**1. Why Only X\_train is Passed to the KMeans Model:**

* **K-Means Clustering:**
  + **Type:** Unsupervised learning algorithm.
  + **Reason:** In unsupervised learning, there are no labels (i.e., no y\_train) provided to the algorithm. The algorithm tries to learn the underlying structure of the data based solely on the input features (X\_train).
  + **Objective:** The K-Means algorithm aims to partition the dataset into k clusters (in this case, k=2). It does this by minimizing the variance within each cluster. The algorithm iterates through the data, assigning each data point to one of the clusters and then updating the cluster centroids until convergence.
* **Contrast with Supervised Learning:**
  + In supervised learning algorithms (like KNN, SVM, etc.), both X\_train and y\_train are provided because the algorithm needs to learn a mapping between input features and labels.

**2. Function of crosstab:**

* **Purpose:**
  + pd.crosstab() is a pandas function that computes a cross-tabulation of two or more factors. In the context of your code, it is used to compare the actual labels (y\_train) with the cluster labels (model.labels\_) assigned by the K-Means algorithm.
* **Interpretation:**
  + The crosstab output shows the frequency of data points assigned to each cluster (as labeled by K-Means) against the true labels from the training set. This allows you to see how well the clusters correspond to the actual classes.

**3. Why Accuracy Increased After model.labels\_ Addition:**

* **Initial Accuracy:**
  + The initial accuracy is calculated using the predictions generated by the K-Means model on the test data. Since K-Means doesn't directly predict the labels but rather assigns clusters, the initial accuracy might not be high.
* **Labels Adjustment:**
  + The model.labels\_ represent the cluster assignments for the training data. These labels can be compared against the true labels (y\_train) using pd.crosstab() to understand the mapping between clusters and actual classes.
  + If you observe a strong correspondence between a specific cluster and a class (e.g., Cluster 0 aligns mostly with class 0), you can adjust the cluster labels accordingly. This adjustment can lead to an increase in the accuracy score since you're effectively relabeling the clusters to match the actual classes more closely.

**Summary:**

* **K-Means only uses X\_train** because it's an unsupervised algorithm and doesn't require labels to form clusters.
* **crosstab** helps compare the actual classes with the cluster labels, allowing you to interpret how well the clustering performed.
* **Accuracy improvement** after relabeling the clusters (using model.labels\_) happens because you're aligning the clusters more accurately with the true classes, thereby improving the prediction accuracy.