

# Ex 5 noteobok

April 12, 2019

```
In [2]: import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from itertools import product
from scipy import stats
```

## 0.0.1 Exercise 1

```
In [99]: def func(t1, t2, reg):
    return (t1 - 3)**2 / 2 + (t2 - 1)**2 / 3 + reg(t1, t2)

def r0(l, t1, t2):
    return l*(t1**0 + t2**0)

def r1(l, t1, t2):
    return l*(abs(t1) + abs(t2))

def r2(l, t1, t2):
    return l*(t1**2 + t2**2)

def plotter(seq, reg, lamb, title):
    fig = plt.figure()
    ax = fig.add_subplot(1,1,1, projection='3d')
    pairs = np.array(list(product(seq, seq)))
    res = np.abs(func(pairs[:, 0], pairs[:, 1], lambda x, y: reg(lamb, x, y)))
    ax.plot_trisurf(pairs[:, 0], pairs[:, 1], res)
    plt.title(title)
    minimum = res.argmin()
    print('Minimum value of %f reached with values %f %f' % (res[minimum], pairs[minimum][0], pairs[minimum][1]))

    plt.show()

def get_min_pair(seq, reg, lamb):
    pairs = np.array(list(product(seq, seq)))
    res = np.abs(func(pairs[:, 0], pairs[:, 1], lambda x, y: reg(lamb, x, y)))
    minimum = res.argmin()
    return np.array([pairs[minimum], np.min(res)])

def plot_3d(x, y, z, title=''):
    fig = plt.figure()
    ax = fig.add_subplot(1,1,1, projection='3d')
    ax.scatter(x, y, z, s=5)
    plt.title(title)
```

```

elev = 25
azim = 60.5

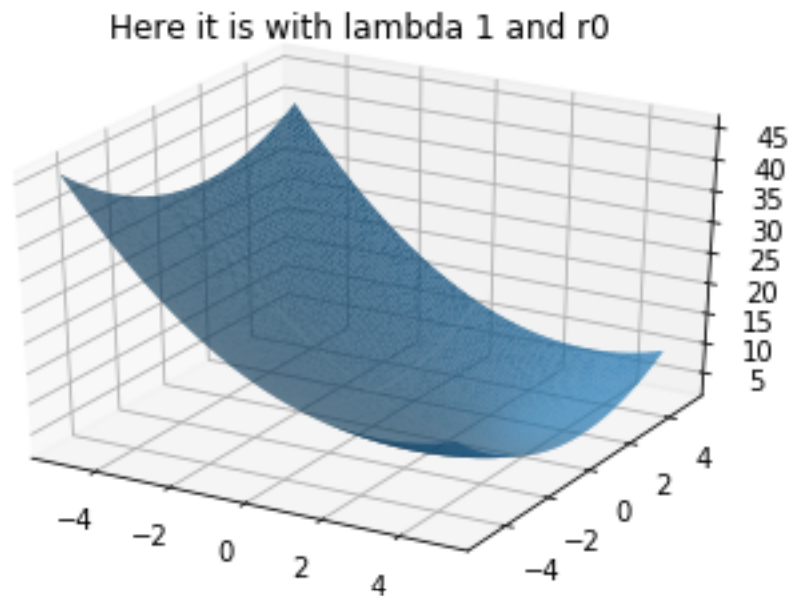
ax.set_zlabel('Lambda')
ax.set_xlabel('First coord')
ax.set_ylabel('Second coord')
ax.view_init(elev, azim)
plt.show()

```

```
In [4]: seq = np.concatenate([np.linspace(-5, 5, 100), np.array([0])])
```

```
In [5]: plotter(seq, r0, 1, 'Here it is with lambda 1 and r0')
```

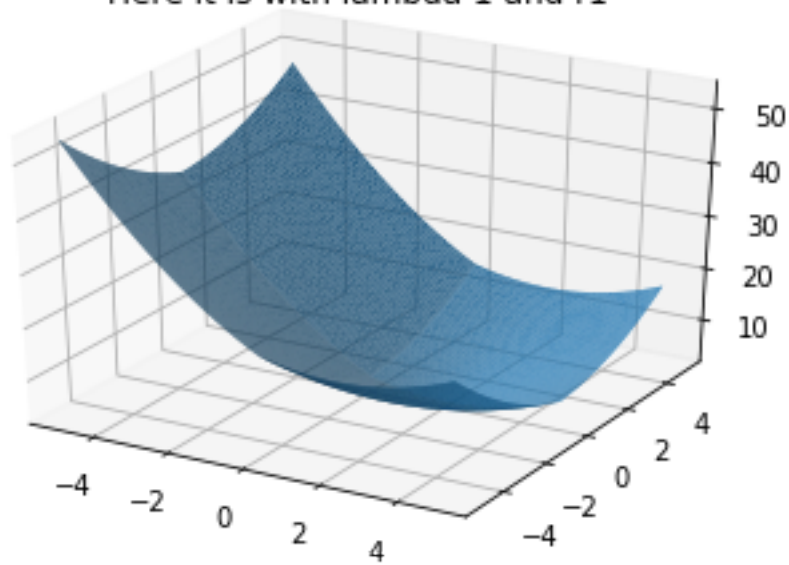
Minimum value of 2.000748 reached with values 2.979798 0.959596



```
In [6]: plotter(seq, r1, 1, 'Here it is with lambda 1 and r1')
```

Minimum value of 2.833792 reached with values 1.969697 0.000000

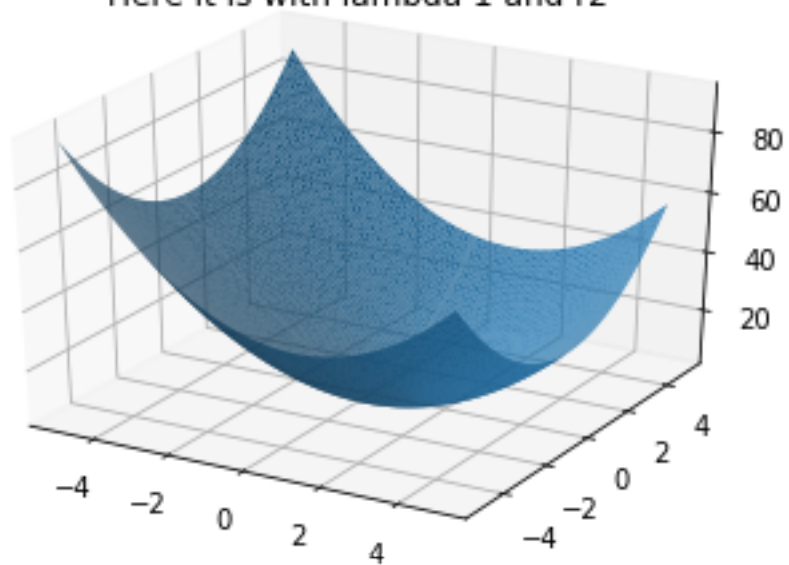
Here it is with lambda 1 and r1



```
In [7]: plotter(seq, r2, 1, 'Here it is with lambda 1 and r2')
```

Minimum value of 3.252457 reached with values 0.959596 0.252525

Here it is with lambda 1 and r2



```
In [40]: coordStore = np.zeros((3, 1000, 2))
```

```
In [100]: lambdas = np.arange(-500, 500)  
          seq = np.concatenate([np.linspace(-5, 5, 100), np.array([0])])
```

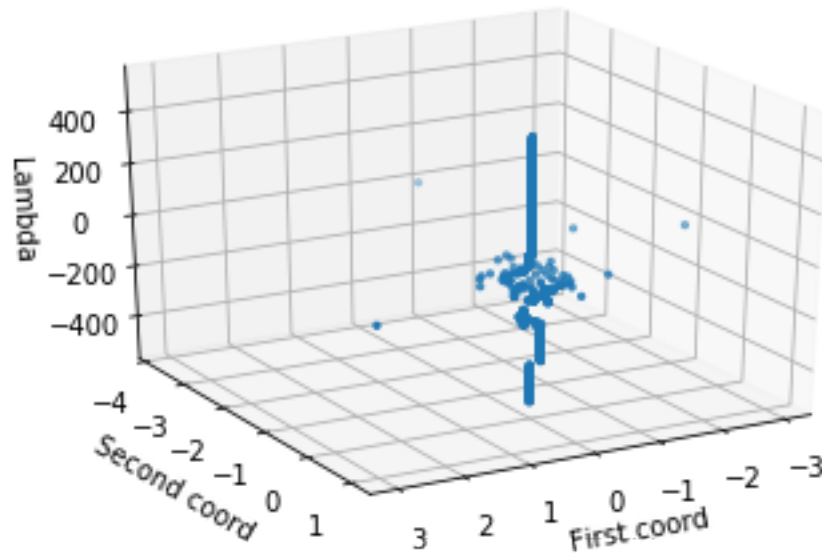
```

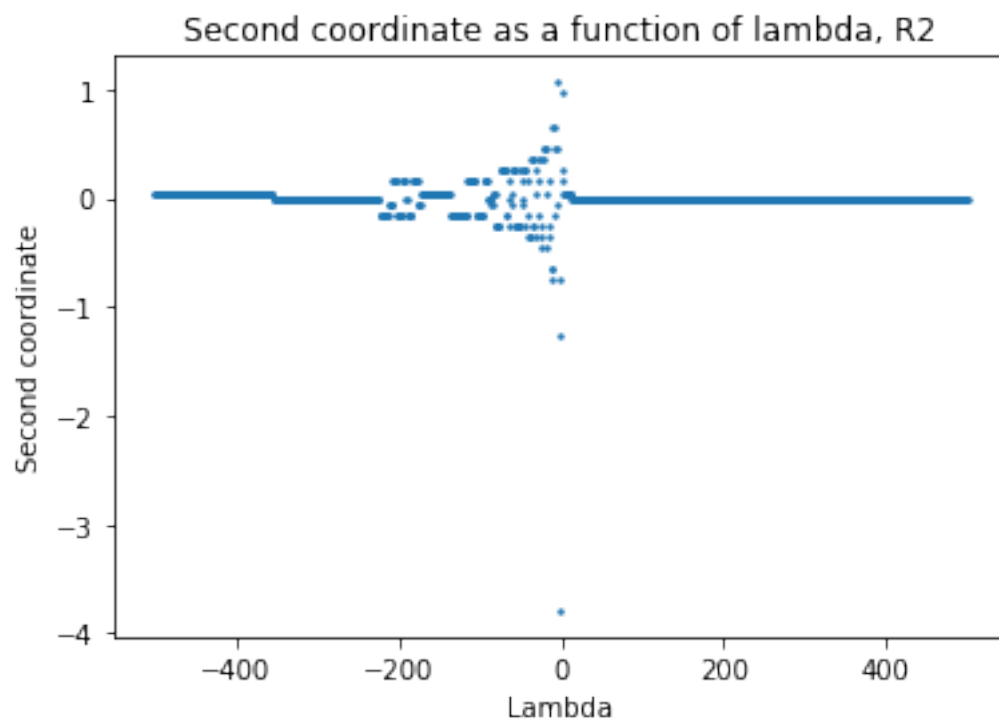
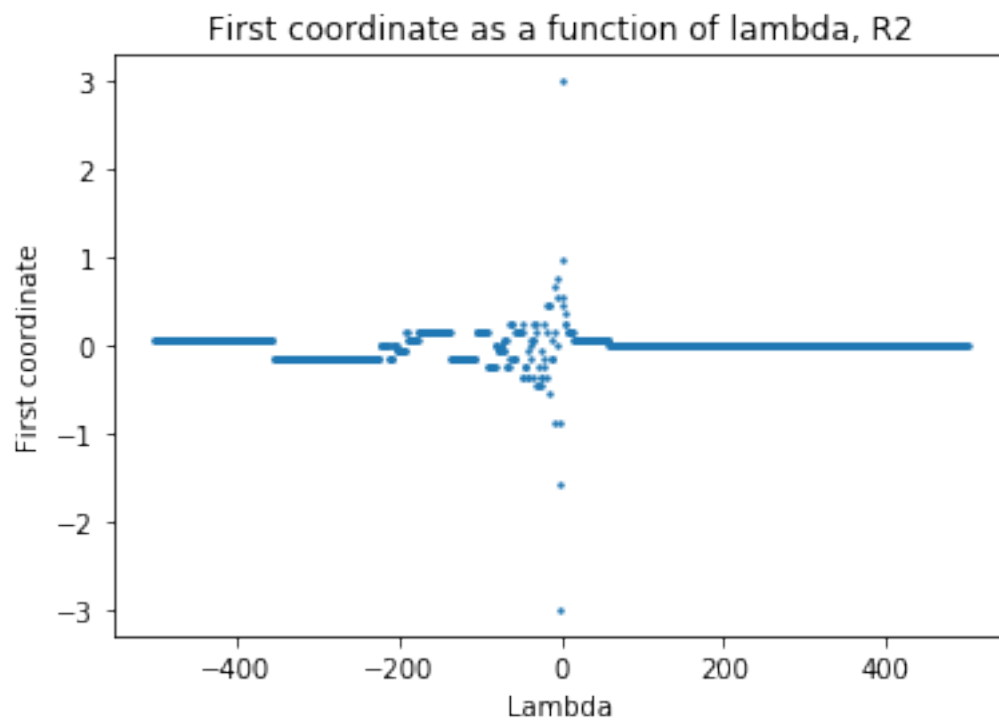
res = np.array([get_min_pair(seq, r2, l) for l in lambdas])
coords = np.array(list(map(lambda x: np.array([x[0], x[1]]), res[:, 0])))

plot_3d(coords[:, 0], coords[:, 1], lambdas, '')
coordStore[2] = coords

plt.scatter(lambdas, coords[:, 0], s=2)
plt.title('First coordinate as a function of lambda, R2')
plt.xlabel('Lambda')
plt.ylabel('First coordinate')
plt.show()
plt.scatter(lambdas, coords[:, 1], s=2)
plt.title('Second coordinate as a function of lambda, R2')
plt.xlabel('Lambda')
plt.ylabel('Second coordinate')
plt.show()

```





```

In [101]: lambdas = np.arange(-500, 500)
          seq = np.concatenate([np.linspace(-5, 5, 100), np.array([0])])

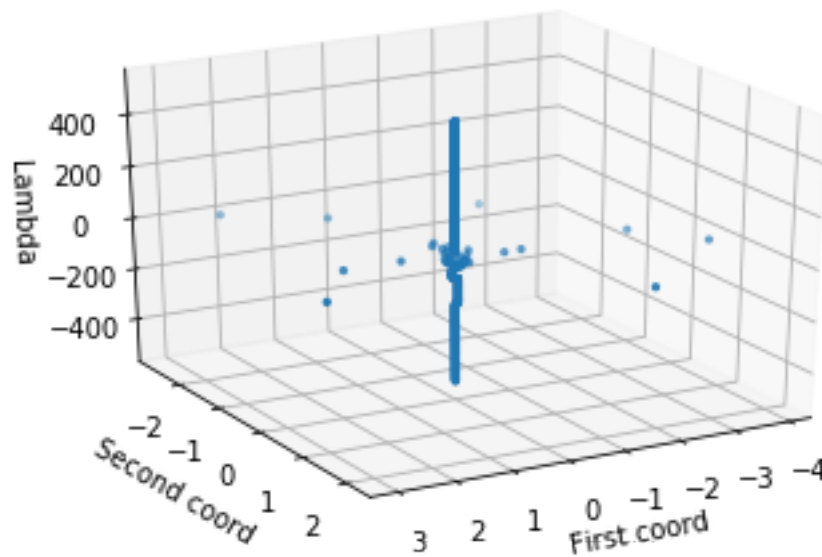
          res = np.array([get_min_pair(seq, r1, l) for l in lambdas])
          coords = np.array(list(map(lambda x: np.array([x[0], x[1]]), res[:, 0])))

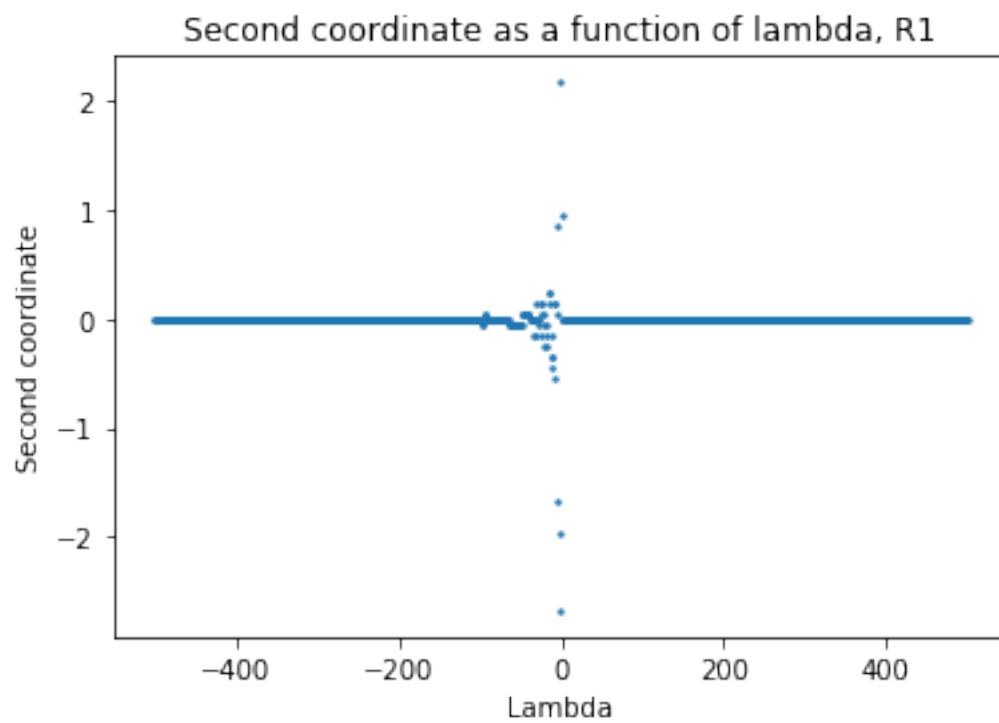
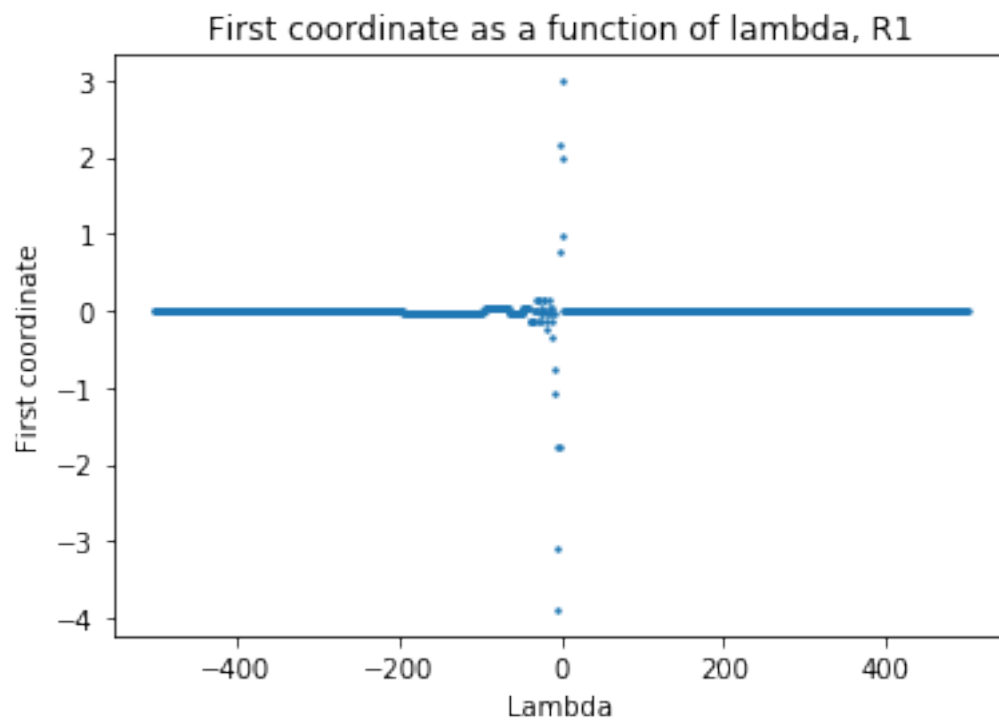
          plot_3d(coords[:, 0], coords[:, 1], lambdas, '')

          coordStore[1] = coords

          plt.scatter(lambdas, coords[:, 0], s=2)
          plt.title('First coordinate as a function of lambda, R1')
          plt.xlabel('Lambda')
          plt.ylabel('First coordinate')
          plt.show()
          plt.scatter(lambdas, coords[:, 1], s=2)
          plt.title('Second coordinate as a function of lambda, R1')
          plt.xlabel('Lambda')
          plt.ylabel('Second coordinate')
          plt.show()

