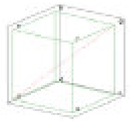


ch10 sec10.1 泰勒多项式

Table of Contents


```
begin
        using PlutoUI   , Plots   ,DataFrames   ,HypertextLiteral   ,LaTeXStrings
,Symbolics
        gr()
        theme(:bright)
         PlutoUI.TableOfContents()

    end
```

线性近似

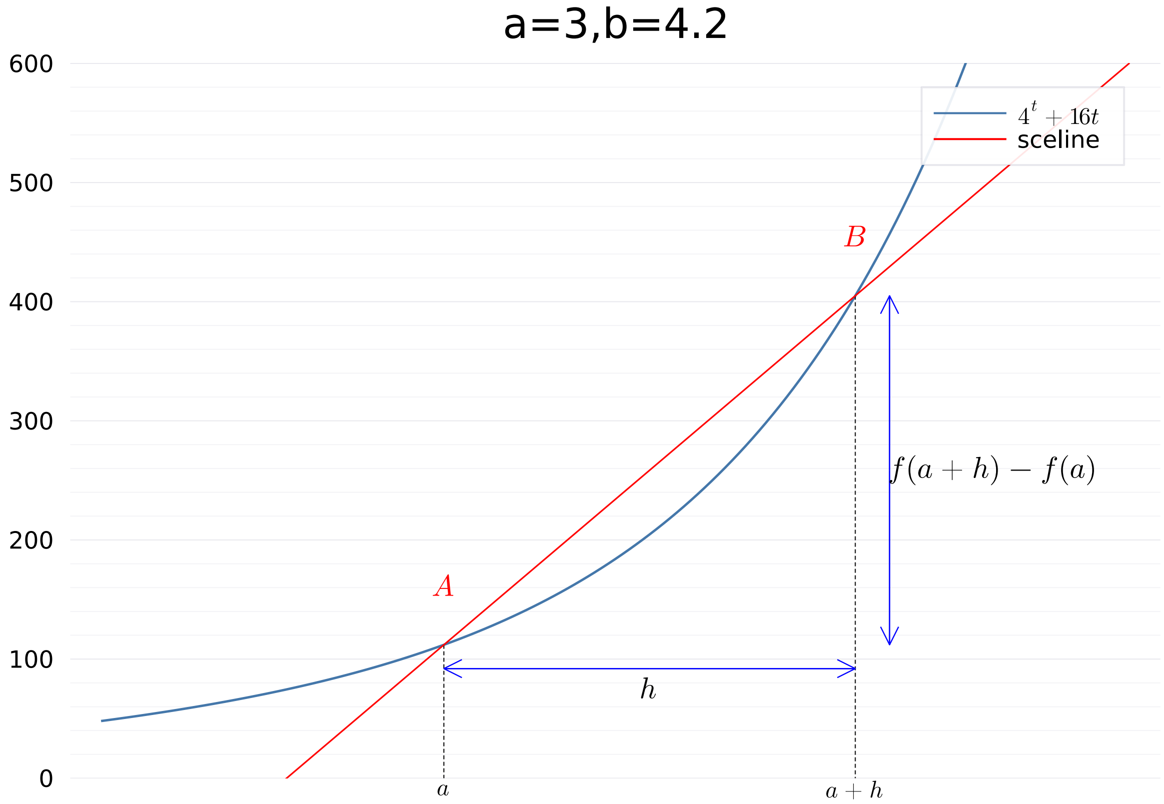
前面在学习时已经多次研究用某点出的切线来近似曲线的变化. 切线是该处的一阶导数, 所以切线方程可以看做函数在该点附近的一阶近似

$$f(x) \approx f(a)+f'(a)(x-a)$$

```
md"""
## 线性近似

前面在学习时已经多次研究用某点出的切线来近似曲线的变化.  切线是该处的一阶导数, 所以切线方程可以看做函数在该点附近的一阶近似

$f(x) \approx f(a)+f'(a)(x-a)$


"""
```

