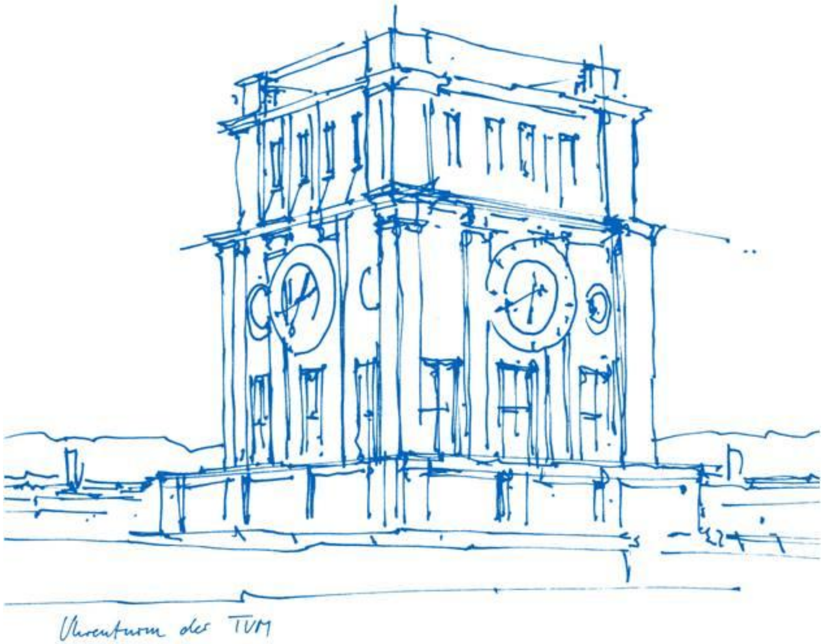
Anomaly Detection Practical – Challenge I

