



Detecting Irregularities in Electronical Patient Reported Data



John van Bemmelen, OCS Consulting, Netherlands

Contents



- ▶ Data collection & context
- ▶ Entry clusters per date
- ▶ Cluster score per subject pair
- ▶ Resulting dataset
- ▶ Results made visual
- ▶ Adaptations
- ▶ Conclusion



Data collection & context

- ▶ Between visits to the clinic: home situation
- ▶ Each subject one entry per day, between 6 and 10 AM
- ▶ Electronic device for data entry
- ▶ Device captures metadata: entry time



Observed problem

- ▶ On one day **entries of 7 subjects within 3 minutes.**
Next day 6 of these again had entries within 3 minutes, but about **an hour later.**
- ▶ Implausible that entries were done independently.



Entry clusters per date

- ▶ Per date: entries of all subjects, sorted by time
- ▶ Missing: no entry, but entry expected for the subject
- ▶ Cluster:
 - ▶ At most 60 seconds apart
 - ▶ Missing times
 - ▶ Identified by a number

Time		SubjID	
Missing	1	2104	
Missing	1	1965	
6:22:38	2	1789	
6:22:58	2	2033	≤60 sec
6:23:51	2	1907	≤60 sec
6:24:59		2049	>60 sec
6:57:03		1932	>60 sec
6:58:10	3	2002	>60 sec
6:59:03	3	1857	≤60 sec
7:06:43		1921	>60 sec



Entry clusters per date: code (1)

```
* Longest seconds-gap between consecutive
  entries still considered clustering
;
%let timegap=60;

* Intermediate cluster numbering:
  also numbers NON-cluster obs
;
data entries_2 (drop=_prevtime);
  retain _prevtime _clustnr;
  set entries;
  by date time;

  if first.date then _clustnr=1;
  else if missing(_prevtime) and not missing(time)
    then _clustnr+1;
  else if time-_prevtime>&timegap then _clustnr+1;

  _prevtime=time;
run;
```

Time	
Missing	1
Missing	1
6:22:38	2
6:22:58	2
6:23:51	2
6:24:59	3
6:57:03	4
6:58:10	5
6:59:03	5
7:06:43	6



Entry clusters per date: code (2)

```
* Define definitive cluster number:
- Ignore non-clusters: consisting of only
  a single observation
- Number true clusters consecutively,
  starting at 1 per date
;
data entries_clustnr (keep=subjid date time clustnr);
  set entries_2;
  by date _clustnr;
  if first.date then __clustnr=0;
  if not(first._clustnr and last._clustnr) then
  do;
    if first._clustnr then __clustnr+1;
    clustnr=__clustnr;
  end;
run;
```

Time	
Missing	1
Missing	1
6:22:38	2
6:22:58	2
6:23:51	2
6:24:59	3
6:57:03	4
6:58:10	3
6:59:03	3
7:06:43	6

Repeated for all dates



SubjID	23May	24May	25May	26May	27May
1789	6:22:5 2	6:23:2 1	6:35:1 1	6:31:47	6:44:2 1
1857	6:59:0 3	7:05:1 2	7:01:3 2	6:57:52	6:47:5 1
1907	6:23:5 2	6:24:1 1	6:36:3 1	6:38:34	6:45:0 1
1921	7:06:43	7:25:44	7:02:1 2	7:03:38	6:47:2 1
1932	6:57:03	6:47:23	6:45:22	6:50:00	6:53:49
1965	Missing 1	8:35:37	8:51:59	9:02:3 2	<Stop
2002	6:58:1 3	7:05:5 2	7:15:37	6:55:40	7:13:12
2033	6:22:5 2	6:19:38	6:22:43	6:28:4 1	6:46:5 1
2049	6:24:59	6:24:5 1	6:36:0 1	6:29:1 1	6:45:5 1
2104	Missing 1	9:04:55	8:56:52	9:03:1 2	9:02:55
2126	Start>	8:25:24	8:30:09	8:14:45	8:19:06
2141	7:10:51	6:58:33	7:01:3 2	7:10:31	7:11:57



