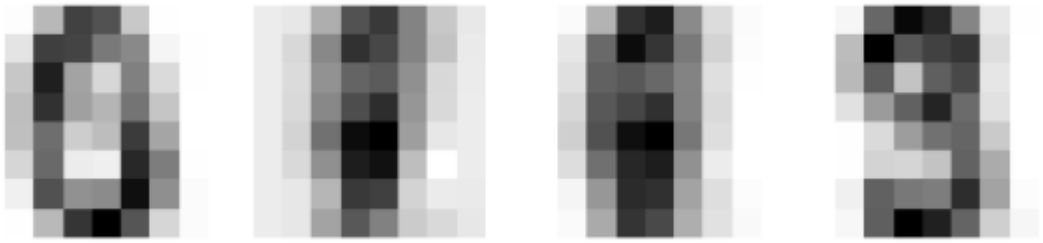
20191571 김세영

1. 기존 데이터와 축소한 결과를 복원한 데이터와의 MSE 및 복원 데이터를 시각화

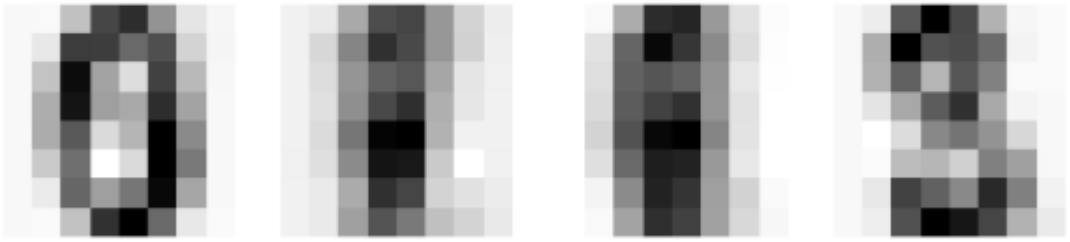
-2차원

2차원 : MSE Error: 13.421012200761453



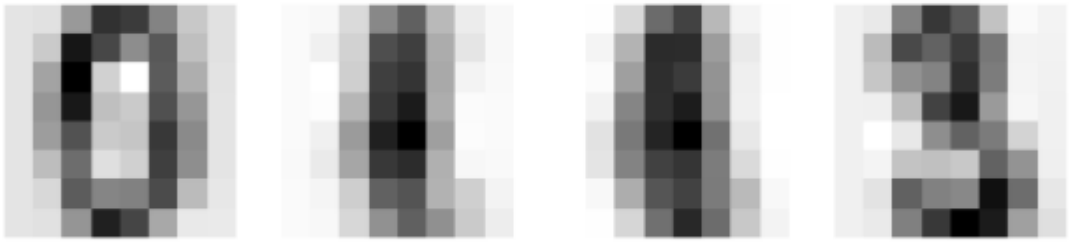
-3차원

3차원 : MSE Error: 11.206800697129164



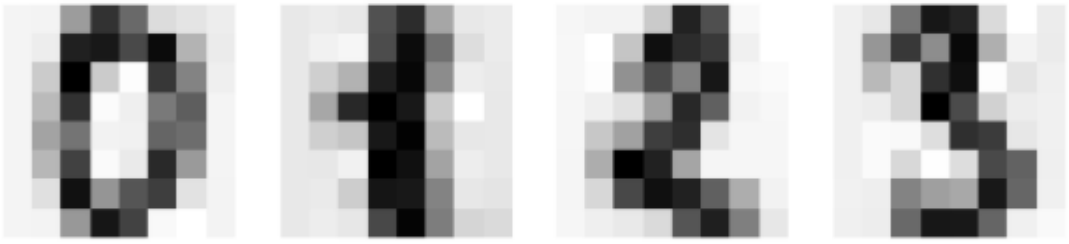
-4차원

4차원 : MSE Error: 9.627986407129 21



-32차원

32차원: MSE Error:0.6316360146108383



-코드

```
def recon(X,n):
    pca_digit = student_pca(X, n_components=n) #pca
    cov_X=(X-X.mean(axis=0)).T
    pca_cov=np.cov(cov_X)
    eigenValues,eigenVectors=np.linalg.eig(pca_cov)

    pca_re=np.transpose(np.dot(eigenVectors[:,: n],np.transpose(pca_digit)))+X.mean(axis=0)

    mse_error=((pca_re-X)**2).mean()
    print(f"{n}차원: MSE Error: {mse_error}")

    n_samples = pca_re.shape[0]
    images = pca_re.reshape((n_samples, -1))
    _, axes = plt.subplots(nrows=1, ncols=4, figsize=(10, 3))
    for ax, image in zip(axes, images):
        ax.set_axis_off()
        image = image.reshape(8, 8)
        ax.imshow(image, cmap=plt.cm.gray_r, interpolation='nearest')
```

recon(X,2)

recon(X,3)

recon(X,4)

recon(X,32)

2. 오차율이 다른 이유

pca에서 n차원으로 압축할 때, eigenvector에 투영하는 과정에서 d개인 eigenvector 중 내림차순으로 정렬한 n개만 사용한다. 데이터를 복원할 때 사용하는 eigenvector도 총 d개 중 n개만 사용하기 때문에 n이 높을수록 eigenvector가 많이 사용되어 원본 데이터와 오차율이 줄어든다. 실제 실험 결과에도 2차원으로 압축했을 때는 MSE Error: 13.4210122007614 53이지만 32차원으로 압축했을 때는 MSE Error: 0.6316360146108383로 32차원일 때가 MSE가 더 낮다.