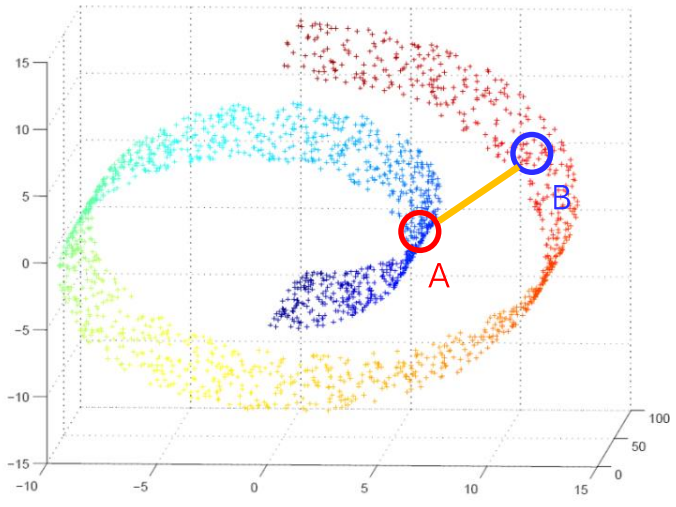
Dimensionality Reduction

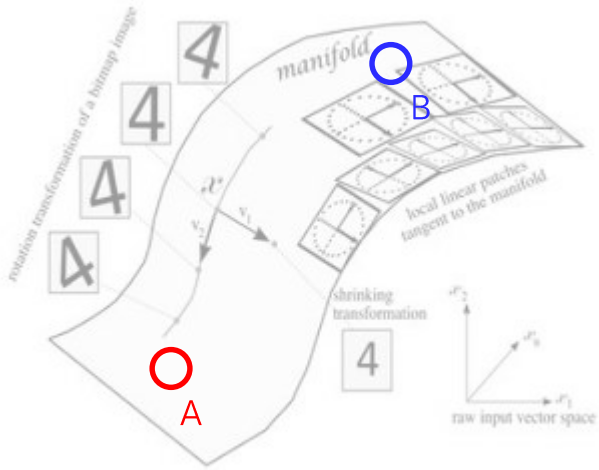
❖ Manifold란?

- 매니폴드란 데이터가 있는 공간이며 이 공간 상에는 각 데이터 포인트들이 존재함
- 두 점 사이의 거리 혹은 유사도가 근거리에서는 Euclidean 거리인 직선거리를 따르지만 원거리에서는 그렇지 않은 공간을 말함
- 고차원 상에서는 가까운 데이터를 나타내보면 실제로는 의미상으로 가깝지 않을 수 있음
- 그러나 저차원을 기준으로 가까운 데이터를 보면 의미상으로 보다 가까움을 알 수 있음



