Analisando a qualidade do meu sono

Reuben Nascimento Morais

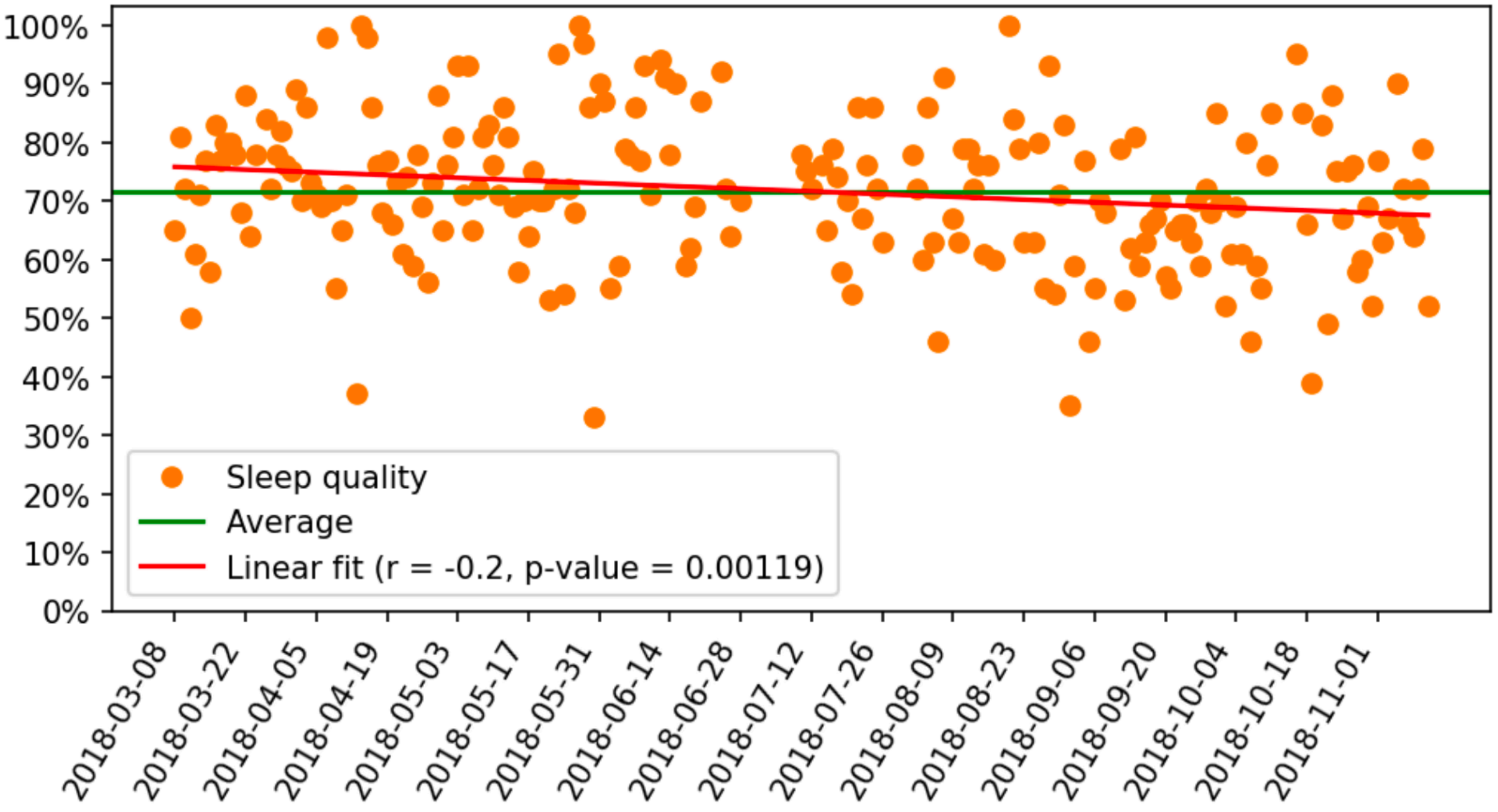
http://reub.in/sono.pdf

Fontes de dados:

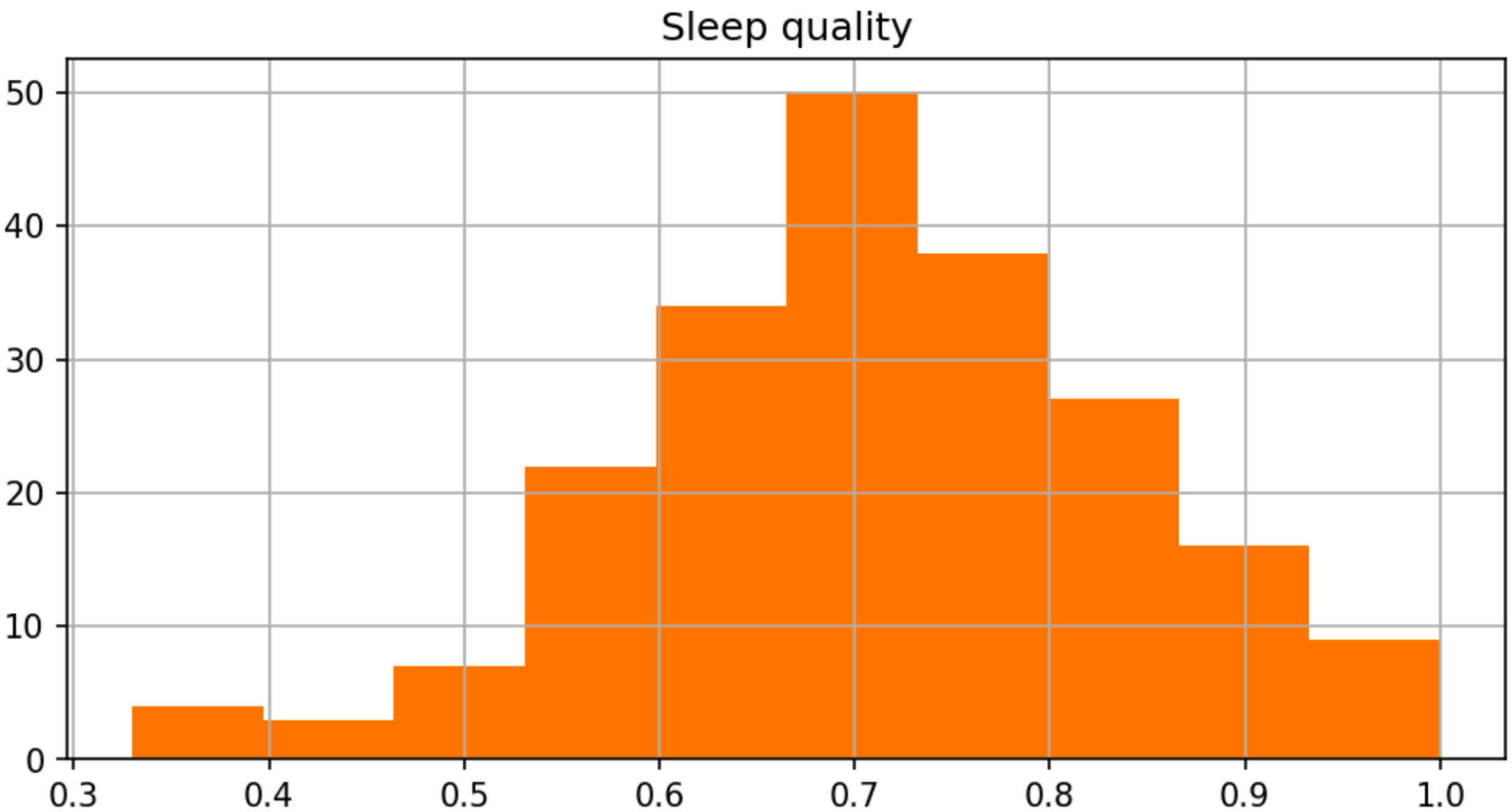
- Sleep Cycle (qualidade do sono, horas de sono)
- Life Cycle (tempo em casa/em aula/viajando)
- Apple Health (batimentos cardiácos por minuto, tempo de atividade física)
- Fases da lua

Dados de sono e localização coletados de 2018-03-07 a 2018-11-11 (232 dias).

