# Temporal Extrapolation within a Static Clustering

Tim K. Cocx, Walter A. Kosters and Jeroen F.J. Laros

Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands
tcocx@liacs.nl

**Abstract.** Predicting the behaviour of individuals is a core business of policy makers. This paper discusses a new way of predicting the "movement in time" of items through pre-defined classes by analysing their changing placement within a static, preconstructed 2-dimensional clustering. It employs the visualization realized in previous steps within item analysis, rather than performing complex calculations on each attribute of each item. For this purpose we adopt a range of well-known mathematical extrapolation methods that we adapt to fit our need for 2-dimensional extrapolation. Usage of the approach on a criminal record database to predict evolvement of criminal careers, shows some promising results.

## 1 Introduction

The ability to predict (customer) behaviour or market trends plays a pivotal role in the formation of any policy, both in the commercial and public sector. Ever since the coming of the information age, the procurement of such prognoses is becoming more and more an automated process, extracting and aggregating knowledge from data sources, that are often very large.

Mathematical computer models are frequently used to both describe current and predict future behaviour. In many cases these models are chosen on the basis of *detection theory* [2]. They employ algorithms like *Naive Bayes*, *K-Nearest Neighbour* or concepts like *Support Vector Machines*. Next to the prediction of certain unknown attributes by analysing the available attributes, it might also be of interest to predict behaviour based upon past activities alone, thus predicting the continuation of a certain sequence of already realized behaviour.

Sequences play an important role in classical studies of instrumental conditioning [3], in human skill learning [10], and in human high-level problem solving and reasoning [3]. It is logical that sequence learning is an important component of learning in many task domains of intelligent systems. Our approach aims to augment the currently existing set of mathematical constructs by analysing the "movement in time" of a certain item through a static clustering of other items. The proposed model can be added seamlessly to already performed steps in item analysis, like clustering and classification, using their outcome as direct input.

Section 2 mentions extrapolation methods. The main contribution of this paper is in Section 3, where the new insights into temporal sequence prediction are discussed. Section 4 shows experiments, and Section 5 concludes.

## 2 Background

A lot of work has been done in the development of good clustering and strong extrapolation methods that we can resort to within our approach.

### 2.1 Clustering

It is common practice to visualize a clustering within the 2-dimensional plane, utilizing some form of *Multi Dimensional Scaling* (MDS) [6] to approximately represent the correct, multi-dimensional distances. These methods include, e.g., "associative array" clustering techniques [9] and systems guided by human experience [5]. An obvious choice for our approach would be to select the method yielding the smallest error.

### 2.2 Extrapolation

*Extrapolation* is the process of constructing new data points outside a discrete set of known data points, i.e., predicting some outcome on a yet unavailable moment. It is closely related to the process of interpolation, which constructs new points between known points and therefore utilizes many of its concepts, although its results are less reliable.

**Interpolation** Interpolation is the method of constructing a function which closely fits a number of known data points and is sometimes referred to as *curve fitting* or *regression analysis*. There are a number of techniques available to interpolate such a function, most of the time resulting in a polynomial of a predefined degree $n$. Such a polynomial can always exactly fit $n+1$ data points, but if more than $n+1$ points are available one needs to resort to approximation measures like the *least squares error* method [1]. The two main interpolation methods that are suitable to be incorporated in our approach are *polynomial interpolation* and *splines*. Polynomial interpolation uses linear algebra to solve a system of linear equations in order to find one polynomial that best approximates the given data points (see Figure 1).

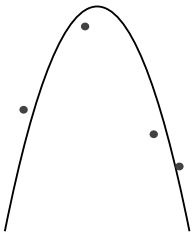

**Fig. 1.** A function of degree 2 that best fits 4 data points

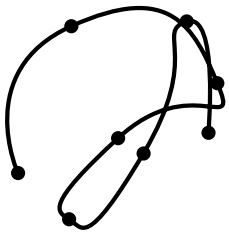

**Fig. 2.** An example of a spline

Data points can also be interpolated by specifying a separate low degree polynomial (e.g., degree 2 or 3) between each couple of data points or *knots*. This interpolation scheme, called a *spline*, exactly fits the derivative of both polynomials ending in the same knot. Demanding that the second derivatives also match and specifying the requested derivative in both end points yields $4n$ equations for $4n$ unknowns. Following Bartels et al. [4] one can specify a third degree polynomial for both the $x$ and $y$ coordinates between two separate knots, resulting in an interpolation like the graph in Figure 2. Due to the liberty this method allows in the placement of the existing data points, it seems well suited for the task of 2-dimensional extrapolation, see below.

**Extrapolation** All interpolation schemes are suitable starting points for the process of extrapolation. It should, however, be noted that higher level polynomials can lead to larger extrapolation errors: the *Runge phenomenon*. Polynomials of degrees higher than 3 are often discouraged here.

In most cases, it is sufficient to simply continue the fabricated interpolation function after the last existing data point. In the case of the spline, however, a choice can be made to continue the polynomial constructed for the last interval (which can lead to strange artifacts), or extrapolate with a straight line, constructed with the last known derivative of that polynomial. The difference between the two methods is displayed in Figure 3.

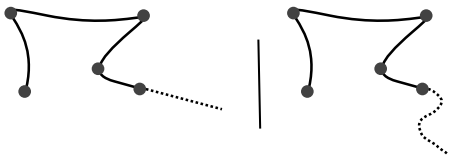

**Fig. 3.** Straight line extrapolation (left) and polynomial continuation (right)

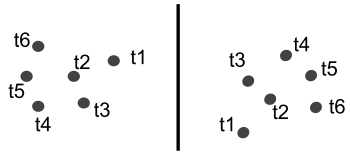


**Fig. 4.** Rotation with the best left-right ordering on the $x$-axis. Note that 2 and 3 remain in the wrong order

**2-Dimensional Extrapolation** In our approach both $x$ and $y$ are coordinates and therefore inherently independent variables. They depend on the current visualization alone. Within our model, they do however depend on the time variable $t$. Because our methods aims to extrapolate $x, y$ out of one other variable $t$, we need a form of 2-dimensional extrapolation. After rotation and under the assumption that $x$ is in fact the independent variable guiding $y$, standard methods can be used. For this scenario we need to establish a rotation that best fits the time ordering to a left-right ordering on the $x$-axis as displayed in Figure 4.

It is also possible to use the polynomial extrapolation for the $x$ and $y$ variables separately and combine them into a linear system, much like spline interpolation, only for the entire domain (referred to as $x$,$y$ system): $x = p_1(t)$, $y = p_2(t)$. Naturally, the dependence of $x$ and $y$ on $t$ within the spline interpolation scheme makes that method very well suited for the task of 2-dimensional extrapolation.

This leaves six methods that are reasonably suited for our approach:

1. Second degree polynomial extrapolation
2. Third degree polynomial extrapolation
3. $x$,$y$ system with second degree polynomial extrapolation
4. $x$,$y$ system with third degree polynomial extrapolation
5. Spline extrapolation with straight line continuation
6. Spline extrapolation with polynomial continuation

## 3 Approach

The number of attributes describing each item in a database can be quite large. Taking all this information into account when extrapolating can therefore be quite a hassle. Since this information is inherently present in an already performed visualization of a clustering, we can theoretically narrow the information load down to two attributes ($x$ and $y$) per item whilst retaining the same accuracy. The stepwise strategy is illustrated in Figure 5.

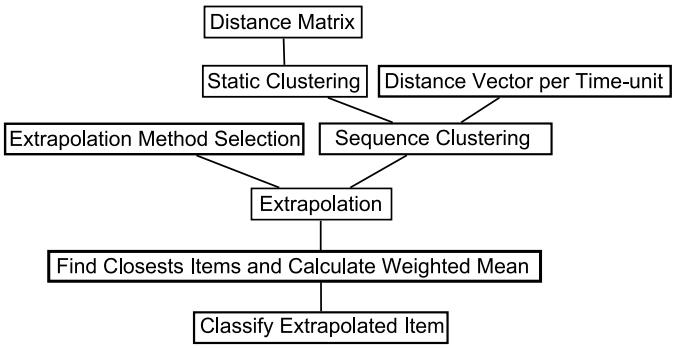


**Fig. 5.** Stepwise approach

### 3.1 Distance Matrix and Static Clustering

The data used as reference within our approach is represented by a square $q \times q$ distance matrix describing the proximity between all $q$ items. These items are such that their data is fully known beforehand. The number of items $q$ should be large enough to at least provide enough reference material on which to base the extrapolation. These items are clustered and visualized according to some MDS technique resulting in a 2-dimensional plane with dots representing our reference items. This step in the approach is done only once so the focus should be on the quality of the clustering instead of the computational complexity. From this point on this clustering is considered to be fixed or static.

### 3.2 Distance Vector Time-unit and Sequence Clustering

Analysis of the behaviour of new items should start with the calculation of the values for each time-unit $t$. These units are supposed to be cumulative,

meaning that they contain all the item's *baggage*, i.e., its whole history, up to the specified moment. Using the same distance measure that was used to create the initial distance matrix, the *distance vector per time-unit* can now be calculated. This should be done for all $t$ time-units, resulting in $t$ vectors of size $q$. These vectors can now be visualized as before. The chosen visualization method should naturally allow for incremental placement of individual items, e.g., as in [9]. These new data points within the clustering will be used to extrapolate the items behaviour through the static clustering.

### 3.3 Extrapolation

After selecting the best extrapolation scheme for our type of data our method creates a function that extrapolates item behaviour. For the same data the different schemes can yield different results as illustrated in Figure 6, so care should be taken to select the right type of extrapolation for the data under consideration.

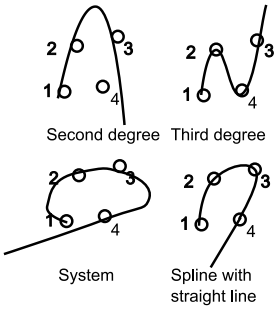


**Fig. 6.** Different extrapolation methods yield very different results

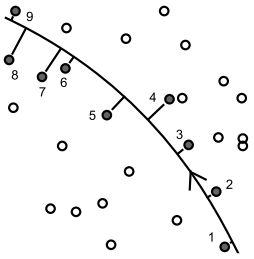

**Fig. 7.** Selecting points with the shortest distance to the extrapolation line

One advantage of this approach is that the extrapolation or prediction is immediately visualized to the end-user rather than presenting him or her with a large amount of numerical data. If the user is familiar with the data under consideration, he/she can analyse the prediction in an eye blink. Augmenting the system with a click and point interface would enable the end-user to use the prediction as a starting point for further research.

### 3.4 Final Steps

In most cases it is desirable to predict which class the item under consideration might belong to in the future. In that case it is important to retrieve further information from some of the reference items and assign future attribute values and a future class to the item.

A first step would be to select $r$ reference items closest to the extrapolation line. This can easily be done by evaluating the geometric distance of all reference points to the line and selecting those with the smallest distance, see Figure 7.

We order these points by their respective distance to the last known data point of the extrapolated item: the confidence of the prediction declines with this distance. We estimate the value for the future attribute $j$:

$$Attrib_j(future) = \frac{2}{r(r+1)} \cdot \sum_{i=1}^{r}(r-i+1)\ Attrib_j(point\ i)$$

The extrapolated item can now be visualized into the clustering according to its future attributes and be classified accordingly.

## 4 Experiments

The detection, analysis, progression and prediction of criminal careers is an important part of automated law enforcement analysis [7, 8]. Our approach of temporal extrapolation was tested on the national criminal record database of The Netherlands. This database contains approximately one million offenders and their respective crimes (approximately 50 types).

We clustered 1,000 criminals on their criminal careers, i.e., all the crimes they committed throughout their careers. In this test-case $r$ will be set to 30. We employed a ten-fold cross validation technique within this group using all of the different extrapolation methods in this static clustering, and standard extrapolation on each of the attributes (methods 7 and 8). For each item (i.e., person) in the test set we only consider the first 4 time periods. The accuracy is described by the mean similarity between the calculated and the expected values of the attributes. The results are presented in Table 1, where *time factor* represents runtime slowdown with respect to the fastest method under consideration.

**Table 1.** Results of Static Clustering Extrapolation for the analysis of Criminal Careers

| | *method* | *time factor* | *accuracy* |
|---|---|---|---|
| 1 | Second degree polynomial extrapolation | 1.0 | 79.1% |
| 2 | Third degree polynomial extrapolation | 1.1 | 79.3% |
| 3 | $x$,$y$ system with second degree polynomial extrapolation | 1.9 | 81.5% |
| 4 | $x$,$y$ system with third degree polynomial extrapolation | 2.1 | 87.5% |
| 5 | Spline extrapolation with straight line continuation | 13.4 | 88.7% |
| 6 | Spline extrapolation with polynomial continuation | 13.4 | 79.6% |
| 7 | Regular second degree attribute extrapolation | 314.8 | 89.0% |
| 8 | Regular third degree attribute extrapolation | 344.6 | 82.3% |

Although the runtime needed for visual extrapolation is much less than that of regular methods, the accuracy is comparable. For this database the best result is still a regular second degree extrapolation but its accuracy is just marginally higher than that of the spline extrapolation with a straight line, where its runtime is much larger. The simpler $x$,$y$ system with third degree extrapolation is very fast but still reaches an acceptable accuracy.

## 5 Conclusion and Future Directions

In this paper we demonstrated the applicability of temporal extrapolation by using the prefabricated visualization of a clustering of reference items. We demonstrated a number of extrapolation techniques and employed them to predict the future development of item behaviour. Our methods were tested within the arena of criminal career analysis, predicting the future of unfolding criminal careers.

We showed that our novel approach largely outperforms standard prediction methods in the sense of computational complexity, without a loss in accuracy larger than 1 percentage point. The visual nature of our method enables the analyst of the data to immediately continue his/her research since the prediction results are easily displayed within a simple graphical interface.

Future research will aim at reaching even higher accuracy values by improving the selection of reference items close to the extrapolation line. Different types of data might well be more susceptible to errors, providing another research issue.

**Acknowledgment** The authors would like to thank Kees Vuik and Robert Brijder. This research is part of the DALE (Data Assistance for Law Enforcement) project as financed in the ToKeN program from the Netherlands Organization for Scientific Research (NWO) under grant number 634.000.430.

## References

1. H. Abdi. Least squares. In M. Lewis-Beck, A. Bryman, and T. Futing, editors, *Encyclopedia for Research Methods for the Social Sciences*, pages 792–795. Thousand Oaks (CA): Sage, 2003.
2. H. Abdi. Signal detection theory. In N.J. Salkind, editor, *Encyclopedia of Measurement and Statistics*. Thousand Oaks (CA): Sage, 2007.
3. J. Anderson. *Learning and Memory*. Wiley and Sons, New York, 1995.
4. R.H. Bartels, J.C. Beatty, and B.A. Barsky. *An Introduction to Splines for Use in Computer Graphics and Geometric Modelling*. Morgan Kaufmann, 1987.
5. J. Broekens, T. Cocx, and W.A. Kosters. Object-centered interactive multi-dimensional scaling: Let's ask the expert. In *Proceedings of the Eighteenth Belgium-Netherlands Conference on Artificial Intelligence (BNAIC2006)*, pages 59–66, 2006.
6. M.L. Davison. *Multidimensional Scaling*. John Wiley and Sons, New York, 1983.
7. J.S. de Bruin, T.K. Cocx, W.A. Kosters, J.F.J. Laros, and J.N. Kok. Data mining approaches to criminal career analysis. In *Proceedings of the Sixth IEEE International Conference on Data Mining (ICDM 2006)*, pages 171–177, 2006.
8. J.S. de Bruin, T.K. Cocx, W.A. Kosters, J.F.J. Laros, and J.N. Kok. Onto clustering criminal careers. In *Proceedings of the ECML/PKDD 2006 Workshop on Practical Data Mining: Applications, Experiences and Challenges*, pages 92–95, 2006.
9. W.A. Kosters and M.C. van Wezel. Competitive neural networks for customer choice models. In *E-Commerce and Intelligent Methods, Studies in Fuzziness and Soft Computing 105*, pages 41–60. Physica-Verlag, Springer, 2002.
10. R. Sun, E. Merrill, and T. Peterson. From implicit skills to explicit knowledge: A bottom-up model of skill learning. *Cognitive Science*, 25(2):203–244, 2001.