

희소 투영행렬 획득을 위한 RSR 개선 방법론

안정호*

요약

본 논문은 패턴인식에서 자주 사용되는 투영행렬을 희소화하는 문제를 다룬다. 최근 임베디드 시스템이 널리 사용됨에 따라 탑재되는 프로그램의 용량이 제한받는 경우가 빈번히 발생한다. 개발된 프로그램은 상수 데이터를 포함하는 경우가 많다. 예를 들어, 얼굴인식과 같은 패턴인식 프로그램의 경우 고차원 벡터를 저차원 벡터로 차원을 축소하는 투영행렬을 사용하는 경우가 많다. 인식성능 향상을 위해 영상으로부터 매우 높은 차원의 고차원 특징벡터를 추출하는 경우 투영행렬의 사이즈는 매우 크다.

최근 라소 회귀분석 방법을 이용한 RSR(rotated sparse regression) 방법론[1]이 제안되었다. 이 방법론은 여러 실험을 통해 희소행렬을 구하는 가장 우수한 알고리즘 중 하나로 평가받고 있다. 우리는 본 논문에서 RSR을 개선할 수 있는 세 가지 방법론을 제안한다. 즉, 학습데이터에서 이상치를 제거하여 일반화 성능을 높이는 방법, 학습데이터를 랜덤 샘플링하여 희소율을 높이는 방법, RSR의 목적함수에 엘라스틱 넷 회귀분석의 페널티 항을 사용한 E-RSR(elastic net-RSR) 방법을 제안한다. 우리는 실험을 통해 제안한 방법론이 인식률을 희생하지 않으며 희소율을 크게 증가시킴으로써 기존 RSR 방법론을 개선할 수 있음을 보였다.

키워드 : RSR, 엘라스틱 넷 회귀분석, 희소투영행렬, 이상치, 샘플링

An Improved RSR Method to Obtain the Sparse Projection Matrix

Jung-Ho Ahn*

Abstract

This paper addresses the problem to make sparse the projection matrix in pattern recognition method. Recently, the size of computer program is often restricted in embedded systems. It is very often that developed programs include some constant data. For example, many pattern recognition programs use the projection matrix for dimension reduction. To improve the recognition performance, very high dimensional feature vectors are often extracted. In this case, the projection matrix can be very big.

Recently, RSR(roated sparse regression) method[1] was proposed. This method has been proved one of the best algorithm that obtains the sparse matrix. We propose three methods to improve the RSR; outlier removal, sampling and elastic net RSR(E-RSR) in which the penalty term in RSR optimization function is replaced by that of the elastic net regression. The experimental results show that the proposed methods are very effective and improve the sparsity rate dramatically without sacrificing the recognition rate compared to the original RSR method.

Keywords : RSR, elastic net regression, outlier, sampling, sparse projection matrix

1. 서론

※ Corresponding Author: Jung-Ho Ahn

Received : June 22, 2015

Revised : August 29, 2015

Accepted : August 31, 2015

* Division of Computer Media Information Engineering, Kangnam University.

Tel: +82-2-280-3661, Fax: +82-2-280-3979

email: jungho@kangnam.ac.kr

본 논문에서는 투영행렬(projection matrix)을 희소화(稀少化)하는 문제를 다룬다. 투영행렬을 희소화한다는 것은 성능저하가 크지 않는 범위에서 기존의 투영행렬을 적절한 희소행렬(sparse

■ 본 연구는 2013년도 강남대학교 교내연구비 지원에 의한 것임.

matrix)로 대치함을 말한다. 투영행렬은 고차원 데이터를 저차원 데이터로 대응시키는 행렬이다. 고차원 데이터를 저차원으로 투영하여 차원을 축소하면 처리 속도를 향상시킬 수 있을 뿐 아니라 차원간의 상관을 줄일 수 있어 얼굴, 문자 인식 등에서 성능향상에 큰 도움을 주는 경우가 많다. 최근 제안된 대부분의 얼굴인식 알고리즘들은 얼굴 영상으로 부터 상당히 높은 차원의 고차원 특징벡터를 추출한 후 이를 PCA 등의 투영행렬을 통해 저차원 벡터로 차원을 축소하여 분류한다[1, 2, 3, 4].

차원축소 방식은 선형 방식과 비선형 방식으로 나눌 수 있다. 대표적인 비선형 차원축소 방식으로는 KPCA[5], KLDA[6] 등 커널을 이용한 방식과 신경망을 이용한 방식[7]이 있다. 과거에는 비선형 차원 축소방식이 많이 연구되고 활용되었다. 하지만, 성능 및 복잡도 측면에서 비선형방식이 선형방식보다 반드시 우월하다고 볼 수 없기 때문에 최근에는 선형 차원축소방식이 널리 사용되고 있다. 대표적인 선형 차원축소 방식으로는 PCA, LDA 등이 있다[8].

