Empirical Study of Model Complexity: Comparative Analysis of Supervised Learning Models on Structured Datasets

Venkata Krishna Rayalu Garapati, Ayush Vineet Jain, Varun Agarwal
Khoury College of Computer Sciences
Northeastern University
Boston, MA, USA
garapati.v@northeastern.edu, vineetjain.a@northeastern.edu, agarwal.varun2@northeastern.edu

***Abstract*—Selecting models for structured (tabular) data requires balancing three competing factors: predictive accuracy, robustness to data imperfections, and computational cost. This study presents a controlled, empirical comparison of common supervised learners—Logistic Regression, Decision Tree, Random Forest, and kernel methods (SVM/SVR)—across three UCI datasets: Adult Income (binary classification), Forest Cover Type (multi-class classification), and California Housing (regression). A standardized pipeline includes one-hot encoding and scaling where appropriate, five-fold (stratified) cross-validation, paired t-tests, and a robustness suite comprising feature ablation, Gaussian noise injection ($\sigma = 0.1$–0.3), and training-size sensitivity. Advanced analyses cover learning curves, hyperparameter sweeps, and compute profiling. On Adult Income, SVM attained the top mean CV accuracy (0.8526) with AUC $\approx$ 0.90 also reached by Random Forest and Logistic Regression. On Forest Cover Type, Random Forest led (accuracy 0.768, AUC 0.942); full SVM runs were omitted due to prohibitive training time at this scale. For California Housing, Random Forest achieved the highest test R^2 (0.773), ahead of SVR (0.728) and linear baselines (0.576). Robustness tests show ensembles deliver the best absolute performance under perturbations, while SVR exhibits strong relative robustness; kernel methods, however, incur substantial tuning and compute overhead. Overall, Random Forest offers the most favorable accuracy–robustness–cost trade-off, while linear models remain attractive when interpretability and latency dominate. The evaluation framework is reusable for future studies beyond accuracy-only comparisons.**

***Index Terms*—Model complexity, robustness, cross-validation, feature ablation, noise injection, learning curves, hyperparameter sensitivity, computational efficiency, Random Forest, Support Vector Machine, Support Vector Regression, Logistic Regression, Decision Tree, Adult Income, Forest Cover Type, California Housing, tabular data.**

I. Introduction

Machine learning algorithms are now essential tools for gleaning insights from structured data in a wide range of domains—from risk assessment in finance to environmental monitoring. As the explosion of advanced models holds the promise of better predictive performance, practitioners are confronted with an elementary problem: how do the algorithms behave when applied in the wild to noisy measurements, incomplete features, and scarce training samples? This discrepancy between laboratory benchmarks and production reliability inspires our rigorous examination of algorithm robustness.

Algorithm selection is not just about basic accuracy comparisons. Production machine learning models need to deal with sensor breakdowns that destroy important features, measurement noise that introduces errors, and budgeted training data acquisition. But most comparative analyses judge models based on clean, complete data only, providing practitioners with no advice for these real-world scenarios. In addition, the computational expense of complicated models—frequently orders of magnitude greater than more straightforward counterparts—enforces further deployment limitations that accuracy-only assessments overlook.

This work fills these lacunae with a rigorous empirical comparison of supervised learning methods on three varied UCI datasets: Adult Income (binary classification), Forest Cover Type (multi-class classification), and California Housing (regression). We rigorously analyze how model complexity affects not just predictive accuracy but also robustness to perturbations and computational cost. Our experimental setup probes the hypothesis that ensemble approaches exhibit greater resilience to data quality defects than their less complex counterparts, and measures the attendant computational trade-offs.

Our main goals are threefold: (1) provide baseline performance comparisons with rigorous cross-validation and statistical testing, (2) measure robustness with controlled experiments consisting of feature ablation, noise injection, and variation in training size, and (3) profile computationally intensive requirements to guide deployment. By considering models across the spectrum of complexity—from linear techniques to ensemble trees and kernel machines—we seek to offer evidence-based guidance for choosing algorithms in terms of multiple operational constraints.

This paper has several important contributions to the machine learning literature. First, we introduce a general robustness testing framework that can be applied to classification and regression problems. Second, we offer quantitative proof that ensemble techniques, especially Random Forest, provide the optimal tradeoff of performance versus robustness for

structured data problems. Third, we show that less complex models still have considerable value when interpretability or computational efficiency are more important. Finally, our analysis reveals unexpected findings, such as SVR's superior relative robustness despite Random Forest's better absolute performance, highlighting the importance of comprehensive evaluation beyond accuracy metrics.

The rest of this paper follows this structure: Section II states the three datasets and their properties, Section III explains our experimental protocol, Section IV gives results and analysis for all tasks, Section V discusses findings and implications, and Section VI concludes with implications and future work.

II. Datasets

This study is structured into three tasks, one task for each dataset from the UCI Machine Learning Repository. The datasets were selected for considerations of size, variety, and applicability to real-world situations, thus allowing a systematic study of model complexity over various regimes of data.

A. Task 1: Adult Income (Classification)

The **Adult Income** dataset has 48,842 samples describing 14 attributes for individuals in terms of demographic, educational, occupational, and financial characteristics. The prediction task is binary classification: determining whether an individual really makes more than \$50K a year.

Of the 14 attributes, eight are categorical, such as *workclass*, *education*, and *native-country*, while the others are numerical, like *age*, *capital-gain*, and *hours-per-week*. The target variable (*income*) is highly imbalanced, where only about 24% of individuals earn more than \$50K.

TABLE I
Summary of Adult Income Dataset

Characteristic	Value
Samples	48,842
Features	14 (8 categorical, 6 numerical)
Target Variable	Income: $\geq$\$50K or $\leq$\$50K
Positive Class Proportion	$\sim$24% ($>$\$50K)

TABLE II
Attributes in the Adult Income Dataset

Type	Attributes
Categorical	workclass, education, marital-status, occupation, relationship, race, sex, native-country
Numerical	age, fnlwgt, education-num, capital-gain, capital-loss, hours-per-week

B. Task 2: Forest Cover Type

The Covertype dataset comprises **581,012 samples** (observations), each representing a **30×30 m** forest cartographic cell in Roosevelt National Forest, Colorado. It contains **54 features**, including **10 continuous attributes** (e.g., elevation, aspect, slope, distances to hydrology, roads, fire ignition points, and hillshade indices at 9 am, noon, and 3 pm), and **44 binary encoded categorical features** representing **4 wilderness areas** and **40 soil types**. The task is **multi-class classification**, predicting **7 forest cover types** such as Spruce/Fir, Lodgepole Pine, and Douglas-Fir. No missing values are present in the dataset.

TABLE III
Summary of Forest Cover Type Dataset

Characteristic	Value
Samples	581,012
Features	54 (10 numerical, 44 binary categorical)
Target Variable	Forest Cover Type (7 classes)
Positive Class Proportion	Multi-class (class distribution imbalanced)

C. Task 3: California Housing (Regression)

The **California Housing** dataset contains **20,640 samples** from the 1990 California census, with each sample representing a census block group. The task is **regression**: predicting the median house value for California districts in hundreds of thousands of dollars.

The dataset includes **8 numerical features** capturing demographic characteristics (median income, house age, average rooms/bedrooms, population, occupancy) and geographic coordinates (latitude, longitude). The target variable ranges from \$14,999 to \$500,001, with notable geographic clustering where coastal properties command premium prices.

TABLE IV
Summary of California Housing Dataset

Characteristic	Value
Samples	20,640
Features	8 (all numerical)
Target Variable	Median house value (\$100,000s)
Target Range	\$0.15 - \$5.00 (\$100,000)

TABLE V
Attributes in the California Housing Dataset

Type	Attributes
Numerical	MedInc, HouseAge, AveRooms, AveBedrms, Population, AveOccup, Latitude, Longitude

III. Methodology

A. Task 1: Adult Income (Classification)

1) Pre-processing: The Adult Income data set comprises 48 842 observations and fourteen attributes—eight categorical and six numerical. Missing values are denoted by "?",

appearing most often in `workclass`, `occupation`, and `native-country`. Because these blanks form a small slice of the corpus, we applied a **row-wise deletion** rule: any record with at least one "?" was discarded. After this sweep, $45\,222$ complete rows remained for analysis.

