

# Jupyter, aneb vědecké výpočty moderně a snadno

Robert Mařík

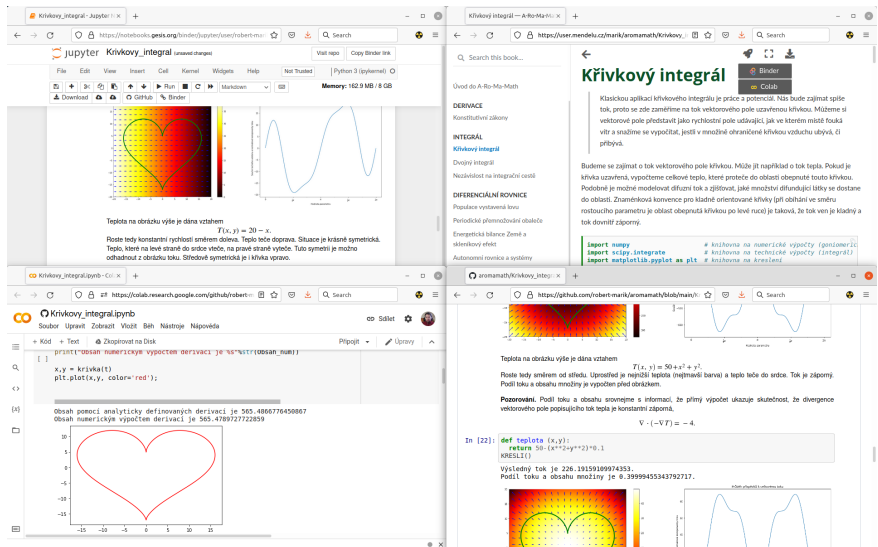
Rychlý rozvoj Internetu, virtualizačních nástrojů a cloudových služeb v posledních letech zcela změnil způsob práce uživatelů, kteří se zabývají vědeckými či jakýmkoliv výpočty. Výpočetní nástroje mají běžně možnost provádět výpočty v cloudu a uživatel může tedy pracovat v podstatě na jakkoli výpočetně slabém zařízení. Samozřejmostí je sdílení souborů se spolupracovníky nebo se všemi uživateli Internetu. Pokud není přímo k dispozici možnost používat některý z verzovacích systémů (například kvůli binárnímu ukládání dat), mívají výpočetní prostředí v sobě integrovánu možnost ukládání verzí, návratu k dřívější verzi nebo zobrazení rozdílu mezi verzemi. Samozřejmostí je, že delší poznámky se nepiší mezi řádky kódu za komentářové znaky, ale je možné přirozeně kombinovat výpočty s textem. Mimo jiné i díky tomu je moderní věda reprodukovatelná, otevřená a férová (z ustálených anglických obrátů *reproducible science*, *open science* a *fair science*).

## 1 Jupyter

V tomto příspěvku si představíme otevřený nástroj, který umožní provádět libovolné výpočty moderním přístupem se všemi vlastnostmi popsány v úvodu, tj. výpočty v cloudu, s možností sdílení a se správou verzí. Jedná se prostředí Jupyter, resp. Jupyter Notebook, [1]. Jupyter je ekosystém vědeckých nástrojů pro numerické výpočty, symbolické výpočty, kreslení grafů a obrázků, statistické zpracování dat, analýzu obrazových nebo zvukových dat, analýzu textu, strojové učení a rozpoznávání textu či obrázků a mnoho dalšího.

Z uživatelského hlediska se Jupyter chová jako webová aplikace pro přístup k programovacím jazykům Python, Julia a statistickému jazyku R. Odsud, z názvů těchto tří jazyků, je ostatně složen i název projektu. Zejména volba jazyka Python se ukazuje velice šťastná, protože ve všech oborech zmíněných v předchozím odstavci, patří Python a jeho knihovny mezi nejvýkonnější nástroje. To platí ať výkon posuzujeme z hlediska vzdělávání budoucích inženýrů a výpočtářů, nebo z hlediska výpočtářů řešících úlohy v inženýrské praxi. To, že vše je umístěno v jednotném prostředí, umožňuje zmíněné nástroje použít jako stavební kostky k řešení sofistikovaných úloh, které mohou kombinovat náročnou práci s obrazovými vstupy, numerické výpočty nad obrovskými

objemy dat, vizualizaci výstupů a případně export výpočtů ve formě csv souborů nebo v jakémkoliv běžně užívaném formátu.<sup>1</sup>



Obr. 1: Jupyter zápisník otevřený v čtyřech prostředích. Vpravo dvě statické verze pro vystavení nebo sdílení, vlevo dvě dynamické verze umožňující spouštět vlastní příkazy.

