THE EUROPEAN
PHYSICAL JOURNAL C

Regular Article - Experimental Physics

The Linear Template Fit

Daniel Britzger[a]

Max-Planck-Institut für Physik, Föhringer Ring 6, 80805 Munich, Germany

Abstract The estimation of parameters from data is a common problem in many areas of the physical sciences, and frequently used algorithms rely on sets of simulated data which are fit to data. In this article, an analytic solution for simulation-based parameter estimation problems is presented. The matrix formalism, termed the *Linear Template Fit*, calculates the best estimators for the parameters of interest. It combines a linear regression with the method of least squares. The algorithm uses only predictions calculated for a few values of the parameters of interest, which have been made available prior to its execution. The Linear Template Fit is particularly suited for performance-critical applications and parameter estimation problems with computationally intense simulations, which are otherwise often limited in their usability for statistical inference. Equations for error propagation are discussed in detail and are given in closed analytic form. For the solution of problems with a nonlinear dependence on the parameters of interest, the *Quadratic Template Fit* is introduced. As an example application, a determination of the strong coupling constant from inclusive jet cross section data at the CERN Large Hadron Collider is studied and compared with previously published results.

1 Introduction

The interpretation of collected research data is a frequent and important step of the data analysis procedure in many scientific fields. A common task in this interpretation is the estimation of parameters using predictions obtained from analytic calculations or simulation programs, usually referred to as *fitting*. Frequently used fitting algorithms rely on numerical methods and utilize iterative function minimization algorithms [1–7]. The availability of more computational power and the development of improved algorithms such as machine learning techniques [8,9] have led to more comprehensive

statistical methods that can be employed for the estimation of parameters [10–13]. At the same time, simulation programs have become more complex and more computationally demanding, placing constraints on the inference algorithms. For example, in the field of particle physics, such simulations provide fully differential predictions in perturbative quantum chromodynamics and the electroweak theory (see Refs. [14–22] for some recent advances) and may take several hundreds up to a few thousand years on a single, modern CPU core. Sometimes, the simulation of the physical process is followed by a detailed simulation of the experimental apparatus in order to provide synthetic data as close as possible to the recorded raw data. At the experiments at the CERN Large Hadron Collider (LHC), the simulated data include the physics of the proton–proton collision [23–25] and the simulation of processes that take place subsequently in the detector material [26]. This results in the simulation of about 100 million electronic channels [27], which are processed similarly to real data [28]. Consequently, these simulated data cannot be used in iterative optimization algorithms because of their computational cost. In addition, these predictions can be provided only for a few selected values of the theory parameters of interest.

Besides the restrictions associated with computationally demanding simulations, there exist other reasons why less complex simulations cannot be used by certain optimization algorithms. Numerical instabilities or statistical fluctuations in the predictions can result in fit instabilities, input quantities for the predictions are sometimes only available for selected discrete values of the parameters of interest [for example, parameterizations of the parton distribution functions of the proton (PDFs) which are only available for a few values of the strong coupling constant α_s [29,30]], or simply because there are technical limitations when interfacing a simulation program to the inference algorithm and the parameter(s) of interest cannot be made explicit to the minimization algorithm. A further frequent limitation for simulation-based inference is related to the intellectual property of the simulation pro-

[a] e-mail: britzger@mpp.mpg.de (corresponding author)

gram, where the computer code is not publicly available, but the obtained predictions are available for a given set of reference values. In short, in many cases, the inference algorithm can only use predictions that have been provided previously since a recomputation by the inference algorithm is not possible.

Several variants of simulation-based template fits, or template-like fits, are nowadays used in high-energy physics [13,31–44]. These make use of polynomial interpolation or regression between the reference points or apply related techniques [45–47]. Some algorithms use un-binned quantities, while others make use of summary statistics. Furthermore, different algorithms to find the best estimator are employed; for example, numerical methods or iterative minimization algorithms are used to find the extremum of a likelihood or χ^2 function. In general, typical iterative minimization algorithms can be considered to be template fits, since every iteration generates a new template at a new reference point, which are then used to find the extremum of the likelihood function.

In this article, the equations of the *Linear Template Fit* (LTF) are derived, which can be written in a closed matrix equation form. The Linear Template Fit provides an analytic solution for the determination of the best estimator in a least-squares problem under the assumption that the predictions are available only for a finite set of values of the parameters of interest. The equations are obtained from a two-step marginalization of the underlying statistical model, which assumes normally distributed uncertainties. The first step provides a linearized, but continuous, estimate of the prediction [1], and the second step provides the best estimator of the fitting problem. The Linear Template Fit is suited for a wide range of parameter estimation problems, where the input data can be cross section measurements, event counts, or other summary statistics like histograms.

This article is structured as follows. After a brief review of the method of template fits in Sect. 2, the equation of the Linear Template Fit is derived in Sect. 3 for a univariate problem. While the Linear Template Fit turns out to be a simple matrix equation, to the author's knowledge it has not been published in its closed form before. The emphasis of this article is on variations of the Linear Template Fit, its applicability, and the propagation of uncertainties. The multivariate Linear Template Fit is discussed in Sect. 5, and the Linear Template Fit with relative uncertainties is presented in Sect. 7. A detailed discussion on error treatment and propagation is given in Sect. 8. A (detector) response matrix is inserted into the Linear Template Fit in Sect. 9. Several con-

siderations for the applicability of the Linear Template Fit are provided in Sect. 10, as well as a simple algorithm for cross-checks and the relation to other algorithms. While the prerequisite of the Linear Template Fit is the linearity of the prediction in the parameters of interest, potential nonlinear effects are estimated and discussed in Sect. 11. There, the Quadratic Template Fit is introduced, an algorithm with fast convergence using second-degree polynomials for the parameter dependence of the model. Three toy examples are discussed in Sects. 4, 6, and 13, and are also remarked upon occasionally in between. A comprehensive and real example application of the Linear Template Fit is given in Sect. 14, where the value of the strong coupling constant $\alpha_s(m_Z)$ is determined from inclusive jet cross section data obtained by the CMS experiment at the LHC. The best estimators are compared with previously published results, obtained with other inference algorithms. Section 15 provides a summary. Additional details are collected in the Appendix, where a table can be found of the notation adopted in this article.

2 Template fits

The objective function of a multivariate optimization problem assuming normally distributed random variables is a likelihood function calculated as the joint probability distribution of Gaussians

$$\mathcal{L} = \prod_i^n \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(\frac{-(d_i - \lambda_i(\boldsymbol{\alpha}))^2}{2\sigma_i^2}\right), \tag{1}$$

where d_i are the i data values, σ_i^2 the variances, and λ_i is the value that is dependent on the model parameters of interest $\boldsymbol{\alpha}$. Gaussian probability distributions are often appropriate assumptions for real data due to the central limit theorem [48,49]. For numerical computation and optimization algorithms, it is convenient to rewrite $\mathcal{L}$ in terms of a least-squares equation using $\chi^2 = -2\log\mathcal{L}$ and omitting constant terms. In matrix notation, and using a covariance matrix, the objective function becomes [2]

$$\chi^2(\boldsymbol{\alpha}) = (\boldsymbol{d} - \boldsymbol{\lambda}(\boldsymbol{\alpha}))^{\mathrm{T}} V^{-1} (\boldsymbol{d} - \boldsymbol{\lambda}(\boldsymbol{\alpha})). \tag{2}$$

The maximum likelihood estimators (MLE) of the parameters $\boldsymbol{\alpha}$ are found by minimizing χ^2

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \chi^2(\boldsymbol{d}; \boldsymbol{\alpha}). \tag{3}$$

For this task, iterative function optimization algorithms are commonly employed (see e.g. Refs. [4–6,50]). The least-

---

[1] Usually, there exists prior knowledge about the value of the parameters of interest, for example from previous studies or theoretical considerations. The templates are then constructed in the vicinity of the expected best estimator, such that the linearized model is a good approximation in general.

[2] A table of the notation is provided in Appendix A. It is assumed, that the hypothesized function λ is obtained from the simulator and will be denoted as *model*, since it represents commonly a comprehensive calculation from a complicated theoretical *model*.

squares method is then equivalent to the maximum likelihood method in Eq. (1).

A common problem that physicists often face at this point is that the model $\boldsymbol{\lambda}(\boldsymbol{\alpha})$ may be computationally intense, time-consuming to calculate, or not be available in a compatible implementation. Hence, the objective function χ^2 cannot be evaluated repeatedly and for arbitrary values of $\boldsymbol{\alpha}$, as it would be required in gradient methods. In contrast, it is often the case that the model is available for a set of j model parameters $\dot{\boldsymbol{\alpha}}_{(j)}$. These predictions $\boldsymbol{y}_{(j)} := \boldsymbol{\lambda}(\dot{\boldsymbol{\alpha}}_{(j)})$ are denoted as *templates* for given *reference values* $\dot{\boldsymbol{\alpha}}_{(j)}$ in the following. Consequently, a template fit exploits only previously available information and can be considered as a two-step algorithm:

1. In a first step, a (continuous) representation of the model in the parameter $\boldsymbol{\alpha}$ needs to be derived from the templates. This can be done by interpolation, regression, or any other approximation of the true model with some reasonable function $\boldsymbol{y}(\boldsymbol{\alpha}, \hat{\boldsymbol{\theta}})$, like

$$\boldsymbol{\lambda}(\boldsymbol{\alpha}) \approx \boldsymbol{y}(\boldsymbol{\alpha}, \hat{\boldsymbol{\theta}}), \tag{4}$$

where $\boldsymbol{y}(\boldsymbol{\alpha}, \hat{\boldsymbol{\theta}})$ is dependent on $\boldsymbol{\alpha}$ and several parameters $\boldsymbol{\theta}$. The values $\boldsymbol{\theta}$ are then determined from the templates, for example with a least-squares method and a suitable χ^2, similar to

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} \chi^2(\boldsymbol{y}_{(0)}, \ldots, \boldsymbol{y}_{(j)}; \boldsymbol{\theta}). \tag{5}$$

Such a procedure can be done for each point i separately, or for all points simultaneously.

2. In a second step, which is often decoupled from the first, the best estimators of the parameter of interest are determined using the approximated true model from the first step. It can be expressed as

$$\hat{\boldsymbol{\alpha}} = \arg \min_{\boldsymbol{\alpha}} \chi^2(\boldsymbol{d}; \boldsymbol{\alpha}), \tag{6}$$

where the objective function could be a least-squares expression, like the one of Eq. (2), but using the approximated model $\boldsymbol{y}(\boldsymbol{\alpha}, \hat{\boldsymbol{\theta}})$ from the first step, like

$$\chi^2(\boldsymbol{\alpha}) = (\boldsymbol{d} - \boldsymbol{y}(\boldsymbol{\alpha}, \hat{\boldsymbol{\theta}}))^{\mathrm{T}} V^{-1} (\boldsymbol{d} - \boldsymbol{y}(\boldsymbol{\alpha}, \hat{\boldsymbol{\theta}})). \tag{7}$$

Such a two-step algorithm appears to be cumbersome, and in the following the equation of the Linear Template Fit is introduced, which combines the two steps into a single equation, under certain assumptions.

3 The basic method of the Linear Template Fit

The basic methodology of the Linear Template Fit is introduced, and in order to simplify the discussion, a *univariate* model $\lambda(\alpha)$ is considered, where the parameter of interest is denoted as α. It is assumed that the model is available for several values $\dot{\alpha}_{(j)}$ of the parameter of interest, and these j predictions are denoted as the *templates*, $\lambda(\dot{\alpha}_{(j)})$. These templates will be confronted with the vector of the data $\boldsymbol{d}$ in order to obtain the best estimator $\hat{\alpha}$.

We consider a basic optimization problem based on Gaussian probability distributions, and the objective function is written in terms of a least-squares equation

$$\chi^2(\alpha) = (\boldsymbol{d} - \boldsymbol{\lambda}(\alpha))^{\mathrm{T}} V^{-1} (\boldsymbol{d} - \boldsymbol{\lambda}(\alpha)). \tag{8}$$

In a first step, the model $\lambda(\alpha)$ is approximated linearly from the template distributions, and in every entry i, $\lambda_i(\alpha)$ is approximated through a linear function $y_i(\alpha; \theta_0, \theta_1)$ like

$$\lambda_i(\alpha) \approx y_i(\alpha; \hat{\boldsymbol{\theta}}) \quad \text{using} \quad y_i(\alpha; \boldsymbol{\theta}_{(i)}) := \theta_0^{(i)} + \theta_1^{(i)} \alpha. \tag{9}$$

The best estimators of the function parameters, $\hat{\boldsymbol{\theta}}$, are obtained from the templates by linear regression ("straight line fit," see e.g. Refs. [51–53]), whose formalism also follows the statistical concepts that were briefly outlined above. Each of the templates is representative of a certain value of α, which will be denoted in the following as *reference points* $\dot{\alpha}_j$, and all reference values form the j vector $\dot{\boldsymbol{\alpha}}$. Next, an extended Vandermonde design matrix [54], the *regression matrix*, is constructed from a unit column vector and $\dot{\boldsymbol{\alpha}}$:

$$M := \begin{pmatrix} \mathbf{1} & \dot{\boldsymbol{\alpha}} \end{pmatrix} = \begin{pmatrix} 1 & \dot{\alpha}_1 \\ \vdots & \vdots \\ 1 & \dot{\alpha}_j \end{pmatrix}. \tag{10}$$

From the method of least squares, the best estimators of the polynomial parameters for the ith entry are [55]

$$\hat{\boldsymbol{\theta}}_{(i)} = \begin{pmatrix} \hat{\theta}_0 \\ \hat{\theta}_1 \end{pmatrix}_{(i)} = M_{(i)}^+ \begin{pmatrix} y_{(1),i} \\ \vdots \\ y_{(j),i} \end{pmatrix} \quad \text{using}$$

$$M_{(i)}^+ = (M^{\mathrm{T}} \mathcal{W}_{(i)} M)^{-1} M^{\mathrm{T}} \mathcal{W}_{(i)}, \tag{11}$$

where the matrix $M_{(i)}^+$ is a g-inverse of a least-squares problem [54,56], $y_{(j),i}$ denotes the ith entry of the jth template vector, and $\mathcal{W}_{(i)}$ is an inverse covariance matrix which represents uncertainties in the templates, e.g. $\mathcal{W}_{(i),j,j} = \sigma_{y_{(j),i}}^{-2}$. However, for the purpose of the Linear Template Fit, an important simplification is obtained since the special case of an unweighted linear regression is applicable to a good approximation. The (inverse) covariance matrix $\mathcal{W}_{(i)}$ in this problem has two features: it is a diagonal matrix, since the templates are generated independently, and secondly, it has, to a very good approximation, equally sized diagonal elements, $\mathcal{W}_{(i),j,j} \approx \mathcal{W}_{(i),j+1,j+1}$. This simplification is com-

monly well applicable, for example, if the model is obtained from an exact calculation or if Monte Carlo event generators were employed and the templates all have a similar statistical precision. Consequently, the size of the uncertainties in the templates $\sigma_{y_{(j)},i}$ factorizes from $\mathcal{W}_{(i)}$ and subsequently cancels in the calculation of $M_{(i)}^{+}$, eq. (11). Thus, an unweighted polynomial regression is applicable, and the g-inverse simplifies to a left-side Moore–Penrose pseudoinverse matrix [57–59],

$$M_{(i)}^{+} \simeq M^{+} = (M^{\mathrm{T}} M)^{-1} M^{\mathrm{T}} , \tag{12}$$

with M as defined in Eq. (10). It is observed that the matrix M^{+} is *universal*, so it is independent on the quantities $y_{(j),i}$ and $\mathcal{W}_{(i)}$, but is calculated only from the reference points $\dot{\boldsymbol{\alpha}}$. Consequently, it is equally applicable in every bin i.

Henceforth, we decompose the matrix M^{+} (Eq. (12)), which is a $2 \times j$ matrix, into two row vectors, like

$$M^{+} =: \begin{pmatrix} \bar{\boldsymbol{m}}^{\mathrm{T}} \\ \tilde{\boldsymbol{m}}^{\mathrm{T}} \end{pmatrix} , \quad \text{so} \quad \begin{array}{l} \bar{\boldsymbol{m}} = (M^{+})^{\mathrm{T}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ \tilde{\boldsymbol{m}} = (M^{+})^{\mathrm{T}} \begin{pmatrix} 0 \\ 1 \end{pmatrix} . \end{array} \tag{13}$$

This introduces the j column vectors $\bar{\boldsymbol{m}}$ and $\tilde{\boldsymbol{m}}$.

We further introduce the *template matrix* Y, a $i \times j$ matrix, which is constructed from the column vectors of the template distributions, and can be written as

$$
\begin{aligned}
Y &:= \begin{pmatrix} Y_{1,1} & \dots & Y_{1,j} \\ \vdots & \ddots & \vdots \\ Y_{i,1} & \dots & Y_{i,j} \end{pmatrix} = \begin{pmatrix} \boldsymbol{y}_{(\dot{\alpha}_1)} & \cdots & \boldsymbol{y}_{(\dot{\alpha}_j)} \end{pmatrix} \\
&= \begin{pmatrix} y_{(1),1} & \cdots & y_{(j),1} \\ \vdots & \vdots & \vdots \\ y_{(1),i} & \cdots & y_{(j),i} \end{pmatrix} .
\end{aligned}
\tag{14}
$$

Hence, substituting the best estimators $\hat{\boldsymbol{\theta}}$ (Eq. (11)) into the linearized model $\boldsymbol{y}(\alpha, \hat{\boldsymbol{\theta}})$ (Eq. (9)), and using $\bar{\boldsymbol{m}}$ and $\tilde{\boldsymbol{m}}$ (Eq. (13)) and Y (Eq. (14)), the model can be expressed as a matrix equation[3]:

$$\boldsymbol{\lambda}(\alpha) \approx \boldsymbol{y}(\alpha, \hat{\boldsymbol{\theta}}) = Y \bar{\boldsymbol{m}} + Y \tilde{\boldsymbol{m}} \alpha . \tag{15}$$

It is seen that the polynomial parameters $\boldsymbol{\theta}$ are no longer explicit in this expression. Consequently, the objective function (Eq. (8)) becomes a linear least-squares expression

$$\chi^2(\alpha) = (\boldsymbol{d} - Y \bar{\boldsymbol{m}} - Y \tilde{\boldsymbol{m}} \alpha)^{\mathrm{T}} W (\boldsymbol{d} - Y \bar{\boldsymbol{m}} - Y \tilde{\boldsymbol{m}} \alpha) , \tag{16}$$

where W is the inverse covariance matrix, $W = V^{-1}$.

---

[3] Note that Eq. (15) represents already a very useful (linearly approximated) representation of the simulation, since it transforms a discrete representation of the model into a continuous function. This equation may become useful for several other purposes, or optimization algorithms, whenever only templates are available, but a (quickly calculable) continuous function is required. Higher-degree polynomials beyond the linear approximation can also be used for that purpose, so $\boldsymbol{\lambda}^{\mathrm{T}}(\alpha) \simeq (\alpha^0 \ \dots \ \alpha^n) \mathcal{M}^{+} Y^{\mathrm{T}}$, using a n-degree regression matrix $\mathcal{M}$ (Eq. (81)).

In the second step of the derivation of the Linear Template Fit, the best estimator $\hat{\alpha}$ is determined. Due to the linearity of χ^2 in α, the best-fit value of α is defined by the stationary point of χ^2 [49,55,60], and the best linear unbiased estimator of the parameter of interest is

$$\hat{\alpha} = \frac{(Y \tilde{\boldsymbol{m}})^{\mathrm{T}} W}{(Y \tilde{\boldsymbol{m}})^{\mathrm{T}} W Y \tilde{\boldsymbol{m}}} (\boldsymbol{d} - Y \bar{\boldsymbol{m}}) . \tag{17}$$

When introducing the g-inverse of least squares, a $1 \times i$ matrix,

$$F := \frac{(Y \tilde{\boldsymbol{m}})^{\mathrm{T}} W}{(Y \tilde{\boldsymbol{m}})^{\mathrm{T}} W Y \tilde{\boldsymbol{m}}} , \tag{18}$$

the master formula of the *Linear Template Fit* is found to be

$$\hat{\alpha} = F(\boldsymbol{d} - Y \bar{\boldsymbol{m}}) , \tag{19}$$

where Eqs. (13), (14), and (18) were used, and $\boldsymbol{d}$ denotes the vector of the data.

Since the distribution of the data is assumed to be normally distributed, from the linearity of the least squares, it follows that the estimates are also normally distributed. From error propagation, the variance of the best estimator is

$$\sigma_{\hat{\alpha}}^2 = F V F^{\mathrm{T}} = (F^{\mathrm{T}} W F)^{-1} . \tag{20}$$

Given the case that the approximation in Eq. (15) holds, the estimator $\hat{\alpha}$ represents a best and unbiased estimator according to the Gauß–Markov theorem [49,55], and also the variances are the smallest among all possible estimators. Since $\boldsymbol{d}$ does not enter the calculation of the variances, Eq. (20), the uncertainties are equivalent to the expected uncertainties from the Asimov [61] data $\boldsymbol{\lambda}(\hat{\alpha})$.

