Eat

In [1]:

```
import pandas as pd
eat = pd.read_csv('กิน.csv', index_col=0)
eat
```

Out[1]:

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G M
4:46:08 PM	-32.13760	1178.339000	329.339400	11148.340	-5621.0200	-7607.6900	1477	46
4:46:09 PM	27.13514	-595.279000	42.225230	11184.750	-5726.9400	-7554.8600	2873	56
4:46:10 PM	455.52680	-271.714000	149.839300	11437.730	-6413.0400	-6707.3100	4188	37
4:46:11 PM	125.00000	128.423400	201.225200	11531.870	-6622.7300	-6493.9200	2916	34
4:46:12 PM	-63.92040	-64.548700	-11.929200	11515.860	-6716.1100	-6370.9100	1851	38
4:46:13 PM	-237.76600	77.747750	-45.144100	11385.450	-6237.8600	-6856.5900	983	30
4:46:14 PM	38.86607	-174.536000	155.491100	11224.800	-6479.0400	-6961.1900	1370	23
4:46:15 PM	85.90991	-118.270000	-57.063100	11313.420	-6514.0800	-6853.7700	2049	23
4:46:16 PM	425.21100	561.412800	397.935800	11400.800	-6499.2600	-6783.6700	2628	27
4:47:58 PM	222.48860	402.261400	19.500000	11794.100	-574.5000	-8523.9400	930	15
4:47:59 PM	-96.62280	593.710500	274.868400	11726.590	100.4035	-8505.8200	1190	20
4:48:00 PM	111.10090	75.073390	125.486200	11782.540	236.1560	-8427.6600	1071	8
4:48:01 PM	-44.61400	369.061400	-46.271900	11727.350	541.0526	-8482.4000	735	9
4:48:02 PM	103.82460	46.429820	16.798250	11734.320	921.0526	-8507.3500	707	6
4:48:03 PM	89.65789	5.491228	83.938600	11747.230	1022.9300	-8421.4900	536	3
4:48:04 PM	74.29204	1.345133	41.185840	11754.520	1016.2740	-8447.9500	543	5
4:48:05 PM	66.32174	37.921740	89.252170	11784.190	945.7304	-8380.4000	476	3
4:48:06 PM	89.40909	20.000000	51.163640	11832.320	913.2909	-8373.6600	428	4
4:48:07 PM	-37.50000	-116.875000	63.375000	11834.880	1080.1250	-8274.8800	17	
4:49:47 PM	1575.33300	-394.667000	-8.000000	11951.670	-7370.0000	-5392.0000	1709	2
4:49:48 PM	-41.12500	-156.598000	29.276790	11721.190	-7281.5300	-5446.6700	1933	34
4:49:49 PM	-172.18000	95.459460	-53.441400	11683.200	-7109.6500	-5804.9500	849	27

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G M
4:49:50 PM	197.87610	368.991200	205.212400	11620.760	-6981.5800	-5890.4400	1395	28
4:49:51 PM	-173.01800	-223.793000	156.927900	11428.060	-7336.5600	-5904.3900	1242	31
4:49:52 PM	104.38940	29.955750	50.424780	11427.430	-7227.4800	-6038.0400	2237	32
4:49:53 PM	-243.44500	-180.145000	-130.182000	11449.220	-7257.1900	-6030.9900	1212	25
4:49:54 PM	130.03570	284.267900	149.794600	11385.780	-6717.8500	-6450.7900	1279	16
4:49:55 PM	-77.34580	-345.449000	-9.579440	11416.630	-6989.7500	-6352.2900	1250	23
4:49:56 PM	-369.68200	407.803700	-128.019000	11450.060	-6753.9800	-6518.6800	1397	31
7:45:00 AM	-242.42900	117.714300	-469.429000	2084.429	-7623.0000	-12138.3000	130	2
7:45:01 AM	-248.89400	34.946900	185.283200	2366.460	-7559.6700	-12366.0000	1559	16
7:45:02 AM	-4.39286	64.669640	97.098210	1921.143	-7535.7400	-12359.6000	3206	13
7:45:03 AM	171.59650	460.675400	315.815800	1473.632	-6843.4600	-12815.5000	3749	19
7:45:04 AM	338.42860	-350.420000	218.455400	1979.661	-6930.6200	-12684.1000	2631	4
7:45:05 AM	13.30088	4.522124	150.141600	2286.938	-7485.1900	-12313.3000	2006	17
7:45:06 AM	-566.67300	355.736400	-9.727270	1754.800	-7330.8200	-12561.2000	1173	21
7:45:07 AM	143.01850	-268.537000	-4.990740	1431.278	-7015.4400	-12761.2000	3957	10
7:45:08 AM	109.49550	17.585590	-89.045000	1596.387	-7381.1800	-12532.1000	2397	10
7:45:09 AM	311.05000	-85.330000	369.560000	1840.140	-7441.9700	-12525.3000	1842	9
7:46:51 AM	-143.25500	-62.212800	-685.798000	4387.681	-13596.4000	784.3936	428	1
7:46:52 AM	-27.96550	279.344800	5.974138	4160.940	-13704.2000	1127.4830	174	8
7:46:53 AM	126.56140	51.271930	81.640350	4290.518	-13613.6000	1455.7810	715	5
7:46:54 AM	106.70430	22.443480	20.886960	4406.774	-13612.6000	1410.9130	436	2
7:46:55 AM	92.43363	91.221240	60.707960	4540.894	-13543.2000	1475.8410	307	5
7:46:56 AM	67.48276	-9.137930	54.793100	4595.733	-13526.2000	1489.7590	259	3
7:46:57 AM	66.04425	45.725660	41.283190	4642.088	-13517.4000	1480.6730	258	4