고차원 공간

2차원 manifold로 변환

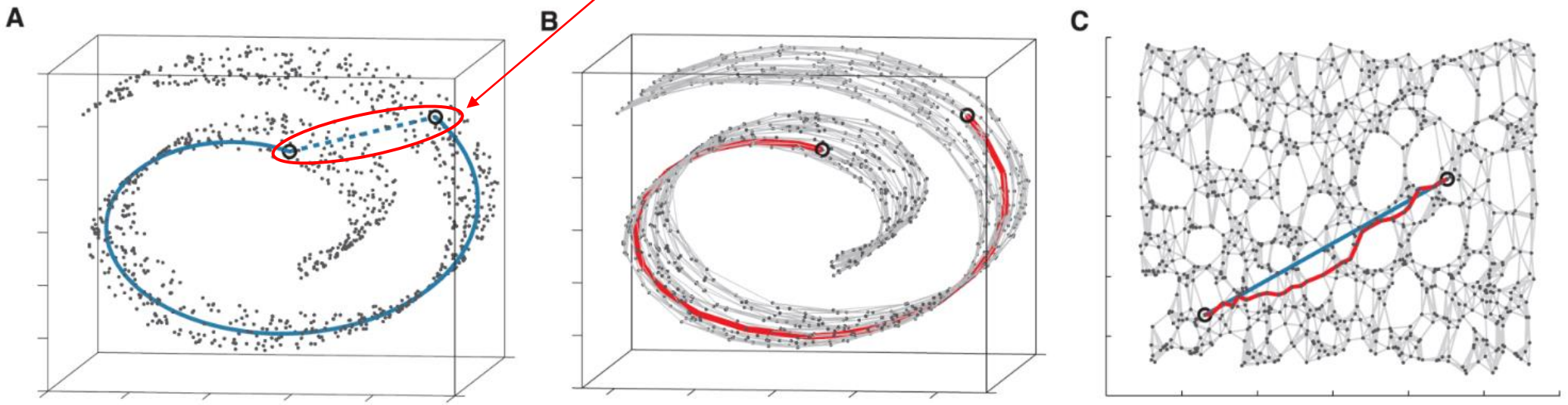


저차원 공간

Dimensionality Reduction

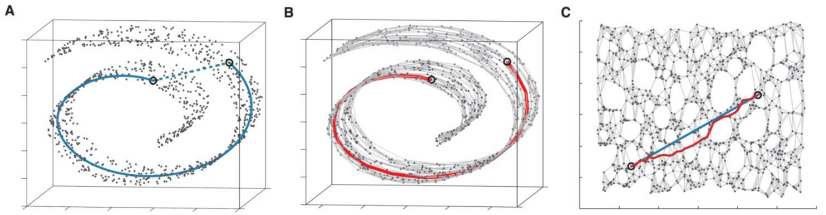
❖ ISOMAP

- 다차원 스케일링(MDS)과 주성분 분석(PCA)의 확장이자 두 방법론을 결합한 방법론임
- MDS 혹은 PCA 같은 선형 방법론을 사용하면 공간상의 Euclidean 거리 등을 구하게 됨
- 그러나 데이터가 구조적 혹은 기하학적으로 선형이 아닐 경우 비선형 변환이 필요함
- ISOMAP은 MDS에서 distance matrix를 변환하는 부분만 바뀜



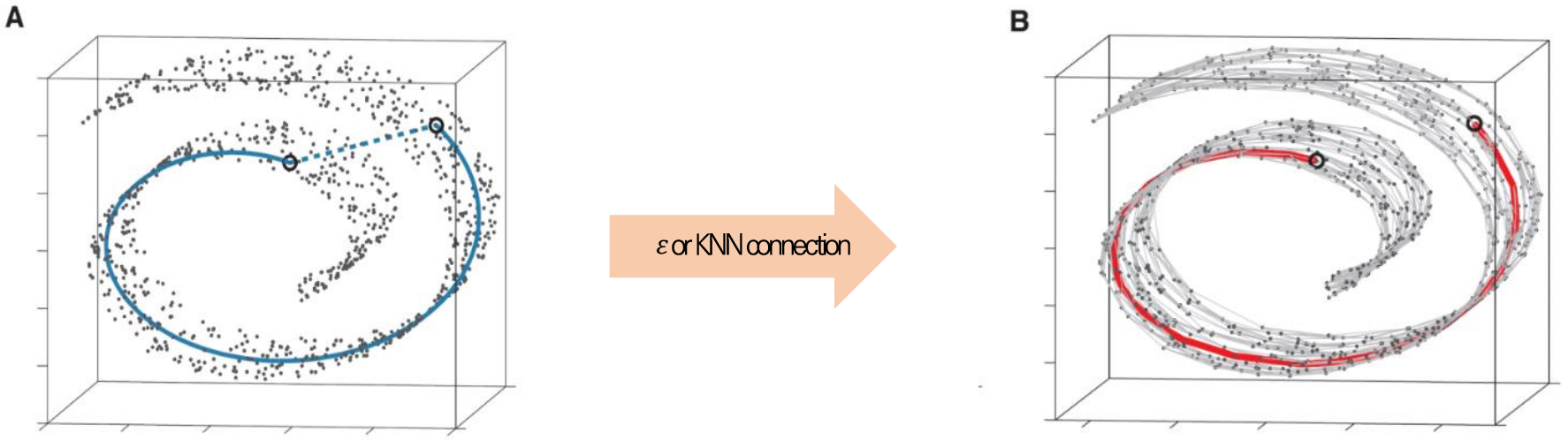
〈Swiss Roll Example〉

Dimensionality Reduction



❖ ISOMAP – step1: Neighborhood graph 구축

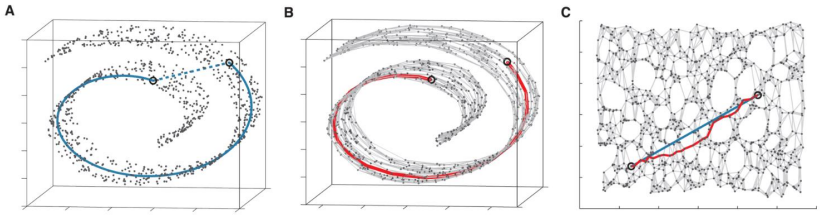
- 첫번째 단계에서는 고정된 기준값 ε 또는 KNN 방식을 통해서 모든 점들을 서로 연결함
- Original manifold의 점들이 모두 연결되면 인접한 이웃의 관계가 연결된 그래프가 구축됨
- 점끼리 연결되었을 때 그래프 엣지들의 가중치는 두 연결된 점 사이의 Euclidean 거리가 됨