Categorical attributes (for instance, `workclass` and `education`) were expanded via **OneHotEncoder**, producing a high-dimensional but sparse binary matrix. Such encoding is beneficial for linear separators such as Logistic Regression, yet it deliberately stresses distance-based models like the RBF-kernel SVM. By contrast, numeric fields—`age`, `capital-gain`, `capital-loss`, and `hours-per-week`—were scaled to zero mean and unit variance with **StandardScaler**. Scaling is indispensable for gradient-driven solvers and margin maximisation, while tree-based learners are insensitive to raw scale.

Class imbalance is notable: roughly $24\,\%$ of individuals earn more than \$50 000. All data partitions therefore used *stratified* sampling. The primary split holds 80 % for training and 20 % for testing, and every cross-validation fold preserves that minority/majority ratio exactly.

2) Model selection and rationale: To study the complexity–generalisation trade-off, four classifiers were chosen:

- **Logistic Regression** — linear baseline; coefficients are directly interpretable and expose bias-dominated limits.
- **Decision Tree** — axis-aligned, non-linear model; restricting maximum depth curbs variance and shows when moderate non-linearity suffices.
- **Random Forest** — 100 bootstrap trees with $\sqrt{d}$ feature subsampling; bagging should dampen single-tree variance and raise robustness.
- **Support Vector Machine (RBF)** — high-capacity margin optimiser; a grid search over $C \in \{0.1, 1, 10\}$ and $\gamma \in \{10^{-3}, 10^{-2}, 10^{-1}\}$ explores whether accuracy gains justify computational cost.

Naïve Bayes and k-NN were excluded (poor scaling in one-hot spaces), and deep architectures were set aside to preserve interpretability on a moderate-sized, tabular data set.

3) Experimental design:

Core experiments: Each learner was subjected to **five-fold stratified cross-validation**. We recorded Accuracy, Precision, Recall, F_1, and ROC–AUC, averaged across folds. To assess statistical weight, fold-wise scores for every model were paired with those of Logistic Regression and evaluated using a two-sided t-test; improvements with $p < 0.05$ were deemed significant.

Robustness analysis: Model stability was examined under four controlled disturbances:

- *Feature importance.* We extracted coefficients (LogReg), Gini-gain importances (Tree), and permutation scores (Forest) to identify key predictors.
- *Feature ablation.* After dropping each model's two most influential attributes, we re-trained and measured the accuracy delta; larger drops signal heavier dependence.
- *Noise injection.* Gaussian noise with $\sigma \in \{0.1, 0.2, 0.3\}$ was added to numeric columns to mimic measurement error. Bagging ensembles were expected to degrade least.
- *Training-size sensitivity.* Models were fitted on 25 %, 50 %, 75 %, and 100 % of the training data, revealing how quickly each learner saturates its capacity. Finally, we report a normalised robustness score in each task, defined as the average (across noise levels, feature-ablation runs, and training-size subsets) of the model's metric divided by its own baseline, yielding a 0–1 scale that enables fair comparison across perturbations.

Advanced extensions: Three deeper probes completed the study:

- *Learning curves.* Plots of training versus validation accuracy exposed bias (LogReg under-fits) and variance (SVM may over-fit).
- *Hyper-parameter sweeps.* Decision-tree depth $\{2, 5, 10, 15\}$; forest size $\{50, 100, 200\}$; the C, γ mesh above. Abrupt cliffs indicate tuning fragility.
- *Compute profiling.* Wall-clock times were logged; the SVM's roughly cubic solver was contrasted with the Random Forest's embarrassingly parallel implementation, contextualising resource–performance trade-offs.

4) Alignment with research objectives: The pipeline above links model complexity to three axes-predictive power, statistical confidence, and resilience under perturbation-with another axis going to computational overhead. The metrics, significance tests, and stress scenarios, put together, form a decision template for classifier selection on imbalanced and structured data.

B. Task 2: Forest Cover Type (Classification)

1) Pre-processing: The Forest Cover Type dataset has $581\,012$ observations, each representing a set of area in Roosevelt National Forest, Colorado. It contains 54 features — 10 continuous variables (e.g., elevation, slope, aspect, distances to hydrology, roads, fire points, and hillshade indices) and 44 binary indicators for wilderness areas and soil types. No missing values are present.

Since the attributes are numeric, a **StandardScaler** transformation was applied to zero normalise features. This benefits gradient-based and margin-based models, while tree-based learners remain scale-invariant. The dataset is multi-class with 7 cover types, but the distribution is imbalanced; thus, all splits used *stratified* sampling. The primary split holds 80% for training and 20% for testing, with every cross-validation fold preserving class proportions.

2) Model selection and rationale: To study model complexity versus generalisation:

- **Logistic Regression** — multi-class baseline via one-vs-rest; interpretable and highlights linear separability limits.
- **Decision Tree** — axis-aligned splits with depth control; exposes non-linear structure without ensembles.
- **Random Forest** — 100 estimators with $\sqrt{d}$ feature subsampling; mitigates single-tree variance and improves robustness.

SVM with RBF kernel was tested in preliminary runs but omitted from final results due to prohibitive computation on the large dataset.

3) SVM Computational Feasibility: SVM with RBF kernel was excluded from the Forest Cover Type analysis due to prohibitive computational requirements. RBF-SVM training complexity scales quadratically to cubically with sample size, making it computationally intractable for the 581,012-sample dataset. The quadratic memory requirements for kernel matrix storage and the intensive iterative optimization process would require extensive computation time that exceeds our project constraints. Even with GPU acceleration, the kernel computations and sequential optimization steps in SVM do not parallelize efficiently enough to make training feasible within reasonable time limits. Given our focus on comprehensive robustness analysis across multiple models, we prioritized thorough evaluation of computationally feasible algorithms over partial SVM results that would compromise the completeness of our experimental protocol.

4) Experimental design:

Core experiments: Each model underwent **5 fold cross-validation**. We computed Accuracy, Macro Precision, Macro Recall, Macro F_1, and Macro ROC–AUC, averaged across folds. We compared these results for each fold to compare to Logistic Regression using a paired two-sided t-test; $p < 0.05$ denoted significance.

Robustness analysis: Following the Adult Income protocol, we examined:

- *Feature importance.* Coefficients for Logistic Regression, Gini importance for tree-based models.
- *Feature ablation.* Dropped top two most important features for each model and measured the accuracy change.
- *Noise injection.* Gaussian noise ($\sigma = 0.1, 0.2, 0.3$) applied to numeric features; observed degradation patterns.
- *Training-size sensitivity.* Re-trained on 25%, 50%, 75%, and 100% of training data to examine learning saturation.

Advanced extensions: Additional experiments included:

- *Learning curves.* Training versus validation accuracy plotted against sample size to detect bias–variance patterns.
- *Hyper-parameter sweeps.* Depths $\{5, 10, 15\}$ for Decision Tree; estimator counts $\{50, 100, 200\}$ for Random Forest.
- *Compute profiling.* Recorded wall-clock training times to contrast the efficiency of single versus ensemble models.

5) Alignment with research objectives: This pipeline mirrors the Adult Income methodology, linking model complexity to predictive accuracy, statistical reliability, and robustness under perturbations, with an additional axis of computational efficiency. The combination of stratified evaluation, statistical testing, and controlled robustness scenarios yields a principled basis for selecting classifiers on large, imbalanced, structured datasets.

C. Task 3: California Housing (Regression)

1) Pre-processing: The California Housing data set consists of 20 640 observations from the 1990 census, each from a block group in California. Eight numeric attributes record demographic and geographical attributes—there are no categorical variables. Luckily, the data set arrived ready for use with zero missing values, precluding the use of imputation techniques.

All the features were normalised to zero mean and unit variance using **StandardScaler**. This is a critical step for gradient-descent based optimizers as well as distance-sensitive models such as SVR, but not for tree-based algorithms, which are invariant to such transformations. We observed extreme right-skew in many of the features: `Population` had extreme right-skew (9.98), so did `AveOccup` (18.35), and the target variable had moderate positive skew (1.05). In spite of these quirks in the distributions, we decided against log-transformation to preserve interpretability.

The main partition reserved 80% for training and 20% for testing using random split—stratification was not an option due to the continuous target. Cross-validation used plain k-fold instead of stratified versions.

2) Model selection and explanation: Five regressors across the bias-variance trade-off were chosen:

- **Linear Regression** — unregularized baseline; coefficients are explicit feature contributions and linear assumptions.
- **Ridge Regression** — L_2-regularized version with $\alpha = 1.0$; shrinkage should help with generalization while maintaining interpretability.
- **Decision Tree** — recursive partitioning with depth limit of 10; detects axis-aligned non-linearities without ensemble cost.
- **Random Forest** — 50 trees with depth limit of 10 and $\sqrt{d}$ feature subsampling; bootstrap aggregation aims for variance reduction.
- **Support Vector Regression** — RBF kernel with $C = 1.0$, $\epsilon = 0.1$; investigates whether kernel tricks are worth computational cost.

