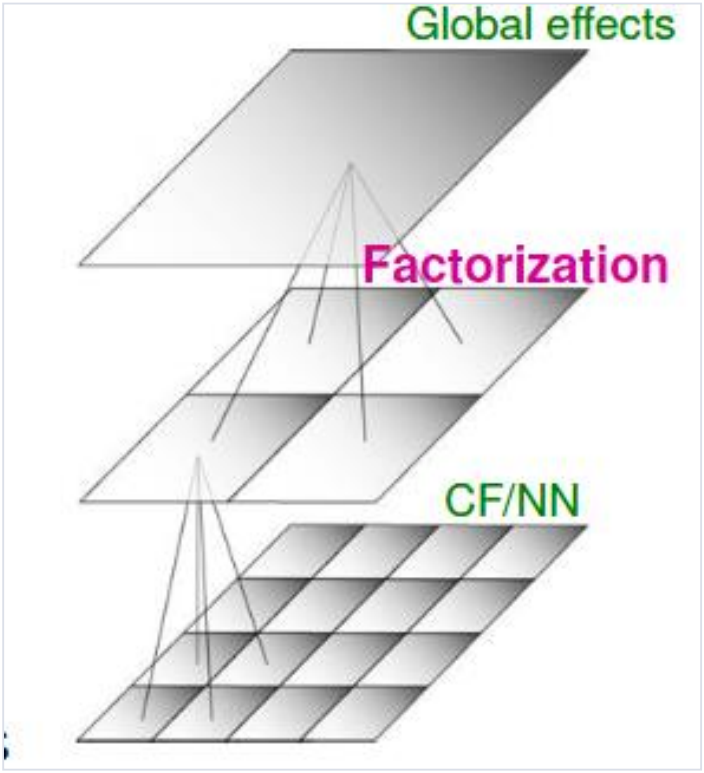
Источники: Montanari A., Collaborative filtering: Models and Algorithms

Пример 1

NETFLIX

Победители соревнования Netflix использовали многоуровневую модель, на подобие нейронных сетей:

- Глобальные сравнения .
- Факторизация матрицы используя сингулярное разложение.
- Анализ «ближайших соседей» основываясь на мере схожести самих фильмов.
- Регуляризация для борьбы с переобучением.
- Важный скачок, связанный с параметризацией модели временем.

