PolyMath version 1.0 and beyond

Serge Stinckwich, Oleksandr Zaytsev
Hemal Varambhia, Konrad Hinsen
Atharva Khare, Nikhil Pinnaparaju
Werner Kassens

UMMISCO Unite Mixte Internationale
SORBONNE UNIVERSITÉ CRÉATEURS DE FUTURS DEPUIS 1257
IRD Institut de Recherche pour le Développement FRANCE

Current Status of PolyMath

- A generic computation framework for Pharo
- MIT Licence
- PolyMath 1.0 release
- https://github.com/PolyMathOrg/PolyMath
- 806 green unit tests, 1034 commits, 100 stars on github (9th Pharo project on github)

Since last year

- Release 1.0 version + new logo
- Clean the code (All PM prefix classes), add more unit tests
- GSOC students implement : PCA and t-SNE
- Random Numbers cleaning (Hemal)
- Automatic Differentiation package cleaning (Serge)
- Data Transformer hierarchy (PMStandardizationScaler) like in scikit-learn
- Trunk-based development since 1.0 (all code is committed to master branch + one release branch)
- Add a contributing guide and code of conduct

Contributors

- Thank you !

[PolyMath]

Moose analysis

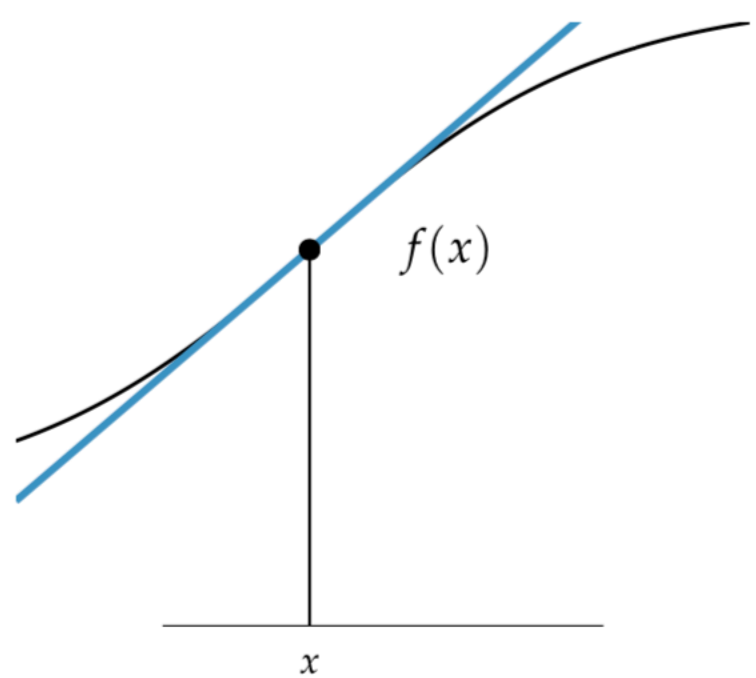
DEMO

Automatic Differentiation

- Explain why this topic is interesting especially in the context of Machine Learning

Derivatives

- The derivatives f'(x) of a function of a single variable x is the rate at which the value of f changes at x.
- How to compute derivatives numerically ?



$$f'(x) = \frac{\Delta f(x)}{\Delta x}$$

Numerical differentiation by using finite difference

$$f'(x) \approx \underbrace{\frac{f(x+h)-f(x)}{h}}_{\text{forward difference}} \approx \underbrace{\frac{f(x+h/2)-f(x-h/2)}{h}}_{\text{central difference}} \approx \underbrace{\frac{f(x)-f(x-h)}{h}}_{\text{backward difference}}$$

- Theoretically, the smaller the h is, the better the derivative estimates
- Practically very small value of h can result in numerical cancellation errors …

```
f := [ :x| x sin ].
x := 1/2.
errorDiffForward := [:h| ((((f value: (x+h)) - (f value: x)) / h) - (x cos)) abs ].

errorDiffForward value: 0.1.
```

Numerical differentiation by the Complex Step Method

- The Complex Step Method bypass the subtractive errors, by using only one single function evaluation (but more complex).

- Taylor expansion for an imaginary step is:

$$f(x+ih) = f(x) + ihf'(x) - h^2\frac{f''(x)}{2!} - ih^3\frac{f'''(x)}{3!} + \cdots$$

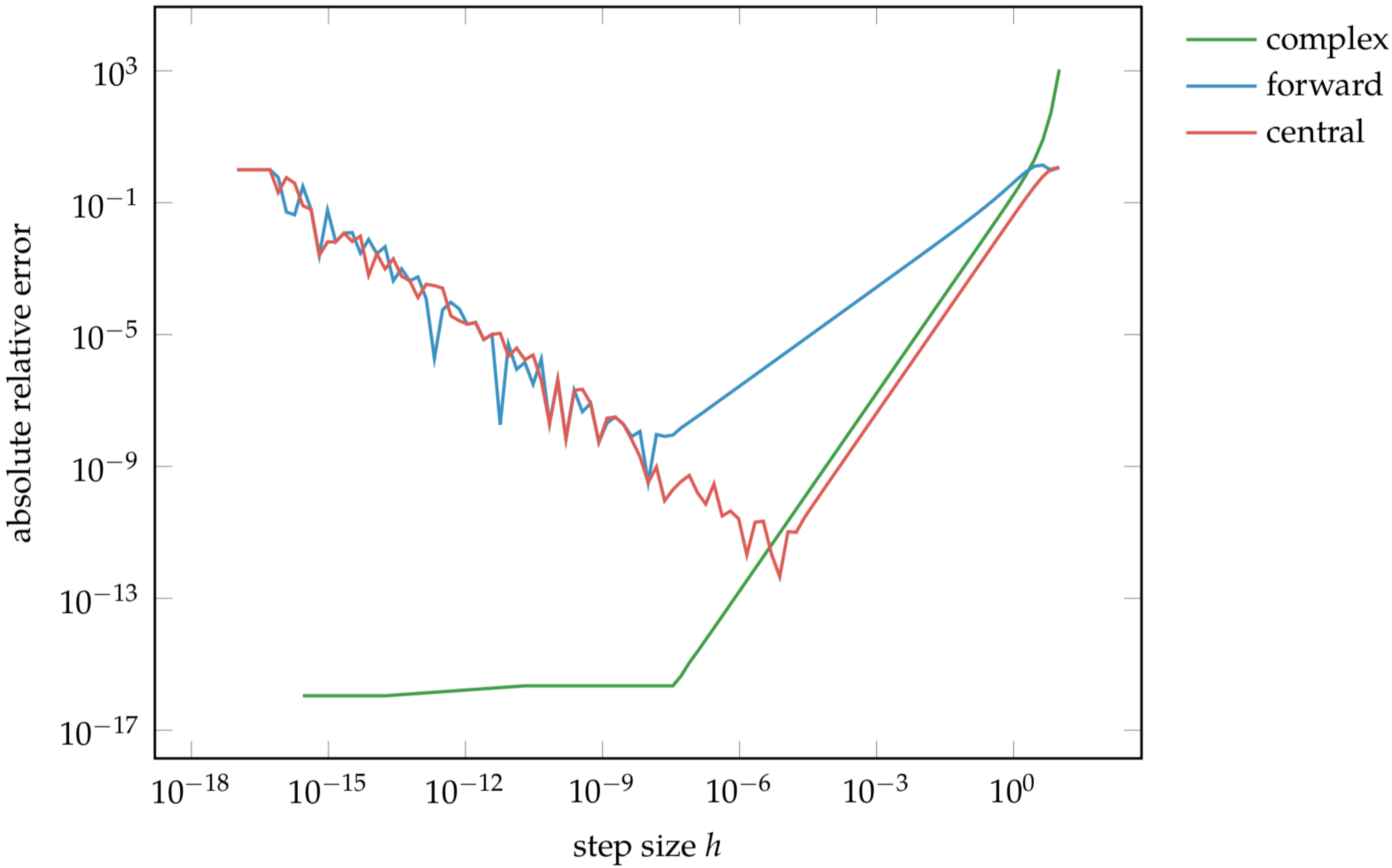
$$\mathrm{Im}(f(x+ih)) = hf'(x) - h^3\frac{f'''(x)}{3!} + \cdots$$

$$\begin{aligned}\Rightarrow f'(x) &= \frac{\mathrm{Im}(f(x+ih))}{h} + h^2\frac{f'''(x)}{3!} - \cdots \\ &= \frac{\mathrm{Im}(f(x+ih))}{h} + O(h^2) \text{ as } h \to 0\end{aligned}$$

Consider $f(x) = \sin(x^2)$. The function value at $x = \pi/2$ is approximately 0.624266 and the derivative is $\pi\cos(\pi^2/4) \approx -2.45425$. We can arrive at this using the complex step method:

```
f := [ :x| (x*x) sin ].
pi := Float pi.
x := pi /2.
h := 0.001.
((f value: (h*1i +x )) imaginary) / h.

pi * ((pi*pi/4) cos)
```



absolute relative error
10^{3}
10^{-1}
10^{-5}
10^{-9}
10^{-13}
10^{-17}
10^{-18}
10^{-15}
10^{-12}
10^{-9}
10^{-6}
10^{-3}
10^{0}
step size h
complex
forward
central

Automatic Differentiation

- Introduce DualNumber library in PolyMath
- Give an example

t-SNE

- Introduce what is t-SNE (dimensionality reduction)
- Demo of Atharva
- Demo of Oleks to include here

Future of PolyMath

- ActivePapers (1 slide)
- Domains (1 slide)
- Towards Scientific AI (1 slide)
- Fundings (1 slide)