LAPPEENRANTA UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF INFORMATION TECHNOLOGY

PREDICTING DIFFUSION OF INNOVATIONS WITH SELF-ORGANISATION AND MACHINE LEARNING

The topic of the master's thesis has been accepted in departmental council of department of Information Technology, February 12, 2003.
Supervisors: Professor Heikki Kälviäinen, Professor Seppo Pitkänen

Lappeenranta, March 6, 2003

Jarmo Ilonen
Laserkatu 2 D 52
53850 Lappeenranta
Tel. 040-5895888
ilonen@lut.fi

ABSTRACT

Lappeenranta University of Technology

Department of Information Technology

Jarmo Ilonen

Predicting diffusion of innovations with self-organisation and machine learning

Master's thesis

2003

75 pages, 24 figures, 6 tables and 2 appendices.

Supervisors: Professor Heikki Kälviäinen, Professor Seppo Pitkänen

Keywords: diffusion of innovations, neural computing, Self-Organizing Map, SOM, differential evolution, prediction

The main subject of this master's thesis was predicting diffusion of innovations. The prediction was done in a special case: product has been available in some countries, and based on its diffusion in those countries the prediction is done for other countries. The prediction was based on finding similar countries with Self-Organizing Map (SOM), using parameters of countries. Parameters included various economical and social key figures. SOM was optimised for different products using two different methods: (a) by adding diffusion information of products to the country parameters, and (b) by weighting the country parameters based on their importance for the diffusion of different products. A novel method using Differential Evolution (DE) was developed to solve the latter, highly non-linear optimisation problem.

Results were fairly good. The prediction method seems to be on a solid theoretical foundation. The results based on country data were good. Instead, optimisation for different products did not generally offer clear benefit, but in some cases the improvement was clearly noticeable. The weights found for the parameters of the countries with the developed SOM optimisation method were interesting, and most of them could be explained by properties of the products.

TIIVISTELMÄ

Lappeenrannan teknillinen korkeakoulu

Tietotekniikan osasto

Jarmo Ilonen

Predicting diffusion of innovations with self-organisation and machine learning

Diplomityö

2003

75 sivua, 24 kuvaa, 6 taulukkoa ja 2 liitettä.

Tarkastajat: Professori Heikki Kälviäinen, Professori Seppo Pitkänen

Hakusanat: innovaatioiden diffuusio, neurolaskenta, itseorginisoituva kartta, SOM, differential evolution, ennustaminen

Diplomityössä tutkittiin innovaatioiden diffuusion ennustamista. Ennustuksessa keskityttiin yhteen tapaukseen: tuote on ollut saatavilla muutamissa maissa, joissa tapahtuneen diffuusion perusteella ennustetaan diffuusiota muissa maissa. Ennustusmenetelmä perustuu itseorganisoituvaan karttaan (Self-Organizing Map, SOM), jonka opetusaineistona käytettiin maiden taloudellisia ja sosiaalisia parametreja. Kartta optimoitiin eri tuotteille käyttäen kahta eri menetelmää: (a) opetusaineistoon lisättiin tietoa tuotteen diffuusiosta, ja (b) maiden parametreille laskettiin painoarvot eri tuotteiden diffuusion kannalta. Jälkimmäisen, erittäin epä-lineaarisen, optimointiongelman ratkaisemiseksi kehitettiin differentiaalista evoluutiota (DE) käyttävä menetelmä.

Tulokset olivat kohtuullisen hyviä. Ennustusmenetelmä vaikuttaa olevan vahvalla teoreettisella pohjalla. Maakohtaiset tulokset olivat hyviä. Sen sijaan tuotekohtaiset optimoinnit eivät tarjonneet yleisesti selviä etuja, mutta joissakin tapauksissa parannus oli huomattava. Maiden parametreille itseorganisoituvan kartan optimointimenetelmällä löydetyt painot olivat mielenkiintoisia, ja suurin osa niistä oli helposti selitettävissä tuotteiden ominaisuuksilla.

Preface

This master's thesis was done in Lappeenranta University of Technology in department of Information Technology during years 2002 and 2003. The thesis was part of a larger project called "The Global Diffusion of Innovations in Telecommunication Business: Forecasting and the Management of the Diffusion Process" which has been done in Telecom Business Research Center (TBRC) also in Lappeenranta University of Technology.

I'd like to thank my supervisors professor Heikki Kälviäinen and professor Seppo Pitkänen for support. Kaisu Puumalainen and Sanna Sundqvist also receive my sincere gratitude for information and help with economic sciences, and Joni Kämäräinen for ideas and expertise with neural methods.

Family, friends, colleagues and various other earthlings have also been of support, and I express my thankfulness to them.

"If it weren't for the last minute, nothing would get done."
– anonymous

Contents

ABBREVIATIONS

ATO	Automatic Train Operation
CIS	Commonwealth of Independent States
CR	Crossover
DE	Differential Evolution
DOI	Diffusion of Innovations
EU	European Union
GDP	Gross Domestic Product
GNP	Gross National Product
HDI	Human Development Index
ISDN	Integrated Services Digital Network
ITU	International Telecommunications Union
ITU-T	ITU Telecommunication Standardization Sector
MAE	Mean Average Error
MLP	Multiple Layer Perceptron
MSE	Mean Square Error
NUI	Non-Uniform Influence
OECD	Organisation for Economic Co-Operation and Development
OLS	Ordinary Least Squares
SOM	Self-Organizing Map
TAI	Technology Advancement Index

1 Introduction

1.1 Background

There is a research project “The Global Diffusion of Innovations in Telecommunication Business: Forecasting and the Management of the Diffusion Process” in Telecom Business Research Center (TBRC) in Lappeenranta University of Technology. Its goal is to develop an easily applied and exact model for forecasting of diffusion of innovations (DOI) mainly in the field of telecommunications. The model would help companies in the business to estimate success of products already in the development phase. The objective of the research is to produce models which could help companies in telecommuncations business to intensify their R&D operations. Adding theoretical knowledge to the field is also sought after.

The diffusion of an innovation is measured by counting number of purchased products, usually in periods of one year. Diffusion follows often an S-shaped pattern, like many phenomena in the nature. The S-shaped pattern starts slowly and later gathers more speed of growth. After a certain point – called the inflection point – speed of growth starts to decrease and saturation will soon be reached. There are several models for diffusion of innovations. Especially within marketing literature, the most widely applied model is the the Bass model [1], which is also used as a diffusion model in this master’s thesis.

Knowing the diffusion of an innovation beforehand would help companies enormously. This is, of course, quite impossible if very accurate predictions are sought after. The problem of prediction is of the same class as predicting stock exchange rates; even if the predictions are correct most of the time, external events – like natural catastrophes and terrorist attacks – can make predictions go awry. However, the diffusion can be predicted to a certain extent, and predicting even the trend of the diffusion would be helpful. Knowing the correct timing of entering the market and the overall success of the product would be of help when developing the product and planning its deployment.

This master’s thesis concentrates on prediction using machine learning methods. The most important method is Self-Organizing Map (SOM) [2] which is also known as the

Kohonen net or map after its developer Professor Teuvo Kohonen.

1.2 Objectives and limitations

The objective of this master's thesis is to develop a method for predicting DOI. The developed method is not intended to be a general prediction method, but its main objective is to predict DOI in a specific case.

The prediction is wanted in the case where a product has been available in some countries, and its diffusion is wanted to be predicted in a country where it has not yet been available. The prediction is based on the diffusion in the countries where a product has already been available. Another possible case would be to predict the diffusion for a completely new product based on the diffusion of other similar products, but because of lack of available data for a large enough number of products this case could not be tested.

To maximise benefits, the forementioned method should be used together with other prediction methods. It would be unwise to use only a predicted diffusion when there is actually some real diffusion data available for that country – as there will be soon after entering a market – or some other information which cannot be included as a numerical value in the model. Combining information from different sources to make a better prediction is a separate and non-trivial problem.

1.3 Structure of the thesis

The master's thesis consists of three parts.

Section 2 is an introduction to diffusion of innovations. This includes diffusion models, such as the Bass model, and previous research on predicting diffusion using simple parameter estimation with diffusion models and prediction using neural methods.

Section 3 deals with machine learning methods, their theory and applications. Presented machine learning methods are Self-Organizing Map (SOM) and Differential Evolution (DE).

Section 4 combines previous sections to define a method for predicting diffusion of innovations using machine learning methods. The results achieved with the method are presented in Section 5. Conclusions derived from the results and possible future extensions are discussed in Section 6.

2 Diffusion of innovations

Diffusion of innovation (DOI) measures how fast a new innovation, e.g., a new kind of cell-phone or a colour TV, is taken into use in a population. DOI is connected closely to a life-cycle of a product. It is hypothesised that the product life-cycle goes through stages of launch, growth, maturity and decline. The growth often follows S-shaped pattern (Figure 1). The pattern is explained by behavioural and communication theory. An innovation is first adopted by a few innovators, who then influence others by word of mouth. [3]

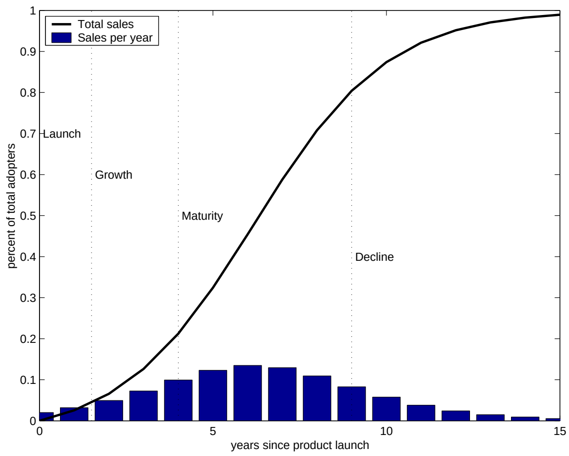

Figure 1: S-curve of sales which is divided approximately to four stages of product life-cycle.

In this section some models for diffusion of innovations, mainly the Bass model [1], and to which kind of tasks they can be used are presented. Then the idea of global or cross-national diffusion is introduced. Finally methods for predicting diffusion using diffusion models and neural methods are discussed in the end.

2.1 Diffusion models

2.1.1 Introduction

One of the first experiments in modelling diffusion was done by investigating the use of hybrid corn in the United States [4]. The hybrid corn was a method of breeding superior corn for specific localities. Not also farmers had to adopt the hybrid corn, but the corn had to be separately bred for each area. Thus, there is not only a problem of acceptance, but there is also a problem of availability. In the research several "economic" variables (such as the availability of corn seeds, the profitability of the changeover and the superiority of hybrids compared to regular corn) were used quite successfully to explain the difference in the diffusion in different localities.

For the purposes of diffusion models consumers have been divided to five classes: innovators, early adopters, early majority, late majority and laggards. These classes are based on the timing of adoption of the product, innovators are the first and laggards are the last. Some models assume that the percentages of these classes are always the same, some other models assume that the percentages are innovation dependent. [3]

Later on numerous diffusion models have been developed. One of the best known models is the Bass model [1], which was published in 1969 and is still widely used. The model works by using three parameters: innovation coefficient, imitation coefficient and the final saturation level of the market. These three parameters define the S-shaped pattern of the diffusion of an innovation. The model will be described in detail later.

Traditionally diffusion models have been used for sales forecasting. However, sales forecasting is only one of the possible uses. Most useful other applications are using them for descriptive and normative purposes. A diffusion model is an analytical approach for defining the diffusion of a product, and thus, diffusion models can be used to test various hypotheses. Diffusion models can be used to make strategic decisions, before a product launch and later during the life-cycle of a product. Diffusion models have been used for the prelaunch forecasting of demand of the products (for example, demand of high-definition televisions has been forecasted). Also the effect of sampling, giving free samples of the product beforehand, has been studied with the help of the diffusion models. In the postlaunch, it is used for determining the correct timing of successive product generations, estimating pirated sales and their effect to the diffusion and determining the market value of a business. [3]

There are following examples of descriptive uses (testing of hypotheses) [5]:

- Product life-cycles are shortening because of technological advancements. 25 products were tested and the hypothesis was deemed to be correct.
- Potential adopters' perceptions of the product affect the diffusion pattern. 14 different investment alternatives were tested using the perceived cost and perceived likelihood of negative returns of the investments. Hypotheses were confirmed.
- The diffusion of technological products in different countries can be explained by level of cosmopolitanism, mobility and role of women in a society. The impact of the three attributes were tested with diffusion of 14 different products in European countries. Findings confirm the hypothesis.

Normative uses are concerned with finding the correct pricing for a product to maximise the profits. Two different pricing strategies have been derived from the Bass model to which the pricing has been added as one variable. One strategy is to increase the price after the introduction of the product and later on decrease the price again. An explanation why such model works is that a cheap price at the beginning will have a strong positive effect to early adopters and the price can be later increased when the product is established. Another strategy is called skimming: start with a high price and decrease it later. At the start those who really want the product are forced to pay a high price, and later the price is decreased after the demand from the early adopters has started to fall.

There are following examples of the normative uses (finding a marketing mix strategy to maximise profits) [5]:

- How should a monopolist price a new product which can be copied? The answer was that the price should be initially high and it should be decreased when the copying increases.

- How should a monopolist price a new product or service whose value increases as a function of the number of adopters, for example e-mail? The price should increase over time.
- When should a monopolist introduce a second generation product? Should it be shelved first or should it be introduced as soon as it is available? Optimal timing seems to be early in the life-cycle of the first generation product.
- How will firms price a new product class over time? When the imitation effect is strong, the price will first increase and later decrease towards the planning horizon.

A monopoly is often used as the market setting, since taking several competitors into account makes models more complicated and more prone to errors. A monopoly or cartel is also a quite prevalent market setting in several markets, for example, computer software and entertainment industry (e.g., compact disc audio).

2.1.2 Bass diffusion model

One of the most widely used models for diffusion of innovations (DOI) is the Bass model, which was conceived by F.M. Bass during the 1960s. [1]

A part of the population adopts the innovation independently of other people. These people are referred as innovators p. Innovators are the people who adopt the new innovation first. Innovators are not influenced by other people, and thus, they decide independently when to adopt a product because of advertising, mass-media or other kinds of external influence.

Apart from innovators, there is another class of product adopters, called imitators q. They are influenced by word of mouth and social pressure. The pressure increases the more there are people who have already adopted the product. The imitaror-class combines later four classes mentioned in Section 2.1.1 (early adopters, early majority, late majority and laggards). Bundling all of these four classes together is justified, because all of them are affected by the timing of purchases by other members of the social system, which can be called internal influence. Innovators are a completely separate class because they are affected only by external influence, not by the other members of the social system. [1]

The formula for cumulative diffusion as a function of time is

$$S(t) = m\frac{(p+q)^2}{p}\frac{e^{-(p+q)t}}{(1+\frac{q}{p}e^{-(p+q)t})^2} \tag{1}$$

where p is the innovation coefficient, q is the imitation coefficient and m is the potential number of adopters. The first derivative of the cumulative diffusion equation, called non-cumulative diffusion, which tells the diffusion at a specific time, is

$$s(t) = m\left[\frac{p(p+q)^2 e^{-(p+q)t}}{(p+qe^{-(p+q)t})^2}\right]. \tag{2}$$

Both innovation p and imitation q coefficients are always positive. p is for most products not larger than 0.1 and q is not larger than 1. m is usually normalised by the total population, so it is usually [0..1]. However, m might larger than 1 for some products, since some people may have two cell-phones, for example.

The model has following features [1]:

- During the life-cycle of the product there will be m initial purchases. Since we are dealing with products with long life-cycles, the number of initial purchases is quite close to the total number of purchases. Total purchases also include resales.
- Initial purchases of the product are done both by innovators and imitators. The difference between two classes is not the timing of purchase, but the product adoption influence.
- The role of innovators will be greater at beginning and it will diminish over time (see Figure 2).

Terms innovators and imitators are somewhat inaccurate, since innovators affect the diffu-

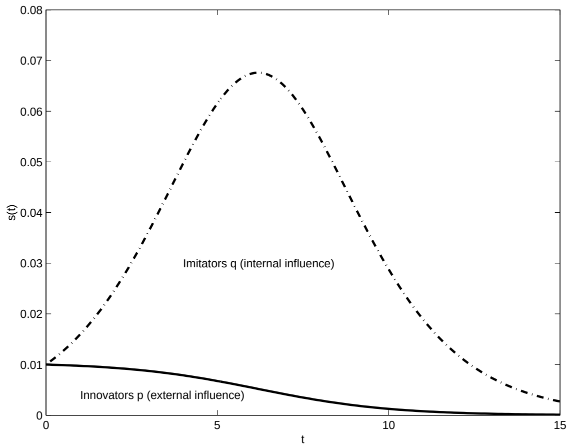

Figure 2: Graph of the non-cumulative Bass model. [5]

sion throughout the product's life-cycle. The word innovator is derived from the verb "to innovate", which means "to bring something new to an environment"; therefore, labeling someone as an innovator when purchasing a product very late in the product's life-cycle seems rather nonsensical. Thus, more accurate terms might be external influence for innovators and internal influence for imitators [6]. However, original terms (innovators and imitators) introduced by Bass will be used in this Master's thesis.

How the parameters affect the diffusion curve? Innovators p affect the diffusion most at the beginning, right after the product's launch. The lower the value of p, the slower the diffusion will start. The speed in the beginning of the diffusion process affects mainly the place of the peak of sales: a slow start will lead to the peak of sales being late, a rapid start will lead to the peak of sales being early (see Figure 3). The height of the peak of sales is more dependent on the imimators q. A low value of q will lead to a low peak of sales, with higher values the peak will be higher and also earlier (see Figure 4). The value of m linearly scales the diffusion graphs (see Figure 5).

2.1.3 Limitations of the Bass model

There are several restrictive assumptions underlying the Bass model. It does not consider several aspects which affect sales in the real world. It is a model which works only for

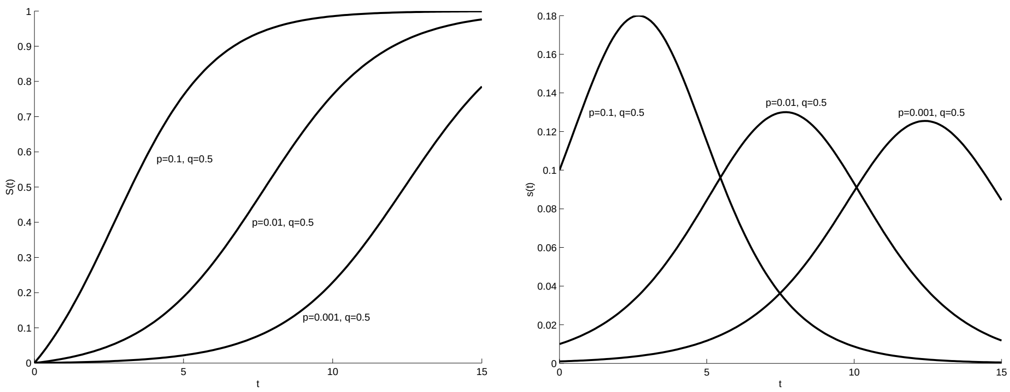

Figure 3: Graph of the Bass model with different values of innovators p. The cumulative diffusion is on the left and the non-cumulative one on the right.

