

آزمایشگاه ریاضی (آموزش Sage)

درس سوم:

آشنایی با توابع و رسم نمودار

مدرس: میثم مدنی madani@mehr.sharif.ir
دانشکده علوم ریاضی - دانشگاه صنعتی شریف

ترم اول سال ۱۳۹۱

نسخه ابتدایی (لطفا نظرات خودت و ایرادات فایل را ایمیل کنید)

رسم نمودار

- ۱- نمودارهای دو بعدی
- ۲- نمودارهای سه بعدی
- ۳- طراحی رابط کاربری

آشنایی

□ فرم ساده رسم نمودار به فرم زیر است.

```
plot(sin, (-2*pi, 2*pi), thickness=2.0, rgbcolor=(0.5, 1, 0)  
legend_label='sin(x)')
```

□ اختیاراتی که داریم

```
sage: plot.options
```

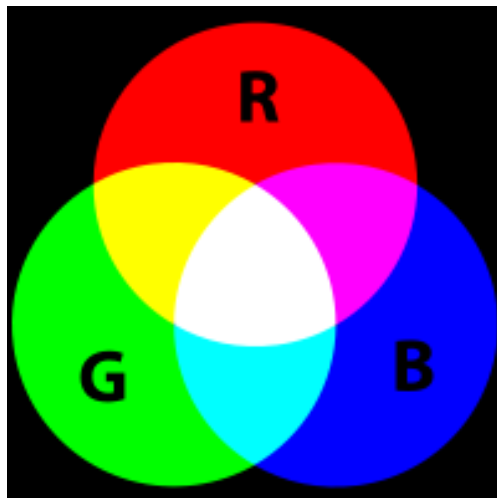
```
{'fillalpha': 0.5, 'detect_poles': False, 'plot_points': 200,  
'thickness': 1, 'alpha': 1, 'adaptive_tolerance': 0.01, 'fillcolor':  
'automatic', 'adaptive_recursion': 5, 'exclude': None,  
'legend_label': None, 'rgbcolor': (0, 0, 1), 'fill': False}
```

اختیارات در رسم نمودار

اختیارات

میزان مرئی بودن پر کردن نواحی (0=opaque, 1=transparent)	alpha	<input type="checkbox"/>
اگر True باشد زیر نمودار به صورت پر رنگ می شود	fill	<input type="checkbox"/>
میزان مرئی بودن پر کردن نواحی (0=opaque, 1=transparent)	fillalpha	<input type="checkbox"/>
رنگ پر کردن نواحی	fillcolor	<input type="checkbox"/>
رنگ خط	rgbcolor	<input type="checkbox"/>

اختیارات در رسم نمودار



رنگ های rgb

ترکیب سه رنگ آبی و سبز و قرمز

لیست رنگهای از پیش تعیین شده را می توان با

`sorted(color)`

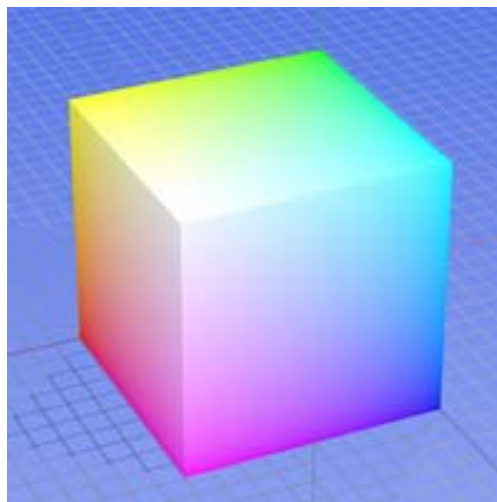
مشاهده کرد.

در ضمن در صورت تولید کد رنگ دلخواه

می توانید از نرم افزاری مثل `paint` استفاده کنید

(البته بایستی تمامی اعداد را تقسیم بر ۲۵۵ کنید)

