Classification of Internet Users using Discriminant Analysis and Neural Networks

António Nogueira[†], M. Rosário de Oliveira[‡], Paulo Salvador[†], Rui Valadas[†], António Pacheco[‡]

[†]University of Aveiro/Institute of Telecommunications, Campus de Santiago, 3810-193 Aveiro, Portugal

e-mail: nogueira@av.it.pt, salvador@av.it.pt, rv@det.ua.pt

[‡]Instituto Superior Técnico - UTL, Department of Mathematics and CEMAT, Av. Rovisco Pais, 1049-001 Lisboa, Portugal

e-mail: rosario.oliveira@math.ist.utl.pt, apacheco@math.ist.utl.pt

***Abstract*— The (reliable) classification of Internet users, based on their hourly traffic profile, can be advantageous in several traffic engineering tasks and in the selection of suitable tariffing plans. For example, it can be used to optimize the routing by mixing users with contrasting hourly traffic profiles in the same network resources or to advise users on the tariffing plan that best suits their needs. In this paper we compare the use of Discriminant Analysis and artificial Neural Networks for the classification of Internet users. The classification is based on a predefined set of clusters which, in the first case, is used to define the function that best discriminates among clusters and, in the second case, is used to train the neural network.**

We classify the Internet users based on a data set measured at the access network of a Portuguese ISP. Using Cluster Analysis performed over the first half of users we have identified three groups of users with similar behavior. The classification methods were applied to the second half of users and the obtained classification results compared with those of cluster analysis performed over the complete set of users. Our findings indicate that Discriminant Analysis outperforms Neural Networks as a classification procedure.

Keywords: Internet traffic characterization, Traffic measurements, Discriminant analysis, Neural networks.

I. Introduction

The classification of Internet users into groups of similar hourly traffic utilization can be exploited to enhance several traffic engineering tasks and in the selection of tariffing plans. It is usually advantageous to apply different policies to users with markedly different behavior. We give two examples:

- Traffic engineering - One way to improve network utilization is to mix in the same set of network resources users that have a contrasting hourly traffic behavior (e.g. users whose periods of higher utilization are in disjoint time intervals). Thus, in general, it is beneficial for network operators to cluster their users into groups of similar hourly traffic utilization and to apply routing policies that are a function of the clustering solution. Following a measurement phase, a user can be classified into one of the predefined groups and its routing can be adjusted accordingly; this may free some resources which, in turn, will allow additional users to access the network.
- Tariffing plans - Hourly based tariffs can be used, for example, to promote Internet access in the least busy hours. Clustering users according to their hourly traffic utilization allows ISPs to assess whether or not hourly based tariffs are advantageous, and to decide on the number and type of tariffs. This may be the case when a significant number of users follows a non-flat usage profile. In case an ISP offers hourly based tariffs, classification can be used to advise users on the type of tariff they should select, based on their previous usage.

In this paper we compare the use of Discriminant Analysis and artificial Neural Networks for the classification of the Internet users. Users are characterized by their average transfer rates for downloaded traffic measured in half-hour periods (over one day).

Discriminant Analysis (DA) [1] is a multivariate statistical technique whose aim is to find the so called *discriminant functions* which highlight existing differences between coherent groups of objects, i.e., groups of objects with similar characteristics. The discriminant rules provide a way to classify each new object into one, and only one, of the previously defined groups. In particular, we will use linear DA, which seeks for linear functions of the variables, (linear discriminant functions), that best separate the groups.

Artificial Neural Networks (NNs) have been successfully used in a number of applications due to their advantageous properties like parallel processing of information, capacity to handle non-linearity and quick adaptability to system dynamics. They can be trained to efficiently recognize patterns of information in the presence of noise and non-linearity and classify information using those patterns. These properties can also be exploited for classifying Internet users. However, NNs have also some weaknesses: their inputs must be managed to be in a particular range, which requires additional transformations and manipulations of the input data; they cannot explain the results, and they may converge on an inferior solution. Neural networks usually converge to some solution for any given training set. However, there is no guarantee that this solution provides the best model of the data. Thus, we must use a test set to determine when a model provides good enough performance to be used on unknown data.

Since the input data can have highly correlated variables and/or exhibit peculiar behaviors for a small number of users, Principal Component Analysis (PCA) is usually employed as

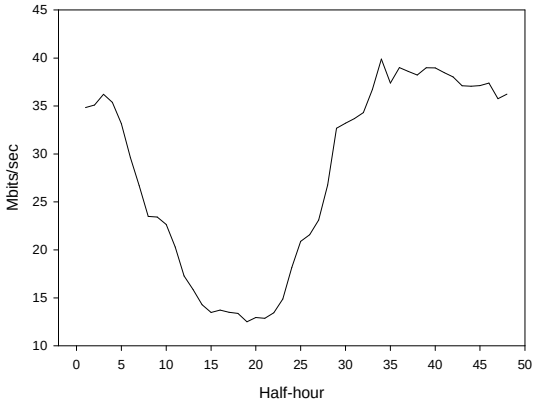

Fig. 1. Transfer rate of downloaded aggregate traffic.

a way to reduce the dimensionality (viz. number of variables, [2]). This choice was particularly pertinent to apply in the case of the NN.

