Classification of Digital Images using Textural Features

A Major Project Report submitted in partial fulfilment of

the requirement for the award of the Degree of

MASTER OF ENGINEERING

in

COMPUTER TECHNOLOGY AND APPLICATION

by

GEETA GUPTA

M.E. Computer Technology and Application

Delhi College of Engineering

Under the guidance of
Dr. S. K. Saxena

Delhi College of Engineering

July 2005

DEPARTMENT OF COMPUTER ENGINEERING

DELHI COLLEGE OF ENGINEERING

DELHI UNIVERSITY, DELHI-110042

CERTIFICATE

This is to certify that the work which is being presented in this major project entitled "**Classification of Digital Images using Textural Features**" submitted by **Geeta Gupta**, Roll No. 3016 in partial fulfilment of the requirement for the award of degree of **Master of Engineering in Computer Technology and Application**, is an authentic record of her own work carried out under the supervision and guidance of **Dr. S.K. Saxena**, Department of Computer Engineering, Delhi College of Engineering.

The work embodied in this project, has not been submitted for award of any other degree by the student to the best of our knowledge.

Dr. D. Roy Choudhury
Head of Department
Deptt. of Computer Engineering
Delhi College of Engineering
Delhi - 42

Dr. S. K. Saxena
Project Guide
Deptt. of Computer Engineering
Delhi College of Engineering
Delhi - 42

ACKNOWLEDGEMENTS

I would like to express indebted gratitude to my supervisor Dr. S.K. Saxena, Department of Computer Engineering, Delhi College of Engineering, for his meticulous guidance and suggestions. His motivation and encouragement has always been a driving force for successful completion of this project work.

I acknowledge my gratitude to my organisation, Defence Terrain Research Laboratory (DTRL), Defence Research and Development Organisation, Delhi for providing me an opportunity to pursue ME in CTA at DCE, Delhi. I profoundly thank Shri S.S. Prasad, Director (DTRL) for the kind help extended to me in all possible ways. My thanks are also due to Shri V.K. Panchal for his continued support extended to me.

I would like to express my sincere thanks to Dr. D. Roy Choudhury, Professor and Head, Department of Computer Engineering, DCE for his invaluable help, needful suggestions and constant encouragement to carry out my work comfortably in all respect.

I am thankful to all my friends at DCE for their unbending help to complete this assignment. I would like to thank Dr. Sujata Dash for her invaluable cooperation in all respect. I also thank my brother Akhilesh for being very supportive. Finally, I am grateful to my family members for being very cooperative throughout my work.

(Geeta Gupta)
July, 2005

ABSTRACT

Digital images are classified in order to identify objects of interest and extract useful information. Satellite and aerial images in digital form are a rich source of information about the earth's surface. By applying image classification techniques on these images, we can identify and categorise various natural and man-mage objects in the form of ground cover maps. Ground cover maps provide information about urban areas, agricultural fields, forest areas, water bodies and others. Availability of such ground cover maps are central to many resource management, planning and monitoring programmes.

In this project work, a supervised method of image classification based on artificial neural network approach has been implemented. This method makes use of *texture* information to identify distinct ground cover objects. Textural features are extracted from grey level co-occurrence matrices. These features are used to train a three-layer neural network in supervised manner with *backpropagation algorithm*. The trained network is used for classifying images to derive the corresponding ground cover map.

The performance of the developed classification technique is evaluated on satellite/ aerial images and on Brodatz texture images. Here, the results obtained so far have shown the efficacy of the developed approach in generating classification maps using texture information with backpropagation algorithm.

The complete software is developed in C language on Windows platform.

CONTENTS

Page No

Chapter

Chapter 1
Introduction

Digital images are classified in order to identify objects of interest and extract useful information. Here, the focus is on satellite and aerial images, which are rich source of information about the earth's surface. By applying image classification techniques on these images, we can identify and categorise various natural and man-made objects in the form of ground cover maps. Ground cover maps provide terrain related information about urban areas, agricultural fields, forests, water bodies (sea, river, lake) and others. Availability of such ground cover maps are central to many resource management, planning and monitoring programmes. These maps help in management of resources like water, forest and agricultural land. The maps derived from flood and earthquake affected areas are useful in damage assessment and planning relief activities. Image classification technique is also applicable in identification of targets like airstrips, roads and bridges for military applications.

Digital image classification is the process of assigning a pixel of satellite image to a ground cover object. Normally, a digital image is classified using image tone or spectral information of pixels. Image pixels can also be classified on the basis of their spatial relationship with a group of pixels surrounding them. Texture represents one such spatial relationship of image pixels. Here, the aim is to use texture information for identifying objects of interest in a digital image. The implemented technique uses textural features of distinct ground cover objects to perform image classification. Textures of natural and man-made objects are often visible in satellite images and can form the basis for identifying these objects. There are two broad techniques of classifying an image - supervised and unsupervised. In supervised method, data of known ground cover objects is provided to train classification algorithm. After training process, the algorithm performs the task of image classification. Unsupervised methods do not utilise training data as the basis for classification. Rather, these classifiers aggregate image pixels into a number of clusters based on defined similarity criterion. The ground cover category of the clusters is identified with the help of reference image or map data.

In this project work, a supervised method of image classification based on *artificial neural network* approach has been implemented. Artificial neural networks have been employed for many years in different application areas such as pattern recognition and speech recognition. Since, classifiers based on neural network approach make no prior assumptions about the statistical properties of data, hence are a better choice for processing satellite image data whose underlying distribution is quite often not well defined. In addition, such classifiers have the ability to learn and generalise i.e. they learn with the help of training data and produce reasonable outputs for inputs that were not encountered during training process. Here, a set of defined textural parameters of known ground cover objects is provided in the form of training data. These parameters are used to train a three-layer neural network in supervised manner with backpropagation algorithm. The well trained network is used to perform image classification. The resulting classified map is useful for generating estimates on relative presence of water bodies, urban land, forest areas and other ground cover objects.

Objectives of Dissertation

- To study digital image classification techniques.
- To study textural features and neural network approach.
- To implement backpropagation algorithm for classifying images based on texture information.

Organisation of Dissertation

This dissertation is organised into the following 6 chapters:

- Chapter 1 presents introduction about the developed classification method.
- Chapter 2 provides definition of texture and textural features.
- Chapter 3 discusses about image classification techniques.
- Chapter 4 describes neural network approach and its implementation.
- Chapter 5 deals with the results and discussion.
- Chapter 6 includes conclusion of this dissertation.

Chapter 2
Textural Features

A digital image is usually characterised by two aspects: *Tone* and *Texture*. Image tone consists of gray level variations of the pixels throughout the entire image, while image texture is characterised using both the grey value for a given pixel and the grey level pattern in the neighbourhood surrounding the pixel over relatively small areas. Texture is a spatial relationship of grey values which is a useful feature for identifying objects or regions of interest in an image. Despite its importance, there is no unique definition of texture. Several authors have attempted qualitatively to define texture [1]. Some of the definitions given by them are:

- Haralick [1979]
  "The image texture we consider is non-figurative and cellular. An image texture is described by the nature and type of its (tonal) primitives and the spatial organisation or layout of its (tonal) primitives…"

- Faugeras and Pratt [1980]
  "The basic pattern and repetition frequency of a texture sample could be perceptually invisible, although quantitatively present... In the deterministic formulation, texture is considered as a basic local pattern that is periodically or quasi-periodically repeated over some area."

- Jain and Karu [1996]
  "Texture is characterised not only by the grey value at a given pixel, but also by the grey value 'pattern' in a neighbourhood surrounding the pixel."

Texture perception is an important part of human vision. Objects may often be distinguished by their characteristic textures in spite of similar colours or shapes. The visual effect of texture is produced by spatial distribution of tonal or gray level variations over relatively small areas. Texture of a surface may be perceived as being directional or non-directional, smooth or rough, coarse or fine, regular or irregular, etc. The following images in fig.1 illustrate such characteristics of texture.

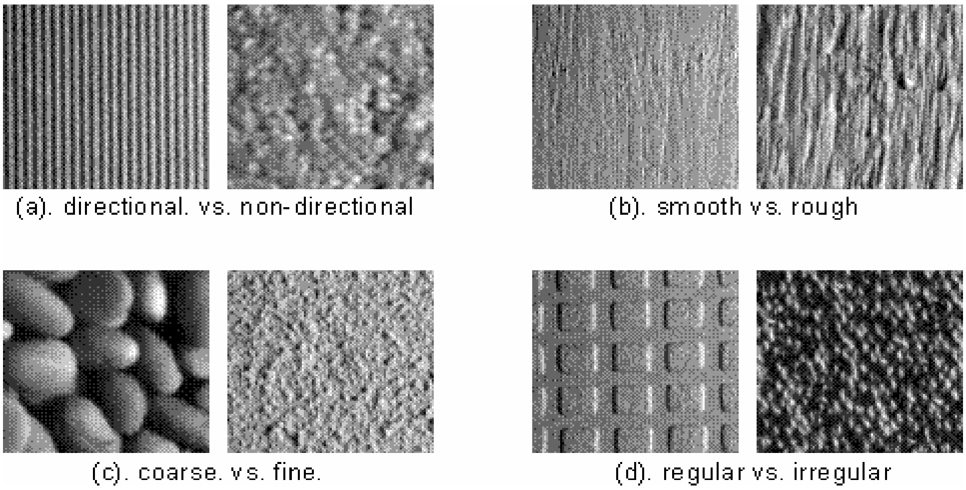


Figure 1. Different perception of textures

Similarly, satellite and aerial images of ground cover are viewed as textures of natural and man-made objects. Examples of some of the textures of terrain (elevation, vegetation) are shown in fig. 2(a). Man-made objects (buildings, roads) as visible in satellite images are displayed in fig. 2(b).

(a) (b)

Figure 2. Satellite images showing textures of natural and man-made objects

Natural objects such as agricultural fields, forests, water bodies etc. have their own distinct textures. Similarly, man-made objects such as roads, street patterns, buildings can be identified in images with the help of their textural properties. Characteristics of textures of distinct natural and man-made objects can be used to classify satellite and aerial images. The characteristics of texture such as homogeneity, contrast, coarseness, smoothness and others are defined with the help of **textural features** [2]. Textural features describe textures in terms of measurable parameters. The approaches to describe texture are grouped into structural and *statistical* methods. Structural methods define textures as repetition of basic primitive patterns with certain placement rules. Here, textures with more regular structures are analysed. The extracted primitives and their placement rules are used to identify distinct textures. In general, it is difficult to find such primitives in textures of natural objects. For such textures statistical methods are used. Statistical methods as the name implies define textures by a set of statistical parameters.

One of the most widely used statistical methods for describing texture was introduced by RM Haralick. This method is based on second order statistics and is called GLCM approach. In this approach the existing spatial relationship is characterised by a set of grey level co-occurrence matrix (GLCM). GLCM records the relative frequencies of grey values of any two pixels, at a specified distance and orientation. RM Haralick proposed a list of 13 measurable parameters or textural features that can be extracted from co-occurrence matrices to describe texture [3]. In the present work, four such texture measures namely *contrast, energy, entropy and inverse difference moment* have been implemented to carry out image classification [2],[4].

Brief explanation and formulae to compute each of these four measures for a given co-occurrence matrix P having L levels of grey values are as follows:

i) Contrast - It is defined as measure of amount of local variations present in an image. High values of contrast for small pixel distance indicate a fine texture. Contrast is calculated using the following equation:

$$Contrast = \sum_{i=1}^{L} \sum_{j=1}^{L} (i - j)^2 * P_{ij} \tag{2.1}$$

ii) Energy - Energy measures texture uniformity i.e. pixel pair repetitions. For uniform textures energy is high. The mathematical expression for energy is :

$$Energy = \sum_{i=1}^{L} \sum_{j=1}^{L} (P_{ij})^2 \tag{2.2}$$

iii) Entropy - Entropy is measure of the disorder of an image. When the image is not texturally uniform, energy is large. Formula for entropy is:

$$Entropy = -\sum_{i=1}^{L} \sum_{j=1}^{L} P_{ij} * \log(P_{ij}) \tag{2.3}$$

iv) Inverse Difference Moment (IDM) - This parameter measures local homogeneity. Its value is high for images having slow changes in grey values. IDM is expressed as :

$$IDM = \sum_{i=1}^{L} \sum_{j=1}^{L} \frac{1}{1 + (i - j)^2} * P_{ij} \tag{2.4}$$

Each of these four measures is computed from four directional (0º, 45º, 90º and 135º) co-occurrence matrices. These computations result in a total of 16 textural features. These 16 textural features form the basis of texture based image classification.

The textures of natural objects as visible in aerial or satellite images do not have well defined texture primitives. Hence, statistical methods are normally adopted to define their characteristics.

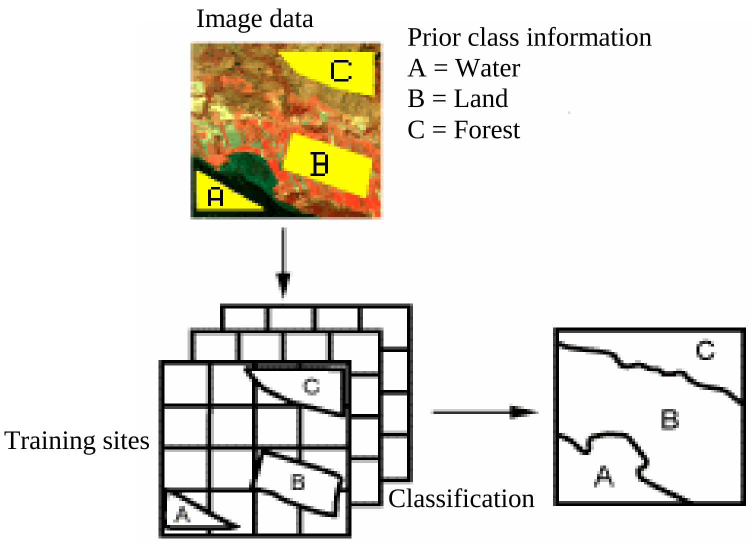
Chapter 3
Image Classification

Digital images are classified in order to identify objects of interest and extract useful information. In the field of remote sensing, image classification technique is used to identify ground cover objects and extract information in the form of vegetation, urban area, water bodies (sea, river, lake) etc. from satellite or aerial images. In order to obtain such information, each pixel in a given image need to be labelled with its correct ground cover class (i.e. vegetation, urban area and others). This process of labelling image pixels with their correct classes is referred to as *image classification.* The resulting classified image is comprised of a mosaic of pixels, each of which belongs to a ground cover class. Normally, image data are classified using image tone or spectral information. Image pixels can also be classified on the basis of their spatial relationship with pixels surrounding them. Such relationships include image texture, shape, feature size etc. Image classification methods based on spatial relationships tend to be more complex and computationally intensive than spectral methods [27].

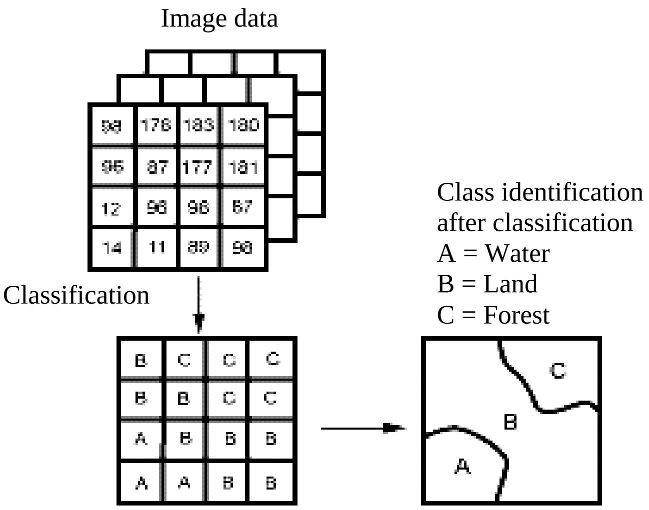
Methods of image classification are often grouped into following two categories :

- Supervised approach
- Unsupervised approach

In supervised approach, labelling of image pixels is *supervised* by specifying the identity of ground cover classes like water bodies, forests, agricultural fields, urban land etc. In order to do this, representative sample sites of the known classes are selected in the image. These areas are commonly referred to as *training sites* because the spectral / textural characteristics of these known areas are used to train the classification algorithm (fig. 3(a)). Statistical parameters are calculated for each training site in order to determine their “signatures” i.e. spectral or textural characteristics. Ground cover mapping of the image is performed by comparing each pixel’s characteristics with the signatures of known classes and labelling with the class it most closely “resembles”. A few pixels remain unclassified because these may belong to a class not recognised or defined in the imagery.



3 (a) Supervised classification



3 (b) Unsupervised classification

Figure 3. Supervised and Unsupervised classification methods

Unsupervised classification approaches do not utilise training data as the basis for classification. Rather, these classifiers aggregate image pixels into a defined number of classes based on natural groupings or clusters present in the image. The basic principle of grouping is such, that the pixels within a cluster have similar spectral / textural attributes. Since, the classes extracted are based solely on natural groupings, their identity with respect to ground cover is not known. The ground cover identity of each class is determined with the help of some reference data available in the form of maps or images. In fig. 3(b), this method of image classification based on spectral information is illustrated.

The fundamental difference between these two approaches is that supervised classification involves a training step followed by a classification step. Whereas in unsupervised approach, the image data is first classified by aggregating them into clusters present in the image. Then the ground cover identity of these clusters is determined by comparing the classified image with ground reference data.

The classification approaches can be further divided into parametric and non-parametric methods. Parametric classifiers usually require assumptions about the underlying statistics of the data. For instance, maximum-likelihood classifier is a supervised parametric algorithm which requires that the training data have a Gaussian statistical distribution which is not always true with satellite data. Non-parametric classifiers work with arbitrary distributions of data. They do not depend much on statistical behaviour of input data. Non-parametric classifiers which are based on neural network approach, have received considerable interest because they have the ability to learn and generalise. Due to these properties, neural network classifiers are often preferred over conventional classifiers.

Here, the aim is to implement neural network classifier based on texture information. Image texture has been extensively used to classify satellite images. Texture is quite useful when only single band image is available for classification.

Several authors in the field of remote sensing have studied the performance of textural features and neural network classifier for identification of ground cover in satellite images. Some of them have compared the performance of neural network classifier with conventional classification methods [5] – [15]. A detail discussion on neural network approach and its implementation is covered in chapter 4.

Supervised classification methods tend to be labour-intensive in order to obtain a sufficiently comprehensive set of training sites. Still these methods are preferred in the analysis of remotely sensed data because in most of the applications the types of ground cover classes to be extracted are known in advance [16]. In addition, because of this kind of prior information, the accuracy of supervised methods is better than unsupervised classification algorithms.

Chapter 4

Neural Network Approach and Implementation

Neural network has been applied to a number of image classification problems and has shown considerable success with respect to the performance of conventional algorithms. This success stems from the ability of neural network to overcome several limitations of conventional algorithms. The most prominent limitation of conventional classifiers is the presumption of statistical behaviour of input data. Neural network makes no prior assumptions about the behaviour of input data, instead determines a metric specific to the problem during training procedure. The increasing popularity of neural network is due to its ability to learn and generalise i.e. a neural network learns with the help of training data, and produces reasonable outputs for inputs it did not encounter during training process. The simplicity and generality of neural network approach often leads to classifiers, which are more efficient than conventional methods. Neural networks are a better choice for classifying aerial and satellite image data whose underlying distribution is quite often not well defined [17]-[19]. In addition, once trained, neural network can perform image classification relatively rapidly, although training process can be time consuming.

The following sub-sections describe briefly about the implemented neural network architecture, representation of training data, network training protocol and training algorithm respectively.

4.1 Network Architecture

A three layer neural network has been implemented to perform image classification using backpropagation algorithm. A neural network has the following characteristics :

- Network structure - A neural network is composed of a number of computational elements called neurons or nodes. These nodes are organised into layers. These layers are addressed as – input layer, hidden layer and output layer (fig. 4). The input layer serves as distribution structure for the data being presented to the network. This layer consists of nodes equal to the number of texture features used for classification. The output layer is composed of as many neurons as the number of distinct texture classes to be identified. The processed information by the network is retrieved from the output layer. The layer present between the output and the input layers is known as hidden layer. The number of nodes in hidden layer affects the performance of neural network. If the patterns are well separated or linearly separable then few hidden nodes are needed. Conversely, if the patterns are from complicated densities that are highly interspersed, then more hidden nodes are needed. Thus, number of hidden nodes to be used is problem dependent [20]. A schematic diagram of a three layer neural network is presented in fig. 4.

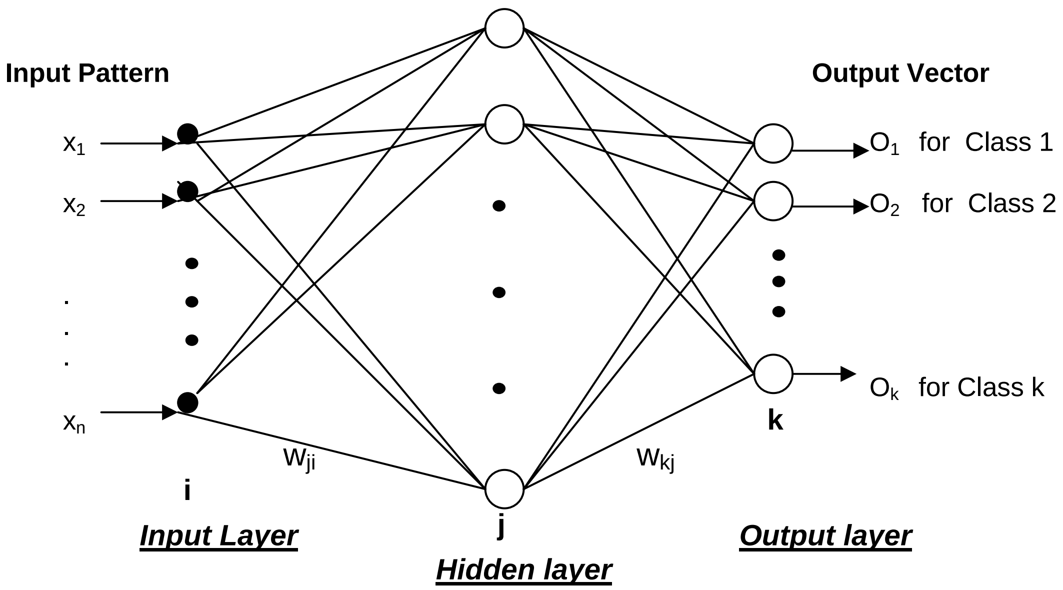


*Figure 4. Three layer neural network with layers labelled as **i, j** and **k.***

- Connection weights - Every node of input and output layer is fully connected to all of the nodes in the hidden layer. There are no connections between nodes of same layer. Each connection between any two nodes is associated with a numerical value called as connection "*weight* ". Initialisation of network learning takes place by setting all these connection weights with small random values that are modified during training process [21]-[23]. These connection weights as a whole contain the learned information about the input patterns.

- Neuron computation - In neural network, each neuron performs two functions. First, it computes weighted sum of all of its incoming inputs. This weighted sum is then passed through an activation function to produce an output value. The mathematical expressions for computing weighted sum net_j and output o_j of a given neuron j are as follows :

$$net_j = \sum_i w_{ji} o_i + w_b \quad (4.1)$$

$$o_j = \phi(net_j) = (1 + e^{-net_j})^{-1} \quad (4.2)$$

where o_i is incoming inputs to neuron j,
w_b is bias weight and
ϕ is activation function

All nodes in the network except the ones in the input layer perform the same two computations (eq. 4.1 and 4.2) to produce their output. The computed outputs of nodes in one layer are sent as inputs to many other nodes of next higher layer through the weighted connections (fig. 5).

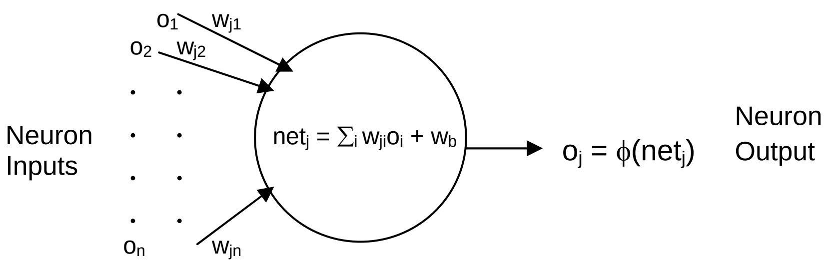


Figure 5. Computations at neuron j.

- Activation function - The model of each neuron in the network includes a non-linear activation function. The most common used form of such non-linearity is the sigmoid function ϕ defined by equation 4.3. The important point about this function is that it is smooth (i.e. differentiable) and its derivative ϕ' can be expressed in terms of the function itself (eq. 4.4).

$$\phi(net_j) = (1 + e^{-net_j})^{-1} \qquad (4.3)$$

$$\phi'(net_j) = \phi(net_j)*(1 - \phi(net_j)) \qquad (4.4)$$

where net_j is defined previously by eq. 4.1.

Sigmoid function adds non-linearity to network calculations, which is an important property allowing the network to solve problems more accurately than linear techniques [24],[25]. Another property of this function is that its output values are bounded. Using expression 4.3, it produces an output value of one when weighted sum i.e. net_j of a node j reaches infinity and a value of zero when net_j goes to minus infinity. This property of sigmoid function keeps the network training time reasonable.

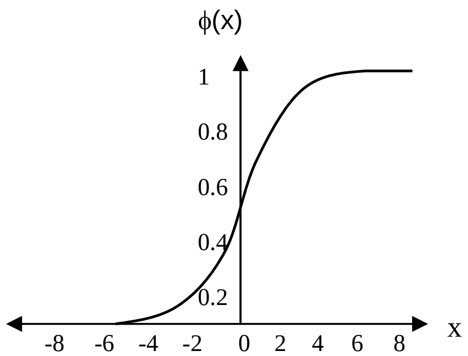


Figure 6. Sigmoid function $\phi(x) = (1 + e^{-x})^{-1}$

4.2 Training data and its representation

In supervised classification approach, identity of known classes is required in the form of training data as mentioned in chapter 3. The number of distinct texture classes to be labelled is specified in an image. Samples of these classes are collected to extract textural features using grey level co-occurrence matrices.

These textural features are recorded in vector form and are employed for training the network. Collection of such vectors of all k distinct texture classes forms the complete training set.

In neural network approach, training data is presented in the form of unique input pattern and its corresponding desired (true) output value. This form of presentation is expressed by a pair of data vectors (s^d, t^k) where s^d is the d dimensional input pattern to be learned and t^k is its corresponding k dimensional desired output vector to be produced by the network. Encoding of input data is very important in neural network approach. The components of the input vectors should be normalised to lie in some fixed range to avoid very large values of net_j (eq 4.1).

Representations at the output nodes also need to be defined. The natural way to encode the output classes is to use one output neuron per texture class. Generally, desired output values consist of low values of 0 (or –1) for outputs that do not correspond to the pixel's assigned class and a high value of 1 for the output that does correspond to the pixel's assigned class. With this concept, the results obtained in the output layer are coded as binary vectors. Such vectors determine the class of input data presented to the network.

For example in a classification problem of three categories of texture patterns say A, B and C, using this concept, the output vectors can be coded as:

$$\begin{pmatrix}1\\0\\0\end{pmatrix},\ \begin{pmatrix}0\\1\\0\end{pmatrix} \text{ and } \begin{pmatrix}0\\0\\1\end{pmatrix}$$

By using this coding scheme for a given input pattern, if the output vector is (1,0,0) then it implies that the input has been identified as class A by the network. Similarly, if output vectors are (0,1,0) or (0,0,1) then that pattern has been qualified for class B or class C respectively.

Thus, the coded output vectors help in deciding the class of a given input pattern. While using the standard sigmoid function (eq. 4.3), the target values for output vectors are usually selected as 0.1 and 0.9 with value 0.9 to represent the correct class and 0.1 to represent all other remaining classes. Based on this representation, an input pattern is assigned to class i if output node i has the highest value of all k outputs and is above a defined threshold. Similarly, a pattern is classified as unknown, if the numerical values of all output nodes are below a pre-specified threshold or the values of two or more output nodes are above the threshold and similar.

4.3 Network Training Protocol

The three most useful training protocols described in literature are - *stochastic,* batch and online [24]. In stochastic training, the network weights are adjusted after presenting each pattern of the training set. In batch training, learning takes place after all the patterns are presented to the network. In online training, each pattern is presented once and only once. In the current work, *stochastic training* protocol has been followed. In this mode of training, connection weights are updated on a pattern-by-pattern basis, which makes the search in weight space stochastic in nature. This in turn makes it less likely, the network training algorithm, to be trapped in a local minimum.

Since the training set comprises of N training samples, arranged in the order ((s_1, t_1) (s_2, t_2) (s_3, t_3)…….. (s_n, t_n)). In stochastic approach, the first training sample (s_1, t_1) is presented to the input layer, computations take place through the network and output is determined at the output layer. This network response is compared with the desired one to calculate the error, which is fed back to the system to adjust the weights. Next, the sample (s_2, t_2) is presented and further adjustments are done. The process continues until last sample (s_n, t_n) for an epoch is accounted for. One complete presentation of all samples in the training set to the network is called an epoch. The training process includes a number of such epochs until the network response is close to the desired output of the training samples. The total number of such epochs is an indication of the relative amount of learning exercised by the network.

4.4 Network Training Algorithm

Neural networks have been applied successfully to solve problems by training them in a supervised manner with a popular algorithm known as ***backpropagation*** algorithm. Backpropagation algorithm was initially proposed by PJ Werbos in 1974, and later it was independently developed by D Parker in 1986, and DE Rumelhart, et al. in 1986. Backpropagation algorithm attempts to find a solution by qualifying mismatch between the desired output of a pattern and its actual value computed by the network.

The backpropagation algorithm consists of two phases. All the training samples go through these two phase processing. In the first phase (i.e. forward pass), the input is presented and propagated through the network to produce its actual output. During this phase connection weights of the network are all fixed. While in the second phase, a backward pass through the network (analogous to the initial forward pass) is made to adjust all the weights in accordance with an error correction rule. In this rule, the actual output of the network is subtracted from the desired (target) output to produce an error signal. The error signal is then propagated backwards through the network to carry out the adjustment of connection weights. These adjustments are made in an attempt to move network outputs closer to desired outputs. The procedure of weight modification is repeatedly performed for all the training samples in epoch by epoch basis.

For each training sample, backpropagation aims to maximise the output of the neuron related to its assigned class and to minimise the activation values of all other output neurons so that the network output moves closer to the desired output after several learning steps. This is achieved by the process of adjusting the connection weights. The weight adjustment factor is defined by *delta rule*. At iteration n, the delta rule computes weight adjustment factor as the product of learning rate, local gradient and input of neuron. Learning rate η indicates the relative size of the changes in connection weights and affects the speed of training. The practice is to select small values of η between 0 and 1 on experimental basis. For a three layered neural network (fig. 4), the values of weight adjustment factors for weights between these layers are computed at iteration n, using the following two equations:

$$\Delta w_{kj}(n) = \eta * \delta_k(n) * O_j(n) \quad (4.5)$$

$$\Delta w_{ji}(n) = \eta * \delta_j(n) * O_i(n) \quad (4.6)$$

The local gradient δ at iteration n depends on whether a neuron is an output node or a hidden node. If neuron j is an output node then the local gradient is equal to the product of the corresponding error signal $e_j(n)$ and derivative $\phi'(net_j(n))$ of associated activation function. If neuron j is a hidden node then the local gradient is equal to the product of $\phi'(net_j(n))$ and weighted sum of δs computed for the neurons in the output layer that are connected to neuron j. These two distinct local gradients represented by δ_k and δ_j for neurons of output and hidden layer are calculated by the following two equations:

$$\delta_k(n) = e_k(n) * \phi'(net_k(n)) \quad (4.7)$$

$$\delta_j(n) = \sum_k \delta_k(n) * w_{kj}(n) * \phi'(net_j(n)) \quad (4.8)$$

The error function e_k for node k in output layer is expressed as:

$$e_k(n) = (t_k(n) - O_k(n)) \quad (4.9)$$

After carrying out the above calculations, connection weights w_{ji} and w_{kj} are finally updated as :

$$w_{ji}(n+1) = w_{ji}(n) + \Delta w_{ji}(n) \quad (4.10)$$

$$w_{kj}(n+1) = w_{kj}(n) + \Delta w_{kj}(n) \quad (4.11)$$

The network training performed must be stopped after a reasonable number of epochs because excessive training leads to poor generalisation. To stop the network from over-training some well defined criterion functions are adopted [24],[25]. Most often, mean square error (MSE) is used as stopping criterion for network training. Mean square error is defined as mean of summed square of e_k (eq. 4.9) for all training patterns. Using this criterion, the process of learning is stopped when MSE falls below a defined threshold. Another method, which is used to stop training, is based on the generalisation capability of the trained network. In this method, after some finite number of learning steps, the network is tested for its generalisation performance. In generalisation method, a set of test samples (different from training samples) is used to validate the performance of the network. The learning process stops when the generalisation performance is adequate enough to satisfy the requirement. Both the methods have been used as stopping criteria in this project.

Summary of the implemented *backpropagation* training algorithm is presented below in the following steps:

Step 1: Initialise network parameters and set all connection weights and biases to small random values.

Step 2 : Present input and output values of training samples.

Step 3 : Perform forward computations using eq. 4.1 and eq. 4.2.

Step 4 : Perform backward computations by calculating the local gradients using eq. 4.7 and eq. 4.8. Adjust the connection weights according to eq. 4.10 and eq. 4.11.

Step 5: Iterate the forward and backward computations under steps 3 and step 4 by presenting training samples in new epochs until stopping criterion is met.

In this section, the implementation of textural features and backpropagation algorithm is discussed. The first parameter is the number of texture measures to be employed for classification. Only four texture measures have been used, therefore

$$FN = 4$$

Each of these four measures is computed using equations (2.1) - (2.4) for four defined orientations i.e. 0º, 45º, 90º and 135º. This computation results in a total of 16 textural features. Hence

$$FLen = 16$$

The identification number and the corresponding orientation of these 16 features are illustrated in table 1.

Table 1. Details of 16 textural features

Feature No.	*Feature Name*	*Orientation*
1	Contrast	0º
2	Energy	0º
3	Entropy	0º
4	IDM	0º
5	Contrast	45º
6	Energy	45º
7	Entropy	45º
8	IDM	45º
9	Contrast	90º
10	Energy	90º
11	Entropy	90º
12	IDM	90º
13	Contrast	135º
14	Energy	135º
15	Entropy	135º
16	IDM	135º

The following three variables are used to represent the size of the three layers of the network.

I is the number textural features used for classification

H is the number of nodes in Hidden layer

K is the number texture classes to be identified in an image

These three numbers together represent the network structure as I-H-K.

The connection weights between Input and Hidden layer are recorded in matrix W1 of dimension I x H. Similarly, the connection weights between Hidden and Output layer are recorded in matrix W2 of dimension H x K. These two matrices are initialised with small random numbers in the range [-0.5 , +0.5].

$$\begin{bmatrix} W1 \end{bmatrix}_{I \times H} \qquad \begin{bmatrix} W2 \end{bmatrix}_{H \times K}$$

Network learning takes place in stochastic mode by modifying the values of connection weights in W1 and W2. Matrices dw1 and dw2 record the adjustment factors for the corresponding weights in W1 and W2 respectively.

$$\begin{bmatrix} dw1 \end{bmatrix}_{I \times H} \qquad \begin{bmatrix} dw2 \end{bmatrix}_{H \times K}$$

These two matrices W1 and W2 contain the learned information after modifying the weights in W1 and W2 for a defined number of EPOCHS.

TRNG defines total number of training samples

EPOCHS is the number of iterations to train the network

η is the network learning rate and its value lies in the range [0 – 1]

The input is presented to the network in vector form represented by **X**. The numerical values of this vector are in the range [-1 , +1]. The network response for a given input sample **X** is recorded by vector **O.** The numerical values of the elements of this vector decide the texture class of input sample **X**. A threshold value of 0.4 has been used to evaluate the class of a given sample.

$$\mathbf{X} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \cdot \\ \cdot \\ \cdot \\ x_l \end{pmatrix} \qquad \mathbf{O} = \begin{pmatrix} o_1 \\ o_2 \\ o_3 \\ \cdot \\ \cdot \\ \cdot \\ o_k \end{pmatrix}$$

Input and output vectors

In supervised network learning, each training sample is presented to the network together with its corresponding class vector **t**. Each element of this vector is associated with one texture class, therefore the values of each t_i are set to 0.1 or 0.9.

$$\mathbf{t} = \begin{pmatrix} t_1 \\ t_2 \\ t_3 \\ \cdot \\ \cdot \\ \cdot \\ t_k \end{pmatrix}$$

Desired output vector

The difference of actual output of the network and the desired output of each training sample is recorded as network error. The error vector of a given training sample is defined as **:**

$$\mathbf{e} = \mathbf{t} - \mathbf{O}$$

$$\begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ . \\ . \\ . \\ e_k \end{pmatrix} = \begin{pmatrix} t_1 \\ t_2 \\ t_3 \\ . \\ . \\ . \\ t_k \end{pmatrix} - \begin{pmatrix} o_1 \\ o_2 \\ o_3 \\ . \\ . \\ . \\ o_k \end{pmatrix}$$

Error vector

This error is fed back to the network which propagates through the network in backward direction, from output to hidden layer and from hidden to input layer in order to carry out the required weight adjustments using the *delta rule* (eq. 4.5 and 4.6*)*. The procedure of weight adjustment is repeatedly performed for all the training samples for given number of EPOCHS, so that the MSE falls below a minimum value and the generalisation capability of the network is also maintained. Mean square error of all training samples in one EPOCH is calculated as :

$$MSE = 1/TRNG \sum_i (\sum_j e_j^{\,2}) \qquad (4.12)$$

where i = 1 to TRNG and j = 1 to K.

The complete process of the implemented supervised image classification using backpropagation algorithm is shown in fig. 7.

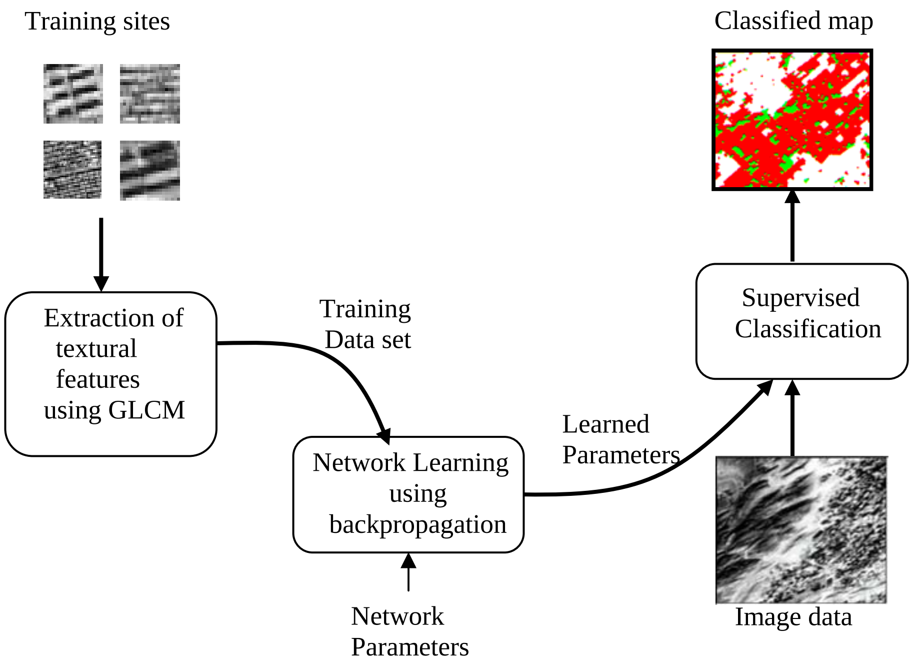


Figure 7. Supervised image classification

After the completion of supervised learning process, the trained network is used as image classifier. In order to classify a given image, the defined 16 textural parameters of table 1 are computed. The numerical values of the computed textural features of each image pixel are used as elements of 16 dimensional vector X. The network response for vector X is determined with the help of matrices W1 and W2 and is recorded by vector O. The values of this output vector O are evaluated to identify the class category of the input sample X. The output node with the highest value and above a pre-specified threshold of 0.4, determines the class category of the pixel. As explained earlier, each output node is associated with one particular texture class (fig. 4) and is assigned with a unique color code. The classified pixel is then labeled with the color code of that particular class. This process of labeling of classified pixels is carried out in order to highlight the identified ground cover objects in the image.

Chapter 5

Results and Discussion

To evaluate the performance of the developed supervised image classification method, the following two different types of data are used:

- Satellite / Aerial Images
- Brodatz Texture Images

Textures of different ground cover objects are visible in satellite and aerial images [28],[29]. Such texture information is used to classify the satellite images in terms of ground cover objects. The developed classification method was also applied on Brodatz images. Brodatz texture images are one of the popular benchmarks used for texture analysis [4],[30]. Hence, three such texture images namely, leather (D24), wood grain(D68), and raffia(D84) have been used in one of the test cases. Originally, all the images consisted of 256 grey levels. For texture processing the grey levels were reduced from 256 to 16 or 8 levels. These reduced grey level images were used for generating training sites and classification maps. Selection of proper training sites is a labour-intensive task but affects performance of the network. Textural features were extracted with defined texture windows from the training sites. The numerical values of these features were normalised in the range [-1,+1] and recorded in vector form upto three places of decimal. Each of these training samples was associated with their class tag.

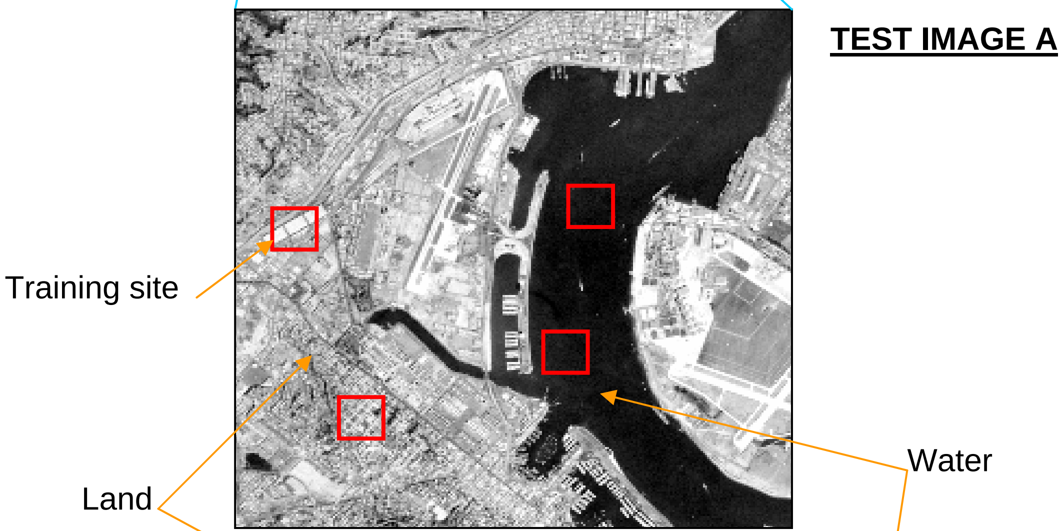
All the test images used for classification were of size 256 by 256 pixels. Network learning was performed with fixed learning rate. The well trained network is used to classify the test images. In the classified map, the pixels are labelled with color codes of their respective classes. The developed software also records the number of pixels contributing to each class. The results obtained so far have shown the efficacy of the developed approach in generating classification maps using only four texture measures with back propagation algorithm.

