**SMOTE and ADASYN ( Handling Imbalanced Data Set)**

**NOTE:** Imbalanced Data set is not the same as Skewed Data set. In general skewed just means an asymmetrical distribution around the mean of the probability function: [Skewness](https://en.wikipedia.org/wiki/Skewness). So it is used for individual variables or calculated for each column separately . However, Imbalanced Dataset (also called an unbalanced dataset ) refer to datasets where the observations or target class or dependent Column values has an uneven distribution. This means that one class label has a very fewer number of observations and the other has very high numbers of observations.

* It's better to call *distributions* as skewed and *datasets* as imbalanced to avoid confusion, as you've discovered.

**SMOTE and ADASYN ( Handling Imbalanced Data Set )**

Recently I was working on a project where the data set I had was completely imbalanced. It was a binary classification problem and the ratio of classes (dependent variables) 0 and 1 was 99:1.

What happens when the data set is imbalanced ?

If the data set is imbalanced the model will be Biased. Think of it like this if you are feeding the model only 0 for every possible combination it will give you a 0 for every set of input.

How do we Know if the model is imbalanced or not?

1. We check the count of the dependent categorical values the ratio should be 10:1 for the data set to be considered as a balanced Data set.

2 . Confusion matrix : After the prediction is done check the confusion matrix.

[[TRUE POSITIVE] [ FALSE POSITIVE ]

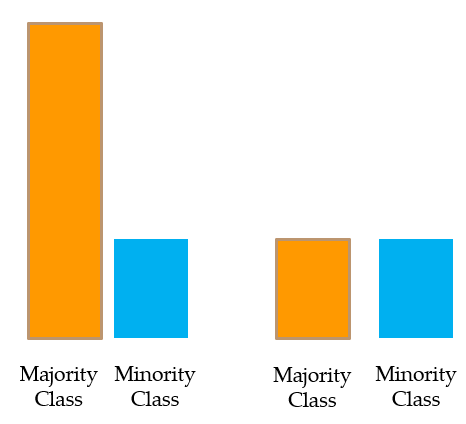
[FALSE NEGATIVE][TRUE NEGATIVE]]

if any of the values become 0. Well your model is Biased and your data set is imbalanced

**Ways to Handle Imbalanced data set:**

The basic approaches are called resampling techniques. There are two basic approaches.

**1. Undersampling:-**

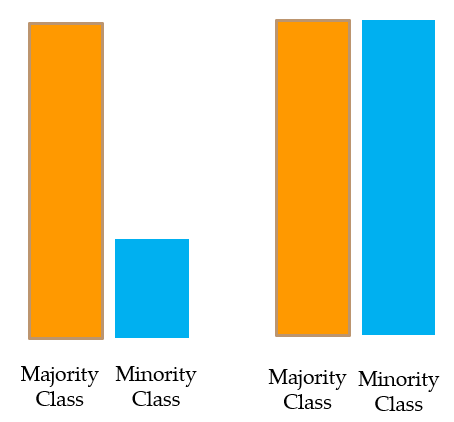


Undersampling or downsampling the majority class .

We pick random samples from the majority class and make it equal to the minority class count. This is called the*undersampling or downsampling of the majority class.*

**Issue:** **It’s not a good idea to ignore or let go so much of original data.**

**2. Oversampling:-**



Oversampling or upsampling the minority class .

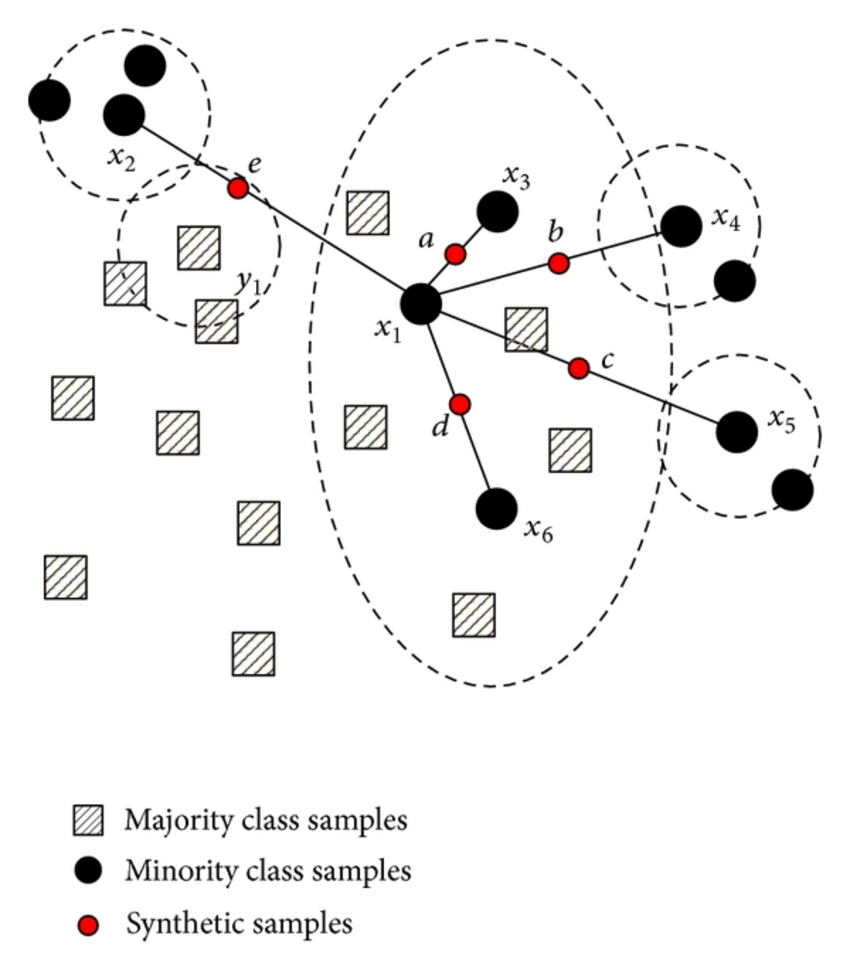
Here, sampling **with replacement** (i.e., same data pts from minority class are repeated) is applied to the minority class to create as many observations as there are in the majority class and the two classes are balanced. This is called*oversampling or upsampling the minority class.*

