PROCEEDINGS OF SCIENCE

Automatic differentiation for stochastic processes

Guilherme Catumba,[a,*] **Alberto Ramos**[a] **and Bryan Zaldívar**[a]

[a] *Instituto de Física Corpuscular (IFIC) CSIC - Universitat de Valencia.*
46071, Valencia, Spain

E-mail: gtelo@ific.uv.es, alberto.ramos@ific.uv.es, b.zaldivar.m@csic.es

Automatic differentiation methods allow to determine the Taylor expansion of any deterministic function. The generalization of these techniques for stochastic problems is not trivial. In this work we explore two approaches to extend automatic differentiation to stochastic processes, one based on reweighting (importance sampling) and another based on ideas from numerical stochastic perturbation theory using the hamiltonian formalism. A numerically implemented power series expansion is central for the extraction of the functional dependence on the parameter. The methods are tested and compared on a Bayesian inference model.

The 39th International Symposium on Lattice Field Theory,
8th-13th August, 2022,
Rheinische Friedrich-Wilhelms-Universität Bonn, Bonn, Germany

[*] Speaker

1. Introduction

The question of estimating the functional dependence of a given function on its input parameters is relevant for many areas, namely optimization problems and machine learning applications. In many of these cases the object of focus is an expectation value

$$\mathbb{E}_{p_\theta}\left[f(\phi;\theta)\right], \tag{1}$$

where $p_\theta(\phi)$ is a (possibly unnormalized) distribution function. The distribution, and hence the expectation value depends on the parameters θ. Our interest is to determine the gradient of eq. (1) w.r.t. the parameters θ. In this work we will consider cases where the expectation values are obtained through Markov chain Monte Carlo (MCMC) methods.

As examples of this general scheme, we can imagine eq. (1) as a lattice field theory expectation value, determined through a MCMC average. The distribution p_θ is related to the action, and the θ may include the bare couplings of the theory. The gradient of eq. (1) w.r.t. the couplings gives important information about the theory.

A similar case is the problem of Bayesian inference (BI). Bayesian predictions are also determined as expectation values of the form of eq. (1), with the posterior distribution depending, in general, on some *hyper-parameters*, θ. These represent the assumptions on the data and the model, and it is of interest to study the sensitivity of the predictions under changes of the parameters.

Although there are some known solutions to the tackle changes in distribution functions, these usually require the knowledge of the normalization factor, which rule out the use of MCMC methods. Hence, particularly for MCMC methods, the question of estimating derivatives of expectation values taken over possibly complicated distributions is unanswered.

For the general purpose of numerically estimating derivatives, the method of automatic differentiation (AD) [1] is widely used. It is based on the premise that any deterministic function amounts to the composition of basic operations and functions. In AD the differentiation of each basic operation is explicitly coded and the derivatives of the composition follows by the chain rule. The key factor is the deterministic character of these functions. Consequently, AD is not applicable when a stochastic element, *e.g.* Monte Carlo integration, is included in the construction of the function.

The idea of this work is to explore two approaches that extend AD to Monte Carlo processes. First we use reweighting methods (importance sampling), that allow to evaluate an expectation value w.r.t. a distribution using samples obtained from a different distribution function. Secondly, and specifically for processes involving stochastic differential equations (Langevin equation, Hamiltonian Monte Carlo), we consider a technique inspired in numerical stochastic perturbation theory (NSPT) [4, 5].

In order to extend AD to stochastic processes we also borrow some of its ideas. In this work employ power series expansions around the parameter of interest. The introduction of series expansion is, in a way, a generalization of the forward accumulation approach for AD that uses dual-numbers [3]. When combined with the series expansions, the two methods introduced above allow us to obtain the Taylor expansion of functions including stochastic elements, specifically, Monte Carlo methods.

After introducing the BI model, as well as the Hybrid Monte Carlo algorithm in section 2, the numerical implementation of the power series is introduced in section 3. In sections 4 and 5 the reweighting and NSPT methods are introduced, with the respective results shown in section 6. Finally, conclusions are drawn in section 7.

