

whoop_process

July 6, 2024

1 Setup

```
[82]: import pandas as pd
import numpy as np
import seaborn as sns
from matplotlib import pyplot as plt
import os

from sklearn.preprocessing import LabelEncoder
```

```
[83]: WHOOP_DIR = 'data/whoop/'
WHOOP_EDA_DIR = 'data/whoop/eda'
WHOOP_PROCESS_DIR = 'data/whoop/process/v1'
```

2 Import

2.1 Journal

```
[84]: df_journal_valid = pd.read_csv(
    os.path.join(WHOOP_EDA_DIR, 'journal_valid.csv'),
    parse_dates=[
        'Cycle start time',
        'cycle_date'
    ]
)
df_journal_valid
```

```
[84]:
```

	Cycle start time	Question text	Answered yes \
0	2024-06-17 00:35:19	Have any alcoholic drinks?	False
1	2024-06-17 00:35:19	Use CBD oil in any form?	True
2	2024-06-17 00:35:19	Avoid consuming processed foods?	False
3	2024-06-17 00:35:19	Take an ice bath?	False
4	2024-06-17 00:35:19	Have an injury or wound	False
...
5284	2022-09-14 21:52:24	Take prescription sleep medication?	True
5285	2022-09-14 00:00:00	Use CBD oil in any form?	False
5286	2022-09-14 00:00:00	Eat any food close to bedtime?	True

```

5287 2022-09-14 00:00:00 See direct sunlight upon waking up? True
5288 2022-09-14 00:00:00 Take prescription sleep medication? True

```

```

    cycle_date
0    2024-06-17
1    2024-06-17
2    2024-06-17
3    2024-06-17
4    2024-06-17
...
5284 2022-09-14
5285 2022-09-14
5286 2022-09-14
5287 2022-09-14
5288 2022-09-14

```

```
[5289 rows x 4 columns]
```

```
[85]: df_journal_valid.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5289 entries, 0 to 5288
Data columns (total 4 columns):
 #   Column                Non-Null Count  Dtype
---  -
 0   Cycle start time      5289 non-null   datetime64[ns]
 1   Question text         5289 non-null   object
 2   Answered yes          5289 non-null   bool
 3   cycle_date            5289 non-null   datetime64[ns]
dtypes: bool(1), datetime64[ns](2), object(1)
memory usage: 129.3+ KB

```

2.2 Physiological

```

[86]: df_phys_add = pd.read_csv(
    os.path.join(WHOOP_EDA_DIR, 'phys_add.csv'),
    parse_dates=[
        'Cycle start time',
        'Sleep onset',
        'cycle_start_time_adj',
        'cycle_date'
    ]
)
df_phys_add

```

```

[86]:      Cycle start time  Recovery score %  Resting heart rate (bpm) \
0    2024-06-15 23:12:59              0.84              52.0
1    2024-06-14 23:39:45              0.94              48.0

```

2	2024-06-14 00:07:47	0.83	50.0
3	2024-06-13 00:12:09	0.73	49.0
4	2024-06-11 23:11:54	0.74	50.0
..
627	2022-09-21 22:01:31	0.49	46.0
628	2022-09-20 22:20:13	0.48	47.0
629	2022-09-19 22:41:43	0.45	49.0
630	2022-09-18 22:20:02	0.55	49.0
631	2022-09-17 22:36:20	0.65	52.0

	Heart rate variability (ms)	Skin temp (celsius)	Blood oxygen %	\
0	92.0	33.30	0.9614	
1	95.0	34.30	0.9889	
2	87.0	34.00	0.9782	
3	80.0	33.20	0.9900	
4	78.0	33.90	0.9840	
..	
627	68.0	31.83	0.9661	
628	65.0	32.80	0.9669	
629	67.0	32.44	0.9584	
630	68.0	33.30	0.9733	
631	85.0	32.69	0.9691	

	Day Strain	Energy burned (cal)	Max HR (bpm)	Average HR (bpm)	...	\
0	12.6	2520.0	153.0	73.0	...	
1	17.3	3689.0	159.0	82.0	...	
2	14.8	2700.0	184.0	73.0	...	
3	12.8	2505.0	161.0	72.0	...	
4	10.1	2202.0	145.0	72.0	...	
..	
627	15.5	2753.0	179.0	64.0	...	
628	4.8	1794.0	119.0	57.0	...	
629	6.1	1871.0	153.0	58.0	...	
630	16.7	2863.0	175.0	69.0	...	
631	10.3	2104.0	154.0	66.0	...	

	Sleep efficiency %	Sleep consistency %	Awake %	Light sleep %	\
0	0.93	0.77	0.064912	0.405263	
1	0.89	0.90	0.105372	0.615702	
2	0.94	0.87	0.063047	0.478109	
3	0.89	0.81	0.131673	0.393238	
4	0.92	0.81	0.088283	0.569823	
..	
627	0.88	0.87	0.120287	0.456014	
628	0.85	0.89	0.147163	0.457447	
629	0.86	0.89	0.158111	0.472279	
630	0.88	0.87	0.122040	0.331512	

```
631          0.86          0.84  0.138462          0.513462
```

```

      Deep (SWS) %      REM %  Restorative sleep duration (min) \
0          0.240351  0.289474                                302.0
1          0.196281  0.082645                                135.0
2          0.220665  0.238179                                262.0
3          0.218861  0.256228                                267.0
4          0.210273  0.131621                                213.0
..          ...          ...                                ...
627         0.204668  0.219031                                236.0
628         0.234043  0.161348                                223.0
629         0.201232  0.168378                                180.0
630         0.253188  0.293260                                300.0
631         0.196154  0.151923                                181.0

```

```

      Restorative sleep %  cycle_start_time_adj  cycle_date
0          0.529825      2024-06-15 14:12:59  2024-06-16
1          0.278926      2024-06-14 14:39:45  2024-06-15
2          0.458844      2024-06-13 15:07:47  2024-06-14
3          0.475089      2024-06-12 15:12:09  2024-06-13
4          0.341894      2024-06-11 14:11:54  2024-06-12
..          ...          ...          ...
627         0.423698      2022-09-21 13:01:31  2022-09-22
628         0.395390      2022-09-20 13:20:13  2022-09-21
629         0.369610      2022-09-19 13:41:43  2022-09-20
630         0.546448      2022-09-18 13:20:02  2022-09-19
631         0.348077      2022-09-17 13:36:20  2022-09-18

```

```
[632 rows x 31 columns]
```

```
[87]: df_phys_add.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 632 entries, 0 to 631
Data columns (total 31 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Cycle start time                      632 non-null    datetime64[ns]
1   Recovery score %                     632 non-null    float64
2   Resting heart rate (bpm)             632 non-null    float64
3   Heart rate variability (ms)          632 non-null    float64
4   Skin temp (celsius)                 632 non-null    float64
5   Blood oxygen %                      632 non-null    float64
6   Day Strain                          632 non-null    float64
7   Energy burned (cal)                 632 non-null    float64
8   Max HR (bpm)                       632 non-null    float64
9   Average HR (bpm)                   632 non-null    float64
10  Sleep onset                         632 non-null    datetime64[ns]

```

11	Sleep performance %	632 non-null	float64
12	Respiratory rate (rpm)	632 non-null	float64
13	Asleep duration (min)	632 non-null	float64
14	In bed duration (min)	632 non-null	float64
15	Light sleep duration (min)	632 non-null	float64
16	Deep (SWS) duration (min)	632 non-null	float64
17	REM duration (min)	632 non-null	float64
18	Awake duration (min)	632 non-null	float64
19	Sleep need (min)	632 non-null	float64
20	Sleep debt (min)	632 non-null	float64
21	Sleep efficiency %	632 non-null	float64
22	Sleep consistency %	632 non-null	float64
23	Awake %	632 non-null	float64
24	Light sleep %	632 non-null	float64
25	Deep (SWS) %	632 non-null	float64
26	REM %	632 non-null	float64
27	Restorative sleep duration (min)	632 non-null	float64
28	Restorative sleep %	632 non-null	float64
29	cycle_start_time_adj	632 non-null	datetime64[ns]
30	cycle_date	632 non-null	datetime64[ns]

dtypes: datetime64[ns](4), float64(27)

memory usage: 153.2 KB

```
[88]: df_phys_rec = pd.read_csv(
    os.path.join(WHOOP_EDA_DIR, 'phys_rec.csv'),
    parse_dates=[
        'Cycle start time',
        'Sleep onset',
        'cycle_start_time_adj',
        'cycle_date',
        'recovery_date'
    ]
)
df_phys_rec
```

```
[88]:
```

	Cycle start time	Recovery score %	Resting heart rate (bpm)	\
0	2024-06-15 23:12:59	0.84	52.0	
1	2024-06-14 23:39:45	0.94	48.0	
2	2024-06-14 00:07:47	0.83	50.0	
3	2024-06-13 00:12:09	0.73	49.0	
4	2024-06-11 23:11:54	0.74	50.0	
..	
627	2022-09-21 22:01:31	0.49	46.0	
628	2022-09-20 22:20:13	0.48	47.0	
629	2022-09-19 22:41:43	0.45	49.0	
630	2022-09-18 22:20:02	0.55	49.0	
631	2022-09-17 22:36:20	0.65	52.0	

	Heart rate variability (ms)	Skin temp (celsius)	Blood oxygen %	\
0	92.0	33.30	0.9614	
1	95.0	34.30	0.9889	
2	87.0	34.00	0.9782	
3	80.0	33.20	0.9900	
4	78.0	33.90	0.9840	
..	
627	68.0	31.83	0.9661	
628	65.0	32.80	0.9669	
629	67.0	32.44	0.9584	
630	68.0	33.30	0.9733	
631	85.0	32.69	0.9691	

	Sleep onset	Sleep performance %	Respiratory rate (rpm)	\
0	2024-06-15 23:12:59	0.98	14.8	
1	2024-06-14 23:39:45	0.87	14.0	
2	2024-06-14 00:07:47	1.00	14.4	
3	2024-06-13 00:12:09	1.00	14.8	
4	2024-06-11 23:11:54	1.00	14.3	
..	
627	2022-09-21 22:01:31	1.00	14.2	
628	2022-09-20 22:20:13	0.93	13.7	
629	2022-09-19 22:41:43	0.78	13.7	
630	2022-09-18 22:20:02	0.95	14.0	
631	2022-09-17 22:36:20	0.85	13.8	

	Asleep duration (min)	...	Awake %	Restorative sleep duration (min)	\
0	533.0	...	0.064912	302.0	
1	433.0	...	0.105372	135.0	
2	535.0	...	0.063047	262.0	
3	488.0	...	0.131673	267.0	
4	568.0	...	0.088283	213.0	
..	
627	490.0	...	0.120287	236.0	
628	481.0	...	0.147163	223.0	
629	410.0	...	0.158111	180.0	
630	482.0	...	0.122040	300.0	
631	448.0	...	0.138462	181.0	

	Restorative sleep %	Sleep need (min)	Sleep debt (min)	\
0	0.529825	544.0	31.0	
1	0.278926	495.0	0.0	
2	0.458844	483.0	0.0	
3	0.475089	472.0	0.0	
4	0.341894	501.0	0.0	
..	

