



Improving Counterfactual Truthfulness for Molecular Property Prediction Through Uncertainty Quantification

Jonas Teufel¹ , Annika Leinweber¹, and Pascal Friederich^{1,2}  

¹ Institute of Theoretical Informatics, Karlsruhe Institute of Technology,
Kaiserstr. 12, 76131 Karlsruhe, Germany
{jonas.teufel,pascal.friederich}@kit.edu

² Institute of Nanotechnology, Karlsruhe Institute of Technology,
Kaiserstr. 12, 76131 Karlsruhe, Germany

Abstract. Explainable AI (xAI) interventions aim to improve interpretability for complex black-box models, not only to improve user trust but also as a means to extract scientific insights from high-performing predictive systems. In molecular property prediction, counterfactual explanations offer a way to understand predictive behavior by highlighting which minimal perturbations in the input molecular structure cause the greatest deviation in the predicted property. However, such explanations only allow for meaningful scientific insights if they reflect the distribution of the true underlying property—a feature we define as counterfactual truthfulness. To increase this truthfulness, we propose the integration of uncertainty estimation techniques to filter counterfactual candidates with high predicted uncertainty. Through computational experiments with synthetic and real-world datasets, we demonstrate that traditional uncertainty estimation methods, such as ensembles and mean-variance estimation, can already substantially reduce the average prediction error and increase counterfactual truthfulness, especially for out-of-distribution settings. Our results highlight the importance and potential impact of incorporating uncertainty estimation into explainability methods, especially considering the relatively high effectiveness of low-effort interventions like model ensembles.

Keywords: Counterfactual Explanations · Truthfulness · Graph Neural Networks · Uncertainty Estimation · Molecular Property Prediction

1 Introduction

Recent advances in the study of artificial intelligence (AI) have revolutionized various branches of society, industry, and science. Despite their numerous advantages, the opaque black-box nature of modern AI methods remains a challenge. Although complex neural network models often display superior predictive

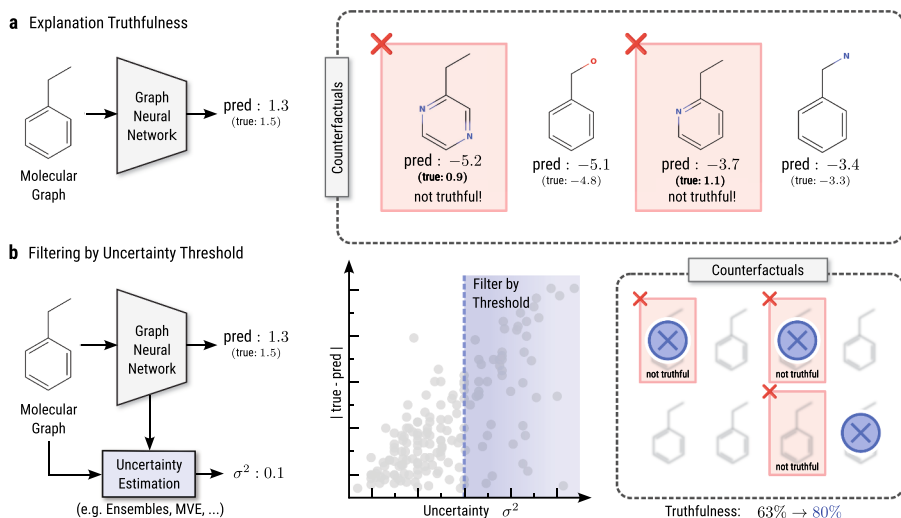


Fig. 1. **a** Truthful explanations should not only reflect the model’s behavior but also the properties of the underlying true data distribution. **b** Uncertainty quantification methods predict an additional uncertainty value as an approximation of the model’s prediction error. By filtering high-uncertainty elements, it is possible to reduce the cumulative error and, by extension, the fraction of truthful counterfactuals of the remaining set.

performance, their inner workings remain largely intransparent to humans. Explainable AI (xAI) aims to address these shortcomings by developing methods to better understand the inner workings of these complex models.

Traditionally, xAI methods are meant to improve trust in human-AI relationships, provide tools for model debugging, and ensure regulatory compliance [9]. More recently, xAI has been proposed as a potential source of new scientific insight [24, 40, 41, 51]. This potential of gaining new insights primarily concerns tasks about which little to no prior human knowledge exists. By elucidating the behavior of high-performing models in complex property prediction tasks, xAI can offer insights not only into the model’s behavior but, by extension, into the underlying rules and relationships governing the data itself. However, to gain meaningful insights, the given explanations must be *truthful* regarding this underlying data distribution. However, for these explanations to yield meaningful insights, they must accurately reflect the true data distribution. This imposes a more stringent requirement for explanations: they must be valid not only in terms of the model’s behavior but also with respect to the predicted property itself.

In this work, we explore counterfactual explainability within chemistry and materials science—a domain where insights derived from XAI would have a substantial impact to accelerate scientific discovery. In short, counterfactual explanations locally explain the model’s behavior by constructing multiple “what if?”

scenarios of minimally perturbed input configurations that cause large deviations in the model’s prediction. By itself, a counterfactual only has to explain the model’s behavior, regardless if that behavior reflects the underlying property, causing significant conceptual overlap between the counterfactuals and adversarial examples [10]. As an extension, we define a *truthful* counterfactual as one that satisfies constraints regarding the model and the underlying ground truth—causing a large deviation of the prediction while maintaining low prediction error (see Fig. 1).

Given the general unavailability of ground truth labels for counterfactual samples, we propose uncertainty quantification methods as a means to approximate prediction error and ultimately improve overall counterfactual truthfulness by filtering high-uncertainty explanations. We empirically investigate various common methods of uncertainty quantification and find that an ensemble of mean-variance estimators (MVE) yields the greatest improvement of relative model error and can substantially improve counterfactual truthfulness. Qualitative results affirm these findings, showing that uncertainty-based filtering removes unlikely molecular configurations that lie outside the training distribution. Our results underscore the potential benefits of integrating uncertainty estimation into explainability methods, such as counterfactual explanations.

2 Related Work

Graph Counterfactual Explanations. Insights from social science indicate that humans prefer explanations to be *contrastive*—to explain why something happened *instead of* something else [26]. Counterfactuals aim to provide such contrastive explanations by constructing hypothetical “what if?” scenarios to show which small perturbations to a given input sample would have resulted in a significant deviation from the original prediction outcome.

While Verma *et al.* [44] present an extensive general review on the topic of counterfactual explanations across different data modalities, Prado-Romero *et al.* [30] explore specifically counterfactual explanations in the graph processing domain. The authors find that the existing approaches can be categorized by which kinds of perturbations to the input graph are considered. Many existing methods create perturbations using masking strategies on the node-, edge- or feature-level in which masks are optimized to maximize output deviations [3, 23, 39]. However, masking-based strategies often yield uninformative explanations for molecular property prediction. In this context, it is more insightful to perturb the molecular graph by adding or removing bonds and atoms [38]. Some authors successfully adopt such approaches for molecular property predictions [27, 29, 46]. One particular difficulty for these kinds of approaches is the necessity of including domain knowledge to ensure that modifications result in a valid graph structure (e.g. chemically feasible molecules). In one example, Numeroso and Bacciu [29], propose to train an external reinforcement learning agent to propose suitable graph modifications resulting in counterfactual candidates for molecular property predictions. In this case, the authors also introduce domain knowledge by restricting the action space of the agent to chemically feasible modifications.

Uncertainty Quantification. Predictive machine learning models often encounter uncertainty from various sources, including, for example, inherent measurement noise (aleatoric uncertainty) or regions of the input space insufficiently covered in the training distribution (epistemic uncertainty). Consequently, a model’s predictions may be more accurate for some input samples than for others. Uncertainty quantification methods aim to measure this variability, identifying those samples that a model can predict with greater confidence [1, 11].

Similarly to the broader field of xAI, one aim of uncertainty quantification is to improve user trust by indicating the reliability of a prediction [35]. Beyond uncertainty quantification for target predictions, Longo *et al.* [22] propose to introduce elements of uncertainty estimation on the explanation level as well.

