

# ePump running

Keping Xie<sup>1,\*</sup>

<sup>1</sup>*Pittsburgh Particle Physics Astrophysics and Cosmology Center,  
Department of Physics and Astronomy, University of Pittsburgh, Pittsburgh, PA 15260, USA*

(Dated: August 17, 2022)

In this note, we run the ePump updating of PDF and compare with the xFitter profiling.

## I. THE METHODOLOGY OF THE PDF UPDATING

### A. The Hessian representations

In the hessian representations, we can construct the dependence of observable  $X$  on the PDF parameters  $a_i$  with the eigenvector sets in terms of

$$X(\{a_i\}) = X(\{z_i\}) = X_0 + \sum_i \left( \frac{\partial X}{\partial a_i} z_i + \frac{1}{2} \frac{\partial^2 X}{\partial a_i^2} z_i^2 \right), \quad (1)$$

where  $z_i = a_i - a_{i,0}$ , and the derivates can be constructed as

$$\frac{\partial X}{\partial a_i} = \frac{X_i^+ - X_i^-}{2h}, \quad \frac{\partial^2 X}{\partial a_i^2} = \frac{X_i^+ - 2X_0 + X_i^-}{h^2}, \quad (2)$$

with the step  $h = 1$ . The  $X$ 's dependence on the SMEFT operators  $C_i \mathcal{O}_i$  can be constructed in a similar way as

$$X(\{C_i\}) = X_0 + \sum_i \left( \frac{X(C_{i,0}) - X(-C_{i,0})}{2C_{i,0}} C_i + \frac{1}{2} \frac{X(C_{i,0}) - 2X_0 + X(-C_{i,0})}{C_{i,0}^2} C_i^2 \right) \quad (3)$$

The existing ePump takes the linear shift as default. We need to enable the quadratic extension, which is necessary for the SMEFT scenario.

### B. The $\chi^2$ updating

The ePump (error PDF Updating Method Package) is developed by Carl Schmidt *et al.*, for quickly examining the impact of the new data set (or sets) on PDFs [1, 2]. A similar treatment can be also found in the Hessian profiling [3] implemented in the xFitter package, inherited from HERAFitter [4].

The basic idea is found a new minimum based on the Hessian approximation of the  $\chi^2$  along PDF eigen vector,  $\hat{z}$ , illustrated in Fig. 1. The original  $\chi^2$  along the eigenvector is assumed to be increased by 100 units, with the form as

$$\chi_{\text{old}}^2 = \sum_i T_i^2 z_i^2. \quad (4)$$

With 3 points, we can determine the  $\chi^2$  for new data with the quadratic assumption as

$$\chi_{\text{new}}^2 = \sum_{\alpha} \frac{(X_{\alpha}(\{z_i\}) - X_{\alpha}^E)^2}{\sigma_{\alpha}^2}, \quad (5)$$

