David Holcer and Kusha Sareen

Fashion-MNIST Challenge Deliverable

1. Implementation

We wrote a CNN to classify the images with the following structure:

```
model_cnn = Sequential()
model_cnn.add(Conv2D(32, (3, 3), padding='same', activation='relu',
              input_shape=(28, 28, 1)))

model_cnn.add(MaxPooling2D((2, 2), strides=2))

model_cnn.add(Conv2D(64, (3, 3), padding='same', activation='relu'))
model_cnn.add(MaxPooling2D((2, 2), strides=2))


model_cnn.add(Flatten())
model_cnn.add(Dense(256, activation='relu'))
model_cnn.add(Dropout(0.2))
model_cnn.add(Dense(128, activation='relu'))
model_cnn.add(Dropout(0.2))
model_cnn.add(Dense(10, activation='softmax'))
```

2. Results

We were able to achieve a 90% training accuracy over 5 epochs. Our best submission (submitted after the end of the contest) had a test accuracy of 88.4%.

3. Challenges

It was challenging initially to learn how to use keras and the tensorflow library to implement a neural network. The documentation, however, was very insightful. Preprocessing the data into the format accepted by the model was also difficult to begin with. Our greatest challenge was trying to modify the model in a way to beat the CNN baseline and increase our test accuracy beyond 90%.

4. Conclusion

This assignment was a great way to learn to implement a CNN on a real dataset. We developed skills doing the preprocessing/postprocessing for the model. It was interesting to see how modifying the hyperparameters of the network and adding different types of layers affected the model accuracy.

5. Individual Contribution

Both team members discussed ideas, drafted an implementation and wrote the code in a collaborative manner.