

Clustering Inside Classes Improves Performance of Linear Classifiers

Dmitriy Fradkin
Siemens Corporate Research

Overview

- The idea of using unsupervised clustering as a supplement to supervised classification has been used informally since early 1960's.
- In the early 90s, the machine learning community started actively investigating local learning, classifier ensembles, mixtures of experts or input partitioning [Jacobs et al. 1991, Xu et al. 1995]. All of these approaches involved (implicit or explicit) clustering of points across the classes, i.e. without considering the labels.
- A natural question in this context is whether the class label information can be used in the clustering in a way that would improve the classification.

We analyze Clustering Inside Classes (CIC) approach:

- show that it can lead to improved accuracy over the state of the art linear classifiers.
- evaluate the role of the size of the training set
- evaluate the effect of number of clusters per class
- experiment with an automated method for choosing the number of clusters

Clustering Inside Classes (CIC)

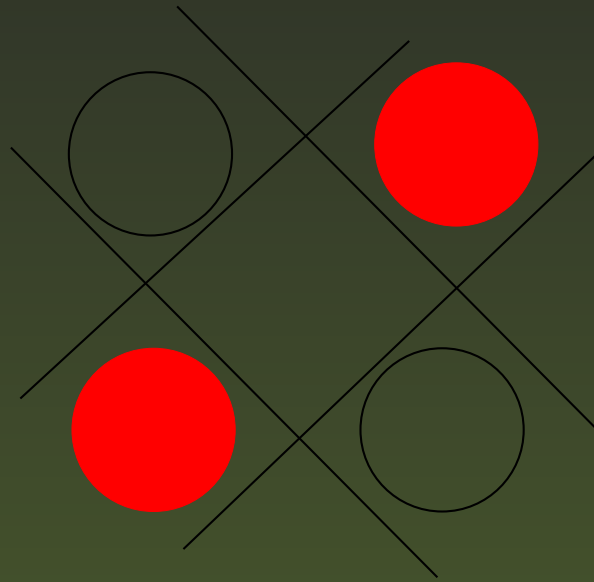


Figure 1: The two classes, represented on a plane with red and white circles, each consist of two clusters/subclasses. The lines demonstrate that each cluster/subclass can be easily separated from the others, while the classes themselves are not linearly separable.

CIC Approach

Algorithm:

- Partition each class into clusters.
- Train a classifier to recognize these clusters.
- When a new point appears, assign it to a cluster.
- Each cluster corresponds to a single class, therefore the point is assigned to that class.

The idea seems counterintuitive: the number of classes the classifier has to handle increases, seemingly making the problem more difficult. However, the new classes (clusters) are “compact” and should be easily separable. Also, mistakes between clusters of the same class do not matter.

Related Work

- CIC is sometimes used in the initialization of supervised learning algorithms [Dong et al., 2002], such as Generalized Learning Vector Quantization (GLVQ) [Sato and Yamada, 1998].
- CIC classification approach was first described in [Japkowicz, 2002]. However that work only examined combinations (bagging and boosting) of several CIC classifiers obtained from multiple partitions of the data. It also did not address the choice of k , leaving it up to the user, or evaluate CIC completely on its own merits.
- CIC was examined in [Vilalta et al. 2003] as a way to address weaknesses of Naive Bayes classifier. Parameter k was selected with cross-validation.
- The effect of k and training set size was examined in [Fradkin 2006] , but selection of k was left to the user.
- Recently, [Wu et al. 2007] considered CIC as a way to compensate for class imbalances and for rare classes.

Experiments on UCI Datasets

Dataset	Classes	Dimensions	Training Set Size	Training Class Size	Test Set Size	Test Class Size
Image	7	19	210	30	2100	300
Pendigit	10	16	7494	[719,780]	3498	[335,364]
Satimage	6	36	4435	[415,1072]	2000	[211,470]
Vowel	11	10	528	48	462	42

Table 1: Summary of properties of the benchmark datasets

All features were independently normalized to have 0 mean and standard deviation 1.

Methods Used

Classification Methods:

- Support Vector Machines (SVM) [Vapnik, 1995], a maximum margin classifier. A particular implementation - LIBSVM [Chang and Lin, 2001].
- Bayesian Logistic Regression - depending on the choice of parameters equivalent to Ridge or to Lasso logistic regression. BBR and BMR programs developed at DIMACS by [Genkin, Madigan and Lewis, 2004].

