**고급소프트웨어실습1 - 8주차 과제**

PCA에서 압축을 위한 최적 eigenvector 개수 구하기

|  |  |  |
| --- | --- | --- |
| **분반** | **:** | 3 |
| **학번** | **:** | 20121635 |
| **이름** | **:** | 장종석 |

1. **Eigenvector의 개수는 원래 데이터의 차원(dimension)과 같으므로 모든 eigenvetor를 이용하여 원래 데이터를 투영하면 데이터 압축의 효과를 볼 수 없다. 또한 데이터의 분산 특성을 고려하지 않고 특정 몇 개의 eigenvector를 이용하여 투영시키면 데이터의 유용한 정보가 손실될 수 있다. 유용한 정보를 손실하지 않으면서 최대로 압축 효과를 얻을 수 있도록 eigenvector의 개수를 설정할 수 있는 방법이 있는 지 설명해 보자.**

PCA는 데이터의 차원을 낮춤으로써, 데이터를 압축하는 효과를 나타낸다. 그러나, 차원이 낮아지면서 원 데이터에 비해 데이터 손실이 발생한다. 압축에 사용할 Eigenvector의 수를 늘리게 되면, 원 데이터로의 회기능력이 좋아지지만, 그만큼 많은 데이터를 저장하고, 작업하기 때문에 작업량이 많아지게 된다. 하지만, 이러한 문제를 해결하기 위해 Eigenvector의 수를 줄이게 되면, 데이터 손상률이 커지게 된다. 두 가지 문제를 해결하기 위해서 Eigenvector선정하는 방법이 중요하다.

일반적으로, Eigenvalue값이 큰 Eigenvector의 경우, 데이터들 사이에 변화량이 크다는 것을 의미하기 때문에, 데이터 복구에 유용한 정보를 갖고, 반대로 Eigenvalue값이 작은 Eigenvector의 경우 작은 노이즈 정도로 취급하여, 압축할 때 무시할 수 있다.

일반적인 경우라면 상위 Eigenvector들을 선택하되, Eigenvector들의 Eigenvalue들도 어떠한 분포를 이룰 것이기 때문에 이 분포를 파악하여 데이터 복구에 그다지 도움이 되지 않는 부분에 위치한 Eigenvector들은 제외하는 방법을 이용해야 한다. 그 기준은 구현하기 나름이지만 Eigenvalue들의 평균보다 낮은 값의 Eigenvalue를 갖는 Eigenvector들을 제외하거나 원래 데이터의 차원수의 특정 상위 몇 퍼센트만큼의 Eigenvector를 선택하거나 원래 데이터의 차원수의 제곱근만큼의 Eigenvector를 선택하는 등등의 방법이 있다.