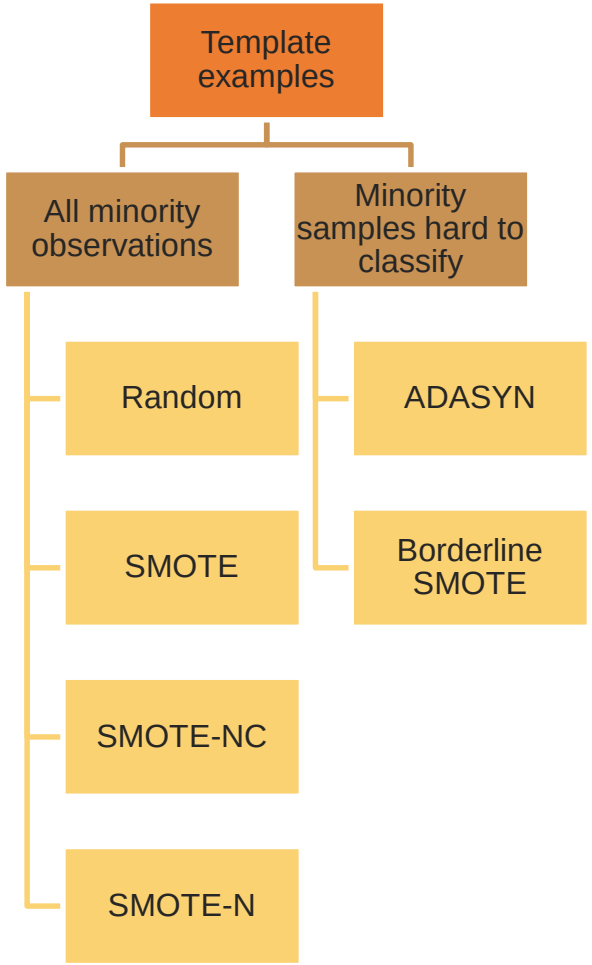
Train In Data

SVM SMOTE

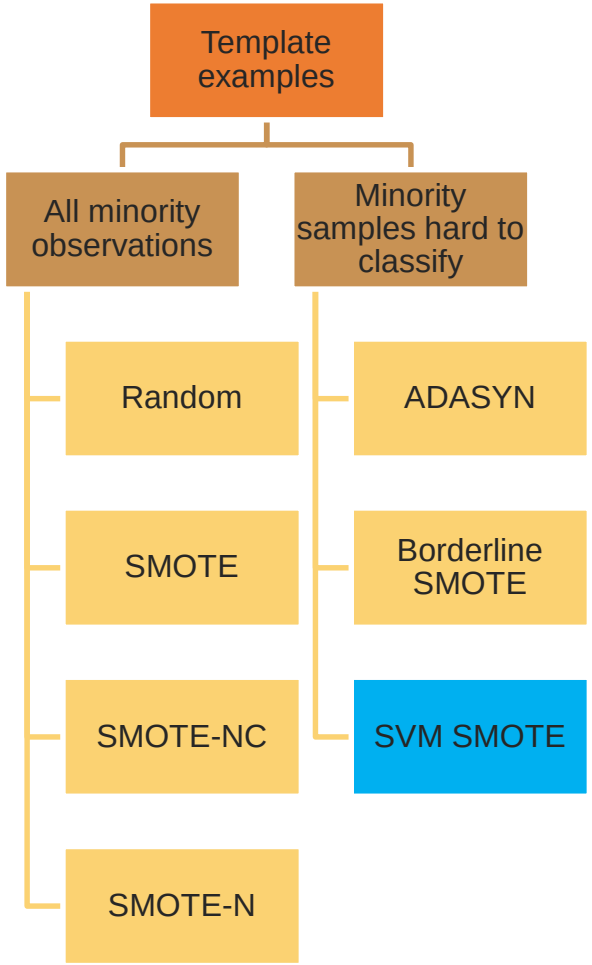
Let's recap on over-sampling



- **Random Oversampling, SMOTE** and its variants for categorical data ➔ use all samples in the minority class as templates

- **ADASYN and Borderline SMOTE:** use samples which are closer to the boundary with the other class, as templates

Let's recap on over-sampling



- **Random Oversampling, SMOTE** and its variants for categorical data ➔ use all samples in the minority class as templates
- **ADASYN and Borderline SMOTE:** use samples which are closer to the boundary with the other class, as templates
- **SVM SMOTE** also uses samples that are harder to classify

