Heuristic Prototype Selection for Regression
Debraj Basu, Deepak Pai, Joshua Sweetkind-Singer

Motivation
k-NN regression faces
- High storage costs
- Expensive computation
- Sensitivity to outliers
A smaller set of prototypes can afford
- Lower storage and compute costs
- Robustness to outliers
- Better generalization

Properties
- Prototypes are representative points that exhibit certain desirable properties
- The neighborhood of a prototype is determined by an $\epsilon$ covering ball around
- These covering balls are expected to
  - Cover many points with similar labels
  - Avoid covering points with dissimilar labels
- Proximity in label space is defined by a customizable metric
- These properties are governed by the size of the covering balls, and automatically induce sparsity in the data

Description: Methods which leverage the separation of samples based on class have verified the above benefits for classification
This work generalizes the set-cover based approach by Bien et al. 2012 to identify representative points in the regression setting as prototypes

Formulation
For $n$ samples, the properties are encoded into a mixed-integer program with indicator variables $a_i$ for each sample, and slack variables $\eta_i$ and $\xi_i$ where $i \in \{1, \ldots, n\}$

\[
\begin{align*}
\min & \quad \sum_{i=1}^{n} \alpha_i + \sum_{i=1}^{n} \eta_i + \lambda \sum_{i=1}^{n} \alpha_i \\
\text{s.t.} & \quad \sum_{j: x_j \in B(x_j)} \left( 1 - \frac{\|y_i - y_j\|}{\Delta Y_{\max}} \right) \geq 1 - \xi_i \quad \forall i \in [n] \\
& \quad \sum_{j: x_j \in B(x_j)} \frac{\|y_i - y_j\|}{\Delta Y_{\max}} \leq \eta_i \quad \forall i \in [n] \\
& \quad \alpha_i \in \{0, 1\}, \quad \xi_i, \eta_i \geq 0 \quad \forall i \in [n]
\end{align*}
\]

Heuristic Prototype Selection
The set of prototypes is successively refined by greedily selecting data points based on the objective
It terminates when the objective cannot be improved anymore

1: Initialize $P = \emptyset$
2: do
3: \quad Find $i' \leftarrow \arg \max_{i \in [n]} \Delta \text{Obj}(x_i)$
4: \quad \text{If } \Delta \text{Obj}(x_{i'}) > 0 \text{ then do } P \leftarrow P \cup \{i'\}
5: \quad \text{while } \Delta \text{Obj}(x_{i'}) > 0$
6: return $P$

Comments
- One hot encoding reduces the formulation and solution to multi-class classification (Bien et al. 2012)
- Time Complexity $O(n P \max C(I))$; $P$ is the number of prototypes and $C(I)$ is the number of neighbors of $x_i$
- Varying $\lambda$ enables data set condensation by controlling the number of prototypes

Prediction Performance
Experiments
For each minimum allowable % compression, our greedy solution exhibits the best prediction performance.
This plot summarizes results from extensive cross validation done across a range of $k$ nearest neighbors, $\epsilon$ covering balls

Tolerance to Outliers
Prototypes are selected from increasingly corrupted data sets
Prototypes selected by our method exhibit more robust performance
Methods which utilize the labels are observed to be more tolerant to outliers