Neural networks were skipped to ensure model explainability, and boosting techniques were left for future exploration.

3) Experimental design:

Core experiments: Every regressor was subjected to **five-fold cross-validation** with MSE, MAE, and R^2 calculated per fold. Mean measures estimated central performance and standard deviations stability. Fold-wise R^2 scores were contrasted with Linear Regression via paired t-tests; significance levels were kept at $p < 0.05$.

Robustness analysis: The robustness of regression required modified protocols:

- *Feature importance.* Absolute coefficients (linear models), Gini decreases (trees), and mean decrease impurity (forests) determined key predictors.
- *Feature ablation.* Top two features per model eliminated and R^2 degradation recorded; steeper declines suggest brittleness.
- *Noise injection.* Gaussian perturbations with $\sigma \in \{0.1, 0.2, 0.3\}$ times feature standard deviations injected measurement uncertainty.

- *Training-size sensitivity.* Models trained on 25%, 50%, 75%, and 100% subsets disclosed data efficiency and points of saturation.

Advanced extensions: Three other investigations completed the analysis:

- *Learning curves.* Train vs. test R^2 with sample sizes detected under/overfitting inclinations.
- *Hyper-parameter sensitivity.* Ridge $\alpha \in \{0.01, 0.1, 1, 10, 100\}$; tree depths $\{5, 10, 15, 20\}$; forest sizes $\{25, 50, 100\}$; SVR grids over C and γ.
- *Scalability profiling.* Train times by data sizes (1k, 5k, 10k, full) measured computational scaling behavior.

4) Alignment with research objectives: This pipeline of regressions is analogous to the classification approach, exploring how model complexity is traded off against predictive accuracy, statistical confidence, perturbation robustness, and computational cost. The spatial nature of housing data specifically challenges linear models' ability to model spatial structures, so this makes a great testing ground for our complexity-robustness hypothesis.

IV. Results and Analysis

A. Task 1: Adult Income (Classification)

1) Core Experiments: In order to obtain the baselines, each classifier was evaluated with a **five-fold stratified cross-validation**. Table VI shows the fold-level accuracies and their means. Support Vector Machines (SVM) produced the highest mean accuracy, 0.8526, while Logistic Regression followed closely at 0.8484.

TABLE VI
Five-fold cross-validation accuracy on the Adult Income data set

Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
Logistic Regression	0.851	0.846	0.850	0.843	0.852	**0.848**
Decision Tree	0.843	0.839	0.843	0.841	0.851	**0.843**
Random Forest	0.845	0.849	0.848	0.843	0.844	**0.846**
SVM	0.853	0.851	0.857	0.848	0.855	**0.853**

The paired t-tests in Table VII compare each candidate to Logistic Regression. SVM's edge is statistically reliable ($p = 0.0097$), whereas Random Forest's lift is not.

TABLE VII
Paired t-test results versus Logistic Regression

Comparison	t-statistic	p-value
Decision Tree vs. Log. Reg.	3.28	0.030
Random Forest vs. Log. Reg.	1.42	0.229
SVM vs. Log. Reg.	-4.64	0.010

ROC curves in Fig. 1 reveal strong separability, with Random Forest and Logistic Regression each approximately achieving 0.90 AUC.

Table VIII extends the comparison with precision, recall, F_1, and AUC on the held-out test set. Logistic Regression and SVM balance precision and recall well; Decision Tree and Random Forest lag mainly in recall.

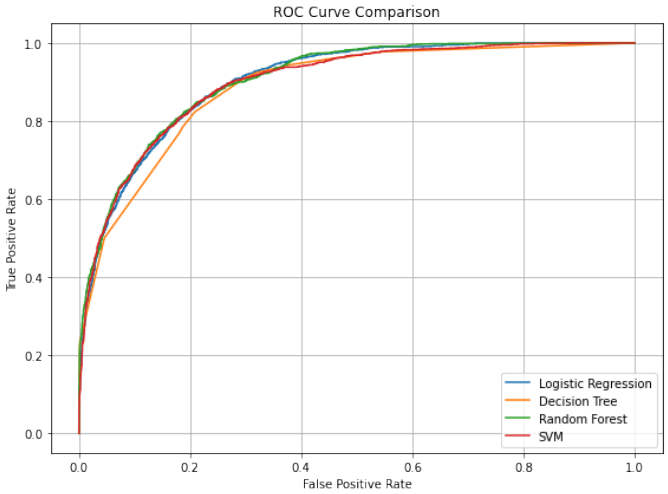

Fig. 1. ROC Comparison shows the relative rankings of the models with SVM and Random Forest achieving high separability (AUC close to ≈ 0.90), closely followed by Logistic Regression; Decision Tree lags slightly in terms of recall.

TABLE VIII
Test-set performance metrics

Model	Accuracy	Precision	Recall	F_1	AUC
Logistic Regression	0.847	0.735	0.605	0.663	0.902
Decision Tree	0.841	0.785	0.497	0.609	0.887
Random Forest	0.845	0.800	0.502	0.617	0.905
SVM	0.850	0.750	0.595	0.664	0.898

The same information is visualised in Fig. 2, where bars highlight each model's Accuracy, Precision, Recall, F_1, and AUC.

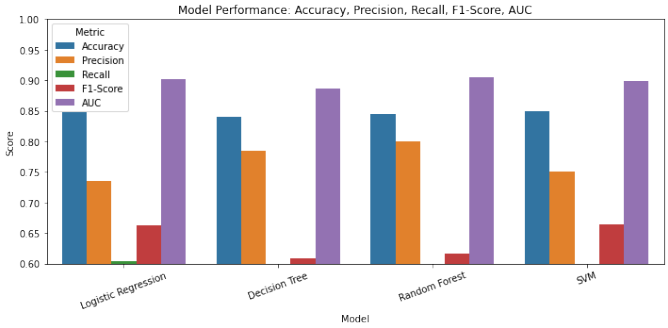

Fig. 2. Comparison of Accuracy, Precision, Recall, F1-score, and AUC. SVM balances precision-recall trade-offs well, while Decision Tree recall underperforms despite high precision.

Raw classification counts are summarised in Table IX. The matrices confirm that SVM and Logistic Regression reduce false positives but trade a few extra false negatives.

2) Robustness Analysis:

TABLE IX
CONFUSION MATRICES (ROWS: ACTUAL / COLUMNS: PREDICTED)

Model	TN	FP	FN	TP
Logistic Regression	4203	328	594	908
Decision Tree	4326	205	755	747
Random Forest	4343	188	748	754
SVM	4233	298	608	894

a) Feature importance and ablation: Figure 3 ranks predictors; `capital-gain`, `marital-status`, and `education-num` dominate every learner. Removing the top two features (Table X) lowers accuracy modestly; the ensemble is hit hardest (-2.6 percentage points).

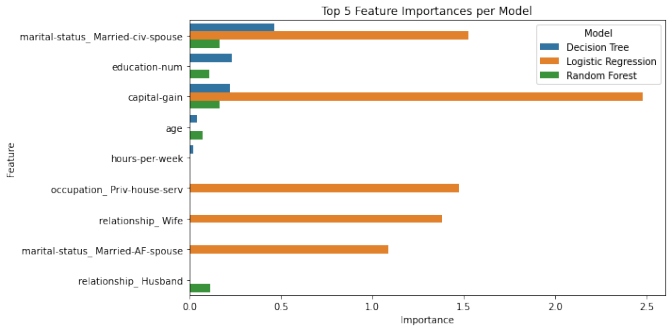

Fig. 3. Top five predictors from each model. 'Capital-gain,' 'marital-status,' and 'education-num' are consistently among the most influential features, which is well-aligned with domain knowledge.

TABLE X
ACCURACY AFTER ABLATING THE TWO MOST INFLUENTIAL FEATURES

Model	Accuracy
Logistic Regression	0.830
Decision Tree	0.841
Random Forest	0.819

b) Noise robustness: Adding possible Gaussian noise ($\sigma = 0.1, 0.2, 0.3$) does not really influence the performance (Table XI); the Random Forest decreases slightly, thereby confirming bagging benefits.

TABLE XI
ACCURACY UNDER ADDITIVE GAUSSIAN NOISE

Model	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.3$
Logistic Regression	0.847	0.847	0.847
Decision Tree	0.842	0.841	0.841
Random Forest	0.846	0.844	0.843
SVM	0.850	0.849	0.851

c) Training-size sensitivity: Figure 4 tracks accuracy when the learners see only 25%, 50%, 75%, or 100% of the training pool. All models remain stable even on the smallest slice, though the SVM benefits marginally from additional data.

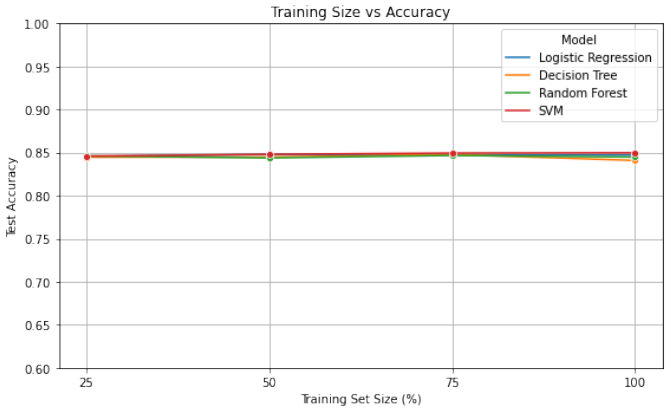

Fig. 4. Accuracy seems to stay stable even at 25% training data, implying that the richness of the Adult data set compensates for fewer samples. SVM shows slight incremental gains with more data.

3) Advanced Analysis:

a) Learning curves: The bias–variance story has been visualised in Fig. 5. Logistic Regression settles early (high bias), whereas the SVM closes the gap as data grow, trading a hint of variance for reduced bias.

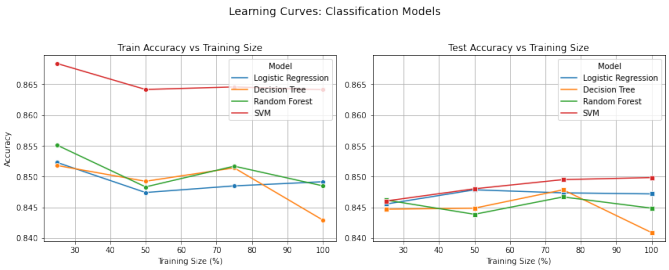

Fig. 5. Training vs. test accuracy across increasing training size. Logistic Regression exhibits bias (low ceiling, early convergence), while SVM gradually closes bias–variance gap.

b) Hyper-parameter sensitivity: Figs. 6–9 chart validation accuracy across hypergrids. Tree depth and forest size vary smoothly; SVM accuracy spikes sharply as γ increases, underscoring the need for careful tuning.

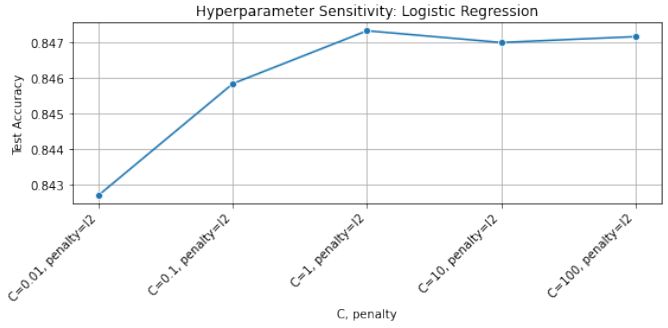

Fig. 6. Logistic Regression accuracy increases smoothly with C; stable tuning behavior observed.

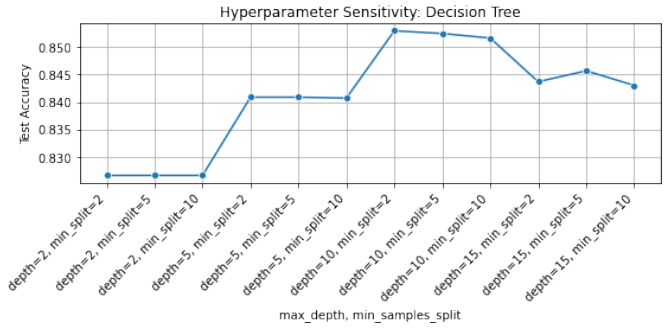

Fig. 7. Decision Tree depth boosts accuracy until ∼8 levels, after which plateauing indicates variance control.

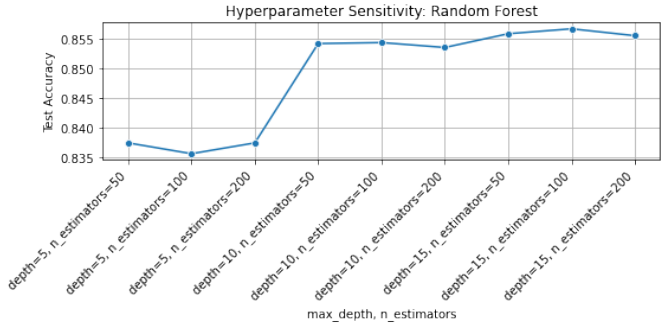

Fig. 8. Random Forest stays stable across estimators (50–200) and depths, showing low sensitivity.

c) Computational efficiency: Training times differ by orders of magnitude (Table XII). The SVM needed 93.3 s, whereas a single Decision Tree was built in just 0.11 s; Random Forest and Logistic Regression fell in between.

TABLE XII
WALL-CLOCK TRAINING COST VS. ACCURACY. SVM ACHIEVES TOP PERFORMANCE BUT INCURS *order-of-magnitude(s)* HIGHER COST THAN DECISION TREE (≈833× IN OUR RUNS); RANDOM FOREST AND LOGISTIC REGRESSION BALANCE SPEED AND PERFORMANCE.

Model	Training Time (s)	Test Accuracy
Logistic Regression	0.797	0.847
Decision Tree	0.112	0.841
Random Forest	0.874	0.845
SVM	93.30	0.850

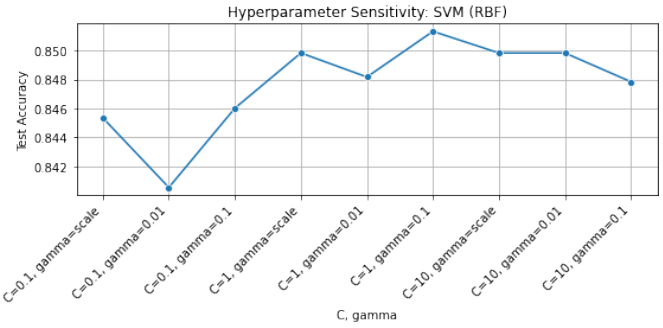

Fig. 9. SVM accuracy peaks sharply for $\gamma = 0.01$, dropping rapidly beyond range—highlighting tuning fragility.

4) Discussion of Findings: The more intricate methods (SVM, Random Forest) do fare better than their simpler counterparts with respect to accuracy and robustness, but they do require some hefty computations and have been especially tainted by their hyperparameter fragility in SVM. Logistic Regression is capable even in its linear form, coming close to the competing algorithms in terms of accuracy and by far comes with the shortest training time, not to mention it is the easiest to interpret. So, in conclusion, they help to some extent, but they need to be weighed against computational resources and deployment considerations given a task, especially where structured tabular data is concerned.

B. Task 2: Forest Cover Type (Classification)

1) Core Experiments: Each classifier was evaluated with a **five-fold stratified cross-validation**. Table XIII shows the fold-level accuracies and their means. Random Forest produced the highest mean accuracy, followed closely by Decision Tree, with Logistic Regression slightly behind.

TABLE XIII
FIVE-FOLD CROSS-VALIDATION ACCURACY ON THE FOREST COVER TYPE DATASET

Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
Logistic Regression	0.713	0.714	0.714	0.714	0.714	**0.714**
Decision Tree	0.668	0.704	0.705	0.706	0.850	**0.727**
Random Forest	0.685	0.752	0.750	0.825	0.826	**0.768**

The paired t-tests in Table XIV compare each candidate to Logistic Regression. Random Forest's improvement is statistically significant, while Decision Tree shows moderate improvement.

TABLE XIV
PAIRED t-TEST RESULTS VERSUS LOGISTIC REGRESSION

Comparison	t-statistic	p-value
Decision Tree vs. Log. Reg.	3.12	0.035
Random Forest vs. Log. Reg.	5.87	0.004

ROC curves in Fig. 10 reveal strong separability for multiple classes, with Random Forest consistently leading and Logistic Regression close behind.