`rgbcolor(1,0,0)`



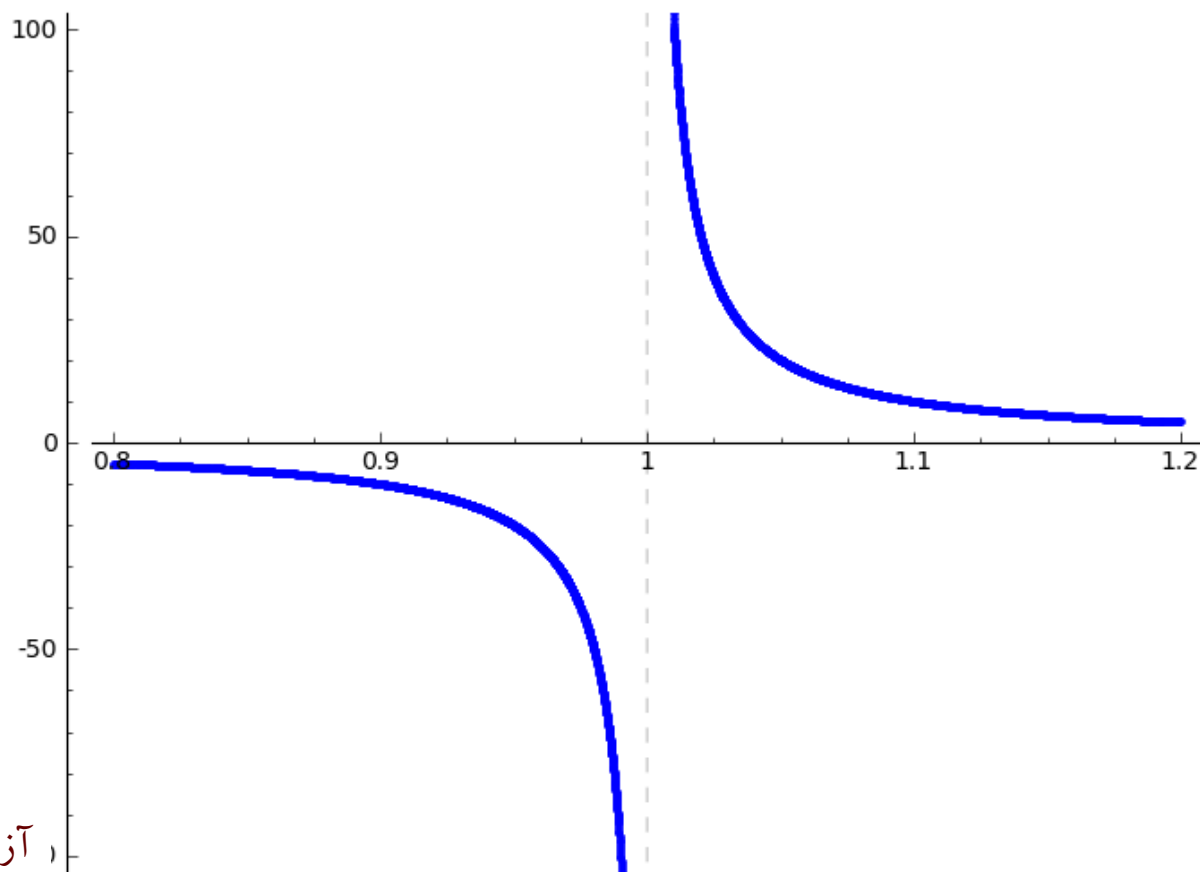
اختیارات در رسم نمودار

اختیارات

حداکثر عمق زمانیکه تابع به شدت تغییر می کند	<code>adaptive_recursion</code>	<input type="checkbox"/>
تغییراتی که باعث توقف بازگشت می شود	<code>adaptive_tolerance</code>	<input type="checkbox"/>
تشخیص جاهایی که تابع بی نهایت می شود	<code>detect_poles</code>	<input type="checkbox"/>
لیست نقاطی که از نمودار جا افتاده اند	<code>exclude</code>	<input type="checkbox"/>
تعداد نقاطی که در نمودار به کار گرفته شده اند	<code>plot_points</code>	<input type="checkbox"/>

مثالی برای حالت بی نهایت شدن تابع

```
pole_plot = plot(1 / (x - 1), (0.8, 1.2), detect_poles='show', marker='.')  
print("min y = {0} max y = {1}".format(pole_plot.ymax(), pole_plot.ymin()))  
pole_plot.ymax(100.0)  
pole_plot.ymin(-100.0)  
pole_plot.show()
```



چند نمودار در یک شکل

□ می توان با یک جمع ساده نمودارها را در کنار یکدیگر قرار داد.

```
p1 = plot(sin, (-2*pi, 2*pi), thickness=2.0, rgbcolor=(0.5, 1, 0))  
legend_label='sin(x)'
```

```
p2 = plot(cos, (-2*pi, 2*pi), thickness=3.0, color='purple',  
alpha=0.5, legend_label='cos(x)')
```

```
plt = p1 + p2
```

```
plt.axes_labels(['x', 'f(x)'])
```

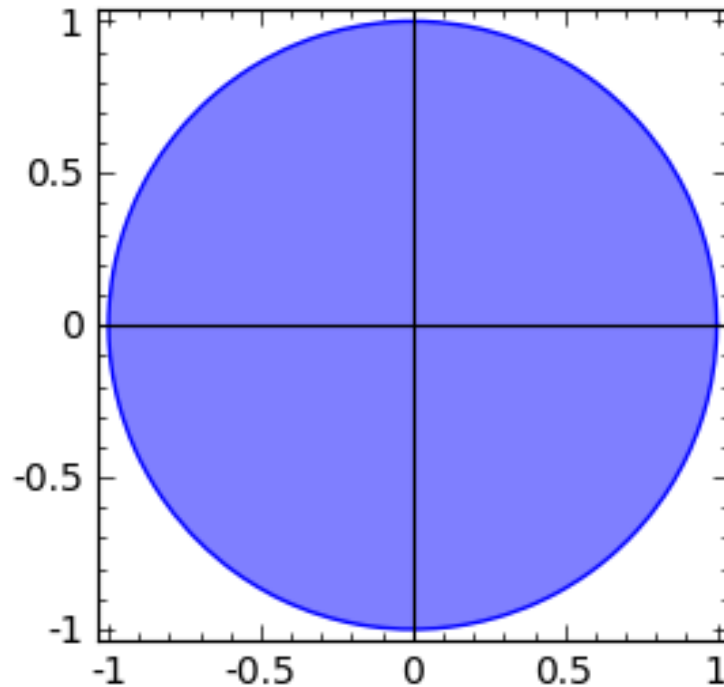
```
show(plt)
```


رسم توابع پارامتری

var('t')

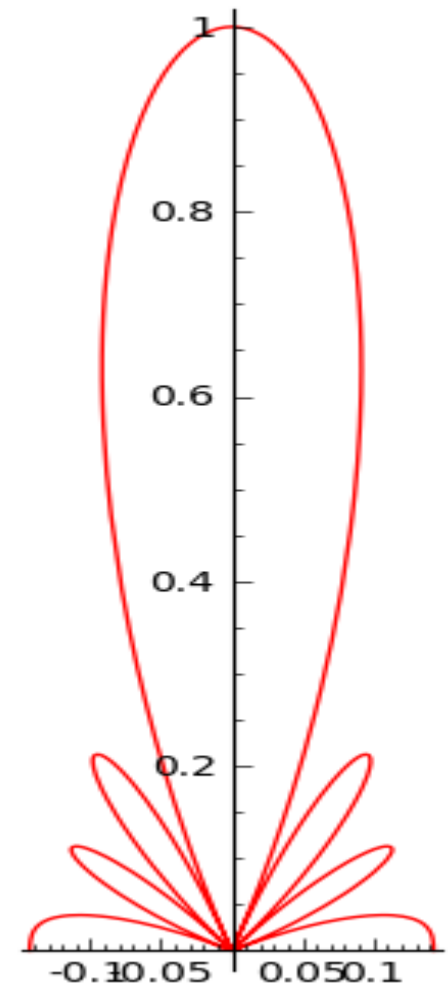
```
pp = parametric_plot((cos(t), sin(t)), (t, 0, 2*pi), fill=True, fillcolor='red')
```

```
pp.show(aspect_ratio=1, figsize=(3, 3), frame=True)
```



رسم توابع قطبی

```
var('r, theta')
N=7
array_factor = 1 / N * sin(N * pi / 2 * cos(theta)) \
/ sin(pi / 2 * cos(theta))
array_plot = polar_plot(abs(array_factor), (theta, 0, pi),
color='red',)
show(array_plot)
```