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G M
7:46:58 AM	68.53448	49.982760	83.655170	4686.974	-13504.3000	1484.0780	269	3
7:46:59 AM	47.28155	43.825240	89.155340	4737.495	-13474.7000	1504.3110	263	4
11:38:12 AM	323.64910	96.649120	150.842100	2018.754	-5989.1100	-13235.1000	1792	6
11:38:13 AM	-143.58400	-104.841000	-54.327400	1889.327	-6054.2800	-13108.7000	1118	8
11:38:14 AM	-44.93640	-7.790910	53.954550	1320.655	-6111.7600	-13204.7000	2829	9
11:38:15 AM	298.65790	-29.377200	-18.412300	1578.202	-6334.3000	-13076.5000	1808	15
11:38:16 AM	-213.94600	95.348210	274.785700	1824.696	-5891.4500	-13139.4000	2987	18
11:38:17 AM	264.07890	8.043860	265.219300	1831.281	-6075.8400	-13149.4000	2247	11

54 rows × 42 columns

In [2]:

```
from sklearn.preprocessing import MinMaxScaler
mms = MinMaxScaler()
eat_scaled = pd.DataFrame(mms.fit_transform(eat),
                          columns=eat.columns,
                          index=eat.index)
eat_scaled
```

Out[2]:

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G2: Max	G
4:46:08 PM	0.249549	1.000000	0.936704	0.924435	0.546740	0.381793	0.350036	0.826740	0.
4:46:09 PM	0.277221	0.000000	0.671773	0.927860	0.539576	0.385378	0.684728	1.000000	0.
4:46:10 PM	0.477216	0.182432	0.771072	0.951657	0.493168	0.442880	1.000000	0.655422	0.
4:46:11 PM	0.322909	0.408037	0.818488	0.960512	0.478985	0.457357	0.695037	0.616743	0.
4:46:12 PM	0.234711	0.299236	0.621803	0.959006	0.472669	0.465703	0.439703	0.678206	0.
4:46:13 PM	0.153551	0.379465	0.591154	0.946739	0.505017	0.432752	0.231599	0.533027	0.
4:46:14 PM	0.282697	0.237223	0.776288	0.931627	0.488704	0.425655	0.324383	0.411692	0.
4:46:15 PM	0.304660	0.268947	0.580156	0.939963	0.486334	0.432943	0.487173	0.423349	0.
4:46:16 PM	0.463063	0.652165	1.000000	0.948183	0.487336	0.437699	0.625989	0.484281	0.
4:47:58 PM	0.368422	0.562432	0.650804	0.985178	0.888082	0.319630	0.218892	0.280996	0.
4:47:59 PM	0.219444	0.670375	0.886441	0.978828	0.933732	0.320859	0.281228	0.367185	0.
4:48:00 PM	0.316420	0.377958	0.748601	0.984091	0.942915	0.326162	0.252697	0.146591	0.
4:48:01 PM	0.243724	0.543714	0.590114	0.978899	0.963538	0.322448	0.172141	0.162310	0.
4:48:02 PM	0.313023	0.361808	0.648311	0.979555	0.989240	0.320756	0.165428	0.115860	0.
4:48:03 PM	0.306409	0.338726	0.710264	0.980769	0.996131	0.326581	0.124431	0.060226	0.
4:48:04 PM	0.299236	0.336388	0.670814	0.981455	0.995681	0.324786	0.126109	0.090957	0.
4:48:05 PM	0.295515	0.357011	0.715167	0.984246	0.990910	0.329369	0.110046	0.067997	0.
4:48:06 PM	0.306293	0.346906	0.680021	0.988773	0.988715	0.329826	0.098538	0.079830	0.
4:48:07 PM	0.247046	0.269733	0.691289	0.989014	1.000000	0.336528	0.000000	0.000000	0.
4:49:47 PM	1.000000	0.113109	0.625428	1.000000	0.428440	0.532118	0.405658	0.050336	0.
4:49:48 PM	0.245353	0.247337	0.659825	0.978320	0.434424	0.528408	0.459362	0.606323	0.
4:49:49 PM	0.184170	0.389452	0.583498	0.974747	0.446050	0.504101	0.199473	0.487637	0.

