In [1]:

```
import pandas as pd
import csv

df = pd.read_csv("location0322.csv",sep = ',', header=None)

#print(max(df[0])) the max number of cols0
#print(len(df[0])) the rows of cols0
cols1 = []
cols2 = []
cols3 = []
result01 = []
result02 = []
for i in range(1,max(df[0])+1):
    for j in range(len(df[0])):
        if(df[0][j] == i):
            cols1.append(df[1][j])
            cols2.append(df[2][j])
            cols3.append(df[3][j])
        else:
            continue
    if(len(cols1)):
        header = str(i).zfill(4)+":"
        #for m in range(len(cols1)):
        #   header = header + str(cols1[m])+ ","
        #print(header)
        #result.append(header)
        #print(header) # convert to writelines
        #rows = ""
        for n in range(len(cols1)):
            #rows = cols1[n] + ","
            #print(header)
            for p in range(len(cols1)):
                if(n<p):
                    if(cols2[n] == cols2[p]):
                        rows = header + str(cols2[n]) + "," + str(cols3[n])
                    else:
                        rows = header + str(cols2[n]) + "," + str(cols3[n])
                        print(rows)
            rows = ""
        #result.append("\n")
        #print(i)
        #print(cols1)
        #print(cols2)
        #print(cols3)
        #print("---")
        cols1.clear()
        cols2.clear()
        cols3.clear()
```

```
0001:5,55625845,9,19049427,
0001:5,55625845,1,38838198,
0001:9,19049427,1,38838198,
0002:X,72044545,6,26017085,
0003:17,42113111,1,77562416,
0003:17,42113111,3,9779860,
0003:17,42113111,20,18138118,
0003:17,42113111,3,20040446,
0003:17,42113111,3,183697797,
0003:1,77562416,3,9779860,
0003:1,77562416,20,18138118,
0003:1,77562416,3,20040446,
0003:1,77562416,3,183697797,
0003:1,77562416,17,37406886,
0003:3,9779860,20,18138118,
0003:3,9779860,17,37406886,
0003:20,18138118,3,20040446,
0003:20,18138118,3,183697797,
0003:20,18138118,17,37406886,
```

In [12]:

```
from dcicutils import jh_utils
import cooler
import numpy as np
file_path = jh_utils.mount_4dn_file("4DNFIB59T7NN")
c = cooler.Cooler(file_path + "::/resolutions/100000")
import pandas as pd
import csv
df = pd.read_csv("Same.csv",sep = ",", header = None)
cols1 = []
cols2 = []
cols3 = []
cols4 = []
result = []
resultList = []
for i in range(0,max(df[0])):
    cols1.append(df[1][i])
    cols2.append(df[2][i])
    cols3.append(df[3][i])
    cols4.append(df[4][i])

for p in range(len(cols1)):
    A1 = c.matrix().fetch("chr" + str(cols1[p]) + ":" + str(cols2[p]) + "-"
                          "chr" + str(cols3[p]) + ":" + str(cols4[p]) + "-"
    if(str(np.log10(A1)[0][0]) != "nan" and str(np.log10(A1)[0][0]) != "-in
        result.append(np.log10(A1)[0][0])
        print(p)
print(result)
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
```

```
In [13]: import math
         print(type(result[0]))
         mean = np.mean(result)
         std = np.std(result)
         print(mean)
         print(std)
```

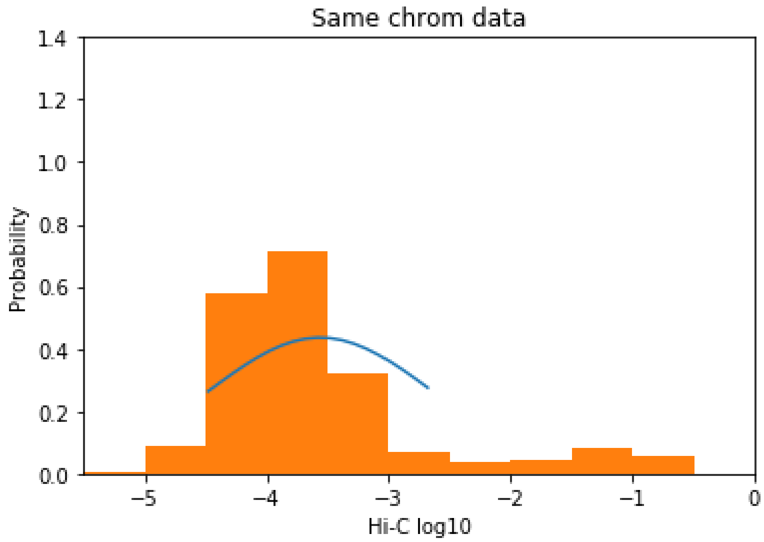
```
<class 'numpy.float64'>
-3.5596018714100333
0.9155104072112874
```

```
In [35]: import matplotlib.pyplot as plt
         def normfun(x,mu,sigma):
             pdf = np.exp(-((x - mu)**2)/(2*sigma**2)) / (sigma * np.sqrt(2*np.pi))
             return pdf

         x = np.arange(-4.48,-2.64,0.1)
         y = normfun(x, mean, std)
         plt.plot(x,y)

         #画出直方图，最后的“normed”参数，是赋范的意思，数学概念
         plt.hist(result, bins=[-5.5, -5, -4.5, -4, -3.5, -3, -2.5, -2, -1.5, -1, -0
         plt.title('Same chrom data')
         plt.xlabel('Hi-C log10')
         plt.ylabel('Probability')
         plt.xlim((-5.5, 0))
         plt.ylim((0, 1.4))
         #输出
         plt.show()
```

```
/opt/conda/lib/python3.6/site-packages/ipykernel_launcher.py:11: Matplotl
ibDeprecationWarning:
The 'normed' kwarg was deprecated in Matplotlib 2.1 and will be removed i
n 3.1. Use 'density' instead.
  # This is added back by InteractiveShellApp.init_path()
```



In [18]:

```
from dcicutils import jh_utils
import cooler
import numpy as np
file_path = jh_utils.mount_4dn_file("4DNFIB59T7NN")
c = cooler.Cooler(file_path + "::/resolutions/100000")
import pandas as pd
import csv
ddf = pd.read_csv("diff.csv",sep = ",", header = None)
dcols1 = []
dcols2 = []
dcols3 = []
dcols4 = []
result1 = []
resultList1 = []
for i in range(0,max(df[0])):
    dcols1.append(ddf[1][i])
    dcols2.append(ddf[2][i])
    dcols3.append(ddf[3][i])
    dcols4.append(ddf[4][i])

for p in range(len(dcols1)):
    A2 = c.matrix().fetch("chr" + str(dcols1[p]) + ":" + str(dcols2[p]) + "
                          "chr" + str(dcols3[p]) + ":" + str(dcols4[p]) + "
    if(str(np.log10(A2)[0][0]) != "nan" and str(np.log10(A2)[0][0]) != "-in
        result1.append(np.log10(A2)[0][0])
        print(p)
print(result1)
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
```

In [20]:

```
import math
print(type(result1[0]))
mean2 = np.mean(result1)
std2 = np.std(result1)
print(mean2)
print(std2)
```

```
<class 'numpy.float64'>
-4.5938672952573985
0.28196419401098993
```

In [25]:

```
b = mean2 + std2
print(b)
```

```
-4.311903101246409
```

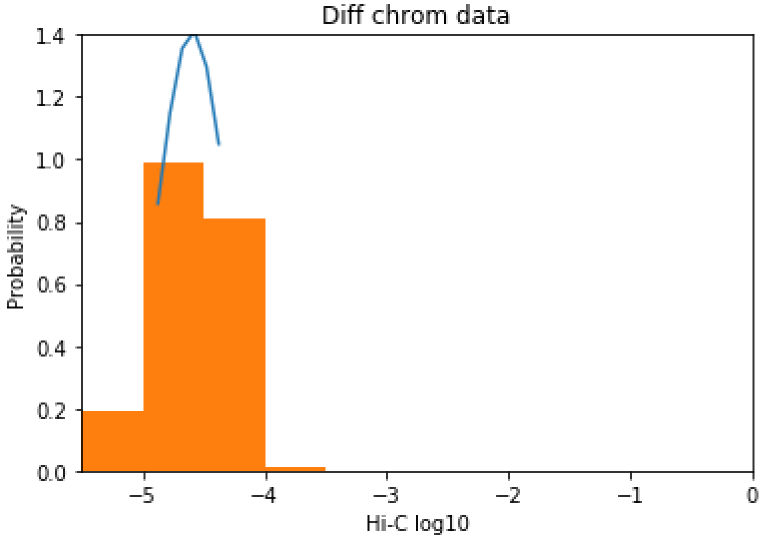
In [34]:

```
import matplotlib.pyplot as plt
def normfun(x,mu,sigma):
    pdf = np.exp(-((x - mu)**2)/(2*sigma**2)) / (sigma * np.sqrt(2*np.pi))
    return pdf

x2 = np.arange(-4.876,-4.312,0.1)
y2 = normfun(x2, mean2, std2)
plt.plot(x2,y2)

#画出直方图，最后的“normed”参数，是赋范的意思，数学概念
plt.hist(result1, bins=[-5.5, -5, -4.5, -4, -3.5, -3, -2.5, -2, -1.5, -1, -
plt.title('Diff chrom data')
plt.xlabel('Hi-C log10')
plt.ylabel('Probability')
plt.xlim((-5.5, 0))
plt.ylim((0, 1.4))
#输出
plt.show()
```

```
/opt/conda/lib/python3.6/site-packages/ipykernel_launcher.py:11: Matplotl
ibDeprecationWarning:
The 'normed' kwarg was deprecated in Matplotlib 2.1 and will be removed i
n 3.1. Use 'density' instead.
  # This is added back by InteractiveShellApp.init_path()
```



In [ ]: