
Learning Calibrated and Samplable Predictive Distributions of Surface-Ocean $\Delta p\text{CO}_2$ Using Tabular Diffusion Models

Gabriele Accarino^{1*} Mohammad M. Amirian² Thea H. Heimdal³ Amanda R. Fay³
Galen A. McKinley³ Viviana Acquaviva^{2,3}

¹Department of Earth and Environmental Engineering, Columbia University, New York, NY, USA

²CUNY New York City College of Technology, Brooklyn, NY, USA

³Lamont-Doherty Earth Observatory, Columbia University, Palisades, NY, USA

Abstract

The ocean absorbs almost 30% of anthropogenic CO_2 emissions, but quantifying this sink requires estimating surface-ocean CO_2 from sparse observations in space and time. Most existing machine-learning estimates of this field are deterministic, giving a single value per grid point, with uncertainty approximated post hoc via residuals, ensembles, or prescribed distributions. We instead learn, for each observation, a conditional predictive distribution: using Treeffuser, a conditional diffusion model with gradient-boosted-tree score functions, we estimate the full predictive distribution of $\Delta p\text{CO}_2$ (the sea-air $p\text{CO}_2$ difference) from 13 environmental, spatial, and temporal predictors. On a held-out SOCAT test set, Treeffuser attains, among the uncertainty-quantification methods evaluated, the lowest continuous ranked probability score (8.229 μatm) and the coverage closest to the nominal 68% / 95% levels (0.686 / 0.930), while retaining point-prediction skill comparable to a deterministic XGBoost baseline (RMSE 18.4 vs 17.7 μatm ; R^2 0.77 vs 0.79). These results show that conditional diffusion can provide well-calibrated, observation-level uncertainties while largely preserving point-prediction skill. Because Treeffuser is generative, its samples can support probabilistic analyses of $\Delta p\text{CO}_2$; extending this to spatially coherent field reconstruction will require modeling dependence across grid cells and evaluation on spatially held-out data.

1 Introduction

Anthropogenic carbon dioxide (CO_2) emissions are the primary drivers of climate change. The ocean mitigates such effect by absorbing a large fraction of these emissions through physical [1] and chemical processes [2], while biological processes also play an important role in the ocean’s natural carbon cycle [3, 4]. However, substantial uncertainties remain in quantifying ocean carbon uptake [5–7]. Quantifying the ocean carbon sink requires resolving the air–sea CO_2 flux, whose direction and magnitude depend strongly on the difference between surface-ocean and atmospheric CO_2 partial pressures ($\Delta p\text{CO}_2$), together with gas-transfer velocity and CO_2 solubility.

Decades of coordinated international observations have recorded millions of direct surface-ocean $p\text{CO}_2$ measurements [8], but these sample only a small and uneven fraction of the ocean, requiring the reconstruction of the full spatio-temporal field. This problem can be formulated as supervised learning from heterogeneous environmental, spatial, and temporal predictors. Tree-based methods are particularly effective in this setting, as they capture nonlinear interactions with limited preprocessing.

*Corresponding author: ga2673@columbia.edu

Prominent examples include the physically informed $p\text{CO}_2$ -Residual XGBoost framework [9] and its extensions [10, 11], alongside other methods which employ self-organizing maps [12], ensemble feed-forward networks [13], extremely randomized trees [14], and high-resolution target-transformation approaches [15]. Importantly, most existing approaches represent the predicted ocean state by a single point estimate. Existing uncertainty estimates are generally obtained indirectly by repeatedly perturbing model inputs or by comparing predictions across multiple independently developed reconstruction methods evaluated within model-ensemble test beds [16]. These procedures are valuable, but they do not directly learn the conditional distribution of $p\text{CO}_2$ given the observed environmental variables. A generative model that learns a calibrated and samplable distribution, $p(y \mid x)$, would instead make uncertainty an intrinsic component of each prediction, helping to identify poorly constrained conditions and providing a building block for probabilistic inference in the reconstruction problem.

We investigate whether such a model can be obtained without abandoning the tabular, tree-based framework already widely used by the ocean carbon community. We use Treeffuser [17], which represents the score function of a conditional diffusion process using gradient-boosted trees. The model therefore retains the inductive biases that make tree ensembles effective for heterogeneous environmental data while learning a full, samplable conditional distribution $p(y \mid x)$ for each observation, represented directly in the native target space rather than in a learned latent embedding. Using the predictors structure and experimental setup of a state-of-the-art framework [10], we model $\Delta p\text{CO}_2$ observations from 2012 to 2022 as a function of 13 variables, evaluating both point-prediction accuracy and distributional calibration. Our **main contributions** are:

- To our knowledge, the first application of tree-based conditional diffusion to learn samplable predictive distributions of $\Delta p\text{CO}_2$, complementing ensemble- and error-propagation-based uncertainty estimates in existing products.
- Observation-specific uncertainty estimates evaluated using distributional calibration and prediction-interval coverage.
- A comparison with point (including a neural-network) and probabilistic tree-based baselines, showing comparable point-prediction accuracy, the lowest CRPS, and coverage closest to the nominal 68% and 95% levels.

2 Data and task

Observations and Predictors. We use gridded SOCATv2025 [18] $p\text{CO}_2$ monthly observations at $1^\circ \times 1^\circ$ spatial resolution over 2012–2022. The data is tabular and sparse in space and time, with one record per grid cell and month. Following the state-of-the-art $p\text{CO}_2$ -Residual framework [9], each record contains 13 predictors (see Table 1 in [9]): temperature, salinity and base-10 log-transformed chlorophyll concentration at sea-surface together with their corresponding anomalies, log-transformed mixed-layer depth, the atmospheric CO_2 trend, three unit-sphere spatial coordinates $(A, B, C) = (\sin \lambda, \sin \mu \cos \lambda, -\cos \mu \cos \lambda)$ derived from latitude λ and longitude μ , and two annual harmonics $(T_0, T_1) = (\cos \frac{2\pi d}{365}, \sin \frac{2\pi d}{365})$ of the day-of-year d of the (mid-month) observation.

The $\Delta p\text{CO}_2$ Target. We define as target variable $\Delta p\text{CO}_2$, *i.e.* the difference between the fugacity of CO_2 and the atmospheric CO_2 mole fraction, following the approach of [10, 14, 15, 19]:

$$\Delta p\text{CO}_2 = f\text{CO}_2 - x\text{CO}_2.$$

In this definition, $x\text{CO}_2$ varies with time, but has been averaged across the spatial dimensions. The use of $\Delta p\text{CO}_2$ as a target aims to remove the long-term trend of ocean $f\text{CO}_2$, alleviating the issue of modeling an evolving target.

Split. We use a temporally interleaved hold-out: within the global monthly sequence every fifth month is assigned to the test set and the remainder to training, so that test months are spread across all seasons and years (e.g. test months $\{1, 6, 11\}$ in 2012, $\{4, 9\}$ in 2013, and so on). This evaluates temporal *interpolation* between observed months rather than forward extrapolation, and yields 28,879 test records. All models see identical train/test splits and identical predictors.

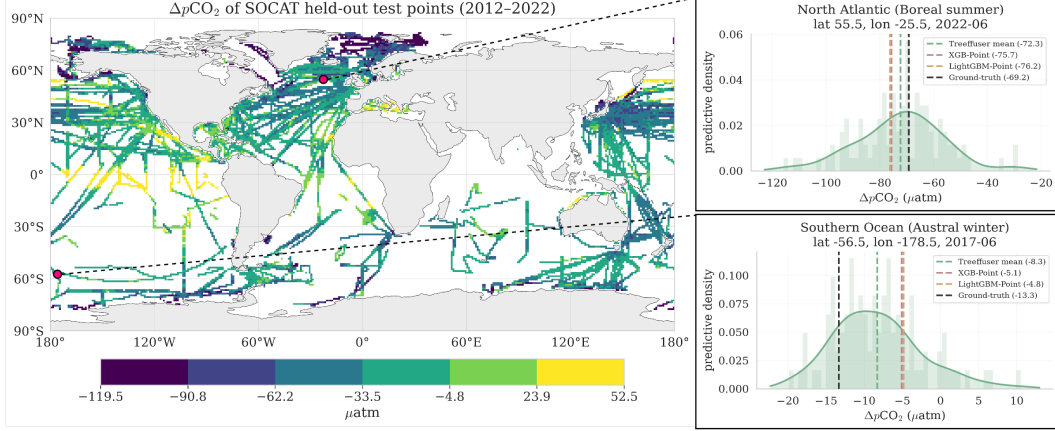


Figure 1: Point estimates versus the full predictive distribution of $\Delta p\text{CO}_2$ at test-set locations. (Left) Spatial distribution of the held-out SOCAT observations used for evaluation. (Right) Treeffuser predictive distributions at the highlighted locations. Vertical lines indicate the Treeffuser predictive mean, the XGB-Point and LightGBM-Point predictions, and the observed SOCAT value.

