2.6 Measurement Errors

To be able to handle measurement error, it is necessary for the curve fitting tool to interpolate data in a very well behaved manner. GRNN has the ability to interpolate the data. In order to do that, GRNN does not use a minimization method like curve fits or B-Splines but uses a statistical approach. The necessary parameter σ has to be selected by one of the available methods. In the figures below the results for using the wiggle-method is shown. The data for the following predictions was artificially generated and a measurement error was artificially introduced. For the first figure (Fig. 2.6-1) the allowable number of wiggles was two. For (Fig. 2.6-2) the allowable number of wiggles was four. The curves below show that GRNN has an ability to interpolate data. The curve follows clearly the trend the noisy data has. The wiggle method was capable, despite the noisy data, to select a smoothness parameter for which the predicted curve still fit the data. The impact of noisy data is more thoroughly discussed in Chapter 2.

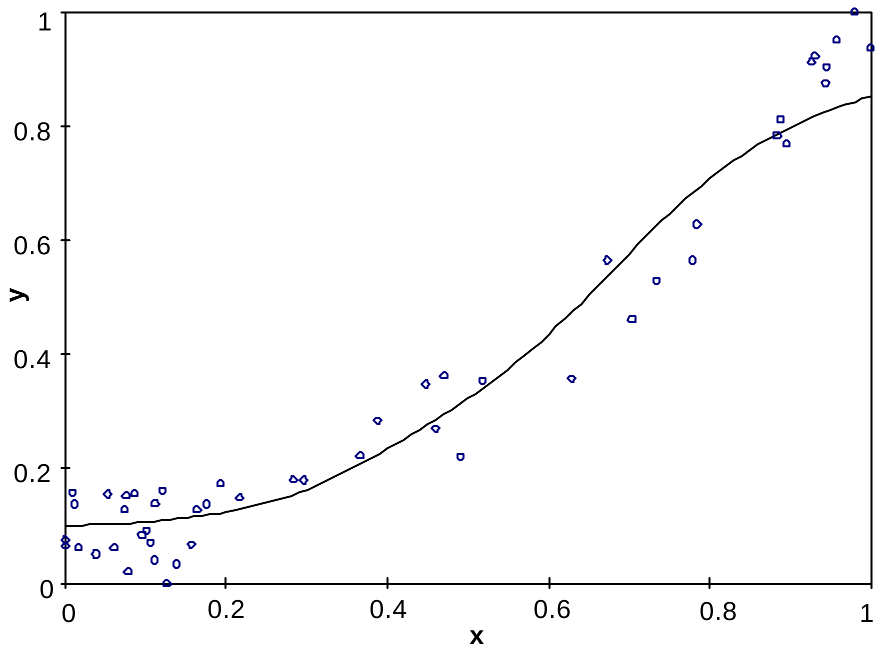

Fig. 2.6-1 GRNN prediction of data including measurement errors, 2 inflection points allowed

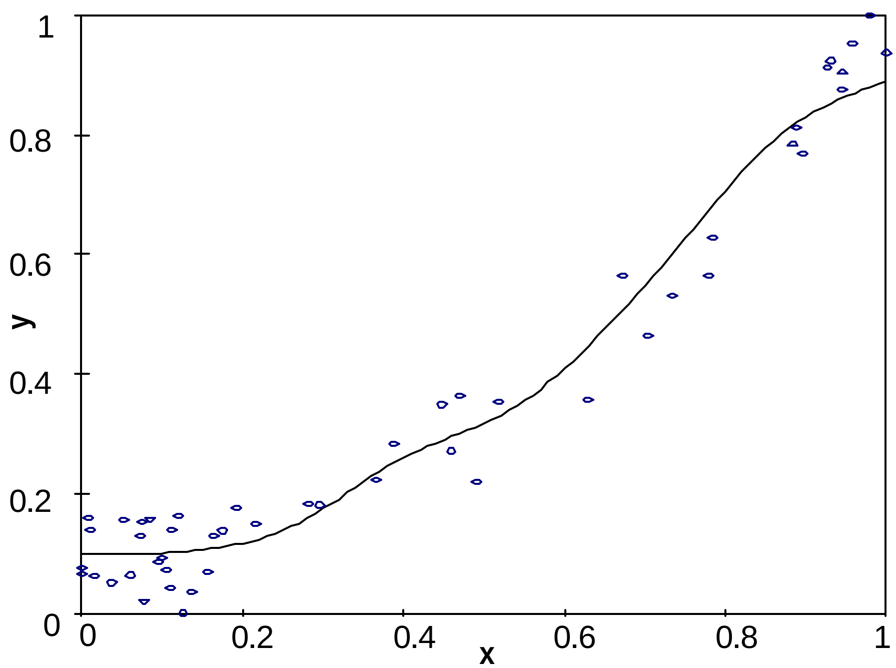

Fig. 2.6-2 GRNN prediction of data including measurement errors, 4 inflection points allowed

2.7 Extrapolation

Extrapolation is a desired ability for any curve fitting tool. Only very few methods are able to extrapolate beyond the range of available data. For the prediction in (Fig. 2.7-1) the training samples shown as circles were used. The solid line is the prediction by GRNN. GRNN fits the data in the known manner in the range where data are available. As soon as the range of available data is left, the prediction levels off and yields the value of the closest training sample.

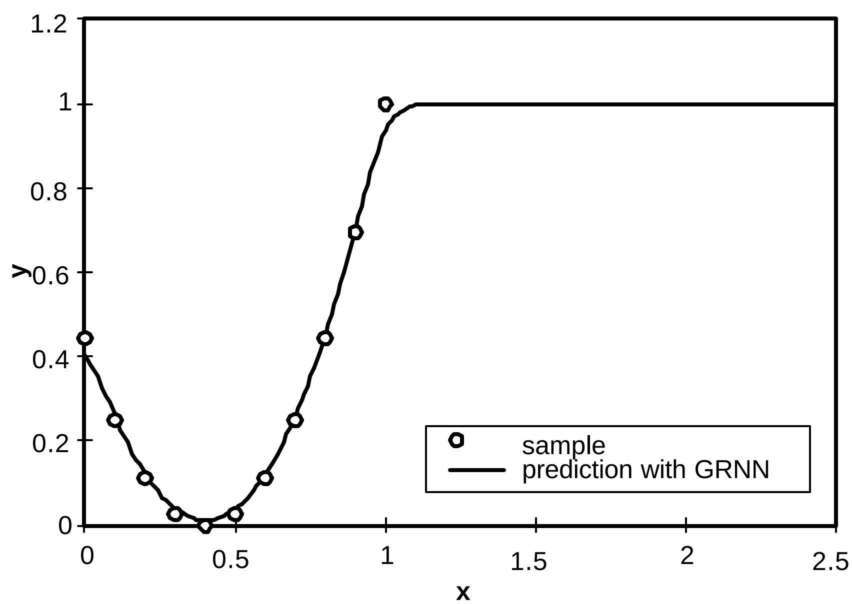

Fig. 2.7-1 Extrapolation ability of GRNN

The proof that GRNN really predicts the value of the last training sample is shown below. A sufficiently small value for σ has to be assumed for this proof. The equation used for GRNN (Eqn. 2.1-1) is examined.

$$\lim_{\substack{s=const \\ X \to \infty}} (\boldsymbol{Prediction}) = \lim_{\substack{s=const \\ X \to \infty}} \left(\frac{\sum_{i=1}^{n} Y_i \, \boldsymbol{exp}\left({-D_i^2}/{2s^2} \right)}{\sum_{i=1}^{n} \boldsymbol{exp}\left({-D_i^2}/{2s^2} \right)} \right)$$

Eqn. 2.7-1

with (Eqn. 2.2-1) this means for an approximation for a small σ that only the last data point, in this case the data point for the biggest value of X_i, has influence on the prediction such that (Eqn. 2.7-1) becomes

$$= \lim_{\substack{s=const \\ X \to \infty}} \left(\frac{Y_j \, \boldsymbol{exp}\left({-D_j^2}/{2s^2} \right)}{\boldsymbol{exp}\left({-D_j^2}/{2s^2} \right)} \right) = Y_j$$

Eqn. 2.7-2

This equation (Eqn. 2.7-2) states that GRNN will continue to predict the value of the training sample with the biggest value for X. Respectively GRNN will predict the value of the last training sample as the range of training samples is left to the other side of the range for available training samples.

2.8 Underlying function

A method that better accommodates all the needs for both precision **and** smoothness is the use of an underlying function. The goal is to introduce two procedures (Fig. 2.8-1) to work together as a team, using the abilities of each one of them to come to a better solution as a whole. One of the procedures is GRNN, the other one is incorporate knowledge about the data available. This knowledge will be used to give the trend; the direction for the fit. The knowledge could be a simplified model or equation that is available for many components in practice. The difference between this trend and the available training data will be corrected by GRNN.

A simple model is very often known for a complicated problem. The use of this solution will not represent the problem correctly and a certain error is made. Depending on the conditions the error varies. This error shall be corrected. Since the simple model is not completely wrong, the simple model will support the prediction.

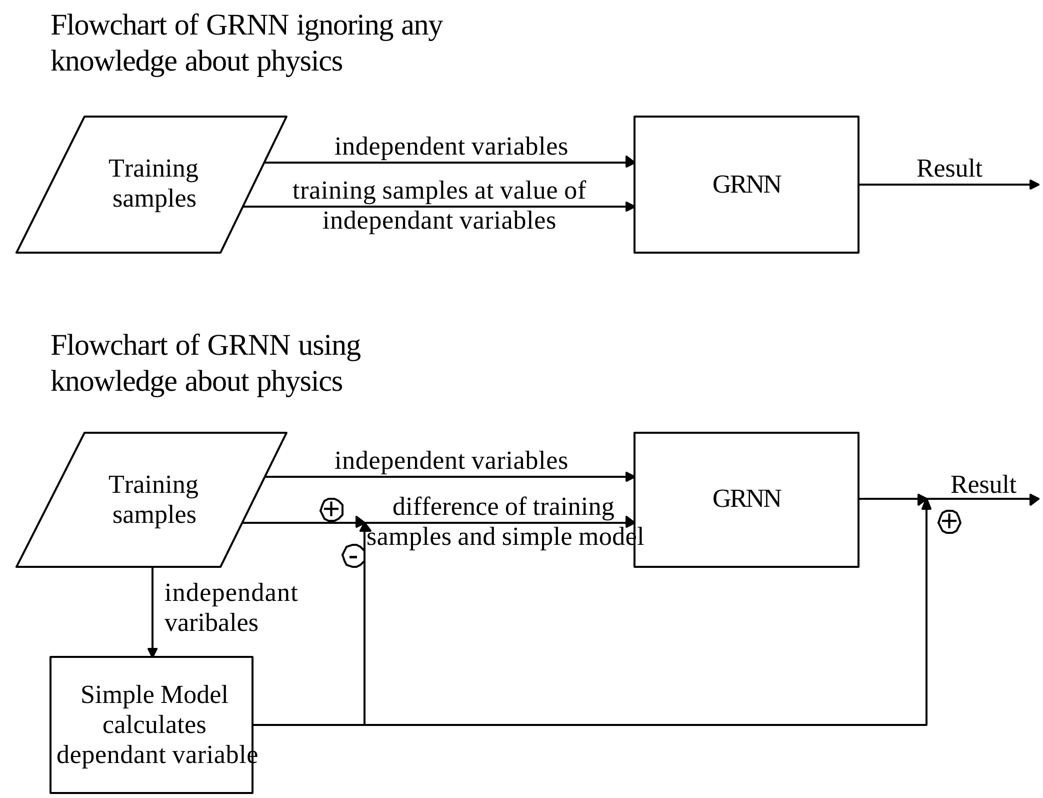

Fig. 2.8-1 Different prediction methods, using only training samples and using an underlying function in addition to training samples

GRNN will still have the same kind of problems as before with extreme values, unequally spaced data, multi-dimensions and extrapolation. The impact of these problems will be smaller because the error that is made is on a smaller scale compared to the case when GRNN predicts results without any trend.

In the following examples the use of an underlying function compared to a prediction without an underlying function is compared. The training data for the case not using an underlying function is again represented by circles. The prediction used again a value for σ that resulted out of a search using the wiggle method. The prediction shows as before problems for extreme values of the prediction.

The approach using an underlying function only uses the difference between the underlying function and the training samples for training. GRNN then only predicted a difference between the underlying function and the final result. The final result is a summation of the simple model and the difference predicted by GRNN.

GRNN had previously problems to predict extreme values. The prediction in (Fig. 2.8-2) using only GRNN still shows this problem, as seen at the values for x=0 or x=1.0. The use of the underlying function supports the prediction for these values. The problems at the edges becomes less and the problem with the extreme point at x=0.5 is reduced too.