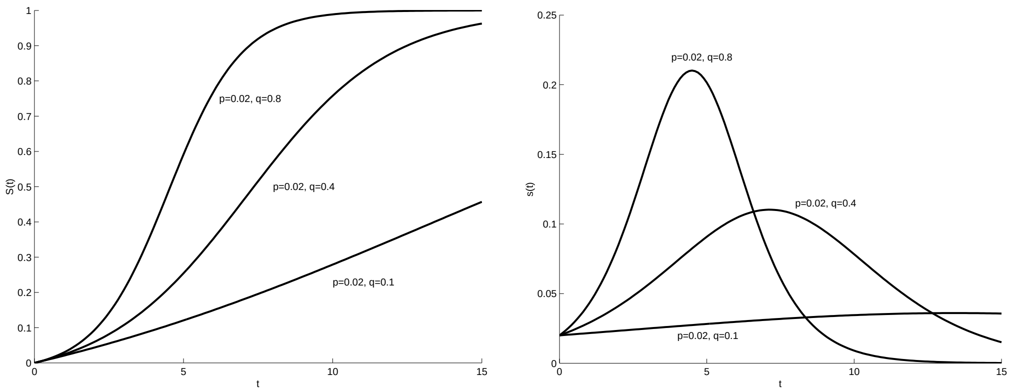

Figure 4: Graph of the Bass model with different values of imitators q.

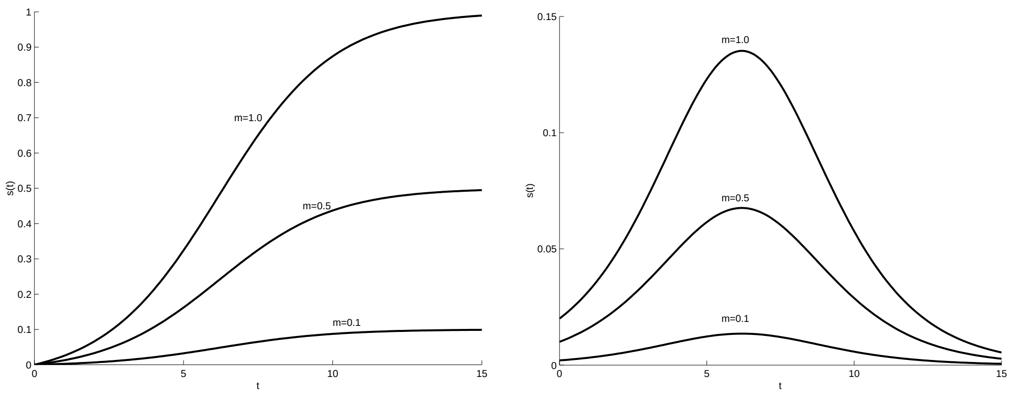

Figure 5: Graph of the Bass model with different values of m.

products which are purchased only once per person, or rather it considers only the first purchase of a product, i.e., it does not work with products which are sold repeatedly. Most assumptions are due to simplifying the model. Simplifications make possible to generate a working model with a plausible number of parameters. However, it is important to be aware that such assumptions exist.

The following limitations and assumptions exist [5]:

- Market potential of a new product stays the same. The model assumes that m will remain constant during the product's life-cycle. Theoretically there is no rationale for this, since market potential could be changing all the time.

- The diffusion of an innovation does not depend on any other innovations: In reality other innovations are always on the marketplace competing or complementing more or less directly with each other. Taking this into account is essential when products are directly tied, e.g., compact discs (CD) and CD-players, or when they directly compete, e.g., VHS and Beta video devices.

- The innovation stays the same: With new technological products, such as cellular phones and video gaming systems, new product generations are introduced frequently. Each new generation should be better in relevant features, and it will take market share from the previous generation of the product.

- The geographic boundaries are not considered. New products are often rolled out at different times in different markets, inside of a nation and internationally. Such strategy capitalises on the word-of-mouth communication between geographical neighbours.

- Marketing does not influence the diffusion of an innovation. Price and advertising of the product, and their changes, are not considered.

- No supply restrictions exist. In reality, sales cannot be greater than the supply available in the marketplace.

- There is only one adoption per one consumer. For most products there are also repeat buyers.

2.1.4 Other diffusion models

The Bass model, as well as many other simple diffusion models, is symmetric. The point of inflection is the point in a diffusion curve where the maximum rate of the diffusion has been reached. The Bass model is symmetric on both sides of that point. This restricts the model so that the maximum rate of diffusion cannot occur after product has gained 50% of the market. This is not a valid assumption since both in practice and in theory the maximum rate of the diffusion should be able to occur at any point of time. Also diffusion patterns cannot be expected to always be symmetric.

One of the flexible diffusion models (where the maximum rate of the diffusion can happen at any time) is the non-uniform-influence (NUI) model [5], [7]. The model is modified from the Bass model by specifying a coefficient of internal influence which varies over time as a function of the penetration level as follows:

$$\frac{dF(t)}{dt} = \left[p + qF(t)^{\delta}\right]\left[1 - F(t)\right], \tag{3}$$

where $F(t) = S(t)/m$ and δ is the non uniform influence factor. The NUI model enables the maximum adoption rate to occur in any point of time and the adoption curve does not have to be symmetrical.

On the other hand models have been extended to frameworks which do not consider only the adoption process, but also the availability of stock and the abandonment process. The abandonment means forsaking the product usually because there is a new and better generation of the product available. There is no simple S-curve anymore in such more complicated frameworks, since the products will be abandoned at some point of time and the cumulative number of adopters does not rise monotonically.

One such framework is introduced by Kang, Han and Yim [8]. The framework uses the adoption process, together with availability and abandonment, and it also takes income levels, products price and consumer's taste (and their development) into account. The model based on the framework was tested with consumer durables (VCRs, televisions) and telecommunication service products (telephone and cellular phone service subscribers) in Korea and Japan. The model gave better results than Bass and NUI models.

2.2 Cross-national diffusion

DOI is usually considered inside one population which is usually one country. Broadening the diffusion model to a larger population, the whole world for example, is quite problematic because of huge economical and cultural discrepancies between different countries. Additionally marketing strategy is often to "roll-out" products intentionally at different times at different places. Bundling diffusion from all countries to one model is quite impractical and besides useless for purposes of making managerial decisions, because of a too large scale. An interest in the multi-market or global diffusion has risen lately because of the removal of economic and political barriers as well as the rapidly advancing globalisation. Finding specific cultural and economical straits which affect the diffusion of a specific product would be helpful. [3]

There are several important questions. How consumers react to a new product in different countries? What cultural and economic attributes affect consumers' reactions? Why some product is adopted fast in one country and slowly in another country? Is it possible to predict how consumers will react to a new product in a specific country, and furthermore, how fast this product will reach a certain market penetration? All these questions should optimally have a correct answer before introduction of a product to a new market. By studying the diffusion of new products and technologies in different countries, some answers can be provided. V. Kumar et al. have reviewed research in this field. They have summarised the following findings [9]:

- Differences in the diffusion of a product in different countries can be explained to a great extent by country specific factors, such as cosmopolitanism, mobility and women in labour force.
- There is a "lead-lag" effect, also called demonstration effect: the later a product is introduced in a country, the faster the adoption will be. The effect can be explained by cross-national word-of-mouth effect.

- When countries are clustered based on the diffusion charasteristics of a product, clusters will likely be different when clustering is done with the diffusion charasteristics of a different product. Country-clusters are not stable and they depend on a product.

Findings are partially based on the same data, and are contradictory on some points. Most of the findings are nevertheless confirmed by Kumar, Ganesh and Echambadi [9]. It is recommended to create a conceptual framework for identifying factors that potentially affect the diffusion.

Finding countries which are early adopters and which are late adopters is the main focus of Dekimpe, Bodnovich and Selvi [10]. The task is based on using country-specific demographic, economic, political factors and social factors. Also exogenous effects are investigated: elapsed time since the introduction of an innovation and the demonstration effect. It was found that non-East-block countries with high GNP per capita, few ethnic groups and many major population centers were usually those to first adopt an innovation. Following hypotheses were supported by tests:

- The higher the relative advantage of the innovation for a country, the sooner the country adopts it.
- Wealthier countries adopt the innovation earlier.
- Isolated economies tend to adopt innovations later.
- A country's adoption timing is negatively related to its society's heterogeneity.
- The higher the proportion of similar countries having adopted the innovation, the higher the probability that a country will also adopt the innovation (demonstration effect).

2.3 Model parameter estimation and prediction

Parameter estimation with a diffusion model serves as the simplest form of prediction. After estimating the parameters of the model with sales data of some product, future sales can be predicted by calculating the next years diffusion using the estimated parameters. For example, with the Bass model at least three data points (usually data for three years of sales) is needed for parameter estimation, and it is suggested that six data points are needed for correct estimation because of the chaotic behaviour of the diffusion in the beginning of the product's life-cycle [3]. Models with more parameters need, of course, even more data for parameter estimation.

Parameter estimation can be done using for example OLS (Ordinary Least Squares) which F.M. Bass used in his original research with the Bass model [1]. However, there are several shortcomings using OLS. First, it may yield parameter estimates which are unstable or have a wrong sign. Second, standard errors of the estimates are not provided, and thus the statistical significance cannot be established. Third, time-interval bias exists because discrete time-series data is used to estimate a solution for a continuous diffusion model.

Some methods have been suggested to overcome these limitations. A maximum likelihood estimation procedure estimates parameters directly from the differential equation specification of the Bass model. However, the method underestimates standard errors of the parameters, so wrong inferences about statistical significance can be made. Another suggested method, nonlinear regression also uses the solution of the differential equation specification of the Bass model. Of these methods OLS performs overall worst, and nonlinear regression method is slightly better than the maximum likelihood one. [3]

The need of data from such long period of time is a huge problem for this kind of prediction. Nowadays product life-cycles are quite short, and having reliable prediction six years after introdtion of a product to a market is quite irrelevant. The parameter estimation is mainly useful for purposes of model testing and comparisons between products.

The parameter estimation can be combined with some method for updating results every time a new data point becomes available. However, even with a method for updating the parameters when new data is available three data-points are needed to get an initial estimate which is anyway vulnerable to often observed chaotic effects at the start of the diffusion process. To get predictions before this, some other methods are needed to get the initial prediction. [5]

There is no self-evident way to get an initial estimation for the diffusion of a new product. The easiest way is guessing, augmented with the knowledge of the diffusion of previous similar products. Management judgements can be used to either directly guess the diffusion curve and thus the parameters of the Bass model, or the market size and peak time of sales can be estimated to get the parameters indirectly. There are also rules of thumb for estimating probable parameters for different kinds of products.

2.4 Prediction using neural computing methods

Neural networks [11], also called artificial neural networks, have been used quite heavily in business applications research. A part of the applications are in the field of prediction. Neural networks have been used, for example, to predict bankruptcies, future prices, workplace behaviour and energy use [12] [13]. Reviewing research and comparing neural methods with statistical methods have been done by Hill et al. [14]. Prediction using neural networks seemed to work as well or better than statistical methods for time-series prediction.

An analysis of using neural networks for prediction including information on choosing a suitable neural network type, architecture, training algorithm, data normalisation and so on is presented by Zhang, Patuwo and Hu [15], [16]. Research seems to mostly concentrate on using traditional feed-forward multi-layer neural networks. For example Self-Organizing Map is very rarely used.

Using neural methods for time-series prediction usually bypasses the diffusion models. The most used method is to use past observations as inputs of a neural network and the predicted value for the next year as the output. Input patterns are composed of a moving window of a fixed length along the time series data. A neural network performs the following mapping

$$y_t = f\left(y_{t-1}, y_{t-2}, ..., y_{t-p}\right) \tag{4}$$

where y_t is the observation at time t, and the dimension of the input vector or number of past observations is p, and f is a non-linear function which is determined by the neural network. An example of a Multi Layer Perceptron (MLP) is shown in Figure 6. In this case input values are $y_{t-1}, y_{t-2}, ..., y_{t-p}$ and output value is y_t. f is defined by the weights between the perceptrons and transfer functions inside the perceptrons. Each perceptron first computes a linear combination of its inputs, then applies a threshold or a continuous threshold function (e.g., a sigmoid function) to the result which is then used as the output of the perceptron. There can be more than one hidden layer in the neural network. With more hidden layers more complex functions can be simulated by the neural network.

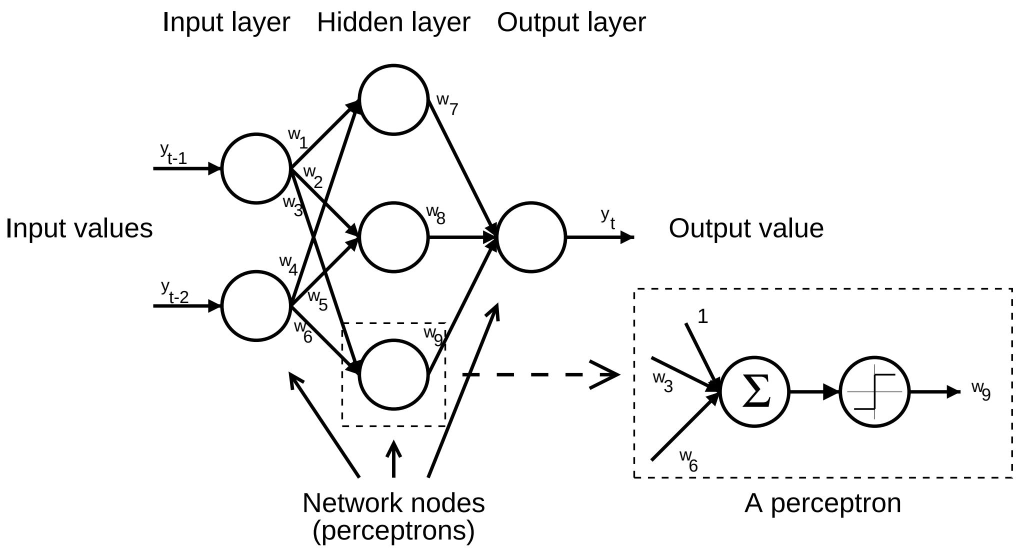

Figure 6: A Multi-Layer Perceptron (MLP) neural network.

The use of the neural network consists of training, also called teaching, the network and using it, in this case, to predict a new diffusion value based on the input values. Training is done by having a set of data where both the input and the expected output is known. The output values for all the input vectors in the set of data are computed using the neural network and the error between the output values of the network and the expected values is measured. After measuring the error the weights between perceptrons of the network are changed so that error gets lower. These training steps are continued until the error is acceptably low. There are several different training algorithms, and one example is presented in Algorithm 1.

Algorithm 1 MLP: A simplified gradient descent backpropagation algorithm for training MLP feedforward networks.

1. Create a feed-forward network with input, hidden and output layers.
2. Initialise all network weights to small random numbers.
3. Until termination condition is met, do
   - For each training example (a pair of input and output vectors), do
     3.1 Propagate the error through the network: compute output of every unit using the input vector.
     3.2 For each node in the output layer, compute its error term.
     3.3 For each node in the hidden layer, compute its error term using error terms of the output layer and current weights between nodes.
     3.4 Update each network weight based on the error and the input vector: the error will become smaller for the input vector.

The correct configuration and size of the network is important for good performance. There should not be too few or too many nodes in the network. Too few nodes will lead to under learning, and the error remains high because the network is too simple to learn the function f. Too many nodes will lead to over learning, because the network learns the used training data too perfectly but does badly when used with any other data. The number of hidden layers in the network affects how complicated the non-linear function f described by the neural network can be. There is no reliably working method to determine the network configuration, and thus, it is usually done by trial-and-error. [16]

3 Machine learning

Machine learning is concerned with a question of creating computer systems that improve automatically with experience. Several algorithms have been invented to make computers learn. However, these algorithms have to be hand-crafted to certain types of learning problems. Machine learning is yet not even close to the learning ability of humans. Still, some theoretical understanding of machine learning is starting to emerge. The most widely used machine learning methods include decision tree learning, artificial neural networks, Bayesian learning and genetic algorithms. [17]

One useful application of machine learning is Movielens [18]. It is a system that makes predictions of the user's ratings to new movies based on movies rated earlier by the user him/herself and all the other users. It can also recommend movies by listing movies that the user is predicted to like the most. Movielens employs item based collaborative filtering to achieve these goals [19]. Various other recommendation systems used in E-commerce are reviewed by Schafer, Konstan and Riedl [20]. The importance of such systems for E-commerce is high: consumers make more purchases when they can easily find interesting products. Creating such systems is challenging. There is often a huge amount of data available in various formats and recommendations should update regularly when consumer makes ratings or purchases new products.

In this section two machine learning methods are presented. The first one is Self-Organizing Map (SOM) [2], [21] which is a neural method for ordering multi-dimensional data spatially. The second one is differential evolution (DE) [22] which is a genetic algorithm suitable for optimising complex and non-linear functions. These methods will be later used to solve the problem of predicting diffusion of innovations.

3.1 Self-Organizing Map

3.1.1 Introduction

SOM (Self-Organizing Map) [2], [21] is also known as Kohonen map or Kohonen net after its developer Prof. Teuvo Kohonen. It is an unsupervised and competitive neural computing method, which can be used for clustering and classification of data.

Neural networks can be divided to three classes based on how they work. (A) Feed-forward networks change the input signal to the output signal. How the transform is made is determined in the training phase by changing the network supervisedly. (B) In the feedback networks the input signal determines the initial state of the network. From the initial state the network progresses computationally to some stable terminal state which depends on the input signal and should be as close as possible to the input signal using some metric. (C) Third class is unsupervised or self-organising neural networks in which the network elements adapt to sense different input signals by using competitive learning [2]. On the other hand, this class of neural networks is also known as competitive neural networks because the training is done by competing elements against each other and the winning elements are then adapted. [23]

SOM belongs to the third class. It is a planar neural network where cells attune to detect different kinds of input signals or input signal classes. In the basic SOM only one cell is active at time. This active cell is determined by the input signal. Similar input signals tend to activate cells which are close to each other [21]. SOM creates a transformation or a mapping from the higher dimensional space to one or two dimensional map. This is shown in Figure 7. Points which are close to each other in the higher dimensional space are also close in the one or two dimensional map. Data is called to be clustered. [11]

3.1.2 History and biological background

The history of SOM begins from research of brains and the creation of brain-maps in the 18th century (Figure 8). It was noticed that different parts of brains are specialised

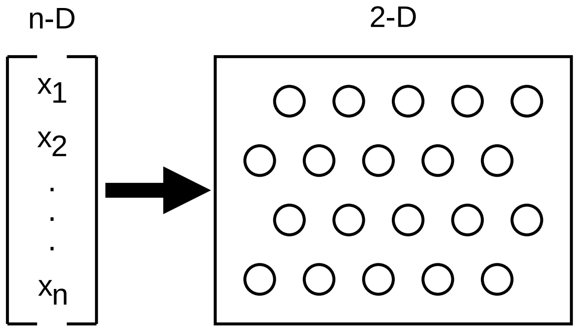

Figure 7: SOM generates a transformation or mapping from n-dimensional data to 2-dimensional map.

in different tasks. Topological maps of different regions of brains were generated by inferring from functional deficiencies caused by injuries and diseases. Nowadays, brain-photography is advanced enough to get brain-maps with accuracy of millimetres. Brain-maps are not, however, completely similar among people.

