On Classification: An Empirical Study of Existing Algorithms Based on Two Kaggle Competitions

CAMCOS Report Day

December 9th, 2015

San Jose State University

Project Theme: Classification

The Kaggle Competition

- Kaggle is an international platform that hosts data prediction competitions
- Students and experts in data science compete
- Our CAMCOS team entered two competitions

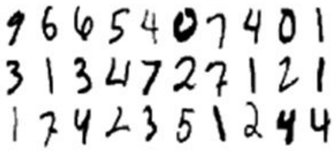
Team 1: Digit Recognizer (Ends December 31st)

Team 2: Springleaf Marketing Response (Ended October 19th)

Overview of This CAMCOS

Team 1

Wilson A. Florero-Salinas
Carson Sprook
Dan Li
Abhirupa Sen



Problem: Given an image of a handwritten digit, determine which digit it is.

Team 2

Xiaoyan Chong
Minglu Ma
Yue Wang
Sha Li

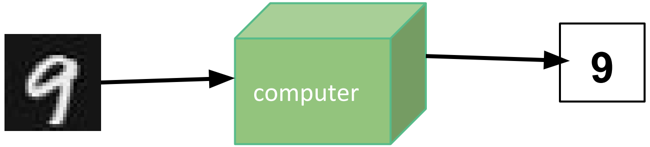
Springleaf
Lending made personal

Problem: Identify potential customers for direct marketing.

Project supervisor: Dr. Guangliang Chen

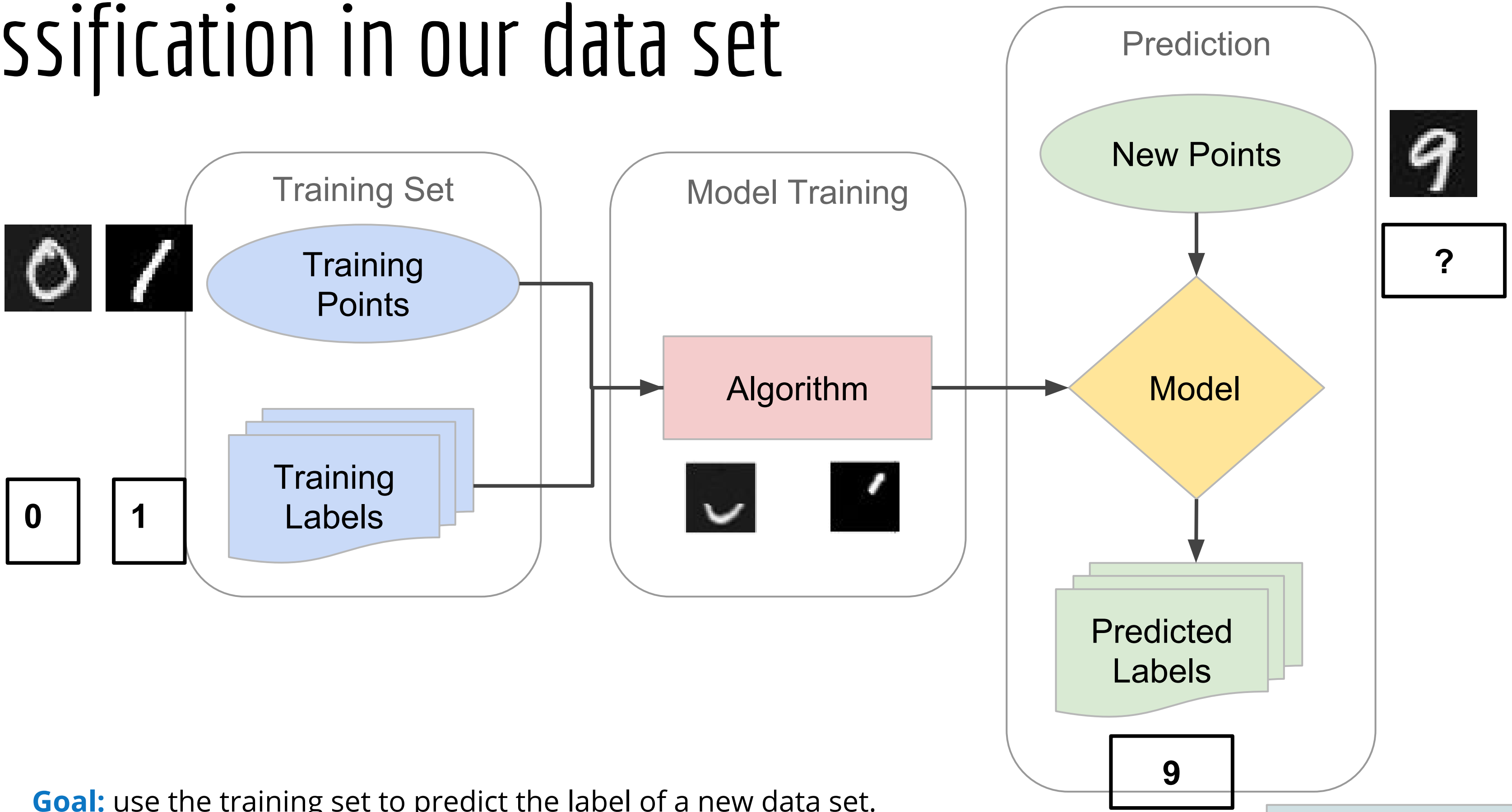
Presentation Outline (Team 1)

1. **The Digit Recognition Problem**
2. **Classification in our Data Set**
3. **Data Preprocessing**
4. **Classification Algorithms**
5. **Summary**



Theme: Classification Problem

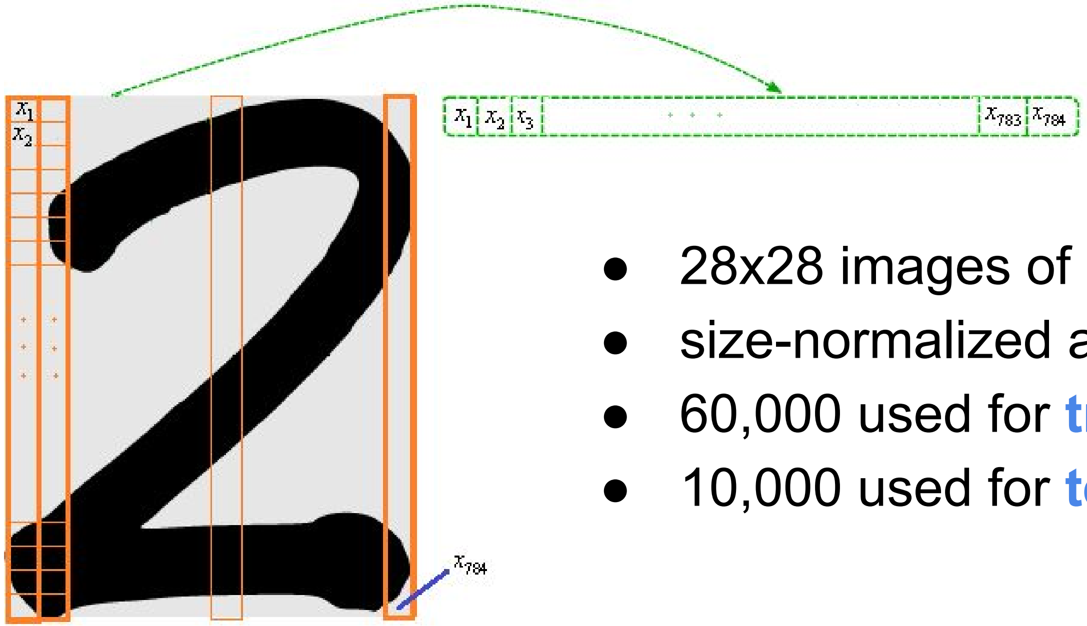
Classification in our data set



- **Goal:** use the training set to predict the label of a new data set.

