8. Dimensionality Reduction Using PCA

Project due Oct 21, 2020 05:29 IST *Completed*

PCA finds (orthogonal) directions of maximal variation in the data. In this problem we're going to project our data onto the principal components and explore the effects on performance.

You will be working in the files `part1/main.py` and `part1/features.py` in this problem

Project onto Principal Components

3.0/3.0 points (graded)

Fill in function `project_onto_PC` in **features.py** that implements PCA dimensionality reduction of dataset X.

Note that to project a given $n \times d$ dataset X into its k-dimensional PCA representation, one can use matrix multiplication, after first centering X:

$$\widetilde{X}V$$

where $\widetilde{X}$ is the centered version of the original data X using the mean learned from training data and V is the $d \times k$ matrix whose columns are the top k eigenvectors of $\widetilde{X}^T\widetilde{X}$. This is because the eigenvectors are of unit-norm, so there is no need to divide by their length.

Function input:: You are given the full principal component matrix V' as `pcs` and the features mean computed from the training data set as `feature_means` in this function. Note that `pcs` and `feature_means` are learned from the training data set, which should not be computed in this function using X.

Available Functions: You have access to the NumPy python library as `np`.

Correction on features.py and main.py: Please download a corrected version of the project archive mnist.tar.gz or correct these files as decribed above before this problem.

```
def project_onto_PC(X, pcs, n_components, feature_means):
    """
    Given principal component vectors pcs = principal_components(X)
    this function returns a new data array in which each sample in X
    has been projected onto the first n_components principcal components.
    """
    # TODO: first center data using the feature_means
    # TODO: Return the projection of the centered dataset
    #       on the first n_components principal components.
    #       This should be an array with dimensions: n x n_components.
    # Hint: these principal components = first n_components columns
    #       of the eigenvectors returned by principal_components().
    #       Note that each eigenvector is already be a unit-vector,
    #       so the projection may be done using matrix multiplication.
    try:
        X_centered = center_data(X)
```

Press ESC then TAB or click outside of the code editor to exit

Correct

Test results

See full output

CORRECT

See full output

Submit You have used 1 of 25 attempts

Note: we only use the training dataset to determine the principal components. It is **improper** to use the test dataset for anything except evaluating the accuracy of our predictive model. If the test data is used for other purposes such as selecting good features, it is possible to overfit the test set and obtain overconfident estimates of a model's performance.

Testing PCA

1.0/1.0 point (graded)
Use `project_onto_PC` to compute a 18-dimensional PCA representation of the MNIST training and test datasets, as illustrated in `main.py` .

Retrain your softmax regression model (using the original labels) on the MNIST training dataset and report its error on the test data, this time using these 18-dimensional PCA-representations rather than the raw pixel values.

If your PCA implementation is correct, the model should perform nearly as well when only given 18 numbers encoding each image as compared to the 784 in the original data (error on the test set using PCA features should be around 0.15). This is because PCA ensures these 18 feature values capture the maximal amount of variation from the original 784-dimensional data.

Error rate for 18-dimensional PCA features = 0.1483 ✔ **Answer:** 0.1474

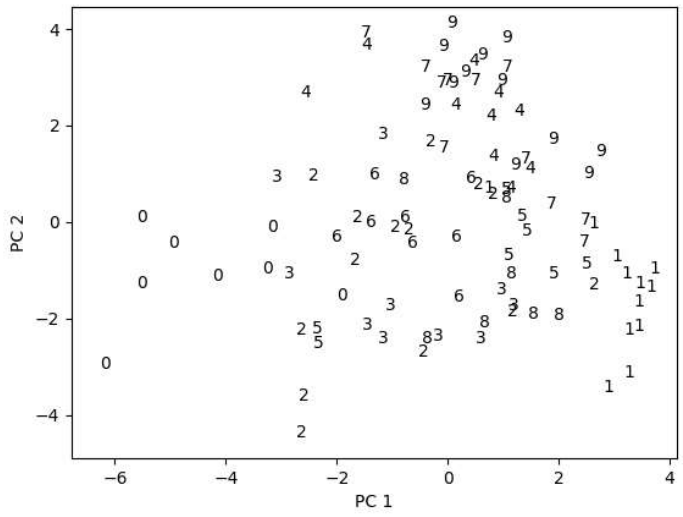
Submit You have used 2 of 5 attempts

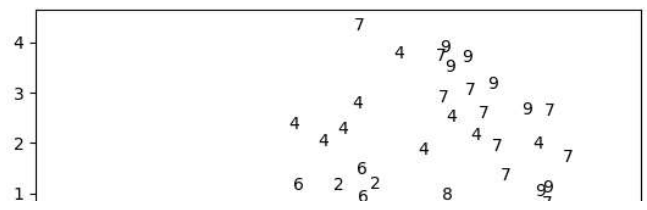
Answers are displayed within the problem

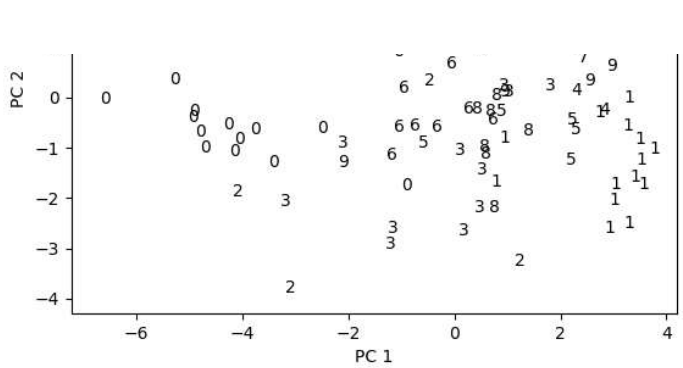
Testing PCA (continued)

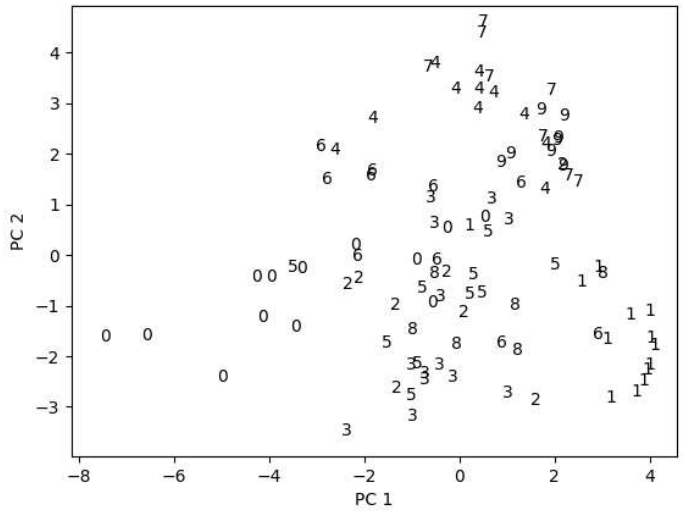
1.0/1.0 point (graded)
Use `plot_PC` in **main.py** to visualize the first 100 MNIST images, as represented in the space spanned by the first 2 principal components of the training data.

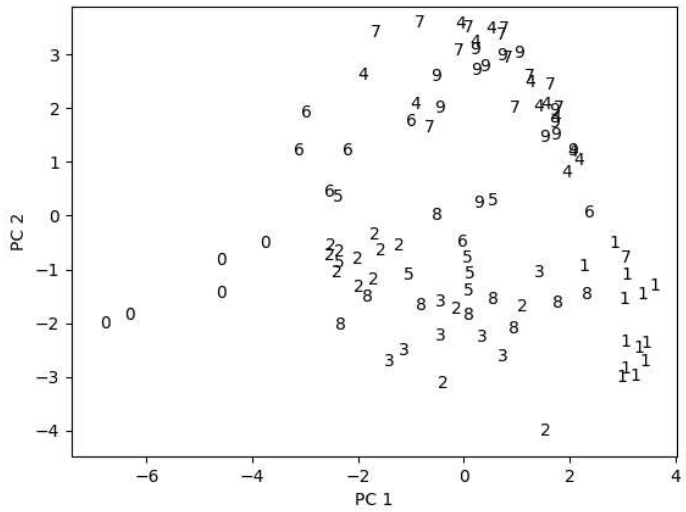
What does your PCA look like?

