Dimensionality Reduction and Bandwidth Selection for Spatial Kernel Discriminant Analysis

Soumia Boumeddane1, a, Leila Hamdad1, b, Hamid Haddadou1, c and Sophie Dabo-Niang2, d

1 Laboratoire de la Communication dans les Systèmes Informatiques, Ecole Nationale Supérieure d’Informatique, BP 68M, 16309, Oued-Smar, Algiers, Algeria
2 Univ. Lille, CNRS, UMR 8524, Laboratoire Paul Painlevé, F-59000 Lille, France

Keywords: Spatial Kernel Discriminant Analysis, Feature Extraction, Principle Component Analysis, Particle Swarm Optimization, Hyperspectral Image Classification.

Abstract: Spatial Kernel Discriminant Analysis is a powerful tool for the classification of spatially dependent data. It allows taking into consideration the spatial autocorrelation of data based on a spatial kernel density estimator. The performance of SKDA is highly influenced by the choice of the smoothing parameters, also known as bandwidths. Moreover, computing a kernel density estimate is computationally intensive for high-dimensional datasets. In this paper, we consider the bandwidth selection as an optimization problem, that we resolve using Particle Swarm Optimization algorithm. In addition, we investigate the use of Principle Component Analysis as a feature extraction technique to reduce computational complexity and overcome curse of dimensionality drawback. We examined the performance of our model on Hyperspectral image classification. Experiments have given promising results on a commonly used dataset.

1 INTRODUCTION

Probability density estimation is a key concept for many machine learning tasks and real-world applications, such as hotspot detection (of pandemics, crimes and accidents), wind speed prediction, cluster analysis, images analysis ... etc. Kernel density estimation is a popular non-parametric density estimation method with a well-known application in Kernel Discriminant Analysis (KDA). In this work, we focus on Spatial Kernel Discriminant Analysis (SKDA), proposed by (Boumeddane et al., 2019) and (Boumeddane et al., 2020). SKDA is a supervised classification algorithm accounting spatial dependency of data. This algorithm is built using Spatial Kernel Density Estimation (SKDE) (Dabo-Niang et al., 2014) which includes two kernels: one controls the observed values while the other controls the spatial locations of observations. Since SKDA is based of a kernel density estimation technique, its performance is highly influenced, in one hand, by the choice of the smoothing parameters, also known as bandwidths. Bandwidth selection consists of finding the optimal values that minimize the error between the estimated and the real density, most proposed methods are costly since they involve brute-force or exhaustive search strategies (Zaman et al., 2016). Moreover, these methods may note be suitable for classification purpose (Ghosh and Chaudhuri, 2004). In (Dabo-Niang et al., 2014), cross-validation was used to determine the best bandwidths for SKDE from a list of proposed values, for a clustering application. Moreover, in (Boumeddane et al., 2020), the bandwidths were determined experimentally using a grid-search like approach.

In addition to bandwidth selection problem in kernel density estimation, the choice of the most relevant features is also crucial. In fact, the rise of high performance technologies has resulted in an exponential increase in collected data, in terms of both size and dimensionality. Nonetheless, these data usually contain a high level of noise with irrelevant or redundant features which can lead to lower classification accuracy and an unnecessary increase in computational costs and storage.

An optimal choice of features can improve the estimation performances and the learning speed, which affects the classification performance of a kernel...
density estimator based classifier (Sheikhpour et al., 2016). Moreover, Kernel density estimation is computationally intensive for high dimensional data, due to the design of this estimator. Dimensionality reduction has been proven to be efficient to preprocess high dimensional data and to remove noisy (i.e. irrelevant) and redundant features. It aims to reduce the complexity of a model and build a more comprehensible, simpler and understandable data (Li et al., 2018).

In this study, we address the problem of bandwidth selection for Spatial Kernel Discriminant Analysis in a context of high-dimensional feature space. We propose a new hybrid approach to resolve both dimensionality reduction and bandwidth selection for Spatial Kernel Discriminant Analysis. Using Principal Component Analysis (PCA) as a feature extraction technique which consist of projecting data to a new feature subspace with lower dimensionality. Moreover, we consider the bandwidth selection as an optimisation problem, that we resolve using Particle Swarm Optimization, a powerful and efficient population-based optimization technique which has been successfully applied in many complex optimisation problems (Wang et al., 2018).

We validate our approach on hyperspectral image (HSI) classification. These images provide rich information despite other remote sensing technologies. However, this technique offers a large spectral vectors with hundreds of wavelengths including possible irrelevant or redundant data. This increases significantly the computation time and model complexity of classification algorithms.

This paper is organized as follows: In Section 2, we present kernel density estimation and highlight the effect of the bandwidth selection. In section 3, we give a brief overview of Spatial Kernel Discriminant Analysis (SKDA). Section 4 is dedicated to related work. In section 5, we present Principle component Analysis and Particle Swarm Optimisation. Then, we explain in section 6 our proposed approach for dimensionality reduction and the selection of SKDA bandwidths. Finally, we present the experimental results of our approach for HSI classification.

2 KERNEL DENSITY ESTIMATION

Suppose we have a random variable $X$ with a probability density function $f$. Typically, the precise form of the density function $f$ is not known and needs to be estimated. Kernel density estimation (KDE), also known as Parzen Window method (Silverman, 1986) is a common approach to compute a non-parametric estimates of the probability density functions $f$. This non-parametric nature makes KDE a flexible and attractive method, since it can estimate the density directly from data samples without any assumptions about the form of the distribution (Gramacki, 2018).

Let $X$ be a random variable with an unknown probability density function $f$. Given $n$ $d$-dimensional, independent and identically distributed samples $X_1, X_2, ..., X_n$ of $X$, the kernel estimate of the density function $f$ denoted $\hat{f}$ is given by:

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K \left( \frac{x - X_i}{h} \right), x \in \mathbb{R}^d$$ (1)

Where: $h$ is a hyperparameter named the bandwidth which controls the amount of smoothing, and $K$ is a smoothing function called the kernel function, that assigns a weight according to the distance between $X_i$ and $x$.

Fig. 1 illustrates a set of data points (black vertical lines on the $x$-axis) and their individual kernels (red lines) computed using Gaussian kernel. The overall density (blue curve) is computed by summing these individual functions according to Equation 1.

![Figure 1: Kernel density estimate (KDE).](image)

The bandwidth $h$ plays the role of a smoothing scale, and determines how much emphasis is put on the closest points. It is well recognized that the choice of the bandwidth is a crucial problem for KDE (Gramacki, 2018). This problem is often known as bandwidth selection.

To illustrate the effect of this hyperparameter, Fig. 2 shows the kernel density estimate of a random sample of 1000 samples from a Gaussian distribution using different bandwidth values. As we can see, the most appropriate estimation has been obtained when $h = 0.22$. However, when $h$ is too small, $(h = 0.03)$ the density is undersmoothed. On the contrast, for a large value of the bandwidth $(h = 1.5)$, the density is oversmoothed (Chen, 2017).
Figure 2: Kernel density estimate (KDE) with different bandwidths.

One of the well-known applications of Kernel Density Estimation is its use to build a supervised classifier upon Bayes’ theorem, called Kernel Discriminant Analysis (KDA). In fact, for a training set of \( n \) \( d \)-dimensional samples grouped into \( m \) classes, the Bayes discriminant rule consists of assigning a \( d \)-dimensional observation \( x \) to the class \( k_0 \) with highest posterior probability, formulated as:

\[
k_0 = \arg \max_{k \in \{1, \ldots, m\}} \pi_k f_k(x)
\]  

(2)

Where \( f_k \) is the probability density function of \( x \) in the class \( k \) and \( \pi_k \) is the a priori probability that an observation belongs to the class \( k (k = 1, \ldots, m) \). Kernel Discriminant Analysis rule consists of using Kernel Density Estimation (Equation 1) to estimate the density functions \( f_k \).

3 SPATIAL KERNEL DISCRIMINANT ANALYSIS

Like most classification algorithms, Kernel Discriminant Analysis assumes that data samples are independent and identically distributed (i.i.d). However, this assumption is often violated in many real-life problems, for example, in georeferenced data containing a spatial dimension and characterized by spatial autocorrelation phenomena. This characteristic comes from the fact that more the objects are close to each other, the higher is the correlation between them (Miller, 2004).

In (Boumeddane et al., 2019) and (Boumeddane et al., 2020), the authors proposed an extension of Kernel Discriminant Analysis (KDA) for spatially dependent data, based on a Spatial Kernel Density Estimator (SKDE) proposed by (Dabo-Niang et al., 2014) which takes into account the spatial positions of data.

