In [1]:
```
from BSplineBase import BSplineBase
from utils import plotExampleBase,plot2DExample,plotSurface,generate_random_P
import numpy as np
```

绘制双三次B样条曲面

B样条基函数

定义B样条基函数

```
import numpy as np

class BSplineBase:
    def __init__(self,U=np.asarray([0,0,0,0,0.5,1,1,1,1])):
        self.U=U
        # self.n=n
    def GetValue(self,u,i,degree=3):
        U=self.U
        p=degree
        if degree==0:
            if u>=U[i] and u<=U[i+1]:
                return 1
            else:
                return 0
        else:
            degree=degree-1
            if u>=U[i] and u <=U[i+p+1]:
                if U[i+p]!=U[i] and U[i+p+1]!=U[i+1]:
                    return (u-U[i])/(U[i+p]-U[i])*self.GetValue(u,i,degree)+(U[i+p+1]-u)/(U[i+p+1]-U[i+1])*self.G
etValue(u,i+1,degree)
                elif U[i+p]!=U[i]:
                    return (u-U[i])/(U[i+p]-U[i])*self.GetValue(u,i,degree)
                elif U[i+p+1]!=U[i+1]:
                    return (U[i+p+1]-u)/(U[i+p+1]-U[i+1])*self.GetValue(u,i+1,degree)
                else:
                    return 0
            else:
                return 0
```

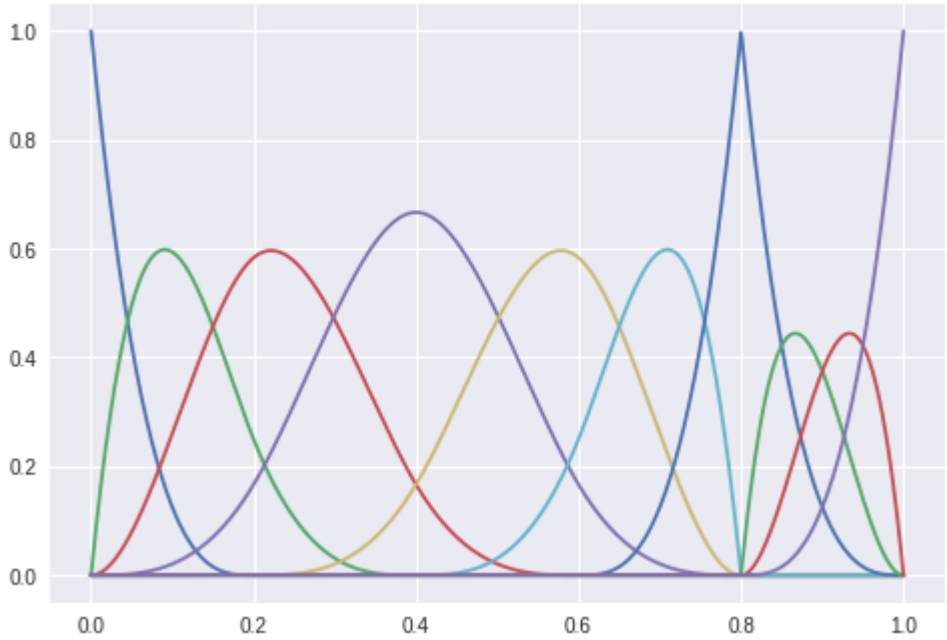
In [2]:
```
U = np.array([0,0,0,0,0.2,0.4,0.6,0.8,0.8,0.8,1,1,1,1])
V = np.array([0,0,0,0,0.2,0.2,0.2,0.4,0.6,0.8,1,1,1,1])
```

