APPROVAL SHEET

Title of Thesis: Calculating Representativeness of Geographic Sites Across the World

Name of Candidate: Ashwinkumar Ganesan,
M.S. in Computer Science,
Department of Computer Science &
Electrical Engineering, 2012

Thesis and Abstract Approved: ______________________
Dr. Tim Oates,
Professor,
Department of Computer Science and
Electrical Engineering

Date Approved: ______________________

Curriculum Vitae

Name: Ashwinkumar Ganesan

Permanent Address: 4749 Drayton Green, Halethorpe, MD - 21227

Degree: Masters in Computer Science, August 2012

Date of Birth: 01/29/1986

Place of Birth: Pune, India

Secondary Education: Loyola High School & Junior College, Pune, India

Collegiate institutions attended:

University of Maryland Baltimore County, M. S. Computer Science, 2012
Maharashtra Institute Of Technology, Pune, B. E. Computer Engineering, 2007

Major: Computer Science

Professional positions held:

Embedded Software Engineer, Breakaway Consulting, MD. (Aug. 2012 – Present).
Software Development Intern, Symantec Corporation. (June 2011 – Aug. 2011).
Senior Member Of Technical Staff, Niyuj Enterprise Software Solutions, Pune, India. (Nov. 2009 – Aug. 2010).
Assistant Systems Engineer, Tata Consultancy Services Limited, Mumbai, India. (Sept. 2007 – Oct. 2009).

ABSTRACT

Title of Thesis: Calculating Representativeness of Geographic
Sites Across the World
Ashwinkumar Ganesan,
M.S. in Computer Science, 2012

Thesis directed by: Dr. Tim Oates,
Professor,
Department of Computer Science and
Electrical Engineering

GLOBE is a global correlation engine, a project to study the effects on Land Change based on a set of parameters that include temperature, forest cover, human population, atmospheric parameters and many other variables. The aim of this research is to understand, how a study or a set of studies of specific geographic areas is *representative* of other areas of the world. The generic form of the question is, given a set of data points with a set of variables, how to determine how much a selected subset of points represents the rest of the distribution.

The research aims to answer a set of questions which include the definition of representativeness of a geographical site and how the representativeness can be computed. Researchers studying land change will dynamically select a subset of variables which they would like to study. Hence the method developed not only computes representativeness, but does so in an efficient manner. For this purpose, we apply dimension reduction techniques to reduce the size of computation and analyze the effectiveness of using these techniques to calculate representativeness.

Keywords: Principal Component Analysis (PCA), Dimension Reduction, Representativeness, Land Change Science

Calculating Representativeness of Geographic Sites Across the World

by

Ashwinkumar Ganesan

Thesis submitted to the Faculty of the Graduate School
of the University of Maryland in partial fulfillment
of the requirements for the degree of
M.S. in Computer Science,
2012

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

Dedicated to my Dad, Mom and my entire family, especially my amazing niece. A special dedication to all democratic republics that, I believe, continuously learn what it really means to vote and elect representatives.

ACKNOWLEDGMENTS

I would like to thank my advisor and mentor, Dr. Tim Oates for the amazing guidance and support he gave me during my thesis research. I am grateful for the opportunity given to me to work on the Globe Project. It has been a privilege to work with him. During this past year, I have learned a lot from him. Starting from how to define a problem, understanding the process of thinking and asking questions, to eventually coming up with a solution to the problem. It has been a great learning experience all along.

I would like to thank Dr. Matt Schmill for helping me with the project. It would have been difficult to engineer the solutions we designed into Globe without his constant help and support. I would like to thank him and Dr. Tim Finin for being on my committee.

I would like to express my gratitude to Dr. Erle Ellis for all the help and support on the project. He has introduced me to the new area of Land Change Science and the use of statistics in it.

I would like to thank Dr. Niyati Chhaya for being an awesome friend and CORAL labmate. Her continous help and encouragement, made the work and lab a fun environment. Her insights helped me during my research and while writing this document.

A special vote of thanks to Varish Mulwad and Dr. Kalpakis for their lively discussions that kept me on my toes. And finally, my gratitude to my friends at home and colleagues in the CORAL, Ebiquity and DREAM lab, who made this journey a great experience.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

The study of environmental sciences and especially *climate change* has grown in importance over the last two decades. Some reasons to study these areas are to understand their effects on human populations and on natural resources available on the planet. They encompass a host of issues that are related to diverse areas, including ecology, government policies, public health and economic policies [1]. Global climate change study attempts to understand (and predict) how the climate evolves based on current and historical evidence. It explores the change in climate occurring due to natural phenomenon and anthropogenic reasons. Our effort is to find a way to globally utilize these studies by measuring similarity.

1.1 Land Change Science & Globe

Land Change Science (LCS) is a part of global climate change research and studies the effect of human activity on land and on the climate. As part of LCS, studies of specific geographic locations are conducted across the entire world. The locations where these studies are conducted are based on a number of factors that are decided depending upon the goals of the study that could be social or ecological. Goals that are social could include understanding the nature of human interaction with the environment and the impact of the change on human lives. Ecological studies try to measure parameters such as deforestation and its effect on the environment as a whole. The size or scale of a study could be based on the amount of funds available or to perform a study for measurement of global climate change (like an IPCC study) that can be used for making policy [2]. These studies are conducted by scientists on-site or by remotely monitoring these locations through stations that are constructed on the site.

Globe is a global correlation engine to study land change science [1]. It tries to fulfil the basic requirement of providing the ability to generalize LCS studies. It has multiple objectives which include constructing of an online social media system to facilitate collaborative work between land change scientists, providing a set of tools to analyse and generalize local observations to larger regions or the entire globe, and constructing scientific workflows [1].

1.2 Motivation

Land change studies are expensive. Hence the main motivation is to provide an understanding of how the conclusions of these studies can be generalized to other parts of the world. The notion of how generalized the studies are can help land change scientists to:

1. Reduce the number and cost of studies conducted in the future.

2. Select new locations in the world where studies have not been conducted.

3. Analyse existing studies in a different manner by showing a global pattern of the distribution of a set of selected parameters.

1.3 Representativeness

There are a set of parameters or variables whose information is available for all parts of the world. They are the set of *global* variables. These variables include potential vegetation, land utilized for agriculture, temperature, and many others. Scientists studying land change select a case study or a set of case studies which have been conducted. They select the variables which represent the environment in which the studies were conducted or the variables for which they want to analyze the case study. To generalize the results of these studies, they use these parameters to find other parts of the world that are *similar*. The *representativeness* of a given set of case studies is defined as the extent to which they *cover* the rest of the world. For example, consider a scientist analyzing the Van Vliet Study [3] on trends in swidden cultivation. She would perform the following steps

1. Create a collection of all the locations that are part of study.
2. Select this *collection* from *Globe's* User Interface.
3. Select parameters such as %crops, %tree cover, temperature, Market Access Index and Potential Vegetation as the meta-study uses these variables.
4. Compute representativeness for the collection. Display it on a world map to show which other regions of the world are similar or to what extent the rest of regions are dissimilar to the ones in the study.

1.4 Challenges & Contributions

The number of distinct geographical regions across the world into which the world map can be divided is very large. Each region is has its own global variable information. There are a large number of dimensions which, coupled with the number of sites, makes the representativeness computation expensive. The users (scientists in this case) select a subset of variables in realtime. The results for this calculation are to be provided in realtime. There are essentially two challenges here:

1. To find a method to measure representativeness.
2. To provide an algorithm, to find a minimum set of new locations where studies can be conducted so as to maximize representativeness for the selected set of variables.

Our contribution in this thesis is to provide a mathematical formulation for representativeness, reduction of computation time and increase in efficiency by using a dimension reduction technique and providing a method to validate the results of our selected locations. Dimension reduction techniques are required since the number of dimensions is large and methods that use the original set of dimensions, such as clustering, will not be able to perform computation in realtime. The dimension reduction method we apply is Principal Component Analysis (PCA). PCA is useful because it prioritizes dimensions with higher variance. We show in this thesis how the algorithm can be used to select new locations and the correlation between the geographic points in the original space and single dimension PCA space. This correlation helps us measure the effectiveness of PCA and selected locations. Representativeness is displayed in the form of a *Heat Map* using Google Maps. The map has markers that show the location where each case study was conducted.

Chapter 2

REPRESENTATIVENESS & RELATED WORK

This chapter defines what representativeness means, and explores various methods that can been used to compute representativeness.

2.1 Representativeness

2.1.1 Definition

Representativeness describes how a data point or a set of data points can be used to generalize to the rest of the data set. In case of the Globe project, a data point refers to a specific region in the world where the study is conducted. Consider a distribution of data points where

- D is the given dataset of points
- S is a sample set of points such that $S \subseteq D$
- H is a histogram based on D
- $Bin(H,s)$ is the bin where the data value s falls in H
- $P(H,i)$ is the height / probability of bin i in histogram H
- All unique bins are defined in a set $B = \{b|\forall_{s \in S} b = bin(H,s)\}$

We define *representativeness* R of a sample set S for a given global dataset D as

$$R(S|D) = \sum_{b \in B} p(H, b) \tag{2.1}$$

where $0 \leq R(S \mid D) \leq 1$. When a sample set has higher representativeness, then R reaches 1.

The definition of representativeness R is theorized for a dataset D which has a single variable or attribute for each data point. The histogram for the dataset D gives us the frequency of data points in each bin that is defined. Once we know which data points fall in which bin and where points in sample set S lie in the histogram, we know which data points are represented by S. These data points are in set B that is the set of bins b where sample points in S lie (i.e., $b \in B$). The data point $x \in D$ is said to be represented by a sample point $s \in S$, when a certain pre-defined criteria is fulfilled.

Thus representativeness can be explained as a fraction of the total number of data points that fall within a predefined threshold criteria for atleast one of the points in the sample set. All data points within the threshold are completely represented by one of the sample points in S. If a data point falls within the threshold criteria of multiple sample set points, then it is represented by the sample set point where criteria is optimal. In Globe, representativeness shows the fraction of the total land surface on earth that are similar to the locations that are part of a case study and have been studied based on a specific set of parameters.

If we use a multivariate dataset with m dimensions, then the criteria used are modified to consider m dimensions. For example, if the criteria is based on

Euclidean Distance, then distance in a single dimension would be

$$d = |(s - x)| \tag{2.2}$$

where

- d is the distance
- s is a sample point such that $s \in S$
- x is a data point such that x $\in$ D

The distance formula for m-dimensional data points would be

$$d = \sqrt{\sum_{i \in m} (s_i - x_i)^2} \tag{2.3}$$

The distance between the sample point $s \in S$ and x shows how close the data point is to the sample point. As $d \to 0$, the data point is considered to be closer. Representation of a data point by a sample point is inversely proportional to the distance. Hence, representation r is defined as

$$r(x|s) = |1 - d_x| \tag{2.4}$$

r is thus a value between 0 and 1 (and maybe greater than 1 in some outlier cases). A scale is created from 0 to 1 and the data points are assigned to each section of the scale (forming histogram H). Representativeness R is taken as the proportion of the total number of data points that are there in the first scale between 0 and 1. This is because the representativeness provided in the definition is for a binary scale where as represetativeness can be in degrees (like in the case of a heat map explained in the next chapter).

The histogram *H* can be of 2 types: equal probability and equal area as shown in the diagram below where *x* is a single variable and *p(x)* is the probability distribution function (pdf).

