

*Deus
Ex
Machina*

Victor Espidol and Adam
Novak



Overview of Pipeline

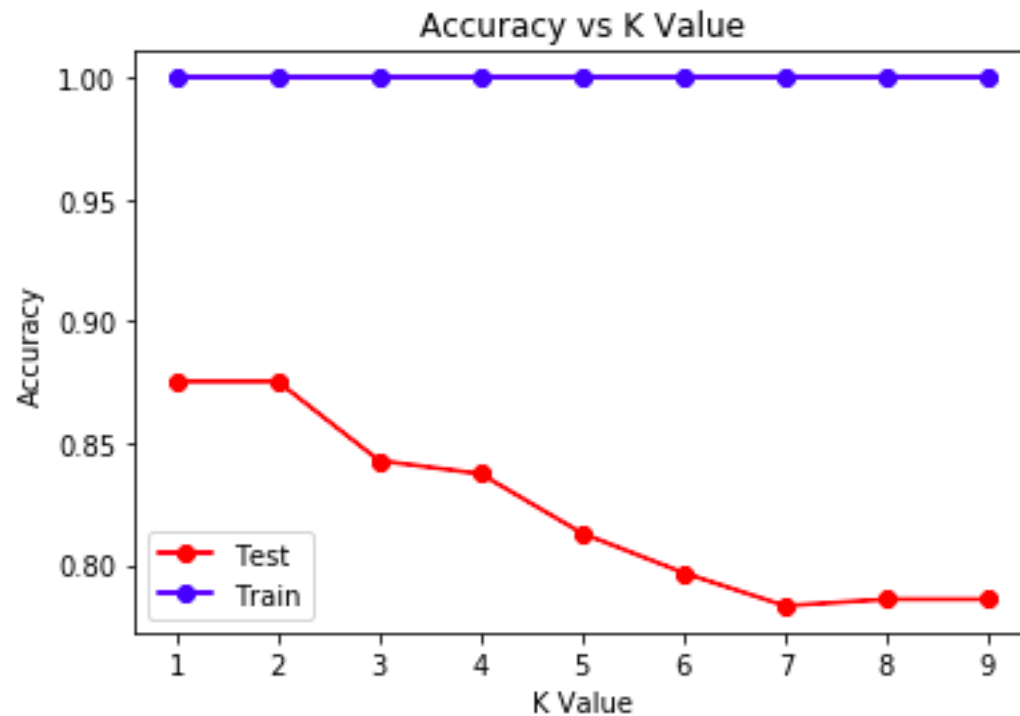
- ◇ Ready the data
 - ◇ Split data to training and testing
 - ◇ Standardize / Normalize data
 - ◇ Gray scale and flatten data
- ◇ Attempted to use four different classifiers to determine which is best
 - ◇ K-Nearest Neighbors
 - ◇ Random Forests
 - ◇ Multilayer Perceptron
 - ◇ Convolutional Neural Network
- ◇ Perform Accuracy Checks on each model

Model #1: K-NN

- ◇ Easy to implement and understand
 - ◇ Compares associated neighbors
- ◇ Parameters Tweaked:
 - ◇ Number of neighbors evaluated 1-10
- ◇ Unweighted neighbors
- ◇ Evaluated with Euclidean Distance

K-NN Performance

Training Accuracy w/ k = 1: 1.0
Test Accuracy w/ k = 1: 0.8753387533875339



◇ Downfalls

◇ Overfitting

◇ Could not get test accuracy > 90%.

Random Forests

- ◆ Constructs many decision trees during training
 - ◆ Then outputs the class of the individual trees
- ◆ Parameters Changed
 - ◆ Number of estimators
 - ◆ 1, 5, 10, 20, 50, 100, 500

Random Forests Performance

- ◆ Best accuracy achieved with 100 estimators
 - ◆ 85% accuracy on test data
- ◆ Downfalls
 - ◆ No matter how parameters tuned, we could not get higher than 85%

MLP

- ◆ Feedforward artificial neural network
 - ◆ Involving multiple layers of perceptrons with activation functions
- ◆ Input layer, hidden layers, output layers
- ◆ Tested with tanh and logistic activation
- ◆ Tested with sgd and adam solver
- ◆ Best combination: logistic activation with sgd solver

MLP Performance

- ◇ Test data yielded 81% accuracy
- ◇ Downfalls
 - ◇ Many parameters need to be fine tuned
 - ◇ Run time was very long
 - ◇ Not enough time to play around to increase score

CNN

- ◇ Construct a feature hierarchy by combining low level features to create a high-level feature
 - ◇ Such as blobs and edges
- ◇ Very popular for image processing
 - ◇ Thus invested time into this

CNN - Nets

- ◇ Used different nets to determine which accuracy we could get
- ◇ Net 1:
 - ◇ Used tanh activation and linear transformation
- ◇ Net 2:
 - ◇ Same as Net 1 with inclusion of dropout and batch normalization
- ◇ Net 3:
 - ◇ Same as Net 1 but uses ReLU for activation instead
- ◇ Net 4:
 - ◇ Same as Net 4 but uses ReLU for activation instead

CNN - Performance

All Characters

- ◇ Net 1 - 86%
- ◇ *Net 2 - 90%*
- ◇ Net 3 - 74%
- ◇ Net 4 - 71%

Easy Characters

- ◇ Net 1 - 93%
- ◇ *Net 2 - 98%*
- ◇ Net 3 - 83%
- ◇ Net 4 - 81%

Accuracy Comparison

Normal Characters

Classifier	Training Accuracy	Test Accuracy
K-NN	100%	87.5%
Random Forests	100%	85%
MLP	98%	81%
CNN	97%	90%

Easy Characters

Classifier	Training Accuracy	Test Accuracy
K-NN	100%	92%
Random Forests	100%	89%
MLP	99%	88%
CNN	100%	98%

Thank you, questions?