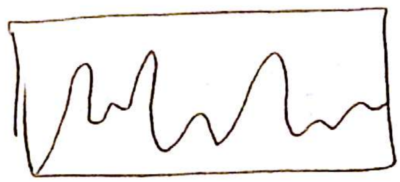
Supervised Dictionary Learning

report By :

Mohsen Nabian

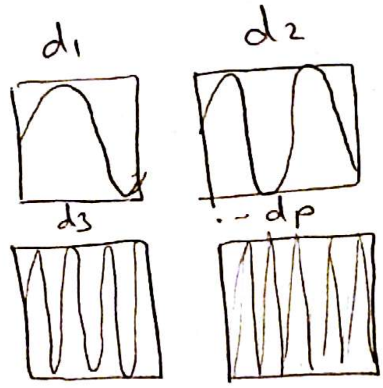
$$\underbrace{y}_{\text{measurements}}^{\text{image}} = \underbrace{X_{orig}}_{\text{original image}} + \underbrace{w}_{\text{noise}}$$

x

Let x in $\mathbb{R}^m$ be a signal: 

Let $D = [d_1, \dots, d_p] \in \mathbb{R}^{m \times p}$ be a set of normalized "basis vectors" we call it <u>dictionary</u>.

$D = [d_1, \dots d_p]$



D is adapted to y if it can represent it with a few basis vector - that is, there exist a sparse vector α, in $\mathbb{R}^p$ such that $y \approx D\alpha$

we call α the <u>sparse code</u>.

$\alpha \in \mathbb{R}^p$ sparse

$$\begin{pmatrix} y \end{pmatrix} \approx \begin{pmatrix} d_1 \,|\, d_2 \,|\, \cdots \,|\, d_p \end{pmatrix} \begin{pmatrix} \alpha_0[1] \\ \alpha[2] \\ \vdots \\ \alpha[p] \end{pmatrix}$$

let $(y^i, x^i)_{i=1}^n$ be a training set, where vector x^i are in $\mathbb{R}^p$ and called features. The scalars y^i are in

$$y^i \begin{cases} \{-1, +1\} & \text{for binary class classification.} \\ \{1, \dots, N\} & \text{for multi class} \quad \text{"} \\ \mathbb{R} & \text{for regression problems.} \end{cases}$$

in Linear model: $y \approx w^T x$ or $y \approx \text{sign}(w^T x)$

and solve:

$$\min_{w \in \mathbb{R}^p} \underbrace{\frac{1}{n} \sum_{i=1}^n \ell(y^i, w^T x^i)}_{\text{data fitting}} + \underbrace{\lambda \Omega(w)}_{\text{regularization}}$$

Optimization for Dictionary learning

$$\min_{\substack{\alpha \in \mathbb{R}^{p \times n} \\ D \in C}} \sum \frac{1}{2} \| y_i - D\alpha_i \|_2^2 + \lambda \|\alpha_i\|_1 \qquad (*)$$

$$C \triangleq \{ D \in \mathbb{R}^{m \times p} \text{ s.t. } \forall j = 1, \dots, p, \ \|d_j\|_2 \leq 1 \}$$

·) classical optz. alternate between D & α. but they are <u>slow</u>.

- ~~we~~ we can represent each data as a ~~Comb~~ sparse linear Combination of ~~dictionary~~ dictionary vectors: assume we know dictionary D and x given $\Rightarrow$ we have to find sparse Coefficients α:

~~$R^*(x, D)$~~

$$R^*(x, D) = \min_{\alpha \in \mathbb{R}^k} \|x - D\alpha\|_2^2 + \lambda_1 \|\alpha_0\|_1$$

in this paper, we consider ~~a~~:

- each signal x_i belongs to a class $y_i \in \{y_1, y_2, \dots y_p\}$
- Consider $p = 2 \Rightarrow y_i \in \{-1, +1\}$
- goal: learn jointly a single dictionary D adapted to classification task and function f which should be positive for each signal in class $+1$ and negative otherwise.

model i) [linear in α]: $f(x, \alpha, \theta) = w^T\alpha + b$

(L) where $\theta = \{w \in \mathbb{R}^k, b \in \mathbb{R}\}$

model ii) [bilinear in x and α]

(BL) $f(x, \alpha, \theta) = x^T W \alpha + b$

where $\theta = \{W \in \mathbb{R}^{n \times k}, b \in \mathbb{R}\}$

model ii has more parameter ~~so~~ is richer model.

So we must ~~know~~ learn α first and then ~~θ~~ Learn f parameters.

we classically obtain α as follows:

$$\min_{D,\alpha} \sum_{i=1}^{m} \|x_i - D\alpha_i\|_2^2 + \lambda_1 \|\alpha_i\|_1 \qquad (2)$$

(2) will end up with sparse α_i and ~~close~~ Singledictionary D

Then we learn function parameters from supervised learning :

$$\min_{\theta} \sum_{i=1}^{m} C(y_i f(x_i, \alpha_i, \theta)) + \lambda_2 \|\theta\|_2^2 \qquad (3)$$

$$C(x) = \log(1+e^{-x})$$

However our goal is to learn D and θ jointly.

