Mini-Project 1.
Feature Extraction and Transfer Learning

CHEN Zhixian
Department of Mathematics
HKUST
Clear Water Bay
Student ID: 20573039
zchencz@connect.ust.hk

QIAN Yueqi
Dept of Mathematics
HKUST
Clear Water Bay
Student ID: 20669292
yqianai@connect.ust.hk

ZHANG Shunkang
Dept of Mathematics
HKUST
Clear Water Bay
Student ID: 20580484
szhangcj@connect.ust.hk

Abstract

In this project, we implement feature extraction by scattering net with known invariants and pre-trained deep neural network VGG19 based on MNIST dataset, with visualizing these features using classical unsupervised learning method T-distributed neighbor embedding for dimension reduction, and classification of traditional supervised learning method linear regression with stochastic gradient descent.

Remark of contribution

CHEN Zhixian: Feature extraction by scattering net with known invariants and feature visualization

QIAN Yueqi: Feature extraction by pre-trained deep neural network VGG19 and feature visualization

ZHANG Shunkang: Image classifications using traditional supervised learning methods based on the features extracted and analysis the results

1 Dataset

We have chosen MNIST dataset to work, using 5,000 training data and 10,000 testing data.

2 Feature extraction

2.1 Scattering net

We concentrate on wavelet Scattering transforms[1], which linearize translations and provide appropriate Gaussianization to extract the feature. The Scattering transform can provides an embedding for image classification without training by taking advantage of prior information on natural signals such as translation and deformation properties. We then just use a linear layer as a classifier based on the Scattering embedding on training data.

A wavelet transforms can decompose the signal at different scales. The linearization of translations at a scale 2^J is obtained by an averaging implemented with a convolution with a low-pass filter at this scale. A Scattering transform uses non-linearities to compute interactions between coefficients across multiple scales, which restores some information lost due to the averaging. A Scattering operator S_J transforms $x(u)$ into a tensor $x_J(u,k)$, where the spatial parameter u is sampled at intervals of size 2^J and the channels are indexed by k. The number K_J of channels increases with J to partially compensate for the loss of spatial resolution. These K_J channels are computed by a non-linear translation covariant transformation Ψ_J. Ψ_J is computed as successive convolutions with complex two-dimensional wavelets followed by a pointwise complex modulus, with no channel interactions. And we choose a Morlet wavelet ψ, scaled by 2^l for different values of l and rotated along $Q = 8$ angles $\theta = q\pi/Q$:

$$\psi_{\ell,q}(u) = 2^{-2\ell}\psi\left(2^{-\ell}r_\theta u\right) for 0 \leq q < Q.$$

To abtain an order two Scattering operator S_J here $J = 2$, the operator Ψ_J computes sequences of up to two wavelet convolutions and complex modulus:

$$\Psi_J(x) = \left[x, \left|x \star \psi_{\ell,q}\right|, \left\|x \star \psi_{\ell,q}\right| \star \psi_{\ell',q'}\right|\right]_{1 \leq \ell < \ell' \leq J, 1 \leq q,q' \leq Q}.$$

Therefore, there are $K_J = 1 + Q_J + Q^2J(J-1)/2 = 81$ channels. A Scattering transform is then obtained by averaging each channel with a Gaussian low-pass filter $\phi_J(u) = 2^{-2J}\phi(2^{-J}u)$ whose spatial width is proportional to 2^J:

$$S_J(x) = \Psi_J(x) \star \phi_J = \left[x \star \phi_J, \left|x \star \psi_{\ell,q}\right| \star \phi_J, \left\|x \star \psi_{\ell,q}\right| \star \psi_{\ell',q'}\right| \star \phi_J\right]_{1 \leq \ell < \ell' \leq J, 1 \leq q,q' \leq Q}.$$

Convolutions with ϕ_J are followed by a subsampling of 2^J; as a result, if x has p pixels then S_J is a dimension $p\alpha_J$ where:

$$\alpha_J = 2^{-2J}(1 + Q_J + Q^2J(J-1)/2).$$

In our experiment, the image size of MNIST is $p = 28^2$, thus the dimension of S_J is $p\alpha_J = 7^2$. The wavelets separate the variations of x at different scales 2_l along different directions $q\pi/Q$, and second order Scattering coefficients compute interactions across scales. And finally, the feature size is $[81, 7, 7]$.

2.2 Pre-trained VGG19

The network architecture of VGG19 is shown as the following figure 1[3].

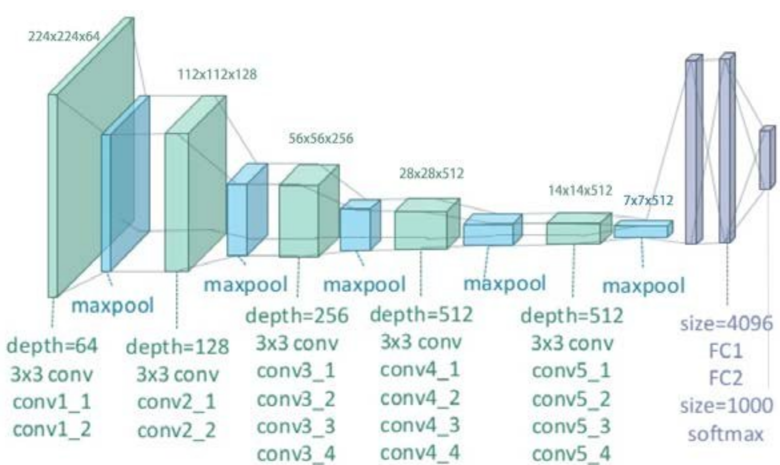

Figure 1: Architecture of VGG19

We consider the output of layer "FC2" which is the last layer before softmax as our features, with size of 4,096.

