Wp

WORKING PAPERS

EXPLORING THE POTENTIAL OF MACHINE LEARNING FOR AUTOMATIC SLUM IDENTIFICATION FROM VHR IMAGERY
23/11/2016
N° 2016/13

EXPLORING THE POTENTIAL OF MACHINE LEARNING FOR AUTOMATIC SLUM IDENTIFICATION FROM VHR IMAGERY

Duque, Juan Carlos
Patiño, Jorge Eduardo
Betancourt, Alejandro

CAF DEVELOPMENT BANK OF LATIN AMERICA

EXPLORING THE POTENTIAL OF MACHINE LEARNING FOR AUTOMATIC SLUM IDENTIFICATION FROM VHR IMAGERY

Duque, Juan Carlos
Patiño, Jorge Eduardo
Betancourt, Alejandro

CAF – Working paper N° 2016/13
23/11/2016

ABSTRACT

Slum identification in urban settlements is a crucial step in the process of formulation of pro-poor policies. However, the use of conventional methods for slums detection such as field surveys may result time consuming and costly. This paper explores the possibility of implementing a low-cost standardized method for slum detection. We use spectral, texture and structural features extracted from very high spatial resolution imagery as input data and evaluate the capability of three machine learning algorithms (Logistic Regression, Support Vector Machine and Random Forest) to classify urban areas as slum or no-slum. Using data from Buenos Aires (Argentina), Medellin (Colombia), and Recife (Brazil), we found that Support Vector Machine with radial basis kernel deliver the best performance (over 0.81). We also found that singularities within cities preclude the use of a unified classification model.

Small sections of text, that are less than two paragraphs, may be quoted without explicit permission as long as this document is stated. Findings, interpretations and conclusions expressed in this publication are the sole responsibility of its author(s), and it cannot be, in any way, attributed to CAF, its Executive Directors or the countries they represent. CAF does not guarantee the accuracy of the data included in this publication and is not, in any way, responsible for any consequences resulting from its use.

CAF DEVELOPMENT BANK OF LATIN AMERICA

Wp WORKING PAPERS

EXPLORACIÓN DEL POTENCIAL DEL APRENDIZAJE AUTOMÁTICO PARA LA IDENTIFICACIÓN AUTOMÁTICA DE ASENTAMIENTOS URBANOS INFORMALES CON IMÁGENES SATELITALES DE ALTA RESOLUCIÓN

Duque, Juan Carlos
Patiño, Jorge Eduardo
Betancourt, Alejandro

CAF - Documento de trabajo N° 2016/13
23/11/2016

RESUMEN

La identificación de los asentamientos urbanos informales es un paso crucial en el proceso de formulación de políticas dirigidas a mitigar la pobreza. Sin embargo, el uso de métodos convencionales para la detección de asentamientos informales, como las encuestas de campo, puede resultar lento y costoso. Este artículo explora la posibilidad de implementar un método estandarizado, de bajo costo, para la detección de asentamientos urbanos informales. Utilizamos como datos de entrada variables espectrales, estructurales y de textura extraídas de imágenes satelitales de alta resolución, y evaluamos la capacidad de tres algoritmos de aprendizaje automático (Regresión Logística, Máquinas de Vectores de Soporte y Bosque Aleatorio) para clasificar las áreas urbanas entre asentamientos formales o informales. Usando los datos de Buenos Aires (Argentina), Medellín (Colombia) y Recife (Brasil), encontramos que la Máquinas de vectores de soporte, con kernel radial, proporcionan el mejor desempeño (más de 0,81). También encontramos que las singularidades dentro de las ciudades impiden el uso de un modelo de clasificación unificado.

Small sections of text, that are less than two paragraphs, may be quoted without explicit permission as long as this document is stated. Findings, interpretations and conclusions expressed in this publication are the sole responsibility of its author(s), and it cannot be, in any way, attributed to CAF, its Executive Directors or the countries they represent. CAF does not guarantee the accuracy of the data included in this publication and is not, in any way, responsible for any consequences resulting from its use.

Exploring the potential of machine learning for automatic slum identification from VHR imagery.

Juan C. Duque · Jorge E. Patino · Alejandro Betancourt

November 23, 2016

Juan C. Duque (✉)
Research in Spatial Economics (RiSE-group), Department of Economics, Universidad EAFIT,
Carrera 49 7 Sur - 50, Medellin, Colombia
e-mail: jduquec1@eafit.edu.co
Phone: (574) 261-9354, Fax: (574) 261-9294

Jorge E. Patino
Research in Spatial Economics (RiSE-group), Department of Economics, Universidad EAFIT,
Carrera 49 7 Sur - 50, Medellin, Colombia
e-mail: jpatinoq@eafit.edu.co

Alejandro Betancourt
Department of Engineering (DITEN), University of Genova, Genova, Italy
Department of Industrial Design, Eindhoven University of Technology, Eindhoven, Netherlands
e-mail: a.betancourt@tue.nl

Exploring the potential of machine learning for automatic slum identification from VHR imagery

Abstract

Slum identification in urban settlements is a crucial step in the process of formulation of pro-poor policies. However, the use of conventional methods for slums detection such as field surveys may result time consuming and costly. This paper explores the possibility of implementing a low-cost standardized method for slum detection. We use spectral, texture and structural features extracted from very high spatial resolution imagery as input data and evaluate the capability of three machine learning algorithms (Logistic Regression, Support Vector Machine and Random Forest) to classify urban areas as slum or no-slum. Using data from Buenos Aires (Argentina), Medellin (Colombia), and Recife (Brazil), we found that Support Vector Machine with radial basis kernel deliver the best performance (over 0.81). We also found that singularities within cities preclude the use of a unified classification model.

Keywords: Remote Sensing · Slum detection · Machine Learning

JEL Classification: C8 · R14

1 Introduction

According to UN-Habitat (2015) slums are the most deprived and excluded form of informal settlements characterized by poverty and agglomerations of inadequate housing often located in hazardous urban land. About one in eight people live in slums by 2016 and even though the recent progress in improving slums and preventing their formation, the absolute number of urban population living in slums continue to grow and it is a critical factor for the persistence of poverty in the world (UN-Habitat, 2016).

There is a recent boom in the number of studies regarding the usefulness of remote sensing imagery to measure socioeconomic variables (Duque et al., 2015; Engstrom et al., 2015; Sandborn and Engstrom, 2016). This trend is in part the result of the increasing availability of satellite platforms and the decreasing costs of these imagery (Patino and Duque, 2013; Kuffer et al., 2016a). Remote sensing imagery may become an alternative source of information in urban settings where survey data is scarce, or to complement socioeconomic data for different dates than those of socioeconomic surveys (Duque et al., 2015). The use of remote sensing data to estimate socioeconomic variables is based in the premise that the physical appearance of a human settlement is a reflection of the society that created it and on the assumption that people living in urban areas with similar physical housing conditions have similar social and demographic characteristics (Jain, 2008; Taubenböck et al., 2009).

Slum detection or slum mapping is one of the most recurrent applications in this field, with at least 87 published papers in scientific journals in the last 15 years (Kuffer et al., 2016a). These works have demonstrated that the physical characteristics of slums are distinguishable from those of formal settlements using remote sensing data (Kuffer and Barros, 2011; Kuffer et al., 2016b; Taubenböck and Kraff, 2014). This is an important area of study because many local governments do not fully acknowledge the existence of slums or informal settlements (UN-Habitat, 2015); which is a drawback for the formulation of pro-poor policies in cities (Kuffer et al., 2016a).

The methods for the identification of slum areas from remote sensing imagery are diverse. Object based image analysis (OBIA) is the most used method until now, followed by visual interpretation, texture/morphology analysis and machine learning, which shows higher accuracies and it is often combined with OBIA (Kuffer et al., 2016a). Machine learning (ML) approaches usually combine textural, spectral and structural features (Kuffer et al., 2016b). The Random Forest Classifier (RFC) is one of the most popular ML methods for slum extraction from very high spatial resolution (VHR) imagery (Kuffer et al., 2016b). Support Vector Machine (SVM) and Neural Networks (NN) are also used (Kuffer et al., 2016a). However most of these ML algorithms are implemented at pixel level, which have limitations when working with VHR imagery as opposed to OBIA (Kohli et al., 2016); and the selection of the ML methods to be used is often done following the intuition of the researcher.

According to Kuffer et al. (2016a), most of the published works in slum mapping from remote sensing used expensive commercial imagery with near-infrared (NIR) information (Owen and Wong, 2013) or three dimensional data such as LIDAR (Taubenböck and Kraff, 2014). Many small cities in developing countries do not have available budget to purchase full satellite imagery, and restrict themselves to use RGB data for data extraction via interpretation (Kohli et al., 2016). Google Earth (GE) imagery may be the only available source of aerial imagery for small local governments, and GE images are free to the public (Yang et al., 2012; Hu et al., 2013). Also, Google Earth provides historical VHR imagery for many places, which could be useful for spatio-temporal urban analysis.

The purpose of this paper is threefold: First, we explore the possibility of detecting slums within cities using solely very high spatial resolution (VHR) RGB GE imagery without ancillary data. Second, we compare the performance of different ML algorithms for slum identification, given the same inputs, to determine which performs better. Third, we seek to implement a low-cost standardized method to perform slum detection, which must be flexible and easy to automate, and which could be useful in other urban settings with scarce data. We use data for three Latin American cities with different physical and climate conditions as well as different urban layout characteristics: Buenos Aires (Argentina), Medellin (Colombia), and Recife (Brazil).

The structure of this chapter is as follows: section 2 describes the implemented methodology including a description of the data, the three classification models utilized in this paper. Section 3 shows the results and discussion of the implemented approach. Section 4 presents the main conclusions, suggestions for future research and policy making implications of this line of research for local governments and authorities.

2 Methods

Our goal is to design an algorithmic structure to automatically identify the parts of the city with the urban characteristics of a Slum. This problem can be defined as a binary classification problem on which the inputs are features extracted from the GE images and the output a binary variable taking the value of 1 if a particular part of the city is slum and 0 otherwise. Figure 1 summarizes the proposed approach for slum detection. The process starts with the collection of the input data that consists of the administrative boundaries, obtained from Open Street Maps (OSM), and the GE images for two different time instances (upper part of the figure). The second stage of the process (middle part of the figure) consist of calculating spectral, textural and structural variables (i.e., the image feature extraction) from the GE images. For this, the images are discretized by overlapping a regular grid whose outer border is defined by the OSM boundary. This procedure generates the Spatial Datasets (one per year and city) composed by regular polygons with their corresponding spectral, textural and structural variables. Finally, the third stage (lower part of the figure) consists of the classification analysis on which the data of the most recent year is used to train the classification models and identify the best-performing model for slum identification. The best model is then applied on the images from previous years to identify urban changes in the most critical areas of the city.

2.1 The Data

We selected three different Latin American cities to test the transferability of this approach: Buenos Aires, Argentina; Medellin, Colombia; and Recife, Brazil (figure 2). These cities represent different climate and environmental conditions, different cultures and different availability of building materials. Buenos Aires is located at $34° \ 35' \ 59''$ S $58° \ 22' \ 55''$ W at sea level bordering La Plata river outlet to the ocean over plain lands and has a dry climate with marked seasons. Medellin is located at $6° \ 14' \ 41''$ N $75° \ 34' \ 29''$ W in an intramountain valley at 1,460 meters above mean sea level and has a tropical wet climate. Recife is located at $8° \ 03' \ 14''$ S $34° \ 52' \ 51''$ W at sea level in a hilly terrain and also has tropical wet climate. Slum areas in the three cities share some common characteristics: crowdness, small dwelling sizes, very narrow roads, and few vegetated areas. Table 1 shows general descriptors of these cities.

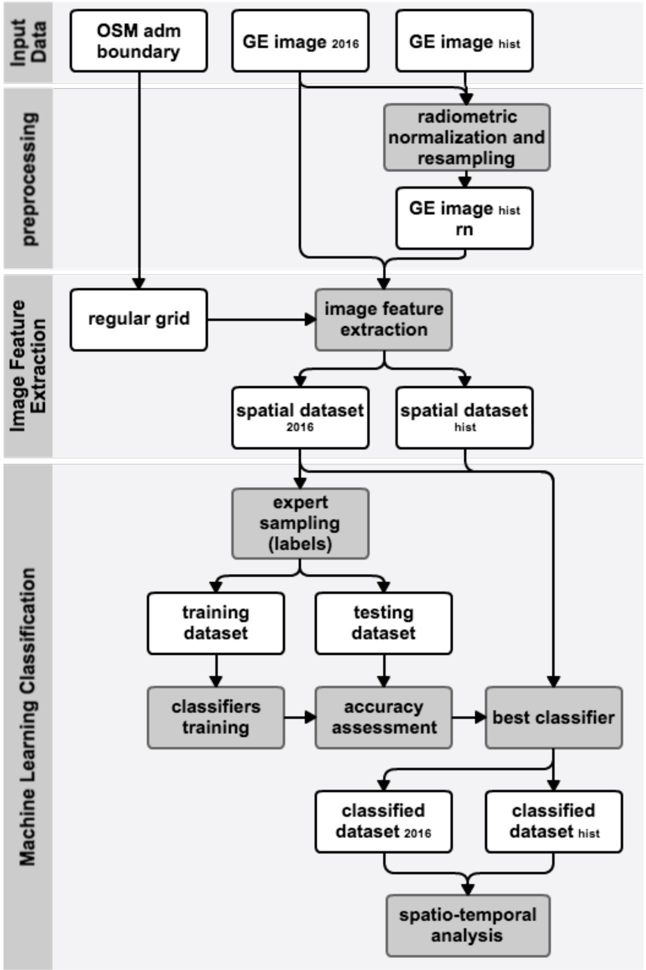


Figure 1. Flow diagram of the proposed approach for slum detection and change analysis.

Variable	Buenos Aires	Medellin	Recife
Population estimates for 2010 [people]	2,890,151	2,309,446	1,537,794
Area [Km2]	200	105	218
Density [people/Km2]	14,451	21,995	7,040
Elevation [masl]	0	1,480	0
Mean annual temperature [C]	18	24	25
Average relative humidity [%]	72.0	68.3	79.8

Table 1. Descriptives per city. Data sources: www.buenosaires.gob.ar/laciudad/ciudad, www.medellin.gov.co, and cidades.ibge.gov.br

We downloaded the most updated (up to March 2016) GE images for each city with enough zoom level to be similar to VHR imagery with sub-meter pixel size. Google Earth imagery with very high spatial resolution are available for almost all urban areas worldwide. Those VHR images come from a number of providers or satellite platforms; e.g., Digital Globe, Geo Eye, and CNES / Astrium, among others. The images are captured by different sensors and on different dates and with different spatial resolution, but almost all have submeter pixel size and they are served as natural-colored images with three bands: red, green and blue (RGB). Due to the differences in platforms and dates of adquisition, images of a same place from different dates can show differences

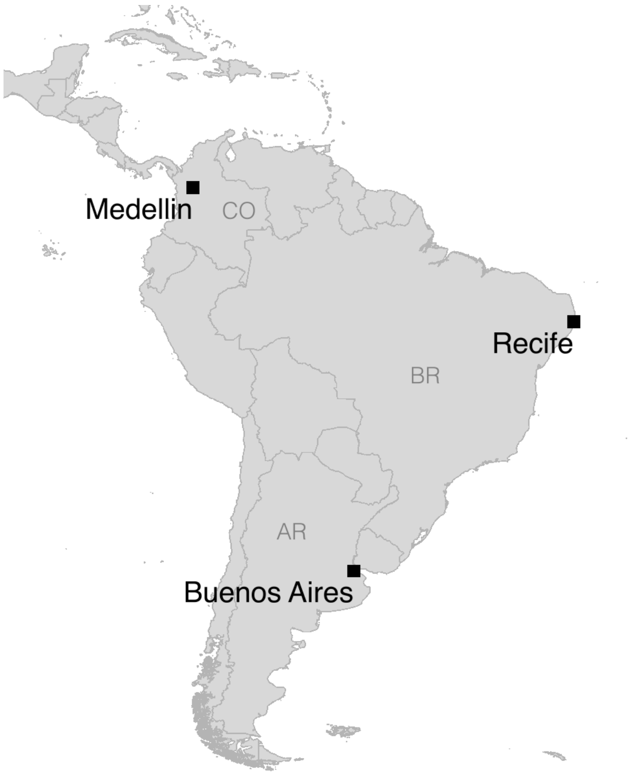


Figure 2. Cities included in the study.

in illumination conditions and color intensity between them. The GE images were georreferenced and rescaled between 0 and 255. We kept the preprocessing of the images to a minimum to gain speed in the workflow and to mantain the ease of automation of the whole approach.

Previous research has stated that block-level spatial units of analysis are most useful for urban planing purposes (Taubenböck and Kraff, 2014; Cabrera-Barona et al., 2016). OpenStreetMap (OSM) data layers of streets and roads are useful to delineate urban blocks. However, in many cities of developing countries the available street network is incomplete due to the high density and complexity of slum areas (Taubenböck and Kraff, 2014), or because the areas that have been occupied recently have not been registered at all in the OSM datasets, as is the case in the northeastern part of Medellin city. In those instances the delineation of urban blocks would add considerable processing time to the approach as it would require the visual interpretation and manual digitalization of roads and pedestrian paths to delineate the blocks.

An alternative that is easy to automate is the use of a regular grid to perform slum detection from remote sensing imagery. Previous works have used regular grids to extract, aggregate and classify image data (Kuffer et al., 2016a; Schöpfer et al., 2007; Tapiador et al., 2011). A regular grid in vector format, or fishnet, can be drawn in any GIS software and the required input is solely the boundary of the study area, thus adding speed to this approach. We tested the use of two fishnets with different polygon size: a fishnet of square cells of 100 meters each side and another fishnet of square cells of 50 meters each side for image feature extraction and classification. The results obtained with the 100 m grid outperformed those obtained by using the 50 m grid in regard to the correct classification of slum-like areas. These 100 m square cells are similar in size to actual urban bolcks, which have been recognized as a useful spatial unit of analysis to study intra-urban poverty and to ease urban planning and policy making (Cabrera-Barona et al., 2016). We downloaded the administrative boundarys of each city from OSM usign QGIS (QGIS Development Team, 2015) and then we created a regular grid of square cells of 100 meters each side over the urban areas of each city to perform image feature extraction.

We also selected recognized slum areas for each city and then downloaded cloud-free GE images for each sector from about a decade ago to test the approach for change analysis of slum areas. We seeked to have images from the same city separated almost a decade to see if the proposed approach could pick up the changes between dates. This time span is restricted by the availability of VHR images in Google Earth for each city and by the quality of the available images, which can be affected by the presence of clouds and shadows. The historical VHR imagery served in GE is also restricted to the availability of commercial VHR data, which started to be released after the launch of the Ikonos satellite in 1999. The most dated good quality VHR images available for Buenos Aires, Medellin, and Recife are from 2006, 2008, and 2008 respectively. Even though there are available images from previous dates for these cities in GE, those images come from medium spatial resolution platforms and they are not a suitable source for the extraction of spatial pattern descriptors at intra-urban scale.

The historical GE images were then resampled to the same pixel size than the 2016 images of each city, and we performed radiometric normalization between the historical images and the 2016 image, taking the 2016 images as reference. Resampling and radiometric normalization were performed to obtain historical images with the same pixel size and similar color intensity as the 2016 images (i.e., pixel values in each RGB band). This preprocessing of historical images is aimed to ease the identification of actual changes instead of changes in intensity due to differences in illumination and atmospheric conditions between the two images.

2.1.1 Feature Extraction

We used current GE images (March 2016) and the regular grid of each city to automatically extract image information using FETEX 2.0. Figure 3 shows the outline of the urban areas of each city and selected sectors of 500 by 500 meters to illustrate the regular grid over the 2016 GE images. FETEX is an interactive computer package for image, object-oriented feature extraction (Ruiz et al., 2011), and it is available at the Geo-Environmental Cartography and Remote Sensing Research Group website: http://cgat.webs.upv.es. We calculated three sets of variables: a set of spectral features, a set of texture features and a set of structural features. The different image features are extracted from the image by processing the pixels that are located within the same polygon, without changing the image resolution or pixel values. Spectral features inform about color, while texture and structure features inform about the spatial arrangement of the elements in within the image. The urban layout in slum-like neighborhoods often displays a more organic, crowded and cluttered pattern than that in more formal and wealthy neighborhoods, and texture and structure features are useful to differentiate between slum and no-slum areas (Duque et al., 2015; Kuffer et al., 2015, 2016b). Table 2 shows the complete list of remote sensing variables used in this analysis.

Spectral features: The spectral features are the summary statistics of pixel values inside each polygon. These features inform about the spectral response of objects, which depends on land coverage types, state of vegetation, soil composition, building materials, etc. (Ruiz et al., 2011). We selected the mean and standard deviation in each RGB band, as well as the majority statistic, to be extracted within this group. These features are easy to understand and could inform better about the spectral differences across the cities than the other summary statistics (minimum,

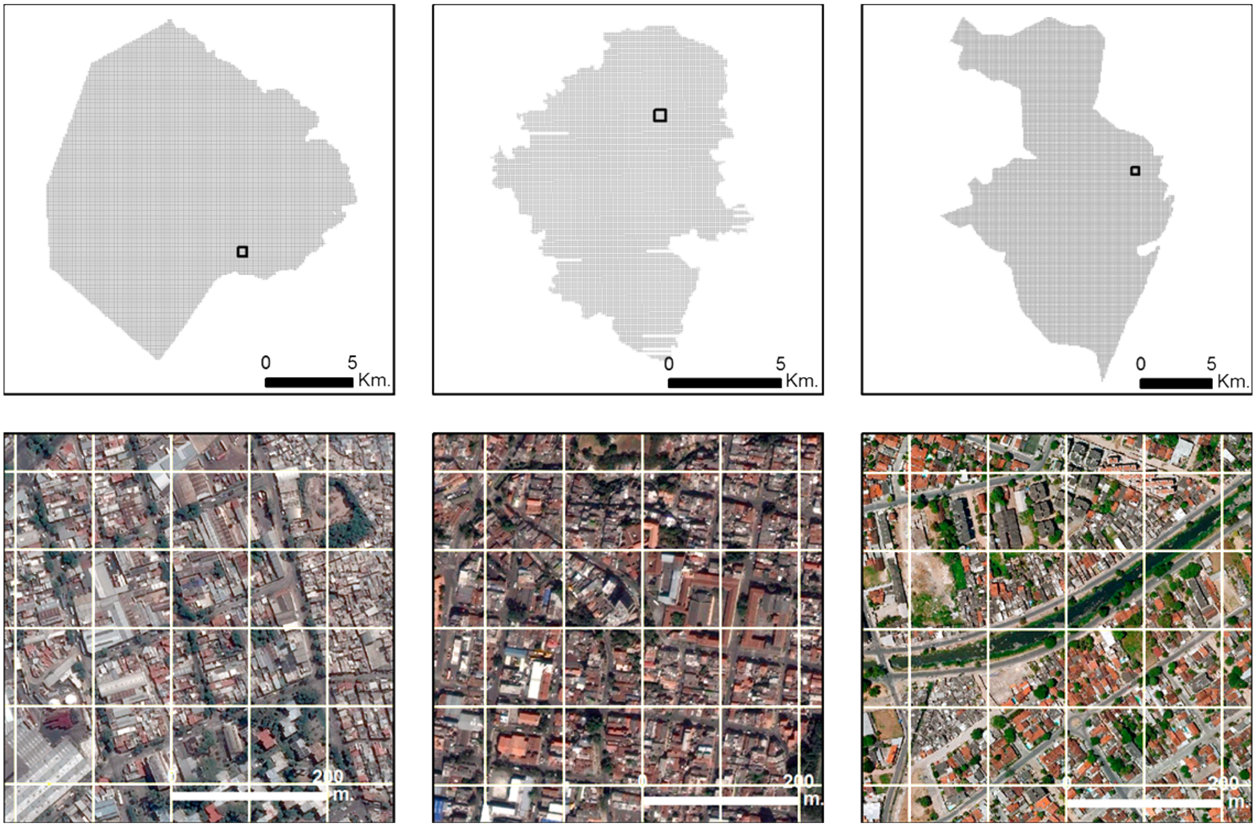


Figure 3. Urban areas and selected sectors showing the regular grid over the 2016 GE images of each city. From left to right: Buenos Aires, Medellin, Recife.

maximum, range, and sum).

Texture features: Texture features characterize the spatial distribution of intensity values in the image and provide information about contrast, uniformity, rugosity, etc. (Ruiz et al., 2011). FETEX 2.0 performs texture feature extraction based on the Grey Level Co-occurrence Matrix (GLCM) and the histogram of pixel values inside each polygon. The kurtosis and skewness features are based on the histogram of the pixel values inside the polygon, while the GLCM describes the co-occurrences of the pixel values that are separated at a distance of one pixel inside the polygon, and it is calculated considering the average value of four principal orientations, 0°, 45°, 90° and 135°, to avoid the influence of the orientation of the elements inside the polygon (Ruiz et al., 2011). The GLCM is employed in FETEX 2.0 to calculate a set of variables, proposed by Haralick et al. (1973), that are widely used in image processing, including uniformity, entropy, contrast, inverse difference moment (IDM), covariance, variance, and correlation. The edgeness factor is another useful feature that represents the density of edges present in a neighborhood, and the mean and standard deviation of the edgeness factor (MEAN_EDG, STDEV_EDG) are also computed within this set of texture features in FETEX 2.0 (Ruiz et al., 2011).

Structural features: These features provide information of the spatial arrangement of elements inside the polygons in terms of randomness or regularity of the distribution of the elements (Balaguer et al., 2010; Balaguer-Beser et al., 2013; Ruiz et al., 2011). Structure features are calculated in FETEX using the experimental semivariogram approach. According to Ruiz et al. (2011), the semivariogram quantifies the spatial associations of the values of a variable, measures the degree of spatial correlation between different pixels in an image and is a suitable tool for the characterization of regular patterns. FETEX 2.0 obtains the experimental semivariogram of each polygon by computing the mean of the semivariogram calculated in six different directions, from 0° to 150°, with step increments of 30°. Then, each semivariogram curve is smoothed using a Gaussian filter to reduce experimental fluctuations (Ruiz et al., 2011). Structural features extracted from the semivariogram are based on the zonal analysis defined by a set of singular points on the semivariogram, such as the first maximum, the first minimum, and the second maximum (Ruiz et al., 2011).[1]

Group	Variable name	Description
Spectral features	MEAN1	Mean of pixel values in band 1
	DEVST1	Standard deviation of pixel values in band 1
	MAJORITY1	Majority of pixel values in band 1
	MEAN2	Mean of pixel values in band 2
	DEVST2	Standard deviation of pixel values in band 2
	MAJORITY2	Majority of pixel values in band 2
	MEAN3	Mean of pixel values in band 3
	DEVST3	Standard deviation of pixel values in band 3
	MAJORITY3	Majority of pixel values in band 3
Texture features	MEAN_EDG	Mean of the edgeness factor
	DEVST_EDG	Standard deviation of the edgeness factor
	UNIFOR	GLCM uniformity
	ENTROP	GLCM entropy
	CONTRAS	GLCM contrast
	IDM	GLCM inverse difference moment
	COVAR	GLCM covariance
	VARIAN	GLCM variance
	CORRELAC	GLCM correlation
	SKEWNESS	Skewness value of the histogram
	KURTOSIS	Kurtosis value of the histogram
Structure features	RVF	Ratio variance at first lag
	RSF	Ratio between semivariance values at second and first lag
	FDO	First derivative near the origin
	SDT	Second derivative at third lag
	MFM	Mean of the semivariogram values up to the first maximum
	VFM	Variance of the semivariogram values up to the first maximum
	DMF	Difference between the mean of the semivariogram values up to the first maximum and the semivariance at first lag
	RMM	Ratio between the semivariance at first local maximum and the mean semivariogram values up to this maximum
	SDF	Second order difference between first lag and first maximum
	AFM	Area between the semivariogram value in the firs lag and the semivariogram function until the first maximum

Table 2. Image-derived variables.

2.1.2 The Dataset

Once the image features are extracted, the next step is to create the dataset. This process consist of selecting, for each city, a ground truth sample. Each polygon of the sample is manually labeled in one of two categories: slum, no-slum. Ancillary information and previous studies where used as reference to identify slum areas in each city for sampling. The location of slum areas in Buenos

[1] For a full description of these features, see Balaguer et al. (2010); Balaguer-Beser et al. (2013) and Ruiz et al. (2011).

Aires where identified from the web site “Caminos de la Villa” (www.caminosdelavilla.org) which shows an interactive map of the city and the location of recognized “villas” (slums), and we used the delineation of urban slums from Duque et al. (2013) and Duque et al. (2015) for Medellin; and from Da Silva and Shaw (2011) for Recife. Sampling of no-slum areas in each city included different formal urban layouts across each city, i.e., high and low rise residential areas, as well as parks, urban forests and green spaces, and commercial and industrial areas like malls, transport facilities and factories. When benchmark information of slum areas is not available to build the ground truth sample, practicioners must seek reference information with local authorities or use an experienced photo-interpreter who can visually pick up slum and no-slum areas to build it. Figure 4 shows the spatial distribution of the sample in each city.

The final step in this stage is to divide the dataset into two sets: the training set, 60% of the sampled polygons for training and tuning the classification models, and the testing set, 40% of the sampled polygons to evaluate the predictive capability of the classification models. Table 3 summarizes the composition of the datasets.

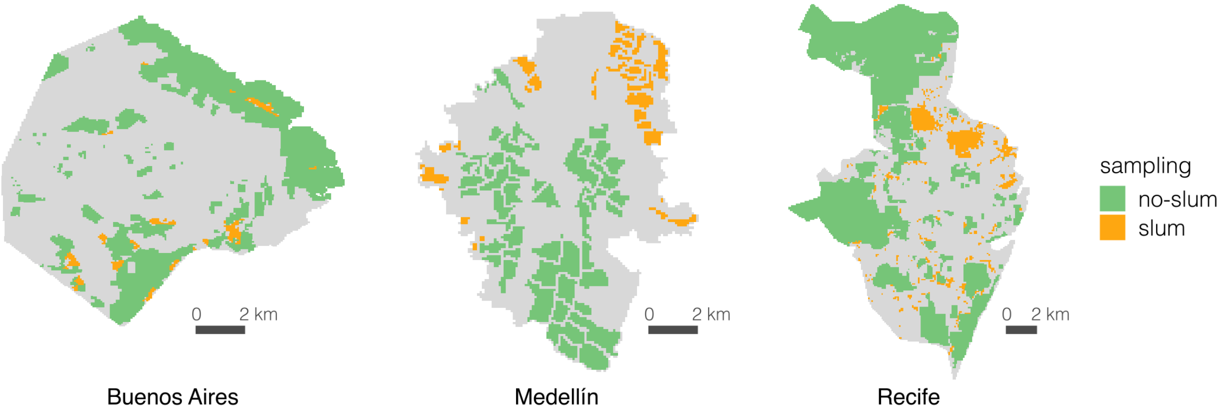


Figure 4. Sampling scheme of slum and no-slum areas in each city. From left to right: Buenos Aires, Medellin, Recife.

Table 3. Composition of the dataset

City	Total	Labeled	No Labeled	L. Slum	L. No Slum	Training	Testing
Buenos Aires	21,516	6,558	14,958	369	6,189	3,934	2,624
Medellin	10,255	2,891	7,364	602	2,289	1,734	1,157
Recife	22,037	11,218	10,819	1,274	9,944	6,730	4,488

Once the ground truth sampling was complete, we used these data and the Kolmogorov-Smirnov (KS) test (Marsaglia et al., 2003) implemented in the R package “kolmin” (Carvalho, 2015; R Core Team, 2013) to better understand the discriminating power of the image-derived variables for the differentiation of slum areas from no-slum areas within each city. The results of the Kolmogorov-Smirnov test indicate that the distributions of all image-derived variables are statistically different for the slum areas compared to the no-slum areas. Figure 5 shows the boxplots of the top 5 most discriminant image-extracted variables for each city.[2] It is very interesting that two of the top 5 most discriminating variables are present in all 3 cities (SDF and CONTRAS), and that the top 5 are the same for Buenos Aires and Medellin (SFD, CONTRAS, IDM, MEAN_EDG and FDO).

These variables are from the texture and structure groups, with the exception of MEAN1, wich belongs to the spectral group and informs about the mean of the intensity values in band

[2] The results for the other variables are available upon authors request.

1, which corresponds to the red channel. SDF is a structure variable that informs homogeneity at short distances (Balaguer et al., 2010; Balaguer-Beser et al., 2013) and slum areas show lower homogeneity than no-slum areas because is often common the presence of different small dwelling units with different roof colors in close proximity to each other (Duque et al., 2015). CONTRAS is a texture variable that informs about the differences in color and intensity of the objects present in the image (Haralick et al., 1973), and in these cities the slum areas show higher values than no-slum areas. MEAN_EDG is an aggregated measure of the density of edges present in the image (Ruiz et al., 2011), and slum areas in these cities tend to show higher values than no-slum areas because the smaller size of dwelling units, narrower roads, and the presence of shadows between housing units and their surroundings. IDM is a texture measure that informs about the general homogeneity (Haralick et al., 1973; Ruiz et al., 2011), and slum areas are characterized by lower values of this feature than no-slum areas (Duque et al., 2015). FDO is a structure feature that informs about the variability changes at short distances (Balaguer et al., 2010), and slum areas show higher values than no-slum areas because pixel values can change abruptly at short distances. AFM and DMF are structure features that are related also to the variability of the pixel values in the image (Balaguer et al., 2010), and Recife slum areas tend to show higher values of both features than no-slum areas, meaning that the slum areas often display more variability and less homogeneity than no-slum areas.

2.2 Classification Model

Classification literature is broad and multiple methods and algorithms have been proposed in the last decades (Hastie et al., 2009). In general, the main goal is to develop a quantitative classification method capable of learning and generalize the relationship between a set of variables (X) and a categorical variable (Y). For our particular classification problem X is a matrix with the spectral, texture, and structure values of each polygon in the grid, and Y a categorical variable defined as 1 or -1 if a polygon is a slum or no-slum respectively. The capability of the classification method is ruled by two factors: i) The theoretical definition of the classification boundary of the classifier (e.g., linear, non-linear), and ii) the complexity of the data.

Based on the classification boundary, the classifiers are commonly divided between linear and non-linear. Linear classifiers, such as logistic regression and linear SVMs, assume that the categorical variable (Y) can be obtained by exploiting a linear combination of the input features (X). Non-linear classifiers generalize the boundary by adjusting polynomial boundaries, Gaussian kernels, or algorithmic criteria based on feature thresholding. Figure 6 illustrates a linear and non-linear decision boundary. Non-linear classifiers are able to capture more complex patterns from the data, but as consequence, they are computationally more complex than their linear counterpart, and more prune to memorize the training data (overfitting).

Regarding the data, its intrinsic complexity can not be easily obtained or described, in particular for high dimensional datasets. The most intuitive way to understand the data complexity is by visualizing its features and the respective classes. This approach is usually restricted to low dimensional data (2D or 3D), or to simplified versions of the feature space obtained by manifold algorithms like Principal Components Analysis (PCA), IsoMaps, Self Organizing Maps, among others (Betancourt et al., 2016). A common approach when dealing with hihgly dimensional data is to infer its complexity by comparing the capability of different classification algorithms to capture know patters. Roughly summarized, a simple classifier (linear) will perform poorly on complex data (non-linear), while complex classifiers (non-linear) are able to fit more complex data in exchange of a large risk of overfitting. This is known as the bias-variance tradeoff Geman et al. (1992). Fortunately, as will present bellow, machine learning literature contains different mechanisms to face the

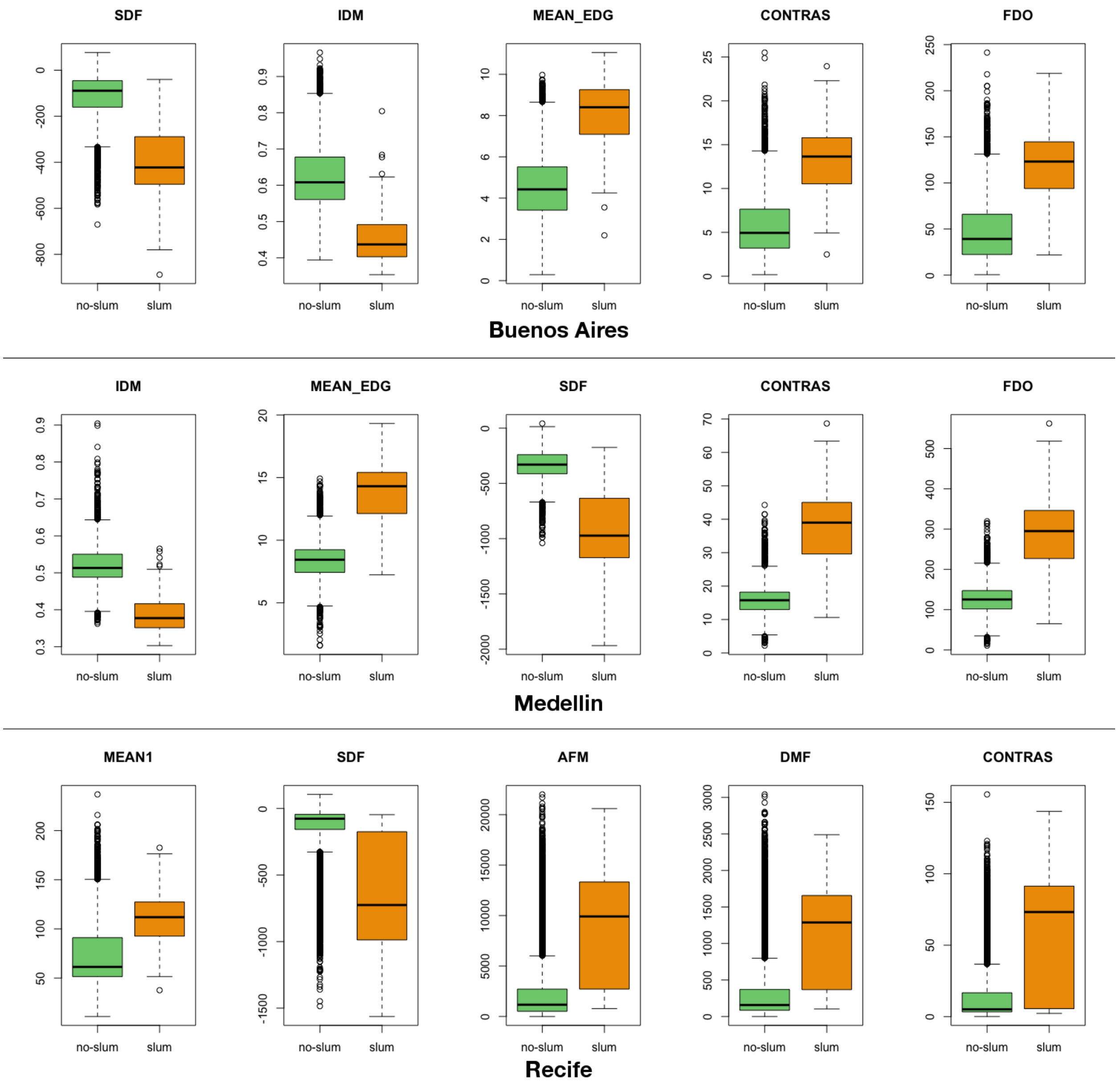
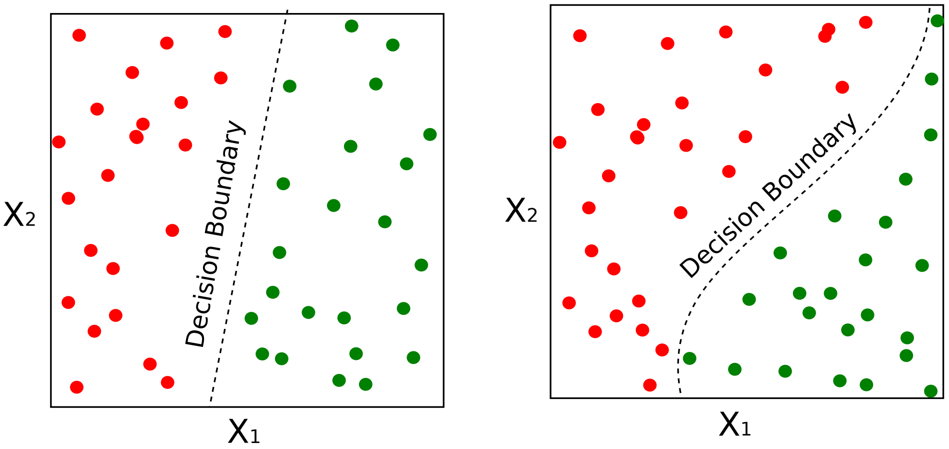


Figure 5. Boxplots of the distributions of the top 5 most discriminant image-derived variables in each city. No-slum distributions in yellow and slum distributions in red. Variables organized from left to right with higher values of the Kolmogorov-Smirnov test at left and lower values at right.

bias-variance dilemma. Regarding the size of the training sets, there is not a direct answer about the number of obserbations required to train the models and is commonly pointed as a consequence of the complexity of the problem to solve. Recent advances in data-science and deep-learning frequently point to the benefits large datasets; however, when data collection is expensive and time consuming a common practice is to observe the changes in the evaluation criteria while sequentially increasing the number of observations used to train the models. If the evaluation criteria does not improve (converge) while the number of training samples increase, then it is valid to reject the necessity of collect more training data.

Since our data has high dimensionality (30 features extracted per polygon), with unknown dis-



(a) Linear Boundary on separable data (b) Non-Linear Boundary on separable data

Figure 6. Linear and no-linear classification boundary in 2D

tributions, and comes from three different cities, we explored two approaches for training our model for slum identification: i) train a unique classifier on an unified dataset (i.e., without differentiating the cities) and then evaluate if the resulting slums are reliable, and ii) follow a multi-model approach by training the classifier in each city. We explore the performance of linear (Logistic Regression, linear SVM) and non-linear classifiers (Polynomial and Radial Basis Kernel SVMs, and Random Forests).[3] Next, we will provide a brief description of these classifiers.

Logistic Regression: It is the most common linear clasifier and commonly used by policy makers in econometric literature. It can be considered a mathematical approach, with the main goal of exploit the binary logistic model, represented in equation (1), to estimate the probability of a categorical value Y given the input features X. For practical purposes equation (1) can be solved for $X\beta$ to obtain (2), and the value of β estimated by optimizing the error function (J), which in our case is defined by (3) which is the Least Square Error plus a regularization term. Where C is a regularization constant to avoid extremely large or small values of the coefficients β, and controls the overfitting/underfiting problem. Large values of C will encourage wiggly boundaries by giving more importance to the least square term in equation (3). In our formulation, the regularization constant is applied to the LSE term to keep it consistent with the formulation of the SVM, and it is equivalent to apply $1/C$ to the regularization term. The selection of the regularization constant is explained at the end of this chapter.

$$Y = \frac{1}{1+e^{-XB}} \tag{1}$$

$$\hat{Y} = X\beta = log\left(\frac{Y}{1-Y}\right) \tag{2}$$

$$J = C(Y-X\beta)^2 + \frac{\beta^T\beta}{2} \tag{3}$$

[3] These classifiers are available in the Python library Scikit-learn by Pedregosa et al. (2011).

Regression analysis including the error function proposed in Equation 3 is commonly known in statistical analysis as Ridge regression and is frequently used to avoid large β values. Despite of being using a method frequently applied in econometrics, our interest is the capability of the trained model to classify new observations and not in the role of the exogenous variables to achieve it, which allow us to relax the assumption of multicolinearity (Cessie and Houwelingen, 1992). This correlation, from an econometric perspective, will considerably affect the β values and their p-values; however, the ontained model will perform the same in the classification tasks. Other types of regularization terms like Lazzo (also known as L1) can be applied to the proposed algorithm, but machine learning literature frequently point to Ridge regularization (also known as L2) as the most efective approach (Tibshirani, 1996; Ng, 2004). This can be illustrated by assumimg correlated variables in the dataset.

Support Vector Machine: It is a popular non-probabilistic classification algorithm, commonly known by its capability to maximize the margin between the decision boundary and the data. SVM, as the logistic regression, relies on a mathematical formulation to express the classification task as an optimization problem. It is highly popular due to its capability to easily include non-linear boundaries (kernels) from the theoretical formulation, and its explicit goal of locating the boundaries as far as possible of the training data. To introduce the SVM formulation lets bring back the training data X and their categorical labels Y to construct n pairs of training samples: (X_1, y_1), (X_2, y_2), $\dots, (X_n, y_n)$, with $X_i \in R^p$ and $y_i \in \{-1, 1\}$. Equation (4) defines a linear classification hyperplane and equation (5) its induced classification rule, where β is a unit vector and $h(x)$ are a family of basis functions to enlarge the input space (e.g., Linear, Polynomial, Radial Basis, Splines, etc.). From this formulation it is possible to understand SVM as a strategy to map the data from an input space to a different space defined by $h(x)$ (on which the differences in the input data are exposed), and then define the classification rules by using $G(F)$, which measures the side of the boundary on which the observations are located. The advantage of SVM resides then on finding a well defined $h(x)$ and the parameters B.

$$\{F : f(X) = h(X)^T\beta + \beta_0 = 0\} \tag{4}$$

$$G(F) = sign(f(X)) = sign(h(X)^T\beta + \beta_0) \tag{5}$$

$$\tag{6}$$

Assuming that the classes are not separable (i.e., some observations are non separable by a predefined boundary shape) then the values of β and β_0 are the solution of the optimization problem given by (7), where $\xi = (\xi_1, \xi_2, \dots, \xi_N)$ are referred as the slack variables which account the distance of the missclassified obserbations to the decision boundary, and C is a regularization constant having the same effect of the one defined for the linear regresssion. Note that conceptually the SVM procedure looks to minimize the distance of the non-separable obserbations to the boundary, plus the magnitude of the coefficients β.

$$\min_{\beta,\beta_0} ||\beta||^2 + C\sum_{i=1}^{N}\xi_i \tag{7}$$

$$\text{subject to: } y_i(h(X_i)^T\beta + \beta_0) \geq 1 - \xi_i,\ \forall i, \tag{8}$$

$$\xi_i \geq 0 \tag{9}$$

Interestingly, the SVM formulation constitutes a quadratic problem with linear constraints that can be solved via Lagrange Multipliers to obtain the Lagrange Dual function (10) and the solution of the problem (7) as (11), where α_i are the lagrange multipliers of the constraint given by equation (8). The solution of the problem only involves the basis function as inner products, which constitutes one of the most popular characteristic of the SVM formulation, allowing to solve the problem without define the basis function but its inner product (kernel). Note that, the decision function of a new obserbation x is obtained from equation 11 by using the information of the N missclassified points, namely their lagrange mutipliers α_i, their original categorical variable y_i, and their inner product with respect to the new obserbation $\langle h(x), h(x_i)\rangle$.

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i'=1}^{N} \alpha_i \alpha_{i'} y_i y_{i'} \langle h(x_i), h(x_{i'})\rangle \tag{10}$$

$$f(X) = \sum_{i=1}^{N} \alpha_i y_i \langle h(x), h(x_i)\rangle + \beta_0 \tag{11}$$

In the experimental section we use the $d^t h$ polynomial kernel (1 to 5 order) and the radial basis kernel defined by equation (12) and (13). To gain intuition behind the kernel advantages, the larger the grade of the polynomial the more complex curves in the classification boundary. On the other hand, the RBF kernel allows the boundary to create other shapes like circles or ellipses if the data requires it. For the SVM case, the concept of overfiting is quite relevant in particular for complex kernels, highlighting the importance of a proper tunning to the regularization constant. We use the subindices of the SVM as reference to the kernel used, for example SVM_k uses the k^{th} polynomial kernel and SVM_{rbf} the radial basis kernel. In seek of simplicity the description of the SVM procedure presented in this paper is hihgly summarized. See (Hastie et al., 2009) for a complete overview of the optimization procedure and more details about the kernel functions.

$$K_{dpol}(x, x') = \langle h(x), h(x)'\rangle = (1 + \langle x, x'\rangle)^2 \tag{12}$$

$$K_{rbf}(x, x') = \langle h(x), h(x)'\rangle = e^{\gamma ||x-x'||^2} \tag{13}$$

Random Forest: Contrary to the Linear Regression and the SVM, the Random Forest takes its decision by a sequential set of thresholding rules on the input space and its procedure is closer to an algorithmic strategy rather than an optimization method. Theoretically, a Random Forest (RF) is an ensamble method formed by multiple decision trees, each of wich is trained by using a different part of the training data. A Decision Tree, respectively, is an algorithmic strategy to divide in feature space to fit the output variable (MathSoft, 1999).

Lets assume the already introduced data pairs (X, y), and let define the number of variables in the input space as p. It is possible to define an arbitrary partition of the input space X^p into M regions as $R_1, R_2, \ldots, R_M$, and a response constant c_m per region in the partition. Given a partition, it is possible to find the best values of c_m by defining an error function $J(c_m)$ and evaluating it for all the observations withing each region, see (14). The role of the response constant is to capture the output variable of the observations inside each region R_m and can be defined, for example, as the dominant label in the region (for a classification problem) or as the average value (for a regression problem based on the squared error). On its more general way, the goal of the decision tree is to reach a good partition of the input space and as thus define the response constants.

$$\begin{aligned}
\hat{c}_m &= \operatorname*{argmin}_{cm} J(c_m) && \forall m \in 1, \ldots, M \\
&= \operatorname*{argmin}_{cm} J(f(y_i, x_i, c_m) | (x_i, y_i) \in R_m) && \forall m \in 1, \ldots, M \qquad (14)
\end{aligned}$$

Starting with all the training observations consider a splitting feature j and value s to divide the feature space in two half-planes (R_1 and R_2) defined in (15). The best pair of (j, s) is the one that minimize the overall error of R_1 and R_2 as shown in (16). The same splitting procedure can be applied recursively on each half-plane until a stop criteria reached. In our case we use the maximum depth as stop criteria. The obtained tree can be applied to process new observations by following the rules captured by the sequences of (j, s) and returning the response constant assigned to the output region. The final decision of the Random Forest is the average of the decisions of its Decision Trees. In our case the error function is the Least Squared Error, the maximum depth is set to 10, and each each random forest contains 10 decision trees.

$$R_1(j, s) = \{X | X_j \leq s\} \qquad \text{and} \qquad R_2(j, s) = \{X | X_j > s\} \qquad (15)$$

$$\begin{aligned}
min_{j,s}\left[J(\hat{c}_1) + J(\hat{c}_2)\right] \quad = min_{j,s}[\quad & J(f(y_i, x_i, \hat{c}_1) | (x_i, y_i) \in R_1(j, s)) + \qquad (16) \\
& J(f(y_i, x_i, \hat{c}_2) | (x_i, y_i) \in R_2(j, s))]
\end{aligned}$$

Our comparison of the classifiers is based on the β score (F_β), which is a numeric performance defined by equation (17), where the precision and recall are defined by equation (18) and (19) respectively. In general the precision measures how reliable are the slums detected (how pure are the regions detected as slum) and the recall measures how good the classifier is to retrieve the areas defined as slum (how many of the real slums are detected). The F_β score as well as the precision and the recall are bounded between 0 and 1, being 1 a perfect classifier. The value of β must be selected according to the problem to solve and is usually set to 0.5, 1 or 2. A value of $\beta = 0.5$ gives a larger weight to the precision while a value of $\beta = 2$ prioritize the recall. In the remain part of this paper β is defined as 2 (i.e., $F_{\beta=2}$) to give a larger importance to the recall. This implies that, when classifying areas as slum or no-slum, we prefer type I errors over type II errors to prevent that vulnerable population is left out of consideration.

$$F_\beta \quad = \quad (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}} \qquad (17)$$

$$\begin{aligned}
precision &= \frac{TruePositives}{TruePositives + FalsePositives} \qquad (18) \\
recall &= \frac{TruePositives}{TruePositives + FalseNegatives} \qquad (19)
\end{aligned}$$

Once defined the best performing approach (unified or multi-model) and the best classifier, the next step is to tune the regularization constant to avoid overfiting the data and fine-tune the decision threshold to obtain the final F_2 scores. The regularization constant is tuned exhaustively by evaluating the F_2 score obtained while changing the regularization constant. The regularization constant leading to the higher F_2 score is then defined as the final choice. The decision threshold is the value on which the classifier decides weather a particular observation is accepted as slum or not. For the logistic regression it can be understood as a final horizontal shift of the logistic curve, and for the SVMs as a final tuning of β_0. The decision threshold is selected by using the Receiver Operating Characteristic (ROC) curve, which is a visualization of the False-Positives (X-axis) and True-Positives (Y-axis) while changing the decision thresholds. The machine learning bibliography suggest to define the threshold as the one that generates the closest point to the upper-left corner in the ROC curve. It is important to note that the decision thresholds of the logistic regression reported in section 3 are not bounded between 0 and 1, which is equivalent to use the X axis for the final decision.

To keep the tunning process fair (regularization constant, decision threshold), it is necesary to use only observations in the training dataset. This is accomplished by using cross-validation $F2$-scores. To obtain the cross-validation $F2$-scores scores, the first step is to divide the training dataset into k equal sized parts. On a single iteration, a classifier (with a particular regularization constant and decision threshold) is trained on $k-1$ parts, and tested in the remaining one to keep the $F2$-score. This process is repeated k times ensuring that each part is used once for testing. The final cross-validation $F2$-score is the average of the obtained $F2$-score on each iteration. Our parameter selection is based on $10-fold$ cross-validation.

2.3 Slum Changes in Time

As stated above, we downloaded historical GE images for selected sectors in each city from about a decade ago to perform change analysis ($t-1$ period). We extracted image features using the same regular grid of square cells and we used the classifier model trained with the 2016 image-extracted data (t period) to classify the corresponding areas into slum or no-slum classes. Then we compared the results of the two dates (t vs $t-1$) in a cell by cell basis and assigned different colors to differentiate the areas that were classified as slum in both dates, those that were classified as no-slum in both dates, those that were classified as no-slum in $t-1$ date and were classified as slum in t date, and those that were classified as slum in $t-1$ date and no-slum in t date.

Following this rationale, we tested if the proposed approach could be useful to analyze slum dynamics over time by detecting areas that have become slum, stable areas (no change), and areas that were slum and have become no-slum by upgrading or urban renovation processes.

3 Results and discussion

As the first step in our experimental analysis we use the data of the three cities to build a unified model. Table 4 shows the F_2 score for each type of classifier in the testing set of each city. It is evident from the table that SVM_{rbf} stands out as the best performing model. Regarding the polynomial SVMs (SMV_2,...,SMV_5) there were some signals of underfiting particularly in the higher order models. The linear models (Logistic Regression and SMV_1) reach a good classification score and does not show signal of overfiting/underfiting. However, the Gaussian kernel of the SMV_{rbf} is able to disambiguate difficult cases. Finally, it is noteworthy the poor performance of the Random Forest. The low classification scores of some algorithms suggest the existence of singularities within cities that complicates the identification of slums using an unified model.

Table 4. Unified Model for Slum Detection (F_2-Scores).

City	Log. R	SVM_1	SVM_2	SVM_3	SVM_4	SVM_5	SVM_{rbf}	RF
Buenos Aires	0.649	0.581	0.472	0.419	0.226	0.134	**0.671**	0.277
Medellin	0.757	0.516	0.794	0.817	0.821	0.841	**0.872**	0.550
Recife	0.671	0.552	0.559	0.565	0.564	0.592	**0.803**	0.516

Motivated by the differences in the urban structures of the cities we train a classification model for each city. Table 5 shows the testing F_2 score of each model and city. In this case the best classification score is obtained by the Logistic Regression and the SVM_{rbf}, both achieving F_2 improvements between 2 and 5 points with respect to the unified model. The rest of the models show some improvements against their unified counterpart but the performance is still poor in comparison with the Logistic Model and the SVM_{rbf}. These results confirm the intuition of structural differences in the features of the slums for each city that preclude the implementation of an unified model.

Table 5. Individual Model for Slum Detection (F_2-Scores).

City	Log. R	SVM_1	SVM_2	SVM_3	SVM_4	SVM_5	SVM_{rbf}	RF
Buenos Aires	**0.737**	0.642	0.649	0.678	0.652	0.642	**0.688**	0.596
Medellin	**0.928**	0.886	0.783	0.839	0.809	0.832	**0.909**	0.687
Recife	**0.775**	0.734	0.702	0.644	0.585	0.576	**0.821**	0.532

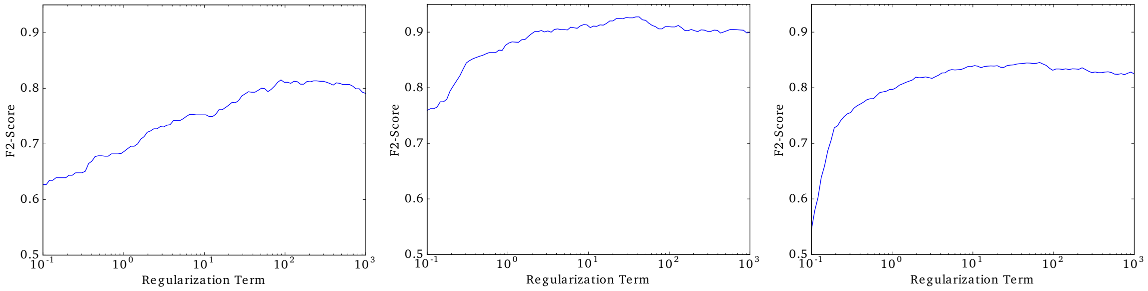
The next step is to remove signs of overfiting/underfiting of the best performing models and tune the decision threshold (th). This part of the analysis only includes the Logistic Regression and the SMV_{rbf}. As explained in section 2.2 the regularization term is selected by an exhaustive incremental search and the best threshold by using the ROC curve. Table 6 shows the F_2 scores of the default configuration (default), using only the tuner regularization term (Reg.), and using the regularization term and the best threshold (Reg + th). The table confirms the benefit of the final tunning and allows us to conclude that the best strategy is to use single model per city and take into account the regularization parameter and tune the decision threshold.

Table 6. Tuning the best classification models (F_2-Scores).

	Log. Regression			SVM_{rfb}		
City	Default	Reg.	Reg + th	Default	Reg.	Reg + th
Buenos Aires	0.737	0.759	0.715	0.688	0.813	**0.817**
Medellin	0.928	0.949	0.957	0.909	0.937	**0.976**
Recife	0.775	0.767	0.827	0.821	0.87	**0.872**

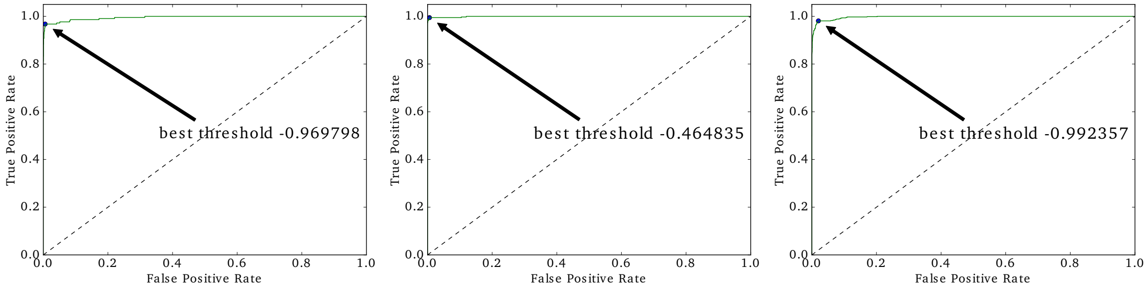
Figure 7 shows the F_2 score of each individual SVM_{rbf} while changing the regularization term. The regularization value that maximizes the F_2 score is set as the regularization term of the model. Figure 8 shows the ROC curves given the best regularization term. As explained before the decision threshold is set as the one that generates the closest point to the Upper-Left corner of the curve. Finally, Table 7 show the parameters selected for Table 6.

Figure 9 shows the maps of detected slum areas from the classification process of the 2016 GE images in each city. Most of the variables used for input for the machine learning algorithm (i.e., spectral, texture and structural features) are very useful for the quantitative characterization of the spatial pattern of the elements in an image (Ruiz et al., 2011), and they are also useful to quantify the heterogeneity of the image elements in each polygon or 100 by 100 meters cell (Duque et al., 2015). Thus, these sets of image-derived features perform better for slum detection in urban



(a) Buenos Aires (b) Medellin (c) Recife

Figure 7. Tuning the regularization terms.



(a) Buenos Aires (b) Medellin (c) Recife

Figure 8. ROC curves to select the best decision threshold. For visualization purposes the x axis is reported in logarithmic scale. The final regularization term is reported in Table 7

Table 7. Final parameters of the logistic regression and the SVM_{rbf}. The values of the decision threshold for the logistic regression are reported in the x axis, and are not bounded between 0 and 1. For both cases (Logistic Regression and SVM), the most negative the threshold the more prone the classifier to set an obserbation as slum.

	Log. Regression		SVM_{rfb}	
City	Reg. Term	threshold	Reg. Term	threshold
Buenos Aires	12.618	-3.111	394.42	-0.969
Medellin	20.092	-1.494	16.681	-0.464
Recife	225.70	-1.788	18.307	-0.992

settings where the slum areas have a different spatial pattern than no-slum areas and where those slum areas exhibit a high diversity of building and roofing materials. As showed in Table 6 the approach performs better for Medellin (F_2 score 0.976), then for Recife (0.872), and then for Buenos Aires (0.817). Even though the F_2 scores are high, there are some overestimation of slum areas in all three cities. However, these results are still encouraging for a rapid approach to slum detection, given the simplified input data, the speed of the method and the ease to automate the process.

The image-extracted variables used to separate slum areas from no-slum areas in all three cities worked best for Medellin and Recife than for Buenos Aires. The confusion matrix of the classification results show the magnitude of the overestimation (no-slum areas classified as slum) and underestimation (slum areas classified as no-slum) for the testing dataset in each city. Buenos Aires shows less than 3% of overestimation and about 1% of underestimation; Medellin is the best case, with 2% of overestimation and no underestimation; and Recife shows 4.5 % of overestimation and less than 1% of underestimation. Figure 10 shows known slum sectors in each city at the same

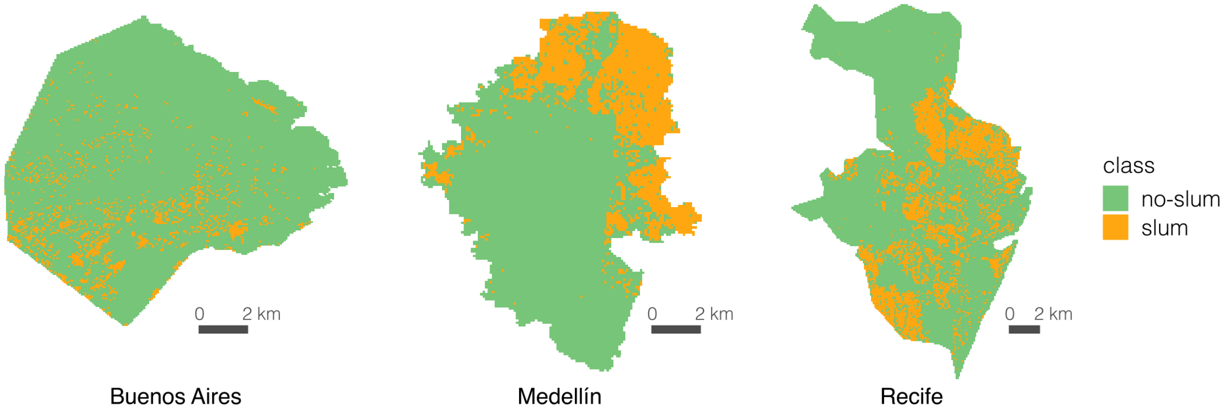


Figure 9. Classification results for 2016 GE images of each city to slum and no-slum areas. From left to right: Buenos Aires, Medellin, Recife.

Table 8. Confusion Matrix of the SVM_{rbf} with the parameters reported in Table 7

(a) Buenos Aires

	Predicted No-Slum	Predicted Slum
No-Slum	2387	84
Slum	17	136

(b) Medellín

	Predicted No-Slum	Predicted Slum
No-Slum	903	27
Slum	0	227

(c) Recife

	Predicted No-Slum	Predicted Slum
No-Slum	3800	203
Slum	32	453

spatial scale: Villa Zavaleta (21-14) in Buenos Aires, Comuna Santa Cruz in Medellin, and Chao de Estrelas in Recife. Table 9 shows a general characterization of slum areas in each city in terms of the image-derived features to help the understanding of the classification results. The slums in all three cities are composed of clusters of small dwelling units and very few vegetated areas. However, as was expected from the boxplots of figure 5, the appareance of slums in Buenos Aires and Medellin have more similarities when compared to the appareance of the slums in Recife. The slum areas in Buenos Aires and Medellin are characterized by high heterogeneity at short distances and high homogeneity at large distances (different things in close proximity, but the same pattern at larger scale), and similar organic patterns but the slum in Buenos Aires is more cluttered than in Medellin. The slum in Recife shows more homogeneity in color with most roofs showing clay tiles or similar. This explains the high discriminating power of the variable MEAN1 in this city, as the band 1 records the intensity values of the red channel in the visible spectrum and slum areas have many pixels with the same red tone in this city. Also, the slums in Recife show more regularity in the spatial pattern of the urban layout than in the slums of the other two cities.

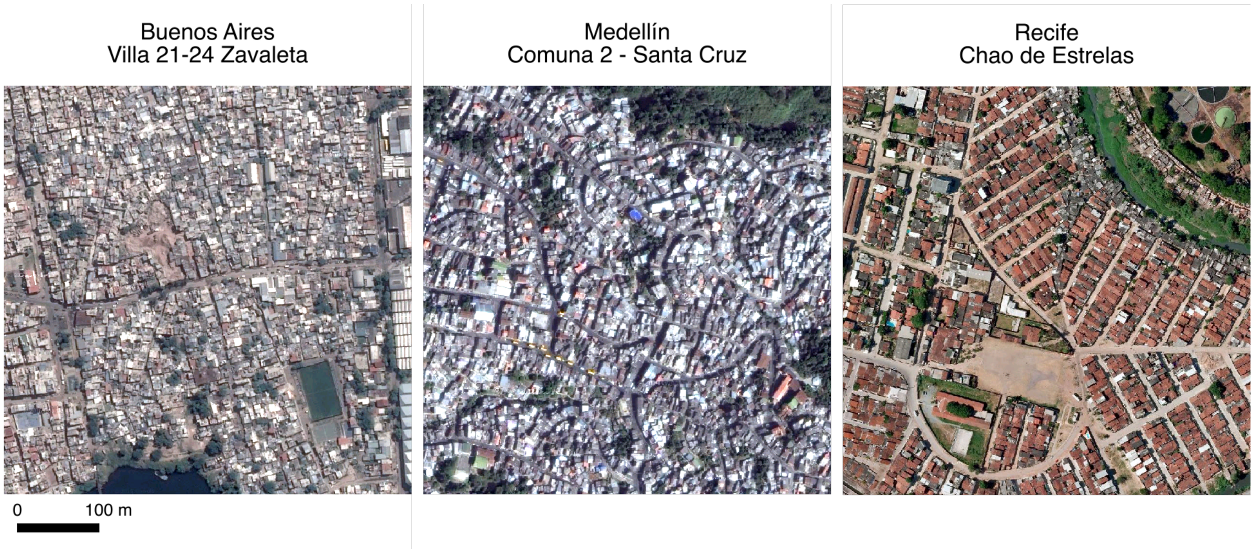


Figure 10. Sectors of known slum areas in each city. From left to right: Buenos Aires, Medellin and Recife.

Table 9. General characteristics of slums in the analyzed cities.

Observable feature	Buenos Aires	Medellin	Recife
Short distance heterogeneity	high	high	medium to high
Large distance homogeneity	very high	very high	medium to high
Roofing material diversity	high	high	medium to high
Vegetation coverage	very low	low	low
Urban layout pattern	organic	very organic	regular
Crowdedness	very high	high	medium to high
Dwelling size	very small	very small	small
Roads material	not paved	paved	not paved
Roads width	very narrow	narrow	standard

The lower score obtained for Buenos Aires could be explained by the quality of the input GE image and the fact that some of the no-slum areas of the city show some characteristics that are also present in slum areas. The Buenos Aires GE image shows low contrast (differences in color intensity tend to be low across the image), which could lower the quantifiable differences between the slum and no-slum areas. Figure 11 shows the Zavaleta villa next to a no-slum area. Both areas show very few vegetated areas between buildings and high heterogeneity at short distances, but the no-slum sector shows more regularity in the spatial pattern of the urban layout and large homogeneous surfaces interleaved with clusters of smaller buildings.

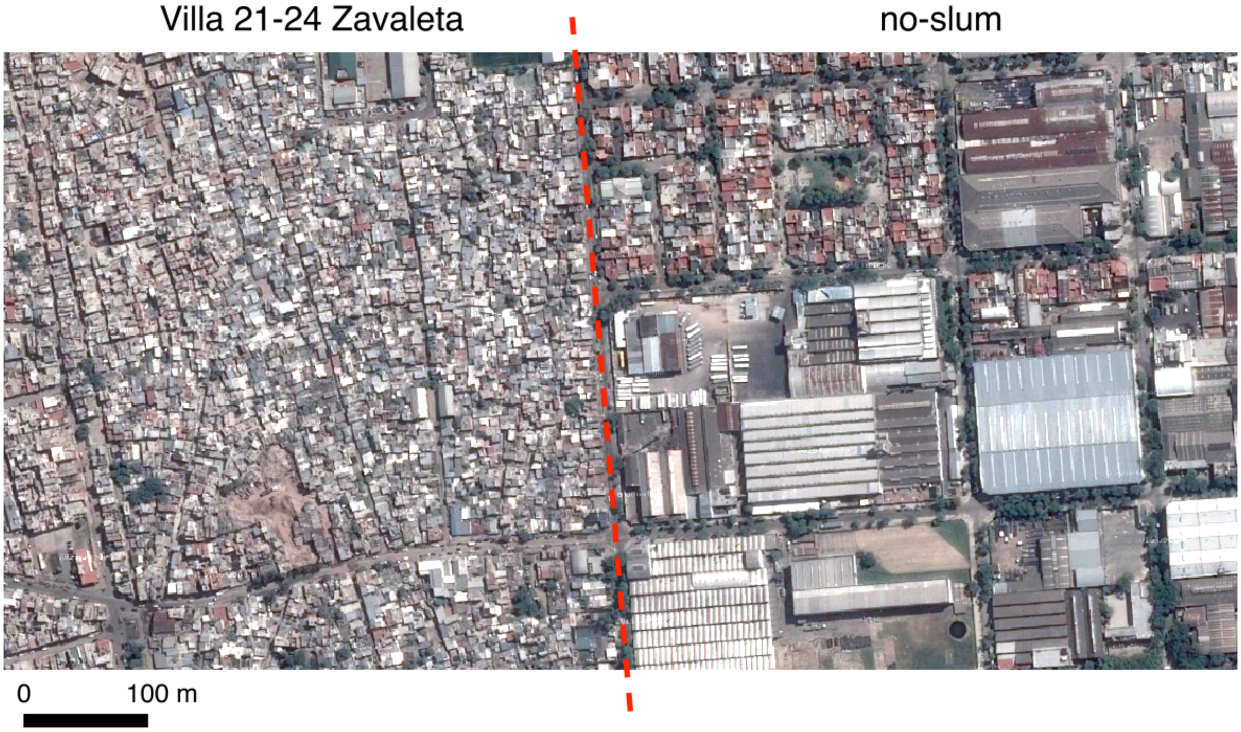


Figure 11. Slum sector in Buenos Aires (Villa Zavaleta) compared to the no-slum area next to it. The red line shows the slum boundary as mapped in www.caminosdelavilla.org.

Figure 12 shows the results of the temporal analysis of the selected sectors of 1 squared kilometer in each city. The approach seems to be useful to inform from a global perspective of the areas that have changed from no-slum to slum and vice versa between the analyzed dates. However, as in the implemented algorithm the recall was given priority over the precision for a good identification of the more problematic regions within the city, we expected to have some false positives in the classification results. These false positives difficult the interpretation at a detailed scale or in a cell by cell (of the regular grid) basis, and they can mask real changes that an interpreter could pick up visually by comparing the two GE images of each sector.

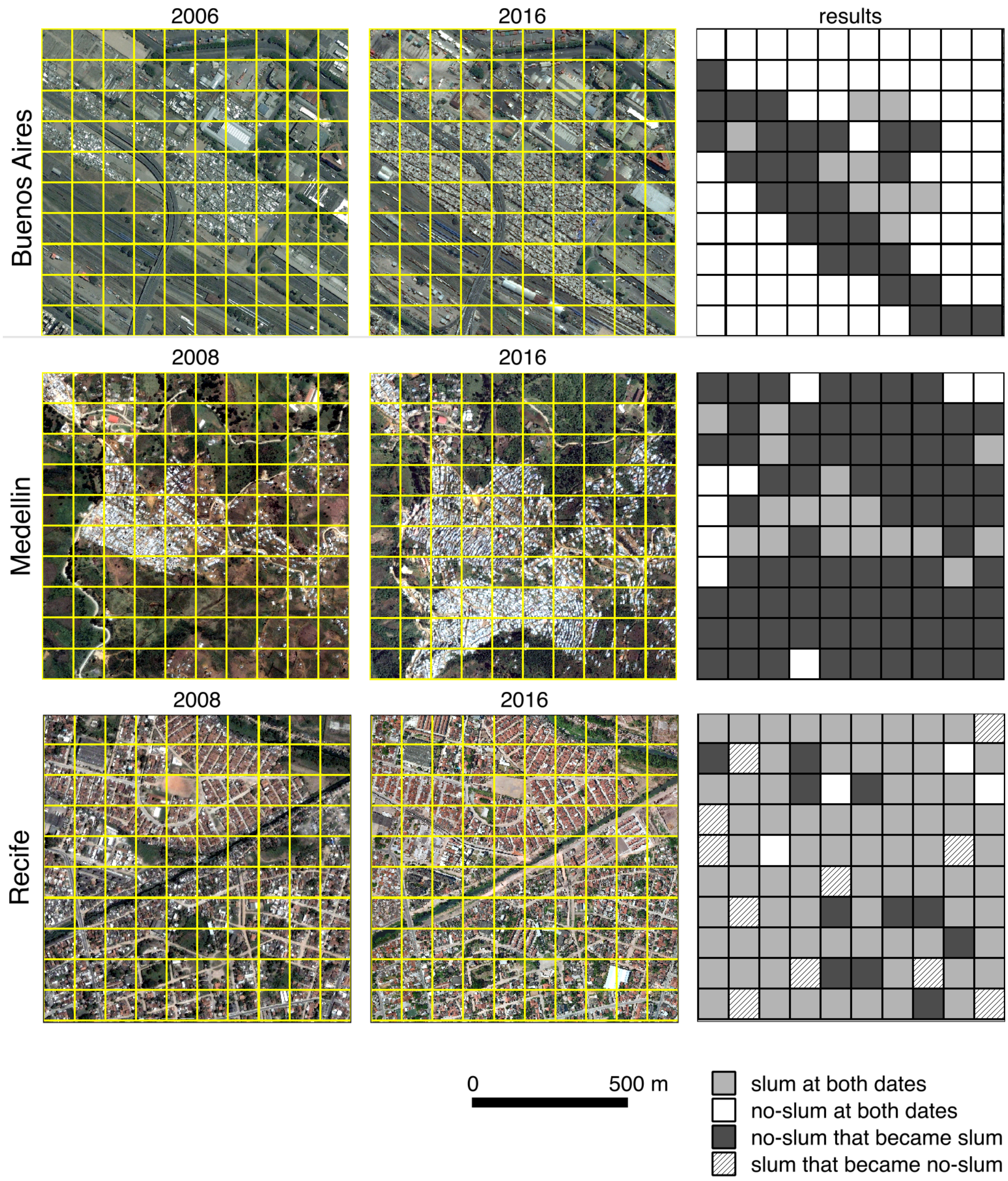


Figure 12. Classification results for historical GE images of selected sectors in each city.

The proposed approach works better to identify recently informally occupied urban areas with slum characteristics than to pick up changes due to slum upgrading processes. Upgrading of slum areas often involves the improvement of dwelling units and the offer of public services (UN-Habitat, 2016), and less often with modification of the urban layout as that implies relocation of population and many slum residents fear that redevelopment will leave them homeless (Stellmacher, 2011). In this regard, upgrading processes that do not significantly change the spatial pattern and texture of the urban areas cannot be picked up by this approach, as most of the used image-derived features quantify aspects of the urban scene related to the spatial pattern and texture of the urban layout.

This workflow worked well for slum detection on a single date, but it has drawbacks for spatio-temporal analysis. Even though the historical images were resampled to match the pixel size of the updated reference images, and they were radiometrically normalized to match the color intensity; there are still differences in view angles, lighting, and vegetation phenology cycles between images that can affect vegetation appearance and shadow extent, and hence the values of the image-derived features and the classification results. To minimize the differences in view angles and vegetation phenology the practicioner must use historical images of the same day of the year as the reference or most updated images; but this is nearly imposible to control using the available data in Google Earth. Commercial satellite VHR imagery are more fit to this purpose, as they that can be acquired for specific dates to match the day of the year and minimize those differences.

4 Conclusions

This paper explored the possibility of implementing a low-cost standardized method for slum detection using spectral, texture and structural features extracted from VHR GE imagery as input data and assessed the capability of three ML algorithms to classify urban areas as slum or no-slum. Using data from Buenos Aires (Argentina), Medellin (Colombia), and Recife (Brazil) we found that Support Vector Machine with radial basis kernel (SVM_{rbk}) delivers the best performance with a F_2 score over 0.81.

We also found that the particularities of each city are important to take into account and preclude the use of a unified classification model. The ML algorithms worked best for Medellin and Recife, with F_2 scores of 0.98 and 0.87 respectively. The image-derived features performed better for slum detection in these cities as their slum areas have a different spatial pattern and texture than no-slum areas and they exhibit a high diversity of building and roofing materials.

The proposed workflow need more sophistication to properly track changes in time. As in the implemented ML algorithms the recall was given priority over the precision for a good identification of the more problematic regions within the city, there are some false positives in the classification results that difficult the change analysis between different dates. However, the proposed approach did work to identify recently informally occupied urban areas with slum characteristics, where the changes in local heterogeneity and the spatial pattern are clearly seen and different from those of occupied formal areas. Changes of the slum status of an area due to upgrading processes would be still difficult to pick up, since those processes do not significantly change the spatial pattern and texture of the urban areas, which are the aspects quantified by the image-derived variables.

Another future line of research will be to use algorithms for object and scene recognition on images from Google Street View to generate a new set of features that can improve the performance of our classification models. Street view and satellital imagery for slum identification can also be an important tool for supporting programs such as the Trust Fund for the Improvement of Family Housing lead by the Development Bank of Latin America, and the Foundation in favor of Social Housing.

Acknowledgments: The authors acknowledge the financial support from CAF - Development

Bank of Latin America. The usual disclaimer applies.

References

Balaguer, A., Ruiz, L. A., Hermosilla, T., and Recio, J. A. (2010). Definition of a comprehensive set of texture semivariogram features and their evaluation for object-oriented image classification. *Computers & Geosciences*, 36(2):231–240.

Balaguer-Beser, A., Ruiz, L. A., Hermosilla, T., and Recio, J. A. (2013). Using semivariogram indices to analyse heterogeneity in spatial patterns in remotely sensed images. *Computers and Geosciences*, 50:115–127.

Betancourt, A., Díaz-Rodríguez, N., Barakova, E., Marcenaro, L., Rauterberg, M., and Regazzoni, C. (2016). Unsupervised Understanding of Location and Illumination Changes in Egocentric Videos. *arXiv preprint*.

Cabrera-Barona, P., Wei, C., and Hagenlocher, M. (2016). Multiscale evaluation of an urban deprivation index: Implications for quality of life and healthcare accessibility planning. *Applied Geography*, 70:1–10.

Carvalho, L. (2015). Package 'kolmin'. Version 1.0. An Improved Evaluation of Kolmogorov's Distribution. CRAN Repository, 2015-06-15 17:38:00.

Cessie, S. L. and Houwelingen, J. V. (1992). Ridge estimators in logistic regression.

Da Silva, R. d. C. O. and Shaw, K. (2011). *Cartography of the Favela. Community resources to resist violencia in Recife and Olinda.* Shine a Light, Santa Fe and Florianópolis.

Duque, J. C., Patino, J. E., Ruiz, L. A., and Pardo-Pascual, J. E. (2015). Measuring intra-urban poverty using land cover and texture metrics derived from remote sensing data. *Landscape and Urban Planning*, 135:11–21.

Duque, J. C., Royuela, V., and Noreña, M. (2013). A stepwise procedure to determine a suitable scale for the spatial delineation of urban slums. In Fernandez, E. and Rubiera Morollón, F., editors, *Defining the spatial scale in modern regional analysis. Advances in Spatial Science*, chapter 12, pages 237–254. Springer-Verlag, Berlin Heidelberg.

Engstrom, R., Sandborn, A., Yu, Q., and Graesser, J. (2015). Assessing the relationship between spatial features derived from high resolution satellite imagery and census variables in Accra, Ghana. *IGARSS 2015*, pages 2544–2547.

Geman, S., Bienenstock, E., and Doursat, R. (1992). Neural Networks and the Bias/Variance Dilemma.

Haralick, R. M., Shanmugam, K., and Dinstein, I. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics,*, SMC-3(6):610–621.

Hastie, T., Tibshirani, R. J., and Friedman, J. (2009). *The Elements of Statistical Learning*, volume 1. Springer, 10 edition.

Hu, Q., Wu, W., Xia, T., Yu, Q., Yang, P., Li, Z., and Song, Q. (2013). Exploring the use of Google Earth imagery and object-based methods in land use/cover mapping. *Remote Sensing*, 5(11):6026–6042.

Jain, S. (2008). Remote sensing application for property tax evaluation. *International Journal of Applied Earth Observation and Geoinformation*, 10(1):109–121.

Kohli, D., Sliuzas, R., and Stein, A. (2016). Urban slum detection using texture and spatial metrics derived from satellite imagery. *Journal of Spatial Science*, 8596(May):1–22.

Kuffer, M. and Barros, J. (2011). Urban morphology of unplanned settlements: The use of spatial metrics in VHR remotely sensed images. *Procedia Environmental Sciences*, 7:152–157.

Kuffer, M., Pfeffer, K., and Sliuzas, R. (2016a). Slums from Space — 15 Years of Slum Mapping Using Remote Sensing. *Remote Sensing*, 8(6):1–29.

Kuffer, M., Pfeffer, K., Sliuzas, R., and Baud, I. (2016b). Extraction of Slum Areas From VHR Imagery Using GLCM Variance. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(5):1830–1840.

Kuffer, M., Sliuzas, R., Pfeffer, K., and Baud, I. (2015). The utility of the co-occurrence matrix to extract slum areas from VHR imagery. The case of Mumbai, India. *2015 Joint urban remote sensing event (JURSE)*, pages 3–6.

Marsaglia, G., Tsang, W. W., and Wang, J. (2003). Evaluating Kolmogorov's Distribution. *Journal of Statistical Software*, 8(18):1–4.

MathSoft (1999). Classification and regression trees. *Guide to Statistics*, 1(February):369–401.

Ng, A. Y. (2004). Feature selection, L 1 vs. L 2 regularization, and rotational invariance. In *Iinternational conference on Machine learning*, page 78.

Owen, K. K. and Wong, D. W. (2013). An approach to differentiate informal settlements using spectral, texture, geomorphology and road accessibility metrics. *Applied Geography*, 38:107–118.

Patino, J. E. and Duque, J. C. (2013). A review of regional science applications of satellite remote sensing in urban settings. *Computers, Environment and Urban Systems*, 37:1–17.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python Gaël Varoquaux. *Journal of Machine Learning Research*, 12:2825–2830.

QGIS Development Team, . (2015). QGIS 2.8 Wien. QGIS Geographic Informatio System.

R Core Team (2013). *R: A language and environment for statistical computing.* R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Ruiz, L. A., Recio, J. A., Fernández-Sarría, A., and Hermosilla, T. (2011). A feature extraction software tool for agricultural object-based image analysis. *Computers and Electronics in Agriculture*, 76(2):284–296.

Sandborn, A. and Engstrom, R. N. (2016). Determining the Relationship Between Census Data and Spatial Features Derived From High-Resolution Imagery in Accra , Ghana. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(5):1970–1977.

Schöpfer, E., Tiede, D., Lang, S., and Zeil, P. (2007). Damage assessment in townships using VHSR data the effect of Operation Murambatsvina / Restore Order in Harare, Zimbabwe. *2007 Urban Remote Sensing Joint Event, URS*, pages 2–6.

Stellmacher, G. (2011). Mapping Kibera: New strategies for mapping and improving the slum. Technical report, University of Washington, Seattle, Washington.

Tapiador, F. J., Avelar, S., Tavares-Corrêa, C., and Zah, R. (2011). Deriving fine-scale socioeconomic information of urban areas using very high-resolution satellite imagery. *International Journal of Remote Sensing*, 33(21):6437–6456.

Taubenböck, H. and Kraff, N. J. (2014). The physical face of slums: a structural comparison of slums in Mumbai, India, based on remotely sensed data. *Journal of Housing and the Built Environment*, 29(1):15–38.

Taubenböck, H., Wegmann, M., Roth, A., Mehl, H., and Dech, S. (2009). Urbanization in India – Spatiotemporal analysis using remote sensing data. *Computers, Environment and Urban Systems*, 33(3):179–188.

Tibshirani, R. (1996). Regression Selection and Shrinkage via the Lasso.

UN-Habitat (2015). Habitat III Issue Papers 22 – Informal Settlements. *United Nations Conference on Housing and Sustainable Urban Development*, 2015(May):0–8.

UN-Habitat (2016). Slum Almanac 2015/2016. Tracking improvement in the lives of slum dwellers. Technical report, Participatory Slum Upgrading Programme, Nairobi.

Yang, X., Jiang, G.-M., Luo, X., and Zheng, Z. (2012). Preliminary mapping of high-resolution rural population distribution based on imagery from Google Earth: A case study in the Lake Tai basin, eastern China. *Applied Geography*, 32(2):221–227.