Kernel Methods for Image Classification

Efforts By: Manish Kumar(201811085), Rishi Dave (201811073)
Supervisor: Prof. Pankaj Kumar

Dhirubhai Ambani
Institute of Information and Communication Technology.

30-November-2019

Outline

Outline

1. Problem Definition
2. Motivation
3. Kernel Methods
4. Experiments
   - Datasets Used
   - Kernels Used
   - Results
   - Confusion Matrix

Problem Definition

Classification

1. Technique for assigning objects to a particular class based on some properties.

Problem Definition

Classification

1. Technique for assigning objects to a particular class based on some properties.
2. Vast research topic in Machine Learning.

Problem Definition

Classification

1. Technique for assigning objects to a particular class based on some properties.
2. Vast research topic in Machine Learning.
3. Class labels are known (Supervised Learning).

Problem Definition

Classification

1. Technique for assigning objects to a particular class based on some properties.
2. Vast research topic in Machine Learning.
3. Class labels are known (Supervised Learning).
4. Examples:
    - Weather Prediction.
    - Patient has Cancer or not.
    - Tweet Sentiment Analysis.
    - Speaker Identification.
    - Biometric Identification.

Problem Definition

Image Classification

1. Subset of classification.

Problem Definition

Image Classification

1. Subset of classification.
2. Assigning images to a particular class.

Problem Definition

Image Classification

1. Subset of classification.
2. Assigning images to a particular class.
3. Examples:
    - Gender and Age Classification from a photo.
    - Patient has Cancer or not from MRI Images.
    - Detection of Facial Expression.
    - Biometric Identification.
    - Terrain Classification from a Satellite Image.
    - Hand Written Character Classification.

Outline

Motivation

1. Kernel representation offer an alternative solution by projecting the data into a high dimensional feature space to increase the computational power of the linear learning machines.[a]

[a] Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000. DOI: 10.1017/CBO9780511801389.

Motivation

1. Kernel representation offer an alternative solution by projecting the data into a high dimensional feature space to increase the computational power of the linear learning machines.[a]
2. Another attraction of the kernel method is that the learning algorithms and theory can largely be decoupled from the specifics of the application area, which must simply be encoded into the design of an appropriate kernel function.[a]

[a] Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000. DOI: 10.1017/CBO9780511801389.

Motivation

1. Kernel representation offer an alternative solution by projecting the data into a high dimensional feature space to increase the computational power of the linear learning machines.[a]
2. Another attraction of the kernel method is that the learning algorithms and theory can largely be decoupled from the specifics of the application area, which must simply be encoded into the design of an appropriate kernel function.[a]
3. Hence the problem of choosing an architecture for a neural network application is replaced by the problem of choosing a suitable kernel for Support Vector Machine (SVM).[a]

[a] Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000. DOI: 10.1017/CBO9780511801389.

Motivation

1. Kernel methods have been proven very effective on non vectorial data, in this way creating a connection with other branches of pattern analysis (graphs and strings).[a]

[a] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004. DOI: 10.1017/CBO9780511809682.

Motivation

1. Kernel methods have been proven very effective on non vectorial data, in this way creating a connection with other branches of pattern analysis (graphs and strings).[a]

2. Kernel based learning methods that finally enabled researchers to analyse nonlinear relations with the efficiency that had previously been reserved for linear algorithms.[a]

[a] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004. DOI: 10.1017/CBO9780511809682.

Motivation

1. Kernel methods have been proven very effective on non vectorial data, in this way creating a connection with other branches of pattern analysis (graphs and strings).[a]

2. Kernel based learning methods that finally enabled researchers to analyse nonlinear relations with the efficiency that had previously been reserved for linear algorithms.[a]

3. In short, kernel methods provides a new viewpoint whose full potential we are still far from understanding.[a]

---

[a] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004. DOI: 10.1017/CBO9780511809682.

Outline

Kernel Methods

1. Kernel Method Maps one feature vector to another feature vector.

Kernel Methods

1. Kernel Method Maps one feature vector to another feature vector.
2. $\Phi : X \rightarrow F$ is the mapping function (F is Feature Space and X is Input Space).

Kernel Methods

1. Kernel Method Maps one feature vector to another feature vector.
2. $\Phi : X \rightarrow F$ is the mapping function (F is Feature Space and X is Input Space).
3. SVM or other linear learning machines uses F for learning and predicting instead of X.

Kernel Methods

1. Kernel Method Maps one feature vector to another feature vector.
2. $\Phi : X \rightarrow F$ is the mapping function (F is Feature Space and X is Input Space).
3. SVM or other linear learning machines uses F for learning and predicting instead of X.
4. Example: Consider two dimensional(features a and b) data-points as shown in figure-1.

Why do we use kernels?

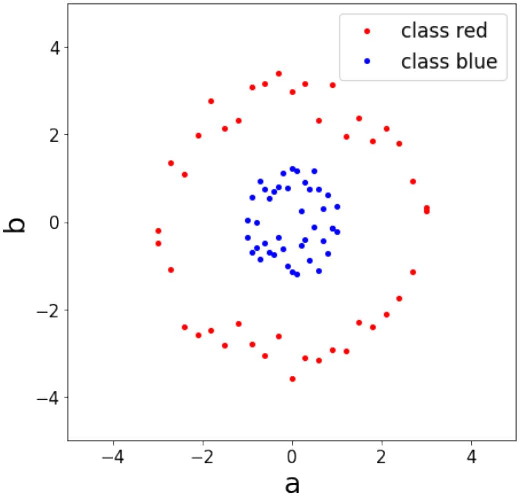

Figure 1: Input training examples(two features) containing two classes.

Why do we use kernels?

1. Linear separation between two classes is not possible.

Why do we use kernels?

1. Linear separation between two classes is not possible.
2. SVM or other linear learning machines will give bad accuracy of classification.

Why do we use kernels?

1. Linear separation between two classes is not possible.
2. SVM or other linear learning machines will give bad accuracy of classification.
3. So we use mapping function Φ and we get new 3 dimensional feature vector.

Why do we use kernels?

1. Linear separation between two classes is not possible.
2. SVM or other linear learning machines will give bad accuracy of classification.
3. So we use mapping function Φ and we get new 3 dimensional feature vector.
4. $\Phi((a, b)^T) = ((p, q, r)^T)$. Here $p = a, q = b, r = a^2 + b^2$

Why do we use kernels?

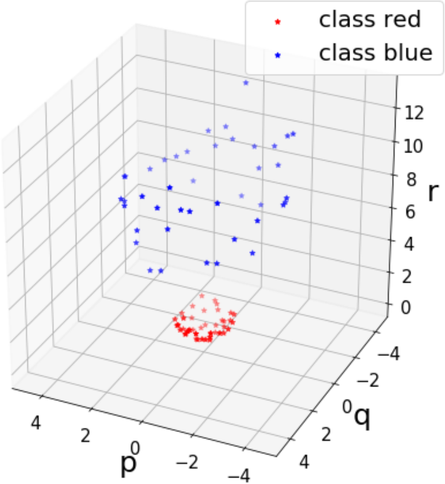

Figure 2: Training examples after Mapping with Φ.

Why do we use kernels?

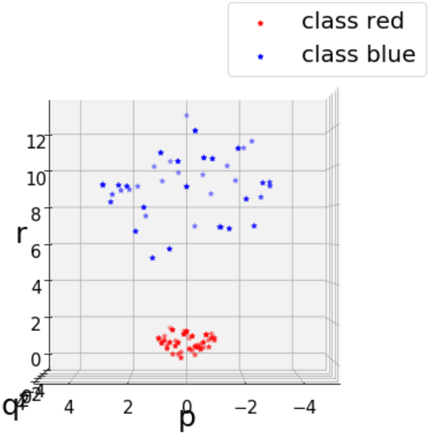

Figure 3: Front view of figure-2.

