A PURE Study of Jamovi for One-Way ANOVA

Dissecting the Pufferfish

LI Kanyu

17.03.2026

A PURE Study of Jamovi for One-Way ANOVA

Dissecting the Pufferfish

LI Kanyu

TL; DR

A prior usability test revealed a core paradox: users performing ANOVA in Jamovi perceived low task load but had a high error rate. This study used expert evaluation to pinpoint the root causes in the interface design. Three UX experts applied the PURE method to score 14 steps and conducted a Heuristic Evaluation to identify key user friction.

The experts found Jamovi's overall usability to be good. The total PURE score was 19/42 ("mostly easy"), and the average heuristic severity was 1.67 (minor issues). With 10 of 14 steps rated as easy, Jamovi's basic experience framework is sound.

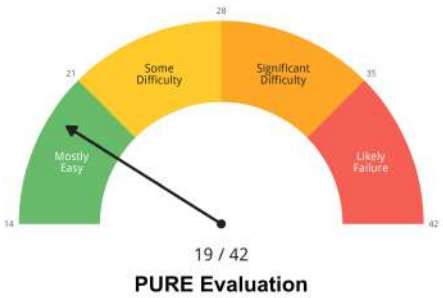
The cognitive bottleneck is concentrated in the diagnostic phase. The few high-load steps all required user inference without system feedback. Notably, "identifying data type errors" was the only step to score a 3 (likely to fail), as types are distinguished only by a minuscule icon.

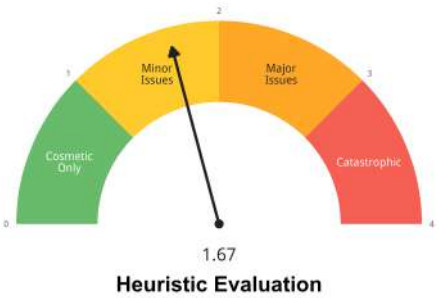
Crucially, the system is weakest in diagnosis and help. The heuristics for "Help and Documentation" (9/12) and "Error Recovery" (8/12) were the most violated. When a user fails, the system's only response is a flashing icon with no explanation.

We discovered a clear error propagation effect. An unobservable cognitive failure, like not identifying a data type, led to repeated operational failures. Consequently, 29 of 70 user errors occurred during the simple "dragging variables" step, caused by this upstream oversight.

The immediate priority is to provide contextual diagnosis and instant help. When a drag-and-drop fails due to a data mismatch, an explicit message explaining "what, why, and how" should replace the flashing icon. A top-level search bar should also be integrated for direct guidance.

Jamovi excels in surface-level design, but its core gap is its silence when users get stuck. Mending the cognitive barrier of data type identification with real-time support is the key to reducing downstream errors.





Drafted Jan 18, 2026

Contents

Research Objectives 3
Research Method 3
Analytical Framework 6
Key Insights 7
Actions 17
Epilogue 22
References 22
Supplemental Materials 22
Procedural Materials 30
Facility Guide 33

Correspondence to
LI Kanyu
im@not.ci

Competing Interests
The authors declare no competing interests.

1. Research Objectives

This study aims to build upon the findings of the first usability test by further dissecting the internal causes of cognitive friction users experience when performing a one-way ANOVA in Jamovi. The first study revealed a significant contradiction between "low subjective task load" and "high objective error rate," identifying "lack of feedback" and "cognitive errors (MISTAKE)" as key contributing factors. However, the specific cognitive barriers behind these behavioral patterns, and the precise interface design elements that cause them, remain to be clarified.

Therefore, the core objective of this research is to systematically deconstruct and locate the interface design roots of user cognitive friction. Specifically, this study will achieve the following sub-objectives:

1. **Quantify Cognitive Load at the Task Level:** Employ a step-by-step cognitive walkthrough (PURE) with experts to score the key steps from data import to ANOVA execution. This will identify which specific parts of the established workflow impose the highest cognitive load on users, thereby pinpointing the exact locations of friction.
2. **Diagnose Design Flaws at the Heuristic Level:** Utilize Nielsen's ten usability heuristics to conduct a comprehensive evaluation of the Jamovi interface. The goal is to identify which specific violations of these principles make the most significant contributions to the observed usability issues.
3. **Synthesize Multi-Source Data for Attribution:** Integrate PURE scores (high cognitive load steps) with heuristic evaluation findings (specific design flaws), and combine them with expert roundtable discussions. This will help construct an explanatory model that connects "interface design elements" to "cognitive barriers" and then to "user behavioral errors." Ultimately, this will link the subjective experiences and objective behaviors from the first study with the diagnostic analysis from this second study, providing precise and actionable recommendations for optimizing Jamovi.

2. Research Method

To achieve the above objectives, this study employed a two-phase expert evaluation method that combines PURE (Pragmatic Usability Rating by Experts) and Heuristic Evaluation, followed by a final roundtable discussion for synthesis.

2.a. Evaluation Experts

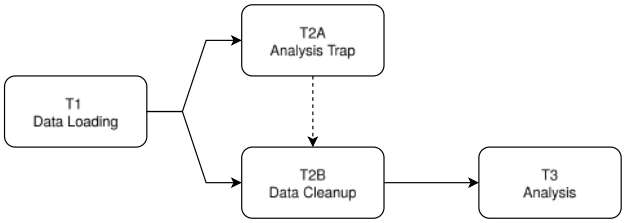
We invited three graduate students specializing in User Experience (UX) Design to serve as our evaluation experts. All three experts shared the following qualifications:

1. **Interdisciplinary Backgrounds:** They held undergraduate degrees in social science fields such as psychology and economics. This gave them a foundational understanding of statistical concepts, including ANOVA, enabling them to assess the software from both a user's and a researcher's perspective.
2. **Professional Knowledge:** All had systematically studied UX design theory, were familiar with Nielsen's heuristics, and had been trained in cognitive walkthrough methodologies, equipping them to identify and diagnose interface issues.

It is noteworthy that none of the three experts had prior experience with Jamovi; this was their first encounter with the software.

2.b. Evaluation Materials and Tools

1. **Target Software and Task Flow:** The evaluation focused on the open-source statistical software Jamovi, specifically its core workflow for conducting a one-way ANOVA.
2. **Task Script:** Based on the tasks from the first study, the complete analysis process was broken down into structured operational steps corresponding to three core phases:
    - T1 Data Loading: Import a CSV file.
    - T2A Data Trap: Simulate a user attempting analysis with uncorrected data format errors to expose the system's error-prevention mechanisms.
    - T2B Data Cleaning: Identify and handle an outlier (200) and a formatting error (3a).
    - T3 ANOVA Analysis: Correctly configure variables (dependent, grouping) and execute the analysis.



1. **PURE Rating Scale:** For each atomic step in the task script (e.g., "drag 'cheese_type' into the grouping variables box"), a three-point Likert scale was used for experts to rate the cognitive load on the target user (a novice with some statistical background):
    - 1 point: Low cognitive load, the target user can complete it with ease.
    - 2 points: Medium cognitive load, the target user can complete it with effort.
    - 3 points: High cognitive load, the target user may fail or abandon the task at this step.
2. **Heuristic Evaluation Questionnaire:** Based on Nielsen's ten usability heuristics, a five-point severity rating scale was designed for each principle:
    - 0 points: Not a usability problem at all.
    - 1 point: Cosmetic problem only, no need to fix.
    - 2 points: Minor usability problem, low priority to fix.
    - 3 points: Major usability problem, high priority to fix.
    - 4 points: Usability catastrophe, imperative to fix before release.

2.c. Evaluation Process

The evaluation was conducted in a single, continuous session, divided into three phases:

1. **Phase 1: PURE-based Cognitive Walkthrough and Rating**
    - The moderator presented standardized screenshots of the task interface to the experts (as shown in Jamovi_PURE_Research.docx.pdf).

- Experts simulated the mental state of the target user (a student or researcher with a statistical background but no Jamovi experience) while performing each step of the task script.
- After completing each step, experts independently assigned a PURE score to record the anticipated cognitive load.
- This phase was designed to identify the "most strenuous" steps in the task flow.

2. **Phase 2: Heuristic Evaluation and Rating**
    - After completing the cognitive walkthrough of all tasks, experts formed an overall impression of the Jamovi interface, particularly the parts related to the one-way ANOVA.
    - They then independently completed the heuristic evaluation questionnaire.
    - For each Nielsen heuristic, experts provided a severity rating based on their observations of interface interactions during the walkthrough and could add brief descriptions of the issues.
    - This phase aimed to connect specific points of friction to universal, theoretical design principles.
3. **Phase 3: Roundtable Discussion and Consensus Building**
    - After all experts completed their independent ratings, the moderator facilitated a roundtable discussion.
    - The core of the discussion was to synthesize the findings from the PURE and heuristic evaluations. Experts shared the reasoning behind their scores, debated any disagreements, and ultimately reached a consensus on the most critical design flaws and user friction points.
    - The qualitative insights from this phase were recorded and transcribed to supplement and provide depth to the quantitative ratings.

2.d. Bridging the Two Coding Systems

The cross-analysis in Section 5 maps behavioral events from Study 1 onto the PURE task structure from Study 2. Because the two studies were designed independently, their task systems required post-hoc alignment. This section describes the alignment procedure; the upstream event coding process is documented in Supplementary Material.

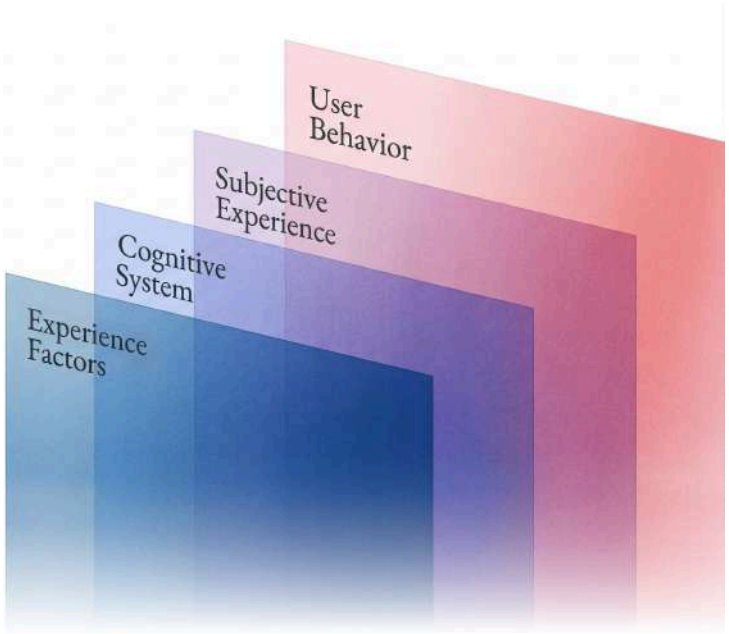
Each event in Study 1 was assigned to either the Happy Path or the Trap condition based on a single observable criterion: whether the participant had, by that point in the session, opened the variable editor and corrected the variable type. All events before this corrective action were classified as Trap. If a participant never performed this correction (as with P04), their entire session remained in the Trap condition.

For the cross-analysis, we used the PURE happy path as the reference frame and mapped each atomic event from Study 1 onto the corresponding PURE sub-step. Because Study 1's coding was sufficiently fine-grained (each code captured a single interface action), no event required splitting across multiple PURE steps. Events that could not be mapped onto any PURE sub-step were collectively categorized as Off Happy Path. This category is a structural residual of the alignment: the PURE framework models a predefined sequence of cognitive and operational nodes, but does not cover the exploratory, orienting, or recovery behaviors that users engage in between those nodes. The 36% of events (and 49% of session time) falling into this category

therefore reflects a known difference in coverage between the two analytical layers, rather than a failure of the mapping procedure.

3. Analytical Framework

Before presenting the specific results, it is necessary to establish an analytical framework that integrates the data from both rounds of research. The first-round usability test and the second-round expert evaluation produced different types of evidence. These are not mutually independent, parallel conclusions but rather point to different levels of the same user experience. To create a unified explanatory basis for the following sections, we adopt a four-layer analysis model.



Layer	Content Captured	Corresponding Method	Specific Data in This Study
User Behavior	Observable actions and errors.	Usability Testing: Event Coding	70 error events, task phase severity.
Subjective Experience	User's self-reported state.	Usability Testing: SUS & NASA-TLX	The "low perceived load" finding from the first study.
Cognitive System	Cognitive prerequisites for task completion.	PURE Cognitive Walkthrough	Ratings for 14 sub-steps.
Experiential Elements	Interface design elements that shape the cognitive environment.	Heuristic Evaluation	Severity ratings for 10 Nielsen principles.

The core logic of this model is a bottom-up causal direction:

Experiential Elements form the foundational layer. Specific interface design elements, such as icon legibility, the presence or absence of error feedback, and operational consistency, shape the user's cognitive environment. The heuristic evaluation operates at this layer, identifying specific flaws that violate design principles.

The **Cognitive System** sits above the experiential elements. It describes the cognitive prerequisites a user must meet at each operational step: what they need to understand, infer, and remember to proceed. The PURE cognitive walkthrough works at this layer, predicting the cognitive load of each step for the target user. Design flaws at the experiential layer directly impact whether these cognitive prerequisites can be met. For example, if an icon's affordance is too subtle (an experiential flaw), the cognitive inference required to identify a data type error (a prerequisite at the cognitive layer) becomes difficult to complete.

Subjective Experience is located above the cognitive system. It reflects the user's meta-cognitive assessment of their own state: their perceived task difficulty, confidence level, and frustration. This layer is modulated by the cognitive system. When an interface fails to provide adequate feedback signals, users may be unable to accurately assess their actual predicament, leading to a disconnect between subjective perception and objective performance.

User Behavior is the top-most and only directly observable layer. The user's actual operations, the errors they make, and the recovery strategies they employ are all recorded here. This is what behavioral event coding captures. However, behavior itself does not explain its cause; it is the manifest outcome of the combined effects of the three underlying layers.

This framework provides a unified lens for interpreting the results in the following sections. PURE scores will pinpoint the sources of difficulty at the cognitive system layer. Heuristic evaluation results will identify design flaws at the experiential elements layer. The cross-analysis will connect the cognitive system and user behavior layers, revealing how cognitive difficulties predicted by experts manifest in the actions of real users. The core paradox from the first study, low perceived task load versus high objective error rate, can thus be explained through the relationship between the subjective experience and user behavior layers.

4. Key Insights

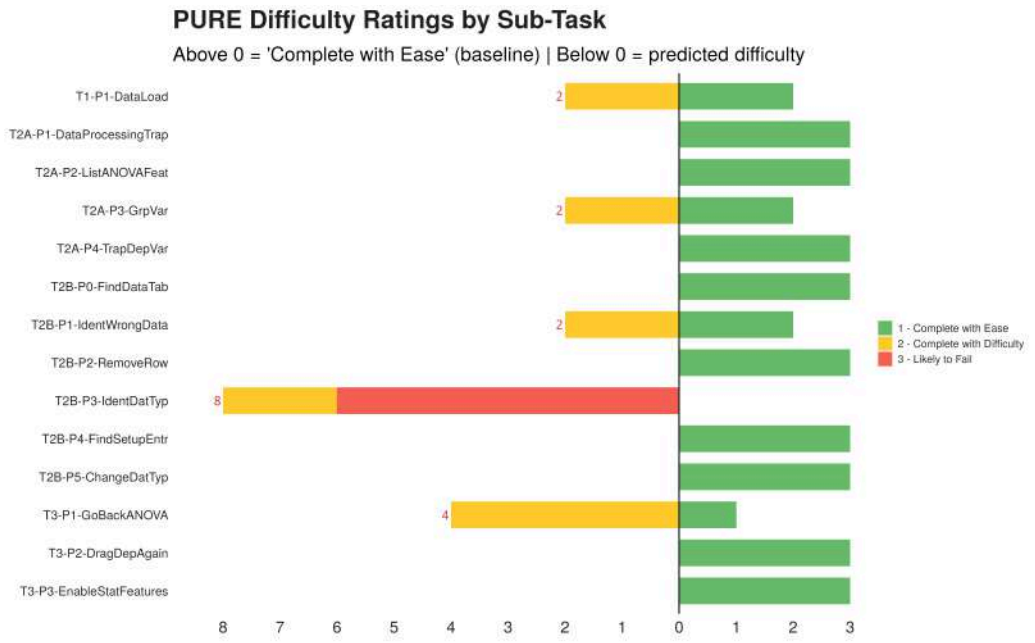
4.a. PURE Evaluation

The Vast Majority of Steps Are Easy, with a Cognitive Bottleneck in a Few Steps Requiring Self-Diagnosis.

The total PURE score was 19/42, placing it at the high end of the "Mostly Easy" range. Among the 14 "Happy Path" sub-steps evaluated, 10 received a unanimous score of 1 ("can be completed with ease") from all three experts. These steps included data import, opening the analysis menu, selecting methods, deleting rows, changing variable types, and dragging variables into the dependent variable box. In other words, Jamovi's overall interaction architecture, including its menu structure, drag-and-drop

interface, and parameter configuration, is smooth for target users with a basic statistical background.

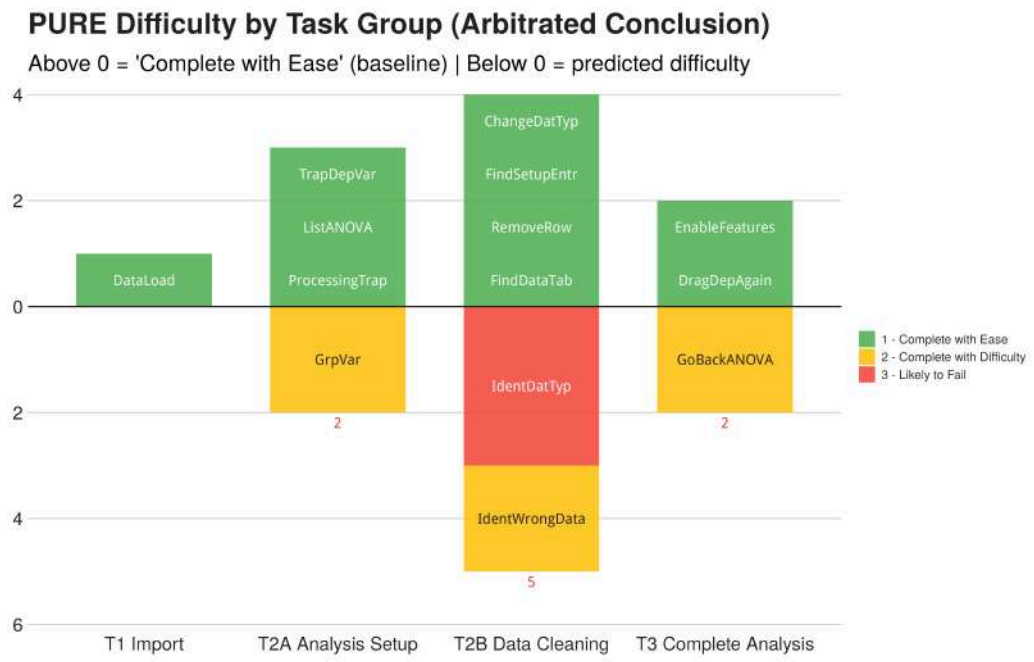
However, the scores for the remaining steps deviated significantly from this baseline. They all shared a common characteristic: they required the user to independently perform a cognitive inference with little to no clear system feedback.



T2B-P3: Identifying data type errors (consensus score of 3, the highest of all steps) was the only step to receive a "likely to fail" rating. This step required the user to infer that the `nightmare_frequency` column was incorrectly classified as text instead of a continuous variable based on a minuscule icon next to the column header. Expert S highlighted the icon's lack of legibility: *"I can only see that the type for this column and the second column are the same… It's very difficult for a user to understand this."* Expert A was more direct: *"If you hadn't told me, I would have never known that was an icon… I thought it was just for aesthetics."* The difficulty of this step did not stem from operational complexity, as no action was required. Instead, it came from a cognitive judgment that lacked system support. The user had to notice a subtle visual difference, understand its meaning, and from that, deduce the cause of subsequent operational failures.

T3-P1: Returning to the ANOVA analysis interface (consensus score of 2) involved a discoverability issue. The user needed to click on the previously created, empty ANOVA entry in the right-side panel to return to the configuration interface, but this entry did not visually appear to be an interactive element. Expert A said, *"That box doesn't look clickable, it looks more like a display window."* Expert F added to the ambiguity: *"There are two one-way ANOVA options, and I don't know which one is the one I should be using."*

T2B-P1: Identifying incorrect data (consensus score of 2) and **T2A-P3: Dragging a variable into the analysis slot** (consensus score of 2) presented difficulties stemming from data scalability and a lack of sequential guidance in the workflow, which will not be elaborated on here.



