Low Rank Matrix Completion Formulation and Algorithm

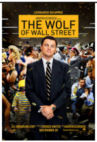
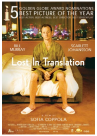
Jian Zhang

Department of Computer Science, ETH Zurich

zhangjianthu@gmail.com

March 25, 2014

Movie Rating

			
Critic A	5	5	
Critic B	6	5	
Jian	9		8
Kind Guy B	9		9

- Observation: Matrix formulation, Clusters of opinion
- Challenge: Missing entries
- Task: Recover the matrix with constraints

Low Rank Matrix Completion

Low Rank Matrix Completion

$$\begin{array}{ll} \text{minimize} & \mathbf{Rank}(X) \\ s.t. & P_\Omega(X) = P_\Omega(M) \end{array}$$

where

$$P_\Omega(S)_{i,j} = \begin{cases} S_{i,j} & \text{if } (i,j) \in \Omega \\ 0 & \text{otherwise} \end{cases}$$

Ω index set of observed entries in M

Outlines

- Linear Algebra Recap
- Two Formulations
- Optimization Algorithms
- Performance Comparison

Recap: SVD and Nuclear Norm

- Singular Value Decomposition

$$\boldsymbol{X} = \boldsymbol{U\Sigma V^T}$$

where

$$\boldsymbol{X} \in \mathbb{R}^{m\times n} \quad \boldsymbol{UU^T} = \boldsymbol{I}_m \quad \boldsymbol{VV^T} = \boldsymbol{I}_n$$

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{u}_1, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_m \end{bmatrix} \begin{bmatrix} \sigma_1 & & & & & \\ & \cdots & & & & \\ & & \sigma_r & & & \\ & & & 0 & & \\ & & & & \cdots & \\ & & & & & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_n \end{bmatrix}^{\boldsymbol{T}}$$

$$\sigma_i \geq 0$$

- Nuclear Norm

$$||X||_* = \sum_{i=1}^{r} \sigma_r$$

Recap: Trace Heuristic for Minimizing Rank

Trace Heuristic

$$\begin{array}{ll} \text{minimize} & \textbf{Rank}(X) \\ s.t. & \boldsymbol{X} \in \mathcal{C} \end{array} \xrightarrow{relaxation} \begin{array}{ll} \text{minimize} & ||\boldsymbol{X}||_* \\ s.t. & \boldsymbol{X} \in \mathcal{C} \end{array}$$

- Positive semidefinite matrix case

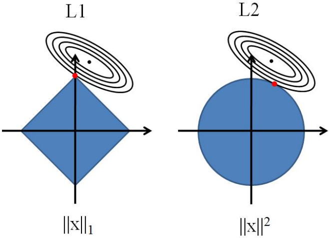
$$\boldsymbol{X} = \boldsymbol{U} \begin{bmatrix} \lambda_1 & & & & & \\ & \cdots & & & & \\ & & \lambda_r & & & \\ & & & 0 & & \\ & & & & \cdots & \\ & & & & & 0 \end{bmatrix} \boldsymbol{U}^{\boldsymbol{T}}$$

$$||\boldsymbol{X}||_* = \sum_{i=1}^{r} \lambda_i = \boldsymbol{Tr}(\boldsymbol{X}) = \sum_{i=1}^{r} |\lambda_i|$$

Recap: Trace Heuristic for Minimizing Rank

$$\min ||\boldsymbol{X}||_* = \min \sum_{i=1}^{r} |\lambda_i|$$

- Minimize L_1 norm encourage zero entries



- $\mathrm{Rank}(\boldsymbol{X}) = \#$ of non-zero eigenvalues

Recap: Trace Heuristic for Minimizing Rank

- General case

$$\begin{array}{lll} \text{minimize} & \textbf{Rank}(X) & \iff \text{minimize} \quad \textbf{Rank}\left(\text{diag}(\boldsymbol{Y}, \boldsymbol{Z})\right) \\ s.t. & \boldsymbol{X} \in \mathcal{C} & \qquad s.t. \quad \begin{bmatrix} \boldsymbol{Y} & \boldsymbol{X} \\ \boldsymbol{X}^T & \boldsymbol{Z} \end{bmatrix} \geq 0, \boldsymbol{X} \in \mathcal{C} \end{array}$$

$\Downarrow$ **PSD Trace Heuristic**

$$\begin{array}{lll} \text{minimize} & ||\boldsymbol{X}||_* & \iff \text{minimize} \quad \textbf{Tr}\left(\text{diag}(\boldsymbol{Y}, \boldsymbol{Z})\right) \\ s.t. & \boldsymbol{X} \in \mathcal{C} & \qquad s.t. \quad \begin{bmatrix} \boldsymbol{Y} & \boldsymbol{X} \\ \boldsymbol{X}^T & \boldsymbol{Z} \end{bmatrix} \geq 0, \boldsymbol{X} \in \mathcal{C} \end{array}$$

Recap: Rank and Decomposition

Rank and Decomposition

$\boldsymbol{X} = \boldsymbol{LR}$ is at most rank r, if $L \in \mathbb{R}^{m \times r}$ and $R \in \mathbb{R}^{r \times n}$

- Right null space of R is subspace of right null space of X
- $\dim(\ker(\boldsymbol{X})) + \dim(\mathrm{X}) = n$, $\dim(\boldsymbol{X}) \leq \dim(\boldsymbol{R}) \leq r$
- much fewer variables if r is much smaller than m and n.

Recap: Summary

Trace Heuristic

$$\begin{array}{ll} \text{minimize} & \mathbf{Rank}(X) \\ s.t. & \boldsymbol{X} \in \mathcal{C} \end{array} \xrightarrow{relaxation} \begin{array}{ll} \text{minimize} & ||\boldsymbol{X}||_* \\ s.t. & \boldsymbol{X} \in \mathcal{C} \end{array}$$

Rank and Decomposition

$\boldsymbol{X} = \boldsymbol{LR}$ is at most rank r, if $L \in \mathbb{R}^{m \times r}$ and $R \in \mathbb{R}^{r \times n}$

Formulation: Barrier

Low Rank Matrix Completion

$$\begin{array}{ll} \text{minimize} & \mathbf{Rank}(X) \\ s.t. & P_\Omega(X) = P_\Omega(M) \end{array}$$

- Barrier:
  $\mathrm{Rank}(\boldsymbol{X})$ is not continuous, not differentiable.
  Thus an objective function difficult to optimize.

Formulation: Least Square

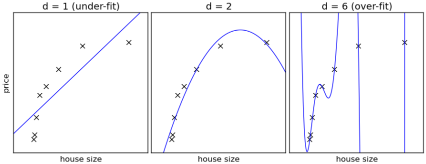
Least Square Formulation

$$\begin{aligned} \text{minimize} \quad & \|P_\Omega(\boldsymbol{X}) - P_\Omega(\boldsymbol{M})\|_2^2 \\ s.t. \quad & \boldsymbol{X} = \boldsymbol{L}\boldsymbol{R}^{\boldsymbol{T}} \end{aligned}$$

- Essentially a Least Square problem.
- Enforce $\boldsymbol{X}$ to be at most rank r, if $\boldsymbol{L} \in \mathbb{R}^{m \times r}$

Formulation: Least Square

- Overfitting: degrade generalization



- Regularizer: control the parameter complexity
  - $L2$ norm of parameter vector is a common regularizer, which is $\sum_{i=1}^{m} ||L_i||_2^2 + \sum_{j=1}^{n} ||R_j||_2^2 = ||\boldsymbol{L}||_F^2 + ||\boldsymbol{R}||_F^2$

Regularized Least Square Formulation

$$\text{minimize} \sum_{(i,j)\in\Omega} ((\boldsymbol{L}\boldsymbol{R}^T)_{i,j} - \boldsymbol{M}_{i,j})^2 + \frac{\mu}{2}||\boldsymbol{L}||_F^2 + \frac{\mu}{2}||\boldsymbol{R}||_F^2$$

