



# GlassPy Process Data Analytics Solution and its Applications at Educational Testing Service

Jiangang Hao  
Educational Testing Service  
6/18/2020

# About ETS and Process Data

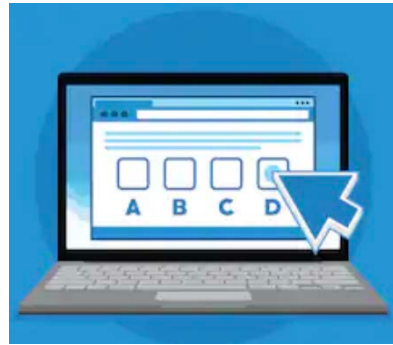
- ETS, funded in 1947, is a large non-profit testing company headquartered in Princeton, New Jersey, USA.
- We carry out about 50 million tests worldwide every year. Some well-known tests includes GRE, TOEFL, and SAT.
- We have a large R&D division, doing research on psychometrics, statistics, natural language processing, reading, writing, etc.
- <https://www.ets.org/research>
- ETS scientists work on process data in various programs and projects:
  - PISA and PIAAC
  - NAEP
  - Winsight
  - Writing analytics
  - Collaborative problem solving
  - Remote testing
  - ...
- Assessment Analytics and Data Science: a group dedicated to research on process data

# Process Data and their Uses

**Process data** capture fine-grained interaction information between learners and learning/assessment tasks, allowing better understanding of the learners and the tasks



People



Tasks

## Typical uses

- Improve psychometric quality
- Evidence for process-based constructs
- Group difference and fairness
- Test security
- Feedback to learners and other stakeholders
- Improve the tasks and delivery platform

# Challenges of Process Data in Practice

- Engineering aspect
  - Capturing, hosting, transferring, and manipulating data with proper software infrastructure, tools and techniques.
  - Data engineer/Architect
- Scientific aspect
  - Making sense of data to realize their potential value for intended interpretations and uses
  - Data scientist, Psychometrician, Statistician
- Organizational aspect
  - Workforce and professional development
  - Identifying suitable workflow (e.g., agile, waterfall, etc.)

# GlassPy Analytics Solution at ETS

- GlassPy: **G**ame **L**og **A**nalysis in **P**ython, started in 2014 at ETS
- We envision glassPy more as a generic framework for handling process data from digital-based assessments (DBAs)
- It is an “adaptor” to connect the Evidence Centered Design (ECD) to data in practice
- GlassPy analytics solution Includes three main components

