

# PJ01\_Romberg

November 11, 2018

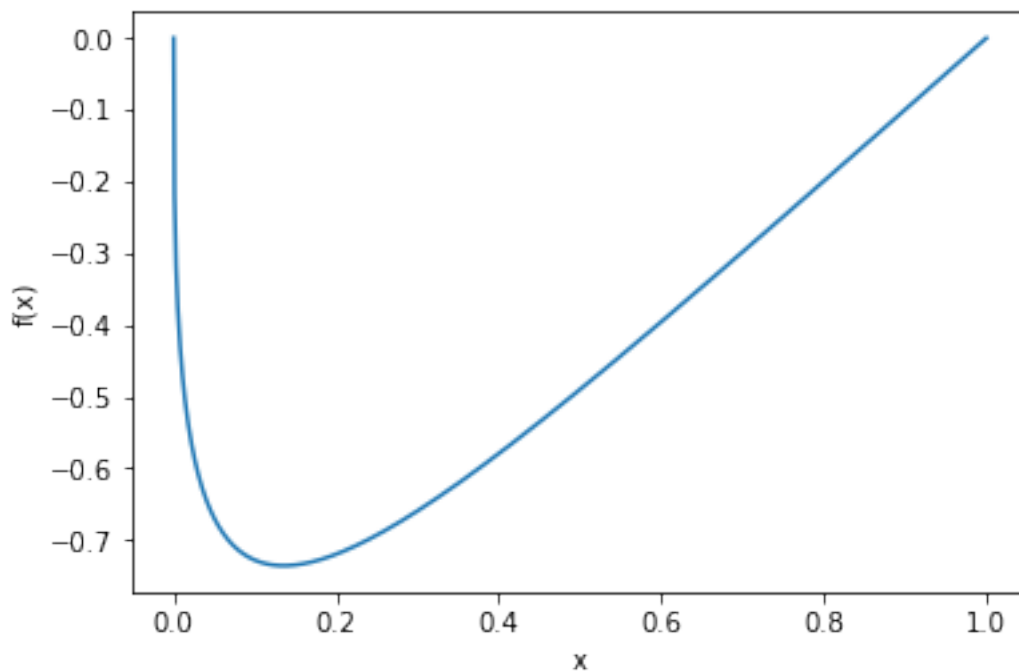
```
In [1]: %matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
from math import *
```

```
In [2]: def f(x):
        if x==0:
            return 0
        else :
            return sqrt(x)*log(x)
```

```
In [3]: x = np.linspace(0,1,1000)
```

```
In [4]: plt.plot(x, [f(i) for i in x])
plt.xlabel("x")
plt.ylabel("f(x)")
```

```
Out[4]: Text(0,0.5,'f(x)')
```



```

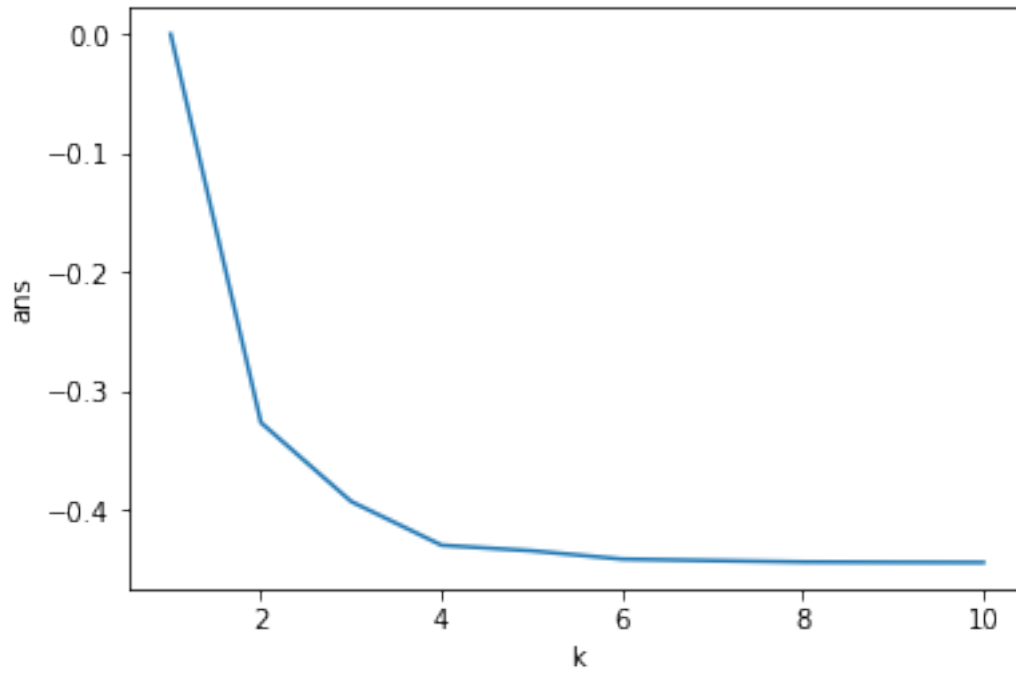
In [38]: h=1
         r=h/2*(f(1)+f(0))
         k=np.arange(1,11)
         H=np.zeros_like(k, dtype=float)
         ans=np.zeros_like(k, dtype=float)
         y0=np.zeros_like(k, dtype=float)
         y=np.zeros_like(k, dtype=float)
         y0[0]=r
         H[0]=h

In [40]: for i in range(1,len(k)):
         n = k[i]
         X = 0
         for j in range(1,2**(n-2)+1):
             X+=f((j-0.5)*h)
         y[0]=0.5*(y0[0]+X*h)
         for j in range(2, n+1):
             y[j-1]=y[j-2]+(y[j-2]-y0[j-1])/(pow(4,j-1)-1)
         h*=0.5
         for j in range(n):
             y0[j]=y[j]
             y[j]=0
         H[i]=h
         ans[i]=y0[n-1]

In [41]: plt.plot(k,ans)
         plt.xlabel("k")
         plt.ylabel("ans")

Out[41]: Text(0,0.5,'ans')

```



```
In [42]: plt.plot(H,ans+4/9)
plt.xlabel("h")
plt.ylabel("error")
```

```
Out[42]: Text(0,0.5,'error')
```