Formulation: Nuclear Norm Based

Nuclear Norm Based Formulation

$$\text{minimize} \quad \sum_{(i,j)\in\Omega} (\boldsymbol{X}_{i,j} - \boldsymbol{M}_{i,j})^2 + \mu||\boldsymbol{X}||_*$$

- Soft squared error instead of hard constraints
- Nuclear norm and square error are convex functions of $\boldsymbol{X}$
- μ to keep balance between error and rank.
- $||\boldsymbol{X}||_*$ still need to be updated as a easier explicit function of entries in $\boldsymbol{X}$

Formulation: Nuclear Norm Based

Decomposition of Nuclear Norm

$$||\boldsymbol{X}||_* = \inf\{\frac{1}{2}||\boldsymbol{L}||_F^2 + \frac{1}{2}||\boldsymbol{R}||_F^2 : \boldsymbol{X} = \boldsymbol{L}\boldsymbol{R}^T\}$$

- $||\boldsymbol{A}||_F = (\sum_i \sum_j A_{i,j}^2)^{\frac{1}{2}}$
- Verification with $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Sigma}^{\frac{1}{2}}$ and $\boldsymbol{R} = \boldsymbol{V}\boldsymbol{\Sigma}^{\frac{1}{2}}$ from SVD

$$\begin{aligned} ||\boldsymbol{L}||_F^2 &= \mathbf{Tr}(\boldsymbol{L}\boldsymbol{L}^T) && = \mathbf{Tr}(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}^T) \\ &= \mathbf{Tr}(\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{\Sigma}) && = \mathbf{Tr}(\boldsymbol{\Sigma}) \\ &= ||\boldsymbol{X}||_* \end{aligned}$$

- $\frac{1}{2}||\boldsymbol{L}||_F^2 + \frac{1}{2}||\boldsymbol{R}||_F^2$, as a upper bound for $||\boldsymbol{X}||_*$, can be used to approximate $||\boldsymbol{X}||_*$.

Formulation: Nuclear Norm Based

Approximated Nuclear Norm Based Formulation

$$\text{minimize} \sum_{(i,j)\in\Omega} ((\boldsymbol{L}\boldsymbol{R}^T)_{i,j} - \boldsymbol{M}_{i,j})^2 + \frac{\mu}{2}||\boldsymbol{L}||_F^2 + \frac{\mu}{2}||\boldsymbol{R}||_F^2$$

Formulation: Converged Formulation

Converged Formulation

$$\text{minimize} \sum_{(i,j)\in\Omega} ((\boldsymbol{L}\boldsymbol{R}^T)_{i,j} - \boldsymbol{M}_{i,j})^2 + \frac{\mu}{2}||\boldsymbol{L}||_F^2 + \frac{\mu}{2}||\boldsymbol{R}||_F^2$$

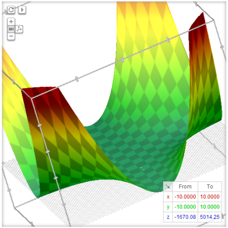
- Story Line 1: Approximately minimize rank with nuclear norm and square error.
- Story Line 2: Least square estimation with L_2 regularizer preventing overfitting.
- **A convergency of beautiful minds.**

Optimization: Non-Convexity

Converged Formulation

$$\text{minimize} \sum_{(i,j)\in\Omega} ((\boldsymbol{L}\boldsymbol{R}^T)_{i,j} - \boldsymbol{M}_{i,j})^2 + \frac{\lambda}{2}||\boldsymbol{L}||_F^2 + \frac{\lambda}{2}||\boldsymbol{R}||_F^2$$

- A non-convex problem with local minimum. Ex. $(x * y - 1)^2 + 0.001x^2 + 0.001y^2$
- Under mild conditions, a local minimum here is globally optimal for $\min \sum_{(i,j)\in\Omega}(\boldsymbol{X}_{i,j} - \boldsymbol{M}_{i,j})^2 + \mu||\boldsymbol{X}||_*$
- Convex if λ large enough.
- **Local mininum is worth our efforts.**