Outline

Experiments

Datasets Used:

MNIST[a]

- Handwritten English digits.
- Contains 10 classes (0 to 9 digits).
- 10000 images (nearly 1000 per class) are used for SVM training.
- 1000 images (nearly 100 per class) are used for testing.

[a] Yann LeCun and Corinna Cortes. "MNIST handwritten digit database". In: (2010). URL: http://yann.lecun.com/exdb/mnist/.

Experiments

Datasets Used:

Devanagari[a]

- Handwritten digits in Devanagari script.
- Whole dataset contains 46 classes and 2000 images per each class.
- We use only digits. So there are 10 classes.
- 4000 images (400 per class) are used for SVM training.
- 1000 images (100 per class) are used for testing.

[a] S. Acharya, A. K. Pant, and P. K. Gyawali. "Deep learning based large scale handwritten Devanagari character recognition". In: *2015 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*. Dec. 2015, pp. 1–6. DOI: 10.1109/SKIMA.2015.7400041.

Experiments

Datasets Used:

JAFFE[a]

- The Japanese Female Facial Expression.
- Contains 7 classes (facial expressions).
- There are total 213 images.
- 178 images (nearly 25 per class) are used for SVM training.
- 35 images(5 per class) are used for testing.

[a] Michael J. Lyons et al. "Coding facial expressions with Gabor wavelets". In: *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition* (1998), pp. 200–205.

Experiments

Datasets Used:

Corel[a]

- Whole dataset contains 10000 images and 100 classes.
- We have used only 600 images with 6 classes.
- 420 images(70 per class) are used for SVM training.
- 180 images(30 per class) are used for testing.

[a] Guang-Hai Liu, Jing-Yu Yang, and ZuoYong Li. "Content-based image retrieval using computational visual attention model". In: *Pattern Recognition* 48 (Feb. 2015). DOI: 10.1016/j.patcog.2015.02.005.

Experiments

Kernels Used

- Poly[a]

$$k(x,y) = < x.y >^d \; (x, y \in X, d \in \mathbb{N}) \tag{1}$$

- RBF (Radial basis function)[a]

$$k(x,y) = \exp\left[-\gamma(||x-y||^2)\right](x, y \in X, \gamma > 0) \tag{2}$$

- Cosine[a]

$$k(x,y) = \frac{< x.y >}{||x||||y||}(x, y \in X) \tag{3}$$

- Tanh[a]

$$k(x,y) = tanh(\alpha < x.y > + \beta)(x, y \in X, \alpha, \beta \in \mathbb{R}) \tag{4}$$

[a] B. Schölkopf and AJ. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Adaptive Computation and Machine Learning. Parts of this book, including an introduction to kernel methods. Cambridge, MA, USA: MIT Press, Dec. 2002, p. 644.

Experiments

Results

	Linear	Poly	RBF	Cosine	Tanh
MNIST	90.30	84.20	90.60	91.80	**92.60**
Devanagari Digits	89.80	**91.40**	68.60	65.90	10.00
JAFFE	**82.85**	**82.85**	54.28	14.28	31.42
Corel	**77.22**	74.44	32.77	20.55	19.44