Traditionally used methods for uncertainty quantification include the joint prediction of a distribution’s mean and variance (MVE) [28], assessing the variance between the predictions of a Deep Ensemble [16] and using bayesian neural networks (BNNs) [4, 12, 42] which aim to directly predict an output distribution rather than individual values. More recent alternatives include stochastic weight averaging gaussians (SWAG) [25] and the idea of Repulsive Ensembles [7, 43] as an extension to Deep Ensembles built on the general framework of particle based variational inference (ParVI) [20, 21] introducing explicit member diversification.

In the domain of molecular property prediction, Hirschfeld *et al.* [13] and Scalia *et al.* [32] independently investigate the performance of various traditional uncertainty quantification methods across many standard property prediction datasets. Busk *et al.* specifically investigate uncertainty quantification using an ensemble of graph neural networks [6].

Uncertainty Quantification and Counterfactuals. Using xAI to gain new insights into the underlying properties of the data distribution requires the given explanations to be *truthful* regarding the *true* property values. In the same context, Freiesleben [10] addresses the conceptual distinction between counterfactual explanations and adversarial examples. Although essentially based on the same optimization objective, the author argues that adversarial examples necessitate a misprediction while counterfactual explanations should be different—yet still correct.

While uncertainty quantification in the context of counterfactual explanations remains largely unexplored, we find Delaney *et al.* [8] to use uncertainty quantification methods as a possible measure to increase counterfactual reliability for image classification tasks. In terms of UQ interventions, the authors explore Trust Scores and Monte Carlo dropout, finding Trust Scores to be an effective measure. Schut *et al.* [33] propose the direct optimization of an ensemble-based uncertainty measure as a secondary objective for the generation of *realistic* counterfactuals for image classification. In another work, Antorán *et al.* [2] introduce Counterfactual Latent Uncertainty Explanations (CLUE),

which is subsequently extended to δ -CLUE [18] and GLAM-CLUE [19]. Instead of employing uncertainty quantification to improve counterfactual explainability, CLUE aims to use counterfactual explanations to explain uncertainty estimates in probabilistic models—effectively explaining why certain inputs are more uncertain than others.

3 Method

In this work, we explore the generation of counterfactual samples x' for molecular property prediction tasks, whereby a graph neural network model is trained to regress a continuous property y of a given molecular graph x . To gain meaningful insights on the underlying property, we specifically focus on the generation of *truthful* counterfactuals which maximize the prediction difference $|\hat{y} - \hat{y}'|$ between original prediction \hat{y} and counterfactual prediction \hat{y}' while maintaining a minimal ground truth error $|y' - \hat{y}'|$.

3.1 Graph Neural Network Regressors

We represent each molecule as a generic graph structure $x = (\mathcal{N}, \mathcal{E}, \mathbf{V}^{(0)}, \mathbf{U}^{(0)}) \in \mathcal{X}$ defined by a set of N node indices $\mathcal{N} = \{1, \dots, N\}$ and a list of E edge tuples $\mathcal{E} \subseteq N \times N$ where a tuple $(i, j) \in \mathcal{E}$ indicates an edge between nodes i and j . The nodes of this graph structure represent the atoms of the molecule and the edges represent the chemical bonds between the atoms. Furthermore, each graph structure consists of an initial node feature tensor $\mathbf{V}^{(0)} \in \mathbb{R}^{N \times V}$ and an initial edge feature tensor $\mathbf{U}^{(0)} \in \mathbb{R}^{E \times U}$.

In the case of molecular graphs, the node features contain a one-hot encoding of the atom type, the atomic weight, and the charge, whereas the edge features contain a one-hot encoding of the bond type. For a given dataset of molecules annotated with continuous target values $y \in \mathbb{R}$, the aim is to train a graph neural network regressor

$$f_{\theta} : \mathcal{X} \rightarrow \mathbb{R}; \quad (\mathcal{N}, \mathcal{E}, \mathbf{V}^{(0)}, \mathbf{U}^{(0)}) \mapsto \hat{y} \quad (1)$$

with learnable parameters θ to find an optimal set of parameters

$$\theta^* = \arg \min_{\theta} \sum_{x \in \mathcal{X}} (y - f_{\theta}(x))^2 \quad (2)$$

that minimizes the mean-squared error between the predicted value \hat{y} and target y value.

3.2 Molecular Counterfactual Generation

Counterfactual explanations map a model’s local decision boundary by producing a set of minimally perturbed input instances that induce maximal predictive

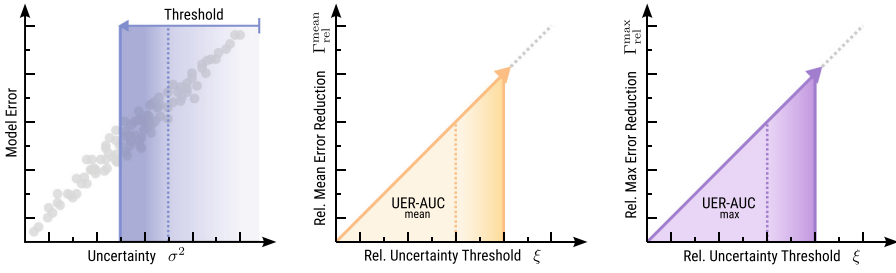
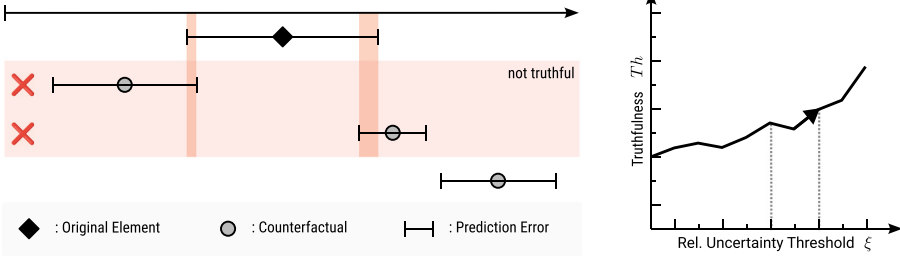
a Uncertainty-based Error Reduction**b** Truthful Counterfactuals

Fig. 2. **a** Evaluation of the uncertainty-based error reduction over many different thresholds yields characteristic error reduction curves. The Area under the uncertainty error reduction curve (UER-AUC) provides a generic metric for the error reduction potential independent of a specific threshold choice. **b** Truthful counterfactuals are defined as those whose prediction error interval does not overlap with that of its corresponding original element. Besides a reduction of the cumulative error, filtering by uncertainty thresholds may also increase the relative fraction of truthful counterfactuals.

divergence, thereby revealing which kinds of modifications the model is especially sensitive toward.

Given the combination (x, \hat{y}) of an original input element x and its corresponding model prediction \hat{y} , a counterfactual sample (x', \hat{y}') consists of input samples x' which are minimally

$$\min_{x'} \text{dist}(x', x) \quad (3)$$

different from the original input. At the same time, these minimally perturbed input samples should cause a large deviation

$$\max_{x'} \text{dist}(\hat{y}', \hat{y}) \quad (4)$$

in the model's prediction.

We generate counterfactual samples according to the given constraints by adopting a procedure similar to that presented by Numero and Bacciu [29]. However, we omit the training of a reinforcement learning agent to induce the

local changes of the molecular structure and opt for a complete enumeration of the entire k -edit neighborhood instead. Due to the limited number of chemically valid modifications and the relatively small size of molecular graphs, we find it computationally feasible to generate all possible modifications to a given input molecule x . As possible modifications, we consider the addition, deletion, and substitution of individual atoms and bonds that satisfy the constraints of atomic valence. Subsequently, the predictive model f_θ is used to obtain the predicted values for all the perturbed graph structures. The structures are then ranked according to the mean absolute prediction difference

$$\text{dist}(\hat{y}', \hat{y}) = |\hat{y}' - \hat{y}| \quad (5)$$

regarding the original prediction \hat{y} . We finally choose the 10 elements with the highest prediction difference to be presented as counterfactual explanations.