绘制的B样条基函数如下图所示

U 方向

In [3]:
```
plotExampleBase(U=U,n=3)
```

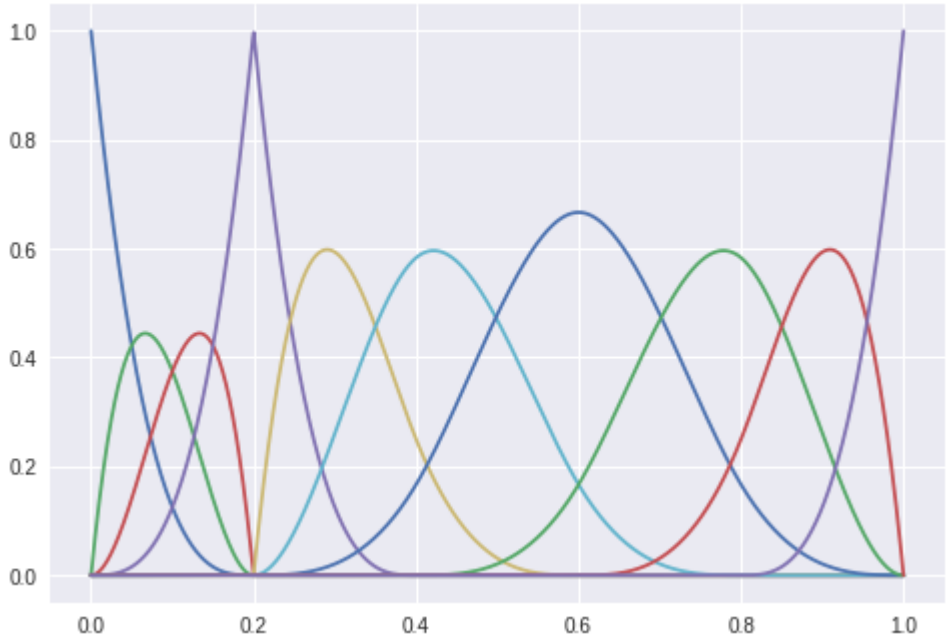
```
10
```



V 方向

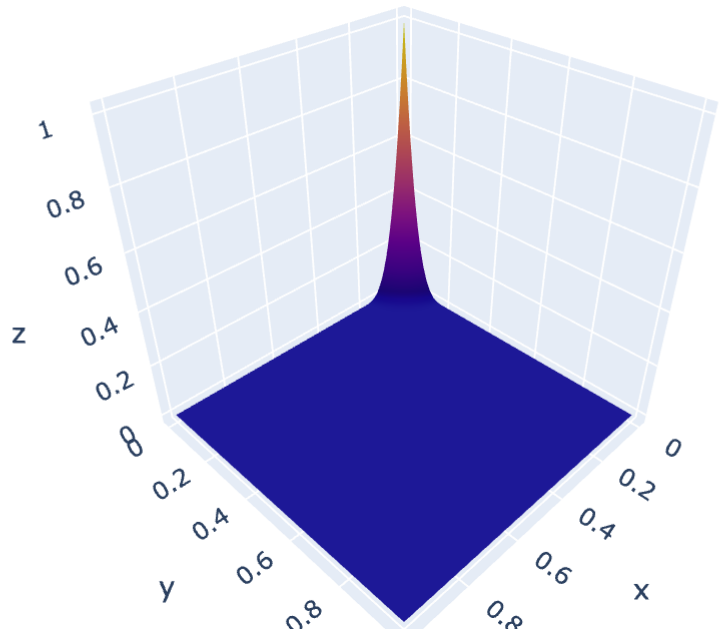
In [10]:
```
plotExampleBase(U=V,n=3)
```

```
10
```

