Optimizing Relationships Information in Repertory Grids

Enrique Calot[1], Paola Britos[2], and Ramón García-Martínez[3]

Abstract The Repertory Grid method is widely used in knowledge engineering to infer functional relationships between constructs given by an expert. The method is ignoring information that could be used to infer more precise dependencies. This paper proposes an improvement to take advantage on the information that is being ignored in the current method. Furthermore, this improvement fixes several other limitations attached to the original method, such as election in a discrete set of two values as a similarity pole or a contrast pole, the arbitrary measurement of distances, the unit-scale dependency and the normalization, among others. The idea is to use linear regression to estimate the correlation between constructs and use the fitness error as a distance measure.

1 Introduction

The Repertory Grid method is widely used in knowledge engineering to infer functional relationships between constructs given by an expert. The the original method is ignoring information that could be used to infer more precise dependencies [1], [2] and [3].

This paper proposes an improved method using linear regression to calculate the dependencies using the given values and interpreting the scales and units. Vectorial constructs like colors and location are also be supported by this method.

[1] Enrique Calot
Intelligent Systems Laboratory. School of Engineering. University of Buenos Aires. ecalot@fi.uba.ar
[2] Paola Britos
Software & Knowledge Eng. Center. Buenos Aires Institute of Technology. pbritos@itba.edu.ar
[3] Ramon Garcia-Martinez
Software & Knowledge Engineering Center. Buenos Aires Institute of Technology. rgm@itba.edu.ar

To do that the relationships are described using a relationship function or "model" with coefficients to bee calculated using the least squares method. The residuals that could not have been fit by the regression are the ones not explained by the model and are used to calculate a better measure of the distance. In Sec. 4 a use case is provided.

2 Deficiencies of the original repertory grid method

The following paragraphs enumerate some deficiencies in the original repertory grid method, the goal is to solve most of them.

2.1 Generated trees rely on the scale and units inherent to the data, which has been arbitrarily normalized

When comprising distances between the constructs C_1 and C_2 there could be errors resulting from the measurement using different magnitudes that make the data sensible to scale changes, in the repertory grid method this issue is ignored simply taking the numerical values without checking the unit; even though this could be dangerous because the results may be different depending on what units the expert is measuring on, even when normalized to numbers from 1 to 5. e.g. Celsius, Fahrenheit and Kelvin scales measure temperature, but the zeros are in different places, so the measured values are not proportional. Even when normalized, the values will be different depending on the expert's scale choice. Another examples are the logarithmic scales such as pH, decibels and even musical notes.

2.2 Vectorial constructs are not supported

There are constructs that have a vectorial nature such as colors, coordinates and so forth. They cannot be reduced to a linear scale because they could not only depend on the way they were converted to a scalar value but also much information could be lost (and degrees of freedom). e.g. a color may be represented in values like 1=red 3=blue 5=black, but the sense of that measure is completely lost. Neither can be separated into different components and be studied independently because one of the components is inherently related to the others and the construct must be studied as a whole and not by the parts. e.g. A dead pixel may depend on the intensity of the red value (in its RGB scale) but the human eye percepts the colors better using the HSL scale. When measuring in HSL it is very unlikely to see the direct correspondence between the probability to lose a pixel and the red component of the color if it is measured in the HSL scale. Studying the HSL as a whole should find an association and then will be possible to realize that this

relation is very similar to the red transformation function in the HSL to RGB conversion method.

2.3 It is discrete

The grid could be much better generalized if continuous values are used. Greater precision could be acquired when comparing our results.

2.4 The distance measurement is a kind of dubious

Using the 1-norm is arbitrary. How do we know that this is the better choice?

3 The Proposed Method

The objective of the repertory grid method is to find functional relationships between constructs, the original method proposes the equality between two constructs as the optimal dependency and then measures how deviated are the constructs to each other using the 1-norm.

This paper, in contrast, proposes the use of a regression method to fit the given data using the resultant fitness as a measure of the relationship between the adjusted constructs.

3.1 Definitions on matrices

Before stating the method some definitions on the original repertory grid method are to be explained.

3.1.1 Repertory grid matrix

Let G be the grid matrix. It has n elements and m characteristics. The notation g_{G_i,C_j} with $0 \le i < n, 0 \le j < m$ to each of its elements will be used.

3.1.2 Distance matrix

Let D be the matrix containing the distance between characteristics. It has m columns and m rows, one for each characteristic. It is superior triangular without diagonal, so d_{C_i,C_j} with $0 \le i < m, i < j < m$ for each of its elements. In the original repertory grid method it is the 1-norm distance between two columns (i and j) in the G matrix.

When adding the trivial twist to support the contrast construct, it should use the minimum value between the 1-norm distance from the first construct to both: the second construct and its contrast (each value of this column C_i is replaced by $6 - C_i$).

3.2 Measuring distances

The measurement is based on the hypothesis that F(Ci,Cj)=1 where Ci and ,Cj are two constructs, then it measures how deviated the is fitness to the hypothetical value (1 in this case). The obtained residue should reflect how both constructs are explained by a model and the degree of dependency between both constructs.

Before doing the measurements, the knowledge engineer should define a model, that is equivalent to state arbitrarily the equations where the fitting will be made. Defining a model is the most important step in the method, because the measurements not only depend on the relationships between the constructs but on how the model applies to the situation. It is possible to use different models to measure distances between different pairs of constructs; the method should support such case.

The distance matrix could be filled by the fitness error. This is really an excellent way to measure the dependency between two characteristics. If the model were completely generic (ideal, but impossible), the fitness error would be the optimal distance to measure the dependence. This is an ideal case and not useful in practice, it is recommended to use simple models to find real relationships and avoid complex equations.

3.3 Defining a fitting model

To calculate the distance between two constructs the knowledge engineer must define a model. In the original method, the model was linear, scalar and discrete. Let $\zeta_{U,V}$ be the model to correlate variables U and V. Examples of models are the linear $\zeta_{U,V}=\{1,u,v\}$ and the quadratic $\zeta_{U,V}=\{1,u,u^2,v,v^2,uv\}$.

The cardinality n of $\zeta_{U,V}$ is the number of coefficients to be calculated by our regression. Let $\vec{\zeta}_{U,V}\in\Re^n$ be the vector representing that model and $\alpha_{\zeta_{U,V}}\in\Re^n$ the vector of coefficients represented by that model. These definitions arrive to the current ideal equation

$$F(C_1,C_2)=1=\alpha_{\zeta_{C_1,C_2}}\vec{\zeta}_{C_1,C_2} \quad (1)$$

if there is a $\alpha_{\zeta_{C_1,C_2}}$ that satisfies the equation for all pair of c_1 and c_2, then the fitness is perfect.

For example, the linear model $\zeta_{U,V}=\{1,u,v\}$ derives to the plane

$$F(U,V) = 1 = \alpha_1 + \alpha_2 u + \alpha_3 v \quad (2)$$

and the quadratic $\zeta_{U,V} = \{1, u, u^2, v, v^2, uv\}$ to

$$F(U,V) = 1 = \alpha_1 + \alpha_2 u + \alpha_3 u^2 + \alpha_4 v + \alpha_5 v^2 + \alpha_6 uv \quad (3)$$

Let ζ_W be the one-variable model related to the model $\zeta_{U,V}$ and may be obtained by

$$\zeta_W = \zeta_{U,V} \lfloor_{U=W,V=0} \cup \zeta_{U,V} \lfloor_{U=0,V=W} \quad (4)$$

that is $\zeta_W = \{1, w\}$ for the linear model and $\zeta_W = \{1, w, w^2\}$ for the quadratic.

3.4 Limitations of the proposed method

The method is linear since linear regression has been used. This means that the resultant relationships will be shown in euclidean subspaces resulting from the sum of terms with the form of coefficient α_n multiplied by a function dependent of the input data. This function is part of the model and does not necessarily need to be linear. It is not on the scope of this paper to study non-linear dependencies.

3.5 Calculating the regression

The least squares method should find the best fit. The matrix A related to two constructs is calculated by the evaluation of each construct's value in the desired model. The matrix is calculated column by column for each row in G as

$$A_i = \vec{\zeta}_{U,V} \lfloor_{U=G_{C_1,i}, V=G_{C_2,i}} \quad (5)$$

which is exactly evaluating the model (except the first 1) with the values of each row from the repertory grid matrix.

Finally the coefficients may be obtained by the multiplication of the pseudo-inverse matrix [2] and the unit vector.

$$\alpha = (A^T A)^{-1} A^T \vec{1} \quad (6)$$

The first value in the model (the constant part) must not be used because it will be in the other side of que equation as the unit vector $\vec{1}$. By doing that resultant equation Eq. (1) representing the model has been calculated.

3.6 Measuring the residuals

The desired measure of the fitness may be expressed by the residuals, that is the difference between the model evaluated with the repertory grid elements and the ideal result that is the unit vector.

$$\alpha \vec{\zeta}_{U,V} = \vec{1} + \vec{\varepsilon}_{U,V} \Rightarrow \mathrm{P}\alpha \vec{\zeta}_{U,V} - \vec{1}\mathrm{P} = R_{U,V} \quad (7)$$

where $R_{U,V} = \mathrm{P}\vec{\varepsilon}_{U,V}\mathrm{P}$.

The first impression is that $R_{U,V}$ is a good measure of the correlation between U and V but the fact is that it is a good measure of "what may not be explained by the model". It is possible that a construct is very attached to itself, its variance is very small and therefore the fitness is very small too.

For example, in the linear model $\zeta_{U,V} = 1, u, v$, it is possible that the construct U has values strictly around 6 for all of its elements. The resultant plane will be $\frac{u}{6} + 0v = 1$. In this case $R_{U,V}$ represents the fitness of U by itself and not the fitness of U related to V.

This paper proposes the use of two more regressions, the one related to U and the one related to V with the residuals R_U and R_V respectively. Having known that regressions, a good redefinition of the distance could be "what is explained by the two-variable model that was not already explained by each separated variable using a single variable model". This is the definition

$$d_{i,j} = \frac{\min\{R_{C_i}, R_{C_j}\}}{R_{C_i,C_j}} \tag{8}$$

with $R_{C_i} > R_{C_i,C_j}$ and $R_{C_j} > R_{C_i,C_j}$ because the least squares method had minimized R_{C_i,C_j} with more degrees of freedom. It is easy to show that the distance matrix will have values between 0 and 1, being 0 the stronger relationship according to the chosen model and 1 the weakest one.

3.7 Vectorial constructs

In vectorial constructs the process is mostly the same, the only difference is that the whole vector must be evaluated in the model for each component. That is for example a construct color $\vec{C} = (R, G, B)$ and another scalar lightness L should be evaluated in the quadratic model as

$$\alpha_1 r + \alpha_2 r^2 + \alpha_3 g + \alpha_4 g^2 + \alpha_5 b + \alpha_6 b^2 + \alpha_7 l + \alpha_8 l^2 = 1.$$

The residuals should be divided by the minimum of the regression of each separated construct, in the example by the residual of

$\alpha_1 r + \alpha_2 r^2 + \alpha_3 g + \alpha_4 g^2 + \alpha_5 b + \alpha_6 b^2 = 1$ and $\alpha_1 l + \alpha_2 l^2 = 1$.

3.8 Normalization is no longer needed

A side effect by the use of a dependency function is that the units and scales are inside the model being completely abstracted to the method. There is no longer need to have a discretized input of numbers from 1 to 5, now, the normalization is

inside the method which will find the best fit regardless the scale and the units. If the scale is logarithmic adding a logarithm to the model should be enough.

4 Case Study

Our expert is providing the knowledge engineer with four constructs; the first one is the vectorial location of a city ($\vec{L} \in \Re^2$), its population ($P \in N$), temperature ($T \in \Re$) and the average level of pollution ($O \in \Re$). The obtained values are shown in Table 1.