Clustering Methods:

- K-Means [Forgy, 1965; MacQueen, 1967].

Selecting the number of clusters:

- X-Means [Pelleg and Moore 1999]. X-Means implements a fast version of K-Means using pre-computed kd-trees and allows automated selection of the number of clusters based on BIC (only the upper limit on k needs to be specified).

Results of CIC Approach

SVM

k	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 8$	$k = 10$	$k = 15$	X
image	92.10	91.71	89.38	89.90	89.71	89.57	90.19	89.81	89.67	89.91
pendigit	95.85	97.48	97.51	98.00	97.71	97.74	97.88	97.68	97.57	97.97
satimage	86.05	85.90	87.30	88.45	89.35	89.45	89.75	89.60	90.15	90.00
vowel	51.08	53.25	56.06	56.71	52.16	54.11	51.52	53.68	51.30	53.25

BMR

k	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 8$	$k = 10$	$k = 15$	X
image	91.67	91.19	88.95	87.33	88.33	88.81	89.62	89.90	88.95	88.86
pendigit	92.08	95.83	95.71	96.17	96.23	96.54	96.31	96.57	96.43	96.57
satimage	83.90	84.35	85.30	86.90	87.90	87.65	87.45	87.10	87.05	87.25
vowel	46.54	49.13	51.52	46.10	46.32	48.05	45.24	44.59	45.89	48.05

Table 2: Red indicates significance at 0.999 confidence level compared to $k = 1$; Bold shows best results in each row

Observations

- On all datasets except for Image CIC leads to improvement in classification accuracy for at least some values of k .
- Results with SVM and BMR are qualitatively similar
- Notice that the value of k for which the best results are obtained appears related to the number of points per class in the training set. Intuitively, increasing the number of clusters leads to clusters with fewer point, which in turn makes it difficult to train good classifiers for distinguishing them.
- However these differences in performance are not completely due to the training set size, but are also affected by the intrinsic structure of the classes.
- Using X-Means does not give the best results, however where improvement with CIC is possible, using X-Means to determine the number of clusters also leads to an improvement.

CIC with RBF SVM

k	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$	$k = 8$	$k = 10$	$k = 15$	X
image	92.71	91.14	90.38	90.67	90.33	90.19	90.19	90.00	89.48	90.42
pendigit	97.80	97.68	97.74	97.74	97.94	97.71	97.88	97.68	97.48	97.77
satimage	91.60	91.55	91.05	90.55	91.60	91.70	91.65	90.95	90.30	90.65
vowel	66.45	59.31	56.49	56.49	54.98	54.55	51.73	52.38	50.87	51.73

Table 3: Red indicates significance at 0.999 confidence level compared to $k = 1$; Bold shows best results in each row

- The results of the RBF SVM alone are better than those of linear classifiers.
- Using CIC with RBF SVM did not result in significant improvements on any dataset.

Non-linear methods such as RBF SVM can represent more complicated decision boundaries - and separate odd-shaped classes better - and therefore stand to benefit less from CIC than linear methods.

Effect of the Training Set Size

Setup:

