

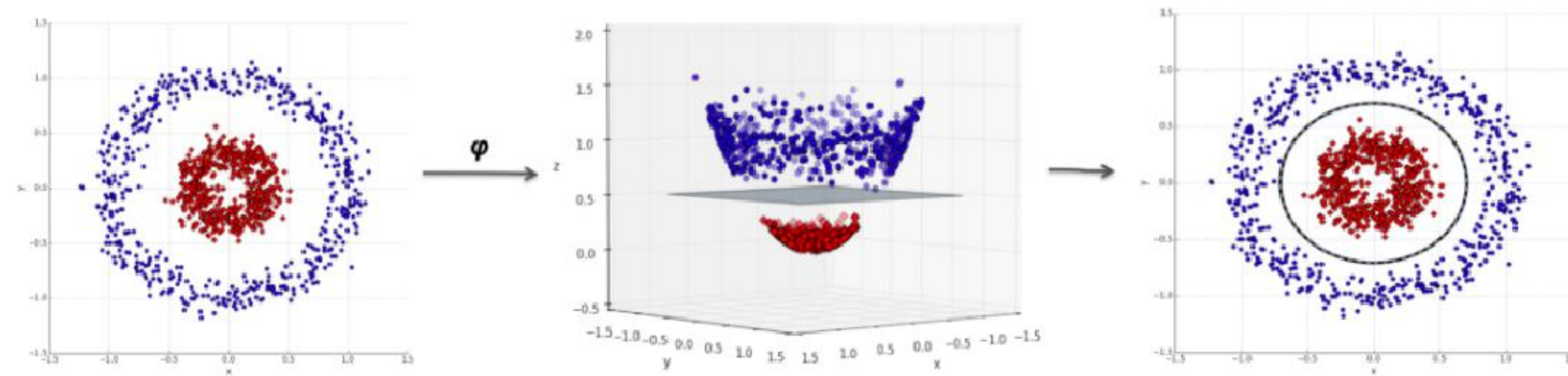
# Practice 9

## *Kernel SVM*

---

# Problem

- Use kernel SVM in Scikit-Learn library
- Predict whether each data point was extracted from facial skin image or not.
  - Use predefined function in `sklearn.svm`



# Dataset

## ➤ Dataset description

- The dataset is constructed over Blue, Green, Red color space.

## ➤ 3 Features

1. Blue
2. Green
3. Red
4. Skin image(=1) or not(=2)

❖ The last column of the dataset indicates the class labels.

\* UCI Machine Learning Repository :

<http://archive.ics.uci.edu/ml/datasets/Skin+Segmentation>

➤ You can download the pre-processed train and test dataset on i-campus

## Practice 9

1. Compare accuracy, F1 score and confusion matrix of linear SVM and kernel SVM.
2. Use predefined classes in *sklearn.svm.SVC*
  - Configure “kernel” parameter of SVC class to set the type of SVM model.
  - Linear SVM : *kernel = “linear”*
  - Kernel SVM : *kernel = “rbf”*

## Practice 9

### 3. How to train the model using RDD data format

- Before training the model, you need to save data into your memory using *cache()* function.
- For example

```
trRDDs.cache()  
tsRDDs.cache()
```

- In this example, *trRDDs*: training data points(RDD) & *tsRDDs*: test data points(RDD)
- Then, you can easily train SVM model provided by scikit-learn using *fit()* function
- For example

```
Kernel = SVC(kernel="rbf")  
Kernel.fit(trRDDs.collect(), trY)
```

- In this example, *trY*: training data points' label

### 4. Due date: June 11th 23:59

# Submission

- You need to submit *result.txt* file
  - ✓ Write *F1 score, accuracy and confusion matrix of linear SVM*
  - ✓ Write *F1 score, accuracy and confusion matrix of kernel SVM*

```
Linear ACC: 0.9480, Kernel ACC: 0.9900
Linear F1score: 0.9252, Kernel F1score: 0.9847
Linear Confusion
99 25
1 375
Kernel Confusion
100 5
0 395
```

Windows

```
Linear ACC: 0.9480, Kernel ACC: 0.9900
Linear F1score: 0.9252, Kernel F1score: 0.9847
Linear Confusion
99 25
1 375
Kernel Confusion
100 5
0 395
~
```

Linux