When using the right expression of Eq. (13), it can be directly seen that the best estimator is indeed obtained from a single matrix equation using only the template quantities $(Y, \dot{\boldsymbol{\alpha}})$ and the data details $(\boldsymbol{d}, W)$, and the Linear Template Fit can alternatively be written as

$$\hat{\alpha} = \left(\left(\Upsilon \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right)^{\mathrm{T}} W \Upsilon \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right)^{-1} \left(\Upsilon \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right)^{\mathrm{T}} W \left(\boldsymbol{d} - \Upsilon \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right)$$

using

$$\tag{21}$$

$$\Upsilon = Y (M^{+})^{\mathrm{T}} = Y \begin{pmatrix} \mathbf{1} & \dot{\boldsymbol{\alpha}} \end{pmatrix} \left(\begin{pmatrix} \mathbf{1} & \dot{\boldsymbol{\alpha}} \end{pmatrix} \begin{pmatrix} \mathbf{1} & \dot{\boldsymbol{\alpha}} \end{pmatrix}^{\mathrm{T}} \right)^{-1} . \tag{22}$$

In contrast, when the (commonly well-justified) approximation of the unweighted regression in each bin is not applicable [see left side of Eq. (12)], then the bin-dependent regression matrices $M_{(i)}^{+}$ remain explicit. Such a case may be present when a Monte Carlo event generator is used for the generation of the templates and the statistical uncertainties have to be considered, and one sample was generated with higher statistical precision than the others, for instance, since it also serves for further purposes like unfolding. However, when using such a weighted regression, the best estimator may

still be expressed using Eq. (17), but the elements of the two vectors $Y\bar{m}$ and $Y\tilde{m}$ would need to be calculated separately and become

$$(Y\bar{m})_i = \begin{pmatrix} 1 \\ 0 \end{pmatrix}^{\mathrm{T}} M_{(i)}^+ x_{(i)} \quad \text{and}$$

$$(Y\tilde{m})_i = \begin{pmatrix} 0 \\ 1 \end{pmatrix}^{\mathrm{T}} M_{(i)}^+ x_{(i)} , \qquad (23)$$

where the vectors $x_{(i)}$ denote the ith row of the template matrix, $x_{(i)} = Y_i^{\mathrm{T}}$, and the matrices $M_{(i)}^+$ may include uncertainties in the templates $x_{(i)}$ through $\mathcal{W}_{(i)}$ (see Eq. (11), right). This case, however, will not be considered any further in that article.

In the following sections, an example application of the Linear Template Fit is presented, and subsequently, generalizations of the Linear Template Fit are discussed, and formulae for error propagation are presented.

4 Example 1: the univariate Linear Template Fit

An example application is constructed from a random number generator within the ROOT analysis framework [62]. The physics model is a normal distribution with a standard deviation of 6.0. The mean value, denoted as α, is subject to the inference. Similarly to particle physics, a counting experiment is simulated, and the (pseudo-)data are generated from 500 *events* at a *true* mean value of $\alpha = 170.2$, while limited acceptance restricts the measurement to values larger than 169. Some measurement distortions are simulated by using a standard deviation of 6.2 for the pseudo-data. Limited acceptance of the experimental setup is simulated by considering only one side of the normal distribution, and altogether 14 "bins" with unity width are "measured." Let us assume that from a previous measurement it is known that the physics parameter of interest has a value of about $\alpha \approx 170.5 \pm 1.0$. Therefore, seven templates in a range from $\dot{\alpha} = 169.0$ to 172.0 in steps of 0.5 are generated, and each template is generated using 40,000 random numbers. From Eq. (20) it is then found that the pseudo-data will be able to determine the value of α approximately with an uncertainty of ± 0.5, and subsequently, additional templates at values of $\dot{\alpha} = 169.0$, 169.5, 170.5, 171, and 171.5 are generated.

The Linear Template Fit using these seven templates is illustrated in Fig. 1. The Linear Template Fit from Eq. (19) reports a value of $\hat{\alpha} = 170.03 \pm 0.41$, which is fully consistent within the statistical uncertainty with the generated true value of 170.2, although a somewhat different standard deviation was used. In Fig. 1 (right), the result from a hypothesized weighted linear regression is shown (cf. Eq. (23)), but it is almost indistinguishable from the unweighted linear regression as is used in the Linear Template Fit. This is because the generation of the individual templates always employs the same methodology, as already argued above. Furthermore,

the estimated best model is displayed in Fig. 1 (left), which is defined from Eq. (15) when substituting the best estimator, $\hat{y} = y(\hat{\alpha})$. This results in $\chi^2 = 5.6$ for 14 data points and one free parameter. Other seeds for the random number generator of course result in different values. It should be noted that this example is purposefully constructed with large (statistical) uncertainties in order to obtain a visually clearer presentation in the figures, although in some bins the assumption of normally distributed random variables is then a rather poor approximation to the Poisson distribution.

5 The multivariate Linear Template Fit

Phenomenological models often depend on multiple parameters, and thus it is a common task to determine multiple parameters at a time. Such k parameters of interest are referred to as $\alpha_1, \ldots, \alpha_k$, or simply $\boldsymbol{\alpha}$, and the best estimators are denoted as $\hat{\alpha}_1, \ldots, \hat{\alpha}_k$, or $\hat{\boldsymbol{\alpha}}$. The linear representation of the model is a hyperplane $y_i(\boldsymbol{\alpha})$ in each of the i data points, defined as

$$\boldsymbol{\lambda}(\boldsymbol{\alpha}) \approx \mathbf{y}(\boldsymbol{\alpha}; \hat{\boldsymbol{\theta}}) = \hat{\theta}_0 + \hat{\theta}_{(1,1)}\alpha_1 + \cdots + \hat{\theta}_{(1,k)}\alpha_k , \qquad (24)$$

where a constant θ_0 and the first-degree parameters $\theta_{(1,k)}$ are considered. Since higher-degree terms or interference terms are not included, the fit parameters need to be (sufficiently) independent or they have been made orthogonal by applying a variable transformation beforehand.

In the multivariate case, each template $\mathbf{y}_{(j)}$ is representative of a reference point $\dot{\boldsymbol{\alpha}}_{(j)}$ in the k-dimensional space. The regressor matrix M is constructed as a $j \times (1 + k)$ design matrix:

$$M := \begin{pmatrix} 1 & \dot{\alpha}_{(1),1} & \ldots & \dot{\alpha}_{(1),k} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \dot{\alpha}_{(j),1} & \ldots & \dot{\alpha}_{(j),k} \end{pmatrix} . \qquad (25)$$

As in the univariate case, the pseudoinverse M^+ is calculated from Eq. (12), and also the same considerations for the justification of the *unweighted* multiple regression are applicable. Instead of a single vector $\tilde{m}$, there is now a vector for each regression parameter θ_k. Therefore, the $j \times k$ matrix $\tilde{M}$ is introduced, which is defined by decomposing M^+ like

$$M^+ =: \begin{pmatrix} \bar{m}^{\mathrm{T}} \\ \tilde{M}^{\mathrm{T}} \end{pmatrix} . \qquad (26)$$

Hence, the best estimator for the linearized multivariate model $\mathbf{y}(\boldsymbol{\alpha}; \boldsymbol{\theta})$ becomes

$$\boldsymbol{\lambda}(\boldsymbol{\alpha}) \approx \hat{\mathbf{y}}(\boldsymbol{\alpha}) = Y\bar{m} + Y\tilde{M}\boldsymbol{\alpha} , \qquad (27)$$

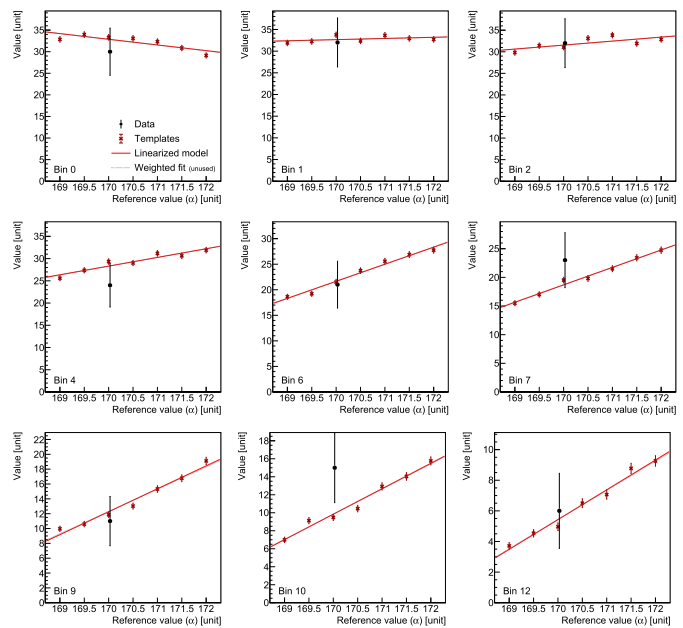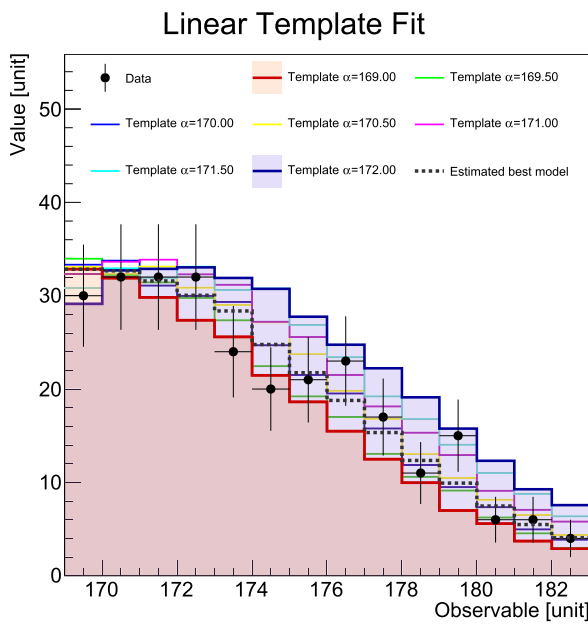where the θ parameters are again no longer explicit.

Fig. 1 Two views of the template matrix Y of an example application of the Linear Template Fit. Left: the template distributions Y as a function of the observable. The full circles display a pseudo-data set with statistical uncertainties, and the dotted line indicates the estimated model after the fit, $\mathbf{y}(\hat{\alpha})$. Right: the template distributions Y of an example application of the Linear Template Fit, but as a function of the refer-

ence values of the parameter of interest α. Only nine selected "bins" i are displayed. The red line shows the linearized model $\hat{y}_i(\alpha)$, Eq. (15), in the respective bin, and the full circles are again the pseudo-data. The dotted line indicates a weighted linear regression, which is not used in the Linear Template Fit

When using the linear approximation of the model, $\boldsymbol{\lambda}(\boldsymbol{\alpha}) \approx \hat{\mathbf{y}}(\boldsymbol{\alpha})$, the χ^2 function (Eq. (2)) for the multivariate Linear Template Fit becomes

$$\chi^2 \simeq \left(\boldsymbol{d} - \hat{\mathbf{y}}(\boldsymbol{\alpha})\right)^{\mathrm{T}} W \left(\boldsymbol{d} - \hat{\mathbf{y}}(\boldsymbol{\alpha})\right) \tag{28}$$

$$= \left(\boldsymbol{d} - Y\bar{\boldsymbol{m}} - Y\tilde{M}\boldsymbol{\alpha}\right)^{\mathrm{T}} W \left(\boldsymbol{d} - Y\bar{\boldsymbol{m}} - Y\tilde{M}\boldsymbol{\alpha}\right) \tag{29}$$

$$= \left(\boldsymbol{d} - \sum_l \hat{\epsilon}_l \boldsymbol{s}_{(l)} - \hat{\mathbf{y}}(\boldsymbol{\alpha})\right)^{\mathrm{T}} \mathbb{V}^{-1}$$
$$\times \left(\boldsymbol{d} - \sum_l \hat{\epsilon}_l \boldsymbol{s}_{(l)} - \hat{\mathbf{y}}(\boldsymbol{\alpha})\right) + \sum_l \hat{\epsilon}_l^2 . \tag{30}$$

The last equation introduces an equivalent expression in terms of nuisance parameters [63]. The factors $\hat{\epsilon}_l$ are related to uncertainties with full bin-to-bin correlations when writing the covariance matrix as

$$V = \mathbb{V} + V_{\mathrm{corr}} \quad \text{using} \quad V_{\mathrm{corr}} = \sum_l \boldsymbol{s}_{(l)} \boldsymbol{s}_{(l)}^{\mathrm{T}} , \tag{31}$$

where the sum l runs over all uncertainties with full bin-to-bin correlations and the vectors $\boldsymbol{s}_l$ denote the individual systematic uncertainties (also called *shifts*), while the matrix $\mathbb{V}$ includes all other uncertainty components. It is common practice that the systematic shifts are calculated from relative uncertainties and multiplied with the measured data. Implications of this practice are discussed in Sects. 7 and 8.7, and care must be taken that the result does not become biased

[64–69]; a common technique to avoid that bias is discussed in Sect. 8.7.

Equation (30) is again a linear least-squares expression, and the best linear unbiased estimators for the parameters $\hat{\boldsymbol{\alpha}}$ and the nuisance parameters $\hat{\boldsymbol{\epsilon}}$ are obtained from the stationary point. Hence, the best estimators from the *multivariate Linear Template Fit* become

$$\hat{\boldsymbol{a}} = \begin{pmatrix} \hat{\boldsymbol{\alpha}} \\ \hat{\boldsymbol{\epsilon}} \end{pmatrix} = \mathcal{F}(\boldsymbol{d} - Y\bar{\boldsymbol{m}}) , \tag{32}$$

where $\hat{\boldsymbol{a}}$ was introduced and the shorthand notations $\mathcal{F}$ (a g-inverse of least squares), and $\mathcal{D}$ are calculated as

$$\mathcal{F} := \mathcal{D}^{-1} \left(Y\tilde{M} \ \ S\right)^{\mathrm{T}} \mathbb{W}$$
$$:= \left(\left(Y\tilde{M} \ \ S\right)^{\mathrm{T}} \mathbb{W}\left(Y\tilde{M} \ \ S\right) + \begin{pmatrix} 0_k & 0 \\ 0 & 1_l \end{pmatrix}\right)^{-1} \left(Y\tilde{M} \ \ S\right)^{\mathrm{T}} \mathbb{W}, \tag{33}$$

using $\mathbb{W} = \mathbb{V}^{-1}$. The matrix with the fully bin-to-bin correlated uncertainties S is composed from the column vectors of the systematic shifts $\boldsymbol{s}_{(l)}$ as

$$S := \left(\boldsymbol{s}_{(1)} \ \dots \ \boldsymbol{s}_{(l)}\right) , \tag{34}$$

and the symbol $\left(Y\tilde{M} \ \ S\right)$ denotes a composed $i \times (k + l)$ matrix from $Y\tilde{M}$ and S, while 0_k and 1_l denote a $k \times k$ or $l \times l$ zero or unit matrix, respectively. The matrix $\mathcal{D}$ is commonly a full-ranked symmetric matrix and thus invertible. A brief

discussion about a more efficient numerical calculation of $\mathcal{D}^{-1}$ is given in Appendix D. If all uncertainty components are represented through a single covariance matrix V, cf. left side of Eq. (31), the multivariate Linear Template Fit simplifies to

$$\hat{\boldsymbol{\alpha}} = \left((Y\tilde{M})^{\mathrm{T}} W Y \tilde{M} \right)^{-1} (Y\tilde{M})^{\mathrm{T}} W (\boldsymbol{d} - Y\bar{\boldsymbol{m}}) \,. \tag{35}$$

6 Example 2: the multivariate Linear Template Fit

In Example 1, only the mean value of the model (which is a normal distribution) was determined, although the pseudo-data had a slightly different standard deviation than the model. In the following example, a multivariate Linear Template Fit is performed and the same pseudo-data as in Example 1 are used, but both values of the model will be determined: the mean value and the standard deviation of the Gaussian. Therefore, templates are generated for some selected values for the mean (between values of 169.5 and 171) and the standard deviation (between values of 5.8 and 6.4) of a Gaussian, and again, 40,000 events are used for each template. The four "extreme" variations of the two parameters are omitted. The multivariate Linear Template Fit using all these templates is illustrated in Fig. 2.

The best estimators from the multivariate Linear Template Fit (Eq. (35)) are 169.91 ± 0.46 for the mean and 6.28 ± 0.38 for the standard deviation. Their correlation coefficient is found to be -0.3. Within the statistical uncertainties, both values are in very good agreement with the simulated input values of 170.2 and 6.2, respectively. Thus, this example illustrates the application of the multivariate Linear Template Fit with two free parameters, where a visual representation of the linearized model is still possible. However, any number of free parameters is in principle possible, and for an n-parameter fit, the minimum number of linearly independent templates is just $n + 1$.

7 The Linear Template Fit with relative uncertainties

In this section we present the equations for the Linear Template Fit when the estimators obey a log-normal distribution. This could be the case for data when the determination of a variable is affected by a number of multiplicative factors that are subject to uncertainties. Consequently, the variable follows a log-normal distribution due to the central limit theorem [48,49]. Also, when the value of an observed variable is a random proportion of the previous observation, it follows a log-normal distribution [49]. An example would be the measurement of the electron energy in a calorimeter which is affected by a number of fractional energy losses and corrections [10]. Another example would be measurements

that are dominated by systematic multiplicative uncertainties: a prominent multiplicative error is due to uncertainties of the luminosity measurement in particle collider experiments which results in a relative uncertainty. Contrary to additive errors, multiplicative errors cannot change the sign of the variable, and a positively defined observable always remains positive, which is an important prerequisite for several physical quantities such as cross sections. A comparison of the log-normal, the normal, and the Poisson probability distribution function for two selected values of their mean value are displayed in Fig. 3.

The likelihood of a set of i measurements is a joint function of log-normal PDFs

$$\mathcal{L} = \prod_i^N \frac{1}{\sqrt{2\pi \varsigma_i^2 d_i^2}} \exp\left(\frac{-(\log d_i - \mu_i)^2}{2\varsigma_i^2} \right) \,. \tag{36}$$

When writing $\mu_i = \log m_i$ [70,71], it can be directly seen from the residuals that the ratios d_i/m_i enter the likelihood calculation instead of their differences. The variance ς denotes a relative quantity, thus it represents *relative uncertainties*. In fact, the log-normal distribution is equivalent to considering random variables with normally distributed relative uncertainties. For small relative errors, $\varsigma_i \lesssim 20\,\%$, which is a common case in particle physics, the log-normal, the normal, and the Poisson distributions become similar. For larger relative errors, the log-normal distribution provides some approximation of the Poisson distribution, and for practical purposes one benefits from its strictly positiveness, in contrast to the often used normal distribution.

Similarly to the case for normally distributed data, we choose as a first-order approximation of the model in every bin

$$\mu_i(\boldsymbol{\alpha}) \approx \theta_0^{(i)} + \theta_{(1,1)}^{(i)} \alpha_1 + \cdots + \theta_{(1,k)}^{(i)} \alpha_k \,, \tag{37}$$

and the best estimators for the θ parameters are again determined from the templates; consequently, the best estimator of the approximated model is

$$\mu(\boldsymbol{\alpha}) \approx \hat{\mu}(\boldsymbol{\alpha}) = \log(Y)\bar{\boldsymbol{m}} + \log(Y)\tilde{M}\boldsymbol{\alpha} \,. \tag{38}$$

The term $\log(Y)$ denotes a coefficient-wise application of the logarithm to the elements of the template matrix Y, and it is used to match the logarithmized data ($\log d_i$) in the likelihood. Equation (38) is an equally valid first-order approximation of the model as compared with the default linear approximation in Eq. (24).

