Data analysis

Binus University Workshop Day 4

https://osdoc.cogsci.nl/binus2022

Sebastiaan Mathôt

cogsci.nl/smathot   cognitivescience   @smathot   @cogscinl   sebastiaanmathot

Today (day 4)

- Before the break
    - Preparing data from online experiments
    - Preparing data from lab experiments
    - From 'long' to 'wide' format
- After the break
    - Conducting a statistical analysis
    - Q&A and workshop end!

Preparing data from online experiments

JATOS results

- For online experiments, data is stored in JATOS

- Each session is one entry in the results

- Each result has a status

  - FINISHED → completed without error

  - FAIL → canceled or an error occurred

  - DATA_RETRIEVED or STARTED → started and still running

- Many non-finished results is normal

Exporting JATOS results

- Export (download) results from JATOS
  - Format: a text file (JSON) that is hard to work with
- Convert this to a spreadsheet with OpenSesame
  - xlsx
  - csv

What does the data look like?

- The data is now in 'long format'

    – One row → one trial (or strictly: one logger call)

    – One column → one variable (response time, conditions, etc.)

circle_radius	circle_x	circle_y	color_circ	color_resp	cursor_x	cursor_y
47.419084957	-41.32610381	-21.8718	60	#ff0019	120.2	170.6
30.145964825	0.321294013	54.5191	60	#00fffa	-124.8	-208.4
40.65773726	-61.24083894	-9.23086	180	#f6ff00	-114	-221
30.731492496	-52.77042171	50.9574	300	#ff0c00	142	-215
46.15249148	42.52746686	-22.3364	240	#8cff00	220	-141
44.448383954	-8.719520073	-8.61174	60	#2100ff	166	-208
27.367328103	0.898460084	49.6481	0	#00d0ff	-184	-38
28.975009611	46.52949998	-40.1248	0	#ff002e	250	-47
43.707697434	-1.420912771	-15.6205	180	#00ffcb	274	-56
25.12529554	-54.33439349	-33.7059	60	#ff0072	198	133
41.057334121	8.777735904	-5.68886	180	#ff00d0	-167	187
26.057125586	18.58981971	39.1514	240	#5dff00	245	-97
38.352137231	49.65882024	15.7425	0	#faff00	98.3333	177.667
25.561184151	4.500545638	21.0521	0	#00faff	-185.667	-4.33333
32.657980964	52.36086182	16.2992	180	#ffe900	-119.667	-177.333
34.638935809	39.41841023	12.8508	180	#ff0800	-237.667	-9.33333
45.262287876	-47.54301088	-21.0713	0	#faff00	129	235.5
47.978277993	-1.680985789	-9.07625	0	#005dff	-159	-128.5
34.412113376	-2.752387559	-14.1043	300	#ff0050	41	-204.5
38.623631552	57.29050931	-46.044	60	#e5ff00	-158	211.5
35.428911871	52.74208872	8.18203	120	#ff1d00	-105	135.5
45.230168006	50.30082125	-31.8168	240	#00ffd0	148	173.5
26.491416544	41.77216612	30.4769	300	#a5ff00	210	81.5
27.335235404	-60.36282163	37.9789	300	#ffaa00	204	-73.5
43.0331223	-35.73379107	26.2844	60	#ff0033	158	177.5
30.366838391	34.29655236	-48.886	240	#ff5900	-36	-237.5
29.527831583	19.91670268	-57.1047	0	#ff0043	230	-65.5
30.540731238	-19.18571872	1.771	240	#00ff72	226	117.5
39.106640976	-44.2331753	4.75701	240	#ffe900	98	-202.5
37.425961922	-21.2322251	-33.3255	0	#ff0055	248	-85.5
35.98376704	47.41639887	-28.3113	240	#50ff00	239	-81.5
26.209392184	-8.592370884	-49.1479	180	#ff0094	-180	121.5
34.325752	-25.69184663	-36.0924	60	#00ff7f	-177	-103.5
35.058814762	-36.77356374	3.40385	180	#ff0c00	-229	-12.5
25.696674164	4.649316361	-20.2214	180	#ff003b	-226	53.5

Preparing data from lab experiments

OpenSesame data files

- For each participant, OpenSesame creates a log file
  - long format
  - .csv

Merging data files

- You generally need to merge data

  - From multiple participant data files

  - Into a single data file

- There are many ways to do this

- I will demonstrate a simple Python script[1]


```python
from datamatrix import DataMatrix, io, operations as ops

# Change this to the folder that contains the .csv files
SRC_FOLDER = 'subject-data'
# Change this to a list of column names that you want to keep
COLUMNS_TO_KEEP = [
    'response_time_mouse_response',
    'correct_mouse_response',
    'target_match',
    'distractor_match',
    'subject_nr'
]


dm = DataMatrix()
for basename in os.listdir(SRC_FOLDER):
    path = os.path.join(SRC_FOLDER, basename)
    print('Reading {}'.format(path))
    dm <<= ops.keep_only(io.readtxt(path), *COLUMNS_TO_KEEP)
io.writetxt(dm, 'merged-data.csv')

# %%
```

What does the data look like?

- The data is now again in long format

  - One row → one trial (or strictly: one logger call)

  - One column → one variable (response time, conditions, etc.)

	A	B	C	D	E
1	correct_mouse_response	distractor_match	response_time_mouse_response	subject_nr	target_match
2	1	1	1108.48185771	12	0
3	1	1	2096.65474527	12	1
4	1	0	1101.28525253	12	1
5	0	0	975.422994121	12	0
6	1	0	2125.30502787	12	1
7	1	1	1356.98071608	12	0
8	1	1	1345.61001349	12	0
9	1	0	1524.51844536	12	1
10	1	0	1522.42294462	12	0
11	1	1	1037.48477124	12	0
12	1	0	1421.08124457	12	0
13	1	0	719.820201086	12	1
14	1	0	790.404949028	12	1
15	1	0	1355.32290996	12	0
16	1	0	1218.32056875	12	0
17	1	1	1483.00175199	12	1
18	1	1	1525.66399784	12	1
19	1	1	1016.29582361	12	1
20	1	1	801.837532586	12	0
21	1	1	1249.93027072	12	0
22	1	0	1076.46042094	12	1
23	1	0	898.423152294	12	1
24	1	1	741.809375238	12	1
25	1	1	1029.47012991	12	1
26	1	0	925.91580808	12	1
27	0	1	1438.77255938	12	1

From 'long' to 'wide' format

From 'long' to 'wide' format

- There are different ways to organize data
  - long format is most common
- Statistical software often expects a specific format
  - Such as 'wide' format
  - In JASP, SPSS, etc.

Wide format

- Wide format is an aggregated format
  - It results from averaging
  - One row corresponds to one participant
  - One column corresponds to one condition
  - Cells correspond to means of a dependent measure (e.g. mean RT)

	A	B	C	D	E	F
	correct_mous	1				
	Average - re	target_mal	distractor_	tch		
		0		1		
	subject_nr	0	1	0	1	Total Result
	1	1056.164727	1306.510413	936.5798118	1137.402803	**1106.075938**
	2	844.8905335	2076.169215		1624.553065	**1515.204271**
	3	1078.891567	1275.48846	886.9407202	1007.329076	**1057.668477**
	4	1386.365212	1508.690111	1028.876542	1229.160839	**1285.207215**
	5	1019.081172	1087.052486	852.8266307	911.1125923	**966.3662377**
	6	1304.578031	1403.471202	1248.661603	1220.768674	**1294.120898**
	7	1191.539317	1290.638633	1047.105806	1094.581652	**1156.670486**
	8	1055.623389	1235.147314	995.7164992	1250.776422	**1132.482608**
	10	930.5312436	1000.285413	815.9206353	980.1719324	**930.5539206**
	11	968.311716	987.6710955	831.5132788	928.4413636	**928.0128749**
	12	946.36394	1001.62517	944.8001103	913.0381323	**951.2157919**
	13	1321.172898	1225.701347	1105.507983	1178.413811	**1205.756235**
	14	1065.068116	1128.600226	971.5181289	983.0255563	**1036.729518**
	15	1432.674534	1511.653474	1181.010434	1290.183952	**1353.880598**
	16	817.1036834	903.740692	843.5744405	817.7446968	**845.4376101**
	17	825.3354587	954.8947764	782.2730768	836.7013905	**849.8011756**
	18	1191.029703	1144.191564	1032.400124	980.5667989	**1086.635827**
	20	787.3731174	823.3265984	767.6949876	885.8239236	**815.4480873**
	21	1021.14792	1160.421141	893.4197234	943.7635713	**998.4927842**
	22	950.4450039	1017.44031	757.7052348	805.5565359	**883.2343202**
	23	898.9451556	870.7244105	979.9482282	932.4222365	**921.0094885**
	24	959.436864	1091.244253	846.8215119	921.3281191	**952.553421**
	25	938.9522977	1059.210343	865.0644538	880.4628886	**933.8798525**

Pivot tables

- A pivot table is a tool to go from long to wide format

- In spreadsheet software
  - Excel
  - LibreOffice Calc
  - Google Sheets

- Or programmatically
  - Python DataMatrix
  - Python Pandas
  - R

Let's get to work!

Slides: https://osdoc.cogsci.nl/binus2022

Statistical analysis

JASP

- JASP is free software for statistics[1]
  - Graphical user interface
  - Based on R

JASP

- Modify the pivot table
  - One row with column headers
- Open in JASP
- Conduct Repeated Measures ANOVA

Let's get to work!

Slides: https://osdoc.cogsci.nl/binus2022