Cost-Benefit Modeling of Interactive Visual Machine Learning Systems

Yixuan Wang, Rolando Garcia, Alex Endert, and Ross Maciejewski

Abstract—Machine learning systems often benefit from human involvement, but human-in-the-loop support is not free: teams may need to build specialized interfaces, analysts must spend time interacting with them, and some interactions trigger costly retraining or re-optimization. The central design question is therefore not simply whether interaction helps, but when it is worth the added effort relative to other interventions, such as improving the data, revising the model or training specification, or relying on automated workflows alone. In this paper, we build upon Van Wijk's foundational economic model of visualization, extend it by explicitly incorporating machine learning components, and present a cost-benefit framework for reasoning about these choices in interactive visual machine learning systems. We identify measurable quantities that govern practical trade-offs, including analyst time, interaction frequency, training latency, model improvement, downstream operational savings, and failure-prevention value. We organize these quantities into workflow-specific models for three common intervention targets: data iteration, model or specification iteration, and evaluating and understanding model behavior. We then instantiate the framework on representative systems to show how it supports break-even analysis, latency bounds, and stopping criteria for interactive analysis. The contribution is a practical decision framework for reasoning about when interactive visual tooling is likely to be justified, which intervention can provide value, and where human effort can be allocated more effectively across IVML workflows.

Index Terms—Visual analytics for machine learning, interactive visual machine learning, cost-benefit analysis

1 Introduction

Interactive Visual Machine Learning (IVML) systems differ from traditional visualization systems in several ways. First, they incorporate machine learning models as active computational components that transform data, generate intermediate representations, and evolve over time. Second, interaction in IVML systems often triggers computationally expensive processes such as model retraining or re-optimization [59]. Third, IVML workflows span multiple stages of the machine learning lifecycle [2], from data preparation and model construction to post-deployment analysis. Finally, unlike general visualization where "knowledge gain" can be subjective, IVML systems expose a suite of quantitatively measurable variables, such as training time, model accuracy, convergence behavior, and interaction frequency, that make formal cost–benefit modeling both feasible and necessary. Together, these characteristics make IVML systems central to complex data analysis and model development tasks [22]. By enabling users to inspect data, understand models, and provide feedback through visual interfaces, IVML systems aim to combine the complementary strengths of human judgment and automated learning. As machine learning is increasingly adopted across more domains, IVML systems have become a central research topic in visualization and visual analytics [9, 31].

Prior surveys show that visual analytics for machine learning spans diverse functional classes across the ML lifecycle, from data preparation, labeling, and feature engineering to model understanding, debugging, steering, and post-hoc output analysis [8, 51, 75]. While these systems demonstrate the expressive power of interaction and visualization, their evaluation typically focuses on usability [74], task performance [15], or qualitative insight generation through user studies [25], case studies [62], or expert feedback [39]. Much less attention has been paid to a fundamental question underlying IVML system design: **Do the benefits gained, such as improved model performance and user insight, justify the compounded costs of both developing the specialized visualization infrastructure and the iterative operational expenses of human labor and model retraining?** This question is not new in visualization research. Van Wijk's visualization model frames visualization as a process that yields knowledge while explicitly accounting for the costs incurred during this process [63]. This cost–benefit perspective provides a principled foundation for reasoning about effectiveness, yet existing discussions often overlook the unique overhead of the iterative feedback loops and computational retraining inherent to IVML.

Recent work has begun to explore quantitative and simulation-based approaches to evaluating interactive machine learning, demonstrating that human-in-the-loop interaction can exhibit diminishing returns and that its effectiveness depends strongly on data characteristics and task conditions [67]. Simulations of "perfect agents" have shown that there often exists an optimal stopping point where the benefits gained from further label correction become marginal relative to the increasing costs. While these efforts provide valuable insights, they typically focus on a single type of interaction or a specific stage of the workflow. As a result, it remains unclear how cost–benefit trade-offs vary across different classes of IVML systems and how these trade-offs are shaped by the placement of machine learning within the visualization workflow.

In this work, we develop a general cost–benefit modeling framework for IVML systems. Building on Van Wijk's visualization model, we propose multiple workflow variants that explicitly incorporate machine learning components and capture how interaction primarily affects different system states: the data, the model, or the user's understanding of model behavior. These workflows capture key structural differences in how data, models, visualization, and user interaction are coupled, serving as the foundation for identifying measurable cost and benefit metrics. Based on these workflows, we formulate cost–benefit models that characterize the trade-offs inherent in different IVML system types. We then instantiate these models using real-world IVML systems, and apply a structured method to evaluate when interaction is effective, when it is unnecessary, and how IVML systems should be designed to maximize their return on human and computational effort. The framework is not intended to provide universal economic predictions or replace empirical IVML evaluation. Instead, it serves as a structured cost-benefit accounting framework that makes otherwise implicit trade-offs explicit: which costs are introduced by a particular form of interaction, which benefits can plausibly offset them, and which assumptions must hold for an IVML workflow to be justified. In this sense, the framework complements existing IVML evaluation methods by connecting conventional outcomes to the human, computational, and operational costs required to obtain them.

• *Y. Wang, R. Garcia, and R. Maciejewski are with Arizona State University, USA. E-mail: {ywan1290, rolando.garcia, rmacieje}@asu.edu.*
• *A. Endert is with Georgia Institute of Technology, USA. E-mail: endert@gatech.edu*

Manuscript received xx xxx. 202x; accepted xx xxx. 202x. Date of Publication xx xxx. 202x; date of current version xx xxx. 202x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.202x.xxxxxxx

2 Background

In this section, we review prior work relevant to cost-benefit reasoning in IVML systems, focusing on how existing evaluation methods and interactive learning frameworks account for interaction benefits, human effort, and computational cost. We then identify the need for a unified framework that relates these factors across different IVML workflows.

2.1 Evaluation and Cost-Benefit Reasoning in Interactive Visual Machine Learning Systems

Evaluation has long been a central concern in IVML systems. In the visual analytics community, systems are commonly evaluated through controlled experiments [78], case studies [62], expert feedback [39], and qualitative assessments [10]. Conceptual frameworks such as Van Wijk's economic model of visualization formalize visualization as a process that transforms data into knowledge at a cost, providing a theoretical foundation for reasoning about value in terms of knowledge gain versus computational and cognitive expense [63]. Subsequent work has extended this line of visualization process modeling toward knowledge generation, explicit knowledge, guidance, and model-coupled interaction. Knowledge generation frameworks formalize the integration of user interaction, system response, and reasoning, along with the externalization and use of explicit knowledge [23, 53]. Complementary work extends foundational visual analytics models to capture guidance and to couple user interaction directly to computational models through semantic interaction [11, 21]. While these works provide important conceptual grounding for knowledge-assisted, guidance-supported, and interaction-driven visual analytics, they primarily explain how users, knowledge, and system responses interact during analysis. They do not account for the cost-benefit trade-offs introduced by iterative retraining, model update latency, or interaction-driven changes to data or model state. In contrast, machine learning systems are primarily evaluated through quantitative performance metrics such as accuracy, loss, precision–recall, and generalization error, alongside computational measures such as training time and resource utilization. ML evaluation frameworks emphasize performance and efficiency, often focusing on quantitative metrics while accounting less for the human labor involved in data curation or model steering. Recent work highlights the operational complexity and lifecycle costs of deploying ML models. For instance, Shankar et al. [59] document how engineers struggle to anticipate model behavior before deployment. Sculley et al. [56] highlight that real-world ML systems carry significant hidden technical debt, with maintenance and operational costs often exceeding the initial modeling effort. These studies reveal hidden costs in deploying ML systems but do not provide formal mechanisms for comparing interactive approaches against automated baselines.

Human-in-the-loop and interactive machine learning (IML) research occupies an intermediate space between visualization and machine learning evaluation. Foundational IML work establishes that human input is not merely an external annotation source but part of an interactive learning loop in which users steer model behavior through feedback, and subsequent surveys taxonomize the resulting space of methods, including active learning, interactive machine learning, and machine teaching [3, 43]. A complementary line of human-AI interaction research examines how such systems should be designed and evaluated from the user's perspective, contributing interface design guidance and structured evaluation of user-facing factors such as trust, interpretability, and explainability [4, 19, 60]. Within visualization, design study methodology provides a foundation for problem-driven visual analytics development, and surveys of visual analytics for machine learning organize these techniques across the model lifecycle [57, 75].

While the aforementioned works establish broad methodologies and evaluation principles for human-in-the-loop systems, explicit cost-aware reasoning is especially prominent in active learning and visual-interactive labeling. Active learning aims to improve model performance while reducing labeling cost by selecting informative instances for human annotation [24, 58]. *Visual-interactive labeling* similarly studies how users, models, views, and feedback loops support labeling workflows; for example, VIAL formalizes a unified process for visual-interactive labeling, and subsequent work compares visual-interactive labeling with active learning strategies [7, 8]. Related work has also modeled human effort in visual analytic systems, studied machine-initiated interaction strategies for reducing user involvement in interactive classification, and reported human training-time requirements in interactive reinforcement learning [17, 69, 77].

Beyond labeling and interactive training, conceptual frameworks for VA assisted machine learning further describe how humans, data, models, and visual interfaces are structurally coupled. Sacha et al.'s framework [52] describes how interactive visualization can make machine learning components inspectable and changeable. Building on this process-oriented view, VIS4ML [51] provides a formal ontology for VA-assisted machine learning. Recent frameworks such as HDMI Canvas [6] further characterize how humans, data, and models contribute to and benefit from interactive visual analytics processes. These works provide important foundations for human-centered interaction, labeling efficiency, and IVML workflow characterization; they do not provide a unified cost-benefit formulation for comparing interaction targets across IVML workflows.

2.2 Research Gap

IVML systems introduce structural characteristics that challenge existing evaluation paradigms. As formalized in our generalized workflow (Section 3), these systems couple data, model state, visualization, and user knowledge through iterative feedback loops.