In order to train the NN, estimate the linear discriminant functions, and evaluate the performance of DA and NN in classifying Internet users, a previous classification of the users based on Cluster Analysis (CA) was used.

The classification results obtained show that NNs are able to classify Internet users. However, the results have shown that DA outperforms NNs. Moreover, DA is easier to use than NNs and the results of DA have a simple interpretation, whereas those of NNs do not.

The paper is organized as follows. Section II gives an overview of the traffic trace analyzed. In Section III we give some background on Cluster Analysis and present the clustering results. Then, in sections IV and V we describe the two classification methods under comparison: Discriminant Analysis and artificial Neural Networks. Section VI presents and discusses the results. Finally, in Section VII we state our conclusions.

II. Overview of the traffic trace

Our analysis resorts to a data trace measured in a Portuguese ISP that uses a CATV network and offers several types of services, characterized by the following maximum allowed transfer rates (in Kbit/s) in the downstream/upstream directions: 128/64, 256/128 and 512/256. The trace was measured on November 9, 2002, a Saturday. The measurements were detailed packet level measurements, where the arrival instant and the first 57 bytes of each packet were recorded. This includes information on the packet size, the origin and destination IP addresses, the origin and destination port numbers, and the IP protocol type. The traffic analyzer was a 1.2 GHz AMD Athlon PC, with 1.5 Gbytes of RAM and running WinDump. No packet drops were reported by WinDump in both measurements.

Users were identified by matching IP addresses with accounting information. The data set includes 3432 users. In this paper, we classify users based on their individual download transfer rates measured in half-hour intervals. We will denote the transfer rate of a user (in Kbits/s) in the k-th half-hour interval by X_k, $k = 1, 2, \ldots, 48$. Fig. 1 shows the transfer rates of the downloaded aggregate traffic as a function of the time period, along the day. The aggregate transfer rates exhibit coherent hourly profiles, showing a quasi-sinusoidal shape, with the lowest utilization in the morning period and the highest one in the afternoon period. However this representation hides groups of users with specific hourly profiles, markedly distinct from the aggregate hourly profile.

III. Cluster analysis

Classification techniques such as DA or NN rely on a predefined set of groups that can be determined using cluster analysis. The aim of this methodology is to partition a set of objects into groups or clusters in such a way that objects in the same group are similar, whereas objects in different clusters are distinct. The concept of cluster is linked with the concept of proximity between objects and groups of objects [3]. There are two common approaches to clustering the observations: hierarchical and partitioning.

The hierarchical clustering techniques proceed by either a successive series of merges (agglomerative hierarchical methods) or by a series of successive divisions (divisive hierarchical methods). The agglomerative methods start with as many clusters as objects and end with only one cluster, containing all the objects. The divisive methods work in the opposite direction. These methods are based on a measure of proximity between two objects and a criterion, relying on the distance between clusters, to decide which are the two closest clusters to be merged in each step of the agglomerative hierarchical procedure. Different approaches to measure the distance between clusters give rise to different hierarchical methods. A widely used method is the *Wards's method*, also known as the *incremental sum of squares method*, that uses the within-cluster (squared) and between-cluster (squared) distances to decide which clusters should be merged.

In this paper, cluster analysis will be based on the (partitioning around) *medoids method*, which performs better with the dataset under analysis [4]. In this method, the analyst has to decide in advance how many clusters, say K, he wants to consider. The method starts by choosing K medoids, here denoted by $m_1, m_2, \ldots, m_K$. These are representative objects that are chosen such that the total (Euclidean) distance of all objects to their nearest medoid is minimal, i.e., the algorithm finds a subset $\{m_1, \ldots, m_K\} \subset \{1, \ldots, n\}$ (where n is the number of objects to be clustered) which minimizes the function

$$\sum_{i=1}^{n} \min_{t=1,\ldots,K} d_{im_t}.$$

Each object is then assigned to the cluster corresponding to the nearest medoid. That is, object i is assigned to cluster C_j whose associated medoid, m_j, is nearest to object i, i.e., $d_{im_j} \leq d_{im_t}$, for all $t \in \{1, 2, \ldots, K\}$. In the present study, the users are the objects and the variables are the half-hour interval transfer rates along the day. The number of clusters was selected based on the average sillhouette width [3], as well as the dendrograms obtained from the Ward's method, and the coherence between the partitions obtained by the medoids and the Ward's method. The same approach was followed in [4].