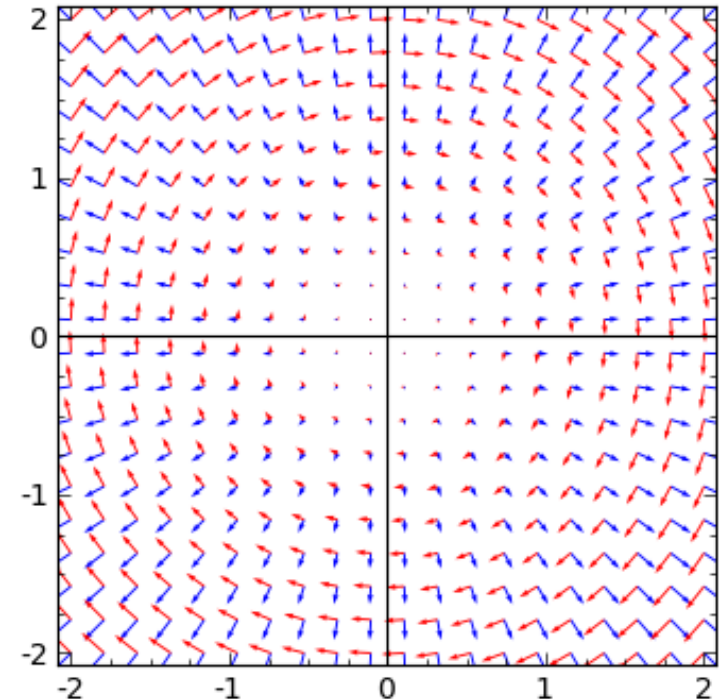
رسم میدان های برداری

```
var('x, y')
```

```
a = plot_vector_field((x, y), (x, -2, 2), (y, -2, 2), color='blue')
```

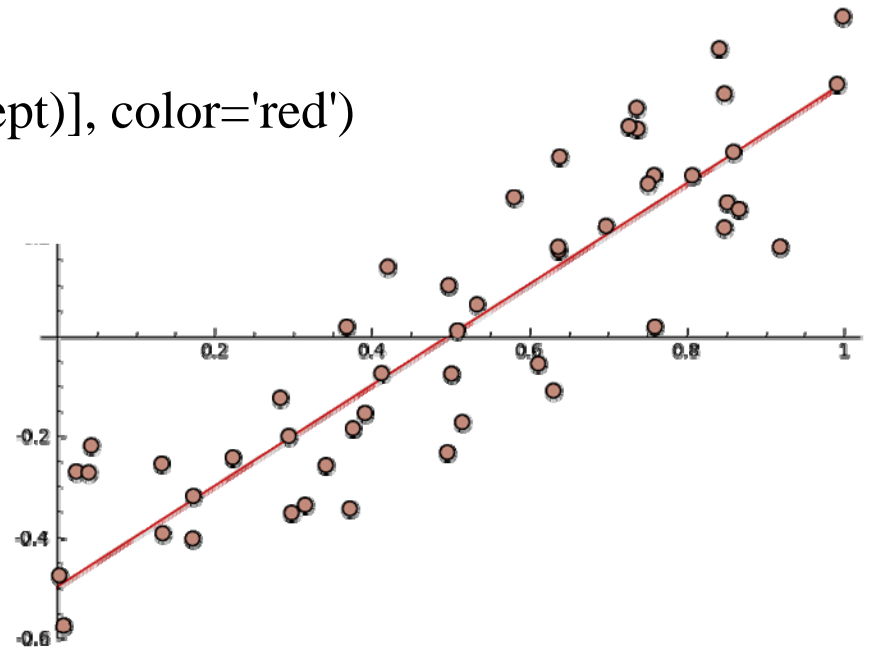
```
b = plot_vector_field((y, -x), (x, -2, 2), (y, -2, 2), color='red')
```

```
show(a + b, aspect_ratio=1, figsize=(4, 4))
```



رسم نمودارهای اسکاتر

```
def noisy_line(m, b, x):  
    return m * x + b + 0.5 * (random() - 0.5)  
slope = 1.0  
intercept = -0.5  
x_coords = [random() for t in range(50)]  
y_coords = [noisy_line(slope, intercept, x) for x in x_coords]  
sp = scatter_plot(zip(x_coords, y_coords))  
sp += line([(0.0, intercept), (1.0, slope+intercept)], color='red')  
sp.show()
```