Table 1. Population, temperature and pollution of a city regarding to its location on an arbitrary coordinate plane.

Location km;km	**Population** Hab	**Temperature** °C	**Pollution** ppm
(9.83982;40.4372)	73272	30.8322	36.8086
(17.3862;69.5633)	65115	27.916	41.1447
(24.1684;89.5489)	94737	25.7233	42.2755
(26.9449;57.6548)	85173	25.5949	28.7814
(47.1808;33.3024)	102663	29.8456	15.5273
(67.8653;72.7391)	118860	27.717	24.7249
(48.1759;80.8657)	19293	24.4881	26.9948
(16.3168;20.8034)	105084	29.317	28.6556
(28.1486;43.4684)	57170	29.8976	29.4861
(55.1659;3.49954)	3431	30.2999	17.1146
(45.7604;63.8792)	4965	26.7262	25.5383

The knowledge engineer calculates the regressions and compares the results as shown in Table 2. Finally, as the temperature quadratic model has a very small variance by itself, the engineer decides to use the linear model for this construct. To calculate the distances between constructs the smallest one-construct residual is taken to divide the two-construct to be measured. The resultant distance matrix calculated by all this divisions is shown in Table 3.

Finally we perform the tree building method as shown in Fig. 1. As we can see, the Pollution is primarily related to the location, then to the temperature and finally to the population. In Fig. 2 it is shown the level of pollution over a region of $\vec{L}$, deduced from the resultant subspace, in Fig. 3 the temperature is shown under the linear model and in Fig. 4 under the wrong quadratic model which had been discarded by the knowledge engineer.

The pollution equation suggests that the proximity to (60km;20km) has low pollution, perhaps it is the top of a mountain. As we can observe, the method found the dependencies.

Table 2. Comparison between models.

Construct	Model	**Residual**	**Subspace**
T	Quadratic	0.0144453	$0.0723193\ t - 0.00130023\ t^2 = 1$
T	Linear	0.244034	$0.035479\ t = 1$
P	Quadratic	1.37606	$0.000030232\ p - 1.9863\ 10^{-10}\ p^2 = 1$
O	Quadratic	0.291629	$0.0697022\ c - 0.00113199\ c^2 = 1$
$\vec{L}_1$	Quadratic	0.733654	$0.0574155\ l_1 - 0.000689149\ l_1^2 = 1$
$\vec{L}_2$	Quadratic	0.979385	$0.0391949\ l_2 - 0.00033698\ l_2^2 = 1$
$\vec{L}$	Quadratic	0.560544	$0.0356493\ l_1 - 0.000410520\ l_1^2 + 0.0168284\ l_2 - 0.00015598\ l_2^2 = 1$
$\vec{L},T$	Quadratic	0.0137924	$0.0722\ t - 0.0013\ t^2 + 0.0002\ l_1 - 3.0117\ 10^{-6}\ l_1^2 - 0.0001\ l_2 + 1.403\ 10^{-6}\ l_2^2 = 1$
$\vec{L},P$	Quadratic	0.43544	$0.000013\ p - 9.1028\ 10^{-11}\ p^2 + 0.0308\ l_1 - 0{,}0003\ l_1^2 + 0.0033\ l_2 - 0.00035\ l_2^2 = 1$
$\vec{L},O$	Quadratic	0.1191	$0.03239\ o - 0.0002\ o^2 + 0.0176\ l_1 - 0.0001\ l_1^2 + 0.0008\ l_2 - 0.00004\ l_2^2 = 1$
T,O	Quadratic	0.0143339	$0.00039\ o - 7.008\ 10^{-6}\ o^2 + 0.0719\ t - 0.0013\ t^2 = 1$
P,O	Quadratic	0.286957	$0.0687\ o - 0.00111\ o^2 - 4.633\ 10^{-7}\ p + 6.992\ 10^{-12}\ p = 1$
T,P	Quadratic	0.0128094	$8.9589\ 10^{-8}\ p - 1.1170\ 10^{-12}\ p^2 + 0.0723\ t - 0.0013\ t^2 = 1$
$\vec{L},T$	Linear	0.12522	$0.0309\ t + 0.0006\ l_1 + 0.002\ l_2 = 1$
$\vec{L},T$	Combined	0.0952247	$0.0291\ t + 0.0063\ l_1 - 0.00008\ l_1^2 + 0.0009\ l_2 + 0.00001\ l_2^2 = 1$
O,P	Combined	0.155222	$0.0289\ o - 0.0005\ o^2 + 0.0204\ t = 1$
T,P	Combined	0.243501	$4.78\ 10^{-7}\ p - 3.855\ 10^{-12}\ p^2 + 0.0351\ t = 1$