The data set described in Section II is used to form clusters in two different situations: considering all users (as in [4]) and considering only half of the users selected randomly. The results of the first analysis will be used to evaluate the classification methods. The second case simulates a system were only half of the data, which will be called the training set, is initially available to obtain the clusters. After that the system classifies the other users (the remaining half of the original data set) in one of the previously identified clusters (i.e. profiles of utilization).

As in [4], the data set is transformed according to:

$$Y_j = \ln(1 + X_j) \tag{1}$$

for $j = 1, \ldots, 48$. This transformation helps smoothing the variability of Internet utilization in half-hour intervals along the day, which was seen to increase with the daily average Internet utilization.

The two partitions based on the complete data set and the training set have a similar interpretation. Figures 2 and 3 represent the average half-hour transfer rates along the day within each cluster, for the partition obtained from the complete data set and from the training set, respectively. Thus, the first cluster, C_1, contains users with high transfer rates in all day periods, the users in C_2 have low transfer rates in the morning and high transfer rates in the afternoon and C_3 contains users with low transfer rates in all day periods. The interpretation of the clusters in both partitions is summarized in Table II. According to Table I, cluster C_3 has the highest percentage of users and C_1 the lowest. The main differences in the two partitions are in clusters C_2 and C_3. In Table III we show a contingency Table crossing the two partitions. All the 1216 users of C_3 in the partition based on the training set remained in the same cluster in the partition based on the complete data set. However, around 18% ($\approx 272/1504 \times 100\%$) of the users of C_3 in the partition based on the complete data set and belonging to the training set were assigned to C_2 in the partition based on the training set.

In order to illustrate the differences between the partitions we calculated the first two principal components, based on the correlation matrix of the complete data set, and projected the observations (associated with the 1716 users from the training set) in these two orthogonal directions (see Appendix). The first principal component (PC1) can be interpreted as an average of Internet utilization along the day and the second principal component (PC2) as a measure of the contrast between the morning and afternoon utilization (vide [4]). In Fig. 4 we represent the partition based on the complete data set and in Fig. 5 the partition based on the training set. We represent users associated with different clusters using different marks. These Figures illustrate the fact that a great number of users clustered in C_3, in the partition based on the complete data set, are assigned to C_2 by the partition based on the training set.

TABLE I
CLUSTER SIZES

	Complete set		Training set	
	Size	Percentage	size	Percentage
C1	145	4.23%	87	5.07%
C2	266	7.75%	413	24.07%
C3	3021	88.02%	1216	70.86%

TABLE II
CLUSTERS INTERPRETATION

Cluster	Interpretation
C1	high transfer rate in all periods
C2	low/high transfer rate in the morning/afternoon
C3	low transfer rate in all periods

IV. CLASSIFICATION USING DISCRIMINANT ANALYSIS

Discriminant analysis is a multivariate technique concerned with the separation among different sets of objects and the classification of a new object into one of the previous defined groups. Linear discriminant analysis seeks for linear functions of the variables, called discriminant functions, that best separate g groups characterized by p random variables.

R. Fisher suggested a sensible procedure to distinguish between groups [5]. The first discriminant function is the linear combination of the observed variables, $\mathbf{l}_1^t \mathbf{x} = l_{11}x_1 + \ldots + l_{1p}x_p$, that maximizes the ratio of the between-group sum of squares to the within-group sum of squares. Which means that the separation is made in such a way that within each group the objects are as similar as possible but, at the same time, the groups are as different as possible. A maximum of $s = \min(g-1, p)$ discriminant functions can be defined as linear combinations of the observed variables, uncorrelated with the previous ones, which verify the same optimality criteria.

Let $\mathbf{x}_{ik}$ be the vector of observations on user i for group k (with n_k users) where $\mathbf{x}_{ik} = (x_{i1k}, \ldots, x_{ipk})^t$. The sample mean vector for group k is $\bar{\mathbf{x}}_{\cdot k} = (\bar{x}_{\cdot 1k}, \ldots, \bar{x}_{\cdot pk})^t$, where $\bar{x}_{\cdot jk} = \sum_{i=1}^{n_k} x_{ijk}/n_k$, $j = 1, \ldots, p$, $k = 1, \ldots, g$. $\bar{\mathbf{x}}_{\cdot k}$ is also called the centroid of cluster k. The within-group sum of squares, W, is defined by

$$W = \sum_{k=1}^{g} \sum_{i=1}^{n_k} (\mathbf{x}_{ik} - \bar{\mathbf{x}}_{\cdot k})(\mathbf{x}_{ik} - \bar{\mathbf{x}}_{\cdot k})^t,$$

and the between-group sum of squares matrix is

$$\begin{aligned} B &= \sum_{k=1}^{g} \sum_{i=1}^{n_k} (\bar{\mathbf{x}}_{\cdot k} - \bar{\mathbf{x}}_{\cdot\cdot})(\bar{\mathbf{x}}_{\cdot k} - \bar{\mathbf{x}}_{\cdot\cdot})^t \\ &= \sum_{k=1}^{g} n_k (\bar{\mathbf{x}}_{\cdot k} - \bar{\mathbf{x}}_{\cdot\cdot})(\bar{\mathbf{x}}_{\cdot k} - \bar{\mathbf{x}}_{\cdot\cdot})^t, \end{aligned}$$

where the overall sample mean is $\bar{\mathbf{x}}_{\cdot\cdot} = \sum_{k=1}^{g} \sum_{i=1}^{n_k} \mathbf{x}_{ik}/n$, and $n = n_1 + \ldots + n_g$. It can be proved that $\mathbf{l}_j$ is the eigenvector associated with the j-th largest eigenvalue of the matrix $W^{-1}B$, scaled such that $\mathbf{l}_j^t S_p \mathbf{l}_j = 1$, where $S_p = W/(n-g)$.

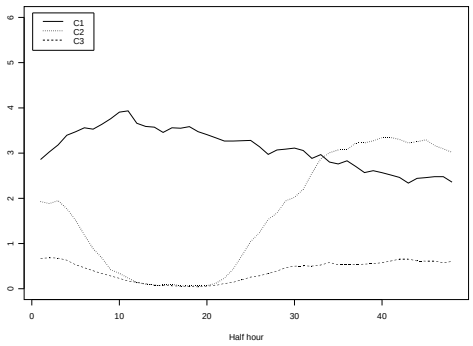

Fig. 2. Half hour sample mean for each cluster, complete data set.

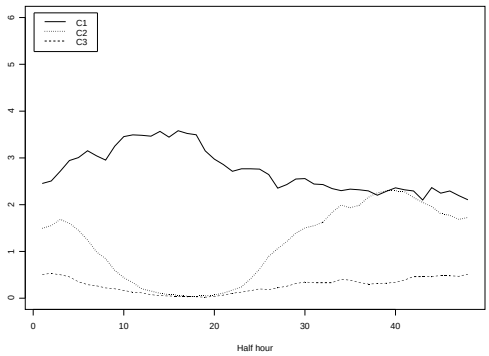

Fig. 3. Half hour sample mean for each cluster, training set.

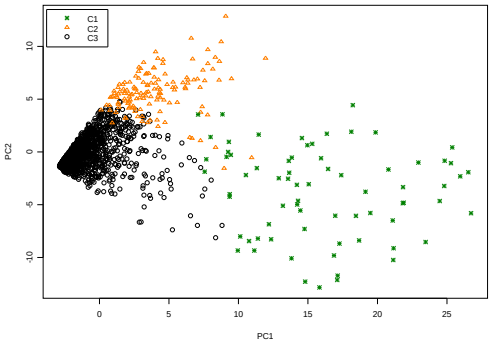

Fig. 4. Scores of the users (training set) written in the first two principal components, obtained from the complete data set. The clusters were formed taking into consideration all users.

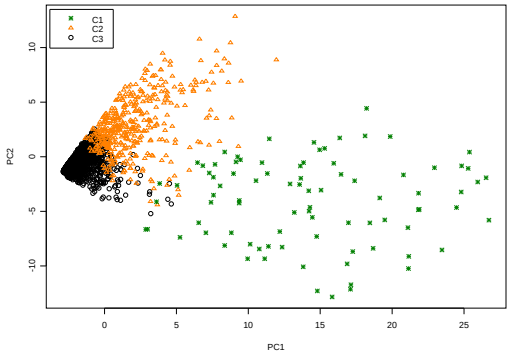

Fig. 5. Scores of the users (training set) written in the first two principal components, obtained from the complete data set. The clusters were formed taking into consideration the users from the training set.

TABLE III
CONTINGENCY TABLE FOR THE TWO PARTITIONS

	Training set			
Complete set	C_1	C_2	C_3	
C_1	68	4	0	72
C_2	3	137	0	140
C_3	16	272	1216	1504
	87	413	1216	1716

The Fisher's discriminant functions were derived to obtain a representation of the data that separates the population as much as possible. However, it can be used to produce a discrimination rule [6]. Let $(\mathbf{l}_1^t \bar{\mathbf{x}}_{\cdot k}, \ldots, \mathbf{l}_s^t \bar{\mathbf{x}}_{\cdot k})^t$ be the sample mean vector of the discriminant scores associated with group k. To classify the object $\mathbf{x}_0$ we have to evaluate the discriminant functions on this object $(\mathbf{l}_1^t \mathbf{x}_0, \ldots, \mathbf{l}_s^t \mathbf{x}_0)^t$. The object should be allocated to the group for which its square Euclidean distance to the group sample mean vector $(\mathbf{l}_1^t \bar{\mathbf{x}}_{\cdot k}, \ldots, \mathbf{l}_s^t \bar{\mathbf{x}}_{\cdot k})^t$ is the smallest. If the groups have very different sizes, prior probabilities associated with each group can be used to obtain a better classification rule.