In brains there is among other things a “map” of skin, where sense of touch of different parts of skin maps to specific places in brains. There is also a similar map for controlling the parts of the body. In these maps, regions for sense of touch and controlling of fingers are next to each other. The structure of brain-maps can be explained mainly by genetic properties, but the brain-map can adapt if some part of the brain is damaged because of a injury.

It has also been speculated that learned things are ordered in the brains by their properties: things belonging to the same category are close to each other. There is some evidence that information is ordered spatially in brains. Also SOM attempts for the same goal. [2]

The learning law used in SOM was presented in neural network research in 1962, or maybe even earlier [23]. During 60’s, research of unsupervised neural networks advanced and in 70’s first competitive neural networks were proposed which could spatially order the input vectors. The term “Self-Organizing Map” was first used by von der Malsburg [24]. These first experiments at self-organisation were limited, since they could only create order in some parts of the map or only in one dimension. Prof. Kohonen started

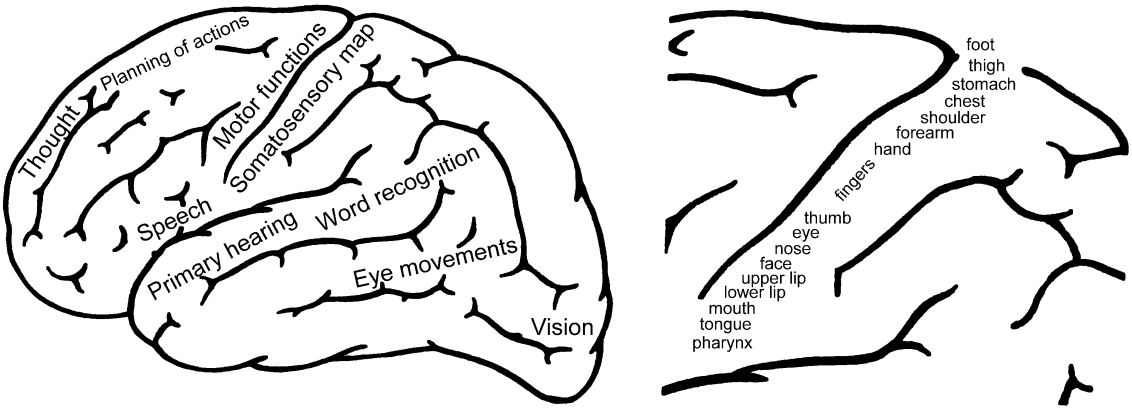

Figure 8: Brainmaps: On the left is a map of brain areas, on the right is the somatotopic map.

research leading to his self-organizing map in 1981, and the results were published in the next year. [2]

There has been some questions whether the Self-Organizing Map should be called a neural network at all. It does not imitate methods of biological neural networks as directly as for example perceptrons of traditional neural networks. However, it usually finds a very natural order for input data which resembles the ordering of information happening in brains. Additionally, the ordering of information in brains may actually use very similar methods as used in SOM. In general, SOM is usually included in the group of neural networks. [21]

3.1.3 Theory and algorithm

The algorithm is presented in Algorithm 2.

Algorithm 2 The kernel of the SOM algorithm:

1. Compute distance from the input vector to each cell in the map.
2. Select the best matching unit (BMU) in the map, i.e., find the cell with the minimum distance.

3. Adapt the weights of the cell(s) in the neighbourhood of BMU.

The steps of Algorithm 2 are repeated, possibly several times, for all the input vectors.

SOM is a competitive neural network. In the competition phase the best matching cell from the network is searched. The input vector, also called feature vector, is defined as $\bar{x} = [x_1, x_2, \ldots, x_n]$. The input vector is assumed to be connected to all cells in the network. Each cell of the network has weights, a weight vector for cell i is $\bar{m}_i = [m_{i1}, m_{i2}, \ldots, m_{in}]$ (see Figure 9).

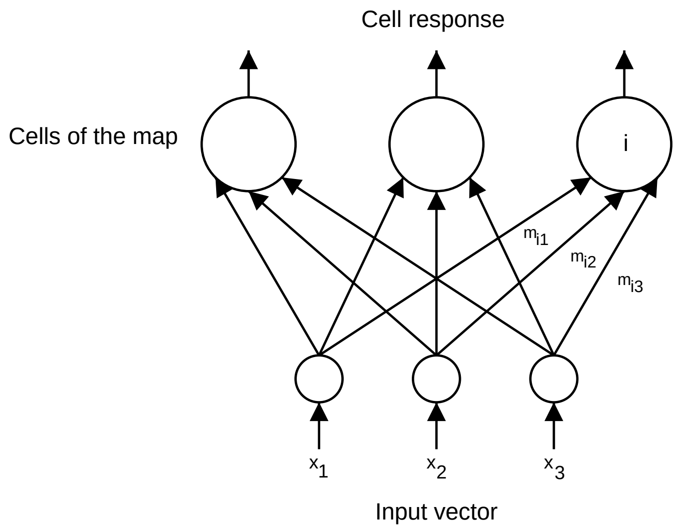

Figure 9: Kohonen layer.

Distance between $\bar{x}$ and $\bar{m}_i$ can be analytically measured using many different distance measures. The simplest method for measuring a distance would be the inner product, $\bar{x}\bar{m}_i{}^T$. However, since SOM is usually used with patterns relating to metric vector spaces, a better and more convenient measure based on the Euclidean distance can be used. The winning cell (best matching unit, BMU), $\bar{m}_c$, is defined by the smallest distance as follows:

$$|\bar{x} - \bar{m}_c| = \min_i \{|\bar{x} - \bar{m}_i|\} \tag{5}$$

Cells can be arranged in hexagonal, rectangular, etc. pattern. All arrangements basically work just the same, but a hexagonal pattern is usually favoured because of more natural looking visualisation compared to for example a rectangular pattern.

The next step is to adapt cells in the neighbourhood of the winning cell m_c. Several cells have to be adapted in the neighbourhood of the winning cell, not only the winning cell. During the process of adaptation the network is shaped to conform to the data. Also, weight vectors tend to attain values that are ordered along the axes of the network. [21]

A neighbourhood in the map can be defined can be seen Figure 10. Three different sized neighbourhoods have been defined around the winning cell. Usually the size of the neighbourhood is gradually decreased during the adaptation process. In the beginning of the adaptation, a large neighbourhood size generates a rough global ordering in the map and later a smaller neighbourhood increases spatial resolution without destroying the global ordering which has been created earlier.

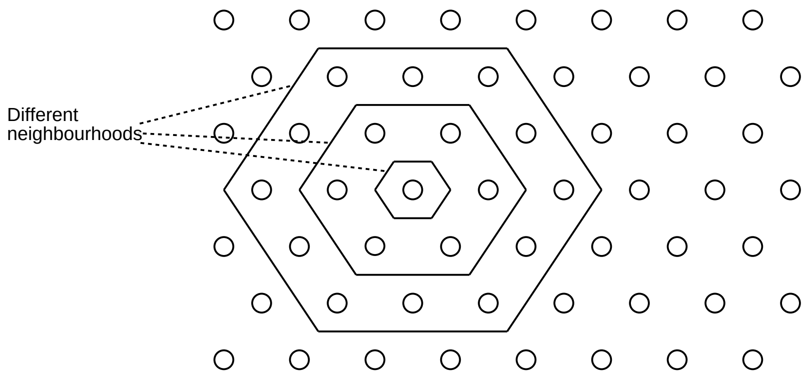

Figure 10: Different neighbourhoods in a hexagonal map.

The adaptation is done in discrete-time notation using

$$\bar{m}_i(t+1) = \bar{m}_i(t) + \alpha(t)\left[\bar{x}(t) - \bar{m}_i(t)\right] \tag{6}$$

where t is time, and $\alpha(t)$ is the adaptation gain $0 < \alpha(t) < 1$. The adaptation gain is used to change the speed of the adaptation, and it should decrease over time as does the size of the neighbourhood. Only cells inside the specified neighbourhood are adapted, other cells are not touched.

A tightly defined neighbourhood can be substituted by using a function which depends on the distance between cell i and the winning cell c. In the previous function $\alpha(t)$ is replaced by $h_{ci}(t)$. The cells which are closer to the winning cell are adapted faster than cells which are farther away. The following Gaussian function is often used

$$h_{ci}(t) = \alpha(t)e^{\frac{-|r_c - r_i|^2}{2\sigma(t)^2}} \tag{7}$$

where r_c and r_i are the coordinates of the winning and adapted cells, and $\sigma(t)$ is a monotonically decreasing function which affects how wide the Gaussian function is. Using this kind of function is computationally more intensive since all cells of the map are adapted, not only those inside some defined neighbourhood. On the other hand, the methods can be combined: changes are made to cells inside the neighbourhood and the amount of adaptation is defined by using for example the Gaussian function. [2]

A more detailed SOM algorithm is presented in Algorithm 3.

Algorithm 3 The SOM algorithm:

1. Compute from the input vector $\bar{x} = [x_1, x_2, \ldots, x_n]$ the distance $|\bar{x} - \bar{m_i}|$ to each cell in the map.
2. Find the cell c with the minimum distance (BMU), $|\bar{x} - \bar{m_c}| = \min_i\{|\bar{x} - \bar{m_i}|\}$.
3. Adapt the weights of the cells in the neighbourhood of cell c, $\bar{m_i}(t+1) = \bar{m_i}(t) + \alpha(t)\,[\bar{x}(t) - \bar{m_i}(t)]$.

3.1.4 Practical considerations for the use of SOM

The weights of the network cells must be somehow initialised. Constant numbers cannot be used because finding winning cells would be impossible because the distance would be same to all cells. The most simple method is to use random numbers as initial weights. This does not require any knowledge of the used training data. When there is knowledge of the data available, random initialisation can be improved, for example, by finding minimum and maximum values and limiting random initialisation between those values. A more complex variation of initialisation is to calculate a correlation matrix from data and to make linear initialisation by using the largest eigenvectors. After this kind of initialisation the weights of the cells of the network form a plane in n-dimensional space. The plane is directed so that data is scattered as evenly as possible along it, providing the best starting-point for training.

Visualisation is usually needed, or sometimes it is the main goal of using SOM. There are two different for visualisation methods: adding labels to the map and colouring the map. They can be compared to a normal terrain map where labels are names of places and the colouring is based on either to the elevation or the type of terrain. Adding labels is simple: When there are some feature vectors with known labels, just find their closest cells on the map and add the labels there. Sometimes instead of labels colouring can be used to indicate which kinds of vectors go to which areas in the map. For example, if there are only two separate classes of feature vectors it is possible to colour the cells with colours of the classes instead of adding labels. When feature vectors from several classes are closest to a cell, majority voting or shades between the class colours can be used.

The norm of the weight vector, $|m_i|$ for cell i, can be directly used for colouring the map. This usually does not work too well because weight vectors have many dimensions. A better method for visualisation is to colour based on differences between the vectors in the neighbouring cells, e.g, Euclidean distance between two weight vectors. Usually a dark colour marks a large difference and a light colour marks a small difference. [2]

The structure of the map is usually hexagonal, i.e., the cells are ordered in hexagonal order. Other orderings are possible, for example a rectangular structure. Compared to the rectangular structure the hexagonal structure often looks more natural. SOM creates an elastic net which should have one fixed direction to be stable. The fixed direction is usually the longer edge of the map. Thus, the map should have a longer edge; if the map is rectangular or even spherical, there is no fixed direction and training might not work well because the net tends to rotate. The edges of the map can be joined and the map can be thought of as a cylinder (when only vertical or horizontal edges are joined), or torus (when both vertical and horizontal edges are joined). [25]

Training the map takes several thousands or even hundreds of thousands steps. If there is not enough data, same data units have to be trained very many times. Training can be done in random order, but it has been noticed that there is no problem even if consequential trainings are done in the same order. There might also be a problem if there is not enough data of some special type. For example, when making a map of states of some system there might not be enough data of error states. Having them on the map would be essential, but there might not be enough data of error states to actually get them presented on the map. This problem can be solved by training the special data several times or using larger $\alpha(t)$ with them.

Scaling of the data can be problematic. Since SOM uses simple Euclidean distance metric by default, discrepancies between scales of different parameters of feature vectors will cause a problem. If one of the feature vector parameters is considerably larger than others, it will dominate the generation of the map. For example, when creating a map of countries of the world using such attributes as population, gross national product per person, urbanisation percent, etc. the population will have a much larger value than all other attributes. This problem can be solved by using normalisation. Normalisation methods range from simple scaling of all attributes to $[0 \ldots 1]$ to, for example, standard deviation based normalisation.

Sometimes certain types of feature vectors are required to be in a specific place in the map. For example, when using SOM for visualising the state of a system, the "normal" state might be wanted to reside in the centre of the map. The solution is to place those feature vectors directly to the map (as the weights of the centering cells), and possibly also lower $\alpha(t)$ at that place to prevent them from moving to another place in the map. [2]

SOM can also cope with incomplete data. If some of values are missing in the input vector, it is still possible to find the closest cell in the map by skipping the missing values in calculations. If there is not a huge amount of values missing, there is no problem training the map. [2]

Some of the important things to be aware of and to decide what is needed when solving a certain problem using SOM are listed as follows:

- Initialisation: random or linear.
- Visualisation: colouring and adding labels.
- Structure of the map: usually 2-dimensional and hexagonal.
- Training: maybe using some of the data several times or forcing certain data to a part of the map might be needed.
- Scaling of the data: values have to be approximately in the same range.
- Missing data is no problem.

3.1.5 Applications

There are a huge number of SOM applications, even in Finland. Many SOM applications from Finland can be found in [26]. One of the most important advantages of SOM is that it clusters data by unsupervised training. Thus, no data specific models or a priori rules are needed. The more data, the better, so SOM also copes with huge data sets. It is also tolerant to distortions and noise.

SOM can be used for process analysis. Traditionally this has been done with various analog and digital displays which show values of different sensors of the system. It might be difficult to comprehend the state of the system if there is a large number of separate displays showing values of sensors. SOM can help to visualise the state when the values of sensors are gathered in several different situations and are trained in the SOM. This generates an easily comprehendable view of the state of the system. Abnormalities are easy to notice and the map can also be used for automatic alarms and possibly for the control of the system. [27]

Computer vision is one of the most important areas of neural networks. Important application areas are in parts of industrial automation, visual quality control, robotics, document processing, medical imaging, etc. Traditionally this has been done in several processing steps. Steps might be preprocessing of the image, feature extraction and classification of features. Often a priori knowledge about the objects is used with some kind of artificial intelligence based approach for interpretation. This approach has been criticised because of its rigidity. It does not work well in cases where flexibility is needed, for example recognising characters in handwritten text. Traditional neural methods can be used to overcome some of the inflexibility problems, but they need large amounts of preclassified data for training. SOM on the other hand is an unsupervised method, so only a small part of the training data has to be preclassified. [27]

One of the most known specific SOM applications is WebSOM [28]. It uses several millions of Usenet articles and orders them so that similar articles are close to each other in the map. When one interesting article has been found, other similar articles are easy to find. On the other hand, the map can be viewed at a wider scale where names of the Usenet newsgroups can be seen. This kind of use might not be very helpful for finding information, but seeing which kinds of articles are found where two completely separate topics meet in the map might be interesting. The main topic of the research has been developing methods for ordering large text collections and in the same time optimising SOM to cope with immense amount of information with a huge map. [29]

Another similar SOM application is PicSOM [30]. It is an image retrieval system which uses Tree Structured SOM (TS-SOM) to retrieve images from a database. PicSOM is designed to be an open and adaptable system. It can be used with a general image database or with images of some specific field. Also, different features can be used with different databases with textual keyword information. The use of the system consists of choosing the most relevant image from a presented set of images. After that similar images are searched from the database and a new set is presented to the user. After continuing the process for a few steps, a suitable image will be hopefully found. [31]

3.2 Differential Evolution

3.2.1 Introduction

Differential evolution (DE) [22] is a heuristic approach for minimising possibly non-linear and non-differentiable continuous functions. The function to be minimised is called the objective function. In the following sections its background, theory and few applications are presented.

Direct search methods are usually applied when the objective function is non-linear and non-differentiable. All direct search methods use some strategy to generate variations of parameter vectors. When a variation is generated, a decision must be made whether it should be kept. Basic methods use a greedy criterion, where a new variation is accepted only if it is better (reduces the value of the objective function). Using the greedy decision optimisation tends to converge quickly, but the problem is that the point found may be a local minimum, and not necessarily the global minimum. Parallel search techniques, like differential evolution, succeed better in this sense because they have several different parameter vectors simultaneously.

The design goals of the differential evolution are as follows [22]:

1. Method should find the true global minimum.
2. Convergence should be fast.
3. Minimise the number of control parameters for the ease of use.

3.2.2 Theory and algorithm

Differential Evolution is a parallel direct search method. It utilises NP D-dimensional vectors as a population for each iteration. Please note that some two letter parameter names are used like in the original paper. Iterations can be considered as generations since the algorithm uses a genetic approach. The population should be initialised randomly, and all the parameter space should be covered uniformly by vectors of the initial population.

The algorithm works basically by adding the weighted difference between two randomly selected population vectors to a third vector. If the value of the objective function is lower with this new vector than with a predetermined vector chosen for comparison, the vector to which comparison was made is replaced by the new vector in the next generation. If the new vector does not give a lower value of the objective function, the original vector will be preserved to the next generation. Usually all NP vectors of the population are chosen for the comparison, one by one, in every generation. Thus, all the vectors can change between two consecutive generations. These basic principles can be altered when making practical implementations of the DE algorithm. Existing vectors can be changed by adding more than one weighted difference vector to it. In the most cases it is also beneficial to mix the parameters of the old vector with those of the changed vector. This is called cross-over.

Various different schemes have been cultivated from the basic principle. The most common scheme is named DE/rand/1 which will be presented here. The scheme follows closely the original principle of DE.

Let us assume that function $T(\overline{x})$ is wanted to be minimised. For each vector $\overline{x}_{i,G}$, where i is $[0, 1, \ldots, NP - 1]$ and G is number of the generation, a perturbed vector $\overline{v}_{i,G+1}$ is generated with formula

$$\overline{v}_{i,G+1} = \overline{x}_{r_1,G} + F\left(\overline{x}_{r_2,G} - \overline{x}_{r_3,G}\right) \tag{8}$$

where r_1, r_2, r_3 are mutually different and randomly selected from $[0, 1, \ldots, NP - 1]$, and F is a factor for controlling differential variation, usually $0 < F < 2$.

Crossover CR is used to increase the potential diversity of the population. Crossover can be formulated by creating vector

$$\overline{u}_{i,G+1} = \left(u_{0i,G+1}, u_{1i,G+1}, \ldots, u_{(D-1)i,G+1}\right) \tag{9}$$

with

$$u_{ji,G+1} = \begin{cases} v_{ji,G+1} & t_j < CR \\ x_{ji,G} & \text{otherwise} \end{cases}, j = 0 \dots (D-1), \tag{10}$$

where $\overline{t} = (t_0, t_1, \dots, t_{D-1})$ is a D-dimensional vector with random values $[0, 1]$. Thus, crossover CR controls how large chance there is that a value from the transformed vector $\overline{v}_{i,G+1}$ is chosen instead of a value from the original vector $\overline{x}_{i,G}$ for the vector $\overline{u}_{i,G+1}$. The value of the objective function is then calculated with the vector $\overline{u}_{i,G+1}$, and if the value is lower than with the original vector from the previous generation $\overline{x}_{i,G}$, then $\overline{x}_{i,G+1} = \overline{u}_{i,G+1}$, otherwise it will remain unchanged, $\overline{x}_{i,G+1} = \overline{x}_{i,G}$. In other words, if the new vector with crossover is better than the original vector, the original vector will be substituted in the next generation by the new vector, otherwise the original vector will live to the next generation. An example of using differential evolution is given in Figure 11. [32]

The method is presented in Algorithm 4.

Algorithm 4 Minimising the value of the function $T(\overline{x})$ with DE

1. Initialise the population, set values of vector $\overline{x}_{i,0}$ for all $i = (0, 1, \dots, NP-1)$ with random values.
2. Generate population for the next generation $G+1$
   - 2.1 For all population members $i = (0, 1, \dots, NP-1)$
     - 2.1.1 Select three random, mutually different values r_1,r_2,r_3 from $(0, 1, \dots, NP-1)$.
     - 2.1.2 Generate a perturbed vector $\overline{v}_{i,G+1} = \overline{x}_{r_1,G} + F\,(\overline{x}_{r_2,G} - \overline{x}_{r_3,G})$.
     - 2.1.3 Generate a trial vector $\overline{u}_{i,G+1}$ by crossing over vectors $\overline{x}_{i,G}$ and $\overline{v}_{i,G+1}$ with probability CR using Equation 10.
     - 2.1.4 Select a member for next generation. If $T(\overline{u}_{i,G+1}) < T(\overline{x}_{i,G})$ then $\overline{x}_{i,G+1} = \overline{u}_{i,G+1}$, otherwise $\overline{x}_{i,G+1} = \overline{x}_{i,G}$.
3. Find the minimum $T(\overline{x}_{c,G+1}) = \min_i T(\overline{x}_{i,G+1})$, $i = (0, 1, \dots, NP-1)$.

4. Unless $T(\overline{x}_{c,G+1})$ is good enough or the maximum number of generations G is reached, go to Step 2.

3.2.3 Practical considerations for the use of DE

Following notes on the use of the DE can be given [22]:

- After initialisation population should be spread as evenly as possible over plausible objective function area.
- Usually CR should be considerably lower than 1.0 (for example 0.3), sometimes if no convergence can be achieved values of 0.8 or more may help.
- For many applications $NP = 10D$ is a good size for population. $0.5 < F < 1.0$ is also a common choice.
- The higher the population size, the lower the weighting factor F should be.
- It is a sign of good convergence if the best population member changes a lot during the first generations, even if the value of the objective function decreases slowly.
- It is not necessarily a sign of non-convergence if the value of the objective function does not decrease for long periods during the minimisation process. The minimisation, however, could take a long time; increasing the size of the population NP might be necessary.
- If the value of objective function of the best population member decreases rapidly, it might indicate that optimisation gets stuck in a local minimum.
- The proper choice of the objective function is very important for the success of the minimisation process. If the objective function "hides" too much information, convergence can take a long time.

The DE/rand/1 notation specifies that the vector to be changed is randomly chosen, and that one weighted difference vector is added to it. DE/rand/2 means that instead of using

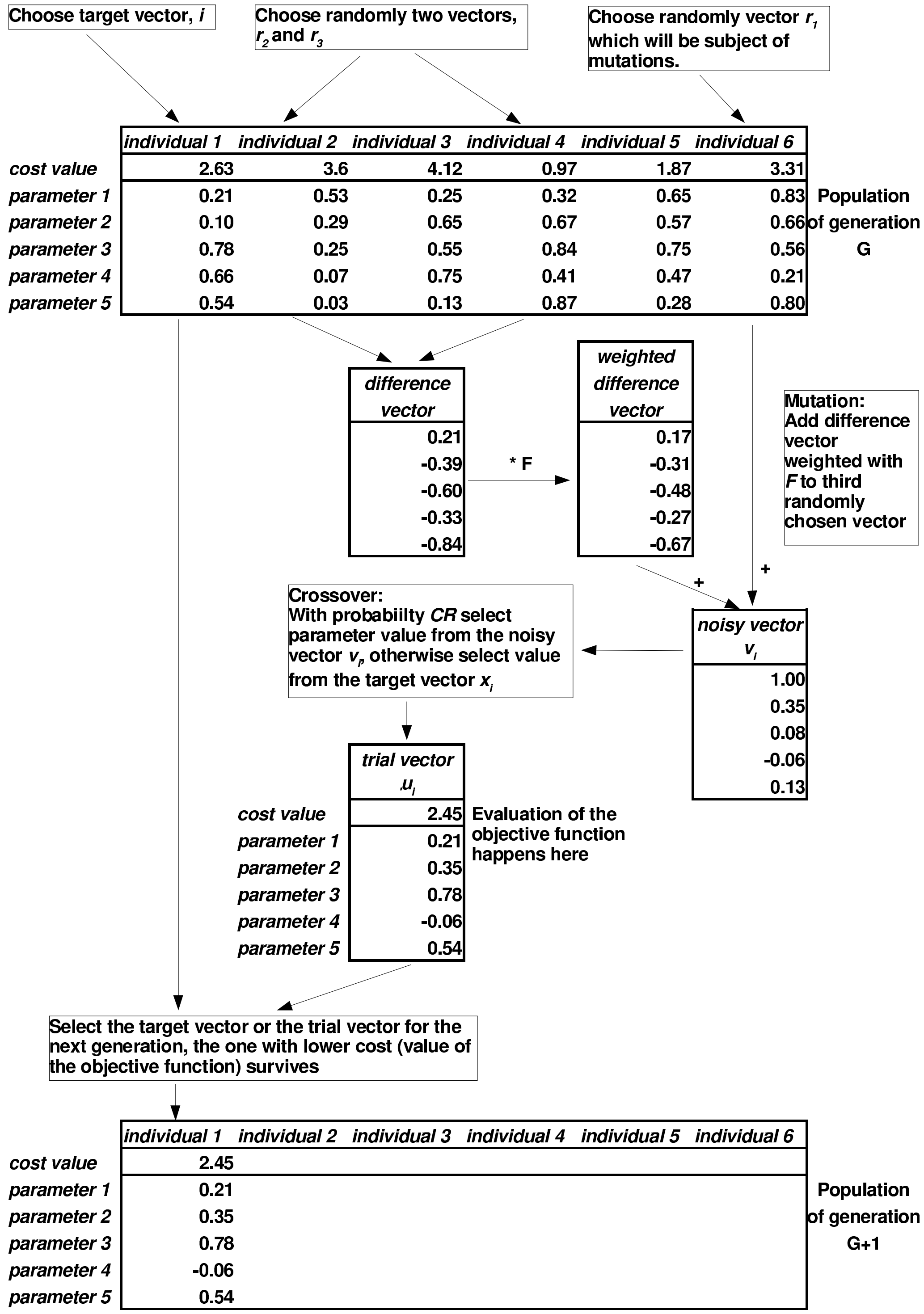

Figure 11: An example of using differential evolution (DE/rand/1). The variables used are $D = 5$, $NP = 6$, $F = 0.80$, $CR = 0.50$.

one weighted difference vector, two such vectors are added. Other schemes are for example DE/best/1, where instead of randomly choosing the vector to be changed, the best vector (vector with the lowest value of the objective function) of the current generation is chosen to be transformed. This scheme is more directed since it always concentrates on the best vector of the current population. This will lead to faster convergence than DE/rand/1, but the trade-off is that finding a local minimum instead of the global minimum is much more probable.

The DE algorithm is computationally intensive, like all the other genetic algorithms. While the algorithm itself uses only simple floating points operations, calculating the value of the objective function might be computationally very expensive. This is a serious drawback in many situations since DE needs to evaluate the value of the cost function every time a new population member is generated, which may happen thousands or even millions of times during the optimisation process. However, DE like other genetic algorithms is quite easily modified to a parallel computing version. There are several possible models for parallelisation. The most simple model, called the standard model, distributes only the evaluation of the cost function to other nodes. Efficiency might be low since the whole generation is reproduced at the same time and the evaluation of all the new population members has to be done before continuing to the next generation. When the evaluation of some population members takes longer than the other members for any reason, the rest of the nodes will stay idle. The problem of low efficiency can be solved by using a migration model. In the model all nodes have their internal populations which evolve at their own pace, and population members migrate periodically between nodes. While the migration model uses hardware efficiently, it might not be very effective in solving the problem faster, because the migration model is logically different from sequential non-parallel or standard models. Also the standard model can be made more efficient by using work queues and relaxing the requirement of evaluating all the population of the generation at the same time. [32]

3.2.4 Applications

Most popular application areas and numerous applications using genetic algorithms are reviewed by Alander [33]. While many of the applications presented use generic genetic algorithms, not specifically DE, same application areas apply also for DE. Genetic algorithms are most applicable when the problem is inherently an optimisation problem or it can be converted to an optimisation problem. Optimising controllers and mechanical structures have been an important application area, as well as solving various scheduling problems. Genetic algorithms have also been used for training and designing the topology of neural networks, optimising filters in signal processing applications and in various tasks in field of robotics. Some applications using genetic algorithms are presented in following.

Genetic algorithms have been used to find an optimal shape for a diesel fuel injection equipment cam [34]. The cam controls the fuel injection cycle and combustion process which affect the thermodynamic efficiency of of the engine. An optimal cam shape will lead to low fuel consumption, high engine performance and low exhaust and noise emissions. The shape of the cam was optimised using genetic algorithm together with a simulation software. Genetic algorithm produced different cam shapes for the simulation software to evaluate. There were various goals and design constraints. Genetic algorithm was successful in finding a good shape for the cam. General considerations on how to represent a geometrical shape for use with genetic algorithms are presented in [35]. A shape has to be presented by individual chromosomes forming a complete shape. The method for presenting a shape is important because the search space should be as small as possible (i.e., as few control variable for the shape as possible) while still being expressive enough to present an solution accurate enough. Additionally, the method should be computationally non-intensive so that computation of the shape does not form a bottleneck for the optimisation process. Proposed methods are Bezier-curves and B-splines.

DE has been used to tune automatic train operation (ATO) systems [36]. The ATO controller schedules train movement from a departure to the next scheduled stop under normal conditions. Usually such controllers are not designed to handle situations such as deviations from normal schedule, e.g. the train should try to get faster to the next station than normally. Some of these problems have been alleviated using fuzzy ATO controllers which can automatically change the operation status when there are unexpected deviations. Such systems fare as well as human operators, but tuning fuzzy ATO controllers is a very complicated task because there are many factors to be considered like interstation distances, gradient profiles, different schedules and speed limits. DE has been used for optimising fuzzy ATO controllers successfully. The optimised controllers were able to follow schedule more closely, have a lower jerk factor (less strong braking or acceleration) and additionally have a lower energy consumption. Also, under a tight schedule, when the goal is to get to the next stations as fast as possible while still obeying speed limits, all previous advantages were observed together with a faster journey time.

DE has been applied for scheduling problems with non-linear objective functions and multiple dependent restrictions. One scheduling application is presented in [37]. DE has been used to schedule core blowers in foundries. Scheduling is important since the core blowers are strongly connected to automatic molding plants with high production costs. If the cores are not produced early enough the molding plants will stand still while waiting, and if the cores are done too early they will grow old and lose quality while waiting for the molding plant to be available. Scheduling generated by an approach based on DE was good.

4 Predicting diffusion of innovations using machine learning methods

In this work the main focus is to forecast diffusion of an innovation in a country where the product has not been introduced yet. The prediction can be divided to few separate cases, depending on whether the product has been sold already in some countries or in none at all. Also similar products can be considered.

Prediction methodology is structured in Figure 12. The most relevant part is the function predict which predicts the diffusion for a product in a country based on other similar countries. Finding similar countries can be automated since many of the parameters of countries can be expressed as numbers (population, gross national product per capita, quality of telecom infrastructure, etc.). On the other hand, finding similar products is a harder task to automate, since products do not have such easily numerated attributes. For that reason finding similar product is left to humans. Testing prediction with similar products is also left out of this master's thesis completely because of lack of data of large enough number of products.

While the proposed method does not necessitate the use of any diffusion model, the Bass model is used to estimate the product and market specific diffusion patterns. The Bass model has a well-grounded link to behavioural theory, since it takes into account both internal and external influence and information. The estimation of parameters is also easier than with many other models since there is only three parameters. Additionally, the Bass model has been widely applied when international diffusion has been studied, also in other parts of the project this master's thesis is part of, so comparisons with other studies is more feasible when the Bass model is used [38]. The output of the function predict (in Figure 12) is thus the three parameters of the Bass model.

In this section the proposed method is presented: Finding similar countries using SOM, using the map of similar countries for prediction, and finally, how the evaluation of the prediction performance is done.

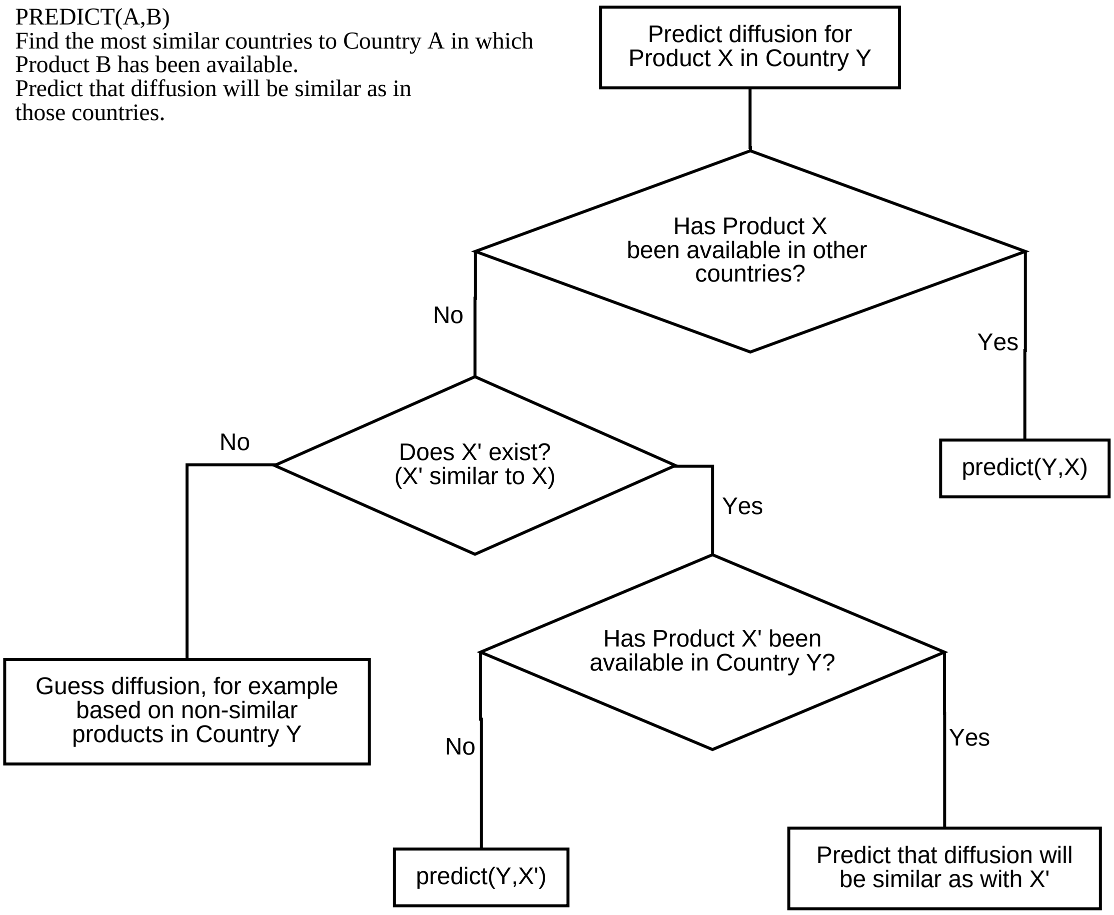

Figure 12: The decision tree of different kinds of prediction cases.

4.1 Finding similar countries

There are several possible methods for finding similar countries. It could be done by hand, since in several cases it is quite easy to determine few similar countries, often by just choosing geographically closest countries. For example, Sweden is quite similar to Finland. The task gets harder when similar countries could be found even in different continents and not in the immediate neighbourhood. For example, Japan might be more similar to some European or American countries than to its geographical neighbours.

So, a method for determining similar countries automatically from some parameters of the countries is needed. Previously in Section 2.2, some studies were presented in which cosmopolitanism, mobility and percentage of women in labour force were identified as most important factors by which similar countries (in context of diffusion of few products) can be determined [9].

However, using more parameters for countries would be useful, since it cannot be determined beforehand which parameters are essential for capturing similarities between different countries for all different products. Also, not all parameters are known for all countries, less developed countries tend to have less complete statistics available. Therefore, the method has to cope with incomplete data. Additionally, similarities between countries might depend on the product studied, in other words, there should be a way to get different results with different products.