3 Feature visualization

T-distributed Stochastic Neighbor Embedding is a tool to visualize high-dimensional data, which converts similarities between data points to joint probabilities and tries to minimize the Kullback-Leibler divergence between the joint probabilities of the low-dimensional embedding and the high-dimensional data[2].

We use the library "sklearn.manifold" to implement TSNE, with PCA initialization of embedding, 501 as the random state. After embedding the features we have extracted into two dimension space, it is much eaiser to make visualization that can be directly perceived by people.

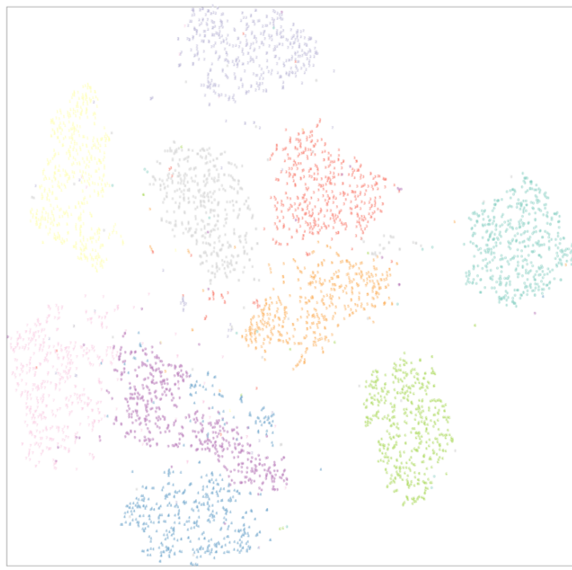

Figure 2: Features extracted by scattering net

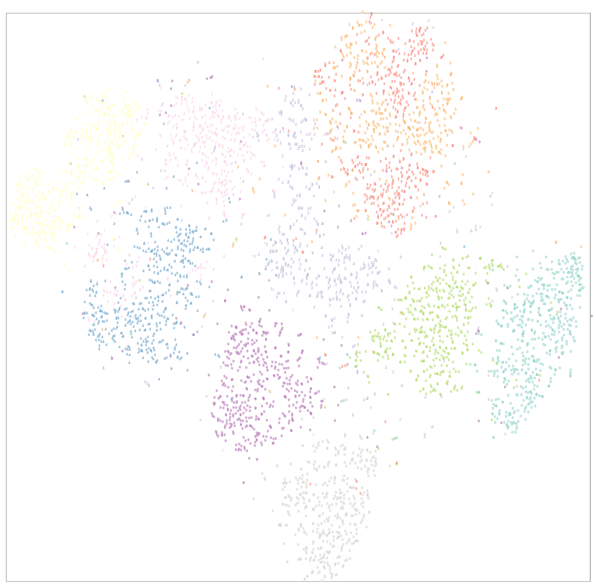

Figure 3: Features extracted by Pre-trained VGG19

4 Image classifications

Logistic regression is used to make classification in our case. More specifically, we choose Stochastic Gradient Descent to optimize.

Table 1: Classification Accuracy

Feature Method	Accuracy (%)
Scattering net	98.18
VGG19	95.67

5 Analysis and Explanation

Based on the feature extracted by using scattering net and pre-trained VGG19 network, we use the logistic regression to further classify the MNIST data. According to Table 1, we see that the logistic regression model can reach 98.18% accuracy, which is higher than VGG19 95.67%. From the visualization of 2 dimensional manifold, we see that it is easier to use linear decision boundary to separate the different classes. In the other hand, we see that features extracted by pretrained-VGG 19 are more complicated. For example, the class of digit 5 and digit 3 is mixed and it is impossible to use a linear classifier to classify these two classes perfectly.

Moreover, we train another nonlinear classifier based on the simple convolution neural network and the accuracy is more than 99%. Based on these results, we conclude that it is hard to get a great performance by directly using classifier trained on another dataset and the transferability of classifier is limited.

references

[1] Mallat, Stéphane. "Group invariant scattering." Communications on Pure and Applied Mathematics 65.10 (2012): 1331-1398.

[2] Angles, Tomás, and Stéphane Mallat. "Generative networks as inverse problems with scattering transforms." arXiv preprint arXiv:1805.06621 (2018).

[3] Zheng, Yufeng & Yang, Clifford & Merkulov, Aleksey. (2018). Breast cancer screening using convolutional neural network and follow-up digital mammography. 4. 10.1117/12.2304564.

[4] Maaten, Laurens van der & Geoffrey Hinton. "Visualizing data using t-SNE." *Journal of machine learning research* 9.Nov (2008): 2579-2605.