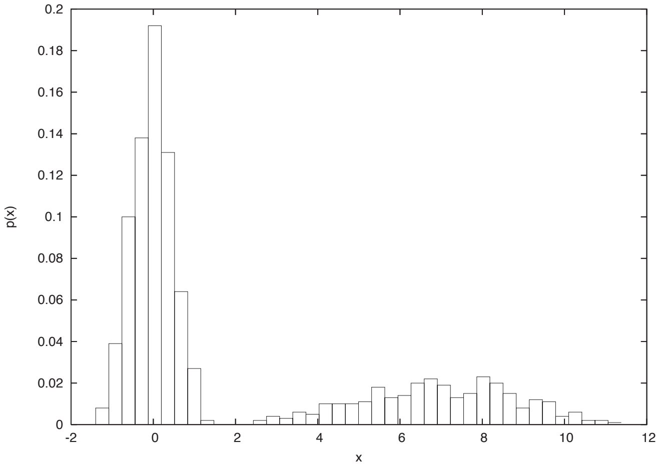


FIG. 2.1. *A histogram of a sample of data from a distribution in which bins have equal area.*

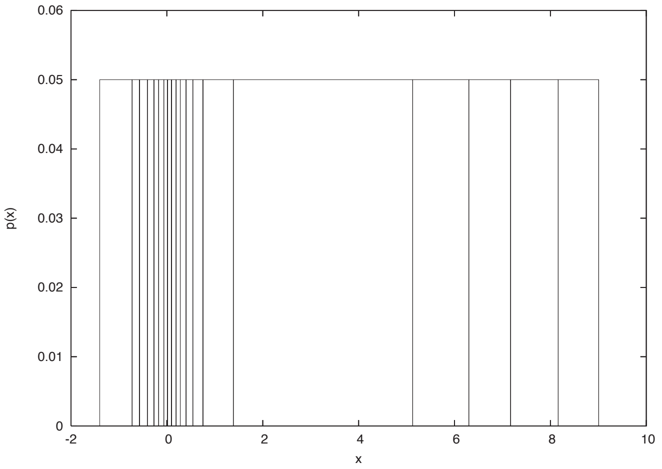


FIG. 2.2. *A histogram of a sample of data from a distribution in which bins have equal empirical probability.*

An equal area histogram is one where the data dimension is divided into bins of equal size (Figure 2.1). Thus each bin contains the points that fall within a certain range in the given dimension. Representativeness *R* in such a histogram can be maximized by choosing mode points. An equal probability histogram is where the number of points in each bin is equal. Hence the size or width of each bin changes according to the density of the points (Figure 2.2). Since the bins are equiprobable, points can be selected from any random bin to represent the entire points in that bin.

2.1.2 Kernels

A kernel function is a function that maps a point onto a scale and is denoted by *K(s - d)*. In the case of a histogram, the kernel function implemented is a step function. It is a set of bins whose points match the threshold criteria (like maximum distance) such that any point in any of the bins represents the all the other points. When the histogram is equiprobable, the step function is used to maximize the number of bins that are part of within the function's limits. The step function is shown in the diagram below.

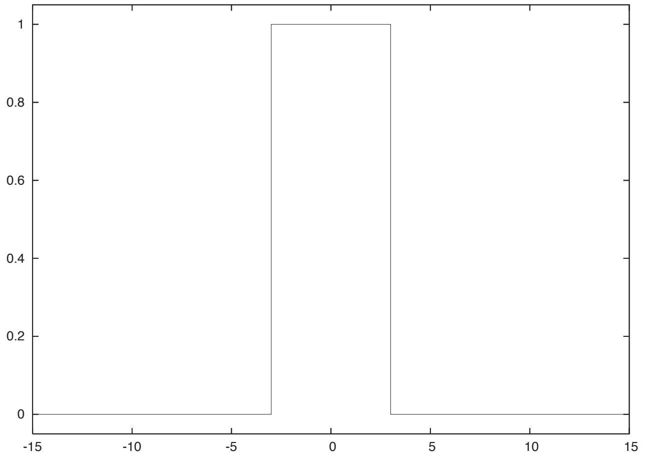


FIG. 2.3. *Sample Step kernel function.*

2.2 Methods to Maximize Representativeness

In this section, we discuss various methods that can be used to maximize represetativeness. The main aim of the methods described below is to find an optimal set of sites or points such that representativeness can be maximized.

2.2.1 Clustering

Clustering techniques are a set of methods to group data points that are similar together [4]. These characteristics of the groups are defined by a pattern of values in their variables. Clustering is an unsupervised learning method. It does not require a training data set to create a model. The groups in which the data points are to be classified need not be known at the start. Hence, clustering can be used for exploratory data analysis to identify patterns in the data. Clustering is a three stage process

1. Extract features from the given set of points.
2. Perform similarity measurement between data points.
3. Create groups based on the similarity measurement.

Clustering techniques are of different types and mainly divided into 2 categories:

1. *Hierarchial Clustering Techniques* - These techniques create groups of points which are similar to each other. Once a group is formed, it creates the next level by combining groups that are similar. In this way, a hierarchy of groups is created with all groups merged at the top most level of the hierarchy. The structure is called a dendogram [4].
2. *Partitional Clustering Techniques* - These techniques try to create a single partition in the dataset as compared to a dendogram which may have a high computation time. The problem occurs when the size of the dataset is large. Partitional techniques try to optimize a certain function based on which the partition is made. Calculating the optimal set of values for the function could again be computationally expensive. Hence an approximation is calculated by executing the algorithm multiple times on the same dataset until the function reaches a state that is *close* to optimal. For example, using squared error a as function to create partitions [4]. The algorithm is executed until the squared error is reduced to a value that is below a certain pre-determined threshold.

2.2.1.1 K-Means Clustering: The k-means clustering algorithm is a widely used algorithm [5]. This is a centroid or partition based clustering technique. The algorithm clusters all the data points into *k* clusters. The algorithm starts by selecting an arbitrary set of centroids $c_1, c_2...c_k$. It then assigns each point to the closest centroid c_i. Once the points are clustered, it calculates the center of mass for each cluster to get a new set of centroids. The previous steps are then repeated for the new centroids. After each iteration the set of centroids moves closer to the final set such that the next iteration does not change the set of centroids chosen. This means the center of mass for the *k* clusters calculated remains constant. The algorithm stops computing after this point. The worst case time complexity is $O(n^{kd})$ [5] where *n* is the number of data points, *k* is the number of clusters and the points are in a *d*-dimensional space.

2.2.1.2 Nearest Neighbor Clustering: This is a hierarchial clustering technique. In this clustering method, the nearest neighbor to each data point is found and the point is assigned to that cluster. A Voronoi decomposition of the data points is performed [6]. There is a threshold or quality function Q_n to put a threshold on the distance that is considered between the point and the cluster. Thus all the points are put into *k* clusters where *k* is user-defined. The clustering is implemented using a graph based structure. Whenever a point closest to the current point is found, an edge is created between them thus linking them in the same cluster [4]. It is also called agglomerative single-link clustering technique and has a time complexity of $O(n^2)$ [7].

K-means clustering and nearest neighbor clustering can be used to find a set of *k* centroids that maximize representativeness. K-means clustering generates *k* clusters with unique centroids that are the representative points. For nearest neighbor clustering, we can select any point randomly from each of the *k* clusters generated (as all clusters adhere to the quality function Q_n), as they represent the other points within the cluster.

2.3 Dimension Reduction Techniques

Consider a data set where each point has a large number of variables. These variables may have different scales of values, and different densities and variances. There are a number of possible problems with high dimensional data [8]:

1. Processing high dimensional data (especially when the number of data points is large) is expensive.

2. Even though the number of dimensions is high, the data could be classified or clustered using a smaller subset of variables.

3. As the number of dimensions increases, the values for some variables may become sparse. This is known as the *empty space problem* [8].

4. The *Curse of Dimensionality* states that the number of sample points required to approximate a function increases exponentially as the number of variables / dimensions increases.

A dimension reduction technique is a transformation which reduces number of dimensions required to represent a sample. The reduced set of dimensions may be a subset of the original set of dimensions (for example, using information gain) or could be a completely new set of dimensions. Some of standard dimension reduction techniques that can be used to transform a high dimensional data set are Principal Component Analysis (PCA), and Self Organizing

Maps (SOM)[8]. Neural Networks with GIS have also been used for constructing a *Land Transformation Model* which tries to forecast how usage changes [9]. Self Organizing Maps have been used to perform environmental assessment of regions, grouping based on environmental conditions, and finding out which areas might deteriorate in the future [10]. Once a SOM is trained, the *k* nodes from the weight vector can be used as centroids representing their respective clusters. As the nodes may not be actual data points, the point closest to each node will be used as a representative point. Training a SOM may require updating the weight vector over several iterations of the data set. The time complexity of a SOM is $O(|E| \bullet (|E| + |V|))$[11].

2.3.1 Clustering Using a Combination Of Methods

Hoffman et. al.[12] use *Multivariate Spatio-Temporal Clustering (MSTC)* to calculate representativeness of sampling networks. MSTC can be performed using a combination of PCA and K-Means clustering[12]. The data set considered is high dimensional and is assumed to contain a lot of redundant information. Hence the method involves reducing the number of dimensions using PCA at the beginning and then performing standard k-means clustering. Hoffman et. al. also provide a set of improvements for performing PCA and k-means clustering. The time required to perform k-means clustering is reduced by decreasing the number of distance computations between the centroid and the other points, based on cluster created and new distances computed. The time complexity of PCA computation is reduced by parallelizing it. The summation of all euclidean distances from points to their nearest sample locations or centroids, is used to measure representativeness of the sample set. Higher the sum, lower is the representativeness of the sample set.

In the next chapter, we introduce PCA and our method to calculate representativeness, as well as how we find new locations such that representativeness can be improved.

Chapter 3

PROJECT ARCHITECTURE

This chapter discusses the goals & objectives of the Globe project, the architecture of the current prototype, examples of how the system works and describes the kind of data that is used. This helps us understand the workflows in the next chapter as well as the examples used while performing experiments.

3.1 Globe Project

During the course of scientific study on climate change, there have been improvements in creating models for the climate and the environment, but the effect of human activity is not directly observable [1]. Scientists find it difficult to create models at the global level because comprehensive models at the local and regional levels have not been created. *Coupled Human and Natural Systems (CHANS)* [1] involves research in understanding the effects of human activity on the Earth. Research in the *Land Change Science* community involves conducting two types of case studies for this purpose. *Local case studies* investigate changes in the environment of a specific locality and observation of local land managers. These case studies are normally limited to a pre-defined area. The other type of studies are *regional case studies*, which involve combining local observations and remote sensing data to analyze land change beyond the defined boundaries of a locality [1]. Using these cases, and regional and generic global information, *meta-studies* are created which aggregate knowledge at a global scale using quantitative methods.

The Globe project, as introduced in the first chapter, is a project to maintain old workflows as well as to create new workflows in Land Change Science (LCS). It provides a set of quantitative tools to help practitioners analyze case studies better. These include methods to calculate the similarity between case studies, measure how relevant a case study is, and visualize globally the coverage of a case study across the world. The major objectives of the project are:

1. To create an online environment where researchers can collaborate.
2. To find methods and metrics to evaluate workflows.
3. To create a framework where information from case studies can be globally stored and retrieved to futher analyze the data. Our work tries to find a method to generalize case studies by finding similar locations like the ones mentioned in the studies for a predefined set of parameters.