As an illustration, the linearized model from Eq. (38) is displayed in Fig. 4 for some selected bins of Example 1 (Sect. 4) and compared with the approximations of Eq. (27). The two are reasonably similar in that example, since the templates are present in the vicinity of the best estimator and in a range where a linear approximation is valid. Although the parameter dependence in Example 1 is in fact truly linear,
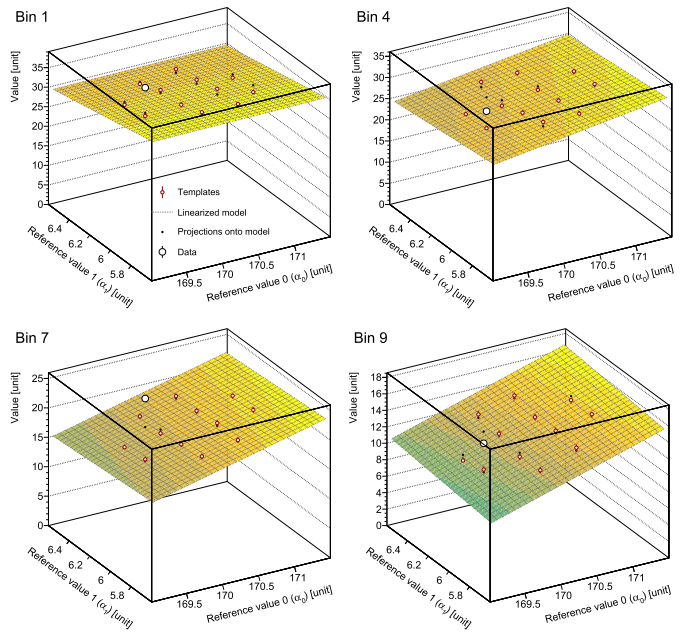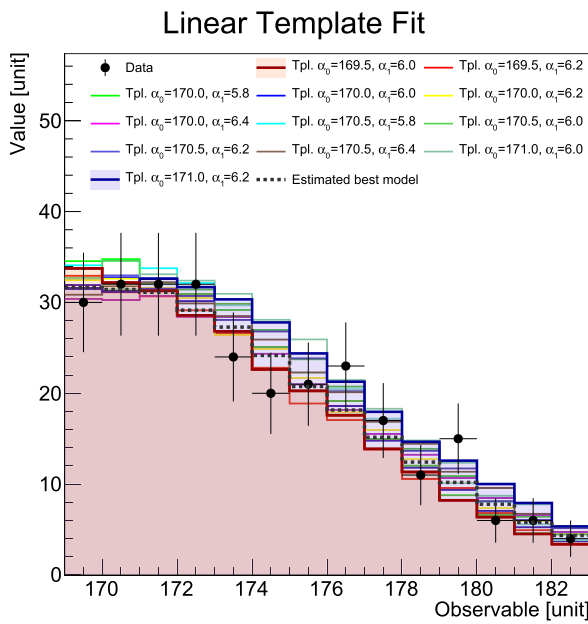
Fig. 2 Two views of the template matrix Y of an example application of the multivariate Linear Template Fit. Left: the template distributions Y as a function the observable. More details are given in the caption of Fig. 1. Right: the template distributions Y (open medium-sized circles) as a function of the two reference values for four selected "bins" i. The plane displays the linearized model $\hat{y}_i(\boldsymbol{\alpha})$, Eq. (27), in the respective bin. The large open circles are again the pseudo-data, and small circles indicate the projection onto the plane $\hat{y}$
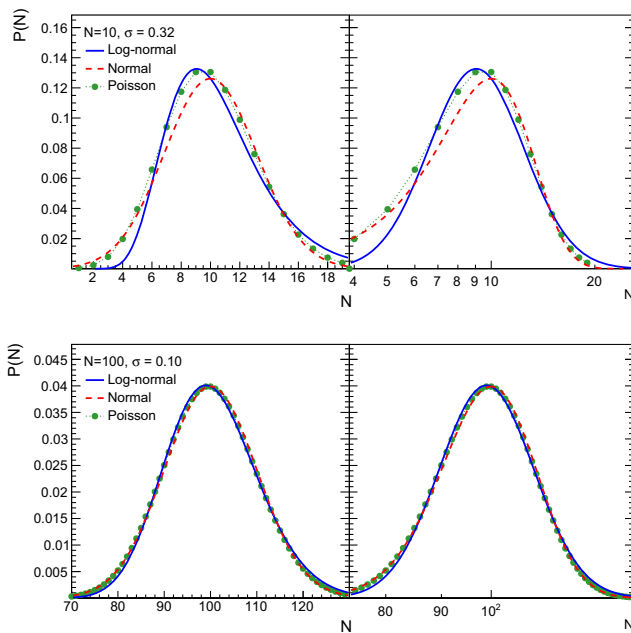
Fig. 3 The log-normal, normal, and Poisson probability distribution functions with mean values of 10 (left) and 100 (right) on a linear and logarithmic axis. These values are commonly referred to as relative uncertainties of 32 % and 10 %, respectively. For larger uncertainties, the log-normal distribution provides a good approximation of the Poisson distribution around the peak. The log-normal and the Poisson distributions are strictly positive. For large N, all distributions become similar

from the generated statistical precision of the templates, the templates cannot discriminate between these two best model estimators.

When using log-normally distributed variables, which corresponds to relative uncertainties, the objective function χ^2 of the Linear Template Fit becomes

$$\chi^2 = \left(\log \boldsymbol{d} - \sum_l \epsilon_l \mathbf{s}_{(l)} - \hat{\boldsymbol{\mu}}(\boldsymbol{\alpha})\right)^{\mathrm{T}} \mathbb{W}$$
$$\times \left(\log \boldsymbol{d} - \sum_l \epsilon_l \mathbf{s}_{(l)} - \hat{\boldsymbol{\mu}}(\boldsymbol{\alpha})\right) + \sum_l \epsilon_l^2 \,. \quad (39)$$

The covariance matrix $\mathbb{V} = \mathbb{W}^{-1}$ comprises normally distributed *relative* uncertainties, and the uncertainties with bin correlations $\mathbf{s}$ also enter the calculation through their relative values. The notation $\log \boldsymbol{d}$ denotes a coefficient-wise application of the logarithm. A common relation between relative and absolute uncertainties would be

$$\mathbf{s}_{(l),i} \simeq \frac{s_{(l),i}}{d_i} \,, \quad (40)$$

and although this equation does not strictly translate between normal and log-normal uncertainties, it is often used in practical applications. Note that we use a Roman font s for relative uncertainties, and italic s for absolute uncertainties.

This χ^2 function is again a linear least-squares problem, and the best estimators for $\boldsymbol{\alpha}$ and $\boldsymbol{\epsilon}$ in the Linear Template Fit are

$$\hat{\boldsymbol{a}} := \begin{pmatrix} \hat{\boldsymbol{\alpha}} \\ \hat{\boldsymbol{\epsilon}} \end{pmatrix} = \mathcal{F}\big(\log \boldsymbol{d} - \log(Y)\bar{\boldsymbol{m}}\big) \,, \quad (41)$$

where the matrix $\mathcal{F}$ is defined in a similar way as Eq. (33), but from relative uncertainties in $\mathbb{V}$ and s (S) and employing the substitution $Y\tilde{M} \rightarrow \log(Y)\tilde{M}$. Due to the treatment of systematic uncertainties as relative uncertainties, the Linear Template Fit with the log-normal p.d.f.'s may be preferred in practical applications over the Linear Template Fit with absolute uncertainties.

When repeating Example 1 (Sect. 4) with relative uncertainties, the best estimator for the mean value is found according to Eq. (41) to be $\hat{\alpha} = 170.40 \pm 0.46$, which is in good agreement with the generated value of 170.2. Some details are further displayed in Fig. 4.

When considering Example 2 (Sect. 6), under the assumption of log-normally distributed pseudo-data, the best estimators are found to be 170.18 ± 0.47 for the mean and 6.33 ± 0.39 for the standard deviation, with a correlation coefficient of -0.3. When further considering a normalization uncertainty of the size of 10%, so a relative uncertainty, the best estimators for the mean and variance become $\alpha_0 = 170.20 \pm 0.76$ and $\alpha_1 = 6.33 \pm 0.49$, respectively. Their correlation coefficient is -0.6, and the nuisance parameter of the normalization uncertainty is $\hat{\epsilon} = -0.02 \pm 0.5$.

---

8 Errors and error propagation

In many applications and fields, like particle physics, it is of great importance to provide detailed uncertainties together with the fit parameters and to propagate uncertainties from all relevant input quantities. Due to its analytic and closed form, the Linear Template Fit provides unique opportunities for detailed error propagation and analysis, that is otherwise not easily possible in other inference algorithms. For example, every single uncertainty component can be propagated separately and analytically to the fit results. In the following, the equations for error propagation are discussed, based on the equation of the Linear Template Fit in Eq. (32).

8.1 Statistical uncertainties and uncertainties without bin-to-bin correlations

Statistical or epistemic uncertainties in the data are represented by the covariance matrix $\mathbb{V}$. The matrix $\mathbb{V}$ may include further uncertainty components with or without bin-to-bin correlations, like systematic or aleatoric uncertainties. From linear error propagation [60], the covariance matrix is propagated to the best estimators $\hat{a}$ as

$$\hat{\mathbb{V}}_{\hat{a}} = \mathcal{F}\mathbb{V}\mathcal{F}^{\mathrm{T}},\tag{42}$$

using $\mathcal{F}$ as defined in Eq. (33). Due to the Gauß–Markov theorem, the estimators $\hat{a}$ for the given objective function χ^2, Eqs. (30) and (39), have the least variance among all possible estimators.

If $\mathbb{V}$ is a sum of multiple individual matrices, like $\mathbb{V} = \mathbb{V}_{\mathrm{stat.}} + \sum \mathbb{V}_{\mathrm{uncorr.}} + \sum \mathbb{V}_{\mathrm{corr.}} + \cdots$, each covariance matrix can be propagated separately using Eq. (42). By doing so,

the contribution from each uncertainty component to the full uncertainty is determined separately.

8.2 Systematic uncertainties or uncertainties with bin-to-bin correlations

Systematic uncertainties associated with the input data are often represented as uncertainties with full bin-to-bin correlations and are represented either as systematic shifts, $s_{(l)}$ (Eq. (34)), or as covariance matrices, $\mathbb{V}_l^{\mathrm{corr}} = s_{(l)}s_{(l)}^{\mathrm{T}}$. Using linear error propagation, these are propagated to the best estimators $\hat{a}$ as

$$\sigma_{\hat{a}}^{(s_{(l)})} = \mathcal{F}s_{(l)}.\tag{43}$$

Alternatively, Eq. (42) can be employed. For the Linear Template Fit with relative uncertainties, one literally has the same equation but using relative values $s_{(l)}$ (compare Eq. (40)).

In Example 2, where a normalization uncertainty of 10% was considered the uncertainty breakdown for the two parameters would be $\sigma_{\hat{\alpha}_0} = \pm 0.56_{(\mathrm{stat.})} \pm 0.51_{(\mathrm{norm.})}$ and $\sigma_{\hat{\alpha}_1} = \pm 0.42_{(\mathrm{stat.})} \pm -0.25_{(\mathrm{norm.})}$.

When systematic uncertainties are included in the Linear Template Fit, their corresponding nuisance parameters ϵ_l are constrained by data, and the obtained reduction of the systematic uncertainty may be used for further analyses. In applications where the fitted parameters are dominated by systematic uncertainties in the input data, it is good practice to constrain the associated systematic shifts by adding additional measurements to the fitting problem [31,35,72,73]. This practice provides additional constraints on the related parameters, ϵ_l, and smaller experimental uncertainties can be obtained. The uncertainty in the nuisance parameter becomes smaller than unity.

8.3 External uncertainties

The generalized transfer matrix $\mathcal{F}$ provides the straightforward opportunity for linear error propagation; see Eqs. (42) or (43). While these equations are predominantly employed to propagate the uncertainties that are included in the Linear Template Fit, equivalent equations are also applicable to propagate further uncertainties that are not used in the fit, but which would still need to be propagated to the fit result. For example, these may be uncertainties in the underlying theory prediction which cannot be constrained by the data. An actual case for such an uncertainty is given below for the non-perturbative correction factors (cf. Fig. 10).
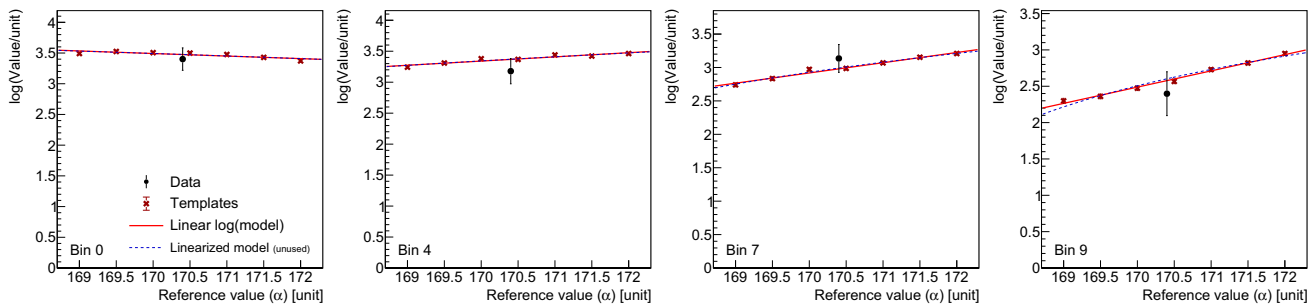
Fig. 4 The best estimate of the parameter dependence of the model, Eq. (38) for four selected bins of Example 1, when performing a Linear Template Fit with log-normally distributed estimators. Note that the *y*-axis indicates the logarithmized values of the templates. The pseudo-

data are displayed for comparison at the best estimator $\hat{\alpha}$ of the Linear Template Fit. For comparison, the linearized model from Eq. (15) is also displayed, which is otherwise not used when log-normally distributed estimators are considered

8.4 Unconstrained uncertainties with full bin-to-bin correlations

A special case of an uncertainty with full bin-to-bin correlations may be considered through an *unconstrained systematic uncertainty*. These are defined by including the respective systematic shift $s_{(l)}$ into the Linear Template Fit, but in contrast to common *constrained* systematic uncertainties, these have a zero value (instead of unity) in the respective diagonal element of the $l \times l$ matrix 1_l in Eq. (33). As a consequence, the Linear Template Fit can exploit the degree of freedom of that systematic shift, but the data does not provide any constraint on that uncertainty source. A possible application would be fits to data, where the normalization is unknown and is a free, unconstrained, systematic uncertainty.

8.5 Uncertainties in the templates

The template distributions Y may be associated with uncertainties from various sources. These can be statistical uncertainties in the elements Y_{ij} or systematic uncertainties with correlations among all entries of Y. One may think about Y being determined from two distinct models, or templates that were generated with different parameters, and the resulting difference in Y may be considered as a systematic uncertainty.

The uncertainties in $\hat{\boldsymbol{a}}$ due to uncertainties in Y are obtained from linear error propagation. The uncertainty in a single element of Y is denoted as $\sigma_{Y_{ij}}$, and using the shorthand notation $\mathcal{D}$ from Eq. (33), the partial derivative of Eq. (32) is

$$\frac{\partial \hat{\boldsymbol{a}}}{\partial Y_{ij}} = \mathcal{D}^{-1} \Bigg[\left(\mathbb{1}_{(ij)} \tilde{M} \, 0_{il} \right)^{\mathrm{T}} \mathbb{W} (\boldsymbol{d} - Y \bar{\boldsymbol{m}}) $$
$$- \left(Y \tilde{M} \, S \right)^{\mathrm{T}} \mathbb{W} \mathbb{1}_{(ij)} \bar{\boldsymbol{m}}$$
$$- \Big[\left(Y \tilde{M} \, S \right)^{\mathrm{T}} \mathbb{W} \left(\mathbb{1}_{(ij)} \tilde{M} \, 0_{il} \right)$$

$$+ \left(\mathbb{1}_{(ij)} \tilde{M} \, 0_{il} \right)^{\mathrm{T}} \mathbb{W} \left(Y \tilde{M} \, S \right) \Big] \hat{\boldsymbol{a}} \Bigg], \tag{44}$$

where $\mathbb{1}_{(ij)}$ denotes an $i \times j$ zero matrix with only the element (i, j) being unity, and 0_{il} denotes an $i \times l$ zero matrix. It should be noted that the usage of the matrices $\mathbb{1}_{(ij)}$ and 0_{il} in numerical calculations is discouraged, since a naive matrix multiplication becomes numerically inefficient, but that it remains instructive to write the equations in that extensive format. The resulting size of the uncertainty of the best estimators due to $\sigma_{Y_{ij}}$ becomes

$$\sigma_{\hat{\boldsymbol{a}}}^{(Y_{ij})} = \frac{\partial \hat{\boldsymbol{a}}}{\partial Y_{ij}} \sigma_{Y_{ij}} . \tag{45}$$

Hence, uncertainties without entry-to-entry correlations, such as statistical uncertainties, are propagated to the estimated parameters as

$$\sigma_{\hat{\boldsymbol{a}}}^{(Y_{\text{uncorr.}})} = \sqrt{\sum_{i,j} \left(\frac{\partial \hat{\boldsymbol{a}}}{\partial Y_{ij}} \sigma_{Y_{ij}} \right)^2} , \tag{46}$$

where the square root and the exponentiation are performed coefficient-wise, while uncertainties that have full entry-to-entry correlations are propagated as

$$\sigma_{\hat{\boldsymbol{a}}}^{(Y_{\text{corr.}})} = \sum_{i,j} \frac{\partial \hat{\boldsymbol{a}}}{\partial Y_{ij}} \sigma_{Y_{ij}} . \tag{47}$$

In examples 1 and 2 (Sects. 4 and 6), the templates are each generated with 40 000 *events* and independently from each other. Hence, all their entries have statistical uncertainties. Using Eq. (46), the statistical uncertainties from Y under Gaussian approximation are ± 0.025 in Example 1, and ± 0.042 and ± 0.060 for the two parameters in Example 2.

The equations for error propagation of the Linear Template Fit with relative uncertainties (Sect. 7) are altogether similar to the presented equations and do not need to be repeated. However, since relative uncertainties are considered, the partial derivatives for error propagation in Eqs. (46) and (47)

become

$$\frac{\partial \hat{\boldsymbol{a}}}{\partial Y_{tj}} \rightarrow \frac{\partial \hat{\boldsymbol{a}}}{\partial \log Y_{tj}} \,. \tag{48}$$

> Example 2 with log-normal uncertainties would report statistical uncertainties from Y of ± 0.052 and ± 0.066 for the two fit parameters.

Note that the uncertainties in Y are not considered within the fit, but propagated separately, since it is assumed that these are negligibly small and independent of the model parameter $\boldsymbol{\alpha}$, and cannot be constrained by the data.

8.6 Full uncertainty

The covariance matrix for the best estimators, including all uncertainty components that are considered in the Linear Template Fit, are calculated from Eqs. (42) and (43):

$$V_{\hat{\boldsymbol{a}}} = \mathbb{V}_{\hat{\boldsymbol{a}}} + \sum_l \boldsymbol{\sigma}_{\hat{\boldsymbol{a}}}^{(s_{(l)})} (\boldsymbol{\sigma}_{\hat{\boldsymbol{a}}}^{(s_{(l)})})^{\mathrm{T}} \tag{49}$$

Further uncertainties, which are not explicitly considered in the fit, can be propagated to the result by using Eqs. (42) and (43) or also Eqs. (46) and (47).

8.7 Error (re-)scaling

When multiple data points are considered in the fit and they are correlated through some (systematic) uncertainties, the best-fit estimators may become biased [64–69]. This is mainly because uncertainties with bin-to-bin correlations are commonly valid as relative uncertainties, or so-called multiplicative uncertainties, such as normalization uncertainties, but those are included in the fit with absolute values (i.e., as variances). Since the data are subject to random noise from statistical fluctuations, or simply because of the different size of the values, for example when considering different data sets, the result may become biased since smaller values are effectively preferred by the fit. While such a bias is largely reduced or even absent from first principles when working with relative uncertainties, it may still be useful to discuss possible solutions to this problem, and several solutions are discussed in the literature (see e.g. Ref. [69]).

A common solution is to consider (multiplicative) uncertainties with bin-to-bin correlations through their relative values in the χ^2 computation, and their relative values are multiplied coefficient-wise with the prediction(s) rather than the data values [68]. However, since the minimization of χ^2 is then no longer linear in the fit parameters, there is no closed analytic solution for the best estimators. However, an approximately equivalent result is obtained when calculating the covariance matrices and systematic shifts (cf. Eq. (31)) from theoretical predictions [66], and the coefficients of the shifts

are recalculated according to

$$s_{(l),i} \rightarrow \frac{s_{(l),i}\, y_i}{d_i} = \mathrm{s}_{(l),i}\, y_i, \tag{50}$$

and analogously for covariance matrices. Various options for the predictions $\boldsymbol{y}$ for this *error rescaling* may be considered, and the prediction $\boldsymbol{y}$ could be chosen to be

- One of the template distributions, $\boldsymbol{y}_{(j)}$;
- The prediction of the linearized model with ad hoc selected parameters $\boldsymbol{\alpha}$, $\hat{\boldsymbol{y}}(\boldsymbol{\alpha})$;
- The best estimator of the linear model, $\hat{\boldsymbol{y}}(\hat{\boldsymbol{a}})$; or
- The Linear Template Fit may be iteratively repeated with the best estimator, $\hat{\boldsymbol{y}}(\hat{\boldsymbol{a}})$.

The rather compact equations of the Linear Template Fit are well suited for an iterative algorithm. Such procedures are equivalent to an alternative formulation of the χ^2 quantity, but the convergence of such iterative algorithms should be critically assessed.

9 Linear Template Fit with a detector response matrix

Many measurements need to be corrected for detector effects like resolution or acceptance in order to compare the measurement with theoretical predictions. This is referred to as *unfolding* (for reviews see e.g. [10,74–79]) and is commonly performed by first simulating the detector response and representing it in terms of a *response matrix A*,

$$\boldsymbol{y} = A\boldsymbol{x} \,, \tag{51}$$

where $\boldsymbol{y}$ denotes the measured detector-level distribution and $\boldsymbol{x}$ the "true" underlying distribution. However, it is not straightforward to apply the inverse response matrix to the distribution $\boldsymbol{y}$, since the unfolding problem represents an ill-posed inverse problem, or the matrix A needs to be a square matrix, and thus more sophisticated unfolding algorithms need to be employed to determine the "true" distribution $\boldsymbol{x}$ [80–85]. When performing a parameter determination, however, it is equivalent whether using the unfolded distribution x together with templates of the predictions, or alternatively using the detector-level measurement y and applying the response matrix to the templates instead. These two procedures are supposed to be equivalent if the unfolding problem and algorithm are unbiased, and are referred to as *folding*, *up-folding*, or *forward folding*. Because the simulation of the detector effects and the determination of the response matrix A may be computationally expensive, as is the case for complex particle physics experiments, it may be reasonable to determine the matrix A only once and with high precision, and then apply it to the templates.

