Hochschule
für Technik
Stuttgart
University of Applied Sciences

Master of Science Programme
Photogrammetry and Geoinformatics
Master Thesis
Summer Term 2022

Assessing the spatial transferability of Random Forest models trained on the basis of WSF3D dataset for cities

by

Amita Gnana Prakash

Supervisors: Prof. Dr. Dietrich Schröder
Dr. Thomas Esch

Assessing the spatial transferability of Random Forest models trained on the basis of WSF3D dataset for cities

by

Amita Gnana Prakash

A dissertation presented in partial fulfillment of the requirements for the degree of Master of Science in the Department of Geomatics, Computer Science and Mathematics, Stuttgart University of Applied Sciences

Declaration

The following Master thesis was prepared in my own words without any additional help. All used sources of literature are listed at the end of the thesis.

I hereby grant to Stuttgart University of Applied Sciences permission to reproduce and to distribute publicly paper and electronic copies of this document in whole and in part.

Stuttgart, 29.04.2022

Amita Gnana Prakash

Approved by:

Prof. Dr. Dietrich Schröder

Acknowledgement

Thanks to the Almighty!

First of all, I extend my sincere gratitude to my supervisors, for accepting to supervise my thesis and their constant support - My first supervisor Prof. Dr. Dietrich Schröder for his guidance, useful recommendations and timely support; my second supervisor Dr. Thomas Esch for his support in helping me acquire the World Settlement Footprint 3D dataset and for granting access to resources that helped me complete this thesis.

I would like to thank Ms. Daniela Palacios López for her constant support, sharing her technical skills and guiding me throughout the thesis.

I extend my thanks to the other members of faculty of the Hochschule für Technik, Stuttgart for helping me to evolve academically.

My sincere thanks to the team of German Aerospace Center and World Resources Institute for the World Settlement Footprint 3D dataset and the Land Use / Land Cover maps of India respectively.

I would like to express my heartfelt thanks to my parents Amala, Prakash, Tamil and John, my uncle Xavier, my sister Anita and my brothers Ashwin and Stephen for their never ending love and encouraging me in all walks of life. To my friends Jeri and Thillai, thank you for your love, motivation and being there for me.

This thesis is dedicated to my husband for being my pillar of support and without whom this would not have been possible.

Augustine, this is for you!

Assessing the spatial transferability of Random Forest models trained on the basis of WSF3D dataset for cities

Abstract

The main objective of this research is to evaluate if the Random Forest models trained on the basis of spatial metrics derived solely from the WSF3D dataset can be transferred from one city to another.

Though with the increased availability of remotely sensed data, new machine learning techniques are constantly emerging for land use mapping, the challenges of collecting validation data and spatial transferability are yet to be addressed.

The WSF3D dataset and the technique of 'Dissimilarity Index' are used to address these challenges.

The main factors allowing model transferability; the association between prediction accuracies and transferability of cities; and the morphological similarities existing between transferable cities are analysed.

The 'Area of Applicability' is identified, to make assessments for successfully transferring a model to areas where validation data is not available.

Keywords: World Settlement Footprint 3D, Land Use Land Cover, Land Use Mapping, Remote Sensing, Machine Learning, Spatial Transferability, Random Forest, Spatial metrics, Structural similarity, Urban Settlements, Class separability, Similarity Measures.

Table of Contents

Table of Figures

Table of Tables

Abbreviations

GDP	Gross Domestic Product
LULC	Land Use / Land Cover
VHR	Very High Resolution
WSF3D	World Settlement Footprint 3D
DI	Dissimilarity Index
AOA	Area of Applicability
Q1	First quartile
Q3	Third quartile
IQR	Inter Quartile Range
IRS	Indian Remote Sensing
AWiFS	Advanced Wide Field Sensor
SVM	Support Vector Machine
MLC	Maximum Likelihood Classifier
NN	Neural Network
CAS	Cellular Automata Simulation
RF	Random Forest
CART	Classification And Regression Tree
NDVI	Normalised Difference Vegetation Index
EVI	Enhanced Vegetation Index
ESA -CCI - LC	European Space Agency Climate Change Initiative Land Cover Data
GLC – FCS	Global Land Cover Fine Classification System
EEA	European Environment Agency
MSME	Micro Small and Medium Enterprises

1 Introduction

In this chapter, an introduction to the thesis work is presented in four sections. In the first section, the scientific relevance behind this research is discussed. The second section explains the motivation and focus behind the thesis. In the third section, the goal, research questions and objectives of the thesis are presented. In the fifth section, the region of interest used for this study is introduced. And in the final section, the organisation of work adopted for this research work is summarised.

1.1 Scientific relevance

Urban areas have been the hub of civilisation since time immemorial. Recently they account for approximately 60% of the global Gross Domestic Product (GDP), over 60 % of the global resource consumption and 70% of gas emissions of the world. Today, more than 50% of the world population resides in urban areas, and this number is projected to increase up to 68% by the year 2050 (United Nations Department of Economic and Social Affairs, 2018). Besides a growth in population, regions such as Asia, Africa and Latin America witness an increase of 65%, 50% and more than 33.4% in terms of built-up footprint (European Commission, 2019). In this context, cities without proper planning experience problems such as unprecedented slum growth (Rasool Rehana et al., 2016) resulting in reduced quality of life (Ray Bhaswati, 2016), deterioration of environmental quality (Imam U.K. Aabshar et al., 2016), sanitation hazards (Beard A. Victoria et al., 2022) etc.

Therefore, to promote sustainable urban growth it is important to understand its trends and spatial structure. It is at this stage where urban land use classification and mapping play a pivotal role (Chen Bin et al., 2021). Additionally, the prevalent information on land use changes and patterns is also important for domains such as environmental monitoring (Sauti Raymond et al., 2022), disaster management (Sheykhmousa Mohammadreza et al., 2019, Yulianto Fajar et al., 2020), conserving biodiversity (Sharma Roshan et al., 2018) to name a few. In the framework of the United Nations Sustainable Development Goals (UN-SDG), land use data is also used to support and monitor different indices and indication, in particular Goal 11 “Sustainable Cities”, which aims at promoting sustainable urbanisation (United Nations Department of Economic and Social Affairs, 2021).

Mapping urban land use is identifying the nature and concentration of human activities taking place in a location, classifying the location represented by a pixel or a group of pixels according to their human use and assigning them categories accordingly (Huang Bo et al., 2018). Starting from the period of green revolution in 1950, remote sensing plays a vital role in classifying land use. Remotely sensed data also known as ancillary data gives information about the topography, climate, hydrology etc. of the region of interest. This information facilitates the creation of classes where areas displaying similar spectral characteristics are grouped together (Hermosilla Txomin et al., 2022). Since 2000s, researches are performed by employing methodologies which combine ancillary data and spatial metrics (Parker C. Dawn et al., 2004). Spatial metrics provide descriptive information of a landscape with respect to its spatial pattern (Herold Martin et al., 2003) and when coupled with remotely sensed data refine land use mapping results (Cengiz Serhat., 2022).

In order to handle the increased availability of remotely sensed data as a result of the dynamic nature of land, new research and methodologies are constantly emerging aiming at improving the production of land use and land cover maps (Masolele N. Robert et al., 2021). Today, with the introduction of Artificial Intelligence land use classification is performed for larger territorial extents using few human inputs (Alshari A. et al., 2021). Methods combining a semi-automatic supervised classification technique such as Random Forest (Zhang Hongsheng et al., 2012, Hernandez et al., 2017), Support Vector Machine (Mondal Arun et al., 2012, Singh Kumar Sudir et al., 2013) and neural network (Pacifici Fabio et al., 2009, Erbek Sunar F. et al., 2004) etc., are gaining particular attention, due to their high accuracy and performance. However, though new machine learning classifiers provide promising results for land use classification, a set of relevant challenges are yet to be completely addressed.

The first challenge is that high resolution land use maps are not yet available over large territorial extents. On one hand, most of the existing products[1] have spatial resolutions that range from 1 kilometer to 30 meters (Zhang Xiao et al., 2021). While they provide global coverage, they have the limitation of inadequate demarcation of the different land use classes, limiting their applicability and accuracy for further analyses (Carrão Hugo et al., 2007). On the other hand, research that reports the highest accuracies on land use classification are based on methods that employ very localized and often expensive ancillary data like using very high resolution (VHR) satellite imagery or LiDAR data etc. (Chang Shouzi et al., 2020, Mao Wanliu et al., 2020). This

[1] Examples of existing global land cover products: Global Land Cover product by the Copernicus Land Service at 100 meter resolution, land cover maps by the Climate Change Initiative project of European Space Agency at 300 meter resolution (European Space Agency, 2014), GLC-FCS30 at 30 meter resolution (Zhang Xiao et al., 2021).

means that while their methods are efficient, they have weak spatial transferability thereby limiting the analysis to single cities or small territorial extents. The second challenge is that the existing frameworks for semi-automatic land use classification are also affected by the time consuming task of training sample annotation (Han Wei et al., 2018). This is especially true for land use mapping over large spatial extents and remote areas, where manual collection of training data is unrealistic and reference data to automate the process of training data collection and validation tasks are often unavailable, or lack proper validation such as Open Street Map datasets (Hu Tengyun et al., 2016).

Therefore, generating reliable, cost effective and consistent large scale land use products necessitates of 1) accessible high resolution products with large scale coverage that could be used as ancillary datasets which provide descriptive information and aid in identifying the associations between different land use classes in the region of interest (Zhang Xiao et al., 2019) and 2) methods that allow evaluating the spatial transferability of machine learning models with limited amount of data over large territorial extents (Wenger J. Seth et al., 2012). In this context, the World Settlement Footprint 3D (WSF3D) dataset and the technique of Dissimilarity Index which is introduced by Meyer Hanna et al. (2021) open a new window of opportunity to solve the challenges in the field of large-scale land use mapping discussed earlier.

1.2 Research motivation and focus

Based on the scientific background presented in section 1.1, the main motivation of the present thesis is to respond to the identified challenges in the field of large-scale land use mapping, by exploiting remote sensing solutions coupled with state-of-the-art products and methods. As mentioned earlier, two new developments namely the WSF3D dataset and the method of Dissimilarity Index by Meyer Hanna et al. (2021), in particular, have the potential to address the challenges in large scale land use mapping.

On one hand, the WSF3D is a global dataset that belongs to the World Settlement Footprint suite and comprises of the Building Fraction, Building Height, Building Volume and Average Height layers. Research shows that spatial metrics derived solely from the WSF3D dataset used to identify “residential” and “non residential” classes at a Pan European scale yielded significant results (Palacios-Lopez Daniela., 2021). On the other hand, Meyer Hanna et al. (2021) have developed a methodology to calculate the Dissimilarity Index which measures the similarities in properties between regions. This technique can be used to validate the spatial transferability of machine learning models for larger territorial extents when few data exist for validation (Meyer Hanna et al., 2021).

By implementing these solutions, 1) the user would be able to assess if a model trained in one city can be transferred to another city when validation data is not available 2) large scale land use mapping with minimum user intervention would be possible 3) from an economic point of view, cost cutting would be possible due to lesser resources (reduced needs for collecting validation data, less manual intervention etc.) utilised for the mapping process 3) the results would in turn aid in large scale population modelling, efficient urban planning etc. In this context, following on the exploration of the mapping capabilities of the WSF3D dataset, one question that remains open is whether models trained on the basis of these dataset are transferable across space. To the best of this author's knowledge, no comprehensive assessment has been carried out within this direction, including an exploration of the mapping capabilities of the layer outside Europe. Therefore, coupled with the method developed by Meyer Hanna et al. (2021), this thesis aims to explore the spatial transferability of models trained with spatial metrics derived from the WSF3D dataset, thereby contributing to exploring the mapping capabilities of the WSF3D dataset for land use classification.

1.3 Research goal, questions and objectives

1.3.1 Research goal

As outlined in section 1.2, the goal of this master's research is to investigate the spatial transferability performance of models trained with spatial metrics derived from the WSF3D dataset. For this, the state-of-the-art methodological framework presented by Meyer Hanna et al. (2021), which allows to examine whether models trained at one city can be transferred to another city in the absence of validation data is explored.

1.3.2 Research questions

The main research question that this thesis attempts to answer is:

How spatially transferable are the models trained on the basis of the WSF3D dataset?

This is answered in a series of sub questions which are as follows:

- Are there any distinct patterns in prediction accuracies of the cities spatially?
- To which area can a model trained be transferred with confidence?
- Are there any associations between prediction accuracies and model transferability?
- Are there any similarities between cities based on their transferability performances?

1.3.3 Objectives

To address the research questions, the following objectives are followed:

- Extract training data from the eleven cities in India to identify three land use classes namely industrial, residential and small non residential.
- In an iterative manner calculate the Dissimilarity Index between the training data of one city against the training data of the other cities and assess its validity.
- Identify if there are any existing patterns of prediction accuracies observed between points above and below Dissimilarity Index threshold.
- Identify the main factors allowing good transferability.
- Assess if there is an association between prediction accuracies and transferability performances of cities.
- Analyse if a high prediction accuracy implies that successful model transferability.
- Assess the morphological similarities existing between cities according to their transferability performances.

1.4 Study area

For this thesis, research is carried out in eleven cities of India (see Figure 1) namely Ahmedabad, Belgaum, Hindupur, Hyderabad, Jaipur, Kanpur, Malegaon, Parbhani, Pune, Singrauli and Sitapur. This study area has been chosen for this master’s research since the World Resources Institute has prepared comparable and standardised land use / land cover maps of the above mentioned cities, with 5 meter spatial resolution which are used to validate the results. These land use / land cover maps are generated using Sentinel 2 imagery and ground truth collected from the Atlas of Urban Expansion, employing Convolutional Neural Network (CNN) machine learning algorithm. The maps originally display six land use classes such as open space, roads, water, non residential, residential formal and no data (Kerins Peter et al., 2021). For this research, the non residential and residential formal land use classes in particular are used for validation purposes (see Figure 2).

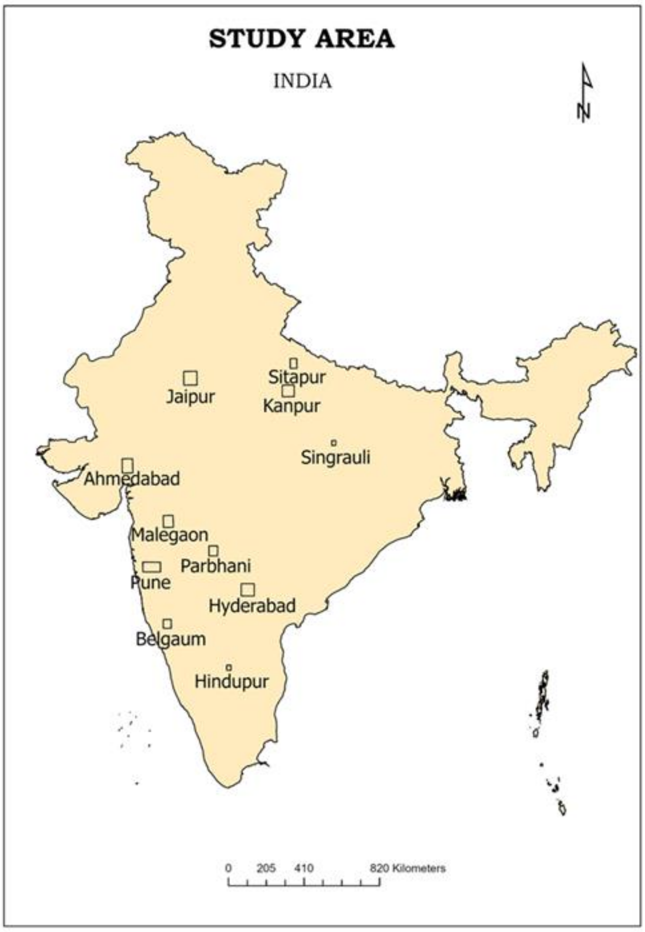


Figure 1: Study area (Source: Author).

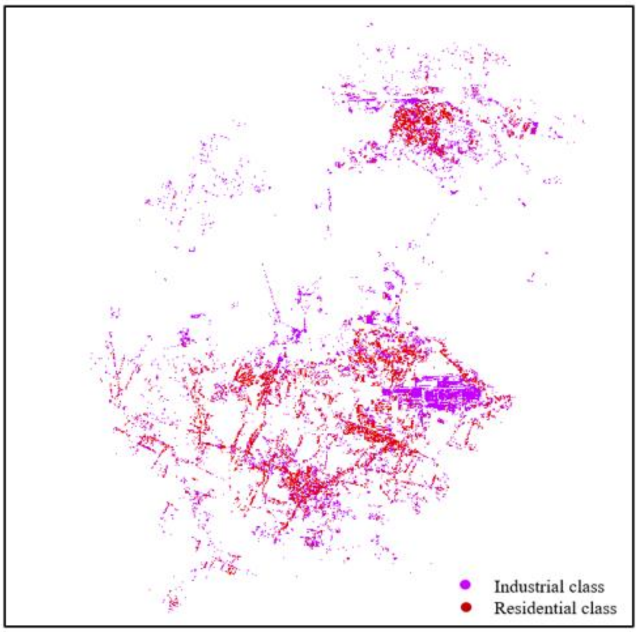


Figure 2: Land Use / land cover map of Singrauli showing industrial and residential classes (Source: World Resources Institute, 2021).

1.5 Organisation of work

In this section, the content orientation of the master's research is discussed. The thesis is divided into six chapters.

To begin with, the "Introduction" chapter gives an overview of the scientific relevance behind the thesis, motivation behind the research, goal of the thesis, research questions, objectives and study area.

In the second chapter namely "Theoretical background", the basics and concepts adopted in this research such as Random Forest classifier, spatial metrics, spatial transferability and Area of Applicability are discussed.

The third chapter which is "Literature review" presents the findings of the different research literature reviewed. In this section, an attempt is made to answer the questions such as - the reasons for employing Random Forest as the machine learning algorithm for this study, the significance behind using spatial metrics for this research and the scope of spatial transferability in land use classification.

In chapter four which is "Data and Methodology", a summary of the input dataset used for this study is presented, followed by the methodological framework used for implementing the research.

Chapter five namely "Results and Discussion" presents the findings of the master thesis followed by a discussion where the highlights of the results and an attempt to answer the research questions is made.

In the final "Conclusion" chapter, a summary of the research work is presented, followed by recommendations for future work.

2. Theoretical Background

The chapter on theoretical background presents the basics and concepts adopted in this thesis. Initially, an overview of Random Forest which is the machine learning algorithm employed in this research for training the model is presented. It is followed by a discussion on spatial metrics as this research uses as input the spatial metrics derived solely from the WSF3D dataset. Finally, the basics behind spatial transferability and Area of Applicability are discussed, in order to introduce the fundamentals behind the methodology adopted for this research.

2.1 Random Forest classifier

Random Forest is an ensemble machine learning algorithm recently used widely in land use / land cover applications. Being an ensemble method, it combines several models (decision trees in this case) that together produce one prediction model. Random Forest Classifier consists of several decision tree classifiers and each decision tree is formed by a vector which is a randomly sampled subset of the input vector (Pal M, 2005). This process is called bootstrapping. The sample which is not involved in the bootstrapping process is known as “out of the bag (OOB)” data. It gives an “out of the bag” error which is used to assess the prediction performance of the algorithm. As shown in Figure 3, each tree produces an output (a particular land use / land cover class) and the class with the maximum number of votes is shown as output.

For an effective learning process by any machine learning algorithm, its default parameters could be tuned. Belgiu Mariana et al., 2015 state that the two most important parameters in a Random Forest Classifier are NTree (number of decision trees) and Mtry (the random set of variables used to determine the point where the split is made). Probst Philipp et al. (2019) state that for an effective learning process and prediction performance of a Random Forest algorithm, the measure of observations belonging to each tree, the quantity of variables under one split, the minimum samples a node can have and the quantity of trees used need to be tuned.

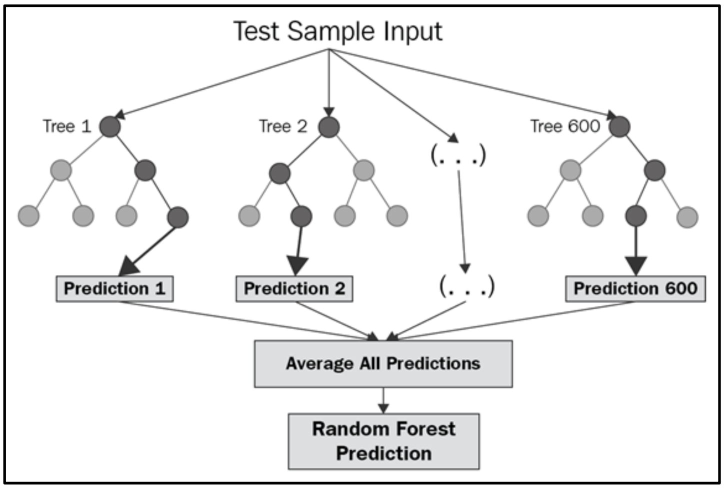


Figure 3: Working of a Random Forest algorithm (Source: Chakure Afroz, 2019).

2.2 Spatial metrics

According to Herold Martin et al. (2005) spatial metrics are “measurements derived from the digital analysis of thematic-categorical maps exhibiting spatial heterogeneity at a specific scale and resolution”. Originally developed in the late 1980s as landscape metrics, spatial metrics is based on the concept of patches which are entities with homogenous pixels falling under a particular land use/ land cover class**.** Spatial metrics are quantitative measures such as mean, density, standard deviation etc. that are used to describe objects on land.

A number of spatial metrics combined together are used to assess urban land use and detect land use changes. Research shows that land use mapping accuracy can be improved by using a combination of spatial metrics and remotely sensed data rather than using only remotely sensed data, since the integration of spatial metrics with remote sensing provides a consistent and elaborate insight on the urban landscape (Thapa Bahadur Rajesh et al., 2009, Effati Fatemeh et al., 2021). According to Prastacos Poulicos et al. (2017), a number of spatial metrics namely volume, area, perimeter, inter patch distance, nearest neighbour distance, fractal dimension, contagion, edge density, percentage of a land use class in the urban area etc. are proposed in various literature. Utilising the potential of spatial metrics is limited to domains such as landscape studies where they are widely used for temporal predictive modelling. Herold Martin et al., 2005 call for further research into employing it for urban modelling. Recently they are being integrated along with multi source data such as remote sensing, point of interest, night light data etc. in machine learning algorithms for spatial prediction modelling (Grippa T. et al., 2018, Hernandez et al., 2018), especially due to their ability to identify the smallest differences between land use classes and produce accurate results.

2.3 Spatial transferability and Area of Applicability

Spatial transferability is the capacity of the model to make accurate predictions in areas where training data is not available. It is the process of training a model in one area and transferring it to another area (Junttila Samuli et al., 2022). During spatial transferability, there is a problem that the new space might have properties different from the space where the model is trained. This causes errors in predictions. To overcome this problem and model spatial gaps in an effective manner, Hanna Meyer et al., 2021 train a model in a location and use it to predict a new space by using the Dissimilarity Index and Area of Applicability approach.

As discussed in section 1.2, Dissimilarity Index measures how similar two areas are based on their properties. Each training point has a Dissimilarity Index, from which a Dissimilarity Index threshold is calculated (see Figure 4). The threshold is defined by values that fall below the “upper whisker- larger than 75 percentile plus 1.5 times the interquartile range of the dissimilarity index values”. According to Meyer Hanna et al., 2021, all points below the threshold are suitable for transferring the model, since they yield high prediction accuracies when transferring the model. The area where points below threshold lie is called as the Area of Applicability. In Figure 4, the subset image shows a box plot which represents the Area of Applicability and where the model transferability can be applied. It is based on the concept that not all locations have the same properties and is the area where the spatial prediction model is successfully transferred.

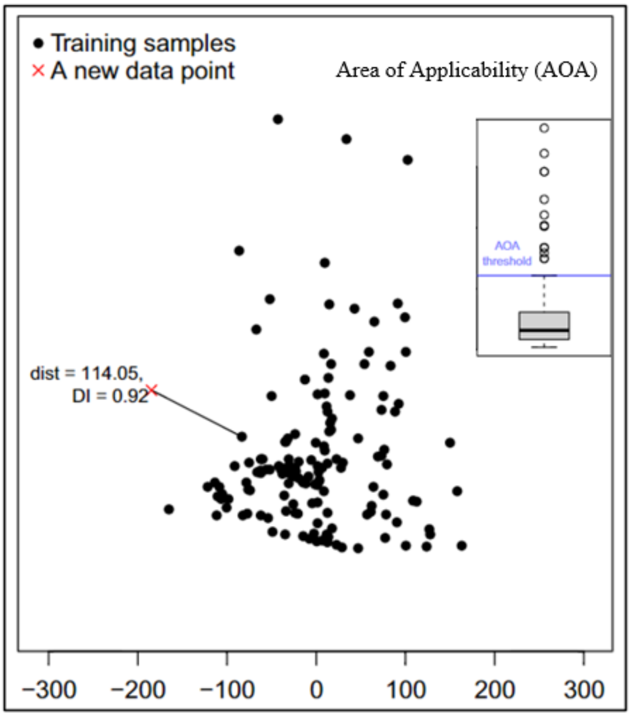


Figure 4: Area of Applicability (Source: Meyer Hanna et al., 2021).

3. Literature Review

A number of research work has been carried out by using machine learning techniques, remotely sensed data and spatial metrics in the field of land use classification. This chapter discusses the findings from different research studies that prove beneficial for this thesis. The first section presents the different machine learning classifiers used in mapping land use / land cover. In the second section, the reasons for employing Random Forest classifier for this study is discussed. The third section explains how a combination of spatial metrics and remotely sensed data yields accurate results. In the fourth section, an overview of research studies involving spatial transferability of prediction models is presented. And in the final section, the key points and highlights from review of literature is summarised.

3.1 Scope of machine learning in mapping land use / land cover

Starting from using manual methods in 1950s to the recent machine learning methods, new techniques are constantly emerging to map land use / land cover (Alshari A. Eman et al., 2021). This section presents the scope of machine learning in land use / land cover classification. First, the different machine learning algorithms used in land use / land cover classification is presented. Second, a comparison of the machine learning classifiers is discussed. In the final sub section, the observations made from the literature reviewed are presented.

3.1.1 Use of different machine learning algorithms in land use / land cover classification

One of the earliest and most simple classifiers in land use mapping is the decision tree method. Several research on using this approach for land use classification has been made. For instance, Kandrika Sreenivas et al., 2008 use multi-temporal IRS P6 Advanced Wide Field Sensor (AWiFS) data for land use land cover classification of Orissa, India using decision tree classifier. Sixteen land use classes are mapped and an accuracy of 87.5% is obtained. Punia Milap et al., 2011 use the same dataset for classifying the land use of Delhi, India. They get an accuracy of 89% for mapping eight land use classes. The authors observe that the classifier is flexible, simple, efficient and provides satisfactory result by using the temporal dataset.

Neural network techniques known for their accuracy and performance was introduced in the early 90s in the field of remote sensing. Ritter D. Niles et al., 1990 classify the land cover of Washington, USA using Artificial Neural Network and Landsat Thematic Mapper image. They observe that it returned an accuracy of 70% as compared to that of Bayesian method which returned 59% accuracy. In recent times, to overcome the limitations of Convolutional Neural Network with respect to multispectral and hyperspectral imageries, Hu Yunfeng et al., 2018 propose a deep convolutional neural network using the Landsat 8 Operational Land Imager dataset and obtain an accuracy of 82%. They state that though the results are accurate, the method is time consuming and requires powerful hardware to run effectively.

Introduced in 1992, Support Vector Machines (Hamed Rahel, 2020) are observed to produce accurate results in the case of clear demarcation of boundaries between the classes. Singh Kumar Sudhir et al., 2014 use multi temporal Landsat images to map the land cover of Pichavaram mangrove forest, India. They obtain accuracies of 94.53%, 94.14% and 89.45% for the three different images. Fragou Sotiria et al., 2020 classify Landsat TM images using Support Vector Machine. They get an output of nine classes with 89.85%, 91.01% and 90.24% accuracies for the years 1993, 2001 and 2010 respectively and state that it performs better even by using a training dataset with limited size.

3.1.2 Comparison studies between machine learning classifiers

Huang C. et al., 2010 compare the performance of Support Vector Machine (SVM), Maximum Likelihood Classifier (MLC), Neural Network (NN) and decision tree classifiers using a Thematic Mapper image and six land cover classes are identified. They observe that Support Vector Machine gives overall stable output and the highest accuracy when using seven variables. However Neural Networks show more accurate results when using lesser variables. The least accurate output is given by Maximum Likelihood Classifier. It is also found that better results are obtained as the number of variables and the training data size is increased.

Petropoulos P. George et al., 2012 use Hyperion hyperspectral images to identify land cover classes employing Support Vector Machine and Artificial Neural Network algorithms. They observed that the Support Vector Machines produced a better accuracy of 91.9% to 98.4% than the other, the accuracy of which is 70.6% to 78.4%. The authors attribute the high accuracy to the ability of Support Vector Machine to find the optimal separation plane between the different classes.

For the city of Soran, Iraq Hamad Rahel, 2020 uses Sentinel 2A data to assess the functioning of Artificial Neural Network, Support Vector Machine and decision tree classifiers. He finds that the results of Artificial Neural Network are the most accurate with 90 %, followed by the Support Vector Machine with 65 % and decision trees with 60% accuracies and concludes that Artificial Neural Networks could be a solution for mixed pixel problems.

3.1.3 Observations

As discussed above, a number of research work carried out utilising machine learning techniques to classify land use / land cover are proved to produce accurate results. Additionally, there is also studies being performed to model land use / land cover changes using time series data and then using it to forecast changes in the future. For instance, Roy Bishal (2021) uses 2001- 2019 Landsat 8 images for land use/ land cover classification of Cox's Bazar district, Bangladesh employing Random Forest classifier results in an accuracy of 89.04%-95.24%. He then simulates the land cover changes for 2025 using Cellular Automata Simulation (CAS) and Random Forest regressor with the accuracy of CAS being 93.2% and that of Random Forest regressor between 87% and 99.8%.

However limited literature is found on exploring the capabilities of transferring a land use / land cover prediction model which is generated in one area to a different area. From literature review it is observed that Random Forest classifier provides potential solutions to address this limitation, in addition to requiring less computational time and resources compared to other machine learning classifiers discussed in sub sections 3.1.1 and 3.1.2. The significance of Random Forest classifier for land use / land cover mapping and why it is used for this research is discussed in the next section.

3.2 Significance of Random Forest classifier for land use / land cover mapping

Owing to its accuracy, processing speed, quality to rank the variables according to their ability to differentiate the classes and so on Random Forest classifier is widely used for land use / land cover classification (Belgiu Mariana et al., 2016) like in the case where Tavares Amador Paulo et al., (2019) use Sentinel 1 and Sentinel 2 images to map the land use/ land cover classification of the urban area of Belem, Brazil. The study aims to evaluate the results of fused dataset of the satellite images on land use mapping by employing a Random Forest machine learning algorithm. It is observed that by using the integrated dataset, better results (91.07%) are obtained compared to using separate data. Due to its growing popularity, several research is performed comparing its performance to other machine learning classifiers. A few examples of this work is reviewed in this section.

Talukdar Swapan et al., (2020) compare the performance of machine learning algorithms for land use land cover classification such as Random Forest, Support Vector Machines, Artificial Neural Network, fuzzy adaptive resonance theory, supervised predictive mapping, Spectral Angle Mapper and Mahalanobis based distance. The accuracy assessment results based on kappa coefficient, index-based classification show that the highest accuracy of 0.89 is obtained using Random Forest and they conclude that it is the most suited for land use / land cover classification.

Rodriguez-Galiano F. Victor et al., 2012 evaluate the performance of machine learning algorithms such as classification trees, Artificial Neural Network, Support Vector Machines and Random Forest for land cover mapping using Landsat 5 data. The results show that Random Forest produced the most accurate results (0.92) using only two parameters - number of classification trees and number of predictive variables. Accuracy assessment was carried out by checking overall accuracy, user's and producer's accuracy and kappa coefficient. They also assess the performance of the classifiers by including noise in the data. It is found that Random Forest and Support Vector Machines give robust results in the case of noise less than 25%. Furthermore, they evaluate the sensitivity of the classifiers to reduced training area size. It is found that the accuracy of Random Forest along with Support Vector Machines and Artificial Neural Network remain stable up to a reduction of 50% of the training area size.

A similar finding that Random Forest algorithm produces higher accuracy (91.3%) compared to classification and regression tree (CART) and maximum likelihood classification (MLC) is stated by Xiaodong Na et al., (2012) when they map wetlands in Saniang plain using Landsat TM image.

Chachondhia Prachi et al., 2021 evaluate the performance of Support Vector Machines and Random Forest algorithms using optical (sentinel 2), microwave data (sentinel 1) and a fusion of both datasets. They find that Random Forest algorithm provided better results with 92%, 94.3% and 37.1% accuracy for optical, microwave and fused data respectively. Upon visual analysis, they also observe that though optical and microwave data provided more accurate results, the fused dataset gives realistic classification results.

To conclude, Random Forest classifier is observed to show better results than the other classifiers in the case of land use / land cover mapping. Above all, its simplicity, less computation time, stability to noise less than 25%, ability to produce robust results up to 50% reduction of the training data size, ability to work with multisource data has made it a popular machine learning classifier for land use / land cover mapping.

3.3 Random Forest and spatial metrics in land use / land cover classification

As discussed in the chapter on theoretical background, spatial metrics explain the configuration of the landscape in terms of size, distribution and arrangement of the object on the ground within a given window and differences in these metrics are used to identify objects on the ground. For example, the spatial metrics of a residential area are normally smaller than those of industrial areas. Due to their capacity to differentiate the smallest difference between classes and the bettered accuracies of machine learning classifiers when combined with them, methodologies using spatial metrics are gaining increasing attention (Hernandez Ruiz Elias Ivan et al., 2017). A few instances of research work combining Random Forest classifier and spatial metrics in the field of land use / land cover classification is reviewed below.

Grippa T. et al., 2018 propose a technique to map urban land use at street block levels in which besides land cover maps and normalised digital surface models, street block features extracted from open street map data is classified using Random Forest method. The spatial metrics from the street blocks which are computed using FRAGSTATS for this research are patch number, patch density, mean patch size, standard deviation of patch size, patch size coefficient of variation, range of patch size, shape index and proportion. In this case, the spatial metrics are patch based since the landscape as a patch is considered. Prior to the classification, an automated feature selection method namely variable selection using Random Forest algorithm is done to avoid redundant spatial metrics. They produce a land use map with six classes and conclude that this almost automated procedure could be used for mapping large areas.

A method for mapping urban land use employing Random Forest algorithm and combining spatial metrics and texture analysis is proposed by Hernandez Ruiz et al., 2017. They prepare a land cover map using geographic object-based image analysis from which spatial metrics is obtained. The spatial metrics used in this study are selected after visually analysing the digital values obtained from the land cover map. They are as follows: class area, largest patch index, contagion, interspersion and juxtaposition index, edge density, landscape shape index, number of patches, patch density, total edge and perimeter area ratio. Along with texture metrics, they are then used to perform land use classification using Random Forest algorithm. An urban land use model is finally developed with an accuracy of 92.3%.

Xing Hanfa et al., 2018 integrate building and region based spatial metrics and socio-economic data obtained through crowdsourcing and use Random Forest algorithm to classify the different urban functions of Futian district, China. Here, the indices of building level blocks such as building area, height, structure and edge are used to calculate the spatial metrics. The building metrics used for the study include the total, percentage, mean, maximum of the building area, mean building height etc. and the region metrics include landscape shape index, mean patch fractal dimension and degree of landscape division. The result obtained shows an overall accuracy of 81.8%. Using an analysis of variance, they find that total building area, mean building edge, building edge standard deviation and building structure index standard deviation play an important role in the classification among the building metrics and among the region metrics, all contribute highly to the classification.

Zhang X.M. et al., 2017 apply Random Forest classifier on multitemporal spectral images from Landsat 5 of Wuhan, for land cover mapping. In addition, Due to the challenges of producing land cover classification maps in the region, pre-existing land cover maps prepared by the ecosystem and survey assessment project is used as reference data for generating training dataset and validating the classification results. The authors use spectral metrics such as NDVI, EVI and MNDVI, temporal features and spatial metrics like mean, variance, homogeneity, correlation, entropy, contrast, dissimilarity and angular second moment as input data. After observing the ranking of variables depending on their influence on classification, they are included / excluded from the process. Using stratified random sampling, reference data is chosen which is then split into training and testing data (one third of the reference data). The output shows an accuracy of 89.2%. The authors observe that the classified results are better than the reference maps and reducing the number of variables used could optimise the algorithm performance.

Apart from land cover mapping, spatial metrics are also capable of discriminating the different types of buildings such as commercial, industrial, administrative etc. For instance, Du Shihong et al., 2015 use GIS and Very High Resolution image to semantically classify urban buildings by applying a Random Forest classifier. The authors use spectral metrics such as mean, standard deviation, contrast border etc. and geometric metrics like area, border length, roundness etc. to identify different types of buildings. For buildings which are hard to visually distinguish, texture metrics such as homogeneity, entropy, contrast etc. are used. These metrics are derived using a two-level image segmentation process. This is followed by sample selection using a semi supervised method and finally the Random Forest classifier is applied. The output has an accuracy of 71.50%. The authors conclude by adding that shadows and occlusion in the image could affect the classification results and studies related to semantic classification of buildings could benefit in urban population studies.

To conclude, the accuracy of results produced by Random Forest are bettered by using spatial metrics which are able to identify the smallest differences in classes. When this data is available on a global scale, it is beneficial to obtain land use / land cover information in places which have limited access to very high resolution data sources.

3.4 Spatial transferability of prediction models

Spatial transferability of a model is the generation of a model in a location and using it to predict in a new location. In urban land use, the potential of model transferability is yet to be fully investigated. Few studies are found such as in the case where Chen Bin et al. (2021) compare different machine learning algorithms using inputs such as land surface, height, demography and social media data in addition to street block features obtained from open street map data and produce level 1 and level 2 land use maps. Then, they generate a model using training data gathered from one city and validate it using testing data from another city. The output shows relatable residential buildings among the five metropolitan cities studied, compared to other land use types. It is then concluded that model transferability holds true in case of areas with similar characteristics. However related work is extensively carried out in other domains such as environmental, ecological, social and behavioural sciences, which is discussed as follows.

Jin Schichao et al., 2018 use LiDAR derived canopy data, topographic, climatic data and optical images to test the transferability of a canopy height prediction model generated by Random Forest algorithm for USA. Different Random Forest models are generated according to the different vegetation types and locations using variables such as elevation slope aspect etc. The authors observe that a Random Forest model trained for a particular vegatation type or location could not be used to accurately predict the canopy height in another location or for a different vegetation type, the reason being the sensitivity of the model to a site. They find that increased spatial resolution leads to better prediction and conclude that this result could be improved by using representative samples from all locations.

Waśniewski Adam et al., 2020 use the spectral information from Sentinel 2 images, Digital Elevation Model and Normalised Difference Vegetation Index to classify the forest cover of Northwest cabin by applying Random Forest classifier. The Sentinel 2 image is divided into four sections. The Random Forest model is trained in one section and used to predict the remaining three sections. The authors find that among the variables used, Digital Elevation Model contributed most to the forest type classification. They conclude that the spatial transferability is successful however they face complexity because of the heterogenous nature of the environment and forest. They obtain an accuracy of 95% for the image where it is trained and an accuracy ranging from 90% to 83% for the new images where predictions are made.

Juel Anders et al., 2015 classify coastal vegetation type by applying Random Forest classifier using an object-based image analysis, Digital Elevation Models and aerial ortho photos. The authors observe an accuracy of 92.1% to 91.8% in the case where the training and validation data are taken from the same location and an accuracy of 85.8% to 81.9% when the training and validation data are spatially separated. When the trained model is applied to totally new locations, accuracies between 69% to 59.42% are obtained. They attempt to limit the node size parameter and reduce the number of variables to obtain better accuracies for the models transferred, however it remained the same. In contrary, they find improved results by increasing the number of variables.

In the field of disaster management Valentijn Tinka et al., 2020 use 175289 buildings belonging to xBD dataset, comprising of human annotated disaster labelling based on high resolution satellite images to assess the spatial transferability of a convolutional neural network model. Thirteen disasters are used for this study with different types of damages, image parameters and locations. 80% of the data is used for training, 10% for testing and 10% for validation in this study. The authors state According to the authors, the results are satisfactory for modelling building damage in cases where labelling is a challenge and that the accuracy of the model depends on the test disaster where it is transferred.

By applying a Random Forest classifier, Machado et al., 2019 use a soil map on a scale of 1:10,000, Digital Elevation Model and fourteen topographic indices including catchment slope, valley depth, plan curvature etc. to predict soil type. Variable reduction is carried out to optimise the performance of the algorithm. The authors observe that increasing the number of observations decreases the classification error. Also training data in the form of points leads to better accuracy than polygon training data. They conclude that finding the most representative training samples is essential for a successful spatial transferability.

Apart from classification models, there are also cases of regression models being successfully transferred to new spaces – a) Poplawski Karla et al., 2009 generate a land use regression model for Vancouver, Canada and transfer it to Victoria, Canada and Seattle, USA to study the nitrogen dioxide concentrations. They find that the model is transferred performed better in Victoria (R^2 = 0.51) than in Seattle with a value of R^2 = 0.33. According to the authors, input data consistency and similar geographical conditions play a major role in determining the success of model transferability. b) Marcon Alessandro et al., 2015 develop a land use regression model to measure nitrogen dioxide concentration in the Veneto region of Italy and transfer the model to Verona, Italy. Altitude, surface area, population and length of roads are among the 67 predictor variables used in this research. After getting a value R^2 = 0.18, the authors conclude that the model could not be successfully transferred to a region nested within the training region and also to not transfer a model generated on a region with background spaces or rural sites to an urban region.

As discussed above, the success of spatial transferability of a model depends on several factors such as similarities in properties of the environments where the model is trained and transferred, collecting the most representative training samples and so on. Besides, parameters related to the classifier such as number of input variables, size of training data also affect the performance of spatial transferability.

3.5 Key points

The following key points are observed after a review of literature:

- Machine learning algorithms are being developed continually and are observed to perform with high accuracies, in particular Neural Networks, Support Vector Machine etc. However, the machine learning classifiers discussed in sub sections 3.1.1 and 3.1.2 have certain limitations such as – high computation cost and time, and their scope on spatial transferability in the land use / land cover classification is yet to be completely investigated, to name a few.
- Random Forest Classifier with its flexibility, robustness, less computation time, ability to handle multisource data, stability against noise etc. is used widely for land use / land cover classification. It is observed to produce results with higher accuracy and is uses lesser computation time and resources compared to the machine learning algorithms discussed in section 3.1.
- Spatial metrics combined with Random Forest produce accurate results. If this data is available on a global scale, it can be used to produce high resolution land use / land cover information in areas where accessing very high resolution data is challenging.
- Spatial transferability of models is affected by factors such as – geographic similarity, finding the best training data, size of the training sample, number of input variables used and so on. Research on spatial transferability is actively carried out in environmental, ecological, social and behavioural sciences and yet to be carried out on a wide scale in the field of land use / land cover classification.

4. Data and Methodology

This chapter describes the input data, tools and methodological framework employed for this thesis and is divided into two sections. In the first section, the nature and scope of the input dataset used for the thesis which is the World Settlement Footprint 3D dataset is discussed. In the second section, the workflow of the methodology implemented for testing the spatial transferability of models trained using the spatial metrics derived from WSF3D dataset is explained in detail.

4.1. The World Settlement Footprint 3D dataset

The Global Urban Footprint was released by the German Aerospace Center (DLR) in 2016 with a 12 meter resolution. To overcome its limitations of high cost and images being affected by the prevailing conditions led to the development of the World Settlement Footprint 2015. It is an open-source dataset with a 10 metre resolution, the quality of which is evaluated by crowd sourced photo interpretations (Marconcini Mattia, 2020). The World Settlement Footprint suite consists of several products such as the World Settlement Footprint Evolution and the World Settlement Footprint 2019 (The European Space Agency, 2021). WSF3D is one of the components of the WSF Suite.

World Settlement Footprint 3D is developed by using TandDEM-X Digital Elevation Model (TDX-DEM) and its amplitude mask (TDX-AMP) with a 12 metre spatial resolution and the WSF Imperviousness layer with 10 metre spatial resolution. The WSF3D consists of thematic layers (see Figure 5) showing Building Fraction (BF), Building Height (BH), Building Volume (BV) and Average Height (AH) within each cell of size 12 meter. A summary of the mentioned thematic layers is as follows.

Building Height (BH), given in metres is a measure of the differences in heights at the building edges (BE). It is derived from the settlement areas given in the WSF Imperviousness layer in the12 meter TDX-DEM within the settlement areas defined by the WSF-Imp layer. Building Fraction (BF) is represented in percentage and its values range from 0-100. For producing this layer, the built-up area is measured from the WSF Imperviousness layer, TDX-AMP and BE. Building Area (BA) is given in square meters and produced by multiplying the BF times the area of each grid cell. Building Volume (BV), expressed in cubic meters is derived by multiplying the BH with the area of the 12m pixels. Building Mask (BM) is the binary layer representing the Building Fraction layer (Palacios-Lopez Daniela., 2021).

For this thesis, sixteen spatial metrics derived from the WSF3D dataset are used which are generated for the above mentioned layers in a 25 x 25 window. The metrics which are computed are mean (Building Area), median (Building Area), standard deviation (Building Area), Building Area, mean (Building Fraction), median (Building Fraction), standard deviation (Building Fraction), Building Fraction, mean (Building Height), median (Building Height), standard deviation (Building Height), Building Height, mean (Building Volume), median (Building Volume), standard deviation (Building Volume) and Building Volume. These sixteen metrics correspond to the sixteen bands in the multiband raster image used as input data for the study (Palacios-Lopez Daniela et al., 2022).

Applications of the World Settlement Footprint include the utility of the World Settlement Footprint mask along with the Global Urban Footprint 3D technique, to predict the Covid -19 pandemic risk hotspots in African, Asian and South American cities (Bhardwaj G. et al., 2020). According to Palacios-Lopez Daniela et al., 2022 WSF3D dataset is beneficial for large scale population modelling. She uses Random Forest Classifier and the spatial metrics of WSF3D dataset to classify industrial and non-industrial settlements in Europe, which returned an accuracy of 86%. She states that by adding volume and the settlement type information, the estimation of population could be improved. However further research is needed to validate the functioning of the dataset in regions outside Europe and also on developing transferable prediction models.