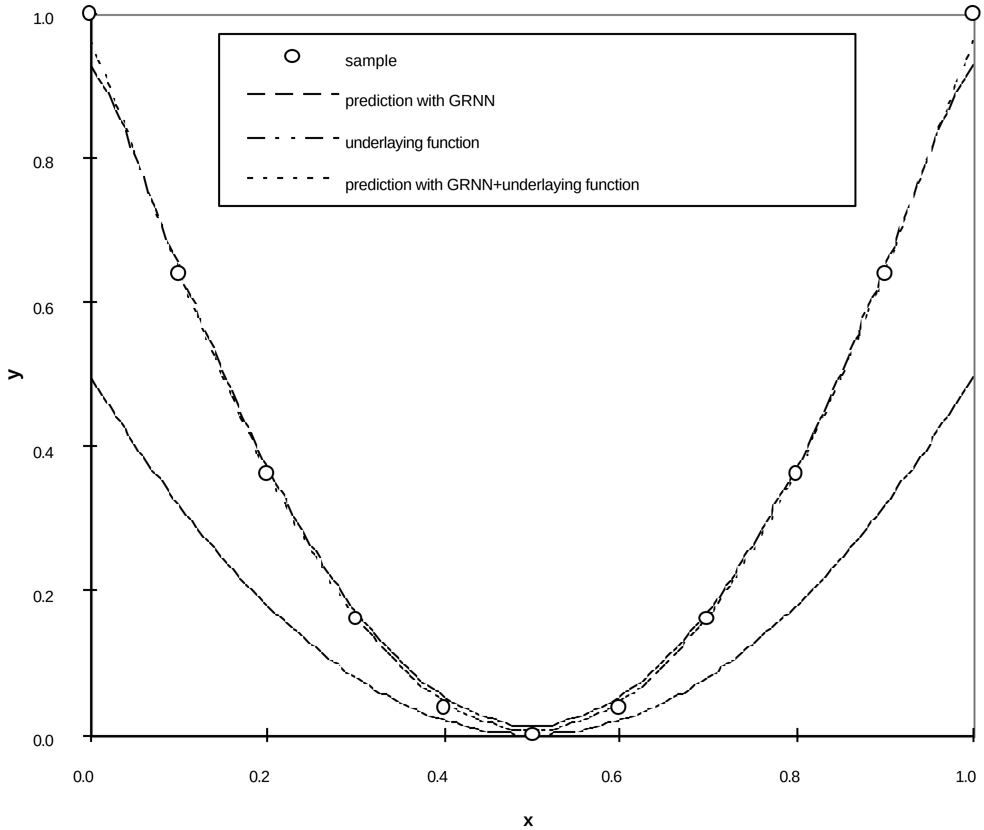

Fig. 2.8-2 Influence of an underlying function on the extreme values

If the data are non-equally spaced, the use of an underlying function minimizes the influence of the problems GRNN has. The use of an underlying function is not a cure of the problem, rather it is a cure of a symptom (Fig. 2.8-3, Fig. 2.8-4). In Section 3.4 the influence of using an underlying function on the quality of the prediction using data is thoroughly studied on a technical example.

In (Fig. 2.8-3, Fig. 2.8-4) the example that was shown in Section 2.2.3 is used again here. The training samples are again shown as circles. The underlying function is shown too. The training and the prediction are performed as it was described for the example using equally spaced data shown in (Fig. 2.8-2). The difference between the prediction using no underlying function and the one using the underlying function is bigger than for the example of equally spaced data. The prediction using the underlying function performs better than the one without the underlying function. The prediction is more precise for the edges and as well for the minimum at x=0.5.

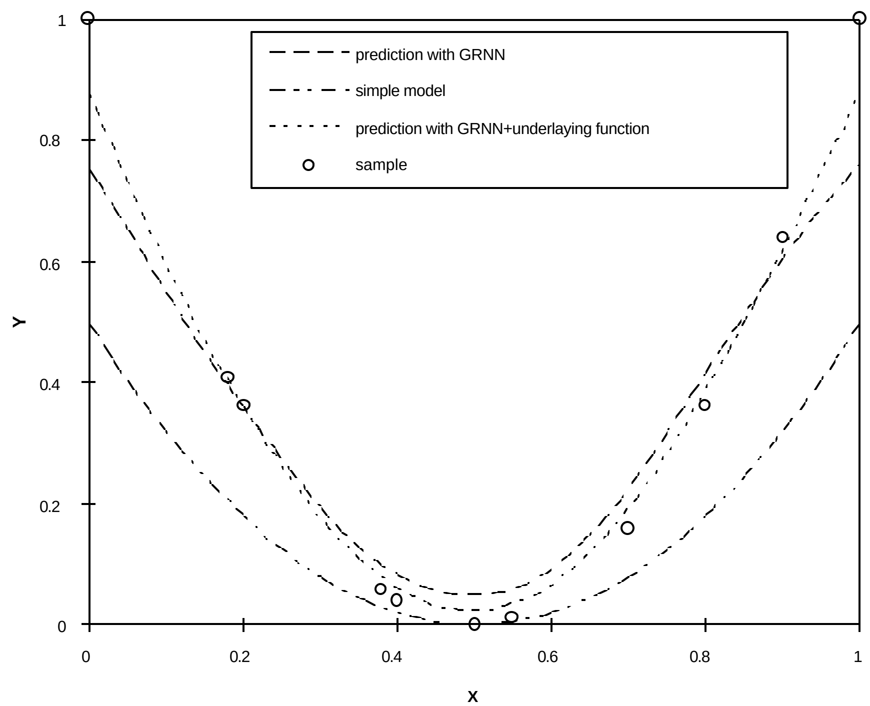

Fig. 2.8-3 Influence of an underlying function on the effects of unequally spaced data, two inflection were allowed

