太原理工大学
TAIYUAN UNIVERSITY OF TECHNOLOGY

太原理工大学
大数据学院
COLLEGE OF DATA SCIENCE
TAIYUAN UNIVERSITY OF TECHNOLOGY

深度学习中的正则化

Regularization for Deep Learning

本章内容概况

主要内容

2

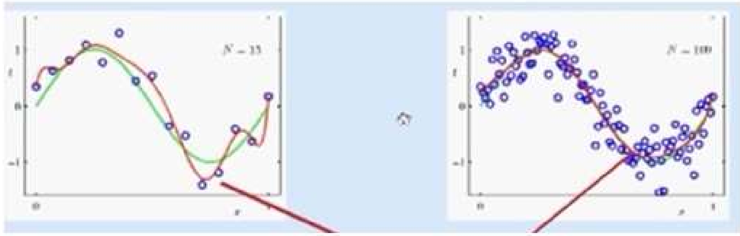
PART

数据增强与
提前终止

ONE

数据增强与提前终止

数据增强



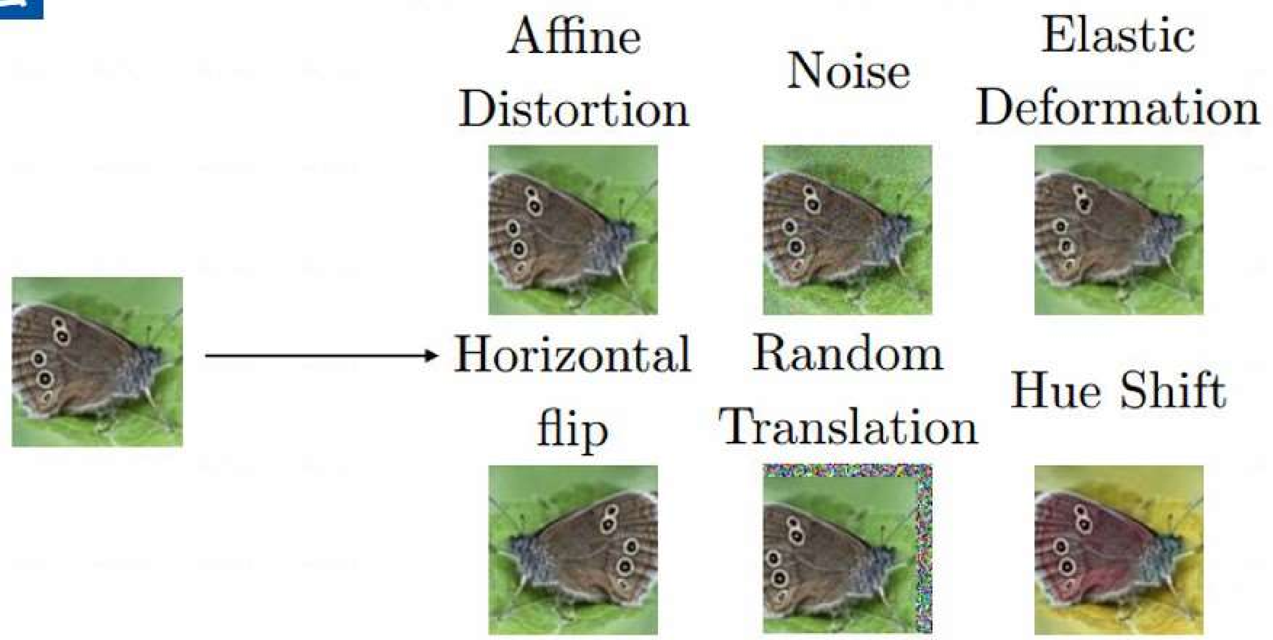
- 图像数据的增强主要是通过算法对图像进行转变，引入噪声等方法来增加数据的多样性。

- 图像数据的增强方法:
  - 旋转（Rotation）： 将图像按顺时针或逆时针方向随机旋转一定角度;
  - 翻转（Flip）： 将图像沿水平或垂直方法随机翻转一定角度;
  - 缩放（Zoom In/Out）： 将图像放大或缩小一定比例;
  - 平移（Shift）： 将图像沿水平或垂直方法平移一定步长;
  - 加噪声（Noise）： 加入随机噪声。