Optimization: Alternating Least Square

Alternating Least Square Algorithm

$$\text{minimize} \sum_{(i,j)\in\Omega} ((\boldsymbol{L}\boldsymbol{R}^T)_{i,j} - \boldsymbol{M}_{i,j})^2 + \frac{\mu}{2}||\boldsymbol{L}||_F^2 + \frac{\mu}{2}||\boldsymbol{R}||_F^2$$

1 Fix $\boldsymbol{L}$, $\forall j$

$$\min||\hat{\boldsymbol{L}}_j \boldsymbol{R}_{j.} - \hat{\boldsymbol{M}}_{.j}||_2^2 + \frac{\mu}{2}||\boldsymbol{R}_{j.}||_2^2$$

$\hat{\boldsymbol{L}}_j$: include row $\boldsymbol{L}_{i.}$, $\forall i, (i,j) \in \Omega$ (On Blackboard)
$\hat{\boldsymbol{M}}_{.j}$: include entry $M_{i,j}$ $\forall i, (i,j) \in \Omega$

2 Fix $\boldsymbol{R}$, $\forall i$

$$\min||\hat{\boldsymbol{R}}_i^T \boldsymbol{L}_{.i} - \hat{\boldsymbol{M}}_{i.}^T||_2^2 + \frac{\mu}{2}||\boldsymbol{L}_{.i}||_2^2$$

3 Repeat 1 and 2 until convergency.

Optimization: Alternating Least Square

Pros:

- Convex sub-problem.
- Closed-form solution in each step.

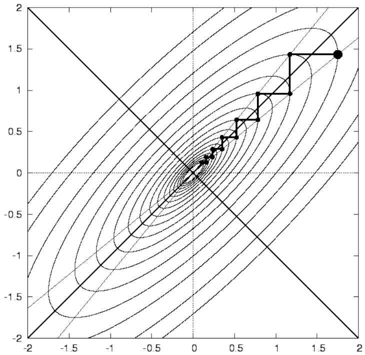
$$\begin{aligned} \boldsymbol{R}_{j.}^{*} &= \operatorname{argmin} ||\hat{\boldsymbol{L}}_j \boldsymbol{R}_{j.} - \hat{\boldsymbol{M}}_{.j}||_2^2 + \frac{\mu}{2} ||\boldsymbol{R}_{j.}||_2^2 \\ &= \left(\hat{\boldsymbol{L}}_j^T \hat{\boldsymbol{L}}_j + \frac{\mu}{2} \boldsymbol{I} \right)^{-1} \hat{\boldsymbol{L}}_j^T \hat{\boldsymbol{M}}_{.j} \end{aligned}$$

Cons:

- $O(\max(nr^3, mr^3))$ update operation in each step.

Optimization: Gradient Descent Recap

- $\nabla f(x)$ is the direction with most rapid increase.
- $x^{(k+1)} = x^{(k)} - \eta^{(k)} \nabla f(x)$ gives decreasing sequence if step size is small enough.



Optimization: Gradient Descent

Gradient Descent Algorithm

$$\text{minimize} \sum_{(i,j)\in\Omega} ((\boldsymbol{L}\boldsymbol{R}^T)_{i,j} - \boldsymbol{M}_{i,j})^2 + \frac{\mu}{2}||\boldsymbol{L}||_F^2 + \frac{\mu}{2}||\boldsymbol{R}||_F^2$$

$$\text{where} \quad \boldsymbol{X} = \boldsymbol{L}\boldsymbol{R}^T, \quad L \in \mathbb{R}^{m\times r}, R \in \mathbb{R}^{n\times r}$$

1 $\forall i, j$

$$\boldsymbol{L}_{i.}^{(k+1)} = \boldsymbol{L}_{i.}^{(k)} - \eta^{(k+1)} \left[\sum_{j:(i,j)\in\Omega} \left(\boldsymbol{L}_{i.}^{(k)} \boldsymbol{R}_{j.}^{(k)T} - \boldsymbol{M}_{i,j} \right) \boldsymbol{R}_{j.}^{(k)T} + \mu \boldsymbol{L}_{i.}^{(k)} \right]$$

Similar for $\boldsymbol{R}_{.j}^{(k+1)}$

2 Repeat 1 until convergency.

Optimization: Gradient Descent

Pros:

- $O((m+n)r^2)$ update operation in each step.
- $O((m+n)r)$ update operation with Stochastic Gradient Descent.

