Ťahák ANALÝZA – Jupyter + Sage

Predmety ZMF, DGS – Jozef Hanč

Podľa ref. William Stein, Sage Quick Reference: Calculus
http://wiki.sagemath.org/quickref

Vstavané konštanty, funkcie, množiny

Konštanty: $\pi =$ `pi` $e =$ `e` $\infty =$ `oo` $=$ `infinity`
$i =$ `I` $=$ `i` $\log(2) =$ `log2` $\phi =$ `golden_ratio` ...

Približne: `pi.n(digits=18)` $= 3.14159265358979324$

Vstavané funkcie: `sin cos tan cot sec csc sinh cosh tanh coth sech csch exp ln log` ...

Množiny (okruhy): $\mathbb{Z} =$ `ZZ` $\mathbb{Q} =$ `QQ` $\mathbb{R} =$ `RR` $\mathbb{C} =$ `CC` ...

Komplexné čísla $z = a + bi$: `2+3*i; 1/2-sqrt(2)*I`
- reálna a imaginárna časť z: `z.real(); z.imag()`
- modul a argument z: `abs(z); arg(z)`
- komplexne združené $\bar{z}$: `z.conjugate()`

Definícia symbolických výrazov

Vytvorenie symbolických premenných t, u, θ:
`t,u,theta = var('t,u,theta')` alebo `%var t,u,theta`

Použi vždy `*` pre násobenie, `^` alebo `**` pre umocňovanie:
$2x^5 + \sqrt{2}$ = `2*x^5 + sqrt(2)` = `2*x**5 + sqrt(2)`

Štand. mat. vzhľad: všetky výstupy `%display latex`
jeden výstup `show(2*theta^5+sqrt(2))` $\longrightarrow 2\theta^5 + \sqrt{2}$

Vlastné symbolické funkcie

Vlastná funkcia (možno ju integrovať, derivovať, atď.):
`f(a,b,theta) = a + b*theta^2`

Formálna symbolická funkcia f závisiaca na premennej t:
`f(t) = function('f')(t)` alebo `f = function('f')(t)`

Vlastné pythonovské funkcie

Definícia (s nepovinným prednast. parametrom *theta*):
```
def f(a, b, theta = 1.5):
  value = a + b*theta^2
  return value
```
Ekvivalentná jednoriadková definícia:
`f = lambda a,b,theta=1.5: a + b*theta^2`

Jednoduchý graf a body funkcie

2D graf na $\langle a, b\rangle$: `plot(f(x), (x,a,b))`
`g = plot(x*sin(x), (x,-2,10)); show(g)`

Body v grafe g: `points[`*zoznam bodov, farba, veľkosť bodu*`]`
`z = (2,2), (4,-2), (1/2,1/2);`
`b = points(z,color='red',pointsize=50); show(g+b)`

3D graf na $\langle a, b\rangle \times \langle c, d\rangle$: `plot(f(x,y),(x,a,b),(y,c,d))`
`p = plot3d(x*sin(y),(x,-5,5),(y,-5,5)); show(p)`

Zjednodušovanie a substitúcia

Symbolický výraz `f` alebo funkciu `f(x,y,...)` možno zjednodušiť: `f.simplify()`, `f.full_simplify()`, `f.canonicalize_radical()`

roznásobiť: `f.expand()`, združiť členy s x: `f.collect(x)`

trigonometrické funkcie: `f.trig_simplify()`, `f.trig_expand()`, `f.trig_reduce()`

substitúcia (za výraz `v1` dosadíme `v2`): `f.subs(v1==v2)`

Rozklad na súčin

`f.factor()`: `(x^3-y^3).factor()` alebo `360.factor()`

Zoznam dvojíc (činiteľ, jeho exponent):
`(x^3-y^3).factor_list()` alebo `list(360.factor())`

Relácie

Relácie $f = g$: `f == g`, $f \neq g$: `f != g`, $f \leqq g$: `f <= g`, $f \geqq g$: `f >= g`, $f < g$: `f < g`, $f > g$: `f > g`

Rovnice, nerovnice, sústavy

Množina (zoznam) riešení $f(x) = g(x)$ alebo $f(x) \geqq g(x)$:
`solve(exp(2*x)==1/7,x); solve(x^2-6 >= 8,x)`

