Principles of Data Science Project 1
Dimension Reduction

Hongzhou Liu
517030910214
deanlhz@sjtu.edu.cn

Xuanrui Hong
517030910227
aaa@bbb.ccc

Qilin Chen
517030910155
1017856853@sjtu.edu.cn

To Xuanrui Hong:
Method
24
git

- ppt
- , i.e. paraphrase
- sci-kit learnPCAUser Guide2.Unsupervised Learning-2.5.Decomposing signals in components (matrix factorization problems)
-

To Qilin Chen:
ExperiementFeature SelectionConclusionXuanrui HongFeature Selectionpushgit pull 24
git

- Prj1/ResultSVMSVMRBF2
- Baselinebaseline
- figuresubfigure
- SVMSVM
- AutoEncoderPrj1/Projection/AutoEncoder.py
- sci-kit learnPCAUser Guide2.Unsupervised Learning-2.5.Decomposing signals in components (matrix factorization problems)
-
- author
- git pullgit push

Abstract

TODO: Hongzhou Liu

1 Method

1.1 Feature Selection

1.1.1 Select-k-best

SelectKBest is one of the methods of univariate feature selection, which works by selecting the best features based on univariate statistical tests. It removes all but the k highest scoring features.

1.1.2 Variance Threshold

VarianceThreshold is a simple approach to feature selection. It removes all features whose variance doesn't meet some threshold. The principle is that features with small variance often contain less data information.

1.1.3 Tree-based Selection

Tree-based feature selection combines SelectFromModel and ExtraTreesClassifier. SelectFromModel is a meta-transformer that can be used along with any estimator that has a $coef\_$ or $feature\_importances\_$ attribute after fitting. The features whose $coef\_$ or $feature\_importances\_$ values are below the provided threshold parameterare are considered unimportant and removed. ExtraTreesClassifier can be used to compute feature importances, which happens to cooperate with SelectFromModel to discard irrelevant features.

1.2 Feature Projection

1.2.1 PCA

Principal component analysis is one of the most widely used data dimensionality reduction algorithms. It performs a linear mapping of the data to a lower-dimensional space in such a way that the variance of the data in the low-dimensional representation is maximized.[1] Formally, the optimization goal is

$$\max_{v} \frac{1}{n} \sum_{i=1}^{n} (v^T x_i)^2 = \frac{1}{n} v^T X X^T v \tag{1}$$

where v is the new axis.

$$s.t. \quad v^T v = 1 \tag{2}$$

Using lagrange Multiplier we can get

$$X X^T v = \lambda v \tag{3}$$

We can see that v is the eigenvector of XX^T, and λ is the corresponding eigenvalue. Therefore, v can be calculated by performing eigenvalue decomposition to the co-variance matrix XX^T. Then we can get the data after dimensionality reduction.

1.2.2 Kernel PCA

In general, principal components analysis is suitable for linear dimensionality reduction of data. Kernel PCA can achieve nonlinear dimensionality reduction of data and is used to process linear inseparable data sets.

The general idea of KPCA is: for the matrix in the input space, we first use a non-linear mapping to map all samples in a high-dimensional or even infinite-dimensional space, and then perform PCA dimensionality reduction in this high-dimensional space.

1.2.3 LDA

TODO: Xuanrui Hong

1.3 Feature Learning

1.3.1 t-SNE

TODO: Xuanrui Hong

1.3.2 LLE

TODO: Xuanrui Hong

1.3.3 AutoEncoder

TODO: Xuanrui Hong

2 Experiment

2.1 Baseline

TODO: Hongzhou Liu
Training Set : Test Set = 6:4
Linear SVM: Best C = 0.002, best accuracy = 0.932815 (baseline)
Kernel SVM with RBF kernel: Best C = 5.0, best accuracy = 0.935160 (baseline)

2.2 Feature Selection

2.2.1 Select-k-best

TODO: Xuanrui Hong

2.2.2 Variance Threshold

TODO: Xuanrui Hong

2.2.3 Tree-based Selection

TODO: Xuanrui Hong

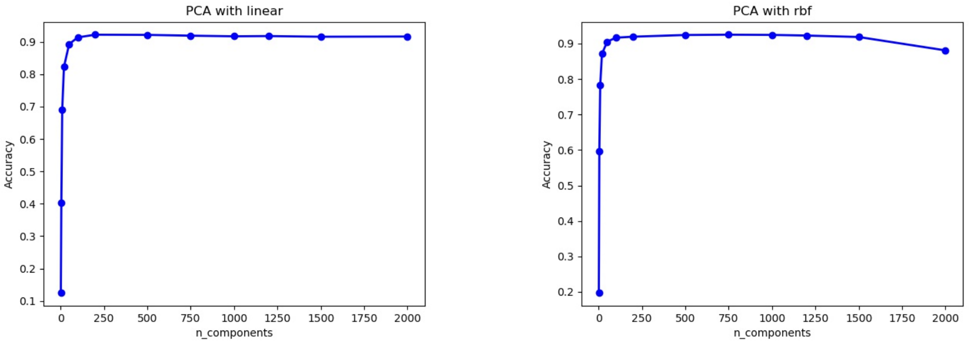
2.3 Feature Projection

2.3.1 PCA

Principal component analysis(PCA) is the most typical feature projection method based on dimension reduction, since PCA provides a roadmap for how to reduce a complex data set to a lower dimension to reveal the sometimes hidden, simplified dynamics that often underlie it. Kernel principal component analysis(kernel PCA) provides an extension of traditional PCA using techniques of kernel methods, which project source data into a higher dimensional space, providing with better reduction and classification performance. In this section, two experiment settings can derive from the rule: We use kernel PCA on different number of aim component $[2, 5, 10, 20, 50, 100, 200, 500, 750, 1000, 1200, 1500, 2000]$, and we give our results on two types of kernel: linear kernel and radial basis function kernel. We adopt classifiction accuracy as metric in this section.

We summarize the experimental results of kernel PCA in Tab. 1 and Fig. 1. As is shown, linear kernel PCA model and RBF kernel PCA model expectively exhibit the best performances at 200 components and 750 components, reaching 92.20% and 92.50% on the classification task. We can compare the result with the simple linear SVM and simple RBF SVM, find the kernel PCA model's performance is worse than the baselines, and we deem the reason is that compoents reduction of PCA remove some effective feature as it remove the most noisy feature. And we can find RBF kernel have better performance than linear kernel in PCA tasks and none-PCA tasks, it proves the provided dataset have some features which can't be handled merely by linear kernel. Besides, in kernel PCA tasks, we can find if we have less components, RBF kernel have better performance than linear kernel.

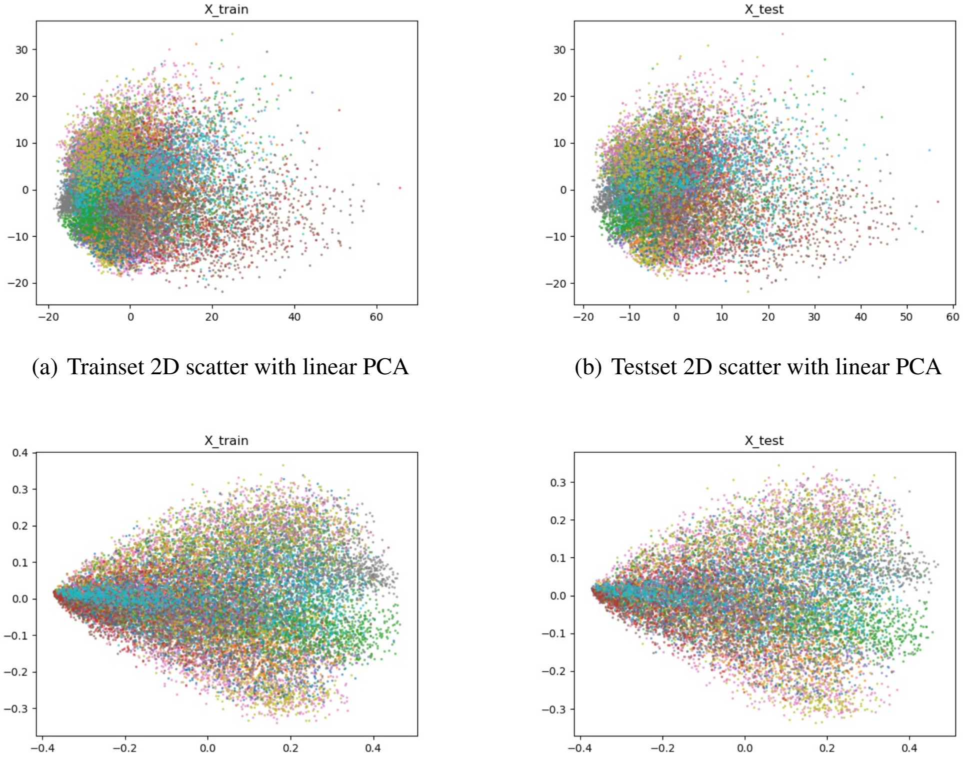
We give the 2-components scattering figure of different kernel and dataset in Fig. 2, which can support the result that RBF kernel have better performance than linear kernel since the RBF figure can be better splited. What's more, if we compare the trainset and testset, we can find the dataset is randomly splited.



(a) metric accuracy comparison on linear PCA

(b) metric accuracy comparison on RBF PCA

Figure 1: Kernel PCA performance on linear kernel and RBF kernel



(a) Trainset 2D scatter with linear PCA

(b) Testset 2D scatter with linear PCA

(c) Trainset 2D scatter with RBF PCA

(d) Testset 2D scatter with RBF PCA

Figure 2: Dataset 2D scatter with linear kernel and RBF kernel

Table 1: Comparison of kernel PCA and baslines in Classification Task

Model	PCA+SVM		SVM	
components number	Linear kernel(%)	RBF kernel(%)	Linear kernel(%)	RBF kernel(%)
2	12.46	19.68	93.28	93.52
5	40.31	59.68	93.28	93.52
10	68.99	78.25	93.28	93.52
20	82.27	87.07	93.28	93.52
50	89.31	90.53	93.28	93.52
100	91.37	91.66	93.28	93.52
200	92.20	91.94	93.28	93.52
500	92.14	92.42	93.28	93.52
750	91.90	92.50	93.28	93.52
1000	91.70	92.44	93.28	93.52
1200	91.78	92.26	93.28	93.52
1500	91.57	91.85	93.28	93.52
2000	91.64	88.08	93.28	93.52

2.3.2 LDA

TODO: Qilin Chen

2.4 Feature Learning

2.4.1 t-SNE

TODO: Qilin Chen

2.4.2 LLE

TODO: Qilin Chen

2.4.3 AutoEncoder

TODO: Qilin Chen

3 Conclusion

TODO: Qilin Chen

Acknowledgement

References

[1] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.