Algorithmic tools for data-oriented law enforcement

Cocx, T.K.

Chapter 7

Temporal Extrapolation within a Static Visualization

Predicting the behavior of individuals is a core business of policy makers. The creation of large databases has paved the way for automatic analysis and prediction based upon known data. This chapter discusses a new way of predicting the "movement in time" of items through pre-defined classes by analyzing their changing placement within a static preconstructed two-dimensional visualization of pre-clustered individuals. 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 two-dimensional extrapolation. Usage of the approach on a criminal record database to predict evolvement of criminal careers, shows some promising results.

7.1 Introduction

The ability to predict (customer) behavior or market trends plays a pivotal role in the formation of any policy, both in the commercial and public sector. Prediction of later revenues might safeguard investments in the present. Large corporations invest heavily in this kind of activity to help focus attention on possible events, risks and business opportunities. Such work brings together all available past and current data, as a basis on which to develop reasonable expectations about the future. 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 behavior. This branch of data mining, known as *predictive modeling*, provides predictions of future events and may be transparent and readable in for example *rule based systems* and opaque in others such as *neural networks*. In many cases these models

are chosen on the basis of *detection theory* [2]. Models often employ a set of classifiers to determine the probability of a certain item belonging to a dataset, like, for example, the probability of a certain email belonging to the subset "spam". These models employ algorithms like *Naive Bayes*, *K-Nearest Neighbor* or concepts like *Support Vector Machines* [6, 24]. These methods are well suited to the task of predicting certain unknown attributes of an individual by analyzing the available attributes, for example, estimating the groceries an individual will buy by analyzing demographic data. It might, however, also be of interest to predict shopping behavior based upon past buying behavior alone, thus predicting the continuation of a certain sequence of already realized behavior. Examples of this are the prediction of animal behavior when their habitats undergo severe changes, in accordance with already realized changed behavioral patterns, or the prediction of criminal careers based upon earlier felonies.

Sequence learning is arguably the most prevalent form of human and animal learning. Sequences play a pivotal role in classical studies of instrumental conditioning [4], in human skill learning [40], and in human high-level problem solving and reasoning [4]. It is logical that sequence learning is an important component of learning in many task domains of intelligent systems: inference, planning, reasoning, robotics, natural language processing, speech recognition, adaptive control, time series prediction, financial engineering, DNA sequencing, and so on [39]. Our approach aims to augment the currently existing set of mathematical constructs by analyzing the "movement in time" of a certain item through a static visualization of other items. The nature of such a two-dimensional visual representation is far less complicated than the numerical models employed by other methods, but can yield results that are just as powerful. The proposed model can also be added seamlessly to already performed steps in item analysis, like clustering and classification, using their outcome as direct input for its algorithms.

In Section 7.2 we lay out the foundations underpinning our approach, discussed in Section 7.3. In Section 7.4 we discuss the results our method yielded within the area of predicting criminal careers. Section 7.5 concludes this chapter. A lot of effort in this project has gone into the comparison between different types of two-dimensional extrapolation, a field not widely explored in the mathematical world. The main contribution of this chapter is in Section 7.3, where the new insights into temporal sequence prediction are discussed.

7.2 Background

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

7.2.1 Clustering

The goal of *clustering* is the partitioning of a dataset into subsets, that share common characteristics. Proximity within such a dataset is most often defined by some distance measure. It is common practice to visualize such a clustering within the two-dimensional plane, utilizing some form of *Multi-Dimensional Scaling* (MDS) [15] to approximate

the correct, multi-dimensional solution. These methods include “associative array” clustering techniques [27], systems guided by human experience [9] and visualizations on different kinds of flat surfaces [25] yielding an image like Figure 7.1. In such an image, the axes do not represent any specific value, but *Principal Component Analysis* [34] could potentially reveal lines that do. Naturally, transforming a multi-dimensional problem to a two-dimensional plane is only possible whilst allowing for some error. Since our approach relies on a static prefabricated clustering, an obvious choice would be to incorporate the method with the smallest error margins in contrast with the method that is the most cost-effective.

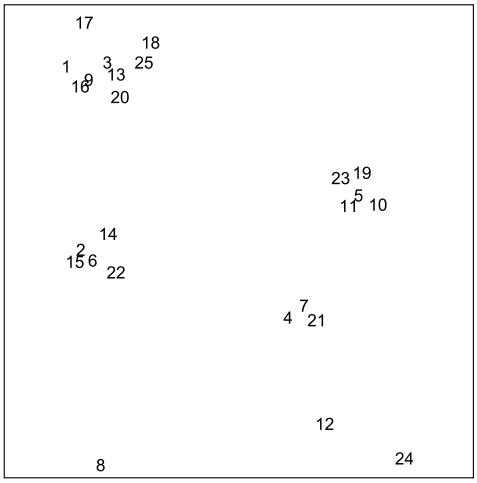

Figure 7.1: A typical clustering of 25 points visualized on a two-dimensional plane

7.2.2 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 (see Figure 7.2). 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 often 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 always exactly fits $n+1$ data points, but needs to be approximated if more than $n+1$ points are available.