2. Toy model from Bayesian inference

In the BI paradigm a prediction starts with a dataset, D, assumed to be sampled from the *likelihood* (a distribution function built from the product of the assumed sampling densities), $p(D|\phi)$, dependent on some variables ϕ. In general, we are interested in modeling the dataset in order to extract predictions. Following the introduction of a *prior* (a distribution function over the variables ϕ modeling our assumptions about the data), $p_\theta(\phi)$, the *posterior* is obtained using Bayes theorem up to a ϕ, and θ-independent normalization

$$p_\theta(\phi|D) \propto p(D|\phi) \times p_\theta(\phi). \tag{2}$$

This represents the probability distribution of the parameters ϕ given the observed data D. The dependence on the hyper-parameters θ is written explicitly.

A prediction of the model is obtained through eq. (1) with p being replaced by the posterior $p_\theta(\phi|D)$. In addition to the normalization being, in general, unknown, the usual complexity of the posterior distribution makes the possibly highly dimensional integral difficult to compute – MCMC is a common solution.

For the proof of concept a linear model was considered. The dataset $D = \{x_i, y_i, \sigma_i\}_{i=1}^{20}$, shown in fig. 1, was taken from a third order polynomial, $y(x) = 1 + x + x^2 + x^3$, with Gaussian noise for the mean and the variance. The assumed probability distributions are normal, $N(y_i|f(x_i, \vec{\phi}), \sigma_i)$, and the likelihood becomes

$$p(\vec{y}|x, \vec{\phi}) = \prod_{i=1}^{N} N(y_i|f(x_i, \vec{\phi}), \sigma_i), \tag{3}$$

where the mean value is assumed to be a third order polynomial $f(x, \vec{\phi}) = \phi_0 + \phi_1 x + \phi_2 x^2 + \phi_3 x^3$.

The prior is taken as an uncorrelated normal distribution $p_\theta(\vec{\phi}) = N(\vec{\phi}|\vec{\mu}, \Sigma),\ \ \Sigma_p = \mathbf{1}\sigma_p^2$ with the parameter σ_p representing the 'confidence' in the model. The hyper-parameters of the model are $\theta = \{\vec{\mu}, \sigma_p\}$.

A given prediction for a quantity $g(\vec{\phi}; s)$ is obtained by an average over the probability distribution $p_\theta(\vec{\phi}|D)$

$$\langle g(s)\rangle = \int \prod_i d\phi_i g(\vec{\phi}; s) p_\theta(\vec{\phi}|D). \tag{4}$$

MCMC follows by generating samples $\{\vec{\phi}_i\}$ from the probability distribution $p_\theta(\vec{\phi}|D)$ and the integration is replaced by a finite average. We used Hybrid Monte Carlo (HMC) [6] where the samples are obtained by solving the equations of motion from a fictitious Hamiltonian $H(\vec{\phi}, \vec{\pi}) = \frac{\vec{\pi}^2}{2} - \log\Big(p(\vec{\phi}|D)\Big)$. $\vec{\pi}$ is the fictitious momentum conjugated to $\vec{\phi}$ which is randomly updated at each integration cycle. A final accept/reject Metropolis step is added to account for energy violations,

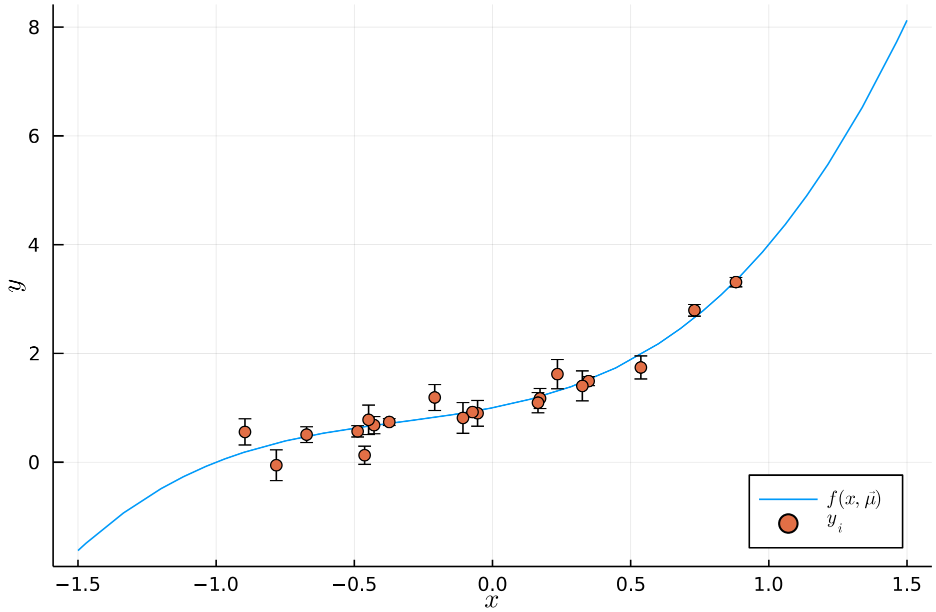

Figure 1: Randomly generated dataset $\{x, y, \sigma\}$ from a third order polynomial $y(x) = 1 + x + x^2 + x^3$ with Gaussian noise.

$\Delta H \neq 0$, due to errors in the numerical integration. For the case at hand, the precise form of the equations of motion is

$$\dot{\phi}_j = \pi_j, \tag{5}$$

$$\dot{\pi}_j = -\frac{1}{\sigma_p^2}(\phi_j - \mu_j) + \sum_{i=0}^{N} \frac{1}{\sigma_i^2} \left(y_i - f(x_i, \vec{\phi}) \right) (x_i)^j. \tag{6}$$

By iterating the above procedure a Markov chain is obtained, which is then used to generate a prediction. This is done for a given set of input hyper-parameters, $\theta = \{\vec{\mu}, \sigma_p\}$, but we are interested in estimating the functional dependence of the results on said parameters. This means that we want to know the derivatives of the predictions with respect to θ, *i.e.*, its Taylor expansion. For simplicity, we chose $\mu_i = 1,\ i = 0, 1, 2, 3$ (the same value as the one used to generate the dataset) and thus we focus solely on the σ_p dependence. In the following we take $\theta = \sigma_p$.

3. Numerical power series expansions

Being interested in numerically evaluating an expansion around a definite value σ_p^* for a general function $f(\phi)$, we start by defining the power series up to a given order δ^{N+1},

$$\tilde{\phi} = \phi_0 + \phi_1\delta + \phi_2\delta^2 + \ldots + \phi_N\delta^N, \tag{7}$$

where $\delta = (\sigma_p - \sigma_p^*)$. Subsequent operations (addition/subtraction/multiplication/division) of series elements are defined by neglecting all contributions $O(\delta^{N+1})$.

By the Taylor theorem, a function evaluated at $\tilde{\phi} = \phi_0 + \delta$ gives the Taylor expansion around σ_p^*

$$f(\tilde{\phi}) = f(\phi_0) + f'(\phi_0)\delta + \frac{1}{2}f''(\phi_0)\delta^2 + \ldots \tag{8}$$

If the variables of the functions are replaced by power series, and computations done order by order, the Taylor series of a general function is easily obtained.

For a numerical implementation this is done automatically by overloading all basic operations, $(+, *, \ldots)$, and basic functions $(\sin, \log, \ldots)$. In this way all operations can be carried in a natural way. Notice that the expansion parameter, σ_p^*, may appear explicitly in the equations and thus it also requires to be written as an expansion. Its expansion coefficients are $(\sigma_p^*, 1, 0, 0, \ldots)$, and further powers of the parameter follow by the product rule.

4. Reweighting method

Having samples $\{\phi_i\}$ from a distribution $p(\phi)$, used for MC integration, we can compute expectation values w.r.t. a different distribution, $q(\phi)$, using a weighted average

$$\langle f(s)\rangle_q = \frac{\left\langle f(s)\frac{q(\phi)}{p(\phi)}\right\rangle_p}{\left\langle \frac{q(\phi)}{p(\phi)}\right\rangle_p}, \tag{9}$$

where the subscript p, q indicates the distribution used for the average. If the distributions are characterized by hyper-parameters, θ, a change in its values defines a different distribution $p(\phi) = p_\theta(\phi),\ q(\phi) = p_{\theta'}(\phi)$.

When performed with normal variables the weighted average gives the total effect of the variation of the parameter. This is, in fact, one of the currently used methods to estimate electromagnetic (QED) corrections to hadronic (QCD) observables [7]. Since the simulation of the full theory (non-zero QED coupling and non-degenerate quark masses) presents additional complications, the MCMC samples are obtained from a pure QCD action with degenerate quark masses, $m_u = m_d$, and decoupled from the electromagnetic theory, $e = 0$. Reweighting provides expectation values for $m_u \neq m_d$ and $e \neq 0$. Recently, this type of importance sampling was also used as an improvement for variational inference in Bayesian methods [2]

If, however, the method is combined with a series expansion in the parameter, it can be used to obtain the derivatives of the average with respect to the parameter. This is done by replacing the weight term (quotient of the distribution functions) by a power series through the introduction the variables in eq. (7). Consequently, the reweighting average is correctly propagated at each order, and we obtain the Taylor expansion of the averages – the derivatives with respect to σ_p.

