Lecture 5 Matrix Factorization

Sunday, September 11, 2016 7:52 PM

- Basic ways of working with matrices
- matrix, rank, norms
  - Recall: matrix $M \in \mathbb{R}^{n \times m}$ is of rank r, if

$$M = \sum_{i=1}^{r} u_i v_i^T$$

  - In ML, often assume matrix is close to low rank. (e.g. in topic model $M = AW$)
  - how to measure closeness?
  - Def: Frobenius norm $\|M\|_F = \sqrt{\sum_{i,j} M_{i,j}^2}$

    Spectral/operator norm

$$\|M\| = \max_{\|u\|=\|v\|=1} u^T M v = \max_{\|v\|=1} \|Mv\|$$

- Q: How to find the closest low rank matrix?
- Singular Value Decomposition

  Def: The SVD of M has form

$$M = UDV^T = \sum_i \sigma_i u_i v_i^T$$

  U, V orthonormal, $D = \mathrm{diag}(\sigma_1, \sigma_2, \cdots, \sigma_n)$

  $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \cdots \geq \sigma_n \geq 0$

  - Optimization view of SVD

$$\sigma_1 = \max_{\|u\|=\|v\|=1} u^T M v \quad (u_1, v_1) = \arg\max$$

$$\forall i > 1 \quad \sigma_i = \max_{\substack{\|u\|=\|v\|=1 \\ \forall j<i,\ u \perp u_j \\ v \perp v_j}} u^T M v \quad (u_i, v_i) = \arg\max$$

  - Relationship to eigenvalue and eigenvectors.

    (Recall: if $Av = \lambda v$, then λ is eigenvalue, $v \neq 0$ is eigenvector)

    u_i's are eigenvectors of MM^T, with eigenvalues σ_i^2

    v_i's are eigenvectors of $M^T M$, with eigenvalues σ_i^2.

- Eckart Young: Let $M_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$ be the truncated SVD

$$\|M - M_k\|_F = \min_{\text{rank}(A) \le k} \|M - A\|_F = \sqrt{\sum_{i=k+1}^{n} \sigma_i^2}$$

$$\|M - M_k\| = \min_{\text{rank}(A) \le k} \|M - A\| = \sigma_{k+1}$$

- Computing SVD: the Power Method.

  - SVD can be computed in $O(n^3)$ time.
  - rank-k SVD takes $O(n^2 k)$ time.

Power method: initialize $u^{(0)}$ = random Gaussian

$$u^{(t+1)} = \frac{(MM^T)\, u^{(t)}}{\|(MM^T)\, u^{(t)}\|}$$

Lemma: If $\frac{\sigma_1^2}{\sigma_2^2} \ge 1+\rho$, then after $O\left(\frac{\log \frac{n}{\varepsilon\delta}}{\min(1,\rho)}\right)$ iterations

w.p. $1-\delta$, u becomes ε-close to u_1 (in l_2 norm)

Proof: Since matrix multiplication is linear, can do normalization at the end.

write $u^{(0)}$ in the singular value basis.

$$u^{(0)} = \sum_{i=1}^{n} c_i^{(0)} u_i$$

$$u^{(t+1)} = (MM^T) u^{(t)} = \sum_{i=1}^{n} \sigma_i c_i^{(t)} u_i$$

therefore. $c_i^{(t)} = \sigma_i^t c_i^{(0)}$

with probability $\ge 1-\delta$, $\frac{c_1^{(0)}}{\sqrt{\sum_{i=2}^{n} c_i^{(0)}}} \ge \frac{0.1\delta}{\sqrt{n}}$ (anti-concentration)

$$\frac{c_1^{(t)}}{\sqrt{\sum_{i=2}^{n} c_i^{(t)}}} \ge \left(\frac{\sigma_1^2}{\sigma_2^2}\right)^t \frac{c_1^{(0)}}{\sqrt{\sum_{i=1}^{n} c_i^{(0)}}} \ge (1+\rho)^t \cdot \frac{0.1\delta}{\sqrt{n}}$$

$$\ge \frac{10}{\varepsilon} \text{ for large } t. \; \square$$

- Finding top k singular vector
  - Method 1: Deflation

after findind $\hat{u}_1 \approx u_1$, $\hat{v}_1 \approx v_1$, $\hat{\sigma}_1 \approx \sigma_1$

after findind $\hat{u}_1 \approx u_1$, $\hat{v}_1 \approx v_1$, $\hat{\sigma}_1 \approx \sigma_1$

$$M \Leftarrow M - \hat{\sigma}_1 \hat{u}_1 \hat{v}_1^T$$

recursively find top $k-1$ singular vectors

Pros: can find each singular vector with good accuracy

Cons: error might accumulate

often needs spectral gap between all σ_i, σ_{i+1} ($i \le k$)

- Method 2: Subspace Iteration

$U^{(0)}$ = random Gaussian $n \times k$ matrix

$$U^{(t+1)} = \text{orthonormalize}((MM^T) U^{(t)})$$

orthonormalize(U) tries to find R such that

$$(UR)^T (UR) = I$$

can be done by QR factorization

(search Gram-Schmidt)

Pros: finds top k subspace in one shot

only require $\sigma_k > \sigma_{k+1}$

Cons: usually no guarantee on individual vectors.