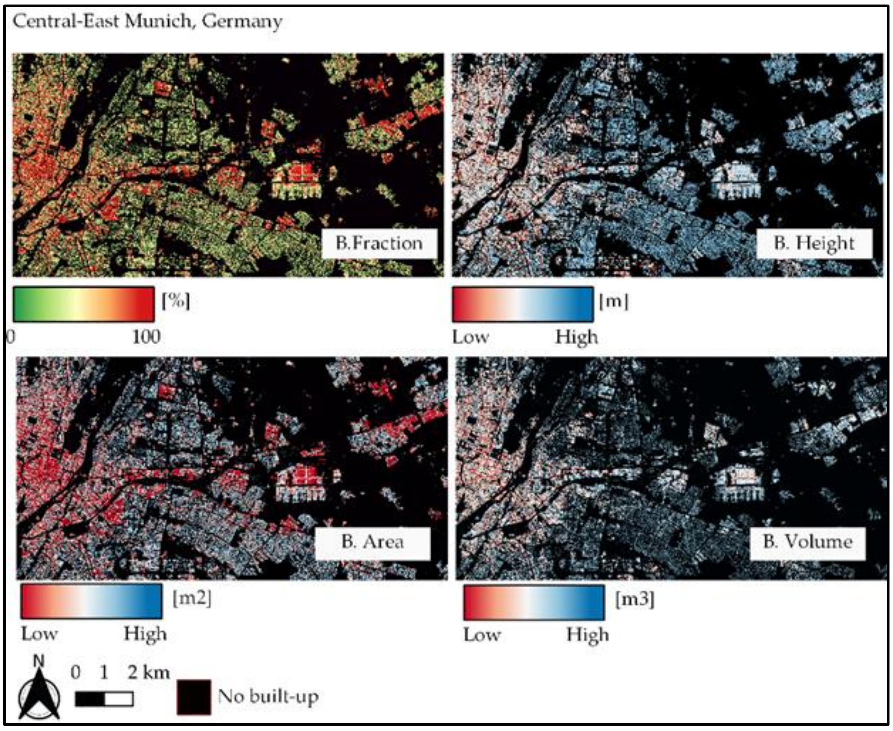


Figure 5: Thematic layers of WSF3D dataset (Source: Palacios-Lopez Daniela et al., 2022).

4.2 Methodology

In this section, the methodology (see Figure 7) used to achieve the research objectives is described. It is carried out in six steps namely – 4.1) collecting training samples, 4.2) training the Random Forest Model, 4.3) determining the Area of Applicability, 4.4) spatially transferring the trained model, 4.5) assessing the transferability performances of the models and 4.6) conditional classification. In section 4.1, the methods adopted for collecting training data and challenges faced during the process is summarised, followed by the parameters used to train the Random Forest model in section 4.2. Then a description of the Dissimilarity Index and Area of Applicability is given in section 4.3 after which the process of testing the model with other cities is explained in section 4.4. Section 4.5 discusses the steps taken to analyse the accuracies obtained for all the spatially transferred models. Finally in section 4.6, the process of conditional classification is presented. It is also to be mentioned that the whole methodology is an iterative process. In each iteration, one city is selected as the train city and all the other cities are used as validation cities where the trained model is transferred (see Figure 6). The tools used to execute this methodology are ArcGIS Pro 2.7.1 and Python 3.8 libraries namely OSGeo, Numpy, Pandas, Scikit learn and Scipy.

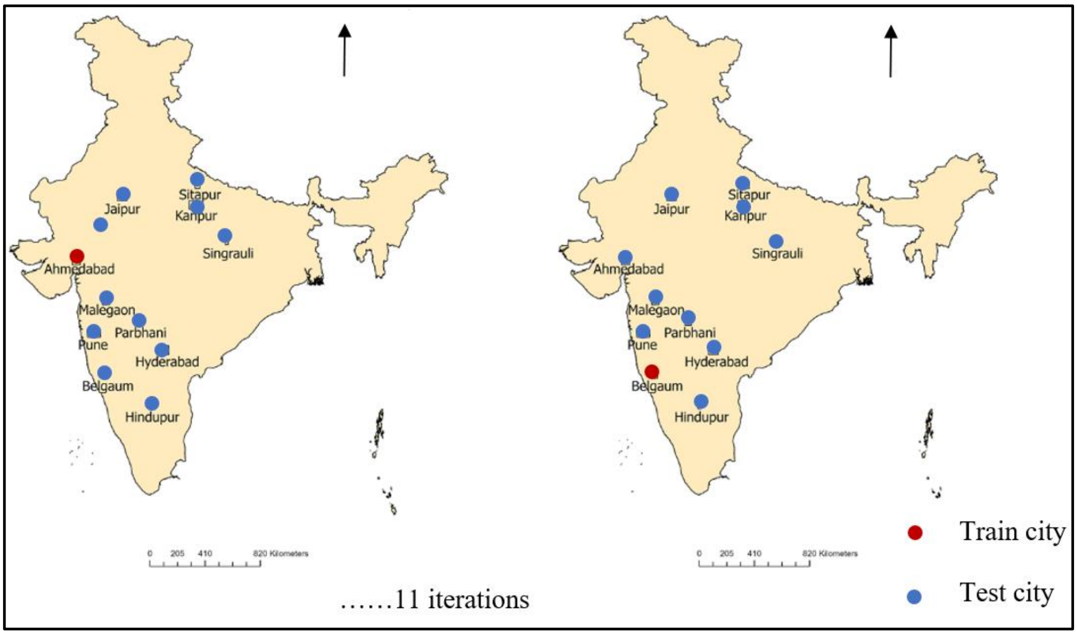
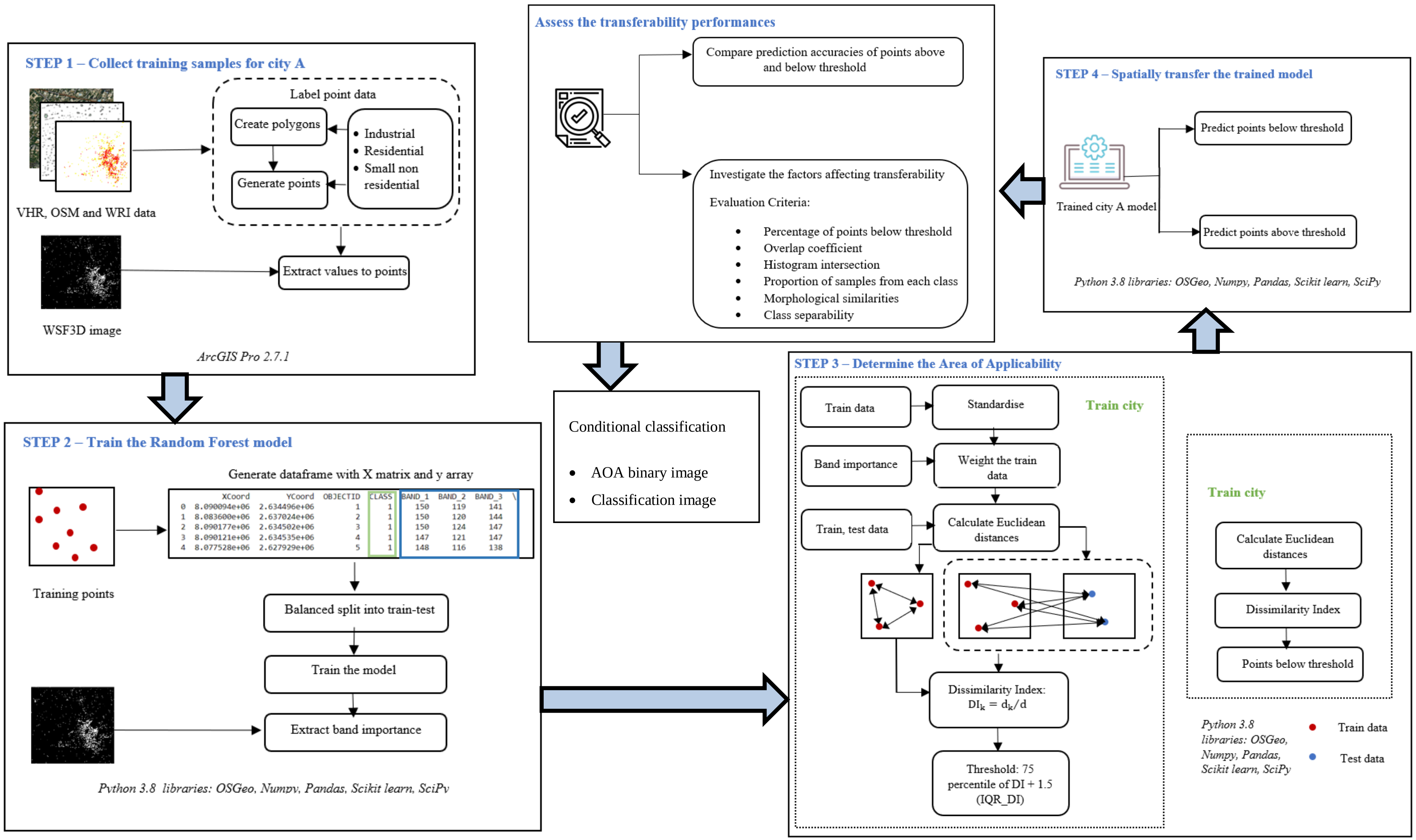


Figure 6: Iterative nature of research methodology (Source: Author).



STEP 1 – Collect training samples for city A
Label point data
Create polygons
Generate points
• Industrial
• Residential
• Small non residential
VHR, OSM and WRI data
Extract values to points
WSF3D image
ArcGIS Pro 2.7.1
STEP 2 – Train the Random Forest model
Generate dataframe with X matrix and y array
Training points
Balanced split into train-test
Train the model
Extract band importance
Python 3.8 libraries: OSGeo, Numpy, Pandas, Scikit learn, SciPy
Assess the transferability performances
Compare prediction accuracies of points above and below threshold
Investigate the factors affecting transferability
Evaluation Criteria:
• Percentage of points below threshold
• Overlap coefficient
• Histogram intersection
• Proportion of samples from each class
• Morphological similarities
• Class separability
Conditional classification
• AOA binary image
• Classification image
STEP 4 – Spatially transfer the trained model
Trained city A model
Predict points below threshold
Predict points above threshold
Python 3.8 libraries: OSGeo, Numpy, Pandas, Scikit learn, SciPy
STEP 3 – Determine the Area of Applicability
Train city
Train data
Standardise
Band importance
Weight the train data
Train, test data
Calculate Euclidean distances
Dissimilarity Index: DI_k = d_k/d
Threshold: 75 percentile of DI + 1.5 (IQR_DI)
Train city
Calculate Euclidean distances
Dissimilarity Index
Points below threshold
Python 3.8 libraries: OSGeo, Numpy, Pandas, Scikit learn, SciPy
● Train data
● Test data

	XCoord	YCoord	OBJECTID	CLASS	BAND_1	BAND_2	BAND_3	\
0	8.090094e+06	2.634496e+06	1	1	150	119	141	
1	8.083600e+06	2.637024e+06	2	1	150	120	144	
2	8.090177e+06	2.634502e+06	3	1	150	124	147	
3	8.090121e+06	2.634535e+06	4	1	147	121	147	
4	8.077528e+06	2.627929e+06	5	1	148	116	138	

Figure 7: Methodology workflow (Source: Author).

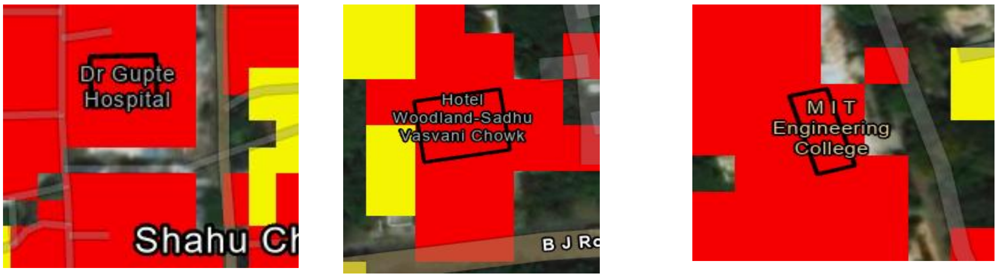
4.2.1 Collecting training samples

In this stage, the training data used for generating the Random Forest model is collected using a thorough investigation of multiple sources of data. As seen in Figure 7, this stage is divided into two parts namely collecting label point data and extracting multivalues to points.

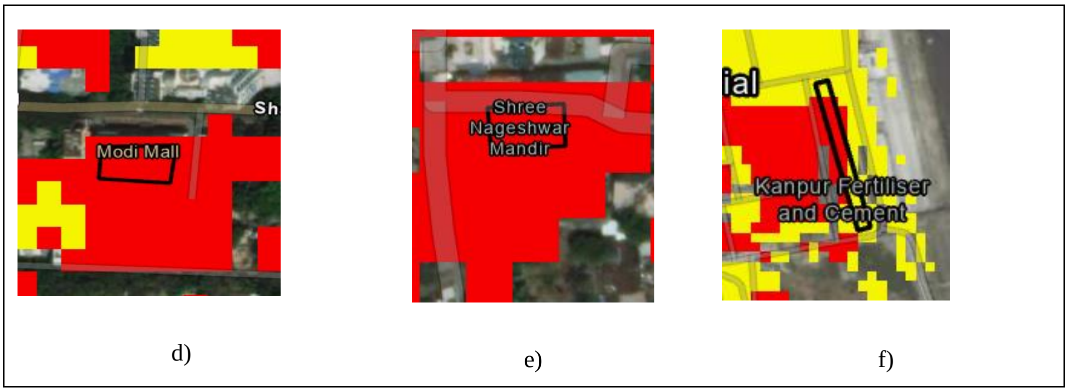
Collect label point data

In this step, initially polygons belonging to different land use classes are created after identified by visual assessment. Then points are generated inside these polygons, which are finally labelled according to their respective classes.

For collecting the training data, VHR satellite imagery, Open Street Map data, Land Use Land Cover Classification maps (World Resources Institute) and the WSF3D data are used. This research started with an objective to train the Random Forest model on the basis of two classes – residential and non residential. However, upon visual assessment it is found that a few pixels which belong to the commercial or retail sector such as hospital (see Figure 8 a), hotel (see Figure 8 b), college (see Figure c), mall (see Figure d) are misclassified as residential class. Also, as seen in Figure 8 e, a place of worship such as a temple is misclassified as residential class. And a few tall apartments and houses are misclassified under the non residential category. In addition, a few small buildings surrounding an industry are misclassified as residential class (see Figure 8 f). Thus, misclassifications are observed mainly between residential buildings and small industries or commercial buildings.



a) b) c)

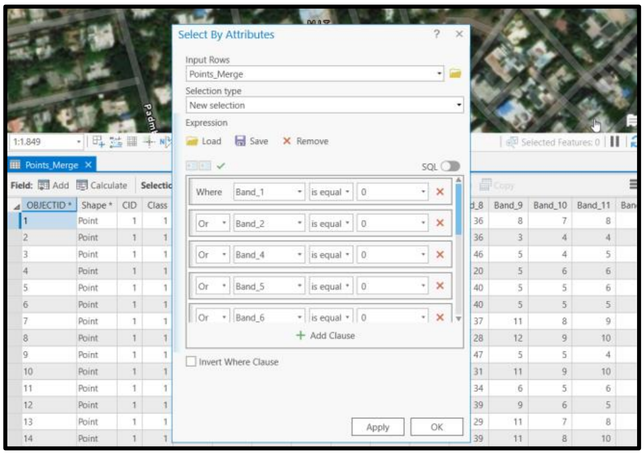


d) e) f)

Figure 8: Misclassifications in the Land Use / Land Cover Classification reference maps (World Resources Institute, 2021).In the figure, residential class is represented in red and non residential class in yellow.

Considering the visual assessment results and to avoid residential buildings misclassified as non residential and vice versa, the training data is collected in three classes namely – industrial, residential and small non residential. Under the industrial category, large industries are selected, for the residential category the polygons which are only residences are carefully selected after a thorough observation using VHR satellite imagery and Open Street Map data. Under the small non residential category, places of worship, bus stations, retail and commercial centers are selected. The steps undertaken for collecting the samples are as follows:

- After carefully selecting polygons in these four categories, new polygons are created manually ensuring that they fall only within the WSF3D masked area and not in places where no data is available for the WSF3D layer
- 150 points are then generated inside these polygons for each class. As shown in Figure 9, the points are then cross checked that none of the bands had a zero value except the bands containing standard deviation information (Bands 3, 7, 11 and 15).



Select By Attributes

Input Rows: Points_Merge

Selection type: New selection

Expression

Where Band_1 is equal 0
Or Band_2 is equal 0
Or Band_4 is equal 0
Or Band_5 is equal 0
Or Band_6 is equal 0

□ Invert Where Clause

OBJECTID *	Shape *	CID	Class
1	Point	1	1
2	Point	1	1
3	Point	1	1
4	Point	1	1
5	Point	1	1
6	Point	1	1
7	Point	1	1
8	Point	1	1
9	Point	1	1
10	Point	1	1
11	Point	1	1
12	Point	1	1
13	Point	1	1
14	Point	1	1

d_8	Band_9	Band_10	Band_11	Ban
36	8	7	8	
36	3	4	4	
46	5	4	5	
20	5	6	6	
40	5	5	6	
40	5	5	5	
37	11	8	9	
28	12	9	10	
47	5	5	4	
31	11	9	10	
34	6	5	6	
39	9	6	5	
29	11	7	8	
39	11	8	10	

Figure 9: Selection of points with no zero values.

Extract values to points

As discussed in section 3.5, each of the sixteen bands in the input raster image correspond to the sixteen spatial metrics of the WSF3D dataset. In this step, the values of the sixteen bands belonging to WSF3D dataset are then extracted to these points. The points with the extracted values are exported as tables which are used in the next stage.

To conclude, the training data is classified into industrial, residential and small non residential categories. 150 points each are collected for the four classes after extensive visual assessment of many data sources, for each of the eleven cities.

4.2.2 Training the Random Forest model

As discussed in the sub section 2.1 of the chapter on Theoretical Background, Random Forest is widely used in remote sensing due to its robustness, flexibility, less computation time etc. For this research, the classification is performed using a Random Forest classifier which is trained with the data collected in the previous step. The steps taken to train the Random Forest classifier, make predictions for test data and classify the image is given in this section (see Figure 7 and Figure10).

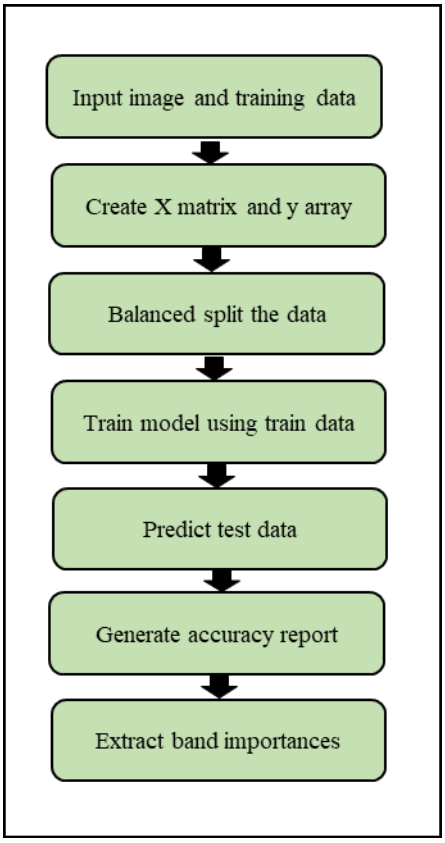
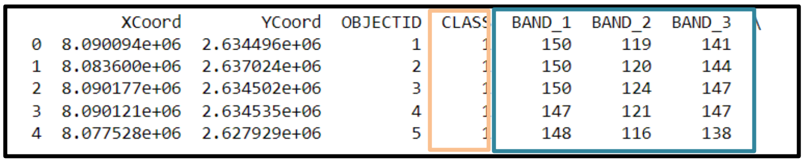


Figure 10: Stage two – Training the Random Forest model.

- For training the Random Forest model for a particular city, the point data generated in sub section 4.1.2 and the WSF3D image with sixteen bands for a city are used as input.
- Then, the point data which is originally in the form of a csv table is converted to a dataframe (see Figure 11). The X matrix for the model is created with the values from the sixteen columns containing the band information for all 600 points. The column containing the class information of all points is selected as y array.



```
          XCoord        YCoord  OBJECTID  CLASS  BAND_1  BAND_2  BAND_3  \
0   8.090094e+06  2.634496e+06         1      1     150     119     141
1   8.083600e+06  2.637024e+06         2      1     150     120     144
2   8.090177e+06  2.634502e+06         3      1     150     124     147
3   8.090121e+06  2.634535e+06         4      1     147     121     147
4   8.077528e+06  2.627929e+06         5      1     148     116     138
```

Figure 11: Dataframe with X matrix and y array (X matrix is highlighted in blue and y array in green).

- The data is then split in a balanced manner using ‘StratifiedShuffleSplit’ function in the Scikit learn Python library to ensure that all the three classes contain the same number of training samples. The data is split in such a way that 70% of the data is considered as train data and the remaining 30% as test data.
- The Random Forest model is trained on the train data and predicted using the test data. The best parameters for the model are found using the ‘GridSearchCV’ function in the Scikit learn Python library, and the model is trained accordingly.
- The accuracy assessment report is generated and it is made sure that the model is neither underfitting nor overfitting. Overfitting occurs when the model gives a high accuracy only for the data for which it is trained and not for the data for which it is tested. The model becomes so used to the train data set that it would not be able to detect the patterns in the test data and hence performs poorly (Allamy Khalaf Haider et al., 2015). The accuracy of the model is investigated to make sure that this bias does not occur.
- Finally, the importance of the sixteen bands in the WSF3D input raster image are extracted. Based on their roles in training the model, each of the sixteen bands is given a value which is automatically generated by the Random Forest classifier. This step is carried out because for the forthcoming section which is to calculate the Area of Applicability, one of the variables required is the importance of bands in training the model.

Thus, in this step a Random Forest model is trained using label point data as input. Then the training points are split in a balanced manner into test and train data, the model is trained using the train data and tested using the test data. Its performance is evaluated using confusion matrix and accuracy report. Using the WSF3D sixteen band image as input, the importance values of the bands are extracted.

4.2.3 Determining the Area of Applicability

As defined in the sub section 2.3 of the chapter on Theoretical Background, Area of Applicability is the region within which the Random Forest model is successfully transferred. It is assessed using a dissimilarity index (Meyer Hanna et al., 2021). The steps in calculating the Dissimilarity Index and thereafter the Area of Applicability is described in this section (see Figure 7 and Figure 12). It is to be mentioned that this stage of the methodology is an iterative process. As represented in Figure 13, the train city remains fixed and the remaining ten cities become the test cities. Among the test cities, one becomes the validation city with each iteration. The steps performed in the train city and the validation city are explained below.

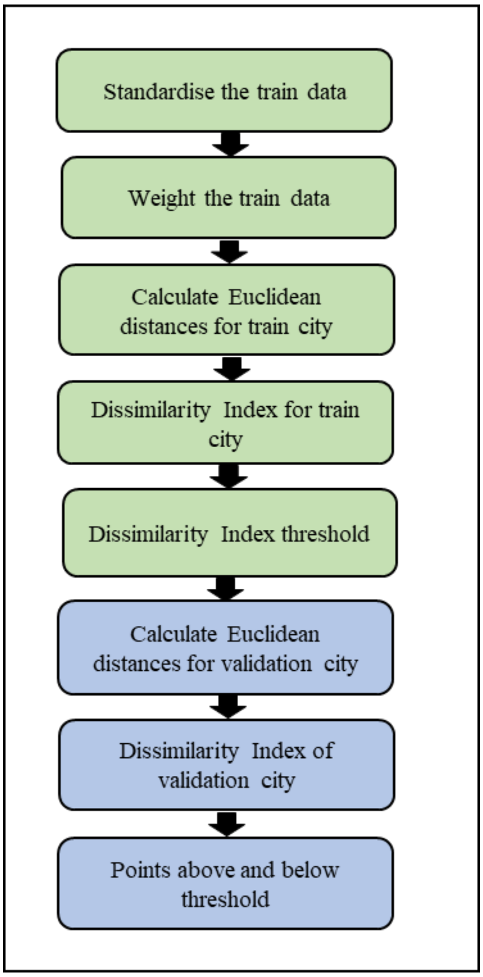

Standardise the train data
Weight the train data
Calculate Euclidean distances for train city
Dissimilarity Index for train city
Dissimilarity Index threshold
Calculate Euclidean distances for validation city
Dissimilarity Index of validation city
Points above and below threshold

Figure 12: Stage three – determining the Area of Applicability (the process involving only the train city is represented in green and the process involving both train and validation cities is represented in blue).

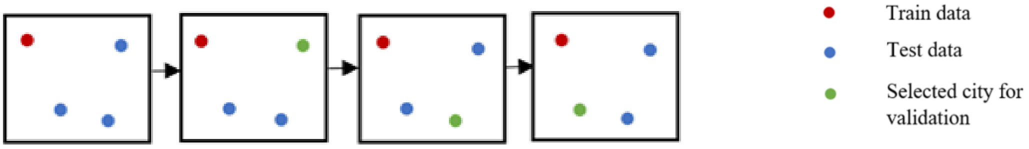


Figure 13: Iterative process of stage three (determining the Area of Applicability)

Steps performed in the train city

To determine the Area of Applicability, a Dissimilarity Index threshold is calculated by using the data of the train city (see Figure 7). For these steps, the train and test data of train city and the band importance values generated earlier (with reference to sub section 4.2.2) are used as input. These processes which are performed solely using the data of train city are explained as follows:

- *Standardise the train data:* Before calculating the Dissimilarity Index, the data is standardised, so that every point has equal importance. For standardisation, the mean value and standard deviation of each band is calculated for the train data. Then, every value in the train data is subtracted from its corresponding band mean, the result of which is divided by standard deviation. It is given by the formula (Meyer, Hanna et al., 2021),

$$X^{s}_{i,j} = (X_{i,j} - \overline{X_{.j}})/\sigma_j \text{ .... (Formula 1)}$$

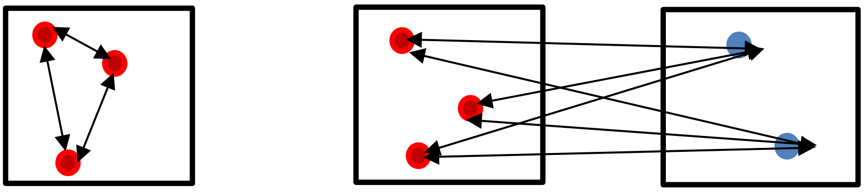
where $X^{s}_{i,j}$ is the standardised value for ith observation corresponding to jth band, X is the train data, i is the observation, j is the corresponding band, $\overline{X_{.j}}$ and σ_j are the mean and standard deviation corresponding to band j respectively.

- *Weight the train data:* The Random Forest model when trained, generates the significances of all the variables, which is similar to ranking of variables according to their roles in the prediction process. The significances of the 16 bands obtained after the model training is used to weight the standardised data. This is done so that all bands are not given equal importance since a few bands may be more significant than the other for predicting the test data and on the Dissimilarity Index. The standardised train data is multiplied with the corresponding band weights. The weighting is done using (Meyer Hanna et al., 2021) the formula,

$$X_{i,j}^{sw} = w_j X_{i,j}^{s} \text{ .... (Formula 2)}$$

where w_j is the significance of band j, $X_{i,j}^{s}$ and $X_{i,j}^{sw}$ are the standardised and weighted values for ith observation corresponding to jth band.

- *Calculate Euclidean distances for train city:* It is to be mentioned that this calculation is performed solely using the train city data. At this step, two sets of pairwise distances are calculated, which are required for the calculation of Dissimilarity Index. In the first set, the pairwise distances between train data points of train city are calculated (see Figure 14 a). The average of the first set of pairwise distances is then computed. In the second set, the pairwise distances between train and test data of the train city are calculated (see Figure 14 b). This is followed by computing the minimum of the second set of pairwise distances.



a) Pairwise Euclidean distances of train data (red)

b) Pairwise Euclidean distances between train (red) and test (blue) data.

Figure 14: Euclidean distance calculations.

- *Dissimilarity Index for train city:* As described in the sub section 2.3 of the chapter on Theoretical Background, the Dissimilarity Index shows how similar the test data is to the train data. It ranges from 0 to ∞, where 0 means complete similarity between the test and train data. It is given by the formula (Meyer Hanna et al., 2021),

$$DI_k = d_k/d \text{ .... (Formula 3)}$$

where d_k is the minimum Euclidean distance for each test data point to train data and d is the average of all pairwise Euclidean distances of the train data.

- *Dissimilarity Index threshold:* The Area of Applicability of the trained model is the area up to which reliable predictions can be made. This is given by a threshold on the Dissimilarity Index. It is calculated by (Meyer Hanna et al., 2021) the formula,

  DI threshold = 75 percentile of DI + 1.5 (IQR_DI) ……… (Formula 4)

  where DI is Dissimilarity Index and IQR_DI is the Inter Quartile Range of the Dissimilarity Index values. According to Meyer Hanna et al., the points whose Dissimilarity Indices are below the threshold are places which show high prediction accuracies and where the trained Random Forest model is successfully transferred.

Steps performed in the validation city

After calculating the threshold using the train city points, the Dissimilarity Index for points of the validation city is calculated (see Figure 7). For these steps, the train data of train city and the data of validation city are used as input. These processes which are performed using the data of train city and validation city are explained as follows:

- *Calculate Euclidean distances for validation city:* At this step, the train points of the train city are considered as train data, and the points of the validation city are considered as the test data. Similar to the Euclidean distance calculation for the train city discussed earlier, this step also involves the calculation of two sets of pairwise Euclidean distances, to compute the Dissimilarity Index for the validation city. Since the first set of pairwise Euclidean distances (pairwise Euclidean distances between train data of train city) has already been calculated as shown in Figure 14 a, the second set of pairwise Euclidean distances is now computed. For this, the pairwise Euclidean distances for each point in test data to train data is calculated as shown in Figure 14 b. This is followed by calculating the minimum of the pairwise Euclidean distances for each point in the validation city.

- *Dissimilarity Index for validation city:* The Dissimilarity Index for the validation city is calculated using Formula 3, by calculating the average of pairwise Euclidean distances of train data of the train city and the minimum of pairwise Euclidean distances between train data of train city and the data of validation city.

- *Points above and below threshold:* Finally, the points which lie above and below the Dissimilarity Index threshold are found. Since in their research study, Meyer Hanna et al., 2021 have concluded that points below threshold are suitable for transferring the model owing to their high prediction accuracies, this step is performed to analyse if the points below threshold are genuinely suitable for transferring a model. By validating the genuineness of the finding by Meyer Hanna et al. 2021, the points below threshold can be confidently used for transferring a model trained in one city to another city.

At this stage, the train data of the train city is standardised, weighted with the corresponding band weights and the Euclidean distance calculations are carried out. From this, the Dissimilarity Index threshold is determined. Then, in the validation city the Euclidean distances, Dissimilarity index are calculated and finally the points above and below threshold is found.

4.2.4 Spatially transferring the trained model

Spatially transferring the trained model means using the trained model to make predictions on new cities. In this step, the trained model (with reference to sub section 4.2.2) is transferred to the remaining ten cities and their accuracy reports are obtained. The following steps are carried out at this stage:

- At this stage, the points of the test city are used as test data.
- Similar to the steps discussed in section 4.2, the data of the test city in the form of csv table is converted to a dataframe (see Figure 11). The X matrix for the model is created with the values from the sixteen columns containing the band information for all points and the column containing the class information is selected as y array.
- The trained Random Forest model is now used to predict the test city. Predictions are made using the trained Random Forest model for points above and below threshold. As discussed in section 4.3, this step is performed to analyse the genuineness of the finding by Meyer Hanna et al., 2021 that points below threshold are suitable for model transferability due to their high prediction accuracies.
- The classification report including precision, recall, f1 score and the kappa coefficient is then generated. This process is repeated for the remaining ten test cities.

Thus, the trained model is transferred to the cities for which the Dissimilarity Indices have already been determined. Finally, the accuracies of the predicted models, in particular the accuracies of the points above and below threshold are determined using the classification report and the kappa coefficient calculated.

4.2.5 Assessing the transferability performances

As shown in Figure 10, this stage involves two steps. In the first step, the prediction accuracies of points above and below threshold are analysed to assess if points below threshold show a higher prediction accuracy than points above threshold, thereby making points below threshold suitable for model transferability (Meyer Hanna et al., 2021). In the second step, the different factors which affect model transferability are discussed. It is also to be mentioned that, from review of literature it is observed that a "high prediction accuracy" for a Random Forest model at new locations where it is transferred implies an overall accuracy between 70% to 82% and a kappa score of >= 0.5 (Juel Anders et al. (2015), Waśniewski Adam et al. (2020), Munoz Sergio et al., 2010, Dharumarajan Subramanian et al., 2020). Considering these values, for this research an overall accuracy of >= 0.7 and a kappa score of >= 0.5 are regarded as "acceptable" values.

Comparing prediction accuracies of points above and below threshold

According to Meyer Hanna et al., 2021, points below threshold show high prediction accuracies when the model is transferred and hence are the places where the model transferability applies. To analyse this finding from their research, the prediction accuracies of points of validation city below and above threshold are compared. This is carried out by assessment of the overall and kappa accuracies. In the classification report generated, its metrics such as accuracy, precision, recall and f1 score are given. Accuracy is calculated by the number of correctly classified labels divided by the total number of labels. However, since it is a generalised metric and includes both the true positives and the true negatives, a measure for the performance of the individual classes is needed. The precision and recall scores are significant for this purpose. Precision is also called as user's accuracy. It is a measure of how many labels predicted as a particular class, are actually that class. Recall also known as producer's accuracy shows the percentage of correctly predicted labels for all labels in that particular class. F1 score is a combination of precision and recall, with the highest and lowest scores being 1 and 0 respectively. It is important for the fact that it shows if the classifier makes correct predictions and not classify the points as belonging to the largest class, in the case of an imbalanced dataset. The weighted average of f1 score is also considered as a measure of the model accuracy (scikit-yb.org., 2016-2019, Narkhede Sarang., 2018). Apart from the classification report metrics, another significant measure of accuracy is the kappa score. It shows the agreement level between ground truth labels and the predicted labels (Scikit-learn.org., 2007-2021). Thus, based on the precision, recall, f1 scores, overall accuracies and the kappa scores generated, the prediction accuracies are assessed.

In this context, based on the recommendation by Meyer Hanna et al., 2021, which is points below threshold have high prediction accuracies and are suitable to transfer the model, a set of questions remain open which are – 1) Does higher number of points below threshold imply successful transferability? 2) Does an acceptable kappa score of > = 0.5 and accuracy of > = 0.70 (with reference to the introduction of section 4.5) mean successful transferability? 3) Does an acceptable kappa score and overall accuracy imply substantiable prediction accuracy, with respect to the individual class F1 scores? An attempt to answer these questions is carried out after an assessment of the prediction accuracies of points below threshold.

Factors affecting model transferability

After extensive research and investigations, certain factors are found that impact the transferability performance of cities, which are as follows:

- *Percentage of points below threshold:* Since Meyer Hanna et al., 2021 recommend that the points below threshold show a higher accuracy than points above and are suitable for model transferability, this is the first criteria that is considered.
- *Overlap coefficient:* The overlap coefficient is an index to analyse the similarities between datasets and ranges between 0 and 1. It is a widely used similarity measure in machine learning and is calculated by the ratio of size of intersection between two datasets to the minimum of the two datasets (Verma Vijay et al., 2020). In this research, the overlap coefficient of two sets of points is observed as factors affecting model transferability. The first set of points includes all points of the validation city and the second set includes the points of the validation city that are below threshold.
- *Histogram intersection:* Similar to overlap coefficient, the histogram intersection is also a similarity index which ranges from 0 to 1. It measures the similarities between images based on the intersection of the histograms (Lee S.M. et al. (2005), Hwang Alex et al. (2009)). For this, the training samples of the cities are converted to arrays, their histograms are generated, normalised and the intersection value is calculated. Similar to the previous step, the histogram intersection of two sets of points is calculated where the first set comprises of all points of the validation city and the second set consists of points of the validation city which are below threshold.
- *Proportion of samples from each class:* For this criterion, the number of samples of validation city belonging to each class namely industrial, residential and small non residen-

tial are considered. It is to be mentioned that for a successful transferability, the proportion of samples belonging to residential class is higher than that of industrial and small non residential classes.

- *Morphological similarities:* As discussed in section 3.4 of the chapter on Literature Review, a model can be successfully transferred when the areas where the model is trained and tested are similar in properties with respect to structure and environment. Based on this recommendation, the morphological similarities between cities are assessed visually and using existing literature. In this assessment, the similarities of structures between the buildings of the cities belonging to the three classes namely industrial, residential and small non residential are investigated.
- *Class separability:* A class separability analysis shows how well separated the classes are considering the spectral values of the bands available and the separability of spectral bands is observed to impact the land use / land cover classification results (Park No-Wook et al., 2016). In this step for each city, based on the training points of the three classes namely industrial, residential and small non residential box plots are generated (see Figure 15). The plots show the distribution of data with respect to their minimum value, maximum value, median, quartile and outliers which provide information about the class separability for each band. At this step an attempt is made to answer two questions, which are 1) Which bands show the highest and least separability? 2) Does substantiable separability between classes imply successful transferability?

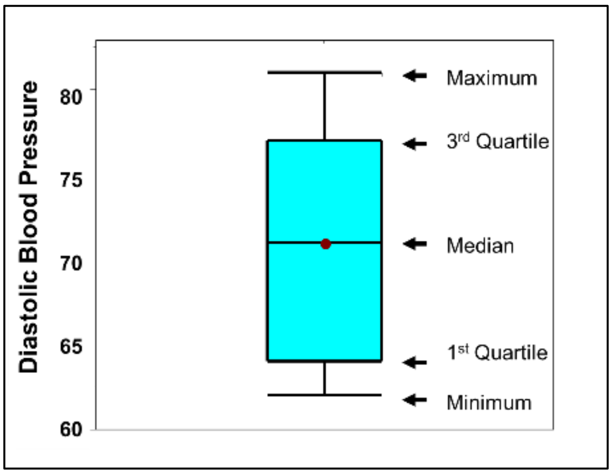


Figure 15: Class separability box plot (Source: Sullivan Lisa, 2016).

Further, it is observed from review of literature that a “high prediction accuracy” for a new location where the Random Forest model is transferred, implies an accuracy of >=0.7 and a kappa score of >=0.5. Considering these values, an “acceptable (substantiable) accuracy and kappa scores” is >=0.7 and >=0.5 respectively; and the “best accuracy and kappa scores” that can be reached is >= 0.8, for a spatially transferred model in this research. Also, as explained in sub section 4.5.1, the F1 score is an accuracy assessment metric that shows if the classifier makes the correct predictions and is especially useful in the case of imbalanced datasets. Though theoretically the maximum F1 score that could be reached is 1, it is observed from review of literature (Shiefer Felix et al. (2020), Yang Wanting et al. (2021)) that a “high F1 score” implies a value of >=0.7. Considering this value, the best F1 score that can be reached for the validation city is >=0.7, and a substantiable score would imply a F1 score of >=0.6 (since a score lower than 0.5 would mean that almost half of the labels have been misclassified, as discussed in sub section 4.5.1).

In this context, the criteria for model transferability are analysed for two different ranges of prediction accuracies for the transferred models – 1) where kappa score and accuracy of 0.8 – 0.9 and individual class F1 scores of >=0.7 (best scores in this research), 2) where kappa score and accuracy of 0.7 – 0.8 and individual class F1 scores of >=0.6. Prediction accuracies below the range of 0.7 are not used for the study since they could not be considered as “acceptable accuracies and kappa scores”. Table 1 shows the different values of the criteria for model transferability discussed above, that are required to yield different prediction accuracies of the transferred models. It is also observed that for varying prediction accuracies of the transferred models, the values of the factors vary.

Criteria	**Minimum value of criteria to yield prediction accuracy**	
	0.8 – 0.9 kappa and individual class F1 scores of > = 0.7	**0.7 – 0.8 kappa and individual class F1 scores of > = 0.6**
Percentage of validation points below threshold	66	52
Overlap coefficient of all points of validation city	0.79	0.83
Histogram intersection of all points of validation city	0.62	0.65
Overlap coefficient of validation points below threshold	0.70	0.70
Histogram intersection of validation points below threshold	0.71	0.78

Percentage of samples from industrial class of validation city	30	26.83
Percentage of samples from residential class of validation city	38	43.11
Percentage of samples from small non residential class of validation city	15	9.40

Table 1: Level of confidence for transferring a model.

Thus, an analysis of the accuracies obtained for the points of the validation city above and below threshold transferred models is carried out. Then the factors affecting model transferability are analysed such as percentage of points below threshold, overlap coefficient of all points of validation city and points of validation city below threshold, histogram intersection of all points of validation city and points of validation city below threshold, proportion of training samples of the validation city belonging to the three classes, morphological similarities and class separability.

4.2.6 Conditional classification

At this final stage, a conditional classification of the cities is carried out. This means that only if the validation city satisfies all the transferability criteria, the model is transferred and the city is classified into three classes which are industrial, residential and small non residential. As seen in Figure 7 there are two steps in this stage which are, 1) to create a binary image showing the Area of Applicability and 2) to create the final classification image.

Create a binary image showing Area of Applicability

- Initially the sixteen band WSF3D raster image of the validation city is converted into an array and stacked (see Figure 16). The array is then reshaped from three dimension to two dimension, to facilitate the classification process. It is to be mentioned that for the following processes, the data from the image, in the form of array is considered as validation city data.
- Similar to the steps performed in sub section 4.2.3, two sets of Euclidean distances are calculated. For the first set, the pairwise Euclidean distances between train data are cal-

culated. And for the second set, the pairwise Euclidean distances between train data of train city and the data of validation city are computed.

- The Dissimilarity Index for the validation city is calculated using Formula 3, by calculating the average of pairwise Euclidean distances of train data of the train city and the minimum of pairwise Euclidean distances between train data of train city and the data of validation city.
- The points which fall below the threshold (calculated in the earlier steps, with reference to sub section 4.2.3) are found.
- A value of "1" is assigned to only those points with their Dissimilarity Index below threshold and "0" for the remaining points.
- The two dimension array is then reshaped to its original raster extent which is in the form of rows x columns x number of bands. The result which is a binary image showing the Area of Applicability where the model can be successfully transferred is displayed.

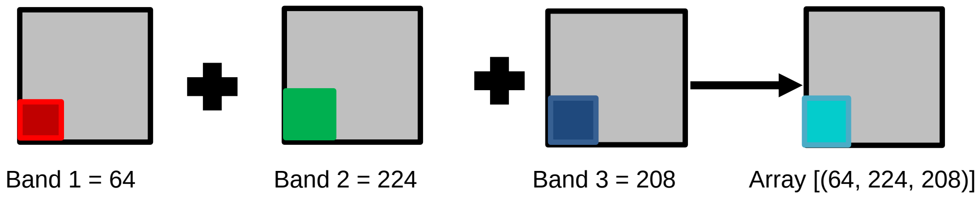


Figure 16: Image converted to an array and stacked.

Create the final classification image

- As seen from the previous steps, the image is read, converted to an array, stacked and reshaped from three dimension to two dimension.
- The image array is then predicted with the Random Forest model trained (with reference to sub section 4.2.2) using the data of the train city
- After prediction, the two-dimension array is reshaped to its original raster extent which is in the form of rows x columns x number of bands. The result which is the final classification image showing three land use classes namely industrial, residential and small non residential is displayed.

5. Results and Discussion

In this chapter, the results achieved using the methodology discussed in chapter four are presented. In section 5.1 the results obtained by investigating the concept by Meyer Hanna et al., 2021, which is the points below threshold are where the model can be successfully transferred, is presented. In section 5.2, the level of confidence for which the transferred model shows a kappa score of 0.8 – 0.9 and individual class F1 scores of >= 0.7 is analysed using all the eleven cities under study. Section 5.3 shows the investigations done using all the cities where the transferred model shows 0.7 – 0.8 kappa score and individual class F1 scores of >= 0.6. Section 5.4 attempts to find any existing relation between model transferability and class separability. In section 5.5, a visual assessment of the predictions made by the transferred models is carried out. In the final section 5.6, the highlights of results obtained are discussed. For the purpose of discussion, the concise results are given in the form of tables and images and the results are presented in the Appendix. Also, in this chapter, the images and diagrams which represent the three classes namely industrial, residential and small non residential buildings are shown in purple, red and light pink respectively in accordance with Corine Land Cover classes and RGB colour codes (European Environment Agency, 2022).

5.1 Prediction accuracies of points below and above threshold

As discussed in section 2.3 of the chapter on Theoretical Background, Hanna et al., 2021 recommend that points below threshold are suitable for model transferability as they show high prediction accuracies when the model is transferred. To investigate the validity of the recommendation by Meyer Hanna et al. (2021), which is if points below threshold are suited for spatially transferring a model, the prediction accuracies of points below and above threshold of all the eleven cities are analysed. Initially, for all the cities the kappa scores obtained for the transferred model using points below threshold, above threshold and all points for the validation cities is assessed (see Figure 17). This is followed by the results of overall accuracies obtained for points below threshold, above threshold and all points of the validation cities (see Figure 18).

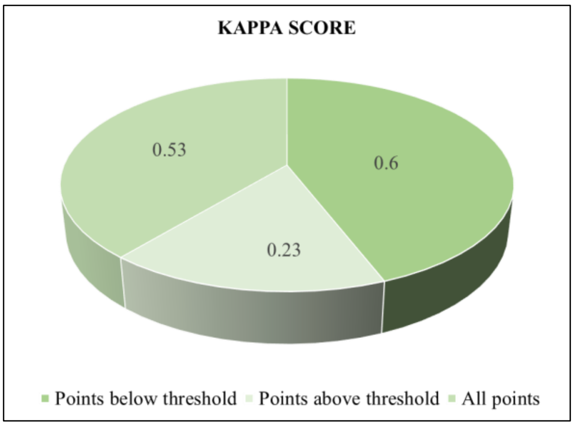


Figure 17: Kappa scores of all cities.

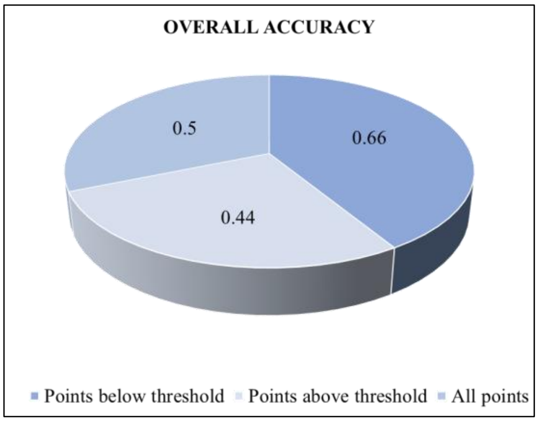


Figure 18: Overall accuracies of all cities.

From the results it is observed that on an average for all the validation cities, the points below threshold show the highest kappa scores (see Figure 17) and overall accuracies (see Figure 18) of 0.6 and 0.66 respectively. This is followed by the kappa scores and overall accuracies of all validation points which are 0.53 and 0.5 respectively. The least kappa scores and overall accuracies for the transferred models are shown by the points above threshold which are 0.23 and 0.44 respectively. The results prove the theory of Meyer Hanna et al., 2021 that the points below threshold of the validation cities show the highest prediction accuracies compared to all validation points and points above threshold, when a trained model is transferred to these validation cities. Hence the forthcoming investigations of this research are based on the points below Dissimilarity Index threshold.

5.2 Level of confidence for a 0.8 – 0.9 kappa score and overall prediction accuracy: case 1

As discussed in the section 4.5.2 of the chapter on Methodology, the best scores in this research are identified as kappa scores and overall accuracies of 0.8 – 0.9 and individual F1 scores of >= 0.7. The minimum criteria that have to be satisfied by a validation city to attain the above mentioned best scores are presented in Table 1. Due to the large extent of cities, in this section the most representative city is selected to explain the results of the analytical process. The city selected for presenting the results in this section is Pune. A set of full results are found in Appendix. In this section, initially the results obtained by investigating these transferability criteria are presented. This is followed by an analysis of the association between model transferability and points below threshold. Finally, the relation between model transferability and morphological similarities is analysed.

5.2.1 Criteria for successful model transferability

As discussed in the chapter four on Methodology, initially the city (Pune) is trained and its threshold is obtained as 12.56. Then the Dissimilarity Indices for the individual cities are found and the points below and above threshold for all the remaining ten cities is obtained. As described in sub section 4.3.2 of the chapter on Methodology, this is an iterative process where each of the ten test cities is selected for validation in one iteration. In this sub section, the results obtained for all validation cities, when a Random Forest model trained using the city of Pune is transferred to these cities, are presented. The investigations performed to analyse which validation cities fulfil the minimum transferability criteria in order to attain kappa scores and overall accuracies of 0.8 – 0.9 and individual F1 scores of >= 0.7, thereby making them suitable for model transferability are as follows:

- The first criterion for a successful transferability and obtaining a kappa and overall accuracy of 0.8 – 0.9 for the transferred model and >= 0.7 individual class F1 scores is that a minimum of 66% of points of the test city have to lie below threshold (with reference to sub section 4.5.2). As seen in Figure 19, it could be observed that eight cities namely Ahmedabad (78.67%), Hindupur (81.56%), Hyderabad (73.56%), Jaipur (81.78%), Kanpur (66.44%), Malegaon (69.56%), Parbhani (67.33%) and Singrauli (78.67%) satisfy this criterion. These cities are filtered and investigated if they satisfy the next criterion.

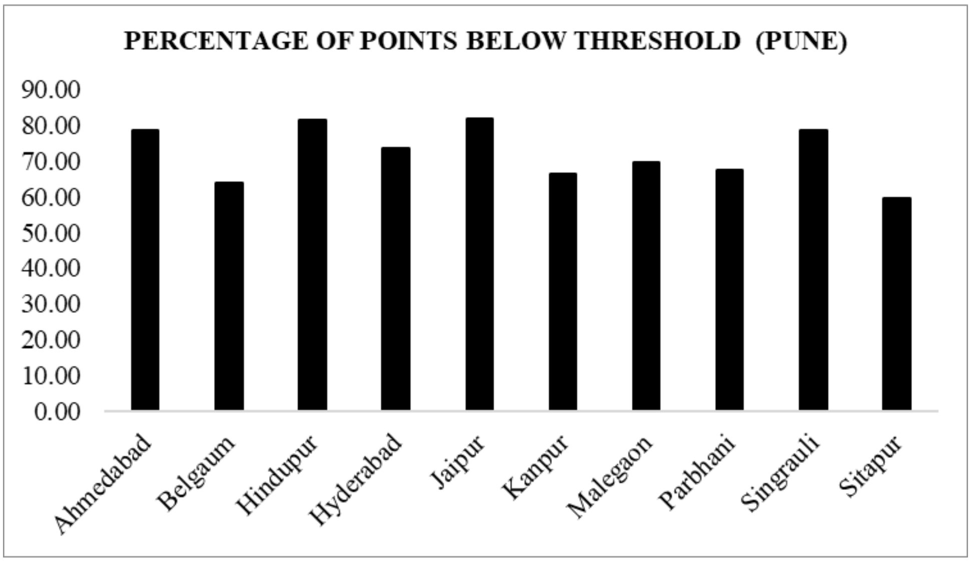


Figure 19: Percentage of points below threshold: case 1.

- The second transferability criterion is that for all points of the validation city, the overlap coefficient has to be at least 0.79 (with reference to sub section 4.5.2). Since from the previous step it is observed that eight cities namely Ahmedabad, Hindupur, Hyderabad, Jaipur, Kanpur, Malegaon, Parbhani and Singrauli satisfy the first criterion for spatial transferability, these cities are only considered at this step. Among these cities, it is observed that (see Figure 20) five cities namely Ahmedabad (0.9), Hyderabad (1), Jaipur (0.92), Kanpur (0.84) and Singrauli (0.79) satisfy this criterion. These cities are then filtered and the third transferability criterion is applied on them.

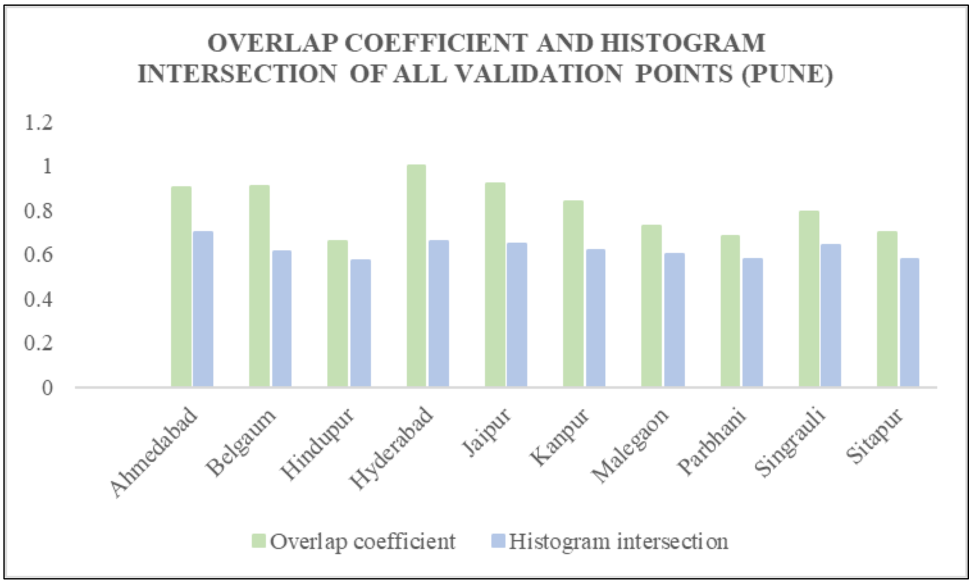


Figure 20: Overlap coefficient and histogram intersection of all points: case 1.

- The third transferability criterion is that for all points of the validation city, the histogram intersection has to be at least 0.62. From the results of the previous step, the five cities such as Ahmedabad, Hyderabad, Jaipur, Kanpur and Singrauli are only used for this analysis. As seen in Figure 20, it could be observed that the same five cities namely Ahmedabad (0.7), Hyderabad (0.66), Jaipur (0.65), Kanpur (0.62) and Singrauli (0.64) fulfil this transferability condition of histogram intersection for all validation points to be a minimum of 0.62.
- The fourth transferability criterion for successful transferability of a model and obtaining a kappa and overall accuracy of 0.8 – 0.9 for the transferred model and >= 0.7 individual class F1 scores is that for points of the validation city below threshold, the overlap coefficient has to be at least 0.70. The five cities such as Ahmedabad, Hyderabad, Jaipur, Kanpur and Singrauli which have satisfied the first three transferability criteria are used for this analysis. It is observed that (see Figure 21) among these cities, all of them namely Ahmedabad (0.8), Hyderabad (0.89), Jaipur (0.85), Kanpur (0.70) and Singrauli (0.73) satisfy this condition. As a next step, the fifth transferability criterion is

investigated on these five cities which are observed to satisfy the first four conditions for successful transferability.

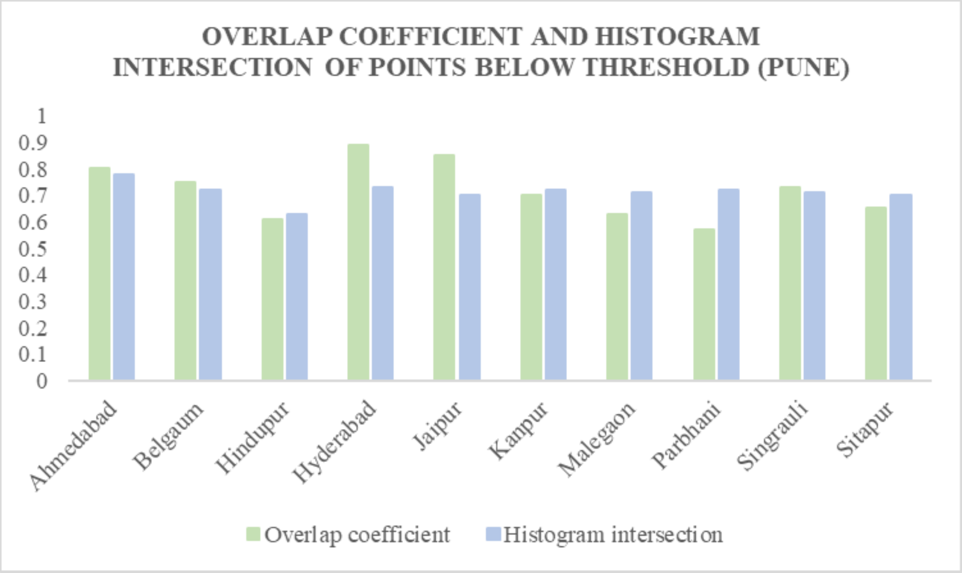


Figure 21: Overlap coefficient and histogram intersection of points below threshold: case 1.

- The fifth transferability criterion is that for validation points below the threshold, the histogram intersection has to be a minimum of 0.71. Among the five cities which have satisfied the first four conditions such as Ahmedabad, Hyderabad, Jaipur, Kanpur and Singrauli, it is observed that (see Figure 21), only four cities namely Ahmedabad (0.78), Hyderabad (0.73), Kanpur (0.72) and Singrauli (0.71) satisfy this transferability condition of histogram intersection for points below threshold of the validation city.
- The sixth criterion for a successful model transferability is that, out of the total validation points a minimum of 30% of the samples have to belong to the industrial class. As seen in Figure 22, it is observed that the same four cities which have satisfied the first five criteria also fulfil this transferability condition. The cities which are observed to fulfil all the six criteria are Ahmedabad (34.18%), Hyderabad (31.72%), Kanpur (39.46%) and Singrauli (30.79%).

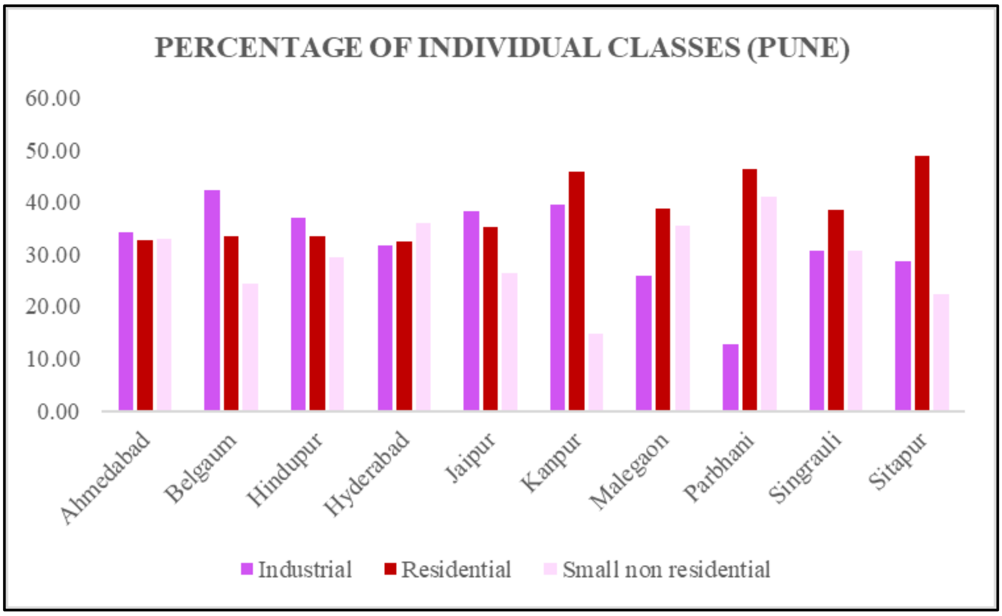


Figure 22: Proportion of training samples from individual classes: case 1.

- The seventh factor that impacts a successful model transferability and necessitates for obtaining a kappa and overall accuracy of 0.8 – 0.9 for the transferred model and >= 0.7 individual class F1 scores is that, out of the total validation points a minimum of 38% of the samples have to belong to the residential class. The four cities such as Ahmedabad, Hyderabad, Kanpur and Singrauli which have satisfied the first six transferability criteria are used for this analysis. It is observed that (see Figure 22) among these cities, only two cities namely Kanpur (44.82%) and Singrauli (38.42%) are observed to satisfy this condition.
- The eighth criterion for a successful model transferability is out of the total validation points, a minimum of 15% of the samples have to small non residential class. It is observed (see Figure 22) that the two cities which are filtered in the previous step are also found to satisfy this criterion, which are Kanpur (15.72%) and Singrauli (30.79%). Thus, it is observed that Kanpur and Singrauli fulfil all the conditions required for successfully transferring the model and to obtain a kappa and overall accuracy of 0.8 – 0.9 for the transferred model and >= 0.7 individual class F1 scores for the transferred model. Hence it could be stated with confidence that a model trained in Pune can be successfully transferred to Kanpur and Singrauli.

5.2.2 Model transferability and points below threshold

Having obtained the results for prediction accuracies of points below threshold (with reference to sections 5.1) and model transferability (with reference to section 5.2), analysis is carried out in this section to answer a set of questions. The questions as discussed in the sub section 4.5.1 of the chapter on Methodology are 1) Does a high number of points below threshold imply successful model transferability? 2) Does an acceptable kappa score and overall accuracy of > = 0.5 and > = 0.70 respectively (with reference to section 4.5 of the chapter on Methodology) imply successful transferability? and 3) Do acceptable prediction accuracies imply substantiable prediction accuracies, with respect to individual class F1 scores? In this context, the analysis of the questions mentioned above are as follows:

- **Does higher number of points below threshold means that the model can be successfully transferred?** To answer this question, the number of points below threshold for all validation cities are examined to find if there is an association between the number of points below threshold and model transferability. It is observed that (see Figure 23) Jaipur (368) shows the highest number of points below threshold, followed by Hindupur (367), Singrauli (354) and Ahmedabad (354). The cities which least number of points below threshold are Sitapur (268), Belgaum (288) and Kanpur (299). Thus, it could be observed that the cities such as Singrauli and Kanpur which are most suitable for transferring the model, have one of the highest and least number of points below threshold respectively. Hence, it could be stated that higher number of points below threshold does not necessarily mean successful transferability.

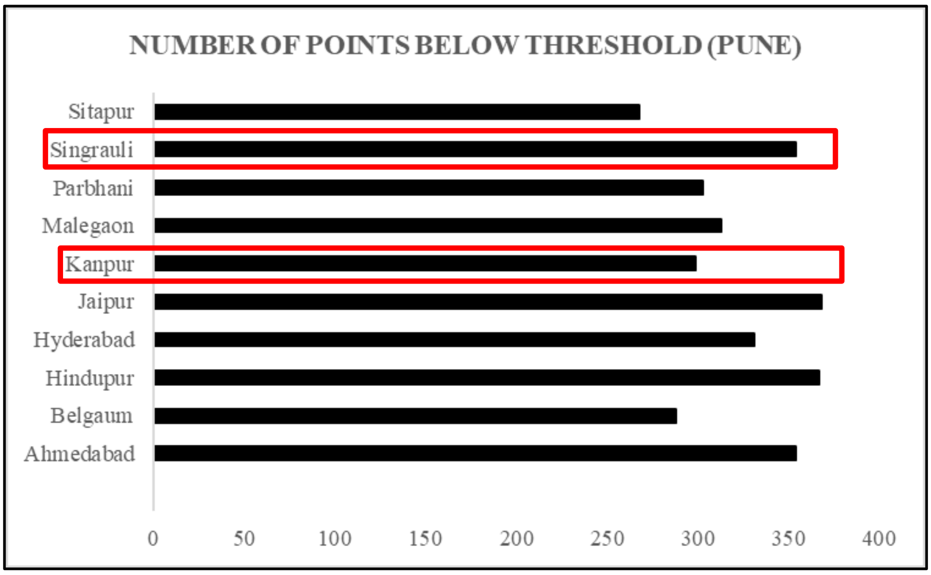


Figure 23: Number of points below threshold: case 1 (cities suitable for model transferability are represented by a red border).

- **Does an acceptable kappa score of > = 0.5 and accuracy of > = 0.70 imply successful model transferability?** To answer this question, the prediction accuracies of points below threshold for all validation cities are examined to find if there is an association between acceptable prediction accuracies and model transferability. As seen from the Figure 24, it is observed that Belgaum (kappa score of 0.62 and accuracy of 0.72), Hindupur (kappa score of 0.64 and accuracy of 0.73), Kanpur (kappa score of 0.80 and accuracy of 0.86), Malegaon (kappa score of 0.59 and accuracy of 0.70) and Singrauli (kappa score of 0.93 and accuracy of 0.94). From the results obtained in sub section 5.2.1, it is known that only Kanpur and Singrauli are suitable for transferring a model trained using Pune. But from the results in this section, it is observed that Belgaum, Hindupur, Kanpur, Malegaon and Singrauli yield an acceptable kappa and overall accuracy of >=0.5 and >=0.7 respectively. Hence, it could be stated that an acceptable kappa score and overall accuracy does not necessarily imply that the model can be successfully transferred.

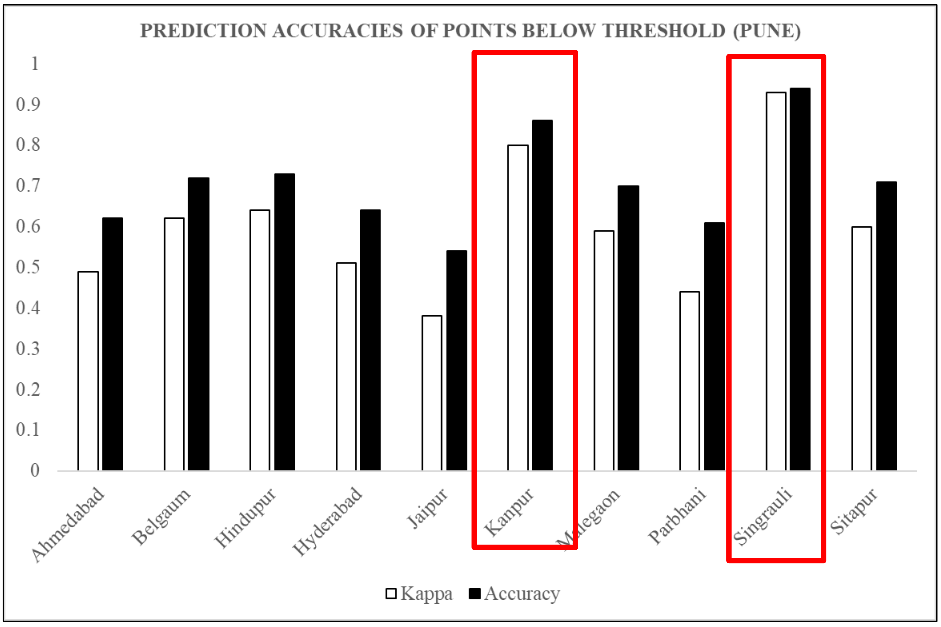


Figure 24: Prediction accuracies of points below threshold: case 1 (cities suitable for model transferability are represented by a red border).

- **Does an acceptable kappa score and overall accuracy imply substantiable individual class F1 scores?** It is to be mentioned that as discussed in the sub section 4.5.2 of the chapter on Methodology, a F1 score of >= 0.6 is considered as substantiable score in this research. For this analysis, the individual F1 scores of the three classes namely industrial, residential and small non residential are examined. To answer this question, the cities such as Belgaum, Hindupur, Kanpur, Malegaon and Singrauli are used, which are observed to yield acceptable kappa and overall accuracies in the previous step. As seen from the Figure 25 and Table 2, it is observed that Belgaum has F1 scores of 0.6, 0.39 and 0.64 respectively for the industrial, residential and small non residential classes, where the residential class is observed to show low F1 score of 0.39. Hindupur has F1 scores of 0.69, 0.69 and 0.43 respectively for the three classes, where a low score of 0.43 is observed for small non residential class. Kanpur has F1 scores of 0.71, 0.83 and 0.76 respectively for all the three classes. Malegaon has F1 scores of 0, 0.62 and 0.71 in all three classes, which implies that in the industrial class no predicted pixels are identified thereby indicating an almost complete dissimilarity in the structures and properties of Pune and Malegaon. Singrauli shows F1 scores of 0.97, 0.89 and 0.89 respectively for all the three classes. Thus, it could be observed that though Belgaum, Hindupur, Kanpur, Malegaon and Singrauli yield an acceptable kappa and overall accuracy, the individual F1 scores of Belgaum, Hindupur and Malegaon are observed to be low for the residential, small non residential and industrial classes respectively. However, the cities which are observed to be the most suitable for model transferability such as Kanpur and Singrauli are observed to yield high F1 scores.

Hence, it could be stated that an acceptable kappa and overall accuracy of >=0.5 and >=0.7 respectively does not necessarily mean substantiable individual F1 scores of >=0.7. And also, for a successful transferability, it is necessary to evaluate the individual class accuracies besides analysing the overall accuracies and kappa scores. In addition, the cities which are most suitable for model transferability are observed to have not only acceptable kappa and overall accuracies but also substantiable individual class F1 scores.

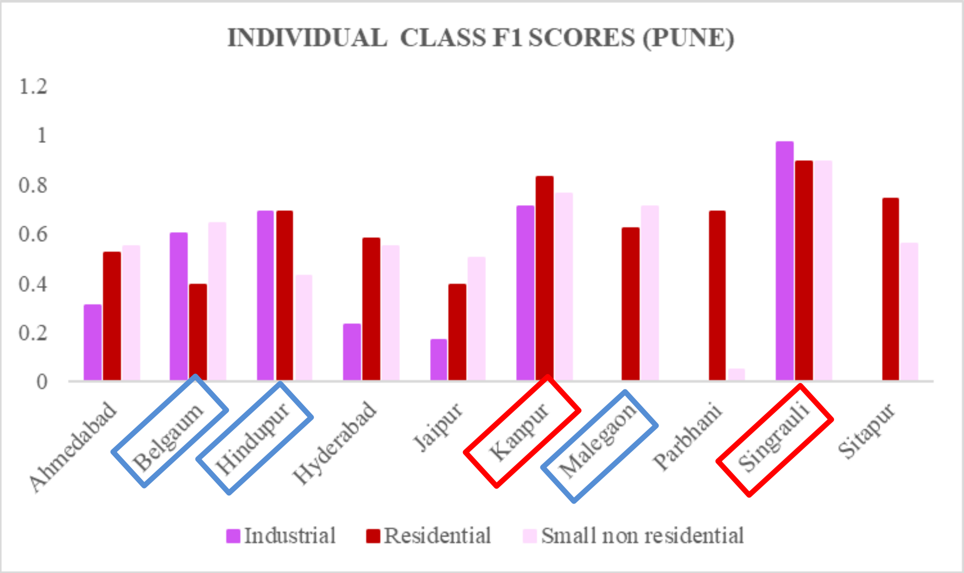


Figure 25: Individual class F1 scores: case 1 (the names of the cities which show acceptable kappa and overall accuracies are represented by a blue border and the names of the cities which are suitable for model transferability are represented by a red border).

City	Industrial (class 1)	Residential (class 2)	Small non residential (class 3)
Belgaum	0.6	0.39	0.64
Hindupur	0.69	0.69	0.43
Kanpur	0.71	0.83	0.76
Malegaon	0	0.62	0.71
Singrauli	0.97	0.89	0.89

Table 2: Individual class F1 scores of selected validation cities: case 1

5.2.3 Morphological similarities between transferable cities

It is observed that similarity of properties of regions under study is essential for successful spatial transferability (Wasniewski Adam et al., 2020, Dharumarajan S. et al., 2017), as discussed in sub section 4.5.2 of the chapter on Methodology. Hence in this section the similarities between cities suited for model transferability are assessed to find any association between the structural conditions of buildings in a city and their corresponding transferability. It has been found from the analysis in sub section 5.2.1 that a model trained in Pune can be successfully transferred to Kanpur and Singrauli (see Figure 26) and a kappa and overall accuracy of 0.8 – 0.9 and individual class F1 scores of >=0.7 can be obtained. Therefore, in this section the morphological similarities between the buildings of these cities are analysed, with respect to the industrial, residential and small non residential classes which are as follows:

- Pune is a major industrial area, with pharmaceutical, mechanical and automotive industries such as Tata motors. Among the small non residential buildings, it has a number of software companies in Information Technology parks. Among residential buildings, it has tall apartments as well as informal settlements such as slums. (Sarode., 2016). It is assessed visually that the city has big industries which large areas.
- Kanpur is famous for its industries such as leather, garments, industrial machines etc. and is a major industrial region (Government of Uttar Pradesh, 2022) It also has many small-scale industries (MSME Development Institute of Kanpur, 2022). The planned residential buildings are well spaced and found around the core of the city, whereas the slums are found in the unplanned areas and found in the fringes of the city, with limited spacing between them (Ahlawat Joginder., 2017). Through visual assessment it is observed that the city has large industries such as India Oil, Small Arms Factory etc.
- Singrauli has coal mining industries and coal fields surrounding the region. It also has many power plants and is called as India’s energy capital (Government of Madhya Pradesh – District Singrauli, 2022). From visual assessment it is observed that a few portions are concentrated with industrial areas mainly in the eastern part of the city. Also, different types of well-spaced residential areas are found from high rise apartments to low lying houses including slums where around 16% of the urban population resides (Census of India, 2011).

Through visual assessment and literature review it is observed that Pune and Kanpur show similarities in the industrial and small non residential class buildings, especially in the spacing of the industrial buildings. And striking similarities are observed between Pune and Singrauli mainly in the industrial buildings.

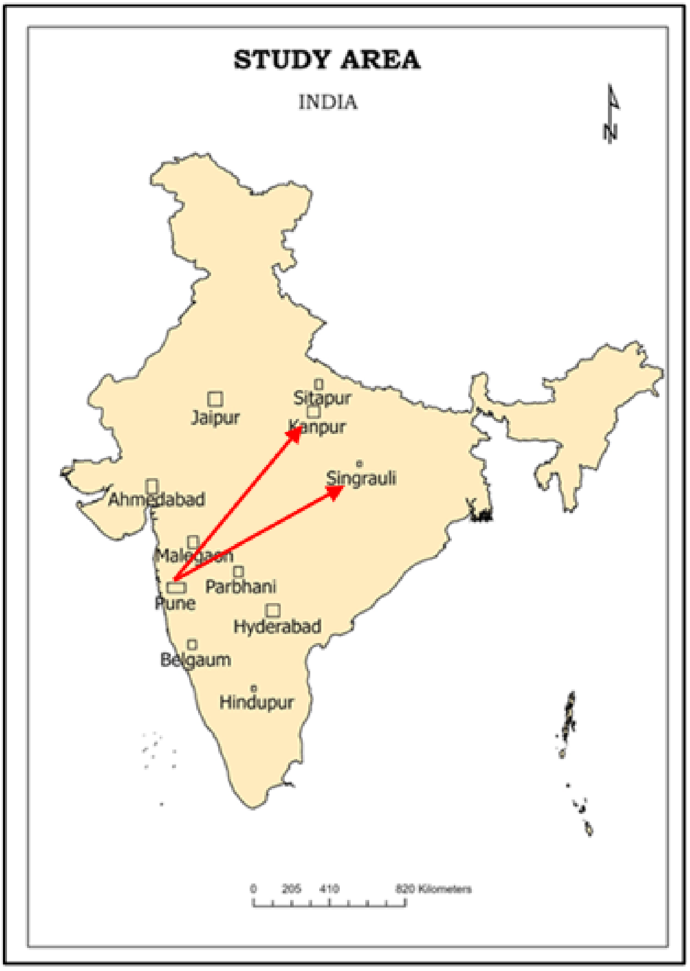


Figure 26: Model transferability: case 1 (the red lines represent that a model trained in Pune can be transferred to Kanpur and Singrauli).

5.3 Level of confidence for a 0.7 – 0.8 kappa and overall prediction accuracy: case 2

As discussed in the section 4.5.2 of the chapter on Methodology, the second best scores in this research are identified as kappa scores and overall accuracies of 0.7 – 0.8 and individual F1 scores of >= 0.6. The minimum criteria that have to be satisfied by a validation city to attain the above mentioned best scores are presented in Table 1. Due to the large extent of cities, in this section the most representative city is selected to explain the results of the analytical process. The city selected for presenting the results in this section is Belgaum. A set of full results are found in Appendix. In this section, initially the results obtained by investigating these transferability criteria are presented. This is followed by an analysis of the association between model transferability and points below threshold. Finally, the relation between model transferability and morphological similarities is analysed.

5.3.1 Criteria for successful model transferability

As discussed in the chapter four on Methodology, initially the city (Belgaum) is trained and its threshold is obtained as 9.50. Then the Dissimilarity Indices for the individual cities are found and the points below and above threshold for all the remaining ten cities is obtained. As described in sub section 4.3.2 of the chapter on Methodology, this is an iterative process where each of the ten test cities is selected for validation in one iteration. In this sub section, the results obtained for all validation cities, when a Random Forest model trained using the city of Belgaum is transferred to these cities, are presented. The investigations performed to analyse which validation cities fulfil the minimum transferability criteria in order to attain kappa scores and overall accuracies of 0.7 – 0.8 and individual F1 scores of >= 0.6, thereby making them suitable for model transferability are as follows:

- The first criterion for a successful transferability and obtaining a kappa and overall accuracy of 0.7 – 0.8 for the transferred model and >= 0.6 individual class F1 scores is that a minimum of 52% of points of the test city have to lie below threshold (with reference to sub section 4.5.2). As seen in Figure 27, it could be observed that seven cities namely Ahmedabad (52.67%), Hindupur (79.33%), Hyderabad (67.78%), Jaipur (58%), Kanpur (54.67%), Parbhani (76.44%) and Sitapur (54.44%) satisfy this criterion. These cities are filtered and investigated if they satisfy the next criterion.

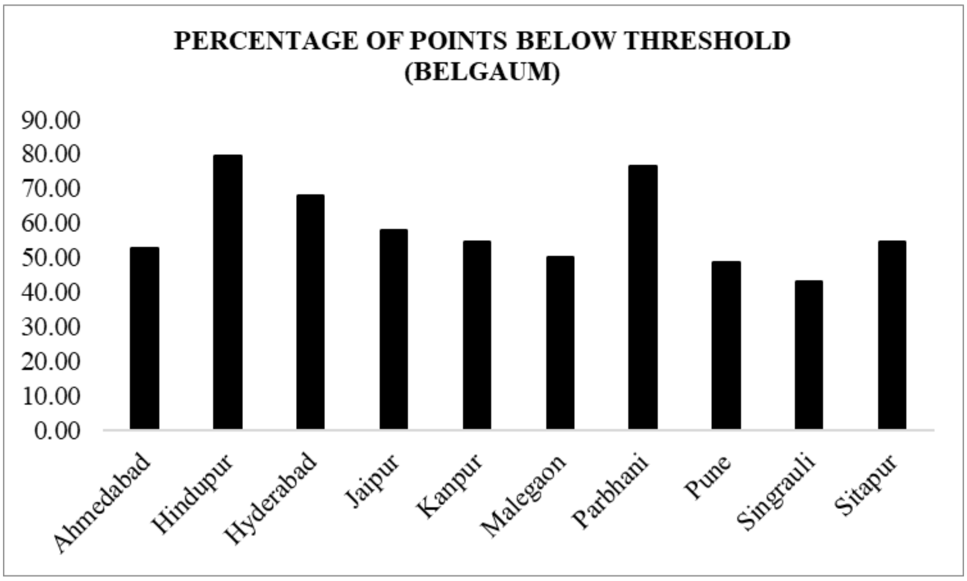


Figure 27: Percentage of points below threshold: case 2.

- The second transferability criterion is that for all points of the validation city, the overlap coefficient has to be at least 0.83 (with reference to sub section 4.5.2). Since from the previous step it is observed that seven cities namely Ahmedabad, Hindupur, Hyderabad, Jaipur, Kanpur, Parbhani and Sitapur satisfy the first criterion for spatial transferability, these cities are only considered at this step. Among these cities, it is observed that (see Figure 28) six cities namely Ahmedabad (0.95), Hindupur (0.85), Hyderabad (0.99), Jaipur (0.95), Kanpur (0.93) and Sitapur (0.85) satisfy this criterion. These cities are then filtered and the third transferability criterion is applied on them.

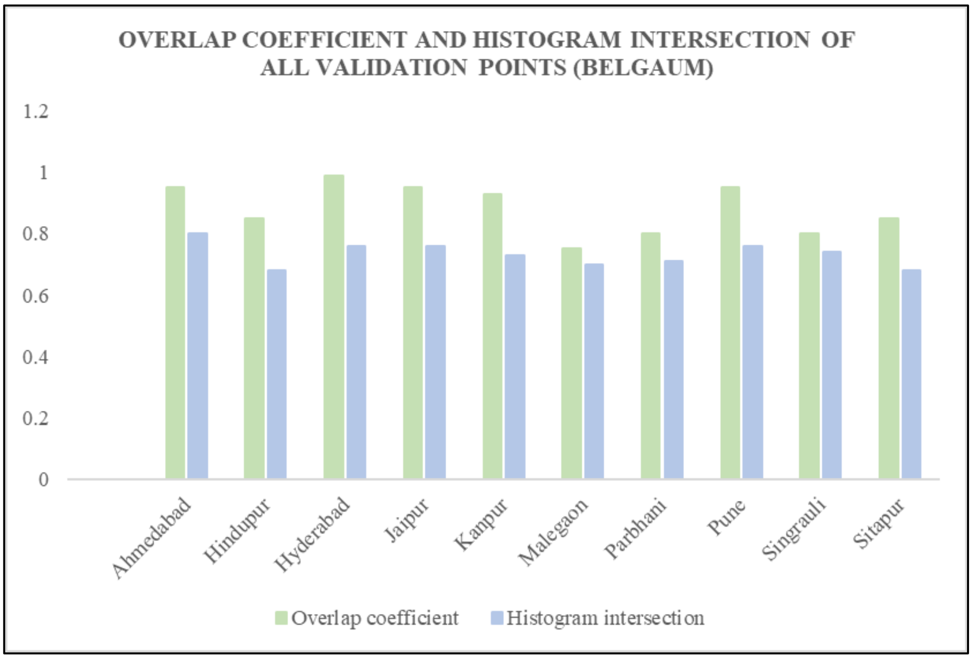


Figure 28: Overlap coefficient and histogram intersection of all points: case 2.

- The third transferability criterion is that for all points of the validation city, the histogram intersection has to be at least 0.65. From the results of the previous step, the six cities such as Ahmedabad, Hindupur, Hyderabad, Jaipur, Kanpur and Sitapur are only used for this analysis. As seen in Figure 28, it could be observed that the same six cities namely Ahmedabad (0.8), Hindupur (0.68), Hyderabad (0.76), Jaipur (0.76), Kanpur (0.73) and Sitapur (0.68) fulfil this transferability condition of histogram intersection for all validation points to be a minimum of 0.65.
- The fourth transferability criterion for successful transferability of a model and obtaining a kappa and overall accuracy of 0.7 – 0.8 for the transferred model and >= 0.6 individual class F1 scores is that for points of the validation city below threshold, the overlap coefficient has to be at least 0.70. The six cities such as Ahmedabad, Hindupur, Hyderabad, Jaipur, Kanpur and Sitapur which have satisfied the first three transferability criteria are used for this analysis. It is observed that (see Figure 29) among these cities, all of them namely Ahmedabad (0.8), Hindupur (0.8), Hyderabad (0.83), Jaipur (0.81), Kanpur (0.79) and Sitapur (0.76) satisfy this condition. As a next step, the fifth transferability criterion is investigated on these six cities which are observed to satisfy the first four conditions for successful transferability.

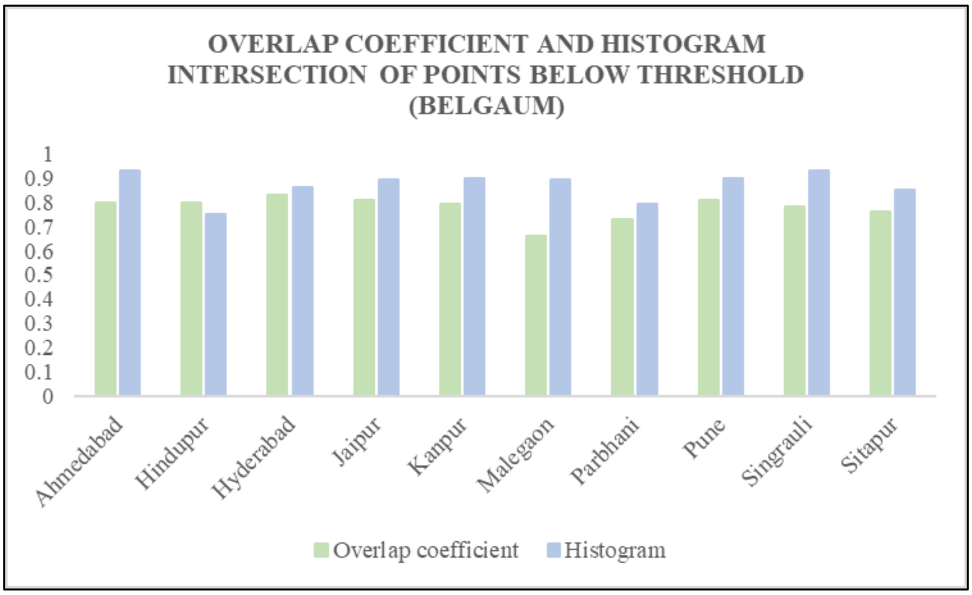


Figure 29: Overlap coefficient and histogram intersection of points below threshold: case 2.

- The fifth transferability criterion is that for validation points below the threshold, the histogram intersection has to be a minimum of 0.78. Among the six cities which have satisfied the first four conditions such as Ahmedabad, Hindupur, Hyderabad, Jaipur, Kanpur and Sitapur it is observed that (see Figure 29), only four cities namely Ahmedabad (0.93), Hyderabad (0.86), Jaipur (0.81) and Kanpur (0.79) satisfy this transferability condition of histogram intersection for points below threshold of the validation city.
- The sixth criterion for a successful model transferability is that, out of the total validation points a minimum of 26.83% of the samples have to belong to the industrial class. As seen in Figure 30, it is observed that among the four cities such as Ahmedabad, Hyderabad, Jaipur and Kanpur, which have satisfied the first five criteria, only three cities namely Hyderabad (36.39%), Jaipur (34.48%) and Kanpur (26.83%) are observed to fulfil this criterion for model transferability.

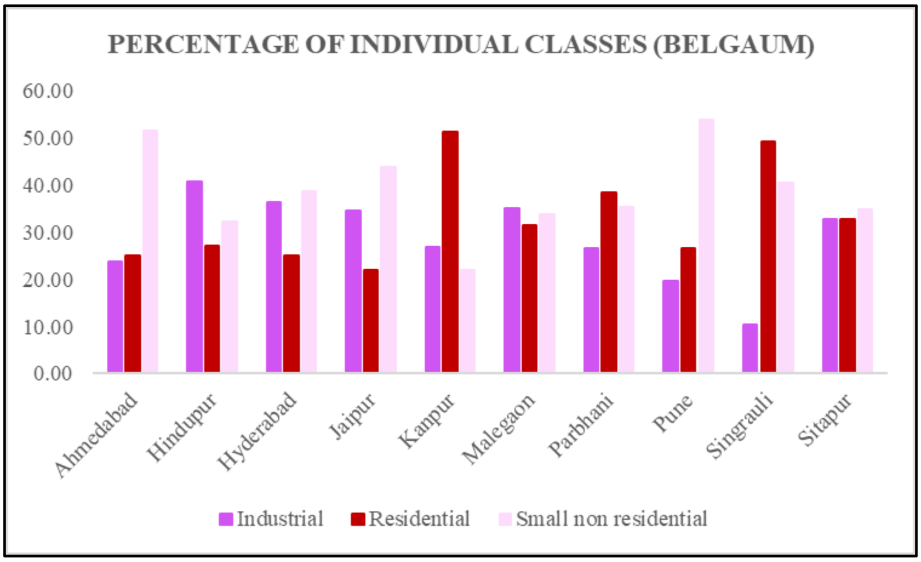


Figure 30: Proportion of training samples from individual classes: case 2.

- The seventh factor that impacts a successful model transferability and necessitates for obtaining a kappa and overall accuracy of 0.7 – 0.8 for the transferred model and >= 0.6 individual class F1 scores is that, out of the total validation points a minimum of 43.11% of the samples have to belong to the residential class. The three cities such as Ahmedabad, Hyderabad, Jaipur and Kanpur which have satisfied the first six transferability criteria are used for this analysis. It is observed that (see Figure 30) among these cities, only Kanpur (51.22%) is observed to satisfy this condition.

- The eighth criterion for a successful model transferability is out of the total validation points, a minimum of 9.40% of the samples have to small non residential class. It is observed (see Figure 30) that city which is filtered in the previous step is also found to satisfy this criterion, which is Kanpur (21.95%). Thus, it is observed that Kanpur fulfils all the conditions required for successfully transferring the model and to obtain a kappa and overall accuracy of 0.7 – 0.8 for the transferred model and >= 0.6 individual class F1 scores for the transferred model. Hence it could be stated with confidence that a model trained in Belgaum can be successfully transferred to Kanpur.

5.3.2 Model transferability and points below threshold

Having obtained the results for prediction accuracies of points below threshold (with reference to sections 5.1) and model transferability (with reference to section 5.2), analysis is carried out in this section to answer a set of questions. The questions as discussed in the sub section 4.5.1 of the chapter on Methodology are 1) Does a high number of points below threshold imply successful model transferability? 2) Does an acceptable kappa score and overall accuracy of > = 0.5 and > = 0.70 respectively (with reference to section 4.5 of the chapter on Methodology) imply successful transferability? and 3) Do acceptable prediction accuracies imply substantiable prediction accuracies, with respect to individual class F1 scores? In this context, the analysis of the questions mentioned above are as follows:

- **Does higher number of points below threshold means that the model can be successfully transferred?** To answer this question, the number of points below threshold for all validation cities are examined to find if there is an association between the number of points below threshold and model transferability. It is observed that (see Figure 31) Hindupur (357) shows the highest number of points below threshold, followed by Parbhani (344) and Hyderabad (305). The cities which least number of points below threshold are Singrauli (193), Pune (218), Malegaon (226) and Kanpur (246). Thus, it could be observed that Kanpur which is the most suitable city for transferring the model, has one of the least number of points below threshold. Hence, it could be stated that higher number of points below threshold does not necessarily mean successful transferability.

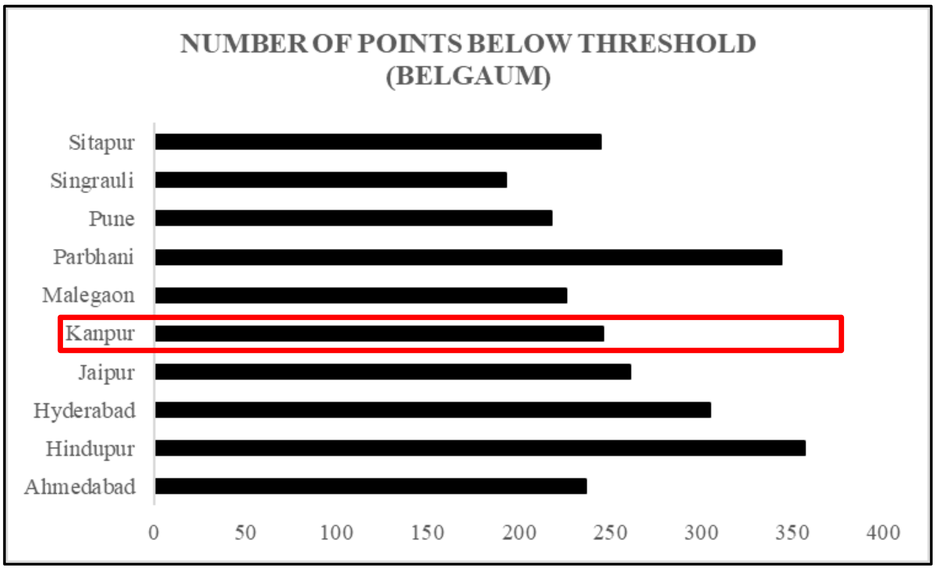


Figure 31: Number of points below threshold: case 2 (cities suitable for model transferability are represented by a red border).

- **Does an acceptable kappa score of > = 0.5 and accuracy of > = 0.70 imply successful model transferability?** To answer this question, the prediction accuracies of points below threshold for all validation cities are examined to find if there is an association between acceptable prediction accuracies and model transferability. As seen from the Figure 32, it is observed that Jaipur (kappa score of 0.62 and accuracy of 0.71), Kanpur (kappa score of 0.80 and accuracy of 0.86), Malegaon (kappa score of 0.64 and accuracy of 0.74), Pune (kappa score of 0.6 and accuracy of 0.72) and Singrauli (kappa score of 0.69 and accuracy of 0.78). From the results obtained in sub section 5.3.1, it is known that only Kanpur is suitable for transferring a model trained using Pune. But from the results in this section, it is observed that Jaipur, Kanpur, Malegaon, Pune and Singrauli yield an acceptable kappa and overall accuracy of >=0.5 and >=0.7 respectively. Hence, it could be stated that an acceptable kappa score and overall accuracy does not necessarily imply that the model can be successfully transferred.

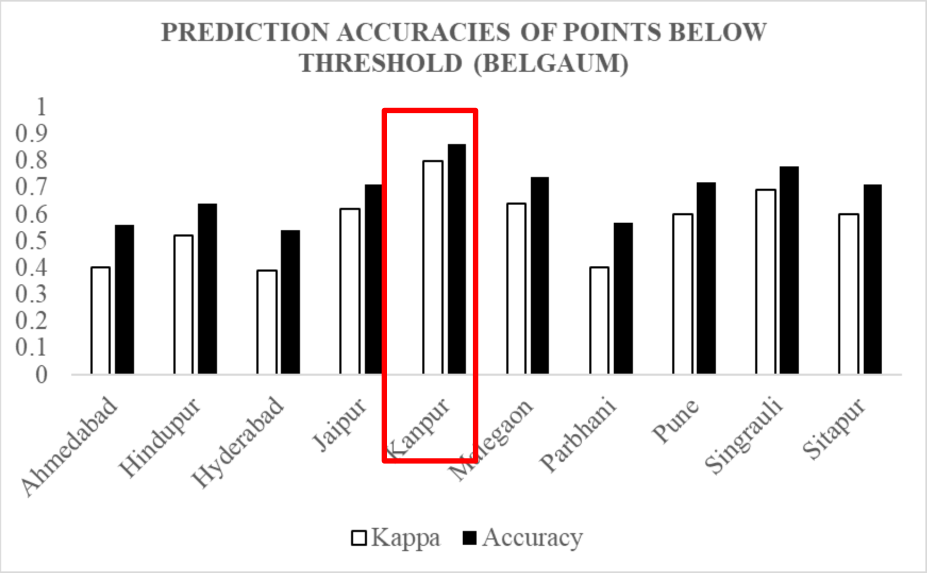


Figure 32: Prediction accuracies of points below threshold: case 2 (cities suitable for model transferability are represented by a red border).

