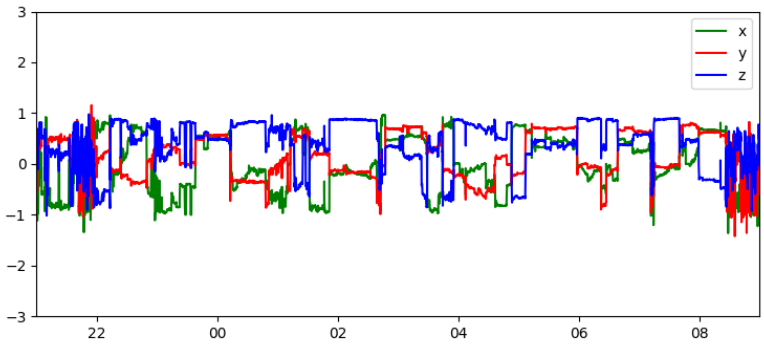
Usato **past_code** e modificato per plottare x,y,z variations:

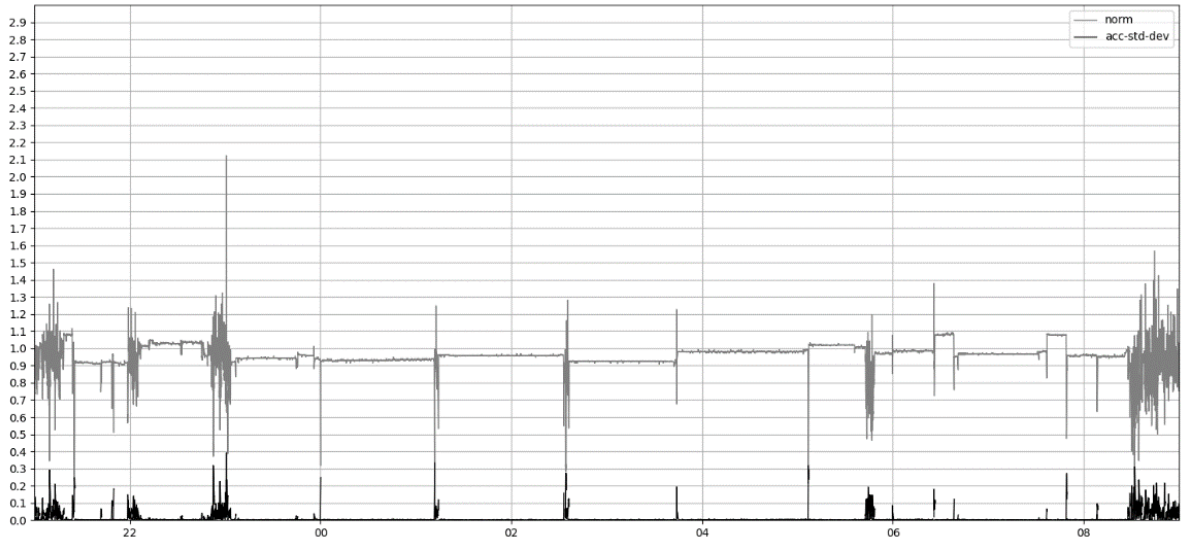
(il plot qui a destra è relativo a un soggetto diverso ripetto ai plot mostrati sotto)

Calcoli fatti con 30seconds window su dati a 200Hz estratti in chunks da 10000

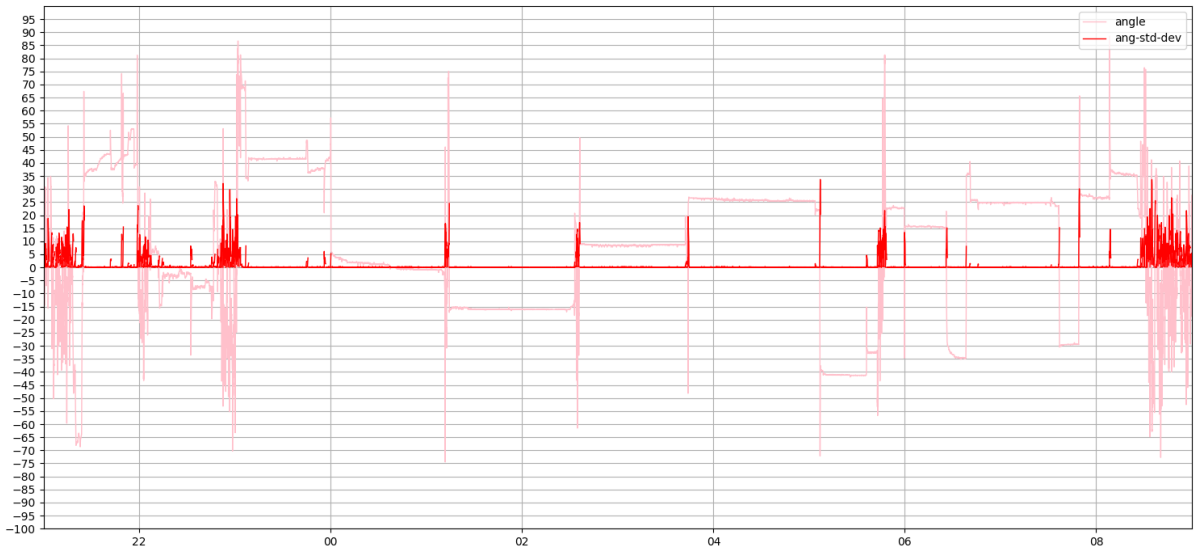


Analizzate features per trovare le più indicative:

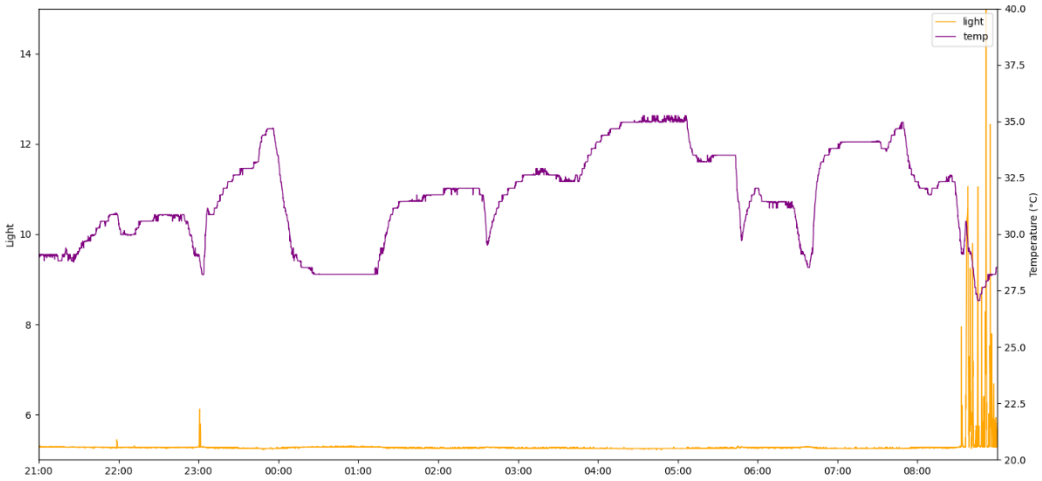
Accelerations Norm e sua standard deviation →



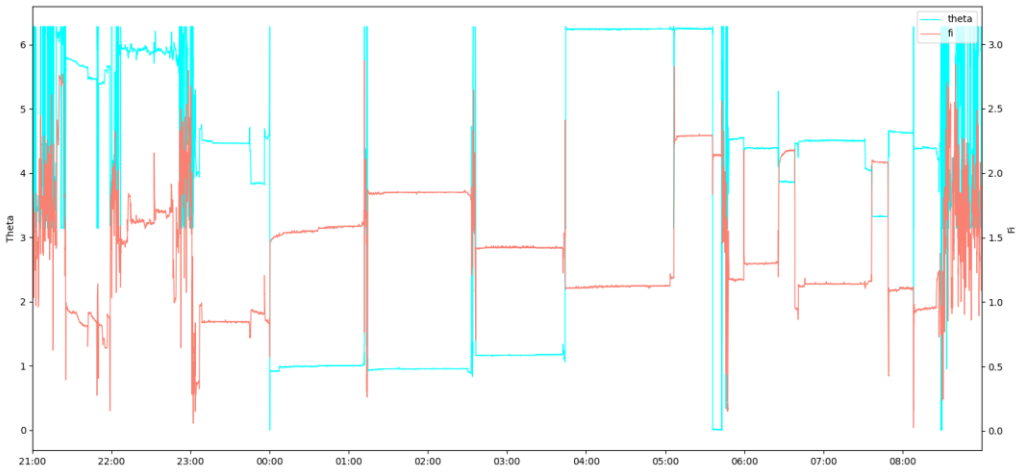
Arm Angle e sua standard deviation →



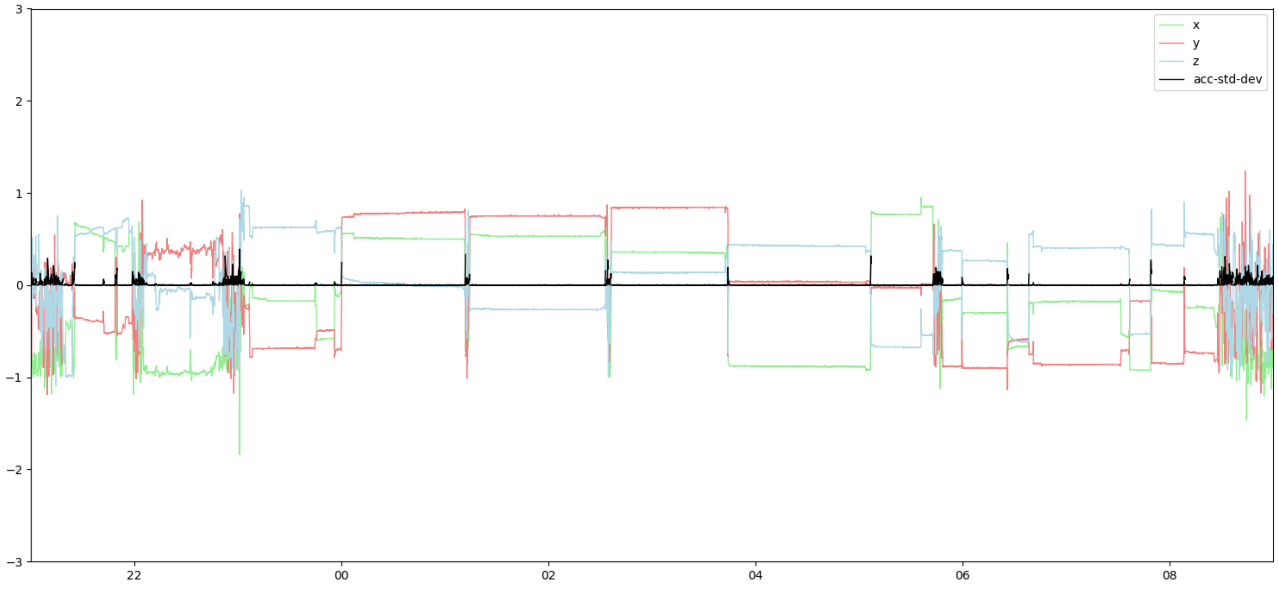
Temperatura e Luce →



Aggiunto calcolo coordinate polari e plottate per visualizzare variazioni →



Aggiunta acc-dev-std al plot di ax,ay,az e aggiustamenti alla visualizzazione:

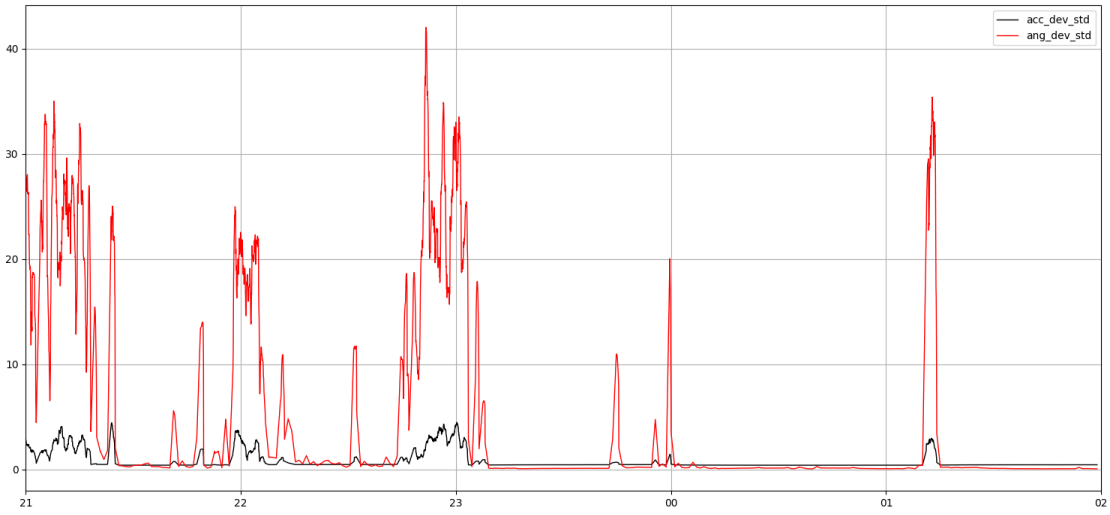


Nuovo codice, **classificator.py**:

dati resampled a 20Hz; per ora considerando solo inizio prima notte primo soggetto:

- Window size fissa a 1min

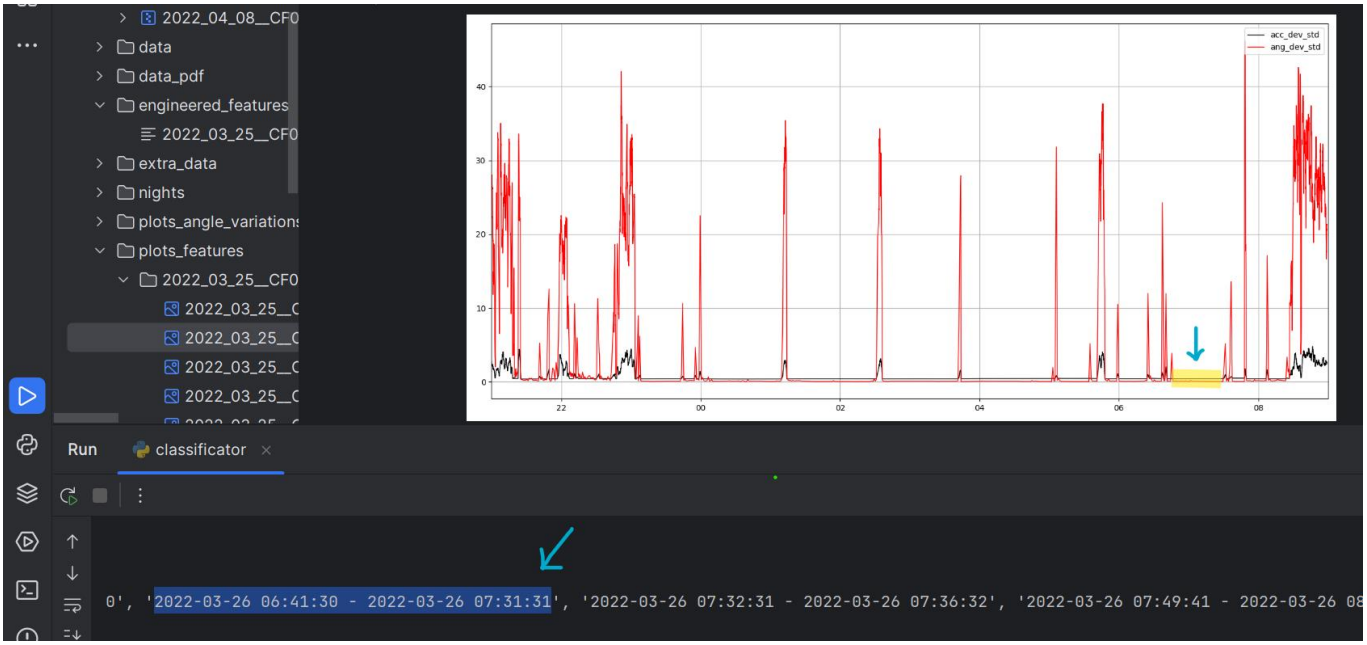
- Window slitta di 1sec se sveglio, 1min se addormentato



- Periodo considerato 21:00 → 09:00

- Window size fissata a 1min

- Test limiti per identificare periodi di sonno: acc_dev_std → 0.5, ang_dev_std → 0.5



- Test window size dinamica: 1h (sleep) e 1min (awake)

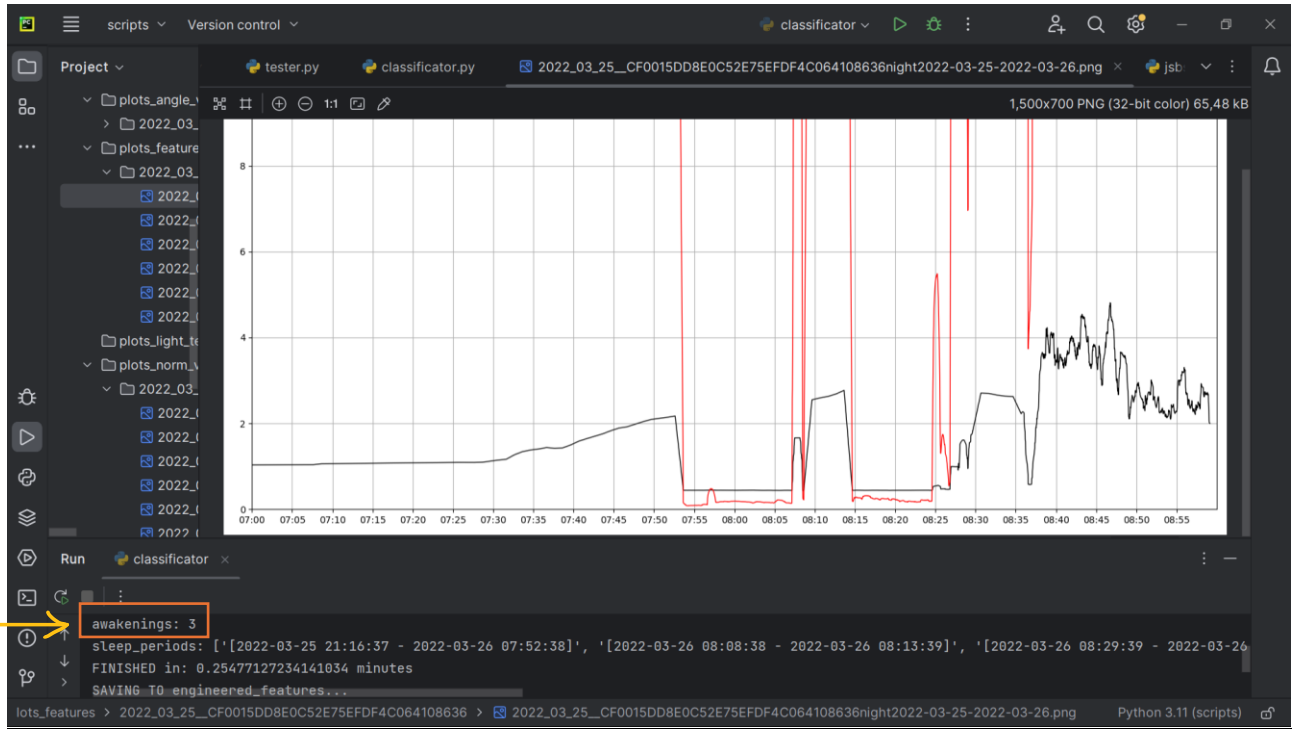
(Limiti per identificare periodi di sonno: acc_dev_std → 0.5, ang_dev_std → 0.5)

[ingrandimento plot tra le 7:00 e le 9:00 ma calcoli fatti su tutta la notte]

- Window size = 1h (sleep) e 1min (awake)

- Test limiti per identificare periodi di sonno: acc_dev_std → 2 ***(!errore corretto dopo)***

- Test tempo no movimenti per *sonno* = 15 min e tempo con movimenti per *risveglio* = 5 min (c'è offset)