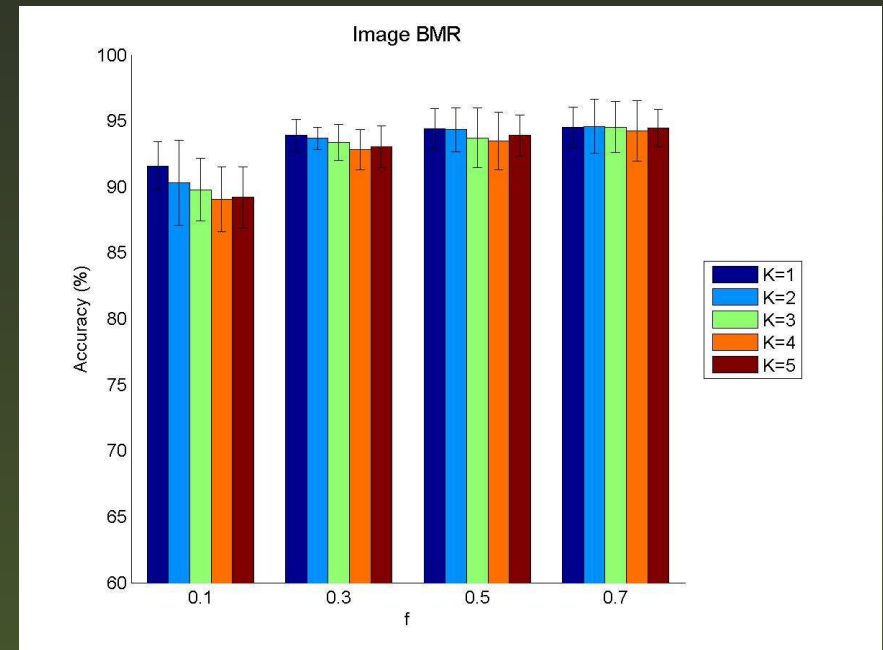
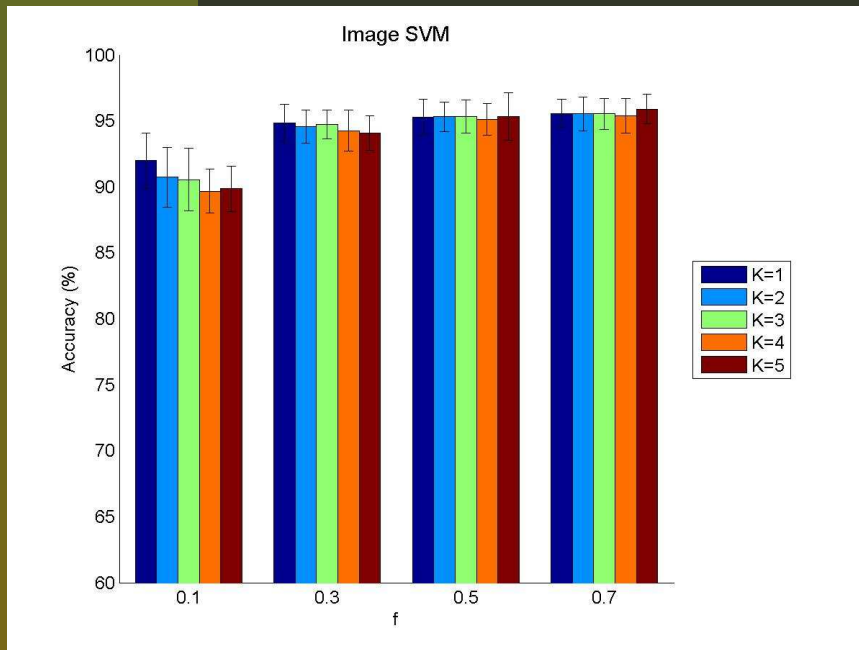
- Vary fraction of the data f used for training: 0.03, 0.1, 0.3, 0.5, 0.7. Also, observe correlation with the number of points per class
- Randomly partition data 10 times for each f and evaluate

Note that the number of points per class is approximately the same for Pendigit and Satimage. For the Image dataset however, with $f = 0.3$ number of points per class is approximately the same as for Pendigit and Satimage with $f = 0.1$; and $f = 0.1$ is comparable to $f = 0.03$ for the larger datasets.