MNIST Data

Team 1: The MNIST[1] data set

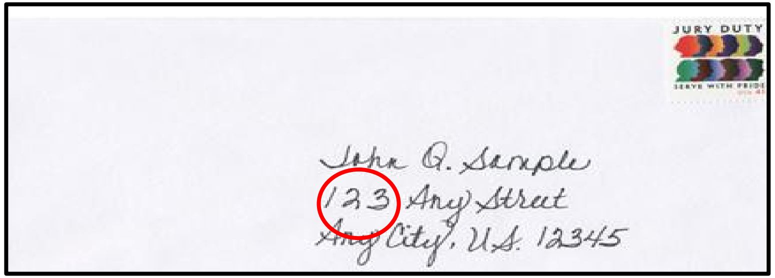


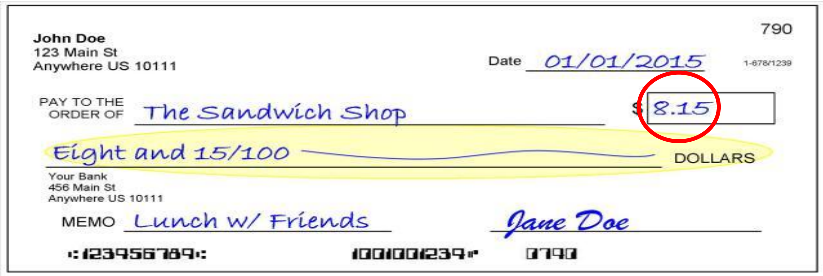
- 28x28 images of handwritten digits 0,1,...,9
- size-normalized and centered
- 60,000 used for **training**
- 10,000 used for **testing**

[1] subset of data collected by NIST, the US's National Institute of Standards and Technology

Potential Applications

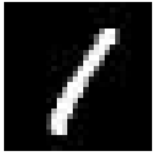
- **Banking**: Check deposits
- **Surveillance**: license plates
- **Shipping**: Envelopes/Packages





Initial Challenges and Solutions

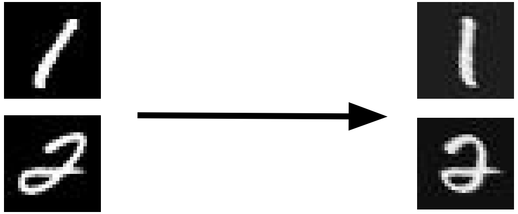
- **High dimensional data set**
  - Images stored as 784x1 vectors
  - Computationally expensive
- **Digits are written differently by different people**
  - Left-handed vs right-handed



- **Preprocess the data set**
  - Reduce dimension→ increase computation speed
  - Apply some transformation→ enhance features important for classification

Data Preprocessing Methods

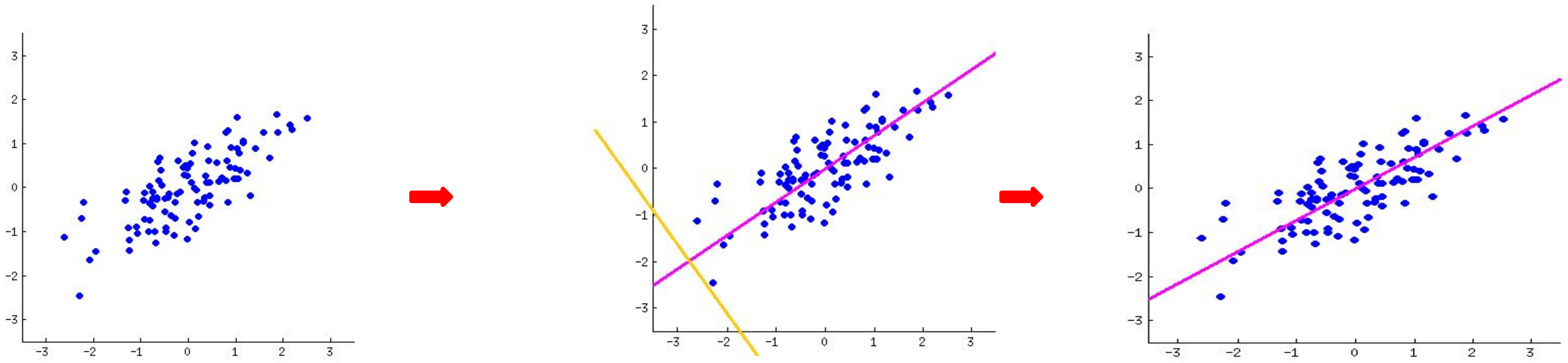
- In our experiments we have used the following methods
  - Deskewing
  - Principal Component Analysis (PCA)
  - Linear Discriminant Analysis (LDA)
  - 2D LDA
  - Nonparametric Discriminant Analysis (NDA)
  - Kernel PCA
  - t-Distributed Stochastic Neighbor Embedding (t-sne)
  - parametric t-sne
  - kernel t-sne



PCA & LDA

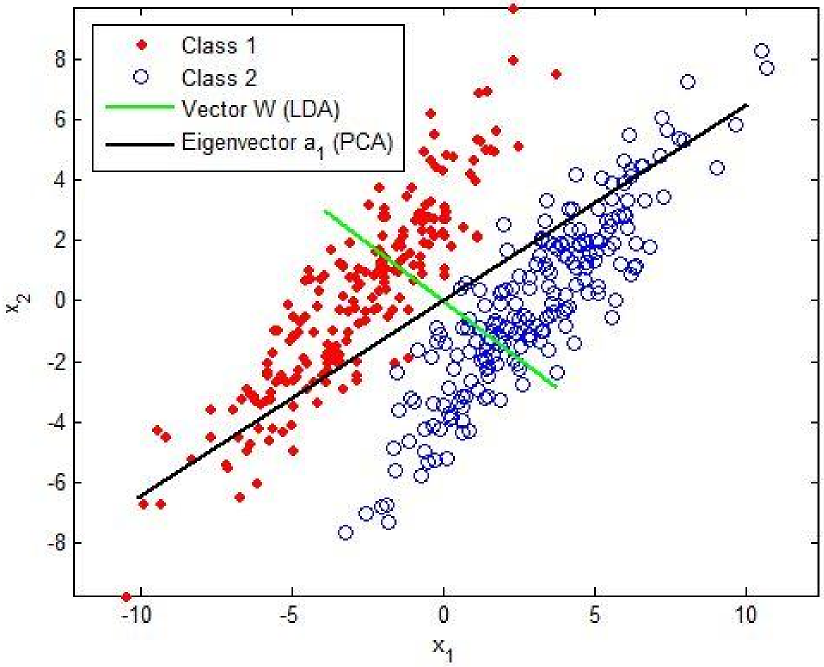
Principal Component Analysis (PCA)

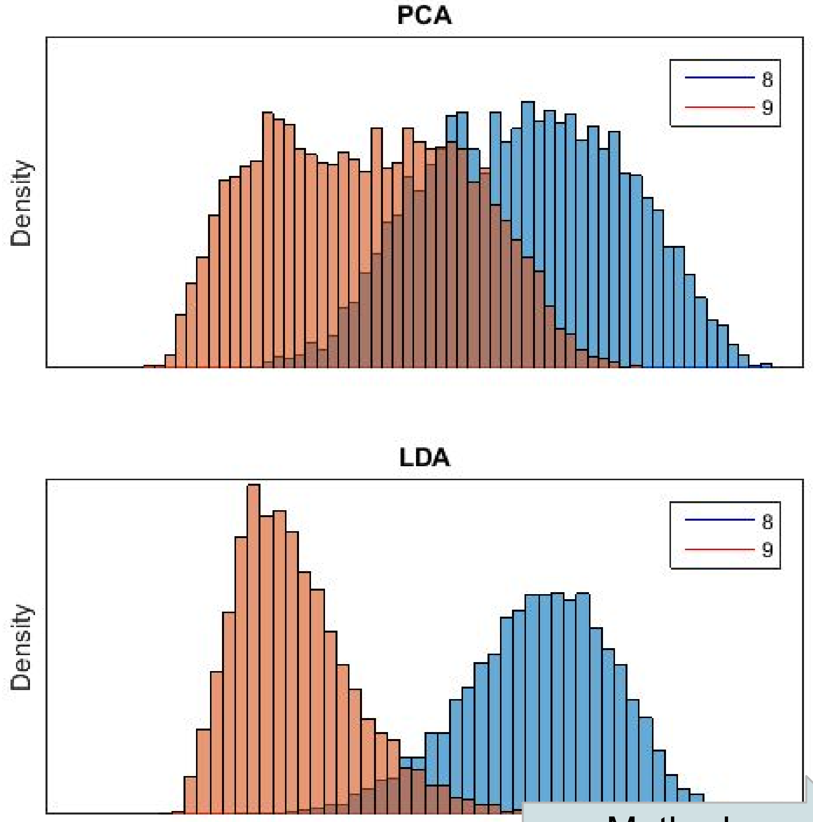
- Using too many dimensions (784) can be computationally expensive.
- Uses variance as dimensionality reduction criterion
- Throw away directions with lowest variance



Linear Discriminant Analysis:

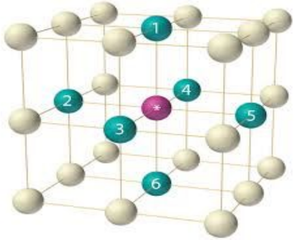
Reduce dimensionality, preserve as much class discriminatory information as possible.



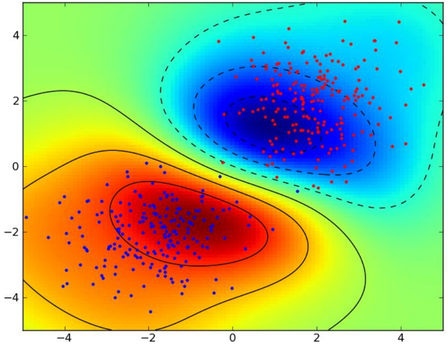


Methods

Classification Methods

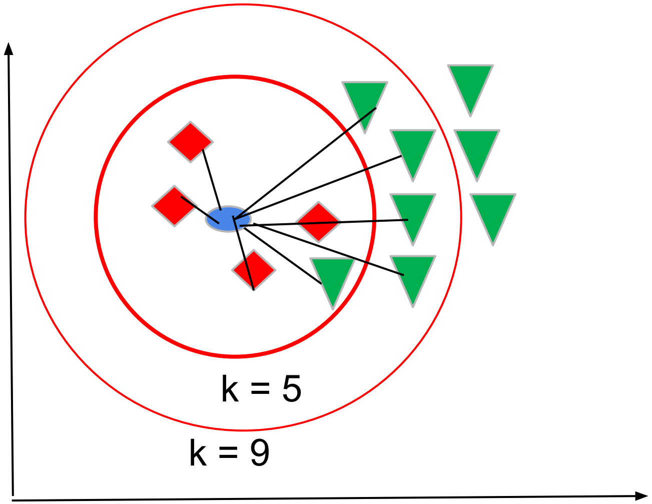


- In our experiments we have used the following methods
  - Nearest Neighbors Methods (Instance based)
  - Naive Bayes (NB)
  - Maximum a Posterior (MAP)
  - Logistic Regression
  - Support Vector Machines (Linear Classifier)
  - Neural Networks
  - Random Forests
  - Xgboost

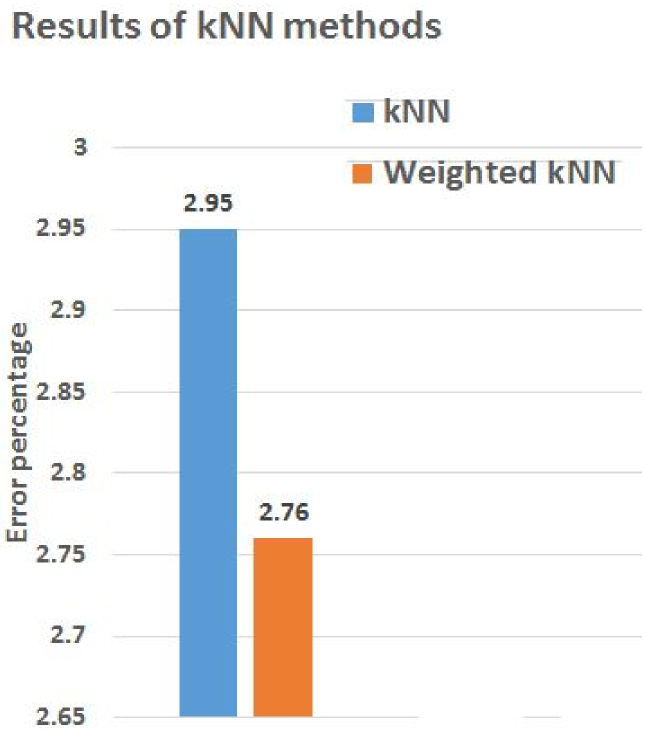


K nearest neighbors

- A new data point is assigned to the group of its k nearest neighbors



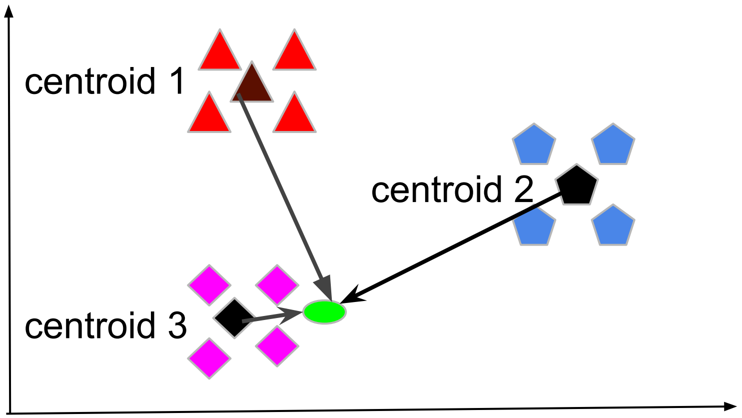
Majority of the neighbors are from class 2.
Test data is closer to class 1.



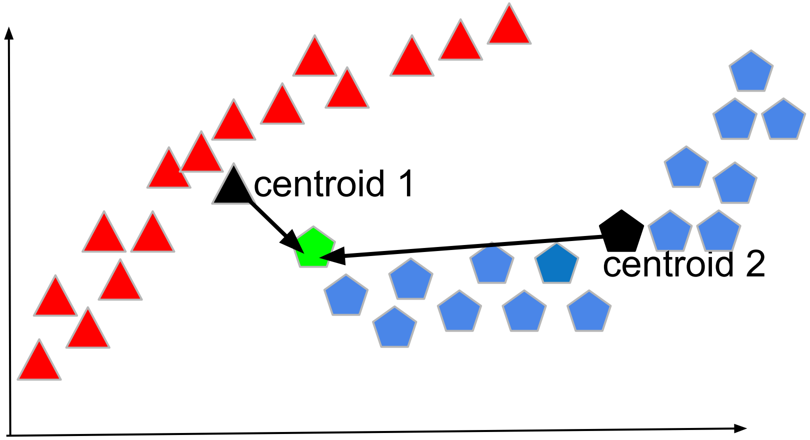
K means

- Situation 1: Data is well separated.
- Each class has a centroid/average.

- Situation 2: Data has non convex clustering.



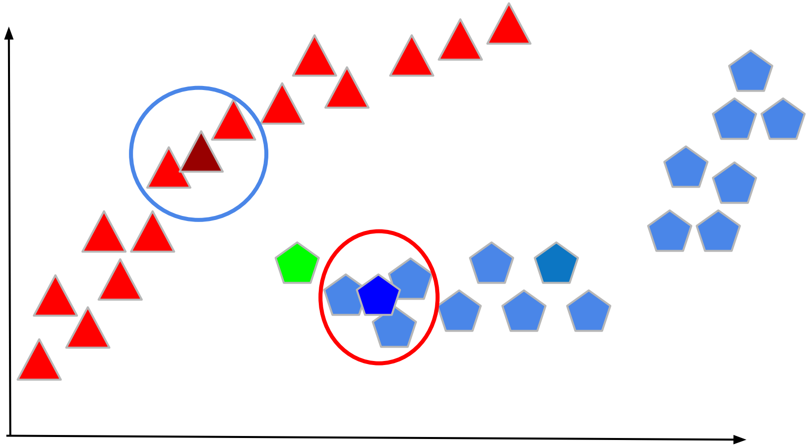
Test data is predicted to be from class 3.

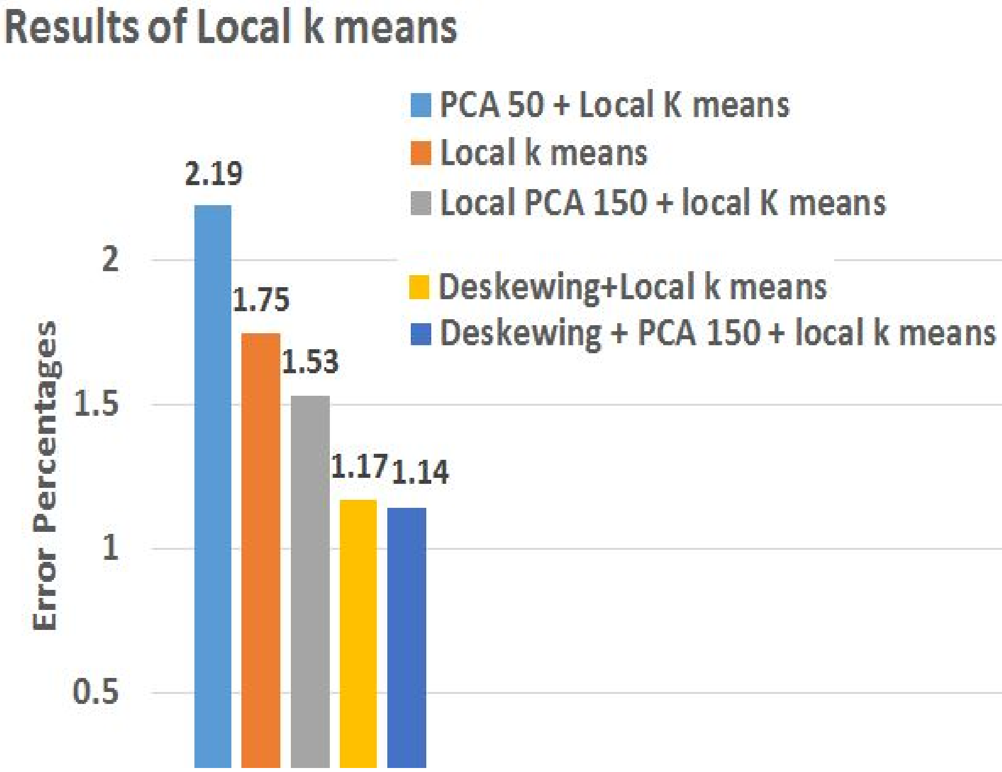


Test data belongs to class 2.
Misclassified to class 1.

Solution : Local k means

- For every class local centroids are calculated around the test data.

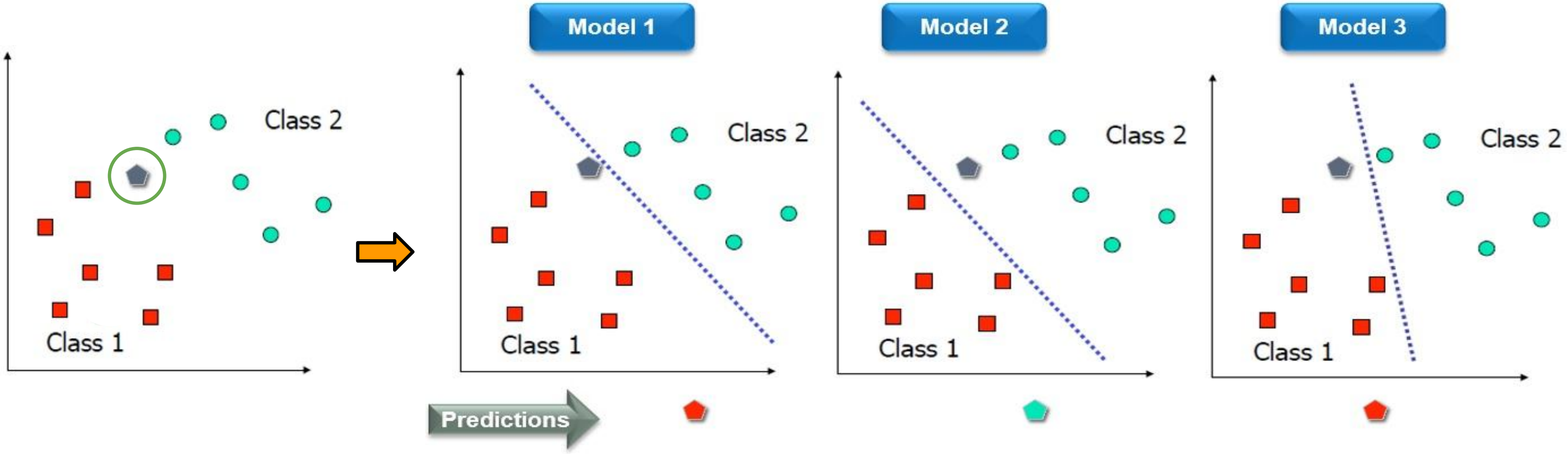




SVMs

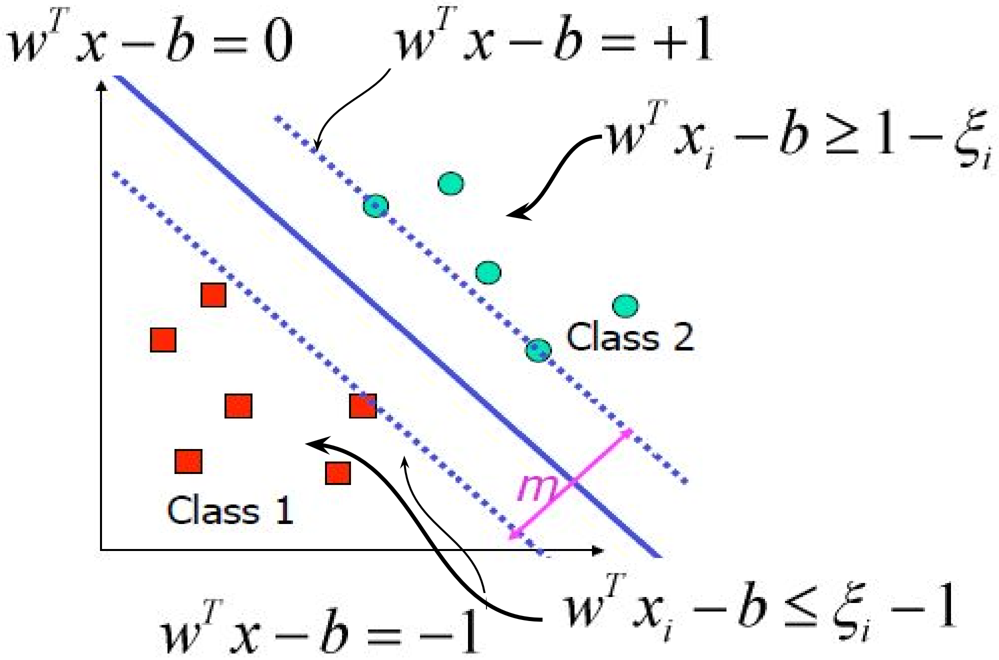
Support Vector Machines (SVM)

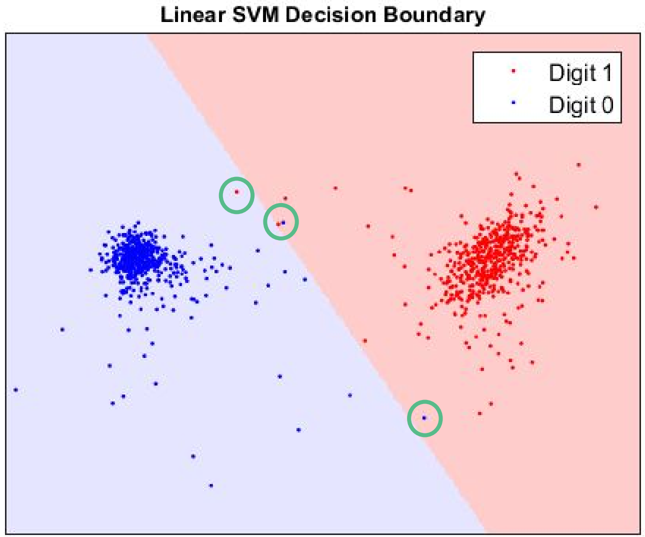
- classify new observations by constructing a **linear decision boundary**



Support Vector Machines (SVM)

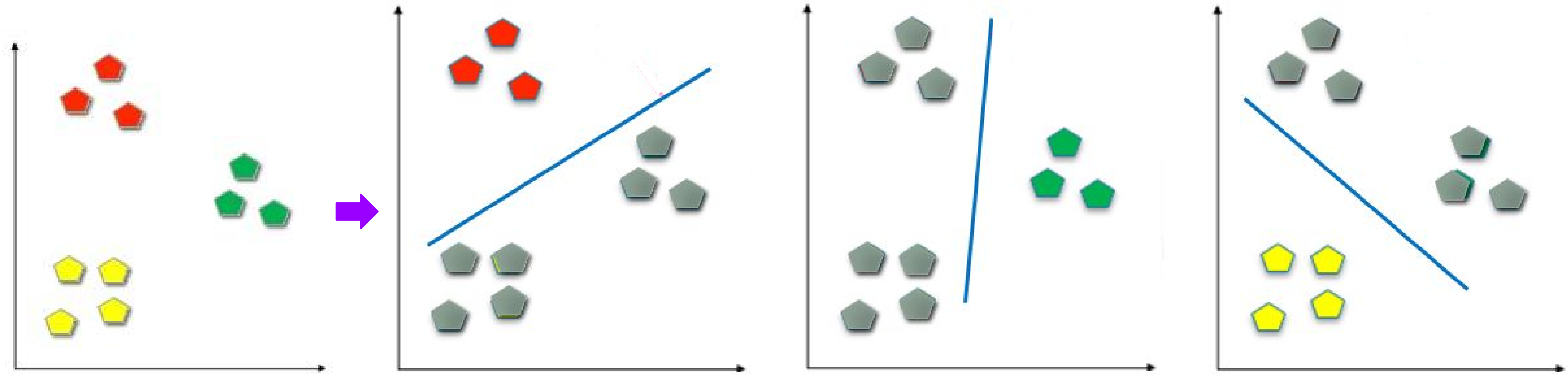
- Decision boundary chosen to **maximize** the separation m between classes





SVM with multiple classes

- SVM is a binary classifier. What if there are more than two classes?
- **Two methods**: 1) **One vs. Rest** 2) **Pairs**
- **One vs Rest**
  - Construct one SVM model for each class
  - Each SVM separates one class from the rest

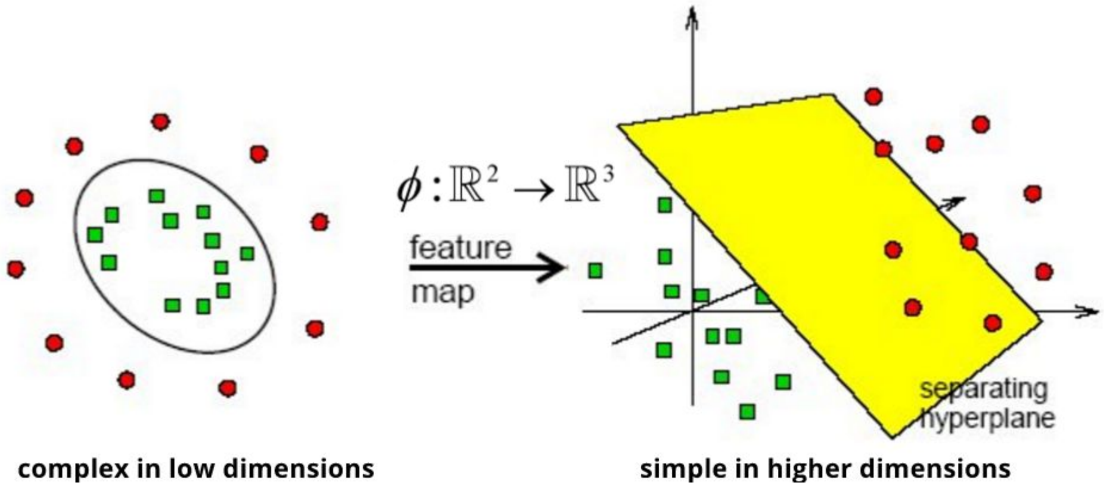


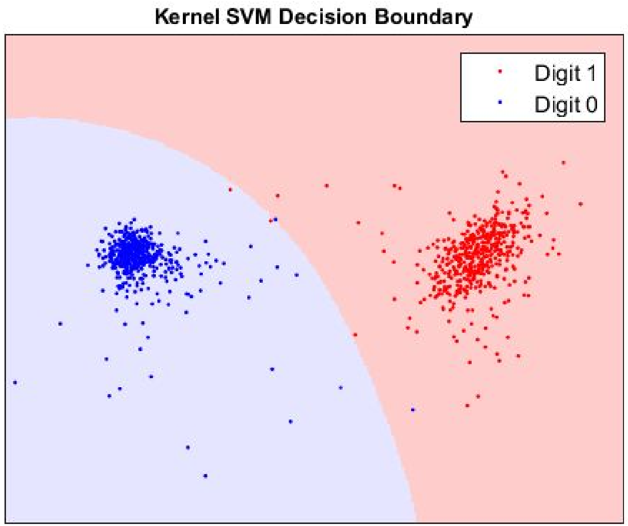
Support Vector Machines (SVM)

$$\kappa(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

$$\kappa(x_i, x_j) = \exp\left(-\frac{\| x_i - x_j \|^2}{2\sigma^2}\right)$$

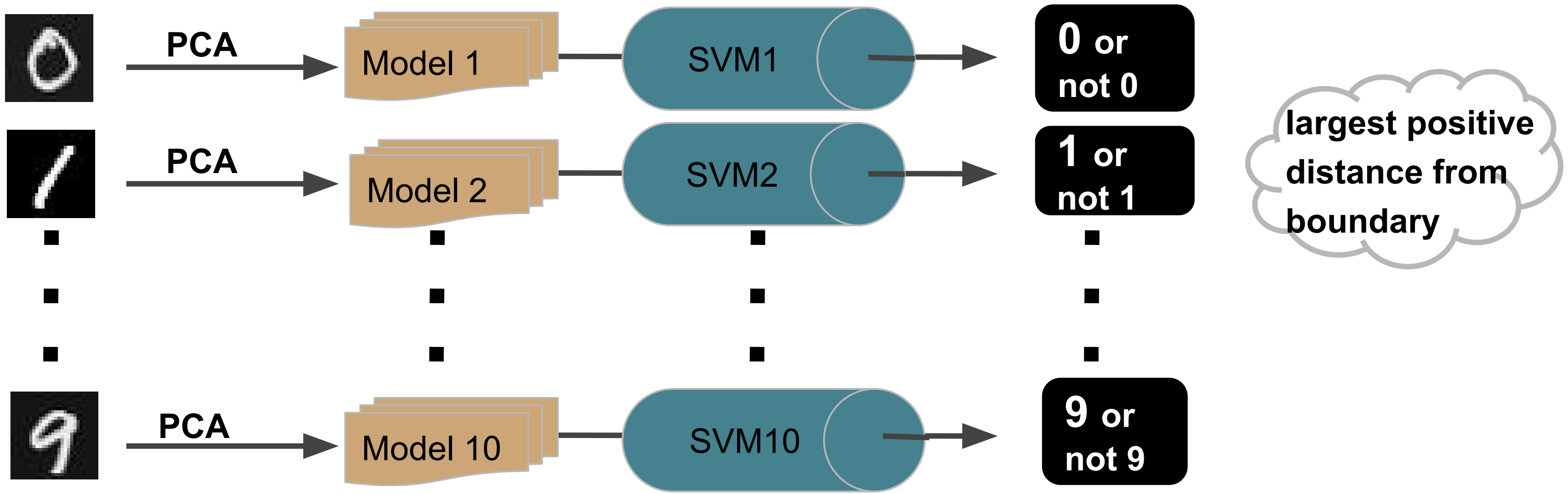
- What if data cannot be separated by a line?
- Kernel SVM: Separation may be easier in higher dimensions



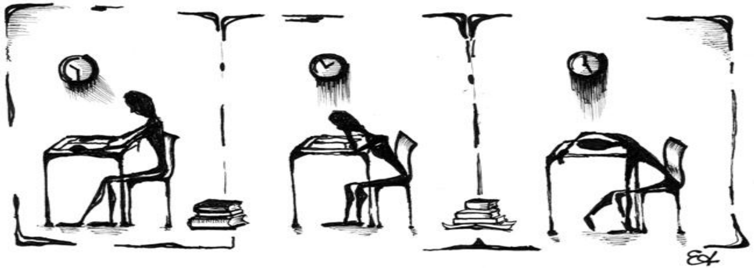


Combining PCA with SVM

- Traditionally: Apply PCA globally to entire data
- Our approach: Separately apply PCA to each digit space
- This extracts the patterns from each digit class
- We can use different parameters for each digit group.



Some Challenges for kernel SVM



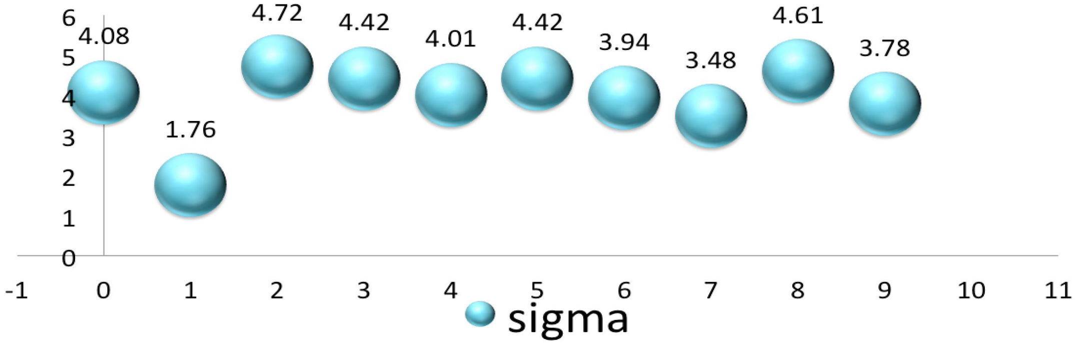
- It is not obvious what parameters to use when training multiple models
- Within each class, compute a corresponding sigma

$$\sigma_{C_i} = \frac{1}{n_{C_i}} \sum_{x \in C_i} \| x - kNN(C_i, x) \|$$

- This gives a starting point for parameter selection
- How to obtain an approximate range of parameters for training?