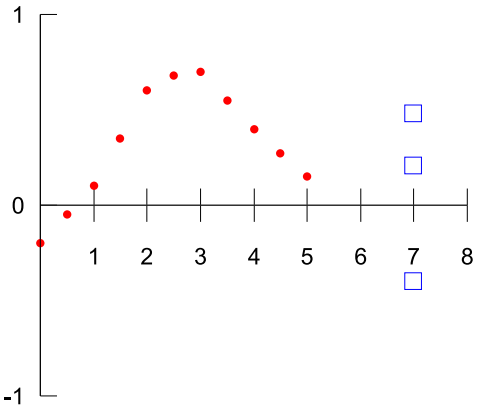

Figure 7.2: The process of trying to determine which of the boxes at time 7 best predicts the outcome based upon the known data points is called extrapolation

In such a case one needs to resort to approximation measures like the *least squares error* method [1].

There are at least two main interpolation methods that are suitable to be incorporated in our approach: *polynomial interpolation* and a *spline*.

Polynomial interpolation tries to find a function

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_2 x^2 + a_1 x + a_0$$

that satisfies all existing points $p(x_i) = y_i$ for all $i \in \{0, 1, \ldots, n\}$, leading to a system of linear equations in matrix form denoted as:

$$\begin{bmatrix} x_0^n & x_0^{n-1} & x_0^{n-2} & \ldots & x_0 & 1 \\ x_1^n & x_1^{n-1} & x_1^{n-2} & \ldots & x_1 & 1 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ x_n^n & x_n^{n-1} & x_n^{n-2} & \ldots & x_n & 1 \end{bmatrix} \begin{bmatrix} a_n \\ a_{n-1} \\ \vdots \\ a_0 \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix},$$

from now on written as $X \cdot A = Y$.

Solving this system leads to the interpolating polynomial $p(x)$ that exactly fits $n + 1$ data points. It is also possible to find a best fit for a polynomial of degree n for $m > n + 1$ data points. For that purpose we project the matrix X on the plane Π_n, the vector space of polynomials with degree n or less, by multiplying both sides with X^T, the transposed matrix of X, leading to the following system of linear equations:

$$X^T \cdot X \cdot A = X^T \cdot Y.$$

Solving this system leads to a best fit polynomial $p(x)$ that best approximates the given data points (see Figure 7.3).

Data points can also be interpolated by specifying a separate polynomial between each couple of data points. This interpolation scheme, called a *spline*, exactly fits the

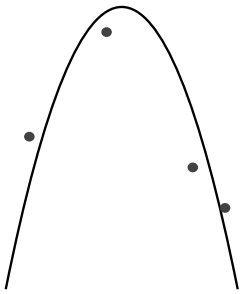

Figure 7.3: A function of degree 2 that best fits 4 data points

derivative of both polynomials ending in the same data point, or *knot*. These polynomials can be of any degree but are usually of degree 3 or *cubic* [5]. Demanding that the second derivatives also match in the knots and specifying the requested derivative in both end points yields $4n$ equations for $4n$ unknowns. Rearranging all these equations according to Bartels et al. [5] leads to a symmetric tridiagonal system, that, when solved, enables us to specify third degree polynomials for both the x and y coordinates between two separate knots, resulting in an interpolation like the graph in Figure 7.4. Due to the liberty this method allows in the placement of the existing data points, this method seems well suited for the task of two-dimensional extrapolation (see Section 7.2.2).

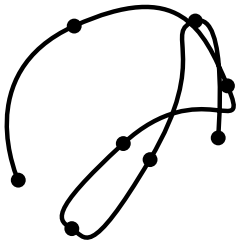

Figure 7.4: An example of a spline

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. This effect is known as the *Runge phenomenon* [35]. Since extrapolation is already much less precise than interpolation, polynomials of degrees higher than 3 are often discouraged.

In most of the cases, it is sufficient to simply continue the fabricated interpolation function after the last existing data point, like for example in Figure 7.2. 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 7.5.