**Issue:** **Repetition of the same minority class data leads to overfitting.**

*Thus, we need better techniques than Undersampling and Oversampling.*

**3. SMOTE:**

What smote does is simple. First it finds the n-nearest neighbors in the minority class for each of the samples in the class . Then it draws a line between the the neighbors an generates random points on the lines.



Image

See the above image so it finds the 5 nearest neighbors to the sample points. then draws a line to each of them. Then create samples (or, new data pts) on the lines with class == minority class.

Method of Generating Syntheic Numbers:

Let there be two observations (x1,y1) and (x2,y2) from the minority class. As a first step, a random number between 0 and 1 is created, let’s call it r. The synthetic point will be (x1 + r\*(x2 -x1), y1 + r\*(y2 -y1)). It’s illustrated further with the following example.

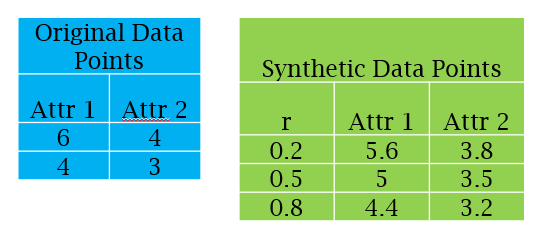


Image: Synthetic points generate from minority class

**An issue with SMOTE:**

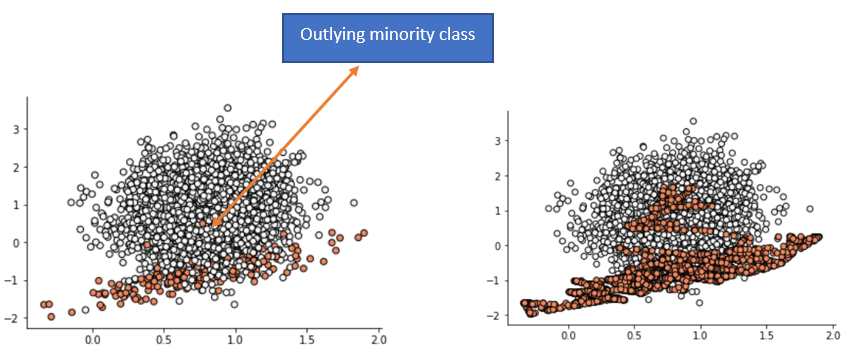


Image: Left Side: Original Data Right Side: Data after SMOTE is Applied

If there are observations in the minority class which are outlying and appears in the majority class, it causes a problem for SMOTE, by creating a line bridge with the majority class.

*Borderline SMOTE :-*

This solves the above issue.

***ADASYN:***

It is a improved version of Smote. What it does is same as SMOTE just with a minor improvement. After creating those sample it adds a random small values to the points thus making it more realistic. In other words instead of all the sample being linearly correlated to the parent they have a little more variance in them i.e they are bit scattered.

ADASYN is a more generic framework, for each of the minority observations it first finds the impurity of the neighborhood, by taking the ratio of majority observations in the neighborhood and k.

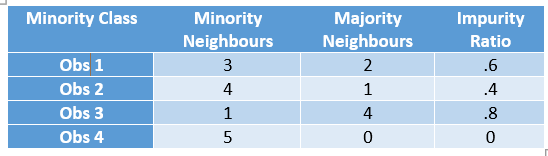


Fig: ADASYN Impurity Ratio

Now, first of all, this impurity ratio is converted into a probability distribution by making the sum as 1. Then higher the ratio more synthetic points are generated for that particular point.**Hence the number of synthetic observations to be created for Obs 3 is going to be double that of Obs 2**.