Pick (i,j) randomly.
Only consider error and $||.||_F$ related to entry (i,j)

$$\boldsymbol{L}_{i.}^{(k+1)} = \boldsymbol{L}_{i.}^{(k)} - \eta^{(k+1)} \left[\left(\boldsymbol{L}_{i.}^{(k)} \boldsymbol{R}_{j.}^{(k)T} - \boldsymbol{M}_{i,j} \right) \boldsymbol{R}_{j.}^{(k)T} + \frac{\mu}{|\Omega_{(i.)}|} \boldsymbol{L}_{i.}^{(k)} \right]$$

Cons:

- Iterative approach might need much more epochs.

Optimization: Conclusion

Alternating Least Square

- Closed-form update with high stepwise complexity.
- Might need fewer epochs.

(Stochastic) Gradient Descent

- Iterative update with low stepwise complexity.
- Might need more epochs.

Experiment: Efficiency

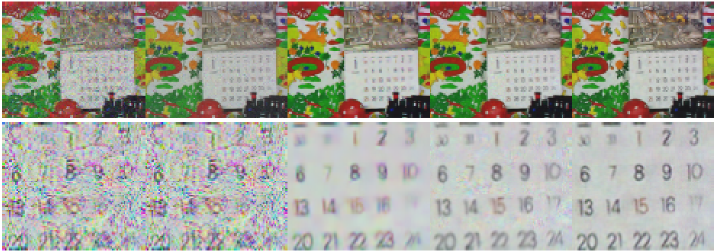
- Netflix Dataset: Movie rating from 480K reviewer's rating for more than 18K movies.
- Configuration: maximal possible rank r, $\boldsymbol{L} \in \mathbb{R}^{m \times r}$.

method	# of epochs	time/min (r = 50)	time/min (r = 100)
ALS	6	66	290
SGD	31	58	52.8

Experiment: More Applications and Qualitative Results

Video Denoising:

- Observe reliable pixels.
- Vectorizing and stacking similar patches to matrix.
- Employ low rank structure within the matrix.



Experiment: More Applications and Qualitative Results

Background Substraction:

- Vectorizing and stacking different video frame.
- Background is the low rank components.
- Foreground is sparse noise.

Conclusion

Technically

- Matrix Completion is a fundamental problem for many real world application.
- Nuclear norm based and Least Square based formulation converge into one with different relaxation.
- Alternating Least Square and (Stochastic) Gradient Descent are widely used alternatives to solve matrix completion problems.

Methodologically

- Relaxing the formulation and employing heuristic is a wise way to approximately solve difficult problems.
- Tradeoff between elegant closed form solution and "brute force" operations.

Thank You !

Q & A

- **Acknowledgement:**
  Thanks to Tobias for helpful discussions, suggestions and chips!!!

Reference

1. Low-rank Matrix Completion using Alternating Minimization.
   Prateek et al. STOC 2013.
2. Distributed Matrix Completion.
   Teflioudi et al. ICDM 2012.
3. Large-scale Parallel Collaborative Filtering for the Netflix Prize.
   Zhou et al. AAIM 2008.
4. Parallel Stochastic Gradient ALgorithms for Large-Scale Matrix Completion.
   Recht et al. MPC 2013.
5. Matrix Rank Minimization with Applications.
   Fazel. Phd Thesis.
6. Robust Video Denoising using Low Rank Matrix Completion.
   Ji et al. CVPR 2010.
7. Robust Principal Component Analysis:Exact Recovery of Corrupted Low-Rank Matrices via Convex Optimization
   Wright et al. NIPS 2009.