Pattern Recognition
CS-688
Fall 2010

Instructor: Carlotta Domeniconi

Ideas for Projects

Project idea 1

Learning Adaptive Metrics for Classification or Clustering

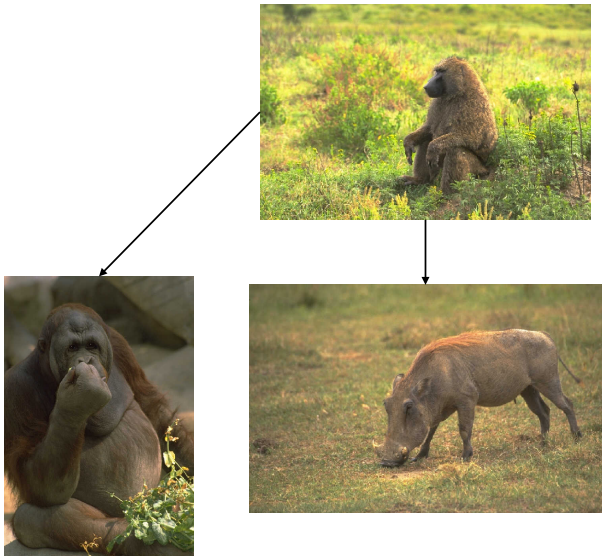
Why Learn Distance functions?

Nearest Neighbor

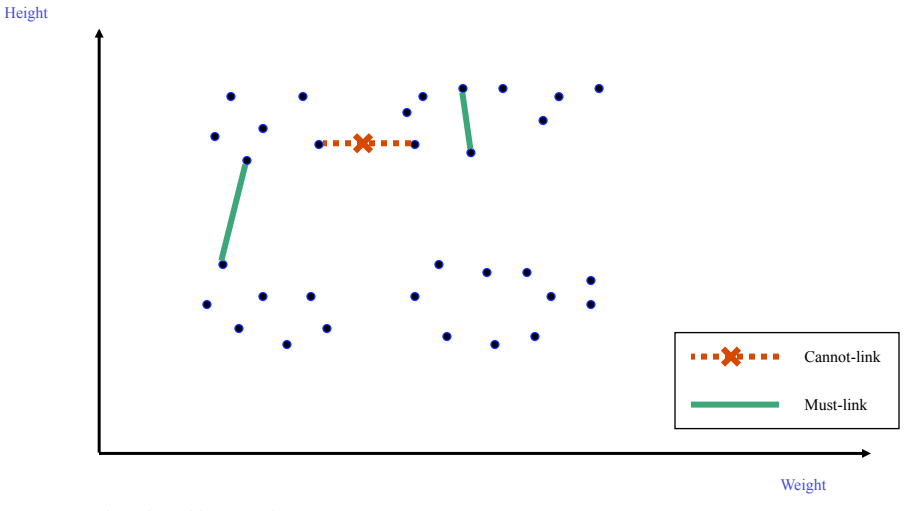
Image retrieval

Given a query image return the K-nearest neighbors of the image from the database.

Euclidean distance on Color Coherence Vectors returns both images as similar to query image



