Intellectual Capital:
From Intangible Assets to Fitness Landscapes

Brendan Kitts
Vignette Corporation,
19 Newcrossing Road, Reading, MA. 01867. USA.
Ph: (US) 781 942-3600x136, Fax: (US) 781 942-2163,
Email: bkitts@vignette.com

Leif Edvinsson
Skandia Corporation,
Skandia Future Center, Villa Askudden, PO Box 153, S-185 22 Vaxholm, Sweden.
Email: ledvinsson@skandia.com

Tord Beding
Hagatornet AB,
Haga Nygata 28, S-411 22 Gothenburg, Sweden,
Email: hagatornet@goteborg.mail.telia.com

Abstract

Intellectual Capital (IC) has been proposed by Edvinsson and Malone (1997) as a technique for quantifying a company's intangible assets. A careful analysis can result in hundreds of variables, and extracting knowledge from these measurements can be difficult. We introduce a knowledge management technique called *IC mapping* that attempts to synthesize this data into a fitness landscape. Using the map, managers can query the surrounding landscape, view past progress as a trajectory across the landscape, and calculate what parameters need to be changed to reach new locations. IC mapping provides a novel knowledge management tool for understanding, managing, and representing a company's intangible knowledge assets.

Introduction

Managing corporations is a difficult and risky business. Doing it well requires the ability to understand factors including market, customers, employees, technology, culture, history, and opportunities. Management is made all the more difficult because variables interact. For instance, increasing expenditure on information technology may decrease next year's profitability, but increase the number of projects completed on-time. Such variable interactions may be separated in time, non-causal, non-linear, and involve multiple variables.

This is where a knowledge management tool and can help. This paper introduces a knowledge management technique we call "IC mapping". IC mapping extracts knowledge from historical company data and converts them into an interactive three-dimensional landscape. Using this map, managers can interactively query the landscape, perform what-if analysis, identify problems in their business, and understand their company's performance in relation to others.

Multivariate measures of company performance

With the growth of the services sector in major industrialized countries (USBLS, 1999), many authors have suggested that non-traditional or "intangible" assets of business operations - such as customer relationships, and skills of employees may be increasingly important, and worthy of reporting on profit-loss sheets in their own right. In effect, a wider measurement net needs to be cast in order to capture the fitness of a company (Karlgaard, 1993; Kaplan and Norton, 1996; Edvinsson and Malone, 1997; Madison Valuation Associates, 1999).

Leif Edvinsson pioneered the development of diversified measures of company performance from 1990-1997 (Edvinsson and Malone, 1997). The system he devised, "Intellectual Capital" (IC), measures performance in five major areas: Financial, what appears on ordinary balance sheets; Human, the skills and experience of the employees; Customer, goodwill, relationships, and brandname; Process, which measures how efficient internal functions are; and Renewal, which measures growth and long-term research and development.

By using variables from diverse areas, Edvinsson hoped to probe for problems that would remain hidden in ordinary profit and loss balance sheets. This information could then be used to inform strategic decisions as to expenditure in different organizational areas, new investments, and business reorganization.

Top variables for predicting future earnings

1 year into the future

Rank	Variable	Type	Correlation	Rank correlation	F	number of observations
1	Operating result	F	0.7632	0.7544	17.4723	32
2	Number of contracts	C	0.4182	0.4794	4.7225	29
3	Number of contracts / employee	P	0.4642	0.4395	3.6638	19
4	Points of sale	C	0.77	0.8076	3.557	8
5	Percent managers	H	-0.9422	-0.9918	3.5511	6
6	Operating income	F	0.8034	0.8143	3.2272	7
7	Value added / employee	F	0.7022	0.7594	2.9586	8
8	Assets under management	F	0.4033	0.4888	2.6027	18
9	Change and development of existing holdings	R	0.701	0.4113	1.9656	6
10	Percent of managers who are women	H	0.457	0.4434	1.8793	11
11	Payroll costs / administrative expenses (%)	P	0.5933	0.6609	1.7602	7
12	Savings / contract (000s)	C	0.5388	0.6096	1.7421	8
13	Adm exp / gross premiums written (%)	P	-0.358	-0.3498	1.538	14
14	Return on capital employed (%)	F	0.5238	0.5479	1.3718	7
15	Developmental expense as a percentage of Gross profit	H	0.4203	0.4942	1.0599	8

2 years into the future

Rank	Variable	Type	Correlation	Rank correlation	F	number of observations
1	Operating result	F	0.6172	0.6609	6.0944	18
2	Number of contracts / employee	P	0.6717	0.8518	4.5123	12
3	Number of contracts	C	0.5135	0.5869	4.2187	18
4	Assets under management	F	0.5133	0.5834	2.3714	11
5	Increase in number of contracts (%)	R	-0.3894	-0.3053	1.8196	14
6	Adm exp / gross permiums written (%)	P	-0.4827	-0.4598	1.6308	9

3 years into the future

Rank	Variable	Type	Correlation	Rank correlation	F	number of observations
1	Number of contracts / employee	P	0.6643	0.8015	3.0894	9
2	Adm exp / gross permiums written (%)	P	-0.7027	-0.8822	2.4691	7
3	Number of contracts	C	0.4315	0.4077	1.6759	11
21*	Operating result	F	-0.112	0.0972	0.1004	10

* included for comparison

Figure 1: Top variables (in terms of strength of effect) for predicting Total Operating Result (MSEK) at Skandia companies 1, 2 and 3 years into the future. These results were generated by taking data from Skandia's Annual reports, joining together variables which were common between these companies, and then collecting together training data and operating result 1, 2 and 3 years after training data sample is taken. Past Operating Result becomes unpredictive for predicting revenue three years into the future (F=0.1, R=-0.1), but Number of Contracts Per Employee remains reliable over all years (F>3, R>0.4).

Our own detailed analysis of Skandia companies has confirmed the importance of diverse company measurements. We acquired data from Skandia over the period 1991 to 1997 and ran a correlation between variables and future earnings. The problem was to predict Total Operating Result in the future, based on observed variables for a company.

Fifty-nine variables were available, and variables were transformed into zscores within their company timeseries, to prevent any effect from the absolute operating size of each company. Results are shown in figure 1.

The best variable for predicting operating income 12 months into the future is Total Operating Result in previous year, which is a financial variable. However, as the prediction horizon extends to 2 and 3 years into the future, the predictive utility of financials decreases. At 2 and 3 years in the future, Contracts Processed Per Employee, and Value-Add Per Employee become more strongly correlated with future success than past operating income. Administrative Expenses and Percent Of Employees Who Are Managers have negative correlations with future Operating Result.

Unfortunately, even with detailed statistical analysis, extracting knowledge can still be difficult. With potentially hundreds of interacting, non-linear and co-linear IC variables, it can be difficult for a human being to grasp the overall state of an organization.

Edvinsson and Malone recognized this problem, and wrote of the need for a way to collect these variables together into a form which could be easily comprehended by a human being:

> "Somehow, then, all of these regions must be pulled together into an overall format …. what is needed in Intellectual Capital is a map that captures all of the value of an enterprise, color-coded so that one can quickly ascertain the quality of the topology – where there are swamps and lush forests, mountains and deserts." (Edvinsson and Malone, 1997, pp. 67)

The ideal method of presentation should make it possible to see such interactions, and allow human beings to quickly comprehend what was happening within the company. This article describes a new method that we believe fulfills these requirements. The technique extracts knowledge from historical IC measurements, and represents them using a knowledge map. The map shows the fitness of a company for different combinations of parameters, and supports interactive what-if analysis.

