

# Notes on the algorithms used in artemide ver.3.

Alexey A. Vladimirov  
July 25, 2023

## I. CONVOLUTION AND GRIDS FOR DISTRIBUTIONS WITH TWIST-2 INPUT

### A. X grid

The value of  $x$  spans  $0 < x < 1$ , but never reaches  $x = 0$ . The grid is characterized by

- $x_{\min}$  the minimal value of  $x$
- $N_x$  number of segments. The number of nodes is  $N_x + 1$ .
- Function  $x_i$  which is  $x_0 = x_{\min}$  and  $x_{N_x} = 1$ .

These functions determine the following extra functions

- $\text{inv}[x]$  which is inverse of  $x_i$ . The integer part of  $\text{inv}[x]$  gives the number  $i$ , such that  $x_i < x < x_{i+1}$
- $\Delta_x$  typical increment

For any grid one has

$$\text{inv}[x_i] = i. \quad (1.1)$$

There are many functional forms for grid. Here are the most convenient

- **Linear grid**

$$x_i = x_{\min} + i\Delta_x, \quad \Delta_x = \frac{1 - x_{\min}}{N_x}, \quad \text{inv}[x] = \frac{x - x_{\min}}{\Delta_x}. \quad (1.2)$$

Not convenient since PDFs usually grows at small- $x$

- **Logarithm grid**

$$x_i = x_{\min} 10^{i\Delta_x}, \quad \Delta_x = \frac{-\log_{10} x_{\min}}{N_x}, \quad \text{inv}[x] = \frac{\log_{10}(x/x_{\min})}{\Delta_x}. \quad (1.3)$$

This is almost standard in the PDF community. However, at  $x \sim 1$  the points became sparse, and it rise the problem of precision loss. Usual solution is to use several grids for different ranges of  $x$ .

- **Arcosh grid**

$$x_i = \frac{1}{\cosh(\Delta_x(i - N_x))}, \quad \Delta_x = \frac{\text{arccosh}(1/x_{\min})}{N_x}, \quad \text{inv}[x] = N_x - \frac{\text{arccosh}(1/x)}{\Delta_x}. \quad (1.4)$$

This grid is alike logarithmic at small- $x$ , but has extra points at  $x \rightarrow 1$ . Therefore, one does not need extra gridding at large- $x$ .

### B. Lagrange interpolation

The Lagrange interpolation is defined by the order and the shift. Basically one defines

- $K = \{s, s + 1, \dots, s + r + 1\}$ , where  $s$  is a shift, and  $r$  is the order.

The standard choices are

- **Qubic backward grid:**  $s = -2, r = 3, K = \{-2, -1, 0, 1\}$ . In this case the function is determined in the last segment from three.
- **Qubic central grid:**  $s = -1, r = 3, K = \{-1, 0, 1, 2\}$ . In this case the function is determined in the middle segment from three.
- **Qubic forward grid:**  $s = 0, r = 3, K = \{0, 1, 2, 3\}$ . In this case the function is determined in the first segment from three.

The interpolation is done by

$$F(x) = \sum_{i=0}^{N_x-1} \theta(x_i < x < x_{i+1}) \sum_{k \in K} P_{i,k}(x) F(x_{i+k}), \quad (1.5)$$

where  $P$  is the Lagrange polynomial

$$P_{i,k}(x) = \prod_{\substack{l \in K \\ l \neq r}} \frac{\text{inv}[x] - \text{inv}[x_{i+l}]}{\text{inv}[x_{i+k}] - \text{inv}[x_{i+l}]}. \quad (1.6)$$

Note, that since  $\text{inv}[x_i] = i$  it simplifies

$$P_{i,k}(x) = \prod_{\substack{l \in K \\ l \neq r}} \left( \frac{\text{inv}[x]}{k-l} - \frac{i+l}{k-l} \right). \quad (1.7)$$

The interpolation sum can be rearranged as

$$F(x) = \sum_{i=0}^{N_x-1} W_i(x) F(x_i), \quad (1.8)$$

where

$$W_i(x) = \sum_{k \in K} \theta(x_{i-k} < x < x_{i-k+1}) P_{i-k,k}(x). \quad (1.9)$$

This function has the following properties

- $W_i(x) \neq 0$  for  $x_{i-K_{\max}} < x < x_{i-K_{\min}+1}$  which is the same as  $x_{i-s-r-1} < x < x_{i-s+1}$
- $W_i(x_j) = \delta_{ij}$
- The shape of the function  $W$  is independent on  $i$ . I.e. being evaluated in the terms of node-variables it is the same, but up to a shift.

One of the problems with gridding PDFs is the unknown value of PDF out side of the grid. The possible solutions are 1) reduce the order of interpolation at the borders 2) approximate points outside the grid.

*I have checked that* with the arcosh-grid [ $x_{\min} = 10^{-5}$ ,  $N_x = 200$ ] the error for forward and backward grids at  $x \rightarrow 1$  are very similar [test function behaved  $(1-x)^{2+\dots}$ ] if  $f(x > 1) = 0$ . Therefore, the forward grid (in this case) is a good choice.