Transform Subject-IDs and Dates

- ▶ For application of algorithm:
 - ▶ Subject-IDs recoded to consecutive Subject Numbers
 - ▶ Dates recoded to Day numbers

SubjNo	Day 30	Day 31	Day 32	Day 33
119	6:22:52	6:23:21	6:35:10	6:31:47
127	6:59:03	7:05:12	7:01:32	6:57:52
137	6:23:52	6:24:11	6:36:31	6:38:34
141	7:06:43	7:25:44	7:02:12	7:03:38
152	6:57:03	6:47:23	6:45:22	6:50:00
165	Missing	8:35:37	8:51:59	9:02:32
202	6:58:13	7:05:52	7:15:37	6:55:40

- ▶ For data presentation:
 - ▶ Format: Subject Number → Subject-ID
 - ▶ Macro variable for date-value of Day 1



Next: data of subject pairs

SubjNo	Day 30	31	32	33	34
1	6:22:5 ²	6:23:2 ¹	6:35:1 ¹	6:31:47	6:44:2 ¹
2	6:59:0 ³	7:05:1 ²	7:01:3 ²	6:57:52	6:47:5 ¹
3	6:23:5 ²	6:24:1 ¹	6:36:3 ¹	6:38:34	6:45:0 ¹
4	7:06:43	7:25:44	7:02:1 ²	7:03:38	6:47:2 ¹
5	6:57:03	6:47:23	6:45:22	6:50:00	6:53:49
6	Missin ¹	8:35:37	8:51:59	9:02:3 ²	<Stop
7	6:58:1 ³	7:05:5 ²	7:15:37	6:55:40	7:13:12
8	6:22:5 ²	6:19:38	6:22:43	6:28:4 ¹	6:46:5 ¹
9	6:24:59	6:24:5 ¹	6:36:0 ¹	6:29:1 ¹	6:45:5 ¹
10	Missin ¹	9:04:55	8:56:52	9:03:1 ²	9:02:55
11	Start>	8:25:24	8:30:09	8:14:45	8:19:06
12	7:10:51	6:58:33	7:01:3 ²	7:10:31	7:11:57



Cluster score per subject pair (1)

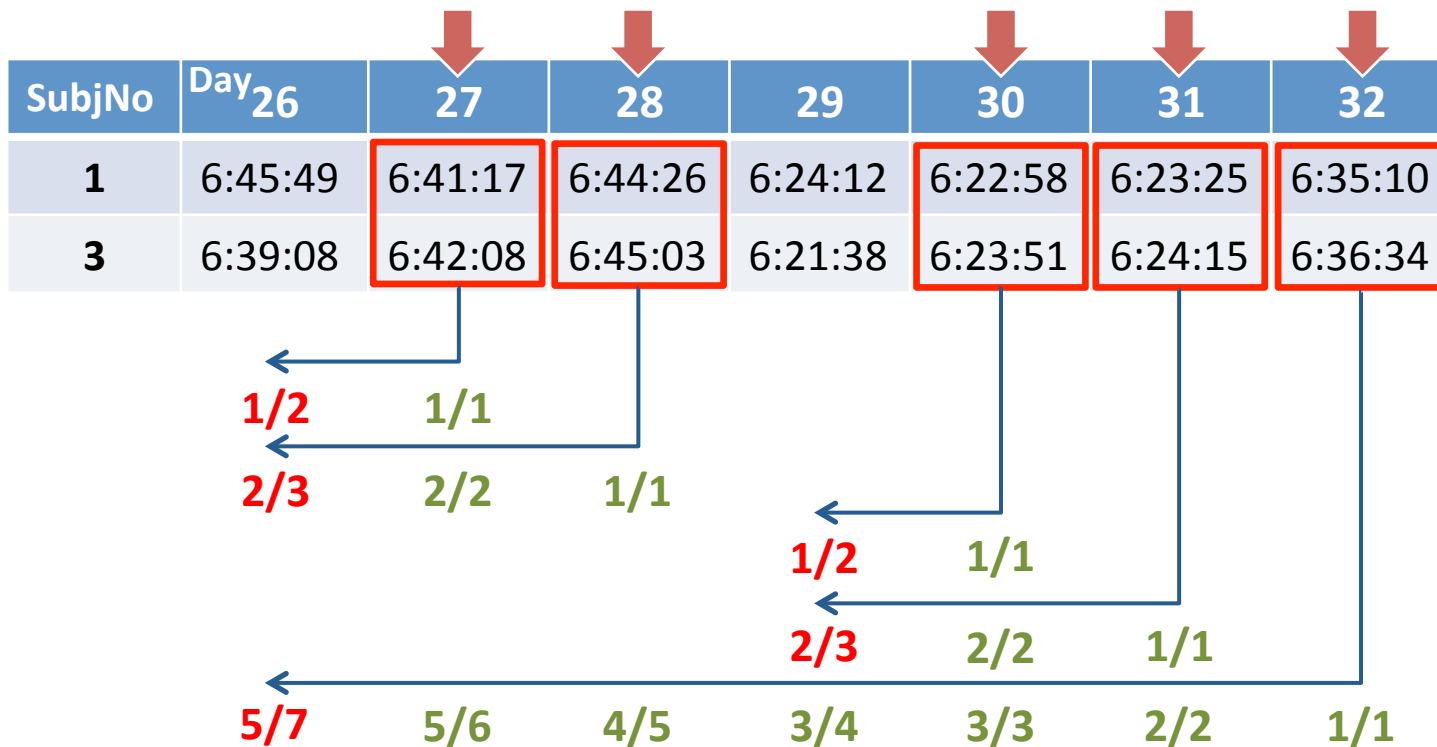
- ▶ Contribution to cluster score:
 - ▶ Days where both entries have same cluster number