선형 차원축소는 학습데이터를 이용하여 만든 투영행렬에 의해 수행된다. 매우 높은 고차원 (특징)벡터를 저차원 벡터로 차원축소 또는 투영하는 경우 투영행렬의 크기는 매우 크다. 참고로 4절의 실험에서 사용된 투영행렬 사이즈는 9440×200이다. 이 행렬은 각 원소를 4바이트 부동소수점 변수로 저장하는 경우 4×9440×200 = 7,552,000 바이트, 약 7.5 MB를 차지한다. 이는 우리가 수행중인 얼굴인식 프로그램 총 용량의 40%이상을 차지한다. 일반적으로, 이러한 과도한 데이터 용량으로 인해 개발된 프로그램이 임베디드 플랫폼에 부적합한 경우가 빈번히 발생한다. 이러한 이유로 프로그램에서 데이터로 사용되는 큰 사이즈의 행렬에 대해 희소화의 필요성이 제기된다.

특히, 패턴인식 분야에서 모델의 파라미터를 희소화하기 위한 연구가 많이 진행되었었다. 대표적인 희소화 방법은 목적함수에 파라미터의 L₁-노름(norm) 항을 제약조건(regularization)으로 추가하는 것이다[1, 9, 10, 11]. 이는 본 논문에서 기초하고 있는 RSR[1] 및 라소(lasso)[12], 엘라스틱 넷 회귀분석(elastic net regression)[10]에서도 사용되고 있다.

1.1. 희소 투영행렬 문제

우리는 선형 차원축소 방법으로 다음 식(1)을 만족하는 투영행렬을 $B=(b_{ij})$ 를 얻을 수 있다.

$$y = B^T x \quad (1)$$

여기서 x 는 고차원 특징 벡터, y 는 저차원 특징 벡터이고, 일반적으로 이 투영행렬 B 는 희소행렬이 아니다. 희소 투영행렬 문제는 다음을 만족하는 행렬 \tilde{B} 을 구하는 문제이다.

- (i) 모든 학습데이터 x 에 대해, $B^T x \approx \tilde{B}^T x$.
- (ii) 행렬 \tilde{B} 의 많은 요소가 0이다.
- (iii) 투영행렬 B 를 \tilde{B} 로 대체했을 때, 알고리즘의 성능저하가 크지 않아야 한다.

본 논문의 연구배경이 되는 얼굴인식의 경우, 기존의 투영행렬 B 를 사용하여 얻어진 인식률보다 인식률 저하가 크지 않는 범위에서 행렬 B 를 대체할 희소행렬 \tilde{B} 을 찾는 문제이다. 이때 인식률과 희소성(sparsity)은 교환거래(trade-off) 관계가 있다.

주어진 투영행렬 B 의 희소 투영행렬을 구하기 위한 학습데이터는 $N \times m$ 고차원 데이터행렬 X 와 $n \times m$ 저차원 데이터 행렬 Y 로 구성된다. 데이터행렬 X 의 k 번째 열은 고차원 학습데이터 $x^k = (x_1^k, x_2^k, \dots, x_N^k)^T$ 이고 Y 의 k 번째 열은 y^k 이 다음 식(2)에 의해 차원 축소된 저차원 벡터 $y^k = (y_1^k, y_2^k, \dots, y_n^k)^T$ 이다.

$$y^k = B^T x^k \quad (2)$$

따라서 희소 투영행렬을 구하는 문제는 x^k 들을 독립변수, y^k 를 종속변수로 하는 회귀분석 문제에 포함된다.

우리는 2절에서 희소성 제약조건이 있는 회귀분석 모델을 소개한다. 본 논문에서 제안하는 RSR 개선 방법론들은 3절에서 기술하고, 4절에서는 실험 결과를 소개하고 분석한다.

2. 회소성 제약이 있는 회귀모델

다중선형회귀분석(multiple linear regression)은 고차원 반응벡터들과 고차원 입력벡터간의 선형적인 인과관계를 설명할 수 있는 행렬을 구할 수 있는 방법론이다. 구해진 행렬은 초평면(hyperplane)의 법선벡터(normal vector)들과 절편들로 이루어진다.

다중회귀분석의 성능향상을 위해서, 또는 사용하고자 하는 목적에 따라 제약조건을 부과함으로써 원하는 선형회귀모델을 얻을 수 있다. 본절에서 소개하는 엘라스틱 넷 회귀분석과 RSS 방법론은 다중회귀분석의 목적함수에 특정 제약조건을 부과하는 방법론이다.

2.1. 엘라스틱 넷 회귀분석

엘라스틱 넷 회귀분석 모델은 Friedman 등 [10]이 제안한 방법론으로 다변량 선형회귀모델에 다음과 같은 목적함수를 가진다.

$$\operatorname{argmin} R_{\lambda}(\beta_0, \beta) \text{ s.t.} \quad (3)$$

$$R_{\lambda}(\beta_0, \beta) = \frac{1}{N} \sum_{i=1}^N (y_i - \beta_0 - x_i^T \beta)^2 + \lambda P_{\alpha}(\beta)$$

여기서 $P_{\alpha}(\beta)$ 는 회귀모델의 파라미터들 β 에 대한 제약조건 또는 페널티 항으로서 다음과 같이 정의된다.

$$P_{\alpha}(\beta) = (1 - \alpha) \frac{1}{2} \|\beta\|_2^2 + \alpha \|\beta\|_1 \quad (4)$$

$$= \sum_{j=1}^p \left[\frac{1}{2} (1 - \alpha) \beta_j^2 + \alpha |\beta_j| \right]$$

여기서 $\alpha = 0$ 이면 리지(ridge) 회귀모델이 얻어지고, $\alpha = 1$ 이면 라소(lasso) 회귀모델이 얻어진다.

2.2 RSR 방법론

RSR 방법론은 Chen 등 [1]이 제안한 회소행렬을 구하는 방법론으로 널리 사용되는 유클리디안(Euclidean) 또는 코사인(cosine)과 같은 거리함수(metric)는 회전변환에 대해 불변하는

(invariant) 성질이 있다는 점에 착안하였다. 데이터의 회전변환 행렬에 대해 자유도를 부여함으로써 정확도를 크게 희생하지 않으면서 회소성을 증진시킨 방법이다. RSR 방법론의 목적함수는 다음과 같다.

$$\min_{B,R} \|R^T Y - B^T X\|_2^2 + \lambda \|B\|_1 \quad (5)$$

여기서 행렬 R 은 $R^T R = I$ 을 만족하는 회전변환(orthogonal transformation) 행렬이다. 행렬 R 의 초깃값을 항등행렬 I 로 설정한 후 다음과 같은 두 과정을 반복함으로써 행렬 R 과 행렬 B 를 교대로 구할 수 있다.

과정 1: 주어진 행렬 R 에 대해 $\tilde{Y} = R^T Y$ 라 하면, 위 목적함수는 다음과 같이 다시 쓸 수 있다.

$$\min_B \|\tilde{Y} - B^T X\|_2^2 + \lambda \|B\|_1 \quad (6)$$

위 식에서 행렬 B 의 각 열은 독립적이기 때문에, 우리는 각 열에 대해 개별적으로 최적화를 수행한다. 즉, 행렬 B 의 i 번째 열은 \tilde{Y} 의 i 번째 행을 종속변수로 하고 데이터 행렬 X 를 독립변수로 하여 라소 회귀분석을 수행하여 구한다.

과정 2: 행렬 B 를 고정시키고 다음을 만족하는 행렬 R 을 구한다.

$$\min_R \|R^T Y - B^T X\|_2^2 \text{ s.t. } R^T R = I \quad (7)$$

행렬 $YX^T B$ 의 SVD(singular value decomposition)를 UDV^T 라 할 때, 이 목적함수의 해 R 은 UV^T 이다.

과정 1은 행렬 R 이 주어졌을 때, 행렬 B 를 구하는 과정이며, 과정 2는 행렬 B 가 주어졌을 때, 행렬 R 을 구하는 과정이다. 이 과정 1과 2를 반복할 때 수렴하는 행렬 B 와 R 중 행렬 B 가 RSR의 해이다. 참고로, 과정 1의 라소 회귀분석에 의해 얻어지는 파라미터에는 기울기 외에 절편이 있다. 따라서 데이터 행렬 X 은 첫 번째 열이 모두 1인 확장 데이터 행렬(augmented

matrix)이어야 한다.

Chen 등[1]은 실험을 통해 대표적인 희소 회귀분석 방법론인 후방 탐욕(backward greedy) 알고리즘[13]과 구조적 희소성(structure sparsity) 알고리즘[14]보다 우월함을 입증하였다.

3. RSR 개선 방법론

이 절에서는 RSR 알고리즘을 개선할 수 있는 엘라스틱 넷 RSR 방법론을 제안한다. 또한 희소성을 높이기 위해 학습데이터의 이상치 제거 및 학습데이터 샘플링 방법을 함께 사용할 것을 제안한다.

3.1. 엘라스틱 넷 RSR

RSR의 목적함수 식(5)에서 제약식 $\lambda \|B\|_1$ 을 식(4)의 엘라스틱 넷 제약식으로 교체하면 다음과 같다.

$$\min_{B,R} \|R^T Y - B^T X\|_2^2 + \lambda P_\alpha(B) \quad (8)$$

이 목적함수를 사용한 회귀분석 방법을 엘라스틱 넷 RSR(E-RSR)이라 부르기로 하자.

Friedman[10] 등은 선형회귀분석에서 데이터의 차원이 학습데이터의 개수보다 크거나 많은 입력변수들에 상관관계 존재할 경우, 제약조건으로 L_1 -노름이나 L_2 -노름보다 $P_\alpha(\beta)$ 을 사용하는 것이 더 좋은 결과를 도출함을 실험적으로 보여준 바 있다. 우리는 본 논문에서 RSR에서도 $P_\alpha(\beta)$ 제약조건을 사용하는 것이 더 좋은 결과를 가져올 수 있음을 실험적으로 보여줄 것이다.

목적함수 식(8)의 해를 구하기 위해서는 파라미터 λ 와 α 를 결정하여야 한다. $\alpha=1$ 인 경우, 위 식은 라소(lasso) 제약식을 가지는 RSR과 동일하게 되어 일반적으로 희소성이 가장 높은 투영행렬을 얻을 수 있다. 반면, 양수 α 가 0에 가까울수록 투영행렬의 희소성이 작아진다.

우리는 실험에서 α 를 고정시키고, K-묶음 교차검증(K-fold cross-validation)을 이용한 1-표준-오차 규칙(one-standard-error rule)[12]에 의해 λ 를 결정하였다. 즉, $CV(\lambda)$ 를 교차검증 에

리라고 할 때, 다음과 같이 최소 에러 추정치 $\hat{\lambda}$ 을 구한다.

$$\hat{\lambda} = \operatorname{argmin}_\lambda CV(\lambda) \quad (9)$$

그리고 λ 를 양의 방향으로 이동시키면서 다음 식을 만족하는 최대 λ 값을 구한다.

$$CV(\lambda) \leq CV(\hat{\lambda}) + SE(\hat{\lambda}) \quad (10)$$

여기서 $SE(\hat{\lambda})$ 은 $\lambda = \hat{\lambda}$ 일 때, K 묶음의 교차검증에러들의 표준편차이다. 이렇게 구해진 λ 는 최소 교차검증 에러의 1-표준오차 안에서 제약식을 최대화한다.

우리는 λ 를 결정하기 위해 사용하는 저차원 데이터의 종류에 따라 다음과 같은 두 가지 알고리즘을 제안한다.

<표 1> 고정 λ E-RSR 알고리즘

```

Learn  $\lambda$ 's for each low dimension using  $X$  and  $Y$ 
R = I
while (not converge) {
    solving B given R using the learned  $\lambda$ 's
    solving R given B
}
    
```

<Table 1> Fixed λ E-RSR algorithm

<표 2> 가변 λ E-RSR 알고리즘

```

R = I
while (not converge) {
    Learning  $\lambda$ 's for each low dimension using  $X$  and  $RY$ 
    solving B given R using the learned  $\lambda$ 's
    solving R given B
}
    
```

<Table 2> Variable λ E-RSR algorithm

위 <표 1>과 <표 2>에서 'solving B given R'은 using the learned λ 's'는 각 차원마다 결정된 n 개의 λ 를 이용하여 엘라스틱 넷 RSR 식(8)의 해를 구하는 것이다.

주어진 학습데이터 행렬을 X , Y 라 할 때, RSR 알고리즘의 반복문 내에서 회전형렬 R 이 변하기 때문에 적합한 저차원 데이터 행렬은

Y 가 아니라 $\tilde{Y}=RY$ 가 된다. <표 1>은 원본 데이터 행렬 (X, Y)을 이용하여 각 차원마다 λ_i ($i=1, \dots, n$)를 결정한 후 RSR 반복시에 λ_i 들을 고정시키는 방법이다. 반면, <표 2>는 반복시 달라지는 저차원 데이터 행렬 \tilde{Y} 에 따라 λ_i 들을 반복적으로 다시 결정하는 방법이다. 참고로, $\alpha=1$ 인 경우 <표 2>의 가변 λ 알고리즘은 기존의 RSR 방법론과 동일하다.

3.2. 학습데이터의 이상치 제거

RSR을 포함하여 행렬의 희소 파라미터를 구하는 대부분의 방법론은 회귀분석에 기반하고 있다. 회귀분석은 이상치(outlier)에 민감하다. 이러한 이유로 많은 통계분석 방법론은 분석 전 데이터의 이상치들을 검출하여 데이터 셋으로부터 이들을 제거한다. 우리도 회귀분석 분석에 기반을 둔 RSR의 성능향상을 위해 다음과 같이 학습데이터 셋으로부터 이상치들을 제거하였다.

k 번째 고차원 학습데이터 x^k 에 대응하는 저차원 학습데이터를 $y^k = (y_1^k, y_2^k, \dots, y_n^k)$ 라 하자.

m_i 와 iqr_i 를 i 번째 차원의 저차원데이터 $y_1^k, y_2^k, \dots, y_n^k$ 의 중앙값(median)과 사분위수(interquartile range)이라 할 때, 다음을 이상치를 판단하기 위한 조건[15]으로 설정하였다.

$$y_i^k < m_i - 1.5iqr_i, y_i^k > m_i + 1.5iqr_i \quad (11)$$

즉, 위 조건을 만족하는 i 가 존재하는 k 번째 학습데이터의 쌍 x^k 와 y^k 는 학습데이터에서 제외한다.

4절의 실험결과 <표 4>는 이상치를 제외시킴으로써 인식 성능이 향상시킬 수 있음을 보여주고 있다. 즉, 이 방법은 일반화 성능을 높이는 데 기여한다고 할 수 있다.

3.3. 학습데이터 샘플링

투영행렬의 희소성을 높이는 방법으로 샘플링(sampling) 방식을 제안한다. 이 방법은 랜덤으로 학습데이터 중 일부를 선택하여 선택된 데이터만 사용하여 희소 투영행렬을 구하는 것이다. 일반적으로 RSR 등 파라미터의 L_1 -노름을 제약식에 추가하면 희소 파라미터를 구할 수 있다.

우리는 4절의 (그림 2), <표 5>에서 샘플링 방법으로 더 높은 희소성을 얻을 수 있음을 실험을 통해 입증하였다.

행렬의 희소율(sparsity rate)은 행렬 원소 중 0의 개수를 행렬의 크기(dimension)로 나눈 수로 정의한다. 예를 들어, 희소율이 0.7이라 함은 행렬의 원소 100개당 70개가 0이고 30개가 0이 아님을 의미한다. 행렬의 희소율은 0과 1사이의 값을 가지며 희소성이 크다는 말은 희소율이 크다는 말과 같다.

4. 실험 및 결과

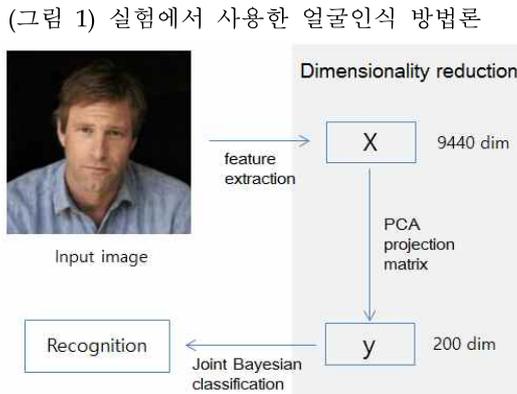
우리는 이절에서 얼굴인식 알고리즘에서 사용되는 투영행렬을 희소화하는 실험을 수행하였다. 파라미터 α 의 값에 따라 제안한 방법론의 희소율과 인식률을 측정하고, 기존 RSR[1] 방법론과 비교실험도 수행하였다. 파라미터 λ 는 5-몸음 교차검증으로 3.1 절에서 소개한 1-표준-오차 규칙에 의해 결정하였다.

우리는 실험에서 최근 이슈가 되고 있는 LFW(Labeled Faces in the Wild)[16]와 LFW-a[17] DB를 사용하였다. LFW DB는 5,749 인물에 대해 13,233개의 영상을 포함하고 있다. LFW-a의 영상들은 LFW DB의 모든 영상들을 상업용 소프트웨어를 이용하여 얼굴 배열을 수행한 영상들의 모음이다.

LFW DB에는 실험 규약이 정의되어 있다. 10개의 폴드(fold)에 영상의 쌍(pair)들이 정해져 있다. 이 중 한 폴드를 테스트 폴드로 정하고, 나머지 9개 폴드를 학습 폴드로 정한다. 학습 폴드에 속하는 모든 영상을 학습 과정에서 이용할 수 있다. 학습 후, 테스트 폴드에 있는 각각의 영상 쌍에 대해 두 장에 있는 얼굴이 동일 인물의 것인지 여부를 맞히는 것이다. 이 과정을 테스트 폴드를 바꿔가며 10회 반복하고 평균 인식률을 제시한다.

우리는 실험에서 첫 번째 폴드를 제외한 나머지 폴드의 영상들을 학습하고, 첫 번째 폴드에 있는 300쌍의 같은 인물사진과 300쌍의 다른 인물 사진에 대해 ‘같다’, ‘다르다’를 판단하여 인식률을 측정하였다. 각 영상에 대해 9440차원의 LBP 히스토그램 특징벡터[3]를 추출한 후,

PCA에 의해 200차원 벡터로 축소한 후, Joint Bayesian 방법론[2]을 이용하여 인식하였다. (그림 1)의 도식은 실험에서 사용한 얼굴인식 방법론을 보여 준다. 우리는 여기서 차원 축소를 위해 사용한 9440×200 투영 행렬을 제안한 방법론을 이용하여 얻어진 9440×200 희소 행렬로 대체하고 인식률을 측정하였다.



(그림 1) 실험에서 사용한 얼굴인식 방법론

(Figure 1) Face recognition method for experiments

모든 학습데이터를 이용하여 PCA를 통해 구한 투영행렬을 이용하였을 때 인식률은 86.17%이다. 우리는 PCA 투영행렬을 대신할 희소행렬을 구하고자 한다. 이 희소행렬은 희소성이 높으면서 86.17%에 근접하는 인식률을 산출할 수 있어야 한다. 우리는 실험을 통해 다음과 같은 물음에 답하고자 한다.

- 제안한 샘플링과 이상치 제거 방법은 인식률을 크게 저하시키지 않으면서 희소성을 높일 수 있는가?
- 제안한 방법 엘라스틱 넷 방법론이 RSR 방법론 보다 우수한가?