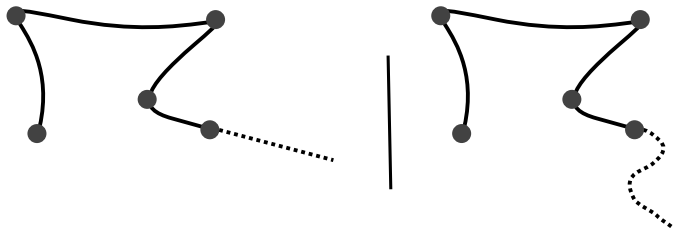

Figure 7.5: The difference between straight line extrapolation (left) and polynomial continuation (right)

Two-Dimensional Extrapolation

In our approach both x and y are coordinates and therefore inherently independent variables. They depend on the current visualization alone and have no intended meaning outside the visualized world. Within our model, they do however depend on the variable t that describes the time passed between sampling points. Because our methods aims to extrapolate the two variables x, y out of one other variable t, we need a form of two-dimensional extrapolation. The standard polynomial function always assumes one independent variable (most often x) and one dependent variable (most often y) and is therefore not very well suited to the required task. However, after rotation and under the assumption that x is in fact the independent variable guiding y, the method can be still be useful. 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 7.6.

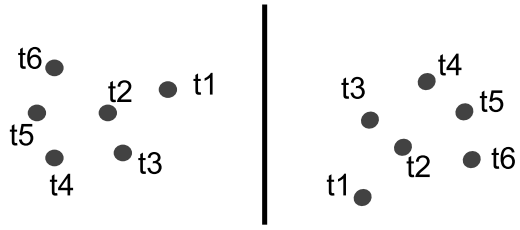

Figure 7.6: Rotation with the best left-right ordering on the x-axis. Note that events 2 and 3 remain in the wrong order

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):

$$p_{x,y}(t) = \begin{cases} x = p_1(t) \\ y = p_2(t) \end{cases}$$

Naturally, the dependence of x and y on t within the spline interpolation scheme makes that method very well suited for the task of two-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

7.3 Approach

The number of attributes describing each item in a database can be quite large. Taking all this information into account when extrapolating sequences of behavior through time can therefore be quite a hassle. Since this information is inherently present in a visualization, we can theoretically narrow the information load down to two attributes (x and y) per item whilst retaining some of the accuracy. We designed the stepwise strategy in Figure 7.7 for our new approach.

7.3.1 Distance Matrix and Static Visualization

The data used as reference within our approach is represented by a square distance matrix of size $q \times q$ describing the proximity between all q items. These items are considered to be complete in the sense 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 into a two-dimensional plane with dots representing our reference items, see, e.g., Figure 7.1. This step in the approach is done only once so the focus should be on the quality of the visualization and clustering instead of the computational complexity. From this point on this visualization is considered to be static and describing the universal domain these items live in. Note that all offenders, with a career we consider to be finished are present in this visualization.

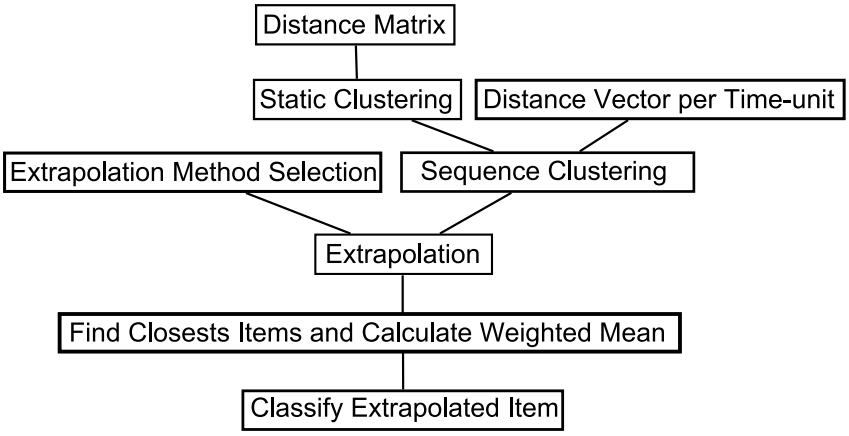

Figure 7.7: Stepwise approach

7.3.2 Distance Vector Time Frame and Sequence Clustering

Analysis of the behavior of new items should start with the calculation of the values for each time frame t. These frames are supposed to be cumulative, meaning that they contain all the item's *baggage* up to the specified moment. Using the same distance measure that was used to create the initial distance matrix, the *distance vector per time frame* can now be calculated. This should be done for all t time frames, resulting in t vectors of size q.

These vectors can now be visualized in the previously created clustering on the correct place using the same visualization technique, while leaving the original reference points in their exact locations. The chosen visualization method should naturally allow for incremental placement of individual items, e.g., as in [27]. These new data points within the visualization will be used to extrapolate the items behavior through the static visualization. because of accuracy reasons, we will only consider items for which three or more time frames are already known. Note that the first time frame will be placed in the cloud of one-time offenders with a very high probability.

7.3.3 Extrapolation

After selecting the best extrapolation scheme for our type of data our method creates a function that extrapolates item behavior. For the same data the different schemes can yield different results as illustrated in Figure 7.8, so care should be taken to select the right type of extrapolation for the data under consideration.