---

\* xiekeping@pitt.edu

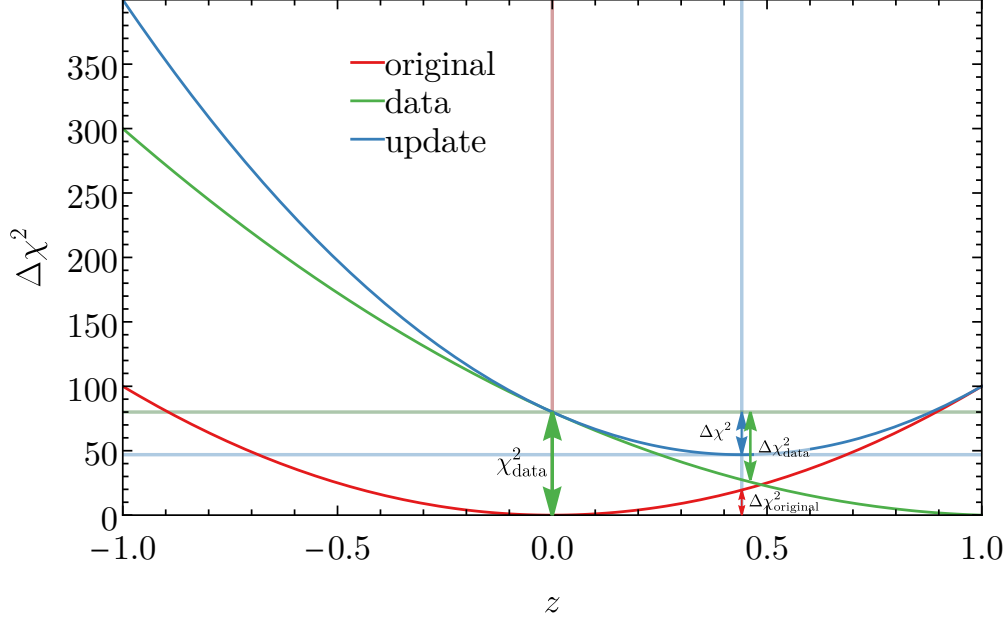


FIG. 1. An illustration of the  $\chi^2$  update in the Hessian profiling.

Tolerance	Fixed $T$	Dynamical $T_i$
ePump $\chi_{\text{old}}^2$		$T^2 \sum_i z_i^2$
Modification	$T^2 \sum_i z_i^2 \leq T^2$	$T^2 \sum_i \frac{z_i^2}{(T_i/T)^2} \leq T^2$
Eigen values $\pm z_i$	$\pm 1$	$\pm T_i/T$
$\frac{\partial X}{\partial z_i} = \Delta X_i$	$\frac{X_i^+ - X_i^-}{2}$	$\frac{T_i^-}{2T_i} \frac{X_i^+ - X_0}{T_i^+/T} + \frac{T_i^+}{2T_i} \frac{X_0 - X_i^-}{T_i^-/T}$
$T_i = T_i^+ = T_i^-$		$\frac{X_i^+ - X_i^-}{2T_i/T}$
$\frac{\partial^2 X}{\partial z_i^2} = \Delta^2 X_i$	$X_i^+ - 2X_0 + X_i^-$	$\frac{T}{T_i} \left( \frac{X_i^+ - X_0}{T_i^+/T} - \frac{X_0 - X_i^-}{T_i^-/T} \right)$
$T_i = T_i^+ = T_i^-$		$\frac{X_i^+ - 2X_0 + X_i^-}{(T_i/T)^2}$
$X(z) = X_0 + \sum_i (\Delta X_i z_i + \Delta^2 X_i z_i^2 / 2)$		
$X(\pm z_i)$	$X_i^\pm$	$X_i^\pm$
$X(\pm T_i^\pm / T)$		$X_i^\pm$

TABLE I. Comparison of fixed and dynamical tolerances. We define the average  $T_i = (T_i^+ + T_i^-)/2$ .

where  $X_\alpha(\{z_i\})$  is the theoretical prediction for  $\alpha$ -th data, while  $X_\alpha^E$  is the corresponding experimental data. Therefore, the new  $\chi^2$  can be obtained as

$$\chi_{\text{tot}}^2 = \chi_{\text{old}}^2 + \chi_{\text{new}}^2. \quad (6)$$

We find the new minimum at  $z = z_0$ . Therefore, we obtain the update  $\chi^2$  for the old and new data as

$$\Delta\chi^2 = \Delta\chi_{\text{old}}^2 + \Delta\chi_{\text{new}}^2, \quad (7)$$

where  $\Delta\chi_{\text{old}}^2 > 0$ ,  $\Delta\chi_{\text{new}}^2 < 0$  and  $\Delta\chi^2 < 0$ , shown in Fig. 1.

## II. THE PRACTICAL IMPLEMENTATIONS

### A. The D' Agostini's Bias and extended $t_0$ method

D' Agostini realize the dependence of the effect of offset (additive) error and normalization (multiplicative) error will plays a potential strong role in the  $\chi^2$  fit for new physics search [9]. In two different normalization treatments,

$$\begin{aligned}\chi_A^2 &= \sum_i \frac{(fx_i - k_i)^2}{(f\sigma_i)^2} + \frac{(f-1)^2}{\sigma_f^2}, \\ \chi_B^2 &= \sum_i \frac{(fx_i - k_i)^2}{\sigma_i^2} + \frac{(f-1)^2}{\sigma_f^2},\end{aligned}\tag{8}$$