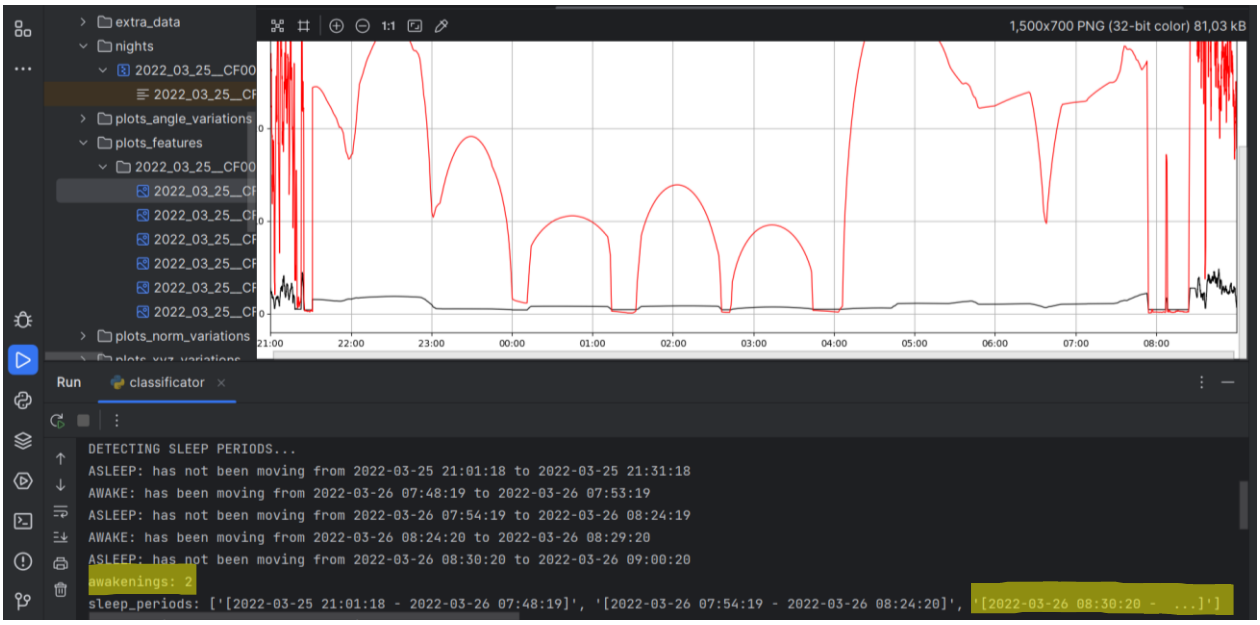


Rendendo tempo senza movimenti per sonno = 30 min →

```
# sliding windows to be used for calculations:
window_sleep = pd.Timedelta(hours=1)  # we don't
window_awake = pd.Timedelta(minutes=1)  # they
window_current = window_awake  # we start hypoth
window_slide_asleep = pd.Timedelta(minutes=1)
window_slide_awake = pd.Timedelta(seconds=1)

# intervals for sleep-wake detection:
interval_for_asleep = pd.Timedelta(minutes=30)
interval_for_awake = pd.Timedelta(minutes=5)  #
```

2 awakenings:



Tornando a intervallo di 15min per identificare sonno e togliendo offset:

```
window_sleep = pd.Timedelta(hours=1)  # we don'
window_awake = pd.Timedelta(minutes=1)  # they
window_current = window_awake  # we start hypot
window_slide_asleep = pd.Timedelta(minutes=1)
window_slide_awake = pd.Timedelta(seconds=1)

# intervals for sleep-wake detection:
interval_for_asleep = pd.Timedelta(minutes=15)
interval_for_awake = pd.Timedelta(minutes=5)
```

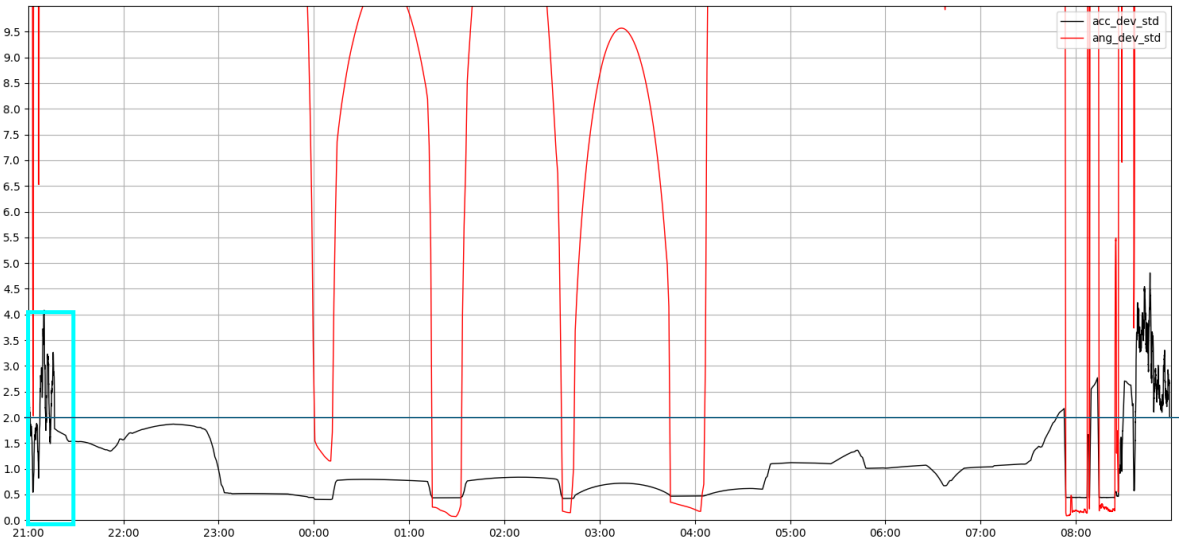
sleep_periods:

['[2022-03-25 21:01:37 - 2022-03-26 07:47:38]', problema

'[2022-03-26 07:53:38 - 2022-03-26 08:08:39]',

'[2022-03-26 08:14:39 - 2022-03-26 08:29:40]',

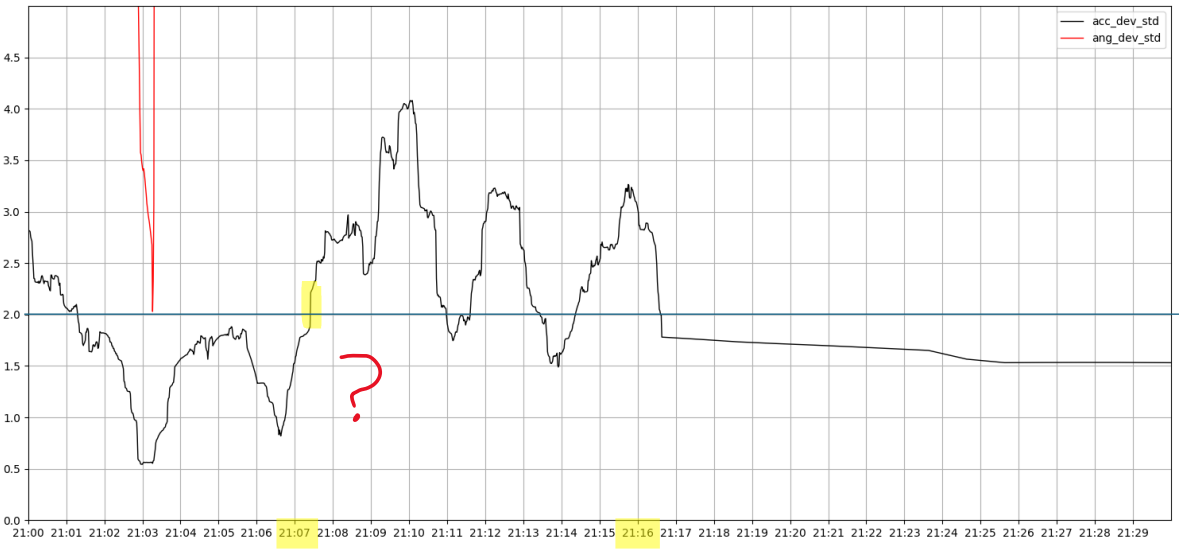
'[2022-03-26 08:44:09 - ...]'] → awakenings: 3