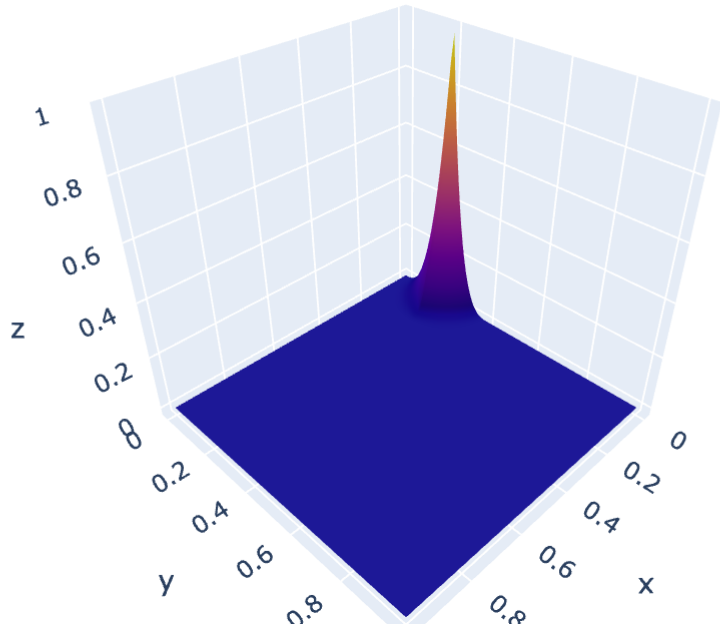


张量积基函数的绘制

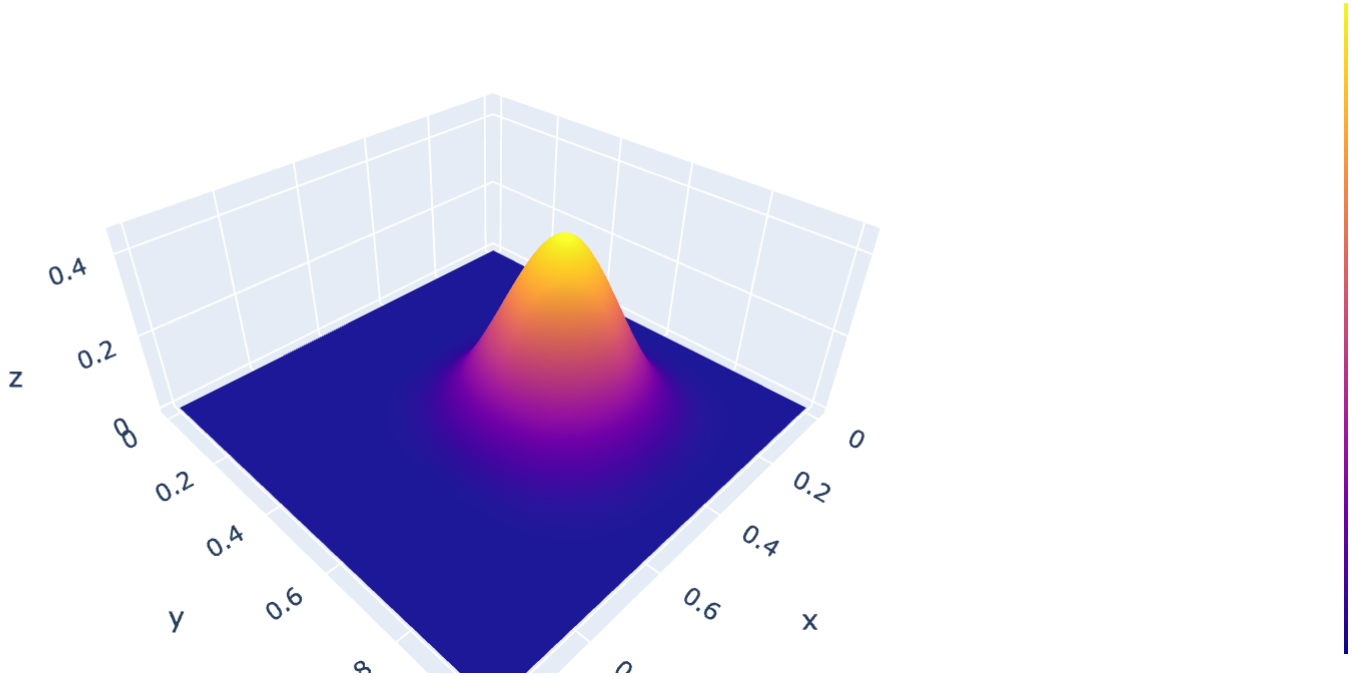
In [5]:
```
plot2DExample(U,V,i=0,j=0)
```