- **Does an acceptable kappa score and overall accuracy imply substantiable individual class F1 scores?** It is to be mentioned that as discussed in the sub section 4.5.2 of the chapter on Methodology, a F1 score of >= 0.6 is considered as substantiable score in this research. For this analysis, the individual F1 scores of the three classes namely industrial, residential and small non residential are examined. To answer this question, the cities such as Jaipur, Kanpur, Malegaon, Pune and Singrauli are used, which are observed to yield acceptable kappa and overall accuracies in the previous step. As seen from the Figure 33 and Table 3, it is observed that Jaipur has F1 scores of 0.57, 0.5 and 0.59 respectively for the industrial, residential and small non residential classes. These scores are less than 0.6 and are not considered substantiable (with reference to the sub section 4.5.2). The next city Kanpur has F1 scores of 0.64, 0.86 and 0.69 respectively for the three classes, where all the scores are observed to be substantiable. For the next city Malegaon, F1 scores of 0.56, 0.56 and 0.58 respectively for the three classes are observed. This also is not considered substantiable since the scores are less than 0.6. Similarly, Pune has F1 scores of 0.36, 0.35 and 0.69 in all three classes, which shows that low accuracies are observed in the industrial and residential classes. Finally, Singrauli shows F1 scores of 0.38, 0.88 and 0.24 respectively for all the three classes, where low scores are observed in the small non residential and residential classes. Thus, it could be observed that though Jaipur, Kanpur, Malegaon, Pune and Singrauli yield an acceptable kappa and overall accuracy, the individual F1 scores of all cities except Kanpur are observed to be low especially for the small non residential class. However, the city which is observed to be the most suitable for model transferability namely Kanpur is observed to yield substantiable and the highest F1 scores among all validation cities.

Hence, it could be stated that an acceptable kappa and overall accuracy of >=0.5 and >=0.7 respectively does not necessarily mean substantiable individual F1 scores of >=0.7. And also, for a successful transferability, it is necessary to evaluate the individual class accuracies besides analysing the overall accuracies and kappa scores. In addition, the cities which are most suitable for model transferability are observed to have not only acceptable kappa and overall accuracies but also substantiable individual class F1 scores.

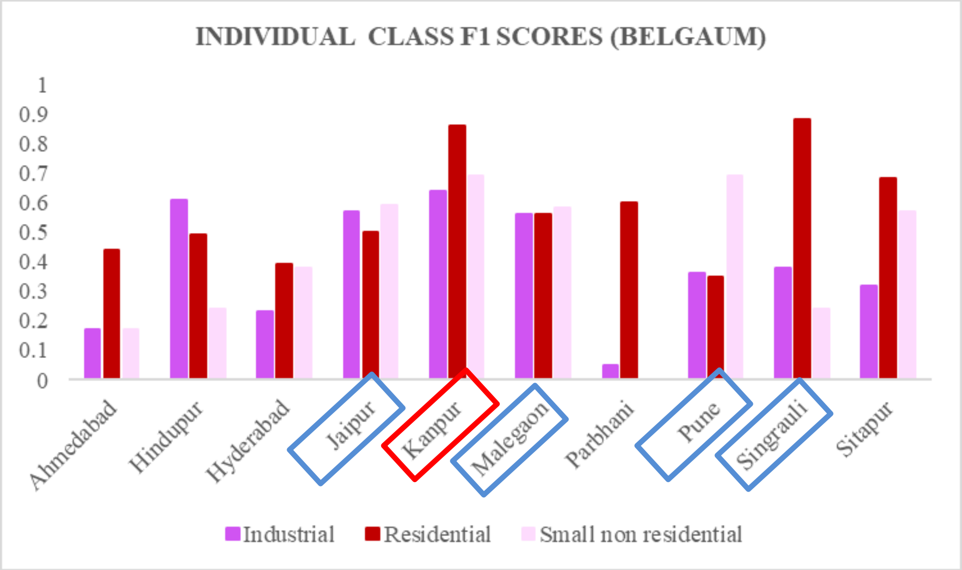


Figure 33: Individual class F1 scores: case 2 (the names of the cities which show acceptable kappa and overall accuracies are represented by a blue border and the names of the cities which are suitable for model transferability are represented by a red border).

City	Industrial (class 1)	Residential (class 2)	Small non residential (class 3)
Jaipur	0.57	0.5	0.59
Kanpur	0.64	0.86	0.69
Malegaon	0.56	0.56	0.58
Pune	0.36	0.35	0.69
Singrauli	0.38	0.88	0.24

Table 3: Individual class F1 scores of selected validation cities: case 2

5.3.3 Morphological similarities between transferable cities

It is observed that similarity of properties of regions under study is essential for successful spatial transferability, as discussed in sub section 4.5.2 of the chapter on Methodology. Hence in this section the similarities between cities suited for model transferability are assessed to find any association between the structural conditions of buildings in a city and their corresponding transferability. It has been found from the analysis in sub section 5.3.1 that a model trained in Belgaum can be successfully transferred to Kanpur (see Figure 34) and a kappa and overall accuracy of 0.7 – 0.8 and individual class F1 scores of >=0.6 can be obtained. Therefore, in this section the morphological similarities between the buildings of these cities are analysed, with respect to the industrial, residential and small non residential classes which are as follows:

- Belgaum has sugar factories, aluminium companies such as INDAL Aluminum factory, powerloom industries etc. It also has many higher education institutions such as management, medical, engineering colleges etc. (https://belagavi.nic.in/en/., 2022). Through visual assessment, a few big industries in separate industrial zones are observed. The residential buildings range from high rise apartments to low lying houses, and a number of tall apartments are found. In addition, slums are also found which account for 36 among the 58 wards of the city (Kurani., 2022).
- Kanpur is famous for its industries such as leather, garments, industrial machines etc. and is a major industrial region (Government of Uttar Pradesh, 2022) It also has many small-scale industries (MSME Development Institute of Kanpur, 2022). The planned residential buildings are well spaced and found around the core of the city, whereas the slums are found in the unplanned areas and found in the fringes of the city, with limited

spacing between them (Ahlawat Joginder., 2017). Through visual assessment it is observed that the city has large industries such as India Oil, Small Arms Factory etc.

Through visual assessment and literature review it is observed that Belgaum and Kanpur show similarities in the industrial and small non residential class buildings, especially in the small non residential buildings.

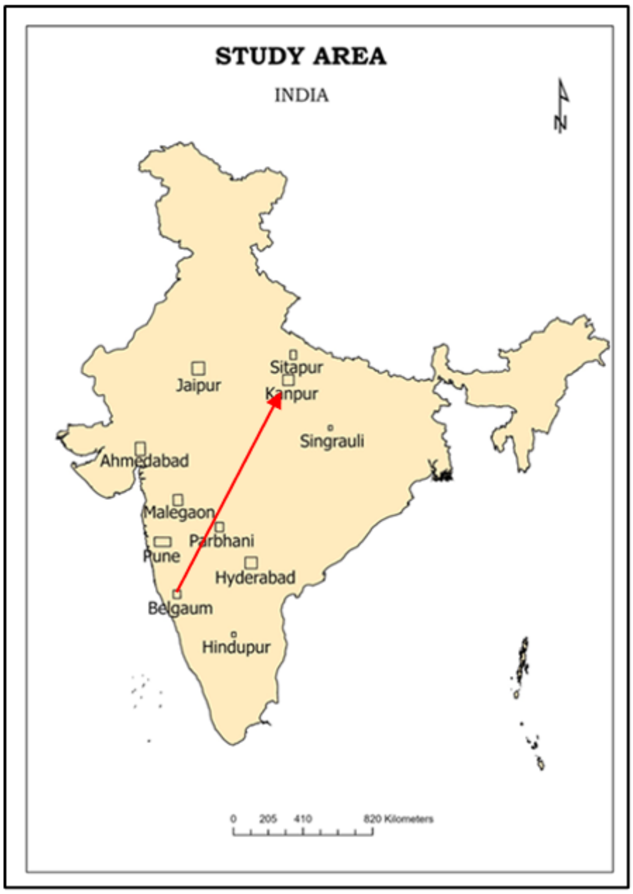


Figure 34: Model transferability: case 2 (the red lines represent that a model trained in Belgaum can be transferred to Kanpur).

5.4 Class separability and model transferability

In this section, the results of the class separability analysis are presented for all the validation cities to analyse if the separability of classes has an impact on prediction accuracies of the transferred models. For this purpose, the separability of all three classes namely industrial, residential and small non residential are analysed for each city and the following observations are made.

- Overall, volume band is observed to show the highest separability for all the cities. Specifically, the volume band (band sixteen) as seen in Figures 35 a to 35 f, followed by the median of volume band (band fourteen) as seen in Figures 35g to 35 j are found to be the most separable in all cities.

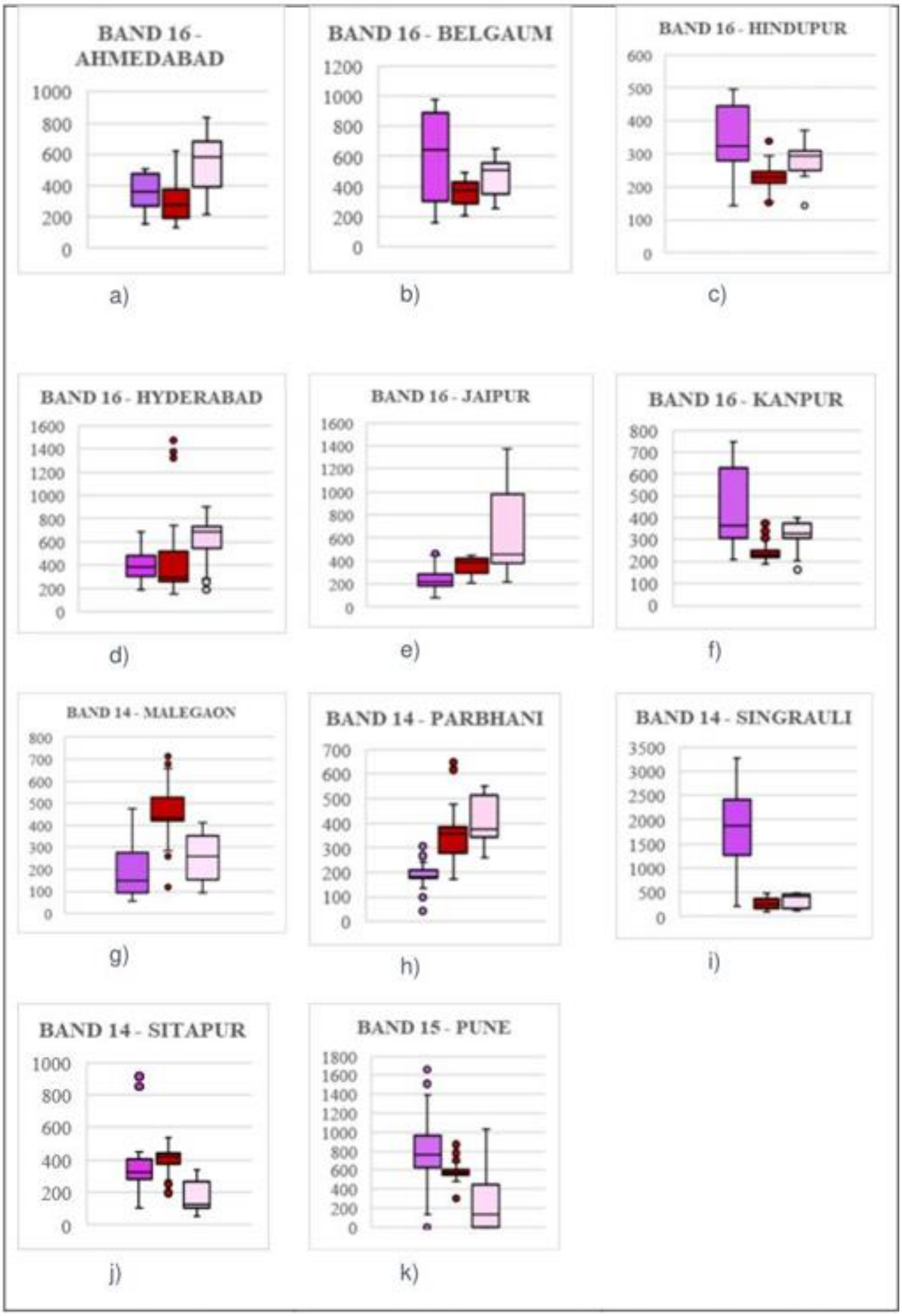


Figure 35: Most separable bands for all the classes: a) to f) showing the volume bands, g) to j) showing the median of volume bands and k) showing the standard deviation of volume band (industrial class is represented in purple, residential class in red and small non residential class in pink, according to the Corine Land Cover classes and RGB colour codes).

- Area band shows the least separability, especially the mean of area band (band one) a seen in Figures 36 a to 36 e, followed by the area band (band four) as shown in Figures 36 f to 36 h. The next least separability is observed in the height bands (band twelve) shown in Figures 36 i and 36 j, followed by the mean of fraction band (band five) in a few cities (see Figure 36 k).

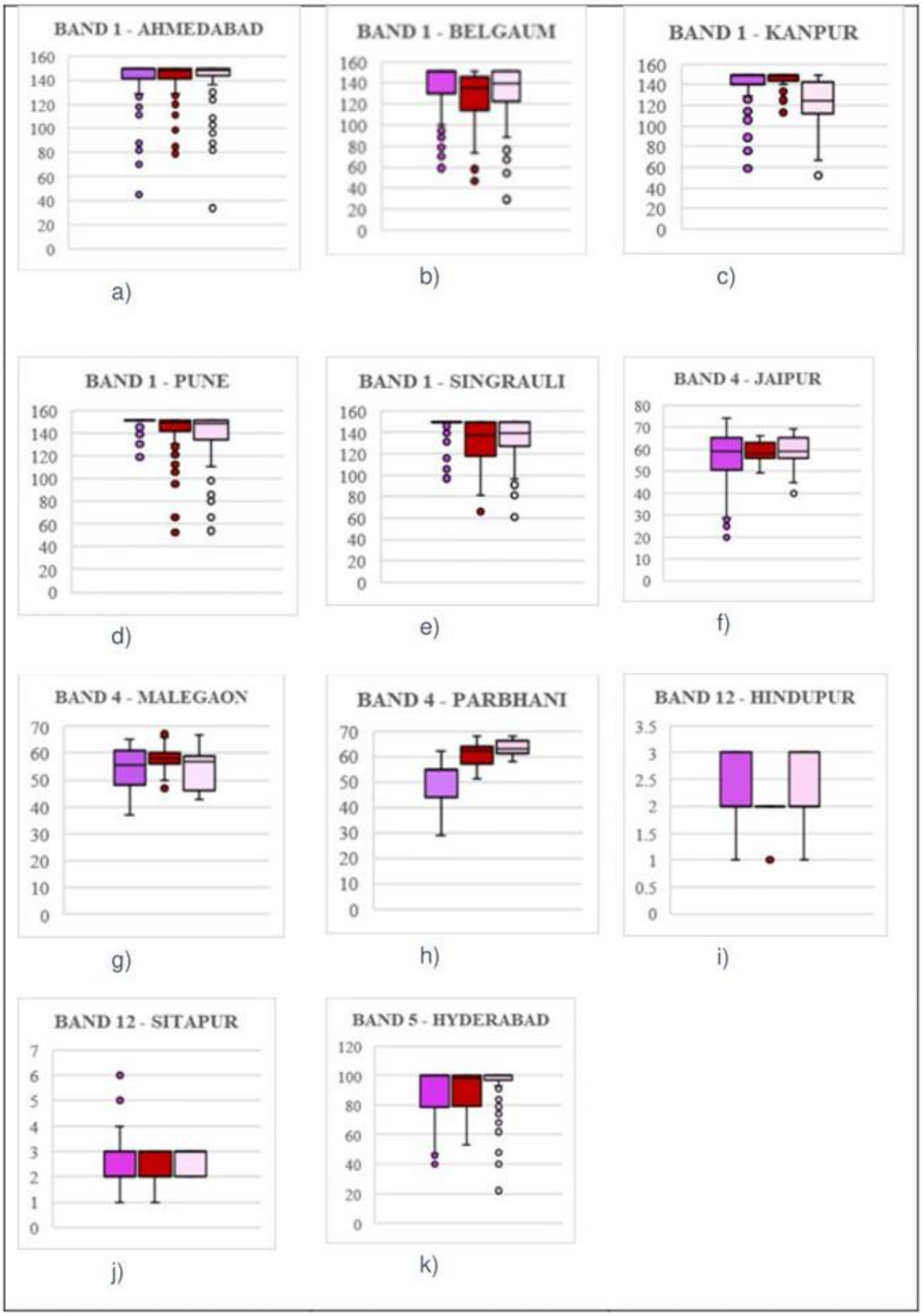


Figure 36: Least separable bands for all the classes: a) to e) showing the mean of area bands, f) to h) showing the area bands, i) and j) showing the height bands and k) showing the mean of

fraction band (industrial class is represented in purple, residential class in red and small non residential class in pink, according to the Corine Land Cover classes and RGB colour codes).

- In a few bands for some cities, the values of classes are observed to be almost similar thereby indicating that the building metric corresponding to that particular band is almost the same for such buildings in the city. Due to this factor, the model is trained in a way that all buildings corresponding to that band and classes is similar, which might affect the model transferability. Almost similar values is observed in – 1) Ahmedabad for bands one (mean of area) and five (mean of fraction) in all three classes (see Figure 37 a and Figure 37 b), 2) Belgaum for band ten (median of height of building) in all three classes (see Figure 37 c) 3) Jaipur for band four (area) in all three classes (see Figure 37 d) 4) Kanpur for band ten (median of height) in residential and small non residential classes (see Figure 37 e) 5) Sitapur for band ten (median of height) in industrial and residential classes (see Figure 37 f); band twelve (height) in all the three classes (see Figure 37 g).

^

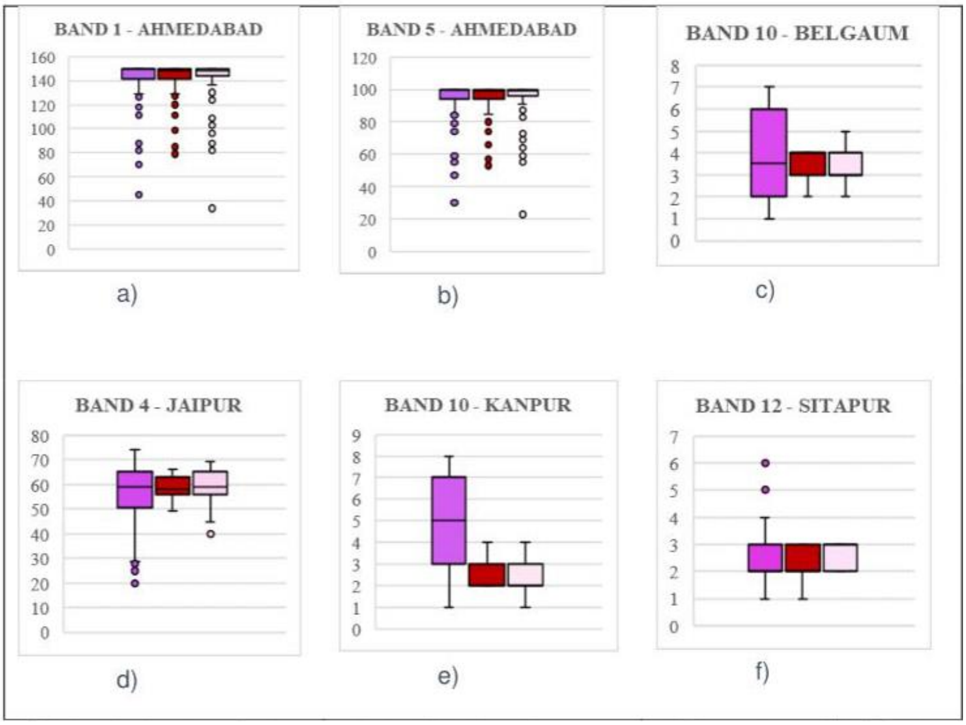


Figure 37: Similarity of values in classes (industrial class is represented in purple, residential class in red and small non residential class in pink, according to the Corine Land Cover classes and RGB colour codes).

- Another finding from the class separability analysis is that though the classes are well separated in almost all the bands, the variability of data in classes and outliers impact prediction accuracies. For instance, in Sitapur though all bands show good separability the prediction accuracies are observed to be low, the results of which are given in the Appendix.

5.5 Visual assessment of transferred model predictions

As discussed in the sub section 4.2.6 of the chapter on Methodology, in this section the binary image showing the Area of Applicability (see Figure 38) produced based on a model trained in Pune and validated in Singrauli is assessed visually. For this, initially the correctly classified areas are analysed followed by an analysis of the misclassified areas.

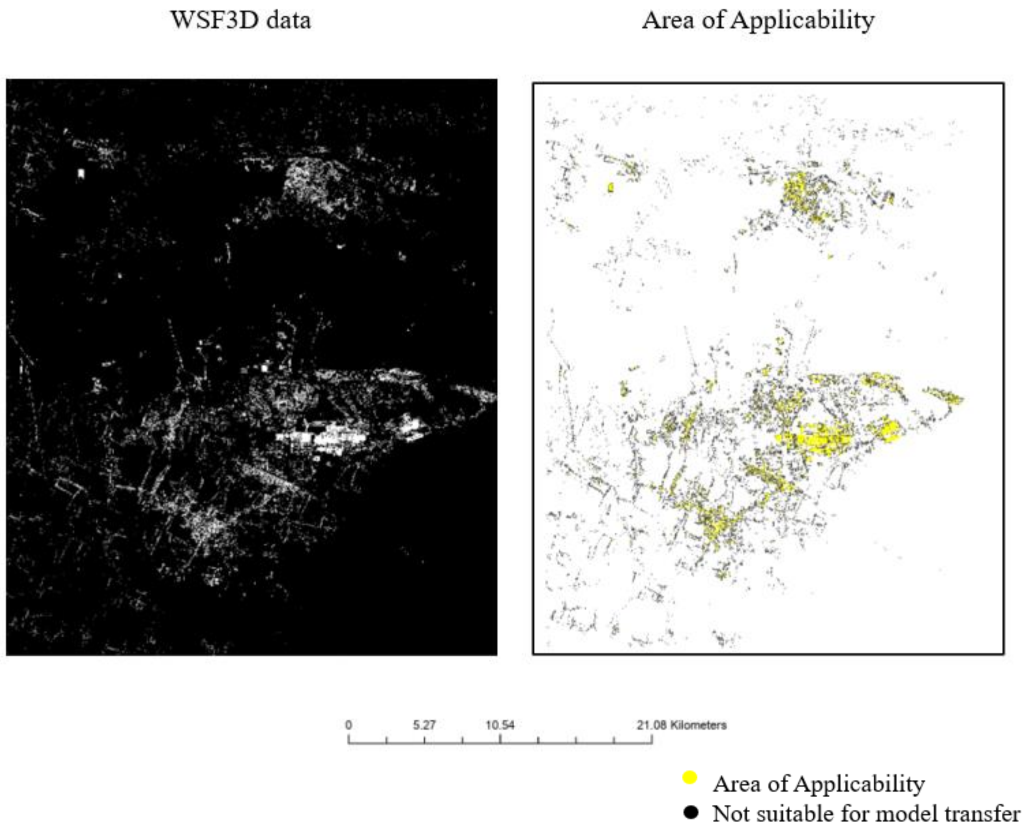
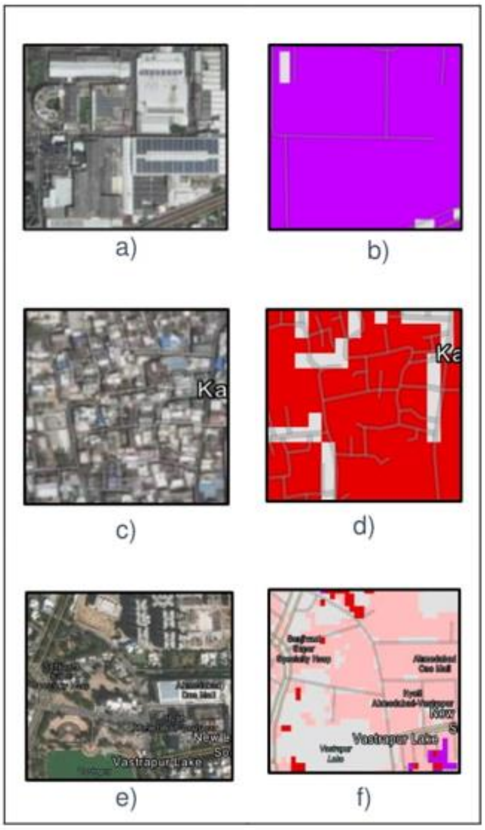


Figure 38: Area of Applicability for a model trained in Pune and validated in Singrauli.

From Figure 38, the Area of Applicability is observed which is the area where the model can be be successfully transferred. In this area, it is found visually that the buildings are correctly predicted for all the three classes (see Figure 39). It could be observed that industrial buildings (see

Figure 39 a and Figure 39 b), residential buildings (see Figure 39 c and Figure 39 d) and small non residential buildings (see Figure 39 e and Figure 39 f) are correctly predicted.



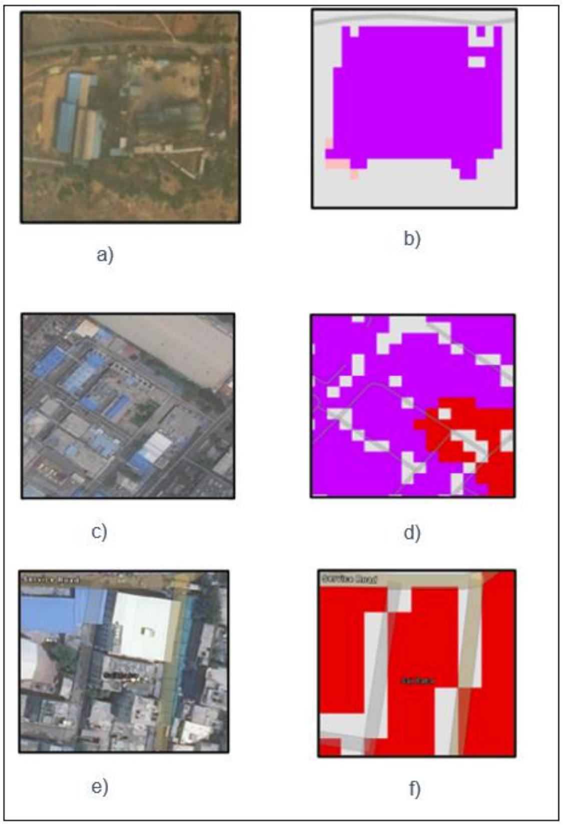


Figure 39: Correctly classified labels for a model trained in Pune and validated in Singrauli.

Figure 40: Misclassified labels for a model trained in Pune and validated in Singrauli.

In the areas other than the Area of Applicability (see Figure 38), certain misclassified labels are observed (see Figure 40). For instance, it is found visually that the smaller buildings surrounding an industry is misclassified as small non residential building (see Figure 40 a and Figure 40 b), few low rise industrial buildings are misclassified as residential buildings (see Figure 40 c and Figure 40 d) and a place of worship which belongs to the small non residential class is misclassified as residential (see Figure 40 e and Figure 40 f).

5.6 Key findings

- After evaluating the results, the main highlights of the findings are discussed as follows:
- The highest accuracies and kappa scores for the transferred model are observed in points below threshold, followed by all validation points. The points above threshold show comparatively least accuracies. This makes the points below threshold more suitable for transferring the model.
- To obtain a kappa and overall accuracy of 0.8 – 0.9 and individual class F1 scores of >= 0.7 for the transferred model, the validation city has to satisfy certain transferability criteria such as atleast 66% of the validation points have to be below threshold, a minimum of 0.79 and 0.62 of overlap coefficient and histogram intersection respectively for all points of the validation city, a minimum of 0.70 and 0.71 of overlap coefficient and histogram intersection respectively for validation points below threshold and a minimum of 30%, 38% and 15% of samples from the validation city belonging to industrial, residential and small non residential classes respectively. The model trained in Pune and transferred to Kanpur and Singrauli is found to satisfy these criteria and hence considered suitable for model transferability.
- For a kappa and overall accuracy of 0.7 – 0.8 and individual class F1 scores of >= 0.6 for the transferred model, the validation city has to satisfy certain transferability criteria such as atleast 52% of the validation points have to be below threshold, a minimum of 0.83 and 0.65 of overlap coefficient and histogram intersection respectively for all points of the validation city, a minimum of 0.70 and 0.78 of overlap coefficient and histogram intersection respectively for validation points below threshold and a minimum of 26.83%, 43.11% and 9.40% of samples from the validation city belonging to industrial, residential and small non residential classes respectively. The model trained in Belgaum and transferred to Kanpur is found to satisfy these criteria hence considered suitable for model transferability.
- More number of points of the validation city below threshold does not necessarily mean successful model transferability.
- Acceptable kappa (>=0.5) and overall accuracies (>=0.7) does not necessarily mean that the model can be successfully transferred.
- Though cities show substantiable kappa and overall accuracies, it does not always imply good prediction accuracies since the F1 scores of individual classes also have to be taken into account.

- The cities which are most suitable for model transferability are observed to have not only acceptable kappa (>=0.5) and overall accuracies (>=0.7) but also substantiable individual class F1 scores (>=0.6).
- Cities which are transferable show morphological similarities.
- Volume band shows the highest separability and area band shows the least separability.
- Similar values of classes in bands might impact the model transferability corresponding to the metric of that band.
- Besides class separability, variability of data in classes and outliers also affect the prediction accuracies, thereby affecting model transferability.
- The Area of Applicability which comprises of the points below threshold shows correctly classified labels when observed visually. This implies that the model can be successfully transferred in these areas.

6. Conclusion

The aim of this dissertation project was to investigate the spatial transferability performance of models trained with spatial metrics derived from the WSF3D dataset. For this, India along with its eleven cities namely Ahmedabad, Belgaum, Hindupur, Hyderabad, Jaipur, Kanpur, Malegaon, Parbhani, Pune, Singrauli and Sitapur were selected as the regions of interest.

For achieving the aim, the research questions answered through this thesis were, to identify any patterns in the prediction accuracies of the cities spatially, assess in which areas the model transferability can be transferred with confidence, find any existing associations between prediction accuracies and model transferability and identify the similarities between cities based on their transferability performances.

The main methodology was based on the recommendation by Meyer Hanna et al., 2021 which is, points below threshold are suitable for model transferability owing to their high prediction accuracies when the model is transferred. It was found that that the points below threshold showed the highest prediction accuracies compared to the other points. However, higher number of points of the validation city below threshold does not necessarily mean successful model transferability.

A combination of different criteria was investigated to assess in which areas the model can be successfully transferred with two different ranges of prediction accuracies. It was found that a model trained in Pune and transferred to Kanpur and Singrauli fulfilled the criteria to obtain a kappa and overall accuracy of 0.8 – 0.9 and individual class F1 scores of >= 0.7 for the transferred model. And, a model trained in Belgaum and transferred to Kanpur fulfilled the criteria to obtain a kappa and overall accuracy of 0.7 – 0.8 and individual class F1 scores of >= 0.6 for the transferred model. Therefore, the above mentioned cities were found to be suitable for transferring a model.

An analysis of the existing associations between prediction accuracies and model transferability showed that an acceptable kappa and overall accuracy does not necessarily mean successful model transferability. Though cities showed substantiable kappa and overall accuracies, it did not always imply good prediction accuracies since the F1 scores of individual classes also had to be taken into account.

Apart from these criteria, transferable cities show morphological similarities, with respect to the buildings belonging to industrial, residential and small non residential classes. The class separability analysis results showed that besides class separability, similar values of classes in bands, variability of data in classes and outliers also affect the model transferability. Finally, the Area of Applicability which comprises of the points below threshold showed correctly classified labels when observed visually, which implies that the model can be successfully transferred in these areas.

Hence it can be stated that the implemented methodology is able to successfully assess the spatial transferability of models trained solely on the basis of spatial metrics derived from the WSF3D dataset. On a final note, this methodology which predicts the Area of Applicability (area where the model trained in one city can be successfully transferred to another city), facilitates the user to assess and make informed decisions with respect to model transferability in areas where validation data is not available.

Recommendations for future work

The proposed methodology has scope to be modified and improved in different aspects. Some recommendations for future work are as follows:

- The validation and testing performed between cities for this research could be extended to the scale of a state, country or continent level.
- Investigations could be made by training a model on a city level and making predictions for country level.
- Additional criteria that impact successful transferability could be explored, by making analysis using different metrics.
- The proposed methodology could be implemented using other machine learning models, to evaluate their individual performances.

References

Aher B. Jayshree, Shraddha Chavan, 2021. Implementation of Principles of New Urbanism on Malegaon City (Nashik District). International Journal for Research in Applied Science & Engineering Technology, 9(8). Available at https://doi.org/10.22214/ijraset.2021.37627. Accessed on [24.02.2022].

Ahlawat Joginder, 2017. A Geographical Review of the Land Use Patterns of Kanpur City, India. International Journal of Humanities and Social Science Invention, 6(7), pp.21-27.

Allamy Khalaf Haider, Rafiqul Zaman Khan, 2015. Methods to avoid over-fitting and under-fitting in supervised machine learning (comparative study). Computer Science, Communication & Instrumentation Devices, pp. 163-172. Available at http://dx.doi.org/10.3850/978-981-09-5247-1_017 . Accessed on [09.02.2022].

Alshari A. Eman, Bharti W. Gawali, 2021. Development of classification system for LULC using remote sensing and GIS. Global Transitions Proceedings, 2(1), pp.8-17. Available at https://doi.org/10.1016/j.gltp.2021.01.002. Accessed on [17.11.2021].

Bansal Megha, 2016. Occupational health hazards and awareness of occupational safety among workers of textile dyeing industries in Jaipur, India. Suresh Gyan Vihar University International Journal of Environment, Science and Technology, 2(2), pp.30-38.

Beard A. Victoria, Satterthwaite David, Mitlin Diana, Du Jillian, 2022. Out of sight, out of mind: Understanding the sanitation crisis in global South cities. Journal of Environmental Management, 304, 114285. Available at https://doi.org/10.1016/j.jenvman.2021.114285. Accessed on [15.10.2021].

Belgiu Mariana, Lucian Drăguţ, 2016. Random forest in remote sensing: A review of applications and future directions. ISPRS Journal of Photogrammetry and Remote Sensing, 114, pp.24-31. Available at https://doi.org/10.1016/j.isprsjprs.2016.01.011. Accessed on [14.09.2021].

Carrão Hugo, Pedro Sarmento, António Araújo, Mário Caetano, 2007. Separability analysis of land cover classes at regional scale: A comparative study of MERIS and MODIS data. Proceedings ‘Envisat Symposium 2007’, SP-636.

Cengiz Serhat, Sevgi Görmüş, Dicle Oğuz, 2022. Analysis of the urban growth pattern through spatial metrics; Ankara City. Land Use Policy, 112, 105812. Available at https://doi.org/10.1016/j.landusepol.2021.105812. Accessed on [14.09.2021].

Census of India, 2011. District Census Handbook Singrauli, Series – 24, Part XII – A.

Chachondhia Prachi, Achala Shakya, Gaurav Kumar, 2021. Performance evaluation of machine learning algorithms using optical and microwave data for LULC classification. Remote Sensing Applications: Society and Environment, 23. Available at https://doi.org/10.1016/j.rsase.2021.100599. Accessed on [15.10.2021].

Chakure Afroz, 2019. Random Forest Regression. Available at https://medium.com/swlh/random-forest-and-its-implementation-71824ced454f. Accessed on [17.03.2022].

Chen Bin Ying Tu, Yimeng Song, David M. Theobald, Tao Zhang, Zhehao Ren, Xuecao Li, Jun Yang, Jie Wang, Xi Wang, Peng Gong, Yuqi Bai, Bing Xu, 2021. Mapping essential urban land use categories with open big data: Results for five metropolitan areas in the United States of America. ISPRS Journal of Photogrammetry and Remote Sensing, 178, pp.203-218. Available at https://doi.org/10.1016/j.isprsjprs.2021.06.010. Accessed on [08.09.2021].

Dharumarajan Subramanian, Rajendra Hegde, 2020. Digital mapping of soil texture classes using Random Forest classification algorithm. Soil Use and Management, 38(1), pp.135-149. Available at https://doi.org/10.1111/sum.12668. Accessed on [23.09.2021].

Du Shihong, Fangli Zhang, Xiuyuan Zhang, 2015. Semantic classification of urban buildings combining VHR image and GIS data: An improved random forest approach. ISPRS Journal of Photogrammetry and Remote Sensing, 105, 107-119. Available at https://doi.org/10.1016/j.isprsjprs.2015.03.011. Accessed on [18.11.2021].

Effati Fatemeh, Hazhir Karimi, Ahmadreza Yavari, 2021. Investigating effects of land use and land cover patterns on land surface temperature using landscape metrics in the city of Tehran, Iran. Arabian Journal of Geosciences, 14, 1240. Available at https://doi.org/10.1007/s12517-021-07433-4 . Accessed on [14.09.2021].

Erbek Sunar F., C. Özkan, M. Taberner, 2004. Comparison of maximum likelihood classification method with supervised artificial neural network algorithms for land use activities. International Journal of Remote Sensing, 25(9), pp.1733-1748. Available at https://doi.org/10.1080/0143116031000150077. Accessed on [19.01.2022].

Bhardwaj G., Thomas Esch, Somik V. Lall, M. Marconcini, Maria E. Soppelsa, S. Wahba, 2020. Cities, crowding and the coronavirus: Predicting contagion risk hotspots. Available at https://thedocs.worldbank.org/en/doc/979461590683301675-

0200022020/original/Citiescrowdingandthecoronaviruspredictingcontagionriskhotspots.pdf Accessed on [07.09.2021].

European Commission, 2019. The Future of Cities – Trends and Drivers. Available at https://urban.jrc.ec.europa.eu/thefutureofcities/urbanisation#the-chapter . Accessed on [08.09.2021].

European Environment Agency, 2021. Corine land cover classes and RGB color codes. Available at https://www.eea.europa.eu/data-and-maps/data/corine-land-cover-2/corine-land-cover-classes-and . Accessed on [02.03.2022].

European Space Agency, 2014. Land Cover CCI – Product User Guide – Version 2. Available at https://www.esa-landcover-cci.org/?q=webfm_send/84. Accessed on [08.12.2021].

Fragou Sotiria, Kleomenis Kalogeropoulos, Nikolaos Stathopoulos, Panagiota Louka, Prashant K. Srivastava, Sotiris Karpouzas, Dionissios P. Kalivas, George P. Petropoulos, 2020. Quantifying Land Cover Changes in a Mediterranean Environment Using Landsat TM and Support Vector Machines. Forests, 11(7), pp.750. Available at https://doi.org/10.3390/f11070750. Accessed on [15.10.2021].

Government of Karnataka, 2022. District Belagavi. Available at https://belagavi.nic.in/en/. Accessed on [23.03.2022].

Government of Madhya Pradesh, 2022. District Singrauli. Available at https://singrauli.nic.in/en/. Accessed on [23.03.2022].

Government of Uttar Pradesh, 2022. Kanpur Nagar. Available at https://kanpurnagar.nic.in/. Accessed on [23.03.2022].

Grippa T., Stefanos Georganos, Soukaina Zarougui, Pauline Bognounou, Eric Diboulo, Yann Forget, Moritz Lennert, Sabine Vanhuysse, Nicholus Mboga, Eléonore Wolff, 2018. Mapping urban land use at street block level using OpenStreetMap, Remote Sensing data, and spatial metrics. ISPRS International Journal of Geo-Information, 7(7), pp. 246. Available at https://doi.org/10.3390/ijgi7070246. Accessed on [09.09.2021].

Hernandez Ruiz Elias Ivan, W. Shi, 2017. A Random Forests classification method for urban land-use mapping integrating spatial metrics and texture analysis. International Journal of Remote Sensing, 39 (4), pp. 1175-1198. Available at https://doi.org/10.1080/01431161.2017.1395968. Accessed on [23.09.2021].

Hamad Rahel, 2020. An assessment of Artificial Neural Networks, Support Vector Machines and Decision Trees for land cover classification using Sentinel-2A data. Applied Ecology and Environmental Sciences, 8(6), pp.459-464.

Han Wei, Ruyi Feng, Lizhe Wang, Yafan Cheng, 2018. A semi-supervised generative framework with deep learning features for high-resolution remote sensing image scene classification. ISPRS Journal of Photogrammetry and Remote Sensing, 145 (A), 23-43. Available at https://doi.org/10.1016/j.isprsjprs.2017.11.004. Accessed on [07.10.2021].

Hermosilla Txomin, Michael A. Wulder, Joanne C. White, Nicholas C. Coops, 2022. Land cover classification in an era of big and open data: Optimizing localized implementation and training data selection to improve mapping outcomes. Remote Sensing of Environment, 268, 112780. Available at https://doi.org/10.1016/j.rse.2021.112780. Accessed on [09.09.2021].

Herold Martin, Helen Couclelis, Keith C. Clarke, 2003. The role of spatial metrics in the analysis and modeling of urban land use change. Computers, Environment and Urban Systems, 29, 369-399. Available at https://doi.org/10.1016/j.compenvurbsys.2003.12.001. Accessed on [13.09.2021].

Hu Tengyun, Jun Yang, Xuecao Li, Peng Gong, 2016. Mapping Urban Land Use by Using Landsat Images and Open Social Data. Remote Sensing, 8(2), pp.151. Available at https://doi.org/10.3390/rs8020151. Accessed on [09.09.2021].

Hu Yunfeng, Qianli Zhang, Yunzhi Zhang, Huimin Yan, 2018. A Deep Convolution Neural Network Method for Land Cover Mapping: A Case Study of Qinhuangdao, China. Remote Sensing, 10(12), pp.2053. Available at https://doi.org/10.3390/rs10122053. Accessed on [08.12.2021].

Huang Bo, Bei Zhao, Yimeng Song, 2018. Urban land-use mapping using a deep convolutional neural network with high spatial resolution multispectral remote sensing imagery. Remote Sensing of Environment, 214, 73-86. Available at https://doi.org/10.1016/j.rse.2018.04.050. Accessed on [15.11.2021].

Huang C., L.S. Davis, J.R.G. Townsend, 2010. An assessment of support vector machines for land cover classification. International Journal of Remote Sensing, 23(4), pp.725-749. Available at https://doi.org/10.1080/01431160110040323. Accessed on [29.10.2021].

Hwang Alex, Emily C. Higgins, Marc Pomplun, 2009. A model of top-down attentional control during visual search in complex scenes. Journal of Vision, 9(5):25, 1-18. Available at http://dx.doi.org/10.1167/9.5.25. Accessed on [03.03.2022].

Imam U.K. Aabshar, Uttam Kumar Banerjee, 2016. Urbanisation and greening of Indian cities: Problems, practices, and policies. Ambio, 45, pp.442-457. Available at https://dx.doi.org/10.1007%2Fs13280-015-0763-4. Accessed on [16.03.2022].

Jin Schichao, Yanjun Su, Shang Gao, Tianyu Hu, Jin Liu, Qinghua Guo, 2018. The transferability of Random Forest in canopy height estimation from multi-source remote sensing data. Remote Sensing, 10(8), pp.1183. Available at https://doi.org/10.3390/rs10081183. Accessed on [16.09.2021].

Juel Anders, Geoffrey Brian Groom, Jens-Christian Svenning, Rasmus Ejrnæs, 2015. Spatial application of Random Forest models for fine-scale coastal vegetation classification using object based analysis of aerial orthophoto and DEM data. International Journal of Applied Earth Observation and Geoinformation, 42, pp.106-114, Available at https://doi.org/10.1016/j.jag.2015.05.008. Accessed on [14.09.2021].

Junttila Samuli, Roope Näsi, Niko Koivumäki, Mohammad Imangholiloo, Ninni Saarinen, Juha Raisio, Markus Holopainen, Hannu Hyyppä, Juha Hyyppä, Päivi Lyytikäinen-Saarenmaa, Mikko Vastaranta, Eija Honkavaara, 2022. Multispectral Imagery Provides Benefits for Mapping Spruce Tree Decline Due to Bark Beetle Infestation When Acquired Late in the Season. Remote Sensing, 14 (4), 909. Available at https://doi.org/10.3390/rs14040909 . Accessed on [04.11.2021].

Kandrika Sreenivas, P. S. Roy, 2008. Land use land cover classification of Orissa using multi-temporal IRS-P6 awifs data: A decision tree approach. International Journal of Applied Earth Observation and Geoinformation, 10(2), pp.186-193. Available at https://doi.org/10.1016/j.jag.2007.10.003. Accessed on [16.09.2021].

Kerins Peter, Guzder-Williams Brookie, Mackres Eric, Rashid Taufiq, Pietraszkiewwicz Eric, 2021. Mapping Urban Land Use in India and Mexico using Remote Sensing and Machine Learning. Technical Note, World Resources Institute. Available at https://doi.org/10.46830/writn.20.00048. Accessed on [10.09.2021].

Killemsetty Namesh, 2013. Understanding the evolution of slums in Ahmedabad through the integration of survey Data sets. Innovations in Urban Development: Incremental Housing, Big Data and Gender, pp.127-145.

Kurani, 2022. Environment and quality of life in urban slums a case study of Gandhinagar slum area, Belagavi city Karnataka, India. Geographical Analysis, 4(1), pp.1-6.

Lee S. M, J. H. Xin. John, Stephen Westland, 2005. Evaluation of image similarity by histogram intersection. Color Research and Application, 30(4), 265 – 274. Available at http://dx.doi.org/10.1002/col.20122. Accessed on [03.03.2022].

Machado, Diego Fernandes Terra, 2019. Soil type spatial prediction from Random Forest: different training datasets, transferability, accuracy and uncertainty assessment. Soils and Plant Nutrition, 76(3). Available at https://doi.org/10.1590/1678-992X-2017-0300. Accessed on [14.09.2021].

Mahapatra Diptiranjan, Priyadarshi Shukla, Subash Dhar, 2012. External cost of coal based electricity generation: A tale of Ahmedabad city. Energy Policy, 49, pp.253-265. Available at https://doi.org/10.1016/j.enpol.2012.06.014. Accessed on [23.03.2022].

Mao Wanliu, Debin Lu, Li Hou, Xue Liu, Wenze Yue, 2020. Comparison of Machine-Learning Methods for Urban Land-Use Mapping in Hangzhou City, China. Remote Sensing, 12(17), pp. 2817. Available at https://doi.org/10.3390/rs12172817. Accessed on [11.09.2021].

Marcon Alessandro, Kees de Hoogh, John Gulliver, Rob Beelen, Anna L. Hansell, 2015. Development and transferability of a nitrogen dioxide land use regression model within the Veneto region of Italy. Atmospheric Environment, 122, pp.696-704. Available at https://doi.org/10.1016/j.atmosenv.2015.10.010. Accessed on [07.10.2021].