4.1. 실험 결과 및 분석

RSR[1]은 반복문(loop)을 돌며 반복마다 희소행렬을 생성한다. RSR 알고리즘의 수렴성은 반복 횟수가 증가함에 따라 이전 희소행렬과 새로 구해진 희소행렬의 차(matrix difference norm)가 점차 줄어들음을 통해 확인할 수 있었다. <표 3>은 반복(iteration)마다 구해진 희소행렬을 이용하여 인식률을 측정한 결과와 행렬의 희소율

의 변화를 보여준다. 우리는 많은 실험을 통해 희소율은 반복회수가 증가함에 따라 대체로 점진적으로 증가(monotone increase)하는 경향을 보임을 확인할 수 있었다. 하지만 인식률은 오르내리며 동요하는 경향을 보였다.

<표 3> 반복에 의한 RSR의 희소율과 인식률의 변화

| Iteration | sparsity rate (%) | recongition rate (%) |
|-----------|-------------------|----------------------|
| 1 | 42.64 | 86.67 |
| 2 | 42.58 | 85.33 |
| 3 | 42.52 | 84.83 |
| 4 | 42.70 | 85.17 |
| 5 | 42.60 | 85.33 |
| 6 | 42.76 | 85.33 |
| 7 | 42.70 | 85.33 |
| 8 | 42.68 | 85.33 |
| 9 | 42.72 | 86.17 |

<Table 3> Sparsity and recognition rates of RSR during iteration

<표 3>에서 반복회수 1에서 구해진 희소행렬은 원소의 42.64%가 0이고 이 희소행렬을 이용한 인식률은 86.67%이다. 참고로, 86.67%는 목표 인식률인 86.17%보다도 높은 수치이다. 기존 모델의 파라미터를 희소화시키는 다른 알고리즘들에서도 목표 인식률을 약간 상회하는 경우가 종종 있다.[9]

우리는 희소행렬 생성시 이상치 제거 효과를 알아보기 위해, 동일한 파라미터를 사용할 때, 이상치를 제외하지 않고 모든 학습데이터를 이용한 경우와 이상치가 아닌 학습데이터만 이용한 경우에 대해 희소율과 인식률을 비교해 보았다. <표 4>는 $\alpha = 1$, 샘플링 비율이 0.2일 때의 실험결과이다. 학습데이터로부터 이상치를 제외시키고 희소행렬을 구하는 경우 더 높은 인식률을 보이는 것을 볼 수 있다.

<표 4> 반복에 의한 RSR의 희소율과 인식률의 변화

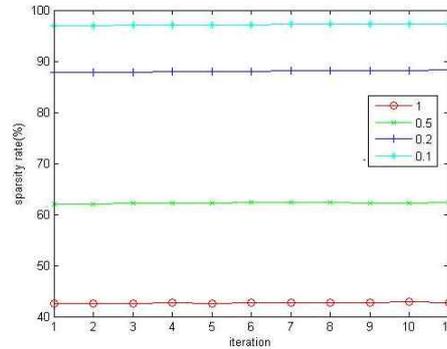
| Iteratio n | Outlier removal(%) | | All training data(%) | |
|---------------|--------------------|----------------------|----------------------|----------------------|
| | sparsity rate | recogniti on rate | sparsity rate | recogniti on rate |
| 1 | 78.10 | 85.00 | 85.82 | 84.67 |
| 2 | 77.25 | 85.33 | 85.76 | 84.50 |
| 3 | 76.44 | 85.17 | 85.75 | 82.83 |
| 4 | 76.51 | 84.67 | 85.69 | 84.00 |
| 5 | 77.06 | 86.00 | 85.54 | 83.33 |
| 6 | 77.74 | 85.67 | 86.06 | 83.67 |
| 7 | 76.35 | 86.33 | 85.58 | 83.33 |
| 8 | 76.30 | 85.83 | 86.25 | 84.33 |
| 9 | 76.07 | 86.50 | 86.05 | 84.33 |

<Table 4> Changes of sparsity and recognition rates of RSR during iteration

우리는 이 인식률의 차이가 통계적으로 유의한지 알아보기 위해 단측 t-검정(t-test)을 수행하였다. 그 결과 p-값이 1.2×10^{-5} 으로, 이상치를 제거했을 때 인식률이 높음을 통계적으로 확인할 수 있었다. α 와 샘플링 비율을 달리 하여 실험했을 때에도 모든 실험에서 이상치를 제거했을 때 더 높은 인식률을 보였으나, 희소율은 이상치를 제거하지 않았을 때 더 높았다.