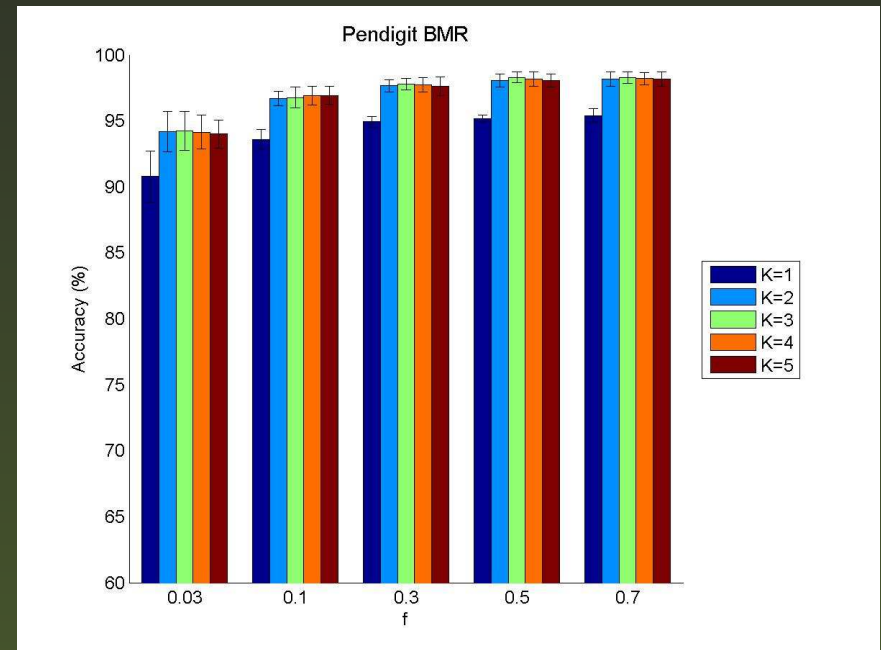
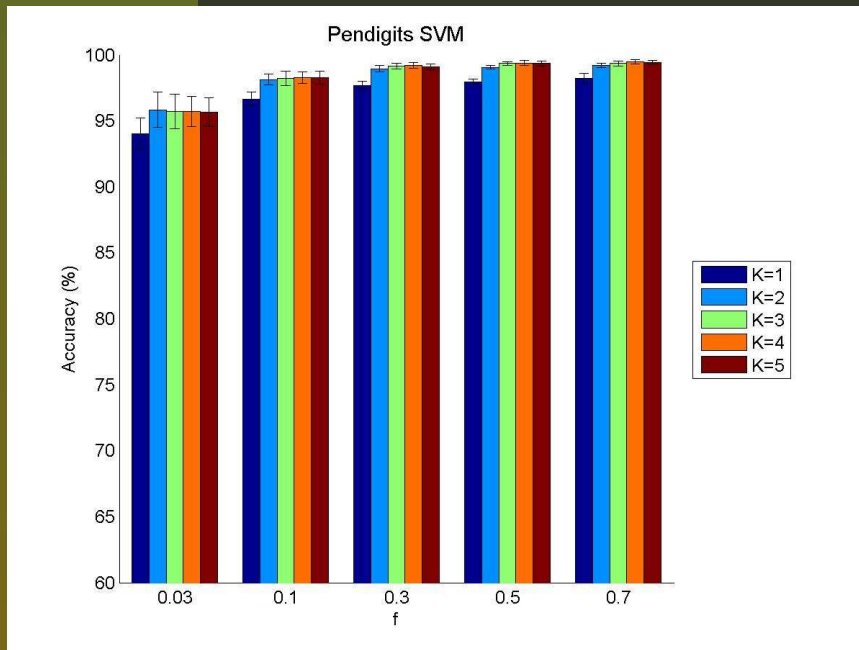
Observations:

- On Pendigit and Satimage datasets CIC improves results for almost any setting of parameters.
- On the Image dataset CIC does not decrease accuracy when the training set was sufficiently large (50% of the data or 175 points per class in the training set).
- Failure of CIC to produce improvements on Image suggests that classes there have a single intrinsic component.

