

# kmax\_iteration

June 7, 2018

```
In [2]: import bootstrap
import matplotlib.pyplot as plt
import time
import datetime
import numpy as np

In [10]: def plot_grid(dim,table,sig_range,eps_range):
    allowed_sig=[]
    allowed_eps=[]
    disallowed_sig=[]
    disallowed_eps=[]
    for sig in sig_range:
        for eps in eps_range:
            sdp=bootstrap.SDP(sig,table)
            sdp.set_bound(0,float(dim))
            sdp.add_point(0,eps)
            result=sdp.iterate()
            if result:
                allowed_sig.append(sig)
                allowed_eps.append(eps)
            else:
                disallowed_sig.append(sig)
                disallowed_eps.append(eps)
    plt.plot(allowed_sig,allowed_eps,'r+')
    plt.plot(disallowed_sig,disallowed_eps,'b+')
    plt.show()

def iterate_k_max(k_range):
    bootstrap.cutoff=1e-10
    dim=3
    l_max=15
    n_max=4
    m_max=2
    for k in k_range:
        tab1=bootstrap.ConformalBlockTable(dim,k,l_max,m_max,n_max)
        tab2=bootstrap.ConvolvedBlockTable(tab1)
        plot_grid(dim,tab2,sig_set,eps_set)
