LUISS GUIDO CARLI Undergraduate course of Management and Artificial Intelligence - 2nd Year Machine Learning - Academic Year 2025/2026

Compliance Radar

Team Members Di Renzo Carla ID 323411 Moriondo Tommaso ID 316331 Pazienza Lorenzo ID 308741 Raimondo Eleonora ID 321521

[Section 1] Introduction

This project develops a predictive and interpretive framework designed to identify potential compliance risks within an organisation. Using the *org_compliance_data.db* dataset, which brings together operational, audit, reporting, financial, and employee-related indicators, the analysis aims to highlight departments that may be more exposed to compliance challenges and to characterize the organisational conditions associated with such risks.

Formulated as a binary classification task with `is_at_risk` $\in \{0, 1\}$, the model achieves evidence-based insights intended to strengthen internal accountability and support ethical, informed decision-making across the organisation.

[Section 2] Methods

2.1 Dataset Analysis

The dataset includes a wide range of departmental indicators, such as: - Audit outcomes - Operational exposures - Governance support measures - Reporting behavior - Training participation - Employee engagement metrics

Two structural factors strongly influence the methodological design: 1. Class imbalance in the target variable. 2. Surveillance bias, where at-risk departments show artificially lower missingness due to more intensive auditing.

This requires careful handling of missing values and strong leakage control.

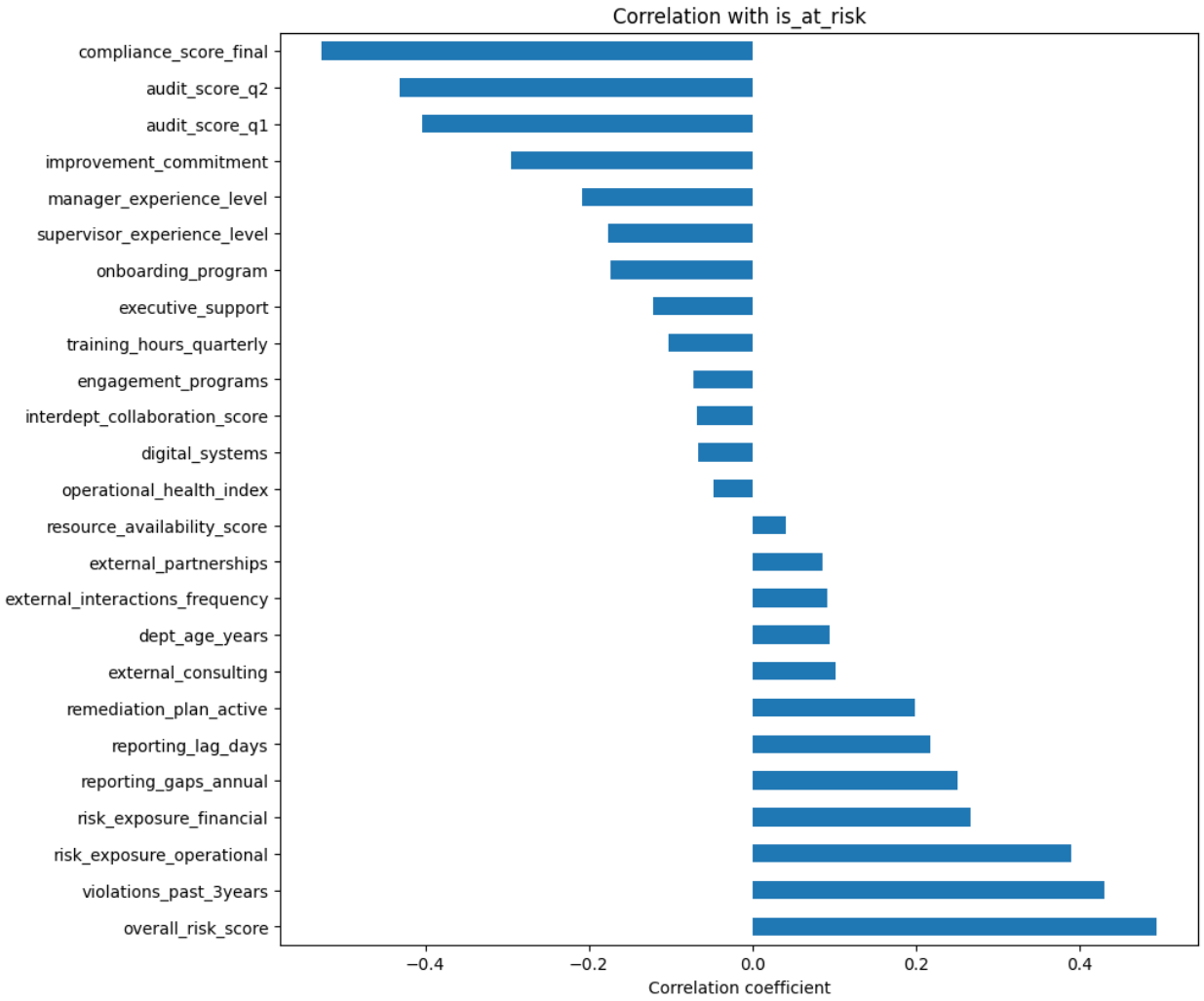
2.2 Exploratory Data Analysis (EDA)

The exploratory phase examines:

- Numerical variable distributions (histograms, box plots)
- Categorical frequency patterns
- Row-wise missingness scoring
- The relationship between missingness levels and risk status, reinforcing evidence of surveillance bias

Figure 1 – Correlation with the Target Variable

Below we report the Pearson correlations between each numerical feature and the target `is_at_risk`.



Strong negative correlations correspond to audit-related leakage, later removed from the modelling pipeline.

Figure 2 – Missingness Patterns and Surveillance Bias

Missing values differ systematically between at-risk and not-at-risk departments, confirming surveillance bias.

At-risk departments systematically show fewer missing values due to increased audit scrutiny.

2.3 Leakage Detection and Removal

Data leakage is confirmed because compliance and audit scores act as post-evaluation outcomes rather than independent predictors, showing near-perfect separation between risk classes. Furthermore, non-random missingness patterns betray the target, as high-risk departments rarely have missing data due to more frequent monitoring.

2.4 Pre-processing

Imputation

- Binary variables -> Mode
- Ordinal variables -> Median
- Continuous variables -> Median

Feature Engineering**

To represent organisational dynamics more explicitly, several composite indicators are constructed: - Governance Support Index - Operational Complexity - Training Efficiency - Experience Gap - Risk Gap

These engineered features are fully leakage-free and provide interpretable signals relevant to managerial decision-making.

#2.5 model preparation

Data Splitting

A stratified split preserves class proportions across folds.

Encoding

Categorical variables are transformed using target encoding with smoothing parameter `\alpha = 10`.

Scaling (only for logistic regression)

Numerical features are standardized using z-score scaling within the logistic regression pipeline.

2.6 Model Design and Training Pipeline

Three modelling approaches are implemented within a unified scikit-learn pipeline: - Logistic Regression (selected final model) - Decision Tree (CART) - Random Forest (RF)

Logistic Regression

Hyperparameters explored: - $C \in \{0.01, 0.1, 1, 10\}$ - Penalties: ℓ_1 and ℓ_2

Best configuration: - Penalty: ℓ_2 - $C = 0.01$

Decision Tree (CART)

Grid search explores: - Maximum depth - Minimum samples per split - Minimum samples per leaf - Maximum features

Random Forest

Hyperparameter search considers: - Number of trees - Maximum depth - Maximum features - Node-splitting constraints

2.7 Metrics evaluation Setup**

Metrics evaluated: - ROC-AUC - Accuracy - Precision - Recall - F1-score

Additional analysis includes: - precision–recall curves - decision-threshold evaluation

2.8 Environment Setup

The repository includes: - `main.ipynb` — Main notebook - `images/` — Figures generated during analysis - `requirements.txt` - Environment and setup

Environment and setup

This project was developed and tested with **Python 3.13.9** using a dedicated virtual environment (`.venv`). All required packages are listed in `requirements.txt`.

To reproduce the environment: #### **Create and activate the virtual environment** From the project root folder (where `main.ipynb` and `requirements.txt` are located): `python -m venv .venv`

Activate the environment:

- **Windows (PowerShell)** `.\.venv\Scripts\Activate.ps1`
- **Windows (cmd)** `.\.venv\Scripts\activate`
- **macOS / Linux** `source .venv/bin/activate`

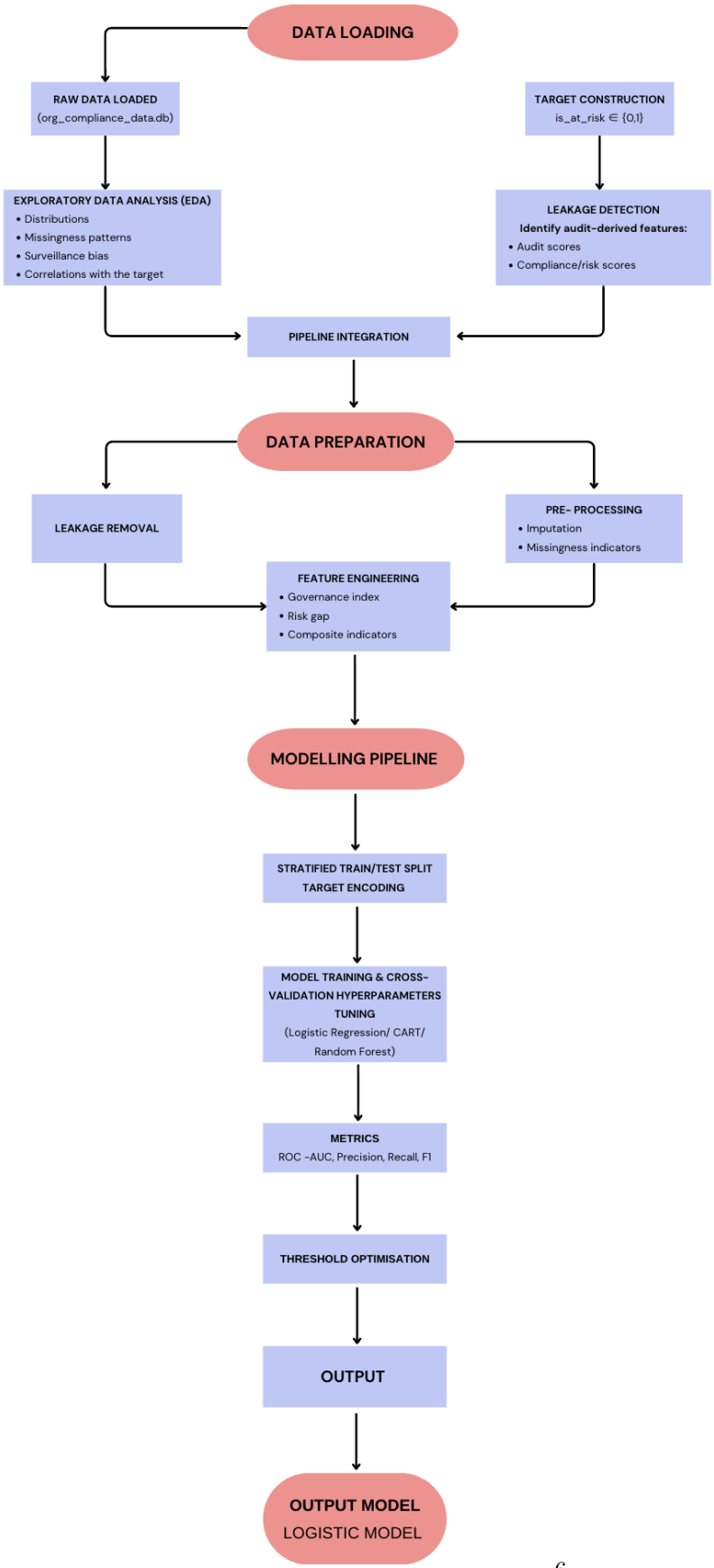
Install the dependencies With the virtual environment active: `pip install -r requirements.txt`

(Optional) Register the kernel for Jupyter / VS Code `python -m ipykernel install --user --name orgcompliance-ml --display-name "Org Compliance ML"`

In VS Code, open **main.ipynb** and select the kernel "Org Compliance ML (Python 3.13.9)" (or the '.venv' environment)

Run the notebook Once the kernel is selected, run **main.ipynb** from top to bottom to reproduce all analyses and models.

Figure 3 – Pipeline Flowchart



A schematic representation of preprocessing, feature engineering, cross-validation and model selection.

[Section 3] Experimental Design

3. Experimental Design

Our experiments were based on two things: 1. Develop an efficient Compliance Radar to control risky departments. 2. Measure the extent to which performance of model is conditional on surveillance-related signals (patterns of missingness and remedial action).

In every experiment we take the following pipeline: - Stratified train test split on is at risk - Target-encoded features which are leak-free (fitted on the training set only) - Model selection or stratified K-fold cross-validation on the training set - Final test on the held-out test set

3.1 Experiment 1 — Models Training and Evaluation with full set of features

The subject in this experiment was to compare three models with the complete set of features.

Main purpose. Compare diverse trained models on the complete feature-set, including missing-value, flags and the remediation plan active variable, and choose a primary Compliance Radar model.

Methods. - Standardized Regression (of Logistic) (regularized) - decision tree classifier (CART) - Random Forest classifier

On the training-set, we tune each model by stratified K-fold cross-validation hyperparameters (strength of regularization and type penalty of LR; depth and splitting CART parameters; the number of trees, depth and leaf parameters of RF).

Evaluation metrics We report on validation folds and on the test set: - ROC-AUC (general discriminating ability) - F1-score, precision and recall of the at-risk group (is at-risk = 1) - Confusion matrix and accuracy in it are complete

This experiment demonstrates that a regularized Logistic Regression already gives a powerful and interpretable Compliance Radar, and CART and Random Forest provide non-linear benchmarks.

3.2 Experiment 2 — Threshold Optimization

In the second experiment, threshold optimization of Logistic Regression was applied to the data collected in experiment 1 and the results were compared with the initial analysis.

Main purpose Tune the decision level of the tuned Logistic Regression model in order to represent the better cost of business of missing at-risk departments (false negatives) versus triggering false positives (additional warnings).

Baseline - The 0.5 default threshold on the probability forecast. - P (is at risk = 1), which was used in Experiment 1

Methods With the tuned Logistic Regression of Experiment 1: - Calculate individualized predictions on the test set - Assess a grid of decision thresholds (e.g. between 0.1 and 0.9) - At every threshold, calculate precision, recall, F1-score and the confusion matrix

Evaluation metrics - Recall on the at-risk class (priority: do not miss really risky departments) - Accuracy on the at-risk group (modulate the volume of false notifications) - F1-score as a summary of the trade-off

We will then choose an operating threshold which will retain a high recall at low cost. accuracy at an acceptable degree. It is the primary model, this calibrated Logistic Regression presented in the Results section.

3.3 Experiment 2 Models Training and Evaluation without Surveillance Bias

The experiment 3 involved a sensitivity analysis of the influence of surveillance bias on the results of the experiment.

Main purpose Determine the extent of model performance that relies on the behavior of monitoring (surveillance bias) using comparison between models trained with and without variables encoding activity of monitoring and remediation.

Baselines - Experiment 1 best Logistic Regression and the best random forest models (all features, including missingness flags and remediation plan active).

Methods - Weed out a diminished set of features by dropping: - missing-value indicator features; - the active remediation plan variable, which is named remediation plan active. - Retrain and retune: - Logistic Regression (cross-validation procedure as in Experiment 1); - Random Forest (equivalent hyperparameter search procedure). - Re-use the threshold-selection reasoning of Experiment 2 on the shrunk models.

Evaluation metrics On each of our compared models: between full and reduced feature sets: - test ROC-AUC - F1-score, accurate and recalls on the at-risk class - modification of the confusion matrix, particularly of the number of missed at-risk departments.

The evident decrease in performance when absenteeism signs and remediation-plan are eliminated. It is gathered that a significant portion of predictive power is due to how and where the organisation gathers data and implements remediation not only underlying risk drivers. This experiment is empirical justification of the discussion on surveillance-bias in the Results and Conclusions sections.

[Section 4] Results

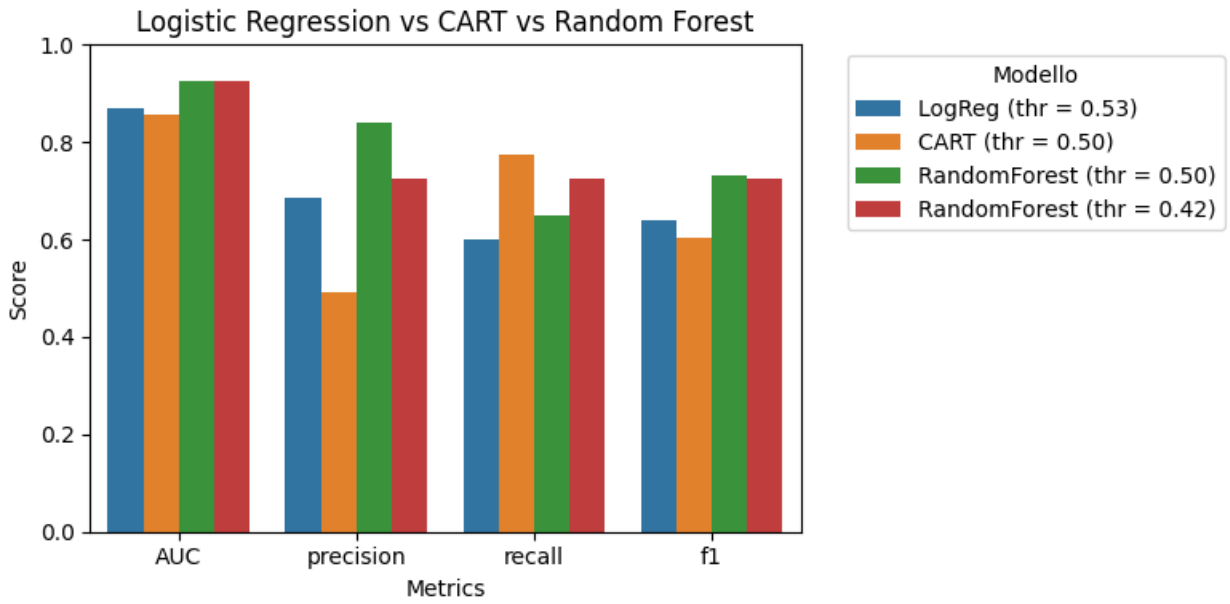
4.1 Models trained on the full feature set (with missing flags)

In this first block we train and tune three models on the full feature set, including: - Engineered numerical variables
- Target-encoded categorical variables
- Missing-value flags
- `remediation_plan_active` and other monitoring-related features

The final comparison is based on the test set and uses the thresholds selected in the notebook (Section 16.6):

Model Threshold | ROC-AUC | Precision (risk=1) | Recall (risk=1) | F1 (risk=1) Logistic Regression 0.53 | 0.871 | 0.686 | 0.600 | 0.640 CART (Decision Tree) 0.50 | 0.857 | 0.492 | 0.775 | 0.602 Random Forest (baseline) 0.50 | 0.926 | 0.839 | 0.650 | 0.732 Random Forest (optimized) 0.42 | 0.926 | 0.725 | 0.725 | 0.725

Figure 4 – Model Comparison Surveillance Bias



Logistic Regression (full feature set)
- Regularized LR with class weights balanced.
- Good overall discrimination (AUC = $\alpha \approx 0.87$).
- With a threshold of $\alpha \approx 0.53$, it achieves precision = $\alpha \approx 0.69$ and recall = $\alpha \approx 0.60$ on at-risk departments, giving a balanced but slightly conservative radar.

CART (Decision Tree)
- Similar AUC ($\alpha \approx 0.86$), but with a different error profile:
– High recall ($\alpha \approx 0.78$) on at-risk departments -> catches many risky units.
– Lower precision ($\alpha \approx 0.49$) -> more false alarms.
- Provides simple, rule-based splits but is less stable and more sensitive to small variations in the data.

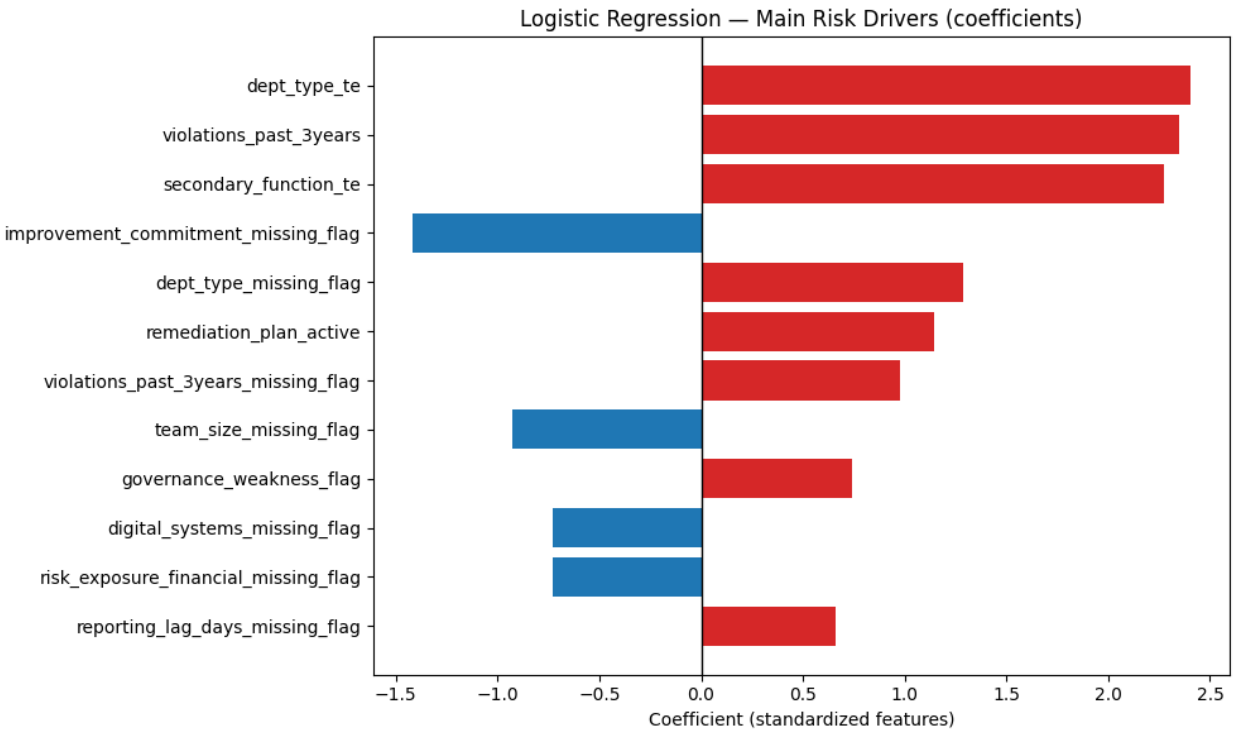
Random Forest (full feature set)
- Highest AUC ($\alpha \approx 0.93$).
- At threshold 0.50, precision is high ($\alpha \approx 0.84$) but recall is moderate ($\alpha \approx 0.65$).
- After threshold optimization (thr = $\alpha \approx 0.42$), RF reaches a balanced recall and precision of $\alpha \approx 0.73$, with F1 = $\alpha \approx 0.73$.
- However, it remains a complex ensemble with a noticeable train–test gap, and is substantially less interpretable than Logistic Regression.

4.2 Interpretation of Logistic Regression coefficients (with missing flags)

Using the full feature set (including missing flags and remediation signals), Logistic Regression provides a transparent view of which variables act as risk drivers or buffers.

The coefficient ranking (sorted by absolute value) shows the following top factors:

Figure 5 – Logistic Regression Coefficient Analysis (Surveillance Bias)



Main risk drivers (positive coefficients)

- **`dept_type_te`**
  Certain department types, as captured by target encoding, are structurally more exposed and strongly associated with being at risk.
- **`violations_past_3years`**
  A higher number of past violations is one of the strongest predictors of being classified as at risk.
- **`secondary_function_te`**
  Some secondary functions (e.g. support / non-core roles with weak control ownership) correspond to higher estimated risk.
- **`dept_type_missing_flag`**
  Missing information on department type tends to increase risk, suggesting that incomplete structural data correlates with problematic units.
- **`remediation_plan_active`**
  Departments under an active remediation plan are much more likely to be classified as at risk, reflecting supervisory attention to already critical situations.

- `violations_past_3years_missing_flag`
  Missing history on violations is itself associated with higher risk, likelybecause problematic departments are scrutinised and documented differently from low-risk ones.

- `governance_weakness_flag`
  Explicit governance weaknesses strongly increase the probability of being flagged at risk.

- `reporting_lag_days_missing_flag`
  Gaps or missing data on reporting lags also contribute positively to risk, again consistent with patchy monitoring around problematic units.

- `digital_systems`
  Certain configurations of digital systems (e.g. legacy or fragmented setups) are associated with higher risk in the learned pattern.

Protective factors (negative coefficients)

- `improvement_commitment_missing_flag` **(negative)**
  Missingness on improvement commitment is associated with lower risk, indicating that heavily monitored, higher-risk departments are precisely those with detailed improvement plans recorded — another manifestation of surveillance bias.

- `team_size_missing_flag`, `digital_systems_missing_flag`, `risk_exposure_financial_missing_flag` **(negative)**
  Several missing flags are protective in the model: departments with fewer recorded details on size, systems, or financial exposure tend to be the ones that are perceived as low risk and therefore monitored less.

- `onboarding_program` **(negative)**
  The presence of a structured onboarding program reduces the predicted risk, signalling more mature HR and process practices.

Overall, the full-feature Logistic Regression confirms that: - True risk drivers (violations, risk exposure, governance weaknesses) are important - Many monitoring / missingness signals (`*_missing_flag`, `remediation_plan_active`) also carry strong weight.

This justifies a second round of modelling without missing flags, to separate underlying risk from surveillance behavior.

4.3 Models trained without missing-value flags

In the second block we rebuild the models on a reduced "no-leakage" feature set, where we remove: - All missing-value indicator variables
- Other obvious surveillance-related flags

We focus on Logistic Regression and Random Forest as the two main benchmarks.

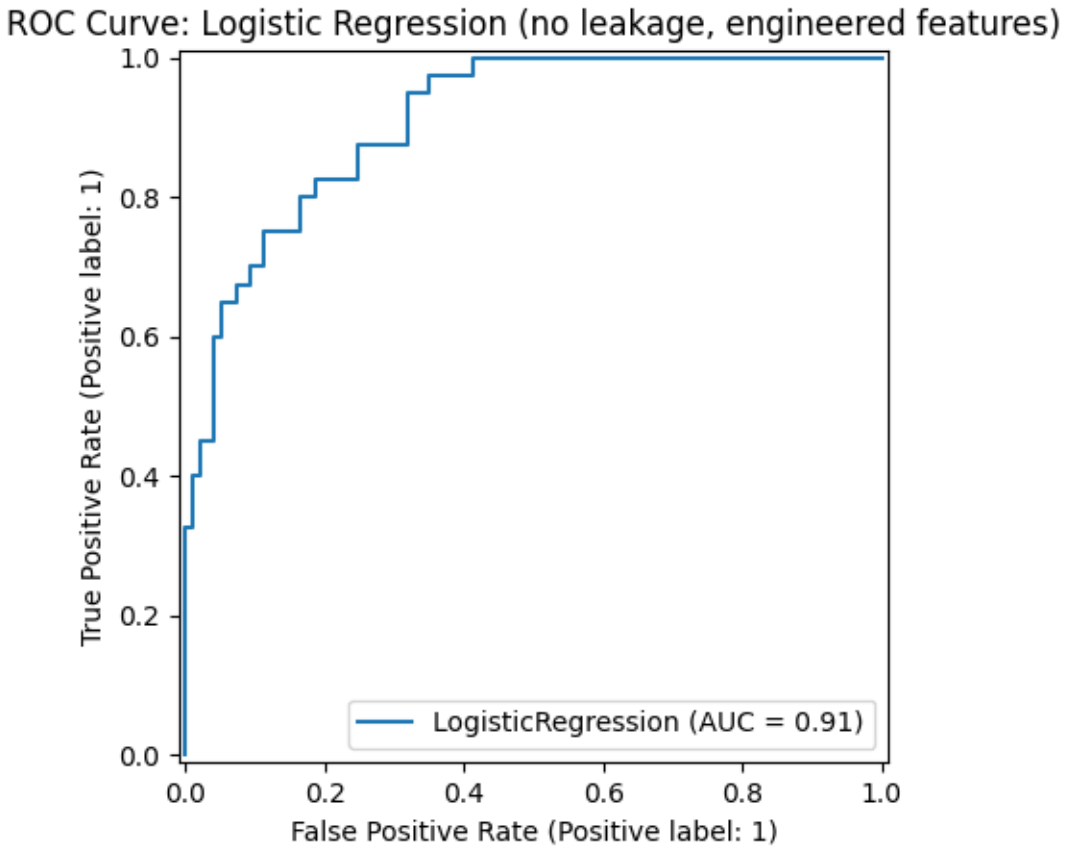
Logistic Regression (no missing flags)

On the leakage-free, target-encoded feature set: - Best parameters (9-fold CV): C = $\alpha \approx 0.01$, penalty = $\alpha \approx L2$ - Test ROC-AUC = $\alpha \approx 0.87$ (threshold = 0.5).
- At threshold 0.5: – Precision (risk=1) = $\alpha \approx 0.54$
– Recall (risk=1) = $\alpha \approx 0.88$ – F1 = $\alpha \approx 0.67$ This default threshold heavily prioritizes recall on at-risk departments.

To avoid overly skewing towards recall, the notebook searches for a balanced threshold using the precision–recall curve: - Optimal balanced threshold = $\alpha \approx 0.62$
- At this threshold: – Precision (risk=1) = $\alpha \approx 0.72$ – Recall (risk=1) = $\alpha \approx 0.72$ – F1 = $\alpha \approx 0.72$

Cross-validation (9-fold) shows: - ROC-AUC: train = $\alpha \approx 0.91$, test = $\alpha \approx 0.89$ (gap = $\alpha \approx 0.02$)
- F1: train = $\alpha \approx 0.70$, test = $\alpha \approx 0.67$ (small gap)
Indicating a stable and not overfitted linear model.

Figure 6 – Logistic Regression ROC - AUC (No-bias)



Random Forest (no missing flags)

Using the same no-leakage feature set: - Tuned via grid search (5-fold CV) with balanced class weights - Test ROC-AUC $\alpha \approx 0.92$ at threshold 0.5 - At threshold 0.5: – Precision (risk=1) = $\alpha \approx 0.87$ – Recall (risk=1) = $\alpha \approx 0.50$ – F1 = $\alpha \approx 0.63$

Cross-validation diagnostics show: - ROC-AUC: train = $\alpha \approx 0.99$, test = $\alpha \approx 0.92$ (larger gap than LR) - F1: train = $\alpha \approx 0.82$, test = $\alpha \approx 0.68$

This indicates a more complex, higher-variance model, with stronger performance in terms of AUC but also more sensitivity and some overfitting.

Comparison (no missing flags)

On the no-leakage feature set: - Both models keep high AUC (= $\alpha \approx 0.87$ for LR, = $\alpha \approx 0.92$ for RF).
- Logistic Regression, after threshold adjustment, offers balanced recall and precision around 0.72, with much better interpretability and smaller train–test gaps. - Random Forest provides slightly higher discriminating power, but with: – Lower recall at the default threshold, – Stronger overfitting signals, – and a black-box nature that complicates governance discussions.

For these reasons, the no-leakage Logistic Regression is adopted as the final Compliance Radar model, while Random Forest serves as a non-linear benchmark.

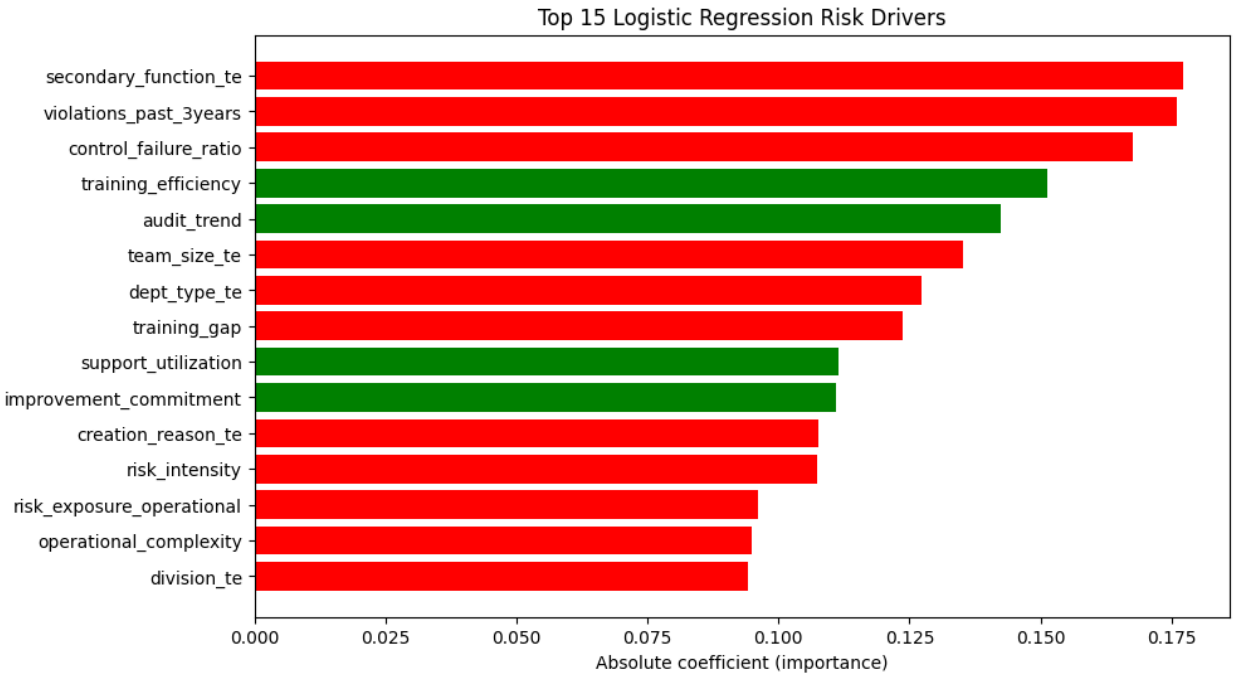
[Section 5] Conclusion

5.1 Final business interpretation (Logistic Regression without missing flags)

Removing missing flags and surveillance-related variables does not destroy the model's predictive ability (AUC remains = $\alpha \approx 0.87$), and highlights more clearly the structural and behavioral risk drivers.

From the coefficient analysis on the no-leakage feature set:

Figure 7 – Top 15 Logistic Regression Abs Value Coefficient (Risk Drivers / Buffers)



Key risk drivers

- **Violations history**
  `violations_past_3years` is one of the strongest positive predictors: departments with repeated past breaches are much more likely to be at risk.
- **Functional profile and role**
  `secondary_function_te`, `dept_type_te` and `creation_reason_te` capture the fact that some functions and department types are structurally more exposed (e.g. risk-taking or control-light roles, newly created units with immature processes).
- **Control failures and risk intensity**
  `control_failure_ratio`, `risk_intensity` and `risk_exposure_operational` all increase risk, linking operational complexity and control breakdowns directly to compliance vulnerability.
- **Training and performance gaps**
  `training_gap` (misalignment between required and actual training) and low training efficiency are associated with higher risk.

- **Operational complexity**
  `operational_complexity` contributes positively: more complex operations tend to be harder to control and more exposed to errors or misconduct.

Protective factors

- **Training efficiency and improvement commitment**
  High `training_efficiency` and strong `improvement_commitment` reduce risk: departments that convert training into concrete improvements and actively work on remediation are less likely to be flagged.
- **Audit trend and support utilization**
  Positive `audit_trend` (improving audit results over time) and effective `support_utilization` (using central support resources) act as buffers against risk.
- **Managerial and supervisory quality**
  Higher `manager_experience_level` and `supervisor_experience_level` correlate with lower risk, emphasizing the importance of experienced leadership.
- **Onboarding and staff development**
  The presence of `onboarding_program` and higher `training_hours_quarterly` are associated with reduced predicted risk, indicating more mature people processes.

Business takeaway

The final no-leakage Logistic Regression model delivers a transparent Compliance Radar that: - Flags at-risk departments based on substantive drivers (violations, exposure, complexity, governance quality, training gaps) - Avoids relying on missingness and remediation flags as shortcut proxies for surveillance intensity - Provides interpretable coefficients that can be directly mapped to actionable levers for the compliance function (strengthening training, improving governance, supporting high-exposure departments, etc.)

5.2 Main Takeaways

Data integrity is central to meaningful compliance-risk modelling. Addressing audit-related leakage and surveillance bias is crucial to obtain realistic and interpretable models that reflect organisational reality. Logistic Regression offers a strong balance between: - Performance — ROC-AUC around $\alpha \approx 0.87$ on the no-leakage dataset - Interpretability — straightforward explanation of risk drivers

5.3 Future Work

Potential extensions include: - Fairness analysis across departments and groups - Causal inference methods to distinguish correlation from causation - Temporal modelling, incorporating time dynamics of risk and audits - Scenario-simulation dashboards for decision-makers

Overall, this project provides a transparent and responsible foundation for compliance analytics in organizations.