4.0

```
@bind deltah Slider(3.2:0.2:4.2, default=4.0 ,show_value=true)
```




```
let

    a=3        # 值域中的 a
    b= deltah     # 值域中的 b 由上面Slider 的 deltah 控制
    h=b-a      # 差值
    f(t)=4^t+16t    # 函数
    tspan=2:0.02:5
    m=(f(b)-f(a))/h   #仿射割线的斜率
    fl(x)=m*x-m*a+f(a)   #经过 AB两点的仿射直线方程
    http://ambrnet.com/MathCalc/Line/Line_.htm
    yoffset=50    #用于注释的 y轴偏移距离
    pA=[a f(a)]    #曲线上 A 点的坐标
    pB=[b f(b)]    #曲线上 B 点的坐标
    pa=[a 0]       #  x轴上的投影坐标
    pb=[b 0]      #  y轴上的投影坐标
    ph=[a+0.5h f(a)-20]   # 用于标注 h的中间点坐标
    pf=[b+0.1 f(a)+(f(b)-f(a))/2]
    lAa=[pA;pa]    # 从 A到 a 的线段的 矩阵 下同
    lBb=[pB;pb]
    ldeltah=[pA;[a f(a)]]
    arrowhA=[ph;[a f(a)-20]]
    arrowhB=[ph;[b f(a)-20]]
    arrowfA=[pf;[b+0.1 f(a)]]
    arrowfB=[pf;[b+0.1  f(b)]]
    ann=[(a,f(a)+yoffset,text(L"A",color=:red,pointsize=10)),
          (b,f(b)+yoffset,text(L"B",color=:red,pointsize=10)),
          (a,-10,text(L"a",pointsize=8)),
           (b,-10,text(L"a+h",pointsize=8)),
           (ph[:,1],ph[:,2],text(L"h",pointsize=10,valign=:top)),
           (pf[:,1],pf[:,2],text(L"f(a+h)-f(a)",pointsize=10, halign=:left))

    ]
```

```
    plot(f,tspan,label=L"4^t+16t",ylims=
    (0,600),ann=ann,xticks=:none,title="a=$(a),b=$(deltah)")
    plot!(lAa[:,1],lAa[:,2], ls=:dash,color=:black,lw=0.5,label=false)
    plot!(lBb[:,1],lBb[:,2], ls=:dash,color=:black,lw=0.5,label=false)
    plot!(ldeltah[:,1],ldeltah[:,2], ls=:dash,color=:black,lw=0.5,label=false)
    plot!(arrowhA[:,1],arrowhA[:,2],arrow=0.1,color=:blue,lw=0.6,label=false)
    plot!(arrowhB[:,1],arrowhB[:,2],arrow=0.1,color=:blue,lw=0.6,label=false)
    plot!(arrowfA[:,1],arrowfA[:,2],arrow=0.1,color=:blue,lw=0.6,label=false)
    plot!(arrowfB[:,1],arrowfB[:,2],arrow=0.1,color=:blue,lw=0.6,label=false)
    plot!(fl,tspan,color=:red,label="sceline",lw=0.8)


end
```

首先把焦点放在 $a=0$处, 当$x=o$,是切线在该点对曲线的近似成为一阶近似, 又叫做 一阶麦克劳林多项式,写作:

$$f(x)=P_1(x)=f(0)+f'(0)x$$

Example

example 1

函数为 $g(x)=cosx$,x 表示为弧度, 求$x=0$附近的函数一阶近似表达式

余弦函数在$x=0$ 处的导函数为$sinx$, 所以当$x=0$,$sin(0)=0$, 着$x=0$处的切线近似方程为:

$$P_1(x)=cos(x)+sin(x)*x=1$$

再看看当取不同x时,函数的真实值

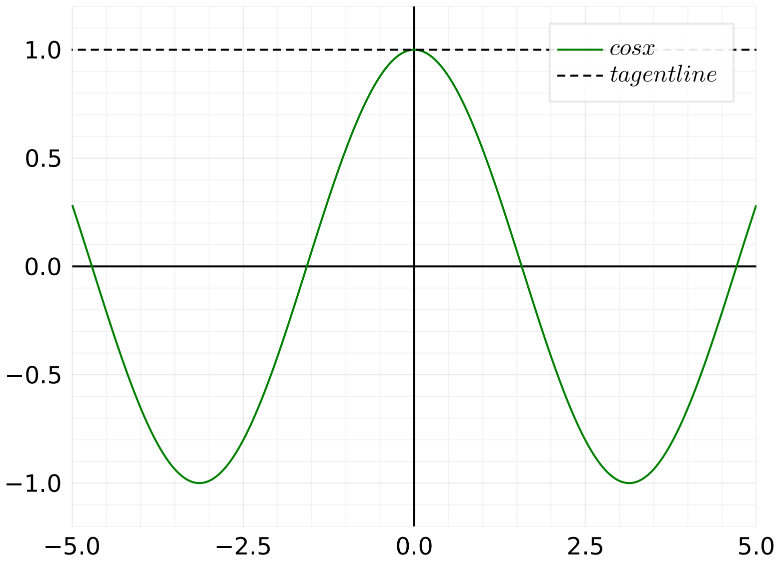
	x	y
1	-0.4	0.921061
2	-0.3	0.955336
3	-0.2	0.980067
4	-0.1	0.995004
5	0.0	1.0
6	0.1	0.995004
7	0.2	0.980067
8	0.3	0.955336
9	0.4	0.921061

当$x \to 0$时, $P_1(x) = 1$的值与函数真实值越接近. 当x远离0的时候, 近似效果就比较差了.

