Remote Sensing of Environment 204 (2018) 648–658

Contents lists available at ScienceDirect

Remote Sensing of Environment

journal homepage: www.elsevier.com/locate/rse

Effect of classifier selection, reference sample size, reference class distribution and scene heterogeneity in per-pixel classification accuracy using 26 Landsat sites

Shahriar S. Heydari, Giorgos Mountrakis*

Department of Environmental Resources Engineering, State University of New York, College of Environmental Science and Forestry, 1 Forestry Drive, Syracuse, NY 13210, United States

ARTICLE INFO

Keywords:
Land-cover mapping
Classifier accuracy assessment
Image classification
Image heterogeneity

ABSTRACT

A major issue in land cover mapping is classifier selection. Here we investigated classifier performance under different sample sizes, reference class distribution, and scene complexities. Twenty six 10 km × 10 km blocks with complete reference information across the continental US are used. Per-pixel classification took place using six spectral bands from Landsat imagery. The tested classifiers included Naïve Bayes (NB), Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Bootstrap-aggregation ensemble of decision trees (BagTE), artificial neural network up to 2 hidden layers, and deep neural network (DNN) up to 3 hidden layers. For the entire block, our accuracy assessment indicated that all classifiers, with the exception of NB (a Maximum Likelihood variant), performed similarly. However, when we concentrated on edge pixels (pixels at the border of adjacent land cover classes), it was clear that the SVM and KNN offer considerable accuracy advantages, especially for larger reference datasets. Because of their relatively low execution times SVM and KNN would be recommended for classifications using Landsat's spectral inputs and Anderson's 11-level classification scheme. However, both SVM and KNN demonstrated substantial accuracy degradation during the parameter grid search. For this reason, an exhaustive parameter optimization process is suggested. While the ANN and DNN neural network variants did not perform as well, their performance may have been restricted by the lack of rich contextual information in our simple six band per-pixel input space. The effect of class distribution in the training dataset was also evident on the calculated accuracy metric. Gradual accuracy degradation as edge pixel presence increased was also observed. Future work could focus on data-rich classification problems such as change detection using Landsat stacks or expand in high spectral or spatial resolution sensors.

1. Introduction

Classification of remotely sensed data is essential in generating thematic maps. Thematic maps have many applications in environmental management, agricultural planning, health studies, climate and biodiversity monitoring, and land change detection (Khatami et al. 2016). A wide range of regional and global datasets for classification are currently available, facilitating studies at unprecedented scales (Grekousis et al. 2015). The classification process, in general, is composed of different tasks, from the selection of data source and sampling design, to classification method selection and classifier performance evaluation (Lu and Weng 2007). Although all of these tasks are important and their successful implementation is dependent on each other, a major task is the selection of a suitable classification method.

One type of classification method may be more suitable for a specific target objective, problem condition, or imaging details over another method (see Table 1 in Lu and Weng 2007). The classifiers performance assessment is also highly dependent on data quality, data values distribution, and sampling design (Jin et al. 2014; Li et al. 2014a); and it can also be evaluated under various criteria like accuracy, reproducibility and/or robustness (Cihlar et al. 1998). Even for the most widely used assessment criteria for classification accuracy, there are important concerns that limit the ability to properly assess the accuracy of resulting map (see Foody 2002, for a review). This line of research has been followed by more recent papers discussing the problems arising from increasing accuracy degradation over time in temporal land cover analysis and change detection (Giles M. Foody 2010), or stressing the importance of sample size or statistical hypothesis testing when comparing different classifiers or scenarios performance (Giles M. Foody 2009).

* Corresponding author.
E-mail addresses: sshahhey@syr.edu (S.S. Heydari), gmountrakis@esf.edu (G. Mountrakis).

http://dx.doi.org/10.1016/j.rse.2017.09.035
Received 20 February 2017; Received in revised form 18 September 2017; Accepted 28 September 2017
Available online 06 October 2017

Table 1
Classifiers parameters

Classifier	Parameter	Parameter values/range
Naïve Bayesian (NB)	Probability distribution type	Normal, Kernel
	Smoothing function	Normal, Box, Triangle, Epanechnikov
K-Nearest Neighbor	Distance metric	Chebychev, Euclidean, Mahalanobis, Minkowski
(KNN)	Distance weight	Inverse, squared inverse
	Number of neighbors	1 to 40 (step of 2)
Support Vector	Kernel function	Fixed at Gaussian
Machine (SVM)	Box constraint (C)	0.01, 0.1, 0.5, 1, 2, 5, 10, 25, 50, 100, 300
	Kernel scale (gamma)	0.1, 0.5, 1, 2, 5, 10, 25, 50
Tree ensemble	Ensemble method	Bagging
(BagTE)	Number of trees	50, 100, 200, 500
	Maximum number of tree splits	10, 25, 50, 100, 200
	Minimum tree leaf size	1, 3, 5, 10, 25
	Number of simulation iterations	10
Artificial Neural	Training algorithm	Resilient backpropagation (trainrp)
Network	# of nodes in 1st hidden layer	5 to 15 (step of 1)
(1 or 2 hidden layers,	# of nodes in 2nd hidden layer	0 to 8 (step of 1)
followed by a softmax classifier)	Number of simulation iterations	100
	Training parameters (specific to chosen training algorithm):	Changed randomly in each iteration within given range:
	- Learning rate	- 0.01–1
	- Delta0	- 0.01–0.5
	- Delta_inc	- 1–5
	- Delta_dec	- 0.1–1
Deep Neural Network, autoencoder-based	Training algorithm	Standard backpropagation
(1, 2, or 3 hidden layers, followed by a softmax	# of nodes in 1st hidden layer	5 to 30 (step of 2)
classifier)	# of nodes in 2nd hidden layer	0 to 20 (step of 2)
	# of nodes in 3rd hidden layer	0 to 10 (step of 2)
	Number of simulation iterations	100
	Training parameters (specific to chosen training algorithm):	
	- Lambda	
	- Rho	- 1E-8–1E-3
	- Beta	- 0.05–0.7
		- 1–9

Therefore, it is difficult to generate a general statement to advise on classifiers ranking. One should always declare the specific conditions that the classifier performance assessment is based on. There are good review papers that introduce the classifiers in general and discuss their application conditions, strengths and weaknesses (Lu and Weng 2007; C. Li et al., 2014; X. Li et al., 2014), but they are mostly qualitative without specific quantitative results for example, best attainable classifiers accuracy. Other papers discuss classifiers for certain problem types. For example, see (Weng 2012) for a discussion on classifiers for mapping of impervious surfaces, (Mallinis and Koutsias 2012) for a comparison of ten classifiers for burned area mapping, (J. He et al. 2015), for comparing four main classifiers in generation of arctic geological maps, or (Pelletier et al. 2016) for assessing the robustness of random forest (RF) classifier for a specific area. Still, other researchers seek to review the application of a specific classifier in more detail. For example, see (Mountrakis et al. 2011) for a review of SVM classifiers; (Pal and Mather 2003), for an assessment of decision tree methods for land use classification; or (Belgiu and Drăguţ 2016), for an overview of random forest classifier. Additional processing is another focus of research which includes making ensemble of classifiers (X. Li et al. 2014), controlling of misclassification by post-processing (Marcos Martinez and Baerenklau 2015), or using ancillary data to aid in classification by field visits (Meddens et al. 2016) or other sources and sensors (Zhu et al. 2016). Based on numerous case studies, one can perform a meta-analysis of previously researched cases and assess the comparative results of case studies at a higher level. This meta-analysis has been done for a single type of classifier such as KNN (Chirici et al. 2016), or more general including pairwise comparisons among many classifiers (Khatami et al. 2016).

While fragmented comparisons between traditional classifiers can be found in existing literature, they are limited in terms of: i) number of case studies incorporated, ii) the search space of the classifier parameters (often resorting to default values), and iii) absence of a promising new classification family based on deep neural networks (DNN). To the best of our knowledge, there are just a few studies that investigate per-pixel classifier accuracy performance over multiple case studies or over a large area. For example, (Ballantine et al. 2005) performed mapping for continental North Africa using MODIS data but comparisons were restricted to a few classifiers. In (Gong et al. 2013) a global sampling and classification was implemented using four different classifiers, but they used a fixed set of parameters for each classifier. Similarly, (Lawrence and Moran 2015) tested classification accuracy for multiple classifiers for 30 data sets but they used a fixed set of classifier parameters that did not allow classifiers to reach their best potential. (Pelletier et al. 2016) performed a grid search on classifier parameters over two large areas in France, focusing on SVM and Random Forest classifiers. Finally, W. Li et al. (2016) employed numerous popular classifiers plus the new autoencoder-based DNN implementations over one composite set sampled through the entire Africa, but they only reported a fixed parameter set (except for DNN).

Our research goals fill this gap by overcoming the three aforementioned limitations. Along these lines, we: i) compared classifiers' best achievable accuracy, ii) identified the accuracy costs associated with the reduction of the parameter grid and training dataset size and iii) investigated how landscape heterogeneity influences classifier performance. We tested six different classifiers in our research: Naïve Bayes (NB), Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Bootstrap-aggregation ensemble of decision trees (BagTE), artificial neural network (ANN) up to 2 hidden layers, and autoencoder-based deep neural network (DNN) up to 3 hidden layers. We used a dataset of 26 Landsat images for classifiers comparison, and ran each classifier with a grid of parameter settings to evaluate its performance.

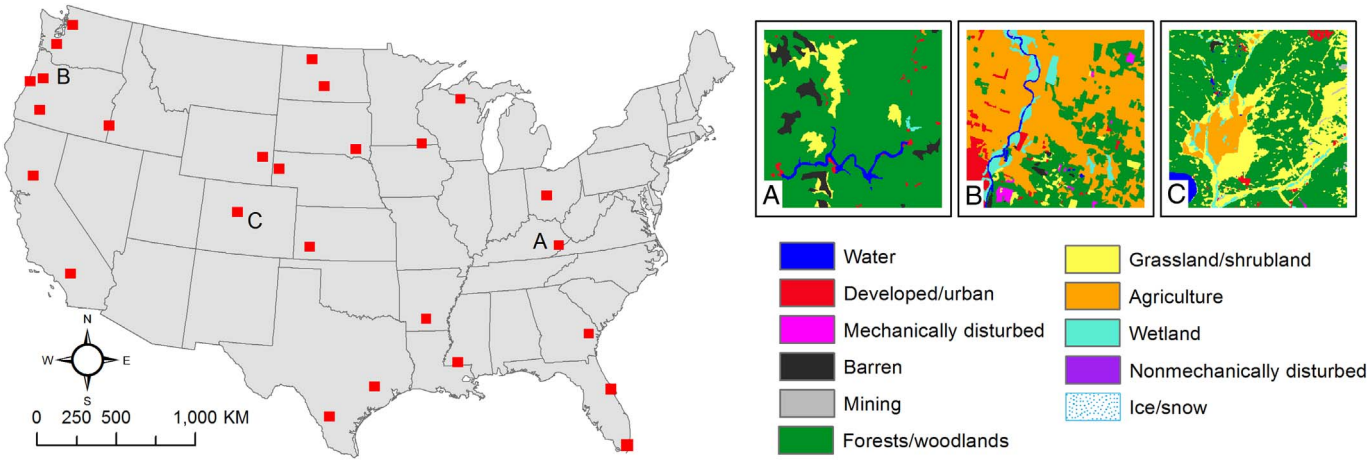

Fig. 1. Spatial distribution and three samples of the 26 images used in this study (from Khatami et al. 2017).

2. Study area

This study incorporated the same input data reported in (Khatami et al. 2017). This data was based on a set of 26 Landsat images (blocks), each covering a 10kmx10km area at 30 m spatial resolution and represented by a matrix of 333 × 333 6-band pixel values. It was accompanied with an entire block of reference data on land cover classes for the complete 26 blocks. This set was part of a larger work maintained by US Geological Survey under the Land Cover Trends program. Reference data was created with the help of aerial photography, and over 33,000 geographically referenced field photos with associated keywords capturing existing land cover (the field photo map is available at http://landcovertrends.usgs.gov/fieldphotomap/map.html). Land cover types were represented by a single value for each pixel and coded in 11 different classes according to modified Anderson scheme, Anderson et al. 1976 (see also http://landcovertrends.usgs.gov/maixn/classification.html). The selected blocks covered a range of climate and topographic conditions throughout the continental US, and all had the same spatial resolution of 30 m (Fig. 1). The incorporated Landsat images reflected the same acquisition years of the high resolution data. Land cover class composition for every block is provided in Table S1.

3. Methodology

3.1. Sampling design

A fixed-rate stratified class-proportional random sampling was conducted on each image to train the classifiers and validate them. Having full reference data for each of the 26 blocks, we randomly sampled each land cover type at 2% and 0.2% to assess the effect of sample size. Note that land cover types with less than two samples in the sampled set (less than 122 instances in the entire image) were dropped. To increase the statistical confidence on the performance results, the process was replicated 10 times to create 10 independent sampled data sets (referred to as *calibration sets* hereafter) for each image. Each calibration set was further divided into training and validation parts. The training part comprised 82% of the calibration set and is used to train a classifier, and the validation part was used to check the classifier's generalization capability and pick the best model for accuracy assessment. Assessed accuracy was reported for the entire block according to the procedure described in Section 3.3. To avoid sampling bias the calibration datasets were kept constant for all classifiers. Optimizing training sample selection and recent advances in active learning area are not covered in this paper.

3.2. Classifier parameterization and training

Six popular classifiers were selected, and their implementation in Mathwork's Matlab was used to run the experiments. Details on these classifiers can be found in multiple sources, for example (Domingos and Pazzani 1997) for Bayesian classifiers; (G.M. Foody and Mathur 2004) for SVM; (Calvo-Zaragoza et al. 2015) for KNN; (Breiman 1996) and (Breiman 2001) for tree ensembles and Random Forests; (Mas and Flores 2008) for ANN; and (Chen et al. 2014) for DNN. We recommend consulting the Matlab documentation (Mathworks Inc., 2016) and Matlab help pages for classifier parameters description, especially for neural network classifiers.

Each classifier has a set of tuning parameters; we selected the most important ones (based on past studies) as indicated in Table 1. We defined ranges of applicable values for each parameter and tested the classifiers' performance for each possible combination of individual parameters to identify the best performer (i.e. a grid search approach). The range of values for each parameter was chosen to cover the practically important cases. In some cases, a subset of all available parameter settings was used through a quick initial assessment in order to constrain the large number of possible parameter combinations.

Additional considerations pertaining to the choice of a specific classifier include the following:

- NB: The smoothing function was used only when the probability distribution type was set to ‘Kernel’.
- KNN: The ‘Minkowski’ metric setting requires an additional parameter named ‘exponent’, it was

set at a fixed value of 3 in all simulations of this distance type.

- BagTE: Due to the randomization involved in the bagging algorithm, we repeated each single run of classifier for a number of iterations and picked best result to record. Our experiments showed that the tree ensemble performance result varied marginally between iterations, we limited the number of iterations to 10. Note that the BagTE slightly differs from the Random Forest implementation. The Random Forest preselected the features used to make each tree branch randomly among all the feature sets, but in the BagTE all the features were available at each branching.
- ANN: As with the BagTE parameter initialization values may affect the result. In the ANN this effect is more pronounced than in the BagTE (standard deviation more than 20% accuracy in some cases) therefore we set the number of iterations to 100. For some parameters, an exhaustive grid search took place (# of nodes) while for other parameters, random values within the predefined range were

Table 2
Pseudocode of accuracy calculation for each classifier and each image block.

```
Define classifier to use
    For replication =1:10 (10 calibration datasets per block)
        For each parameter combination (dependent on classifier characteristics, see Table 1)
            For iterations =1:n (n=10 for BagTE, n=100 for ANN and DNN, n=1 for others)
                Train classifier using training data
                Estimate accuracy metric using validation data
            End
        End
        Identify optimal parameter set defined as the set with the highest validation accuracy metric for
            given replication
        Calculate the entire block accuracy metric for the selected optimal parameters
    End
Calculate average best entire block accuracy metric (and standard deviation) over the ten calibration datasets
```

assigned for each run to keep the combination choices at a reasonable level.

- DNN: Training followed the same considerations as the ANN. To have control over training parameters, the DNN implementation was based on custom code with the help of the

  "Unsupervised Feature Learning and Deep Learning" web site at http://deeplearning.stanford.edu/wiki/index.php/UFLDL_Tutorial.

After building the training/validation data sets and setting up the classifier parameter grid, each classifier was run on each image separately, iterating through all parameters grid points and all ten input data sets (repetitions for same parameters).

3.3. Accuracy assessment

The accuracy assessment replicates the typical algorithmic training procedure with the advantage that entire block accuracy metric generalizations can be extracted for further study. Table 2 presents the general steps followed to obtain accuracy estimations for each classifier and block.

The above process was repeated for the two sampling rates, 0.2% and 2% separately. Accuracy assessment was first conducted on the entire image block (Sections 4.1 and 4.2). Even though the entire block contains the training pixels, the influence of the latter is negligible due to their small presence (up to 2%). To assess the influence of landscape heterogeneity, we tested a subset of edge pixels (i.e. pixels that lay on the border line of different land cover classes). This process is further discussed in Section 4.4. In some cases, the simulations were also repeated by changing the class distribution in data sets as described in Section 4.3.

There are many metrics available to assess classifiers performance in the literature. We chose Overall Accuracy (OA) as it is one of the most widely used metrics with easy interpretation and high practical value. The drawback is that OA hides the class specific performance and as previously discussed (H. He and Garcia 2009), the OA value can be deceiving when the input dataset is highly imbalanced. In such a case, the OA value mostly reflects the dominant class performance while the rare classes may be classified very poorly. Using other metrics such as Precision/Recall or Receiver Operating Characteristics (ROC) is more favoured when performance on rare classes is more important. Picking OA as the assessment metric, the best class distribution is the naturally occurring one (H. He and Garcia 2009) and therefore our stratified sampling for training matches the selected metric. Another metric, the Kappa statistic, has also been used in prior literature to reflect the possibility of chance agreement. However, its usage is less favoured nowadays and it is even highly criticized to be "useless, misleading and/or flawed for the practical applications in remote sensing" (Pontius and Millones 2011). For these reasons we opted to solely report OA results.

In our presented results, we compare classifiers according to their performance (measured by OA) on the same dataset, therefore dependency of OA to class distribution does not bias results. In Section 4.4 we look at the change in OA over all blocks by change in frequency of edge pixels, which may affect results. We therefore discuss the class distribution issue in Section 4.3 before presenting the edge pixel analysis results.

4. Results

4.1. Classifier accuracy comparison for a typical 2% reference dataset

Table 3 shows the obtained average best entire block overall accuracy following the procedure of Table 2 for the 2% calibration dataset (Table S2 contains the corresponding results for the 0.2% calibration dataset). The coefficient of variation (ratio of standard deviation to mean value) is shown in parenthesis. The SVM was the best classifier in 18 out of 26 cases followed by BagTE (3 cases), ANN (3), and DNN (2). However, apart from NB that had significantly lower performance, all other classifiers performed similarly with minor practical variations. This result indicated that when sufficient training data and parameter searches are fed to these popular algorithms, performance does not differ substantially.

We also looked at the best setting of classifier parameters for each image but found that there is no specific parameter value that can be advised for a classifier as the best parameters over all study cases. In fact, raising one configuration as the winner over the others is not justified and everything was dependent on a specific image (and sampling design). The NB method was not considered further in this manuscript due to its considerably lower performance.

(for ANN and DNN classifiers, the listed OA is the highest achieved by any number of hidden layers and nodes per layer within the parameter limits).

4.2. Effect of sample size on classification accuracy

Although large training data sets are desirable, accurately located and labelled training samples in a remote sensing application are generally difficult to obtain. Our sampling rate of 2% for each land cover type (which translates to the total of 2218 pixels for each of our 110,889-pixel blocks) reflected a practical upper bound. For completeness, we also tested the classifiers' performance with a larger 5% sampling rate and the results were very close to the 2% sampling ratio,

Table 3
Best average overall accuracies and their coefficient of variation for 2% sample size.

Classifier→	NB	SVM	KNN	BagTE	ANN	DNN
ImageNo ↓						
1	95.10% (0.80%)	98.11% (0.32%)	98.08% (0.21%)	98.19% (0.13%)	98.05% (0.16%)	98.10% (0.10%)
2	64.45% (6.76%)	83.84% (0.40%)	83.14% (0.52%)	82.97% (0.42%)	82.42% (0.55%)	83.12% (0.56%)
3	73.01% (1.01%)	92.01% (0.96%)	91.25% (0.26%)	91.32% (0.49%)	91.72% (0.40%)	91.81% (0.37%)
4	74.71% (1.46%)	80.38% (0.50%)	79.28% (0.48%)	80.24% (0.31%)	79.93% (0.63%)	80.35% (0.62%)
5	96.31% (0.41%)	98.37% (0.15%)	98.15% (0.13%)	98.09% (0.06%)	98.31% (0.20%)	98.24% (0.23%)
6	68.71% (4.47%)	77.95% (0.54%)	77.71% (0.72%)	78.18% (0.18%)	78.10% (0.34%)	78.02% (0.51%)
7	87.99% (0.16%)	90.85% (0.25%)	90.57% (0.19%)	90.48% (0.19%)	90.73% (0.26%)	90.80% (0.32%)
8	79.88% (0.85%)	84.66% (0.43%)	84.32% (0.51%)	84.47% (0.31%)	84.58% (0.32%)	84.38% (0.47%)
9	57.26% (1.21%)	65.29% (0.72%)	64.17% (1.50%)	64.76% (0.59%)	64.15% (1.39%)	64.57% (1.05%)
10	58.37% (0.96%)	77.16% (0.32%)	76.26% (0.75%)	75.99% (0.36%)	75.56% (0.73%)	76.43% (0.51%)
11	87.83% (0.62%)	92.26% (0.31%)	92.21% (0.34%)	92.39% (0.13%)	92.18% (0.21%)	92.19% (0.18%)
12	65.22% (0.82%)	76.01% (1.03%)	75.94% (0.37%)	75.95% (0.35%)	75.72% (0.65%)	76.11% (0.33%)
13	80.39% (0.50%)	83.85% (0.33%)	83.75% (0.34%)	83.72% (0.19%)	83.88% (0.34%)	83.99% (0.18%)
14	84.91% (0.46%)	87.66% (0.30%)	87.19% (0.56%)	87.45% (0.20%)	87.70% (0.36%)	87.67% (0.23%)
15	64.13% (4.40%)	86.72% (0.42%)	86.24% (0.26%)	86.20% (0.54%)	86.27% (0.57%)	86.54% (0.51%)
16	79.09% (0.95%)	86.74% (0.28%)	85.79% (0.28%)	85.97% (0.37%)	86.80% (0.29%)	86.56% (0.44%)
17	77.68% (2.96%)	85.83% (0.45%)	84.78% (0.69%)	85.06% (0.55%)	85.21% (0.47%)	85.35% (0.37%)
18	68.95% (7.39%)	85.09% (0.44%)	85.24% (0.30%)	85.12% (0.17%)	85.34% (0.27%)	85.17% (0.26%)
19	72.62% (2.49%)	80.53% (0.41%)	80.26% (0.50%)	80.29% (0.26%)	80.39% (0.17%)	80.33% (0.23%)
20	65.33% (1.73%)	78.70% (0.99%)	77.29% (1.54%)	76.97% (0.63%)	78.05% (0.59%)	77.90% (1.07%)
21	80.20% (0.86%)	87.20% (0.33%)	86.88% (0.32%)	86.55% (0.31%)	87.17% (0.32%)	86.68% (0.55%)
22	58.18% (2.24%)	71.99% (1.00%)	71.23% (0.40%)	70.99% (0.54%)	70.99% (0.64%)	71.79% (0.59%)
23	74.43% (0.40%)	87.74% (0.31%)	86.80% (0.30%)	86.74% (0.37%)	87.36% (0.46%)	87.37% (0.71%)
24	81.86% (1.03%)	89.19% (0.16%)	88.77% (0.32%)	88.63% (0.25%)	88.83% (0.37%)	88.89% (0.30%)
25	76.18% (1.50%)	80.38% (0.54%)	80.26% (0.25%)	80.20% (0.32%)	79.91% (0.44%)	79.84% (0.56%)
26	85.33% (1.04%)	93.71% (0.24%)	93.29% (0.19%)	93.21% (0.28%)	93.60% (0.16%)	93.39% (0.45%)

Table 4
Best attainable accuracy (over all classifiers) for sampling rates of 2% and 0.2%.

