Question 1

Principal Component Analysis Overview and Use Cases

https://www.originlab.com/doc/Origin-Help/PrincipleComp-Analysis

https://towardsdatascience.com/a-one-stop-shop-for-principal-component-analysis-5582fb7e0a9c

https://jakevdp.github.io/PythonDataScienceHandbook/05.09-principal-component-analysis.html (this is a long article but very insightful into future lectures)

https://districtdatalabs.silvrback.com/principal-component-analysis-with-python

Libraries to download the Dow Jones Index stock data:

Yahoo Finance APIs (Recommended): https://pypi.org/project/yfinance/
RapidAPI: https://docs.rapidapi.com/docs/basics-creating-a-project

The DJI contains the aggregated stock index for 30 major corporations like Microsoft, Apple, JP Morgan, Disney, Coca-Cola, Procter and Gamble etc. in the United States. Find these companies below described in their stock tickers:

dji_list= [‘AXP’, ‘AAPL’, ‘BA’, ‘CAT’, ‘CSCO’, ‘CVX’, ‘DD’, ‘XOM’, ‘GE’, ‘GS’, ‘IBM’, ‘INTC’, ‘JNJ’, ‘KO’, ‘JPM’, ‘MCD’, ‘MMM’, ‘MRK’, ‘MSFT’, ‘NKE’, ‘PFE’, ‘PG’, ‘TRV’, ‘UNH’, ‘UTX’, ‘VZ’, ‘V’, ‘WMT’, ‘DIS’, ‘DJI’]

Data Standardization

Since the DJI has stocks from different sectors of the economy, the values may not be on the same scale. So, you may need to standardize the dataset using the Z- score
$z = (X - U) / S$

Where X is the features, U is the mean and S is the standard deviation.

You could also use Sklearn's standard scaler
https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

Principal Component Analysis in Python

Basic (linear) PCA

https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html

You can get the variance by looking at the **explained_variance_ratio_** attribute from the PCA class. Something like pca.explained_variance_ratio_

You can also get the weights by looking at the **components_** attribute. To get the first and second component weight, you could do:

First_component = pca.components_ [0]
Second component = pca.components_ [1]

Question 2: Dendrogram

How to implement a dendrogram using Scipy's library
https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.dendrogram.html

Using Sklearn's approach
https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

This tutorial shows you how to implement it using a sample data
https://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/

Question 3 and 4: Ensemble learning

Random Forest Classifier
https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

Random Forest Regressor
https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html