3.2 Globe Architecture

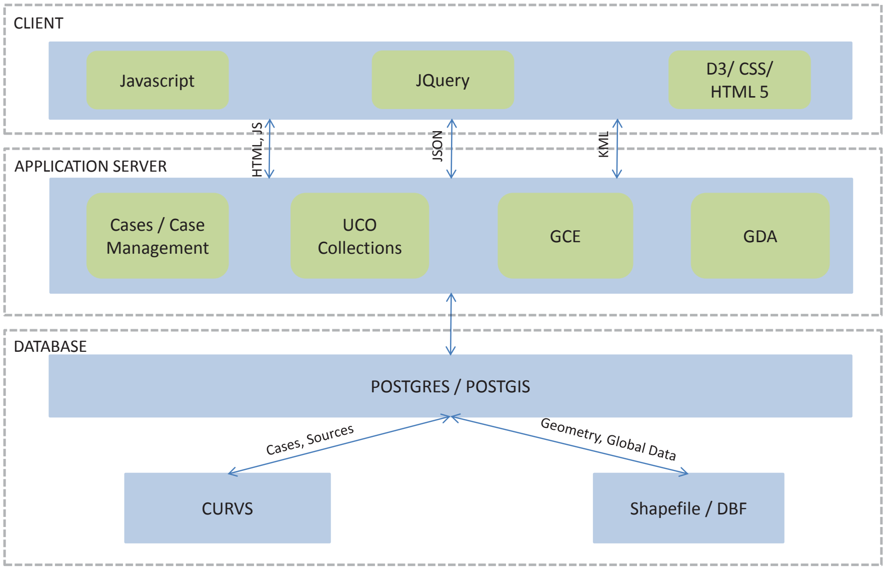


FIG. 3.1. *Globe Architecture*

As the diagram above shows, the Globe architecture consists of three distinct layers:

1. **Database Layer**: This layer maintains all the raw global information and case geometries available.

2. **Application Server**: The current application server used is Glassfish [13]. It contains all the function modules for managing cases, collections, the global data, and the global correlation engine.

3. **Client Layer**: This contains the website that is visible to the end user. The web layer is constructed in Javascript and HTML5, and uses D3 [14].

3.2.1 Database Layer

The database layer uses a **Postgres** object-relational database management system, i.e., it is similar to a relational database management system but with object oriented design where classes can be directly mapped to database schema. Table data can be accessed using standard SQL commands. It implements all properties of ACID (atomicity, consistency, durability and isolation) and is an open source database available on most operating system platforms including various flavors of Linux and Windows. The version of Postgres SQL used is 9.1.

PostGIS is an open source software project which is used for working with geographical data. It can be used to maintain geospatial databases and is built on PostgreSQL. It follows the published standards from the Open Geospatial Consortium (OGC). The package provides many useful functions such as storing geometries of geographical areas, calculating metrics such as distance and area of a region, and data structures to store geometries to efficiently search them.

The data is available in the form of shape and DBF files. DBF Files are database files (started by dbase) used to store information in the form of tables. The shape file format is used to store geospatial data in files and is used by Geographical Information Systems (GIS). The shape file format consists of a set of files which include files to maintain the main geometries, indexes, the attributes associated with these geometries, and co-ordinates of the geometries. The current global dataset contains around 45 variables. There are six categories of variables Surface, Climate, Human, Biological and Remote Sensing[3].

3.2.2 Application Server

Globe is a web-based application which is hosted on a *Glassfish* application server. *Glassfish* is an open source application server by Oracle Corporation. The application performs the following functions:

1. **Case Management**: This module works with all the meta case studies information. It provides the API to perform data management activities.

2. **Collections**: A collection is a set of case studies that are used in a workflow. *Collections* maintains these sets.

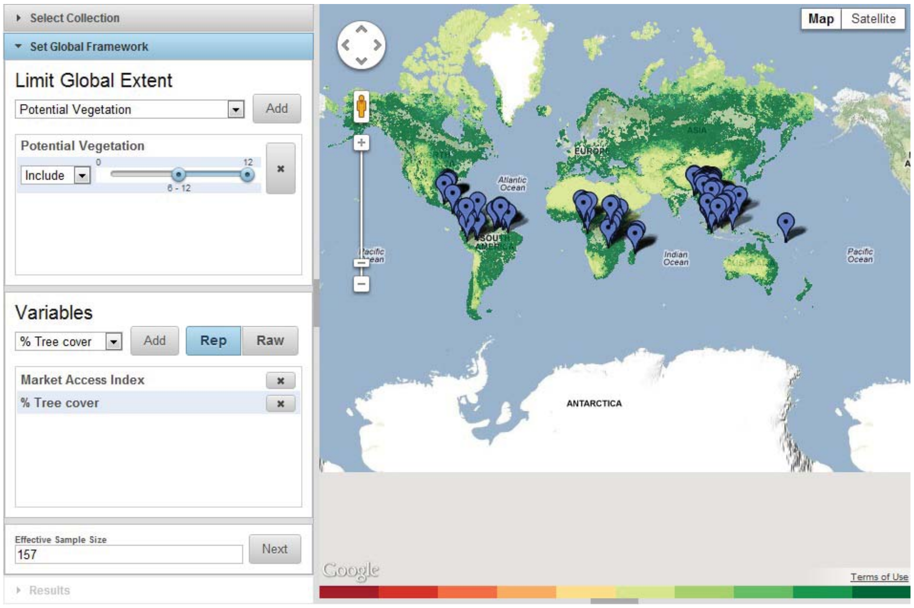
3. **GCE**: The GCE is the *Global Correlation Engine* which performs two main functions of Globe, i.e., similarity and representativeness computation. Similarity measures how similar the other regions of the world are as compared to a region in a case study based ob a set of global variables. Representativeness, as introduced in the previous chapter, calculates how much a case study covers the rest of the world for a selected set of variables from the set of global variables. Our work focuses on measuring representativeness.

4. **GDA**: This is the global data array. It maintains all the global Data in memory. The information is stored in the form of two data structures, a quad tree and a floating point matrix. The floating point matrix is used to store the global variable information of every region on the world map. The world map is divided into regions depending on the resolution at which the map is viewed. For the current maximum resolution that can be visualized in Globe, the world map is divided into a total of about 1.4

million grid cells, which is the size of the matrix. This is part of the *Discrete Global Grid* that is explained in the following section.

3.2.3 Client / User Interface Layer

The client is web-based client and constructed in HTML5 and JS. The data is obtained from the server dynamically using AJAX. The maps are overlaid onto Google Maps, and charts are constructed using D3 [14]. D3 is a Javascript library which manipulates documents based on the data that is provided. It is used for the purpose of generating graphs and visualizations. An example image of the web client is given is below:



Limit Global Extent: Potential Vegetation

Potential Vegetation: Include 6 - 12

Variables: % Tree cover

Market Access Index

% Tree cover

Effective Sample Size 157

FIG. 3.2. *Globe Web Client*

3.2.4 Constructing Google Maps

The *Discrete Global Grid* (DDG) [15] divides earth into a set of regions which are hexagonal. The number of hexagons and the area covered by each hexagon changes according to the resolution which is used. At the smallest resolution, each hexagon is represented by point which is the center of the hexagon. For the Globe project, the resolution used is ISea3H level 12. The total number of cells are *5,314,412* and the area of the each hexagon is *95.978* km^2 Google Maps as shown in the previous figure is used to display the representativeness of sites. Google Maps is made up of tiles as shown in the next figure.

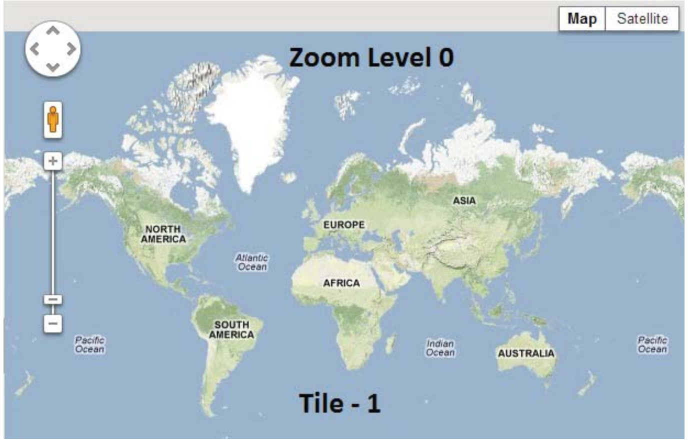


FIG. 3.3. *Zoom Level - 0 Google Tiles*

Each tile is of size 256x256 pixels. The request sent by Google Maps contains three parameters z, x and y, where z is the zoom level and (x,y) is the specific tile upon which an image is displayed. The image is constructed in the form of a bitmap and then converted to a PNG file. Thus the tiles generated are superimposed over the google map imagery. When the request is sent for a specific tile, the Globe server maps the tile's location to a latitude and longtitude co-ordinate for its boundaries. The specific regions of the world which fall within the range of the coordinates are searched.

Spatial Indexing methods are used to manage spatial data (which could be multi-dimnesional) points and to help improve the performance of retrieving geometries or regions from the defined space. Hierarchial data structures are used for the purpose of spatial indexing. They employ the *divide and conquer* strategy. The data structure contains the information of the entire space (the world map in this case) at top (or root) level and they subsequently divide each region into smaller regions recursively [16]. When a predefined depth is reached, the data point is stored. The quadtree is a type of hierarchial data structure which is a tree structure having exactly four child nodes. Each node contains the data point, or in case of Globe, a pointer to the data point in the global matrix. There are many types of quadtrees [17] viz. region quadtree, point quadtree, edge quadtree, polygon map quadtree. In Globe, a point quadtree is used. In a point quadtree each node stores the location of a point or *Global Land Unit (GLU)* which is the centroid of its respective hexagon.

Once the hexagons / GLUs within the range of coordinates are found, they are colored based on a value (between 0 & 1) calculated after a representativeness computation is performed for each GLU. The color is mapped linearly to the values, i.e., the scale between 0 & 1 is divided into equal parts based on the number of colors in the color pallete and then the GLU's are assigned a color based on which bucket the value falls into. The PNG which is created is then sent to Google Maps for rendering.

Chapter 4

A DIMENSION REDUCTION TECHNIQUE FOR CALCULATING REPRESENTATIVENESS

This chapter discusses Principal Component Analysis (PCA) in detail, the steps required to compute it, the method used to select points to maximize representativeness, and how the final metric of representativeness is calculated.

4.1 Preliminaries

Listed in this section are the parameters that need to be calculated to perform PCA.

4.1.1 Standard Deviation

The standard deviation gives us an idea of the spread of a distribution. It describes the variation in data on both sides on the mean. A standard deviation *SD* for given dataset *D* is

$$\sigma(a) = \sqrt{\frac{\sum\limits_{x \in D} (x_a - \bar{a})^2}{N - 1}} \tag{4.1}$$

where *N* is the number of data points, *a* is the variable, x_a is the value of variable a in data point x, and $\bar{a}$ is the average of variable a. In a multivariate dataset, the standard deviation is calculated for every variable.

4.1.2 Variance

Variance is defined as the square of the standard deviation.

$$\textit{VAR(a)} = \sigma^2(a) \tag{4.2}$$

Like the standard deviation, it is calculated separately for each variable.