soglia

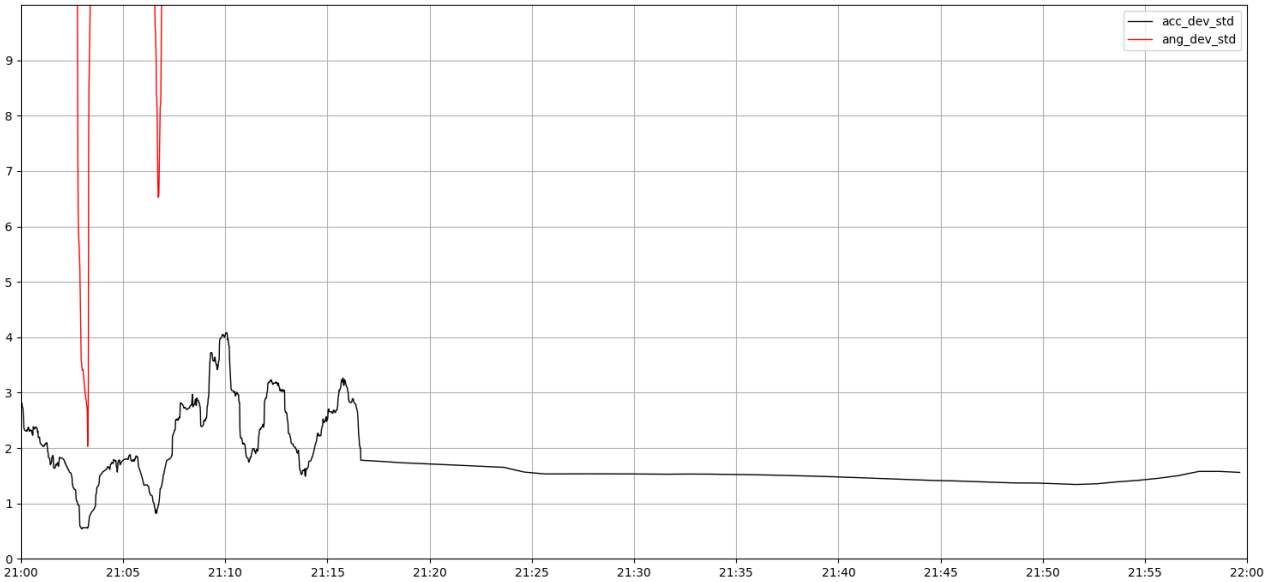
```
DETECTING SLEEP PERIODS...
ASLEEP: has not been moving from 2022-03-25 21:01:18 to 2022-03-25 21:16:37
AWAKE: has been moving from 2022-03-26 07:47:38 to 2022-03-26 07:52:38
ASLEEP: has not been moving from 2022-03-26 07:53:38 to 2022-03-26 08:08:38
AWAKE: has been moving from 2022-03-26 08:08:39 to 2022-03-26 08:13:39
ASLEEP: has not been moving from 2022-03-26 08:14:39 to 2022-03-26 08:29:39
AWAKE: has been moving from 2022-03-26 08:29:40 to 2022-03-26 08:34:40
ASLEEP: has not been moving from 2022-03-26 08:36:05 to 2022-03-26 08:59:09
```

Ulteriore ingrandimento:



Analizzando solo dati tra le 21 e le 22:

```
ASLEEP: has not been moving from 2022-03-25 21:01:18 to 2022-03-25 21:16:37
awakenings: 0
sleep_periods: ['[2022-03-25 21:01:37 -  ...]']
```



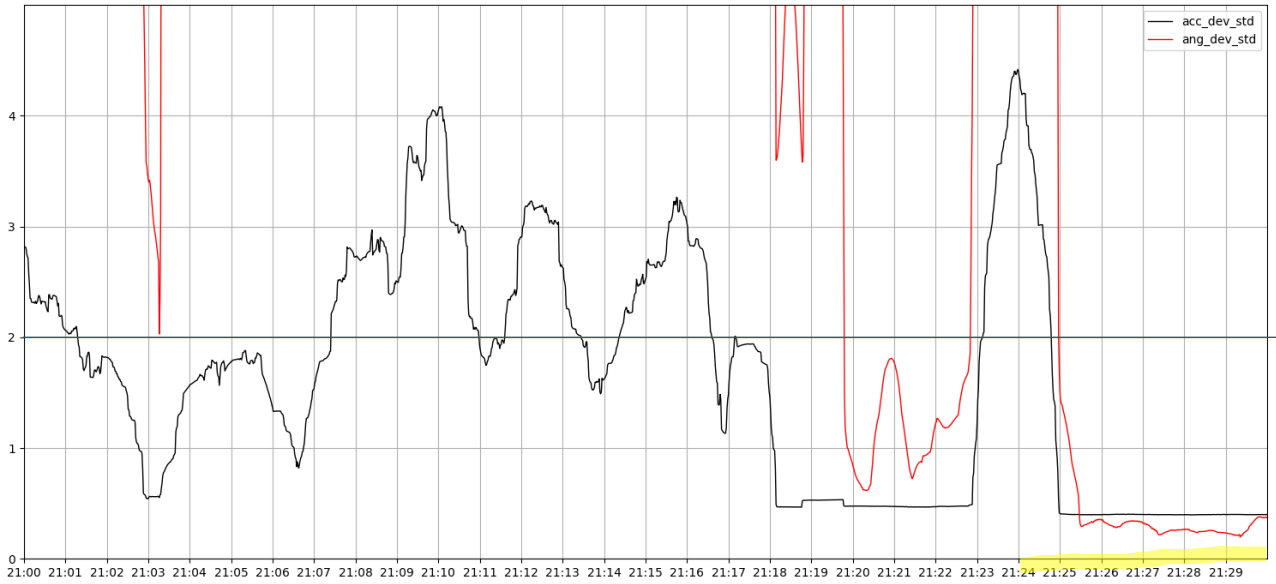
→ manca reset di stopped_moving_at

Aggiunto reset (e anche per started_moving_at):

```
else:  # was awake and is still considered awake
    print("(AWAKE) acc_dev_std > 2 --> ACTUALLY STILL AWAKE, reset stopped_moving_at")
    stopped_moving_at = None
    # add new row in df_windowed with associated features computed in the window
    df_windowed.loc[len(df_windowed.index)] = [df_in_window.index.min(), acc_dev_std, ang_dev_std,
                                               "Awake"]
```

