

CO₂ classification analysis for AI models

Juan Felipe Gallo Rendón* *Engineering department*
University of Antioquia
 Medellín, Colombia

Abstract—

*Index Terms—*Cluster analysis, K-means, Classification models,

I. INTRODUCTION

II. METHODOLOGY

Classification methods are applied to the selected dataset. A workflow comprising *six* procedures is followed: **univariate analysis**, **cluster analysis**, **k-means clustering**, **linear classification**, and **nonlinear classification**. Upon completion of these steps, the results are compared and the best-performing model is identified.

A. Univariate Analysis

In this stage, the dataset **HFCO2.csv**—originating from a CO₂-emissions report on the Hugging Face platform—is profiled to characterize marginal distributions, identify outliers, and screen variables for subsequent modeling. The variables under study are enumerated below. Variables such as `modelId`, `datasets`, `co2_reported`, `createdat`, and `libraryname` are excluded due to limited analytical relevance; `database_efficiency` is removed because it is functionally dependent on `co2_eq_emissions` and `size`. Because no native target is present, classification labels are defined to enable supervised evaluation. First, `model_type` is created to align the `performance_score` with the appropriate metric family (e.g., accuracy, F1, Rouge); rows lacking performance metrics are discarded. Univariate summaries and frequency counts are then computed for qualitative variables (e.g., `model_type`) and proportions are reported for domain; variables exhibiting near-constant distributions (e.g., `library_name = pytorch`), or extreme sparsity (`geographical_location`, `environment`) are removed. Finally, a fairness-oriented label is introduced to relate predictive performance to environmental impact, yielding four categories—*Fair and Efficient*, *Powerful but Expensive*, *Green but Weak*, and *Inefficient*—and a derived Boolean variable `is_fair`. The resulting curated dataset and label structure provide the basis for the next methodological step, cluster analysis.

B. Cluster analysis

A hierarchical clustering analysis was applied to segment AI models using numeric performance-and-cost variables. Records with N/A/∞ were removed and features were standardized to z-scores, to mitigate dominance of high skewed counts in features like `downloads` and `likes`,

simple transformation (log/winsorizing) were considered. Two distances were evaluated: Euclidean and Mahalanobis, the latter computed with a Ledoit-Wold covariance estimator. Four linkage rules (Ward, average, complete and single) were compared. The dendrogram was cut either at a target k ($k \in \{3, 4\}$) to improve size balance or at the k that maximized the average silhouette; the equivalent distance threshold reproducing that partition was also recorded. Internal quality was assessed with the silhouette score, while cophenetic correlation was used descriptively to gauge dendrogram fidelity. As external validation, labels (`is_fair`, `fairness_class`, `model_type`) were not used for training and were contrasted post-hoc via contingency tables, χ^2 /Crammer’s V, and adjusted Rand index. Cluster interpretation relied on profiles in original units and on z-centroids; a `lzl` ranking highlighted the most discriminative variables. A comparative summary across configurations (metrics, thresholds, balance) and visualizations (dendograms with horizontal cut, stacked proportions and heatmaps) were produced for review. Final selection prioritized overall separability and operational usefulness, balancing silhouette with stability, cluster-size balance and consistency with external labels.

K-means analysis: K-means was applied to the same numeric variables used in the hierarchical analysis (`performance_score`, `co2_eq_emissions`, `likes`, `downloads`, `size`). Rows with non-numeric/inf values were removed and features were standardized to z-scores (`StandardScaler`). We explored $k \in \{2, \dots, 10\}$ with k-means++ initialization (`n_init = 50`, `max_iter=500`, `random_state=42`).

For each k we reported: Silhouette (higher is better), Calinski–Harabasz (CH; higher is better), Davies–Bouldin (DBI; lower is better), and the inertia (*elbow curve*). The primary selector was the maximum Silhouette; CH/DBI and the elbow were used as secondary evidence.

For the selected k we exported cluster sizes, original-unit profiles (count/mean/median), z-centroids, and a variable ranking using `max|z|` (to highlight the most discriminative features). External labels (`is_fair`, `classification_fairness`, `model_type`) were held out from training and only used for validation via contingency tables, χ^2 /Cramér’s V, Fisher’s exact test when 2×2 , and ARI when applicable.

