# CS63 Fall 2020
# Lab 6: Convolutional Neural Networks

Mavis Gao, Yana Yuan

## 1 Data Set

The data set we used to train our convolutional network is called fashion and it is a MNIST data set. It includes training images and testing images to train our ConvNet in identifying the type of the cloth in the image. In our case, we used 60000 training images and 10000 test images, each of size 28*28. The categories are t-shirt, trouser, pullover, dress, coat, sandal, shirt, sneaker, bag and boot.

## 2 Network

The general design of our convolutional neural network is an input layer, a hiddden layer, and an output layer. The hidden layer is composed of 3 convolutional 2D layers, a max pooling 2D layer, a flatten layer, and a dense (fully connected) layer.

More specifically, our best performing network has an ascending number of filters in each of the convolutional layer. Details of the number of filters in the summary below.

```
net = Network(neural_net)
net.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1 (Conv2D)              (None, 26, 26, 40)        400

 conv2 (Conv2D)              (None, 24, 24, 60)        21660

 conv3 (Conv2D)              (None, 22, 22, 80)        43280

 pool1 (MaxPooling2D)        (None, 11, 11, 80)        0

 flatten1 (Flatten)          (None, 9680)              0

 hidden1 (Dense)             (None, 100)               968100

 output (Dense)              (None, 10)                1010

=================================================================
Total params: 1,034,450
Trainable params: 1,034,450
Non-trainable params: 0
```

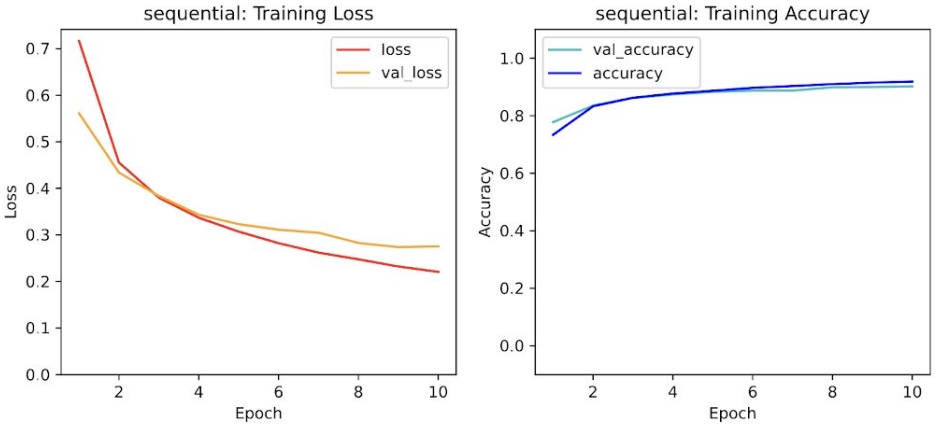
Figure 1: Summary of our convolutional neural network. The size and number of filters in each convolutional layer is shown in the "Output Shape" section.

Thoughts on why our ConvNet performed well:
**Increasing filters in Conv2D layers:** The increasing filters in convolutional layers from 40 to 80 allows the network to learn increasingly complex and higher-level features from the input image. As our input images include complex patterns in different kind of clothes, the network is capable of performing simple tasks such as identifying edges and textures to complicated tasks such as identifying the shape of high heels and long sleeves.
**MaxPooling 2D layer:** We tried different size and different number of pooling layers and a pool size of (2,2) seemed to perform well, especially at reducing the problem of overfitting. A pooling layer could reduce the spatial dimensions and reduce computational cost as well.
**Dense layer:** The hidden dense layer allows our ConvNet to learn non-linear combinations of the higher-level features so that it can perform better at identifying complicated patterns.

# 3 Training

| Trial | Validation Accuracy | Validation Loss |
|---|---|---|
| 1 | 0.9015 | 0.2769 |
| 2 | 0.9021 | 0.2754 |
| 3 | 0.9046 | 0.2780 |
| 4 | 0.9021 | 0.2881 |
| 5 | 0.9011 | 0.2792 |
| Average | 0.9023 | 0.2795 |

Table 1: Validation results of five trials.




```
Epoch 10/10 loss: 0.2206946313381195 - accuracy: 0.918833315372467 - val_loss: 0.2753925919532776 - val_accuracy: 0.9021000266075134
```

Figure 2: Learning graphs of the second trial.

# 4 Evaluation

The network is good at identifying:
1. Trousers: with 974 true positives and few miss classifications
2. Bags: 976 positives
3. Ankle boots: 956 positives

The network is bad at identifying:
1. Shirts: with 346 shirts that are missed, which is confused with T-shirts, pullovers, and coats
2. Pullovers: with 214 instances of missed pullovers

Easiest categories to learn: Trouser, Bags, Ankle boots
Hardest categories to learn: Shirts


Figure 3: Feature maps of three Convolutional layers and one MaxPooling layer.

**Speculation on why this is the case:** We think this might be the case because trousers have distinctive patterns, such as long, straight legs, and bags have unique shoes with a large block, while shirts have similar patterns with other objects such as pullovers and T-shirts, which is hard

for the network to capture the subtle features, and the variability of the shirt itself could contribute to the misclassification.

**Description and analysis:** The model achieved an accuracy of 92.38 percent on the training set and 90.15 percent on the test set in the first trial, which is relatively high; and based on the confusion matrix, the matrix seem to confuse shirts, T-shirts, and pullovers, this could be due to similar textures or patterns that the clothes have and the filter of the network could not identify, and the least number of incorrect classifications were trousers and bags, which implies that the network's filters could identify the patterns of these categories.

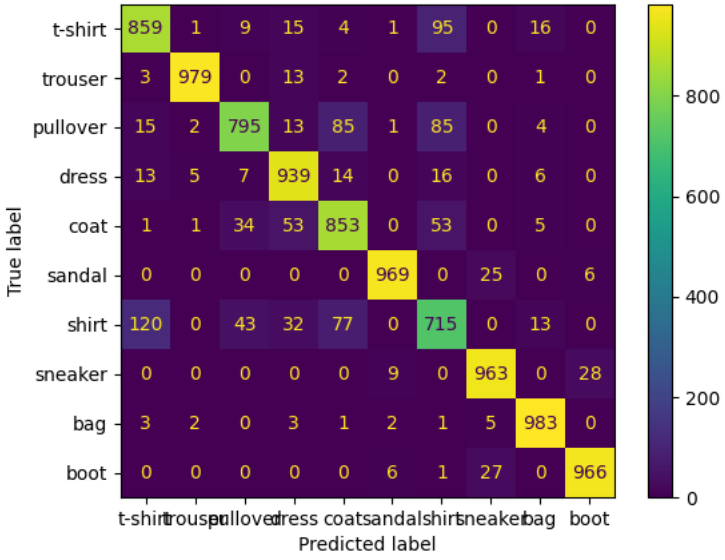



Figure 4: Matrix indicating what categories the ConvNet was good and bad at identifying. This matrix shows the result of Trial2.