Sparsity in Machine Learning

Sparsifiers
SVD
Linear Regression
K-means

M. Magdon-Ismail
Rensselaer Polytechnic Institute

(Joint work with C. Boutsidis and P. Drineas)

June 20, 2013.
Out-of-Sample Prediction

- A pattern exists ($f$)
- We don’t know it
- We have data to learn it ($\mathcal{D}$)

\[
f = \begin{cases} 
-1 & 
\end{cases},
\quad
f = \begin{cases} 
+1 & 
\end{cases}
\quad
f = \begin{cases} 
? & 
\end{cases}
\]
Data

$n$ data points

\[
\begin{bmatrix}
\begin{array}{ccc}
- & \mathbf{x}_1^T & - \\
- & \mathbf{x}_2^T & - \\
\vdots & & \\
- & \mathbf{x}_n^T & -
\end{array}
\end{bmatrix}
= \begin{bmatrix}
\begin{array}{cccc}
x_{11} & x_{12} & \cdots & x_{1d} \\
x_{21} & x_{22} & \cdots & x_{2d} \\
\vdots & \vdots & \ddots & \vdots \\
x_{n1} & x_{n2} & \cdots & x_{nd}
\end{array}
\end{bmatrix}
\begin{bmatrix}
\begin{array}{c}
y_1^T \\
y_2^T \\
\vdots \\
y_n^T
\end{array}
\end{bmatrix}
\]

\[X \in \mathbb{R}^{n \times d}\]
\[Y \in \mathbb{R}^{n \times \omega}\]

viewers × movie ratings

credit applicants × credit features

\[y = \pm 1\ (\text{approve or not})\]
Data

\[ X \in \mathbb{R}^{231 \times 174} \]

\[ Y \in \mathbb{R}^{231 \times 166} \]
Sparsity

Represent your solution using only a few . . .
Sparsity

Represent your solution using **only a few** ...

**Example:** linear regression

\[
\begin{bmatrix}
\vdots \\
\end{bmatrix}
\begin{bmatrix}
\vdots \\
\end{bmatrix} = \begin{bmatrix}
\vdots \\
\end{bmatrix}
\]

\[
Xw = y
\]

\(y\) is an optimal linear combination of the columns in \(X\).
Sparsity

Represent your solution using **only a few** . . .

**Example:** linear regression

\[
\begin{bmatrix}
\text{red} & \text{red} \\
\text{red} & \text{red} \\
\end{bmatrix}
\begin{bmatrix}
\text{red} \\
\text{red} \\
\end{bmatrix}
= 
\begin{bmatrix}
\text{red} \\
\text{red} \\
\end{bmatrix}
\]

\[Xw = y\]

\(y\) is an optimal linear combination of **only a few** columns in \(X\).
(sparse regression; regularization \((\|w\|_0 \leq k)\); feature subset selection; . . .)
Sparse solutions generalize to out-of-sample better.
Sparse solutions are easier to interpret.
Computations are more efficient.

**Problem:** sparsity is a combinatorial constraint.
Singular Value Decomposition (SVD)

\[
X = [U_k \ U_{d-k}] \begin{bmatrix}
\Sigma_k & 0 \\
0 & \Sigma_{d-k}
\end{bmatrix} \begin{bmatrix}
V_k^T \\
V_{d-k}^T
\end{bmatrix}
\]

\(O(nd^2)\)

\[
X_k = U_k \Sigma_k V_k^T
\]

\[
= XV_k V_k^T
\]

\(X_k\) is the best rank-\(k\) approximation to \(X\).

Reconstruction of \(X\) using only a few features.

\(V_k\) is an orthonormal basis for the best \(k\)-dimensional subspace of the row space of \(X\).
**$V_k$ and Sparsity**

**Principal Components Analysis (PCA):**

$$Z = XV_k$$

($n \times k$)

**Feature subset selection:** Important “dimensions” of $V_k^T$ are important for $X$

$$\begin{bmatrix}
\times s_1 & \times s_2 & \times s_3 & \times s_4 & \times s_5 \\
\end{bmatrix}
\begin{bmatrix}
V_k^T
\end{bmatrix}
\rightarrow
\begin{bmatrix}
\hat{V}_k^T
\end{bmatrix}
\in \mathbb{R}^{k \times r}
$$

The sampled $r$ columns are “good” if

$$I = V_k^T V_k \approx \hat{V}_k^T \hat{V}_k.$$

Sampling schemes:
- Largest norm (Jollife, 1972);
- Randomized norm sampling (Rudelson, 1999; RudelsonVershynin, 2007);
- Greedy (Batson et al, 2009; BDM, 2011).
Approximate SVD

\[ X = XV_kV_k^T + E \]