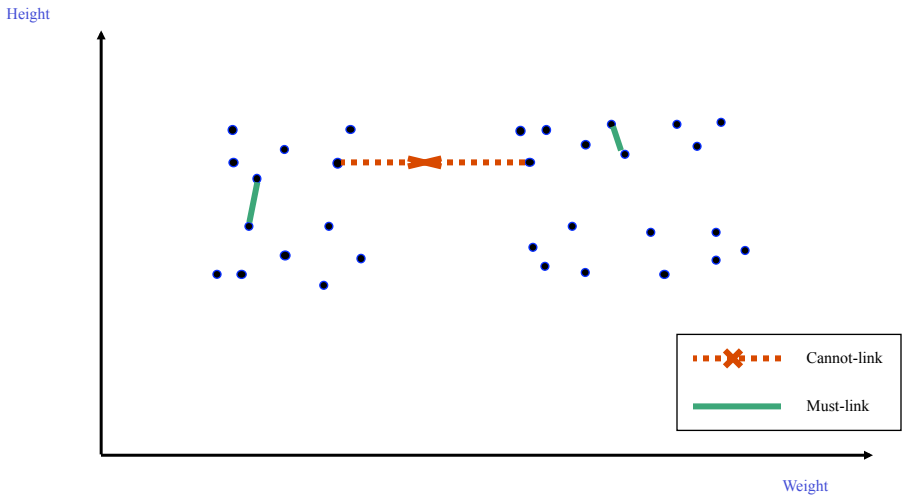
Example: Learning Distance Function



Clustering with Constraints

Example: Learning Distance Function

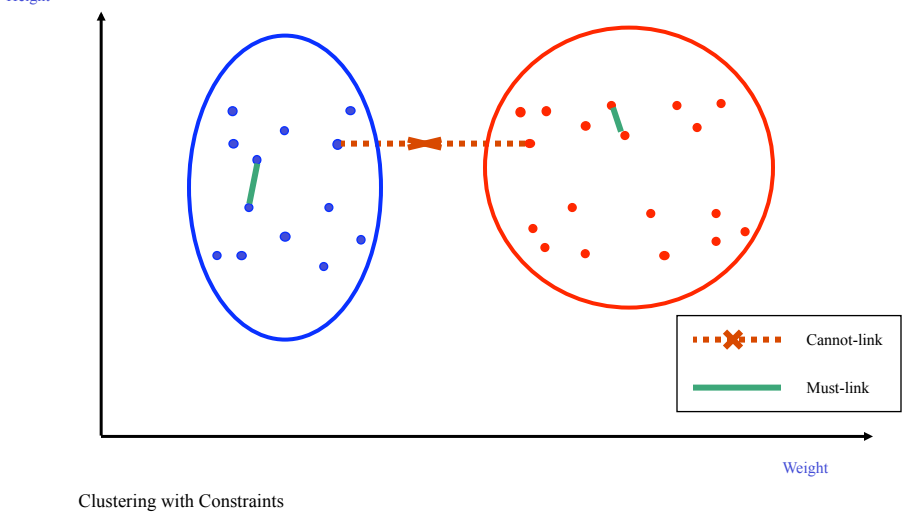
Space Transformed by Learned Function



Clustering with Constraints

Example: Learning Distance Function

Clustering with Trained Function



Clustering with Constraints

Learning Distance Techniques

- Machete (96)
- Scythe (96)
- Discriminant Adaptive Nearest Neighbor (DANN) (IEEE-PAMI 98)
- ADAptive MEtric Nearest Neighbor (ADAMENN) (IEEE-PAMI 02)
- LArge MArgin Nearest Neighbor Algorithm (LAMANNA) (NIPS -02, IEEE-TNN 05)
- Distance Metric Learning for Large Margin Nearest Neighbor Classification (NIPS 05)
- And more…

Learning Distance Techniques

- The literature lacks a survey or comparison of these techniques:
  - The goal is similar, but the way it is achieved is different
  - Exploring underlying assumptions
  - Achieve a better understanding of limitations and advantages of some of the recent methods
  - Considering derived variables: local vs. global. Does it help?
  - Very high dimensionality: do the methods still work?
  - Modify existing approaches to improve scalability or/and accuracy.
  - Come up with better methods all together

Learning Distance Techniques

– Making a *global* approach *local*:
  - Weinberger et al. approach is *global*: a *global* linear transformation is learned to optimize K-NN classification.
  - A cost function is minimized that penalizes *large distances* between each input and its *target neighbors*, and at the same time penalizes *small distances* between each input and all other inputs that do not share the same label.
  - We could make the transformation local to a test point ***t*** by considering a neighborhood of ***t***, and minimizing the cost for all its neighbors

Learning Distance Techniques

– References:
  - J Friedman, **Flexible Metric Nearest Neighbor Classification**, Stanford University, *Department of Statistics Tech Report*, 94.
  - T Hastie and R Tibshirani, **Discriminant Adaptive Nearest Neighbor Classification**, *IEEE T. on PAMI*, 96.
  - C Domeniconi, D Gunopulos, and J Peng, **Locally Adaptive Metric Nearest Neighbor Classification**, *IEEE T on PAMI*, 02.
  - C. Domeniconi, D. Gunopulos, and J. Peng, **Large Margin Nearest Neighbor Classifiers**, *IEEE T on NN,* 05.
  - K Weinberger, J Blitzer, L Saul, **Distance Metric Learning for Large Margin Nearest Neighbor Classification**, *NIPS 05*.

Project idea 2

Reverse Nearest Neighbor

Reverse Nearest Neighbor

- $N_k(x)$ the number of times x occurs among the k NNs of all other points in the data.
- The concept of reverse nearest neighbor can be used for classification, clustering, and anomaly detection.
- Recently, properties of RNN in high dimensional spaces have been explored:
  - Emergence of hubs: points which appear in many more k-NN lists than the others.

Reverse Nearest Neighbor

More analysis needed:

- Exploitation of RNN properties in high dimensional spaces to design effective classification/clustering/anomaly detection techniques
- Develop distance learning techniques based on RNNs
- Leverage hubs for seeding iterative clustering algorithms like k-means

Reverse Nearest Neighbor

References:

M. Radovanovic et al., Nearest Neighbors in High-Dimensional data: The Emergence and Influence of Hubs. International Conference on Machine Learning, 2009.

