SVM in R

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **200** | | **400** | | **800** | | **1600** | |
|  | **Runtime** | **MSE Loss** | **Runtime** | **MSE Loss** | **Runtime** | **MSE Loss** | **Runtime** | **MSE Loss** |
| **Linear SVM** | 1.42s | 6.245846 | 2.4s | 7.023637 | 3.4s | 7.825877 | 11.17s | 3.220385 |
| **Non-Linear SVM** | 1.7s | 5.254215 | 2.87s | 2.585515 | 5.61s | 0.7079316 | 14.25 | 0.236146 |

Two methods of SVM were used, linear SVM and non-linear (Gaussian) SVM. Non-linear SVM consistently outperformed Linear SVM across all sample sizes. The MSE loss for non-linear SVM decreases as the sample size increases with the lowest loss achieved with the 1600 dataset. The performance of linear SVM decreases using the 400 and 800 samples sizes and then starts to improve again using the 1600 sample size.

Linear and non-linear SVM were implemented in RStudio v 1.3.1093 using the caret package, as the machine learning suite included in Mathematica 12.0 does not include SVM. The models were not tuned using the cross-validation function to align with the process followed in Mathematica. The non-linear model was trained using the radial basis function kernel and the linear model was trained using the linear kernel function.