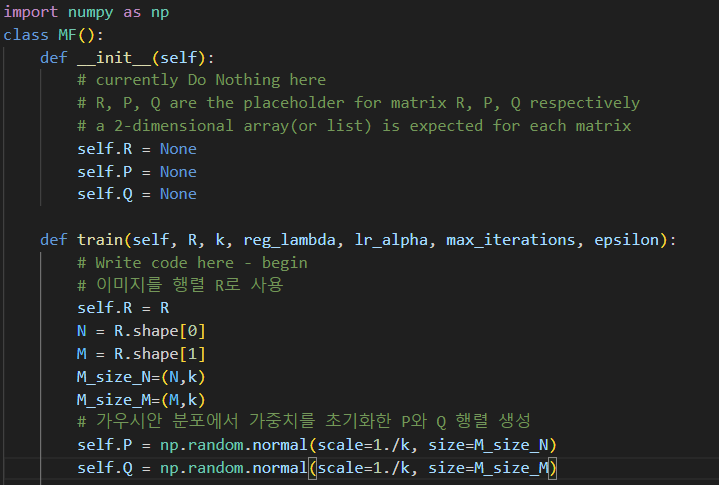
**REPORT**

Deep Learning 과제#2

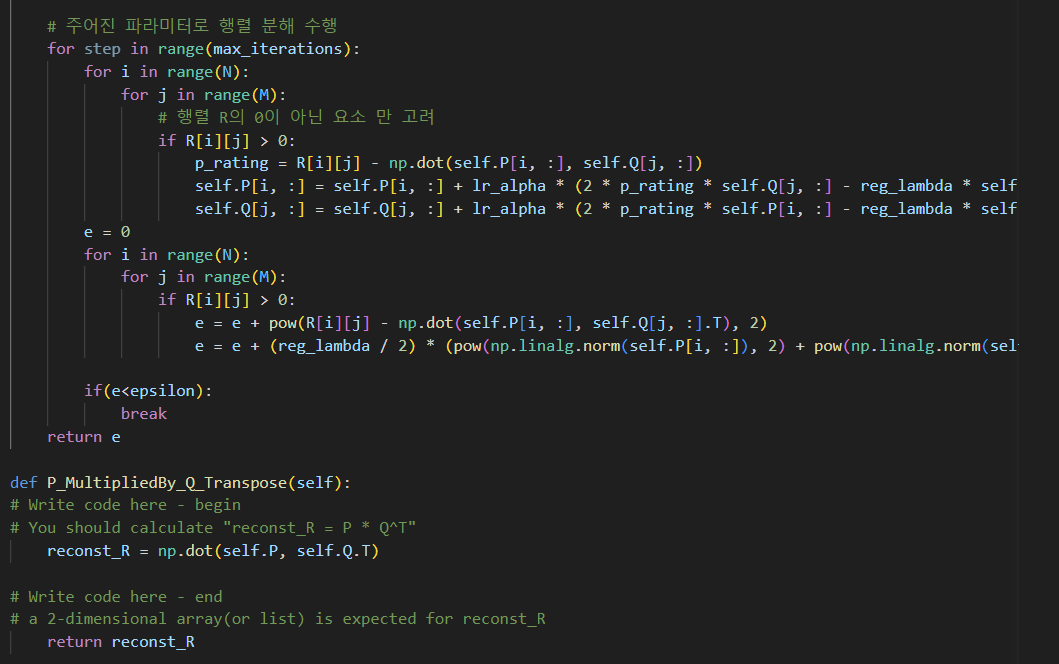


|  |  |
| --- | --- |
| **과목명** |  |
| 학과 | 컴퓨터공학과 |
| 학번 | 1824172 |
| 이름 | 박혁 |
| 제출일자 | 23/06/05 |
| 담당교수 | 한정규 |

**자신이 짠 소스 코드를 왜 그렇게 짰는지 이유를 설명할 것**

****

먼저 변수로 self.R self.P self.Q를 정의하고 None으로 초기화 합니다. 이 변수들은 매소드 호출시재할당 후 사용합니다. self.R 변수에 행렬 R을 할당하고 N과M에 각각 행의수, 열의수를 저장합니다. 그후 가중치를 저장할 행렬인 P,Q행렬의 크기를 구하고 행렬을 초기화 합니다.