3 Methods and metrics

Benchmarked models. We focus on tree-based methods and group them by how they represent predictive uncertainty (full descriptions in Appendix B). XGB-Point [20] and LightGBM-Point [21] estimate only the conditional mean and yield no distribution; as a non-tree reference we also include ANN-Point, a feed-forward network matching the regression stage of SOM-FFN [22]. XGB-Quantile fits a pinball-loss model on a grid of levels [23]; XGB-Residual adds a single, input-independent Gaussian noise model to a point prediction; XGB-Ensemble uses the spread of bootstrap-fitted models, analogous to deep ensembles [24]; and NGBoost [25] predicts the parameters of a Gaussian via natural-gradient boosting. Treeffuser [17] learns the conditional distribution of $\Delta p\text{CO}_2$, using LightGBM as the score model. Samples are generated by integrating the reverse-time SDE, giving a samplable distribution with no fixed parametric family. Because its score model is LightGBM, we add a LightGBM-Point baseline to separate the effect of the tree library from that of the diffusion layer.

Fair-comparison protocol. All methods use the same training-test split, predictors, preprocessing, and evaluation observations. Where architectures permit, the tree-based methods also share hyperparameters and a fixed random seed; however, some differences remain as reported in Appendix B.10. In XGB-Ensemble, each member is fitted to a distinct bootstrap resample of the training data (drawn with replacement) using a different seed. For Treeffuser, we use 100 draws from the predictive distribution and 50 integration steps of the underlying SDE, selected by the analysis in Appendix C.

Metrics. We evaluate the predicted distributions using the continuous ranked probability score (CRPS), a strictly proper scoring rule for the full predictive distribution; empirical coverages (Cov68, Cov95) and the corresponding mean widths of the prediction intervals (Wid68, Wid95); and the 68% and 95% interval (Winkler) scores (IntScore68, IntScore95), which penalize both wide intervals and observations falling outside them [26]. Lower values indicate better performance for CRPS, the interval scores, and interval widths, whereas Cov68 and Cov95 are evaluated by their proximity to the corresponding nominal coverage levels of 68% and 95%. Point predictions are evaluated using the Root Mean Squared Error (RMSE) and the coefficient of determination R^2 . All metrics are expressed in μatm , except for Cov68, Cov95, and R^2 , which are dimensionless.

4 Results

Probabilistic accuracy and calibration. Table 1 summarizes probabilistic performance on the SOCAT held-out test set. Treeffuser achieves the lowest CRPS (8.229 μatm), the lowest 95% interval score (81.7 μatm), and coverage closest to the nominal 68% and 95% levels (0.686 and 0.930).

XGB-Quantile is the closest competitor in CRPS (8.339 μatm) and has the lowest 68% interval score (45.6 μatm), but substantially under-covers the observations (0.560 and 0.880), indicating overconfident predictions. NGBoost provides the next-best coverage (0.659 and 0.927) at the cost of wider intervals and higher CRPS and interval scores. XGB-Ensemble is the most overconfident, with narrow intervals covering only 0.230 and 0.425 and yielding the highest interval scores. Overall, Treeffuser offers the best balance of distributional accuracy and calibrated coverage.

The cost in point accuracy is modest. A natural concern is whether modeling the full predictive distribution comes at the expense of point-prediction accuracy. Table 1 shows that Treeffuser’s predictive mean achieves an RMSE of 18.41 μatm and $R^2 = 0.77$, compared with 17.73 μatm and $R^2 = 0.79$ for XGB-Point. Since Treeffuser uses LightGBM as its base learner, we additionally include a LightGBM-Point model with matched hyperparameters. Its performance (RMSE = 17.82 μatm , $R^2 = 0.78$) is close to that of XGB-Point, indicating that the modest reduction in point-prediction accuracy may not be primarily due to the type of base learner, but may instead reflect the additional objective of modeling the full predictive distribution. The neural-network baseline, ANN-Point, is less accurate than all tree-based point models (RMSE = 20.02 μatm , $R^2 = 0.73$), supporting gradient-boosted trees as the backbone for this task. Overall, Treeffuser provides a calibrated, samplable predictive distribution while retaining competitive first-moment accuracy.

Per-observation uncertainty. Figure 1 illustrates Treeffuser predictive distributions at two selected test grid points; at these locations the predictive mean lies closer to the observed $\Delta p\text{CO}_2$ than the baseline point estimates, most visibly in the Southern Ocean, although this is not the case at every location. Each predictive distribution is conditional on the predictors of a single location-month and does not encode correlations between locations or months; the resulting per-observation uncertainty can be mapped and summarized by basin (Figures A3–A5). Its monthly climatology reveals a physically interpretable seasonal structure, with lowest confidence in spring–summer of each hemisphere (Figures A1 and A2).

Table 1: Probabilistic and point scores of candidate models on the SOCAT test set (N=28,879). For point models probabilistic scores are not defined. Best values in bold; for coverage, closest to nominal. Values in μatm except coverage and R^2 .

Model	CRPS	Cov68	Wid68	IntScore68	Cov95	Wid95	IntScore95	RMSE	R^2
Treeffuser	8.229	0.686	27.452	45.8	0.930	55.04	81.7	18.41	0.77
NGBoost	9.717	0.659	31.015	53.0	0.927	61.13	91.2	20.34	0.72
XGB-Quantile	8.339	0.560	20.771	45.6	0.880	50.38	93.0	17.74	0.79
XGB-Residual	8.708	0.729	26.186	50.5	0.900	51.61	121.7	17.73	0.79
XGB-Ensemble	9.190	0.230	7.575	55.1	0.425	14.85	225.1	17.37	0.80
XGB-Point	–	–	–	–	–	–	–	17.73	0.79
LightGBM-Point	–	–	–	–	–	–	–	17.82	0.78
ANN-Point	–	–	–	–	–	–	–	20.02	0.73

5 Discussion

Observation-based $\Delta p\text{CO}_2$ datasets are typically provided as gridded products, with uncertainty either omitted or represented through parametric or heuristic assumptions. Our results show that Treeffuser recovers a *calibrated*, samplable predictive distribution of $\Delta p\text{CO}_2$ for each observation while retaining almost all of the accuracy of XGB-Point. It attains the lowest CRPS and 95% interval score in Table 1, and the coverage closest to nominal 68% and 95%. Calibrated observation-level uncertainty is a prerequisite for carbon-budget applications, where overconfidence would understate flux uncertainty, but it is not sufficient: propagating it to basin or global fluxes also requires the covariance between grid cells and months, which our model does not learn. The cost of obtaining a trustworthy distribution is small, an RMSE increase of 0.68 μatm ($\approx 3.8\%$) and a drop of 0.02 in R^2 relative to XGB-Point, which we regard as a favorable trade for a full predictive distribution, and may potentially be further improved by using a different score model for Treeffuser.

Limitations. Treeffuser’s benefits come at a cost. Sampling requires solving a reverse SDE, so inference is slower than a single tree pass (897 s for the 28,879 test records with 100 draws and

50 integration steps; Appendix C). Our evaluation also covers the easier of two regimes. Because test months are interleaved with training months, test records come from the same years, seasons, and SOCAT-sampled regions as the training data, so the model only interpolates in time between observed months. It does not test extrapolation, that is, prediction at locations SOCAT never sampled or in periods after the training data, where the predictors are supplied by Earth system or global ocean biogeochemistry models; this reconstruction regime is harder and more relevant operationally. Finally, each prediction is a conditional distribution $p(y | x)$ for a single location-month, so the model encodes no spatial or temporal covariance between predictions. Capturing it would require map-based approaches that depart from the tabular SOCAT setting.

Conclusion. We applied Treeffuser, a tree-based diffusion model, to learn the conditional distribution of $\Delta p\text{CO}_2$ given 13 environmental, spatial, and temporal predictors. A single trained model, evaluated at the predictors of each location-month, yields an observation-specific predictive distribution from which samples can be drawn. Rather than embedding observations into a latent space, this approach preserves the tabular structure of SOCAT and the tree-based paradigm already adopted by the ocean carbon community, replacing a single predicted value with a full conditional distribution. Among the methods tested, it produced the best-calibrated predictions at only a modest cost in first-moment accuracy. These results point to a practical way of equipping data-driven ocean carbon products with samplable, observation-level predictive uncertainty. Because each distribution describes one location-month independently of the others, combining the model with a representation of spatio-temporal dependence, and evaluating it on spatially held-out data, are the next steps toward probabilistic reconstruction of the sparse and biased observational field.