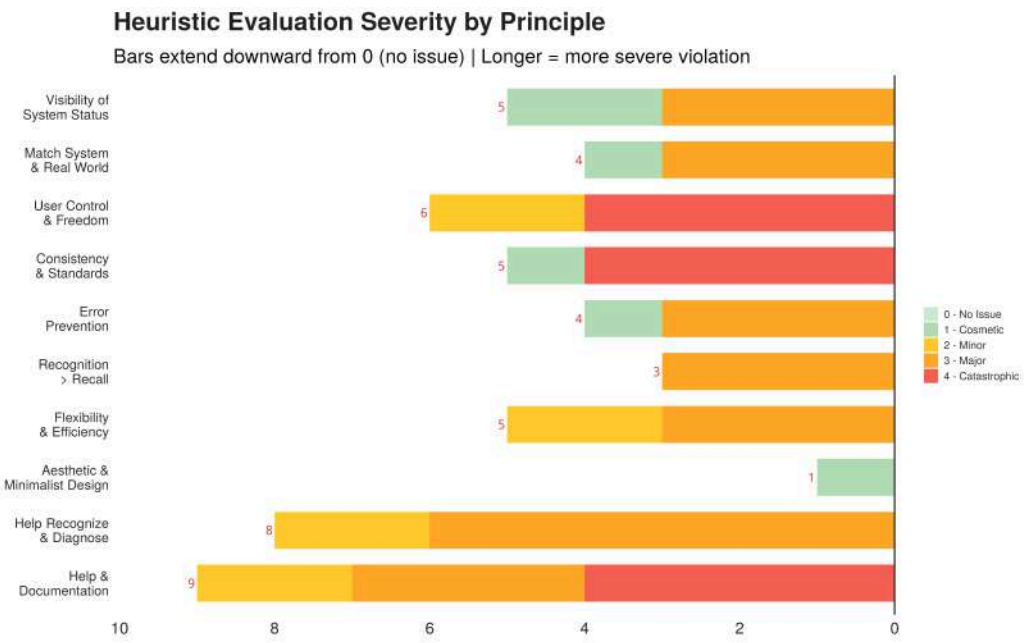
From a task-group perspective, the distribution of these four high-difficulty steps explains why T2B (Data Cleaning) was the largest source of cognitive load (accumulating 5 points above the baseline). This task group contained the two highest-scoring steps (IdentDatTyp and IdentWrongData), which are also cognitive prerequisites for all downstream analysis steps. T2A (Analysis Configuration) and T3 (Completing Analysis) each accumulated 2 difficulty points, while T1 (Data Import) had none.

Regarding the arbitration for T1-P1 (Data Loading), Expert S initially rated this step a 2, because an Apple Numbers file could not be directly imported into Jamovi. In the roundtable discussion, the three experts analyzed the root of this difficulty and concluded it was a matter of file format support rather than a flaw in the import process itself. Consequently, the consensus score was arbitrated to 1. The functional suggestion from Expert S, “Many Mac users use Numbers instead of Excel,” was recorded separately.

4.b. Heuristic Evaluation

Generally speaking, the System Is Weakest in Diagnosis and Help.

If the PURE evaluation answers the question, “Where will users encounter difficulty?”, then the heuristic evaluation answers, “What system capabilities are lacking that cause this difficulty?” The two frameworks converge strongly at the diagnostic level.



The heuristic scores, aggregated from the three experts (with each expert's rating ranging from 0 to 4), clearly fall into three tiers:

The most severe gaps are in diagnosis and help capabilities. H10: Help and Documentation (total score of 9) and H09: Help users recognize, diagnose, and recover from errors (total score of 8) ranked first and second, respectively. The issues corresponding to these two principles are precisely the system capabilities that the high-difficulty steps from the PURE evaluation rely upon. The reason "IdentDatTyp" is difficult is not because the "ChangeDatTyp" operation itself is complex (its consensus score was only 1), but because before the user can even get to that operation, they must first independently complete the cognitive step of "diagnosing that the data type is wrong." The system's deficiencies in H09 and H10 mean that the user receives no assistance at this critical juncture.

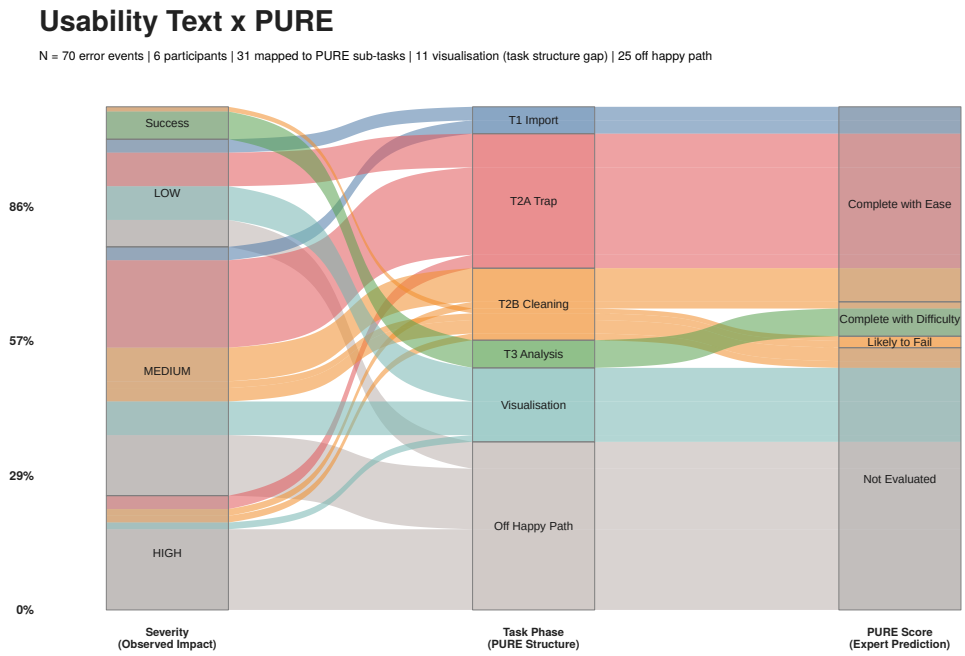
Expert F gave H10 a score of 4 (catastrophic), and their comment was incisive: *"When an error happens and you don't know what to do, usually there's some option that says help or a search bar. I don't see nothing like this happening here."* Expert A gave H09 a score of 3 and added a constructive suggestion: *"There could be optional tutorial pop-ups… or at least just an explanation of why the operation failed."* The expert panel's consensus was that while Jamovi has online documentation, it lacks contextual, in-app, on-demand help, which is precisely what users need most when they are stuck.

The second tier of issues relates to user control and interaction conventions. H03: User control and freedom (total score of 6), H01: Visibility of system status (5), H04: Consistency and standards (5), and H07: Flexibility and efficiency of use (5) formed a cluster of moderately violated principles. The disagreement over H03 is worth noting: Expert F, focusing on layout, gave it a 4, questioning why "almost half the screen is blank space… why is it designed this way?" In contrast, Expert A and Expert S gave it scores of 2 and 0, respectively. The disagreement on H04 was even more extreme. Expert F (4) consistently used Excel as a reference point for evaluating Jamovi, while

Expert S (0) used Stata, finding Jamovi's interactions perfectly consistent within the context of statistical software. This score disparity, ranging from 0 to 4 for the same interface, is itself a meaningful finding. It demonstrates that a user's mental model, shaped by which software they "migrate" from, significantly influences their judgment of a given design. This point will be specifically addressed in the later action-oriented recommendations.

The third tier reflects an appreciation for the visual design. H06: Recognition rather than recall (3) and H08: Aesthetic and minimalist design (1) received the lowest scores, indicating the expert panel's positive assessment of Jamovi's visual design. Expert F described it as "clean and minimalist" but also offered a suggestion connecting the visual layer to the diagnostic layer: "When an error occurs, it should be highlighted with a color like red," once again pointing to the same core deficiency.

4.c. Convergence of the Two Frameworks: Operational Simplicity Does Not Equal Cognitive Simplicity



When the PURE and heuristic evaluations are viewed together, a coherent picture emerges. Jamovi is successful in its shallow-level user experience design. The low difficulty scores for 10 of the 14 steps and the low violation score for H08 (Aesthetic and minimalist design) both confirm this. All three experts expressed a willingness to use Jamovi again, with Expert A calling it "quite intuitive," and Expert S noting it was "much better than Stata… much clearer."

However, operational simplicity does not equate to cognitive simplicity. When the core difficulty of a step in the workflow is not "how to do it" but "why it cannot be done" or "what should be done," the system offers no diagnostic feedback or contextual help. This is precisely the capability gap identified by H09 and H10. A statement from Expert F near the end of the roundtable discussion accurately captured this problem: *"Without your guidance, I probably wouldn't be able to do anything here. I'd have to go study a tutorial first."*

This convergence establishes the foundation for the subsequent cross-analysis. The PURE evaluation located the source of cognitive load (which steps are the hardest), and the heuristic evaluation diagnosed the system's capability gap (why the hardest steps lack support). The next section will connect these two layers of analysis with the user behavior data from the first usability study, tracing how difficulties at the cognitive layer propagate downstream through the workflow, ultimately manifesting as observable errors at the behavioral layer.

4.d. Cross-Analysis with Usability Study Results

4.d.i. *Bridging the Two Types of Evidence:*

The expert evaluation in the previous section pinpointed two things: the source of cognitive load (the highest-rated PURE steps) and the system's capability gap (the most severely violated heuristic principles). However, these are predictions based on experts simulating a user's mental state. This section will systematically integrate these predictions with the 70 unique error events from 6 real users in the first usability study. It will answer a core question: how does the cognitive difficulty predicted by experts manifest in the behavior of actual users?

Returning to the four-layer analytical framework, the PURE and heuristic evaluations operate on the bottom two layers (Cognitive System and Experiential Elements), while the usability test captures the top two layers (User Behavior and Subjective Experience). This section focuses on the connection between the Cognitive System layer and the User Behavior layer, where the predicted cognitive difficulty meets the observed error patterns.

4.d.ii. *Constructing the Cross-Analysis:*

To operationalize this connection, we constructed a three-column alluvial diagram. The left column represents the observed severity of the error (LOW / MEDIUM / HIGH), the middle column represents the task phase in which the error occurred, and the right column represents the corresponding PURE expert difficulty score for that sub-task. The color of the ribbons represents the task category, from T1 to T3, plus an additional visualization task that fell outside the happy path.

The task phase in the middle column was extracted from the sub-task labels in the event encoding, allowing for a direct mapping to the PURE task structure. The distribution of the 70 error events is as follows:

1. **34 events (48%)** could be mapped to sub-tasks with PURE scores. These were distributed across Data Import (T1 Import, 4 events), the Analysis Setup Trap (T2A Trap, 20 events), and Data Cleaning (T2B Cleaning, 7 events).
2. **11 events (16%)** were related to attempts at data visualization. These are shown separately in the alluvial diagram, and their cause will be discussed later.
3. **25 events (36%)** were exploratory actions that deviated from the predefined path (Off Happy Path), concentrated in users' impromptu attempts during the analysis process.

No error events were recorded for the on-path analysis phase (T3 Analysis). This means that users who successfully navigated the initial hurdles and entered the correct

analysis workflow did not make mistakes in this phase. T3 is only visible in the alluvial diagram as a "Success" fill line.