Templates for the syntethic data

ADASYN

- Samples from the minority if some of their closest neighbours are from opposite clases
- The more neighbours from the opposite class, the more likely it is to be used as template

Borderline SMOTE

- Observations from the minority for which the majority of the neighbours are from the opposite class

SVM SMOTE

- Templates are observations from the minority that are the support vectors of a support vector machine

Sample for interpolation

ADASYN

- Interpolate from template to closest neighbour from minority class only

Borderline SMOTE

- Version 1: Interpolate from template to closest neighbour from minority class only
- Version 2: Interpolate from template to closest neighbour from minority or majority, but with half the distance in the latter

SVM SMOTE

- Inter or extrapolates to neighbours from the minority

SVM SMOTE

Templates for synthetic data: the support vectors of a SVM

Samples for the interpolation: closest neighbours to the templates that also belong to the minority class

Method for the data creation:

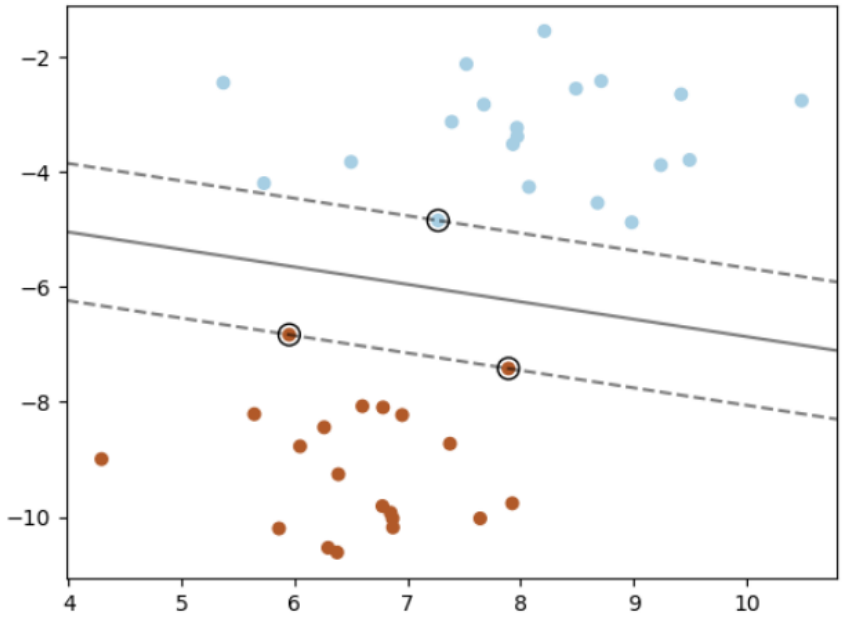
If most neighbours are from **minority** class ➔ **extrapolation**

- to expand the boundary

If most neighbours are from **majority** class ➔ **interpolation**

- keep the boundary as it is

Support vectors – the templates



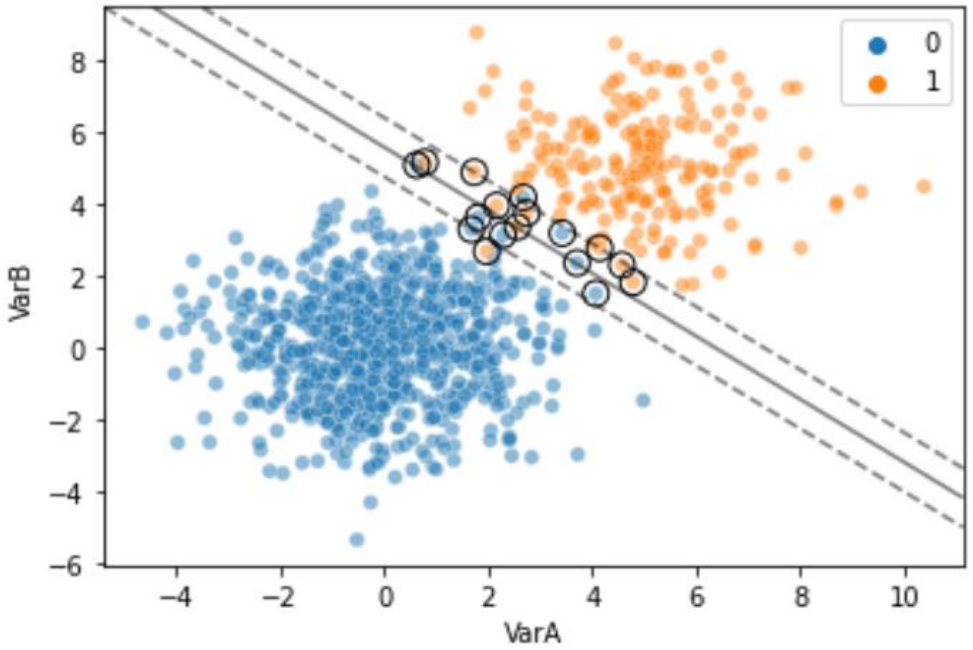
SVMs construct hyperplanes that separate the classes