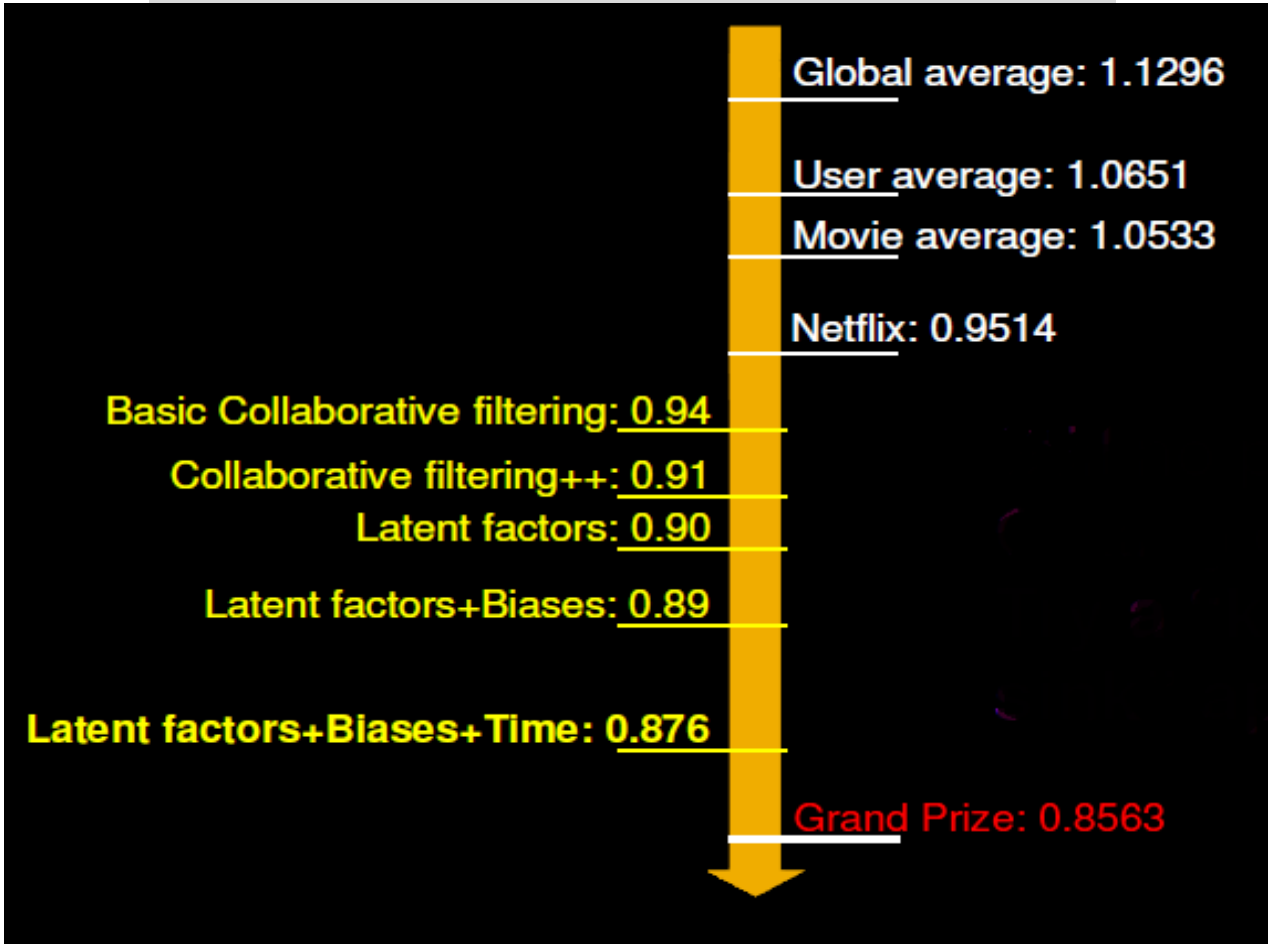


Источники: Montanari A., Collaborative filtering: Models and Algorithms

Пример 1

NETFLIX

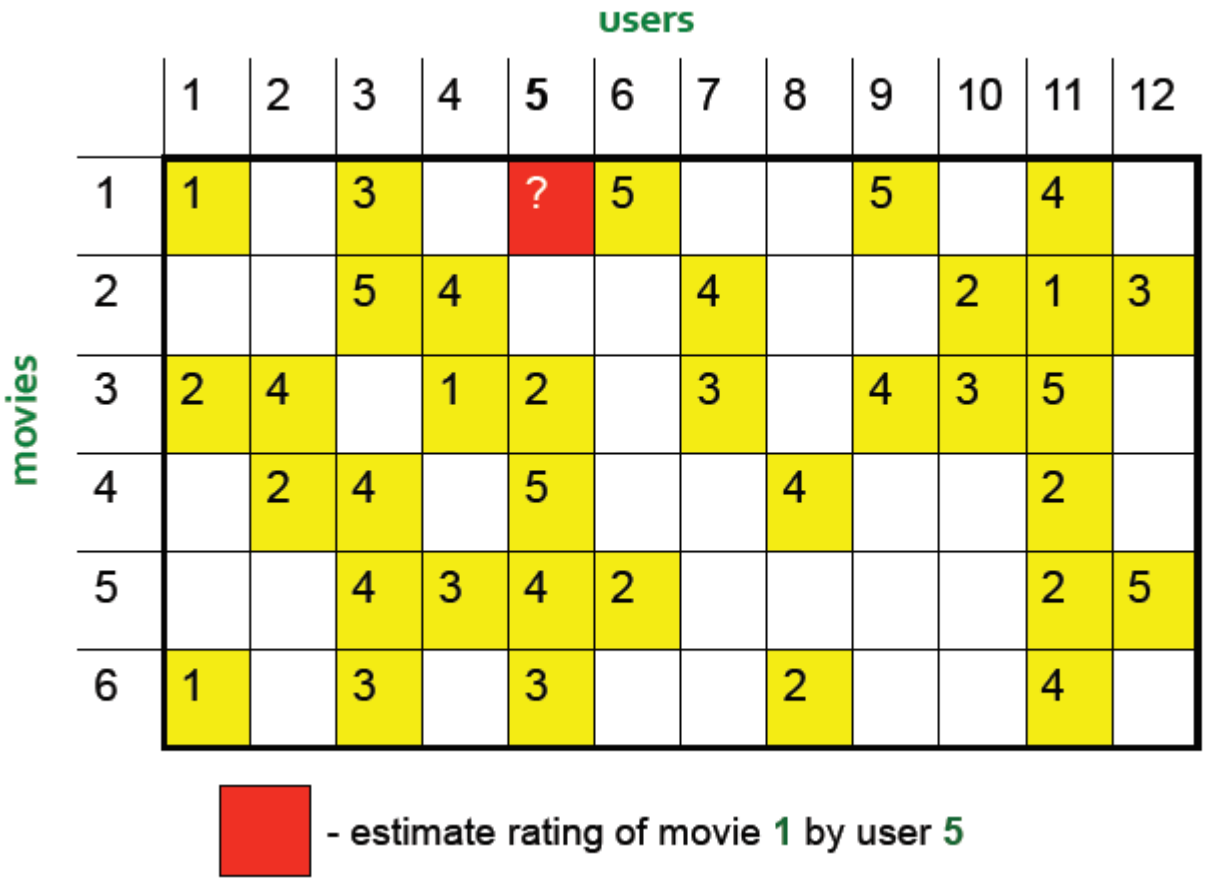
Сравнение результатов различных методов



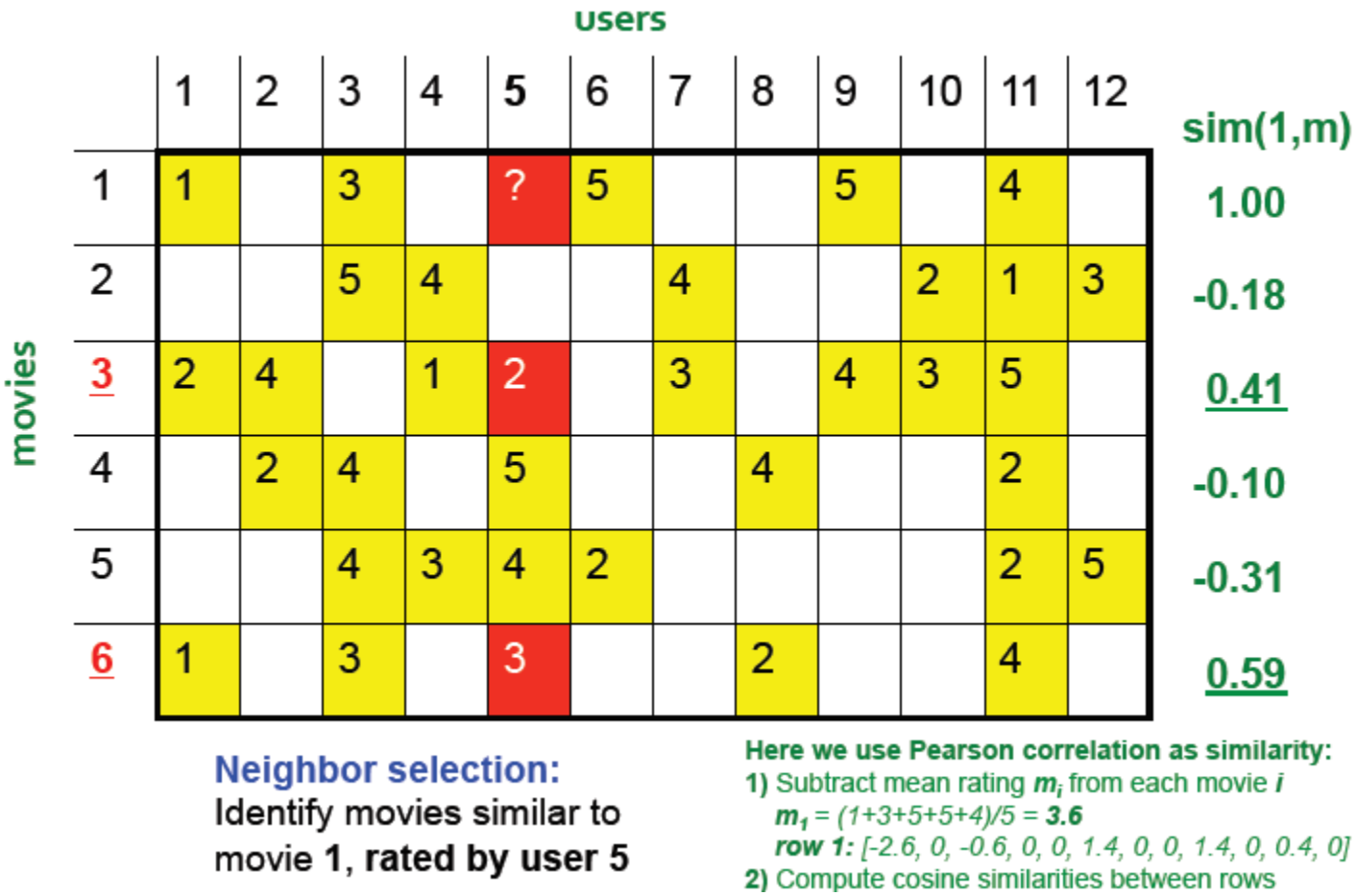
- Стоит отметить использование отличных от описанных в теории регуляризаций на практике (alternating).