Project idea 3

Co-training, Tri-training and Co-evolution

Semi-supervised learning

- In many data mining applications, *unlabeled data* are easily available, while *labeled* ones are expensive to obtain because they require human effort
- For example: Web page classification; Content-based image retrieval
- *Semi-supervised learning is a recent learning paradigm*: it exploits unlabeled examples, in addition to labeled ones, to improve the generalization ability of the resulting classifier

Co-training paradigm

- Two classifiers are trained separately on two different views of the same problem, i.e., two independent set of features
- The predictions of each classifier on unlabeled data are used to augment the training set of the other classifier
- Under certain assumptions, the co-trained classifiers can make fewer generalization errors by maximizing their agreement over the unlabeled data

Tri-training paradigm

- A co-training style semi-supervised learning algorithm
- It exploits unlabeled data using three classifiers
- Three *different* classifiers are generated from the original set of labeled data (bootstrap sampling)
- These classifiers are then refined in each round of tri-training: an unlabeled example is labeled for a classifier if the other two agree on the labeling
- The final hypothesis is produced via *majority voting*

Tri-training paradigm (contd)

- Results on UCI data and on Web page classification show that unlabeled data can be exploited effectively
- Issues to explore:
  - Better performance can be expected with more classifiers
  - Different learning algorithms can be used to train different classifiers
  - Exploring ensemble learning techniques for a semi-supervised setting
  - In combination with *active learning*: labels for selected unlabeled examples are asked from the user

Co-training and Co-evolution: investigating the connection

- **Co-evolution**: Form of evolutionary computation in which the fitness evaluation of an individual is based on interactions between multiple individuals
- Thus, an individual's ranking in a population can change depending on other individuals

Coevolutionary Algorithms (CEAs)

- Very similar to traditional EA methods
  - Individuals encode aspect of potential solutions
  - They are altered during search with genetic operators
  - Search directed by selection based on fitness
- But differ in fundamental ways:
  - Evaluation requires interaction between multiple individuals
  - Interacting individuals may reside in same population or in different populations
  - Evokes notions of cooperation and competition
  - Methods of evaluation are particularly important

Cooperation & Competition

- **Cooperative algorithms** are those in which interacting individuals succeed or fail together.
  Example: Coevolving a multiagent team responsible for jointly defending a resource (*solution: Team behaviors*)

- **Competitive algorithms** are those in which individuals succeed at the expense of other individuals.
  Example: Coevolving a classifier and challenging datasets (*solution: general classifiers*)

Co-training and Tri-training are examples of CEAs

- The "fitness" (or quality) of a classifier in co-training or tri-training is **not** evaluated in isolation, but depends on the fitness of the other classifiers (by means of the propagated training data)
- The interactions seem to be cooperative: the higher the fitness of an individual classifier, the higher the quality of the information distributed to the other individuals. Is there also a competitive component?
- Cooperative coevolution is usually used when the given problem can be split into smaller sub-problems. In co-training or tri-training either the feature space is partitioned into two disjoint sets, or the data are bootstrapped into multiple subsets. Each individual has a different view of the same problem.

Pathologies

- Pathologies have been observed and studied in coevolutionary algorithms:
  - Coevolution sometimes fail to produce desired results
  - ***Cycling***: typically refers to an oscillation in some metric of algorithm behavior
- Are there pathologies in the dynamics of co-training or tri-training? Oscillations? Convergence? Under which conditions? Can we use techniques developed within COEs to understand this?

References

- A. Blum and T. Mitchell, **Combining Labeled and Unlabeled Data with Co-training**, *Proc. 11th Ann. Conference on Computational Learning Theory*, 1998.
- Z. Zhou and M. Li, **Tri-training: Exploiting Unlabeled Data Using Three Classifiers**, *IEEE Transactions on Knowledge and Data Engineering*, 17(11), 2005.
- Y Zhou and S Goldman, **Democratic Co-Learning**, *16th IEEE International Conference on Tools with Artificial Intelligence*, 2004
- E de Jong, K Stanley, P Wiegand, **Introductory Tutorial on Coevolution**, *GECCO 2007*
- E de Jong, K Stanley, P Wiegand, **Advanced Tutorial on Coevolution**, *GECCO 2006*

Data from The US Bureau of Labor Statistics

BLS data

- Data on:
  - Inflation & Prices;
  - Employment;
  - Unemployment; etc.

- Use historical data to make useful prediction as a decision support system to users