Cosmic Ray Flux Reconstruction using Machine Learning

Carter Chapman, Andrei Gogosha
Phys/Astr 448

December 6, 2025

Abstract

We analyze data from the Pierre Auger Observatory to identify which atmospheric variables most affect cosmic-ray count rates. Using several years of temperature, pressure, and air density measurements, we train a linear `SGDRegressor` and a nonlinear `HistGradientBoostingRegressor`, optimized via cross-validation. The `SGDRegressor` achieves $\mathbf{7.87 \pm 0.01}$ MAE, while the gradient boosting model improves accuracy to $\mathbf{6.59 \pm 0.03}$ MAE. SHAP-based feature importance reveals the dominant atmospheric drivers. Our results demonstrate that machine-learning methods, particularly gradient boosting, effectively model the atmospheric modulation of cosmic-ray flux.

1 Introduction

High-energy particles enter Earth's atmosphere and interact to produce secondary particles, including high-energy muons and nucleons. Gamma-ray telescopes and particle detectors around the world record these interactions daily. If an experiment is searching for a specific interaction, the flux and nature of these secondary particles must be well understood. Temperature, pressure, and air density are weather properties known to influence cosmic ray showers [11] and can lead to unexpected variations in measured flux. Machine learning provides an ideal way to study this relationship due to its ability to handle large datasets with many interdependent variables. The boosted model used in this work builds an ensemble of decision trees that iteratively improve upon previous predictions by correcting regions of poor fit. The SGD Regressor is a linear model, making it a useful baseline for comparison against the more complex ensemble methods. Together, these models allow us to evaluate how different learning strategies capture the relationship between atmospheric variables and cosmic ray detection rates. The Pierre Auger Observatory [2] is a cosmic ray observatory located in Argentina designed to detect high-energy cosmic rays. It does this through two main methods: tracking particle interactions in large water-Cherenkov detectors on the surface and observing ultraviolet radiation from air showers in the upper atmosphere [2]. The observatory provides an open dataset containing 81,121 showers detected via the ultraviolet fluorescence method, along with complementary data from the surface detectors and atmospheric measurements from five meteorological stations in the region. These datasets are available in JSON and CSV formats and include information on particle hits for detectors and weather features such as temperature, pressure, and density. To our knowledge, no published studies have yet used machine learning to predict atmospheric particle flux based on weather variables, although the effects of weather on cosmogenic particles have been explored analytically [8][9]. Our approach aims to more accurately model cosmic ray flux while incorporating environmental variables that are typically neglected in current analyses [3].

2 Methods

2.1 Data preprocessing

The Pierre Auger Open Release data [4] provides the data for this search. The scaler data contains 15-minute averaged particle event rates uncorrected for atmospheric pressure dependence, fraction of the array in operation, and Unix timestamps. The atmospheric data contains temperature, barometric pressure, humidity-corrected air density, and Unix timestamps taken every five or ten minutes with gaps of ten minutes to three hours interpolated from the endpoints of the interval. If the gap was longer than three hours, the period was not included. To account for the differing rates of atmospheric data, cosmic ray data, and missing periods of atmospheric data, we time-aligned the scaler and atmospheric data using timestamps. Due to changes made in the Pierre Auger Data Taking strategy, there are two distinct populations in the count rate distributions shown below 1.

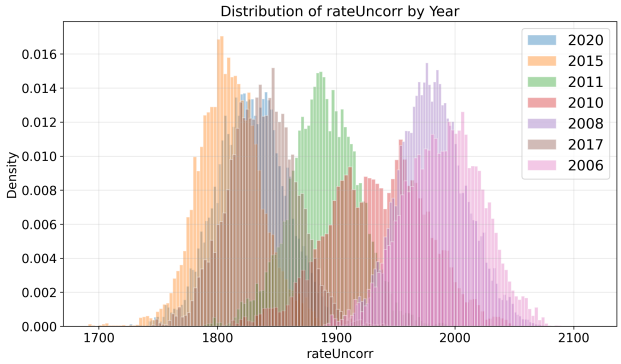


Figure 1: Particle Count Rate by Year

The two populations come from years after 2011 and years before 2010. The exact reason for this is unclear; however, changes made to the trigger scheme [1] in the past offer a likely explanation for this behavior. For this reason, all our models used only data that came after 2012. This also led us to use the 'year' as a one-hot encoded feature using the `OneHotEncoder` preprocessor from `scikit-learn`.

The final list of features used for training our models is: Temperature, Pressure, Air Density, Array Fraction, Density 2 Hours Prior, and `OneHotEncoder` Year.

2.2 Machine Learning Models

All trained models used a nested cross-validation scheme of five outer and three inner shuffled folds when performing hyperparameter optimization. The mean absolute error (MAE) was used as the scoring metric for training and performance evaluation.