The inclusion of the detector effects in the Linear Template Fit is straightforward and is realized by the simple substitution

$$Y \rightarrow AX \tag{52}$$

in Eqs. (32) and (33), where X represents the template matrix without detector effects at the "truth" or "particle" level. When using relative uncertainties, the substitution becomes

$$\log(Y) \rightarrow \log(AX), \tag{53}$$

and the full expression of the Linear Template Fit becomes

$$\begin{pmatrix} \hat{\alpha} \\ \hat{\epsilon} \end{pmatrix} = \mathcal{F}(\log d - \log(AX)\bar{m}) \quad \text{using}$$

$$\mathcal{F} = \left(\left(\log(AX)\tilde{M} \ S \right)^{\mathrm{T}} \mathbb{W} \left(\log(AX)\tilde{M} \ S \right) \right.$$

$$\left. + \begin{pmatrix} 0_k & 0 \\ 0 & 1_l \end{pmatrix} \right)^{-1} \left(\log(AX)\tilde{M} \ S \right)^{\mathrm{T}} \mathbb{W}. \tag{54}$$

Since Y and AX are $i \times j$ matrices, the index t for the number of entries ("bins") on the truth level was introduced. Thus, A is an $i \times t$ matrix and X has dimension $t \times j$, and in order to obtain a higher resolution, it may be reasonable to consider a non-square response matrix A, i.e., $i \neq t$. It is important that the detector response A is required to be independent from the physics parameter of interest.

The response matrix A is typically associated with uncertainties, and these may be entry-to-entry correlations (like systematic or "model" uncertainties) or without such correlations (like statistical uncertainties). Note, that the response matrix is commonly normalised and therefore the statistical uncertainties are without entry-to-entry correlations only in an approximation.

Uncertainties of the best estimators that arise from uncertainties in A can be obtained from linear error propagation. After applying Eq. (52), the partial derivative of Eq. (32) is

$$\frac{\partial \hat{a}}{\partial A_{it}} = \mathcal{D}^{-1} \left[\left(\mathbb{1}_{(it)} X \tilde{M} \ 0_{il} \right)^{\mathrm{T}} \mathbb{W} (d - AX\bar{m}) \right.$$

$$- \left(AX\tilde{M} \ S \right)^{\mathrm{T}} \mathbb{W} \mathbb{1}_{(it)} X\bar{m}$$

$$- \left[\left(AX\tilde{M} \ S \right)^{\mathrm{T}} \mathbb{W} \left(\mathbb{1}_{(it)} X \tilde{M} \ 0_{il} \right) \right.$$

$$\left. + \left(\mathbb{1}_{(it)} X \tilde{M} \ 0_{il} \right)^{\mathrm{T}} \mathbb{W} \left(AX\tilde{M} \ S \right) \right] \hat{a} \right]. \tag{55}$$

Consequently, uncertainties in the response matrix A without or with entry-to-entry correlations are propagated to the fitted results as

$$\sigma_{\hat{a}}^{(A_{\mathrm{uncorr.}})} = \sqrt{ \sum_{i,t} \left(\frac{\partial \hat{a}}{\partial A_{it}} \sigma_{A_{it}} \right)^2 } \quad \text{and}$$

$$\sigma_{\hat{a}}^{(A_{\mathrm{corr.}})} = \sum_{i,t} \frac{\partial \hat{a}}{\partial A_{it}} \sigma_{A_{it}}, \tag{56}$$

respectively, where $\sigma_{A_{it}}$ denotes the size of the uncertainty in the element A_{it}, and the power and square-root operations are applied coefficient-wise. When using a response matrix, the uncertainties in the template matrix X (cf. Sect. 8.5) are propagated using

$$\frac{\partial \hat{a}}{\partial X_{tj}} = \mathcal{D}^{-1} \left[\left(A\mathbb{1}_{(tj)} \tilde{M} \ 0_{il} \right)^{\mathrm{T}} \mathbb{W} (d - AX\bar{m}) \right.$$

$$- \left(AX\tilde{M} \ S \right)^{\mathrm{T}} \mathbb{W} A\mathbb{1}_{(tj)} \bar{m}$$

$$- \left[\left(AX\tilde{M} \ S \right)^{\mathrm{T}} \mathbb{W} \left(A\mathbb{1}_{(tj)} \tilde{M} \ 0_{il} \right) \right.$$

$$\left. + \left(A\mathbb{1}_{(tj)} \tilde{M} \ 0_{il} \right)^{\mathrm{T}} \mathbb{W} \left(AX\tilde{M} \ S \right) \right] \hat{a} \right]. \tag{57}$$

Note that for the Linear Template Fit with relative uncertainties, it is reasonable to consider uncertainties in A_{it} as absolute uncertainties, rather than relative, because the elements A_{it} represent already relative quantities and may be interpreted as probabilities. Therefore, one employs the partial derivative $\partial \hat{a} / \partial A_{ij}$ for error propagation, instead of $\partial \hat{a} / \partial \log A_{ij}$, although elsewhere relative uncertainties are considered. This partial derivative is straightforward to derive. The propagation of uncertainties in the templates X_{tj}, however, becomes somewhat more complicated when using a response matrix A, since one has to calculate the partial derivative $\partial \hat{a} / \partial \log X_{tj}$ for error propagation of the relative uncertainties $\sigma_{X_{tj}}$.[4]

10 Considerations, validations, and cross-checks

As in any optimization problem, the result of the Linear Template Fit needs to be validated and cross-checked. Due to its close similarity to the iterative Gauß–Newton minimization algorithm of nonlinear least squares [5,50,51,86,87], the Linear Template Fit is expected to result in an unbiased and optimal result, in particular if the templates provide a sufficiently accurate approximation of the model and its Jacobi matrix at the best estimator. In fact, when assuming that the model $\lambda(\alpha)$ could be used with continuous values of α for inference, the best estimator from the Linear Template Fit becomes equivalent to the best estimator that would be obtained when using $\lambda(\alpha)$ directly, if two approximations are

---

[4] In brief, the partial derivative $\partial \hat{a} / \partial \log X_{tj}$ is similar to Eq. (57), but each element of the $i \times j$ matrix $(A\mathbb{1}_{(tj)})$ becomes $(A\mathbb{1}_{(tj)})_{ij} = (A\mathbb{1}_{(tj)} X_{tj})_{ij} / (AX)_{ij}$.

exactly fulfilled,[5]

$$\lambda(\hat{\boldsymbol{\alpha}}) \approx \hat{\mathbf{y}}(\hat{\boldsymbol{\alpha}}) = Y\bar{\mathbf{m}} + Y\tilde{M}\hat{\boldsymbol{\alpha}} \quad \text{and} \tag{58}$$

$$\left.\frac{\partial\lambda(\boldsymbol{\alpha})}{\partial\boldsymbol{\alpha}}\right|_{\boldsymbol{\alpha}=\hat{\boldsymbol{\alpha}}} \approx \left.\frac{\partial\hat{\mathbf{y}}(\boldsymbol{\alpha})}{\partial\boldsymbol{\alpha}}\right|_{\boldsymbol{\alpha}=\hat{\boldsymbol{\alpha}}} = Y\tilde{M} . \tag{59}$$

The first equation requires that the model is correctly represented by the interpolated templates at the best estimator, which is needed for the value of the best estimator, and the second equation demands that the first derivatives are correctly approximated, which is a requirement for the error propagation and minimization. These equations taken together represent just a linear approximation of $\lambda(\boldsymbol{\alpha})$ at $\hat{\boldsymbol{\alpha}}$, but using the templates only. However, since for the motivation of the Linear Template Fit it was assumed that $\lambda(\boldsymbol{\alpha})$ cannot be used for inference itself, these two approximations also cannot be validated directly. In the following, some possible studies for the validation of the results and the practical application of the Linear Template Fit are discussed.

10.1 Template range

If the model is nonlinear in the parameters $\boldsymbol{\alpha}$, an unbiased result may only be obtained if the best estimator $\hat{\boldsymbol{\alpha}}$ lies within the interval of the reference values,

$$\hat{\boldsymbol{\alpha}} \in [\dot{\boldsymbol{\alpha}}_{(1)}, \dot{\boldsymbol{\alpha}}_{(j_{\max})}] . \tag{60}$$

This is because the templates provide the approximation of the model and its first derivatives, but an extrapolation beyond the template range is inappropriate for a nonlinear model. If Eq. (60) does not hold, additional templates at further reference points should be generated and the Linear Template Fit should be repeated with an improved selection of reference values. For similar reasons, the templates should be in the close vicinity of the best estimator, such that the linear approximations are justified. As a rule of thumb, the distance between the reference points should not greatly exceed the size of the variance of the best estimator

$$\dot{\boldsymbol{\alpha}}_{(j)} - \dot{\boldsymbol{\alpha}}_{(j+1)} \lesssim \boldsymbol{\sigma}_{\hat{\boldsymbol{\alpha}}} . \tag{61}$$

For similar reasons, the best estimator should ideally be found in the center of the template range, rather than at its boundary,

$$\hat{\boldsymbol{a}} - \dot{\boldsymbol{\alpha}}_{(0)} \approx \dot{\boldsymbol{\alpha}}_{(j_{\max})} - \hat{\boldsymbol{a}} . \tag{62}$$

However, for rather linear parameter dependencies, these requirements may be more relaxed.

The choice of the reference values may be related to the size of the uncertainties of the best estimator. If one considers the uncertainties of the best estimator to be normally dis-

tributed,[6] as a consequence, the model also needs to exhibit a sufficiently linear behavior at least within a few σ. As a rule of thumb, one could demand that the reference values are approximately within 3 standard deviations:

$$\dot{\boldsymbol{\alpha}}_{(j)} \in [\hat{\boldsymbol{\alpha}} - 3\boldsymbol{\sigma}_{\hat{\boldsymbol{\alpha}}}, \hat{\boldsymbol{\alpha}} + 3\boldsymbol{\sigma}_{\hat{\boldsymbol{\alpha}}}] . \tag{63}$$

10.2 The value of $\hat{\chi}^2$, its uncertainty, and the partial χ^2

The value of χ^2 at the minimum, $\hat{\chi}^2$, is a meaningful quantity to assess the agreement between the data and the model. Since the calculation of χ^2 is not explicit in the Linear Template Fit, its value needs to be calculated separately, and $\hat{\chi}^2$ is calculated from data and the templates at $\hat{\boldsymbol{a}}$ using the somewhat simplified equation

$$\hat{\chi}^2 = \left(\boldsymbol{d} - Y\bar{\boldsymbol{m}} - Y\tilde{M}\hat{\boldsymbol{a}}\right)^{\mathrm{T}} W \left(\boldsymbol{d} - Y\bar{\boldsymbol{m}}\right) . \tag{64}$$

Since the data and the templates are associated with uncertainties, the value of $\hat{\chi}^2$ consequently may also be considered as uncertain. The uncertainty in $\hat{\chi}^2$ is obtained from linear error propagation of the input uncertainties and calculated as

$$\sigma_{\hat{\chi}^2} = \sqrt{\boldsymbol{\xi}^{\mathrm{T}}\mathbb{V}\boldsymbol{\xi} + \sum_l (\boldsymbol{\xi}\boldsymbol{s}_{(l)}^{\mathrm{T}})^2} , \tag{65}$$

where the i-vector $\boldsymbol{\xi}$ is introduced and whose coefficients represent the partial derivatives $\frac{\partial\hat{\chi}^2}{\partial d_i}$, which are calculated as $\xi_i = 2W_i(\boldsymbol{d} - \hat{\mathbf{y}}(\hat{\boldsymbol{a}}))$, and W_i denotes the ith row of the matrix W. Also, the uncertainties in the templates Y may be propagated to $\hat{\chi}^2$ with similar equations. The interpretation of $\sigma_{\hat{\chi}^2}$ is not trivial, but it may well be that this quantity has interesting features and may serve as an additional quality parameter in the future.

The value of χ^2 is inherently obtained from a sum of various sources of uncertainties. It is instructive to calculate the contribution from any uncertainty source ℓ to $\hat{\chi}^2$ separately using [92]

$$\hat{\chi}_\ell^2 = (\boldsymbol{d} - \hat{\mathbf{y}}(\hat{\boldsymbol{a}}))^{\mathrm{T}} W V_\ell W (\boldsymbol{d} - \hat{\mathbf{y}}(\hat{\boldsymbol{a}})) , \tag{66}$$

where V_ℓ is any of the covariance matrices, see Eq. (31). It is obvious that $\hat{\chi}^2 = \sum_\ell \hat{\chi}_\ell^2$. This may become particularly useful when multiple measurements are considered in the fit, and thus their individual contributions to $\hat{\chi}^2$, i.e. the partial χ^2's, may be calculated by summing the respective $\hat{\chi}_\ell^2$ values.

---

[5] One would have to further require that there is no second minima when using $\lambda(\boldsymbol{\alpha})$ within the range covered by the reference values.

[6] A reasonable argument for a normally distributed parameter can also be obtained from previous analyses, for example, if these published normally distributed uncertainties, or if the results are input to some combination procedure which considers uncertainties to be normally distributed [48,88–91].

10.3 Interpretation of the χ_j^2 values as a cross-check

An obviously simple validation of the result from the Linear Template Fit is performed by comparing $\hat{\chi}^2$ with the χ^2 values calculated from the individual templates, defined as

$$\chi_j^2 = (d - y_{(j)})^T W (d - y_{(j)}) \quad \text{or}$$
$$\chi_j^2 = (\log d - \log y_{(j)})^T W (\log d - \log y_{(j)}), \qquad (67)$$

whether absolute or relative normally distributed uncertainties are considered.

An alternative best estimator, denoted in the following as $\check{\alpha}$, is obtained from the interpretation of all of the χ_j^2 values, and serves as a validation of the best estimator from the Linear Template Fit ($\hat{\alpha}$). Based on the assumption that the χ^2 distribution $\chi^2(\alpha)$ follows a second-degree polynomial, the so-called χ^2 parabola, its minimum is interpreted as best estimator. For a univariate problem, when introducing

$$\chi^2 = \begin{pmatrix} \chi_1^2 \\ \vdots \\ \chi_j^2 \end{pmatrix}, \quad C_{(\text{pol-2})} = \begin{pmatrix} 1 & \dot{\alpha}_1 & \dot{\alpha}_1^2 \\ \vdots & \vdots & \vdots \\ 1 & \dot{\alpha}_j & \dot{\alpha}_j^2 \end{pmatrix} \quad \text{and}$$

$$\boldsymbol{\vartheta} = \begin{pmatrix} \vartheta_0 \\ \vartheta_1 \\ \vartheta_2 \end{pmatrix}, \qquad (68)$$

with χ_j^2 from Eq. (67), the best estimator of the model parameter is given by the stationary point of the polynomial

$$\check{\alpha} = \frac{\hat{\vartheta}_1}{-2\hat{\vartheta}_2}, \qquad (69)$$

where the best estimators of the (three) polynomial parameters are $\hat{\boldsymbol{\vartheta}} = (C^T C)^{-1} C \chi^2$. An illustration for Example 1 is shown in Fig. 5. The uncertainty in $\check{\alpha}$ is given by the criterion $\chi_{\min}^2 + 1$ [10] and hence

$$\sigma_{\check{\alpha}} = \frac{\sqrt{\hat{\vartheta}_1^2 - 4\hat{\vartheta}_2(\hat{\vartheta}_0 - \check{\chi}^2 - 1)}}{-2\hat{\vartheta}_2} \quad \text{using}$$
$$\check{\chi}^2 = (\chi^2 - C\hat{\boldsymbol{\vartheta}})^T (\chi^2 - C\hat{\boldsymbol{\vartheta}}). \qquad (70)$$

The equations for the multivariate case are straightforward and not repeated here. This alternative best estimator, $\check{\alpha}$, as well as its variance $\sigma_{\check{\alpha}}$ and the minimum $\check{\chi}^2$, should be sufficiently consistent with the results from the Linear Template Fit:

$$\check{\alpha} \sim \hat{\alpha}, \quad \sigma_{\check{\alpha}} \sim \sigma_{\hat{\alpha}} \quad \text{and} \quad \check{\chi}^2 \sim \hat{\chi}^2, \qquad (71)$$

and may serve as a valuable cross-check. As a quick and practical visual test, if the templates line up on the parabolic fit (as seen in Fig. 5), the problem is essentially linear, and the Linear Template Fit will provide an unbiased and optimal best estimator. Note that for the calculation of Eq. (69), at least three templates are required, whereas the minimum number
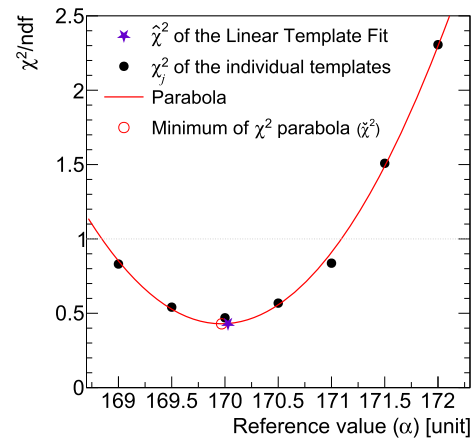

Fig. 5 Illustration of various χ^2/n_{df} values in the scope of the Linear Template Fit for Example 1 (Sect. 4). The red star indicates the best-fit χ^2 of the Linear Template Fit, $\hat{\chi}^2$. The full circles display the χ^2 values of the individual templates and the red line displays the χ^2 parabola calculated from them, and its minimum is indicated with an open circle ($\check{\chi}^2$, see text)

of templates for the univariate Linear Template Fit is only two.

10.4 Power law factor for the linear regression

In the Linear Template Fit, the functions $\hat{y}(\alpha)$ are (multidimensional) linear functions, see Eqs. (15), (27), and (38). Although the use of higher-degree polynomials or multivariate polynomials with interference terms could in principle be considered (see next Sect. 11), in such cases the optimization problem cannot be solved analytically, and no closed matrix expression for the best estimators is found. However, an analytic solution is still obtained when only power law factors are introduced, and in a univariate Linear Template Fit the model approximation becomes

$$y_i(\alpha; \theta_0, \theta_1) = \theta_0^{(i)} + \theta_1^{(i)} \alpha^\gamma \qquad (72)$$

with $\gamma > 0$ being a real number. The matrix M, Eq. (10), then has elements $M_{j,0} = 1$ and $M_{j,1} = \dot{\alpha}_j^\gamma$. In the multivariate case, different power law factors for each fit parameter can be introduced, and one writes γ_k and has $M_{j,(k+1)} = \dot{\alpha}_{(j),k}^{\gamma_k}$. The best estimators of the multivariate Linear Template Fit, Eq. (32), then become

$$\hat{\alpha}_k = \left(\mathcal{F}(d - Y\bar{m}) \right)_k^{\frac{1}{\gamma_k}}. \qquad (73)$$

This solution can immediately be seen by a variable substitution of α^γ in Eqs. (32) and (72). The application of the power law factor(s) γ when working with relative uncertainties (see Sect. 7) is straightforward, since all equations remain linear in α_k. Similarly, the equations for the error propagation (see Sect. 8) can easily be adopted with that variable substitu-

tion. For instance, the elements of the systematic shifts (see Eq. (43)) become

$$(\sigma_{\hat{a}}^{(s_{(l)})})_k = \frac{(\mathcal{F}s_{(l)})_k}{\gamma_k \hat{\alpha}_k^{\gamma_k - 1}}, \tag{74}$$

and similarly for other uncertainty components or covariance matrices. The power law factor γ may be of practical use in order to validate the assumptions of the linearity of the regression analysis, or to improve the fit, such as when it is known that the model is proportional to α^2. This would be the case in determinations of the strong coupling constant from n-jet production cross sections at the LHC, which are proportional to $\alpha_s^n(m_Z)$ at the leading order in perturbative quantum chromodynamics (QCD).

10.5 Finite differences

In a use case where a larger number of templates are available, various choices of subsets of these templates can become input to the Linear Template Fit, and some of such potential choices have already been suggested in the previous paragraphs. In the case where only two (or three) templates are provided as input to the Linear Template Fit, the selected templates may have a special meaning, because the templates can also be considered as a numerical first derivative through finite differences. Such numerically calculated derivatives are the essence of many gradient descent optimization methods [5,49,50,86,87], and it is instructive to study the relation.

