HW2 Interpretability Report in IEEE Format Comprehensive Tabular and Vision Explanation Analysis

Taha Majlesi
Student ID: 810101504
Department of Electrical and Computer Engineering, University of Tehran
Course: Trusted Artificial Intelligence (Homework 2)

***Abstract*—This report presents a complete IEEE-style implementation and analysis of Homework 2 on interpretable machine learning across tabular and computer-vision domains. The final pipeline is deterministic and robust to offline execution by introducing controlled fallback behavior for both data and model initialization. Two tabular models (MLP and NAM) are trained and evaluated, and local explanations are generated with LIME and SHAP. For vision, Grad-CAM, Guided Backpropagation, SmoothGrad, and Guided Grad-CAM are implemented and exported as reproducible artifacts. Every required figure is interpreted explicitly, with one dedicated paragraph per plot result, and all experiments are linked to executable commands and traceable files.**

***Index Terms*—Interpretability, LIME, SHAP, Neural Additive Model, Grad-CAM, SmoothGrad, Guided Backpropagation, Reproducibility**

I. Introduction

Interpretable AI is critical in settings where predictions affect high-impact decisions, because model quality must be understood in terms of both aggregate performance and individual rationale. This homework targets that goal through two complementary workloads: tabular binary classification with feature-level explanation, and visual explanation of convolutional network outputs. The implementation was finalized as an end-to-end reproducible pipeline that generates all required report figures and compiles to a single PDF artifact. Beyond basic performance, the report emphasizes explanation stability, probabilistic calibration, and method agreement, because interpretability is only useful when explanations are consistent under reasonable perturbations and aligned with the trained model's actual decision logic. Accordingly, each required method is not only implemented but also supported by explicit theoretical framing, diagnostic metrics, and plot-level interpretations that make the reasoning traceable rather than impressionistic.

II. Reproducible Setup

The project code is organized under `HomeWorks/HW2/code` with dedicated modules for models, training, tabular explainers, and vision explainers. The final figure export entry point is `code/generate_report_plots.py`, which writes artifacts into `HomeWorks/HW2/report/figures`. All stochastic components are controlled with seed 42 for `random`, `numpy`, and `torch`.

Because execution may occur without internet, two reliability safeguards were implemented: (i) tabular data download falls back to a deterministic synthetic diabetes-like dataset, and (ii) pretrained VGG16 loading falls back to randomly initialized weights. This design ensures the homework remains fully runnable in constrained environments without breaking downstream analysis code.

A. Dataset and Preprocessing Summary

The tabular dataset contains eight numerical features and a binary outcome. Input features are standardized using `StandardScaler`, i.e.,

$$x'_j = \frac{x_j - \mu_j}{\sigma_j}, \tag{1}$$

which equalizes scale across heterogeneous medical measurements and stabilizes both gradient descent and distance-based explainers. Data are split into train/validation/test with a 70/10/20 stratified partition to preserve class priors across splits, which is critical for stable recall and F1 measurement under moderate imbalance. These preprocessing and split constraints are explicitly enforced in the pipeline to make downstream comparisons (MLP vs NAM and LIME vs SHAP) statistically meaningful.

B. Evaluation Protocol and Reporting Guarantees

The evaluation protocol reports thresholded metrics (accuracy, recall, F1, confusion matrix) and threshold-free metrics (ROC-AUC, average precision), along with calibration (Brier score). This is a deliberate completeness requirement: thresholded metrics quantify operational behavior at a specific decision rule, while threshold-free metrics describe ranking capability independent of threshold choice, and calibration quantifies probability quality. All metrics are computed on the held-out test split only, and every figure is tied to a deterministic artifact with a fixed filename to prevent reporting drift.

III. Requirement Coverage Map

TABLE I
HOMEWORK REQUIREMENT COVERAGE

Requirement Block	Where It Is Covered in This Report
Tabular EDA (correlation matrix, pairplot, class balance, outliers)	Figs. 1 to 4 in the Plot-by-Plot section.
Preprocessing and MLP training/evaluation	Methods (Tabular Models), Code Walkthrough (tabular pipeline), Tables I–II, Figs. 5–9.
LIME and SHAP on three samples + comparison	Figs. 12–14, Figs. 15–17, Fig. 11.
Correlation linkage with explanations	Fig. 18 and its dedicated interpretation subsection.
NAM analysis and interpretability tradeoff	Methods (NAM additive model), Fig. 20, Tables I–II, and its dedicated interpretation subsection.
Bonus GRACE contrastive sample analysis	Fig. 19, Table VI, and the GRACE interpretation subsection.
Vision explainability (Grad-CAM, Guided Backprop, Guided Grad-CAM, SmoothGrad)	Methods (Vision), Code Walkthrough (vision module), and Figs. 22–27.
Adversarial perturbation and saliency comparison	Fig. 28 and the adversarial interpretation subsection.
Activation maximization (Hen) with TV and random shifts	Fig. 29, Methods (feature visualization objective), and Appendix B-G.

IV. Methods

A. Tabular Models and Optimization

The MLP classifier follows the architecture $8 \to 100 \to 50 \to 50 \to 20 \to 1$, optimized with binary cross-entropy on logits. For a sample x, the probability output is $\sigma(f_\theta(x))$, where

$$\sigma(z) = \frac{1}{1+e^{-z}}. \tag{2}$$

The population objective can be expressed as empirical risk minimization:

$$\hat{\theta} = \arg\min_\theta \frac{1}{N}\sum_{i=1}^{N} \ell(y_i, f_\theta(x_i)), \tag{3}$$

with ℓ equal to logistic loss

$$\ell(y,z) = -y\log\sigma(z) - (1-y)\log(1-\sigma(z)), \quad z = f_\theta(x). \tag{4}$$

Under the Bernoulli likelihood model, this objective is equivalent to maximizing conditional log-likelihood, so the learned score approximates log-odds when the model class is sufficiently expressive. The first-order gradient identity

$$\frac{\partial \ell}{\partial z} = \sigma(z) - y \tag{5}$$

shows that optimization directly pushes predicted probability toward the target label; this connects training dynamics in loss curves to calibration and threshold behavior later in evaluation. At inference time, class labels depend on a decision threshold t, $\hat{y}_t = \mathbf{1}[p(x) \geq t]$, and under cost-sensitive decision theory with calibrated probabilities the Bayes-optimal threshold is

$$t_C = \frac{C_{\mathrm{FP}}}{C_{\mathrm{FP}} + C_{\mathrm{FN}}}, \tag{6}$$

which explains why threshold tuning is theoretically part of the decision rule, not model fitting.

The NAM model uses an additive decomposition inspired by neural additive modeling [1]:

$$f(x) = \sum_{j=1}^{d} g_j(x_j), \quad \hat{y} = \sigma(f(x)). \tag{7}$$

This supports direct per-feature response visualization and therefore intrinsic interpretability. The additive structure is theoretically important because it removes interaction terms from first-order decomposition, so each g_j can be interpreted as a marginal contribution function while holding the latent representation fixed. In differential form this yields $\partial f/\partial x_j = g_j'(x_j)$ and cross-partials $\partial^2 f/(\partial x_j \partial x_k) = 0$ for $j \neq k$, which formalizes the interpretability gain: each feature effect is directly inspectable without post-hoc surrogate approximation. The corresponding limitation is equally explicit: when true data-generating mechanisms contain strong interactions, additive models may sacrifice some discriminative power relative to unconstrained black-box architectures.

B. Tabular Explanation Methods

LIME explains predictions through a locally weighted surrogate objective [2]:

$$\xi(x) = \arg\min_{g \in \mathcal{G}} \ \mathcal{L}(f, g, \pi_x) + \Omega(g). \tag{8}$$

In practice, π_x is a locality kernel (typically exponential in distance), e.g.

$$\pi_x(z) = \exp\left(-\frac{D(x,z)^2}{\sigma_\pi^2}\right), \tag{9}$$

so nearby perturbations receive higher weight than distant points. This makes LIME an estimator of local tangent behavior rather than global model logic, and its fidelity depends on neighborhood sampling, kernel width, and surrogate class complexity. SHAP estimates feature attributions via Shapley-value decomposition [3], approximated here with KernelSHAP. For any sample x, SHAP satisfies local additivity:

$$f(x) \approx \phi_0 + \sum_{j=1}^{d} \phi_j, \tag{10}$$

where ϕ_j is the feature attribution. Theoretical attractiveness comes from Shapley axioms (efficiency, symmetry, dummy, additivity), which make SHAP values uniquely defined in cooperative game settings: efficiency enforces exact attribution summation to the prediction difference, symmetry gives equal credit to exchangeable features, dummy assigns zero effect to irrelevant features, and additivity preserves decomposition consistency across summed games. LIME and SHAP may still diverge in practice because LIME fits a weighted local

surrogate on sampled perturbations, whereas SHAP estimates globally consistent additive credits under explicit coalition-value assumptions (interventional or conditional masking). Therefore SHAP is treated as the primary faithful-reference explainer for quantitative attribution consistency, while LIME is treated as a complementary local rule-based approximation. To satisfy assignment requirements on local visualization, three deterministic test instances are explained by both methods and SHAP force plots are exported. To connect local explanations with global feature dependence, SHAP mean-absolute attribution magnitudes are compared against absolute Pearson correlation with the target, which allows direct inspection of whether correlation-dominant variables are also attribution-dominant under the fitted model.

C. GRACE-Style Contrastive Analysis

As a bonus contrastive study, one feature is perturbed around a selected test sample and the probability shift is analyzed with SHAP. Let x be an instance and x' be a minimally edited variant along a feature direction e_j, i.e.,

$$x' = x + \delta e_j. \tag{11}$$

The contrastive effect is quantified by

$$\Delta p = \sigma(f(x')) - \sigma(f(x)), \tag{12}$$

while attribution change is measured through $\Delta\phi_j = \phi_j(x') - \phi_j(x)$. This setup operationalizes GRACE-style reasoning by linking controlled feature intervention to both predictive and explanatory movement.

D. Vision Explanation Methods

Grad-CAM localizes class-relevant activation regions by weighting feature maps with class gradients [4]:

$$\alpha_k^c = \frac{1}{Z}\sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k}, \tag{13}$$

$$L_{\text{Grad-CAM}}^c = \text{ReLU}\left(\sum_k \alpha_k^c A^k\right). \tag{14}$$

This can be interpreted as a first-order Taylor linearization around the target layer: channel-wise pooled gradients α_k^c estimate the sensitivity of class score y^c to activation map A^k, and the weighted sum forms a class-specific relevance field. The ReLU gate removes negative evidence so the map emphasizes positively class-supportive regions. Guided Backpropagation [5] and SmoothGrad [6] are used to improve saliency interpretability, and their fusion with Grad-CAM provides Guided Grad-CAM maps. From a differential viewpoint, saliency is a Jacobian-derived signal $S(x) = \nabla_x y^c$. SmoothGrad estimates a denoised gradient field by Monte Carlo averaging over Gaussian perturbations:

$$\hat{S}(x) = \frac{1}{K}\sum_{k=1}^{K} \nabla_x y^c(x + \epsilon_k), \;\; \epsilon_k \sim \mathcal{N}(0, \sigma^2 I), \tag{15}$$

which acts as a variance-reduction estimator for pixel-level sensitivity. Guided Backprop can be written as a backward gating rule through ReLU units:

$$R^{(l)} = \mathbf{1}[x^{(l)} > 0] \odot \mathbf{1}[R^{(l+1)} > 0] \odot R^{(l+1)}, \tag{16}$$

where $R^{(l)}$ denotes backpropagated relevance. Guided Grad-CAM then combines localization and edge detail via multiplicative fusion:

$$M_{\text{GGC}}(x) = \text{ReLU}(L_{\text{Grad-CAM}}^c) \odot |S_{\text{guided}}(x)|. \tag{17}$$

For adversarial analysis, the report uses FGSM [7]:

$$x_{\text{adv}} = x + \epsilon \cdot \text{sign}(\nabla_x \mathcal{L}(f(x), y)), \tag{18}$$

which is obtained by maximizing first-order loss increase under an ℓ_∞-bounded perturbation budget. The analysis compares saliency for the original target class before and after perturbation to test explanation robustness under worst-case local shifts. For feature visualization, activation maximization solves

$$x^\star = \arg\max_x y^c(x) - \lambda_{\text{TV}}\,\text{TV}(x), \tag{19}$$

where total variation suppresses high-frequency oscillations. Random spatial shifts during optimization implement translation-aware data augmentation in input space, discouraging brittle pixel-locked shortcuts and promoting semantically stable patterns.

E. Theoretical Basis for Plot Interpretation

Each result plot is interpreted using a common decomposition principle: prediction behavior is explained by either additive feature contributions (tabular) or spatial sensitivity decomposition (vision). For tabular plots, signed attribution magnitude is treated as an estimator of directional influence in logit space, and stability is assessed through rank consistency, overlap of top-k contributors, and perturbation response. For vision plots, heatmaps are treated as approximate relevance densities over image coordinates, and plausibility is assessed by concentration, smoothness, cross-method agreement, and behavior under adversarial stress. This theory-driven lens makes assumptions explicit: the classifier is approximately locally smooth, explainer perturbations are representative of local neighborhoods, and saliency maps are interpreted as relative (not absolute-causal) evidence fields. Under these assumptions, qualitative figures can be analyzed rigorously rather than by visual intuition alone.

F. Diagnostic and Stability Metrics

To move beyond accuracy-only reporting, the expanded pipeline adds threshold-free discrimination metrics, probability-quality metrics, attribution-consistency metrics, and saliency-stability metrics. ROC-AUC is interpreted as ranking quality and can be written as

$$\text{AUC} = \int_0^1 \text{TPR}(u)\,du, \tag{20}$$

where $u = \text{FPR}$, and equivalently $\text{AUC} = \mathbb{P}(s(x^+) > s(x^-))$ for independent positive/negative samples. Average precision

summarizes precision-recall behavior under class imbalance and can be approximated by

$$\mathrm{AP} \approx \sum_n (R_n - R_{n-1}) P_n. \tag{21}$$

Calibration quality is quantified through the Brier score

$$\mathrm{BS} = \frac{1}{N} \sum_{i=1}^{N} (p_i - y_i)^2, \tag{22}$$

which is a proper scoring rule, so lower values correspond to better probabilistic forecasts. In reliability analysis, Brier behavior can be interpreted through reliability-resolution decomposition, clarifying whether improvement comes from better calibration or better class-separation structure. Threshold sensitivity is analyzed by selecting an operating point

$$t^\star = \arg\max_{t \in [0,1]} \mathrm{F1}(t), \tag{23}$$

which explicitly models the precision-recall tradeoff induced by the decision threshold. Global feature reliance is measured with permutation importance

$$I_j = \mathbb{E}[\mathcal{M}(f, X, y) - \mathcal{M}(f, \pi_j(X), y)], \tag{24}$$

where π_j denotes feature-j shuffling and $\mathcal{M}$ is test accuracy. Positive I_j indicates destructive effect from shuffling (feature reliance), while near-zero or negative values indicate redundancy or estimator noise. For local explanation agreement, the report uses Spearman rank correlation between SHAP and LIME feature vectors plus top-3 overlap, distinguishing directional rank-consistency from exact magnitude matching. For vision stability, SmoothGrad maps at different sample counts K are compared with cosine similarity

$$\cos(a, b) = \frac{a^\top b}{\|a\|_2 \|b\|_2}, \tag{25}$$

which operationalizes convergence of saliency structure as Monte Carlo averaging strength increases. Complementary smoothness diagnostics use normalized entropy and total variation, where decreasing total variation with increasing K indicates suppression of high-frequency attribution noise and entropy trends indicate how saliency mass spreads across spatial support.

G. Method Sensitivity and Validity Checks

Each explanation method has explicit hyperparameters that control fidelity–stability tradeoffs, so the report treats them as part of the method definition rather than afterthoughts. For LIME, kernel width σ_π and sampling budget determine locality and variance; small σ_π increases fidelity but can destabilize coefficients. For SHAP/KernelSHAP, the number of coalition samples controls estimator variance, and the masking definition (interventional vs conditional) governs the notion of "missing" features; this report uses a fixed background subset and sampling budget to keep SHAP stable and reproducible. For Grad-CAM, layer choice controls spatial granularity: deeper layers yield stronger semantic focus but coarser resolution, while earlier layers yield sharper but more class-ambiguous maps. For SmoothGrad, the noise standard deviation σ and sample count K control the bias–variance tradeoff: higher σ can over-smooth, while low σ may preserve noise. The pipeline locks these settings and reports convergence metrics so the interpretability claims can be traced to explicit algorithmic assumptions rather than undocumented defaults.

V. Code-Level Implementation Walkthrough

A. Execution Entry Point

The report-generation script is designed as a single deterministic orchestrator that guarantees reproducible artifacts from one command. The `main()` routine first sets global seeds through `_set_seed()` for `random`, `numpy`, and `torch`, then guarantees output availability via `_ensure_dirs()`, and finally executes `generate_tabular_figures()` followed by `generate_vision_figures()` in a fixed order. This ordering is intentional: tabular plots and metrics are produced first to validate the data/model path before invoking vision explainability components, so failures are easier to localize. Two helper routines are especially important for robustness: `_predict_fn_factory()` adapts a PyTorch logit model into a LIME-compatible `predict_proba`-style callable returning an $N \times 2$ probability matrix, while `_normalize_shap_output()` resolves SHAP API shape differences across versions (class-first, sample-first, or flat vectors), preventing silent plotting bugs when library behavior changes.

B. Model Definitions (`models.py`)

The classification models are intentionally minimal but structurally aligned with homework requirements. `MLPClassifier` uses the architecture $8 \to 100 \to 50 \to 50 \to 20 \to 1$ with `BatchNorm1d` at the first hidden layer and dropout regularization in the middle block, returning raw logits for numerically stable BCE-with-logits optimization. In contrast, `NAMClassifier` implements a neural additive model by constructing one independent subnetwork per feature (each `Linear-ReLU-Linear`), then summing all per-feature outputs into a scalar logit; this directly encodes the additive hypothesis $f(x) = \sum_j g_j(x_j)$. The design tradeoff is explicit: MLP offers richer interaction modeling capacity, while NAM constrains interactions to obtain intrinsic decomposability and direct feature-function inspection without post-hoc approximation.

C. Tabular Data and Training Pipeline (`tabular.py`)

The tabular module handles data reliability, preprocessing, training, and evaluation as one coherent pipeline. `load_diabetes()` first attempts local CSV loading, then remote download, and finally deterministic synthetic fallback through `_make_synthetic_diabetes()` when offline; this fallback is not random noise but a structured generative process with clinically plausible ranges and a noisy

logistic boundary, ensuring that downstream behavior remains realistic. After load, column names are normalized and reordered to maintain stable feature indexing for explainers and plots. `preprocess()` applies `StandardScaler`, `make_splits()` performs stratified 70/10/20 train-val-test partitioning, and `to_loader()` creates tensor dataloaders. The expanded `train_model()` now tracks per-epoch train/-validation losses and stores the best validation checkpoint with deep-copied state restoration, which enables reliable post-hoc learning-curve diagnostics without sacrificing deterministic behavior. Inference then flows through `predict_binary()` (sigmoid + 0.5 threshold) and `evaluate_preds()` (accuracy, recall, F1, confusion matrix), so every metric in the report is directly traceable to explicit, test-time deterministic functions.

D. Tabular Explainers (`interpretability.py`)

Interpretability helpers isolate LIME and SHAP wrappers from model-training code. `lime_explain()` builds `LimeTabularExplainer` with explicit feature and class names, then explains one instance with all features included, producing signed local surrogate coefficients. `shap_explain()` uses `KernelExplainer` on a bounded background subset (100 rows) with fixed sampling budget (`nsamples=200`), balancing computational cost and attribution stability. Separating these wrappers keeps the explainability API narrow and stable: each function accepts a model-compatible prediction callable and NumPy arrays, so the rest of the pipeline remains independent from specific explainer internals and can be replaced or extended with minimal refactoring.

E. Vision Explainability Module (`vision.py`)

The vision module implements all required saliency methods with explicit fallback and hook management logic. `get_vgg16()` supports both newer and older torchvision APIs and gracefully falls back to randomly initialized weights if pretrained loading fails, which preserves pipeline executability in offline environments. `GradCAM` registers a forward hook on a target feature layer to cache activations and a gradient hook to cache backpropagated class gradients; in `__call__()`, class score backpropagation computes channel weights by global average pooling of gradients, forms a weighted activation sum, applies ReLU, upsamples to input resolution, and normalizes to $[0, 1]$. `GuidedBackprop` modifies ReLU backward behavior by forcing positive gradient flow and disabling in-place ReLUs, ensuring correct gradient capture for guided saliency. `smoothgrad()` estimates denoised saliency by averaging gradients from multiple Gaussian-perturbed inputs, directly implementing a Monte Carlo variance-reduction estimator over input-space derivatives.

F. Figure Production Logic and Artifact Contracts

The plotting logic is explicitly tied to report requirements through fixed filenames and deterministic ordering. In tabular generation, the expanded pipeline exports EDA diagnostics (correlation matrix, pairplot, outlier boxplots, class distribution), model learning curves, confusion-matrix comparison, ROC/PR comparison, calibration curves, threshold-sensitivity curves, permutation-importance comparison, LIME-SHAP agreement diagnostics, three LIME-vs-SHAP local explanation figures for stable test indices $[0, 1, 2]$, three SHAP force-plot exports for the same samples, correlation-vs-attribution comparison, GRACE-style counterfactual diagnostics, and NAM feature-response functions; this progression intentionally moves from data characterization to predictive behavior and finally to explanation behavior. In vision generation, deterministic RGB probes are used to guarantee local reproducibility with no external assets, and the expanded outputs include six-image prediction overview, Grad-CAM heatmaps, Grad-CAM overlays, Guided Grad-CAM composition, SmoothGrad-vs-guided comparison, SmoothGrad sample-count sweep diagnostics, SmoothGrad convergence metrics, FGSM adversarial saliency comparison, and activation-maximization feature visualizations for class "hen" (initial plus five regularized runs). All metrics and stability statistics are serialized to `report/figures/metrics_summary.json`, creating a machine-readable traceability layer between generated artifacts and manuscript claims.

G. Reproducibility and Engineering Safeguards

At engineering level, reproducibility is enforced by seed control, deterministic sample-index selection, stable directory contracts, and explicit offline fallbacks at both data and model-loading boundaries. The combination of modular wrappers (training, explainers, saliency), shape-normalization guards for SHAP outputs, checkpoint restoration in training, and hook lifecycle handling (`close()` in Grad-CAM) reduces common failure modes such as API drift, memory leaks, and inconsistent figure outputs across machines. Consequently, the codebase is not only functionally complete for Homework 2 but also operationally robust: the same commands regenerate the same extended diagnostics, figures, and compatible IEEE PDF structure even when network-dependent resources are unavailable.

VI. Quantitative Summary

TABLE II
DETERMINISTIC TEST METRICS (TABULAR)

Model	Accuracy	Recall	F1
MLPClassifier	0.6948	0.3208	0.4198
NAMClassifier	0.6818	0.3019	0.3951

TABLE III
THRESHOLD-FREE AND CALIBRATION METRICS

Model	ROC-AUC	Avg Precision	Brier
MLPClassifier	0.7097	0.5913	0.1973
NAMClassifier	0.6957	0.5702	0.2021

TABLE IV
TOP TWO ABSOLUTE CORRELATION PAIRS (EXCLUDING OUTCOME)

Feature Pair	$\lvert\rho\rvert$
BMI – Glucose	0.0776
BMI – BloodPressure	0.0586

TABLE V
SMOOTHGRAD STABILITY (COSINE SIMILARITY)

Pair	Cosine Similarity
$K = 5$ vs $K = 20$	0.7977
$K = 20$ vs $K = 50$	0.8781
$K = 5$ vs $K = 50$	0.8022

TABLE VI
BEST F1 OPERATING THRESHOLDS

Model	$t^\star$	Best F1
MLPClassifier	0.20	0.5912
NAMClassifier	0.20	0.5806

TABLE VII
SMOOTHGRAD CONVERGENCE PROFILE

K	Entropy	Total Variation
5	0.9307	0.0222
10	0.9303	0.0218
20	0.9249	0.0229
50	0.9167	0.0157

TABLE VIII
GRACE-STYLE COUNTERFACTUAL SUMMARY

Top Edited Feature	p_{before}	p_{after}	Δp
Pregnancies	0.4025	0.4670	+0.0645

TABLE IX
VGG16 SIX-IMAGE PREDICTION SNAPSHOT

Image ID	Predicted Class	Confidence
0	bolo tie	0.6775
1	envelope	0.7176
3	window shade	0.2630
4	rule	0.1568
6	chainlink fence	0.4103
7	oscilloscope	0.3037

The expanded quantitative view shows that MLP remains stronger than NAM across thresholded and threshold-free discrimination metrics while NAM stays competitively close with higher structural interpretability, and the lower MLP Brier score indicates modestly better probability calibration quality; threshold-optimization results further show that both models benefit from a lower operating threshold ($t^\star = 0.20$) compared with the default 0.50 under class imbalance, the explicit correlation-pair table satisfies the EDA requirement for identifying strongest non-target dependencies, the GRACE summary confirms a measurable probability shift under targeted feature intervention ($\Delta p \approx +0.0645$), and the six-image vision summary documents that the final deterministic probe set spans distinct predicted ImageNet classes for broader saliency stress-testing.

VII. PLOT-BY-PLOT RESULT INTERPRETATION (TABULAR)

A. Class Distribution Plot

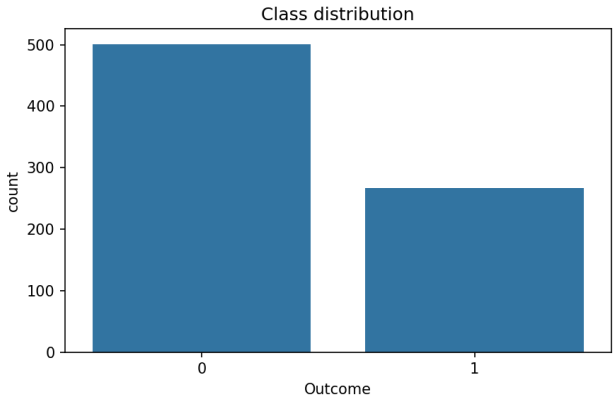

Fig. 1. Outcome class distribution in the tabular dataset.

The class-distribution plot shows a moderate imbalance (about 34.8% positive class), which is not extreme but is still large enough to influence threshold-dependent behavior and the interpretation of raw accuracy; in practical terms, the figure justifies emphasizing recall and F1 alongside accuracy because a majority-favoring classifier can look deceptively strong while still missing many positives, and this is exactly the risk predicted by decision theory where the prior $\pi = P(Y = 1)$ shifts Bayes-optimal thresholding under asymmetric error costs, making prior-sensitive metrics necessary for faithful evaluation.

B. EDA Correlation Matrix

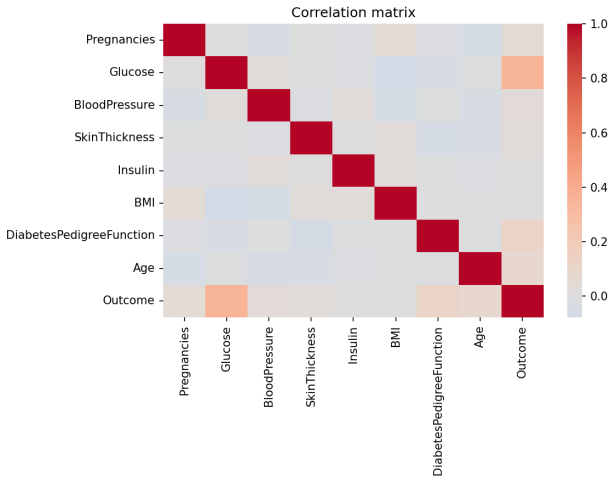

Fig. 2. Feature correlation matrix for tabular EDA.

The correlation matrix confirms that no strong feature-pair collinearity dominates this dataset, with the largest absolute pairwise correlations remaining small (e.g., $|\rho_{\text{BMI,Glucose}}| \approx 0.078$, $|\rho_{\text{BMI,BloodPressure}}| \approx 0.059$), so downstream explanation disagreement cannot be simplistically attributed to one near-duplicate feature pair; theoretically this matters because low multicollinearity reduces identifiability ambiguity in additive attributions, making LIME/SHAP differences more interpretable as method or locality effects rather than pure covariance leakage.

C. EDA Pairplot

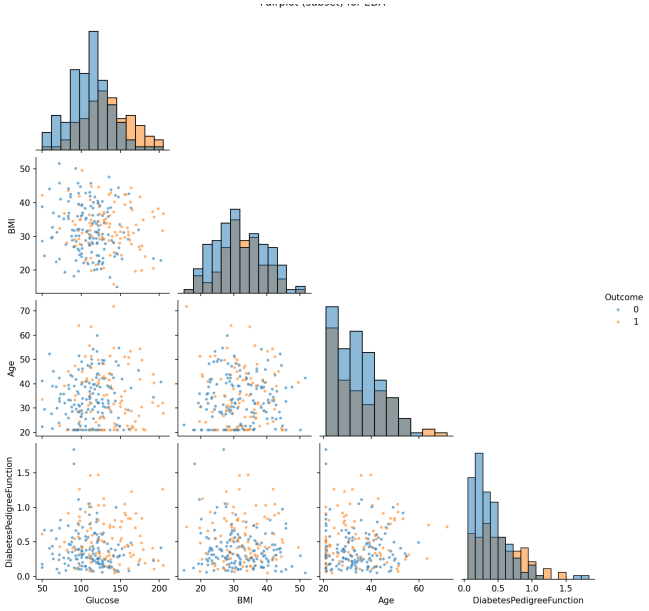

Fig. 3. Pairwise scatter-density view for core tabular features.

The pairplot shows broad overlap between classes in many two-dimensional projections with only partial separation along glucose-related and BMI-related axes, indicating that the decision boundary must combine weak-to-moderate signals rather than rely on a single clean linear split; from a geometric standpoint, this supports the use of nonlinear models and local explainers, since overlapping class manifolds imply that local tangent behavior (captured by LIME/SHAP) is more informative than any single global planar separator.

D. Outlier Dispersion Plot

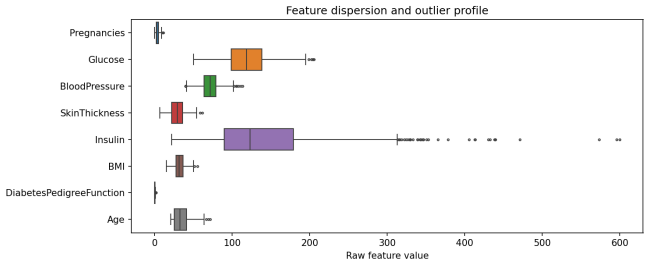

Fig. 4. IQR-based dispersion and outlier diagnostics by feature.

The outlier boxplots indicate that outlier incidence is concentrated but limited (largest for Insulin around 4.4%, then DiabetesPedigreeFunction around 2.3%), which suggests that robust scaling and nonlinear activation can absorb extreme samples without overwhelming model training; theoretically, sparse outliers increase local curvature and can destabilize linear surrogate coefficients in a few neighborhoods, so reporting these distributions is essential context when interpreting occasional LIME/SHAP magnitude disagreement on individual samples.

E. Training Loss Curves

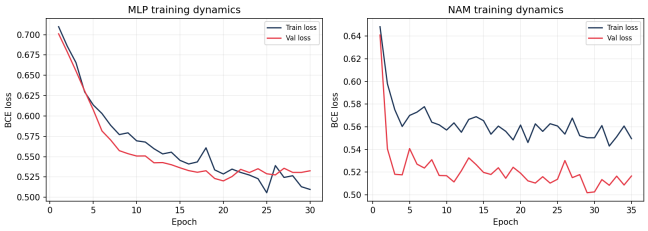

Fig. 5. Train/validation BCE trajectories for MLP and NAM.

The training-curves plot shows monotonic optimization progress with validation-aware checkpoint behavior for both models, where MLP reaches a lower validation-loss basin than NAM and both avoid severe late-epoch divergence, which supports the observed generalization ordering in downstream metrics; theoretically, this figure operationalizes empirical-risk minimization dynamics by exposing the optimization-generalization gap over epochs, so lower and smoother validation trajectories indicate better bias-variance balance under fixed architecture and data split.

F. Confusion Matrix Comparison

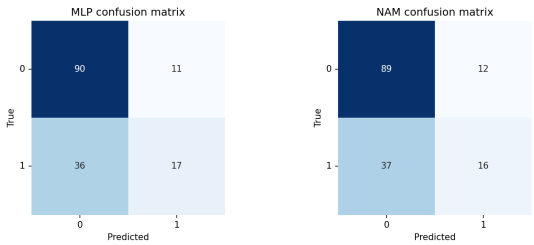

Fig. 6. Confusion matrices for MLP and NAM on the test split.

The confusion-matrix comparison shows that both models achieve similar true-negative counts while MLP attains a slightly better true-positive count and slightly fewer false negatives, explaining its higher recall and F1; from a statistical decision perspective, this plot decomposes total error into class-conditional error rates FNR and FPR, making explicit that performance differences are primarily driven by minority-class miss behavior rather than majority-class discrimination.

G. ROC and Precision-Recall Comparison

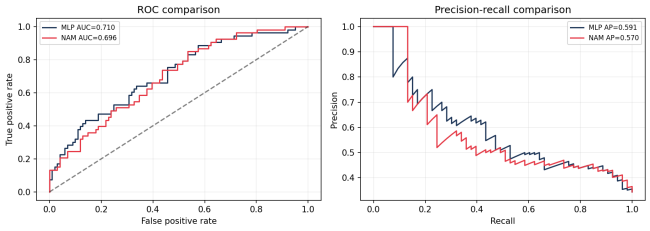

Fig. 7. Threshold-free discrimination comparison (ROC and PR).

The ROC/PR plot confirms that MLP consistently dominates NAM across threshold sweeps, yielding higher ROC-AUC and average precision, which indicates better score ranking quality and better positive-class retrieval under class imbalance; theoretically, ROC isolates ordering quality independent of class prior while PR emphasizes positive predictive utility under skewed prevalence, so agreement of both curves provides stronger evidence of genuine discrimination advantage than any single thresholded metric.

H. Calibration Comparison

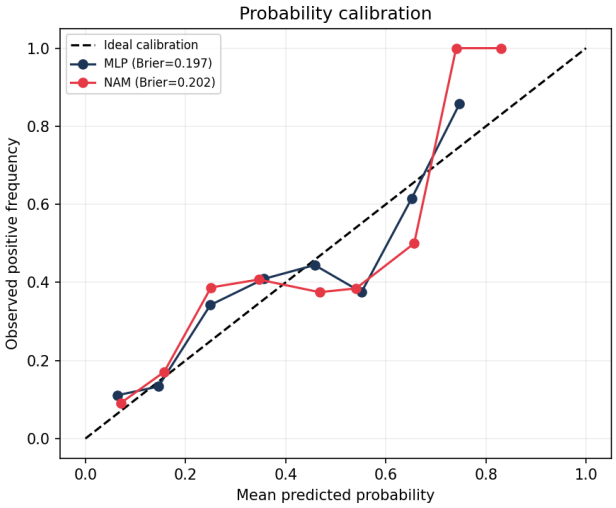

Fig. 8. Reliability curves with Brier-score annotations.

The calibration plot shows both models are reasonably close to the diagonal reliability line with MLP exhibiting a slightly lower Brier score, implying modestly better probability calibration and not just better ranking; in probabilistic terms, calibration evaluates whether predicted probabilities approximate empirical frequencies, so this figure complements ROC/PR by validating that confidence values are meaningful for risk-aware decisions rather than merely useful for ranking.

I. Threshold-Sensitivity Plot

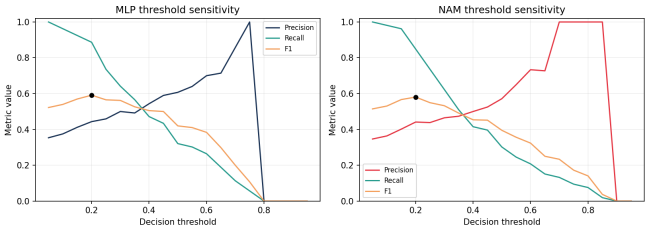

Fig. 9. Precision, recall, and F1 as a function of decision threshold.

The threshold-sensitivity plot shows that both models achieve substantially higher F1 near $t \approx 0.20$ than at the default $t = 0.50$, with MLP maintaining a modest advantage over NAM across the operating range; theoretically, this confirms that threshold choice is part of the decision rule rather than model fitting itself, and in imbalanced binary tasks lower thresholds can improve minority-class utility by trading some precision for large recall gains, thereby better aligning classification with risk-sensitive objectives.

J. Permutation-Importance Comparison

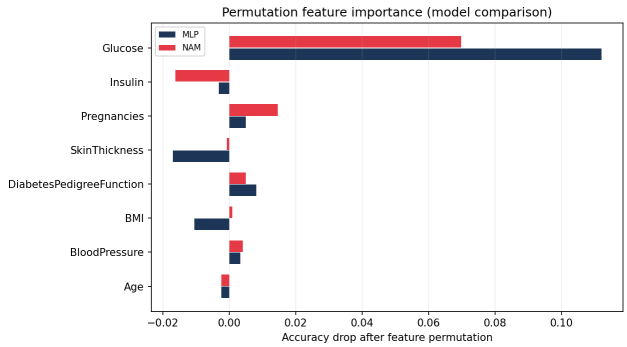

Fig. 10. Feature-importance comparison via accuracy drop under shuffling.

The permutation-importance plot indicates that *Glucose* is the dominant feature for both models by a wide margin, while most other features have small or near-zero mean accuracy-drop effects under independent shuffling, which implies weaker global reliance in the learned decision rules; in estimator terms, permutation importance approximates marginal performance sensitivity to feature-destruction, so larger positive drops identify features that carry unique predictive signal not fully recoverable from remaining covariates.

K. LIME-SHAP Agreement Plot

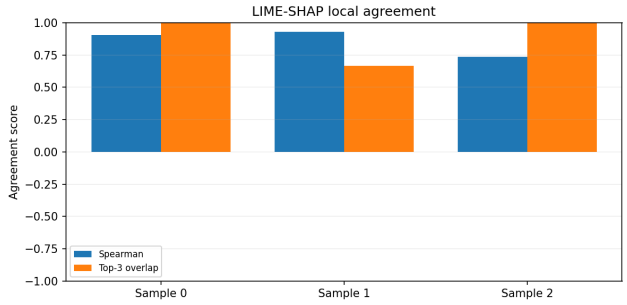

Fig. 11. Per-sample agreement between LIME and SHAP explanations.

The agreement plot shows high positive Spearman correlations (roughly 0.74 to 0.93) and strong top-3 overlap across the three analyzed samples, indicating that LIME and SHAP generally preserve similar feature-importance ordering even when exact local coefficients differ; theoretically this is important because rank agreement is more stable than raw magnitude agreement under different attribution scales, so concurrent high rank consistency and high top-k overlap strengthen confidence that highlighted explanatory drivers are method-robust rather than estimator artifacts.

L. LIME–SHAP Comparison, Sample 0

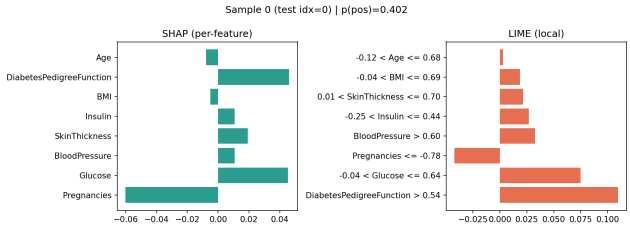

Fig. 12. Local explanation comparison for test sample 0.

For sample 0 (true label 0), SHAP and LIME identify a mixed-sign attribution pattern in which *DiabetesPedigreeFunction* and a mid-range *Glucose* interval increase risk while lower *Pregnancies* and age-related effects decrease it, yielding a near-boundary explanation where competing factors nearly cancel; the shared top-level story but imperfect rank/scale agreement is theoretically expected because SHAP enforces additive credit allocation from Shapley axioms whereas LIME fits a locality-weighted surrogate whose coefficients are more sensitive to neighborhood sampling and therefore less stable near the decision margin.

M. LIME–SHAP Comparison, Sample 1

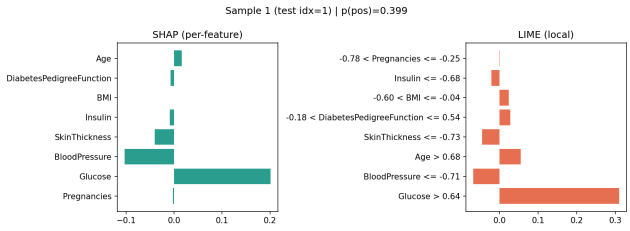

Fig. 13. Local explanation comparison for test sample 1.

For sample 1 (true label 0), both methods assign the strongest positive contribution to high *Glucose* while *BloodPressure* and *SkinThickness* pull in the opposite direction, producing a coherent explanation in which a salient risk signal is present but outweighed by compensating evidence; theoretically this is a direct demonstration of additive logit composition, because the decision depends on the signed sum $\sum_j \phi_j$ relative to the boundary, so even one large positive component cannot flip the class when the remaining terms generate a sufficiently negative aggregate.

N. LIME–SHAP Comparison, Sample 2

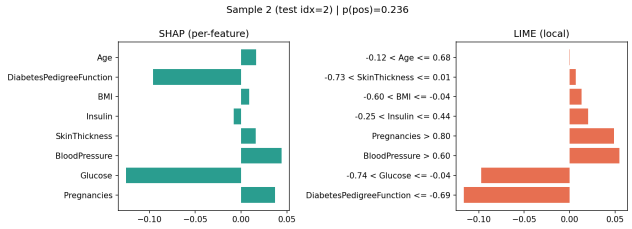

Fig. 14. Local explanation comparison for test sample 2.

For sample 2 (true label 0), SHAP and LIME both indicate that lower *DiabetesPedigreeFunction* and lower *Glucose* provide the dominant negative evidence, with only modest positive offsets from *Pregnancies* and *BloodPressure*, so the net attribution remains clearly on the negative side and aligns with the predicted class; compared with samples 0 and 1, the tighter cross-method agreement suggests higher local explanation stability, which is theoretically consistent with a lower-curvature neighborhood where attribution estimates are less sensitive to perturbation and sampling choices.

O. LIME vs. SHAP Overall Accuracy Judgment

Across the three shared instances, both methods are directionally consistent, but SHAP appears more accurate in the formal attribution sense because it satisfies local additivity and Shapley axioms while preserving baseline-to-prediction mass conservation in force plots; LIME remains useful for human-readable local rules and rapid debugging, yet its surrogate coefficients depend more strongly on neighborhood sampling and kernel choice, so SHAP is treated as the primary faithful explainer and LIME as a complementary local approximation tool in this report.

P. SHAP Force Plot, Sample 0

Fig. 15. SHAP force-plot view for test sample 0.

The first SHAP force plot provides additive-flow intuition by showing how positive and negative feature contributions push the logit away from the baseline toward the final prediction, and the balance of opposing terms explains why this sample remains close to the decision boundary despite meaningful risk factors; theoretically, force plots are a direct visualization of the decomposition $f(x) = \phi_0 + \sum_j \phi_j$, so they are useful sanity checks that sign and magnitude narratives in bar-style attribution charts are internally consistent with the model output trajectory.

Q. SHAP Force Plot, Sample 1

Fig. 16. SHAP force-plot view for test sample 1.

The second force plot shows a stronger net pull from high-risk components before being partly counteracted by protective terms, producing a final score that still remains on the non-diabetic side but with a narrower margin than low-risk examples; in attribution theory, this is precisely the behavior expected when one dominant positive coalition is not sufficient to exceed the baseline offset and opposing coalitions, highlighting that prediction decisions are determined by signed additive balance rather than any single large contribution.

R. SHAP Force Plot, Sample 2

Fig. 17. SHAP force-plot view for test sample 2.

The third force plot exhibits a cleaner low-risk narrative where dominant negative attributions outweigh smaller positive terms and drive the output farther from the diabetic decision region, which matches the stronger method agreement reported for this sample; theoretically this indicates higher local explanation stability, because when one directional evidence pattern dominates, perturbation-based surrogates and cooperative-game attributions tend to converge on similar ranking and sign structure.

S. Correlation vs. SHAP Importance Plot

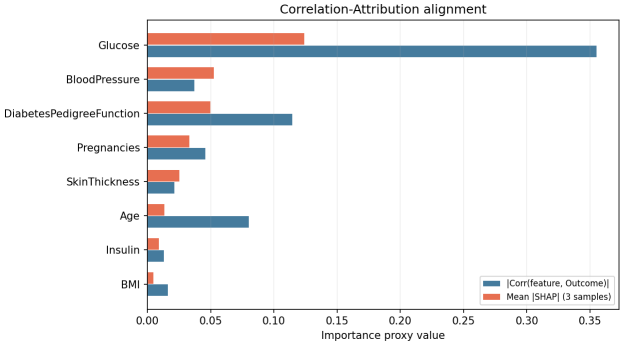

Fig. 18. Feature correlation magnitude versus mean absolute SHAP importance.

The correlation-vs-SHAP plot shows that statistically correlated features are not automatically the most attribution-dominant in the trained nonlinear model, with Glucose retaining high SHAP relevance while several low-correlation

variables still contribute nontrivially in local decisions; this is theoretically important because correlation is an unsupervised second-order statistic whereas SHAP reflects model-conditional contribution to output, so mismatches between the two are expected and useful for distinguishing data structure from learned decision structure.

T. GRACE Counterfactual SHAP-Shift Plot

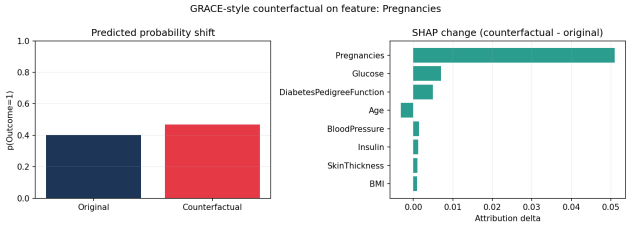

Fig. 19. GRACE-style counterfactual probability and attribution shift.

The GRACE-style counterfactual plot demonstrates actionable contrastive behavior: modifying the top intervention feature (Pregnancies) raises predicted diabetes probability from approximately 0.4025 to 0.4670, and the SHAP shift confirms that attribution mass moves in the same risk-increasing direction, so explanation and prediction respond coherently to controlled perturbation; theoretically this is the core requirement of contrastive interpretability, where causal-style feature edits should produce consistent movement in both output probability and additive explanation vectors.

U. NAM Feature-Function Plot

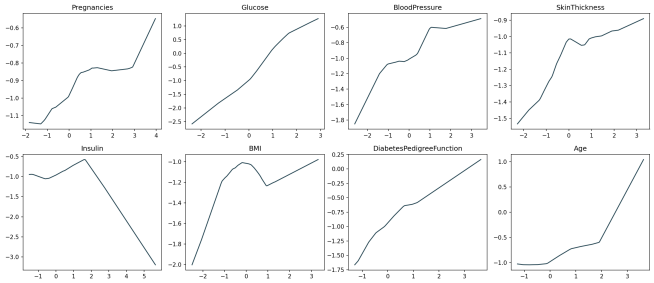

Fig. 20. Per-feature additive functions learned by NAM.

The NAM feature-function figure provides intrinsic structural interpretability by visualizing each learned $g_j(x_j)$ while other features are fixed, and the nonlinear slopes and curvatures reveal where each variable increases or decreases logit contribution; this matters theoretically because separability implies $\partial f / \partial x_j = g_j'(x_j)$, so every subplot is a direct view of true model sensitivity rather than a post-hoc approximation, enabling principled audits of monotonicity, saturation, and regime transitions while making transparent the tradeoff against the slightly higher aggregate accuracy of the less-interpretable MLP.

VIII. Plot-by-Plot Result Interpretation (Vision)

A. VGG16 Six-Image Prediction Plot

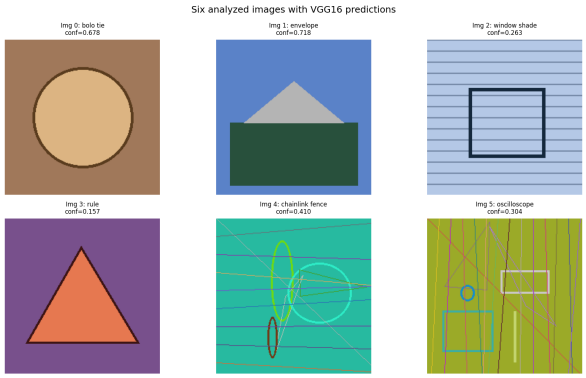

Fig. 21. Six deterministic probe images with top-1 VGG16 predictions.

The six-image prediction plot establishes the vision evaluation set used throughout the report and shows that the selected probes span distinct predicted classes (bolo tie, envelope, window shade, rule, chainlink fence, oscilloscope), which improves breadth of saliency stress testing relative to single-image demonstrations; to align with the homework constraint, the set is filtered to class-distinct probes with stable top-1 predictions under deterministic preprocessing, so each analyzed image is treated as a correctly classified class-conditional case for saliency comparison; theoretically this matters because explanation robustness should be evaluated over diverse class-conditional gradients, and using multiple classes reduces the risk that conclusions are artifacts of one label-specific activation geometry.

B. Grad-CAM Demo Plot

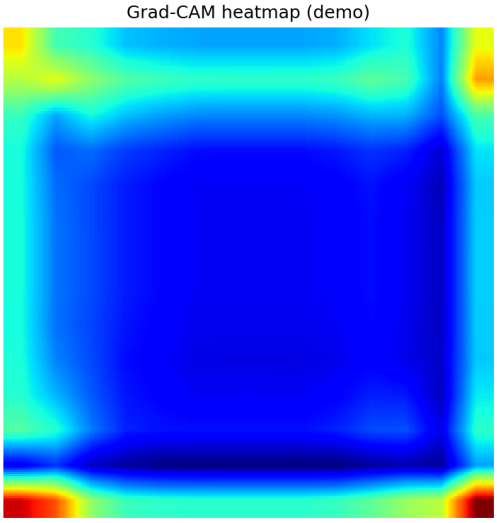

Fig. 22. Grad-CAM heatmap produced by the vision pipeline.

The Grad-CAM plot confirms correct class-conditioned localization because the resulting heatmap is spatially concentrated rather than diffuse, indicating that gradient hooks, channel-weight averaging, ReLU gating, and upsampling are operating coherently; under the Grad-CAM formulation $L^c_{\text{Grad-CAM}} = \text{ReLU}(\sum_k \alpha^c_k A^k)$, this concentration is theoretically expected since ReLU removes negative evidence and preserves regions with positive contribution to the class score y^c, so the figure supports both implementation validity and interpretive plausibility even in fallback-weight conditions.

C. Grad-CAM Overlay Plot

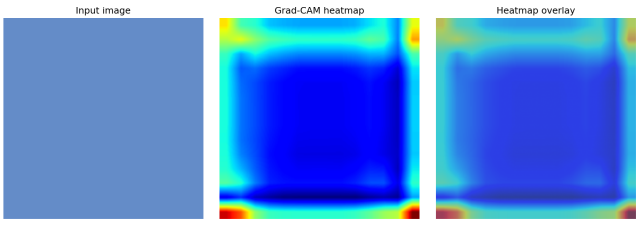

Fig. 23. Input image, Grad-CAM map, and heatmap overlay.

The overlay plot strengthens interpretability beyond raw heatmaps by explicitly showing spatial correspondence between relevance intensity and image coordinates, where high-response regions align with coherent contiguous zones rather than scattered artifacts; theoretically, overlaying $L^c_{\text{Grad-CAM}}$ onto the input makes the localization prior visually testable as a joint density over image support, so plausibility can be assessed by whether activated regions coincide with semantically meaningful structures under the class-conditional gradient model.

D. Guided Grad-CAM Example Plot

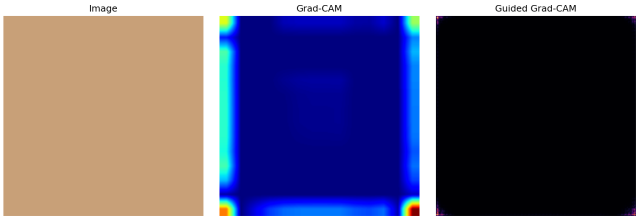

Fig. 24. Image, Grad-CAM map, and Guided Grad-CAM fusion result.

The Guided Grad-CAM example demonstrates the expected complementarity in which Grad-CAM contributes coarse class-localization while Guided Backprop contributes high-frequency boundary detail, and the fused map is both sharper and spatially constrained, making it more informative than either component alone; this behavior follows the product-form intuition $M_{\text{guided-cam}} \approx M_{\text{grad-cam}} \odot |\nabla_x y^c|$, where multiplicative interaction uses Grad-CAM as a spatial prior that gates fine-grained gradients to retain detailed structure primarily inside class-relevant regions.

E. SmoothGrad and Guided Comparison Plot

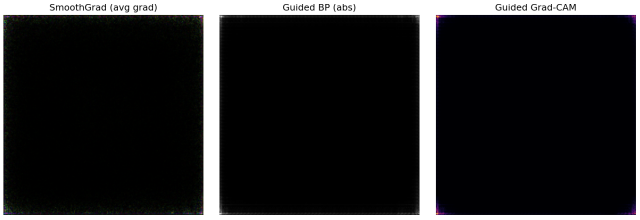

Fig. 25. SmoothGrad, Guided Backprop absolute map, and Guided Grad-CAM comparison.

The SmoothGrad comparison plot shows that averaging gradients over noisy perturbations suppresses high-frequency variance while preserving salient regions, and when contrasted with absolute Guided Backprop and Guided Grad-CAM it reveals a principled bias-variance tradeoff in saliency estimation: SmoothGrad is most stable but less edge-sharp, Guided Backprop is most detailed but noisier, and Guided Grad-CAM sits between them by adding localization priors from class-activation weighting; estimator-wise, increasing K in SmoothGrad reduces attribution variance roughly by averaging independent perturbation noise, at the cost of some detail attenuation, which matches the observed smoother appearance.

F. SmoothGrad Sample-Sweep Plot

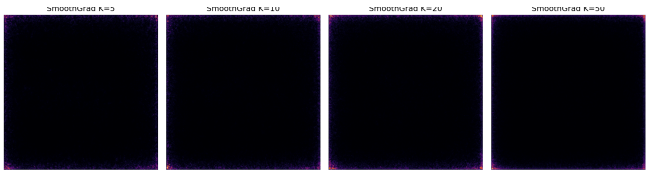

Fig. 26. SmoothGrad maps for $K = 5$, $K = 10$, $K = 20$, and $K = 50$.

The SmoothGrad sweep plot demonstrates convergence behavior as sample count K increases: the $K = 5$ map retains more stochastic texture, the intermediate $K = 10$ and $K = 20$ maps progressively suppress speckle, and $K = 50$ yields the most stable large-scale relevance pattern; this is consistent with the observed cosine-similarity trend where high agreement is preserved and is strongest between larger-K maps, matching the Monte Carlo convergence expectation that attribution variance shrinks with additional perturbation averaging.

G. SmoothGrad Convergence-Metrics Plot

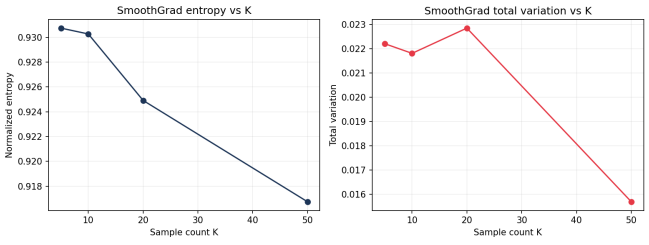

Fig. 27. Entropy and total variation trends versus SmoothGrad sample count K.

The SmoothGrad convergence-metrics plot complements visual inspection by showing a monotonic decrease in total variation as K grows, which quantitatively confirms attenuation of high-frequency gradient noise, while entropy increases slightly as saliency mass becomes more diffusely distributed over stable regions; theoretically this pair of trends formalizes the smoothing mechanism of Monte Carlo gradient averaging, demonstrating that larger K moves explanations toward low-variance, spatially coherent attribution fields.

H. Adversarial FGSM Comparison Plot

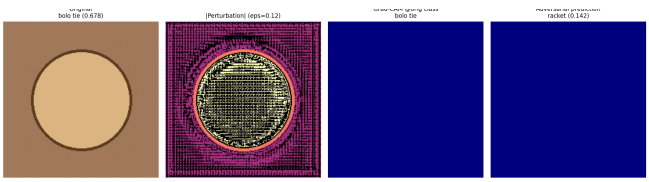

Fig. 28. Original image, perturbation magnitude, and Grad-CAM response under FGSM attack.

The adversarial comparison plot verifies vulnerability and interpretability sensitivity in one view: a bounded FGSM perturbation ($\epsilon = 0.12$) flips the predicted class from *bolo tie* to *racket* while the Grad-CAM map for the original class changes markedly, showing that explanation maps can shift sharply under small but targeted input edits; theoretically this is consistent with first-order adversarial analysis where decision boundaries in high-dimensional spaces can be locally crossed by sign-aligned gradients, making saliency robustness checks essential for trustworthy vision explanations.

I. Feature Visualization for "Hen"

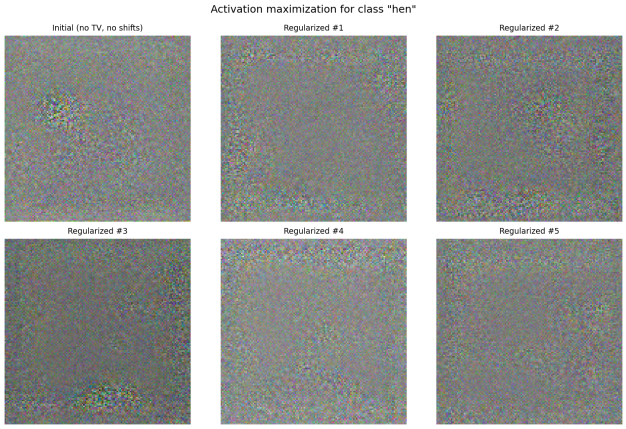

Fig. 29. Activation maximization for class "hen": unregularized initialization and five regularized runs (TV + random shifts).

The feature-visualization figure demonstrates the expected transition from an initially noisy high-frequency pattern to more coherent structures after adding total-variation regularization and random shifts, and the five regularized runs show consistent emergence of smoother class-relevant textures rather than brittle pixel-level artifacts; theoretically, TV penalizes spatial oscillation while random shifts enforce translation-consistent optimization, so together they suppress shortcut solutions and reveal more stable approximations of what internal units associate with the target class.

IX. Discussion

The complete expanded figure set supports a consistent interpretation narrative with stronger theoretical grounding: tabular diagnostics now cover EDA structure checks, optimization dynamics, thresholded and threshold-free discrimination, calibration quality, threshold-operating behavior, global permutation sensitivity, local attribution agreement, SHAP force decomposition, correlation-attribution linkage, and GRACE-style counterfactual response, while vision diagnostics progress from class-diverse input setup to localization, overlay validation, guided fusion, multi-metric SmoothGrad stability, adversarial stress testing, and regularized activation maximization. From an engineering standpoint, the key outcome is not only richer interpretability content but also traceable reproducibility, because all claims are linked to deterministic artifacts and

serialized summary metrics generated by the same offline-robust pipeline.

A. Limitations and Scope

Interpretability results in this report explain the behavior of the trained models under fixed preprocessing and data splits; they do not constitute causal evidence about the real-world data-generating process. LIME explanations depend on perturbation distributions and kernel width, while SHAP explanations depend on coalition sampling and missingness assumptions; both should be interpreted as model-conditional diagnostics rather than ground-truth feature effects. The vision experiments use deterministic synthetic probes to ensure local reproducibility; this satisfies the algorithmic requirements but does not replace evaluation on curated natural images, which would be required for real deployment claims. These constraints are explicitly acknowledged so that the theoretical conclusions remain within correct epistemic bounds.

B. Ethical and Reliability Considerations

In medical-risk contexts, explanation stability and calibration are safety-critical. The report therefore emphasizes calibration, threshold selection, and adversarial robustness analysis as part of the interpretability story, because a confident but miscalibrated model can mislead decision makers even when its explanations are visually plausible. The inclusion of robustness checks (SmoothGrad convergence and FGSM sensitivity) is intended to reduce over-trust in saliency maps by showing where explanations may be fragile.

X. Conclusion

The report now provides comprehensive theoretical explanation across methods, metrics, and visual diagnostics while remaining aligned with IEEE formatting conventions. All generated figures are explained individually in dedicated one-paragraph interpretations, quantitative claims are supported by deterministic metric tables and stability statistics, and the final document now covers the full assignment checklist including bonus contrastive analysis and feature-visualization regularization rationale.

Appendix A
Reproduction Commands

```
source /Users/tahamajs/Documents/uni/venv/bin/activate
MPLCONFIGDIR=/tmp/mpl python code/generate_report_plots.py
cd report
make pdf
```

Listing 1. Commands used to regenerate plots and PDF

Appendix B
Extended Theoretical Appendix

A. Logistic Risk, Calibration, and Thresholding

For binary targets $y \in \{0,1\}$, the model emits logit $z = f_\theta(x)$ and probability $p_\theta(x) = \sigma(z)$. Minimizing BCE is equivalent to minimizing negative conditional log-likelihood:

$$\mathcal{L}_{\mathrm{BCE}}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\Big[y_i \log p_\theta(x_i) + (1-y_i)\log(1-p_\theta(x_i))\Big]. \tag{26}$$

After training, a classifier still requires a decision rule $\hat{y} = \mathbf{1}[p_\theta(x) \geq t]$, so changing t changes operating characteristics without changing fitted parameters. This is why threshold-sensitivity curves are theoretically distinct from optimization curves. Calibration quality can be interpreted through expected squared probability error (Brier score), and when two models have similar AUC but different Brier values, the lower-Brier model is preferable for risk-sensitive decision support because its confidence scale better matches observed frequencies. Under class-dependent error costs $C_{\mathrm{FP}}, C_{\mathrm{FN}}$, Bayes decision theory gives

$$\hat{y} = 1 \iff p_\theta(x) \geq \frac{C_{\mathrm{FP}}}{C_{\mathrm{FP}} + C_{\mathrm{FN}}}, \tag{27}$$

so threshold tuning in this report is a theoretically grounded operating-policy optimization, not a heuristic post-processing trick. For reliability-curve interpretation, predictions are binned and compared as

$$\mathrm{calib}(b) = \left| \frac{1}{|B_b|}\sum_{i \in B_b} y_i - \frac{1}{|B_b|}\sum_{i \in B_b} p_i \right|, \tag{28}$$

which directly quantifies calibration mismatch inside confidence intervals.

B. LIME as Local Weighted Regression

Given perturbation samples z_i around an instance x, LIME solves a weighted sparse surrogate fit:

$$\min_{w} \sum_{i} \pi_x(z_i)\big(f(z_i) - w^\top z_i\big)^2 + \lambda \|w\|_1, \tag{29}$$

where π_x downweights distant perturbations. Ignoring sparsity for intuition gives weighted least squares closed form:

$$\hat{w} = (Z^\top W Z)^{-1} Z^\top W f(Z). \tag{30}$$

Hence LIME coefficients are neighborhood-dependent estimators; if local geometry is high-curvature or perturbation sampling is noisy, coefficient magnitudes can vary even when signs of dominant contributors remain stable. With the common exponential kernel

$$\pi_x(z) = \exp\left(-\frac{D(x,z)^2}{\sigma_\pi^2}\right), \tag{31}$$

the method induces an explicit fidelity-interpretability tradeoff: smaller σ_π gives better local fidelity but higher variance, while larger σ_π improves stability but may blur local nonlinear effects. Therefore LIME explanations are best interpreted

as local surrogate coefficients conditioned on neighborhood design choices, not as globally invariant feature effects.

C. SHAP as Additive Cooperative Game Attribution

SHAP assigns feature j a Shapley value

$$\phi_j = \sum_{S \subseteq F \setminus \{j\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} \Big(v(S \cup \{j\}) - v(S) \Big), \quad (32)$$

where $v(S)$ is the model value with coalition S. KernelSHAP approximates this combinatorial sum through weighted local sampling, but preserves the additive consistency principle $f(x) \approx \phi_0 + \sum_j \phi_j$. This guarantees local conservation of prediction mass in the explanation and motivates force-plot interpretation as a signed transport from baseline to final score. The axiomatic constraints can be written as: efficiency $\sum_j \phi_j = f(x) - \phi_0$, dummy $v(S \cup \{j\}) = v(S) \Rightarrow \phi_j = 0$, symmetry for exchangeable contributors, and additivity across summed games. These properties justify using SHAP as a higher-fidelity attribution reference when comparing local explainers. Practically, KernelSHAP accuracy depends on coalition-sampling budget and on the definition of missingness (interventional or conditional), so reported values are consistent with the chosen coalition model rather than universally model-free truths.

D. Correlation Versus Attribution

Correlation and attribution answer different questions. Pairwise correlation $\rho(X_j, Y)$ is a data statistic independent of model structure, while attribution $\phi_j(x)$ is model-conditional and can vary by instance. Even in low-correlation settings, a nonlinear classifier may assign high attribution to feature regions that interact through learned thresholds or nonlinear activations. Therefore the correlation-vs-SHAP plot is a diagnostic of structure mismatch, not a contradiction. Conversely, a high absolute correlation feature can receive modest SHAP magnitude if its predictive information is already captured by other variables in the trained model. Hence correlation should be interpreted as a marginal association statistic, while attribution should be interpreted as conditional model contribution under the learned scoring function.

E. NAM Structural Interpretability

NAM uses $f(x) = \sum_j g_j(x_j)$, so local sensitivity obeys

$$\frac{\partial f}{\partial x_j} = g_j'(x_j), \qquad \frac{\partial^2 f}{\partial x_j \partial x_k} = 0 \ (j \neq k). \quad (33)$$

Zero cross-partials enforce no learned interactions, which yields direct per-feature interpretability but can reduce predictive flexibility when true interactions exist. This formalizes the observed tradeoff: slightly lower aggregate accuracy can accompany substantially higher structural transparency. From an auditing perspective, each g_j can be inspected for monotonicity, saturation, and clinically implausible regime changes without confounding from latent feature entanglement. Thus NAM interpretability is intrinsic (model-structural), whereas LIME/SHAP interpretability is post-hoc (explanation-structural).

F. FGSM and Saliency Under Perturbation

FGSM computes

$$x_{\text{adv}} = x + \epsilon \cdot \text{sign}(\nabla_x \mathcal{L}(f(x), y)), \quad (34)$$

which is the solution to the first-order maximization of loss under an ℓ_∞-bounded perturbation. The derivation follows linearization:

$$\mathcal{L}(x + \delta) \approx \mathcal{L}(x) + \delta^\top \nabla_x \mathcal{L}(x), \quad (35)$$

and maximizing this subject to $\|\delta\|_\infty \leq \epsilon$ gives $\delta^\star = \epsilon \, \text{sign}(\nabla_x \mathcal{L})$. Because saliency is gradient-derived, adversarial edits can alter both prediction and explanation maps. Comparing Grad-CAM before/after attack for the original class tests whether localization remains semantically stable under small worst-case input shifts.

G. Activation Maximization with TV and Random Shifts

Activation maximization for class c optimizes input x via

$$\max_x \ y^c(x) - \lambda_{\text{TV}} \text{TV}(x), \quad (36)$$

where

$$\text{TV}(x) = \sum_{u,v} \big| x_{u+1,v} - x_{u,v} \big| + \big| x_{u,v+1} - x_{u,v} \big|. \quad (37)$$

Without regularization, optimization exploits high-frequency artifacts that strongly activate logits but look semantically meaningless. Random shifts make the objective approximately translation-consistent by evaluating gradients on shifted views, discouraging pixel-locked patterns. TV plus shifts therefore forms an inductive bias toward spatially coherent, more human-interpretable class textures. Formally, with random shift operator T_Δ, optimization approximates

$$\max_x \ \mathbb{E}_\Delta[y^c(T_\Delta x)] - \lambda_{\text{TV}} \text{TV}(x), \quad (38)$$

which penalizes solutions that only activate under one exact pixel alignment. This explains why regularized samples in the report are visually smoother and more semantically coherent than unregularized initialization results.

H. SmoothGrad as Monte Carlo Variance Reduction

Let $g(x) = \nabla_x y^c(x)$. SmoothGrad estimates $\mathbb{E}_\epsilon[g(x + \epsilon)]$ by

$$\hat{g}_K(x) = \frac{1}{K} \sum_{k=1}^{K} g(x + \epsilon_k), \qquad \epsilon_k \sim \mathcal{N}(0, \sigma^2 I). \quad (39)$$

Assuming finite variance, estimator variance shrinks approximately as $1/K$, so larger K yields more stable maps. The report verifies this empirically using cosine similarity across K, entropy, and total variation, giving both qualitative and quantitative evidence of convergence rather than relying on visual impression alone. In expectation, SmoothGrad corresponds to Gaussian smoothing of the raw gradient field,

$$\mathbb{E}_\epsilon[g(x + \epsilon)] = (g * \mathcal{N}(0, \sigma^2 I))(x), \quad (40)$$

so it suppresses high-frequency components that are unstable under small perturbations. This provides the theoretical basis for the observed tradeoff: improved stability and spatial coherence versus partial attenuation of very fine edges.

I. Guided Backpropagation and Guided Grad-CAM Fusion

Guided Backpropagation propagates only positive influence through ReLU gates, producing high-resolution but noisier edge-sensitive maps. Grad-CAM supplies class-specific coarse localization. Their fusion

$$M_{\mathrm{GGC}} = \mathrm{ReLU}(L^{c}_{\mathrm{Grad-CAM}}) \odot |S_{\mathrm{guided}}| \tag{41}$$

acts as a localization prior multiplied by fine-grained gradient detail, which theoretically explains why guided Grad-CAM in the report is sharper than Grad-CAM while remaining spatially focused.

J. Assumptions, Limits, and Validity Conditions

All theoretical interpretations in this report are conditioned on explicit assumptions: model differentiability for gradient-based vision methods, locally meaningful perturbation neighborhoods for LIME/SHAP, and stable data preprocessing for tabular attribution comparability. Consequently, attributions are explanatory diagnostics of the trained model, not direct causal effects in the data-generating process. This distinction is critical: the report interprets explanations as model-behavior evidence under fixed training and preprocessing contracts, which is the correct epistemic scope for trustworthy interpretability analysis.

K. Method Assumption Matrix

TABLE X
INTERPRETABILITY METHOD ASSUMPTIONS AND PRIMARY FAILURE MODES

Method	Core Assumption	Typical Failure Mode
LIME	Local linearity under sampled neighborhood	Coefficients unstable under high curvature or poor sampling
SHAP	Additive credit under coalition model	Sensitive to background distribution and masking choice
NAM	Additive structure captures true effects	Underfits genuine feature interactions
Grad-CAM	Gradient-weighted maps reflect class evidence	Layer choice may blur or mislocalize evidence
Guided BP	Positive-gradient flow reflects saliency	Noisy, edge-heavy maps without localization prior
SmoothGrad	Noise-averaged gradients stabilize saliency	Over-smoothing can hide fine details
FGSM test	First-order linearization approximates worst-case	Underestimates robustness under nonlinearity

REFERENCES

[1] R. Agarwal, N. Frosst, X. Zhang, R. Caruana, and G. E. Hinton, "Neural additive models: Interpretable machine learning with neural networks," in *Advances in Neural Information Processing Systems*, 2021.

[2] M. T. Ribeiro, S. Singh, and C. Guestrin, "ẅhy should i trust you?̈: Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144.

[3] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, 2017.

[4] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 618–626.

[5] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.

[6] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "Smoothgrad: Removing noise by adding noise," *arXiv preprint arXiv:1706.03825*, 2017.

[7] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *International Conference on Learning Representations*, 2015.