- Для данной проблемы существует развитая теоретическая база.

Источники: Montanari A., Collaborative filtering: Models and Algorithms

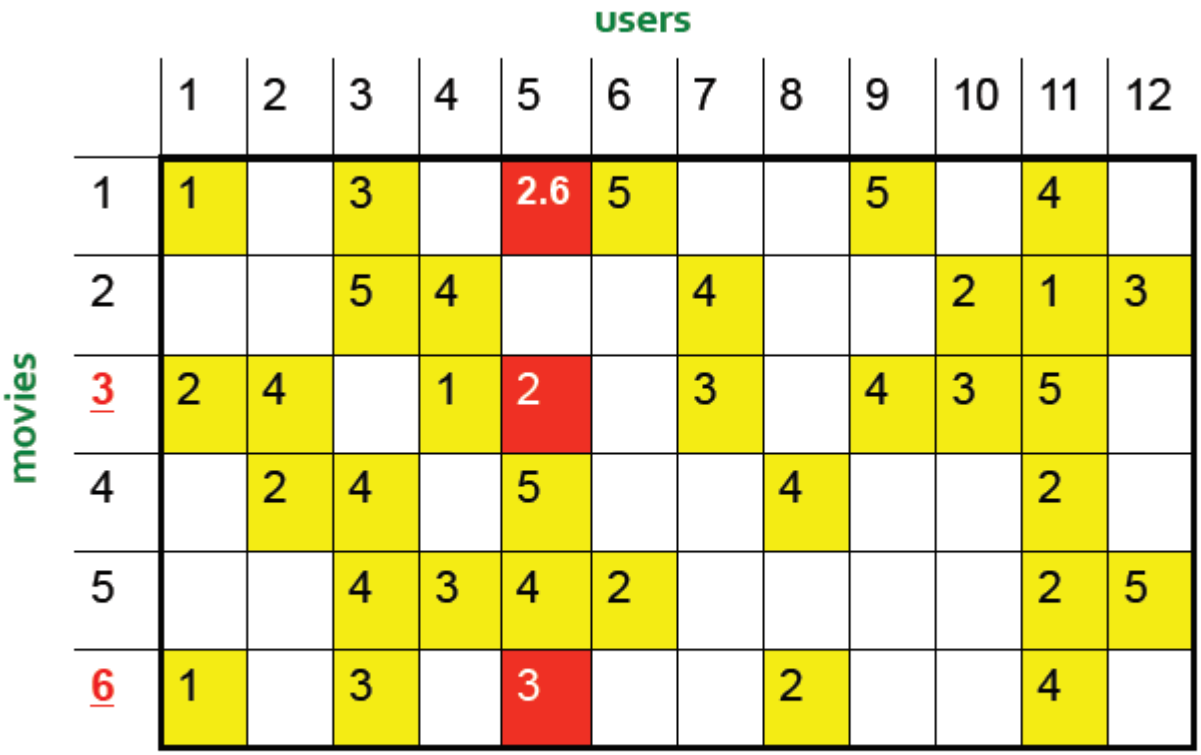
Пример 1


users
movies
1 2 3 4 5 6 7 8 9 10 11 12
1 1 3 ? 5 5 4
2 5 4 4 2 1 3
3 2 4 1 2 3 4 3 5
4 2 4 5 4 2
5 4 3 4 2 2 5
6 1 3 3 2 4
- estimate rating of movie 1 by user 5

