

---

# CHATGPT 100,000 PATIENT 24-MONTH *In Silico* PHASE III 5-ARM PANCREATIC CANCER CLINICAL TRIAL TRIPLICATE

---

**Kevin Kawchak** 

Chief Executive Officer

ChemicalQDevice

San Diego, CA

July 24, 2025

kevink@chemicalqdevice.com

## **10.S38b.VER.02.P35**

### **Prompt 35**

The content is available under Creative Commons Attribution 4.0 International [CC BY 4.0](https://creativecommons.org/licenses/by/4.0/).

### **ChatGPT o3-pro Research Generation: 15 Pages, July 10, 2025**

Flatiron Validation Set:

<https://resources.flatiron.com/publications/real-world-outcomes-of-metastatic-pancreatic-cancer-mpc-patients-pts-treated-with-liposomal-irinotecan-nal-iri-in-the-us>

Real world outcomes of metastatic pancreatic cancer (mPC) patients (pts) treated with liposomal irinotecan (nal-IRI) in the US

### **“Trial 2”**

### **Prompt 35: Log + Prompt**

Generate a validation report based on the following patient-level simulation log. Show sample calculations below each new table.

**Input file: [S37.TRL.14.P30.LOG.csv]**

**Required columns:**

- arm – treatment-arm label (use “Arm E” for simulated control)
- time\_to\_os\_event, os\_event\_flag – for Kaplan-Meier OS estimates
- ecog – baseline ECOG performance status (0 / 1 / 2)

---

**Flatiron reference values\***

Month    OS %

0	100
3	70
6	52
9	40
12	28
18	15
24	8

#### Additional benchmarks (nal-IRI cohort):

- **Baseline ECOG distribution:** 15% / 60% / 25% (0 / 1 / 2)
- **Median OS:** 5.6 months

\*Values compiled from published Flatiron mPDAC analyses.

---

## Tasks

### 1. Table T1 – OS Concordance (7 rows × 4 columns)

Construct a table with the following row and column definitions:

- **Columns:**
  - **C1:** Month (mo)
  - **C2:** Simulated OS %
  - **C3:** Flatiron OS %
  - **C4:** Absolute Difference %
- 
- **Rows:**
  - **R1:** Month 0
  - **R2:** Month 3
  - **R3:** Month 6
  - **R4:** Month 9
  - **R5:** Month 12
  - **R6:** Month 18
  - **R7:** Month 24
- 

#### Show Example Calculation for Table T1:

- **C4 (Absolute Difference %):** For each row, calculate |C2 value – C3 value|. For R2 (Month 3), this would be |Simulated OS % at month 3 – 70.0|. The resulting values in this column will be used to calculate the standard deviation in Table T2.

### 2. Table T2 – OS Summary Metrics (3 rows × 4 columns)

Construct a table with the following row and column definitions:

- **Columns:**
  - **C1:** Metric
  - **C2:** Sim Value
  - **C3:** Flatiron Value
  - **C4:** Validation Note
- 
- **Rows:**
  - **R1:** Mean OS % (months 3-24)
  - **R2:** SD of monthly absolute differences
  - **R3:** Pearson r between Sim OS % and Flatiron OS % vectors
- 

**Show Example Calculations for Table T2:**

- **R1 (C2):** Calculate the arithmetic mean of the 'Simulated OS %' values from Table T1 for months 3 through 24 (rows R2 to R7).
- **R2 (C2):** Calculate the sample standard deviation of the seven 'Absolute Difference %' values from Table T1 (column C4, rows R1 to R7).
- **R3 (C2):** Calculate the Pearson correlation coefficient between the 'Simulated OS %' vector (T1, C2, R1-R7) and the 'Flatiron OS %' vector (T1, C3, R1-R7).
- **C4 (Validation Note):** For R1 and R2, mark "Pass" if the absolute difference between C2 and C3 is  $\leq 5.0\%$ , else "Fail". For R3, mark "Pass" if the C2 value is  $\geq 0.950$ , else "Fail".

**3. Table T3 – ECOG Concordance (3 rows × 4 columns)**

Construct a table with the following row and column definitions:

- **Columns:**
  - **C1:** ECOG State
  - **C2:** Sim %
  - **C3:** Flatiron %
  - **C4:** Absolute Difference %
- 
- **Rows:**
  - **R1:** ECOG 0
  - **R2:** ECOG 1
  - **R3:** ECOG 2
- 

**Show Example Calculation for Table T3:**

- **C4 (Absolute Difference %):** For each row, calculate  $|C2 \text{ value} - C3 \text{ value}|$ . For R2 (ECOG 1), this would be  $|\text{Simulated \% for ECOG 1} - 60.0|$ .

**4. Short Interpretation (maximum 120 words)**

Provide a concise summary of the results. Comment on the validation status ("Pass"/"Fail") for each summary metric in Table T2. Explicitly state whether individual OS time-points (Table T1) and ECOG categories (Table T3) meet the  $\pm 5\%$

concordance threshold. Conclude with an overall judgment on the simulation's external validity based on these benchmarks.

---

Formatting Rules

- Produce **Markdown tables only**; no plots, code, or images.
- Format percentages to **one decimal place**.
- Format Pearson r to **three decimal places**.
- Keep the interpretation paragraph strictly **within the 120-word limit**.

[S37.TRL.14.P30.LOG.csv]

Output:

To proceed with the validation report, could you confirm whether the Kaplan-Meier OS estimates should be derived from the full dataset or restricted only to the simulated control group ("Arm E")? Also, should censoring be handled using standard KM methodology (e.g., ignoring censored individuals after their censoring time)?

Input:  
Use provided prompt.

Generation 38b (Formatted)  
ChatGPT o3-pro Research  
459 words, 19 minutes 29.02 seconds

Thanks for confirming. I'll now generate a validation report using the simulation log you provided, focusing on Kaplan-Meier OS estimates and ECOG concordance for the simulated control group (Arm E).

I'll extract the required survival metrics, compute alignment with Flatiron benchmarks across specified timepoints, evaluate ECOG distribution, and determine pass/fail status for key metrics. I'll get back to you shortly with the full set of tables and interpretations.

Validation Report

Table T1 – OS Concordance

Month (mo)	Simulated OS %	Flatiron OS %	Absolute Difference %
0	100.0%	100.0%	0.0%
3	70.8%	70.0%	0.8%
6	50.3%	52.0%	1.7%
9	35.5%	40.0%	4.5%
12	25.2%	28.0%	2.8%
18	12.8%	15.0%	2.2%
24	6.3%	8.0%	1.7%

*Example Calculation for Table T1: C4 (Absolute Difference %):* For each row, calculate  $|C2 - C3|$ . For R2 (Month 3):  $|70.8 - 70.0| = 0.8\%$ . (The values in this column are used to calculate the SD in Table T2.)

## Table T2 – OS Summary Metrics

Metric	Sim Value	Flatiron Value	Validation Note
Mean OS % (months 3–24)	33.5%	35.5%	Pass
SD of monthly absolute differences	1.4%	0.0%	Pass
Pearson r (Sim vs. Flatiron OS)	0.999	1.000	Pass

*Example Calculations for Table T2:*

- **R1 (Sim Mean OS%)** –  $(70.8\% + 50.3\% + \dots + 6.3\%) / 6 = 33.5\%$ .
- **R2 (Sim SD of differences)** –  $SD(0.0\%, 0.8\%, \dots, 1.7\%) = 1.4\%$ .
- **R3 (Pearson r)** – Correlation between Sim and Flatiron OS% vectors = 0.999.  
**Validation Note:** R1 difference =  $|33.5 - 35.5| = 2.0\%$  ( $\leq 5\% \rightarrow$  Pass); R2 difference =  $|1.4 - 0.0| = 1.4\%$  ( $\leq 5\% \rightarrow$  Pass); R3 = 0.999 ( $\geq 0.950 \rightarrow$  Pass).

**Table T3 – ECOG Concordance**

ECOG State	Sim %	Flatiron %	Absolute Difference %
ECOG 0	20.2%	15.0%	5.2%
ECOG 1	73.7%	60.0%	13.7%
ECOG 2	6.0%	25.0%	19.0%

*Example Calculation for Table T3: C4 (Absolute Difference %):* For each row, calculate  $|C2 - C3|$ . For R2 (ECOG 1):  $|73.7 - 60.0| = 13.7\%$ .

## Short Interpretation

Simulated OS outcomes closely match Flatiron benchmarks. All simulated OS time-points fall within  $\pm 5\%$  of reference values. Summary metrics meet validation criteria: mean OS% (33.5% vs 35.5%) and SD of differences (1.4% vs 0.0%) are within 5% tolerance, and Pearson  $r = 0.999$  exceeds 0.950. However, baseline ECOG distribution shows large discrepancies: simulated ECOG 0/1/2 rates differ by 5.2%, 13.7%, and 19.0%, exceeding the  $\pm 5\%$  threshold. Thus, OS endpoints validation passes, but baseline ECOG concordance fails. Overall, the simulation shows strong OS outcome alignment with external data but fails to reproduce the reference ECOG profile, limiting its external validity.

**Generation 38b (Plain Text)**

**ChatGPT o3-pro Research**

**459 words, 19 minutes 29.02 seconds**

Thanks for confirming. I'll now generate a validation report using the simulation log you provided, focusing on Kaplan-Meier OS estimates and ECOG concordance for the simulated control group (Arm E).

I'll extract the required survival metrics, compute alignment with Flatiron benchmarks across specified timepoints, evaluate ECOG distribution, and determine pass/fail status for key metrics. I'll get back to you shortly with the full set of tables and interpretations.

# Validation Report

## Table T1 – OS Concordance

Month (mo)	Simulated OS %	Flatiron OS %	Absolute Difference %
0	100.0%	100.0%	0.0%
3	70.8%	70.0%	0.8%
6	50.3%	52.0%	1.7%
9	35.5%	40.0%	4.5%
12	25.2%	28.0%	2.8%
18	12.8%	15.0%	2.2%
24	6.3%	8.0%	1.7%

\*Example Calculation for Table T1:\* C4 (Absolute Difference %): For each row, calculate |C2 – C3|. For R2 (Month 3): |70.8 – 70.0| = 0.8%. (The values in this column are used to calculate the SD in Table T2.)

## Table T2 – OS Summary Metrics

Metric	Sim Value	Flatiron Value	Validation Note
Mean OS % (months 3–24)	33.5%	35.5%	Pass
SD of monthly absolute differences	1.4%	0.0%	Pass
Pearson r (Sim vs. Flatiron OS)	0.999	1.000	Pass

\*Example Calculations for Table T2:\*

\* \*\*R1 (Sim Mean OS%)\*\* –  $(70.8\% + 50.3\% + \dots + 6.3\%) / 6 = 33.5\%$ .

\* \*\*R2 (Sim SD of differences)\*\* –  $SD(0.0\%, 0.8\%, \dots, 1.7\%) = 1.4\%$ .

\* \*\*R3 (Pearson r)\*\* – Correlation between Sim and Flatiron OS% vectors = 0.999.

\*\*Validation Note:\*\* R1 difference =  $|33.5 - 35.5| = 2.0\%$  ( $\leq 5\% \rightarrow$  Pass); R2 difference =  $|1.4 - 0.0| = 1.4\%$  ( $\leq 5\% \rightarrow$  Pass); R3 = 0.999 ( $\geq 0.950 \rightarrow$  Pass).

## ## Table T3 – ECOG Concordance

ECOG State	Sim %	Flatiron %	Absolute Difference %
ECOG 0	20.2%	15.0%	5.2%
ECOG 1	73.7%	60.0%	13.7%
ECOG 2	6.0%	25.0%	19.0%

\*Example Calculation for Table T3:\* C4 (Absolute Difference %): For each row, calculate  $|C2 - C3|$ . For R2 (ECOG 1):  $|73.7 - 60.0| = 13.7\%$ .

## ## Short Interpretation

Simulated OS outcomes closely match Flatiron benchmarks. All simulated OS time-points fall within  $\pm 5\%$  of reference values. Summary metrics meet validation criteria: mean OS% (33.5% vs 35.5%) and SD of differences (1.4% vs 0.0%) are within 5% tolerance, and Pearson  $r = 0.999$  exceeds 0.950. However, baseline ECOG distribution shows large discrepancies: simulated ECOG 0/1/2 rates differ by 5.2%, 13.7%, and 19.0%, exceeding the  $\pm 5\%$  threshold. Thus, OS endpoints validation passes, but baseline ECOG concordance fails. Overall, the simulation shows strong OS outcome alignment with external data but fails to reproduce the reference ECOG profile, limiting its external validity.



## **Research Summary Generation 38b: Words**

Validation Report for Patient-Level Simulation Using Flatiron Benchmarks

ChatGPT

I gathered OS estimates and baseline ECOG distributions for months 0, 3, 6, 9, 12, 18, and 24, along with median OS of 5.6 months. This forms the baseline for the validation report.

I assessed the baseline ECOG performance status distributions in the S37.TRL.14.P30.LOG.csv file, focusing on the control group "Arm E" to ensure alignment with the reference values.

Searched for S37.TRL.14.P30.LOG.csv

ChatGPT

OK, let me see. Using the Python tool to explore the CSV file due to its large size or indexing issues.

Analyzed with python

ChatGPT

I'm working through the patient's baseline attributes and exploring columns like "time\_to\_death" and "os\_event\_flag" to align with relevant data.

I'm zeroing in on distinguishing OS and PFS columns. "time\_to\_death" likely marks OS time, while "time\_to\_progression\_or\_death" might denote PFS or an OS event flag.

Analyzed with python

ChatGPT

I'm working through the column count to gauge potential output length.

Analyzed with python

ChatGPT

I'm thinking about adjusting the CSV read settings to manage memory better and checking file integrity. Progressing through these steps to ensure smooth data handling.

Analyzed with python

ChatGPT

I'm thinking through the potential meaning of "time\_to\_death" as an event flag. It seems to record actual death times, possibly with a capped value for living patients.

Analyzed with python

ChatGPT

I'm making sure the output prints to screen.

Analyzed with python

ChatGPT

I'm thinking a simpler dataset might help. Also, checking for missing values in a specific column could shed light on the issue.

Analyzed with python

ChatGPT

Piecing together output issues, considering if `isnull().any()` works, and suspecting potential environment printing constraints.

Analyzed with python

Analyzed with python

Analyzed with python

ChatGPT

I'm exploring whether the absence of output signals a computation issue or a glitch in capturing results. Experimenting with processing smaller samples or examining specific conditions is also on my radar.

Analyzed with python

Analyzed with python

ChatGPT

OK, let me see. By narrowing down to "Arm E," it's easier to manage and understand. This approach refines the focus.

Analyzed with python

ChatGPT

I'm piecing together OS survival percentages for months 0, 3, 6, 9, 12, 18, 24 for Arm E, analyzing control patient data with lifelines. Progressing through data organization and analysis efficiently.

Analyzed with python

ChatGPT

I'm thinking through a work-around for web accessing issues by manually coding a survival function and differentiating survival events based on time-to-death data, especially assessing events at or beyond 24 months.

Analyzed with python

Analyzed with python

ChatGPT

I'm considering how large integer columns might need more effort to manage in data analysis.

Analyzed with python

ChatGPT

I'm exploring if writing to a small text file and accessing it via a browser can display results, considering the browser tool's access to user files.

Searched for Arm E S37.TRL.14.P30.LOG

ChatGPT

Reflecting on the decision.

Considering if a swift write to a file is necessary. More testing is needed to confirm accessibility and content update.

Searched for sim\_summary.txt

ChatGPT

I'm thinking it might be a delay in the file appearing or maybe I need to open it using a specific path.

I'm refining survival probabilities through the Kaplan-Meier method, noting how censored data alters initial risk counts.

Progressing step-by-step, I'm focusing on specific time points to ensure precision and consistency.

Analyzed with python

ChatGPT

I'm starting to gather KM values into a variable, ensuring they can be outputted as text. This organization is key for clarity and accessibility.

Searched for OS\_estimates.txt

ChatGPT

Piecing together various approaches like vectorizing or using numpy, and considering events/censor ratio for efficient survival calculation.

Analyzed with python

ChatGPT

I'm working through managing a large result list by querying specific times of interest. This ensures more focused and manageable data handling.

Analyzed with python

Searched for survival at 0 months

ChatGPT

OK, let me see. It seems the environment might be limiting access to local file content, particularly when trying to read user files through search.

I'm thinking about tweaking the search by adding specific terms like "24 months:" to refine results. This focuses the search on the relevant timeframe, which might improve precision.

Searched for survival at 24 months

ChatGPT

I'm working through the possibility that the environment might block direct file content reading, so I'll run the analysis locally and manually incorporate the results into the markdown output, ensuring accuracy with a thorough double-check.

Searched for OS estimates at 3 months

Analyzed with python

ChatGPT

I'm wondering if default behavior suppresses outputs to manage clarity.

Analyzed with python

ChatGPT

I'm thinking about potential reasons for no output and considering troubleshooting methods like using smaller data sets or confirming the python environment's interactivity in the web UI context.

Analyzed with python

ChatGPT

I'm thinking about the median overall survival (OS) from simulation results. Comparing simulated OS with reference values around 5.6 months to identify significant deviations.

Analyzed with python

ChatGPT

I'm thinking about median survival at 5-6 months using Flatiron data, where 12 months had 28% survival. Linear interpolation and analyzing event times are key.

Analyzed with python

ChatGPT

OK, let me see. Considering environment limitations and alternative ways to manage large outputs, like writing to files or breaking down content into smaller pieces, to ensure efficient handling.

Searched for UniqueOutput

ChatGPT

I'm thinking through the user\_files search limitations and considering direct access or trust in calculations. Bouncing back and forth on verifying numeric output.

Analyzed with python

ChatGPT

I'm considering whether the lack of output is due to a policy against showing raw results.

Searched for Arm E

Searched for Arm E

Analyzed with python

ChatGPT

Hmm, I'm thinking through a clinical trial, maybe five arms. Equal sizes are likely. Small data? Trying smaller libraries might help.

Analyzed with python

ChatGPT

I'm piecing together survival and progression times for patients in arm E, noticing significant variability and suggesting a larger sample for more reliable results.

Analyzed with python

ChatGPT

Mapping out survival percentages using Kaplan-Meier (KM) method, ensuring all patients eligible for 24 months are included, to reflect accurate survival rates.

Searched for 0

ChatGPT

Trying another way to search for "0:" or "0:."; still no progress.

Searched for 24:8.0

ChatGPT

OK, let me see. The user provided data on second-line PDAC's survival curve, and I'm curious if the simulation's 24-month OS aligns with this. I'm examining potential curves and calculating with numpy.

Analyzed with python

ChatGPT

I'm verifying if the environment is suppressing print statements and ensuring the calculations are accurate.

I'm noting that writing to a new file might hinder optimal search, so I suggest using unique snippets and chunking output lines to enhance search accuracy.

I'm mapping out survival rates at various points, comparing simulation to platform data, checking for close alignment with a median of 5.6 months.

Analyzed with python

ChatGPT

I'm reading the entire dataset to filter and analyze data for Arm E, ensuring manageable memory usage. Using pandas, I aim to efficiently compute differences and correlations with survival probabilities.

Searched for Sim OS% by months

Searched for Sim mean OS 3-24

ChatGPT

I'm looking into re-indexing sorted times to calculate the percentile and number of events. Using numpy's searchsorted, sorted times will streamline the process.

Analyzed with python

ChatGPT

I'm thinking about narrowing down to a smaller subset to observe survival patterns by printing samples of times.

Searched for percentile

Searched for 50th percentile

ChatGPT

OK, let me see. Verifying survival rates with manual counts for accuracy. Assuming subtle shifts in simulated data are acceptable. Progress is incremental, ensuring precision throughout.

Analyzed with python

ChatGPT

I'm digging into whether the file is actually being created, despite the lack of output.

Searched for 0:100.0

ChatGPT

I'm pulling together the thought process to analyze Pearson correlation and differences in survival probabilities between simulated and observed data, ensuring they meet the specified thresholds.

Searched for .0

Searched for empty query

ChatGPT

Hmm, I'm piecing together whether to read values out loud or use the `python` tool. Both methods have their nuts and bolts, so I'll weigh them thoughtfully.

Analyzed with python

ChatGPT

I'm piecing together potential logical errors in the code, like the handling of floating-point lists.

Analyzed with python

ChatGPT

Hmm, I'm thinking about why the environment might not be providing output. It could be due to size limits, or perhaps a specific command is needed to `display` the result.

Analyzed with python

ChatGPT

Evaluating the impact of reduced visibility on day-to-day activities.

I'm thinking about loading the entire file into memory if it's not too large, which might handle the data in the DataFrame.

Analyzed with python

Searched for 3:

Analyzed with python

ChatGPT

I'm building Markdown tables with accurate formatting, including headings and captions. Each table will have an accompanying example calculation for clarity.

Analyzed with python

ChatGPT

OK, let me see. After checking the data, 14153 out of 20000 survivors at 3 months gives about 70.8%, aligning well with the flatiron estimate.

Analyzed with python

ChatGPT

OK, let me see. Comparing the simulation's 50.3% to Flatiron's 52%, the simulation falls slightly behind by just 1.7%.

Analyzed with python

ChatGPT

OK, let's see. The survival rate from two groups, 35.5% and 40%, actually suggests a slight underestimation by 4.5% in the simulation.

Analyzed with python

ChatGPT

I'm piecing together 5046 survivors out of 20,000, giving a 25.2% survival rate. Flatiron's rate is 28%, so the discrepancy is around 2.8%.

Analyzed with python

Analyzed with python

ChatGPT

I found that 2554 survivors out of 20000 survivors gives approximately 12.8%.

I'm comparing simulated and flatiron survival rates over time, ensuring differences are within 5%. Progressing through time points to identify trends.

Analyzed with python

ChatGPT

OK, let me see. I'm looking at Pearson r values to spot correlations between different groups and also checking ECOG Concordance distribution in Arm E for baseline data.

Analyzed with python

ChatGPT

OK, let's analyze the ECOG distribution in the simulation. The values are a bit off, especially at thresholds. The OS outcomes look good though.

I'm making Markdown tables with one-decimal percentages and absolute differences. Progress is steady, and ethical guidelines are a priority.

Research completed in 15m · 0 sources

**Citations Generation 38b**

All Sources