While the reweighting average is exact, its practical success is, in general, dependent on how similar the two distributions are. In an explicit diagrammatic approach such as [7] this involves delicate cancellations from the disconnected terms between the numerator and the denominator which increases the noise in the computation.

5. Numerical stochastic perturbation theory based approach

Another method to estimate the functional dependence on the parameters is based on NSPT. In stochastic quantization [4], a stochastic differential equation in a fictitious time, t, is introduced such that its stationary solutions reproduce a given quantum field theory. In the context of lattice theory NSPT identifies Euclidean quantum field theory as the equilibrium of a statistical system coupled to a heat reservoir.

Within this scheme observables are computed through an average over the noise configurations entering the stochastic equations and the physical results are obtained for large times. In practice, assuming ergodicity, the stationary average is replaced by a time average over a single noise configuration. For our application the key connection is that stochastic differential equations are a typical form of sampling from a distribution function (*e.g.* HMC introduced in section 2). In this sense, the MCMC average is nothing but a stochastic time average, where the Markov chain time is identified with the integration time of the equations of motion.

To extend AD for this cases, we start by replacing the variables entering the equations of motion by expansions in δ. In the toy model, the parameters ϕ become expansions

$$\phi_i(\sigma_p^* + \delta) = \sum_n \phi_i^{(n)} \delta^n = \sum_n \frac{1}{n!} \left. \frac{\partial^n \phi_i}{\partial \sigma_p^n} \right|_{\sigma_p = \vec{\sigma}_p^*} \delta^n. \tag{10}$$

The stochastic evolution is combined with the series expansion by plugging the expansion variables into the stochastic differential equation. By solving this tower of coupled equations of motion at each order, the MCMC samples also become expansions in δ. Following this, all the ensuing analysis, if carried properly at each order, carries the Taylor expansion of the whole computation.

While the overloading implies minimal changes to the usual HMC algorithm, some details are important. For the case of HMC, the random momentum update assumes the role of the stochastic noise and although π_i also becomes an expansion, its refreshment is done only at zeroth order, due to the way it enters the Hamiltonian. The stochastic character of the equations is propagated to higher orders by the mixed order terms.

Additionally, it is not possible to carry out the usual Metropolis step to correct for the numerical integration errors. While conservation of energy should still be ensured by the equations of motion, this correspondence cannot be made at each order and thus the results should be extrapolated to vanishing step-size of the integrator.

6. Results

For the model introduced in section 2 we performed the expansions around σ_p^*. Due to the linearity of the model and the numerical choice for the prior mean, $\vec{\mu}$, only the variance of observables are affected by a change in variance of the prior σ_p. In the following we analyze the variance of the parameters of the polynomial model $\delta\phi_j = \left\langle \phi_j^2 \right\rangle - \left\langle \phi_j \right\rangle^2$, as well as the variance for a prediction of a new point $\langle \delta y(x_n) \rangle$.

Reweighting requires a conventional Markov chain obtained at σ_p^* from which we obtain the Taylor expansion through the procedure introduced in section 4. In the case of NSPT the simulation provides the MC samples as expansion variables. With these components the Taylor expansion is