Table 3. Distance as relationships between constructs using an arbitrary model.

	$\vec{L}$	P	T	O
$\vec{L}$		0.776819	0.390211	0.212472
P			0.997819	0.98398
T				0.636069
O				

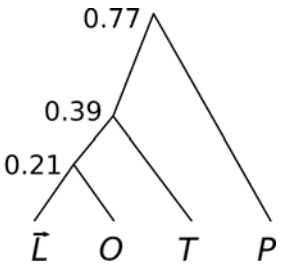

Fig. 1. Tree view of the distances built by the proposed repertory grid method

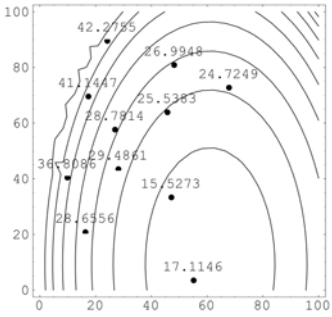

Fig. 2. Regressed pollution depending on the location.

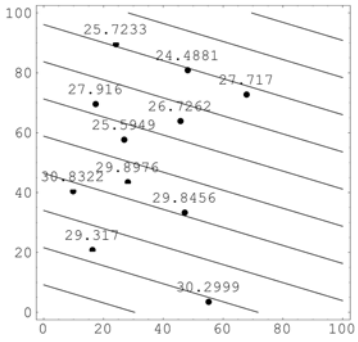

Fig. 3. Regressed temperature under a linear model depending on the location.

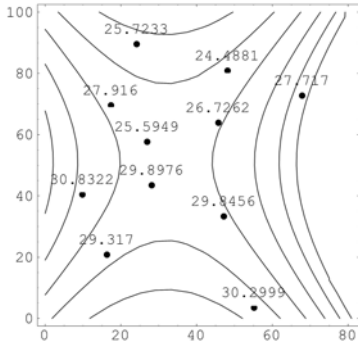

Fig. 4. Regressed temperature under a wrong quadratic model depending on the location.

5 Conclusions

The proposed method has potential application on several fields, specially in knowledge acquisition. The usage of pre-designed model instead of the discrete-linear one may fit with more constructs and helps the knowledge engineer in the exploration of the construct. Future lines of development may find better ways to choose the appropriate model. Using the fitness coefficient as a measure is refined generalization of the method.

References

1. Bradshaw, J.M., Ford, K.M., Adams-Webber, J.R. and Boose, J.H. *Beyond the repertory grid: new approaches to constructivist knowledge acquisition tool development*. International Journal of Intelligent Systems 8(2) 287-33. (1993).
2. Acton. F. S. *Analysis of Straight-Line Data*. Dover Publications. (1966).
3. Beeri, C., Fagin, R. and Howard J. H. (1977). *A complete axiomatization for functional and multivalued dependencies in database relations*, In Proceedings of the 1977 ACM SIGMOD international Conference on Management of Data (Toronto, Ontario, Canada, August 03 - 05, SIGMOD '77. ACM, New York, NY, 47-61. (1977).
4. Barlett. D. *General Principles of the Method of Least Squares*. Dover Publications. (2006).