所以需要捕获更多的变化, 引入更高的项可以达到这个目的

```
md"""

首先把焦点放在 $a=0$处, 当$x=o$,是切线在该点对曲线的近似成为一阶近似, 又叫做 一阶麦克劳林多项式,写作:

$f(x)=P_1(x)=f(0)+f'(0)x$


!!! example
    example 1

    函数为 $g(x)=cosx$,$x$ 表示为弧度, 求$x=0$附近的函数一阶近似表达式


余弦函数在$x=0$ 处的导函数为$sinx$, 所以当$x=0$,$sin(0)=0$, 着$x=0$处的切线近似方程为:
```

```
$P_1(x)= cos(x)+sin(x)*x=1$

再看看当取不同$x$时,函数的真实值

![](https://tva1.sinaimg.cn/mw690/e6c9d24egy1h38yiq1z7ij20c40iujrw.jpg)

当$x \to 0$ 时， $P_1(x)=1$ 的值与函数真实值越接近. 当$x$ 远离$0$ 的时候, 近似效果就比较差了.

所以需要捕获更多的变化, 引入更高的项可以达到这个目的

"""
```




```julia
let
    tspan=-5:0.02:5
    plot(cos,tspan ,label=L"cosx", lw=1,color=:green,xlims=(-5,5),ylims=
    (-1.2,1.2),frame=:zerolines,size=(400,300))
    hline!([1], lw=1,color=:black, ls=:dash,label=L"tagentline")

end
```

	x	y
1	-0.4	0.921061
2	-0.3	0.955336
3	-0.2	0.980067
4	-0.1	0.995004
5	0.0	1.0
6	0.1	0.995004
7	0.2	0.980067
8	0.3	0.955336
9	0.4	0.921061

```
let
    xspan=-0.4:0.1:0.4
    ys=[cos(x) for x in xspan]
    df=DataFrame(;x=xspan, y=ys)
end
```

二次多项式近似

> Example
>
> example 2
>
> 函数为 $g(x) = cosx$,x 表示为弧度, 求$x = 0$附近的函数二阶表达式

要完美的近似一个函数, 需要 值, 和一阶,二阶导数都近似相等即:

$$P(0) = g(0)\ P'(0) = g'(0)P''(0) = g''(0)$$

假设要构造的二次多项式为:

$$P_2(x) = C_0 + C_1x + C_2x^2$$

所以对构造多项式和$g(x)$ 分别求导数:

$$P_2(x) = C_0 + C_1x + C_2x^2\ , g(x) = cosx$$

$$P_2'(x) = C_1 + 2C_2x\ , g'(x) = -sinx$$

$$P_2''(x) = 2C_2\ , g''(x) = -cosx$$

当$x = 0$ 时, 带入各式化简得:

$$C_0 = P_2(0) = g(0) = cos(0) = 1\ , 所以C_0 = 1$$

$$C_1 = P_2'(0) = g'(0) = -sin(0) = 0\ , 所以C_1 = 0$$

$$2C_2 = P_2''(0) = g''(0) = -cos(0) = -1\ , 所以C_0 = -\frac{1}{2}$$

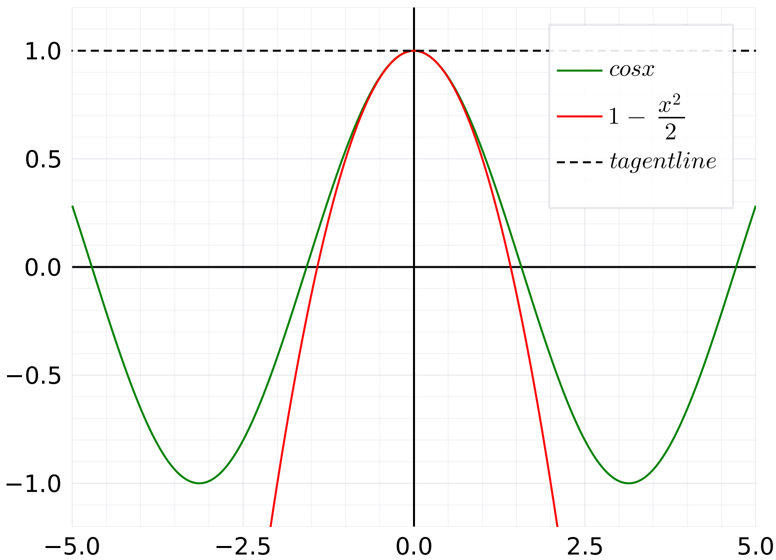
由此得到 $g(x) = cosx$ 在$x = 0$ 处的二阶近似为:

$$cosx \approx P_2(x) = 1 + 0 \cdot x - \frac{1}{2}x^2 = 1 - \frac{x^2}{2}$$

图形如下:

```
md"""

## 二次多项式近似

!!! example
    example 2

    函数为 $g(x)=cosx$,$x$ 表示为弧度, 求$x=0$附近的函数二阶表达式

要完美的近似一个函数, 需要 值, 和一阶,二阶导数都近似相等即:

$P(0)=g(0) \  P'(0)=g'(0) P''(0)=g''(0)$
```

```
假设要构造的二次多项式为:

$P_2(x)=C_0+C_1x+C_2x^2$

所以对构造多项式和$g(x)$ 分别求导数:

$P_2(x)=C_0+C_1x+C_2x^2 \ ,  g(x)=cosx$
$P'_2(x)=C_1+2C_2x \ , g'(x)=-sinx$
$P''_2(x)=2C_2 \  , g''(x)=-cosx$

当$x=0$ 时, 带入各式化简得:

$C_0=P_2(0)=g(0)=cos(0)=1 \ , 所以  C_0=1$
$C_1=P_2'(0)=g'(0)=-sin(0)=0 \ , 所以  C_1=0$
$2C_2=P_2''(0)=g''(0)=-cos(0)=-1 \ , 所以  C_0=-\frac{1}{2}$


由此得到 $g(x)=cosx$ 在$x=0$ 处的二阶近似为:

$cosx \approx P_2(x)=1+0\cdot x-\frac{1}{2}x^2=1-\frac{x^2}{2}$

图形如下:

"""
```




```
let
    tspan=-5:0.02:5
    p2(x)=1-x^2/2
    plot([cos,p2],tspan ,label=[L"cosx" L"1-\frac{x^2}{2}"], lw=1,color=[:green
    :red],xlims=(-5,5),ylims=(-1.2,1.2),frame=:zerolines,size=(400,300))

    hline!([1], lw=1,color=:black, ls=:dash,label=L"tagentline")

end
```

归纳上面实例里的计算可以得到:

Definition

当函数为$f(x)$, 在$x=0$附近函数的泰勒二阶多项式可以表示为:

$$f(x) \approx P_2(x)=f(0)+f'(0)+\frac{f''(0)}{2}x^2$$

更高阶多项式

尽管二阶多项式从图形看比一阶多项式更接近于原来函数, 当 x 远离0时, 弯曲程度还是和原来函数有差别. 可以用更高阶的导数来减小差别.

$$f(x) \approx P_n(x)=C_0+C_1x+C_2x^2+...+C_{n-1}+C_nx^n$$

总结来说, 多项式的阶数越高,越能在更大的区间内近似函数.

分别对多项式和函数求多阶导数, 经过化简可以求出各项系数, 得到一般通项式:

Definition

当函数为$f(x)$, 在$x=0$附近函数的泰勒n阶多项式可以表示为:

$$f(x) \approx P_n(x)=f(0)+f'(0)x+\frac{f''(0)}{2!}x^2+\frac{f'''(0)}{3!}x^3+...+\frac{f^{(n)}()}{n!}x^n$$

$P_n(x)$就定义为函数$f(x)$在$x=0$附近的n阶近似

```
md"""
归纳上面实例里的计算可以得到:

!!! definition

    当函数为$f(x),在x=0$附近函数的泰勒二阶多项式可以表示为:

    $f(x) \approx P_2(x)=f(0)+f'(0)+\frac{f''(0)}{2}x^2$


### 更高阶多项式

尽管二阶多项式从图形看比一阶多项式更接近于原来函数, 当 $x$ 远离$0$时, 弯曲程度还是和原来函数有差
别.
可以用更高阶的导数来减小差别.


$f(x) \approx P_n(x)=C_0+C_1x+C_2x^2+...+C_{n-1}+C_nx^n$

总结来说, 多项式的阶数越高,越能在更大的区间内近似函数.

分别对多项式和函数求多阶导数, 经过化简可以求出各项系数, 得到一般通项式:

!!! definition
```

```
    当函数为$f(x),在x=0$附近函数的泰勒$n$阶多项式可以表示为:

    $f(x) \approx P_n(x)=f(0)+f'(0)x+\frac{f''(0)}{2!}x^2+\frac{f'''(0)}{3!}x^3+...+\frac{f^{(n)}()}{n!}x^n$

   $P_n(x) 就定义为函数 f(x) 在 x=0 附近的 n 阶 近似$

"""
```

Example

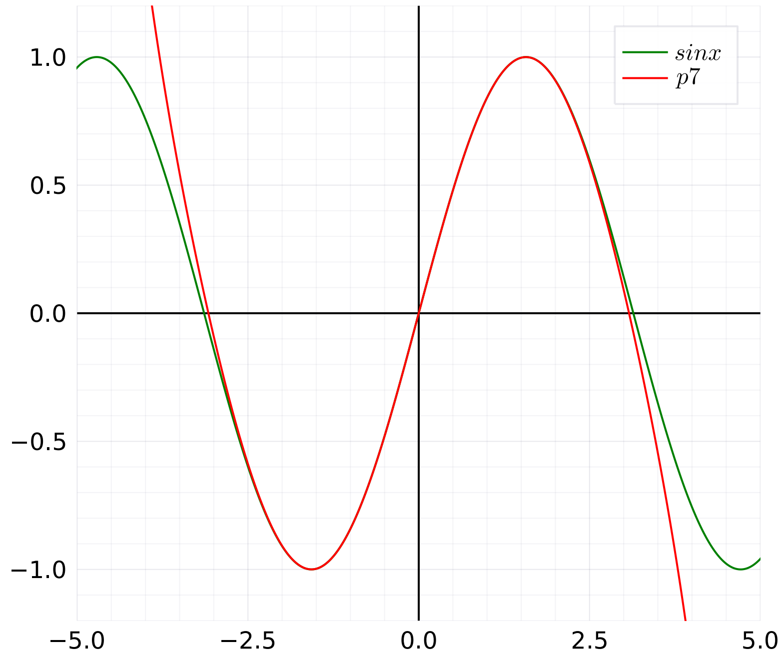
example 3

函数为 $g(x)=sinx$,x 表示为弧度，求$x=0$附近的函数7阶阶表达式

使用定义的公式可以得到：

$$sinx=P_7(x)=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\frac{x^7}{7!}$$

```
md"""
!!! example
    example 3

    函数为 $g(x)=sinx$,$x$ 表示为弧度，求$x=0$附近的函数7阶阶表达式



使用定义的公式可以得到 :

$sinx=P_7(x)=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\frac{x^7}{7!}$
"""
```




```
let
    tspan=-6:0.02:6
    f(x)=sin(x)
    p7(x)=x-x^3/(factorial(3))+x^5/(factorial(5))-x^7/(factorial(7))
    plot([f,p7],tspan ,label=[L"sinx" L"p7"], lw=1,color=[:green :red],xlims=
(-5,5),ylims=(-1.2,1.2),frame=:zerolines,size=(400,350))
end
```

Example

example 4

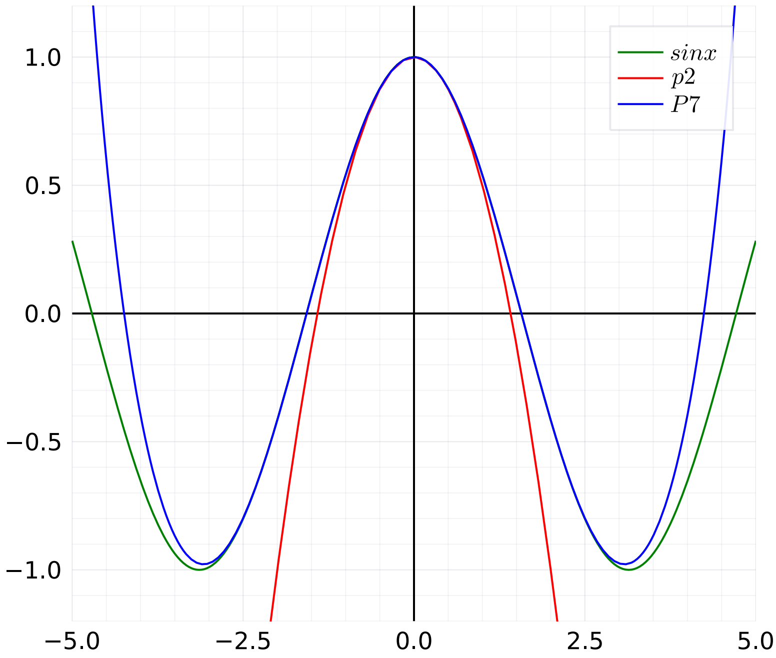
函数为 $g(x)=cosx$ 求$x=0$附近的函数8阶阶表达式

使用定义的公式可以得到：

$$cosx \approx P_8(x)=1-\frac{x^2}{2!}+\frac{x^4}{4!}-\frac{x^6}{6!}+\frac{x^8}{8!}$$

绘图如下，可见8阶多项式在更大范围内近似了$cosx$ 函数

```
md"""
!!! example
    example 4

    函数为 $g(x)=cosx$  求$x=0$附近的函数8阶阶表达式



使用定义的公式可以得到 ：

$cosx \approx P_8(x)=1-\frac{x^2}{2!}+\frac{x^4}{4!}-\frac{x^6}{6!}+\frac{x^8}{8!}$

绘图如下， 可见$8$阶多项式在更大范围内近似了$cosx$ 函数
"""
```




```
let
     tspan=-6:0.02:6
     f(x)=cos(x)
     p2(x)=1-(x^2/2)
     p8(x)=1-x^2/factorial(2)+x^4/factorial(4)-x^6/factorial(6)+x^8/factorial(8)
    plot([f, p2,p8],label=[L"sinx" L"p2" L"P7"], lw=1,color=[:green :red
    :blue],xlims=(-5,5),ylims=(-1.2,1.2),frame=:zerolines,size=(400,350))

end
```

Example

example 5

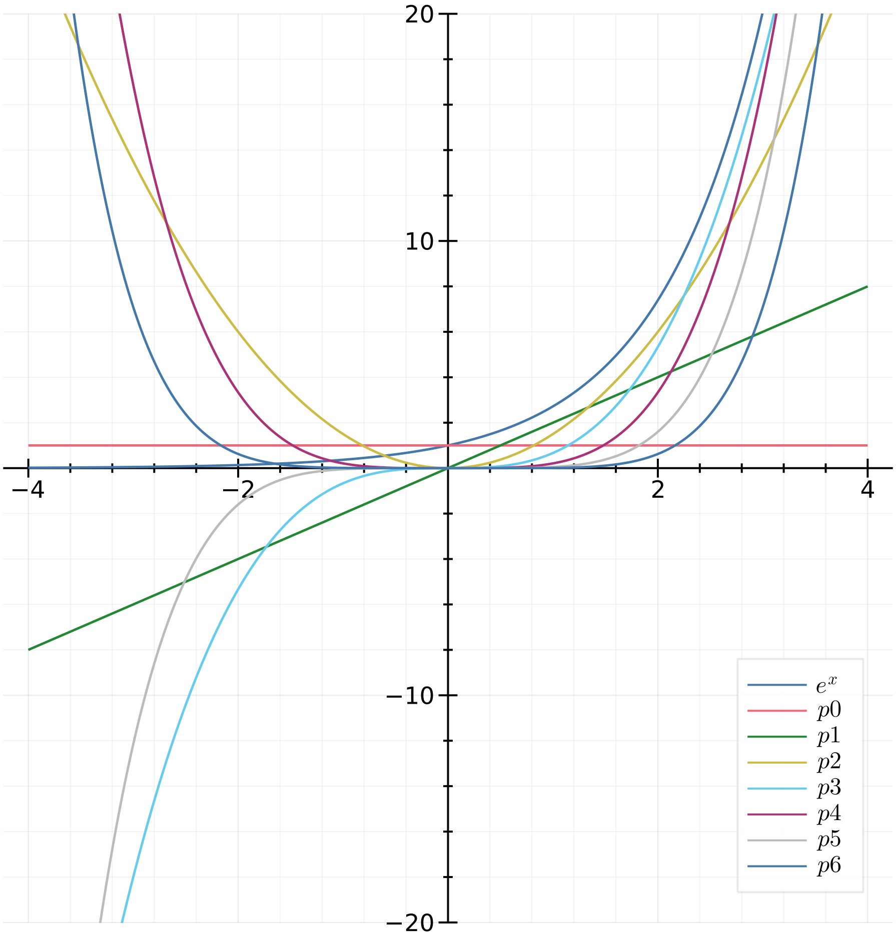
在$x=0$ 处构建函数 $f(x)=e^x$ 的10 阶多项式

我们有$f(0)=1$, 因为$f(x)$的导数为自己的表达式,$f'(x)=e^x$,$f''(x)=e^x$,由此归纳为: $f^{(k)}=e^x$,所以当$x=0$时, 所有的高阶导数$f^{(k)}=e^0=1$

带入通项式可得:

$$e^x \approx P_{10}(x)=1+x+\frac{x^2}{2!}+\frac{x^3}{3!}+...+\frac{x^{10}}{10!}$$

```
md"""
!!! example
    example 5

    在$x=0$ 处构建函数 $f(x)=e^x$ 的$10$ 阶多项式

我们有$f(0)=1$, 因为$f(x)$的导数为自己的表达式,$f'(x)=e^x$,$f''(x)=e^x$,由此归纳为:$f^{(k)}=e^x$ ,所以当$x=0$时, 所有的高阶导数$f^{(k)}=e^0=1$

带入通项式可得:

$e^x \approx P_{10}(x)=1+x+\frac{x^2}{2!}+\frac{x^3}{3!}+...+\frac{x^{10}}{10!}$
"""
```




```
let
    tspan=-4:0.02:4
    label=[]
    f(x)=e^x

    function pn(n)

         return function (x)
             res=0
             for i  in 0:n
              if n==0
                 res=1
              else
                 res=res+(x^n)/factorial(n)
              end
           end
          return res
        end

   end


funarr=[pn(n)  for n in 0:6]

label=[L"e^x" L"p0" L"p1" L"p2" L"p3" L"p4" L"p5" L"p6"]


plot([f, funarr...],tspan,label=label,ylims=(-20,20),size=
(500,500),legend=:bottomright,frame=:origin)

```

```
end
```

$x=a$ 附近的函数多项式近似

前面的近似都是在$x=0$附近的近似,有些函数在$x=0$处可能就根本没有定义, 所以需要更为一般的近似形式, 比如自然对数函数$ln(x)$,在$x0$ 没有定义. 下面看看这个如何解决

函数在$x=a$ 处的切线近似为:

$$y=f(a)+f'(a)(x-a)$$

这其实就是一阶近似, $f'(a)(x-a)$ 作为修正项部分捕获了当$x \to a$ 时, 函数值的变化量.

类似于在$x=0$ 点, 在$x=a$,可以用$f(a)$取值加修正项的方法获得. 可以构造多项式为:

$$f(x) \approx P_n(x)=C_0+C_1(x-a)+C_2(x-a)^2+...+C_n(x-a)^n$$

类似$x=0$ 处可以获得多项式:

> Definition
>
> **函数在$x=a$ 附近的多项式近似表达式表示为:**
>
> $$f(x) \approx f(a)+f'(a)(x-a)+\frac{f''(a)}{2!}(x-a)^2+...+\frac{f^{n}(a)}{n!}(x-a)^n$$

```
md"""
 ##  $x=a$ 附近的函数多项式近似

前面的近似都是在$x=0$附近的近似,有些函数在$x=0$处可能就根本没有定义, 所以需要更为一般的近似形式,比如自然对数函数$ln(x)$,在$x0$ 没有定义. 下面看看这个如何解决

函数在$x=a$ 处的切线近似为:

$y=f(a)+f'(a)(x-a)$

这其实就是一阶近似, $f'(a)(x-a)$ 作为修正项部分捕获了当$x \to a$ 时, 函数值的变化量.

类似于在$x=0$ 点, 在$x=a$ ,可以用$f(a)$取值加修正项的方法获得. 可以构造多项式为:

$f(x) \approx P_n(x)=C_0+C_1(x-a)+C_2(x-a)^2+...+C_n(x-a)^n$

类似$x=0$ 处可以获得多项式:

!!! definition
    **函数在$x=a$ 附近的多项式近似表达式表示为:**

    $f(x) \approx f(a)+f'(a)(x-a)+\frac{f''(a)}{2!}(x-a)^2+...+\frac{f^{n}(a)}{n!}(x-a)^n$

"""
```