Пример 1


users
movies
1 2 3 4 5 6 7 8 9 10 11 12
sim(1,m)
1 1 3 ? 5 5 4 1.00
2 5 4 4 2 1 3 -0.18
3 2 4 1 2 3 4 3 5 0.41
4 2 4 5 4 2 -0.10
5 4 3 4 2 2 5 -0.31
6 1 3 3 2 4 0.59
Neighbor selection:
Identify movies similar to movie 1, rated by user 5
Here we use Pearson correlation as similarity:
1) Subtract mean rating m_i from each movie i
$m_1 = (1+3+5+5+4)/5 = 3.6$
row 1: [-2.6, 0, -0.6, 0, 0, 1.4, 0, 0, 1.4, 0, 0.4, 0]
2) Compute cosine similarities between rows

Пример 1



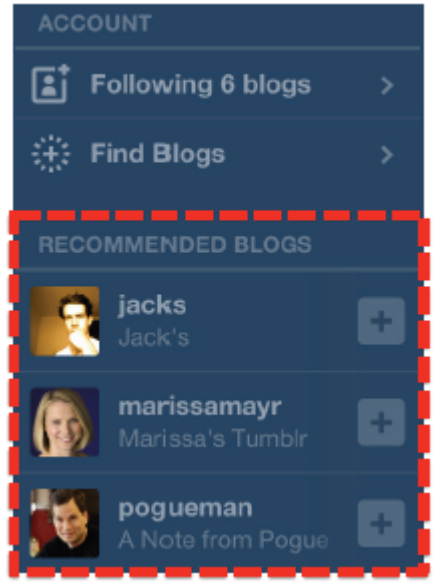
Predict by taking weighted average:

$r_{1.5} = (0.41*2 + 0.59*3) / (0.41+0.59) = 2.6$

Пример 2

tumblr.

Inductive Matrix Completion for Tumblr.



- Граф Подписчика определён , как $G = (V_1; V_2; E)$, где **n,m** - мощности множеств V_1; **E** – набор рёбер между ними(«подписки») . A_{ij} - матрица смежности; **X, Y**– соответствующие матрицы характеристик блогов и пользователей.

- Для того чтобы облегчить работу с **разреженными** данными, вводим модель IMC, - объясняя с помощью матрицы малого ранга Z :

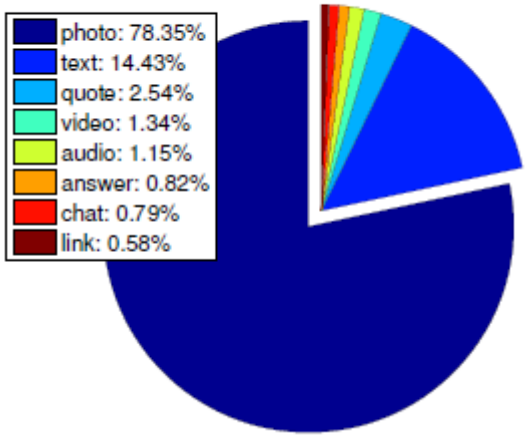
$$A_{ij} = x_i^T Z y_j$$

- Факторизуя Z=WH, целью алгоритма являются матрицы W и H, где функцией потерь является **выпуклая функция** , например:

$$Loss_s(a,b) = (a-b)^2 \text{ или } Loss_l(a,b) = log(1+e^{-ab})$$

Модель IMC

$$\min_{W,H} \sum_{(i,j)} Loss(A_{ij}; x_i^T W H^T y_j) + \frac{\lambda}{2}(\|W\|_F^2 + \|H\|_F^2)$$



Источники: Shin D., Cetintas S.; Tumblr Blod Reccomendation with Boosted Inductive Matrix Completion

Пример 2

tumblr.