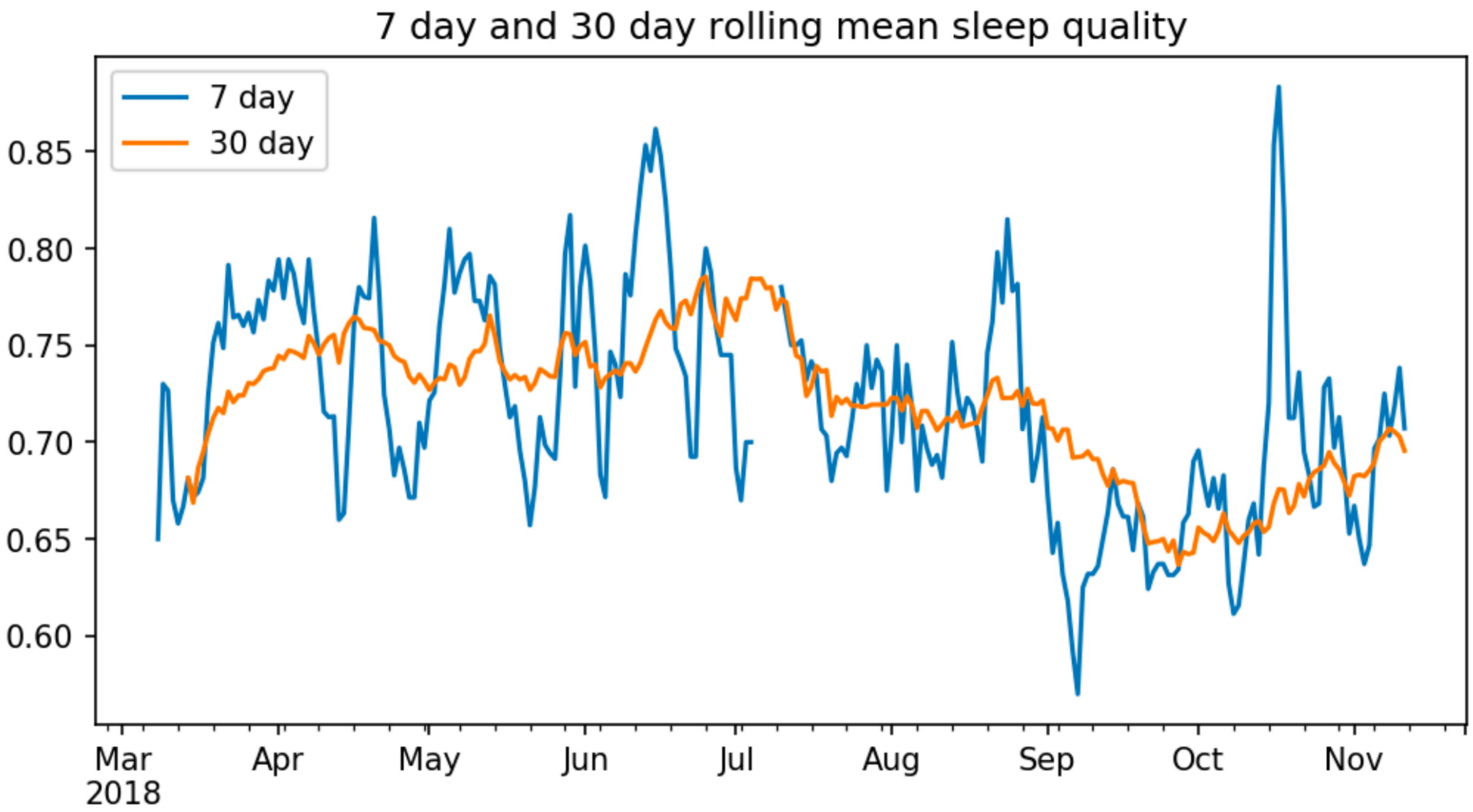
Dados de batimentos cardíacos e esportes coletados de 2018-06-13 a 2018-11-11 (152 dias).


100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Sleep quality
Average
Linear fit (r = -0.2, p-value = 0.00119)
2018-03-08
2018-03-22
2018-04-05
2018-04-19
2018-05-03
2018-05-17
2018-05-31
2018-06-14
2018-06-28
2018-07-12
2018-07-26
2018-08-09
2018-08-23
2018-09-06
2018-09-20
2018-10-04
2018-10-18
2018-11-01

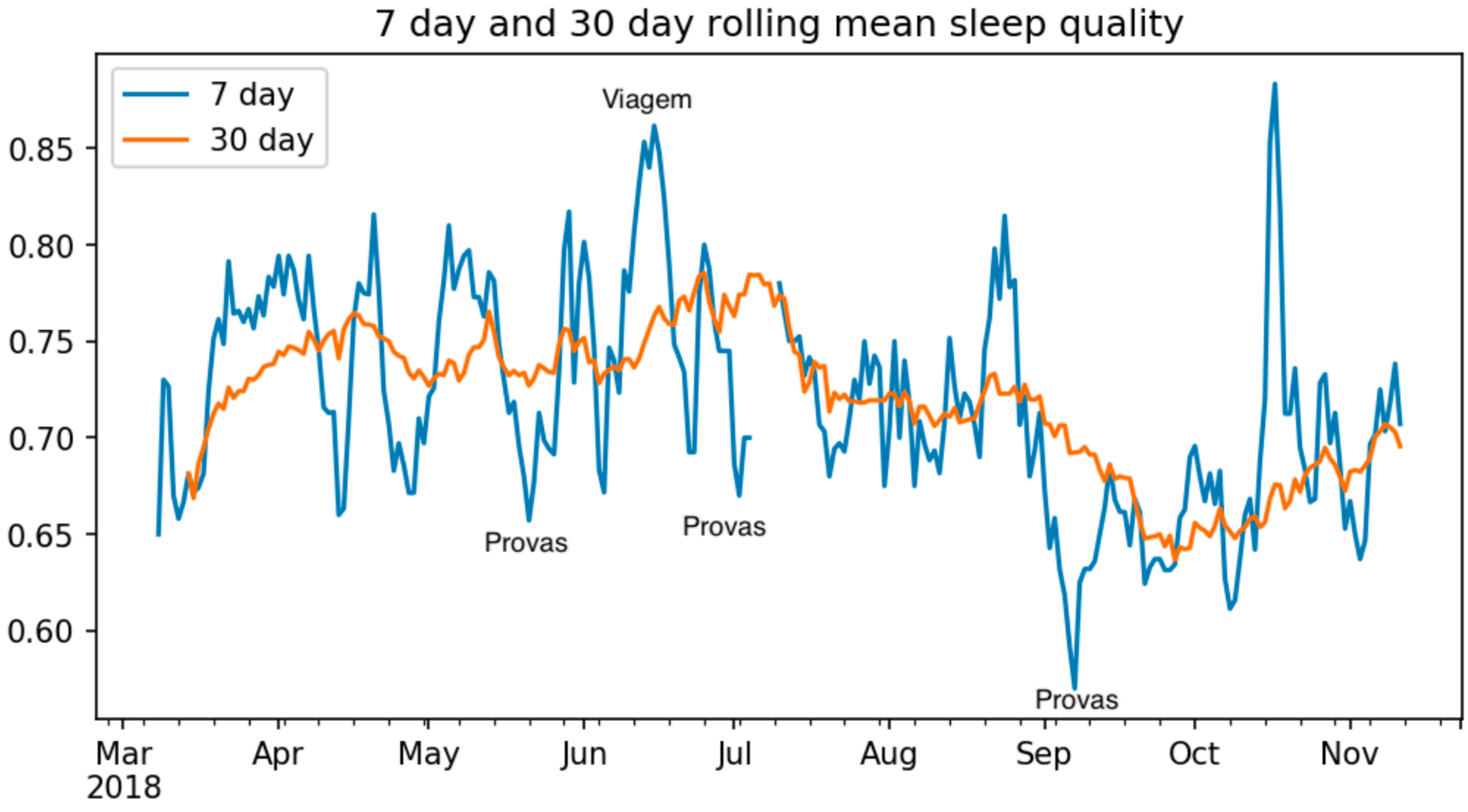
```
sleepquality.hist(color='C1')
plt.title('Sleep quality')
plt.show()
```



