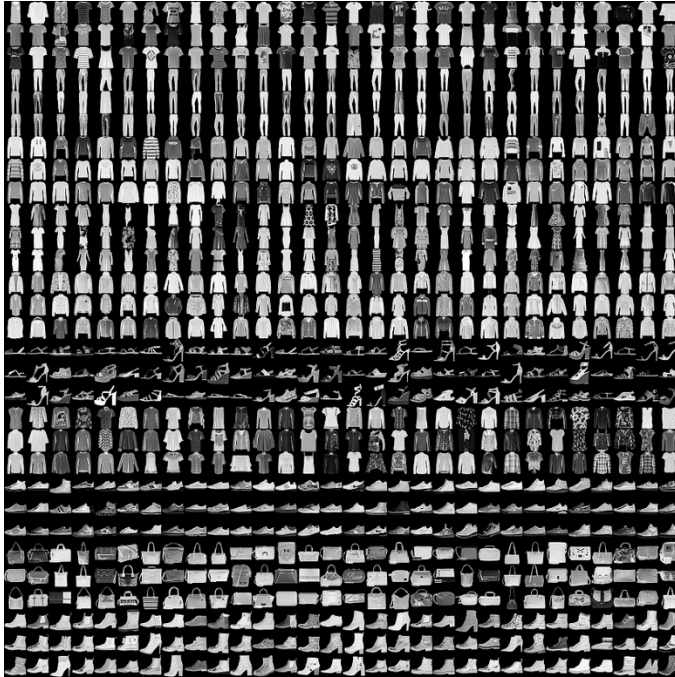


Ve581 Convolutional Neural Networks for Visual Recognition

Summer 2019

Homework #1 (Due 7/4, Thursday)



Label	Description
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot

Note:

- In Homework #1 mini-project assignment, you will build and train your “Hello, world” machine learning programs of computer vision based on Fashion MNIST dataset (<https://github.com/zalandoresearch/fashion-mnist>). Fashion MNIST is intended as a drop-in replacement for the classic MNIST dataset.
- Although we’ve mainly introduced TensorFlow, you are free to choose any other API to build your model. Please make sure to add comments to your code whenever needed.
- Please submit your Jupyter notebook or python code to Canvas.

1. (50 points)

Please build and train a simple deep neural network (DNN) to classify the 10 image classes as shown above following the steps below:

- (1) (5 points) Import Fashion MNIST dataset
- (2) (5 points) Explore (and preprocess if needed) your data
- (3) (10 points) Setup three layers: input layer, hidden layer with 128 units using ReLu, and output layer
- (4) (10 points) Compile your model: choose your loss function, optimizer, and metrics
- (5) (10 points) Train your model
- (6) (10 points) Evaluate the performance and make prediction based on 10 randomly selected testing images

2. (50 points)

Please build and train a convolutional neural network (CNN) to classify the 10 image classes as shown above, to obtain a higher performance than the DNN model, following the steps below:

- (1) Import Fashion MNIST dataset
- (2) Explore (and preprocess if needed) your data
- (3) (10 points) Setup the necessary layers: convolutional layer, pooling layer, dense layer, output layer
- (4) (10 points) Compile your model: choose your loss function, optimizer, and metrics
- (5) (10 points) Train your model
- (6) (10 points) Evaluate the performance and make prediction based on 10 randomly selected testing images
- (7) (10 points) Applying error analysis for your model and provide suggestion to improve your model's performance