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 analyze 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.

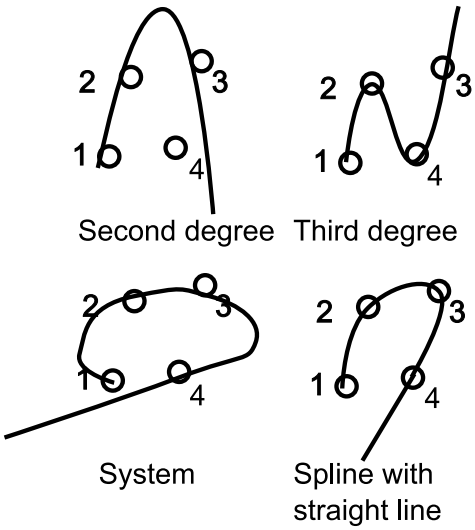

Figure 7.8: Different extrapolation methods yield very different results

7.3.4 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 a number of reference items r closest to the extrapolation curve. This can easily be done by evaluating the geometric distance of all reference points to the line and selecting r items with the smallest distance. This process can be seen in Figure 7.9.

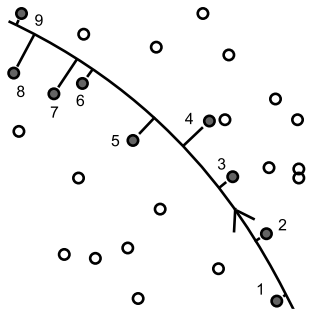

Figure 7.9: Selecting points with the shortest distance to the extrapolation line

Each of these points gets a number i assigned based upon their respective distance to the last known data point of the extrapolated item (see Figure 7.9). Because of the declining confidence of the prediction further away from that last point this number represents the amount of influence this item has on the extrapolated classification. We are now able to calculate the weighted mean value for each future attribute j according to the following formula:

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

Here, $Attrib_j(i)$ denotes the value of the j^{th} numbered attribute of i.

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

7.4 Experimental Results

The detection, analysis, progression and prediction of criminal careers is an important part of automated law enforcement analysis [10, 11]. Our approach of temporal extrapolation was tested on the criminal record database (cf. Appendix B), containing approximately one million offenders and their respective crimes

As a first step we clustered 1000 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 visualization and compared them with each other and standard extrapolation on each of the attributes (methods 7 and 8). For each item in the test set we only consider the first 4 time periods. The accuracy is described as the percentage of individuals that was correctly classified (cf. Chapter 5, compared to the total amount of individuals under observation. The results are presented in Table 7.1, where *time factor* represents how many times longer the method took to complete its calculation than the fastest method under consideration.

Although the calculation time needed for visual extrapolation is much less than that of regular methods, the accuracy is very 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 computation complexity is much higher. The simpler x,y system with third degree extrapolation has got a very low runtime complexity but still manages to reach an accuracy that is only 1.5 percentage points lower than the best performing method.

7.5 Conclusion and Future Directions

In this chapter we demonstrated the applicability of temporal extrapolation by using the prefabricated visualization of a clustering of reference items. This method assumes that the visualization of a clustering inherently contains a certain truth value that can yield

Table 7.1: Results of Static Visualization Extrapolation for the analysis of Criminal Careers

	method	*time factor*	*accuracy*
1	Second degree polynomial extrapolation	1	79.1%
2	Third degree polynomial extrapolation	1.1	79.3%
3	*x*,*y* system with second degree polynomial extrap.	1.9	81.5%
4	*x*,*y* system with third degree polynomial extrap.	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%

results just as powerful as standard sequence extrapolation techniques while reducing the runtime complexity by using only two information units per reference item, the *x*- and *y*-coordinates. We demonstrated a number of extrapolation techniques that are suitable for the extrapolation of the temporal movement of a certain item through this visualization and employed these methods to come to a prediction of the future development of this item's behavior. Our methods were tested within the arena of criminal career analysis, where it was assigned the task to predict the future of unfolding criminal careers.

We showed that our approach largely outperforms standard prediction methods in the sense of computational complexity, without a loss in accuracy larger than 1 percentage point. On top of that, 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 point and click interface, rather than presenting him with unnecessary detailed numerical results.

It is important to perform the test described in Section 7.4 for each new type of data that is subject to this approach. Different types of data might well be more susceptible to errors in one of the extrapolation methods and less in others.

Future research will aim at reaching even higher accuracy values by improving the selection of reference items close to the extrapolation line. Incorporation of this approach in ongoing research in the area of criminal career analysis should reveal the power and use of this approach in law enforcement reality and should provide a plethora of improvements to the method.

Any concerns about privacy and judicial applicability on the usage of the methods described here are discussed in Appendix A.