Výpis jednotlivých riešení rovnice – použitie *slovníka*
`s = solve(ln(x^2) == 5/3,x, solution_dict=True)`
`s[0][x], s[1][x]` sú jej dve riešenia (*Sage čísluje od 0*)

Presné korene polynómu: `(x^3+2*x+1).roots(x)`
Reálne korene: `(x^3+2*x+1).roots(x,ring=RR)`
Komplexné korene: `(x^3+2*x+1).roots(x,ring=CC)`

Zoznam riešení sústavy rovníc r_1, r_2:
`r1 = (x^2+y^2==1); r2 = ((x-1)^2+y^2==1);`
`solve([r1,r2],x,y)`

Fitovanie a optimalizácia

Fitovanie (MNŠ): dáta = *zoznam bodov*, model = *funkcia*
`data = (0,1),(1,1/2),(2,0); model(x) = a+b*x;`
`find_fit(data,model)`

Riešenie rovnice $f(x) = 0$ na $\langle a, b\rangle$: `f.find_root(a, b)`
`f(x) = x^2 - 2; f.find_root(1,2)`

Nájdenie maxima alebo minima $f(x)$ na $\langle a, b\rangle$:
`f.find_local_maximum(a, b)`
`f.find_local_minimum(a, b)`

Limity

$\lim_{x\to a} f(x) =$ `limit(f(x), x=a)`
$\lim_{x\to a^+} f(x) =$ `limit(f(x), x=a, dir='plus')`
$\lim_{x\to a^-} f(x) =$ `limit(f(x), x=a, dir='minus')`
`limit(1/x, x=0, dir='minus')`; `limit(1/x^2, x=-oo)`

Derivácie

$\frac{d}{dx} f(x) =$ `diff(f(x),x) = f.diff(x)`
$\frac{\partial}{\partial x} f(x, y) =$ `diff(f(x,y),x)`
`diff` = *diferencovať* = *derivovať*
`diff(x*y + sin(x^2) + e^(-x), x)`

Integrály

Neurčitý: $\int f(x)dx =$ `integral(f,x)` = `f.integrate(x)`
`integral(x*cos(x^2), x)`

Rozklad na parciálne zlomky:
`(x^2/(x+1)^3).partial_fraction()`

Určitý: $\int_a^b f(x)dx =$ `integral(f,x,a,b)`
`integral(x*cos(x^2), x, 0, 1)`

Numerický výpočet integrálu — [0] výsledok, [1] chyba:
$\int_a^b f(x)dx \approx$ `numerical_integral(f(x),a,b)[0]`
`numerical_integral(x*cos(x^2),0,1)[0]`

Použi `assume(...)`, ak integrácia žiada rozhodnutie o parametri integrálu, napr. `assume(n>0)`

Obyčajné diferenciálne rovnice

Diferenciálna rovnica (DR) 1. rádu: $y' = f(x, y)$
`y(x) = function('y')(x)`
`DR1 = diff(y,x)==f(x,y)`

Lineárna DR 2. rádu: $y'' + p(x)y' + q(x)y = f(x)$
`DR2 = diff(y,x,2)+p(x)*diff(y,x)+q(x)*y==f(x)`

Riešenie všeobecné (`ivar` = *nezávislá premenná*):
`desolve(DR1, y, ivar=x); desolve(DR2, y, ivar=x)`

Riešenie s počiatoč. podmienkou (x_0, y_0), resp. (x_0, y_0, y_0'):
`desolve(DR1, y, ivar=x, ics=[x0,y0])`
`desolve(DR2, y, ivar=x, ics=[x0,y0,y0'])`

Vektorová analýza

Skalárna funkcia (pole): `f(vars)`, kde `vars` = `x,y,z,...`
`g(x,y) = x^2*y+y^2+y`
`f(x,y,z) = sin(x^2+y^2+z^2)`

Gradient a Hessián skalárnej funkcie (poľa):
`f.gradient()` `f.hessian()`
`gradg = g.gradient(); gradg(1,1)`

Vektorová funkcia (pole): `(f1(vars), ... ,fn(vars))`
`f(x,y,z) = (sin(y), cos(x), tan(z))`

Jakobián vektorovej funkcie: `jacobian(f,[vars])`
`Jf = jacobian(f,[x,y,z]); Jf(2,1,-3)`

Divergencia a rotácia vektor. poľa: `f.div()`, `f.curl()`
`divf = f.div(); divf(1,1,z)`
`rotf = f.curl(); rotf(x,y,z)`