Competitive Behavioral assessment of Rivalry and Admiration-seeking (CoBRA) task – Manual

Short description

The CoBRA task is a social decision-making task designed to capture narcissistic reactions to defeat, using a computer game competition where the outcomes are rigged. Players are told they will play against previous participants whose performance has been pre-recorded and that in doing so, they will take part in the competition's global ranking.

Striving for dominance is measured both on a one-to-one competitive level and on a global hierarchical level by 2 distinct outcome measures:

- <u>outcome measure 1 – point stealing (rivalry)</u>: at the beginning of each trial, participants can choose to steal points from their next opponent to increase their own chances of victory.

- <u>outcome measure 2 – rank buying (admiration-seeking)</u>: at the end of each trial, participants can pay for boosters that improve their rank in the competition's global league table.

Please, note that this game is designed to work on PCs and Windows tablets, the code may need adaptations to run on Mac.

Installation

- Decompress the .zip file "cobra_task_v1-main.zip" in a location of your choice.
  Warning: Do <u>not</u> put it in MATLAB's folder; anywhere else is fine.

- Make sure you have <u>MATLAB</u> installed on your computer (version 2013 or older) as well as <u>Python</u> (versions 2.7 and 3.5 should both work).

Instructions

- The task is about 30 minutes long, so make sure your device is plugged in, or that it has enough battery.

- Sound must be turned on; check that the volume is audible without being too loud.

- Open the 'cobra_task_v1-main' folder. Once you are in the folder, open the file 'main_game_launcher.m' with MATLAB (right click -> Open with -> MATLAB), and simply click "run" from the MATLAB editor window (green arrow in the upper menu).

- First, go through *Competition Rules* with the participant. It is very important not to say anything that would destroy the illusion of a real competition. Ask the participant whether they have any questions after reading the rules. Do not answer anything about what strategy they should use; just make sure that they understood all the rules correctly. If it is not the case, feel free to cycle through them one more time.

- Second, go to *Snake practice*. Emphasize to participants that there are only 2 command keys in this game: the left and the right arrow key. The snake turns to <u>its own</u> left each time they press left (it is not necessarily the participant's left, depending on how the snake is oriented on the screen). The same goes for turning right.
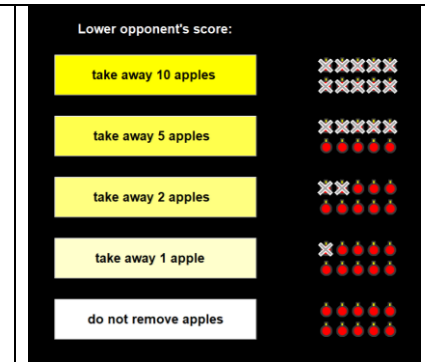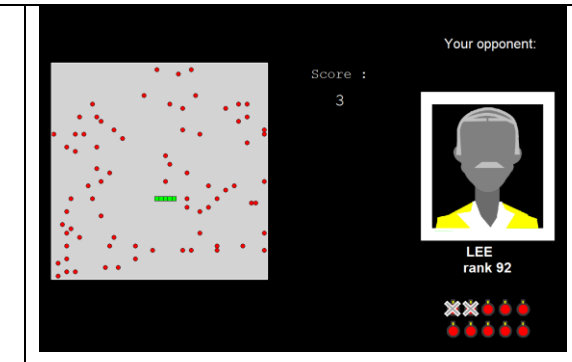 If the participant wants to, they can practice a second time before entering the competition. Do not forget to make them answer the rank estimation question with the green slider button at the end of Snake practice. Some people do not realize they can move the slider and press "confirm" too soon. If this is the case, make them go through Snake practice one more time.

- You can then start the game by pressing "START":
  - Select "24" for the number of rounds (trials).
  - The ID number should be the patient's ID in the lab, but any other string of characters will work (avoid using special characters though).
  - Participants can enter any name (middle names, nicknames or anything else they feel comfortable using).
  - If the task does not let you go to the next page, it is because some data is missing.

  Warning*: The .csv file with the participant's data is created based on this ID page (the file's name is the participant's ID number). So, if you enter the same ID several times in a row, previous files will be overwritten. It is therefore very important to copy all collected data files elsewhere before starting a new experiment.*
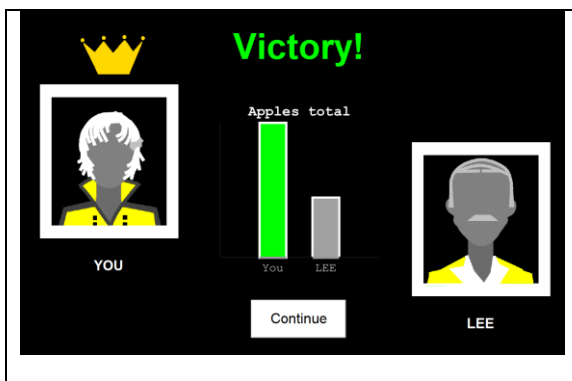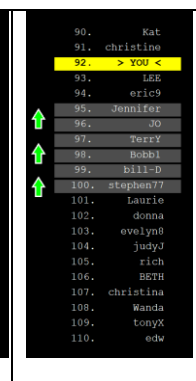
- On the next page, participants will have to choose their avatar for the game by pressing the little rectangle with the corresponding number under the chosen picture. They also need to choose whether they agree to being listed in the ranking after they played (both choices are part of the deception to increase the competition's credibility, they do not impact the game in any way).

- Next, participants fill out the PANAS affective scale (simply follow the instructions in the task). They will have to do it at the beginning and at the end of the game. If they miss out one or two answers (by clicking too quickly on the screen), just carry on with the experiment, this item will be automatically coded as "0" in the .csv file (valid ratings go from 1 to 5).

- The competition begins, with the following sequence of events on each trial:
  1. Participants see their next opponent for a few seconds (Screen 1 below).
  2. They choose to take away apples (points) from their opponent (Screen 2 below).
  3. The snake game window appears. They can start the snake whenever they are ready <u>by pressing the left arrow key</u> (Screen 3 below).

Screen 1: Next opponent *(goes automatically to 2nd screen after a few seconds)*	Screen 2: Point stealing phase	Screen 3: Snake game phase *(the apples below the opponent's picture are a reminder of the player's choice for point stealing on that same trial)*

4. After the snake game phase is over, the results are displayed, paired with a winning or a losing sound (Screen 4 below). When the participant is ready, they can move to the next screen.
5. Here, they can choose to purchase rank-enhancing boosters if they want to (Screen 5 below).
6. The final window displays their rank with the standard rank gain/loss + the chosen booster's effect (Screen 6 below). Their position at the end of the trial is highlighted in yellow in the ranking. From here, they automatically move to the next trial.

Warning: The researcher should watch each participant's first two trials, and then let them play on their own, to avoid influencing their decision-making during the task.

Screen 4: Outcome window *(the bars in the middle only indicate the final apple count; participants do not know how many apples were stolen from them)*	Screen 5: Standard rank gain and possibility to by boosters to increase one's rank.	Screen 6: Additional rank gain with a booster

- At the end of the game, participants will have to fill out the PANAS affective scale again. Then, after seeing a screen with their final rank and monetary earning, they will have to answer a series of end-game questions, also incorporated in the task.

- Pay the participant $25 (or any other amount that was decided in advance*) for the Snake game, without letting them know that this is a fixed amount. If other participants still remain to be tested, it is important to keep the illusion that this is an individually computer reward.

  Warning: If possible, do not disclose any information about the game's hidden design to participants before the whole cohort has played it.

  How to change monetary rewards: If your participants needs to receive a different amount than $25, go to the 'functions' subfolder and open 'round_counter.m'. In line 78, change '$25' to your amount. Save your changes.

- Centralize the .csv files generated during the game by copying them to a safe location after each participant.
  Warning: the .csv files are semicolon delimited; they may not be directly converted to tables in Excel depending on Excel's settings. To visualize them properly, select the first column of your data in Excel, then go to "Data > Text to columns" and change the delimiter to semicolon.

Troubleshooting

o *If the code does not start in MATLAB, or if Snake practice does not work properly, verify that MATLAB is in the right folder (in the folder 'cobra_task_V1' where the file 'main_game_launcher.m' is). If it is not the case, go to **MATLAB's command window** and enter "cd" followed by a space, and then the folder's location. E.g.:* `cd C:\Users\npgames\Desktop\cobra_task_V1`

o *Only the keyboard should be used during the snake game phase of the task (this part is coded in Python and is only superimposed on the MATLAB interface). Make sure to remind the participants that they are not supposed to put their finger on the screen/mouse during this phase. Most importantly, one must <u>not</u> click anywhere else on the screen while the window with the snake is visible. Doing so, may hide the snake game screen behind the other MATLAB windows. If this happens, the game will continue in the background, <u>so do not close any windows</u>! Just minimize all MATLAB windows, click anywhere on the snake game window and let the patient play the game normally. Once finished (after the usual 20 minutes timeout), you can restore all MATLAB windows to full size. The game is then back on track.*

o *If the game crashes, you will need to press "ctrl + C" in **MATLAB's command window**, in order to end the script completely. Once you did this, make a screenshot of the error message (or copy the error message somewhere else). Make a copy of the data collected until then. <u>Do not relaunch the game if the participant has already started the second trial</u> because then they can know that the outcomes are rigged, since they will be pitted against the same opponents again.*

Codebook of collected variables

Variable name	Measured construct
ID	Particpant's ID
name	The name entered by the participant
gender	1 = male, 2 = female (if the participant identifies as neither, change it manually to "3" in the datafile)
age	participant's age
gameExp	Gamimg experience. Continuous variable with range 1 = 'played < 5 times/lifetime' to 5 = 'plays weekly'
rankEstim1	Final rank estimated by participant before competing. 1 = highest, 200 = lowest.
avatarChoice	Choice of avatar picture between the four available options.
consentChoice	1 = agrees to add own performance to the public ranking. 0 = doesn't agree.
snakeLevel	Speed of the snake game: set on the basis of the Snake practice (number of apples eaten during the practice phase). 1 = slowest, 2 = medium (default), 3 = fastest
excited1, upset1, … determined1	PANAS affective scales at the <u>beginning</u> of the task. Values are coded between 1 and 5 on a Likert scale. 0 = 'NA' (cycled through the affect too quickly).
trial	the current trial: integer between 1 and 24
score	number of points (apples) the participant got on a given trial (beware
oppScore	number of points (apples) got by the opponent on a given trial (computed based on the following formula (scoreDiff is added if victory = 0, and substracted if victory = 1): round{(score **+** 1 **+/-** scoreDiff)**/**2}
scoreDiff	absolute value of the difference between the participants' score and the opponent's (pseudorandom)
win	1 = victory on a given trial, 0 = defeat (pseudorandom)
close	Message displayed 10 seconds before the end of the Snake game phase. 1 = close-run, 0 = not a close-run (pseudorandom)
oppName	the opponent's name (pseudorandom)
oppRank	the opponent's rank (pseudorandom); 1 = best rank, 200 = worst rank
rankStart	rank at the <u>end</u> of the trial, before rank buying (rankChoice); 1 = best rank, 200 = worst rank
rankEnd	rank when finishing the trial, after rank buying (rankChoice); 1 = best rank, 200 = worst rank
appleChoice	point stealing: 1 = no points stolen, 2 = 1 point stolen, 3 = 2 points stolen, 4 = 5 points stolen, 5 = 10 points stolen
rankChocie	rank buying: 1 = no rank bought, 2 = 1 rank bought, 3 = 2 ranks bought, 4 = 3 ranks bought, 5 = 5 ranks bought
excited2, upset2, … determined2	PANAS affective scales at the <u>end</u> of the task. Values are coded between 1 and 5 on a Likert scale. 0 = 'NA' (cycled through the affect too quickly).
mot1, mot2, …	end-game motivation questions, measured on a Likert scale (1 = strongly disagree to 5 = strongly agree): mot1 = I wanted to perform as well as I possibly could on the task. mot2 = Maximizing my personal record of apples eaten was important to me. mot3 = I wanted to perform better than everyone else on the task. mot4 = I did not want to perform more poorly than everyone else on the task. mot5 = Attaining the highest rank among all the competitors was important to me. mot6 = I wanted to take revenge on people who defeated me. mot7 = I did not want to perform more poorly than I possibly could on the task. mot8 = I wanted to ensure that I win.

enjoyed, satisfied, …, credible	other end-game questions, answered on an analog scale from 0 (not at all) to 10 (absolutely): enjoyed = 'How much did you enjoy playing?' satisfied = 'How well do you think you played?' fair = 'On average, how fair did your opponents play?' credible = 'To what extent did your actions influence the outcome of the game?'
rankEstim2	The rank participant thinks they should have obtained after finishing the competition. 1 = highest, 200 = lowest.