Nscript

Got It!

#### 1. Dimension reduction with PCA

#### 2. Dimension reduction

Dimension reduction represents the same data using less features and is vital for building machine learning pipelines using real-world data. Finally, in this video, you'll learn how to perform dimension reduction using PCA.

#### 3. Dimension reduction with PCA

We've seen already that the PCA features are in decreasing order of variance. PCA performs dimension reduction by discarding the PCA features with lower variance, which it assumes to be noise, and retaining the higher variance PCA features, which it assumes to be informative.

#### 4. Dimension reduction with PCA

To use PCA for dimension reduction, you need to specify how many PCA features to keep. For example, specifying n_components=2 when creating a PCA model tells it to keep only the first two PCA features. A good choice is the intrinsic dimension of the dataset, if you know it. Let's consider an example right away.

#### 5. Dimension reduction of iris dataset

The iris dataset has 4 features representing the 4 measurements. Here, the measurements are in a numpy array called samples. Let's use PCA to reduce the dimension of the iris dataset to only 2. Begin by importing PCA as usual. Create a PCA model specifying n_components=2, and then fit the model and transform the samples as usual. Printing the shape of the transformed samples, we see that there are only two features, as expected.

#### 6. Iris dataset in 2 dimensions

Here is a scatterplot of the two PCA features, where the colors represent the three species of iris. Remarkably, despite having reduced the dimension from 4 to 2, the species can still be distinguished. Remember that PCA didn't even know that there were distinct species. PCA simply took the 2 PCA features with highest variance. As we can see, these two features are very informative.

#### 7. Dimension reduction with PCA

PCA discards the low variance features, and assumes that the higher variance features are informative. Like all assumptions, there are cases where this doesn't hold. As we saw with the iris dataset, however, it often does in practice.

#### 8. Word frequency arrays

In some cases, an alternative implementation of PCA needs to be used. Word frequency arrays are a great example. In a word-frequency array, each row corresponds to a document, and each column corresponds to a word from a fixed vocabulary. The entries of the word-frequency array measure how often each word appears in each document. Only some of the words from the vocabulary appear in any one document, so most entries of the word frequency array are zero.

#### 9. Sparse arrays and csr_matrix

Arrays like this are said to be "sparse", and are often represented using a special type of array called a "csr_matrix". csr_matrices save space by remembering only

the non-zero entries of the array.

#### 10. TruncatedSVD and csr_matrix

Scikit-learn's PCA doesn't support csr_matrices, and you'll need to use TruncatedSVD instead. TruncatedSVD performs the same transformation as PCA, but accepts csr matrices as input. Other than that, you interact with TruncatedSVD and PCA in exactly the same way.