Table 1: Accuracy of different types of kernels on different datasets

Confusion Matrix

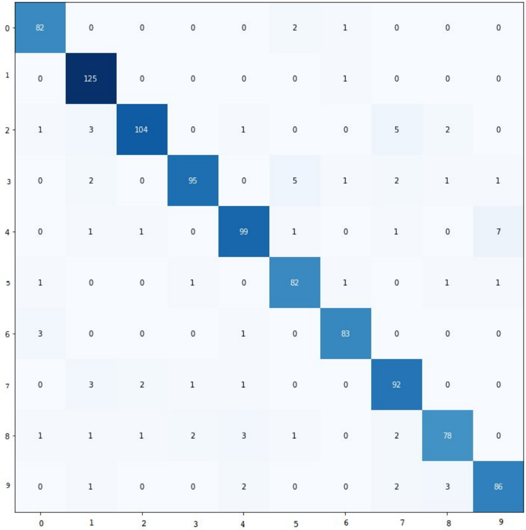

Figure 4: MNIST Dataset with tanh kernel

Confusion Matrix

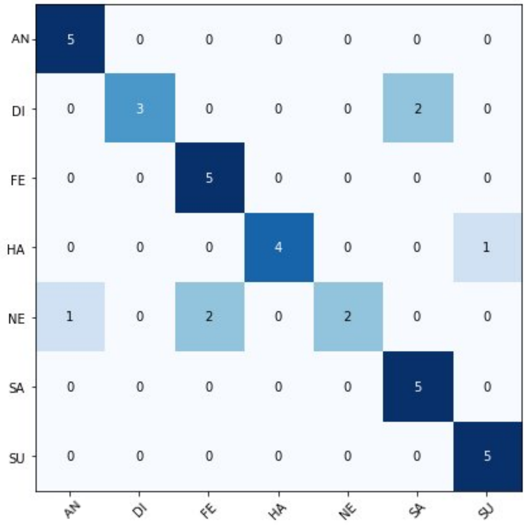

Figure 5: JAFFE Dataset with poly kernel

References I

S. Acharya, A. K. Pant, and P. K. Gyawali. "Deep learning based large scale handwritten Devanagari character recognition". In: *2015 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*. Dec. 2015, pp. 1–6. DOI: 10.1109/SKIMA.2015.7400041.

Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000. DOI: 10.1017/CBO9780511801389.

Tommi S. Jaakkola and David Haussler. "Exploiting Generative Models in Discriminative Classifiers". In: *Proceedings of the 1998 Conference on Advances in Neural Information Processing Systems II*. Cambridge, MA, USA: MIT Press, 1999, pp. 487–493. ISBN: 0-262-11245-0. URL: http://dl.acm.org/citation.cfm?id=340534.340715.

References II

Yann LeCun and Corinna Cortes. "MNIST handwritten digit database". In: (2010). URL: http://yann.lecun.com/exdb/mnist/.

Guang-Hai Liu, Jing-Yu Yang, and ZuoYong Li. "Content-based image retrieval using computational visual attention model". In: *Pattern Recognition* 48 (Feb. 2015). DOI: 10.1016/j.patcog.2015.02.005.

Michael J. Lyons et al. "Coding facial expressions with Gabor wavelets". In: *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition* (1998), pp. 200–205.

References III

Pedro J. Moreno, Purdy P. Ho, and Nuno Vasconcelos. "A Kullback-Leibler Divergence Based Kernel for SVM Classification in Multimedia Applications". In: *Advances in Neural Information Processing Systems 16*. Ed. by S. Thrun, L. K. Saul, and B. Schölkopf. MIT Press, 2004, pp. 1385–1392. URL: http://papers.nips.cc/paper/2351-a-kullback-leibler-divergence-based-kernel-for-svm-classification-in-multimedia-applications.pdf.

Andrew Y. Ng and Michael I. Jordan. "On Discriminative vs. Generative Classifiers: A comparison of logistic regression and naive Bayes". In: *Advances in Neural Information Processing Systems 14*. Ed. by T. G. Dietterich, S. Becker, and Z. Ghahramani. MIT Press, 2002, pp. 841–848. URL: http://papers.nips.cc/paper/2020-on-discriminative-vs-generative-classifiers-a-comparison-of-logistic-regression-and-naive-bayes.pdf.

References IV

B. Schölkopf and AJ. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Adaptive Computation and Machine Learning. Parts of this book, including an introduction to kernel methods. Cambridge, MA, USA: MIT Press, Dec. 2002, p. 644.

John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004. DOI: 10.1017/CBO9780511809682.