Multivariate polynomial regression regularized via fit function uncertainty

Péter Kövesárki[1] **and Ian C Brock**[2]

[1]University of Wroclaw, Poland
[2]University of Bonn, Germany

E-mail: Peter.Koevesarki@cern.ch

Abstract. This article describes a multivariate polynomial regression method where the uncertainty of the input parameters are approximated with Gaussian distributions, derived from the central limit theorem for large weighted sums, directly from the training sample. The estimated uncertainties can be propagated into the optimal fit function, as an alternative to the statistical bootstrap method. This uncertainty can be propagated further into a loss function like quantity, with which it is possible to calculate the expected loss function, and allows to select the optimal polynomial degree with statistical significance. Combined with simple phase space splitting methods, it is possible to model most features of the training data even with low degree polynomials or constants.

1. Introduction

Regression methods are frequently used in particle physics, usually to quantify a continuous curve or surface that simplifies a statistical sample. Typical examples are the calibration curves for certain detector responses and neural networks trained to identify particle. The mathematical goal is finding a $f : x \to y$ map between the x input space to the y target space in such a way that $f(x)$ predicts the $\mathbb{E}(y|x)$ conditional expectation value with statistical certainty. The least squares algorithm is known to converge to the conditional mean, given it is a finite number and the parametric f function is in the family that contains the solution. This latter information is not always given and one must chose a function family general enough to cover unexpected features. Such a function family are the logistic functions and the radial base functions and generally the *kernels*. For these one usually has to determine an ideal degree of freedom for the fit, namely the number of base functions to be used in order to avoid overtraining of the data and so avoiding picking up non-significant features from the statistical fluctuation. Although these are straightforward procedures, it is computationally intensive to find the global minimum of the sum of squares for the fit. A usually unexploited feature of the least squares method is that the global minimum can be exactly determined for kernels with fixed position in the x space, because the amplitude that minimizes the sum of squares can be calculated with a linear equation, without numerical optimization.

With given $k_i(x), i \in 1..n_d$ kernels and a_i amplitudes the sum of squares for the data points $\{x_j, y_j\}, j \in \{1..N\}$ will take the form

$$E_{\chi^2} = \frac{1}{N}\sum_j \left(y_j - \sum_i a_i k_i\left(x_j\right)\right)^2 = \langle y^2\rangle - \sum_i 2a_i\langle y k_i(x)\rangle + \sum_{il} a_i a_l \langle k_i(x) k_l(x)\rangle\,, \qquad (1)$$

where the angled brackets $\langle\rangle$ indicate averaging over the sample. The *loss function* E_{χ^2} in eq. (1) can be minimized in respect of the f_i amplitudes with

$$f_i = \sum_l G^{-1}_{il} h_l \qquad (2)$$

by using the matrix $G_{il} = \langle k_i(x) k_l(x)\rangle$ and the vector $h_i = \langle y k_i\rangle$. The G_{il} matrix is symmetric and has $\frac{1}{2}n_d(n_d-1)$ parameters. A possible way to decrease the number of parameters is to use kernels which are power series $k_i(x) = k_1^i(x)$, resulting in a Hankel-type matrix $G_{il} = \langle k_1(x)^{i+l}\rangle$. A simple power series kernel might be based on the monomials, $k_1(x) = x, k_i(x) = x^i$, resulting in polynomial fitting. Another advantage of fitting a polynomial with a fixed degree instead of Gaussian or sigmoid kernels is that polynomials are not sensitive to the shift of features in the data, in other words they are translation invariant.

2. Uncertainty and covariance of large weighted random sums

The advantage of using the matrix formalism in eq. (1) is that the original training data $\{x_j, y_j\}$ is compressed into the h_i vector and the G_{il} matrix, which can be a great reduction in the number of input parameters. These input parameters are themselves random variables having a certain distribution that in principle could be derived from the generating distribution of the $\{x_j, y_j\}$ sample and the number of measurements N. Due to the central limit theorem, the generating distribution itself does not need to be known, however it has to fulfill certain criteria. Probably the most widely known of the central limit theorems is the one stating that if the generating distribution of the $X_i, i \in \{1..N\}$ random variables have the finite mean $\bar{X}$ and variance σ_X, then the distribution of the variable $s = \frac{1}{\sigma_X}(\sum_i X_i/N - \bar{X})$ converges to the normal distribution. With small modifications this theorem is applicable to h_i and G_{ij}.

The approximation of the covariance matrix of the $p_m = (h_1, ..., h_{n_d}, g_1, ..., g_{2n_d}), m \in \{1, ..., 3n_d\}$ input parameters for polynomial regression is the following. In a general formalism, every data point with index j is a triplet, consisting of an input value x_j, a target value y_j and a weight w_j. With $b \in \{0,1\}$, $k_0\{1, ..., n_d\}$, $k_1 \in \{1, ..., 2n_d\}$ the input parameter with pseudo index $m = k_b + bn_d$ is calculated as

$$p_m = \frac{1}{\sum_j w_j}\sum_j w_j y_j^b x_j^{k_b}\,.$$

The product $a_j = w_j y_j^b x_j^{k_b}$ for each i index can be treated as a compound random variable. The $1/\sum_j w_j$ was not introduced into this new variable, because that would ruin the independence between the variables. To estimate the probability distribution of p_m the following variables are defined. Let $\langle aw\rangle = \frac{1}{N}\sum_j w_j a_j$ be the weighted average of a_j. This is approximately a Gaussian variable with mean $\langle aw\rangle$ and variance $\sigma^2_{\langle aw\rangle} = \frac{1}{N(N-1)}\sum_j (a_j w_j - \langle aw\rangle)^2$. Define the average weight similarly as $\langle w\rangle = \frac{1}{N}\sum_j w_j$, which is also distributed as a Gaussian with mean $\langle w\rangle$ and variance $\sigma^2_{\langle w\rangle} = \frac{1}{N(N-1)}\sum_j (w_j - \langle w\rangle)^2$. These variables are indeed correlated, and their covariance is $\mathrm{Cov}\left(\langle aw\rangle, \langle w\rangle\right) = \frac{1}{N(N-1)}\sum_j (a_j w_j - \langle aw\rangle)(w_j - \langle w\rangle)$. The above definitions show, that p_m is a ratio of two Gaussian variables. Its expectation value is p_m, while its variance can be approximated with error propagation. Assuming that $\sigma_{\langle w\rangle}/\langle w\rangle \gg 0$, the approximation of the variance of p_m is

$$\sigma^2_{p_m} = \begin{pmatrix} \frac{\partial p_m}{\partial \langle aw \rangle} \\ \frac{\partial p_m}{\partial \langle w \rangle} \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \sigma^2_{\langle aw \rangle} & \mathrm{Cov}\left(\langle aw \rangle, \langle w \rangle\right) \\ \mathrm{Cov}\left(\langle aw \rangle, \langle w \rangle\right) & \sigma^2_{\langle w \rangle} \end{pmatrix} \begin{pmatrix} \frac{\partial p_m}{\partial \langle aw \rangle} \\ \frac{\partial p_m}{\partial \langle w \rangle} \end{pmatrix}$$

$$= \frac{1}{\langle w \rangle^2} \sigma^2_{\langle aw \rangle} + \frac{\langle aw \rangle^2}{\langle w \rangle^4} \sigma^2_{\langle w \rangle} - 2 \frac{\langle aw \rangle}{\langle w \rangle^3} \mathrm{Cov}\left(\langle aw \rangle, \langle w \rangle\right) \tag{3}$$

$$= \frac{1}{\langle w \rangle^2} \sum_j \frac{w_j^2 (\langle aw \rangle - a_j)^2}{N(N-1)} \,. \tag{4}$$