```





```

In [102]: lambdas = np.arange(-500, 500)
          seq = np.concatenate([np.linspace(-5, 5, 100), np.array([0])])

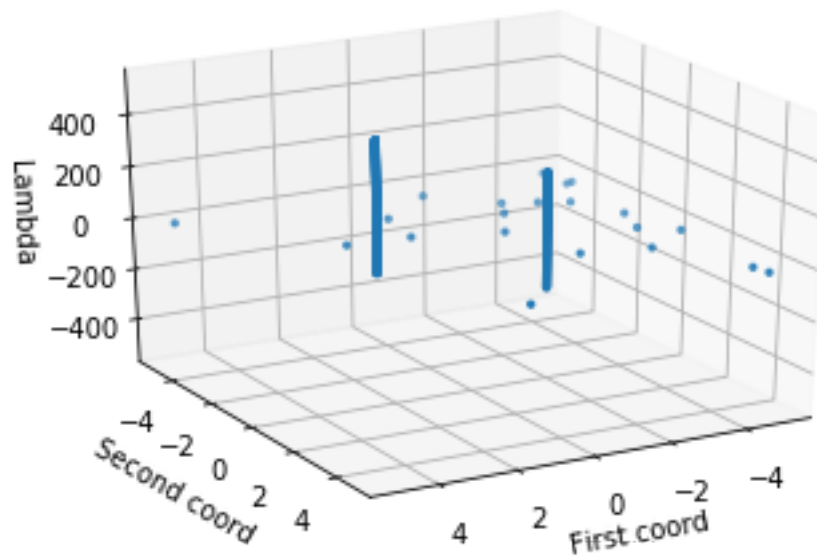
          res = np.array([get_min_pair(seq, r0, l) for l in lambdas])
          coords = np.array(list(map(lambda x: np.array([x[0], x[1]]), res[:, 0])))

          plot_3d(coords[:, 0], coords[:, 1], lambdas, '')

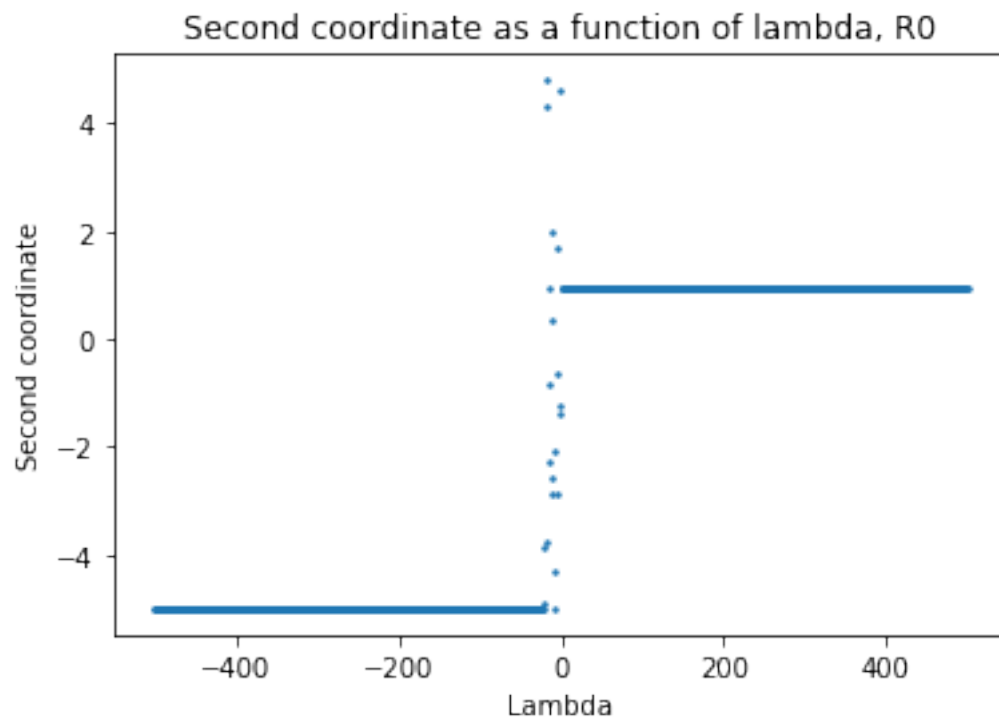
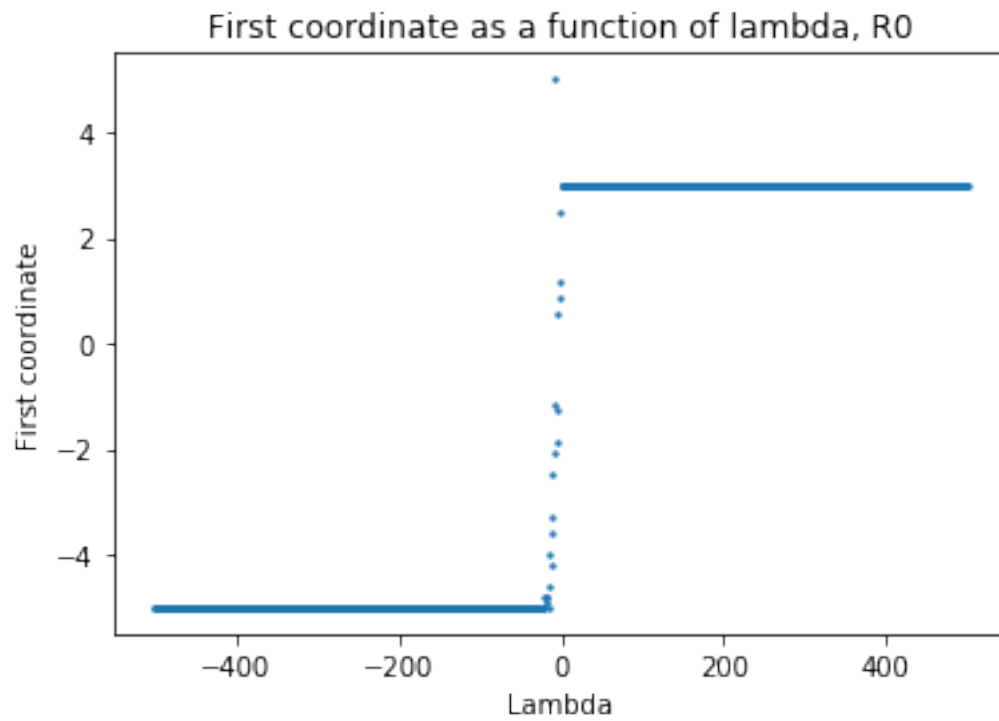
          coordStore[0] = coords

          plt.scatter(lambdas, coords[:, 0], s=2)
          plt.title('First coordinate as a function of lambda, R0')
          plt.xlabel('Lambda')
          plt.ylabel('First coordinate')
          plt.show()
          plt.scatter(lambdas, coords[:, 1], s=2)
          plt.title('Second coordinate as a function of lambda, R0')
          plt.xlabel('Lambda')
          plt.ylabel('Second coordinate')
          plt.show()

