Yang Liu
3323643

Lab Exercise 2

COMP6248
yl58n21@soton.ac.uk

1.1 Implement gradient-based factorisation using PyTorch's AD

```
def sgd_factorise_ad(A:torch.Tensor, rank:int, num_epochs=1000, lr=0.01) ->
Tuple[torch.Tensor, torch.Tensor]:
    m, n = A.shape
    U = torch.rand(m, rank, requires_grad=True)
    V = torch.rand(n, rank, requires_grad=True)
    for epoch in range(num_epochs):
        U.grad = V.grad = None
        loss = torch.nn.functional.mse_loss(A, U @ V.t(),
        reduction="sum")
        loss.backward()
        U.data = U - lr * U.grad
        V.data = V - lr * V.grad
    return U, V
```

1.2 Factorise and compute reconstruction error on real data

The reconstruction error is 15.228847. The loss of a rank-2 reconstruction computed using a truncated SVD is 15.228833 which is nearly the same as our previous reconstruction error.

1.3 Compare against PCA

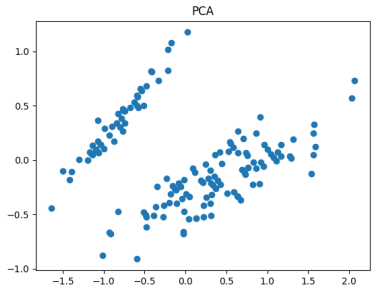

Figure 1: data projected using PCA

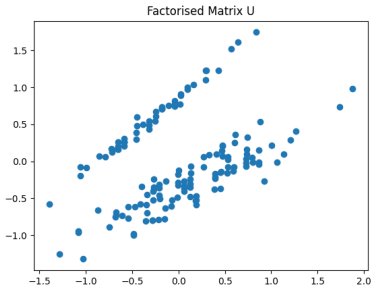

Figure 2: data from factorised matrix $\hat{U}$

I find that the scatter plots of data projected using PCA(**Figure 1**)and matrix $\hat{U}$ (**Figure 2**)look similar, it seems like **Figure 2** is like the rotated and scaled version of **Figure 1**. When we are doing dimension reduction like this , maximizing the variance is equal to minimizing the reconstruction error.

2.1 Implement the MLP

```
for epoch in range(epochs):
    W1.grad = W2.grad = b1.grad = b2.grad = None
    logits = torch.relu(data_tr @ W1 +b1) @ W2 + b2
    loss=torch.nn.functional.cross_entropy(logits ,targets_tr ,reduction="sum")
    loss.backward()
    W1.data -= lr *W1.grad
    W2.data -= lr *W2.grad
    b1.data -= lr* b1.grad
    b2.data -= lr* b2.grad
```

2.2 Test the MLP

Number	Training Set Accuracy	Test Set Accuracy
1	0.96	0.92
2	0.98	0.94
3	0.99	0.92
4	1	0.92
5	0.99	0.90
6	0.97	0.92
7	1	0.94
8	0.99	0.92

From the results we got, the accuracy of a training set is always higher than test set, which probably means the problem of overfitting. And the difference between accuracy also indicates that a different initialization of weights will lead to different results(some better, some worse).