Hamza Tahir & Muhammad Hamza Usmani

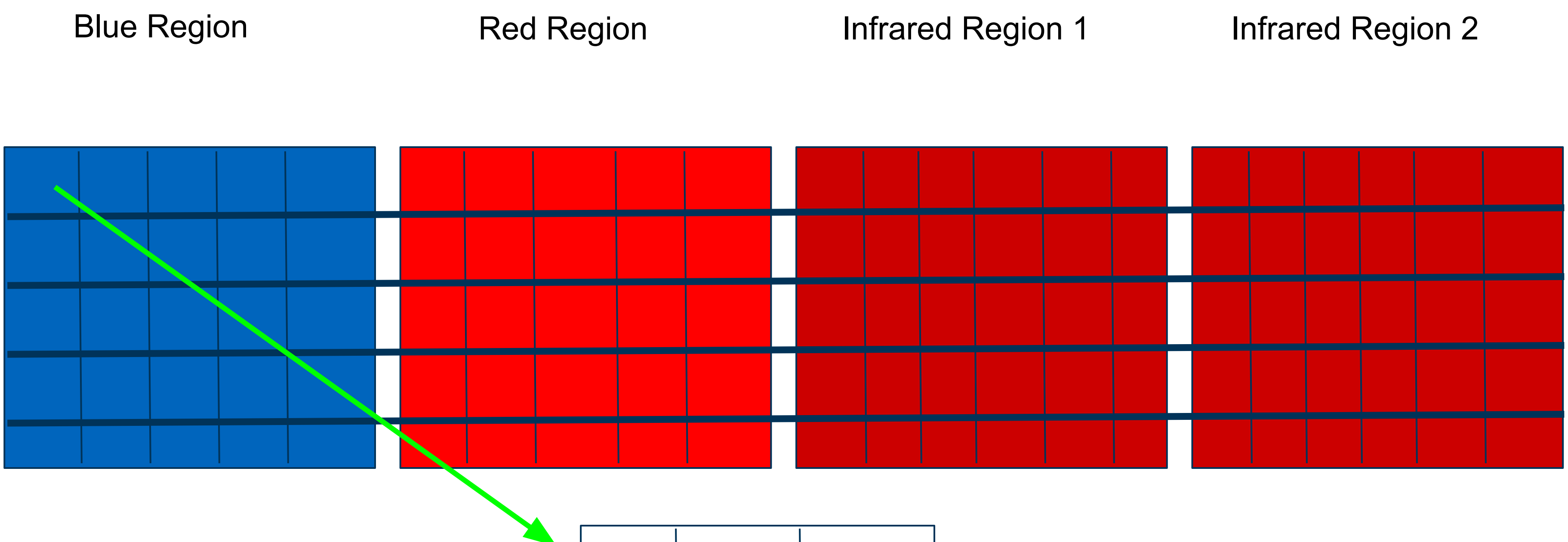
Technische Universität München

Fakultät für Informatik

8 November 2016



Data



92	86	93
93	90	93
70	75	84

Data Preprocessing

Feature Mean: The missing value of a particular feature is replaced by average value of of the feature. This averaging technique is generic to replace missing values for all kinds of datasets.

Feature Maximum: The missing value of a particular feature is replaced by the max value of the feature.

Nine-Neighborhood Mean: Missing values were replaced by averages in a particular nine-neighborhood.

Immediate Neighborhood Mean: This approach is similar to the Nine-Neighborhood one but here image locality is leveraged to better replace missing values. The underlying assumption is that closely related pixels have similar values. Therefore, rather than using the entire 9 neighborhood, only 'immediate' neighbors are used for the average value. 'Immediate' neighbors are dened as the pixels that are to the left, right, top and bottom of the 9-neighborhood.

Data Preprocessing

Feature Weighting

Assumption: Not all features contribute equally to determine the class of a sample

- different features can have different weights that represent their relative importance to the classication model.
- In particular, we could not be certain that a certain spectral band was more or less important for the classication of cotton soil.
- It might be the fact that the green spectrum was more important, or maybe the infra-red spectrum.

Feature Weighting

Assumption: Not all features contribute equally to determine the class of a sample

- We used basic techniques based upon correlation to determine weights of the features.

- First, we extracted each feature out individually and trained our classifiers on each separately.

- The feature that had the lowest accuracy after 3-fold validation was then completely removed from the training of the model.

- However, this did **NOT** prove to be a good replacement model..

Techniques Used

1. Naive Bayes
2. Support Vector Classier (SVC)
3. K-Nearest Neighbors (KNN)
4. Decision Trees
5. Neural Network - Multi Layer Perceptron
6. Random Forest

Used 3-fold cross validation for each to calculate the **training error**.