$\chi_B^2$  reproduces the D' Agostini's bias. We can match it to the standard  $\chi^2$  definition in Eq. (11), by taking

$$\lambda = f - 1, \quad \sigma_f = 1, \quad k_i = T_i, \quad x_i = D_i.\tag{9}$$

Therefore, we have

$$\begin{aligned}\chi_A^2 &= \sum_i \frac{(D_i - T_i/(1+\lambda))^2}{\sigma_i^2} + \lambda^2 \approx \sum_i \frac{(D_i - T_i + \lambda T_i)^2}{\sigma_i^2} + \lambda^2, \\ \chi_B^2 &= \sum_i \frac{((1+\lambda)D_i - T_i)^2}{\sigma_i^2} + \lambda^2 = \sum_i \frac{(D_i - T_i + \lambda D_i)^2}{\sigma_i^2} + \lambda^2,\end{aligned}\tag{10}$$

where we have assumed  $\lambda \ll 1$ , such that  $1/(1+\lambda) \approx 1 - \lambda$ . We see that the nuisance parameter shifts are normalized to theory  $\lambda T_i$  or data  $\lambda D_i$ , which maps to the correlated uncertainties  $\beta_i$  in Eq. (13). In statistics, the  $\chi_A^2$  corresponds to Pearson  $\chi^2$ , while  $\chi_B^2$  is Neyman's definition.

NNPDF realized this issue and proposed the  $t_0$  method to multiply the normalization uncertainties  $\sigma^{(\mathcal{L})}$ , such as luminosity, with theories in the QCD global analysis [10]. It is extended to deal with the correlated systematic uncertainties  $\sigma^{(c)}$  as well, and dubbed as extended  $t_0$  method in the 2012 benchmark study [11]. CT10 NNLO analysis has performed a systematic investigation for different treatments [12].

However, it is still an open question which one will give a better prescription. Usually, we know that the central values of data point will suffer some random fluctuations, while theoretical predictions are much stable. Under this scenario, the data fluctuations will propagate into the nuisance parameter shifts  $\beta_i$ , and hence give a larger  $\chi^2$ . This is the argument for the extended  $t_0$  ( $T^{(0)}$  in the CTEQ notation) prescription to normalize relative uncertainty by theory, which gives the smoothest behavior [12].

### B. Four error types

The goodness-of-ft can be quantified as the the log-likelihood function

$$\chi^2 = \sum_i^{N_{\text{pt}}} \frac{1}{\Delta_i^2} \left( D_i - T_i - \sum_{\alpha}^{N_{\lambda}} \lambda_{\alpha} \beta_{i,\alpha} \right) + \sum_{\alpha}^{N_{\lambda}} \lambda_{\alpha}^2.\tag{11}$$

The  $\Delta_k^2$  is the quadrature sum of the statistical and the uncorrelated systematical uncertainty

$$\Delta_i^2 = \Delta_{i,\text{stat}}^2 + \Delta_{i,\text{unc.sys}}^2.\tag{12}$$

The  $\lambda_{\alpha}$  is the nuisance parameter with respect to (w.r.t.) the corresponding correlated uncertainty  $\beta_{i\alpha}$ . The correlated uncertainties are usually given as a relative percentage w.r.t.

the central values. We have the freedom to construct the absolute correlated uncertainties in terms either theoretical predictions or experimental data, as

$$\beta_{i,\alpha} = \begin{cases} T_i \cdot \text{sys}\% & (\text{ET}=1) \\ D_i \cdot \text{sys}\% & (\text{ET}=4) \end{cases}, \quad (13)$$

which corresponds to the error type **ET=1(4)** in ePump, respectively. We dub this  $\chi^2$  definition as the nuisance parameter representation. With the nuisance parameter, we can construct the covariance matrix uniquely, without any ambiguity, as

$$\text{cov}_{ij} = \delta_{ij}(\Delta_i^{\text{stat}}\Delta_j^{\text{stat}} + \Delta_i^{\text{unc.sys}}\Delta_j^{\text{unc.sys}}) + \sum_{\alpha}^{N_{\lambda}} \beta_{i\alpha}\beta_{j\alpha}, \quad (14)$$

with the corresponding  $\chi^2$  calculated as

$$\chi^2 = \sum_{i,j}^{N_{\text{pt}}} (D_i - T_i)(\text{cov}^{-1})_{ij}(D_j - T_j). \quad (15)$$

