NOISE-RESPONSE CALIBRATION: A CAUSAL INTERVENTION PROTOCOL FOR LLM-JUDGES

Maxim Khomiakov[1,2,3] **& Jes Frellsen**[1,3]
[1]Technical University of Denmark, Kgs. Lyngby, Denmark
[2]Normal Computing Corporation, New York, NY, USA
[3]Pioneer Centre for Artificial Intelligence, Copenhagen, Denmark
maxim@normalcomputing.com, {maxk,jefr}@dtu.dk

ABSTRACT

Large language models (LLMs) are increasingly used as automated judges and synthetic labelers, especially in low-label settings. Yet these systems are stochastic and often overconfident, which makes deployment decisions difficult when external ground truth is limited. We propose a practical calibration protocol based on controlled input interventions: if noise severity increases, task performance should exhibit a statistically significant deterioration trend. We operationalize this with a slope-based hypothesis test over repeated trials, using signal-to-noise-ratio (SNR) perturbations for tabular data and lexical perturbations for text data. Across UCI tabular benchmarks and four text classification datasets, we find clear modality-dependent behavior. Our results reveal a modality gap: while text-based judges degrade predictably, the majority of tabular datasets show a lack of statistically significant performance deterioration even under significant signal-to-noise reduction. Interestingly we find that model performance is lower on datasets that are insensitive to noise interventions. We present a reproducible methodology and reporting protocol for robust LLM-judge calibration under distribution shift.

1 INTRODUCTION

High-quality labels remain costly and difficult to produce: annotation policy design, expert recruitment, and quality control are all substantial bottlenecks (Gebru et al., 2021; Snow et al., 2008). In parallel, LLM-based pipelines now support two high-value workflows: (1) synthetic labeling, and (2) LLM-as-a-Judge decision systems for automated triage and ranking (Liu et al., 2023; Zheng et al., 2023).

The practical failure mode is not only low average quality, but unreliable behavior under shift: small input changes can create non-trivial output volatility, especially when large trusted holdout labels are unavailable (Lin et al., 2022; Huang et al., 2025).

Core question. Can we reject an LLM judge for a target task *before* deployment by stress-testing its response to controlled interventions? We frame this as a causal question in the interventionist sense of Pearl (2009): apply a controlled perturbation to the input and test whether performance degrades as predicted.

We formalize a monotone-deterioration hypothesis over a calibrated noise schedule. Let $\mathcal{P}$ denote a task-specific performance score (e.g., accuracy for classification, R^2 for regression) measured on clean inputs, and $\mathcal{P}_k$ the same score measured under perturbation magnitude $k \in K$, where K is ordered from weak to strong noise. Define $\Delta_k = \mathcal{P}_k - \mathcal{P}$. Our conceptual intervention hypothesis is

$$H_0 : \exists\, k_1 < k_2 \text{ such that } \mathbb{E}[\mathcal{P}_{k_2}] > \mathbb{E}[\mathcal{P}_{k_1}], \tag{1}$$

$$H_1 : \mathbb{E}[\mathcal{P}_{k_1}] \geq \mathbb{E}[\mathcal{P}_{k_2}] \quad \forall k_1 < k_2, \tag{2}$$

with at least one strict inequality under H_1.

Here, H_0 (the null hypothesis) states that the expected performance is *not* monotone under increasing perturbation i.e., there exists at least one pair of noise magnitudes where performance improves

despite stronger noise. H_1 (the alternative) encodes the expected deterioration property: as noise increases, the expected performance does not increase, and it decreases for at least one step. The expectation $\mathbb{E}[\mathcal{P}_k]$ is taken over the randomness in the procedure at severity level k, including (i) the sampled perturbations/noise realizations and (ii) any stochasticity in the LLM inference; In practice, we operationalize this ordered alternative using a one-sided linear trend test (negative slope) as a pragmatic surrogate; thus sensitive indicates significant deterioration with increasing noise, which is consistent with but not equivalent to strict monotonicity.

Intuitively, if increasing perturbation does *not* induce expected deterioration, the model is either exploiting artifacts or is insufficiently sensitive to task signal, both of which reduce trust in judge-style deployment. Conversely, when the expected deterioration holds with statistical significance, we obtain a useful calibration signal about where the model can be trusted.

This paper presents a reproducible methodology around this idea: a pipeline for perturbation design, repeated inference, and hypothesis testing across tabular and text settings, presenting findings which indicate that employing such a methodology prior to deployment could provide an indicator of trustworthy LLM results.

Overview of Contributions

1. We propose a practical *noise-response calibration protocol* for deciding whether an LLM judge is reliable for a specific task, based on intervention schedules and slope-based hypothesis testing.
2. We report large-scale empirical findings spanning UCI tabular tasks and four widely used text classification datasets, performing experiments applying both correlated and uncorrelated Gaussian noise for tabular data and lexical noise for text.
3. We demonstrate that for the majority of included UCI tabular tasks, performance is largely *insensitive* to the tested noise schedules (i.e., it does not exhibit statistically significant deterioration as noise increases). We discuss how this insensitivity can serve as a practical warning signal for LLM-judge deployment, complementing average-performance reporting.

2 Related works

Hallucination, alignment, and judge reliability. Recent work has studied hallucination mechanisms and mitigation strategies in depth, including retrieval augmentation, verification, and calibration-oriented prompting (Huang et al., 2025). In parallel, alignment methods such as RLHF and constitutional training have improved instruction-following behavior, but do not eliminate robustness failures under distribution shift (Ouyang et al., 2022; Bai et al., 2022).

For LLM-as-a-Judge pipelines, reporting and evaluation protocol quality is itself a major source of variance. A central recommendation in recent methodology work is that judge results should be reported with stronger controls for prompt sensitivity, variance, and statistical uncertainty (Lee et al., 2026).

Perturbation sensitivity in reasoning and NLP. Our work is closely related to perturbation-based robustness analysis. In mathematical reasoning, Mirzadeh et al. (2025) show that seemingly benign symbolic variations can induce substantial performance changes. More broadly, LLM reasoning has been shown to degrade under distractor-like or superficially irrelevant perturbations (Shi et al., 2023).

For text corruption, lexical perturbation studies demonstrate that character-level noise can strongly hurt model behavior even when semantics are mostly preserved (Belinkov & Bisk, 2018; Pruthi et al., 2019). Our text intervention family is inspired by this line of work but is used for *trust calibration* rather than adversarial training.

Causal framing and benchmark context. We adopt an intervention-based causal framing: the perturbation operator is treated as an intervention on inputs, and the downstream performance response is the measured effect (Pearl, 2009; Peters et al., 2017). This perspective makes explicit what is being manipulated and what constitutes expected behavior under increasing uncertainty.

Algorithm 1 Noise-response calibration protocol

Require: Dataset D, noise schedule $N = \{n_k\}_{k=1}^{K}$, repetitions R, significance level α
Initialize LLM Judge for the prediction task and compute baseline performance $\mathcal{P}$
for each $n_k \in N$ **do**
  for each repetition $r \in \{1, \ldots, R\}$ **do**
    Generate noisy inputs, query the LLM, record performance $\mathcal{P}_{k,r}$
  end for
end for
Fit linear model $\mathcal{P}_{k,r} = \beta_0 + \beta_1 n_k + \epsilon$
Perform one-sided slope test $H_0 : \beta_1 \geq 0$ vs. $H_1 : \beta_1 < 0$
Accept task-level trust calibration only if deterioration is significant and directionally consistent

For tabular benchmarks, we use datasets from the UCI repository (Asuncion & Newman, 2007), one of the largest tabular datasets designed for ML prediction tasks, and include canonical datasets such as Iris (Fisher, 1936) and Wine Quality (Cortez et al., 2009). For text experiments we use IMDB (Maas et al., 2011), Yelp Review Full (Zhang et al., 2015), SST-2 (Socher et al., 2013), and Financial PhraseBank (Malo et al., 2014), datasets that all revolve around sentiment classification tasks.