Results: Training Error with 3-Cross Validation

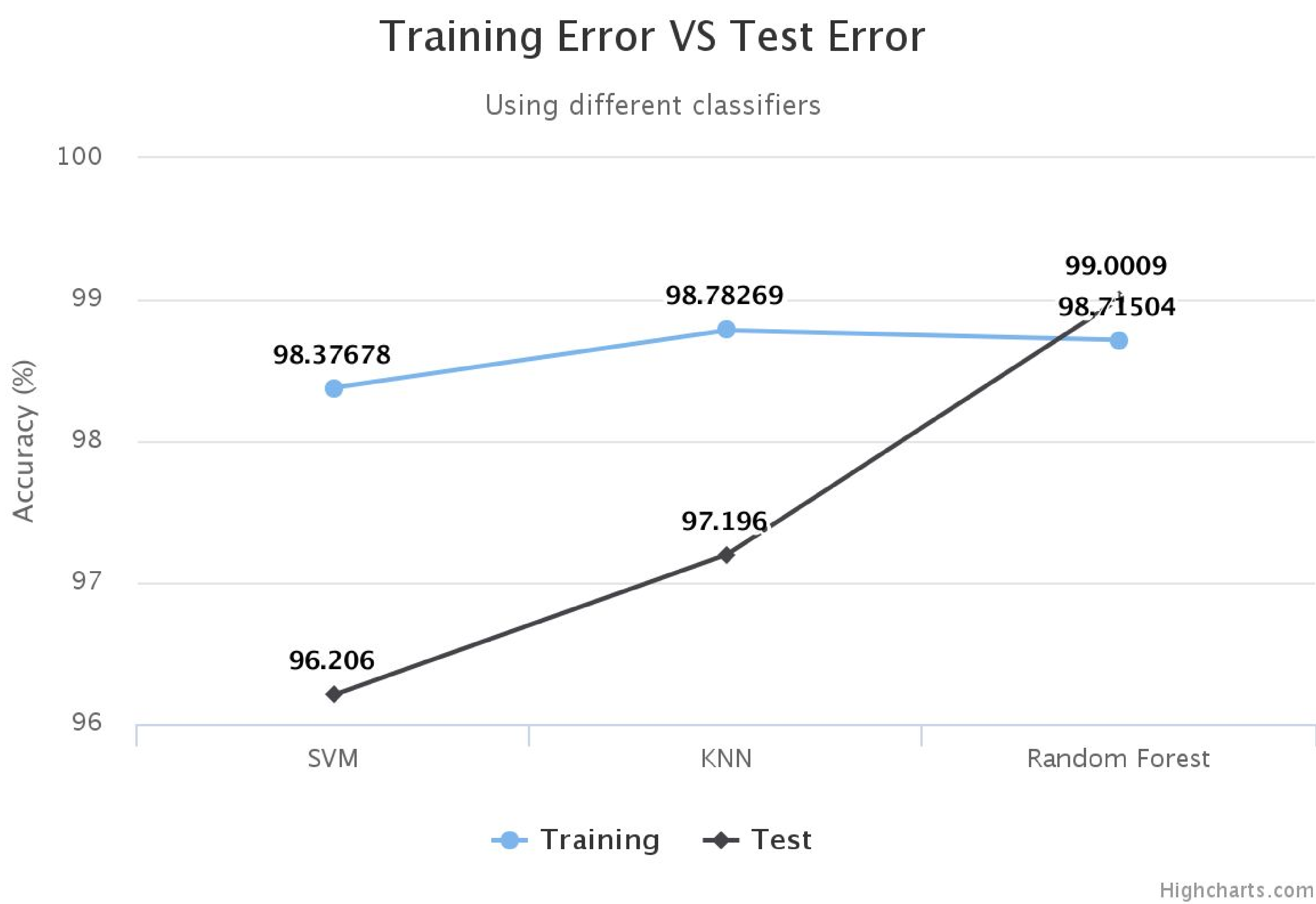
Table 2. Best Average Cross-Validation Accuracies (without feature weighing)