V. CLASSIFICATION USING NEURAL NETWORKS

In general, Neural Networks (NNs) include several layers of *neurons* or processing units: the input layer that receives inputs from the outside, one or more hidden layers that receive inputs only from other processing units, and an output layer that receives the outputs of a previous layer of processing units. Each input value of a processing unit (that corresponds to a single element of the network input or to each output of the previous layer) is multiplied by a weight and the summation of all these values together with a scalar bias (specific to each neuron) are applied to a transfer function (previously defined for each layer), producing the output value of the neuron. There are three major connection topologies that define how data flows between the input, hidden, and output processing units: feed-forward, limited recurrent, and fully recurrent networks [7], [8]. Feed-forward networks are appropriate for solving problems where all the information can be presented to the neural network at once.

The application of a NN to solve a particular problem involves two phases: a training phase and a test phase. In the training phase, the training set is input to the NN which iteratively adjusts network weights and biases in order to produce an output that matches, within a certain degree of accuracy, a previously known result (named target set). In

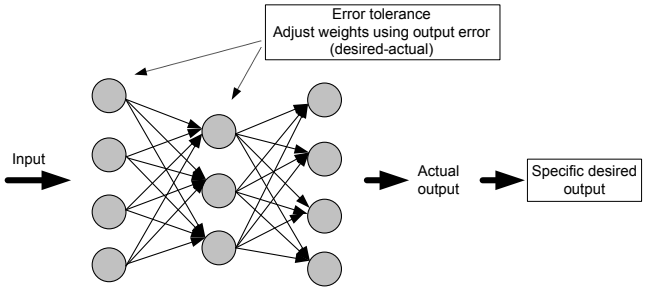

Fig. 6. Back propagation networks.

the test phase, a new input is presented to the network and a new result is obtained based on the network parameters that were calculated during the training phase. For classification problems, the input vectors are mapped to the desired classification categories. The training of the neural network amounts to setting up the correct set of discriminate functions to correctly classify the inputs. There are two learning paradigms (supervised or non-supervised learning) and several learning algorithms that can be applied, depending essentially on the type of problem to be solved.

The combination of topology, learning paradigm and learning algorithm define a NN model. Back propagation is an appropriate learning algorithm for training multilayer feed-forward networks for vector classification, modeling and time-series forecasting [9]. It is a general purpose learning algorithm, that is powerful but also expensive in terms of computational requirements for training. A back propagation NN uses a feed-forward topology, supervised learning, and the back propagation learning algorithm. A back propagation network with a single hidden layer of processing elements can model any continuous function to any degree of accuracy (given enough processing elements in the hidden layer) [9].

The basic back propagation algorithm consists of three steps (Fig. 6). The input vector is presented to the input layer of the network. These inputs are propagated through the network until they reach the output units. This forward pass produces the actual or predicted output vector. Because back propagation is a supervised learning algorithm, the desired outputs are given as part of the training set. The actual network outputs are subtracted from the desired outputs and an error signal is produced. This error signal is then the basis for the back propagation step, whereby the errors are passed back through the neural network by computing the contribution of each hidden processing unit and deriving the corresponding adjustment needed to produce the correct output. The connection weights are then adjusted and the NN has just learned from an experience. Two major learning parameters are used to control the training process of a back propagation network: the learning rate is used to specify whether the neural network is going to make major adjustments after each learning trial or if it is only going to make minor adjustments; the momentum is used to control possible oscillations in the weights, which could be caused by alternately signed error signals. These two parameters are the ones that usually produce the most impact on the NN training time and performance.

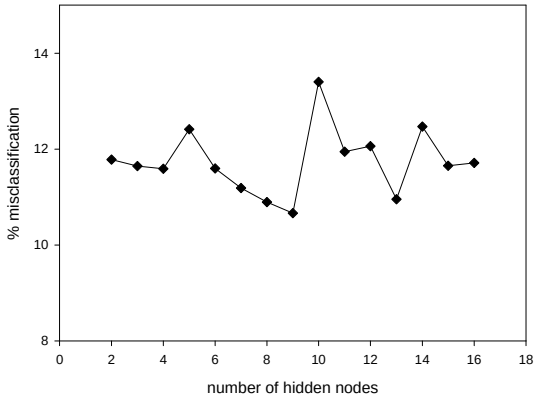

Fig. 7. Percentage of misclassification vs number of nodes in the hidden layer.