4.1.3 Covariance

Covariance is a generalization of the concept of variance for two dimensions. It defines how the two dimensions co-vary. In our example, the two attributes are a & b, Covariance is given by the following formula

$$\textit{COVAR(a,b)} = \frac{\sum_{x \in D} (x_a - \bar{a})(x_b - \bar{b})}{N - 1} \tag{4.3}$$

where *N* is the number of data points. The covariance calculation for two dimensions forms a 2x2 matrix. For 2 dimensions a & b, it is given in the form[18]:

$$\begin{pmatrix} \textit{COVAR(a,a)} & \textit{COVAR(a,b)} \\ \textit{COVAR(b,a)} & \textit{COVAR(b,b)} \end{pmatrix}$$

4.1.4 Eigenvalues & Eigenvectors

A linear transformation on a vector is a transformation that maintains the property of additivity and homogenity. An Eigenvector is a vector that remains unchanged after a linear transformation. The only change that the vector undergoes is a change in magnitude. The scalar magnitude by which the vector changes is called the Eigenvalue. The Eigenvector is orthogonal to the original vector [19]. Consider a matrix A, then

$$Ax = \lambda x \tag{4.4}$$

where x is the Eigenvector and λ is the Eigenvalue.

4.2 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a technique used to analyze multivariate datasets and to find patterns in the data. It is used as a dimension reduction technique where higher dimension data points can be projected onto a lower dimension space [20]. It takes a data set of n dimensions and projects it onto a set of new dimensions (which again can be a maximum of n) that are orthogonal to each other (although all n need not be used). The new dimensions are not correlated with each other. They are called *Principal Components*. The principal components are calculated by performing an eigen decomposition of the covariance matrix. The covariance between dimensions in the original dataset is used because those dimensions which have a higher variance, provide more information regarding the nature of the dataset [21]. Another reason to use PCA is because it is easy to understand and efficient algorithms exist to compute each step [21]. It is also a widely used method because it is able to reduce noise and the dimension reduction performs data clustering [22]. As we shall see, the cost of computing distance is also reduced as the final distance is in a single dimension.

Consider the following example dataset which we will use to explain various concepts and PCA as well as show the steps which are required to perform PCA. The dataset has 10 points has given in table below:

Data =

a	**b**
2.8	1.8
3.5	2.9
4.9	3.5
6.2	8.8
8.5	6.7
1.6	5.5
3.1	4.3
3.3	3.2
2.1	8.1
9.2	7.5

Table 4.1. *Sample Data for performing PCA*

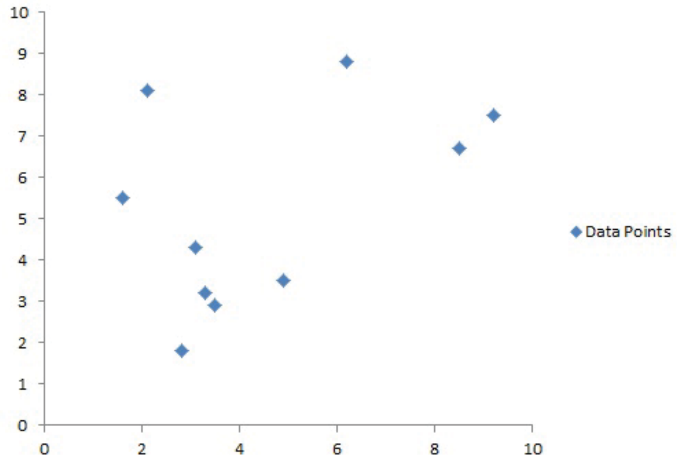


FIG. 4.1. *Sample Data*

The following steps are required to perform PCA [18]

Step 1

Calculate the arithmetic mean for each dimension and then subtract the mean from the every data point. The arithmetic mean for a & b are

Mean(a) = 4.52 & Mean(b) = 5.23

The table below shows the data points after the values are adjusted with the mean.

Newdata =

a	**b**
-1.72	-3.43
-1.02	-2.33
0.38	-1.73
1.68	3.57
3.98	1.47
-2.92	0.27
-1.42	-0.93
-1.22	-2.03
-2.42	2.87
4.68	2.27

Table 4.2. *Sample Data after Subtracting Mean*

Step 2

Calculate the covariance matrix using equation 4.3 [18]. The covariance matrix is

$$\begin{pmatrix} 6.95 & 2.90 \\ 2.90 & 5.94 \end{pmatrix}$$

The standard deviations calculated using equation 4.1 of attributes a & b are

SD(a) = 2.63, SD(a) = 2.43

The variances can be calculated using equation 4.2 of attributes a & b

VAR(a) = 6.95, VAR(a) = 5.94

The variances for a and b, as seen from this example, is present in the covariance matrix along the diagonal.

Step 3

Calculate the Eigenvalues and Eigenvectors for the covariance matrix [18]. For our example they are:

$$\text{Eigenvalue} = \begin{pmatrix} 3.50 \\ 9.40 \end{pmatrix}$$

$$\text{Eigenvector} = \begin{pmatrix} 0.644 & -0.765 \\ -0.765 & 0.644 \end{pmatrix}$$

Step 4

The first principal component is the vector with the highest eigenvalue [18]. In our example, the second vector has an the highest corresponding eigenvalue of *9.40*. We can use it as a vector on which all the data points are projected.

Step 5

Project the data points on the new dimension by using the formula

$$\textit{FinalData} = \textit{MeanAdjustedData} \bullet \textit{PrincipalComponent} \tag{4.5}$$

The equation calculates the dot product the *Mean Adjusted Data* and the principal component used. The table below shows the final value of each data point and figure 4.2 shows the data points.

FinalData =

First	**Second**
3.524	1.516
2.280	1.125
0.823	1.568
-3.584	-1.649
-3.991	1.438
2.060	-2.086
1.685	-0.202
2.240	0.767
0.003	-3.754
-5.042	1.277

Table 4.3. *Final Adjusted Data along First Principal Component*

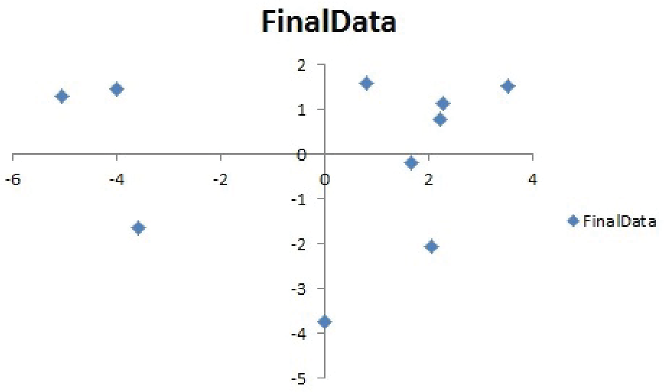


FIG. 4.2. *Final Adjusted Data*

4.3 Measuring Representativeness Using PCA & Histograms

The following section provides the details of the algorithm to calculate Representativeness of a given sample set of regions and how to select a new set of sites to improve representativeness across the entire world for the selected set of attributes or variables. Land Change scientists know many regions across the world are not represented by sample sites in their case studies by design. In *Globe*, they are provided the option to filter out these regions, so that they are not considered as part of the analysis. Hence we apply our method to the list of unfiltered regions.

The following diagram shows the algorithm's workflow:

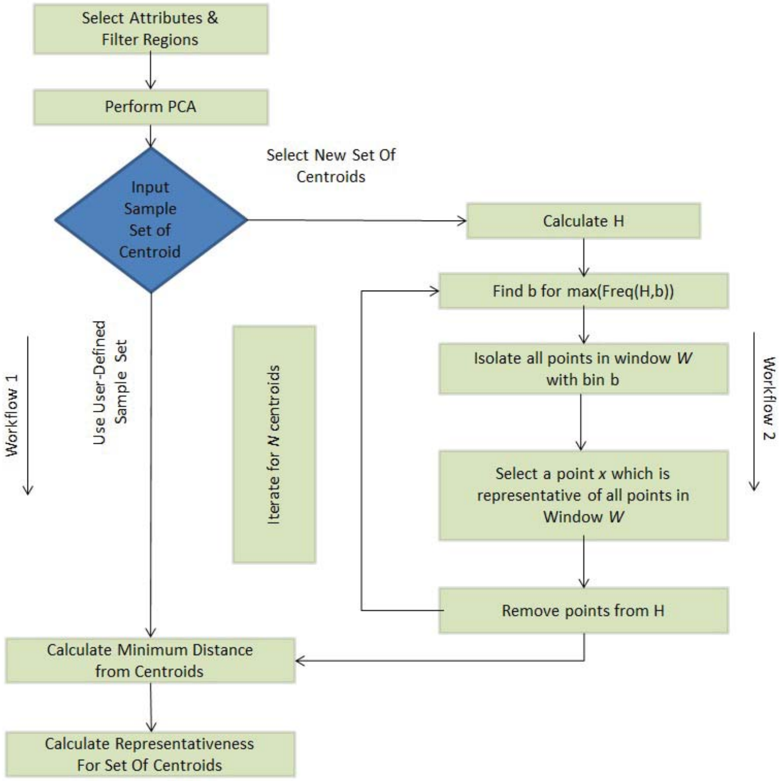

Select Attributes & Filter Regions
Perform PCA
Input Sample Set of Centroid
Select New Set Of Centroids
Calculate H
Find b for max(Freq(H,b))
Isolate all points in window *W* with bin b
Select a point *x* which is representative of all points in Window *W*
Remove points from H
Workflow 1
Use User-Defined Sample Set
Iterate for *N* centroids
Workflow 2
Calculate Minimum Distance from Centroids
Calculate Representativeness For Set Of Centroids

FIG. 4.3. *Workflow to Calculate Representativeness*

As shown in the diagram above, the algorithm has two distinct workflows to solve two objectives:

1. The first workflow calculates the representativeness for the user-defined Sample Set of regions where case studies were conducted.

2. The second workflow provides a method to select *N* regions to maximize representativeness for the user-defined set of attributes. These are the places where the system can recommend that a study could be conducted. Once the regions are selected the representativeness of these sites is calculated.

Both workflows use a distance metric to calculate how close a location is to another location where a study was conducted.

$$\mathit{FD(x)} = \mathit{min}(D_p(x) - D_p(s)_{s \in S}) \tag{4.6}$$

where *x* is a specific location across the world and *s* is a place in the world where a study was conducted. *s* is part of a larger Sample Set *S*. D_p is the projection of a location (or data point) onto the first principal component. The final distance *FD(x)* is the minimum distance between *x* and *S*.

Representativeness of the Sample Set, as given in equation 2.1 can now be transformed to the following equation:

$$\mathit{R(S \mid D)} = \frac{\sum\limits_{x \in C} \sum\limits_{b \in W} Freq(H, b)}{D} \tag{4.7}$$

where *R(S | D)* is the representativeness of Sample Set *S* and *D* is the complete dataset. *H* is the histogram of D_p. *Freq(H, b)* is the number of points in (or the frequency of) any bucket *b* in histogram *H*. The window size *W* that is a set of buckets *b*. A selected region represents all regions in *W*. *W* can be user-defined. *C* is the set of regions or centroids selected.

4.3.1 Preconditions

Before any of the following workflows are executed, the user selects a subset of the global variables (such as temperature, %tree cover) for which representativeness is measured. Also, the user is allowed to filter out a set of regions from the dataset. For example, the user may choose to limit the dataset only to regions which are in the tropics (i.e., having a temperature between 15^oC and 31^oC). Once the data is filtered, we get a final list of regions and matrix containing the value of each user selected variable for each of these regions.