They construct the hyperplane that has the largest distance to the nearest training points (the margins)

The samples on the margin are the support vectors

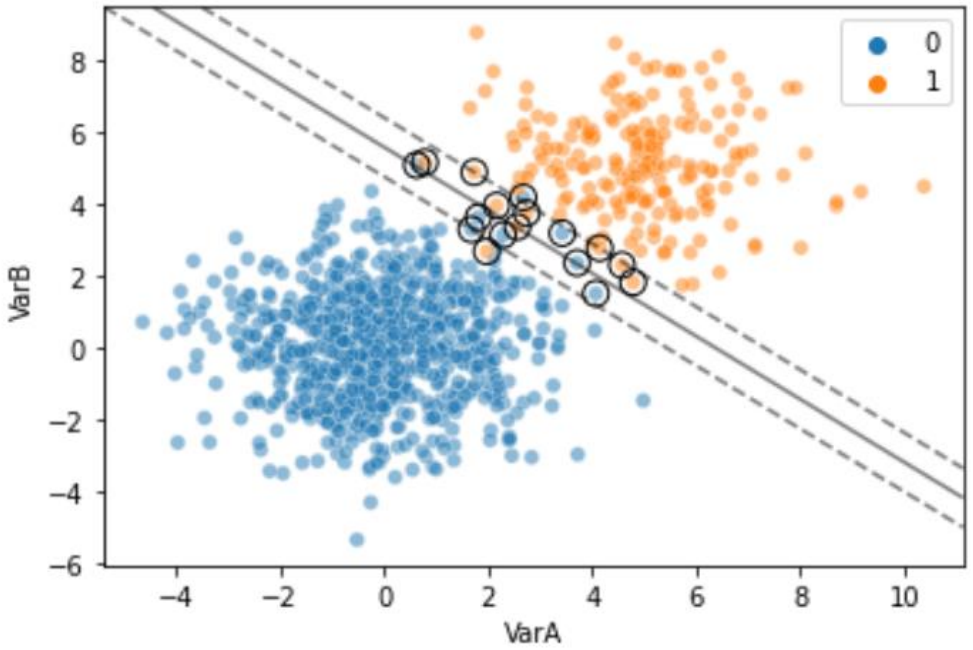
https://scikit-learn.org/stable/modules/svm.html#mathematical-formulation

Support vectors – the templates



In real life, if the data is not perfectly separable, the support vectors will fall within the separation boundary

Support vectors – the templates

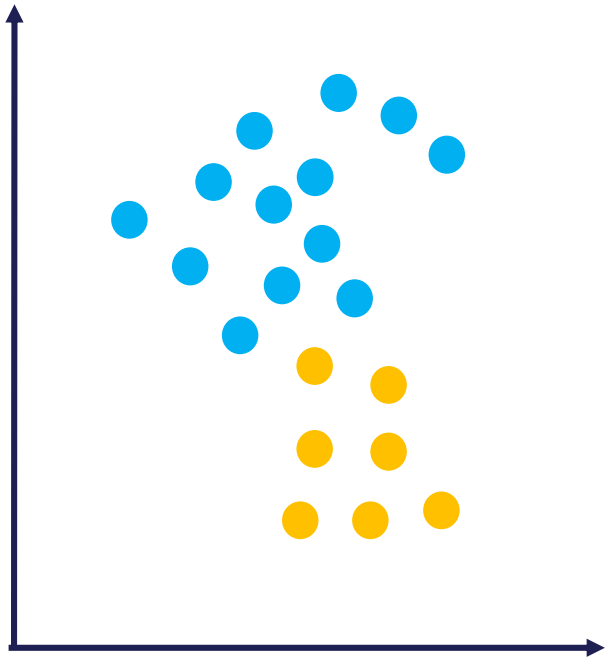


Step 1:

- With a SVM find the support vectors
- Select the support vectors from the minority class
- These are the templates

Inter- vs extrapolation

Graph taken from Nguyen, Cooper, Kamei, “Borderline over-sampling for imbalanced data classification,” International Journal of Knowledge Engineering and Soft Data Paradigms, 3(1), pp.4-21, 2009.