At this point, it is worth noting that other possible variations of choosing counterfactual explanations exist. Instead of using the criteria of absolute distance, depending on the use case, it might make sense to select counterfactuals only among those samples with monotonically higher or lower predicted values.

3.3 Counterfactual Truthfulness

A counterfactual explanation x' has to be a minimal perturbation of the original sample x while causing a large deviation $|\hat{y} - \hat{y}'|$ in the model's prediction. To gain meaningful insight from such counterfactual explanations and to distinguish them from mere adversarial examples [10], we impose the additional restriction of *truthfulness*. We define a truthful counterfactual to additionally maintain a low error $|y' - \hat{y}'|$ regarding its ground truth label y' .

For classification problems, we would understand a truthful counterfactual not only to flip the predicted label but to also correctly be associated with that label. For the regression case, there may exist various equally useful definitions of counterfactual truthfulness. In this context, we define a regression counterfactual as truthful if its ground truth error interval does not overlap with the error interval of the original prediction (see Fig. 2). This definition ensures that there is at least *some* predictive divergence with the predicted directionality.

For a given original sample, its absolute ground truth error

$$\epsilon = |y - \hat{y}| \quad (6)$$

is calculated as the absolute difference of the true value y and the predicted label \hat{y} . The ground truth error

$$\epsilon' = |y' - \hat{y}'| \quad (7)$$

of a counterfactual sample x' can be calculated accordingly. We subsequently define the truthfulness

$$\text{tr}(x') = \begin{cases} 1 & [y' - \epsilon', y' + \epsilon'] \cap [y - \epsilon, y + \epsilon] = \emptyset \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

of an individual counterfactual as a binary property which is fulfilled if its ground truth error interval does not overlap with the error interval of the original sample.

Beyond the truthfulness of individual counterfactual samples, we are primarily interested in the average truthfulness across a whole set $\mathcal{X}' \subset \mathcal{X}$ of counterfactuals. We, therefore, define the relative truthfulness

$$\text{Tr}(\mathcal{X}') = \frac{1}{|\mathcal{X}'|} \sum_{x' \in \mathcal{X}'} \text{tr}(x') \in [0, 1] \quad (9)$$

for a set \mathcal{X}' of counterfactuals as the ratio of individual truthful counterfactuals it contains.

At this point, it should be noted that evaluating counterfactual truthfulness proves difficult. Since the generated counterfactual samples generally aren't contained in existing datasets, evaluating the truthfulness would not only require ground truth labels but rather a ground truth oracle. Consequently, truthfulness can only be evaluated for a small selection of tasks for which such an oracle exists.

3.4 Error Reduction Through Uncertainty Thresholding

For the given definition of truthfulness, one viable method of improving the relative truthfulness $\text{Tr}(\mathcal{X}')$ is to filter counterfactuals with especially large error intervals. Since it is generally impossible to infer the true label, and by extension the truthfulness, of a given input x in practice, an alternative is to approximate the ground truth error by means of uncertainty quantification (UQ). If the predicted uncertainty proves to be a suitable approximation of the true error, filtering high-uncertainty counterfactuals should have the same effect of improving the relative truthfulness.

This objective can be framed as an overall reduction of the cumulative error

$$\Gamma^g = g(\{|\hat{y}_i - y_i| : x_i \in \mathcal{X}^\circ\}) \quad (10)$$

for a given set $\mathcal{X}^\circ \subset \mathcal{X}$ of input elements, where $g(\cdot)$ is some function that accumulates individual error values (e.g. mean, median, max).

In the context of uncertainty quantification, each sample x is additionally assigned a predicted uncertainty value σ^2 . Ideally, a high uncertainty value indicates a potential error in the model prediction, while a low value indicates the prediction to be likely correct. By filtering individual samples with high predicted uncertainties, it should, therefore, be possible to reduce the cumulative error Γ^g among the remaining elements. For this purpose, we can define the absolute cumulative error

$$\Gamma^g(\xi) = g\left(\left\{|\hat{y}_i - y_i| : x_i \in \mathcal{X}^\circ \mid \frac{\sigma_i^2}{\sigma_{max}^2} < \xi\right\}\right) \quad (11)$$

as a function of the relative uncertainty threshold $\xi \in [0, 1]$ used for the filtering.

This definition of cumulative error faces two issues. Firstly, values of the cumulative error will strongly depend on the specific uncertainty threshold ξ that was chosen. Secondly, the absolute error scales will be vastly different between different tasks and model performances, therefore not being comparable. Consequently, we propose the *area under the uncertainty error reduction curve (UER-AUC)* as a metric to assess the potential for uncertainty filtering-based error reduction that is comparable across different error scales. To compute the metric, we define the relative cumulative error reduction

$$\Delta\Gamma_{rel}^g(\xi) = \frac{\Gamma^g(1) - \Gamma^g(\xi)}{\max_{\xi} \Gamma^g(\xi)} \in [0, 1] \quad (12)$$

which is a value in the range $[0, 1]$, where 0 indicates no error reduction while 1 indicates a 100% error reduction. We finally define the UER-AUC_g as the area under the curve of the relative error reduction $\Delta\Gamma_{rel}^g(\xi)$ as a function of the relative uncertainty threshold ξ . Consequently, the proposed metric is independent of any specific threshold and comparable across different error ranges as both the uncertainty threshold ξ and the relative error reduction $\Delta\Gamma_{rel}^g$ are normalized to the range $[0, 1]$.

In terms of accumulation functions g , we primarily investigate the mean and the maximum, resulting in the two metrics $\text{UER-AUC}_{\text{mean}}$ and $\text{UER-AUC}_{\text{max}}$. Figure 2a illustrates a simple intuition about these metrics: A perfect correlation between uncertainty and model error will result in a UER-AUC of 0.5. Likewise, a UER-AUC of 0 would be the result of a non-existent correlation between uncertainty and error.

4 Computational Experiments

Computational experiments are structured in two major parts: In the first part, we systematically investigate the general error reduction potential of uncertainty estimation methods for different graph neural network architectures, different uncertainty estimation methods, various out-of-distribution settings, and a range of different datasets. In the second part, we consider the use of uncertainty quantification methods in the context of counterfactual explanations and their effect on overall counterfactual truthfulness as previously defined in Sect. 3.

4.1 Uncertainty Quantification Methods and Metrics

Uncertainty Quantification Methods. As part of the computational experiments, we compare the following uncertainty quantification methods.

Deep Ensembles (DE). We train 3 separate models with bootstrap aggregation, whereby the training data is sampled with replacement. The overall prediction is subsequently obtained as the mean of the individual model outputs, while the standard deviation of the individual predictions is used as an estimate of the uncertainty.

Mean Variance Estimation (MVE). The base model architecture is augmented to predict not only the target value \hat{y} but also an uncertainty term σ^2 by adding additional fully connected layers to the final prediction network [28]. The training loss

$$\mathcal{L}_{\text{MVE}} = \frac{1}{N} \sum_i^N \frac{\text{sg}(\sigma_i^{2\beta})}{2} \cdot \left(\frac{(y_i - \hat{y}_i)^2}{\sigma_i^2} + \log(\sigma_i^2) \right) \quad (13)$$

is augmented to optimize both terms at the same time. We specifically integrate the modification proposed by Seitzer *et al.* [34], which scales the loss by an additional factor of $\sigma^{2\beta}$ but without contributing to the gradient. Furthermore, during training, we follow best practices described by Sluijterman *et al.* [36] by using gradient clipping and including an MSE warm-up period before switching to the MVE loss. By combining these measures, we substantially improve the performance degradation otherwise reported in the literature.

Ensemble of mean-variance estimators (DE+MVE). We combine deep ensembles and mean-variance estimation by constructing an ensemble of 3 independent MVE models, each of which predicts an individual mean and standard deviation, as proposed by Busk *et al.* [6]. The total uncertainty

$$\sigma^2 = \frac{1}{2} (\sigma_{\text{DE}}^2 + \bar{\sigma}_{\text{MVE}}^2) \quad (14)$$

is calculated as the average of the ensemble uncertainty and the mean MVE uncertainty.