TEST IMAGE A

This aerial image is showing a view of central part of Sandiego in southernmost California. It shows the Bay harbour and airports. Two distinct types of textures corresponding to water and land have been identified in test image, which has been extracted from the full scene. The areas (20x20 pixels) selected as training sites for the two textures are highlighted in red box. Texture window of 9x9 pixels was used to extract textural parameters from these training sites. A total of 400 training samples (with 200 samples per class) were used for training the three layer neural network. Network learning was performed for 279 epochs with a fixed learning rate of 0.001. The classification map produced is displayed with water and land features labelled in two respective colors.

The training data used for classifying water and land are included in appendix– I A.

AERIAL IMAGE



Texture window = 9x9 pixels

Network 16-7-2

TRNG = 400

EPOCHS = 279

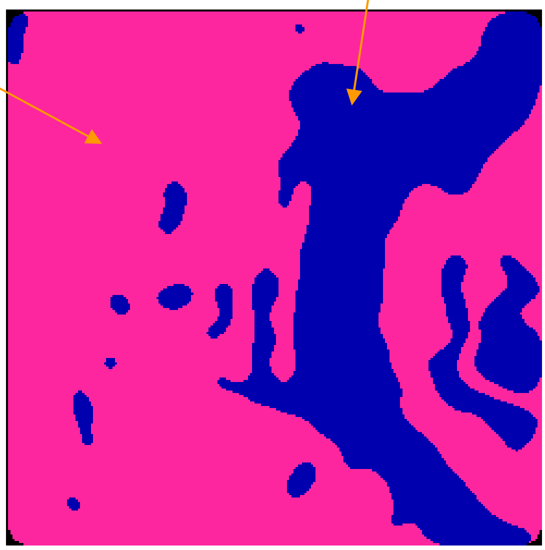
η = 0.001

Initial MSE = 0.2661

Final MSE = 0.1518

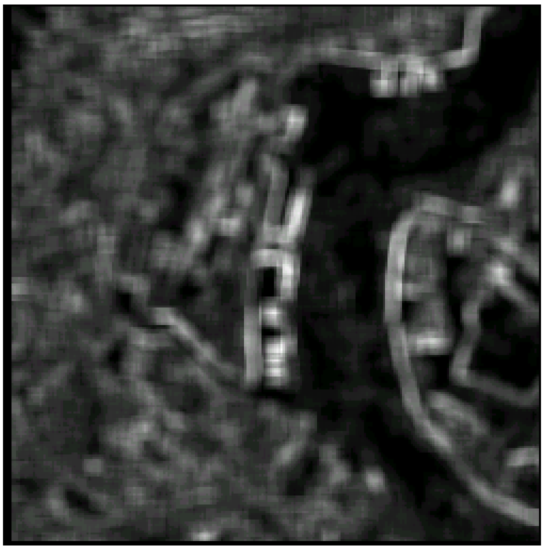
No of pixels classified as Land = 40503

No of pixels classified as Water = 20504

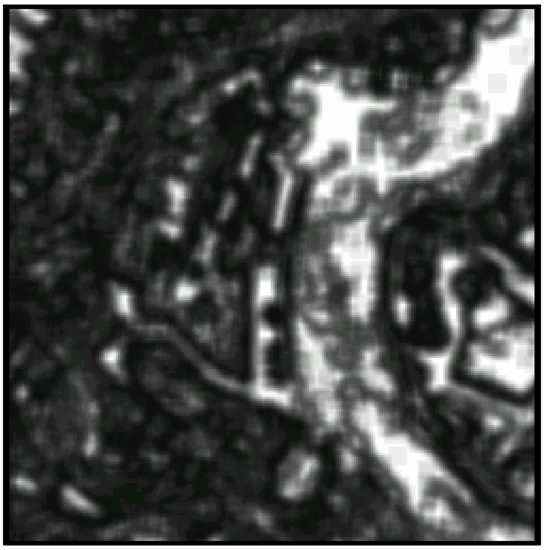


Classified Map

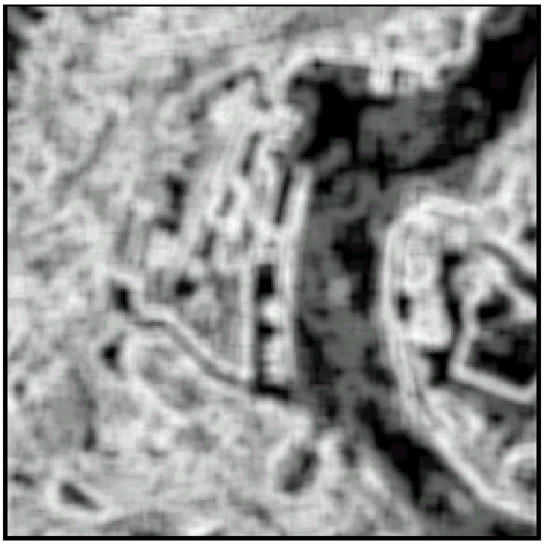
TEXTURE MAPS OF TEST IMAGE A



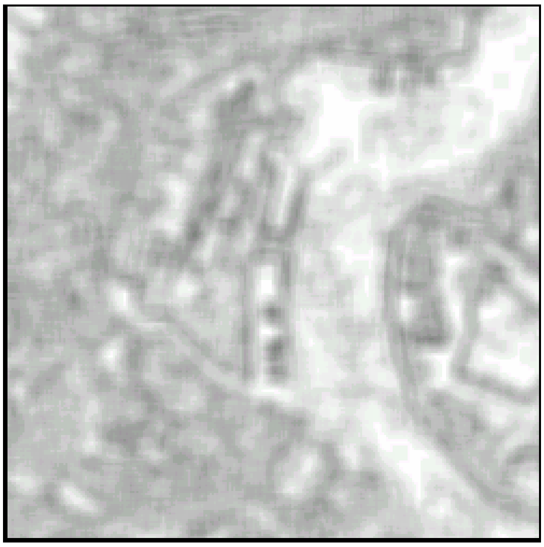
Contrast



Energy



Entropy



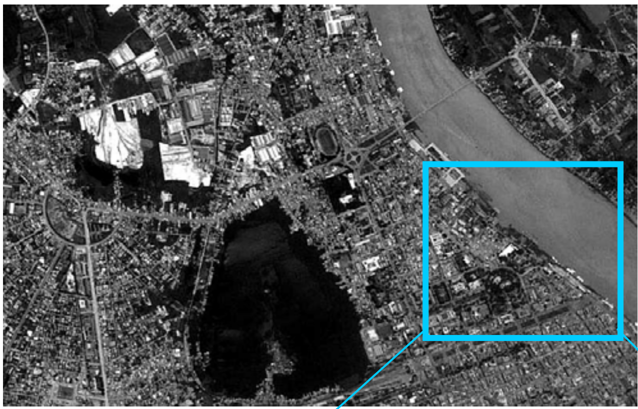
IDM

TEST IMAGE B

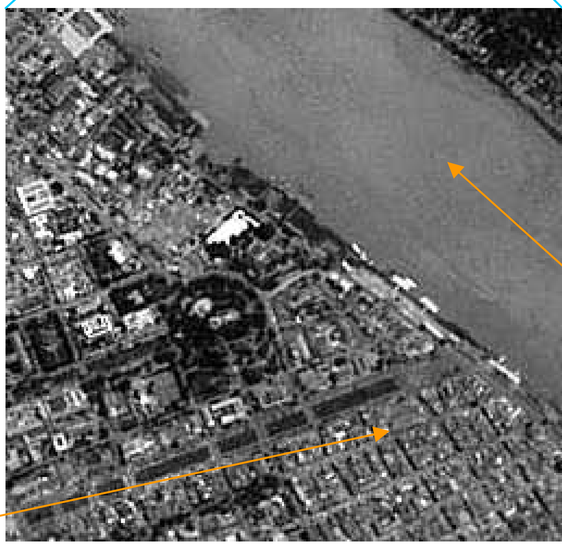
This is a Spot image of Cambodia showing features of urban areas, such as street patterns, buildings, roads and bridges. In the following test image classification was performed for two ground cover objects. Training samples of urban and non-urban area were selected from full scene. Texture window of 7x7 pixels was used for extracting the textural parameters. A total of 400 training samples were used to perform supervised learning. Network was trained for about 162 epochs with constant learning rate η = 0.0004. The trained network was applied to classify IMAGE B to obtain the displayed classified map. Black areas signify the pixels, which could not be classified for any of the two selected classes.

The training data used for classifying urban and non-urban area are included in appendix – I B.

SPOT IMAGE



TEST IMAGE B



Non-Urban area

Urban area

Texture window = 7x7 pixels

Network 16-9-2

TRNG = 400

EPOCHS = 162

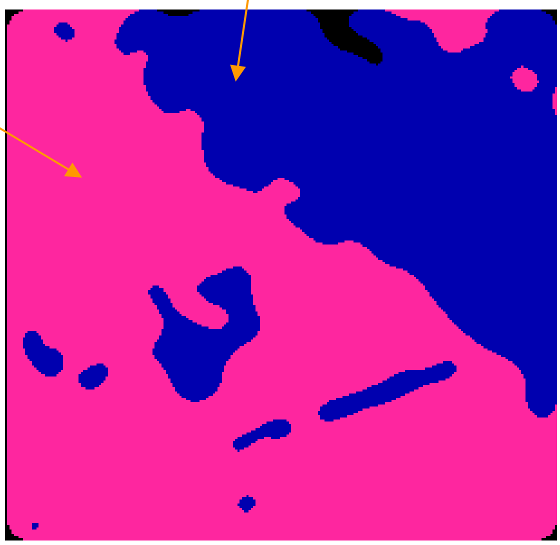
η = 0.0004

Initial MSE = 0.2610

Final MSE = 0.1381

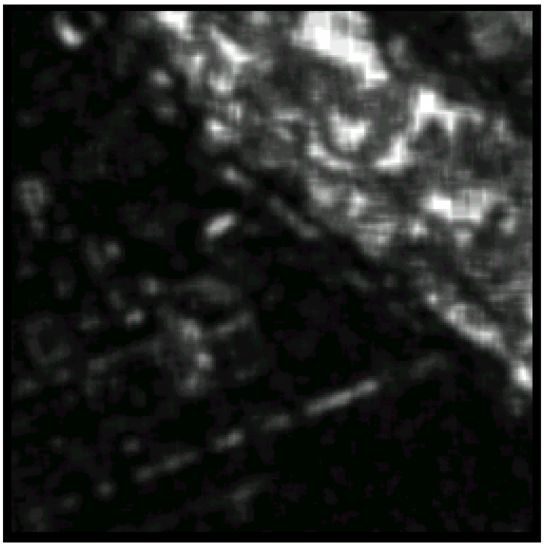
Pixels classified as Urban = 33437

Pixels classified as Non-urban = 26932

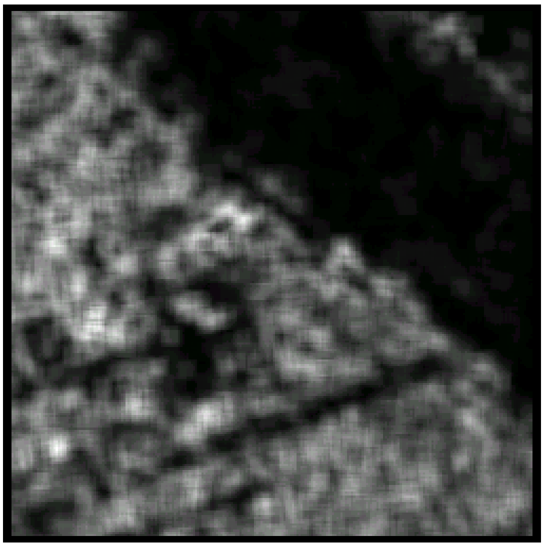


Classified Map

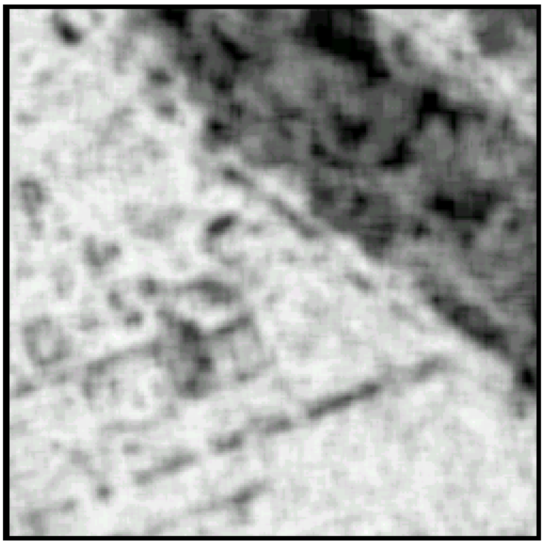
TEXTURE MAPS OF TEST IMAGE B



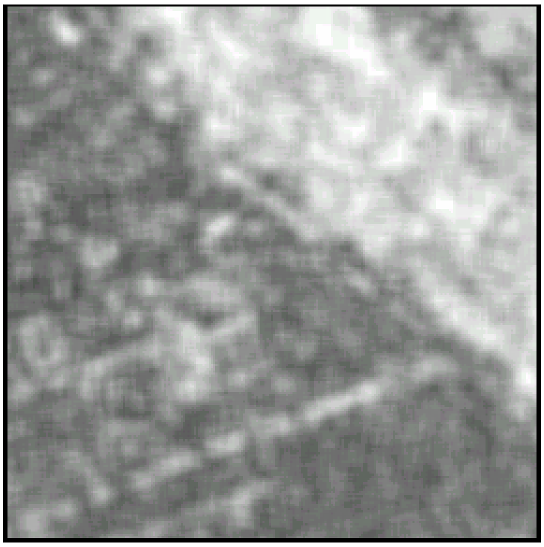
Contrast



Energy



Entropy

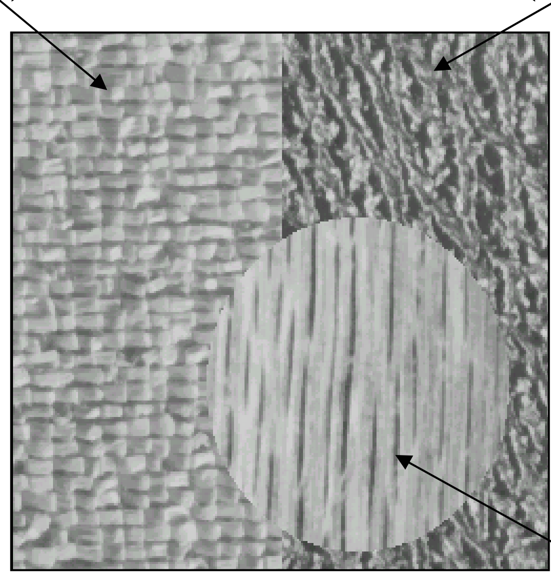


IDM

TEST IMAGE C

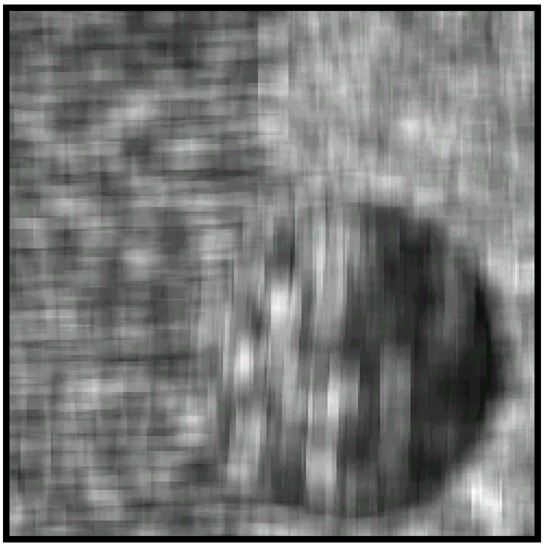
This test image is a composite of three Brodatz textures namely D24, D68 and D84. It was created from these images having circular and vertical boundaries between the textures. This test image has been created in order to evaluate the network performance for classification of three distinct textures. Training sites were selected from the orignal images of these textures, which are of size 512x512 pixels [30]. A total of 600 training samples (200 samples per class) were used to train the network. Texture window of 11x11 pixels was used for extracting the texture parameters. Many trials were made with different number of hidden nodes and values of learning rate to get satisfactory output. The displayed classified map is obtained with 15 hidden nodes and constant learning rate of value 0.0004. Three different colors are used to label the corresponding three textures.

The training data of D24, D68 and D84 are listed in appendix – I C.

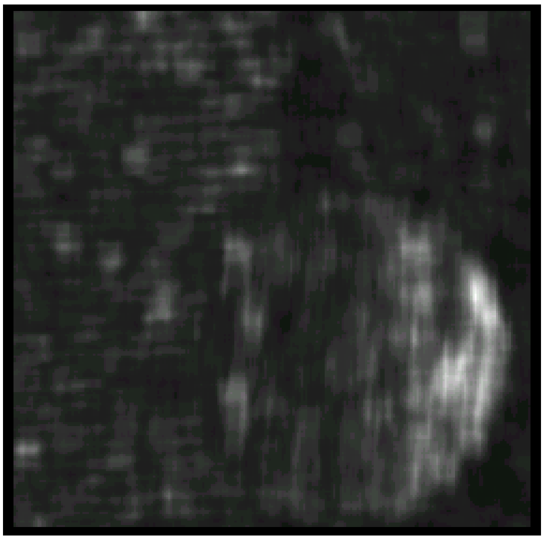


TEST IMAGE C : Mosaic of three distinct textures

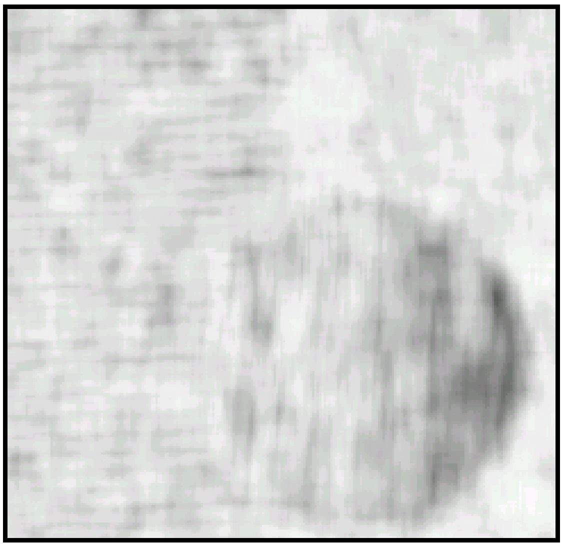
TEXTURE MAPS OF TEST IMAGE C



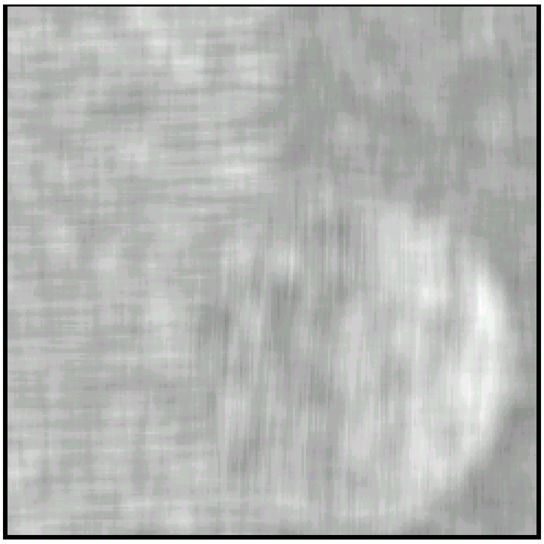
Contrast



Energy

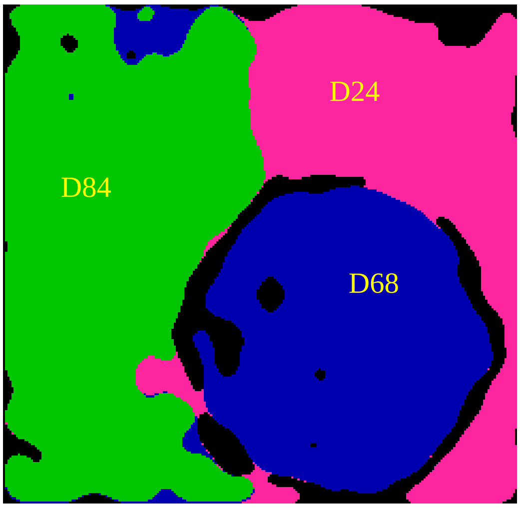


Entropy



IDM

Classified map of TEST IMAGE C



Texture window size = 11x11 pixels

Network 16-15-3

TRNG = 600

EPOCHS = 692

η = 0.0004

Initial MSE = 0.2165

Final MSE = 0.1644

No. of pixels classified as D24 = 13504

No. of pixels classified as D68 = 15263

No. of pixels classified as D84 = 21230

Black areas represent the pixels, which could not qualify for any of the three classes.

Chapter 6
Conclusion

In this project work, the implemented neural network classifier was evaluated for identifying ground cover objects using texture information in aerial and satellite images. The developed method appeared to be feasible for classifying images based on texture information. Selection of proper training samples of known texture classes in images has an important effect on the performance of the network.

The four parameters namely contrast, energy, entropy and inverse difference moment proved to have good potential for texture discrimination. More such texture measures can be included to study their effects on image classification. Besides the implemented GLCM algorithm for texture feature extraction other methods can be explored.

One drawback of backpropagation algorithm is its lengthy time for training process. This can possibly be solved by implementing techniques that improve slow convergence of backpropagation [24],[25]. Efficient methods of network learning could be of interest for future development work.

References

1. W.K. Pratt, *Digital Image Processing*, John Wiley, New York, 1991.
2. R.W. Conners and C.A. Harlow, “A theoretical comparison of texture algorithms”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-2, pp. 204-222, 1980.
3. J.S. Weszka, C.R. Dyer and A. Rosenfeld, "A comparative study of texture measures for terrain classification", IEEE Transactions on Systems, man and Cybernetics, vol. 6, pp. 269-285, 1976.
4. P.P. Ohanian and R.C. Dubes, "Performance evaluation for four classes of texture features", Pattern Recognition, vol. 25, no. 8, pp. 819-833, 1992.
5. J.A. Benediktsson, P.H. Swain and O.K. Ersoy, “Neural network approaces versus statistical methods in classification of multisource remote sensing data”, IEEE Transactions on Geoscience and Remote Sensing, vol 28, pp 540-552, 1990.
6. G.F. Hepner, “ Artificial neural network classification using a minimal training set: comparison to conventional supervised classification”, Photogrammetric Engineering & Remote Sensing, vol. 56, No 4, April 1989, pp.469-473.
7. P.D. Heerman and N. Khazenie, “Classification of multispectral remotesensing data using back-propagation neural network”, IEEE Transactions on Geoscience and Remote Sensing, vol 30, pp 81-88, 1992.
8. H. Bischof, W. Schneider and A.J. Pinz, “ Multispectral classification of Landsat images using neural networks”, IEEE Transactions on Geoscience and Remote Sensing, vol 30, pp 482-490, 1992.
9. L. Bruzzone, C. Conese, F. Maselli and F. Roli. ”Multisource classification of complex rural areas by statistical and neural network approaches”, Photogrammetric Engineering & Remote Sensing, vol. 63, No 5, May 1997, pp. 523-533.
10. S.E. Franklin and D.R. Peddle, “Classification of SPOT HRV imagery and texture features”, Int. J. Remote Sensing, vol. 11, No 3, 1990, pp. 551-556.
11. E. Sali and H. Wolfson, “Texture classification in aerial photographs of satellite data”, Int. J. Remote Sensing, vol. 13, No 8, 1992, pp. 3395- 3408.
12. A. Baraldi and F. Parmiggiani, “Urban area classification by multispectral SPOT images”, IEEE Transactions on Geoscience and Remote Sensing, vol 28, pp 674-680, 1990.
13. T. Ojala, M. Pietikainen, "A comparative study of texture measures with classification based on feature distributions”, Pattern Recognition, vol. 29, no. 1, pp. 51-59, 1996.
14. M.F. Augusteijn, L.E. Clemens and K.A. Shaw, “Performance evaluation of texture measures for ground cover identiication in satellite images by means of a neural network classifier”, IEEE Transactions on Geoscience and Remote Sensing, vol 33,No 3, pp 616-626,1995.
15. J. Lee, R.C. Weger, S.K. Sengupta and R.M. Welch, “ A neural network approach to cloud classification”, IEEE Transactions on Geoscience and Remote Sensing, vol 28, pp 846-855,1990.

16. T. Yoshida and S. Omatu, “ Neural network approach to landcover mapping”, IEEE Transactions on Geoscience and Remote Sensing, vol 32,No 5, pp 1103-1109,1994.
17. I. Kanellopoulos, A. Varfis, G.G. Wilkinson and J. Megier, “Landcover discrimination in SPOT HRV imagery using an artificial neural network- a 20 class experiment”, Int. J. Remote Sensing, vol. 13, No 5, 1992, pp. 917-924.
18. J.D. Paola and R.A. Schowengerdt, ” The effect of Neural Network structure on a multispectral landuse/landcover classification“, Photogrammetric Engineering & Remote Sensing, vol. 63, No 5, May 1997, pp. 535-544. In bpnn
19. J.D. Paola and R.A. Schowengerdt , “ A detailed comparison of backpropagation neural network and maximum-likelihood classifiers for urban landuse classification”, IEEE Transactions on Geoscience and Remote Sensing, vol 33, No 4, pp 981-996,1995.
20. Y.V. Venkatesh and S.K. Raja, “ On the classification of multispectral satellite images using the multilayer perceptron”, Pattern Recognition, vol. 36, pp. 2161-2175, 2003.
21. G. Thimm and E. Fiesler, “ High-order and multilayer perceptron initialisation”, IEEE Transactions on Neural Networks, vol 8, No 2, pp 349-359, 1997.
22. L.F.A. Wessels and E. Barnard, “Avoiding false local minima by proper initialisation of connections”, IEEE Transactions on Neural Networks, vol 3, No 6, pp 899-905, 1992.
23. J.Y.F. Yam and T.W.S. Chow, “ Feedforward networks training speed enhancement by optimal initialisation of the synaptic coefficients”, IEEE Transactions on Neural Networks, vol 12, No 2, pp 430-434, 2001.
24. R.O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification*, 2nd edition, John Wiley & Sons, New York, 2001.
25. Simon Haykin, *Neural Networks* , Prentice Hall of India, 1999.
26. R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, Addison Wesley, United States of America, 1993.
27. T.M. Lillesand and R.W. Keifer, *Remote Sensing and Image Interpretation*, Wiley, New York, 1999.
28. http : //www.nrsa.gov.in/engnrsa/imagegallery
29. http : //landsat.gsfc.nasa.gov/main/images.html
30. http:// sipi.ucs.edu/database

APPENDIX - I

(TRAINING DATA)

Water

1) -0.415 0.434 0.094 0.849 -0.367 0.339 0.173 0.814 -0.430 0.399 0.155 0.862 -0.466 0.384 0.094 0.829
2) -0.346 0.217 0.305 0.836 -0.233 0.105 0.410 0.786 -0.352 0.162 0.412 0.849 -0.466 0.189 0.264 0.829
3) -0.415 0.066 0.365 0.849 -0.233 -0.059 0.501 0.786 -0.430 0.057 0.400 0.862 -0.466 0.018 0.357 0.829
4) -0.346 -0.066 0.495 0.836 -0.100 -0.200 0.654 0.757 -0.274 -0.077 0.563 0.836 -0.403 -0.117 0.489 0.814
5) -0.208 -0.148 0.603 0.811 0.167 -0.300 0.790 0.700 0.039 -0.186 0.734 0.785 -0.279 -0.193 0.590 0.786
6) -0.210 0.703 0.772 0.367 -0.336 0.846 0.657 0.196 -0.228 0.800 0.760 0.031 -0.280 0.732 0.714
7) 0.277 -0.251 0.785 0.721 0.633 -0.330 0.854 0.600 0.274 -0.201 0.788 0.747 0.217 -0.283 0.759 0.671
8) 0.277 -0.219 0.755 0.721 0.567 -0.280 0.795 0.614 0.196 -0.152 0.730 0.760 0.341 -0.260 0.741 0.643
9) 0.485 -0.252 0.806 0.683 0.767 -0.294 0.809 0.571 0.196 -0.119 0.698 0.760 0.403 -0.261 0.744 0.629
10) 0.415 -0.233 0.781 0.696 0.700 -0.277 0.789 0.586 0.352 -0.132 0.728 0.734 0.466 -0.261 0.743 0.614
11) -0.485 0.519 -0.014 0.862 -0.433 0.429 0.060 0.829 -0.430 0.477 0.070 0.862 -0.528 0.477 -0.022 0.843
12) -0.415 0.323 0.186 0.849 -0.300 0.217 0.285 0.800 -0.352 0.253 0.326 0.849 -0.528 0.306 0.135 0.843
13) -0.485 0.175 0.250 0.862 -0.300 0.052 0.385 0.800 -0.430 0.148 0.322 0.862 -0.528 0.135 0.236 0.843
14) -0.485 0.039 0.363 0.862 -0.233 -0.100 0.537 0.786 -0.352 0.007 0.468 0.849 -0.466 -0.025 0.392 0.829
15) -0.346 -0.077 0.504 0.836 -0.033 -0.223 0.688 0.743 -0.039 -0.132 0.670 0.798 -0.341 -0.145 0.532 0.800
16) -0.138 -0.165 0.634 0.798 0.167 -0.297 0.787 0.700 0.039 -0.174 0.723 0.785 -0.093 -0.246 0.680 0.743
17) 0.069 -0.227 0.731 0.760 0.433 -0.349 0.866 0.643 0.117 -0.209 0.769 0.772 0.031 -0.283 0.736 0.714
18) 0.138 -0.234 0.750 0.747 0.433 -0.338 0.854 0.643 0.039 -0.186 0.734 0.785 0.217 -0.314 0.789 0.671
19) 0.346 -0.276 0.816 0.709 0.633 -0.364 0.892 0.600 0.039 -0.174 0.723 0.785 0.279 -0.323 0.803 0.657
20) 0.277 -0.259 0.792 0.721 0.567 -0.350 0.874 0.614 0.196 -0.200 0.775 0.760 0.341 -0.330 0.815 0.643
21) -0.554 0.608 -0.133 0.874 -0.567 0.576 -0.156 0.857 -0.509 0.608 -0.118 0.874 -0.652 0.627 -0.241 0.871
22) -0.485 0.439 0.052 0.862 -0.433 0.389 0.023 0.829 -0.430 0.400 0.133 0.862 -0.590 0.435 -0.013 0.857
23) -0.554 0.302 0.115 0.874 -0.367 0.187 0.243 0.814 -0.430 0.263 0.217 0.862 -0.590 0.276 0.090 0.857
24) -0.554 0.159 0.239 0.874 -0.300 0.024 0.411 0.800 -0.352 0.113 0.376 0.849 -0.528 0.106 0.260 0.843
25) -0.485 0.039 0.363 0.862 -0.167 -0.111 0.561 0.771 -0.117 -0.027 0.559 0.811 -0.403 -0.038 0.424 0.814
26) -0.346 -0.053 0.485 0.836 0.033 -0.209 0.686 0.729 -0.039 -0.075 0.618 0.798 -0.217 -0.147 0.567 0.771
27) -0.138 -0.151 0.623 0.798 0.300 -0.305 0.810 0.671 0.039 -0.166 0.716 0.785 -0.093 -0.223 0.660 0.743
28) -0.201 0.696 0.772 0.367 -0.330 0.841 0.657 -0.039 -0.152 0.687 0.798 0.155 -0.304 0.772 0.686
29) 0.208 -0.252 0.777 0.734 0.567 -0.365 0.891 0.614 -0.039 -0.165 0.699 0.798 0.217 -0.321 0.795 0.671
30) 0.138 -0.239 0.754 0.747 0.500 -0.358 0.879 0.629 0.117 -0.206 0.767 0.772 0.279 -0.336 0.816 0.657
31) -0.623 0.657 -0.261 0.887 -0.500 0.630 -0.287 0.843 -0.509 0.702 -0.290 0.874 -0.714 0.733 -0.425 0.886
32) -0.485 0.481 -0.042 0.862 -0.300 0.436 -0.041 0.800 -0.430 0.522 -0.053 0.862 -0.652 0.579 -0.230 0.871
33) -0.485 0.330 0.111 0.862 -0.233 0.259 0.180 0.786 -0.352 0.361 0.137 0.849 -0.590 0.394 -0.021 0.857
34) -0.485 0.203 0.226 0.862 -0.167 0.106 0.351 0.771 -0.274 0.219 0.295 0.836 -0.528 0.234 0.149 0.843
35) -0.485 0.123 0.294 0.862 -0.033 -0.027 0.500 0.743 -0.039 0.084 0.467 0.798 -0.466 0.127 0.264 0.829
36) -0.346 0.056 0.394 0.836 0.100 -0.088 0.574 0.714 -0.039 0.084 0.467 0.798 -0.341 0.052 0.363 0.800
37) -0.208 -0.027 0.501 0.811 0.300 -0.171 0.669 0.671 -0.039 -0.016 0.564 0.798 -0.155 -0.075 0.515 0.757
38) 0.069 -0.136 0.650 0.760 0.367 -0.218 0.723 0.657 0.039 -0.045 0.604 0.785 0.217 -0.216 0.690 0.671
39) 0.277 -0.206 0.742 0.721 0.633 -0.279 0.794 0.600 0.117 -0.111 0.679 0.772 0.341 -0.260 0.741 0.643
40) 0.208 -0.211 0.738 0.734 0.633 -0.298 0.816 0.600 0.274 -0.176 0.763 0.747 0.403 -0.296 0.783 0.629
41) -0.554 0.703 -0.315 0.874 -0.433 0.632 -0.333 0.829 -0.509 0.750 -0.366 0.874 -0.652 0.733 -0.428 0.871
42) -0.415 0.565 -0.136 0.849 -0.233 0.489 -0.171 0.786 -0.430 0.609 -0.163 0.862 -0.590 0.628 -0.285 0.857
43) -0.415 0.438 0.014 0.849 -0.167 0.348 0.057 0.771 -0.352 0.478 0.005 0.849 -0.528 0.481 -0.101 0.843
44) -0.415 0.325 0.131 0.849 -0.100 0.220 0.221 0.757 -0.274 0.358 0.150 0.836 -0.466 0.346 0.054 0.829
45) -0.415 0.257 0.196 0.849 -0.033 0.142 0.313 0.743 -0.117 0.244 0.290 0.811 -0.466 0.265 0.135 0.829
46) -0.346 0.252 0.216 0.836 -0.033 0.142 0.313 0.743 -0.274 0.321 0.191 0.836 -0.403 0.262 0.149 0.814
47) -0.208 0.178 0.310 0.811 0.100 0.071 0.392 0.714 -0.274 0.251 0.263 0.836 -0.279 0.180 0.247 0.786
48) 0.069 0.045 0.473 0.760 0.300 -0.026 0.495 0.671 -0.117 0.178 0.361 0.811 0.031 0.001 0.451 0.714
49) 0.346 -0.069 0.607 0.709 0.567 -0.134 0.610 0.614 -0.039 0.057 0.494 0.798 0.155 -0.091 0.553 0.686
50) 0.208 -0.111 0.643 0.734 0.500 -0.193 0.693 0.629 0.039 -0.045 0.604 0.785 0.217 -0.171 0.642 0.671
51) -0.623 0.800 -0.452 0.887 -0.500 0.683 -0.394 0.843 -0.587 0.799 -0.433 0.887 -0.652 0.733 -0.428 0.871
52) -0.485 0.704 -0.327 0.862 -0.367 0.583 -0.276 0.814 -0.509 0.702 -0.290 0.874 -0.590 0.681 -0.361 0.857
53) -0.485 0.610 -0.193 0.862 -0.300 0.485 -0.117 0.800 -0.430 0.609 -0.163 0.862 -0.528 0.578 -0.222 0.843
54) -0.485 0.522 -0.087 0.862 -0.233 0.391 0.009 0.786 -0.352 0.521 -0.047 0.849 -0.466 0.480 -0.100 0.829
55) -0.554 0.484 -0.058 0.874 -0.233 0.345 0.073 0.786 -0.274 0.436 0.060 0.836 -0.466 0.434 -0.043 0.829
56) -0.485 0.481 -0.042 0.862 -0.367 0.434 -0.030 0.814 -0.430 0.522 -0.053 0.862 -0.403 0.434 -0.045 0.814
57) -0.415 0.438 0.014 0.849 -0.300 0.389 0.023 0.800 -0.509 0.524 -0.067 0.874 -0.403 0.434 -0.045 0.814
58) -0.069 0.243 0.252 0.785 -0.100 0.262 0.166 0.757 -0.274 0.396 0.107 0.836 -0.093 0.222 0.190 0.743
59) 0.138 0.074 0.445 0.747 0.167 0.074 0.380 0.700 -0.117 0.210 0.327 0.811 0.093 0.073 0.358 0.700
60) -0.005 0.517 0.772 0.167 -0.030 0.510 0.700 0.039 0.052 0.509 0.785 0.155 -0.061 0.518 0.686
61) -0.623 0.800 -0.452 0.887 -0.500 0.683 -0.394 0.843 -0.587 0.799 -0.433 0.887 -0.652 0.733 -0.428 0.871
62) -0.554 0.751 -0.387 0.874 -0.433 0.632 -0.333 0.829 -0.509 0.750 -0.366 0.874 -0.652 0.733 -0.428 0.871

63) -0.554 0.703 -0.315 0.874 -0.367 0.583 -0.276 0.814 -0.430 0.703 -0.303 0.862 -0.590 0.681 -0.361 0.857
64) -0.554 0.656 -0.253 0.874 -0.300 0.535 -0.222 0.800 -0.352 0.657 -0.243 0.849 -0.528 0.629 -0.298 0.843
65) -0.623 0.657 -0.261 0.887 -0.300 0.535 -0.222 0.800 -0.274 0.612 -0.187 0.836 -0.528 0.629 -0.298 0.843
66) -0.623 0.703 -0.317 0.887 -0.500 0.683 -0.394 0.843 -0.430 0.703 -0.303 0.862 -0.528 0.683 -0.402 0.843
67) -0.623 0.703 -0.317 0.887 -0.500 0.683 -0.394 0.843 -0.509 0.750 -0.366 0.874 -0.590 0.735 -0.466 0.857
68) -0.277 0.479 -0.032 0.823 -0.300 0.535 -0.222 0.800 -0.274 0.566 -0.112 0.836 -0.279 0.488 -0.183 0.786
69) -0.138 0.280 0.209 0.798 -0.100 0.306 0.103 0.757 -0.117 0.356 0.162 0.811 -0.155 0.306 0.086 0.757
70) -0.208 0.147 0.341 0.811 -0.033 0.142 0.313 0.743 0.039 0.173 0.381 0.785 -0.031 0.105 0.332 0.729
71) -0.485 0.704 -0.327 0.862 -0.433 0.632 -0.333 0.829 -0.509 0.750 -0.366 0.874 -0.590 0.681 -0.361 0.857
72) -0.485 0.704 -0.327 0.862 -0.433 0.632 -0.333 0.829 -0.509 0.750 -0.366 0.874 -0.590 0.681 -0.361 0.857
73) -0.554 0.751 -0.387 0.874 -0.433 0.632 -0.333 0.829 -0.509 0.750 -0.366 0.874 -0.652 0.733 -0.428 0.871
74) -0.554 0.751 -0.387 0.874 -0.500 0.683 -0.394 0.843 -0.509 0.750 -0.366 0.874 -0.590 0.681 -0.361 0.857
75) -0.623 0.751 -0.379 0.887 -0.500 0.683 -0.394 0.843 -0.430 0.703 -0.303 0.862 -0.590 0.681 -0.361 0.857
76) -0.623 0.800 -0.452 0.887 -0.633 0.789 -0.527 0.871 -0.587 0.799 -0.433 0.887 -0.652 0.789 -0.533 0.871
77) -0.623 0.800 -0.452 0.887 -0.700 0.844 -0.601 0.886 -0.587 0.850 -0.537 0.887 -0.714 0.844 -0.607 0.886
78) -0.346 0.566 -0.144 0.836 -0.567 0.681 -0.351 0.857 -0.352 0.610 -0.168 0.849 -0.403 0.583 -0.287 0.814
79) -0.277 0.357 0.116 0.823 -0.433 0.434 -0.028 0.829 -0.274 0.396 0.107 0.836 -0.279 0.345 0.056 0.786
80) -0.277 0.183 0.295 0.823 -0.367 0.224 0.207 0.814 -0.196 0.214 0.313 0.823 -0.217 0.142 0.292 0.771
81) -0.485 0.704 -0.327 0.862 -0.367 0.583 -0.276 0.814 -0.430 0.703 -0.303 0.862 -0.528 0.629 -0.298 0.843
82) -0.485 0.704 -0.327 0.862 -0.367 0.583 -0.276 0.814 -0.430 0.703 -0.303 0.862 -0.528 0.629 -0.298 0.843
83) -0.554 0.751 -0.387 0.874 -0.433 0.632 -0.333 0.829 -0.430 0.703 -0.303 0.862 -0.590 0.681 -0.361 0.857
84) -0.485 0.704 -0.327 0.862 -0.433 0.632 -0.333 0.829 -0.509 0.702 -0.290 0.874 -0.528 0.629 -0.298 0.843
85) -0.554 0.656 -0.253 0.874 -0.433 0.580 -0.227 0.829 -0.430 0.609 -0.163 0.862 -0.528 0.578 -0.222 0.843
86) -0.554 0.656 -0.253 0.874 -0.500 0.630 -0.287 0.843 -0.509 0.655 -0.225 0.874 -0.590 0.628 -0.285 0.857
87) -0.623 0.657 -0.261 0.887 -0.633 0.680 -0.342 0.871 -0.587 0.702 -0.292 0.887 -0.590 0.628 -0.285 0.857
88) -0.415 0.438 0.014 0.849 -0.500 0.481 -0.087 0.843 -0.352 0.438 0.052 0.849 -0.341 0.389 0.007 0.800
89) -0.346 0.219 0.247 0.836 -0.367 0.224 0.207 0.814 -0.274 0.219 0.295 0.836 -0.217 0.142 0.292 0.771
90) -0.346 0.056 0.394 0.836 -0.300 0.024 0.411 0.800 -0.196 0.046 0.475 0.823 -0.155 -0.050 0.491 0.757
91) -0.485 0.704 -0.327 0.862 -0.367 0.583 -0.276 0.814 -0.430 0.703 -0.303 0.862 -0.528 0.629 -0.298 0.843
92) -0.485 0.704 -0.327 0.862 -0.367 0.583 -0.276 0.814 -0.430 0.703 -0.303 0.862 -0.528 0.629 -0.298 0.843
93) -0.554 0.751 -0.387 0.874 -0.433 0.632 -0.333 0.829 -0.430 0.703 -0.303 0.862 -0.590 0.681 -0.361 0.857
94) -0.415 0.657 -0.270 0.849 -0.367 0.583 -0.276 0.814 -0.509 0.655 -0.225 0.874 -0.466 0.580 -0.239 0.829
95) -0.485 0.565 -0.137 0.862 -0.367 0.482 -0.094 0.814 -0.430 0.522 -0.053 0.862 -0.466 0.480 -0.100 0.829
96) -0.485 0.522 -0.087 0.862 -0.433 0.481 -0.086 0.829 -0.509 0.524 -0.067 0.874 -0.528 0.481 -0.101 0.843
97) -0.554 0.484 -0.058 0.874 -0.567 0.483 -0.098 0.857 -0.587 0.527 -0.088 0.887 -0.528 0.435 -0.050 0.843
98) -0.346 0.252 0.216 0.836 -0.433 0.265 0.153 0.829 -0.352 0.256 0.242 0.849 -0.279 0.180 0.247 0.786
99) -0.277 0.047 0.420 0.823 -0.300 0.024 0.411 0.800 -0.274 0.056 0.448 0.836 -0.155 -0.050 0.491 0.757
100) -0.277 -0.080 0.528 0.823 -0.233 -0.134 0.566 0.786 -0.196 -0.081 0.588 0.823 -0.093 -0.199 0.639 0.743
101) 0.371 0.210 0.331 0.773 0.245 0.144 0.336 0.716 0.100 0.280 0.259 0.811 0.300 0.144 0.326 0.729
102) 0.265 0.321 0.186 0.786 0.082 0.262 0.194 0.744 -0.100 0.396 0.114 0.836 0.214 0.265 0.165 0.743
103) -0.053 0.440 0.032 0.824 -0.164 0.389 0.047 0.787 -0.200 0.521 -0.041 0.849 0.043 0.348 0.074 0.771
104) -0.265 0.524 -0.085 0.849 -0.327 0.480 -0.065 0.815 -0.200 0.521 -0.041 0.849 -0.129 0.435 -0.026 0.800
105) -0.265 0.612 -0.210 0.849 -0.491 0.628 -0.258 0.843 -0.400 0.655 -0.221 0.874 -0.300 0.580 -0.215 0.829
106) -0.053 0.616 -0.274 0.824 -0.327 0.632 -0.318 0.815 -0.400 0.610 -0.163 0.874 -0.300 0.632 -0.323 0.829
107) -0.476 0.658 -0.277 0.875 -0.409 0.578 -0.193 0.829 -0.400 0.610 -0.163 0.874 -0.471 0.628 -0.263 0.857
108) -0.371 0.484 -0.046 0.862 -0.327 0.389 0.047 0.815 -0.300 0.402 0.086 0.862 -0.129 0.344 0.097 0.800
109) -0.159 0.224 0.271 0.837 -0.164 0.113 0.359 0.787 -0.100 0.130 0.388 0.836 0.043 0.072 0.409 0.771
110) -0.053 0.029 0.483 0.824 -0.082 -0.081 0.557 0.772 0.100 -0.046 0.586 0.811 0.300 -0.156 0.662 0.729
111) 0.371 0.210 0.331 0.773 0.327 0.108 0.376 0.702 0.200 0.243 0.306 0.798 0.386 0.108 0.365 0.714
112) 0.265 0.321 0.186 0.786 0.082 0.262 0.194 0.744 0.000 0.356 0.166 0.823 0.300 0.226 0.207 0.729
113) -0.053 0.440 0.032 0.824 -0.164 0.389 0.047 0.787 -0.200 0.521 -0.041 0.849 0.043 0.348 0.074 0.771
114) -0.265 0.524 -0.085 0.849 -0.327 0.480 -0.065 0.815 -0.200 0.521 -0.041 0.849 -0.129 0.435 -0.026 0.800
115) -0.265 0.612 -0.210 0.849 -0.491 0.628 -0.258 0.843 -0.400 0.655 -0.221 0.874 -0.300 0.580 -0.215 0.829
116) -0.053 0.616 -0.274 0.824 -0.327 0.632 -0.318 0.815 -0.400 0.610 -0.163 0.874 -0.214 0.583 -0.265 0.814
117) -0.371 0.612 -0.208 0.862 -0.409 0.578 -0.193 0.829 -0.200 0.521 -0.041 0.849 -0.300 0.529 -0.138 0.829
118) -0.159 0.400 0.077 0.837 -0.245 0.344 0.104 0.801 -0.200 0.361 0.144 0.849 -0.043 0.301 0.148 0.786
119) 0.053 0.152 0.377 0.811 -0.082 0.076 0.407 0.772 -0.100 0.130 0.388 0.836 0.043 0.072 0.409 0.771
120) 0.159 -0.032 0.578 0.799 -0.082 -0.081 0.557 0.772 0.000 -0.017 0.542 0.823 0.214 -0.128 0.626 0.743
121) 0.371 0.210 0.331 0.773 0.245 0.184 0.280 0.716 0.100 0.317 0.216 0.811 0.129 0.220 0.241 0.757
122) 0.265 0.321 0.186 0.786 0.000 0.348 0.082 0.758 -0.100 0.436 0.066 0.836 0.043 0.348 0.074 0.771
123) -0.053 0.440 0.032 0.824 -0.245 0.482 -0.073 0.801 -0.300 0.609 -0.158 0.862 -0.214 0.482 -0.080 0.814
124) -0.265 0.524 -0.085 0.849 -0.409 0.578 -0.193 0.829 -0.200 0.565 -0.099 0.849 -0.300 0.529 -0.138 0.829
125) -0.371 0.612 -0.208 0.862 -0.491 0.681 -0.337 0.843 -0.300 0.655 -0.223 0.862 -0.300 0.580 -0.215 0.829
126) -0.053 0.525 -0.092 0.824 -0.245 0.583 -0.260 0.801 -0.100 0.521 -0.043 0.836 -0.043 0.488 -0.158 0.786
127) -0.265 0.442 0.017 0.849 -0.245 0.434 -0.007 0.801 0.100 0.356 0.169 0.811 0.043 0.348 0.074 0.771
128) -0.053 0.219 0.292 0.824 0.000 0.179 0.303 0.758 0.100 0.210 0.335 0.811 0.214 0.142 0.335 0.743