Bylo by faux-paux mluvit o moderních výpočetních metodách a rozepisovat se detailně o způsobu práce, příkazech, nastaveních. Všechno máte na dosah ruky, tak proč to rovnou nezkusit? Ukázkou jednoduchých učebních textů opírajících se o výpočty v prostředí Jupyter je webový miniportál A-Ro-Ma-Math [2]. Jedná se o takzvaný Jupyter-book, kniha sestavená ze zápisníků Jupyter-notebook. Jeden z těchto zápisníků, zápisník Křivkový integrál, je na obrázku otevřen ve čtyřech různých provedeních ze čtyřech různých lokalit a tím ve čtyřech různých podobách.

- Vpravo nahoře je dokument otevřen ze základního umístění na domovské instituci autora. Po rozkliknutí ikonky s raketou v pravém horní rohu se nabízí možnost otevřít dokument v interaktivní pobobě s možností dokument měnit a spouštět příkazy. V současnosti jsou aktivní možnosti využít projektů Binder a Colab.
- V levé polovině je dokument otevřen v prostředí Binder (nahore) a Colab (dole). Zde pracujete s kopií původního souboru a můžete si jej libovolně upravovat, expe-

<sup>1</sup> Z inženýrského hlediska zde chybí pravděpodobně jenom metoda konečných prvků, pro kterou sice také máme k dispozici knihovnu, ale ta nedosahuje kvalit svých komerčních konkurentů.

rimentovat s příkazy a poté uložit buď do lokálního souboru, nebo v případě Colab i na Google Drive nebo do repozitáře GitHub.

- Vpravo dole je tentýž dokument v prostředí Github, v prostředí zaměřeném na správu verzí dokumentů a softwarových projektů. Toto prostředí umí přímo renderovat Jupyter zápisníky tak, jak je vidíme při práci. Používá se pro sdílení, ale sdílet je možno samozřejmě i soubor uložený v Colab na Google Drive.

Díky balíčkovacímu systému jazyka Python je možné a snadné si Jupyter nainstalovat na vlastní počítač s libovolným běžným operačním systémem. Potom je možno pracovat bez připojení k Internetu a být tak odolní vůči případným omezením služeb. Práce v cloudu jako je Binder či Colab je však často pohodlnější. Odpadá například veškeré ukládání při přenášení na jiný počítač (učebna, vedlejší kancelář, práce, domov, ...). Přes nutnost přistupovat ke cloudovým službám není nutné se bát malého výkonu. Naopak. Pro strojové učení je začátečníkům prostředí Colab zpravidla doporučováno, protože poskytne krátkodobě dostatečný výpočetní výkon i uživatelům bez drahých grafických karet, které jsou pro tyto aplikace zpravidla téměř nutností.

## 2 Knihovny pro Jupyter

Pokud tedy máte spuštěn Jupyter v prostředí Colab, Binder, nebo z URL <https://jupyter.org/try> nebo jste investovali úsilí do instalace na vlastní počítač, můžete se pustit do učení a do práce. Níže si uvedeme jednotlivé součásti. Každá z nich má obsáhlou dokumentaci a nápovědu, často i podrobnou nástěnku a ukázkami výstupů. Proto je zbytek textu míněn spíše jako návěstí u rozcestníku. Co čtenáře zaujme, to může okamžitě vyzkoušet a není těžké vygooglit jednoduché i sofistikované ukázky možností a aplikací.

### 2.1 Python

Python je jeden z nejjednodušších programovacích nebo skriptovacích jazyků. Pokud nějaký skriptovací jazyk používáte, nemusíte se pro začátek Python pravděpodobně vůbec učit. Stačí se na příkladech seznámit se syntaxí. Můžete se rovnou podívat na použití v dokumentu [3], který oproti svému názvu není zaměřen pouze na knihovnu Scipy, ale pokrývá i Python obecně a řadu dalších knihoven, zmíněných níže.

Pokud jste v programování začátečníkem, existuje řada kurzů, jak do problematiky jazyka Python snadno proniknout, například [4] nebo v češtině [5].

## 2.2 Numpy

Numerické výpočty zajišťuje knihovna numpy, což je zkratka pro Numerical Python. Knihovna definuje datové struktury a nástroje pro snadnou manipulaci a matematické numerické výpočty s čísly nebo s číselnými poli. S knihovnou je možné se seznámit v tutoriálu [4].

## 2.3 Scipy

Knihovna pro vědecké výpočty, obsahující nástroje a funkce pro optimalizaci, interpolaci, numerickou integraci, řešení diferenciálních rovnic, grafové algoritmy a další časté inženýrské úlohy. Viz [3], [4]. Knihovna umožňuje ukládání a načítání datových struktur uložených programem Matlab.

V [2] je knihovna scipy použita k řešení diferenciálních rovnic.

## 2.4 Matplotlib

Knihovna pro kreslení grafů všeho druhu je knihovna matplotlib. Nejlepší vypovídací hodnotu o možnostech má návštěva domovské stránky projektu [6] a sekce Examples a Tutorials. Další tutoriál je součástí [4].

Nejedná se o jedinou knihovnu pro kreslení grafů v jazyce Python. Pro kreslení statistických grafů je možno zvážit jako alternativu Seaborn, pro kreslení interaktivních grafů umožňujících zoom, posun nebo vypínání některých křivek je Bokeh.