4.3.2 Workflow 1 - Calculate representativeness of given sample set

The steps to calculate representativeness of a sample set of sites (equation 4.7) that are provided by the user are:

Step 1

Select the a set of variables based on which representativeness is measured. Calculate the eigenvalues and eigenvectors for all the unfiltered regions across the world. If the unfiltered regions do not contain the sites that are part of the Sample Set, then find the first principal component and project all the regions (including the sites) onto this dimension.

Step 2

Calculate the distance between each region and a region in the Sample Set. The Final Distance is the minimum distance found. It shows which location in the Sample Set is closest to the current location being considered and how close it is.

Step 3

Display all the locations on Google Maps with a color chosen according to the final distance calculated. All the distance are a value between 0 & 1. A color pallete consists of a set of shades (e.g. from Green to Red where Green is considered as a place completely represented by one of the sample sites while Red is completely unrepresented). The scale between 0 & 1 is divided based on the number of shades which are in the color pallete. The color associated with the bucket in which the final distance of a place falls, is applied on Google Maps, on top of the location of that place. The following diagram (figure 4.3.2) is an example of how the color scheme looks.

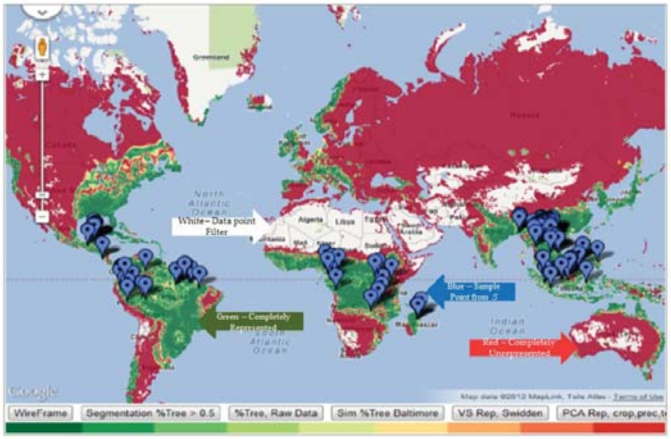


FIG. 4.4. *Representativeness Coloring Scheme*

4.3.3 Workflow 2 - Creating a set of *Ideal* sites

An *ideal* set of sites is a set of sites that maximizes representativeness but at the same time has a minimum number of sites required to reach that measure of representativeness. The steps to find a set of *ideal* set of sites that are representative of all other regions in the world are:

Step 1

As seen in the previous workflow, calculate the eigenvalues and eigenvectors for all the unfiltered regions across the world.

Step 2

Once the regions are projected onto the first principal component, create the histogram of that dimension. The histogram requires a number of buckets into which the dimension is divided. The number of buckets is defined after testing. Once the histogram is constructed, a window of size *W* is defined such that

$$1 \leq \mathrm{W} \leq \textit{number of buckets}$$

The window is a set of buckets (containing regions) such that any region in the bucket represents all the regions in the buckets which are present in *W*. *W* is either user-defined or an arbitrarily fixed size. The algorithm performs the following steps *N* times where *N* is the number representative sites to be found.

Algorithm 1: Finds M points to maximize Representativeness of Points

```
**Input**: A finite set $FB_P = \{fb_1, fb_2, \ldots, fb_m\}$ of Frequency of
        buckets in Histogram $H$, window size $W$
**Output**: A finite set $C$ containing a set of $N$ representative Points
*rand()* is picks a random point from a bucket $b_k$
*WUsed* is bit array of size $\lfloor \frac{B_P}{W} \rfloor$
$B_P$ contains points each bucket. $B_P = \{b_1, b_2, \ldots, b_m\}$
$MW$ is the window with Maximum Frequency.
for $i \leftarrow 1$ to $\lfloor \frac{B_P}{W} \rfloor$ do
    $WUsed_i \leftarrow false$
for $i \leftarrow 1$ to $N$ do
    $max \leftarrow -1$
    $j \leftarrow 1$
    $MW \leftarrow -1$
    $maxBin \leftarrow -1$
    while $k \leq n$ do
        if $(fb_k \geq max)$ && $(\text{WUsed}_j = false)$ then
            $max \leftarrow fb_k$
            $maxBin \leftarrow b_k$
            $MW \leftarrow j$
        if $mod(k, W) = 0$ && $(k/W) \geq 0$ then
            $j \leftarrow j + 1$
        $k \leftarrow k + 1$
    $WUsed_{MW} \leftarrow true$
    $P \leftarrow rand(b_{maxBin})$
    $C_i \leftarrow P$
return $C$
```

Algorithm 1 shows how an *ideal* set of sites is selected. Consider a histogram *H*, with window size *W*. Let B_P contains points each bucket. MW is the window with the maximum frequency. $\lfloor \frac{B_P}{W} \rfloor$ is the total number of windows in histogram *H*. We maximize representativeness by selecting a single point from a window. Once a window is utilized, it not used again. This is because a point from a given window of buckets represents all the points in the window completely. *WUsed* is an array of bits that shows which of the windows have used. *WUsed* is initialized to $false$. Then, the algorithm iterates through each bucket in the histogram and finds the bucket which has the maximum frequency (mode) and that is part of a window that has not been used before (i.e., where *WUsed*$_i$ is $false$) . A bucket b_k is part of a window $\lfloor (k/W) \rfloor$. If bucket b_k is the bucket with the maximum freqeuncy, then make *WUsed*$_{\lfloor (k/W) \rfloor}$ as $true$. Then, select any point from b_k and add it to the final list of centroids *C*. The above set of steps performed to find a centroid, is repeated *N* times to get a set *C* containing *N* centroids.

The algorithm has to take care of a specific condition i.e. the case when the number of centroids required is higher as compared to the number of bins that are greater than 0. In such a case, the number of centroids returned is limited to number of non-zero bins that can be found.

Step 3

This step is the same as the previous workflow, where the regions selected are displayed on a World Map with the representativeness of the entire set.

4.4 Optimization

To improve the accuracy and the runtime efficiency of the method, optimizations were performed.

4.4.1 Covariance Calculation

The standard method to calculate the covariance value between two attributes is given in equation 4.3. This method requires two passes over the entire data, i.e., the first pass to calculate the average for each attribute and then to calculate the variance. Instead we can reduce equation 4.3 to

$$\textit{COVAR(a,b)} = \frac{\sum\limits_{x \in D} x_a x_b}{N-1} - \frac{\sum\limits_{x \in D} x_a \sum\limits_{x \in D} x_b}{N(N-1)} \tag{4.8}$$

Now, $\sum\limits_{x \in D} x_a x_b$ and $\sum\limits_{x \in D} x_a \sum\limits_{x \in D} x_b$ can be calculated in a single pass of the data, while the number of elements for which it is calculated is counted.

4.4.2 Data Sphering

The final adjusted data (i.e., equation 4.5) can be affected by attributes which have very large values as compared to others, even though the variance in the attribute is low. Hence, before performing PCA, we sphere the data, i.e., change the scale of the data in each attribute so that values are between 0 & 1. We use the following formula for this purpose:

$$\textit{SphereData} = \sum_{x \in D} \sum_{i \in M} \frac{(x_i - \bar{i})}{SD(i)} \tag{4.9}$$

where *M* is the set of attributes, *SD(i)* is the standard deviation of attribute *i* from equation 4.1. $\bar{i}$ is the average value of the attribute *i*.

4.4.3 Computing on Reduced Data Set

Even with the optimizations described above, computing PCA values and then the final distances for about 1.4 million regions is time consuming. This is because as the attributes selected change for every request (user-input), the covariance matrix and the final adjusted data have to be recomputed. Hence to reduce the computation time even further, the 1.4 million regions were reduced to approximately 160,000 regions that represented the same distribution. The 160,000 regions are created by reducing the resolution of the map and increasing the size of the each the hexagons. The number of hexagons is based on ISea3H levels 10 and 12. Then, the center of each these hexagons is used to represent all the regions that are part of the hexagon.

Chapter 5

EXPERIMENTAL RESULTS & ANALYSIS

This chapter discusses the results of experiments conducted and the effectiveness of the methods described in the previous chapter compared to random sampling. Land Change scientists also compare their sample sites against random sampling, as a standard practice. Each experiment conducts 3 sets of tests. The first test is to calculate the representativeness of the sample sites given by the user. The second test uses the *histogram* method discussed in the previous chapter to generate a new set of sites (the number of sites is equal to that defined by the user) where a study can be conducted. It also calculates the representativeness for the same. The third test generates representativeness for a random set of sites for the same number of sites as in the previous two cases. Random sampling is performed 1000 times to eliminate any bias created from a limited set of random sampling tests. We define the ideal sample set as the set of samples generated by the histogram method 1.

5.1 Measuring Representativeness

The Van Vliet study has been used for the purpose of conducting experiments [3]. This meta-study conducts a global assessment of swidden cultivation i.e. slash and burn. It is an agricultural technique where forested areas are burned to create fields for agriculture. There are a total of 157 sites that are part of the study. These sites are used as centroids in our experiment to measure its representativeness. The window size for all experiments below is 1 and number of bins in the histogram is 157.

Figure 5.1 shows the location of sites that are part of the Van Vliet Study.

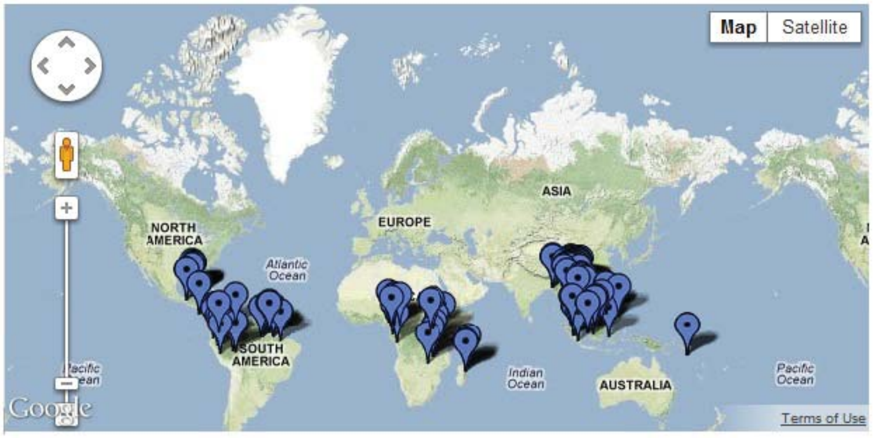


FIG. 5.1. *Sites in Van Vliet Study*

5.1.1 Measuring with a Filter

The following parameters are applied in the experiment:

1. Filter: A filter is applied so that regions of the world the author does not claim to represent are not considered in the analysis. In the example, the parameter *potential vegetation* is used to filter the data set. Potential vegetation has a range of values form 0 to 12 [23]. The values considered in the experiment are from 1 to 2. These values are used filter out all regions except the tropical regions (and some forested areas) across the world.

2. Variables: A total of 3 attributes or variables are used in the experiment viz. *potential vegetation*, *market access* and *temperature*. The selection of the variables is based on the study.

3. Zoom Level: The zoom level decides the total number of points which are considered in the dataset. For example, at zoom level 4, the total number of points or regions is 160,000. The experiments are conducted at zoom level 6. The total number of points at this level is about 1.4 million. When the filter is applied, the total number of *Global land units GLU's* is reduced to about 250,000.

