Active clustering with union of subspace structure

Simons Institute Workshop on Interactive Learning
Feb 16, 2017
Laplace (1749-1827) trained his telescope where “the discrepancy between prediction and observation [was] large enough to give a high probability that there is something new to be found.”
Data Structure
Structure for Messy Data

- **Structured Single Index Models**
  \[ E[y|x] = g(x^T w) \]

- **PCA with heteroscedastic data**

- **Matrix completion or factorization with streaming data**

- **Union of subspace data with missing entries**
Active Learning
Outline

✧ The Union of Subspaces Model

✧ Subspace Margin

✧ Subspace Clustering with Pairwise Active Constraints (SUPERPAC)

✧ Empirical results
Subspace Representations

Sense a length-n vector:

n temperature sensors

n router monitors

n image pixels or features
Subspace Representations

ordered singular values (normalized)
Data are often modeled well by a low-dimensional subspace.

In some ML problems, however, we need a mixture of these spaces.
Union of subspaces
Union of subspaces
Unsupervised methods to cluster these data include:

- **Threshold Subspace Clustering** [Heckel Bolcskei 2013]
- **Greedy Subspace Clustering** [Park Caramanis Sanghavi 2014]

They get classification errors ranging from 8% (SSC for ten Yale faces) to 31% (GSC for ten MNIST digits).
They get classification errors ranging from 8% (SSC for ten Yale faces) to 31% (GSC for ten MNIST digits).

This is still significantly worse than the “Oracle UoS” error of <1% (for ten Yale faces) and 7% (for ten MNIST digits).
Active label selection

Which labels you select in what order has a major impact.
Guarantees

Most algorithms (SSC, GSC, TSC) output an affinity matrix and then use spectral clustering.

Their guarantees build on a clean affinity matrix for spectral clustering. However regularized spectral clustering is now known to succeed provably for input SBM affinity matrices with a sufficient spectral gap in expectation [Coja-Oghlan 2010, Mossel Neeman Sly 2014, Le Levina Vershynin 2017]
The Union of Subspaces Model

Subspace Margin

Subspace Clustering with Pairwise Active Constraints (SUPERPAC)

Empirical results
Clustering: unsupervised

Classification (binary or multi-class): supervised, semi-supervised, or actively supervised

Clustering: metrics for in-class cohesiveness and between-class disparity

Classification: metrics for between-class separation
Classifier margin
Subspace margin
For a subspace $S_k$ with orthogonal projection matrix $P_k$, let the distance of a point to that subspace be

$$\text{dist}(x, S_k) = \|x - P_k x\|_2.$$ 

Let $k^*$ be the index of the true subspace for a point $x \in \mathcal{X}$. Then the margin of $x$ is defined as

$$1 - \max_{j \neq k^*} \frac{\text{dist}(x, S_{k^*})}{\text{dist}(x, S_j)} =: 1 - \mu(x).$$

(1)
Theorem 1. Consider two d-dimensional subspaces $S_1, S_2 \subset \mathbb{R}^D$ with corresponding orthogonal projection matrices $P_1$ and $P_2$. Let $y = x + n$, where $x \in S_1$ and $n \sim \mathcal{N}(0, \sigma^2 I_D)$. Then we have

$$\frac{(1 - \varepsilon)\sqrt{\sigma^2(D - d)}}{(1 + \varepsilon)\sqrt{\sigma^2(D - d) + \|x - P_2 x\|^2}} \leq \mu(y) \leq \frac{(1 + \varepsilon)\sqrt{\sigma^2(D - d)}}{(1 - \varepsilon)\sqrt{\sigma^2(D - d) + \|x - P_2 x\|^2}},$$

with probability at least $1 - 4e^{-c\varepsilon^2(D - d)}$, where $c$ is an absolute constant.

This allows us to prove that for random points, the points near the intersection of the two subspaces have lower margin.

It is well known that near-intersection points are the ones that confound subspace clustering algorithms.
Two d-dimensional subspaces share d principal angles.

\[ \varphi_1 = 0 \]
\[ \varphi_2 = 30^\circ \]
Corollary 2. Let $\phi_i, i = 1, \ldots, d$ be the principal angles between $d$-dimensional subspaces $S_1, S_2 \subset \mathbb{R}^D$. Let $\gamma_i = \sin^2(\phi_i)$ and for $x_1 \in S_1$ fix

$$\|P_2^\perp x_1\|^2 = \gamma_1 + \delta \left(\frac{1}{d} \sum_{i=1}^{d} \gamma_i\right)$$

for some small $\delta$. Let $x_2 \in S_1$ be drawn uniformly from $S_1$ and $y_i = x_i + n_i$ be observations of $x_1, x_2$ with Gaussian additive noise. Then

$$1 - \mu(y_1) < 1 - \mu(y_2)$$

with high probability if

$$\delta < \frac{5}{7} - \frac{1}{\tau}$$

and

$$\gamma_1 + c \leq \frac{1}{\tau} \left(\frac{1}{d} \sum_{i=1}^{d} \gamma_i\right)$$

where $c$ depends only on $D, d$, and the variance of the additive noise.
Outline