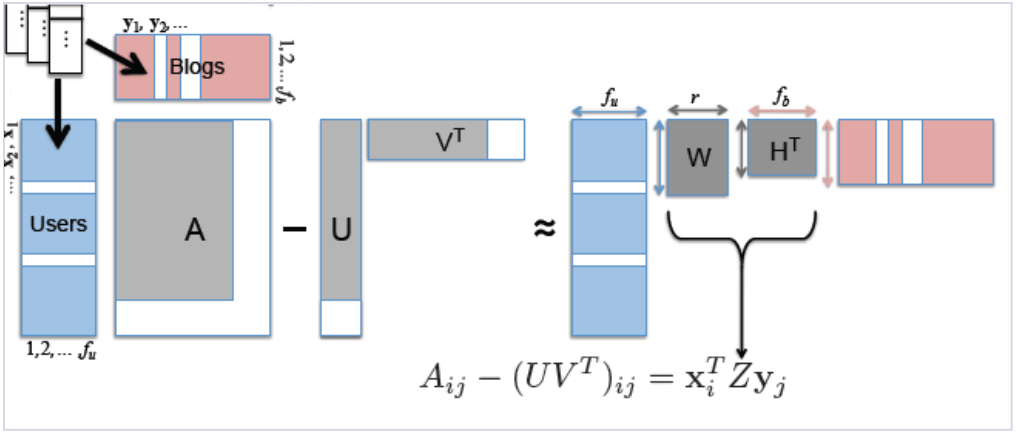
BoostedIMC

- Преобразуя матрицу A через **SVD** получим,

$$A = U_x \hat{Z} U_Y^T \text{ , где } \hat{Z} = \Sigma_X V_X^T Z V_Y \Sigma_Y$$

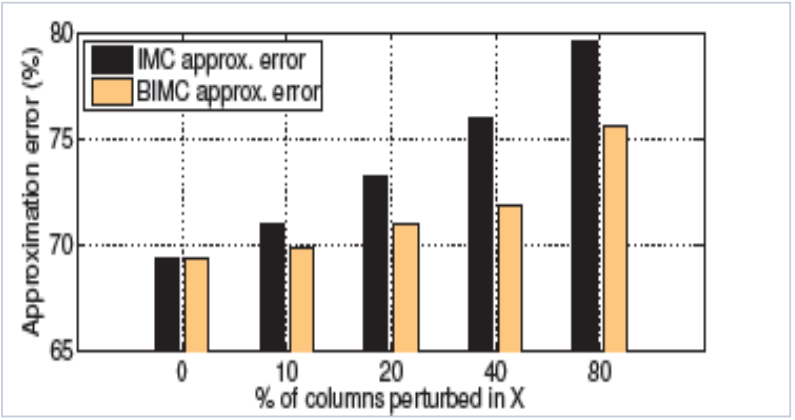
- Также вводится новое объяснение матрицы A, основанное на том, что не всякие регрессоры хорошо объясняют предпочтения пользователей (например, это плохо делает текст): $A_{ij} = (UV^T)_{ij} + \alpha x_i^T Z y_j$

Схема работы метода



Модель BIMC

$$\min_{W,H} \sum_{(i,j)} Loss(A_{ij} - (UV^T)_{ij}; x_i^T W H^T y_j) + \frac{\lambda}{2}(||W||_F^2 + ||H||_F^2)$$



Источники: Shin D., Cetintas S.; Tumblr Blod Reccomendation with Boosted Inductive Matrix Completion

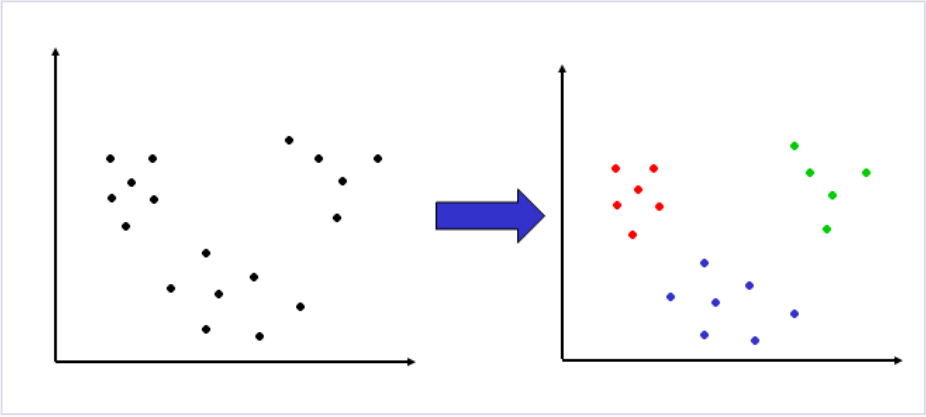
Semi-Supervised Clustering

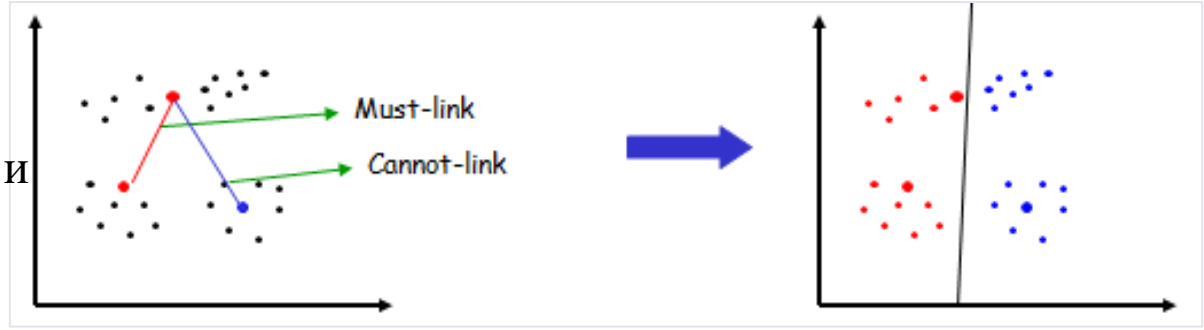
- Как и любой другой метод кластеризации, этот необходим для разделения входящих данных на некоторое (**k**) количество кластеров

- Должны соблюдаться принципы кластеризации и ограничения типа **«must-links» .**

- Алгоритмы такой кластеризации делятся на два вида: **«ограниченную»** кластеризацию и кластеризацию с обучением метрики.

Иллюстрация Supervised/ semi-supervised clustering





Источники: Jinze Liu, Semi-supervised Clusteting, Special Topics in Data Mining

Semi-Supervised Clustering

Идея в том, чтобы свести Semi-Supervised clustering к задаче **Matrix Completion :**

- Используем дополнительную информацию, представленную в виде паросочетаний: матрицы M (must-link) и C(cannot-links).
- Большинство алгоритмов сталкиваются с задачами невыпуклой оптимизации. К тому же неизвестно насколько хорошо алгоритм работает при увеличении количества «пар».

Начальные условия задачи восстановления матрицы:

- Обозначения переменных: $n;\ X = (x_1, \ldots, x_n);\ r;\ n_{min};$
- Вектора **u** , матрица **S**: $S = \sum_{i=1}^{r} u_i u_i^T$
- Задача разбиения данных эквивалентна задаче восстановления матрицы подобия (*Jalali et al., 2011; Yi et al., 2012a*)

Источники: Yi,et al: Semi-Supervised Clustering by Input Pattern Assisted Pairwise Similarity MC; Jalali et al., 2011; Yi et al., 2012a

Semi-Supervised Clustering

Таким образом, задача выглядит как $\min_{P \in \mathbb{R}^{n \times n}} |P|_{tr} \text{ s. t. } \mathcal{R}_{\Delta}(P) = \mathcal{R}_{\Delta}(S)$

Необходимое количество ограничений для восстановления матрицы подобия **S:** $O(kn[logn]^2)$

Предпосылка 1!