```

```

In [11]: start_time=time.time()
        start_cpu=time.clock()

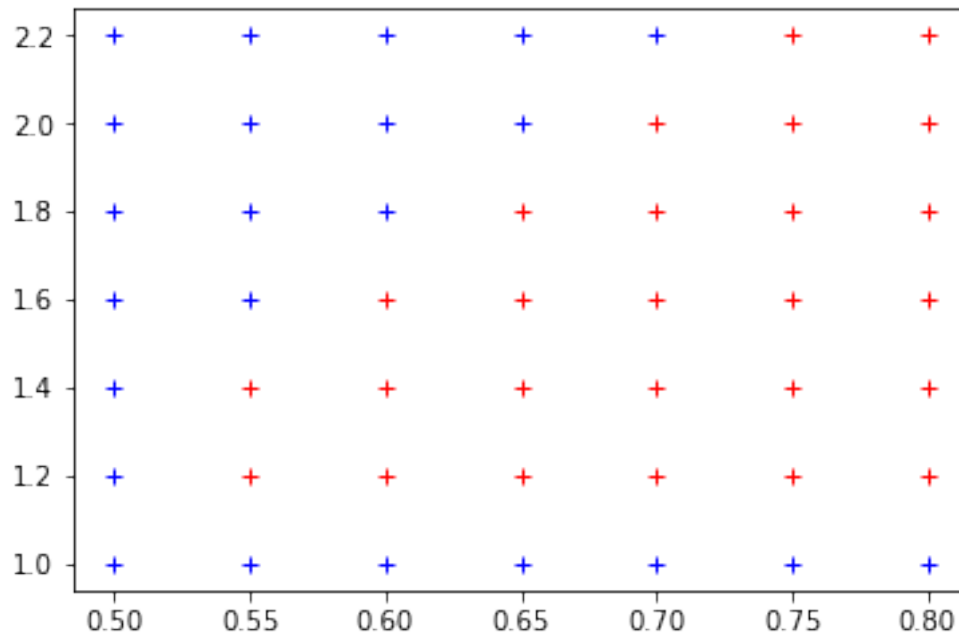
        k_set=np.arange(3,10,1)
        sig_set=np.arange(0.5,0.85,0.05)
        eps_set=np.arange(1.0,2.2,0.2)
        iterate_k_max(k_set)

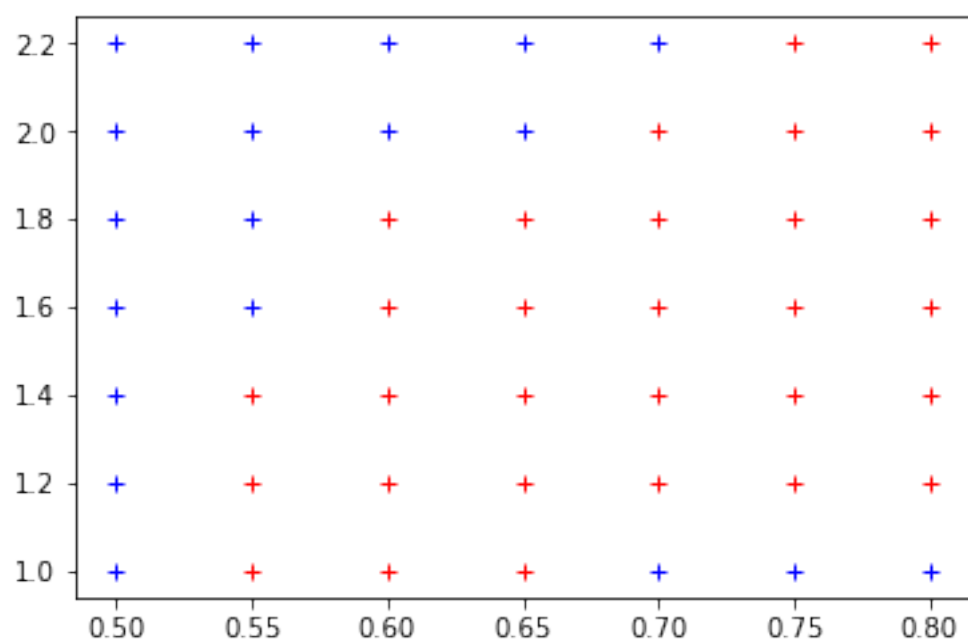
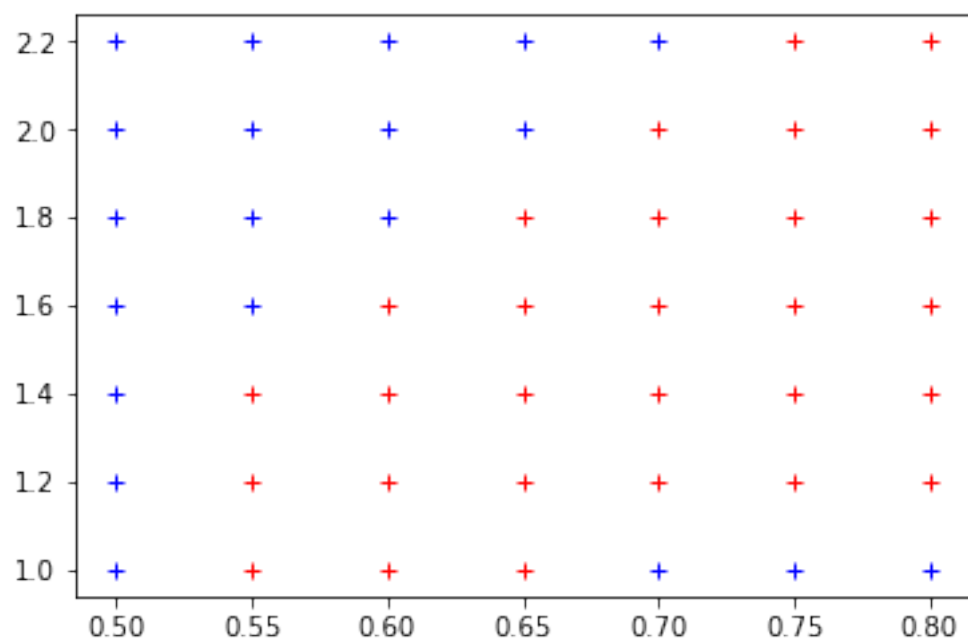
        end_time=time.time()
        end_cpu=time.clock()
        run_time=time.strftime("%H:%M:%S",time.gmtime(end_time-start_time))
        cpu_time=time.strftime("%H:%M:%S",time.gmtime(end_cpu-start_cpu))