The promise of mapping

IC Mapping is designed to provide a visual tool that unifies the operating state of the company, and is easy to use. The method is based on a body of well-known statistical methods including Multi-dimensional scaling, which was pioneered by Torgerson (1958), Kruskal (1978), Shepard (1974), Young (1987; 1998), and others; and non-parametric estimation (applied widely in the field of Neural Networks), investigated by Girosi, Jones and Poggio (1993) among others.
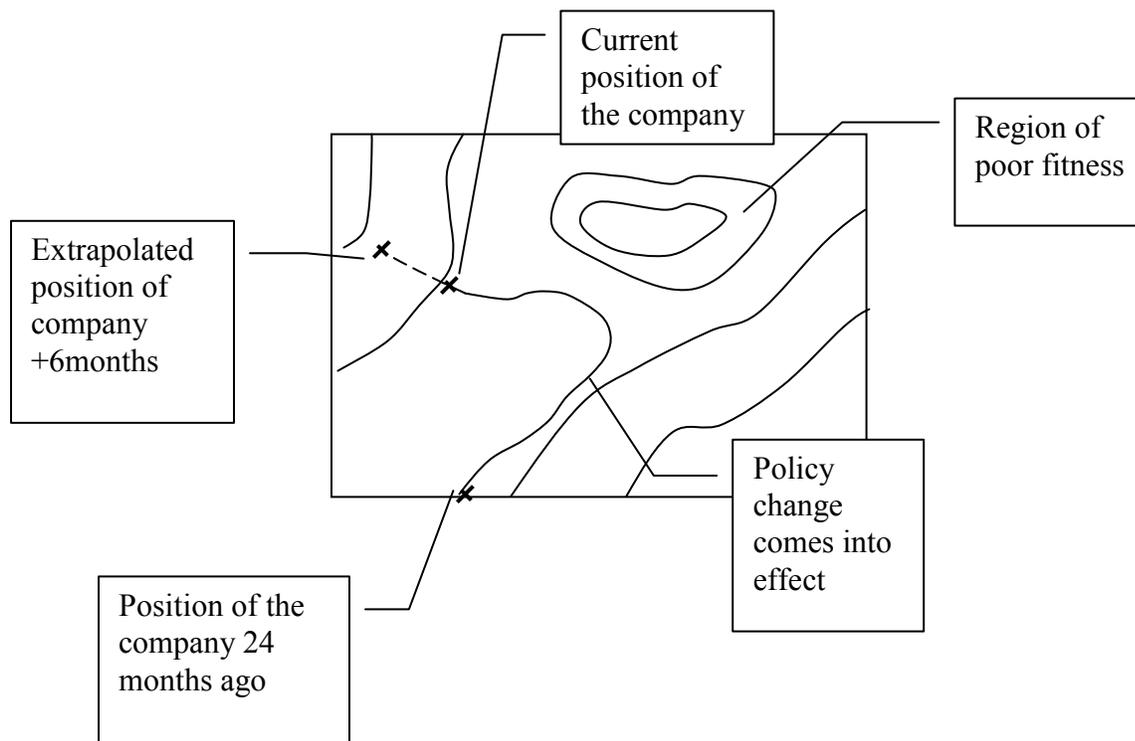
Figure 3: Viewing a company's weight timeseries allows us to plot its speed and direction of movement, and allows us to anticipate problems it will encounter in the future.

Figure 3 shows an example of an IC map, with a company traveling across a fitness landscape. Each point on the map represents a possible company state vector. At each location, the height represents the fitness of that state. This gives rise to mountains and valleys for different company states. The user can see their company in relation to other companies (and their own company's past states) on the map. In this way, the user can use those companies like navigation stars, to chart a course across the landscape.

The applications of this map are numerous.

a) Viewing Company Trajectory across landscape:

By connecting the company's positions in chronological order, we can view the historical movement of the company as a trail of points across the map.

b) Landscapes constructed using multiple companies

Multiple companies can be displayed on the same fitness landscape, allowing a company to compare itself to competitors, and get some idea of the shape of the fitness terrain far from its present location.

c) Labeling important terrain

Important points or regions, such as a company that went bankrupt, can be explicitly labeled on the map. These will become landmarks, and companies can identify if they are heading towards or away from them.

d) Predicting future location and avoiding poor regions of fitness

By extrapolating the trajectory of the company, for instance, drawing a line through the last few points, we can also estimate where the company will be in the next period of time. This may enable decision makers to recognize that their company is moving towards a bankruptcy several years before reaching that condition.

e) Time to reach destination

The time to reach a new destination can be estimated by using historical parameter change speed as the basis for predicting time to reach new points.

f) <u>What-if analysis</u>

We can interactively test the outcome of making changes to one or more variables, for instance, doubling the administration budget. After a change is made, the updated position of the company can be displayed on the landscape.

g) <u>Means-ends analysis</u>

We can look for desirable regions, and find out the variable settings that underlie that point, and are required to reach that location. Means-ends analysis involves building an inverse model mapping 2D surface position back to high-dimensional IC vector.

h) <u>Path planning</u>

Optimal trajectories can be calculated by finding a path that maximizes fitness between source and destination, constrained by time allowed to reach destination; in other words, maximizing the path integral, bounded by a particular time allowed. The path planning model can be made more elaborate by estimating "currents" or natural interactions between variables which will tend to move parameters in particular directions. Exploitation of drift currents might enable positions to be reached faster at lower cost.

Building a Map

The mapping process consists of two stages. The first task is to project the high dimensional company data into two dimensions. The second is to add a fitness variable as a third dimension,

and then interpolate between those points to predict the shape of the fitness surface connecting these regions.

Step 1: Multidimensional scaling

The objective of MDS is to represent high dimensional data in 1-3 dimensions so that a human being can visually understand the data. Because its not possible to directly visualize more than three spatial dimensions, the method focuses instead on preserving the inter-point distances of the high-dimensional data. MDS therefore finds a set of points in two dimensions which have distances as close as possible to the inter-point distances in the high dimensional space. If the method succeeds, then a human looking at the points will be able to correctly judge that their present position is "very different from" some other point on the map, and that judgement will also hold in the difference between the two high-dimensional vectors. (Young, 1998; Young and Hamer, 1987; Norusis, 1997)

It should be noted that the axes of the new, low-dimensional space are not intuitively interpretable since each is constructed out of convenience to retain the distance relationships. The only features that make sense in the new landscape are the concepts of "more similar to-" or "more different from-" x, where x is a known case. Thus, this is somewhat analogous to a paper topographic map, where landmarks are shown (the other data points), and the user can see how distant these landmarks are to his or her position.

Algorithms for Multi-dimensional Scaling

To generate the new coordinates one has to minimize the error between the distance between points in the orignal data $\mathbf{d_{ij}}$, and in the 2D representation ∂_{ij}. This "error" is known as Stress, and was introduced by Kruskal and Shepard (Kruskal, 1978; Shepard, 1974):

$$\mathbf{Stress(d,\partial)} = \sum_{i}^{n} \sum_{j}^{n} \frac{(\mathbf{d_{ij}} - \partial_{ij})^2}{\partial_{ij}^{2}} \qquad (1)$$

Many different algorithms can be used to minimize stress (Kohonen, 1996; Young, 1987). In our own work we have found that the Torgerson Projection (Torgerson, 1958) followed by several iterations of a Nelder-Mead simplex optimization (Nelder and Mead, 1965; Betteridge, et. al, 1985) can generate very good results. Other empirical comparisons of MDS algorithms can be found in Li et. al. (1993, 1995) and Duch and Naud (1998a, 1998b).

Torgerson's Classic metric MDS algorithm

Torgerson showed in the 1950s that if a distance matrix was double centered such that, given a raw distance matrix $\mathbf{B}$, the double-centered matrix $\mathbf{D}$ is defined as

$$\mathbf{d_{ij}} = -0.5 \ (\mathbf{b_{ij}}^2 - \mathbf{b_{i.}}^2 - \mathbf{b_{.j}}^2 + \mathbf{b_{..}}^2) \qquad (2)$$

then the following relationship held:

$$\mathbf{D} = \mathbf{XX^T} \qquad (3)$$

This meant that any distance matrix could be changed into a coordinate matrix by taking a matrix square root of **D**. To derive a set of coordinates with less than the original number of dimensions, singular value decomposition could be used to identify the two largest eigenvalues, and then reconstruct coordinates with only the two largest eigenvalues. This will result in coordinates which capture as much of the variance as possible in the selection of coordinates.

$$\mathbf{U\Sigma U^T = D} \tag{4}$$

$$\mathbf{X = U\Sigma^{1/2}} \tag{5}$$

Self-organizing map

Self-organizing maps are a new technique originally formulated as a model of human visual cortex (von der Malsburg, 1973). The self-organizing map consists of an array of neurons which each store a prototype of the input they most prefer (their tuning curve). As input comes in, the cell with the prototype vector most similar to the input "wins", and adjusts its prototype to be more similar to the input. Each cell also has a "neighborhood" of other cells. This can be seen in the figures below (figure 4 and 9) and as a mesh connecting the cells. When all the cells start off, they are in a perfect grid. Over time this grid will deform to "map" the input.