This representation matches to the error type **ET=3** in ePump. In many cases, experimentalists provide the correlation matrix, which is to construct the covariance matrix in terms of

$$\text{cov}_{ij} = \text{Corr}_{ij}\Delta_i^{\text{sys}}\Delta_j^{\text{sys}}. \quad (16)$$

In a more general setup, we can have both nuisance parameter and correlation matrix as inputs

$$\text{cov}_{ij} = \delta_{ij}\Delta_i^{\text{stat}}\Delta_j^{\text{stat}} + \text{Corr}_{ij}\Delta_i^{\text{sys}}\Delta_j^{\text{sys}} + \sum_{\alpha}^{N_{\lambda}} \beta_{i\alpha}\beta_{j\alpha}, \quad (17)$$

which corresponds to ePump's error type **ET=2**.

In summary, we have four options of error types in ePump, implemented in the latest version v20211102, summarized as:

- **Error\_type=1**, nuisance parameter, with relative correlated systematical uncertainty normalized to theory (NuiT);
- **Error\_type=2**, correlation matrix (Corr);
- **Error\_type=3**, covariance matrix (Cov);
- **Error\_type=4**, nuisance parameter, with relative correlated systematical uncertainty normalized to data (NuiD).

### C. The setup in ePump

We take ID=266, the CMS 7 TeV  $W \rightarrow \mu\nu$  asymmetry data [5], as an illustration. We need three files to run ePump,

- theory table 266CMS7Wmu.theory,
- data table 266CMS7Wmu.data,
- input file 266CMS7Wmu.in.

The input file is to specify setups.

+++ N(EV pairs)	N(Data Sets)	PDFtype(C/L/N)	Dyn_Tol?(Y/N)	Tol_squared	
29	1	C	y	100.0	
+++ ObservableFile	N(Observables)	Data?(Y/N)	Error_type	Weight	PS:
266CMS7Wmu/ID266_ET1	11	Y	1	1.0	

```
+++ PDFname          PDFout
PDFs/CT18NNLO.ev/i2Tn3  266CMS7Wmu/CT18NNLO_ID266_ET1/CT18NNLO_ID266_ET1
```

The theory table is universal as

```
Theory Column
5
Data : i2Tn3.00.dta
1.000e-01  8.039e+01  7.000e+03  1.12500e-01  1.12688e-01  2.594e-03
.....
Data : i2Tn3.01.dta
1.000e-01  8.039e+01  7.000e+03  1.12500e-01  1.12237e-01  2.594e-03
```

We only need to specify the location (column) of the theory predictions for each PDF error sets.

For ET=1(4), we need the CTEQ-like data table, as

```
# of corr_err  ,  DataColumn,  StatErr,  UncSys,  corr_err
10              4              5        7        9
```

The code will firstly read the number of correlated uncertainty, and the specific locations (column) for data, stat., unc. sys., and corr. sys., which are invoked for initialization. An dummy line followed by, as for comments. The code will read the data table accordingly. The code will construct the covariance matrix in terms of Eq. (14). The default  $\beta_{k,\alpha}$  is normalized with theory.

**[Question: Can we add an option to directly read  $\beta_{i,\alpha}$  rather the percentage? This should correspond to ET=4 case.]**

In ET=2 (Corr), we need a similar head file

```
# of corr_err  ,  Data Column,  StatErr,  UncSys,  corr_err
0              4              5        7        8
```

Besides, we need the correlation matrix, list as

```
#a      #b      CorrCoeff
1       1 1.0
2       1 0.046
.....
```

We can extract the correlation matrix from Tab. 3 of Ref. [5] or construct the correlation matrix according to the covariance matrix, provided by HEPData. Then, ePump construct the covariance matrix according to Eq. (17). In such a way, it can read both correlation matrix and nuisance parameter (such as luminosity). Note that the  $\Delta_i^{\text{SYS}}$  in Eq. (17). is read as the UncSys column.

