Dimensionality Reduction for Bag-of-words Models

Benjamin Fayyazuddin Ljungberg
benfl@stanford.edu

Introduction

The original task was to train a model to classify texts by author. This turned out to be very easy, so instead I used this (easily classifiable) set of texts to test two forms of feature reduction: latent semantic analysis (LSA) and principal component analysis (PCA).

LSA projects data onto a smaller linear subspace while PCA projects onto a smaller affine subspace. Intuitively, we expect PCA to work better.

Experiment

The 90 most significant principal components and 200 most significant LSA features (for the train and test sets taken together) were found. A support vector machine (SVM) was trained, using a one-vs-rest method, on the lower dimensional data. The accuracy after training was recorded.

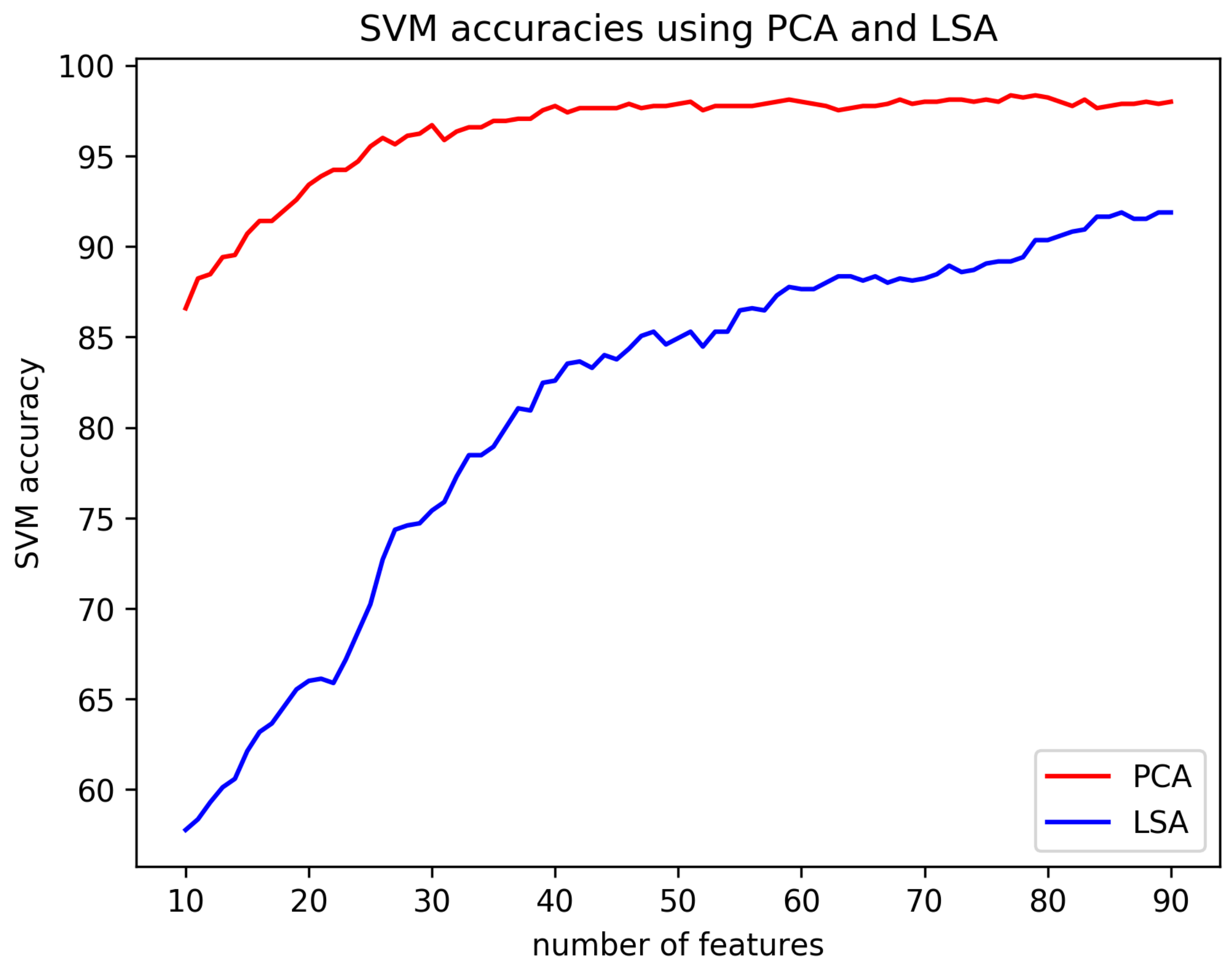
The SVM used squared hinge loss. That is (in the two-class setting with labels $y_i = \pm 1$ and points x_i) it fit a hyperplane $\{x \,|\, w^T x = b\}$ minimizing $\frac{1}{2}\|w\|^2 + \sum_i \max(0, 1 - y_i(w \cdot x_i - b))^2$.

Results

The PCA features were much more useful than the LSA features for authorship classification. Using PCA, 50 features were enough to get an accuracy of $\sim 98\%$ on the test set, while around 200 features were needed if LSA was used. The figure shows test set accuracy when SVMs are trained using between 10 and 90 features from PCA and LSA.

Data

85 different novels by 13 different authors were downloaded from Project Gutenberg[1] and split into 100 pieces of text each, giving a total of 8500 data points with 13 different labels. These were stemmed using NLTK[3] and represented as bags of words. The vocabulary consisted of 4731 words, and words not in the vocabulary were ignored. The data was split into a training set and a test set (a 90:10 split).



Discussion

As expected, PCA appears to be a much better method for dimensionality reduction for this purpose. This doesn't mean that LSA doesn't have a purpose, but in this particular setting, it doesn't work well. PCA may appear to be more difficult computationally, as it requires a normalization step which kills sparseness of the data, but standard algorithms are able to take advantage of the underlying sparseness anyway [2]. SVMs are a pretty basic model (not tailored to this particular data set), so this result is a good indication that PCA outperforms LSA more generally.

Naive Bayes

As a test that the features that were extracted contain sufficient information to distinguish the authors before any dimensionality reduction, we can run a Naive Bayes classifier (with the multinomial event model and Laplace smoothing). This gives a 99.53% accuracy, indicating that we have enough richness to classify authorship.

Future work

- Which features are most indicative of authorship (as opposed to e.g. text topic).
  - This might allow for unsupervised authorship classification.
- Are there better (e.g. nonlinear?) methods than PCA?

Bibliography

[1] Project Gutenberg. `https://www.gutenberg.org/`.

[2] James Baglama and Lothar Reichel. Augmented implicitly restarted Lanczos bidiagonalization methods. *SIAM Journal on Scientific Computing*, 27(1):19–42, 2005.

[3] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. O'Reilly Media, 2009.