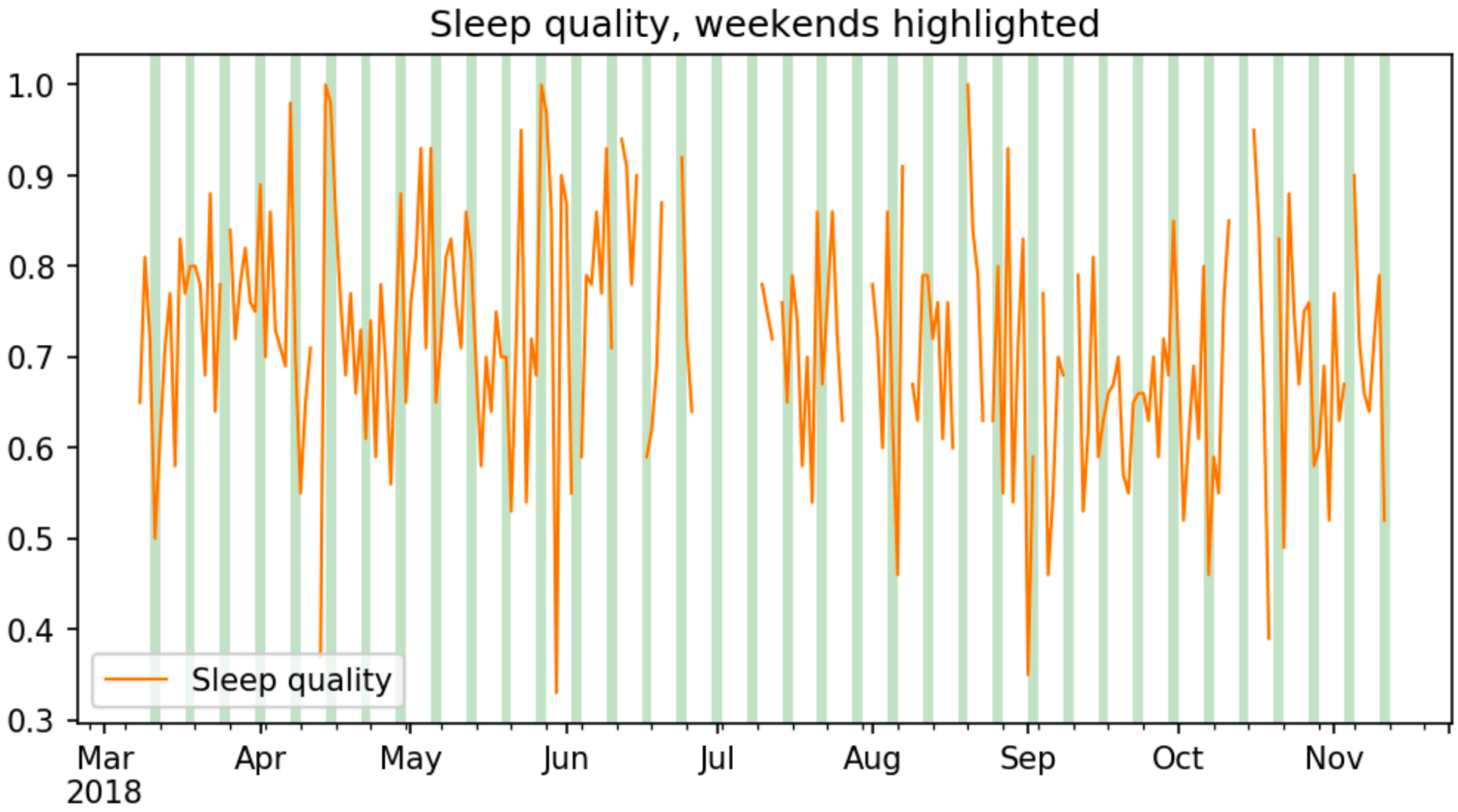
```
sleepquality.rolling('7d').mean().plot(label='7 day')
sleepquality.rolling('30d', min_periods=7).mean().plot(label='30 day')
plt.title('7 day and 30 day rolling mean sleep quality')
plt.legend()
plt.show()
```



```
Image('annotated.png')
```



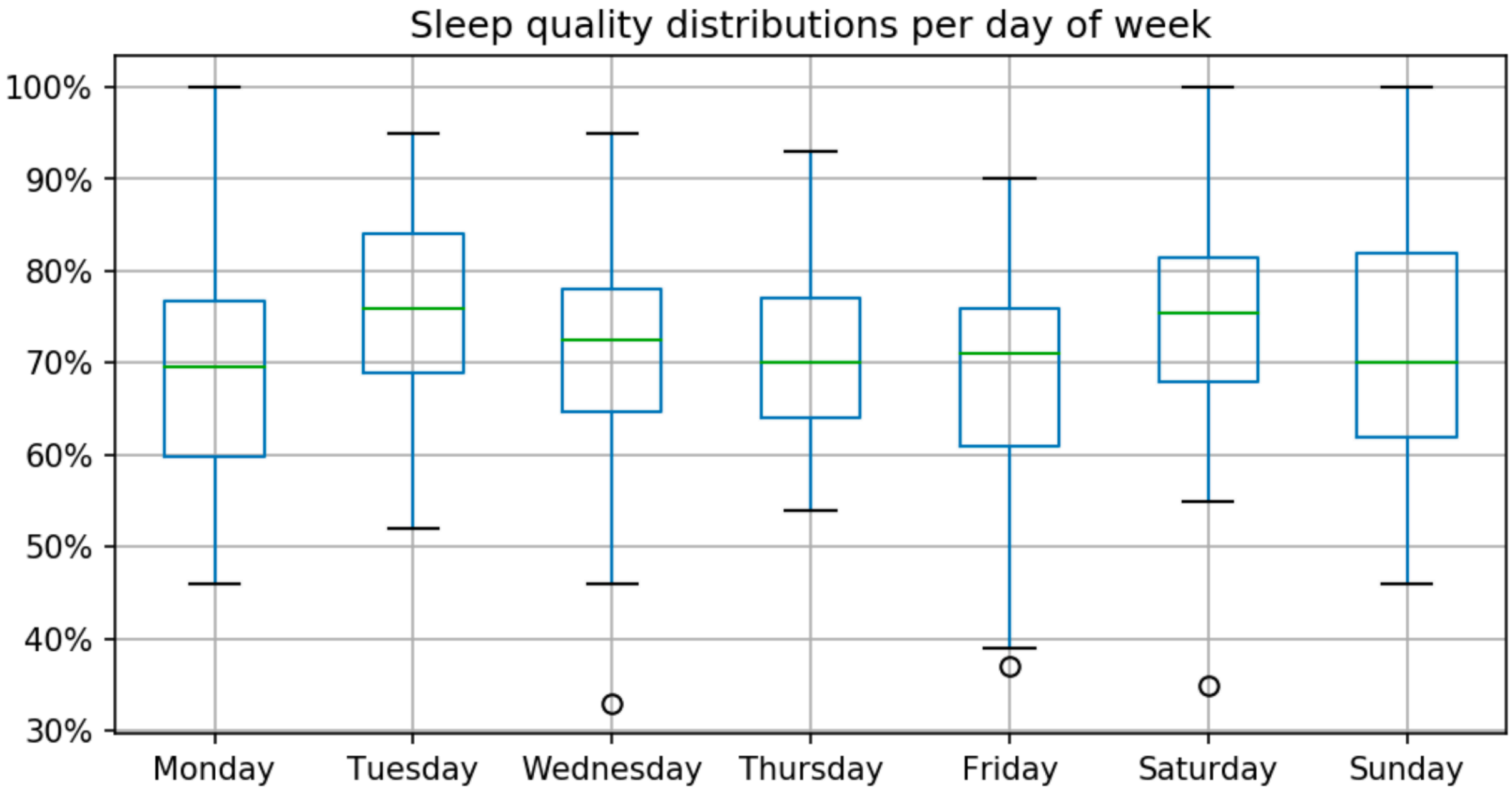
```
sleepquality.plot(color='C1', linewidth=1)

for i in sleepdata.index[sleepdata.index.weekday >= 5]:
    plt.gca().axvspan(i, i+pandas.to_timedelta('1d'), facecolor='green', edgecolor='none', alpha=.2)

plt.title('Sleep quality, weekends highlighted')
plt.legend(loc='lower left')
plt.show()
```



```python
weekends = sleepquality[sleepquality.index.weekday >= 5]
weekdays = sleepquality[sleepquality.index.weekday < 5]
print(f'Global average: {quality_average}, n = {len(sleepdata)}')
print(f'Weekend average: {weekends.mean()}, n = {len(weekends)}')
print(f'Weekday average: {weekdays.mean()}, n = {len(weekdays)}')
print('P(quality > mean | weekend):', (weekends > quality_average).mean())
print('P(quality > mean | weekday):', (weekdays > quality_average).mean())
```

```
Global average: 0.7167142857142856, n = 249
Weekend average: 0.7341818181818, n = 72
Weekday average: 0.710516129032258, n = 177
P(quality > mean | weekend): 0.4027777777777778
P(quality > mean | weekday): 0.4350282485875706
```