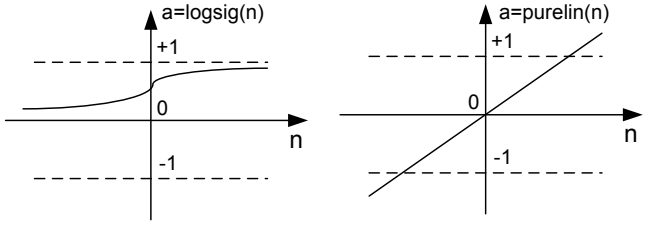

Fig. 8. Log-sigmoid and linear transfer functions.

In our case, each input vector should contain the transfer rate in each one of the k-th half-hour intervals, X_k, $k = 1, 2, \ldots, 48$, that is, should have a dimension of 48 elements and has to be classified in each one of the 3 clusters. We have used Principal Component Analysis to reduce the dimensionality of the original data. In our case, the dimension of each input vector is reduced to 22 by eliminating the principal components that contribute less than 0.5% to the total variation in the data set.

For a problem of this dimension, a conventional feed-forward back propagation network with three layers seems to be appropriate. The input layer will have 22 neurons, corresponding to the dimensionality of the input vectors, and the output layer will have 1 neuron. The number of nodes in the hidden layer is empirically selected such that the performance function, which is the mean square error for feed-forward networks, is minimized. We have considered neural networks with a variable number of neurons in the hidden layer, trained each neural network using the training set and tested it using the test set. Fig. 7 plots the percentage of misclassification (compared with the classification performed by cluster analysis based on the training set) when the trained networks (with different number of neurons in the hidden layer) are used to classify the test set. From these results, it can be seen that increasing the number of hidden nodes beyond 9 does not give any improvement in performance, so the number of hidden nodes in the NN used for classification was selected to be 9.

The NN structure is shown in Fig. 9. The scalar $w^i_{j,k}$ represents the weight value corresponding to layer i, $i = 1, 2, 3$, that is multiplied by the input k of neuron j, where j and k have different ranges depending on the network layer. The scalar b^i_j represents the bias associated with neuron j of

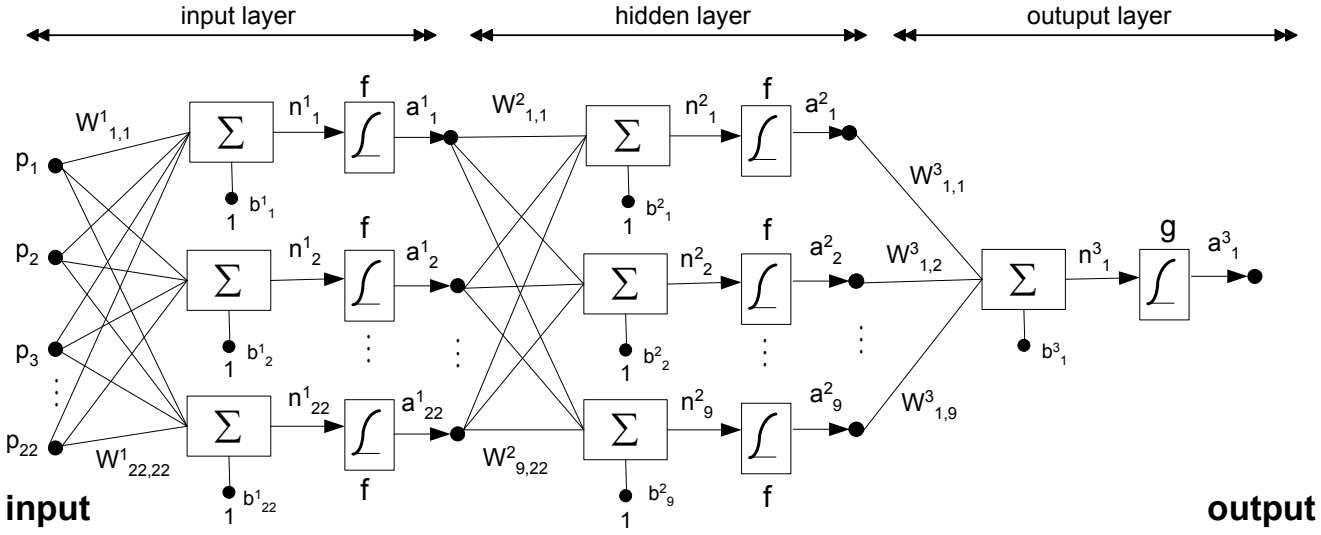

Fig. 9. Architecture of the Neural Network used for classifying Internet users.

layer i.