As an example, let us consider a univariate problem where the objective function χ^2 is given by Eq. (8). Close to the minimum, χ^2 is nearly quadratic, and the Newton step, $\Delta\alpha = -(\chi^2)'/(\chi^2)''$, exhibits a good convergence [5,50,51,87]. The first and second derivatives at an expansion point α_0 are

$$(\chi^2)' = -2\lambda'^{\mathrm{T}}(\alpha_0)W(d - \lambda(\alpha_0)) \quad \text{and}$$
$$(\chi^2)'' = 2\lambda'^{\mathrm{T}}(\alpha_0)W\lambda'(\alpha_0), \tag{75}$$

where the second derivatives λ'' are zero or neglected in the latter equation. Hence, one obtains the estimator from the Newton step as

$$\hat{\alpha}_{(\text{Newton})} = \alpha_0 + \left[\lambda'^{\mathrm{T}}(\alpha_0)W\lambda'(\alpha_0)\right]^{-1}$$
$$\times \lambda'^{\mathrm{T}}(\alpha_0)W(d - \lambda(\alpha_0)), \tag{76}$$

and the calculation of the first derivative could be done numerically using a finite difference with nonzero step size h

$$\lambda'(\alpha_0) = \frac{\lambda(\alpha_0 + h) - \lambda(\alpha_0)}{h}. \tag{77}$$

The analogous Linear Template Fit makes use of the templates at α_0 and $\alpha_0 + h$ and defines the following matrices:

$$Y = \begin{pmatrix} \lambda(\alpha_0) & \lambda(\alpha_0 + h) \end{pmatrix} \quad \text{and} \quad M = \begin{pmatrix} 1 & \alpha_0 \\ 1 & \alpha_0 + h \end{pmatrix}, \quad \text{hence}$$

$$M^+ = \begin{pmatrix} \frac{\alpha_0 + h}{h} & -\frac{\alpha_0}{h} \\ -\frac{1}{h} & \frac{1}{h} \end{pmatrix} = \begin{pmatrix} \bar{\boldsymbol{m}}^{\mathrm{T}} \\ \tilde{\boldsymbol{m}}^{\mathrm{T}} \end{pmatrix}. \tag{78}$$

From the Linear Template Fit, Eq. (17), when resolving $\bar{\boldsymbol{m}}$, one obtains the best estimator

$$\hat{\alpha} = \alpha_0 + \left[(Y\tilde{\boldsymbol{m}})^{\mathrm{T}}W(Y\tilde{\boldsymbol{m}})\right]^{-1}(Y\tilde{\boldsymbol{m}})^{\mathrm{T}}W(d - \lambda(\alpha_0)), \tag{79}$$

and since $Y\tilde{\boldsymbol{m}} = \lambda'(\alpha_0)$, it can be directly seen that it is equivalent to the estimator $\hat{\alpha}_{(\text{Newton})}$ from the Newton step using a numerical finite difference.

11 Nonlinear model approximation

The Linear Template Fit can be regarded such that linear functions are fitted to every row i of the template matrix Y.[7] In the following, nonlinear functions are considered for the model $\lambda_i(\boldsymbol{\alpha})$ in each "bin" i. For a simplified discussion, the formulae for only a univariate model are sometimes discussed, while the multivariate case is commonly straightforward. Also, the concepts of the γ factor, relative uncertainties, or "systematic shifts" s_l can equally be considered in a multivariate formulation.

The model can be approximated from the templates in every bin i with an nth-degree polynomial function

$$\lambda_i(\alpha) \approx \mathrm{y}_i(\alpha; \hat{\boldsymbol{\theta}}_{(i)}) =: \hat{\mathrm{y}}_i(\alpha) \quad \text{using}$$
$$\mathrm{y}_i(\alpha; \boldsymbol{\theta}_{(i)}) = \theta_0^{(i)} + \theta_1^{(i)}\alpha + \cdots + \theta_n^{(i)}\alpha^n. \tag{80}$$

The best estimators for the polynomial parameters are obtained from regression analysis,

$$\hat{\boldsymbol{\theta}}_{(i)} = \begin{pmatrix} \hat{\theta}_0 \\ \hat{\theta}_1 \\ \vdots \\ \hat{\theta}_n \end{pmatrix}_{(i)} = \mathcal{M}^+ \begin{pmatrix} \mathrm{y}_{(1),i} \\ \vdots \\ \mathrm{y}_{(j),i} \end{pmatrix}, \tag{81}$$

where $\mathrm{y}_{(j),i}$ is the ith value of the jth template, and the g-inverse is

$$\mathcal{M}^+ = \left(\mathcal{M}^{\mathrm{T}}\mathcal{M}\right)^{-1}\mathcal{M}^{\mathrm{T}} \quad \text{using here}$$

---

[7] Conceptionally, (nonlinear) functions can also be fitted to the elements of each column j, and so interpreting the shape of the templates and the data distributions. Subsequently, the resulting functions, when normalized, can be interpreted as probabilities and are often used as input to binned or un-binned maximum-likelihood optimizations [31,35,40]. Similarly, the parameters of such a transformation can also become again input to a (Linear) Template Fit. An example of this approach is discussed in Appendix B.

$$
\mathcal{M} = \begin{pmatrix} 1 & \dot{\alpha}_1 & \dots & \dot{\alpha}_1^n \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \dot{\alpha}_j & \dots & \dot{\alpha}_j^n \end{pmatrix} . \tag{82}
$$

For reasons outlined in Sect. 3, an unweighted regression is considered. Using this model approximation in the objective function of the fitting problem, Eq. (2), the χ^2 expression for a univariate model becomes of order $\mathcal{O}(2n)$ in α,

$$
\chi^2_{\mathcal{O}(2n)} := \left(d - \hat{y}(\alpha) \right)^{\mathrm{T}} W \left(d - \hat{y}(\alpha) \right) \tag{83}
$$
$$
= \left(d - Y m_0 - Y m_1 \alpha - \cdots - Y m_n \alpha^n \right)^{\mathrm{T}}
$$
$$
\times W \left(d - Y m_0 - Y m_1 \alpha - \cdots - Y m_n \alpha^n \right) , \tag{84}
$$

where the vectors m were used as the representation of $\mathcal{M}^+$ according to

$$
\mathcal{M}^+ =: \begin{pmatrix} m_0^{\mathrm{T}} \\ m_1^{\mathrm{T}} \\ \vdots \\ m_n^{\mathrm{T}} \end{pmatrix} . \tag{85}
$$

It is obvious that for $n = 1$, this is the Linear Template Fit, and the generalizations to the multivariate fit (Eq. (30)), and with relative uncertainties (Eq. (39)), are straightforward. Note that for different degrees n, the vectors m differ unless the equations are not rewritten for orthogonal polynomials [93].

Since the first derivative of $\chi^2_{\mathcal{O}(2n)}$ is nonlinear, in α, no analytic solution of the optimization problem is found. Therefore, in the following, two iterative algorithms to find the minimum of the objective function $\chi^2_{\mathcal{O}(2n)}$ are discussed, which refer to the Gauß–Newton and the Newton algorithm (see also Refs. [5,50,51,87]). In both cases, if the starting value is identified with the best estimator from the Linear Template Fit, $\alpha = \hat{\alpha}$, the step size of the next iterative step can be interpreted as convergence criterion or uncertainty of the Linear Template Fit. Reasonable values for n are $n = 2, 3$, or at most 4 (since $\mathcal{M}$ becomes quickly ill-conditioned for large n, e.g., Refs. [94,95] and therein), while for $n = 1$, consistent results with the Linear Template Fit are found.

11.1 Linearization of the nonlinear approximation $\hat{y}(\alpha)$

An iterative minimization algorithm based on first derivatives is defined through linearly approximating the (nonlinear) function $\hat{y}(\alpha)$, Eq. (80). For the purpose of validating the result from the Linear Template Fit, we consider $\hat{\alpha}$ as the expansion point, although this could be any other value as well. Hence, the linear approximation of $\hat{y}(\alpha)$ in every bin i is

$$
\tilde{y}_i(\alpha) \approx \hat{y}_i(\hat{\alpha}) + \left. \frac{\partial \hat{y}_i(\alpha)}{\partial \alpha} \right|_{\alpha=\hat{\alpha}} (\alpha - \hat{\alpha})
$$
$$
= Y \left(\sum_0^n (1-n) m_{(n)} \hat{\alpha}^n \right) + Y \left(\sum_0^n n m_{(n)} \hat{\alpha}^{n-1} \right) \alpha
$$
$$
=: Y \bar{\tilde{m}} + Y \tilde{\tilde{m}} \alpha . \tag{86}
$$

The last term defines the shorthand notations $\bar{\tilde{m}}$ and $\tilde{\tilde{m}}$. From comparison with Eq. (15), it is seen that an alternative best estimator of the template fit (see Eq. (19)) is obtained as

$$
\tilde{\tilde{\alpha}} = \frac{(Y \tilde{\tilde{m}})^{\mathrm{T}} W}{(Y \tilde{\tilde{m}})^{\mathrm{T}} W Y \tilde{\tilde{m}}} (d - Y \bar{\tilde{m}}) . \tag{87}
$$

The difference $\hat{\alpha} - \tilde{\tilde{\alpha}}$ is supposed to become small for a valid application of the Linear Template Fit, and one should find

$$
|\hat{\alpha} - \tilde{\tilde{\alpha}}| \ll \sigma_{\hat{\alpha}} . \tag{88}
$$

If Eq. (88) holds for a particular application, then nonlinear effects are negligible, and the best estimator of the Linear Template Fit is insensitive to nonlinear effects in the model λ. If Eq. (88) does not hold, then it should be studied if the nonlinearity is present in the vicinity of $\hat{\alpha}$ or if it is rather introduced by a long-range nonlinear behavior of $\lambda(\alpha)$. In the latter case, the range of the templates should be reduced such that one has about

$$
\dot{\alpha}_{\max} - \dot{\alpha}_{\min} \lesssim 2\sigma_{\hat{\alpha}} \tag{89}
$$

and the Linear Template Fit should be repeated. Note that if the templates are subject to statistical fluctuations and if only a few templates are provided, then a nonlinear approximation is more sensitive to those fluctuations than linear approximations, and $|\hat{\alpha} - \tilde{\tilde{\alpha}}|$ may become larger. In such an application, $\sigma_{\hat{\alpha}}^Y$ should also be considered as a relevant source of uncertainty (see Eqs. (46) and (47)).

In Example 1, when using a second-degree polynomial, a value $|\hat{\alpha} - \tilde{\tilde{\alpha}}| = 0.02$ is found, which is much smaller than $\sigma_{\hat{\alpha}} = \pm 0.41$ and of similar size than the statistical uncertainties from Y (see Sect. 8.5). When working with log-normal uncertainties, a value $|\hat{\alpha} - \tilde{\tilde{\alpha}}| = 0.05$ is found. Although the model is highly nonlinear in $\log(y)$, the assumptions of the linear template fit are well justified, since the templates are in the close vicinity of the best estimator. In Example 2, for a second-degree approximation, values of $|\hat{\alpha}_0 - \tilde{\tilde{\alpha}}_0| = 0.03$ and $|\hat{\alpha}_1 - \tilde{\tilde{\alpha}}_1| = 0.14$ are found. When further considering interference terms (see Appendix C), the values are 0.01 and 0.11, respectively. All these values are smaller than the respective statistical uncertainties in the data, although the values suggest that for α_1 (the variance of the Gaussian model), a finer grid for the reference values may be preferred for a more accurate linear approximation.

Furthermore, the notation of Eq. (86) suggests an alternative interpretation of the nonlinear behavior of the model. The vectors $\bar{\tilde{m}}$ and $\tilde{\tilde{m}}$ may be interpreted in terms of uncertainties of $\bar{m}$ and $\tilde{m}$ when writing

$$
\sigma_{\bar{m}} := \bar{\tilde{m}} - \bar{m} \quad \text{and} \quad \sigma_{\tilde{m}} := \tilde{\tilde{m}} - \tilde{m} . \tag{90}
$$

Through linear error propagation, one can define an uncertainty in $\hat{\alpha}$ as

$$
\sigma_{\hat{\alpha}}^{(m)} = \sum_j \left(- FY \mathbb{1}_j \sigma_{\tilde{m}_j} + D^{-1} \left[(Y \mathbb{1}_j)^{\mathrm{T}} W (d - Y \bar{m}) \right] \right.
$$

$$- \left[(Y \mathbb{1}_j)^{\mathrm{T}} WY \tilde{m} + (Y \tilde{m})^{\mathrm{T}} WY \mathbb{1}_j \right] \hat{a} \right] \sigma_{\tilde{m}_j} \bigg)$$

$$= \sum_i \bigg(- F \mathbb{1}_i \sigma_{\overline{Ym}_i} + D^{-1} \big[\mathbb{1}_i^{\mathrm{T}} W (d - Y \bar{m})$$

$$- \left[\mathbb{1}_i^{\mathrm{T}} WY \tilde{m} + (Y \tilde{m})^{\mathrm{T}} W \mathbb{1}_i \right] \hat{a} \big] \sigma_{\widetilde{Ym}_i} \bigg). \tag{91}$$

Note that the sum runs over j in the first and over i in the second expression. The second expression makes use of

$$\boldsymbol{\sigma}_{\overline{Ym}} := Y \boldsymbol{\sigma}_{\bar{m}} \quad \text{and} \quad \boldsymbol{\sigma}_{\widetilde{Ym}} := Y \boldsymbol{\sigma}_{\tilde{m}} \tag{92}$$

and it has the advantage that the summands represent a useful quantifier for every bin i. For a sufficiently linear problem, both the linear sum (Eq. (91)) and the quadratic sum should be sufficiently small. In particular, one should find $\sigma_{\hat{\alpha}}^{(m)} \approx \tilde{\hat{\alpha}} - \hat{\alpha}$ for reasonable nth-degree polynomials, i.e. $n=2$, and also, according to Eq. (88),

$$|\sigma_{\hat{\alpha}}^{(m)}| \ll |\sigma_{\hat{\alpha}}|. \tag{93}$$

11.2 Minimization of $\chi^2_{\mathcal{O}(2n)}$

When $\hat{\alpha}$ is obtained from the Linear Template Fit, the objective function $\chi^2_{\mathcal{O}(2n)}$, Eq. (84), is expected to exhibit a minimum in the close vicinity of $\hat{\alpha}$. This provides good motivation for the application of the Newton optimization at $\hat{\alpha}$, which is based on second derivatives of $\chi^2_{\mathcal{O}(2n)}$, so a parabolic approximation. The Newton distance to the minimum is given by

$$\Delta \alpha = - \frac{(\chi^2_{\mathcal{O}(2n)})'}{(\chi^2_{\mathcal{O}(2n)})''} = - \left(\frac{\partial^2 \chi^2_{\mathcal{O}(2n)}}{\partial^2 \alpha} \bigg|_{\alpha = \hat{\alpha}} \right)^{-1} \frac{\partial \chi^2_{\mathcal{O}(2n)}}{\partial \alpha} \bigg|_{\alpha = \hat{\alpha}} \tag{94}$$

and in the multivariate case by [5,55]

$$\boldsymbol{\Delta \alpha} = - H^{-1} \boldsymbol{g} \big|_{\alpha = \hat{\alpha}}. \tag{95}$$

The gradient vector $\boldsymbol{g}$ and the Hesse matrix H are defined by the first and second derivatives of $\chi^2_{\mathcal{O}(2n)}$, respectively. Both quantities can be calculated directly at an expansion point $\boldsymbol{\alpha_0}$, which was identified in the above equations with $\hat{\boldsymbol{\alpha}}$, since $\chi^2_{\mathcal{O}(2n)}$ is just a known second-degree polynomial and dependent on M, Y, d, S, and W. The value of $\boldsymbol{\Delta \alpha}$ can be interpreted as the expected distance to the minimum (EDM) based on the nonlinear model approximation.[8]

For practical purposes and to validate the result of the Linear Template Fit, one should require that its value be smaller

than the uncertainty from the fit

$$|\Delta \alpha| \ll \sigma_{\hat{\alpha}}. \tag{96}$$

The most reasonable degree for this purpose is $n = 2$ (which is exploited in the following in greater detail), and one expects that $\Delta \alpha$ will be negligibly small. This observation can also be interpreted such that the result from the Linear Td an alternative result from a minimization of $\chi^2_{\mathcal{O}(2n)}$ are consistent. If the reference points are in the vicinity of $\hat{\alpha}$, this is to be expected because of the underlying statistical model of (log-)normally distributed estimators. In contrast, if $\Delta \alpha$ becomes large, it may well be that in such a case not even the reported uncertainties in the best estimator, $\sigma_{\hat{\alpha}}$, can be considered to be normally distributed.

12 The Quadratic Template Fit

Despite all the considerations and cross-checks as discussed in the previous sections, in some problems the linearized model may simply be an inaccurate approximation of the true model (cf. Eq. (27)). Reasons could be that the templates cannot be generated in a sufficiently close vicinity around the best estimator, perhaps for technical reasons, or in multivariate problems some parameters are highly nonlinear, a dimension is poorly constrained and the reference points span a large range, or interference terms between the parameters are non-negligible. In such cases, the quantities $\tilde{\hat{\boldsymbol{\alpha}}} - \hat{\boldsymbol{\alpha}}$, $\sigma_{\hat{\alpha}}^{(m)}$, or $\boldsymbol{\Delta \alpha}$ (Eqs. (88), (91) and (95)) become non-negligible when they are calculated for second-degree polynomials $n = 2$.

An improved approximation of the model is obtained when using a second-order approximation, which may also include interference terms. The equations are summarized in Appendix C (Eqs. (C.7)–(C.9)). Using such a second-degree polynomial model $\hat{\mathbf{y}}(\boldsymbol{\alpha})$ that includes terms $\mathcal{O}(\alpha_k)$, $\mathcal{O}(\alpha_k^2)$, and $\mathcal{O}(\alpha_{k_1} \alpha_{k_2})$ in the optimization problem results in a nonlinear least-squares problem. Therefore, an iterative algorithm is required to obtain the best estimator.

A robust algorithm, denoted as *Quadratic Template Fit*, to obtain the best estimator and to perform a full error propagation is defined as

1. The Linear Template Fit is performed to obtain a first estimator $\hat{\boldsymbol{\alpha}}_{(0)}$, for example using Eq. (32);
2. The Newton algorithm [5,50,51,87], Eq. (95), is employed with a few m iterations in order to obtain improved best estimators $\hat{\boldsymbol{\alpha}}_{(m)} = \hat{\boldsymbol{\alpha}}_{(m-1)} + \boldsymbol{\Delta \alpha}_{(m)}$ and for which the Hesse matrix H is directly analytically calculable;
3. The best estimator and the error calculation are obtained using the linearized approximations for $\tilde{m}$ and $\tilde{M}$ (see Eqs. (86) or (C.13)) in the equations of the Linear Template Fit.

---

[8] When the widely used approximation of the Hesse matrix with the Jacobi matrix of first derivatives is used, $H \approx J^{\mathrm{T}} J$ (*truncated* Hesse), then the Newton step becomes equivalent to the Gauß–Newton step, which is equivalent to linearization of the model, Eq. (87), and one obtains $\boldsymbol{\alpha} + \boldsymbol{\Delta \alpha} = \tilde{\hat{\boldsymbol{\alpha}}}$, when using $J^{\mathrm{T}} J$ as H.

The first step provides the starting point for the Newton algorithm, and since it is obtained with the Linear Template Fit, it is already close to the minimum. The second step provides a fast and stable convergence for several reasons: i) the starting point is already in the vicinity of the minimum, ii) the Hesse matrix is commonly positive definite and it is analytically exactly calculable[9], as well as the gradient vector, and iii) the Newton method has good convergence for nearly quadratic functions [5,46,47], which is the case for $\chi^2_{\mathcal{O}(4)}$. Although the algorithm is expected to converge quickly after a few iterations, several stopping criteria are thinkable, for example when EDM $\ll |\Delta\alpha|$ or $\Delta\alpha_{(m)} \approx \Delta\alpha_{(m-1)}$. The last step provides an even more improved best estimator, but mainly gives access to the equations of the error propagation. Note that in the first step, the true model $\lambda(\alpha)$ is linearly approximated, and in the latter steps, by quadratic functions.

> When applying the Quadratic Template Fit to Example 2, and using $m = 2$ Newton iterations, the best estimator for the mean is found to be 169.90 ± 0.44 and for the variance it is 6.23 ± 0.32. Both values are very close to the best estimators from the Linear Template Fit; see Sect. 6. The EDM $\Delta\alpha$ and $\tilde{\hat{\alpha}} - \hat{\alpha}$ are found to be smaller than 10^{-4} for both parameters, and $\sigma_{\hat{\alpha}}^{(m)}$ is equally small.

13 Example 3: the Quadratic Template Fit

As an application of the Quadratic Template Fit, an example is constructed where the model is nonlinear and the templates purposefully span a (too) large range. Therefore, the criteria for a valid application of the Linear Template Fit as discussed in Sect. 10 are mainly not fulfilled.

As in Examples 1 and 2 (Sects. 4, 6), the model is considered to be a normal distribution, and the same pseudo-data as in the previous examples are also used. In contrast to Examples 1 and 2, only the variance of the normal distribution shall be determined. Six templates are generated similarly to Examples 1 and 2, with variances between 3 and 8, and each template is generated using $4 \cdot 10^6$ events. The pseudo-data and the templates are displayed in Fig. 6. Recall that the pseudo-data were generated with a variance of 6.2 (due to statistical fluctuations, the true variance is about 6.3).

In this example, the Linear Template Fit results in a best estimator of 6.61 ± 0.32, and is thus not quite consistent with the expectation. The EDM is comparably large, with a value of -0.27, and $\sigma_{\hat{\alpha}}^{(m)} = -0.27$, and the minimum of the χ^2-parabola results in 6.52 ± 0.27. Consequently, Eqs. (61), (62), (71), and (93) are not well fulfilled. In Fig. 6

(right), the parameter dependence in some selected bins is displayed, and it is clearly seen that the templates are not described by a linear function. It is also observed from the χ^2 values of the individual templates that these cannot be described by a parabola as would be the case for a linear problem, as displayed in Fig. 7.

The Quadratic Template Fit, in contrast, results in a best estimator of 6.36 ± 0.40 and is thus in excellent agreement with the expectation. This is because the algorithm employs a second-degree polynomial in each bin to represent the model. This provides an adequate representation of the parameter dependence even of a nonlinear model, as also seen in Fig. 6 (right). A linearization at the value of the best estimator is used only for linear error propagation, and appears to be appropriate given the size of the final uncertainties, ± 0.40.

14 Validation study: determination of the strong coupling constant from inclusive jet cross section data

As an example application of the Linear Template Fit, the strong coupling constant in quantum chromodynamics (QCD), $\alpha_s(m_Z)$, is determined from a measurement of inclusive jet cross sections in proton–proton collisions at a center-of-mass energy of $\sqrt{s} = 7$ TeV [96]. The data were taken with the CMS detector at the LHC at CERN. Inclusive jets were measured as a function of the transverse momentum p_T in five regions of absolute rapidity $|y|$. Altogether 133 data points are available, and 24 different sources of uncertainties are associated with the data.

The model is obtained from a calculation in next-to-leading order perturbative QCD (NLO pQCD) using the program `nlojet++` [97,98] that was interfaced to the tool `fastNLO` [99–101]. The latter makes it possible to provide templates for any value of $\alpha_s(m_Z)$ and different parameterizations of parton distribution functions of the proton (PDFs) without rerunning the calculation of the pQCD matrix elements. Multiplicative non-perturbative (NP) [102] and electroweak corrections [103] are applied to the NLO predictions.

The value of $\alpha_s(m_Z)$ was already determined earlier from these data and in NLO accuracy in two independent analyses [104,105]. The two analyses differ in the assumption of the statistical model, the inference algorithm, and the uncertainties considered in the fit. This provides a comprehensive testing ground to compare the results from the Linear Template Fit.

In Ref. [104], the value of $\alpha_s(m_Z)$ is determined from the minimum of a χ^2 parabola, where the χ^2 is derived from normally distributed PDFs. In order to avoid a bias in that procedure [64,68,69], some uncertainty components are treated as multiplicative [104], which means that the covariance matrix is calculated from relative uncertainties that are multiplied

---

[9] The analytic calculation of the Hesse matrix results in an exact application of the Newton minimization, which is in contrast to many other iterative minimization algorithms, where the Hesse matrix is only approximated, iteratively improved, or numerically calculated. This results in a quick convergence with only a few steps, and no adaptations to the Newton steps are required.
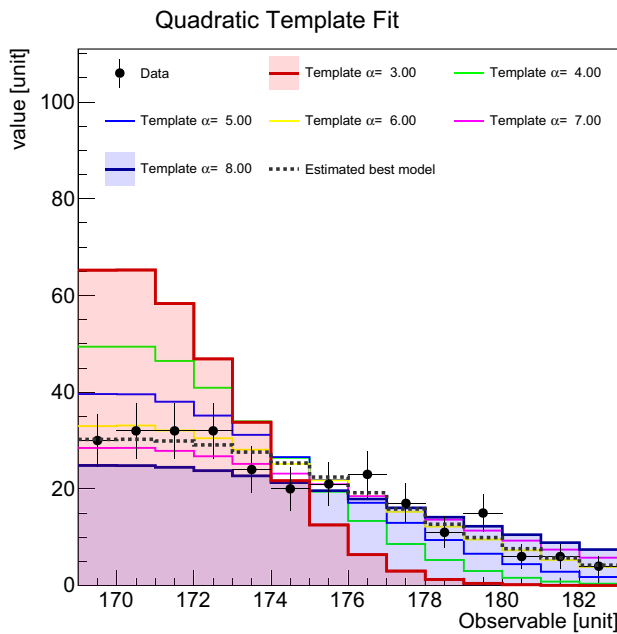
Fig. 6 Some illustrations of Example 3. Left: the template distributions Y as a function of the observable, and more details are given in the caption of Fig. 1. Right: the templates Y for some selected bins. The blue line indicates a second-degree polynomial regression of the
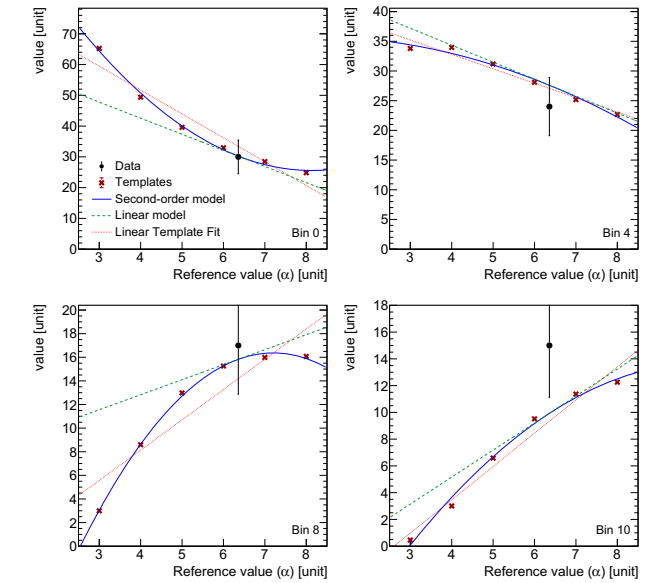
templates, as is used by the Quadratic Template Fit. The green dashed line indicates the linearization at the best estimator, and the red dotted line indicates the linear regression of the templates as it would be used by the plain Linear Template Fit

Fig. 7 Visualization of the χ^2 values of the templates in Example 3. A thin line connects these values. The red line displays a fitted parabola, which is not able to describe the χ^2 values adequately, which is indicative of a nonlinear problem. The star indicates the best estimator from the Quadratic Template Fit

with theory predictions. We chose predictions obtained with $\alpha_s(m_Z) = 0.116$ for that purpose, and the total covariance matrix in χ^2 is calculated as described in Ref. [104].

The templates are generated in the range $0.112 \leq \alpha_s(m_Z) \leq 0.121$ in steps of 0.001 using the MSTW PDF set [106]. In this methodology, the model predictions are not available for continuous values of $\alpha_s(m_Z)$, since the PDFs also exhibit an α_s dependence, but are provided only in a limited range and for discrete $\alpha_s(m_Z)$ values [106]. There-

fore, this kind of $\alpha_s(m_Z)$ inference is a typical use case for the Linear Template Fit. The templates and the linear model are displayed for some selected bins in Fig. 8.

The results from the Linear Template Fit (Eq. (32)), the Quadratic Template Fit (Sect. 12), and the parabolic fit (Sect. 10.3) are compared with the published results from CMS in Table 1. An illustration of the χ^2 values is displayed in Fig. 9 (left).

The best estimator for $\alpha_s(m_Z)$, its uncertainties, and the value of χ^2 are in good agreement among each other, and with the published CMS values [104], and differences are only in the rounding digit. It is also found that the total "fit" uncertainty is reproduced, which comprises the experimental and the PDF uncertainties, $\delta\alpha_s(m_Z) = \delta\alpha_s^{(\text{exp})} \oplus \delta\alpha_s^{(\text{PDF})} = \pm 0.0018$. The small differences in the individual uncertainty components between Ref. [104] and the Linear Template Fit are likely because the error breakdown in Ref. [104] is only approximate, whereas the Linear Template Fit provides an analytic error propagation at the value of the best estimator. In fact, all 24 uncertainty components of the data, the 20 symmetrized PDF uncertainties, and the NP uncertainties are propagated separately (cf. Sect. 8) and are displayed in Fig. 10. It is observed that the largest individual experimental uncertainty source is the luminosity uncertainty and the statistical uncertainty. The nuisance parameters are quite similar to those from (yet another) fit in Ref. [104]. The uncertainty component with the largest contribution to the total uncertainty stems from the PDFs and the luminosity uncertainty.
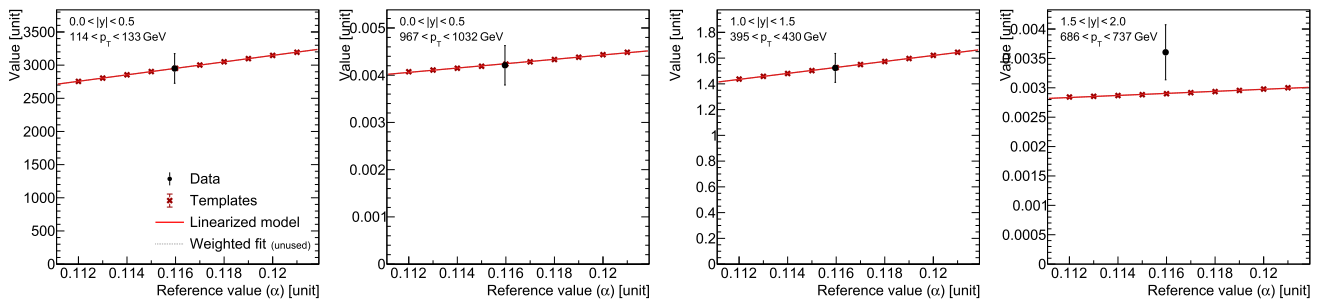
Fig. 8 Visualization of the templates, the data, and the implicit linearization of the model in some selected bins of the Linear Template Fit to the CMS inclusive jet cross sections. More details are given in the caption of Fig. 1

Table 1 Results from a Linear Template Fit of NLO pQCD predictions using the MSTW PDF set to CMS inclusive jet cross section data. The results are compared with the Quadratic Template Fit and to the analytic calculation of the minimum of the χ^2 parabola, and to the published results by the CMS collaboration in the last row. Shown are the best estimators for $\alpha_s(m_Z)$, the quadratic sum of all experimental uncertainties (exp), the propagated PDF uncertainties (PDF), the propagated NP uncertainties (NP), and the last column shows the χ^2/n_{dof} values. The difference between the linear and the Quadratic Template Fit is only 0.000017, but appears larger due to a rounding effect. Differences to the CMS result may also be due to numerical limitations of the input data, or how the PDF uncertainties are symmetrized. The total uncertainties that are considered in the fit sum to ±0.0018 in all cases

Fit method	Best estimator $\alpha_s(m_Z)$	χ^2/n_{dof}
Linear Template Fit	$0.1159 \pm 0.0014_{(exp)} \pm 0.0011_{(pdf)} \pm 0.0001_{(NP)}$	107.4/132
Quadratic Template Fit	$0.1160 \pm 0.0014_{(exp)} \pm 0.0011_{(pdf)} \pm 0.0001_{(NP)}$	107.2/132
χ^2 parabola	$0.1160 \pm 0.0018_{(exp,PDF)}$	107.2/132
CMS [104]	$0.1159 \pm 0.0012_{(exp)} \pm 0.0014_{(pdf)} \pm 0.0001_{(NP)}$	107.2/132
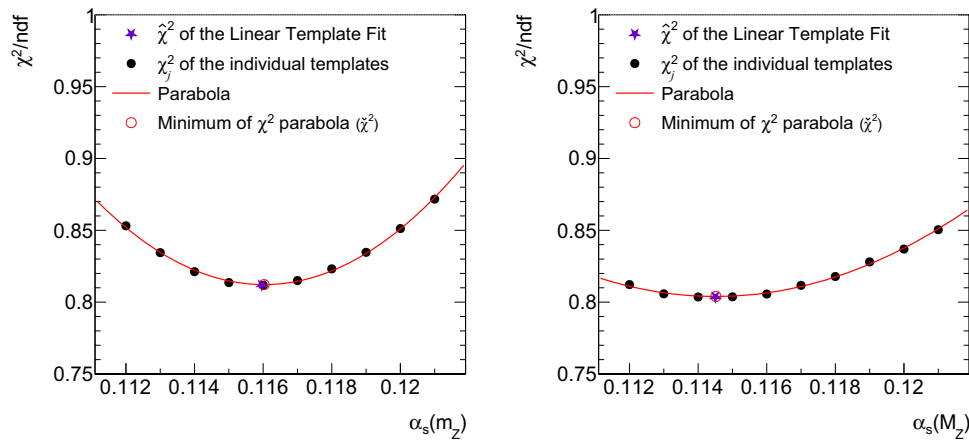
Fig. 9 Illustration of χ^2/n_{dof} values in the fit of NLO pQCD predictions to CMS inclusive jet cross section data. Shown are the χ^2/n_{dof} value of the Linear Template Fit at its best estimator (star), the χ^2/n_{dof} values of the individual templates (full circle), and the calculated χ^2 parabola (see Sect. 10.3) and its minimum (open circle). Left: χ^2/n_{dof} values of the Linear Template Fit with normally distributed uncertainties. Right: χ^2/n_{dof} values of the linear template with log-normally distributed uncertainties. Note that the left figure uses the PDF set MSTW for the NLO predictions with a varying $\alpha_s(m_Z)$ value, whereas the right one uses the PDF set NNPDF3.0

The statistical and uncorrelated uncertainties have the largest contribution to $\hat{\chi}^2$. The NP uncertainties are not included in the nominal χ^2 computation and are propagated separately. The EDM of the Linear Template Fit is $7 \cdot 10^{-5}$ and thus smaller than the rounding digit of the result.

In Ref. [105], as an intermediate result, the value of $\alpha_s(m_Z)$ was determined from CMS inclusive jet cross sections as well. In contrast to Ref. [104], the estimators are assumed to follow a log-normal distribution, and some theoretical uncertainties are further considered in the χ^2 expression, which was then minimized with Minuit [4]. The NNPDF3.0 PDF sets [107] are used for the NLO predictions and provide a covariance matrix with PDF uncertainties. This inference provides a useful testing ground for the Linear Template Fit with relative uncertainties (Sect. 7). Templates are generated in the range $0.112 \leq \alpha_s(m_Z) \leq 0.121$, and the same uncer-
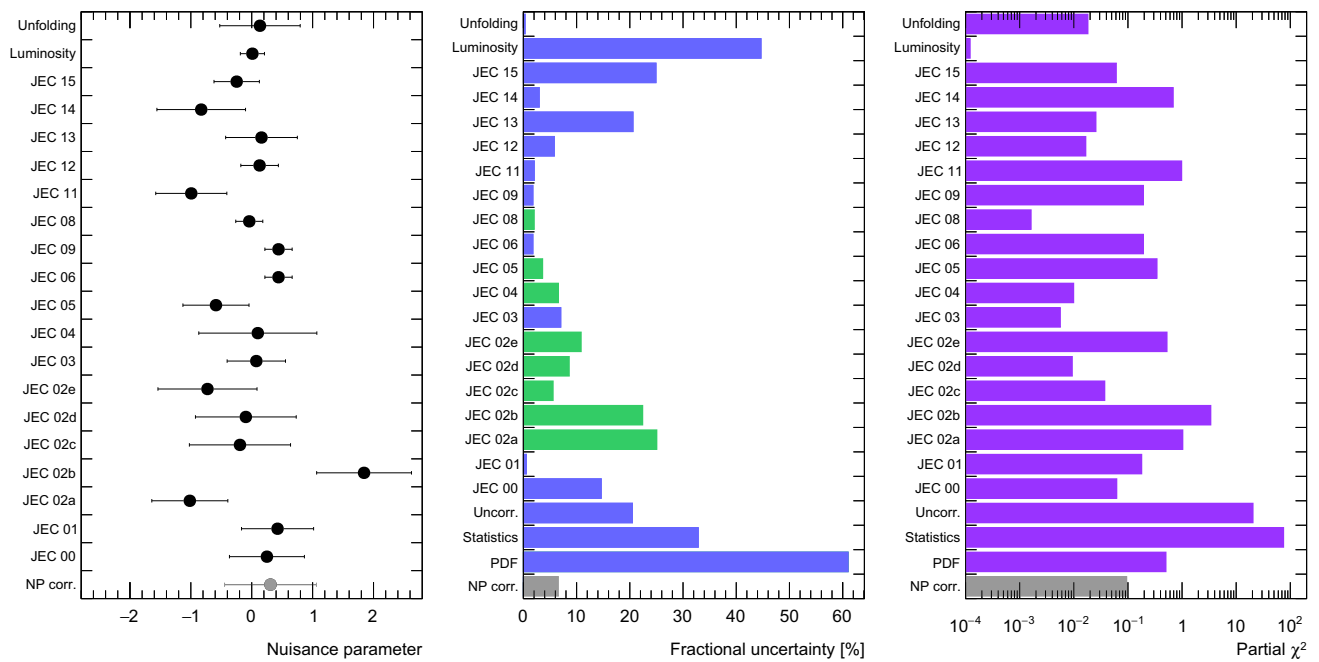
Fig. 10 Left: nuisance parameters of the systematic uncertainties in the $\alpha_s(m_Z)$ fit to CMS inclusive jet cross section data. Details on these uncertainties are found in Refs. [96,104]. The gray value(s) are not included in the χ^2 computation but are propagated separately. Middle: size of the individual uncertainty components to the total uncertainty of the best estimator, $\alpha_s(m_Z)$, in the Linear Template Fit to CMS inclusive jet cross section data. The color indicates the sign of the uncertainty, which, however, is of relevance only for multivariate fits. Right: partial χ^2 from the individual uncertainty components
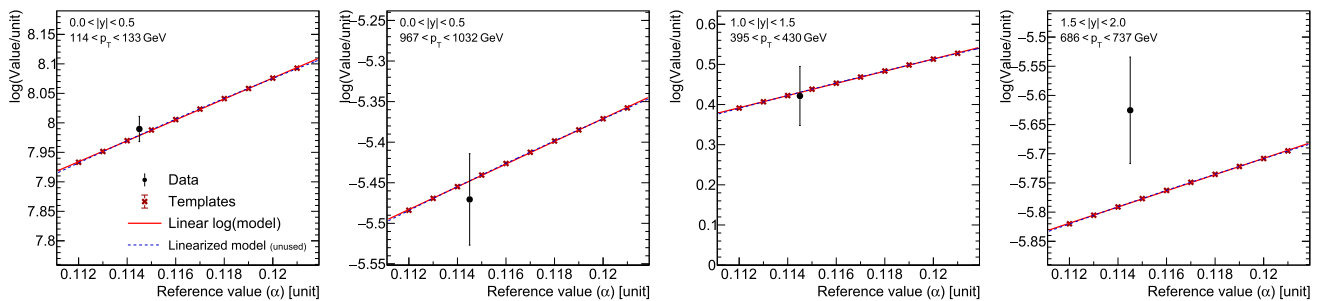
Fig. 11 Similar visualization of the templates than in Fig. 8, but using relative uncertainties (log-normally distributed PDFs) in the Linear Template Fit

tainty components as in Ref. [105] are considered. Some selected bins are displayed in Fig. 11. The results from the Linear Template Fit (Eq. (41)), the Quadratic Template Fit, and the parabolic fit are compared with the published results in Table 2, and an illustration of χ^2 values, which here are calculated from normally distributed relative uncertainties (Eq. (39)), is displayed in Fig. 9 (right).

The best estimator, the uncertainties, and the value of χ^2 are in good agreement among each other and with the results from Ref. [105]. The results differ only in the rounding digit, and different sizes of the uncertainty breakdown (into (exp), (PDF), and (NP) uncertainties) may be explained since the error breakdown in Ref. [105] is only approximate, but the total uncertainty is consistently $\delta\alpha_s = \pm0.0026$.

It is worth noting that due to the application of the factorization theorem [108], both the hard coefficients and the PDFs exhibit an α_s sensitivity. Consequently, in the two approaches discussed above, not only is the value of $\alpha_s(m_Z)$ determined, but the PDFs are determined simultaneously as well [109], which is realized through the inclusion of the PDF uncertainties in the Linear Template Fit, and which is commonly referred to as *PDF profiling*. This technique avoids a possible bias that is present when the PDFs are kept fixed [110], and it is particularly valid since the PDFs and the value of $\alpha_s(m_Z)$ are only weakly correlated in PDF determinations, and thus their correlations are negligible [111]. The different size of the uncertainties in Tables 1 and 2 is then a consequence of how the PDFs are considered in the fit. Since the

Table 2 Results from a Linear Template Fit with relative uncertainties where NLO pQCD predictions using the NNPDF3.0 PDF set are fitted to CMS inclusive jet cross section data. The results are compared with the Quadratic Template Fit and to the analytic calculation of the minimum of the χ^2 parabola, as well as to previously published results in the last row. Shown are the best estimators for $\alpha_s(m_Z)$, the quadratic sum of all experimental uncertainties (exp), the propagated PDF uncertainties (PDF), the propagated NP uncertainties (NP), and the last column shows the χ^2/n_{dof} values

Fit method	Best estimator $\alpha_s(m_Z)$	χ^2/n_{dof}
Linear Template Fit	$0.1144 \pm 0.0024_{\text{(exp)}} \pm 0.0011_{\text{(pdf)}} \pm\ <0.0001_{\text{(NP)}}$	106.1/132
Quadratic Template Fit	$0.1145 \pm 0.0024_{\text{(exp)}} \pm 0.0011_{\text{(pdf)}} \pm\ <0.0001_{\text{(NP)}}$	106.1/132
χ^2 parabola	$0.1145 \pm 0.0026_{\text{(exp,PDF)}}$	106.1/132
Ref. [105]	$0.1144 \pm 0.0022_{\text{(exp)}} \pm 0.0014_{\text{(pdf)}} \pm 0.0001_{\text{(NP)}}$	0.81

PDFs themselves are determined from data as well, the ansatz from CMS also exploits, to some extent, the α_s sensitivity of these further data by using the α_s dependent PDF, whereas the other ansatz from Ref. [105] exploits only the jet data and their sensitivity to $\alpha_s(m_Z)$ in the hard coefficients.