$\{u_i\}_{i=1}^{r}$ лежит в подпространстве, образованном $\{z_i\}_{i=1}^{k}$

, где **Z_k – первые k левых сингулярных векторов X, k>r**

Используя эту предпосылку преобразуем задачу ⇨ **S=ZMZ** :

$$\min_{M \in \mathbb{R}^{n \times n}} |M|_{tr}$$
$$\text{s.t. } \mathcal{R}_{\Omega}(ZMZ^T) = \mathcal{R}_{\Omega}(S)$$

Источники: Yi,et al : Semi-Supervised Clustering by Input Pattern Assisted Pairwise Similarity MC

Semi-Supervised Clustering

Input pattern assisted matrix completion.

$$\min_{M \in \mathbb{R}^{n \times n}} |M|_{tr}$$

$$\text{s.t. } \mathcal{R}_\Omega(ZMZ^T) = \mathcal{R}_\Omega(S)$$

Theorem 1.

Theorem 1. *Let $\mu(Z)$ be the coherence measure for matrix Z given by*

$$\mu(Z) = \max_{1 \le i \le n} \frac{n}{k} |[ZZ^\top]_{i,i}|^2 \tag{3}$$

Define

$$\mu_0 = \max\left(\mu(Z), \sqrt{\frac{n}{r n_{\min}}}\right). \tag{4}$$

For fixed $\beta > 2$, define a and B as

$$a = \frac{1}{2}(1 + \log_2 k - \log_2 r) \tag{5}$$

$$B = \frac{512\beta}{3} \mu_0 r k \ln n \tag{6}$$

Then, under assumption **A1** *with a probability* $1 - 4(a+1)n^{-\beta+1} - 2an^{-\beta+2}$, $M_* = Z^\top S Z$ *is the unique optimizer to (2) provided* $|\Omega| \ge aB$.

Источники: Yi,et al: Semi-Supervised Clustering by Input Pattern Assisted Pairwise Similarity MC;

Semi-Supervised Clustering

- Необходима **релаксация** первой предпосылки:

$$P_k = ZZ^T$$

$$\varepsilon^2 = \max_{1 \le i \le r} \frac{1}{n^2} ||u_i - P_k u_i||_F^2$$

- Исходная задача в свою очередь принимает вид:

$$\min_{M \in \mathbb{R}^{k \times k}} |M|_{tr} + \frac{C}{2} ||\mathcal{R}_\Omega(ZMZ^T) - \mathcal{R}_\Omega(S)||_F^2$$

Сравнение результатов работы алгоритма MCCC (с помощью NMI **).**

Datasets	#pairwise constraints	MCCC	Base	MPCK	CCSKL	PMMC	DCA	LMNN	ITML
Mushrooms	2,000	**0.982**	0.540	0.645	0.652	0.876	0.873	0.980	0.971
	4,000	**0.991**	0.540	0.684	0.786	0.898	0.977	0.982	0.981
	6,000	**0.998**	0.540	0.713	0.754	0.923	0.988	0.983	0.984
USPS M2	2,000	0.979	0.866	0.950	0.979	0.976	0.971	0.976	**0.982**
	4,000	**0.984**	0.866	0.977	0.981	0.979	0.975	0.979	0.983
	6,000	**0.991**	0.866	0.989	0.982	0.987	0.981	0.985	0.986
Segment	2,000	**0.750**	0.651	0.693	0.721	0.718	0.723	0.714	0.706
	4,000	**0.755**	0.651	0.701	0.695	0.734	0.741	0.744	0.740
	6,000	**0.774**	0.651	0.718	0.684	0.748	0.760	0.751	0.743

Источники: Yi,et al : Semi-Supervised Clustering by Input Pattern Assisted Pairwise Similarity MC;

Matrix Completion with Noisy Side Information. DirtyIMC

- Идейным отличием этой модели от предыдущей является предпосылка о «**зашумлёных» признаках.**
  Поэтому предлагается восстанавливать матрицу используя робастную модель, состоящую из двух частей:

$$\min_{M,N} \sum_{(i,j)\in\Omega} \ell((XMY^T + N)_{ij}, R_{ij}) + \lambda_M \|M\|_* + \lambda_N \|N\|_*$$

- Первая часть: оценка матрицей малого ранка из множества признаков **XMY^T**
  Вторая часть : **N** – это оценка тех наблюдений, которые «зашумленные» признаки не могут описать.

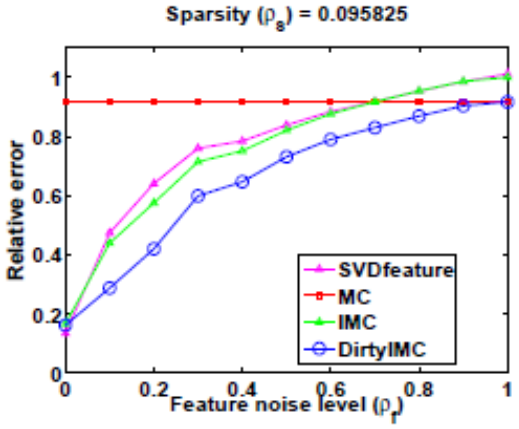
- Авторы гарантируют глобальную сходимость алгоритма, т.к. задача выпукла по **M** и **N**

- Важнейшим вопросом является подбор правильного значения $\lambda_M \backslash \lambda_N$, чтобы модель оставалась робастной и охватывала полезные признаки

