Machine Learning for Materials Scientists
Summer Semester 2024

Prof. Bernhard Eidel
Rahul Vishnu Narkhede
rahul-vishnu.narkhede@student.tu-freiberg

TECHNISCHE UNIVERSITÄT BERGAKADEMIE FREIBERG
The University of Resources. Since 1765.

Exercise 2 - Principal Component Analysis (PCA)

June 4, 2024

The learning objectives of this exercise are

1. to write code that implements PCA (based on the projection perspective),
2. to apply the code to the popular MNIST digit dataset for dimensionalty reduction,
3. to apply logistic regression for binary classification in MNIST dataset

In the given jupyter notebook `Ex02_PCA_LogisticRegression.ipynb` update the functions `normalize`, `eig`, `projection_matrix` and `PCA`. These functions execute the steps involved in PCA. Next, update the class `MyPCA` and its corresponding methods. Apply this class to perform dimensionality reduction in subsequent code blocks. For a sample dataset `X`, the implementation of PCA should look as given in the listing 1:

```
mypca = MyPCA(n_coomponents = 10)
mypca.fit(X, mu = None, std = None)
z = mypca.transform(X)
reconst = mypca.inverse_transform(z)
total_expl_variance_ratio = np.sum(mypca.explained_variance())
print("total explained variance ratio: ", total_expl_variance_ratio.round(4))
```

Listing 1: Sample of using `MyPCA` class.

The notebook also contains `LogisticRegression` class which implements logistic regression for binary classification between two digits of the MNIST dataset. Apply the `MyPCA` class with the logistic regression implementation and evaluate the effect of number of PCA components on the accuracy of the classification.

Go through the documentation for `sklearn.linear_model.LogisticRegression` and apply the class for binary classification. Use the class `MyPCA` and evaluate the effect of number of PCA components on the accuracy of the classification.

Step by step hints for finishing the code block are provided in the `ipython` notebook as well.

Update the code in the jupyter notebook which has been provided for the exercise. Account for the instructions and comments above.

For submission, provide a PDF document(based on LaTeX) explaining the results with:

1. The completed code block with all relevant comments (wherever required).
2. Plots showing the change in accuracy for an increasing number of components ranging from 2 to 100 components.
3. A visualization of those digits which are being mislabelled even for using 100 components. Explain why this mislabelling might be occurring and propose ways to increase prediction accuracy.

Deadline: 18-June-2024, 23:59h