4.1.1 Generating a map for finding similar countries

The Self-Organizing Map (SOM) is the selected method. Data of all countries is trained in the map, and countries are ordered depending on their attributes, i.e., similar countries will be close to each other. SOM is not disturbed by incompleteness of the data (to a certain point), so countries with some missing parameters are not a problem. However, incompleteness tends to cluster to the least developed countries. Mainly the countries of African continent are problematic. An example of a SOM teached with parameters of the countries is shown in Figure 13. This kind of map is called a country map. Parameters used for training include various social and economical parameters, like population, urbanisation percent, gross domestic product, etc. The exact list of used parameters can be found in Section 5.1.1. The country codes can be found in Appendix 1.

The map does not take different kinds of products into account at all. The map is always the same. There are a few ways to make the map optimised for a specific product.

A simple method is to use also sales data of the product. Training of the map is done with the parameters of the countries and the sales data of the product from those countries where it is already available. This way the map and the relations of countries will be affected by the diffusion of the product.

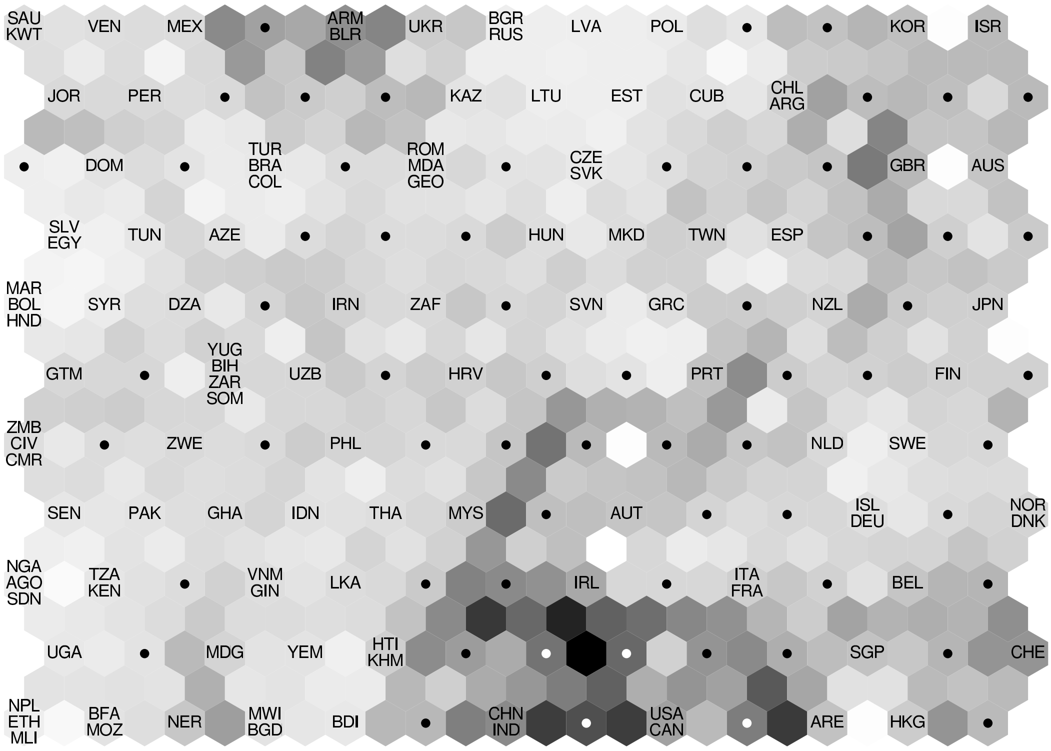

Figure 13: A SOM teached with parameters of the countries.

A more complicated method is finding “weights” for country parameters. This means that the importance of specific parameters for the diffusion of a product is determined. Because of incompleteness of data and high dimensionality, this is problematic for traditional statistical methods. So, the sales data of product to generate another map. After this, there will be two maps: one shows relations of countries based on their parameters, and another map shows relations of countries based on the diffusion of a specific product as shown in Figure 14.

Usually countries go to different positions relatively to each other in the two maps which have been trained by using parameters of the countries and by using diffusion of a product in the countries. To get the country map optimised for the product, countries should be in the same places (relatively to each other) in both maps, or actually they should be at the same distance from each other. The same should hold true for all country pairs. This can be done by finding suitable weights for country parameters. There might be, for example,

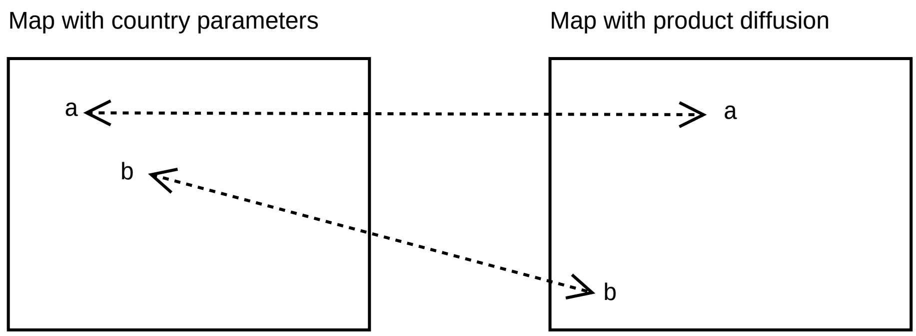

Figure 14: Two maps trained with different data, where a and b are countries.

a product which is mainly used in urban areas. For such a product weighting country parameter of urbanisation percent over others might be correct.

Finding weights to make two maps similar is a highly complicated and non-linear task. Training of SOM is not a deterministic process and a very small change in the weights can generate a completely different map, because even minor changes will escalate during the training process. This can be considered as a form of the "butterfly effect" which is named after a presentation on chaos theory by one of its first experimenters, Edward Lorenz, "Predictability: Does the Flap of a Butterfly's Wings in Brazil set off a Tornado in Texas?". Results were later presented in [39]. The article states that long term prediction of complex systems is impossible because underlying conditions can never be grasped at sufficient – infinite – precision. Nevertheless, even if the maps seem to be quite different in the global scale when trained with slightly different weights, they are quite similar in the sense that the relations between countries are preserved. The same countries tend to be always close to each other.

The first step for generating a method for making to different SOMs similar is to develop a method for measuring the similarity. This is done by comparing each country pair which can be found in both maps. Ideally, every country pair would be at the same distance from each other at both maps. This is almost never the case.

Computing the average of distance differences between all country pairs in both maps is the most self-evident method. The method is presented in Algorithm 5.

Algorithm 5 Average distance difference

1. Find all countries $c_{1..N}$ that exist in both maps where N is number of countries.
2. Form all possible country pairs, $p_i = (c_x, c_y)$, where $x = [1..N-1]$, $y = [x..N]$ and $i = \left[1..\frac{N(N-1)}{2}\right]$.
3. For all country pairs p_i calculate distance D_i.
   3.1 Calculate distance d_{i1} between the countries in first map.
   3.2 Calculate distance d_{i2} between the countries in the second map.
   3.3 $D_i = |d_{i1} - d_{i2}|$
4. Calculate the average of values in vector D.

However, it does not work because there will most definitely be differences between maps in the "global" level even if the smaller neighbourhoods are similar. The reason for this behaviour is Step 3.3 of Algorithm 5. If both d_{i1} and d_{i2} are large numbers (the countries are far from each other in the both maps), it should not matter at all if the distance changes. When the countries are far from each other then they should not be allowed to have an effect on the calculation. What should be done instead is to check whether the country pair is close to each other at least in one of the maps and to weight the result based on that. A better choice for Step 3.3 would be for example

$$D_i = e^{-min(d_{i1},d_{i2})}|d_{i1} - d_{i2}| \tag{11}$$

where $e^{-min(d_{i1},d_{i2})}$ will weight the equation based on the smaller of the distances in the two maps. This will give the desired result, since the absolute difference of the distances d_{i1} and d_{i2} affects heavily the result only when one of them is small.

It is possible to optimise or to find optimal weights for parameters so that maps become similar now that there is a method for measuring similarity of two maps. Differential Evolution (DE) is an optimisation method suitable for non-linear problems, and it works even if the optimised function does not always give the same results with the same values, which is a possibility in this case. The price for these valuable characteristics is that the optimisation is not guaranteed to find an optimal solution and the time needed for optimisation is usually long because there is a lot of unneeded work done. Usually DE works quite well if a sufficient amount of generations and a population large enough can be used. A large population and a large number of generations mean that more time will be spent in optimisation. Because handling one member of a new generation (making the new member by mutation and crossover, and evaluating its performance) takes a long time, there is a problem. In this case the step of measuring the prediction performance requires training a SOM with the weights of the population member. The time needed for training a map is often measured in minutes. This leads to long optimisation times, but at least in the testing phase it is still possible to use a method which will take a few days to apply.

4.1.2 Using the map for prediction

After we have a map of similar countries, whether it is optimised for a product some way or not, the map has to be used for prediction. After finding the country – for which the prediction is needed – in the map, neighbouring cells are checked for other countries in which the product has already been available. The estimate for diffusion is then combined from diffusion information of the neighbouring countries. Closer neighbours in the map affect the prediction more than those that are further away. The weighting is done by formula

$$w = \frac{1}{d+1} \tag{12}$$

where d is a distance in the map between the country and its neighbour. Combining is done by taking the weighted average of the Bass model curves that are estimated from the real diffusion. The Bass model is used because combining real diffusion curves is non-trivial; different countries have different number of diffusion data points available and real diffusion tends to be chaotic at least in the beginning. At the end the combined graph can be fitted again to the Bass model for example to make comparison to the “correct” values during testing easier.

There is an example of the map used for prediction in Figure 15. When predicting the diffusion of cellular phones in Norway, first step is to find it in the map. Norway is near the upper right corner. The closest neighbour in the map is Denmark, which is in the same place in the map. At the distance of two elements there are many countries, including USA, Canada, Singapore, Switzerland and Finland. The prediction based on the closest countries is presented graphically in the Figure 16. The prediction is quite good, even though in this case the prediction would have been better if it was based only on the closest country, Denmark.

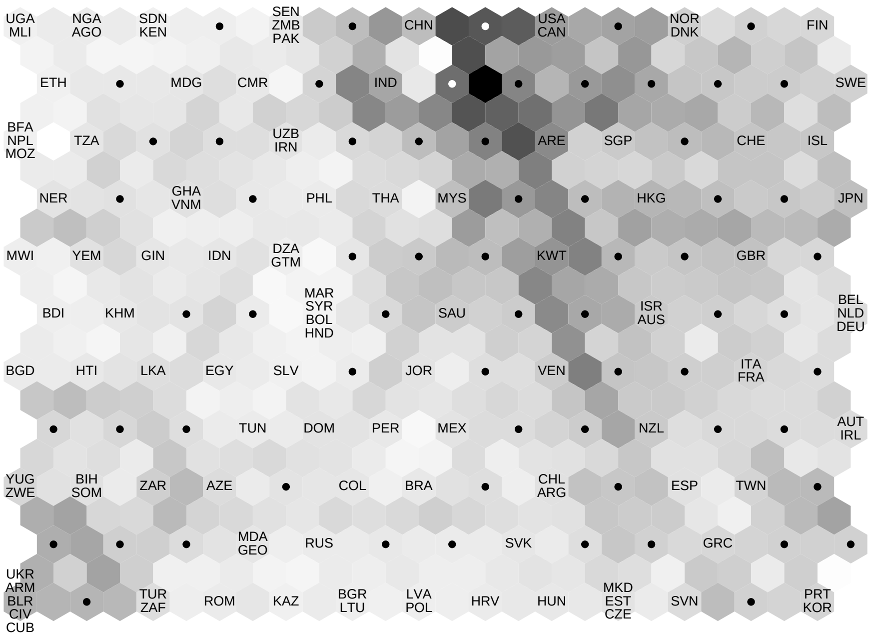

Figure 15: A SOM optimised with cellular phones sales data.

This leads to question how many neighbouring countries are needed to make the prediction to work? In Figure 17, it is shown that five neighbouring countries seem to be enough, after that results do not get better or they get even slightly worse. There was not much difference when using different kinds of distance based weighting, but since the idea that closer countries are more important is quite natural, Equation 12 is used.

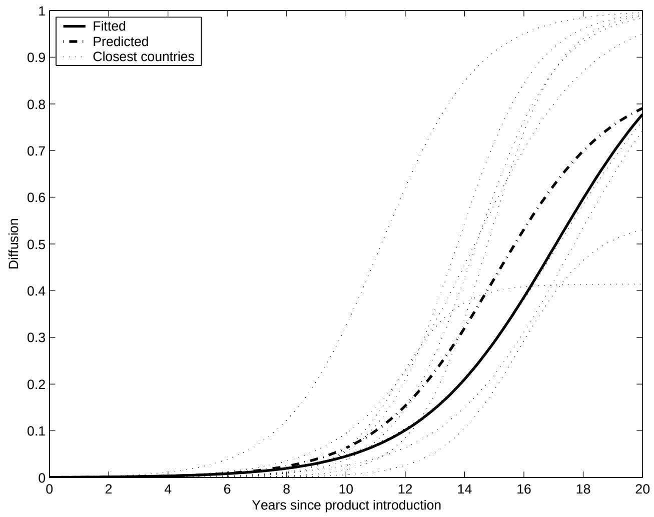

Figure 16: Graph of diffusion of cellular phones in Norway (graph labeled fitted), diffusion in the closest countries (from the map in Figure 15), and the predicted diffusion.

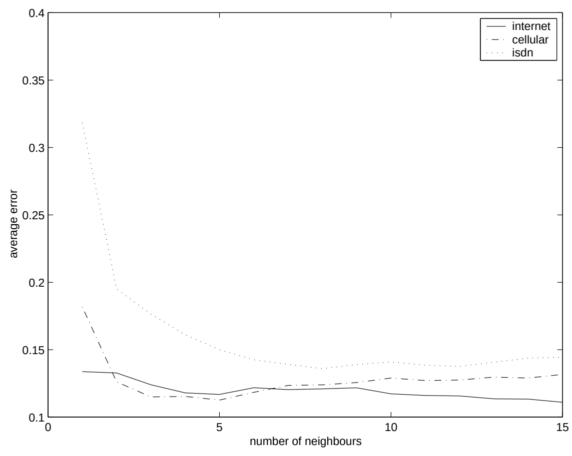

Figure 17: Prediction performance with a different number of neighbouring countries used for prediction.

4.2 Evaluating the prediction performance

The next problem for testing is how to measure the goodness of prediction. Basically, when the Bass model is used there are three parameters which could be directly compared. However, direct comparison of the parameters is not a good idea, since p and q have a quite interdependent effect on the diffusion curve. By shifting one of them down and the other one up one can get almost the same curve. By directly comparing p and q values these kind of interdependencies are missed and the comparison does not reflect reality.

So, the measuring has to be done using the curves of the Bass model, either analytically or approximately using computed values. Since the formula of the Bass model is quite complex for analytical handling, the approximation was deemed to be accurate enough.

Computing differences between two graphs of the Bass model can be done with many different methods. A couple of widely used measures are Mean Square Error (MSE) and correlation. Other two tested methods are a common distance metric, Euclidean distance and Mean Absolute Error (MAE). Graphs with different methods are presented in Figure 18. Although all of the different methods provide reasonable metrics that regard interdependencies of p and q parameters, still some are better than others: MSE belittles differences too much (a large area of white in the graph in the middle), and correlation behaves weirdly when there is a large difference between graphs (the left lower corner of the graph).

Additionally, there is one more variable m which serves as scaling factor for the graphs. The correlation is not changed at all by the scaling of the graphs, so that metric is not good because one important factor is not measured at all. MSE and Euclidean distance square the errors, so differences in the value of m are given too much importance. The best method seems to be MAE, since there is a balance between not noticing changes in m at all and giving it too much weight. The equation for the used distance function MAE is

$$\frac{1}{n}\sum_{j=1}^{n}|a_j - b_j| \tag{13}$$

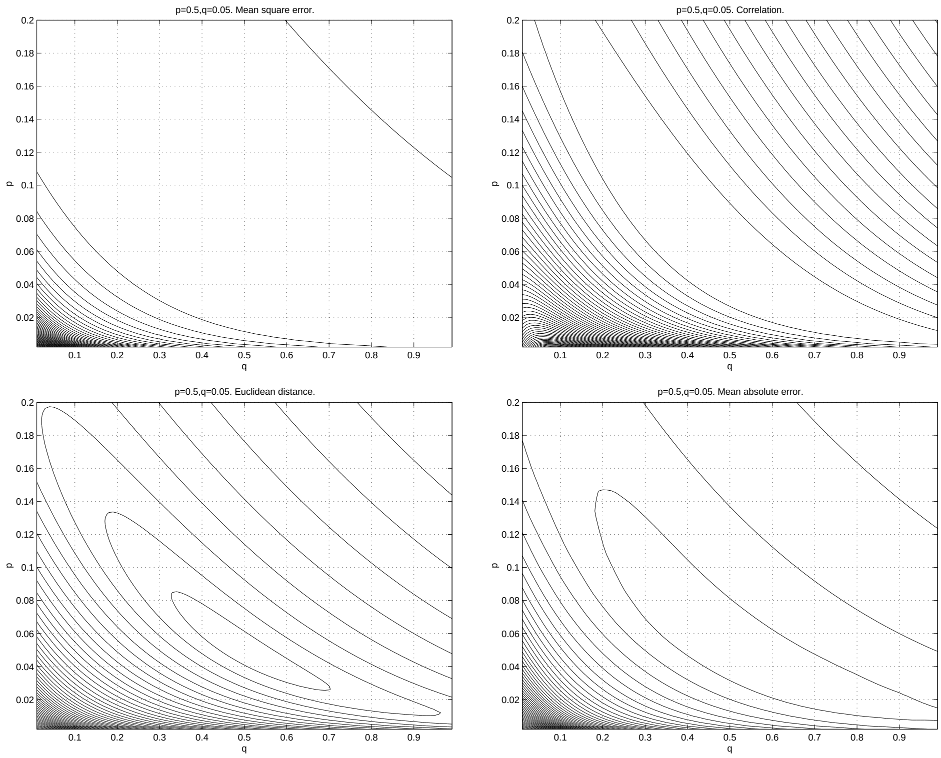

Figure 18: Comparing two graphs of the Bass model with different methods. Graphs are contour maps, lines show areas with the same difference level.

where a and b are generic vectors of length n.

There is still one problem with MAE. Some scaling is needed in case where both graphs to be compared have very small value of m, i.e., very low diffusion. So, before the comparison graphs are scaled so that higher of them reaches 1.0. Since the cumulative Bass curve is strictly increasing, its last value is the largest. The used function becomes

$$\frac{1}{max(a_n, b_n)} \frac{1}{n} \sum_{j=1}^{n} |a_j - b_j|. \tag{14}$$

5 Experiments and results

This section introduces the used data, shows experiments, and finally, gives the results.

5.1 Data

5.1.1 Country data

Some of the data for countries and all data for products was obtained from ITU-T [40] telecommunications database.

There exists data, called country data, describing charasteristics of each country. Most data of the was from years 1990 and 1991. The reason for not using newer values is that newer values would be "in the future" compared to the introduction time of the products which have been available since eighties in most countries. The country data consisted of 16 parameters as follows:

- The people:
  1. Population.
  2. Population growth percent (per year).
  3. Number of ethnical groups.
  4. Number of used languages.
  5. Urbanisation percent.
- Various indices which combine several statistics:
  6. Political and economic risk rating.
  7. HDI (Human Development Index) from year 1990.
  8. HDI (Human Development Index) from year 1991.
  9. TAI (Technology Advancement Index).

- Economic factors:

  10. Gross domestic product per capita in US dollars.
  11. Gross domestic product per capita with purchasing power parity.
  12. Inflation.

- Telecommunications and technology:

  13. Telecom investments per capita.
  14. International telephone traffic minutes per capita.
  15. Personal computers per capita.
  16. The year mobile communications was started.

Some of the parameters are raw, since they are simple statistics such as population and inflation. Other parameters have a huge list of different parameters combined, political and economic risk rating and HDI foremost.

These parameters express, for example, internal wealth and development state (GDP, country risk, human development index) and the internal diversity of the country (number of ethnical groups and used languages). As said in Section 2.2, it has been noticed earlier that cosmopolitanism, mobility and number women in labour force were the important differentiating factors for the diffusion of certain technological products in different countries. While the used parameters do not include explicitly these factors, they are implicitly included in various parameters.

There were data from 113 countries. All of the parameters were not known for all countries, but as it has been said earlier, it does not matter when using SOM if the number of missing parameters is not too high. However, this was the case with some of the least developed countries, so many of mainly central and southern African countries had to be left out.

Raw values of parameters should not be fed directly to the SOM. Normalisation, which was discussed in Section 3.1.4, has to be performed since the scale of the parameters varies too much: Population has always the greatest value, and thus, it has way too much weight when creating the order in the map. So the parameters have to be normalised in a way or another. By testing few of possible normalisation methods it was determined that simple scaling of parameters to $[0 \dots 1]$ is sufficient and more elaborate normalisation methods (scaling by standard deviation or mean) do not offer noticeable advantage.

5.1.2 Product data

Product data were obtained solely from ITU database of World Telecommunication Indicators. There were time-series of diffusion of several products. The following time-series were used:

- ISDN subscribers.
- Cellular mobile phones.
- Estimated Internet users.

Since the time-series have absolute numbers they had to be normalised between countries.

For the Internet users it is clearly best method to divide the value by population of the country. Thus, the number will be the percentage of the population using the Internet. Population was included also as time-series, so division was done by yearly population values. Population does not change much for many of the most developed countries, so for these countries considering population as a time-series is not essential. However, there are some countries where the population growth must be considered.

For the cellular mobile phones normalisation by time-series of population was also done. Since one person can have many cellular phones, this number could rise to values of over 1.0 even after the normalisation.

For the ISDN subscribers best source for the normalisation would be the number of households in the country. This data are not available widely and the numbers are not as reliable as the population statistics, so the time-series of population was used for normalisation.

After the normalisation maximum percentages probably cannot rise above 0.4, when every household would be an ISDN subscriber.

5.2 Experiments

Experiments included dividing countries of the world to different groups. Some of the groups were solely geographical and some of them were based on political or economical facts. Groups are listed in the following in order of probable “difficulty” of prediction. The more developed and more homogeneous countries are inside the group, the easier the prediction should be. The countries in the groups are listed in Appendix 2. The following groups were used.

- EU (European union), all countries which are members of European Union. All countries are quite homogeneous, wealthy, industrialised nations. The method should work at least within this easiest case.
- OECD (Organisation for Economic Cooperation and Development), all countries belonging to the organisation. Includes the most industrialised nations, a group like EU but more diverse.
- EU candidates are countries which are applying for EU. More diverse group of countries, most of which have had free market only for a short while and some of them are developing very fast.
- South-America includes all countries of South-America.
- CIS (Commonwealth of Independent States), countries which used to belong to Soviet Union, not including Baltic countries.

Two following experiments have been tested with the country groups. The first experiment is to drop one country out and to test if the diffusion of a product can be predicted in that country by using the knowledge of diffusion in all the other countries. This kind of prediction is based on the most similar countries. The test is not done for all countries, but to one country group at a time. After the experiment it is computed how close the prediction was on average. This is a baseline test: if this does not work relatively well with the countries of EU, then there is not much hope for more interesting cases to work.

The second experiment is to drop one country group away and to test if the diffusion can be predicted for the countries in that group by using the knowledge of the diffusion from all the other countries in the world. This is a more interesting case, since the roll-out of new products might make possible to predict diffusion in the parts of the world where new products are often introduced later than for example in Europe.

The experiments are done with the following options for the step of finding most similar countries. First, countries are always ordered only by their unweighted parameters. This means that neighbouring countries are always the same, and they do not depend on the product. In Section 2.2 it was noted that country-groupings depend on the product, so this is not an optimal solution. Second, the test is optimised by adding information of the diffusion of the product to the parameters of the countries. The order of the countries in the map is affected by how the diffusion of the product in the countries has advanced. With this optimisation method diffusion data are not added for those countries which are currently tested, because the data would not be available when doing the prediction in real case. Third, SOM is optimised with the DE-based method described in Section 4.1.1 for different products. This also reveals which parameters of the countries affect mostly the diffusion for that product.

5.3 Results and discussions

First, some separate prediction results are spotlighted. They demonstrate both good and bad prediction results and some reasons for particular results. After that, more comprehensive test results with average prediction errors are presented.

Some examples of prediction results in different countries for different products are presented in Figure 19. The figures include a fitted graph, which is a Bass model graph fitted to the real diffusion data, and a predicted graph, which is the graph that the method has predicted the diffusion to be. There is also fitted graphs from the five closest countries.

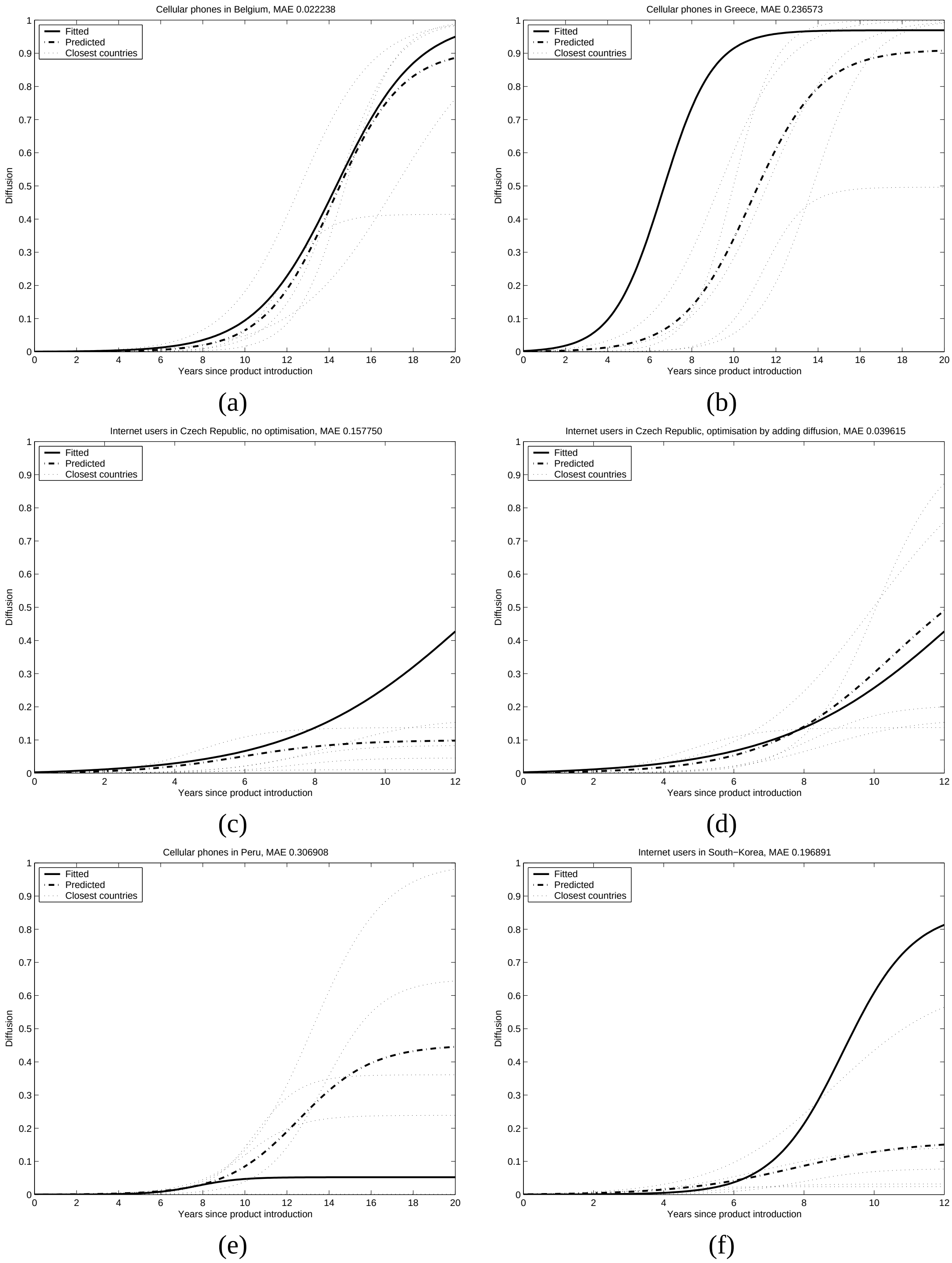

Figure 19: Examples of prediction results: (a) Belgium; (b) Greece; (c) Czech Republic (no optimisation); (d) Czech republic (with optimisation); (e) Peru; (f) South-Korea.

Figure 19.a presents prediction results for the diffusion of cellular phones in Belgium. The prediction was almost flawless, and it is no surprise since many of the closest countries found had quite similar diffusion. Figure 19.b has also prediction results for diffusion of cellular phones, this time in Greece. The prediction did not work nearly as well as in the case of Belgium. The reason for the result is that in Greece the diffusion of cellular phones has started later than in the closest countries found by the method. Therefore, the demonstration effect – imitation in the scale of countries – has led to faster diffusion in Greece than in the closest countries. The shape of the predicted diffusion is still quite close to the fitted graph, but the prediction is late by several years.

In the second row there are two figures of prediction of diffusion of Internet users in Czech Republic. On the left, Figure 19.c, no optimisation has been used when generating the SOM used for prediction. The prediction result was quite bad. On the right, Figure 19.d, optimisation by adding also diffusion information to the SOM has been used. The prediction result was very good. These two graphs demonstrate that a) optimisation is at least sometimes profitable, and b) it is advantageous to base the prediction to a weighted average from several closest countries, not only the closest. In this case, the fitted graph is quite dissimilar to the diffusion in the closest countries, yet a good prediction has been achieved because the prediction has been based to several graphs from different countries. If only a graph from the closest country had been used for the prediction, a good result could not have been achieved.

Figure 19.e presents prediction of the diffusion of cellular phones in Peru. The result seemed to be bad. However, in case of Peru and cellular phones, there was only 9 years of data available. After more than 9 years even the fitted graph is an extrapolation from the previous data. The extrapolation is very pessimistic, since according to it the market will be saturated after ten years when not even 10% of population has a cellular phone. The predicted graph extrapolates that the diffusion will steadily rise for ten more years. How the diffusion will really happen is not known, but barring any dramatic changes in the country the diffusion should increase. So, actually the prediction is, most probably, more accurate than the graph fitted to real data. It would make sense to calculate the difference for only the duration that the real data exists, but the problem does not exist for the most of the countries and products used for testing, and adjusting the duration of error calculations country-by-country basis would lead to complications in the test programs.

Finally, Figure 19.f presents the prediction of the diffusion of Internet usage in South-Korea. The prediction is really pessimistic compared to the reality. In this case the reason is the rapid progress of South-Korea, since it has advanced far faster than other countries in the area and especially in the area of Internet usage they have advanced to one of the leading countries of the world. So, there are no countries that are in general sense similar to South-Korea, and with this method this leads to the prediction performing quite badly. Fortunately, such countries are rare and they do not pose a serious hindrance to the method.

Prediction errors for the EU countries are presented in Figure 20 and average prediction errors are presented in Table 1. Please note that all the figures do not have all the countries because if there has been no data available for a product in a country the prediction accuracy cannot be measured. The results were good for cellular phones and Internet users, with ISDN subscribers the results are slightly worse.

Average prediction errors were lowest for the cellular data set. Results with both optimisation methods were slightly better than the results without optimisations, but the difference was minimal. While the prediction error was higher when the whole country group was missing, the results are still very consistent and the variance stays low. However, optimisation actually degraded the prediction accuracy when the whole country group was missing.

With the Internet users the results with both optimisation methods were slightly better in all cases, even with the whole country group missing. With optimisation by adding diffusion information the difference was significant in the case of the whole country group missing, the prediction error was lower in almost every country. Also, the variance stayed low when optimisations were used, and without optimisations the results with the whole country group missing had quite large variance.

The prediction for the ISDN subscribers worked almost as well as with the other data sets when tests were done by dropping only one country, but the results with the whole country group missing were not very good. The reason for worse results with the ISDN subscribers is that EU forms are a large part of the countries that have ISDN subscriber data available, dropping them will unsurprisingly make the prediction quite hard.

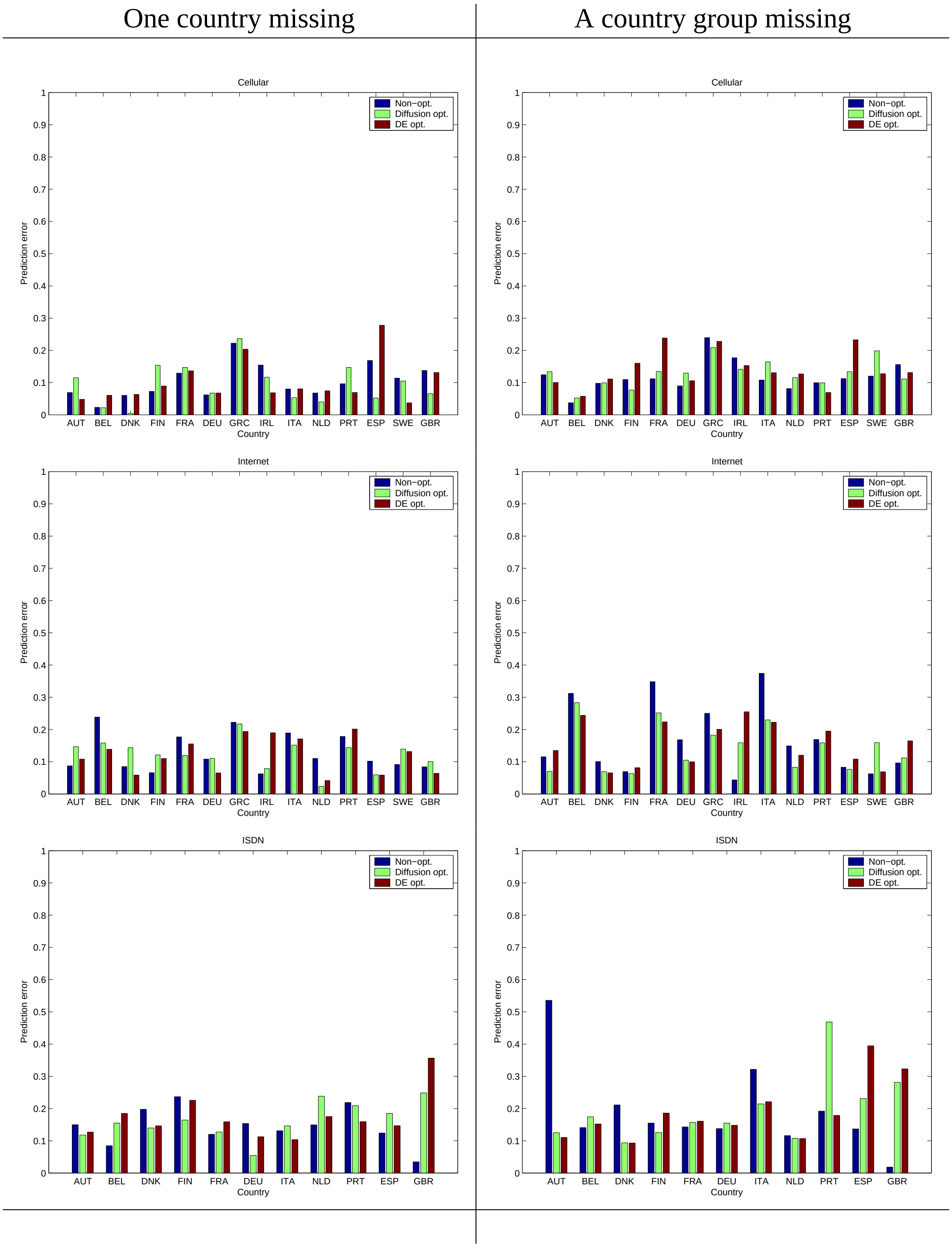

Figure 20: Prediction errors with different optimisations for all products in the EU countries.

Table 1: Average prediction errors for products in the EU countries: upper results with on country missing, lower results with the whole country group missing.

Product	Non-optimised		Diffusion optimised		DE optimised	
	Average	Variance	Average	Variance	Average	Variance
Cellular	0.1042	0.0028	0.0948	0.0040	0.1007	0.0045
Internet	0.1286	0.0035	0.1222	0.0022	0.1204	0.0032
ISDN	0.1455	0.0034	0.1624	0.0032	0.1724	0.0049
Cellular	0.1190	0.0023	0.1286	0.0018	0.1410	0.0033
Internet	0.1670	0.0122	0.1429	0.0053	0.1558	0.0045
ISDN	0.1917	0.0184	0.1939	0.0115	0.1887	0.0087

Prediction errors for the EU candidate countries are presented in Figure 21 and average prediction errors are presented in Table 2. In this case, most consistent results were achieved when predicting the Internet users. The prediction for the cellular phones is worse and the predictions for ISDN subscribers are meaningless.

The results for the Internet users and cellular phones were interesting. While the results were not as good as for the countries of EU, they were still quite good. What makes the results interesting is the fact that predictions with both optimisation methods, particularly with the optimisation by adding diffusion information, were better than the results with no optimisations, and, more importantly, the prediction is better in the case of the whole country group missing. The reason might be that while the countries in this group are quite similar – they are close to each other in the non-optimised map – they have still quite different diffusion charasteristics for the products. So, the prediction works better when the diffusion information is taken into account, or when the whole country group is left out, which helps because some similar countries with non-similar diffusion will not be used for computing the prediction.

Prediction errors for the South-American countries are presented in Figure 22 and average prediction errors are presented in Table 3. In this group there were no countries with ISDN subscriber data available. The prediction of Internet users worked very well, as well as it did work among the EU countries. The prediction of cellular phones was worse. Average errors are similar to the errors in the same data set among the EU candidate countries.

In this case, there were no particular surprises in the prediction performance. Results with

One country missing	A country group missing

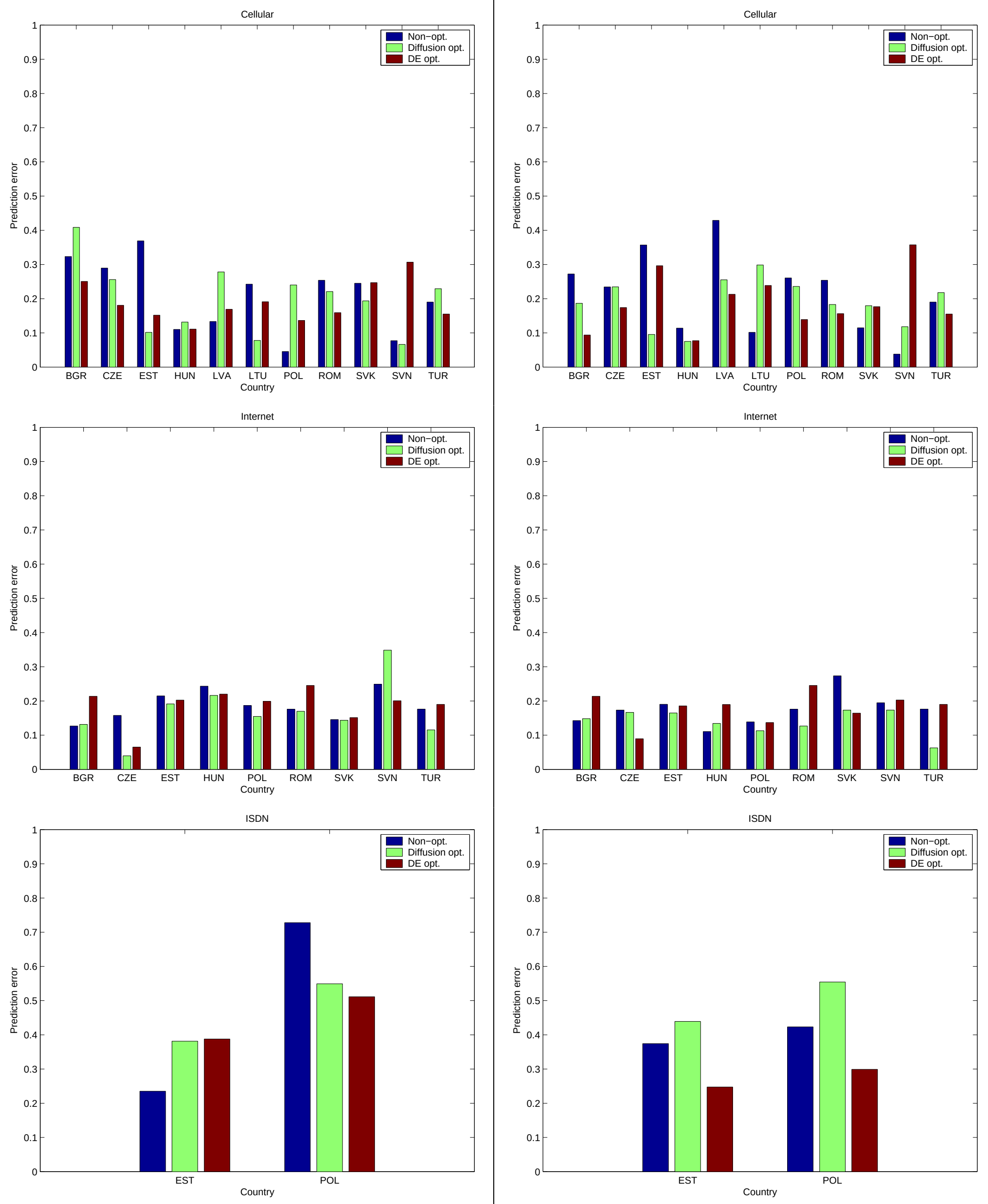

Figure 21: Prediction errors with different optimisations for all products in the EU candidate countries.

Table 2: Average prediction errors for products in the EU candidate countries: upper results with on country missing, lower results with the whole country group missing.

Product	Non-optimised		Diffusion optimised		DE optimised	
	Average	Variance	Average	Variance	Average	Variance
Cellular	0.2072	0.0109	0.2004	0.0103	0.1872	0.0034
Internet	0.1863	0.0018	0.1678	0.0071	0.1875	0.0027
ISDN	0.4815	0.1214	0.4654	0.0141	0.4496	0.0076
Cellular	0.2151	0.0138	0.1891	0.0048	0.1889	0.0070
Internet	0.1751	0.0021	0.1404	0.0013	0.1796	0.0021
ISDN	0.3988	0.0012	0.4969	0.0066	0.2733	0.0013

the optimisations are slightly better for most cases and the results with the whole country group missing are slightly worse than the results with only one country missing. Still, the prediction errors when predicting the Internet users are delightfully low even with the whole country group missing.

Prediction errors for the CIS countries are presented in Figure 23 and average prediction errors are presented in Table 4. The prediction errors were highest among the tested country groups as was expected because of lowest development level among the tested country groups, and there are no surprises considering results with the different optimisations and whether only one country or the whole country was dropped.

Prediction errors for the OECD countries are presented in Figure 24 and average prediction errors are presented in Table 5. In this case only tests with dropping one country at a time were carried out. The results are almost as good as with the EU countries, only with ISDN subscribers the prediction errors are considerably higher. The results with optimisations are slightly worse than with no optimisations in all but one case.

Generally the optimisations were not very relevant. In several cases results were even somewhat worse when using the optimisations, though the differences are not great. However, in some cases the optimisations did offer substantial benefits, most noticeably with the EU candidate countries.

Another observation based on all the results is that the prediction without all the countries in a country group worked about as well as prediction with only one country dropped.

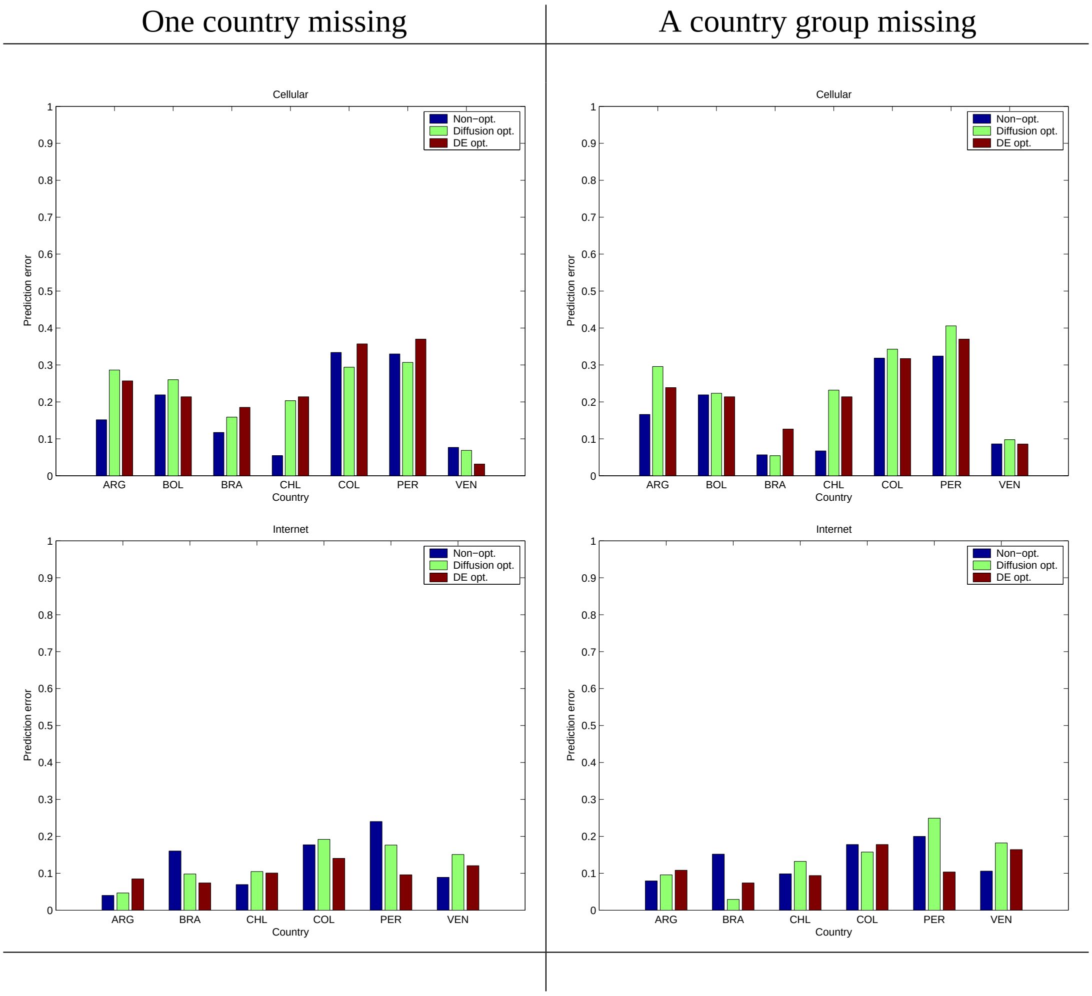

Figure 22: Prediction errors with different optimisations for all products in the South-American countries.

Table 3: Average prediction errors for products in South-America: upper results with on country missing, lower results with the whole country group missing.

Product	Non-optimised		Diffusion optimised		DE optimised	
	Average	Variance	Average	Variance	Average	Variance
Cellular	0.1833	0.0131	0.2254	0.0076	0.2327	0.0130
Internet	0.1294	0.0057	0.1281	0.0030	0.1027	0.0006
Cellular	0.1769	0.0130	0.2359	0.0160	0.2237	0.0098
Internet	0.1356	0.0023	0.1410	0.0057	0.1203	0.0017

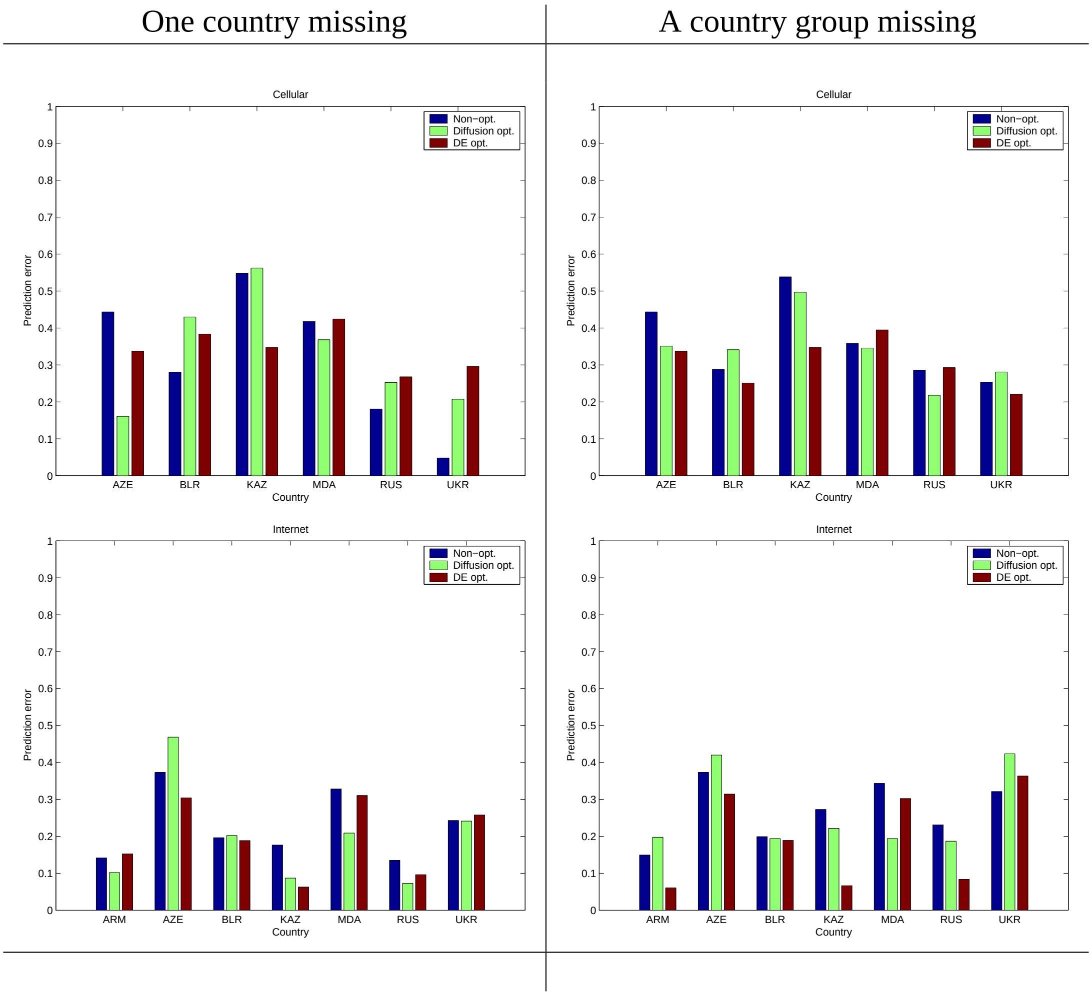

Figure 23: Prediction errors with different optimisations for all products in the CIS countries.

Table 4: Average prediction errors for products in the CIS countries: upper results with on country missing, lower results with the whole country group missing.

Product	Non-optimised		Diffusion optimised		DE optimised	
	Average	Variance	Average	Variance	Average	Variance
Cellular	0.3198	0.0344	0.3301	0.0230	0.3428	0.0032
Internet	0.2277	0.0085	0.1975	0.0187	0.1962	0.0097
Cellular	0.3612	0.0122	0.3389	0.0086	0.3073	0.0042
Internet	0.2700	0.0066	0.2625	0.0120	0.1971	0.0169

One country missing

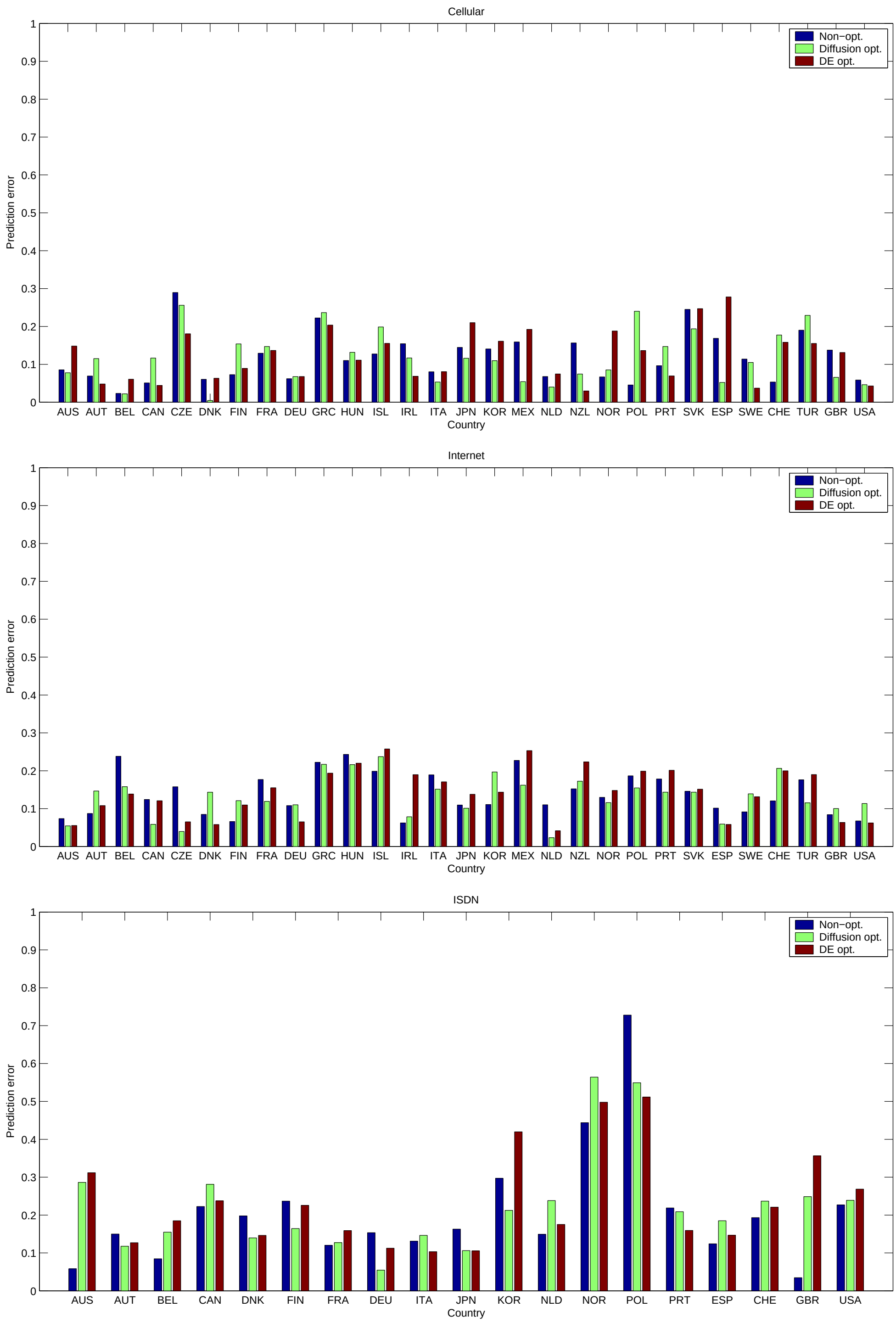

Figure 24: Prediction errors with different optimisations for all products in the OECD countries.

Table 5: Average prediction errors for products in the OECD countries.

Product	Non-optimised		Diffusion optimised		DE optimised	
	Average	Variance	Average	Variance	Average	Variance
Cellular	0.1167	0.0041	0.1184	0.0048	0.1231	0.0046
Internet	0.1388	0.0031	0.1310	0.0030	0.1418	0.0041
ISDN	0.2070	0.0242	0.2243	0.0176	0.2352	0.0164

This is very significant. The most important use of the method is to predict diffusion when products are marketed in new marketplaces, and generally product roll-outs are done to geographically neighbouring countries. This means that in a country group, e.g., South-America, the countries get a new product at almost the same time, and the prediction has to be based on countries outside of the group. The method seems to be applicable in real world because this appears to work.