Consequently, these examples represent a comprehensive application of the Linear Template Fit, where in addition to determining the value of $\alpha_s(m_Z)$ a PDF profiling is performed, as well as a simultaneous constraint of the systematic uncertainties. Such constraints are obtained when the uncertainties in the nuisance parameters become smaller than unity. A PDF determination from these data alone, instead of a PDF profiling of an existent PDF, would be possible by considering the PDF uncertainties as *unconstrained* uncertainties in the Linear Template Fit, as described in Sect. 8.4. Such a fit exploits the PDF parameter space of the PDF set that is used to derive the PDF uncertainties. For the given example of CMS inclusive jet cross sections, however, this is not possible, since these data do not have sufficient constraints on the separate PDF flavors.

15 Summary and conclusions

In this article, the equations of the Linear Template Fit were presented. The Linear Template Fit provides an analytic expression for a maximum likelihood estimator in a uni- or multivariate parameter estimation problem. The underlying statistical model is constructed from normal or log-normal probability distribution functions and refers to a maximum likelihood or a minimum χ^2 method. For parameter estimation with the Linear Template Fit, the model needs to be provided at a few *reference* values in the parameter(s) of interest: the templates. The multivariate Linear Template Fit can be written as

$$\hat{\boldsymbol{\alpha}} = \left((Y\tilde{M})^{\mathrm{T}}WY\tilde{M}\right)^{-1}(Y\tilde{M})^{\mathrm{T}}W(\boldsymbol{d} - Y\bar{\boldsymbol{m}}),$$

where Y is the template matrix, $\boldsymbol{d}$ is the data vector (random vector), W is the inverse covariance matrix, and $\bar{\boldsymbol{m}}$ and $\tilde{M}$ are calculated from the reference points (Eq. (26)).

The equation of the Linear Template Fit is derived using two key arguments: (i) in the vicinity of the best estimator the model is linear, and (ii) the templates for different reference values are all generated in the same manner. The first precondition is commonly justified if the templates are provided within a reasonably small range around the (to be expected) best estimator. From the second, it follows that the bin-wise polynomial regression is unweighted, and thus the identical regression matrix is applicable in all bins. Several quantities to validate the applicability of the Linear Template Fit in a particular application were discussed.

If the linearity of the model is not justified, the Quadratic Template Fit is a suitable alternative algorithm for parameter estimation. It considers second-degree polynomials for the parameter dependence of the model and employs the quickly converging exact Newton minimization algorithm. For error propagation, the equations of the Linear Template Fit are applicable.

Both the linear and the Quadratic Template Fit implicitly transform the discrete representation of the model (the templates) into analytic continuous functions using polynomial regression. These expressions themselves may also become useful in several other applications where templates are available but continuous functions would be needed.

The equations for error propagation of different sources of uncertainties were presented. The analytic nature of the Linear Template Fit allows the straightforward propagation of each uncertainty component separately, which provides additional insights into the parameter estimation analysis.

As an example application, previously published results on the determination of the strong coupling constant $\alpha_s(m_Z)$ from inclusive jet cross section data taken at the LHC were repeated. The Linear and the Quadratic Template Fits reproduce these previously published results within the rounding digit.

In summary, key features of the Linear and the Quadratic Template Fit are its profound statistical model based on normal- or log-normally distributed PDFs, its simple formulae, stable results, low computational demands, full analytic error propagation, and its simple applicability. It is believed

that these template fits may become useful for statistical inference in a large variety of problems, such as performance-critical applications, and also in several fields outside of high-energy particle physics.

Data availability An implementation of the Linear Template Fit in C++ using the linear algebra package `Eigen` [112] can be found in https://github.com/britzger/LinearTemplateFit.

Appendix A: notation

See Table 3.

Table 3 Summary of the notation used. Capital letters denote a matrix, and small bold letters denote a column vector. The letters i, j, k, l, and t are indices. Throughout this article, a vector notation is used and, for instance, the vector $\boldsymbol{d}$ denotes a set of (random) variables, so $\boldsymbol{d} = \{d_i\}$. The same letters for indices and for the maximum value of an index are used. A subscript index denotes the entry of a vector (or matrix), and a bracketed index denotes one vector out of several others. Some letters have multiple meanings, if e.g. normal- or log-normally distributed random variables are considered, or if nonlinear effects are discussed, while the meaning always becomes clear from the context. Occasionally, a single entry of a vector will be denoted as a *bin*, like it would be a histogram entry. The dot notation, "$\dot{\alpha}$," denotes the *reference* values and looks similar to a "point" of a grid, but it should not be mistakenly understood as a derivative

Letter(s)	Description	Example(s) or definition
d_i, $\boldsymbol{d}$	Set of random variables $\{d_i\}$ (random vector; data distribution)	E.g., histogram of an observable
V	Covariance matrix including all uncertainty sources	$V = \mathbb{V} + \sum \boldsymbol{s}_{(l)}\boldsymbol{s}_{(l)}^{\mathrm{T}}$
$\mathbb{V}$	Covariance matrix (not including uncertainties s)	
$\boldsymbol{s}_{(l)}$, $\mathbf{s}_{(l)}$	Uncertainties with full bin-to-bin correlations (shifts)	
S	Matrix of uncertainties s, Eq. (34)	$S = \begin{pmatrix} \boldsymbol{s}_{(1)} & \dots & \boldsymbol{s}_{(l)} \end{pmatrix}$ or $\begin{pmatrix} \mathbf{s}_{(1)} & \dots & \mathbf{s}_{(l)} \end{pmatrix}$
W, $\mathbb{W}$	Inverse of V or $\mathbb{V}$	$W = V^{-1}$, $\mathbb{W} = \mathbb{V}^{-1}$
A	A detector response matrix (migration matrix) [optional]	
α, α_k, $\boldsymbol{\alpha}$	Model parameter(s) of interest	
$\boldsymbol{\lambda}(\boldsymbol{\alpha})$	The (physics) model with free parameter(s) $\boldsymbol{\alpha}$	
$\hat{\mathbf{y}}(\boldsymbol{\alpha})$	Best estimator of the linearized model	$\boldsymbol{\lambda}(\boldsymbol{\alpha}) \approx \hat{\mathbf{y}}(\boldsymbol{\alpha}) \equiv \mathbf{y}(\boldsymbol{\alpha}; \hat{\boldsymbol{\theta}})$
ϵ_l, $\boldsymbol{\epsilon}$	Nuisance parameter(s)	
$\hat{\alpha}$, $\hat{\alpha}_k$, $\hat{\boldsymbol{\alpha}}$	The best estimator(s) of the model parameter (an MLE)	
$\hat{\epsilon}_l$, $\hat{\boldsymbol{\epsilon}}$	The best estimator(s) of the nuisance parameter(s)	
$\hat{\boldsymbol{a}}$	A $k+l$ vector for the best estimator and nuisance parameters	$\hat{\boldsymbol{a}} = \begin{pmatrix} \hat{\boldsymbol{\alpha}} \\ \hat{\boldsymbol{\epsilon}} \end{pmatrix}$
$\dot{\alpha}_j$, $\dot{\alpha}_{j,k}$, $\dot{\boldsymbol{\alpha}}$	Reference values of the (jth) templates (for a multivariate Linear Template Fit, the vector $\dot{\boldsymbol{\alpha}}$ becomes a $j \times k$ matrix)	
$\boldsymbol{y}_{(j)}$, $\boldsymbol{x}_{(j)}$	A template distribution represented as an i-vector (or t vector),	E.g., histograms of an observable
	i.e. a model prediction for the reference value $\dot{\boldsymbol{\alpha}}$	$\boldsymbol{y}_{(j)} = \boldsymbol{\lambda}(\dot{\boldsymbol{\alpha}}_{(j)})$
	value(s) to match the data distribution in the fit	
Y, X	Template matrix, Eq. (14)	$Y = \begin{pmatrix} \boldsymbol{y}_{(1)} & \dots & \boldsymbol{y}_{(j)} \end{pmatrix}$
M	Matrix of reference values (i.e., given $\dot{\alpha}_j$) and a unit vector	Eqs. (10) or (25) (also Eq. (82))

Table 3 continued

Letter(s)	Description	Example(s) or definition
M^+	The g-inverse of the linear regression	$M^+ = (M^T M)^{-1} M^T$
$\bar{m}$	First column of M^{+T}, Eqs. (13) and (26)	
$\tilde{M}, \tilde{m}$	A submatrix of M^{+T} (i.e., all but $\bar{m}$), Eqs. (13) and (26)	$(\bar{m} \ \tilde{M}) = M^{+T}$
$F, \mathcal{F}$	Solution matrix of Linear Template Fit	
i	Index of an entry of the data vector; number of data points	
j	Index of the reference points; number of templates and reference points	
k	Index of the fit parameter for multivariate Linear Template Fit; number of fit parameters	
l	Index for uncertainties with full bin-to-bin correlations; number of systematic uncertainties	
t	Number of entries (bins) on truth level, if a response matrix A is used	
n	Degree of a polynomial function, or nth-order expansion	

Appendix B: pre-fitting the template distributions

It is tempting to fit the distributions of the templates with some well-motivated function and then use the best estimators of the function parameters as input parameters to a (linear) template fit for the determination of a model parameter. In this common practice, it is often believed that the fitted function may better exploit the sensitivity of the shape of the template distribution to the parameter of interest and thus enhance the quality of the results.

Let us consider to fit the template distribution with a polynomial of degree n. Unless there is no particular motivation for any particular other $n+1$ parameter function, e.g. from the underlying physics model, a polynomial is an equally valid function as any other $n+1$ parameter function for that purpose. In matrix notation, the polynomial is expressed using

$$P\theta = \begin{pmatrix} 1 & \dot{z}_1 & \dots & \dot{z}_1^n \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \dot{z}_i & \dots & \dot{z}_i^n \end{pmatrix} \theta, \tag{B.1}$$

where $\dot{z}_i$ may denote the observable value of the ith bin (bin center) of the template distribution. The best estimators for the polynomial parameters from a distribution $y_{(j)}$ are thus

$$\hat{\theta} = (P^T \mathcal{W}_{(j)} P)^{-1} \mathcal{W}_{(j)} P y_{(j)} := B_{(j)} y_{(j)}, \tag{B.2}$$

and $\mathcal{W}_{(j)} = \mathcal{V}_{(j)}^{-1}$ is the inverse covariance matrix of the uncertainties of the template values. Since in the Linear Template Fit the templates are sufficiently similar (cf. Sect. 10.1, and $\dot{\alpha}_{(j+1)} \sim \dot{\alpha}_{(j)} + \epsilon$), it is appropriate to assume that the uncertainties are (approximately) equivalent for all templates

$$\mathcal{V} := \mathcal{V}_{(1)} \sim \mathcal{V}_{(2)} \sim \dots \sim \mathcal{V}_{(j)} \sim c^2 \mathcal{V}_{\text{data}}, \tag{B.3}$$

and the uncertainties of the data scale by a constant factor c^2 only, probably because of a different amount of statistics.

Therefore, the g-inverse B of the nth-degree polynomial fit (Eq. (B.2)) is equivalent for any of the templates and also for the data distribution, $B_{(j)} = B$. Thus, the Linear Template Fit when using the $n+1$ polynomial function parameters becomes

$$\chi^2 = \left(Bd - BY\bar{m} - BY\tilde{m}\right)^T P^T W P \left(Bd - BY\bar{m} - BY\tilde{m}\right), \tag{B.4}$$

which after re-sorting becomes equivalent to the χ^2 equation of the Linear Template Fit

$$\chi^2 = \left(d - Y\bar{m} - Y\tilde{m}\right)^T \mathcal{W} \left(d - Y\bar{m} - Y\tilde{m}\right), \tag{B.5}$$

but using

$$\mathcal{W} = \left[P(P^T \mathcal{W} P)^{-1} \mathcal{W} P\right]^T W \left[P(P^T \mathcal{W} P)^{-1} \mathcal{W} P\right]. \tag{B.6}$$

It is therefore observed that a preceded fit of a function to the templates results only in a change of the Hesse matrix $W \to \mathcal{W}$. This observation is valid for all pre-fits that behave linearly, like B, and if Eq. (B.3) holds.

Appendix C: nonlinear approximation of a multivariate model

In Sect. 11, the nonlinear approximation of a univariate model using an nth-degree regression was described; see Eq. (81). When applying a first-order Taylor expansion to it, a linearized model is obtained, and from the equations of the Linear Template Fit, a best estimator for the fitting problem is found. A similar ansatz for a nonlinear model approximation of a multivariate model is straightforward. However,

in the nonlinear approximation of a multivariate model, it is also of interest to consider interference terms ($\sim \alpha_{k_1}\alpha_{k_2}$), and this will be discussed in the following. It will be shown that the subsequent linear approximation can again become input to a linear least-squares problem similar to the Linear Template Fit.

In each bin i, the parameter dependence of a multivariate model with k parameters $\boldsymbol{\alpha}$ can be approximated as

$$
\begin{aligned}
\lambda_i(\boldsymbol{\alpha}) &\approx y_i(\boldsymbol{\alpha}; \hat{\theta}_{(0,i)}, \hat{\boldsymbol{\theta}}_{(1,i)}, \hat{\boldsymbol{\theta}}_{(2,i)}, \hat{\boldsymbol{\theta}}_{(\text{Infrc},i)}) =: \hat{y}_i(\boldsymbol{\alpha}) \\
&= \hat{\theta}_{(0,i)} + \hat{\theta}_1^{(1,i)}\alpha_1 + \cdots + \hat{\theta}_k^{(1,i)}\alpha_k \\
&\quad + \hat{\theta}_1^{(2,i)}\alpha_1^2 + \cdots + \hat{\theta}_k^{(2,i)}\alpha_k^2 + \hat{\theta}_1^{(\text{Infrc},i)}\alpha_1\alpha_2 + \cdots
\end{aligned}
\tag{C.7}
$$

The best estimators for the θ parameters are obtained from a regression analysis using a Vandermonde-like design matrix (similar to Ref. [113])

$$
\begin{aligned}
M &= \begin{pmatrix} M_{(\text{linear fit})} & M_{(\text{squared})} & M_{(\text{interference})} \end{pmatrix} \\
&= \begin{pmatrix}
1 & \dot{\alpha}_{1,1} & \cdots & \dot{\alpha}_{1,k} & \dot{\alpha}_{1,1}^2 & \cdots & \dot{\alpha}_{1,k}^2 & \dot{\alpha}_{1,1}\dot{\alpha}_{1,2} & \cdots & \dot{\alpha}_{1,k_1}\dot{\alpha}_{1,k_2} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
1 & \dot{\alpha}_{j,1} & \cdots & \dot{\alpha}_{j,k} & \dot{\alpha}_{j,1}^2 & \cdots & \dot{\alpha}_{j,k}^2 & \dot{\alpha}_{j,1}\dot{\alpha}_{j,2} & \cdots & \dot{\alpha}_{j,k_1}\dot{\alpha}_{j,k_2}
\end{pmatrix},
\end{aligned}
\tag{C.8}
$$

where the entries are calculated from the reference points of the templates $\dot{\alpha}_{j,k}$. The first equation shows literally the three block matrices that form that design matrix. The best estimators of the θ parameters in each bin are then obtained from its g-inverse similar to Eq. (11)

$$
\begin{pmatrix}
\hat{\theta}_{(0,i)} \\
\hat{\boldsymbol{\theta}}_{(1,i)} \\
\hat{\boldsymbol{\theta}}_{(2,i)} \\
\hat{\boldsymbol{\theta}}_{(\text{infrc},i)}
\end{pmatrix} = M^+ \begin{pmatrix} y_{(1),i} \\ \vdots \\ y_{(j),i} \end{pmatrix}.
\tag{C.9}
$$

For reasons outlined in the main text, an unweighted regression is applicable, and thus the regression matrix is equivalent for every bin i. When substituting the $\hat{\theta}$ estimators into Eq. (C.7), one obtains the approximated model in every bin, and thus the full model as a second-order function $\hat{\mathbf{y}}(\boldsymbol{\alpha})$.

Next, a first-order Taylor expansion is applied to the model $\hat{\mathbf{y}}(\boldsymbol{\alpha})$. The expansion point is identified [10] with the best estimator of the Linear Template Fit $\hat{\boldsymbol{\alpha}}$. For that purpose it is useful to refer to the columns of the transposed g-inverse matrix with vectors ($\bar{\boldsymbol{m}}$ and multiple $\tilde{\boldsymbol{m}}$) like

---

[10] Of course, any other expansion point can be used in these equations, for instance the result from a previous measurement or the result from a previous nonLinear Template Fit. The latter defines naturally an iterative fitting algorithm.

$$
M^{+\text{T}}
$$
$$
=: \begin{pmatrix} \bar{\boldsymbol{m}} & \tilde{\boldsymbol{m}}_{[1]} & \cdots & \tilde{\boldsymbol{m}}_{[k]} & \tilde{\boldsymbol{m}}_{[1^2]} & \cdots & \tilde{\boldsymbol{m}}_{[k^2]} & \tilde{\boldsymbol{m}}_{[1\,2]} & \cdots & \tilde{\boldsymbol{m}}_{[k_1 k_2]} \end{pmatrix}.
\tag{C.10}
$$

The resulting terms from the first-order Taylor expansion are organized in the $k \times j$ matrix

$$
\tilde{\tilde{M}} =: \begin{pmatrix} \tilde{\tilde{\boldsymbol{m}}}_{[1]}^{\text{T}} \\ \vdots \\ \tilde{\tilde{\boldsymbol{m}}}_{[k]}^{\text{T}} \end{pmatrix} \quad \text{with elements} \quad \tilde{\tilde{\boldsymbol{m}}}_{[k]} := \tilde{\boldsymbol{m}}_{[k]}
$$
$$
+ 2\tilde{\boldsymbol{m}}_{[k^2]}\hat{\alpha}_k + \sum_{k_2 \neq k} \tilde{\boldsymbol{m}}_{[kk_2]}\hat{\alpha}_k\hat{\alpha}_{k_2}
\tag{C.11}
$$

and the vector

$$
\tilde{\tilde{\boldsymbol{m}}} = \bar{\boldsymbol{m}} - \sum_k \tilde{\boldsymbol{m}}_{[k^2]}\hat{\alpha}_k^2 - \sum_{k_1} \sum_{k_2 \neq k_1} \tilde{\boldsymbol{m}}_{[k_1 k_2]}\hat{\alpha}_{k_1}\hat{\alpha}_{k_2}.
\tag{C.12}
$$

Hence, the estimated model, similar to Eq. (86), is found as

$$
\tilde{\tilde{\mathbf{y}}}(\boldsymbol{\alpha}) = Y\tilde{\tilde{\boldsymbol{m}}} + Y\tilde{\tilde{M}}\boldsymbol{\alpha}.
\tag{C.13}
$$

This model, $\tilde{\tilde{\mathbf{y}}}(\boldsymbol{\alpha}) \approx \boldsymbol{\lambda}(\boldsymbol{\alpha})$, can be used together with the equations of the Linear Template Fit to calculate the best estimator, since it is linear in all parameters of interest.

Appendix D: calculation of the template fit matrix $\mathcal{F}$

The calculation of the template fit matrix $\mathcal{F}$ (Eq. (33)) involves the calculation of M^+ as well as the inversions of $\mathbb{V}$ and $\mathcal{D}$. These calculations shall be briefly discussed in the following.

The calculation of M^+ includes the calculation and subsequent inversion of the sum of squares and cross products (SSCP) matrix of an extended Vandermonde, $(M^{\text{T}}M)^{-1}$, which is widely studied in literature (see e.g. Ref. [54]) and typically does not involve numerical problems for $n = 1$ (or 2).

The inverse covariance matrix $\mathbb{W} = \mathbb{V}^{-1}$ is commonly calculable since $\mathbb{V}$ includes uncertainties without bin-to-bin correlations and it is thus positive definite. In fact, it needs to be invertible for a valid problem, or may be even a diagonal matrix. An efficient algorithm to prove that $\mathbb{V}$ is positive definite is a Cholesky decomposition, and when successful, the inverse is almost already calculated [86].