C. Multivariate Normality & LDA Feasibility — Methods

Check whether the feature vector `X=performance_score, co2_eq_emissions, likes, downloads, size` is (approximately) multivariate normal—globally and within classes—so that classical LDA is justified.

Preprocessing. Remove rows with NA/inf; standardize all variables to z-scores.

Robust scatter & distances. Fit a shrinkage (Ledoit–Wolf) covariance $\hat{\Sigma}$ to mitigate outliers. Compute squared Mahalanobis distances $d_i^2 = (x_i - \hat{\mu})^\top \hat{\Sigma}^{-1} (x_i - \hat{\mu})$ (i) globally and (ii) per class.

Normality diagnostic. Under MVN with p variables, $d_i^2 \sim \chi_p^2$. Compare empirical quantiles of $\{d_i^2\}$ against χ_p^2 using QQ-plots: global and class-conditional. Large, systematic upward deviations (heavy tails/mixtures) indicate non-normality.

Optional formal checks. (a) Mardia or Henze–Zirkler tests on z -scores; (b) Box’s M (or visual comparison) for equality of class covariances.

Decision rule. nosep,leftmargin=*

- 1) *Proceed with classical LDA* if QQ-plots track the χ_p^2 line reasonably well (globally *and* by class) and class covariances appear similar.
- 2) *Prefer robust/regularized LDA or nonparametric classifiers* (e.g., shrinkage LDA, logistic regression, tree-based) if tails inflate, classes differ in scatter, or tests reject MVN/homoscedasticity.

D. LDA

Multivariate normality was examined via Mahalanobis Q–Q diagnostics (globally and by class). Because clear departures from normality were observed, regularized discriminants were preferred. Linear Discriminant Analysis (LDA) with automatic shrinkage (and, for projections, the `eigen` solver) and Quadratic Discriminant Analysis (QDA) with mild regularization were fit on a stratified train/test split. Performance was summarized with confusion matrices, precision/recall/F1, ROC–AUC, and 5-fold stratified cross-validation. Model parameters (priors, class means, coefficients, scalings), LD scores, and plots (ROC, score histograms, dendrograms) were exported to files to enable reporting and reproducibility.

E. Non-linear models comparison

A supervised classification workflow was implemented to benchmark a linear baseline (Linear Discriminant Analysis, LDA) against two non-linear alternatives: an RBF-kernel Support Vector Machine (SVM–RBF) and Gradient Boosting (GB). The target was the binary label `is_fair`; predictors were the numeric performance–cost variables used in previous sections. Missing values were removed, then features were transformed with $\log(1 + x)$ (for skewed counts: CO₂, likes, downloads, size) and standardized (zero mean, unit variance). A stratified train/test split (70/30) preserved the empirical class imbalance.

Models were trained in `scikit-learn` as follows. LDA used the `eigen` solver with automatic shrinkage, providing a robust shared covariance estimate and interpretable discriminant directions. SVM–RBF used probability calibration (`probability=True`) and an RBF kernel; GB used logistic loss with modest depth and learning rate (standard defaults), both selected via stratified 5-fold cross-validation (CV) on the training fold. Evaluation focused on rank-based and decision-based criteria: AUC(ROC), average precision (AP) for the positive class (*True*), F1 for the positive class, overall accuracy, and Brier score (probability calibration). Error structure

was inspected through confusion matrices. For comparability, models were scored on the same held-out test set; CV accuracy was also reported to assess stability across resamples. Curves (ROC and Precision–Recall) and panels (confusion matrices) were produced for visual diagnostics and threshold selection.

III. RESULTS AND DISCUSSIONS

LOREMPIPSUM

A. Univariate Analysis

The database **HFCO2.csv** is a product of the CO₂ emissions report generated on the Hugging Face platform. In this case, the analysis is focused on the following variables:

- 1) `co2_eq_emissions`: the resulting carbon footprint
- 2) `downloads`: number of model downloads
- 3) `likes`: number of model likes
- 4) `performance_metrics`: (accuracy, F1, Rouge-1, Rouge-L)
- 5) `performance_score`: the harmonic mean of the normalized performance metrics
- 6) `size`: size of the final trained model in MB

Variables like `modelId`, `datasets`, `co2_reported`, `createdat`, and `libraryname` were removed because of their lack of importance in the analysis. `database_efficiency` was also removed because it is a dependent variable of `co2_eq_emissions` and `size`. The first step was to choose metrics for classification, as the database *per se* does not have a relevant dependent variable to be classified. Therefore, some labels were created with the aim of creating a dependent variable to be predicted by a couple of machine learning models. Accordingly, `model_type` is the label used to determine which metric the model uses for its performance score, because not all models use the same metric for evaluation. After applying this classification, some rows were unusable because hundreds of them do not have performance metrics at all, so they were deleted. After the redundant rows were erased, a count was made on the qualitative variables, as shown in the following table:

TABLE I: model_type counts

model type	count
type1 (accuracy & f1)	773
type2 (rouge)	228
type3 (accuracy)	72
type4 (rouge1)	3
type5 (f1)	2

In an effort to find relationships between model types and the other categories, a summary of the other variables was prepared, focusing on the domain variable, which is the model’s use case. The following table shows each domain and the associated percentage:

TABLE II: percentage of domains

model type	count
NLP	81,5%
Computer Vision	16,0%
Not specified	2,5%

In conclusion, there are only two major types of domain models. For the purpose of this analysis, this is not an influential variable, so it was removed. Other variables, like `library_name`, revealed that all models use `pytorch` as a library, so this was also removed. The same reasoning applied to `geographical_location`: the analysis revealed that 99% of the models do not specify the training location. Similarly, 99.7% of the models do not report the hardware used in `environment`. After applying analysis upon the mean of performance metrics and CO₂ emissions, the analysis revealed that those models could be labeled again with a more precise metric: *fairness*, defined as a relation between performance score and CO₂ emissions. The following are the labels chosen for the models

- 1) Fair and Efficient
- 2) Powerful but Expensive
- 3) Green but Weak
- 4) Inefficient

Finally, models were classified by a new boolean variable "is_fair", and those that were previously labeled with respective performance metrics, were evaluated: Results are consigned in the route:

TABLE III: Model Classification based on their fairness and performance metrics

Model Type	Classification Fairness	Count
type1 (accuracy & f1)	Fair and Efficient	229
	Powerful but Expensive	214
	Green but Weak	181
	Inefficient	142
	Incomplete data	7
type2 (rouge)	Inefficient	140
	Green but Weak	57
	Powerful but Expensive	18
	Fair and Efficient	9
	Incomplete data	4
type3 (accuracy)	Fair and Efficient	50
	Powerful but Expensive	9
	Green but Weak	7
	Incomplete data	4
	Inefficient	2
type4 (f1)	Incomplete data	2
type5 (rouge1)	Incomplete data	3

/notebooks/part2/1-univarated-analysis.ipynb
Those classifications are required for the next step in the methodology, Cluster analysis.

B. Cluster Analysis

A grid over *distance* \times *linkage* found that the highest internal separation was achieved by **Single-Euclidean** with

$k = 3$ (silhouette ≈ 0.62 , cophenetic ≈ 0.71). Close contenders were *Complete-Mahalanobis* ($k = 3$, silhouette ≈ 0.52 , cophenetic ≈ 0.71) and *Average-Mahalanobis* ($k = 3$, silhouette ≈ 0.43 , cophenetic ≈ 0.81). Among the “non-single” options, **Average-Euclidean** with $k = 4$ offered a more conservative structure (silhouette ≈ 0.34 , cophenetic ≈ 0.81). *Ward-Euclidean* showed the lowest separation (silhouette ≈ 0.20 , cophenetic ≈ 0.44).

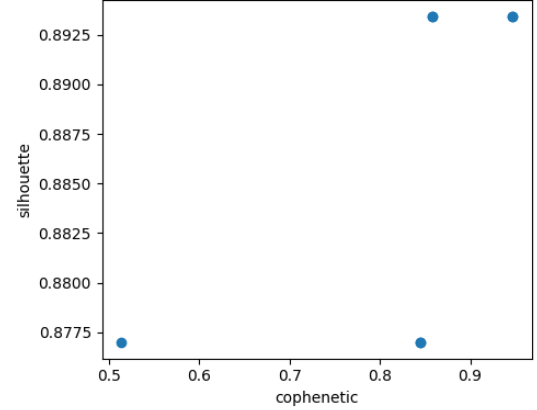


Fig. 1: Cophenetic vs. silhouette for all configurations. Top-right indicates better fidelity and separation.

Because *single* linkage may form chaining clusters, we report two complementary views: (i) the best-silhouette solution (**Single-Euclid**, $k = 3$) and (ii) a robust alternative without single linkage (**Average-Euclid**, $k = 4$). The horizontal cut at the recorded equivalent threshold reproduces the same k in the dendrogram.

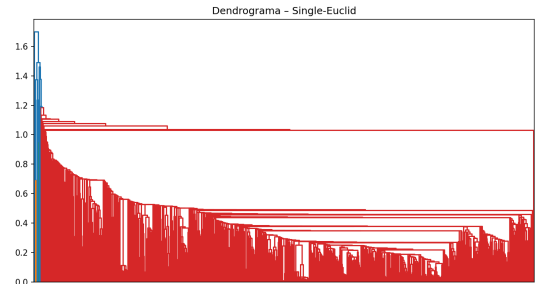


Fig. 2: Dendrogram—Single-Euclidean. The horizontal cut at the equivalent threshold yields $k = 3$.

Across configurations, a *dominant* “catalog” cluster concentrates the bulk of models near global means, while 1–3 *small* “elite” clusters group items with markedly higher *downloads/likes/performance* and (on average) lower *size/CO₂*. Variable importance from z -centroids is consistent: the most discriminative feature is **downloads** (max $|z| \approx 5.39$), followed by **likes** (≈ 2.03), then *CO₂* and *performance* (moderate), with *size* contributing less (≈ 1.32).

External validation against labels was limited, as expected (labels were not used to train the clustering): ARI vs. `is_fair` stayed near zero across settings; however, cluster-label contingency showed reasonable purities

(mean ~ 0.73 – 0.90 depending on the configuration), with `is_fair=True` enriched in one of the small clusters and scarce in the large catalog group.

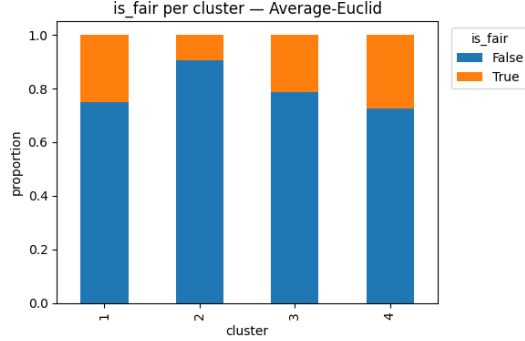


Fig. 3: Stacked proportions of `is_fair` per cluster (Average-Euclid, $k = 4$).

a) Cluster profiles.: Tables IV–VI report *original-unit* profiles (count/mean/median) for *performance_score*, *co2_eq_emissions*, *likes*, *downloads* and *size*. For brevity, we include the robust alternative (Average-Euclid, $k = 4$); the z -centroids and the $|z|$ ranking (*downloads* \rightarrow *likes*) appear in the Supplement.

TABLE IV: Cluster profiles (original units): performance and CO_2

cluster	n	performance_score		co2_eq_emissions	
		mean	median	mean	median
1	4	0.988	0.991	49.974	3.189
2	1 029	0.727	0.822	71.382	2.749

TABLE V: Cluster profiles (original units): likes and downloads

cluster	n	likes		downloads	
		mean	median	mean	median
1	4	2.250	0.500	104 439.25	102 572.50
2	1 029	0.181	0.000	65.245	5.00

TABLE VI: Cluster profiles (original units): size ($\times 10^8$)

cluster	n	mean	median
1	4	2.649	2.654
2	1 029	8.953	4.987

b) Overall quality.: Across linkages and distances, the highest internal separation was obtained by *Single-Euclid* with $k = 3$ (silhouette ≈ 0.62 , cophenetic ≈ 0.71), followed by *Complete-Mahalanobis* (silhouette ≈ 0.52). Average-based variants trailed behind (silhouette ≈ 0.30 – 0.44) and *Ward-Euclid* showed the lowest separation (silhouette ≈ 0.20).

c) Cluster profiles (what differentiates groups).: Tables IV–V report original-unit profiles for a robust reference solution. Consistent with the winning configuration, a small cluster concentrates very high *downloads/likes* (and comparatively lower *size/CO₂*), while a large “catalog” cluster sits near the global mean. In z -space, *downloads* shows the largest between-cluster contrast ($\max |z| \approx 5.39$), followed by *likes* (≈ 2.03); CO_2 , performance and size contribute more modestly (≈ 1.3 – 1.7).

d) External validation (not used for training).: Contingency tables indicate a small-to-moderate association between clusters and external labels. For *Single-Euclid* ($k = 3$), the largest cluster concentrates the majority of `is_fair=False` (share of *True* ≈ 0.27), whereas minor clusters show either higher *True* share or are too small to be conclusive. (mean/weighted), χ^2 with Cramér’s V (≈ 0.03 – 0.07 across configs), and Fisher’s exact test when applicable, together with ARI vs. `is_fair` (near zero, as expected for weak alignment). Figure 3 visualizes the `is_fair` proportions per cluster.

e) Sensitivity and limitations.: The pattern is robust across $k \in [2, 6]$ and across distance/linkage families that avoid overly compact (*ward*) or overly chained (*single*) artifacts; nonetheless, cluster sizes remain unbalanced and the *single* linkage can induce chaining in dense regions. These caveats do not affect the substantive finding that *downloads* (and, secondarily, *likes*) drive the segmentation.

C. Univariate Analysis

We ran k-means on the standardized variables (*performance_score*, *co2_eq_emissions*, *likes*, *downloads*, *size*) for $k \in \{2, \dots, 10\}$ with k-means++, $n_{\text{init}}=50$, $\text{max_iter}=500$. Model selection combined four internal criteria:

- **Silhouette (max is best).** The curve peaked at $k = 2$ with ≈ 0.286 , and decreased thereafter (Fig. 4).
- **Calinski–Harabasz (max is best).** The maximum occurred at $k=3$ (≈ 330), with $k=2$ close behind (≈ 290) (Fig. 4).
- **Davies–Bouldin (min is best).** DBI decreased monotonically with k , reaching ≈ 1.21 at $k=10$ (Fig. 5).
- **Elbow (inertia).** Inertia dropped rapidly up to $k \sim 6$ and then flattened, with no sharp elbow afterwards (Fig. 5).

Given the primary criterion (silhouette), we selected $k = 2$. The resulting partition produced **unbalanced cluster sizes** (about $n \sim 740$ vs. $n \sim 300$; Fig. 6). Cluster profiles (in original units) and z -centroids were computed; the ranking by $\max |z|$ again highlighted *downloads* as the most discriminative variable, followed by *likes*, with the remaining variables contributing less. Cross-tabs against the external labels (`is_fair`, `clasification_fairness`, `model_type`) were generated for $k=2$ (see Supplement for the full tables).

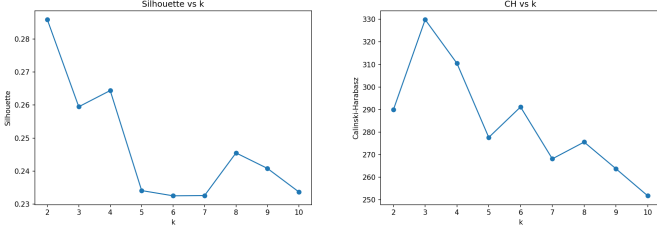


Fig. 4: Silhouette and Calinski–Harabasz across k .

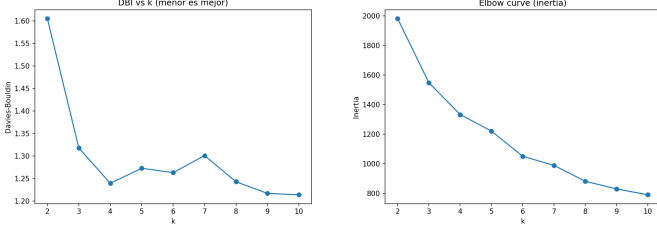


Fig. 5: Davies–Bouldin (lower is better) and inertia (elbow) across k .

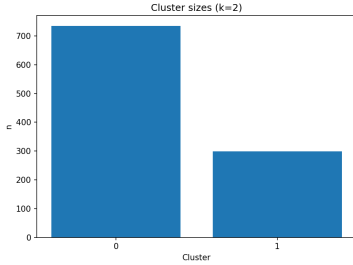


Fig. 6: Cluster sizes for the selected solution ($k=2$).

The internal criteria point to different trade-offs: silhouette clearly favors $k=2$, Calinski–Harabasz prefers $k=3$, and Davies–Bouldin keeps improving as k grows. In this context, we prioritized silhouette because it balances compactness and separation and is less biased by the growth of k than CH/DBI. The *elbow* curve shows diminishing returns beyond $k \sim 6$, supporting the choice of a small number of groups.

The $k=2$ solution is *interpretable* and *stable* (across restarts) but *unbalanced*: one major segment (catalog-like) and a smaller, high-intensity segment. The profile tables indicate that the minor cluster concentrates substantially higher *downloads* and *likes* (and slightly higher *performance*), while medians for *size/CO₂* are not dominant—consistent with the hierarchical analysis. The $\max |z|$ ranking confirms *downloads* (then *likes*) as the strongest separators, which aligns with the business intuition that usage/engagement variables drive the segmentation more than footprint or size.

Regarding *external* labels, the cross-tabs for $k=2$ show heterogeneous mixes rather than perfectly pure clusters, i.e., they are useful for characterization but should not be read as supervised classes. This is expected because those labels were not used for training. Overall, the k-means segmentation complements the hierarchical results: it recovers the same two-segment story (catalog vs. high-traction niche), is easier

to deploy, and preserves the same ordering of discriminative variables. If more granularity is needed, $k=3$ is a reasonable alternative per CH, though with a slight loss in silhouette and added complexity.

D. Multivariate Normality & LDA Feasibility

Visual diagnostics. Mahalanobis QQ-plots (global and by class) depart markedly from the χ_p^2 reference line in the upper tail. The global plot shows a pronounced upward bend, indicating heavy-tailed behavior and/or mixture structure. Class-conditional plots (for `is_fair=False/True`) display the same pattern rather than aligning with the diagonal, so non-normality is not restricted to a single group.

Likely drivers. The heaviest deviations coincide with variables that are naturally skewed and highly dispersed in this domain (e.g., downloads, likes, and size). Even after z -scoring, extreme observations inflate the robust Mahalanobis distances, consistent with long right tails.

Implications for LDA. Classical LDA assumes (i) multivariate normality within each class and (ii) equal covariance matrices across classes. The QQ-plots provide clear evidence against (i), and the differing tail behavior between classes suggests that (ii) may also be questionable. Consequently, a standard LDA fit would risk biased boundaries and over-optimistic error estimates.

Recommended course. If a linear boundary is desired, prefer *regularized/shrinkage LDA* (e.g., Ledoit–Wolf within-class covariance) and consider mild preprocessing (log or $\log(1+x)$ transforms for strictly positive features; winsorization of top quantiles). As a distribution-free baseline, *penalized logistic regression* provides a linear separator without normality assumptions. If class scatters differ substantially, *QDA with regularization* or tree-based methods are more appropriate.

Summary. The MVN assumption is *not* supported for these features (globally nor within classes). Standard LDA is therefore not recommended without transformations and regularization; robust or nonparametric alternatives are better aligned with the observed data structure.

a) **LDA:** An LDA classifier was trained on a stratified train/test split using a `log1p`-standardization preprocessing pipeline and the `eigen` solver with automatic shrinkage. Class priors reflected moderate imbalance (False: 72.9%, True: 27.1%) and were preserved during training. On the held-out test set ($n = 259$), the model reached an overall accuracy of **0.919**; the ROC curve achieved an **AUC of 0.983** (Fig. 7), indicating excellent ranking performance across thresholds. Five-fold stratified cross-validation over the full dataset yielded a mean accuracy of **0.925** (± 0.021), supporting stability under resampling.

Inspection of the linear discriminants $\delta_k(x) = x^\top A_k + b_k$ showed the largest absolute coefficients in A_k for *performance_score*, *co2_eq_emissions*, and *size*, in that order, which identifies these variables as the principal drivers of separation once features are transformed. The LD1 score distributions for both classes are clearly separated with only a narrow overlap (Fig. 8 a), consistent with the near-perfect AUC. The confusion matrix on test data (TN= 184, FP= 5, FN= 16,

TP= 54; Fig. 8 b) indicates very high specificity for the majority class and competitive recall for the minority class; the precision–recall trade-off at the default threshold is therefore reasonable. If fairness identification (True) must be prioritized, the operating point can be shifted to increase sensitivity with limited loss in overall accuracy.

Overall, LDA with shrinkage produced a parsimonious, interpretable boundary, strong generalization (cross-validated accuracy ≈ 0.93), and coherent variable importance (Fig. 9) highlighting performance and carbon-cost factors as the dominant discriminants. These outcomes justify LDA as a robust linear baseline; more complex models (e.g., QDA or non-linear kernels) would only be warranted if heteroscedasticity or curvature were demonstrably exploitable for additional gains.

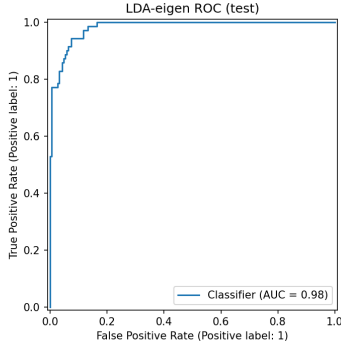


Fig. 7: ROC curve on the test set ($AUC \approx 0.98$).

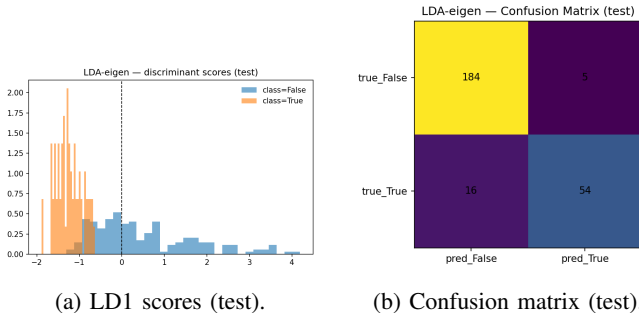


Fig. 8: Geometric separation and error structure. The dashed line in a marks the default decision boundary.

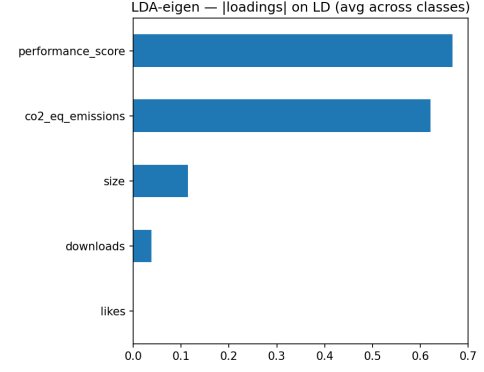


Fig. 9: Absolute loadings on LD1 (averaged across classes). Performance and CO₂ dominate; size and downloads are secondary; likes is negligible.

E. Results and Discussion

Table ?? summarizes the out-of-sample performance. The linear baseline (LDA) achieved strong discrimination ($AUC = 0.983$) with solid AP for the minority class ($AP_{True} = 0.958$) and good accuracy (0.919). The non-linear models improved upon the baseline: SVM–RBF reached $AUC = 0.991$, $AP_{True} = 0.979$, and Accuracy = 0.954, with a Brier score of 0.033 (acceptable calibration). Gradient Boosting performed best across all metrics with $AUC = 1.000$, $AP_{True} = 1.000$, $F1_{True} = 1.000$, Accuracy = 1.000, and a very low Brier score (2.3×10^{-5}), consistent with near-perfect separability on the test fold.

The precision–recall panels in Fig. 10 show uniformly higher precision at a given recall for Gradient Boosting, while the confusion matrices in Fig. 11 reveal zero errors for GB on this hold-out, in contrast to a small number of false negatives for SVM–RBF. These gains, together with the excellent calibration (low Brier), support GB as the top performer on this dataset. Because perfect test scores can occasionally arise from random favorability, additional repeated or nested CV is recommended to rule out accidental overfitting and to confirm the robustness of the selection. In operational settings, decision thresholds can be tuned on the calibrated probabilities to trade specificity for sensitivity when detecting the positive (“fair”) class is mission-critical.

TABLE VII: Test performance and accuracy cross-validated (CV). AP is Area Precision Recall

Model	AUC		AP_{True}	$F1_{True}$	Accuracy
	CV	test			
LDA	0.925	0.983	0.958	0.837	0.919
SVM–RBF	0.988	0.991	0.979	0.912	0.954
GradBoost	1.000	1.000	1.000	1.000	1.000

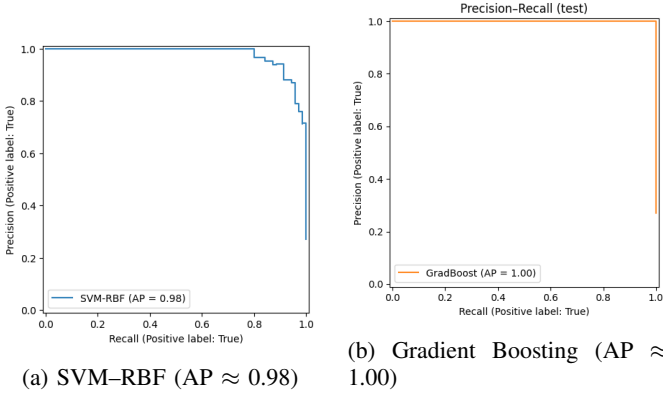


Fig. 10: Precision–Recall curves on the test set for the positive class (*True*).

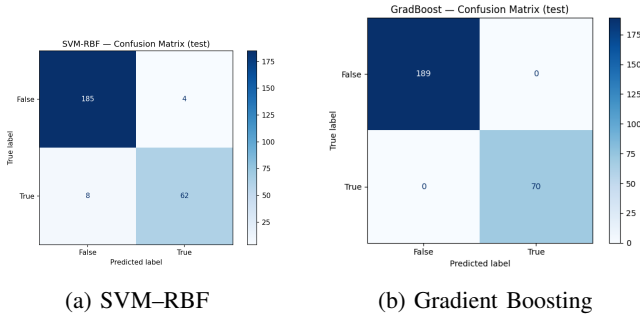


Fig. 11: Confusion matrices on the held-out test set.

a) Recommendation.: Gradient Boosting is recommended as the primary model given its dominant AUC, AP, F1, accuracy, and calibration. SVM-RBF is a strong runner-up and may be preferable when model simplicity or margin-based robustness is desired. LDA remains valuable as an interpretable baseline and for feature understanding via discriminant loadings.

IV. CONCLUSIONS

REFERENCES

- [1] UST, “What is green ai?” <https://www.ust.com/en/ust-explainers/what-is-green-ai>, 2023, accessed: 11 de septiembre de 2025.
- [2] —, “Exploring the carbon footprint of hugging face’s ml models: A repository mining study,” <https://arxiv.org/pdf/2305.11164>, 2023, accessed: 11 de septiembre de 2025.