Weights that the differential evolution based SOM optimisation method found for different products are shown in Table 6. It has to be noted that these results were gained from a single optimisation run with DE per product, so no final conclusions of the importance of parameters should be made even. However, some observations of weights of parameters for all products follow.

For the diffusion of cellular phones the large importance of population was quite logical, since many of the worlds most populated countries are lagging behind. Also, the high importance of ethnical groups was as excepted, since it has been noted before that countries with high internal diversity are slower at accepting new innovations. Risk-rating, which measures perceived instability of a country, has unsurprisingly quite a high importance. High weight of HDI (Human Development Index) might have been understandable, but because there was HDI parameters from two years and the weight for other year was very low, the high weight for one of them might be considered as noise. Besides, HDI is a variable which is homogeneous inside country groups that are formed by many other variables (i.e., at the same development level), so the weight of HDI might not actually matter very much. Inflation might be another variable like HDI. It is very stable among developed countries. Again, the importance of GDP was very understandable, rich countries are leading the diffusion. The high importance of personal computers per capita and telecom investments seemed quite logical, since advancement in many technical products

Table 6: Weights found for different parameters of the countries for different products with the differential evolution based SOM optimisation method. The weights are absolute, whether the effect of a specific parameter is negative or positive is not known.

Parameter	Cellular phones	Internet users	ISDN subscribers
Population	0.8588	0.9105	0.1425
Population growth	0.3027	0.0957	0.8811
Ethnical groups	0.7189	0.4236	0.6632
Languages	0.3311	0.3982	0.9882
Urbanisation	0.0598	0.4914	0.9891
Risk-rating	0.6292	0.4914	0.4139
HDI	0.0602	0.7903	0.5816
HDI from 1990	0.9464	0.6683	0.2403
TAI	0.3453	0.0362	0.2811
GDP per capita in USD	0.8049	0.4870	0.3241
GDP per capita with PPP	0.9317	0.6354	0.3535
Inflation	0.8188	0.4943	0.7249
Telecom investments	0.6249	0.0718	0.9668
International telephone traffic	0.2118	0.2806	0.2387
Personal computers per capita	0.9978	0.5603	0.8998
Year of starting mobile comm.	0.1303	0.3375	0.8931

goes hand in hand, and telecom investments are a pre-requisite to a wide deployment of cellular phones. What seems quite surprising, is the low weight of year of starting mobile communications.

For the Internet users population had again a high weight; however, in this case there does not seem to be as clear explanation as with cellular phones. The largest countries, India and China, have a low number of the Internet users, but below that there appears not to be very clear correlation between population and the Internet usage. HDI from both years had a high weight this time, these figures might be helping to group countries in a more fine-grained manner since the Internet usage has spread faster to less developed countries than cellular phones. Again, GDP had understandably a quite high weight. A low weight of telecom investments and not a very high weight of personal computers – one could think that a personal computer is a requirement to using Internet – might be explained by the fact that Internet cafes have spread very fast even to less developed countries, making it possible to use Internet without very up-to-date telecom infrastructure and a high number of personal computers.

ISDN is the least interesting product, even more so in this case because it is available only in the most developed countries which tend to have similar economical and social parameters. Still, few understandably important parameters were found: high telecom investment is a requirement for availability of ISDN, since there have to be lots of personal computers, and high urbanisation helps deploying ISDN fast to a large part of the population. Then again, there were a couple of parameters whose correlation to diffusion of ISDN seems quite dubious: population growth and the number of used languages. However, these might be explained by the fact mentioned earlier; those parameters are probably nearly same for most of the countries having ISDN available.

For the most part the weights found by the DE based optimisation were sensible. However, the DE based optimisation did not lead to significantly better prediction performance. Either the information revealed by the DE about the weights of parameters is correct but is not descriptive enough to help creation of the country map, or there is too much noise in the found weights or in the overall process of SOM training so that using the weights is not of any help. The weights seemed to make at least some sense, so the reason for not getting better results when using them might be the latter: randomness caused by SOM training. So, while the DE based SOM optimisation method does not seem to be useful for this task, the weights found might be of interest for other uses, if their validity can be confirmed for example by repeating the tests several times.

6 Conclusion

In this master's thesis a method for predicting the diffusion of an innovation in certain circumstances was developed. The method is designed to work when the product has already been sold in some countries or the diffusion of some similar product is known. Previously, cross-national diffusion has been researched and some attributes that affect diffusion, or divide countries to separate classes from a point of view of diffusion of (technological) innovations, have been found. This has been broadened to work automatically with multiple parameters. The method successfully creates ordering for countries taking into account the known diffusion of the product. Experiments gave promising results.

The development of the method should not end here because to be usable in the real world it should be combined with other methods to create a working system. One choice would be to create an expert system which uses various prediction methods together with knowledge and insight of the user.

Automating all the steps of the prediction while covering all cases would need many different methods, such as a method for finding similar products. Finding similar products would need collected information of all the products. All needed information is not easily given as numerical values, for example, intended target audience and pricing policy, and these parameters are quite easily conveyed compared to the use of the product, e.g., is it a cellular phone or dish washing machine. Complete automation is not a very sensible goal.

When complete automation is not sought after, some compromises can be made. Creating initial prediction for a new product could be done by basing it to other similar products, where similar products have been defined by the user of the system. Other prediction methods (neural or based on the diffusion models) could be used to predict following years when there is some diffusion data already available. Predictions with different methods then could be combined to create the final prediction, or possibly give a more complex prediction which gives details of the predictions of the different prediction methods. That would give to an user more direct feedback whether the different methods agree on the future diffusion, and an user could make his/her own conclusions. The similar methodology is used when doing weather forecast: the user – meteorologist in this case – is able to override forecast given by the computer program in case it is wrong in the opinion of the highly educated user.

After all, predicting the diffusion of an innovation is mostly useful for managers making decisions of when to enter a marketplace and what kind of volumes of the product are needed. The application would not need to be usable by a layman, since all the real users would be professionals and have knowledge of the area themselves. Because of this the system does not need to be very user-friendly or simple and some of the burden of making final predictions can be left to the user.

References

[1] F. Bass. A new-product growth model for consumer durables. *Management Science*, 15:215–227, January 1969.

[2] T. Kohonen. *Self Organizing Maps*. Springer Series in Information Sciences. Springer-Verlag Berling Heidelberg, 1995. ISBN 3-540-58600-8.

[3] V. Mahajan, E. Mueller, and Y. Wind. *New-Product Diffusion Models*. International Series in Quantitavive Marketing. Kluwer Academic Publishers, 2000. ISBN 0-7923-7751-6.

[4] Z. Griliches. Hybrid corn: An exploration in the economics of technological change. *Econometrica*, 25(4):501–522, October 1957.

[5] V. Mahajan, E. Muller, and F. M. Bass. New product diffusion models in marketing: A review and directions for research. *Journal of Marketing*, 54:1–26, January 1990.

[6] P. M. Parker. Aggregate diffusion forecasting models in marketing: A critical review. *International Journal of Forecasting*, 10:353–380, 1994.

[7] C. J. Easingwood, V. Mahajan, and E. Muller. A nonuniform influence innovation diffusion model of new product acceptance. *Marketing Science*, 2:273–295, 1983.

[8] B. Kang, C. Han, and C.-H. Yim. An adaptive framework for forecasting demand and technological substitution. *ETRI Journal*, 18(2):87–106, July 1996.

[9] V. Kumar, J. Ganesh, and R. Echambadi. Cross national diffusion research: What do we know and how certain we are? *Journal of Product Innovation Management*, 15:255–268, 1998.

[10] M. G. Dekimpe, P. M. Parker, and M. Sarvary. “globalization”: Modeling technology adoption timing across countries. *Technological Forecasting and Social Change*, 63:25–42, 2000.

[11] S. Haykin. *Neural Networks - A Comprehensive Foundation*, pages 397–443. Macmillan College Publishing, 1994. ISBN 0-02-352761-7.

[12] B. K. Wong, T. A. Bodnovich, and Y. Selvi. A bibliography of neural network business applications research: 1988-September 1994. *Expert Systems*, 12(3), August 1995.

[13] B. K. Wong and Y. Selvi. Neural network applications in finance: A review and analysis of literature (1990-1996). *Information & Management*, 34:129–139, 1996.

[14] T. Hill, L. Marquez, M. O'Connor, and W. Remus. Artificial neural network models for forecasting and decision making. *International Journal of Forecasting*, 10:5–15, 1994.

[15] G. Zhang, B. E. Patuwo, and M. Y. Hu. Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14:35–62, 1998.

[16] G. P. Zhang, B. E. Patuwo, and M. Y. Hu. A simulation study of artificial neural networks for nonlinear time-series forecasting. *Computers & Operations Research*, 28:381–396, 2001.

[17] T. M. Mitchell. *Machine Learning*. McGraw-Hill Series in Computer Science. McGraw-Hill, 1997. ISBN 0-07-042807-7.

[18] Movielens. Website. http://www.movielens.umn.edu/.

[19] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International World Wide Web Conference*, pages 285–295, Hong-Kong, May 2001.

[20] J. B. Schafer, J. Konstan, and J. Riedl. Electronic commerce recommender applications. *Journal of Data Mining and Knowledge Discovery*, 5(1):115–153, January 2001.

[21] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 79(9):1464–1480, 1990. ISSN 0018-9219.

[22] R. Storn and K. Price. Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical report, International Computer Science Institute, University of California, Berkeley, California, USA, March 1995. PostScript file downloadable from http://www.icsi.berkeley.edu/techreports/1995.abstracts/tr-95-012.html.

[23] R. Hecht-Nielsen. *Neurocomputing*, pages 63–75. Addison-Wesley Publishing Company, 1989. ISBN 0-201-09355-3.

[24] C. von der Malsburg. Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik*, 14:85–100, 1973.

[25] T. Kohonen. Things you haven't heard about the self-organizing map. In *IEEE International Conference on Neural Networks*, volume 3, pages 1147–1156, San Francisco, California, USA, March 1993.

[26] E. Jurva and O. Taipale. Adaptive and intelligent systems applications 1994-1999. Technical Report 18, TEKES, 2000. Available in http://www.cis.hut.fi/neuronet/Tekes/.

[27] T. Kohonen. Engineering applications of the self-organizing map. *Proceedings of the IEEE*, 84(10):1358–1384, 1996. ISSN 0018-9219.

[28] WebSOM. Website. http://websom.hut.fi/websom/.

[29] T. Kohonen, S. Kaski, and et al. K. Lagus and. Self organization of a massive document collection. *IEEE Transactions of Neural Networks, Special Issue on Neural Networks for Data Mining and Knowledge Discovery*, 11(2):574–585, 2000.

[30] PicSOM. Website. http://www.cis.hut.fi/picsom/.

[31] J. Laaksonen, Koskela M, and E. Oja. Picsom - a framework for content-based image database retrieval using self-organizing maps. In *Proceedings of 11th Scandinavian Conference on Image Analysis (SCIA'99)*, Kangerlussuaq, Greenland, June 1999.

[32] J. Lampinen. Differential evolution: New naturally parallel approach for engineering design optimization. In B.H.V. Topping, editor, *Developments in Computational Mechanics with High Performance Computing*, pages 217–228. Civil-Comp Press, Edinburgh, Scotland, 1999. ISBN 0-948749-59-8.

[33] J. T. Alander. Geneettisten algoritmien mahdollisuudet. Technical Report 59/98, TEKES, 1998. Available in ftp://ftp.uwasa.fi/cs/GA/.

[34] J. T. Alander and J. Lampinen. Shape optimization of diesel fuel injection cam by genetic algorithm. In *Proceedings of the Second Nordic Workshop on Genetic Algorithms and their Applications (2NGWA)*, pages 195–204, Vaasa, Finland, August 1996.

[35] J. Lampinen. Choosing a shape presentation method for optimization of 2d shapes by genetic algorithms. In *Proceedings of the Third Nordic Workshop on Genetic Algorithms and their Applications (3NGWA)*, pages 305–319, Helsinki, Finland, August 1997.

[36] C.S. Chang and D.Y. Xu. Differential evolution based tuning of fuzzy automatic train operation for mass rapid transit system. In *IEEE Proceedings of Electric Power Applications*, volume 147, pages 206–212, May 2000. ISSN 1350-2352.

[37] M. Rüttgers. Differential evolution: A method for optimization of real scheduling problems. Technical report, International Computer Science Institute, University of California, Berkeley, California, USA, 1997. PostScript-file downloadable from http://www.icsi.berkeley.edu/techreports/1997.

[38] K. Puumalainen. *Global diffusion of innovations in telecommunications: effect of data aggregation and market environment*. PhD thesis, Lappeenranta University of Technology, 2002.

[39] E. Lorenz. Deterministic nonperiodic flow. *Journal of Atmospheric Science*, 20:130–141, 1963.

[40] ITU Telecommunication standardization sector (ITU-T). Website. http://www.itu.int/ITU-T/.

Appendix 1. Country codes

ABW Aruba
AFG Afghanistan
AGO Angola
ALB Albania
AND Andorra
ANT Netherlands Antilles
ARE United Arab Emirates
ARG Argentina
ARM Armenia
ASM American Samoa
ATG Antigua and Barbuda
AUS Australia
AUT Austria
AZE Azerbaijan
BDI Burundi
BEL Belgium
BEN Benin
BFA Burkina Faso
BGD Bangladesh
BGR Bulgaria
BHR Bahrain
BHS Bahamas
BIH Bosnia and Herzegovina
BLR Belarus
BLZ Belize
BMU Bermuda
BOL Bolivia
BRA Brazil
BRB Barbados
BRN Brunei Darussalam
BTN Bhutan
BWA Botswana
CAF Central African Rep.
CAN Canada
CHE Switzerland
CHL Chile
CHN China
CIV Cote d'Ivoire
CMR Cameroon
COG Congo
COL Colombia
COM Comoros
CPV Cape Verde
CRI Costa Rica
CUB Cuba
CYM Cayman Islands
CYP Cyprus
CZE Czech Republic
DEU Germany
DJI Djibouti
DMA Dominica
DNK Denmark
DOM Dominican Rep.
DZA Algeria
ECU Ecuador
EGY Egypt
ERI Eritrea
ESP Spain
EST Estonia
ETH Ethiopia
FIN Finland
FJI Fiji
FRA France
FRO Faroe Islands
FSM Micronesia
GAB Gabon
GBJ Jersey
GBR United Kingdom
GEO Georgia
GHA Ghana
GIB Gibraltar
GIN Guinea
GLP Guadeloupe
GMB Gambia
GNB Guinea-Bissau
GNQ Equatorial Guinea
GRC Greece
GRD Grenada
GRL Greenland
GTM Guatemala
GUF French Guiana
GUM Guam
GUY Guyana
HKG Hongkong
HND Honduras
HRV Croatia
HTI Haiti
HUN Hungary
IDN Indonesia
IND India
IRL Ireland
IRN Iran
IRQ Iraq
ISL Iceland
ISR Israel
ITA Italy
JAM Jamaica
JOR Jordan
JPN Japan
KAZ Kazakhstan
KEN Kenya
KGZ Kyrgyzstan

KHM	Cambodia
KIR	Kiribati
KNA	Saint Kitts and Nevis
KOR	South-Korea
KWT	Kuwait
LAO	Lao P.D.R.
LBN	Lebanon
LBR	Liberia
LBY	Libya
LCA	Saint Lucia
LIE	Liechtenstein
LKA	Sri Lanka
LSO	Lesotho
LTU	Lithuania
LUX	Luxembourg
LVA	Latvia
MAC	Macau
MAR	Morocco
MDA	Moldova
MDG	Madagascar
MDV	Maldives
MEX	Mexico
MHL	Marshall Islands
MKD	T.F.Y.R. Macedonia
MLI	Mali
MLT	Malta
MMR	Myanmar
MNG	Mongolia
MOZ	Mozambique
MRT	Mauritania
MTQ	Martinique
MUS	Mauritius
MWI	Malawi
MYS	Malaysia
MYT	Mayotte
NAM	Namibia
NCL	New Caledonia
NER	Niger
NGA	Nigeria
NIC	Nicaragua
NLD	Netherlands
NOR	Norway
NPL	Nepal
NZL	New Zealand
OMN	Oman
PAK	Pakistan
PAN	Panama
PER	Peru
PHL	Philippines
PNG	Papua New Guinea
POL	Poland
PRI	Puerto Rico
PRK	North-Korea
PRT	Portugal
PRY	Paraguay
PYF	French Polynesia
QAT	Qatar
ROM	Romania
RUS	Russia
RWA	Rwanda
SAU	Saudi Arabia
SDN	Sudan
SEN	Senegal
SGP	Singapore
SLB	Solomon Islands
SLE	Sierra Leone
SLV	El Salvador
SOM	Somalia
STP	Sao Tome and Principe
SUR	Suriname
SVK	Slovak Republic
SVN	Slovenia
SWE	Sweden
SWZ	Swaziland
SYC	Seychelles
SYR	Syria
TCD	Chad
TGO	Togo
THA	Thailand
TJK	Tajikistan
TKM	Turkmenistan
TON	Tonga
TTO	Trinidad and Tobago
TUN	Tunisia
TUR	Turkey
TWN	Taiwan
TZA	Tanzania
UGA	Uganda
UKR	Ukraine
URY	Uruguay
USA	United States of America
UZB	Uzbekistan
VCT	St. Vincent and the Grenadines
VEN	Venezuela
VIR	Virgin Islands (U.S.)
VNM	Viet Nam
VUT	Vanuatu
WSM	Western Samoa
YEM	Yemen
YUG	Yugoslavia
ZAF	South Africa
ZAR	Congo
ZMB	Zambia
ZWE	Zimbabwe

Appendix 2. Groups of countries

EU	EU candidates	OECD	South-America	CIS
Austria	Bulgaria	Australia	Argentina	Armenia
Belgium	Cyprus	Austria	Bolivia	Azerbaijan
Denmark	Czech Republic	Belgium	Brazil	Belarus
Finland	Estonia	Canada	Chile	Georgia
France	Hungary	Czech Republic	Colombia	Kazakhstan
Germany	Latvia	Denmark	Ecuador	Kyrgyzstan
Greece	Lithuania	Finland	Paraguay	Moldova
Ireland	Malta	France	Peru	Russia
Italy	Poland	Germany	Uruguay	Tajikistan
Luxembourg	Romania	Greece	Venezuela	Turkmenistan
Netherlands	Slovak Republic	Hungary		Ukraine
Portugal	Slovenia	Iceland		Uzbekistan
Spain	Turkey	Ireland		
Sweden		Italy		
United Kingdom		Japan		
		Luxembourg		
		Mexico		
		Netherlands		
		New Zealand		
		Norway		
		Poland		
		Portugal		
		Republic of Korea		
		Slovakia		
		Spain		
		Sweden		
		Switzerland		
		Turkey		
		USA		
		United Kingdom		