Example

example 7

构造自然对数函数在$x=1$ 附近的4阶泰勒近似

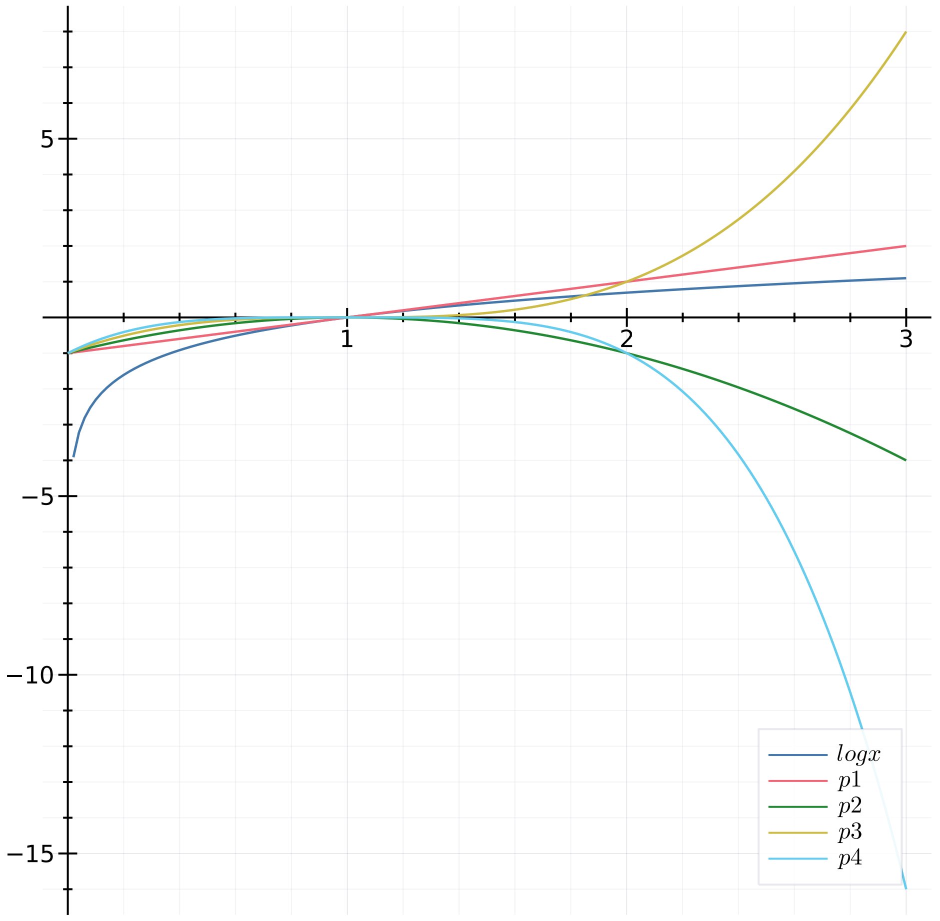
对函数求各阶导数, 并带入$x=1$,求值

然后带入上面表达式得:

$$ln(x)=(x-1)-\frac{(x-1)^2}{2}+\frac{(x-1)^3}{3}-\frac{(x-1)^4}{4}$$

绘图如下, 可见在$x=1$处能近似函数,但是当x 远离a 时, 结果偏移就比较大了

```
md"""
!!! example
    example 7

    构造自然对数函数在$x=1$ 附近的$4$阶泰勒近似


对函数求各阶导数, 并带入$x=1$,求值

然后带入上面表达式得:

$ln(x)=(x-1)-\frac{(x-1)^2}{2}+\frac{(x-1)^3}{3}-\frac{(x-1)^4}{4}$

绘图如下, 可见在$x=1$处能近似函数,但是当$x$ 远离$a$ 时, 结果偏移就比较大了

"""
```




```
let
    tspan=0:0.02:3

    f(x)=log(x)

    function pn(n)

         return function (x)
             res=0
             t=x-1
             for i  in 1:n

                 res=res+(-1)^(n+1)*(t^n/n)
              end
                return res
           end

     end




funarr=[pn(n)  for n in 1:4]

label=[L"logx" L"p1" L"p2" L"p3" L"p4"]


plot([f, funarr...],tspan,label=label,size=
(500,500),legend=:bottomright,frame=:origin)

```

```
end
```

```
    @htl("""<script src="https://cdn.bootcdn.net/ajax/libs/mathjax/3.2.0/es5/tex-svg-full.min.js"></script>
        """)
```