Stochastic Weight Averaging Gaussian (SWAG). The training process is augmented to store snapshots of the model weights during the last 25 epochs. This history of model weights is then used to calculate a mean weight vector μ_θ and a covariance matrix Σ_θ such that a new set of model weights can approximately be obtained by drawing from a gaussian distribution $\theta \sim \mathcal{N}(\mu_\theta, \Sigma_\theta)$. During inference, we sample 50 distinct model weights from this distribution and obtain the target value prediction as the mean of the individual predictions and an uncertainty estimate as the standard deviation.

Trust Scores (TS). Unlike the previously described UQ methods, trust scores are independent of the predictive model and provide an uncertainty estimate based directly on the training data [8]. Originally introduced for classification problems, the trust score for a given input element x is calculated as the fraction

$$T = \frac{\text{dist}(x, x_s)}{\text{dist}(x, x_o)} \quad (15)$$

between the distances of the closest training element x_s of the same class and the closest training element x_o of a different class. We adapt this approach for regression tasks by using the distance to the closest element. This definition relies on the existence of a suitable distance metric $\text{dist}(x_i, x_j)$ between two input

Table 1. Test set results of 5 independent repetitions of computational experiments on the ClogP dataset regarding different model architectures and uncertainty quantification methods. Normal case numbers are the average result, and lower case gray numbers are the standard deviation. For each combination of model and UQ method, the best results are highlighted in bold, and the second-best results are underlined.

Model	UQ Method	$R^2 \uparrow$	$\rho \uparrow$	UER-AUC \uparrow <i>mean</i>	UER-AUC \uparrow <i>max</i>	RLL \uparrow
—	Random	1.00±0.00	0.01±0.03	0.01±0.04	0.10±0.10	—
GCN	DE	1.00±0.00	0.41±0.17	0.21±0.06	0.36±0.18	0.75±0.02
	MVE	0.99±0.01	0.45±0.08	0.20±0.04	0.63 ±0.18	<u>0.77</u> ±0.03
	DE+MVE	1.00±0.00	0.55 ±0.10	0.25 ±0.02	<u>0.58</u> ±0.20	0.78 ±0.00
	SWAG	1.00±0.00	<u>0.50</u> ±0.08	<u>0.21</u> ±0.03	0.50±0.23	0.56±0.11
	TS _{eucl.}	0.99±0.01	0.15±0.17	0.15±0.09	0.43±0.26	0.69±0.10
	TS _{tanim.}	1.00±0.00	0.15±0.05	0.11±0.04	0.12±0.10	0.39±0.33
GATv2	DE	1.00±0.00	<u>0.51</u> ±0.11	0.22±0.04	0.63±0.28	<u>0.73</u> ±0.05
	MVE	0.98±0.03	0.48±0.08	<u>0.28</u> ±0.06	<u>0.72</u> ±0.15	0.72±0.08
	DE+MVE	1.00±0.00	0.64 ±0.15	0.34 ±0.02	0.75 ±0.09	0.82 ±0.03
	SWAG	0.99±0.00	0.49±0.16	0.21±0.02	0.61±0.21	−0.06±0.47
	TS _{eucl.}	1.00±0.00	0.07±0.04	0.17±0.07	0.59±0.00	0.64±0.01
	TS _{tanim.}	1.00±0.00	0.20±0.04	0.13±0.03	0.10±0.07	0.59±0.06
GIN	DE	0.99±0.01	<u>0.62</u> ±0.17	<u>0.27</u> ±0.06	<u>0.70</u> ±0.11	0.80 ±0.04
	MVE	0.99±0.01	0.48±0.11	0.22±0.05	0.56±0.22	0.75±0.05
	DE+MVE	1.00±0.00	0.63 ±0.05	0.29 ±0.03	0.70 ±0.15	<u>0.78</u> ±0.01
	SWAG	0.98±0.02	0.58±0.20	0.23±0.07	0.58±0.08	0.02±0.43
	TS _{eucl.}	0.99±0.00	0.15±0.12	0.17±0.05	0.45±0.22	0.64±0.03
	TS _{tanim.}	0.99±0.00	0.17±0.08	0.13±0.05	0.11±0.11	0.52±0.12

elements. In this study, we examine two distance metrics for comparing input elements. The first is the Tanimoto distance, which is calculated as the Jaccard distance between two Morgan fingerprint representations of two molecules. The second is the Euclidean distance, which is measured between the graph embeddings generated by an intermediate layer of the graph neural network models.

Uncertainty Calibration. After training, we apply uncertainty calibration to each UQ method to align the predicted uncertainties with the scale of the actual prediction errors. For this purpose, we use a held-out validation set containing 10% of the data to subsequently fit an isotonic regression model.

Uncertainty Quantification Metrics. We evaluate the aforementioned UQ methods with the following metrics.

Uncertainty-Error Correlation ρ . The Pearson correlation coefficient between the absolute prediction errors $|\hat{y} - y|$ and the predicted uncertainties σ^2 on the elements of the test set.

Error Reduction Potential UER-AUC. As described in Sect. 3, the UER-AUC is the area under the curve that maps relative error reduction to relative uncertainty thresholds. For each uncertainty threshold, all elements with higher predicted uncertainty are omitted from the test set. The relative error reduction describes the reduction of the cumulative error of the remaining elements relative to the full set.

Relative Log Likelihood RLL. Following the work of Kellner and Ceriotti [14] we use the Relative Log Likelihood

$$\text{RLL} = \frac{\sum_i \text{NLL}(\hat{y}_i - y_i, \sigma_i^2) - \text{NLL}(\hat{y}_i - y_i, \text{RMSE})}{\sum_i \text{NLL}(\hat{y}_i - y_i, |\hat{y}_i - y_i|) - \text{NLL}(\hat{y}_i - y_i, \text{RMSE})} \quad (16)$$

which standardizes the arbitrary range of the Negative Log Likelihood

$$\text{NLL}(\Delta y, \sigma^2) = \frac{1}{2} \left(\frac{\Delta y^2}{\sigma^2} + \log 2\pi\sigma^2 \right) \quad (17)$$

into a more interpretable range $(-\infty, 1]$.

4.2 Experiments on Error Reduction Potential

Impact of GNN Model and UQ Method on Error Reduction. In this first experiment, we evaluate the impact of the model architecture and uncertainty quantification method on uncertainty-based error reduction. The experiment is based on the ClogP dataset, which consists of roughly 11k small molecules annotated with values of Crippen’s logP [47] calculated by RDKit [17]. This logP value is an algorithmically calculated and deterministic property—making it possible to near-perfectly regress it with machine learning models.

In terms of model choice, we compare three standard GNN architectures based on the GCN [15], GATv2 [5], and GIN [50] layer types, respectively. For each repetition of the experiment, we randomly choose 10% of the dataset as the test set, 10% as the calibration set and train the model on the remaining. Therefore, the test set can be considered IID w.r.t. to the training distribution.

Table 1 shows the results of the first experiment. A “Rando” baseline, generating random uncertainty values, was included as a control. As expected, this baseline demonstrates negligible error reduction, reflecting the absence of correlation between assigned uncertainty and prediction error. In contrast, the remaining uncertainty quantification methods exhibit varying degrees of error reduction.

Using trust scores with the input-based Tanimoto distance yields substantially worse results than the embedding-based Euclidean distance. Contrary to the encouraging results of Delaney *et al.* [8], we believe trust scores underperform in this particular application due to the challenge of defining suitable distance metrics on graph-structured data [48].

Overall, we find deep ensembles, mean-variance estimation, and a combination thereof to work the best in terms of error reduction potential, as well as relative log likelihood. Out of these methods, we observe a slight advantage in

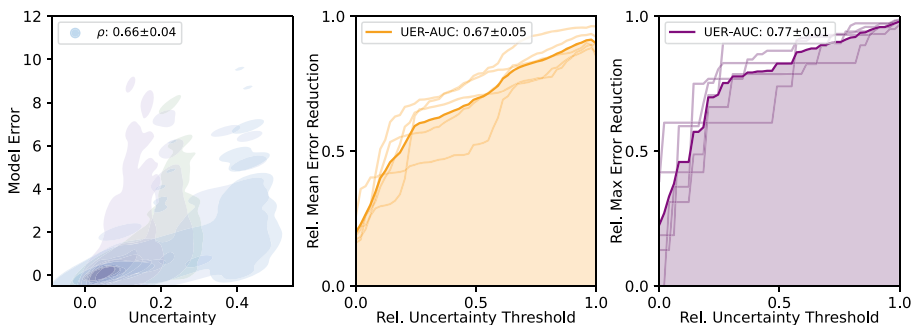


Fig. 3. Results for 5 independent repetitions of a GATv2 trained on the ClogP dataset and uncertainties estimated with a combination of ensembles and mean variance estimation in the OOD-Value scenario. Panels from left to right illustrate the correlation between the predicted uncertainty & model error, the mean error reduction potential, and the max error reduction potential through filtering by uncertainty thresholds. Faint lines represent the results of individual runs; bold lines represent the overall average.

mean error reduction for the combined ensemble and mean-variance estimation approach.

Moreover, regarding the different model architectures (GCN, GATv2, and GIN), we observe comparable results, both in terms of predictive performance ($R^2 \geq 0.99$) and in terms of uncertainty quantification methods. Based on these initial observations, model architecture appears to have a limited effect on the relative performance of the uncertainty quantification methods. Consequently, subsequent experiments were conducted using the GATv2 architecture, which exhibited the highest mean error reduction potential in this experiment.

Out-of-Distribution Effect on Error Reduction. The previous experiment examined the error reduction potential on a randomly sampled IID test set of the ClogP dataset. However, a critical aspect of counterfactual analysis involves identifying input perturbations that yield out-of-distribution (OOD) samples. To address this, we established two OOD scenarios for the ClogP dataset. The first, designated *OOD-Struct*, employs a scaffold split, where the test set comprises molecules with structural scaffolds absent from the training set. The second, *OOD-Value*, involves a split where the test set contains approximately the 10% most extreme target values, not represented in the training set. Due to the results of the previous experiments, for each scenario, we restrict experiments to use the GATv2 model architecture and compare uncertainty estimation based on ensembles, mean variance estimation, and the combination thereof.

Table 2 reports the results of the second experiment. For the OOD-Struct scenario, we observe slightly worse results than for the IID case. All three methods show lower correlation, error reduction potential, and relative log likelihood.

Table 2. Test set results of 5 independent repetitions of computational experiments on the ClogP dataset regarding different out-of-distribution scenarios and uncertainty quantification methods. The best result for each scenario is highlighted in bold, and the second-best result is underlined. Results were obtained based on a GATv2 model architecture.

Scenario	UQ Method	$R^2 \uparrow$	$\rho \uparrow$	UER-AUC \uparrow <small><i>mean</i></small>	UER-AUC \uparrow <small><i>max</i></small>	RLL \uparrow
OOD struct	DE	1.00 ± 0.00	<u>0.45 ± 0.05</u>	0.23 ± 0.04	0.66 ± 0.07	<u>0.34 ± 0.06</u>
	MVE	0.99 ± 0.00	0.32 ± 0.15	0.14 ± 0.06	0.20 ± 0.11	0.41 ± 0.03
	DE+MVE	1.00 ± 0.00	0.46 ± 0.05	<u>0.21 ± 0.04</u>	<u>0.42 ± 0.17</u>	0.55 ± 0.00
OOD value	DE	0.97 ± 0.01	<u>0.62 ± 0.07</u>	0.71 ± 0.10	0.82 ± 0.07	<u>-3.79 ± 2.92</u>
	MVE	0.99 ± 0.00	0.50 ± 0.08	0.51 ± 0.11	0.36 ± 0.27	-8.04 ± 9.72
	DE+MVE	0.98 ± 0.00	0.66 ± 0.04	<u>0.67 ± 0.05</u>	<u>0.77 ± 0.01</u>	-1.49 ± 1.09

Conversely, the OOD-Value scenario exhibited substantial performance gains relative to the IID case. Mean and max error reduction potential cross decisively exceed the $\text{UER-AUC} \geq 0.5$ threshold. Only the negative RLL values indicate poorly calibrated uncertainty estimates with respect to the actual prediction error. This is to be expected since the calibration set was sampled IID while the test set contains previously unseen target values—likely resulting in vastly different error scales.

When comparing the different UQ methods, the ensembles by themselves and the combination of ensembles and MVE seem to perform equally well. For both scenarios, OOD-struct and OOD-value, the ensembles seem to offer higher error reduction potential, while the combination seems to offer better calibrated uncertainty estimates, as indicated by the higher RLL values.

In summary, uncertainty-based filtering demonstrates a moderate error reduction effect on in-distribution data and structural outliers. Notably, the error reduction potential increases substantially under a distributional shift of the target values (see Fig. 3). These results provide a foundation for filtering counterfactuals, where perturbations can be expected to create outliers with respect to both structure and target value.

Error Reduction on Real-World Datasets. Previous experiments were based on the ClogP dataset, which is a deterministically computable property and, therefore, relatively easy to regress. To assess the generalizability of these findings to more complex scenarios, computational experiments were conducted on multiple properties derived from the AqSolDB [37], Lipop [49], COMPAS [45], and QM9 [31] datasets. Based on the results of previous experiments, we use the GATv2 model to predict each property and a combination of ensembles and mean-variance estimation for the uncertainty quantification.

Table 3 presents the results for the real-world property regression datasets. Despite varying levels of predictivity ($R^2 \in [0.74, 0.99]$) for the different datasets, some degree of error reduction can be reported for each one ($\text{UER-AUC}_{\text{mean}} \in$

Table 3. Test set results of 5 independent repetitions of computational experiments to evaluate uncertainty-based error reduction on various molecular property prediction datasets. The first row represents the previously introduced deterministic CLogP graph regression task, and the following rows represent various real-world molecular property regression datasets. Results are obtained by a GATv2 graph neural network and uncertainties are estimated by a method combining deep ensembles and mean-variance estimation.

Dataset	Property	$R^2 \uparrow$	$\rho \uparrow$	UER-AUC \uparrow <i>mean</i>	UER-AUC \uparrow <i>max</i>	RLL \uparrow
ClogP	logP	1.00 \pm 0.00	0.58 \pm 0.17	0.27 \pm 0.05	0.66 \pm 0.22	0.76 \pm 0.02
AqSolDB [37]	logS	0.88 \pm 0.02	0.35 \pm 0.05	0.24 \pm 0.02	0.26 \pm 0.17	0.45 \pm 0.03
Lipop [49]	logD	0.74 \pm 0.03	0.15 \pm 0.06	0.10 \pm 0.02	0.22 \pm 0.12	0.32 \pm 0.02
COMPAS [45]	rel. Ener.	0.90 \pm 0.05	0.65 \pm 0.04	0.37 \pm 0.03	0.45 \pm 0.11	0.66 \pm 0.03
	GAP	0.97 \pm 0.01	0.44 \pm 0.05	0.27 \pm 0.05	0.59 \pm 0.20	0.71 \pm 0.01
QM9 [31]	Dip. Mom.	0.78 \pm 0.00	0.57 \pm 0.01	0.45 \pm 0.01	0.76 \pm 0.03	0.53 \pm 0.00
	HOMO	0.93 \pm 0.00	0.54 \pm 0.02	0.23 \pm 0.01	0.61 \pm 0.10	0.63 \pm 0.01
	LUMO	0.99 \pm 0.00	0.48 \pm 0.02	0.23 \pm 0.01	0.67 \pm 0.02	0.73 \pm 0.00
	GAP	0.97 \pm 0.00	0.52 \pm 0.02	0.25 \pm 0.01	0.76 \pm 0.04	0.68 \pm 0.00

[0.10, 0.45]). Notably, the highest error reduction is found for the prediction of the Dipole Moment in the QM9 dataset with a mean error reduction of $\text{UER-AUC}_{\text{mean}} = 0.45$ and a max error reduction of $\text{UER-AUC}_{\text{max}} = 0.78$. In contrast, the lowest error reduction can be observed for the prediction of the Lipophilicity with a mean error reduction of only $\text{UER-AUC}_{\text{mean}} = 0.10$.

The extent of error reduction potential does not appear to correlate strongly with the predictive performance of the model, as both the highest and lowest error reductions were associated with models demonstrating lower predictivity ($R^2 \approx 0.7$). In addition, even models with high predictivity, such as the prediction of the LUMO value ($R^2 = 0.99$), show moderate amounts of error reduction potential ($\text{UER-AUC}_{\text{mean}} = 0.23$, $\text{UER-AUC}_{\text{max}} = 0.67$). We hypothesize that the error reduction may be connected to the complexity of the underlying data distribution and the presence of labeling noise. The Lipophilicity dataset, for example, consists of inherently noisy experimental measurements while values for the dipole moment in the QM9 dataset were obtained by more precise DFT simulations.

Overall, the results of this experiment indicate that uncertainty threshold-based filtering can be used as an effective tool to decrease the overall prediction error even for complex properties, which may have been obtained through noisy measurements.

4.3 Experiments on Counterfactual Truthfulness

In the second part of the computational experiments, we investigate the potential of uncertainty-based filtering to improve the overall truthfulness of counterfactual explanations.

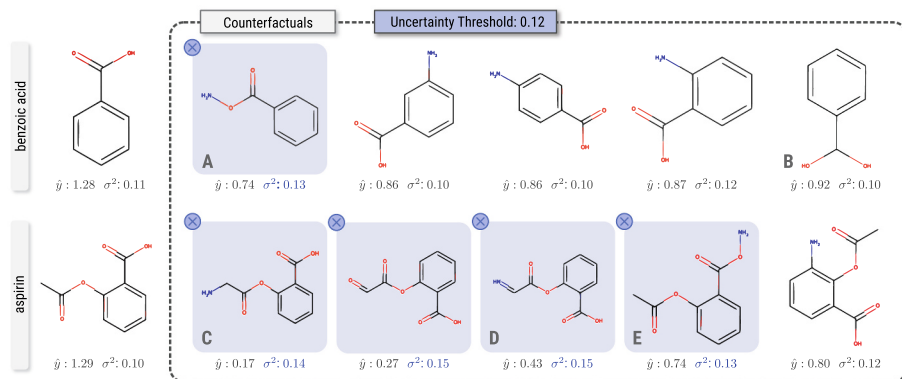


Fig. 4. Qualitative results of uncertainty-based counterfactual filtering for two example molecules. Predictions are made by a GATv2 graph neural network and uncertainties are estimated by a combination of ensembling and mean-variance estimation. The uncertainty threshold ξ_{20} was chosen on the test set such that the 20% lowest uncertainty elements remain.

Improving Counterfactual Truthfulness. For this experiment, we use the CLogP dataset, as the underlying property is deterministically calculable for any valid molecular graph. This availability of a ground truth oracle is necessary for the computation of the relative truthfulness as defined in Sect. 3.3. As before, we use the GATv2 model architecture and investigate the effectiveness of ensembles, mean-variance estimation, and the combination thereof. We split the dataset into a test set (10%), a calibration set (20%), and a train set (70%). All models are fitted with the train set, and uncertainty estimates are subsequently calibrated on the validation set. On the test set, we determine a single uncertainty threshold ξ_{20} such that exactly the 20% elements with the lowest predicted uncertainties remain.

As described in Sect. 3.2, we generate counterfactual samples by ranking all graphs in a 1-edit neighborhood according to the prediction divergence and choosing the top 10 candidates. This set of counterfactuals is then filtered using the threshold ξ_{20} and examined regarding its relative truthfulness.

The results of this experiment are reported in Table 4. A “Random” baseline was included as a control. As expected, this control’s randomly generated uncertainty values result neither in test set error reduction nor an increase of counterfactual truthfulness. All other uncertainty quantification (UQ) methods demonstrated a moderate potential for error reduction on both the test set and the set of counterfactuals ($\text{UER-AUC} \geq 0.2$). Furthermore, all UQ methods exhibited some capability to increase relative truthfulness when filtering with the uncertainty threshold ξ_{20} . However, it has to be noted that the initial truthfulness in the unfiltered set of counterfactuals is rather high (up to 95%), leaving little room for further improvement. It is important to note, however, that the initial truthfulness in the unfiltered set of counterfactuals was relatively high (up

Table 4. Results of 5 independent repetitions of computational experiments on the ClogP dataset to evaluate counterfactual truthfulness using a fixed uncertainty threshold ξ_{20} determined on the test set. Results are obtained using a GATv2 graph neural network, and uncertainties are estimated by the combination of Deep Ensembles and MVE. [†]*Tr. Init.* represents the initial percentage of truthful counterfactuals in the unfiltered set of all counterfactuals. [‡]*Tr. Gain.* shows the increase in the relative percentage of truthful counterfactuals after filtering according to the uncertainty threshold ξ_{20} .

Method	Test Set		Counterfactuals			
	$R^2 \uparrow$	UER-AUC \uparrow mean	$\rho \uparrow$	UER-AUC \uparrow mean	Tr. Init. [†] \uparrow (%)	Tr. Gain [‡] \uparrow (%)
Random	1.00 \pm 0.01	-0.00 \pm 0.06	-0.01 \pm 0.03	-0.02 \pm 0.03	0.95 \pm 0.03	-0.00 \pm 0.05
MVE	0.98 \pm 0.02	0.26 \pm 0.05	0.53 \pm 0.15	0.23 \pm 0.05	0.77 \pm 0.17	0.09 \pm 0.05
DE	1.00 \pm 0.00	0.27 \pm 0.04	0.45 \pm 0.12	0.20 \pm 0.05	0.94 \pm 0.03	0.04 \pm 0.05
DE+MVE	1.00 \pm 0.00	0.28 \pm 0.03	0.44 \pm 0.16	0.23 \pm 0.05	0.95 \pm 0.02	0.05 \pm 0.02

to 95%), limiting the scope for further improvement. Notably, the mean variance estimation model displayed a substantially lower initial truthfulness (0.77), most likely due to its slightly lower predictivity.

In addition to the results for the fixed uncertainty threshold ξ_{20} , Fig. 5 visualizes the progression of mean error reduction and truthfulness results across a range of possible uncertainty thresholds. The plots show that for increasingly strict uncertainty thresholds, the relative counterfactual truthfulness also increases near-monotonically, reaching 100% with a small subset of 5% remaining counterfactuals.

In summary, we find that all UQ methods exhibit some capacity to improve the relative counterfactual truthfulness through uncertainty-based filtering. However, the results may be influenced by the already elevated values observed for the unfiltered set. Future work should explore more complex property prediction tasks with lower predictive performance and, consequently, lower starting points of counterfactual truthfulness.

Qualitative Filtering Results. Besides a quantitative evaluation of counterfactual truthfulness, some qualitative results of uncertainty-based filtering are illustrated in Fig. 4 for two example molecules. Uncertainty estimates are obtained by an ensemble of mean variance estimators. As before, the uncertainty threshold ξ_{20} is chosen such that only 20% of the test set elements with the lowest uncertainty values remain.

