15기 정규세션

ToBig's 14기 박준영

차원축소 추가자료

Appendix

질문 목록

Q1) T-SNE의 원리가 궁금해요 !

Q2) SNE의 Crowding Problem에 대해서 설명해주세요.

Q3) PCA+T-SNE를 같이 쓰면 어떻게 되는지 궁금해요!

Q4) 공분산 행렬 대신 상관계수 행렬을 쓸 수 있나요?

Q5) PCA 표준화 할 때 스케일러를 사용하면 어떤 차이가 있나요???

Appendix

Q1) T-SNE의 원리가 궁금해요 !

- Stochastic Neighbor Embedding(SNE)
  - 고차원의 원공간에 존재하는 데이터 x의 이웃 간의 거리를 최대한 보존하는 저차원의 y를 학습하는 방법론
  - Stochastic이란 이름이 붙은 이유는 거리정보를 '확률적'으로 나타내기 때문!

$$p_{ij} = \frac{\exp\left(-\|x_i - x_j\|^2/2\sigma^2\right)}{\sum_{k\neq l}\exp\left(-\|x_k - x_l\|^2/2\sigma^2\right)},$$

<- 첫번째 식의 p는 고차원 원공간에 존재하는 i번째 개체 xi가 주어졌을 때 j번째 이웃인 xj가 선택될 확률

$$q_{ij} = \frac{\exp\left(-\|y_i - y_j\|^2\right)}{\sum_{k\neq l}\exp\left(-\|y_k - y_l\|^2\right)},$$

<- 두번째 식의 q는 저차원에 임베딩된 i번째 개체 yi가 주어졌을 때 j번째 이웃인 yj가 주어졌을 때 j번째 이웃인 yj가 선택될 확률

  - SNE의 목적은 위의 p와 q의 분포 차이가 최대한 작게끔 하고자 한다
  - 차원축소가 제대로 이뤄졌다면 고차원 공간에서 이웃으로 뽑힐 확률과 저차원 공간에서 선택될 확률이 비슷할 것이기 때문에!

Appendix

Q1) T-SNE의 원리가 궁금해요 !

- Stochastic Neighbor Embedding(SNE)
  - 두 확률분포가 얼마나 비슷한지 측정하는 지표 -> Kullback-Leibler divergence 사용
  - 두 분포가 완전히 다르면 1, 동일하면 0의 값을 가짐

$$Cost = \sum_i KL(P_i||Q_i)$$
$$= \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}$$

  - SNE는 위의 비용함수를 최소화하는 방향으로 학습 진행

Appendix

Q1) T-SNE의 원리가 궁금해요 !

- Stochastic Neighbor Embedding(SNE)의
  - 하지만, 여기서

  (p와 q를 정의할 때 등장했던) σ_i = 각 개체마다 데이터 밀도가 달라서 이웃으로 뽑힐 확률이 왜곡되는 현상을 방지하기 위한 값

  - 이 값은 고정된 값을 써도 성능에 큰 차이를 보이지 않아서 이 계산을 생략하고 새로 비용함수를 쓰면,

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2}, \quad q_{ij} = \frac{q_{j|i} + q_{i|j}}{2}$$

$$Cost = \sum_i KL(P_i||Q_i) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

$$\frac{\partial C}{\partial y_i} = 4 \sum_j (y_j - y_i)(p_{ij} - q_{ij})$$

- 우리가 최종적으로 구하고자 하는 미지수는 저차원에 임베딩된 자표관 yi이고, SNE는 gradient descent 방식으로 yi를 업데이트 함!

Appendix

Q1) T-SNE의 원리가 궁금해요 !

- T- Stochastic Neighbor Embedding
  - SNE를 전제하는 확률분포는 가우시안 분포이지만,
  - (강의에서 살펴봤듯이) 가우시안 분포는 꼬리가 두텁지 않아서 i번째 개체에서 적당히 떨어져 있는 이웃 j와 아주 많이 떨어져 있는 이웃 k가 선택될 확률이 크게 차이가 나지 않게 됨! => Crowding problem
  - 위의 문제를 해결하기 위해 가우시안 분포보다 꼬리가 두터운 t-분포를 쓴 것이 바로 t-SNE!
  - T-SNE는 q값에만 아래와 같이 t분포를 적용하고 p값은 SNE와동일

$$q_{ij} = \frac{\left(1 + \left|y_i - y_j\right|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \left|y_k - y_l\right|^2\right)^{-1}}$$

Appendix

Q2) SNE의 Crowding Problem에 대해서 설명해주세요.

$$Cost = \sum_i KL(P_i||Q_i)$$
$$= \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}$$

비용함수에서 저차원 공간에서 두 점이 함께 있을 확률 (q)는 분모에 있다.

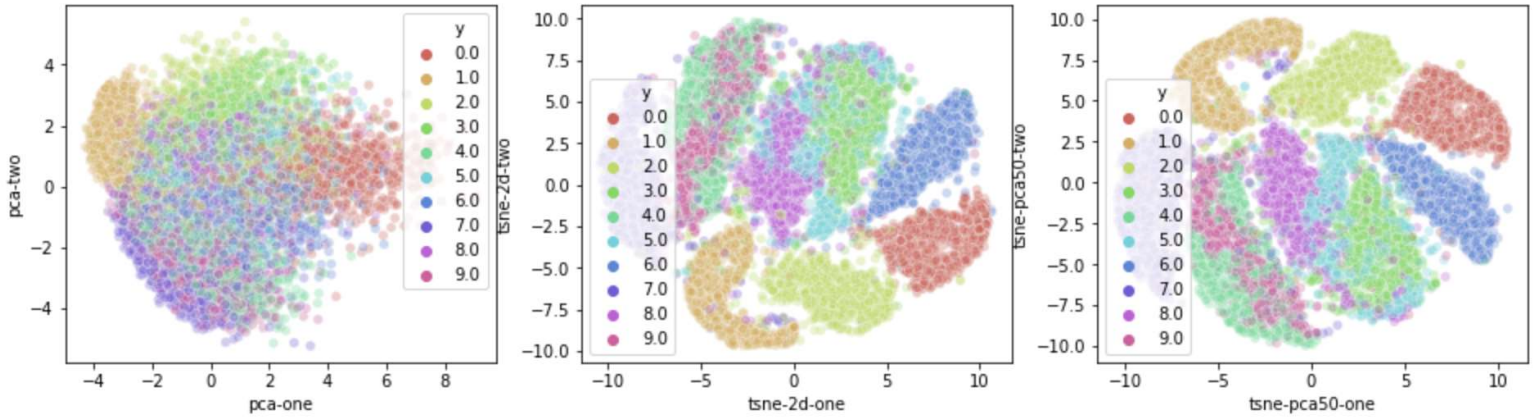
만약 q 확률이 1에 가까워지면 q와 관련하여 손실이 최소화되며 이것이 crowding problem이 된다.

만약 이상치를 생각해보면 p가 0에 매우 가깝고 q가 높다면 다면 비용함수가 매우 낮으므로 SNE가 이들을 분리하도록 학습이 이루어지지 않는다.

https://medium.com/@layog/i-do-not-understand-t-sne-part-2-b2f997d177e3

Appendix

Q3) PCA+T-SNE를 같이 쓰면 어떻게 되는지 궁금해요!



위 그래프는 mnist 파일을 PCA만 수행했을 때 T-sne만 수행했을 때 PCA+T-sne했을 때의 그래프입니다.

PCA를 통해 데이터의 구조를 유지하면서 t-sne를 해도 t-sne만 했을 때의 그래프와 차이가 별로 없는 것을 볼 수 있습니다.

https://towardsdatascience.com/visualising-high-dimensional-datasets-using-pca-and-t-sne-in-python-8ef87e7915b

Appendix

Q4) 공분산 행렬 대신 상관계수 행렬을 쓸 수 있나요?

공분산 행렬 대신 상관계수 행렬을 스펙트럼 분해할 경우 단위 변화에 불변하고 변수 간 측정 단위 차이가 크면 상관계수 행렬을 사용하여 스펙트럼 분해를 수행하여 PCA를 사용할 수 있습니다.

http://contents2.kocw.or.kr/KOCW/document/2016/kunsan/jungkangmo/8.pdf

Appendix

Q5) PCA 표준화 할 때 스케일러 간 어떤 차이가 있나요???

PCA 표준화할 땐 평균을 0을 만들고 분산을 1로 만들어야 하는데

Nomalizer엔 zero-mean이 없고

Min-Max scaler는 단위 분산이 없고

Robust는 이상치에도 동작하기때문에 StandardScaler를 사용한다고 합니다.

https://stackoverflow.com/questions/37221425/which-feature-scaling-method-to-use-before-pca

Appendix

- 참고자료

#논문

Visualizing Data using t-SNE

https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf

https://ratsgo.github.io/machine%20learning/2017/04/28/tSNE/

들어주셔서 감사합니다.