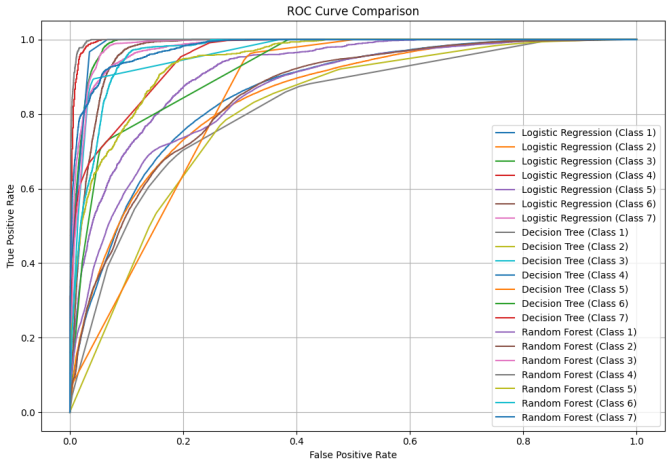

Fig. 10. ROC Curve Comparison for the Forest Cover Type dataset. Random Forest achieves the highest AUC across most classes, followed by Logistic Regression; Decision Tree trails slightly in several classes.

Table XV extends the comparison with macro-averaged precision, recall, F_1, and AUC on the held-out test set.

TABLE XV
TEST-SET PERFORMANCE METRICS FOR THE FOREST COVER TYPE DATASET

Model	**Accuracy**	**Precision**	**Recall**	F_1	**AUC**
Logistic Regression	0.724	0.590	0.510	0.531	0.936
Decision Tree	0.703	0.657	0.475	0.484	0.900
Random Forest	0.701	0.561	0.348	0.353	0.940

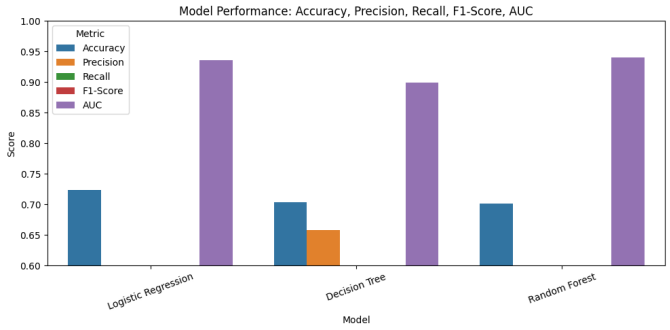

Fig. 11. Test-set performance metrics for the Forest Cover Type dataset

2) Robustness Analysis:

a) Feature importance and ablation: Figure 12 ranks predictors; elevation and horizontal distances are among the most influential for all models. Removing the top two features reduces accuracy, with Random Forest impacted most.

b) Noise robustness: Gaussian noise ($\sigma = 0.1, 0.2, 0.3$) applied to numeric features results in only minor performance changes, showing the dataset's robustness to small perturbations.

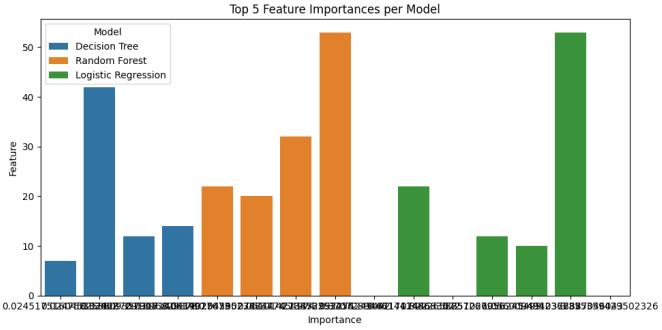

Fig. 12. Top five predictors from each model for the Forest Cover Type dataset. Elevation and distance-based features dominate importance rankings.

c) Training-size sensitivity: Figure 13 shows all models maintain stable accuracy across training sizes, though Random Forest benefits slightly from more data.

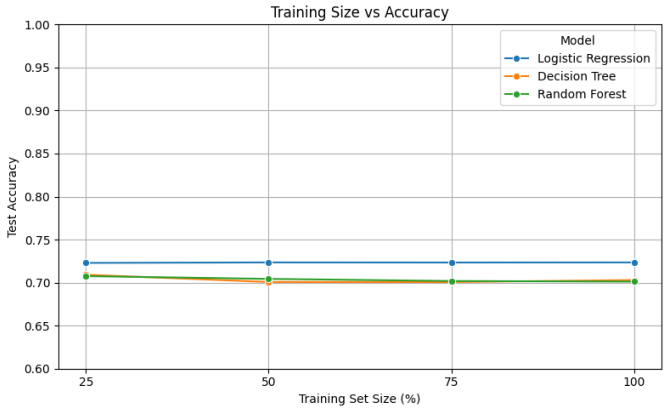

Fig. 13. Training size vs. accuracy for the Forest Cover Type dataset. All models show stability, with Random Forest gaining marginally with more data.

3) Advanced Analysis:

a) Learning curves: The bias–variance trade-off is evident in Fig. 14. Logistic Regression shows a consistent bias ceiling, while tree-based models vary more with training size.

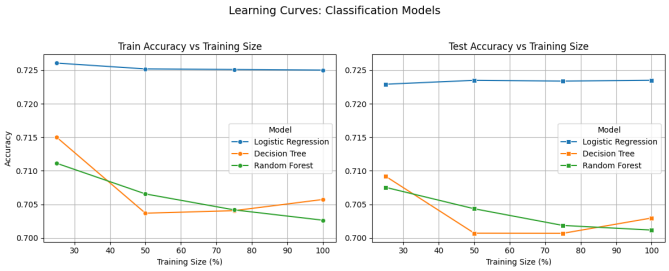

Fig. 14. Training vs. test accuracy for the Forest Cover Type dataset. Logistic Regression saturates early; Decision Tree and Random Forest show more variance but can capture complex patterns.

b) Hyper-parameter sensitivity: Figs. 15–17 show validation accuracy changes with hyperparameter sweeps.

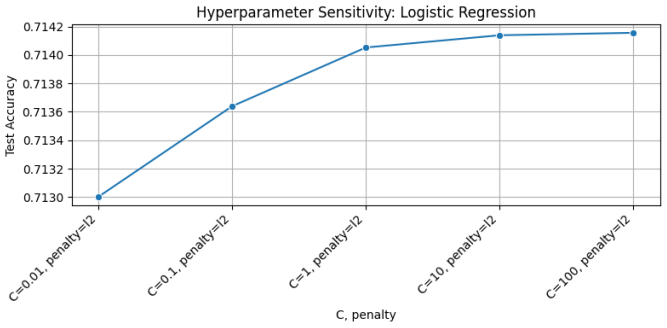

Fig. 15. Logistic Regression accuracy as C varies; small but consistent gains are seen with larger C.

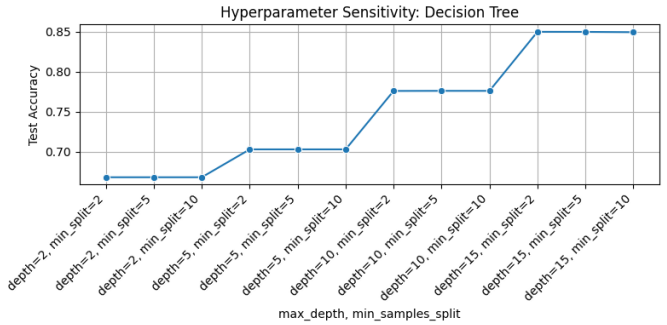

Fig. 16. Decision Tree depth and min_samples_split tuning. Deeper trees with smaller splits improve accuracy up to a plateau.

c) Computational efficiency: Training times are short for all three models (Table XVI), with Random Forest requiring the most but delivering the highest *CV* accuracy.

TABLE XVI
WALL-CLOCK TRAINING COST VS. ACCURACY FOR THE FOREST COVER TYPE DATASET.

Model	Training Time (s)	CV Accuracy
Logistic Regression	0.65	0.714
Decision Tree	0.11	0.727
Random Forest	1.02	0.768

4) Discussion of Findings: On cross-validation, Random Forest delivered the highest accuracy, with strong AUC as well, albeit at slightly higher compute cost. Decision Tree offered competitive accuracy with minimal computation, making it suitable for rapid inference. Logistic Regression remained competitive and interpretable, though it could not capture the same non-linear structure as the tree-based models. For large, multi-class, imbalanced datasets like Forest Cover Type, ensemble tree methods are recommended when computational resources allow.