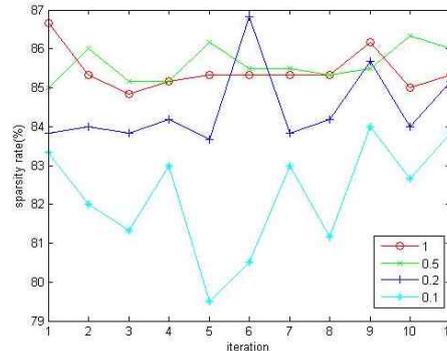
우리는 샘플링과 희소성의 관계를 알아보기 위해 파라미터 α 를 고정하고 샘플링 비율을 변화시켜 보았다. (그림 2)와 (그림 3)은 $\alpha = 1$ 일 때 샘플링 비율을 1, 0.5, 0.2, 0.1로 변화시켰을 때 결과를 보여 준다. 여기서 x축은 루프 반복회수이다. 샘플링 비율이 낮아짐에 따라 희소성은 높아지고, 샘플링 비율이 0.1인 경우를 제외하고는 인식률 저하가 크지 않음을 볼 수 있다. 샘플링 비율이 0.1이라는 것은 학습데이터 10개당 한 개만 취하는 경우이다. 샘플 수가 너무 적은 경우, 샘플의 분포와 학습데이터의 분포의 유사성이 떨어져서 인식률이 다소 떨어지는 것으로 보인다.

(그림 2) 샘플링 비율에 따른 희소율의 변화



(Figure 2) Change of sparsity rates according to sampling rate

(그림 3) 샘플링 비율에 따른 인식률의 변화



(Figure 3) Change of recognition rates according to sampling rate

마지막으로 <표 1>과 <표 2>의 제안한 E-RSR 방법론과 RSR[1]의 성능을 비교해 보았다. 인식률과 희소율 사이에는 교환거래가 있기 때문에 이들의 직접적인 수치 비교는 어렵다. 인식률과 희소율은 교환거래(trade-off) 관계가 있다. 희소율이 지나치게 높은 경우는 좋은 인식률을 얻기 어렵다. 좋은 희소행렬을 얻기 위해서는 인식률의 저하의 한계(tolerance)를 먼저 정해 놓고 희소율이 낮은 행렬을 구하는 것이 바람직하다.

<표 5>는 각 방법론들의 최고 인식률과 그때의 희소율을 제시하였다. 고정 λ 알고리즘은 $\alpha = 1$ 이고 샘플링 비율이 0.2일 때 최고 인식률을 보였고, 가변 λ 알고리즘은 $\alpha = 0.9$ 이고 샘플링 비율이 0.2일 때 최고 인식률을 보였다.

<표 5> 제안한 알고리즘과 기존 RSR 비교

| | Sparsity rate | Recognition rate |
|--------------------------|---------------|------------------|
| RSR | 42.64 % | 86.67 % |
| fixed λ E-RSR | 88.06 % | 86.83 % |
| variable λ E-RSR | 82.54 % | 86.67 % |

<Table 5> Comparison of the proposed algorithms and original RSR

<표 5>에서 보듯이 세 방법론 사이에 유의한 인식률의 차이는 보이지 않는다. 세 방법론 모두 목표 인식률이었던 86.17%를 초과하였다. 일반적으로 희소행렬을 통한 인식률 향상은 기대하기 어렵다. 하지만, 희소율은 많은 차이를 보인다. 기존의 RSR보다 제안한 방법론들이 유의하게 더 높은 희소율을 보였다. 희소율과 인식률 측면에 제안한 두 방법론은 큰 차이가 없다.

4.2. 희소행렬의 저장 용량

우리는 저장용량을 최소화하기 위한 많은 희소행렬 표현방식 중 CSR(Compressed Sparse Row)방식[18]을 이용하여 구해진 희소행렬을 저장하였다. 라소 또는 엘라스틱 넷 회귀로 구한 회귀행렬은 절편 벡터 b 를 갖는다. 따라서 식(1)과 같은 형태가 아닌 다음과 같은 방식으로 투영하여야 한다.

$$y = B^T x + b \quad (12)$$

<표 6>은 CSR 표현을 이용한 희소행렬에 의해 고차원 벡터 x 를 차원축소하여 저차원 벡터 y 에 저장하는 의사코드를 보여 준다.

<표 6> CSR 방식 희소 투영행렬을 이용한 투영 의사 코드

```

for( i=0 ; i<M; i++) {
  y[i] = b[i];
  for(j=row_start[i]; j<row_start[i+1]; j++)
    y[i] = y[i] + values[j]*x[col_idx[j]]
}

```

<Table 6> Projection pseudo code using sparse matrix via CSR representation

희소율이 88.06%인 9440×200 희소행렬은 CSR 표현에 의해 int 또는 float 타입의 배열 row_start, col_idx, values에 저장되며, 절편은 float 형 배열에 저장된다. 이 네 배열의 총용량은 201×4 + 225,427×4 + 225,427×4 + 200×4 = 1,805,020 바이트이다. 같은 사이즈의 PCA 투영행렬의 저장 용량이 9440×200×4 = 7,552,000이므로 희소율이 88.06%인 희소행렬의 실제 저장용량은 11.94%가 아닌 23.90%이다.