Marconcini Mattia, 2020. World Settlement Footprint – where do humans live? Available at https://www.dlr.de/blogs/en/all-blog-posts/world-settlement-footprint-where-do-humans-live.aspx. Accessed on [08.09.2021].

Markandey Kalpana, 2020. Slums of Hyderabad: A spatio temporal analysis. Population Geography, 42(2), pp.19-32

Masolele N., Veronique De Sy, Martin Herold, Diego Marcos, Jan Verbesselt, Fabian Gieseke, Adugna G. Mullissa, Christopher Martius, 2021. Spatial and temporal deep learning methods for deriving land-use following deforestation: A pan-tropical case study using Landsat time series. Remote Sensing of Environment, 264, 112600. Available at https://doi.org/10.1016/j.rse.2021.112600. Accessed on [25.11.2021].

McHugh Mary, 2012. Interrater reliability: the kappa statistic. Biochemia Medica, 22(3), pp.276-282. Available at https://www.researchgate.net/publication/232646799_Interrater_reliability_The_kappa_statistic. Accessed on [13.01.2022].

Meyer Hanna, Edzer Pebesma, 2021. Predicting into unknown space? Estimating the area of applicability of spatial prediction models. Methods in Ecology and Evolution, 12(9), pp.1620-1633. Available at https://doi.org/10.1111/2041-210X.13650. Accessed on [04.04.2022].

Mondal Arun, Sananda Kundu, Surendra Kumar Chandniha, Rituraj Shukla, 2012. Comparison of Support Vector Machine and Maximum Likelihood classification technique using satellite imagery. International Journal of Remote Sensing and GIS, 1(2), pp.116-123.

MSME Development Institute of Kanpur, 2022. Available at http://msmedikanpur.gov.in/views/default.asp. Accessed on [17.03.2022].

Munoz Sergio, Shrikant I. Bangdiwala, 1997. Interpretation of Kappa and B statistics measures of agreement. Journal of Applied Statistics, 1, pp.105-112. Available at https://doi.org/10.1080/02664769723918. Accessed on [13.01.2022].

Narkhede Sarang, 2018. Understanding Confusion Matrix. Towards Data Science. https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62 . Accessed on [13.01.2022].

Pacifici Fabio, Marco Chini, William J. Emery, 2009. A neural network approach using multi-scale textural metrics from very high-resolution panchromatic imagery for urban land-use classification. Remote Sensing of Environment. 113(6), pp.1276-1292. Available at https://doi.org/10.1016/j.rse.2009.02.014. Accessed on [29.09.2021].

Pal M., 2005, Random forest classifier for remote sensing classification. International Journal of Remote Sensing, 26(1), pp.217-222.

Palacios – Lopez Daniela, Thomas Esch, Mattia Marconcini, Julian Zeidler, 2022. Towards an improved large-scale gridded population dataset: A Pan-European quantitative assessment of the contributions of the new WSF3D dataset into the field of human population modelling. Remote Sensing. 14 (2), pp.325.

Park No-Wook, Phaedon C. Kyriakidis, Suk-Young Hong, 2016. Spatial Estimation of Classification Accuracy Using Indicator Kriging with an Image-Derived Ambiguity Index. Remote Sensing, 8 (4), 320. Available at https://doi.org/10.3390/rs8040320. Accessed on [24.01.2022].

Parker C. Dawn, Vicky Meretsky, 2004. Measuring pattern outcomes in an agent-based model of edge-effect externalities using spatial metrics. Agriculture, Ecosystems and Environment, 101 (2-3), 233-250. Available at https://doi.org/10.1016/j.agee.2003.09.007. Accessed on [13.09.2021].

Petropoulos P. George, Kostas Arvanitis, Nick Sigrimis, 2012. Hyperion hyperspectral imagery analysis combined with machine learning classifiers for land use/cover mapping. Expert Systems with Applications, 39(3), pp.3800-3809. Available at https://doi.org/10.1016/j.eswa.2011.09.083. Accessed on [14.09.2021].

Poplawski, Karla, Timoty Gould, Eleanor Setton, **Allen R., Su J., Larson T., Henderson S., Brauer M., Hystad P., Lightowlers C., Keller P., Cohen M., Silva C., Buzzeli M**., 2009. Intercity transferability of land use regression models for estimating ambient concentrations of nitrogen dioxide. Journal of Exposure Science and Environtal Epidemiology, 19, pp. 107–117. Available at https://doi.org/10.1038/jes.2008.15 Accessed on [17.09.2021].

Poulicos Prastacos, Kostas Arvanitis, Nick Sigrimis, 2017. Using the Urban Atlas dataset for estimating spatial metrics - Methodology and application in urban areas of Greece. Cybergeo: European Journal of Geography, 815. Available at https://doi.org/10.4000/cybergeo.28051. Accessed on [20.09.2021].

Probst Philipp, Mavin N. Wright, Anne – Laure Boulesteix, 2019. Hyperparameters and tuning strategies for random forest. WIREs Data Mining Knowledge Discovery, 9(3). Available at https://doi.org/10.1002/widm.1301. Accessed on [13.09.2021].

Punia Milap, P.K. Joshi, M.C. Porwal, 2011. Decision tree classification of land use land cover for Delhi, India using IRS-P6 AWiFS data. Expert Systems with Applications, 38(5), pp.5577-5583. Available at https://doi.org/10.1016/j.eswa.2010.10.078. Accessed on [06.10.2021].

Rasool Rehana, Miftah Ul Shafiq, Pervez Ahmed, 2016. Urbanisation problems and growth of slums in Srinagar urban centre of Kashmir valley (J &K). International Journal of Recent Scientific Research, 7 (1), pp.8596-8601. Available at https://www.researchgate.net/publication/298196552. Accessed on [14.03.2021].

Ray Bhaswati, 2016. Quality of life in selected slums of Kolkata: a step forward in the era of pseudo-urbanisation. The International Journal of Justice and Sustainability, 22 (3), pp.365-387. Available at https://doi.org/10.1080/13549839.2016.1205571. Accessed on [25.03.2021].

Rodriguez-Galiano F. Victor, B. Ghimire, J. Rogan, M. Chica-Olmo, J.P. Rigol-Sanchez, 2012. An assessment of the effectiveness of a random forest classifier for land-cover classification. ISPRS Journal of Photogrammetry and Remote Sensing, 67, pp.93-104. Available at https://doi.org/10.1016/j.isprsjprs.2011.11.002. Accessed on [18.10.2021].

Roy Bishal, 2021. A machine learning approach to monitoring and forecasting spatio-temporal dynamics of land cover in Cox's Bazar district, Bangladesh from 2001 to 2019. Environmental Challenges, 5. Available at https://doi.org/10.1016/j.envc.2021.100237. Accessed on [27.09.2021].

Sarode, 2016. A study of Green HRM and Its Evaluation with Existing HR Practices in Industries within Pune Region. International Journal of Research in Engineering, IT and Social Sciences, 6(4), pp.49-67.

Sauti Raymond, Uzay Karahalil, 2022. Investigating the spatiotemporal changes of land use/land cover and its implications for ecosystem services between 1972 and 2015 in Yuvacık. Environmental Monitoring and Assessment, 194, 311. Available at https://doi.org/10.1007/s10661-022-09912-x. Accessed on [04.10.2021].

Scikit-learn, 2021. Cohen kappa score. Available at https://scikit-learn.org/stable/modules/generated/sklearn.metrics.cohen_kappa_score.html . Accessed on [13.01.2022].

Scikit Yellow Brick, 2019. Classification Report. Available at https://www.scikit-yb.org/en/latest/api/classifier/classification_report.html . Accessed on [13.01.2022].

Scikit-learn, 2021b. F1 score. Available at https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html . Accessed on [13.01.2022].

Sharma Roshan, Udo Nehren, Syed A. Rahman, Maximilian Meyer, Bhagawat Rimal, Gilang Aria Seta, and Himlal Baral, 2018. Modeling Land Use and Land Cover Changes and Their Effects on Biodiversity in Central Kalimantan, Indonesia. Land, 7 (2), 57. Available at https://doi.org/10.3390/land7020057. Accessed on [18.11.2021].

Sheykhmousa Mohammadreza, Norman Kerle, Monika Kuffer, Saman Ghaffarian, 2019. Post-Disaster Recovery Assessment with Machine Learning-Derived Land Cover and Land Use Information. Remote Sensing, 11 (10), 1174. Available at https://doi.org/10.3390/rs11101174. Accessed on [04.11.2021].

Shiefer Felix, Teja Kattenborn, Annett Frick, Julian Frey, Peter Schall, Barbara Koch, Sebastian Schmidtlein, 2020. Mapping forest tree species in high resolution UAV-based RGB-imagery by means of convolutional neural networks. ISPRS Journal of Photogrammetry and Remote Sensing, 170, 205-215. Available at https://doi.org/10.1016/j.isprsjprs.2020.10.015. Accessed on [20.09.2021].

Singh Kumar Sudir, Prashant K. Srivastava, Manika Gupta, Jay Krishna Thakur, Saumitra Mukherjee, 2013. Appraisal of land use/land cover of mangrove forest ecosystem using support vector machine. Environmental Earth Sciences, 71, pp.2245-2255.

Srivastava Ambey, 2018. Heritage Tourism and Urbanisation - A Case of Jaipur City. Travel, Tourism and Hospitality, Part V: Case Studies, pp. 322-338.

Sullivan Lisa, 2016. Box-Whisker Plots for Continuous Variables. Boston University School of Public Health. Available at https://sphweb.bumc.bu.edu/otlt/mph-modules/bs/bs704_summarizingdata/bs704_summarizingdata8.html. Accessed on [28.02.2022].

Talukdar Swapan, Pankaj Singha, Susanta Mahato, Shahfahad, Swades Pal, Yuei-An Liou, Atiqur Rahman, 2020. Land-Use Land-Cover Classification by Machine Learning Classifiers for satellite observations - a review. Remote Sensing, 12(7), pp.1135. Available at https://doi.org/10.3390/rs12071135. Accessed on [11.09.2021].

Thapa Bahadur Rajesh, Yuji Murayama, 2009. Examining Spatiotemporal Urbanization Patterns in Kathmandu Valley, Nepal: Remote Sensing and Spatial Metrics Approaches. Remote Sensing, 1 (3), 534-556. Available at https://doi.org/10.3390/rs1030534. Accessed on [18.11.2021].

Valentijn Tinka, Jacopo Margutti, Marc van den Homberg, Jorma Laaksonen, 2020. Multi-hazard and spatial transferability of a CNN for automated building damage assessment. Remote Sensing, 12(17), pp.2839. Available at https://doi.org/10.3390/rs12172839. Accessed on [04.11.2021].

Verma Vijay, Rajesh Kumar Aggarwal, 2020. A comparative analysis of similarity measures akin to the Jaccard index in collaborative recommendations: empirical and theoretical perspective. Social Network Analysis and Mining, 10, 43. Available at https://doi.org/10.1007/s13278-020-00660-9. Accessed on [03.02.2022].

Waśniewski Adam, Agata Hościło, Bogdan Zagajewski, Dieudonné Moukétou-Tarazewicz, 2020. Assessment of Sentinel-2 satellite images and Random Forest Classifier for rainforest mapping in Gabon. Forests, 11(9), pp.941, Available at https://doi.org/10.3390/f11090941. Accessed on [05.09.2021].

Wenger J. Seth, Olden D. Julian, 2012. Assessing transferability of ecological models: an underappreciated aspect of statistical validation. Methods in Ecology and Evolution, 3 (2), 260-267. Available at https://doi.org/10.1111/j.2041-210X.2011.00170.x. Accessed on [18.11.2021].

Xiaodong Na, Zhang Shuqing, Li Xiaofeng, Yu Huan, Liu Chunyue, 2010. Improved land cover mapping using Random Forests combined with Landsat Thematic Mapper imagery and ancillary geographic data. Photogrammetric Engineering and Remote Sensing, 7, pp.833-840. Available at https://doi.org/10.14358/PERS.76.7.833 . Accessed on [16.09.2021].

Xing Hanfa, Yuan Meng, 2018. Integrating landscape metrics and socioeconomic features for urban functional region classification. Computers, Environment and Urban Systems, 72, pp.134-145. Available at https://doi.org/10.1016/j.compenvurbsys.2018.06.005. Accessed on [22.09.2021].

Yang Wanting, Xianfeng Zhang, Peng Luo, 2021. Transferability of Convolutional Neural Network Models for Identifying Damaged Buildings Due to Earthquake. Remote Sensing, 13 (3), 504. Available at https://doi.org/10.3390/rs13030504. Accessed on [21.10.2021].

Yulianto Fajar, Suwarsono, Udhi Catur Nugroho, Nunung Puji Sunarmodo, Wismu, Khomarudin Muhammad Rokhis, 2020. Spatial-Temporal Dynamics Land Use/Land Cover Change and Flood Hazard Mapping in the Upstream Citarum Watershed, West Java, Indonesia. Quaestiones Geographicae, 39 (1), 125-146. Available at https://doi.org/10.2478/quageo-2020-0010. Accessed on [23.09.2021].

Zhang X.M., G. J. He, Z. M. Zhang, Y. Peng, T. F. Long, 2017. Spectral-spatial multi-feature classification of remote sensing big data based on a random forest classifier for land cover mapping. Cluster Computing, 20, pp.2311-2321. Available at https://link.springer.com/article/10.1007/s10586-017-0950-0#ref-CR8. Accessed on [07.10.2021].

Zhang Xiao, Liu Liangyun, Chen Xidong, Xie Shuai, Gao Yuan, 2019. Fine Land-Cover Mapping in China Using Landsat Datacube and an Operational SPECLib-Based Approach. Remote Sensing, 11, 1056. Available at https://doi.org/10.3390/rs11091056. Accessed on [04.11.2021].

Zhang Xiao, Liu Liangyun, Chen Xidong, Gao Yuan, Xie Shuai, Mi Jun, 2021. GLC_FCS30: global land-cover product with fine classification system at 30 m using time-series Landsat imagery. Earth System Science Data, 13, 2753-2776. Available at https://essd.copernicus.org/articles/13/2753/2021/#section1. Accessed on [04.11.2021].

The European Space Agency, 2021. Mapping our human footprint from space. Available at https://www.esa.int/Applications/Observing_the_Earth/Mapping_our_human_footprint_from_space. Accessed on [08.09.2021].

United Nations Department of Economic and Social Affairs, 2018. 68% of the world population projected to live in urban areas by 2050, says UN. Available at https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html. Accessed on [08.09.2021].

United Nations Department of Economic and Social Affairs, 2021. Make cities and human settlements inclusive, safe, resilient and sustainable. Available at https://sdgs.un.org/goals/goal11. Accessed on [08.09.2021].

Appendix 1

Source code

```
#Import all the necessary packages
from osgeo import gdal, ogr, gdal_array
import numpy as np
import numpy as geek
import random
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
import skimage.io as io
import sys
import math
import numpy.ma as ma
from pathlib import Path
import math, operator
import os

from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.preprocessing import MinMaxScaler
from scipy.spatial.distance import minkowski
from sklearn.metrics import classification_report, accuracy_score,
confusion_matrix, recall_score, precision_score, f1_score,
cohen_kappa_score
from scipy.spatial.distance import pdist
from scipy.spatial.distance import cdist
from scipy.spatial import distance
from sklearn.metrics.pairwise import pairwise_distances
from sklearn.metrics import mean_squared_error
from functools import reduce
from time import process_time

#Tell GDAL to throw Python exceptions, and register all drivers
gdal.UseExceptions()
gdal.AllRegister()




#Part 1: Training the Random Forest model (with reference to sub
section 4.2.2 of the chapter on Methodology)

df = pd.read_csv ('Enter file name and path', sep = ',') #Read the
input csv file containing training points as a dataframe

X = df.drop (['XCoord', 'YCoord', 'OBJECTID', 'CLASS'],
axis=1).values#Create X matrix

y = df['CLASS'] #Create y array

# Split dataset into train set and test. The dataset is balanced in
such a way that all classes have the same number of points in train
and test set.
Stratified_Split = StratifiedShuffleSplit(n_splits = 4, test_size =
0.3, random_state = 0)
```

```
type (Stratified_Split.split (X,y))
for train_index, test_index in Stratified_Split.split (X,y):
    X_train, X_test = X [train_index], X [test_index]
    y_train, y_test = y [train_index], y [test_index]

#Find the best parameters for the model
ParameterRange = {'n_estimators': [10, 100, 200, 300, 400, 500, 750, 
1000], 'max_depth': [1, 2, 3, 4, 5, 6]}
Grid_RF = GridSearchCV(RandomForestClassifier(), ParameterRange, refit 
= True, verbose = 3)
Grid_RF.fit(X_train, y_train)
print(Grid_RF.best_params_)
print (Grid_RF.best_estimator_)

#Train the model using best parameters
rf = RandomForestClassifier (n_estimators=500, n_jobs=-
1,oob_score=True)
rf = rf.fit (X_train, y_train)

y_pred = rf.predict(X_test) #Predict the test data of train city

#Print accuracy report
print ("Confusion matrix of the test data:\n", confusion_matrix 
(y_test, y_pred))
print("Accuracy report of the test data:\n", 
classification_report(y_test, y_pred))
print ("The OOB score is {oob}%".format(oob=rf.oob_score_ * 100))
print ("Kappa coeffecient:\n", cohen_kappa_score (y_test, y_pred))

#Input the sixteen band raster WSF3D image
ras_img1 = 'Enter the file name and path'
im1 = gdal.Open(ras_img1, gdal.GA_ReadOnly)
row1 = im1.RasterYSize
col1 = im1.RasterXSize
n_bands1 = im1.RasterCount
geo_transform = im1.GetGeoTransform ()
projection = im1.GetProjectionRef ()

#Extract importance of sixteen bands and save it in an array
band_array = []
band_name = range(1,n_bands1 + 1)
for i, imp in zip(band_name, rf.feature_importances_):
    print('Band {i} Significance is {imp}'.format(i=i, imp=imp))
    band_array.append (imp)
print (band_array)




#Part 2: Determining the Area of Applicability (with reference to sub 
section 4.2.3 of the chapter on Methodology)

#Standardise the variables
mean = np.mean (X_train, axis=0) #Get mean value for each band
sd = np.std (X_train, axis=0)  #Get standard deviation for each band
sub = np.subtract (X_train, mean)
standardised_var = np.divide (sub, sd)  #Standardise the variables
np.set_printoptions(threshold=np.inf)
print (standardised_var)

#Weight the train data by their corresponding band weights
wtd_var = np.multiply (band_array, standardised_var)
```

```
print (wtd_var)

#Calculate distances
#Find pairwise Euclidean distances between train data of train city
pd_TrainData = pdist (wtd_var)  #Pairwise distances in weighted data
avg_pd = np.average (pd_TrainData)   #Average of pairwise distances

#Find pairwise Euclidean distances between train and test data of
train city
def distances(xy1, xy2):
    d0 = np.subtract.outer(xy1[:,0], xy2[:,0])
    d1 = np.subtract.outer(xy1[:,1], xy2[:,1])
    return np.hypot(d0, d1)
dist = distances (X_test, X_train)
min_dist = dist.min(axis=1)        #Minimum of the pairwise distances

#Calculate Dissimilarity Index for train city
DI = np.divide (min_dist, avg_pd)

#Calculate Area of Applicability by applying a threshold on DI
(Threshold = 75 percentile + 1.5 * IQR of DI)
Q1 = np.percentile(DI, 25, interpolation = 'midpoint')#quartile1 of DI
Q3 = np.percentile(DI, 75, interpolation = 'midpoint')#quartile3 of DI
IQR = Q3-Q1    #Inter Quartile Range of DI
percentile = np.quantile (DI, 0.75)    #75 percentile of DI values
threshold = (percentile + 1.5 * IQR)   #DI threshold
print (threshold)

#Use other city as test data now
df_NewCityAsTestData = pd.read_csv ('Enter file name and path', sep =
',')   # Read the input csv file containing training points as a
dataframe
X_Test_NewCity   =   df_NewCityAsTestData.drop   (['XCoord',   'YCoord',
'OBJECTID', 'CLASS'], axis=1).values  #Create X matrix
y_Test_NewCity = df_NewCityAsTestData ['CLASS']    #Create y array



#Find pairwise Euclidean distances of all points between train city
and validation city
def distances(xy1, xy2):
    d0 = np.subtract.outer(xy1[:,0], xy2[:,0])
    d1 = np.subtract.outer(xy1[:,1], xy2[:,1])
    return np.hypot(d0, d1)
dist_NewCity = distances (X_Test_NewCity, X_train)
min_dist_NewCity = dist_NewCity.min(axis=1)   #Minimum of the pairwise
distances



#Calculate Dissimilarity Index for validation city
DI_New = np.divide (min_dist_NewCity, avg_pd)



#Find the values which fall above DI threshold
output = np.where( DI_New > threshold)
output2 = np.asarray(output)   #Convert the tuple to an array


#Find the values which fall below DI threshold
output3 = np.where( DI_New < threshold)
```

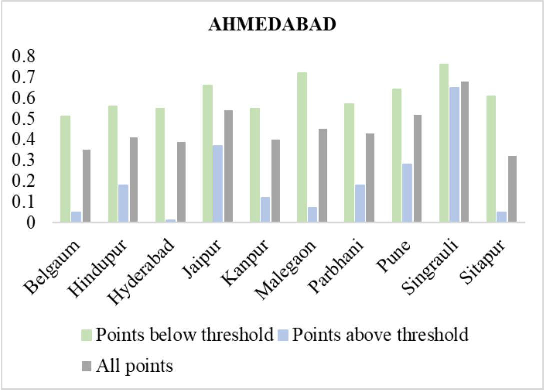
```
output4 = np.asarray(output3)  #Convert the tuple to an array



#Part 3: Spatially transferring the trained model (with reference to
sub section 4.2.4 of the chapter on Methodology)

#Transfer the RF model trained with train city to validation city
using only validation points that are above DI threshold

NewCityArray = df_NewCityAsTestData.to_numpy() #Convert dataframe to
array

NewCityArray2 = np.delete(NewCityArray, np.s_[0:3], axis=1) #Remove
first three columns from the array

NewCityArray_PointsAbove = np.delete(NewCityArray2, output4,
axis=0)#Remove the rows with indices below DI threshold

X_Test_NewCity_PointsAbove = NewCityArray_PointsAbove [:,
np.r_[1:17]]#Create X matrix

y_Test_NewCity_PointsAbove = NewCityArray_PointsAbove [:,
np.r_[:1]]#Create y array

Prediction_NewCity_PointsAbove = rf.predict
(X_Test_NewCity_PointsAbove) #Predict using the trained model

#Accuracy report
print ("Confusion matrix of the new city as test data:\n",
confusion_matrix (y_Test_NewCity_PointsAbove,
Prediction_NewCity_PointsAbove))
print("Accuracy report of the new city as test data:\n",
classification_report(y_Test_NewCity_PointsAbove,
Prediction_NewCity_PointsAbove))
print ("Kappa coeffecient:\n", cohen_kappa_score
(y_Test_NewCity_PointsAbove, Prediction_NewCity_PointsAbove))



#Transfer the RF model trained with train city to validation city
using only validation points that are below DI threshold
NewCityArray_PointsBelow = np.delete(NewCityArray2, output2,
axis=0)#Remove the rows with indices above DI threshold

X_Test_NewCity_PointsBelow = NewCityArray_PointsBelow [:,
np.r_[1:17]]#Create X matrix
y_Test_NewCity_PointsBelow = NewCityArray_PointsBelow [:,
np.r_[:1]]#Create y array

Prediction_NewCity_PointsBelow = rf.predict
(X_Test_NewCity_PointsBelow)  #Predict using the trained model

#Accuracy report
print ("Confusion matrix of the new city as test data:\n",
confusion_matrix (y_Test_NewCity_PointsBelow,
Prediction_NewCity_PointsBelow))
print("Accuracy report of the new city as test data:\n",
classification_report(y_Test_NewCity_PointsBelow,
Prediction_NewCity_PointsBelow))
```

```
print        ("Kappa        coeffecient:\n",        cohen_kappa_score
(y_Test_NewCity_PointsBelow, Prediction_NewCity_PointsBelow))



#Transfer the RF model trained with with train city to validation city
using all points of the validation city

prediction_NewCity3 = rf.predict (X_Test_NewCity) #Predict

#Accuracy report
print  ("Confusion  matrix  of  the  test  data:\n",  confusion_matrix
(y_Test_NewCity, prediction_NewCity3))
print("Accuracy      report      of      the      test      data:\n",
classification_report(y_Test_NewCity, prediction_NewCity3))
#print  ("The  OOB  score  is  {oob}%".format(oob  =  Grid_RF.oob_score_  *
100))
print  ("Kappa  coeffecient:\n",  cohen_kappa_score  (y_Test_NewCity,
prediction_NewCity3))




#Part  4: Assessing the transferability performances(with reference to
sub section 4.2.5 of the chapter on Methodology)

#Find similarity index for all points of the validation city

#Overlap coefficient
l1 = len (X_train)
l2 = len (X_Test_NewCity)
arrays_intersection = np.intersect1d (X_train, X_Test_NewCity)
overlap_coefficient = len (arrays_intersection) / np.minimum (l1, l2)
print (overlap_coefficient)

#Histogram Intersection
hist_1, _ = np.histogram (X_train, bins=100, range=[-15, 15])
hist_2, _ = np.histogram (X_Test_NewCity, bins=100, range=[-15, 15])
def return_intersection (hist_1, hist_2):
    minima = np.minimum (hist_1, hist_2)
    intersection = np.true_divide (np.sum(minima), np.sum(hist_2))
    return intersection
print (return_intersection (hist_1, hist_2))



#Find similarity index for validation points below threshold

#Overlap coefficient
l1 = len (X_train)
l2 = len (X_Test_NewCity_PointsBelow)
arrays_intersection         =         np.intersect1d         (X_train,
X_Test_NewCity_PointsBelow)
overlap_ coefficient = len (arrays_intersection) / np.minimum (l1, l2)
print (overlap_ coefficient)


#Histogram Intersection
hist_1, _ = np.histogram (X_train, bins=100, range=[-15, 15])
hist_2,   _   =   np.histogram   (X_Test_NewCity_PointsBelow,   bins=100,
range=[-15, 15])
def return_intersection (hist_1, hist_2):
    minima = np.minimum (hist_1, hist_2)
    intersection = np.true_divide (np.sum(minima), np.sum(hist_2))
    return intersection

print (return_intersection (hist_1, hist_2))



#Part 5: Conditional classification (with reference to sub section
4.2.6 of the chapter on Methodology)

#Input the sixteen band raster WSF3D image of the validation city
proven suitable for model transferability

ras_img_test = 'Enter file name and path'
im = gdal.Open(ras_img_test, gdal.GA_ReadOnly)
row_test = im.RasterYSize
col_test = im.RasterXSize
n_bands_test = im.RasterCount
geo_transform = im.GetGeoTransform ()
projection = im.GetProjectionRef ()
print (row_test)
print (col_test)

#Convert image to array
im_array_test = []
for i in range (1, n_bands_test + 1):
    band = im.GetRasterBand(i)
    im_array_test.append(band.ReadAsArray ())

#Stack the array
array_new_test = np.stack (im_array_test, axis = 2)
print ("Shape of stacked array is
{sz}".format(sz=array_new_test.shape))

#Reshape the image array from 3d to 2d
array_new_shp_test = (array_new_test.shape[0] *
array_new_test.shape[1], array_new_test.shape[2])
array_new_rshp_test = array_new_test [:, :,
:].reshape(array_new_shp_test)
print("Array is now reshaped from {o} to {n}".format(o =
array_new_test.shape, n = array_new_rshp_test.shape))



#Create binary image showing Area of Applicability
#Find pairwise Euclidean distances of all points between train city
and validation city
def distances(xy1, xy2):
    d0 = np.subtract.outer(xy1[:,0], xy2[:,0])
    d1 = np.subtract.outer(xy1[:,1], xy2[:,1])
    return np.hypot(d0, d1)
dist_NewCityImg = distances (array_new_rshp_test, X_train)
min_dist_NewCityImg = dist_NewCityImg.min(axis=1) #Minimum of the
pairwise Euclidean distances


#Calculate DI
DI_NewImg = np.divide (min_dist_NewCityImg, avg_pd)
```

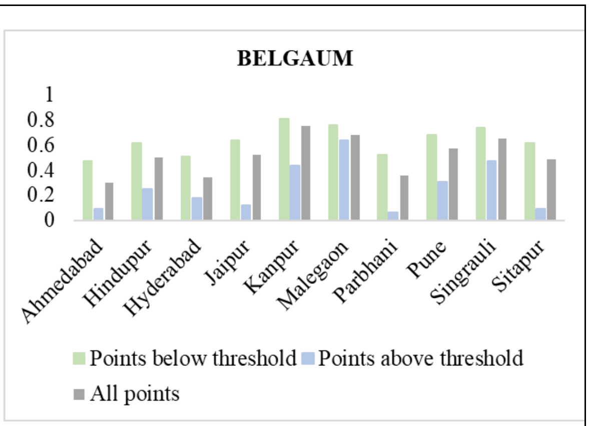
```
#Find the values which fall below threshold (RF could be successfully
applied here)
outputImg = np.where( DI_NewImg < threshold)

#Assign the value of "1" to only those pixels with DI value below
threshold and "0" for those above
outputImg = np.where( DI_NewImg < threshold, 1, 0)

#Reshape the output array to original raster extent
outputImg_rshp = outputImg.reshape(array_new_test [:, :, 0].shape)
print ("Dimension of classified map is {o}". format(o =
outputImg_rshp.shape))

#Display the binary image
Output_BinaryImg = 'Enter file name and path'
def createGeotiff (Output_BinaryImg, X, geo_transform, projection):
    driver = gdal.GetDriverByName('GTiff')
    row, col = X.shape
    rasterDS = driver.Create (Output_BinaryImg, col, row, 1,
                                                         gdal.GDT_Byte)
    rasterDS.SetGeoTransform (geo_transform)
    rasterDS.SetProjection (projection)
    band = rasterDS.GetRasterBand (1)
    band.WriteArray (X)
    rasterDS = None
    createGeotiff (Output_BinaryImg, outputImg_rshp, geo_transform,
projection)

#Create final classification image by transferring the trained model

#Predict the image
im_pred_FinalImg = rf.predict (array_new_rshp_test)

#Reshape the classified map to original raster extent
im_pred_FinalImg_rshp = im_pred_FinalImg.reshape(array_new_test [:, :,
                                                             0].shape)
print ("Dimension of classified map is {o}". format(o =
im_pred_FinalImg_rshp.shape))

#Display the final classification image
Final_Img = 'Enter file name and path'
def createGeotiff (Final_Img, X, geo_transform, projection):
    driver = gdal.GetDriverByName('GTiff')
    row, col = X.shape
    rasterDS = driver.Create (Final_Img, col, row, 1, gdal.GDT_Byte)
    rasterDS.SetGeoTransform (geo_transform)
    rasterDS.SetProjection (projection)
    band = rasterDS.GetRasterBand (1)
    band.WriteArray (X)
    rasterDS = None
createGeotiff (Final_Img, im_pred_FinalImg_rshp, geo_transform,
projection)
```

Appendix 2

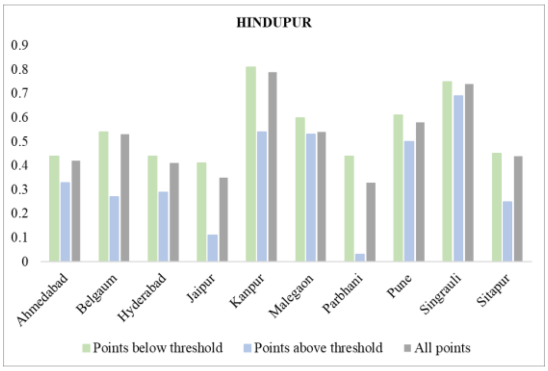
Kappa scores of cities for points below threshold, above threshold and all points



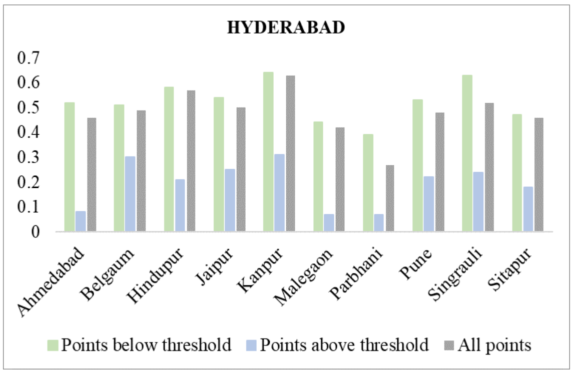
a) Model trained in Ahmedabad



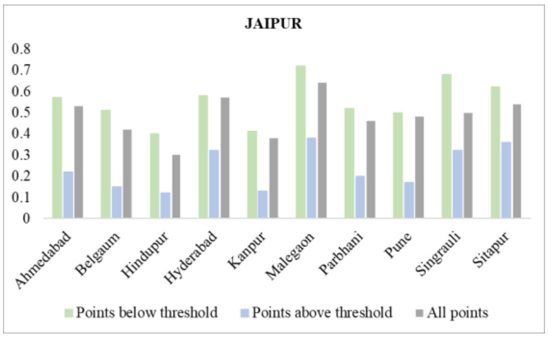
b) Model trained in Belgaum



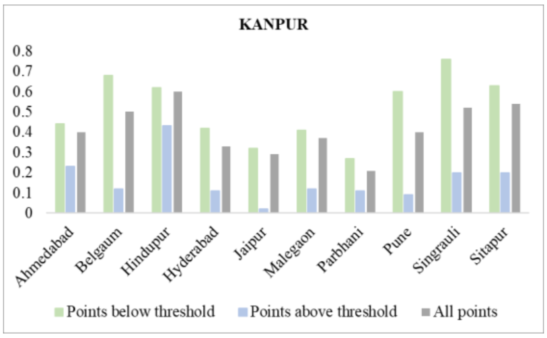
c) Model trained in Hindupur



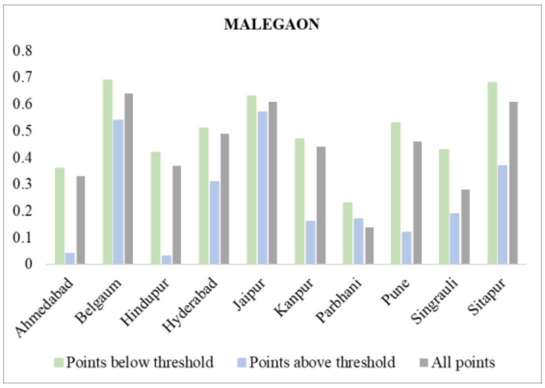
d) Model trained in Hyderabad



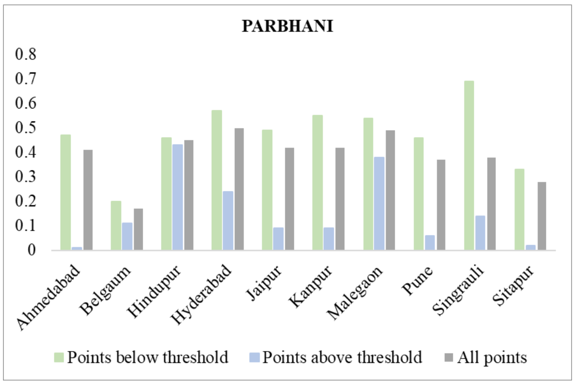
e) Model trained in Jaipur



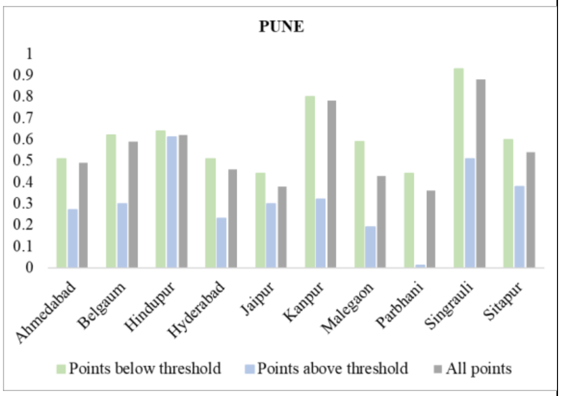
f) Model trained in Kanpur



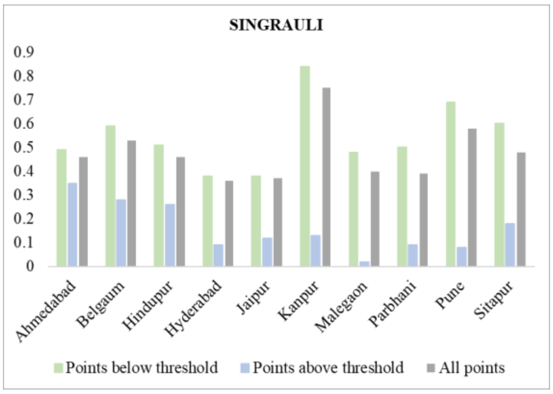
g) Model trained in Malegaon



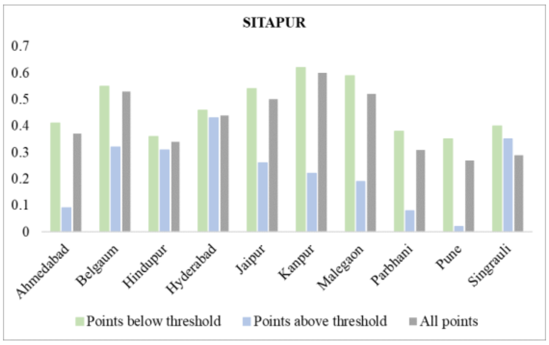
h) Model trained in Parbhani



i) Model trained in Pune



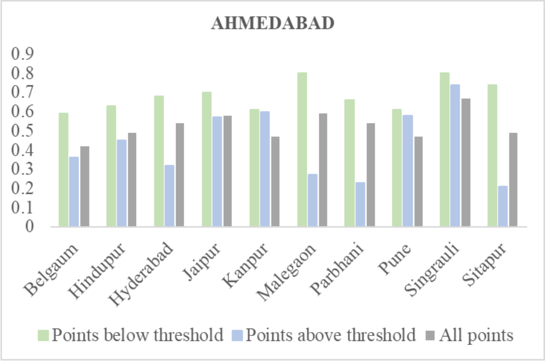
j) Model trained in Singrauli



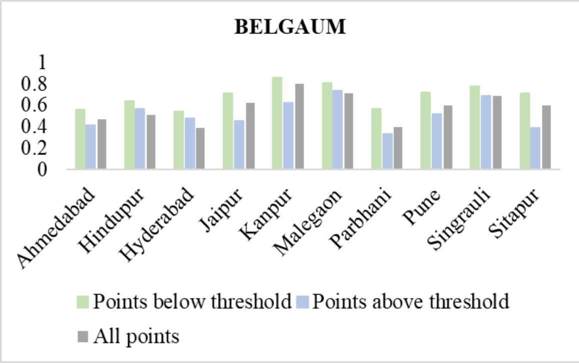
k) Model trained in Sitapur

Appendix 3

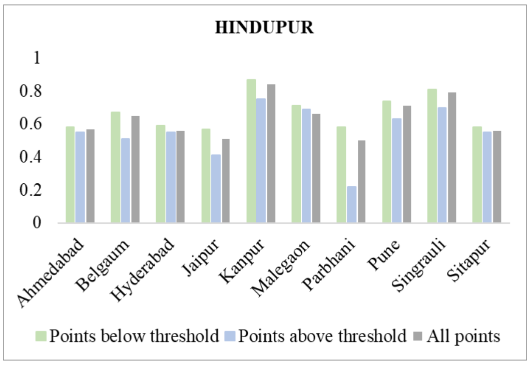
Overall accuracies of cities for points below threshold, above threshold and all points



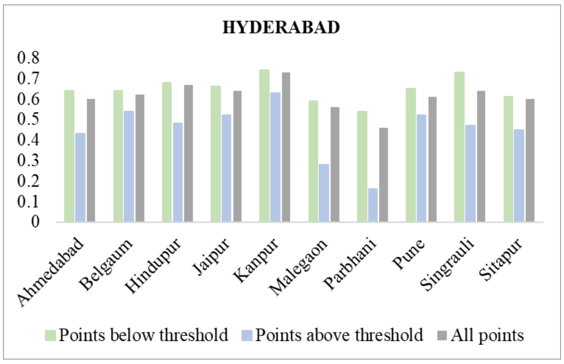
a) Model trained in Ahmedabad



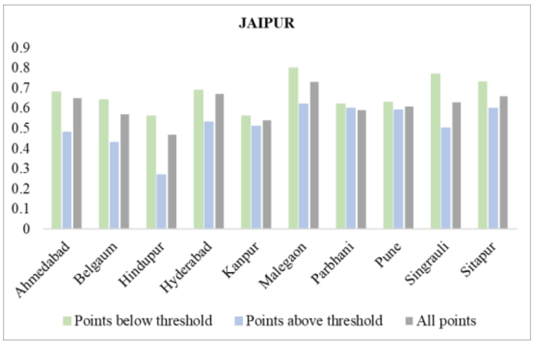
b) Model trained in Belgaum



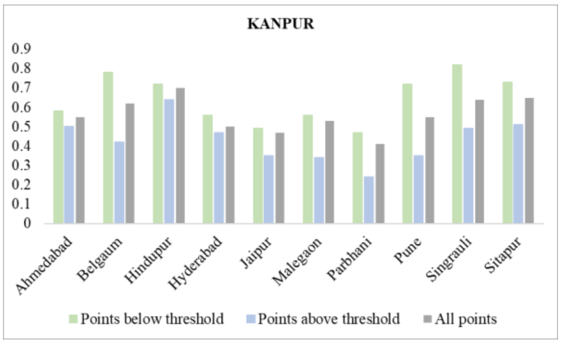
c) Model trained in Hindupur



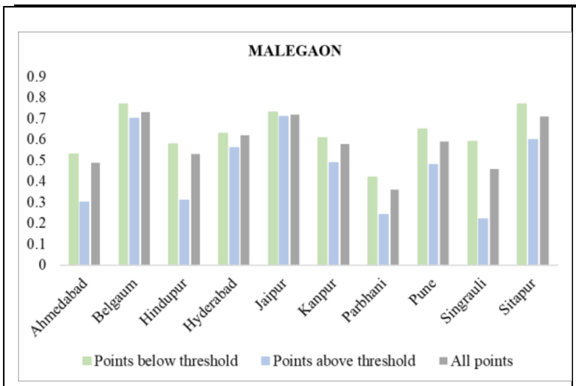
d) Model trained in Hyderabad



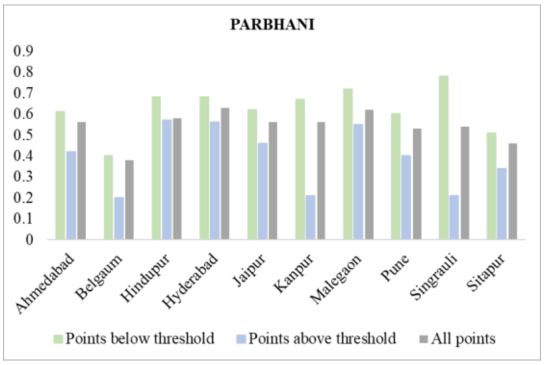
e) Model trained in Jaipur



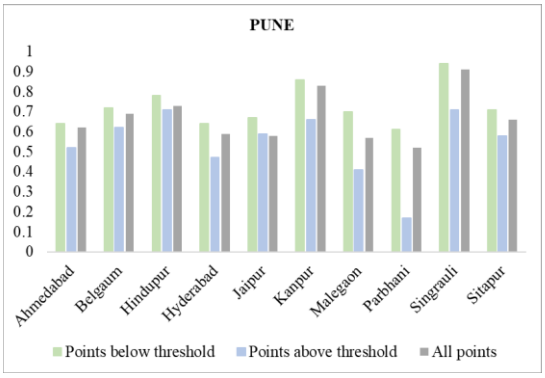
f) Model trained in Kanpur



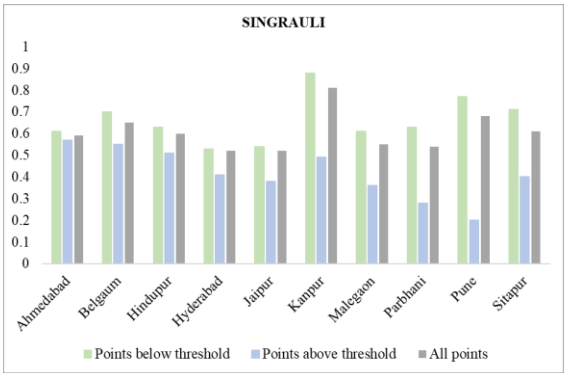
g) Model trained in Malegaon



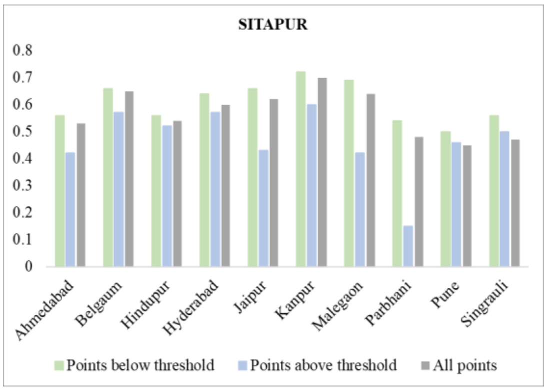
h) Model trained in Parbhani



i) Model trained in Pune



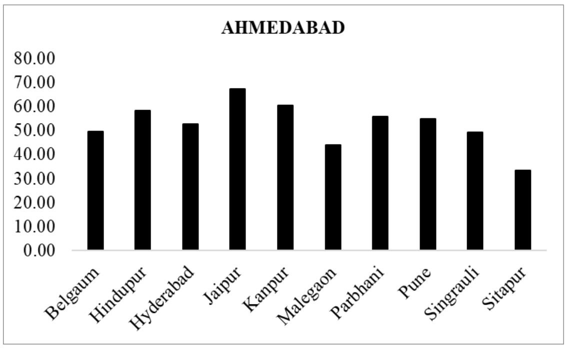
j) Model trained in Singrauli



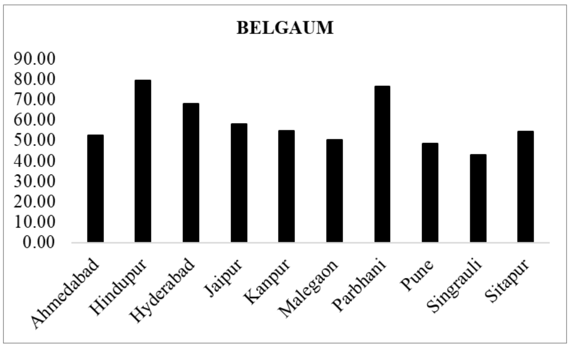
k) Model trained in Sitapur

Appendix 4

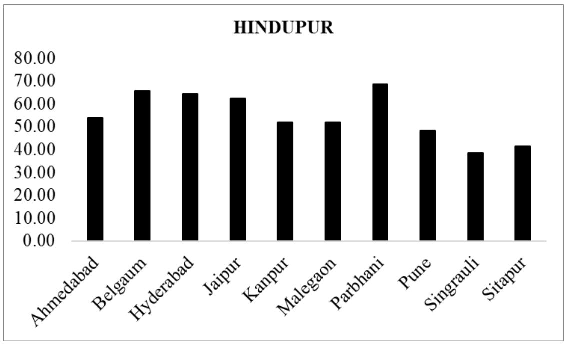
Percentage of validation points below threshold



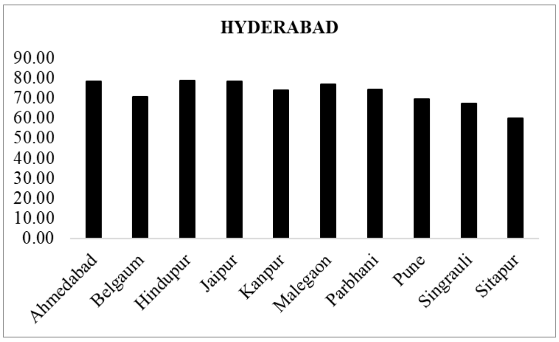
a) Model trained in Ahmedabad



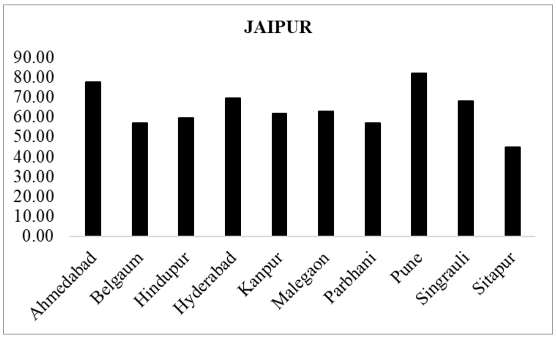
b) Model trained in Belgaum



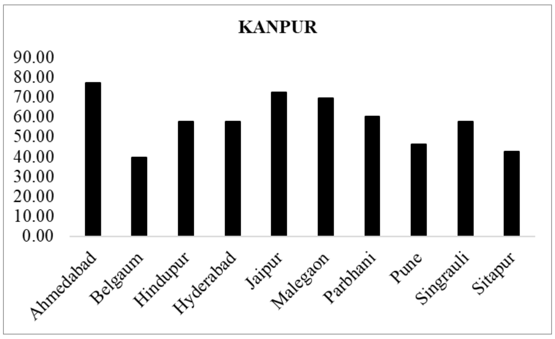
c) Model trained in Hindupur



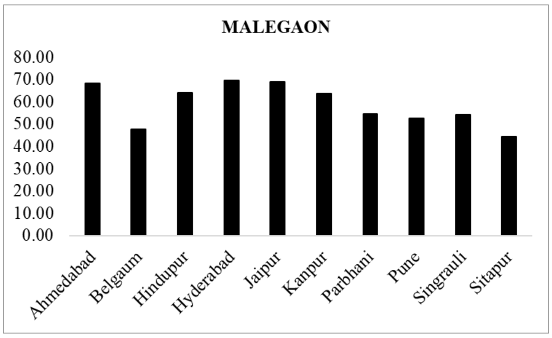
d) Model trained in Hyderabad



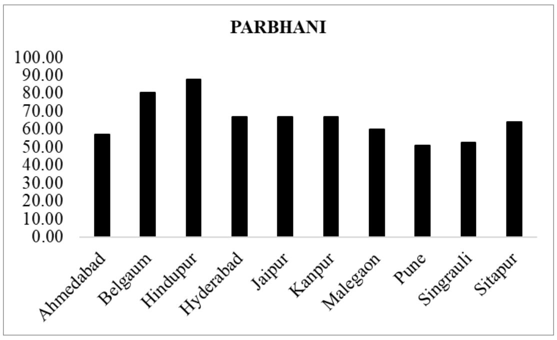
e) Model trained in Jaipur



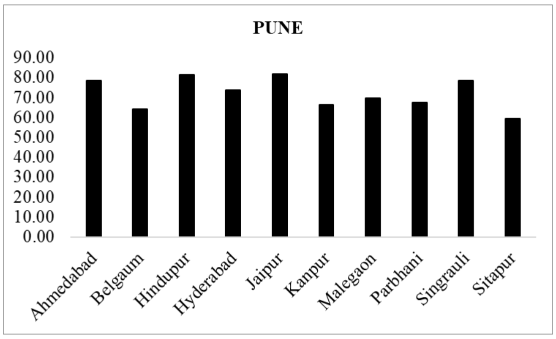
f) Model trained in Kanpur



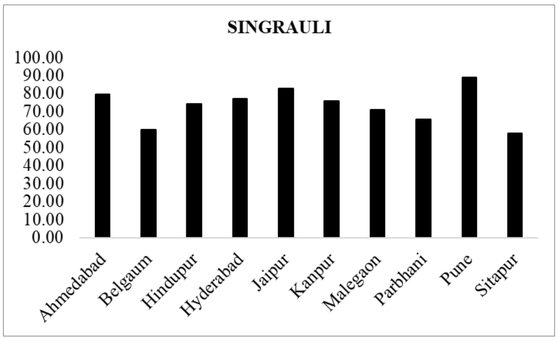
g) Model trained in Malegaon



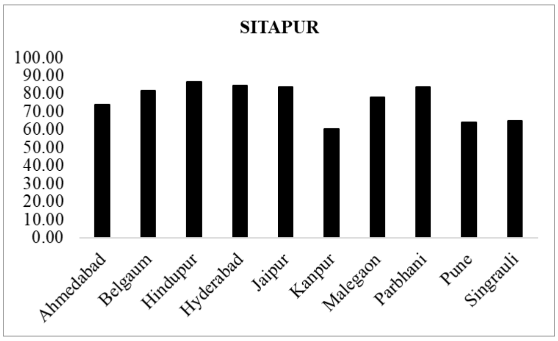
h) Model trained in Parbhani



i) Model trained in Pune



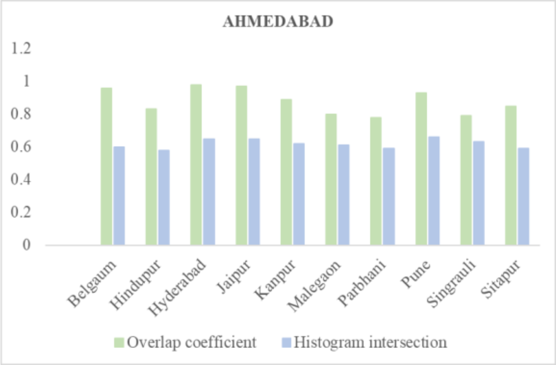
j) Model trained in Singrauli



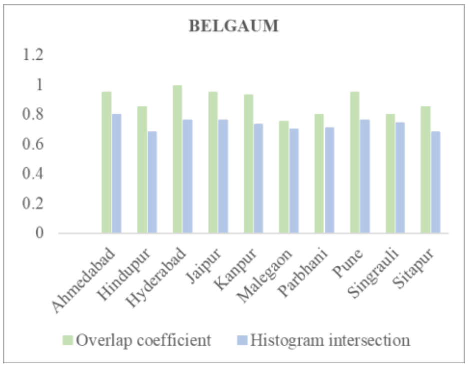
k) Model trained in Sitapur

Appendix 5

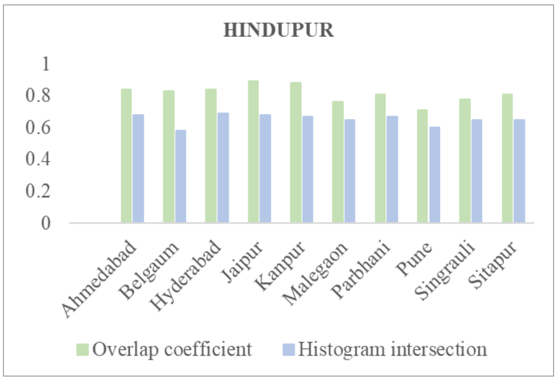
Overlap coefficient and histogram intersection of all points of validation city



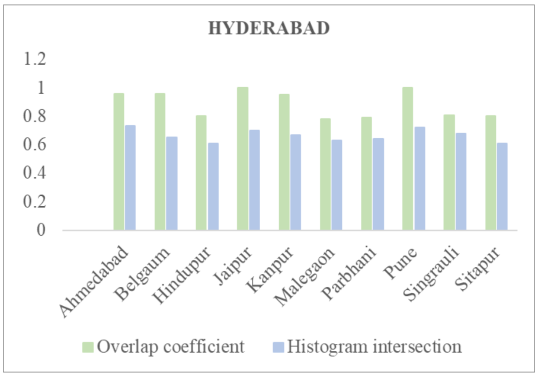
a) Model trained in Ahmedabad



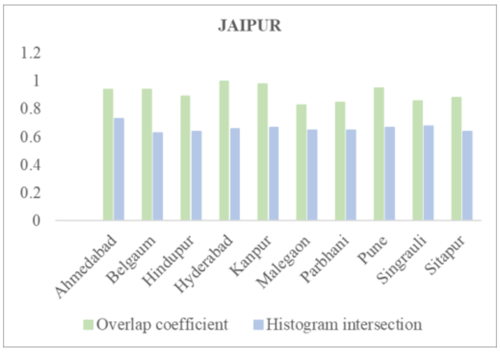
b) Model trained in Belgaum



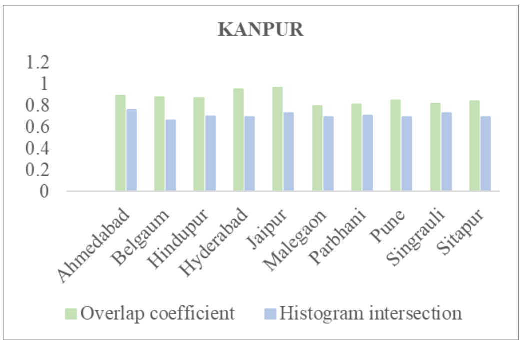
c) Model trained in Hindupur



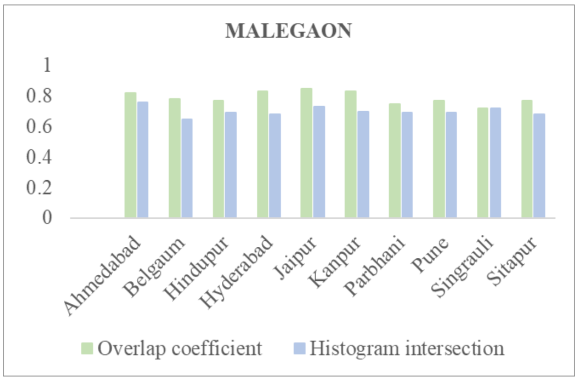
d) Model trained in Hyderabad



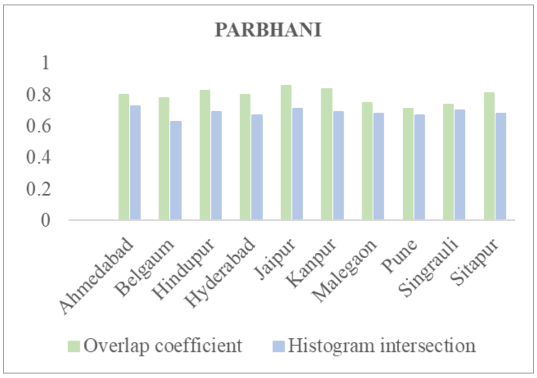
e) Model trained in Jaipur



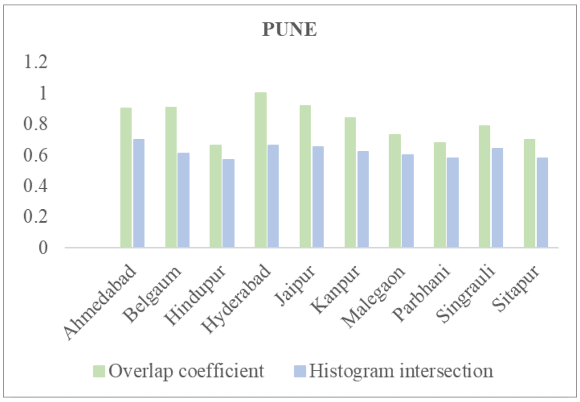
f) Model trained in Kanpur



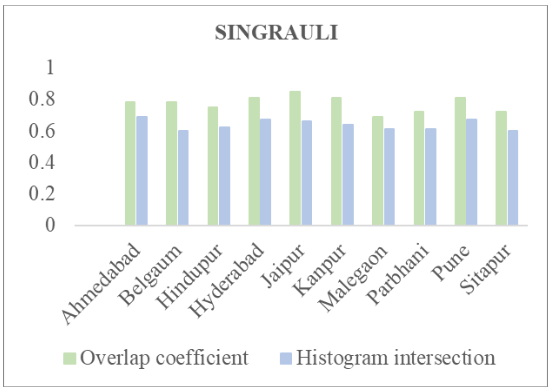
g) Model trained in Malegaon



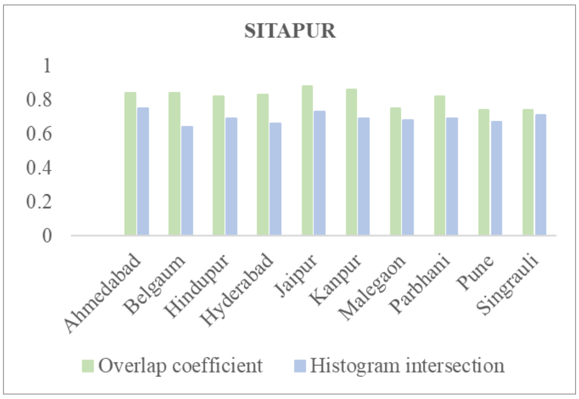
h) Model trained in Parbhani



i) Model trained in Pune



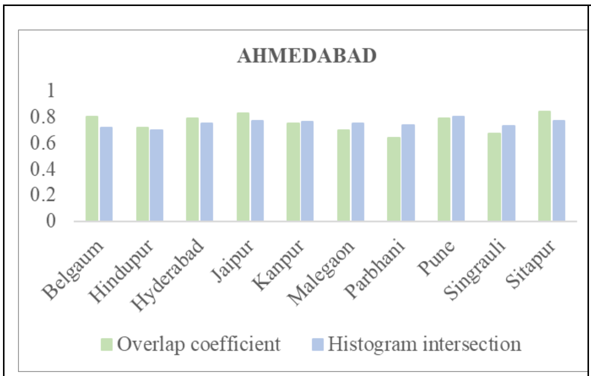
j) Model trained in Singrauli



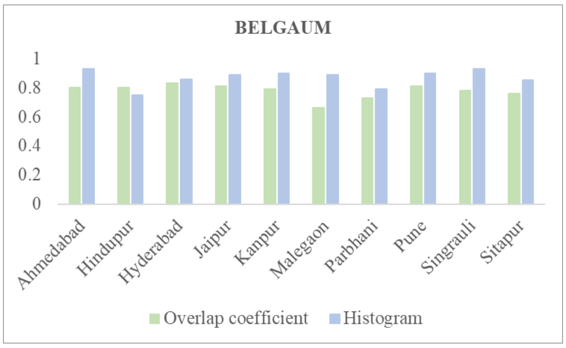
k) Model trained in Sitapur

Appendix 6

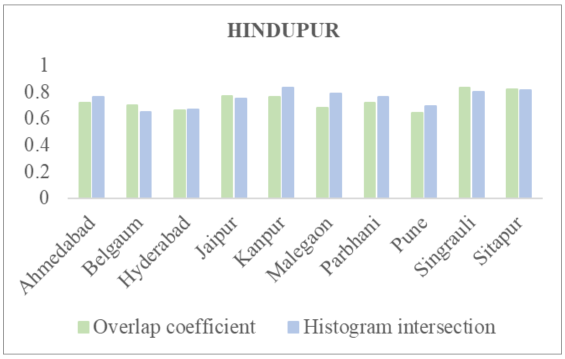
Overlap coefficient and histogram intersection of validation points below threshold



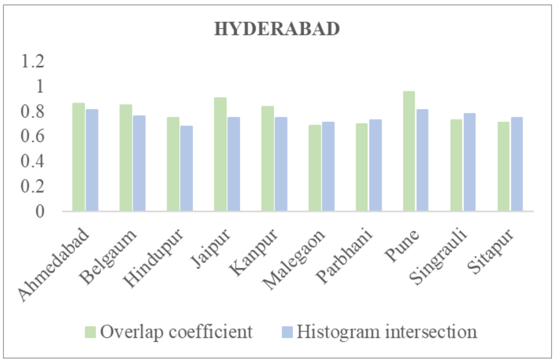
a) Model trained in Ahmedabad



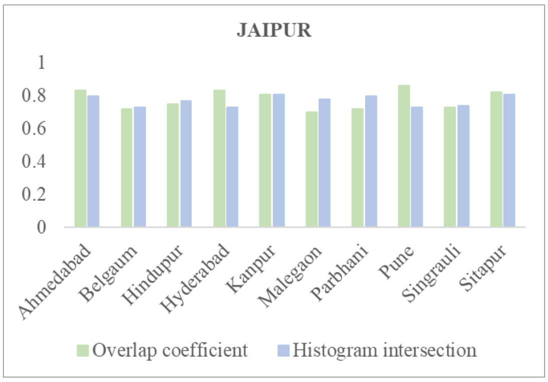
b) Model trained in Belgaum



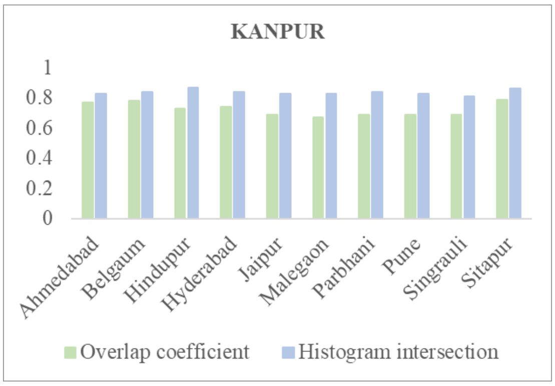
c) Model trained in Hindupur



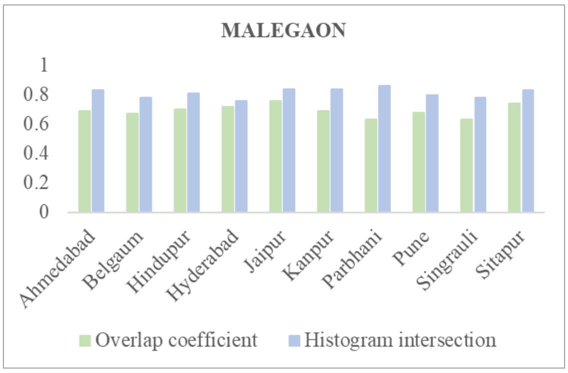
d) Model trained in Hyderabad



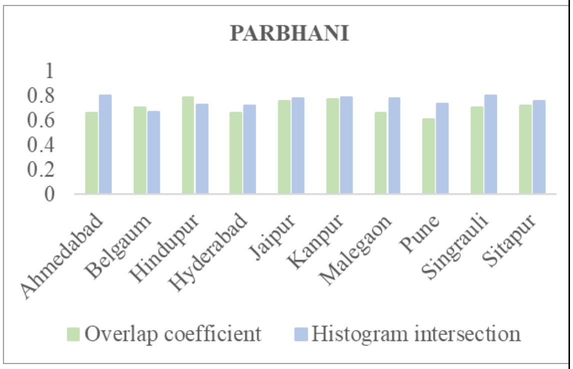
e) Model trained in Jaipur



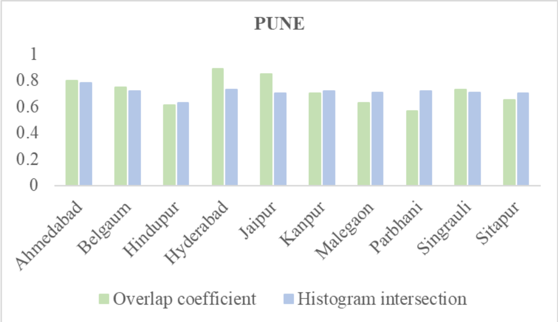
f) Model trained in Kanpur



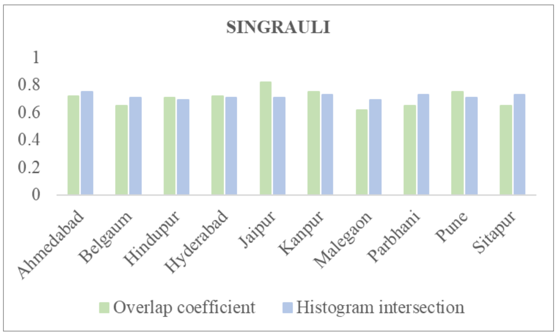
g) Model trained in Malegaon



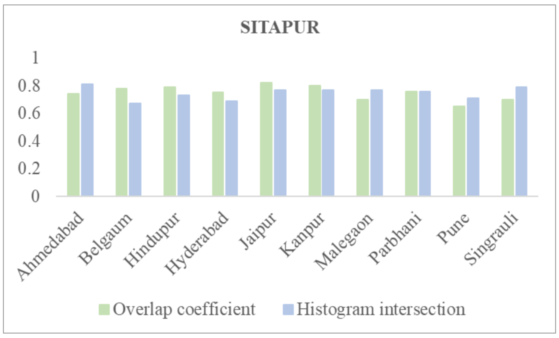
h) Model trained in Parbhani



i) Model trained in Pune



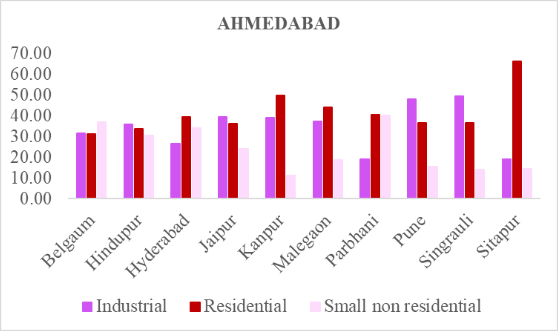
j) Model trained in Singrauli



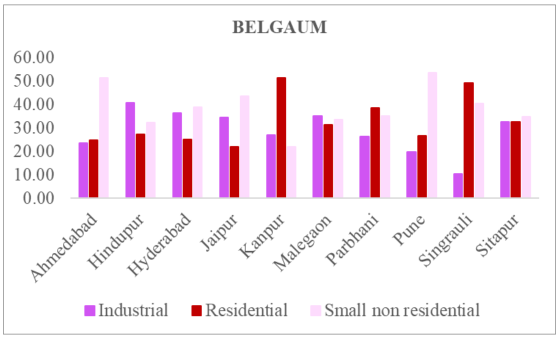
k) Model trained in Sitapur

Appendix 7

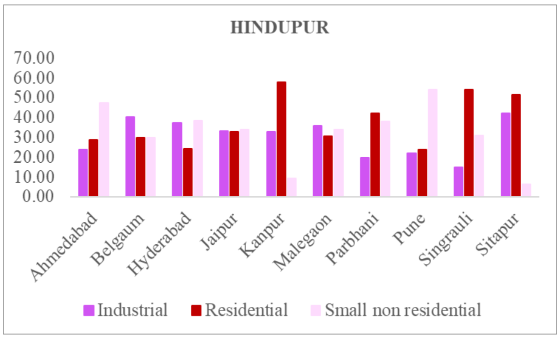
Proportion of individual class samples for validation cities



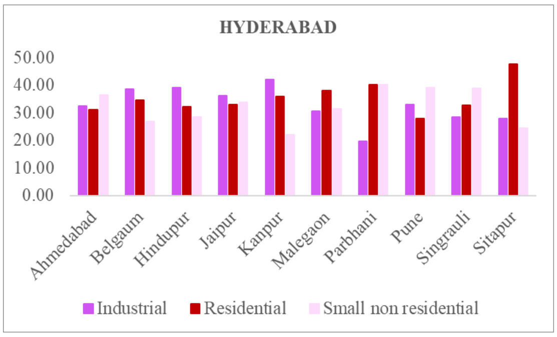
a) Model trained in Ahmedabad



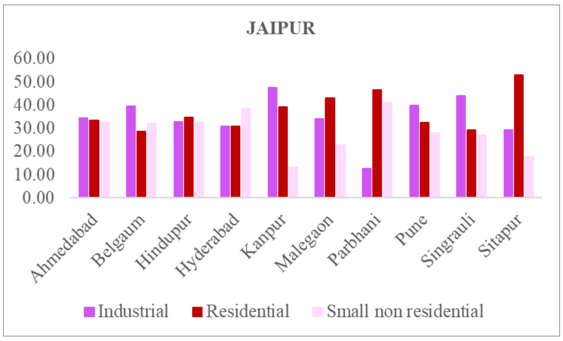
b) Model trained in Belgaum



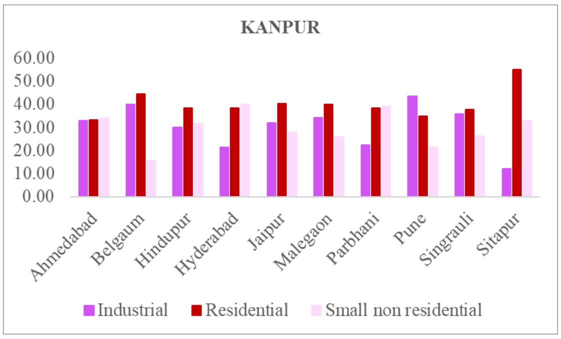
c) Model trained in Hindupur



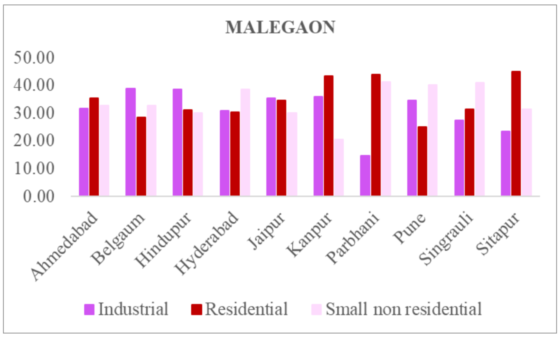
d) Model trained in Hyderabad



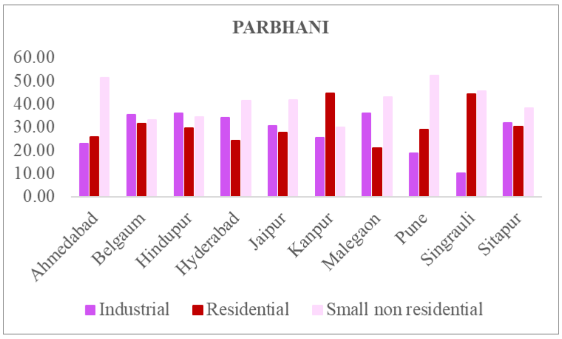
e) Model trained in Jaipur



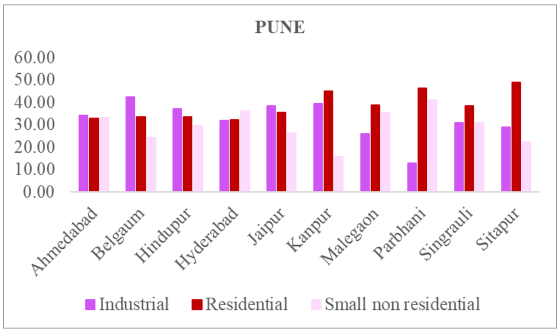
f) Model trained in Kanpur



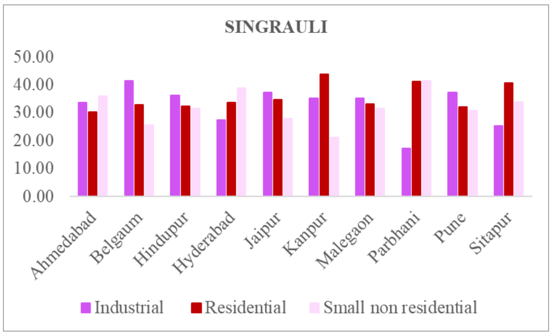
g) Model trained in Malegaon



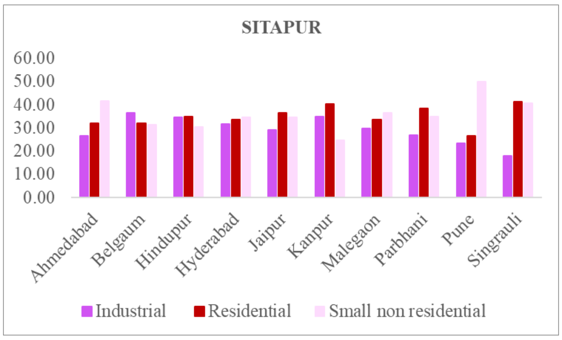
h) Model trained in Parbhani



i) Model trained in Pune



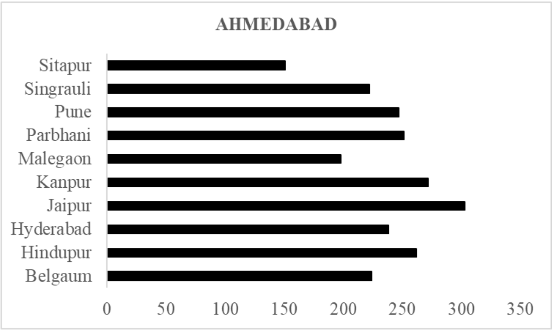
j) Model trained in Singrauli



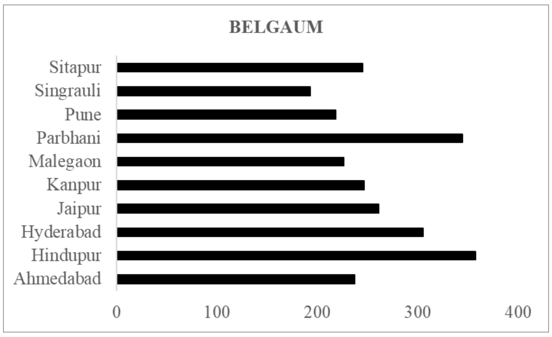
k) Model trained in Sitapur

Appendix 8

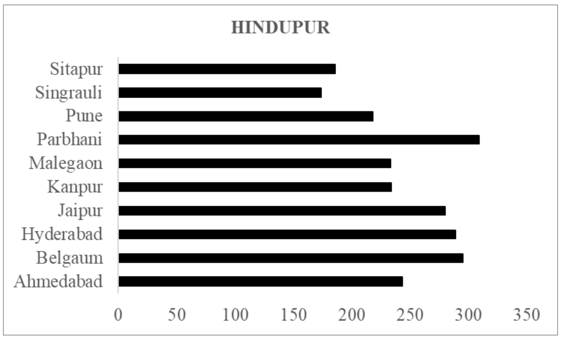
Number of points below threshold for validation cities



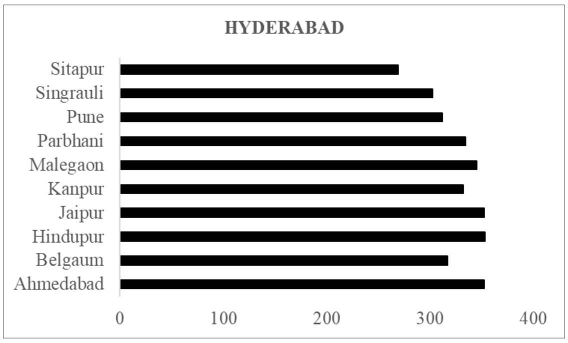
a) Model trained in Ahmedabad



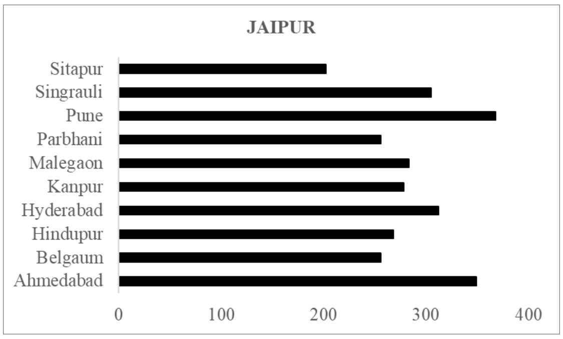
b) Model trained in Belgaum



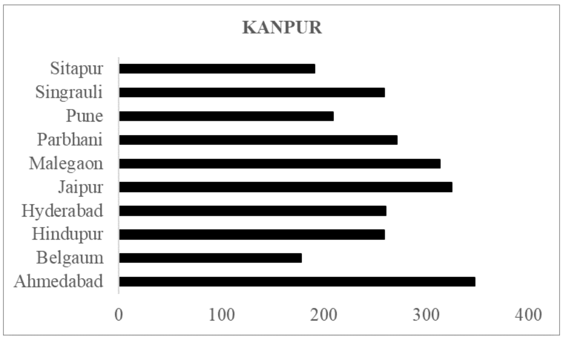
c) Model trained in Hindupur



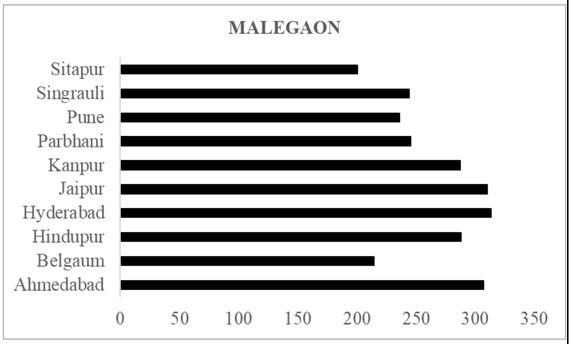
d) Model trained in Hyderabad



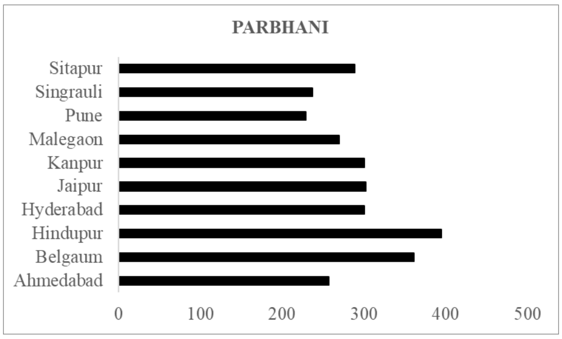
e) Model trained in Jaipur



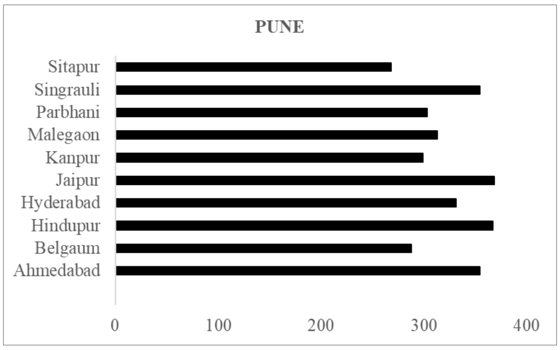
f) Model trained in Kanpur



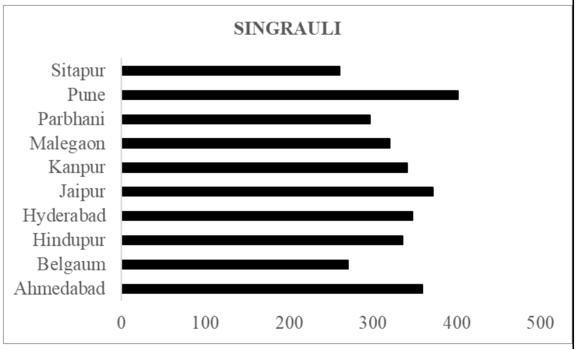
g) Model trained in Malegaon



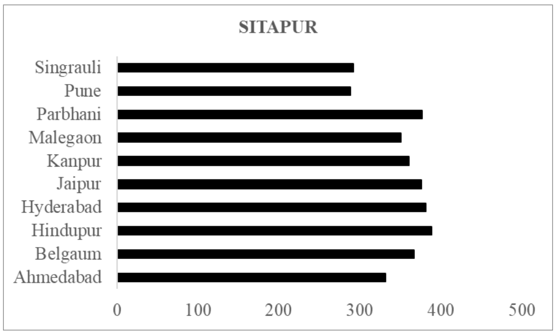
h) Model trained in Parbhani



i) Model trained in Pune



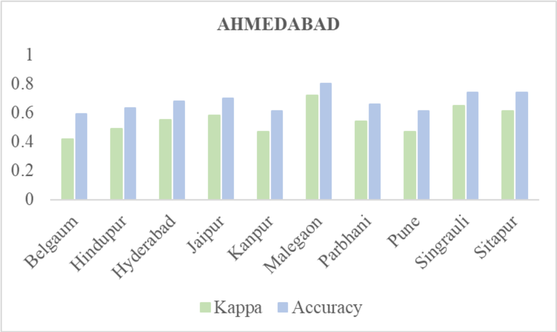
j) Model trained in Singrauli



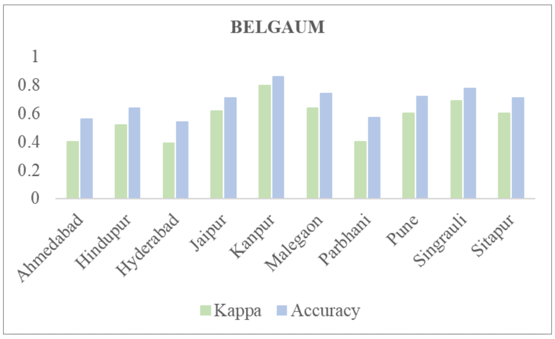
k) Model trained in Sitapur

Appendix 9

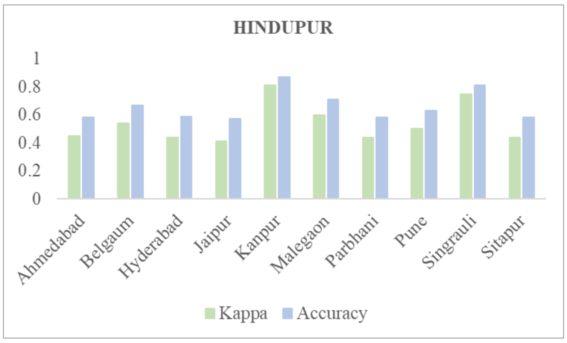
Prediction accuracies for validation points below threshold



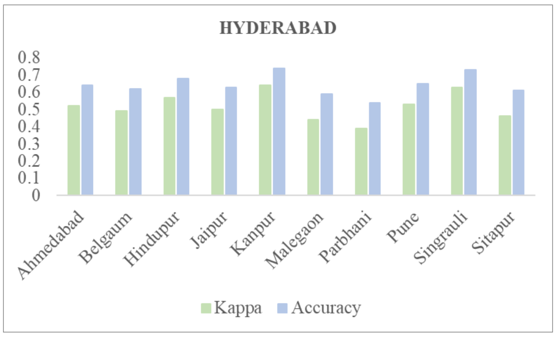
a) Model trained in Ahmedabad



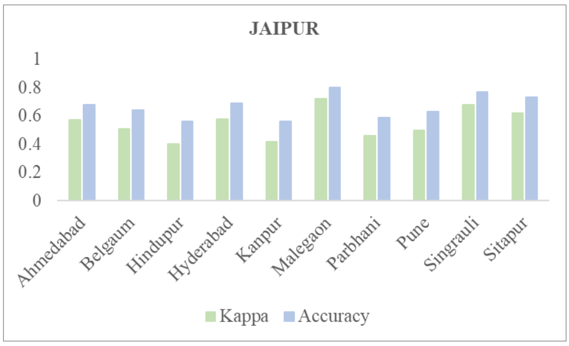
b) Model trained in Belgaum



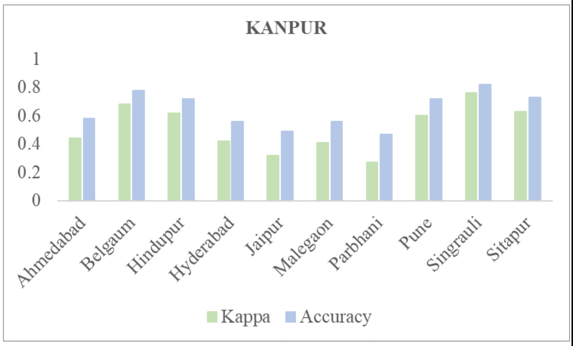
c) Model trained in Hindupur



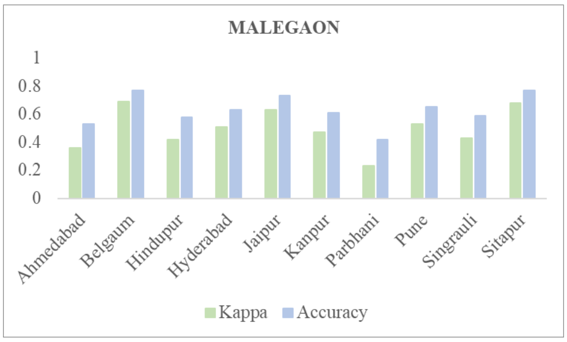
d) Model trained in Hyderabad



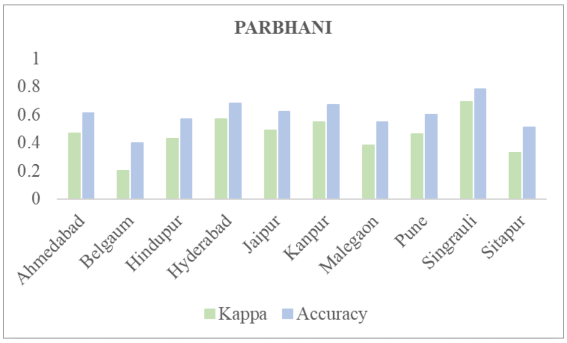
e) Model trained in Jaipur



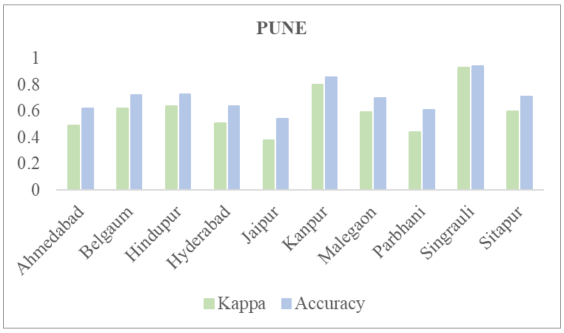
f) Model trained in Kanpur



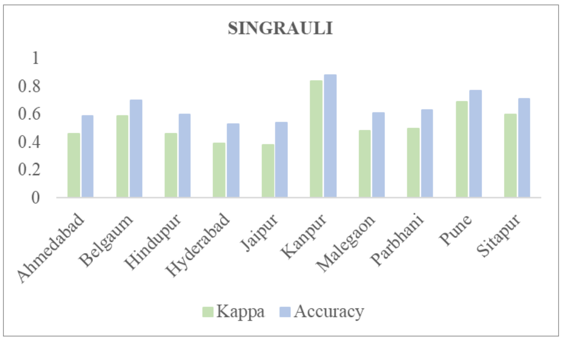
g) Model trained in Malegaon



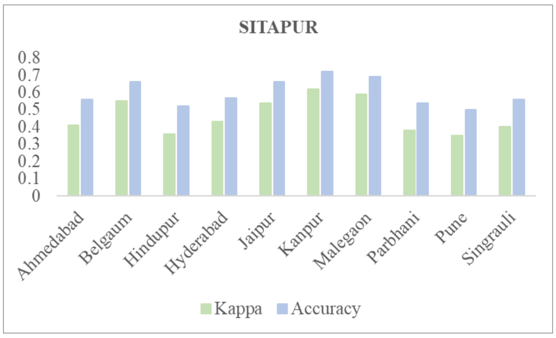
h) Model trained in Parbhani



i) Model trained in Pune



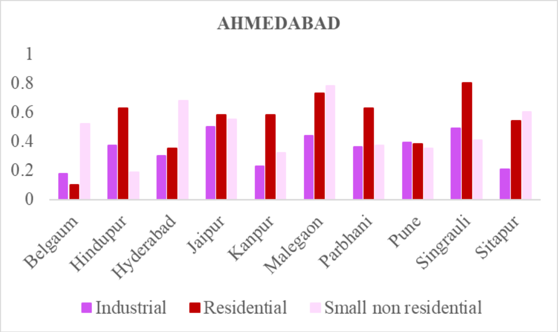
j) Model trained in Singrauli



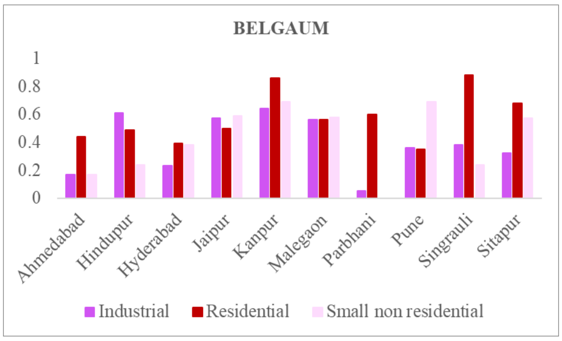
k) Model trained in Sitapur

Appendix 10

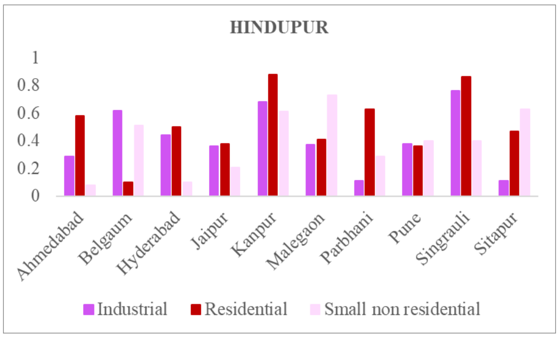
F1 scores of individual classes of validation cities



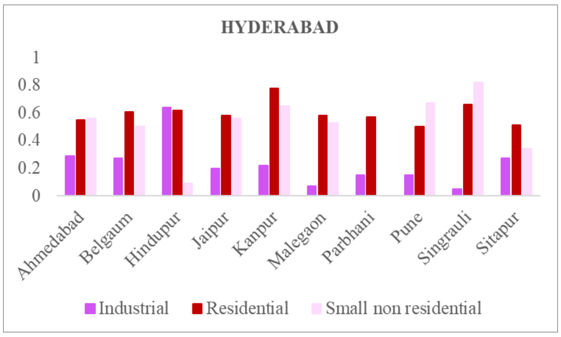
a) Model trained in Ahmedabad



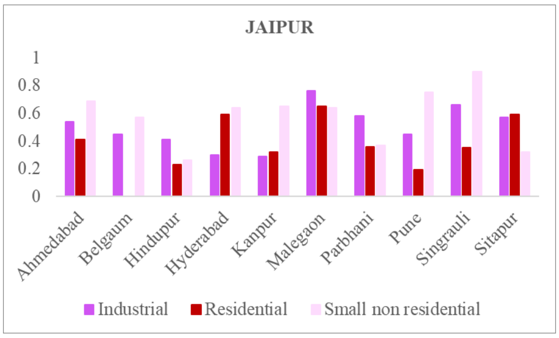
b) Model trained in Belgaum



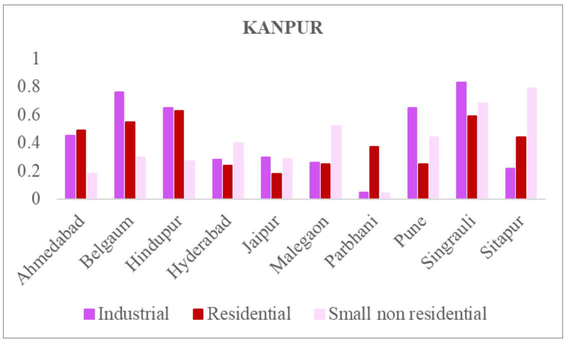
c) Model trained in Hindupur



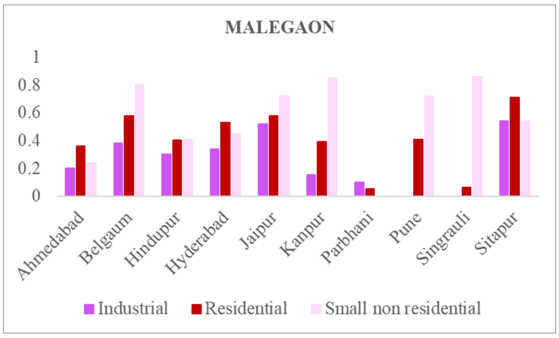
d) Model trained in Hyderabad



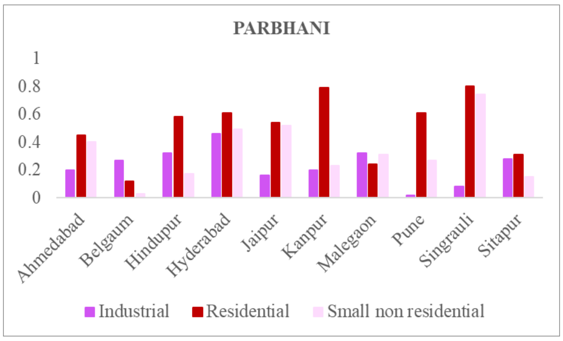
e) Model trained in Jaipur



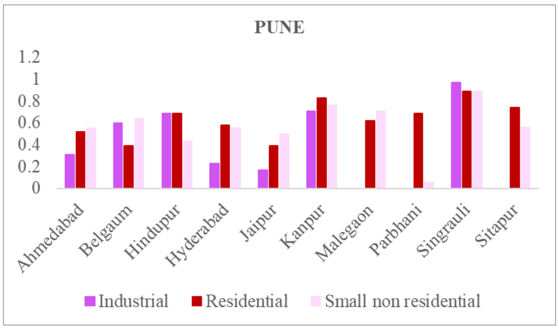
f) Model trained in Kanpur



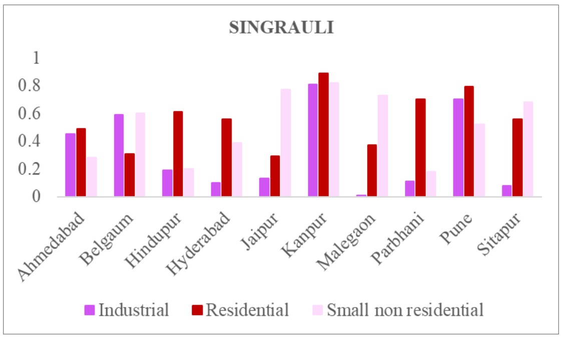
g) Model trained in Malegaon



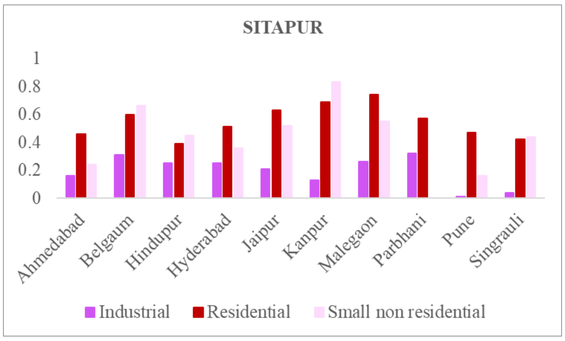
h) Model trained in Parbhani



i) Model trained in Pune



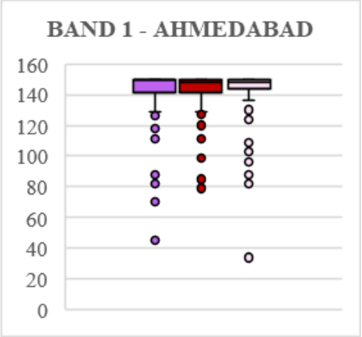
j) Model trained in Singrauli



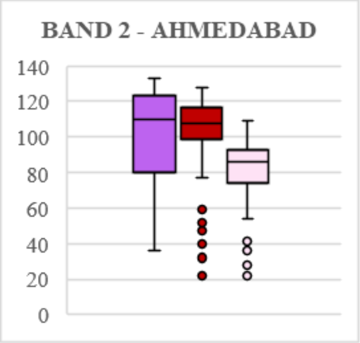
k) Model trained in Sitapur

Appendix 11

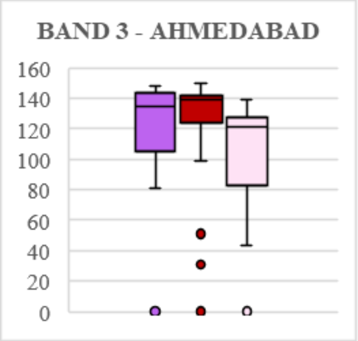
Class separability analysis - Ahmedabad



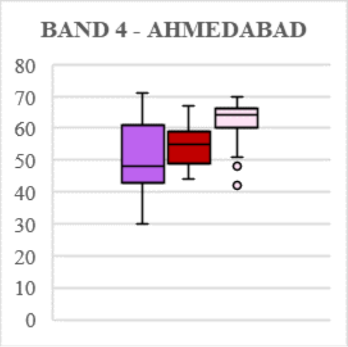
a) Mean (area)



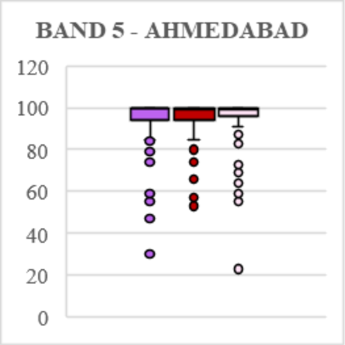
b) Median (area)



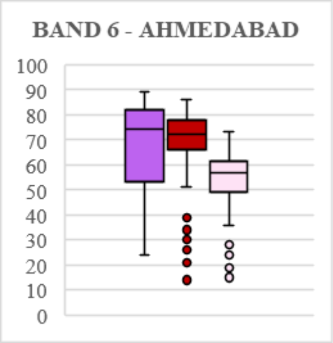
c) Standard deviation (area)



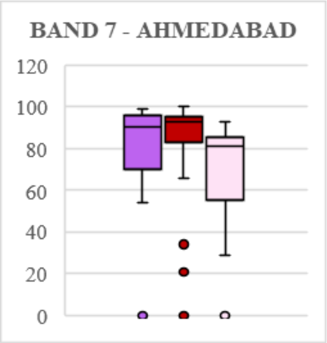
d) Area



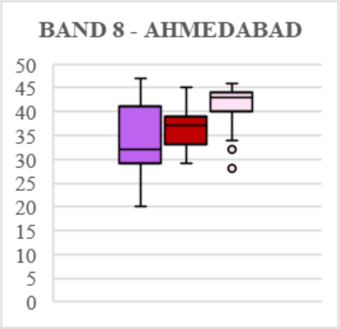
e) Mean (fraction)



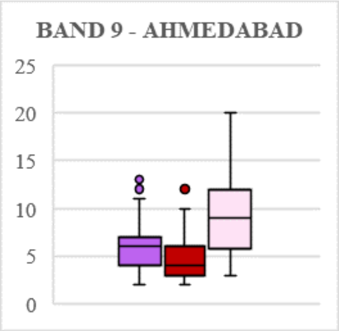
f) Median (fraction)



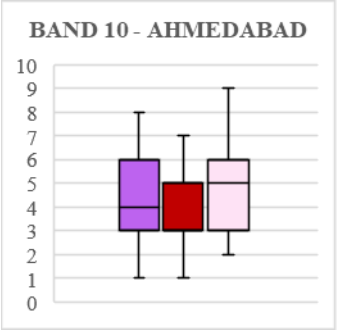
g) Standard deviation (fraction)



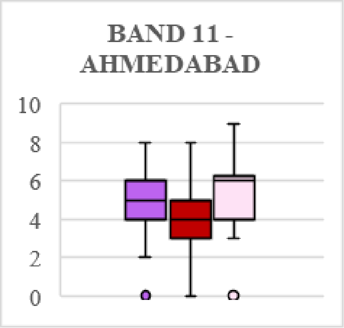
h) Fraction



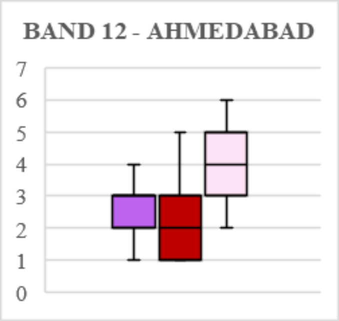
i) Mean (height)



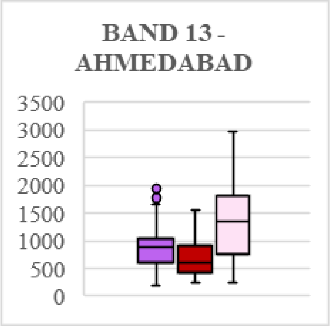
j) Median (height)



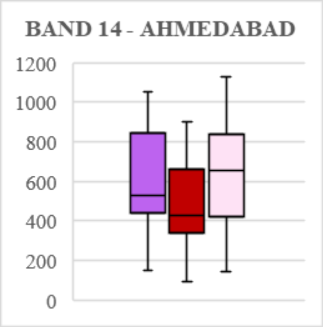
k) Standard deviation (height)



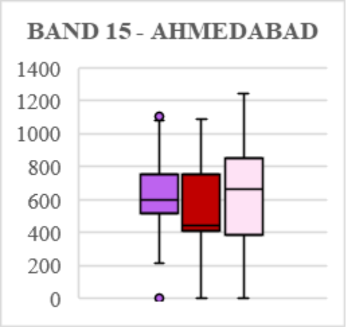
l) Height



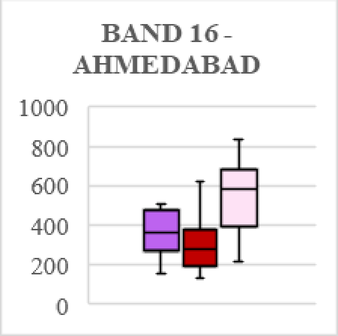
m) Mean (volume)



n) Median (volume)

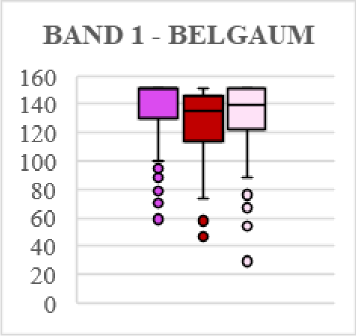


o) Standard deviation (volume)

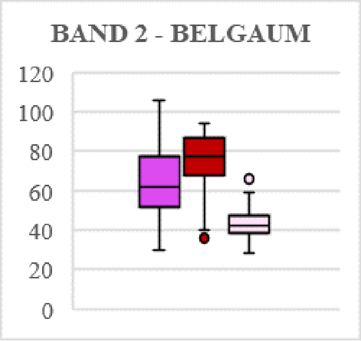


p) Volume

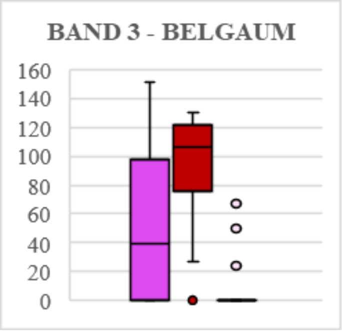
Class separability analysis - Belgaum



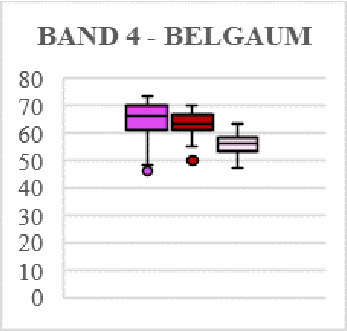
a) Mean (area)



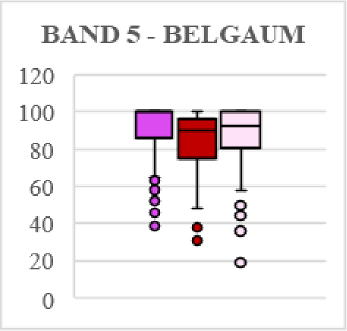
b) Median (area)



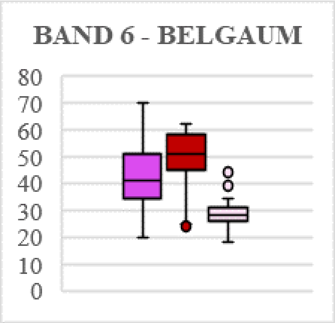
c) Standard deviation (area)



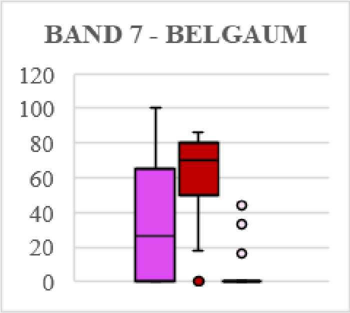
d) Area



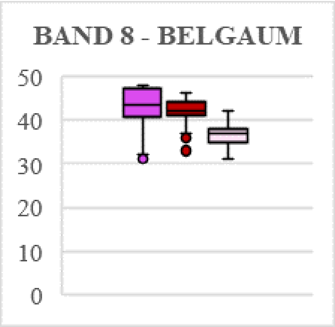
e) Mean (fraction)



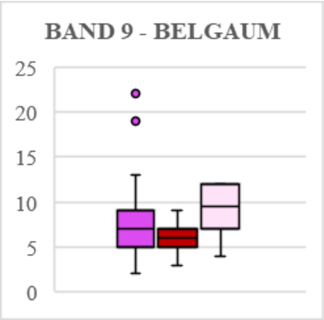
f) Median (fraction)



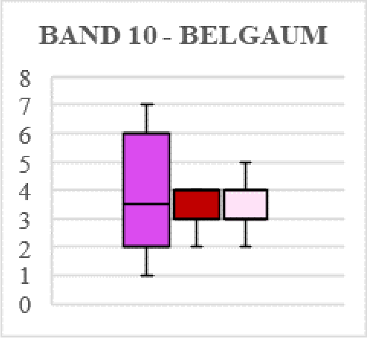
g) Standard deviation (fraction)



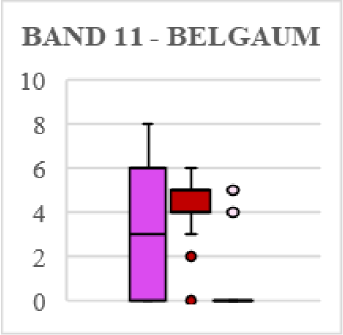
h) Fraction



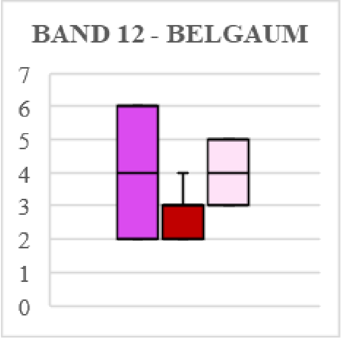
i) Mean (height)



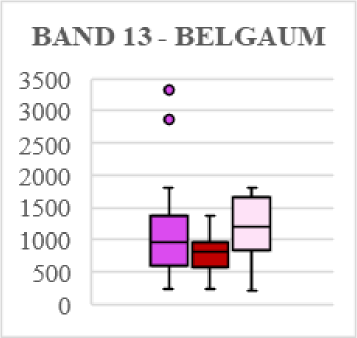
j) Median (height)



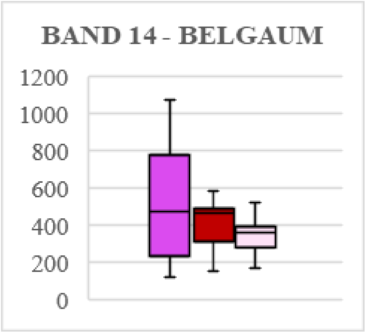
k) Standard deviation (height)



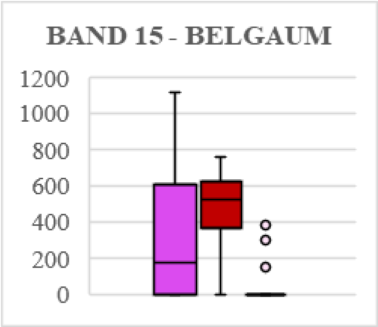
l) Height



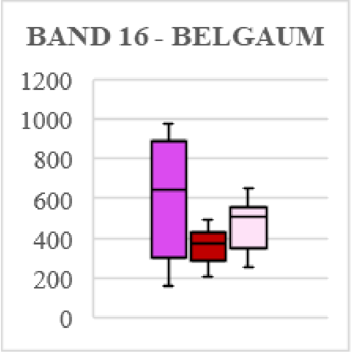
m) Mean (volume)



n) Median (volume)

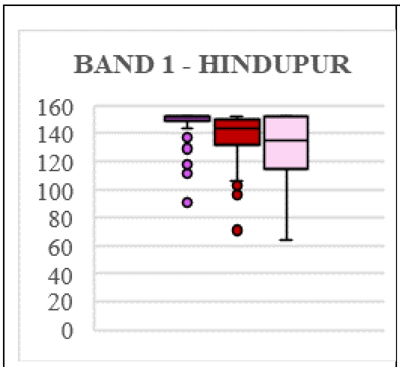


o) Standard deviation (volume)

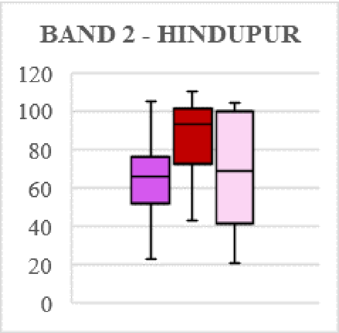


p) Volume

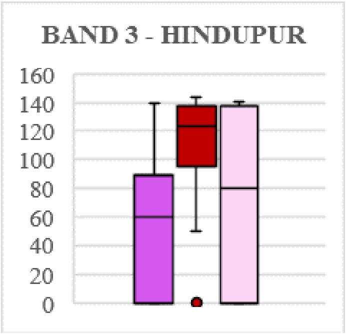
Class separability analysis - Hindupur



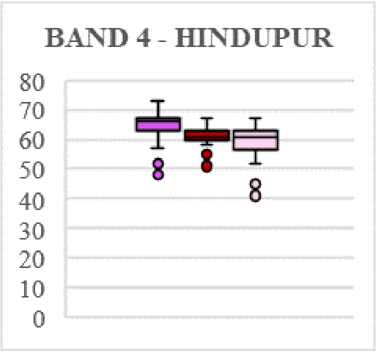
a) Mean (area)



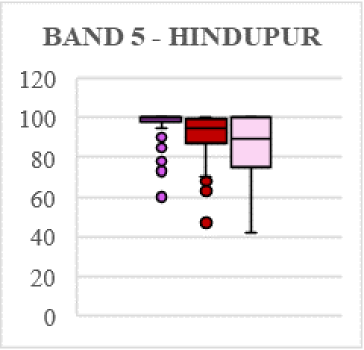
b) Median (area)



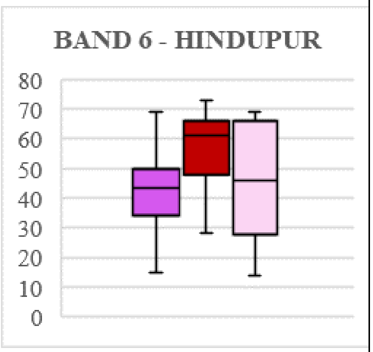
c) Standard deviation (area)



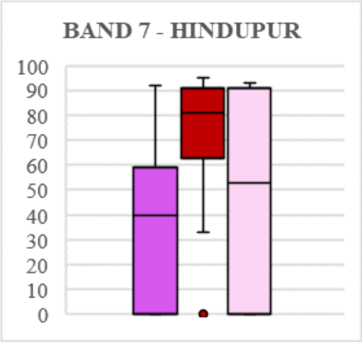
d) Area



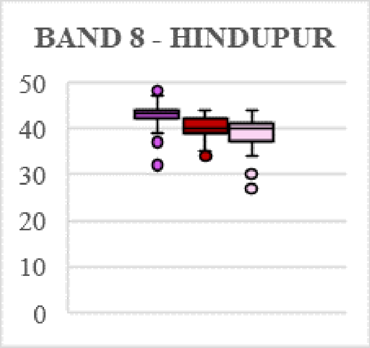
e) Mean (fraction)



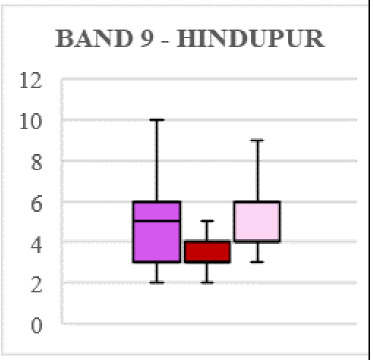
f) Median (fraction)



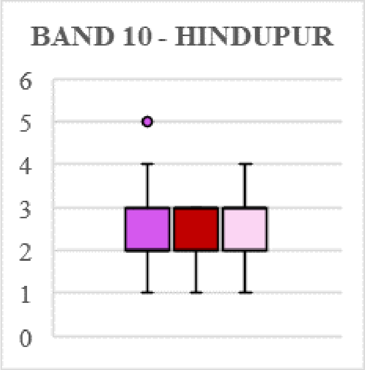
g) Standard deviation (fraction)



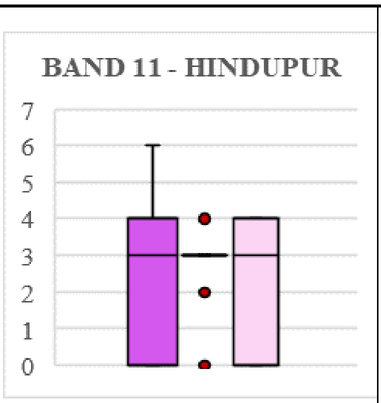
h) Fraction



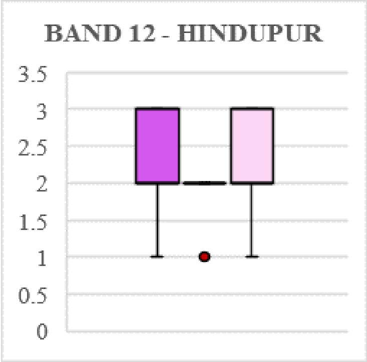
i) Mean (height)



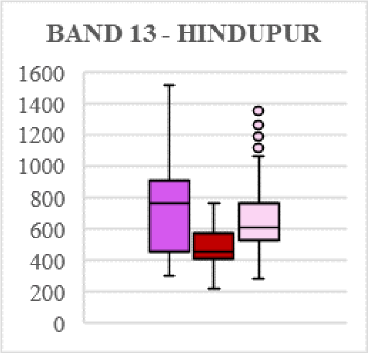
j) Median (height)



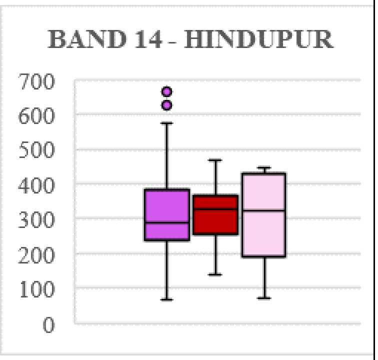
k) Standard deviation (height)



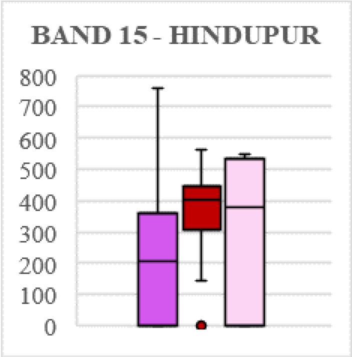
l) Height



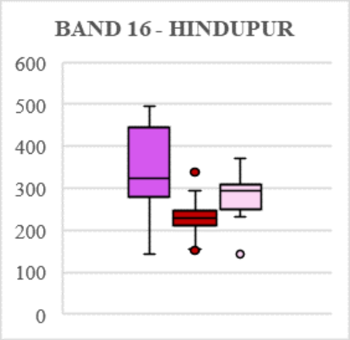
m) Mean (volume)



n) Median (volume)

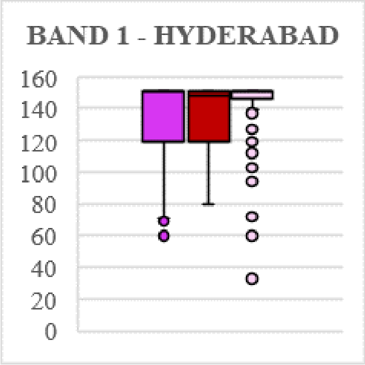


o) Standard deviation (volume)

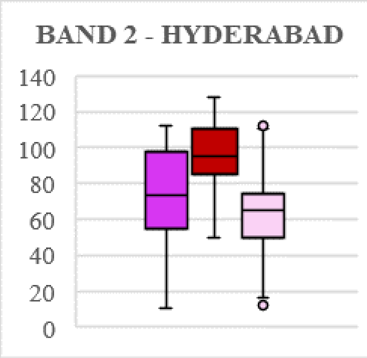


p) Volume

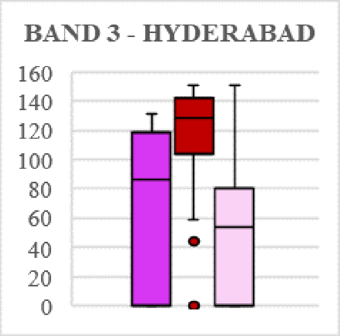
Class separability analysis - Hyderabad



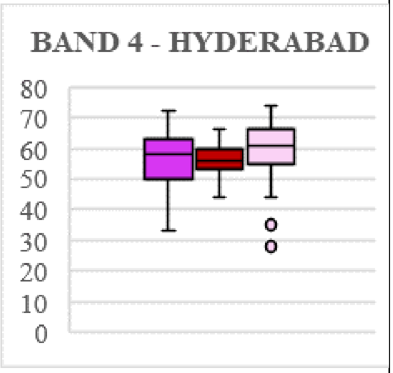
a) Mean (area)



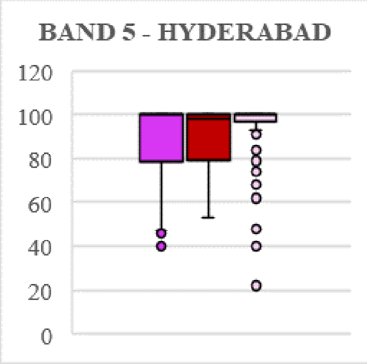
b) Median (area)



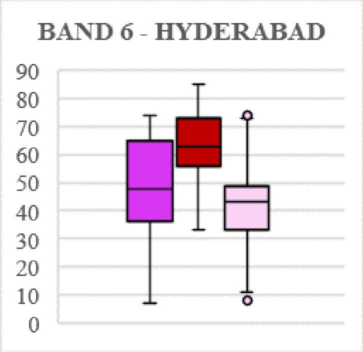
c) Standard deviation (area)



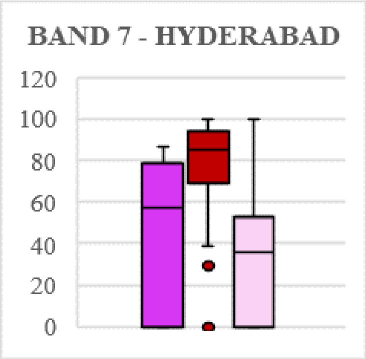
d) Area



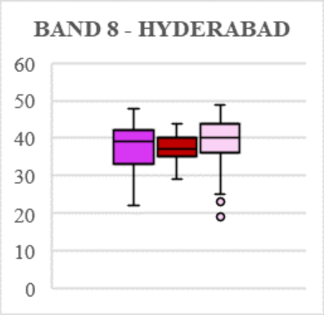
e) Mean (fraction)



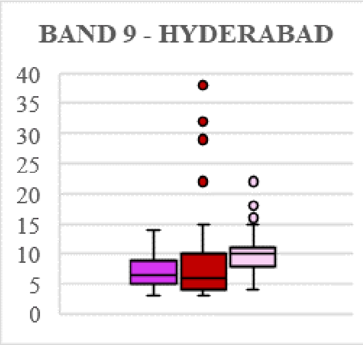
f) Median (fraction)



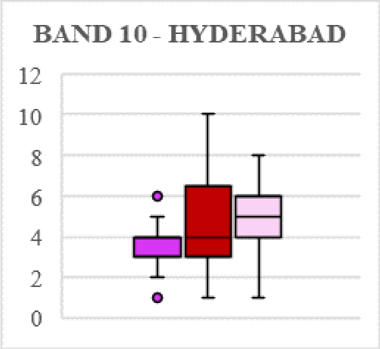
g) Standard deviation (fraction)



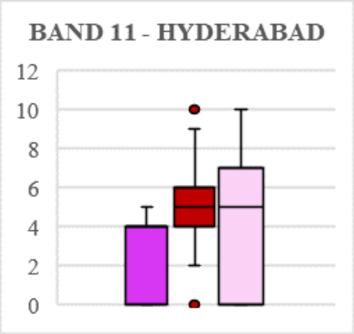
h) Fraction



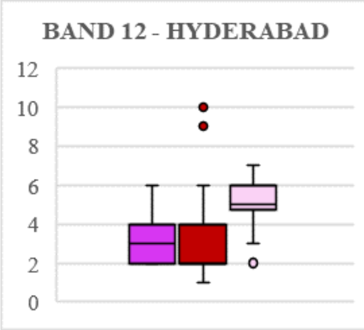
i) Mean (height)



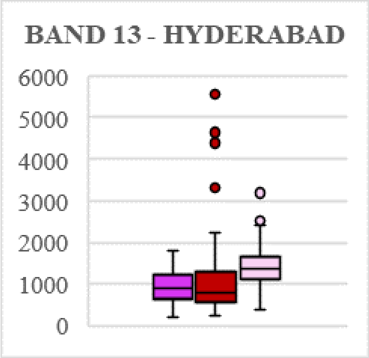
j) Median (height)



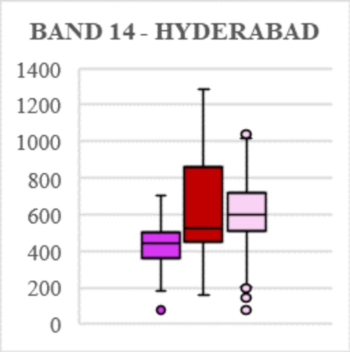
k) Standard deviation (height)



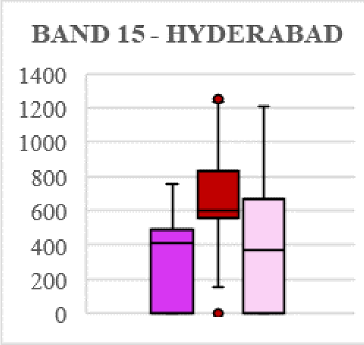
l) Height



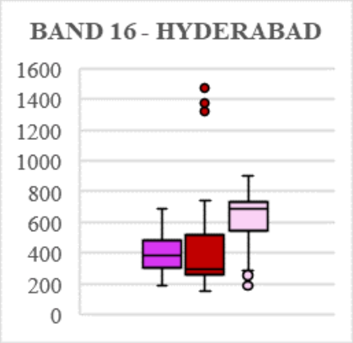
m) Mean (volume)



n) Median (volume)

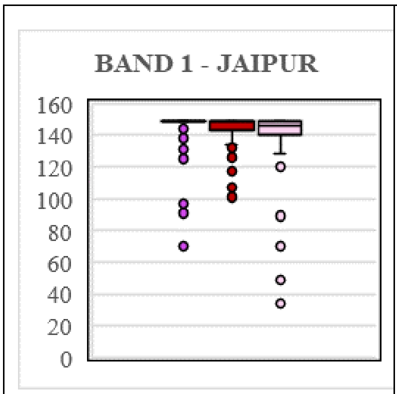


o) Standard deviation (volume)

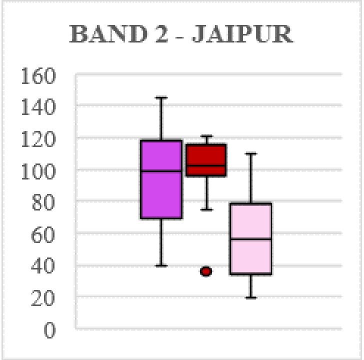


p) Volume

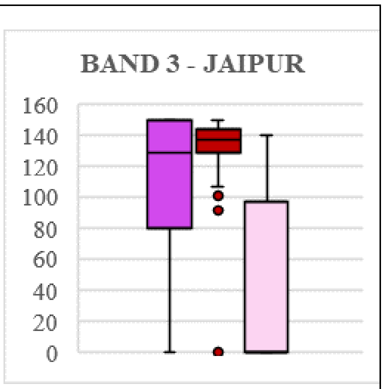
Class separability analysis - Jaipur



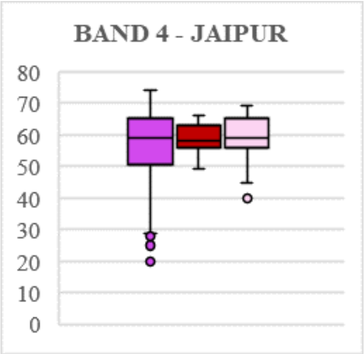
a) Mean (area)



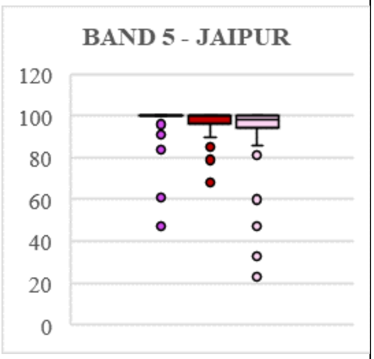
b) Median (area)



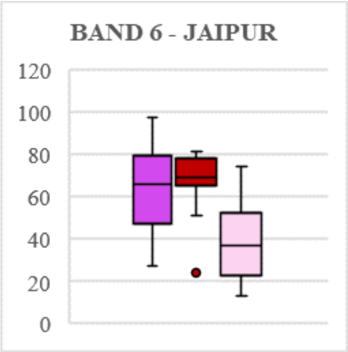
c) Standard deviation (area)



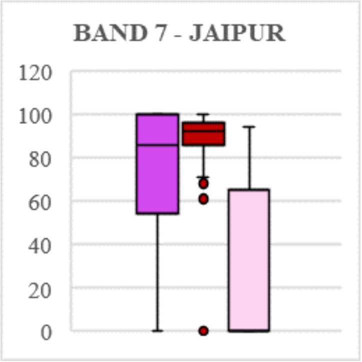
d) Area



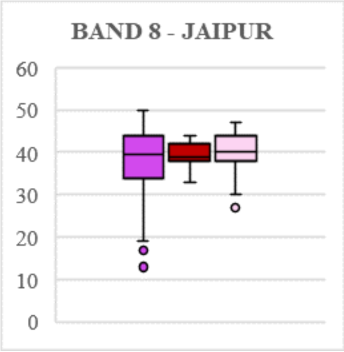
e) Mean (fraction)



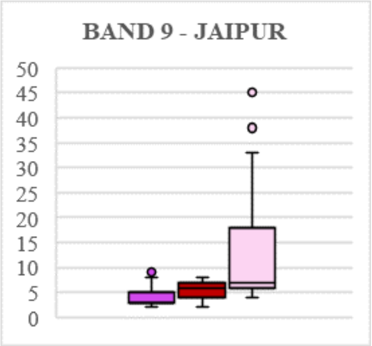
f) Median (fraction)



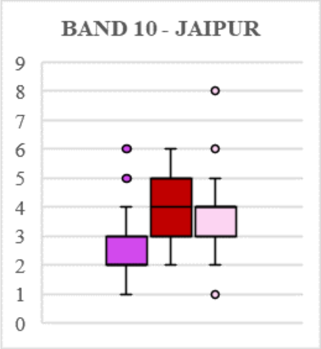
g) Standard deviation (fraction)



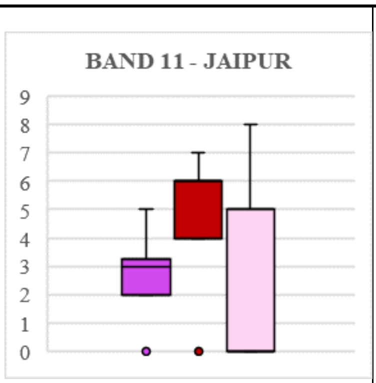
h) Fraction



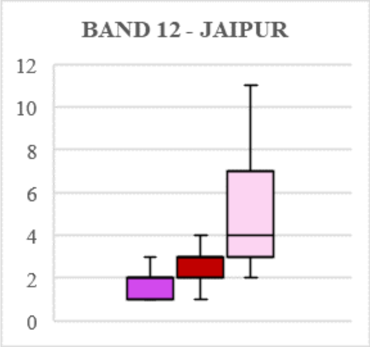
i) Mean (height)



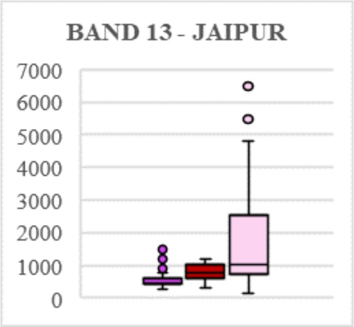
j) Median (height)



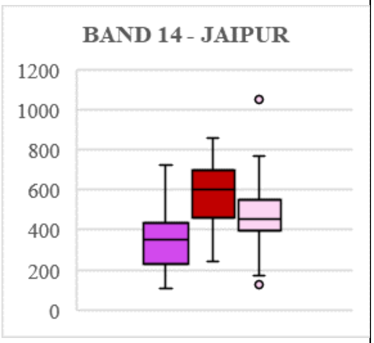
k) Standard deviation (height)



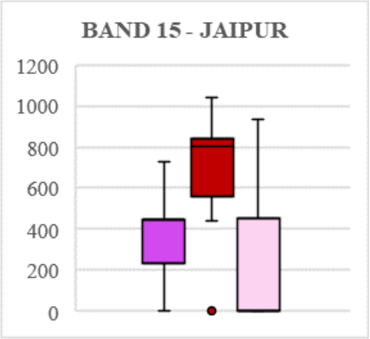
l) Height



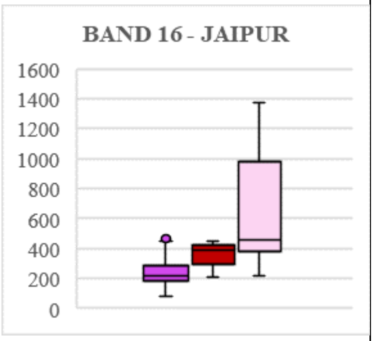
m) Mean (volume)



n) Median (volume)

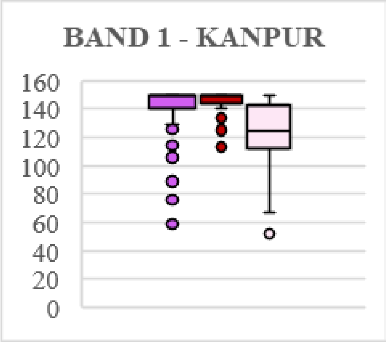


o) Standard deviation (volume)

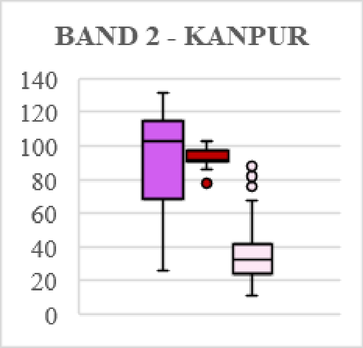


p) Volume

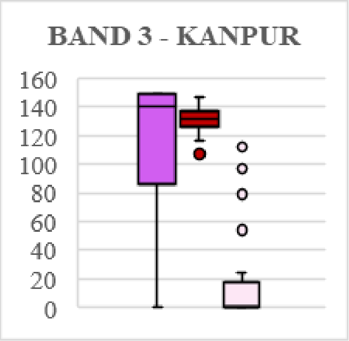
Class separability analysis - Kanpur



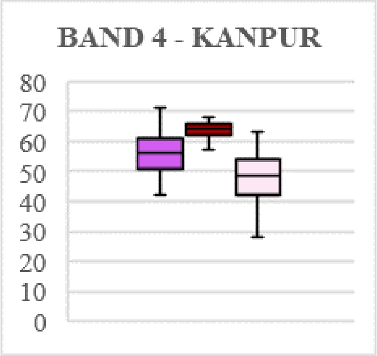
a) Mean (area)



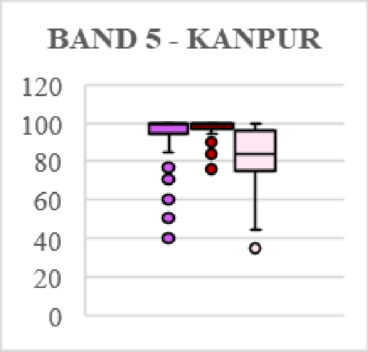
b) Median (area)



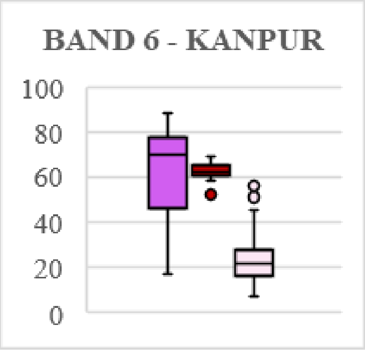
c) Standard deviation (area)



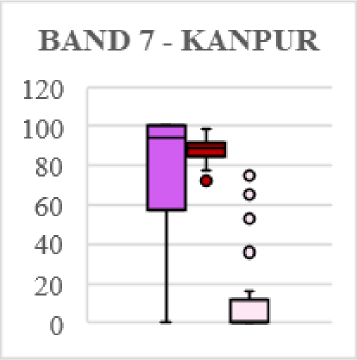
d) Area



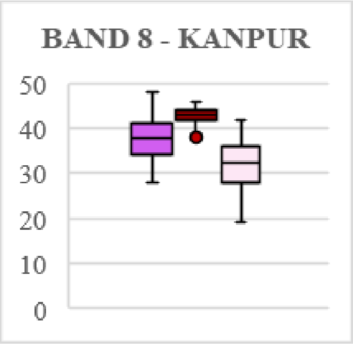
e) Mean (fraction)



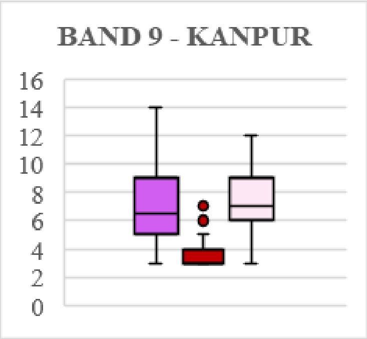
f) Median (fraction)



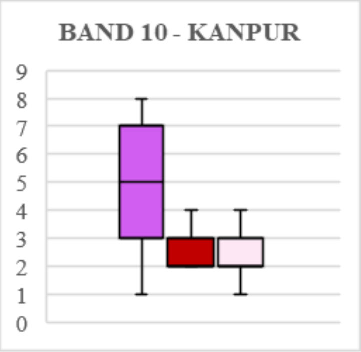
g) Standard deviation (fraction)



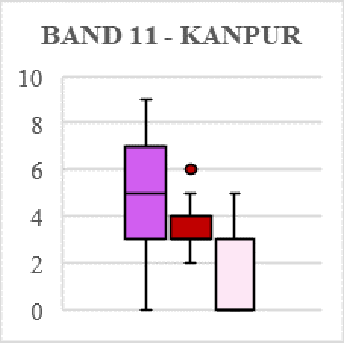
h) Fraction



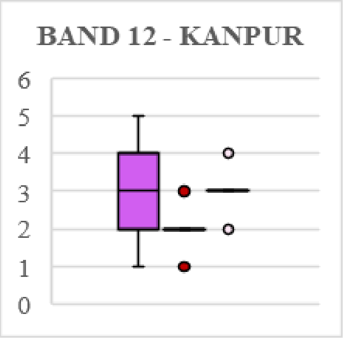
i) Mean (height)



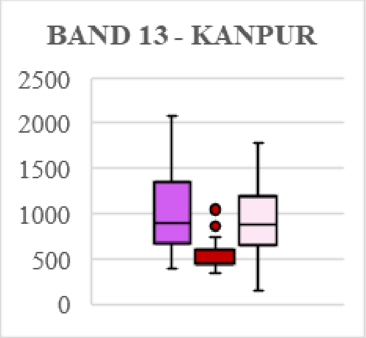
j) Median (height)



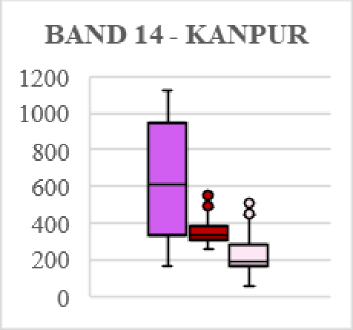
k) Standard deviation (height)



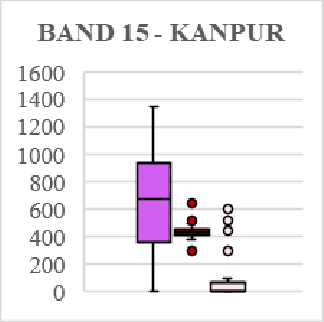
l) Height



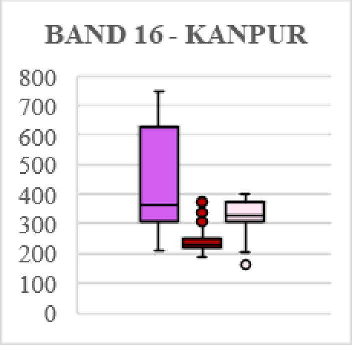
m) Mean (volume)



n) Median (volume)

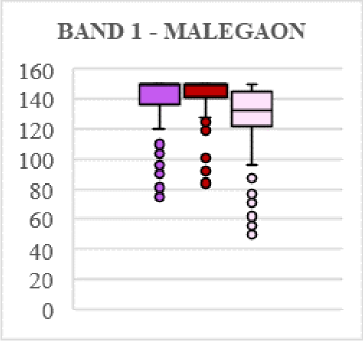


o) Standard deviation (volume)

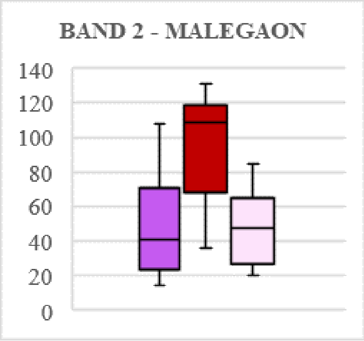


p) Volume

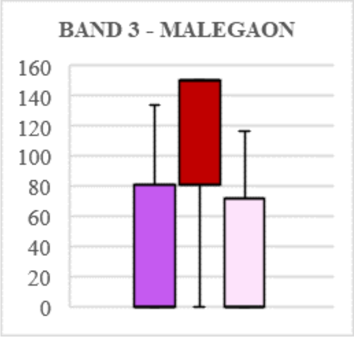
Class separability analysis - Malegaon



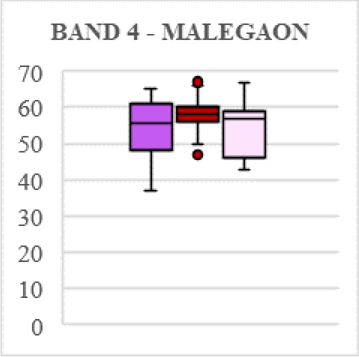
a) Mean (area)



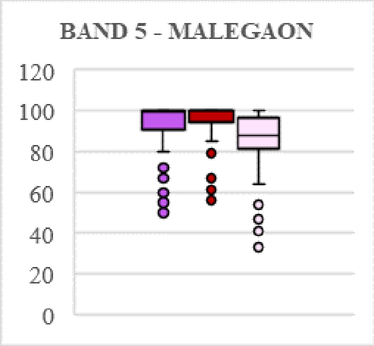
b) Median (area)



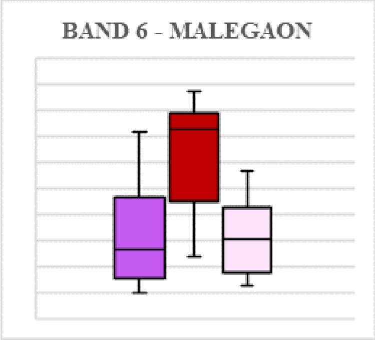
c) Standard deviation (area)



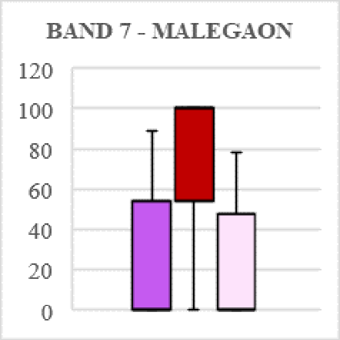
d) Area



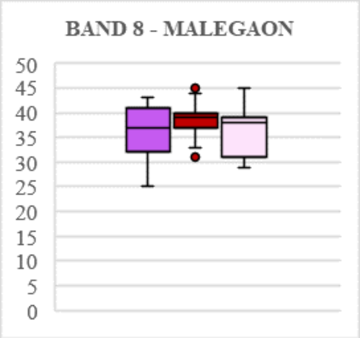
e) Mean (fraction)



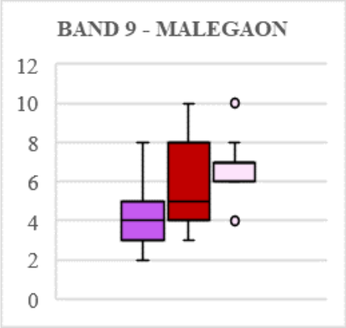
f) Median (fraction)



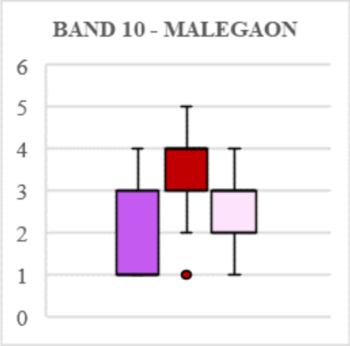
g) Standard deviation (fraction)



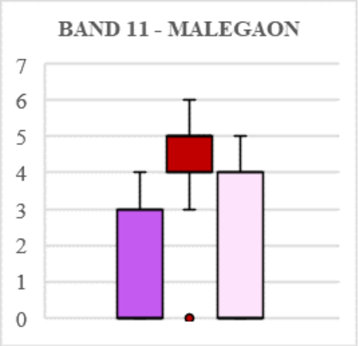
h) Fraction



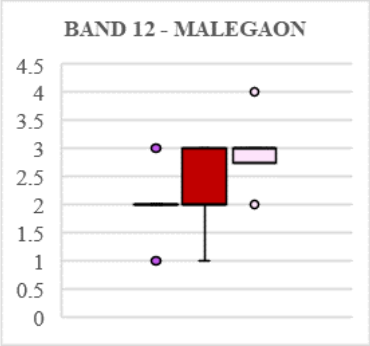
i) Mean (height)



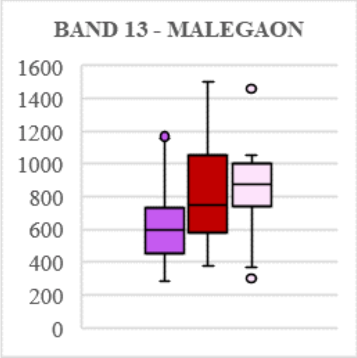
j) Median (height)



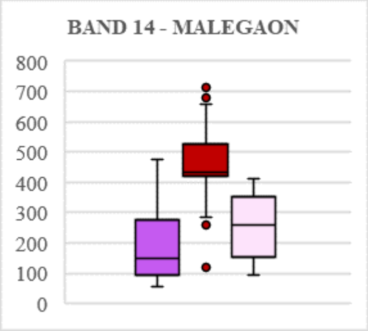
k) Standard deviation (height)



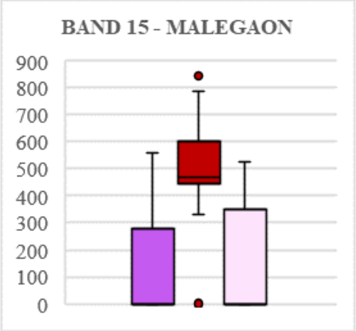
l) Height



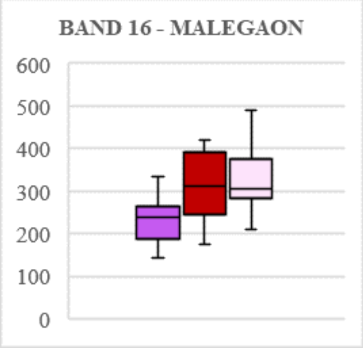
m) Mean (volume)



n) Median (volume)

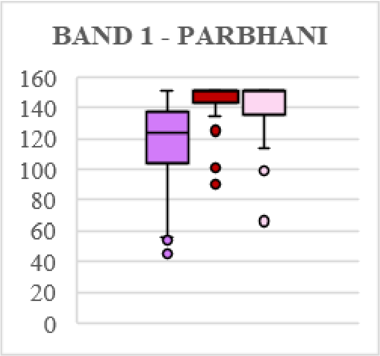


o) Standard deviation (volume)

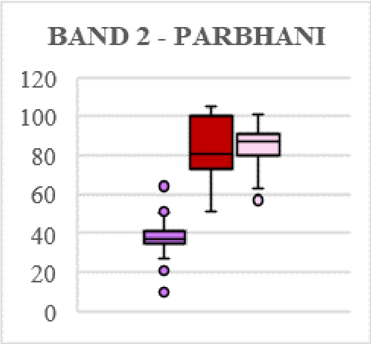


p) Volume

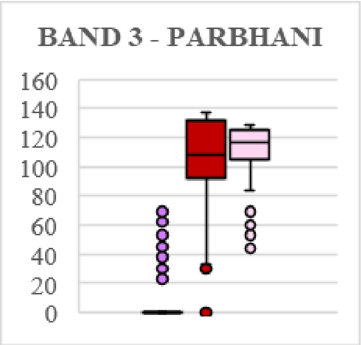
Class separability analysis - Parbhani



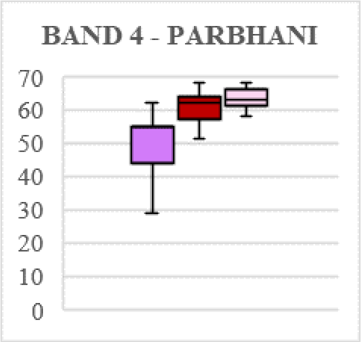
a) Mean (area)



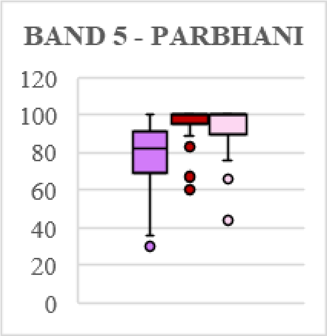
b) Median (area)



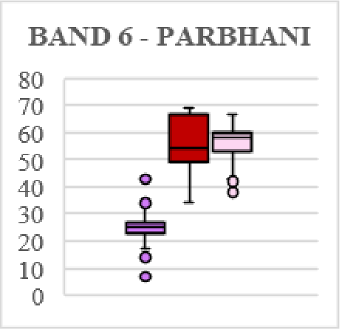
c) Standard deviation (area)



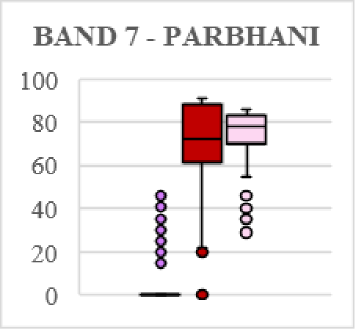
d) Area



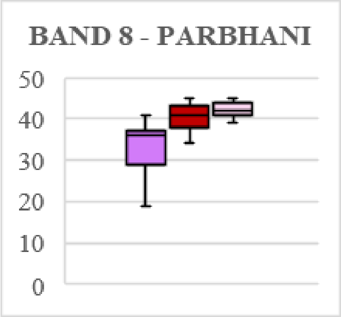
e) Mean (fraction)



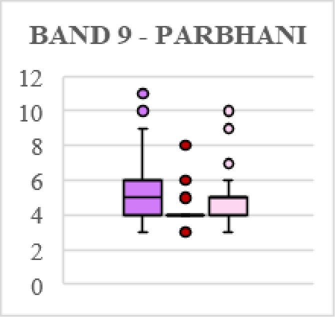
f) Median (fraction)



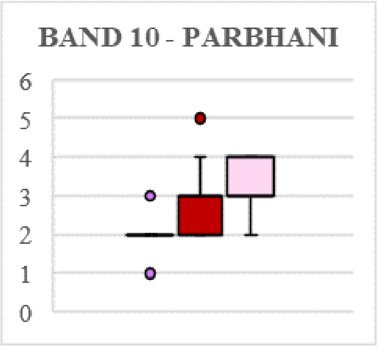
g) Standard deviation (fraction)



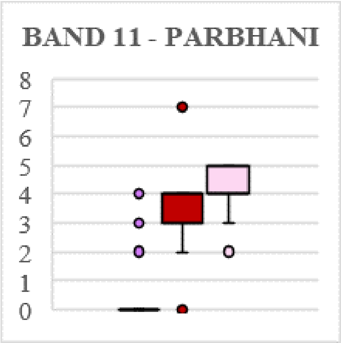
h) Fraction



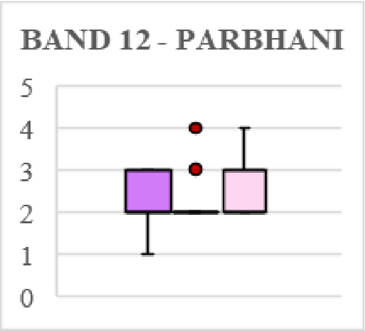
i) Mean (height)



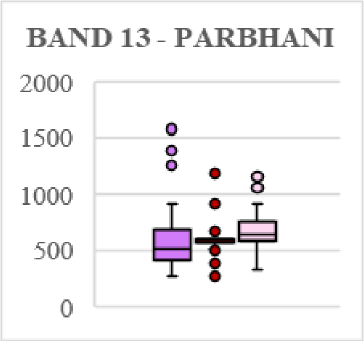
j) Median (height)



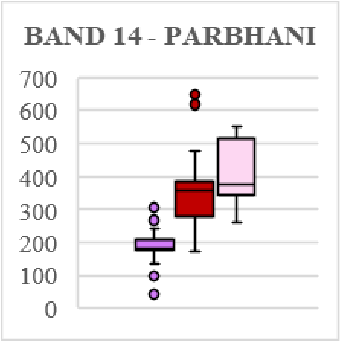
k) Standard deviation (height)



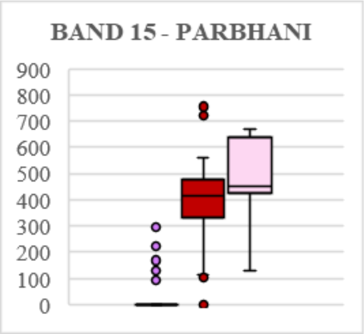
l) Height



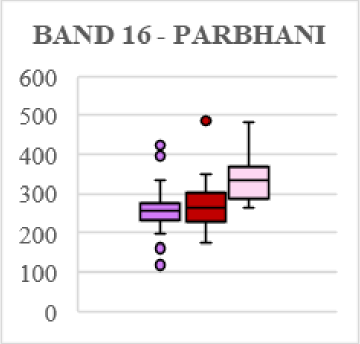
m) Mean (volume)



n) Median (volume)

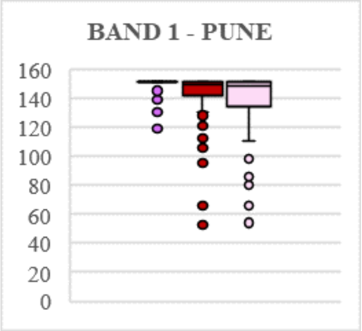


o) Standard deviation (volume)

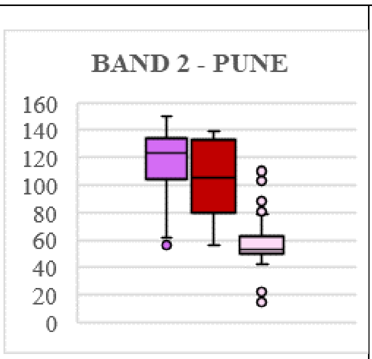


p) Volume

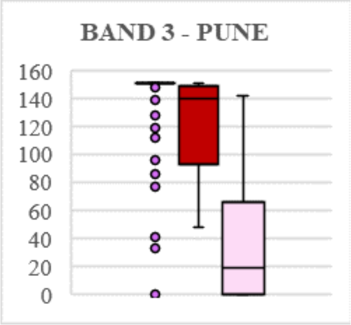
Class separability analysis - Pune



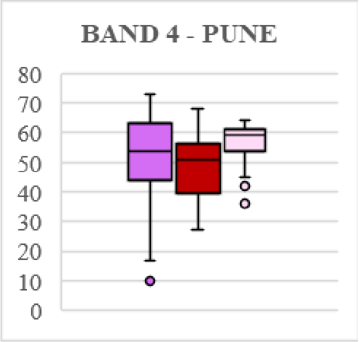
a) Mean (area)



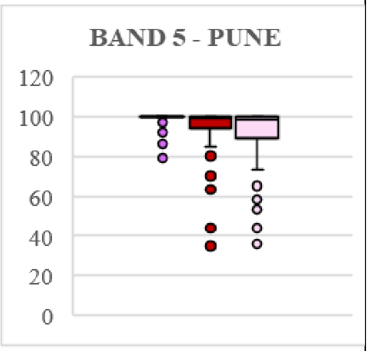
b) Median (area)



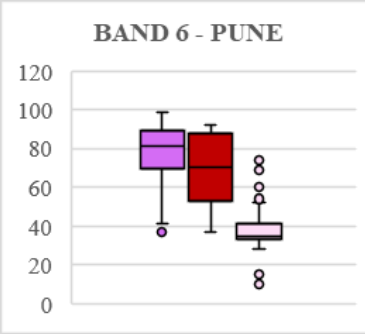
c) Standard deviation (area)



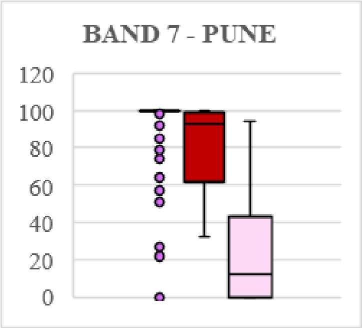
d) Area



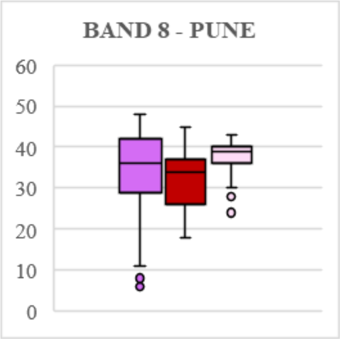
e) Mean (fraction)



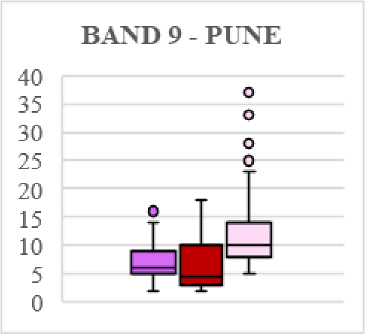
f) Median (fraction)



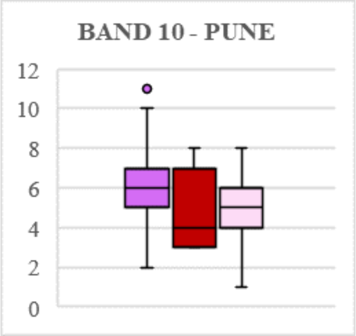
g) Standard deviation (fraction)



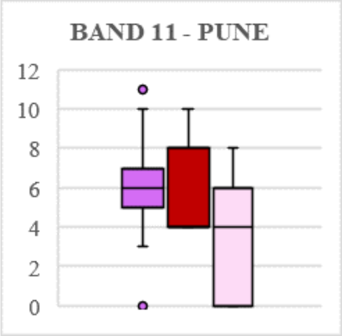
h) Fraction



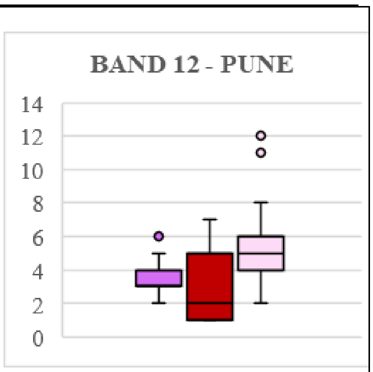
i) Mean (height)



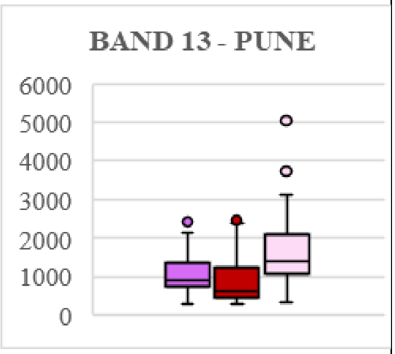
j) Median (height)



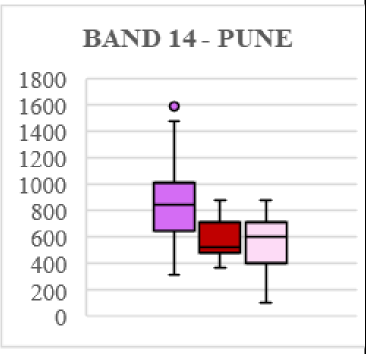
k) Standard deviation (height)



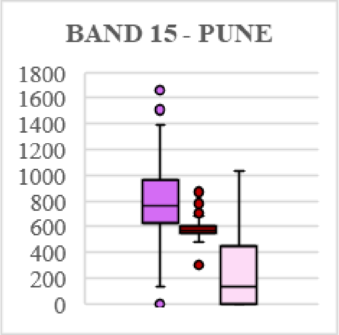
l) Height



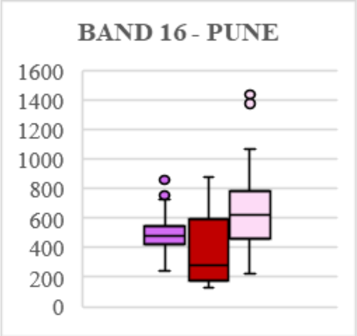
m) Mean (volume)



n) Median (volume)

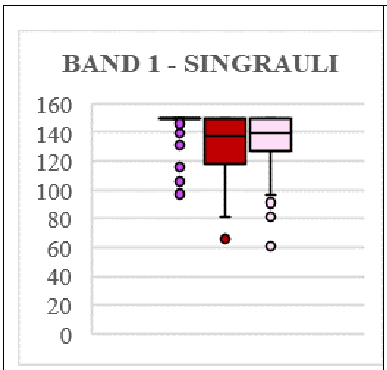


o) Standard deviation (volume)

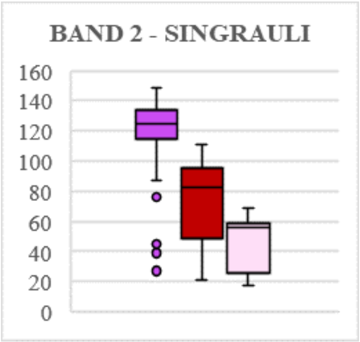


p) Volume

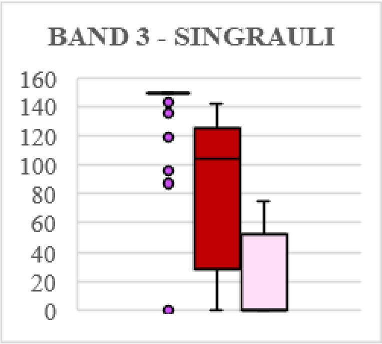
Class separability analysis - Singrauli



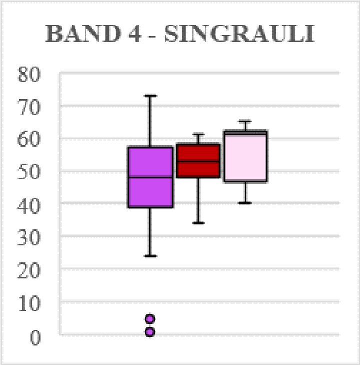
a) Mean (area)



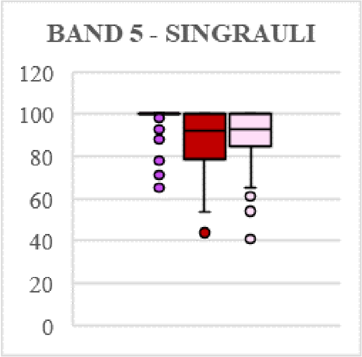
b) Median (area)



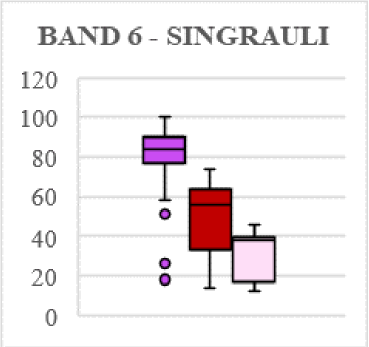
c) Standard deviation (area)



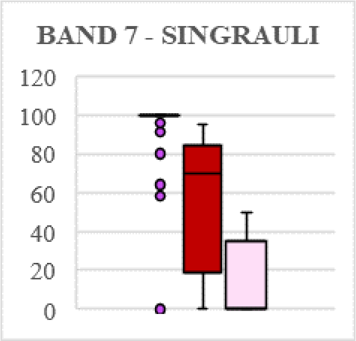
d) Area



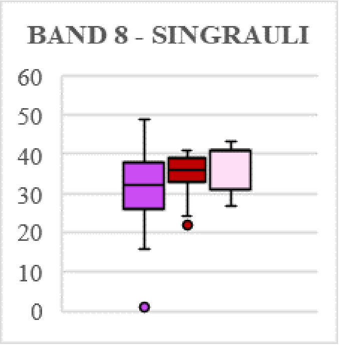
e) Mean (fraction)



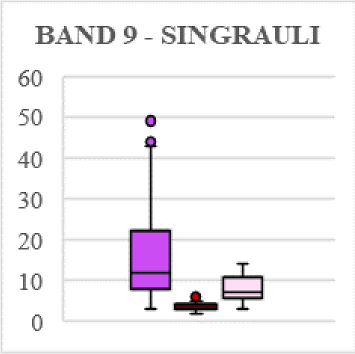
f) Median (fraction)



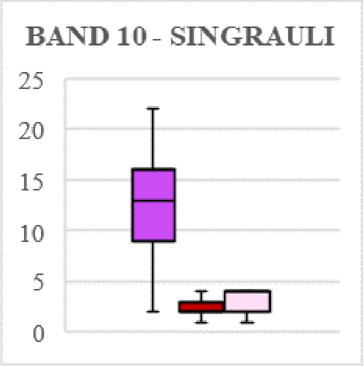
g) Standard deviation (fraction)



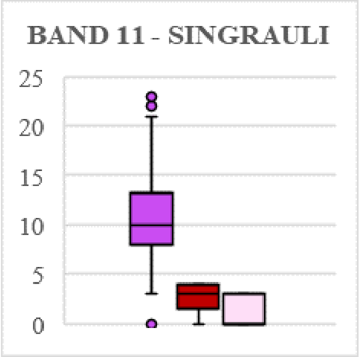
h) Fraction



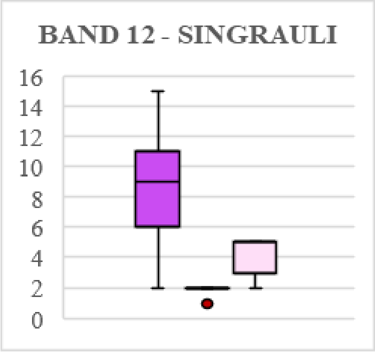
i) Mean (height)



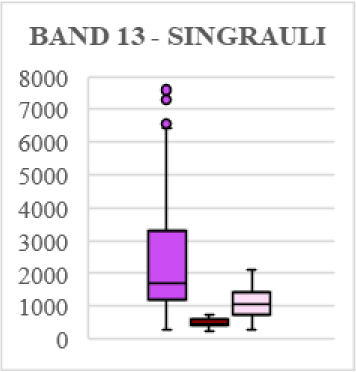
j) Median (height)



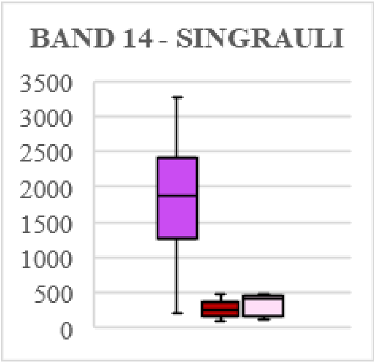
k) Standard deviation (height)



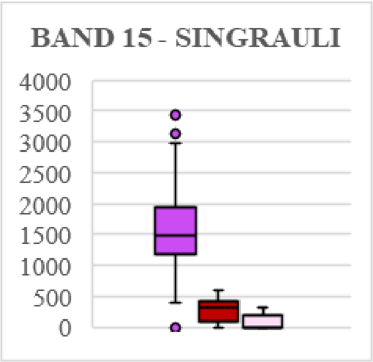
l) Height



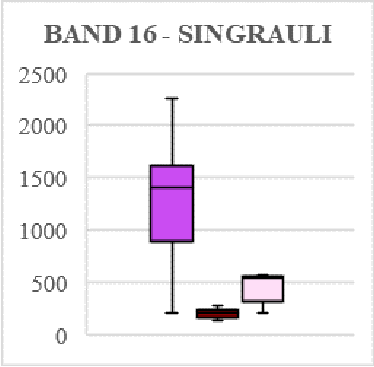
m) Mean (volume)



n) Median (volume)

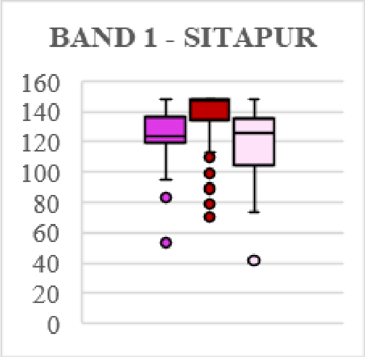


o) Standard deviation (volume)

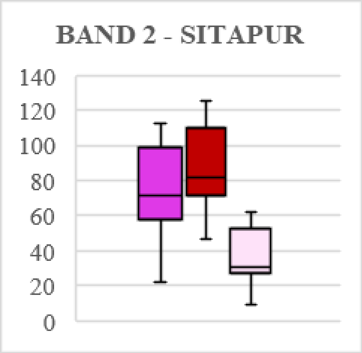


p) Volume

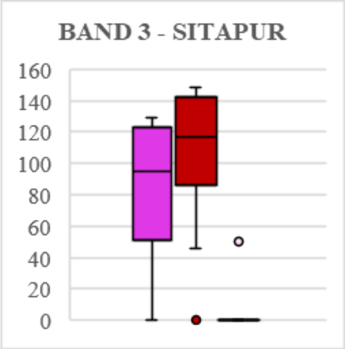
Class separability analysis - Sitapur



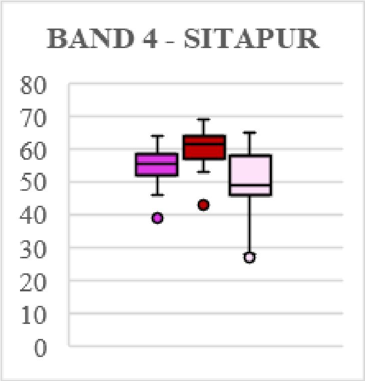
a) Mean (area)



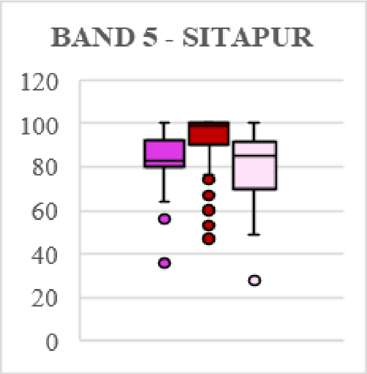
b) Median (area)



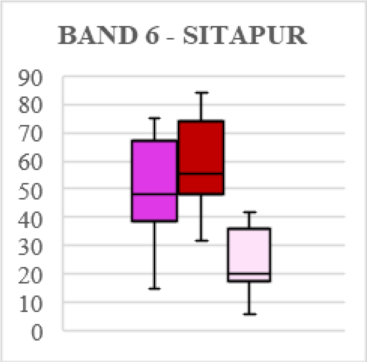
c) Standard deviation (area)



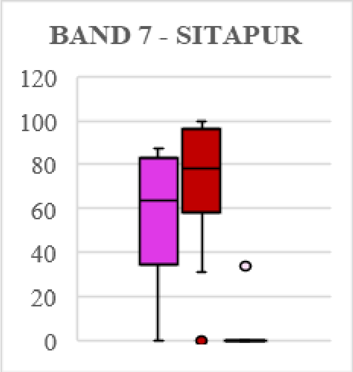
d) Area



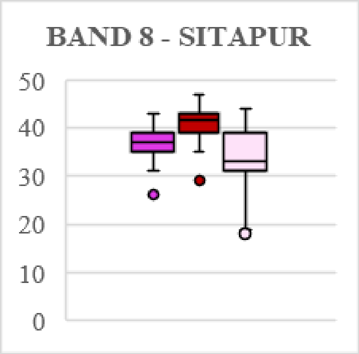
e) Mean (fraction)



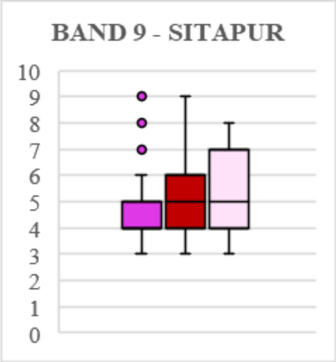
f) Median (fraction)



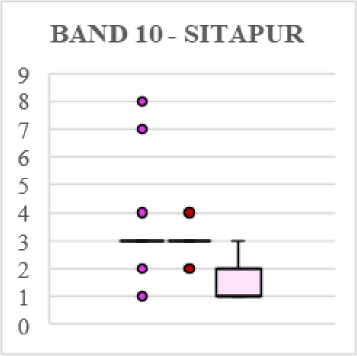
g) Standard deviation (fraction)



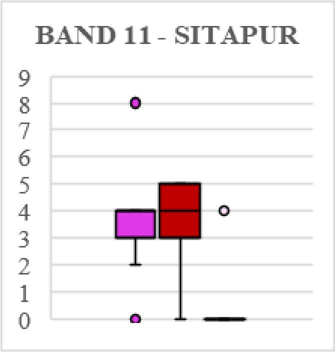
h) Fraction



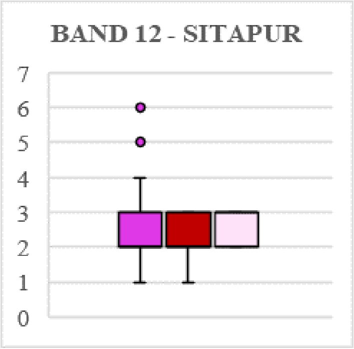
i) Mean (height)



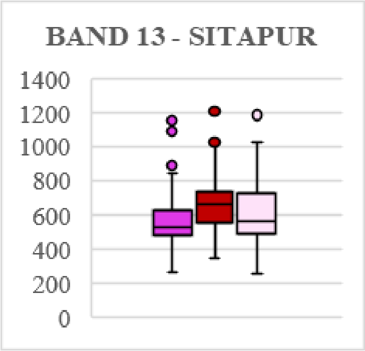
j) Median (height)



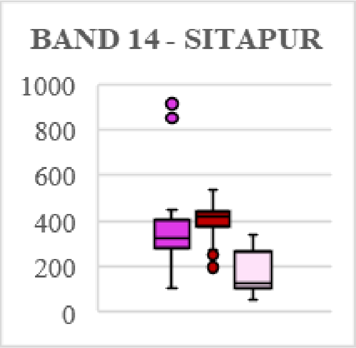
k) Standard deviation (height)



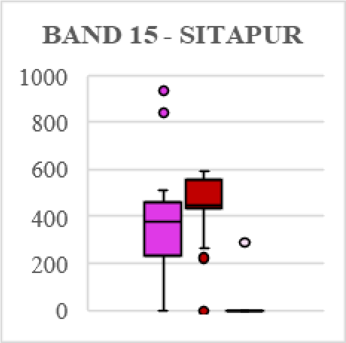
l) Height



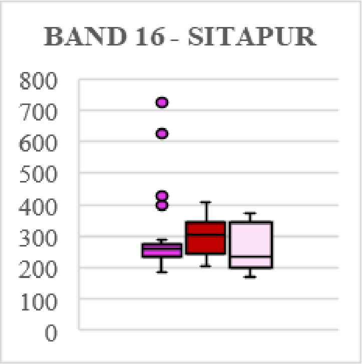
m) Mean (volume)



n) Median (volume)



o) Standard deviation (volume)

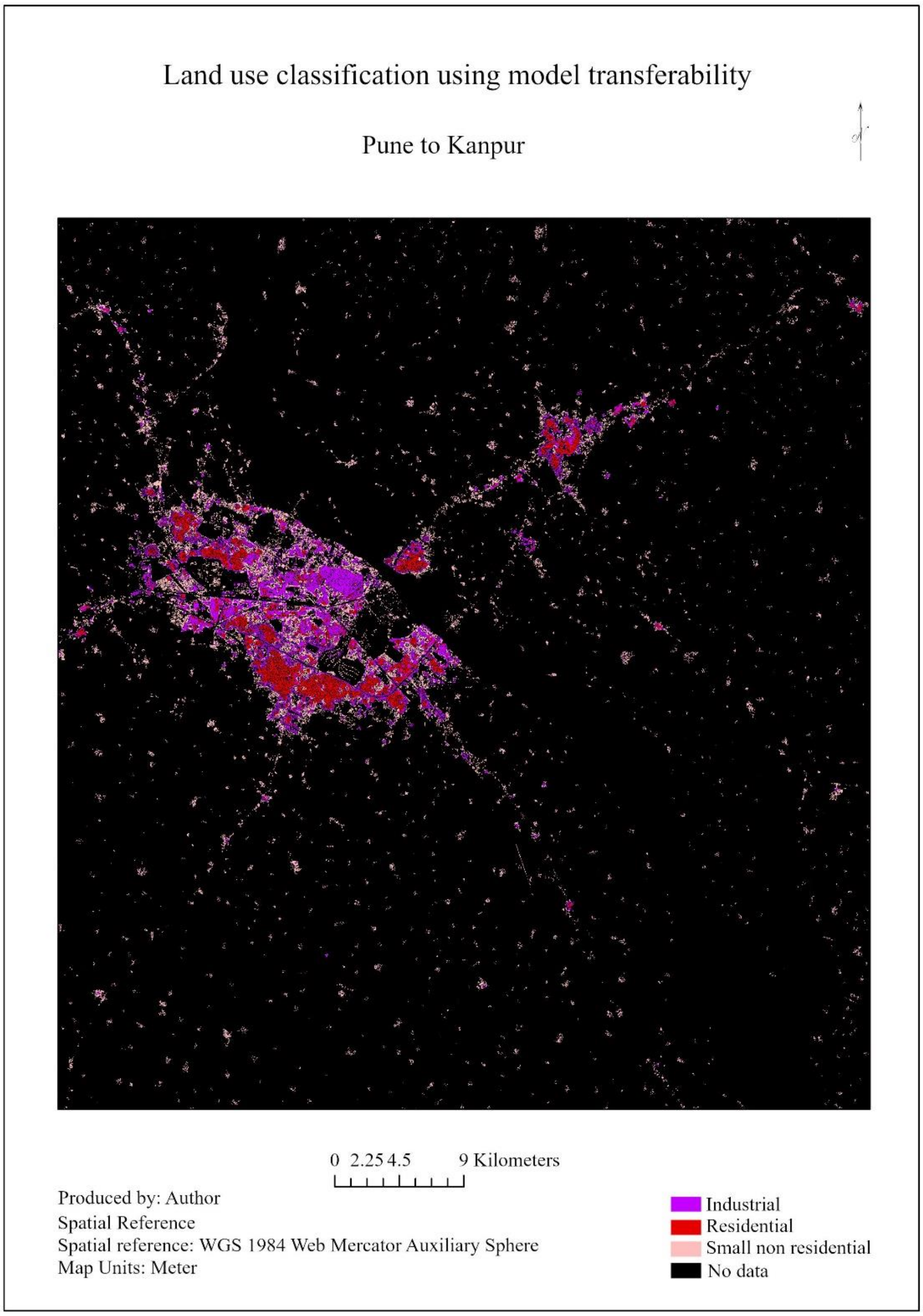


p) Volume

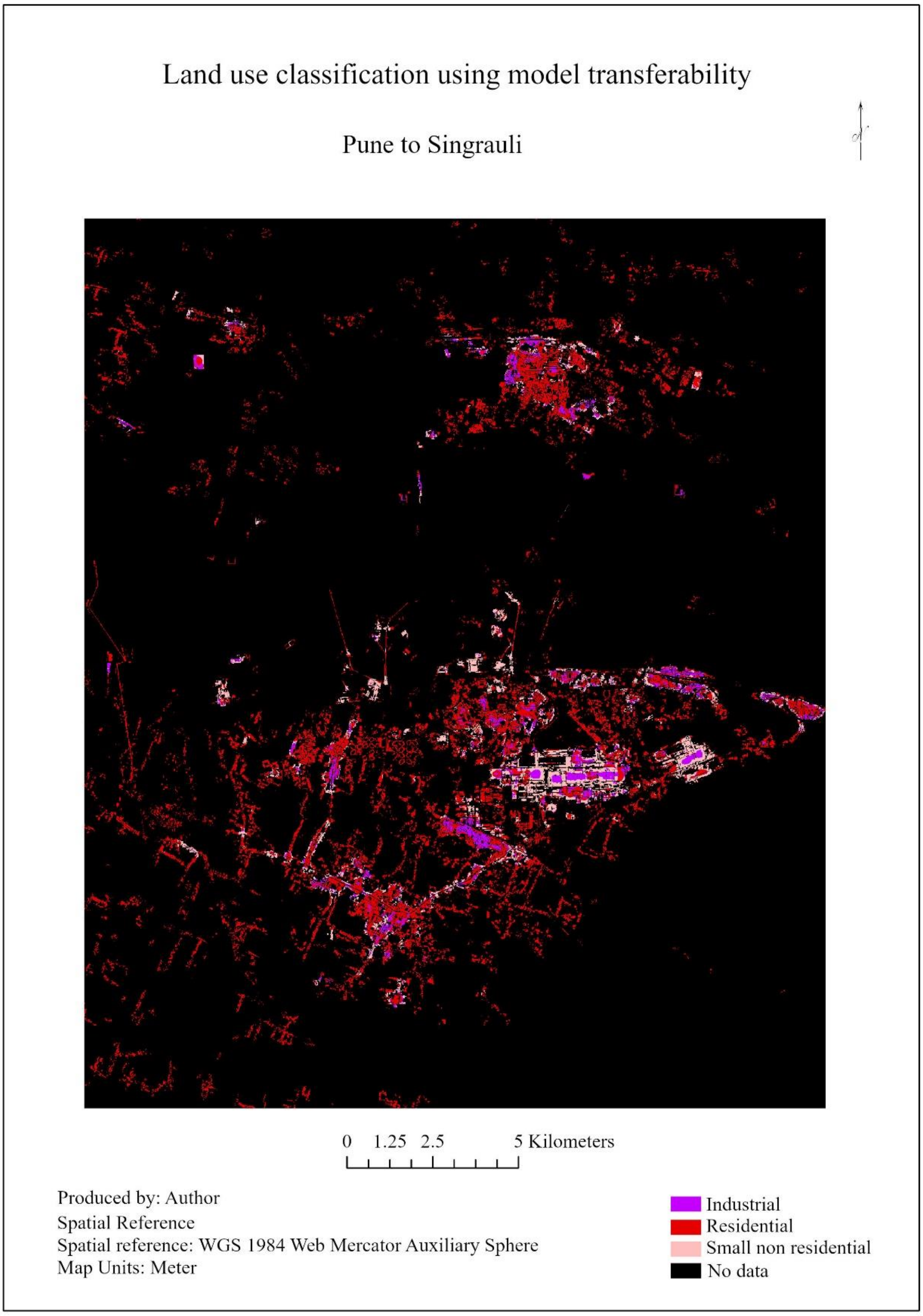
Appendix 12

Classification images of cities suitable for model transferability

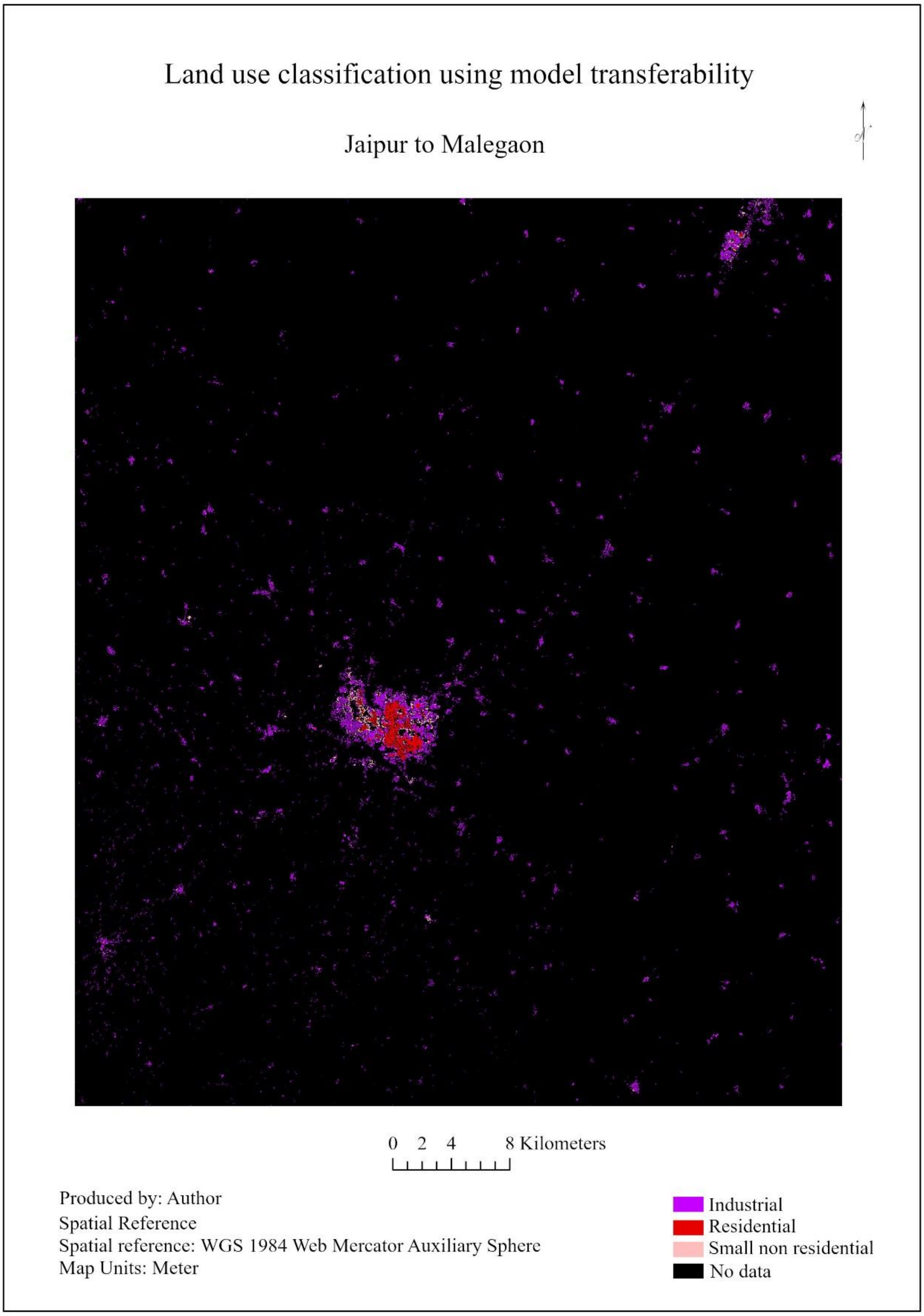
Model trained in Pune and transferred to Kanpur (0.8 – 0.9 prediction accuracy and >=0.7 individual class F1 scores)



Model trained in Pune and transferred to Singrauli (0.8 – 0.9 prediction accuracy and >=0.7 individual class F1 scores)



Model trained in Jaipur and transferred to Malegaon (0.7 – 0.8 prediction accuracy and >=0.6 individual class F1 scores)


Land use classification using model transferability
Jaipur to Malegaon
0 2 4 8 Kilometers
Produced by: Author
Spatial Reference
Spatial reference: WGS 1984 Web Mercator Auxiliary Sphere
Map Units: Meter
🟪 Industrial
🟥 Residential
🟥 Small non residential
⬛ No data