In ET=3 (Cov), we need the covariance matrix table as

```

Experimental Data Values
0.1125
0.1138
.....
Inverse Covariance Matrix:
a=1,b=1,11
1.4998e+05 -3.6015e+03 .....
a=2,b=2,11
1.4113e+05 .....
.....

```

**[Question: Why do we manually construct the inverse of covariance matrix, rather than directly read them and automatically calculate the inverse within the code?]**

With ET=2, it reads data table with ReadInData2. We have both CTEQ data table into `s_sys[k][a]` (taking from `unc.sys.` position), `s_stat[k][a]`, and `beta[k][a][i]`, and correlation matrix into `rhocc[k][aa][bb]`. It constructs the covariance matrix with `ePump.C:ConstructCm2`, as

```

cov[a][b]=s_sys[k][a]*rhocc[k][a][b]*s_sys[k][b];
cov[a][b]+=beta[k][a][i]*beta[k][b][i] (sum i);
cov[a][b]+=s_stat[k][a]*s_stat[k][a] (a==b);

```

We have both correlation matrix and the fully correlated uncertainty (nuisance resource). This is designed to the case in which we both have the correlation matrix and the fully correlated uncertainties, such as the luminosity (and beam energy) in the LHCb  $WZ$  [6] and the CMS 8 TeV  $W \rightarrow \mu\nu$  asymmetry [7]. But in the CMS 7 TeV  $W$  asymmetry case [5], the correlated systematics are included in the correlation matrix already. We should set the  $\beta_i^{\text{lumi}}$  and other correlated uncertainty to be zero.

In default, both the correlation (ET=2) and inverse covariance matrices (ET=3) are read with upper triangle elements and then symmetrized.

$$\text{Corr}_{ji} = \text{Corr}_{ij}, \text{cov}_{ji}^{-1} = \text{cov}_{ij}^{-1}, (j \geq i). \quad (18)$$

**[KP: I have modified this part in order to read the elements with  $j < i$  as well. We should be more flexible in this part. The correlation (inverse covariance) matrix can have two formats, #a #b #CorrCoeff and 2D Npt\*Npt array. In the first format, the code will read the index a and b and fill the corresponding elements `rhocc[k][a][b]`. The index a and b should be an integer. We should enable the conversion from float.]**

### III. THE $\Sigma + K$ DECOMPOSITION

We take Alim's code `SigKDCom_v1` with the CMS 13 TeV  $t\bar{t}$  data as a demonstration, as shown in Tab. III.

CMS13tt: Covariance Matrix -> Nuisance Parameter

ATL13tt: Nuisance Parameter -> Covariance Matrix -> Nuisance Parameter

Conclusion for the  $\Sigma + K$  decomposition

Expt	ET1 (NuiT)	ET2 (Corr)	ET3 (Cov)
CMS7Wmu	7.8799	8.4653 (Mengshi)	8.0483 <sup>†</sup>
	CTEQ 7.880358	Corr. from cov. 8.0482	xFitter/Math 8.04824
	ET4(NuiD) 7.8854*	Extract from Ref. [5] 7.7179	

TABLE II. The ePump  $\chi^2$  for ID=266, CMS 7 TeV  $W \rightarrow \mu\nu$  asymmetry data ( $N_{\text{pt}} = 11$ ) [5]. <sup>†</sup>This small difference is resulted from the precision losing when Mengshi manually constructs the inverse of covariance matrix. \*The disagreement between ET4 and ET2/ET3 indicates the slight inconsistency between the nuisance parameters (read from paper) and covariance matrix (from HEPData) provided by experimentalist.