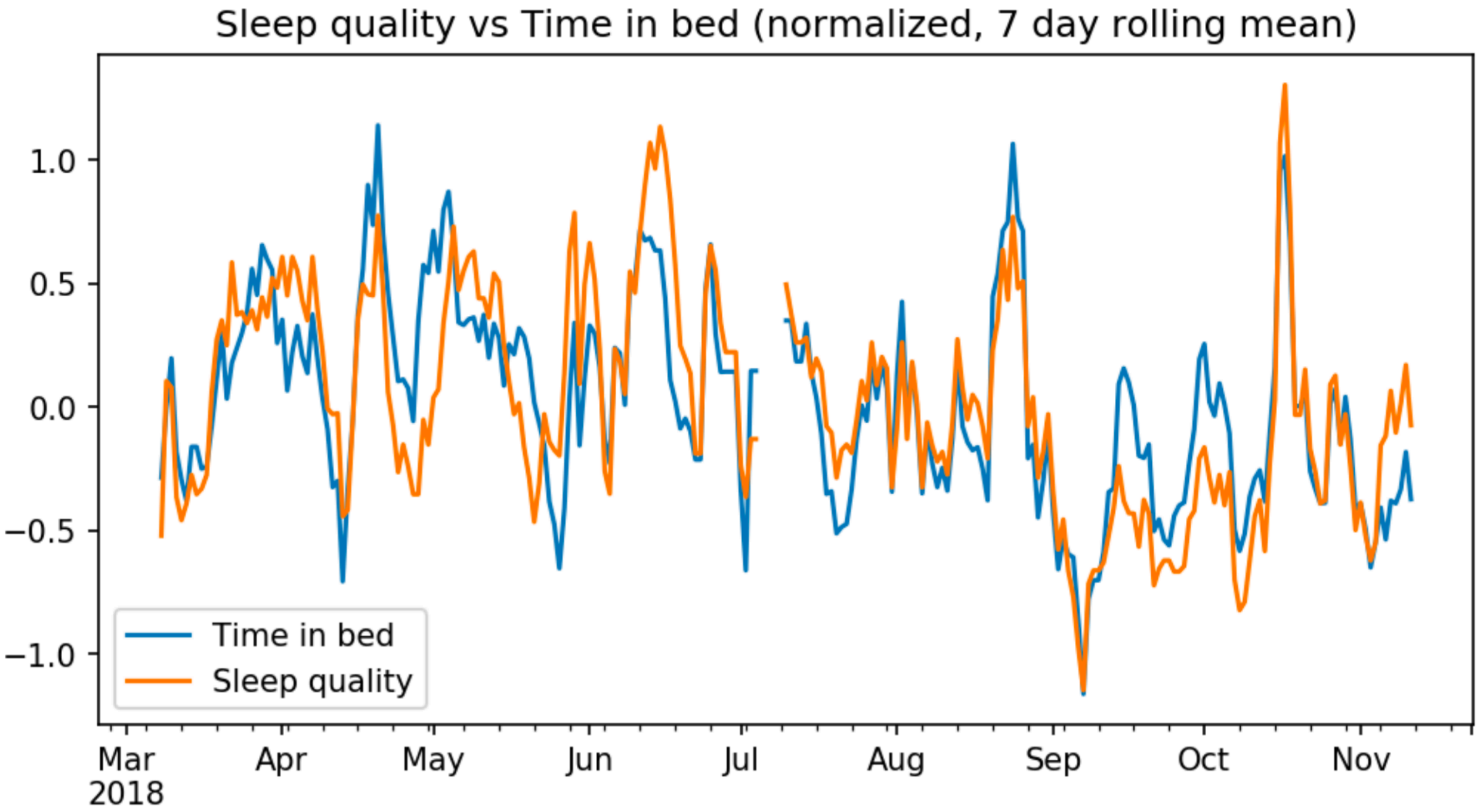
```
weekdays_df = pandas.DataFrame(index=['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'],
                               data=[sleepquality[sleepquality.index.weekday == i].values for i in range(7)]).T
weekdays_df.boxplot()
plt.gca().yaxis.set_major_formatter(ticker.PercentFormatter(xmax=1))
plt.title('Sleep quality distributions per day of week')
plt.show()
```



```
Mean quality by moon phase:

{'First Quarter': 0.7064583333333334,
 'Full Moon': 0.7230769230769233,
 'Last Quarter': 0.7018518518518518,
 'New Moon': 0.7339285714285715}

Global average: 0.7167142857142856
```

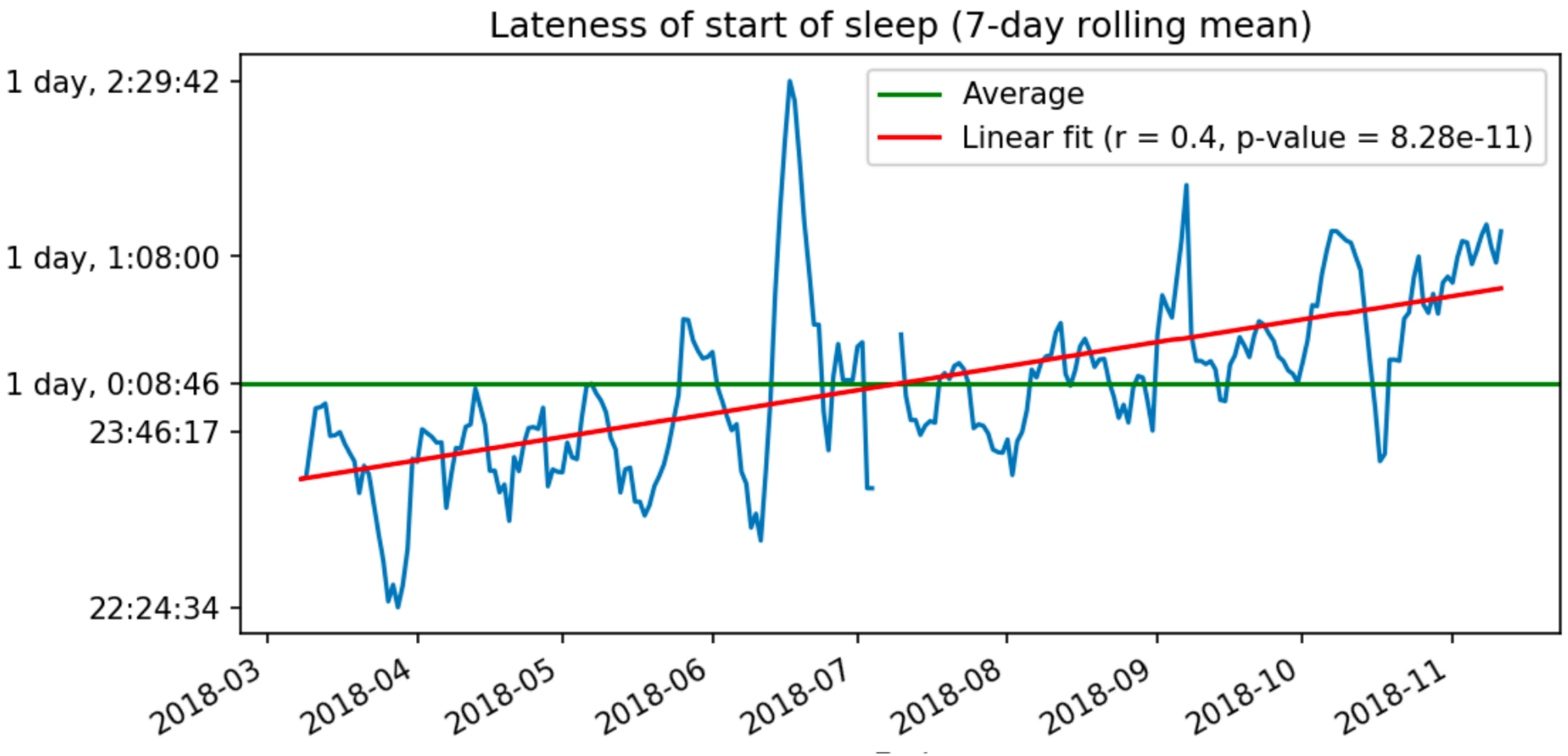
```
day>average vs. First Quarter:    r = -0.0709,    p-value = 0.265, n = 58
day>average vs. Full Moon:        r = 0.0384,     p-value = 0.547, n = 61
day>average vs. Last Quarter:     r = -0.0494,    p-value = 0.438, n = 65
day>average vs. New Moon:         r = 0.0801,     p-value = 0.208, n = 65
```

```
normalized_time.rolling('7d').mean().plot(color='C0')
normalized_quality.rolling('7d').mean().plot(color='C1')

plt.title('Sleep quality vs Time in bed (normalized, 7 day rolling mean)')
plt.legend()
plt.show()
```



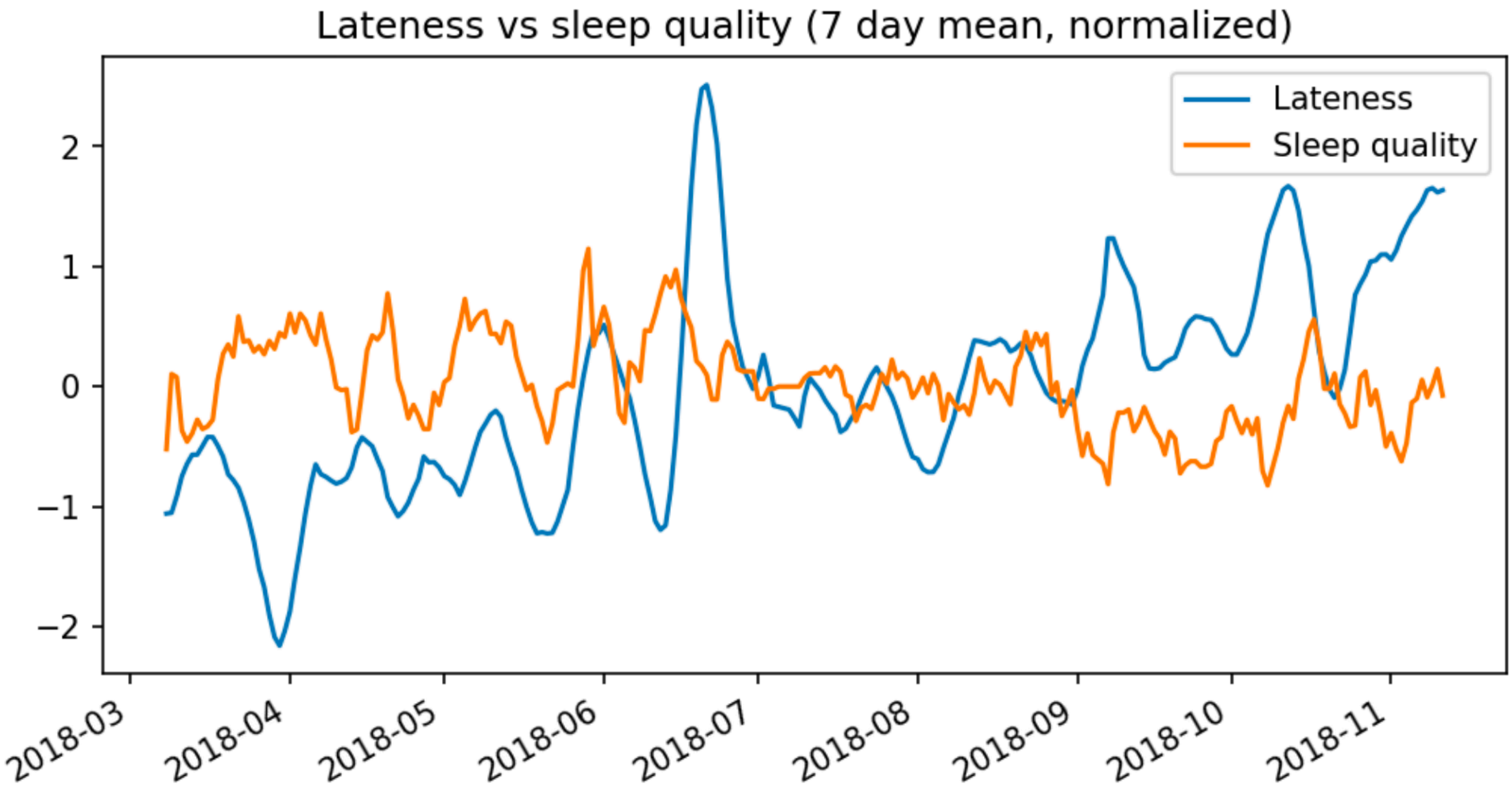
```
r, p = stats.pearsonr(normalized_time.fillna(0), normalized_quality.fillna(0))
print(f'r = {r}, p-value = {p}')
```

```
r = 0.8248180315592906, p-value = 4.164236371099999e-63
```


Lateness of start of sleep (7-day rolling mean)
1 day, 2:29:42
1 day, 1:08:00
1 day, 0:08:46
23:46:17
22:24:34
Average
Linear fit (r = 0.4, p-value = 8.28e-11)
2018-03
2018-04
2018-05
2018-06
2018-07
2018-08
2018-09
2018-10
2018-11