Whenever one neuron adjusts its vector, the other cells in the neighborhood also have their prototypes adjusted in exactly the same direction, but to a smaller degree. The result is that neighboring cells represent similar inputs. If we then "read off" what each cell represents, we will find that input vectors have migrated to different regions of the self-organizing map. These

different regions are the low-dimensional manifestation of the high-dimensional data. Other details of the Self-organizing map can be found in Kohonen (1996).
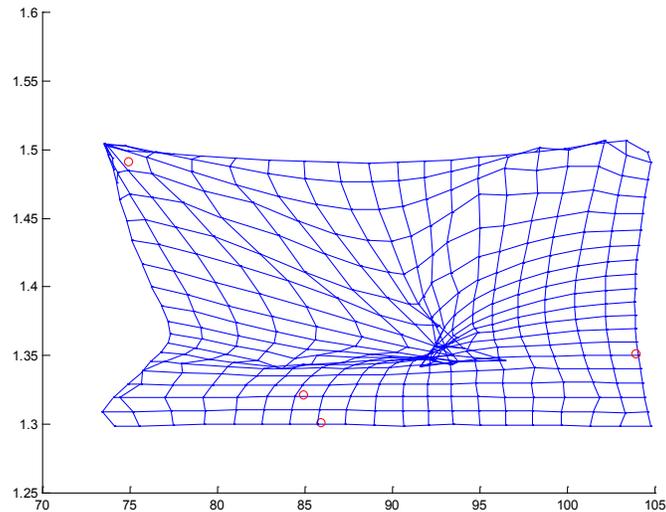
Figure 4: Top-view of the Kohonen self-organizing map, as it attempts to map a higher dimensional object. Areas of "compression" in the map represent regions where high concentrations of datapoints exist. High concentrations of points encourage centroids in the Kohonen map to "crowd in" try to to map those denser regions of points.

The equations for a Kohonen net are as follows:

Given a learning constant $1 > \alpha > 0$, a $c \times v$ matrix of centroids which is initially random, the centroids are changed as follows, upon presentation of **x** to the network:

$$\frac{\partial \mathbf{c}_{n \in N(min)}}{dt} = \alpha \mathbf{F}\left(\left|\mathbf{c_n} - \mathbf{c_{min}}\right|\right)\left(\mathbf{x} - \mathbf{c_n}\right) \tag{6}$$

$$\mathbf{c_{min}} = \mathbf{c_i} \bullet \min\left|\mathbf{c_i} - \mathbf{x}\right| \tag{7}$$

$$\mathbf{grid(i)} = \left(\left\lfloor \frac{\mathbf{i}}{\mathbf{gsize}} \right\rfloor, \mathbf{i} - \mathbf{gsize}\left\lfloor \frac{\mathbf{i}}{\mathbf{gsize}} \right\rfloor\right) \tag{8}$$

$$\mathbf{F(d)} = \begin{cases} 1 - \dfrac{\mathbf{d}}{\mathbf{nsize}} & ,\mathbf{d} < \mathbf{nsize} \\ \mathbf{0, otherwise} \end{cases} \tag{9}$$

$$\mathbf{j} \in \mathbf{N(i)} \leftrightarrow \left|\mathbf{grid(i)} - \mathbf{grid(j)}\right| < \mathbf{nsize} \tag{10}$$

where **nsize** is neighborhood size, **gsize** is gridsize, The first equation states that a neighbor $\mathbf{c_n}$ of the closest centroid $\mathbf{c_{min}}$, is moved towards the input **x** at a rate equal to a function of its distance on the mesh to $\mathbf{c_{min}}$. This speed function, **F**, is linear with decreasing activation from distance 0 from $\mathbf{c_{min}}$ to 0 at distance **nsize**. Thus, the closest centroid moves fastest towards the input.

$N(\mathbf{c_{min}})$ is a set of neighbors of $\mathbf{c_{min}}$. In this implementation neighbors form a square grid around $\mathbf{c_{min}}$, where their grid coordinates are given by **grid(.)**. In addition to the learning algorithm above, the SOM also undergoes annealing of the neighborhood size **nsize**, and learning rate α. These details can be found in Kohonen (1996). Typical parameters used in our implementation were α=0.005, **gsize**=10, **nsize_init**=**gsize**/2.

Comparison of map quality for MDS methods

Duch and Naud (1998a, 1998b) developed a novel test set to evaluate the quality of Multi-dimensional scaling methods. They proposed applying MDS to n-dimensional equilateral simplexes. These simplexes are easy to build, since their distance matrix is simply all 1s. When these simplexes are reconstructed in two dimensions using MDS, intricate symmetrical geometric representations are created. The quality of the MDS method can be visually checked by merely looking at the symmetry of the reconstructed shape. MDS results using Duch and Naud's test set are shown in figure 5 and 6.
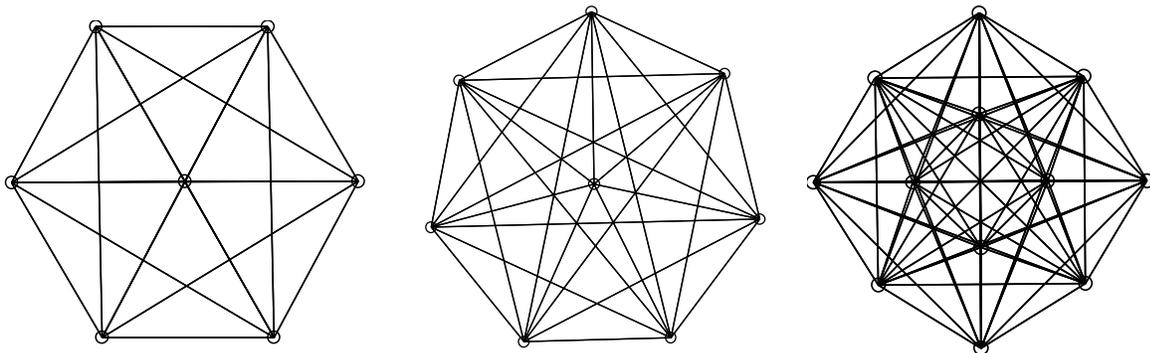
Figure 5: 7-dimensional simplex, 8-dimensional simplex, stress=3.610143. 12-dimensional simplex, stress 5.3911, generated using Nelder-Mead optimization with 10,000 iterations

Torgerson's method (figure 6 right) was the poorest on Duch and Naud's test set. Torgerson's method works by recovering coordinates from the distance matrix, and then finding the principal components of the coordinates, and projecting the coordinates onto these principal components. The resulting 2 dimensional coordinates capture as much of the variance as possible. However, all of the distances between each vertex are equal to 1. Therefore, in this particular application all

of the dimensions are equally important, and taking the principal components just results in loosing **d**-2 important dimensions. Self-organizing maps (figure 6 middle) fair better, but are still twice the stress of Nelder-mead MDS (figure 6 left). Nelder-mead optimization generates the best results by far.
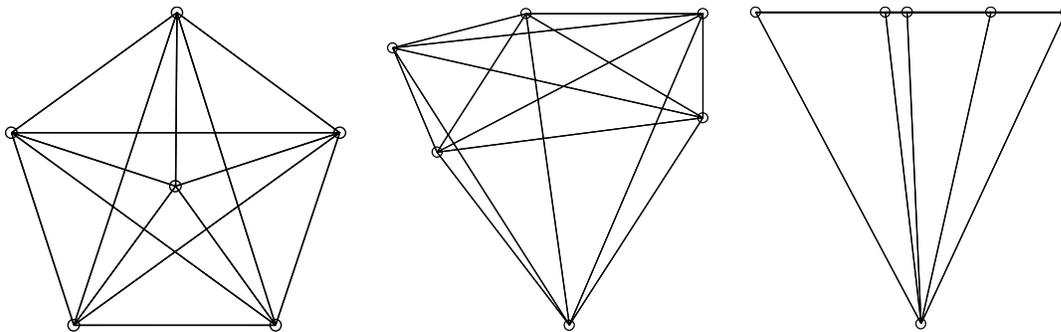
Figure 6: 6D simplexes represented in two-dimensions using three different algorithms. **(left)** MDS stress minimization, Iterations = 10,000, Stress = 2.786404, **(middle)** SOM 10x10 grid, Stress = 5.078828, **(right)** Torgerson, stress = 20.9641

Step 2: Fitness as the Third dimension

The most important feature of the landscape we want to plot is *fitness*. Fitness refers to the health of the company, and examples can include "total operating income", or "net profit after tax". Because knowing the fitness of the company at a given position is so important, we carry fitness into the 2D projection unaltered, and have it plotted as the third dimension.

After the above step we have a set of 3D points. We can think of these 3D points as a scaffold for the landscape that we want to build (figure 7,8,10 shows an example for one company Skandiabanken). Unfortunately, this scaffold is still difficult to interpret. To infer the appearance of a lanscape, we need a model which predicts fitness at any point on the 2D map, and so infers the topographic surface around the known points. To generate this, we can use function estimation techniques such as neural networks, splines, and regression to fill in away from the observed datapoints.

We have found that spline estimators (Karur and Ramachandran, 1995; Girosi, Jones and Poggio, 1993) give the best results in generating our surfaces. Unlike polynomials, splines fit surfaces around local knot points, allowing them to model local surface features.

Splines

Splines are a non-parametric regression technique which approximate a function by (a) finding a set of high-density "centroids" in the function (b) projecting all data onto a new coordinate system where each axis is a centroid, and the value on the axis is the distance from this data point to that centroid, (c) performing a least squares mapping from the new points to the target.

The spline model is defined as follows: Let **C** be a *c×2* basis matrix of centroid vectors, sometimes called knot points. Let **S** be a *r×c* basis-transformed input matrix, and let **W** be a *c×1* matrix of weights. Given an *r×2* data matrix **X**, a spline approximates a function by applying the following formula:

$$\mathbf{S \cdot W} \tag{13}$$

where

$$\mathbf{S = G(D_{XC})} \tag{14}$$

S is a representation of the input which has been transformed by the basis function given by **G(.)**. **G** is usually one of the radial functions given below (Karur and Ramachandran, 1995). The Gaussian spline has come to be known as a "Radial Basis Function network" in the neural networks literature (Girosi, Jones, and Poggio, 1993; Orr, 1996), however all of the functions below are actually radial functions, so 'RBF' is a slight misnomer.

Radial Basis net (Gaussian spline)

$$\mathbf{G(D_{XC}) = e}^{-\left(\frac{\mathbf{D_{XC}}^2}{\sigma}\right)} \tag{15}$$

Thin-plate spline

$$\tag{16}$$

$$\mathbf{G(D_{XC}) = D_{XC}}^2 \mathbf{log(D_{XC} + 1)}$$

Cubic

$$\tag{17}$$

$$\mathbf{G(D_{XC}) = 1 + D_{XC}}^3$$

Multi-quadratic

$$\tag{18}$$

$$G(\mathbf{D}_{\mathbf{XC}}) = \left(\beta^2 + \mathbf{D}_{\mathbf{XC}}^2\right)^{\frac{n}{2}}$$

Linear spline

$$G(\mathbf{D}_{\mathbf{XC}}) = 1 + \mathbf{D}_{\mathbf{XC}} \tag{19}$$

^{2}, **log** and **e** operate over the individual elements of the matrix. The $\mathbf{D}_{\mathbf{XC}}$ term is the distance between each row of **X** and the centroid points, **C**. Therefore, **S** can be thought of as a dot product between the input and the bases, normalized by the size of both **X** and **C**, and then put through a non-linear transform.

$$\mathbf{D}_{\mathbf{XC}} = -2\mathbf{C} \cdot \mathbf{X}^{\mathbf{T}} + \mathbf{C}^{\mathbf{T}}\mathbf{C} + \mathbf{X}^{\mathbf{T}}\mathbf{X} \tag{20}$$

We also ensured that all activations were normalized such that

$$\forall \mathbf{r} \sum_{v=1}^{V} \mathbf{S}_{\mathbf{rv}} = 1 \tag{21}$$

Where **V** is the number of centroids. Normalization meant that extrapolation outside the range of datapoints did not result in overly large or small values. Instead, the surface far from the points converges to the average of the datapoint values. Given the small number of datapoints, we also set the rows of **C** to be equal to the existing datapoints, so each centroid was a datapoint.

Examples of linear, cubic, and thin-plate spline surface approximation for Skandiabanken's fitness landscape are shown in figures 11, 12, 13 and 14.

Adding details to the map

Once the final landscape is built, a variety of details can be added to the basic map. Some observations may coincide with an important event, for instance "competitor came onto the market", "company went public", and so on. We can carry these labels from our high dimensional data into our 2D map, and display them on the map. This can help the user navigate the map.

Finally if the trajectory across the landscape is itself observed once every 12 months, the points in-between are not known. Since linearly connecting these points would be assuming a linear interpolation, we have connected points using smoothing splines to convey less certainty as to the nature of the points in-between the observations.

Time to reach destination

In the early days of navigation, ship captains used calipers to measure the distance between locations on their map. They would measure speed by dropping buoys and using a stopwatch to calculate speed across water, a process known as "heaving the log" (Bowditch, 1826). Using these tools, ships could estimate the time to reach their destination.

On IC Maps the same calculation can be performed. A simple method is to compare the parameter difference needed to reach the new location, with the company's historical parameter change speed per year:

ΔT(C,a,b) = Absolute parameter change /

Average Parameter change per year

$$\Delta T(C, a, b) = \frac{\sum_i \left| a_i - b_i \right|}{\frac{1}{T-1} \sum_{t=1}^{T-1} \sum_i \left| x^C_{t+1, i} - x^C_{t, i} \right|}$$

where **a** and **b** are the points we want to travel between, **C** is the company which will be travelling, x_{ti} is a historical company value for variable **i** at time **t**, and **T** is the total number of historical observations.

