The Journal of Open Source Software

DataInterpolations.jl: Fast Interpolations of 1D data

Sathvik Bhagavan [1¶], **Bart de Koning** [2], **Shubham Maddhashiya**[3], and **Christopher Rackauckas** [1,3,4]

1 JuliaHub **2** Deltares **3** Pumas-AI **4** Massachusetts Institute of Technology ¶ Corresponding author

DOI: 10.21105/joss.06917

Software

- Review
- Repository
- Archive

Editor: Arfon Smith

Reviewers:

- @carstenbauer
- @dawbarton

Submitted: 06 June 2024
Published: 05 September 2024

Summary

Interpolations are used to estimate values between known data points using an approximate continuous function. DataInterpolations.jl is a Julia (Bezanson et al., 2017) package containing 1D implementations of some of the most commonly used interpolation functions. These include:

- Constant Interpolation
- Linear Interpolation
- Quadratic Interpolation
- Lagrange Interpolation (Lagrange, 1898)
- Quadratic Splines
- Cubic Splines (Schoenberg, 1988)
- Akima Splines (Akima, 1970)
- Cubic Hermite Splines
- Piecewise Cubic Hermite Interpolating Polynomial (PCHIP) (Fritsch & Butland, 1984)
- Quintic Hermite Splines
- B-Splines (Curry & Schoenberg, 1988) (de Boor, 1972)
- Regression based B-Splines

and a continually growing list. Along with these, the package also has methods to fit parameterized curves with the data points and Tikhonov regularization (Tikhonov, 1943) (Petters, 2021) for obtaining smooth curves. The package also provides functionality to compute integrals and derivatives upto second order for those interpolations methods. It is also automatic differentiation friendly. It can also be used symbolically with Symbolics.jl (Gowda et al., 2022) and plugged into models defined using ModelingToolkit.jl (Ma et al., 2021).

Statement of need

Interpolations are a very important component of many modeling workflows. Often, sampled or measured inputs need to be transformed into continuous functions or smooth curves for simulation purposes. In many scientific machine learning workflows, interpolating data is essential to learn continuous models. DataInterpolations.jl can be used for facilitating these types of workflows. Several interpolation packages already exist in Julia, such as Interpolations.jl, which primarily specializes in B-Splines and uniformly spaced data with some support for irregularly spaced data. In contrast, DataInterpolations.jl does not assume any specific structure in the data, offering greater flexibility for diverse datasets. Interpolations.jl also doesn't offer methods like Quadratic Interpolation, Lagrange Interpolation, Hermite Splines etc. BasicInterpolators.jl is more similar to DataInterpolations.jl, although it doesn't offer methods like B-Splines. Rest of the interpolation packages focus on particular methods like BSplineKit.jl for B-Splines, FastChebInterp.jl for Chebyshev interpolation, PCHIPInterpolation for PCHIP interpolation etc. Additionally, DataInterpolations.jl includes many novel techniques for accelerating the interpolation searches with specialized caching, quasi-linear guessing, and more to improve the performance algorithmically, beyond the simple computational optimizations. In summary, DataInterpolations.jl is more generic from other packages and offers many fast interpolation methods for arbitrarily spaced 1D data, all within a consistent and simple interface.

Example

The following tutorials are provided in the documentation:

- Tutorial 1 provides how to define each of the interpolation methods and compute the value at any point.
- Tutorial 2 provides explanation for using the interface and interpolated objects for evaluating at any point, computing the derivative at any point and computing the integral between any two points.
- Tutorial 3 provides how to use interpolation objects with Symbolics.jl and ModelingToolkit.jl.

A simple demonstration here:

```julia
using DataInterpolations

# Dependent variable
u = [14.7, 11.51, 10.41, 14.95, 12.24, 11.22]

# Independent variable
t = [0.0, 62.25, 109.66, 162.66, 205.8, 252.3]

A1 = CubicSpline(u, t)

# For interpolation do, A(t)
A1(100.0)

# derivative
## first order
DataInterpolations.derivative(A1, 1.0, 1)

## second order
DataInterpolations.derivative(A1, 1.0, 2)

# integral
DataInterpolations.integral(A1, 1.0, 5.0)
```

References

Akima, H. (1970). A new method of interpolation and smooth curve fitting based on local procedures. *J. ACM*, *17*(4), 589–602. https://doi.org/10.1145/321607.321609

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, *59*(1), 65–98. https://doi.org/10.1137/141000671

Curry, H. B., & Schoenberg, I. J. (1988). On pólya frequency functions IV: The fundamental spline functions and their limits. In C. de Boor (Ed.), *I. J. Schoenberg selected papers* (pp. 347–383). Birkhäuser Boston. https://doi.org/10.1007/978-1-4899-0433-1_17

de Boor, C. (1972). On calculating with b-splines. *Journal of Approximation Theory*, *6*(1), 50–62. https://doi.org/10.1016/0021-9045(72)90080-9

Fritsch, F. N., & Butland, J. (1984). A method for constructing local monotone piecewise cubic interpolants. *SIAM Journal on Scientific and Statistical Computing*, *5*(2), 300–304. https://doi.org/10.1137/0905021

Gowda, S., Ma, Y., Cheli, A., Gwóźzdź, M., Shah, V. B., Edelman, A., & Rackauckas, C. (2022). High-performance symbolic-numerics via multiple dispatch. *ACM Commun. Comput. Algebra*, *55*(3), 92–96. https://doi.org/10.1145/3511528.3511535

Lagrange, J. L. (1898). *Lectures on elementary mathematics, by joseph louis lagrange; tr. By thomas j. McCormack*. Open court publishing Company. https://books.google.co.in/books?id=7h9G0AEACAAJ

Ma, Y., Gowda, S., Anantharaman, R., Laughman, C., Shah, V., & Rackauckas, C. (2021). *ModelingToolkit: A composable graph transformation system for equation-based modeling*. https://doi.org/10.48550/arXiv.2103.05244

Petters, M. D. (2021). Revisiting matrix-based inversion of scanning mobility particle sizer (SMPS) and humidified tandem differential mobility analyzer (HTDMA) data. *Atmospheric Measurement Techniques*, *14*(12), 7909–7928. https://doi.org/10.5194/amt-14-7909-2021

Schoenberg, I. J. (1988). Contributions to the problem of approximation of equidistant data by analytic functions. In C. de Boor (Ed.), *I. J. Schoenberg selected papers* (pp. 3–57). Birkhäuser Boston. https://doi.org/10.1007/978-1-4899-0433-1_1

Tikhonov, A. N. (1943). On the stability of inverse problems. *Proceedings of the USSR Academy of Sciences*, *39*, 195–198. https://api.semanticscholar.org/CorpusID:202866372