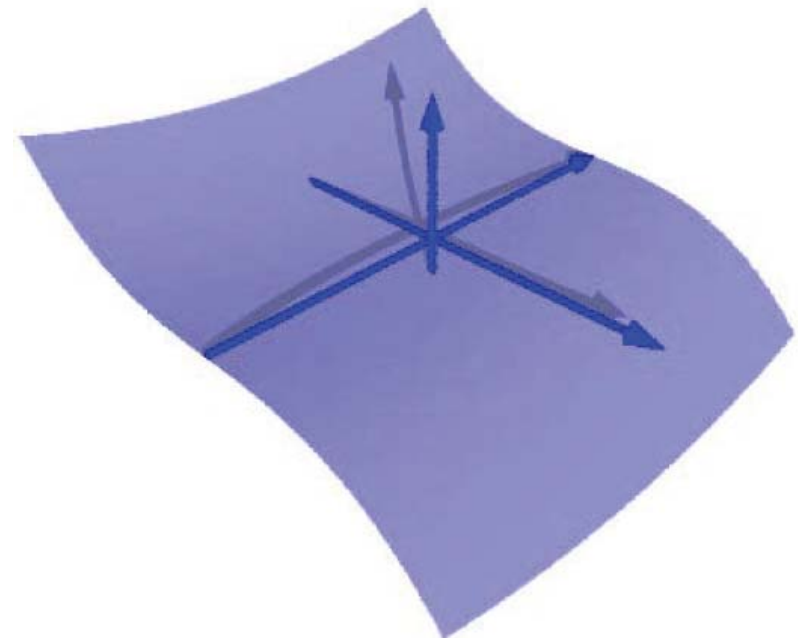
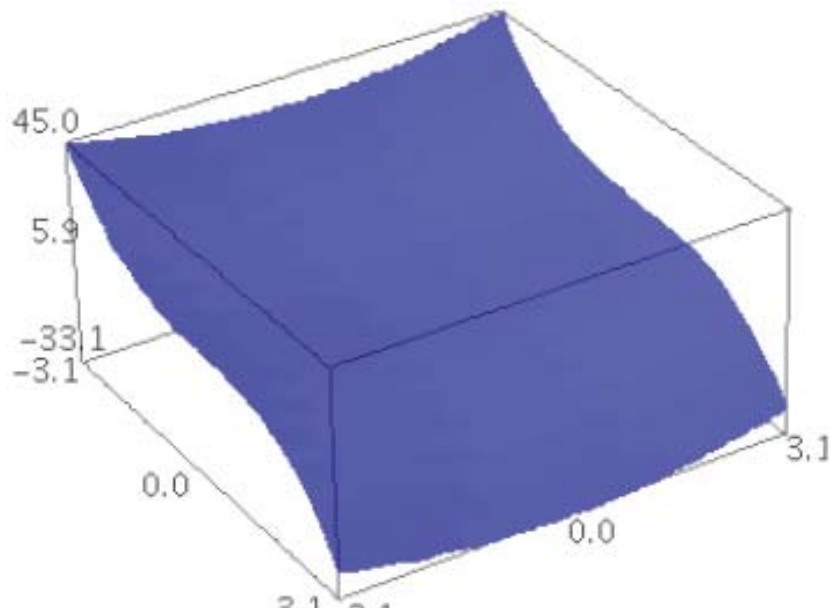
رسم نمودارهای سه بعدی

```
var('x, y')
```

```
p3d = plot3d(y^2 + 1 - x^3 - x, (x, -pi, pi), (y, -pi, pi))
```

```
p3d.show()
```

```
p3d.show(viewer='tachyon', frame=False, axes=True)
```



رسم نمودارهای سه بعدی پارامتری

```
var('u, v')
```

```
f1 = (4 + (3 + cos(v)) * sin(u), 4 + (3 + cos(v)) * cos(u), 4 + sin(v))
```