This assumes that a company is able to change each of its parameters equally well, and is limited by an inherent magnitude of change per year. Other methods for estimating time can also be used; for instance it is also possible to take into account historical interactions between the IC variables to estimate "currents" that might increase or decrease time to reach destination.
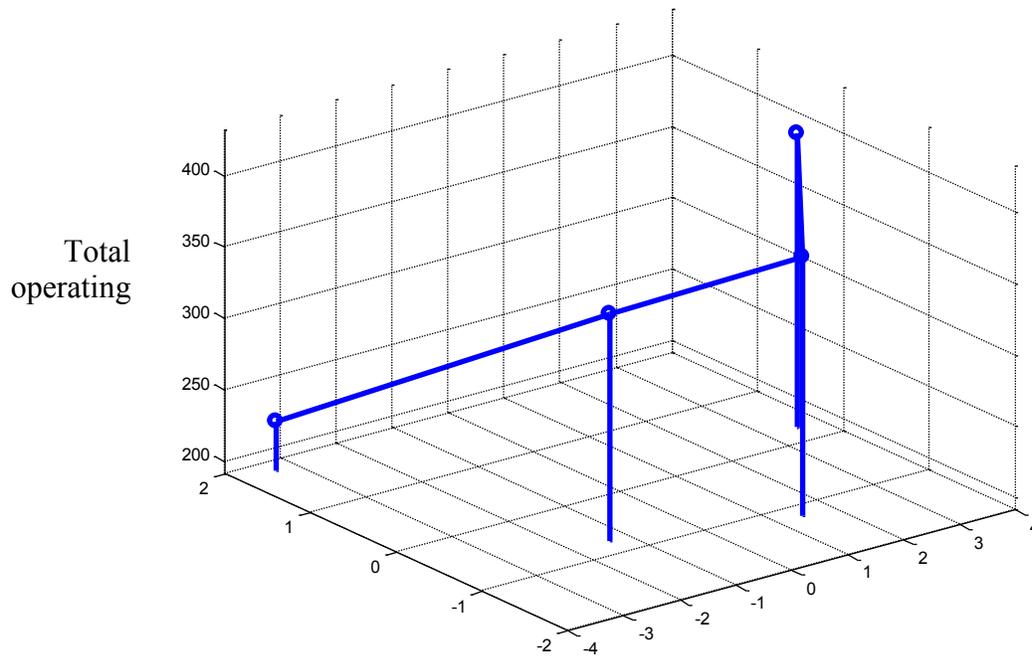
Figure 7: Scaffold for Skandiabanken's low-dimensional fitness landscape generated using Nelder-Mead stress optimization (Stress = 2.001988). Height represents total operating income for years 1994-1997, and 2D position represents 11-dimensional state.
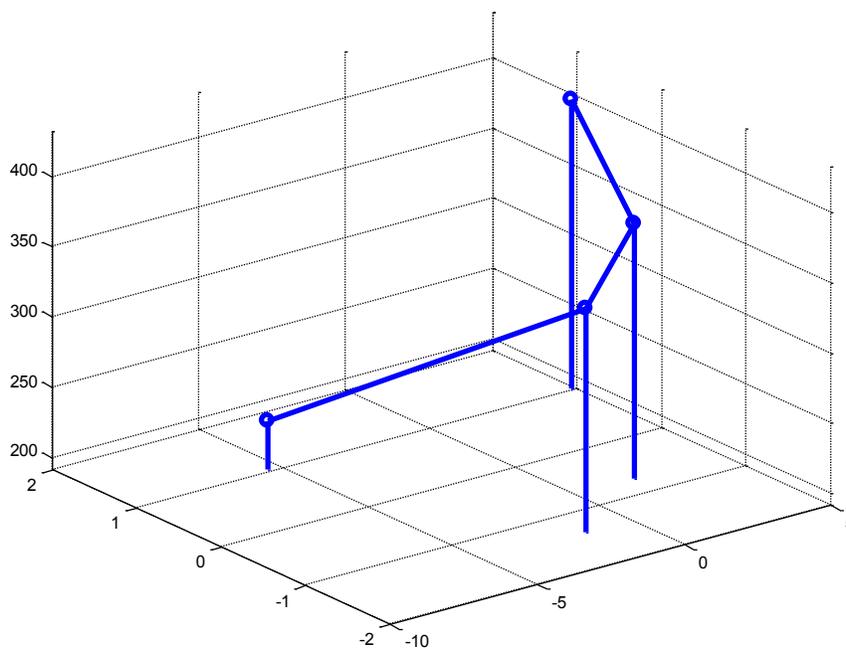
Figure 8: Scaffold for Skandiabanken data, Torgerson MDS (Stress = 42.18)
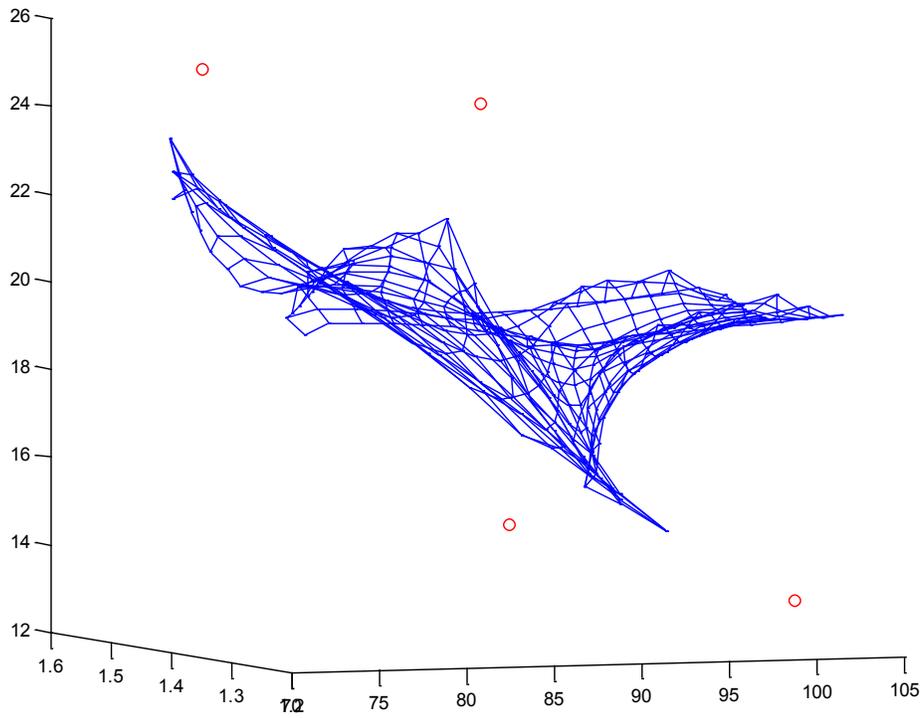
Figure 9: Side view of the Kohonen net, showing how it tries to "reach" out to the different points, constrained by its 2D mesh topology.
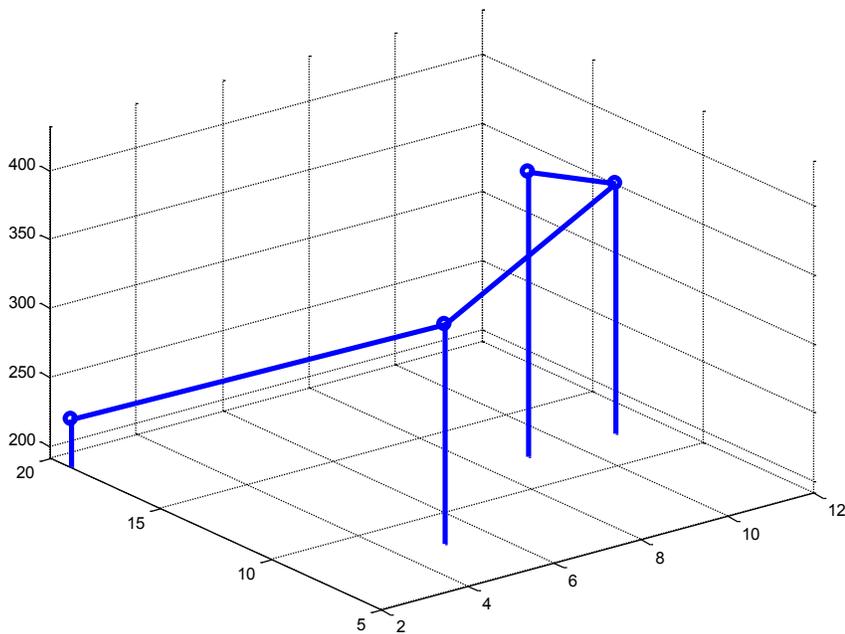
Figure 10: Scaffold for Skandiabanken generated by Self-organizing map, (Stress = 3.59)
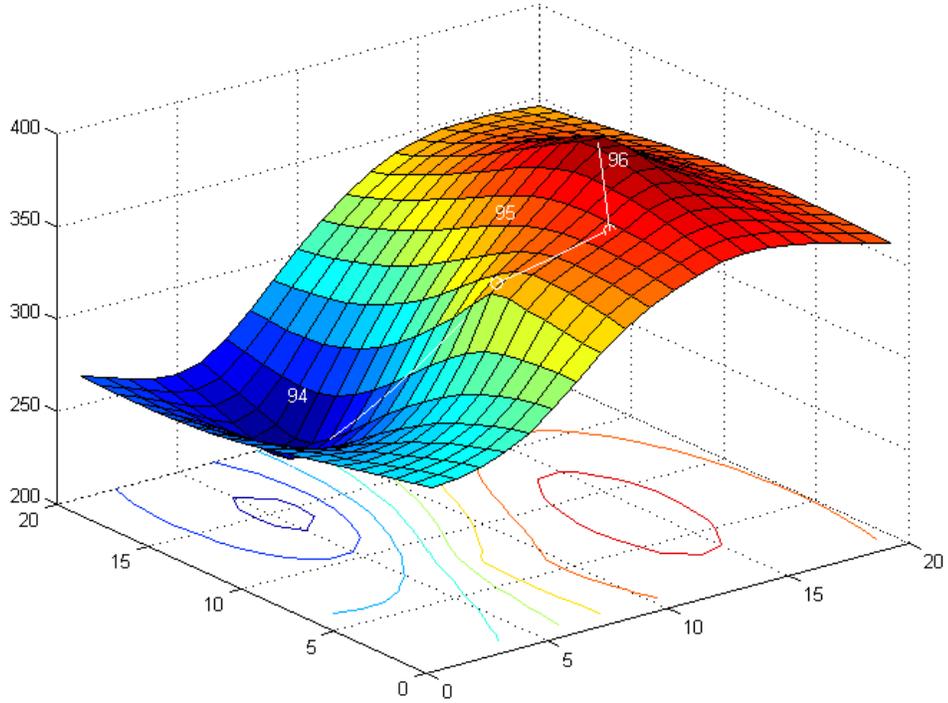
Figure 11 and 12: (top) Linear spline and **(bottom)** cubic spline approximation to SkandiaBanken's surface

thin plate spline | stddev 56 | nd modelerror 25 | 2d stress 2.001988 | 2d modelerror 172