4

数据增强与提前终止

数据增强

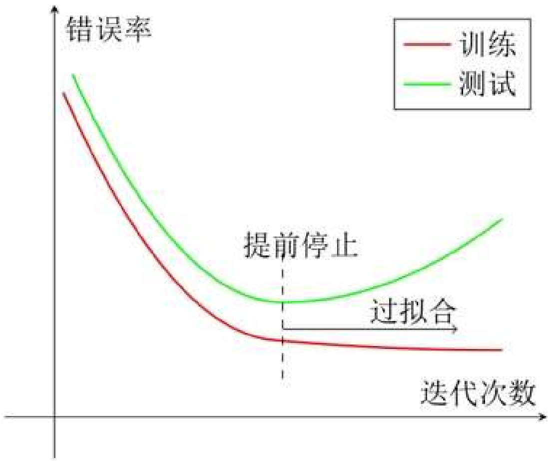


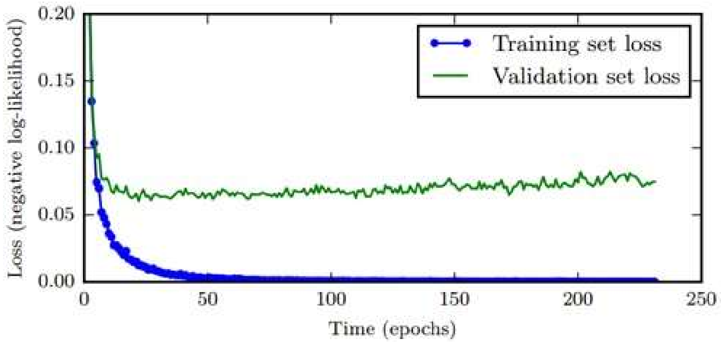
雨课堂 Rain Classroom

数据增强与提前终止

提前终止

- 我们使用一个验证集（Validation Dataset）来测试每一次迭代的参数在验证集上是否最优。如果在验证集上的错误率不再下降，就停止迭代。





6

数据增强与提前终止

提前终止

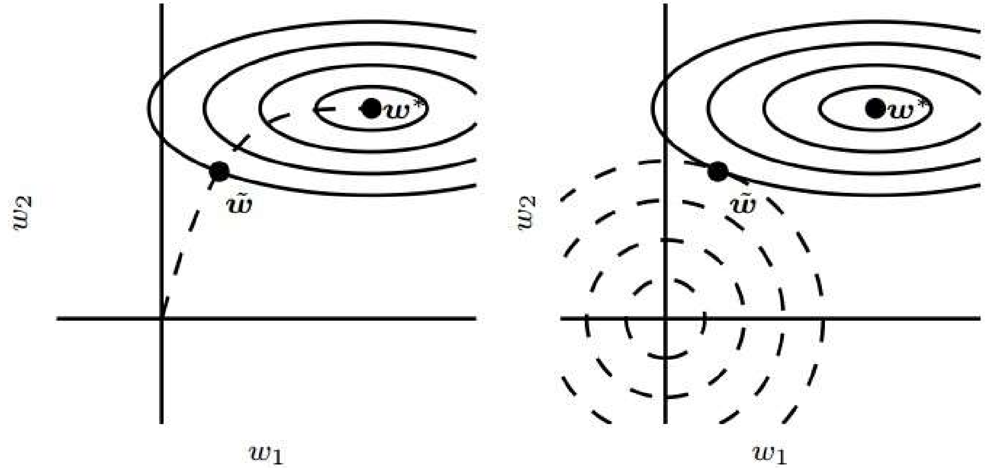


图 7.4: 提前终止效果的示意图。*(左)* 实线轮廓线表示负对数似然的轮廓。虚线表示从原点开始的 SGD 所经过的轨迹。提前终止的轨迹在较早的点 $\tilde{\boldsymbol{w}}$ 处停止，而不是停止在最小化代价的点 $\boldsymbol{w}^*$ 处。*(右)* 为了对比，使用 L^2正则化效果的示意图。虚线圆圈表示 L^2 惩罚的轮廓，L^2 惩罚使得总代价的最小值比非正则化代价的最小值更靠近原点。

7

PART

Dropout

TWO

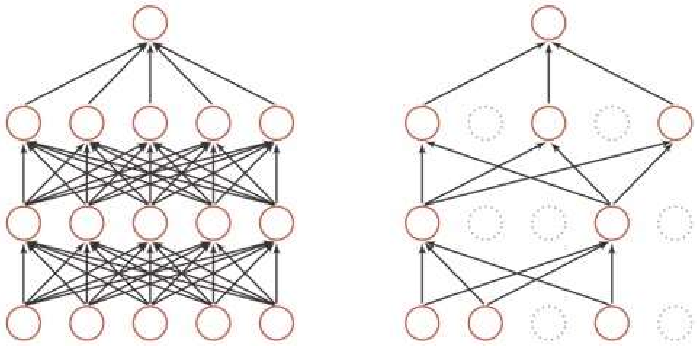
Dropout

Definition

- 对于一个神经层$y = f(Wx + b)$，引入一个丢弃函数$d(\cdot)$使得$y = f(Wd(x) + b)$。

$$d(\mathbf{x}) = \begin{cases} \mathbf{m} \odot \mathbf{x} & \text{当训练阶段时} \\ p\mathbf{x} & \text{当测试阶段时} \end{cases}$$

- 其中$m \in \{0,1\}^d$ 是丢弃掩码（dropout mask），通过以概率为p的贝努力分布随机生成。

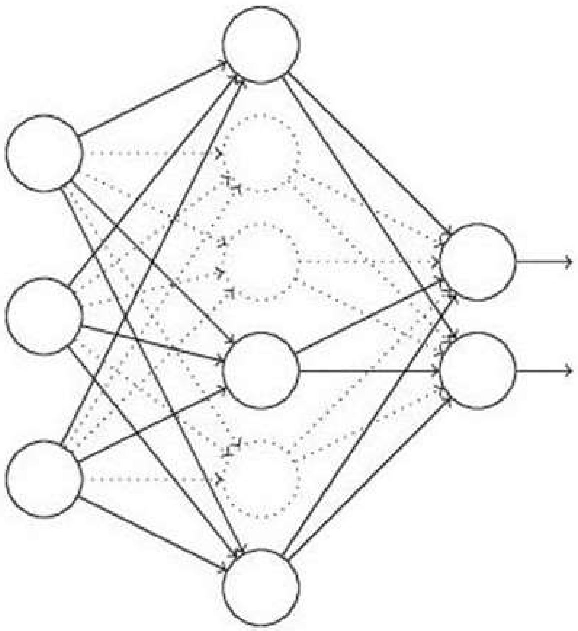


(a) 标准网络　　(b) Dropout 后的网络

9

Dropout

Definition



首先随机（临时）删掉网络中一半的隐藏神经元，输入输出神经元保持不变（图中虚线为部分临时被删除的神经元）

然后把输入x通过修改后的网络前向传播，然后把得到的损失结果通过修改的网络反向传播。一小批训练样本执行完这个过程后，在没有被删除的神经元上按照随机梯度下降法更新对应的参数（w，b）。

然后继续重复这一过程：
恢复被删掉的神经元（此时被删除的神经元保持原样，而没有被删除的神经元已经有所更新）
从隐藏层神经元中随机选择一个一半大小的子集临时删除掉（备份被删除神经元的参数）。
对一小批训练样本，先前向传播然后反向传播损失并根据随机梯度下降法更新参数（w，b）（没有被删除的那一部分参数得到更新，删除的神经元参数保持被删除前的结果）。

10

Dropout

Definition

- At training (each iteration):

  Each unit is retained with a probability p.

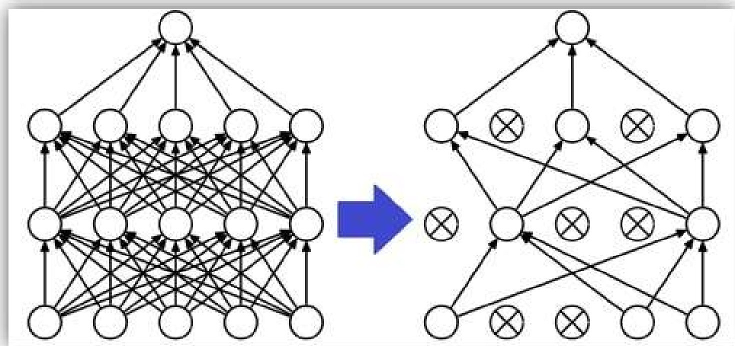
- At test:

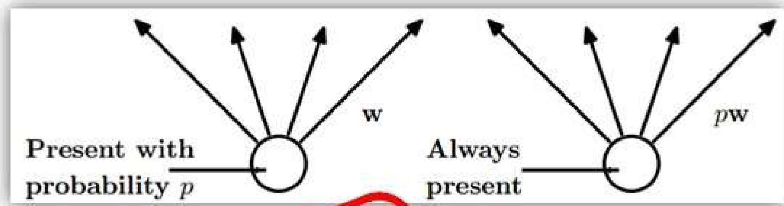
  The network is used as a whole.

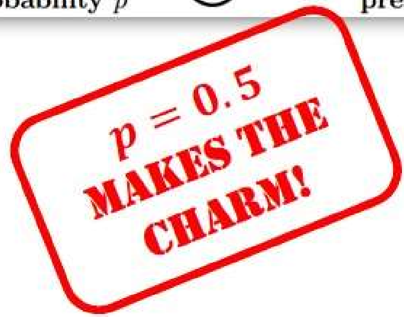
  The weights are scaled-down by a factor of p.

- In practice, dropout trains 2^n networks

  (n – number of units).







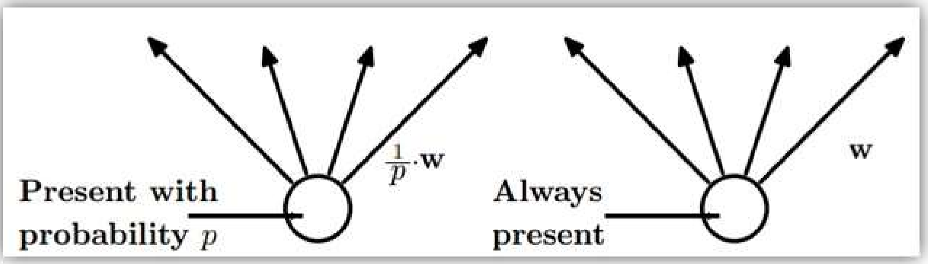
11

Dropout

Definition

- At training, weights are scaled-up by a factor of $\frac{1}{p}$.
- At test time, no scaling is applied.
- This method is used in Tensorflow:

```
tf.nn.dropout(x, keep_prob=p)
```



12

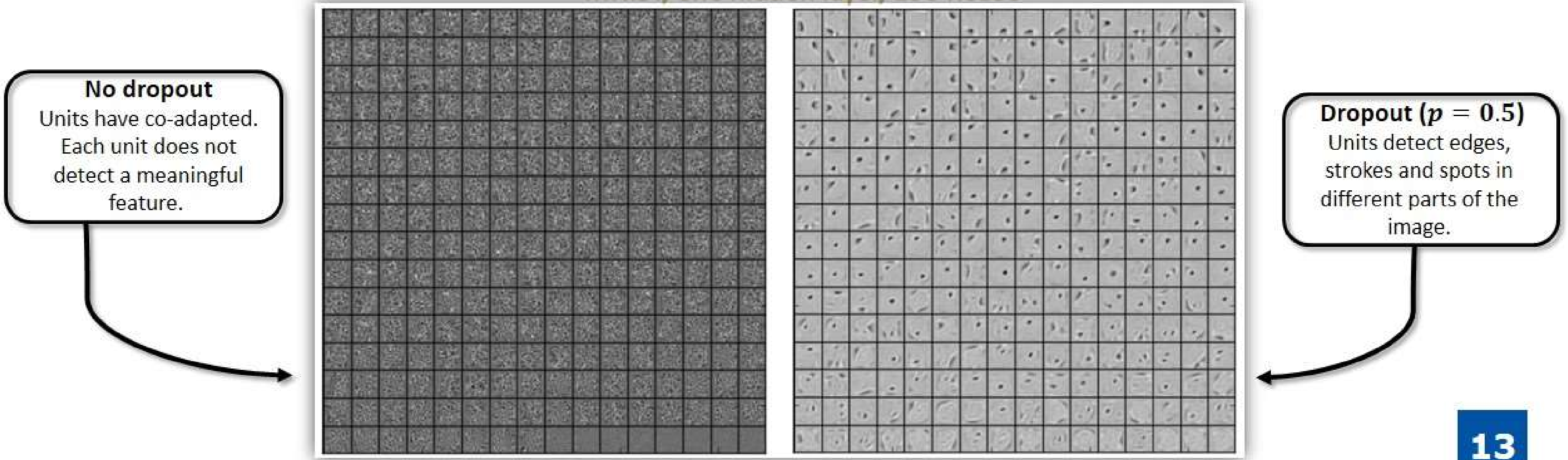
Dropout

实验结果对比

The effect of dropout on learned features:

- Without dropout, units tend to **compensate for mistakes** of other units.
- This leads to overfitting, since **these co-adaptations do not generalize to unseen data**.
- Dropout prevents co-adaptations by making the presence of other hidden units unreliable.

MNIST, one hidden layer, 256 ReLUs



13

Dropout

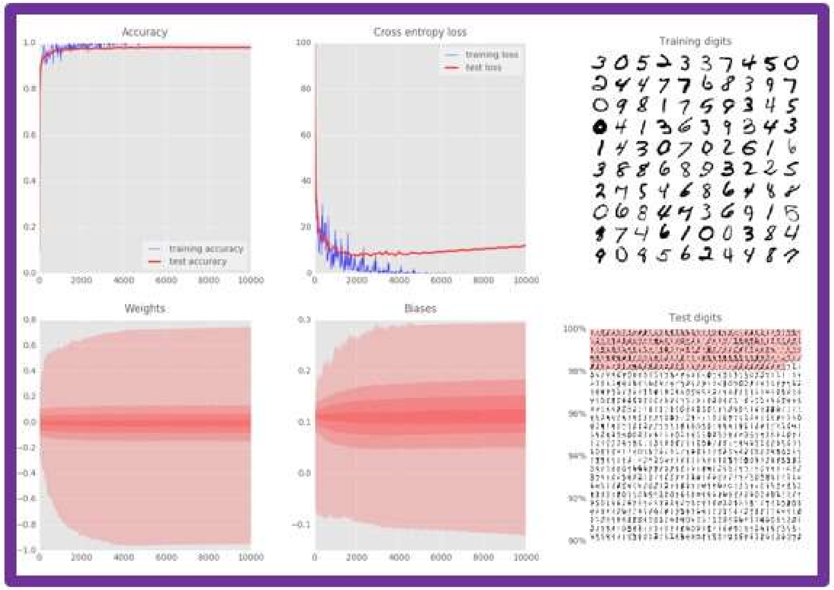
代码不同

```
# The model
XX = tf.reshape(X, [-1, 784])
Y1 = tf.nn.relu(tf.matmul(XX, W1) + B1)
Y2 = tf.nn.relu(tf.matmul(Y1, W2) + B2)
Y3 = tf.nn.relu(tf.matmul(Y2, W3) + B3)
Y4 = tf.nn.relu(tf.matmul(Y3, W4) + B4)
Ylogits = tf.matmul(Y4, W5) + B5
Y = tf.nn.softmax(Ylogits)
```

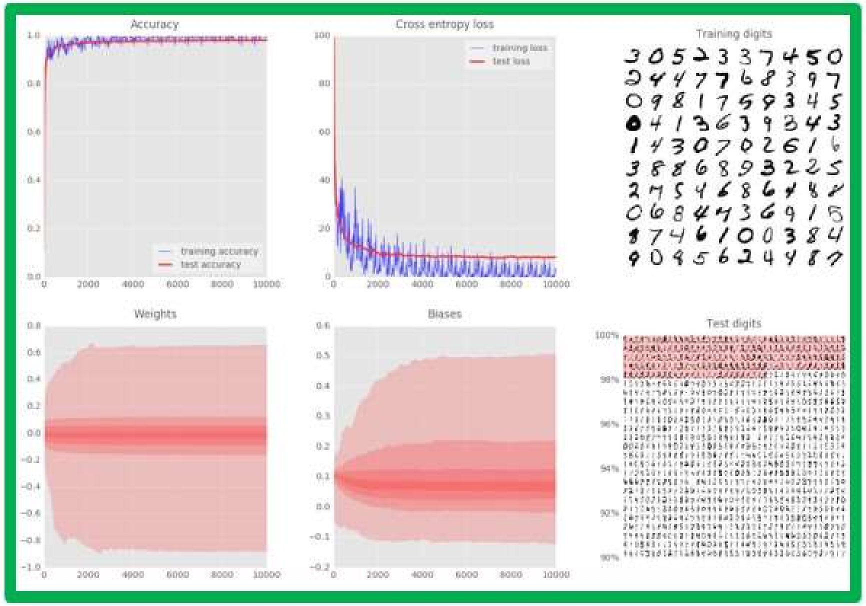
```
# The model, with dropout at each layer
XX = tf.reshape(X, [-1, 28*28])

Y1 = tf.nn.relu(tf.matmul(XX, W1) + B1)
Y1d = tf.nn.dropout(Y1, pkeep)

Y2 = tf.nn.relu(tf.matmul(Y1d, W2) + B2)
Y2d = tf.nn.dropout(Y2, pkeep)

Y3 = tf.nn.relu(tf.matmul(Y2d, W3) + B3)
Y3d = tf.nn.dropout(Y3, pkeep)

Y4 = tf.nn.relu(tf.matmul(Y3d, W4) + B4)
Y4d = tf.nn.dropout(Y4, pkeep)

Ylogits = tf.matmul(Y4d, W5) + B5
Y = tf.nn.softmax(Ylogits)
```

Dropout

性能对比



No Dropout



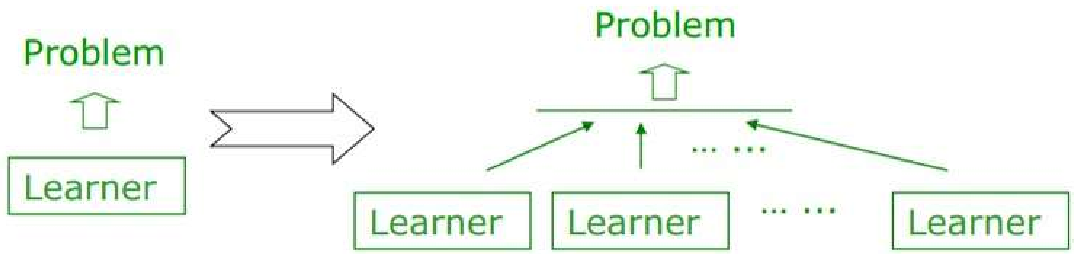
Dropout

Dropout

集成学习

5道题
3个学生
每人会做2道

"Ensemble methods" is a machine learning paradigm where multiple (homogenous/heterogeneous) individual learners are trained for the same problem
e.g. neural network ensemble, decision tree ensemble, etc.



The more **accurate** and **diverse** the component learners, the better the ensemble

16

Dropout

Bagging和Dropout

Dropout as bagging

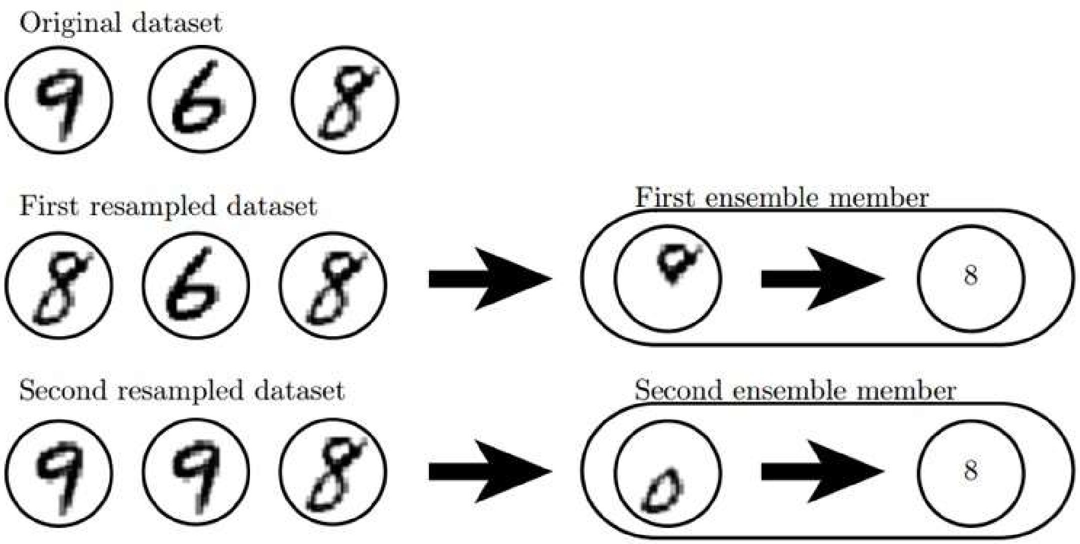
- In bagging we define k different models, construct k different data sets by sampling from the dataset with replacement, and train model i on dataset i
- Dropout aims to approximate this process, but with an exponentially large no. of neural networks

17

Dropout

Bagging

Bagging



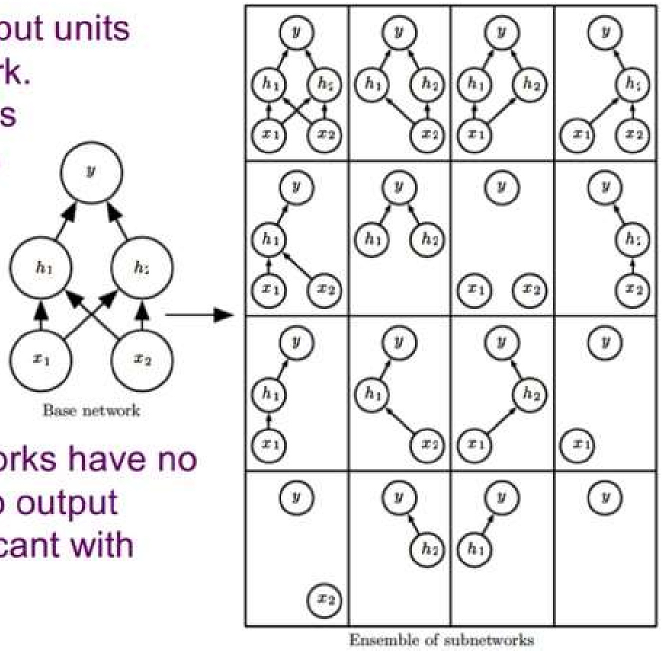
18

Dropout

Bagging和Dropout的联系

Dropout as an ensemble method

- Remove non-output units from base network.
- Remaining 4 units yield 16 networks



- Here many networks have no path from input to output
- Problem insignificant with large networks

19

Dropout

Bagging和Dropout的差异

Bagging training vs Dropout training

- Dropout training not same as bagging training
  - In bagging, the models are all independent
  - In dropout, models share parameters
    - Models inherit subsets of parameters from parent network
    - Parameter sharing allows an exponential no. of models with a tractable amount of memory
- In bagging each model is trained to convergence on its respective training set
  - In dropout, most models are not explicitly trained
    - Fraction of subnetworks are trained for a single step
    - Parameter sharing allows good parameter settings

20

Dropout

model description

Prediction: Bagging vs. Dropout

- Bagging:
  - Ensemble accumulates votes of members
  - Process is referred to as inference
    - Assume model needs to output a probability distribution
    - In bagging, model i produces $p^{(i)}(y|\boldsymbol{x})$
    - Prediction of ensemble is the mean $\boxed{\frac{1}{k}\sum_{i=1}^{k} p^{(i)}(y \mid \boldsymbol{x})}$
- Dropout:
  - Submodel defined by mask vector $\boldsymbol{\mu}$ defines a probability distribution $p(y|\boldsymbol{x},\boldsymbol{\mu})$
  - Arithmetic mean over all masks is $\boxed{\sum_{\mu} p(y \mid \boldsymbol{x},\mu)}$
    - Where $p(\boldsymbol{\mu})$ is the distribution used to sample $\boldsymbol{\mu}$ at training time

21

PART

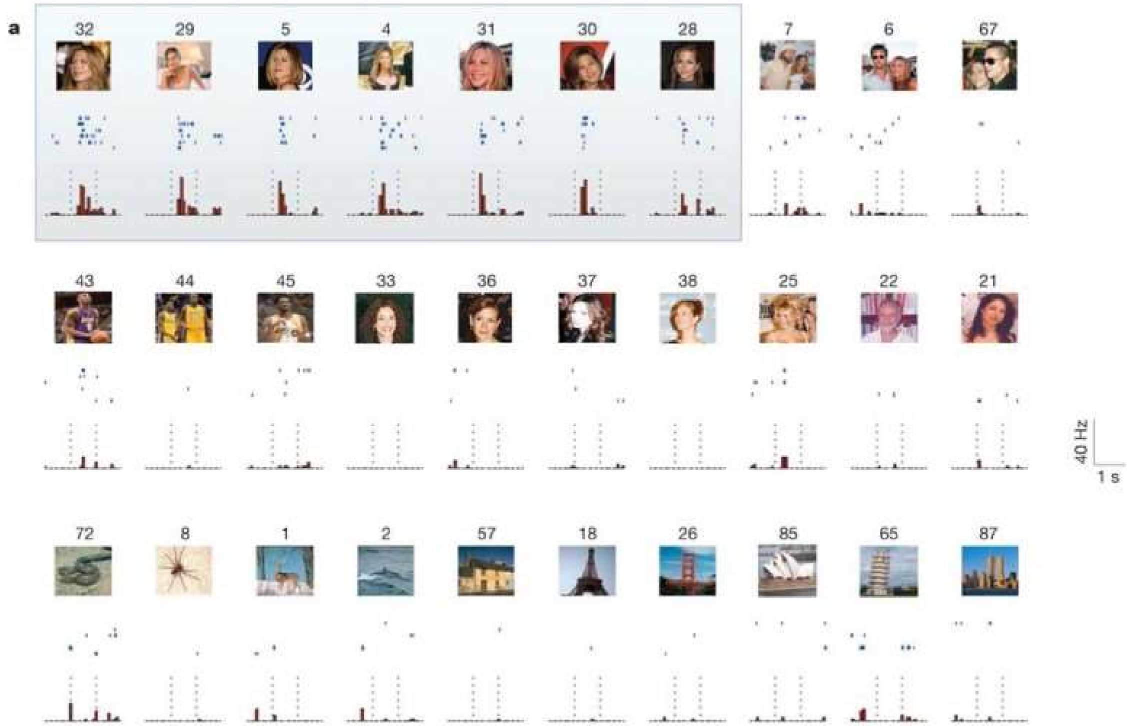
稀疏表示

THREE

概念

Definition

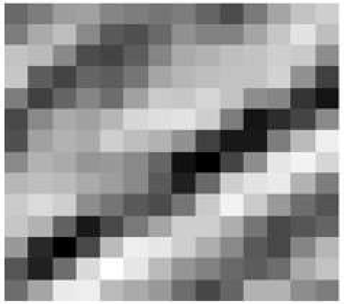
Grandmother cell



[Quiroga, Reddy, & Kreiman Nature2005]

稀疏表示

Definition

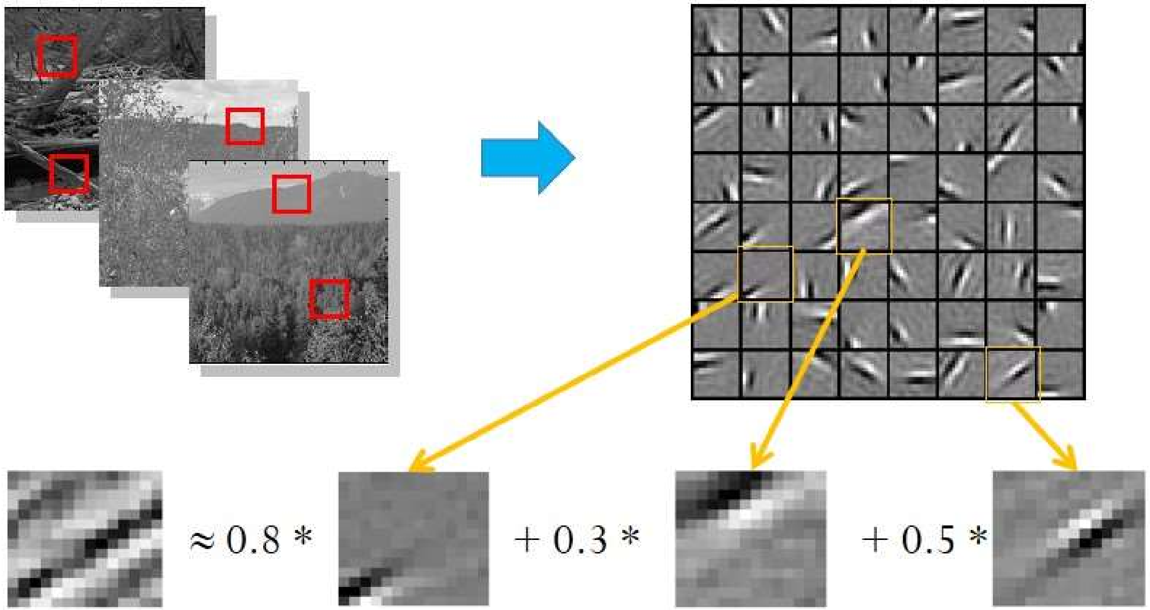


Learn a better way to represent image than pixels

24

稀疏表示

Definition



Code:
http://web.eecs.umich.edu/~honglak/softwares/nips06-sparsecoding.htm

$[a_1, \ldots, a_{64}] = [0, 0, \ldots, 0, \mathbf{0.8}, 0, \ldots, 0, \mathbf{0.3}, 0, \ldots, 0, \mathbf{0.5}, 0]$ (feature representation)

25

稀疏表示

Definition

Input: Patch x_i (in R^d) and previously learned ϕ_i $(i=1, \ldots, k)$

Output: Representation $[a_{i,1}, a_{i,2}, \ldots, a_{i,k}]$ of image patch x_i

$$\min_{a,\phi} \sum_{i=1}^{m} \left(\left\| x_j - \sum_{j=1}^{k} a_{i,j}\phi_i \right\|^2 + \sum_{j=1}^{k} | a_{i,j} | \right)$$

稀疏表示

Definition

Input: Patch x_i (in R^d) and previously learned ϕ_i ($i=1, \ldots ,k$)

Output: Representation $[a_{i,1}, a_{i,2}, \ldots , a_{i,k}]$ of image patch x_i

$$\min_{a,\phi} \sum_{i=1}^{m}\left(\left\| x_j - \sum_{j=1}^{k} a_{i,j}\phi_i \right\|^2 + \sum_{j=1}^{k} | a_{i,j} | \right)$$

稀疏表示

Definition

Input: Patch x_i (in R^d) and previously learned ϕ_i $(i=1, \ldots ,k)$

Output: Representation $[a_{i,1}, a_{i,2}, \ldots , a_{i,k}]$ of image patch x_i

$$\min_{a,\phi} \sum_{i=1}^{m} \left(\left\| x_j - \sum_{j=1}^{k} a_{i,j}\phi_i \right\|^2 + \boxed{\sum_{j=1}^{k} |a_{i,j}|} \right)$$

l_1 sparsity term

稀疏表示

Definition

Input: Patch x_i (in R^d) and previously learned ϕ_i $(i=1, \dots ,k)$

Output: Representation $[a_{i,1}, a_{i,2}, \dots , a_{i,k}]$ of image patch x_i

$$\min_{a,\phi} \sum_{i=1}^{m} \left(\left\| x_j - \sum_{j=1}^{k} a_{i,j} \phi_i \right\|^2 \boxed{+ \sum_{j=1}^{k} | a_{i,j} |} \right)$$

l_1 sparsity term

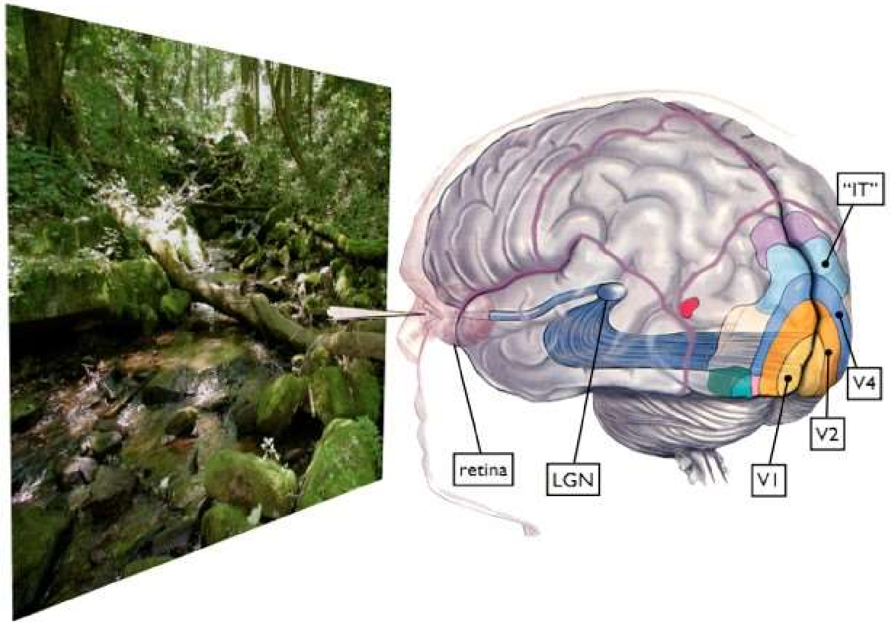
Alternating optimization:

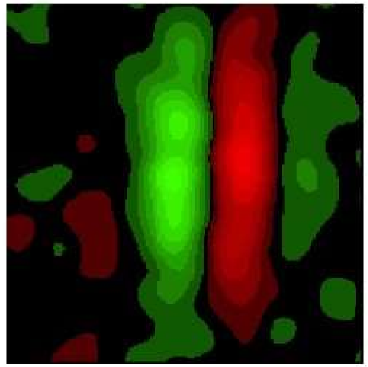
1. Fix dictionary ϕ, optimize a (LASSO problem) Harder
2. Fix activations a, optimize dictionary ϕ (convex QP problem) Easy

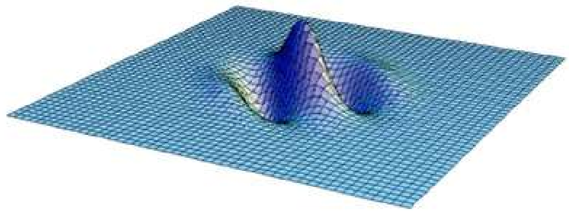
29

稀疏表示

Definition





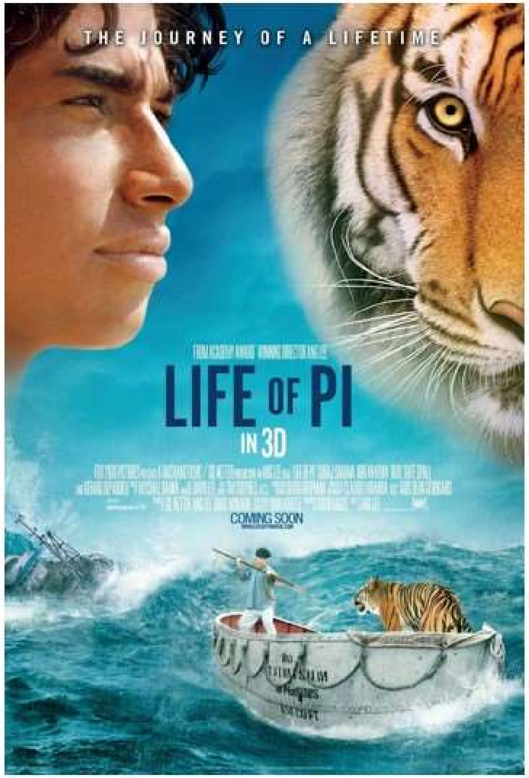


[Olshausen, Field. Nature1996]

稀疏表示

Definition

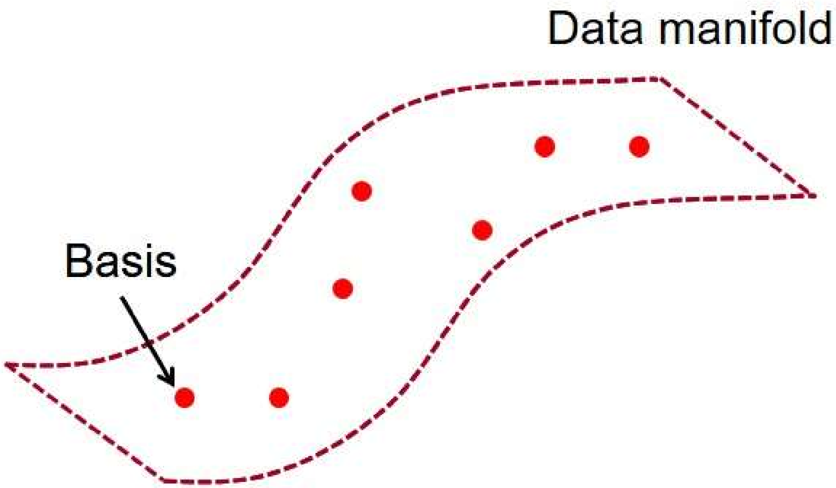
Believing in everything at
the same time
is the same as not believing
in anything at all



31

稀疏表示

Definition

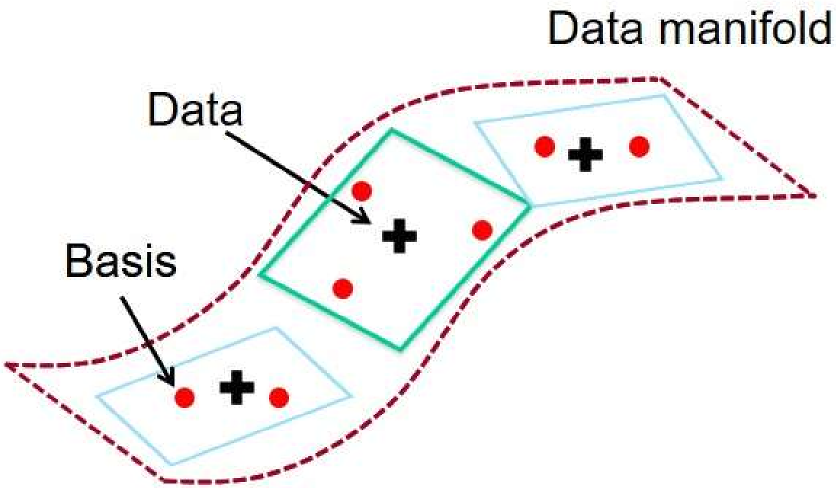


- Each basis is somewhat like a pseudo data point – “anchor point”
- Sparsity: each datum is a sparse combination of neighbor anchors.
- The coding scheme explores the manifold structure of data.

32

稀疏表示

Definition



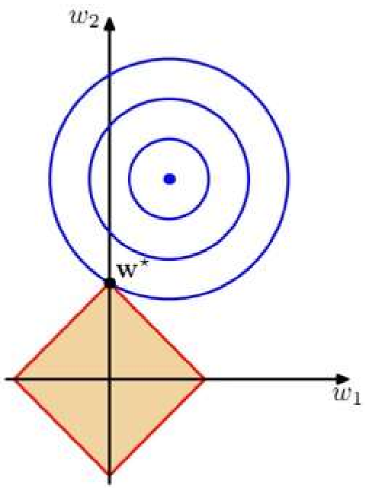
- Each basis is somewhat like a pseudo data point – “anchor point”
- Sparsity: each datum is a sparse combination of neighbor anchors.
- The coding scheme explores the manifold structure of data.

33

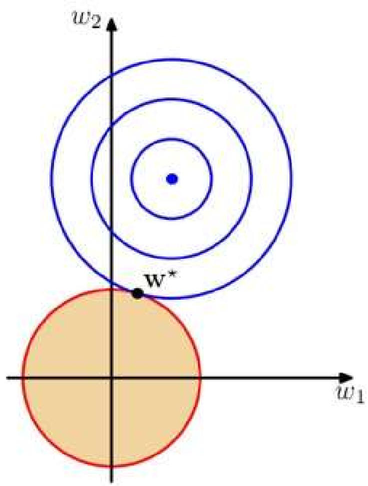
稀疏表示

Definition

$$\min_{a,\phi} \sum_{i=1}^{m} \left(\left\| x_j - \sum_{j=1}^{k} a_{i,j}\phi_i \right\|^2 + \sum_{j=1}^{k} | a_{i,j} | \right)$$



l_1

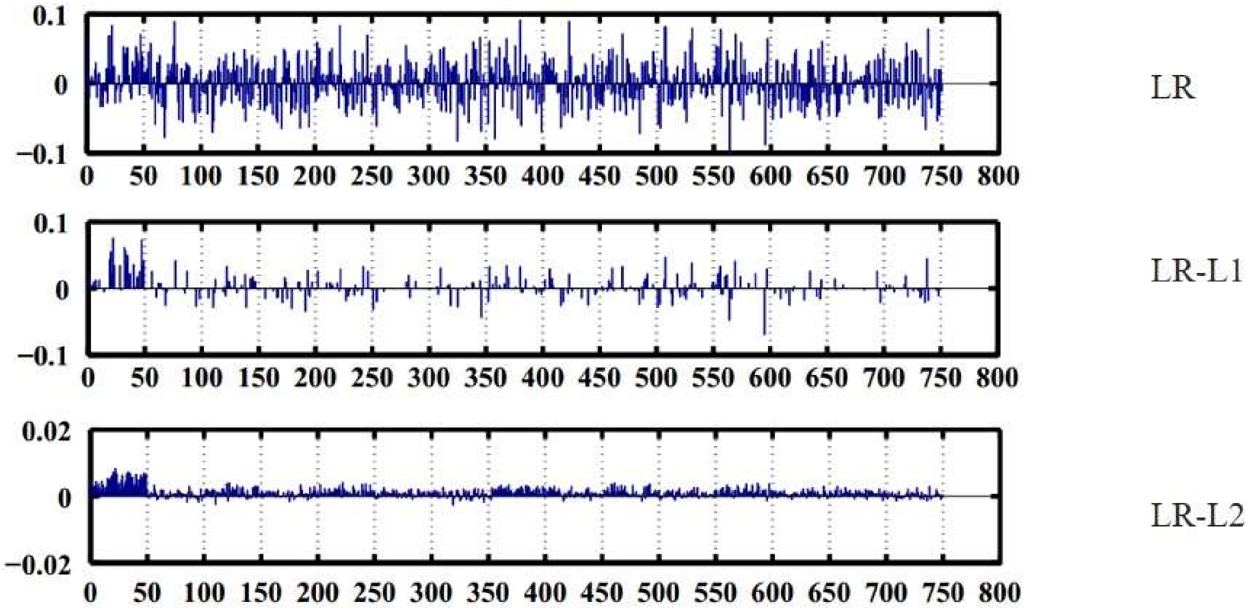


l_2

34

稀疏表示

DEMO

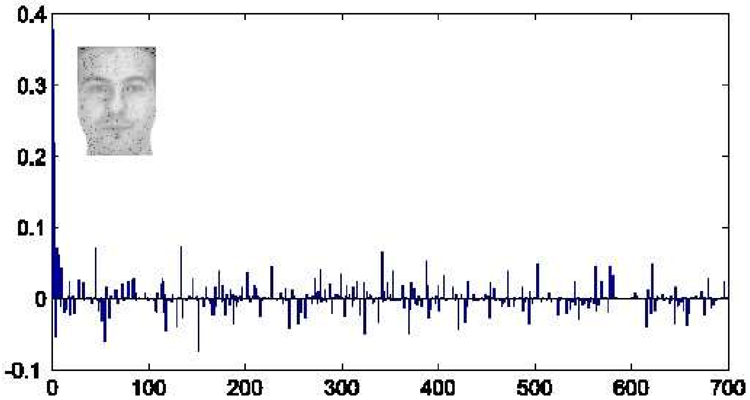

0.1
0
−0.1
0 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800
LR
0.1
0
−0.1
0 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800
LR-L1
0.02
0
−0.02
0 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800
LR-L2