Acknowledgments and Disclosure of Funding

We acknowledge funding from NSF through the Learning the Earth with Artificial intelligence and Physics (LEAP) Science and Technology Center (STC) (Award #2019625). VA acknowledges support from a PIVOT Research award (Award #12871) from the Simons Foundation. GAM, THH and ARF acknowledge support from US funding agencies (NOAA NA24OARX431G0151-T1-01 and NSF OCE 2333608), from Schmidt Sciences, LLC for the COCO2 project, and the European Space Agency (ESA) through the Ocean Carbon for Climate (OC4C; 3-18399/24/I-NB) project. The Surface Ocean CO₂ Observing Network (SOCONET), the Surface Ocean CO₂ Atlas (SOCAT) and the Surface Ocean CO₂ Mapping (SOCOM) program are linked international efforts that, along with additional critical data from satellites, models and re-analysis, enable the routine delivery of ocean carbon and carbonate system data products. The loss of any activity (SOCONET, SOCAT, or SOCOM) or data source (satellite, in situ, or re-analysis) would prevent delivery of data products to global or regional assessments (Global Carbon Budget, IPCC, and others) that guide policy decisions. SOCONET, SOCAT, and SOCOM are endorsed by the International Ocean Carbon Coordination Project (IOCCP, an Infrastructural Project of the Scientific Committee on Oceanic Research) and the Global Ocean Observing System (GOOS) Biogeochemistry Panel. The many researchers, funders, and agencies responsible for these activities and data are thanked for their contributions. We thank Maurie Keppens for her support in configuring the neural network used as one of the baseline models in this work.

References

- [1] Rik Wanninkhof. Relationship between wind speed and gas exchange over the ocean revisited. *Limnology and Oceanography: Methods*, 12(6):351–362, 2014.
- [2] Richard E Zeebe and Dieter Wolf-Gladrow. *CO₂ in seawater: equilibrium, kinetics, isotopes*, volume 65. Gulf Professional Publishing, 2001.
- [3] Mohammad M. Amirian, Zoe V. Finkel, Emmanuel Devred, and Andrew J. Irwin. Parameterization of photoinhibition for phytoplankton. *Communications Earth & Environment*, 6(1):707, 2025.
- [4] E Devred, S Clay, M Ringuette, T Perry, M Amirian, A Irwin, and Z Finkel. Net primary production in the labrador sea between 2014 and 2022 derived from ocean colour remote sensing based on ecological regimes. *Remote Sensing of Environment*, 323:114713, 2025.
- [5] GA McKinley. Global carbon budget rebalanced. *Nature: International Weekly Journal of Science*, (8095), 2026.
- [6] Pierre Friedlingstein, Michael O’sullivan, Matthew W Jones, Robbie M Andrew, Dorothee CE Bakker, Judith Hauck, Peter Landschützer, Corinne Le Quéré, Hongmei Li, Ingrid T Luijkx, et al. Global carbon budget 2025. 2025.
- [7] Galen A McKinley, Amanda R Fay, Nicole S Lovenduski, and Darren J Pilcher. Natural variability and anthropogenic trends in the ocean carbon sink. *Annual review of marine science*, 9:125–150, 2017.
- [8] Dorothee C E Bakker, Benjamin Pfeil, Camilla S Landa, et al. A multi-decade record of high-quality fCO₂ data in version 3 of the Surface Ocean CO₂ Atlas (SOCAT). *Earth System Science Data*, 8(2):383–413, 2016.
- [9] Val Bennington, Tom Galjanic, and Galen A McKinley. Explicit physical knowledge in machine learning for ocean carbon flux reconstruction: The pCO₂-residual method. *Journal of Advances in Modeling Earth Systems*, 14(10):e2021MS002960, 2022.
- [10] Thea Hatlen Heimdal, Amanda R Fay, Abby P Shaum, Viviana Acquaviva, Val Bennington, Amelie E Sharples, Nicoline M Joenson, and Galen A McKinley. An update of the Ideo fco₂-residual data product: algorithmic choices improve ocean carbon sink estimates. *Machine Learning: Earth*, 2026.
- [11] Thea H Heimdal, Abby P Shaum, Viviana Acquaviva, Amanda R Fay, Devan Samant, Julius Busecke, and Galen A McKinley. Targeting bias in algorithm optimization improves reconstructions of surface ocean pco₂. *Machine Learning: Earth*, 1(1):015003, jul 2025. doi: 10.1088/3049-4753/adddc3. URL <https://doi.org/10.1088/3049-4753/adddc3>.
- [12] Peter Landschützer, Nicolas Gruber, and Dorothee C E Bakker. Decadal variations and trends of the global ocean carbon sink. *Global Biogeochemical Cycles*, 30(10):1396–1417, 2016.
- [13] Thi Tuyet Trang Chau, Marion Gehlen, and Frédéric Chevallier. A seamless ensemble-based reconstruction of surface ocean pCO₂ and air–sea CO₂ fluxes over the global coastal and open oceans. *Biogeosciences*, 19(4):1087–1109, 2022.
- [14] Rik Wanninkhof et al. Trends in sea–air CO₂ fluxes and sensitivities to atmospheric forcing using an extremely randomized trees machine learning approach. *Global Biogeochemical Cycles*, 39:e2024GB008315, 2025.
- [15] Luke Gregor, Jamie Shutler, and Nicolas Gruber. High-resolution variability of the ocean carbon sink. *Global Biogeochemical Cycles*, 38(8):e2024GB008127, 2024.
- [16] Lucas Gloege, Galen A McKinley, Peter Landschützer, et al. Quantifying errors in observationally based estimates of ocean carbon sink variability. *Global Biogeochemical Cycles*, 35(4): e2020GB006788, 2021.

- [17] Nicolas Beltran-Velez, Alessandro Antonio Grande, Achille Nazaret, Alp Kucukelbir, and David Blei. Treeffuser: Probabilistic predictions via conditional diffusions with gradient-boosted trees. In *Advances in Neural Information Processing Systems*, 2024.
- [18] Dorothee C E Bakker, Simone R Alin, Takafumi Aramaki, Leticia Barbero, Nicholas Bates, Thanos Gkritzalis, Steve D Jones, Alex Kozyr, Siv K Lauvset, Vlad A Macovei, et al. Surface ocean co2 atlas database version 2025 (socatv2025)(ncei accession 0304549). 2025.
- [19] A. R. Fay, D. R. Munro, G. A. McKinley, D. Pierrot, S. C. Sutherland, C. Sweeney, and R. Wanninkhof. Updated climatological mean δf_{CO_2} and net sea–air CO_2 flux over the global open ocean regions. *Earth System Science Data*, 16(4):2123–2139, 2024. doi: 10.5194/essd-16-2123-2024. URL <https://essd.copernicus.org/articles/16/2123/2024/>.
- [20] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.
- [21] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30, 2017.
- [22] Peter Landschützer, Nicolas Gruber, Dorothee C. E. Bakker, Ute Schuster, Shin ichiro Nakaoka, Mark R. Payne, Tristan P. Sasse, and Jiye Zeng. A neural network-based estimate of the seasonal to inter-annual variability of the atlantic ocean carbon sink. *Biogeosciences*, 10(11):7793–7815, 2013. doi: 10.5194/bg-10-7793-2013.
- [23] Roger Koenker and Gilbert Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978.
- [24] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30, 2017.
- [25] Tony Duan et al. NGBoost: Natural gradient boosting for probabilistic prediction. In *International Conference on Machine Learning*, pages 2690–2700, 2020.
- [26] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.
- [27] Peter Landschützer, Nicolas Gruber, Dorothee C. E. Bakker, and Ute Schuster. Recent variability of the global ocean carbon sink. *Global Biogeochemical Cycles*, 28(9):927–949, 2014. doi: 10.1002/2014GB004853.
- [28] Daniel J. Ford, Josh Blannin, Jennifer Watts, Andrew J. Watson, Peter Landschützer, Annika Jersild, and Jamie D. Shutler. A comprehensive analysis of air-sea CO_2 flux uncertainties constructed from surface ocean data products. *Global Biogeochemical Cycles*, 38(11): e2024GB008188, 2024. doi: 10.1029/2024GB008188.
- [29] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.
- [30] Nicolai Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7: 983–999, 2006.
- [31] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.
- [32] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011.
- [33] Brian D O Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

- [34] François Rozet, Ruben Ohana, Michael McCabe, Gilles Louppe, François Lanusse, and Shirley Ho. Lost in latent space: An empirical study of latent diffusion models for physics emulation. *Advances in Neural Information Processing Systems*, 38:134612–134657, 2026.
- [35] Gabriele Accarino, Juan Nathaniel, Carla Roesch, Pierre Gentine, Sara Shamekh, Duncan Watson-Parris, and Viviana Acquaviva. Wavelet flow matching for multi-scale physics emulation. *arXiv preprint arXiv:2605.16573*, 2026.
- [36] Alexander Jordan, Fabian Krüger, and Sebastian Lerch. Evaluating probabilistic forecasts with scoringRules. *Journal of Statistical Software*, 90(12):1–37, 2019.
- [37] Robert L Winkler. A decision-theoretic approach to interval estimation. *Journal of the American Statistical Association*, 67(337):187–191, 1972.