It can be seen that eq. (4) gives back the known formula for the standard deviation of $\langle a \rangle$ when all weights are $w_j = 1$, which is also true when the weights are independent of the distribution of a. A similar derivation shows that the covariance between variables p_{m_1} and p_{m_2}, with pseudo-indices $m_1 = k_{b_1} + b_1 n_d$ and $m_2 = k_{b_2} + b_2 n_d$ can be calculated as

$$\mathrm{Cov}\left(p_{m_1}, p_{m_2}\right) = \frac{1}{\langle w \rangle^2} \sum_j \frac{w_j^2 (\langle w y^{b_1} x^{k_{b_1}} \rangle - y_j^{b_1} x_j^{k_{b_1}})(\langle w y^{b_2} x^{k_{b_2}} \rangle - y_j^{b_2} x_j^{k_{b_2}})}{N(N-1)} \,. \tag{5}$$

The covariance and the variance estimation can be generalized to non-monomial kernels, by replacing x^k in the above equations with the given kernel.

It must be noted that traditionally the uncertainty estimates on the fit parameters have different formulas. In most cases the sample $\{x_j, y_j\}$ is augmented with the uncertainty of the target, the conditional variance in the y direction, $\sigma^2_{y_j}$, which can be considered as prior knowledge. In that case the data points in the formation of the estimation of expectation values receive a $\sigma^{-2}_{y_j}$ weight and a $c = \sum_j \sigma^{-2}_{y_j}$ normalization factor. This choice of weight comes from the principle that the different measurements should be combined in a way that minimizes the uncertainty of the result, in this case the estimation of the expectation values. An example could be the $F(x) = a + bx$ least squares regression with a, b unknowns on the $\{x_j, y_j, \sigma^2_{y_j}\}$ sample. The input parameters to this fit are

$$c = \sum_j \frac{1}{\sigma^2_{y_j}} \,, h_0 = \langle y \rangle = \frac{1}{c} \sum_j \frac{y_j}{\sigma^2_{y_j}} \,, h_1 = \langle yx \rangle = \frac{1}{c} \sum_j \frac{y_j x_j}{\sigma^2_{y_j}} \,,$$

$$g_0 = \langle x^0 \rangle = 1 \,, g_1 = \langle x \rangle = \frac{1}{c} \sum_j \frac{x_j}{\sigma^2_{y_j}} \,, g_2 = \langle x^2 \rangle = \frac{1}{c} \sum_j \frac{x_j^2}{\sigma^2_{y_j}} \,.$$

The optimal fit parameters are

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} g_0 & g_1 \\ g_1 & g_2 \end{pmatrix}^{-1} \begin{pmatrix} h_0 \\ h_1 \end{pmatrix} \,.$$

As a and b are linear functions of y_j, it is easy to calculate the expectation values needed for the covariance matrix:

$$\mathrm{Cov}\left(a, b\right) = \mathbb{E}(ab) - \mathbb{E}(a)\mathbb{E}(b) = \frac{1}{c} \sum_j \frac{\partial a}{\partial y_j} \frac{\partial a}{\partial y_j} \sigma^2_{y_i} = \langle x \rangle = g_1 \,. \tag{6}$$

Similarly, $\mathrm{Cov}\left(a, a\right) = \sigma^2_a = g_0$ and $\mathrm{Cov}\left(b, b\right) = \sigma^2_b = g_2$. This covariance matrix indeed differs from the one derived in eq. (5). The origin of the difference relies in the prior information that was built into the equations. In the case of eq. (5) the weights were provided with the data points, while in the case of eq. (6) the $\sigma^2_{y_i}$ was given – knowledge of the uncertainty on the $\mathbb{E}(y|x)$

conditional mean. The weights in the former case may come from Monte Carlo integration techniques or from weighted sample separation and it is thought to be fundamentally fixed, while in the latter case it is derived from the principle of optimal data combination. Though it is unclear whether the two methods could be combined, the former method is thought to be superior as it was designed to approximate the $\sigma^2_{y(x)}$ from the sample itself, and also takes into account the uncertainty in the sampling of the input space x. Furthermore, it handles negative and zero weights correctly.

3. Fit function uncertainty

With the knowledge of the uncertainty of the input parameters $p_m = (h_1, ..., h_{n_d}, g_1, ..., g_{2n_d}), m \in \{1, ..., 3n_d\}$, one can estimate the uncertainty of the fitted kernel amplitudes using linear error propagation in eq. (2). The first derivatives of $f_j = \sum_l G^{-1}_{il} h_l, G_{il} = g_{i+l}$ are

$$\frac{\partial f_i}{\partial \langle h_l \rangle} = G^{-1}_{il} \,,$$

$$\frac{\partial f_i}{\partial \langle g_o \rangle} = -\sum_{lmn} G^{-1}_{im} \frac{\partial G_{mn}}{\partial g_o} G^{-1}_{nl} h_l \,.$$

Together with the previously calculated covariance matrix, the uncertainty of the fit function $F(x) = \sum_i f_i x^i$ at a given x point is

$$\sigma^2_{F(x)} = \sum_{nm} \frac{\partial \sum_i f_i x^i}{\partial p_m} \mathrm{Cov}\left(p_m, p_n\right) \frac{\partial \sum_l f_l x^l}{\partial p_n} \,.$$

The uncertainty of the fitted function does not necessarily cover the true $\mathbb{E}(y|x)$ conditional mean. That only happens if the fit function is general enough to describe all the features of the sample. The meaning of this uncertainty is deeply routed in the central limit theorem. When the central limit theorem was applied to the p_m input parameters, only the fact that the distribution of certain sums can be modeled with a Gaussian distribution came from the theorem, the width and the mean of this Gaussian came from a maximum likelihood fit. This is typically interpreted as a posterior distribution for the true mean, but it can also be interpreted as a model fitted to the sample and predicting where the sum may converge with additional data points. The same can be said about the Gaussian uncertainty of the fit function. It tells us the likelihood where the fit function with the same degrees of freedom would converge with additional data, but not the position of the conditional mean. This is why some methodology is needed to compare fit functions with different degrees of freedom and to find out which method describes better the sample.

4. The uncertainty of the loss function

In the case of polynomial fitting one has to determine the degree of the polynomial that is still statistically meaningful. Using too many degrees of freedom in a fit can result in overfitting or eventually in the interpolation of the data points. In the latter case the E_{χ^2} loss function simply reaches its absolute minimum, zero. However, the uncertainty of the fitted function increases with the number of degrees of freedom and this can be exploited in order to select significant features only, though one has to keep in mind that the Gaussian approximation of the distribution of the p_m input parameters has a limitation. First, the Gaussian approximation is only true if the number of input points N is large enough. Second, the uncertainty of the estimated covariance matrix may also increase to be comparable with the covariance matrix itself if the number of degrees of freedom in the fit is comparable to the number of sample points.

The naïve way of comparing the optimized $F_{d_1}^{\rm opt}(x) = \sum_{ik}^{d_1} h_i G_{d_1,ik}^{-1} x^k$ with degrees of freedom d_1 to $F_{d_2}^{\rm opt}(x)$ with degrees of freedom d_2 would be by calculating their loss functions $E_{d_1,\chi^2} = \frac{1}{N}\sum_j (y_i - F_{d_1}^{\rm opt}(x))^2$ and E_{d_2,χ^2} and checking whether their difference is significantly different from zero. This procedure does not work, as the approximate distribution of E_{d_1,χ^2} is not a good measure of fit quality after the $F_{d_1}^{\rm opt}(x)$ is substituted. This can be observed on the example where the x space is thought to be non-random and only the y coordinates of the sample points can vary. After the substitution of $F_d^{\rm opt}(x)$ into E_{d,χ^2} the calculation can be simplified to

$$\begin{aligned} E_{d,\chi^2} &= \langle y^2 \rangle - 2\sum_i F_{d,i}^{\rm opt} h_i + \sum_{ik} F_{d,i}^{\rm opt} G_{ik} F_{d,k}^{\rm opt} \\ &= \langle y^2 \rangle - \sum_{ik} h_i G_{ik}^{-1} h_j \,, \end{aligned}$$

which has an approximate expectation value

$$\begin{aligned} \mathbb{E}(E_{d,\chi^2}) &= \langle y^2 \rangle - \mathbb{E}(\sum_{ik} h_i G_{ik}^{-1} h_j) \\ &= \langle y^2 \rangle - \sum_{ik} h_i G_{ik}^{-1} h_j - \sum_{ik} G_{ik} {\rm Cov}\left(h_i, h_k\right) \,. \end{aligned}$$

It contains a bias term compared to E_{d,χ^2} and allows $\mathbb{E}(E_{d,\chi^2})$ to be lower than E_{d,χ^2}. This bias means that the approximate distribution of E_{d,χ^2} is not related to the goodness of fit anymore, but to the possible E_{d,χ^2} minima.

Ideally, the best measure of fitness would be a distance-like variable between the fit function and the real $\mathbb{E}(y|x)$ conditional expectation value. A good approximation to that is of course the loss function applied to $F_d^{\rm opt}(x)$, but one must differentiate it from the previously described E_{d,χ^2}. In this picture, one must treat the sample as an approximation to $\mathbb{E}(y|x)$, and not as a random variable. Let us call it the cross validation loss function, where only the $F_d^{\rm opt}(x)$ is a random variable:

$$E_{d,\chi^2}^{+} = \underbrace{\langle y^2 \rangle}_{\rm fixed} - 2\sum_i \underbrace{F_{d,i}^{\rm opt}}_{\rm varied} \underbrace{h_i}_{\rm fixed} + \sum_{ik} \underbrace{F_{d,i}^{\rm opt}}_{\rm varied} \underbrace{G_{ik}}_{\rm fixed} \underbrace{F_{d,k}^{\rm opt}}_{\rm varied} \,.$$

As the first derivative of $\frac{\partial E_{d,\chi^2}^{+}}{\partial p_m'} = 0$ at $p_m' = \mathbb{E}(p_m)$, the expectation value of E_{d,χ^2}^{+} can be approximated through its second derivative as

$$\begin{aligned} \mathbb{E}(E_{d,\chi^2}^{+}) &= \langle y^2 \rangle - 2\sum_i h_i \mathbb{E}(F_{d,i}^{\rm opt}) + \sum_{ik} G_{ik} \mathbb{E}(F_{d,i}^{\rm opt} F_{d,k}^{\rm opt}) \\ &= E_{d,\chi^2}^{+} + \frac{1}{2}\sum_{lm} \frac{\partial^2 E_{d,\chi^2}^{+}}{\partial p_m \partial p_l} {\rm Cov}\left(p_m, p_l\right) \,, \end{aligned} \tag{7}$$

where the second derivative of E_{d,χ^2}^{+} was taken at the expectation values of p_m, and can be expressed as a block matrix

$$\frac{\partial^2 E_{d,\chi^2}^{+}}{\partial p_m \partial p_l} = \begin{pmatrix} 2G_{lm}^{-1} & -4\sum_{kno} G_{lk}^{-1} \frac{\partial G_{kn}}{\partial p_m} G_{no}^{-1} h_o \\ -4\sum_{kno} G_{mk}^{-1} \frac{\partial G_{kn}}{\partial p_l} G_{no}^{-1} h_o & 6\sum_{knowqg} h_k G_{ko}^{-1} \frac{\partial G_{ow}}{\partial p_l} G_{wg}^{-1} \frac{\partial G_{gn}}{\partial p_m} G_{nq}^{-1} h_q \end{pmatrix} \,.$$

Since E_{d,χ^2}^{+} is a quadratic formula and only the fit function is varied, the expectation value cannot be lower than the minimum taken at $F_d^{\rm opt}(x)$ by construction. In most cases the second

degree Taylor polynomial approximation in eq. (7) in the p_m variables is enough as it is quadratic in the h_i and the dependence on the G_{ik}^{-1} is usually weak for large samples. The explanation is, that G_{ik} depends on the sampling in x which does not affect the $\mathbb{E}(y|x)$ conditional expectation value in the large sample size limit. Therefore for large sample sizes and approximately error-free G_{ik} matrices $\frac{\partial^2 E^+_{d,\chi^2}}{\partial p_m \partial p_l}$ can be approximated with the its upper left sub-matrix, that represents the derivatives of h_l and h_m. In this approximation E^+_{d,χ^2} is a χ^2 variable with d degrees of freedom, given that the covariance matrix $\mathrm{Cov}\,(h_i, h_k)$ is non-singular.

In the full calculation of the standard deviation of E^+_{d,χ^2} one should include the variable $\langle y^2 \rangle$ and its correlation to the other input parameters p_m. Nevertheless, the $\langle y^2 \rangle$ term disappears when a loss function difference is calculated for two different fit functions, and it can be shown that the variance of the loss function differences can be calculated solely from the variance of the individual E^+_{d,χ^2} terms, neglecting the contributions from $\langle y^2 \rangle$. This type of variance is twice the square of the previously described bias term in the expectation value in eq. (7), as expected for a χ^2 distribution:

$$\sigma^2_{d,E^+} = 2\left(\frac{1}{2}\sum_{lm}\frac{\partial^2 E^+_{d,\chi^2}}{\partial p_m \partial p_l}\mathrm{Cov}\,(p_m, p_l)\right)^2 .$$

Similar to the behavior of χ^2 differences, the expectation value of the $E^+_{d2,\chi^2} - E^+_{d1,\chi^2}$ difference is the difference of their expectation values. The same can be said about the variance of $E^+_{d2,\chi^2} - E^+_{d1,\chi^2}$, as it can be calculated from the square root of the individual E^+_{d2,χ^2} variations:

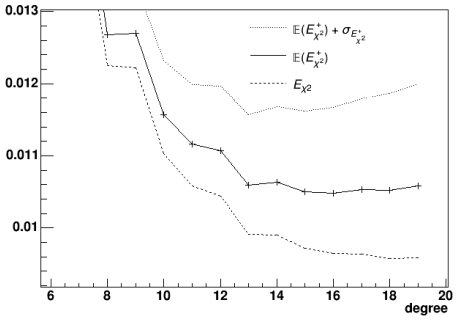
$$\sigma_{d_1,d_2,\Delta E^+} = \left|\sigma_{d_2,E^+} - \sigma_{d_1,E^+}\right| .$$

This additional rule makes it extremely simple to find the optimal degrees of freedom d, as there is no need to calculate covariance matrices belonging to the different fit functions. This makes the significance of a set of monomial kernels not only relatively true, but absolute. To optimize the degree of freedom d with a certain significance level, it is enough to minimize $\mathbb{E}(E^+_{d,\chi^2}) + s\sigma_{d,E^+}$, where s can be arbitrarily chosen. Due to the characteristics of the χ^2 distribution, optimizing for the minimal expected E^+_{d,χ^2} results in selecting features, a set of kernels that are at least 50% significant.

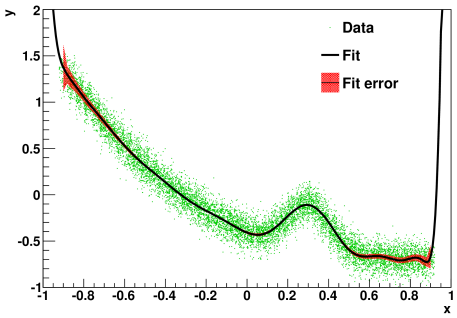
5. Determining the right polynomial order

Selecting the right modeling function can be understood as a repeated hypothesis testing. One must decide *a priori* about the null hypothesis and the series of hypotheses to test. In case of fitting a polynomial function, the ordering seems trivial, going from a constant to higher polynomial degree. However, with multivariate input x_μ, a polynomial can be defined with a different degree belonging to each μ index. In this case, one may still choose a common degree, as the optimizing method described in the previous section allows the comparison of fit functions differing in multiple degrees of freedom. A common polynomial order is special in the sense that it is closed under the group of rotations and translations, treating the different μ directions equally.