2.2.1 SGDRegressor

For modeling the linear relationship between atmospheric conditions and cosmic-ray flux, we trained an `SGDRegressor` from `scikit-learn` [10]. Hyperparameter optimization was performed on which loss function the regressor optimizes, the regularization term `penalty` and its weighting factor `alpha`, the elastic net mixing parameter `l1_ratio`, and the threshold value where the Huber loss changes from quadratic to linear `epsilon`. When training data was scaled using the `StandardScaler` function from `scikit-learn`. The hyperparameter values optimized are summarized in Table 1

Table 1: Hyperparameter values optimized for the `SGDRegressor`.

`loss`	squared_error	huber		
`penalty`	l1	l2	elasticnet	None
`alpha`	0.00001	0.0001	0.001	
`l1_ratio`	0.1	0.5	1	
`epsilon`	0.1	0.5	1	5

2.2.2 HistGradientBoostingRegressor

For modeling the non-linear relationships between atmospheric conditions and cosmic-ray flux, we trained a `HistGradientBoostingRegressor` Decision Tree from `scikit-learn` [10]. We explored a grid of hyperparameters as summarized in Table 2. The maximum tree depth (`max_depth`) controls the complexity of each individual tree, while the number of boosting iterations (`max_iter`) determines how many trees are sequentially added to the ensemble. The learning rate (`learning_rate`) scales the contribution of each tree, and the loss function (`loss`) specifies whether the model minimizes squared error or absolute error. Finally, the minimum number of samples per leaf (`min_samples_leaf`) sets the smallest number of training examples required to form a leaf, helping to prevent overfitting.

Table 2: Hyperparameter values optimized for the `HistGradientBoostingRegressor`.

`max_depth`	3	6	10	None
`max_iter`	50	100		
`learning_rate`	0.05	0.1		
`loss`	squared_error	absolute_error		
`min_samples_leaf`	5	10	15	

2.3 SHAP Feature Importance

To interpret the contributions of individual atmospheric variables to the predicted cosmic-ray count rates, we used SHapley Additive exPlanations (SHAP) [6]. SHAP is a model-agnostic framework that assigns each feature an importance value for a particular prediction, based on cooperative game theory. For the `HistGradientBoostingRegressor`, we computed SHAP values using the TreeExplainer[7], which leverages the structure of gradient-boosted trees for an efficient and exact computation. For the `SGDRegressor`, which is a linear model, we used LinearExplainer, which computes SHAP values analytically based on the model coefficients, allowing us to directly assess the contribution of each feature to the predicted count rate. For each feature, we examined both the mean absolute SHAP value across the test set, which quantifies overall importance, and the distribution of SHAP values, which reveals whether higher or lower feature values increase the predicted count rate. This analysis allowed us to identify which atmospheric variables, such as air pressure, temperature, or density, modulate the predicted particle flux the most, providing information on the physical drivers of the observed variations.

3 Results

Since the performance of both the `SGDRegressor` and `HistGradientBoostingRegressor` models with only the atmospheric data showed worse performance than with the `OneHotEncoder` year feature analogous plots to those found in Figure 2 and Figure 3 and have been omitted for brevity, but can be found in the training notebooks [5].

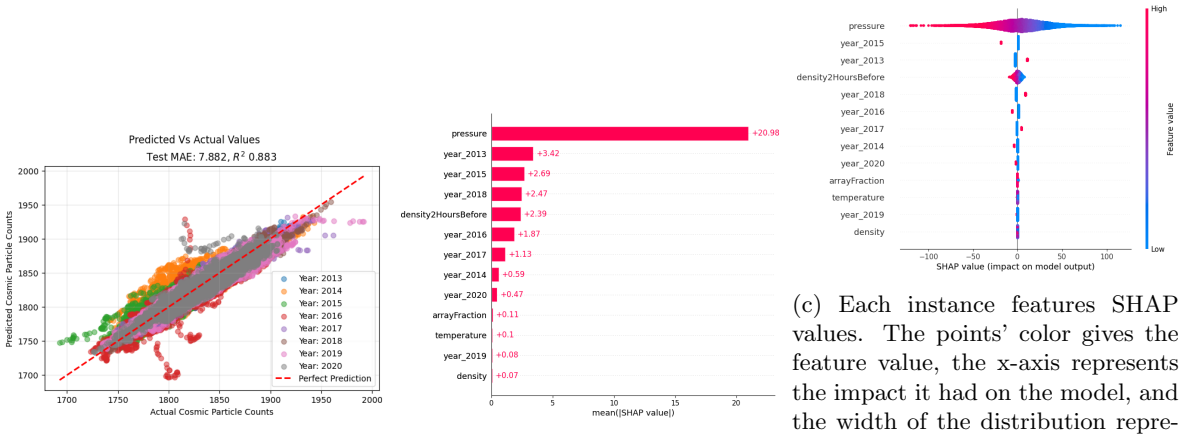
3.1 SGDRegressor

When using temperature, pressure, air density, density 2 hours before, and array fraction as our training features, the `SGDRegressor` model achieved an MAE of 10.88 ± 0.01868 and 7.8748 ± 0.01025 when including the `OneHotEncoder` year feature. The hyperparameters that perform the best are in Table 3. The results

Table 3: Best Performing Hyperparameter values for the `SGDRegressor` model.

	Atmospheric Data	`OneHotEncoder` Year
`loss`	huber	huber
`penalty`	elasticnet	l1
`alpha`	0.0001	0.0001
`l1_ratio`	0.5	0.1
`epsilon`	1	1

for the model trained with `OneHotEncoded` features are in Fig 2.



(a) Predicted vs Actual Cosmic Ray Flux grouped by Year

(b) mean (|SHAP|) value for each feature

(c) Each instance features SHAP values. The points' color gives the feature value, the x-axis represents the impact it had on the model, and the width of the distribution represents the density of points at that SHAP value.

Figure 2: `OneHotEncoder` Year Feature Performance and Feature Importance plots

Plot (2a) shows the models' predicted vs actual values. The data points are also colored by year to provide more insight into problematic years. An R^2 value of 0.883 means that the model accounts for most of the variance in the data, but still has some large outliers unexplained by the model. As you can see in plot (2a), 2016 has some large outliers which are likely the largest contributors to the bias in our model. This suggests that either our features are not providing enough information for our model to fully capture the relationship between our features or the relationship between our features is nonlinear. The results in section 3.2 suggest that the relationship is non-linear due to the better performance of the `HistGradientBoostingRegressor` model. Plot (2b) gives the average SHAP value for each of the features. It shows that pressure contributes the most to the model's output; on average, it changes the predicted count rate by 20 $\frac{counts}{s}$. This plot also shows that, contrary to our proposal's expectation, density is not the most important feature; it is the least important. Plot (2c) shows the distribution of SHAP values by feature, and provides more insight into the maximum and minimum impact features have on the predictions.

3.2 HistGradientBoostingRegressor

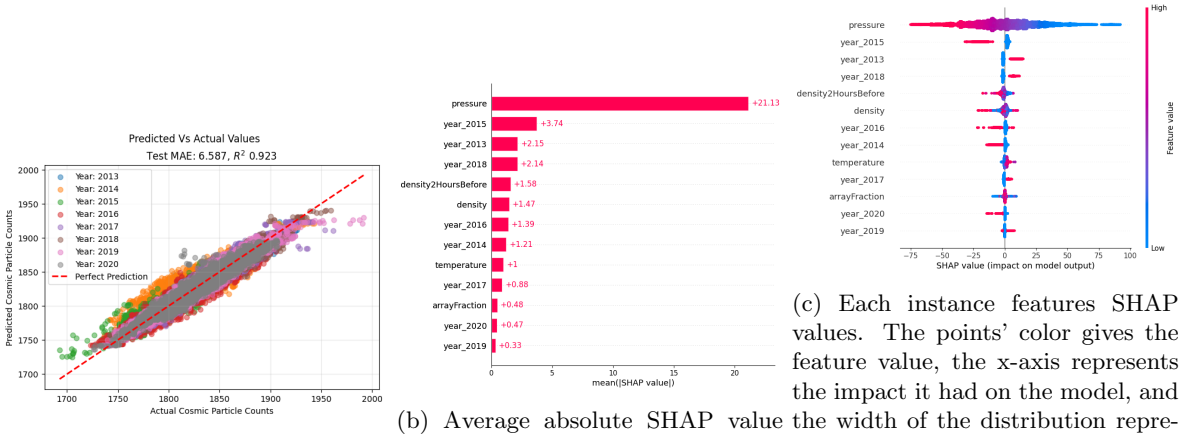
When using temperature, pressure, air density, density 2 hours before, and array fraction as our training features, the `HistGradientBoostingRegressor` model achieved an MAE of 10.1793 ± 0.02920 and an MAE

of 6.5860 ± 0.02472 when including year as a feature. The hyperparameters that perform the best are in Table 4

Table 4: Best Performing Hyperparameter values for the `HistGradientBoostingRegressor` model.

	Atmospheric Data	`OneHotEncoder` Year
`max_depth`	None	None
`max_iter`	100	100
`learning rate`	0.1	0.01
`loss`	absolute_error	squared_error
`min_samples_leaf`	10	10

The results with the `OneHotEncoded` features and the boosting method are given below 3



(a) Predicted vs Actual Cosmic Ray Flux grouped by Year

(b) Average absolute SHAP value for each of the five atmospheric features

(c) Each instance features SHAP values. The points' color gives the feature value, the x-axis represents the impact it had on the model, and the width of the distribution represents the density of points at that SHAP value.

Figure 3: OneHotEncoded Feature Performance and Feature Importance plots

Plot 3a shows the models' predicted vs actual values. Compared to 2a, the boosted model is able to capture non-linear relationships for problematic years such as 2015, and 2016. Other years that still remain difficult for the model to train is 2019. This could come from a change in a triggering scheme or an increase in cosmic rays due to other physics processes. An R^2 value of 0.923 means the model accounts for the vast majority of the variance in the data. Plot (3b) gives the average SHAP value for each of the features. It confirms that pressure is the most important feature by a wide margin, as the pressure value on average changes the predicted count rate by 21 $\frac{counts}{s}$. It is also shown that the year is the second most important feature behind pressure. Density also plays a much more significant role for the boosted model than it does for `SGDRegressor`. Plot (3c) shows how the boosted model treats the `OneHotEncoded` Year features differently as the SHAP values have larger ranges of impact on the count rate.

4 Conclusion

Both the `SGDRegressor` and the `HistGradientBoostingRegressor` accurately predict particle flux from atmospheric data, with the boosted model achieving superior performance due to its ability to capture non-linear relationships. It also trains more efficiently, reaching better accuracy in significantly less time. Initial exploration suggested that the year of data collection influenced the model, a result confirmed through SHAP analysis [6]. SHAP further identified pressure as the most impactful feature in predicting flux variations,

counter to our proposed hypothesis of density being the most important feature. These results demonstrate that machine-learning methods can effectively characterize atmospheric modulation of cosmic-ray flux. Future work should extend these techniques to broader geographic settings and explore their potential for predicting additional cosmic-ray properties, such as energy distributions or particle interactions.

A GitHub repository containing our training notebooks, data, preprocessing scripts, and documentation is available in the references [5].

References

[1] P. Abreu et al. "The Pierre Auger Observatory scaler mode for the study of solar activity modulation of galactic cosmic rays". In: *Journal of Instrumentation* 6.01 (2011), P01003. DOI: 10.1088/1748-0221/6/01/P01003.

[2] Beatrix. *Pierre Auger Observatory*. Pierre Auger Observatory. Feb. 18, 2021. URL: https://www.auger.org/ (visited on 10/18/2025).

[3] Alan Coleman and the Pierre Auger Collaboration. "The influence of weather effects on the reconstruction of extensive air showers at the Pierre Auger Observatory". In: *35th International Cosmic Ray Conference (ICRC 2017)*. Vol. ICRC2017. PoS. arXiv:1708.06507 [astro-ph.IM]. 2017, p. 326. URL: https://inspirehep.net/files/1ef8bb507e47d8bc4ac3b456b7ff55ef.

[4] The Pierre Auger Collaboration. *Pierre Auger Observatory Open Data*. Version 3. Zenodo, 2024. DOI: 10.5281/zenodo.10488964. URL: https://doi.org/10.5281/zenodo.10488964.

[5] Andrei Gogosha and Carter Chapman. *UNC-MLPhysAstr Final Project: ML Cosmic Ray Flux Reconstruction*. URL: https://github.com/AGogosha/MLPhysAstr-CosRay-ReFlux.

[6] Scott M Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.

[7] Scott M. Lundberg et al. "From local explanations to global understanding with explainable AI for trees". In: *Nature Machine Intelligence* 2.1 (2020), pp. 2522–5839.

[8] William Luszczak and Man-Yau Chan. *Using cosmic rays to predict the weather: Meteorological data assimilation of atmospheric muon flux data*. 2025. arXiv: 2509.04627. URL: https://arxiv.org/abs/2509.04627.

[9] A. H. Maghrabi, S. A. Alharbi, and R. M. Alhawsawi. "The Role of Atmospheric Pressure, Temperature, and Humidity on Cosmic Ray Muons at a Low Latitude Station". In: *International Journal of Astronomy and Astrophysics* 13.3 (2023), pp. 137–150. DOI: 10.4236/ijaa.2023.133009. URL: https://www.scirp.org/pdf/ijaa_2023091314361700.pdf.

[10] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[11] Henrik Svensmark. "Influence of Cosmic Rays on Earth's Climate". In: *Phys. Rev. Lett.* 81 (22 Nov. 1998), pp. 5027–5030. DOI: 10.1103/PhysRevLett.81.5027. URL: https://link.aps.org/doi/10.1103/PhysRevLett.81.5027.