A Additional Results

Table A1: Treeffuser predictive standard deviation (σ) of the $\Delta p\text{CO}_2$, by ocean basin (2012–2022). Most/least confident months are the climatological maxima/minima of the mean per-cell σ ; MAD columns give the basin-monthly mean absolute deviation of the Treeffuser predictive mean from the XGB-Point prediction and from observations. Values in μatm .

Basin	N	$\bar{\sigma}$	Most confident		Least confident		MAD vs.	
			Month	$\sigma_{\min}$	Month	$\sigma_{\max}$	XGB-Point	Obs.
Atlantic	10,497	16.89	Jan	11.98	May	22.68	0.87	1.31
Pacific	13,588	15.13	Feb	10.91	Jul	18.65	0.68	1.78
Southern	3,561	16.80	Sep	10.05	Jan	19.15	1.27	3.52

Seasonal structure of predictive uncertainty. Figure A1 shows the monthly climatology of $\Delta p\text{CO}_2$ and its predictive uncertainty across ocean basins. Treeffuser closely follows both XGB-Point and the gridded observations: its basin-monthly mean absolute difference is at most 1.27 μatm from XGB-Point and 3.52 μatm from the observations (Table A1), indicating that distributional modeling retains the central accuracy of the point model. Predictive uncertainty is nevertheless strongly seasonal. Basin-mean σ values are similar (15.13 μatm to 16.89 μatm), but monthly values range from 10.05 μatm to 11.98 μatm in the most-confident months to 18.65 μatm to 22.68 μatm in the least-confident months. The least-confident months generally occur during spring or summer in the corresponding hemisphere, plausibly reflecting greater variability in the $\Delta p\text{CO}_2$ associated with biological activity, stratification, and episodic mixing. Interannual variation is shown in Figure A2.

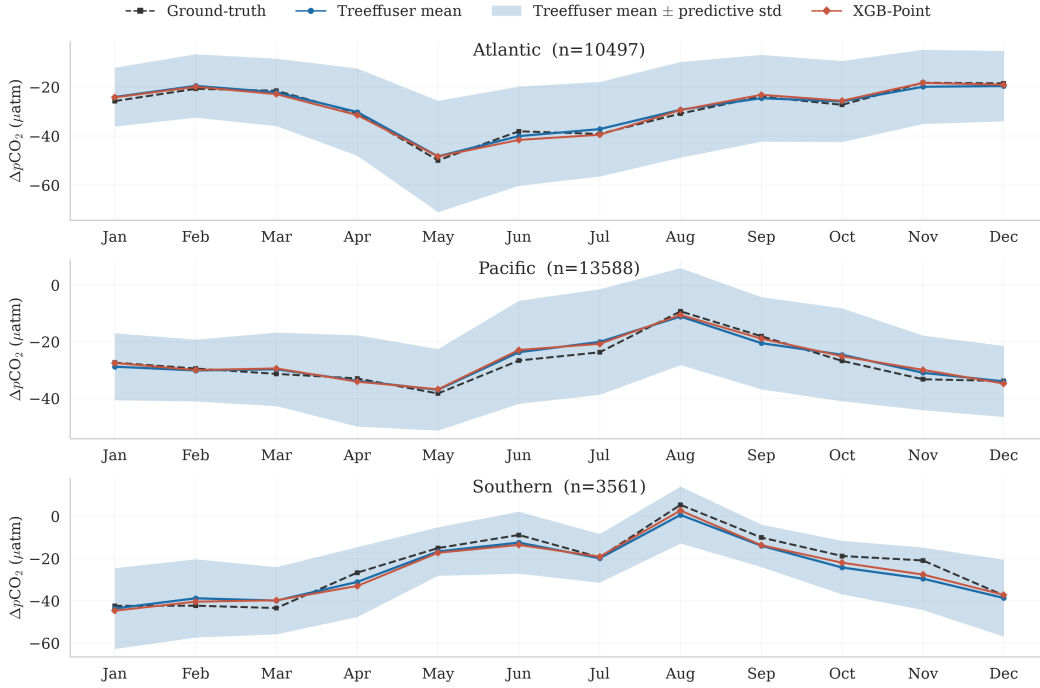


Figure A1: Monthly climatology of $\Delta p\text{CO}_2$ across ocean basins (2012–2022). Solid line: Treeffuser predictive-mean, basin-averaged; shaded band: $\pm$ mean per-cell predictive standard deviation; dashed line: gridded observations; diamonds: XGB-Point. Axes are free per basin.

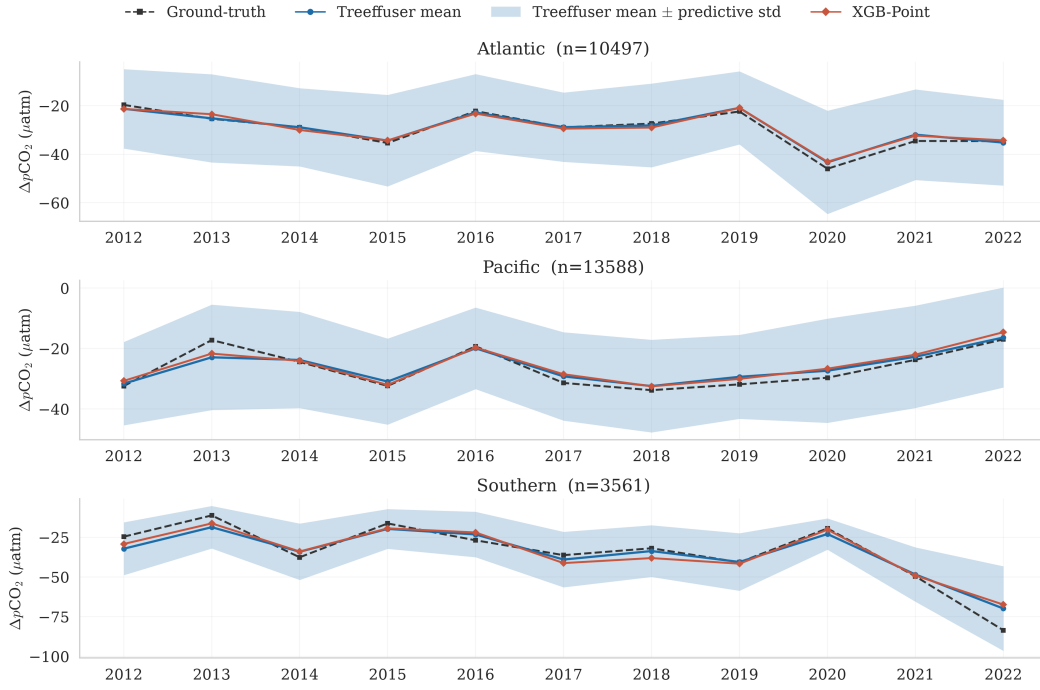


Figure A2: Interannual variation of $\Delta p\text{CO}_2$ across ocean basins (2012–2022). Each point is an annual mean over all cells in the basin. Solid line: Treeffuser predictive-mean, basin-averaged; shaded band: $\pm$ mean per-cell predictive standard deviation; dashed line: gridded observations; diamonds: XGB-Point. Axes are free per basin.