For the first molecule, benzoic acid, only the highest-ranked counterfactual candidate **A** is filtered due to exceeding the uncertainty threshold. This exclusion intuitively makes sense since the added bond between oxygen and nitrogen is an uncommon configuration that is not represented in the underlying dataset and can be considered an out-of-distribution input. In contrast to this expected behavior, the counterfactual candidate **B** is not filtered but represents an equally

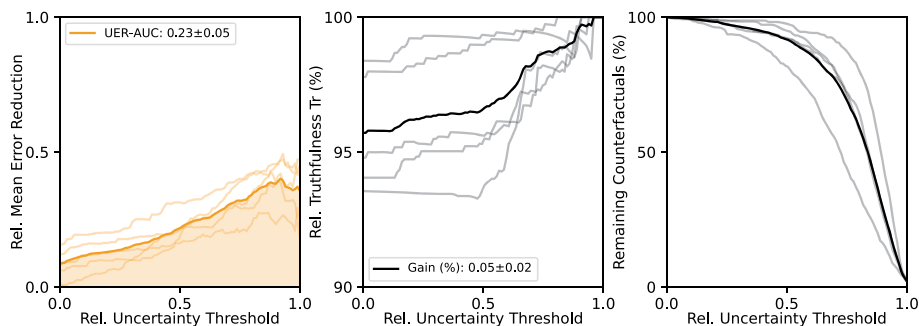


Fig. 5. Results for 5 independent repetitions of computational experiment on the ClogP dataset to evaluate counterfactual truthfulness. Individual results are plotted transparently in the background, and average curves are indicated with bold lines. All plots are based on the set of counterfactuals and show from left to right: The relative mean error reduction, the Truthfulness, and the percentage of remaining counterfactuals for different uncertainty thresholds. Results are obtained by a GATv2 graph neural network, and uncertainties are estimated by an ensemble of mean variance estimators.

uncommon configuration with one carbon being connected to two single-bonded oxygen at the same time. Notably, the model also predicts a significantly incorrect value for this counterfactual candidate.

For the second molecule, aspirin, the four highest-ranked counterfactuals are filtered based on their predicted uncertainty. The exclusion of the counterfactual candidate **E** also intuitively makes sense since it includes the same uncommon bond between nitrogen and oxygen. Excluded counterfactual candidate **D** also contains a rather uncommon substructure but, more importantly, is predicted highly inaccurately by the model. In contrast to these cases, the highest ranked counterfactual candidate **C** is excluded even though the model’s prediction is highly accurate, serving as an example of overly conservative filtering.

Overall, the qualitative examples illustrate that the uncertainty-based filtering can be effective in identifying and removing out-of-distribution input samples and generally inaccurate predictions. However, there are also cases in which the method fails by either failing to filter OOD samples or by being too conservative and filtering perfectly accurate predictions.

5 Discussion

Previous work has investigated the intersection of uncertainty estimation and counterfactual explainability predominantly in the context of image classification [2, 8, 33]. Schut *et al.* [33], for example, include an ensemble-based uncertainty estimate as a direct objective in the optimization of counterfactual explanations. The authors find this intervention to reduce the likelihood of generating uninformative out-of-distribution samples—or in the words of Freiesleben [10] to

steer the generation toward true counterfactual explanations rather than mere adversarial examples.

In our work, we present a distinct perspective to the existing literature, which differs in two important aspects: We focus on (1) regression tasks in the (2) graph processing domain. In image processing, good counterfactual explanations require the modification of multiple pixel values in a semantically meaningful way. This is framed as a non-trivial optimization objective requiring substantial computational effort. In the graph processing domain, however, the limited number of possible graph modifications makes it computationally feasible to search for counterfactual candidates among a full enumeration of all possible perturbations. Consequently, uncertainty quantification does not have to be included in the generation process itself but instead may serve as a simple filter over this set of possible perturbations. Nevertheless, the objective is the same: to use uncertainty quantification methods to present higher-quality counterfactual explanations to the user.

Another key factor is the difference between classification and regression tasks. While classification enables binary assessments of correct and incorrect predictions, regression operates on a continuous error scale, requiring different metrics to assess the impact of uncertainty estimation—motivating our definitions of the UER-AUC and the counterfactual truthfulness.

Consistent with existing literature, our work demonstrates that incorporating uncertainty estimation improves the quality of counterfactual explanations. We specifically find that filtering high-uncertainty elements decreases the average error of the remaining set and increases overall *truthfulness*—meaning the explanation’s alignment with the underlying ground truth data distribution.

In our experiments, we find no substantial differences in the relative effectiveness of UQ interventions between three common graph neural network architectures (GCN, GATv2, GIN). Regarding the choice of the uncertainty estimation method, we find trust scores [8, 33] to be ill-suited to graph processing applications, most likely due to the unavailability of suitable distance metrics. We furthermore come to similar conclusions as previous authors [6, 13, 32] in that the simple application of model ensembles already proves relatively effective. While we find a combination between ensembles and mean variance estimation to be slightly beneficial on IID data, there seems to be no substantial difference in OOD test scenarios.

However, in this context, it is still important to mention the remaining limitations of this approach grounded in the non-perfect correlation of the prediction errors and estimated uncertainties. While quantitative results show that uncertainty-based filtering has a higher relative likelihood to remove truly high-error samples, qualitative results indicate it still occasionally fails to detect some high-error samples and mistakenly filters valid elements. Depending on the severity of these failure cases, it will largely depend on the concrete application whether the increased truthfulness reasonably justifies the loss of some valid explanations.

6 Conclusion

Counterfactual explanations can deepen the understanding of a complex model’s predictive behavior by illustrating which kinds of local perturbations a model is especially sensitive to. In the scientific domain of chemistry and material science, these explanations are often not only desirable to understand the model’s behavior but by extension to understand the structure-property relationships of the underlying data itself. To use counterfactuals to gain insights about the underlying data, the explanations *truthfully* must reflect the properties thereof.

In this work, we explore the potential of uncertainty estimation to increase the overall truthfulness of a set of counterfactuals by filtering those elements with particularly high predicted uncertainty. We conduct extensive computational experiments with different methods to investigate the error-reduction potential of various uncertainty estimation methods in different settings. We find that model ensembles provide strong uncertainty estimates in out-of-distribution test scenarios, while a combination of ensembles and mean variance provide the highest error reduction potential for in-distribution settings.

Based on these initial results, we conclude that uncertainty estimation presents a promising opportunity to increase the truthfulness of explanations—to make sure explanations not only represent the model’s behavior but the properties of the underlying data as well. An interesting direction for future research will be to see if uncertainty estimation can be employed equally beneficially to different explanation modalities, such as local attributional and global concept-based explanations.

Acknowledgments. This work was supported by funding from the pilot program Core-Informatics of the Helmholtz Association (HGF). This research was supported by the German Research Foundation (DFG) under Germany’s Excellence Strategy via the Excellence Cluster “3D Matter Made to Order” (3DMM2O, EXC-2082/1–390761711). We acknowledge funding by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) within Priority Programme SPP 2363.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Abdar, M., et al.: A review of uncertainty quantification in deep learning: techniques, applications and challenges. *Inf. Fusion* **76**, 243–297 (2021)
2. Antor’an, J., Bhatt, U., Adel, T., Weller, A., Hernández-Lobato, J.M.: Getting a CLUE: a method for explaining uncertainty estimates. In: *International Conference on Learning Representations* (2020)
3. Bajaj, M., et al.: Robust counterfactual explanations on graph neural networks. *arXiv preprint* (2021)
4. Bishop, C.M.: Bayesian neural networks. *J. Braz. Comput. Soc.* **4**, 61–68 (1997)
5. Brody, S., Alon, U., Yahav, E.: How attentive are graph attention networks? In: *International Conference on Learning Representations* (2022)

6. Busk, J., Bjørn Jørgensen, P., Bhowmik, A., Schmidt, M.N., Winther, O., Vegge, T.: Calibrated uncertainty for molecular property prediction using ensembles of message passing neural networks. *Mach. Learn. Sci. Technol.* **3**(1), 015012 (2021)
7. D' Angelo, F., Fortuin, V.: Repulsive deep ensembles are Bayesian. In: *Advances in Neural Information Processing Systems*, vol. 34, pp. 3451–3465. Curran Associates, Inc. (2021)
8. Delaney, E., Greene, D., Keane, M.T.: Uncertainty estimation and out-of-distribution detection for counterfactual explanations: pitfalls and solutions (2021)
9. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning. *arXiv preprint [arXiv:1702.08608](https://arxiv.org/abs/1702.08608)* (2017)
10. Freiesleben, T.: The intriguing relation between counterfactual explanations and adversarial examples. *Mind. Mach.* **32**(1), 77–109 (2022)
11. Gawlikowski, J., et al.: A survey of uncertainty in deep neural networks. *Artif. Intell. Rev.* **56**(1), 1513–1589 (2023)
12. Goan, E., Fookes, C.: Bayesian neural networks: an introduction and survey. In: Mengersen, K.L., Pudlo, P., Robert, C.P. (eds.) *Case Studies in Applied Bayesian Data Science: CIRM Jean-Morlet Chair, Fall 2018*, pp. 45–87. Springer, Cham (2020)
13. Hirschfeld, L., Swanson, K., Yang, K., Barzilay, R., Coley, C.W.: Uncertainty quantification using neural networks for molecular property prediction. *J. Chem. Inf. Model.* **60**(8), 3770–3780 (2020)
14. Kellner, M., Ceriotti, M.: Uncertainty quantification by direct propagation of shallow ensembles. *Mach. Learn. Sci. Technol.* **5**(3), 035006 (2024)
15. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: *International Conference on Learning Representations* (2017)
16. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In: *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc. (2017)
17. Landrum, G.: *RDKit: Open-source cheminformatics* (2010)
18. Ley, D., Bhatt, R., Weller, A.: δ -CLUE: diverse sets of explanations for uncertainty estimates. *arXiv preprint* (2021)
19. Ley, D., Bhatt, U., Weller, A.: Diverse, global and amortised counterfactual explanations for uncertainty estimates. *Proc. AAAI Conf. Artif. Intell.* **36**(7), 7390–7398 (2022)
20. Liu, C., Zhuo, J., Cheng, P., Zhang, R., Zhu, J.: Understanding and accelerating particle-based variational inference. In: *International Conference on Machine Learning* (2018)
21. Liu, Q., Wang, D.: Stein variational gradient descent: a general purpose Bayesian inference algorithm. In: *Advances in Neural Information Processing Systems*, vol. 29. Curran Associates, Inc. (2016)
22. Longo, L., et al.: Explainable artificial intelligence (XAI) 2.0: a manifesto of open challenges and interdisciplinary research directions. *Inf. Fusion* **106**, 102301 (2024)
23. Lucic, A., ter Hoeve, M., Tolomei, G., de Rijke, M., Silvestri, F.: CF-GNNExplainer: counterfactual explanations for graph neural networks. In: *International Conference on Artificial Intelligence and Statistics* (2021)
24. Maffettone, P.M., et al.: What is missing in autonomous discovery: open challenges for the community. *Digital Discov.* **2**(6), 1644–1659 (2023)
25. Maddox, W.J., Izmailov, P., Garipov, T., Vetrov, D.P., Wilson, A.G.: A simple baseline for Bayesian uncertainty in deep learning. In: *Advances in Neural Information Processing Systems*, vol. 32. Curran Associates, Inc. (2019)

26. Miller, T.: Explanation in artificial intelligence: insights from the social sciences. *Artif. Intell.* **267**, 1–38 (2019)
27. Nguyen, T.M., Quinn, T.P., Nguyen, T., Tran, T.: Explaining black box drug target prediction through model agnostic counterfactual samples. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **20**(2), 1020–1029 (2023)
28. Nix, D.A., Weigend, A.S.: Estimating the mean and variance of the target probability distribution. In: *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, vol. 1, pp. 55–60 (1994)
29. Numeroso, D., Bacciu, D.: MEG: generating molecular counterfactual explanations for deep graph networks. In: *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8 (2021)
30. Prado-Romero, M.A., Prenkaj, B., Stilo, G., Giannotti, F.: A survey on graph counterfactual explanations: definitions, methods, evaluation, and research challenges. *ACM Comput. Surv.* **56**(7), 171:1–171:37 (2024)
31. Ramakrishnan, R., Dral, P.O., Rupp, M., Lilienfeld, O.A.: Quantum chemistry structures and properties of 134 kilo molecules. *Sci. Data* **1**(1), 140022 (2014)
32. Scalia, G., Grambow, C.A., Pernici, B., Li, Y.-P., Green, W.H.: Evaluating scalable uncertainty estimation methods for deep learning-based molecular property prediction. *J. Chem. Inf. Model.* **60**(6), 2697–2717 (2020)
33. Schut, L., et al.: Generating interpretable counterfactual explanations by implicit minimisation of epistemic and aleatoric uncertainties. In: *International Conference on Artificial Intelligence and Statistics* (2021)
34. Seitzer, M., Tavakoli, A., Antic, D., Martius, G.: On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks. In: *International Conference on Learning Representations* (2022)
35. Seuss, D.: Bridging the gap between explainable AI and uncertainty quantification to enhance trustability. *arXiv preprint* (2021)
36. Sluijterman, L., Cator, E., Heskes, T.: Optimal training of mean variance estimation neural networks. *Neurocomputing* **597**, 127929 (2024)
37. Sorkun, M.C., Khetan, A., Er, S.: AqSolDB, a curated reference set of aqueous solubility and 2D descriptors for a diverse set of compounds. *Sci. Data* **6**(1), 143 (2019)
38. Sturm, H., Teufel, J., Isfeld, K.A., Friederich, P., Davis, R.L.: Mitigating molecular aggregation in drug discovery with predictive insights from explainable AI (2023)
39. Tan, J., et al.: Learning and evaluating graph neural network explanations based on counterfactual and factual reasoning. In: *Proceedings of the ACM Web Conference*, pp. 1018–1027 (2022)
40. Teufel, J., Friederich, P.: Global concept explanations for graphs by contrastive learning. In: Longo, L., Lapuschkin, S., Seifert, C. (eds.) *Explainable Artificial Intelligence*, pp. 184–208. Springer, Cham (2024)
41. Teufel, J., Torresi, L., Reiser, P., Friederich, P.: MEGAN: multi-explanation graph attention network. In: Longo, L., Lapuschkin, S., Seifert, C. (eds.) *Explainable Artificial Intelligence*, pp. 338–360. Springer, Cham (2023)
42. Tishby, Levin, Solla: Consistent inference of probabilities in layered networks: predictions and generalizations. In: *International 1989 Joint Conference on Neural Networks*, vol. 2, pp. 403–409 (1989)
43. Trinh, T., Heinonen, M., Acerbi, L., Kaski, S.: Input-gradient space particle inference for neural network ensembles. In: *The Twelfth International Conference on Learning Representations* (2023)
44. Verma, S., et al.: Counterfactual explanations and algorithmic recourses for machine learning: a review. *ACM Comput. Surv.* **56**(12), 312:1–312:42 (2024)

45. Wahab, A., Pfuderer, L., Paenurk, E., Gershoni-Poranne, R.: The COMPAS project: a computational database of polycyclic aromatic systems. Phase 1: cata-condensed polybenzenoid hydrocarbons. *J. Chem. Inf. Model.* **62**(16), 3704–3713 (2022)
46. Wellawatte, G.P., Seshadri, A., White, A.D.: Model agnostic generation of counterfactual explanations for molecules. *Chem. Sci.* **13**(13), 3697–3705 (2022)
47. Wildman, S.A., Crippen, G.M.: Prediction of physicochemical parameters by atomic contributions. *J. Chem. Inf. Comput. Sci.* **39**(5), 868–873 (1999)
48. Wills, P., Meyer, F.G.: Metrics for graph comparison: a practitioner’s guide. *PLoS ONE* **15**(2), e0228728 (2020)
49. Zhenqin, W., et al.: MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci.* **9**(2), 513–530 (2018)
50. Xu, K., Hu, W., Leskovec, J., Jegelka, S.: How powerful are graph neural networks? In: International Conference on Learning Representations (2018)
51. Zednik, C., Boelsen, H.: Scientific exploration and explainable artificial intelligence. *Mind. Mach.* **32**(1), 219–239 (2022)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