627	0.423698	486.0	20.0
628	0.395390	517.0	50.0
629	0.369610	525.0	14.0
630	0.546448	510.0	34.0
631	0.348077	526.0	23.0

	Sleep efficiency %	Sleep consistency %	cycle_start_time_adj \
0	0.93	0.77	2024-06-15 14:12:59
1	0.89	0.90	2024-06-14 14:39:45
2	0.94	0.87	2024-06-13 15:07:47
3	0.89	0.81	2024-06-12 15:12:09
4	0.92	0.81	2024-06-11 14:11:54
..
627	0.88	0.87	2022-09-21 13:01:31
628	0.85	0.89	2022-09-20 13:20:13
629	0.86	0.89	2022-09-19 13:41:43
630	0.88	0.87	2022-09-18 13:20:02
631	0.86	0.84	2022-09-17 13:36:20

	cycle_date	recovery_date
0	2024-06-16	2024-06-15
1	2024-06-15	2024-06-14
2	2024-06-14	2024-06-13
3	2024-06-13	2024-06-12
4	2024-06-12	2024-06-11
..
627	2022-09-22	2022-09-21
628	2022-09-21	2022-09-20
629	2022-09-20	2022-09-19
630	2022-09-19	2022-09-18
631	2022-09-18	2022-09-17

[632 rows x 28 columns]

```
[89]: df_phys_rec.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 632 entries, 0 to 631
```

```
Data columns (total 28 columns):
```

#	Column	Non-Null Count	Dtype
0	Cycle start time	632 non-null	datetime64[ns]
1	Recovery score %	632 non-null	float64
2	Resting heart rate (bpm)	632 non-null	float64
3	Heart rate variability (ms)	632 non-null	float64
4	Skin temp (celsius)	632 non-null	float64
5	Blood oxygen %	632 non-null	float64
6	Sleep onset	632 non-null	datetime64[ns]

7	Sleep performance %	632 non-null	float64
8	Respiratory rate (rpm)	632 non-null	float64
9	Asleep duration (min)	632 non-null	float64
10	In bed duration (min)	632 non-null	float64
11	Light sleep duration (min)	632 non-null	float64
12	Light sleep %	632 non-null	float64
13	Deep (SWS) duration (min)	632 non-null	float64
14	Deep (SWS) %	632 non-null	float64
15	REM duration (min)	632 non-null	float64
16	REM %	632 non-null	float64
17	Awake duration (min)	632 non-null	float64
18	Awake %	632 non-null	float64
19	Restorative sleep duration (min)	632 non-null	float64
20	Restorative sleep %	632 non-null	float64
21	Sleep need (min)	632 non-null	float64
22	Sleep debt (min)	632 non-null	float64
23	Sleep efficiency %	632 non-null	float64
24	Sleep consistency %	632 non-null	float64
25	cycle_start_time_adj	632 non-null	datetime64[ns]
26	cycle_date	632 non-null	datetime64[ns]
27	recovery_date	632 non-null	datetime64[ns]

dtypes: datetime64[ns](5), float64(23)
memory usage: 138.4 KB

```
[90]: df_phys_cycle = pd.read_csv(
    os.path.join(WHOOP_EDA_DIR, 'phys_cycle.csv'),
    parse_dates=[
        'Cycle start time',
        'cycle_start_time_adj',
        'cycle_date'
    ]
)
df_phys_cycle
```

```
[90]:
```

	Cycle start time	Day	Strain	Energy burned (cal)	Max HR (bpm)	\
0	2024-06-15 23:12:59		12.6	2520.0	153.0	
1	2024-06-14 23:39:45		17.3	3689.0	159.0	
2	2024-06-14 00:07:47		14.8	2700.0	184.0	
3	2024-06-13 00:12:09		12.8	2505.0	161.0	
4	2024-06-11 23:11:54		10.1	2202.0	145.0	
..	
627	2022-09-21 22:01:31		15.5	2753.0	179.0	
628	2022-09-20 22:20:13		4.8	1794.0	119.0	
629	2022-09-19 22:41:43		6.1	1871.0	153.0	
630	2022-09-18 22:20:02		16.7	2863.0	175.0	
631	2022-09-17 22:36:20		10.3	2104.0	154.0	

	Average HR (bpm)	cycle_start_time_adj	cycle_date
0	73.0	2024-06-15 14:12:59	2024-06-16
1	82.0	2024-06-14 14:39:45	2024-06-15
2	73.0	2024-06-13 15:07:47	2024-06-14
3	72.0	2024-06-12 15:12:09	2024-06-13
4	72.0	2024-06-11 14:11:54	2024-06-12
..
627	64.0	2022-09-21 13:01:31	2022-09-22
628	57.0	2022-09-20 13:20:13	2022-09-21
629	58.0	2022-09-19 13:41:43	2022-09-20
630	69.0	2022-09-18 13:20:02	2022-09-19
631	66.0	2022-09-17 13:36:20	2022-09-18

[632 rows x 7 columns]

```
[91]: df_phys_cycle.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 632 entries, 0 to 631
Data columns (total 7 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Cycle start time                      632 non-null    datetime64[ns]
1   Day Strain                            632 non-null    float64
2   Energy burned (cal)                  632 non-null    float64
3   Max HR (bpm)                         632 non-null    float64
4   Average HR (bpm)                     632 non-null    float64
5   cycle_start_time_adj                 632 non-null    datetime64[ns]
6   cycle_date                           632 non-null    datetime64[ns]
dtypes: datetime64[ns](3), float64(4)
memory usage: 34.7 KB
```

```
[92]: df_phys_comb = pd.read_csv(
    os.path.join(WHOOP_EDA_DIR, 'phys_comb.csv'),
    parse_dates=[
        'Sleep onset',
        'recovery_date',
        'Cycle start time',
        'cycle_start_time_adj',
        'cycle_date'
    ]
)
df_phys_comb
```

	Recovery score %	Resting heart rate (bpm)	Heart rate variability (ms) \
0	0.84	52.0	92.0
1	0.94	48.0	95.0
2	0.83	50.0	87.0

3	0.73	49.0	80.0
4	0.74	50.0	78.0
..
620	0.57	48.0	74.0
621	0.49	46.0	68.0
622	0.48	47.0	65.0
623	0.45	49.0	67.0
624	0.55	49.0	68.0

	Skin temp (celsius)	Blood oxygen %	Sleep onset \
0	33.30	0.9614	2024-06-15 23:12:59
1	34.30	0.9889	2024-06-14 23:39:45
2	34.00	0.9782	2024-06-14 00:07:47
3	33.20	0.9900	2024-06-13 00:12:09
4	33.90	0.9840	2024-06-11 23:11:54
..
620	33.13	0.9581	2022-09-22 23:23:45
621	31.83	0.9661	2022-09-21 22:01:31
622	32.80	0.9669	2022-09-20 22:20:13
623	32.44	0.9584	2022-09-19 22:41:43
624	33.30	0.9733	2022-09-18 22:20:02

	Sleep performance %	Respiratory rate (rpm)	Asleep duration (min) \
0	0.98	14.8	533.0
1	0.87	14.0	433.0
2	1.00	14.4	535.0
3	1.00	14.8	488.0
4	1.00	14.3	568.0
..
620	0.93	13.7	453.0
621	1.00	14.2	490.0
622	0.93	13.7	481.0
623	0.78	13.7	410.0
624	0.95	14.0	482.0

	In bed duration (min)	...	Sleep efficiency %	Sleep consistency % \
0	570.0	...	0.93	0.77
1	484.0	...	0.89	0.90
2	571.0	...	0.94	0.87
3	562.0	...	0.89	0.81
4	623.0	...	0.92	0.81
..
620	494.0	...	0.92	0.79
621	557.0	...	0.88	0.87
622	564.0	...	0.85	0.89
623	487.0	...	0.86	0.89
624	549.0	...	0.88	0.87

	recovery_date	Cycle start time	Day	Strain	Energy burned (cal)	\
0	2024-06-15	2024-06-14 23:39:45		17.3	3689.0	
1	2024-06-14	2024-06-14 00:07:47		14.8	2700.0	
2	2024-06-13	2024-06-13 00:12:09		12.8	2505.0	
3	2024-06-12	2024-06-11 23:11:54		10.1	2202.0	
4	2024-06-11	2024-06-10 23:15:19		15.8	3029.0	
..
620	2022-09-22	2022-09-21 22:01:31		15.5	2753.0	
621	2022-09-21	2022-09-20 22:20:13		4.8	1794.0	
622	2022-09-20	2022-09-19 22:41:43		6.1	1871.0	
623	2022-09-19	2022-09-18 22:20:02		16.7	2863.0	
624	2022-09-18	2022-09-17 22:36:20		10.3	2104.0	

	Max HR (bpm)	Average HR (bpm)	cycle_start_time_adj	cycle_date
0	159.0	82.0	2024-06-14 14:39:45	2024-06-15
1	184.0	73.0	2024-06-13 15:07:47	2024-06-14
2	161.0	72.0	2024-06-12 15:12:09	2024-06-13
3	145.0	72.0	2024-06-11 14:11:54	2024-06-12
4	177.0	73.0	2024-06-10 14:15:19	2024-06-11
..
620	179.0	64.0	2022-09-21 13:01:31	2022-09-22
621	119.0	57.0	2022-09-20 13:20:13	2022-09-21
622	153.0	58.0	2022-09-19 13:41:43	2022-09-20
623	175.0	69.0	2022-09-18 13:20:02	2022-09-19
624	154.0	66.0	2022-09-17 13:36:20	2022-09-18

[625 rows x 32 columns]

[93]: df_phys_comb.info()

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 625 entries, 0 to 624
```

```
Data columns (total 32 columns):
```

#	Column	Non-Null Count	Dtype
0	Recovery score %	625 non-null	float64
1	Resting heart rate (bpm)	625 non-null	float64
2	Heart rate variability (ms)	625 non-null	float64
3	Skin temp (celsius)	625 non-null	float64
4	Blood oxygen %	625 non-null	float64
5	Sleep onset	625 non-null	datetime64[ns]
6	Sleep performance %	625 non-null	float64
7	Respiratory rate (rpm)	625 non-null	float64
8	Asleep duration (min)	625 non-null	float64
9	In bed duration (min)	625 non-null	float64
10	Light sleep duration (min)	625 non-null	float64
11	Light sleep %	625 non-null	float64

12	Deep (SWS) duration (min)	625 non-null	float64
13	Deep (SWS) %	625 non-null	float64
14	REM duration (min)	625 non-null	float64
15	REM %	625 non-null	float64
16	Awake duration (min)	625 non-null	float64
17	Awake %	625 non-null	float64
18	Restorative sleep duration (min)	625 non-null	float64
19	Restorative sleep %	625 non-null	float64
20	Sleep need (min)	625 non-null	float64
21	Sleep debt (min)	625 non-null	float64
22	Sleep efficiency %	625 non-null	float64
23	Sleep consistency %	625 non-null	float64
24	recovery_date	625 non-null	datetime64[ns]
25	Cycle start time	625 non-null	datetime64[ns]
26	Day Strain	625 non-null	float64
27	Energy burned (cal)	625 non-null	float64
28	Max HR (bpm)	625 non-null	float64
29	Average HR (bpm)	625 non-null	float64
30	cycle_start_time_adj	625 non-null	datetime64[ns]
31	cycle_date	625 non-null	datetime64[ns]

dtypes: datetime64[ns](5), float64(27)
memory usage: 156.4 KB

2.3 Workouts

```
[94]: df_wo_add = pd.read_csv(
    os.path.join(WHOOP_EDA_DIR, 'wo_add.csv'),
    parse_dates=[
        'Cycle start time',
        'Workout start time',
        'cycle_start_time_adj',
        'cycle_date',
        'wo_date'
    ]
)
df_wo_add
```