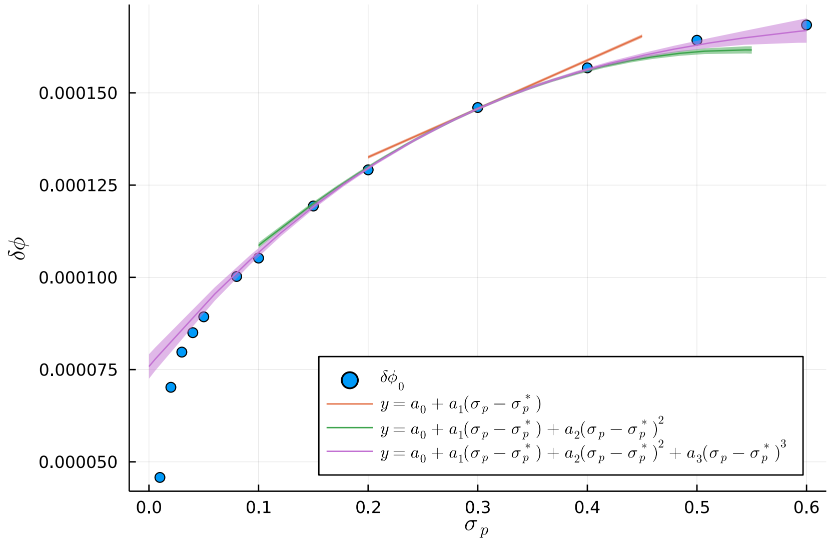

Figure 2: Variance of $\langle\phi_0\rangle$ for multiple simulations using different values of σ_p with conventional MC integration (blue points). The curves correspond to three orders of Taylor reconstruction around $\sigma_p^* = 0.3$ from either the reweighting or NSPT approach.

reconstructed. It is important to notice that all computations and error propagation needs to be done correctly order by order for secondary observables – this is done automatically using the numerical implementation of the expansion variables, eq. (7), together with the Γ-method [11]. In order to test both methods we also obtained results for different σ_p from independent simulations with the conventional methods (*i.e.* the result for σ_p^*).

The results are summarized in fig. 2 for $\delta\phi_0$ with only the results for the NSPT being shown due to the similarity with reweighting. The matching between the points from independent simulations and the Taylor series reconstruction show that with a single simulation we are able to estimate the functional dependence of an observable on a hyper-parameter of the model.

Although the precision slightly decreases with increasing orders (the equations for a given order are coupled with the lower orders) the third order approximation spans a considerable range of σ_p. In fact, the range where the expansion matches the correct value is only limited by the radius of convergence for the chosen observable – this is a property of the underlying function only, and not the method used. The method itself reproduces the exact Taylor expansion, within the precision associated with the computational investment.

Both reweighting and NSPT results match and show the same level of precision. This may lead to the conclusion that the methods are equivalent. However, it is important to relate this with the choice of a variance as the observable. While an advantage of NSPT is the fact that it does not require the computation of disconnected terms coming from the denominator of eq. (9), the variance inherently includes the subtraction of such factors, $\langle\phi_j\rangle^2$. This eliminates the advantage of NSPT and both methods provide the same level of precision. For a different choice of observable the disconnected contribution in the reweighting method becomes noticeable.

The same procedure can be obtained for secondary quantities. Considering a new point, $x_n = 0.5$, the associated prediction is given by $y(x_n) = f(x_n, \vec{\phi})$, according to the model assumption.

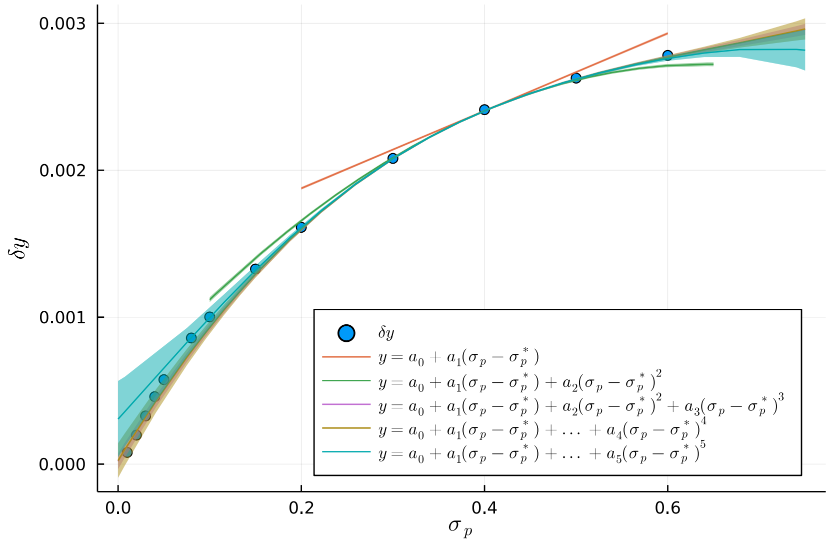

Figure 3: $\langle \delta y(0.5) \rangle$ for multiple simulations using different values of σ_p with conventional MCMC integration (blue points). The curves correspond to five different orders of Taylor reconstruction around $\sigma_p^* = 0.4$ from either the reweighting or NSPT approach.