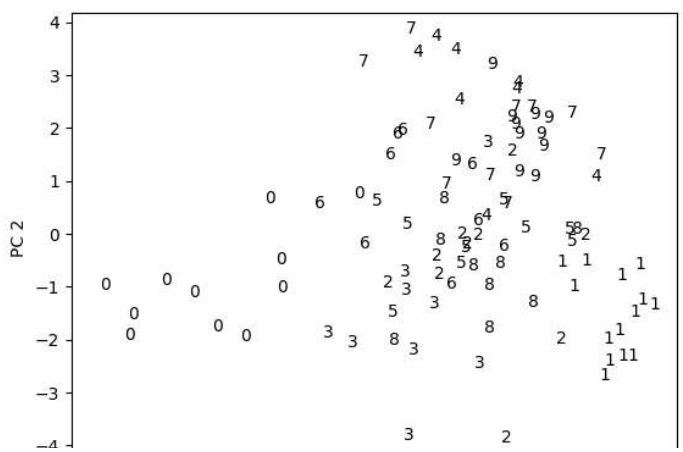


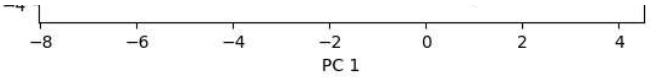



PC 2
PC 1


PC 2
PC 1


PC 2
PC 1


PC 2



Use the calls to `plot_images()` and `reconstruct_PC` in **main.py** to plot the reconstructions of the first two MNIST images (from their 18-dimensional PCA-representations) alongside the originals.

Submit You have used 1 of 2 attempts

Answers are displayed within the problem

Remark: Two dimensional PCA plots offer a nice way to visualize some global structure in high-dimensional data, although approaches based on nonlinear dimension reduction may be more insightful in certain cases. Notice that for our data, the first 2 principal components are insufficent for fully separating the different classes of MNIST digits.

Discussion

Hide Discussion

Topic: Unit 2 Nonlinear Classification, Linear regression, Collaborative Filtering (2 weeks):Project 2: Digit recognition (Part 1) / 8. Dimensionality Reduction Using PCA

Add a Post

Show all posts by recent activity

?	what does project it on the n_components mean? totally lost in terms of concept..	9
	Testing PCA - TypeError: only length-1 arrays can be converted to Python scalars on Testing PCA I'm getting "TypeError: only length-1 arrays can be converted to Python scalars" This happens into the project_onto_...	2
	Error in center_data(X) center_data(X) fucntion in the given project file features.py returns (X - feature_means), feature_means. However, the grader center_...	1
?	ValueError for Testing PCA I have done the projection using matrix multiplication. I tried both @ and np.matmul( ) for matrix multiplication. My code passed the...	10
	Implementing PCA in Python with Scikit-Learn https://stackabuse.com/implementing-pca-in-python-with-scikit-learn/	1
?	PyCharm 2020.2.3 update Hi, my PC is asking for the PyCharm update. Should I update it in the middle of the project?	2
?	V' ?	2
?	[Staff] My PCA plot doesn't look like any of the answers, what should I do? (PyCharm's 'exit code 138') I got the error to the softmax regression using PCA vectors right, but my plot appears to be wrong. Is there anything I could change t...	13
	When will PCA not help? A small collection of my initial effort to understand when PCA will not be helpful. Seems like it is dependent on how and where the v... Community TA	2
	Another approach - from someone who is also new to this PCA thing. Hello, This PCA "thing" is powerful. So I felt the need to spend good time on it. Here is the approach I took. 1) Eat, digest and eat agai... Community TA	1
	Grader Passed for Project onto Principal Components but test.py fails with error	3
☑	Testing PCA (continued) - code already given? I'm not sure if it is a mistake but there is already working code in main.py for this last question, am I correct? I mean after 'Testing PC...	2
	Project onto Principal Components	3

Previous

Next

edX

About
Affiliates
edX for Business
Open edX
Careers
News

Legal

Terms of Service & Honor Code
Privacy Policy
Accessibility Policy
Trademark Policy
Sitemap

Connect

Blog
Contact Us
Help Center
Media Kit
Donate

/