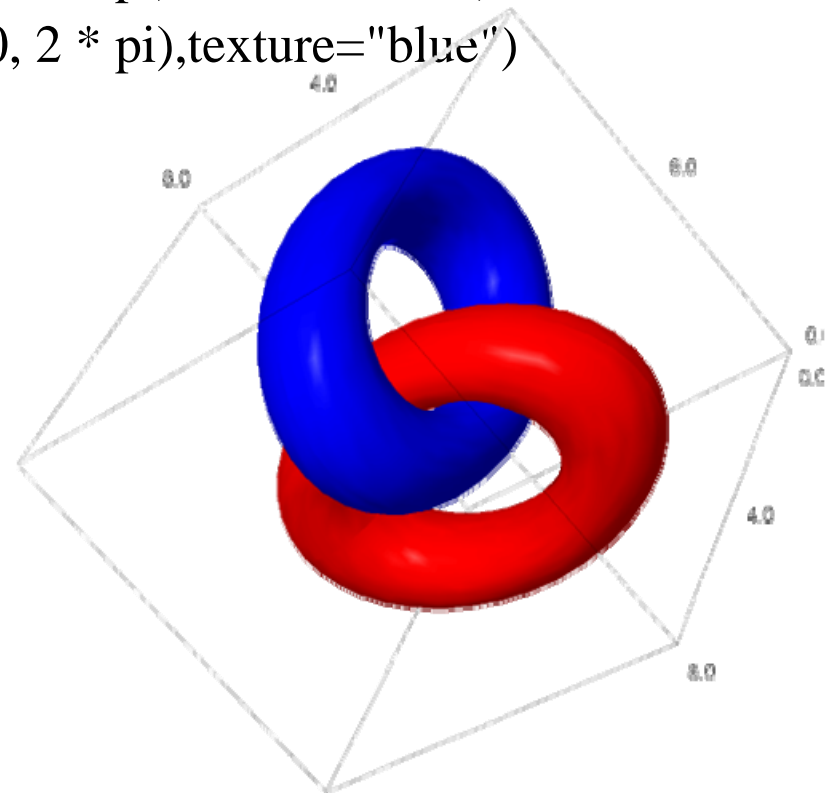
```
f2 = (8 + (3 + cos(v)) * cos(u), 3 + sin(v), 4 + (3 + cos(v)) * sin(u))
```

```
p1 = parametric_plot3d(f1, (u, 0, 2 * pi), (v, 0, 2 * pi), texture="red")
```

```
p2 = parametric_plot3d(f2, (u, 0, 2 * pi), (v, 0, 2 * pi), texture="blue")
```

```
combination = p1 + p2
```

```
combination.show()
```



توابع رسم نمودار

- `parametric_plot()`
- `implicit_plot` یک تابع دومتغیره $f(x,y)$ را می گیرد و منحنی $f(x,y)=0$ را رسم می کند
- `Var(x,y)`
- `f(x,y) = x^2 + y^2 - 2`
- `implicit_plot(f, (-3, 3), (-3, 3), fill=True)`
- `polar_plot()`
- `region_plot()`
- `list_plot()`
- `scatter_plot()`
- `bar_chart()`
- `contour_plot()`
- `density_plot()` یک تابع دومتغیره $f(x,y)$ را می گیرد و ارتفاع $f(x,y)$ را رسم می کند (تیره و روشن).
- `x,y = var('x,y')`
- `density_plot(sin(x)*sin(y), (x, -2, 2), (y, -2, 2))`
- `plot_vector field()`
- `plot slope field()`
- `matrix_plot()` نقاط را می گیرد و برای یک سری بردار به صورت پیکسلی ماتریس را نمایش می دهد
- `matrix_plot([[1, 0], [0, 1]], fontsize=10)`
- `complex_plot()` رسم نمودار یک تابع یک متغیره با ورودی اعداد مختلط $f(z)$
- `complex_plot(sin(x), (-5, 5), (-5, 5))`
- `graphics_array()` چند نمودار را در یک لیست می گیرد و همه را در یک صفحه رسم می کند
- `p1 = plot(sin, (-4, 4))`
- `p2 = plot(cos, (-4, 4))`
- `g = graphics_array([p1, p2])`

توابع رسم

- `circle((1,1), 1) + plot(x^2, (x,0,5))` arrow() یک بردار از مینیمم به ماکسیمم
- circle() رسم دایره با شعاع دلخواه
- ellipse() رسم بیضی با شعاع و زاویه دلخواه
- arc() یک کمان از یک دایره یا بیضی
- disk() رسم یک دیسک یا یک نوار
- line() یک خط با نقاط مشخص شده (نیازی نیست خط راست باشد)
- `L = [[cos(pi*i/100)^3, sin(pi*i/100)] for i in range(200)]` point() یک نقطه
- `line(L, rgbcolor=(1/4,1/8,3/4))` text() یک متن
- `text('A Bulb', (1.5, 0.25))` polygon() یک چند ضلعی
- `polygon([[1,2], [5,6], [5,0]], color='red')`
- `polygon([(-x,y) for x,y in P[0]], color='blue')`

توابع خاص

Function	Sage	Function	Sage
sine	$\sin(x)$	square root	$\text{sqrt}(x)$
cosine	$\cos(x)$	ex	$\text{exp}(x)$
tangent	$\tan(x)$	natural logarithm	$\log(x)$
arcsine	$\arcsin(x)$	absolute value	$\text{abs}(x)$
arccosine	$\arccos(x)$	complex conjugate	$\text{conjugate}(x)$
arctangent	$\arctan(x)$		

کار کردن با معادلات

```
forget()
```

```
expr = -20 * x - 30 <= 4 * x^3 - 7 * x
```

```
print("Expression {0} is {1}".format(expr, bool(expr)))
```

```
p1 = plot(expr.left(), (x, -5, 5), legend_label='lhs')
```

```
p2 = plot(expr.right(),(x, -5, 5), color='red', legend_label='rhs')
```

```
assume(x > 0)
```

```
print("Now assume x > 0")
```

```
print("Expression {0} is {1}".format(expr, bool(expr)))
```

```
show(p1 + p2)
```

اعمال محاسباتی با معادلات

```
var('x, y')
expr = (y - 7) / (x^2 + 1) == x^3 - 5
print("Expression:")
expr.show()
print("Two ways of multiplying both sides:")
show(expr.multiply_both_sides(x^2 + 1))
show(expr * (x^2 + 1))
# also divide_both_sides
expr = expr * (x^2 + 1)
print("Two ways of adding to both sides:")
show(expr.add_to_both_sides(7))
show(expr+7)
```

کار کردن با معادلات کسری

```
var('s')
F(s) = (s + 1) / (s^2 * (s + 2)^3)
show(F)
print("Numerator: ")
show(F.numerator())
print("Denominator: ")
show(F.denominator())
print("Expanded:")
show(F.expand_rational())
print("Partial fraction expansion:")
pf(s) = F.partial_fraction()
show(pf)
```

کار کردن با معادلات (تغییر متغیر)

```
var('x, y')
f(x) = 1 / x + 3 * x^2 + cos(x)
f.show()
print("Substitute for x with a keyword:")
show(f.subs(x=(7 * x)))
print("Substitute for x with a relational expression:")
show(f.substitute(x == 7 * x))
print("Substitute sine for cosine:")
show(f.substitute_function(cos, sin))
print("Substitute using a dictionary:")
show(f.substitute({1 / x: y^3, cos(x):sin(x)}))
```

کار کردن با اتحادها

```
var('x')
```

```
exp1 = (x + 3)^3 == (x - 1)^2
```

```
show(exp1)
```

```
print("Expanded expression:")
```

```
exp2 = exp1.expand()
```

```
show(exp2)
```

```
show(exp1.expand('left'))
```

ساده کردن معادلات مثلثاتی

```
var('x, y')
f(x, y) = sin(x) * cos(x)^3 + sin(y)^2
print("f(x, y)")
f.show()
g(x, y) = f.reduce_trig()
print("After trig_reduce:")
g.show()
print("After expanding:")
show(g.expand_trig())
print("Simplify to get original expression:")
show(g.expand_trig().trig_simplify())
```

ساده کردن معادلات

```
var('x')
```

```
#----- Logs
```

```
f(x) = log(x^2 * sin(x) / sqrt(1 + x))
```

```
print("Original function:")
```

```
f.show()
```

```
print("This form is easier to work with:")
```

```
show(f.expand_log())
```

```
print("Simplify expanded form:")
```

```
show(f.expand_log().simplify_log())
```

```
#-----Rational functions
```

```
f(x) = (x + 1) / (x^2 + x)
```

```
print("Original function:")
```

```
f.show()
```

```
print("Simplified:")
```

```
show(f.simplify_rational())
```

```
#----- Radicals
```

```
f(x) = sqrt(x^2+x)/sqrt(x)
```

```
print("Original function:")
```

```
f.show()
```

```
print("Simplified:")
```

```
show(f.simplify_radical())
```


ساده سازی عبارات

□ `expand_log & log_expand`

باز کردن لگاریتم

□ `simplify_log & log_simplify`

تلاش برای ساده سازی عبارات لگاریتم دار

□ `simplify_rational & rational_simplify`

تلاش برای ساده سازی عبارات کسردار

□ `radical_simplify simplify_radical exp_simplify`

تلاش برای ساده سازی عبارات رادیکالی

□ `simplify_exp`

تلاش برای ساده سازی عبارات تواندار

□ `simplify_factorial & factorial_simplify`

تلاش برای ساده سازی عبارات فاکتوریل دار

□ `simplify_full & full_simplify`

تلاش برای انجام تمام ساده سازی های بالا

محاسبه ریشه

```
var('x')
```

```
f(x) = x^4
```

```
fr=f.roots(x, multiplicities='false')
```

```
print(fr)
```

	Description	Default	Keyword
	اگر false باشد ریشه های غیر دقیق نیز شامل می شود	True	explicit_solutions
	اگر false باشد لیست ریشه های بدون تکرار را نشان می دهد	True	multiplicities
	اگر حلقه مشخص شود روی آن حلقه ریشه ها محاسبه می شود.	None	ring

محاسبه حد

```
var('x')
```

```
f(x) = 1 / x
```

```
print("Limit of 1/x as x->0+: {0}".format(limit(f, x=0, dir='plus')))
```

```
print("Limit of 1/x as x->0-: {0}".format(limit(f, x=0, dir='minus')))
```

Limit of $1/x$ as $x \rightarrow 0^+$: $x \rightarrow +\text{Infinity}$

Limit of $1/x$ as $x \rightarrow 0^-$: $x \rightarrow -\text{Infinity}$

محاسبه انتگرال

```
var('x')
print("Elementary integrals:")
f = x^2
```

```
print(f.integrate(x))
```

```
print(integral(e^x,x))
```

```
print(integral(1/x,x))
```

```
print(integral(sinh(x), x))
```

```
print(integral(1/sqrt(1+x^2),x))
```

```
print("\nIntegration by parts:")
```

```
print(integral(e^x*cos(x), x))
```

```
print(integral(sqrt(x^2-25)/x, x))
```

```
print("\nDefinite integral:")
```

```
print(integral(1/(1+x^2), x, -1, 1))
```

```
print("\nImproper integral:")
```

```
print(integral(1/(1+x^2), x, -infinity, infinity))
```

```
print("\nDivergent integral:")
```

```
print(integral(1/(1-x), x, 1,2))
```

محاسبه مشتق

```
var('x, y')
f(x, y) = 3 * x^4 * y^3 + 9 * y * x^2 - 4 * x + 8 * y
print("f(x,y):")
f.show()
dfdx(x, y) = diff(f, x)
print("df/dx:")
dfdx.show()
dfdy(x, y) = diff(f, y)
print("df/dy:")
dfdy.show()
print("Second derivative:")
d2fdx2(x, y) = derivative(f, x, 2)
d2fdx2.show()
```

محاسبه سری ها

```
var('x, n, k')
```

```
f(x) = sin(x) / x^2
```

```
f.show()
```

```
print("Power series expansion around x=1:")
```

```
s(x) = f.series(x==1, 3)
```

```
s.show()
```

```
print("Sum of alternating harmonic series:")
```

```
h(k) = (-1)^(k + 1) * 1 / k
```

```
print h.sum(k, 1, infinity)
```

```
print("Sum of binomial series:")
```

```
h(k) = binomial(n, k)
```

```
print h.sum(k, 1, infinity)
```

```
print("Sum of harmonic series:")
```

```
h(k) = 1 / k
```

```
print h.sum(k, 1, infinity)
```

سری تیلور

```
var('x')
```

```
x1 = 0
```

```
f(x) = sin(x)
```

```
g=f.taylor(x, x1, 7)
```

```
show(g)
```

```
var('x,y')
```

```
f(x,y) = sin(x)*y
```

```
g=f.taylor(x,0, 7)
```

```
show(g)
```

تبدیلات لاپلاس

```
var('t, a, k, s')  
print("Elementary transform:")  
f(t) = sin(k * t)  
F(s) = f.laplace(t, s)  
F.show()
```

```
-----  
print("Inverse transform:")  
G(s) = 1 / ((s - 1) * (s + 2) * (s + 4))  
G.show()  
g(t) = G.inverse_laplace(s, t)  
g.show()
```


تمرینات

□ (تحویلی) (تحویلی) برنامه ای ۲۰ تمرین را از کتاب ریاضی
عمومی دکتر شهشهبانی با sage حل کنید (هر فصل یک
تمرین)

پروژه