The prediction is given by the MC average over the chain. The results for its variance $\langle \delta y(x_n) \rangle$ are shown in fig. 3 again for independent simulations and the Taylor expansion up to five orders around $\sigma_p^* = 0.4$. The matching spans a large range, extending close to vanishing σ_p where the variance necessarily vanishes (this means we are certain that the model perfectly describes the data). The decrease in precision for larger orders is also noticeable.

7. Conclusion

With a simple model based on Bayesian inference we show that it is possible to determine the Taylor expansion for functions involving Monte Carlo processes. The proper implementation of numerical series expansions allows to reconstruct the Taylor expansion up to a given order in the input parameter. Combined with the power series, both reweighting and NSPT give direct access to the functional dependence on a hyper-parameter or a coupling. It is important to notice that, in principle, this can be simultaneously applied to various parameters.

Reweighting is, in general, applicable for MCMC integration disregarding the theory. However, it may suffer from slightly larger errors due to the terms coming from the denominator of eq. (9). It should be possible to avoid the introduction of these terms by using NSPT, where the sampling algorithm operates directly on the series expansions. This is not visible in the current results due to the choice of observable.

A possible drawback of the NSPT method relates to the convergence of the solutions of the stochastic differential equations for higher orders. In the case at hand, the convergence of the equations of motion is guaranteed given that the drift term is negative for all orders. While convergence can be shown for this simple model, the same is not necessarily true for more complicated theories. For these cases the convergence should be confirmed numerically.

Within lattice field theory, although NSPT has been used for computing perturbative corrections [8–10], the applications have still been limited. In the context of QED corrections to QCD, the commonly used reweighting does not require to simulate the complete theory. However, it involves the computation of noisy disconnected diagrams which create, in general, uncontrolled systematic errors if ignored.

Finally, for machine learning these methods are important for testing the robustness of the assumptions (*i.e.* the sensitiviy of the prediction on the input parameters), for choosing optimal hyper-parameters, and to possibly estimate systematic errors.

Acknowledgements

The authors acknowledge financial support from the Generalitat Valenciana (genT program CIDEGENT/2019/040), and the Ministerio de Ciencia e Innovacion (PID2020-113644GB-I00).

References

[1] Bücker, H., Corliss, G., Hovland, P., Naumann, U. & Norris, B. *Automatic Differentiation: Applications, Theory, and Implementations*, Springer (2006).

[2] G. Jerfel, S. Wang, C. Fannjiang, K. Heller, Y. Ma, & M. Jordan, *Variational Refinement for Importance Sampling Using the Forward Kullback-Leibler Divergence*, [arXiv:2106.15980 [stat.ML]].

[3] J. Fike, & J. Alonso, *The Development of Hyper-Dual Numbers for Exact Second-Derivative*, AIAA Paper, **886** (2011).

[4] G. Parisi & Y. s. Wu, *Perturbation Theory Without Gauge Fixing*, Sci. Sin. **24, 483** (1981).

[5] P. H. Damgaard & H. Huffel, *Stochastic Quantization*, Phys. Rept. **152, 227** (1987).

[6] S. Duane, A. D. Kennedy, B. J. Pendleton & D. Roweth, *Hybrid Monte Carlo*, Phys. Lett. B **195, 216-222** (1987).

[7] G. M. de Divitiis *et al.* [RM123], *Leading isospin breaking effects on the lattice*, Phys. Rev. D **87** (2013), [arXiv:1303.4896 [hep-lat]].

[8] F. Di Renzo & L. Scorzato, *Numerical stochastic perturbation theory for full QCD*, JHEP **10** (2004), [arXiv:hep-lat/0410010 [hep-lat]].

[9] R. Kitano, H. Takaura & S. Hashimoto, *Stochastic computation of g2 in QED*, JHEP **05, 199** (2021), [arXiv:2103.10106 [hep-lat]].

[10] L. Del Debbio, F. Di Renzo & G. Filaci, *Large-order NSPT for lattice gauge theories with fermions: the plaquette in massless QCD*, Eur. Phys. J. C **78, 974** (2018), [arXiv:1807.09518 [hep-lat]].

[11] A. Ramos, *Automatic differentiation for error analysis of Monte Carlo data*, Comput. Phys. Commun. **238** (2019) – igit.ific.uv.es/alramos/aderrors.jl