Software Impacts 6 (2020) 100043

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

vanilla-option-pricing: Pricing and market calibration for options on energy commodities

Emanuele Fabbiani *, Andrea Marziali, Giuseppe De Nicolao

Department of Electrical, Computer and Biomedical Engineering, University of Pavia, Via Ferrata 1, Pavia, Italy

ARTICLE INFO

Keywords:
Option pricing
Quantitative finance
Stochastic models
Market calibration

ABSTRACT

The Python package *vanilla-option-pricing* implements procedures to price European vanilla options under the Black framework, using different stochastic models for the underlying asset. Currently, the geometric Brownian motion, the Ornstein–Uhlenbeck process and a two-factor mean-reverting process are available. The library supports market calibration, providing tools to tune the parameters of the stochastic processes against a set of listed options. The intended audience for the package is made of researchers and practitioners interested in quantitative finance and energy derivatives.

Code metadata

Current code version	0.1.0
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2020-53
Permanent link to Reproducible Capsule	https://codeocean.com/capsule/6447633/tree
Legal Code License	MIT License
Code versioning system used	git
Software code languages, tools, and services used	python
Compilation requirements, operating environments & dependencies	python ≥3.7
If available Link to developer documentation/manual	https://vanilla-option-pricing.readthedocs.io/en/latest/
Support email for questions	emanuele.fabbiani@gmail.com

1. Introduction

Since the 90s, the liberalization of energy markets in Europe and America boosted the interest in financial derivatives: European vanilla options are among the simplest and most widespread derivatives. Most methods to price such securities rely on stochastic models of the underlying asset [1]. A first issue for researchers and practitioners is thus choosing an appropriate model for a specific commodity. A second equally relevant problem is tuning the parameters of the model to fit the market situation.

In order to capture the peculiar features of energy commodities, multi-factor processes have become a popular choice [2,3]. Under the established Black framework [4,5], computing the no-arbitrage price of a European vanilla option requires the evaluation of the variance of the underlying at a given time instant. Expressions for the variance of several relevant processes are known [6,7] and recent works have proposed general methods applicable to linear stochastic models [8].

While several libraries to price options with the Black formula are available [9], to the best of our knowledge no open-source package allows for the adoption of models different from the Geometric Brownian Motion (GBM). The Python package *vanilla-option-pricing* supports two more models: the Ornstein–Uhlenbeck (OU) process and the Log-spot price mean reverting to generalized Wiener process (LMR-GW), a two-factor mean-reverting model originally proposed for power prices [10]. The modular structure of the package easily allows further contributions and extensions.

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author.
E-mail address: emanuele.fabbiani01@universitadipavia.it (E. Fabbiani).

https://doi.org/10.1016/j.simpa.2020.100043
Received 27 October 2020; Received in revised form 10 November 2020; Accepted 19 November 2020

Moreover, *vanilla-option-pricing* provides a simple way to perform market calibration, that is to fit the parameters of a stochastic model to the prices of a set of listed options.

2. Description and features

The theoretical background for *vanilla-option-pricing* is presented in [8]. The software is distributed as a Python package, which can be installed via `pip` or by cloning the GitHub repository - Code meta data Table.

The class `VanillaOption` in the module `option` represents a European vanilla option and allows the efficient computation of the implied volatility via the *py_vollib* library [9]. The classes in the module `models` allow to compute the variance of GBM, OU, and LMR-GW at a given time instant. Moreover, they provide methods to price a European vanilla option under the Black framework. Such methods compute the variance of the underlying at the option maturity date using the model represented by the class and plug the result into the Black formula. The details of the option contract can be passed either using a `VanillaOption` object or keyword arguments.

The class `ModelCalibration` in the module `calibration` provides tools to perform efficient market calibration. The problem of fitting the model parameters to the price of a given set of options is framed as a non-convex optimization program, which is solved using the `scipy` library.

3. Contributing

The code of *vanilla-option-pricing* is freely available on GitHub and pull requests are welcome. The codebase is currently covered by unit tests and a continuous integration pipeline is built on top of *GitHub actions*. Code coverage is measured by *codecov* and the documentation is automatically generated using *sphinx* and published on *Read the Docs*.

4. Impact

The software was created in the context of a research project on energy derivatives, whose main results are reported in [8,11] and is used in ongoing research at the Identification and Control of Dynamic Systems Laboratory of the University of Pavia. A talk about the application of *vanilla-option-pricing* to market calibration was delivered at Energy Finance Italy III [12].

The package was adopted by a major Italian utility in the context of their research on energy derivatives and is currently used to fit stochastic models aimed at pricing exotic derivatives. The procedures implemented in the library significantly cut the time required to calibrate the models with respect to Monte Carlo methods and enabled the adoption of multi-factor processes.

Statistics from the Python package index (PyPI) suggest that the library has been downloaded by hundreds of other users since its release.

5. Conclusion and future improvements

The package *vanilla-option-pricing* has already been adopted by both academic researchers and practitioners. However, it is still under active development: version 0.1.0 was released in October 2020 [13].

In the near future, we plan to increase the efficiency of the model calibration procedure by adopting the vectorized operations enabled by `numpy`. We also intend to feature numerical optimization methods based on the first and second derivatives of the loss function, possibly computed by an automatic differentiation tool like `jax`. Finally, we would like to include visualization tools in the package, to allow for an easy assessment of the goodness of fit achieved by model calibration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We would like to acknowledge and thank the software engineers in *xtream* for their careful review of the code and the valuable contribution in improving its quality.

References

[1] D. Pilipovic, Energy Risk: Valuing and Managing Energy Derivatives, McGraw Hill Professional, 2007.
[2] E.S. Schwartz, The stochastic behavior of commodity prices: Implications for valuation and hedging, J. Finance 52 (3) (1997) 923–973.
[3] V. Fanelli, M.D. Schmeck, On the seasonality in the implied volatility of electricity options, Quant. Finance 19 (8) (2019) 1321–1337.
[4] F. Black, M. Scholes, The pricing of options and corporate liabilities, J. Political Econ. 81 (3) (1973) 637–654.
[5] F. Black, The pricing of commodity contracts, J. Financ. Econ. 3 (1–2) (1976) 167–179.
[6] R. Kiesel, G. Schindlmayr, R.H. Börger, A two-factor model for the electricity forward market, Quant. Finance 9 (3) (2009) 279–287.
[7] E. Schwartz, J.E. Smith, Short-term variations and long-term dynamics in commodity prices, Manage. Sci. 46 (7) (2000) 893–911.
[8] E. Fabbiani, A. Marziali, G. De Nicolao, Fast calibration of two-factor models for energy option pricing, 2018, arXiv preprint arXiv:1809.03941.
[9] Gammon Capital LLC, py_vollib, GitHub, 2017, https://github.com/vollib/py_vollib.
[10] M. Barlow, Y. Gusev, M. Lai, Calibration of multifactor models in electricity markets, Int. J. Theor. Appl. Finance 7 (02) (2004) 101–120.
[11] A. Guerini, A. Marziali, G. De Nicolao, MCMC calibration of spot-prices models in electricity markets, Appl. Stoch. Models Bus. Ind. 36 (1) (2020) 62–76.
[12] E. Fabbiani, Fast calibration of two-factor models for energy option pricing, 2018, https://www.slideshare.net/EmanueleFabbiani/fast-calibration-of-twofactor-models-for-energy-option-pricing, [Online; Accessed 1 August 2020].
[13] E. Fabbiani, Donlelef/Vanilla-Option-Pricing: v0.1.0, Zenodo, 2020, http://dx.doi.org/10.5281/zenodo.4107182.