## 2.5 Pandas

Pandas je knihovna pro práci s datovými soubory, pro práci se seznamy a tabulkami čísel nebo textových údajů. Takové tabulky používáme v databázích nebo v tabulkových procesorech. Přirozeně jsou takové tabulky používány Excelem a co je možné vypočítat v Excelu je možné vypočítat knihovnou Pandas. Navíc můžete díky skriptování snadno automatizovat činnosti, práce je mnohem přehlednější, nejste omezeni velikostí datových souborů snadno se kombinují data z více zdrojů a spousta dalších činností je v Pandas mnohem efektivnější. Proto se vyplatí do studia investovat čas. Překousnout, že první tabulku se součty a průměry budete v Pandas dělat možná dvacetkrát déle než v Excelu. Jakmile stylu práce přijde člověk na kopylku, je práce u běžných činností minimálně stejně efektivní a navíc reprodukovatelná, transparentní (použití příkazů versus postupné klikání v GUI) a snadno začlenitelná do komplexnějších projektů. Bonusem je, že práce s malými datovými soubory stejně náročná jako práce s těmi obrovskými, které byste do Excelu ani nenačetli. Na nástroje typu MS Excel je dobré nezapomínat. Jsou

široce rozšířené, jsou v povědomí mnoha lidí a jsou dobré na krátké analýzy. Na hlubší práci spojenou kromě analýzy například i s čištěním dat je však rozhodně lepší používat Pandas.

Pandas je možné se naučit ve většině kurzů věnovaných využití jazyka Python. Můžete začít například na tutoriálu [4] a pokračovat některým z široké nabídky online kurzů na zpracování dat a práci s daty na portálu [7].

## 2.6 SymPy

SymPy je kompletní systém počítačové algebry v jazyce Python. Obsahuje diferenciální a integrální počet, diferenciální rovnice, lineární algebru, řešení rovnic, kombinatoriku a další nástroje pro symbolické výpočty. Dokumentace na domovské stránce projektu obsahuje i tutoriál a ukázky je možno spouštět přímo z textu dokumentace.

## 2.7 Další knihovny

Výše uvedený seznam není jistě vyčerpávající. Slouží pro rychlé seznámení se s těmi nejpodstatnějšími knihovnami. Existuje i celá řada dalších, které jsou specializovány na jiné činnosti. Například scikit-learn nebo TensorFlow pro strojové učení, Biopython s nástroji pro bioinformatiku, PySB pro systémovou biologii a modelování biochemických systémů a řada dalších.

# 3 Závěr

Výše představené nástroje nejsou interaktivní, jedná se o programovací (skriptovací) jazyk a na rozdíl od nástrojů s GUI je nutné se práci nějakou dobu učit. Nabízí se otázka, za se jedná o skutečně použitelné a používané nástroje, do kterých je vhodné investovat čas a úsilí. Odpověď je možné ilustrovat situací na pravděpodobně nej kvalitnější a nej obsáhlejší elarningovém portále edX [7]. Zde je naprostá většina kurzů zaměřených na zpracování dat nebo na práci s daty postavena okolo jazyka Python a často v prostředí Jupyter. Nejedná se přitom jenom o univerzitní kurzy, ale i o kurzy připravené technologickými firmami jako IBM. To ukazuje popularitu a garantuje životaschopnost inženýrských výpočtů a zpracování dat jazykem Python v prostředí Jupyter.

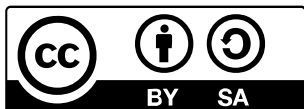
Jako perličku na závěr, která ilustruje kvalitu představených produktů, bych uvedl jeden příklad. Lidé se často ptají, zda je lepší Pandas nebo R. Málokoho napadne do srovnání přibrat ještě dalšího kandidáta. Přirozeně srovnávají dva nejvyspělejší produkty v oblasti statistiky, dva open-source projekty které daleko utekly konkurenci. Oba tyto produkty je možno provozovat v zápisníku Jupyter.

## Literatura

1. *Project Jupyter*, <https://jupyter.org/>
2. MAŘÍK, R. *A-Ro-Ma-Math*, <https://user.mendelu.cz/marik/aromamath> (web),  
<https://github.com/robert-marik/aromamath/> (GitHub repozitář)
3. *Scipy Lecture notes*, <https://scipy-lectures.org/>
4. *Python Tutorial*, <https://www.w3schools.com/python>
5. *Nauč se Python*, <https://nauce.python.cz/>
6. *Knihovna matplotlib*, <https://matplotlib.org/>
7. *edX*, <https://www.edx.org/>

## Autor

Doc. Mgr. Robert Mařík, Ph.D., Ústav matematiky, Lesnická a dřevařská fakulta, Mendelova univerzita v Brně, Zemědělská 1, 613 00 Brno, e-mail: robert.marik@mendelu.cz



Open Access. This article is licensed under the terms of the Creative Commons Attribution-ShareAlike 4.0 International License, CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>)