```
[94]:
```

	Cycle start time	Workout start time	Duration (min)	Activity name \
0	2024-06-14 23:39:45	2024-06-15 12:52:30	89	Activity
1	2024-06-14 00:07:47	2024-06-14 17:30:00	87	Jiu Jitsu
2	2024-06-13 00:12:09	2024-06-13 18:08:57	76	Weightlifting
3	2024-06-10 23:15:19	2024-06-11 17:30:00	80	Jiu Jitsu
4	2024-06-10 23:15:19	2024-06-11 16:30:00	59	Martial Arts
..
444	2022-09-18 22:20:02	2022-09-19 17:25:36	77	Box Fitness
445	2022-09-16 21:53:01	2022-09-17 10:40:25	36	Powerlifting
446	2022-09-16 21:53:01	2022-09-17 09:27:55	61	Box Fitness

```

447 2022-09-15 22:58:46 2022-09-16 16:33:25        66    Box Fitness
448 2022-09-14 00:00:00 2022-09-14 17:22:28        86    Box Fitness

```

```

      Activity Strain  Energy burned (cal)  Max HR (bpm)  Average HR (bpm)  \
0              11.7             705.0             159             127
1              12.8             638.0             184             125
2               9.6             467.0             161             119
3              10.1             511.0             167             121
4              11.8             534.0             177             134
..              ...              ...              ...              ...
444             15.7             884.0             175             144
445              7.4             247.0             153             122
446             13.5             556.0             179             133
447             15.9             779.0             180             147
448             13.5             723.0             174             130

```

```

      HR Zone 1 %  HR Zone 2 %  HR Zone 3 %  HR Zone 4 %  HR Zone 5 %  \
0              0.04           0.78           0.18           0.00           0.00
1              0.28           0.47           0.11           0.09           0.04
2              0.36           0.45           0.19           0.00           0.00
3              0.30           0.48           0.17           0.02           0.00
4              0.04           0.51           0.29           0.15           0.01
..              ...           ...           ...           ...           ...
444             0.00           0.15           0.33           0.49           0.03
445             0.08           0.70           0.20           0.02           0.00
446             0.08           0.44           0.19           0.19           0.10
447             0.02           0.14           0.24           0.35           0.25
448             0.08           0.41           0.30           0.19           0.02

```

```

      cycle_start_time_adj  cycle_date  wo_date
0    2024-06-14 14:39:45 2024-06-15 2024-06-15
1    2024-06-13 15:07:47 2024-06-14 2024-06-14
2    2024-06-12 15:12:09 2024-06-13 2024-06-13
3    2024-06-10 14:15:19 2024-06-11 2024-06-11
4    2024-06-10 14:15:19 2024-06-11 2024-06-11
..              ...           ...           ...
444   2022-09-18 13:20:02 2022-09-19 2022-09-19
445   2022-09-16 12:53:01 2022-09-17 2022-09-17
446   2022-09-16 12:53:01 2022-09-17 2022-09-17
447   2022-09-15 13:58:46 2022-09-16 2022-09-16
448   2022-09-13 15:00:00 2022-09-14 2022-09-14

```

```
[449 rows x 16 columns]
```

```
[95]: df_wo_add.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 449 entries, 0 to 448

```

Data columns (total 16 columns):

#	Column	Non-Null Count	Dtype
0	Cycle start time	449 non-null	datetime64[ns]
1	Workout start time	449 non-null	datetime64[ns]
2	Duration (min)	449 non-null	int64
3	Activity name	449 non-null	object
4	Activity Strain	449 non-null	float64
5	Energy burned (cal)	449 non-null	float64
6	Max HR (bpm)	449 non-null	int64
7	Average HR (bpm)	449 non-null	int64
8	HR Zone 1 %	449 non-null	float64
9	HR Zone 2 %	449 non-null	float64
10	HR Zone 3 %	449 non-null	float64
11	HR Zone 4 %	449 non-null	float64
12	HR Zone 5 %	449 non-null	float64
13	cycle_start_time_adj	449 non-null	datetime64[ns]
14	cycle_date	449 non-null	datetime64[ns]
15	wo_date	449 non-null	datetime64[ns]

dtypes: datetime64[ns](5), float64(7), int64(3), object(1)

memory usage: 56.2+ KB

3 Activity Classification

3.1 Rename cols

```
[96]: df_wo_pr = df_wo_add.copy()

DF_WO_COL_NAMES = [
    'cycle_start_time',
    'wo_start_time',
    'duration_min',
    'activity_name', # label
    'activity_strain',
    'calories',
    'hr_max',
    'hr_avg',
    'hr_zone_1_p',
    'hr_zone_2_p',
    'hr_zone_3_p',
    'hr_zone_4_p',
    'hr_zone_5_p',
    'cycle_start_time_adj',
    'cycle_date',
    'wo_date'
]
```

```
df_wo_pr.columns = DF_WO_COL_NAMES
df_wo_pr.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 449 entries, 0 to 448
Data columns (total 16 columns):
#   Column                Non-Null Count  Dtype
---  -
0   cycle_start_time      449 non-null   datetime64[ns]
1   wo_start_time         449 non-null   datetime64[ns]
2   duration_min          449 non-null   int64
3   activity_name         449 non-null   object
4   activity_strain       449 non-null   float64
5   calories              449 non-null   float64
6   hr_max                449 non-null   int64
7   hr_avg                449 non-null   int64
8   hr_zone_1_p           449 non-null   float64
9   hr_zone_2_p           449 non-null   float64
10  hr_zone_3_p           449 non-null   float64
11  hr_zone_4_p           449 non-null   float64
12  hr_zone_5_p           449 non-null   float64
13  cycle_start_time_adj  449 non-null   datetime64[ns]
14  cycle_date            449 non-null   datetime64[ns]
15  wo_date               449 non-null   datetime64[ns]
dtypes: datetime64[ns](5), float64(7), int64(3), object(1)
memory usage: 56.2+ KB
```

```
[97]: df_wo_pr['cycle_date'].max() - df_wo_pr['cycle_date'].min()
```

```
[97]: Timedelta('640 days 00:00:00')
```

3.2 Encode datetimes

3.2.1 Drop irrelevant datetimes

```
[98]: WO_DROP_DATETIMES = [
        'cycle_start_time',
        'cycle_start_time_adj',
        'cycle_date',
        'wo_date'
    ]

df_wo_dt = df_wo_pr.copy().drop(columns=WO_DROP_DATETIMES)
df_wo_dt
```

```
[98]:
```

	wo_start_time	duration_min	activity_name	activity_strain \
0	2024-06-15 12:52:30	89	Activity	11.7
1	2024-06-14 17:30:00	87	Jiu Jitsu	12.8

2	2024-06-13 18:08:57	76	Weightlifting	9.6
3	2024-06-11 17:30:00	80	Jiu Jitsu	10.1
4	2024-06-11 16:30:00	59	Martial Arts	11.8
..
444	2022-09-19 17:25:36	77	Box Fitness	15.7
445	2022-09-17 10:40:25	36	Powerlifting	7.4
446	2022-09-17 09:27:55	61	Box Fitness	13.5
447	2022-09-16 16:33:25	66	Box Fitness	15.9
448	2022-09-14 17:22:28	86	Box Fitness	13.5

	calories	hr_max	hr_avg	hr_zone_1_p	hr_zone_2_p	hr_zone_3_p	\
0	705.0	159	127	0.04	0.78	0.18	
1	638.0	184	125	0.28	0.47	0.11	
2	467.0	161	119	0.36	0.45	0.19	
3	511.0	167	121	0.30	0.48	0.17	
4	534.0	177	134	0.04	0.51	0.29	
..	
444	884.0	175	144	0.00	0.15	0.33	
445	247.0	153	122	0.08	0.70	0.20	
446	556.0	179	133	0.08	0.44	0.19	
447	779.0	180	147	0.02	0.14	0.24	
448	723.0	174	130	0.08	0.41	0.30	

	hr_zone_4_p	hr_zone_5_p
0	0.00	0.00
1	0.09	0.04
2	0.00	0.00
3	0.02	0.00
4	0.15	0.01
..
444	0.49	0.03
445	0.02	0.00
446	0.19	0.10
447	0.35	0.25
448	0.19	0.02

[449 rows x 12 columns]

3.2.2 Extract & encode datetime features

```
[99]: df_wo_dt['wo_date_ord'] = df_wo_dt['wo_start_time'].dt.date.apply(lambda x: x.
    ↳ toordinal())

df_wo_dt['wo_day_of_week'] = df_wo_dt['wo_start_time'].dt.dayofweek
df_wo_dt['wo_day_of_week_sin'] = np.sin(2 * np.pi * df_wo_dt['wo_day_of_week'] /
    ↳ 7)
```



```

df_wo_dt['wo_day_of_week_cos'] = np.cos(2 * np.pi * df_wo_dt['wo_day_of_week'] /
↳ 7)

df_wo_dt['wo_hour'] = df_wo_dt['wo_start_time'].dt.hour
df_wo_dt['wo_minute'] = df_wo_dt['wo_start_time'].dt.minute
df_wo_dt['wo_start_min'] = df_wo_dt['wo_hour'] * 60 + df_wo_dt['wo_minute']
df_wo_dt['wo_start_min_sin'] = np.sin(2 * np.pi * df_wo_dt['wo_start_min'] /
↳ 1440)
df_wo_dt['wo_start_min_cos'] = np.cos(2 * np.pi * df_wo_dt['wo_start_min'] /
↳ 1440)

WO_INT_DT_FIELDS = [
    'wo_start_time',
    'wo_day_of_week',
    'wo_start_min',
    'wo_hour',
    'wo_minute'
]

df_wo_dt.drop(columns=WO_INT_DT_FIELDS, inplace=True)

```

[100]: df_wo_dt

```

[100]:
   duration_min  activity_name  activity_strain  calories  hr_max  hr_avg  \
0             89      Activity             11.7     705.0    159    127
1             87      Jiu Jitsu             12.8     638.0    184    125
2             76  Weightlifting              9.6     467.0    161    119
3             80      Jiu Jitsu             10.1     511.0    167    121
4             59  Martial Arts             11.8     534.0    177    134
..            ...            ...              ...      ...      ...
444            77    Box Fitness             15.7     884.0    175    144
445            36  Powerlifting              7.4     247.0    153    122
446            61    Box Fitness             13.5     556.0    179    133
447            66    Box Fitness             15.9     779.0    180    147
448            86    Box Fitness             13.5     723.0    174    130

   hr_zone_1_p  hr_zone_2_p  hr_zone_3_p  hr_zone_4_p  hr_zone_5_p  \
0           0.04         0.78         0.18         0.00         0.00
1           0.28         0.47         0.11         0.09         0.04
2           0.36         0.45         0.19         0.00         0.00
3           0.30         0.48         0.17         0.02         0.00
4           0.04         0.51         0.29         0.15         0.01
..            ...            ...              ...      ...      ...
444          0.00         0.15         0.33         0.49         0.03
445          0.08         0.70         0.20         0.02         0.00
446          0.08         0.44         0.19         0.19         0.10
447          0.02         0.14         0.24         0.35         0.25

```

```

448          0.08          0.41          0.30          0.19          0.02

      wo_date_ord wo_day_of_week_sin wo_day_of_week_cos wo_start_min_sin \
0          739052          -0.974928          -0.222521          -0.224951
1          739051          -0.433884          -0.900969          -0.991445
2          739050           0.433884          -0.900969          -0.999391
3          739048           0.781831           0.623490          -0.991445
4          739048           0.781831           0.623490          -0.923880
..          ...
444         738417           0.000000           1.000000          -0.988362
445         738415          -0.974928          -0.222521           0.342020
446         738415          -0.974928          -0.222521           0.619094
447         738414          -0.433884          -0.900969          -0.928810
448         738412           0.974928          -0.222521          -0.986286

      wo_start_min_cos
0          -0.974370
1          -0.130526
2           0.034899
3          -0.130526
4          -0.382683
..          ...
444          -0.152123
445          -0.939693
446          -0.785317
447          -0.370557
448          -0.165048

```

[449 rows x 16 columns]

3.3 Create different class sets

3.3.1 Full class set

Keep all activities with no combination of similar classes.

Poor model performance is expected here, due to massive class imbalances.

```

[101]: df_wo_full_18 = df_wo_dt.copy()
df_wo_full_18['activity_name'].value_counts(sort=True).reset_index()

```

```

[101]:
      activity_name  count
0  Weightlifting     161
1      Jiu Jitsu      96
2  Powerlifting      35
3      Activity       33
4   Box Fitness       33
5     Wrestling       20

```

6	Yoga	17
7	Kickboxing	12
8	Martial Arts	12
9	Spin	7
10	Assault Bike	7
11	Manual Labor	4
12	Boxing	4
13	Operations - Tactical	3
14	Yard Work	2
15	Dance	1
16	Paintball	1
17	Lacrosse	1

3.3.2 Full combined class set

Keep all activities, but combine into 3 similar classes for relative class balance.

With only 3 relatively balanced classes, we should expect much better classification performance. Unfortunately, predicting between just three activity types would likely not have much generalizable benefit to a broader population of users who partake in a different distribution of activities, but that is an obvious limitation of only having access to my personal data. With access to a larger and more diverse dataset, we should expect to achieve similar performance on a broader range of activities.

```
[102]: df_wo_full_3 = df_wo_dt.copy()
df_wo_full_3['activity_name'].value_counts(sort=True).reset_index()
```

```
[102]:
```

	activity_name	count
0	Weightlifting	161
1	Jiu Jitsu	96
2	Powerlifting	35
3	Activity	33
4	Box Fitness	33
5	Wrestling	20
6	Yoga	17
7	Kickboxing	12
8	Martial Arts	12
9	Spin	7
10	Assault Bike	7
11	Manual Labor	4
12	Boxing	4
13	Operations - Tactical	3
14	Yard Work	2
15	Dance	1
16	Paintball	1
17	Lacrosse	1

```
[103]: df_wo_full_3['activity_name'] = df_wo_full_3['activity_name'].
        ↪replace('Powerlifting', 'Weightlifting')
df_wo_full_3['activity_name'].value_counts(sort=True).reset_index()
```

```
[103]:
```

	activity_name	count
0	Weightlifting	196
1	Jiu Jitsu	96
2	Activity	33
3	Box Fitness	33
4	Wrestling	20
5	Yoga	17
6	Kickboxing	12
7	Martial Arts	12
8	Assault Bike	7
9	Spin	7
10	Manual Labor	4
11	Boxing	4
12	Operations - Tactical	3
13	Yard Work	2
14	Dance	1
15	Paintball	1
16	Lacrosse	1

```
[104]: MARTIAL_ARTS_ACTIVITES = [
        'Jiu Jitsu',
        'Wrestling',
        'Kickboxing',
        'Boxing'
    ]
for activity in MARTIAL_ARTS_ACTIVITES:
    df_wo_full_3['activity_name'] = df_wo_full_3['activity_name'].
    ↪replace(activity, 'Martial Arts')

df_wo_full_3['activity_name'].value_counts(sort=True).reset_index()
```

```
[104]:
```

	activity_name	count
0	Weightlifting	196
1	Martial Arts	144
2	Activity	33
3	Box Fitness	33
4	Yoga	17
5	Spin	7
6	Assault Bike	7
7	Manual Labor	4
8	Operations - Tactical	3
9	Yard Work	2
10	Dance	1

11	Paintball	1
12	Lacrosse	1

```
[105]: OTHER_ACTIVITIES = [activity for activity in df_wo_full_3['activity_name'].
    ↪unique() if activity not in ['Weightlifting', 'Martial Arts']]
for activity in OTHER_ACTIVITIES:
    df_wo_full_3['activity_name'] = df_wo_full_3['activity_name'].
    ↪replace(activity, 'Other')

df_wo_full_3['activity_name'].value_counts(sort=True).reset_index()
```

```
[105]:   activity_name  count
0  Weightlifting    196
1   Martial Arts    144
2         Other     109
```

3.3.3 Limited class set (count >10)

As an alternative to combining class labels in order to achieve better relative class balance, I will also test a class set that simply drops the most uncommon classes. I also drop the generic ‘Activity’ class, as this label contains miscellaneous activities that may or may not belong together. This method may introduce less noise within class labels, as we’re not combining distinct activities into single classes, but it comes with the significant drawbacks of greater class imbalance and a smaller dataset overall.

```
[106]: df_wo_lim_8 = df_wo_dt.copy()
df_wo_lim_8['activity_name'].value_counts(sort=True).reset_index()
```

```
[106]:   activity_name  count
0   Weightlifting    161
1      Jiu Jitsu     96
2   Powerlifting     35
3      Activity     33
4   Box Fitness     33
5    Wrestling     20
6        Yoga      17
7   Kickboxing     12
8   Martial Arts     12
9        Spin       7
10  Assault Bike     7
11  Manual Labor     4
12      Boxing       4
13 Operations - Tactical  3
14      Yard Work     2
15      Dance        1
16   Paintball       1
17   Lacrosse        1
```

```
[107]: COUNT_THRESH_10 = 10
wo_lim_8_class_counts = df_wo_lim_8['activity_name'].value_counts(sort=True).
↳reset_index()
df_wo_lim_8_classes =
↳wo_lim_8_class_counts['activity_name'][wo_lim_8_class_counts['count'] >=
↳COUNT_THRESH_10].to_list()
df_wo_lim_8_classes.remove('Activity') # 'Activity' represents various
↳unlabeled activities, and should therefore be excluded

df_wo_lim_8 = df_wo_lim_8[df_wo_lim_8['activity_name'].
↳isin(df_wo_lim_8_classes)]
df_wo_lim_8['activity_name'].value_counts(sort=True).reset_index()
```

```
[107]:   activity_name  count
0  Weightlifting    161
1      Jiu Jitsu     96
2  Powerlifting     35
3   Box Fitness     33
4    Wrestling     20
5         Yoga      17
6  Martial Arts     12
7   Kickboxing     12
```

3.4 Encode Classes

```
[108]: wo_activity_encoder = LabelEncoder()
```

```
[109]: df_wo_full_18['activity_code'] = wo_activity_encoder.
↳fit_transform(df_wo_full_18['activity_name'])
df_wo_full_18['activity_code'].unique()
```

```
[109]: array([ 0,  5, 14,  9,  3, 10,  8,  4, 13,  6,  1, 15,  2, 17, 11, 12,  7,
          16])
```

```
[110]: df_wo_full_3['activity_code'] = wo_activity_encoder.
↳fit_transform(df_wo_full_3['activity_name'])
df_wo_full_3['activity_code'].unique()
```

```
[110]: array([1, 0, 2])
```

```
[111]: df_wo_lim_8['activity_code'] = wo_activity_encoder.
↳fit_transform(df_wo_lim_8['activity_name'])
df_wo_lim_8['activity_code'].unique()
```

```
[111]: array([1, 5, 3, 2, 6, 0, 7, 4])
```

3.5 Export

```
[112]: df_wo_full_18.to_csv(
        os.path.join(WHOOP_PROCESS_DIR, 'wo_full_18.csv'),
        index=False
    )
```

```
[113]: df_wo_full_3.to_csv(
        os.path.join(WHOOP_PROCESS_DIR, 'wo_full_3.csv'),
        index=False
    )
```

```
[114]: df_wo_lim_8.to_csv(
        os.path.join(WHOOP_PROCESS_DIR, 'wo_lim_8.csv'),
        index=False
    )
```

4 Recovery Score Regression

I will perform two rounds of regression to predict Whoop's recovery scores.

First, I will just use the daily strain and recovery metrics already included in the Physiology datasets to predict the following nights' Recovery scores.

Second, I will attempt to improve performance by also including activities and journal entries for the previous day to predict Recovery scores.

4.1 Daily strain metrics

4.1.1 Rename cols

```
[116]: df_phys = df_phys_comb.copy()

DF_PHYS_COL_NAMES = [
    'recovery_score_p', # label

    # recovery features
    'hr_rest',
    'hr_var',
    'skin_temp_c',
    'blood_ox_p',
    'sleep_onset_time',
    'sleep_performance_p',
    'resp_rate',
    'asleep_min',
    'in_bed_min',
    'light_sleep_min',
    'light_sleep_p',
    'deep_sleep_min',
```

```

'deep_sleep_p',
'rem_sleep_min',
'rem_sleep_p',
'awake_min',
'awake_p',
'restorative_sleep_min',
'restorative_sleep_p',
'sleep_need_min',
'sleep_debt_min',
'sleep_efficiency_p',
'sleep_consistency_p',
'recovery_date',

# strain features
'cycle_start_time',
'day_strain', # whoop calculated metric -- probably drop
'calories',
'hr_max',
'hr_avg',
'cycle_start_time_adj',
'cycle_date'
]

df_phys.columns = DF_PHYS_COL_NAMES
df_phys.info()

```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 625 entries, 0 to 624
```

```
Data columns (total 32 columns):
```

#	Column	Non-Null Count	Dtype
0	recovery_score_p	625 non-null	float64
1	hr_rest	625 non-null	float64
2	hr_var	625 non-null	float64
3	skin_temp_c	625 non-null	float64
4	blood_ox_p	625 non-null	float64
5	sleep_onset_time	625 non-null	datetime64[ns]
6	sleep_performance_p	625 non-null	float64
7	resp_rate	625 non-null	float64
8	asleep_min	625 non-null	float64
9	in_bed_min	625 non-null	float64
10	light_sleep_min	625 non-null	float64
11	light_sleep_p	625 non-null	float64
12	deep_sleep_min	625 non-null	float64
13	deep_sleep_p	625 non-null	float64
14	rem_sleep_min	625 non-null	float64
15	rem_sleep_p	625 non-null	float64
16	awake_min	625 non-null	float64


```

17  awake_p                625 non-null    float64
18  restorative_sleep_min  625 non-null    float64
19  restorative_sleep_p    625 non-null    float64
20  sleep_need_min         625 non-null    float64
21  sleep_debt_min         625 non-null    float64
22  sleep_efficiency_p     625 non-null    float64
23  sleep_consistency_p    625 non-null    float64
24  recovery_date          625 non-null    datetime64[ns]
25  cycle_start_time       625 non-null    datetime64[ns]
26  day_strain             625 non-null    float64
27  calories               625 non-null    float64
28  hr_max                 625 non-null    float64
29  hr_avg                 625 non-null    float64
30  cycle_start_time_adj   625 non-null    datetime64[ns]
31  cycle_date             625 non-null    datetime64[ns]
dtypes: datetime64[ns](5), float64(27)
memory usage: 156.4 KB