Parameter selection for SVMs

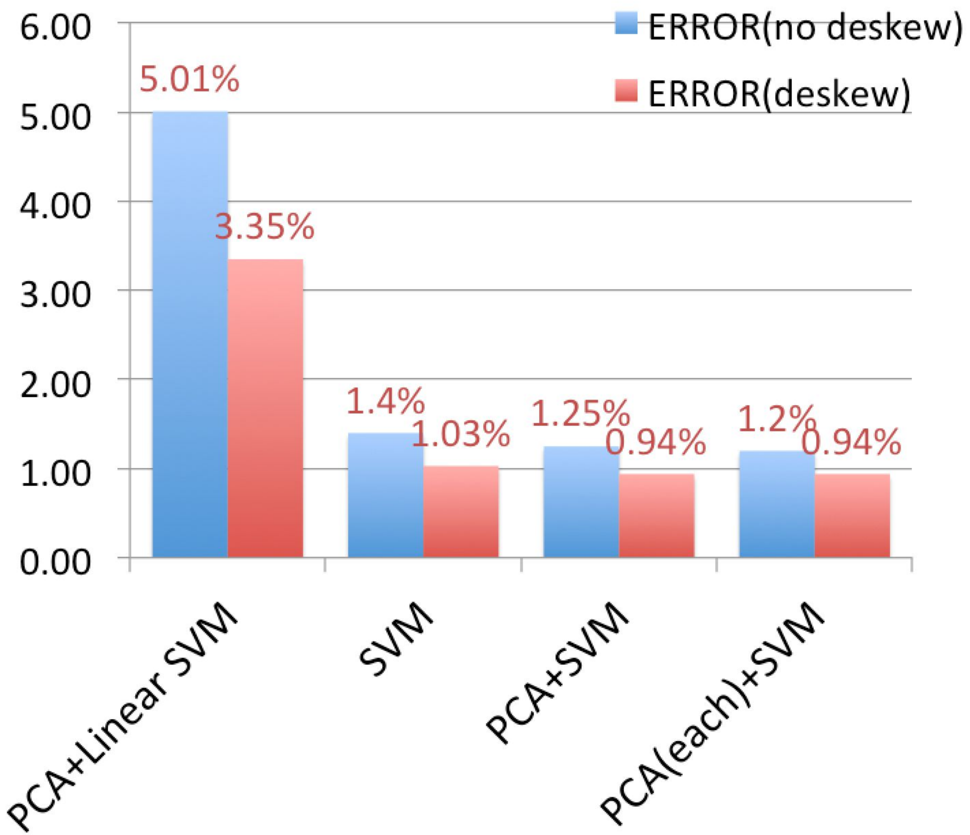
- Using kNN, with k=5, sigma values for each class
- Error 1.25% using kNN different sigma on each model

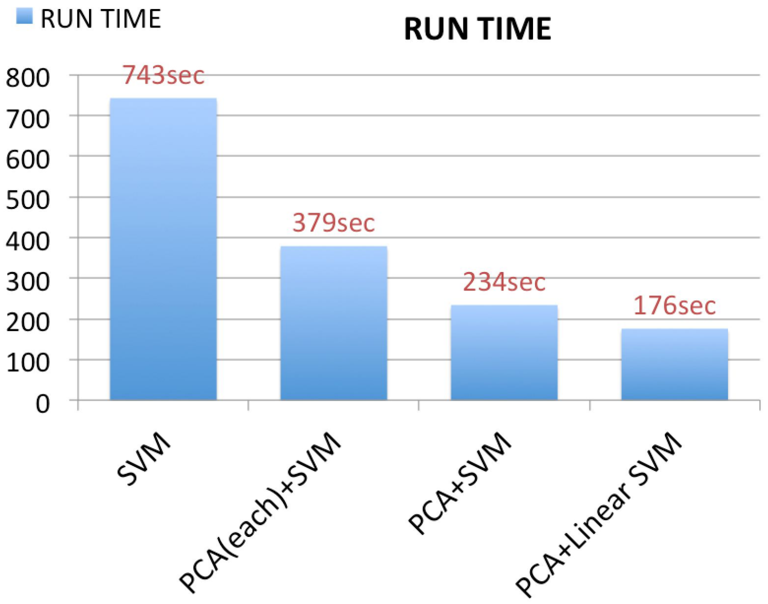


- Error 1.2% is achieved with the averaged sigma = 3.9235

$$\bar{\sigma} = \frac{1}{10} \sum \sigma_C = 3.9235$$

SVM + PCA results

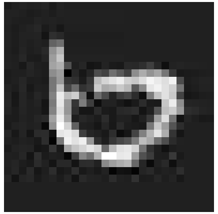




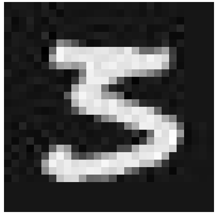
Some misclassified digits

(Based on local PCA + SVM, deskewed data)

