Kernel Methods in Machine Learning: Data challenge report

Gabriele Degola [1] **Marco Fioretti** [1]

Abstract

Image classification is a classic problem in machine learning. Nowadays, convolutional neural networks are mostly used to achieve good results for this task, but traditional machine learning approaches can still perform well. This report describes our experiments with kernel methods, focusing on appropriate data processing. The followed method allowed us to reach the 3rd place in the data challenge leaderboard, with accuracy score 0.616 and 0.602 on public and private data respectively.

1. Introduction

This report addresses the data challenge for the "Kernel Methods in Machine Learning" course, regarding a multi-class classification of image data. The objective is the implementation and understanding of machine learning algorithms for classification, exploiting kernel methods to work with structured data as images.

Next sections describe our approach to kernel methods for image classification and show that simple algorithms can obtain good results if data are processed in the right way. Section 2 concerns the followed data processing and feature extraction methods. Section 3 reviews kernel ridge regression (KRR) for multi-class classification and lists the tested kernels. Finally, Sections 4 and 5 present the contribution of employed methods in terms of performances and conclude the report.

2. Data processing

Provided data consist of 5000 training and 2000 test images from 10 classes (airplane, automobile, bird, cat, deer, dog, frog, horse, ship and truck), extracted from the famous CIFAR-10 dataset(2) and preprocessed. Each image is a $32 \times 32 \times 3$ vector, given as a sequence of 3072 numerical values.

2.1. Data augmentation

Data augmentation is a widely used technique in machine learning. A set of transformations are applied to training data, in order to artificially increase their amount by adding slightly modified copies of the available data. It also acts as regularization technique and helps avoid overfitting.

When dealing with images, commonly used transformations are flipping, rotation, shift and color-based operations. To some extent, machine learning models are "dumb": if the training dataset only contains images of dogs facing towards the right, a model will learn that dogs can only look that way and that, if an animal looks left, it cannot be a dog. However, humans can easily recognize dog in any pose. Horizontal flipping and other geometrical transformations tackle this issue, by also training the model on images of dogs in the opposite direction.

2.2. Feature extraction

Feature extraction is an important step when dealing with images. Indeed, if a model is simply fed with pixel values, it cannot extract meaningful relations between adjacent pixels in the three color channels. Nowadays, feature extraction is mostly performed by means of Convolutional Neural Networks (CNNs), capable of learning increasingly complex non-linear relationships between adjacent pixels in input images thanks of sequences of convolutional filters with non-linear activation functions.

Several feature extraction algorithms were otherwise proposed for traditional computer vision. Among them, histogram of oriented gradients (HOG)(1) is a local feature descriptor which consists in dividing each image in cells and compute a one-dimensional histogram of gradient directions for the pixels of each cell. Histograms are therefore built using magnitude and orientation of gradients. Every histogram corresponds to a feature for image classification.

[1]Team name: Degola Fioretti. Correspondence to: Julien Mairal <julien.mairal@m4x.org>.

MoSIG DSAI & MSIAM , Grenoble INP - Ensimag.

3. Algorithm

3.1. Classifier

Recalling the least-square regression as the search for the function with the lowest empirical risk, called mean squared error (MSE), one can say that the KRR is obtained by regularizing the MSE by the RKHS norm:

$$\hat{f} = \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i)^2) + \lambda \|f\|_{\mathcal{H}}^2$$

This has two main effects: it prevents overfitting by penalizing non-smooth functions, and it simplifies the solution.

In our method, we have implemented KRR following a one-vs-all approach for multi-class classification, so the learning algorithm takes a labeled training set as input where pairs of examples are supposed *i.i.d.* with respect to an unknown yet fixed probability distribution. A classifier is learned for each class against the other nine (respectively mapped as 1 and -1). The prediction function is found according to the empirical risk minimization (ERM) principle, as in binary classification, and each example is assigned to the class whose binary classifier returns the highest output value.

3.2. Kernels

Several kernel functions are used with KRR for classification:

- **linear kernel**: $K(x, x') = \langle x, x' \rangle$;
- **Gaussian kernel**: $K(x, x') = \exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right)$;
- **exponential kernel**: $K(x, x') = e^{\alpha\left(\langle x, x' \rangle - 1\right)}$;
- **Laplacian kernel**: $K(x, x') = \exp\left(-\frac{\|x-x'\|_1}{\sigma}\right)$.

4. Experiments and results

For augmenting our training dataset, horizontal flip, random rotation and random shift are tested with different proportions, always keeping original images in the dataset. Finally, our augmented training dataset consists of 15000 images and associated labels, divided as follows:

- 5000 original training images;
- 5000 original images, but all horizontally flipped;
- 5000 original images, but all horizontally flipped with probability 0.5 and then rotated of an angle which is sampled uniformly from the interval $[-30, 30]$ degrees.

Increasing the number of modified images and/or using random shift did not improve the classification results.

Table 1. Accuracy results on public and private leaderboards, with different kernel functions and data processing. `HF` is horizontal flip, while `Rot.` is random rotation in $[-30, 30]$ degrees.

Kernel	HOG	HF	Rot.	Public	Private
Linear				0.186	0.198
Gaussian				0.224	0.220
Linear	✓			0.451	0.464
Gaussian	✓			0.578	0.573
Gaussian	✓	✓		0.608	0.594
Gaussian	✓	✓	✓	**0.616**	**0.602**

HOGs are extracted from the augmented training dataset and used to train a KRR model. Grid-search cross validation on five folds is performed to select the best values for the regularization and for the parameters of each kernel.

Best accuracy results on the public and private leaderboards (each computed on approximately half of the test data) are both obtained with the Gaussian kernel, setting the regularization parameter $\lambda = 0.00001$ and $\gamma = \frac{1}{2\sigma^2} = 1$. Table 1 highlights the improvements achieved by stacking the different experimented methods. As expected, performances are poor when raw pixels are used as features and raise after HOG extraction. Gaussian kernel generally performs better than linear kernel (for public data, 0.578 against 0.451 on HOGs). Data augmentation slightly improves the accuracy and helps us to reach the 3rd place in both public and private leaderboards, showing our method is robust and does not overfit public data. Exponential and Laplacian kernels do not improve the results obtained with the Gaussian kernel.

5. Conclusion

In this data challenge, we experimented kernel methods for multi-class image classification. Our results show the importance of adequate data processing and feature extraction, that make a simple classifier as KRR obtain good performance on this complex task. Several kernel functions are tested, after tuning the kernel and classifier parameters, and their pros and cons are highlighted, as well as the improvements obtained with every applied method.

References

[1] N. DALAL AND B. TRIGGS, *Histograms of oriented gradients for human detection*, in 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol. 1, Ieee, 2005, pp. 886–893.

[2] A. KRIZHEVSKY, G. HINTON, ET AL., *Learning multiple layers of features from tiny images*, (2009).