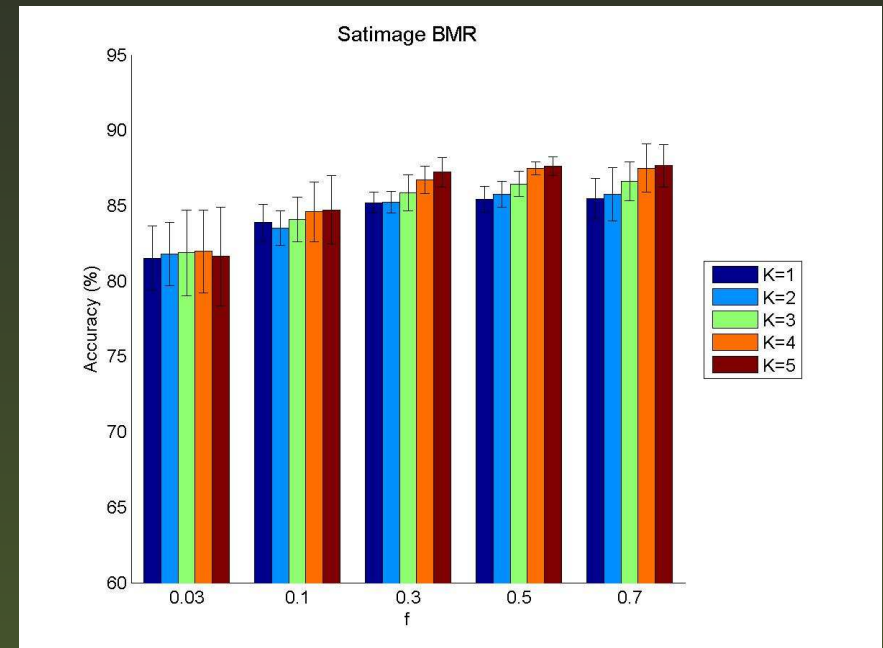
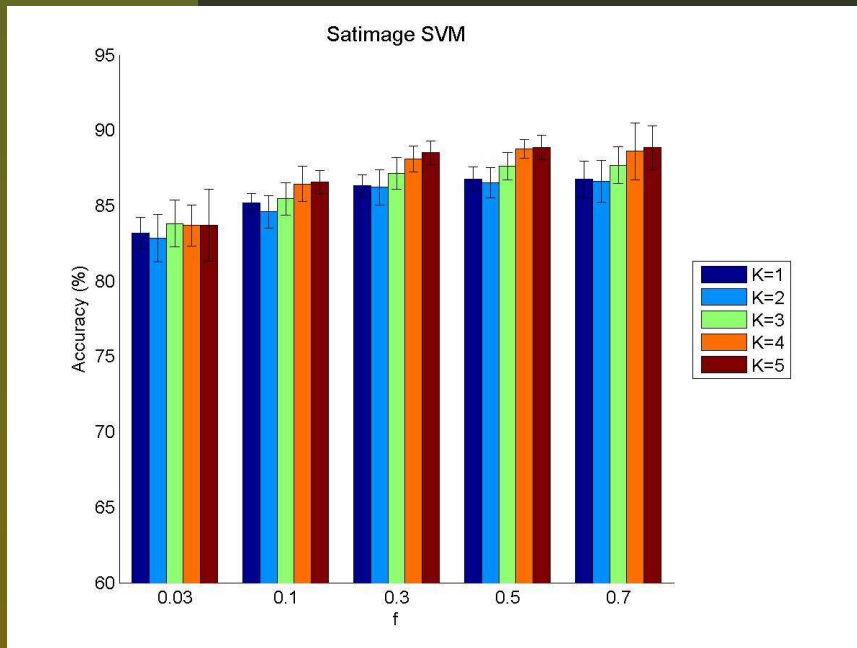
Multiple Splits: Image



Multiple Splits: Pendigits



Multiple Splits: Satimage



Conclusions

- CIC improves accuracy of linear classifiers provided that sufficient training data is available and classes have more than one intrinsic component. Even when classes appear to have only one such component, CIC does not cause performance to deteriorate with sufficient training data for each class.
- The choice of k does affect the results, though there tend to be whole ranges of values of k for which CIC improves performance of linear classifiers.
- Using X-Means approach for automated selection of k improved results compared to a single global classifier. These results were only slightly worse those with the best values of k selected for each dataset *a posteriori*.
- The accuracy improvements obtained with CIC and linear classifiers do not match results of a non-linear classifier. However, CIC with linear classifiers may have advantages in speed, parallelization and interpretability.

Future Work

- Combine CIC with non-linear classifiers by performing cluster analysis in a non-linear space (ex. [Girolami 2002]).
- The datasets examined in our work arose in signal-processing applications. Text and bioinformatics datasets are frequently characterized by large number of features, sometimes exceeding the number of available training points. While we believe that the CIC approach will also be successful in such applications, the characteristics of the data would have to be taken into account, most likely by utilizing a more appropriate clustering approach.
- Develop visualization/representation methods for the extracted class structure (i.e. within-class clusters) as a data exploration method.