We consider a spatial process \( \{Z_s = (X_s, Y_s) \in \mathbb{R}^d \times [1, m], s \in \mathbb{Z}^2, d \in \mathbb{N}^+, m \in \mathbb{N}^+ \} \), representing a set of \( d \)-dimensional observations \( x_i \) measured each at a site \( s \) \((s \in \mathbb{Z}^2)\) and \( y_i \) is the class label \((1, \ldots m)\) to which \( x_i \) belongs.

Suppose that \( x_1^{(k)}, x_2^{(k)}, \ldots, x_m^{(k)} \) are \( d \)-dimensional observations from the \( k \)-th class \((k = 1, 2, \ldots, m)\) of the training set \((\text{of a total size} \ n)\), measured at the sites \( s_1, s_2, \ldots, s_m \) respectively.

An observation \( x_j \in \mathbb{R}^d \) located at a site \( s_j \), will be assigned to the class \( k_0 \), where:

\[
k_0 = \arg \max_{k \in \{1, \ldots, m\}} (\hat{f}_k(x_j))
\]  

(3)

The class density function \( f_k \) of the \( k \)-th class is estimated using the spatial kernel density estimator (SKDE) of (Dabo-Niang et al., 2014), defined as:

\[
\hat{f}_k(x_j) = \frac{1}{m_k h^d_v h^d_w} \sum_{i=1}^{m_k} K_d\left(\frac{x_j - x_i^{(k)}}{h_v}\right) K_s\left(\frac{\|s_j - s_i\|}{h_s}\right)
\]  

(4)

where:

- \( h_v \) and \( h_s \) are two bandwidths controlling, respectively, features and spatial neighbourhood,
- \( K_d \) and \( K_s \) are two kernels respectively defined in \( \mathbb{R}^d \) and \( \mathbb{R}^s \) where \( K_d \) manages observations’ values while \( K_s \) deals with the spatial dimension of data,
- \( \|s_j - s_i\| \) is the Euclidean distance between the sites \( s_j \) and \( s_i \).

4 RELATED WORK

In literature, a set of data-based bandwidth selectors have been proposed for kernel density estimation. The core idea behind these methods is to find the optimal bandwidth which minimises the Mean Integrated Squared Error (MISE) between the estimated and actual probability density function (Chen, 2017), formulated as:

\[
\text{MISE}(h) = E\left[\int (\hat{f}(x) - f(x))^2 dx\right]
\]  

(5)

These methods include: (1) Rules-of-thumb: such as, Silverman’s rule of thumb and Scott’s rule of thumb. These rules rely upon some assumption on the density, like the assumption of a normal density in Silverman’s rule, which makes these methods not totality data-driven (Zhou et al., 2018), (2) Cross-validation based approaches: including least square cross validation, biased cross validation and smoothed cross validation, (3) Plug-in approaches and (4) Bootstrap approaches (Chen, 2017).
These bandwidth selection techniques that aim to minimise the MISE may not be suitable for classification purpose, in other words the optimal bandwidth that minimise the MISE may not be optimal to minimise the misclassification rate (Ghosh and Chaudhuri, 2004). In addition, SKDA depends on two bandwidths related to features on one hand, and spatial neighbourhood on the other hand. These bandwidth selection methods do not consider the spatial dimension of data.

Few works has addressed dimensionality reduction and bandwidth selection for Kernel Discriminant Analysis. In (Sheikhpour et al., 2016), the authors proposed a hybrid model of Particle Swarm Optimization (PSO) and Kernel Discriminant Analysis using PSO metaheuristic for both features and optimal bandwith selection. This model was applied for breast cancer diagnosis. Moreover, (Baek et al., 2016) proposed an approach which integrates Kernel Discriminant analysis and the information theoretic measure of complexity (ICOMP) with genetic algorithm (GA), used simultaneously for features and bandwidth selection. Recently, (Sheikhpour et al., 2017) proposed a kernelized non-parametric classifier based on feature ranking in anisotropic Gaussian kernel (KNR-AGK). This approach uses features’ ranks for both feature selection and parameters learning of the anisotropic Gaussian kernel, considering these ranks as the kernel bandwidths of different dimensions.

5 BACKGROUND

In this section we present an overview of the two concepts that we will use in our hybrid method: Principle Component Analysis and Particle Swarm Optimisation.