```

```
[117]: df_phys[df_phys['sleep_onset_time'].dt.date != df_phys['cycle_date']]
```

```

[117]:      recovery_score_p  hr_rest  hr_var  skin_temp_c  blood_ox_p  \
2          0.83      50.0   87.0      34.00      0.9782
3          0.73      49.0   80.0      33.20      0.9900
7          0.42      61.0   45.0      33.90      0.9788
8          0.20      70.0   39.0      33.50      0.9650
9          0.14      73.0   31.0      34.79      0.9779
..          ...      ...   ...      ...      ...
568        0.54      52.0   63.0      34.30      0.9805
569        0.59      48.0   63.0      33.60      0.9869
583        0.53      52.0   71.0      34.30      0.9686
594        0.49      47.0   70.0      33.40      0.9733
607        0.16      55.0   53.0      33.40      0.9708

      sleep_onset_time  sleep_performance_p  resp_rate  asleep_min  \
2  2024-06-14 00:07:47          1.00      14.4      535.0
3  2024-06-13 00:12:09          1.00      14.8      488.0
7  2024-06-07 01:32:10          0.94      14.9      499.0
8  2024-06-06 00:33:45          0.74      16.5      377.0
9  2024-06-05 02:31:30          0.85      15.5      417.0
..          ...      ...      ...      ...
568 2022-11-14 00:01:29          0.90      14.7      435.0
569 2022-11-13 00:09:09          1.00      14.1      473.0
583 2022-10-30 01:46:48          0.92      14.5      500.0
594 2022-10-19 00:11:04          0.81      14.8      394.0
607 2022-10-06 01:40:06          0.70      14.9      379.0

      in_bed_min  ...  sleep_efficiency_p  sleep_consistency_p  recovery_date  \

```

2	571.0	...	0.94	0.87	2024-06-13
3	562.0	...	0.89	0.81	2024-06-12
7	595.0	...	0.84	0.66	2024-06-06
8	447.0	...	0.86	0.70	2024-06-05
9	501.0	...	0.83	0.67	2024-06-04
..
568	464.0	...	0.94	0.78	2022-11-13
569	533.0	...	0.89	0.71	2022-11-12
583	556.0	...	0.90	0.33	2022-10-29
594	438.0	...	0.91	0.76	2022-10-18
607	440.0	...	0.87	0.55	2022-10-05

	cycle_start_time	day_strain	calories	hr_max	hr_avg	\
2	2024-06-13 00:12:09	12.8	2505.0	161.0	72.0	
3	2024-06-11 23:11:54	10.1	2202.0	145.0	72.0	
7	2024-06-06 00:33:45	7.4	2034.0	127.0	79.0	
8	2024-06-05 02:31:30	10.5	2021.0	139.0	81.0	
9	2024-06-04 01:04:32	7.4	2074.0	135.0	79.0	
..	
568	2022-11-13 00:09:09	12.9	2426.0	156.0	66.0	
569	2022-11-11 22:21:22	10.4	2311.0	138.0	67.0	
583	2022-10-28 22:42:12	17.0	3364.0	168.0	74.0	
594	2022-10-17 22:16:39	8.5	2166.0	153.0	60.0	
607	2022-10-04 22:43:37	18.0	3312.0	182.0	72.0	

	cycle_start_time_adj	cycle_date
2	2024-06-12 15:12:09	2024-06-13
3	2024-06-11 14:11:54	2024-06-12
7	2024-06-05 15:33:45	2024-06-06
8	2024-06-04 17:31:30	2024-06-05
9	2024-06-03 16:04:32	2024-06-04
..
568	2022-11-12 15:09:09	2022-11-13
569	2022-11-11 13:21:22	2022-11-12
583	2022-10-28 13:42:12	2022-10-29
594	2022-10-17 13:16:39	2022-10-18
607	2022-10-04 13:43:37	2022-10-05

[132 rows x 32 columns]

4.1.2 Encode datetimes

Drop irrelevant datetimes

```
[118]: PHYS_DROP_DATETIMES = [
        'recovery_date',
        'cycle_start_time_adj'
    ]
```

```
df_phys_dt = df_phys.copy().drop(columns=PHYS_DROP_DATETIMES)
df_phys_dt
```

```
[118]:
```

	recovery_score_p	hr_rest	hr_var	skin_temp_c	blood_ox_p	\
0	0.84	52.0	92.0	33.30	0.9614	
1	0.94	48.0	95.0	34.30	0.9889	
2	0.83	50.0	87.0	34.00	0.9782	
3	0.73	49.0	80.0	33.20	0.9900	
4	0.74	50.0	78.0	33.90	0.9840	
..	
620	0.57	48.0	74.0	33.13	0.9581	
621	0.49	46.0	68.0	31.83	0.9661	
622	0.48	47.0	65.0	32.80	0.9669	
623	0.45	49.0	67.0	32.44	0.9584	
624	0.55	49.0	68.0	33.30	0.9733	

	sleep_onset_time	sleep_performance_p	resp_rate	asleep_min	\
0	2024-06-15 23:12:59	0.98	14.8	533.0	
1	2024-06-14 23:39:45	0.87	14.0	433.0	
2	2024-06-14 00:07:47	1.00	14.4	535.0	
3	2024-06-13 00:12:09	1.00	14.8	488.0	
4	2024-06-11 23:11:54	1.00	14.3	568.0	
..	
620	2022-09-22 23:23:45	0.93	13.7	453.0	
621	2022-09-21 22:01:31	1.00	14.2	490.0	
622	2022-09-20 22:20:13	0.93	13.7	481.0	
623	2022-09-19 22:41:43	0.78	13.7	410.0	
624	2022-09-18 22:20:02	0.95	14.0	482.0	

	in_bed_min	...	sleep_need_min	sleep_debt_min	sleep_efficiency_p	\
0	570.0	...	544.0	31.0	0.93	
1	484.0	...	495.0	0.0	0.89	
2	571.0	...	483.0	0.0	0.94	
3	562.0	...	472.0	0.0	0.89	
4	623.0	...	501.0	0.0	0.92	
..	
620	494.0	...	486.0	0.0	0.92	
621	557.0	...	486.0	20.0	0.88	
622	564.0	...	517.0	50.0	0.85	
623	487.0	...	525.0	14.0	0.86	
624	549.0	...	510.0	34.0	0.88	

	sleep_consistency_p	cycle_start_time	day_strain	calories	hr_max	\
0	0.77	2024-06-14 23:39:45	17.3	3689.0	159.0	
1	0.90	2024-06-14 00:07:47	14.8	2700.0	184.0	
2	0.87	2024-06-13 00:12:09	12.8	2505.0	161.0	

3	0.81	2024-06-11	23:11:54	10.1	2202.0	145.0
4	0.81	2024-06-10	23:15:19	15.8	3029.0	177.0
..	
620	0.79	2022-09-21	22:01:31	15.5	2753.0	179.0
621	0.87	2022-09-20	22:20:13	4.8	1794.0	119.0
622	0.89	2022-09-19	22:41:43	6.1	1871.0	153.0
623	0.89	2022-09-18	22:20:02	16.7	2863.0	175.0
624	0.87	2022-09-17	22:36:20	10.3	2104.0	154.0

	hr_avg	cycle_date
0	82.0	2024-06-15
1	73.0	2024-06-14
2	72.0	2024-06-13
3	72.0	2024-06-12
4	73.0	2024-06-11
..
620	64.0	2022-09-22
621	57.0	2022-09-21
622	58.0	2022-09-20
623	69.0	2022-09-19
624	66.0	2022-09-18

[625 rows x 30 columns]

Extract & encode datetime features

```
[119]: df_phys_dt['cycle_date_ord'] = df_phys_dt['cycle_date'].apply(lambda x: x.
    ↳toordinal())

df_phys_dt['cycle_day_of_week'] = df_phys_dt['cycle_date'].dt.dayofweek
df_phys_dt['cycle_day_of_week_sin'] = np.sin(2 * np.pi *
    ↳df_phys_dt['cycle_day_of_week'] / 7)
df_phys_dt['cycle_day_of_week_cos'] = np.cos(2 * np.pi *
    ↳df_phys_dt['cycle_day_of_week'] / 7)

df_phys_dt['sleep_onset_hour'] = df_phys_dt['sleep_onset_time'].dt.hour
df_phys_dt['sleep_onset_minute'] = df_phys_dt['sleep_onset_time'].dt.minute
df_phys_dt['sleep_onset_min'] = df_phys_dt['sleep_onset_hour'] * 60 +
    ↳df_phys_dt['sleep_onset_minute']
df_phys_dt['sleep_onset_min_sin'] = np.sin(2 * np.pi *
    ↳df_phys_dt['sleep_onset_min'] / 1440)
df_phys_dt['sleep_onset_min_cos'] = np.cos(2 * np.pi *
    ↳df_phys_dt['sleep_onset_min'] / 1440)

df_phys_dt['prev_sleep_onset_hour'] = df_phys_dt['cycle_start_time'].dt.hour
df_phys_dt['prev_sleep_onset_minute'] = df_phys_dt['cycle_start_time'].dt.minute
```

```

df_phys_dt['prev_sleep_onset_min'] = df_phys_dt['prev_sleep_onset_hour'] * 60 +
    df_phys_dt['prev_sleep_onset_minute']
df_phys_dt['prev_sleep_onset_min_sin'] = np.sin(2 * np.pi *
    df_phys_dt['prev_sleep_onset_min'] / 1440)
df_phys_dt['prev_sleep_onset_min_cos'] = np.cos(2 * np.pi *
    df_phys_dt['prev_sleep_onset_min'] / 1440)

PHYS_INT_DT_FIELDS = [
    'cycle_date',
    'cycle_day_of_week',
    'sleep_onset_time',
    'sleep_onset_hour',
    'sleep_onset_minute',
    'sleep_onset_min',
    'cycle_start_time',
    'prev_sleep_onset_hour',
    'prev_sleep_onset_minute',
    'prev_sleep_onset_min'
]

df_phys_dt.drop(columns=PHYS_INT_DT_FIELDS, inplace=True)
df_phys_dt.info()

```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 625 entries, 0 to 624
```

```
Data columns (total 34 columns):
```

#	Column	Non-Null Count	Dtype
0	recovery_score_p	625 non-null	float64
1	hr_rest	625 non-null	float64
2	hr_var	625 non-null	float64
3	skin_temp_c	625 non-null	float64
4	blood_ox_p	625 non-null	float64
5	sleep_performance_p	625 non-null	float64
6	resp_rate	625 non-null	float64
7	asleep_min	625 non-null	float64
8	in_bed_min	625 non-null	float64
9	light_sleep_min	625 non-null	float64
10	light_sleep_p	625 non-null	float64
11	deep_sleep_min	625 non-null	float64
12	deep_sleep_p	625 non-null	float64
13	rem_sleep_min	625 non-null	float64
14	rem_sleep_p	625 non-null	float64
15	awake_min	625 non-null	float64
16	awake_p	625 non-null	float64
17	restorative_sleep_min	625 non-null	float64
18	restorative_sleep_p	625 non-null	float64

```

19 sleep_need_min          625 non-null    float64
20 sleep_debt_min          625 non-null    float64
21 sleep_efficiency_p      625 non-null    float64
22 sleep_consistency_p     625 non-null    float64
23 day_strain              625 non-null    float64
24 calories                625 non-null    float64
25 hr_max                  625 non-null    float64
26 hr_avg                  625 non-null    float64
27 cycle_date_ord          625 non-null    int64
28 cycle_day_of_week_sin   625 non-null    float64
29 cycle_day_of_week_cos   625 non-null    float64
30 sleep_onset_min_sin     625 non-null    float64
31 sleep_onset_min_cos     625 non-null    float64
32 prev_sleep_onset_min_sin 625 non-null    float64
33 prev_sleep_onset_min_cos 625 non-null    float64
dtypes: float64(33), int64(1)
memory usage: 166.1 KB

```