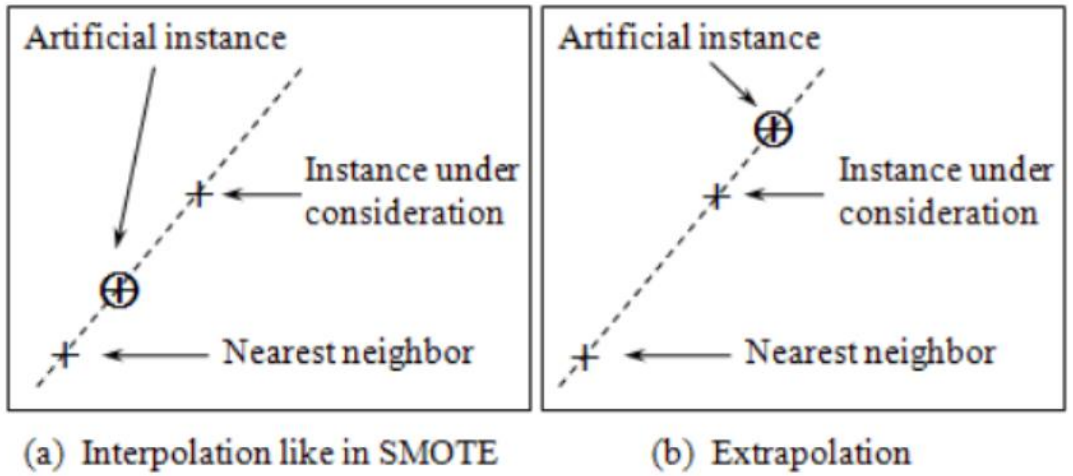
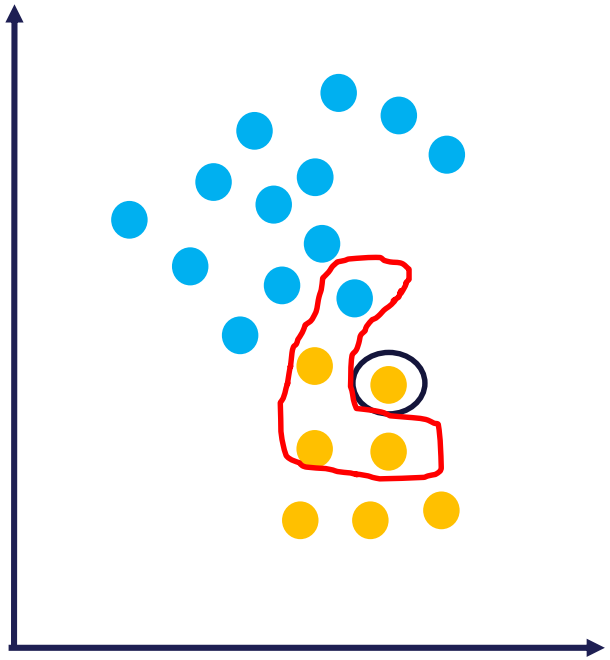


Fig. 2. Utilize interpolation (a) and extrapolation (b) techniques to create artificial instances.

- Train a KNN algorithm on the **entire** dataset
- Usually to find the 10 closest neighbours (m)

Inter- vs extrapolation

Graph taken from Nguyen, Cooper, Kamei, “Borderline over-sampling for imbalanced data classification,” International Journal of Knowledge Engineering and Soft Data Paradigms, 3(1), pp.4-21, 2009.



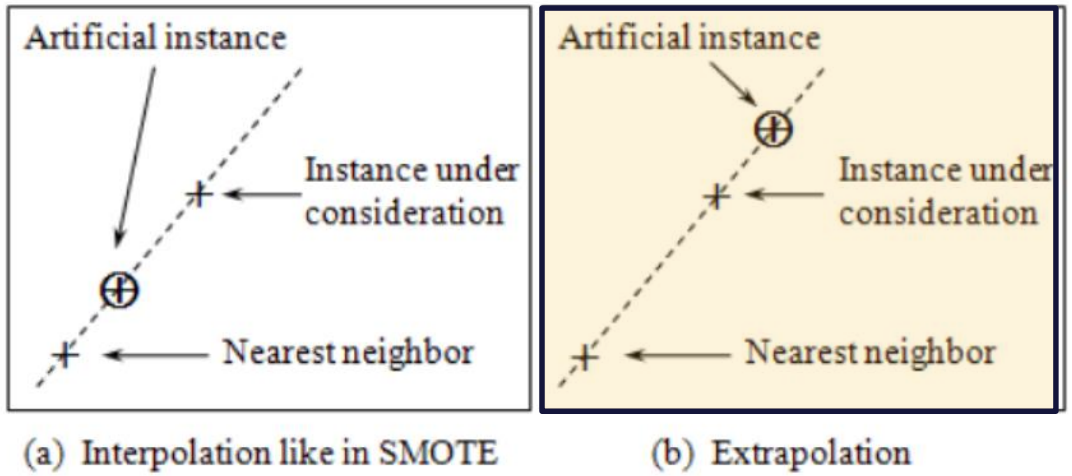


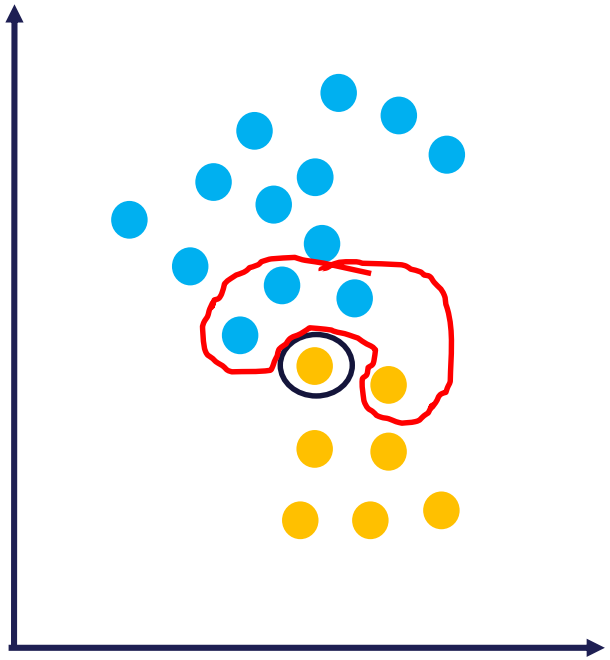
Fig. 2. Utilize interpolation (a) and extrapolation (b) techniques to create artificial instances.

If most neighbours from the minority ➔ extrapolation

Expand the boundary, goes “beyond” the sample

Inter- vs extrapolation

Graph taken from Nguyen, Cooper, Kamei, “Borderline over-sampling for imbalanced data classification,” International Journal of Knowledge Engineering and Soft Data Paradigms, 3(1), pp.4-21, 2009.



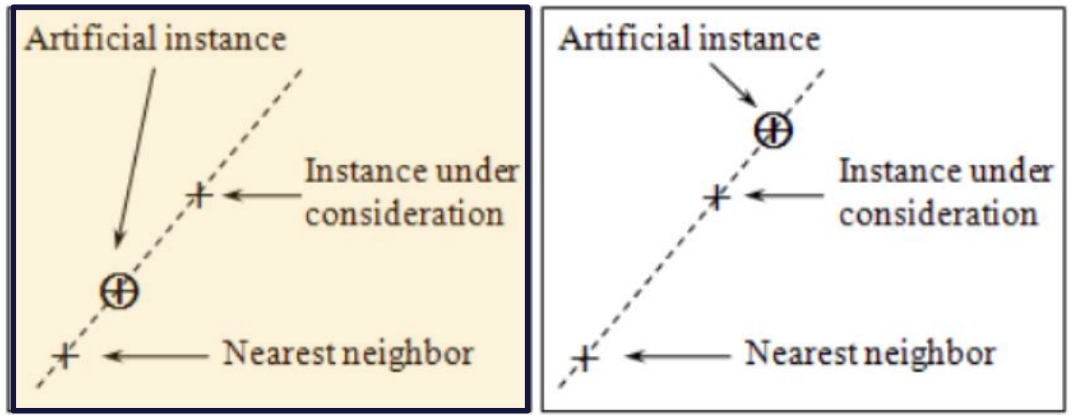


Fig. 2. Utilize interpolation (a) and extrapolation (b) techniques to create artificial instances.

If most neighbours from the majority ➔ interpolation

New sample is created within the 2 existing samples, stays within the boundary

Extrapolation

$$\text{new sample} = \text{support vector} + \text{factor} * (\text{support vector} - \text{neighbour})$$

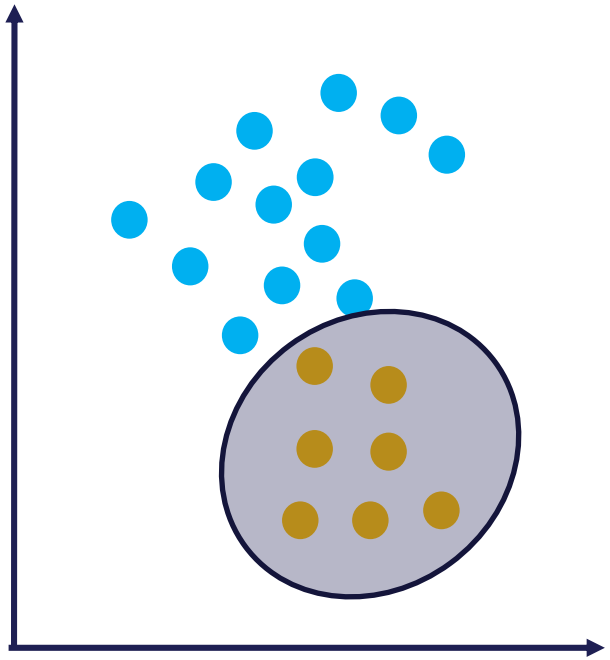
- Support vector and neighbour are from minority class
- Factor is a value taken at random between 0 and 1
- Differently from SMOTE, the neighbour is not chosen at random, it selects from closest to furthest neighbour in order.

Interpolation

$$\text{new sample} = \text{support vector} + \text{factor} * (\text{neighbour} - \text{support vector})$$

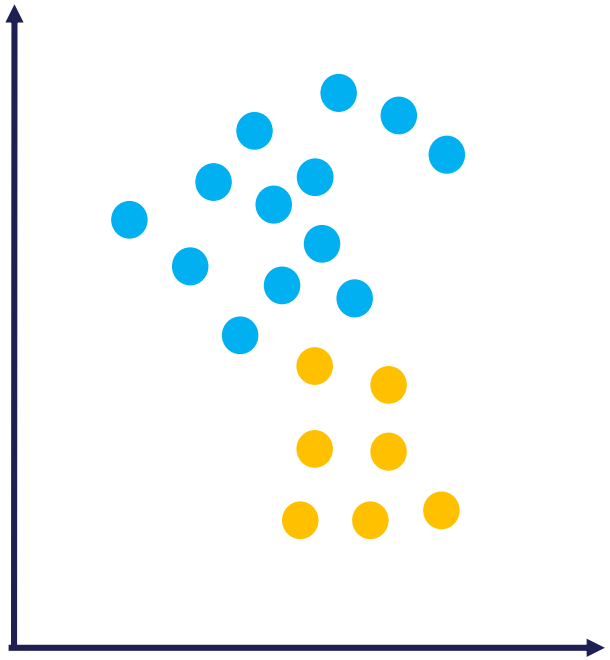
- Support vector and neighbour are from minority class
- Factor is a value taken at random between 0 and 1
- Differently from SMOTE, the neighbour is not chosen at random, it selects from closest to furthest neighbour in order.

Find the neighbours



- Another KNN is trained only on the minority group
- Usually to find the 5 closest neighbours

Find the neighbours



For SVM SMOTE, we fit 1 SVM and 2 KNNs

Imbalanced-learn: SVM SMOTE

```
sm = SVMSMOTE(
    sampling_strategy='auto',  # samples only the minority class
    random_state=0,  # for reproducibility
    k_neighbors=5,
    m_neighbors=10,
    n_jobs=4,
    svm_estimator = svm.SVC(kernel='linear')
)

X_res, y_res = sm.fit_resample(X, y)
```

To find the neighbours of the minority

To decide if we inter or extra-polate

Train In Data

THANK YOU

www.trainindata.com