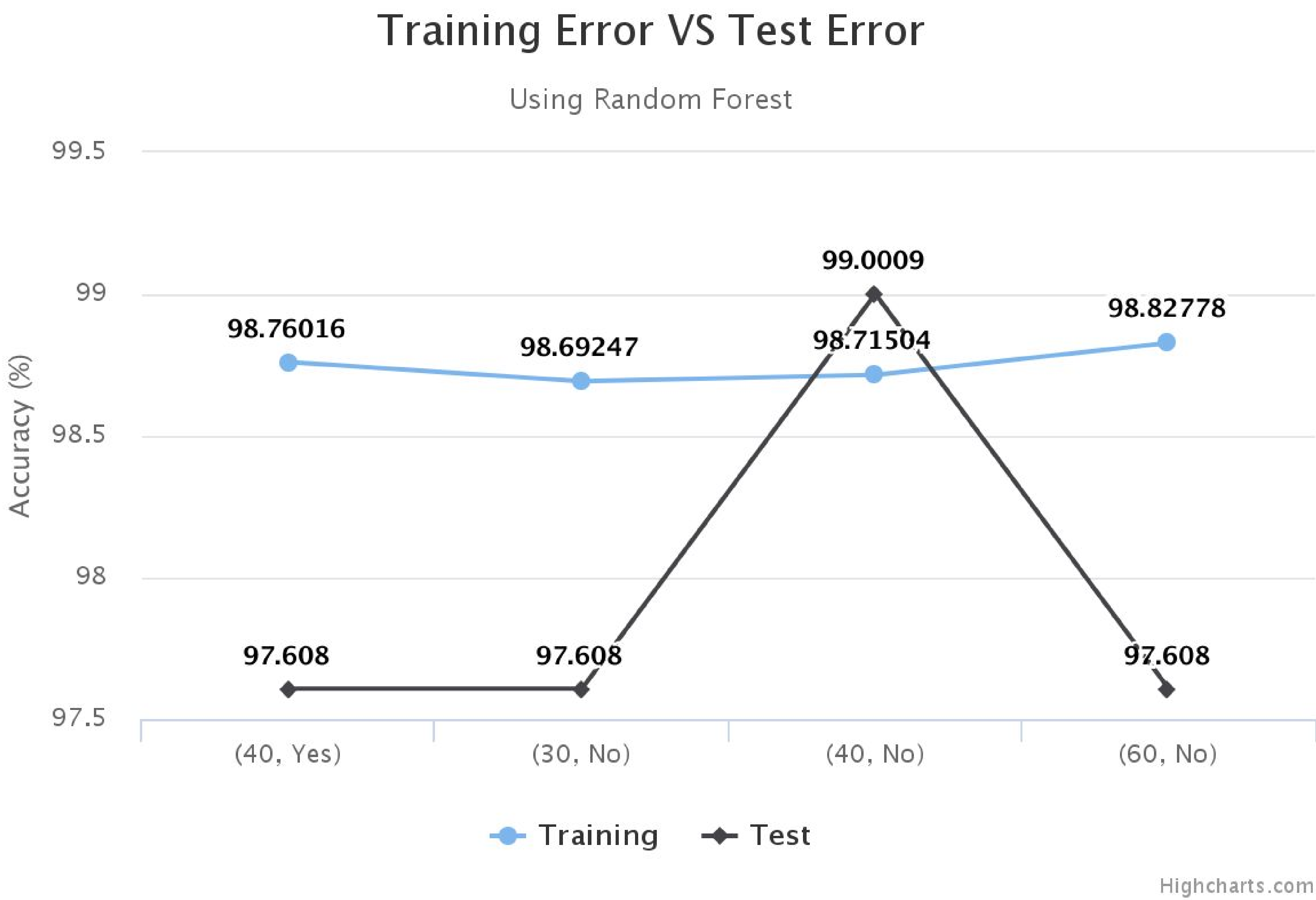
Training Accuracies				
Technique	Feature Mean	Feature Max	Nine-Neighbor	Immediate-Neighbor
Decision Trees	97.519	98.528	96.550	97.723
Multi-Layer Perceptron	96.280	97.768	97.768	98.016
Naive Bayes	98.422	96.979	98.535	98.512
KNN	98.858	96.731	98.557	98.783
SVC	97.813	94.295	97.858	98.377
Random Forest	98.647	97.610	97.970	98.715
Average	**97.796**	**96.985**	**97.873**	**98.354**

Results: Training Error with 3-Cross Validation

Table 3. Best Average Cross-Validation Accuracies (with feature weighing)

Training Accuracies				
Technique	Feature Mean	Feature Max	Nine-Neighbor	Immediate-Neighbor
Decision Trees	97.994	97.002	97.610	97.566
Multi-Layer Perceptron	97.250	95.942	97.565	97.632
Naive Bayes	98.377	96.397	98.557	98.512
KNN	97.881	96.619	98.693	98.85
SVC	97.565	94.656	98.197	98.377
Random Forest	98.625	98.197	98.715	98.625
Average	**97.948**	**96.468**	**98.222**	**98.260**


Training Error VS Test Error
Using different classifiers
100
99
98
97
96
Accuracy (%)
98.37678
98.78269
99.0009
98.71504
97.196
96.206
SVM
KNN
Random Forest
Training
Test
Highcharts.com


Training Error VS Test Error
Using Random Forest
99.5
99
98.5
98
97.5
Accuracy (%)
98.76016
98.69247
98.71504
98.82778
99.0009
97.608
97.608
97.608
(40, Yes)
(30, No)
(40, No)
(60, No)
Training
Test
Highcharts.com

Thank you!