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G2: Max	G
4:49:50 PM	0.356931	0.543674	0.822167	0.968873	0.454713	0.498301	0.330376	0.512363	0.
4:49:51 PM	0.183779	0.209451	0.777613	0.950747	0.430702	0.497354	0.293695	0.558107	0.
4:49:52 PM	0.313287	0.352519	0.679339	0.950688	0.438080	0.488287	0.532246	0.580714	0.
4:49:53 PM	0.150900	0.234061	0.512687	0.952737	0.436071	0.488765	0.286502	0.456729	0.
4:49:54 PM	0.325260	0.495905	0.771031	0.946770	0.472551	0.460284	0.302565	0.294242	0.
4:49:55 PM	0.228443	0.140859	0.623971	0.949672	0.454160	0.466966	0.295613	0.414694	0.
4:49:56 PM	0.091966	0.565557	0.514683	0.952816	0.470107	0.455678	0.330856	0.550336	0.
7:45:00 AM	0.151374	0.401999	0.199651	0.071844	0.411328	0.074413	0.027092	0.044331	0.
7:45:01 AM	0.148356	0.355334	0.803778	0.098373	0.415611	0.058964	0.369696	0.290887	0.
7:45:02 AM	0.262502	0.372092	0.722406	0.056485	0.417230	0.059399	0.764565	0.247086	0.
7:45:03 AM	0.344663	0.595367	0.924225	0.014390	0.464055	0.028468	0.894749	0.341575	0.
7:45:04 AM	0.422549	0.138056	0.834387	0.061989	0.458160	0.037383	0.626708	0.079654	0.
7:45:05 AM	0.270762	0.338179	0.771351	0.090893	0.420649	0.062540	0.476864	0.307135	1.
7:45:06 AM	0.000000	0.536201	0.623835	0.040838	0.431090	0.045721	0.277152	0.372307	0.
7:45:07 AM	0.331321	0.184223	0.628205	0.010406	0.452422	0.032152	0.944618	0.183151	0.
7:45:08 AM	0.315671	0.345545	0.550645	0.025937	0.427684	0.047695	0.570607	0.182091	0.
7:45:09 AM	0.409767	0.287519	0.973817	0.048865	0.423572	0.048157	0.437545	0.162487	0.
7:46:51 AM	0.197674	0.300553	0.000000	0.288498	0.007292	0.951157	0.098538	0.032851	0.
7:46:52 AM	0.251497	0.493130	0.638323	0.267170	0.000000	0.974434	0.037641	0.152066	0.
7:46:53 AM	0.323638	0.364538	0.708143	0.279358	0.006128	0.996707	0.167346	0.092194	0.
7:46:54 AM	0.314368	0.348284	0.652084	0.290294	0.006196	0.993663	0.100456	0.043448	0.
7:46:55 AM	0.307705	0.387062	0.688828	0.302910	0.010890	0.998068	0.069528	0.089544	0.
7:46:56 AM	0.296057	0.330478	0.683370	0.308068	0.012040	0.999013	0.058020	0.057224	0.
7:46:57 AM	0.295385	0.361411	0.670904	0.312429	0.012635	0.998396	0.057780	0.073119	0.

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G2: Max	G
7:46:58 AM	0.296548	0.363811	0.710002	0.316651	0.013521	0.998627	0.060417	0.065348	0.
7:46:59 AM	0.286626	0.360339	0.715077	0.321403	0.015523	1.000000	0.058979	0.075062	0.
11:38:12 AM	0.415649	0.390122	0.771998	0.065666	0.521843	0.000000	0.425557	0.121688	0.
11:38:13 AM	0.197520	0.276518	0.582681	0.053492	0.517435	0.008576	0.263965	0.151183	0.
11:38:14 AM	0.243574	0.331237	0.682596	0.000000	0.513547	0.002062	0.674179	0.177323	0.
11:38:15 AM	0.403982	0.319066	0.615821	0.024226	0.498494	0.010760	0.429393	0.268810	0.
11:38:16 AM	0.164671	0.389389	0.886365	0.047412	0.528448	0.006493	0.712059	0.331155	0.
11:38:17 AM	0.387838	0.340165	0.877538	0.048032	0.515976	0.005814	0.534644	0.205758	0.