### C. Convolution on the grid

The convolution has the general form

$$[C \otimes f](x, b, \mu) = \sum_f \int_x^1 \frac{dy}{y} C_{f,f'} \left( \frac{x}{y}, b, \mu \right) f_{f'}(y, \mu). \quad (1.10)$$

Here the  $\mu$  is internal parameter specified in the model. Furthermore, In the grid I save the function multiplied by  $x$  I get

$$\begin{aligned} x[C \otimes f](x, b, \mu) &= \sum_f \int_x^1 dy \frac{x}{y} C_{f,f'} \left( \frac{x}{y}, b, \mu \right) f_{f'}(y, \mu) \\ &= \sum_f \int_x^1 dy \frac{x}{y} C_{f,f'}(y, b, \mu) f_{f'} \left( \frac{x}{y}, \mu \right) \\ &= \sum_f \int_x^1 dy C_{f,f'}(y, b, ) F_{f'} \left( \frac{x}{y}, \mu \right), \end{aligned} \quad (1.11)$$

where  $F(x) = xf(x)$  is the function stored in LHAPDF grids.

For it I use the algorithm from QCDnum. The coefficient function is presented as

$$C_{f,f'}(x, b, \mu) = \sum_{n=0}^n \sum_{k=0}^n a_s^n \mathbf{L}^k C_{f,f'}^{(n,k)}(x). \quad (1.12)$$

Since the function is the function on the grid one can compute

$$[C \otimes f](x, b, \mu) = \sum_{f'} \sum_{n=0}^n \sum_{k=0}^n \sum_{i=0}^{N_x-1} a_s^n \mathbf{L}^k F_{f'}(x_i, \mu) \mathfrak{T}_{ff'}^{(n,k)}(x, x_i), \quad (1.13)$$

where

$$\mathfrak{T}_{ff'}^{(n,k)}(x, x_i) = \int_x^1 dy C_{ff'}^{(n,k)}(y) W_i \left( \frac{x}{y} \right). \quad (1.14)$$

For the  $x = x_j \in \text{grid}$  one has

$$[C \otimes f](x_i, b, \mu) = \sum_{f'} \sum_{n=0}^n \sum_{k=0}^n \sum_{j=0}^{N_x-1} a_s^n \mathbf{L}^k \mathfrak{T}_{ff';ij}^{(n,k)} f_{f'}(x_j, \mu), \quad (1.15)$$

where

$$\mathfrak{T}_{ff',ij}^{(n,k)} = \int_{x_i}^1 dy C_{ff'}^{(n,k)}(y) W_j \left( \frac{x_i}{y} \right). \quad (1.16)$$

Note that since the function  $W$  has restricted support one has

$$\mathfrak{T}_{ff',ij}^{(n,k)} = \int_{\max(x_i/x_j - \kappa_{\min+1}, x_i)}^{\min(1, x_i/x_j - \kappa_{\max})} dy C_{ff'}^{(n,k)}(y) W_j \left( \frac{x_i}{y} \right). \quad (1.17)$$

More over since  $x < 1$

$$\mathfrak{T}_{ff',ij}^{(n,k)} = \int_{x_i/x_j - \kappa_{\min+1}}^{\min(1, x_i/x_j - \kappa_{\max})} dy C_{ff'}^{(n,k)}(y) W_j \left( \frac{x_i}{y} \right). \quad (1.18)$$

The coefficient function has three parts

$$C(x) = C_\delta(x) + C_S(x) + C_R(x), \quad (1.19)$$

where

$$C_\delta(x) = c_\delta \delta(1-x), \quad C_S(x) = c_S [g(x)]_+, \quad (1.20)$$

and  $C_R$  is a regular part. Clearly the contribution of  $C_\delta$  to  $\mathfrak{T}$  is

$$c_\delta \delta_{ij} \rightarrow \mathfrak{T}_{ff',ij}^{(n,k)} \quad (1.21)$$

The regular part contributes

$$\int_{x_i/x_{j-K_{\min}+1}}^{\min(1, x_i/x_{j-K_{\max}})} dy C_R(y) W_j\left(\frac{x_i}{y}\right) \rightarrow \mathfrak{T}_{ff',ij}^{(n,k)} \quad (1.22)$$

Finally, the plus-part contributes

$$\begin{aligned} \int_{x_i/x_{j-K_{\min}+1}}^{\min(1, x_i/x_{j-K_{\max}})} dy C_S(y) W_j\left(\frac{x_i}{y}\right) &= \int_{x_i/x_{j-K_{\min}+1}}^{\min(1, x_i/x_{j-K_{\max}})} dyc_S[g(y)]_+ W_j\left(\frac{x_i}{y}\right) \\ &= \int_{x_i/x_{j-K_{\min}+1}}^{\min(1, x_i/x_{j-K_{\max}})} dyc_S g(y)_+ (W_j\left(\frac{x_i}{y}\right) - W_j(x_i)) - c_S \int_0^{j-K_{\min}+1} dy g(y) W_j(x_i) \\ &= \int_{x_i/x_{j-K_{\min}+1}}^{\min(1, x_i/x_{j-K_{\max}})} dyc_S g(y)_+ (W_j\left(\frac{x_i}{y}\right) - \delta_{ij}) - c_S \int_0^{j-K_{\min}+1} dy g(y) \delta_{ij} \rightarrow \mathfrak{T}_{ff',ij}^{(n,k)}, \end{aligned} \quad (1.23)$$

where it is used that  $W_j(x_i) = \delta_{ij}$ .

**In the case of forward cubic grid:**

$$\int_{x_i/x_{j-K_{\min}+1}}^{\min(1, x_i/x_{j-K_{\max}})} \rightarrow \int_{x_i/x_{j+1}}^{\min(1, x_i/x_{j-3})}$$

This method appears to have a fixed precision. I.e. its precision is equal proportional to the grid spacing. For reasonable grid the integral produces  $10^{-4}$  relative precision, which cannot be improved. Precision drops for smaller  $x$  (because of the range of integration). This is due to the fact that this method effectively incorporates the mistake of interpolation into the integration.

**Since I need only a finite number of convolutions. I turn back to the usual computation of the integral.**