DATA MINING FOR DIGITAL MOBILE TELECOMMUNICATION NETWORK'S QUALITY OF SERVICE PERFORMANCE MEASUREMENTS

Pekko Vehviläinen

Institute of Automation and Control, Tampere University of Technology,
P.O. Box 692, FIN-33101 Tampere, Finland

Abstract: Digital mobile telecommunication networks are complex systems and thus their management and optimization are challenging tasks. The subscriber expectations constitute the Quality of Service (QoS). The operating personnel have to measure the network in terms of the QoS. By analyzing the information in the measurements, they can manage and improve the QoS. Two data mining methods were applied to actual GSM network performance measurements. The personnel's prior knowledge of the network and their possible inexperience of the theory behind the methods was taken into account. *Copyright © 2005 IFAC*

Keywords: data processing, quality, telecommunication, networks, classifiers, self-organizing systems, knowledge-based systems.

1. INTRODUCTION

Telecommunications have developed into cellular radio networks, which enable subscribers to connect and communicate regardless of their location and movement. The second-generation networks (2G) digital mobile telecommunication networks, whose most widely spread realizations are based on the Global System for Mobile communications standard (GSM).

The subscribers, connected to the network via their mobile stations, expect network availability, connection throughput, and affordability. Moreover, the connection should not degrade or be lost abruptly as the user moves within the network area. The user expectations constitute Quality of Service (QoS), specified as '*the collective effect of service performances, which determine the degree of satisfaction of a user of a service*' (ITU-T E.800). The operating personnel have to measure the network in terms of QoS. By analyzing the information of their measurements, they can manage and improve the quality of their services.

However, because the operating staff is easily overwhelmed by hundreds of measurements, the measurements are aggregated Key Performance Indicators (KPI).

Personnel expertise with the KPIs and the problems occurring in the cells of the network vary widely, but at least the personnel know the desirable KPI value range. Their knowledge may be based on simple rules such as 'if any of the KPIs is unacceptable, then the state of a cell is unacceptable.' The acceptance limits of the KPIs and the labeling rules are part of the *a priori* knowledge for analysis.

Information needed to analyze QoS issues exists in KPI data, but sometimes it is not easy to recognize. The techniques of Knowledge Discovery in Databases (KDD) and data mining help to find useful information in the data.

The most important criterion for selecting data mining methods was their suitability as tools for the operating staff of a digital mobile telecommunications network to alleviate their task of interpreting QoS-related information from measured data. For this paper, two methods were chosen that fulfilled the criterion: classification trees and self-organizing map type neural networks.

In particular, the automatic inclusion of prior knowledge in preparing the data is a novelty because *a priori* knowledge has so far been overlooked in any previous work on the subject (Vehviläinen 2004).

2. KNOWLEDGE DISCOVERY IN DATABASES AND DATA MINING

Knowledge Discovery in Databases (KDD), a multi-step, interactive, and iterative process requiring human involvement (Fayyad *et al*. 1996), aims to find new knowledge about an application domain.

2.1 Knowledge Discovery in Databases.

The KDD process (figure 2) consists of consecutive tasks, out of which data mining produces the patterns of information for interpretation. The results of data mining then have to be evaluated and interpreted in the result interpretation phase before we can decide whether the mined information qualifies as knowledge (Fayyad *et al*. 1996). The discovery

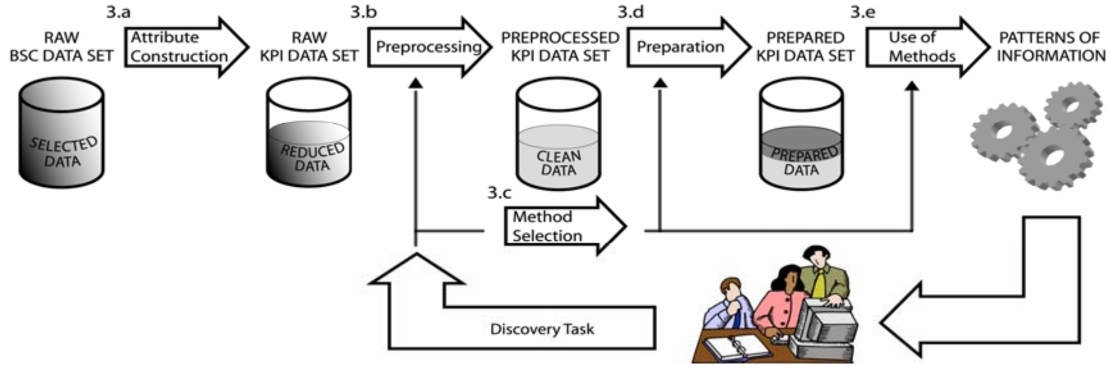


Figure 1. Data mining for QoS analysis of mobile telecommunications networks (Vehviläinen 2004).