129) 0.053 0.020 0.510 0.811 0.245 -0.059 0.575 0.716 0.300 0.000 0.570 0.785 0.300 -0.059 0.563 0.729
130) 0.159 -0.116 0.657 0.799 0.327 -0.201 0.728 0.702 0.500 -0.136 0.726 0.760 0.557 -0.231 0.758 0.686
131) 0.371 0.210 0.331 0.773 0.409 0.194 0.221 0.687 0.300 0.318 0.212 0.785 0.129 0.262 0.185 0.757
132) 0.371 0.284 0.232 0.773 0.164 0.357 -0.008 0.730 0.200 0.397 0.110 0.798 0.129 0.351 0.045 0.757
133) 0.053 0.359 0.138 0.811 0.000 0.443 -0.101 0.758 0.100 0.525 -0.077 0.811 -0.043 0.439 -0.051 0.786
134) -0.159 0.400 0.077 0.837 -0.164 0.484 -0.096 0.787 0.200 0.439 0.046 0.798 -0.043 0.439 -0.051 0.786
135) -0.265 0.442 0.017 0.849 -0.327 0.529 -0.131 0.815 0.100 0.480 -0.002 0.811 -0.043 0.439 -0.051 0.786
136) 0.053 0.321 0.186 0.811 -0.164 0.389 0.047 0.787 0.200 0.317 0.216 0.798 0.129 0.306 0.121 0.757
137) -0.159 0.224 0.271 0.837 -0.164 0.220 0.248 0.787 0.300 0.140 0.425 0.785 0.300 0.105 0.376 0.729
138) -0.053 0.052 0.461 0.824 0.000 0.011 0.483 0.758 0.300 0.025 0.546 0.785 0.386 -0.061 0.569 0.714
139) 0.053 -0.090 0.614 0.811 0.245 -0.176 0.695 0.716 0.500 -0.136 0.726 0.760 0.471 -0.207 0.728 0.700
140) 0.053 -0.147 0.665 0.811 0.327 -0.267 0.792 0.702 0.700 -0.232 0.839 0.734 0.643 -0.306 0.841 0.671
141) 0.371 0.176 0.374 0.773 0.245 0.144 0.336 0.716 0.100 0.280 0.259 0.811 0.129 0.179 0.292 0.757
142) 0.371 0.246 0.284 0.773 0.082 0.262 0.194 0.744 0.100 0.317 0.216 0.811 0.129 0.262 0.185 0.757
143) -0.053 0.360 0.133 0.824 0.000 0.303 0.147 0.758 0.100 0.396 0.118 0.811 -0.043 0.345 0.090 0.786
144) -0.053 0.360 0.133 0.824 0.000 0.303 0.147 0.758 0.100 0.317 0.216 0.811 -0.129 0.344 0.097 0.800
145) -0.159 0.362 0.121 0.837 -0.082 0.301 0.156 0.772 0.100 0.317 0.216 0.811 -0.043 0.301 0.148 0.786
146) 0.159 0.213 0.321 0.799 0.164 0.142 0.345 0.730 0.200 0.175 0.381 0.798 0.129 0.179 0.292 0.757
147) -0.053 0.102 0.413 0.824 0.000 0.040 0.453 0.758 0.300 0.025 0.546 0.785 0.214 0.003 0.493 0.743
148) -0.053 -0.030 0.538 0.824 0.164 -0.128 0.639 0.730 0.300 -0.065 0.634 0.785 0.214 -0.128 0.626 0.743
149) 0.053 -0.134 0.653 0.811 0.409 -0.267 0.802 0.687 0.500 -0.191 0.777 0.760 0.300 -0.235 0.739 0.729
150) 0.053 -0.156 0.673 0.811 0.409 -0.300 0.834 0.687 0.700 -0.256 0.862 0.734 0.471 -0.300 0.820 0.700
151) 0.688 -0.062 0.664 0.735 0.655 -0.117 0.644 0.645 0.400 0.048 0.531 0.772 0.300 -0.029 0.532 0.729
152) 0.582 0.019 0.567 0.748 0.409 0.002 0.507 0.687 0.300 0.109 0.459 0.785 0.300 0.034 0.461 0.729
153) 0.159 0.118 0.426 0.799 0.245 0.069 0.431 0.716 0.400 0.139 0.432 0.772 0.129 0.104 0.379 0.757
154) 0.159 0.118 0.426 0.799 0.245 0.069 0.431 0.716 0.400 0.077 0.500 0.772 0.129 0.104 0.379 0.757
155) 0.053 0.123 0.408 0.811 0.164 0.069 0.432 0.730 0.400 0.077 0.500 0.772 0.214 0.069 0.421 0.743
156) 0.265 0.030 0.529 0.786 0.327 -0.031 0.548 0.702 0.500 -0.009 0.598 0.760 0.300 0.001 0.498 0.729
157) 0.053 -0.023 0.551 0.811 0.082 -0.075 0.578 0.744 0.600 -0.106 0.705 0.747 0.386 -0.115 0.628 0.714
158) 0.053 -0.103 0.626 0.811 0.164 -0.184 0.693 0.730 0.500 -0.152 0.741 0.760 0.300 -0.193 0.699 0.729
159) 0.053 -0.152 0.669 0.811 0.327 -0.274 0.799 0.702 0.600 -0.229 0.825 0.747 0.386 -0.274 0.785 0.714
160) 0.053 -0.147 0.665 0.811 0.327 -0.283 0.808 0.702 0.800 -0.274 0.889 0.721 0.557 -0.313 0.841 0.686
161) 0.688 -0.209 0.819 0.735 0.655 -0.257 0.809 0.645 0.400 -0.111 0.690 0.772 0.300 -0.193 0.699 0.729
162) 0.582 -0.158 0.756 0.748 0.491 -0.192 0.731 0.673 0.500 -0.100 0.690 0.760 0.471 -0.188 0.708 0.700
163) 0.159 -0.070 0.614 0.799 0.409 -0.166 0.699 0.687 0.600 -0.106 0.705 0.747 0.300 -0.134 0.641 0.729
164) 0.159 -0.070 0.614 0.799 0.409 -0.166 0.699 0.687 0.600 -0.144 0.743 0.747 0.300 -0.134 0.641 0.729
165) 0.053 -0.059 0.585 0.811 0.327 -0.162 0.689 0.702 0.600 -0.144 0.743 0.747 0.386 -0.162 0.676 0.714
166) 0.159 -0.102 0.644 0.799 0.409 -0.207 0.742 0.687 0.700 -0.184 0.792 0.734 0.386 -0.182 0.696 0.714
167) -0.053 -0.100 0.600 0.824 0.164 -0.199 0.707 0.730 0.800 -0.232 0.848 0.721 0.471 -0.241 0.762 0.700
168) -0.053 -0.130 0.626 0.824 0.245 -0.254 0.770 0.716 0.700 -0.241 0.847 0.734 0.386 -0.267 0.778 0.714
169) -0.053 -0.133 0.629 0.824 0.327 -0.285 0.810 0.702 0.800 -0.274 0.889 0.721 0.471 -0.301 0.820 0.700
170) -0.053 -0.100 0.600 0.824 0.245 -0.246 0.762 0.716 0.900 -0.276 0.900 0.709 0.557 -0.296 0.825 0.686
171) 0.582 -0.243 0.840 0.748 0.573 -0.321 0.870 0.659 0.400 -0.195 0.768 0.772 0.214 -0.246 0.736 0.743
172) 0.476 -0.221 0.805 0.761 0.409 -0.286 0.820 0.687 0.500 -0.209 0.794 0.760 0.386 -0.274 0.785 0.714
173) 0.159 -0.164 0.701 0.799 0.409 -0.286 0.820 0.687 0.600 -0.229 0.825 0.747 0.300 -0.254 0.756 0.729
174) 0.159 -0.164 0.701 0.799 0.409 -0.286 0.820 0.687 0.600 -0.239 0.835 0.747 0.300 -0.254 0.756 0.729
175) 0.053 -0.147 0.665 0.811 0.327 -0.274 0.799 0.702 0.600 -0.239 0.835 0.747 0.386 -0.274 0.785 0.714
176) 0.053 -0.152 0.669 0.811 0.327 -0.280 0.804 0.702 0.600 -0.242 0.837 0.747 0.386 -0.280 0.791 0.714
177) -0.159 -0.115 0.588 0.837 0.000 -0.214 0.691 0.758 0.600 -0.244 0.839 0.747 0.386 -0.285 0.796 0.714
178) -0.159 -0.111 0.584 0.837 0.000 -0.209 0.687 0.758 0.400 -0.206 0.778 0.772 0.214 -0.246 0.736 0.743
179) -0.159 -0.084 0.561 0.837 0.000 -0.175 0.657 0.758 0.500 -0.200 0.786 0.760 0.300 -0.235 0.739 0.729
180) -0.159 -0.034 0.517 0.837 -0.082 -0.101 0.575 0.772 0.500 -0.152 0.741 0.760 0.300 -0.176 0.682 0.729
181) 0.688 -0.253 0.863 0.735 0.409 -0.293 0.826 0.687 0.300 -0.190 0.748 0.785 0.129 -0.225 0.704 0.757
182) 0.582 -0.240 0.837 0.748 0.327 -0.280 0.804 0.702 0.500 -0.225 0.809 0.760 0.300 -0.261 0.762 0.729
183) 0.371 -0.206 0.775 0.773 0.409 -0.293 0.826 0.687 0.600 -0.235 0.830 0.747 0.300 -0.261 0.762 0.729
184) 0.371 -0.206 0.775 0.773 0.409 -0.293 0.826 0.687 0.600 -0.221 0.817 0.747 0.300 -0.261 0.762 0.729
185) 0.265 -0.186 0.739 0.786 0.327 -0.274 0.799 0.702 0.600 -0.221 0.817 0.747 0.386 -0.274 0.785 0.714
186) 0.159 -0.158 0.695 0.799 0.327 -0.267 0.792 0.702 0.600 -0.212 0.809 0.747 0.300 -0.246 0.748 0.729
187) -0.159 -0.074 0.552 0.837 -0.082 -0.148 0.617 0.772 0.500 -0.166 0.754 0.760 0.386 -0.233 0.746 0.714
188) -0.159 -0.034 0.517 0.837 0.000 -0.111 0.599 0.758 0.400 -0.111 0.690 0.772 0.214 -0.149 0.646 0.743
189) -0.265 0.050 0.416 0.849 -0.082 -0.011 0.492 0.772 0.400 -0.052 0.633 0.772 0.300 -0.086 0.592 0.729
190) -0.371 0.154 0.294 0.862 -0.164 0.113 0.359 0.787 0.400 0.048 0.531 0.772 0.214 0.035 0.459 0.743
191) 0.265 -0.128 0.686 0.786 0.327 -0.201 0.728 0.702 0.200 -0.106 0.657 0.798 0.043 -0.130 0.604 0.771
192) 0.265 -0.128 0.686 0.786 0.327 -0.201 0.728 0.702 0.400 -0.142 0.719 0.772 0.214 -0.167 0.664 0.743
193) 0.159 -0.102 0.644 0.799 0.409 -0.207 0.742 0.687 0.600 -0.144 0.743 0.747 0.300 -0.176 0.682 0.729
194) 0.159 -0.102 0.644 0.799 0.409 -0.207 0.742 0.687 0.600 -0.106 0.705 0.747 0.300 -0.176 0.682 0.729
195) 0.053 -0.075 0.600 0.811 0.327 -0.182 0.710 0.702 0.600 -0.106 0.705 0.747 0.386 -0.182 0.696 0.714
196) -0.053 -0.030 0.538 0.824 0.245 -0.134 0.654 0.716 0.600 -0.085 0.684 0.747 0.300 -0.134 0.641 0.729
197) -0.265 0.072 0.396 0.849 -0.082 0.016 0.466 0.772 0.300 0.025 0.546 0.785 0.214 -0.055 0.554 0.743
198) -0.159 0.136 0.359 0.837 -0.082 0.109 0.374 0.772 0.300 0.109 0.459 0.785 0.129 0.070 0.417 0.757
199) -0.265 0.262 0.214 0.849 -0.164 0.260 0.205 0.787 0.300 0.207 0.350 0.785 0.214 0.181 0.285 0.743
200) -0.371 0.406 0.042 0.862 -0.245 0.434 -0.007 0.801 0.300 0.358 0.156 0.785 0.129 0.351 0.045 0.757

Land

1) -0.074 0.133 0.495 0.770 0.028 -0.022 0.585 0.704 0.329 -0.127 0.546 0.526 -0.077 -0.056 0.498 0.538
2) -0.103 0.085 0.517 0.785 -0.084 -0.066 0.594 0.730 0.307 -0.197 0.595 0.576 -0.044 -0.164 0.569 0.527
3) -0.074 -0.032 0.564 0.770 -0.141 -0.135 0.607 0.723 0.351 -0.241 0.638 0.546 -0.011 -0.236 0.607 0.516
4) -0.074 -0.086 0.585 0.770 -0.112 -0.208 0.647 0.707 0.439 -0.278 0.661 0.523 0.077 -0.308 0.663 0.502
5) 0.015 -0.130 0.638 0.727 -0.084 -0.223 0.719 0.730 0.615 -0.324 0.747 0.511 0.263 -0.357 0.733 0.455
6) 0.133 -0.118 0.700 0.704 -0.084 -0.218 0.747 0.730 0.659 -0.335 0.790 0.517 0.373 -0.309 0.743 0.491
7) 0.103 -0.143 0.725 0.719 -0.028 -0.253 0.777 0.698 0.571 -0.349 0.805 0.576 0.318 -0.331 0.778 0.538
8) 0.221 -0.196 0.773 0.696 -0.056 -0.259 0.768 0.714 0.637 -0.315 0.791 0.567 0.406 -0.325 0.783 0.567
9) 0.251 -0.183 0.762 0.681 0.028 -0.253 0.769 0.665 0.724 -0.222 0.751 0.579 0.505 -0.296 0.769 0.534
10) 0.310 -0.084 0.741 0.687 0.112 -0.193 0.769 0.656 0.746 -0.102 0.710 0.599 0.615 -0.168 0.727 0.527
11) -0.103 0.054 0.560 0.785 -0.253 0.057 0.546 0.788 0.132 -0.153 0.585 0.622 -0.110 -0.087 0.543 0.592
12) -0.133 -0.010 0.578 0.799 -0.253 -0.024 0.577 0.788 0.110 -0.179 0.606 0.672 -0.088 -0.185 0.606 0.599
13) -0.133 -0.077 0.605 0.799 -0.225 -0.124 0.621 0.771 0.220 -0.233 0.658 0.634 -0.044 -0.241 0.629 0.570
14) -0.133 -0.069 0.606 0.799 -0.225 -0.139 0.638 0.771 0.307 -0.225 0.666 0.611 0.044 -0.271 0.668 0.556
15) -0.103 0.029 0.612 0.785 -0.169 -0.082 0.683 0.778 0.461 -0.170 0.719 0.614 0.154 -0.192 0.692 0.549
16) 0.015 0.124 0.628 0.762 -0.141 0.018 0.668 0.762 0.527 -0.121 0.726 0.605 0.263 -0.059 0.671 0.585
17) -0.015 0.110 0.644 0.776 -0.112 -0.006 0.692 0.746 0.461 -0.113 0.723 0.649 0.252 -0.070 0.696 0.603
18) 0.133 0.014 0.692 0.739 -0.084 -0.078 0.706 0.730 0.527 -0.092 0.714 0.640 0.351 -0.115 0.714 0.614
19) 0.310 -0.027 0.707 0.687 0.000 -0.073 0.700 0.681 0.571 0.006 0.678 0.646 0.450 -0.099 0.695 0.581
20) 0.280 0.065 0.675 0.701 0.112 -0.020 0.698 0.656 0.615 0.119 0.632 0.652 0.483 0.018 0.653 0.570
21) -0.074 -0.009 0.603 0.770 -0.281 0.054 0.570 0.804 0.132 -0.158 0.603 0.657 -0.088 -0.174 0.614 0.599
22) -0.133 -0.011 0.585 0.799 -0.253 -0.024 0.594 0.788 0.132 -0.153 0.607 0.693 -0.077 -0.177 0.621 0.625
23) -0.103 -0.038 0.607 0.785 -0.225 -0.057 0.607 0.771 0.198 -0.126 0.624 0.684 -0.055 -0.175 0.628 0.632
24) -0.133 0.062 0.575 0.799 -0.253 0.029 0.585 0.788 0.263 -0.017 0.598 0.675 -0.011 -0.096 0.630 0.646
25) -0.133 0.255 0.553 0.799 -0.197 0.214 0.587 0.794 0.351 0.120 0.613 0.687 0.066 0.092 0.595 0.650
26) -0.015 0.405 0.553 0.776 -0.169 0.376 0.548 0.778 0.439 0.213 0.596 0.663 0.198 0.303 0.540 0.650
27) -0.044 0.403 0.556 0.791 -0.169 0.371 0.552 0.778 0.373 0.252 0.571 0.707 0.176 0.308 0.547 0.686
28) 0.192 0.240 0.620 0.744 -0.112 0.226 0.575 0.746 0.373 0.237 0.553 0.707 0.230 0.183 0.582 0.683
29) 0.369 0.148 0.642 0.693 -0.028 0.184 0.575 0.698 0.373 0.288 0.551 0.707 0.252 0.138 0.579 0.646
30) 0.310 0.202 0.620 0.721 0.084 0.185 0.580 0.672 0.417 0.348 0.535 0.713 0.252 0.192 0.573 0.646
31) -0.044 -0.017 0.599 0.756 -0.169 0.006 0.603 0.739 0.154 -0.176 0.628 0.643 -0.099 -0.169 0.619 0.618
32) -0.103 0.079 0.543 0.785 -0.169 0.021 0.578 0.739 0.154 -0.085 0.592 0.678 -0.121 -0.074 0.563 0.654
33) -0.074 0.107 0.550 0.770 -0.197 0.095 0.532 0.755 0.220 0.006 0.574 0.669 -0.110 0.013 0.538 0.679
34) -0.103 0.207 0.521 0.785 -0.197 0.205 0.487 0.755 0.285 0.179 0.516 0.660 -0.066 0.161 0.518 0.694
35) -0.133 0.397 0.473 0.799 -0.197 0.449 0.447 0.794 0.351 0.327 0.495 0.687 0.022 0.347 0.471 0.679
36) -0.074 0.588 0.441 0.805 -0.197 0.645 0.395 0.794 0.329 0.463 0.446 0.701 0.055 0.617 0.376 0.712
37) -0.133 0.635 0.403 0.834 -0.253 0.675 0.369 0.826 0.176 0.552 0.385 0.769 -0.055 0.650 0.363 0.762
38) 0.103 0.480 0.467 0.788 -0.112 0.506 0.420 0.784 0.154 0.528 0.388 0.783 -0.011 0.447 0.412 0.733
39) 0.251 0.313 0.515 0.750 0.000 0.315 0.484 0.720 0.176 0.455 0.421 0.769 0.011 0.311 0.441 0.697
40) 0.192 0.262 0.526 0.779 0.141 0.183 0.562 0.678 0.241 0.389 0.444 0.760 0.011 0.218 0.496 0.697
41) -0.074 0.050 0.555 0.770 -0.169 0.030 0.543 0.739 0.044 -0.088 0.559 0.681 -0.198 -0.058 0.538 0.694
42) -0.162 0.183 0.477 0.814 -0.141 0.095 0.482 0.723 0.044 0.070 0.480 0.716 -0.230 0.081 0.455 0.748
43) -0.162 0.224 0.457 0.814 -0.197 0.188 0.417 0.755 0.088 0.176 0.430 0.722 -0.209 0.176 0.420 0.755
44) -0.192 0.329 0.406 0.828 -0.225 0.291 0.389 0.771 0.066 0.347 0.369 0.736 -0.198 0.310 0.389 0.780
45) -0.221 0.512 0.351 0.842 -0.309 0.519 0.323 0.820 -0.044 0.492 0.321 0.774 -0.198 0.463 0.340 0.780
46) -0.310 0.759 0.272 0.886 -0.338 0.745 0.257 0.836 -0.066 0.625 0.267 0.789 -0.230 0.763 0.221 0.835
47) -0.339 0.854 0.230 0.900 -0.366 0.816 0.212 0.852 -0.110 0.717 0.221 0.818 -0.296 0.821 0.209 0.857
48) -0.074 0.708 0.298 0.840 -0.253 0.663 0.270 0.826 -0.110 0.601 0.272 0.818 -0.209 0.610 0.265 0.799
49) 0.103 0.428 0.428 0.788 -0.169 0.392 0.394 0.778 -0.088 0.455 0.358 0.804 -0.176 0.370 0.381 0.744
50) 0.103 0.276 0.510 0.788 -0.028 0.194 0.519 0.736 -0.022 0.333 0.402 0.795 -0.176 0.200 0.456 0.744
51) -0.044 0.151 0.473 0.756 -0.084 0.061 0.497 0.730 -0.022 0.058 0.478 0.725 -0.252 0.046 0.462 0.741
52) -0.103 0.280 0.383 0.785 -0.056 0.136 0.452 0.714 -0.044 0.228 0.411 0.774 -0.285 0.192 0.394 0.795
53) -0.103 0.317 0.379 0.785 -0.112 0.236 0.417 0.746 -0.066 0.340 0.357 0.789 -0.263 0.294 0.358 0.802
54) -0.133 0.416 0.352 0.799 -0.141 0.354 0.372 0.762 -0.088 0.518 0.312 0.804 -0.307 0.437 0.311 0.831
55) -0.192 0.595 0.293 0.828 -0.225 0.582 0.296 0.810 -0.198 0.671 0.243 0.842 -0.340 0.580 0.260 0.842
56) -0.251 0.824 0.226 0.857 -0.309 0.859 0.199 0.858 -0.176 0.749 0.203 0.827 -0.362 0.841 0.172 0.878
57) -0.310 0.900 0.191 0.886 -0.338 0.900 0.168 0.874 -0.220 0.792 0.174 0.856 -0.428 0.900 0.140 0.900
58) -0.074 0.753 0.265 0.840 -0.253 0.766 0.225 0.865 -0.263 0.669 0.245 0.885 -0.362 0.664 0.209 0.835
59) 0.133 0.399 0.455 0.773 -0.281 0.476 0.382 0.842 -0.307 0.469 0.341 0.880 -0.329 0.351 0.365 0.780
60) 0.103 0.195 0.561 0.788 -0.112 0.240 0.508 0.784 -0.220 0.307 0.405 0.856 -0.318 0.089 0.487 0.762
61) -0.044 0.181 0.454 0.756 -0.141 0.157 0.461 0.762 -0.044 0.198 0.418 0.739 -0.209 0.109 0.431 0.712
62) -0.044 0.259 0.426 0.756 -0.112 0.220 0.445 0.746 -0.132 0.367 0.367 0.798 -0.263 0.232 0.394 0.759
63) -0.044 0.309 0.411 0.756 -0.169 0.320 0.414 0.778 -0.176 0.462 0.332 0.827 -0.274 0.320 0.364 0.777
64) -0.074 0.408 0.377 0.770 -0.197 0.387 0.372 0.794 -0.176 0.559 0.303 0.827 -0.318 0.438 0.321 0.806

65) -0.133 0.549 0.323 0.799 -0.253 0.522 0.319 0.826 -0.198 0.602 0.264 0.842 -0.318 0.518 0.283 0.806
66) -0.251 0.772 0.258 0.857 -0.338 0.753 0.249 0.874 -0.198 0.674 0.238 0.842 -0.329 0.682 0.225 0.824
67) -0.339 0.806 0.248 0.900 -0.338 0.789 0.236 0.874 -0.220 0.669 0.245 0.856 -0.428 0.702 0.211 0.857
68) -0.074 0.633 0.329 0.840 -0.225 0.651 0.294 0.849 -0.285 0.538 0.293 0.900 -0.362 0.467 0.270 0.791
69) 0.133 0.295 0.503 0.773 -0.281 0.388 0.433 0.842 -0.307 0.296 0.426 0.880 -0.296 0.176 0.408 0.726
70) 0.103 0.077 0.634 0.788 -0.112 0.142 0.576 0.784 -0.176 0.120 0.521 0.827 -0.241 -0.107 0.577 0.679
71) 0.044 0.232 0.490 0.747 -0.169 0.254 0.463 0.778 -0.154 0.338 0.396 0.777 -0.252 0.192 0.448 0.741
72) 0.044 0.261 0.489 0.747 0.000 0.211 0.503 0.720 -0.198 0.404 0.377 0.806 -0.307 0.265 0.425 0.788
73) 0.103 0.219 0.500 0.719 -0.028 0.233 0.493 0.736 -0.132 0.375 0.391 0.798 -0.263 0.225 0.433 0.759
74) 0.044 0.264 0.482 0.747 -0.028 0.203 0.486 0.736 -0.110 0.359 0.392 0.783 -0.274 0.273 0.406 0.777
75) -0.044 0.346 0.446 0.791 -0.084 0.245 0.477 0.768 -0.154 0.348 0.364 0.812 -0.307 0.284 0.389 0.788
76) -0.133 0.509 0.365 0.834 -0.141 0.383 0.428 0.800 -0.176 0.366 0.358 0.827 -0.329 0.352 0.362 0.824
77) -0.310 0.556 0.337 0.886 -0.197 0.422 0.412 0.833 -0.154 0.323 0.378 0.812 -0.428 0.366 0.319 0.857
78) -0.074 0.379 0.412 0.840 -0.056 0.307 0.430 0.791 -0.198 0.220 0.386 0.842 -0.307 0.153 0.382 0.744
79) 0.251 0.093 0.591 0.750 -0.028 0.112 0.572 0.775 -0.220 0.022 0.517 0.821 -0.198 -0.078 0.520 0.650
80) 0.221 -0.082 0.712 0.765 -0.056 -0.014 0.658 0.752 -0.154 -0.092 0.596 0.777 -0.176 -0.252 0.616 0.614
81) 0.103 0.111 0.578 0.719 -0.056 0.134 0.547 0.714 0.066 0.213 0.494 0.701 -0.099 0.079 0.549 0.661
82) 0.103 0.101 0.592 0.719 0.253 -0.003 0.646 0.614 0.066 0.174 0.502 0.701 -0.165 0.076 0.552 0.726
83) 0.133 0.048 0.595 0.704 0.197 -0.036 0.657 0.646 0.110 0.110 0.522 0.707 -0.154 0.001 0.563 0.708
84) 0.044 0.077 0.568 0.747 0.169 -0.085 0.660 0.662 0.044 0.073 0.516 0.716 -0.252 0.012 0.523 0.741
85) -0.044 0.132 0.540 0.791 0.112 -0.069 0.650 0.694 -0.066 0.059 0.490 0.754 -0.329 0.003 0.508 0.780
86) -0.133 0.249 0.471 0.834 0.028 0.016 0.606 0.742 -0.110 0.081 0.478 0.783 -0.351 0.037 0.470 0.817
87) -0.310 0.310 0.436 0.886 -0.056 0.113 0.554 0.791 -0.066 0.006 0.506 0.754 -0.428 0.055 0.442 0.857
88) -0.044 0.151 0.518 0.825 0.112 0.043 0.573 0.733 -0.132 -0.008 0.494 0.798 -0.307 -0.093 0.485 0.744
89) 0.369 -0.014 0.635 0.727 0.225 -0.025 0.629 0.707 -0.198 -0.117 0.571 0.806 -0.209 -0.213 0.584 0.668
90) 0.339 -0.063 0.711 0.742 0.056 -0.066 0.677 0.726 -0.154 -0.138 0.616 0.777 -0.187 -0.291 0.657 0.632
91) 0.103 -0.001 0.636 0.719 -0.056 -0.004 0.632 0.714 0.088 0.028 0.580 0.687 -0.033 -0.050 0.593 0.639
92) 0.133 -0.049 0.657 0.704 0.281 -0.172 0.746 0.598 0.088 -0.047 0.599 0.687 -0.121 -0.101 0.625 0.697
93) 0.133 -0.107 0.663 0.704 0.225 -0.217 0.755 0.630 0.110 -0.115 0.617 0.707 -0.154 -0.178 0.638 0.708
94) 0.044 -0.056 0.626 0.747 0.225 -0.255 0.761 0.630 0.022 -0.136 0.606 0.731 -0.296 -0.148 0.588 0.770
95) -0.044 0.027 0.586 0.791 0.141 -0.200 0.735 0.678 -0.088 -0.076 0.555 0.769 -0.329 -0.122 0.568 0.780
96) -0.103 0.129 0.525 0.819 0.056 -0.087 0.677 0.726 -0.132 0.011 0.525 0.798 -0.351 -0.057 0.524 0.817
97) -0.280 0.206 0.484 0.871 0.000 -0.016 0.642 0.759 -0.088 -0.058 0.552 0.769 -0.428 -0.017 0.490 0.857
98) -0.044 0.126 0.541 0.825 0.141 -0.036 0.637 0.717 -0.154 0.010 0.516 0.812 -0.318 -0.100 0.516 0.762
99) 0.369 0.056 0.612 0.727 0.338 -0.012 0.637 0.681 -0.220 -0.046 0.546 0.821 -0.252 -0.125 0.544 0.697
100) 0.339 0.063 0.663 0.742 0.112 0.054 0.631 0.733 -0.176 0.004 0.563 0.792 -0.209 -0.145 0.598 0.668
101) 0.082 -0.207 0.562 0.720 -0.398 -0.157 0.468 0.768 -0.332 -0.076 0.449 0.774 -0.106 -0.184 0.555 0.737
102) 0.082 -0.164 0.557 0.720 -0.384 -0.124 0.465 0.752 -0.261 -0.113 0.485 0.733 -0.010 -0.147 0.561 0.693
103) 0.035 -0.097 0.555 0.714 -0.339 -0.102 0.495 0.702 -0.166 -0.084 0.556 0.710 0.068 -0.075 0.571 0.666
104) 0.012 -0.067 0.547 0.729 -0.236 -0.130 0.537 0.666 -0.071 -0.065 0.596 0.688 0.087 -0.090 0.597 0.649
105) -0.058 -0.092 0.546 0.772 -0.236 -0.213 0.568 0.666 0.024 -0.097 0.590 0.666 0.203 -0.182 0.635 0.629
106) -0.269 -0.087 0.500 0.833 -0.280 -0.248 0.573 0.676 0.189 -0.322 0.702 0.601 0.126 -0.239 0.640 0.656
107) -0.175 -0.096 0.530 0.810 -0.192 -0.206 0.558 0.656 0.166 -0.328 0.687 0.615 0.242 -0.307 0.675 0.595
108) 0.058 -0.150 0.596 0.769 0.118 -0.310 0.662 0.547 0.284 -0.387 0.770 0.579 0.242 -0.309 0.677 0.595
109) 0.129 -0.215 0.647 0.726 0.251 -0.403 0.732 0.478 0.379 -0.425 0.810 0.557 0.261 -0.385 0.742 0.578
110) 0.316 -0.351 0.734 0.645 0.310 -0.447 0.760 0.452 0.403 -0.472 0.838 0.543 0.435 -0.437 0.791 0.507
111) 0.082 -0.147 0.522 0.720 -0.295 -0.163 0.479 0.692 -0.332 -0.039 0.426 0.774 -0.203 -0.157 0.511 0.741
112) 0.105 -0.089 0.521 0.705 -0.280 -0.111 0.481 0.676 -0.261 -0.040 0.455 0.733 -0.106 -0.082 0.516 0.697
113) 0.035 -0.015 0.517 0.714 -0.310 -0.023 0.470 0.669 -0.166 0.007 0.494 0.710 -0.106 -0.002 0.514 0.697
114) 0.012 -0.030 0.549 0.729 -0.221 -0.045 0.504 0.649 -0.095 0.029 0.541 0.702 -0.106 -0.018 0.553 0.697
115) -0.012 -0.114 0.583 0.743 -0.207 -0.185 0.566 0.633 -0.047 0.030 0.545 0.707 -0.010 -0.088 0.603 0.693
116) -0.222 -0.080 0.531 0.804 -0.251 -0.214 0.567 0.643 0.142 -0.232 0.670 0.629 -0.029 -0.110 0.597 0.710
117) -0.199 -0.066 0.523 0.790 -0.221 -0.169 0.545 0.649 0.118 -0.240 0.653 0.643 0.068 -0.178 0.623 0.666
118) 0.082 -0.125 0.609 0.755 0.030 -0.272 0.631 0.567 0.261 -0.333 0.739 0.593 0.126 -0.172 0.589 0.656
119) 0.175 -0.190 0.656 0.697 0.133 -0.352 0.677 0.531 0.379 -0.373 0.783 0.557 0.300 -0.273 0.677 0.585
120) 0.339 -0.323 0.735 0.630 0.148 -0.407 0.716 0.514 0.379 -0.421 0.801 0.557 0.474 -0.396 0.763 0.514
121) -0.012 0.015 0.435 0.743 -0.339 -0.044 0.397 0.702 -0.284 0.027 0.375 0.747 -0.223 -0.023 0.416 0.717
122) -0.035 0.193 0.399 0.758 -0.325 0.100 0.382 0.686 -0.237 0.119 0.388 0.719 -0.203 0.122 0.380 0.700
123) -0.199 0.303 0.359 0.790 -0.354 0.166 0.360 0.679 -0.142 0.200 0.404 0.696 -0.300 0.263 0.338 0.744
124) -0.222 0.287 0.391 0.804 -0.251 0.091 0.414 0.643 -0.047 0.183 0.469 0.674 -0.281 0.191 0.395 0.727
125) -0.222 0.170 0.440 0.804 -0.221 -0.066 0.488 0.610 0.000 0.172 0.490 0.680 -0.145 0.055 0.503 0.690
126) -0.386 0.244 0.385 0.871 -0.236 -0.068 0.487 0.626 0.166 -0.087 0.596 0.615 -0.106 0.053 0.517 0.697
127) -0.386 0.258 0.379 0.871 -0.266 -0.023 0.477 0.659 0.118 -0.105 0.589 0.643 -0.010 -0.013 0.539 0.653
128) -0.152 0.138 0.475 0.830 -0.089 -0.135 0.545 0.580 0.261 -0.252 0.707 0.593 0.068 -0.074 0.567 0.625
129) -0.035 0.017 0.528 0.758 0.059 -0.228 0.604 0.534 0.426 -0.295 0.760 0.562 0.223 -0.180 0.623 0.571
130) 0.129 -0.194 0.613 0.691 0.089 -0.371 0.673 0.501 0.426 -0.366 0.761 0.562 0.455 -0.332 0.692 0.490

131) -0.105 0.248 0.311 0.766 -0.384 0.166 0.282 0.712 -0.308 0.161 0.302 0.760 -0.242 0.203 0.291 0.734
132) -0.152 0.498 0.249 0.795 -0.369 0.388 0.213 0.696 -0.284 0.389 0.253 0.747 -0.261 0.453 0.224 0.751
133) -0.362 0.692 0.179 0.856 -0.413 0.469 0.154 0.705 -0.189 0.501 0.239 0.724 -0.397 0.695 0.152 0.829
134) -0.386 0.748 0.191 0.871 -0.325 0.440 0.209 0.686 -0.118 0.547 0.276 0.716 -0.377 0.686 0.182 0.812
135) -0.386 0.666 0.232 0.871 -0.310 0.365 0.259 0.669 -0.071 0.600 0.274 0.721 -0.223 0.521 0.284 0.758
136) -0.432 0.671 0.231 0.900 -0.280 0.418 0.254 0.676 0.118 0.273 0.384 0.643 -0.145 0.468 0.304 0.731
137) -0.362 0.611 0.273 0.891 -0.310 0.420 0.256 0.709 0.166 0.240 0.382 0.649 0.106 0.370 0.350 0.673
138) -0.175 0.401 0.384 0.845 -0.177 0.213 0.353 0.640 0.237 0.001 0.549 0.607 0.203 0.221 0.426 0.629
139) -0.058 0.157 0.479 0.772 -0.074 0.022 0.479 0.603 0.403 -0.106 0.649 0.576 0.358 0.010 0.541 0.575
140) 0.105 -0.125 0.583 0.705 -0.044 -0.228 0.597 0.570 0.403 -0.249 0.691 0.576 0.532 -0.231 0.653 0.503
141) -0.082 0.271 0.257 0.752 -0.398 0.239 0.217 0.729 -0.284 0.214 0.240 0.747 -0.126 0.225 0.264 0.673
142) -0.152 0.588 0.166 0.795 -0.398 0.474 0.144 0.729 -0.284 0.474 0.166 0.747 -0.184 0.537 0.167 0.724
143) -0.362 0.802 0.064 0.856 -0.502 0.635 0.047 0.765 -0.284 0.604 0.117 0.747 -0.319 0.806 0.089 0.802
144) -0.362 0.866 0.065 0.856 -0.457 0.699 0.038 0.755 -0.213 0.726 0.101 0.738 -0.319 0.900 0.057 0.802
145) -0.362 0.857 0.083 0.856 -0.457 0.700 0.037 0.755 -0.261 0.862 0.055 0.766 -0.300 0.832 0.071 0.785
146) -0.362 0.858 0.088 0.856 -0.487 0.774 0.018 0.788 -0.166 0.561 0.169 0.710 -0.242 0.848 0.073 0.775
147) -0.292 0.797 0.140 0.848 -0.516 0.776 0.038 0.821 -0.142 0.506 0.212 0.730 -0.010 0.716 0.151 0.734
148) -0.105 0.558 0.284 0.801 -0.384 0.511 0.215 0.752 -0.071 0.224 0.427 0.688 0.087 0.457 0.308 0.690
149) -0.012 0.224 0.444 0.743 -0.266 0.216 0.387 0.699 0.047 0.052 0.544 0.652 0.261 0.144 0.499 0.619
150) 0.129 -0.124 0.581 0.691 -0.221 -0.114 0.525 0.649 0.095 -0.157 0.614 0.624 0.455 -0.174 0.635 0.531
151) 0.058 0.206 0.330 0.700 -0.339 0.147 0.263 0.663 -0.308 0.250 0.255 0.760 -0.068 0.176 0.308 0.663
152) -0.012 0.500 0.249 0.743 -0.369 0.406 0.188 0.696 -0.308 0.464 0.198 0.760 -0.126 0.465 0.217 0.714
153) -0.199 0.693 0.167 0.790 -0.487 0.627 0.075 0.748 -0.284 0.573 0.179 0.747 -0.203 0.697 0.167 0.781
154) -0.199 0.758 0.152 0.790 -0.443 0.691 0.067 0.738 -0.308 0.771 0.108 0.760 -0.242 0.844 0.132 0.815
155) -0.292 0.829 0.114 0.813 -0.472 0.752 0.056 0.771 -0.355 0.900 0.078 0.788 -0.300 0.860 0.097 0.825
156) -0.316 0.900 0.111 0.827 -0.531 0.900 0.014 0.837 -0.308 0.637 0.177 0.760 -0.223 0.858 0.129 0.798
157) -0.199 0.826 0.180 0.790 -0.561 0.890 0.054 0.870 -0.284 0.571 0.234 0.780 0.029 0.703 0.237 0.741
158) -0.058 0.573 0.329 0.772 -0.502 0.623 0.223 0.844 -0.213 0.292 0.414 0.738 0.087 0.433 0.380 0.731
159) 0.012 0.196 0.470 0.729 -0.398 0.263 0.377 0.808 -0.095 0.073 0.536 0.702 0.261 0.082 0.555 0.659
160) 0.105 -0.125 0.581 0.705 -0.354 -0.108 0.513 0.758 -0.047 -0.159 0.594 0.674 0.377 -0.219 0.669 0.598
161) 0.082 0.084 0.423 0.685 -0.266 0.019 0.358 0.620 -0.189 0.054 0.373 0.691 0.029 0.015 0.428 0.619
162) 0.012 0.298 0.362 0.729 -0.280 0.194 0.313 0.636 -0.213 0.188 0.338 0.705 0.010 0.165 0.370 0.636
163) -0.152 0.409 0.300 0.761 -0.384 0.326 0.228 0.673 -0.213 0.273 0.326 0.705 -0.068 0.324 0.346 0.703
164) -0.175 0.463 0.286 0.775 -0.325 0.316 0.230 0.646 -0.213 0.386 0.284 0.705 -0.126 0.432 0.328 0.754
165) -0.292 0.530 0.243 0.813 -0.369 0.361 0.239 0.696 -0.284 0.486 0.260 0.747 -0.242 0.448 0.293 0.775
166) -0.269 0.534 0.239 0.798 -0.428 0.421 0.224 0.762 -0.213 0.271 0.333 0.705 -0.145 0.452 0.298 0.731
167) -0.129 0.410 0.327 0.746 -0.398 0.349 0.298 0.768 -0.189 0.211 0.415 0.724 0.068 0.309 0.416 0.707
168) -0.082 0.214 0.433 0.752 -0.398 0.159 0.396 0.768 -0.166 -0.008 0.542 0.710 0.126 0.092 0.530 0.697
169) -0.012 -0.070 0.536 0.708 -0.369 -0.052 0.489 0.775 -0.071 -0.168 0.634 0.688 0.300 -0.171 0.655 0.625
170) -0.012 -0.248 0.589 0.708 -0.339 -0.270 0.572 0.742 -0.071 -0.285 0.643 0.688 0.377 -0.344 0.724 0.598
171) 0.105 -0.050 0.485 0.671 -0.177 -0.135 0.443 0.560 -0.166 -0.087 0.461 0.677 0.029 -0.087 0.487 0.619
172) 0.035 0.094 0.440 0.714 -0.192 -0.040 0.423 0.577 -0.189 -0.012 0.443 0.691 -0.010 -0.031 0.462 0.653
173) -0.129 0.150 0.386 0.746 -0.266 -0.008 0.359 0.580 -0.142 -0.022 0.456 0.663 -0.068 -0.000 0.467 0.703
174) -0.129 0.149 0.383 0.746 -0.192 -0.049 0.362 0.537 -0.118 -0.013 0.449 0.649 -0.126 0.033 0.461 0.754
175) -0.269 0.208 0.337 0.798 -0.295 -0.025 0.377 0.613 -0.213 0.063 0.421 0.705 -0.242 0.043 0.427 0.775
176) -0.245 0.201 0.341 0.784 -0.354 0.009 0.372 0.679 -0.166 -0.055 0.465 0.677 -0.184 0.078 0.429 0.764
177) -0.129 0.097 0.428 0.746 -0.325 -0.051 0.453 0.686 -0.166 -0.060 0.525 0.710 0.029 0.015 0.522 0.741
178) -0.058 -0.053 0.522 0.737 -0.325 -0.156 0.528 0.686 -0.095 -0.237 0.648 0.668 0.106 -0.152 0.621 0.714
179) -0.035 -0.212 0.583 0.723 -0.310 -0.258 0.576 0.709 0.000 -0.322 0.711 0.646 0.281 -0.315 0.719 0.642
180) -0.012 -0.296 0.606 0.708 -0.266 -0.379 0.629 0.659 0.071 -0.377 0.708 0.638 0.416 -0.402 0.755 0.605
181) 0.082 -0.113 0.497 0.650 -0.148 -0.178 0.457 0.527 0.024 -0.176 0.531 0.599 0.184 -0.165 0.536 0.564
182) -0.012 0.022 0.449 0.708 -0.148 -0.154 0.458 0.527 0.000 -0.126 0.517 0.613 0.087 -0.127 0.503 0.608
183) -0.082 0.062 0.414 0.717 -0.162 -0.159 0.420 0.504 0.047 -0.146 0.524 0.585 0.106 -0.129 0.523 0.632
184) -0.058 0.018 0.426 0.703 -0.030 -0.220 0.435 0.435 0.071 -0.164 0.526 0.571 0.048 -0.144 0.527 0.683
185) -0.199 0.035 0.393 0.755 -0.118 -0.241 0.469 0.495 0.024 -0.167 0.521 0.599 -0.048 -0.176 0.518 0.686
186) -0.175 0.030 0.394 0.740 -0.192 -0.191 0.455 0.577 0.047 -0.219 0.543 0.585 -0.010 -0.162 0.519 0.693
187) -0.058 -0.024 0.467 0.703 -0.207 -0.237 0.528 0.593 0.071 -0.245 0.621 0.604 0.223 -0.189 0.602 0.653
188) -0.012 -0.123 0.546 0.708 -0.207 -0.301 0.594 0.593 0.118 -0.359 0.723 0.576 0.242 -0.311 0.688 0.636
189) -0.012 -0.259 0.601 0.708 -0.177 -0.389 0.653 0.600 0.142 -0.415 0.756 0.562 0.281 -0.385 0.740 0.602
190) -0.012 -0.317 0.612 0.708 -0.148 -0.446 0.681 0.567 0.189 -0.434 0.743 0.568 0.397 -0.414 0.753 0.581
191) -0.105 0.063 0.413 0.732 -0.251 -0.074 0.406 0.603 0.000 -0.097 0.507 0.613 0.126 -0.059 0.473 0.615
192) -0.199 0.250 0.353 0.790 -0.266 0.023 0.368 0.620 -0.047 -0.014 0.469 0.640 0.010 0.015 0.436 0.676
193) -0.175 0.295 0.341 0.775 -0.221 0.001 0.360 0.570 -0.024 -0.005 0.462 0.627 0.029 0.089 0.428 0.700
194) -0.152 0.203 0.362 0.761 -0.089 -0.073 0.372 0.501 0.000 -0.026 0.454 0.613 -0.029 0.072 0.423 0.751
195) -0.269 0.142 0.348 0.798 -0.192 -0.093 0.404 0.577 -0.024 -0.089 0.467 0.627 -0.106 -0.037 0.435 0.737
196) -0.222 0.095 0.357 0.769 -0.251 -0.053 0.385 0.643 -0.024 -0.149 0.491 0.627 -0.029 -0.098 0.466 0.710
197) -0.082 -0.002 0.440 0.717 -0.266 -0.114 0.461 0.659 0.024 -0.199 0.567 0.632 0.223 -0.164 0.561 0.653
198) -0.012 -0.139 0.535 0.708 -0.251 -0.216 0.536 0.643 0.024 -0.289 0.652 0.632 0.165 -0.279 0.639 0.663
199) 0.012 -0.281 0.603 0.694 -0.207 -0.334 0.595 0.633 0.047 -0.367 0.685 0.618 0.145 -0.361 0.687 0.639
200) 0.035 -0.314 0.606 0.679 -0.162 -0.415 0.635 0.584 0.118 -0.408 0.698 0.610 0.261 -0.384 0.715 0.619

B

Urban area

1) -0.387 -0.376 0.582 0.376 0.133 -0.495 0.609 -0.116 0.108 -0.441 0.671 0.201 -0.451 -0.464 0.585 0.363
2) -0.138 -0.422 0.612 0.253 0.305 -0.523 0.648 -0.195 0.069 -0.449 0.674 0.236 -0.393 -0.495 0.621 0.283
3) -0.121 -0.407 0.601 0.237 0.165 -0.501 0.628 -0.009 -0.101 -0.390 0.620 0.366 -0.426 -0.421 0.568 0.262
4) -0.178 -0.350 0.554 0.315 0.063 -0.452 0.577 0.086 -0.160 -0.355 0.579 0.419 -0.472 -0.405 0.523 0.308
5) -0.206 -0.330 0.560 0.387 0.142 -0.436 0.551 0.052 0.056 -0.328 0.562 0.330 -0.433 -0.415 0.542 0.247
6) -0.003 -0.348 0.583 0.403 0.049 -0.461 0.600 0.096 0.167 -0.376 0.608 0.246 -0.181 -0.448 0.603 0.133
7) 0.087 -0.442 0.667 0.350 -0.188 -0.408 0.597 0.356 0.069 -0.347 0.594 0.334 -0.056 -0.498 0.638 0.102
8) -0.110 -0.378 0.616 0.482 -0.477 -0.305 0.514 0.550 -0.075 -0.290 0.583 0.424 -0.009 -0.522 0.647 0.119
9) -0.037 -0.376 0.611 0.446 -0.491 -0.212 0.443 0.568 -0.128 -0.253 0.541 0.519 0.027 -0.448 0.601 0.181
10) -0.087 -0.148 0.494 0.563 -0.533 -0.056 0.401 0.624 -0.121 -0.089 0.474 0.489 0.013 -0.255 0.499 0.238
11) -0.381 -0.424 0.627 0.348 0.258 -0.514 0.639 -0.152 0.160 -0.408 0.654 0.178 -0.584 -0.415 0.510 0.506
12) -0.223 -0.381 0.564 0.278 0.240 -0.492 0.620 -0.114 0.043 -0.376 0.612 0.283 -0.494 -0.427 0.529 0.437
13) -0.229 -0.322 0.523 0.306 0.067 -0.464 0.590 0.075 -0.121 -0.309 0.561 0.383 -0.519 -0.286 0.439 0.429
14) -0.285 -0.230 0.448 0.384 -0.119 -0.377 0.526 0.217 -0.180 -0.263 0.500 0.436 -0.483 -0.283 0.426 0.415
15) -0.313 -0.210 0.476 0.456 -0.012 -0.352 0.516 0.211 0.029 -0.218 0.490 0.377 -0.440 -0.301 0.486 0.319
16) -0.195 -0.240 0.504 0.447 -0.165 -0.349 0.551 0.295 0.193 -0.304 0.553 0.269 -0.275 -0.372 0.540 0.188
17) -0.116 -0.361 0.588 0.451 -0.356 -0.315 0.560 0.468 0.147 -0.306 0.558 0.334 -0.113 -0.415 0.592 0.057
18) -0.223 -0.345 0.563 0.530 -0.491 -0.265 0.501 0.577 -0.049 -0.293 0.582 0.448 -0.142 -0.519 0.654 0.086
19) -0.217 -0.363 0.587 0.502 -0.491 -0.234 0.490 0.577 -0.069 -0.309 0.583 0.466 -0.153 -0.479 0.622 0.194
20) -0.268 -0.151 0.479 0.619 -0.519 -0.078 0.428 0.614 -0.049 -0.191 0.556 0.448 -0.181 -0.313 0.552 0.222
21) -0.403 -0.396 0.583 0.326 0.202 -0.501 0.620 -0.032 0.245 -0.435 0.685 0.007 -0.573 -0.387 0.508 0.484
22) -0.262 -0.343 0.534 0.272 0.202 -0.458 0.575 -0.043 0.101 -0.379 0.616 0.160 -0.516 -0.353 0.468 0.480
23) -0.268 -0.274 0.469 0.300 0.049 -0.402 0.538 0.171 -0.029 -0.312 0.548 0.254 -0.548 -0.224 0.393 0.458
24) -0.353 -0.253 0.461 0.384 -0.091 -0.327 0.474 0.225 -0.088 -0.309 0.527 0.307 -0.480 -0.273 0.417 0.379
25) -0.375 -0.256 0.504 0.428 -0.058 -0.327 0.511 0.223 -0.043 -0.263 0.531 0.356 -0.462 -0.286 0.457 0.286
26) -0.262 -0.302 0.548 0.398 -0.151 -0.305 0.536 0.276 0.095 -0.352 0.593 0.295 -0.286 -0.393 0.582 0.134
27) -0.183 -0.350 0.588 0.402 -0.351 -0.268 0.537 0.437 -0.056 -0.352 0.607 0.415 -0.217 -0.424 0.619 0.108
28) -0.245 -0.381 0.613 0.458 -0.444 -0.258 0.537 0.490 -0.259 -0.325 0.627 0.559 -0.278 -0.534 0.689 0.138
29) -0.251 -0.409 0.632 0.486 -0.421 -0.383 0.575 0.409 -0.259 -0.368 0.641 0.559 -0.286 -0.534 0.697 0.210
30) -0.229 -0.199 0.504 0.568 -0.440 -0.237 0.520 0.459 -0.219 -0.293 0.637 0.523 -0.318 -0.409 0.602 0.274
31) -0.494 -0.350 0.565 0.371 0.207 -0.533 0.660 -0.008 0.383 -0.459 0.705 -0.054 -0.433 -0.479 0.616 0.419
32) -0.364 -0.332 0.535 0.373 0.188 -0.473 0.586 0.030 0.141 -0.408 0.651 0.125 -0.465 -0.418 0.541 0.484
33) -0.364 -0.276 0.470 0.373 0.030 -0.377 0.520 0.275 -0.043 -0.325 0.556 0.242 -0.516 -0.307 0.470 0.470
34) -0.375 -0.271 0.468 0.428 -0.165 -0.346 0.514 0.350 -0.101 -0.339 0.561 0.295 -0.390 -0.372 0.509 0.323
35) -0.403 -0.286 0.516 0.501 -0.165 -0.364 0.568 0.350 -0.003 -0.325 0.576 0.321 -0.397 -0.359 0.517 0.233
36) -0.268 -0.376 0.598 0.426 -0.240 -0.333 0.572 0.353 0.023 -0.403 0.641 0.344 -0.221 -0.442 0.600 0.081
37) -0.229 -0.404 0.630 0.432 -0.398 -0.262 0.504 0.458 -0.088 -0.414 0.658 0.429 -0.207 -0.470 0.647 0.023
38) -0.268 -0.432 0.662 0.444 -0.472 -0.311 0.536 0.462 -0.265 -0.422 0.665 0.526 -0.210 -0.534 0.671 -0.014
39) -0.274 -0.460 0.681 0.472 -0.449 -0.405 0.550 0.381 -0.265 -0.390 0.652 0.526 -0.232 -0.599 0.736 0.115
40) -0.245 -0.325 0.595 0.525 -0.412 -0.305 0.502 0.356 -0.226 -0.368 0.647 0.490 -0.264 -0.510 0.654 0.179
41) -0.505 -0.309 0.536 0.426 0.100 -0.520 0.678 0.119 0.219 -0.395 0.664 0.054 -0.487 -0.482 0.619 0.441
42) -0.375 -0.251 0.491 0.428 -0.035 -0.420 0.569 0.217 0.003 -0.341 0.603 0.186 -0.508 -0.433 0.569 0.484
43) -0.370 -0.182 0.419 0.401 -0.174 -0.290 0.478 0.412 -0.187 -0.271 0.525 0.333 -0.573 -0.319 0.467 0.527
44) -0.296 -0.192 0.444 0.432 -0.328 -0.305 0.508 0.430 -0.285 -0.296 0.556 0.421 -0.397 -0.430 0.555 0.385
45) -0.375 -0.245 0.509 0.554 -0.435 -0.346 0.554 0.428 -0.344 -0.314 0.567 0.474 -0.419 -0.415 0.573 0.352
46) -0.285 -0.350 0.582 0.502 -0.533 -0.346 0.547 0.513 -0.317 -0.403 0.623 0.498 -0.268 -0.510 0.646 0.203
47) -0.240 -0.453 0.671 0.480 -0.598 -0.330 0.514 0.509 -0.370 -0.433 0.653 0.529 -0.214 -0.544 0.699 0.095
48) -0.274 -0.499 0.721 0.464 -0.514 -0.405 0.561 0.397 -0.389 -0.465 0.706 0.547 -0.214 -0.605 0.748 0.022
49) -0.285 -0.509 0.734 0.520 -0.477 -0.467 0.592 0.298 -0.331 -0.419 0.680 0.494 -0.181 -0.627 0.786 0.119
50) -0.341 -0.409 0.653 0.598 -0.463 -0.461 0.599 0.279 -0.291 -0.419 0.684 0.459 -0.239 -0.544 0.691 0.187
51) -0.522 -0.261 0.550 0.443 -0.081 -0.355 0.611 0.249 0.056 -0.271 0.653 0.162 -0.508 -0.430 0.642 0.484
52) -0.528 -0.187 0.476 0.471 -0.226 -0.308 0.545 0.335 -0.154 -0.210 0.572 0.265 -0.620 -0.316 0.549 0.596
53) -0.539 -0.087 0.388 0.460 -0.365 -0.134 0.421 0.530 -0.344 -0.210 0.515 0.412 -0.699 -0.194 0.405 0.696
54) -0.409 -0.113 0.391 0.413 -0.444 -0.202 0.477 0.490 -0.435 -0.210 0.521 0.471 -0.490 -0.283 0.482 0.490
55) -0.499 -0.210 0.495 0.524 -0.560 -0.296 0.502 0.475 -0.527 -0.191 0.502 0.600 -0.512 -0.366 0.548 0.457
56) -0.358 -0.378 0.575 0.422 -0.537 -0.396 0.534 0.469 -0.475 -0.341 0.577 0.576 -0.357 -0.525 0.651 0.271
57) -0.296 -0.506 0.694 0.383 -0.523 -0.424 0.562 0.376 -0.422 -0.416 0.639 0.553 -0.264 -0.571 0.684 0.114
58) -0.195 -0.547 0.767 0.341 -0.421 -0.476 0.632 0.215 -0.389 -0.476 0.715 0.547 -0.178 -0.620 0.754 0.007
59) -0.200 -0.539 0.763 0.369 -0.379 -0.511 0.667 0.084 -0.331 -0.414 0.673 0.494 -0.142 -0.627 0.779 0.069
60) -0.313 -0.414 0.652 0.525 -0.444 -0.480 0.622 0.155 -0.298 -0.330 0.598 0.488 -0.232 -0.577 0.712 0.201
61) -0.466 -0.228 0.545 0.432 -0.174 -0.355 0.568 0.208 -0.108 -0.193 0.630 0.333 -0.523 -0.261 0.565 0.627
62) -0.488 -0.169 0.480 0.476 -0.244 -0.355 0.552 0.236 -0.245 -0.164 0.572 0.394 -0.649 -0.237 0.502 0.711

63) -0.499 -0.176 0.457 0.465 -0.356 -0.265 0.496 0.393 -0.376 -0.220 0.532 0.488 -0.713 -0.203 0.425 0.754
64) -0.364 -0.215 0.480 0.391 -0.412 -0.283 0.532 0.347 -0.468 -0.215 0.537 0.547 -0.512 -0.283 0.503 0.533
65) -0.505 -0.299 0.553 0.552 -0.509 -0.336 0.529 0.357 -0.520 -0.150 0.520 0.641 -0.569 -0.366 0.569 0.514
66) -0.336 -0.453 0.636 0.377 -0.472 -0.442 0.577 0.333 -0.494 -0.271 0.563 0.665 -0.426 -0.488 0.643 0.350
67) -0.206 -0.534 0.730 0.330 -0.421 -0.464 0.587 0.215 -0.403 -0.341 0.625 0.606 -0.289 -0.577 0.703 0.182
68) -0.087 -0.550 0.769 0.272 -0.286 -0.489 0.641 0.060 -0.291 -0.430 0.681 0.529 -0.163 -0.608 0.754 0.026
69) -0.042 -0.539 0.741 0.250 -0.179 -0.508 0.674 -0.067 -0.245 -0.392 0.660 0.465 -0.128 -0.577 0.702 0.088
70) -0.155 -0.455 0.653 0.406 -0.281 -0.501 0.640 0.028 -0.213 -0.244 0.580 0.459 -0.207 -0.498 0.616 0.198
71) -0.437 -0.235 0.564 0.426 -0.333 -0.277 0.457 0.238 -0.147 -0.156 0.588 0.430 -0.501 -0.215 0.494 0.584
72) -0.409 -0.215 0.527 0.421 -0.328 -0.305 0.465 0.207 -0.200 -0.142 0.557 0.392 -0.584 -0.206 0.466 0.582
73) -0.432 -0.240 0.542 0.398 -0.328 -0.249 0.460 0.282 -0.180 -0.210 0.574 0.436 -0.609 -0.233 0.481 0.574
74) -0.291 -0.266 0.560 0.296 -0.277 -0.274 0.516 0.183 -0.219 -0.247 0.614 0.472 -0.426 -0.301 0.533 0.360
75) -0.364 -0.366 0.624 0.450 -0.347 -0.318 0.516 0.156 -0.193 -0.218 0.616 0.495 -0.458 -0.412 0.610 0.349
76) -0.059 -0.478 0.682 0.249 -0.263 -0.427 0.582 0.099 -0.324 -0.293 0.614 0.598 -0.361 -0.522 0.672 0.221
77) 0.093 -0.557 0.774 0.157 -0.184 -0.483 0.632 0.010 -0.317 -0.314 0.627 0.568 -0.257 -0.571 0.702 0.118
78) 0.166 -0.583 0.810 0.122 -0.077 -0.523 0.668 -0.108 -0.180 -0.411 0.675 0.445 -0.149 -0.590 0.727 -0.031
79) 0.223 -0.575 0.788 0.110 -0.026 -0.551 0.700 -0.160 -0.160 -0.419 0.694 0.427 -0.084 -0.531 0.667 -0.009
80) 0.116 -0.493 0.701 0.239 -0.170 -0.517 0.662 -0.009 -0.245 -0.185 0.585 0.465 -0.160 -0.448 0.583 0.152
81) -0.477 0.064 0.402 0.488 -0.430 -0.212 0.386 0.248 -0.245 -0.035 0.470 0.519 -0.508 -0.178 0.441 0.570
82) -0.420 0.033 0.384 0.410 -0.374 -0.249 0.426 0.239 -0.278 -0.048 0.469 0.462 -0.562 -0.175 0.432 0.538
83) -0.398 -0.023 0.444 0.365 -0.333 -0.215 0.460 0.258 -0.245 -0.102 0.505 0.519 -0.580 -0.203 0.469 0.546
84) -0.262 -0.118 0.496 0.290 -0.267 -0.265 0.517 0.141 -0.239 -0.196 0.576 0.489 -0.429 -0.316 0.568 0.396
85) -0.319 -0.248 0.549 0.427 -0.100 -0.321 0.537 0.081 0.075 -0.250 0.613 0.432 -0.447 -0.424 0.635 0.328
86) -0.065 -0.399 0.617 0.277 -0.040 -0.448 0.619 0.030 -0.062 -0.296 0.594 0.564 -0.347 -0.498 0.642 0.227
87) 0.099 -0.478 0.686 0.129 0.067 -0.517 0.664 -0.097 -0.016 -0.336 0.614 0.499 -0.250 -0.531 0.647 0.109
88) 0.127 -0.483 0.718 0.183 0.142 -0.545 0.707 -0.212 0.173 -0.433 0.686 0.352 -0.167 -0.568 0.703 -0.037
89) 0.274 -0.506 0.741 0.168 0.151 -0.533 0.703 -0.208 0.272 -0.462 0.738 0.326 -0.092 -0.553 0.689 -0.036
90) 0.189 -0.478 0.708 0.252 0.016 -0.486 0.630 -0.044 0.272 -0.231 0.654 0.326 -0.077 -0.498 0.619 0.056
91) -0.578 0.008 0.366 0.588 -0.560 -0.153 0.336 0.326 -0.475 0.081 0.373 0.639 -0.659 -0.129 0.370 0.570
92) -0.415 -0.066 0.423 0.441 -0.453 -0.209 0.406 0.274 -0.409 -0.013 0.445 0.556 -0.595 -0.258 0.438 0.441
93) -0.370 -0.136 0.506 0.352 -0.374 -0.274 0.494 0.194 -0.331 -0.097 0.505 0.619 -0.566 -0.292 0.491 0.412
94) -0.206 -0.225 0.551 0.271 -0.202 -0.389 0.603 0.079 -0.324 -0.234 0.584 0.589 -0.472 -0.402 0.583 0.331
95) -0.223 -0.291 0.568 0.414 -0.077 -0.448 0.649 0.075 -0.016 -0.220 0.608 0.561 -0.462 -0.433 0.602 0.309
96) -0.025 -0.386 0.612 0.282 -0.035 -0.467 0.634 0.073 -0.128 -0.247 0.561 0.646 -0.390 -0.504 0.661 0.247
97) 0.178 -0.432 0.643 0.141 0.114 -0.526 0.673 -0.109 -0.101 -0.250 0.561 0.599 -0.325 -0.485 0.617 0.194
98) 0.031 -0.353 0.641 0.271 0.133 -0.495 0.665 -0.150 0.082 -0.258 0.565 0.482 -0.332 -0.448 0.616 0.180
99) 0.364 -0.394 0.707 0.219 0.240 -0.489 0.662 -0.156 0.180 -0.282 0.646 0.456 -0.149 -0.482 0.654 0.154
100) 0.217 -0.445 0.694 0.283 0.067 -0.480 0.617 0.032 0.193 -0.234 0.643 0.397 -0.120 -0.498 0.620 0.189
101) 0.161 -0.446 0.819 0.117 0.013 -0.260 0.738 0.183 -0.457 0.287 0.587 0.707 -0.400 -0.145 0.725 0.363
102) 0.344 -0.498 0.858 0.045 0.281 -0.380 0.830 0.008 -0.348 -0.262 0.734 0.554 -0.384 -0.215 0.759 0.339
103) 0.606 -0.479 0.827 0.019 0.568 -0.393 0.837 -0.119 -0.145 -0.278 0.741 0.432 -0.293 -0.194 0.759 0.390
104) 0.433 -0.413 0.797 0.169 0.447 -0.387 0.821 -0.022 -0.102 -0.291 0.745 0.398 -0.298 -0.109 0.713 0.430
105) 0.567 -0.388 0.773 0.151 0.538 -0.380 0.804 -0.071 0.022 -0.307 0.739 0.316 -0.148 -0.046 0.704 0.366
106) 0.422 -0.366 0.728 0.221 0.417 -0.336 0.795 0.109 0.160 -0.250 0.702 0.246 -0.013 0.088 0.618 0.385
107) 0.156 -0.333 0.720 0.395 0.217 -0.273 0.739 0.205 0.276 -0.254 0.700 0.192 0.094 0.018 0.670 0.331
108) 0.117 -0.432 0.825 0.332 0.089 -0.304 0.766 0.202 0.203 -0.287 0.707 0.272 0.218 -0.138 0.772 0.264
109) -0.178 -0.344 0.752 0.395 -0.221 -0.222 0.721 0.378 0.109 -0.198 0.667 0.368 0.121 -0.067 0.727 0.253
110) -0.044 -0.271 0.700 0.378 -0.175 -0.177 0.692 0.439 0.138 -0.194 0.659 0.323 0.239 -0.074 0.699 0.285
111) 0.033 -0.326 0.717 0.160 -0.096 -0.222 0.706 0.229 -0.465 0.492 0.529 0.735 -0.422 0.004 0.636 0.427
112) 0.239 -0.395 0.780 0.039 0.183 -0.317 0.774 0.029 -0.356 -0.186 0.700 0.582 -0.406 -0.011 0.659 0.403
113) 0.572 -0.413 0.783 -0.018 0.579 -0.368 0.811 -0.143 -0.189 -0.234 0.719 0.466 -0.384 -0.046 0.666 0.435
114) 0.383 -0.362 0.742 0.150 0.470 -0.323 0.765 -0.071 -0.138 -0.242 0.726 0.404 -0.433 -0.004 0.647 0.506
115) 0.544 -0.432 0.777 0.127 0.549 -0.393 0.823 -0.096 -0.000 -0.287 0.738 0.333 -0.304 -0.018 0.662 0.411
116) 0.489 -0.395 0.734 0.148 0.523 -0.317 0.779 0.006 0.269 -0.270 0.720 0.221 -0.035 0.039 0.649 0.353
117) 0.217 -0.300 0.705 0.352 0.349 -0.292 0.753 0.098 0.392 -0.287 0.721 0.139 0.073 0.039 0.649 0.299
118) 0.111 -0.344 0.760 0.363 0.221 -0.349 0.788 0.096 0.298 -0.335 0.747 0.235 0.159 -0.116 0.755 0.236
119) -0.211 -0.263 0.710 0.432 -0.164 -0.285 0.750 0.244 0.232 -0.323 0.748 0.286 0.142 -0.124 0.747 0.259
120) -0.039 -0.245 0.672 0.347 -0.115 -0.235 0.706 0.263 0.290 -0.266 0.705 0.264 0.298 0.046 0.674 0.299
121) -0.139 -0.194 0.607 0.310 -0.315 -0.114 0.622 0.333 -0.530 0.617 0.465 0.786 -0.433 0.251 0.537 0.411
122) 0.117 -0.333 0.727 0.133 0.028 -0.235 0.720 0.015 -0.428 -0.024 0.647 0.662 -0.416 0.095 0.633 0.387
123) 0.333 -0.446 0.825 0.087 0.304 -0.285 0.753 -0.144 -0.305 -0.141 0.691 0.520 -0.406 0.004 0.679 0.498
124) 0.194 -0.406 0.767 0.200 0.274 -0.304 0.756 -0.085 -0.225 -0.202 0.723 0.412 -0.465 0.032 0.667 0.554
125) 0.272 -0.468 0.809 0.203 0.349 -0.342 0.786 -0.069 -0.029 -0.303 0.760 0.318 -0.293 -0.145 0.733 0.415
126) 0.300 -0.417 0.761 0.197 0.383 -0.349 0.800 -0.044 0.232 -0.250 0.726 0.234 0.019 -0.131 0.733 0.326
127) -0.307 0.722 0.407 0.266 -0.292 0.767 0.084 0.421 -0.278 0.735 0.101 0.110 -0.102 0.707 0.296
128) -0.128 -0.355 0.756 0.394 0.074 -0.336 0.807 0.200 0.327 -0.319 0.766 0.198 0.191 -0.180 0.773 0.272
129) -0.267 -0.344 0.742 0.379 -0.160 -0.311 0.784 0.301 0.261 -0.339 0.781 0.249 0.185 -0.244 0.800 0.216

130) -0.056 -0.285 0.693 0.300 -0.100 -0.247 0.728 0.296 0.348 -0.274 0.722 0.241 0.411 -0.074 0.718 0.216
131) -0.094 -0.150 0.574 0.212 -0.421 -0.025 0.574 0.436 -0.363 0.557 0.437 0.731 -0.363 0.229 0.560 0.360
132) 0.089 -0.241 0.659 0.075 -0.089 -0.171 0.672 0.143 -0.254 -0.008 0.644 0.577 -0.416 0.053 0.654 0.387
133) 0.206 -0.362 0.784 0.073 0.096 -0.235 0.724 0.022 -0.210 -0.073 0.671 0.483 -0.363 -0.039 0.696 0.371
134) 0.194 -0.373 0.773 0.135 0.142 -0.273 0.743 0.010 -0.123 -0.210 0.726 0.347 -0.422 0.032 0.667 0.427
135) 0.250 -0.439 0.796 0.187 0.236 -0.336 0.794 0.018 0.065 -0.323 0.756 0.282 -0.298 -0.124 0.727 0.360
136) 0.328 -0.355 0.730 0.126 0.240 -0.266 0.788 0.076 0.327 -0.295 0.747 0.198 0.008 -0.102 0.723 0.310
137) 0.072 -0.234 0.702 0.376 0.213 -0.108 0.708 0.166 0.537 -0.283 0.744 0.048 0.099 -0.131 0.715 0.280
138) -0.006 -0.289 0.743 0.373 0.089 -0.127 0.722 0.221 0.334 -0.299 0.754 0.170 0.164 -0.180 0.758 0.280
139) -0.083 -0.293 0.725 0.315 -0.115 -0.127 0.708 0.290 0.254 -0.287 0.753 0.277 0.164 -0.236 0.799 0.280
140) 0.100 -0.241 0.683 0.243 -0.002 -0.082 0.675 0.203 0.370 -0.238 0.734 0.274 0.282 -0.145 0.759 0.253
141) -0.178 -0.230 0.651 0.239 -0.285 -0.196 0.683 0.314 -0.174 0.274 0.555 0.716 -0.142 0.088 0.636 0.304
142) -0.067 -0.234 0.658 0.206 -0.092 -0.235 0.709 0.112 -0.044 -0.129 0.679 0.545 -0.244 -0.053 0.719 0.403
143) 0.072 -0.271 0.707 0.156 0.092 -0.184 0.690 -0.009 0.065 -0.174 0.707 0.400 -0.153 -0.025 0.701 0.299
144) 0.028 -0.220 0.682 0.254 0.058 -0.177 0.684 0.065 0.080 -0.222 0.727 0.343 -0.250 0.159 0.606 0.358
145) 0.083 -0.249 0.712 0.306 0.092 -0.184 0.705 0.150 0.087 -0.161 0.678 0.333 -0.304 0.138 0.636 0.400
146) 0.228 -0.157 0.658 0.236 0.130 -0.082 0.697 0.133 0.312 -0.149 0.709 0.255 0.099 0.074 0.675 0.291
147) 0.044 -0.055 0.645 0.447 0.104 0.038 0.641 0.224 0.552 -0.125 0.692 0.059 0.175 0.074 0.641 0.285
148) 0.122 -0.216 0.698 0.366 0.160 -0.006 0.683 0.260 0.414 -0.218 0.726 0.130 0.293 -0.088 0.705 0.173
149) 0.044 -0.271 0.699 0.309 -0.062 -0.044 0.714 0.336 0.312 -0.182 0.711 0.255 0.298 -0.180 0.765 0.229
150) 0.194 -0.296 0.710 0.208 0.040 -0.082 0.679 0.187 0.370 -0.141 0.702 0.274 0.309 -0.145 0.771 0.245
151) -0.183 -0.293 0.692 0.270 -0.270 -0.317 0.786 0.347 -0.145 -0.040 0.629 0.602 -0.159 -0.173 0.755 0.328
152) -0.139 -0.216 0.652 0.245 -0.225 -0.298 0.777 0.249 -0.022 -0.214 0.672 0.460 -0.277 -0.173 0.755 0.450
153) -0.017 -0.194 0.665 0.213 -0.058 -0.158 0.695 0.136 0.145 -0.210 0.691 0.292 -0.142 -0.032 0.687 0.315
154) -0.017 -0.099 0.617 0.287 -0.058 -0.082 0.638 0.136 0.145 -0.145 0.674 0.292 -0.244 0.314 0.562 0.414
155) -0.044 0.033 0.563 0.367 -0.119 0.171 0.572 0.300 0.029 0.157 0.569 0.364 -0.298 0.632 0.522 0.455
156) 0.139 0.157 0.512 0.303 -0.089 0.469 0.509 0.354 0.138 0.230 0.575 0.339 0.024 0.533 0.562 0.312
157) -0.022 0.110 0.562 0.465 -0.119 0.494 0.453 0.387 0.290 0.234 0.555 0.212 0.046 0.476 0.544 0.332
158) 0.100 -0.055 0.609 0.424 0.074 0.215 0.566 0.343 0.334 -0.032 0.656 0.227 0.164 0.272 0.588 0.221
159) 0.022 -0.168 0.660 0.367 -0.100 0.152 0.622 0.379 0.254 -0.004 0.655 0.335 0.212 -0.011 0.698 0.234
160) 0.061 -0.223 0.685 0.299 -0.108 0.051 0.633 0.276 0.218 0.028 0.675 0.451 0.212 -0.067 0.716 0.234
161) -0.283 -0.326 0.693 0.389 -0.334 -0.323 0.784 0.367 -0.044 -0.246 0.721 0.477 -0.175 -0.236 0.773 0.352
162) -0.272 -0.187 0.638 0.336 -0.319 -0.254 0.741 0.301 0.015 -0.254 0.704 0.386 -0.228 -0.124 0.742 0.463
163) -0.200 -0.168 0.650 0.359 -0.285 -0.127 0.684 0.228 0.174 -0.270 0.720 0.247 -0.110 -0.025 0.716 0.352
164) -0.094 -0.135 0.620 0.357 -0.213 -0.057 0.619 0.126 0.174 -0.141 0.676 0.247 -0.121 0.215 0.626 0.347
165) -0.139 0.011 0.562 0.455 -0.281 0.349 0.529 0.372 0.058 0.210 0.574 0.318 -0.196 0.596 0.527 0.357
166) 0.061 0.150 0.518 0.373 -0.191 0.640 0.480 0.349 0.131 0.331 0.552 0.299 0.116 0.589 0.511 0.197
167) 0.039 0.088 0.562 0.422 -0.108 0.665 0.454 0.309 0.276 0.331 0.534 0.200 0.180 0.589 0.499 0.186
168) 0.172 -0.106 0.635 0.385 0.104 0.412 0.504 0.257 0.327 0.004 0.672 0.188 0.175 0.328 0.591 0.142
169) 0.128 -0.267 0.740 0.290 0.062 0.298 0.611 0.254 0.290 -0.069 0.685 0.262 0.201 0.081 0.649 0.218
170) 0.039 -0.322 0.781 0.275 -0.025 0.127 0.654 0.236 0.196 -0.036 0.717 0.400 0.159 -0.131 0.730 0.250
171) -0.417 0.015 0.547 0.609 -0.402 -0.082 0.664 0.515 -0.007 -0.198 0.723 0.404 -0.180 -0.046 0.692 0.391
172) -0.400 0.084 0.512 0.526 -0.308 -0.013 0.650 0.436 0.225 -0.214 0.729 0.286 -0.212 0.088 0.644 0.439
173) -0.372 0.048 0.551 0.520 -0.270 -0.019 0.633 0.321 0.356 -0.234 0.725 0.192 -0.137 0.187 0.638 0.360
174) -0.211 -0.007 0.555 0.432 -0.145 -0.032 0.608 0.137 0.377 -0.129 0.680 0.175 -0.110 0.236 0.595 0.268
175) -0.228 0.066 0.525 0.450 -0.183 0.349 0.493 0.350 0.247 0.113 0.585 0.235 -0.207 0.611 0.494 0.341
176) -0.006 0.055 0.520 0.319 -0.092 0.653 0.458 0.327 0.181 0.214 0.575 0.287 -0.008 0.575 0.462 0.194
177) 0.067 0.011 0.568 0.288 -0.006 0.602 0.495 0.258 0.203 0.165 0.615 0.270 0.003 0.476 0.544 0.210
178) 0.183 -0.117 0.644 0.270 0.126 0.323 0.611 0.219 0.254 -0.089 0.700 0.257 -0.067 0.364 0.575 0.261
179) 0.167 -0.267 0.733 0.223 0.081 0.279 0.637 0.257 0.007 -0.040 0.657 0.431 0.008 0.166 0.627 0.266
180) 0.078 -0.351 0.773 0.207 -0.036 0.082 0.696 0.272 -0.065 -0.105 0.708 0.552 -0.046 -0.124 0.706 0.282
181) -0.439 0.124 0.505 0.732 -0.413 0.120 0.578 0.539 -0.000 -0.040 0.650 0.375 -0.180 0.194 0.602 0.391
182) -0.461 0.293 0.445 0.716 -0.394 0.254 0.532 0.603 0.203 -0.052 0.643 0.303 -0.212 0.244 0.562 0.439
183) -0.394 0.227 0.489 0.708 -0.308 0.114 0.561 0.436 0.269 -0.165 0.693 0.260 -0.051 0.124 0.627 0.285
184) -0.194 0.044 0.553 0.552 -0.228 -0.025 0.624 0.264 0.283 -0.105 0.674 0.272 -0.024 0.102 0.637 0.193
185) -0.206 0.026 0.575 0.539 -0.175 0.279 0.573 0.439 0.203 0.081 0.603 0.330 -0.191 0.427 0.542 0.317
186) 0.022 -0.018 0.584 0.378 -0.055 0.393 0.553 0.383 0.138 0.036 0.632 0.381 -0.035 0.364 0.524 0.202
187) 0.106 -0.084 0.581 0.286 0.070 0.139 0.636 0.199 0.261 -0.016 0.683 0.367 -0.024 0.279 0.587 0.218
188) 0.239 -0.139 0.638 0.249 0.357 -0.013 0.686 0.004 0.494 -0.226 0.752 0.300 -0.008 0.173 0.625 0.194
189) 0.317 -0.252 0.697 0.176 0.326 -0.032 0.702 0.049 0.254 -0.182 0.711 0.445 -0.024 0.201 0.581 0.314
190) 0.189 -0.296 0.733 0.228 0.255 -0.139 0.722 0.052 0.247 -0.194 0.729 0.516 -0.126 -0.011 0.660 0.401
191) -0.400 0.391 0.422 0.738 -0.413 0.336 0.465 0.554 -0.152 0.161 0.536 0.442 -0.320 0.512 0.480 0.577
192) -0.478 0.794 0.314 0.808 -0.458 0.646 0.383 0.736 0.015 0.319 0.469 0.444 -0.293 0.822 0.360 0.654
193) -0.261 0.728 0.354 0.795 -0.292 0.456 0.428 0.555 0.094 0.182 0.540 0.405 0.040 0.519 0.518 0.486
194) -0.061 0.472 0.434 0.639 -0.232 0.254 0.511 0.490 0.109 0.170 0.566 0.416 0.116 0.194 0.627 0.322
195) -0.067 0.230 0.503 0.596 -0.149 0.279 0.551 0.534 0.044 0.129 0.598 0.417 -0.051 0.335 0.577 0.446
196) 0.300 -0.066 0.626 0.375 0.032 0.146 0.602 0.401 -0.080 0.028 0.638 0.491 0.148 0.138 0.602 0.289
197) 0.389 -0.260 0.698 0.253 0.157 0.006 0.665 0.217 0.029 -0.077 0.694 0.466 0.180 0.018 0.675 0.241
198) 0.550 -0.315 0.748 0.210 0.391 -0.139 0.722 0.017 0.385 -0.270 0.746 0.317 0.201 -0.067 0.701 0.177
199) 0.583 -0.362 0.762 0.162 0.417 -0.120 0.713 0.067 0.152 -0.190 0.697 0.434 0.099 0.046 0.655 0.305
200) 0.411 -0.377 0.789 0.238 0.326 -0.215 0.737 0.079 0.007 -0.137 0.652 0.525 -0.067 0.053 0.666 0.488

