

Module 20 Challenge

New Attempt

Due Feb 27 by 11:59pm **Points** 100 **Submitting** a text entry box or a website url

Background

You are on the data science team of a medical research company that's interested in finding better ways to predict myopia, or nearsightedness. Your team has tried—and failed—to improve their classification model when training on the whole dataset. However, they believe that there might be distinct groups of patients that would be better to analyze separately. So, your supervisor has asked you to explore this possibility by using unsupervised learning.

You have been provided with raw data, so you'll first need to process it to fit the machine learning models. You will use several clustering algorithms to explore whether the patients can be placed into distinct groups. Then, you'll create a visualization to share your findings with your team and other key stakeholders.

Before You Begin

1. Create a new repository for this project called `unsupervised-machine-learning-challenge`. **Do not add this Challenge to an existing repository.**
2. Clone the new repository to your computer.

Files

Download the following files to help you get started:

Module 20 Challenge files  https://static.bc-edx.com/data/dl-1-1/m20/lms/starter/Starter_Code_v1.zip

Instructions

This activity is broken down into four parts:

- Part 1: Prepare the Data.
- Part 2: Apply Dimensionality Reduction.
- Part 3: Perform a Cluster Analysis with K-means.
- Part 4: Make a Recommendation.

Part 1: Prepare the Data

1. Read `myopia.csv` into a Pandas DataFrame.
 - **Note:** This file can be found in your Module 20 Challenge files.

2. Remove the "MYOPIC" column from the dataset.

- **Note:** The target column is needed for supervised machine learning, but it will make an unsupervised model biased. After all, the target column is effectively providing clusters already!

3. Standardize your dataset so that columns that contain larger values do not influence the outcome more than columns with smaller values.

Part 2: Apply Dimensionality Reduction

1. Perform dimensionality reduction with PCA. How did the number of the features change?

HIDE HINT

Rather than specify the number of principal components when you instantiate the PCA model, state the desired **explained variance**. For example, say that a dataset has 100 features. Using `PCA(n_components=0.99)` creates a model that will preserve approximately 99% of the explained variance, whether that means reducing the dataset to 80 principal components or 3.

For this assignment, preserve 90% of the explained variance in dimensionality reduction.

2. Further reduce the dataset dimensions with t-SNE and visually inspect the results. To do this, run t-SNE on the principal components, which is the output of the PCA transformation.

3. Create a scatter plot of the t-SNE output. Are there distinct clusters?

Part 3: Perform a Cluster Analysis with K-means

Create an elbow plot to identify the best number of clusters. Make sure to do the following:

- Use a `for` loop to determine the inertia for each `k` between 1 through 10.
- If possible, determine where the elbow of the plot is, and at which value of `k` it appears.

Part 4: Make a Recommendation

Based on your findings, write up a brief (one or two sentences) recommendation for your supervisor in your Jupyter Notebook. Can the patients be clustered? If so, into how many clusters?

Requirements

Data Preparation (25 points)

- Reads the csv into pandas (5 points)

- Previews the DataFrame (5 points)
- Removes the MYOPIC column from the dataset (5 points)
- Standardizes the dataset using a scaler (5 points)
- Names the resulting DataFrame X (5 points)

Dimensionality Reduction (40 points)

- PCA model is created and used to reduce dimensions of the scaled dataset (10 points)
- PCA model's explained variance is set to 90% (0.9) (5 points)
- The shape of the reduced dataset is examined for reduction in number of features (5 points)
- t-SNE model is created and used to reduce dimensions of the scaled dataset (10 points)
- t-SNE is used to create a plot of the reduced features (10 points)

Clustering (30 points)

- A K-means model is created (10 points)
- A `for`-loop is used to create a list of inertias for each k from 1 to 10, inclusive (5 points)
- A plot is created to examine any elbows that exist (10 points)
- States a brief (1-2 sentence) conclusion on whether patients can be clustered together, and supports it with findings (10 points)

Grading

This assignment will be evaluated against the requirements and assigned a grade according to the following table:

Grade	Points
A (+/-)	90+
B (+/-)	80–89
C (+/-)	70–79
D (+/-)	60–69
F (+/-)	< 60

Submission


To submit your Challenge assignment, click Submit, and then provide the URL of your GitHub repository for grading.

NOTE

You are allowed to miss up to two Challenge assignments and still earn your certificate. If you complete all Challenge assignments, your lowest two grades will be dropped. If you wish to skip this assignment, click Next, and move on to the next Module.

Comments are disabled for graded submissions in BootCamp Spot. If you have questions about your feedback, please notify your instructional staff or your Student Success Manager. If you would like to resubmit your work for an additional review, you can use the Resubmit Assignment button to upload new links. You may resubmit up to three times for a total of four submissions.

References

Reduced dataset from [Orinda Longitudinal Study of Myopia conducted by the US National Eye Institute](https://clinicaltrials.gov/ct2/show/NCT00000169) 
(<https://clinicaltrials.gov/ct2/show/NCT00000169>).