process is repeated until new knowledge is extracted from the data. Iteration distinguishes KDD from the straightforward knowledge acquisition by measurement.

2.2 *Data Mining*

Data mining is a partially automated KDD subprocess, whose purpose is to *nontrivially extract implicit and potentially useful patterns of information from large data sets* (Vehviläinen 2004). Specifically, data mining for QoS analysis of mobile telecommunications networks involves five consecutive steps (figure 1), four of them closely related to the use of data mining methods: attribute construction, method selection, preprocessing and preparation.

3. QUALITY PERFORMANCE MEASUREMENTS

A Key Performance Indicator (KPI) is considered an important performance measurement. In GSM network management, KPIs may be used for several purposes; thus selecting KPIs for analysis is a subjective matter (Laiho *et al*. 2002). The QoS related KPIs in this paper are based on the measurement of SDCCH and TCH logical channels and handovers (HO).

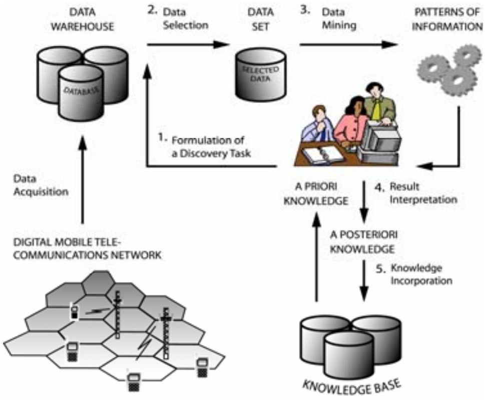


Figure 2. KDD for QoS analysis of network is an interactive and iterative process in five consecutive steps (Vehviläinen 2004).

Intrinsic QoS analysis depends on quality-related KPI measurements available from the network elements. In the GSM network, the most important services are bearer and teleservices. The intrinsic QoS of a bearer service means that the network's radio coverage is available for the subscriber outdoors and indoors. However, availability of the network is necessary for teleservices; therefore, KPI data contains information about those cells where the bearer service is degraded.

Teleservices require a functional bearer service and a successful connection. Speech, short message service, fax, and data depend on the bearer service, and the subscriber's need for teleservices has to be filled in most cases.

Five of the KPIs related to QoS refer to the use of the logical channels of the GSM network that require physical channels. The radio interface of the GSM network uses both Time Division Multiple Access (TDMA) and Frequency Division Multiple Access (FDMA) for receiving and transmitting information (Penttinen 2001). TDMA means that each frequency channel is divided into eight repeated time slots, see figure 3. A separate time slot is a physical channel, and one physical channel can contain logical channels defined as Traffic CHannels (TCH) for call data and Control CHannels (CCH) for transmitting service data between the nodes of the network, see figure 4.

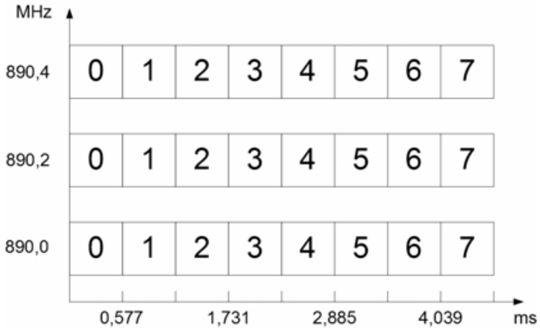


Figure 3. GSM radio interface uses both TDMA and FDMA for transmission of information. The picture shows an example of three TDMA frames comprising eight time slots on three frequencies.

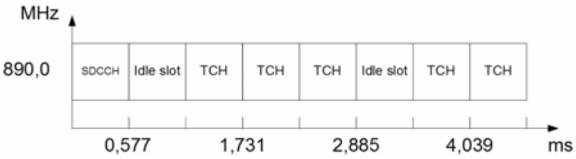


Figure 4. Logical channels like SDCCH and TCH are carried within physical channels (time slots).

CCHs are used to signal or synchronize data and can be divided into four categories: broadcast, common, dedicated and Cordless Telephony System (CTS) CCHs (GSM 05.02). The performance measurement data set in this paper contains measurements only about a Stand-alone Dedicated Control CHannel (SDCCH).

A SDCCH serves many communication purposes between the network and a mobile station. First of all, it is used when a call is initiated. Secondly, radio interface protection signalling, used for encryption, is carried in the SDCCH. Furthermore, a SDCCH is also needed to deliver short messages, when a mobile station is not involved in a call (Penttinen 2001).

Finally, successful handovers are an important element of a mobile network. Handover (HO) means switching a connection from a physical channel to another. Handovers are divided into intra-cell and inter-cell handovers.

KPI Limits Based on A Priori *Knowledge*. The analyst knows roughly the good, normal, bad, and unacceptable range of KPI values. For instance, his *a priori* knowledge of *SDCCH Success* is that it is normal for KPI values to be close to 100. He also knows that if the value drops below 100, a problem ensues because the signaling channels should be available all the time. To ensure that his *a priori* knowledge is justified, the analyst can plot the KPIs' Probability Density Function (PDF) estimates, assuming that the data is acquired from a network that has been under normal operational control. PDF estimates are plotted so that variable data is divided into slots along the horizontal axis, which represents a KPI's value. Each slot has an equal number of data points, which means that the height of the slot is proportional to the density of data points over the range of one slot.

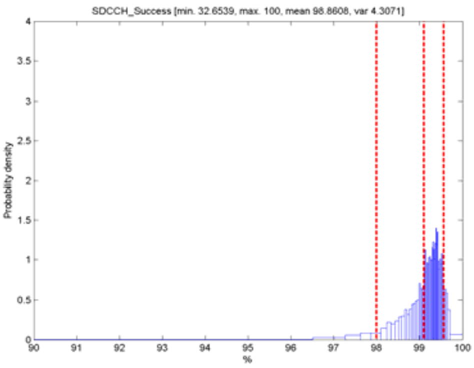


Figure 5. *A priori* limits of value ranges of KPI SDCCH Success.

Based on the limits and his *a priori* knowledge, the operator can then write out his rules to interpret the data as a labeling function.

When the analyst scrutinizes the plotted KPI PDF-distributions, he can justify and possibly refine the limits of a good, normal, bad and unacceptable.

Table 1 Ranged KPI values with corresponding label range limits. The *a priori* limits given by a domain expert are greyed out

KPI	**unacceptable**	**bad**	**normal**	**good**
SDCCH Access	≤ 99.00%	-	> 99.00%	-
SDCCH Success	≤ 98.00%	≤ 99.10%	≤ 99.56%	> 99.56%
TCH Access	≤ 99.00%	-	> 99.00%	-
TCH Success	≤ 98.00%	≤ 98.75%	≤ 99.35%	> 99.35%
HO Failure	≥ 5.00%	≥ 2.08%	≥ 0.91%	< 0.91%
HO Failure Due to Blocking	≥ 5.00%	≥ 0.23%	≥ 0.08%	< 0.08%
TCH Drops	≥ 2.00%	≥ 0.57%	≥ 0.19%	< 0.19%

3.1 Labelling Function

A labeling function is necessary for labeling observations with a decision indicator value, which in turn is necessary for a supervised learning algorithm. The function can be thought of as a formulated inference rule of the operator judging the behavior of the network. The inference and its limits (see table 1) are the operator's *a priori* knowledge. The values of the rest of the limits resulted from subjective inference from the PDF estimate distributions of real operator data set. The measured data set had 3069 observations (93 days of cumulative measurements from 33 GSM network cells) of the seven KPIs.

The function makes use of logical inference based on predetermined limits of the performance indicators. As a result, it labels each observation as good, normal, bad, or unacceptable. It does not include information about the causes of changes in the observations but indicates simply whether a cell is in a more or less acceptable state (good, normal, bad) or whether a state requires immediate attention (unacceptable). The labeling function is a set of four rules on the seven quality related KPIs, that is, *SDCCH Access*, *SDCCH Success*, *TCH Access*, *TCH Success*, *HO Failure*, *HO Failure Due to Blocking* and *TCH Drops*.

The labeling function labels the observations in the KPI data set according the following four rules, which are applied in descending order so that the label is the one that first applies. Thus the state of the network is

- ❑ **unacceptable** if any quality-related KPI is rated as unacceptable,
- ❑ **bad** if any quality-related KPI is rated bad,

- **good** if KPIs *SDCCH Access* and *TCH Access* are classified as normal and KPIs *SDCCH Success*, *TCH Success*, *HO Failure*, *HO Failure Due to Blocking* and *TCH Drops* are rated good,
- **normal** if KPIs *SDCCH Access* and *TCH Access* are classified as normal and KPIs *SDCCH Success*, *TCH Success*, *HO Failure*, *HO Failure Due to Blocking*, and *TCH Drops* are rated either normal or good.

The labels can be coded numerically as in table 2.

Table 2 Labels of the decision class indicator

State of a cell	*Decision class indicator*
good	1
normal	2
bad	3
unacceptable	4

4. CLASSIFICATION TREES

In data mining, a common classification method is the identification of a classification tree (Han & Kamber 2001), which suits both classification and prediction. In this paper the application of Classification And Regression Trees (CART) algorithm is applied to the QoS KPIs.

The benefits of binary splitting, a simple splitting condition, and the CART's being able to process both numerical (KPI data) and nominal values, were the main criteria why the CART was chosen for the classification tree algorithm for this paper.

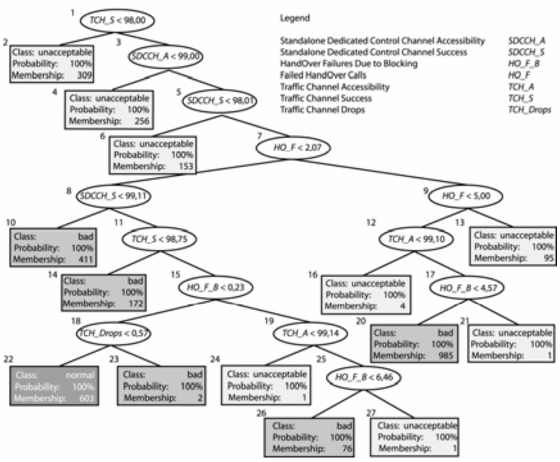
Application. Before analysis with the CART, the KPI data set was preprocessed by removing observations with missing values and prepared by subjecting the data to the labeling function (see section 3.1).

With the aid of the tree growing theory (Breiman *et al*. 1984), the whole KPI data set of 3069 observations was analyzed with the CART algorithm. The Gini index of diversity (equation 1) was chosen as the score function, and tree growing was set to terminate if any further growth reduced the observations in a node to less than 20 observations.

$$gini(t) = 1 - \sum_{v} p^{2}(v \mid t) \qquad (1)$$

where $p(v|t)$ is the estimated probability that an KPI observation is of class v (good, normal, bad, unacceptable), given that it falls into node t.

The CART algorithm resulted in the tree structure shown in figure 6. The tree has nine levels and 27 nodes, 14 of which are terminal nodes and 13 splitting nodes. The nodes are numbered from 1 to 27 with their identification number increasing from left to right and moving up to the next level after passing the rightmost node on a level.



Legend
Standalone Dedicated Control Channel Accessibility SDCCH_A
Standalone Dedicated Control Channel Success SDCCH_S
HandOver Failures Due to Blocking HO_F_B
Failed HandOver Calls HO_F
Traffic Channel Accessibility TCH_A
Traffic Channel Success TCH_S
Traffic Channel Drops TCH_Drops
1 TCH_S < 98,00
2 Class: unacceptable Probability: 100% Membership: 309
3 SDCCH_A < 99,00
4 Class: unacceptable Probability: 100% Membership: 256
5 SDCCH_S < 98,01
6 Class: unacceptable Probability: 100% Membership: 153
7 HO_F < 2,07
8 SDCCH_S < 99,11
9 HO_F < 5,00
10 Class: bad Probability: 100% Membership: 411
11 TCH_S < 98,75
12 TCH_A < 99,10
13 Class: unacceptable Probability: 100% Membership: 95
14 Class: bad Probability: 100% Membership: 172
15 HO_F_B < 0,23
16 Class: unacceptable Probability: 100% Membership: 4
17 HO_F_B < 4,57
18 TCH_Drops < 0,57
19 TCH_A < 99,14
20 Class: bad Probability: 100% Membership: 985
21 Class: unacceptable Probability: 100% Membership: 1
22 Class: normal Probability: 100% Membership: 603
23 Class: bad Probability: 100% Membership: 2
24 Class: unacceptable Probability: 100% Membership: 1
25 HO_F_B < 6,46
26 Class: bad Probability: 100% Membership: 76
27 Class: unacceptable Probability: 100% Membership: 1

Figure 6. Classification tree of KPI data.

The higher the split node number of the KPI, the less important the KPI is in separating large pure groups of observations within the data set.

Examining the 14 terminal nodes, we notice that they are all pure nodes (with 100% class probability in each terminal node). Seven of the nodes are classified as unacceptable (nodes 2, 4, 6, 13, 16, 21, 24, and 27), five bad (nodes 10, 14, 20, 23, and 26), and one normal (node 22). The tree had no good terminal nodes.

Examination of the oval-shaped split nodes in figure 3 reveals that most splits (8 out of 13) are based on KPI PDF-estimates' label range boundaries (see table 1). This is not surprising because the tree is structured based on the decision indicator, which in turn is based on the labeling function (section 3.1), which again preclassifies observations according to label range boundaries.

Is this circular reasoning? Yes, if we are interested only in boundary values, but no, if we seek to identify those KPIs and their corresponding boundaries that separate observation groups in the data set. Splits along the label range boundaries have been added in table 3 (derived from table 1), and the alignment is indicated with a node number in parenthesis.

Table 3 Splits of a pruned tree vs. KPI discretization limits.

KPI	Unacceptable	Bad	Normal	Good
SDCCH Access	≤ 99.00% (node 3)	-	> 99.00%	-
SDCCH Success	≤ 98.00% (node 5)	≤ 99.10% (node 8)	≤ 99.56%	> 99.56%
TCH Access	≤ 99.00%	-	> 99.00%	-
TCH Success	≤ 98.00% (node 1)	≤ 98.75% (node 11)	≤ 99.35%	> 99.35%
HO Failure	≥ 5.00% (node 9)	≥ 2.08% (node 7)	≥ 0.91%	< 0.91%
HO Failure Due to Blocking	≥ 5.00%	≥ 0.23% (node 15)	≥ 0.08%	< 0.08%
TCH Drops	≥ 2.00%	≥ 0.57%	≥ 0.19%	< 0.19%

5. SELF-ORGANIZING MAP

If we have only limited *a priori* knowledge, or we need to check our prior knowledge on the data, we have to apply an unsupervised or self-organized learning method to look for features that are not known before our analysis but that describe our data. One such method is the Self-Organizing Map (SOM), an unsupervised neural network, introduced by Professor Teuvo Kohonen in 1982.

Concepts. The SOM is used mainly to visualize data. The SOM algorithm creates a set of prototype vectors, which represent a training data set, and projects the prototype vectors from *n*-dimensional input space - *n* being the number of variables in the data set - onto a low-dimensional grid. The resulting grid structure is then used as a visualization surface to show features in the data (Vesanto & Alhoniemi 2000).

The created prototype vectors are called *neurons*, connected via neighborhood relations. The training phase of a SOM exploits the neighborhood relation in that parameters are updated for a neuron and its neighboring units.

The neurons of a SOM are organized in a low-dimensional grid with a local lattice topology. The most common combination of local and global structures is the two-dimensional hexagonal lattice sheet, which is preferred in this paper as well.

Theory. Let $\boldsymbol{x} \in \Re^n$ be a randomly chosen observation from data set *X*. Now, the SOM can be thought of as a nonlinear mapping of the probability density function $p(\boldsymbol{x})$ in the observation vector space on a lower (two in our case) dimensional support space. Observation $\boldsymbol{x}$ is compared with all the weight vectors $\boldsymbol{w}_\mathrm{i}$ of the map's neurons, using the Euclidian distance measure $\|\boldsymbol{x} - \boldsymbol{w}_\mathrm{i}\|$.

Among all the weight vectors, the closest match $\boldsymbol{w}_c$ is chosen based on Euclidian distance, to observation $\boldsymbol{x}$ and call neuron *c* (*c* is the neuron's identification number on the map grid) related to $\boldsymbol{w}_c$ the Best-Matching Unit (BMU):

$$\|\boldsymbol{x} - \boldsymbol{w}_c\| = \min_i \|\boldsymbol{x} - \boldsymbol{w}_i\| \qquad (2)$$

After the BMU is found, denoted by *c*, its weight vector $\boldsymbol{w}_c$ is updated so that it moves closer to observation $\boldsymbol{x}$ in the input space. The update rule for the all the weights of the SOM is

$$\boldsymbol{w}_i(t+1) = \boldsymbol{w}_i(t) + \alpha(t)\, h_{ci}(t)\left[\boldsymbol{x} - \boldsymbol{w}_i(t)\right] \qquad (3)$$

where *t* is an integer-discrete time index, $\alpha(t)$ the learning rate function, and $h_{ci}(t)$ the neighborhood function, and $\boldsymbol{x}$ a randomly drawn observation from the input data set. Note that $h_{ci}(t)$ is calculated separately in map the dimension (two), whereas $\boldsymbol{x}$ and weight vectors $\boldsymbol{w}_\mathrm{i}$ have the dimension of the input space (seven in our case).

The learning rate is chosen so that the update effect decreases during the SOM's training phase. One such rate is

$$\alpha(t) = \frac{\alpha_0}{1 + (kt)/T} \qquad (4)$$

where α_0 is the initial value of the learning rate, *k* some arbitrarily chosen coefficient, and *T* the training length.

The neighborhood kernel around the BMU can be defined in several ways, one possibility is the Gaussian function denoted by

$$h_{ci}(t) = e^{-\frac{\|r_c - r_i\|}{2\sigma_t^2}} \qquad (5)$$

where σ_t is the kernel radius at time *t*, $\boldsymbol{r}_c \in \Re^2$ the map coordinates of the BMU, and $\boldsymbol{r}_i \in \Re^2$ the map coordinates of the nodes in the neighborhood.

Application. Like with the CART, the data set was preprocessed by removing the missing values since they are problematic in the SOM algorithm (Kohonen 1995). The variables in the training data set must be rescaled. Should the data have very different scales, the variables with high values are likely to dominate training when the SOM algorithm minimizes the Euclidian distance measure between weight vectors and observations (Vesanto *et al*. 2000).

The variables are commonly scaled so that the variance of each variable is one. But since the ranges of the variables were known *a priori*, that information was used for scaling (Vehviläinen 2004).

To present SOM information in an easily interpretable form, the value of each variable is shown on the map in a variable-specific figure instead of showing all variables in one figure. Such separate figures are called component planes.

Each component plane has a relative distribution of one KPI. The values in component planes are visualized in shades of gray. These values were scaled so that white or light shading represents preferable KPI values and black or dark shading unwanted KPI values. On the side of each component plane, we placed a colorbar to link the shading and actual KPI values. Note that the shading is specific to each component plane. The component planes of the trained SOM are shown in figure 7. In addition, the component planes show the *a priori* information of the labeling function, that is, the value of the decision variable of observations with most occurrences in the node.

We can immediately see that the unwanted values of *SDCCH Success*, *TCH Success*, and *TCH Drops* of the right side of the component planes are almost black.

SDCCH Success may also take unwanted values separately from *TCH Success* and *TCH Drops,* since the nodes in the top left corner are dark, whereas the component plans of *TCH Success* and *TCH Drops* are light in those nodes.

Furthermore, we can see that *TCH Access* correlates with *HO Failure due to Blocking*, since the nodes in the low left corner are dark in both planes.

HO Failure has its worst values in the nodes in the bottom right corner, which are dark. *HO Failure* is somewhat connected to *SDCCH Access*, because its component plane is gray in the same nodes. *SDCCH Access* has its worst values quite independent of the rest of the KPIs.

Hit hexagons show that most observations were distributed among the top and bottom rows of the map and in the middle. The *a priori* knowledge seems to match the component planes well, for the nodes that match normal states are located in the top middle section of the map. The worst observations fall on the left and right sides and in the bottom corners of the map.

6. CONCLUSION

In this paper, data mining was presented as a tool to manage quality of service in digital telecommunications networks. Two data mining methods and *a priori* knowledge were applied to a real QoS data set to interpret and summarize the information content of KPIs. These methods - CART and SOM - were found well matured and suited for the data mining application.

According to the results, the CART is best suited for ruling out the most important KPIs and detecting potential outliers in the data, and the SOM for visualizing data features and checking *a priori* decision making.

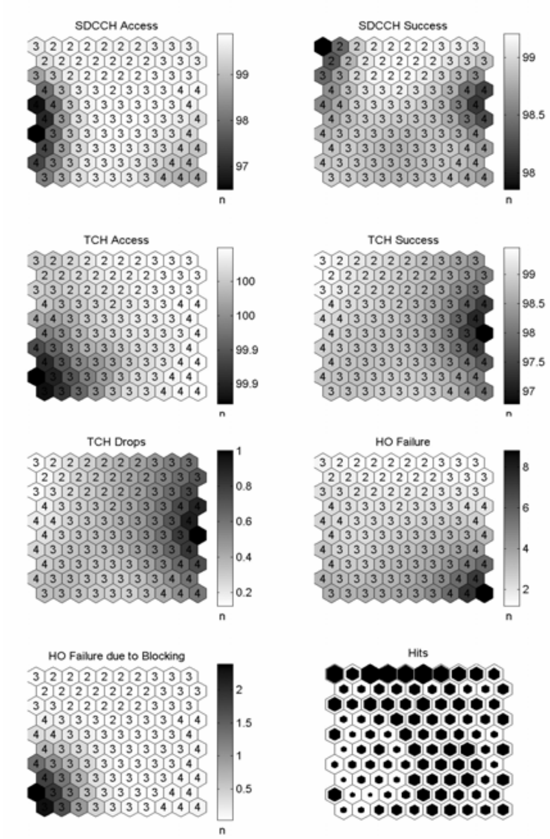


Figure 7. SOM component planes and relative hit counts of nodes. The numbers are labels from the labeling function.

REFERENCES

Breiman, L., Friedman, J., Olshen, R. & Stone, C. (1984). *Classification and Regression Trees*. Boca Raton, FL: Chapman & Hall/CRC Press LLC.

ETSI, TS 100 908 V8.10.0. (2001). *GSM Technical Specification 05.02: Digital Cellular Telecommunications System (Phase 2+); Multiplexing and Multiple Access on the Radio Path.*

Fayyad, U., Piatetsky-Shapiro, G. & Smyth, P. (1996). From Data Mining to Knowledge Discovery. An Overview. In: Fayyad, U., Piatetsky-Shapiro, G., Smyth, P. & Uthurusamy, R. (eds.) *Advances in Knowledge Discovery and Data Mining. Cambridge,* MA: The MIT Press, pp. 1-34.

Han, J. & Kamber, M. (2001). *Data Mining: Concepts and Techniques*. San Francisco, CA, USA: Morgan Kaufmann Publishers.

International Telecommunication Union. (1994). Terms and definitions related to quality of service and network performance including dependability. *ITU-T Recommendation E.800*.

Kohonen, T. (1995). *Self-Organizing Maps*. Berlin: Springer-Verlag.

Laiho, J., Korteniemi, A. Djupsund, M., Toivonen, M. & Grandell, J. (2002). Radio Network Optimisation Process. In: Laiho, J., Wacker, A. & Novosad, T. (eds.) *Radio Network Planning and Optimisation for UMTS*. Chichester, England: John Wiley & Sons, Ltd., pp. 329-363.

Penttinen, J. (2001). *GSM-tekniikka. Järjestelmän toiminta ja kehitys kohti UMTS-aikakautta*. Helsinki, WSOY. (In Finnish).

Vehviläinen, P. (2004). *Data Mining for Managing Intrinsic Quality of Service in Digital Mobile Telecommunications Networks*. Thesis (Doc. Tech.). Tampere University of Technology.

Vesanto, J. & Alhoniemi, E. (2000). Clustering of the Self-Organizing Map. *IEEE Transactions on Neural Networks*, **11** 3, pp. 586-600.

Vesanto, J., Himberg, J., Alhoniemi, E. & Parhankangas, J. (2000). *SOM Toolbox for Matlab 5. Report A57*. Helsinki University of Technology.