Artificial Intelligence and Visualization: Toward Visual Knowledge Discovery

Boris Kovalerchuk
Dept. of Computer Science
Central Washington University, USA
https://orcid.org/0000-0002-0995-9539

Abstract— The integration of artificial intelligence/machine learning (AI/ML) with visualization techniques is emerging as a transformative paradigm for analyzing high-dimensional data for getting trust in the AI/ML predictive models. By bridging computational analytics with human-centric visual reasoning, a new Visual Knowledge Discovery (VKD) field focuses on addressing critical challenges in trustworthiness, interpretability, and cognitive augmentation for AI/ML. While classical orthogonal Cartesian coordinates have been a backbone of science for 400 years, they visualize only 2D or 3D data. Parallel Coordinates partially removed this limitation. The key new opportunity is in a lossless visualization of high-dimensional data with capable graph-based General Line Coordinates (GLC). This paper presents: (1) AI/ML model trust traits with the role of interpretability among them, (2) model interpretability and explainability approaches in the context of linear explanations and their controversy, (3) linear model explainability measure with abilities to generalize it for non-linear models, (4) visual knowledge discovery paradigm to foster trustworthy models, and (5) an outline of future work.

Keywords—visual knowledge discovery, classification, clustering, hyper-rectangles, general line coordinates, glyphs, shifted paired coordinates.

I. Introduction

AI/ML and Visualization domains benefit from their **integration** supporting users in multiple aspects. For the AI/ML this transformative paradigm gives users the ability to analyze high-dimensional data and models visually to get **user trust** in the AI/ML predictive models and to deploy models. It includes dealing with critical challenges in model trustworthiness, interpretability, and cognitive augmentation for AI/ML with visual means. For visualization it is a notable enhancement for users to (1) **automate visualization** generation with prompts in a natural language and (2) to deeper **visual reasoning** and **analytics**. An umbrella term Visual Knowledge Discovery (VKD) [1] is used now for this integrated interdisciplinary field of AI/M and visualization.

On the technical side the VKD challenges are in visualization of high-dimensional data in 2-D or 3-D **without loss** of n-D information. Classical orthogonal Cartesian coordinates have been a backbone of science for 400 years, but they visualize only 2D or 3D data. Parallel Coordinates [2] have been the first step to lifting this limitation for some types of data. A wider set of methods is now emerging in VKD. The new opportunity here is in a lossless visualization of high-dimensional data with emerging General Line Coordinates (GLC) [1].

The ML process fundamentally involves a **user**, who needs to **trust** and accept a model and its explanation to employ it confidently. A long time ago Donald Michie [3,4] formulated three very relevant requirements:

- ML improves predictive *accuracy* with more data.
- ML provides a *hypothesis (model)* in symbolic *comprehensible* form.
- ML *teaches the comprehensible hypothesis to a human* who then performs *better* than the human studying the training data alone (Comprehensibility).

Integration of AI and visualization is an important avenue to meet these requirements, making human operation with these concepts implementable, which is the focus of this work.

This paper is organized as follows. Section II presents AI/ML model trust traits with the role of interpretability among them. Section III illustrates model explainability and interpretability approaches in the context of linear explanations and their controversy. This includes the proposed linear model explainability measure. Section IV presents Visual Knowledge Discovery paradigm to foster trustworthy models. Section V concludes the paper with its summary and an outline of future work.

II. AI/ML model trust traits and explainability issues

A. Model trust traits

The NIST AI Trustworthiness Framework codifies key trust traits such as validity, safety, reliability, security, interpretability, explainability, transparency, privacy, and fairness as essential guidelines for trustworthy AI systems [5] Thus, trustworthy ML requires a multidimensional evaluation framework integrating robustness, interpretability, fairness, safety, user interaction, and transparency [6].

Fairness addresses bias and the prevention of discrimination against subgroups [7] *Robustness* measures a model's stability against noise and adversarial inputs [8] *Confidence and calibration* ensure that the predicted probabilities accurately reflect the true likelihoods [9] *Interpretability* involves understanding model decisions via feature importance rankings and explanations aligned with domain knowledge [10]. *Safety and security* ensure models behave reliably under adversarial or edge cases [11]. *Transparency and accountability* measure a system's understandability and auditability [12].

The FRIES Trust Score combines fairness, robustness, integrity, safety, and explainability into a consolidated metric

[13]. User *behavior* and *interaction metrics* in real deployments offer additional trust signals complementing technical measures [14].

B. Challenges of Explanation of Black-box Models

From the user perspective explanation of current black-box machine learning models is one of the major issues that limit deployment of the ML predictive models as trusted systems. The attempts to address it range from explaining black-box models to building interpretable models in the first place.

Initially Explainable AI (XAI) was focused methods to explain black-box AI/ML predictive models with methods like LIME [15] and SHAP [16] with simple bar chart visualizations.

Later important deficiencies of these *first-generation explanation methods* were revealed with introduction of the **quasi-explanation** term for deficient explantion methods and models [17]. This motivated a demand for the *second generation of explanation method*s to **explain explanations** including explaining explanations **visually** as appealing for the users. Thus, new visual methods emerge to help domain experts to become key players in model discovery, explanation, and "explanation" of explanations without being ML experts.

C. Fundamental Explanability approaches

There are two competing fundamentally different explainability/interpretability approaches in Machine Learning.
Approach 1: **Build** an accurate **intrinsically explainable** model [18] in the first place, such as *self-explained* decision tree and logical rules.
Approach 2: **Explain** accurate shallow and deep **black-box** models [15,16], such as Support Vector Machine (SVM), Random Forest (RF), some linear models, and Convolutional Neural Networks (CNN).

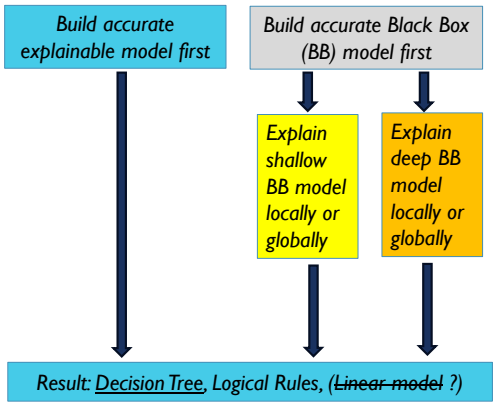


Figure 1. Explanation approached

These approaches are illustrated and contracted in Figure 1. In Figure 1 some **linear predictive models** are listed in the category of the black-box model, while often linear models are considered as *very interpretable* models due to their *simplicity* considering their coefficients as *feature importance* [e.g., 19]. The statements about interpretability of linear models are well presented in data science publications, tutorials, and surveys like "Linearity leads to interpretable models", "…linear equations have an easy-to-understand interpretation on a modular level (i.e., the weights)" [19], "One way to achieve interpretability is to use interpretable models, such as linear regression, logistic regression and decision trees" [25], and "…we design partial linear models that are inherently interpretable." [22].

Below we show the weakness of this popular view. That popular view has a limited conditional basis for data with homogeneous features, which have the same physical meaning as temperature at different time moments. However, in machine learning, typical data are **heterogeneous** with one feature measuring temperature and others measuring blood pressure, pulse, and so on. This explains crossed "linear model" with a question mark in Figure 1 to attract attention to this issue that we discuss below.

There are two categories of linear models: **linear predictive models** and **linear explanation models**. *Linear predictive models* are often considered as interpretable by default, which do not require any further explanation. Respectively, the assumption that linear models are perfectly explainable is a basis of *linear explanation models* in many popular methods like LIME [15] and their different modifications. This is an **extreme simplification** of the actual situation with linear predictive models with important negative consequences as we discuss below. Existing model-agnostic explanation methods for black boxes like LIME [15] or SHAP [16] can distort the original relationships, lead to incorrect and even harmful decisions in a high-stakes decision-making tasks [18,20,21,22].

II. Controversy of Base Linear Explanations

A. Is common acceptance of linear explanations justified?

A popular way to explain complex black-box models is to approximate these models by simple linear models such as in LIME (Local Interpretable Model-Agnostic Explanations) [15]. The assumption is that a linear model is easily explainable to any user without ML knowledge [19]. We just need to tell a user that some attribute is the most important because its coefficient is the largest one in the linear model. Then another attribute is considered as next in importance because it has the next largest coefficient and so on explaining the role of all attributes. A bar chart visualizes it in Figure 2 to get this information easier [23].

Typically, we have an insufficient number of actual cases that are close to the given case. This case is permuted in LIME (altering it presumably slightly) and outputs of the underlying black-box ML model for these cases are computed. Next, (i) these synthetic data are used to produce a *local sparce linear model*, (ii) *alterations* are discovered that *change the decision* of the local model, and (iii) these alterations are considered as *most important* for the underlying black-box ML model for the given case. Defining the *limits* of local area for LIME is difficult in multidimensional space without abilities of the user to observe multidimensional data and to set up such area.

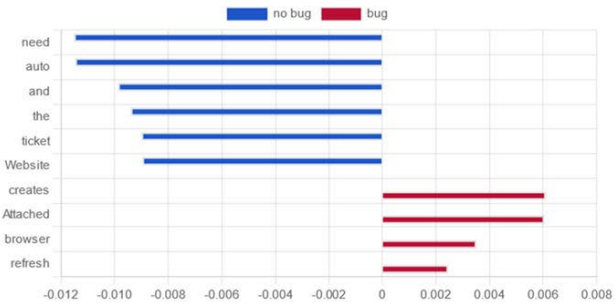


Figure 2. Bar-chart for LIME model [23].

One of the claimed properties of linear models like LIME is that they are *more interpretable* than non-linear models [24]. This statement mixes interpretability and transparency as we discuss in the next section. Higher transparency does not automatically imply higher interpretability. Next, LIME does not take into account *causality* [24] and is not well justified for ML tasks with *heterogeneous* data [17].

In summary typical **arguments** that linear regression models should be viewed as interpretable are [26]: (1) linear models have **simple model structure**, (2) linear *structure is easy* to understand, (3) the *output* of the linear model is *easy* to understand, (4) the *weight* of each feature represents the mean change in the prediction given a one unit increase of the feature, and (5) the features with larger weights have more *effect* on the result. This simplified and rather naïve approach can be misleading [26] and even dangerous for high-stake tasks. LIME may suffer from inaccuracies or lack of reproducibility [27].

B. Transparency vs. Interpretability

Transparency means that the **model's mechanics** and data are openly available and understandable. **Interpretability** means that the model's decisions can be **explained in understandable domain terms** that is appropriate for domain experts and/or end users, often by tracing input features to outputs. While transparency is a prerequisite for interpretability, it is not sufficient by itself. A model can be transparent (e.g., open-source code, clear documentation) but still be difficult to interpret if its internal logic is complex or opaque. So, claiming linear models as interpretable equites interpretability with a much weaker concept of transparency and misleads the users.

Linear models are structurally transparent (coefficients are clearly present in the linear model $y=\beta_0+\beta_1x_1+\beta_2x_2+\ldots+\beta_nx_n$) this transparency does not guarantee actionable interpretability, especially when features are scaled arbitrarily or suffer from multicollinearity [28]. Alas, often the methods that provide transparency are viewed as explanation methods.

While warnings have been published, e.g., [17,24,28,29, 49,50] the unconditional claims about interpretability of linear model continue. Linear models remain dominantly "interpretable" due to their computational efficiency, easiness to implement, and wide support in software. Their mathematical simplicity and long-standing use reinforce the perception of inherent interpretability. These claims hamper research to resolve the issue for users. Moreover, Large Language Models harvest these incorrect claims and mislead a wide audience based on them amplifying the negative impact of this claim.

Interpretability is frequently treated as a binary property rather than a **gradable**, context-dependent quality, which allows linear models to retain their status as "interpretable" by default. There are no standardized benchmarks or validation frameworks for interpretability, so linear models are often accepted without critical evaluation.

III. EXPLIANABILITY MESURES FOR ML MODEL

A. Explainability measure for linear midels

Despite concern about linear models as quasi-explainable [17], they dominate as "interpretable" that requires resolution with operational, context-based interpretability criteria and, validation frameworks [24]. This will help to deploy truly interpretable linear models instead of quasi-explainable ones. To move from blanket acceptance of linear models as interpretable to a **nuanced, operational framework**, below we propose a set of criteria to measure the progress of the models to become interpretable.

We define a **Linear Model Explainability Measure** (**LMEM**) based on 10 criteria below. Each criterion is assigned up to 10 points. LMEM can get a maximum of 100 points and can be applied for a local or a global linear model.

1.Operational criteria

1.1 Explanation is *invariant to arbitrary scaling* (e.g., inches vs. centimeters). *Features* are standardized. *Units* are domain meaningful.

1.2 *Coefficients* are interpretable in *domain-relevant units*.

1.3 *Multicollinearity* is resolved (Correlation between features is low. Linear models with high feature *correlation* are more likely *non-interpretable*).

1.4 Explanations are with *domain-specific causal theories* or *mechanisms* (e.g., medical models should reflect biologically plausible effects).

1.5 Areas with *compensation property* of the linear model are discovered and explanation is *localized* to these areas.

1.6 Explanations are held under *realistic data shifts* (counterfactual sanity test).

2.Validation criterion

A linear model recovers "*ground-truth*" *explanations* known for *benchmark* datasets.

3.Disclosure criteria

3.1 Possible *negative "side effects*" of violation of listed above criteria on interpretability of the linear model are disclosed to the users.

3.2. The *methods* used to confirm all criteria listed above are disclosed to the users. This includes feature normalization, scaling, multicollinearity diagnostics, domain validation of coefficient signs, magnitudes and others.

3.3. The requirement of certification of the linear model's interpretability is disclosed to the users.

4. Certification Criterion

Certification authority is *established and disclosed*, including specific users who can be assigned as a certification authority.

Below we specify criterion 1.1 that explanation is *invariant to arbitrary scaling, features* are standardized and *units* are domain meaningful.

- The domain interpretable features are present (originally given or extracted from raw data). 4 points.
- Explanation is *invariant to arbitrary scaling* (e.g., inches vs. centimeters). *Features* are standardized. *Units* are domain meaningful. 6 points.

Next we specify criterion 1.2 that coefficients are interpretable in domain-relevant units.

- The order of importance of interpretable attributes as the order of coefficients is invariant to unit change (e.g., from inches to centimeters). 3 points
- Coefficients are presented in domain-relevant units. 3 points.
- The invariant order of coefficients is confirmed by a domain expert. 4 points.

Now we specify criterion 1.4 that explanations are with domain-specific causal theories or mechanisms, 10 points):

- The invariant order of coefficients (importance of attributes) is explained in the domain terms/knowledge/ theory. 3 points.
- An interpretable model on interpretable features is developed like a decision tree, or a set of logical rules that accurately approximate a black box model, 3 points.
- A set of similar cases for a given case to be predicted is found in the same class as predicted for a given case, 2 points
- A set of rather similar cases for a given case to be predicted is found in alternative classes to the class predicted for the given case (counterfactual cases), 2 points.

It is desirable to reach a score of 80% or greater. We would call it an **Extensive Explanation** (**EE**). In line with this we will call the range 60-79% as an **Acceptable Explanation** (**AE**) and explanation below 60% as an **Unacceptable Explanation** (**UE**). Currently only a few of these criteria are confirmed for real-world machine learning tasks in the publications that use methods like LIME. So far, our informal analysis of the explanations with linear models in the literature shows that typically linear models are not tested for these 10 criteria or do not satisfy them. This is a topic of future activities.

B. Expansion of explainability measure to non-linear models

A **Model Explainability** Measure (**MEM)** that is applicable to **non-linear models** can be produced by generalizing LMEM. Below we clarify the validation criterion for both linear and non-linear models. A set of cases is selected and explained by a domain expert answering **why** the target attribute values of these cases should be as stated in these cases.

The model should recover these "ground-truth" expert explanations, which is non-trivial. We cannot expect that the explanation that is natural for a domain expert is readily present in the linear and non-linear model. An expert can order attributes by their importance for a given case or a set of similar cases. If this order is consistent with the order of magnitude of coefficients of the linear function, then the linear explanation is validated in this part. For non-linear functions the confirmation of the experts' order of attributes is more challenging and can fail as the analysis in [28] of the SHAP example from [16] shows. Next when expert's explanation contains other components then they may or may not be "recovered" by a linear model some other non-linear models too.

For explanation of a given case **x** provided by the expert in the forms of k nearest cases neighbors $\{\mathbf{y}\}$ from the same class we need to find k nearest cases $\{\mathbf{z}\}$ computed independently by k-NN algorithm and compare $\{\mathbf{y}\}$ and $\{\mathbf{z}\}$. A linear model F can indirectly confirm the k expert's nearest neighbors if values of $F(\mathbf{y})$ and $F(\mathbf{z})$ are close, but it does not guaranty that $\{\mathbf{y}\}$ and $\{\mathbf{z}\}$ will be the same or very similar. When the explanation model is a decision tree then **y** and **z** should be in the same terminal node of the decision tree.

In the proposed approach the domain experts independently offer k nearest cases. It differs from the current practice where the expert only confirms or rejects the k nearest cases computed by the k-NN without generating own k nearest cases.

In general, the current practice is offering an explanation and asking a domain expert to confirm or reject it. This can be biased because a domain expert can be inclined to accept the model's explanation when the accuracy of the model is high. In contrast, when the validation cases are selected independently this bias is avoided. This is in line with the validation practice for evaluating accuracy, where the validation/test cases are independent from the training data.

Generalized Additive Models (**GAM**) generalize linear models by assuming the model $F(\mathbf{x})$ as a weighted sum of functions: $a_1F_1(x)+ a_2F_2(x)+\ldots+ a_nF_n(x)$ to deal with data, which require models that are more complex than linear. In [15] simpler GAM models are considered as more interpretable. Obviously, the simplest form of GAM are linear functions, where each $F_i(\mathbf{x})=x_i$. As we discuss this can be ill justified without producing and analysis the proposed Linear Model Explainability Measure (LMEM).

Thus, insufficiency of simplicity as a measure of interpretability is applicable to GAM too. **Piecewise linear functions** [29] have the same interpretability issues as GAM. **Quantile regression** builds sets of linear models to different percentiles (subsets) of the training data. While these models are claimed to be interpretable, in fact, each of them has the same interpretability issues as linear models discussed above. They cannot be claimed *unconditionally interpretable*.

IV. Visualization and Visual Knowledge Discovery with their Role to Foster Trustworthy Models

A. Background and key challenges

ML methods use visualization for feature importance by bar charts (Figure 2) in LIME and SHAP, and by heat maps to show

salient features. In general visualization methods visualize data, prediction and prediction explanation models. The role of **Visual Knowledge Discovery** (**VKD**) is wider, which is fostering discovering data patterns, features, prediction and prediction explanation models. Below we review the current VKD ecosystem with associated visualization methods.

The key challenge of visualization of high-dimensional data is that we cannot see naturally them in our physical 3-D world by a naked eye in contrast with 3-D physical objects. This led to proliferation of **Dimension Reduction** (**DR**) methods [30,31] like PCA, t-SNE, MDS that convert high-dimensional n-D points to 2D/3D points to visualize them, say 20-D data to 2-D data. This **point-to-point conversion is fundamentally lossy**; it does not preserve all n-D information when, say 20 numbers are converted to 2 numbers. It only *minimizes some specific losses/distortions*, like minimizing the difference of average n-D distance between points in n-D space and average distance between these points in 2-D visualization space. Literature reported significant distortion and studies to discover such distortions [32]. Also, it creates new *artificial* 2-D visualization coordinates that are difficult to interpret/explain in the domain.

Another limitation of DR methods is that they use **classical orthogonal Cartesian coordinates** for visualization. While these coordinates have been a backbone of science for 400 years, they naturally visualize only 2D or 3D data. **Parallel Coordinates** [2] (Figure 3a) have been a first step to lifting this limitation by removing requirements of (1) orthogonality of coordinates, (2) common origin of all coordinates, and (3) a single 2-D/3-D point that visualizes n-D point.

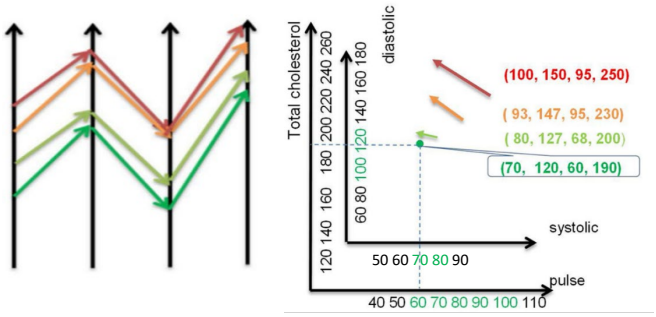
The challenges with high-dimensional data in AI/ML include inability to observe high-dimensional data together and loss of high-dimensional information by using Dimension Reduction (DR) methods. It leads to less accurate and reliable decision models derived from DR. Ability to mitigate loss of high-dimensional information caused by DR are limited as formally established by the Johnson-Lindenstrauss Theorem [1].

B. General Line Coordinates

General Line Coordinates (GLC) [1] are a family of coordinate systems designed to visualize and analyze high-dimensional data in two or three dimensions without information loss. By extending known coordinate systems such as Cartesian, Parallel, and Radial coordinates, GLC enables bijective mappings between *n*-dimensional (*n*-D) data points and their 2D/3D representations. This approach addresses critical challenges in ML, and information visualization in preserving data fidelity while enabling human interpretation.

GLC paradigm leverages **graph-based** visualization strategy in contrast to traditional **point-based** dimension reduction. This enables: (1) preservation of all high-dimensional information without corruption, (2) visualization and analysis of high-dimensional data within these graph-based spaces, and (3) retention of high-dimensional details in explainable forms suitable for domain experts. In contrast, point-based DR methods bring information loss and limited interpretability.

The graph-based GLC methods convert n-D points to **graphs** with several points per graph preserving all n-D information. GLC include Parallel Coordinates (PC), Radial Coordinates (RC), Collocated Paired Coordinates (CPC), Shifted Paired Coordinates (SPC), Paired Radial Coordinates (PRC), In-line coordinates (ILC), Circular, Elliptic, Concentric, n-Gon, etc.



(a) Parallel Coordinates. (b) Shifted Paired Coordinates.

Figure 3. 4-D health monitoring visualization on Parallel Coordinates (a) and Shifted Paired Coordinates (b) with attributes: systolic blood pressure, diastolic blood pressure, pulse, and total cholesterol at four moments. In (b) The green dot is the desired goal state, the red arrow is the initial state, the orange arrow is the health state at the next monitoring time, and the light green arrow is the current health state of the person [1].

A recent review of time series visualization [33] stated "a lack of proposals for entirely new graphical metaphors" referencing only [34] (2008) as really innovating that extend parallel coordinates. Other current works only join exiting metaphors and interactive tools [33]. This shows importance of the GLC-based approach as bringing new graphical metaphors.

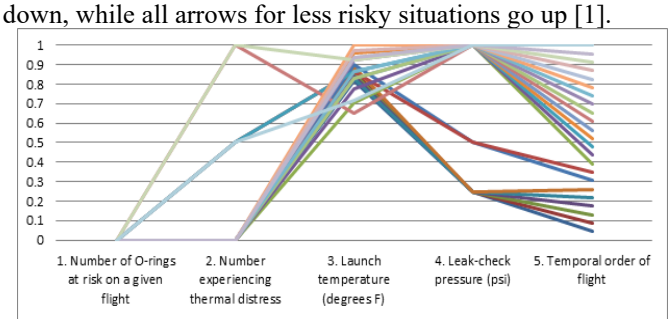
The **main idea of General Line Coordinates** is that they do not require a common center to be orthogonal in contrast with classical orthogonal Cartesian coordinates. GLC can be located *everywhere in the visualization space, be collocated, shifted, at different angles and curvilinear with locations dependent on other coordinates*. Points on coordinates represent values of attributes of cases, e.g., patients. These points are connected by directed edges forming a graph for each case. See Figure 2 for 4-dimensional health data of the patient at 4 moments. In Figure 2a coordinates do not have a common origin but are parallel forming **Parallel Coordinates**.

In Figure 2b two Cartesian coordinates are shifted forming **Shifted Paired Coordinates (SPC)**. This shift is specially created to make the target 4-D health state a single green dot, as its projections to SPC show.

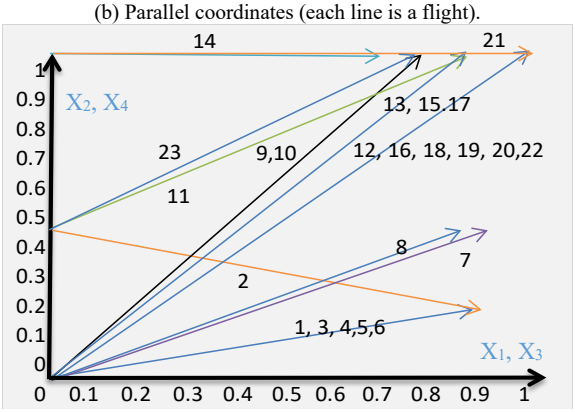
B.Examples of visualiation and learning tasks with GLC

Below we demonstrate other GLC with their use. **Collocated Paired Coordinates (CPC)** [1] re shown are in Figure 3b t in comparison with parallel coordinates in Figure 3a. In CPC two pairs of Cartesian coordinates are collocated. The first pair (X_1,X_2) and the second pair (X_3, X_4) are in the same places. These 4 attributes are attributes of 23 Challenger flights that preceded the Challenger disaster. CPC visualization has a significant advantage over visualization in Figure 3a. Figure 3a does not draw attention to risky situations like the Challenger disaster flight. In contrast Figure 3b in CPC shows three risky

situations, as two horizontal arrow and one arrow that goes down, while all arrows for less risky situations go up [1].



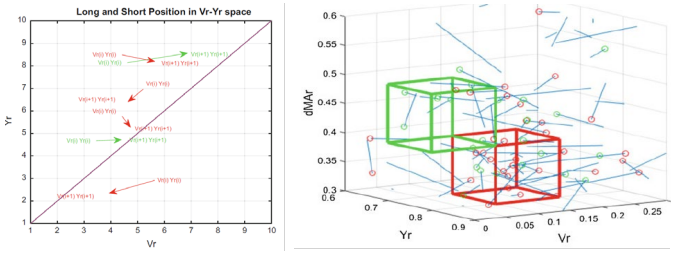
(b) Parallel coordinates (each line is a flight).



(b) Collocated Paired Coordinates.

Figure 4. Challenger USA Space Shuttle normalized O-Ring dataset of 23 flights before Challenger disaster in parallel coordinates and collocated paired coordinates [1].

Figure 5 [1] visualizes dynamics of financial trading data in 2D and 3D in collocated coordinates for trading decisions. Each arrow in (a) shows change of values of two attributes from time *t* to time *t*+1 capturing 4-D data. Each pin in (b) shows time change of 3 attributes as 6-D data.



(a) 4-D data in CPC in 2-D. (b) 6-D data in CPC in 3-D.

Figure 5. Trading data in time sequence (*t*, *t*+1) in CPC. Green and red boxed show areas of positive and negative trading conditions.

C.Explaining Explanations Visually

Often explanation methods for ML models need explantion of their explantion for better understanding by domain experts and end users, to make explanations more complete, and to convert quasi-explanations into true ones [17]. Traditional visualizations only illustrate explanation by showing importance of attributes or pixels with bar charts and heatmaps. They do not explain why some attribute is of higher importance than others. Here VKD and GLC methods are notably useful, as they enable us to trace, visualize, and make sense of the explanation. Figure 6 shows an example with GLC-Linear [1,39] where the attributes of higher importance have smaller angles and larger projection to the horizontal line showing their larger input to the value of the linear classifier.

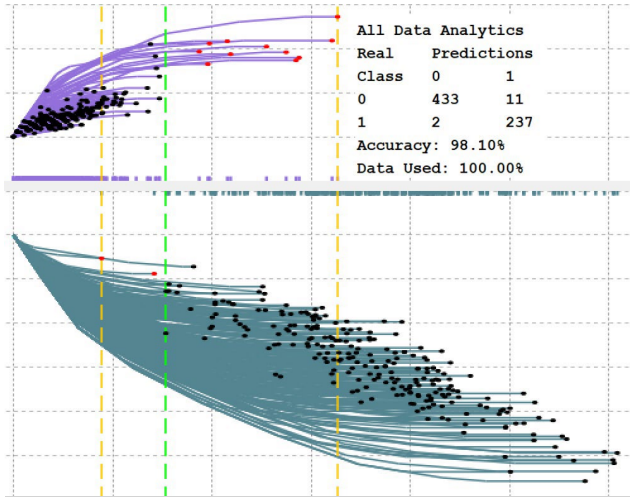


Figure 6. Wisconsin Breast Cancer data in GLC-L with 98.1% accuracy Attribute importance is shown by smaller angles to horizontal line. [39]

The hyperblock visualization in parallel coordinates is another example. Figure 7 shows why new black cases are classified into class C. It is visible that those cases are fully in the hyperblocks that contain only training cases of class C.

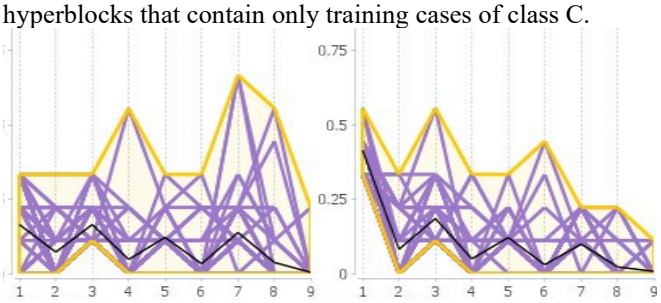


Figure 7. Visual explanation of classification of black cases with visualization in Parallel Coordinates for a hyperblock classifier [39]

D. Capabilities of VKD ecosystem based on GLC

The VKD ecosystem with GLC consists of traditional ML methods and VKD methods that can work in concerts. Table 1 lists major VKD/GLC concepts with references and abbreviations. Table 2 enumerates major classification and clustering ML methods with applicable VKD/GLC methods. Table 3 presents meta level ML techniques to enhance ML with VKD/GLC methods. Table 2 and 3 use abbreviations listed in Table 1. Examples below illustrate combining ML and VKD.

Example 1. Building and visualizing **decision trees** by using Shifted Paired Coordinates (SPC) and Bended Coordinates [35] where users can: (1) visualize in detail a decision tree built by any ML method, (2) trace given n-D cases losslessly, (3) adjust and (4) build a decision tree interactively.

Example 2. Analyzing **additive models** (linear, quadratic, generalized additive models) and support vector machine with GLC-Linear (GLC-L) [1]. The GLC-L benefits are: (1) seeing all n-D cases losslessly; (2) seeing these cases relative to the classification threshold; (3) using (1), (2) to improve the

classification threshold interactively, and (4) using (1), (2) to improve model coefficients interactively. It offers lossless data and model visualization. To the best of our knowledge this is the only available method with this capability.

Table 1. VKD and GLC concepts

VKD and CPC Concepts	Abbreviation
Visual Knowledge discovery [1]	VKD
General Line Coordinates [1]	GLC
Full 2-D Machine Learning Paradigm [44]	Full 2D ML
Computational and Interactive Visual Learning [36]	CIVL
Parallel coordinates [1, 2]	PC
Radial Coordinates [1]	RC
General Line Coordinates – Linear [1, 39]	GLC-L
Collocated Paired Coordinates [1, 55]	CPC
Shifted Paired Coordinates [1, 35, 38, 52]	SPC
Shifted Paired Coordinates for Decision tree [35]	SPC-DT
Bended Coordinates for Decision Tree [35]	BC-DT
Stick Figures [1, 37, 38]	SF
Combined SPC and Stick Figures for Glyphs [1, 37, 38]	SPC-SF
Concentric Coordinates [48]	CoC
Concentric Coordinates for *k*-Nearest Neighbors [48].	CoC-kNN
Elliptic Paired Coordinates [43]	EPC
Dynamic Circular Coordinates [46]	DCC
Static Circular Coordinates [43]	SCC
Dynamic Scaffolding Coordinates 1 [45]	DSC1
Dynamic Scaffolding Coordinates 2 [45]	DSC1
In line Coordinates [1,44]	ILC
Sequential Rule Generation Algorithms [41]	SRG
Monotone Boolean Function [41, 51,54]	MBF
MDF Multiple Disc Form visualization [51.54]	MBF MDF
Monotone Ordinal Expert Knowledge Acquisition [51]	MOEKA
GLC Divide & Classify Process [36]	D&C
Combined SPC with several CPC [55]	SCPC

Table 2. Classification and clustering ML methods with VKD/GLC methods.

ML methods	**GLC methods**
Decision Trees (DT)	SPC-DT, BC-DT [35]
Generalized Decision Tree	DT with nodes as logical rules [36].
Additive classifiers (Linear, Quadratic, Generalized Additive)	GLC-Linear (GLC-L) [1], Divide & Classify (D&C) [36]
Support Vector Machine (SVM)	GLC-Linear [1]
k-Nearest Neighbors (k-NN)	SPC-SF Glyphs [37,38], CoC [48]
Convolutional Neural Network	CPC [42], GLC-L [1]
Hyper-rectangles, hyperblocks (HBs), hyperboxes, boxes algorithms	HB algorithms [1, 39, 40,41], GLC-L [1], HyClu, HyCl [37, 38], EPC [43], ILC [44]
Classifier for categorical & mixed data	SRG0-SRG5 methods, MDF search, frequency-based visualization in PC [41], 3-D SPC [52, 53]
Boolean and k-valued logical rules	MDF [54], MOEKA [51], SRG0-SRG5 [41], 3D SPC [52]
Classifiers for data with missing values	Visualization in Parallel Coordinates [40]

Example 3. Visualizing and analyzing ***k*-nearest neighbors** (k-NN) method: where given case A to be predicted is visualized along its k-NN by a selected GLC method. For example, in Concentric Circular Coordinates (CoC), case A is visualized as a straight vertical line with its k-NNs next to it making comparison and analysis of these cases easier. This visualization is reached by rotating CoC [48].

Example 4. **Representation learning** process where the automated generation of features or embedding vectors from raw data replaces manual feature engineering. VKD and GLC expands it to the visual *form of glyphs*. Users can analyze and classify cases as interpretable glyphs visually without computations beyond homogeneous raw input data [37,38].

Table 3. Meta level ML techniques to enhance ML with VKD/GLC methods.

ML tasks and methods	**GLC methods**
Worst-case accuracy estimates	GLC-L, PC, DSC1, DSC2 [45]
Labeling data	SCC, DCC [46]
Generating synthetic data	SCC, DCC [46], D&C [36]
Convert tabular data to images for classification by CNN	CPC [42], GLC-L [1], SPC-SF Glyphs [37.38]
Representation/feature learning	SPC-SF Glyphs [37.38], GLC-L [1]
Dimension reduction	Minimized HBs 2025 Aus
Model boosting	Divide & Classify (D&C) process and boosting models with FoL rules [36].
Transparency of models	GLC-L [1]
Explainability of models	Boolean models [51], mixed models [41, 52]. Linear models [1], Hyperblock models [39]
Trustworthiness of models	Model sureness metric and VKD with decreased training data [6].
Concept drift for n-D time series	Combined SPC with several CPC [47]
Class summary view	Glyphs [37.38], HB edges [39-41]

V. Conclusion

AI/LM integration with visualization is presented in the form of emerging Visual Knowledge Discovery ecosystem that is a set of interconnected algorithms, software projects, platforms, APIs, and stakeholders that co-evolve around a VKD shared technological base. Interpretable ML algorithms and lossless visualization of high-dimensional data are the core of VKD for the end users to discover trustworthy models. The paper presented VKD paradigm to foster trustworthy models. This includes the model interpretability issue with the proposed linear model explainability measure (LMEM) and abilities to generalize it for non-linear models. The focus of the future work will be exploring further alternative methods, simplifications, attribute encodings, larger dimensions and algorithmic enhancements to optimize user experience.

References

[1] Kovalerchuk B. Visual knowledge discovery and machine learning. Springer; 2018.

[2] Inselberg, A., Parallel Coordinates, Springer, 2009.

[3] Michie, D. Machine learning in the next five years. In: Proceedings of the third European working session on learning, 1988, 107–122, Pitman.

[4] Muggleton SH, Schmid U, Zeller C, Tamaddoni-Nezhad A, Besold T. Ultra-strong machine learning: comprehensibility of programs learned with ILP. Machine Learning. 2018, 119-140.

[5] AI Risks and Trustworthiness; NIST AI Resource Center: Gaithersburg, MA, USA, 2025

[6] Williams A, Kovalerchuk B. Quantifying AI Model Trust as a Model Sureness Measure by Bidirectional Active Processing and Visual Knowledge Discovery. Electronics. 2026 Jan 29;15(3):580.

[7] Uddin, S.; Lu, H.; Rahman, A.; Gao, J. A novel approach for assessing fairness in deployed machine learning algorithms. Scientific Reports. Nature 2024, 14, 17753

[8] Fan, X.; Tao, C. Towards Resilient and Efficient LLMs: A Comparative Study of Efficiency, Performance, and Adversarial Robustness, In Proc. of the 7th AI and Cloud Computing Conference, 2024; pp. 429–436.

[9] Roechner, P.; Marques, H.; Campello, R.; Zimek, A. Evaluating outlier probabilities: Assessing sharpness, refinement, and calibration using stratified and weighted measures.Data Min. Knowl. Discov. 2024, 38, 3719–3757.

[10] Bayram, F.; Ahmed, B.S. Towards trustworthy machine learning in production: An overview of the robustness in mlops approach. ACM Comput. Surv. 2025, 57, 121.

[11] Tēnn, K.P.; Chang, Y.W.; Chen, H.Y.; Fan, T.K.; Lin, T. Toward Trustworthy Artificial Intelligence: An Integrated Framework Approach Mitigating Threats. Computer 2024, 57, 57–67

[12] Wachter, S.; Mittelstadt, B.; Russell, C. Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR. Harv. J. Law Technol. 2018, *31*, 841–887.

[13] Rutinowski, J.; Klüttermann, S.; Endendyk, J.; Reining, C.; Müller, E. Benchmarking Trust: A Metric for Trustworthy Machine Learning. In Communications in Computer and Information Science, Proc. of 2nd World Conf. on Explainable AI, Springer, 2024; pp. 287–307.

[14] Smith, C. Trustworthy by Design. In Proceedings of the IEEE/ACM 46th International Conference on Software Engineering, 2024; pp. 1–4.

[15] Ribeiro MT, Singh S, Guestrin C. Why should i trust you? Explaining the predictions of any classifier. In: Proc. of the 22nd ACM SIGKDD intern, conf. on knowledge discovery and data mining 2016, pp. 1135-1144.

[16] Lundberg SM, Lee SI. A unified approach to interpreting model predictions. Advances in neural information processing systems. 2017;30.

[17] Kovalerchuk, B., Ahmad, M. A., & Teredesai, A. (2021). Survey of explainable machine learning with visual and granular methods beyond quasi-explanations. In Interpretable Artificial Intelligence: A Perspective of Granular Computing (pp. 217-267). Springer.

[18] Rudin C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature machine intelligence. 2019;1(5):206-215.

[19] Molnar, C. Interpretable machine learning: A guide for making black box models explainable. 2020, https://christophm.github.io/interpretable-ml-book/

[20] Slack D, Hilgard A, Singh S, Lakkaraju H. Reliable post hoc explanations: Modeling uncertainty in explainability. Advances in neural information processing systems. 2021;34:9391-404.

[21] Jean-Marie John-Mathews. Critical empirical study on black-box explanations in AI. 42nd Inter. Conf. on Information Systems, AIS, 2021.

[22] Flachaire E, Hué S, Laurent S, Hacheme G. Interpretable Machine Learning Using Partial Linear Models. Oxford Bulletin of Economics and Statistics. 2024 Jun;86(3):519-40.

[23] Schulte L, Ledel B, Herbold S. Studying the explanations for the automated prediction of bug and non-bug issues using LIME and SHAP. Empirical Software Engineering. 2024 Jul;29(4):93.

[24] Hashemi M, Darejeh A, Cruz F. Understanding user preferences in explainable artificial intelligence: A mapping function proposal. ACM Trans. on Intelligent Systems and Technology. 2025 Jun 10;16(4):1-37.

[25] Brandsæter A, Glad IK. Explainable artificial intelligence: How subsets of the training data affect a prediction. Stat. 2020 Dec;1050:1057.

[26] Hoaglin DC. Regressions are commonly misinterpreted. The. Stata Journal. 2016 Mar;16(1):5-22

[27] Mersha M, Lam K, Wood J, Alshami AK, Kalita J. Explainable artificial intelligence: A survey of needs, techniques, applications, and future direction. Neurocomputing. 2024;599:128111.

[28] Kovalerchuk B. Interpretable AI/ML for High-stakes Tasks with Human-in-the-loop: Critical Review and Future Trends, 2024. https://doi.org/10.21203/rs.3.rs-3989807/v1

[29] Lipton, Z. C. (2018). The Mythos of Model Interpretability. ACM Queue, 16(1), 31–57.

[30] Cashman D, Keller M, Jeon H, Kwon BC, Wang Q. A critical analysis of the usage of dimensionality reduction in four domains. IEEE Trans. on Visualization and Computer Graphics. Vol. 31, 10, 2025.pp: 9405 - 9423

[31] Stahnke J, Dörk M, Müller B, Thom A. Probing projections: Interaction techniques for interpreting arrangements and errors of dimensionality reductions. IEEE Trans. on visualiz. and computer graphics. 2015;22(1):

[32] Maszczyk T, Duch W. Support vector machines for visualization and dimensionality reduction. In: Intern. Conf. on Artificial Neural Networks 2008, pp. 346-356. Springer.

[33] Ortigossa ES, Dias FF, Nascimento DC, Nonato LG. Time Series Information Visualization—A Review of Approaches and Tools. IEEE Access. 2025 Sep 12.

[34] Byron L, Wattenberg M. Stacked graphs–geometry & aesthetics. IEEE transactions on visualization and computer graphics. 2008 Oct 24;14(6):1245-52

[35] Kovalerchuk B., Dunn A., Worland A., Wagle S. Interactive decision tree creation and enhancement with complete visualization for explainable modeling. In: Artificial Intelligence and Visualization: Advancing Visual Knowledge Discovery 2024, 3-40. Springer.

[36] Williams, A., Kovalerchuk, B. Boosting of Classification Models with Human-in-the-Loop Computational Visual Knowledge Discovery. In: Degen,H.,Ntoa,S.Eds, Artificial Intelligence in HCI. pp 391–412, Springer.

[37] Cutlip N, Kovalerchuk B. Lossless Interpretable Glyphs for Visual Knowledge Discovery in High-Dimensional Data. In: 27th International Conference Information Visualisation (IV) 2023, 292-299. IEEE.

[38] Cutlip N, Kovalerchuk B. Interpretable Visual Representation Learning with Animated Glyphs in Shifted Paired Coordinates. In2025 29th Intern. Conf. Information Visualisation (IV) 2025, pp. 197-204. IEEE.

[39] Huber L, Kovalerchuk B, Recaido C. Visual knowledge discovery with general line coordinates. In: Artificial Intelligence and Visualization: Advancing Visual Knowledge Discovery 2024, 159-202, Springer,

[40] Hayes D., Kovalerchuk B., Parallel Coordinates for Discovery of Interpretable Machine Learning Models, in: Artificial Intelligence and Visualization: Advancing Visual Knowledge Discovery, Springer, 2024, pp. 125-158.

[41] Kovalerchuk B., McCoy E., Explainable Machine Learning for Categorical and Mixed Data with Lossless Visualization, in: Artificial Intelligence and Visualization: Advancing Visual Knowledge Discovery, Springer, 2024, pp. 73-124.

[42] Kovalerchuk B, Kalla DC, Agarwal B. Deep Learning Image Recognition for Non-images. In: Integrating Artificial Intelligence and Visualization for Visual Knowledge Discovery 2022 (pp. 63-100). Springer, Cham.

[43] McDonald R, Kovalerchuk B. Non-linear visual knowledge discovery with elliptic paired coordinates. In: Integrating Artificial Intelligence and Visualization for Visual Knowledge Discovery 2022, 141-172. Springer.

[44] Kovalerchuk B., Phan H., Full High-Dimensional Intelligible Learning In 2-D Lossless Visualization Space, in: Artificial Intelligence and Visualization: Advancing Visual Knowledge Discovery, Springer, 2024, pp. 41-72.

[45] Recaido C, Kovalerchuk B. Visual Explainable Machine Learning for High-Stakes Decision-Making with Worst Case Estimates. In: Data Analysis and Optimization, 291-329. 2023, Springer.

[46] Williams A., Kovalerchuk B. Synthetic Data Generation and Automated Multidimensional Data Labeling for AI/ML in General and Circular Coordinates, 28th Intern. Conf. Information Visualisation, pp. 272-279, 2024, IEEE.

[47] Gâlmeanu, H., Kovalerchuk, B., Andonie, R. Interactive Discovery of Concept Drift with Lossless Visualization in Machine Learning. In: Degen, H., Ntoa, S. (eds) Artificial Intelligence in HCI. HCII 2025. Lecture Notes in Computer Science, vol 15822. pp 310–324 Springer,

[48] Williams A, Kovalerchuk B. High-Dimensional Data Classification in Concentric Coordinates. In: 29th Intern. Conf. Information Visualisation (IV) 2025, pp. 217-224. IEEE

[49] Salih AM, Raisi‐Estabragh Z, Galazzo IB, Radeva P, Petersen SE, Lekadir K, Menegaz G. A perspective on explainable artificial intelligence methods: SHAP and LIME. Advanced Intelligent Systems. 2025;7(1):2400304

[50] Reddy P, Gujral AS. Improving Neural Network Efficiency Using Piecewise Linear Approximation of Activation Functions. InThe International FLAIRS Conference Proceedings 2025 May 14 (Vol. 38).

[51] Huber H, Kovalerchuk B., Monotone Functions and Models for Explanation of Machine Learning Models, 28th International Conference Information Visualisation, pp.227-235, 2024, IEEE.

[52] Kovalerchuk B., Martinez J., Fleagle M., Visual Explanation of Machine Learning Models in Shifted Paired Coordinates in 3D, 28th Intern. Conf. Information Visualisation, pp.258-265, 2024, IEEE.

[53] Martinez J., Kovalerchuk B., General Line Coordinates in 3D, In 27th Intern. Conf. Information Visualisation, IEEE, 2023, pp 308-315,

[54] Kovalerchuk, B., Delizy, F., Riggs, L., Visual Data Mining and Discovery with Binarized Vectors, in: Data Mining: Foundation and Intelligent Paradigms (2012) 24: 135-156

[55] Kovalerchuk B., Williams, A., Lossless Visual Analysis of Multidimensional Data in Collocated Paired Coordinats for Machine Learning, HCI Intrnational Conference, 2026, LNAI, Springer (in print).