In [6]:
```
plot2DExample(U,V,i=0,j=3)
```



In [7]:
```
plot2DExample(U,V,i=3,j=6)
```



双三次B样条曲面的绘制

随机生成的P

```
import numpy as np
import math
def generate_random_P():
    P = []
    for i in range(-5,5):
        tmp = []
        for j in range(-5,5):
            tmp.append([i,j,10-math.floor((i*i+j*j)/20)+np.random.rand()])
        P.append(tmp)
    P=np.asarray(P)
    return P
```

In [11]:
```
P = generate_random_P()
print(P)
```

```
[[[-5.         -5.          8.40345892]
  [-5.         -4.          8.44557891]
  [-5.         -3.          9.82681644]
  [-5.         -2.          9.86655201]
  [-5.         -1.          9.79615623]
  [-5.          0.          9.48181466]
  [-5.          1.          9.23705771]
  [-5.          2.          9.87263815]
  [-5.          3.          9.92803818]
  [-5.          4.          8.92057119]]

 [[-4.         -5.          8.26126541]
  [-4.         -4.          9.20648574]
  [-4.         -3.          9.7903323 ]
  [-4.         -2.          9.84592371]
  [-4.         -1.         10.90397502]
  [-4.          0.         10.28341045]
  [-4.          1.         10.56462995]
  [-4.          2.          9.24712414]
  [-4.          3.          9.59711187]
  [-4.          4.          9.87036127]]

 [[-3.         -5.          9.41784256]
  [-3.         -4.          9.60266406]
  [-3.         -3.         10.99971199]
  [-3.         -2.         10.92796076]
  [-3.         -1.         10.51937373]
  [-3.          0.         10.86642652]
  [-3.          1.         10.92584685]
  [-3.          2.         10.00912443]
  [-3.          3.         10.27162369]
  [-3.          4.          9.71330262]]

 [[-2.         -5.          9.90670443]
  [-2.         -4.          9.1649982 ]
  [-2.         -3.         10.62553695]
  [-2.         -2.         10.22045737]
  [-2.         -1.         10.92299548]
  [-2.          0.         10.33666927]
  [-2.          1.         10.42686432]
```

```
 [-2.          2.         10.50820218]
 [-2.          3.         10.67503104]
 [-2.          4.          9.93733606]]

[[-1.         -5.          9.3955832 ]
 [-1.         -4.         10.53858005]
 [-1.         -3.         10.04504879]
 [-1.         -2.         10.12469699]
 [-1.         -1.         10.35946259]
 [-1.          0.         10.69516686]
 [-1.          1.         10.66029895]
 [-1.          2.         10.82210764]
 [-1.          3.         10.15670177]
 [-1.          4.         10.0934782 ]]

[[ 0.         -5.          9.22254981]
 [ 0.         -4.         10.42248877]
 [ 0.         -3.         10.80476173]
 [ 0.         -2.         10.28647752]
 [ 0.         -1.         10.81628797]
 [ 0.          0.         10.8917794 ]
 [ 0.          1.         10.07694181]
 [ 0.          2.         10.64895834]
 [ 0.          3.         10.61080486]
 [ 0.          4.         10.80665907]]

[[ 1.         -5.          9.636292  ]
 [ 1.         -4.         10.61756828]
 [ 1.         -3.         10.53711928]
 [ 1.         -2.         10.12948602]
 [ 1.         -1.         10.83841346]
 [ 1.          0.         10.53625719]
 [ 1.          1.         10.77639787]
 [ 1.          2.         10.95344232]
 [ 1.          3.         10.76407113]
 [ 1.          4.         10.61515092]]

[[ 2.         -5.          9.71650097]
 [ 2.         -4.          9.71645262]
 [ 2.         -3.         10.85845072]
 [ 2.         -2.         10.00805556]
```

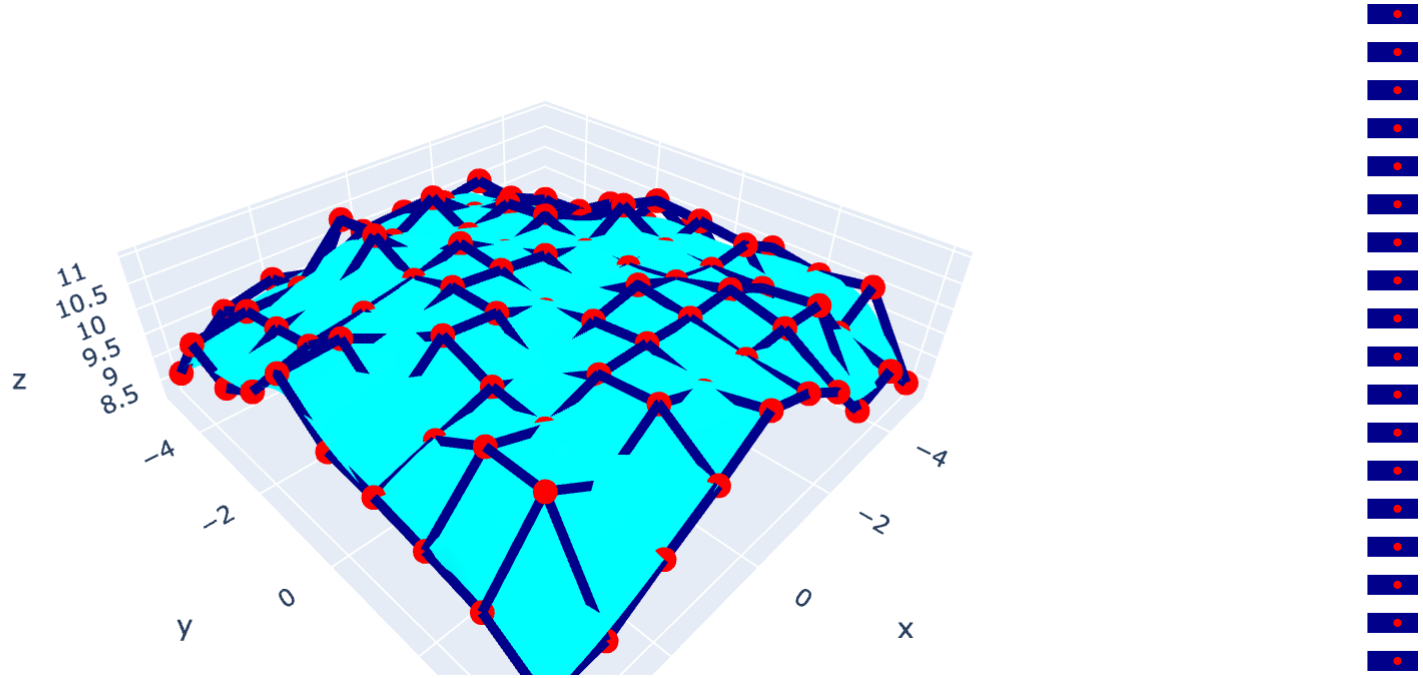
```
 [ 2.         -1.         10.21107822]
 [ 2.          0.         10.20454736]
 [ 2.          1.         10.78479848]
 [ 2.          2.         10.82884946]
 [ 2.          3.         10.35044135]
 [ 2.          4.          9.23355347]]

[[ 3.         -5.          9.03241077]
 [ 3.         -4.          9.07150842]
 [ 3.         -3.         10.01823035]
 [ 3.         -2.         10.29600977]
 [ 3.         -1.         10.60321965]
 [ 3.          0.         10.49554577]
 [ 3.          1.         10.5792061 ]
 [ 3.          2.         10.70732042]
 [ 3.          3.         10.79169745]
 [ 3.          4.          9.45627108]]

[[ 4.         -5.          8.6912883 ]
 [ 4.         -4.          9.3817444 ]
 [ 4.         -3.          9.56360327]
 [ 4.         -2.          9.67113859]
 [ 4.         -1.         10.9144551 ]
 [ 4.          0.         10.68233237]
 [ 4.          1.         10.3683772 ]
 [ 4.          2.          9.76501851]
 [ 4.          3.          9.69002243]
 [ 4.          4.          9.76769399]]]
```

开始绘制

In [9]:
```
plotSurface(U,V,P)
```



In [ ]:
```

```