4. Color Scale: The color scale applied has a total of 10 colors from red to green. It depicts the different levels of representation in the *heat map*.

The figures 5.2, 5.3, and 5.4 depict the *heat map* generated for the Van Vliet study. The color scale is provided at the bottom of each image. The scale goes from red to green, where green depicts "complete represention" and red depicts "complete non-representation". The parts of the world that are filtered out are shown in dark blue. This contains all the water bodies and the other regions of the world that do not fit the filter criteria.

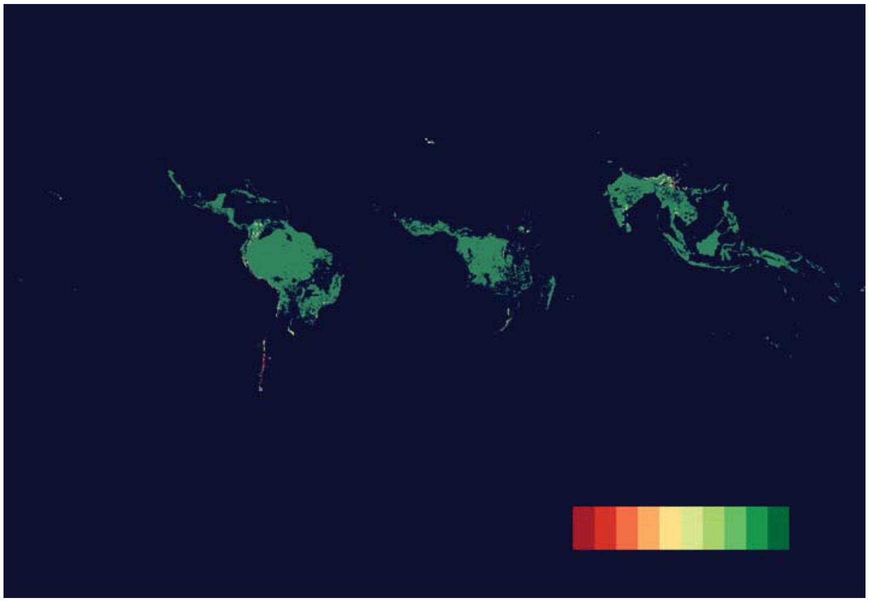


FIG. 5.2. *Representativeness of Filtered World Regions Using Given Samples*

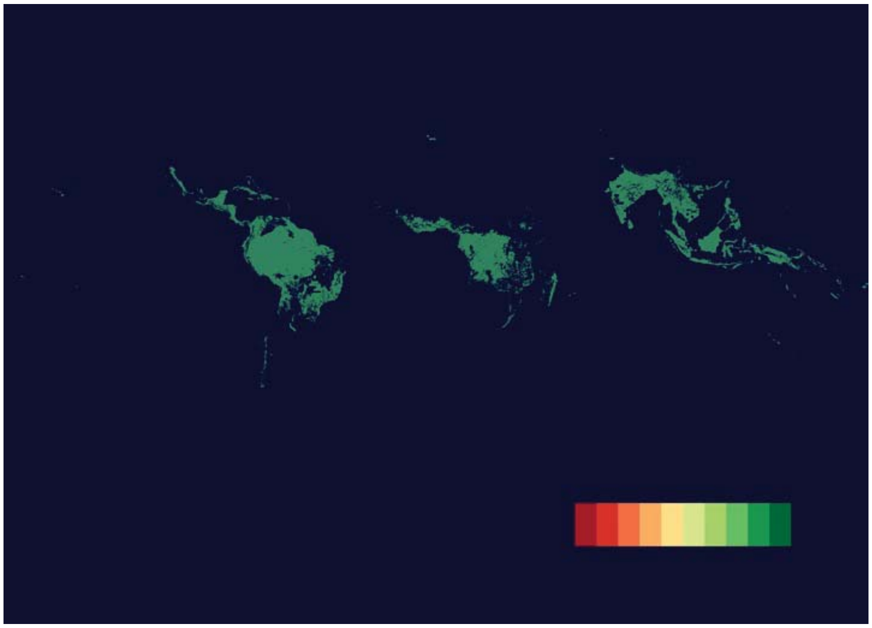


FIG. 5.3. *Representativeness of Filtered World Regions Using Ideal Samples*

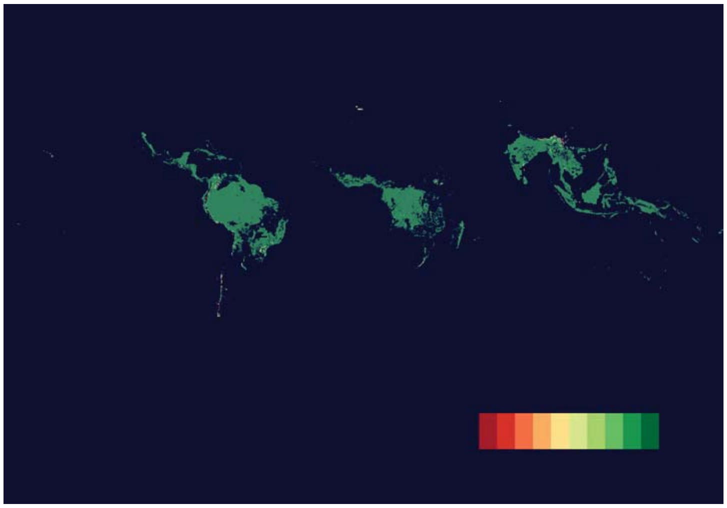


FIG. 5.4. *Representativeness of Filtered World Regions Using Random Samples*

Figure 5.2 shows some regions that are not represented by the locations in the study. The maps generated using given, random and ideal samples are almost the same. This is because of the definition of representativeness. To maximize representativeness, the sampled regions have to "cover" as much of the filtered regions as possible. To represent a set of regions that have similar conditions, only a single point or location is required. Based on the filter used, the potential vegetation, temperature and market access parameters are of similar areas as can be seen on the map. This makes regions cluster with high frequency in specific regions in the PCA space, thus making the number of locations required to represent them lower than 157. This is seen in the histogram in figure 5.5 where the high frequency bins require only a single location in the bin to represent the entire bin. Thus random sampling and the histogram method to calculate new locations have representativeness values that are close to *1.0* or 100%.

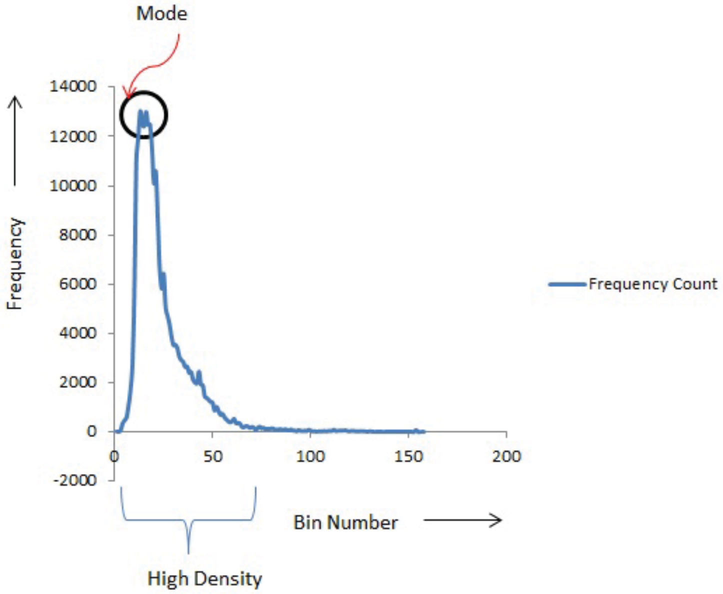


FIG. 5.5. *Histogram Of First Principal Component Values (For Filtered Data) of 1.4 Million Regions*

Figure 5.7 shows the comparison of representativeness of the given sample and the ideal samples generated against random sampling conducted. Representativeness for the given sample set at 48^{th} percentile while the ideal sample set is at 100 percentile. It means that the given sample set is better than random sampling only 48

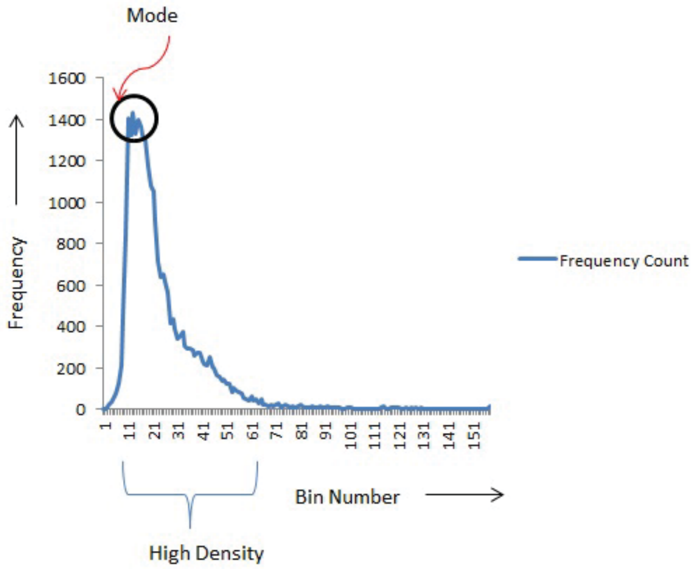


FIG. 5.6. *Histogram Of First Principal Component Values (For Filtered Data) of 160000 Regions*

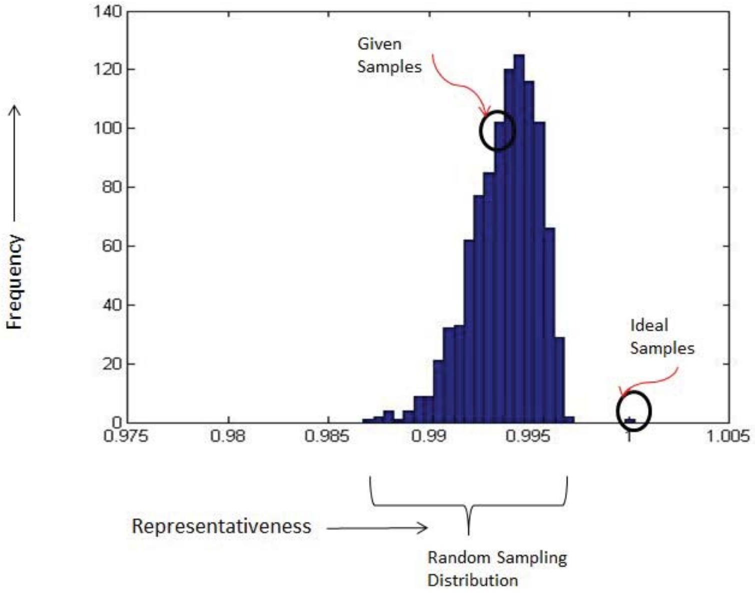


FIG. 5.7. *Histogram Of Representativeness for Random Sampling and Where Other Methods Lie*

Method	Representativeness
Given Sample	0.994
Ideal Sample	1.0
Avg. Random Sampling	0.9937

Table 5.1. *Representativeness Of Samples for Filtered Data*

Figure 5.6 shows that the same histogram trend is maintained when we perform PCA on a downsized dataset. Hence the calculations made on a ISea3H level 10 are applied to the ISea3H level 12 hexagons. Thus representativeness, based on a downsized number of regions, does not change as nature of distribution remains the same. The total number of regions for the downsized set after filtering is 27883.