thin plate spline | stddev 56 | nd modelerror 172 | 2d stress 2.001988 | 2d modelerror 125
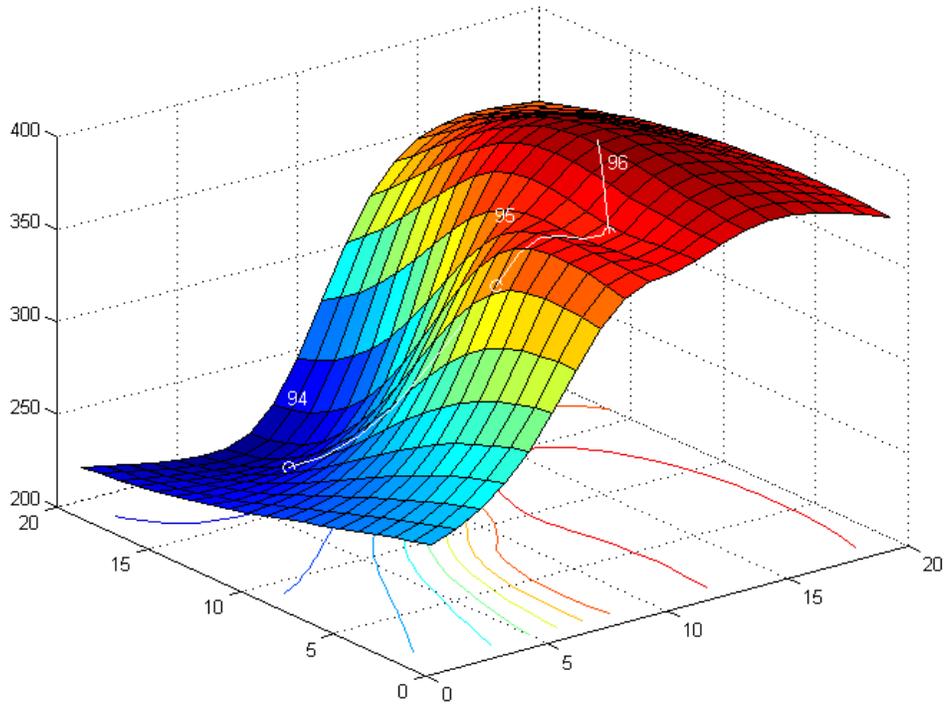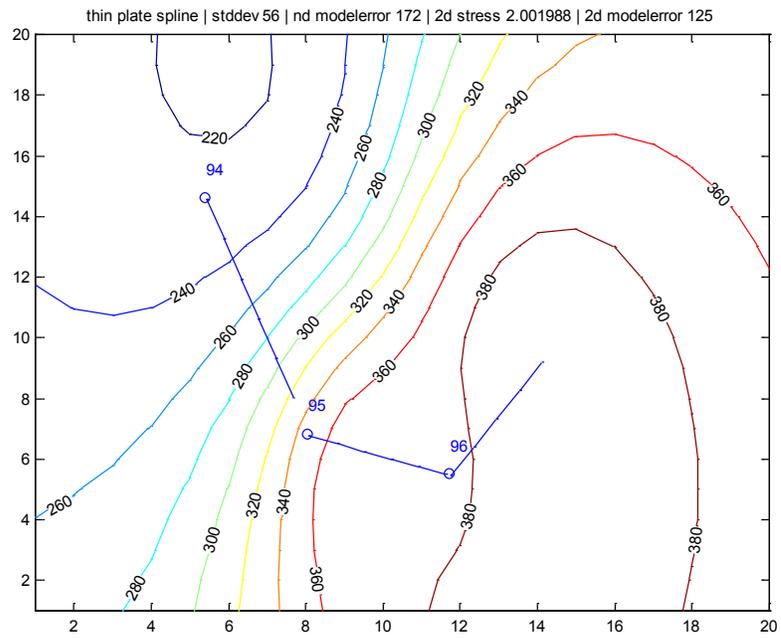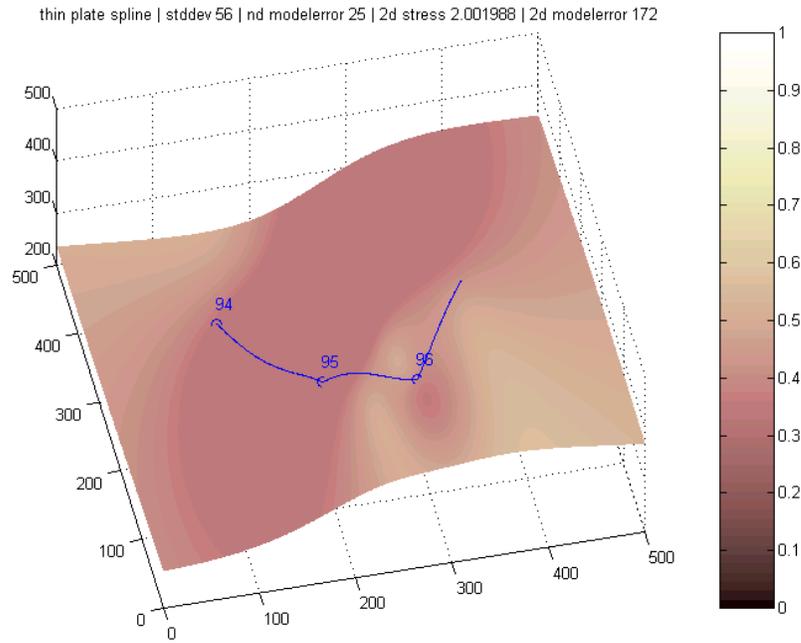
Figure 13 and 14: Different methods for showing the landscape.

Experiment 1: Skandia Fitness Landscape

In order to test whether IC Mapping could be used for real companies, we selected four companies from the large multinational Skandia investment group to be projected together onto a fitness map. The companies were Intercaser, Dial, American, and UK Life. Intercaser, American, and UK Life are the Spanish, American, and UK branches of Skandia Investment firm. Dial is a telemarketing insurance company which operates throughout Nordic countries.

The state of each company was fixed with a vector of five variables - one from each IC focus area. The five variables used were total operating result, number of customers, number of employees, contracts processed per employee, and % new contracts. Missing values were estimated using stepwise timeseries imputation. The variables were transformed into Z-scores. Ideally the fitness metric for the companies should be predictive of future earnings potential, and we could have developed a regression equation for this purpos. However, for the purposes of the test, we decided to let fitness be a linear standardized sum of the IC focus areas.

The multi-dimensional scaling was performed using a Torgerson projection (Torgerson, 1958), and fine-tuned via a Nelder-Mead simplex optimization. Optimizations took a couple of hours on a 200 MHz Pentium computer, and resulted in a stress of approximately 17.32 with correlation between original and projected distances equal to 0.91 (figures 19, 20). Tests using other methods achieved similar stresses and 2D point

positions, lending confidence to the landscape produced (figures 15, 16 shows scaffolds generated using different methods).

Kruskal non-monotonic MDS scaffold of points

Torgerson/Nelder-Mead scaffold of points
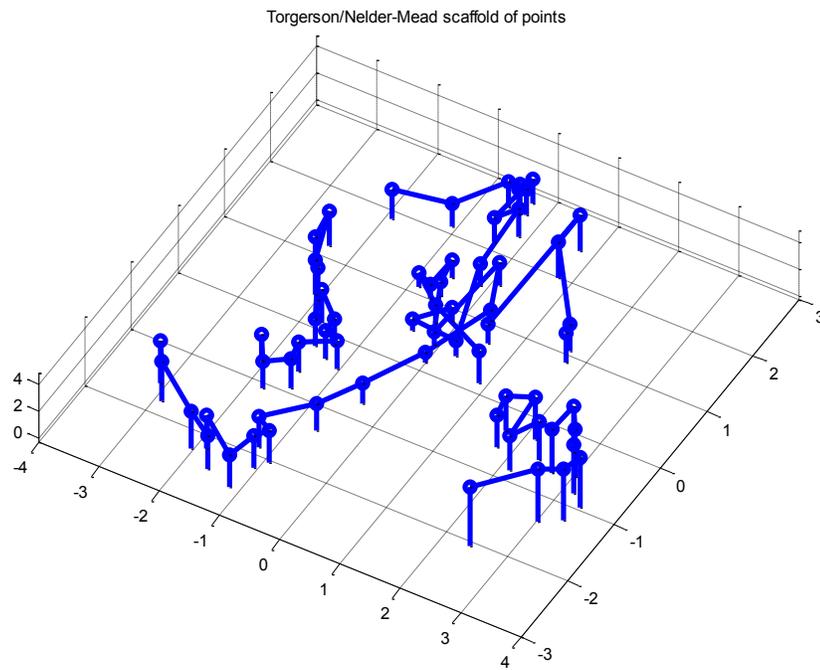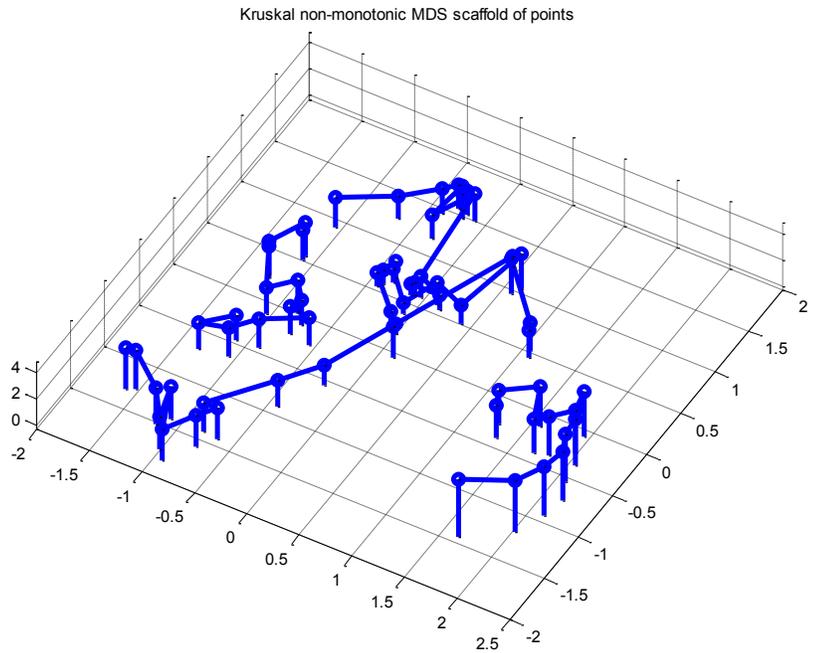
Figure 15 and 16: Scaffold of points for the Multiple company Skandia map