By comparing the previous usability study with this one, we found a complete convergence of evidence:

Issue	Study 1 Evidence	Study 2 Evidence	Convergence Level
Ineffective Feedback	6/6 users did not understand the flashing icon.	H09 total score of 8, Expert A quoted.	Complete Convergence
Data Type Identification Barrier	P03/P04 never independently completed.	PURE's only 3-point step.	Complete Convergence
Lack of Contextual Help	P06 proactively suggested a hyperlink.	H10 total score of 9 (highest).	Complete Convergence

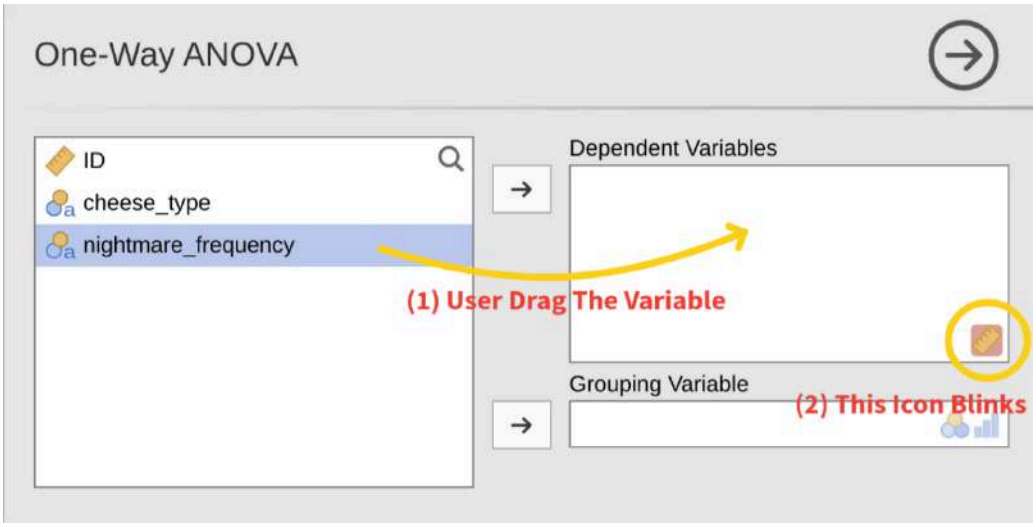
4.d.iii. *Core Finding:*

The Invisibility of Cognitive Prerequisites and the Visibility of Operational Failures.

After mapping the two data types, the most striking pattern is a structural difference between the two analytical methods. This difference reveals an interesting mechanism: errors at the cognitive layer continuously propagate, leading to a large volume of operational errors.

The 14 PURE sub-steps are not all observable interface actions. They are a mix of two distinct types of activities: **operations** (e.g., "drag a variable into the dependent variable box") and **cognitions** (e.g., "identify that the data type is wrong"). The event coding system records observable interface actions, meaning what the user clicked, dragged, or opened. In contrast, "identifying" is a cognitive inference process that occurs internally. It produces no interface action itself, and therefore, an event log will not and should not contain an entry called "user failed to identify data type."

This distinction directly explains the most prominent numerical distribution in the alluvial diagram. Of the 31 error events that mapped to PURE-scored steps, **29 occurred on steps that experts rated as 1 ("can be completed with ease")**. Only 1 error occurred on a PURE 2 step (T2B-P1-IdentWrongData, identifying incorrect data), and 1 on a PURE 3 step (T2B-P3-IdentDatTyp, identifying data type). The PURE 3 step, IdentDatTyp, received almost no error events not because it is actually easy, but because it is a cognitive activity, not an operation. By definition, event coding cannot capture its failure.



4.d.iv. *Three Trajectories of Propagation:*

1. **P05: Repeated Operational Failure.** P05 failed to drag the variable five consecutive times. During this period, they reported that the "ruler icon is flashing" but did not understand its meaning. They only discovered the type issue nearly 6 minutes later by entering the variable editing panel. From a behavioral record, all failures occurred during the "drag" operation. From a cognitive analysis, the real missing piece was the "identify data type error" inference. The latter is not an operation and will not appear in an event log, but its absence doomed every drag attempt to fail.
2. **P03: Spillover Beyond the Preset Path.** Repeated drag-and-drop failures prompted P03 to start an impromptu search for workarounds. They dragged the ID column as a dependent variable (a type match but statistically meaningless), then attempted a non-parametric ANOVA, then an independent samples t-test, and then another non-parametric ANOVA. Over the course of 5 minutes, they traversed multiple analysis tools. All these blind attempts pointed to the same unfulfilled cognitive activity: P03 never independently identified the data type. They eventually gave up on the task and only proceeded after the researcher provided a direct hint.
3. **P04: Silent Task Failure.** P04 also never completed the type identification but did not engage in repeated retries. Instead, they discovered a "workaround": they dragged the row ID (the only correctly formatted variable) into the dependent variable slot, produced a statistically meaningless analysis result, and confidently reported a conclusion based on it. The user did not "fail" but rather "succeeded in producing a wrong conclusion." This is a silent, non-recoverable task failure. Not only does event coding fail to record the "identification failure," but the user themselves was completely unaware that a prerequisite had not been met.

4.d.v. *Visualization Events: A Gap in Task Structure:*

The cause of the 11 visualization-related events is different from the propagation mechanism described above and requires separate discussion. The PURE task structure condensed visualization into a single step, T3-P3-EnableStatFeatures, which simply involves checking the "Descriptive plots" checkbox in the ANOVA panel. On the

surface, it is just one click. However, the user's mental model of "creating a data visualization" is often to "go to a dedicated charts tool." When users did not notice the visualization option within the ANOVA panel, they would look for it in the descriptive statistics tool, the scatterplot tool, the Pareto plot tool, and elsewhere.

These events had a relatively low average severity (1.64), with 5 of the 11 being LOW severity. Users who tried the wrong chart type usually realized the issue and switched. This was not a case of an operational error propagating from a missing cognitive prerequisite. Instead, it was a misalignment between the PURE task decomposition and the user's mental model. In the PURE study, we treated visualization as a sub-option of the analysis workflow; users treated it as an independent task phase.

4.d.vi. *Off-Happy-Path User Activity:*

Excluding the cleaning and visualization events, the remaining 25 Off Happy Path events are concentrated in users' impromptu explorations during the analysis process. They are the core product of the difference in observational granularity between the cognitive system layer and the user behavior layer.

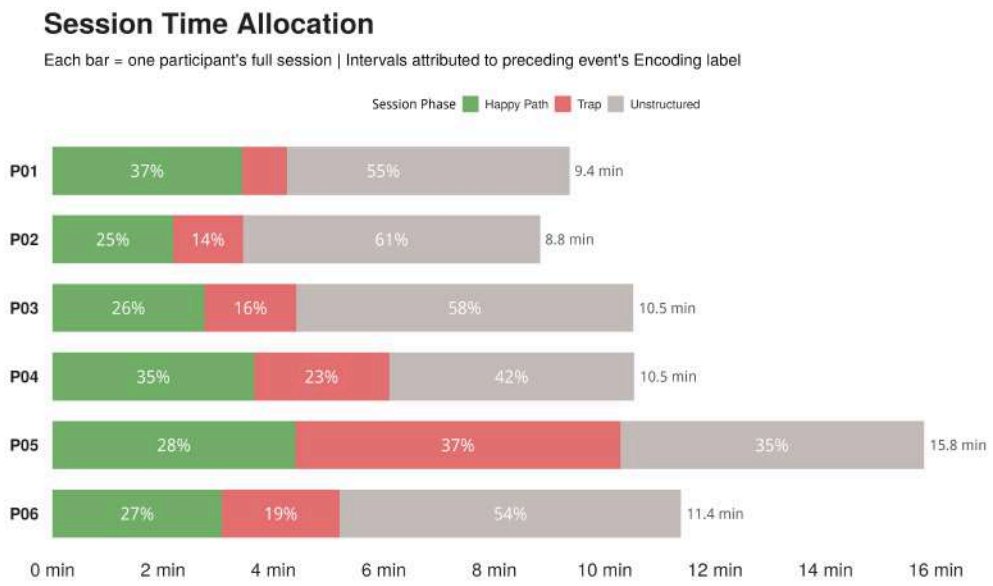
The average severity of these 25 events was 2.32, with 12 being HIGH, the highest of any category. P03's exploratory attempts across four different analysis tools (totaling 7 events, 5 of which were HIGH severity) and P04's entire workflow from incorrect configuration to copying the erroneous result into the data grid (totaling 5 events) fall into this category. When a critical cognitive activity is not completed, user behavior is not confined to the predefined operational steps. Instead, users react by exploring and deviating. These deviations, by definition, exceed the scope of any evaluation method based on task decomposition.

The 36% deviation rate is a structural feature of the PURE framework, not a flaw. It reflects the difference between the layer where PURE works (the cognitive system layer, evaluating a predefined sequence of cognitions and operations) and the layer where behavioral data is collected (the user behavior layer, recording only observable operations).

4.d.vii. *Time Allocation: The Gaps Between Steps:*

The alluvial diagram shows the distribution of error events, but event counts do not reflect the actual time users invested in each activity. To add this dimension, we conducted a time-based analysis of all 152 key user events (not just errors). The method was to attribute the time interval between any two adjacent events to the encoding label of the first event. We then aggregated the time into three categories:

1. **Happy Path:** T1 + T2B + T3, the predefined task steps.
2. **Trap:** T2A, operations under the incorrect data type condition.
3. **Unstructured:** Activities with no corresponding PURE sub-step.



For the 66.4 minutes of total session time from the 6 participants, **Happy Path accounted for 29% (19.5 min)**, **Trap accounted for 21% (14.2 min)**, and **Unstructured accounted for 49% (32.7 min)**. Nearly half of the session time was spent outside the 14 predefined PURE steps.

At the event level, 36% of recorded actions deviated from the Happy Path. At the time level, this rose to 49%. This discrepancy comes from activity types that the alluvial diagram could not capture. A content analysis of the 82 Unstructured events revealed the following categories: off-path visualization attempts (20 events), off-path analysis attempts (19), reviewing and understanding analysis output (14), interface exploration and discovery (7), recovery and management actions like restarting the software or changing the language (7), researcher interventions (4), and other (11). The first two categories correspond to the Off Happy Path and Visualisation errors in the alluvial diagram. The third category, "reviewing and understanding results," is an activity that PURE does not model at all. Users need time to read the output, confirm if the result is reasonable, and decide on the next step. This is not a "step," but it consumes a significant amount of time.

In summary, when we trace back each participant's actual completion of the "identify data type" cognitive activity, we find:

1. **P01:** Identified the type issue on their own, a true success.
2. **P02:** Identified it on their own, but only after over 5 minutes of operational failures.
3. **P03:** Never identified it on their own; the researcher had to provide a direct hint near the end of the task for the analysis to proceed.
4. **P04:** Never identified it, and bypassed the problem by using the row ID.
5. **P05:** Only entered the variable editing panel after nearly 6 minutes of operational failures.
6. **P06:** Accidentally triggered the variable metadata editor while trying to insert a column and stumbled upon the type issue by chance.

This discovery points to a structural blind spot in evaluation methods based on task decomposition. PURE models 14 discrete step nodes and the cognitive prerequisites at those nodes, but it does not model the gaps between the nodes. That is, the time users spend on orientation ("where am I?"), comprehension ("what did my last action produce?"), and decision-making ("what should I do next?"). The actual time allocation of users is far more complex than a sequential list of steps would suggest, with nearly half spent in these gaps. This aligns with the explanation from the four-layer framework: the step-level evaluation at the cognitive system layer cannot cover the entirety of activity that occurs between steps at the behavioral layer.

4.d.viii. *Severity Distribution:*

The overall severity distribution of the 70 error events was: Medium 37 (53%), High 17 (24%), and Low 16 (23%). Calculating the average severity by task phase (1=Low, 3=High) yields the following ranking:

Task Phase	Average Severity	HIGH	MEDIUM	LOW
Off Happy Path	2.32	12	9	4
T2B Cleaning	2.20	2	8	0
T2A Trap	1.85	2	13	5
Visualisation	1.64	1	5	5
T1 Import	1.50	0	2	2

Off Happy Path has the highest average severity (2.32), and it is concentrated with the most severe consequences, such as P03's task abandonment and P04's conclusion based on an invalid analysis. **T2B Cleaning** is the second highest (mean = 2.20). Data cleaning is a cognitive prerequisite for all downstream analysis, so failure at this stage means that all subsequent steps are built on a flawed foundation. **T2A Trap** contributed the most events (n=20), but its average severity is lower (1.85). Most of these were repeated drag-and-drop failures. While operationally frustrating, each individual attempt is recoverable. The danger lies in their cumulative effect: they consume time, erode confidence, and ultimately push the user off the Happy Path.

5. Actions

Study 1 (N=6 users, remote usability testing + behavioral coding) revealed the error patterns and cognitive traps in the actual use of Jamovi. Study 2 (N=3 experts, PURE task evaluation + heuristic evaluation + roundtable discussion) predicted task difficulty and assessed system-level design issues from an analytical perspective. The two studies were conducted independently, yet they converged on several key findings. This convergence itself constitutes a stronger body of evidence than either study could provide alone.

The following recommendations are prioritized. Each one is annotated with the supporting evidence from the studies. When the findings of both studies corroborate each other, we have higher confidence that the issue is systemic and not incidental.

5.a. How might we help users understand the cause of an error instantly, instead of through repeated trial and error?

Priority: High Impact / High Development Effort. This requires establishing an exception handling mechanism and providing a data exception handling framework for all plugins.

Evidence Convergence: This was the most frequently mentioned and highest-severity issue across both studies.

1. **Study 1:** All 6 users encountered the flashing icon feedback when dragging a variable into the dependent variable box, and none understood its meaning. P06 said, *"When I couldn't add it, I was very confused. What does the flashing even mean?"* P05 reported, *"The ruler icon is flashing,"* but did not know what had happened. Observational records show that users would repeat the same action 3-5 times without being able to diagnose the cause of failure.
2. **Study 2:** Expert A stated, *"If you hadn't said anything, I wouldn't know that the icon was even an icon… I just thought it was for aesthetic visuals."* Expert F pointed out that the system was "not responsive enough" and gave H09 (Help users recognize, diagnose, and recover from errors) a score of 3 (Major). The total score for H09 from all three experts was 8, the second-highest violation.

Behavioral coding data further revealed the cascading effect of this issue: of the 70 error events, 29 mapped to the PURE 1 ("Complete with Ease") step T2A-P4-TrapDepVar (dragging a variable to the dependent variable box). The cognitive root of these behavioral failures points directly to the PURE 3 ("Likely to Fail") step T2B-P3-IdentDatTyp (identifying data type). Users did not understand "why this variable cannot be dragged," and the system provided no diagnostic information at this cognitive juncture.

Recommendation: When a drag-and-drop operation is rejected due to a data type mismatch, replace the flashing icon with an explicit, contextual message. The message should contain three levels of information:

1. **What happened:** "The variable `nightmare_frequency` cannot be placed in the dependent variable box."
2. **Why it happened:** "The current data type for this variable is text, but the dependent variable requires a continuous data type."
3. **How to handle it:** "Either convert the data type or consult statistical principles for guidance."

5.b. How might we provide contextual guidance at the moment users need help most, instead of making them search on their own?

Priority: High Impact / High Development Effort. Requires a content indexing system and a search backend.

Evidence Convergence: H10 (Help and Documentation) received the highest total score of 9 in Study 2's heuristic evaluation, making it the most severely violated principle. User interviews from Study 1 independently pointed in the same direction.

1. **Study 1:** In an interview, P06 proactively suggested, *"Put a hyperlink at the bottom of the interface, and when you click it, you see a Wikipedia article."* P02 suggested the system should be able to do "simple keyword matching, for example, 'frequency'

should be identified as a continuous variable." P06 also proposed the concept of an "analysis wizard": *"It asks you questions step-by-step, and then it produces the data."*

2. **Study 2:** Expert F gave H10 a score of 4 (Catastrophic), commenting, *"When an error happens and you don't know what to do, usually there's some option that says help or a search bar. I don't see nothing like this happening here."* Expert A suggested *"tutorial pop-ups on the side that are optional… or even just help explanation of why it's not working."* Expert S believed *"the tutorial may need to be built in the software."*

Recommendation: Jamovi's interface already borrows heavily from Microsoft's design language, from the Ribbon-style function areas to the column layout. In the latest versions of Microsoft Office, the help function has been integrated into a top search bar ("Tell me what you want to do"), allowing users to directly search for functions using natural language to get operational guidance. Considering that Jamovi is already on this design path, introducing a search-based help entry is a natural extension. The specific form could be a search box in the top function area that supports full-text search of documentation and function names. Search results would link directly to the corresponding interface element or a help page. This is more aligned with the user's immediate need; they are not trying to "learn" the software, but to "solve a problem now."

5.c. How might we ensure data quality issues from the preparation phase are discovered before a user enters the analysis phase?

Priority: High Impact / High Development Effort. This is a long-term investment.

Evidence Convergence: The PURE evaluation in Study 2 identified T2B (Data Cleaning) as the most difficult task group (5 points above the baseline), with T2B-P3-IdentDatTyp being the only step to receive a "Likely to Fail" rating. The behavioral data from Study 1 confirmed this prediction: P03 and P04 never independently completed the data type identification. P03 relied on researcher intervention, while P04 bypassed the problem entirely.

Study 1 also uncovered a dangerous pattern of silent processing: when P01 attempted to force-convert a text column to a numeric format, the system automatically converted "3a" to another value without any warning. The researcher later explained, *"You forced the format change, so it turned '3a' into something else, and that data point was never cleaned."* P02 suggested, *"There should be a more obvious warning here."*

Recommendation: Study 1's report proposed the concept of a "data linter," similar to a syntax checker in a code editor but for statistical analysis configurations. This direction received indirect support from Expert S in Study 2, who noted that the system did not highlight the anomalous data in rows 10 and 60. A concrete implementation could include:

1. When a column contains over 95% of values that can be parsed as numeric but also a small number of non-numeric characters, the system proactively prompts, "This column may contain data entry errors."
2. During a data type conversion, display a change preview ("The following N values will be converted: '3a' → NA") and require user confirmation instead of silent execution.

3. When a user switches from the data tab to the analysis tab, if unresolved data quality issues exist (such as mixed-type columns), display a non-blocking notification.

5.d. How might we improve the information architecture of the results panel, allowing users to both freely manage data and structurally manage analysis results?

Priority: Medium Impact / Medium Development Effort. Requires fine-tuning the interface layout and embedding a retrieval mechanism.

Evidence Convergence: Study 1 recorded several independent issues related to the results panel. Study 2's expert evaluation pointed to the same structural flaw from a different angle.

1. **Study 1:** P06, each time they clicked "One-Way ANOVA" in the menu, created a new analysis item, but they thought they were editing the existing one: *"I thought after I clicked it, it would be linked to the one above, not that I was creating a new one."* P04, when trying to delete a single analysis result, right-clicked on an outer area, which the system interpreted as selecting the entire results panel, causing all content to be deleted. When an analysis configuration is incomplete, the results panel only shows an empty table frame with no explanation.
2. **Study 2:** Expert F approached the same problem from a different perspective, questioning the use of space in the side-by-side layout: *"the blank space is almost covering half the screen…"* She gave H03 (User control and freedom) a score of 4 (Catastrophic) and H07 (Flexibility and efficiency of use) a score of 3. In reality, there are almost no scenarios where a user needs to see both the data panel and the results panel simultaneously. Placing them side-by-side is more of a stylistic choice for the layout.

The report from Study 1 already proposed two reference points: the dual-panel Outline Pane structure of SPSS and the collapsible floating table of contents in Wikiwand. The feedback from the experts in Study 2 provided stronger support for the latter. Since users complain about the right-side panel "wasting space," embedding the results navigation into this space is a two-birds-one-stone solution: it resolves the confusion of results management and gives the right-side panel a more practical function.

Recommendation: Without changing the freeform Notebook-style model, superimpose a lightweight results navigation layer on the right-side panel. In its collapsed state, it would appear as a series of small dots (indicating how many analysis items currently exist and the current viewing position). In its expanded state, it would provide a full list of items with rename, delete, and reorder functionalities.

5.e. How might we enable users from different software backgrounds to build the correct mental model for operations?

Priority: Medium Impact / High Development Effort. This is a long-term investment.

Evidence Convergence: Both studies independently discovered mental model conflicts, and the range of reference software involved was broader than expected.

1. **Study 1:** The experience of P02 with GraphPad became a source of negative transfer. He admitted, *"For someone used to GraphPad, suddenly switching to something like SPSS… is quite difficult."* P04 brought their Excel habits to Jamovi,

repeatedly trying to copy and paste analysis results into the data grid: *"I want to paste this whole table next to the data for easy comparison."* This exposed a design issue where the boundary between the data area and the results area in Jamovi is unclear.

2. **Study 2:** Expert F used Excel as her reference for the entire evaluation, giving H04 (Consistency and standards) a score of 4. Expert S, on the other hand, used Stata as her reference and gave the same principle a score of 0. For the same interface, two experts produced a score range from 0 to 4 due to different reference points. This itself illustrates that the diversity of mental models is not an isolated case.

It is worth noting that Jamovi's backend is R, and its interaction logic is closest to the three-part structure of SPSS (data grid + analysis menu + results panel). However, the actual user base includes users of Excel, GraphPad, R, and Stata, as well as complete novices. Each type of user brings different expectations: Excel users expect the data grid to be freely editable and pastable; GraphPad users expect a more streamlined variable configuration process.

Recommendation: At first launch, provide a brief onboarding tour. It should not be a generic feature overview but rather a customized guide based on the user's self-reported software background ("What statistical tool did you primarily use before?"). This would provide tailored hints on key differences. For example, for users from an Excel background, it would emphasize, "The data grid is a data source, not a workspace. Analysis results are generated independently in the right-side panel." For users from a GraphPad background, it would explain that variable types need to be manually confirmed. These targeted hints would only be a few steps but could preemptively calibrate user expectations at the most critical points of mental model divergence.

5.f. How might we better serve users in the macOS ecosystem?

Priority: Medium Impact / Low Development Effort. Most of the work involves system integration with existing software packages.

Evidence Convergence: Study 1 and Study 2 each discovered two independent macOS adaptation issues, which together point to a larger pattern.

1. **Study 1:** P05 found that the `File` menu in the macOS top status bar lacked an "Open" function, violating macOS platform design conventions. He said, *"First I'd think to click 'File' in my status bar, because on macOS you can usually import files from there, but then I saw there was no 'Open File' or anything like it."*
2. **Study 2:** Expert S discovered that a CSV file downloaded from WhatsApp was automatically associated with the Numbers format on macOS, and Jamovi does not support importing Numbers files. She argued, *"You need to support this format because there are a lot of Mac users. They may use Numbers to import data."* In the roundtable, Expert F agreed that this was a reasonable need.

These two issues are different on the surface (one a missing menu item, one an unsupported format), but their root cause is the same: Jamovi's development focus may be biased toward the Windows ecosystem, with insufficient attention paid to macOS

platform conventions and user habits. Given the high penetration of macOS in the academic research community, this gap is worth addressing.

Recommendation:

1. Ensure that the macOS version includes a standard `Open` function in the `File` menu of the top menu bar. This might be an adaptation issue with the Electron framework rather than an intentional design decision.
2. Support the import of the `Numbers` format.

6. Epilogue

The most important methodological insight from these two studies may not lie in any single design recommendation, but in the analytical hierarchy they jointly reveal.

Study 1's behavioral coding recorded at the operational layer "what the user did" and "how many times they failed." Study 2's PURE evaluation predicted at the cognitive layer "what the user needs to understand to proceed." When we superimposed the two using an alluvial diagram, we discovered a cascading phenomenon: a difficulty at the cognitive layer (PURE 3 for IdentDatTyp) propagated downstream to the behavioral layer, erupting in the form of "repeated failures of a preceding step" (PURE 1 for TrapDepVar) in the behavioral record.

This means that if you only look at the behavioral data, you will conclude that "the drag-and-drop interaction is the main problem." If you only look at the expert evaluation, you will conclude that "data type identification is the main problem." Neither is complete, but neither is contradictory. They are capturing different stages of the same causal chain.

For design practice, this teaches us that when a "simple" step with a PURE score of 1 accumulates a large number of behavioral errors, we should not rush to modify the interface of that step itself. Instead, we should trace upstream to find the unresolved cognitive prerequisite. In this study, fixing the drag-and-drop feedback for TrapDepVar (Recommendation 1) is certainly necessary. But the more fundamental intervention is to provide proactive data quality checks at the IdentDatTyp stage (Recommendation 3). By eliminating the difficulty at the cognitive layer, the behavioral errors downstream will naturally decrease.

This also provides a practical recommendation for the PURE method itself. When interpreting PURE evaluation results, one should not look at the score of each step in isolation. Instead, one must pay attention to the cognitive dependencies between steps. If a PURE=3 step is a cognitive prerequisite for several subsequent PURE=1 steps, its actual impact is far greater than its score alone would suggest.

References

7. Supplemental Materials

7.a. Event Coding Procedure

The raw behavioral data consisted of screen recordings from six participants. A researcher watched each recording and produced an atomic Event List, logging every

discrete interface action (e.g., “clicked Analyses menu,” “dragged nightmare_frequency to Dependent Variable box”) along with its timestamp.

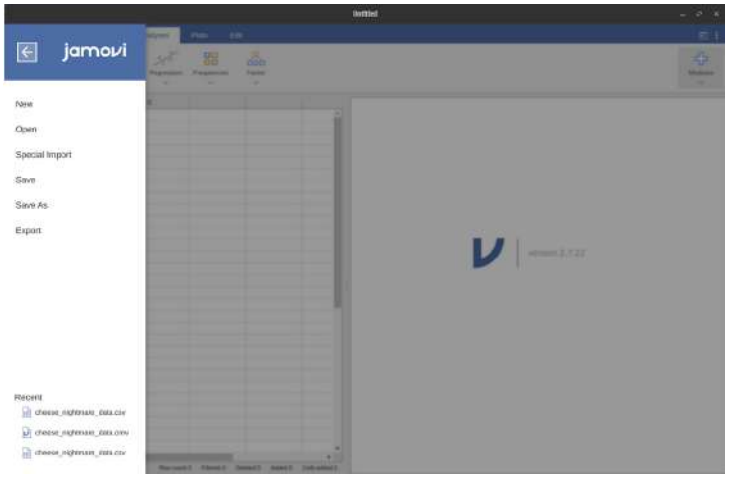
This Event List was then submitted to DeepSeek for initial classification into task phases. The researcher subsequently reviewed and re-labeled every entry. Of the 100 key events that feed into the cross-analysis, three were re-classified during this review.

```
[Research Proposal Attached] This is my research design. I am conducting a Phase
II study. I have some standardized task lists. I would like you to map the
following list of key events for participants to the task breakdown list. If
there is a mismatch, you can fill in N/A.

... Key event attached ...

Your format is

P1-01:31 | TX-PY-XXXXXXXX | The subject began to perform the task and tried to
find the file.
```

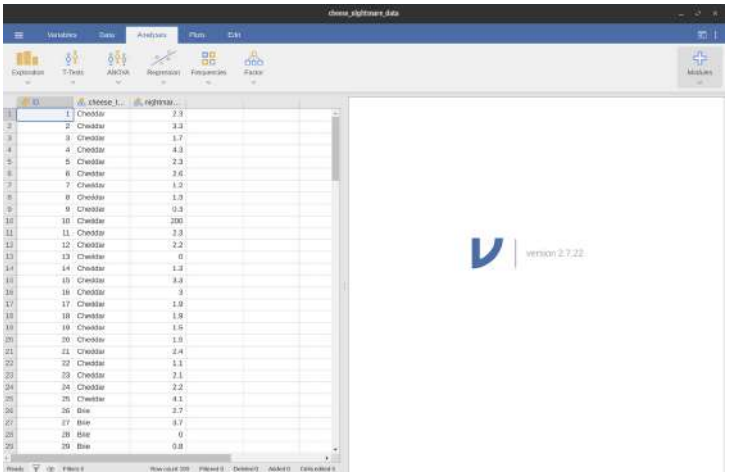
7.b. Surveys

7.b.i. *PURE:*

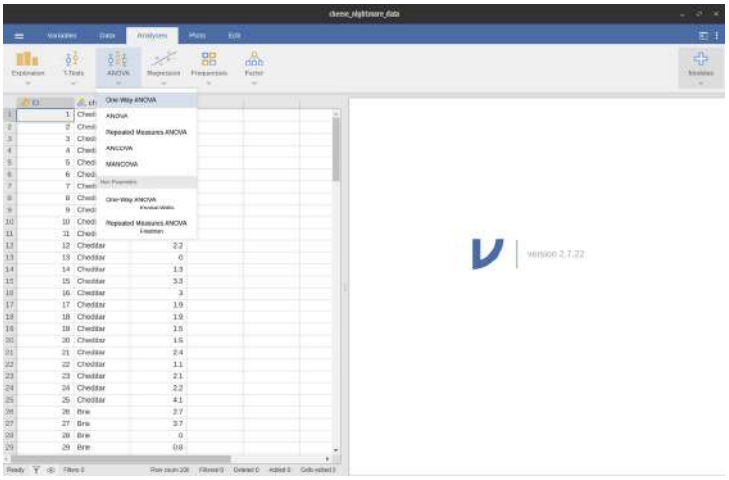
T1 - P1. Click the File menu on the top left, load the CSV file use Open or Special Import.



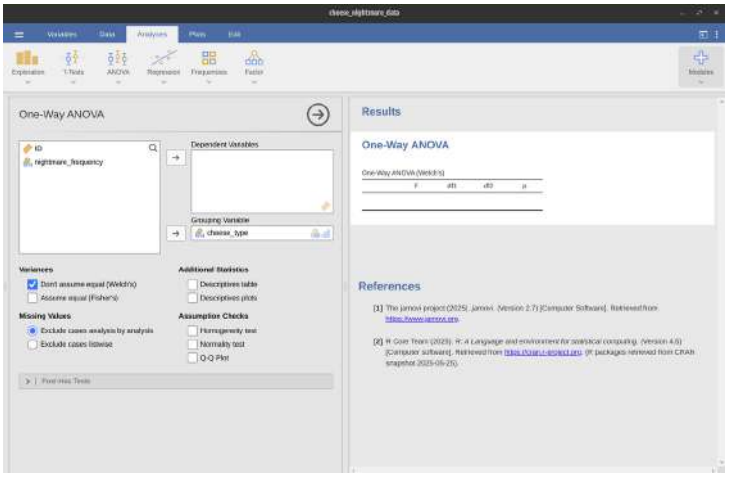
T2A - P1. Find the Analyses tab on the top of the interface, and click it to list all analysis methods.



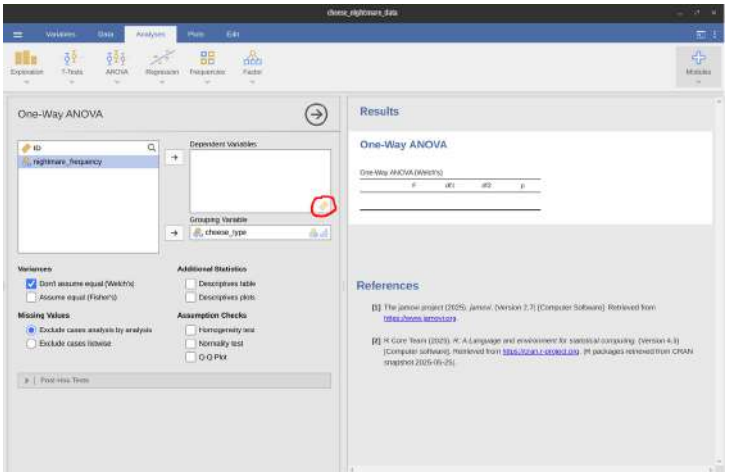
T2A - P2. Click ANOVA and find one-way ANOVA as the analysis method.