〈Original Manifold〉

〈Neighborhood graph〉

Dimensionality Reduction



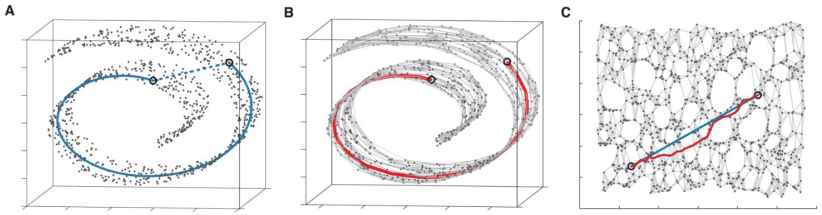
❖ ISOMAP – step2: 두 점 간의 shortest paths 계산

- 두 점 i와 j에 대해 서로 연결되어 있으면 $d_G(i,j) = d_X(i,j)$
- 두 점이 서로 연결되어 있지 않으면 $d_G(i,j)$를 계속 초기화함
- 1부터 N개까지의 k에 대해서 점 i와 j간의 최단 거리인 $d_G(i,j)$를 $\min(d_G(i,j), d_g(i,k) + d_G(k,i))$로 변환
- Step1과 step2를 통해서 두 점 사이의 manifold상 실제 도달 가능거리를 구하는 것임

〈Original Manifold〉

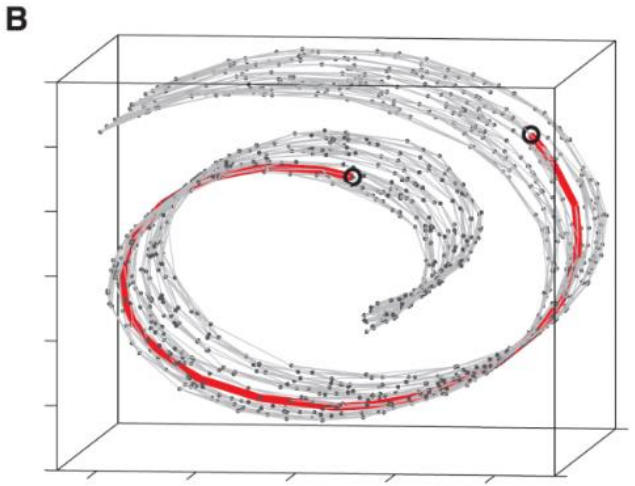
〈Neighborhood graph〉

Dimensionality Reduction



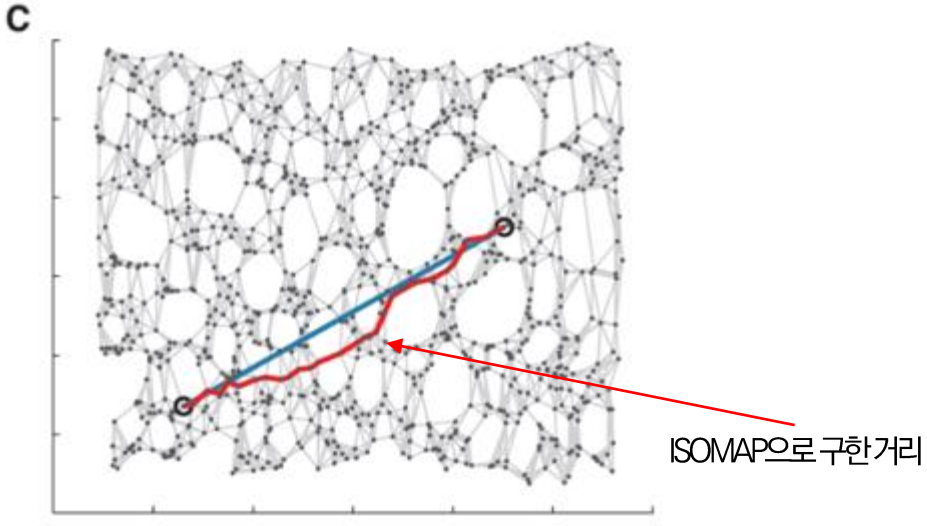
❖ ISOMAP – step3: d차원 임베딩 공간 구축

- Distance matrix를 변환한 후에 기존의 MDS를 그대로 적용하여 임베딩을 구축함
- 두 노드 사이의 최단 경로를 이루는 노드의 수인 geodesic distance를 유지하면서 차원을 축소함
- MDS와 PCA 등 선형방법론의 단점을 보완하면서 데이터 포인트들 사이의 정보를 보존할 수 있음
- 그러나 swiss-roll과 같은 특정한 manifold구성을 보이는 데이터셋에 적절한 방법론이라는 한계점 존재



〈Neighborhood graph〉

MDS를 통해 embedding



〈d-dimensional embedding〉

Dimensionality Reduction

❖ ISOMAP example: Hand digit recognition

- sklearn 패키지에서 제공하는 digits 데이터 셋이며 각 이미지는 64차원으로 이루어진 손글씨 데이터셋
- MDS 방법론과 ISOMAP을 적용했을 때 embedding 비교

A selection from the 64-dimensional digits dataset