SubjNo	Day 26	27	28	29	30	31	32
1	6:45: 2	6:41: 1	6:44: 3	6:24:12	6:22: 2	6:23: 1	6:35: 1
3	6:39:08	6:42: 1	6:45: 3	6:21:38	6:23: 2	6:24: 1	6:36: 1



Cluster score per subject pair (2)

- ▶ Contribution to cluster score:
 - ▶ For each cluster day inspect preceding days, as long as ratio of clustering days in range of days at least 0.75





Cluster score per subject pair (3)

- ▶ Contribution to cluster score:

- ▶ Difference in minutes:

$$1 + \sum_{subj, prevday \uparrow} \text{minutedif} \downarrow_{subj, prevday \uparrow 2}$$

SubjNo	Day ₂₆	27	28	29	30	31	32
1	6:45:49	6:41:17	6:44:26	6:24:12	6:22:58	6:23:25	6:35:10
3	6:39:08	6:42:08	6:45:03	6:21:38	6:23:51	6:24:15	6:36:34

Diagram illustrating time differences between subject pairs (1 and 3) on specific days (27, 28, 30, 31). Red arrows point to the days of interest. Red boxes highlight the time values for subjects 1 and 3 on these days. Curved arrows indicate the time differences between the two subjects on the same day.

- Day 27: 3.2min (between 6:41:17 and 6:42:08), 2.9min (between 6:44:26 and 6:45:03)
- Day 30: 0.4min (between 6:22:58 and 6:23:51)
- Day 31: 0.4min (between 6:23:25 and 6:24:15)

Day 27: 1

Day 28: $1 + 3.2^2 + 2.9^2 = 19.4$

Day 30: 1

Day 31: $1 + 0.4^2 + 0.4^2 = 1.4$



Cluster score per subject pair (4)

- ▶ Contribution to cluster score:

- ▶ Difference in minutes:

$$1 + \sum_{subj, prevday} \uparrow \downarrow \text{minutedif} \downarrow \text{subj, prevday} \uparrow^2$$



SubjNo	Day ₂₆	27	28	29	30	31	32
1	6:45:49	6:41:17	6:44:26	6:24:12	6:22:58	6:23:25	6:35:10
3	6:39:08	6:42:08	6:45:03	6:21:38	6:23:51	6:24:15	6:36:34

6.1min 9.3min 12.2min 11.8min
5.6min 8.5min 12.7min 12.3min

Day 27: 1

Day 28: $1 + 3.2^2 + 2.9^2 = 19.4$

Day 30: 1

Day 31: $1 + 0.4^2 + 0.4^2 = 1.4$

Day 32: $1 + 11.8^2 + 12.3^2 + 12.2^2 + 12.7^2 + 9.3^2 + 8.5^2 + 6.1^2 + 5.6^2 = 827.6$



Cluster score per subject pair (5)

- ▶ Contribution to cluster score:

- ▶ If missing *minutedif*:

$$2 \times (pairdays/pairmis1days + pairmis2days)$$

pairdays = 53
pairmis1days = 3
pairmis2days = 1

SubjNo	Day ₅₅	56	57	58	59	60	61
1	6:58:19	6:55:31	Missing	6:39:47	Missing	Missing	Missing
3	6:50:08	6:54:48	Missing	6:40:22	6:37:01	6:21:36	<Stop

15.7min
missing
14.4min

Day 56: 1

Day 57: $1 + 2 \times (53/4)^2 = 352.1$

Day 58: $1 + 2 \times (53/4)^2 + 15.7^2 + 14.4^2 = 807.0$



Cluster score per subject pair (6)

Day	Score contribution
8	0
...	
26	0
27	1
28	19.4
29	0
30	1
31	1.4
32	827.6
...	
55	0
56	1
57	352.1
58	807.0
59	0
60	0

- ▶ Sum score contributions over all days:
2010.5

- ▶ Divide by pairdays:

$$2010.5 / 53 = \mathbf{37.9}$$

Cluster score



Cluster score calculation: code (1)

Reading all data into two arrays

```
data clusterscores;  
  set entries (keep=subjno day time clustnr) end=eof;  
  
  * T{i,j} = time  
  * C{i,j} = clustnr  
  ;  
  array t{&maxsubj,&maxday} _temporary_;  
  array c{&maxsubj,&maxday} _temporary_;  
  
  t{subjno,day}=time;  
  c{subjno,day}=clustnr;  
  
  if eof then ...
```

Data step continues with
calculating cluster score
(next slide)



Cluster score calculation: code (2)

Walk through subjects and days to calculate cluster score

```
do subj1=1 to &maxsubj-1;
  do subj2=subj1+1 to &maxsubj;
    sumscore=0;                                /* initialise ;
    do dy=1 to &maxday;
      if c{subj1,dy}^=. and c{subj1,dy}=c{subj2,dy} then
        do;
          sumscore+1;                            /* evaluate day itself ;
          do dy2=dy-1 to 1 by -1 until(ratioclust<0.75);
            if "dy2 clustering" ... /* evaluate timedif ;
              "update ratioclust"
          end;
        end;
      end;
    score=sumscore/"pairdays";
    output;
  end;
end;

run;
```



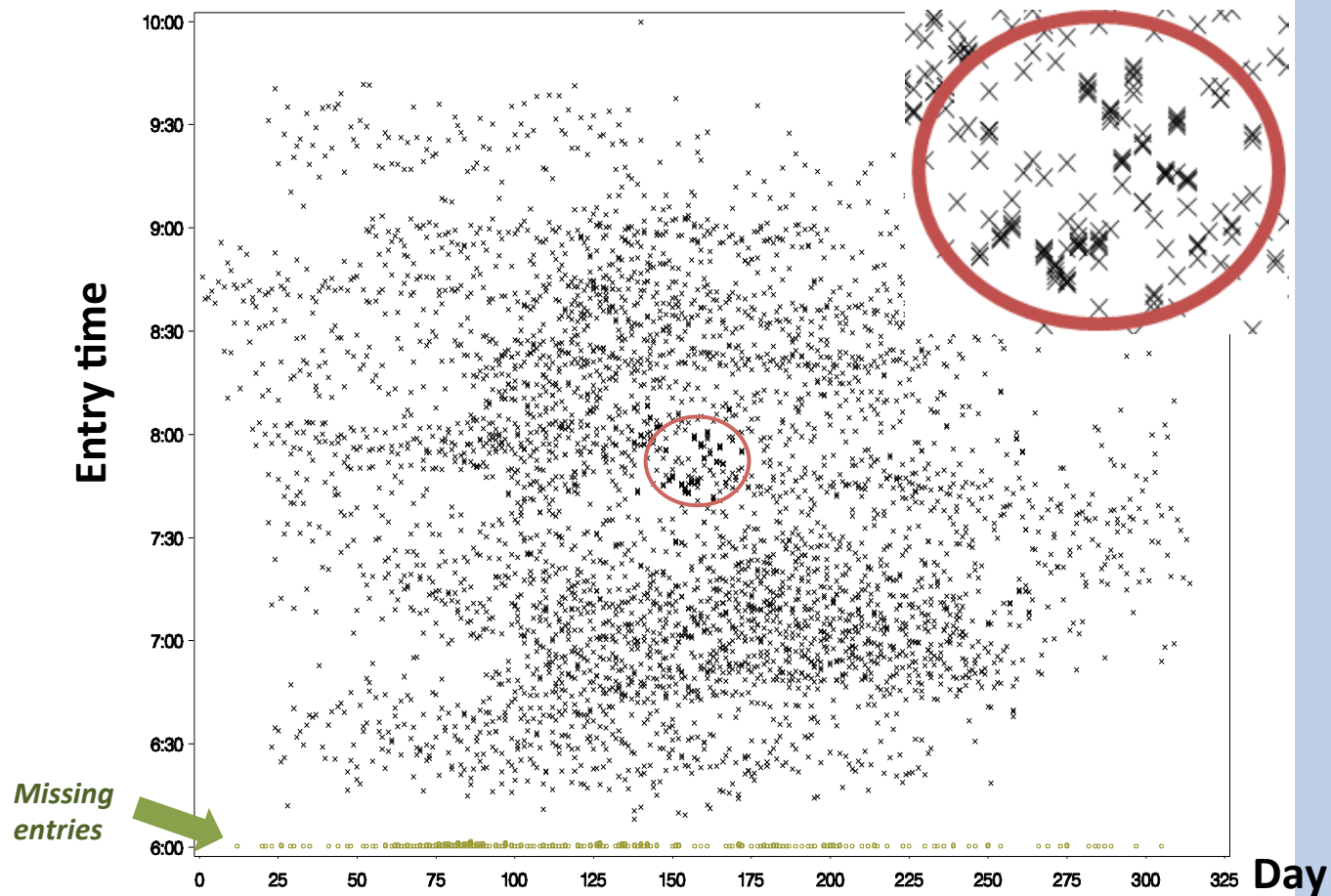
Resulting dataset

SUBJ1	SUBJ2	SCORE	SUM- SCORE	PAIR- DAYS	CLUST- DAYS	MAX- CLUST- RANG	PAIR- MIS1- DAYS	PAIR- MIS2- DAYS
1	3	0.008	1	122	1	1	10	12
1	5	0.016	2	126	2	1	11	11
1	8	0.011	1	90	1	1	6	4
1	11	0.025	2	79	2	1	3	33
2	3	96.420	13402.4	139	11	7	1	12
2	5	92.615	13243.9	143	11	7	1	11
2	7	100.491	13365.3	133	11	7	1	9
2	10	35.550	3590.6	101	7	7	1	7
2	13	0.012	1	85	1	1	1	23

Results made visual



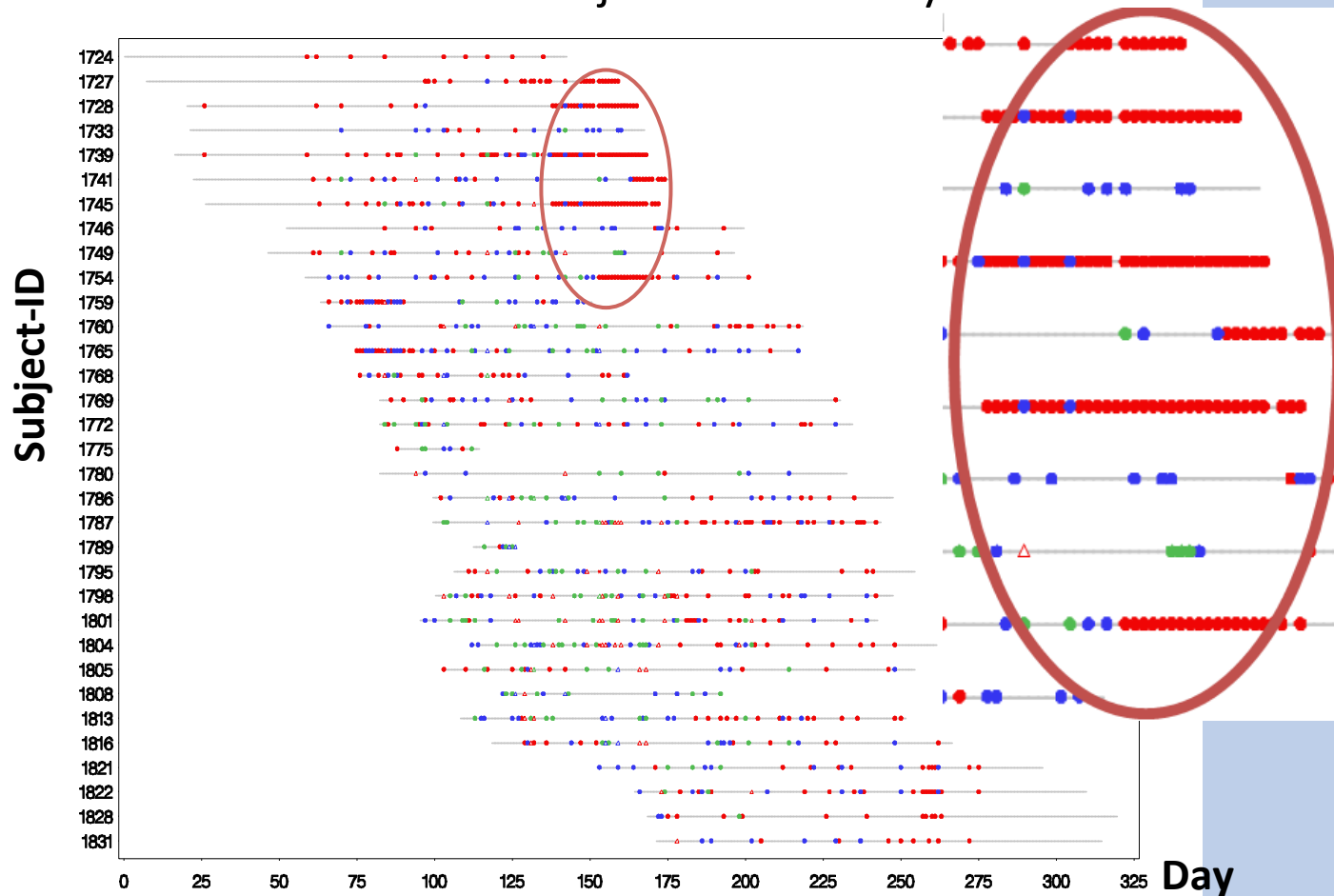
- Scatter plot of entry times over days, for all subjects



Results made visual



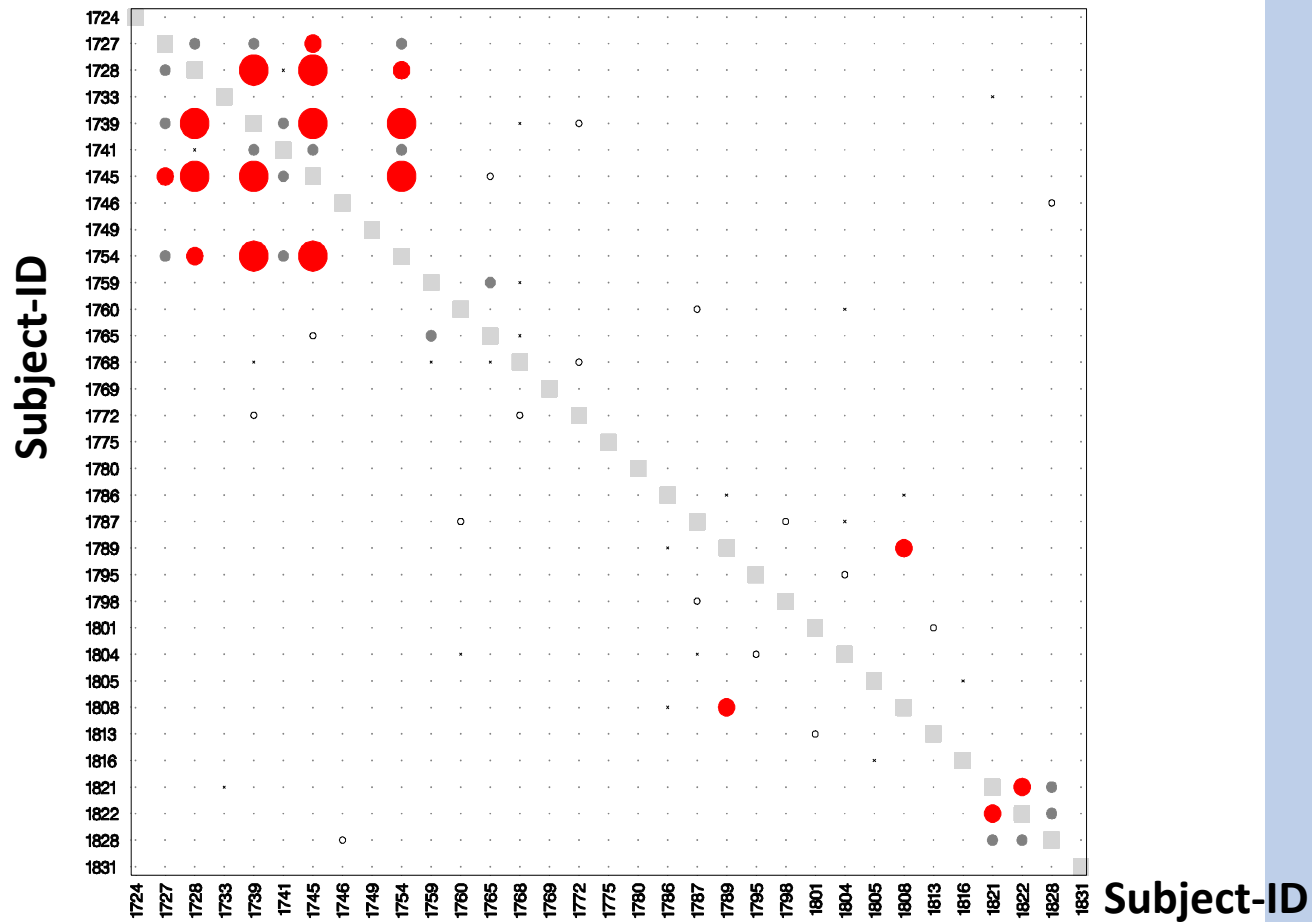
- Plot of cluster numbers for all subjects over all days