For the input and hidden layers, a log-sigmoid transfer function (represented by **f** in Fig. 9) is used (Fig. 8), generating outputs between 0 and 1 as the neuron´s input goes from negative to positive infinity. For the output layer, a linear transfer function (represented by **g** in Fig. 9) is used (Fig. 8). Multiple layers of neurons with nonlinear transfer functions allow the network to learn nonlinear and linear relationships between input and output vectors. A neural network including biases, a sigmoid layer, and a linear output layer is capable of approximating any function with a finite number of discontinuities [7], [8]. Automated Bayesian regularization is used to improve generalization of the neural network, in order to avoid overfitting [10].

Using the above notation, the output a_j^i of each neuron is given by: $a_j^1 = f(w_{j,k}^1 p_k + b_j^1)$, $j,k = 1,...,22$, where p_k represents the k^{th} input, for the input layer; $a_j^2 = f(w_{j,k}^2 a_k^1 + b_j^2)$, $j = 1,...,9$, $k = 1,...,22$, for the hidden layer; and $a_1^3 = g(w_{1,k}^3 a_k^2 + b_1^3)$, $k = 1,...,9$, for the output layer.

VI. Results

The original data set, containing 3432 users, is divided in two subsets of equal size. First, the classification of the users of the first set, called the training set, is used to estimate the discriminant functions in the case of DA, and to train the NN. Afterwards, the users of the second set, called the test set, are classified in one of the 3 groups previously determined by the partition based on the training set. This partition was discussed in Section III.

For each of the 3 clusters, we have computed the respective vector of sample means, also called the centroid of the cluster. Accordingly, the simplest classification rule that can be devised to classify users of the test set it to classify an user in the cluster associated to the nearest centroid.

Starting from the classification of the users from the training set, linear DA was used to classify the users of the test set. Namely, the Fisher discriminant function based on the training set was used to classify the users of the test set. As the 3 clusters of the training set have very different sizes, the error rates of the classification based on DA were calculated considering prior probabilities proportional to the group sizes (vide Table I). The apparent error rate is 6.82%. Since it is known that this rate underestimates the true error rate a leave-one-out error rate was also obtained, leading to an error rate of 7.98%. To calculate this error rate each user of the training set was left out and the other 1715 were used to estimate the discriminant functions. The observation left out is then classified. As both error rates reported are low, we were expecting good results when classifying the test set.

Prior to the use of NN a PCA analysis applied to the data associated to the training set was carried out and 22 principal components, explaining more than 0.5% of the total variability, have been retained and the NN analysis was applied to them instead of the original set of 48 variables.

The results of the classification of the users of the test set based on the three different classification methods (DA, NN, and the distance to the nearest centroid) are summarized in Table IV. The results of these classification procedures are compared with the partition obtained through cluster analysis carried out over the complete data set, i.e., considering all 3432 users. This comparison is made based on a similarity index that measures the percentage of users that are classified in the same group (taking as a reference the partition obtained when considering the complete data set), reported in Table IV.

From Table IV we can conclude that DA is the classification method that lead to the best results. From the analysis of the contingency tables V, VI, and VII, we can argue that the majority of the classification error produced by the three classification procedures comes from users belonging to the third cluster, C_3, being wrongly classified in C_2. This means that the classification procedures have some trouble in distinguishing between users with low transfer rates all day long and users with low transfer rate in the morning and higher transfer rates in the afternoon. This is an expected result since the partition based on the training set revealed the same

TABLE IV

CLASSIFICATION OF THE USERS OF THE TEST SET

Classification method	C_1	C_2	C_3	Measure of similarity
Neural networks	92	275	1349	88.00%
Discriminant analysis	68	256	1392	92.13%
Distance to centroids	75	315	1326	88.69%
Clusters obtained from the complete data set	73	126	1517	

TABLE V

CROSSING THE NN CLASSIFICATION AND THE ORIGINAL PARTITION BASED ON THE COMPLETE DATA SET

	NN			
Complete set	C_1	C_2	C_3	
C_1	57	11	5	73
C_2	9	113	4	126
C_3	26	151	1340	1517
	92	275	1349	1716

TABLE VI

CROSSING THE DA CLASSIFICATION AND THE ORIGINAL PARTITION BASED ON THE COMPLETE DATA SET

	DA			
Complete set	C_1	C_2	C_3	
C_1	63	10	0	73
C_2	0	126	0	126
C_3	5	120	1392	1517
	68	256	1392	1716

TABLE VII

CROSSING THE CLASSIFICATION ON THE NEAREST CENTROID AND THE ORIGINAL PARTITION BASED ON THE COMPLETE DATA SET

	Centroid			
Complete set	C_1	C_2	C_3	
C_1	72	1	0	73
C_2	2	124	0	126
C_3	1	190	1326	1517
	75	315	1326	1716

problem. DA and the classification procedure based on the nearest centroid do not classified users of C_1 into C_3 neither users of C_2 into C_3, this is not true to NN. The users from C_2 were all classified by DA in the right cluster.