35

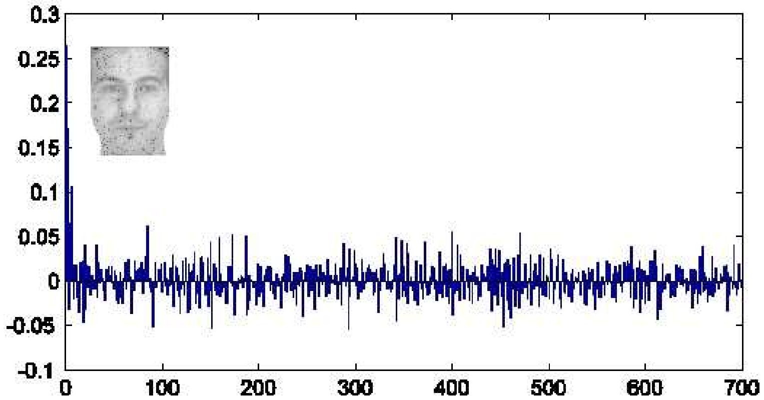
稀疏表示

DEMO

l_1



l_2



Demo

36

稀疏表示

KL **相对熵是一种标准的用来测量两个分布之间差异的方法**

$$\hat{\rho}_j = \frac{1}{m}\sum_{i=1}^{m}\left[a_j^{(2)}(x^{(i)})\right]$$

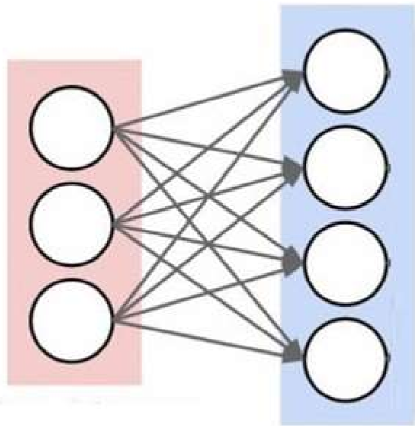
$$\sum_{j=1}^{s_2}\mathrm{KL}(\rho\|\hat{\rho}_j) = \sum_{j=1}^{s_2}\rho\log\frac{\rho}{\hat{\rho}_j} + (1-\rho)\log\frac{1-\rho}{1-\hat{\rho}_j}$$

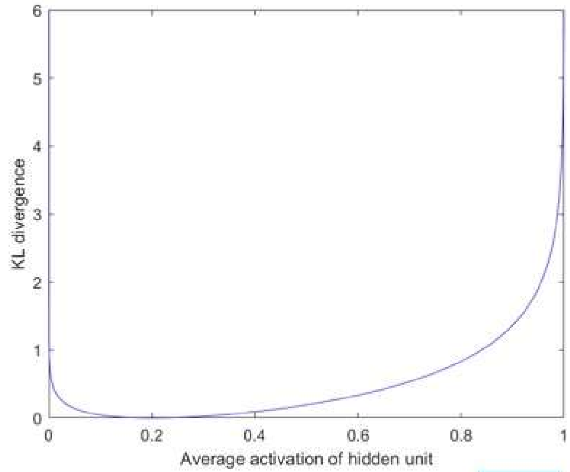
$$J_{\text{sparse}}(W,b) = J(W,b) + \beta\sum_{j=1}^{s_2}\mathrm{KL}(\rho\|\hat{\rho}_j)$$

相对熵在 $\hat{\rho}_j = \rho$ 时达到它的最小值0

而当 $\hat{\rho}_j$ 靠近0或者1的时候，相对熵则变得非常大（其实是趋向于∞）

最小化这一惩罚因子具有使得 $\hat{\rho}_j$ 靠近 ρ 的效果

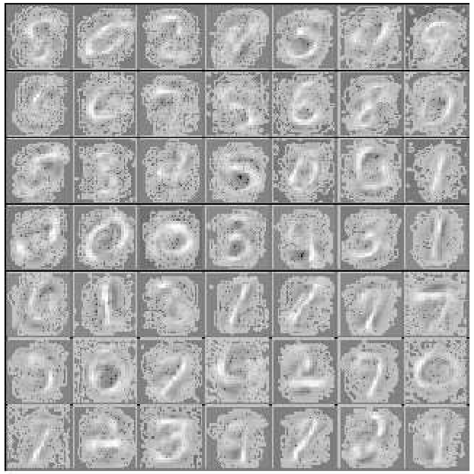




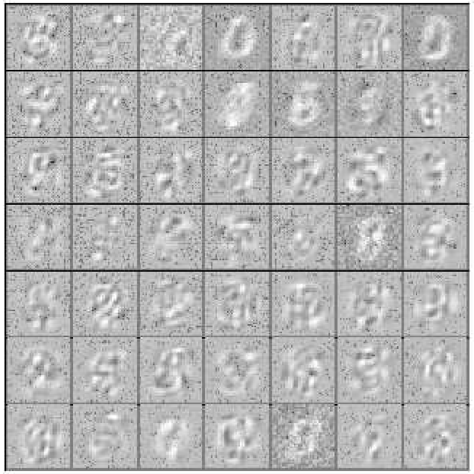
37

稀疏表示

KL



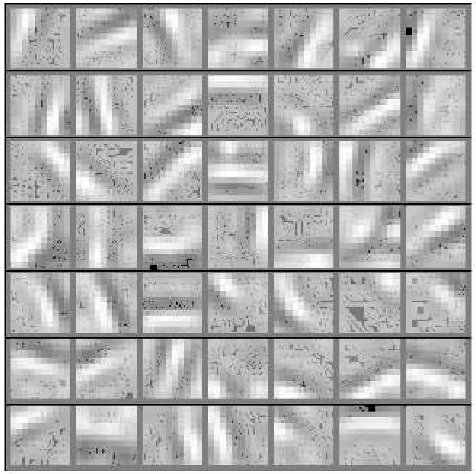
sparsity



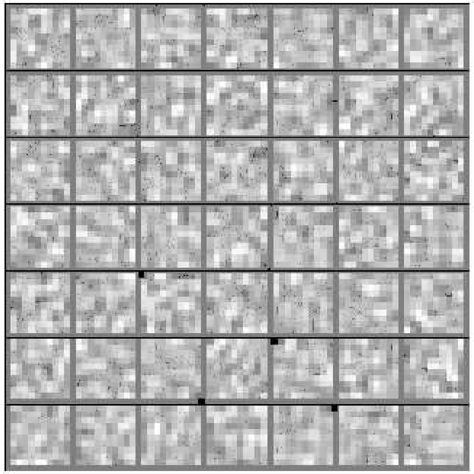
No sparsity

稀疏表示

KL



sparsity



No sparsity

Demo

THANK YOU
Q&A