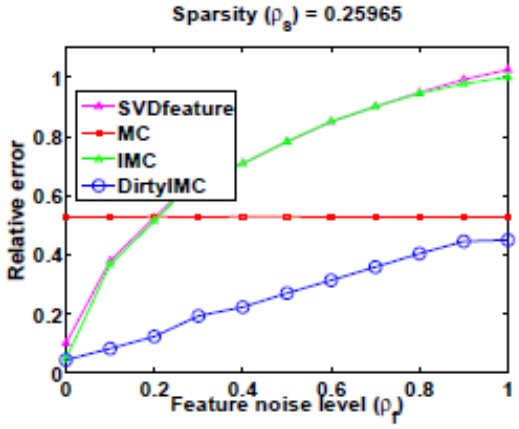
Источники: Chiang; Hsieh; Dhillon: Matrix Completion with Noisy Side information

Matrix Completion with Noisy Side Information

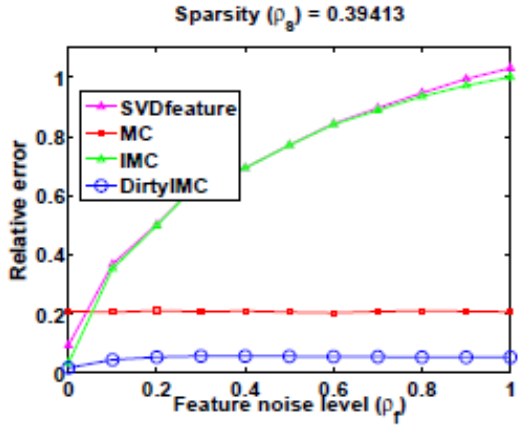
Сравнительные результаты работы DIMC



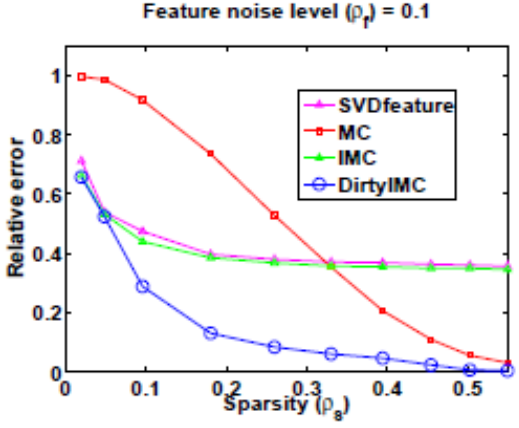
(a) $\rho_s = 0.1$



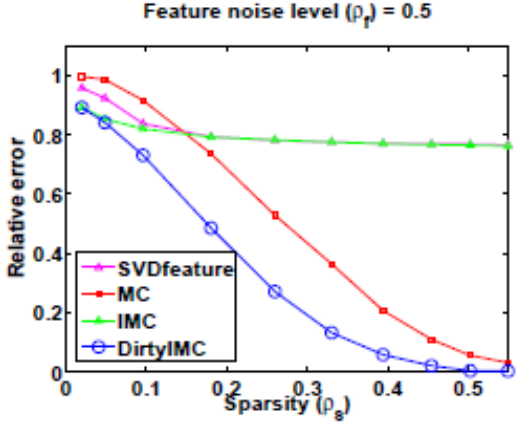
(b) $\rho_s = 0.25$



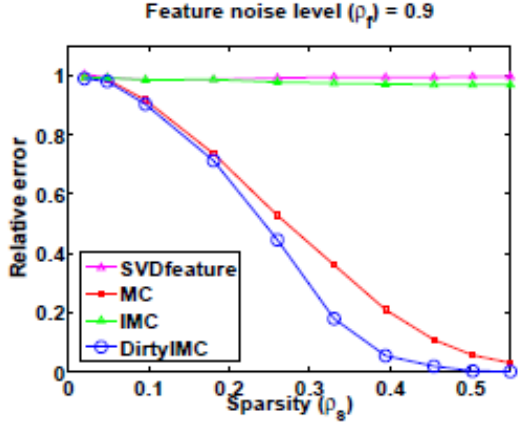
(c) $\rho_s = 0.4$



(d) $\rho_f = 0.1$



(e) $\rho_f = 0.5$



(f) $\rho_f = 0.9$

Источники: Chiang; Hsieh; Dhillon: Matrix Completion with Noisy Side information

Matrix Completion with Noisy Side Information. DirtyIMC

Сравнение DIMC и MCCC

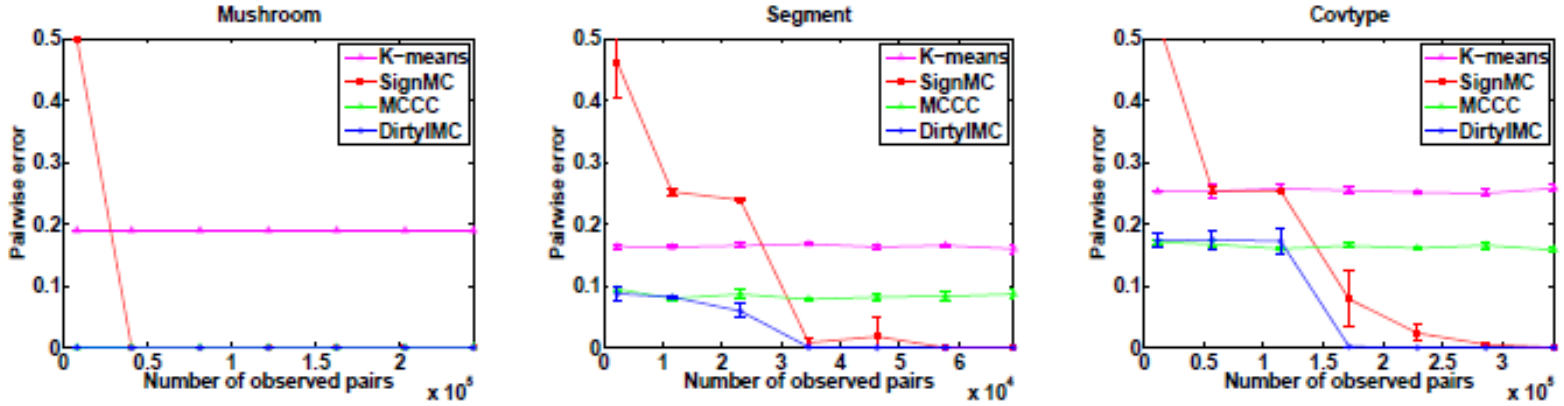


Figure 2: Semi-supervised clustering on real-world datasets. For Mushroom dataset where features are almost ideal, both MCCC and DirtyIMC achieve 0 error rate. For Segment and Covtype where features are more noisy, our model outperforms MCCC as its error decreases given more constraints.

	number of items n	feature dimension d	number of clusters k
Mushrooms	8124	112	2
Segment	2319	19	7
Covtype	11455	54	7

Источники: Chiang; Hsieh; Dhillon: Matrix Completion with Noisy Side information

Идеи и цели

Основной мотивацией нашей работы является идея о том, что оцениваемая матрица должна быть не только малого ранга, но и разреженной. Попробуем реализовать её на примере описанных ранее моделей :

Semi-Supervised Clustering

$$\min_{M \in \mathbb{R}^{k \times k}} |M|_{tr} + \frac{C}{2} ||ZMZ^T - S||_F^2$$

Dirty IMC

$$\min_{M,N} ||XMY^T + N - R||_\Omega + \lambda_N ||N||_* + \lambda_M ||M||_*$$

При этом стоит отдельно упомянуть про разреженный метод главных компонент применительно к нашей постановке задачи:

Sparse PCA

$$\max x^T S x$$
$$\text{s.t. } ||x||_2 = 1$$
$$||x||_0 \leq k$$

- Второе ограничение - на количество ненулевых компонент вектора **x**
- При k=p (размерность S) задача сводится к обычному PCA.
- NP-трудная задача

Идеи и цели. Group Lasso

В некоторых случаях прогнозирующие переменные принадлежат одному классу (например гены). В таких случаях удобно «сужать» такие переменные одновременно.

- P предикатов поделено в L групп. Матрица X описывает принадлежность предикатов к каждой из групп.
- **Group Lasso regression**

$$\min_{\beta \in \mathbb{R}^p} \left(||\mathbf{y} - \beta_0 \mathbf{1} - \sum_{\ell=1}^{L} \mathbf{X}_\ell \beta_\ell||_2^2 + \lambda \sum_{\ell=1}^{L} \sqrt{p_\ell} ||\beta_\ell||_2 \right),$$

- В общем виде используются нормы иного вида.
- Разреженность данных.
- **Целая группа переменных может быть исключена.**

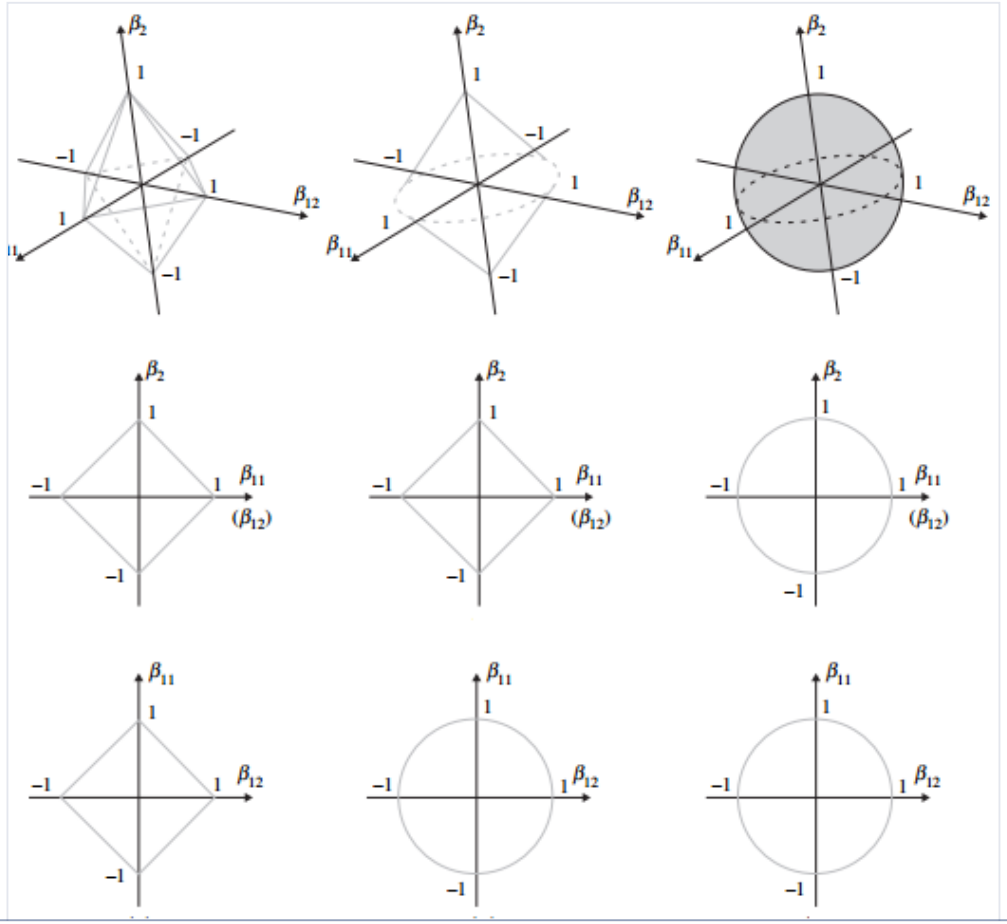


Иллюстрация ограничений соответствующая L_1 -регуляризации , *GroupLasso*, и L_2

Источники: Yuan. et al, Model selection and estimation in regression with grouped variables, 2006; The ESL