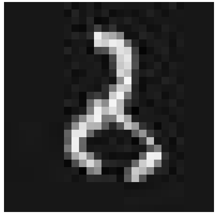
6 to 0



3 to 5

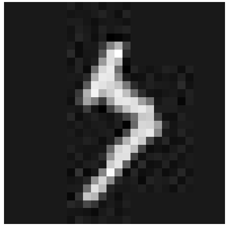


8 to 2

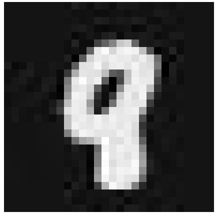


2 to 7

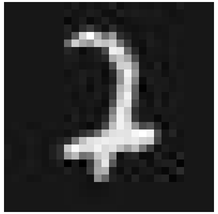
7 to 3



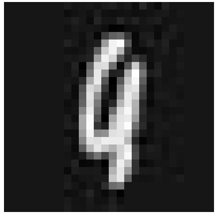
8 to 9



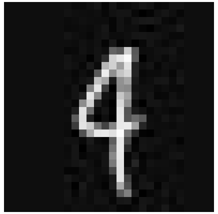
7 to 2



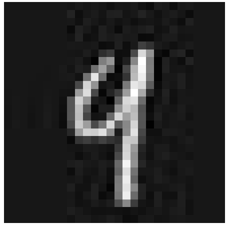
9 to 4



4 to 9

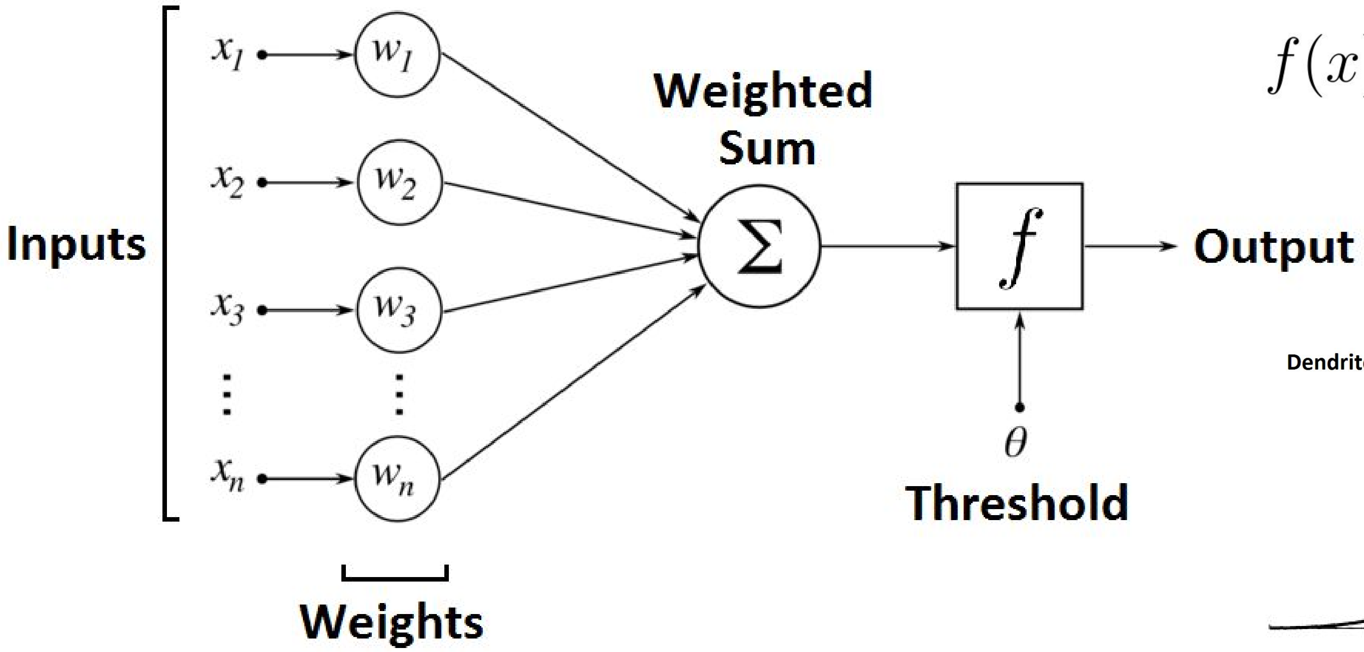


4 to 9

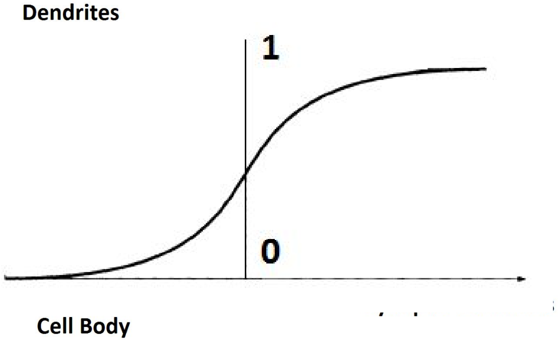


Neural Nets

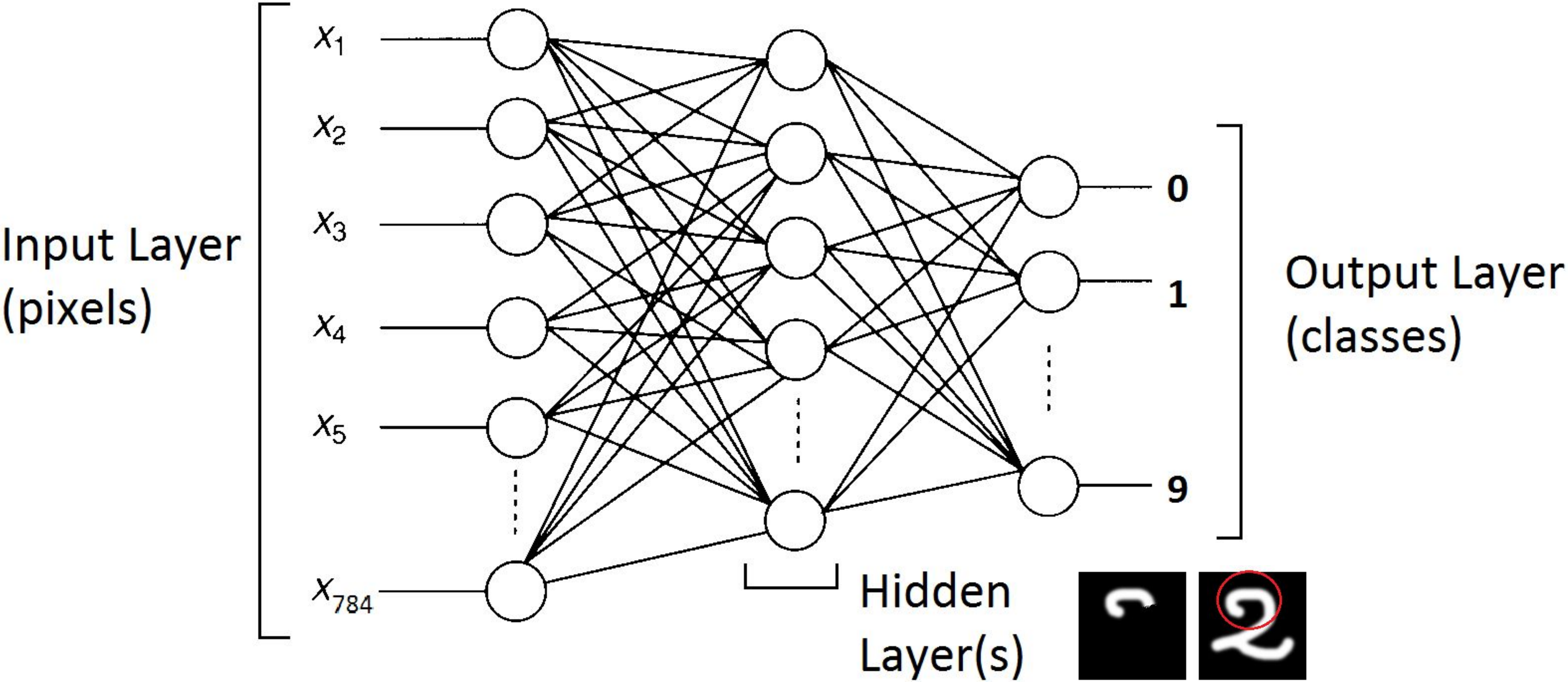
Neural Networks: Artificial Neuron



$$f(x) = (1 + e^{-\beta x})^{-1}$$

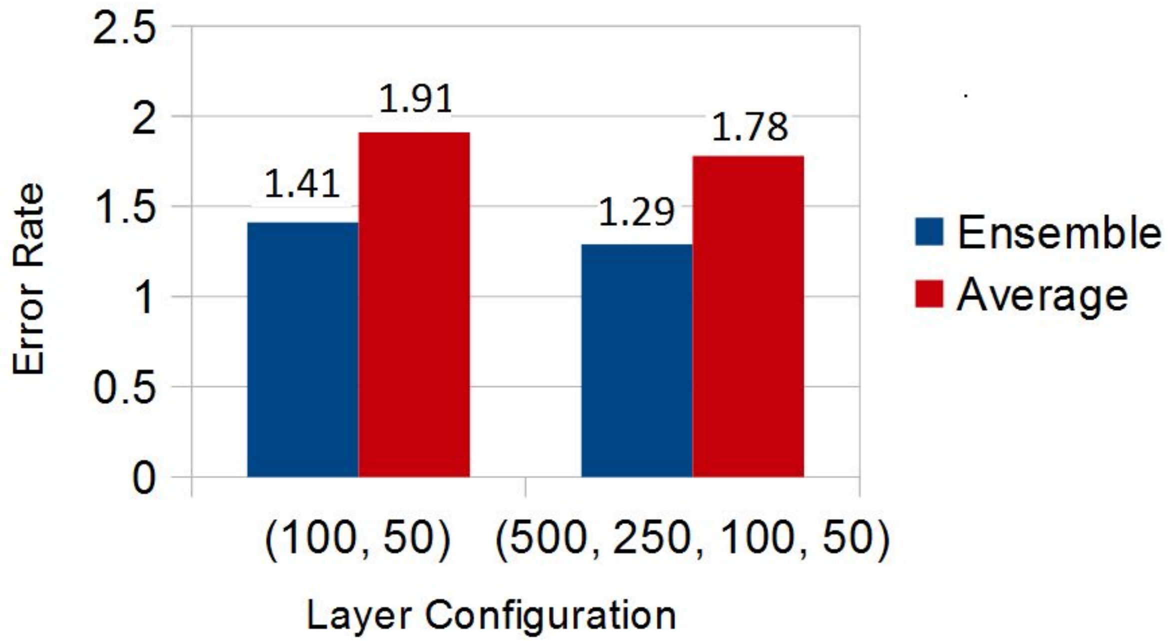


Neural Networks: Learning



Neural Networks: Results

Classification rule for ensembles: **majority voting**

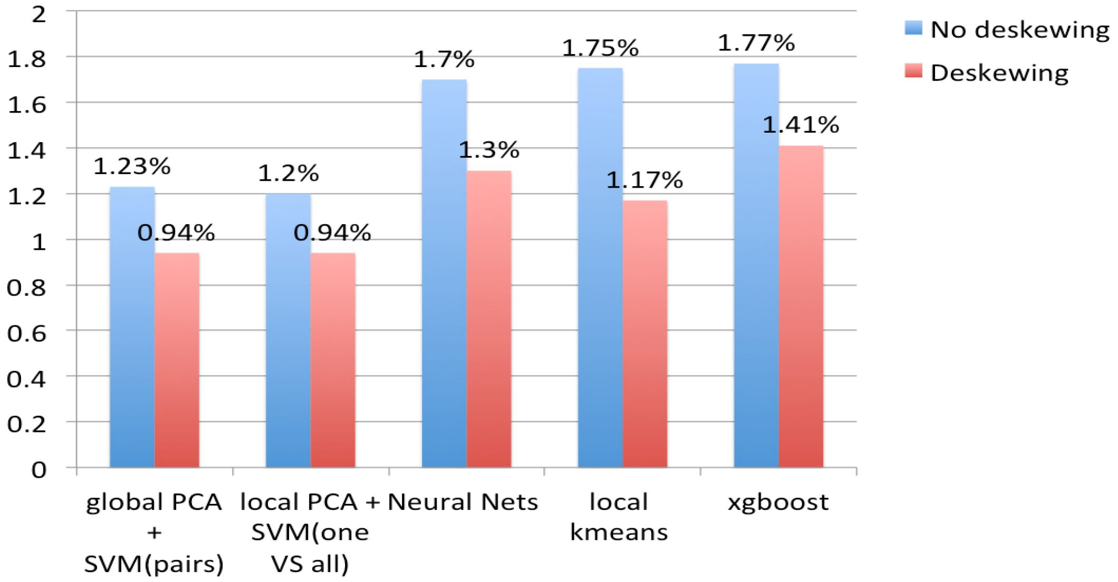


Conclusions

Summary and Results

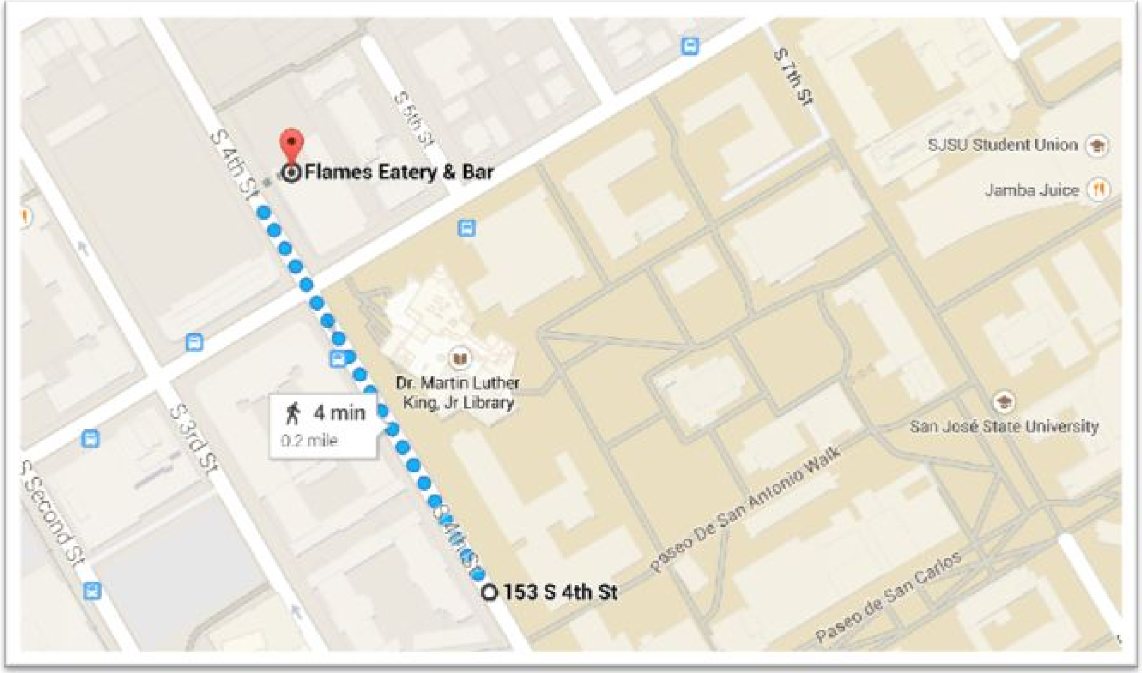
- Linear methods are not sufficient as the data is nonlinear.
- LDA did not work well for our data.