**Evidence Identification Centered Data Design - EICDD**

**GlassPy Library**

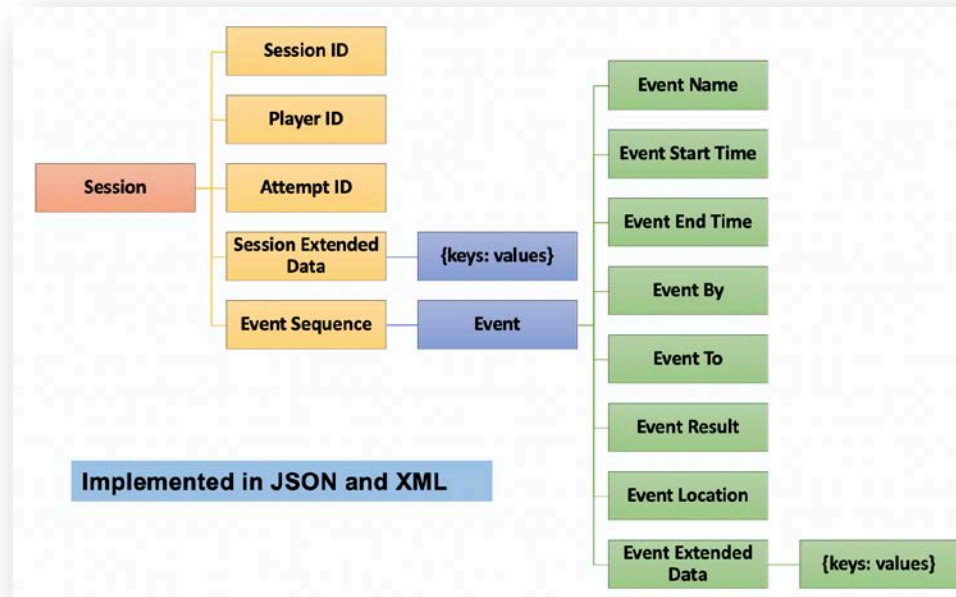
**Interactive Analytics Frontend**



# Evidence Identification Centered Data Design

## Data Model

## Standard Operational Procedure



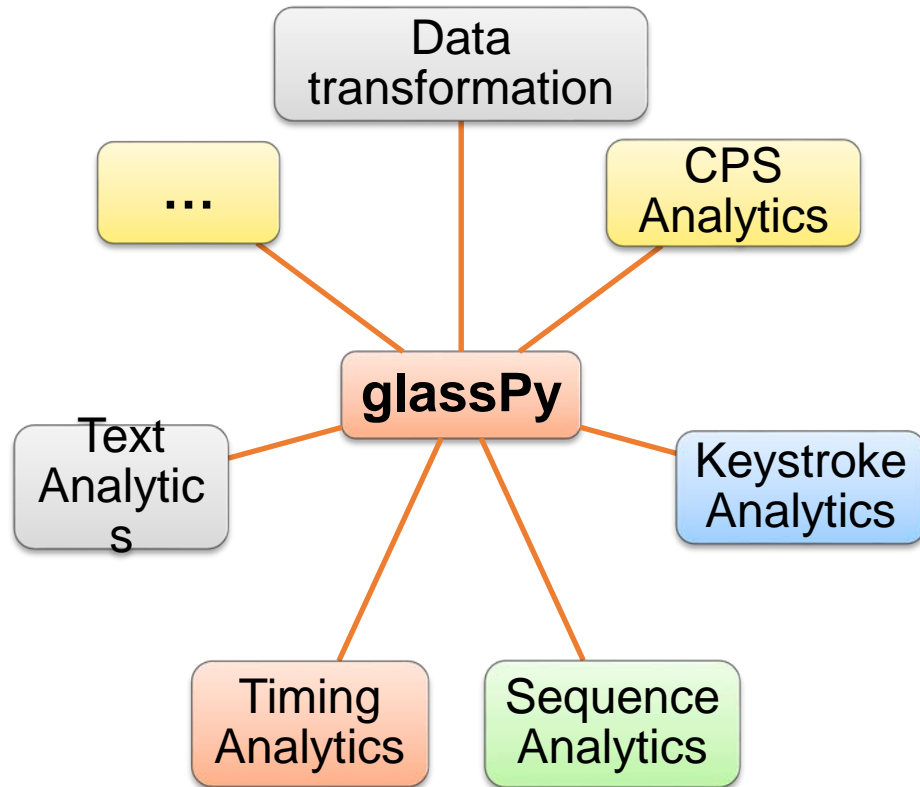
X: participant    XX: coordinator

		Roles				
		Learning Scientist/ Assessment Developer	Psychometrician	Programmer	Data Scientist	Data analyst
Timeline ↓	Step 1: All-party meeting to brief the procedures	X	X	X	XX	X
	Step 2: Specification of "Q matrix" for constructs and evidences	X				
	Step 3: Specification of additional evidence for psychometric modeling need		X			
	Step 4: Translate the evidences into operable task codes			X	X	XX
	Step 5: Design log file structure and schema				X	
	Step 6: Implement the evidence logging into the log file and tryout			X	X	XX
	Step 7: Log file parsing and evidence extraction based on the tryout data				X	
	Step 8: QA the log files to check whether the recovered evidence is sufficient. If not, restart from Step 6. If everything is good, proceed to next step	X	X		X	XX
	Step 9: Finalize the data reduction pipeline and QA procedure				XX	X
	Step 10: Implement the data reduction				X	XX

## Evidence Trace File



# GlassPy Library



```
def find_sub_str(sl, l):
    """
    Purpose: return the location of the sub-string/list in a string/list for multiple matching::

    Syntax: find_sub_str(sl, l)

    Input: sl sub-string/list, l: string/list

    Return: location of the sub-string/list in a string/list.

    Created By: J. Hao @ 2014
    """
    sl = list(sl)
    l = list(l)
    results = []
    sl1 = len(sl)
    for ind in (i for i, e in enumerate(l) if e == sl[0]):
        if l[ind:ind + sl1] == sl:
            results.append([ind, ind + sl1 - 1])
    return results

def ngram_freq(event_list, n=1, sorted=False):
    """
    Purpose: calculate the frequency of ngrams of the event list ::

    Syntax: ngram_freq(eventList, ngram=1)

    Input: GameDataFrame

    Output: Dictionary of ngram and the frequency

    Created By: J. Hao @ 2014
    """
    if event_list is None:
        return 'Syntax: ngramFreq(eventList, ngram)'
    else:
        event_list = list(event_list)
        ngm = ngrams(event_list, n)
        ngm = cL.Counter(ngm)
        df = pd.DataFrame([list(ngm.keys()), list(ngm.values())])
        df = df.T
        df.columns = ['events', 'freq']
        if sorted:
            return df.sort_values(by='freq', ascending=False)
        else:
            return df

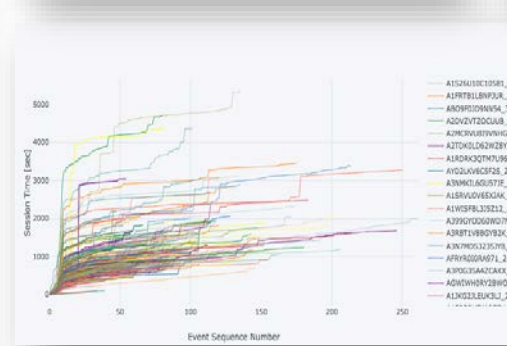
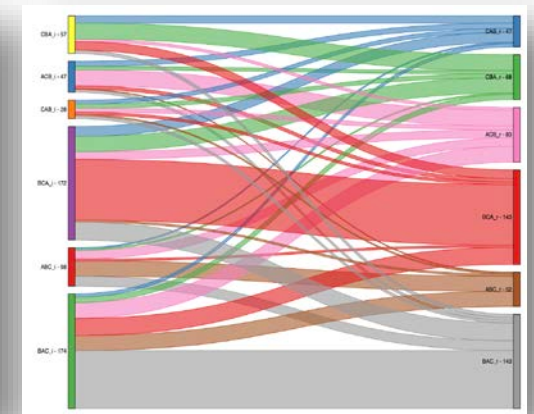
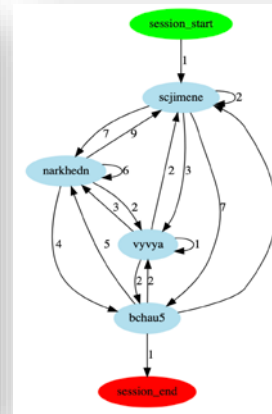
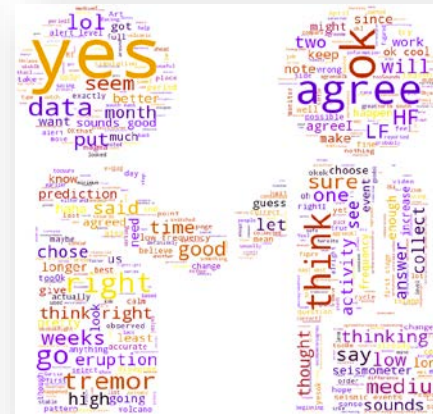
def regularize_htmltext(res):
    reg_response = re.sub('<[<~>+7>', ' ', res).replace('&nbsp;', ' ').replace('\xc2\xa0', ' ')
    reg_response = re.sub('\x00-\xff', '', reg_response)
    return reg_response
```

Not open sourced yet and under active expanding

# Interactive Analytics Frontend

Jupyter/Zeppelin Notebook/Lab + Plotly Dash + Shiny + FastAPI + others

```
jupyter GameLogSafari Last Checkpoint: 04/24/2017 (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Trusted Python 2.0
Code Dashboard View:
GameLogSafari: An Interactive Analytics Frontend for GlassPy
Jiangang Hao
Educational Testing Service, Princeton, NJ 08541 jhao@ets.org
Citing: Hao, J., Smith L., Mislevy, R., von Davier, A., & Bauer, M. (2016). Taming log files from the game and simulation-based assessment: Data model and data analysis tool. ETS Research Report RR-16-11. Princeton, NJ: Educational Testing Service.
Copyright © Educational Testing Service
1. Loading the needed python packages
In [1]:
!matplotlib inline
import re, mpid3, nltk, json, warnings#, ggrid
import glasspy as gp
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb
import cufflinks as cf
from ipynbwidgets import interact
import networkx as nx
import pygraphviz as gv
import graphviz as gv
from networkx.drawing.nx_agraph import graphviz_layout
warnings.filterwarnings('ignore')
cf.go offline()
mpid3.disable_notebook()
2. Specifying the file name
In [2]:
#xml_file="/Users/jhao/PycharmProjects/glasspy/exampleLogFile/simcitedu_sampleLog_short.xml"
#xml_schema="/Users/jhao/PycharmProjects/glasspy/schema/gameLog_schema.xml"
json_file="/Users/jhao/PycharmProjects/glasspy/exampleLogFile/simcitedu_sampleLog_short.json"
#json_file="/Users/jhao/research/extended_game/434-1-80-2-1452534571.json"
json_schema="/Users/jhao/PycharmProjects/glasspy/schema/gameLog_schema.json"
3. Check and read data
3.1. XML Schema validation, read in data
In [5]:
gp.xml_validation(xml_file, xml_schema)
```



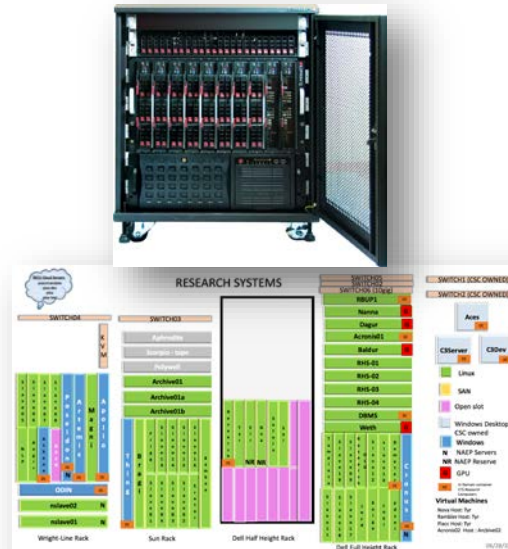


# Scaling up in Operational Work

Individual computers



Computing grids



Cloud computing



# A Full Implementation in EPCAL

Fully implemented in the ETS Platform for Collaborative Assessment and Learning

The screenshot shows the ETS Platform for Collaborative Assessment and Learning (CBAL) interface. At the top, it says "ETS Platform for Collaborative Assessment and Learning" with a "Log out" button. Below that, there are video feeds for two participants: LIN and Jiangang, both marked as "Connected". A progress bar shows "55%" and a "SUBMIT ->" button. The main content area is titled "Question 4 of 7" and contains a simulation about condensation. The simulation text reads: "Ok, we concluded that liquid water formed on the surface of the cold can but not on the warm can. When a gas substance changes to a liquid substance, we call this condensation. Now you will use a simulation to explore the phenomena of condensation. Your goal is to explain why liquid water forms on a cold can surface but not on a warm can surface. In the following simulation, a soda can is placed in a humid environment. You can control the temperature of the can and observe what happens to the particles around the can." Below the text is a simulation window showing a can and blue particles. At the bottom left, there are controls for "Water" and "Temperature of can:" set to "70°F" with a "play" button. On the right, there is a "Text Chat Box" with messages from Jiangang and LIN. Jiangang's messages are "hi, how are you" and "this looks good". LIN's message is "what is your options?". There is a "Send" button at the bottom of the chat box.



```
<?xml version="1.0" encoding="UTF-8" ?>
<session>
  <sessionID> 5931 </sessionID>
  <teamID> A2LCGAJX39Rm7M_A423QQ5W4389 </teamID>
  <playerID>
    <attemptID> 1 </attemptID>
  </playerID>
  <sessionExtData>
    <eventSequence>
      <event>
        <eventName> negotiation_choice </eventName>
        <eventStartTime> 2018-06-22T18:34:04Z </eventStartTime>
        <eventEndTime> 2018-06-22T18:34:04Z </eventEndTime>
        <eventBy> A423QQ5W4389 </eventBy>
        <eventTo> Robot's name </eventTo>
        <eventResult> Survivor </eventResult>
        <eventLocation> slide1-step0--Robot Task Negotiation </eventLocation>
        <eventResultAltData />
        <eventExtData>
          <cpalP>
        </eventExtData>
      </event>
    </eventSequence>
  </sessionExtData>
</session>
```

The screenshot shows the EPCAL Analytics Frontend interface. At the top, it says "EPCAL Analytics Frontend Powered by glassPy" and "Copyright © Educational Testing Service Support contact: [help@ets.org](mailto:help@ets.org)". Below that, there are two sections: "1. Loading the needed python packages" and "2. Specifying the file name". Section 1 shows Python code for importing packages like matplotlib, re, json, warnings, numpy, pandas, matplotlib.pyplot, seaborn, and cufflinks. Section 2 shows Python code for loading an XML file and creating an adjacency matrix. Below the code is a heatmap visualization. The heatmap has a color scale from 0.00 (green) to 0.60 (red). The x-axis and y-axis are labeled with event names: "slide1", "negotiation", "selection\_made", and "user\_animation". The heatmap shows a strong correlation between "slide1" and "negotiation" (red cell).

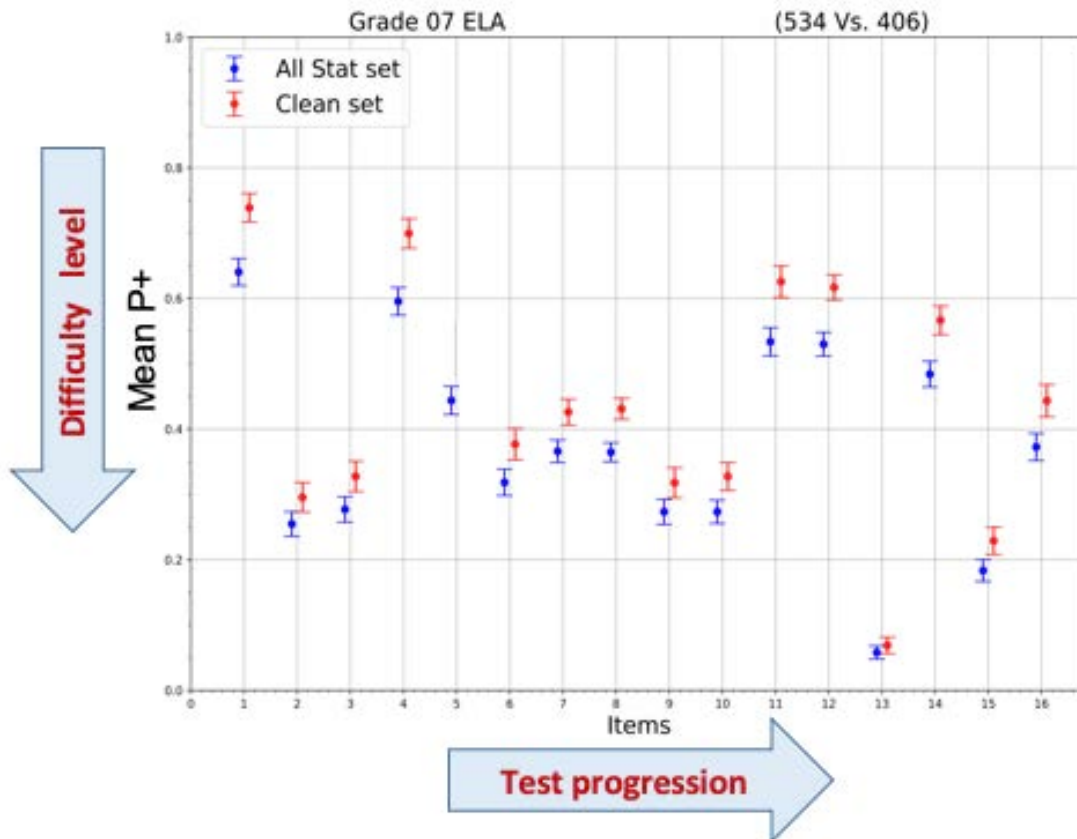


# Analytics vs. Modeling

- Analytics
  - Discovery, interpretation, and communication of meaningful patterns in data and apply these patterns towards effective decision making.
  - Suited for data from most digital learning and assessment tasks
- Modeling
  - Psychometric/statistical modeling (e.g., Bayes net, IRTs, CDM, many others)
  - Cognitive modeling (e.g., ACT-R, SOAR, many others)
  - Better suited for data from carefully designed virtual performance tasks
- Complication in real-world data
  - Response process is highly regulated by task/item design, limiting the generalizability of methods.
  - Stationary condition generally does not hold throughout, requiring regime switching.
  - The response sequences span a big and sparse mathematical space, making it challenging to identify suitable statistical distributions.
  - The response process are often short, challenging the parameter estimation.



# Example 1: process data for item calibration



## Definitions:

- Clean set: students who complete the assessment in a single session
- All Stat set: students who complete the assessment in a single session and in multiple sessions

## Findings

- Different samples lead to different item parameter estimates
- The difference of item mean P+ is **not** affected much by the item location in the test
- Easier items are more affected

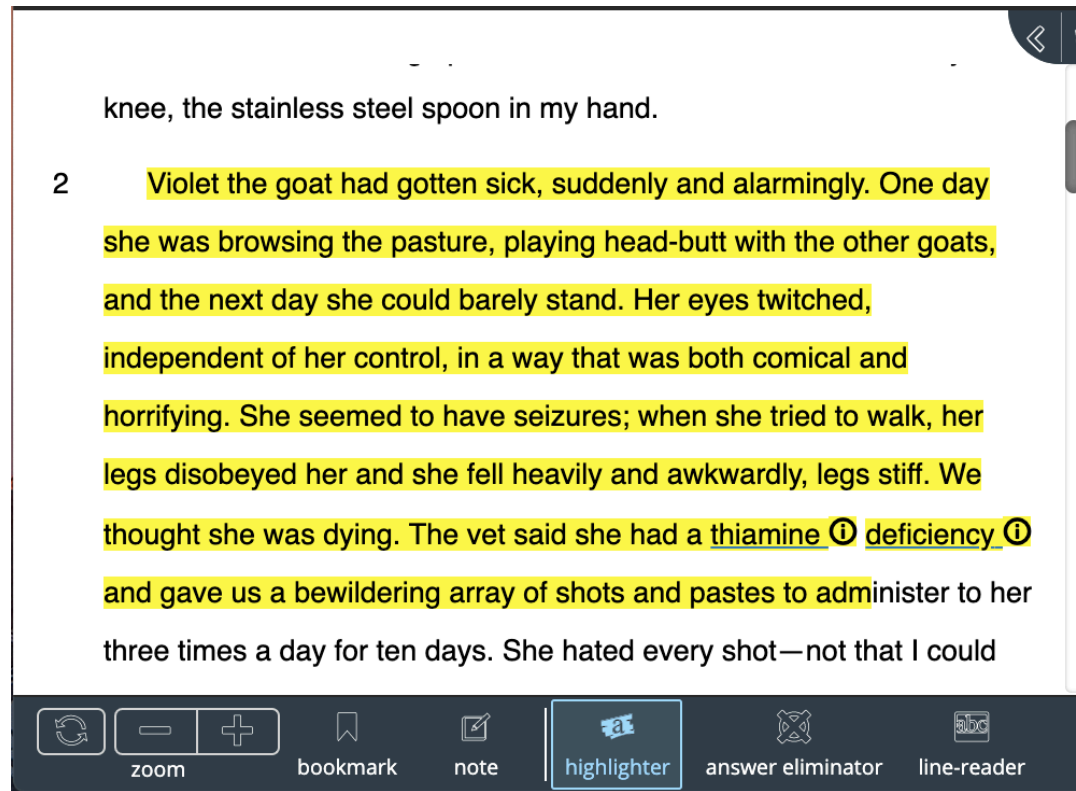
Based on data from ETS Winsight assessment

# Example 2: process data for group difference

## Highlighter tool

knee, the stainless steel spoon in my hand.

2 Violet the goat had gotten sick, suddenly and alarmingly. One day she was browsing the pasture, playing head-butt with the other goats, and the next day she could barely stand. Her eyes twitched, independent of her control, in a way that was both comical and horrifying. She seemed to have seizures; when she tried to walk, her legs disobeyed her and she fell heavily and awkwardly, legs stiff. We thought she was dying. The vet said she had a thiamine deficiency and gave us a bewildering array of shots and pastes to administer to her three times a day for ten days. She hated every shot—not that I could

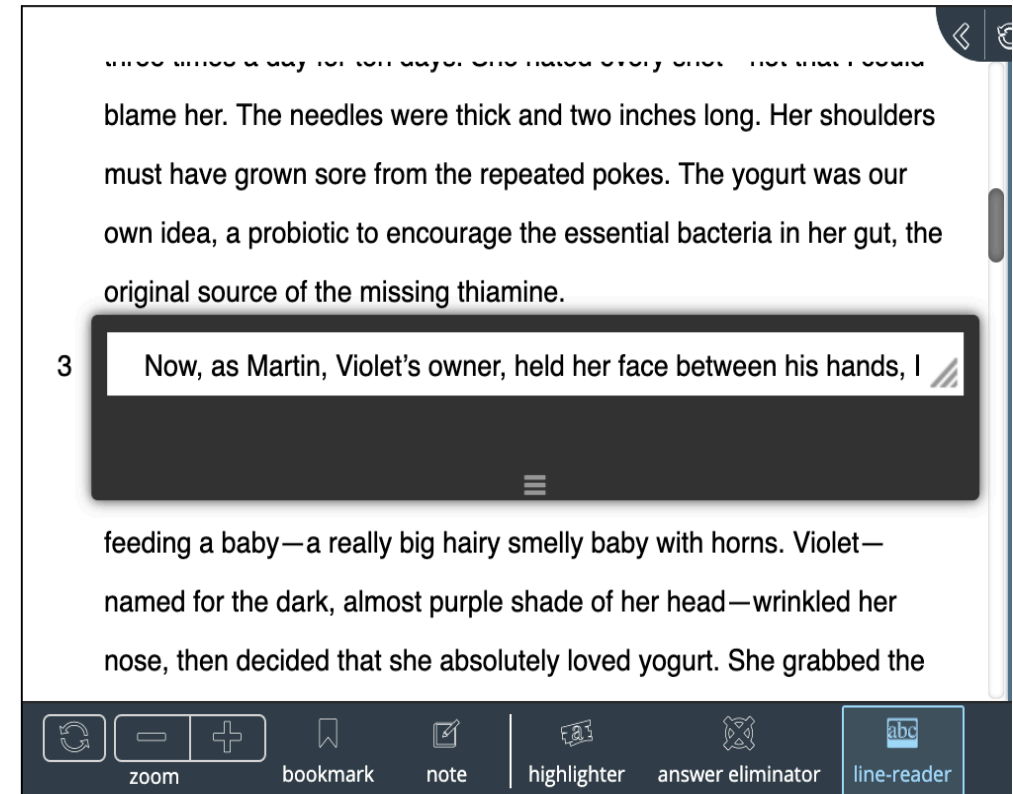


## Line reader tool

blame her. The needles were thick and two inches long. Her shoulders must have grown sore from the repeated pokes. The yogurt was our own idea, a probiotic to encourage the essential bacteria in her gut, the original source of the missing thiamine.

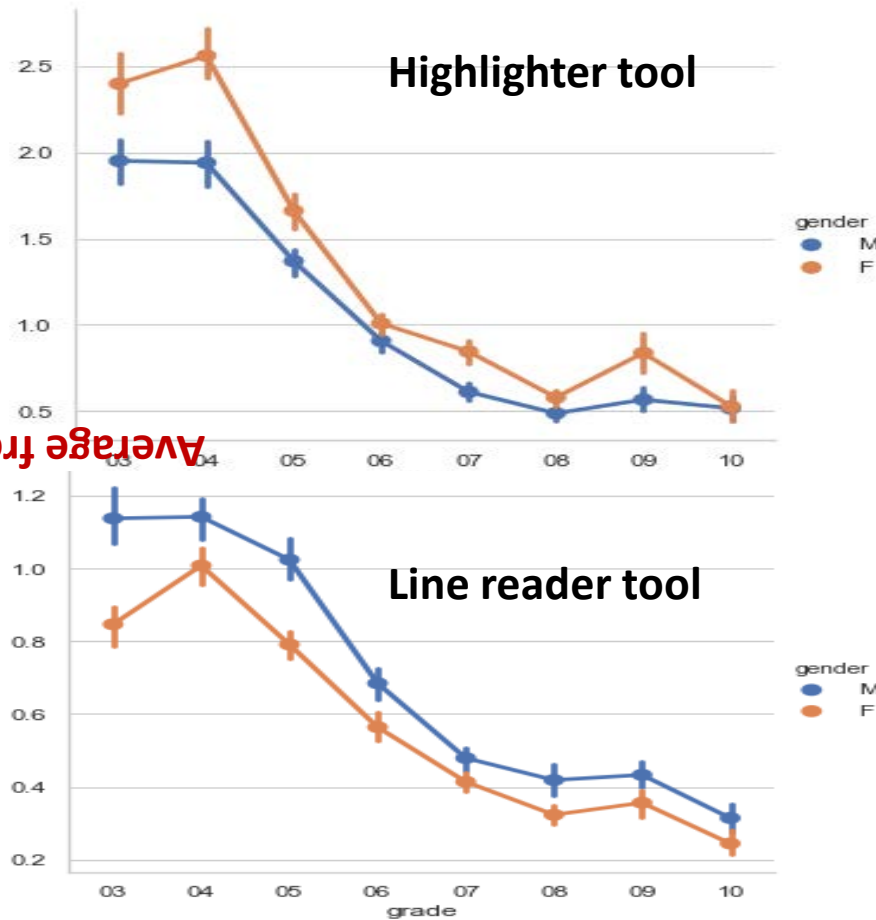
3 Now, as Martin, Violet's owner, held her face between his hands, I

feeding a baby—a really big hairy smelly baby with horns. Violet—named for the dark, almost purple shade of her head—wrinkled her nose, then decided that she absolutely loved yogurt. She grabbed the



From ETS Winsight Assessment Sampler

# Example 2: process data for group difference



Average frequency per session

Age group

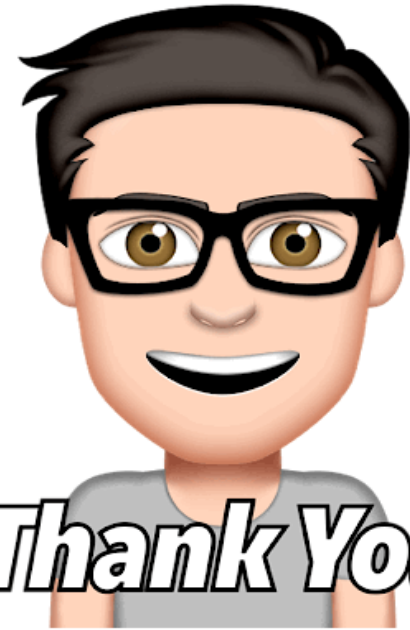
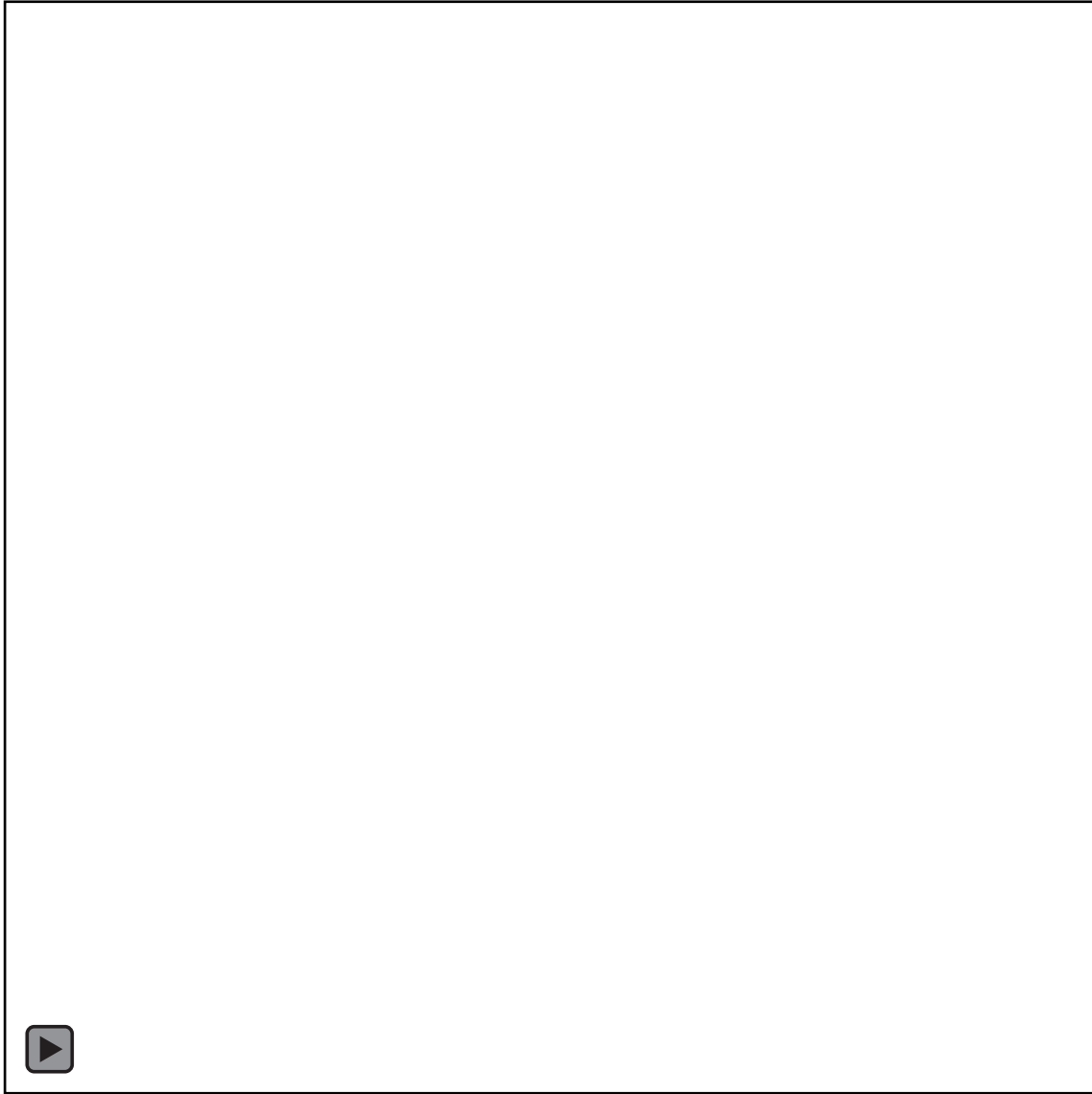
## Observations:

- Girls tend to use more highlighter tool
- Boys tend to use more line reader tool
- Decreasing trend of tool usage

**How will the tool usage affect the performance? What should be the right tools to be provided?**

# Summary

- Leveraging the process data can help to improve psychometric quality, discover new patterns and trends, inform better teaching and learning, and form new hypothesis to extend our knowledge horizon.
- Unleashing the full power of process data in large scale assessments requires the proper methods and tools. I hope the glassPy example can help you to think about your ways of dealing with process data.
- Traditional psychometric methods need to be extended to include methods from data science and machine learning to handle the increasingly complex data
- A forthcoming new book, Computational Psychometrics, edited by Alina von Davier, Robert Mislevy and Jiangang Hao will provide psychometrics researchers with a curated list of tools for addressing the challenges from complex data.



*Thank You*

**Email: [jhao@ets.org](mailto:jhao@ets.org)**