The final Skandia map is shown in figures 17 and 18. The map shows American Skandia rapidly zigzagging across the landscape, and increasing its Intellectual Capital. Intercaser is also doing well, and is headed out of the map area, while Link is maintaining steady fitness. UK Life, however, is losing Intellectual Capital and slipping down the slope.
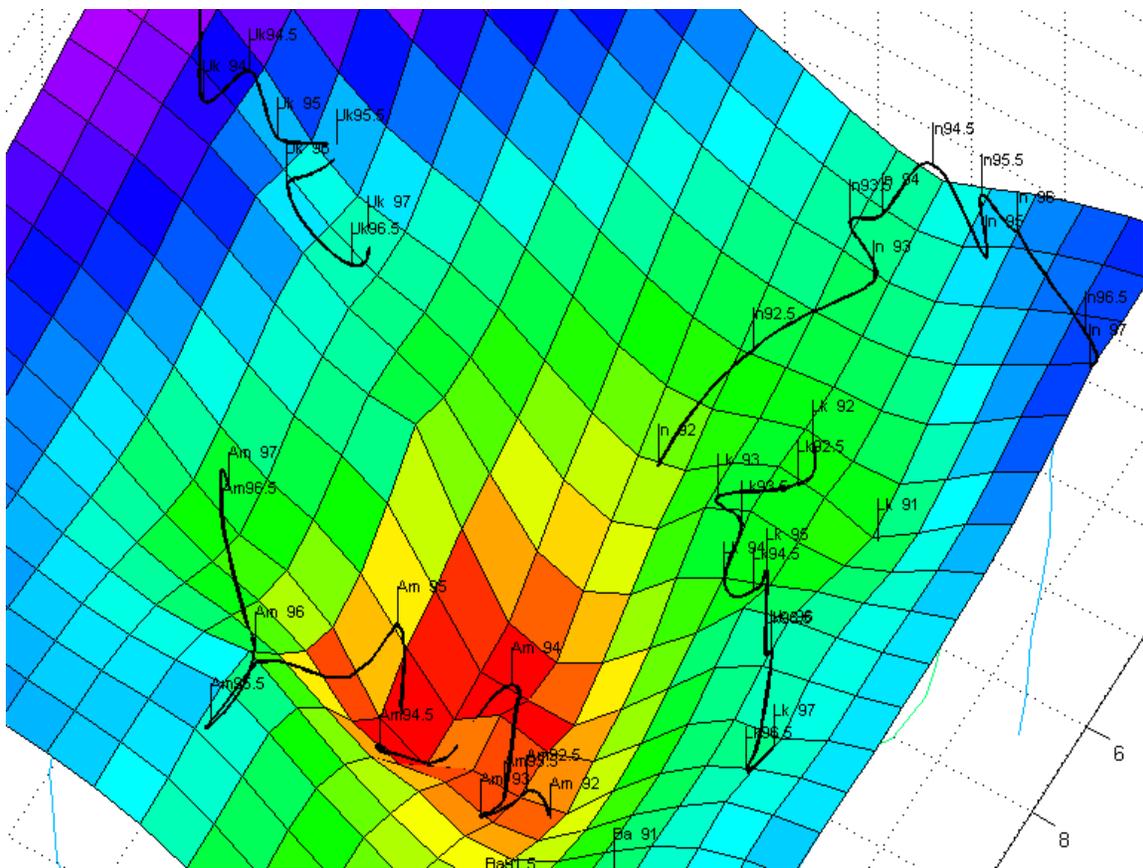
Figure 17: Skandia IC landscape. Height dimension is IC. Am = American Skandia, UK = UK Life, In = Intercaser, Di = Dial
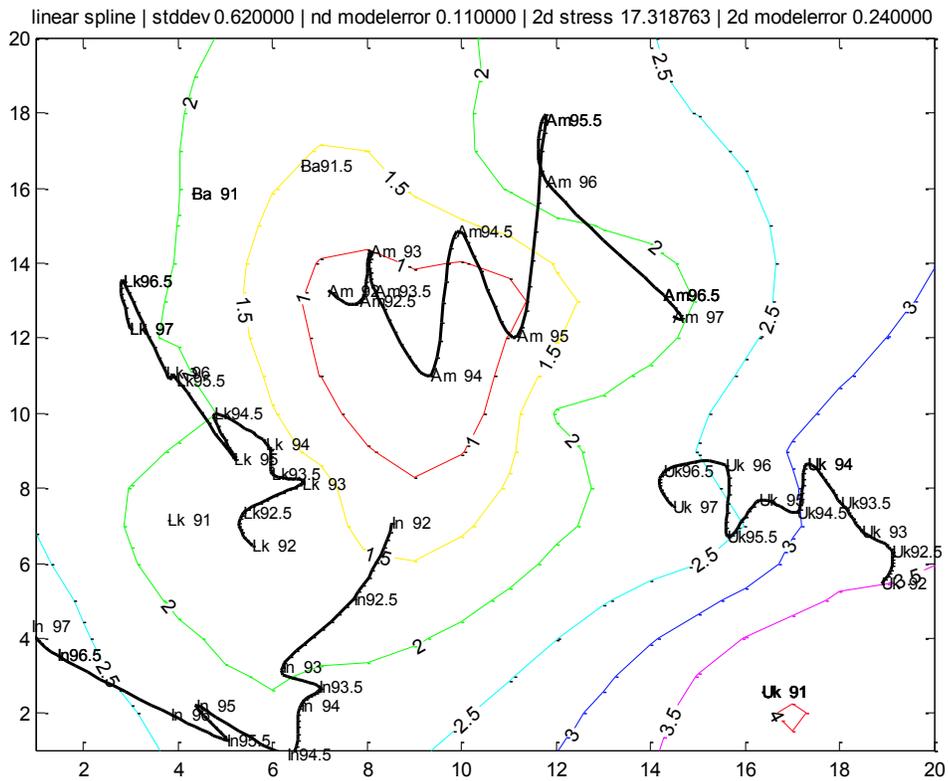
Figure 18: Top-down view of Skandia IC Map

Independent	Mean absolute error	R	Rank correlation
5D company vector	0.18	0.99	0.98
2D map location	0.66	0.91	0.83

Figure 19: Prediction accuracy of Skandia Map calculated using 2-fold cross-validation. The accuracy of using a 5-dimensional vector to predict fitness was also tested, to gauge the degradation in accuracy due to the low-dimensional map. The test revealed that accuracy did degrade, although the map predictions of fitness were still strongly correlated to actual values.
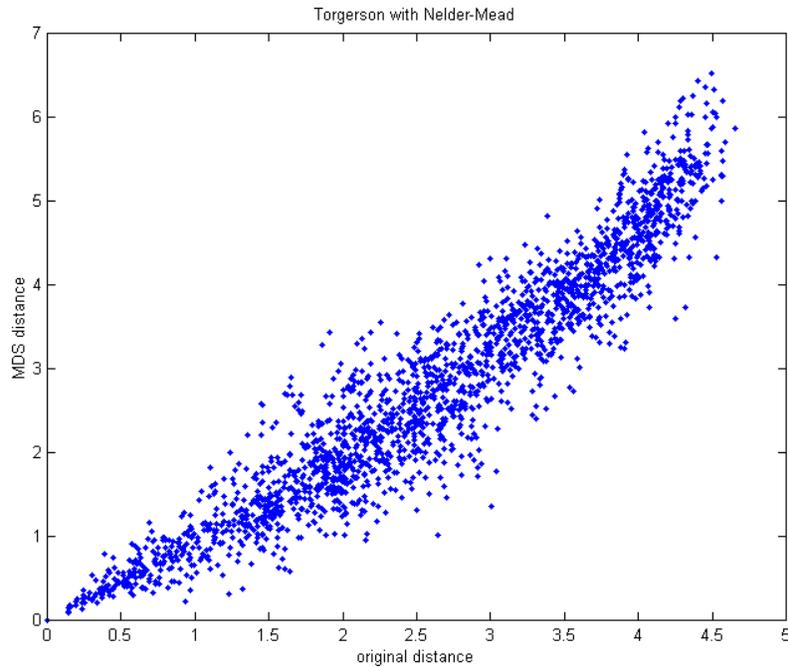
Figure 20: Distances on the Skandia IC Map versus the high-dimensional points. Distances on the IC map are very close to their high-dimensional counterparts, with a correlation of 0.91. Thus map positions a good representation of company differences.

Application

American Skandia has experienced sharp growth in the 1990s, rising from a small company of 94 employees, to a successful investment company employing over 599. The older UK Life, has experienced slower growth during the 1990s. Should UK Life adopt operating practices more like those of American Skandia?

The main difference between UK Life in 97 and 93 are *drops in Customer and Renewal* capital (figure 24). UK Life would need to increase its number of customers and rate of new contracts in order to get back to the position it occupied in 1993.

American Skandia 1997 meanwhile, has a lower level of overall fitness, and reaching Am97 would take 2.8 years, whilst UK93 is 2.2 years away. As a result, management should conclude that it is better for UK Life to return to operating practices similar to 1993 by increasing expenditure on customer satisfaction and running programs to acquire new customers.

	Financial	Customer	Human	Process	Renewal
UK93	1.64	1.19	1.77	-0.65	-0.39
UK97	0.41	0.27	1.62	-0.65	-0.98
UK93-UK97	+1.23	+0.93	+0.15	0	+0.59

Figure 24: Difference between UK97 and UK93

Destination	Mean Standard deviations difference	Time to reach location	Fitness at new location (estimate)	Changes required
Present location	0	0	0.67	No changes
UK93	0.58	2.2 years	3.30	Increase Renewal and Customer
Am97	0.74	2.8 years	0.06	Increase Renewal, decrease Customer

Figure 25: Time for UK Life to reach various destination on the landscape (assuming straight-line movement and no interactions). UK Life has changed its parameters at an average rate of 0.266 standard deviations per year, so to reach UK93 would require 0.5794 / 0.266 = 2.2 years to reach.

Experiment 2: UN Map

Each year the World Bank Organization publishes a document titled *The World Development Indicators Report*, in which statistics for 140 countries over 35 years are listed. From the electronic version of this report, we obtained 64 variables with low percentages of missing values, and chose 44 countries for analysis. For purposes of the experiment, we have chosen to use an extremely simple indicator of the well-being, the "life expectancy" for citizens in these countries.

The resulting landscape is shown in figure 19. The map is stretched along one dimension, with developing countries at one end, and developed countries at the other. This stretching could be caused because many of the variables are co-linear, and correlated with the same underlying cause, such as gross domestic product.

One of the nice surprises about this landscape, is that many developing countries have progressed up the landscape in the 35 year measurement period. Papua New Guinea has moved from a life expectancy of 46 to 58. India and Pakistan have also both improved their life expectancies from 49 to 63 and 49 to 64 respectively. India and Pakistan have similar positions on the map (figure 26). Most of these countries, including the United States and Japan, are traveling "up" the landscape in a direction of greater life expectancy.

The country with poorest life expectancy on earth in 1998 was Sierra Leone. Sierra Leone briefly appears to be getting better from 1980-1990, however by 1995, the direction of movement is at tangent to other countries who are marching up the life-expectancy landscape, and into territory no other nation has traversed. This is probably a bad sign!

.

Figure 26: Fitness landscape for 44 nations from the World Bank's World Development Indicator data, from 1970 to 1995. Height axis is life-expectancy at birth, with lighter colors representing better life expectancy. Close-up of India, Pakistan, and Sierra Leone. US and Japan are shown in the top-left climbing towards a life-expectancy peak.

Conclusion

We have developed a method for organizing company knowledge into a form which people can understand, and used it to provide advanced automated decision support functions such as predicting company fitness at untested parameters (seen as the landscape around known points), what-if analysis, estimation of time and cost required to reach new states.

The techniques used to construct the map are mathematically transparent and robustly minimize error in the low-dimensional representation. The maps themselves are intuitive, compelling, and graphically depict the movement of companies across the landscape.

There are many areas in which these maps can be improved. Error estimates can be provided by adding error-bars to the surface. Graphically these could be represented as an "atmosphere" or "mist" that covers the landscape above and below. The rate of change of the topographic fitness landscape itself is another issue - the surface may deform over time as the economy or market changes. We hope that this geological change will be small compared to the speed of company movement so that it won't greatly impact planning, and future work will need to calculate fair planning horizons are on particular IC maps.

References

Betteridge, D., Wade, A. and Howard, A. (1985), Reflections on the modified simplex II, *Talanta*, Vol. 32, 8B, pp. 723-734.

Bowditch, N. (1826), *The New American Practical Navigator*, sixth edition, Edmund L. Blunt, New York. http://www.iws.net/wier/logline.html

Duch, W. and Naud, A. (1998a), Simplexes, Multi-Dimensional Scaling and Self-Organized Mapping, Technical Report, Department of Computer Methods, Nicholas Copernicus University, Poland. http://www.phys.uni.torun.pl/kmk

Duch, W. and Naud, A. (1998b), Multidimensional scaling and Kohonen's Self-organizing maps, Technical Report, Department of Computer Methods, Nicholas Copernicus University, Poland. http://www.phys.uni.torun.pl/kmk

Edvinsson, L. and Malone, M. (1997) *Intellectual Capital*, Harper.

Girosi, F., Jones, M. and Poggio, T. (1993), Priors, Stabilizers and Basis Functions: from regularization to radial, tensor and additive splines, AI Memo 1430, CBCL Paper 75, http://www.ai.mit.edu/people/girosi/home-page/memos.html

Kaplan, R. and Norton, D. (1996), *The Balanced Scorecard: Translating Strategy into Action*, Harvard Business School Press. MA.

Karlgaard, R. (1993), Rest in Peace, Book Value, *Forbes ASAP*, October 25, pp. 9.

Karur, S. and Ramachandran, P. (1995), Augmented Thin Plate Spline Approximation in DRM, *Boudnary Elements Communications*, Vol. 6, pp. 55-58.
http://wuche.wustl.edu/~karur/papers.html

Kohonen, T. (1996), *Self-Organizing Maps*, second edition, Springer Series in Information Sciences, Vol. 30, Springer, Berlin.

Kruskal, J. (1978), *Multidimensional Scaling*, Sage University series, Beverly Hills, CA.

Li, S. (1993), Dimensionality Reduction using The Self-Organizing map, Honours Thesis, James Cook University, North Queensland. http://www.cs.jcu.edu.au/ftp/pub/

Li, S., de Vel, O., Coomans, D. (1995), Comparative Performance Analysis of Non-linear Dimensionality Reduction Methods, *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*, Fort Lauderdale, Florida.
http://www.cs.jcu.edu.au/ftp/pub/techreports/94-8.ps.gz

von der Malsburg (1973), Self-organization of orientation sensitive cells in striate cortex, *Kybernetik*, Vol. 14, pp. 85-100.

Nelder, J. and Mead, R. (1965), A simplex method for function minimization, *Computer Journal*, Vol. 7, pp. 308-313.

Norusis, M. (1997), Multidimensional scaling Examples, *SPSS Professional Statistics 7.5 User Manual.*

Orr, M. (1996), Introduction to Radial Basis Function Networks, Centre for Cognitive Science, University of Edinburgh, 2 Buccleuch Place, Edinburgh EH8 9LW, Scotland, http://www.cns.ed.ac.uk/people/mark/intro/intro.html

Shepard, R. (1974), Representation of structure in similarity data, *Psychometrika*, Vol. 39, pp. 373-421.

Torgerson, W. (1958), Multidimensional scaling, in P. Colgan (ed), *Quantitative ethology*, pp. 175-217., Wiley, NY.

Young, F. and Hamer, R. (1987), *Multidimensional scaling: History, theory and applications*, Lawrence Erlbaum, NJ.

Young, F. (1998), Multidimensional scaling, Lecture notes, University of North Carolina, http://forrest.psych.unc.edu/teaching/p230/p230.html

World Development Indicators 1998 CD-ROM, The World Bank, ISBN: 0-8213-4375-0, http://www.worldbank.org/html/extpb/wdi99.htm

Comparative Civilian Labor Force Statistics, Ten Countries, 1959-1999, US Department of Labor, Bureau of Labor Statistics, Office of Productivity and Technology, May 27, 1999. http://stats.bls.gov/flsdata.htm

An Introduction to Business Valuation, Madison Valuation Associates, 1999. http://www.madval.com/introduction.html