54 rows × 42 columns

In [3]:

```
eat_scaled.to_csv('eat_scaled.csv')
```

In [4]:

```
from sklearn.cluster import KMeans
cls = KMeans(n_clusters=2, n_jobs=-1)
cls.fit(eat_scaled)
```

Out[4]:

```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
       n_clusters=2, n_init=10, n_jobs=-1, precompute_distances='auto',
       random_state=None, tol=0.0001, verbose=0)
```

In [6]:

```
centroid = pd.DataFrame(cls.cluster_centers_, columns=eat.columns)

import seaborn as sns
sns.heatmap(centroid)

print(centroid)
centroid.to_csv('eat_centroid.csv')
```

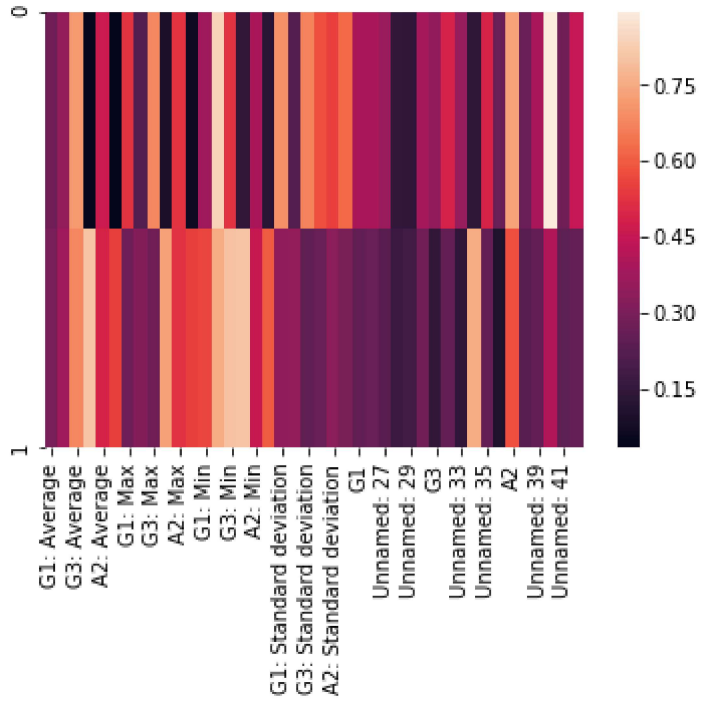
```
   G1: Average  G2: Average  G3: Average  A1: Average  A2: Average  \
0     0.279387     0.350063     0.715581     0.047428     0.463596
1     0.298176     0.373529     0.678848     0.806130     0.493079

   A3: Average   G1: Max   G2: Max   G3: Max   A1: Max  ...  Unnamed: 33  \
0     0.033037  0.526837  0.216664  0.680916  0.096360  ...     0.358697
1     0.551301  0.277190  0.320888  0.283945  0.729703  ...     0.141213

         A1  Unnamed: 35  Unnamed: 36        A2  Unnamed: 38  Unnamed: 39  \
0  0.138548     0.491795     0.261892  0.732985     0.265540     0.398564
1  0.752171     0.253067     0.101936  0.575900     0.228933     0.249501

         A3  Unnamed: 41  Unnamed: 42
0  0.893817     0.284283      0.44716
1  0.406904     0.243461      0.25739

[2 rows x 42 columns]
```



In [17]:

```
x1 = eat
x1['eat'] = cls.predict(eat_scaled)
x1['eat']
#y.to_csv('eat_cluster.csv')
```

Out[17]:

```
Time window
4:46:08 PM     1
4:46:09 PM     1
4:46:10 PM     1
4:46:11 PM     1
4:46:12 PM     1
4:46:13 PM     1
4:46:14 PM     1
4:46:15 PM     1
4:46:16 PM     1
4:47:58 PM     1
4:47:59 PM     1
4:48:00 PM     1
4:48:01 PM     1
4:48:02 PM     1
4:48:03 PM     1
4:48:04 PM     1
4:48:05 PM     1
4:48:06 PM     1
4:48:07 PM     1
4:49:47 PM     1
4:49:48 PM     1
4:49:49 PM     1
4:49:50 PM     1
4:49:51 PM     1
4:49:52 PM     1
4:49:53 PM     1
4:49:54 PM     1
4:49:55 PM     1
4:49:56 PM     1
7:45:00 AM     0
7:45:01 AM     0
7:45:02 AM     0
7:45:03 AM     0
7:45:04 AM     0
7:45:05 AM     0
7:45:06 AM     0
7:45:07 AM     0
7:45:08 AM     0
7:45:09 AM     0
7:46:51 AM     1
7:46:52 AM     1
7:46:53 AM     1
7:46:54 AM     1
7:46:55 AM     1
7:46:56 AM     1
7:46:57 AM     1
7:46:58 AM     1
7:46:59 AM     1
11:38:12 AM    0
11:38:13 AM    0
11:38:14 AM    0
11:38:15 AM    0
11:38:16 AM    0
11:38:17 AM    0
Name: eat, dtype: int32
```

Sleep

In [34]:

```
import pandas as pd
sleep = pd.read_csv('นอน2.csv', index_col=0)
sleep.head()
```

Out[34]:

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	M
21:14:42	-69.823529	-62.411765	82.509804	-14326.29412	6080.686275	-3067.784314	897	
21:14:43	49.026549	61.769911	51.053097	-14398.26549	6024.575221	-2892.274336	715	
21:14:44	-11.008929	111.133929	42.160714	-14336.19643	6055.830357	-2882.687500	491	1
21:14:45	81.230089	39.681416	49.938053	-14415.68142	6065.858407	-2768.442478	678	1
21:14:46	34.574074	89.935185	68.046296	-14411.21296	6086.916667	-2781.870370	695	

5 rows × 42 columns

In [35]:

```
from sklearn.preprocessing import MinMaxScaler
mms2 = MinMaxScaler()
sleep_scaled = pd.DataFrame(mms2.fit_transform(sleep),
                            columns=sleep.columns,
                            index=sleep.index)
sleep_scaled
```

Out[35]:

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G2: Max	G
21:14:42	0.704936	0.629805	0.271588	0.015731	0.092594	0.127331	0.105499	0.011716	0.(
21:14:43	0.733089	0.648158	0.267222	0.005688	0.086572	0.138476	0.086065	0.052117	0.(
21:14:44	0.718868	0.655454	0.265988	0.014349	0.089926	0.139085	0.062146	0.089932	0.(
21:14:45	0.740717	0.644894	0.267067	0.003258	0.091003	0.146339	0.082114	0.067146	0.(
21:14:46	0.729665	0.652321	0.269580	0.003881	0.093263	0.145487	0.083930	0.028846	0.(
21:14:47	0.730259	0.648707	0.264726	0.003115	0.090913	0.148780	0.047304	0.036522	0.(
21:14:48	0.731930	0.641675	0.272486	0.014921	0.098930	0.145273	0.059797	0.014544	0.(
21:14:49	0.747287	0.646478	0.264267	0.000000	0.094538	0.146928	0.061933	0.001535	0.(
21:14:50	0.723558	0.638633	0.266190	0.015808	0.094039	0.141943	0.059370	0.000000	0.(
21:14:51	0.735416	0.642494	0.272210	0.007792	0.095408	0.139999	0.070796	0.008161	0.(
21:14:52	0.740379	0.640970	0.260972	0.008883	0.089880	0.139832	0.054885	0.009615	0.(
21:14:53	0.725980	0.635800	0.273224	0.020534	0.094069	0.133991	0.053070	0.016484	0.(
21:14:54	0.740399	0.642428	0.272940	0.010132	0.090198	0.132651	0.063534	0.019958	0.(
21:14:55	0.693027	0.713863	0.275287	0.011995	0.094191	0.133388	0.046877	0.130414	0.(
21:14:56	0.000000	0.693081	1.000000	0.347350	0.520492	0.285110	0.000000	0.555672	1.(
21:14:57	0.881258	0.873479	0.121942	0.928006	1.000000	0.586338	1.000000	0.885343	0.4
21:14:58	0.970562	1.000000	0.353107	0.846090	0.908590	0.704036	0.792632	0.703458	0.9
21:14:59	0.507407	0.646227	0.279053	0.753627	0.777719	0.839422	0.583876	0.571509	0.3
21:15:00	0.785685	0.780958	0.217382	0.904848	0.721664	0.957156	0.459797	0.438995	0.7
21:15:01	0.820946	0.801729	0.308601	0.829437	0.662409	0.973494	0.716177	0.455640	0.4
21:15:02	0.543586	0.487924	0.326906	0.797067	0.667547	0.970315	0.839402	0.467518	0.4
21:15:03	0.718153	0.615488	0.240267	0.994271	0.714378	0.948851	0.676028	0.479719	0.
21:15:04	0.751615	0.876815	0.374953	1.000000	0.716225	1.000000	0.732408	0.398028	0.4
21:15:05	0.909825	0.460023	0.182614	0.866015	0.707011	0.966704	0.938281	0.337104	0.7
21:15:06	0.768190	0.199056	0.111632	0.844429	0.785671	0.880753	0.740310	0.167825	0.7
21:15:07	1.000000	0.000000	0.000000	0.352668	0.345493	0.033988	0.809397	0.500081	0.7
21:15:08	0.861637	0.912193	0.174741	0.183147	0.175017	0.000000	0.337640	0.388009	0.7
21:15:09	0.734950	0.629827	0.215360	0.154420	0.187828	0.035138	0.150347	0.110941	0.7
21:15:10	0.693545	0.626559	0.267845	0.121998	0.132475	0.029475	0.150667	0.043633	0.(
21:15:11	0.755934	0.623575	0.279610	0.110613	0.106552	0.025654	0.200427	0.056561	0.(
21:15:12	0.708807	0.610947	0.272876	0.151607	0.126092	0.011601	0.158356	0.036926	0.(
21:15:13	0.729016	0.648820	0.249479	0.120977	0.104975	0.020214	0.182381	0.052683	0.(
21:15:14	0.731094	0.662990	0.266590	0.122512	0.103504	0.014945	0.189642	0.081125	0.(
21:15:15	0.718548	0.635933	0.272208	0.116743	0.101014	0.026191	0.088735	0.065368	0.(
21:15:16	0.741060	0.650493	0.263648	0.114139	0.098067	0.023778	0.160171	0.072317	0.(

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G2: Max	G
21:15:17	0.723851	0.637155	0.265738	0.103589	0.097712	0.032829	0.125147	0.084842	0.(
21:15:18	0.783371	0.659999	0.256138	0.096731	0.082056	0.024946	0.154618	0.032240	0.(
21:15:19	0.706865	0.620731	0.277305	0.100730	0.088822	0.031743	0.107528	0.022544	0.(
21:15:20	0.739630	0.640761	0.267140	0.105412	0.074157	0.023813	0.157288	0.073368	0.(
21:15:21	0.723669	0.666599	0.292058	0.111055	0.101349	0.023341	0.103257	0.085569	0.(
21:15:22	0.710554	0.711641	0.194332	0.139416	0.110971	0.023727	0.142659	0.321671	0.(
21:15:23	0.792237	0.645538	0.237804	0.002514	0.000000	0.060295	0.479552	0.449176	0.(
21:15:24	0.715932	0.632958	0.364225	0.073973	0.081098	0.054966	0.109770	0.081448	0.
21:15:25	0.676793	0.603532	0.326753	0.184189	0.210767	0.033533	0.129632	0.080398	0.(
21:15:26	0.728249	0.610085	0.262665	0.162325	0.216602	0.043296	0.100267	0.032482	0.(
21:15:27	0.731834	0.634623	0.264256	0.161615	0.186738	0.031697	0.052109	0.017292	0.(
21:15:28	0.727632	0.631779	0.257950	0.146568	0.171011	0.031466	0.075814	0.030785	0.(
21:15:29	0.730286	0.647625	0.263845	0.144460	0.169777	0.030080	0.137960	0.029008	0.(
21:15:30	0.758051	0.634401	0.292133	0.141647	0.173825	0.026885	0.241751	0.143665	0.(
21:15:31	0.583842	0.834169	0.380509	0.313926	0.420669	0.055500	0.919594	1.000000	0.4

50 rows × 42 columns

In [20]:

```
sleep_scaled.to_csv('sleep_scaled.csv')
```

In [36]:

```
from sklearn.cluster import KMeans
cls = KMeans(n_clusters=2, n_jobs=-1)
cls.fit(sleep_scaled)
```

Out[36]:

```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
       n_clusters=2, n_init=10, n_jobs=-1, precompute_distances='auto',
       random_state=None, tol=0.0001, verbose=0)
```

In [37]:

```
centroid = pd.DataFrame(cls.cluster_centers_, columns=sleep.columns)

import seaborn as sns
sns.heatmap(centroid)

print(centroid)
centroid.to_csv('sleep_centroid.csv')
```

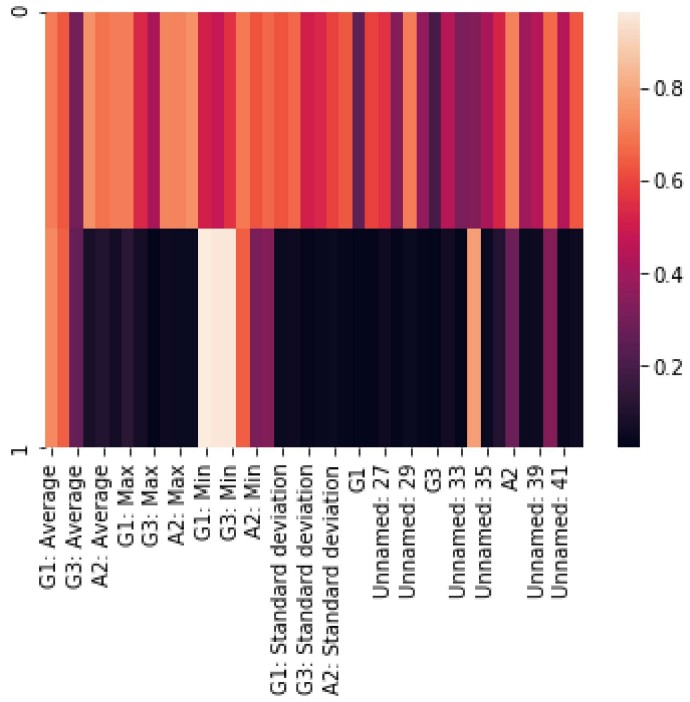
```
   G1: Average  G2: Average  G3: Average  A1: Average  A2: Average  \
0     0.710851     0.636073     0.299767     0.752133     0.688298
1     0.732947     0.650282     0.266174     0.081256     0.113404

   A3: Average   G1: Max   G2: Max   G3: Max   A1: Max  ...  Unnamed: 33  \
0     0.707821  0.708300  0.535453  0.418684  0.726111  ...     0.317399
1     0.070787  0.126298  0.077812  0.024981  0.053323  ...     0.026325

         A1  Unnamed: 35  Unnamed: 36        A2  Unnamed: 38  Unnamed: 39  \
0  0.335469     0.421561     0.524631  0.717466     0.397304     0.444904
1  0.779789     0.048985     0.109164  0.280668     0.049692     0.057456

         A3  Unnamed: 41  Unnamed: 42
0  0.668250     0.443310     0.631766
1  0.328821     0.043978     0.058269

[2 rows x 42 columns]
```



In [38]:

```
x2 = sleep
x2['sleep'] = cls.predict(sleep_scaled)
x2['sleep']
x2.to_csv('sleep_cluster.csv')
```

รวม

In [42]:

```
import pandas as pd
test = pd.read_csv('รวม.csv', index_col=0)
test.head()
```

Out[42]:

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G2: Max	G3: Max	
4:46:08 PM	-32.13760	1178.3390	329.33940	11148.34	-5621.02	-7607.69	1477	4674	1780	12
4:46:09 PM	27.13514	-595.2790	42.22523	11184.75	-5726.94	-7554.86	2873	5655	2584	12
4:46:10 PM	455.52680	-271.7140	149.83930	11437.73	-6413.04	-6707.31	4188	3704	1875	13
4:46:11 PM	125.00000	128.4234	201.22520	11531.87	-6622.73	-6493.92	2916	3485	1121	13
4:46:12 PM	-63.92040	-64.5487	-11.92920	11515.86	-6716.11	-6370.91	1851	3833	1419	13

5 rows × 42 columns

In [43]:

```
from sklearn.preprocessing import MinMaxScaler
mms = MinMaxScaler()
test_scaled = pd.DataFrame(mms.fit_transform(test),
                           columns=test.columns,
                           index=test.index)
test_scaled
```

Out[43]:

Time window	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	A3: Average	G1: Max	G2: Max	G:
4:46:08 PM	0.652145	0.813185	0.305844	0.969560	0.286230	0.235356	0.167432	0.370127	0.1
4:46:09 PM	0.664971	0.551048	0.265997	0.970940	0.282479	0.237565	0.316498	0.447695	0.1
4:46:10 PM	0.757675	0.598870	0.280932	0.980526	0.258184	0.273012	0.456914	0.293429	0.1
4:46:11 PM	0.686149	0.658010	0.288064	0.984093	0.250759	0.281937	0.321089	0.276113	0.0
4:46:12 PM	0.645267	0.629489	0.258481	0.983486	0.247452	0.287082	0.207368	0.303629	0.1
...	...	...	...	...	...	...	...	...	
9:15:27 PM	0.668562	0.634623	0.264256	0.043885	0.731657	0.362241	0.052109	0.038349	0.0
9:15:28 PM	0.664724	0.631779	0.257950	0.039799	0.726468	0.362088	0.075814	0.051554	0.0
9:15:29 PM	0.667148	0.647625	0.263845	0.039227	0.726061	0.361175	0.137960	0.049814	0.0
9:15:30 PM	0.692512	0.634401	0.292133	0.038463	0.727396	0.359071	0.241751	0.162015	0.0
9:15:31 PM	0.533365	0.834169	0.380509	0.085244	0.808845	0.377918	0.919594	1.000000	0.4

104 rows × 42 columns

In [44]:

```
test_scaled.to_csv('test_scaled.csv')
```

In [45]:

```
from sklearn.cluster import KMeans
cls = KMeans(n_clusters=2, n_jobs=-1)
cls.fit(test_scaled)
```

Out[45]:

```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
       n_clusters=2, n_init=10, n_jobs=-1, precompute_distances='auto',
       random_state=None, tol=0.0001, verbose=0)
```

In [46]:

```
centroid = pd.DataFrame(cls.cluster_centers_, columns=test.columns)

import seaborn as sns
sns.heatmap(centroid)

print(centroid)
centroid.to_csv('test_centroid.csv')
```

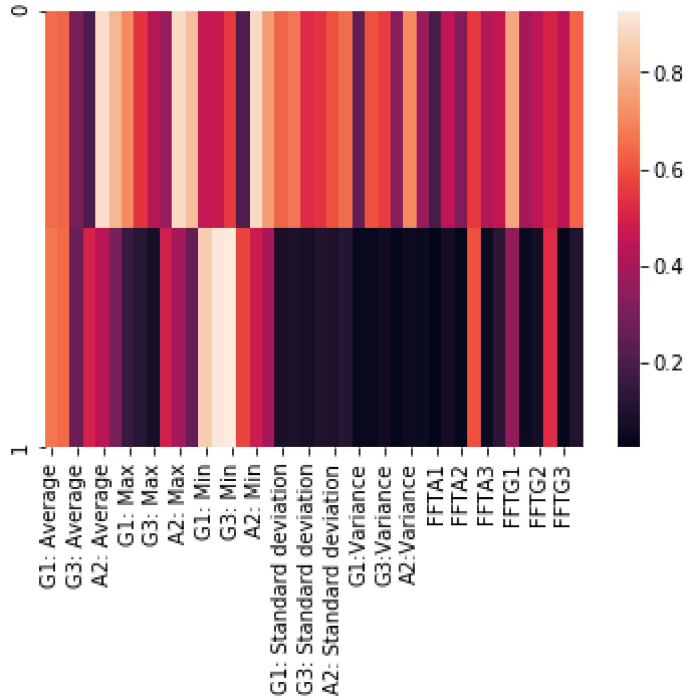
	G1: Average	G2: Average	G3: Average	A1: Average	A2: Average	\
0	0.649394	0.636073	0.299767	0.204236	0.897151	
1	0.671077	0.648418	0.267672	0.502298	0.438115	

	A3: Average	G1: Max	G2: Max	G3: Max	A1: Max	...	FFTA2	\
0	0.807560	0.708300	0.545408	0.434930	0.372262	...	0.317569	
1	0.303248	0.151004	0.116717	0.077763	0.493107	...	0.036351	

	A2	FFTA3	A3	FFTG1	G1	FFTG2	G2	\
0	0.556071	0.421561	0.457869	0.768940	0.397304	0.444904	0.505803	
1	0.599247	0.052642	0.128634	0.357488	0.050594	0.076238	0.526276	

	FFTG3	G3
0	0.443310	0.631766
1	0.038586	0.096238

[2 rows x 42 columns]



In [48]:

```
x = test
x['test'] = cls.predict(test_scaled)
x['test']
#x.to_csv('test_cluster.csv')
```

Out[48]:

```
Time window
4:46:08 PM    1
4:46:09 PM    1
4:46:10 PM    1
4:46:11 PM    1
4:46:12 PM    1
              ..
9:15:27 PM    1
9:15:28 PM    1
9:15:29 PM    1
9:15:30 PM    1
9:15:31 PM    0
Name: test, Length: 104, dtype: int32
```

In [ ]:

```

```