함수호출시 사용되는 하이퍼파라미터인 max\_iterations만큼 훈련을 진행합니다. 각행과 열을 이중포문을 이용하여 반복하며 훈련이 진행됩니다. 행렬R의 값이 0보다 클때 경사하강법을 이용하여 P,Q행렬의 예측값을 업데이트 합니다. 그후 행렬R의 값이 0보다 클때 오차를 계산 합니다. 오차는 실제 값 R[i][j]과 예측 값 np.dot(self.P[i, :], self.Q[j, :])의 차이의 제곱으로 계산됩니다. 노름을 선언하여 이 값이 하이퍼파라미터인 입실론 보다 작아지면 트레이닝을 종료하는 것으로 과적합을 방지하였습니다. 트레이닝 종료 후 P,Q행렬의 곱으로 행렬R을 재구성합니다.

Python 3.9.16버전에서 구현하였습니다.

평가 1. K에 따른 Reconstruction 성능.

이미지의 손상이 없고 λ값이 0이므로 K의 값이 변한다 해도 변화가 없습니다. 출력 또한 나오지 않습니다.

1-1 K=16



1-2 K=32



1-3 K=64

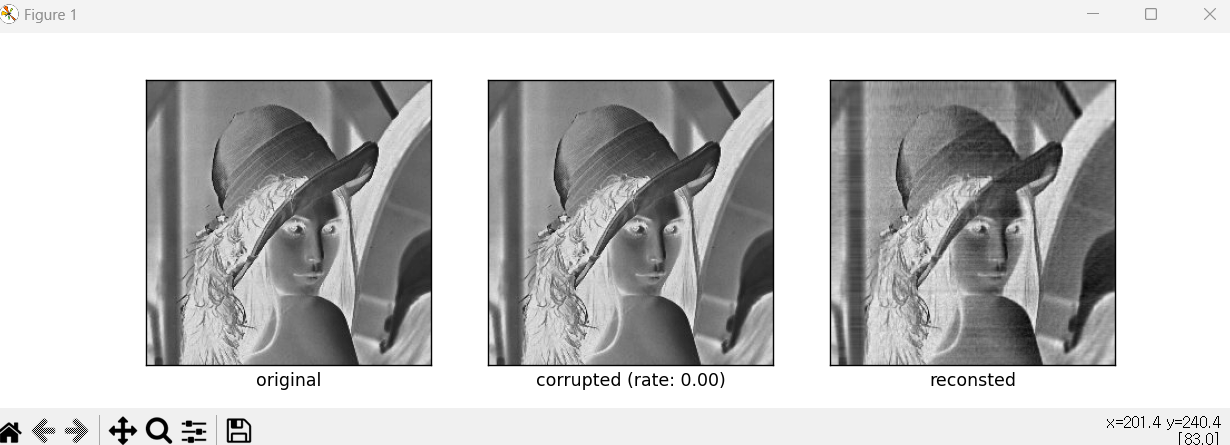


평가 2. λ의 기능

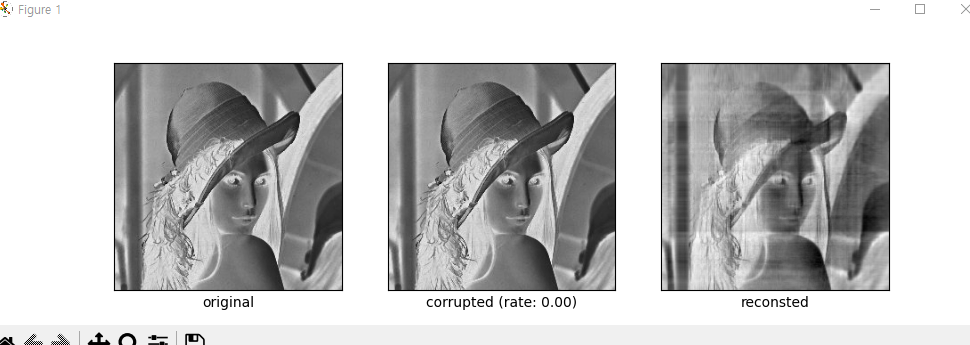
2-1 λ=0



2-2 λ=0.001



2-3 λ=0.01

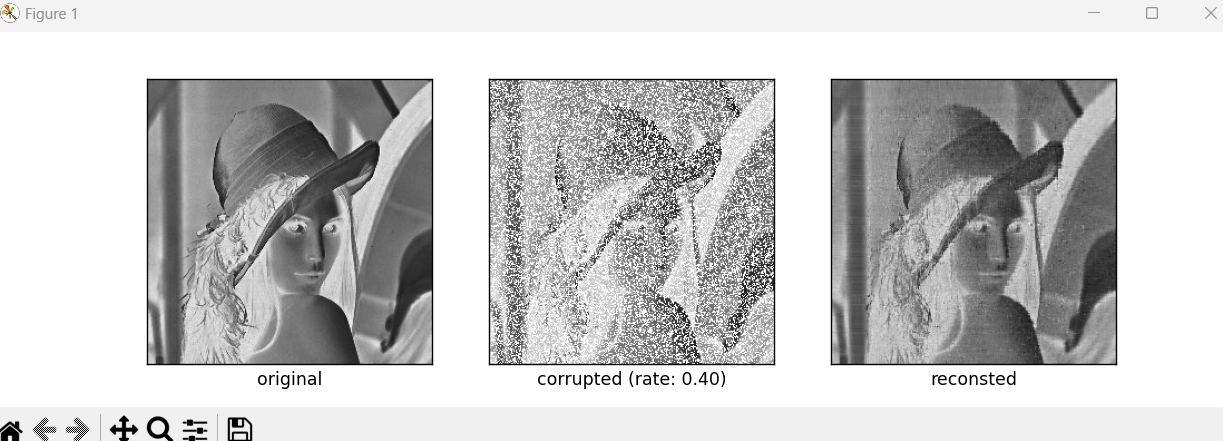


3. 평가: Matrix Factorization을 구현하여 다음 설정에서 실행한 후 결과를 snapshot으로 제시 하여라.

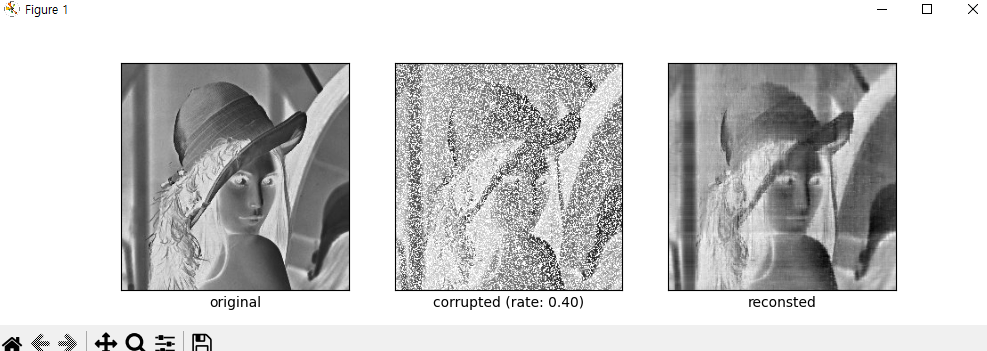
3-1 K=64 λ=0.0 40



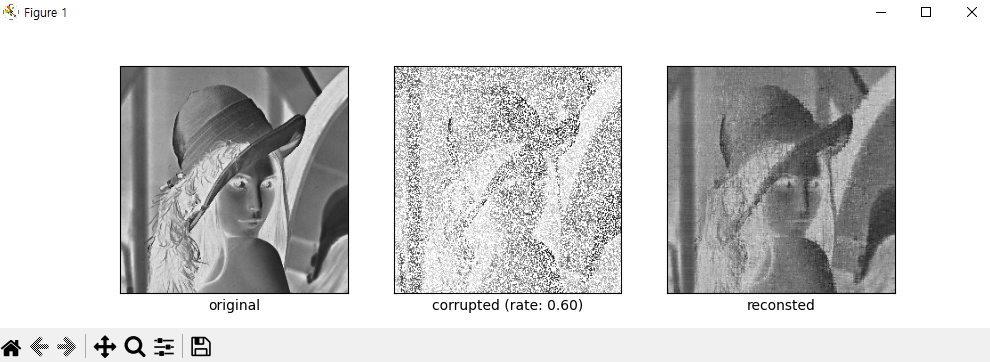
3-2 K=64 λ=0.001 40



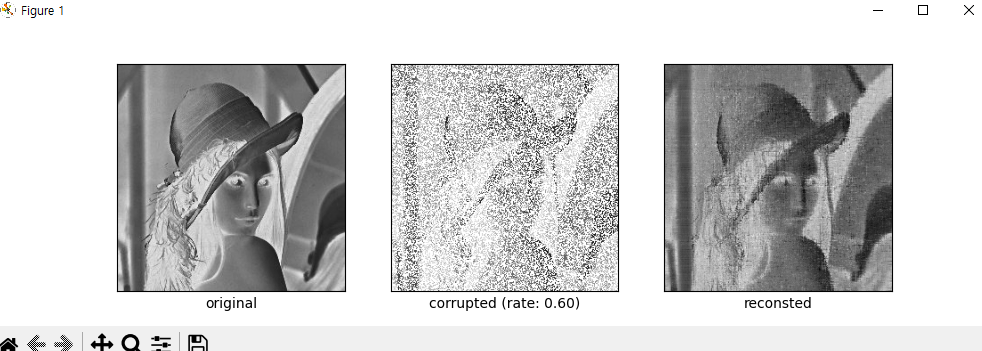
3-3 K=64 λ=0.01 40



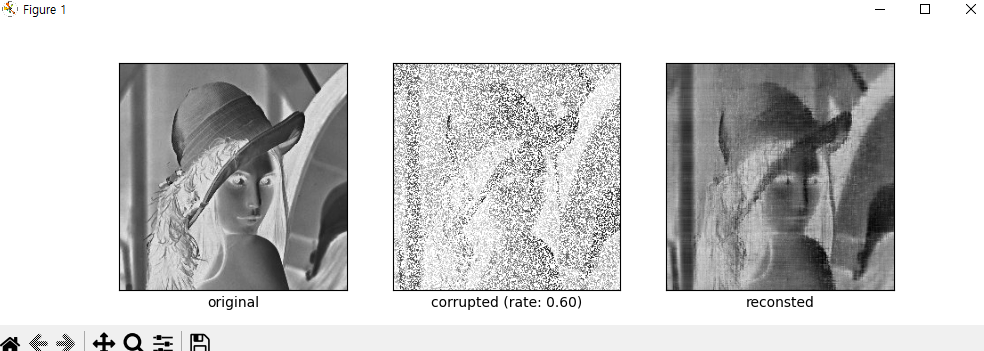
3-4 K=64 λ=0.0 60



3-5 K=64 λ=0.001 60



3-6 K=64 λ=0.01 60



**Feature vector dimension k가 클 경우와 작을 경우의 각각의 장단점을 고찰하여라.**

Feature vector dimension k가 클수록 더욱 선명한 결과의 이미지를 도출할 수 있다.

K값이 클수록 더 자세한 부분까지 트레이닝을 진행할 수 있습니다. 따라서 예측의 정확도가 상승합니다. 하지만 더 많은 계산을 수행하게 됩니다. 따라서 계산 복잡성이 높아지고 훈련의 시간이 더 오래 걸리게 되므로 비효율적인 계산이 될 수 있습니다.

반대로 k가 작을 경우 훈련의 시간이 줄어들고 계산의 복잡도가 감소하지만 자세한 부분까지 트레이닝을 진행 할 수 없어 예측의 정확도가 줄어듭니다.

**Regularization weight 𝜆가 클 경우와 작을 경우 어떤 효과가 있는지 고찰하여 라**

𝜆가 클 경우 정규화 항이 큰 비중을 차지하게 되어 가중치를 작게 가지게 됩니다. 따라서 과적합을 방지하고 일반화 성능을 향상시킬 수 있습니다. 하지만 값이 너무 커진다면 정규화 항이 너무 크게 작용하여 학습을 어렵게 만들 수 있습니다.

반대로 𝜆가 정규화의 강도가 작게 되어 가중치를 제한하는 역할이 약화됩니다. 따라서 정규화 항의 영향이 거의 없어지게 됩니다. 이는 모델이 과적합될 가능성이 증가합니다.

**Matrix Factorization을 작성해서 테스트해 보면서 느낀 점이나 의문점등을 자유롭게 기술할 것.**

𝜆값에 따른 정규화의 강도에 따라서 학습률이 달라지는 현상을 직접 확인하면서 이론으로만 이해하고 있던 경사하강법 알고리즘의 원리를 직접 확인해 볼 수 있었던 좋은 기회였습니다. 또한 코드를 작성하면서 여러 노름들에 대해 알아보게 되어 여러 정규화 방법에 대해서 공부 하게 되었습니다.