```
[120]: df_phys_dt
```

```

[120]:      recovery_score_p  hr_rest  hr_var  skin_temp_c  blood_ox_p  \
0          0.84      52.0   92.0      33.30      0.9614
1          0.94      48.0   95.0      34.30      0.9889
2          0.83      50.0   87.0      34.00      0.9782
3          0.73      49.0   80.0      33.20      0.9900
4          0.74      50.0   78.0      33.90      0.9840
..          ...      ...   ...      ...      ...
620        0.57      48.0   74.0      33.13      0.9581
621        0.49      46.0   68.0      31.83      0.9661
622        0.48      47.0   65.0      32.80      0.9669
623        0.45      49.0   67.0      32.44      0.9584
624        0.55      49.0   68.0      33.30      0.9733

      sleep_performance_p  resp_rate  asleep_min  in_bed_min  light_sleep_min  \
0          0.98      14.8      533.0      570.0      231.0
1          0.87      14.0      433.0      484.0      298.0
2          1.00      14.4      535.0      571.0      273.0
3          1.00      14.8      488.0      562.0      221.0
4          1.00      14.3      568.0      623.0      355.0
..          ...      ...   ...      ...      ...
620        0.93      13.7      453.0      494.0      271.0
621        1.00      14.2      490.0      557.0      254.0
622        0.93      13.7      481.0      564.0      258.0
623        0.78      13.7      410.0      487.0      230.0
624        0.95      14.0      482.0      549.0      182.0

      ...  calories  hr_max  hr_avg  cycle_date_ord  cycle_day_of_week_sin  \

```

0	...	3689.0	159.0	82.0	739052	-0.974928
1	...	2700.0	184.0	73.0	739051	-0.433884
2	...	2505.0	161.0	72.0	739050	0.433884
3	...	2202.0	145.0	72.0	739049	0.974928
4	...	3029.0	177.0	73.0	739048	0.781831
..
620	...	2753.0	179.0	64.0	738420	0.433884
621	...	1794.0	119.0	57.0	738419	0.974928
622	...	1871.0	153.0	58.0	738418	0.781831
623	...	2863.0	175.0	69.0	738417	0.000000
624	...	2104.0	154.0	66.0	738416	-0.781831

	cycle_day_of_week_cos	sleep_onset_min_sin	sleep_onset_min_cos	\
0	-0.222521	-0.207912	0.978148	
1	-0.900969	-0.091502	0.995805	
2	-0.900969	0.030539	0.999534	
3	-0.222521	0.052336	0.998630	
4	0.623490	-0.212178	0.977231	
..	
620	-0.900969	-0.160743	0.986996	
621	-0.222521	-0.496217	0.868199	
622	0.623490	-0.422618	0.906308	
623	1.000000	-0.337917	0.941176	
624	0.623490	-0.422618	0.906308	

	prev_sleep_onset_min_sin	prev_sleep_onset_min_cos
0	-0.091502	0.995805
1	0.030539	0.999534
2	0.052336	0.998630
3	-0.212178	0.977231
4	-0.195090	0.980785
..
620	-0.496217	0.868199
621	-0.422618	0.906308
622	-0.337917	0.941176
623	-0.422618	0.906308
624	-0.358368	0.933580

[625 rows x 34 columns]

4.1.3 Export

```
[121]: df_phys_dt.to_csv(
        os.path.join(WHOOP_PROCESS_DIR, 'phys.csv'),
        index=False
    )
```

4.2 Merge all metrics

4.2.1 Prep journal features

Rename cols

```
[122]: df_journal = df_journal_valid.copy()
```

```
DF_JOURNAL_COL_NAMES = [  
    'cycle_start_time',  
    'question',  
    'answer',  
    'cycle_date'  
]
```

```
df_journal.columns = DF_JOURNAL_COL_NAMES  
df_journal.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 5289 entries, 0 to 5288
```

```
Data columns (total 4 columns):
```

#	Column	Non-Null Count	Dtype
0	cycle_start_time	5289 non-null	datetime64[ns]
1	question	5289 non-null	object
2	answer	5289 non-null	bool
3	cycle_date	5289 non-null	datetime64[ns]

```
dtypes: bool(1), datetime64[ns](2), object(1)
```

```
memory usage: 129.3+ KB
```

Reformat journal entries

```
[123]: df_journal
```

```
[123]:
```

	cycle_start_time	question	answer	\
0	2024-06-17 00:35:19	Have any alcoholic drinks?	False	
1	2024-06-17 00:35:19	Use CBD oil in any form?	True	
2	2024-06-17 00:35:19	Avoid consuming processed foods?	False	
3	2024-06-17 00:35:19	Take an ice bath?	False	
4	2024-06-17 00:35:19	Have an injury or wound	False	
...	
5284	2022-09-14 21:52:24	Take prescription sleep medication?	True	
5285	2022-09-14 00:00:00	Use CBD oil in any form?	False	
5286	2022-09-14 00:00:00	Eat any food close to bedtime?	True	
5287	2022-09-14 00:00:00	See direct sunlight upon waking up?	True	
5288	2022-09-14 00:00:00	Take prescription sleep medication?	True	

	cycle_date
0	2024-06-17
1	2024-06-17
2	2024-06-17


```

3    2024-06-17
4    2024-06-17
...
5284 2022-09-14
5285 2022-09-14
5286 2022-09-14
5287 2022-09-14
5288 2022-09-14

```

[5289 rows x 4 columns]

```

[124]: df_journal_pivot = df_journal.pivot_table(index='cycle_date',
        columns='question', values='answer', aggfunc='first').reset_index()
df_journal_pivot

```

```

[124]: question cycle_date Avoid consuming processed foods? \
0          2022-09-14                               NaN
1          2022-09-15                               NaN
2          2022-09-16                               NaN
3          2022-09-17                               NaN
4          2022-09-18                               NaN
..          ...
550        2024-06-11                               False
551        2024-06-13                               False
552        2024-06-14                                True
553        2024-06-15                                True
554        2024-06-17                               False

```

```

question Eat any food close to bedtime? Have an injury or wound \
0                                True                               NaN
1                                True                               NaN
2                                True                               NaN
3                                True                               NaN
4                                True                               NaN
..                                ...
550                             True                               False
551                             True                               False
552                             False                              False
553                             False                              False
554                             False                              False

```

```

question Have any alcoholic drinks? Masturbate? \
0                                NaN                               NaN
1                                NaN                               NaN
2                                NaN                               NaN
3                                NaN                               NaN
4                                NaN                               NaN

```

..
550	False	NaN
551	True	NaN
552	False	NaN
553	True	NaN
554	False	NaN

question	See direct sunlight upon waking up?	Spend time outdoors?	\
0	True	NaN	
1	False	NaN	
2	True	NaN	
3	True	NaN	
4	True	NaN	
..	
550	NaN	False	
551	NaN	False	
552	NaN	False	
553	NaN	True	
554	NaN	False	

question	Spend time stretching?	Take an ice bath?	\
0	NaN	NaN	
1	NaN	NaN	
2	NaN	NaN	
3	NaN	NaN	
4	NaN	NaN	
..	
550	False	False	
551	False	False	
552	False	False	
553	False	False	
554	False	False	

question	Take prescription sleep medication?	Use CBD oil in any form?	\
0	True	False	
1	True	False	
2	True	False	
3	True	True	
4	True	False	
..	
550	NaN	True	
551	NaN	True	
552	NaN	True	
553	NaN	True	
554	NaN	True	

question Use a sauna? Use tobacco in any form? Used marijuana

0	NaN	NaN	NaN
1	NaN	NaN	NaN
2	NaN	NaN	NaN
3	NaN	NaN	NaN
4	NaN	NaN	NaN
..
550	False	False	False
551	False	True	False
552	False	True	False
553	False	True	False
554	False	True	False

[555 rows x 15 columns]

[125]: `df_journal_pivot.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 555 entries, 0 to 554
Data columns (total 15 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   cycle_date                            555 non-null    datetime64[ns]
1   Avoid consuming processed foods?     248 non-null    object
2   Eat any food close to bedtime?       555 non-null    object
3   Have an injury or wound               387 non-null    object
4   Have any alcoholic drinks?           538 non-null    object
5   Masturbate?                          242 non-null    object
6   See direct sunlight upon waking up?  168 non-null    object
7   Spend time outdoors?                 387 non-null    object
8   Spend time stretching?                387 non-null    object
9   Take an ice bath?                    311 non-null    object
10  Take prescription sleep medication?   439 non-null    object
11  Use CBD oil in any form?              164 non-null    object
12  Use a sauna?                          373 non-null    object
13  Use tobacco in any form?              335 non-null    object
14  Used marijuana                        83 non-null     object
dtypes: datetime64[ns](1), object(14)
memory usage: 65.2+ KB
```

[126]: `df_journal_question_counts = df_journal_pivot.notnull().sum().reset_index().`
`↪ rename(columns={0: 'count'})`
`df_journal_question_counts`

	question	count
0	cycle_date	555
1	Avoid consuming processed foods?	248
2	Eat any food close to bedtime?	555
3	Have an injury or wound	387

4	Have any alcoholic drinks?	538
5	Masturbate?	242
6	See direct sunlight upon waking up?	168
7	Spend time outdoors?	387
8	Spend time stretching?	387
9	Take an ice bath?	311
10	Take prescription sleep medication?	439
11	Use CBD oil in any form?	164
12	Use a sauna?	373
13	Use tobacco in any form?	335
14	Used marijuana	83

```
[127]: N_DAYS = df_phys.shape[0]
QUESTION_COUNT_THRESHOLD = 0.25
```

```
[128]: keep_questions =
    ↪df_journal_question_counts['question'][df_journal_question_counts['count']
    ↪>= N_DAYS * QUESTION_COUNT_THRESHOLD]
keep_questions
```

```
[128]: 0          cycle_date
1      Avoid consuming processed foods?
2          Eat any food close to bedtime?
3          Have an injury or wound
4          Have any alcoholic drinks?
5          Masturbate?
6      See direct sunlight upon waking up?
7          Spend time outdoors?
8          Spend time stretching?
9          Take an ice bath?
10     Take prescription sleep medication?
11          Use CBD oil in any form?
12          Use a sauna?
13          Use tobacco in any form?
Name: question, dtype: object
```

```
[129]: df_journal_pivot_keep = df_journal_pivot.copy()[keep_questions]
df_journal_pivot_keep
```

```
[129]: question cycle_date Avoid consuming processed foods? \
0      2022-09-14      NaN
1      2022-09-15      NaN
2      2022-09-16      NaN
3      2022-09-17      NaN
4      2022-09-18      NaN
..      ...      ...
550    2024-06-11      False
```

551	2024-06-13	False
552	2024-06-14	True
553	2024-06-15	True
554	2024-06-17	False

question	Eat any food close to bedtime?	Have an injury or wound	\
0	True	NaN	
1	True	NaN	
2	True	NaN	
3	True	NaN	
4	True	NaN	
..	
550	True	False	
551	True	False	
552	False	False	
553	False	False	
554	False	False	

question	Have any alcoholic drinks?	Masturbate?	\
0	NaN	NaN	
1	NaN	NaN	
2	NaN	NaN	
3	NaN	NaN	
4	NaN	NaN	
..	
550	False	NaN	
551	True	NaN	
552	False	NaN	
553	True	NaN	
554	False	NaN	

question	See direct sunlight upon waking up?	Spend time outdoors?	\
0	True	NaN	
1	False	NaN	
2	True	NaN	
3	True	NaN	
4	True	NaN	
..	
550	NaN	False	
551	NaN	False	
552	NaN	False	
553	NaN	True	
554	NaN	False	

question	Spend time stretching?	Take an ice bath?	\
0	NaN	NaN	
1	NaN	NaN	

2	NaN	NaN
3	NaN	NaN
4	NaN	NaN
..
550	False	False
551	False	False
552	False	False
553	False	False
554	False	False

	question Take prescription sleep medication?	Use CBD oil in any form? \
0	True	False
1	True	False
2	True	False
3	True	True
4	True	False
..
550	NaN	True
551	NaN	True
552	NaN	True
553	NaN	True
554	NaN	True

	question Use a sauna?	Use tobacco in any form?
0	NaN	NaN
1	NaN	NaN
2	NaN	NaN
3	NaN	NaN
4	NaN	NaN
..
550	False	False
551	False	True
552	False	True
553	False	True
554	False	True