3 METHOD

Our procedure is conceptually close to mutation testing in software engineering where one applies controlled perturbations, observes failure behavior, and decides whether a system should be trusted for deployment-critical use (DeMillo et al., 1978; Jia & Harman, 2011). Designing a sufficiently general noise family is non-trivial and domain-dependent. The intervention family can miss realistic corruption patterns, so acceptance should be interpreted as conditional on the tested noise family, not as a universal robustness claim. We treat this explicitly as a limitation and future direction (Section 6).

For each dataset, we build a dynamic system prompt containing task explanation, dataset metadata, output constraints, and provide few shot examples as part of the input. Predictions are validated through a schema-constrained output parser, where one requires the answer to match the format of the label. We present our noise-response calibration protocol in Algorithm 1.

3.1 TABULAR NOISE VIA SNR SCHEDULES

For tabular inputs, let $X_{\text{clean}} \in \mathbb{R}^{N \times d}$ denote the full clean training matrix restricted to numeric covariates, and let $x \in \mathbb{R}^d$ denote one evaluation row. We estimate the reference covariance once on the full clean training data, $\Sigma_{\text{signal}} = \text{Cov}(X_{\text{clean}})$, and write $\sigma_j^2 = \Sigma_{\text{signal},jj}$ for the empirical variance of feature j. Let $D = \text{diag}(\sigma_1^2, \ldots, \sigma_d^2)$ and $R = \text{Corr}(X_{\text{clean}})$, so that $\Sigma_{\text{signal}} = DRD$. For a target SNR_{dB}, define $\alpha = 10^{-\text{SNR}_{\text{dB}}/10}$. This scales the noise power relative to the fixed clean-data signal statistics, and we evaluate a fixed ordered schedule from mild to strong perturbation.

Our interventions operate only on numeric covariates; datasets with low numeric-feature coverage receive partial or no perturbation, and non-numeric covariates are preserved. We add Gaussian noise $x_{\text{noisy}} = x + \epsilon$ under two covariance structures:

1. **Uncorrelated**: $\epsilon \sim \mathcal{N}(0,\, \alpha D^2)$, equivalently $\epsilon \sim \mathcal{N}(0,\, \alpha\, \text{diag}(\sigma_1^2, \ldots, \sigma_d^2))$.
2. **Correlated**: $\epsilon \sim \mathcal{N}(0,\, \alpha \Sigma_{\text{signal}})$, equivalently $\epsilon \sim \mathcal{N}(0,\, \alpha DRD)$.

Both perturbations are zero-mean and share the same per-feature marginal variances $\alpha\sigma_j^2$; they differ only in whether cross-feature covariance is suppressed (uncorrelated) or preserved according to the empirical correlation structure of the full clean training data (correlated). This lets us test whether sensitivity depends only on marginal noise scale or also on the directional structure of the perturbation.

3.2 TEXT DATASETS

We evaluate IMDB, Yelp Review Full, SST-2, and Financial PhraseBank (Maas et al., 2011; Zhang et al., 2015; Socher et al., 2013; Wang et al., 2019; Malo et al., 2014). All are classification tasks, but they differ in domain, label granularity, and text length.

Noise intervention design. For text, we focus on lexical corruption inspired by prior robust and adversarial NLP work (Belinkov & Bisk, 2018; Pruthi et al., 2019). While prior work primarily uses such perturbations to *train* or *harden* models against corrupted inputs, we leverage the same mechanisms to evaluate whether an LLM judge exhibits the expected degradation as corruption increases.

Lexical noise. Let $x = (w_1, \ldots, w_T)$ be a tokenized input and $\alpha \in [0, 1]$ a severity control. We define

$$x_{\text{noisy}} \sim \mathcal{C}_\alpha(x), \tag{3}$$

where $\mathcal{C}_\alpha$ applies token-level corruptions with probability

$$p(\alpha) = \alpha p_{\max}. \tag{4}$$

For each token w_i, we sample whether to corrupt, then sample an operation: word dropout, adjacent character swap, keyboard-adjacent typo, or random insertion/deletion relative to the corruption severity level α.

3.3 DETERIORATION ANALYSIS

For each setting (dataset $\times$ noise type/intensity), we fit an ordinary least-squares regression

$$\mathcal{P}_{k,r} = \beta_0 + \beta_1 n_k + \epsilon_{k,r}, \tag{5}$$

where $k \in \{1, \ldots, K\}$ indexes the noise severity level, $r \in \{1, \ldots, R\}$ indexes the independent repetition at that level, $\mathcal{P}_{k,r}$ is the observed performance score for severity k and repetition r, β_0 is the intercept (expected performance at zero added noise), β_1 is the slope measuring the rate of performance change per unit of noise intensity, n_k is linear in true noise intensity (for tabular data, $n_k = \text{SNR}_{\max} - \text{SNR}_k$), and $\epsilon_{k,r} \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_\epsilon^2)$ is the residual error term. We then test

$$H_0 : \beta_1 \geq 0, \tag{6}$$

$$H_1 : \beta_1 < 0, \tag{7}$$

at significance level $\alpha = 0.05$, using the standard OLS t-statistic for the slope coefficient (Seber & Lee, 2012):

$$t = \frac{\hat{\beta}_1}{\text{SE}(\hat{\beta}_1)}, \qquad t \sim \text{Student-}t_{N-2} \text{ under } H_0, \tag{8}$$

where $\hat{\beta}_1$ is the OLS slope estimate, $\text{SE}(\hat{\beta}_1)$ is its standard error, and $N = K \times R$ is the total number of observations. We reject H_0 (and label the task *sensitive*) when $t < t_{\alpha,N-2}$, where $t_{\alpha,N-2}$ is the critical value corresponding to significance level α for a Student's t-distribution with $N - 2$ degrees of freedom. That is when the observed slope is significantly negative at the chosen level. This operational test approximates the ordered alternative in Equation (1)–Equation (2) via a one-sided negative-trend surrogate. Thus, for all the datasets we compute a one-directional test of negative slope, to inform us as to whether the dataset in question proved *sensitive* or *insensitive* to the noise intervention.

4 EXPERIMENTAL SETUP

We run all main experiments with OpenAI GPT-5-MINI as the base judge model. For each dataset and noise level, we run 5 repetitions and provide 20 few-shot examples before recording the predicted value. Our goal is to give the model the best possible basis for solving the task at hand before deliberately handicapping its odds. For the prompt serialization we rely on a fairly simple approach, providing a dataset and task description, information about the label and output format constraints. Further details on the prompt leveraged is described in Appendix Section B.2.

Table 1: Stepwise UCI exclusion flow. First filter keeps datasets with $> 50\%$ numerical covariates; second filter applies missing-value incomputability rules; the final row reports the subset used for per-dataset slope analyses in Appendix Table 5.

Step / criterion	Classification	Regression
Benchmark inventory declared in this paper	222	143
After $> 50\%$ numeric feature filter	190 (-32)	127 (-16)
After missing-value drop-off	185 (-5)	125 (-2)
Complete datasets included in this study	138	21

4.1 Datasets and tasks

Our benchmark covers three domains:

1. UCI classification: 222 datasets.
2. UCI regression: 143 datasets.
3. Text classification: 4 datasets (IMDB, Yelp Review Full, SST-2, Financial PhraseBank).

All our datasets are split into train/valid/test fractions $0.70/0.15/0.15$, with stratified training split where applicable. Few shot examples are taken from the training set.

4.2 UCI eligibility and filtering

As previously mentioned, we rely on additive Gaussian noise and therefore apply it only to numeric features. Not all datasets, however, have sufficiently many numerical features available. We therefore use explicit exclusion criteria: (i) a minimum proportion of numeric features, and (ii) removal of datapoints with missing values. If these filters leave too few observations, we skip the dataset. We summarize the resulting dataset eligibility flow in Table 1.

This filtering reflects a common practical limitation in tabular-LLM evaluation pipelines, where (i) numeric-only perturbations do not apply to categorical-heavy datasets and (ii) prompt-length constraints limit how much structured information can be provided to the model (Fang et al., 2024; Hegselmann et al., 2023; Sui et al., 2024).

The final count of 138 classification and 21 regression datasets reflects those for which we obtained complete experimental results across all noise levels and repetitions. Extending coverage to the remaining eligible datasets is discussed in Section 6.

Performance metrics and configuration For the performance metrics we compute confusion metrics such as F1, precision and recall but rely on accuracy as the primary measure for the classification studies. Similarly, we compute Mean Squared Error, Mean Absolute Error for each regression task while relying on R^2 as the primary measure for the regression studies. Unless explicitly stated otherwise, we use $n_{\text{context}} = 20$ few shot examples per prompt. To control prompt length in tabular settings, we cap the number of features shown per example (10 features maximum). For each level of noise we repeat the experiment with five independent trials. Finally, evaluation instances are processed in batches with 500 rows for tabular datasets and 50 examples for text.

Upon concluding the experiments for each dataset, we fit a linear regression to all performance scores $\mathcal{P}_{k,r}$ and compute a one-sided t-test to decide whether to reject H_0 in Equation (7). We then label the task as *sensitive* to the noise-response protocol (Algorithm 1) if we detect significant deterioration, and as *insensitive* otherwise. Across all statistical tests we use a significance level of $\alpha = 0.05$.

5 Results

We present our results in Figures 1 to 3. For the UCI tabular benchmarks, we observe a heterogeneous response to the noise interventions: 49/138 (35.5%) classification tasks reject the null hypothesis in Equation (7) and thus exhibit sensitivity to the noise schedule. Similarly, for the regression tasks we identify 5/21 (23.8%) tasks that react significantly to the noise schedule. One possible

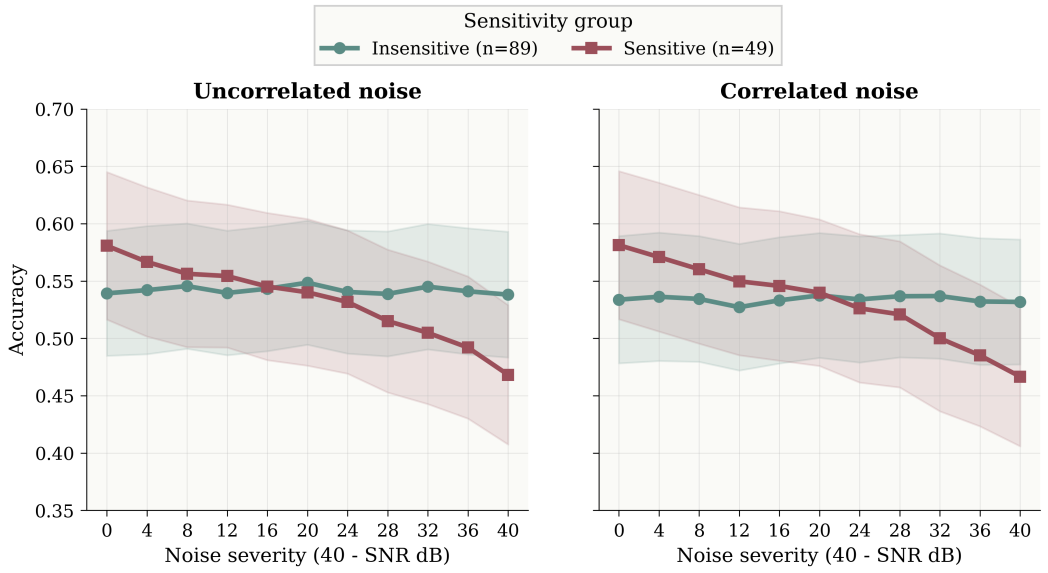

Figure 1: UCI classification accuracy trajectories grouped by noise sensitivity (Sensitive vs Insensitive). Curves are group means across datasets with 95% CI shading, shown separately for uncorrelated and correlated noise.

interpretation is that, on many tabular tasks, the LLM judge relies on shortcuts (e.g., dataset artifacts or memorized associations) rather than robust feature utilization, which may contribute to poor generalization in high-stakes use-cases. Figure 1 shows aggregate accuracy values by sensitivity group and noise type; we see little difference between uncorrelated and correlated noise in these aggregated trajectories.

Tabular behavior While our study aimed to identify tasks where the LLM judge fails under controlled interventions, it is notable that a large fraction of UCI tasks are not measurably influenced by the tested noise patterns. This observation is consistent with recent work suggesting that LLM performance on tabular prediction remains mixed and strongly dependent on task formulation, representation, and prompting choices (Fang et al., 2024; Hegselmann et al., 2023; Sui et al., 2024).

Text-specific behavior We present the per-dataset trajectories in Figure 2, and Figure 3 showing the aggregate lexical trend. Lexical-noise sensitivity is high across all datasets: all four text datasets show statistically significant negative slopes under the one-sided deterioration test. IMDB appears least sensitive, while Yelp Review Full exhibits the largest deterioration. One plausible explanation is that IMDB is a binary task, where the random-guessing floor is 0.5, whereas Yelp is a 5-class sentiment task with a floor of 0.2, making large drops in accuracy more visible under severe corruption. We provide an illustration of the corruption progression in Appendix Figure 6.

5.1 In group vs. outgroup indications on test-set

Suppose that a dataset's sensitivity/insensitivity label is a proxy for whether an LLM judge will generalize to unseen data; how can we validate this thesis? We hypothesize that an LLM judge that proved insensitive to our noise-response protocol (Algorithm 1) will generalize poorly to unseen data in the same domain. To test this, we ran an additional clean-baseline experiment (no additive noise) and compared baseline performance and variability between datasets labeled sensitive vs. insensitive. Due to time constraints, we limited this analysis to datasets with fewer than 1600 examples, yielding 124 datasets total (83 insensitive, 41 sensitive). As before, we run 5 repetitions per dataset and include 20 n_{context} examples from the training set in the prompt. The results are summarized in Figure 4 and Table 2. Overall, insensitive datasets tend to have lower performance; moreover, aside from baseline accuracy, dispersion measures (standard deviation, interquartile range, and trial-to-trial variation) are equal to or more than twice as large for the sensitive group. These patterns are consistent with the interpretation that insensitivity may reflect memorization or reliance on weak

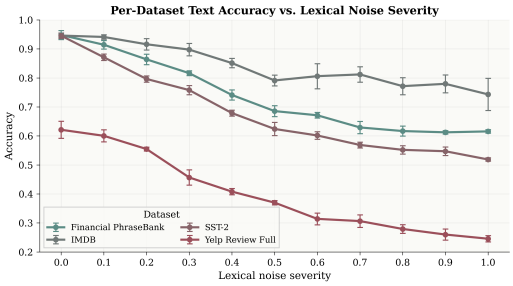

Figure 2: Per-dataset text accuracy vs lexical noise severity. Curves show means across five repetitions, with pointwise 95% confidence intervals.

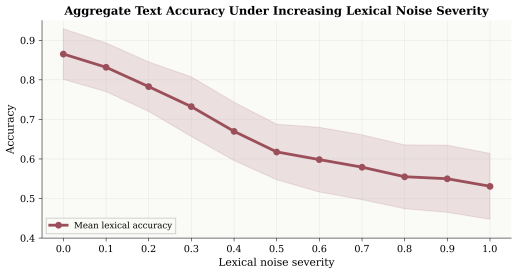

Figure 3: Aggregate text accuracy under increasing lexical noise severity.

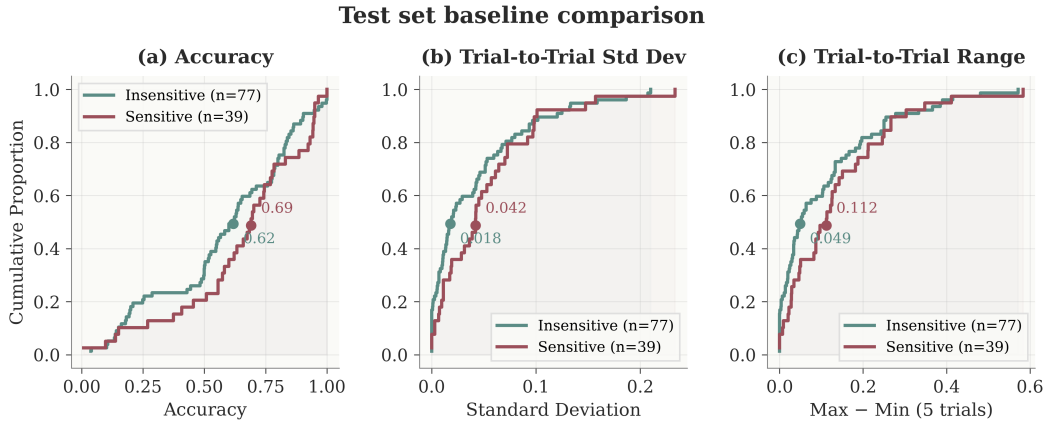

Figure 4: Clean test-set baseline comparison for classification datasets (sensitive vs insensitive): ECDFs for median accuracy, trial-to-trial standard deviation, and trial range.

cues; however, higher variance in the sensitive group can also arise when predictions are closer to the decision boundary. For additional breakdowns, see Appendix Table 5 and Table 6.

5.2 Ablations and sensitivity checks

To guide our model choice, and in particular to assess whether few-shot examples improve performance, we ran model and few-shot ablations with $\{0, 5, 10, 20\}$ context examples for GPT-4O, GPT-4O-MINI, and GPT-5-NANO. As shown in Table 2, all three models benefit from few-shot examples under a paired 0-shot vs. 20-shot comparison: GPT-4O improves from 0.483 to 0.523 (+4.0%), GPT-4O-MINI from 0.412 to 0.443 (+3.0%), and GPT-5-NANO from 0.487 to 0.515 (+2.9 %). At 20-shot, the mean-performance ordering is GPT-4O (0.523) > GPT-5-NANO (0.515) > GPT-4O-MINI (0.443), with a significant gap between GPT-4O and GPT-4O-MINI (+8.0%). Notably, GPT-5-NANO has the strongest 0-shot baseline and shows a smaller incremental gain from additional context than GPT-4O; in this ablation, GPT-4O also remains competitive with the GPT-5 family. We treat these few-shot grid ablations as a *methodological* check, to verify that (i) few-shot context helps on these tasks and (ii) performance is meaningfully model-dependent,rather than as an attempt to exhaustively tune the final judge configuration. Accordingly, for the main noise-response study we fix a single strong setting (GPT-5-MINI with 20-shot context), aiming to operate in a regime where the judge is sufficiently capable on clean inputs so that any observed insensitivity under noise is less likely to be an artifact of inadequate baseline capacity.

Table 2: Test-set baseline comparison (classification). Sensitivity labels derived from training-set perturbation. Test-set uses clean data only (5 trials/dataset). All values are medians; CIs from 10 000 bootstrap resamples.

Metric	Sensitive	Insensitive	Ratio	Δ Median	95% Bootstrap CI
Accuracy	0.691	0.618	1.1×	+0.073	[-0.036, +0.186]
Std Dev	0.0421	0.0178	2.4×	+0.0243	[-0.0013, +0.0394]
IQR	0.0233	0.0116	2.0×	+0.0117	[-0.0021, +0.0490]
Range	0.1121	0.0487	2.3×	+0.0634	[-0.0067, +0.0956]

Table 3: Ablation summary (mean primary metric $\pm$ std across datasets).

Model	0-shot	5-shot	10-shot	20-shot
GPT-4O	0.483 ± 0.235	$\mathbf{0.509} \pm 0.241$	$\mathbf{0.514} \pm 0.240$	0.523 ± 0.245
GPT-4O-MINI	0.412 ± 0.210	0.435 ± 0.209	0.440 ± 0.212	0.443 ± 0.213
GPT-5-NANO	$\mathbf{0.486} \pm 0.257$	0.503 ± 0.258	0.509 ± 0.263	0.515 ± 0.266
GPT-5-MINI baseline	–	–	–	$\mathbf{0.561} \pm 0.253$

6 Limitations and Future Work

While we believe reported findings are indicative of our original thesis, namely, LLM Judges that show indifference to dramatic input changes is an indicator of an untrustworthy judge, several methodological limits remain: Our conclusions are conditional on the chosen perturbation families, we focus on numeric-only tabular interventions and thus under-represent datasets with substantial categorical structure, meanwhile our findings are tied to one primary LLM and therefore do not yet establish cross-model invariance, and finally the linear slope framework favors interpretability but can miss non-linear deterioration regimes.

An additional limitation is diagnostic depth: we identify sensitive versus insensitive regimes but do not fully explain why insensitivity appears for specific datasets. While we performed a simplistic analysis, providing a breakdown by categories of the UCI datasets and their findings (Table 6), a detailed analysis that could fully exonerate potential erroneous interpretations that may not have been fully explained by the our noise protocol. This includes ablations on prompt structure, experiments where we incorporate deniability in the model output (allowing the model to decline uncertain examples) as well as variations on noise schedules not limited by numerical covariates but all covariates in the dataset. This would allow for a more comprehensive understanding of the extent by which the model is confidently uninformed or confidently informed of the problem.

Future directions
Next steps are to expand the study across multiple LLM families to test whether trust decisions are model-invariant or model-specific, extend tabular interventions to categorical and mixed-type features for broader UCI coverage, in addition to expanding towards additional text and image datasets as well, finally we would like to add out-of-distribution plus counterfactual perturbations to separate true causal signal dependence from potentially spurious sensitivity.

7 Conclusion

We introduced an intervention-based calibration protocol for LLM judges under controlled noise. The main empirical finding is a modality gap: lexical perturbations consistently degraded text performance, whereas additive Gaussian noise elicited a response in only 36% of UCI classification tasks and 24% of UCI regression tasks. In a post-hoc clean test-set analysis, noise-insensitive datasets tended to have lower median accuracy, while sensitive datasets showed broader trial-to-trial dispersion, suggesting that absent sensitivity may reflect brittle or weakly grounded behavior. Although preliminary, these results are consistent with noise-response sensitivity tracking whether the LLM judge is grounded in robust, task-relevant signal rather than weak or brittle cues, offering a principled starting point for improving the reliability and trust in LLM-based judges.

REFERENCES

Arthur Asuncion and David Newman. Uci machine learning repository. University of California, Irvine, School of Information and Computer Sciences, 2007. URL https://archive.ics.uci.edu.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback, 2022. URL https://arxiv.org/abs/2212.08073.

Yonatan Belinkov and Yonatan Bisk. Synthetic and natural noise both break neural machine translation. In *Proceedings of ICLR*, 2018. URL https://openreview.net/forum?id=BJ8vJebC-.

Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4): 547–553, 2009. doi: 10.1016/j.dss.2009.05.016.

Richard A. DeMillo, Richard J. Lipton, and Frederick G. Sayward. Hints on test data selection: Help for the practicing programmer. *Computer*, 11(4):34–41, 1978. doi: 10.1109/C-M.1978.218136.