-
The Union of Subspaces Model

-
Subspace Margin

-
Subspace Clustering with Pairwise Active Constraints (SUPERPAC)

-
Empirical results
Querying Pairwise Constraints

- The users may not know the labels
- The users may use different languages
SUbsPace clustERing with Pairwise Active Constraints

- **Init**: Affinity matrix from unsupervised clustering.
- **Init**: “Certain Sets” where each set has only examples from a true cluster.

Form Affinity Matrix $A$

Spectral Clustering

Run PCA on Clusters

Select Test Point $x_T = \arg\min_x 1-\mu(x)$

Impute Label Information

Assign $x_T$ to Certain Set $Z_1, Z_2, Z_3$

$x_T$
Outline

✦ The Union of Subspaces Model
✦ Subspace Margin
✦ Subspace Clustering with Pairwise Active Constraints (SUPERPAC)
✦ Empirical results
Algorithms for Comparison to SUPERPAC-R:

- **URASC: Uncertainty Reducing Active Spectral Clustering**
  - Same as our algorithm with no PCA and a different metric for choosing the best query.

- **SUPERPAC-A**
  - Use a query technique based off the affinity matrix only and not subspace projections.

- **Random**
  - Select next query pair completely at random.

- **Oracle UoS**
  - Using oracle labels, compute PCA and then reassign points by closest subspace.
TABLE 3: Average computation time (in seconds) per query required by PCC query selection algorithms on real datasets with 5th/95th quantiles given in parentheses.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Yale, ( K = 5 ) ( N = 320 ) ( D = 2016, d = 9 )</th>
<th>Yale, ( K = 10 ) ( N = 640 ) ( D = 2016, d = 9 )</th>
<th>Yale, ( K = 38 ) ( N = 2432 ) ( D = 2016, d = 9 )</th>
<th>COIL, ( K = 20 ) ( N = 1440 ) ( D = 1024, d = 9 )</th>
<th>COIL, ( K = 100 ) ( N = 7200 ) ( D = 1024, d = 9 )</th>
<th>USPS, ( K = 10 ) ( N = 9298 ) ( D = 256, d = 15 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUPERPAC-R</td>
<td>1.40 (1.38/1.43)</td>
<td>2.78 (2.76/2.79)</td>
<td>10.42 (9.57/10.98)</td>
<td>0.44 (0.37/0.48)</td>
<td>5.78 (5.53/6.02)</td>
<td>0.19 (0.17/0.20)</td>
</tr>
<tr>
<td>SUPERPAC-A</td>
<td>1.37 (1.35/1.39)</td>
<td>2.73 (2.71/2.76)</td>
<td>9.36 (8.72/9.91)</td>
<td>0.30 (0.23/0.34)</td>
<td>1.68 (1.50/1.79)</td>
<td>0.05 (0.05/0.06)</td>
</tr>
<tr>
<td>URASC-N</td>
<td>0.11 (0.08/0.13)</td>
<td>0.28 (0.23/0.40)</td>
<td>6.38 (5.35/7.22)</td>
<td>4.61 (2.58/5.55)</td>
<td>252.97 (110.63/356.49)</td>
<td>155.02 (53.19/190.86)</td>
</tr>
</tbody>
</table>

SUPERPAC is efficient with large \( N \), small \( D \).

URASC is more efficient with large \( D \), small \( N \).
USPS

number of pairwise comparisons

missclassification %

0 500 1000 1500 2000

SUPERPAC-R
SUPERPAC-A
URASC-N
Random
Oracle UoS
Cluster jumps: COIL

number of pairwise comparisons

missclassification %

0 2 4 6 8 10 12 14 16

0 200 400 600

0 2000 4000

0 50 100 150

SUPERPAC-R
SUPERPAC-A
URASC-N
Random
SUPERPAC-S
Oracle UoS

COIL-20

COIL-100

COIL-20 Smoothing
More experiments

- Sonar
- Balance
- Leaf

Graphs show the missclassification percentage against the number of pairwise comparisons. Different datasets (Sonar, Balance, Leaf) are compared across various algorithms:

- SUPERPAC-R
- URASC-N
- Active Random
- Random

The graphs illustrate how each algorithm performs in terms of reducing missclassification as more comparisons are made.
Conclusion

 assertions

 - Low-rank signal structure helps in many problems
  - (and does not seem to hurt)

 - Subspace margin provides a metric for nearness to subspace intersection

 - Algorithm theory?
Thank you!

Questions?