$\Rightarrow$ we propose:

$$\min_{D,\theta,\alpha} \left\{ \left(\sum_{i=1}^{m} C(y_i f(x_i, \alpha_i, \theta)) + \lambda_0 \|x_i - D\alpha_i\|_2^2 + \lambda_1 \|\alpha_i\|_1 \right) + \lambda_2 \|\theta\|_2^2 \right\}$$

SDL-G (4)

$\lambda_0, \lambda_1, \lambda_2$

training D ✓ θ ✓

define: $S(\alpha, x_i, D, \theta, y_i) = C(y_i f(x_i, \alpha_i, \theta)) + \lambda_0 \|x_i - D\alpha_i\|_2^2 + \lambda_1 \|\alpha_i\|_1$

$$S^*(x_i, D, \theta, y_i) = \min_{\alpha} S(\alpha, x_i, D, \theta, y_i)$$

Once we obtained D, θ from (4):

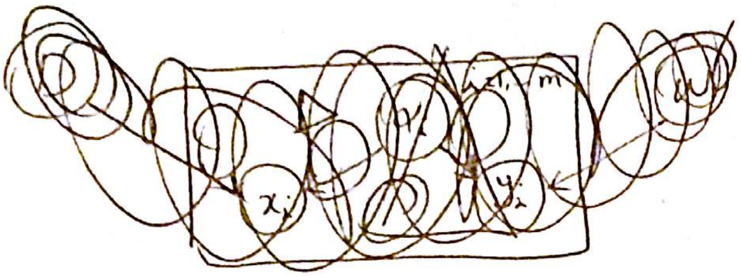
given a new signal x with unknown label y

classification is as follows:

$$(6) \quad \boxed{\min_{y \in \{-1,+1\}} S^*(x, D, \theta, y)}$$

$\leftarrow$ classification of test data

~~Graphical Model for proposed learning framework~~:



Since ~~class~~ test classification (6) compares different costs $S^*(x,D,\theta,y)$ we have to make sure cost $S^*(x_i,D,\theta,-y_i)$ is larger than $S^*(x_i,D,\theta,y_i)$. Therefore we modify (4) to

$$\min \left\{ \sum_{i=1}^{m} \mu C\left(S^*(x_i,D,\theta,-y_i) - S^*(x_i,D,\theta,y_i) \right) + (1-\mu) S^*(x_i,D,\theta,y_i) + \lambda_2 \|\theta\|_2^2 \right\} \tag{8}$$

μ : trade-off coef.

↳ SDL-D

learning jointly D, θ using training set

we can extend (8) to multi-class classification by soft-max.

① Then in the paper the algorithm for SDL-D ((8)) is provided which I do not write it here.

② (6) which has D and θ given, is a <u>Convex</u> optimization. The author suggest to use fixed-point-continuation-method (FPC) to solve this minimization.

~~Experimental Validation~~

Graphical Model of SDL model:

The corresponding graphical model for our model is as follows:

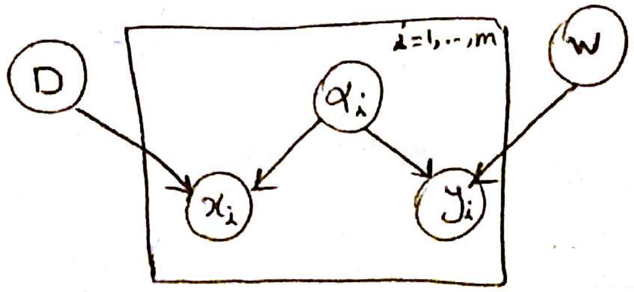


plate model

Experimental Validation :

datasets : (handwritten digits)

MNIST : 70000 28x28 image 60000 train 10000 test

USPS : 16x16 image 7291 train 2007 test

assuming $k = \frac{\lambda_1}{\lambda_0}$ K : size of dictionary

Five-fold cross validation on :

$k = \{0.13, 0.14, 0.15, 0.16, 0.17\}$

$K = \{24, 32, 48, 64, 96\}$

classification results:

	RECL	SDL-GL	SDL-DL	REC DL	k-NN	SVM
MNIST	4.33	3.5	1.05	3.4	5.0	1.4
USPS	6.83	6.67	3.54	4.3	5.2	4.2

$\Downarrow$

outperforms others.

2nd experiment:

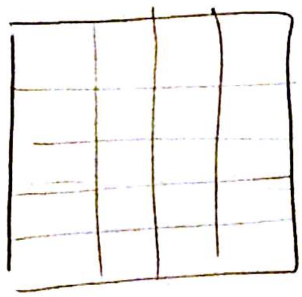
"are the obtained D discriminative per se?"

Trained USPS 10 binary classifiers (one vs all)

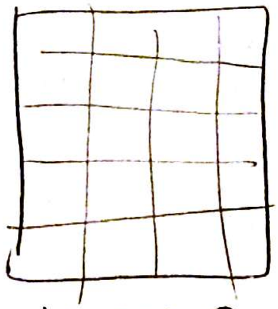
So for given value $\mu \Rightarrow$ obtain SDL-DL model $\left\{ \begin{array}{l} \text{10 dictionary } D \\ \text{10 params } \theta \end{array} \right.$

So we decompose each image $\xRightarrow{\text{gives}} \alpha$

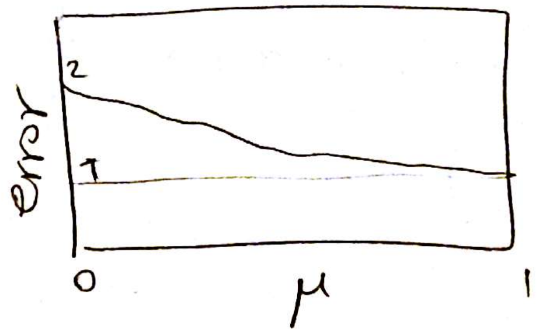
then use α in SVM. $\Rightarrow$ error plot provided below vs μ $\Downarrow$



(a) REC dictionaries



(b) SDL-D dictionaries



3rd experiment

Texture classification:

in experiment 1, L model outperformed BL model. one reason could be simplicity of the problem.

Two different textures (2 classes) are provided.

each image is breaked to 12x12 patches.

all left half patches ⟹ training

" right " " ⟹ testing

- Error rate of classification are provided. and showed that BL models significantly outperformed L models. and the reason is the complexity of the problem.