heatmap of topics over features (from mixEHR outputs)

October 27, 2019

[ ]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
qn = pd.read_csv("../sorted_newpheno.csv")

cols1 =␣
 ↪["typeid","featureid","topic0","topic1","topic2","topic3","topic4","topic5","topic6","topic7"
# cols2 =␣
 ↪["varid","stateid","typeid","varid","topic0","topic1","topic2","topic3","topic4","topic5","to
phi_norm = pd.read_csv("/home/mcb/li_lab/wliu92/data3/Oct_25/
 ↪oct25_5col_JCVB0_nmar_K10_iter300_phi_normalized.csv",names = cols1)
# eta = pd.read_csv("/home/mcb/li_lab/wliu92/data3/tr_out/
 ↪training_5col_JCVB0_nmar_K10_iter300_eta_normalized.csv",names = cols1)
# psi_norm = pd.read_csv("/home/mcb/li_lab/wliu92/data3/tr_out/
 ↪training_5col_JCVB0_nmar_K10_iter300_psi.csv",names = cols1)
# phi_norm["typeid"].unique()
```

[ ]:
```
lst = []
for i in range (0,476,4):#loop each var
    large = -1
    largeid = i
    for l in range (i, i+4,1):
        tmp = eta.iloc[l,3]
        if (tmp>large):
            large = tmp
            largeid = l
        else:
            continue
    lst.append(largeid)#the largest row indexes
```

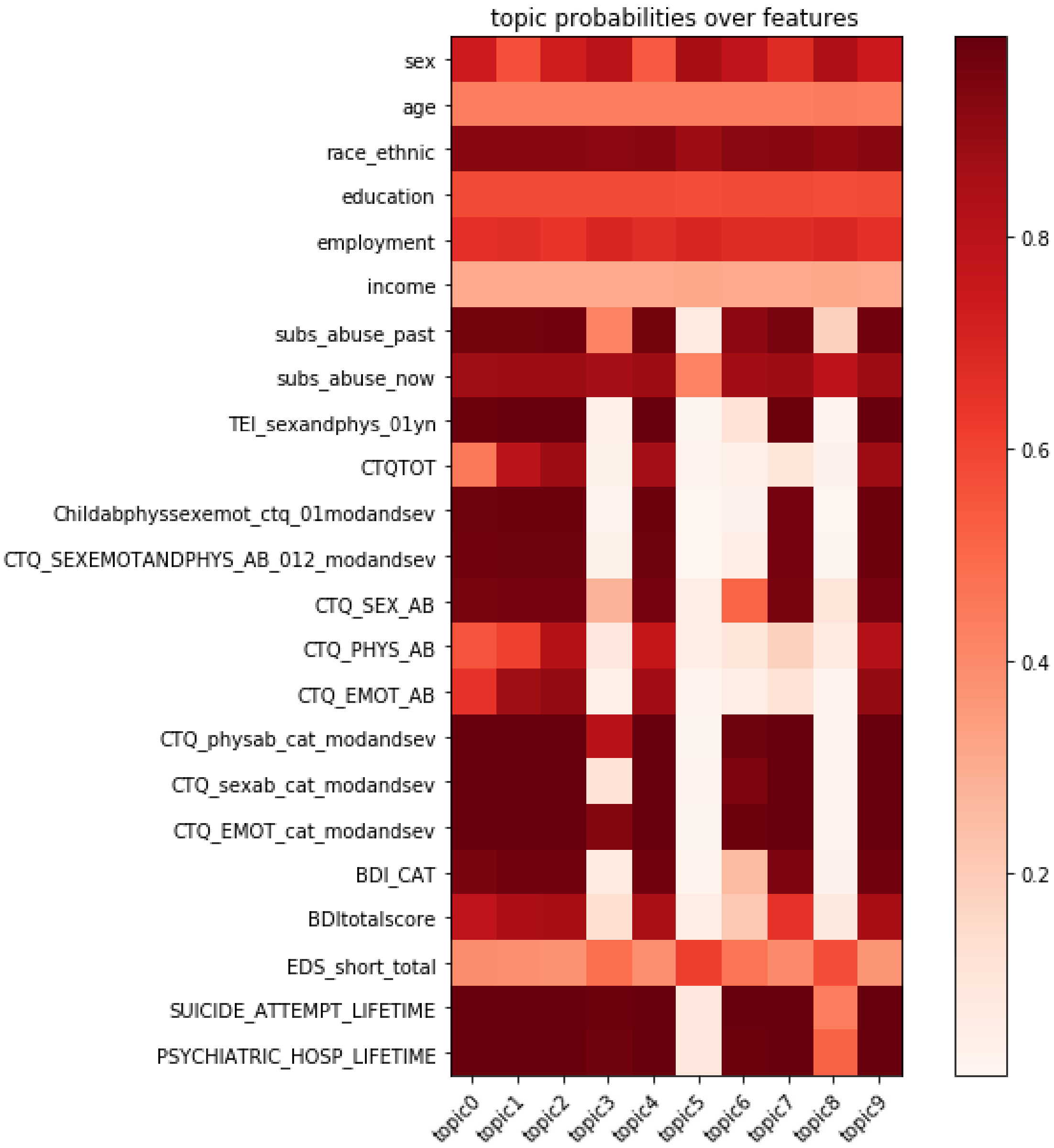
[ ]:
```
#lst is the lst of row indexes with persumably max prob for each topic
data = pd.DataFrame(eta.iloc[i,2:] for i in lst)
x = np.array([])
newlst = []
for i in range(0,119,1):
    count = 0
    for j in range(0,10,1):
        if data.iloc[i,j]==1:
            count +=1
#         else:
#             data.iloc[i,j] = 1/data.iloc[i,j]
    if count !=10:
        if x.size ==0:
            x = np.array(data.iloc[i])
            newlst.append(i+1)
        else:
            x = np.vstack((x,np.array(data.iloc[i])))
            newlst.append(i+1)
    else:
        continue
```

[396]:

```
y_labels = []
y_label_states = []
for i in newlst:
    y_labels.append(qn.iloc[:,i].name)
for l in lst:
    y_label_states.append(int (eta.iloc[l,1]))
# y_label_states
```

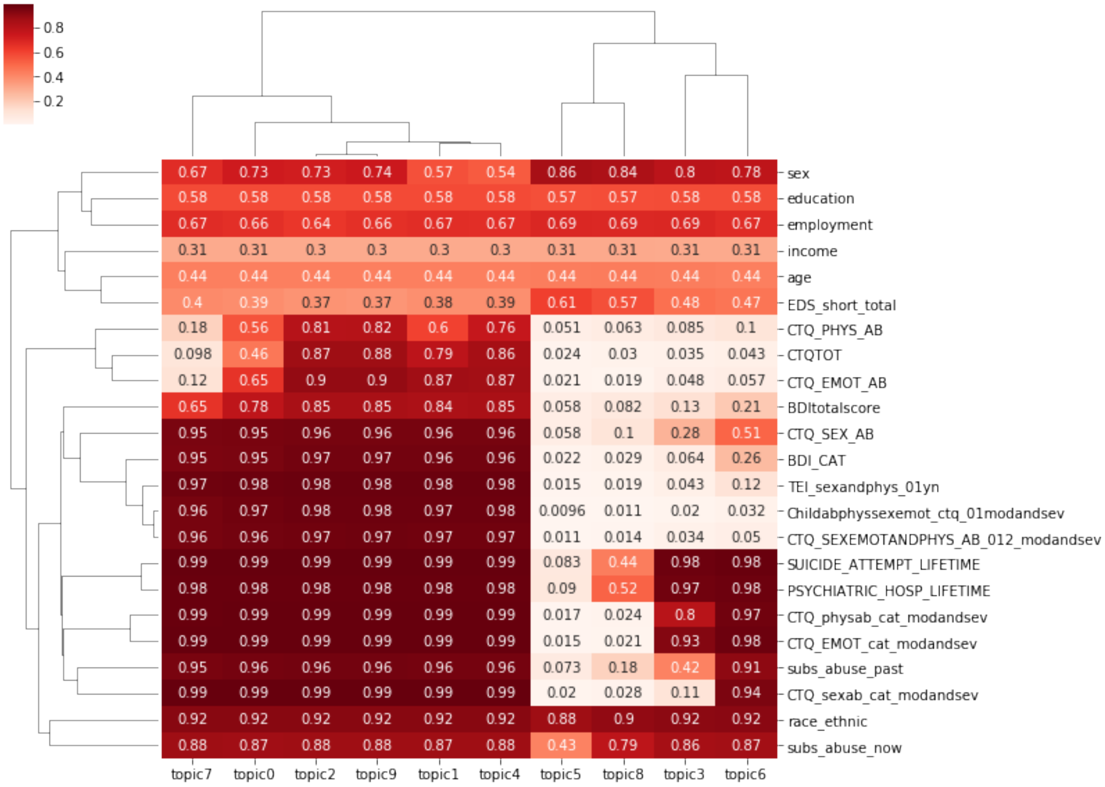
[475]:

```
cols =␣
 ↪["topic0","topic1","topic2","topic3","topic4","topic5","topic6","topic7","topic8","topic9"]
# features = [i for i in range(0,23,1)]
fig,ax = plt.subplots(figsize = (10,10))
im = ax.imshow(x,cmap = "Reds")
plt.colorbar(im)
ax.set_xticks(np.arange(0,10,step=1.0))
ax.set_yticks(np.arange(0,23,step=1.0))
ax.set_xticklabels(cols)
ax.set_yticklabels(y_labels)
plt.setp(ax.get_xticklabels(), rotation=45, ha = "right", rotation_mode =␣
 ↪"anchor")
ax.set_title("topic probabilities over features")
plt.show()
```



[553]:

```
import seaborn as sns
g = sns.clustermap(x, cmap = "Reds",row_cluster = True,col_cluster = True,␣
 ↪yticklabels = y_labels,xticklabels = cols,method = 'single',annot = True)
```



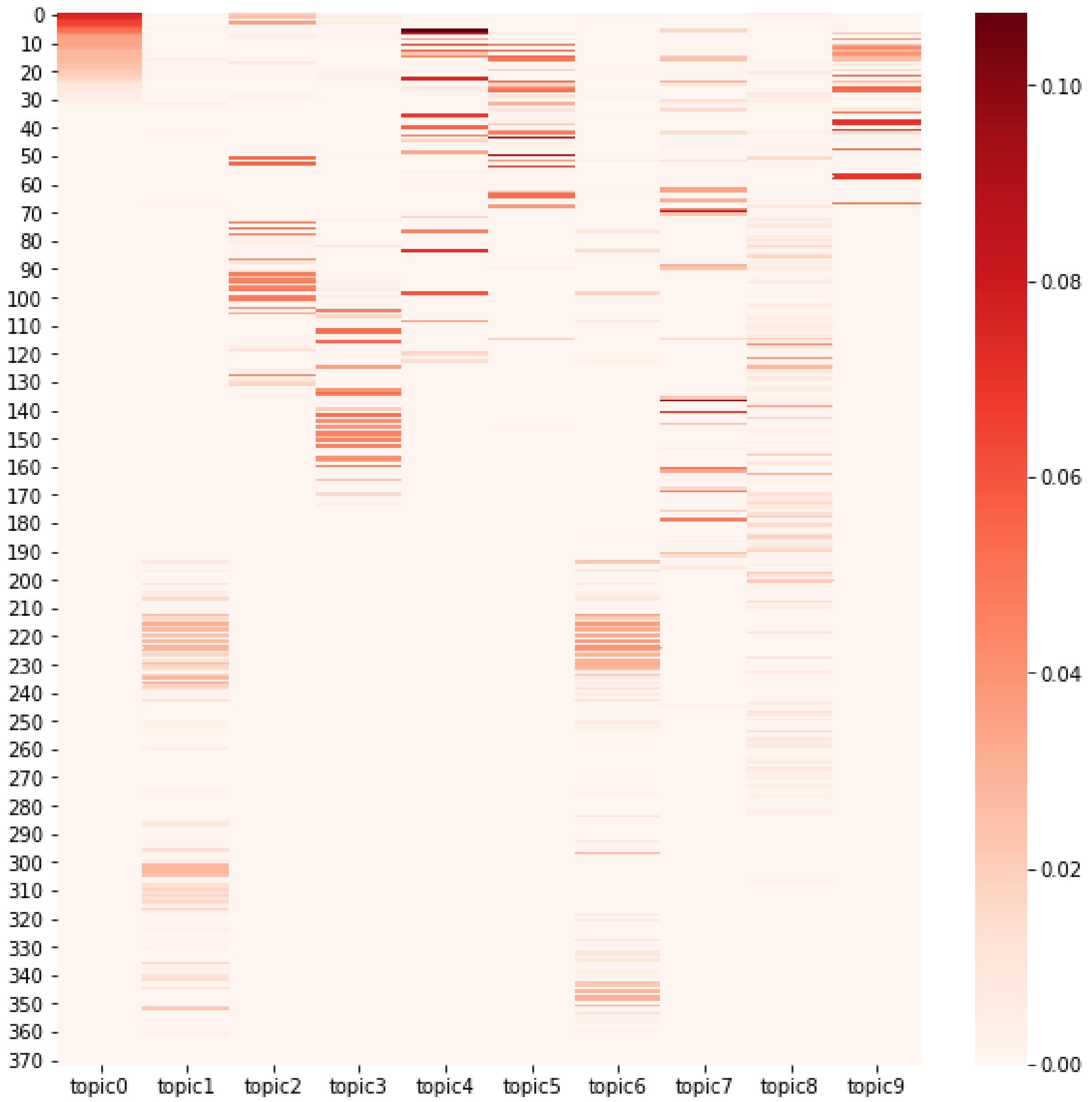
[609]:

```
# phi_norm.iloc[0:5]
```

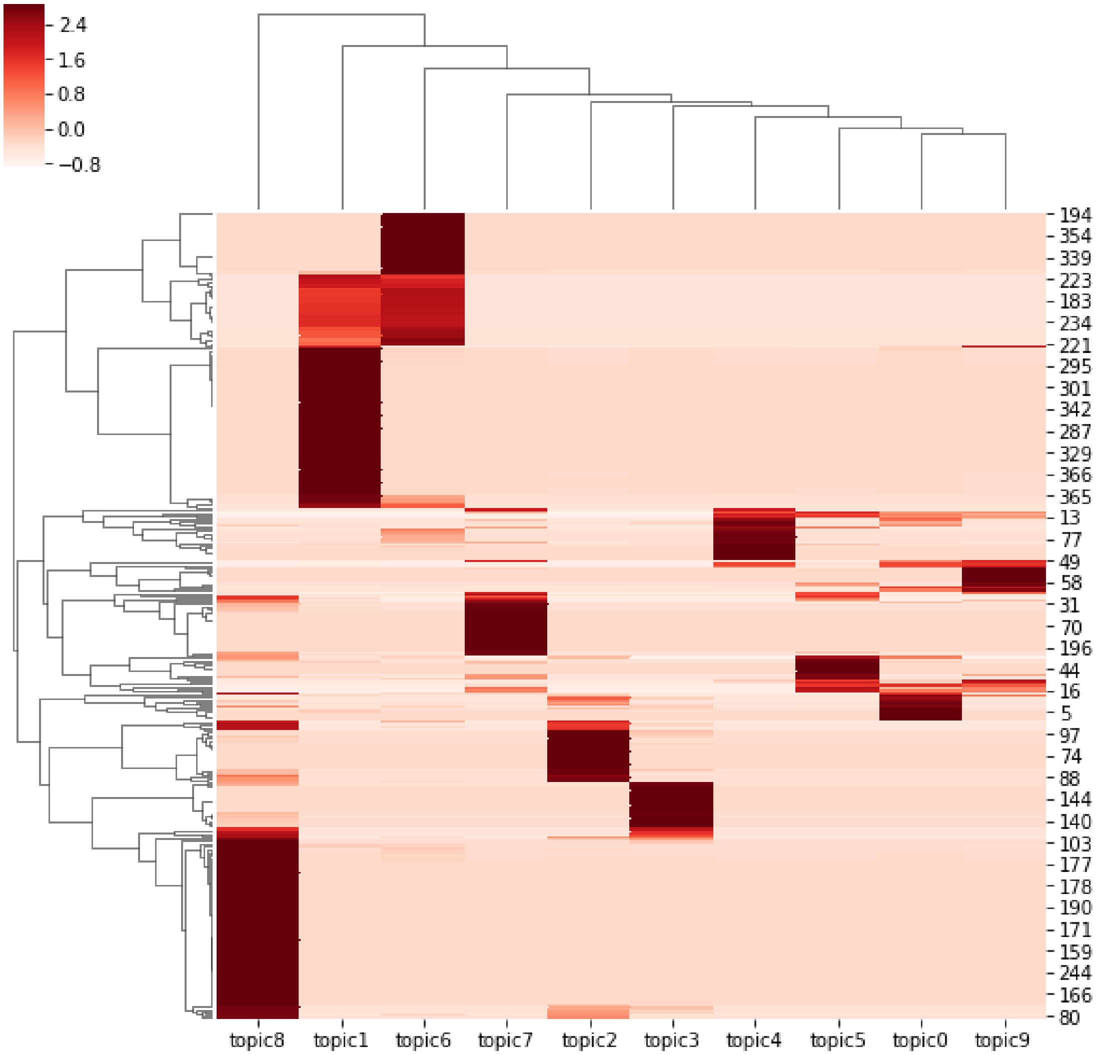
[622]:

```
new_arr = np.array(phi_norm.iloc[:,2:])

g1,ax1 = plt.subplots (1,1,figsize = (10,10))
g1 = sns.heatmap(new_arr, ax =ax1, xticklabels = cols, yticklabels=10, cmap =␣
 ↪"Reds",annot = False)
```



```
[669]: g1 = sns.clustermap(new_arr, xticklabels = cols, yticklabels=10, cmap = "Reds",␣
         ↪z_score = 0,method = "weighted",annot = False)
```


2.4
1.6
0.8
0.0
−0.8
194
354
339
223
183
234
221
295
301
342
287
329
366
365
13
77
49
58
31
70
196
44
16
5
97
74
88
144
140
103
177
178
190
171
159
244
166
80
topic8
topic1
topic6
topic7
topic2
topic3
topic4
topic5
topic0
topic9