        print("Run time "+run_time, "CPU time "+cpu_time)

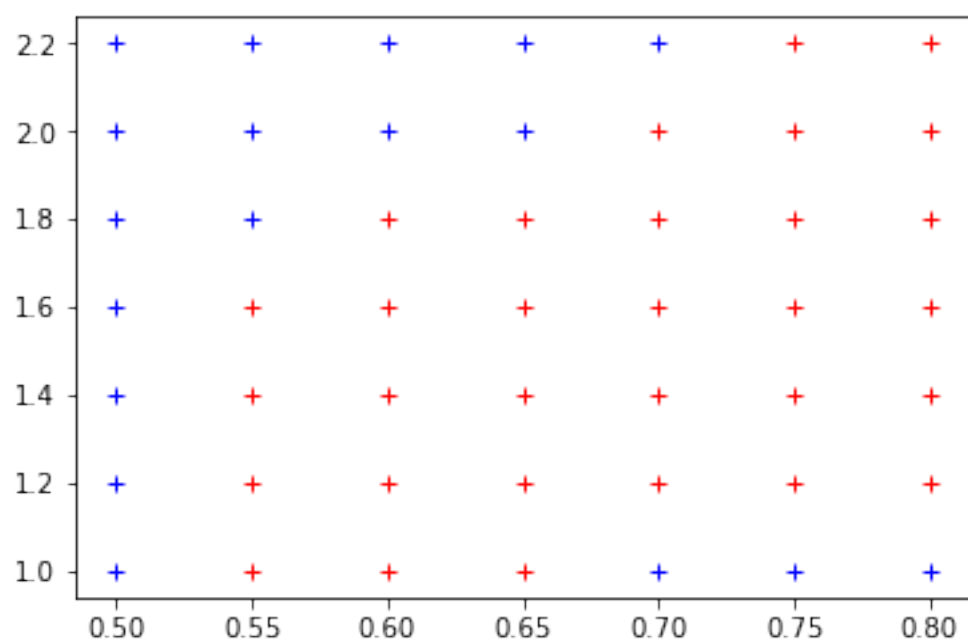
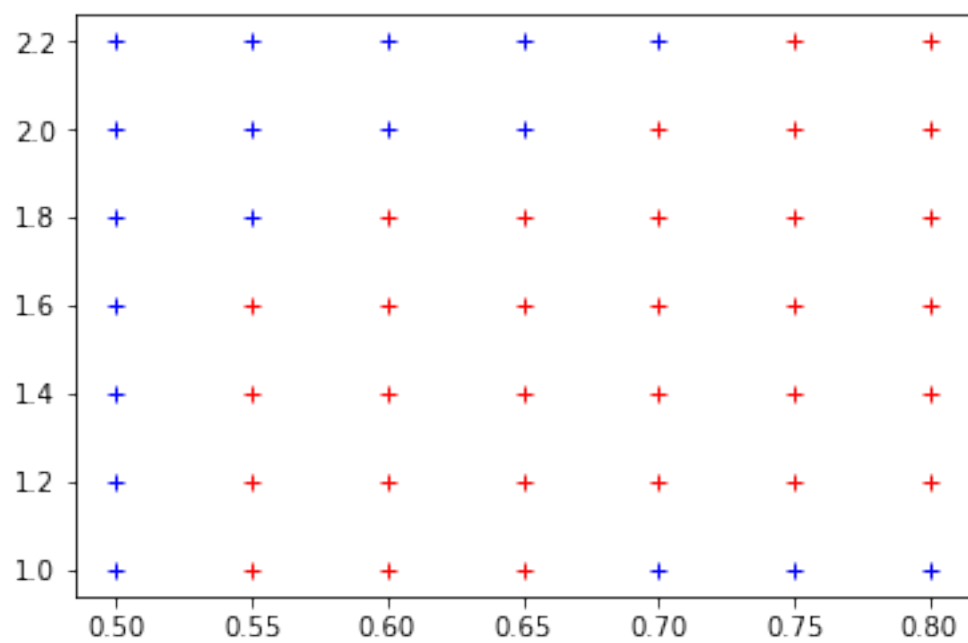
```

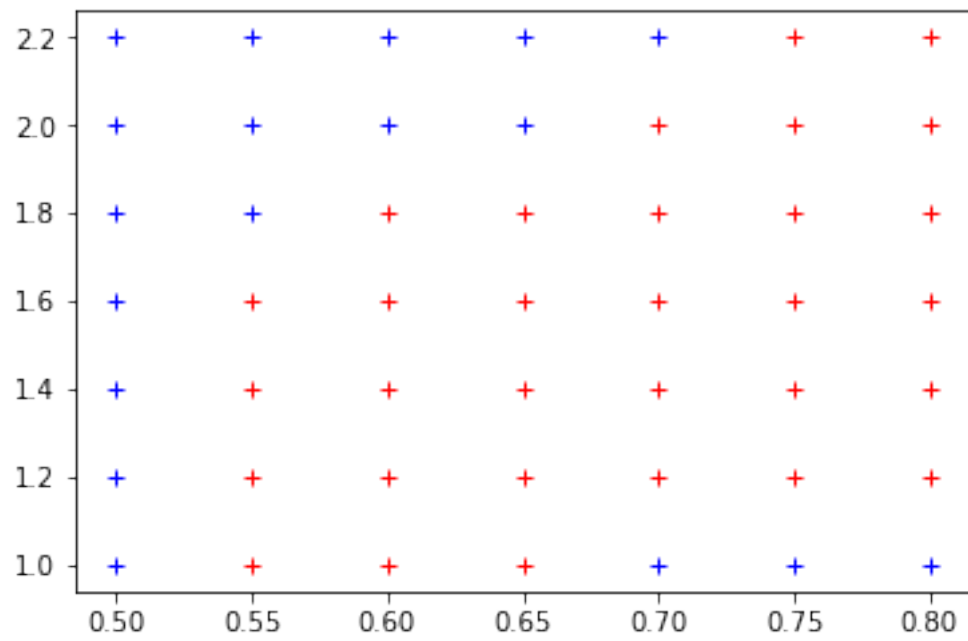
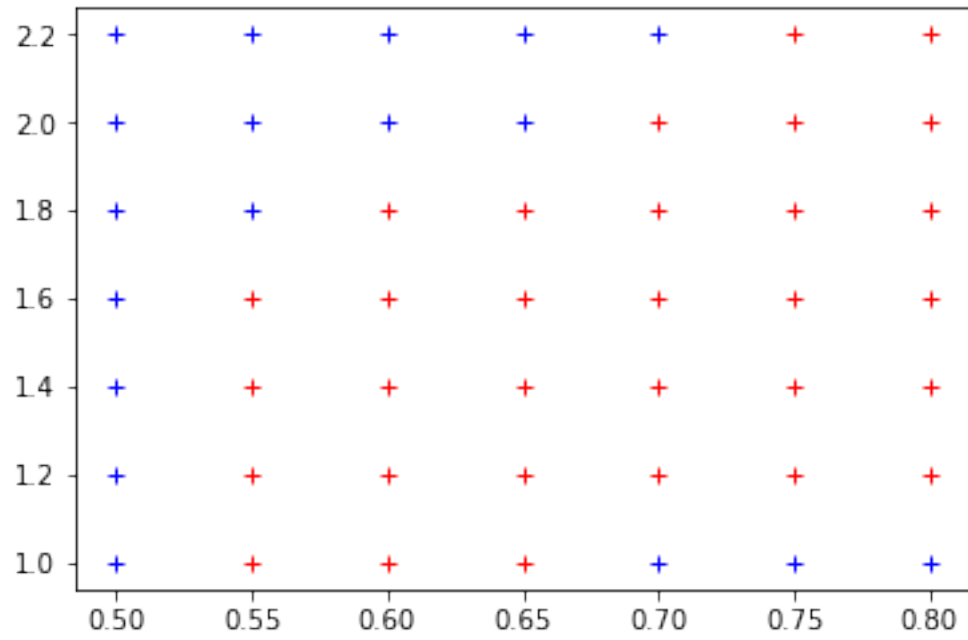
/Users/MatthewDowens/Dropbox/PhD/bootstrap/pycftboot/bootstrap.py:952: RuntimeWarning: invalid v  
product \*= x - (p - shift)

/Users/MatthewDowens/Dropbox/PhD/bootstrap/pycftboot/bootstrap.py:953: RuntimeWarning: invalid v  
return (base \*\* (x + shift)) / product









Run time 00:23:50 CPU time 00:13:26