[555 rows x 14 columns]

Encode datetimes

```
[130]: df_journal_dt = df_journal_pivot_keep.copy()

df_journal_dt['cycle_date_ord'] = df_journal_dt['cycle_date'].apply(lambda x: x.
    ↳toordinal())
df_journal_dt.drop(columns='cycle_date', inplace=True)

df_journal_dt
```

```

[130]: question Avoid consuming processed foods? Eat any food close to bedtime? \
0                                         NaN                                         True
1                                         NaN                                         True
2                                         NaN                                         True
3                                         NaN                                         True
4                                         NaN                                         True
..                                     ...                                     ...
550                                     False                                     True
551                                     False                                     True
552                                     True                                      False
553                                     True                                      False
554                                     False                                     False

question Have an injury or wound Have any alcoholic drinks? Masturbate? \
0                                         NaN                                         NaN                                         NaN
1                                         NaN                                         NaN                                         NaN
2                                         NaN                                         NaN                                         NaN
3                                         NaN                                         NaN                                         NaN
4                                         NaN                                         NaN                                         NaN
..                                     ...                                     ...                                     ...
550                                     False                                     False                                     NaN
551                                     False                                     True                                      NaN
552                                     False                                     False                                     NaN
553                                     False                                     True                                      NaN
554                                     False                                     False                                     NaN

question See direct sunlight upon waking up? Spend time outdoors? \
0                                         True                                         NaN
1                                         False                                        NaN
2                                         True                                         NaN
3                                         True                                         NaN
4                                         True                                         NaN
..                                     ...                                     ...
550                                     NaN                                         False
551                                     NaN                                         False
552                                     NaN                                         False
553                                     NaN                                         True
554                                     NaN                                         False

question Spend time stretching? Take an ice bath? \
0                                         NaN                                         NaN
1                                         NaN                                         NaN
2                                         NaN                                         NaN
3                                         NaN                                         NaN
4                                         NaN                                         NaN
..                                     ...                                     ...
550                                     False                                        False

```

551	False	False
552	False	False
553	False	False
554	False	False

question	Take prescription sleep medication?	Use CBD oil in any form?	\
0	True	False	
1	True	False	
2	True	False	
3	True	True	
4	True	False	
..	
550	NaN	True	
551	NaN	True	
552	NaN	True	
553	NaN	True	
554	NaN	True	

question	Use a sauna?	Use tobacco in any form?	cycle_date_ord
0	NaN	NaN	738412
1	NaN	NaN	738413
2	NaN	NaN	738414
3	NaN	NaN	738415
4	NaN	NaN	738416
..
550	False	False	739048
551	False	True	739050
552	False	True	739051
553	False	True	739052
554	False	True	739054

[555 rows x 14 columns]

4.2.2 Prep workout features

Aggregate features

```
[131]: df_wo_merge_prep = df_wo_full_18.copy()
```

```
[132]: for i in range(5):
        df_wo_merge_prep[f'hr_zone_{i+1}_min'] =_
        df_wo_merge_prep[f'hr_zone_{i+1}_p'] * df_wo_merge_prep['duration_min']

df_wo_merge_prep
```

```
[132]: duration_min  activity_name  activity_strain  calories  hr_max  hr_avg \
0           89      Activity           11.7      705.0    159    127
1           87      Jiu Jitsu           12.8      638.0    184    125
```


2	76	Weightlifting	9.6	467.0	161	119
3	80	Jiu Jitsu	10.1	511.0	167	121
4	59	Martial Arts	11.8	534.0	177	134
..
444	77	Box Fitness	15.7	884.0	175	144
445	36	Powerlifting	7.4	247.0	153	122
446	61	Box Fitness	13.5	556.0	179	133
447	66	Box Fitness	15.9	779.0	180	147
448	86	Box Fitness	13.5	723.0	174	130

	hr_zone_1_p	hr_zone_2_p	hr_zone_3_p	hr_zone_4_p	...	\
0	0.04	0.78	0.18	0.00	...	
1	0.28	0.47	0.11	0.09	...	
2	0.36	0.45	0.19	0.00	...	
3	0.30	0.48	0.17	0.02	...	
4	0.04	0.51	0.29	0.15	...	
..	
444	0.00	0.15	0.33	0.49	...	
445	0.08	0.70	0.20	0.02	...	
446	0.08	0.44	0.19	0.19	...	
447	0.02	0.14	0.24	0.35	...	
448	0.08	0.41	0.30	0.19	...	

	wo_day_of_week_sin	wo_day_of_week_cos	wo_start_min_sin	\
0	-0.974928	-0.222521	-0.224951	
1	-0.433884	-0.900969	-0.991445	
2	0.433884	-0.900969	-0.999391	
3	0.781831	0.623490	-0.991445	
4	0.781831	0.623490	-0.923880	
..	
444	0.000000	1.000000	-0.988362	
445	-0.974928	-0.222521	0.342020	
446	-0.974928	-0.222521	0.619094	
447	-0.433884	-0.900969	-0.928810	
448	0.974928	-0.222521	-0.986286	

	wo_start_min_cos	activity_code	hr_zone_1_min	hr_zone_2_min	\
0	-0.974370	0	3.56	69.42	
1	-0.130526	5	24.36	40.89	
2	0.034899	14	27.36	34.20	
3	-0.130526	5	24.00	38.40	
4	-0.382683	9	2.36	30.09	
..	
444	-0.152123	2	0.00	11.55	
445	-0.939693	12	2.88	25.20	
446	-0.785317	2	4.88	26.84	
447	-0.370557	2	1.32	9.24	

```
448          -0.165048          2          6.88          35.26
```

```

      hr_zone_3_min  hr_zone_4_min  hr_zone_5_min
0           16.02           0.00           0.00
1            9.57            7.83            3.48
2           14.44            0.00            0.00
3           13.60            1.60            0.00
4           17.11            8.85            0.59
..           ...           ...           ...
444          25.41          37.73            2.31
445            7.20            0.72            0.00
446          11.59          11.59            6.10
447          15.84          23.10          16.50
448          25.80          16.34            1.72

```

```
[449 rows x 22 columns]
```

```
[133]: df_wo_merge_prep.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 449 entries, 0 to 448
Data columns (total 22 columns):
#   Column                Non-Null Count  Dtype
---  -
0   duration_min          449 non-null   int64
1   activity_name         449 non-null   object
2   activity_strain       449 non-null   float64
3   calories              449 non-null   float64
4   hr_max                449 non-null   int64
5   hr_avg                449 non-null   int64
6   hr_zone_1_p           449 non-null   float64
7   hr_zone_2_p           449 non-null   float64
8   hr_zone_3_p           449 non-null   float64
9   hr_zone_4_p           449 non-null   float64
10  hr_zone_5_p           449 non-null   float64
11  wo_date_ord           449 non-null   int64
12  wo_day_of_week_sin    449 non-null   float64
13  wo_day_of_week_cos    449 non-null   float64
14  wo_start_min_sin      449 non-null   float64
15  wo_start_min_cos      449 non-null   float64
16  activity_code         449 non-null   int64
17  hr_zone_1_min         449 non-null   float64
18  hr_zone_2_min         449 non-null   float64
19  hr_zone_3_min         449 non-null   float64
20  hr_zone_4_min         449 non-null   float64
21  hr_zone_5_min         449 non-null   float64
dtypes: float64(16), int64(5), object(1)
memory usage: 77.3+ KB

```

```
[134]: df_wo_agg = df_wo_merge_prep.groupby('wo_date_ord').agg({
    'duration_min': 'sum',
    'activity_strain': ['sum', 'max'],
    'calories': 'sum',
    'hr_max': 'max',
    'hr_avg': 'mean',
    'hr_zone_1_min': 'sum',
    'hr_zone_2_min': 'sum',
    'hr_zone_3_min': 'sum',
    'hr_zone_4_min': 'sum',
    'hr_zone_5_min': 'sum'
})

df_wo_agg.columns = [
    'activity_duration_min',
    'activity_strain_sum',
    'activity_strain_max',
    'activity_calories',
    'activity_hr_max',
    'activity_hr_avg',
    'activity_hr_zone_1_min',
    'activity_hr_zone_2_min',
    'activity_hr_zone_3_min',
    'activity_hr_zone_4_min',
    'activity_hr_zone_5_min'
]

df_wo_agg['activity_names'] = df_wo_merge_prep.
    ↳groupby('wo_date_ord')['activity_name'].apply(list)
df_wo_agg['activity_codes'] = df_wo_merge_prep.
    ↳groupby('wo_date_ord')['activity_code'].apply(list)

df_wo_agg['activity_count'] = df_wo_merge_prep.groupby('wo_date_ord').size()

df_wo_agg.reset_index(inplace=True)

df_wo_agg
```

```
[134]:
```

	wo_date_ord	activity_duration_min	activity_strain_sum	\
0	738412	86	13.5	
1	738414	66	15.9	
2	738415	97	20.9	
3	738417	77	15.7	
4	738420	66	14.0	
..	
377	739047	66	8.8	
378	739048	139	21.9	

379	739050	76	9.6
380	739051	87	12.8
381	739052	89	11.7

	activity_strain_max	activity_calories	activity_hr_max	activity_hr_avg	\
0	13.5	723.0	174	130.0	
1	15.9	779.0	180	147.0	
2	13.5	803.0	179	127.5	
3	15.7	884.0	175	144.0	
4	14.0	659.0	179	135.0	
..	
377	8.8	410.0	154	120.0	
378	11.8	1045.0	177	127.5	
379	9.6	467.0	161	119.0	
380	12.8	638.0	184	125.0	
381	11.7	705.0	159	127.0	

	activity_hr_zone_1_min	activity_hr_zone_2_min	activity_hr_zone_3_min	\
0	6.88	35.26	25.80	
1	1.32	9.24	15.84	
2	7.76	52.04	18.79	
3	0.00	11.55	25.41	
4	4.62	17.82	25.08	
..	
377	17.82	37.62	10.56	
378	26.36	68.49	30.71	
379	27.36	34.20	14.44	
380	24.36	40.89	9.57	
381	3.56	69.42	16.02	

	activity_hr_zone_4_min	activity_hr_zone_5_min	\
0	16.34	1.72	
1	23.10	16.50	
2	12.31	6.10	
3	37.73	2.31	
4	12.54	5.94	
..	
377	0.00	0.00	
378	10.45	0.59	
379	0.00	0.00	
380	7.83	3.48	
381	0.00	0.00	

	activity_names	activity_codes	activity_count
0	[Box Fitness]	[2]	1
1	[Box Fitness]	[2]	1
2	[Powerlifting, Box Fitness]	[12, 2]	2

3	[Box Fitness]	[2]	1
4	[Box Fitness]	[2]	1
..
377	[Weightlifting]	[14]	1
378	[Jiu Jitsu, Martial Arts]	[5, 9]	2
379	[Weightlifting]	[14]	1
380	[Jiu Jitsu]	[5]	1
381	[Activity]	[0]	1

[382 rows x 15 columns]

4.2.3 Merge dfs

```
[135]: df_merge = pd.concat(
    [
        df_phys_dt.set_index('cycle_date_ord'),
        df_wo_agg.set_index('wo_date_ord'),
        df_journal_dt.set_index('cycle_date_ord')
    ],
    axis=1
)\
.reset_index()\
.rename(columns={'index': 'cycle_date_ord'})\
.dropna(subset=['recovery_score_p'])

df_merge
```

```
[135]:
```

	cycle_date_ord	recovery_score_p	hr_rest	hr_var	skin_temp_c	\
0	739052	0.84	52.0	92.0	33.30	
1	739051	0.94	48.0	95.0	34.30	
2	739050	0.83	50.0	87.0	34.00	
3	739049	0.73	49.0	80.0	33.20	
4	739048	0.74	50.0	78.0	33.90	
..	
620	738420	0.57	48.0	74.0	33.13	
621	738419	0.49	46.0	68.0	31.83	
622	738418	0.48	47.0	65.0	32.80	
623	738417	0.45	49.0	67.0	32.44	
624	738416	0.55	49.0	68.0	33.30	

	blood_ox_p	sleep_performance_p	resp_rate	asleep_min	in_bed_min	...	\
0	0.9614	0.98	14.8	533.0	570.0	...	
1	0.9889	0.87	14.0	433.0	484.0	...	
2	0.9782	1.00	14.4	535.0	571.0	...	
3	0.9900	1.00	14.8	488.0	562.0	...	
4	0.9840	1.00	14.3	568.0	623.0	...	
..	

620	0.9581	0.93	13.7	453.0	494.0	...
621	0.9661	1.00	14.2	490.0	557.0	...
622	0.9669	0.93	13.7	481.0	564.0	...
623	0.9584	0.78	13.7	410.0	487.0	...
624	0.9733	0.95	14.0	482.0	549.0	...

	Have any alcoholic drinks?	Masturbate?	\
0	True	NaN	
1	False	NaN	
2	True	NaN	
3	NaN	NaN	
4	False	NaN	
..	
620	NaN	NaN	
621	NaN	NaN	
622	NaN	NaN	
623	NaN	NaN	
624	NaN	NaN	

	See direct sunlight upon waking up?	Spend time outdoors?	\
0	NaN	True	
1	NaN	False	
2	NaN	False	
3	NaN	NaN	
4	NaN	False	
..	
620	True	NaN	
621	True	NaN	
622	True	NaN	
623	True	NaN	
624	True	NaN	

	Spend time stretching?	Take an ice bath?	\
0	False	False	
1	False	False	
2	False	False	
3	NaN	NaN	
4	False	False	
..	
620	NaN	NaN	
621	NaN	NaN	
622	NaN	NaN	
623	NaN	NaN	
624	NaN	NaN	

	Take prescription sleep medication?	Use CBD oil in any form?	\
0	NaN	True	

1	NaN	True
2	NaN	True
3	NaN	NaN
4	NaN	True
..
620	True	False
621	True	True
622	True	False
623	True	False
624	True	False

	Use a sauna?	Use tobacco in any form?
0	False	True
1	False	True
2	False	True
3	NaN	NaN
4	False	False
..
620	NaN	NaN
621	NaN	NaN
622	NaN	NaN
623	NaN	NaN
624	NaN	NaN

[625 rows x 61 columns]

```
[136]: df_merge.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
Index: 625 entries, 0 to 624
```

```
Data columns (total 61 columns):
```

#	Column	Non-Null Count	Dtype
0	cycle_date_ord	625 non-null	int64
1	recovery_score_p	625 non-null	float64
2	hr_rest	625 non-null	float64
3	hr_var	625 non-null	float64
4	skin_temp_c	625 non-null	float64
5	blood_ox_p	625 non-null	float64
6	sleep_performance_p	625 non-null	float64
7	resp_rate	625 non-null	float64
8	asleep_min	625 non-null	float64
9	in_bed_min	625 non-null	float64
10	light_sleep_min	625 non-null	float64
11	light_sleep_p	625 non-null	float64
12	deep_sleep_min	625 non-null	float64
13	deep_sleep_p	625 non-null	float64
14	rem_sleep_min	625 non-null	float64

15	rem_sleep_p	625 non-null	float64
16	awake_min	625 non-null	float64
17	awake_p	625 non-null	float64
18	restorative_sleep_min	625 non-null	float64
19	restorative_sleep_p	625 non-null	float64
20	sleep_need_min	625 non-null	float64
21	sleep_debt_min	625 non-null	float64
22	sleep_efficiency_p	625 non-null	float64
23	sleep_consistency_p	625 non-null	float64
24	day_strain	625 non-null	float64
25	calories	625 non-null	float64
26	hr_max	625 non-null	float64
27	hr_avg	625 non-null	float64
28	cycle_day_of_week_sin	625 non-null	float64
29	cycle_day_of_week_cos	625 non-null	float64
30	sleep_onset_min_sin	625 non-null	float64
31	sleep_onset_min_cos	625 non-null	float64
32	prev_sleep_onset_min_sin	625 non-null	float64
33	prev_sleep_onset_min_cos	625 non-null	float64
34	activity_duration_min	374 non-null	float64
35	activity_strain_sum	374 non-null	float64
36	activity_strain_max	374 non-null	float64
37	activity_calories	374 non-null	float64
38	activity_hr_max	374 non-null	float64
39	activity_hr_avg	374 non-null	float64
40	activity_hr_zone_1_min	374 non-null	float64
41	activity_hr_zone_2_min	374 non-null	float64
42	activity_hr_zone_3_min	374 non-null	float64
43	activity_hr_zone_4_min	374 non-null	float64
44	activity_hr_zone_5_min	374 non-null	float64
45	activity_names	374 non-null	object
46	activity_codes	374 non-null	object
47	activity_count	374 non-null	float64
48	Avoid consuming processed foods?	238 non-null	object
49	Eat any food close to bedtime?	539 non-null	object
50	Have an injury or wound	375 non-null	object
51	Have any alcoholic drinks?	526 non-null	object
52	Masturbate?	237 non-null	object
53	See direct sunlight upon waking up?	164 non-null	object
54	Spend time outdoors?	375 non-null	object
55	Spend time stretching?	375 non-null	object
56	Take an ice bath?	299 non-null	object
57	Take prescription sleep medication?	432 non-null	object
58	Use CBD oil in any form?	157 non-null	object
59	Use a sauna?	361 non-null	object
60	Use tobacco in any form?	323 non-null	object

dtypes: float64(45), int64(1), object(15)

memory usage: 302.7+ KB


```
[137]: df_merge['activity_names'] = df_merge['activity_names'].apply(lambda x: [] if x_
↳ is None or (isinstance(x, float) and pd.isna(x)) else x)
df_merge['activity_codes'] = df_merge['activity_codes'].apply(lambda x: [] if x_
↳ is None or (isinstance(x, float) and pd.isna(x)) else x)
```

```
[138]: df_merge[df_merge['activity_names'].isna()]['activity_names']
```

```
[138]: Series([], Name: activity_names, dtype: object)
```

```
[139]: df_merge[df_merge['activity_codes'].isna()]['activity_codes']
```

```
[139]: Series([], Name: activity_codes, dtype: object)
```

```
[143]: merge_activity_encoder = LabelEncoder()
df_merge['activity_codes_comb'] = wo_activity_encoder.
↳ fit_transform(df_merge['activity_codes'].apply(tuple))
df_merge['activity_codes_comb'].unique()
```

```
[143]: array([ 1, 13, 31,  0, 15, 10, 23, 21, 12, 29, 17, 30,  4,  8,  6, 22, 40,
          16,  7, 41, 35, 34,  3, 37,  2, 11, 32,  9, 43, 24, 27, 19, 36, 18,
          14, 42,  5, 39, 38, 25, 28, 26, 33, 20])
```

```
[144]: df_merge.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
Index: 625 entries, 0 to 624
```

```
Data columns (total 62 columns):
```

#	Column	Non-Null Count	Dtype
0	cycle_date_ord	625 non-null	int64
1	recovery_score_p	625 non-null	float64
2	hr_rest	625 non-null	float64
3	hr_var	625 non-null	float64
4	skin_temp_c	625 non-null	float64
5	blood_ox_p	625 non-null	float64
6	sleep_performance_p	625 non-null	float64
7	resp_rate	625 non-null	float64
8	asleep_min	625 non-null	float64
9	in_bed_min	625 non-null	float64
10	light_sleep_min	625 non-null	float64
11	light_sleep_p	625 non-null	float64
12	deep_sleep_min	625 non-null	float64
13	deep_sleep_p	625 non-null	float64
14	rem_sleep_min	625 non-null	float64
15	rem_sleep_p	625 non-null	float64
16	awake_min	625 non-null	float64
17	awake_p	625 non-null	float64
18	restorative_sleep_min	625 non-null	float64

19	restorative_sleep_p	625 non-null	float64
20	sleep_need_min	625 non-null	float64
21	sleep_debt_min	625 non-null	float64
22	sleep_efficiency_p	625 non-null	float64
23	sleep_consistency_p	625 non-null	float64
24	day_strain	625 non-null	float64
25	calories	625 non-null	float64
26	hr_max	625 non-null	float64
27	hr_avg	625 non-null	float64
28	cycle_day_of_week_sin	625 non-null	float64
29	cycle_day_of_week_cos	625 non-null	float64
30	sleep_onset_min_sin	625 non-null	float64
31	sleep_onset_min_cos	625 non-null	float64
32	prev_sleep_onset_min_sin	625 non-null	float64
33	prev_sleep_onset_min_cos	625 non-null	float64
34	activity_duration_min	374 non-null	float64
35	activity_strain_sum	374 non-null	float64
36	activity_strain_max	374 non-null	float64
37	activity_calories	374 non-null	float64
38	activity_hr_max	374 non-null	float64
39	activity_hr_avg	374 non-null	float64
40	activity_hr_zone_1_min	374 non-null	float64
41	activity_hr_zone_2_min	374 non-null	float64
42	activity_hr_zone_3_min	374 non-null	float64
43	activity_hr_zone_4_min	374 non-null	float64
44	activity_hr_zone_5_min	374 non-null	float64
45	activity_names	625 non-null	object
46	activity_codes	625 non-null	object
47	activity_count	374 non-null	float64
48	Avoid consuming processed foods?	238 non-null	object
49	Eat any food close to bedtime?	539 non-null	object
50	Have an injury or wound	375 non-null	object
51	Have any alcoholic drinks?	526 non-null	object
52	Masturbate?	237 non-null	object
53	See direct sunlight upon waking up?	164 non-null	object
54	Spend time outdoors?	375 non-null	object
55	Spend time stretching?	375 non-null	object
56	Take an ice bath?	299 non-null	object
57	Take prescription sleep medication?	432 non-null	object
58	Use CBD oil in any form?	157 non-null	object
59	Use a sauna?	361 non-null	object
60	Use tobacco in any form?	323 non-null	object
61	activity_codes_comb	625 non-null	int64

dtypes: float64(45), int64(2), object(15)

memory usage: 307.6+ KB

4.2.4 Export

```
[145]: df_merge.to_csv(  
        os.path.join(WHOOP_PROCESS_DIR, 'merge.csv'),  
        index=False  
    )
```

```
[141]: # import ast  
# df_imported = pd.read_csv('df_merge.csv')  
# df_imported['activity_names'] = df_imported['activity_names'].apply(ast.  
    ↪ literal_eval)  
# df_imported['activity_codes'] = df_imported['activity_codes'].apply(ast.  
    ↪ literal_eval)
```