Non-Urban area

1) -0.386 -0.114 0.119 0.680 -0.569 -0.216 0.082 0.675 -0.481 -0.103 0.072 0.744 -0.202 -0.276 0.152 0.483
2) -0.386 -0.099 0.171 0.680 -0.496 -0.248 0.112 0.625 -0.440 -0.114 0.142 0.722 -0.276 -0.283 0.222 0.535
3) -0.214 -0.039 0.201 0.645 -0.349 -0.210 0.157 0.585 -0.523 0.048 0.052 0.767 -0.349 -0.154 0.145 0.587
4) -0.343 0.245 0.040 0.711 -0.386 -0.009 0.028 0.610 -0.607 0.331 -0.096 0.811 -0.533 0.171 -0.027 0.717
5) -0.343 0.468 -0.108 0.711 -0.459 0.263 -0.156 0.660 -0.691 0.631 -0.260 0.856 -0.606 0.496 -0.215 0.770
6) -0.343 0.468 -0.108 0.711 -0.496 0.338 -0.199 0.685 -0.733 0.718 -0.316 0.878 -0.606 0.496 -0.215 0.770
7) -0.471 0.462 -0.089 0.777 -0.533 0.253 -0.130 0.710 -0.691 0.482 -0.190 0.856 -0.680 0.489 -0.196 0.822
8) -0.471 0.462 -0.089 0.777 -0.533 0.253 -0.130 0.710 -0.649 0.543 -0.189 0.833 -0.680 0.489 -0.196 0.822
9) -0.514 0.237 0.058 0.746 -0.643 0.112 -0.057 0.725 -0.649 0.321 -0.037 0.833 -0.422 0.145 0.076 0.702
10) -0.429 -0.081 0.302 0.702 -0.569 -0.199 0.226 0.675 -0.565 -0.013 0.204 0.789 -0.276 -0.199 0.358 0.597
11) -0.343 -0.172 0.152 0.659 -0.643 -0.150 0.027 0.725 -0.481 -0.116 0.078 0.744 -0.165 -0.279 0.154 0.457
12) -0.300 -0.215 0.272 0.637 -0.533 -0.249 0.170 0.650 -0.481 -0.148 0.200 0.744 -0.239 -0.327 0.282 0.509
13) -0.086 -0.231 0.369 0.580 -0.239 -0.275 0.293 0.570 -0.523 -0.051 0.147 0.767 -0.276 -0.265 0.265 0.535
14) -0.171 -0.063 0.268 0.623 -0.239 -0.148 0.213 0.570 -0.607 0.196 0.013 0.811 -0.422 -0.043 0.139 0.639
15) -0.171 0.110 0.153 0.623 -0.276 0.030 0.090 0.595 -0.649 0.391 -0.090 0.833 -0.459 0.163 0.003 0.665
16) -0.214 0.170 0.119 0.645 -0.239 -0.030 0.127 0.570 -0.607 0.317 -0.045 0.811 -0.496 0.234 -0.034 0.691
17) -0.343 0.231 0.090 0.711 -0.239 -0.092 0.174 0.570 -0.565 0.132 0.053 0.789 -0.569 0.223 -0.008 0.743
18) -0.214 0.230 0.100 0.698 -0.129 -0.089 0.132 0.555 -0.523 0.166 0.085 0.767 -0.569 0.223 -0.008 0.743
19) -0.300 0.034 0.248 0.689 -0.386 -0.193 0.213 0.610 -0.523 -0.010 0.191 0.767 -0.349 -0.073 0.261 0.650
20) -0.257 -0.236 0.460 0.667 -0.349 -0.413 0.444 0.585 -0.481 -0.247 0.356 0.744 -0.202 -0.395 0.515 0.545
21) -0.386 -0.138 0.131 0.680 -0.643 -0.130 0.018 0.725 -0.523 -0.079 0.053 0.767 -0.239 -0.268 0.148 0.509
22) -0.343 -0.202 0.261 0.659 -0.496 -0.278 0.227 0.625 -0.481 -0.158 0.233 0.744 -0.312 -0.312 0.274 0.561
23) -0.129 -0.253 0.379 0.601 -0.202 -0.327 0.362 0.545 -0.523 -0.090 0.191 0.767 -0.276 -0.320 0.320 0.535
24) -0.129 -0.193 0.345 0.601 -0.165 -0.297 0.344 0.520 -0.565 0.076 0.104 0.789 -0.349 -0.209 0.252 0.587
25) -0.129 -0.110 0.296 0.601 -0.202 -0.209 0.285 0.545 -0.607 0.191 0.038 0.811 -0.386 -0.109 0.191 0.613
26) -0.171 -0.063 0.268 0.623 -0.165 -0.254 0.315 0.520 -0.565 0.076 0.104 0.789 -0.422 -0.054 0.161 0.639
27) -0.300 0.049 0.196 0.689 -0.165 -0.254 0.315 0.520 -0.523 -0.024 0.161 0.767 -0.496 0.005 0.128 0.691
28) -0.129 0.043 0.213 0.654 0.055 -0.248 0.254 0.490 -0.440 -0.024 0.225 0.722 -0.459 0.006 0.127 0.665
29) -0.214 -0.123 0.344 0.645 -0.165 -0.344 0.409 0.520 -0.481 -0.119 0.281 0.744 -0.276 -0.192 0.344 0.597
30) -0.214 -0.345 0.535 0.645 -0.165 -0.505 0.600 0.520 -0.440 -0.296 0.408 0.722 -0.165 -0.435 0.548 0.519
31) -0.386 -0.084 0.104 0.680 -0.643 -0.071 -0.010 0.725 -0.523 -0.040 0.036 0.767 -0.276 -0.187 0.096 0.535
32) -0.343 -0.153 0.236 0.659 -0.459 -0.261 0.222 0.600 -0.440 -0.153 0.240 0.722 -0.276 -0.280 0.253 0.535
33) -0.043 -0.264 0.390 0.558 -0.129 -0.366 0.392 0.495 -0.440 -0.153 0.240 0.722 -0.202 -0.335 0.324 0.483
34) 0.0 -0.264 0.389 0.536 -0.055 -0.393 0.410 0.445 -0.481 -0.037 0.177 0.744 -0.239 -0.312 0.311 0.509
35) 0.0 -0.228 0.365 0.536 -0.092 -0.366 0.391 0.470 -0.523 0.018 0.142 0.767 -0.276 -0.283 0.294 0.535
36) -0.043 -0.193 0.345 0.558 -0.055 -0.393 0.410 0.445 -0.481 -0.108 0.210 0.744 -0.312 -0.249 0.275 0.561
37) -0.171 -0.110 0.295 0.623 -0.055 -0.360 0.384 0.445 -0.398 -0.193 0.267 0.700 -0.386 -0.212 0.256 0.613
38) 0.043 -0.155 0.332 0.566 0.129 -0.359 0.330 0.440 -0.314 -0.215 0.345 0.656 -0.386 -0.164 0.227 0.613
39) -0.129 -0.208 0.396 0.601 -0.129 -0.409 0.457 0.495 -0.356 -0.257 0.378 0.678 -0.202 -0.281 0.397 0.545
40) -0.129 -0.384 0.568 0.601 -0.129 -0.527 0.619 0.495 -0.356 -0.370 0.467 0.678 -0.165 -0.466 0.573 0.519
41) -0.386 -0.007 0.065 0.680 -0.643 0.014 -0.053 0.725 -0.649 0.150 -0.080 0.833 -0.349 -0.076 0.031 0.587
42) -0.300 -0.125 0.224 0.637 -0.496 -0.180 0.170 0.625 -0.523 -0.024 0.161 0.767 -0.312 -0.212 0.211 0.561
43) 0.043 -0.292 0.406 0.514 -0.129 -0.366 0.392 0.495 -0.523 -0.090 0.191 0.767 -0.202 -0.335 0.324 0.483
44) 0.129 -0.336 0.433 0.470 -0.018 -0.438 0.443 0.420 -0.565 -0.007 0.141 0.789 -0.165 -0.353 0.334 0.457
45) 0.086 -0.316 0.421 0.492 -0.018 -0.438 0.443 0.420 -0.565 -0.007 0.141 0.789 -0.202 -0.335 0.324 0.483
46) 0.086 -0.286 0.399 0.492 0.018 -0.432 0.437 0.395 -0.565 -0.067 0.167 0.789 -0.276 -0.283 0.294 0.535
47) 0.086 -0.246 0.411 0.544 -0.018 -0.385 0.401 0.420 -0.481 -0.143 0.244 0.744 -0.239 -0.272 0.352 0.571
48) 0.343 -0.321 0.479 0.465 0.312 -0.423 0.388 0.375 -0.314 -0.258 0.380 0.656 -0.239 -0.272 0.352 0.571
49) 0.129 -0.338 0.532 0.522 0.018 -0.449 0.509 0.455 -0.356 -0.263 0.390 0.678 -0.129 -0.302 0.446 0.556
50) 0.043 -0.424 0.627 0.566 -0.018 -0.527 0.619 0.480 -0.314 -0.350 0.463 0.656 -0.165 -0.404 0.547 0.582
51) -0.214 -0.039 0.201 0.645 -0.569 0.069 -0.036 0.735 -0.565 0.152 -0.004 0.789 -0.276 -0.061 0.052 0.535
52) -0.171 -0.151 0.319 0.623 -0.459 -0.130 0.188 0.660 -0.481 0.000 0.176 0.744 -0.239 -0.203 0.211 0.509
53) 0.214 -0.375 0.511 0.479 -0.092 -0.363 0.405 0.530 -0.481 -0.143 0.244 0.744 -0.092 -0.369 0.351 0.404
54) 0.300 -0.424 0.545 0.435 0.092 -0.482 0.500 0.405 -0.481 -0.143 0.244 0.744 -0.055 -0.398 0.376 0.378
55) 0.300 -0.411 0.535 0.435 0.129 -0.484 0.502 0.380 -0.523 -0.095 0.209 0.767 -0.129 -0.387 0.369 0.430
56) 0.257 -0.356 0.480 0.457 0.129 -0.452 0.476 0.380 -0.481 -0.143 0.244 0.744 -0.239 -0.312 0.311 0.509
57) 0.300 -0.327 0.520 0.487 0.018 -0.393 0.423 0.395 -0.398 -0.187 0.337 0.700 0.129 -0.326 0.444 0.498
58) 0.729 -0.426 0.639 0.373 0.496 -0.477 0.551 0.310 -0.272 -0.308 0.466 0.633 0.165 -0.371 0.484 0.472
59) 0.429 -0.425 0.681 0.474 0.239 -0.493 0.629 0.365 -0.272 -0.343 0.489 0.633 0.092 -0.353 0.517 0.524
60) 0.343 -0.466 0.728 0.518 0.202 -0.546 0.709 0.390 -0.230 -0.360 0.518 0.611 0.018 -0.369 0.552 0.577
61) -0.300 0.181 0.076 0.689 -0.422 0.050 0.012 0.695 -0.398 0.187 0.039 0.753 -0.202 -0.048 0.112 0.545
62) -0.257 0.044 0.203 0.667 -0.312 -0.149 0.236 0.620 -0.356 0.047 0.187 0.731 -0.165 -0.172 0.234 0.519
63) 0.086 -0.257 0.434 0.544 -0.018 -0.406 0.442 0.480 -0.272 -0.206 0.328 0.687 -0.018 -0.365 0.398 0.415
64) 0.214 -0.375 0.511 0.479 0.165 -0.492 0.508 0.355 -0.188 -0.277 0.376 0.642 0.092 -0.403 0.428 0.337

65) 0.214 -0.350 0.494 0.479 0.165 -0.482 0.500 0.355 -0.272 -0.206 0.328 0.687 0.018 -0.403 0.428 0.389
66) 0.171 -0.277 0.428 0.500 0.018 -0.406 0.414 0.395 -0.272 -0.206 0.328 0.687 -0.129 -0.318 0.359 0.493
67) 0.300 -0.268 0.495 0.487 0.092 -0.358 0.394 0.345 -0.314 -0.207 0.369 0.656 0.165 -0.318 0.433 0.472
68) 0.900 -0.419 0.657 0.338 0.606 -0.491 0.598 0.235 -0.188 -0.340 0.494 0.589 0.533 -0.410 0.535 0.399
69) 0.643 -0.432 0.704 0.417 0.496 -0.496 0.636 0.250 -0.147 -0.393 0.528 0.567 0.459 -0.425 0.585 0.451
70) 0.557 -0.476 0.754 0.461 0.459 -0.531 0.700 0.275 -0.105 -0.384 0.545 0.544 0.349 -0.411 0.600 0.530
71) -0.214 0.011 0.173 0.645 -0.312 -0.080 0.108 0.620 -0.356 0.182 0.047 0.731 -0.202 -0.048 0.112 0.545
72) -0.171 -0.099 0.267 0.623 -0.239 -0.197 0.245 0.570 -0.272 0.053 0.148 0.687 -0.165 -0.148 0.185 0.519
73) -0.043 -0.264 0.390 0.558 -0.092 -0.377 0.366 0.470 -0.230 -0.167 0.266 0.664 -0.092 -0.280 0.304 0.467
74) 0.086 -0.341 0.438 0.492 0.018 -0.429 0.401 0.395 -0.188 -0.206 0.291 0.642 -0.018 -0.343 0.348 0.415
75) 0.043 -0.292 0.406 0.514 -0.018 -0.393 0.375 0.420 -0.314 -0.076 0.208 0.709 -0.092 -0.317 0.333 0.467
76) -0.086 -0.140 0.290 0.580 -0.165 -0.280 0.252 0.460 -0.314 -0.076 0.208 0.709 -0.239 -0.178 0.222 0.571
77) 0.043 -0.070 0.322 0.566 -0.018 -0.259 0.307 0.420 -0.230 -0.073 0.301 0.664 0.092 -0.210 0.331 0.524
78) 0.643 -0.267 0.514 0.417 0.386 -0.420 0.529 0.325 -0.105 -0.222 0.413 0.598 0.606 -0.372 0.515 0.410
79) 0.600 -0.313 0.577 0.439 0.459 -0.443 0.548 0.275 -0.063 -0.322 0.474 0.576 0.606 -0.409 0.544 0.410
80) 0.514 -0.368 0.641 0.483 0.459 -0.495 0.625 0.275 0.021 -0.340 0.503 0.531 0.496 -0.390 0.545 0.488
81) -0.171 -0.162 0.270 0.623 -0.202 -0.265 0.227 0.545 -0.230 -0.039 0.169 0.664 -0.092 -0.208 0.215 0.467
82) -0.171 -0.192 0.286 0.623 -0.165 -0.300 0.250 0.520 -0.188 -0.085 0.195 0.642 -0.055 -0.229 0.225 0.441
83) -0.043 -0.281 0.342 0.558 0.018 -0.383 0.302 0.395 -0.188 -0.192 0.252 0.642 -0.055 -0.269 0.262 0.441
84) 0.043 -0.303 0.356 0.514 0.092 -0.376 0.292 0.345 -0.147 -0.198 0.258 0.620 -0.018 -0.282 0.268 0.415
85) -0.043 -0.231 0.312 0.558 0.018 -0.324 0.250 0.395 -0.272 -0.035 0.165 0.687 -0.129 -0.229 0.240 0.493
86) -0.171 -0.068 0.159 0.623 -0.129 -0.187 0.091 0.435 -0.272 -0.035 0.165 0.687 -0.276 -0.059 0.089 0.597
87) -0.043 -0.001 0.235 0.610 -0.018 -0.182 0.172 0.420 -0.188 -0.013 0.239 0.642 0.055 -0.089 0.220 0.550
88) 0.429 -0.164 0.406 0.474 0.239 -0.330 0.424 0.365 -0.147 -0.113 0.298 0.620 0.496 -0.245 0.417 0.488
89) 0.429 -0.213 0.447 0.474 0.276 -0.359 0.442 0.340 -0.147 -0.229 0.363 0.620 0.533 -0.331 0.474 0.462
90) 0.386 -0.298 0.511 0.496 0.276 -0.438 0.529 0.340 -0.105 -0.264 0.385 0.598 0.422 -0.323 0.465 0.540
91) -0.129 -0.248 0.319 0.601 -0.129 -0.376 0.300 0.495 -0.230 -0.182 0.243 0.664 -0.129 -0.329 0.312 0.493
92) -0.129 -0.248 0.319 0.601 -0.129 -0.376 0.300 0.495 -0.230 -0.182 0.243 0.664 -0.129 -0.329 0.312 0.493
93) 0.0 -0.302 0.355 0.536 0.055 -0.408 0.322 0.370 -0.188 -0.236 0.274 0.642 -0.129 -0.329 0.312 0.493
94) 0.043 -0.303 0.356 0.514 0.129 -0.379 0.294 0.320 -0.105 -0.255 0.290 0.598 -0.092 -0.318 0.303 0.467
95) -0.043 -0.231 0.312 0.558 0.055 -0.339 0.259 0.370 -0.230 -0.083 0.193 0.664 -0.239 -0.222 0.250 0.571
96) -0.129 -0.109 0.182 0.601 -0.092 -0.212 0.105 0.410 -0.230 -0.083 0.193 0.664 -0.349 -0.073 0.106 0.650
97) -0.043 -0.053 0.267 0.610 -0.018 -0.238 0.257 0.420 -0.230 -0.014 0.240 0.664 -0.018 -0.049 0.210 0.603
98) 0.386 -0.174 0.424 0.496 0.276 -0.310 0.360 0.340 -0.188 -0.066 0.270 0.642 0.459 -0.198 0.389 0.514
99) 0.429 -0.213 0.447 0.474 0.312 -0.337 0.376 0.315 -0.188 -0.189 0.338 0.642 0.496 -0.294 0.450 0.488
100) 0.429 -0.296 0.508 0.474 0.276 -0.438 0.529 0.340 -0.188 -0.220 0.354 0.642 0.386 -0.280 0.434 0.566
101) -0.683 0.112 -0.075 0.750 -0.540 -0.095 -0.074 0.662 -0.675 0.104 -0.177 0.729 -0.777 0.097 -0.219 0.848
102) -0.652 0.103 -0.062 0.729 -0.568 -0.049 -0.096 0.688 -0.675 0.057 -0.156 0.729 -0.752 0.081 -0.200 0.822
103) -0.528 -0.095 0.064 0.643 -0.540 -0.144 -0.047 0.662 -0.731 -0.012 -0.150 0.771 -0.678 -0.094 -0.100 0.743
104) -0.434 -0.205 0.131 0.579 -0.512 -0.220 -0.008 0.635 -0.759 0.039 -0.186 0.793 -0.604 -0.189 -0.046 0.665
105) -0.279 -0.343 0.354 0.471 -0.457 -0.309 0.147 0.582 -0.759 -0.040 -0.027 0.793 -0.407 -0.356 0.207 0.519
106) -0.155 -0.476 0.500 0.386 -0.402 -0.455 0.316 0.529 -0.731 -0.113 0.054 0.771 -0.308 -0.502 0.344 0.415
107) -0.000 -0.544 0.621 0.279 -0.291 -0.551 0.411 0.424 -0.619 -0.307 0.210 0.737 -0.111 -0.576 0.467 0.269
108) 0.031 -0.514 0.559 0.257 -0.263 -0.531 0.386 0.397 -0.591 -0.351 0.251 0.716 -0.012 -0.570 0.457 0.227
109) 0.466 -0.580 0.741 0.163 0.014 -0.591 0.527 0.323 -0.534 -0.348 0.245 0.724 0.259 -0.608 0.514 0.128
110) 0.652 -0.569 0.735 0.086 0.180 -0.594 0.538 0.228 -0.506 -0.382 0.282 0.703 0.308 -0.585 0.489 0.076
111) -0.652 0.047 0.021 0.729 -0.457 -0.228 0.048 0.582 -0.591 -0.063 -0.037 0.664 -0.727 -0.028 -0.143 0.796
112) -0.652 0.103 -0.062 0.729 -0.512 -0.136 -0.055 0.635 -0.647 0.004 -0.126 0.707 -0.727 0.018 -0.163 0.796
113) -0.528 -0.095 0.064 0.643 -0.457 -0.202 -0.024 0.582 -0.703 -0.060 -0.118 0.750 -0.653 -0.143 -0.073 0.717
114) -0.403 -0.202 0.128 0.557 -0.429 -0.228 -0.012 0.556 -0.731 0.015 -0.160 0.771 -0.555 -0.222 -0.031 0.613
115) -0.248 -0.348 0.367 0.450 -0.402 -0.307 0.153 0.529 -0.759 -0.055 -0.010 0.793 -0.382 -0.362 0.220 0.493
116) -0.093 -0.491 0.512 0.343 -0.318 -0.475 0.329 0.450 -0.788 -0.010 -0.035 0.814 -0.308 -0.481 0.327 0.415
117) 0.062 -0.532 0.606 0.236 -0.180 -0.523 0.379 0.318 -0.675 -0.224 0.121 0.780 -0.111 -0.569 0.459 0.269
118) 0.062 -0.494 0.499 0.236 -0.180 -0.504 0.343 0.318 -0.647 -0.286 0.175 0.759 -0.012 -0.565 0.452 0.227
119) 0.590 -0.576 0.706 0.129 0.152 -0.581 0.509 0.254 -0.563 -0.327 0.211 0.746 0.333 -0.616 0.525 0.112
120) 0.652 -0.550 0.678 0.086 0.208 -0.555 0.485 0.201 -0.563 -0.327 0.211 0.746 0.358 -0.592 0.500 0.086
121) -0.652 0.047 0.021 0.729 -0.457 -0.228 0.048 0.582 -0.563 -0.119 0.021 0.643 -0.703 -0.095 -0.053 0.770
122) -0.683 0.162 -0.100 0.750 -0.540 -0.095 -0.074 0.662 -0.647 0.049 -0.147 0.707 -0.752 0.081 -0.200 0.822
123) -0.528 -0.053 0.039 0.643 -0.457 -0.157 -0.054 0.582 -0.703 -0.035 -0.127 0.750 -0.653 -0.100 -0.094 0.717
124) -0.372 -0.155 0.090 0.536 -0.402 -0.164 -0.064 0.529 -0.731 0.084 -0.188 0.771 -0.530 -0.167 -0.071 0.587
125) -0.217 -0.313 0.334 0.429 -0.374 -0.259 0.110 0.503 -0.759 -0.023 -0.019 0.793 -0.358 -0.316 0.177 0.467
126) -0.062 -0.460 0.469 0.321 -0.291 -0.433 0.277 0.424 -0.788 0.084 -0.079 0.814 -0.284 -0.437 0.274 0.389
127) 0.124 -0.497 0.542 0.193 -0.125 -0.476 0.307 0.265 -0.675 -0.167 0.090 0.780 -0.062 -0.521 0.360 0.217
128) 0.124 -0.475 0.462 0.193 -0.125 -0.476 0.307 0.265 -0.647 -0.264 0.159 0.759 0.037 -0.539 0.391 0.175
129) 0.714 -0.573 0.694 0.094 0.263 -0.578 0.501 0.212 -0.563 -0.327 0.211 0.746 0.432 -0.605 0.481 0.070
130) 0.745 -0.534 0.651 0.073 0.291 -0.528 0.452 0.185 -0.563 -0.327 0.211 0.746 0.432 -0.564 0.436 0.070

131) -0.621 -0.073 0.171 0.707 -0.540 -0.133 0.032 0.662 -0.619 -0.032 -0.006 0.686 -0.727 -0.048 -0.052 0.796
132) -0.652 0.024 0.108 0.729 -0.595 -0.029 -0.021 0.715 -0.703 0.151 -0.141 0.750 -0.752 0.068 -0.127 0.822
133) -0.466 -0.177 0.251 0.600 -0.485 -0.168 0.043 0.609 -0.731 0.005 -0.097 0.771 -0.604 -0.179 0.034 0.665
134) -0.279 -0.244 0.264 0.471 -0.374 -0.196 0.023 0.503 -0.759 0.136 -0.170 0.793 -0.456 -0.258 0.054 0.509
135) -0.124 -0.373 0.375 0.364 -0.291 -0.331 0.127 0.424 -0.788 -0.015 -0.030 0.814 -0.358 -0.363 0.125 0.404
136) -0.000 -0.460 0.450 0.279 -0.180 -0.429 0.194 0.318 -0.844 0.164 -0.196 0.857 -0.284 -0.443 0.184 0.326
137) 0.217 -0.443 0.391 0.129 -0.042 -0.426 0.191 0.185 -0.872 -0.020 -0.149 0.879 -0.160 -0.456 0.192 0.196
138) 0.186 -0.445 0.393 0.150 -0.069 -0.429 0.194 0.212 -0.900 -0.052 -0.196 0.900 -0.160 -0.456 0.192 0.196
139) 0.838 -0.566 0.652 0.060 0.429 -0.555 0.413 0.116 -0.900 -0.141 -0.143 0.900 0.284 -0.574 0.410 0.102
140) 0.838 -0.506 0.581 0.060 0.429 -0.492 0.353 0.116 -0.900 -0.141 -0.143 0.900 0.284 -0.515 0.350 0.102
141) -0.621 0.001 0.166 0.707 -0.568 -0.044 0.035 0.688 -0.647 0.054 -0.003 0.707 -0.727 0.034 -0.044 0.796
142) -0.621 -0.002 0.175 0.707 -0.568 -0.052 0.052 0.688 -0.703 0.185 -0.102 0.750 -0.727 0.034 -0.044 0.796
143) -0.434 -0.244 0.351 0.579 -0.457 -0.248 0.155 0.582 -0.731 -0.045 -0.006 0.771 -0.579 -0.259 0.152 0.639
144) -0.248 -0.343 0.396 0.450 -0.346 -0.317 0.171 0.476 -0.759 0.051 -0.067 0.793 -0.432 -0.371 0.201 0.483
145) -0.062 -0.452 0.460 0.321 -0.235 -0.429 0.233 0.371 -0.788 -0.099 0.012 0.814 -0.308 -0.456 0.232 0.352
146) 0.031 -0.491 0.496 0.257 -0.152 -0.465 0.259 0.291 -0.844 0.064 -0.151 0.857 -0.259 -0.485 0.254 0.300
147) 0.248 -0.437 0.384 0.107 -0.014 -0.419 0.183 0.159 -0.872 -0.060 -0.128 0.879 -0.136 -0.453 0.189 0.170
148) 0.186 -0.445 0.393 0.150 -0.069 -0.429 0.194 0.212 -0.900 -0.052 -0.196 0.900 -0.160 -0.456 0.192 0.196
149) 0.838 -0.566 0.652 0.060 0.429 -0.555 0.413 0.116 -0.900 -0.141 -0.143 0.900 0.284 -0.574 0.410 0.102
150) 0.838 -0.506 0.581 0.060 0.429 -0.492 0.353 0.116 -0.900 -0.141 -0.143 0.900 0.284 -0.515 0.350 0.102
151) -0.652 -0.050 0.203 0.729 -0.540 -0.116 0.095 0.662 -0.506 -0.095 0.068 0.600 -0.604 -0.092 0.034 0.665
152) -0.621 -0.059 0.220 0.707 -0.540 -0.116 0.095 0.662 -0.591 0.059 -0.017 0.664 -0.604 -0.092 0.034 0.665
153) -0.466 -0.257 0.380 0.600 -0.429 -0.315 0.216 0.556 -0.591 -0.194 0.142 0.664 -0.505 -0.311 0.193 0.561
154) -0.310 -0.346 0.426 0.493 -0.374 -0.371 0.242 0.503 -0.647 -0.092 0.067 0.707 -0.382 -0.417 0.238 0.430
155) -0.124 -0.420 0.436 0.364 -0.263 -0.451 0.266 0.397 -0.675 -0.223 0.136 0.729 -0.308 -0.460 0.236 0.352
156) -0.093 -0.442 0.454 0.343 -0.180 -0.455 0.252 0.318 -0.731 -0.078 0.021 0.771 -0.259 -0.480 0.250 0.300
157) 0.124 -0.460 0.406 0.193 -0.069 -0.438 0.200 0.212 -0.675 -0.178 0.092 0.780 -0.160 -0.469 0.202 0.196
158) 0.093 -0.514 0.531 0.214 -0.014 -0.491 0.345 0.222 -0.703 -0.185 0.063 0.801 -0.185 -0.496 0.277 0.222
159) 0.776 -0.611 0.810 0.103 0.402 -0.587 0.490 0.143 -0.450 -0.314 0.195 0.763 0.210 -0.599 0.513 0.117
160) 0.900 -0.556 0.758 0.069 0.512 -0.539 0.478 0.101 -0.478 -0.248 0.130 0.784 0.210 -0.540 0.454 0.117
161) -0.683 -0.007 0.110 0.750 -0.623 -0.080 0.003 0.741 -0.506 -0.164 0.047 0.600 -0.579 -0.177 0.031 0.639
162) -0.652 0.024 0.108 0.729 -0.623 -0.028 -0.022 0.741 -0.591 0.022 -0.054 0.664 -0.579 -0.124 -0.002 0.639
163) -0.466 -0.208 0.328 0.600 -0.512 -0.222 0.130 0.635 -0.563 -0.205 0.154 0.643 -0.481 -0.288 0.133 0.535
164) -0.310 -0.326 0.415 0.493 -0.429 -0.308 0.180 0.556 -0.591 -0.110 0.094 0.664 -0.382 -0.394 0.192 0.430
165) -0.155 -0.400 0.422 0.386 -0.346 -0.375 0.216 0.476 -0.591 -0.244 0.176 0.664 -0.333 -0.443 0.225 0.378
166) -0.093 -0.442 0.454 0.343 -0.235 -0.429 0.233 0.371 -0.675 -0.097 0.065 0.729 -0.308 -0.456 0.232 0.352
167) 0.124 -0.462 0.409 0.193 -0.125 -0.445 0.205 0.265 -0.619 -0.216 0.162 0.737 -0.210 -0.472 0.205 0.248
168) 0.124 -0.512 0.528 0.193 -0.042 -0.522 0.379 0.249 -0.647 -0.234 0.146 0.759 -0.210 -0.523 0.309 0.248
169) 0.714 -0.598 0.789 0.094 0.402 -0.599 0.550 0.143 -0.366 -0.402 0.351 0.699 0.111 -0.608 0.533 0.159
170) 0.807 -0.569 0.774 0.081 0.512 -0.574 0.542 0.101 -0.394 -0.311 0.270 0.720 0.136 -0.572 0.515 0.133
171) -0.714 0.014 -0.042 0.771 -0.678 -0.073 -0.093 0.794 -0.591 -0.154 -0.044 0.664 -0.604 -0.225 -0.028 0.665
172) -0.683 0.068 -0.053 0.750 -0.651 -0.046 -0.098 0.768 -0.647 0.004 -0.126 0.707 -0.604 -0.146 -0.070 0.665
173) -0.497 -0.140 0.220 0.621 -0.540 -0.207 0.106 0.662 -0.619 -0.177 0.089 0.686 -0.505 -0.263 0.092 0.561
174) -0.310 -0.279 0.338 0.493 -0.429 -0.308 0.180 0.556 -0.647 -0.043 0.014 0.707 -0.407 -0.349 0.155 0.457
175) -0.186 -0.371 0.398 0.407 -0.346 -0.370 0.210 0.476 -0.619 -0.193 0.128 0.686 -0.358 -0.414 0.203 0.404
176) -0.124 -0.455 0.507 0.364 -0.235 -0.429 0.233 0.371 -0.647 -0.152 0.114 0.707 -0.333 -0.470 0.280 0.378
177) 0.186 -0.500 0.541 0.201 -0.042 -0.469 0.275 0.249 -0.591 -0.267 0.211 0.716 -0.234 -0.489 0.257 0.274
178) 0.217 -0.530 0.601 0.180 0.069 -0.535 0.411 0.206 -0.619 -0.271 0.188 0.737 -0.210 -0.523 0.309 0.248
179) 0.714 -0.598 0.789 0.094 0.429 -0.593 0.542 0.116 -0.338 -0.426 0.382 0.677 0.037 -0.593 0.509 0.175
180) 0.776 -0.587 0.795 0.103 0.512 -0.598 0.567 0.101 -0.366 -0.350 0.309 0.699 0.037 -0.578 0.512 0.175
181) -0.683 -0.081 0.017 0.750 -0.623 -0.209 -0.022 0.741 -0.591 -0.214 -0.018 0.664 -0.604 -0.294 0.007 0.665
182) -0.621 -0.101 0.050 0.707 -0.623 -0.167 -0.040 0.741 -0.619 -0.141 -0.056 0.686 -0.555 -0.288 0.007 0.613
183) -0.466 -0.245 0.271 0.600 -0.540 -0.273 0.127 0.662 -0.591 -0.261 0.122 0.664 -0.456 -0.337 0.116 0.509
184) -0.279 -0.335 0.370 0.471 -0.429 -0.332 0.185 0.556 -0.619 -0.128 0.052 0.686 -0.358 -0.362 0.144 0.404
185) -0.186 -0.400 0.428 0.407 -0.374 -0.374 0.218 0.503 -0.591 -0.236 0.155 0.664 -0.333 -0.423 0.204 0.378
186) -0.155 -0.491 0.565 0.386 -0.291 -0.462 0.312 0.424 -0.619 -0.242 0.173 0.686 -0.333 -0.489 0.308 0.378
187) 0.155 -0.534 0.633 0.223 -0.014 -0.530 0.426 0.286 -0.534 -0.348 0.279 0.673 -0.234 -0.502 0.283 0.274
188) 0.155 -0.556 0.670 0.223 0.097 -0.580 0.504 0.244 -0.563 -0.333 0.262 0.694 -0.210 -0.518 0.305 0.248
189) 0.590 -0.596 0.782 0.129 0.402 -0.603 0.553 0.143 -0.309 -0.444 0.414 0.656 -0.037 -0.577 0.478 0.190
190) 0.621 -0.601 0.810 0.159 0.457 -0.623 0.596 0.154 -0.338 -0.374 0.349 0.677 -0.062 -0.584 0.503 0.217
191) -0.621 -0.160 0.079 0.707 -0.595 -0.249 0.000 0.715 -0.591 -0.224 -0.013 0.664 -0.604 -0.304 0.012 0.665
192) -0.528 -0.214 0.130 0.643 -0.623 -0.192 -0.030 0.741 -0.619 -0.181 -0.039 0.686 -0.530 -0.331 0.031 0.587
193) -0.434 -0.291 0.279 0.579 -0.595 -0.238 0.081 0.715 -0.619 -0.237 0.081 0.686 -0.456 -0.342 0.091 0.509
194) -0.279 -0.343 0.354 0.471 -0.485 -0.276 0.130 0.609 -0.647 -0.102 0.013 0.707 -0.382 -0.341 0.121 0.430
195) -0.186 -0.423 0.446 0.407 -0.429 -0.343 0.199 0.556 -0.647 -0.187 0.119 0.707 -0.382 -0.414 0.212 0.430
196) -0.186 -0.519 0.602 0.407 -0.374 -0.465 0.336 0.503 -0.675 -0.227 0.150 0.729 -0.382 -0.512 0.346 0.430
197) 0.124 -0.559 0.684 0.244 -0.069 -0.560 0.478 0.339 -0.563 -0.337 0.262 0.694 -0.259 -0.531 0.352 0.300
198) 0.093 -0.575 0.707 0.266 0.042 -0.596 0.531 0.296 -0.563 -0.338 0.266 0.694 -0.259 -0.546 0.368 0.300
199) 0.466 -0.588 0.762 0.163 0.291 -0.595 0.540 0.185 -0.309 -0.429 0.404 0.656 -0.136 -0.566 0.447 0.232
200) 0.466 -0.592 0.792 0.214 0.318 -0.610 0.579 0.222 -0.338 -0.339 0.319 0.677 -0.185 -0.582 0.484 0.284

Leather

1) 0.237 -0.524 0.762 0.374 0.225 -0.577 0.775 0.289 0.137 -0.512 0.755 0.466 -0.120 -0.506 0.729 0.495
2) 0.263 -0.518 0.749 0.384 0.213 -0.567 0.740 0.320 0.105 -0.444 0.690 0.491 -0.004 -0.500 0.720 0.477
3) 0.214 -0.418 0.680 0.466 0.133 -0.469 0.672 0.428 0.003 -0.364 0.631 0.552 -0.025 -0.402 0.654 0.532
4) 0.143 -0.366 0.626 0.477 0.080 -0.449 0.647 0.415 -0.022 -0.305 0.603 0.590 -0.111 -0.353 0.589 0.545
5) 0.133 -0.337 0.627 0.486 0.039 -0.448 0.661 0.399 -0.111 -0.312 0.612 0.609 -0.111 -0.346 0.591 0.545
6) 0.224 -0.482 0.738 0.372 0.339 -0.559 0.763 0.257 0.207 -0.469 0.732 0.453 -0.191 -0.414 0.678 0.581
7) 0.299 -0.498 0.740 0.373 0.329 -0.574 0.747 0.277 0.169 -0.415 0.675 0.487 -0.076 -0.414 0.679 0.563
8) 0.221 -0.405 0.672 0.473 0.249 -0.486 0.686 0.384 0.067 -0.333 0.625 0.548 -0.093 -0.329 0.621 0.605
9) 0.101 -0.344 0.614 0.493 0.157 -0.447 0.652 0.394 0.048 -0.272 0.599 0.577 -0.200 -0.275 0.570 0.620
10) 0.091 -0.299 0.609 0.503 0.114 -0.424 0.661 0.391 -0.022 -0.268 0.611 0.590 -0.209 -0.252 0.570 0.627
11) 0.286 -0.461 0.725 0.371 0.491 -0.540 0.734 0.204 0.226 -0.427 0.720 0.470 -0.215 -0.350 0.637 0.619
12) 0.367 -0.494 0.737 0.351 0.481 -0.561 0.730 0.223 0.188 -0.379 0.665 0.504 -0.096 -0.379 0.661 0.589
13) 0.273 -0.390 0.665 0.454 0.385 -0.470 0.674 0.332 0.086 -0.284 0.616 0.565 -0.126 -0.287 0.607 0.651
14) 0.114 -0.311 0.596 0.477 0.252 -0.400 0.634 0.376 0.086 -0.219 0.589 0.588 -0.257 -0.221 0.549 0.675
15) 0.094 -0.273 0.600 0.492 0.210 -0.377 0.645 0.361 0.035 -0.197 0.600 0.596 -0.259 -0.173 0.544 0.686
16) 0.309 -0.481 0.733 0.323 0.648 -0.536 0.729 0.129 0.353 -0.459 0.735 0.416 -0.242 -0.322 0.619 0.669
17) 0.413 -0.508 0.745 0.305 0.648 -0.554 0.731 0.158 0.308 -0.412 0.688 0.460 -0.120 -0.349 0.645 0.627
18) 0.318 -0.390 0.679 0.407 0.530 -0.450 0.665 0.288 0.213 -0.285 0.637 0.512 -0.141 -0.238 0.587 0.682
19) 0.130 -0.264 0.587 0.449 0.358 -0.366 0.600 0.352 0.175 -0.215 0.582 0.546 -0.271 -0.125 0.509 0.705
20) 0.107 -0.181 0.571 0.475 0.300 -0.305 0.598 0.363 0.118 -0.136 0.578 0.563 -0.271 -0.072 0.513 0.705
21) 0.331 -0.492 0.729 0.300 0.723 -0.535 0.726 0.071 0.270 -0.466 0.735 0.429 -0.363 -0.279 0.588 0.735
22) 0.416 -0.522 0.766 0.294 0.706 -0.555 0.730 0.101 0.245 -0.422 0.699 0.467 -0.265 -0.310 0.617 0.703
23) 0.338 -0.411 0.717 0.370 0.615 -0.453 0.673 0.214 0.169 -0.270 0.643 0.513 -0.289 -0.167 0.554 0.769
24) 0.146 -0.249 0.623 0.422 0.414 -0.343 0.597 0.306 0.124 -0.172 0.587 0.556 -0.423 -0.022 0.478 0.805
25) 0.078 -0.120 0.586 0.469 0.288 -0.241 0.584 0.343 0.003 -0.043 0.564 0.600 -0.431 0.099 0.441 0.812
26) 0.071 -0.567 0.750 0.418 0.092 -0.586 0.716 0.253 0.156 -0.528 0.719 0.506 -0.043 -0.576 0.759 0.475
27) 0.084 -0.554 0.746 0.427 0.128 -0.574 0.704 0.243 0.194 -0.507 0.718 0.538 -0.082 -0.575 0.753 0.458
28) 0.097 -0.544 0.739 0.385 0.126 -0.551 0.682 0.247 0.162 -0.473 0.707 0.562 -0.073 -0.559 0.732 0.422
29) 0.094 -0.515 0.713 0.370 0.135 -0.549 0.701 0.228 0.086 -0.455 0.676 0.542 -0.073 -0.538 0.706 0.394
30) -0.013 -0.496 0.699 0.392 0.031 -0.523 0.681 0.290 0.010 -0.440 0.652 0.588 -0.153 -0.499 0.659 0.430
31) 0.023 -0.569 0.737 0.425 0.070 -0.581 0.711 0.275 0.258 -0.523 0.723 0.491 -0.052 -0.580 0.757 0.453
32) 0.036 -0.553 0.732 0.433 0.116 -0.570 0.691 0.245 0.315 -0.494 0.719 0.517 -0.082 -0.570 0.743 0.430
33) 0.045 -0.545 0.729 0.402 0.126 -0.542 0.676 0.247 0.315 -0.469 0.716 0.517 -0.079 -0.552 0.723 0.418
34) 0.071 -0.534 0.721 0.368 0.145 -0.530 0.686 0.208 0.270 -0.468 0.695 0.472 -0.058 -0.545 0.713 0.363
35) -0.006 -0.521 0.712 0.371 0.102 -0.514 0.671 0.226 0.200 -0.469 0.681 0.508 -0.126 -0.507 0.677 0.408
36) -0.019 -0.556 0.728 0.438 0.036 -0.573 0.705 0.273 0.258 -0.514 0.716 0.491 -0.076 -0.550 0.721 0.463
37) -0.006 -0.543 0.718 0.446 0.075 -0.551 0.676 0.251 0.321 -0.481 0.713 0.507 -0.105 -0.543 0.707 0.439
38) 0.003 -0.535 0.713 0.415 0.075 -0.513 0.656 0.272 0.315 -0.458 0.710 0.517 -0.099 -0.521 0.680 0.415
39) 0.019 -0.522 0.704 0.388 0.082 -0.499 0.662 0.265 0.251 -0.459 0.691 0.477 -0.070 -0.522 0.679 0.354
40) -0.006 -0.519 0.707 0.371 0.094 -0.505 0.663 0.262 0.200 -0.462 0.681 0.508 -0.117 -0.499 0.661 0.401
41) -0.019 -0.553 0.718 0.438 0.007 -0.583 0.714 0.248 0.232 -0.498 0.703 0.506 -0.046 -0.567 0.729 0.451
42) -0.013 -0.539 0.707 0.442 0.017 -0.560 0.688 0.229 0.302 -0.470 0.703 0.513 -0.052 -0.554 0.718 0.418
43) 0.006 -0.521 0.693 0.405 0.027 -0.512 0.647 0.231 0.315 -0.445 0.690 0.517 -0.019 -0.530 0.694 0.373
44) 0.023 -0.514 0.689 0.377 0.034 -0.497 0.651 0.223 0.270 -0.458 0.686 0.472 0.007 -0.525 0.679 0.323
45) 0.019 -0.519 0.695 0.363 0.058 -0.496 0.647 0.219 0.213 -0.451 0.671 0.512 -0.040 -0.515 0.686 0.371
46) -0.104 -0.522 0.686 0.487 -0.044 -0.575 0.702 0.273 0.181 -0.462 0.658 0.537 -0.120 -0.537 0.683 0.467
47) -0.114 -0.492 0.662 0.494 -0.073 -0.536 0.669 0.276 0.175 -0.422 0.642 0.546 -0.099 -0.530 0.682 0.412
48) -0.084 -0.466 0.639 0.449 -0.116 -0.467 0.601 0.294 0.137 -0.385 0.607 0.558 -0.067 -0.500 0.656 0.367
49) -0.052 -0.466 0.634 0.395 -0.121 -0.452 0.595 0.289 0.086 -0.406 0.602 0.522 -0.040 -0.505 0.644 0.317
50) -0.071 -0.475 0.650 0.408 -0.097 -0.468 0.615 0.284 0.048 -0.416 0.608 0.556 -0.093 -0.499 0.652 0.388
51) 0.042 -0.580 0.781 0.358 0.094 -0.635 0.822 0.204 -0.175 -0.527 0.736 0.547 -0.328 -0.530 0.706 0.546
52) -0.068 -0.601 0.785 0.372 0.015 -0.654 0.831 0.208 -0.232 -0.544 0.730 0.541 -0.390 -0.541 0.709 0.596
53) -0.088 -0.604 0.795 0.384 -0.019 -0.640 0.821 0.260 -0.181 -0.569 0.762 0.511 -0.337 -0.582 0.736 0.496
54) 0.032 -0.605 0.807 0.361 0.019 -0.611 0.807 0.313 -0.162 -0.566 0.751 0.482 -0.206 -0.599 0.771 0.398
55) 0.016 -0.565 0.757 0.363 -0.053 -0.547 0.746 0.414 -0.143 -0.531 0.724 0.476 -0.162 -0.566 0.732 0.362
56) -0.016 -0.586 0.778 0.330 -0.010 -0.609 0.802 0.273 -0.149 -0.506 0.724 0.532 -0.325 -0.535 0.709 0.534
57) -0.058 -0.610 0.789 0.340 -0.031 -0.644 0.827 0.266 -0.213 -0.528 0.721 0.535 -0.363 -0.554 0.714 0.575
58) -0.075 -0.617 0.802 0.342 -0.065 -0.631 0.815 0.319 -0.143 -0.561 0.760 0.499 -0.313 -0.584 0.745 0.487
59) 0.065 -0.607 0.802 0.329 -0.010 -0.594 0.787 0.371 -0.175 -0.544 0.737 0.501 -0.162 -0.601 0.778 0.387
60) 0.036 -0.554 0.734 0.348 -0.077 -0.531 0.711 0.447 -0.156 -0.501 0.701 0.495 -0.141 -0.561 0.729 0.361
61) -0.094 -0.551 0.742 0.358 -0.058 -0.573 0.764 0.315 -0.048 -0.496 0.715 0.493 -0.251 -0.548 0.725 0.462
62) -0.110 -0.590 0.767 0.360 -0.092 -0.621 0.802 0.310 -0.137 -0.524 0.716 0.513 -0.262 -0.578 0.754 0.481

63) -0.114 -0.599 0.771 0.345 -0.126 -0.615 0.797 0.363 -0.092 -0.553 0.753 0.492 -0.227 -0.603 0.775 0.424
64) 0.042 -0.593 0.777 0.330 -0.031 -0.590 0.783 0.396 -0.130 -0.534 0.737 0.503 -0.082 -0.610 0.779 0.345
65) 0.010 -0.550 0.728 0.359 -0.097 -0.529 0.708 0.460 -0.111 -0.494 0.709 0.497 -0.061 -0.585 0.745 0.319
66) -0.127 -0.509 0.699 0.387 -0.133 -0.532 0.713 0.374 -0.080 -0.468 0.686 0.518 -0.233 -0.525 0.703 0.448
67) -0.133 -0.564 0.740 0.383 -0.174 -0.583 0.750 0.376 -0.169 -0.505 0.696 0.537 -0.236 -0.573 0.745 0.460
68) -0.136 -0.591 0.765 0.369 -0.210 -0.587 0.752 0.441 -0.149 -0.529 0.719 0.532 -0.209 -0.605 0.773 0.410
69) 0.023 -0.589 0.776 0.342 -0.085 -0.580 0.756 0.432 -0.181 -0.520 0.713 0.534 -0.064 -0.613 0.780 0.331
70) 0.0 -0.560 0.734 0.365 -0.135 -0.529 0.700 0.483 -0.156 -0.487 0.692 0.518 -0.043 -0.595 0.756 0.304
71) -0.136 -0.497 0.682 0.419 -0.177 -0.494 0.674 0.391 -0.080 -0.447 0.665 0.518 -0.227 -0.509 0.689 0.452
72) -0.127 -0.561 0.731 0.387 -0.218 -0.547 0.708 0.394 -0.137 -0.502 0.685 0.513 -0.215 -0.569 0.736 0.433
73) -0.140 -0.588 0.753 0.379 -0.247 -0.561 0.715 0.451 -0.118 -0.526 0.707 0.507 -0.191 -0.611 0.767 0.395
74) 0.029 -0.590 0.760 0.347 -0.111 -0.572 0.738 0.424 -0.156 -0.521 0.704 0.518 -0.058 -0.622 0.777 0.335
75) 0.032 -0.569 0.745 0.361 -0.140 -0.538 0.708 0.453 -0.156 -0.494 0.686 0.518 -0.034 -0.608 0.757 0.297
76) 0.224 -0.538 0.705 0.334 0.143 -0.582 0.698 0.258 -0.010 -0.484 0.660 0.528 -0.093 -0.528 0.653 0.491
77) 0.175 -0.503 0.684 0.391 0.119 -0.571 0.691 0.292 -0.054 -0.477 0.658 0.549 -0.114 -0.515 0.635 0.489
78) 0.201 -0.510 0.686 0.332 0.128 -0.572 0.696 0.273 -0.041 -0.470 0.656 0.530 -0.082 -0.524 0.657 0.472
79) 0.195 -0.484 0.674 0.378 0.097 -0.532 0.676 0.313 -0.010 -0.400 0.623 0.551 -0.076 -0.486 0.638 0.506
80) 0.140 -0.465 0.691 0.430 0.053 -0.510 0.695 0.331 0.048 -0.400 0.651 0.554 -0.090 -0.477 0.655 0.508
81) 0.257 -0.549 0.713 0.330 0.152 -0.573 0.691 0.243 -0.035 -0.487 0.650 0.497 -0.070 -0.538 0.665 0.453
82) 0.211 -0.519 0.697 0.376 0.106 -0.556 0.678 0.299 -0.092 -0.469 0.641 0.537 -0.093 -0.514 0.647 0.463
83) 0.205 -0.520 0.700 0.347 0.080 -0.561 0.687 0.290 -0.137 -0.455 0.629 0.535 -0.061 -0.527 0.675 0.446
84) 0.205 -0.487 0.687 0.372 0.048 -0.519 0.668 0.331 -0.118 -0.376 0.602 0.576 -0.061 -0.476 0.652 0.503
85) 0.120 -0.446 0.679 0.443 0.005 -0.485 0.672 0.348 -0.067 -0.360 0.610 0.588 -0.085 -0.455 0.651 0.513
86) 0.305 -0.546 0.715 0.295 0.177 -0.561 0.681 0.231 0.003 -0.480 0.657 0.509 -0.108 -0.496 0.642 0.473
87) 0.260 -0.523 0.701 0.342 0.131 -0.560 0.680 0.286 -0.054 -0.464 0.646 0.549 -0.120 -0.488 0.632 0.463
88) 0.253 -0.532 0.707 0.312 0.077 -0.564 0.685 0.294 -0.099 -0.452 0.636 0.547 -0.085 -0.500 0.665 0.435
89) 0.224 -0.489 0.699 0.356 0.027 -0.521 0.663 0.345 -0.124 -0.378 0.608 0.585 -0.073 -0.440 0.639 0.473
90) 0.149 -0.439 0.679 0.421 0.019 -0.478 0.651 0.352 -0.048 -0.355 0.602 0.582 -0.102 -0.407 0.626 0.506
91) 0.338 -0.568 0.737 0.292 0.244 -0.562 0.700 0.234 0.041 -0.502 0.688 0.497 -0.123 -0.517 0.662 0.426
92) 0.279 -0.553 0.729 0.333 0.155 -0.561 0.702 0.307 -0.048 -0.495 0.683 0.539 -0.108 -0.528 0.660 0.395
93) 0.279 -0.550 0.733 0.308 0.102 -0.569 0.712 0.315 -0.099 -0.477 0.669 0.547 -0.070 -0.534 0.681 0.354
94) 0.237 -0.485 0.711 0.368 0.012 -0.512 0.680 0.388 -0.149 -0.391 0.632 0.600 -0.064 -0.480 0.661 0.416
95) 0.159 -0.403 0.674 0.444 -0.034 -0.459 0.651 0.414 -0.188 -0.358 0.595 0.612 -0.120 -0.415 0.629 0.471
96) 0.354 -0.564 0.726 0.271 0.305 -0.568 0.709 0.223 0.067 -0.520 0.699 0.482 -0.090 -0.500 0.644 0.431
97) 0.305 -0.564 0.724 0.306 0.186 -0.580 0.721 0.299 -0.029 -0.509 0.693 0.534 -0.079 -0.535 0.666 0.412
98) 0.276 -0.561 0.731 0.300 0.094 -0.580 0.725 0.330 -0.137 -0.477 0.671 0.558 -0.037 -0.543 0.689 0.360
99) 0.237 -0.481 0.699 0.350 0.002 -0.505 0.687 0.415 -0.251 -0.368 0.614 0.638 -0.058 -0.470 0.661 0.443
100) 0.146 -0.376 0.654 0.442 -0.073 -0.422 0.647 0.470 -0.321 -0.310 0.566 0.675 -0.141 -0.372 0.608 0.519
101) 0.305 -0.541 0.761 0.306 0.082 -0.510 0.727 0.322 -0.118 -0.421 0.649 0.550 0.096 -0.567 0.739 0.355
102) 0.283 -0.443 0.696 0.329 0.077 -0.448 0.684 0.317 -0.118 -0.412 0.642 0.527 0.055 -0.502 0.695 0.379
103) 0.140 -0.395 0.655 0.394 -0.046 -0.423 0.648 0.385 -0.149 -0.368 0.591 0.552 -0.019 -0.478 0.664 0.340
104) 0.052 -0.430 0.667 0.406 -0.090 -0.445 0.659 0.432 -0.149 -0.384 0.613 0.575 -0.076 -0.510 0.672 0.370
105) 0.019 -0.444 0.679 0.413 -0.111 -0.465 0.665 0.404 -0.232 -0.377 0.593 0.630 -0.117 -0.517 0.664 0.409
106) 0.263 -0.515 0.743 0.344 0.010 -0.463 0.692 0.380 -0.092 -0.408 0.652 0.535 0.082 -0.556 0.735 0.386
107) 0.240 -0.423 0.694 0.368 0.007 -0.419 0.670 0.363 -0.105 -0.383 0.641 0.531 0.037 -0.484 0.691 0.421
108) 0.104 -0.354 0.646 0.436 -0.094 -0.371 0.626 0.409 -0.143 -0.317 0.585 0.565 -0.022 -0.449 0.662 0.381
109) 0.003 -0.377 0.652 0.465 -0.148 -0.403 0.637 0.446 -0.137 -0.337 0.599 0.579 -0.082 -0.466 0.662 0.422
110) -0.039 -0.384 0.653 0.457 -0.148 -0.430 0.664 0.418 -0.194 -0.345 0.586 0.619 -0.159 -0.465 0.643 0.464
111) 0.182 -0.489 0.707 0.336 -0.075 -0.490 0.708 0.378 -0.092 -0.409 0.651 0.535 0.073 -0.560 0.746 0.393
112) 0.153 -0.386 0.661 0.380 -0.082 -0.421 0.675 0.385 -0.105 -0.378 0.639 0.531 0.022 -0.474 0.699 0.452
113) 0.062 -0.296 0.611 0.450 -0.150 -0.346 0.625 0.433 -0.143 -0.305 0.586 0.565 0.004 -0.419 0.666 0.413
114) -0.049 -0.312 0.618 0.485 -0.213 -0.330 0.616 0.489 -0.130 -0.327 0.603 0.569 -0.064 -0.416 0.656 0.461
115) -0.081 -0.323 0.631 0.470 -0.210 -0.354 0.635 0.448 -0.181 -0.320 0.589 0.600 -0.153 -0.404 0.617 0.498
116) 0.117 -0.502 0.701 0.347 -0.099 -0.524 0.719 0.357 -0.143 -0.433 0.653 0.565 -0.019 -0.577 0.755 0.422
117) 0.084 -0.424 0.671 0.402 -0.099 -0.468 0.692 0.357 -0.130 -0.422 0.652 0.546 -0.064 -0.524 0.722 0.458
118) 0.042 -0.357 0.640 0.462 -0.152 -0.417 0.662 0.391 -0.162 -0.364 0.614 0.571 -0.037 -0.483 0.699 0.408
119) -0.068 -0.350 0.637 0.498 -0.225 -0.383 0.647 0.466 -0.143 -0.377 0.626 0.565 -0.096 -0.455 0.676 0.450
120) -0.130 -0.355 0.636 0.479 -0.259 -0.397 0.651 0.436 -0.200 -0.349 0.606 0.605 -0.182 -0.453 0.652 0.474
121) 0.042 -0.504 0.701 0.365 -0.031 -0.543 0.733 0.352 -0.022 -0.456 0.684 0.521 -0.085 -0.574 0.743 0.438
122) 0.013 -0.482 0.697 0.409 -0.000 -0.535 0.731 0.312 0.048 -0.474 0.699 0.485 -0.102 -0.553 0.729 0.452
123) -0.013 -0.438 0.673 0.467 -0.012 -0.512 0.724 0.314 0.118 -0.448 0.684 0.492 -0.031 -0.531 0.724 0.392
124) -0.127 -0.408 0.656 0.513 -0.060 -0.487 0.711 0.356 0.194 -0.470 0.714 0.469 -0.052 -0.508 0.710 0.418
125) -0.218 -0.407 0.642 0.514 -0.114 -0.491 0.704 0.335 0.143 -0.450 0.696 0.499 -0.076 -0.508 0.696 0.439
126) 0.133 -0.482 0.744 0.344 0.007 -0.530 0.731 0.247 -0.277 -0.385 0.642 0.585 -0.111 -0.426 0.686 0.434
127) 0.237 -0.451 0.728 0.323 0.109 -0.506 0.721 0.185 -0.232 -0.348 0.634 0.564 -0.087 -0.366 0.664 0.450
128) 0.296 -0.398 0.700 0.285 0.235 -0.471 0.710 0.148 -0.181 -0.303 0.628 0.556 -0.099 -0.262 0.624 0.469

129) 0.257 -0.356 0.701 0.361 0.249 -0.453 0.697 0.162 -0.181 -0.277 0.621 0.534 -0.153 -0.238 0.629 0.512
130) 0.221 -0.308 0.694 0.401 0.196 -0.401 0.673 0.254 -0.169 -0.233 0.609 0.537 -0.108 -0.246 0.631 0.476
131) 0.071 -0.509 0.739 0.327 -0.051 -0.559 0.725 0.250 -0.308 -0.433 0.638 0.564 -0.120 -0.454 0.672 0.388
132) 0.169 -0.483 0.732 0.327 0.039 -0.541 0.720 0.220 -0.296 -0.393 0.614 0.568 -0.108 -0.398 0.651 0.422
133) 0.247 -0.442 0.707 0.276 0.150 -0.507 0.705 0.172 -0.226 -0.359 0.615 0.555 -0.120 -0.303 0.611 0.441
134) 0.211 -0.397 0.692 0.341 0.213 -0.494 0.714 0.173 -0.149 -0.345 0.635 0.509 -0.173 -0.285 0.618 0.484
135) 0.162 -0.357 0.694 0.398 0.177 -0.486 0.725 0.238 -0.111 -0.345 0.654 0.497 -0.156 -0.298 0.631 0.470
136) 0.055 -0.543 0.756 0.303 -0.063 -0.578 0.733 0.249 -0.296 -0.473 0.667 0.568 -0.108 -0.496 0.699 0.397
137) 0.162 -0.531 0.762 0.297 0.027 -0.565 0.732 0.219 -0.308 -0.436 0.638 0.587 -0.096 -0.446 0.688 0.432
138) 0.208 -0.486 0.736 0.276 0.087 -0.535 0.719 0.174 -0.270 -0.408 0.635 0.576 -0.105 -0.349 0.647 0.439
139) 0.185 -0.459 0.719 0.324 0.128 -0.527 0.732 0.197 -0.181 -0.416 0.663 0.511 -0.159 -0.359 0.656 0.482
140) 0.149 -0.459 0.745 0.364 0.114 -0.545 0.759 0.241 -0.137 -0.426 0.685 0.490 -0.147 -0.408 0.684 0.463
141) 0.120 -0.573 0.780 0.292 -0.034 -0.588 0.726 0.249 -0.277 -0.498 0.673 0.562 -0.090 -0.559 0.743 0.354
142) 0.205 -0.559 0.780 0.284 0.034 -0.573 0.718 0.219 -0.232 -0.465 0.651 0.564 -0.093 -0.508 0.718 0.392
143) 0.257 -0.520 0.756 0.267 0.114 -0.542 0.708 0.165 -0.194 -0.438 0.653 0.553 -0.117 -0.414 0.673 0.430
144) 0.201 -0.490 0.733 0.345 0.121 -0.529 0.722 0.187 -0.105 -0.441 0.677 0.488 -0.191 -0.409 0.671 0.499
145) 0.169 -0.509 0.765 0.374 0.106 -0.559 0.764 0.230 -0.060 -0.465 0.713 0.467 -0.168 -0.483 0.716 0.461
146) 0.159 -0.584 0.777 0.311 -0.034 -0.607 0.738 0.225 -0.226 -0.505 0.673 0.555 0.001 -0.580 0.764 0.343
147) 0.169 -0.566 0.765 0.301 -0.024 -0.586 0.715 0.206 -0.181 -0.477 0.651 0.556 -0.049 -0.532 0.735 0.402
148) 0.221 -0.536 0.746 0.285 0.085 -0.559 0.706 0.149 -0.111 -0.469 0.662 0.520 -0.073 -0.457 0.702 0.440
149) 0.162 -0.508 0.731 0.347 0.080 -0.561 0.733 0.173 -0.041 -0.469 0.692 0.461 -0.123 -0.435 0.698 0.499
150) 0.123 -0.538 0.780 0.373 0.065 -0.583 0.762 0.216 -0.016 -0.503 0.731 0.446 -0.138 -0.513 0.750 0.477
151) 0.123 -0.493 0.740 0.348 0.421 -0.595 0.762 0.068 0.092 -0.512 0.706 0.418 -0.381 -0.424 0.648 0.649
152) 0.273 -0.533 0.778 0.309 0.484 -0.612 0.785 0.062 0.060 -0.553 0.742 0.443 -0.280 -0.461 0.680 0.613
153) 0.052 -0.537 0.768 0.377 0.269 -0.624 0.790 0.146 0.022 -0.565 0.742 0.454 -0.399 -0.465 0.657 0.664
154) 0.032 -0.530 0.744 0.367 0.155 -0.606 0.774 0.206 0.022 -0.562 0.743 0.454 -0.316 -0.491 0.674 0.584
155) -0.010 -0.518 0.714 0.428 0.044 -0.586 0.738 0.254 -0.041 -0.523 0.693 0.481 -0.295 -0.487 0.653 0.587
156) 0.149 -0.512 0.753 0.340 0.479 -0.590 0.772 0.064 0.086 -0.515 0.714 0.428 -0.363 -0.439 0.652 0.607
157) 0.322 -0.544 0.790 0.300 0.581 -0.605 0.795 0.057 0.060 -0.548 0.746 0.443 -0.242 -0.471 0.692 0.569
158) 0.104 -0.542 0.773 0.357 0.365 -0.607 0.786 0.141 0.022 -0.542 0.729 0.454 -0.375 -0.465 0.665 0.651
159) 0.101 -0.531 0.749 0.345 0.259 -0.577 0.755 0.219 0.022 -0.543 0.737 0.454 -0.283 -0.478 0.677 0.564
160) 0.052 -0.507 0.705 0.402 0.135 -0.546 0.706 0.270 -0.035 -0.506 0.682 0.472 -0.277 -0.452 0.632 0.598
161) 0.156 -0.524 0.761 0.344 0.402 -0.580 0.787 0.106 -0.060 -0.489 0.686 0.490 -0.372 -0.426 0.648 0.585
162) 0.315 -0.546 0.795 0.296 0.508 -0.588 0.804 0.078 -0.060 -0.516 0.716 0.490 -0.248 -0.461 0.710 0.536
163) 0.114 -0.537 0.773 0.351 0.302 -0.583 0.788 0.172 -0.099 -0.493 0.682 0.501 -0.390 -0.440 0.671 0.625
164) 0.107 -0.523 0.747 0.350 0.225 -0.549 0.753 0.221 -0.099 -0.499 0.693 0.501 -0.357 -0.447 0.669 0.579
165) 0.062 -0.499 0.699 0.396 0.140 -0.522 0.696 0.249 -0.143 -0.467 0.637 0.499 -0.378 -0.414 0.620 0.634
166) 0.149 -0.512 0.758 0.365 0.377 -0.587 0.798 0.111 -0.067 -0.486 0.688 0.476 -0.366 -0.405 0.652 0.590
167) 0.292 -0.530 0.789 0.319 0.479 -0.590 0.806 0.078 -0.073 -0.500 0.709 0.486 -0.254 -0.421 0.702 0.559
168) 0.107 -0.519 0.763 0.372 0.315 -0.581 0.787 0.166 -0.086 -0.477 0.674 0.482 -0.393 -0.405 0.665 0.637
169) 0.075 -0.505 0.744 0.379 0.230 -0.555 0.760 0.223 -0.080 -0.485 0.694 0.472 -0.396 -0.399 0.649 0.620
170) 0.029 -0.481 0.706 0.425 0.191 -0.530 0.710 0.221 -0.022 -0.458 0.652 0.453 -0.414 -0.375 0.619 0.663
171) 0.127 -0.500 0.756 0.363 0.278 -0.557 0.781 0.203 -0.239 -0.439 0.657 0.551 -0.357 -0.389 0.648 0.554
172) 0.260 -0.505 0.777 0.323 0.385 -0.563 0.793 0.146 -0.245 -0.452 0.677 0.560 -0.254 -0.408 0.695 0.531
173) 0.117 -0.492 0.748 0.365 0.269 -0.558 0.783 0.221 -0.258 -0.439 0.655 0.556 -0.378 -0.384 0.650 0.606
174) 0.081 -0.480 0.732 0.383 0.203 -0.536 0.752 0.268 -0.200 -0.452 0.677 0.539 -0.375 -0.385 0.649 0.594
175) 0.039 -0.465 0.703 0.419 0.165 -0.522 0.724 0.266 -0.149 -0.437 0.653 0.529 -0.396 -0.364 0.633 0.648
176) 0.315 -0.544 0.737 0.173 0.094 -0.540 0.698 0.155 -0.105 -0.456 0.658 0.511 -0.132 -0.514 0.688 0.408
177) 0.286 -0.572 0.766 0.217 0.073 -0.578 0.727 0.176 -0.099 -0.498 0.681 0.478 -0.132 -0.546 0.714 0.408
178) 0.270 -0.569 0.769 0.244 0.058 -0.571 0.721 0.191 -0.137 -0.496 0.674 0.490 -0.141 -0.536 0.710 0.415
179) 0.201 -0.532 0.738 0.288 0.044 -0.551 0.698 0.205 -0.124 -0.471 0.658 0.471 -0.236 -0.481 0.666 0.538
180) -0.023 -0.503 0.683 0.370 -0.048 -0.532 0.670 0.236 -0.105 -0.466 0.654 0.442 -0.402 -0.419 0.589 0.608
181) 0.240 -0.546 0.724 0.197 0.162 -0.563 0.732 0.153 0.010 -0.492 0.699 0.474 -0.188 -0.517 0.691 0.434
182) 0.198 -0.564 0.743 0.233 0.126 -0.599 0.755 0.164 0.016 -0.519 0.711 0.441 -0.188 -0.538 0.708 0.434
183) 0.182 -0.563 0.742 0.260 0.111 -0.597 0.751 0.178 -0.022 -0.522 0.706 0.453 -0.191 -0.544 0.709 0.417
184) 0.110 -0.539 0.719 0.315 0.075 -0.583 0.736 0.214 -0.010 -0.496 0.687 0.433 -0.277 -0.496 0.664 0.534
185) -0.058 -0.508 0.674 0.410 0.019 -0.559 0.706 0.234 0.029 -0.491 0.686 0.399 -0.420 -0.442 0.589 0.594
186) 0.195 -0.528 0.703 0.193 0.191 -0.592 0.756 0.099 0.010 -0.490 0.704 0.474 -0.233 -0.521 0.692 0.416
187) 0.140 -0.532 0.707 0.246 0.135 -0.618 0.767 0.119 0.016 -0.513 0.711 0.441 -0.242 -0.534 0.700 0.423
188) 0.114 -0.531 0.713 0.279 0.114 -0.612 0.761 0.141 -0.029 -0.514 0.701 0.462 -0.248 -0.535 0.696 0.418
189) 0.042 -0.507 0.690 0.334 0.077 -0.590 0.741 0.177 -0.010 -0.497 0.685 0.433 -0.322 -0.500 0.655 0.516
190) -0.127 -0.473 0.643 0.429 0.005 -0.567 0.706 0.195 0.022 -0.494 0.686 0.409 -0.452 -0.450 0.592 0.586
191) 0.224 -0.542 0.723 0.196 0.191 -0.593 0.755 0.103 0.092 -0.515 0.708 0.441 -0.147 -0.551 0.692 0.324
192) 0.117 -0.545 0.719 0.269 0.128 -0.612 0.758 0.130 0.099 -0.534 0.711 0.409 -0.212 -0.549 0.690 0.361
193) 0.094 -0.534 0.715 0.292 0.106 -0.601 0.748 0.152 0.035 -0.527 0.699 0.435 -0.233 -0.531 0.678 0.387
194) 0.023 -0.508 0.689 0.346 0.060 -0.578 0.728 0.207 0.054 -0.504 0.678 0.407 -0.307 -0.491 0.642 0.485
195) -0.136 -0.468 0.637 0.435 -0.012 -0.563 0.697 0.224 0.067 -0.498 0.674 0.388 -0.417 -0.462 0.599 0.557
196) 0.244 -0.540 0.711 0.206 0.157 -0.584 0.732 0.127 0.073 -0.492 0.688 0.470 -0.129 -0.543 0.681 0.310
197) 0.088 -0.540 0.708 0.287 0.056 -0.602 0.742 0.176 0.073 -0.515 0.691 0.447 -0.242 -0.523 0.664 0.369
198) 0.068 -0.526 0.700 0.300 0.034 -0.584 0.725 0.198 0.035 -0.518 0.689 0.458 -0.265 -0.508 0.659 0.407
199) 0.026 -0.504 0.675 0.336 -0.010 -0.564 0.709 0.241 0.054 -0.501 0.671 0.430 -0.331 -0.468 0.625 0.498
200) -0.107 -0.477 0.635 0.416 -0.082 -0.549 0.675 0.259 0.073 -0.502 0.672 0.401 -0.387 -0.454 0.603 0.552

Woodgrain

1) -0.015 -0.624 0.448 0.288 -0.067 -0.626 0.433 0.260 -0.199 -0.484 0.404 0.698 0.023 -0.635 0.400 0.239
2) -0.038 -0.633 0.458 0.367 -0.098 -0.634 0.448 0.351 -0.245 -0.478 0.393 0.715 0.007 -0.656 0.437 0.309
3) -0.036 -0.651 0.462 0.358 -0.098 -0.654 0.450 0.351 -0.199 -0.525 0.436 0.698 0.005 -0.669 0.437 0.319
4) -0.026 -0.658 0.461 0.334 -0.078 -0.669 0.452 0.309 -0.269 -0.507 0.410 0.723 0.014 -0.681 0.442 0.279
5) -0.028 -0.662 0.469 0.343 -0.071 -0.677 0.465 0.303 -0.269 -0.520 0.425 0.723 -0.002 -0.678 0.439 0.325
6) -0.019 -0.678 0.493 0.306 -0.060 -0.689 0.489 0.277 -0.222 -0.535 0.440 0.707 0.007 -0.685 0.460 0.309
7) -0.356 -0.651 0.416 0.431 -0.370 -0.667 0.418 0.392 -0.199 -0.560 0.481 0.698 -0.342 -0.657 0.399 0.440
8) -0.352 -0.627 0.372 0.435 -0.357 -0.649 0.393 0.380 -0.199 -0.551 0.442 0.698 -0.340 -0.636 0.350 0.430
9) -0.625 -0.628 0.379 0.504 -0.605 -0.650 0.403 0.438 -0.175 -0.561 0.482 0.690 -0.635 -0.639 0.372 0.500
10) -0.547 -0.674 0.491 0.398 -0.535 -0.704 0.523 0.312 -0.152 -0.619 0.622 0.681 -0.542 -0.682 0.499 0.402
11) -0.584 -0.699 0.547 0.453 -0.573 -0.728 0.585 0.361 -0.269 -0.625 0.624 0.723 -0.580 -0.695 0.538 0.476
12) -0.422 -0.748 0.640 0.298 -0.408 -0.768 0.673 0.210 -0.245 -0.663 0.671 0.715 -0.422 -0.742 0.631 0.326
13) -0.420 -0.749 0.641 0.289 -0.408 -0.768 0.666 0.210 -0.222 -0.654 0.666 0.707 -0.410 -0.748 0.646 0.300
14) -0.416 -0.740 0.596 0.270 -0.412 -0.758 0.622 0.206 -0.245 -0.624 0.585 0.715 -0.401 -0.739 0.592 0.284
15) -0.414 -0.741 0.585 0.261 -0.412 -0.755 0.601 0.182 -0.245 -0.609 0.552 0.715 -0.394 -0.749 0.603 0.254
16) -0.412 -0.730 0.573 0.252 -0.416 -0.744 0.573 0.178 -0.339 -0.575 0.517 0.749 -0.394 -0.743 0.592 0.230
17) -0.424 -0.702 0.548 0.307 -0.432 -0.724 0.553 0.248 -0.409 -0.537 0.475 0.774 -0.408 -0.725 0.563 0.266
18) -0.253 -0.733 0.619 0.179 -0.296 -0.745 0.603 0.130 -0.409 -0.565 0.523 0.774 -0.215 -0.752 0.638 0.154
19) -0.210 -0.747 0.657 0.138 -0.274 -0.757 0.642 0.102 -0.503 -0.579 0.521 0.807 -0.156 -0.762 0.657 0.108
20) -0.270 -0.735 0.624 0.184 -0.326 -0.742 0.604 0.150 -0.503 -0.556 0.496 0.807 -0.227 -0.753 0.629 0.154
21) 0.009 -0.626 0.412 0.321 -0.016 -0.626 0.405 0.299 -0.269 -0.466 0.358 0.723 0.036 -0.635 0.379 0.290
22) -0.015 -0.621 0.407 0.409 -0.047 -0.626 0.406 0.390 -0.292 -0.472 0.356 0.732 0.018 -0.640 0.391 0.369
23) -0.015 -0.628 0.404 0.409 -0.049 -0.631 0.400 0.400 -0.245 -0.507 0.392 0.715 0.016 -0.648 0.388 0.379
24) 0.003 -0.637 0.407 0.371 -0.030 -0.641 0.398 0.358 -0.269 -0.501 0.378 0.723 0.034 -0.662 0.396 0.324
25) 0.003 -0.642 0.416 0.371 -0.021 -0.652 0.420 0.342 -0.292 -0.503 0.380 0.732 0.020 -0.654 0.384 0.360
26) 0.009 -0.651 0.433 0.343 -0.019 -0.654 0.431 0.332 -0.269 -0.509 0.388 0.723 0.029 -0.660 0.407 0.344
27) -0.342 -0.626 0.370 0.468 -0.364 -0.632 0.374 0.448 -0.222 -0.542 0.447 0.707 -0.320 -0.634 0.353 0.475
28) -0.340 -0.597 0.326 0.481 -0.359 -0.607 0.337 0.452 -0.222 -0.528 0.406 0.707 -0.315 -0.617 0.310 0.455
29) -0.627 -0.599 0.335 0.536 -0.623 -0.609 0.344 0.494 -0.199 -0.541 0.461 0.698 -0.626 -0.620 0.342 0.508
30) -0.543 -0.649 0.462 0.424 -0.546 -0.667 0.472 0.361 -0.175 -0.604 0.606 0.690 -0.533 -0.664 0.469 0.410
31) -0.576 -0.684 0.527 0.460 -0.573 -0.705 0.546 0.385 -0.269 -0.621 0.623 0.723 -0.567 -0.686 0.519 0.464
32) -0.414 -0.739 0.626 0.305 -0.419 -0.754 0.641 0.236 -0.245 -0.668 0.688 0.715 -0.424 -0.739 0.609 0.314
33) -0.409 -0.746 0.639 0.287 -0.416 -0.762 0.662 0.226 -0.199 -0.673 0.702 0.698 -0.408 -0.751 0.642 0.269
34) -0.407 -0.736 0.594 0.278 -0.419 -0.749 0.607 0.212 -0.199 -0.649 0.635 0.698 -0.399 -0.741 0.601 0.253
35) -0.405 -0.741 0.586 0.268 -0.416 -0.753 0.587 0.178 -0.199 -0.638 0.598 0.698 -0.390 -0.757 0.608 0.213
36) -0.401 -0.731 0.576 0.250 -0.412 -0.747 0.579 0.159 -0.269 -0.605 0.563 0.723 -0.392 -0.743 0.587 0.199
37) -0.414 -0.694 0.541 0.305 -0.427 -0.715 0.544 0.228 -0.316 -0.560 0.516 0.740 -0.406 -0.717 0.546 0.235
38) -0.210 -0.735 0.619 0.160 -0.256 -0.742 0.594 0.092 -0.339 -0.577 0.547 0.749 -0.179 -0.743 0.617 0.114
39) -0.217 -0.744 0.643 0.143 -0.287 -0.750 0.613 0.091 -0.432 -0.588 0.539 0.782 -0.175 -0.751 0.625 0.094
40) -0.270 -0.730 0.607 0.184 -0.333 -0.734 0.577 0.132 -0.432 -0.568 0.518 0.782 -0.238 -0.740 0.592 0.134
41) 0.032 -0.619 0.394 0.344 0.008 -0.629 0.397 0.323 -0.269 -0.477 0.363 0.723 0.043 -0.630 0.376 0.325
42) 0.007 -0.608 0.378 0.432 -0.023 -0.620 0.384 0.414 -0.292 -0.483 0.362 0.732 0.025 -0.627 0.374 0.405
43) 0.005 -0.606 0.370 0.441 -0.025 -0.617 0.373 0.424 -0.245 -0.506 0.389 0.715 0.020 -0.625 0.364 0.425
44) 0.022 -0.605 0.361 0.412 -0.005 -0.623 0.369 0.382 -0.269 -0.501 0.372 0.723 0.039 -0.634 0.367 0.369
45) 0.024 -0.608 0.364 0.402 0.005 -0.630 0.375 0.357 -0.316 -0.490 0.358 0.740 0.027 -0.626 0.353 0.395
46) 0.028 -0.611 0.377 0.384 0.008 -0.627 0.387 0.347 -0.316 -0.485 0.355 0.740 0.027 -0.620 0.354 0.395
47) -0.321 -0.591 0.328 0.500 -0.337 -0.607 0.341 0.463 -0.269 -0.518 0.420 0.723 -0.320 -0.598 0.314 0.517
48) -0.327 -0.558 0.269 0.528 -0.342 -0.580 0.288 0.483 -0.292 -0.500 0.358 0.732 -0.317 -0.579 0.269 0.507
49) -0.629 -0.559 0.279 0.567 -0.623 -0.582 0.312 0.517 -0.269 -0.512 0.411 0.723 -0.628 -0.581 0.291 0.542
50) -0.545 -0.612 0.408 0.455 -0.548 -0.639 0.435 0.395 -0.222 -0.581 0.562 0.707 -0.535 -0.625 0.417 0.444
51) -0.568 -0.660 0.484 0.468 -0.564 -0.689 0.521 0.393 -0.316 -0.608 0.590 0.740 -0.558 -0.660 0.480 0.472
52) -0.420 -0.724 0.581 0.313 -0.425 -0.744 0.612 0.244 -0.316 -0.659 0.669 0.740 -0.415 -0.723 0.580 0.322
53) -0.416 -0.741 0.618 0.295 -0.421 -0.756 0.644 0.225 -0.292 -0.668 0.686 0.732 -0.406 -0.744 0.626 0.283
54) -0.414 -0.731 0.582 0.286 -0.432 -0.746 0.595 0.227 -0.316 -0.632 0.609 0.740 -0.394 -0.735 0.590 0.257
55) -0.409 -0.738 0.572 0.267 -0.432 -0.750 0.570 0.203 -0.316 -0.625 0.577 0.740 -0.383 -0.756 0.602 0.207
56) -0.403 -0.729 0.562 0.240 -0.421 -0.745 0.566 0.177 -0.339 -0.596 0.549 0.749 -0.379 -0.746 0.590 0.187
57) -0.418 -0.681 0.511 0.304 -0.434 -0.705 0.519 0.237 -0.339 -0.550 0.498 0.749 -0.392 -0.712 0.530 0.223
58) -0.196 -0.730 0.595 0.142 -0.243 -0.741 0.576 0.083 -0.386 -0.564 0.516 0.765 -0.147 -0.743 0.601 0.092
59) -0.251 -0.737 0.603 0.150 -0.309 -0.742 0.581 0.097 -0.456 -0.583 0.516 0.791 -0.197 -0.748 0.591 0.098
60) -0.299 -0.722 0.566 0.185 -0.348 -0.726 0.545 0.133 -0.456 -0.566 0.499 0.791 -0.254 -0.736 0.556 0.132
61) 0.079 -0.631 0.423 0.376 0.027 -0.636 0.408 0.364 -0.269 -0.497 0.380 0.723 0.077 -0.651 0.399 0.327
62) 0.056 -0.604 0.390 0.455 -0.003 -0.607 0.370 0.455 -0.292 -0.497 0.375 0.732 0.061 -0.644 0.391 0.396
63) 0.054 -0.588 0.376 0.464 -0.005 -0.589 0.350 0.465 -0.292 -0.488 0.372 0.732 0.054 -0.625 0.370 0.426
64) 0.069 -0.571 0.358 0.444 0.012 -0.582 0.340 0.433 -0.292 -0.488 0.361 0.732 0.068 -0.615 0.359 0.390

65) 0.075 -0.577 0.363 0.416 0.025 -0.589 0.347 0.398 -0.316 -0.478 0.351 0.740 0.068 -0.610 0.350 0.390
66) 0.075 -0.567 0.363 0.416 0.027 -0.579 0.353 0.388 -0.316 -0.466 0.344 0.740 0.066 -0.593 0.341 0.400
67) -0.292 -0.547 0.308 0.533 -0.335 -0.560 0.305 0.494 -0.316 -0.484 0.386 0.740 -0.288 -0.571 0.289 0.528
68) -0.301 -0.515 0.246 0.570 -0.340 -0.535 0.252 0.514 -0.339 -0.466 0.323 0.749 -0.288 -0.552 0.241 0.528
69) -0.627 -0.514 0.243 0.602 -0.623 -0.536 0.271 0.541 -0.316 -0.476 0.371 0.740 -0.626 -0.552 0.248 0.556
70) -0.543 -0.572 0.369 0.490 -0.548 -0.599 0.392 0.419 -0.292 -0.538 0.506 0.732 -0.535 -0.593 0.366 0.468
71) -0.555 -0.635 0.456 0.479 -0.553 -0.664 0.485 0.391 -0.386 -0.578 0.549 0.765 -0.546 -0.643 0.444 0.470
72) -0.407 -0.709 0.562 0.324 -0.414 -0.729 0.587 0.242 -0.386 -0.639 0.639 0.765 -0.404 -0.712 0.554 0.320
73) -0.403 -0.735 0.613 0.306 -0.410 -0.752 0.638 0.223 -0.362 -0.662 0.672 0.757 -0.397 -0.740 0.613 0.291
74) -0.401 -0.727 0.582 0.297 -0.414 -0.748 0.604 0.219 -0.362 -0.632 0.620 0.757 -0.385 -0.736 0.589 0.265
75) -0.401 -0.726 0.560 0.297 -0.416 -0.750 0.570 0.205 -0.339 -0.628 0.582 0.749 -0.379 -0.754 0.589 0.235
76) -0.393 -0.720 0.554 0.260 -0.405 -0.745 0.567 0.179 -0.339 -0.598 0.556 0.749 -0.367 -0.742 0.579 0.209
77) -0.407 -0.655 0.484 0.324 -0.419 -0.693 0.500 0.239 -0.316 -0.548 0.496 0.740 -0.374 -0.703 0.526 0.239
78) -0.169 -0.718 0.561 0.152 -0.201 -0.737 0.559 0.082 -0.339 -0.574 0.516 0.749 -0.127 -0.741 0.581 0.098
79) -0.258 -0.717 0.555 0.175 -0.302 -0.731 0.546 0.112 -0.409 -0.591 0.509 0.774 -0.213 -0.739 0.558 0.120
80) -0.299 -0.702 0.518 0.205 -0.335 -0.715 0.511 0.142 -0.409 -0.576 0.494 0.774 -0.263 -0.726 0.521 0.148
81) 0.104 -0.627 0.404 0.408 0.082 -0.654 0.429 0.358 -0.175 -0.535 0.416 0.690 0.113 -0.671 0.412 0.342
82) 0.083 -0.592 0.359 0.478 0.052 -0.618 0.380 0.449 -0.222 -0.525 0.404 0.707 0.097 -0.649 0.389 0.412
83) 0.079 -0.553 0.334 0.496 0.047 -0.577 0.346 0.469 -0.245 -0.492 0.382 0.715 0.091 -0.608 0.357 0.442
84) 0.089 -0.539 0.315 0.472 0.056 -0.549 0.321 0.453 -0.222 -0.497 0.373 0.707 0.097 -0.590 0.334 0.412
85) 0.097 -0.552 0.328 0.436 0.067 -0.546 0.319 0.428 -0.269 -0.473 0.342 0.723 0.097 -0.578 0.307 0.412
86) 0.100 -0.541 0.332 0.426 0.069 -0.535 0.310 0.418 -0.269 -0.456 0.331 0.723 0.097 -0.563 0.306 0.412
87) -0.288 -0.519 0.275 0.553 -0.311 -0.512 0.263 0.535 -0.269 -0.470 0.367 0.723 -0.281 -0.537 0.242 0.539
88) -0.295 -0.490 0.217 0.580 -0.315 -0.484 0.207 0.555 -0.292 -0.453 0.306 0.732 -0.281 -0.518 0.190 0.539
89) -0.623 -0.488 0.209 0.605 -0.625 -0.483 0.207 0.575 -0.269 -0.461 0.344 0.723 -0.621 -0.517 0.192 0.560
90) -0.541 -0.545 0.329 0.503 -0.553 -0.548 0.323 0.462 -0.269 -0.516 0.459 0.723 -0.530 -0.568 0.306 0.472
91) -0.543 -0.621 0.422 0.468 -0.546 -0.629 0.423 0.409 -0.362 -0.569 0.527 0.757 -0.530 -0.631 0.392 0.448
92) -0.395 -0.697 0.536 0.313 -0.408 -0.701 0.534 0.260 -0.362 -0.636 0.636 0.757 -0.388 -0.703 0.517 0.299
93) -0.393 -0.724 0.589 0.304 -0.405 -0.730 0.599 0.250 -0.339 -0.665 0.677 0.749 -0.383 -0.733 0.586 0.279
94) -0.387 -0.719 0.571 0.299 -0.403 -0.731 0.580 0.241 -0.339 -0.639 0.634 0.749 -0.367 -0.736 0.588 0.257
95) -0.389 -0.714 0.546 0.308 -0.408 -0.739 0.555 0.237 -0.339 -0.631 0.588 0.749 -0.363 -0.749 0.584 0.237
96) -0.383 -0.701 0.535 0.280 -0.395 -0.732 0.549 0.201 -0.339 -0.597 0.557 0.749 -0.351 -0.737 0.574 0.211
97) -0.397 -0.624 0.448 0.345 -0.405 -0.676 0.475 0.250 -0.339 -0.532 0.482 0.749 -0.360 -0.688 0.508 0.251
98) -0.134 -0.696 0.519 0.170 -0.162 -0.727 0.531 0.091 -0.362 -0.565 0.502 0.757 -0.086 -0.730 0.554 0.107
99) -0.256 -0.689 0.500 0.207 -0.298 -0.713 0.494 0.137 -0.432 -0.583 0.484 0.782 -0.209 -0.723 0.519 0.146
100) -0.290 -0.674 0.463 0.231 -0.324 -0.698 0.463 0.161 -0.432 -0.570 0.472 0.782 -0.252 -0.710 0.481 0.168
101) 0.141 -0.645 0.412 0.385 0.124 -0.666 0.436 0.348 -0.058 -0.543 0.441 0.668 0.161 -0.676 0.413 0.311
102) 0.120 -0.608 0.370 0.455 0.096 -0.623 0.387 0.429 -0.105 -0.531 0.430 0.685 0.145 -0.656 0.392 0.381
103) 0.116 -0.558 0.340 0.474 0.091 -0.570 0.348 0.449 -0.152 -0.481 0.393 0.701 0.136 -0.603 0.353 0.421
104) 0.118 -0.531 0.309 0.464 0.093 -0.539 0.313 0.439 -0.152 -0.481 0.374 0.701 0.141 -0.577 0.323 0.401
105) 0.122 -0.527 0.288 0.446 0.093 -0.523 0.284 0.439 -0.245 -0.432 0.305 0.735 0.131 -0.552 0.270 0.417
106) 0.120 -0.500 0.266 0.455 0.096 -0.505 0.262 0.429 -0.316 -0.405 0.272 0.740 0.127 -0.526 0.245 0.437
107) -0.286 -0.476 0.213 0.582 -0.304 -0.478 0.205 0.547 -0.316 -0.428 0.308 0.740 -0.274 -0.498 0.173 0.575
108) -0.292 -0.444 0.149 0.609 -0.309 -0.452 0.146 0.566 -0.339 -0.408 0.240 0.749 -0.274 -0.474 0.116 0.575
109) -0.623 -0.441 0.137 0.627 -0.621 -0.449 0.133 0.579 -0.316 -0.412 0.261 0.740 -0.617 -0.472 0.108 0.588
110) -0.543 -0.500 0.245 0.534 -0.551 -0.514 0.236 0.476 -0.339 -0.475 0.371 0.749 -0.537 -0.526 0.217 0.502
111) -0.535 -0.590 0.342 0.475 -0.542 -0.607 0.336 0.413 -0.409 -0.540 0.450 0.774 -0.526 -0.603 0.322 0.452
112) -0.387 -0.667 0.468 0.321 -0.403 -0.681 0.462 0.264 -0.409 -0.613 0.594 0.774 -0.383 -0.675 0.447 0.303
113) -0.387 -0.695 0.525 0.321 -0.403 -0.711 0.539 0.264 -0.386 -0.649 0.647 0.765 -0.381 -0.706 0.532 0.293
114) -0.375 -0.697 0.536 0.310 -0.397 -0.715 0.545 0.258 -0.362 -0.631 0.620 0.757 -0.365 -0.714 0.552 0.271
115) -0.377 -0.698 0.520 0.319 -0.392 -0.734 0.547 0.239 -0.316 -0.638 0.592 0.740 -0.360 -0.736 0.559 0.251
116) -0.366 -0.694 0.525 0.273 -0.381 -0.723 0.539 0.213 -0.269 -0.616 0.576 0.723 -0.347 -0.724 0.551 0.215
117) -0.381 -0.616 0.437 0.337 -0.390 -0.669 0.465 0.252 -0.269 -0.551 0.495 0.723 -0.351 -0.687 0.495 0.235
118) -0.108 -0.688 0.502 0.161 -0.135 -0.719 0.517 0.091 -0.316 -0.578 0.499 0.740 -0.061 -0.728 0.542 0.090
119) -0.262 -0.674 0.468 0.213 -0.296 -0.704 0.477 0.151 -0.386 -0.588 0.475 0.765 -0.220 -0.713 0.490 0.145
120) -0.290 -0.661 0.430 0.231 -0.315 -0.692 0.451 0.169 -0.386 -0.578 0.459 0.765 -0.245 -0.702 0.453 0.159
121) 0.186 -0.644 0.412 0.370 0.159 -0.651 0.405 0.343 -0.058 -0.524 0.421 0.668 0.211 -0.676 0.414 0.294
122) 0.167 -0.611 0.378 0.431 0.131 -0.610 0.352 0.424 -0.105 -0.511 0.409 0.685 0.197 -0.653 0.396 0.354
123) 0.161 -0.552 0.342 0.458 0.124 -0.549 0.302 0.454 -0.199 -0.437 0.356 0.718 0.186 -0.588 0.349 0.404
124) 0.155 -0.518 0.298 0.464 0.126 -0.517 0.265 0.444 -0.199 -0.446 0.337 0.718 0.184 -0.557 0.313 0.390
125) 0.157 -0.505 0.266 0.454 0.126 -0.496 0.242 0.444 -0.316 -0.380 0.256 0.760 0.172 -0.529 0.254 0.416
126) 0.155 -0.474 0.238 0.464 0.124 -0.467 0.203 0.454 -0.362 -0.347 0.222 0.757 0.168 -0.499 0.212 0.436
127) -0.270 -0.453 0.192 0.591 -0.298 -0.444 0.151 0.582 -0.362 -0.385 0.272 0.757 -0.245 -0.474 0.163 0.559
128) -0.274 -0.425 0.131 0.609 -0.302 -0.415 0.093 0.602 -0.409 -0.367 0.196 0.774 -0.247 -0.455 0.101 0.569
129) -0.606 -0.421 0.109 0.620 -0.623 -0.411 0.070 0.612 -0.409 -0.367 0.196 0.774 -0.598 -0.453 0.086 0.580
130) -0.537 -0.479 0.205 0.528 -0.553 -0.482 0.177 0.510 -0.432 -0.437 0.309 0.782 -0.533 -0.502 0.178 0.506

131) -0.526 -0.581 0.308 0.460 -0.535 -0.592 0.303 0.430 -0.479 -0.517 0.399 0.799 -0.519 -0.592 0.290 0.446
132) -0.379 -0.658 0.434 0.306 -0.397 -0.667 0.429 0.282 -0.503 -0.591 0.533 0.807 -0.365 -0.668 0.415 0.295
133) -0.381 -0.686 0.495 0.315 -0.397 -0.697 0.507 0.282 -0.479 -0.629 0.589 0.799 -0.363 -0.702 0.496 0.285
134) -0.370 -0.685 0.512 0.313 -0.392 -0.693 0.515 0.286 -0.409 -0.616 0.579 0.774 -0.349 -0.706 0.520 0.273
135) -0.373 -0.686 0.493 0.323 -0.386 -0.720 0.524 0.256 -0.339 -0.630 0.563 0.749 -0.345 -0.726 0.532 0.253
136) -0.364 -0.678 0.495 0.286 -0.373 -0.710 0.521 0.221 -0.269 -0.617 0.557 0.723 -0.331 -0.714 0.526 0.217
137) -0.379 -0.602 0.407 0.350 -0.379 -0.664 0.452 0.250 -0.292 -0.551 0.468 0.732 -0.345 -0.671 0.451 0.253
138) -0.108 -0.659 0.464 0.183 -0.124 -0.707 0.500 0.089 -0.339 -0.570 0.467 0.749 -0.052 -0.711 0.498 0.098
139) -0.284 -0.642 0.423 0.248 -0.302 -0.691 0.460 0.157 -0.386 -0.577 0.443 0.765 -0.229 -0.694 0.453 0.161
140) -0.286 -0.636 0.417 0.257 -0.315 -0.680 0.436 0.169 -0.339 -0.578 0.473 0.749 -0.224 -0.691 0.449 0.165
141) 0.221 -0.642 0.419 0.340 0.245 -0.655 0.415 0.277 0.175 -0.499 0.412 0.664 0.245 -0.654 0.368 0.303
142) 0.196 -0.600 0.372 0.406 0.197 -0.600 0.356 0.376 -0.082 -0.470 0.370 0.696 0.231 -0.626 0.346 0.363
143) 0.182 -0.527 0.320 0.449 0.179 -0.513 0.287 0.431 -0.199 -0.382 0.303 0.739 0.222 -0.567 0.311 0.403
144) 0.178 -0.505 0.282 0.445 0.170 -0.476 0.253 0.447 -0.175 -0.407 0.296 0.730 0.222 -0.543 0.272 0.379
145) 0.175 -0.495 0.253 0.432 0.162 -0.446 0.206 0.463 -0.339 -0.345 0.230 0.769 0.215 -0.522 0.231 0.385
146) 0.169 -0.465 0.220 0.460 0.155 -0.414 0.157 0.493 -0.409 -0.308 0.187 0.774 0.211 -0.506 0.203 0.405
147) -0.253 -0.445 0.176 0.577 -0.287 -0.397 0.131 0.621 -0.409 -0.362 0.265 0.774 -0.215 -0.487 0.166 0.518
148) -0.260 -0.423 0.128 0.605 -0.289 -0.372 0.077 0.631 -0.432 -0.353 0.211 0.782 -0.218 -0.475 0.127 0.528
149) -0.611 -0.423 0.128 0.616 -0.641 -0.372 0.077 0.644 -0.409 -0.357 0.232 0.774 -0.574 -0.478 0.142 0.539
150) -0.549 -0.484 0.238 0.539 -0.579 -0.443 0.193 0.557 -0.432 -0.431 0.346 0.782 -0.508 -0.532 0.239 0.465
151) -0.531 -0.596 0.348 0.457 -0.559 -0.573 0.313 0.468 -0.456 -0.526 0.448 0.791 -0.492 -0.630 0.358 0.395
152) -0.373 -0.672 0.459 0.300 -0.408 -0.660 0.446 0.308 -0.456 -0.606 0.575 0.791 -0.326 -0.702 0.480 0.242
153) -0.375 -0.694 0.505 0.310 -0.405 -0.693 0.525 0.298 -0.432 -0.639 0.624 0.782 -0.326 -0.727 0.541 0.242
154) -0.364 -0.688 0.514 0.308 -0.401 -0.689 0.522 0.302 -0.339 -0.625 0.611 0.749 -0.315 -0.719 0.549 0.240
155) -0.368 -0.677 0.486 0.326 -0.395 -0.710 0.522 0.272 -0.292 -0.627 0.579 0.732 -0.313 -0.724 0.541 0.230
156) -0.356 -0.677 0.495 0.271 -0.381 -0.694 0.514 0.237 -0.222 -0.618 0.576 0.707 -0.299 -0.713 0.537 0.194
157) -0.370 -0.609 0.414 0.335 -0.386 -0.658 0.458 0.256 -0.222 -0.565 0.499 0.707 -0.315 -0.668 0.458 0.240
158) -0.100 -0.659 0.465 0.168 -0.133 -0.692 0.499 0.105 -0.269 -0.573 0.485 0.723 -0.023 -0.700 0.488 0.085
159) -0.282 -0.637 0.415 0.239 -0.307 -0.665 0.431 0.177 -0.339 -0.568 0.451 0.749 -0.222 -0.679 0.429 0.155
160) -0.262 -0.636 0.420 0.235 -0.296 -0.662 0.427 0.175 -0.316 -0.568 0.469 0.740 -0.199 -0.679 0.435 0.151
161) 0.297 -0.622 0.399 0.335 0.320 -0.644 0.404 0.240 0.175 -0.476 0.407 0.664 0.281 -0.644 0.365 0.277
162) 0.266 -0.568 0.334 0.407 0.258 -0.561 0.321 0.351 -0.058 -0.445 0.349 0.688 0.268 -0.607 0.326 0.337
163) 0.235 -0.492 0.276 0.456 0.223 -0.467 0.251 0.414 -0.199 -0.351 0.263 0.739 0.252 -0.549 0.286 0.383
164) 0.233 -0.488 0.264 0.443 0.212 -0.440 0.225 0.440 -0.129 -0.404 0.295 0.713 0.254 -0.539 0.270 0.349
165) 0.231 -0.484 0.240 0.431 0.195 -0.414 0.176 0.472 -0.269 -0.368 0.253 0.744 0.245 -0.522 0.232 0.365
166) 0.227 -0.470 0.226 0.449 0.190 -0.405 0.156 0.491 -0.362 -0.345 0.217 0.757 0.238 -0.517 0.225 0.395
167) -0.212 -0.452 0.180 0.577 -0.267 -0.391 0.131 0.621 -0.339 -0.405 0.314 0.749 -0.190 -0.501 0.190 0.518
168) -0.219 -0.427 0.127 0.604 -0.269 -0.366 0.072 0.630 -0.316 -0.401 0.268 0.740 -0.190 -0.488 0.150 0.518
169) -0.596 -0.429 0.146 0.616 -0.625 -0.370 0.101 0.643 -0.292 -0.408 0.307 0.732 -0.571 -0.490 0.171 0.529
170) -0.541 -0.501 0.271 0.544 -0.570 -0.443 0.221 0.562 -0.292 -0.474 0.427 0.732 -0.515 -0.548 0.283 0.471
171) -0.522 -0.615 0.384 0.461 -0.553 -0.579 0.340 0.483 -0.292 -0.569 0.525 0.732 -0.492 -0.648 0.396 0.395
172) -0.364 -0.686 0.502 0.305 -0.399 -0.671 0.481 0.313 -0.292 -0.643 0.645 0.732 -0.315 -0.712 0.495 0.240
173) -0.366 -0.709 0.545 0.314 -0.397 -0.706 0.558 0.303 -0.292 -0.659 0.674 0.732 -0.317 -0.730 0.547 0.250
174) -0.356 -0.698 0.546 0.313 -0.390 -0.702 0.549 0.297 -0.199 -0.646 0.654 0.698 -0.306 -0.719 0.547 0.248
175) -0.360 -0.681 0.517 0.331 -0.381 -0.714 0.540 0.257 -0.175 -0.635 0.606 0.690 -0.302 -0.722 0.535 0.228
176) -0.352 -0.668 0.513 0.294 -0.370 -0.695 0.529 0.232 -0.105 -0.623 0.599 0.664 -0.290 -0.701 0.520 0.202
177) -0.362 -0.612 0.448 0.340 -0.375 -0.660 0.476 0.252 -0.082 -0.584 0.538 0.656 -0.304 -0.664 0.453 0.238
178) -0.102 -0.660 0.491 0.175 -0.135 -0.688 0.507 0.112 -0.152 -0.579 0.510 0.681 -0.014 -0.689 0.472 0.093
179) -0.305 -0.635 0.430 0.252 -0.331 -0.656 0.429 0.191 -0.245 -0.571 0.474 0.715 -0.220 -0.666 0.403 0.169
180) -0.260 -0.637 0.444 0.226 -0.296 -0.661 0.442 0.175 -0.222 -0.581 0.503 0.707 -0.170 -0.669 0.412 0.141
181) 0.356 -0.611 0.412 0.299 0.388 -0.624 0.403 0.229 0.175 -0.447 0.398 0.664 0.329 -0.635 0.376 0.249
182) 0.323 -0.539 0.331 0.380 0.318 -0.535 0.308 0.332 -0.058 -0.409 0.330 0.688 0.313 -0.576 0.310 0.319
183) 0.262 -0.463 0.260 0.438 0.252 -0.447 0.230 0.394 -0.199 -0.311 0.227 0.739 0.279 -0.517 0.258 0.372
184) 0.256 -0.459 0.249 0.443 0.243 -0.441 0.219 0.410 -0.082 -0.392 0.285 0.696 0.277 -0.515 0.250 0.359
185) 0.253 -0.460 0.227 0.430 0.225 -0.422 0.180 0.442 -0.245 -0.359 0.246 0.735 0.268 -0.504 0.216 0.374
186) 0.251 -0.461 0.230 0.440 0.223 -0.433 0.186 0.452 -0.339 -0.359 0.235 0.749 0.263 -0.509 0.231 0.394
187) -0.190 -0.444 0.188 0.577 -0.236 -0.420 0.162 0.591 -0.339 -0.421 0.334 0.749 -0.165 -0.498 0.206 0.518
188) -0.198 -0.415 0.127 0.613 -0.243 -0.388 0.092 0.620 -0.339 -0.403 0.267 0.749 -0.170 -0.477 0.153 0.538
189) -0.584 -0.417 0.149 0.624 -0.623 -0.390 0.106 0.633 -0.316 -0.413 0.315 0.740 -0.560 -0.479 0.176 0.548
190) -0.535 -0.496 0.281 0.558 -0.577 -0.471 0.240 0.568 -0.316 -0.477 0.433 0.740 -0.503 -0.539 0.284 0.490
191) -0.510 -0.614 0.389 0.470 -0.551 -0.600 0.347 0.473 -0.316 -0.573 0.525 0.740 -0.478 -0.651 0.403 0.405
192) -0.350 -0.688 0.512 0.304 -0.399 -0.677 0.478 0.313 -0.292 -0.646 0.641 0.732 -0.299 -0.716 0.511 0.239
193) -0.354 -0.704 0.544 0.323 -0.399 -0.703 0.541 0.313 -0.316 -0.644 0.649 0.740 -0.302 -0.731 0.554 0.249
194) -0.342 -0.695 0.537 0.312 -0.392 -0.693 0.521 0.307 -0.245 -0.623 0.620 0.715 -0.295 -0.721 0.551 0.243
195) -0.344 -0.678 0.514 0.321 -0.384 -0.698 0.508 0.267 -0.269 -0.593 0.553 0.723 -0.292 -0.710 0.525 0.233
196) -0.340 -0.652 0.500 0.303 -0.375 -0.666 0.484 0.252 -0.199 -0.569 0.535 0.698 -0.286 -0.673 0.494 0.227
197) -0.346 -0.608 0.452 0.330 -0.375 -0.639 0.437 0.252 -0.175 -0.540 0.485 0.690 -0.297 -0.644 0.440 0.253
198) -0.095 -0.654 0.493 0.167 -0.144 -0.670 0.482 0.104 -0.245 -0.537 0.453 0.715 -0.032 -0.672 0.468 0.101
199) -0.319 -0.623 0.414 0.250 -0.340 -0.637 0.405 0.183 -0.316 -0.532 0.430 0.740 -0.261 -0.647 0.394 0.183
200) -0.237 -0.633 0.449 0.210 -0.278 -0.645 0.423 0.143 -0.292 -0.547 0.458 0.732 -0.184 -0.656 0.436 0.153

Raffia

1) 0.220 -0.699 0.778 0.309 0.048 -0.724 0.752 0.202 -0.222 -0.665 0.704 0.411 -0.149 -0.693 0.702 0.136
2) 0.096 -0.673 0.717 0.360 0.019 -0.706 0.716 0.213 -0.323 -0.632 0.644 0.435 -0.264 -0.668 0.640 0.239
3) 0.028 -0.650 0.679 0.395 0.041 -0.694 0.693 0.200 -0.343 -0.613 0.607 0.459 -0.321 -0.643 0.603 0.290
4) -0.003 -0.610 0.642 0.441 0.037 -0.682 0.677 0.211 -0.327 -0.601 0.594 0.444 -0.306 -0.631 0.595 0.287
5) -0.158 -0.566 0.596 0.496 -0.101 -0.655 0.628 0.246 -0.316 -0.600 0.591 0.416 -0.312 -0.601 0.578 0.306
6) 0.207 -0.691 0.779 0.305 0.267 -0.717 0.754 0.149 -0.016 -0.667 0.735 0.379 -0.020 -0.695 0.730 0.125
7) 0.084 -0.662 0.727 0.357 0.200 -0.695 0.722 0.175 -0.129 -0.633 0.675 0.408 -0.125 -0.670 0.681 0.222
8) 0.034 -0.641 0.700 0.386 0.164 -0.676 0.693 0.182 -0.168 -0.615 0.646 0.434 -0.194 -0.641 0.642 0.289
9) 0.009 -0.610 0.677 0.422 0.108 -0.658 0.664 0.202 -0.214 -0.596 0.622 0.437 -0.231 -0.621 0.612 0.319
10) -0.145 -0.570 0.634 0.478 -0.058 -0.627 0.615 0.246 -0.265 -0.592 0.609 0.427 -0.303 -0.580 0.577 0.352
11) 0.158 -0.676 0.756 0.335 0.196 -0.712 0.747 0.186 -0.055 -0.657 0.727 0.405 -0.095 -0.692 0.732 0.214
12) 0.040 -0.651 0.715 0.377 0.140 -0.687 0.721 0.206 -0.156 -0.626 0.681 0.429 -0.176 -0.671 0.689 0.303
13) -0.003 -0.631 0.694 0.397 0.108 -0.671 0.692 0.202 -0.179 -0.615 0.660 0.440 -0.234 -0.649 0.656 0.353
14) -0.022 -0.611 0.681 0.424 0.076 -0.661 0.674 0.199 -0.222 -0.601 0.640 0.433 -0.273 -0.628 0.624 0.393
15) -0.152 -0.586 0.652 0.465 -0.034 -0.646 0.654 0.222 -0.265 -0.604 0.633 0.427 -0.336 -0.601 0.595 0.420
16) 0.034 -0.680 0.747 0.344 0.136 -0.717 0.752 0.213 -0.066 -0.656 0.717 0.410 -0.143 -0.689 0.722 0.258
17) -0.059 -0.665 0.720 0.372 0.094 -0.704 0.744 0.216 -0.152 -0.635 0.692 0.419 -0.213 -0.671 0.691 0.329
18) -0.108 -0.647 0.699 0.401 0.065 -0.696 0.720 0.202 -0.175 -0.626 0.676 0.431 -0.276 -0.651 0.660 0.401
19) -0.127 -0.636 0.694 0.429 0.034 -0.688 0.708 0.199 -0.226 -0.611 0.656 0.442 -0.312 -0.635 0.631 0.430
20) -0.158 -0.624 0.681 0.452 -0.044 -0.678 0.686 0.203 -0.273 -0.616 0.645 0.445 -0.360 -0.619 0.601 0.455
21) -0.059 -0.669 0.725 0.394 0.058 -0.718 0.755 0.246 -0.090 -0.653 0.709 0.422 -0.182 -0.688 0.715 0.295
22) -0.127 -0.663 0.710 0.407 0.030 -0.721 0.762 0.232 -0.179 -0.631 0.683 0.440 -0.240 -0.677 0.692 0.349
23) -0.182 -0.644 0.691 0.445 -0.005 -0.706 0.733 0.239 -0.203 -0.625 0.668 0.451 -0.279 -0.657 0.674 0.412
24) -0.189 -0.637 0.687 0.454 -0.037 -0.701 0.721 0.236 -0.257 -0.614 0.650 0.472 -0.324 -0.642 0.642 0.448
25) -0.195 -0.631 0.672 0.463 -0.101 -0.693 0.699 0.223 -0.308 -0.616 0.633 0.484 -0.378 -0.626 0.607 0.494
26) -0.096 -0.234 0.507 0.534 -0.518 -0.134 0.424 0.664 -0.296 -0.119 0.482 0.587 0.110 -0.236 0.531 0.371
27) -0.077 -0.112 0.428 0.506 -0.454 0.015 0.358 0.626 -0.234 -0.018 0.454 0.570 0.173 -0.092 0.460 0.367
28) -0.158 -0.014 0.379 0.559 -0.408 0.103 0.284 0.641 -0.171 0.061 0.394 0.533 0.170 0.037 0.375 0.427
29) -0.263 0.086 0.333 0.627 -0.362 0.178 0.267 0.630 -0.129 0.135 0.375 0.497 0.107 0.114 0.361 0.473
30) -0.498 0.094 0.296 0.714 -0.348 0.133 0.297 0.613 -0.105 0.096 0.387 0.486 -0.047 0.114 0.354 0.519
31) -0.084 -0.289 0.544 0.537 -0.546 -0.188 0.430 0.698 -0.281 -0.203 0.509 0.572 0.182 -0.306 0.568 0.334
32) -0.065 -0.172 0.465 0.510 -0.479 -0.048 0.372 0.649 -0.214 -0.111 0.475 0.545 0.246 -0.167 0.490 0.331
33) -0.152 -0.072 0.399 0.572 -0.444 0.038 0.293 0.670 -0.152 -0.031 0.402 0.508 0.237 -0.044 0.398 0.413
34) -0.263 0.031 0.341 0.649 -0.412 0.112 0.269 0.677 -0.117 0.044 0.357 0.492 0.161 0.036 0.366 0.476
35) -0.547 0.051 0.287 0.743 -0.401 0.076 0.285 0.670 -0.109 0.017 0.356 0.495 -0.071 0.047 0.341 0.554
36) -0.127 -0.316 0.541 0.558 -0.447 -0.284 0.474 0.605 -0.199 -0.286 0.543 0.511 0.188 -0.395 0.607 0.289
37) -0.108 -0.213 0.472 0.530 -0.394 -0.154 0.421 0.573 -0.148 -0.202 0.506 0.499 0.240 -0.267 0.521 0.303
38) -0.195 -0.127 0.420 0.592 -0.405 -0.075 0.352 0.605 -0.101 -0.129 0.443 0.477 0.231 -0.155 0.442 0.385
39) -0.312 -0.028 0.356 0.678 -0.377 -0.004 0.317 0.622 -0.101 -0.055 0.390 0.477 0.116 -0.077 0.399 0.460
40) -0.585 -0.009 0.294 0.754 -0.370 -0.028 0.303 0.626 -0.074 -0.077 0.380 0.479 -0.092 -0.064 0.374 0.529
41) -0.170 -0.373 0.549 0.578 -0.426 -0.390 0.505 0.566 -0.179 -0.390 0.571 0.487 0.152 -0.483 0.643 0.292
42) -0.158 -0.298 0.507 0.559 -0.345 -0.288 0.475 0.526 -0.097 -0.330 0.556 0.468 0.219 -0.407 0.584 0.278
43) -0.220 -0.234 0.478 0.607 -0.338 -0.232 0.438 0.556 -0.047 -0.278 0.526 0.436 0.216 -0.323 0.537 0.338
44) -0.337 -0.150 0.423 0.693 -0.302 -0.177 0.409 0.551 -0.043 -0.215 0.478 0.427 0.104 -0.255 0.492 0.402
45) -0.603 -0.134 0.363 0.759 -0.295 -0.204 0.413 0.556 -0.016 -0.236 0.487 0.429 -0.065 -0.244 0.467 0.434
46) -0.195 -0.443 0.563 0.570 -0.387 -0.504 0.563 0.500 -0.136 -0.510 0.624 0.429 0.137 -0.578 0.683 0.245
47) -0.158 -0.411 0.558 0.537 -0.317 -0.450 0.557 0.466 -0.062 -0.488 0.635 0.406 0.216 -0.543 0.654 0.213
48) -0.220 -0.358 0.530 0.585 -0.309 -0.407 0.531 0.496 -0.016 -0.452 0.613 0.384 0.225 -0.504 0.638 0.256
49) -0.331 -0.285 0.478 0.662 -0.281 -0.360 0.512 0.513 -0.016 -0.403 0.580 0.384 0.134 -0.459 0.618 0.322
50) -0.535 -0.273 0.436 0.703 -0.274 -0.383 0.516 0.517 0.019 -0.420 0.593 0.390 0.011 -0.451 0.600 0.344
51) 0.003 -0.638 0.758 0.429 -0.030 -0.716 0.803 0.276 -0.027 -0.677 0.780 0.386 0.020 -0.684 0.767 0.294
52) 0.059 -0.659 0.789 0.434 0.062 -0.721 0.827 0.295 0.136 -0.669 0.780 0.403 0.216 -0.711 0.806 0.246
53) 0.108 -0.633 0.774 0.449 0.069 -0.701 0.800 0.300 0.238 -0.676 0.787 0.362 0.396 -0.719 0.808 0.225
54) 0.133 -0.590 0.733 0.456 0.051 -0.666 0.756 0.305 0.362 -0.653 0.767 0.343 0.550 -0.697 0.783 0.225
55) 0.170 -0.560 0.696 0.445 0.016 -0.613 0.685 0.338 0.425 -0.626 0.744 0.326 0.713 -0.679 0.764 0.214
56) -0.015 -0.627 0.746 0.434 -0.094 -0.708 0.782 0.291 -0.129 -0.661 0.759 0.427 -0.068 -0.674 0.755 0.333
57) -0.015 -0.640 0.768 0.456 -0.002 -0.716 0.806 0.311 0.035 -0.653 0.760 0.443 0.101 -0.693 0.797 0.304
58) 0.034 -0.618 0.760 0.471 0.005 -0.697 0.779 0.315 0.136 -0.664 0.766 0.403 0.282 -0.698 0.799 0.284
59) 0.071 -0.592 0.739 0.460 -0.012 -0.662 0.738 0.321 0.261 -0.639 0.745 0.384 0.436 -0.674 0.775 0.283
60) 0.108 -0.561 0.699 0.449 -0.048 -0.609 0.654 0.353 0.312 -0.611 0.726 0.372 0.586 -0.670 0.773 0.263
61) -0.084 -0.641 0.749 0.428 -0.051 -0.719 0.797 0.288 -0.074 -0.696 0.797 0.342 -0.041 -0.718 0.806 0.237
62) -0.077 -0.655 0.771 0.441 0.051 -0.725 0.820 0.302 0.121 -0.698 0.810 0.351 0.152 -0.739 0.854 0.200
63) -0.022 -0.645 0.769 0.446 0.087 -0.711 0.796 0.298 0.245 -0.701 0.812 0.319 0.351 -0.737 0.857 0.189
64) 0.009 -0.628 0.751 0.444 0.083 -0.674 0.745 0.311 0.339 -0.672 0.776 0.310 0.475 -0.712 0.833 0.200

65) 0.015 -0.597 0.715 0.457 0.048 -0.637 0.679 0.344 0.386 -0.643 0.751 0.307 0.592 -0.702 0.811 0.199
66) -0.090 -0.661 0.751 0.415 -0.027 -0.722 0.806 0.265 -0.074 -0.700 0.803 0.297 0.005 -0.731 0.820 0.178
67) -0.077 -0.676 0.773 0.419 0.044 -0.730 0.826 0.298 0.027 -0.706 0.815 0.315 0.134 -0.748 0.861 0.144
68) -0.077 -0.658 0.754 0.441 0.065 -0.718 0.795 0.310 0.136 -0.704 0.809 0.297 0.258 -0.744 0.860 0.155
69) -0.071 -0.633 0.723 0.453 0.058 -0.683 0.743 0.335 0.226 -0.672 0.761 0.298 0.366 -0.715 0.824 0.194
70) -0.071 -0.599 0.677 0.475 0.023 -0.651 0.680 0.367 0.269 -0.647 0.734 0.305 0.472 -0.706 0.809 0.210
71) -0.053 -0.683 0.775 0.404 -0.012 -0.729 0.821 0.273 -0.156 -0.706 0.791 0.336 -0.089 -0.733 0.803 0.182
72) -0.065 -0.688 0.782 0.422 0.058 -0.740 0.840 0.306 -0.055 -0.711 0.797 0.354 0.017 -0.745 0.829 0.157
73) -0.059 -0.665 0.756 0.435 0.080 -0.730 0.810 0.319 0.051 -0.700 0.790 0.346 0.137 -0.735 0.822 0.179
74) -0.059 -0.629 0.709 0.457 0.072 -0.698 0.749 0.343 0.144 -0.665 0.740 0.337 0.246 -0.707 0.780 0.218
75) -0.059 -0.586 0.657 0.479 0.041 -0.664 0.695 0.365 0.187 -0.636 0.711 0.344 0.351 -0.694 0.766 0.234
76) -0.059 -0.576 0.651 0.435 -0.225 -0.579 0.597 0.369 -0.436 -0.530 0.585 0.501 -0.237 -0.611 0.641 0.292
77) -0.046 -0.599 0.673 0.439 -0.214 -0.615 0.628 0.388 -0.421 -0.581 0.605 0.508 -0.219 -0.647 0.669 0.279
78) -0.077 -0.595 0.661 0.484 -0.246 -0.628 0.633 0.432 -0.417 -0.564 0.568 0.521 -0.197 -0.654 0.675 0.281
79) -0.139 -0.566 0.633 0.554 -0.278 -0.594 0.592 0.503 -0.401 -0.550 0.555 0.528 -0.182 -0.629 0.652 0.305
80) -0.182 -0.508 0.586 0.596 -0.253 -0.544 0.546 0.502 -0.382 -0.481 0.517 0.549 -0.219 -0.541 0.579 0.384
81) -0.065 -0.612 0.683 0.444 -0.263 -0.583 0.604 0.409 -0.456 -0.504 0.574 0.547 -0.282 -0.610 0.645 0.325
82) -0.059 -0.647 0.721 0.457 -0.249 -0.632 0.649 0.418 -0.436 -0.580 0.608 0.545 -0.267 -0.655 0.679 0.323
83) -0.090 -0.636 0.705 0.503 -0.281 -0.646 0.654 0.462 -0.429 -0.572 0.576 0.549 -0.249 -0.656 0.680 0.336
84) -0.164 -0.601 0.665 0.568 -0.309 -0.612 0.614 0.522 -0.397 -0.564 0.581 0.541 -0.222 -0.637 0.671 0.342
85) -0.207 -0.547 0.624 0.611 -0.253 -0.577 0.599 0.502 -0.343 -0.515 0.574 0.545 -0.237 -0.562 0.624 0.397
86) -0.176 -0.593 0.646 0.458 -0.359 -0.576 0.594 0.466 -0.487 -0.482 0.547 0.599 -0.321 -0.617 0.632 0.339
87) -0.164 -0.630 0.685 0.461 -0.348 -0.622 0.637 0.486 -0.471 -0.554 0.581 0.606 -0.309 -0.659 0.670 0.348
88) -0.176 -0.618 0.662 0.502 -0.377 -0.628 0.623 0.520 -0.464 -0.546 0.555 0.610 -0.285 -0.651 0.657 0.365
89) -0.251 -0.588 0.644 0.567 -0.405 -0.598 0.596 0.579 -0.432 -0.537 0.557 0.603 -0.264 -0.630 0.649 0.367
90) -0.275 -0.547 0.623 0.604 -0.320 -0.573 0.600 0.551 -0.374 -0.504 0.569 0.597 -0.282 -0.562 0.620 0.433
91) -0.226 -0.601 0.630 0.465 -0.419 -0.586 0.591 0.520 -0.514 -0.510 0.535 0.619 -0.375 -0.630 0.624 0.378
92) -0.213 -0.634 0.680 0.469 -0.394 -0.620 0.625 0.522 -0.483 -0.557 0.564 0.612 -0.354 -0.659 0.659 0.380
93) -0.201 -0.617 0.655 0.494 -0.394 -0.615 0.601 0.522 -0.460 -0.549 0.554 0.601 -0.315 -0.640 0.629 0.370
94) -0.294 -0.584 0.642 0.566 -0.437 -0.577 0.571 0.598 -0.429 -0.524 0.550 0.593 -0.306 -0.617 0.632 0.389
95) -0.319 -0.543 0.622 0.602 -0.355 -0.556 0.579 0.580 -0.370 -0.497 0.567 0.588 -0.324 -0.544 0.593 0.455
96) -0.325 -0.620 0.630 0.502 -0.444 -0.620 0.600 0.492 -0.522 -0.534 0.540 0.616 -0.412 -0.640 0.631 0.405
97) -0.300 -0.626 0.658 0.509 -0.423 -0.637 0.619 0.505 -0.479 -0.568 0.575 0.603 -0.363 -0.662 0.670 0.387
98) -0.281 -0.594 0.623 0.525 -0.394 -0.623 0.602 0.496 -0.444 -0.562 0.576 0.586 -0.312 -0.631 0.616 0.359
99) -0.405 -0.550 0.597 0.620 -0.437 -0.586 0.565 0.573 -0.397 -0.538 0.570 0.564 -0.315 -0.605 0.625 0.396
100) -0.430 -0.516 0.576 0.657 -0.362 -0.565 0.572 0.576 -0.335 -0.511 0.581 0.549 -0.324 -0.543 0.596 0.455
101) -0.405 -0.329 0.348 0.686 -0.596 -0.346 0.300 0.691 -0.604 -0.229 0.289 0.702 -0.466 -0.387 0.374 0.575
102) -0.622 -0.189 0.224 0.765 -0.713 -0.214 0.198 0.764 -0.658 -0.087 0.197 0.742 -0.562 -0.241 0.282 0.663
103) -0.455 -0.182 0.280 0.694 -0.614 -0.207 0.242 0.696 -0.639 -0.026 0.191 0.741 -0.502 -0.162 0.278 0.651
104) -0.411 -0.204 0.300 0.651 -0.550 -0.240 0.253 0.632 -0.643 -0.052 0.211 0.728 -0.514 -0.140 0.279 0.668
105) -0.442 -0.244 0.325 0.653 -0.511 -0.313 0.317 0.566 -0.627 -0.095 0.242 0.713 -0.574 -0.161 0.293 0.706
106) -0.448 -0.317 0.340 0.684 -0.631 -0.301 0.275 0.721 -0.619 -0.145 0.253 0.739 -0.505 -0.316 0.344 0.639
107) -0.615 -0.181 0.225 0.756 -0.720 -0.187 0.190 0.785 -0.678 -0.018 0.171 0.789 -0.574 -0.201 0.271 0.706
108) -0.473 -0.148 0.268 0.699 -0.635 -0.167 0.225 0.734 -0.658 0.059 0.161 0.787 -0.526 -0.091 0.252 0.712
109) -0.424 -0.176 0.293 0.648 -0.571 -0.197 0.238 0.670 -0.658 0.019 0.185 0.765 -0.532 -0.080 0.259 0.708
110) -0.455 -0.214 0.314 0.650 -0.529 -0.277 0.304 0.594 -0.639 -0.026 0.218 0.741 -0.586 -0.105 0.272 0.724
111) -0.467 -0.353 0.371 0.668 -0.575 -0.343 0.325 0.678 -0.506 -0.234 0.323 0.668 -0.472 -0.357 0.368 0.594
112) -0.585 -0.269 0.279 0.732 -0.624 -0.288 0.271 0.725 -0.514 -0.165 0.263 0.706 -0.469 -0.310 0.326 0.632
113) -0.467 -0.251 0.326 0.690 -0.511 -0.281 0.296 0.688 -0.413 -0.120 0.268 0.701 -0.357 -0.229 0.313 0.634
114) -0.411 -0.290 0.362 0.630 -0.394 -0.305 0.319 0.615 -0.382 -0.160 0.300 0.671 -0.315 -0.233 0.324 0.610
115) -0.448 -0.298 0.366 0.641 -0.324 -0.359 0.376 0.530 -0.331 -0.178 0.332 0.640 -0.348 -0.230 0.330 0.628
116) -0.467 -0.419 0.443 0.668 -0.423 -0.403 0.427 0.601 -0.382 -0.306 0.422 0.593 -0.418 -0.410 0.430 0.528
117) -0.560 -0.380 0.394 0.717 -0.479 -0.387 0.386 0.643 -0.390 -0.257 0.362 0.632 -0.390 -0.392 0.406 0.558
118) -0.461 -0.376 0.429 0.681 -0.401 -0.383 0.403 0.636 -0.316 -0.227 0.365 0.647 -0.333 -0.327 0.387 0.576
119) -0.436 -0.397 0.446 0.644 -0.341 -0.390 0.407 0.583 -0.319 -0.245 0.376 0.634 -0.306 -0.311 0.395 0.580
120) -0.479 -0.384 0.430 0.664 -0.302 -0.419 0.437 0.517 -0.296 -0.234 0.370 0.623 -0.339 -0.311 0.392 0.598
121) -0.448 -0.437 0.480 0.662 -0.437 -0.439 0.483 0.618 -0.401 -0.366 0.462 0.617 -0.460 -0.443 0.489 0.553
122) -0.516 -0.419 0.444 0.697 -0.469 -0.430 0.460 0.637 -0.405 -0.316 0.412 0.646 -0.430 -0.432 0.468 0.572
123) -0.473 -0.408 0.456 0.677 -0.416 -0.414 0.456 0.627 -0.327 -0.285 0.413 0.653 -0.378 -0.369 0.442 0.586
124) -0.473 -0.412 0.454 0.655 -0.366 -0.416 0.448 0.581 -0.323 -0.300 0.416 0.621 -0.348 -0.354 0.436 0.579
125) -0.523 -0.397 0.434 0.684 -0.331 -0.432 0.461 0.525 -0.284 -0.292 0.419 0.595 -0.342 -0.352 0.427 0.583
126) -0.003 -0.633 0.726 0.506 -0.005 -0.714 0.754 0.303 -0.055 -0.661 0.713 0.366 0.041 -0.695 0.763 0.232
127) -0.040 -0.591 0.693 0.539 -0.051 -0.691 0.724 0.314 -0.062 -0.635 0.687 0.362 0.050 -0.675 0.739 0.225
128) -0.084 -0.569 0.671 0.537 -0.094 -0.650 0.663 0.339 -0.097 -0.607 0.659 0.379 -0.044 -0.624 0.673 0.275
129) -0.145 -0.515 0.625 0.585 -0.221 -0.599 0.590 0.400 -0.160 -0.571 0.614 0.393 -0.104 -0.586 0.641 0.313
130) -0.220 -0.454 0.562 0.629 -0.338 -0.527 0.522 0.499 -0.226 -0.512 0.553 0.440 -0.140 -0.531 0.578 0.340

131) -0.065 -0.573 0.663 0.554 0.292 -0.719 0.772 0.185 0.277 -0.680 0.744 0.254 0.225 -0.694 0.758 0.155
132) -0.096 -0.553 0.643 0.578 0.228 -0.702 0.753 0.198 0.234 -0.651 0.717 0.267 0.201 -0.671 0.733 0.164
133) -0.145 -0.536 0.623 0.585 0.143 -0.670 0.708 0.227 0.179 -0.631 0.701 0.285 0.080 -0.616 0.662 0.234
134) -0.226 -0.479 0.569 0.638 -0.041 -0.619 0.638 0.307 0.055 -0.586 0.651 0.317 -0.026 -0.573 0.623 0.281
135) -0.294 -0.421 0.505 0.673 -0.242 -0.546 0.544 0.416 -0.070 -0.523 0.569 0.372 -0.110 -0.513 0.538 0.328
136) -0.090 -0.515 0.640 0.568 0.133 -0.693 0.752 0.284 0.066 -0.644 0.698 0.356 0.011 -0.665 0.721 0.285
137) -0.121 -0.503 0.623 0.592 0.097 -0.681 0.735 0.288 0.047 -0.622 0.676 0.358 -0.014 -0.646 0.700 0.293
138) -0.170 -0.513 0.608 0.600 0.027 -0.660 0.704 0.300 0.008 -0.619 0.682 0.362 -0.122 -0.611 0.652 0.346
139) -0.257 -0.447 0.535 0.662 -0.126 -0.615 0.636 0.361 -0.078 -0.587 0.654 0.368 -0.219 -0.564 0.591 0.387
140) -0.331 -0.387 0.449 0.706 -0.274 -0.539 0.527 0.460 -0.148 -0.523 0.554 0.424 -0.276 -0.515 0.519 0.414
141) -0.145 -0.432 0.583 0.607 0.069 -0.664 0.739 0.329 0.074 -0.629 0.682 0.382 0.005 -0.650 0.717 0.306
142) -0.158 -0.430 0.570 0.625 0.076 -0.665 0.732 0.308 0.070 -0.605 0.654 0.369 -0.011 -0.636 0.694 0.309
143) -0.232 -0.433 0.540 0.647 0.037 -0.654 0.702 0.300 0.078 -0.617 0.677 0.351 -0.083 -0.613 0.653 0.335
144) -0.312 -0.371 0.467 0.700 -0.019 -0.609 0.636 0.346 0.027 -0.595 0.660 0.340 -0.179 -0.566 0.595 0.376
145) -0.380 -0.319 0.397 0.735 -0.161 -0.547 0.540 0.423 -0.031 -0.535 0.570 0.390 -0.216 -0.515 0.524 0.405
146) -0.145 -0.533 0.633 0.563 0.072 -0.672 0.759 0.344 0.101 -0.648 0.722 0.362 0.068 -0.676 0.748 0.260
147) -0.170 -0.509 0.607 0.600 0.090 -0.674 0.757 0.316 0.109 -0.628 0.704 0.343 0.050 -0.655 0.722 0.274
148) -0.257 -0.489 0.566 0.640 0.062 -0.664 0.739 0.302 0.129 -0.639 0.717 0.319 -0.026 -0.634 0.699 0.311
149) -0.288 -0.442 0.521 0.685 0.055 -0.628 0.681 0.327 0.097 -0.617 0.704 0.307 -0.134 -0.594 0.649 0.343
150) -0.368 -0.393 0.453 0.717 -0.058 -0.572 0.602 0.393 0.051 -0.565 0.629 0.352 -0.152 -0.553 0.587 0.359
151) -0.238 -0.534 0.561 0.549 -0.465 -0.552 0.530 0.454 -0.475 -0.475 0.517 0.616 -0.288 -0.549 0.560 0.455
152) -0.251 -0.521 0.560 0.567 -0.430 -0.541 0.536 0.472 -0.429 -0.444 0.510 0.613 -0.297 -0.518 0.553 0.514
153) -0.281 -0.492 0.541 0.613 -0.384 -0.540 0.543 0.484 -0.417 -0.451 0.511 0.607 -0.303 -0.473 0.523 0.562
154) -0.374 -0.446 0.496 0.641 -0.370 -0.497 0.510 0.518 -0.386 -0.404 0.495 0.600 -0.354 -0.425 0.492 0.590
155) -0.387 -0.407 0.465 0.659 -0.370 -0.459 0.467 0.543 -0.370 -0.363 0.461 0.607 -0.333 -0.387 0.479 0.566
156) -0.145 -0.544 0.592 0.478 -0.444 -0.523 0.482 0.441 -0.530 -0.421 0.464 0.615 -0.315 -0.502 0.503 0.451
157) -0.170 -0.501 0.565 0.514 -0.423 -0.493 0.472 0.476 -0.483 -0.373 0.451 0.612 -0.327 -0.445 0.484 0.521
158) -0.207 -0.437 0.520 0.569 -0.373 -0.465 0.457 0.477 -0.471 -0.362 0.440 0.606 -0.339 -0.373 0.437 0.565
159) -0.325 -0.352 0.450 0.611 -0.366 -0.394 0.396 0.507 -0.436 -0.290 0.394 0.590 -0.412 -0.301 0.388 0.591
160) -0.337 -0.302 0.410 0.630 -0.362 -0.347 0.351 0.522 -0.417 -0.239 0.359 0.588 -0.387 -0.269 0.374 0.556
161) -0.139 -0.542 0.584 0.469 -0.281 -0.547 0.494 0.380 -0.452 -0.436 0.461 0.585 -0.306 -0.516 0.484 0.442
162) -0.176 -0.507 0.567 0.524 -0.320 -0.511 0.478 0.420 -0.452 -0.388 0.442 0.585 -0.324 -0.459 0.462 0.507
163) -0.213 -0.442 0.526 0.578 -0.313 -0.475 0.452 0.424 -0.460 -0.370 0.430 0.581 -0.342 -0.384 0.422 0.547
164) -0.331 -0.347 0.447 0.620 -0.334 -0.406 0.407 0.463 -0.409 -0.293 0.381 0.550 -0.406 -0.294 0.350 0.566
165) -0.337 -0.300 0.418 0.630 -0.331 -0.360 0.368 0.477 -0.394 -0.235 0.342 0.557 -0.384 -0.256 0.332 0.542
166) -0.145 -0.563 0.593 0.456 -0.263 -0.577 0.536 0.378 -0.409 -0.486 0.509 0.550 -0.261 -0.541 0.517 0.383
167) -0.189 -0.525 0.575 0.520 -0.299 -0.541 0.524 0.407 -0.409 -0.435 0.487 0.550 -0.288 -0.486 0.492 0.455
168) -0.238 -0.450 0.522 0.593 -0.292 -0.501 0.501 0.412 -0.421 -0.402 0.470 0.555 -0.318 -0.411 0.458 0.512
169) -0.380 -0.346 0.432 0.650 -0.345 -0.429 0.444 0.469 -0.409 -0.319 0.409 0.550 -0.396 -0.322 0.403 0.560
170) -0.387 -0.295 0.401 0.659 -0.384 -0.379 0.409 0.509 -0.440 -0.255 0.364 0.579 -0.415 -0.273 0.369 0.573
171) -0.195 -0.571 0.581 0.463 -0.281 -0.599 0.548 0.431 -0.444 -0.492 0.508 0.589 -0.294 -0.574 0.558 0.398
172) -0.238 -0.536 0.555 0.527 -0.313 -0.561 0.522 0.450 -0.448 -0.440 0.479 0.598 -0.324 -0.521 0.520 0.481
173) -0.288 -0.460 0.500 0.600 -0.306 -0.516 0.487 0.454 -0.452 -0.404 0.455 0.607 -0.357 -0.445 0.476 0.549
174) -0.430 -0.356 0.409 0.657 -0.359 -0.444 0.432 0.511 -0.444 -0.319 0.390 0.611 -0.439 -0.355 0.414 0.608
175) -0.436 -0.295 0.368 0.666 -0.401 -0.384 0.386 0.562 -0.475 -0.253 0.346 0.640 -0.460 -0.296 0.367 0.632
176) -0.473 -0.211 0.465 0.721 -0.101 -0.380 0.556 0.469 -0.144 -0.380 0.576 0.448 -0.216 -0.388 0.541 0.442
177) -0.220 -0.345 0.597 0.629 0.062 -0.474 0.662 0.400 -0.113 -0.468 0.659 0.418 -0.119 -0.489 0.649 0.354
178) -0.133 -0.465 0.675 0.567 0.118 -0.540 0.707 0.358 -0.090 -0.528 0.690 0.429 -0.077 -0.563 0.699 0.306
179) -0.127 -0.533 0.684 0.558 0.111 -0.588 0.704 0.354 -0.101 -0.559 0.690 0.435 -0.071 -0.611 0.714 0.310
180) -0.139 -0.594 0.692 0.532 0.076 -0.625 0.688 0.358 -0.125 -0.598 0.693 0.446 -0.107 -0.643 0.715 0.310
181) -0.473 -0.217 0.469 0.721 -0.122 -0.413 0.587 0.481 -0.183 -0.406 0.593 0.496 -0.231 -0.425 0.571 0.470
182) -0.294 -0.366 0.592 0.631 -0.041 -0.522 0.691 0.412 -0.164 -0.487 0.664 0.472 -0.134 -0.519 0.677 0.383
183) -0.201 -0.502 0.682 0.560 0.016 -0.598 0.742 0.370 -0.136 -0.557 0.699 0.474 -0.086 -0.604 0.730 0.313
184) -0.189 -0.579 0.700 0.542 0.023 -0.648 0.750 0.349 -0.129 -0.597 0.710 0.455 -0.065 -0.656 0.749 0.289
185) -0.213 -0.640 0.718 0.513 -0.009 -0.679 0.741 0.342 -0.132 -0.636 0.727 0.442 -0.101 -0.689 0.758 0.289
186) -0.442 -0.402 0.551 0.675 -0.108 -0.569 0.675 0.439 -0.168 -0.552 0.664 0.459 -0.188 -0.601 0.662 0.369
187) -0.312 -0.522 0.659 0.593 -0.080 -0.648 0.753 0.379 -0.152 -0.612 0.728 0.444 -0.092 -0.670 0.763 0.282
188) -0.244 -0.608 0.729 0.536 -0.055 -0.698 0.801 0.356 -0.125 -0.654 0.754 0.446 -0.047 -0.711 0.802 0.223
189) -0.232 -0.636 0.726 0.518 -0.055 -0.698 0.782 0.356 -0.125 -0.654 0.748 0.446 -0.029 -0.724 0.803 0.210
190) -0.257 -0.666 0.727 0.489 -0.090 -0.700 0.763 0.360 -0.129 -0.667 0.757 0.433 -0.068 -0.727 0.798 0.221
191) -0.393 -0.553 0.635 0.624 -0.094 -0.686 0.742 0.396 -0.140 -0.648 0.731 0.416 -0.137 -0.692 0.736 0.289
192) -0.312 -0.638 0.720 0.549 -0.076 -0.732 0.807 0.343 -0.125 -0.688 0.783 0.402 -0.065 -0.736 0.822 0.210
193) -0.269 -0.680 0.768 0.507 -0.065 -0.743 0.824 0.337 -0.097 -0.703 0.799 0.404 -0.032 -0.751 0.849 0.168
194) -0.257 -0.678 0.750 0.489 -0.065 -0.727 0.806 0.337 -0.097 -0.698 0.793 0.404 -0.014 -0.743 0.825 0.155
195) -0.139 -0.685 0.753 0.460 -0.090 -0.710 0.774 0.335 -0.066 -0.699 0.805 0.374 0.017 -0.735 0.820 0.167
196) -0.312 -0.655 0.731 0.549 -0.094 -0.711 0.774 0.371 -0.129 -0.665 0.744 0.408 -0.071 -0.713 0.773 0.275
197) -0.257 -0.695 0.775 0.489 -0.072 -0.745 0.828 0.307 -0.117 -0.695 0.788 0.403 -0.026 -0.743 0.838 0.216
198) -0.257 -0.707 0.789 0.467 -0.062 -0.747 0.835 0.301 -0.094 -0.700 0.795 0.414 -0.029 -0.747 0.846 0.201
199) -0.263 -0.691 0.757 0.454 -0.076 -0.730 0.811 0.318 -0.105 -0.696 0.784 0.419 -0.056 -0.736 0.817 0.197
200) -0.053 -0.687 0.759 0.418 -0.080 -0.720 0.799 0.325 -0.136 -0.691 0.774 0.407 0.002 -0.729 0.804 0.208

APPENDIX - II

(SOURCE CODE)

```
/*-----------------------------------------------------------------------------------------------
        SOURCE CODE OF IMAGE TEXTURE CLASSIFICATION
-----------------------------------------------------------------------------------------------*/

#include <stdio.h>
#include <string.h>
#include ".\func.h"

#define FLen 16           // number of texture parameters
#define FN 4
int L;                    // number of gray levels in image
int N;                    // size of square image of NxN pixels
#define FILTER 7          // mask size of median filter

                                        // Network parameters
#define HI 0.9
#define LO 0.1
#define bias 1.0
#define Threshold 0.400
#define MIN  9999999.99

int TRNG ;          // size of training data set
int EPOCHS;

float *O;        // number of output classes
float *zk;            // actual output vector
float *tk;       // true output vector
float *err;           // network error vector
float **W1, **dw1;    // weight matrix for weights between input and hidden layer
float **W2,  **dw2;   // weight matrix for weights between hidden and output layer
float *H_v1,*H_y1;    // weighted sum and output of neurons in hidden layer
float *O_v1,*O_y1;    // weighted sum and output of neurons in output layer
float *delta_k,       // local gradient of neurons in output layer
float *delta_h;            // local gradient of neurons in hidden layer

float *bw1, *d_bw1;   // bias weight vector for hidden layer neurons
float *bw2,*d_bw2;    // bias weight vector for output layer neurons


/*------------------------ FUNCTION PROTOTYPES -----------------------------------*/

void creat_NET(int, int);

void initialise_RAND();

float init_weights(float , float);

void initialise_wts(int, int);

void display_wts(int, int);

void set_tk(float **,int);

float phi(float );

void get_netout(float *, float *,int ,int );

void TRAIN(int, int, float, float, float **, int *, int)     ;
```

```
void TEST(float **,int *, int, int, int);

void save_wts(char wt_file[150], int, int, float, float, float);

float **get_GLCM(unsigned char **, int, int, int, int);

void cal_features(float **,float f[FN]);

void create_texture(char fname[150], int, float **Tx[FLen]);

void Normalise(float**,int,int,int );

unsigned char **smooth_img(unsigned char **, int , int ,int );

unsigned int sort(unsigned int win[FILTER][FILTER]);

void free_network(int, int);



/* ------------------------------------ MAIN FUNCTION --------------------------------------*/

void main()
{
    int n ;                 // texture window size
    int q;
    FILE *fp;
    float **train,**test;
    int *class_TAG;
    int trn;
        int SET1            // no of training patterns per class
    int OUT;        // number of output neurons
    int HIDEN;            // number of hidden neurons
    int *CLASS;
    char f_trn [5][150];        // training data files
        char f_test[5][150];  // test data files
    int counter;
    int i,j,k,m,l,J;
    float p;
    float  MAX;
    int sz;
        int itr_no;
    float eta,alpha;

        printf("\n Program to perform supervised image classification");
        printf("\n Enter number of OUTPUT classes");
        scanf("%d",&OUT);
        printf("\n Enter number of training patterns per class");
        scanf("%d",&SET1);
        printf("\n Enter number of HIDDEN neurons");
        scanf("%d",&HIDEN);
        printf("\n Enter number of EPOCHS");
        scanf("%d",&EPOCHS);
        printf("\n Enter value of learning eta");
        scanf("%f ",&eta);

    itr_no=0;
    alpha=0.0;
```

```
TRNG = SET1*OUT;                   // total training samples
train    = create_fl(TRNG,FLen);
test     = create_fl(TRNG,FLen);
class_TAG = (int *)malloc(sizeof(int)*TRNG);
for(i=0;i<TRNG; i++)
        class_TAG[i]=99;           // since numeric value of 0 is first class

if(train==NULL)
        printf("\n NOT ENOUGH MEMORY..");

creat_NET(HIDEN,OUT);
printf("\n %d x %d x %d NEURAL NET created",FLen,HIDEN,OUT);

     CLASS= (int *)malloc(sizeof(int)*OUT);  // records number of samples of each class
counter=0;
trn = 0;
printf("\n READING TRN files \n");
for(i=0;i<OUT; i++)
{
        fp = fopen(f_trn[i],"r");
        if(fp==NULL)
        {
                printf("\n CANNOT OPEN %s",f_trn[i]);
                exit(0);
        }
        fscanf(fp,"%d\n",&trn);
        printf(" %d SAMPLES IN %s",trn,f_trn[i]);

        for(m=0;m<SET1; m++)
        {
                class_TAG[counter]=i;  // file number is TAG no
                for(J=0;J<FLen;J++)
                {
                        fscanf(fp,"%f",&p);
                        train[counter][J]=p;
                }
                if(m<trn-1) fscanf(fp,"\n");
                counter++;
        }
     printf("\n counter = %d",counter);
fclose(fp);
if(!(counter%50))
      {
        printf("\n counter = %d",counter);
       }
}// end of OUT LOOP
printf("\n TOTAL %d TRAINING SAMPLES PICKED UP for %d CLASSES",
                  counter,OUT);
// Display of training data
for(m=0;m<TRNG; m++)
{
        printf("\n [%d]",class_TAG[m]);
        for(J=0;J<FLen; J++)
```

```
			printf(" %.2f",train[m][J]);

		if(!(m%10)) getchar();
	}

	initialise_RAND();
	initialise_wts(HIDEN,OUT);
	TRAIN(HIDEN,OUT,eta,alpha,train,class_TAG,TRNG);

	strcpy(fname,"d:\\x_june\\save_wtsN.txt");
	save_wts(fname,HIDEN,OUT,eta,alpha,MSE1);	//record learned information
	display_wts(HIDEN,OUT);

	printf("\n VALIDATION DATA... \n");
	int cntr,TAG;
	int GG =0;

	for(i=0;i<OUT; i++)
		CLASS[i]=0;

	while(GG<OUT)
	{
		TAG = GG;
		printf("\n TAG = %d for %s",TAG,f_trn[GG]);
		fp = fopen(f_test[TAG],"r");
		if(fp==NULL)
			printf("\n CANNOT OPEN file*** %s",f_test[GG]);;

		fscanf(fp,"%d\n",&sz);
		test = create_fl(sz,FLen);
		counter2=0;
		for(m=0;m<sz; m++)
		{
			for(J=0;J<FLen; J++)
			{
				fscanf(fp,"%f",&p);
				test[counter2][J]=p;
			}

		if(m<sz-1) fscanf(fp,"\n");
			counter2++;
		}
		fclose(fp);
		for(l=0;l<sz; l++)
		{
			get_netout(test[l],zk,HIDEN,OUT);
			k=99;
			MAX = -MIN;
			for(i=0;i<OUT; i++)
			{
				if((zk[i]>=MAX))
				{
					MAX=zk[i];
					k=i;
```

```
			}
		}

		if ((k==TAG) &&(zk[k]>Threshold) )
				CLASS[TAG]++;

		printf("\n TESTs [%d]: %.3f %.3f %.3f} TAG =%d count = %d",
			l,zk[0],zk[1],zk[2],TAG,CLASS[TAG]);
	}
GG++;
}// end of while loop

printf("\n %d classified out of %d",cntr,sz);
for(i=0;i<OUT; i++)
	printf("\n CLASS[%d] = %d",i,CLASS[i]);


   /* IMAGE TEXTURE CLASSIFICATION */

unsigned char **Im,**Im2;
float **Tx_img[FLen];
float *X1;		// input sample vector

printf("\n Enter test image data");
   scanf("%s", &fname);
   printf("\n Enter size of image");
   scanf("%d", &N);
   printf("\n Enter number of grey levels in image");
   scanf("%d", &L);
   printf("\n Enter texture window size for processing");
   scanf("%d", &n);
q=(n-1)/2;

printf("\n window %d x %d  and L = %d",n,n,L);

create_texture(fname,n,Tx_img);
X1 = (float *)malloc(sizeof(float)*FLen);

for(i=0; i<OUT;  i++)
	CLASS[i]=0;

Im=create_mem(N,N);
for(l=q;l<N-q;l++)
{
	for(m=q;m<N-q; m++)
	{
		for(J=0;J<FLen;J++)
		{
			X1[J]=Tx_img[J][l][m];
		}
		get_netout(X1,zk,HIDEN,OUT);
		k=99;
		MAX = -MIN;
		for(i=0;i<OUT; i++)
```

```
			{
				if((zk[i]>MAX))
				{
					MAX=zk[i];
					k=i;
				}
			}

			if((zk[k]>Threshold))
			{
				CLASS[k]++;
							// label classified pixels
				if(k==0) Im[l][m]=100;
				else if(k==1) Im[l][m]=190;
				else if(k==2) Im[l][m]=250;
			}
		}// for loop m

	}// end of for loop

	for(j=0;j<OUT; j++)
	printf("\n%s = %d  ",f_trn[j],CLASS[j]);

	printf("\n NET = %dX %dX%d",FLen,HIDEN,OUT);
	Im2 = smooth_img(Im,N,N,q);    // median filter

	strcat(fname, "_clss");
	fp  = fopen(fname,"wb");
	save_imagefile(fname,Im2,N,N);
	printf("\n\n CLASSIFIED IMAGE FILE saved in <  %s  >",fname);

	for(J=0;J<FLen;J++)
	{
			free_fl(Tx_img[J],N);
	}
	free_mem(Im,N);
	free_mem(Im2,N);
	free_network(HIDEN,OUT);

}
/*----------------------------------- END OF MAIN FUNCTION ----------------------------*/


void creat_NET(int HIDEN,int OUT)
{
	O    = (float *)malloc(sizeof(float)*OUT);
	zk   = (float *)malloc(sizeof(float)*OUT);
	err  = (float *)malloc(sizeof(float)*OUT);
	delta_k = (float *)malloc(sizeof(float)*OUT);
	delta_h = (float *)malloc(sizeof(float)*HIDEN);
	tk   = create_fl(OUT,OUT);
	W1   = create_fl(FLen,HIDEN);
	dw1  = create_fl(FLen,HIDEN);
	bw1  = (float *)malloc(sizeof(float)*HIDEN);
```

```
	d_bw1= (float *)malloc(sizeof(float)*HIDEN);
	ssum= (float *)malloc(sizeof(float)*HIDEN);
	bw2   = (float *)malloc(sizeof(float)*OUT);
	d_bw2 = (float *)malloc(sizeof(float)*OUT);

	    W2   = create_fl(HIDEN,OUT);
	dw2  = create_fl(HIDEN,OUT);// created and values set to 0.0

	H_v1 = (float *)malloc(sizeof(float)*HIDEN);
	H_y1 = (float *)malloc(sizeof(float)*HIDEN);
	O_v1 = (float *)malloc(sizeof(float)*OUT);
	O_y1 = (float *)malloc(sizeof(float)*OUT);
	}


void initialise_RAND()
{
	srand(4711);
}

float init_weights(float Lo, float Hi)
{
	float t;
	t = ((float)rand()/RAND_MAX) *(Hi-Lo)+Lo;
	return t;
}

void initialise_wts(int H, int O)
{
	int i,j;
	float a1,a2;

	    a1 = a2 =0.5;

	for(i=0;i<FLen; i++)
		for(j=0;j<H; j++)
			W1[i][j]=init_weights(-a1,a1);


	    for(i=0;i<H; i++)
	    {
	      d_bw1[i]=0.0;
	      delta_h[i] =0.0;
	      bw1[i] = init_weights(-a1,a1);
	      for(j=0;j<O; j++)
	      {
			d_bw2[j]=0.0;
			delta_k[j]=0.0;
			bw2[j]  =  init_weights(-a2,a2);
			W2[i][j]= init_weights(-a2,a2);
	      }
	    }

}
```

```
void display_wts(int H,int O)
{
	int i,j;

	for(i=0;i<FLen; i++)
	{
		for(j=0;j<H; j++)
			printf(" %.4f ",W1[i][j]);
		printf("\n");
	}

	for(i=0;i<H; i++)
	{
		for(j=0;j<O; j++)
			printf("%.4f ",W2[i][j]);
		printf("\n");
	}
	    printf("\n BIAS WTS. for HIDEN : ");
	for(i=0;i<H; i++)
		printf("%.4f ",bw1[i]);

	printf("\n of OUT layer  : ");
	for(j=0;j<O; j++)
		printf(" %.4f ",bw2[j]);
	    }


void set_tk(float **tk,int O)   // matrix for true vector of classes
{
	int i,j;

	for(i=0;i<O; i++)
		for(j=0; j<O; j++)
		{
			if ( i==j)
				tk[i][j] = 0.9;
			else
				tk[i][j] = 0.0;
		}

	for(i=0;i<O; i++)
	{
		for(j=0; j<O; j++)
			printf(" %.2f, ",tk[i][j]);
		printf("\n");
	}

}

float phi(float v)
{
	double x,y;

	x = (double)v;
```

```
        y = ( 1.0 + exp(-x));
        x = (1.0/y) ;
        if(x>HI)
                return 19.0;

        else if(x<LO)
                return -19.0;
        else
            return((float)(x));
}


void get_netout(float *I,float *zk,int HIDEN,int OUT)
{
        int i,k;
        float sum;;

        for(i=0;i<HIDEN; i++)
            {
                sum = 0.0;
                for(k=0; k<FLen; k++)
                    sum = sum + (W1[k][i]*I[k]);
                H_v1[i] = sum;
            }

        for(i=0;i<HIDEN; i++)
            {
                H_v1[i] = H_v1[i] + bw1[i];  // add bias weight
                H_y1[i] = phi(H_v1[i]);
            }

        for(i=0;i<OUT; i++)
            {
                sum = 0.0;
                for(k=0; k<HIDEN; k++)
                    sum = sum + (W2[k][i]*H_y1[k]);
                O_v1[i] = sum;
            }

        for(i=0;i<OUT; i++)
            {
                O_v1[i] = O_v1[i] + bw2[i];  // add bias weight
                O_y1[i] = phi(O_v1[i]);
                zk[i] = O_y1[i];

            }
}       //end


void TRAIN(int HIDEN,int OUT,float eta, float alpha, float **train, int *class_TAG, int sz)
{
        int g,count;
        int i,j,J,itr;
        float sum1,dw;
```

```
float MSE,p,MSE_old;
float eta0,alpha0;
float *X;
int FLAG,USED;

X  = (float *)malloc(sizeof(float)*FLen);
eta0 = eta;
alpha0 = alpha;
    //printf("\n in TRAIN sz = %d, eta = %f alpha = %f",sz,eta,alpha);
itr =0;
do{
        p=0.0;  USED =0;
        MSE = 0;
        for(count=0; count<TRNG; count++)
        {
                g = class_TAG[count];
                for(J=0;J<FLen; J++)
                {
                                X[J] = train[count][J];
                }
                FLAG =0;
                get_netout(X,zk,HIDEN,OUT);

                if((zk[0] >=LO) &&(zk[0]<=HI) &&(zk[1] >= LO) &&
                  (zk[1]<=HI)&&(zk[2]>=LO) &&(zk[2]<=HI))
                        FLAG =1;
                else
                        FLAG=0;
                if(FLAG)
                {
                        USED++;
                for(i=0;i<OUT; i++)
                        err[i] = tk[i]- zk[i];

                sum1 =0.0;
                for(i=0;i<OUT; i++)
                        sum1 = sum1 + err[i]*err[i];
                p = p + sum1;

                /***    BACKWARD  COMPUTATIONS       ***/
                for(i=0;i<OUT; i++)// adjust bias weights
                {
                        delta_k[i] = (float)(err[i]* a*O_y1[i]*(1.0 - O_y1[i]));
                        dw = d_bw2[i];
                        d_bw2[i] = eta*delta_k[i] + alpha*dw;   //bias =1.0;
                        bw2[i] =  bw2[i] + d_bw2[i];
                }

                // Update W2 weight matrix
                for(i=0;i<HIDEN; i++)
                        for(j=0;j<OUT; j++)
                {
                        dw = dw2[i][j];// save last dw in temp
                        dw2[i][j]= eta*delta_k[j]*H_y1[i]+ alpha*dw;
```

```
			W2[i][j]= W2[i][j] + dw2[i][j] ;
		}

		for(i=0;i<HIDEN; i++)
		{
			ssum[i]=0.0;;
			for(j=0;j<OUT; j++)
				ssum[i] = ssum[i] +  delta_k[j]*W2[i][j];

		}
		// Update W1 weight matrix
		for(i=0;i<HIDEN; i++)
			delta_h[i] =  a*H_y1[i]*(1.0 - H_y1[i])*ssum[i];

		for(j=0;j<HIDEN;j++)
		{
			for(i=0;i<FLen; i++)
			{
				dw = dw1[i][j];
				dw1[i][j] = eta* delta_h[j]*X[i] + alpha*dw;
				W1[i][j] = W1[i][j]+dw1[i][j];
			}
		}

		for(i=0;i<HIDEN; i++)
		{
			dw = d_bw1[i];
			d_bw1[i] = eta * delta_h[i]+ alpha*dw;
			bw1[i] = bw1[i] + d_bw1[i];
		}

		}// end of FLAG;
	}// end of one EPOCH
	MSE = p/(2.0*USED);
	itr++;
	MSE_old = MSE;

	if(!(itr%20))
		printf("\n ( itr = %d )TRNG  = %d MSE = %f USED
		=%d",itr,TRNG,MSE,USED);
}while ((itr<EPOCHS));
printf("\nitr = %d and  FINAL MSE = %f",itr,MSE);
free(X);
TEST(train,class_TAG,TRNG,HIDEN,OUT);

}

// testing  of  training samples
void TEST(float **train,int *class_TAG,int sz,int HIDEN,int OUT)
{
	int i,k,l;
	int J,B;
	int *CLASS;
	float MAX;
```

```
	CLASS =(int *)malloc(sizeof(int)*OUT);

	for(i=0;i<OUT; i++)
		CLASS[i]=0;

	for(l=0;l<sz; l++)
	{
		get_netout(train[l],zk,HIDEN,OUT);
		MAX = -MIN;
		k=99;

		for(i=0;i<OUT; i++)
		{
			if((zk[i]>=MAX))
			{
					MAX=zk[i];
					k=i;
			}
		}

		if((k==class_TAG[l])&&(zk[k]>Threshold)&&(zk[k]<=HI))
			CLASS[k]++;
	}
	B=SET1/5;
	if((CLASS[0]>=B) &&(CLASS[1]>=B)&&(CLASS[2]>=B))
	{
		for(i=0;i<OUT; i++)
		printf("C[%d]=%d  ",i,CLASS[i]);
	}

	free(CLASS);
}

void save_wts(char wt_file[150],int H,int O,float ETA,float ALPHA,float ERR)
{
FILE *fp;
int i,j;

	fp = fopen(wt_file,"w");
	for(i=0;i<FLen; i++)
	{	for(j=0;j<H; j++)
			fprintf(fp,"%.4f ",W1[i][j]);
	fprintf(fp,"\n");
	}

	for(i=0;i<H; i++)
		fprintf(fp,"%.4f ",bw1[i]);
	fprintf(fp,"\n");

	// record Output  layer bias wts
	for(i=0;i<H; i++)
	{	for(j=0;j<O; j++)
		fprintf(fp,"%.4f ",W2[i][j]);
	fprintf(fp,"\n");
```

```
	}
	// record OUT layer bias wts
	for(j=0;j<O; j++)
		fprintf(fp,"%.4f ",bw2[j]);
fclose(fp);
}

float **get_GLCM(unsigned char **B,int r,int c,int d,int theta)
{

	float **GLC;
	float **C;
	int i,j,g1,g2;

	GLC=create_fl(L,L);
	switch(theta)
	{	// updating GLCM for d=1,theta = 0 degree
	case 0:
			for(i=0;i<r;i++)
			{
				for(j=0;j<c-d;j++)
				{
					g1=(unsigned int)B[i][j];
					g2=(unsigned int)B[i][j+d];
					GLC[g1][g2]++;
					GLC[g2][g1]++;
				}
			}

		break;

	case 45:
				for(i=d;i<r;i++)
				{
					for(j=0;j<c-d;j++)
					{
						g1=(unsigned int)B[i][j];
						g2=(unsigned int)B[i-d][j+d];
						GLC[g1][g2]++;
						GLC[g2][g1]++;
					}
				}
		break;

	case 90:
			for(i=d;i<r;i++)
				{
					for(j=0;j<c;j++)
					{
						g1=(unsigned int)B[i][j];
						g2=(unsigned int)B[i-d][j];
						GLC[g1][g2]++;
						GLC[g2][g1]++;
					}
```

```
                    }
            break;

        case 135:
            for(i=d;i<r;i++)
                    {
                        for(j=d;j<c;j++)
                        {
                            g1=(unsigned int)B[i][j];
                            g2=(unsigned int)B[i-d][j-d];
                            GLC[g1][g2]++;
                            GLC[g2][g1]++;
                        }
                    }
            break;
        default: printf("\n A DEFAULT CASE...");

        } // end of switch

        // DISPLAY  GLCM matrix
/*      printf("\n theta = %d\n",theta);

        for(i=0;i<L; i++)
        {
            for(j=0;j<L; j++)
                printf("%2.0f,",GLC[i][j]);
            printf("\n");

        }
fflush(stdin); getchar();
*/

          float sum;
        sum =0.0;
        C=create_fl(L,L); // L = no of new DN

        for(i=0;i<L;i++)
            for(j=0;j<L;j++)
                sum = sum+ GLC[i][j];

        //printf("\n NORMALISING FACTOR = %f",sum);

        for(i=0;i<L;i++)
            for(j=0;j<L;j++)
                C[i][j] = GLC[i][j]/sum;

free_fl(GLC,L);
return C;
}//    end of GLCM

void cal_features(float **C,float f[FN])
{
        int i,j;
        float CON,ENT,ASM,IDM;
```

```
	float tmp,tmp2,sum;
	float x;

	tmp = tmp2=sum=0;

	for(i=0; i<L; i++)
		for(j=0; j<L; j++)
		{
			sum = sum + C[i][j]*C[i][j];

			x =(float)( (i-j)*(i-j));
			tmp = tmp + x*C[i][j];

			if(C[i][j]>0.0000001)
				tmp2 = tmp2 + C[i][j]*(float)log10((double)C[i][j]);

		}
	ASM = sum;
	CON = tmp;
	ENT = -tmp2;
	sum =0.0;
	   for(i=0; i<L; i++)	//IDM
	{
		for(j=0; j<L; j++)
		{
			tmp = (float)(i-j)*(i-j);
			tmp2 = 1.0/(1.0 + tmp);
			sum = sum + tmp2*C[i][j];
		}
	}
	IDM = sum;
	f[0]=CON;
	f[1]=ASM;
	f[2]=ENT;
	f[3]=IDM;
}

void create_texture(char fname[150],int n, float **Tx[FLen])
{
	unsigned char **Im;
	unsigned char **B;	// texture window for analysis

	float **G;		// GLCM
	int d,q;
	int l,m;
	int i,j;
	int I,J;

	float f1[FN],f2[FN],f3[FN],f4[FN];

	Im = read_img(fname,N,N); // assuming square image
	// memory for texture data
	for(i=0;i<FLen; i++)
		Tx[i]= create_fl(N,N);
```

```
q = (n-1)/2;
d = 1;                  // pixel spacing

B = create_mem(n,n);    // texture window

for(i=0;i<N-n;i++)
    {
     for(j=0;j<N-n; j++)
        {
            for(I = i,l=0; I<(i+n),l<n; I++,l++)
                for(J=j,m=0; J<(j+n),m<n; J++,m++)
                    B[l][m]=Im[I][J];

            G = get_GLCM(B,n,n,d,0);
            cal_features(G,f1);

                G = get_GLCM(B,n,n,d,45);
            cal_features(G,f2);

            G = get_GLCM(B,n,n,d,90);
            cal_features(G,f3);

                    G = get_GLCM(B,n,n,d,135);
            cal_features(G,f4);


            Tx[0][i+q][j+q]=f1[0];
            Tx[1][i+q][j+q]=f1[1];
            Tx[2][i+q][j+q]=f1[2];
            Tx[3][i+q][j+q]=f1[3];

            Tx[4][i+q][j+q]=f2[0];
            Tx[5][i+q][j+q]=f2[1];
            Tx[6][i+q][j+q]=f2[2];
            Tx[7][i+q][j+q]=f2[3];

            Tx[8][i+q][j+q]=f3[0];
            Tx[9][i+q][j+q]=f3[1];
            Tx[10][i+q][j+q]=f3[2];
            Tx[11][i+q][j+q]=f3[3];

            Tx[12][i+q][j+q]=f4[0];
            Tx[13][i+q][j+q]=f4[1];
            Tx[14][i+q][j+q]=f4[2];
            Tx[15][i+q][j+q]=f4[3];
        }
     if(!(i%50))
     {
     printf("\n In creat_text fn = %3d",i);
    // fflush(stdin); getchar();
     }

    }
```

```
		free_mem(B,n);
		free_mem(Im,N);

		for(i=0;i<FLen; i++)
			Normalise(Tx[i],N,N,q);

	}



void Normalise(float** Tx, int r,int c,int q)  // data normalisation in [0.1 – 0.9 ]
{
	int i,j;
	float pp;

	float  MAX  ;
	float  MIN ;
	float xd ;

	MAX = -MIN;
	for(i=q;i<r-q; i++)
	{
		for(j=q;j<c-q; j++)
		{
			if(Tx[i][j] < MIN)
				MIN = Tx[i][j];

			if(Tx[i][j] >MAX)
				 MAX = Tx[i][j];
		}
	}
	xd = MAX-MIN;
	for(i=q;i<r-q; i++)
	    for(j=q;j<c-q; j++)
		{
			pp = Tx[i][j] - MIN;
			Tx[i][j]=((pp/xd)* (HI-LO) )+ LO ;
		}
}

unsigned char **smooth_img(unsigned char **Im,int row, int col,int q)
{
	int i,j;
	int I,J,l,m;
	int sz;
	int off;
	unsigned int median_val;
	unsigned int B[FILTER][FILTER];
	unsigned char **img;

	sz = FILTER;
	off = (sz -1)/2;

	img = create_mem(row,col);
```

```
	for(i=0;i<row-sz;i++)
	{
		for(j=0;j<col-sz; j++)
		{
			for(I = i,l=0; I<(i+sz),l<sz; I++,l++)
				for(J=j,m=0; J<(i+sz),m<sz; J++,m++)
					B[l][m]=(unsigned int)Im[I][J];

			median_val = sort(B);
			img[i+off][j+off]= (unsigned char)median_val;
		}
	}

	return img;
	}


unsigned int sort(unsigned int win[FILTER][FILTER])
{
	int k,i,j,temp;
	unsigned int a[FILTER*FILTER];
	int sz;

	sz = FILTER*FILTER ;  // median  filter
	k=0;
	for(i=0; i<FILTER; i++)
		for(j=0; j<FILTER; j++)
		{
			a[k]= win[i][j];
			k++;
		}

	   for(i=0;i<sz; i++)
		{
			k=i;
			for(j=i+1; j<sz; j++)
			{
				if(a[j]<a[k])
					k=j;
			}
			temp = a[i];
			a[i]   = a[k];
			a[k]  = temp;
		}
return(a[(sz-1)/2]);
}
```

```
void free_network(int HIDEN, int OUT)
{
	free(O);
	free(zk);
	free(err);
	free(delta_k);
	free_fl(tk ,OUT);
	free_fl(W1,HIDEN);
	free_fl(dw1,HIDEN);
	free(bw1);
	free(d_bw1);
	free(ssum);
	free(bw2);
	free(d_bw2);
	free_fl(W2,OUT);
	free_fl(dw2,OUT);
	free(H_v1);
	free(H_y1);
	free(O_v1);
	free(O_y1);
}
```

```
/*------------------------------------------------------------------------------------------------
                    HEADER FILE
// func.h     includes commonly used functions in image classification
------------------------------------------------------------------------------------------------*/

#include<stdlib.h>
#include<math.h>
#include<string.h>

#define  DN 256  // Input image data is 8 bits

char fname[150];

/*----------------------------Function prototypes -----------------------------------------*/

unsigned char **create_mem(int r,int c);

void free_mem(unsigned char **, int);

float **create_fl(int,int );

void free_fl(unsigned char **, int);

unsigned char **read_img(char [150],int ,int);

void save_imagefile(char[150],unsigned char **, int,int);

void HIST_EQU(unsigned char **img,int r, int c,int);

void show_hist(int *f,int len);

/*-------------------------------------------------------------------------------------*/

unsigned char **create_mem(int r,int c)
{
	unsigned char **a;
	int i,j;

	a = (unsigned char **)malloc(sizeof(unsigned char *)*r);
	for(i=0;i<r; i++)
		a[i] = (unsigned char *)malloc(sizeof(unsigned char )*c);

	if(a==NULL)
	{
		printf("\n ERROR  ******* OUT OF MEMORY ********");
		exit(0);
	}

	for(i=0;i<r;i++)
		for(j=0;j<c;j++)
				a[i][j]=0;
	return a;

}

void free_mem(unsigned char **mat, int R)
{
	int i;
```

```
    for(i=0;i<R;i++)            // Release allocated MEMORY
        free(mat[i]);

    free(mat);

}

// Creates 2D array of floats for GLCM
float **create_fl(int r,int c)
{

    float **a;
    int i,j;

  a = (float **)malloc(sizeof(float *)*r);

    for(i=0;i<r; i++)
        a[i] = (float *)malloc(sizeof(float )*c);

    if(a==NULL)
    {
        printf("\n *********** OUT OF MEMORY ********");
        exit(0);
    }
    for(i=0;i<r;i++)
        for(j=0;j<c;j++)
                a[i][j]=0.0;
    return a;

}

// Release allocated memory to 2D float array
void free_fl(float **mat, int R)
{
    int i;
    for(i=0;i<R;i++)
        free(mat[i]);
    free(mat);
}

// reads image file and loads into 2D array
unsigned char **read_img(char fname[150],int R,int C)
{

    int i,j;
    unsigned char **Im,pix;
    FILE *fp;

    Im = create_mem(R,C);  // input image
    fp = fopen(fname,"rb");
    printf("\n FILE NAME   %s",fname);
    if(fp==NULL)
    {
        printf("File NOT Found");
```

```
        exit(0);
    }

    for(i=0;i<R; i++)
        for(j=0;j<C; j++)
        {
            fread(&pix,1,1,fp);
            Im[i][j]=pix;
        }
    fclose(fp);
    printf("\n I have read and closed < %s >",fname);
return Im;
}


// Saves image data into a specified file
void save_imagefile(char fname[30],unsigned char **img, int R,int C)
{

    FILE *fp;
    int i,j;

    fp  = fopen(fname,"wb");

    for(i=0;i<R;i++)
        for(j=0;j<C; j++)
            fprintf(fp,"%c",img[i][j]);

    fclose(fp);
    printf("\n\nIMAGE FILE saved in  <  %s  >",fname);

}

// Maps 256 DN levels to a reduced number of DN levels
void HIST_EQU(unsigned char **img,int r, int c,int L)
{

int freq[DN],cumul_freq[DN],grey;
int i,j;
float T ;
float scal;
int LUT[DN];

for(i=0;i<DN; i++)
    freq[i] = 0;

for(i=0;i<r; i++)
    for(j=0;j<c; j++)
    {
        grey = img[i][j];
        freq[grey]++;
    }

    show_hist(freq,DN);
```

```
// CALCULATE CUMULATIVE FREQ...
cumul_freq[0]=freq[0];

for(i=1;i<DN; i++)
        cumul_freq[i]= cumul_freq[i-1]+freq[i];

show_hist(cumul_freq,DN);

T = (float)r*c;  // total pixels in image
scal = (L-1)/T;

printf("\n DNQuantization = %d levels, SCALE = %f",L,scal);
fflush(stdin); getchar();

// PREPARE LOOK UP TABLE

for(i =0; i<DN; i++)
{

        LUT[i] = (int)(scal*cumul_freq[i] +0.5);
}
printf("\n DISPLAY LUT ..............");

        show_hist(LUT,DN);
// UPdate image for new DN values from LUT

for(i=0;i<r; i++)
        for(j=0;j<c; j++)
        {
                grey = img[i][j];
                img[i][j] = LUT[grey];

        }
}

// Displays DN values and their frequencies
void show_hist(int *f,int len)
{
        int i;
        printf("\n... HISTOGRAM DISPLAY...");

        for(i=0;i<len; i++)
        {
                printf("[%3d] = %6d",i,f[i]);
                if(!(i%100))
                {
                        fflush(stdin); getchar();
                }
        }

}
```