Xi Fang, Weijie Xu, Fiona Anting Tan, Jiani Zhang, Ziqing Hu, Yanjun Qi, Scott Nickleach, Diego Socolinsky, Srinivasan H. Sengamedu, and Christos Faloutsos. Large language models (llms) on tabular data: Prediction, generation, and understanding – a survey. *CoRR*, abs/2402.17944, 2024. doi: 10.48550/arXiv.2402.17944. URL https://doi.org/10.48550/arXiv.2402.17944.

Ronald A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7 (2):179–188, 1936.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. *Commun. ACM*, 64(12):86–92, November 2021. ISSN 0001-0782. doi: 10.1145/3458723. URL https://doi.org/10.1145/3458723.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. Tabllm: Few-shot classification of tabular data with large language models. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 5549–5581. PMLR, 2023. URL https://proceedings.mlr.press/v206/hegselmann23a.html.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, January 2025. ISSN 1558-2868. doi: 10.1145/3703155. URL http://dx.doi.org/10.1145/3703155.

Yue Jia and Mark Harman. An analysis and survey of the development of mutation testing. *IEEE Transactions on Software Engineering*, 37(5):649–678, 2011. doi: 10.1109/TSE.2010.62.

Chungpa Lee, Thomas Zeng, Jongwon Jeong, Jy yong Sohn, and Kangwook Lee. How to correctly report llm-as-a-judge evaluations, 2026. URL https://arxiv.org/abs/2511.21140.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long*

Papers), pp. 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL https://aclanthology.org/2022.acl-long.229/.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: NLG evaluation using GPT-4 with better human alignment, 2023. URL https://arxiv.org/abs/2303.16634.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of ACL: Human Language Technologies*, pp. 142–150, 2011. URL https://aclanthology.org/P11-1015/.

Pekka Malo, Ankur Sinha, Pekka Korhonen, Jyrki Wallenius, and Pyry Takala. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65(4):782–796, 2014. doi: 10.1002/asi.23062.

Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models, 2025. URL https://arxiv.org/abs/2410.05229.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. URL https://arxiv.org/abs/2203.02155.

Judea Pearl. *Causality*. Cambridge University Press, 2009.

Jonas Peters, Dominik Janzing, and Bernhard Scholkopf. *Elements of Causal Inference*. MIT Press, 2017.

Danish Pruthi, Bhuwan Dhingra, and Zachary C. Lipton. Combating adversarial misspellings with robust word recognition. In *Proceedings of ACL*, pp. 5582–5591, 2019. URL https://aclanthology.org/P19-1561/.

George A. F. Seber and Alan J. Lee. *Linear Regression Analysis*. John Wiley & Sons, 2nd edition, 2012. doi: 10.1002/9780471722199.

Nabeel Seedat, Nicolas Huynh, Boris van Breugel, and Mihaela van der Schaar. Curated llm: Synergy of llms and data curation for tabular augmentation in low-data regimes, 2024. URL https://arxiv.org/abs/2312.12112.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context, 2023. URL https://arxiv.org/abs/2302.00093.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. Cheap and fast – but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of EMNLP*, pp. 254–263, 2008. URL https://aclanthology.org/D08-1027/.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*, pp. 1631–1642, 2013. URL https://aclanthology.org/D13-1170/.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. Table meets llm: Can large language models understand structured table data? a benchmark and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pp. 645–654, 2024. doi: 10.1145/3616855.3635752. URL https://doi.org/10.1145/3616855.3635752.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of ICLR*, 2019. URL https://openreview.net/forum?id=rJ4km2R5t7.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Proceedings of NeurIPS*, 2015. URL https://proceedings.neurips.cc/paper/2015/hash/250cf8b51c773f3f8dc8b4be867a9a02-Abstract.html.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

Table 4: Lexical slope estimates (β_1), one-sided p-values, and H_0 decisions.

Dataset	Lexical β_1	One-sided p	Decision
Financial PhraseBank	-0.3687	$< 10^{-10}$	Reject H_0
IMDB	-0.2100	$< 10^{-10}$	Reject H_0
SST-2	-0.4199	$< 10^{-10}$	Reject H_0
Yelp Review Full	-0.4056	$< 10^{-10}$	Reject H_0

A Additional Results

A.1 Per-dataset text lexical slopes

Table 4 lists per-dataset lexical slope estimates for text experiments.

A.2 All UCI per-dataset slopes

Table 5 reports UCI per-dataset slope estimates under uncorrelated and correlated noise, for both classification (accuracy) and regression (R^2). This table covers the full set of datasets with complete per-dataset trajectories in the experiments reported in this paper.

Table 5: Per-dataset UCI slope estimates (β_1) from linear regression of primary metric on noise severity ($40 - \mathrm{SNR}_{\mathrm{dB}}$), with one-sided p-values for $H_1 : \beta_1 < 0$. Buckets are derived using the sensitivity/insensitivity deterioration protocol defined in Algorithm 1. Classification (n=138): Insensitive=89, Sensitive (both)=32, Sensitive (corr-only)=11, Sensitive (uncorr-only)=6. Regression (n=21): Insensitive=16, Sensitive (both)=3, Sensitive (corr-only)=0, Sensitive (uncorr-only)=2.

Task	Dataset	β_1 (Uncorr.)	p (Uncorr.)	β_1 (Corr.)	p (Corr.)	Bucket
Classification	abalone	-0.0002	0.1896	-0.0007	0.0119	Sensitive (corr-only)
Classification	acute_inflammations	-0.0002	0.4039	-0.0000	0.4726	Insensitive
Classification	adult	+0.0008	0.8151	-0.0012	0.1322	Insensitive
Classification	android_permissions	+0.0000	0.7318	-0.0001	0.2699	Insensitive
Classification	annealing	+0.0009	0.6829	+0.0023	0.8607	Insensitive
Classification	auction_verification	-0.0000	0.1738	-0.0000	0.1047	Insensitive
Classification	audiology	+0.0002	0.6896	-0.0002	0.2878	Insensitive
Classification	autism_screening_adult	-0.0002	0.2608	-0.0007	0.0130	Sensitive (corr-only)
Classification	autism_screening_children	-0.0055	$< 1e^{-4}$	-0.0049	$< 1e^{-4}$	Sensitive (both)
Classification	automobile	–	–	–	–	Insensitive
Classification	balance_scale	-0.0087	$< 1e^{-4}$	-0.0076	$< 1e^{-4}$	Sensitive (both)
Classification	balloons	+0.0032	0.9774	-0.0014	0.1878	Insensitive
Classification	bank_marketing	-0.0001	0.2688	+0.0000	0.5582	Insensitive
Classification	banknote_authentication	-0.0009	0.0874	-0.0004	0.2470	Insensitive
Classification	blood_transfusion	-0.0001	0.4184	-0.0007	0.0709	Insensitive
Classification	bone_marrow_transplant	-0.0002	0.2957	–	–	Insensitive
Classification	breast_cancer	-0.0006	0.0221	-0.0005	0.0484	Sensitive (both)
Classification	breast_cancer_coimbra	-0.0002	0.3720	+0.0010	0.9478	Insensitive
Classification	breast_cancer_wisconsin_diagnostic	-0.0010	0.0035	-0.0007	0.0291	Sensitive (uncorr-only)
Classification	breast_cancer_wisconsin_original	-0.0076	$< 1e^{-4}$	-0.0066	$< 1e^{-4}$	Sensitive (both)
Classification	breast_cancer_wisconsin_prognostic	+0.0004	0.9005	+0.0000	0.5134	Sensitive (corr-only)
Classification	car_evaluation	-0.0005	0.3019	+0.0001	0.5493	Insensitive
Classification	cardiotocography	+0.0000	0.5000	+0.0001	0.8262	Insensitive
Classification	cdc_diabetes_indicators	-0.0003	0.3448	-0.0001	0.4114	Insensitive
Classification	census_income	-0.0024	0.0038	-0.0002	0.4092	Sensitive (uncorr-only)
Classification	cervical_cancer_behavior_risk	-0.0018	0.0534	-0.0015	0.0940	Sensitive (both)
Classification	chess_king_rook_vs_king	-0.0001	0.1147	+0.0000	0.6009	Insensitive
Classification	chess_king_rook_vs_king_pawn	+0.0005	0.9968	-0.0002	0.0851	Insensitive
Classification	chronic_kidney_disease	-0.0035	$< 1e^{-4}$	-0.0037	$< 1e^{-4}$	Sensitive (both)
Classification	cirrhosis_survival	-0.0036	$< 1e^{-4}$	-0.0033	$< 1e^{-4}$	Sensitive (both)
Classification	ckd_risk_factors	+0.0004	0.6566	-0.0007	0.2311	Insensitive
Classification	connect_4	-0.0001	0.4264	+0.0010	0.8688	Insensitive
Classification	contraceptive_method_choice	+0.0006	0.8752	+0.0011	0.9872	Insensitive
Classification	covertype	+0.0002	0.7323	-0.0003	0.2363	Insensitive
Classification	credit_approval	-0.0009	0.0010	-0.0008	0.0099	Sensitive (both)
Classification	cylinder_bands	-0.0000	0.1044	+0.0000	0.6225	Insensitive
Classification	default_credit_card	-0.0020	0.0818	-0.0038	0.0061	Sensitive (corr-only)
Classification	dermatology	+0.0002	0.6595	-0.0001	0.3545	Insensitive

Continued on next page

Table 5 continued

Task	Dataset	β_1 (Uncorr.)	p (Uncorr.)	β_1 (Corr.)	p (Corr.)	Bucket
Classification	diabetic_retinopathy	+0.0001	0.6208	+0.0000	0.5718	Insensitive
Classification	dota2_games	-0.0000	0.4447	-0.0000	0.4794	Insensitive
Classification	drug_consumption	-0.0001	0.2661	+0.0000	0.5000	Insensitive
Classification	dry_bean	-0.0043	$< 1e^{-4}$	-0.0016	0.0174	Sensitive (both)
Classification	early_stage_diabetes	+0.0003	0.6591	-0.0010	0.0622	Insensitive
Classification	ecoli	-0.0039	$< 1e^{-4}$	-0.0028	$< 1e^{-4}$	Sensitive (both)
Classification	eeg_eye_state	+0.0000	0.5000	+0.0000	0.5000	Insensitive
Classification	electrical_grid_stability	-0.0000	0.4468	-0.0000	0.2516	Insensitive
Classification	fertility	-0.0003	0.3350	-0.0009	0.0803	Insensitive
Classification	flags	+0.0003	0.8951	-0.0000	0.4622	Insensitive
Classification	garment_productivity	-0.0001	0.0089	-0.0001	0.0020	Sensitive (both)
Classification	gender_by_name	+0.0000	0.5251	-0.0001	0.2565	Insensitive
Classification	glass_identification	-0.0036	$< 1e^{-4}$	-0.0032	$< 1e^{-4}$	Sensitive (both)
Classification	haberman_survival	-0.0018	$< 1e^{-4}$	-0.0024	$< 1e^{-4}$	Sensitive (both)
Classification	hayes_roth	-0.0049	$< 1e^{-4}$	-0.0055	$< 1e^{-4}$	Sensitive (both)
Classification	hcv_data	-0.0045	$< 1e^{-4}$	-0.0045	$< 1e^{-4}$	Sensitive (both)
Classification	heart_disease	-0.0047	$< 1e^{-4}$	-0.0042	$< 1e^{-4}$	Sensitive (both)
Classification	heart_failure_clinical_records	-0.0045	$< 1e^{-4}$	-0.0041	$< 1e^{-4}$	Sensitive (both)
Classification	hepatitis	-0.0004	0.0301	-0.0005	0.0170	Sensitive (corr-only)
Classification	hepatitis_c_virus	-0.0001	0.0843	+0.0000	0.7298	Insensitive
Classification	higher_education_performance	+0.0002	0.6734	-0.0007	0.0295	Insensitive
Classification	horse_colic	-0.0015	0.0199	–	–	Sensitive (uncorr-only)
Classification	htru2	-0.0008	0.0009	-0.0012	0.0006	Sensitive (both)
Classification	ilpd	+0.0019	1.0000	+0.0017	1.0000	Insensitive
Classification	image_segmentation	-0.0025	0.0507	-0.0044	0.0092	Sensitive (both)
Classification	in_vehicle_coupon	+0.0000	0.5475	+0.0002	0.7485	Insensitive
Classification	ionosphere	-0.0001	0.3138	-0.0002	0.0337	Sensitive (corr-only)
Classification	iranian_churn	+0.0003	0.8269	-0.0004	0.1692	Insensitive
Classification	iris	-0.0092	$< 1e^{-4}$	-0.0084	$< 1e^{-4}$	Sensitive (both)
Classification	isolet	-0.0000	0.2703	-0.0001	0.0344	Insensitive
Classification	japanese_credit_screening	-0.0001	0.0504	-0.0000	0.1616	Insensitive
Classification	lenses	-0.0012	0.1729	-0.0005	0.3867	Insensitive
Classification	letter_recognition	+0.0001	0.8953	+0.0001	0.7859	Insensitive
Classification	liver_disorders	-0.0010	0.0089	-0.0011	0.0328	Sensitive (both)
Classification	lung_cancer	-0.0010	0.0368	-0.0008	0.1179	Insensitive
Classification	lymphography	-0.0003	0.3359	-0.0009	0.1317	Insensitive
Classification	magic_gamma_telescope	+0.0004	0.8297	-0.0003	0.2583	Insensitive
Classification	mammographic_mass	-0.0048	$< 1e^{-4}$	-0.0036	$< 1e^{-4}$	Sensitive (both)
Classification	maternal_health_risk	-0.0030	$< 1e^{-4}$	-0.0032	$< 1e^{-4}$	Sensitive (both)
Classification	molecular_biology_splice	+0.0000	0.5000	+0.0000	0.5000	Insensitive
Classification	monks_problems	-0.0049	$< 1e^{-4}$	-0.0050	$< 1e^{-4}$	Sensitive (both)
Classification	mushroom	+0.0004	0.7615	-0.0005	0.1523	Insensitive
Classification	musk_v1	-0.0006	0.2023	+0.0009	0.9304	Insensitive
Classification	musk_v2	+0.0005	0.8262	+0.0008	0.8238	Insensitive
Classification	myocardial_infarction	+0.0000	0.5767	–	–	Insensitive
Classification	nursery	+0.0017	0.9731	-0.0015	0.0244	Sensitive (corr-only)
Classification	obesity_levels	-0.0057	0.0007	-0.0065	0.0001	Sensitive (both)
Classification	occupancy_detection	-0.0001	0.4307	-0.0003	0.2215	Insensitive
Classification	online_shoppers_intention	-0.0008	0.1209	-0.0005	0.1880	Insensitive
Classification	optical_recognition_digits	-0.0000	0.2442	-0.0002	0.0024	Sensitive (corr-only)
Classification	ozone_level_detection	+0.0023	0.9887	-0.0001	0.4521	Insensitive
Classification	page_blocks	+0.0001	0.6251	+0.0012	0.9997	Insensitive
Classification	parkinsons	+0.0005	0.7554	+0.0004	0.7710	Insensitive
Classification	pediatric_appendicitis	-0.0013	0.0758	–	–	Insensitive
Classification	pen_digits	-0.0001	0.3328	-0.0000	0.3256	Insensitive
Classification	phishing_websites	+0.0000	0.5562	-0.0000	0.4411	Insensitive
Classification	poker_hand	-0.0027	$< 1e^{-4}$	-0.0034	$< 1e^{-4}$	Sensitive (both)
Classification	polish_companies_bankruptcy	+0.0000	0.6225	–	–	Insensitive
Classification	post_operative_patient	+0.0001	0.5362	+0.0000	0.5115	Insensitive
Classification	predictive_maintenance	-0.0024	$< 1e^{-4}$	-0.0016	0.0004	Sensitive (both)
Classification	primary_tumor	+0.0017	0.9950	+0.0013	0.9707	Insensitive
Classification	raisin	+0.0006	0.8606	+0.0005	0.8196	Insensitive
Classification	raisin_dataset	-0.0009	0.0514	-0.0012	0.0052	Sensitive (corr-only)
Classification	rice	-0.0011	0.0368	-0.0008	0.0726	Insensitive
Classification	room_occupancy_estimation	-0.0022	0.0013	-0.0030	0.0010	Sensitive (both)
Classification	servo	-0.0004	0.0990	-0.0007	0.0036	Sensitive (both)
Classification	shuttle_landing_control	-0.0010	0.1451	+0.0012	0.9140	Insensitive
Classification	skin_segmentation	-0.0019	$< 1e^{-4}$	-0.0021	$< 1e^{-4}$	Sensitive (both)
Classification	solar_flare	-0.0007	0.1716	-0.0015	0.0355	Insensitive
Classification	sonar	+0.0007	0.9702	-0.0000	0.4928	Insensitive
Classification	soybean_large	-0.0000	0.1196	+0.0000	0.7017	Insensitive
Classification	soybean_small	+0.0004	0.6823	+0.0024	0.9994	Insensitive
Classification	space_shuttle_oring	+0.0000	–	+0.0000	–	Insensitive

Continued on next page

Table 5 continued

Task	Dataset	β_1 (Uncorr.)	p (Uncorr.)	β_1 (Corr.)	p (Corr.)	Bucket
Classification	spambase	+0.0029	1.0000	+0.0011	0.9846	Insensitive
Classification	spect_heart	-0.0028	0.1240	-0.0017	0.2489	Insensitive
Classification	spectf_heart	-0.0003	0.4461	+0.0008	0.6299	Insensitive
Classification	statlog_australian_credit	-0.0003	0.0393	+0.0000	0.5028	Insensitive
Classification	statlog_german_credit	+0.0000	0.6225	+0.0000	0.6225	Insensitive
Classification	statlog_heart	-0.0021	0.0024	-0.0008	0.1294	Sensitive (uncorr-only)
Classification	statlog_image_segmentation	-0.0024	0.0008	-0.0040	$< 1e^{-4}$	Sensitive (both)
Classification	statlog_landsat	-0.0003	0.1784	-0.0007	0.0112	Sensitive (corr-only)
Classification	statlog_shuttle	-0.0008	0.0230	-0.0004	0.1356	Sensitive (uncorr-only)
Classification	statlog_vehicle	-0.0004	0.0842	-0.0004	0.1355	Insensitive
Classification	steel_plates_faults	+0.0000	0.5000	+0.0000	0.5000	Insensitive
Classification	student_academics_performance	+0.0004	0.7696	+0.0001	0.5626	Insensitive
Classification	student_performance	-0.0000	0.4487	+0.0002	0.9596	Insensitive
Classification	taiwanese_bankruptcy	+0.0000	0.7339	+0.0000	0.5000	Insensitive
Classification	thoracic_surgery	+0.0000	0.5000	+0.0001	0.8999	Insensitive
Classification	thyroid_cancer_recurrence	+0.0000	0.5319	+0.0004	0.7767	Insensitive
Classification	tic_tac_toe	-0.0002	0.4207	+0.0014	0.9489	Insensitive
Classification	user_knowledge_modeling	-0.0037	0.0010	-0.0038	$< 1e^{-4}$	Sensitive (both)
Classification	vertebral_column	-0.0029	0.0079	-0.0015	0.0986	Insensitive
Classification	voting_records	-0.0001	0.3331	+0.0001	0.7549	Insensitive
Classification	waveform	-0.0001	0.2656	-0.0003	0.0547	Insensitive
Classification	website_phishing	-0.0005	0.2208	-0.0012	0.0251	Insensitive
Classification	wine	-0.0034	0.0300	-0.0014	0.1969	Insensitive
Classification	wine_quality	-0.0002	0.2812	-0.0003	0.1623	Insensitive
Classification	yeast	-0.0003	0.1673	-0.0008	0.0107	Sensitive (corr-only)
Classification	youtube_spam	-0.0006	0.3147	-0.0020	0.0435	Insensitive
Classification	zoo	-0.0033	0.0129	+0.0016	0.9152	Sensitive (uncorr-only)
Regression	abalone	-0.0021	0.0397	+0.0016	0.8358	Insensitive
Regression	air_quality	+0.0113	0.9515	-0.0027	0.1011	Insensitive
Regression	airfoil_self_noise	+0.0005	0.7006	+0.0002	0.7177	Insensitive
Regression	appliances_energy	-0.0003	0.1510	-0.0050	0.1116	Insensitive
Regression	auto_mpg	-0.0062	$< 1e^{-4}$	-0.0015	0.2315	Sensitive (uncorr-only)
Regression	bike_sharing	-0.0005	0.2273	+0.0014	0.9350	Insensitive
Regression	combined_cycle_power_plant	-0.0018	0.0934	+0.0009	0.7330	Insensitive
Regression	computer_hardware	-0.0202	0.0018	-0.0099	0.0893	Sensitive (both)
Regression	concrete_compressive_strength	+0.0000	0.5064	-0.0008	0.0383	Insensitive
Regression	concrete_slump	-0.0000	0.3550	-0.0002	0.3203	Insensitive
Regression	energy_efficiency	-0.0023	0.0549	-0.0028	0.0593	Insensitive
Regression	forest_fires	+0.0003	0.9083	+0.0003	0.8783	Insensitive
Regression	gas_turbine_emission	+0.0001	0.6587	+0.0001	0.5411	Insensitive
Regression	heart_failure	-0.0004	0.2455	-0.0002	0.1847	Insensitive
Regression	parkinson_telemonitoring	-0.0028	0.0002	-0.0041	0.0071	Sensitive (both)
Regression	power_consumption_tetouan	+0.0007	0.9513	-0.0000	0.4605	Insensitive
Regression	seoul_bike_sharing	-0.0033	0.0232	-0.0017	0.1423	Sensitive (uncorr-only)
Regression	servo	-0.0005	0.1800	+0.0004	0.6813	Insensitive
Regression	student_performance	-0.0008	0.0918	-0.0007	0.1364	Insensitive
Regression	superconductivity	-0.0000	0.2025	-0.0001	0.2004	Sensitive (both)
Regression	wine_quality	-0.0008	0.0941	+0.0002	0.6447	Insensitive

A.3 Noise-induced datapoint shift visualization

Figure 5 illustrates how perturbations shift point clouds and density structure under increasing noise. Rows correspond to representative datasets; columns compare clean data to uncorrelated and correlated noise at matched SNR.

A.4 Breakdown of UCI results by additional levels

In Table 6 we present a breakdown of the results in our study broken down by several categories of interest. For the variety of datasets employed in our UCI study, there is little indication of domain or dataset size dependence. However, we do note possible indications of dependency relative to large sets of features, which may stem from our deliberate limitation of only employing 10 numerical features to mitigate challenges with token length. Similarly it is important to note prompt serialization, for which we have taken a simple approach of dataset description, metadata and covariate distributions in addition to raw numerical values, comparable to studies such as CLLM (Seedat et al., 2024).