5.1.2 Measuring without a Filter

This section describes an experiment to compare the various samples without a filter being applied. The data set is the complete 1.3 million regions across the globe. The world maps generated show a clearer differentiation in the given sample representativeness against that of an ideal and a random sample.

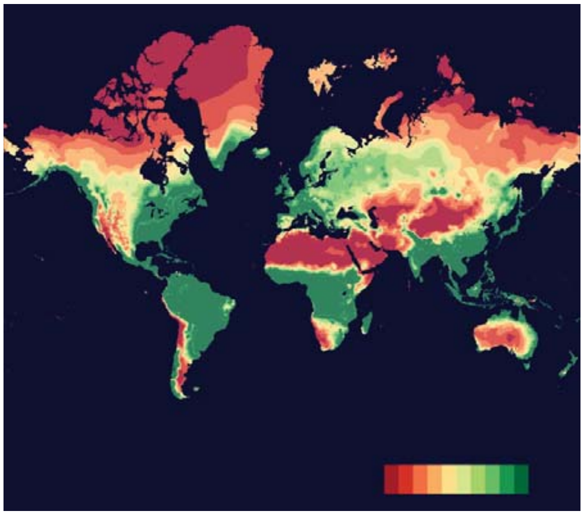


FIG. 5.8. *Representativeness of Unfiltered World Regions Using Given Samples*

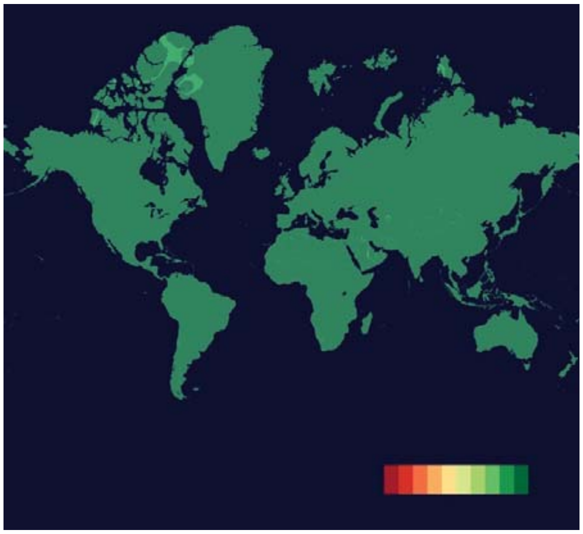


FIG. 5.9. *Representativeness of Unfiltered World Regions Using Ideal Samples*

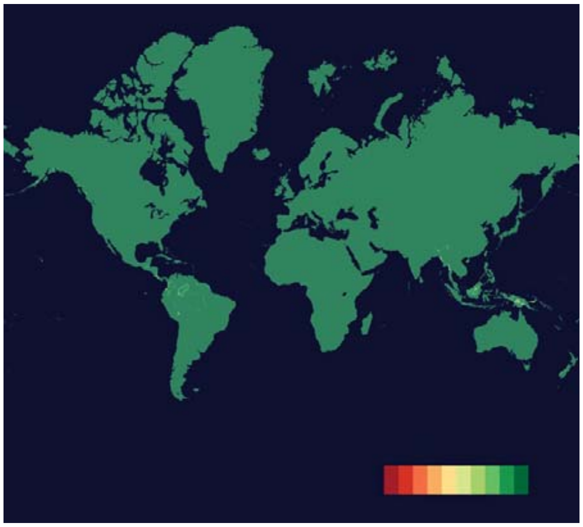


FIG. 5.10. *Representativeness of Unfiltered World Regions Using Random Samples*

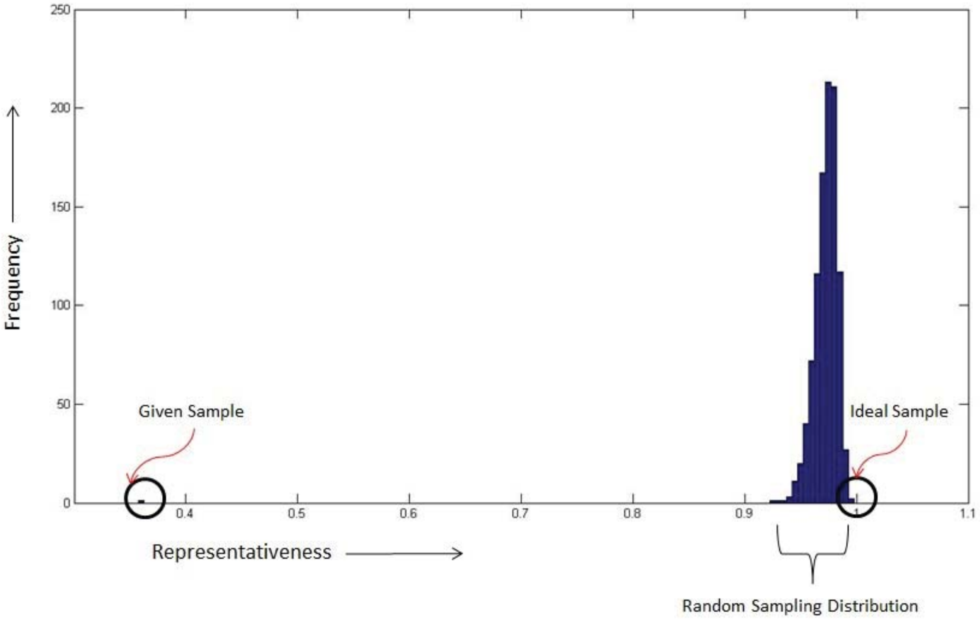


FIG. 5.11. *Histogram Of Representativeness for Random Sampling and Where Other Methods Lie*

Figure 5.11 gives an idea of the distribution of representativeness values for random sampling and shows where the given sample and ideal samples are placed in the distribution. Table 5.2 shows that ideal sampling has the highest representativeness.

Method	**Representativeness**
Given Sample	0.362
Ideal Sample	0.995
Avg. Random Sampling	0.972

Table 5.2. *Representativeness Of Samples for Unfiltered Data*

The variables selected affect the nature of the distribution. A uniform distribution of points across variables will reduce the amount variation covered by a single principal component. For example, if the variables distributed the data points in the form of a circle (in a 2 dimensional graph), then there would not be a one principal component that would cover a large portion of the variation limiting the use of PCA. The variables we have used in our experiments have a large variation (e.g. temperature) where as potential vegetation is a categorical value limiting variation.

5.2 Ideal vs Random Sampling

In the previous section, the results show that ideal sampling has a representativeness close to 1.0, the same as random sampling. This result, as described previously, is a function of the number of sites (or centroids) that need to be generated. Since the number of sites (157) is large, the representativeness is close to *1.0*. We can analyze the effectiveness of ideal sampling by measuring the representativeness of ideal samples against random sampling for a reduced of sample size. We consider the unfiltered data points as in previous example. The following graph shows the trend for representativeness for the two methods, starting from the selection of 1 sample site to a total of 157.

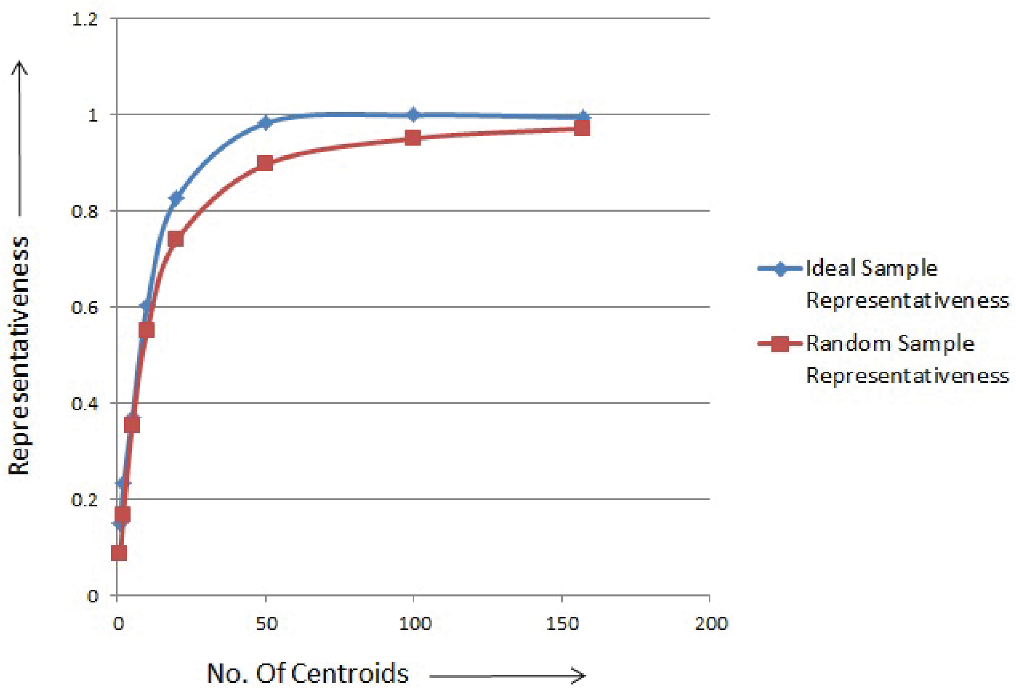


FIG. 5.12. *Representativeness Trend - Increasing Number Of Centroids*

The trend shows that ideal sample representativeness tends to 1 to with fewer centroids as compared to random sampling, making it better to select sites as compared to random sampling. Table 5.3 shows the number of centroids required by each method to reach a value for representativeness close to 1.

Method	**No. Of Centroids**	**Representativeness**
Ideal Sampling	60	0.99
Avg. Random Sampling	100 - 130	0.95 - 0.967

Table 5.3. *Number Of Centroids & Representativeness*

5.3 Measuring the Effect of Histogram Size

The ideal samples are generated by constructing a histogram of the first principal component. The algorithm 1 is affected by 2 parameters, i.e., the number of bins created in the histogram and the window size. The window size is the number bins in which any data point can be considered to represent the rest of the data points with the bins in the window completely. Currently, the window size and number of bins are chosen after testing. Figure 5.13 shows the change in representativeness for various window sizes and varying number of bins.