- Principal Component Analysis worked better than other dimensionality reduction methods.
- Best results were obtained with PCA values between 50 and 200 (55 being best)
- Deskewing improved results in general
- Best classifier for this data set is SVM

Results for MNIST



Questions?

Directions to Lunch



Flames Eatery, 88 South 4th Street (and San Fernando)

Choosing the optimum k

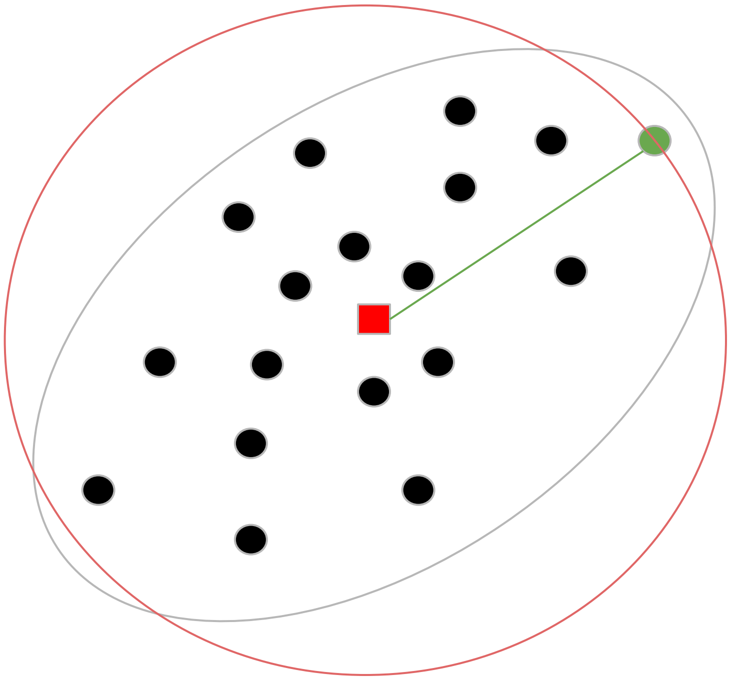
- The optimum k should be chosen with cross validation
- The data set is split into a training and a test set.
- The algorithm is run on the test set with different k values.
- The k that gives the least misclassification is chosen.

Local PCA

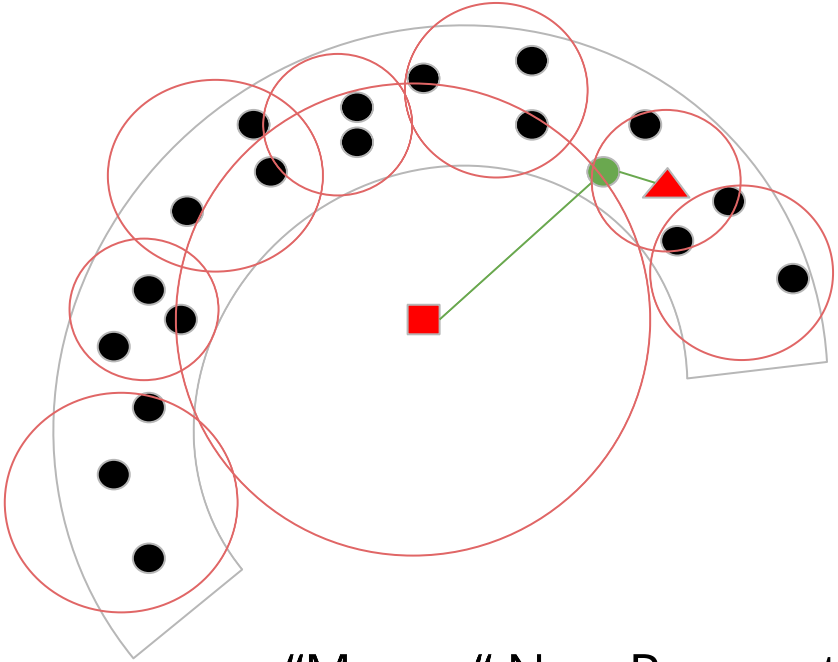
- For each of the class of digits the basis is found by PCA.
- Local PCA has ten bases instead of one global basis.
- Each of the test data point is projected into each of these ten bases.

Local Variance

■ Center
▲ Local k-Center



"Nice " Parametric Data



"Messy " Non-Parametric