5.1 Principle Component Analysis

Principal component analysis (PCA) is a dimensionality reduction technique, which projects a data set of a high-dimensional space into a lower-dimensional sub-space, with the goal of preserving most of the variation. It is a mathematical algorithm, which consists of finding a linear combination converting the original variables into a new set of uncorrelated variables (the principle components) which capture maximal variance. PCA can be performed via an eigen decomposition of the covariance matrix $C$ ($d \times d$) (Pu et al., 2014), given by:

$$ C = W \Lambda W^{-1} $$

where:

- $W$ is a ($d \times d$) matrix of eigenvectors, which represents the principle components,
- $\Lambda$ is a diagonal matrix of $d$ eigenvalues.

Principle components are ordered such that first ones has maximum variance. For a dimensionality reduction purpose, features with small eigenvalues may be dismissed and the data points will be projected onto the first $k$ Principle Components (van der Walt and Barnard, 2017).

5.2 Particle Swarm Optimisation

Particle Swarm Optimisation (PSO) is a nature-inspired metaheuristic originally proposed by Kennedy and Elberhart in 1995, that simulates the social behaviour of some animals such as insects, herds, schools of fish or flocks of birds which cooperate to find food. PSO is a population based stochastic optimization technique, which uses a population called swarm composed of $N$ particles that moves over the research space, iteration to iteration. Each particle has a position vector $X_i(x_{i1}, x_{i2}, \ldots, x_{id})$ and a velocity vector $v_i(v_{i1}, v_{i2}, \ldots, v_{id})$, where $i \in \{1, 2, \ldots, N\}$ and $d$ is the number of dimensions in the vector. Each particle represents a potential solution to the problem in the $d$-dimensional research space (Wang et al., 2018).

The optimization process starts with a randomly initialized population of solutions. During the research process of the optimal solution, each particle adjust its search direction toward a promising search region according to its own optimal experience ($p_{best}$) and the optimal experience of the swarm (called the global best: $g_{best}$). The performance of each particle is measured using a fitness function related to the problem definition.

The velocity and position of each particle are updated using the following equations (Wang et al., 2018):

$$ v_{i,j+1}^d = w v_{i,j}^d + c_1 r_1 (p_{i,j}^d - x_{i,j}^d) + c_2 r_2 (g_{best}^d - x_{i,j}^d) \quad (7) $$

$$ x_{i,j+1}^d = x_{i,j}^d + v_{i,j+1}^d \quad (8) $$

Where: $w$ is the inertia weight, $c_1$ and $c_2$ are acceleration constants, and $r_1$ and $r_2$ are uniformly random values between 0 and 1.

This process is repeated until a stop criteria is satisfied, such as a maximum number of iterations.
6 PROPOSED APPROACH

We summarize in Fig. 3 the steps of our proposed approach that we called PSO-SKDA.

Figure 3: PSO-SKDA approach.

**Step 1: PCA for Features Extraction.** We propose to apply PCA as a preprocessing step prior to SKDA. PCA will serve as a dimensionality reduction technique and also as bandwidth regularizer. As previously explained, only the first principle components with highest eigenvalues will be kept. Moreover, we performed PCA on non-spatial dimensions only, while preserving spatial information. In addition, we performed PCA on the training set samples, than apply the same linear transformation on the test set before the classification. The number of selected principal components will be denoted $d'$.

**Step 2 : PSO for Bandwidths Selection.** In this step we use PSO technique to carry out the selection of the optimal values of the bandwidths $h_s$ and $h_v$ which maximise the classification accuracy of SKDA. That is to say that, we defined the fitness function as the classification overall accuracy of SKDA and bandwidth selection consists of finding the optimal bandwidths which maximise the classification accuracy. Moreover, we encode the solutions as a $(d'+1)$-dimensional vector $H = [h_{v1}, h_{v2}, ..., h_{vd'}, h_s]$ where $d'$ is the number of the retained principal components and $h_{v}$ is the bandwidth of the $i$-th feature.

In each iteration of PSO, the objective function is calculated for all particles using SKDA, based on a $k$-fold cross validation on the training set. Figure 4 illustrates the concept of $k$-fold cross validation, which consists of splitting the training set into $k$ subsets of same size and using $k-1$ folds to train the model and the remaining fold for validation. The output of the model is the average of the values computed in the loop.

Figure 4: $k$-fold cross validation.

Also, the position of particles are updated using Equations 7 and 8, and global and per-particle best solutions are updated.

**Step 3: Classification with the Optimal Determined Kernel Bandwidths.** The final step consists of the classification of the transformed testing data using SKDA classifier based on the determined optimal kernel bandwidths.

7 EXPERIMENTS

7.1 Experimental Setup

We applied our approach for Hyperspectral image classification which consists of assigning a label (ex: water, forest ...) to a pixel. A hyperspectral image (HSI) is a set of simultaneous images collected for the same area on the surface of the earth with hundreds of spectral bands at different wavelength channels and with high resolution. Each pixel of a HSI at a position $s_i$ (representing the spatial dimension) is characterized by a $d$-dimensional spectral vector (representing the spectral dimension).

These images are characterised by the high number of spectral bands. Also, from a spectral point of view, pixels of the same materials have similar spectral signature (Benediktsson and Ghamisi, 2015). Moreover, a strong correlation exists between neighbouring bands of a hyperspectral image. This fact motivates the use of feature selection and feature extraction techniques to reduce the dimensionality of the
Moreover, from the spatial perspective, the spectral signatures are spatially correlated, which means that neighbouring pixels usually belong to the same material especially for high-resolution images. This characteristic known as spatial autocorrelation should be taken into account for an effective analysis.

We conducted our experiments on a 200-dimensional dataset named **Indian Pines** dataset, which consists of a 145 \times 145 pixels image over the Indian Pines site in Northwestern Indiana. The ground truth is classified into 16 classes, containing agriculture, forest and natural vegetation. We used standard training and test sets (Mou et al., 2018), widely used by the HSI classification community. This makes our results entirely comparable to state-of-the-art methods. Fig. 5 visualizes this image and the training and test samples that we used. Table 1 displays the size of each class of this dataset’s training and testing sets.

To evaluate the performance of our algorithm, we use the following measures: 1) Average Accuracy (AA) : representing the average value of the class classification accuracy (CA). 2) Overall accuracy (OA) : which represents the ratio of correctly classified samples, i.e. calculated as the number of correctly classified pixels, divided by the number of test samples. 3) Kappa Coefficient (k) : that provides information regarding the amount of agreement corrected by the level of agreement that could be expected due to chance alone.

### 7.2 Experimental Results

To decide how much principle components to keep, while retaining as much of the information as possible, we use the Cumulative Proportion of Variance Explained graph (Figure 6). We only keep the transformed features with highest eigenvalues such that the cumulative proportion of variance explained is about 98%. For the Indian Pines dataset with an original feature space of 200 variables, only 25 principle components are selected.

Moreover, we executed the PSO process to select the optimal bandwidth with 20 iterations where each swarm contains 20 particles. Using a 5-fold cross-validation and an Epanechnikov kernel for $K_\nu$ and $K_s$.

The parameters of PSO are initialized as follows: $c_1 = c_2 = 1.93$, $w = 0.72$. Also, the solution space is reduced to the values between 1 and 10.

Using our approach, the best solutions found are: $h_\nu = 6$ and $h_s = 3.5$, here we used same $h_\nu$ for all feature dimensions.

We compared PSO-SKDA results to state-of-the-art techniques related to Hyperspectral images classifi-
Table 2: Classification results (%) for Indian Pines dataset using standard training and test sets.

<table>
<thead>
<tr>
<th>Method</th>
<th>SVM</th>
<th>RF-200</th>
<th>RNN</th>
<th>1D-CNN</th>
<th>2D-CNN</th>
<th>SICNN</th>
<th>Res. C-D Net</th>
<th>RNN-GRU-PRetanh</th>
<th>SKDA</th>
<th>PSO-SKDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA1</td>
<td>64.31</td>
<td>54.84</td>
<td>64.74</td>
<td>61.34</td>
<td>82.51</td>
<td>79.84</td>
<td>74.86</td>
<td>70.59</td>
<td>86.99</td>
<td>89.74</td>
</tr>
<tr>
<td>CA2</td>
<td>70.92</td>
<td>58.42</td>
<td>61.35</td>
<td>60.33</td>
<td>88.14</td>
<td>92.47</td>
<td>95.28</td>
<td>70.82</td>
<td>98.72</td>
<td>97.07</td>
</tr>
<tr>
<td>CA3</td>
<td>84.78</td>
<td>82.61</td>
<td>74.46</td>
<td>80.43</td>
<td>100.0</td>
<td>99.46</td>
<td>100.0</td>
<td>81.52</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>CA4</td>
<td>94.05</td>
<td>85.91</td>
<td>83.45</td>
<td>89.04</td>
<td>94.85</td>
<td>93.29</td>
<td>95.08</td>
<td>90.16</td>
<td>94.41</td>
<td>88.14</td>
</tr>
<tr>
<td>CA5</td>
<td>85.94</td>
<td>80.49</td>
<td>77.04</td>
<td>90.53</td>
<td>85.80</td>
<td>92.68</td>
<td>96.56</td>
<td>91.97</td>
<td>97.42</td>
<td>95.84</td>
</tr>
<tr>
<td>CA6</td>
<td>93.62</td>
<td>94.76</td>
<td>87.70</td>
<td>96.13</td>
<td>99.77</td>
<td>96.58</td>
<td>99.09</td>
<td>96.13</td>
<td>100.0</td>
<td>99.77</td>
</tr>
<tr>
<td>CA7</td>
<td>69.17</td>
<td>77.34</td>
<td>76.03</td>
<td>72.11</td>
<td>82.35</td>
<td>86.82</td>
<td>84.42</td>
<td>84.75</td>
<td>87.15</td>
<td>90.63</td>
</tr>
<tr>
<td>CA8</td>
<td>52.90</td>
<td>59.43</td>
<td>60.79</td>
<td>54.47</td>
<td>73.86</td>
<td>69.52</td>
<td>74.57</td>
<td>59.64</td>
<td>85.98</td>
<td>85.98</td>
</tr>
<tr>
<td>CA9</td>
<td>76.60</td>
<td>63.48</td>
<td>61.17</td>
<td>75.71</td>
<td>86.00</td>
<td>83.69</td>
<td>80.14</td>
<td>86.17</td>
<td>87.23</td>
<td>88.30</td>
</tr>
<tr>
<td>CA10</td>
<td>97.53</td>
<td>96.06</td>
<td>93.21</td>
<td>99.83</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>99.38</td>
<td>100.0</td>
<td>99.38</td>
</tr>
<tr>
<td>CA11</td>
<td>77.49</td>
<td>88.26</td>
<td>81.67</td>
<td>80.87</td>
<td>94.53</td>
<td>96.70</td>
<td>95.74</td>
<td>84.97</td>
<td>97.51</td>
<td>98.07</td>
</tr>
<tr>
<td>CA12</td>
<td>73.33</td>
<td>54.85</td>
<td>55.45</td>
<td>78.48</td>
<td>97.27</td>
<td>96.97</td>
<td>96.06</td>
<td>77.58</td>
<td>99.70</td>
<td>99.09</td>
</tr>
<tr>
<td>CA13</td>
<td>100.0</td>
<td>97.78</td>
<td>86.67</td>
<td>91.11</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>95.56</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>CA14</td>
<td>97.91</td>
<td>58.97</td>
<td>69.23</td>
<td>94.87</td>
<td>97.44</td>
<td>94.87</td>
<td>84.62</td>
<td>84.62</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>CA15</td>
<td>90.91</td>
<td>81.82</td>
<td>90.91</td>
<td>90.91</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>90.91</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>CA16</td>
<td>100.0</td>
<td>100.0</td>
<td>80.00</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>OA</td>
<td>70.55</td>
<td>69.79</td>
<td>69.82</td>
<td>70.79</td>
<td>85.43</td>
<td>85.13</td>
<td>85.76</td>
<td>88.63</td>
<td>91.79</td>
<td>92.07</td>
</tr>
<tr>
<td>AA</td>
<td>82.23</td>
<td>77.13</td>
<td>75.24</td>
<td>82.23</td>
<td>92.66</td>
<td>92.68</td>
<td>92.28</td>
<td>85.63</td>
<td>95.79</td>
<td>95.75</td>
</tr>
<tr>
<td>κ</td>
<td>66.90</td>
<td>65.89</td>
<td>65.87</td>
<td>67.07</td>
<td>83.49</td>
<td>83.13</td>
<td>83.85</td>
<td>73.66</td>
<td>90.62</td>
<td>90.94</td>
</tr>
</tbody>
</table>

Table 3: Comparison between SKDA, PCA-SKDA and PSO-SKDA for Indian Pines dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>SKDA</th>
<th>PCA-SKDA</th>
<th>PSO-SKDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>OA</td>
<td>93.89</td>
<td>92.71</td>
<td>92.07</td>
</tr>
<tr>
<td>AA</td>
<td>96.67</td>
<td>96.13</td>
<td>95.75</td>
</tr>
<tr>
<td>κ</td>
<td>93.01</td>
<td>98.18</td>
<td>90.94</td>
</tr>
</tbody>
</table>

REFERENCES


Benediktsson, J. A. and Ghamisi, P. (2015). *Spectral-spatial...
classification of hyperspectral remote sensing images.
Artech House.