subplot_HT1080

July 16, 2019

```
[1]: %matplotlib inline
     import seaborn as sns
     import bbi
     import numpy as np
     import pandas as pd
     import bioframe
     from pomegranate import *
```

```
[23]: import cooler
      import bioframe
      binsize=25000
      chromsizes = bioframe.fetch_chromsizes('hg38')
      df = cooler.util.binnify(chromsizes, binsize)
      df.head()
```

```
[23]:   chrom    start      end
     0   chr1        0    25000
     1   chr1    25000    50000
     2   chr1    50000    75000
     3   chr1    75000   100000
     4   chr1   100000   125000
```

```
[24]: import os

      directory = '/Users/gspracklin/Documents/OneDrive - New England Biolabs/LADseq/
       ↪binsize/HT1080/'
      os.chdir(directory)
      for file in os.listdir(directory):
          if '.bw' == file[-3:]:
              column=file.split('_')
              print(column[1])
              df[column[1]]= bbi.stackup(
                  file,
                  df.chrom,
                  df.start,
                  df.start+25000,
                  bins=1,
                  missing=np.nan
```

```
        )
```

```
H3K9me2
HP1
H3K9me3
Protect-seq
H3K27me3
DamID
Repli-seq
```

[25]: `df.head()`

[25]:
```
   chrom   start     end   H3K9me2       HP1   H3K9me3  Protect-seq  H3K27me3  \
0   chr1       0   25000 -0.335306 -0.246377 -0.221547     0.327338 -0.309055
1   chr1   25000   50000 -0.122812 -0.220896  0.047240    -0.083329  0.921302
2   chr1   50000   75000 -0.176748 -0.162694  0.085468    -0.047988  1.539394
3   chr1   75000  100000 -0.169201 -0.152300  0.065249     0.070210  1.394868
4   chr1  100000  125000 -0.151636 -0.081072  0.082136    -0.119120  1.305918

      DamID  Repli-seq
0 -0.973417   0.011044
1 -0.894692   0.066660
2 -0.665931   0.202384
3 -0.265305   0.188776
4 -1.795540   0.088114
```

[26]:
```python
def hmm(df,num_states):
    "HMM program"
    df['Protect-seq']=df['Protect-seq'].replace(0,np.nan) #this removes
 unmappable areas of chr
    df_dropna=df.dropna(subset=['Protect-seq']) #this removes unmappable areas
 of chr (NaN is otherwise considered 0)
    model = HiddenMarkovModel.
 from_samples(NormalDistribution,X=[df['Protect-seq'].values],
 n_components=num_states)
    states=model.viterbi(df['Protect-seq'].values)
    listofstates = [i[0] for i in states[1]]
    listofstates.pop(0) #first value isn't a state
    df['state']=listofstates
    return df
```

[27]:
```python
df=hmm(df, num_states=3)
df.head()
```

[27]:
```
   chrom   start     end   H3K9me2       HP1   H3K9me3  Protect-seq  H3K27me3  \
0   chr1       0   25000 -0.335306 -0.246377 -0.221547     0.327338 -0.309055
1   chr1   25000   50000 -0.122812 -0.220896  0.047240    -0.083329  0.921302
2   chr1   50000   75000 -0.176748 -0.162694  0.085468    -0.047988  1.539394
3   chr1   75000  100000 -0.169201 -0.152300  0.065249     0.070210  1.394868
```

```
4  chr1  100000  125000 -0.151636 -0.081072  0.082136     -0.119120  1.305918

        DamID  Repli-seq  state
0 -0.973417    0.011044       2
1 -0.894692    0.066660       2
2 -0.665931    0.202384       2
3 -0.265305    0.188776       2
4 -1.795540    0.088114       2
```