Ora sembra andare meglio:

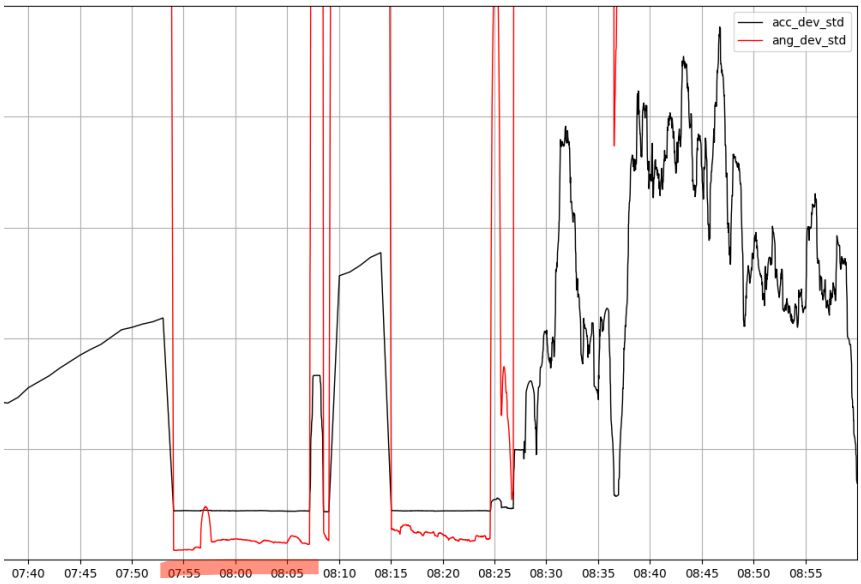
```
it's only been 0 days 00:14:59 minutes
2022-03-25 21:39:48 (AWAKE) acc_dev_std <= 2 --> MAYBE ASLEEP
they had already stopped moving
it's been 0 days 00:15:00 minutes
ASLEEP: has not been moving from 2022-03-25 21:24:48 to 2022-03-25 21:39:48
awakenings: 0
sleep_periods: ['[2022-03-25 21:24:48 -  ...]']
```



soglia

Altro problema a fine notte:

```
it's been 0 days 00:05:00 minutes
AWAKE: has been moving from 2022-03-26 07:48:01 to 2022-03-26 07:53:01
it's been 0 days 00:15:00 minutes
ASLEEP: has not been moving from 2022-03-26 07:54:01 to 2022-03-26 08:09:01
it's been 0 days 00:05:00 minutes
AWAKE: has been moving from 2022-03-26 08:09:02 to 2022-03-26 08:14:02
it's been 0 days 00:15:00 minutes
ASLEEP: has not been moving from 2022-03-26 09:14:26 to 2022-03-26 09:29:26
awakenings: 2
sleep_periods: ['[2022-03-26 07:00:00 - 2022-03-26 07:48:01]', '[2022-03-26 07:54:01 - 2022-03-26 08:09:02]', '[2022-03-26 09:14:26 -  ...]']
```



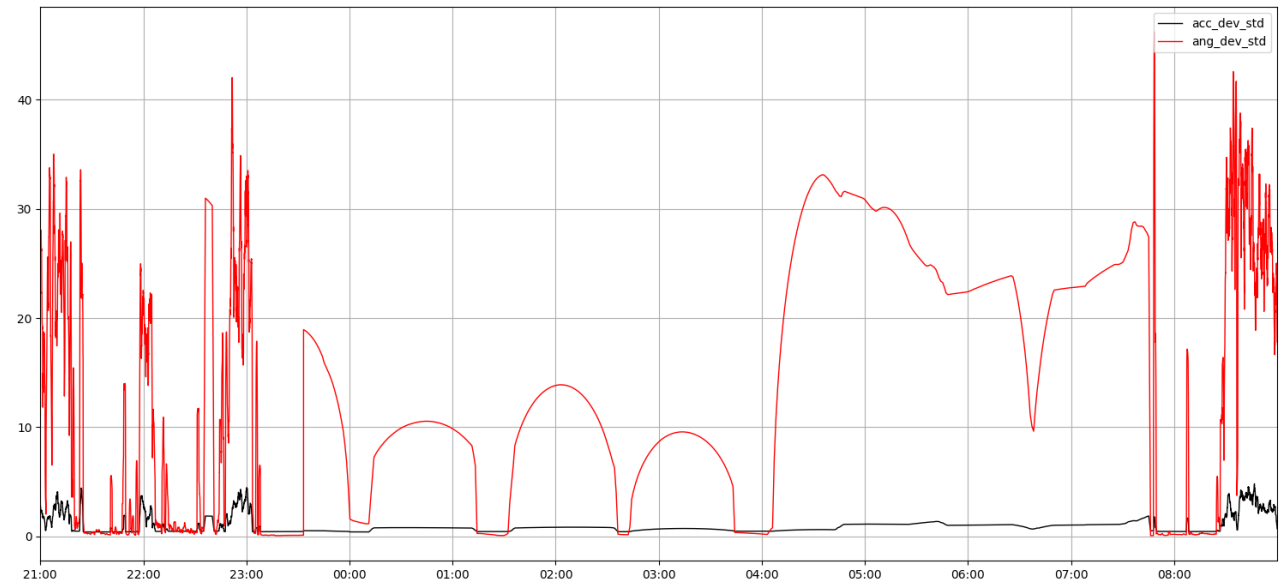
Nelle print e nei calcoli rimane incluso l'intervallo dalle 9 alle 10 (va usato solo per calcolare correttamente valori fino alle 9 nella window)

→Rimozione se orario > 9:00

Intervallo di 15 minuti per decretare sonno da portare a 30 minuti se non ci interessano gli intervalli così brevi di sonno

Generale miglioramento detection con intervallo per sonno portato da 15 a 30 e soglia della acc_dev_std abbassata da 2 a 1.5 ***(!errore corretto dopo)***

```
awakenings: 2
sleep_periods: ['[2022-03-25 22:05:06 - 2022-03-25 22:35:07]', '[2022-03-25 23:03:14 - 2022-03-26 07:40:15]']
```



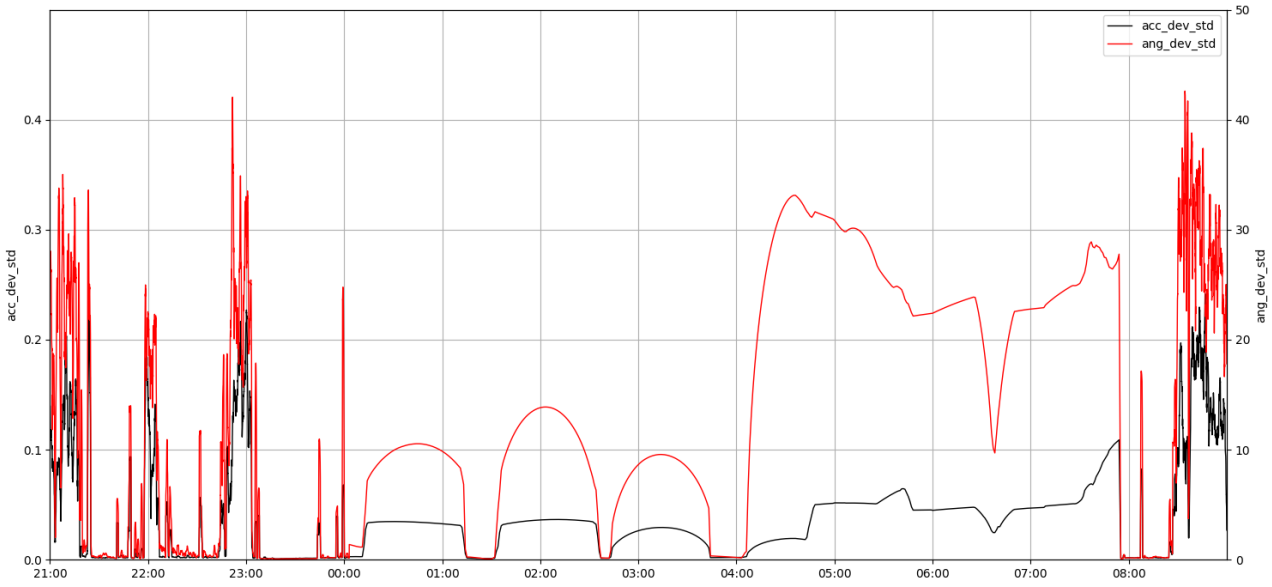
Aumentando intervallo per decretare sonno a 1h aumenta l'accuratezza con il diario del sonno:

Orario	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Venerdì 25 - Mar a letto / mi riposo																								
dormo																								
Sabato a letto / mi riposo	☒	☒	☒	☒	☒	☒	☒	☒	☒															☒
dormo	☒	☒	☒	☒	☒	☒	☒	☒																☒

```
AWAKE: has been moving from 2022-03-26 07:40:15 to 2022-03-26 07:45:15
awakenings: 1
sleep_periods: ['[2022-03-25 23:03:14 - 2022-03-26 07:40:15]']
```

***! correzione, resampling con mean() invece di sum() → soglia acc_dev_std a 0.1**

Aspetto ora con interval_sleep = 1h e interval_awake = 5min:




```
ASLEEP: has not been moving from 2022-03-25 23:03:05 to 2022-03-26 00:03:05
AWAKE: has been moving from 2022-03-26 07:49:06 to 2022-03-26 07:54:06
awakenings: 1
sleep_periods: ['[2022-03-25 23:03:05 - 2022-03-26 07:49:06]']
```

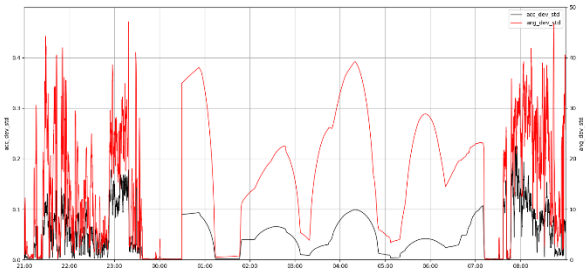
Integrazione di nights.py → medie calcolate con rolling windows da 5s come detto detto nei papers (invece di 30s come in nights.py) per togliere un po' di rumore tramite averaging

Without averaging:

----- DAY: 2022-03-25 -----

sleep_periods:

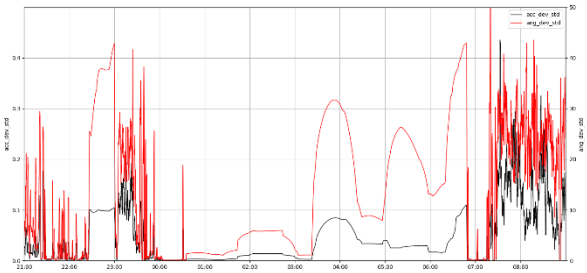
['[2022-03-25 23:03:05 - 2022-03-26 07:49:06]']



----- DAY: 2022-03-26 -----

sleep_periods:

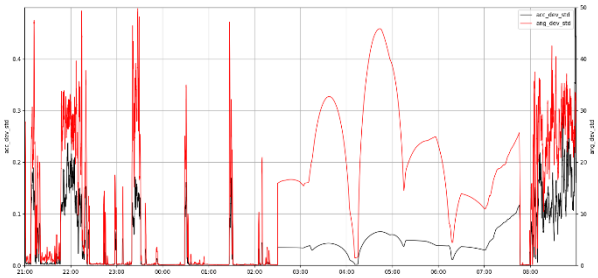
['[2022-03-27 00:06:57 - 2022-03-27 07:32:58]']



----- DAY: 2022-03-27 -----

sleep_periods:

['[2022-03-28 01:30:06 - 2022-03-28 07:41:07]']

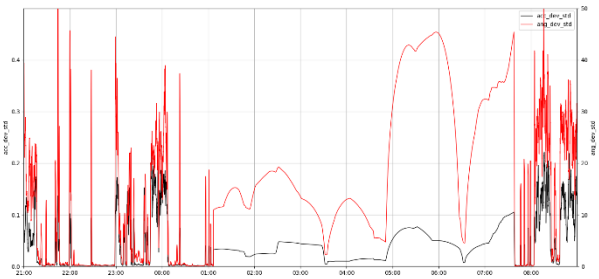


----- DAY: 2022-03-28 -----

sleep_periods:

['[2022-03-28 21:25:51 - 2022-03-28 22:54:52]',

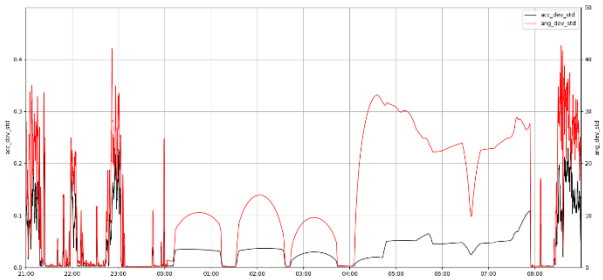
'[2022-03-28 23:35:19 - 2022-03-29 06:43:20]']



----- DAY: 2022-03-29 -----

sleep_periods:

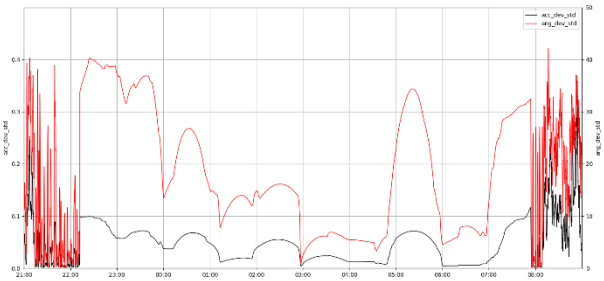
['[2022-03-29 23:29:24 - 2022-03-30 07:05:25]']



----- DAY: 2022-03-30 -----

sleep_periods:

['[2022-03-30 21:12:01 - 2022-03-31 07:49:02]']



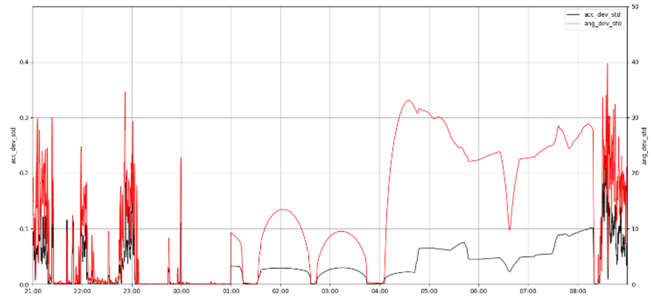
Con averaging post re-sampling:

----- DAY: 2022-03-25 -----

sleep_periods:

['[2022-03-25 23:59:51 - 2022-03-26 08:13:52]']

Orario	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Venerdì 25 - Mar a letto / mi riposo																								
dormo																								
Sabato a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠																⊠

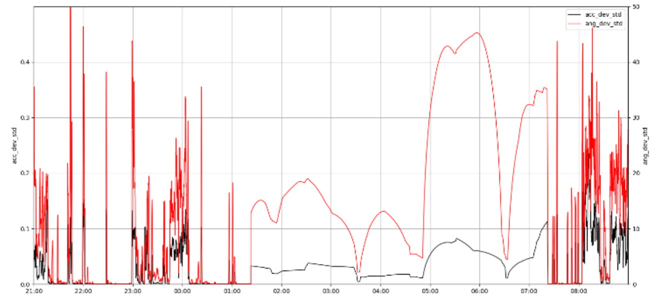


----- DAY: 2022-03-26 -----

sleep_periods:

['[2022-03-27 00:23:02 - 2022-03-27 07:17:03]']

Sabato a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠																⊠
Domenica a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠																⊠

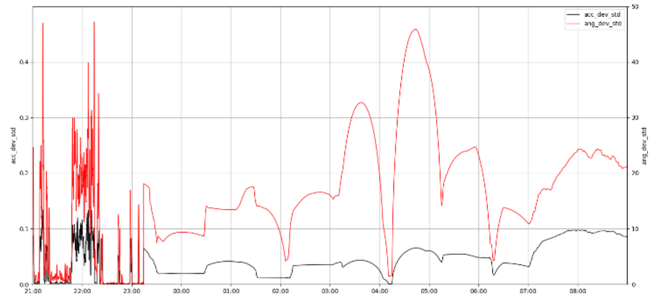


----- DAY: 2022-03-27 -----

sleep_periods:

['[2022-03-27 22:14:13 - ...]']

Domenica a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠																⊠
Lunedì a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠																⊠

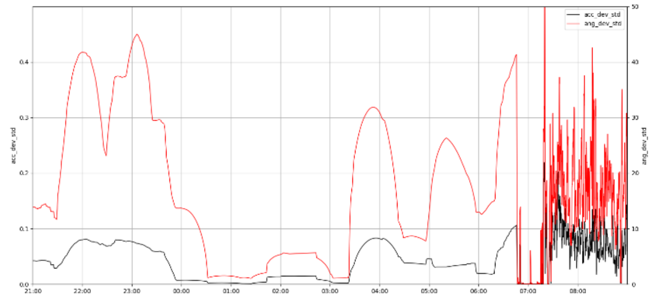


----- DAY: 2022-03-28 -----

sleep_periods:

['[2022-03-28 21:00:00 - 2022-03-29 06:41:01]']

Lunedì a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠																⊠
Martedì a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠																⊠

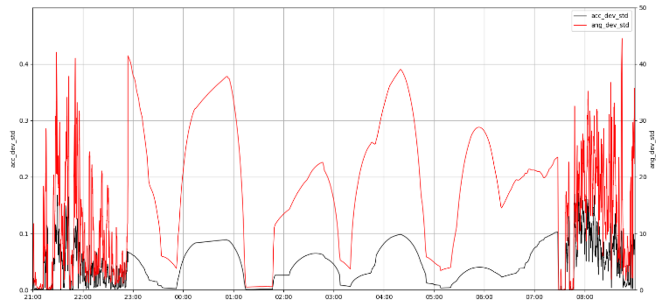


----- DAY: 2022-03-29 -----

sleep_periods:

['[2022-03-29 21:53:52 - 2022-03-30 07:22:53]']

Martedì a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠																⊠
Mercoledì a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠																	⊠

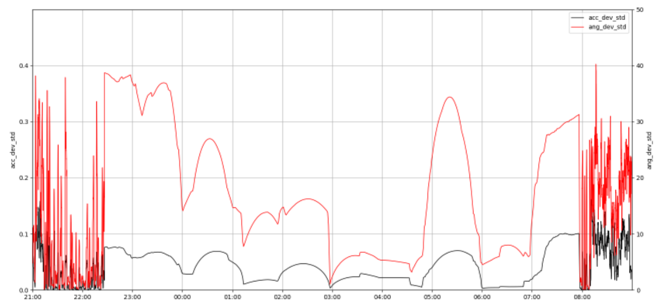


----- DAY: 2022-03-30 -----

sleep_periods:

['[2022-03-30 21:26:31 - 2022-03-31 07:51:32]']

Mercoledì a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															⊠
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠																	⊠
Giovedì 31 - Mar a letto / mi riposo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠															
dormo	⊠	⊠	⊠	⊠	⊠	⊠	⊠	⊠																



Abbassando soglia acc_dev_std a 0.09 miglioramento di alcuni, peggioramento di altri:

25/03: ['[2022-03-25 23:59:56 - 2022-03-26 07:53:57]']

26/03: ['[2022-03-27 00:23:16 - 2022-03-27 07:14:17]']

27/03: ['[2022-03-27 23:31:13 - 2022-03-28 07:39:14]']

28/03: ['[2022-03-28 21:25:23 - 2022-03-29 06:37:24]']

29/03: ['[2022-03-29 21:53:53 - 2022-03-30 04:05:54]',

'[2022-03-30 04:50:42 - 2022-03-30 07:11:43]']

30/03: ['[2022-03-30 21:26:32 - 2022-03-31 07:24:33]']

➔ Opterei per tenere soglia a 0.1

Prossimi passi: integrare features / regolare parametri per aumentare la precisione di rilevamento sonno