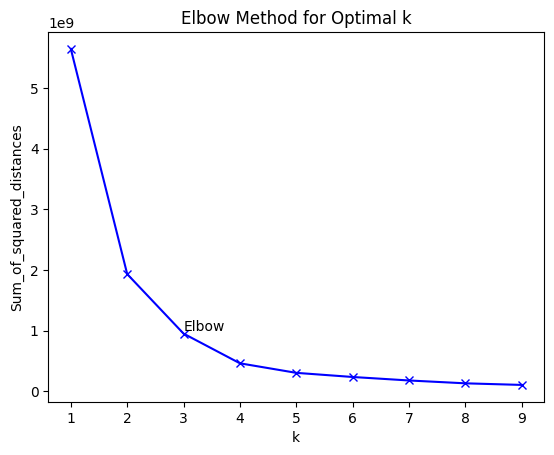
Task 2C

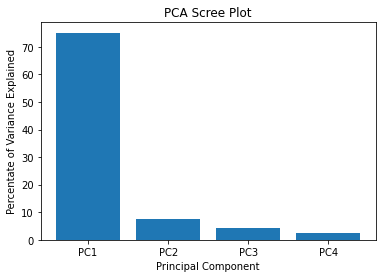
a) In Task 2A, we get overall better performance when using k-nearest neighbor algorithm compared to decision tree on this dataset. Using the k-value = 10 gives higher accuracy (accuracy = 86.9%) compared to k value of 5 with 82% accuracy. In order to get these results, the steps done are: preprocessing data, such as dropping country rows that are not in life.csv (because these does not give much information about life.csv), replacing ‘..’ values by NaN values, splitting features into 2/3 X\_train and 1/3 X\_test, classlabels into 2/3 y\_train and 1/3 y\_test. After splitting, imputating missing data in features is done by calculating median of X\_train only (because we don’t want X\_test to see X\_train data as this would lead to overfitting), then applied to both X\_train and X\_test. Furthermore, we normalize the data before fitting and predicting using the decision tree and k-nearest neighbor algorithm. The decision tree has lower accuracy because it is limited to maximum depth = 4. Though k-nearest neighbor is sensitive to noise and irrelevant attributes, since it scores the highest, this may be because most the features are not highly correlated. There aren’t many outliers in the data in which k-nearest neighbor can perform well.

b) The first steps of preprocessing are the same to task 2A until splitting the train and test set, and imputing the data. Furthermore, we generate feature interaction pairs by multiplying feature 1’s and feature 2’s column data, and we do this repeatedly until all combinations of 20 features are calculated. For k- means, first we decide k using elbow method, then we find KMeans according to the data points from the original 20 features, and assigning each country to cluster labels between 0-2. Feature engineering is done by calculating the top 4 scores found by calculating mutual information by SelectKBest. After selecting the top 4 headings, we standarize the data of these 4 features and apply k-nearest neighbor k = 5. Another way to generate features is using PCA. First, we standardize the imputed data of X\_train and X\_test, then compute the 4 PCA features. These 4 PCA features will be be used in calculating the knn. Lastly, the naïve feature selection approach is done by taking the first 4 features available in the dataset, splitting these data, imputing median, scaling the data and applying the k-nearest neighbor k=5.



c) The method used to select the best number of clusters (k) is Elbow method. Elbow method uses sum of squared distances measures. As k increases, SSE will decrease since we will have fewer number of points in a cluster, therefore points are closer to centroids. However, the decrease will eventually be smaller every time so we need to choose the elbow point where after this point, the decrease will not be as significant. From this graph, the elbow point is when k=3. So, we choose 3 clusters for the clustering label feature generation.

d) To select top 4 feature, SelectKBest function with k=4 is used with parameter mutual\_info\_classif. Since we want to know the features that are well correlated with class, and not independent of class, we calculate the mutual information of each feature with each class. The higher the mutual information, the higher the information gain. Therefore, we choose the features that have top 4 mutual information score from the SelectKBest function.



e) The 5-nearest neighbor classification method with feature selection method by feature engineering has the best accuracy compared to using PCA with n = 4 and naïve blocking method. Feature engineering is useful because we have more possibly better features to choose from, and mutual information is quite reliable because it tells us the information gained. Naïve blocking method coincidentally has the same accuracy as PCA. Feature engineering has higher accuracy than because the n chosen for PCA is 4. According to the graph, when n is 4, the variance explained is really low. We would ideally want to find number of PCA has highest variance explained (around 99%, does not need to be 100%). Finally, naïve blocking method is not recommended since it is only based on choosing the first four headings.

f) Since according to PCA scree plot, PC4 (meaning when n = 4) does not give much variance, we want to take number of features in which we can cover most of the variance (around 90%). Therefore, we choose the best n to for better accuracy by calculating variance explained for 1 < n < 20, then we can choose n where variance explained is almost 99% (meaning the PC can cover 99% of the data). Moreover, we can also do cross validation as the input for the clustering K-Means method instead of using only X\_train. Cross validation can also be used to build K different models, in which we have the advantage of being able to make predictions on all our data. Wrapper ‘greedy’ method to find the best features could give the highest accuracy, however run time will be really slow. Instead of selecting all the top 4 features with highest mutual information, we could use chi2 instead since the top mutual information features could share too close values. We can also use domain knowledge for the calculation of new features Instead of calculating it by f1\*f2.

g) The classification is quite reliable since the best accuracy found is 83.6% by feature engineering. 5-nearest neighbor classification is quite reliable, but we can also consider choosing the best n values for different data. There are room for improvements, such as variating the test and train set data using cross-validation, using feature engineering by generating domain knowledge-based calculations. Although the higher the accuracy, we also need to consider overfitting.