The inversion of the $(k+l) \times (k+l)$ matrix $\mathcal{D}$ may be more challenging, but it can be expressed using three sub-matrices as

$$
\mathcal{D} = \begin{pmatrix} (Y\tilde{M})^{\text{T}}\mathbb{W}Y\tilde{M} & (Y\tilde{M})^{\text{T}}\mathbb{W}S \\ S^{\text{T}}\mathbb{W}Y\tilde{M} & \mathbf{1}_l + S^{\text{T}}\mathbb{W}S \end{pmatrix} =: \begin{pmatrix} D & B \\ B^{\text{T}} & \Sigma \end{pmatrix},
\tag{D.14}
$$

where the right side introduces some shorthand notations (for D, compare also Eq. (33)). The inverse of $\mathcal{D}$ is then given in

two variants by [114]:

$$
\begin{aligned}
\mathcal{D}^{-1} &= \begin{pmatrix} D^{-1} - D^{-1}B\mathcal{L} & \mathcal{L}^{\mathrm{T}} \\ \mathcal{L} & L^{-1} \end{pmatrix} \\
&= \begin{pmatrix} K^{-1} & \mathcal{K} \\ \mathcal{K}^{\mathrm{T}} & \Sigma^{-1} - \Sigma^{-1}B^{\mathrm{T}}\mathcal{K} \end{pmatrix}
\end{aligned}
\tag{D.15}
$$

using

$$
\mathcal{L} = -L^{-1}B^{\mathrm{T}}D^{-1} \quad \text{and}
$$
$$
\mathcal{K} = -K^{-1}B\Sigma^{-1} . \tag{D.16}
$$

This requires one to calculate the inverse of only two matrices, L and D, or K and Σ, which are defined as

$$
D = (Y\tilde{M})^{\mathrm{T}}\mathbb{W}Y\tilde{M} \quad \text{and} \quad L := \Sigma - B^{\mathrm{T}}D^{-1}B , \tag{D.17}
$$

or

$$
\Sigma = \mathbf{1}_l + S^{\mathrm{T}}\mathbb{W}S \quad \text{and} \quad K := D - B\Sigma^{-1}B^{\mathrm{T}} \tag{D.18}
$$
$$
= (Y\tilde{M})^{\mathrm{T}}(\mathbb{W} - \mathbb{W}V_s\mathbb{W} + \mathbb{W}V_s V^{-1}V_s\mathbb{W})Y\tilde{M} . \tag{D.19}
$$

It is seen that these either can be inverted similarly to weighted least-squares matrices [53,115,116] or are just sums of two matrices, and the Sherman–Morrison–Woodbury formula can be applied [117,118] (see also Ref. [54]).

References

1. R. Fletcher, M. Powell, Comput. J. **6**, 163 (1963). https://doi.org/10.1093/comjnl/6.2.163
2. R. Fletcher, Comput. J. **13**(3), 317 (1970). https://doi.org/10.1093/comjnl/13.3.317
3. A. Goldstein, J.F. Price, Math. Comput. **25**, 569 (1971)
4. F. James, M. Roos, Comput. Phys. Commun. **10**, 343 (1975). https://doi.org/10.1016/0010-4655(75)90039-9
5. P.E. Gill, W. Murray, M.H. Wright, *Practical Optimization*, 14th edn. (Elsevier Academic Press, London, 2004)
6. D.P. Kingma, J. Ba, Adam: A method for stochastic optimization (2017)
7. S. Agarwal, K. Mierle, Others. Ceres solver. http://ceres-solver.org
8. Y. LeCun, Y. Bengio, G. Hinton, Nature **521**(7553), 436 (2015). https://doi.org/10.1038/nature14539
9. M. Feickert, B. Nachman, (2021)
10. O. Behnke, K. Kröninger, G. Schott, T. Schörner-Sadenius (eds.), *Data analysis in high energy physics: a practical guide to statistical methods* (Wiley-VCH, Weinheim, 2013). https://doi.org/10.1002/9783527653416
11. M. Baak, G.J. Besjes, D. Côte, A. Koutsman, J. Lorenz, D. Short, Eur. Phys. J. C **75**, 153 (2015). https://doi.org/10.1140/epjc/s10052-015-3327-7
12. G.J. Besjes, M. Baak, D. Côté, A. Koutsman, J.M. Lorenz, D. Short, J. Phys. Conf. Ser. **664**(7), 072004 (2015). https://doi.org/10.1088/1742-6596/664/7/072004
13. K. Cranmer, J. Brehmer, G. Louppe, Proc. Natl. Acad. Sci. **117**(48), 30055 (2020). https://doi.org/10.1073/pnas.1912789117
14. R. Boughezal, J.M. Campbell, R.K. Ellis, C. Focke, W. Giele, X. Liu, F. Petriello, C. Williams, Eur. Phys. J. C **77**(1), 7 (2017). https://doi.org/10.1140/epjc/s10052-016-4558-y
15. A. Gehrmann-De Ridder, T. Gehrmann, E.W.N. Glover, A. Huss, T.A. Morgan, JHEP **07**, 133 (2016). https://doi.org/10.1007/JHEP07(2016)133
16. J. Currie, E.W.N. Glover, J. Pires, Phys. Rev. Lett. **118**(7), 072002 (2017). https://doi.org/10.1103/PhysRevLett.118.072002
17. M. Czakon, P. Fiedler, D. Heymes, A. Mitov, JHEP **05**, 034 (2016). https://doi.org/10.1007/JHEP05(2016)034
18. P. Azzi et al., CERN Yellow Rep. Monogr **7**, 1 (2019). https://doi.org/10.23731/CYRM-2019-007.1
19. M. Cepeda et al., CERN Yellow Rep. Monogr. **7**, 221 (2019). https://doi.org/10.23731/CYRM-2019-007.221
20. M. Grazzini, S. Kallweit, J.M. Lindert, S. Pozzorini, M. Wiesemann, JHEP **02**, 087 (2020). https://doi.org/10.1007/JHEP02(2020)087
21. J. Mazzitelli, P.F. Monni, P. Nason, E. Re, M. Wiesemann, G. Zanderighi, Phys. Rev. Lett. **127**(6), 062001 (2021). https://doi.org/10.1103/PhysRevLett.127.062001
22. G. Heinrich, Phys. Rep. **922**, 1 (2021). https://doi.org/10.1016/j.physrep.2021.03.006
23. T. Gleisberg, S. Hoeche, F. Krauss, M. Schonherr, S. Schumann, F. Siegert, J. Winter, JHEP **02**, 007 (2009). https://doi.org/10.1088/1126-6708/2009/02/007
24. T. Sjöstrand, S. Ask, J.R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C.O. Rasmussen, P.Z. Skands, Comput. Phys. Commun. **191**, 159 (2015). https://doi.org/10.1016/j.cpc.2015.01.024
25. J. Bellm et al., Eur. Phys. J. C **76**(4), 196 (2016). https://doi.org/10.1140/epjc/s10052-016-4018-8
26. S. Agostinelli et al., Nucl. Instrum. Methods **A506**, 250 (2003). https://doi.org/10.1016/S0168-9002(03)01368-8
27. ATLAS Outreach. ATLAS Fact Sheet: To raise awareness of the ATLAS detector and collaboration on the LHC (2010). https://doi.org/10.17181/CERN.1LN2.J772. https://cds.cern.ch/record/1457044
28. ATLAS Computing: technical design report. Technical design report. ATLAS (CERN, Geneva, 2005). https://cds.cern.ch/record/837738
29. J. Gao, L. Harland-Lang, J. Rojo, Phys. Rep. **742**, 1 (2018). https://doi.org/10.1016/j.physrep.2018.03.002
30. A. Buckley, J. Ferrando, S. Lloyd, K. Nordström, B. Page, M. Rüfenacht, M. Schönherr, G. Watt, Eur. Phys. J. C **75**, 132 (2015). https://doi.org/10.1140/epjc/s10052-015-3318-8
31. A. Abulencia et al., Phys. Rev. D **73**, 032003 (2006). https://doi.org/10.1103/PhysRevD.73.032003
32. V.M. Abazov et al., Phys. Rev. Lett. **108**, 151804 (2012). https://doi.org/10.1103/PhysRevLett.108.151804
33. B. Malaescu, P. Starovoitov, Eur. Phys. J. C **72**, 2041 (2012). https://doi.org/10.1140/epjc/s10052-012-2041-y
34. T. Aaltonen et al., Phys. Rev. Lett. **108**, 151803 (2012). https://doi.org/10.1103/PhysRevLett.108.151803
35. G. Aad et al., Eur. Phys. J. C **72**, 2046 (2012). https://doi.org/10.1140/epjc/s10052-012-2046-6
36. G. Aad et al., Eur. Phys. J. C **75**(7), 330 (2015). https://doi.org/10.1140/epjc/s10052-015-3544-0
37. M. Aaboud et al., Phys. Lett. B **761**, 350 (2016). https://doi.org/10.1016/j.physletb.2016.08.042
38. M. Aaboud et al., Eur. Phys. J. C **78**(2), 110 (2018). https://doi.org/10.1140/epjc/s10052-017-5475-4 (**Erratum: Eur. Phys. J. C 78, 898 (2018)**)
39. A.M. Sirunyan et al., Eur. Phys. J. C **77**(7), 467 (2017). https://doi.org/10.1140/epjc/s10052-017-5030-3
40. M. Aaboud et al., Eur. Phys. J. C **79**(4), 290 (2019). https://doi.org/10.1140/epjc/s10052-019-6757-9
41. A.M. Sirunyan et al., Phys. Rev. Lett. **124**, 202001 (2020). https://doi.org/10.1103/PhysRevLett.124.202001

42. C. Herwig, T. Ježo, B. Nachman, Phys. Rev. Lett. **122**, 231803 (2019). https://doi.org/10.1103/PhysRevLett.122.231803

43. ATLAS Collaboration, ATLAS-CONF-2019-046 (2019), https://cds.cern.ch/record/2693954

44. K. Cranmer, A. Held, EPJ Web Conf. **251**, 03067 (2021). https://doi.org/10.1051/epjconf/202125103067

45. A.L. Read, Nucl. Instrum. Methods A **425**, 357 (1999). https://doi.org/10.1016/S0168-9002(98)01347-3

46. K. Cranmer, G. Lewis, L. Moneta, A. Shibata, W. Verkerke, (2012) https://cds.cern.ch/record/1456844

47. M. Baak, S. Gadatsch, R. Harrington, W. Verkerke, Nucl. Instrum. Methods A **771**, 39 (2015). https://doi.org/10.1016/j.nima.2014.10.033

48. G. Cowan, *Statistical Data Analysis* (Oxford University Press, Oxford, 1998)

49. F.E. James, *Statistical Methods in Experimental Physics*, 2nd edn. (World Scientific, Singapore, 2006)

50. J. Nocedal, S.J. Wright, *Numerical Optimization; Springer Series in Operations Research and Financial Engineering*, 2nd edn. (Springer, New York, 2006). https://doi.org/10.1007/978-0-387-40065-5

51. V. Blobel, in *Formulae and Methods in Experimental Data Evaluation*, vol. 3, ed. by R.K. Bock, et al., (1984), p. L1–L49

52. R.J. Barlow, *Statistics: A Guide to the Use of Statistical Methods in the Physical Sciences. Manchester Physics Series* (Wiley, Chichester, 1989)

53. G.A.F. Seber, A.J. Lee, *Linear Regression Analysis* (Wiley, Hoboken, 2003)

54. G.A.F. Seber, *A Matrix Handbook for Statisticians* (Wiley-Interscience, Hoboken, 2008)

55. V. Blobel, E. Lohrmann, *Statistische und numerische Methoden der Datenanalyse* (Teubner Studienbücher, Physik, Teubner, Stuttgart, 1998)

56. C.R. Rao, S.K. Mitra, *Generalized Inverse of a Matrix and Its Applications* (Wiley, New York, 1971)

57. E.H. Moore, in *The Fourteenth Western Meeting of the American Mathematical Society*, vol. 26 (American Mathematical Society, 1920), p. 394. bams/1183425340

58. R. Penrose, Math. Proc. Camb. Philos. Soc. **51**(3), 406 (1955). https://doi.org/10.1017/S0305004100030401

59. C.R. Rao, S.K. Mitra, in *Proceedings of 6th Berkeley Symposium Mathematics Statistical Probability University California. 1970, 1*, p. 601–620

60. G. Cowan, in *2020 Review of Particle Physics*, vol. Prog. Theor. Exp. Phys. 083C01, (2020). https://doi.org/10.1093/ptep/ptaa104

61. G. Cowan, K. Cranmer, E. Gross, O. Vitells, Eur. Phys. J. C **71**, 1554 (2011). https://doi.org/10.1140/epjc/s10052-011-1554-0 (**Erratum: Eur. Phys. J. C 73, 2501 (2013)**)

62. R. Brun, F. Rademakers, Nucl. Instrum. Methods A **389**, 81 (1997). https://doi.org/10.1016/S0168-9002(97)00048-X

63. J. Heinrich, L. Lyons, Ann. Rev. Nucl. Part. Sci. **57**, 145 (2007). https://doi.org/10.1146/annurev.nucl.57.090506.123052

64. L. Lyons, A.J. Martin, D.H. Saxon, Phys. Rev. D **41**, 982 (1990). https://doi.org/10.1103/PhysRevD.41.982

65. D. Lincoln, G. Morrow, P. Kasper, Nucl. Instrum. Methods A **345**, 449 (1994). https://doi.org/10.1016/0168-9002(94)90497-9

66. G. D'Agostini, Nucl. Instrum. Methods A **346**, 306 (1994). https://doi.org/10.1016/0168-9002(94)90719-6

67. T. Takeuchi, Prog. Theor. Phys. Suppl. **123**, 247 (1996). https://doi.org/10.1143/PTPS.123.247

68. G. D'Agostini, *Bayesian Reasoning in Data Analysis: A Critical Introduction* (World Scientific, Singapore, 2003). https://doi.org/10.1142/5262

69. R.D. Ball, L. Del Debbio, S. Forte, A. Guffanti, J.I. Latorre, J. Rojo, M. Ubiali, JHEP **05**, 075 (2010). https://doi.org/10.1007/JHEP05(2010)075

70. C. Forbes, M. Evans, N. Hastings, B.J. Peacock, *Statistical Distributions*, 4th edn. (Wiley, New York, 2010)

71. V. Andreev et al., Eur. Phys. J. C **75**(2), 65 (2015). https://doi.org/10.1140/epjc/s10052-014-3223-6

72. V.M. Abazov et al., Phys. Rev. D **75**, 092001 (2007). https://doi.org/10.1103/PhysRevD.75.092001

73. S. Chatrchyan et al., JHEP **12**, 105 (2012). https://doi.org/10.1007/JHEP12(2012)105

74. V. Blobel, in *Proceedings, CERN School of Computing: Aiguablava, Spain, September 9-22 1984* (1984)

75. G. Cowan, Conf. Proc. C **0203181**, 248 (2002)

76. V. Blobel, in *PHYSTAT 2011* (CERN, Geneva, 2011), p. 240–251. https://doi.org/10.5170/CERN-2011-006.240

77. F. Spano, in *Proceedings, IN2P3 School of Statistics (SOS 2012): Autrans, France, May 28-June 1, 2012*, vol. 55 (2013), p. 03002. https://doi.org/10.1051/epjconf/20135503002

78. S. Schmitt, in *Proceedings, 12th Conference on Quark Confinement and the Hadron Spectrum (Confinement XII): Thessaloniki, Greece*, vol. 137 (2017), p. 11008. https://doi.org/10.1051/epjconf/201713711008

79. L. Brenner, R. Balasubramanian, C. Burgard, W. Verkerke, G. Cowan, P. Verschuuren, V. Croft, Int. J. Mod. Phys. A **35**(24), 2050145 (2020). https://doi.org/10.1142/S0217751X20501456

80. V. Blobel. The RUN manual: regularized unfolding for high-energy physics experiments. OPAL Technical Note TN361 (1996)

81. G. D'Agostini, Nucl. Instrum. Methods A **362**, 487 (1995). https://doi.org/10.1016/0168-9002(95)00274-X

82. A. Hocker, V. Kartvelishvili, Nucl. Instrum. Methods A **372**, 469 (1996). https://doi.org/10.1016/0168-9002(95)01478-0

83. B. Malaescu, An Iterative, dynamically stabilized method of data unfolding (2009). arXiv:0907.3791

84. S. Schmitt, JINST **7**, T10003 (2012). https://doi.org/10.1088/1748-0221/7/10/T10003

85. A. Andreassen, P.T. Komiske, E.M. Metodiev, B. Nachman, J. Thaler, Phys. Rev. Lett. **124**(18), 182001 (2020). https://doi.org/10.1103/PhysRevLett.124.182001

86. K. Madsen, H.B. Nielsen, O. Tingleff, Methods for non-linear least squares problems (1999). http://www2.compute.dtu.dk/pubdb/pubs/660-full.html

87. R. Fletcher, *Practical Methods of Optimization*, 2nd edn. (Wiley, Hoboken, 2013)

88. A.C. Aitken, Proc. R. Soc. Edinb. **55**, 42–48 (1936). https://doi.org/10.1017/S0370164600014346

89. L. Lyons, D. Gibaut, P. Clifford, Nucl. Instrum. Methods A **270**, 110 (1988). https://doi.org/10.1016/0168-9002(88)90018-6

90. A. Valassi, Nucl. Instrum. Methods A **500**, 391 (2003). https://doi.org/10.1016/S0168-9002(03)00329-2

91. R. Nisius, Eur. Phys. J. C **74**(8), 3004 (2014). https://doi.org/10.1140/epjc/s10052-014-3004-2

92. M. Johnson, D. Maítre, Phys. Rev. D **97**(5), 054013 (2018). https://doi.org/10.1103/PhysRevD.97.054013

93. V. Blobel, in *Formulae and Methods in Experimental Data Evaluation*, vol. 3 ed. by R.K. Bock, et al. (1984), p. I1–I31

94. W. Gautschi, Numer. Math. **24**(1), 1–12 (1975). https://doi.org/10.1007/BF01437212

95. V.Y. Pan, SIAM J. Matrix Anal. Appl. **37**(2), 676 (2016). https://doi.org/10.1137/15M1030170

96. S. Chatrchyan et al., Phys. Rev. D **87**(11), 112002 (2013). https://doi.org/10.1103/PhysRevD.87.112002 (**Erratum: Phys. Rev. D 87, 119902 (2013)**)

97. Z. Nagy, Phys. Rev. Lett. **88**, 122003 (2002). https://doi.org/10.1103/PhysRevLett.88.122003

98. Z. Nagy, Phys. Rev. D **68**, 094002 (2003). https://doi.org/10.1103/PhysRevD.68.094002

99. T. Kluge, K. Rabbertz, M. Wobisch, in *14th International Workshop on Deep Inelastic Scattering* (2006). https://doi.org/10.1142/9789812706706_0110

100. D. Britzger, K. Rabbertz, F. Stober, M. Wobisch, in *20th International Workshop on Deep-Inelastic Scattering and Related Subjects* (2012). https://doi.org/10.3204/DESY-PROC-2012-02/165

101. D. Britzger, K. Rabbertz, G. Sieber, F. Stober, M. Wobisch, PoS DIS **2015**, 055 (2015). https://doi.org/10.22323/1.247.0055

102. P.Z. Skands, Phys. Rev. D **82**, 074018 (2010). https://doi.org/10.1103/PhysRevD.82.074018

103. S. Dittmaier, A. Huss, C. Speckner, JHEP **11**, 095 (2012). https://doi.org/10.1007/JHEP11(2012)095

104. V. Khachatryan et al., Eur. Phys. J. C **75**(6), 288 (2015). https://doi.org/10.1140/epjc/s10052-015-3499-1

105. D. Britzger, K. Rabbertz, D. Savoiu, G. Sieber, M. Wobisch, Eur. Phys. J. C **79**(1), 68 (2019). https://doi.org/10.1140/epjc/s10052-019-6551-8

106. A.D. Martin, W.J. Stirling, R.S. Thorne, G. Watt, Eur. Phys. J. C **63**, 189 (2009). https://doi.org/10.1140/epjc/s10052-009-1072-5

107. R.D. Ball et al., JHEP **1504**, 040 (2015). https://doi.org/10.1007/JHEP04(2015)040

108. J.C. Collins, D.E. Soper, G.F. Sterman, Adv. Ser. Direct. High Energy Phys. **5**, 1 (1989). https://doi.org/10.1142/9789814503266_0001

109. H. Paukkunen, P. Zurita, JHEP **12**, 100 (2014). https://doi.org/10.1007/JHEP12(2014)100

110. S. Forte, Z. Kassabov, Eur. Phys. J. C **80**(3), 182 (2020). https://doi.org/10.1140/epjc/s10052-020-7748-6

111. H.L. Lai, J. Huston, Z. Li, P. Nadolsky, J. Pumplin, D. Stump, C.P. Yuan, Phys. Rev. D **82**, 054021 (2010). https://doi.org/10.1103/PhysRevD.82.054021

112. G. Guennebaud, B. Jacob, et al. Eigen v3. http://eigen.tuxfamily.org (2010)

113. Scikit-learn developers. sklearn.preprocessing.polynomialfeatures. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.PolynomialFeatures.html

114. H.V. Henderson, S.R. Searle, SIAM Rev. **23**(1), 53 (1981). http://www.jstor.org/stable/2029838

115. A. Björck, *Numerical Methods for Least Squares Problems* (SIAM, Philadelphia, 1996). https://doi.org/10.1137/1.9781611971484

116. A. Björck, *Numerical Methods in Matrix Computations. Texts in Applied Mathematics* (Springer, Cham, 2015). https://doi.org/10.1007/978-3-319-05089-8

117. W.J. Duncan, Lond. Edinb. Dublin Philos. Mag. J. Sci. **35**(249), 660 (1944). https://doi.org/10.1080/14786444408520897

118. M.A. Woodbury, *Inverting Modified Matrices*, vol. 42 (Princeton, 1950)