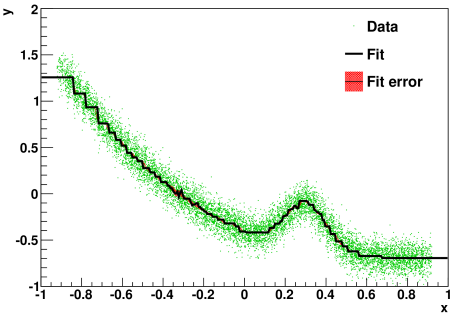
However, it is not possible to compare the loss functions of an infinite number of polynomials. Not only because it is impracticable, but also because one must stop before the numerical errors exceed the estimated uncertainties in $E^+_{\chi^2}$. It is not straightforward to estimate these numerical errors, but the rule of thumb is, that it increases with both the number of polynomial degrees and the number of input dimensions in the multivariate case. The polynomial degree necessitates to calculate high powers of the input parameters, which both appears in the training and in



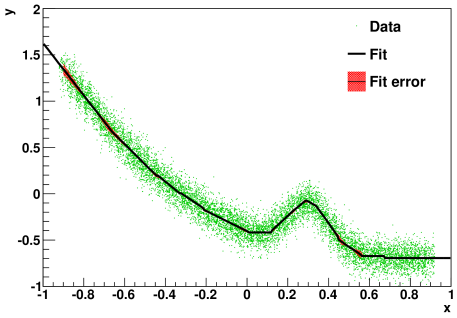
(a) dependence of the $\mathbb{E}(E^+_{\chi^2})$ expected loss function on the degrees of freedom



(b) regression with a single 16 degree polynomial



(c) regression with constants



(d) regression with linear functions

Figure 1: Example of an univariate regression. The training sample is a uniform distribution in $x \in (-0.9, 0.9)$ and a Gaussian smearing in the y direction for every x. The $\mathbb{E}(E^+_{\chi^2})$ expected loss function in fig. 1a has a minimum for the 16 degree polynomial in the examined range. The curve below that belongs to the E_{χ^2} loss function evaluated with the optimal polynomial, and by definition it can only decrease with additional degrees of freedom. The top curve is one standard deviation above the expected loss function. Figure 1b shows the 16 degree optimal polynomial with the largest uncertainties at the two boundaries of the sample. Figure 1c and 1d shows the result of the regression with the splitting procedure described in the text, with allowed maximal polynomial degrees $n_{\max} = 0$ and 1 accordingly. Both of these regressions functions are within one sigma of the conditional $\mathrm{P}(y|x)$ distribution and approximate the $\mathbb{E}(y|x)$ conditional expectation value without picking up statistical noise.

the function evaluation phase. A double precision number can be thought of as a sixteen digit decimal number, and though its n^{th} power is expected to have a relative error of only $n \cdot 10^{-16}$, the numerous subtractions and multiplications needed for the linear equation solver can easily blow this up. In the univariate case, the numerical errors seems to become significant at the polynomial degree around 20 for double precision and 24...30 for 128bit long double precision. In the case of d input dimensions the size of the G matrix grows rapidly with the number of degrees used, since the polynomial coefficients of the $F(x)$ fit function are d-degree symmetric tensors, with $\binom{n+d-1}{n}$ free parameters. All of the free parameters in $F(x)$ contribute to the size of the G matrix. For 20 input dimensions a third degree polynomial has nearly 2000 free parameters, resulting in G matrix size of 2000×2000. Although solving a linear equation with this only takes a few seconds on a modern-day computer, this also means that more than a million instructions are needed to express each unknown of the $F(x)$ polynomial, resulting in large numerical errors.