Results made visual



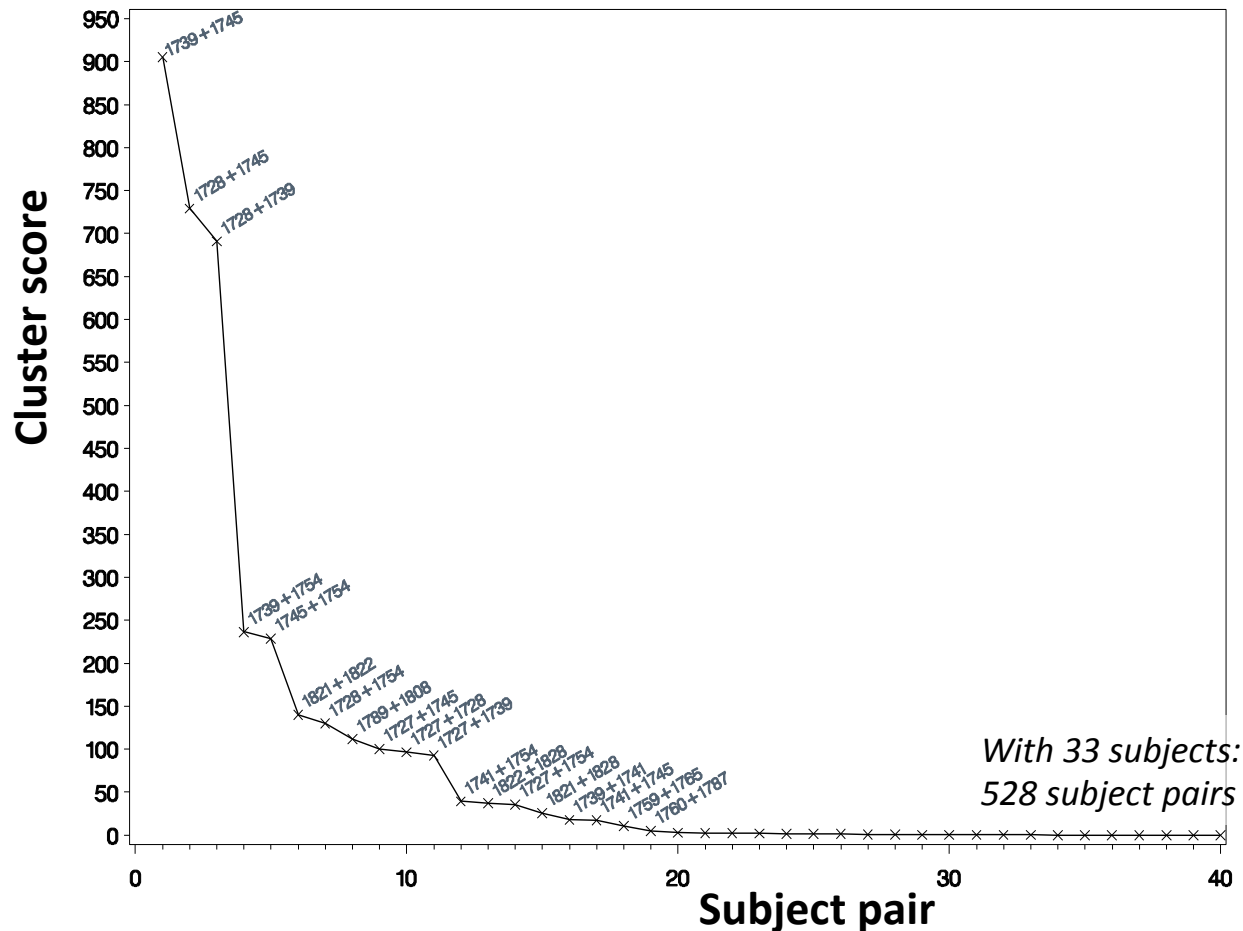
- ▶ Plot of scores between pairs of subjects



Results made visual



- Plot of descending scores for subject pairs





Adaptations

- ▶ By site instead of for full study
- ▶ Scores for other diary data (e.g. evening entries) might be combined
- ▶ Alarm time? Consider lowering score contribution
- ▶ Diary entries requested only 1 week per 4 weeks.

Possible alternatives

- ▶ Entry durations
- ▶ Geographical information
 - ▶ probably not ethical to collect these



Conclusion

- ▶ Programming not complicated
- ▶ Formulas arbitrary
- ▶ Capable of clear distinction
 - ▶ Data review of identified suspicious cases remains important
- ▶ Entry times not relevant for study analysis: no risk for power of study (unless suspicious data are omitted)
- ▶ A lot of work:
 - ▶ Informing stakeholders
 - ▶ Preparing audits
 - ▶ Additional study analyses
- ▶ Wherever possible, **verify that patient reported data are indeed entered by individuals**



Questions?