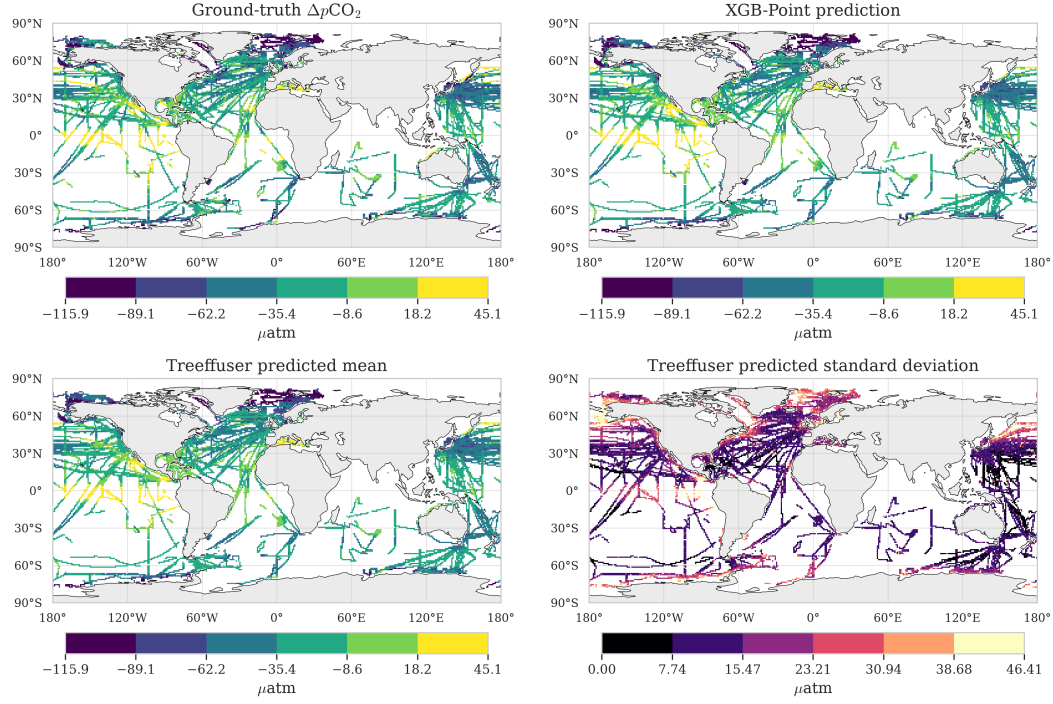


Figure A3: Global maps of test-set $\Delta p\text{CO}_2$, temporally averaged per 1° cell over 2012–2022. Top row: gridded observations (left) and the deterministic XGB-Point prediction (right). Bottom row: Treeffuser predictive mean (left) and per-cell predictive standard deviation (right). The three mean fields share a common colour scale, whereas the standard-deviation panel uses a separate scale (magma). Values in μatm .

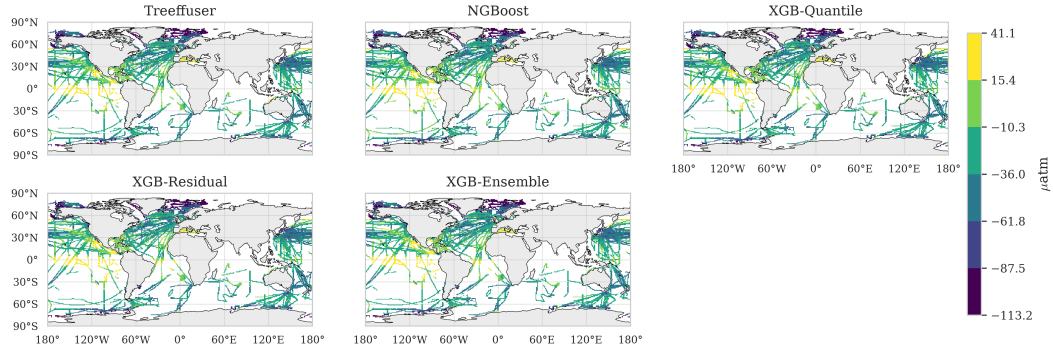


Figure A4: Predicted mean $\Delta p\text{CO}_2$ by model, temporally averaged per grid point over 2012–2022.

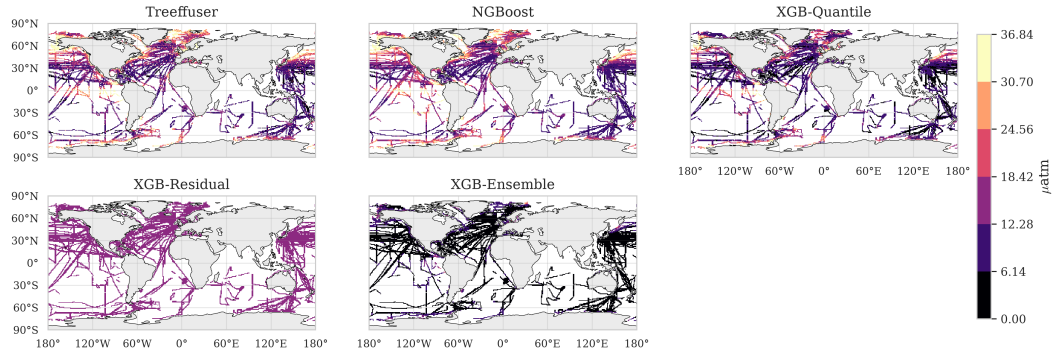


Figure A5: Predicted σ of $\Delta p\text{CO}_2$ by model, temporally averaged per grid point over 2012–2022.

B Model descriptions and training details

This appendix specifies every benchmarked method: its objective, how it is trained, the distributional object it produces, and the properties that explain its behaviour in Table 1. We first fix notation and the shared protocol, then describe the deterministic baselines (XGB-Point, LightGBM-Point, and ANN-Point), the three XGB uncertainty variants, NGBoost, and finally the proposed method, Treeffuser.

B.1 Notation and shared protocol

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ with covariates $x_i \in \mathbb{R}^{13}$ (the predictors of Section 2) and scalar target y_i equal to the $\Delta p\text{CO}_2$. All methods estimate a functional of the conditional law $p(y \mid x)$. The deterministic baselines (XGB-Point, LightGBM-Point, and ANN-Point) estimate only the conditional mean $\mu(x) = \mathbb{E}[y \mid x]$. The remaining methods estimate quantiles, an ensemble spread, a parametric density, or the full density.

Every method uses the *same* temporally interleaved train/test split (Section 2) and the *same* 13 predictors. The tree-based methods are trained on unscaled inputs, since tree ensembles are invariant to monotone feature transforms. The two exceptions are ANN-Point, which z-scores its inputs and target using training-set statistics because a neural network is sensitive to input scale (Appendix B.4), and Treeffuser, which standardizes covariates and target internally before diffusion (Appendix B.9). All predictions are mapped back to μatm before scoring.

To isolate the effect of the uncertainty mechanism from that of tuning, all tree-based methods share the same boosting hyperparameters (`n_estimators=1000`, `max_depth=9`, `learning_rate=0.05`) wherever the method admits them, and all use the same fixed random seed. These values come from a prior grid search of XGB-Point on normalized mean squared error and were not re-tuned for any individual method. The sole exception to the shared seed is XGB-Ensemble, whose members must differ and which therefore uses a grid of seeds (Appendix B.7).

Metrics are reported in Table 1. Point-accuracy metrics use each method’s mean prediction (the median for XGB-Quantile). Probabilistic metrics are computed in closed form for the Gaussian methods (NGBoost, XGB-Residual), from the predicted quantile grid for XGB-Quantile, and from empirical samples for XGB-Ensemble and Treeffuser (Appendix D). Hyperparameters are collected in Tables B1 and B2.

B.2 XGB-Point: deterministic baseline

XGB-Point is a gradient-boosted regression-tree model [20] trained by second-order (Newton) boosting under the squared-error loss. The predictor is an additive ensemble $\hat{\mu}(x) = \sum_{k=1}^K f_k(x)$ of CART trees f_k ; at each round the new tree minimizes a second-order Taylor expansion of the loss augmented with a complexity penalty $\Omega(f) = \gamma L + \frac{1}{2}\lambda\|w\|_2^2 + \alpha\|w\|_1$ over the L leaf weights w , and is added with shrinkage (learning rate) η . Row and column subsampling provide additional regularization. XGB-Point returns only the conditional mean and therefore carries no uncertainty; it plays two roles in this study: it is the point-prediction reference in Table 1, and it is the shared tree backbone reused (with identical hyperparameters and seed) by XGB-Residual and XGB-Ensemble, and mirrored by XGB-Quantile.

B.3 LightGBM-Point: deterministic baseline

LightGBM-Point is a gradient-boosted regression-tree model [21] trained, like XGB-Point, by second-order boosting under the squared-error loss, yielding an additive ensemble $\hat{\mu}(x) = \sum_{k=1}^K f_k(x)$ of regression trees added with shrinkage η and regularized by the same row- and column-subsampling controls. It differs from XGB-Point in the two implementation choices that characterize the library. First, continuous predictors are bucketed into a fixed number of histogram bins and candidate splits are searched over these bins rather than over sorted feature values, lowering the cost of split finding; the histogram learner further incorporates exclusive feature bundling and gradient-based one-side sampling, the accelerations introduced in [21]. Second, trees are grown leaf-wise (best-first), at each step the leaf offering the largest loss reduction is split, rather than level-wise, here capped at the shared `max_depth`. Under the shared hyperparameters (Section B.1) these differences leave

the two libraries closely comparable in capacity. Like XGB-Point, LightGBM-Point returns only the conditional mean and therefore carries no uncertainty; it plays two roles in this study: it is a second deterministic point reference in Table 1, and it exposes the gradient-boosting backbone on which Treeffuser learns its score function, so that the gap between its prediction and the Treeffuser predictive mean isolates the contribution of the probabilistic (diffusion) layer from the choice of tree library.

B.4 ANN-Point: feed-forward neural network baseline

ANN-Point is a fully connected feed-forward neural network (FFN), the non-tree model family most widely used in observation-based $p\text{CO}_2$ reconstruction: it is the regression stage of the SOM-FFN method [22, 27] and of the UExP-FNN-U product [28]. Those methods first cluster the ocean into biogeochemical provinces with a self-organizing map (SOM) and then fit one FFN per province. We reproduce only the regression stage and train a single global network on the same 13 predictors and the same train/test split as all other methods. This mirrors the single global models used for the tree-based methods, so that differences reflect the model family rather than the regionalization. The predictor is

$$\hat{\mu}(x) = w_2^\top \tanh(W_1 \tilde{x} + b_1) + b_2, \quad (\text{B1})$$

with one hidden layer of $H = 210$ units, a hyperbolic-tangent (tangent-sigmoid) activation and a linear output, as in Landschützer et al. [22, 27]. Unlike the tree ensembles, a neural network is sensitive to the scale of its inputs. We therefore z-score each predictor ($\tilde{x}$) and the target using training-set statistics, and invert the target transform at prediction time.

Training follows the cited works where an equivalent exists. The training rows are randomly split 70/30 into fitting and validation subsets [28]. The squared-error loss is minimized without weight decay, and training stops once the validation loss has not improved for 6 consecutive epochs. Early stopping is therefore the only regularizer, as in the original SOM-FFN implementations. The width H was selected from $\{30, 60, \dots, 300\}$ by validation error. The original works train with Levenberg–Marquardt, which is not available in scikit-learn. We instead use Adam [29] (learning rate 10^{-3} , mini-batches of 512, at most 1000 epochs) via scikit-learn’s `MLPRegressor`, with the shared seed. Like XGB-Point and LightGBM-Point, ANN-Point returns only the conditional mean and carries no uncertainty. It is included as the neural-network point reference in Table 1. Its higher error (RMSE 20.02 vs. 17.73 μatm for XGB-Point) is consistent with the use of gradient-boosted trees as the backbone of the probabilistic methods benchmarked here.

B.5 XGB-Quantile

XGB-Quantile keeps the boosting machinery of XGB-Point but replaces the loss with the pinball (quantile) loss, so that gradient boosting performs conditional quantile regression [23, 30]. For a quantile level $\tau \in (0, 1)$ and prediction q ,

$$\ell_\tau(y, q) = (\tau - \mathbf{1}\{y < q\}) (y - q), \quad (\text{B2})$$

whose minimizer is the conditional τ -quantile $q_\tau(x)$. We fit a grid of 39 levels $\tau \in \mathcal{Q} = \{0.025, 0.05, \dots, 0.975\}$ jointly with XGBoost’s native multi-quantile objective (`reg:quantileerror`), using a single model. The predicted quantile function is made monotone by sorting (rearrangement) to remove any quantile crossing. Central 95% intervals use the grid levels (0.025, 0.975) directly; central 68% intervals use (0.16, 0.84), obtained by linear interpolation between adjacent grid levels. XGB-Quantile is distribution-free and heteroscedastic, but it represents the predictive law only as finely as the quantile grid, does not model the tails beyond the extreme levels, and estimates each level independently; on our task it is sharp but under-covered (Table 1), i.e. mildly overconfident.

B.6 XGB-Residual

XGB-Residual augments the XGB-Point mean predictor $\hat{\mu}(x)$ with a single, input-independent Gaussian noise model. We first fit XGB-Point on a random 80% of the training records and estimate the residual standard deviation $\hat{\sigma}$ from its residuals $e_i = y_i - \hat{\mu}(x_i)$ on the remaining 20%, which gives $\hat{\sigma} = 13.17 \mu\text{atm}$. The mean predictor $\hat{\mu}(x)$ is then refit on the full training set with the shared hyperparameters and seed. The predictive law is the homoscedastic Gaussian $\hat{F}(\cdot | x) =$

$\mathcal{N}(\hat{\mu}(x), \hat{\sigma}^2)$, so its CRPS and prediction intervals use the Gaussian closed forms (Appendix D). Because $\hat{\sigma}$ does not depend on x , the predictive spread is *constant* across the ocean. The method captures the average noise level but cannot widen in more uncertain regions. This yields reasonable central coverage but a poor interval score wherever the true spread varies with x (Table 1). Its point predictions are identical to those of XGB-Point.

B.7 XGB-Ensemble (bootstrap ensemble)

XGB-Ensemble estimates uncertainty from the disagreement of $B = 100$ XGB models trained on independent bootstrap resamples of the training rows, the tree analogue of deep ensembles [24]. Each member b is trained on rows resampled with replacement under a distinct random seed, the “seed grid” that permutes the training set, while all boosting hyperparameters are held fixed and equal to the shared backbone. The predictive distribution is the empirical distribution of the member predictions $\{\hat{\mu}_b(x)\}_{b=1}^B$, from which we take the mean, standard deviation, and quantiles. Crucially, this spread reflects only *epistemic* variance induced by resampling the finite training set; it does *not* include the *aleatoric* observation and process noise present in $p\text{CO}_2$. The intervals are therefore systematically too narrow, which is exactly the strong overconfidence observed in Table 1 (95% coverage of only 0.425).

B.8 NGBoost

NGBoost performs probabilistic regression by using natural-gradient boosting to learn the parameters of a parametric predictive distribution [25]. We use a Gaussian predictive law, $y \mid x \sim \mathcal{N}(\mu(x), \sigma(x)^2)$, with parameters $\theta(x) = (\mu(x), \log \sigma(x))$. Parameterizing the scale through $\log \sigma$ ensures that $\sigma > 0$. At each boosting stage, NGBoost fits a separate set of shallow regression trees for each distributional parameter. These learners are fitted to the natural gradient, $\mathcal{I}(\theta)^{-1} \nabla_{\theta} S(\theta, y)$, where $\mathcal{I}(\theta)$ is the Fisher information matrix and S is a strictly proper scoring rule. We use the default log score, $S = -\log \mathcal{N}(y; \mu, \sigma^2)$, corresponding to maximum likelihood. A line search determines the step size at each stage, and the resulting update is scaled by the learning rate. The use of the natural gradient makes the optimization invariant to the particular parameterization of the predictive distribution and helps stabilize the joint estimation of μ and σ . The resulting predictive distribution is Gaussian, allowing quantiles, coverage, and CRPS to be computed analytically. Because $\sigma(x)$ is learned as a function of the input, NGBoost can represent heteroscedastic uncertainty. Its main limitation is the Gaussian assumption: it cannot capture skewness, heavy tails, or multimodality, which limits its performance relative to the nonparametric Treeffuser on CRPS and interval score.

B.9 Treeffuser (proposed)

Treeffuser [17] is a conditional score-based diffusion model [31] whose score function is estimated with gradient-boosted trees [21] rather than a neural network. It learns the full, nonparametric conditional density $p(y \mid x)$ and produces predictions by drawing samples from the learned conditional distribution. All quantities we report, including mean, per-cell standard deviation, intervals, CRPS, are computed from these samples.

Forward diffusion. A variance-exploding SDE (VESDE) progressively corrupts the target on $t \in [0, 1]$,

$$dY = \sqrt{2\sigma(t)\sigma'(t)} dW, \quad \sigma(t) = \sigma_{\min} \left(\frac{\sigma_{\max}}{\sigma_{\min}} \right)^t, \quad (\text{B3})$$

with $\sigma_{\min} = 0.01$ and $\sigma_{\max} = 20$. Because the drift is zero and the diffusion is state-independent, the perturbation kernel is Gaussian in closed form,

$$p_t(y_t \mid y_0) = \mathcal{N}(y_0, (\sigma(t)^2 - \sigma(0)^2)), \quad (\text{B4})$$

and as $t \rightarrow 1$ the marginal approaches the prior $p_1 \approx \mathcal{N}(0, \sigma_{\max}^2)$, from which sampling starts. The covariates x are held fixed throughout and never diffused.

Score parameterization. Treeffuser models the conditional (Stein) score $s(y_t, x, t) \approx \nabla_y \log p_t(y_t \mid x)$ as

$$s(y_t, x, t) = \frac{g(y_t, x, t)}{\sigma(t)}, \quad (\text{B5})$$

where g is a single LightGBM regressor whose input is the concatenation $[y_t, x, t]$: the perturbed target, the covariates, and the diffusion time entered directly as an ordinary tabular feature. Feeding t as a plain feature removes the need for the sinusoidal time embeddings used by neural score networks, and lets the model inherit the strengths of gradient-boosted trees, native handling of heterogeneous, skewed, and missing tabular inputs, scale invariance, and fast training.

Training objective (denoising score matching). The score is fit by denoising score matching (DSM) [32]. Each training pair is replicated R times ($R = 30$); for every replicate we draw a time $t \sim \mathcal{U}(\varepsilon, 1)$ ($\varepsilon = 10^{-5}$) and noise $z \sim \mathcal{N}(0, 1)$, form the perturbed target $y_t = y_0 + \sigma(t) z$ using (B4), and regress g onto the target $-z$ under squared loss:

$$\min_g \mathbb{E}_{t, y_0, z} \|g(y_t, x, t) - (-z)\|_2^2. \quad (\text{B6})$$

This is the standard weighted DSM objective, with weighting $\sigma(t)^2$. Its minimizer satisfies $s = g/\sigma(t) \approx \nabla_y \log p_t(y_t | x)$. We fit one LightGBM model per output dimension. The boosting configuration follows XGB-Point, with `n_estimators=1000`, `max_depth=9`, and `learning_rate=0.05`. We additionally set `num_leaves=127` and `early_stopping_rounds=10`; all other hyperparameters are left at their default values (Table B1).

Sampling. To draw from $p(y | x)$ we integrate the reverse-time SDE [33], which for the VESDE (zero drift, diffusion $g(t) = \sqrt{2\sigma\sigma'}$) is

$$dY = -2\sigma(t)\sigma'(t)s(Y, x, t)dt + \sqrt{2\sigma(t)\sigma'(t)}d\bar{W}, \quad (\text{B7})$$

started at $Y_1 \sim \mathcal{N}(0, \sigma_{\max}^2)$ and integrated from $t = 1$ to $t \approx 0$ with the Euler–Maruyama scheme in S steps. Repeating the integration M times per input yields M conditionally i.i.d. samples, whose empirical mean (or median) is the point prediction, whose standard deviation is the per-cell uncertainty, and whose empirical quantiles give prediction intervals. We use $M = 100$ draws and $S = 50$ integration steps, selected by the ablation in Appendix C.

Characteristics. Treeffuser is the only method in the benchmark that is simultaneously *nonparametric* (it assumes no functional form for $p(y | x)$ and can represent skew, heavy tails, heteroscedasticity, and multimodality), *samplable*, and *tree-based*. The price is inference cost: producing M samples requires $M \times S$ score (LightGBM) evaluations per input, so prediction is substantially slower than the single tree pass of the baselines (Appendix C quantifies the accuracy/compute trade-off).

Representation without a latent space. Unlike neural score models, Treeffuser never maps its inputs into a learned embedding. Two properties follow from its construction. First, the covariates x enter the score model directly as tabular features (Eq. B5): the diffusion time t is appended as an ordinary column and the inputs are never transformed into intermediate activations, so there is no continuous latent vector whose geometry encodes the data. Second, the diffusion process operates directly in the native target space: noise is added to $\Delta p\text{CO}_2$ itself (Eq. B4), rather than to an encoded representation, as in latent diffusion models [34] or approaches that perform generation in alternative representation spaces [35]. What Treeffuser learns is therefore the conditional density $p(y | x)$ itself, carried by the score $\nabla_y \log p_t(y_t | x)$ across noise scales, rather than a geometry of embeddings. The tree ensemble still induces a representation of the inputs, the axis-aligned partition of covariate space defined by its leaves, but this is a discrete partition of physically meaningful variables, not a smooth latent with learned distances, and it is not where the generative process takes place. This keeps the representation interpretable in the original coordinates (each split is a threshold on an observable such as sst or chlorophyll), at the cost of transferability: the learned density is specific to the $\Delta p\text{CO}_2$ task and, unlike a reusable embedding, does not transfer to other targets. Each predictive distribution is conditional on the predictors of a single location-month; the model does not represent dependence between grid cells or months.

B.10 Hyperparameters

Table B1: Treeffuser configuration (LightGBM score model, VESDE, and sampler). Values fixed by the implementation used in this work.

Component	Hyperparameter	Value
LightGBM score model	n_repeats	30
	n_estimators	1000
	early_stopping_rounds	10
	eval_percent	0.1
	num_leaves	127
	max_depth	9
	learning_rate	0.05
	max_bin	255
	subsample_for_bin	200000
	min_child_samples	20
	subsample	1
	subsample_freq	0
	seed	380
VESDE	$\sigma_{\min}, \sigma_{\max}$	0.01, 20
	horizon T	1
Training / sampling	DSM repeats R	30
	number of integration steps S	(see Appendix C)
	number of draws M	(see Appendix C)

Table B2: Hyperparameters for point-prediction models and models providing uncertainty estimates. These settings ensure a fair comparison, as described in Section 3. Values not reported in the table are set to their default values.

Method	Hyperparameter	Value
XGB backbone (Point/Residual/Ensemble/Quantile)	n_estimators	1000
	max_depth	9
	learning_rate (η)	0.05
	seed	380
LightGBM-Point	n_estimators	1000
	max_depth	9
	learning_rate (η)	0.05
	num_leaves	127
ANN-Point	hidden layers	1×210 units
	activation / output	tanh / linear
	optimizer	Adam
	learning rate	10^{-3}
	batch size	512
	L2 penalty (α)	0
	early stopping	30% validation, patience 6
	max epochs	1000
XGB-Quantile	input / target scaling	z-score (training statistics)
	quantile grid Q	<code>np.linspace(0.025, 0.975, 39)</code>
XGB-Ensemble	n_members (B)	100
	per member seed	<code>range(0, B)</code>
NGBoost	n_estimators	1000
	learning_rate	0.05
	distribution	Normal
	score	LogScore

C Treeffuser configuration: draws $\times$ steps ablation

We select 100 draws and 50 integration steps for Treeffuser. CRPS (fair estimator, comparable across draw counts) remains essentially flat across configurations (8.214–8.245). Empirical coverage approaches the nominal levels as the number of draws increases, mainly because more draws resolve the tails of the empirical quantiles; doubling the integration steps to 100 lowers CRPS slightly but does not improve coverage. This sensitivity analysis was run on the test set; since all configurations differ by at most 0.031 μatm in CRPS and 0.028 in coverage, the choice does not affect our conclusions. The selected configuration (100 $\times$ 50) takes 897 s.

Table C1: Treeffuser ablation over predictive draws and integration steps of the underlying SDE. Time is reported in seconds, and the selected configuration in **bold**.

Draws	Steps	CRPS	Cov68	Wid68	Cov95	Wid95	Time (s)
50	50	8.245	0.674	27.045	0.914	52.964	447
100	50	8.229	0.686	27.452	0.930	55.042	897
200	50	8.236	0.692	27.634	0.936	55.945	1804
300	50	8.234	0.693	27.717	0.939	56.367	2689
50	100	8.221	0.670	26.796	0.911	52.405	1005
100	100	8.214	0.681	27.256	0.927	54.649	1990
200	100	8.221	0.688	27.450	0.934	55.579	3959
300	100	8.221	0.690	27.526	0.937	55.993	5992

D Evaluation metrics and how they are computed per method

We evaluate two aspects of each reconstruction: the accuracy of its point prediction and the quality of its predictive distribution. The subtlety is that the benchmarked methods do not expose the same distributional object, some return empirical samples, one returns a closed-form Gaussian, and one returns a quantile function, so each scalar metric must be computed differently for each method. This appendix gives, for every metric, its definition and the recipe used for each predictive object; Table D1 summarizes the recipes. CRPS is a strictly proper scoring rule and the interval score is a proper scoring rule [26]; coverage, interval width, RMSE, and R^2 are complementary diagnostics. All metrics are averaged over the $N=28,879$ test points and reported in μatm , except coverage and R^2 , which are dimensionless.

D.1 Notation and predictive objects

For test point i let y_i denote the observed $\Delta p\text{CO}_2$ and F_i the method’s predictive cumulative distribution function (CDF), with quantile function F_i^{-1} . Depending on the method, F_i is available as one of:

- **Samples** $\{y_i^{(s)}\}_{s=1}^M$ drawn from F_i : Treeffuser ($M = 100$ reverse-SDE draws) and XGB-Ensemble ($M = B = 100$ bootstrap members). The empirical CDF is $\hat{F}_i(z) = \frac{1}{M} \sum_s \mathbf{1}\{y_i^{(s)} \leq z\}$.
- **A parametric Gaussian** $F_i = \mathcal{N}(\mu_i, \sigma_i^2)$: NGBoost, with input-dependent $\sigma_i = \sigma(x_i)$, and XGB-Residual, with $\mu_i = \hat{\mu}(x_i)$ and constant $\sigma_i \equiv \hat{\sigma}$. Here Φ, ϕ denote the standard normal CDF and PDF.
- **A quantile function on a grid** $\{q_{i,\tau}\}_{\tau \in \mathcal{Q}}$ with $q_{i,\tau} = F_i^{-1}(\tau)$: XGB-Quantile (sorted to enforce monotonicity; linearly interpolated between grid levels).

XGB-Point, LightGBM-Point, and ANN-Point return only a conditional mean and hence participate in the point-accuracy metrics alone (Table 1).

D.2 Point-accuracy metrics

Let $\hat{y}_i$ be the method's point summary: the predictive mean for the sample-based and Gaussian methods (Treefuser, XGB-Ensemble, NGBoost, XGB-Residual), and the predictive median $q_{i,0.5}$ for XGB-Quantile.

$$\text{RMSE} = \left(\frac{1}{n} \sum_i (y_i - \hat{y}_i)^2 \right)^{1/2}, \quad R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}. \quad (\text{D8})$$

RMSE is lower-is-better, whereas R^2 , the fraction of variance explained, is higher-is-better. Scores are computed on $\Delta p\text{CO}_2$ in μatm .

D.3 Continuous ranked probability score (CRPS)

CRPS is our headline probabilistic score. For a predictive CDF F and observation y ,

$$\text{CRPS}(F, y) = \int_{-\infty}^{\infty} (F(z) - \mathbf{1}\{z \geq y\})^2 dz = \mathbb{E}_F |X - y| - \frac{1}{2} \mathbb{E}_F |X - X'|, \quad (\text{D9})$$

where $X, X' \sim F$ are independent; the second (energy) form is the basis of the sample estimator. CRPS generalizes the mean absolute error to full distributions and is minimized only by the true conditional law, rewarding calibration and sharpness jointly. It is lower-is-better. We compute it as follows.

Sample-based methods (Treefuser, XGB-Ensemble). Using the M samples $\{y_i^{(s)}\}$,

$$\widehat{\text{CRPS}}(F_i, y_i) = \frac{1}{M} \sum_{s=1}^M |y_i^{(s)} - y_i| - \frac{1}{2M(M-1)} \sum_{s=1}^M \sum_{s'=1}^M |y_i^{(s)} - y_i^{(s')}|, \quad (\text{D10})$$

which is unbiased for any M [36], so scores obtained with different numbers of draws are directly comparable.

Gaussian methods (NGBoost, XGB-Residual; closed form). With $\omega_i = (y_i - \mu_i)/\sigma_i$ [26],

$$\text{CRPS}(\mathcal{N}(\mu_i, \sigma_i^2), y_i) = \sigma_i \left[\omega_i (2\Phi(\omega_i) - 1) + 2\phi(\omega_i) - \frac{1}{\sqrt{\pi}} \right]. \quad (\text{D11})$$

XGB-Quantile (quantile function). Using the quantile (pinball) representation of CRPS, $\text{CRPS}(F, y) = 2 \int_0^1 \Lambda_\tau(F^{-1}(\tau), y) d\tau$ with pinball loss $\Lambda_\tau(q, y) = (\mathbf{1}\{y \leq q\} - \tau)(q - y)$, we integrate numerically over the fitted grid $\mathcal{Q}$,

$$\widehat{\text{CRPS}}(F_i, y_i) = \frac{2}{|\mathcal{Q}|} \sum_{\tau \in \mathcal{Q}} \Lambda_\tau(q_{i,\tau}, y_i), \quad (\text{D12})$$

i.e. twice the mean pinball loss over the 39 equally weighted grid levels, an approximation of the integral over $\tau \in (0, 1)$ that does not extend into the tails beyond 0.025 and 0.975.

D.4 Prediction-interval coverage and width

For a central $(1 - \alpha)$ interval we use the predictive quantiles $\ell_i = F_i^{-1}(\alpha/2)$ and $u_i = F_i^{-1}(1 - \alpha/2)$, and report the prediction-interval coverage probability (PICP) and mean width (MPIW),

$$\text{Cov}_{1-\alpha} = \frac{1}{n} \sum_i \mathbf{1}\{\ell_i \leq y_i \leq u_i\}, \quad \text{Wid}_{1-\alpha} = \frac{1}{n} \sum_i (u_i - \ell_i), \quad (\text{D13})$$

at $\alpha \in \{0.32, 0.05\}$ (the 68% and 95% levels). Coverage is best closest to its nominal level $1 - \alpha$; width (sharpness) is lower-is-better *given* adequate coverage; a narrow interval that under-covers is overconfident, not better. The endpoints (ℓ_i, u_i) are obtained per method as: the empirical $\alpha/2$ and $1 - \alpha/2$ sample quantiles (sample-based methods); $\mu_i \pm z_{1-\alpha/2} \sigma_i$ with $z_{1-\alpha/2} = \Phi^{-1}(1 - \alpha/2)$ (NGBoost and XGB-Residual); the predicted grid quantiles $q_{i,\alpha/2}, q_{i,1-\alpha/2}$ (XGB-Quantile, linearly interpolated at 0.16 and 0.84 for the 68% interval).

D.5 Interval (Winkler) score

To score an interval jointly for width and calibration we use the interval score [26, 37]. For the central $(1 - \alpha)$ interval $[\ell_i, u_i]$,

$$\text{IS}_\alpha(\ell_i, u_i; y_i) = (u_i - \ell_i) + \frac{2}{\alpha}(\ell_i - y_i) \mathbf{1}\{y_i < \ell_i\} + \frac{2}{\alpha}(y_i - u_i) \mathbf{1}\{y_i > u_i\}, \quad (\text{D14})$$

reported at $\alpha = 0.32$ (IntScore68) and $\alpha = 0.05$ (IntScore95), averaged over the test set. It penalizes width plus a coverage penalty of $2/\alpha$ per unit of exceedance for observations that fall outside the interval, and is lower-is-better. It is defined for every method that produces an interval. The endpoints are those of Appendix D.4.

D.6 Summary of metric computation by method

Table D1: How each metric is computed from each method’s predictive object. “samples” denotes the empirical estimator over M draws; “Gaussian” the closed form; “grid” the quantile/pinball route; “–” undefined. $\hat{y}$ is the point summary used for the accuracy metrics.

Method	Predictive object	$\hat{y}$	Interval $[\ell, u]$	CRPS
XGB-Point	mean only	mean	–	–
LightGBM-Point	mean only	mean	–	–
ANN-Point	mean only	mean	–	–
XGB-Quantile	quantile grid	median	grid quantiles	grid
XGB-Residual	Gaussian $(\hat{\mu}, \hat{\sigma})$	$\hat{\mu}$	$\hat{\mu} \pm z\hat{\sigma}$	Gaussian
XGB-Ensemble	members $\{\hat{\mu}_b\}$	member mean	member quantiles	samples
NGBoost	Gaussian (μ, σ)	μ	$\mu \pm z\sigma$	Gaussian
Treeffuser	samples $\{y^{(s)}\}$	sample mean	sample quantiles	samples