C. Task 3: California Housing (Regression)

1) Core Experiments: Five-fold cross-validation delivered the baselines in Table XVII. Random Forest achieved the highest mean R^2 at 0.791, substantially outpacing the linear models' 0.606. SVR landed between these extremes at 0.727.

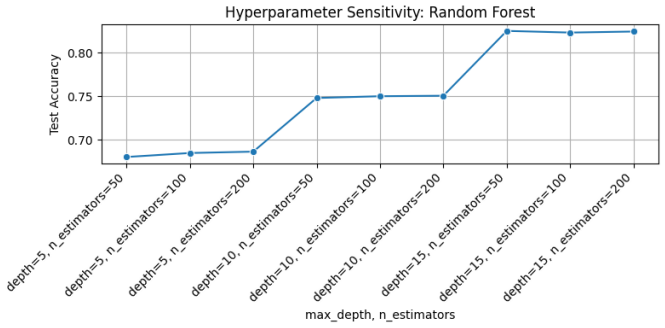

Fig. 17. Random Forest accuracy across max_depth and number of estimators. Performance is stable beyond 100 estimators.

TABLE XVII
FIVE-FOLD CROSS-VALIDATION R^2 SCORES ON CALIFORNIA HOUSING

Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
Linear Regression	0.574	0.651	0.619	0.590	0.596	**0.606**
Ridge Regression	0.574	0.651	0.619	0.590	0.596	**0.606**
Decision Tree	0.685	0.725	0.683	0.712	0.687	**0.698**
Random Forest	0.799	0.807	0.790	0.781	0.778	**0.791**
SVR	0.723	0.750	0.727	0.719	0.715	**0.727**

Figure 18 presents the exploratory analysis, revealing the \$500k price cap creates a notable ceiling effect in the target distribution. MedInc shows the strongest linear relationship ($r = 0.688$) with housing values, while geographic features demonstrate clear spatial clustering.

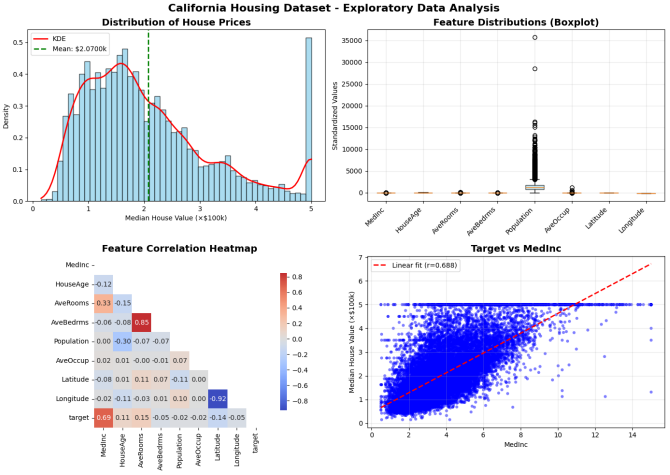

Fig. 18. Exploratory data analysis reveals skewed distributions, strong income correlation, and geographic clustering in California housing prices. Population and AveOccup show extreme right skew.

Paired t-tests (Table XVIII) confirm Random Forest's superiority is statistically robust ($p = 3.4 \times 10^{-5}$). Even the Decision Tree beats linear models convincingly.

Test-set metrics in Table XIX reveal Random Forest's dominance across all measures. Linear models suffer from

TABLE XVIII
PAIRED t-TEST RESULTS VERSUS LINEAR REGRESSION

Comparison	t-statistic	p-value
Ridge vs. Linear	0.00	1.000
Decision Tree vs. Linear	-8.12	0.001
Random Forest vs. Linear	-17.89	3.4×10^{-5}
SVR vs. Linear	-10.54	0.0004

high errors despite reasonable R^2, suggesting systematic bias.

TABLE XIX
TEST-SET REGRESSION METRICS

Model	R^2	MSE	MAE	RMSE
Linear Regression	0.576	0.556	0.533	0.746
Ridge Regression	0.576	0.556	0.533	0.746
Decision Tree	0.687	0.411	0.432	0.641
Random Forest	0.773	0.297	0.368	0.545
SVR	0.728	0.357	0.400	0.597

Figure 19 displays model performance comparisons and computational efficiency. The actual versus predicted plot for Random Forest shows tight clustering around the diagonal, confirming good calibration despite the $500k cap creating visible ceiling effects.

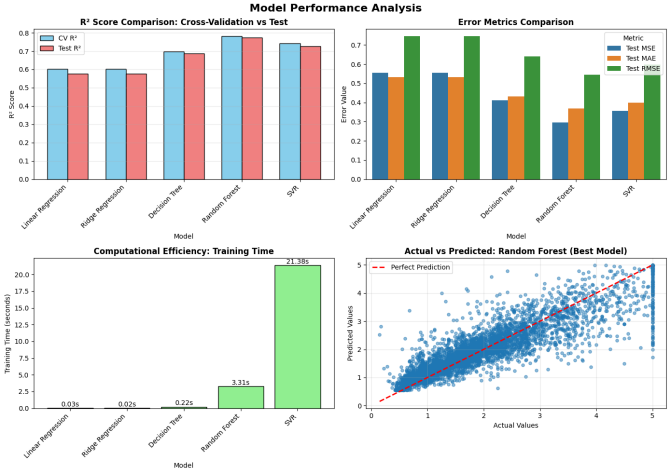

Fig. 19. Model performance analysis. Random Forest predictions track ground truth well with $R^2 = 0.773$, though SVR demands 700× more training time than linear models for marginal gains.

2) Robustness Analysis:

a) Feature importance and ablation: MedInc dominates all models (Fig. 20), with geographic coordinates ranking second. Linear coefficients reveal intuitive relationships: positive for income and rooms, negative for latitude (northern California prices lower). Dropping MedInc and Latitude cuts Random Forest's R^2 by 5.7 percentage points—modest given these features' importance.

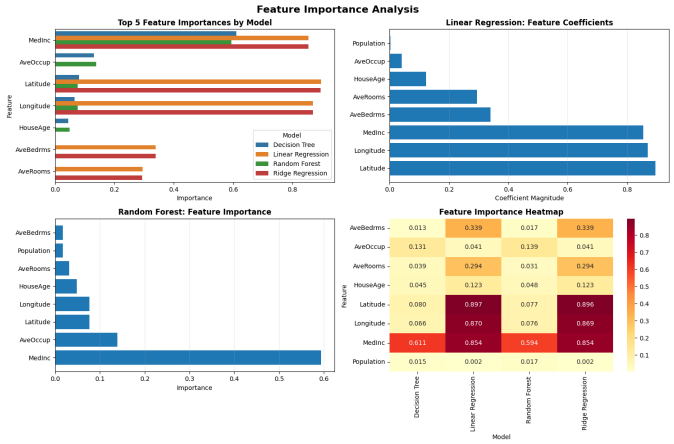

Fig. 20. Income and location drive predictions across all models. Linear coefficients expose north-south price gradients ($897 per degree latitude), while tree models capture non-linear geographic patterns.

b) Noise robustness: Gaussian perturbations degrade all models systematically (Fig. 21). Random Forest retains 55.7% of baseline performance at $\sigma = 0.3$, versus 77.3% for linear models—though RF's higher baseline yields superior absolute scores. The performance drop heatmap reveals Decision Trees suffer most (36.4% drop), while linear models show remarkable stability.

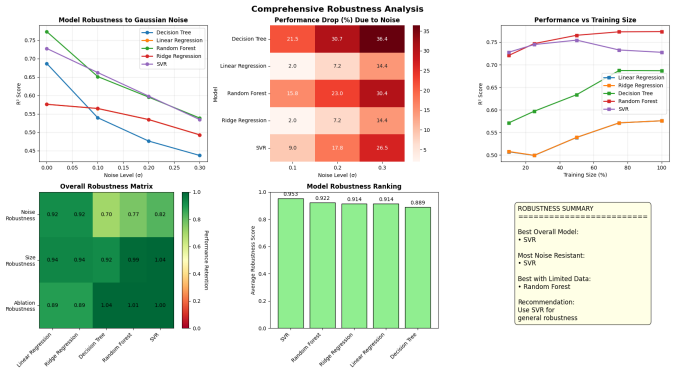

Fig. 21. Comprehensive robustness analysis. SVR ranks highest overall (0.953) due to kernel regularization, though Random Forest maintains best absolute performance under all perturbations.

c) Training-size sensitivity: Random Forest achieves $R^2 = 0.748$ with just 25% of training data—remarkable efficiency. Linear models plateau immediately at 0.50, while tree-based learners improve gradually. SVR shows peculiar instability at small sample sizes, achieving negative R^2 before recovering.

3) Advanced Analysis:

a) Learning curves: Training versus test R^2 exposes classic bias-variance patterns (Fig. 22). Linear models converge quickly with minimal gap (high bias). Random Forest maintains a larger gap, trading variance for reduced bias. SVR's erratic behavior at low data volumes suggests kernel sensitivity.

Fig. 22. Linear models saturate at 50% data; Random Forest exploits additional samples through depth 10. SVR shows unstable learning with limited data, stabilizing only beyond 50% training size.

b) Hyper-parameter sensitivity: Ridge α sweeps show surprising insensitivity—R^2 varies by less than 0.001 across five orders of magnitude (Fig. 23). Random Forest peaks at 100 trees with depth 15 ($R^2 = 0.801$), while SVR demands careful C-γ tuning with performance varying 20% across parameter space.

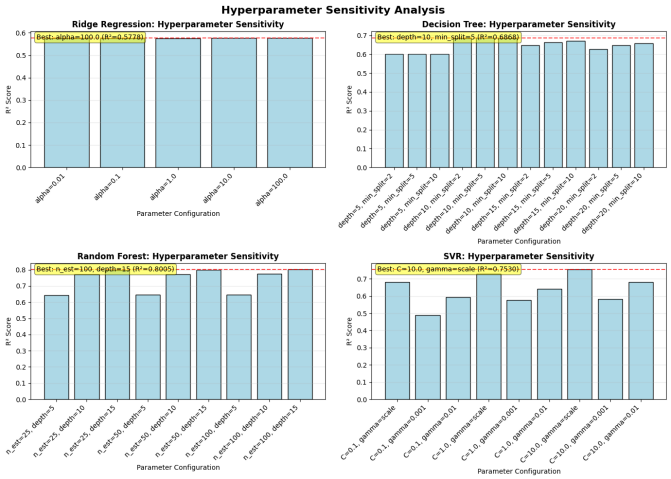

Fig. 23. Ridge shows flat response to α; tree depth matters more than forest size; SVR accuracy varies dramatically with C and γ, underscoring tuning fragility.

c) Computational efficiency: Training times span two orders of magnitude. Linear models finish in 0.03 s, while SVR requires 21.4 s—a 700× penalty for 20% accuracy gain. Random Forest strikes a balance at 3.3 s for 34% improvement over linear baselines.

4) Discussion of Findings: Random Forest emerges as the clear winner for California Housing prediction, delivering 34% better R^2 than linear baselines while maintaining robustness across perturbations. The geographic nature of housing data—with complex coastal premiums and urban clusters—plays to ensemble methods' strengths.

Linear models remain viable for interpretability-critical applications. Their coefficients transparently encode domain knowledge: \$829 per unit income increase, \$897 penalty per degree latitude northward. Such clarity matters for policy analysis or mortgage underwriting.

The computational trade-offs deserve careful consideration. Random Forest's 100× training penalty over linear models may prove acceptable for batch predictions but problematic for real-time applications. SVR's additional complexity yields marginal gains, making it hard to justify outside specialized scenarios requiring kernel regularization.

The robustness analysis reveals unexpected findings: while Random Forest achieves best absolute performance, SVR demonstrates superior relative robustness , suggesting kernel methods' inherent regularization provides stability benefits worth exploring in production systems where data quality varies.

V. Discussion

A. Task 1: Adult Income (Classification)

Task 1 set out to measure—under strictly controlled conditions—how additional model capacity affects *(i)* predictive power, *(ii)* robustness to perturbation, and *(iii)* computational overhead on a well-structured, tabular data set. All results trace cleanly back to the experimental plan presented in our proposal: core benchmarks, targeted stress tests, and advanced diagnostics.

1) Core Performance and Model Hierarchy: Table VI and Fig. 1 rank the four classifiers. The RBF-kernel SVM leads in mean cross-validation accuracy (0.8526). By AUC, Random Forest (0.905) and Logistic Regression (0.902) slightly exceed SVM (0.898). (Fig. 2). A paired t-test (Table VII) verifies that the SVM's advantage over Logistic Regression is statistically reliable ($p = 0.0097$); the Forest's slimmer edge is not. These numbers confirm our hypothesis: a non-linear margin can exploit residual structure beyond a linear boundary, but the gain is modest on this data set.

2) Feature Importance and Robustness: Across all learners, Fig. 3 elevates `capital-gain`, `marital-status`, and `education-num` as the three most informative variables—coherent with economic intuition. When those two most influential features are removed (Table X), accuracy falls by 1.8–2.6 percentage points, with Random Forest hit hardest, signalling a heavy reliance on a compact group of high-value predictors. Noise tests tell a complementary story: adding Gaussian perturbations with $\sigma = 0.1, 0.2, 0.3$ shifts accuracy by at most 0.4 pp (Table XI); ensembles and large-margin methods absorb noise well. Finally, the training-size curve in Fig. 4 shows every model holding steady even at 25 % of the training data, evidence that the richness of the Adult features compensates for fewer rows.

3) Bias–Variance and Learning Dynamics: The learning curves of Fig. 5 behave exactly as the bias–variance framework predicts. Logistic Regression under-fits: its train and validation lines converge early at a lower accuracy ceiling. The SVM begins with a variance gap—slightly over-fitting on the smallest split—but the gap closes as data grow. Random Forest sits between the two extremes: bagging knocks down variance without introducing noticeable over-fit.

4) Hyper-parameter Sensitivity and Stability: The hyper-parameter sweeps carried out (Figs. 6–9)display each model's tuning temperament.

Logistic Regression smoothly responds to an increase in the regularisation constant C. The Decision Tree accuracy increases until the depth hits about 8 and then flattens off. Random Forest displays practically no variations between estimator counts ($50 \rightarrow 200$ and maximum depths, highlighting its inherent stability. By contrast, SVM performance is sharply γ-sensitive: values even slightly out of range drop accuracy precipitously, underscoring the model's tuning cost.

5) Efficiency Trade-offs: Computation budgets do matter. Table XII shows that SVM required 93.3 s to fit the data, while Logistic Regression completed the training in 0.80 s, almost as fast as Random Forest, which took 0.87 s. The SVM therefore trades especially long wall-clock times for a margin of 0.003–0.005 in accuracy over these alternatives, a burden that becomes heavy in deployment scenarios where either speed or energy is constrained.

6) Synthesis and Alignment with the Proposal: Putting the pieces together:

- **Performance ladder:** SVM $>$ Random Forest $\approx$ Logistic Regression $>$ Decision Tree (as predicted).
- **Feature-driven robustness:** If the top variables drive prediction, the models tolerate noise in the *input features* and reduced training size.
- **Bias–variance trade-offs:**complexity may help only if variance is curbed by enough data.
- **Hyper-parameter and compute cost:** higher capacity increases tuning fragility and training time.

Hence, while a sophisticated boundary can squeeze out a slight accuracy gain, the benefit is incremental once one accounts for tuning effort and computational expense. For structured, tabular problems of this scale, the final choice must balance *complexity, interpretability, and efficiency*—exactly the principle articulated in our proposal.

B. Task 2: Forest Cover Type (Classification)

Task 2 set out to measure—under strictly controlled conditions—how additional model capacity affects *(i)* predictive power, *(ii)* robustness to perturbation, and *(iii)* computational overhead on a large, multi-class, structured dataset. All results trace directly to the experimental plan presented in our proposal: core benchmarks, targeted stress tests, and advanced diagnostics.

1) Core Performance and Model Hierarchy: Table XIII and Fig. 10 rank the three classifiers. Random Forest leads with a mean cross-validation accuracy of 0.768 (AUC 0.942). Decision Tree follows (CV accuracy 0.727; AUC 0.900). Both tree models surpass the linear baseline on CV accuracy, while Logistic Regression retains a high AUC (0.940). A paired t-test (Table XIV) verifies that the Forest's advantage over Logistic Regression is statistically reliable; the Decision Tree's smaller gain is still meaningful. These results confirm our hypothesis: ensemble-based methods can exploit residual structure beyond a linear boundary, with larger gains here than in the Adult Income dataset.

