**Technical Report**

**Background**

MNIST database is a legacy database of handwritten digits that are commonly used for training image processing systems. Fashion-MNIST is a dataset of Zalando article images—consisting of a training set of 60,000 Each image is a 28x28 grayscale image, associated with a label from 10 classes.Each image is 28 pixels in height and 28 pixels in width, for a total of 784 pixels in total. Each pixel has a single pixel-value associated with it, indicating the lightness or darkness of that pixel, with higher numbers meaning darker. This pixel-value is an integer between 0 and 255.

**Labels**

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

Column 1 is the class labels. - Remaining columns are pixel numbers (784 total). Each value is the darkness of the pixel (1 to 255)

**Purpose of the report**

* To predict and classify images from the labels
* Regression analysis

**Report**

* Iextractedthe **fashion – Mnist** dataset from Kaggle as CSV file. I divided the dataset into training and testing with a ratio of (50:50)
* For feature extraction, I used hog features from skimage library. For data transformation, I created a data frame of all the features from X-train section by reshaping pixels into 28 \* 28 ending up in 81 columns.
* I ran the logistic regression model on X-train and calculated scores on Y\_train using the model.
* Similarly, I created hog features from Test section and ended up the same size as Train as I divided the dataset with a ratio of 50:50
* I made my predictions on Y\_test and compared the scores using confusion matrix.
* Transformed the actual data frame by retrieving mean-median features and appended to the original data frame, which ended up with 83 columns( 81 features from before and added 2 new columns for mean and median)
* Applied the similar technique of running logistic model on X \_train and Y\_ train as before running my models on test sets for both.
* Compared the scores and printed the confusion matrix for results.
* Made predictions using decision tree classifier and random forest classification techniques.

Predictions:

**1 Model**

Accuracy scores - 0.76695889862995437

**2 Model**

**Accuracy Scores -** 0.76462548751625059

**3 Model**

Accuracy Scores - 0.876795893196

**Conclusion:**

The model did great for more than half of the labels compare to the baseline accuracy which was 10%. After Evaluating the confusion matrix, the model did fail or gave low scores for label 0 and 6. This tells us that experimenting with other features might help us with classification problem.