Prior work lays foundations for human-centered evaluation, active learning, visual-interactive labeling, and workflow characterization, but these perspectives are usually task-specific or studied in isolation. As a result, no single framework jointly weighs human effort, interaction latency, and computational overhead against the value produced by interaction, including model improvement, downstream savings, knowledge gain, and failure prevention. Current evaluation approaches can lack clear criteria for when interaction provides an effective cost-benefit trade-off. Lifecycle and functional categorizations [8, 51, 75] are useful for describing system capabilities, but they do not directly determine which cost and benefit terms become active. For cost-benefit modeling, the critical distinction is whether interaction modifies the data, modifies the model state, or merely supports users' evaluation and understanding of model behavior. Our framework addresses this gap by organizing IVML workflows around these interaction targets and weighing the trade-offs between human effort, computational cost, and the value produced by interaction.

3 Interactive Visual Machine Learning Systems

IVML and visual analytics systems for machine learning are commonly described through lifecycle-based and function-based views. These views help locate an interactive technique within the broader ML workflow and describe what a system supports. For IVML cost-benefit modeling, however, the central question is different: which part of the system the interaction changes. The interaction target determines which computational costs, cognitive costs, and benefit pathways become active, so we organize IVML workflows by interaction target as the basis for cost-benefit modeling.

Interaction can act on one of three targets, each with a distinct feedback pathway. Data-targeted interaction updates the data available to the learning process and can alter future training inputs. Model-targeted interaction modifies the model state itself and can trigger retraining or re-optimization. Understanding-oriented interaction manipulates the visualization specification to support the evaluation and interpretation of data or model behavior. In practice, these targets may appear individually [18, 72] or in combination within a single system [15, 61]. Appendix B maps the IVML categories from Yuan et al. [75] to the interaction targets used here.

3.1 Unified IVML Workflow

We build on the economic model of visualization proposed by Van Wijk [63] and extend it to incorporate machine learning as an explicit computational component. In this model, rectangular nodes denote state containers, such as data and knowledge, and circular nodes denote

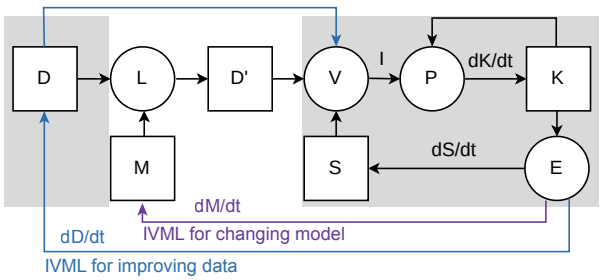

Fig. 1: The generalized workflow of IVML, extending Van Wijk's visualization model [63]. Rectangular nodes denote state containers, including raw training data (D), model state (M), derived data (D'), visualization specification (S), and user knowledge (K). Circular nodes denote processes, including learning (L), visualization/visual interface (V), image (I), perception (P), and interaction (E). The gray-shaded components correspond to *the foundational visualization model* adapted from Van Wijk. Black arrows indicate shared workflow connections, while colored solid arrows indicate IVML-specific interaction pathways: blue arrows correspond to data-targeted interaction, and the purple arrow corresponds to model-targeted interaction.

processes that transform inputs into outputs, such as visualization, perception, and interaction. Figure 1 illustrates a general IVML workflow. We retain the core perceptual interaction loop from Van Wijk's model, including visualization (V), perception (P), interaction (E), visualization specification (S), and knowledge (K). In this loop, derived data is visualized, perceived by the user, and updated through interaction. Unlike the original model, IVML introduces a learning process between data and visualization. Raw training data D is processed by a learning procedure L, parameterized by the model state M, producing derived data D':

$$D'(t) = L(D(t), M(t)). \tag{1}$$

As in Van Wijk's model, the user observes the rendered visual output I and acquires knowledge through perception:

$$\frac{dK}{dt} = P(I, K), \tag{2}$$

where $P(\cdot)$ models perceptual and cognitive processing shaped by prior knowledge and task context. Acquired knowledge K is continuously shaped by the user's existing expertise and memory, rather than being a passive or memoryless process. Knowledge K drives interaction and exploration actions that update the system. In Van Wijk's model, interaction only updates the visualization specification, which can be formalized as:

$$\frac{dS}{dt} = E(K), \tag{3}$$

In IVML, the visualization node (V) represents the visual interface through which users inspect the system and issue interaction commands. Interactions executed within this visual environment may therefore propagate beyond the display specification to affect the underlying data or model. These interaction pathways describe three common IVML configurations, distinguished by whether interaction primarily modifies the data (D), modifies the model (M), or manipulates the visualization specification (S) to support evaluation and understanding of model behavior. Equations 1–3 define the core interaction dynamics shared across IVML systems (black arrows in Figure 1). The following subsections provide representative examples and describe how these dynamics appear when interaction primarily targets data, model state, or model-behavior evaluation.

3.2 Interactions for Improving Training Data

One common interaction target is the training data, where users directly modify labels, features, or weights to improve data quality [15, 32, 66, 72]. In such systems, raw training data D is often visualized directly, as shown by the direct pathway from D to V in Figure 1. The visualization may also incorporate model-derived representations D', such as predicted labels or uncertainty estimates, but interaction primarily targets the underlying data D. Through Equation 2, users gain knowledge about data errors or inconsistencies, which then drives interaction that acts directly on the training data, resulting in updates such as label correction or feature refinement:

$$\frac{dD}{dt} = h(E(K)), \tag{4}$$

where $h(\cdot)$ models interaction-driven data modification. While the human interacts with the data layer, the machine learning model could improve indirectly through retraining on the refined dataset.

3.3 Interactions for Modifying Model

Another interaction target is the model itself, where users directly steer or modify model behavior through interaction [13, 47, 61, 70]. In these systems, data is typically processed by the model before visualization. Users interact with representations of the model's internal behavior, such as learned embeddings, training dynamics, or decision boundaries. As shown in the Figure 1, the purple arrow denotes that the interaction directly propagates to the model (M), and model evolution can be expressed as:

$$\frac{dM}{dt} = f(D, E(K)). \tag{5}$$

where $f(\cdot)$ represents the learning or update process that maps interaction-driven inputs to changes in the model state, such as retraining, fine-tuning, or constraint-based optimization. Compared to IVML systems for improving training data, these configurations often incur significantly higher computational costs due to more frequent iterative retraining or re-optimization.

3.4 Interactions for Model Evaluation and Understanding

A third interaction target focuses on evaluating and understanding model behavior, where the model remains fixed and interaction is used to inspect predictions, explanations, uncertainty, failure cases, or internal representations [18, 26, 39, 73]. In this configuration, the machine learning model functions as a static generator of results rather than a dynamic target for refinement:

$$\frac{dM}{dt} = 0. \tag{6}$$

User interaction is confined to manipulating the visualization specification S to gain insights K (governed by Equations 2 and 3), but it does not flow back to modify the internal model parameters nor training data. This layer is primarily governed by human cognitive constraints, such as attention and memory, as well as physical display limits such as available pixels and visual bandwidth.

4 Cost-Benefit Models

IVML systems involve iterative feedback loops that introduce both analytical benefits and additional operational costs. To evaluate this trade-off, we build upon the economic model of visualization proposed by Van Wijk [63] and extend its cost-benefit reasoning to accommodate machine learning workflows. In contrast to traditional static visualization pipelines, IVML systems may dynamically update the underlying model state during interaction, introducing additional computational cost from retraining as well as cognitive effort as users reinterpret evolving results.

In this section, we formalize these cost and benefit components. Our formulation synthesizes prior cost-benefit reasoning from visualization, HCI, machine learning, and decision analysis, adapting these perspectives to the visual interaction loops and model-update dynamics of IVML systems. The key variables used in the proposed cost-benefit model are summarized in Table A.1. Rather than aiming for precise economic prediction, the model provides a structured parameterization of both human and computational factors, enabling systematic reasoning about when an interactive workflow may be economically justified

relative to a non-interactive baseline. At a high level, an IVML workflow is economically justified when its expected total benefit exceeds its expected total cost over the analysis horizon T,

$$B^{total}_{IVML}(T) \geq C^{total}_{IVML}(T) \tag{7}$$

We next decompose the cost and benefit sides of this inequality and then discuss how the active terms vary across IVML configurations. Our formulation rests on the following assumptions:

- **Heterogeneous Interaction Costs:** Cognitive and operational loads are not static. The perceptual cost (C_P) varies dynamically with the complexity of the intervention (ω), recognizing that a single high-effort cognitive deduction incurs a structurally different cost than multiple low-effort routine interactions.
- **Additive Integration:** While instantaneous cognitive and computational rates may fluctuate over time (e.g., compounding mental fatigue), the cost and benefit components are independent and additive when integrated over the total session duration T. This assumption allows distinct forms of effort and value to be summed over the session duration.
- **Expected Monotonicity:** The expected trajectories of data quality $Q(t)$ and the task-relevant quality induced by model state $M(t)$ improve monotonically with supervised human refinement, i.e., $\frac{dQ}{dt} \geq 0$ and $\frac{dM}{dt} \geq 0$ in expectation. While M represents the high-dimensional model state, we use the notation $\frac{dM}{dt} \geq 0$ as shorthand for expected improvement in model quality, rather than a monotonic increase of raw model parameters. We assume human interventions generally reduce system entropy rather than introduce adversarial degradation.
- **Bounded Utility of Knowledge:** The acquisition of actionable insight ($\Delta K(T)$) and the resulting risk reduction are strictly bounded. Once a conceptual failure or data anomaly is fully understood ($\Delta K(T) = 1$), further exploration of that specific artifact yields zero marginal benefit, driving the overall benefit curve toward an asymptotic limit.
- **Cognitive Re-alignment Penalty:** Model evolution could impose perceptual overhead. In dynamic systems where interaction alters the underlying logic ($\frac{dM}{dt} \neq 0$), updating the analyst's mental model to reconcile with the new system state incurs an explicit 'cognitive reset' cost (C_R) distinct from initial exploration.
- **Rational Economic Agency:** We assume the overarching objective of deploying IVML is economic and operational efficiency. While we recognize that practical deployment may be subject to external constraints, such as hard time limits or task quotas, we model the analyst or system protocol as terminating the interactive loop when the instantaneous marginal cost exceeds the marginal benefit, which yields a theoretical foundation for optimal stopping criteria.

4.1 Costs of IVML Systems

The total cost of deploying an IVML system includes both the development cost of building the system and the runtime cost incurred during interactive analysis sessions. We decompose the total cost as:

$$C^{total}_{IVML}(T) = C^{dev}_{IVML} + C^{runtime}_{IVML}(T) \tag{8}$$

where C^{dev}_{IVML} represents the one-time development cost of building the interactive system, and $C^{runtime}_{IVML}$ captures the operational cost (e.g., consumed human and computational resources) accumulated during interactive sessions. While our main focus is on runtime trade-offs, we include the fixed development cost (C^{dev}_{IVML}) to reason about amortized viability in deployment settings. In academic settings, this term can be treated as contextual rather than as the primary criterion for research value. Over an interaction horizon T with N_{iter} interaction-triggered model updates, the operational cost of an IVML session is defined as:

$$\begin{aligned} C^{runtime}_{IVML}(T) = &\int_0^T \left(C_P(t,\omega) + C_E(t) + C_S(t) + C_V(t)\right) dt \\ &+ C_{M,0} + \sum_{j=1}^{N_{\text{iter}}} [C_{M,j} + C_R(M_j, M_{j-1}, t_{\text{train},j}; \tau)]. \end{aligned} \tag{9}$$

Perceptual and Cognitive Cost C_P: the complexity-weighted perceptual and cognitive cost associated with interpreting visual representations and reasoning about system behavior [5]. It is a function of time t and intervention complexity ω, where ω captures the user's cognitive load in planning and executing a specific category of interaction [35]. This term also captures cognitive overhead introduced by task switching [71], which could occur when users alternate between inspecting visualizations, updating system specifications, and reasoning about model behavior.

Interaction and Exploration Cost C_E**:** this captures the effort associated with actively probing the system through user interactions [76]. This includes actions such as selecting data points, adjusting parameters, navigating views, or issuing exploratory queries to examine alternative hypotheses. These interactions require both motor effort and cognitive planning as users decide which actions to perform next in order to advance their analytical goals.

ML Training Cost $C_{M,j}$: this reflects the computational resources required for the jth training or retraining operation [41]. $C_{M,0}$ denotes the cost of the initial training cycle performed before the first interactive update, and this cycle produces the baseline model state M_0. In many IVML systems, interactions such as label correction, parameter adjustment, or constraint specification may trigger a partial or full model update. For instance, when using GPUs to train a model:

$$C_{M,j} = N_{\text{GPU},j} \cdot t_{\text{train},j} \cdot c_{\text{GPU},j}. \tag{10}$$

where $N_{\text{GPU},j}$ is the hardware allocation for the jth training, $t_{\text{train},j}$ is the training time for the jth cycle, and $c_{\text{GPU},j}$ is the corresponding unit cost of compute. $N_{\text{iter}} = 0$ for systems designed to evaluate or understand model behavior without modification. In practice, $C_{M,j}$ may also include costs associated with CPUs, memory, storage, data transfer, or other computational resources. This cost also depends on factors such as dataset size, model complexity, optimization procedures, and hardware resources.

Cognitive Re-alignment Cost C_R: This term acts as a penalty for model updates, reflecting the cost of re-understanding the system after an intervention. When interaction triggers model retraining ($\frac{dM}{dt} \neq 0$), the user incurs a latent "cognitive reset" cost. This effect is consistent with prior work on task interruption and resumption, which shows that users incur additional cognitive overhead when reconstructing context after a disruption, beyond the cost of the primary task [28]. To capture the impact of interactive machine learning, we decompose this cost into a state-driven evaluation component and a time-driven latency penalty:

$$C_R(M_j, M_{j-1}, t_{\text{train},j}; \tau) = C_{\text{eval}}(M_j, M_{j-1}) + P_{\text{resume}}(t_{\text{train},j}; \tau) \tag{11}$$

Here, C_{eval} represents the evaluation cost, scaling with the magnitude of change between the previous model state (M_{j-1}) and the newly updated state (M_j). It captures the visual and logical effort required to interpret the new outputs. Independent of this state change is P_{resume}, a context-resumption penalty modulated by the model training latency ($t_{\text{train},j}$) [38]. If the wait time exceeds the human attention threshold (τ), we model C_R as increasing non-linearly to reflect the additional effort required to resume the task and recover the prior analytical context [1, 45]. Even when $t_{\text{train},j} \leq \tau$, latency can still introduce workflow friction as analysts wait for model updates and interpret asynchronous system responses [38].

Visual Specification Management Cost C_S: This cost arises from constructing, modifying, and maintaining the visualization specification that governs how data is visually represented [44, 63]. C_S includes defining visual encodings, selecting attributes to display, adjusting mappings, and managing view configurations such as filtering, aggregation,

or layout [55]. Unlike exploratory interactions, these operations require deliberate reasoning about how information is presented, and may accumulate as the analytical session grows more complex.

Visualization Generation Cost C_V: this represents the computational overhead required to produce visual representations of intermediate or final results. This includes data aggregation, layout computation, rendering pipelines, and the updating of graphical encodings in response to interaction [33, 44, 63]. In large-scale or high-dimensional datasets, visualization generation may require substantial preprocessing or GPU rendering resources to maintain interactive responsiveness.

This decomposition mirrors the explicit overhead accounting used in prior inequality-based analyses of interactive learning systems.

4.2 Benefits of IVML Systems

While interactive analysis introduces additional costs, it can also generate substantial benefits by integrating human knowledge into the learning process. In IVML systems, interactions include exploring or modifying the data, model, or both, which results in changes in system state and in the analyst's understanding. Across different IVML systems, the benefits typically arise through four general mechanisms: 1) reducing downstream operational costs B_{ds}, 2) improving model performance utility B_{model}, 3) mitigating failure risks B_{risk}, and 4) increasing domain or machine learning knowledge B_K. Although different IVML configurations emphasize these mechanisms differently, they provide a common perspective on how interaction generates value in machine learning workflows. We generalize the total benefit (B^{total}_{IVML}) as the sum of these four components:

$$B^{total}_{IVML}(T) = B_{ds}(T) + B_{model}(T) + B_{risk}(T) + B_K(T) \tag{12}$$

Benefit of Reduction of Downstream Operational Cost B_{ds}: This benefit reflects the reduction of routine operational costs resulting directly from improvements in data quality (Q) and model state (M) [54, 58, 77]. These routine costs include manual verification, debugging, correction, and other recurring operational tasks. We let $\Delta Q(T)$ and $\Delta M(T)$ denote the cumulative improvements in data and model over the interactive session duration T. Under the Expected Monotonicity assumption, these terms are interpreted as non-negative expected improvements; if interaction is expected to degrade data or model state, the corresponding benefit terms should be reduced, omitted, or modeled as costs. As the data becomes cleaner and the model becomes more reliable, the subsequent human effort required for manual intervention progressively decreases.

$$B_{ds}(T) = \Delta Q(T) \cdot C^Q_{ds} + \Delta M(T) \cdot C^M_{ds} \tag{13}$$

Here, C^Q_{ds} and C^M_{ds} represent the specific unit costs of downstream operations saved per unit of improvement in data and model, respectively. B_{ds} is strictly associated with continuous and repeatable cost reduction driven by these incremental system refinements. It does not account for the value of preventing discrete failure events.

Benefit of Model Performance Utility B_{model}: This benefit captures the domain-specific business value or operational utility generated directly by improvements in the model's predictive capabilities [30, 64]. We let $\Delta M(T)$ denote the cumulative improvement in the model state over the interactive session, and U_M denote the domain-specific utility factor for a unit increase in model performance. The benefit is defined as:

$$B_{model}(T) = \Delta M(T) \cdot U_M \tag{14}$$

B_{model} represents the intrinsic, vertical economic surplus created by the model's enhanced output. For example, in a medical domain, U_M reflects the high intrinsic value of increased diagnostic accuracy leading to better patient outcomes, which scales independently of the hourly labor rate. While we model this utility using a constant linear factor for illustrative baseline calculations, real-world utility gains are frequently non-linear. Because the framework aggregates these terms additively at the macro level, researchers can directly substitute a non-linear, domain-specific utility function for U_M without changing the main equations.

Benefit of Risk Reduction B_{risk}: This benefit arises from preventing failure events that are not addressed by continuous system improvement [20, 59]. Such failures may result from model errors, biases, or unexpected behaviors that are not reflected in standard performance metrics. We use C_f to denote the cost of a failure and P_{fp} to denote the probability that interactive analysis prevents such a failure.

$$B_{risk} = \Delta K(T) \cdot C_f \cdot P_{fp} \tag{15}$$

In IVML, $\Delta K(T) \in [0,1]$ represents the extent to which discovered evidence is internalized as understanding over the interaction horizon T. B_{risk} is associated with failure avoidance and expected loss reduction. It does not include routine operational cost captured by B_{ds}.

Benefit of Knowledge Gain B_K: This term captures the intrinsic value of knowledge gained during interaction, exploration, verification, and reasoning processes [53, 63]. The value does not depend on changes to the system state or downstream outcomes.

$$B_K = \Delta K(T) \cdot U_K \tag{16}$$

B_K only accounts for knowledge itself that is not already captured by improvements in system performance or risk reduction.

4.3 Configuration-Specific Instantiations of IVML Systems

The cost and benefit terms defined in Sections 4.1 and 4.2 describe a general model for IVML systems. The core objective is to ensure that interaction produces sufficient value to justify its cost (Equation 7). However, the activation of specific terms depends on the analytical task, the system's objective, and the interaction target. Moreover, advanced systems may activate multiple mechanisms at the same time. As such, our cost-benefit framework provides the flexibility to support diverse IVML scenarios. For instance, data labeling or feature refinement interfaces primarily rely on downstream operational savings (B_{ds}) to offset the computational costs of iterative retraining ($\sum_{j=1}^{N_{\text{iter}}} C_{M,j}$). Similarly, model-steering systems balance the utility of improved predictive performance (B_{model}) against retraining costs. Conversely, for systems designed to evaluate or understand model behavior without modification ($N_{\text{iter}} = 0$), the evaluation shifts to balancing human perceptual effort against failure prevention (B_{risk}) and knowledge gain (B_K). By selecting relevant cost and benefit components, practitioners can instantiate Equation 7 with customized economic thresholds, allowing them to evaluate the operational trade-offs of their specific system architecture.

5 Instantiating the Cost-Benefit Model

To illustrate how the proposed framework can be instantiated, we apply it to existing IVML systems. Following prior analytical and scenario-based approaches in visual analytics [14] and risk-informed economic modeling [65], we analyze representative systems from the literature rather than conduct a user study on a single interface. We select two case studies based on three criteria: representativeness of distinct interaction-target configurations, availability of reported empirical quantities for parameter grounding, and contrast in cost-benefit structures. For each system, we apply the same three-step procedure. This procedure translates the general terms from Section 4 into system-specific quantities and measurable proxies.

1. **Variable Mapping**: We map each interaction loop to our cost-benefit framework. To maintain tractability, we absorb C_S and C_V into aggregate operational costs when not dominant. We define the knowledge-to-action translation ($\Delta K(T)$) as a bounded parameter, setting $\Delta K(T) = 1$ for upper-bound benefit estimates; while practical gains are fractional due to prior knowledge ($K_{start} > 0$) and incomplete final understanding ($K_{end} < 1$), this only scales the benefit without altering the model structure. Finally, we capture interaction complexity (ω) via proxies derived from reported task time and effort.
2. **Break-even Analysis**: Drawing on applied information economics [27], we introduce *proxy anchoring* to map abstract variables to financial baselines and estimate macro-level economic

viability. Unlike traditional break-even analyses driven by production volume or time, IVML viability is domain-dependent. For example, a 1% accuracy gain yields a significantly different economic value in healthcare versus entertainment. We thus treat domain-specific parameters, such as model utility (U_M) or failure cost (C_f), as independent variables, allowing us to determine the threshold where expected benefits exceed costs.

3. **Marginal Efficiency Analysis**: This analysis evaluates the temporal evolution of costs and benefits. While break-even analysis determines whether interaction is economically justified, marginal efficiency analysis identifies the optimal stopping point for the interaction session. For consistency, we define the total extractable benefit using the system's break-even value (i.e., the point where total expected benefit equals total system cost). We then distribute this value over time to model how it is progressively realized. By parameterizing marginal cost and benefit curves using empirical boundary conditions and sensitivity analysis, their intersection identifies the point of diminishing returns, where the marginal cost of further refinement exceeds the marginal rate of return.

Together, these steps connect system structure, economic viability, and stopping conditions within a single instantiation. All numerical values are proxy-based and are intended to support relative reasoning rather than exact prediction.

5.1 Case 1: Interactions for Improving Data and Models

A representative example of combined interaction targets is MitoVis, a visual analytics system for end-to-end neuronal mitochondria analysis. We selected MitoVis [15] as it modifies both data and models while providing the empirical measures necessary to instantiate our framework. In this end-to-end visual analytics system for neuronal mitochondria, users interactively proofread segmentation masks ($dD/dt \neq 0$) and fine-tune the neural network via active learning ($dM/dt \neq 0$). This dual-target interaction activates multiple cost and benefit terms.

Variable Mapping In MitoVis, the analyst's interaction consists of manually correcting segmentation errors in the model output, which are then incorporated as updated training data. This maps directly to C_P and C_E. The system also incurs an initial computation and training cost $C_{M,0}$, while these corrections trigger model fine-tuning and incur $C_{M,j}$ at each subsequent model update. In addition, model updates require the analyst to reinterpret the outputs, which introduces C_R. Prior HCI and task interruption studies suggest that the attention threshold τ for maintaining analytical context is around 10 seconds [34, 45]. Since MitoVis reported fine-tuning latency of 23s, its retraining latency is non-negligible for cognitive re-alignment. We therefore include C_R and approximate it as a constant per iteration, capturing both evaluation cost (C_{eval}) and latency-induced resumption cost (P_{resume}). For simplicity, we treat C_S and C_V as minor terms and absorb them into the interaction cost. The interaction complexity parameter ω is captured through time-based proxy estimates. On the benefit side, the stated goal of MitoVis is to improve model performance for future use and to accelerate the end-to-end analysis process for domain experts. We therefore focus on downstream operational savings (B_{ds}) and model performance utility (B_{model}) in this case study.

Break-even Analysis The analysis is based on reported results from the MitoVis study. We use the hMFF dataset evaluated by domain expert U1 to anchor the runtime comparison, where the traditional manual annotation workflow required 191.4 minutes and the IVML session concluded in 19.0 minutes. For model improvement, we use the reported performance of the neurite-wise classification network, where the accuracy improved from 77.0% to 87.1% over three interactive fine-tuning iterations, giving $\Delta M(T) = 10.1\%$. For analysis, we model the reported three fine-tuning iterations as an aggregated three-session workflow, with each refinement session taking approximately 19 minutes ($t = 19 \times 3 = 57$ minutes). We also adopt the reported system processing times, including 32 seconds for the initial automated processing and 23 seconds per interactive fine-tuning iteration. We combine these reported results into a representative and consistent instantiation scenario. The runtime cost for this case is defined as:

$$C_{IVML}^{runtime} = \int_0^T (C_P(t,\omega) + C_E(t))\,dt + C_{M,0} + \sum_{j=1}^{N_{\text{iter}}} [C_{M,j} + C_R(M_j, M_{j-1}, t_{\text{train},j}; \tau)]. \tag{17}$$

Using industry salary proxies for machine learning engineers [37], we value expert labor at about \$120/hr (\$2.00/min). We also use \$3.00/hr as the hourly cost of cloud GPU computation, following the \$0 to \$7 price range of NVIDIA GPU rental [36]. For the human-side cost, the direct labor component ($C_P + C_E$) captures the domain expert's continuous observation and interaction. Thus, we anchor the 57.00-minute IVML session to the expert's temporal value, which yields

$$C_P + C_E = 57.00 \text{ mins} \times \$2.00/\text{min} = \$114.00.$$

For the machine computation cost, we account for the model's full active computational cycle: 32 seconds for the initial automated processing ($C_{M,0}$), plus three subsequent interactive fine-tuning iterations at 23 seconds each ($C_{M,j}$ for $j \geq 1$). Assuming identical GPU allocation and unit price across the initial and the subsequent training operations, we obtain

$$C_{M,0} + \sum_{j=1}^{N_{\text{iter}}} C_{M,j} = (32 + 3 \times 23) \text{ s} \times \$3.00/3600 \text{ s} \approx \$0.08.$$

We also quantify the cognitive re-alignment cost (C_R). For tractability, we approximate C_R as constant across iterations, assuming comparable update magnitude and latency over the three reported refinements. To account for the overhead of re-engaging with the updated model, we assume a 1.5-minute cognitive re-alignment period per steering iteration. This value is an order-of-magnitude estimate of re-alignment overhead rather than a precise empirical constant. Anchoring this 1.5-minute period to the expert's temporal wage, the three model updates introduce an estimated cognitive overhead of

$$C_R = 3 \text{ updates} \times 1.5 \text{ mins/update} \times \$2.00/\text{min} = \$9.00.$$

As a result, the approximate runtime cost is

$$C_{IVML}^{runtime} = \$114.00 + \$0.08 + \$9.00 = \$123.08,$$

and the total cost is

$$C_{IVML}^{total} = \$123.08 + C_{IVML}^{dev}.$$

The total expected economic benefit (B_{IVML}^{total}) generated across the three-iteration sessions is formulated as the sum of the operational savings achieved during the session and the downstream utility of the improved model:

$$B_{IVML}^{total} = \left[\Delta Q(T) \cdot C_{ds}^{Q} + \Delta M(T) \cdot C_{ds}^{M}\right] + [\Delta M(T) \cdot U_M] \tag{18}$$

While we treat $\Delta M(T) \cdot U_M$ as a linear utility for baseline calculations, this term can be adapted to specific domain requirements. In mission-critical settings, where performance gains near deployment thresholds are disproportionately valuable, practitioners can replace this linear approximation with domain-specific nonlinear or threshold-based utility functions without altering our overarching economic equations.

In MitoVis, user interaction has two coupled effects: it corrects local data (ΔQ) and provides training signals that improve the model (ΔM). In the reported study, a 19.0-minute IVML session is compared against a 191.4-minute manual workflow on the same dataset. Because our analysis aggregates the three reported refinement iterations into a three-session workflow, we use the corresponding manual effort at the same aggregation level as a proxy for joint operational savings. Under this aggregation, the manual baseline becomes 574.2 minutes (191.4×3), representing the expert effort required to produce comparable corrected outputs and supervision signals across the three-session analysis. Anchored to the expert temporal proxy of \$2.00/min, this yields

$$\left[\Delta Q(T) \cdot C_{ds}^{Q} + \Delta M(T) \cdot C_{ds}^{M}\right] = 574.20\text{mins} \times \$2.00/\text{min} = \$1148.40,$$

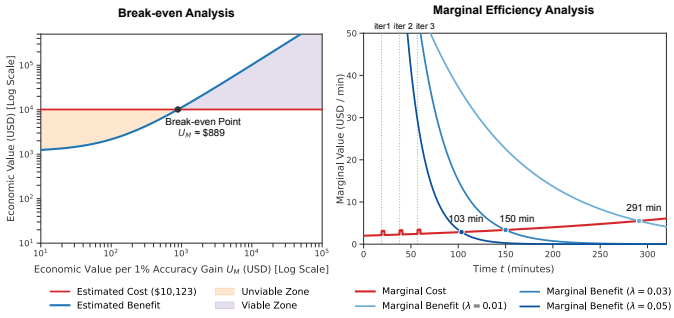

Fig. 2: Case Study 1 - MitoVis [15]. The break-even analysis (the left plot) shows the development-cost threshold and the domain-specific utility factor U_M required for the aggregated three-session workflow to break even. The marginal efficiency analysis (the right plot) under different information-scent settings illustrates how retraining-induced cost increases affect the stopping point.

which gives the baseline of the total benefit in Figure 2. Beyond this immediate operational savings term, the improved model contributes future scalable utility through $\Delta M(T) \cdot U_M$. Here, the explicit performance improvement is the 10.1% gain in accuracy. The valuation parameter U_M represents the domain-specific economic value of a 1% accuracy gain when deployed at scale. Substituting these values gives

$$B_{IVML}^{total} = \$1148.40 + (10.1 \times U_M),$$

which is plotted as the ascending solid blue curve in Figure 2.

While the proxy-based joint labor savings ($1,148.40) offset the $123.08 runtime cost, assessing broader viability requires incorporating the upfront development cost (C_{IVML}^{dev}). Assuming $C_{IVML}^{dev} =$ $10,000 [29], which also serves as a proxy for a few months of a Computer Science PhD student's research stipend and associated institutional overhead, the total system cost (C_{IVML}^{total}) for the aggregated three-iteration workflow becomes $10,123.08. This fixed development cost carries different implications across contexts: academia often prioritizes methodological contributions over immediate financial returns, whereas industry amortizes such expenses across repeated deployments. Therefore, we treat the PhD-stipend estimate purely as an illustrative prototyping proxy for our break-even calculation. The primary focus of this case study remains on runtime trade-offs, specifically, balancing interaction effort and retraining latency against downstream savings and model improvement.

By equating the total expected benefit to this total cost ($B_{IVML}^{total} = C_{IVML}^{total}$), we obtain an illustrative break-even threshold of $U_M \approx 889$. As shown in Figure 2, the intersection of the benefit curve and the total cost line divides the space into two regions. The unviable zone indicates domains in which the expected benefit remains below the total cost, whereas the viable zone indicates domains in which the expected benefit exceeds it. Under this parameterization, the three-iteration workflow breaks even when a 1% accuracy gain is worth on the order of $889. This threshold is defined by the model utility factor U_M, capturing the domain-specific economic value of performance gains. Consequently, the break-even condition is not anchored to a fixed workload, but depends on the value generated per unit of model improvement in a given application. This threshold shifts rightward (requiring a higher U_M) if development costs increase, and leftward as the system is amortized across additional workflows. While our present analysis focuses on a single aggregated session, repeated use can strengthen the economic case for MitoVis, as operational savings recur and model improvements yield compounding value over time.

Marginal Efficiency Analysis Since the break-even analysis is now anchored to the aggregated three-iteration workflow (57 minutes in total), the marginal analysis extends this setting to model longer interactions and identify the model-based optimal stopping point. On the cost side, MitoVis requires continuous steering and repeated model retraining. We anchor the initial operational cost (c_0) to $120/hour ($2/min), representing expert labor for interactive tuning. As the session progresses, cognitive fatigue increases the interaction cost over time. Prior vigilance studies show that sustained-attention performance declines with time on task, with measurable degradation emerging within the first 20–30 minutes [48, 68]. Mackworth's experiments [40] describe an approximate 10-15% drop in detection performance over the first 30 minutes. As fatigue also increases mental workload, we use performance decline as a proxy for fatigue-induced increases in cognitive cost [68]. We adopt an optimistic lower-bound estimate: assuming a 10% performance drop in the 30 minutes, which corresponds to a cost increase of $1/0.9 \approx 1.11$. We use this value as a calibration reference rather than a direct empirical estimate of fatigue dynamics. We therefore solve

$$e^{30\alpha} = 1.11,$$

which gives

$$\alpha = \frac{\ln(1.11)}{30} \approx 0.00348.$$

This yields the baseline marginal cost

$$MC_{\text{base}}(t) = 2e^{0.00348t}.$$

Since we assume that retraining occurs at the end of each 19-minute interaction session, we add local cost increases at $t_k \in \{19, 38, 57\}$ minutes, for $k = 1, 2, 3$. Here, t_k denotes the occurrence time of the kth retraining event on the interaction timeline. Each retraining event introduces two forms of additional overhead: a machine retraining cost and a cognitive re-alignment cost. The per-update machine cost is anchored to the reported 23-second fine-tuning time and the assumed GPU price of $3/hr, which yields

$$23 \text{ s} \times \$3/3600 \text{ s} \approx \$0.019.$$

The per-update cognitive re-alignment cost is anchored to the 1.5-minute recovery period used in the break-even analysis, valued at $2/min, which yields

$$1.5 \text{ min} \times \$2/\text{min} = \$3.00.$$

Thus, the total additional cost per retraining event is approximately $3.019. To incorporate this overhead into the marginal model without introducing additional decay assumptions, we distribute this cost uniformly over a short post-update window of 3 minutes. This value is anchored to empirical estimates of retraining-related disruption. Each retraining event introduces approximately 23 seconds of model update latency (≈ 0.38 minutes) and 1.5 minutes of cognitive re-alignment cost. These components suggest a post-update disruption period of approximately 2 minutes. We adopt a slightly conservative 3-minute window to capture these costs, alongside the additional time required for the analyst to re-orient, inspect updated outputs, and resume exploration. This gives a stepwise marginal cost increment $\Delta_C = \frac{3.019}{3} \approx 1.01$ USD/min. The total marginal cost is therefore modeled as

$$MC(t) = 2e^{0.00348t} + s_C(t),$$

where $s_C(t)$ represents the additional cost induced by retraining events. Specifically, for each retraining time t_k, we assume a short post-update window of 3 minutes during which marginal cost is elevated:

$$s_C(t) = \begin{cases} \Delta_C, & t_k \le t < t_k + 3 \\ 0, & \text{otherwise.} \end{cases}$$

Under this formulation, retraining does not appear as an instantaneous spike, but as a short post-update phase in which marginal cost is elevated due to training overhead and cognitive re-engagement.

On the benefit side, $MB(t)$ is governed by the marginal probability of identifying useful patterns that improve predictive performance. We ground this formulation in Information Foraging Theory [49], which models search within an information patch as a process with diminishing returns over time. This perspective is closely related to the Marginal Value Theorem [12], where the rate of gain decreases as search continues within the same region. Consistent with this idea, we represent

the cumulative probability of achieving the target model improvement using a saturating function $1-e^{-\lambda t}$. We use this exponential form as a simple proxy for diminishing marginal gains over time, rather than as a claim about the exact shape of all IVML benefit trajectories. In this formulation, λ represents the efficiency of information acquisition, which we relate to the notion of information scent in prior work [50]. It captures how quickly the interface exposes useful information to the analyst. To keep this analysis consistent with the updated break-even analysis, we set the total extractable benefit to the break-even system value of $10,123.08, which represents the total value that must be realized for the aggregated three-iteration workflow to break even under this parameterization. We then write the marginal benefit as

$$MB(t) = 10123.08 \cdot \lambda e^{-\lambda t}.$$

Unlike the cost model, we do not introduce retraining-triggered pulses or steps in the marginal benefit formulation. In this analysis, retraining is treated as a mechanism that helps realize the pre-existing total benefit, rather than as an event that creates new economic value beyond that total. The benefit of retraining is therefore already represented within the overall extractable value budget of $10,123.08.

In this formulation, the information scent parameter λ captures the efficiency of the system's visualization and interaction design. Interfaces with clear visual encodings, effective guidance, and low clutter expose useful information more quickly, yielding a higher λ. Conversely, complex or disorganized interfaces increase cognitive load and slow down exploration, leading to a lower λ. Targeted interactions, such as filtering, highlighting, and recommendation, can also increase λ by directing the analyst's attention to relevant data. Together, these factors affect the rate of insight discovery. To model these design qualities, we evaluate three representative settings ($\lambda \in \{0.01, 0.03, 0.05\}$). In the $1-e^{-\lambda t}$ model, this range spans plausible information acquisition rates: larger values would imply rapid saturation, whereas smaller values push diminishing returns beyond the considered session lengths.

Because the retraining-adjusted marginal cost is piecewise rather than fully smooth, we determine the stopping point numerically as the first intersection between $MB(t)$ and $MC(t)$. Under this formulation, the stopping point occurs at approximately 291 minutes for $\lambda = 0.01$, 150 minutes for $\lambda = 0.03$, and 103 minutes for $\lambda = 0.05$. This sensitivity analysis shows how the resulting stopping point varies with the information scent parameter λ. The stopping point is determined by the balance between value discovery and cost accumulation, defined by the intersection of marginal benefit and marginal cost. Weaker information scent leads to slower value discovery and a later intersection, whereas stronger information scent accelerates value discovery and results in an earlier stopping point. Beyond this point, continued interaction yields diminishing returns, where additional cost exceeds marginal benefit.

Constructive IVML systems introduce short post-update phases of elevated marginal cost following retraining. These phases reflect both computational latency and cognitive re-alignment, and their timing and frequency directly affect the stopping condition. More frequent retraining introduces repeated cost increases that move the stopping point earlier, whereas less frequent retraining reduces interruptions and allows interaction to continue longer. In this sense, retraining improves model quality but also introduces interaction friction, making cost-effectiveness sensitive to both update latency and the analyst's ability to re-engage with evolving outputs.

5.2 Case 2: Interactions for Model Evaluation and Understanding

Next, we consider the visual analytics system of Ma et al. [39], an IVML system focused on model evaluation, debugging, and explainability without modifying training data or model parameters. We selected this system because its costs and benefits center on failure prevention and knowledge gain rather than cumulative improvement. The system is designed to explore data poisoning vulnerabilities in adversarial machine learning. The analysis therefore uses the financial cost of model failure (C_f) as the primary domain-dependent variable.

Break-Even Analysis To evaluate economic viability, we ground the break-even analysis in a representative adversarial vulnerability audit

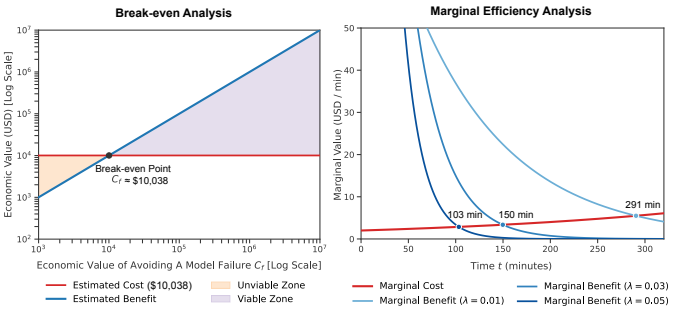

Fig. 3: Case study 2. The break-even analysis (the left plot) shows the failure cost threshold C_f required for a single evaluation session to break even under the upper-bound risk-mitigation scenario. The marginal efficiency analysis (the right plot) under different information-scent settings illustrates how the stopping point varies when value is realized through failure detection rather than model refinement.

documented by Ma et al. [39]. For cross-case consistency, we use a normalized interaction duration of 19 minutes, matching the MitoVis case study. This choice is intended to support comparison across instantiations rather than to represent a universal audit duration. It is also broadly consistent with the original system context reported by Ma et al. [39], where four domain experts participated in a 1.5-hour interview session that included a system introduction, a guided tutorial, and free exploration of the interface. We use the same expert labor rate of $120 per hour and the same development cost baseline of $10,000. For this case, the evaluated model is treated as already trained and fixed, so its initial training cost falls outside the accounting boundary ($C_{M,0} = 0$) and no interaction-triggered model update occurs ($N_{iter} = 0$). The combined perceptual and exploratory cost is therefore

$$C_P + C_E = 19.00 \text{ mins} \times \$2.00/\text{min} = \$38.00,$$

which serves as the runtime cost ($C_{IVML}^{runtime}$). Thus,

$$C_{IVML}^{total} = C_{IVML}^{dev} + C_{IVML}^{runtime} = \$10,000 + \$38.00 = \$10,038.00.$$

The benefit in this case relies on risk mitigation. For an upper-bound instantiation, we set $\Delta K(T) = 1$ and $P_{fp} = 1$, corresponding to a case in which the analysis fully identifies and prevents the relevant failure. Under this optimistic assumption, the total expected benefit reduces to the cost of failure:

$$B_{risk} = \Delta K(T) \cdot P_{fp} \cdot C_f = C_f.$$

Equating this benefit with the total system cost gives an illustrative break-even threshold of

$$C_f = \$10,038.00.$$

As shown in Figure 3, this threshold divides the economic space into two regions. The unviable zone corresponds to settings in which the expected cost of model failure is below $10,038.00. The viable zone corresponds to settings in which the expected benefit of risk reduction exceeds the total system cost. Under this parameterization, the evaluation becomes economically justified only when the domain-specific cost of adversarial failure is sufficiently high. Still, the development cost of the system can also greatly affect the break-even point. Similar to the MitoVis case study, this analysis focuses on a single evaluation session. In practice, evaluation-oriented systems are often deployed across multiple models or datasets, accumulating value through successive opportunities for failure prevention. However, unlike constructive workflows where benefits compound on a single model instance, evaluative benefits scale with the number of discrete tasks. The economic viability of these systems thus depends more on deployment frequency than on iterative refinement within a single session.

Marginal Efficiency Analysis We reuse the same formulation as in the MitoVis case study: break-even analysis defines the total value, while marginal analysis models how this value is accrued over time. We adapt this formulation to the evaluation-centric IVML setting. On

the cost side, this system does not trigger model retraining, so the cost is governed solely by perceptual and exploratory effort. Using the same labor baseline and fatigue parameterization as in Section 5.1, the marginal cost is

$$MC(t) = 2e^{0.00348t}.$$

On the benefit side, we reuse the same exponential form as a simple proxy for diminishing discovery over time. The cumulative probability of failure prevention is parameterized as $P_{fp}(t) = 1 - e^{-\lambda t}$, which yields the marginal benefit

$$MB(t) = 10038 \cdot \lambda e^{-\lambda t},$$

where the total extractable value is set by the break-even threshold of \$10,038.00. As such, equating $MB(t)$ and $MC(t)$ gives the optimal stopping point

$$t^* = \ln(5019 \cdot \lambda)/(\lambda + 0.00348).$$

As in the MitoVis case study, we evaluate three information-scent settings. As illustrated in Figure 3, the stopping point is $t^* \approx 291$ minutes, 150 minutes, and 103 minutes for $\lambda = 0.01, 0.03, 0.05$ respectively.

This analysis illustrates that under a fixed marginal cost, the optimal stopping point (t^*) is governed by the rate of failure discovery. A larger λ accelerates this value realization, shifting the intersection earlier, whereas a lower λ delays the stopping point due to slower information acquisition. Together, these case studies are not intended for direct numeric comparison. Rather, they demonstrate how distinct interaction targets activate different value channels: cumulative refinement and downstream savings in Case 1, versus repeated risk detection in Case 2. Consequently, these differences dictate distinct break-even thresholds, reuse patterns, and stopping conditions, illustrating how a unified cost-benefit framework accommodates diverse IVML reasoning.

6 Discussion and Conclusion

This work introduces a cost-benefit modeling framework for Interactive Visual Machine Learning (IVML) systems. Unlike traditional evaluations focused on isolated metrics such as accuracy or qualitative insight, our framework incorporates both human cognitive effort and computational overhead into a unified economic formulation. By structuring IVML workflows based on whether interaction modifies data, alters model state, or facilitates users' evaluation and understanding of model behavior, the framework provides a basis for reasoning about when interaction is justified, how its value is generated, and where human effort should be allocated. Through representative case studies, we demonstrate how this framework provides a structured approach for elucidating break-even assumptions, identifying dominant cost-benefit terms, and analyzing stopping conditions.

A key implication is that the economic value of interaction depends on where it intervenes in the workflow. When interaction modifies data or model state ($dD/dt \neq 0$ or $dM/dt \neq 0$), it changes the system trajectory, driving value through model performance (B_{model}) and downstream efficiency (B_{ds}) while incurring retraining ($C_{M,j}$) and cognitive re-alignment (C_R) costs. Conversely, when interaction is limited to evaluating and understanding model behavior, the model remains fixed; value is generated instead through risk reduction (B_{risk}) and knowledge acquisition (B_K). Recognizing these divergent channels highlights that interaction is not inherently beneficial-it is economically justified only when the task-specific gains outweigh the corresponding human and computational burdens.

The proposed framework is intended to complement, rather than replace, existing evaluation paradigms. Traditional methods, such as usability studies, controlled experiments, qualitative case studies, expert feedback, and algorithmic benchmarks, remain essential for measuring user experience, task performance, insight generation, model accuracy, and computational efficiency. Our framework addresses a distinct, higher-order question: whether the observed improvements justify the additional human, computational, and development costs introduced by interaction. It provides a unifying cost-benefit structure for translating traditional empirical measurements into a broader account of the value of IVML systems. Furthermore, the framework can be applied modularly under realistic project constraints: researchers may instantiate only the terms most relevant to their system and use measured values, literature baselines, system logs, or sensitivity ranges for the remaining terms. This modular use allows the VIS community to reason about system-level cost-benefit trade-offs without requiring every IVML project to conduct a full economic study. The break-even and marginal-efficiency analyses used in our case studies are therefore illustrative rather than exhaustive; they show how practitioners can instantiate the framework by selecting cost and benefit terms appropriate to their own systems, rather than prescribing a fixed evaluation procedure in advance.

Beyond evaluation, the framework can inform IVML system design across planning, development, and deployment. Before a system is built, it helps compare candidate interaction targets; during design, it informs choices regarding retraining frequency, update latency, and the allocation of human effort; and after deployment, it can be instantiated using observed logs, task times, compute traces, and outcome measures. By framing interaction as an economic trade-off rather than merely an added feature, the framework turns costs into design constraints and benefits into explicit sources of value. This mapping helps designers make these trade-offs explicit and clarify which assumptions must hold for added interaction to be worthwhile.

This work has several limitations. First, the framework remains a high-level simplification of operational reality. Although proxy anchoring helps instantiate the framework when deployment data are unavailable, the estimates depend on subjective, local, or domain-specific parameter choices. Because errors across multiple parameters may compound, the numerical results in our case studies should be interpreted as sensitivity-oriented illustrations rather than precise predictions. Furthermore, the approximations required to instantiate the framework vary across IVML systems. This system-specificity can make parameter selection challenging across different IVML settings. Second, some cost terms, such as specification management (C_S) and visualization generation (C_V), are simplified or absorbed into the aggregate interaction cost, which may not hold for actual systems with complex pipelines or expensive rendering. Third, the marginal efficiency analysis relies on general behavioral assumptions, such as diminishing returns and cognitive fatigue, without modeling user [42,46], visual, and interaction design-specific variation [16]. Incorporating user and design factors such as expertise and interaction strategy would improve model fidelity. Finally, the framework is demonstrated on a limited set of systems. Applying it to additional settings, such as collaborative workflows, streaming data, or post-deployment monitoring, would further test its generality across broader IVML contexts.

These limitations motivate two main directions for future work. The first is moving from proxy-based parameterization toward empirical calibration, drawing on interaction logs, retraining traces, and longitudinal deployment data from real workflows. To make this feasible, future IVML papers should report key operational metrics, such as interaction duration, iteration counts, retraining latency, compute usage, model-performance changes, downstream effort, and utility assumptions. This applied and empirical work would support reusable parameter libraries, reporting templates, and calibrated defaults, making the framework more robust and easier to instantiate. The second direction is evaluating the prediction potential of the framework itself. Our case studies show how it is instantiated, but future work should test its predictive accuracy by comparing its estimates against observed costs in real deployments. Beyond accuracy, future research should assess whether the framework helps researchers surface trade-offs, assumptions, or evaluation gaps that existing IVML taxonomies miss.

Despite these limitations, the framework provides a practical foundation for reasoning about the economics of interaction in machine learning workflows. Its value lies not in exact prediction but in structure: it lets practitioners compare alternatives, surface dominant trade-offs, and judge when interaction is worth its human and computational cost. We see this as a step toward treating interaction not as a free design choice but as an investment to be weighed carefully from the earliest stages of design.

References

[1] E. M. Altmann and J. G. Trafton. Timecourse of recovery from task interruption: Data and a model. *Psychonomic bulletin & review*, 14(6):1079–1084, 2007. doi: 10.3758/bf03193094 4

[2] S. Amershi, A. Begel, C. Bird, R. DeLine, H. Gall, E. Kamar et al. Software engineering for machine learning: a case study. In *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Practice*, ICSE-SEIP '19, pp. 291–300, 2019. doi: 10.1109/ICSE-SEIP.2019.00042 1

[3] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza. Power to the people: The role of humans in interactive machine learning. *AI magazine*, 35(4):105–120, 2014. doi: 10.1609/aimag.v35i4.2513 2

[4] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson et al. Guidelines for human-ai interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pp. 1–13. Association for Computing Machinery, NY, USA, 2019. doi: 10.1145/3290605.3300233 2

[5] E. W. Anderson, K. C. Potter, L. E. Matzen, J. F. Shepherd, G. A. Preston, and C. T. Silva. A user study of visualization effectiveness using eeg and cognitive load. *Computer Graphics Forum*, 30(3):791–800, 2011. doi: 10.1111/j.1467-8659.2011.01928.x 4

[6] J. Bernard. The human-data-model interaction canvas for visual analytics. *arXiv preprint arXiv:2505.07534*, 2025. doi: 10.48550/arXiv.2505.07534 2

[7] J. Bernard, M. Hutter, M. Zeppelzauer, D. Fellner, and M. Sedlmair. Comparing visual-interactive labeling with active learning: An experimental study. *IEEE transactions on visualization and computer graphics*, 24(1):298–308, 2018. doi: 10.1109/tvcg.2017.2744818 2

[8] J. Bernard, M. Zeppelzauer, M. Sedlmair, and W. Aigner. Vial: a unified process for visual interactive labeling. *The Visual Computer*, 34(9):1189–1207, 2018. doi: 10.1007/s00371-018-1500-3 1, 2

[9] N. Boukhelifa, A. Bezerianos, R. Chang, C. Collins, S. Drucker, A. Endert et al. Challenges in evaluating interactive visual machine learning systems. *IEEE Computer Graphics and Applications*, 40(6):88–96, 2020. doi: 10.1109/MCG.2020.3017064 1

[10] A. Cabrera, W. Epperson, F. Hohman, M. Kahng, J. Morgenstern, and D. H. Chau. Fairvis: Visual analytics for discovering intersectional bias in machine learning. In *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 46–56, 2019. doi: 10.1109/VAST47406.2019.8986948 2

[11] D. Ceneda, T. Gschwandtner, T. May, S. Miksch, H.-J. Schulz, M. Streit et al. Characterizing guidance in visual analytics. *IEEE transactions on visualization and computer graphics*, 23(1):111–120, 2017. doi: 10.1109/tvcg.2016.2598468 2

[12] E. L. Charnov. Optimal foraging, the marginal value theorem. *Theoretical Population Biology*, 9(2):129–136, 1976. doi: 10.1016/0040-5809(76)90040-X 7

[13] A. Chatzimparmpas, R. M. Martins, and A. Kerren. t-visne: Interactive assessment and interpretation of t-sne projections. *IEEE Transactions on Visualization and Computer Graphics*, 26(8):2696–2714, 2020. doi: 10.1109/TVCG.2020.2986996 3

[14] M. Chen, K. Gaither, N. W. John, and B. McCann. An information-theoretic approach to the cost-benefit analysis of visualization in virtual environments. *IEEE transactions on visualization and computer graphics*, 25(1):32–42, 2019. doi: 10.1109/TVCG.2018.2865025 5

[15] J. Choi, H.-J. Oh, H. Lee, S. Kim, S.-K. Kwon, and W.-K. Jeong. Mitovis: A unified visual analytics system for end-to-end neuronal mitochondria analysis. *IEEE Transactions on Visualization and Computer Graphics*, 30(7):3457–3473, 2024. doi: 10.1109/TVCG.2022.3233548 1, 2, 3, 6, 7

[16] A. Crisan, M. Shang, and E. Brochu. Eliciting model steering interactions from users via data and visual design probes. *IEEE Transactions on Visualization and Computer Graphics*, 30(9):6005–6019, 2024. doi: 10.1109/TVCG.2023.3322898 9

[17] R. J. Crouser, L. Franklin, A. Endert, and K. Cook. Toward theoretical techniques for measuring the use of human effort in visual analytic systems. *IEEE transactions on visualization and computer graphics*, 23(1):121–130, 2017. doi: 10.1109/tvcg.2016.2598460 2

[18] R. Duan, N. Karthik, Y. Chen, J. Shi, R. Jain, M. Yang et al. Conceptvis: Exploring and visualizing design concepts with large language models using interactive knowledge graph. *Journal of Computing and Information Science in Engineering*, 26(1):011002, 2026. doi: 10.1115/1.4069499 2, 3

[19] J. J. Dudley and P. O. Kristensson. A review of user interface design for interactive machine learning. *ACM Trans. Interact. Intell. Syst.*, 8(2), art. no. 8, 37 pp., 2018. doi: 10.1145/3185517 2

[20] C. Elkan. The foundations of cost-sensitive learning. IJCAI'01, pp. 973–978. Morgan Kaufmann Publishers Inc., SF, USA, 2001. 5

[21] A. Endert, P. Fiaux, and C. North. Semantic interaction for visual text analytics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pp. 473–482. Association for Computing Machinery, NY, USA, 2012. doi: 10.1145/2207676.2207741 2

[22] A. Endert, W. Ribarsky, C. Turkay, B. W. Wong, I. Nabney, I. D. Blanco et al. The state of the art in integrating machine learning into visual analytics. *Computer Graphics Forum*, 36(8):458–486, 2017. doi: 10.1111/cgf.13092 1

[23] P. Federico, M. Wagner, A. Rind, A. Amor-Amorós, S. Miksch, and W. Aigner. The role of explicit knowledge: A conceptual model of knowledge-assisted visual analytics. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 92–103, 2017. doi: 10.1109/VAST.2017.8585498 2

[24] Y. Fu, X. Zhu, and B. Li. A survey on instance selection for active learning. *Knowledge and information systems*, 35(2):249–283, 2013. doi: 10.1007/s10115-012-0507-8 2

[25] J. Hong, R. Maciejewski, A. Trubuil, and T. Isenberg. Visualizing and comparing machine learning predictions to improve human-ai teaming on the example of cell lineage. *IEEE Transactions on Visualization and Computer Graphics*, 30(4):1956–1969, 2024. doi: 10.1109/TVCG.2023.3302308 1

[26] J. Huang, A. Mishra, B. C. Kwon, and C. Bryan. Conceptexplainer: Interactive explanation for deep neural networks from a concept perspective. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):831–841, 2023. doi: 10.1109/TVCG.2022.3209384 3

[27] D. W. Hubbard. *How to measure anything: Finding the value of intangibles in business*. John Wiley & Sons, 2014. doi: 10.1002/9781118983836 5

[28] S. T. Iqbal and E. Horvitz. Disruption and recovery of computing tasks: field study, analysis, and directions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 677–686, 2007. doi: 10.1145/1240624.1240730 4

[29] ITRex Group. Machine learning costs: Price factors and real-world estimates. https://itrexgroup.com/blog/machine-learning-costs-price-factors-and-estimates/, 2025. Accessed: March 28, 2026. 7

[30] D. Jannach and M. Jugovac. Measuring the business value of recommender systems. *ACM Trans. Manage. Inf. Syst.*, 10(4), art. no. 16, 23 pp., 2019. doi: 10.1145/3370082 5

[31] L. Jiang, S. Liu, and C. Chen. Recent research advances on interactive machine learning. *Journal of Visualization*, 22(2):401–417, 2019. doi: 10.1007/s12650-018-0531-1 1

[32] J. Krause, A. Perer, and E. Bertini. Infuse: Interactive feature selection for predictive modeling of high dimensional data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1614–1623, 2014. doi: 10.1109/TVCG.2014.2346482 3

[33] J. Kress, M. Larsen, J. Choi, M. Kim, M. Wolf, N. Podhorszki et al. Opportunities for cost savings with in-transit visualization. In *High Performance Computing: 35th International Conference, ISC High Performance 2020, Frankfurt/Main, Germany, June 22–25, 2020, Proceedings*, pp. 146–165, 2020. doi: 10.1007/978-3-030-50743-5_8 5

[34] K. Labonté and F. Vachon. Resuming a dynamic task following increasingly long interruptions: The role of working memory and reconstruction. *Frontiers in Psychology*, 12:659451, 2021. doi: 10.3389/fpsyg.2021.659451 6

[35] H. Lam. A framework of interaction costs in information visualization. *IEEE transactions on visualization and computer graphics*, 14(6):1149–1156, 2008. doi: 10.1109/tvcg.2008.109 4

[36] Lambda Labs. AI cloud pricing: GPU compute and infrastructure. https://lambda.ai/pricing, 2026. Accessed: March 28, 2026. 6

[37] Levels.fyi. Machine learning and AI engineer salaries. https://www.levels.fyi/t/software-engineer/focus/ml-ai?countryId=254, 2026. Accessed: March 28, 2026. 6

[38] Z. Liu and J. Heer. The effects of interactive latency on exploratory visual analysis. *IEEE transactions on visualization and computer graphics*, 20(12):2122–2131, 2014. doi: 10.1109/tvcg.2014.2346452 4

[39] Y. Ma, T. Xie, J. Li, and R. Maciejewski. Explaining vulnerabilities to adversarial machine learning through visual analytics. *IEEE transactions on visualization and computer graphics*, 26(1):1075–1085, 2020. doi: 10.1109/TVCG.2019.2934631 1, 2, 3, 8

[40] N. H. Mackworth. The breakdown of vigilance during prolonged visual search. *Quarterly Journal of Experimental Psychology*, 1(1):6–21, 1948. doi: 10.1080/17470214808416738 7

[41] A. Mahadevan and M. Mathioudakis. Cost-aware retraining for machine learning. *Knowledge-Based Systems*, 293:111610, 2024. doi: 10.1016/j.knosys.2024.111610 4

[42] M. Millecamp, N. N. Htun, C. Conati, and K. Verbert. To explain or not to explain: the effects of personal characteristics when explaining music recommendations. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, pp. 397–407, 2019. doi: 10.1145/3301275.3302313 9

[43] E. Mosqueira-Rey, E. Hernández-Pereira, D. Alonso-Ríos, J. Bobes-Bascarán, and Á. Fernández-Leal. Human-in-the-loop machine learning: a state of the art. *Artificial Intelligence Review*, 56(4):3005–3054, 2023. doi: 10.1007/s10462-022-10246-w 2

[44] T. Munzner. A nested model for visualization design and validation. *IEEE transactions on visualization and computer graphics*, 15(6):921–928, 2009. doi: 10.1109/tvcg.2009.111 4, 5

[45] J. Nielsen. *Usability engineering*. Morgan Kaufmann, 1994. 4, 6

[46] M. Nourani, J. King, and E. Ragan. The role of domain expertise in user trust and the impact of first impressions with intelligent systems. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, vol. 8, pp. 112–121, 2020. doi: 10.1609/hcomp.v8i1.7469 9

[47] H. Park, Y. Nam, J.-H. Kim, and J. Choo. Hypertendril: Visual analytics for user-driven hyperparameter optimization of deep neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1407–1416, 2021. doi: 10.1109/TVCG.2020.3030380 3

[48] N. Pattyn, X. Neyt, D. Henderickx, and E. Soetens. Psychophysiological investigation of vigilance decrement: boredom or cognitive fatigue? *Physiology & Behavior*, 93(1-2):369–378, 2008. doi: 10.1016/j.physbeh.2007.09.016 7

[49] P. Pirolli and S. Card. Information foraging. *Psychological review*, 106(4):643, 1999. doi: 10.1037/0033-295X.106.4.643 7

[50] P. Pirolli and W.-T. Fu. Snif-act: a model of information foraging on the world wide web. pp. 45–54, 2003. 8

[51] D. Sacha, M. Kraus, D. A. Keim, and M. Chen. Vis4ml: An ontology for visual analytics assisted machine learning. *IEEE transactions on visualization and computer graphics*, 25(1):385–395, 2019. doi: 10.1109/tvcg.2018.2864838 1, 2

[52] D. Sacha, M. Sedlmair, L. Zhang, J. A. Lee, J. Peltonen, D. Weiskopf et al. What you see is what you can change: Human-centered machine learning by interactive visualization. *Neurocomputing*, 268:164–175, 2017. doi: 10.1016/j.neucom.2017.01.105 2

[53] D. Sacha, A. Stoffel, F. Stoffel, B. C. Kwon, G. Ellis, and D. A. Keim. Knowledge generation model for visual analytics. *IEEE transactions on visualization and computer graphics*, 20(12):1604–1613, 2014. doi: 10.1109/TVCG.2014.2346481 2, 5

[54] N. Sambasivan, S. Kapania, H. Highfill, D. Akrong, P. Paritosh, and L. M. Aroyo. "everyone wants to do the model work, not the data work": Data cascades in high-stakes ai. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, art. no. 39, 15 pp. Association for Computing Machinery, NY, USA, 2021. doi: 10.1145/3411764.3445518 5

[55] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-lite: A grammar of interactive graphics. *IEEE transactions on visualization and computer graphics*, 23(1):341–350, 2017. doi: 10.1109/TVCG.2016.2599030 5

[56] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner et al. Hidden technical debt in machine learning systems. In *Advances in Neural Information Processing Systems*, vol. 28. Curran Associates, Inc., 2015. 2

[57] M. Sedlmair, M. Meyer, and T. Munzner. Design study methodology: Reflections from the trenches and the stacks. *IEEE transactions on visualization and computer graphics*, 18(12):2431–2440, 2012. doi: 10.1109/tvcg.2012.213 2

[58] B. Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009. 2, 5

[59] S. Shankar, R. Garcia, J. M. Hellerstein, and A. G. Parameswaran. "we have no idea how models will behave in production until production": How engineers operationalize machine learning. *Proc. ACM Hum.-Comput. Interact.*, 8(CSCW1), art. no. 206, 34 pp., 2024. doi: 10.1145/3653697 1, 2, 5

[60] F. Sperrle, M. El-Assady, G. Guo, R. Borgo, D. H. Chau, A. Endert et al. A survey of human-centered evaluations in human-centered machine learning. *Computer Graphics Forum*, 40(3):543–568, 2021. doi: 10.1111/cgf.14329 2

[61] T. Spinner, U. Schlegel, H. Schäfer, and M. El-Assady. explAIner: A visual analytics framework for interactive and explainable machine learning. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):1064–1074, 2020. doi: 10.1109/TVCG.2019.2934629 2, 3

[62] H. Strobelt, S. Gehrmann, M. Behrisch, A. Perer, H. Pfister, and A. M. Rush. Seq2Seq-Vis: A visual debugging tool for sequence-to-sequence models. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):353–363, 2019. doi: 10.1109/TVCG.2018.2865044 1, 2

[63] J. van Wijk. The value of visualization. In *VIS 05. IEEE Visualization, 2005.*, pp. 79–86, 2005. doi: 10.1109/VISUAL.2005.1532781 1, 2, 3, 4, 5

[64] A. J. Vickers and E. B. Elkin. Decision curve analysis: a novel method for evaluating prediction models. *Medical Decision Making*, 26(6):565–574, 2006. doi: 10.1177/0272989x06295361 5

[65] D. Von Winterfeldt, R. S. Farrow, R. S. John, J. Eyer, A. Z. Rose, and H. Rosoff. Assessing the benefits and costs of homeland security research: A risk-informed methodology with applications for the us coast guard. *Risk analysis*, 40(3):450–475, 2020. doi: 10.1111/risa.13403 5

[66] E. Wall, S. Das, R. Chawla, B. Kalidindi, E. T. Brown, and A. Endert. Podium: Ranking data using mixed-initiative visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):288–297, 2018. doi: 10.1109/TVCG.2017.2745078 3

[67] Y. Wang, J. Zhao, J. Hong, R. G. Askin, and R. Maciejewski. A simulation-based approach for quantifying the impact of interactive label correction for machine learning. *IEEE Transactions on Visualization and Computer Graphics*, 31(9):5687–5703, 2025. doi: 10.1109/TVCG.2024.3468352 1

[68] J. S. Warm, R. Parasuraman, and G. Matthews. Vigilance requires hard mental work and is stressful. *Human factors*, 50(3):433–441, 2008. doi: 10.1518/001872008X312152 7

[69] G. Warnell, N. Waytowich, V. Lawhern, and P. Stone. Deep tamer: interactive agent shaping in high-dimensional state spaces. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, art. no. 189, 9 pp. AAAI Press, 2018. 2

[70] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. Viégas, and J. Wilson. The what-if tool: Interactive probing of machine learning models. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):56–65, 2020. doi: 10.1109/TVCG.2019.2934619 3

[71] G. Wylie and A. Allport. Task switching and the measurement of "switch costs". *Psychological research*, 63(3):212–233, 2000. doi: 10.1007/s004269900003 4

[72] S. Xiang, X. Ye, J. Xia, J. Wu, Y. Chen, and S. Liu. Interactive correction of mislabeled training data. In *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 57–68, 2019. doi: 10.1109/VAST47406.2019.8986943 2, 3

[73] T. Xie, Y. Ma, J. Kang, H. Tong, and R. Maciejewski. Fairrankvis: A visual analytics framework for exploring algorithmic fairness in graph mining models. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):368–377, 2022. doi: 10.1109/TVCG.2021.3114850 3

[74] X. Xuan, X. Wang, W. He, J. P. Ono, L. Gou, K.-L. Ma et al. Vista: A visual analytics framework to enhance foundation model-generated data labels. *IEEE Transactions on Visualization and Computer Graphics*, 31(10):6991–7003, 2025. doi: 10.1109/TVCG.2025.3535896 1

[75] J. Yuan, C. Chen, W. Yang, M. Liu, J. Xia, and S. Liu. A survey of visual analytics techniques for machine learning. *Computational Visual Media*, 7(1):3–36, 2021. doi: 10.1007/s41095-020-0191-7 1, 2, 13

[76] Y. Zhang, B. Coecke, and M. Chen. On the cost of interactions in interactive visual machine learning. In *2019 IEEE workshop on Evaluation of Interactive Visual Machine Learning systems (Eviva-ML). IEEE*, pp. 1–5, 2019. 4

[77] Y. Zhang, B. Coecke, and M. Chen. Mi3: Machine-initiated intelligent interaction for interactive classification and data reconstruction. *ACM Trans. Interact. Intell. Syst.*, 11(3–4), art. no. 18, 34 pp., 2021. doi: 10.1145/3412848 2, 5

[78] J. Zhao, Y. Wang, M. V. Mancenido, E. K. Chiou, and R. Maciejewski. Evaluating the impact of uncertainty visualization on model reliance. *IEEE Transactions on Visualization and Computer Graphics*, 30(7):4093–4107, 2024. doi: 10.1109/TVCG.2023.3251950 2