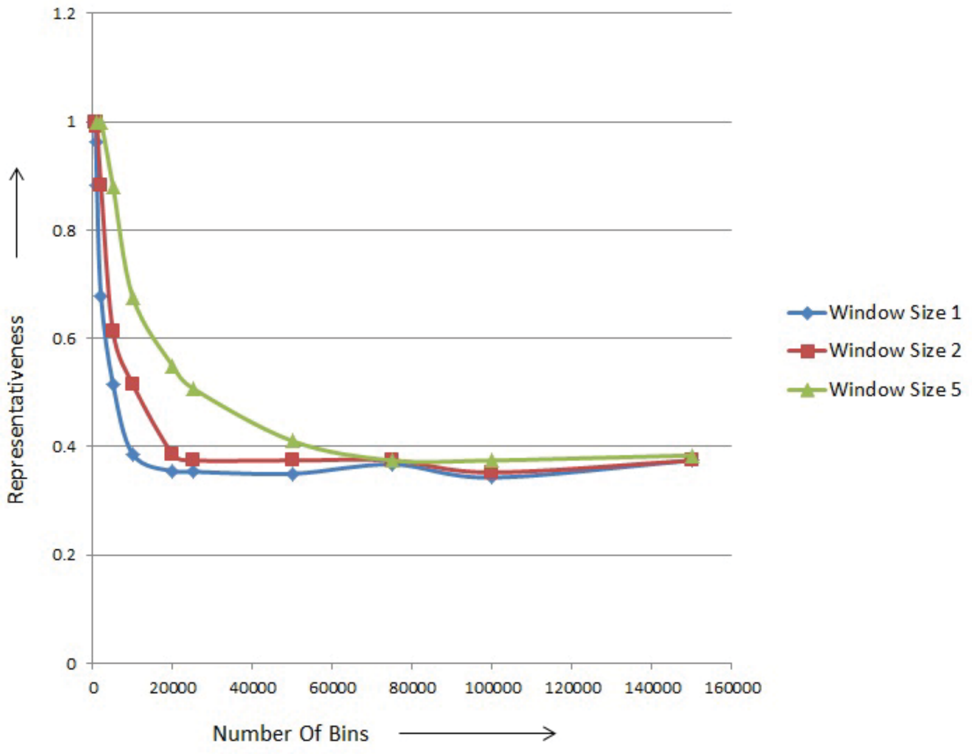


FIG. 5.13. *Representativeness Trend - Increasing Number Of Bins*

The test has been conducted for window sizes 1, 2 & 5. The minimum number of bins taken is equal to the number of centroids required which is maintained at 157 throughout all the tests. As the window size is 1, there are a total of 157 bins from which a data point can be selected (refer to the algorithm 1). As the number of bins is increased, the representativeness decreases, until it reaches a set of bins after which the representativeness becomes stable. The second observation is that the rate of decrease in representativeness is lower when the window size is increased. Thus a higher window size can reduce rate of decrease in representativeness. We explain the reasons for the decrease in the next section.

5.3.1 Effect of the Number of Bins

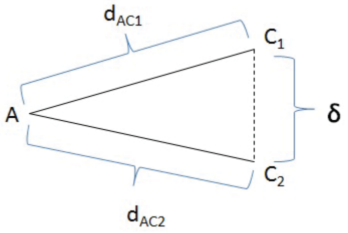


FIG. 5.14. *Example for Bin Distance*

Consider figure 5.14. There are 3 points $A, C_1,$ *and* C_2, where $C_1 \& C_2$ are 2 centroids. $d_{AC1} \& d_{AC2}$ are the distances between the points $A \& C_1$ and $A \& C_2$ in the first principal component respectively. δ is the distance between $C_1 \& C_2$. From the diagram, we see if $\delta \rightarrow 0$, then $(d_{AC1} - d_{AC2}) \rightarrow 0$.

$$\textit{IntervalSize} = \frac{p_{max} - p_{min}}{NoOfBins} \tag{5.1}$$

where p_{max} and p_{min} are the maxmimum and minimum values in the first principal component respectively.

Thus, when the number of bins increases, IntervalSize decreases. Consider the case when window size is 1. Consider figure 5.5 where there are certain parts of the histogram that have high density. As the IntervalSize decreases, centroids are chosen from bins adjacent to the mode where $\delta \rightarrow 0$. The final representativeness is based on the color scale that is used. All the data points that fall in the first bin of the scale are considered completely represented by the sample set. As more centroids are selected, the difference in the distance either remains the same or get smaller. The redundant centroids generated are unable to *cover* the rest of the points in the distribution to minimize the distance and maximize coverage. Thus,

$$\mathit{NoOfBins}\ \alpha\ \mathit{Colorscale} \tag{5.2}$$

Representativeness stabilizes after a certain number of bins because the histogram is divided into small parts such that coverage is only for the high density part of the histogram from where all the centroids are selected. In such a case, increasing the number of bins no longer affects the representativeness.

5.3.2 Effect of the Window Size

The window size creates a minimum distance between any 2 centroids that are selected as only a single point can be selected from within a bin in a certain window at a time. Hence figure 5.13 shows a lower rate of decrease in representativeness as the number of bins increase. For a optimal representativeness, the ideal sample set selected should be function of the number of bins and size of the window used. Hence equation 5.2 can be modified to,

$$\frac{NoOfBins}{WindowSize}\,\alpha\,Colorscale \tag{5.3}$$

Chapter 6

CONCLUSION & FUTURE WORK

We have provided a definition and an algorithm for calculating the representativeness of a set of sample sites. When the number of dimensions increases, clustering methods become computationnally inefficient. This is specifically when the representativeness needs to be calculated in near realtime. Hence dimension reduction techniques are used. We have used principal component analysis to perform n-dimension reduction into a single dimension based on the variance of attributes. This helps us project points onto the first principal component with maximum difference or spacing between points. To nullify the effect of the magnitude of the values in each attribute, normalization of each column is performed. This makes all the values fall in the scale from 0 to 1. The distances from the given sample set to other points are calculated using their first principal component projected values. We see that the results of the heat map generated to show representativeness across the globe is as expected. To maximize representativeness, we have provided a method that is based on the creating a histogram of the PCA values and selecting modes. The method is able to maximize representativeness as seen in the experiments conducted. The samples drawn from land change literature and the ideal samples are compared against random sampling. We show that if a filter is applied, the given sample set has the lowest representativeness as compared to the other two methods (figure 5.7 and figure 5.11), even though representativeness is greater than 0.9. It also shows that ideal sampling is better than random sampling. We are also able to see, as shown in figure 5.12, that ideal sampling reaches the same measure of representativeness as compared to random sampling with fewer samples. Hence it is better than random sampling at performing site selection. We also analyze the properties of the ideal sampling method for each of the parameters affecting the method, mainly the number of bins in the histogram and the window size. We show (figure 5.13) that when the number of bins is increased the representativeness decreases until it reaches a stable level.

In the future, a set of improvements can be performed. These are:

1. Creating a function correlation between the number of bins in the histogram to the scale applied for representativeness. The current method applies an arbitrary number of bins in the histogram. This helps to maximize representativeness under all conditions.

2. The scale applied is a linear scale from 0 to 1 that is divided equally. There is no correlation between the scale and the actual distance calculated between the centroids and the other points. If the initial projected values are very small, then the distances calculated are also small. Thus a region that is not related to any of the centroids can be shown as being represented. One solution to this problem is to normalize the projected values again, so that in case the values are very small they are scaled up accordingly to a value between 0 and 1. The points that have a value greater than 1 can be considered as outliers.

3. The problem of the distance (between a sample and another region in PCA space) being very small may still exist. The representativeness scale can be changed to account for regions in the principal component where the density of points is concentrated. Thus, density estimation and PCA outlier detection can be performed to create a tighter lower and upper bound of projected data points.

4. The visualization of the map can be improved by implementing isolines or contour lines. Contour lines or isolines are lines across which the function has the same output value. Thus isolines can be implemented for all the areas that have the same projected value on the first principal component.

REFERENCES

[1] E. Ellis, “Cdi-type ii: Globe: Evolving new global workflows for land change science.”

[2] I. P. on Climate Change, “Summary for policymakers - land use, land-use change, and forestry,” tech. rep., Intergovernmental Panel on Climate Change, 2000.

[3] N. van Vliet, O. Mertz, A. Heinimann, T. Langanke, U. Pascual, B. Schmook, C. Adams, D. Schmidt-Vogt, P. Messerli, S. Leisz, J.-C. Castella, L. Jrgensen, T. Birch-Thomsen, C. Hett, T. Bech-Bruun, A. Ickowitz, K. C. Vu, K. Yasuyuki, J. Fox, C. Padoch, W. Dressler, and A. D. Ziegler, “Trends, drivers and impacts of changes in swidden cultivation in tropical forest-agriculture frontiers: A global assessment,” *Global Environmental Change*, vol. 22, no. 2, pp. 418 – 429, 2012. Adding Insult to Injury: Climate Change, Social Stratification, and the Inequities of Intervention.

[4] A. K. Jain, M. N. Murty, and P. J. Flynn, “Data clustering: a review,” *ACM Comput. Surv.*, vol. 31, pp. 264–323, Sept. 1999.

[5] A. Vattani, “k-means requires exponentially many iterations even in the plane,” in *Proceedings of the 25th annual symposium on Computational geometry*, SCG ’09, (New York, NY, USA), pp. 324–332, ACM, 2009.

[6] S. Bubeck, U. V. Luxburg, and C. Elkan, “Nearest neighbor clustering: A baseline method for consistent clustering with arbitrary objective functions.”

[7] C. D. Manning, P. Raghavan, and H. Schtze, *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press, 2008.

[8] M. A. Carreira-Perpinan, “A review of dimension reduction techniques,” tech. rep., Dept. of Computer Science, University of Sheffield, January 27 1997. Technical Report CS-96-09.

[9] B. C. Pijanowski, D. G. Brown, B. A. Shellito, and G. A. Manik, “Using neural networks and gis to forecast land use changes: a land transformation model,” *Computers, Environment and Urban Systems*, vol. 26, no. 6, pp. 553 – 575, 2002.

[10] L. T. Tran, C. G. Knight, R. V. ONeill, E. R. Smith, and M. OConnell, “Self-organizing maps for integrated environmental assessment of the mid-atlantic region,” *Environmental Management*, vol. 31, pp. 822–835, 2003. 10.1007/s00267-003-2917-6.

[11] D. Brugger, M. Bogdan, and W. Rosenstiel, “Automatic cluster detection in kohonen’s som,” *Neural Networks, IEEE Transactions on*, vol. 19, pp. 442 –459, march 2008.

[12] R. T. M. S. M. D. J. E. Forrest M. Hoffman, William W. Hargrove and R. J. Oglesby, “Multivariate spatio-temporal clustering (mstc) as a data mining tool for environmental applications.,” *In Miquel S‘anchez-Marr‘e, Javier Bejar, Joaquim Comas, Andrea E. Rizzoli, and Giorgio Guariso, editors, Proceedings of the iEMSs Fourth Biennial Meeting: International Congress on Environmental Modelling and Software (iEMSs 2008)*, Barcelona, Catalonia, Spain, July 2008.

[13] http://glassfish.java.net/.

[14] https://github.com/mbostock/d3/wiki.

[15] A. J. K. Kevin Sahr, Denis White, “Geodesic discrete global grid systems,” *Cartography and Geographic Information Science*, vol. 30, 2003.

[16] H. Samet, “Applications of spatial data structures,” Addison-Wesley, 1990.

[17] H. Samet and R. E. Webber, “Storing a collection of polygons using quadtrees,” *ACM Trans. Graph.*, vol. 4, pp. 182–222, July 1985.

[18] L. I. Smith, “A tutorial on principal components analysis [eb/ol],” 2003 [Online].

[19] K. Kuttler, *An introduction to linear algebra*. Online e-book in PDF format, Brigham Young University, 2007.

[20] H. Abdi and L. J. Williams, “Principal component analysis,” *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[21] M. A. Carreira-Perpinan, “A review of dimension reduction techniques.,” Tech. Rep. CS-96-09, Department of Computer Science, University of Sheffield, 1997.

[22] C. Ding and X. He, “K-means clustering via principal component analysis,” in *Proceedings of the twenty-first international conference on Machine learning*, ICML ’04, (New York, NY, USA), pp. 29–, ACM, 2004.

[23] J. F. Ramankutty, N., “Estimating historical changes in global land cover: croplands from 1700 to 1992,” *Global Biogeochemical Cycles*, vol. 13(4), pp. 997–1027, 1999.