	ET1 (NuiT)	ET2 (Corr)	ET3 (Cov)	ET4 (NuiD)	Marco (ET1)	Marco (ET4)	Math
$m_{t\bar{t}}$	21.7941/21.2780	26.0323/24.8597	26.0322/24.8596	26.0322/24.8596	22.9299/22.3147	25.9680/24.8033	26.0322
$y_{t\bar{t}}$	6.3234/5.9249	8.4473/7.7947	8.4473/7.7947	8.4473/7.7947	6.4586/6.0415	8.4469/7.7944	8.44729
$y_t$	6.6442/6.4916	7.6433/7.4427	7.6433/7.4427	7.6433/7.4427	6.8897/6.7305	7.6563/7.4563	7.64331
$p_T^t$	18.6091/18.1005	22.6071/21.7422	22.6071/21.7422	22.6071/21.7422	19.3111/18.7462	22.6071/21.7422	22.6071

TABLE III. The ePump old (Constructed by Marco’s  $\Sigma + K$  decomposition) and new (from Alim’s ET1-4)  $\chi^2$  for CMS 13 TeV  $t\bar{t}$  data with different error types. The “Math” is the `Mathematica`  $\chi^2$  directly calculated with covariance matrix. Theoretical predictions are directly taken from NNLO version of fastNLO table [8].

- We need to normalize the correlated uncertainty (nuisance parameter) to experimental data, rather than theory, to match the  $\chi_{\text{cov}}^2$ .
- Pavel introduces an tolerance  $\epsilon$  to approximate the covariance matrix as  $\Sigma + K$ , such that the  $L_p$  norm  $\sum_{i,j}^{N_{\text{pt}}} |\text{cov}_{ij} - (\Sigma + K)_{ij}|^p < \epsilon$ . As a result, we get a smaller number of nuisance parameters, as the  $K$  matrix rank  $N_\lambda \sim \text{rank}(\widetilde{\text{cov}}) \leq N_{\text{pt}}$ . Alim’s algorithm enable that the nuisance parameter number up to data point,  $N_\lambda = N_{\text{pt}}$ , independent of the smallness of the eigenvalues.
- Treating the residuals as uncorrelated uncertainties can approximate the  $\chi_{\text{cov}}^2$  well when  $\epsilon$  is small enough, due to  $\Sigma + K \approx \text{cov}$ . But, treat residuals as correlated uncertainties exactly reproduce the  $\chi_{\text{cov}}^2$ , as a result of  $\sum_\alpha \beta_{i,\alpha} \beta_{j,\alpha} = \text{cov}_{ij}$ , independent of the choice of  $\epsilon$ .

#### IV. WARNING MESSAGE

In code `src/ReadIn.C`, we have the following functions to deal with 4 different error types.

```

case 1: ReadInData1(Datafile,i);
case 2: ReadInData2(Datafile,i);
case 3: ReadInData3(Datafile,i);
case 4: ReadInData1(Datafile,i);
    • Trim the starting space characters, as
        getline(infile, rawstring);

```

	ET1 (NuiT)	ET2 (Corr)	ET3 (Cov)	ET4 (NuiD)	Default	Math
$H_T^{t\bar{t}}$	19.4566/19.3695	19.5714/19.3867	19.5720/19.3873	19.5720/19.3873	19.5720/19.3873	19.572
$m_{t\bar{t}}$	13.1816/13.1534	13.9192/13.7816	13.9191/13.7816	13.9191/13.7816	13.9191/13.8496	13.9191
$y_{t\bar{t}}$	<b>12.8164/12.7967</b>	12.7636/12.7357	12.7636/12.7357	12.7636/12.7357	12.7636/12.7357	12.7636
$p_T^{t_1}$	<b>13.4658/13.4406</b>	10.4157/10.3651	10.4155/10.3648	10.4155/10.3648	10.4155/10.3648	10.4155
$p_T^{t_2}$						

TABLE IV. The ePump default (nuisance parameter in ePump and Mathematica) and new (nuisance to covariance matrix to decomposition)  $\chi^2$  for ATLAS 13 TeV  $t\bar{t}$  data with different error types. The theoretical predictions are taken from APPLgrid together with the NNLO/NLO  $K$ -factor generated with MATRIX.

```
inputstring=trim(rawstring);
```

- Do we want to enable new comment lines starting with # character?  
**Put aside**, as some part has adopted # of corr or #a #b to read information.
- Enable to read the lower half of inverse covariance matrix. **Not yet.**
- Enable to read covariance matrix and construct the inverse by add **ConstructCm3(k)**. We take a line in data table to specify **Inverse** to read the default, inverse covariance matrix. Otherwise, it directly reads covariance matrix and perform the conversion.

```
* Inverse Covariance Matrix:
Covariance Matrix:
```

- Similarly, we enable to read correlation matrix in  $N_{\text{pt}} \times N_{\text{pt}}$  form. The flag is **Matrix** in data table.

```
* #a #b CorrCoeff
#Correlation Matrix 9X9
```

- Print out  $\chi^2$  for all PDF sets.
- Specify the theory file, when many data share the same theoretical predictions. We take advantage of the last PS option in the data set line of input file.

+++ ObservableFile	N(Observables)	Data?(Y/N)	Error_type	Weight	PS:TheoryFile
test/ATL13mtt_ET2	9	Y	2	1	test/ATL13mtt

- 
- [1] C. Schmidt, J. Pumplin, C. P. Yuan, and P. Yuan, “Updating and optimizing error parton distribution function sets in the Hessian approach,” *Phys. Rev. D* **98** no. 9, (2018) 094005, [arXiv:1806.07950 \[hep-ph\]](#).
- [2] T.-J. Hou, Z. Yu, S. Dulat, C. Schmidt, and C. P. Yuan, “Updating and optimizing error parton distribution function sets in the Hessian approach. II.,” *Phys. Rev. D* **100** no. 11, (2019) 114024, [arXiv:1907.12177 \[hep-ph\]](#).
- [3] H. Paukkunen and P. Zurita, “PDF reweighting in the Hessian matrix approach,” *JHEP* **12** (2014) 100, [arXiv:1402.6623 \[hep-ph\]](#).



- [4] S. Alekhin *et al.*, “HERAFitter,” *Eur. Phys. J. C* **75** no. 7, (2015) 304, [arXiv:1410.4412 \[hep-ph\]](#).
- [5] **CMS** Collaboration, S. Chatrchyan *et al.*, “Measurement of the Muon Charge Asymmetry in Inclusive  $pp \rightarrow W + X$  Production at  $\sqrt{s} = 7$  TeV and an Improved Determination of Light Parton Distribution Functions,” *Phys. Rev. D* **90** no. 3, (2014) 032004, [arXiv:1312.6283 \[hep-ex\]](#).
- [6] **LHCb** Collaboration, R. Aaij *et al.*, “Measurement of forward W and Z boson production in  $pp$  collisions at  $\sqrt{s} = 8$  TeV,” *JHEP* **01** (2016) 155, [arXiv:1511.08039 \[hep-ex\]](#).
- [7] **CMS** Collaboration, V. Khachatryan *et al.*, “Measurement of the differential cross section and charge asymmetry for inclusive  $pp \rightarrow W^\pm + X$  production at  $\sqrt{s} = 8$  TeV,” *Eur. Phys. J. C* **76** no. 8, (2016) 469, [arXiv:1603.01803 \[hep-ex\]](#).
- [8] M. Czakon, D. Heymes, and A. Mitov, “fastNLO tables for NNLO top-quark pair differential distributions,” [arXiv:1704.08551 \[hep-ph\]](#).
- [9] G. D’Agostini, “On the use of the covariance matrix to fit correlated data,” *Nucl. Instrum. Meth. A* **346** (1994) 306–311.
- [10] **NNPDF** Collaboration, R. D. Ball, L. Del Debbio, S. Forte, A. Guffanti, J. I. Latorre, J. Rojo, and M. Ubiali, “Fitting Parton Distribution Data with Multiplicative Normalization Uncertainties,” *JHEP* **05** (2010) 075, [arXiv:0912.2276 \[hep-ph\]](#).
- [11] R. D. Ball *et al.*, “Parton Distribution Benchmarking with LHC Data,” *JHEP* **04** (2013) 125, [arXiv:1211.5142 \[hep-ph\]](#).
- [12] J. Gao, M. Guzzi, J. Huston, H.-L. Lai, Z. Li, P. Nadolsky, J. Pumplin, D. Stump, and C. P. Yuan, “CT10 next-to-next-to-leading order global analysis of QCD,” *Phys. Rev. D* **89** no. 3, (2014) 033009, [arXiv:1302.6246 \[hep-ph\]](#).