(a) original (b) remodeled

Figure 2: Figure (a) shows a photo of a Chinese terra-cotta soldier, whose intensity map was remodeled in fig. (b) in small patches with polynomials. The photo was treated as a random sample, meaning prior knowledge of the uncertainty of the pixel intensities were neglected. The region boundaries were determined with the algorithm described in text, based on the principal axes of the input distribution. The regions were modeled with a two dimensional linear polynomial, with a second layer of univariate regression upon it, to simulate a sigmoid-like behavior. Over- and undershoots from the displayable range were rounded to the maximum and minimum intensities accordingly. A photo instead of a real 3D distribution was chosen here in order to demonstrate that this simple fitting and splitting method can detect the significant features in the data while smoothing the small details, as one would require for a regression method on a statistical sample.

6. Minimizing numerical errors

To overcome the problem of the high polynomial degrees and the large matrices, one can split the sample into many smaller phase spaces, which may require smaller polynomial degrees. For this one must decide on a maximum number of degrees $n_{\max}$ of the polynomial function which is allowed in regression, and $n_{\max} + k$, $k > 1$ for which the expected loss function is scanned. If the optimal degree of the expected loss function is larger than $n_{\max}$, one can apply a predefined algorithm that splits the input phase space. One such algorithm for the univariate case simply splits the input phase space at the x mean, as demonstrated on fig. 1. This requires practically no additional computation, since the $\langle x \rangle$ was already calculated for the regression. The splitting and fitting can be repeated until the full sample is regressed. A possible extension of this approach in the multivariate case finds the multivariate mean $\langle x_\mu \rangle$ first, then splits the sample at this point parallel to the principal axis which is given by the eigenvector with the largest

eigenvalue of the $\langle x_\mu x_\nu \rangle - \langle x_\mu \rangle \langle x_\nu \rangle$ matrix; see fig. 2. These methods have the advantage that they place the cut boundaries within the distribution, so the regression on these phase spaces are less likely to produce degenerate solutions. A seemingly more optimal splitting method would be finding the place where the fitted polynomial with $n_{\max} + 1$ degrees have the largest derivative, since this is a hard place to model with an $n_{\max}$ degree polynomial. However, it is non-trivial to define and find this boundary in the multivariate case, and this boundary typically appear nearby the tails of the x distribution, where the fitted function has the largest uncertainty.

7. Conclusions

The presented method is capable of modeling multivariate statistical data with polynomials by detecting the significant features in the data. It is a fast and robust method, as most calculations are computationally very simple and does not require numerical optimization. Similarly to the statistical bootstrap method, the uncertainty of the regression function can be determined from the training sample, but in this case analytically. In combination with a phase space splitting method, it can be extended to fit very complex data, still maintaining numerical stability.

References

[1] Kövesárki P 2012 Polynomial expansion of the binary classification function (*Preprint* `http://arxiv.org/abs/1203.5647`)

[2] Kövesárki P 2012 *Multivariate methods and the search for single top-quark production in association with a W boson in ATLAS* Ph.D. thesis University of Bonn URL `http://hss.ulb.uni-bonn.de/2013/3188/3188.htm`

[3] Metzger W J 2010 *Statistical Methods in Data Analysis* (Radboud Universiteit Nijmegen, The Netherlands) ISBN HEN-343 URL `http://www.hef.ru.nl/~wes/stat_course/statist.pdf`

[4] The European Mathematical Society *Encyclopedia of Mathematics* 1 URL `http://www.encyclopediaofmath.org/index.php?title=Hankel_matrix&oldid=23850`

[5] Bishop C M 2006 *Pattern Recognition and Machine Learning* (Secaucus, NJ, USA: Springer-Verlag New York, Inc.) ISBN 0387310738 iSBN 0-387-31073-8

[6] Ripley B D 1996 *Pattern Recognition and Neural Networks* (Cambridge University Press) ISBN 0521460867 iSBN 978-0521-71770-0

[7] Scargle J D, Norris J P, Jackson B and Chiang J 2013 *The Astrophysical Journal* **764** 167