Figure 4: 2D Feature Distribution Shift Under Noise - Multiple Datasets

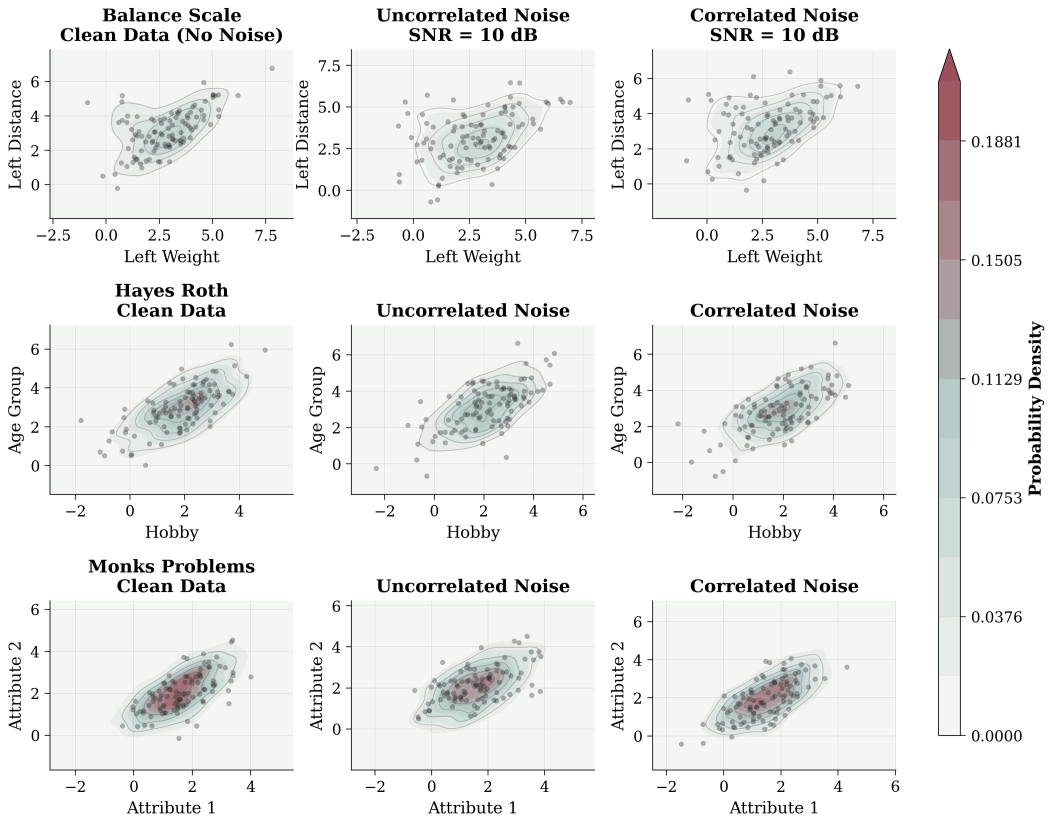

Figure 5: 2D datapoint and density-shift illustration under noise perturbations Left: clean baseline. Middle: uncorrelated Gaussian noise. Right: correlated Gaussian noise.

B Implementation and Reproducibility Details

B.1 IMDB lexical perturbation progression

In Figure 6 below we show a single IMDB review example and the resulting lexical corruption sequence at four severities from 0 to 1.

Factor / Level	Task	N	Sens.	Sensitive %	Breakdown (Count) Both	Corr.	Unc.
Area of Application							
Biology	C	14	8	57.1	3	3	2
	R	1	0	0.0	0	0	0
Business	C	13	3	23.1	2	1	0
	R	2	1	50.0	0	0	1
Comp. Science	C	20	6	30.0	5	1	0
	R	7	2	28.6	2	0	0
Health/Medicine	C	43	17	39.5	13	2	2
	R	1	0	0.0	0	0	0
Physics/Chem	C	17	4	23.5	2	1	1
	R	3	1	33.3	1	0	0
Social Science	C	14	5	35.7	2	2	1
	R	3	0	0.0	0	0	0
Dataset Size (Samples)							
≤ 500	C	54	23	42.6	17	3	3
	R	4	2	50.0	1	0	1
$501 - 1,000$	C	20	8	40.0	5	2	1
$1,001 - 5,000$	C	26	6	23.1	4	2	0
$> 5,000$	C	38	12	31.6	6	4	2
	R	10	3	30.0	2	0	1
Feature Count							
≤ 10	C	48	23	47.9	18	4	1
	R	9	2	22.2	1	0	1
$11 - 20$	C	40	18	45.0	13	2	3
$21 - 50$	C	33	7	21.2	1	4	2
> 50	C	17	1	5.9	0	1	0
	R	2	1	50.0	1	0	0
Class Count (Classification Only)							
2	C	70	21	30.0	12	6	3
3–5	C	33	12	36.4	10	1	1
6–10	C	22	12	54.5	7	3	2
> 10	C	12	4	33.3	3	1	0

Table 6: Sensitivity analysis: Relationship between dataset characteristics and performance deterioration. Bars represent relative sensitivity (%). (C: Classification, R: Regression)

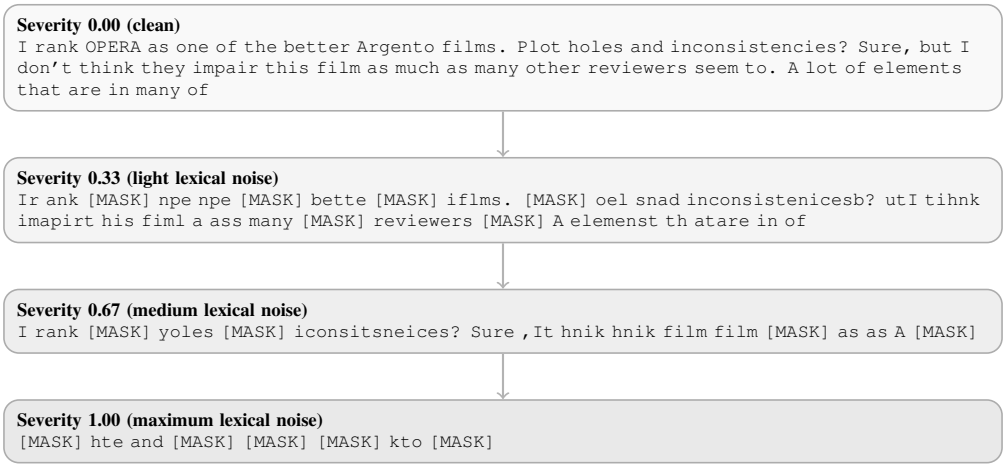

Figure 6: Lexical perturbation progression for a random IMDB sample (`test_022423`, label: positive), generated with the same corruption pipeline used in the experiments. As severity increases, token-level corruption and masking rapidly erase lexical signal.

B.2 Prompt construction and run setup

Prompt template & Execution

System prompt
Each request starts with dynamic dataset context, task definition, label space (or regression target), and explicit output-format constraints.

User prompt
The user message contains few-shot examples as (X, y) pairs, followed by evaluation rows as X only. For tabular settings, inputs are restricted to at most 10 numeric features to control context length.

Run protocol
Each dataset $\times$ noise setting is run for five independent trials using fixed random seeds. Text experiments apply the lexical corruption at the requested severity; tabular experiments apply Gaussian perturbations under correlated or uncorrelated covariance structure.

Output schema
Model outputs are schema-validated against task-specific constraints (label set for classification, numeric parsing for regression). Invalid schema/parse outputs are retried up to 3 times; otherwise counted missing.

C Acknowledgements

We thank Jan Ole Ernst and Lars Holdijk for helpful discussions and thoughtful conversations related to this work.