

# Machine learning Assignment

## **Problem statement:**

“To build an image recognition model which is capable of identifying the pattern on a dress image.”

## **Dataset Information :**

1. This dataset contains links to images of women's dresses.
2. The corresponding images are categorized into 17 different pattern types.
3. Most pattern categories have hundreds to thousands of examples.

## **Per class data available:**

plain	8385
floral	2776
stripes	701
Polka dot	651
Other	513
tribal	507
squares	439
ikat	355
animal	352
geometry	342
cartoon	261
scales	112
chevron	96
letter_numb	82

houndstooth	66
stars	45
skull	19

### **Approach 1:**

We will be using transfer learning since we have fewer data.

#### **Using Bottleneck feature:**

##### **1. Vgg16**

We got a training accuracy of around 78% and test accuracy 60%. So it was overfitting since we had less data corresponding to some class rather than the majority of the class. To avoid the problem we need to get more data for the corresponding class having fewer data.

##### **2. Resnet50**

Same for resnet we got the same result train accuracy of 80% and test accuracy of 65%.

### **Approach 2:**

Using data augmentation and transfer learning with fine-tuning the model.

##### **1. Vgg16**

We got test accuracy 75%.

##### **2. Vgg19**

We got a test accuracy of 74.48%.

### 3. ResNet50:

We got a test accuracy of 60%.

#### **Final model :**

**Test accuracy: 75%**

**We can increase the accuracy by increasing the dataset for the corresponding class making it a balanced dataset and we can also try more pre-trained models like Inception, mobile net...**

#### **Image Similarity:**

**Task: “ Given a random query image we need to return the similar products.”**

#### **Approach:**

**We will be using cnn feature generated from Vgg16. By using these feature we will compare the pairwise distance between the random image and from our data and then sorting the value with the nearest distance to image.**

**The package can be found from sklearn-**

**! from sklearn.metrics import pairwise\_distances**