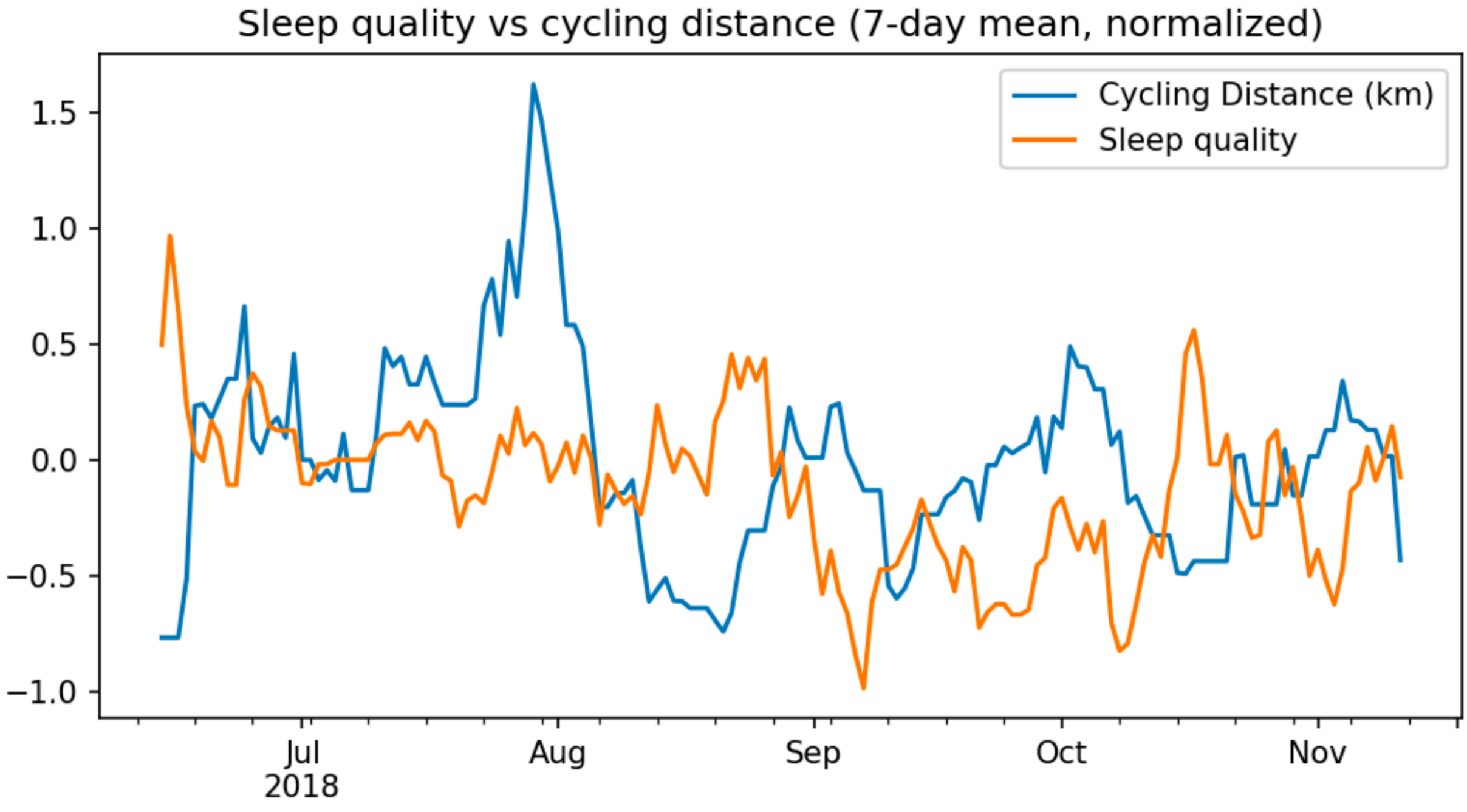
```
aligned_lateness = align_and_compare(normalized_quality, normalized_lateness)
plt.title('Lateness vs sleep quality (7 day mean, normalized)')
plt.legend()
plt.show()
```

```
r = -0.19498935295208325, p-value = 0.0021244810286180703
```



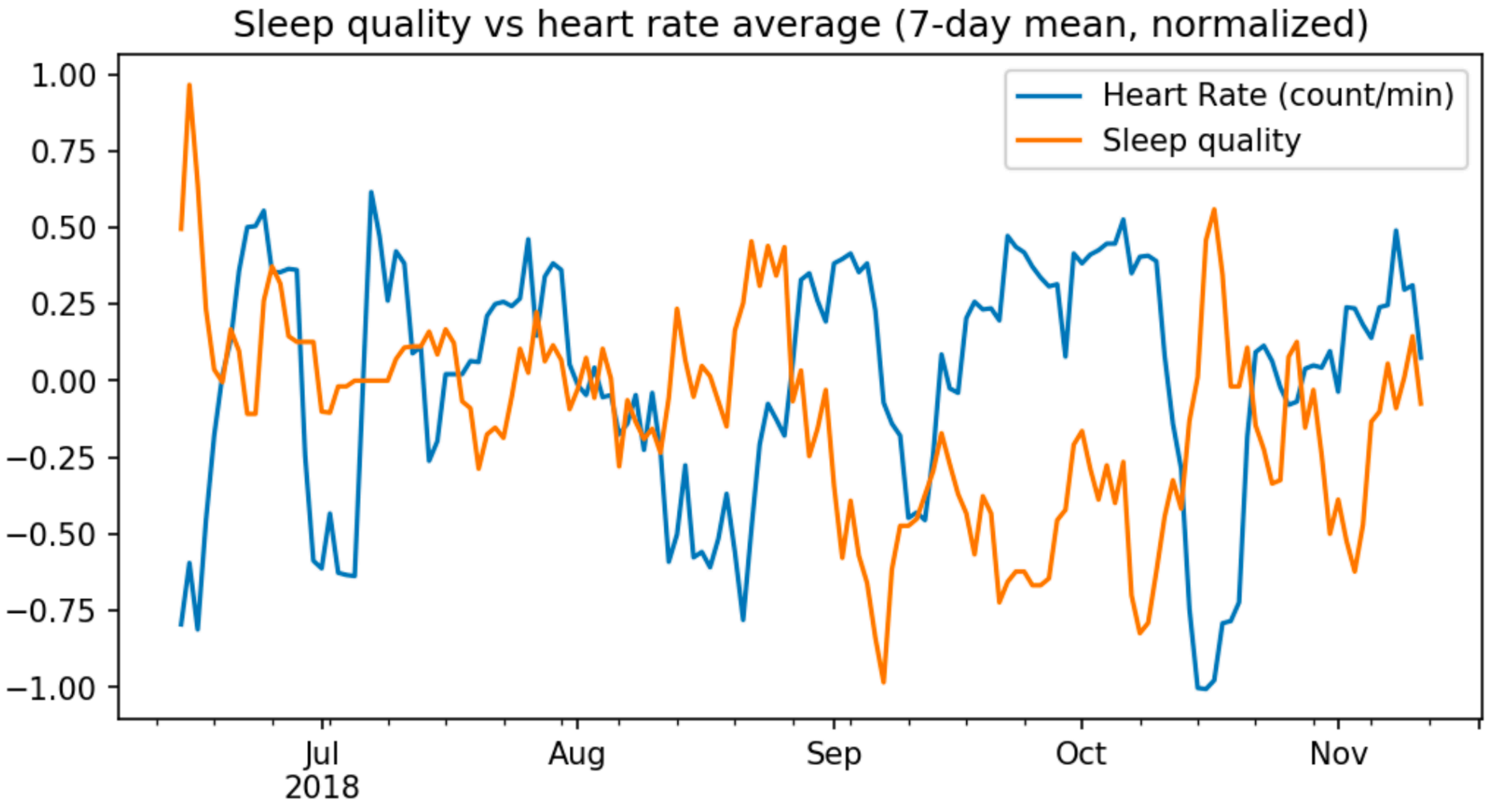
```
aligned_cycling = align_and_compare(normalized_quality, normalized_cycling)
plt.title('Sleep quality vs cycling distance (7-day mean, normalized)')
plt.legend()
plt.show()
```

```
r = -0.062472522639179254, p-value = 0.4460357077804835
```



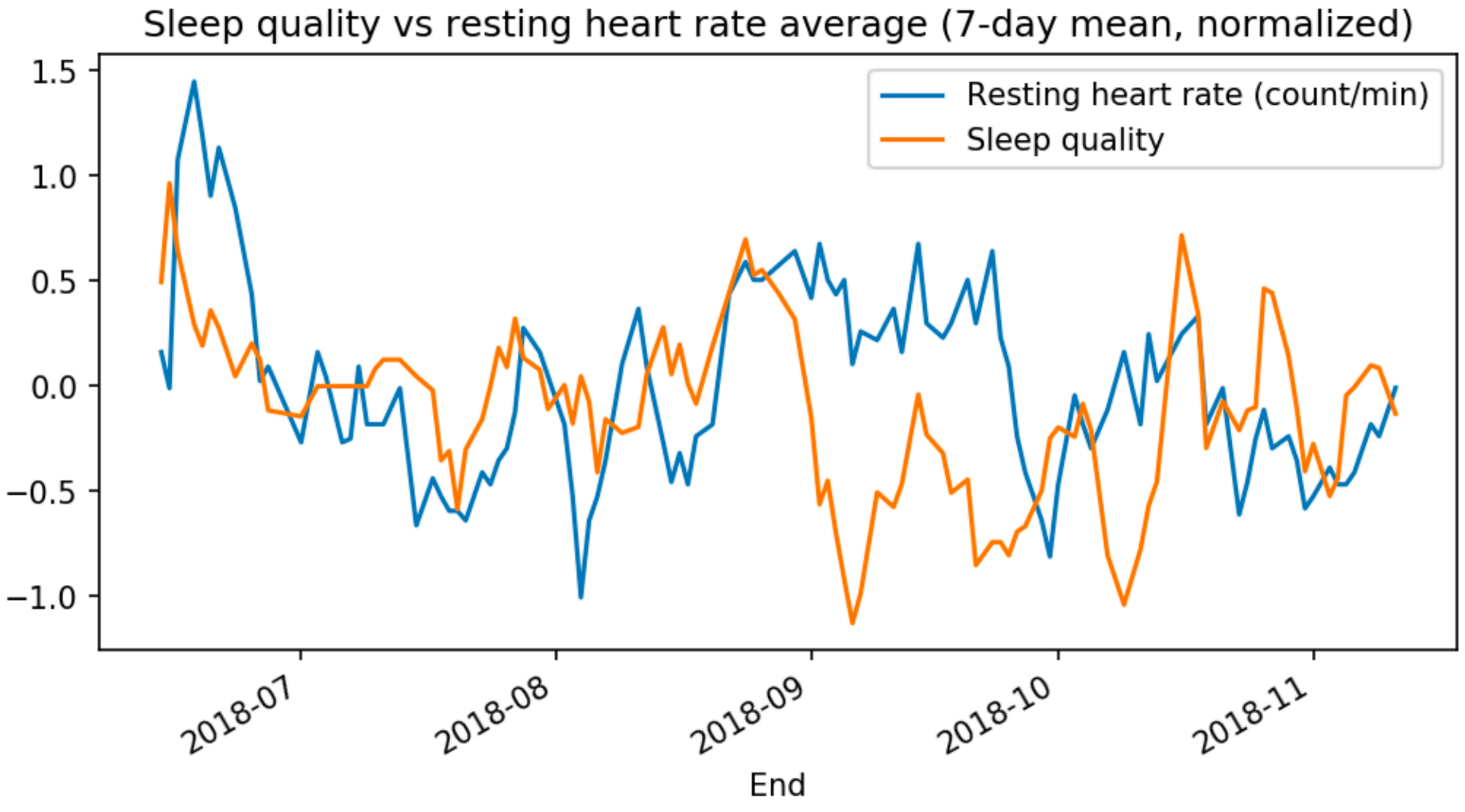
```
aligned_hr = align_and_compare(normalized_quality, normalized_heartrate)
plt.title('Sleep quality vs heart rate average (7-day mean, normalized)')
plt.legend()
plt.show()
```

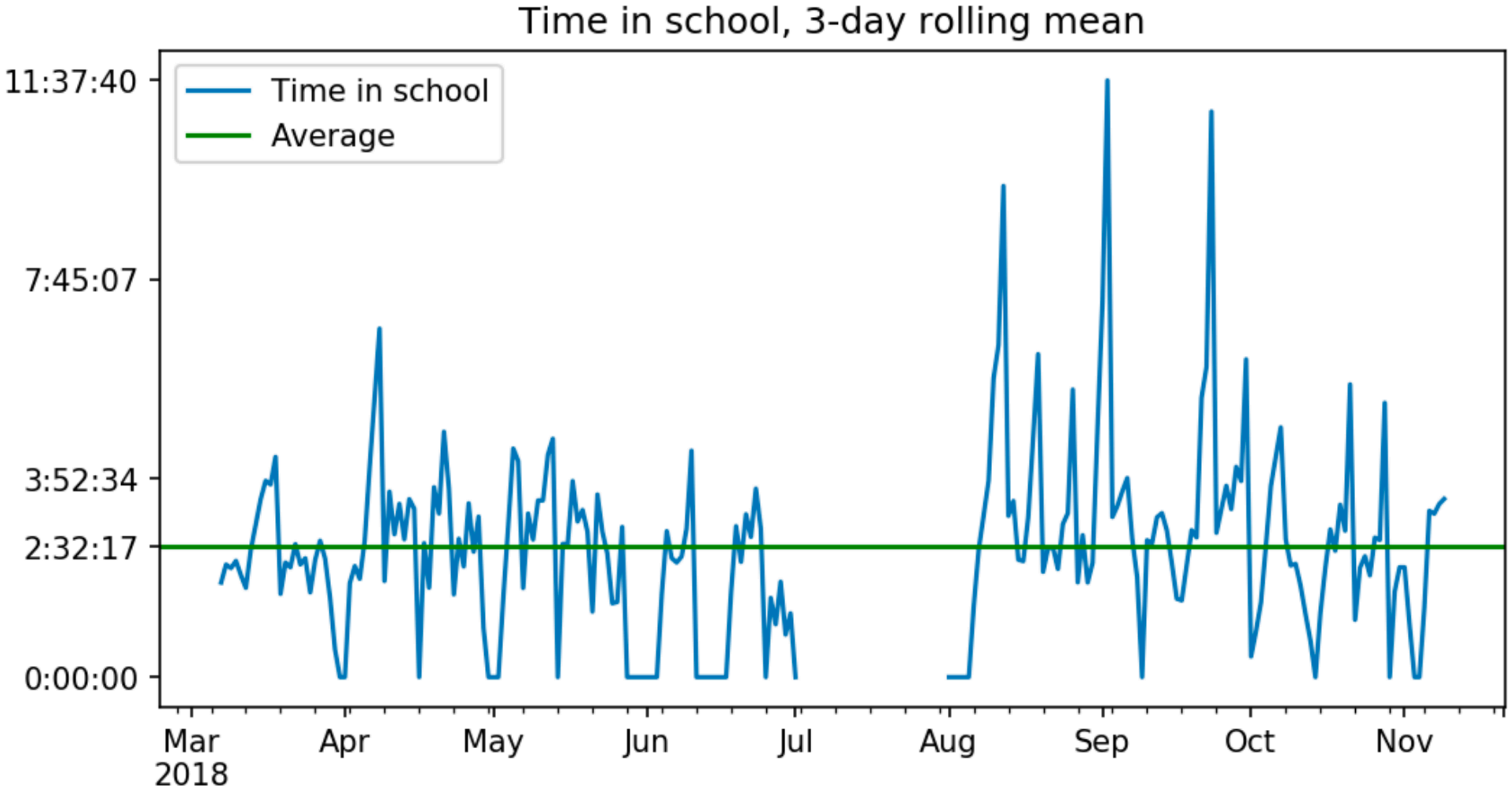
```
r = -0.11030434760685069, p-value = 0.17756284414832138
```



```
aligned_resting = align_and_compare(normalized_quality, normalized_resting)
plt.title('Sleep quality vs resting heart rate average (7-day mean, normalized)')
plt.legend()
plt.show()
```