VII. CONCLUSIONS

The classification of Internet users into groups of similar hourly traffic utilization can be applied to increase the efficiency of several traffic engineering tasks and to help in the definition and selection of tariffing policies. This paper addresses the classification of Internet users into groups according to their average transfer rate for downloaded traffic measured in half-hour periods (over one day). Two different techniques were considered: Discriminant Analysis (DA) and artificial Neural Networks (NNs). In order to perform the training of the NN, estimate the linear discriminant functions of the DA, and evaluate the performance of both DA and NN in classifying Internet users, a previous classification of the users based on Cluster Analysis needs to be determined. We analyzed a data set measured at the access network of a Portuguese ISP. Using the first half of users, we have identified three groups of users with similar hourly traffic utilization. The classification methods were then applied to the second half of users. To evaluate the classification methods, we compared the classification results with those obtained from Cluster Analysis performed over the complete set of users. Our results indicate that Discriminant Analysis outperforms Neural Networks as a classification procedure.

APPENDIX

PRINCIPAL COMPONENT ANALYSIS

Given a set of n observations on the random variables $X_1, X_2, \ldots, X_p$, the k-th principal component (PC k) is defined as the linear combination,

$$Z_k = \alpha_{k1}X_1 + \alpha_{k2}X_2 + \ldots + \alpha_{kp}X_p$$

such that the loadings of Z_k, $\boldsymbol{\alpha}_k = (\alpha_{k1}, \alpha_{k2}, \ldots, \alpha_{kp})^t$, have unitary Euclidean norm, maximum variance and PC k, $k \geq 2$, is uncorrelated with the previous PCs, which in fact means that $\boldsymbol{\alpha}_i^t \boldsymbol{\alpha}_j = 0$ if $i \neq j$ and $\boldsymbol{\alpha}_i^t \boldsymbol{\alpha}_i = 1$. Thus, the first principal component is the linear combination of the observed variables with maximum variance. The second principal component verifies a similar optimal criteria and is uncorrelated with PC 1, and so on. As a result, the principal components are indexed by decreasing variance, i.e., $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p$, where λ_r denotes the variance of PC r and p is the maximum number of PCs ($n > p$).

It can be proved [2] that the vector of loadings of the k-th principal component, $\boldsymbol{\alpha}_k$, is the eigenvector associated with the k-th highest eigenvalue, λ_k, of the covariance matrix of the observed variables. Therefore, the k-th highest eigenvalue of the covariance matrix is the variance of PC k, i.e. $\lambda_k = \text{Var}(Z_k)$.

The proportion of the total variance explained by the first r principal components is

$$\frac{\lambda_1 + \ldots + \lambda_r}{\lambda_1 + \ldots + \lambda_p}. \quad (2)$$

If this proportion is close to one, than there is almost as much information in the first r principal components as in the original p variables. In practice, the number r of considered principal components should be chosen as small as possible, taking into account that the proportion of the explained variance, (2), should be large enough.

Once the loadings of the principal components are obtained, the score of individual i on PC j is given by

$$z_{ij} = \alpha_{j1}x_{i1} + \alpha_{j2}x_{i2} + \ldots + \alpha_{jp}x_{ip}$$

where $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^t$ is the data corresponding to individual i.

REFERENCES

[1] G. J. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition*, Wiley, 1992.

[2] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag, 1986.

[3] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley, 1990.

[4] R. de Oliveira, R. Valadas, A. Pacheco, and P. Salvador, "Cluster analysis of internet users based on hourly traffic utilization," in *First International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks (HET-NETs '03)*, July 2003.

[5] J. D. Jobson, *Applied Multivariate Data Analysis. Volume II: Categorical and Multivariate Methods*, Springer-Verlag, 1992.

[6] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*, Prentice-Hall, Inc, 1982.

[7] L. Fausett, *Fundamentals of Neural Networks*, Prentice Hall, 1994.

[8] K. Gurney, *An Introduction to Neural Networks*, UCL Press, 1997.

[9] H. Demuth and M. Beale, *Neural Network Toolbox Users Guide*, The MathWorks, Inc., 1998.

[10] F. Foresee and M. Hagan, "Gauss-newton approximation to bayesian regularization," in *Proceedings of the 1997 International Joint Conference on Neural Networks*, 1997, pp. 1930–1935.