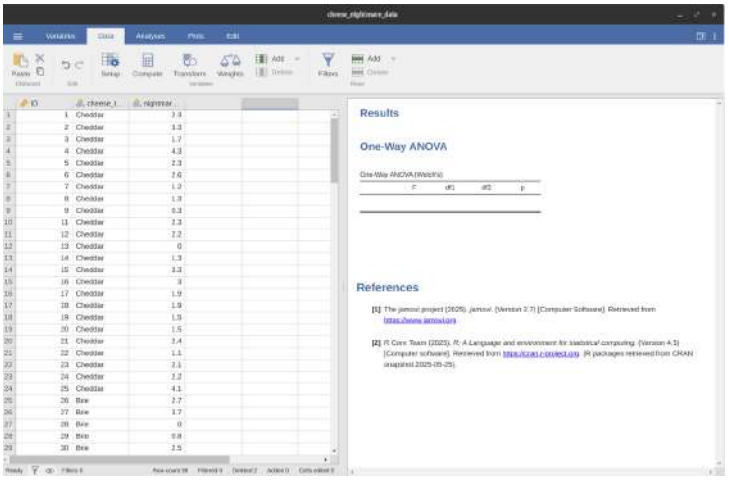
T2A - P3. Drag cheese_type to the Grouping Variable



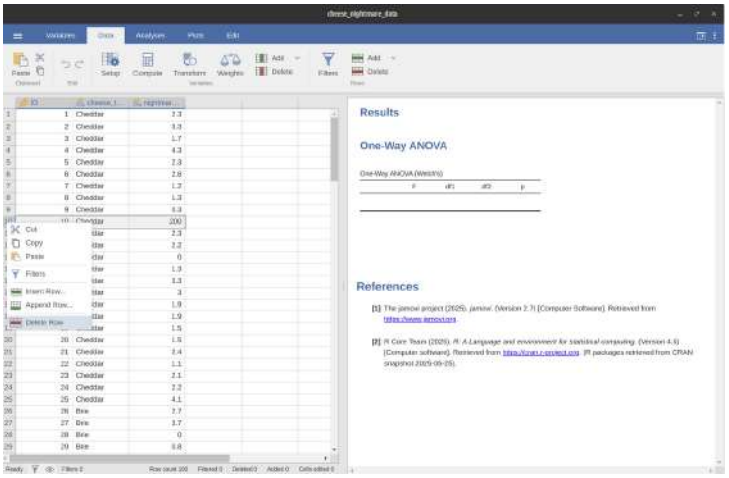
T2A - P4. Drag nightmare_frequency to the Dependent Variable, and you will notice an error happens, the icon circled will blink, means a data type mis-alignment.



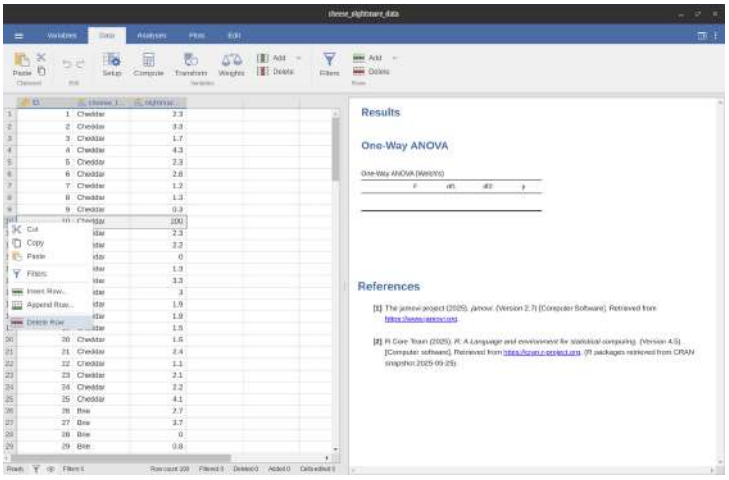
T2B - P0. Click the Data Tab to see the Data view



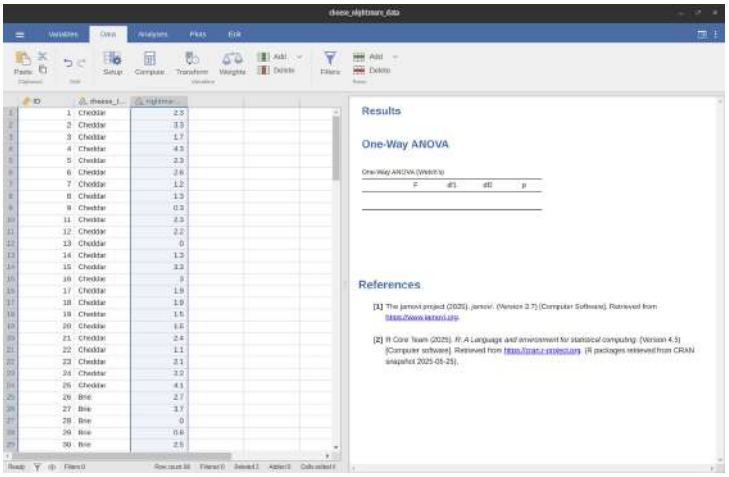
T2B - P1. Find the Outlier at Line 10 and the Typo at Line 60.



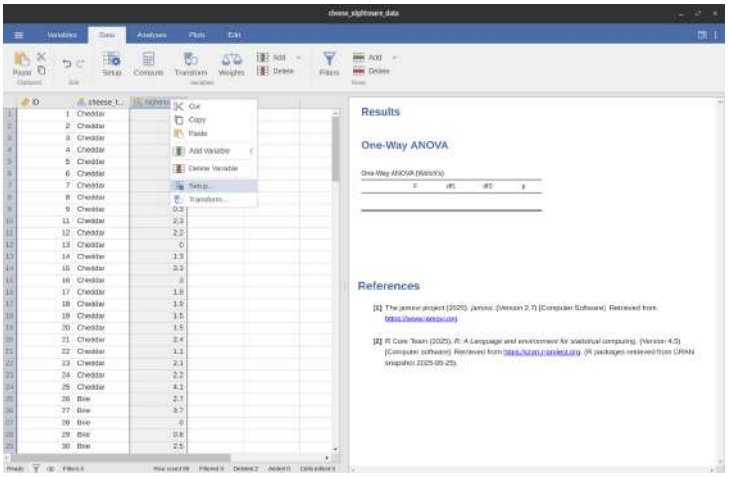
T2B - P2. Right click the column and Delete this row.



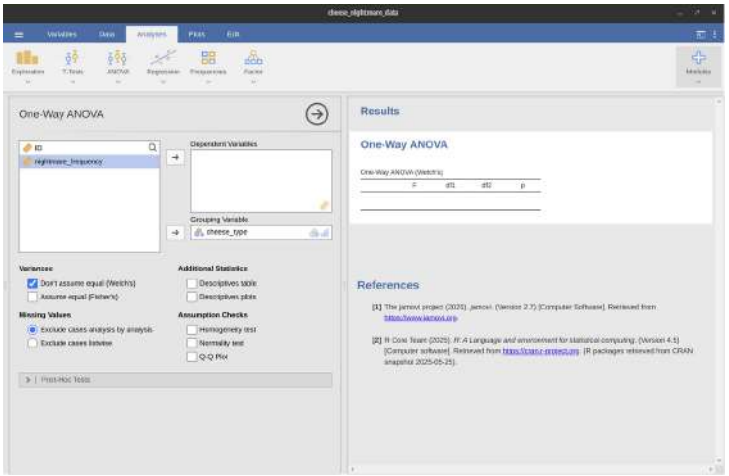
T2B - P3. Identify the Column data type of the nightmare_frequency is wrong via the column icon.



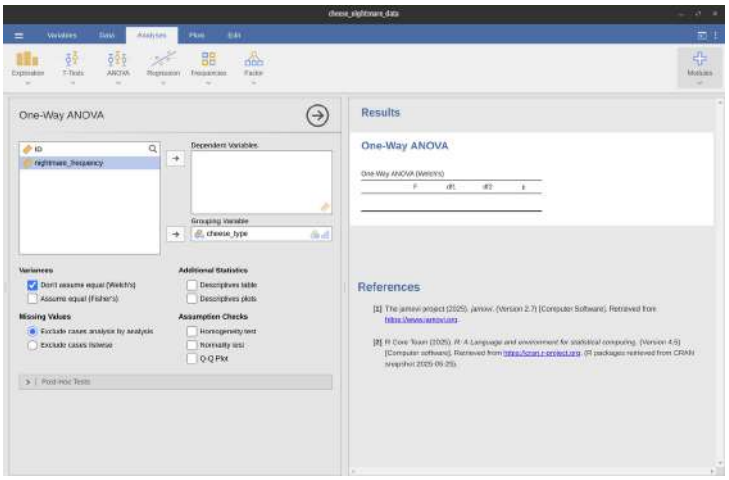
T2B - P4. Right click the nightmare_frequentcy column, and find the Setup context menu item.



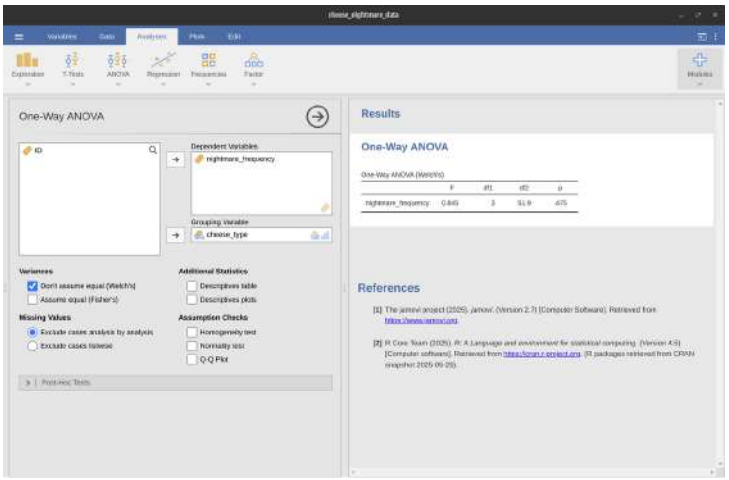
T2B - P5. Change the Measure Type to Continuous.



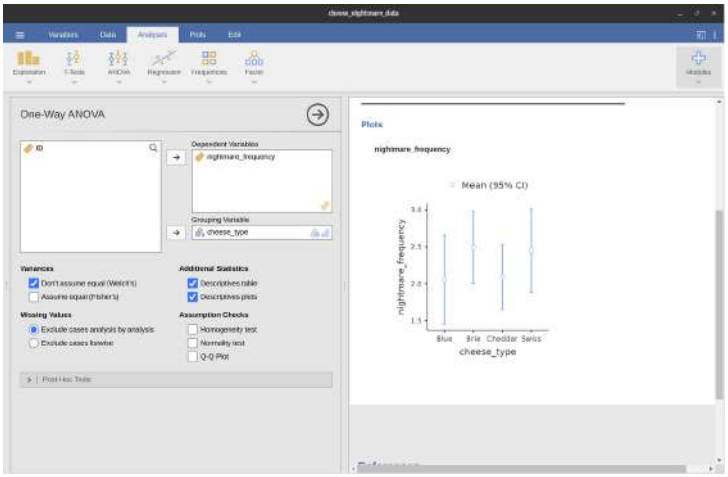
T3 - P1. Click the Empty One-Way ANOVA report on the right panel to go back to the ANOVA analysis screen.



T3 - P2. Drag the nightmare_frequency to the Dependent Variables box.



T3 - P3. Enable Descriptive Table and Descriptive Plots on the stat feature panel.



For each task, we have a multiple choice question:

- 1 - The step can be accomplished **easily** by the target user, due to low cognitive load or because it's a known pattern, such as the acceptance of a terms-of-service agreement.
- 2 - The step requires a **notable degree of cognitive load** (or physical effort) by the target user, but can generally be accomplished with some effort.
- 3 - The step is **difficult** for the target user, due to significant cognitive load or confusion; some target users would likely fail or abandon the task at this point.

7.b.ii. *Heuristic Evaluation:*

Q1: Visibility of System Status The design should always keep users informed about what is going on, through appropriate feedback within a reasonable amount of time.

- Communicate clearly to users what the system's state is — no action with consequences to users should be taken without informing them.
- Present feedback to the user as quickly as possible (ideally, immediately).
- Build trust through open and continuous communication.

Q2: Match Between the System and the Real World

The design should speak the users' language. Use words, phrases, and concepts familiar to the user, rather than internal jargon. Follow real-world conventions, making information appear in a natural and logical order.

- Ensure that users can understand meaning without having to go look up a word's definition.
- Never assume your understanding of words or concepts will match that of your users.
- User research will uncover your users' familiar terminology, as well as their mental models around important concepts.

Q3: User Control and Freedom

Users often perform actions by mistake. They need a clearly marked “emergency exit” to leave the unwanted action without having to go through an extended process.

- Support Undo and Redo.
- Show a clear way to exit the current interaction, like a Cancel button.
- Make sure the exit is clearly labeled and discoverable.

Q4: Consistency and Standards

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform and industry conventions.

- Improve learnability by maintaining both types of consistency: internal and external.
- Maintain consistency within a single product or a family of products (internal consistency).
- Follow established industry conventions (external consistency).

Q5: Error Prevention

Good error messages are important, but the best designs carefully prevent problems from occurring in the first place. Either eliminate error-prone conditions, or check for them and present users with a confirmation option before they commit to the action.

- Prioritize your effort: Prevent high-cost errors first, then little frustrations.
- Avoid slips by providing helpful constraints and good defaults.
- Prevent mistakes by removing memory burdens, supporting undo, and warning your users.

Q6: Recognition Rather than Recall

Minimize the user’s memory load by making elements, actions, and options visible. The user should not have to remember information from one part of the interface to another. Information required to use the design (e.g. field labels or menu items) should be visible or easily retrievable when needed.

- Let people recognize information in the interface, rather than forcing them to remember (“recall”) it.
- Offer help in context, instead of giving users a long tutorial to memorize.
- Reduce the information that users have to remember.

Q7: Flexibility and Efficiency of Use

Shortcuts — hidden from novice users — may speed up the interaction for the expert user so that the design can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

- Provide accelerators like keyboard shortcuts and touch gestures.
- Provide personalization by tailoring content and functionality for individual users.
- Allow for customization, so users can make selections about how they want the product to work.

Q8: Aesthetic and Minimalist Design

Interfaces should not contain information that is irrelevant or rarely needed. Every extra unit of information in an interface competes with the relevant units of information and diminishes their relative visibility.

- Keep the content and visual design of UI focused on the essentials.
- Don't let unnecessary elements distract users from the information they really need.
- Prioritize the content and features to support primary goals.

Q9: Help Users Recognize, Diagnose, and Recover from Errors

Error messages should be expressed in plain language (no error codes), precisely indicate the problem, and constructively suggest a solution.

- Use traditional error-message visuals, like bold, red text.
- Tell users what went wrong in language they will understand — avoid technical jargon.
- Offer users a solution, like a shortcut that can solve the error immediately.

Q10: Help and Documentation

It's best if the system doesn't need any additional explanation. However, it may be necessary to provide documentation to help users understand how to complete their tasks.

- Ensure that the help documentation is easy to search.
- Whenever possible, present the documentation in context right at the moment that the user requires it.
- List concrete steps to be carried out.

For each rules, we have the following multiple choice question:

0 = I don't agree that this is a usability problem at all

1 = Cosmetic problem only: need not be fixed unless extra time is available on project

2 = Minor usability problem: fixing this should be given low priority

3 = Major usability problem: important to fix, so should be given high priority

4 = Usability catastrophe: imperative to fix this before product can be released

8. Procedural Materials

8.a. Informed Consent Form

Wilfrid Laurier University Informed Consent Form

Research Title: User Experience (UX) Analysis of Jamovi Statistical Software

Researcher: LI Kanyu, lixx4442@mylaurier.ca

Supervisor: Susie Simon, ssimon@wlu.ca

1. Purpose of the Study: This study aims to explore user experience with the open-source statistical software Jamovi. By collecting user feedback regarding the interface, functional logic, and visual feedback, we hope to identify the software's strengths and weaknesses, thereby providing insights for the future optimization of open-source statistical tools.

2. Procedures: If you decide to participate, you will be asked to complete several specific statistical tasks under observation while sharing your thoughts in real-time (approximately 30 minutes) and join a round table to share your idea. This study does not involve any tasks that cause physiological or psychological stress.

3. Risks and Discomforts: This study poses minimal risk to participants. You may experience mild eye strain from using the software or brief anxiety if you encounter difficulty with the tasks. You may request a break or stop participating at any time.

4. Benefits: You may not receive any direct personal benefit from participating in this study. However, your feedback will contribute to improving open-source scientific tools, making statistics more accessible to students and researchers globally.

5. Conflict of Interest: The researcher, Li Kanyu, and the supervisor, Susie Simon, have no financial or other conflicts of interest with the official Jamovi team or its commercial competitors. This study is purely academic in nature.

6. Confidentiality

    - Anonymity: All collected data will be anonymized. Your name will not be linked to the research results.

    - Storage: Data will be encrypted and stored on a secure WLU cloud server or on a password-protected computer.

    - Retention Period: Raw data will be retained for 3 months after the study is published or completed, after which it will be permanently deleted.

7. Voluntary Participation:

Participation is entirely voluntary. You may withdraw from the study at any time without providing a reason and without penalty. If you choose to withdraw, your data will be destroyed.

8. Contact Information

    - Concerning the study: Please contact Li Kanyu (lixx@mylaurier.ca) or the faculty supervisor, Susie Simon (ssimon@wlu.ca).

    - Concerning ethics: If you have questions about your rights as a research participant, please contact:

    - WLU Research Ethics Board > Phone: 548.889.3518 Email: REB@wlu.ca

9. Consent: By clicking “Agree” or signing below, you confirm that you have read and understood the information above, are at least 18 years old, and agree to participate in this study.

8.b. Raw Questionnaire Result

Question	Expert A	Expert F	Expert S	Conclusion
Task Evaluation				
T1-P1 DataLoad	1	1	2	1
T2A-P1 DataProcessingTrap	1	1	1	1
T2A-P2 ListANOVAFeat	1	1	1	1
T2A-P3 GrpVar	1	2	1	1
T2A-P4 TrapDepVar	1	1	1	1
T2B-P0 FindDataTab	1	1	1	1
T2B-P1 IdentWrongData	1	1	2	1
T2B-P2 RemoveRow	1	1	1	1
T2B-P3 IdentDatTyp	3	2	3	3
T2B-P4 FindSetupEntr	1	1	1	1
T2B-P5 ChangeDatTyp	1	1	1	1
T3-P1 GoBackANOVA	2	2	1	2
T3-P2 DragDepAgain	1	1	1	1
T3-P3 EnableStatFeatures	1	1	1	1
Heuristic Evaluation				
H01 Visibility Of System Status	1	3	1	
H02 Match Between The System And The Real World	1	3	0	
H03 User Control And Freedom	2	4	0	
H04 Consistency And Standards	1	4	0	
H05 Error Prevention	0	3	1	
H06 Recognition Rather Than Recall	0	3	0	
H07 Flexibility And Efficiency Of Use	2	3	0	
H08 Aesthetic And Minimalist Design	0	1	0	

H09	Help Users Recognize Diagnose	3	3	2
H10	Help And Documentation	3	4	2

8.c. Facility Guide

8.d. Participant Tracker

Participant	Consented
Expert A	Yes
Expert F	Yes
Expert S	Yes

Video Link: https://drive.google.com/file/d/13N0HcFRSeFnkNN5TcgLvRsvC5lwkkTsl/view?usp=sharing

9. Facility Guide

9.a. Pre-Session Preparation

9.a.i. *For the Moderator::*

- Compile a **score summary table** showing:
  - PURE step ratings (by expert)
  - Heuristic Evaluation ratings (by expert)
- Highlight all items where **ratings differ by more than 1 point** (e.g., one expert gave 1, another gave 3).
- Prepare a **discussion agenda** with time allocation (approx. 60–90 minutes).
- Ensure recording/transcription tools are ready (if applicable).

9.a.ii. *For Experts::*

- Be prepared to explain the **rationale behind each rating**, especially for items they rated differently from others.
- Avoid discussing ratings with other experts before the session.

9.b. Round Table Structure

9.b.i. *Review of PURE Ratings:*

For each PURE sub-step:

- If **all ratings are identical**, briefly acknowledge and move on.
- If **ratings differ**:
  - Ask each expert to explain their score.
  - Probe for:
    - What cognitive load they imagined for the target user.
    - Whether the difficulty came from **interface ambiguity**, **lack of feedback**, or **domain knowledge**.
  - Facilitate a brief discussion to see if a **consensus score** can be reached.
  - If consensus is not possible, note the **range and rationale** as a finding.

9.b.ii. *Review of Heuristic Evaluation Ratings:*

For each of the 10 Nielsen heuristics:

- If **ratings are identical**, document as agreed.
- If **ratings differ**:
  - Ask each expert to share examples from the interface that influenced their rating.
  - Encourage comparison of **reference software** (e.g., Excel vs. Stata vs. SPSS) if that shaped their perception.
  - Discuss whether the issue is **cosmetic, minor, major, or catastrophic**.
  - Attempt to align on a **consensus severity score**.

9.b.iii. *Closing (5 min):*

- Thank experts for their participation.
- Remind them of next steps (e.g., data analysis, reporting).
- Confirm consent to use anonymized quotes and scores.