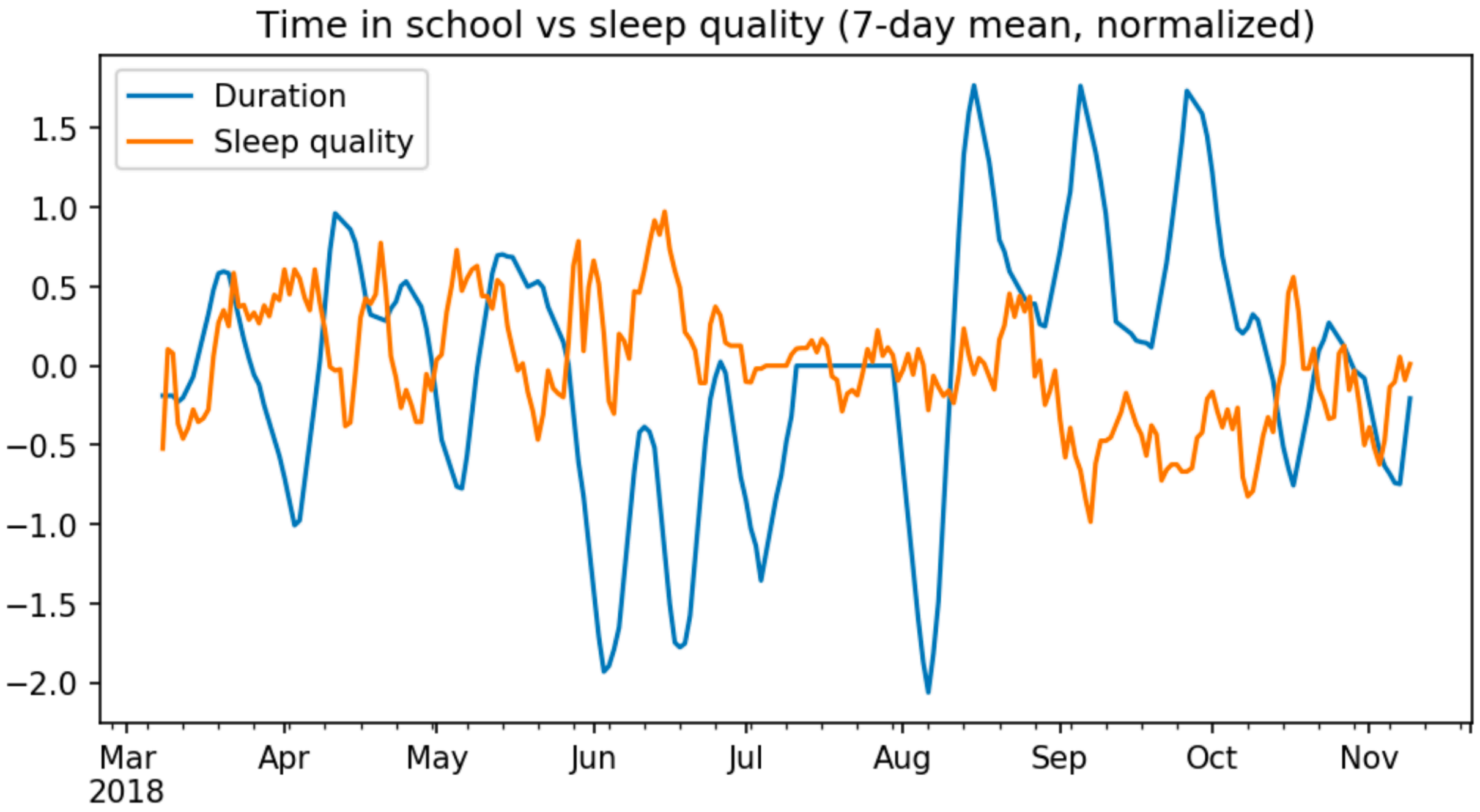
```
r = 0.08934727412678754, p-value = 0.35780248929367564
```




Time in school, 3-day rolling mean
Time in school
Average
11:37:40
7:45:07
3:52:34
2:32:17
0:00:00
Mar
2018
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov

```
normalized_school = normalize_series(school.rolling('7d').mean())
aligned_school = align_and_compare(normalized_quality, normalized_school)
plt.title('Time in school vs sleep quality (7-day mean, normalized)')
plt.legend()
plt.show()
```

```
r = -0.1789462436620316, p-value = 0.004788912541843903
```



```
data = pandas.concat((sleepquality.fillna(sleepquality.mean()).rename('Quality'),
                      normalized_time.rename('Time'),
                      aligned_lateness,
                      aligned_cycling.rename('Cycle_distance'),
                      aligned_hr.rename('HR'),
                      aligned_school.rename('School')), axis=1)
model = smf.ols('Quality ~ Time + Lateness + Cycle_distance + HR + School', data).fit()
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                Quality   R-squared:                       0.773
Model:                            OLS   Adj. R-squared:                  0.763
Method:                 Least Squares   F-statistic:                     72.38
Date:                Thu, 22 Nov 2018   Prob (F-statistic):           1.37e-32
Time:                        19:09:57   Log-Likelihood:                162.15
No. Observations:                 112   AIC:                            -312.3
Df Residuals:                     106   BIC:                            -296.0
Df Model:                           5
Covariance Type:            nonrobust
================================================================================
                   coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
Intercept        0.7152      0.007    106.205      0.000       0.702       0.729
Time             0.1128      0.006     18.290      0.000       0.101       0.125
Lateness         0.0010      0.007      0.149      0.882      -0.012       0.014
Cycle_distance  -0.0072      0.007     -1.001      0.319      -0.021       0.007
HR               0.0047      0.008      0.570      0.570      -0.012       0.021
School          -0.0128      0.006     -2.315      0.023      -0.024      -0.002
==============================================================================
Omnibus:                        3.835   Durbin-Watson:                   1.960
Prob(Omnibus):                  0.147   Jarque-Bera (JB):                3.938
Skew:                          -0.187   Prob(JB):                        0.140
Kurtosis:                       3.839   Cond. No.                         2.32
==============================================================================
```

```
model = smf.ols('Quality ~ Time + School', data).fit()
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                Quality   R-squared:                       0.690
Model:                            OLS   Adj. R-squared:                  0.687
Method:                 Least Squares   F-statistic:                     227.7
Date:                Thu, 22 Nov 2018   Prob (F-statistic):           8.33e-53
Time:                        19:10:01   Log-Likelihood:                 255.41
No. Observations:                 208   AIC:                            -504.8
Df Residuals:                     205   BIC:                            -494.8
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      0.7178      0.005    144.563      0.000       0.708       0.728
Time           0.1035      0.005     20.737      0.000       0.094       0.113
School        -0.0151      0.005     -2.863      0.005      -0.026      -0.005
==============================================================================
Omnibus:                       10.647   Durbin-Watson:                   1.790
Prob(Omnibus):                  0.005   Jarque-Bera (JB):               14.335
Skew:                          -0.352   Prob(JB):                     0.000771
Kurtosis:                       4.076   Cond. No.                         1.15
==============================================================================
```

```
model = smf.ols('Quality ~ Time + Time.shift(-1) + School', data).fit()
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                Quality   R-squared:                       0.679
Model:                            OLS   Adj. R-squared:                  0.673
Method:                 Least Squares   F-statistic:                     131.0
Date:                Thu, 22 Nov 2018   Prob (F-statistic):           1.26e-45
Time:                        19:22:48   Log-Likelihood:                 230.49
No. Observations:                 190   AIC:                            -453.0
Df Residuals:                     186   BIC:                            -440.0
Df Model:                           3
Covariance Type:            nonrobust
================================================================================
                   coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
Intercept        0.7178      0.005    135.113      0.000       0.707       0.728
Time             0.1020      0.005     19.221      0.000       0.092       0.112
Time.shift(-1)  -0.0043      0.006     -0.763      0.446      -0.015       0.007
School          -0.0149      0.006     -2.601      0.010      -0.026      -0.004
==============================================================================
Omnibus:                        7.750   Durbin-Watson:                   1.797
Prob(Omnibus):                  0.021   Jarque-Bera (JB):                9.488
Skew:                          -0.301   Prob(JB):                      0.00870
Kurtosis:                       3.915   Cond. No.                         1.20
==============================================================================
```