5. 결론 및 토의

우리는 본 논문에서 E-RSR, 이상치 제거, 샘플링 등 RSR 방법론을 개선할 수 있는 방법론들을 제안하였다. 실험을 통해 학습데이터로부터 이상치를 제거함으로써 인식률을 향상시킬 수 있음을 보였으며, 샘플링을 통해 희소율을 크게 향상시킬 수 있음을 보였다. 제안한 고정 λ E-RSR과 가변 λ E-RSR 방법은 모두 기존의 RSR과 인식률 측면에서 차이가 없었고, 고정 λ E-RSR의 희소율이 근소한 차이로 가장 높게 나타났다. 수렴속도도 계산량이 적은 고정 λ E-RSR이 가장 빨랐다. 참고로, 학습데이터가 많을 때, 희소행렬을 구하는 알고리즘의 수행시간은 매우 길다. 4절의 실험에서 RSR이 수렴할 때까지 이틀 정도의 시간이 걸린다.

이론적으로, RSR 방식이 반복문 내에서 회전 변환행렬에 의해 저차원 학습데이터가 조금씩 변하기 때문에 가변 λ E-RSR 방법이 더 합리적으로 보인다. 하지만, 여러 논의와 실험결과를 종합해 볼 때, 고정 λ E-RSR 방법이 가장 우수하다고 볼 수 있다.

References

- [1] D. Chen, X. Cao, F. Wen, and J. Sun, "Blessing of Dimensionality: High-dimensional Feature and Its Efficient Compression for Face Verification", Proc. CVPR, pp.3025-3032, June. 2013.
- [2] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun, "Bayesian Face Revisited: A Joint Formulation," Proc. EC CV, pp.566 - 579, 2012.

- [3] S. Han and J.-H. Ahn, "An Improved Joint Bayesian Method using Mirror Image's Features", Journal of Digital Contents Society, in press, 2015.
- [4] Y. Sun, D. Liang, X. Wang and X. Tang, "DeepID3: Face Recognition with Very Deep Neural Networks", arXiv:1502.00873v1, 2015.
- [5] B. Scholkopf, A. Smola, and K.-R. Muller, "Nonlinear Component Analysis as Kernel Eigenvalue Problem", Neural Computation, Vol.10, No.5, pp. 1299-1319, 1998.
- [6] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K.-R. Muller, "Fisher Discriminant Analysis with Kernels", Proceedings of the IEEE Signal Processing Society Workshop, pp. 41-48, 1999.
- [7] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Network", Science, Vol. 313, pp. 504-507, 2006.
- [8] C. M. Bishop, Pattern Recognition and Machine Learning, 1st ed., Springer, 2006.
- [9] J. Bi, K. Bennett, M. Embrechts, C. Breneman, and M. Song, "Dimensionality Reduction via Sparse Support Vector Machine", Journal of Machine Learning Research, Vol.3, pp. 1229-1243, 2003.
- [10] J. Friedman, T. Hastie and R. Tibshirani, "Regularization Paths for Generalized Linear Model via Coordinate Descent", Journal of Statistical Software, Vol.33, No.1, pp.1-22, 2010.
- [11] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust Face Recognition via Sparse Representation", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 31, No. 2, pp. 210-227, 2009.
- [12] T. Hastie, R. Tibshirani, and J. Friedman, "The Elements of Statistical Learning: Prediction, Inference and Data Mining", 2nd Edition, Springer-Verlag, New York, 2009.
- [13] T. Zhang, "Adaptive forward-backward greedy algorithm for learning space representations", Information Theory. Vol. 57, No.7, pp. 4689-4708, 2011.
- [14] J. Liu, S. Ji, and J. Ye, "Multi-task feature learning via efficient L2, 1-norm minimization", In Conference on Uncertainty in Artificial Intelligence, pp. 339-348, 2009.
- [15] R. Mason, R. Gunst, and J. Hess, "Statistical design and analysis of experiments: with applications to engineering and science", 1st Edition, Wiley, 1989.
- [16] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments", University of Massachusetts, Amherst, Technical Report 07-49, Oct. 2007.
- [17] L. Wolf, T. Hassner, and Y. Taigman, "Effective Face Recognition by Combining Multiple Descriptors and Learned Background Statistics", IEEE T. PAMI, Vol.33, No.10, 2011.
- [18] Y. Saad, "Iterative Methods for Sparse Linear Systems", 2nd Edition, SIAM, 2003.



안 정 호

1998년 : 연세대학교 대학원 (공학 석사-수학)

2001년 : Texas A&M University 대학원 (이학석사-통계학)

2006년 : 연세대학교 대학원 (공학 박사-컴퓨터과학)

2006년~2007년: 연세대학교 BK21 지능형모바일서비스를 위한 차세대 단말소프트웨어 사업단, 박사후 연구원

2006년~현재: 강남대학교 컴퓨터미디어정보공학부, 부교수

관심분야 : 얼굴인식, 패턴인식, 기계학습, 딥러닝, 영상처리, 컴퓨터비전 등