## **Costly PANDA Interference Coding Manual**

## **Last Update**

## 2/8/21

## **Experimenter Key**

186 – Anjola Tega

## General Information:

This manual outlines how to use Datavyu to code for interference in PANDA videos for the Costly PANDA study. We will be coding for instances of interference (from a parent, sibling, or other) during the study that could have influenced the child’s responses in a way that would make the data for that question unreliable. Our primary goal is to determine the **rate** of interference in a randomly selected subset of Costly PANDA videos (20%), in each condition. If the rate of interference is higher than that detected from previous PANDA studies, we will code all of the Costly PANDA videos and exclude on a trial-by-trial basis. If more than 25% of any child’s responses need to be excluded due to interference, we would instead drop the participant entirely.

Interference can take many different forms. They may occur verbally, through gestures, or even via subtle glances. Some interactions may cause serious interference, and other interactions will not cause interferences at all. To account for the nuances in what may be considered interference, we will first code the *type* of interference (e.g., talking, gesturing, looking) and we will then code the *level* of interference: (1) parents **restating or reframing** the question in potentially biased way, (2) parents **directly steering the child** toward a particular response, and (3) parents **overruling a child’s response** or answering themselves. More information on coding the *type* and *level* of interference this can be found on page 6 (see “Interference Column”).

## Accessing Datavyu Spreadsheets:

The coding spreadsheets for this study will be saved on the server (Data 🡪 Research Assistants 🡪 PANDA Datavyu Coding 🡪 DV Interference Coding 🡪 DV Spreadsheets 🡪 Costly PANDA). However, before coding, you should *save the spreadsheet you intend to code to your Desktop* *in a folder called “Datavyu”*, and code from that file. This practice is important for two reasons. **First**, coding from local Datavyu files (i.e., those saved to your Desktop) will be much less glitchy than coding from files saved to the server. **Second**, the script you will run to import timing data from Qualtrics into your spreadsheet is programmed to run *only* onDatavyu files that are saved to your Desktop in a folder called “Datavyu.” Once you have finished coding a Datavyu spreadsheet from your Desktop, please move it back to the server (… 🡪 DV Spreadsheets 🡪 Costly PANDA). Lastly, don’t forget to update the Costly coding log so we can track your progress.

## Accessing PANDA Videos:

All of the PANDA videos necessary for coding Costly PANDA will be in the server (Video 🡪 PANDA Videos 🡪 Costly PANDA 🡪 Costly PANDA\_Converted). However, it is critical that you *do not access the video directly from the server*. Instead, please copy the video from the server, save that copy to your Desktop, and then add *that* copy to the Datavyu file you’re coding (i.e., when you click “Add Data”). Just as with the spreadsheets, coding from video that is directly on the server will slow down your coding process, so you *always want to be coding from a local video file*. Once you finish coding a video, please delete it from your Desktop (and make sure the video you copied from the server is still there!). Please remember to save all Datavyu spreadsheets in the following format: [Subject ID]\_[study name]\_[study date] (e.g., “480H\_costly\_9.16.20”).

## File Required for Importing Timing Information:

Many studies, including Costly PANDA, capture the *time* that it takes participants to reach different points in the study (e.g., soundcheck, consent, a certain task, etc.). For these studies, Step #5 of the “Coding Protocol” will ask you to run a script to import this timing information into your Datavyu file—this will make it easier for us to keep track of *when in the study* the various interferences you code for, occurred. To complete this step, you will need a copy of the raw Qualtrics data, and this will need to be saved to your Desktop in a folder called “Qualtrics”; otherwise, the script won’t run properly. Your supervisor will send you that file; you don’t need to do anything with it other than save it (keeping it in .csv format) and moving it into folder on your Desktop labeled “Qualtrics.”

## Coding Notes:

Before you start, don’t code files that break from protocol or if you find an example that doesn’t match the coding manual. Make a note in the Costly PANDA Video Coding log so that another video can be selected for coding.

### For example, if the video is for the wrong study, the participant is absent from the video, or if there are any other major deviations to the protocol (e.g., an apparent technical error that crashes the program).

## Coding Protocol (Study-Specific):

1. Check the excel spreadsheet titled “**costly\_coding\_log**” to find which participants’ videos need to be coded (Data 🡪 Research Assistants 🡪 PANDA Datavyu Training 🡪 Datavyu Interference Coding 🡪 DV Spreadsheets 🡪 Costly PANDA). When you are ready to start coding, write your researcher ID in the “Coder” column, and write the date in the “Coded Date” column.
2. Locate the Datavyu file for the PANDA participant selected in the coding log from Data 🡪 Research Assistants 🡪 PANDA Datavyu Training 🡪 Datavyu Interference Coding 🡪 DV Spreadsheets 🡪 Costly PANDA.
3. Code the **ID** column (see *Coding Instructions*, below).
4. Code the **Task** column (see *Coding Instructions*, below).
5. Run the “**import\_qualtrics.rb**” Ruby script to embed timing information *specific to the participant video you’re coding* into the Datavyu spreadsheet. To run this script, please do the following:
	1. In Datavyu, go to Script 🡪 Run Script.
	2. Navigate to the Server (on a Mac, under “Devices on the left), and select Data 🡪 Research Assistants 🡪 PANDA Datavyu Coding 🡪 Scripts for Supervisors 🡪 Ruby Scripts 🡪 Timing Data Import 🡪 import\_qualtrics.
	3. Let the script run (this may take a few seconds).
	4. Once the script is finished running, close out of Datavyu.
	5. Locate the Datavyu spreadsheet you were working on in your Desktop (🡪 Datavyu). You should find that a column has been added called “Trial” – this column contains cells for each part of the study, anchored at the exact times that participant saw it.
	6. Drag the “Trial” column in between “Task” and “Consent.”
6. Code the **Consent** column (see *Coding Instructions*, below).
7. Code the **Interference** column(see *Coding Instructions*, below).
	1. Note: you should *only* code for interference in the following trials: [X, Y, Z]. All parts of the video outside of these trials do not need to be coded for interference.
8. Enter comments in the **Comments** columns as needed throughout coding (see *Coding Instructions*, below).

**Speed**

Please watch video at 1x (real-time) speed.

# **Columns**

Each Datavyu file will contain the following columns:

id <id>, <dot>, <coder>

 task <task\_cp>

interference <interference\_tqlgo, interference\_rlo>

comments <comment>

# **Coding Instructions**

## 1. Id column

### Coding Strategy

-Input information into cell; no watching of video required.

-Set onset and offset at 00:000 (no timing information needs to be logged here), this is often known as a “point cell”.

### <id>

Subject number. Can’t be missing data. (This is part of the video file label.)

e.g. 51H

### <dot>

Test date. Can’t be missing data. (This is part of the video file label.)

Format: mm/dd/yyyy (e.g., 12/30/2008). 🡨 Note slashes, not periods!

### <coder>

Your researcher ID

## 2. Task column

### Coding Strategy

-Watch the video at full speed.

-When you hear verbal consent from the parent and child jump back 5 seconds (-).

-Keep listening in full speed to set onset and offset.

### <task\_cp>

#### c = consent

part of the video when parent and child agree to participate in the study

##### Onset

5 seconds before the narrator started to talk (listen until you hear narrator say *“Now, we need the parent or legal guardian and the child participating in this study to give us their permission to be a part of our research”,* then jump back 5 seconds) and set onset (enter).

##### Offset

Set offset when child says “yes or no” (9)

#### p = protocol

main part of study when child is answering questions

##### Onset

First frame after consent.

Set by selecting the consent cell under Task, hitting the “jog” button once (3 on the keypad), and then hitting 0 on the keypad. This creates a new cell just after the consent cell, one frame away.

##### Offset

End of video!

\*\*\*RUN IMPORT\_QUALTRICS.RB SCRIPT\*\*\*\*

## 3. Consent column

### Coding Strategy

-Highlight the consent cell from the Task column.

-Select Spreadsheet 🡪 New Cell to the Right to create a cell with identical onsets/offsets in the Consent column (or, simply click Command + R).

-Watch video in real time (1x speed) to code for parent and child consent in this cell.

### <parent\_ynp>

#### y = yes

parent reads script

#### n = no

parent says no, parent does not read script

#### p = passive yes

parent doesn’t read script but still participating

### <child\_ynp>

#### y = yes

child says yes or shakes head yes

#### n = no

child says no or does not respond, child says something random

#### p = passive yes

child doesn’t answer but still participating

#### Onset

Copy from task column

#### Offset

Copy from task column

## 4. Interference column

### General Criteria

The goal of this coding pass is to find and define all instances of interference from a parent, sibling, or other person. Interference from non-human sources (e.g. a doorbell) should also be coded. However. You should **ONLY** code events that could potentially affect a child’s responses. If you are uncertain about whether an event might have influenced a child’s response, code it and make a comment explaining why you are unsure in the Comments column.

### Coding Strategy

-Jump to the onset of the trial you’re coding.

-Start watching the video in full speed.

-Once you hear/see something that you think is an example of interference, jump back 5 seconds and set the onset.

🡪**Note:** If more than one instance of interference occurs within the same 5-second timespan (such that coding each separately would result in the cells overlapping), code this as one cell. Use the letter code that captures the *greatest* or most salient type of interference from that interaction, and note which other types of interferences were present in the Comments section.

-Next, you will code two elements of the interaction, **each as a separate argument**: (1) the *type* of interaction that occurred (interferencetype\_tqlgo), and (2) the *level* of interference that occurred (interferencelevel\_rlo).

🡪 **Note:** For example, a case of interference in which the interference type was “talking” and the interference level was “leading a response” would be coded as (t, l) under the Interference column.

🡪 **Note:** In general, remember to only code for instances that would constitute one of the interference levels (i.e., a form of reframing/rephrasing, leading, overruling). A non-verbal gesture *may* still count as an interference even if it doesn’t receive one of those codes (instead receiving an *na*), but it is unlikely—so please be sure to only code for behaviors that you think may constitute interference.

-Play forward in full speed until the event ends.

-Set the offset.

### <interferencetype\_tqlgo>

**t = talking**

Parent is talking to the child about the study in a way that could influence their answer. Examples include verbalizing a response that the child should choose, reframing a question in a way that emphasizes one answer to a greater degree than the others, providing context or any other information that we did not originally provide, and that the child could use in their response, or providing validation for responses (e.g., “I agree”). If a child speaks to their parent but their parent does not respond, this should *not* be coded as “t” (since there could not have been any parent interference); if you think the child speech may be important or relevant (e.g., “this game is making me hate Zarpies!”), you can make a note about it in the “Comments” column (see below).

#### q = question

Child is asking parent question (e.g., “which one should I pick?”) or parent is asking child question (“are you sure that’s your answer?”) in a way that could influence child’s response. If anything else is said *other than* (or in addition to) a question (e.g., a child saying “what should I chose?” and the parent responding “this one”), this should be coded as a *t*.

#### l = looking

Parent and child are communicating nonverbally in a way that could influence child’s answer. Please note that this does not include instances in which children are gazing in a different direction or at something in the room—this is not the same as an interference (but can always be noted in the Comments, if you think it might have influenced the study session in other ways).

#### g = gesture

Parent is pointing or gesturing to an answer with the intention of altering or determining the child’s response.

### o = other

Other source of interference that influenced child’s answer. Examples include a sibling interfering, a loud noise distracting the child’s attention so they don’t hear the question, or technical interference that prevented the question from playing fully **in a way that prevented the child from being able to answer.**

### <interferencelevel\_rlo>

#### r = reframing or rephrasing question

Parent restates or reframes the question to help aid the child’s comprehension. Sometimes parents will do so neutrally, and other times they may emphasize one response over another—however, if all they are doing is reframing/rephrasing the question, the interaction would earn this code.

#### l = leading a response

Parent tries to lead their child toward a particular response by either challenging the child’s response, gesturing toward the screen in a suggestive way, or exhibiting another behavior that makes clear the parents’ intention to steer the child in a particular direction, especially after the child had already responded.

**o = overruling**

Parent provides a response to the question instead of the child (i.e., by overruling the child’s answer or simply answering on their own).

**na = not applicable**

 When the interaction is not verbal (e.g., when the first code in this column receives an **l, g,** or **o**)**,** it may be difficult to tell whether the parent is reframing/rephrasing a question, leading a question, or overruling a response. Of course, it is not impossible for a non-verbal behavior like a gesture to steer a child’s answer—in which case, the code would be (g, l)—but it may not always be easy to tell in the absence of words. In these cases, the second argument for this code should be “na” (e.g., (g, na)).

## Comments column

### General Criteria

* Type in any necessary explanations for relevant interactions, touching upon both the *type* and *level* of the interference.
* For example, who spoke, looked at, or gestured to whom (parent to child, child to parent, sibling to child, etc.), what was said if important, and any other information about interaction that is helpful.
* Be as brief but informative as possible.
* Grammar does not matter much, skipping articles is fine, but be careful with shorthand to avoid confusion.
* Don’t add too much unnecessary detail, only enough to inform another of what the situation was when warranted.
* Mark any codes you are uncertain about (and why) here.
* Also, type in any spontaneous comments made by the child or parent that is relevant to the study. For example, a child explaining or justifying their response choice, or a parent’s feelings about a question or the survey as a whole.
* Please also use this column to mark any interesting or insightful comments made by the child or parent that could be interesting to include as example videos in a future talk. Please type in the comment.

### Coding Strategy

Set onset and offset for the same time as the parent-child interaction cell the note corresponds (to do so, go to Spreadsheet 🡪 New Cell to the Right, or click Command+R). Overall, just be sure that cells in the comment columns match to their corresponding cell in the interference column.