Let \( \hat{V}_k \) be an approximate \( V_k \)

\[ X = X\hat{V}_k\hat{V}_k^T + \hat{E} \]

\( \hat{V}_k \) is good if

\[ \| \hat{E} \| \leq (1 + \epsilon)\| E \|. \]
Approximate SVD

1. $Z = XR$
   \[ R \sim \mathcal{N}(d \times r), \ Z \in \mathbb{R}^{n \times r} \]

2. $Q = \text{QR.FACTORIZE}(Z)$

3. $\hat{V}_k \leftarrow \text{SVD}_k(Q^T X)$

**Theorem.** Let \( r = \left\lceil k(1 + \frac{1}{\epsilon}) \right\rceil \) and \( E = X - X\hat{V}_k \hat{V}_k^T \). Then,

\[
\mathbb{E} \left[ \| E \| \right] \leq (1 + \epsilon) \| X - X_k \|
\]

running time is \( O(ndk) = o(\text{SVD}) \)

[BDM, FOCS 2011]
Approximate SVD

$k = 20$  
$k = 40$  
$k = 60$

Exact SVD

Approx. SVD
Sparse PCA

A principal component is a “dense” combination of the feature dimensions. A sparse principal component is a combination of a few feature dimensions. Want $V_k$ to have a few non-zeros in each column

$$V_k = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix}$$
Sparse PCA

1. Choose a few columns $C$ of $X$; $C \in \mathbb{R}^{n \times r}$.
2. Find the best rank-$k$ approximation of $X$ in the span of $C$, $X_{C,k}$.
3. Compute the $\text{SVD}_k$ of $X_{C,k}$:

$$X_{C,k} = U_{C,k} \Sigma_{C,k} V_{C,k}^T.$$

4. $Z = X_{C,k} V_{C,k} = U_{C,k} \Sigma_{C,k}.$

Each feature in $Z$ is a mixture of only the few original $r$ feature dimensions in $C$.

$$\|X - ZZ^\dagger X\| \leq \|X - Z V_{C,k}^T\| = \|X - X_{C,k}\|.$$
Sparse PCA

1. Choose a few columns $C$ of $X$; $C \in \mathbb{R}^{n \times r}$.
2. Find the best rank-$k$ approximation of $X$ in the span of $C$, $X_{C,k}$.
3. Compute the SVD$_k$ of

$$X_{C,k} = U_{C,k} \Sigma_{C,k} V_{C,k}^T.$$  

4. 

$$Z = X_{C,k} V_{C,k}.$$ 

Each feature in $Z$ is a mixture of only the few original $r$ feature dimensions in $C$.

$$\| X - ZZ^\dagger X \| \leq \| X - ZV_{C,k}^T \| = \| X - X_{C,k} \| \leq \left( 1 + O\left( \frac{2k}{r} \right) \right) \| X - X_k \|.$$ 

[BDM, FOCS 2011]
Theorem. One can construct, in $o(\text{svd})$, sparse features that are as good as exact dense PCA-features.
Feature Subset Selection: $K$-Means

Choose a few features

Cluster the data using these features

- PCA - dense features.
- Sparse features: feature subset selection.

Compare the clusterings on all the dimensions.
Theorem. There is a subset of features of size $O(\#\text{clusters})$ which produces nearly the optimal partition (within a constant factor). One can quickly produce features with a log-approximation factor.

[BDM,2013]
Feature Subset Selection: Regression

\[ \hat{Y} = X \cdot w = Y \]
Feature Subset Selection: Regression

**Theorem.** There are $O(k)$ pure features which performs as well regressing on $\text{PCA}_k$ features (to within small additive error).

[BDM, 2013]
The Proofs

All the algorithms use the sparsifier of $V_k^T$ in [BDM,FOCS2011].

1. Choose columns of $V_k^T$ to preserve its singular values.
2. Ensure that the selected columns preserve the structural properties of the objective with respect to the columns of $X$ that are sampled.
   
   (In all cases, the objective is a squared (Frobenius) error.)
Focussed on columns of $V_k^T$ to “sparsify” dimensions.
Can quickly approximate $V_k$.
Can efficiently use it to obtain
  - sparse PCA
  - small subset of features for $k$-means, which results in near optimal clustering.
  - small subset of features for regression, which results regression comparable to PCA$_k$.
Sparse solutions: easy to interpret; better generalizers; faster computations.

Using $U_k$ instead of $V_k^T$ one can “sparsify” data points to get coresets. [BDM,2013]