|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Error Rates | | Data Sets | | | | | |
| X1 | X2 | X3 | X4 | X5 | X\_test |
| Classifier | SVM | .032 | .128 | .105 | .157 | .126 | .104 |
| KNN | .107 | .054 | .091 | .148 | .098 | .084 |
| Logistic Regression | .113 | .106 | .006 | .147 | .107 | .098 |
| Decision Tree | .325 | .317 | .311 | 0 | .315 | .332 |
| Random Forest | .122 | .107 | .123 | .134 | 0 | .122 |

Overall KNN had the best performance on the test dataset and consistently did better than most other classifiers on the bagged data. It would appear that decision trees and consequently random forests, performed very poorly on the datasets they weren’t trained on in comparison to the training data. SVM was very consistent across all datasets, but just didn’t perform as well as KNN.

Overall I would say that bagging helps to show what classifiers are going to be overfit to the training data as is evident with decision trees and random forests.

2.

For image 1 and image 2, for a constant low valued K the changes in iterations and R values, produced minimal differences between pictures, at most a handful of pixels.

Examples

A panda bear in the grass

Description automatically generatedImg1\_K3\_iters7\_R5 Img1\_K3\_iters20\_R25

A panda bear in the grass

Description automatically generated

Img2\_K3\_iters7\_R5 Img2\_K3\_iters20\_R25

A pixelated bird on a branch

Description automatically generated A pixelated bird on a branch

Description automatically generated

Again very little difference, but for higher values of K, as the iterations and Rs increase, the resultant images get closer to the original reference picture

Img1\_K7\_iters7\_R5 Img1\_K7\_iters20\_R25

A panda bear in the grass

Description automatically generated A panda bear in the grass

Description automatically generated

Img2\_K7\_iters7\_R5 Img2\_K7\_iters20\_R25

A red bird on a branch

Description automatically generated A red bird on a branch

Description automatically generated

As you can see increasing iters and R helps to clarify areas of finer detail or gradual gradient, such as the cardinal’s wings, the plants around the panda, and the sky behind the cardinal

For image 3, the performance is poor overall. The complexity of colors in the original image means that a large amount of K clusters (more than 7) is needed to accurately replicate the original image.

Even at K=r7, iters=20 and R=25, the resulting image fails to capture the color complexity seen in the original

A pixel art of a city

Description automatically generatedA city skyline with trees and water

Description automatically generated

As you can see, there is no green on the shoreline and the red/orange accents on some of the buildings on the right are not captured.

For fun I did two additional pictures, both with iters=20 and R=25, but with K=15 and K=25

A city skyline with a bridge over water

Description automatically generatedA city skyline with a bridge over water

Description automatically generated