```







```

In [92]: for ind in range(0,3):
    print('\n\n\n\nRESULTS FOR R%i' % ind)
    i = coordStore[ind]

    oneMask = ((i[:, 0] == 0) & (i[:, 1] != 0)) | ((i[:, 0] != 0) & (i[:, 1] == 0))
    oneZero = i[oneMask]
    twoZero = i[((i[:, 0] == 0) & (i[:, 1] == 0))]
    if len(oneZero) > 0:
        print('Unique solutions where one coordinate is zero for R%i\n' % ind, np.unique(oneZero))
        print('Lambda values \n', lambdas[oneMask])
    else:
        print('No coordinates where only one is zero for R%i' % ind)
    if len(twoZero) > 0:
        print('Number of pairs where both coordinates were zero %i' % len(twoZero))
        print('Lambda values \n', lambdas[oneMask])
    else:
        print('No coordinates where only one is zero for R%i' % ind)

```

RESULTS FOR R0

No coordinates where only one is zero for R0

No coordinates where only one is zero for R0

RESULTS FOR R1

Unique solutions where one coordinate is zero for R1

```

[[-0.15151515  0.          ]
 [-0.05050505  0.          ]
 [ 0.          -0.25252525]
 [ 0.          -0.15151515]
 [ 0.          -0.05050505]
 [ 0.          0.05050505]
 [ 0.          0.15151515]
 [ 0.05050505  0.          ]
 [ 0.15151515  0.          ]
 [ 0.95959596  0.          ]
 [ 1.96969697  0.          ]]

```

Lambda values

```

[-194 -193 -192 -191 -190 -189 -188 -187 -186 -185 -184 -183 -182 -181
 -180 -179 -178 -177 -176 -175 -174 -173 -172 -171 -170 -169 -168 -167
 -166 -165 -164 -163 -162 -161 -160 -159 -158 -157 -156 -155 -154 -153
 -152 -151 -150 -149 -148 -147 -146 -145 -144 -143 -142 -141 -140 -139
 -138 -137 -136 -135 -134 -133 -132 -131 -130 -129 -128 -127 -126 -125
 -124 -123 -122 -121 -120 -119 -118 -117 -116 -115 -114 -113 -112 -111
 -110 -109 -108 -107 -106 -105 -104 -103 -102 -101 -100 -99 -98 -97
 -96 -95 -94 -93 -92 -91 -90 -89 -88 -87 -86 -85 -84 -83
 -82 -81 -80 -79 -78 -77 -76 -75 -74 -73 -72 -71 -70 -69
 -68 -67 -66 -65 -39 -38 -37 -36 -35 -34 -33 -32 -31 -30
 -29 -28 -20  1  2]

```

Number of pairs where both coordinates were zero 803

Lambda values

```

[-194 -193 -192 -191 -190 -189 -188 -187 -186 -185 -184 -183 -182 -181
-180 -179 -178 -177 -176 -175 -174 -173 -172 -171 -170 -169 -168 -167
-166 -165 -164 -163 -162 -161 -160 -159 -158 -157 -156 -155 -154 -153
-152 -151 -150 -149 -148 -147 -146 -145 -144 -143 -142 -141 -140 -139
-138 -137 -136 -135 -134 -133 -132 -131 -130 -129 -128 -127 -126 -125
-124 -123 -122 -121 -120 -119 -118 -117 -116 -115 -114 -113 -112 -111
-110 -109 -108 -107 -106 -105 -104 -103 -102 -101 -100 -99 -98 -97
-96 -95 -94 -93 -92 -91 -90 -89 -88 -87 -86 -85 -84 -83
-82 -81 -80 -79 -78 -77 -76 -75 -74 -73 -72 -71 -70 -69
-68 -67 -66 -65 -39 -38 -37 -36 -35 -34 -33 -32 -31 -30
-29 -28 -20 1 2]

```

## RESULTS FOR R2

Unique solutions where one coordinate is zero for R2

```

[[-0.35353535 0.      ]
[-0.25252525 0.      ]
[-0.15151515 0.      ]
[ 0.          -0.25252525]
[ 0.          -0.15151515]
[ 0.           0.15151515]
[ 0.           0.25252525]
[ 0.           0.35353535]
[ 0.           1.06060606]
[ 0.05050505 0.      ]
[ 0.15151515 0.      ]
[ 0.25252525 0.      ]]

```

Lambda values

```

[-354 -353 -352 -351 -350 -349 -348 -347 -346 -345 -344 -343 -342 -341
-340 -339 -338 -337 -336 -335 -334 -333 -332 -331 -330 -329 -328 -327
-326 -325 -324 -323 -322 -321 -320 -319 -318 -317 -316 -315 -314 -313
-312 -311 -310 -309 -308 -307 -306 -305 -304 -303 -302 -301 -300 -299
-298 -297 -296 -295 -294 -293 -292 -291 -290 -289 -288 -287 -286 -285
-284 -283 -282 -281 -280 -279 -278 -277 -276 -275 -274 -273 -272 -271
-270 -269 -268 -267 -266 -265 -264 -263 -262 -261 -260 -259 -258 -257
-256 -255 -254 -253 -252 -251 -250 -249 -248 -247 -246 -245 -244 -243
-242 -241 -240 -239 -238 -237 -236 -235 -234 -233 -232 -231 -230 -229
-228 -227 -226 -225 -224 -223 -222 -221 -220 -219 -218 -217 -216 -215
-214 -213 -207 -206 -205 -204 -203 -191 -190 -90 -89 -88 -87 -81
-80 -79 -75 -74 -65 -48 -37 -4 13 14 15 16 17 18
19 20 21 22 23 24 25 26 27 28 29 30 31 32
33 34 35 36 37 38 39 40 41 42 43 44 45 46
47 48 49 50 51 52 53 54 55 56 57 58]

```

Number of pairs where both coordinates were zero 441

Lambda values

```

[-354 -353 -352 -351 -350 -349 -348 -347 -346 -345 -344 -343 -342 -341
-340 -339 -338 -337 -336 -335 -334 -333 -332 -331 -330 -329 -328 -327
-326 -325 -324 -323 -322 -321 -320 -319 -318 -317 -316 -315 -314 -313
-312 -311 -310 -309 -308 -307 -306 -305 -304 -303 -302 -301 -300 -299
-298 -297 -296 -295 -294 -293 -292 -291 -290 -289 -288 -287 -286 -285
-284 -283 -282 -281 -280 -279 -278 -277 -276 -275 -274 -273 -272 -271

```

-270	-269	-268	-267	-266	-265	-264	-263	-262	-261	-260	-259	-258	-257
-256	-255	-254	-253	-252	-251	-250	-249	-248	-247	-246	-245	-244	-243
-242	-241	-240	-239	-238	-237	-236	-235	-234	-233	-232	-231	-230	-229
-228	-227	-226	-225	-224	-223	-222	-221	-220	-219	-218	-217	-216	-215
-214	-213	-207	-206	-205	-204	-203	-191	-190	-90	-89	-88	-87	-81
-80	-79	-75	-74	-65	-48	-37	-4	13	14	15	16	17	18
19	20	21	22	23	24	25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40	41	42	43	44	45	46
47	48	49	50	51	52	53	54	55	56	57	58]		

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: