Créer une fonction moyenne d’abord pour faire ça et un tableau croisé des nombres en lignes et colonne pour les pixels

Avec la moyenne pour chaque nombre des pixels

comparatif

Les numéros semblables sont censés se ressembler et avoir des pixels proches. Par exemple, pour chaque colonne de pixel, évaluer la distance entre la différence entre le nombre et la moyenne relative à chaque nombre.

A chaque colonne, ajouter un +1 pour la colonne

Classification banale d’abord.

On va faire le K- means mais comment ??????

knn\_final = KNeighborsClassifier()

knn\_final = KNeighborsClassifier(n\_neighbors=4,weights='distance',p=1, n\_jobs=-1)

knn\_final.fit(mnist\_features\_prepared, mnist\_labels)

KNeighborsClassifier(n\_jobs=-1, n\_neighbors=4, p=1, weights='distance')

test\_copy = test.copy()

test\_prep = num\_pipeline.transform(test\_copy) # Transform data for kneighbors

mnist\_submission = pd.DataFrame(columns=['ImageId','Label'])

test\_prep\_df = pd.DataFrame(data=test\_prep)

# fill submission file

i=0

for row in test\_prep:

i = i+1

index = i

label = knn\_final.predict(row.reshape(1,-1))

mnist\_submission = mnist\_submission.append({'ImageId' : index , 'Label': label} , ignore\_index=True)

pass

# Preparing the submission file

mnist\_submission.ImageId = mnist\_submission.ImageId.astype(int)

mnist\_submission.Label = mnist\_submission.Label.astype(int)

mnist\_submission.to\_csv("mnist\_submission.csv",index=False)

mnist\_submission

#Modifier le mnist