Image# →		***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***	***10***	***11***	***12***	***13***
Sampling rate	2	98.2	83.8	92.0	80.4	98.4	78.2	90.9	84.7	65.3	77.2	92.4	76.1	84.0
Sampling rate	0.2	97.6	79.5	88.5	77.5	97.6	76.1	88.9	81.8	59.6	71.6	91.5	73.6	82.1
Image# →		***14***	***15***	***16***	***17***	***18***	***19***	***20***	***21***	***22***	***23***	***24***	***25***	***26***
Sampling rate	2	87.7	86.7	86.8	85.8	85.3	80.5	78.7	87.2	72.0	87.7	89.2	80.4	93.7
Sampling rate	0.2	86.0	84.1	84.3	82.7	83.4	79.1	73.3	84.4	66.9	84.5	86.2	78.5	91.2

and showed a saturation in classifiers' accuracy. We also examined a lower bound by generating a second, considerably smaller calibration dataset (with the same proportional stratified sampling design) at 0.2% of the image size (222 pixels). Table 4 shows the best result among all classifiers for 2% and 0.2% sampling scenarios. Detailed accuracy metrics similar to Table 3 but for the 0.2% sampling rate are offered in the supplementary material. As expected, there was a decline in best attainable accuracy with the 0.2% sampling ranging from 0.6% to 5.7% accuracy.

To investigate further individual classifier performance, each classifier's accuracy was contrasted with SVM accuracy. Fig. 2 depicts this comparison for the 2% and 0.2% calibration datasets (excluding NB due to low performance). For the 2% case the SVM typically outperformed other classifiers. In the 0.2% case the BagTE was the best classifier in 12 out of 26 cases, and SVM was better in only 5 cases. However, independently of the calibration dataset size, the magnitude of the accuracy difference was still small and practically insignificant.

4.3. Effect of dataset class imbalance on classification accuracy

In the previous two sections we compared different classifiers given similar training and testing data separately for each image in order for class distribution to not bias the results. In this section, we specifically assess the effect of class distribution in obtained accuracy. Two training sample scenarios were examined, one with a stratified proportional training set (imbalanced training dataset), and another by training classifiers with randomly selected almost equal class member datasets. In addition to the two training scenarios, performance for each image was examined using two different testing datasets, one covering the entire image (imbalanced testing dataset) and another constraining equal members per class (balanced testing dataset).

Fig. 3 shows the result of calculating OA for the imbalanced training dataset and Fig. 4 for balanced training dataset. In each case, OA values express the maximum attainable value among different classifiers for each image. Due to the low number of pixels in rare classes, the overall dataset size (and therefore the training part) in the balanced scenario is smaller than the original imbalanced case, therefore accuracy comparisons are only applicable within the same training dataset (i.e. Fig. 3 and Fig. 4 should not be combined).

The OA dropped considerably by changing class distribution with the imbalanced training dataset (Fig. 3), however the difference was limited under the balanced distribution training (Fig. 4). This suggests that stratification for sample selection can significantly impact obtained accuracy. Thus, sampling design should reflect study preferences (highest overall scene accuracy vs. balanced accuracy between classes).

4.4. Effect of landscape heterogeneity on classification accuracy

Numerous metrics have been developed over time to characterize and assess the scene and landscape heterogeneity, and software packages are also available for help (for example see Turner 2005; or Lausch et al. 2015). These metrics can be defined in many ways and vary in scope, considering the number of different landscape classes and/or their spatial distribution. An comprehensive list of landscape

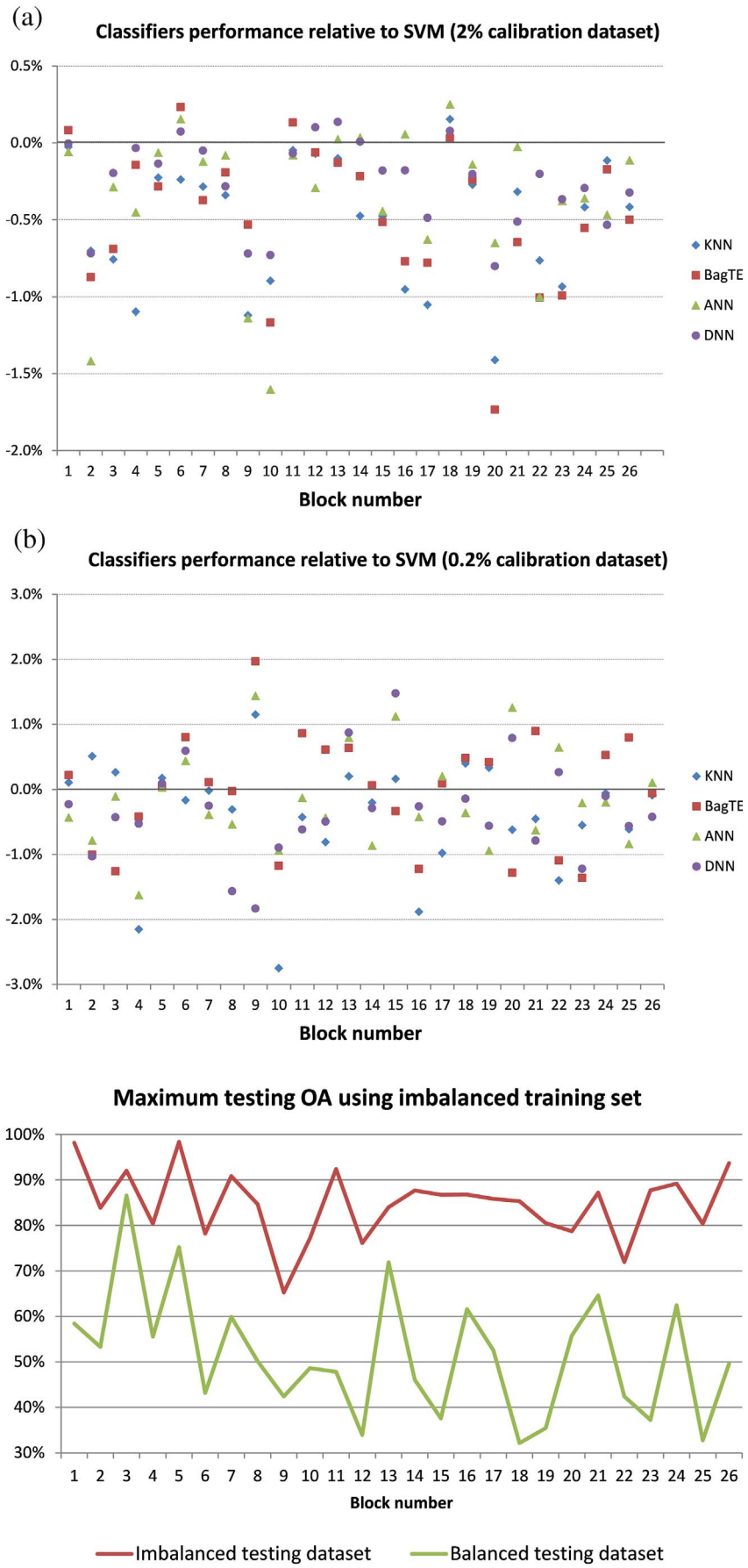

Fig. 2. Classifiers overall accuracy relative to SVM for (a) 2% and (b) 0.2% calibration dataset.

Fig. 3. Effect of changing class distribution in test sets on best attainable accuracy for imbalanced training set distribution.

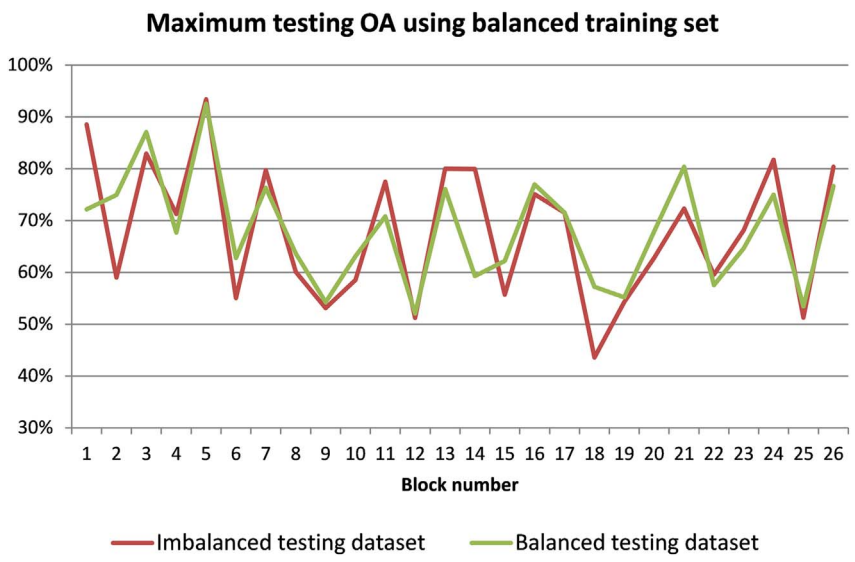

Fig. 4. Effect of changing class distribution in test sets on best attainable accuracy for balanced training set distribution.

metrics can be found in FragStat software's documentation (McGarigal 2015). Although there is no general rule to pick among them, edge statistics are a good candidate to represent scene heterogeneity as it is affected by both class variety and class spatial arrangement. Here we defined the edge pixels as pixels lying on land cover change boundaries; they were extracted from the ground truth data. This selection was also intuitive from a remote sensing point of view, because Landsat images are of medium resolution and in the edge pixels, there is a high chance of land cover mixing. Our 26 blocks exhibited a wide range of edge pixel presence ranging from 2% to more than 40% of the overall block pixels. Two separate analyses are presented in the next two sections. First, we isolate each block and examine algorithmic performance on the edge pixels in order to identify best performing classifiers. Second, we combine classifier performance across all blocks to investigate accuracy degradation as scene heterogeneity increases through higher edge pixel presence.

4.4.1. Algorithmic accuracy assessment on edge pixels separately for each block

Having previously trained classifiers on stratified proportional samples, we can calculate the test accuracy by limiting the test pixels only to edge pixels for each image. The idea is to investigate how different classifiers perform particularly on these difficult-to-classify pixels. Resulting accuracies are depicted in Fig. 5 relative to SVM performance. In Fig. 5(a) the classifiers have been trained on 2% stratified proportional sample and tested on edge pixels (for each image). Fig. 5(b) shows the same result but the training has been done by 0.2% sampling rate. Fig. 5 shows superiority of the SVM and KNN classifiers, especially for the larger 2% calibration sample. BagTE, ANN and DNN performance is not consistent.

4.4.2. Effect of edge pixel presence on accuracy across blocks

It is interesting to seek a potential relationship between scene heterogeneity and classification accuracy. Fig. 6 shows the OA results versus the ratio of edge pixels in each block. For clarity purposes and guided by results in Fig. 5 we limited assessment to the two best performing classifiers, SVM and KNN. The results are based on the balanced training dataset to limit potential class influence on the obtained results. For testing purposes the entire block dataset was used as accuracy differences between an unbalanced and a balanced testing dataset were minor (see Fig. 4).

A clear decreasing trend is identified with approximately 8–9% accuracy reduction for every 10% increase in edge pixel presence for SVM. While the model explains about one third of the variability, it is an important finding considering the multitude of additional factors that may affect classification accuracy in our 26 different sites (e.g. variable spectral signatures and separability of classes).

4.5. Trade-off between execution time and accuracy

Average run-time requirements for the experiment reported in Table 3 (2% reference dataset) are presented in Table 5. To compensate for the usage of different machine configurations and parallel processing capabilities (i.e. number of CPU cores) the NB classifier ran on all machines and the run times were used as a benchmark to provide a common base for comparison. Our intention is not to provide exact execution times but simply an approximate estimation to guide user decisions.

The total run time is directly dependent on the number and range of configuration parameters. We also calculated the average runtime per each parameter setting in the last column. Results showed NB and SVM to be the fastest classifiers per image, but DNN classifiers tend to be the quickest classifiers per each parameter setting (on average).

Lacking a specific protocol on how to set the classifier parameters for best performance, our approach was to do a complete set of simulations for each image/classifier over all reasonable parameter combinations. However, building on our experiments we can investigate best attainable accuracy (on average) taking only a subset of the initial parameter values. A smaller set may miss some of the best parameter combinations, resulting in a decrease in best attainable accuracy. Table 6 and Table 7 show worst case estimates of this gap for different cases of the parameter set contraction for 2% and 0.2% sampling rates respectively. In generating these tables, we assumed that by extracting the accuracy at a given percentile the worst case scenario is obtained. For example in the 100%–75% column, the 75th percentile of the obtained accuracies for each case (image/classifier) was identified. It may be unlikely that by randomly constraining the parameter combinations to 75% of the total possibilities the resulting accuracy will also be bounded by accuracy's 75th percentile, but this is the worst case. Then we averaged the gap between top and 75th percentile accuracy over all blocks for each classifier and reported the results in the 100%–75% column (same procedure for other columns by changing 75th percentile to other percentile values). In the special case of 1-Layer DNN, which has only 13 different configurations, the 5th percentile is not meaningful and is set to N/A. Results indicate that the most tolerant classifier to limiting parameter search space is the BagTE, while the least tolerant is the SVM.

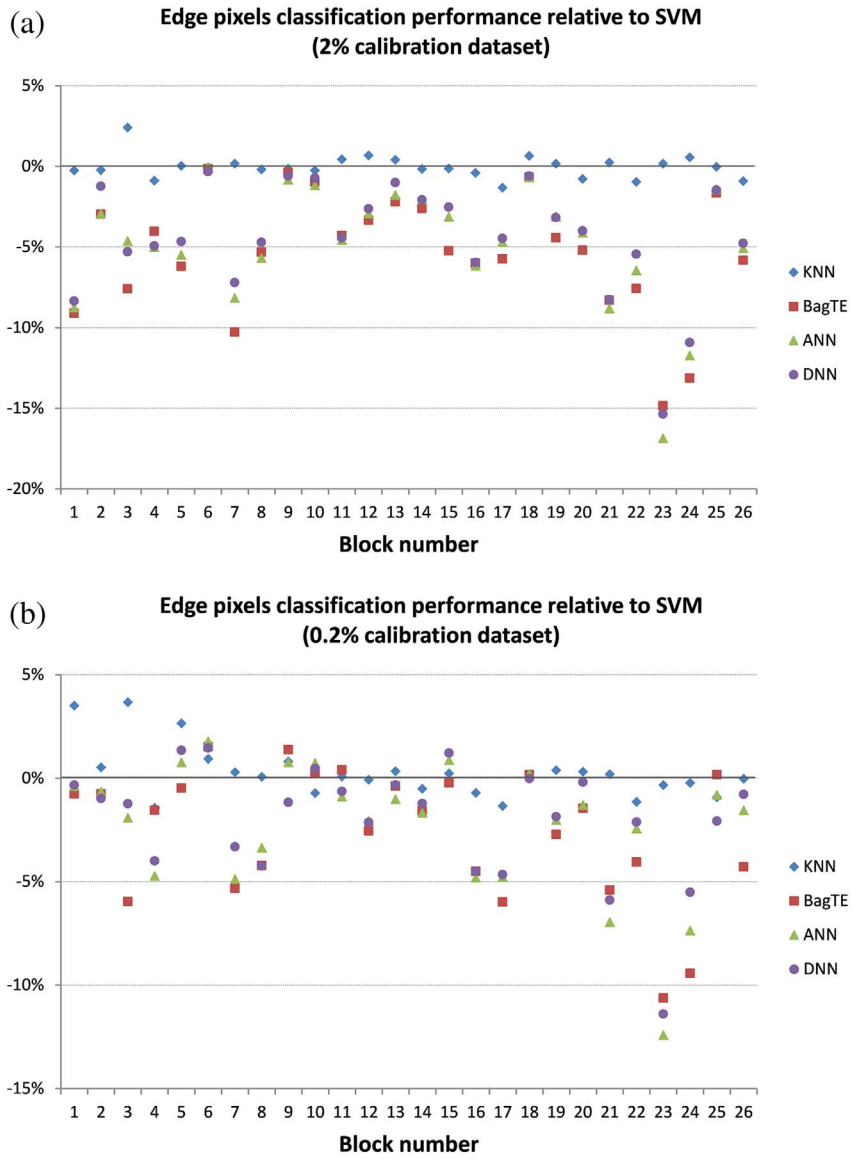

Fig. 5. Edge pixels classification accuracy relative to SVM for (a) 2% and (b) 0.2% calibration dataset.

5. Discussion and concluding remarks

Our goal was to investigate classifier performance under different sampling scenarios and landscape complexities. For the entire block, our accuracy assessment indicated that all classifiers, with the exception of NB, performed similarly. The general performance gap with Naïve Bayes compared to the other classifiers can be explained by the high level of band correlation, which invalidates the class conditional

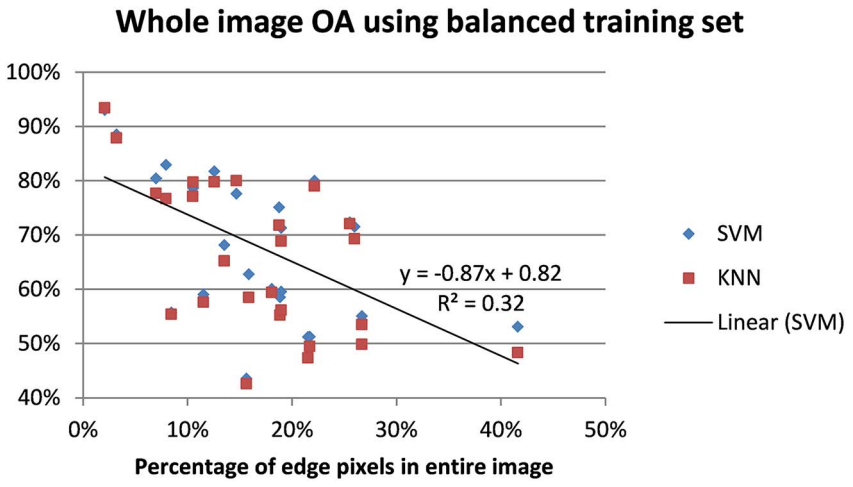

Fig. 6. Effect of edge pixel presence in classifiers overall accuracy, trained on balanced data set and tested on entire image.

Table 5
Average computer run times per image per CPU core for different classifiers.

Classifier	Average run time per Image per CPU core for all combinations (min.)	# of parameter combinations	# of iterations per parameter	Average single run time per parameter setting (sec.)
NB	1.5	5	1	18.1
SVM	4.0	88	1	2.7
KNN	22.1	160	1	8.3
BagTE	53.3	100	10	3.2
ANN	464.3	99	100	2.8
DNN (1-Layer)	15.0	13	100	0.7
DNN (2-Layer)	278.2	130	100	1.3
DNN (3-Layer)	1365.8	650	100	1.3

Table 6
Percentiles performance gap for 2% reference dataset.

Classifier	100%–75%	100%–50%	100%–25%	100%–10%	100%–5%
SVM	0.9%	3.0%	8.7%	13.4%	13.7%
KNN	0.3%	0.5%	1.0%	2.1%	3.8%
BagTE	0.2%	0.6%	1.1%	1.7%	1.8%
ANN	0.5%	0.7%	1.1%	2.1%	2.6%
DNN (1-Layer)	0.5%	0.7%	1.1%	1.7%	N/A
DNN (2-Layer)	0.7%	1.1%	1.5%	2.0%	2.4%
DNN (3-Layer)	1.2%	1.8%	2.5%	3.1%	3.5%

Table 7
Percentiles performance gap for 0.2% reference dataset.

Classifier	100%–75%	100%–50%	100%–25%	100%–10%	100%–5%
SVM	1.9%	5.0%	11.7%	11.9%	12.0%
KNN	0.8%	1.3%	2.3%	3.9%	5.1%
BagTE	0.5%	0.8%	1.4%	2.9%	3.1%
ANN	1.3%	2.0%	2.9%	3.9%	4.7%
DNN (1-Layer)	0.7%	1.4%	2.5%	3.7%	N/A
DNN (2-Layer)	2.6%	3.5%	4.5%	5.4%	6.0%
DNN (3-Layer)	2.9%	3.8%	4.8%	5.7%	6.3%

independence assumption. For other classifiers, similar results can be obtained assuming sufficient search of optimal parameter identification. This suggests that parameter optimization is a key component in the training process and results using a pre-determined parameter set could be misleading (as presented in Lawrence and Moran 2015). Unfortunately, optimal parameter values may vary significantly across sites, and an extensive grid search is therefore required. Moving beyond individual classifiers, training data characteristics can be more influential than classifier selection as shown by limiting the test data to edge pixels. A similar conclusion has been made in other studies, for example in (C. Li et al., 2014).

However, when we concentrated on the edge pixels, it was clear that the SVM and KNN offer considerable accuracy advantages. This could be attributed to the right balance between algorithmic and data complexity. Other methods (ANN, DNN) may offer higher modelling capabilities. Still, relatively small training datasets result in unpredictable generalizations in the feature space. SVM and KNN may also work better than decision trees in the presence of imbalanced data and rare classes because decision trees require enough training samples to find optimum branching decisions and divide-and-conquer strategies may fail on imbalanced data sets. Coupled with their relatively low execution times we would recommend SVM or KNN for classifications using Landsat's spectral inputs and Anderson's 11-level classification scheme. We should also caution though that primarily the SVM and secondarily the KNN demonstrated substantial accuracy degradation during the parameter grid search, therefore an exhaustive optimization process is suggested.

We contrasted our work with previous studies including, (Khatami et al. 2016) and selected similar case studies (i.e. analyzing Landsat multispectral images with no ancillary data) along with other recent works. According to (Ouyang and Ma 2006), (Zhong et al. 2007), (Dixon and Candade 2008), (Qing et al. 2010), and (C.-H. Li et al. 2012), SVM outperformed the Maximum-Likelihood classifier (which is based on the same principle as our NB classifier) by at least 5% in overall accuracy, but the SVM gain was less than 5% compared to multilayer neural networks and less than 3% when compared to KNN classifier. In a different experiment Maximum Likelihood, Neural Network, and SVM achieved overall accuracies with difference less than around 1%, and it was not statistically significant (Mallinis and Koutsias 2012).

(J. He et al. 2015) used Landsat images and reported SVM as the best classifier, followed by Neural.

Networks, Random Forest, and lastly Maximum Likelihood. The average performance difference between the first two or the last two classifiers was not statistically significant, but it was significant (although less than 5%) between the two groups. In another recent study, the Maximum Likelihood classifier was 2% less accurate than the Random Forest, with the latter achieving 86.8% (J. Liu et al. 2016a). (Lawrence and Moran 2015) reported higher performance for Random Forest than SVM, although their use of a fixed set of parameters may not allow either algorithm to reach their potential. One recent study (W. Li et al. 2016) reported a 1.2% increase in overall accuracy of a 3-layer DNN compared to SVM; also RF was 1.8% worse than SVM, with differences being statistically significant. However, it is not clear if an extensive grid search was used in their analysis for SVM and RF. Finally, (Pelletier et al. 2016) did a large grid search on SVM and RF parameters and found the RF to perform significantly better than SVM. They also noticed the RF's low sensitivity to parameter changes.

With respect to the BagTE, a Random Forest variant, our experiments showed the highest potential when the parameter search space is minimized, similar to (Pelletier et al. 2016). This is attributed to the ensemble nature of this classifier that potentially makes it more tolerant to small or noisy samples. Neural network classifiers (ANN and DNN) did not reach their promising credentials in our study. In other fields, ANNs and particularly DNNs have provided significant advances when fed with large amounts of information. Rich data was not the case in our experiment as we restricted input data to pixel-based multispectral information and we found neural networks generally less promising in our case compared to SVM, KNN, and tree ensembles. This may be the result of insufficient or low quality training samples, or data overfitting attributed to the higher complexity of the classification network compared to the data structure. In our simulations, simpler (1-layer) deep networks worked generally better than deeper ones. Furthermore, in additional testing - not reported in this manuscript - increasing the sampling ratio (from 2% to 5%) or using edge pixels for classifier training (case of active learning) did not make the neural network a

perform better Therefore it is more probable that the neural network underperformance comes from low number of features (and their dependence) and data overfitting due to higher complexity. As described in (Zhang et al. 2016), the main benefit of DNN use is for processing hyperspectral data, or mix the spectral data with spatial and contextual information and then combine the spectral and other information in a composite per-pixel analysis. Therefore, while neural networks offer limited benefits in our six-dimensional spectral feature space, they still may offer advances when feature space dimensionality increases and spatial relationships are included. Further studies are required to investigate this topic.

Another finding across classifiers is a reduction of classification accuracy as scene complexity increases. While this simple accuracy assessment has been reported in the past (for example in Mallinis and Koutsias 2012; Collin and Planes 2012; Roelfsema and Phinn 2010; and Andrefouet et al. 2003), comparing performances over multiple scenes based on ratio of edge pixels has not and therefore we can offer more conclusive results.

We see two important areas for further evaluation: selection of performance evaluation metric, and sampling design alternatives. Although overall accuracy is widely used, there are some suggestions that prefer ROC or Precision/Recall curves over overall accuracy for analysis of imbalanced cases (e.g. Jeni et al. 2013). A closer look could also identify land cover classes that exhibit higher confusion and try to at least balance the misclassification errors over different classes (Puertas et al. 2013) or perform a one-class classification and modify the evaluation metric (Wenkai Li & Qinghua Guo, 2014). Another approach is to use accuracy metrics at the individual pixel level (Khatami et al. 2017).

With respect to sampling design alternatives (for training) many approaches have been recently devised especially for learning from imbalanced data, as reviewed (H. He and Garcia 2009) and later presented in (H. He and Ma 2013). Systematic inclusion of difficult-to-classify samples like edge pixels is another option to consider, which has been investigated recently in other research (M. Liu et al. 2016b). This approach can be considered as an example of a group of techniques named active learning, which is well known in machine learning and has been used and discussed in the remote sensing field as well (see Bachmann 2003 and Crawford, Tuia, & Yang, (2013). As discussed and reviewed in (Tuia et al. 2011), active learning “aims at building efficient training sets by iteratively improving the model performance through sampling.” In other words, samples used for training are selected interactively. Most of the research in this area is, for now, concentrated on very high spatial/spectral resolution imagery, and Landsat type data is not examined. There are also cases of unexpected results with active learning (Wuttke et al. 2016), so caution should be exercised.

To summarize, our experiments identified SVM and KNN as the best performing methods for Landsat classifications. Caution should be exercised though as their performance is dependent on a wide search of their parameter space. Furthermore, the selection of the training sample composition (class balance) will have a considerable effect on the obtained accuracy, therefore users should consider accuracy priorities (overall scene vs specific classes) in their sampling design. Finally, edge pixel presence, a heterogeneity metric, was shown to have a considerable effect on the classification accuracy.

Acknowledgments

This work was supported by the USDA McIntire Stennis program, a SUNY ESF Graduate Assistantship and NASA's Land Cover Land Use Change Program (grant # NNX15AD42G). We thank Dr. Reza Khatami for sharing the processed datasets and Dr. Steve Stehman for suggestions for the experimental design. We also appreciate the efforts from Kristi Sayler and Mark Drummond (USGS) for providing the Trends data. We thank the reviewers for providing constructive corrections and insights, especially with respect to the influence of the class distribution.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.rse.2017.09.035.

References

Anderson, J.R., Hardy, E.E., Roach, J.T., Witmer, R.E., 1976. A Land Use and Land Cover Classification System for Use With Remote Sensor Data (Report No. 964) (Retrieved from). http://pubs.er.usgs.gov/publication/pp964.

Andrefouet, S., Kramer, P., Torres-Pulliza, D., Joyce, K.E., Hochberg, E.J., Garza-Perez, R., ... Muller-Karger, F.E., 2003. Multi-site evaluation of IKONOS data for classification of tropical coral reef environments. Remote Sens. Environ. 88 (1–2), 128–143. http://dx.doi.org/10.1016/j.rse.2003.04.005.

Bachmann, C.M., 2003. Improving the performance of classifiers in high-dimensional remote sensing applications: an adaptive resampling strategy for error-prone exemplars (ARESEPE). IEEE Trans. Geosci. Remote Sens. 41 (9), 2101–2112. http://dx.doi.org/10.1109/TGRS.2003.817207.

Ballantine, J.-A.C., Okin, G.S., Prentiss, D.E., Roberts, D.A., 2005. Mapping North African landforms using continental scale unmixing of MODIS imagery. Remote Sens. Environ. 97 (4), 470–483. http://dx.doi.org/10.1016/j.rse.2005.04.023.

Belgiu, M., Drăguţ, L., 2016. Random forest in remote sensing: a review of applications and future directions. ISPRS J. Photogramm. Remote Sens. 114, 24–31. http://dx.doi.org/10.1016/j.isprsjprs.2016.01.011.

Breiman, L., 1996. Bagging predictors. Mach. Learn. 24 (2), 123–140. http://dx.doi.org/10.1023/A:1018054314350.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32. http://dx.doi.org/10.1023/A:1010933404324.

Calvo-Zaragoza, J., Valero-Mas, J.J., Rico-Juan, J.R., 2015. Improving kNN multi-label classification in Prototype Selection scenarios using class proposals. Pattern Recogn. 48 (5), 1608–1622. http://dx.doi.org/10.1016/j.patcog.2014.11.015.

Chen, Y., Lin, Z., Zhao, X., Wang, G., Gu, Y., 2014. Deep learning-based classification of hyperspectral data. IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens. 7 (6), 2094–2107. http://dx.doi.org/10.1109/JSTARS.2014.2329330.

Chirici, G., Mura, M., McInerney, D., Py, N., Tomppo, E.O., Waser, L.T., McRoberts, R.E., 2016. A meta-analysis and review of the literature on the k-Nearest Neighbors technique for forestry applications that use remotely sensed data. Remote Sens. Environ. 176, 282–294. http://dx.doi.org/10.1016/j.rse.2016.02.001.

Cihlar, J., Xiao, Q., Chen, J., Beaubien, J., Fung, K., Latifovic, R., 1998. Classification by progressive generalization: a new automated methodology for remote sensing multichannel data. Int. J. Remote Sens. 19 (14), 2685–2704. http://dx.doi.org/10.1080/014311698214451.

Collin, A., Planes, S., 2012. Enhancing coral health detection using spectral diversity indices from worldview-2 imagery and machine learners. Remote Sens. 4 (10), 3244–3264. http://dx.doi.org/10.3390/rs4103244.

Crawford, M.M., Tuia, D., Yang, H.L., 2013. Active learning: any value for classification of remotely sensed data? Proc. IEEE 101 (3), 593–608. http://dx.doi.org/10.1109/JPROC.2012.2231951.

Dixon, B., Candade, N., 2008. Multispectral landuse classification using neural networks and support vector machines: one or the other, or both? Int. J. Remote Sens. 29 (4), 1185–1206. http://dx.doi.org/10.1080/01431160701294661.

Domingos, P., Pazzani, M., 1997. On the optimality of the simple bayesian classifier under zero-one loss. Mach. Learn. 29 (2–3), 103–130. http://dx.doi.org/10.1023/A:1007413511361.

Foody, Giles M., 2002. Status of land cover classification accuracy assessment. Remote Sens. Environ. 80 (1), 185–201. http://dx.doi.org/10.1016/S0034-4257(01)00295-4.

Foody, Giles M., 2009. Classification accuracy comparison: Hypothesis tests and the use of confidence intervals in evaluations of difference, equivalence and non-inferiority. Remote Sens. Environ. 113 (8), 1658–1663. http://dx.doi.org/10.1016/j.rse.2009.03.014.

Foody, Giles M., 2010. Assessing the accuracy of land cover change with imperfect ground reference data. Remote Sens. Environ. 114 (10), 2271–2285. http://dx.doi.org/10.1016/j.rse.2010.05.003.

Foody, G.M., Mathur, A., 2004. A relative evaluation of multiclass image classification by support vector machines. IEEE Trans. Geosci. Remote Sens. 42 (6), 1335–1343. http://dx.doi.org/10.1109/TGRS.2004.827257.

Gong, P., Wang, J., Yu, L., Zhao, Y., Zhao, Y., Liang, L., Chen, J., 2013. Finer resolution observation and monitoring of global land cover: first mapping results with Landsat TM and ETM + data. Int. J. Remote Sens. 34 (7), 2607–2654. http://dx.doi.org/10.1080/01431161.2012.748992.

Grekousis, G., Mountrakis, G., Kavouras, M., 2015. An overview of 21 global and 43 regional land-cover mapping products. Int. J. Remote Sens. 36 (21), 5309–5335. http://dx.doi.org/10.1080/01431161.2015.1093195.

He, H., Garcia, E.A., 2009. Learning from imbalanced data. IEEE Trans. Knowl. Data Eng. 21 (9), 1263–1284. http://dx.doi.org/10.1109/TKDE.2008.239.

He, H., Ma, Y., 2013. Imbalanced Learning: Foundations, Algorithms, and Applications. Wiley-IEEE Press.

He, J., Harris, J.R., Sawada, M., Behnia, P., 2015. A comparison of classification algorithms using Landsat-7 and Landsat-8 data for mapping lithology in Canada's Arctic. Int. J. Remote Sens. 36 (8), 2252–2276. http://dx.doi.org/10.1080/01431161.2015.

1035410.
Mathworks Inc, 2016. Matlab Statistics and Machine Learning Toolbox User's Guide R2016a; Matlab Neural Network Toolbox User's Guide R2016a.
Jeni, L.A., Cohn, J.F., De La Torre, F., 2013. Facing Imbalanced Data–Recommendations for the Use of Performance Metrics. pp. 245–251. (IEEE). https://doi.org/10.1109/ACII.2013.47.
Jin, H., Stehman, S.V., Mountrakis, G., 2014. Assessing the impact of training sample selection on accuracy of an urban classification: a case study in Denver, Colorado. Int. J. Remote Sens. 35 (6), 2067–2081. http://dx.doi.org/10.1080/01431161.2014.885152.
Khatami, R., Mountrakis, G., Stehman, S.V., 2016. A meta-analysis of remote sensing research on supervised pixel-based land-cover image classification processes: general guidelines for practitioners and future research. Remote Sens. Environ. 177, 89–100. http://dx.doi.org/10.1016/j.rse.2016.02.028.
Khatami, R., Mountrakis, G., Stehman, S.V., 2017. Mapping per-pixel predicted accuracy of classified remote sensing images. Remote Sens. Environ. 191, 156–167. http://dx.doi.org/10.1016/j.rse.2017.01.025.
Lausch, A., Blaschke, T., Haase, D., Herzog, F., Syrbe, R.-U., Tischendorf, L., Walz, U., 2015. Understanding and quantifying landscape structure – a review on relevant process characteristics, data models and landscape metrics. Ecol. Model. 295, 31–41. http://dx.doi.org/10.1016/j.ecolmodel.2014.08.018.
Lawrence, R.L., Moran, C.J., 2015. The AmericaView classification methods accuracy comparison project: a rigorous approach for model selection. Remote Sens. Environ. 170, 115–120. http://dx.doi.org/10.1016/j.rse.2015.09.008.
Li, Wenkai, Guo, Qinghua, 2014. A new accuracy assessment method for one-class remote sensing classification. IEEE Trans. Geosci. Remote Sens. 52 (8), 4621–4632. http://dx.doi.org/10.1109/TGRS.2013.2283082.
Li, C.-H., Kuo, B.-C., Lin, C.-T., Huang, C.-S., 2012. A spatial-contextual support vector machine for remotely sensed image classification. IEEE Trans. Geosci. Remote Sens. 50 (3), 784–799. http://dx.doi.org/10.1109/TGRS.2011.2162246.
Li, C., Wang, J., Wang, L., Hu, L., Gong, P., 2014a. Comparison of classification algorithms and training sample sizes in urban land classification with landsat thematic mapper imagery. Remote Sens. 6 (2), 964–983. http://dx.doi.org/10.3390/rs6020964.
Li, X., Liu, X., Yu, L., 2014b. Aggregative model-based classifier ensemble for improving land-use/cover classification of Landsat TM Images. Int. J. Remote Sens. 35 (4), 1481–1495. http://dx.doi.org/10.1080/01431161.2013.878061.
Li, W., Fu, H., Yu, L., Gong, P., Feng, D., Li, C., Clinton, N., 2016. Stacked Autoencoder-based deep learning for remote-sensing image classification: a case study of African land-cover mapping. Int. J. Remote Sens. 37 (23), 5632–5646. http://dx.doi.org/10.1080/01431161.2016.1246775.
Liu, J., Feng, Q., Gong, J., Zhou, J., Li, Y., 2016a. Land-cover classification of the Yellow River Delta wetland based on multiple end-member spectral mixture analysis and a Random Forest classifier. Int. J. Remote Sens. 37 (8), 1845–1867. http://dx.doi.org/10.1080/01431161.2016.1165888.
Liu, M., Cao, X., Li, Y., Chen, J., Chen, X., 2016b. Method for land cover classification accuracy assessment considering edges. Sci. China Earth Sci. 59 (12), 2318–2327. http://dx.doi.org/10.1007/s11430-016-5333-5.
Lu, D., Weng, Q., 2007. A survey of image classification methods and techniques for improving classification performance. Int. J. Remote Sens. 28 (5), 823–870. http://dx.doi.org/10.1080/01431160600746456.
Mallinis, G., Koutsias, N., 2012. Comparing ten classification methods for burned area mapping in a Mediterranean environment using Landsat TM satellite data. Int. J. Remote Sens. 33 (14), 4408–4433. http://dx.doi.org/10.1080/01431161.2011.648284.
Marcos Martinez, R., Baerenklau, K.A., 2015. Controlling for misclassified land use data: a post-classification latent multinomial logit approach. Remote Sens. Environ. 170, 203–215. http://dx.doi.org/10.1016/j.rse.2015.09.025.
Mas, J.F., Flores, J.J., 2008. The application of artificial neural networks to the analysis of remotely sensed data. Int. J. Remote Sens. 29 (3), 617–663. http://dx.doi.org/10.1080/01431160701352154.
McGarigal, K., 2015. Fragstats Help. University of Massachusetts, Amherst.
Meddens, A.J.H., Kolden, C.A., Lutz, J.A., 2016. Detecting unburned areas within wildfire perimeters using Landsat and ancillary data across the northwestern United States. Remote Sens. Environ. 186, 275–285. http://dx.doi.org/10.1016/j.rse.2016.08.023.
Mountrakis, G., Im, J., Ogole, C., 2011. Support vector machines in remote sensing: a review. ISPRS J. Photogramm. Remote Sens. 66 (3), 247–259. http://dx.doi.org/10.1016/j.isprsjprs.2010.11.001.
Ouyang, Y., Ma, J., 2006. Classification of multi-spectral remote sensing data using a local transfer function classifier. Int. J. Remote Sens. 27 (24), 5401–5408. http://dx.doi.org/10.1080/01431160600823222.
Pal, M., Mather, P.M., 2003. An assessment of the effectiveness of decision tree methods for land cover classification. Remote Sens. Environ. 86 (4), 554–565. http://dx.doi.org/10.1016/S0034-4257(03)00132-9.
Pelletier, C., Valero, S., Inglada, J., Champion, N., Dedieu, G., 2016. Assessing the robustness of Random Forests to map land cover with high resolution satellite image time series over large areas. Remote Sens. Environ. 187, 156–168. http://dx.doi.org/10.1016/j.rse.2016.10.010.
Pontius, R.G., Millones, M., 2011. Death to Kappa: birth of quantity disagreement and allocation disagreement for accuracy assessment. Int. J. Remote Sens. 32 (15), 4407–4429. http://dx.doi.org/10.1080/01431161.2011.552923.
Puertas, O.L., Brenning, A., Meza, F.J., 2013. Balancing misclassification errors of land cover classification maps using support vector machines and Landsat imagery in the Maipo river basin (Central Chile, 1975–2010). Remote Sens. Environ. 137, 112–123. http://dx.doi.org/10.1016/j.rse.2013.06.003.
Qing, J., Huo, H., Fang, T., 2010. Supervised land cover classification based on the locally reduced convex hull approach. Int. J. Remote Sens. 31 (8), 2179–2187. http://dx.doi.org/10.1080/01431161003636708.
Roelfsema, C., Phinn, S., 2010. Integrating field data with high spatial resolution multispectral satellite imagery for calibration and validation of coral reef benthic community maps. J. Appl. Remote. Sens. 4 (1), 043527-28–043527. http://dx.doi.org/10.1117/1.3430107.
Tuia, D., Volpi, M., Copa, L., Kanevski, M., Munoz-Mari, J., 2011. A survey of active learning algorithms for supervised remote sensing image classification. IEEE J. Sel. Top. Signal Process. 5 (3), 606–617. http://dx.doi.org/10.1109/JSTSP.2011.2139193.
Turner, M.G., 2005. Landscape ecology: what is the state of the science? Annu. Rev. Ecol. Evol. Syst. 36 (1), 319–344. http://dx.doi.org/10.1146/annurev.ecolsys.36.102003.152614.
Weng, Q., 2012. Remote sensing of impervious surfaces in the urban areas: Requirements, methods, and trends. Remote Sens. Environ. 117, 34–49. http://dx.doi.org/10.1016/j.rse.2011.02.030.
Wuttke, S., Middelmann, W., Stilla, U., 2016. Active Learning With SVM for Land Cover Classification-What Can Go Wrong? (Retrieved from). https://pdfs.semanticscholar.org/6cd0/0aaed7eec8e83611d272345399e99c6e8da2.pdf.
Zhang, L., Zhang, L., Du, B., 2016. Deep learning for remote sensing data: a technical tutorial on the state of the art. IEEE Geosc. Remote Sens. Mag 4 (2), 22–40. http://dx.doi.org/10.1109/MGRS.2016.2540798.
Zhong, Y., Zhang, L., Gong, J., Li, P., 2007. A supervised artificial immune classifier for remote-sensing imagery. IEEE Trans. Geosci. Remote Sens. 45 (12), 3957–3966. http://dx.doi.org/10.1109/TGRS.2007.907739.
Zhu, Z., Gallant, A.L., Woodcock, C.E., Pengra, B., Olofsson, P., Loveland, T.R., ... Auch, R.F., 2016. Optimizing selection of training and auxiliary data for operational land cover classification for the LCMAP initiative. ISPRS J. Photogramm. Remote Sens. 122, 206–221. http://dx.doi.org/10.1016/j.isprsjprs.2016.11.004.