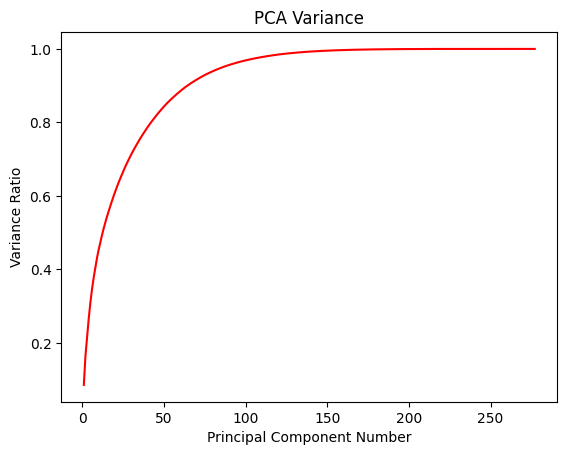
**Principal Component Analysis (PCA)** to select the most important features from the dataset. This is a crucial step in machine learning, especially when you have a lot of features, as it helps simplify the model and potentially improve its performance.

## What is PCA?

PCA is a technique used to reduce the dimensionality of data. It does this by finding new features called "principal components" which are combinations of the original features. These principal components are chosen in such a way that they capture most of the variation in the data. In simpler terms, it's like summarizing the data with fewer, but more informative, features.

In our project code snippet uses PCA to identify and select the most important features (principal components) from the Cardiac Arrhythmia dataset. This reduces the dimensionality of the data, which can improve the efficiency and performance of subsequent machine learning models. The code also includes visualizations to aid in selecting the optimal number of components to retain.

Graph of pca



1st plot:

**What it shows:**

* **X-axis:** Represents the number of principal components used.
* **Y-axis:** Represents the cumulative explained variance ratio. This ratio indicates the proportion of the total variance in the data that is explained by the selected principal components.
* **The Curve:** Shows how the cumulative explained variance ratio increases as you add more principal components.

**Interpretation:**

* **Initial Steep Slope:** The curve initially rises steeply, indicating that the first few principal components capture a large portion of the variance in the data.
* **Flattening Out:** As more components are added, the curve starts to flatten. This means that the additional components explain less and less of the remaining variance.
* **Reaching 1:** Ideally, the curve would reach a cumulative explained variance ratio of 1, meaning that all variance is explained. However, this often requires using all the original features.

**Purpose:**

This graph helps you determine the optimal number of principal components to retain. You want to find a point where the curve starts to flatten significantly, indicating that adding more components wouldn't provide much additional information. This point represents a good trade-off between dimensionality reduction and information retention. The code aims for 95% explained variance.

Graph 2

**What it shows:**

* **X-axis:** Represents the number of principal components (same as the first graph).
* **Y-axis:** Represents the eigenvalue associated with each principal component. Eigenvalues indicate the amount of variance explained by a particular principal component. Higher eigenvalue means more variance explained.
* **The Curve:** Shows how the eigenvalues decrease as you consider more principal components.

**Interpretation:**

* **Descending Order:** Eigenvalues are typically plotted in descending order, with the first principal component having the highest eigenvalue.
* **Sharp Drop:** Often, you'll see a sharp drop in eigenvalues initially, followed by a more gradual decline. This again highlights that the first few components are the most important.
* **"Elbow" Point:** The "elbow" point on this graph can also be used to determine the optimal number of components. It's the point where the curve starts to bend significantly.

**Purpose:**

Similar to the first graph, this graph helps in determining the optimal number of components to keep. By identifying the "elbow" point or components with eigenvalues above a certain threshold (often 1), you can reduce the dimensionality while retaining the most important information.

**In the given code,** both graphs are used to help in selecting the number of components. However, instead of looking for the elbow or components with eigen values greater than 1, **the code selects the component number at which the cumulative explained variance ratio crosses 95%** using this line