Near-optimal sample compression for nearest neighbors

Lee-Ad Gottlieb¹, Aryeh Kontorovich², Pinhas Nisnevitch¹

¹Ariel University, Ariel, Israel ²Ben-Gurion University of the Negev, Beer Sheva, Israel

Cons of kernel: ➔ hyperplane separator
• Assumes Euclidean distances
• Natural distances often highly non-Euclidean

Pros of nearest neighbors:
• Simple, classic learning algorithm (early 50’s)
• Requires minimal structure
• Immediate extension to multiclass
• Well-understood consistency properties

See:
• In Defense of Nearest-Neighbor Based Image Classification” (Boiman et al., 2008)
• Often yields competitive results” (Weinberger and Saul, 2009)

2. Sample condensing: for 1-NN?

Consistent condensing
• Input: Sample \( S = S_n \cup S_1 \) in a metric space \( (X, \rho) \)
• Condensing: \( \tilde{S} \subseteq S \)
• Consistency: For any point \( x \notin \tilde{S} \)'s nearest neighbor in \( \tilde{S} \) has the same label as \( x \)

Nearest neighbor condensing problem
• Input: Sample \( S \)
• Output: Minimal consistent \( \tilde{S} \subseteq S \)
• NP-hard (Willings, 1991; Zuckha, 2010)

Heuristic solutions
• Hart (1968) heuristic
  – Init \( \tilde{S} = 0 \)
  – Greedily add misclassified points of \( S \) to \( \tilde{S} \)
  – Runtime: \( O(n^3) \)
• Other proposed heuristics: Gates (1972); Ritter et al. (1975); Wilson and Martinez (2000)
• Theoretical guarantees: none

3. Benefits of sample compression

Sample condensing
• Pro: Reduced memory usage
• Pro: Faster evaluation on new points
• Pro: Improved generalization bounds

Sample size bound (Graepel et al., 2005)
• wzphp/}
  \( \epsilon \)
holds whenever \( G(\epsilon) = 0 \)

4. Background to doubling dimension

Definition: Doubling dimension
• For any metric space \( X \)
• Doubling constant of \( X \): Minimum \( \lambda \)
  such that every big ball of diameter \( b \) can be covered by \( \lambda \) small balls with diameter \( b/2 \)
• Doubling dimension of \( X \): \( \log \lambda \)

History
• Introduced by Assouad (1983)
• Generalizes Euclidean dimension
• Used algorithmically by Clarkson (1999)
  ... and others

Definition: \( \epsilon \)-net
• Subset \( S' \subseteq S \)
  • Packing property: Minimum inter-point distance in \( S' \) is \( \epsilon \)
  • Covering property: Every point in \( v \in S \) satisfies \( d(v, S') < \epsilon \)
  • Construction time: \( O(S(\log(S)))(\log(1/\epsilon)) \) Krathagam and Lee (2004)

5. Main result: Near-optimal sample compression for NN

Definition \( \gamma \) = \( \min(\gamma) = \rho(S_n, \tilde{S}_\gamma) \)
Our condensing algorithm: Build a \( \gamma \)-net

Theorem: Suppose \( \rho(\tilde{S}) = 1 \) and \( \gamma(\tilde{S}) > 0 \).
There exists an algorithm that in time
\( \min \{ |\tilde{S}|^2, \rho(\tilde{S})(\log(1/\gamma)) \} \)
computes a consistent set \( \tilde{S} \subseteq S \) of size
\( \lfloor \gamma^{-1}\log(1/\gamma) \rfloor + 1 \)

Lower bounds: Algorithm close to best-possible

• Theorem: Unless \( P=NP \), cannot approximate \( S' \) within factor \( \rho(S')(\log(1/\gamma))^{\omega(1)} \)
• More precisely:
  1. There exists \( S \) with minimal consistent \( S' \subseteq S \)
  2. It is NP-hard to find any consistent set of size
\( |\tilde{S}|^2, \rho(\tilde{S})(\log(1/\gamma))^{\omega(1)} \)
    • almost matches upper bound \( \log(1/\gamma) \)

6. Net construction algorithm

Require: \( S \)
1. \( p \leftarrow \) arbitrary point of \( S \)
2. \( S_0 \leftarrow \{p\} \)
3. \( C(p, 0) = N(p, 0) \leftarrow \{p\} \)
4. for all \( q \in S \) do
   5. \( P(q, 0) \leftarrow p \)
   6. end for
7. for \( i = 0, 1, \ldots, \lfloor \log_3 \gamma \rfloor + 1 \) do
   8. for all points \( l \) present in level \( i \) - 1
   9. \( S_{i-1} = S_i \)
   10. end for
11. for all points \( l \) present in level \( i \) - 1
   12. for all \( p \in S_{i-1} \) do
   13. \( C(p, i-1) = \emptyset \)
   14. end for
15. for all \( q \in S_i \) do
16. for all points \( l \) present in level \( i \) - 1
17. \( T \leftarrow \cup_{r \in S_i} C(p, r) \)
   18. end for
19. for all \( q \in T \) with \( \rho(q, i) < 2^{-i} \) do
   20. \( S_{i-1} \leftarrow S_{i-1} \cup \{q\} \)
   21. end for
22. Update child list of \( q \)’s parent
23. end for
24. end for
25. end for
26. for all \( q \in S_i \) do
27. end for
28. end for
29. end for
30. end for

7. Hardness lower bound

NP-hard to find a better compressed set

Reduction:
• From Label Cover problem.
• Reduction holds for Euclidean sets too

Label cover:
• Input: Graph, set of valid labels
• Output: Valid labelling
• Minimization version: Can use multiple colors, minimize number of labels
• Dinur and Safra (2004) showed NP-hard to approximate within a factor \( \rho(S)/(\log(1/\gamma)) \)

8. Empirical results

Experiments:
• Data sets: UCI Machine Learning Repository
• Metric: \( \ell_1 \)-norm

Table:

<table>
<thead>
<tr>
<th>Data set</th>
<th>Original sample</th>
<th>In net</th>
</tr>
</thead>
<tbody>
<tr>
<td>Skin Segmentation</td>
<td>16000</td>
<td>130.10</td>
</tr>
<tr>
<td>Staging Shuttle</td>
<td>2000</td>
<td>65.75</td>
</tr>
<tr>
<td>Covertype 1 vs. 4</td>
<td>2000</td>
<td>35.85</td>
</tr>
<tr>
<td>Covertype 4 vs. 6</td>
<td>2000</td>
<td>94.50</td>
</tr>
<tr>
<td>Covertype 4 vs. 7</td>
<td>2000</td>
<td>4.40</td>
</tr>
</tbody>
</table>