[28]:
```python
import matplotlib.pyplot as plt

state0=df.loc[df.state==0].replace(0,np.nan)
state1=df.loc[df.state==1].replace(0,np.nan)
state2=df.loc[df.state==2].replace(0,np.nan)
#state3=df_data[df_data['state']==3]

plt.subplot(4,4,7)
ax = sns.kdeplot(state0['H3K27me3'].dropna(),label='state 0',shade=True)
ax = sns.kdeplot(state1['H3K27me3'].dropna(),label='state 1',shade=True)
ax = sns.kdeplot(state2['H3K27me3'].dropna(),label='state 2',shade=True)
#ax = sns.kdeplot(state3['H3K27me3'].dropna(),label='state 3',shade=True)
plt.xlim(-2,2)
ax.set_title("H3K27me3")
plt.legend(bbox_to_anchor=(1, 1), loc=1, borderaxespad=0.5)

plt.subplot(4,4,5)
ax = sns.kdeplot(state0['H3K9me3'].dropna(),label='state 0',shade=True)
ax = sns.kdeplot(state1['H3K9me3'].dropna(),label='state 1',shade=True)
ax = sns.kdeplot(state2['H3K9me3'].dropna(),label='state 2',shade=True)
#ax = sns.kdeplot(state3['H3K9me3'].dropna(),label='state 3',shade=True)
plt.xlim(-1,1)
ax.set_title("H3K9me3")
ax.get_legend().remove()

plt.subplot(4,4,6)
ax = sns.kdeplot(state0['H3K9me2'].dropna(),label='state 0',shade=True)
ax = sns.kdeplot(state1['H3K9me2'].dropna(),label='state 1',shade=True)
ax = sns.kdeplot(state2['H3K9me2'].dropna(),label='state 2',shade=True)
#ax = sns.kdeplot(state3['H3K9me2'].dropna(),label='state 3',shade=True)
plt.xlim(-1,1)
ax.set_title("H3K9me2")
ax.get_legend().remove()

plt.subplot(4,4,1)
ax = sns.kdeplot(state0['Protect-seq'].dropna(),label='state 0',shade=True)
ax = sns.kdeplot(state1['Protect-seq'].dropna(),label='state 1',shade=True)
ax = sns.kdeplot(state2['Protect-seq'].dropna(),label='state 2',shade=True)
```
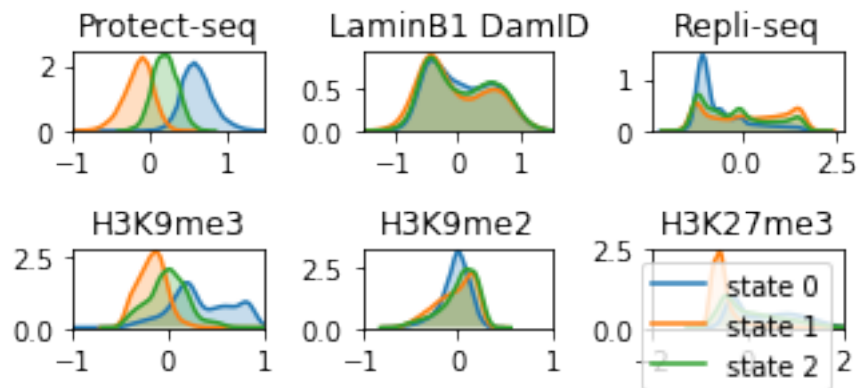
```
#ax = sns.kdeplot(state3['LADseq'].dropna(),label='state 3',shade=True)
plt.xlim(-1,1.5)
ax.set_title("Protect-seq")
ax.get_legend().remove()


plt.subplot(4,4,3)
ax = sns.kdeplot(state0['Repli-seq'].dropna(),label='state 0',shade=True)
ax = sns.kdeplot(state1['Repli-seq'].dropna(),label='state 1',shade=True)
ax = sns.kdeplot(state2['Repli-seq'].dropna(),label='state 2',shade=True)
#ax = sns.kdeplot(state3['repli'].dropna(),label='state 3',shade=True)
ax.set_title("Repli-seq")
ax.get_legend().remove()


plt.subplot(4,4,2)
ax = sns.kdeplot(state0['DamID'].dropna(),label='state 0',shade=True)
ax = sns.kdeplot(state1['DamID'].dropna(),label='state 1',shade=True)
ax = sns.kdeplot(state2['DamID'].dropna(),label='state 2',shade=True)
#ax = sns.kdeplot(state3['damID'].dropna(),label='state 3',shade=True)
plt.xlim(-1.5,1.5)
ax.set_title("LaminB1 DamID")
ax.get_legend().remove()



#lines, labels = ax.get_legend_handles_labels()
#plt.figlegend(lines, labels, loc = 'right', labelspacing=0.5 )
#plt.legend()
plt.tight_layout()
plt.savefig('HT1080_kdeplot_bystate_subplots.pdf')
```


[ ]: