***CHECKING OBESITY LEVEL OF A PERSON USING MACHINE LEARNING ALGORITHM***

**Problem Definition:**

Overweight and obesity are defined as abnormal or excessive fat accumulation that may impair health. Overweight and obesity are linked to more deaths worldwide than underweight. Childhood obesity is associated with a higher chance of obesity, premature death and disability in adulthood. Therefore, by using the certain details of persons, an efficient machine learning model is built by using machine learning algorithms.

**Data Analysis:**

The dataset contains 16 attributes which involves the details of a person like height, weight, gender, family history with overweight, etc., and the target is a classification variable which gives us the obesity level of the person. By using the describe function, one has to analyse the whole information of the dataset.

**EDA Concluding Remarks**:

We have to plot boxplots of all numerical columns of a dataset to see whether data contains outliers or not. If outliers are present, then we have to remove the outliers by using zscore parameter. By drawing a heatmap and corr function, observe the correlation between the attributes and target. Label Encoder is used for columns which are string type, to convert them to numerics. By using isnull funtion, we can see whether the dataset contain null values. If there are null values, we have to drop the columns of the dataset which contains null values. After these modifications, by using iloc function, we have to divide the dataframe into input(x) and output(y). Input contains all the attributes and output contains the target. Next we use train\_test\_split function to divide the dataset into four parts(x\_train,x\_test,y\_train,y\_test).

**Building Machine Learning Models:**

Now we have to build various machine learning models by using machine learning algorithms. We are building MultinomialNB(), DecisionTreeClassifier(), SVC(), KNeighborsClassifier(), RandomForestClassifier(), AdaBoostClassifier(), GradientBoostingClassifier() and Cross validation models of all algorithms. We have to see the various metrics like accuracy\_score, confusion\_matrix and classification\_report for all the models. By observing these metrics, we can tell which is the best model to use for prediction.

**Concluding Remarks:**

At last, we have to save the model which has high accuracy score. In our dataset, GradientBoostingClassifier has high accuracy score(96.4) than other classifier models. By using joblib.dump(), we have to save the object file.