The following example allowed four inflection points in the wiggle method. The difference in the prediction between the two approaches using an underlying function or not can be again seen in the more accurate prediction for extreme values. Another difference that can be seen in (Fig. 2.8-4) is that the additional inflections at x=0.2 and x=0.4 are not as distinct for the prediction using the underlying function as for the prediction not using the underlying function.

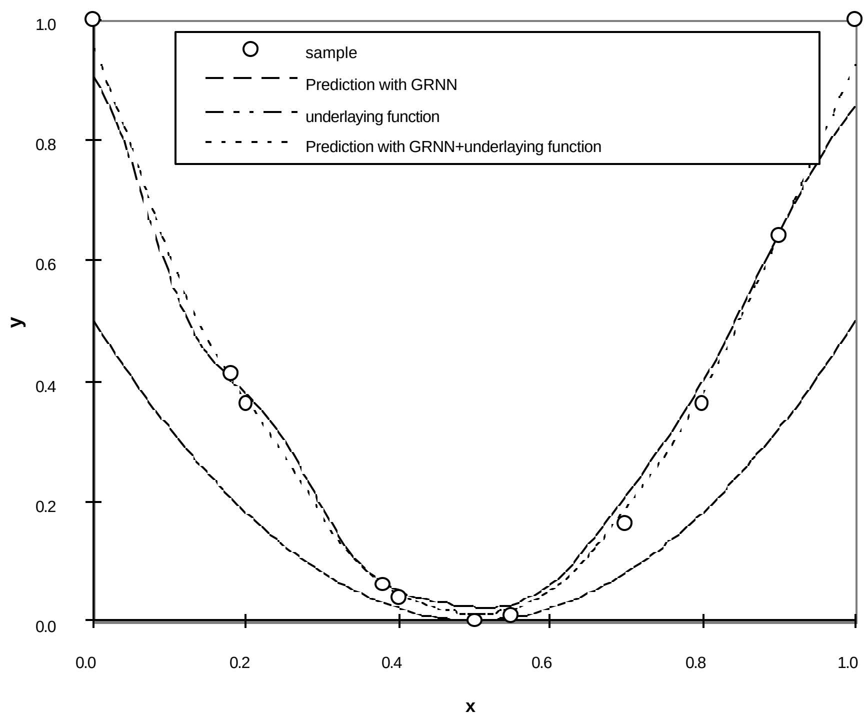

Fig. 2.8-4 Influence of an underlying function on the effects of unequally spaced data, four inflections were allowed

The choice of the smoothness parameter is no longer that sensitive as for the approach using no underlying function. The problems that GRNN has are cushened by the use of the underlying function such is the selection of the smoothness parameter. I all the previous examples the smoothness parameter was chosen with the wiggle-method. The same emphasis was put on the smoothness as for all the other examples.

2.9 Chapter Summary

The selection of the smoothness parameter is the critical step for a good prediction with GRNN. The method Specht suggests, the holdout method, does not perform as desired. The holdout method simply optimizes for precision and ignores an important issue of smoothness. The wiggle method allows the user to decide what is more important, smoothness or precision. It can be adjusted such that it accommodates the needs for the individual application. The wiggle method is more flexible than the holdout method.

GRNN has several limitations. One very severe problem is the prediction for extreme values. With the use of a normal distribution in GRNN, GRNN becomes sensitive to unequally spaced data. This is a very limiting aspect. Another problem is that GRNN with one parameter cannot accommodate different characteristics of the individual dimensions. Measurement errors are very easy to cope with. The wiggle method gives enough room to adjust sigma such that measurement errors are not a big problem. Extrapolations using GRNN and no simple model are not possible. The predicted value after leaving the range of the available data is the value of the closest training sample to the point of prediction. A constant value will be predicted for the correction when a underlying function is used to support the prediction.

Despite the mentioned problems, GRNN is still a very powerful tool, but it needs a lot of attention and design of GRNN depending on the context and the use of the prediction.