LDA, Decision Trees, and Extra Trees
on the MNIST and Yale B Datasets

Kudiyar Orazymbetov, Nico Casale

NCSU

2017/04/23

outline

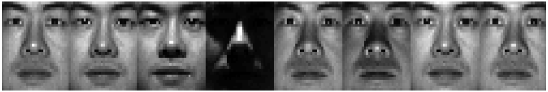
datasets

Modified Nat'l Institute of Standards and Technology (MNIST) database

- source: Yann LeCun et al. [1]
- 70k 28x28 images of handwritten digits (0-9)

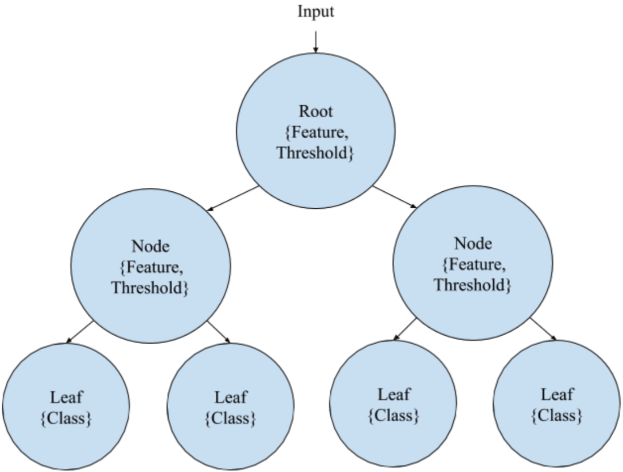
Yale Exended Face Database B

- source: Yale University [2]
- 2414 32x32 images of 38 subjects



progress

decision trees



features generated with LDA

training decision trees

recursive training algorithm [3] :

1. check stopping conditions
   - no more features
   - set is smaller than `minLeaf`
   - all samples in the same class
   - no feature improves information gain (IG)
2. iterate over each available feature, perform a line search to approximate the highest IG
3. recur over the subsets given by splitting at the feature and threshold with the highest IG

$$\mathrm{IG}(X) = \mathrm{H}(X) - \sum_{i=1}^{2} \frac{|S_i|}{|X|} \mathrm{H}(S_i) \tag{1}$$

$$\mathrm{H}(X) = -\sum_{i=1}^{n} \mathrm{P}(x_i) \log_2 \mathrm{P}(x_i) \tag{2}$$

progress

1 datasets

2 decision trees

3 extra trees

4 linear discriminant analysis (LDA)

5 results

6 conclusion

extremely randomized (extra) trees

recursive, random training algorithm [4] :

1. check stopping conditions
   - no more features
   - set is smaller than `minLeaf`
   - all samples in the same class
2. choose *random* feature. simply use the *raw pixels as features*
3. find the mean and variance of this feature across the set. generate a random value from a *normal distribution* with this mean and variance
4. recur on the subsets obtained by splitting the parent set on the randomly chosen feature and threshold

ensemble of random trees votes on test data to build extra-tree classifier

progress

LDA - dimension reduction

Steps:

1. within-class scatter matrix for each class

$$\Sigma_i = \frac{1}{N_i - 1} \sum_{\boldsymbol{x} \in D_i}^{n} (\boldsymbol{x} - \boldsymbol{\mu}_i)\,(\boldsymbol{x} - \boldsymbol{\mu}_i)^T \tag{3}$$

   then sum them to obtain

$$\Sigma_W = \sum_{i=1}^{n} (N_i - 1)\Sigma_i \tag{4}$$

2. between-class scatter matrix

$$\Sigma_B = \sum_{i=1}^{n} \frac{N_i}{N} (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \tag{5}$$

3. find eigenvectors and eigenvalues of $\Sigma^{-1}\Sigma_b$
4. using the eigenvectors we transform our data onto a new subspace

LDA - classification

Assumptions:

- we assume normality for each independent variable
- homogeneity of variance/covariance
- independence between samples

Classification steps:

- for each class μ_i, we compute

$$f_i(x_k) = \mu_i {w_a}^{-1} {x_k}^T - \frac{1}{2} \mu_i {w_a}^{-1} {\mu_i}^T + ln(P_i) \tag{6}$$

- we classify it to the class with $\max(f_i(x_k))$

progress

1 datasets

2 decision trees

3 extra trees

4 linear discriminant analysis (LDA)

5 results

6 conclusion

results

best performance

algorithm	MNIST	Yale B
LDA	13.5%	6.8%
decision tree	16.6%	57.9%
extra-trees	4.9%	34%

5-fold cross-validated performance

algorithm	MNIST	Yale B
LDA	14.5%	2.5%
decision Tree	17.9%	74.9%
extra-trees	5.4%	36.3%

conclusion

- intuition needed for feature generation
- choice of algorithm depends on
  - type of data
  - time available
  - accuracy needed
- future work
  - alternative features
  - parallelize algorithms

references

Yann LeCun and Corinna Cortes.
MNIST handwritten digit database.
2010.

Yale face database b.

Wikipedia contributors.
C4.5 algorithm — wikipedia, the free encyclopedia, 2018.
[Online; accessed 14-March-2018].

Wikipedia contributors.
Random forest — Wikipedia, the free encyclopedia, 2018.
[Online; accessed 23-April-2018].

thanks!