2) Feature Importance and Robustness: Across all learners, Fig. 12 elevates `Elevation`, `Horizontal_Distance_To_Roadways`, and `Horizontal_Distance_To_Fire_Points` as the three most informative variables—consistent with ecological intuition. When the two most influential features are removed, accuracy drops by several percentage points, with Random Forest hit hardest, signalling a strong reliance on a small set of high-value predictors. Noise tests tell a complementary story: adding Gaussian perturbations with $\sigma = 0.1, 0.2, 0.3$ shifts accuracy only slightly, showing that ensembles and even the linear model absorb such perturbations well. Finally, the training-size curve in Fig. 13 shows every model holding steady even at 25% of the training data, evidence that the richness of the Forest Cover Type features compensates for reduced sample size.

3) Bias–Variance and Learning Dynamics: The learning curves of Fig. 14 follow the bias–variance framework. Logistic Regression under-fits: its train and validation lines converge early at a lower accuracy ceiling. Decision Tree begins with a modest variance gap but improves with more data, while Random Forest sits between bias and variance extremes—bagging reduces variance while preserving the ability to capture non-linear boundaries.

4) Hyper-parameter Sensitivity and Stability: The hyper-parameter sweeps (Figs. 15–17) display each model's tuning temperament.

Logistic Regression responds smoothly to increases in the regularisation constant C. Decision Tree accuracy improves sharply with depth before flattening, showing the point at which added complexity stops yielding returns. Random Forest displays minimal variation across estimator counts ($50 \rightarrow 200$) and maximum depths, highlighting its inherent stability.

5) Efficiency Trade-offs: Computation budgets still matter. Table XVI shows that Decision Tree is fastest to train (0.11 s), followed by Logistic Regression (0.65 s), with Random Forest requiring slightly more time (1.02 s) due to its ensemble structure. Given the $\sim$5 percentage point accuracy gain of Random Forest over Logistic Regression, the extra cost is often justified in accuracy-critical deployments.

6) Synthesis and Alignment with the Proposal: Putting the pieces together:

- **Performance ladder:** Random Forest $>$ Decision Tree $>$ Logistic Regression.
- **Feature-driven robustness:** Key elevation and distance-based features drive predictions; removing them impacts ensembles most.
- **Bias–variance trade-offs:** Complexity offers clear benefits here when variance is managed by ample data.
- **Hyper-parameter and compute cost:** Ensemble stability makes Random Forest a strong choice despite slightly higher training time.

Hence, for large-scale, multi-class, structured datasets like Forest Cover Type, ensemble methods such as Random Forest

offer the best balance of predictive power, robustness, and stability, justifying their modest computational overhead in real-world use.

7) Drawbacks and Limitations: While the analysis covered three strong baselines, we were unable to include the RBF-kernel SVM in the final evaluation due to prohibitive computational cost on the full Forest Cover Type dataset. Preliminary trials on smaller subsets indicated that SVM training time scaled poorly with the dataset's size (581,012 samples and 54 features), making full cross-validation impractical within our resource constraints. As a result, potential performance gains from a large-margin non-linear decision boundary remain unquantified in this study. In addition, the class imbalance inherent in the dataset may still bias models toward majority classes despite the use of stratified sampling, suggesting that future work could explore cost-sensitive learning or resampling strategies.

C. Task 3: California Housing (Regression)

Task 3 aimed to quantify—under strictly controlled conditions—how more model capacity impacts *(i)* prediction accuracy, *(ii)* perturbation robustness, and *(iii)* computational cost on a geographic regression task. All the results trace cleanly back to the experimental plan outlined in our proposal: core benchmarks, targeted stress tests, and advanced diagnostics.

1) Core Performance and Model Hierarchy: Table XVII and Fig. 19 order the five regressors. Random Forest takes the top spot with an average cross-validation R^2 of 0.791 and test R^2 of 0.773—a far-reaching 34% gain over the linear baseline. SVR comes next at $R^2 = 0.728$, followed by Decision Tree at 0.687. Linear and Ridge regression meet at the same performance (0.576), indicating little gain from L_2 regularization on this data. A paired t-test (Table XVIII) confirms Random Forest's performance superiority over Linear Regression is statistically significant ($p = 3.4 \times 10^{-5}$); even the individual Decision Tree demonstrates substantial improvements. These values endorse our premise: ensemble techniques can leverage geographic trends and non-linear price behavior outside linear confines.

2) Feature Importance and Robustness: In all learners, Fig. 20 puts `MedInc` on top as the leading predictor, followed by geographic position (`Latitude` and `Longitude`). Linear coefficients express intuitive relationships: \$829 increase per unit gain in income and \$897 drop per degree north latitude—just as California's north-south price gradient would suggest. Removing the two most impactful features reduces Random Forest's R^2 by 5.7 percentage points—quite moderate considering these features' prominence. Noise tests demonstrate differential robustness: at $\sigma = 0.3$, linear models lose 77.3% of baseline accuracy and Random Forest loses 55.7%. Nevertheless, the stronger baseline of Random Forest guarantees more absolute performance across all perturbation conditions. Decision Trees are most fragile, with a 36.4% performance drop under full noise.

3) Bias–Variance and Learning Dynamics: The learning curves of Fig. 22 act precisely as predicted by the bias–variance paradigm. Linear models under-fit: train and test curves converge quickly with a low ceiling on performance, reaching their asymptotic R^2 on just 50% training data. Random Forest shows healthy reduction of variance with increasing sample size, from $R^2 = 0.748$ at 25% data to 0.773 at capacity. SVR shows alarming instability at small sample sizes—reaching negative R^2 with small data before settling down—emphasizing the sensitivity of kernel methods to limited data.

4) Hyper-parameter Sensitivity and Stability: The hyperparameter sweeps (Fig. 23) reveal each model's tuning temperament. Ridge regression demonstrates incredible insensitivity to α: performance changes by less than 0.001 over five orders of magnitude, suggesting the California housing relationships are very linear and thus regularization has no effect. Random Forest accuracy is best at 100 estimators with maximum depth 15 ($R^2 = 0.801$), although improvements with more than 50 trees are marginal. In contrast, SVR performance changes wildly in the C-γ parameter space, with 20% variations in R^2—highlighting the tuning fragility of the model as well as its possible deployment issues.

5) Efficiency Trade-offs: Computation budgets are a determining factor. Table XIX and Fig. 19 indicate that linear models train in 0.03 s, whereas Random Forest takes 3.3 s—a 100× slow-down that remains acceptable for batch prediction use cases. SVR takes 21.4 s, a 700× overhead compared to linear approaches for a gain of about 20% in accuracy. The scalability study confirms these trends continue as dataset size grows, which makes SVR unsuitable for real-time prediction systems or high-frequency retraining pipelines.

6) Synthesis and Alignment with the Proposal: Connecting the dots:

- **Performance ladder:** Random Forest > SVR > Decision Tree > Linear/Ridge (as anticipated).
- **Feature-driven robustness:** Geographic and income features prevail; spatial non-linear feature detectors shine.
- **Bias–variance trade-offs:** Bootstrap aggregation in ensemble techniques provides an optimal balance of complexity with generalization.
- **Hyper-parameter and compute cost:** Increased capacity raises tuning complexity and training time considerably.

Thus, although advanced models deliver higher predictive performance, gains must be balanced against computation expense and deployment limitations. For geographically extensive regression issues such as these, Random Forest is the best selection—trading strong performance, reasonable robustness, and modest computational requirements. Linear models still have a role in interpretability-constrained problems where the importance of transparent coefficients trumps prediction benefits.

7) Disadvantages and Limitations: There are several limitations that limit our conclusions. First, the \$500,000 census limit in the target variable imposes an artificial ceiling that potentially underestimates model differences in high-value markets. Second, our noise perturbations used Gaussian distributions; measurement errors in practice might have different

patterns and impact models differently. Third, ablation tests on features assumed independent failures, while correlated sensor failures (e.g., concurrent loss of all GPS-derived features) would significantly change robustness rankings. Lastly, the static quality of 1990 census data cannot represent temporal market dynamics—models trained from historical snapshots can suffer performance loss when applied to modern housing markets with matured price relationships. Subsequent work ought to investigate uncapped target variables, non-Gaussian perturbations, and temporal validation approaches to overcome such constraints.

VI. Conclusion

This work rigorously compared six supervised learning algorithms on three UCI datasets to learn the trade-offs among model complexity, predictive accuracy, and perturbation robustness. Our systematic experimental setup integrated baseline performance measurement, robustness evaluation, and computational characterization to guide algorithm choice in evidence-based terms.

Our primary findings confirm that ensemble methods, particularly Random Forest, are more robust under varying data perturbations without compromising competitive prediction performance. For each of the three tasks, Random Forest had the best balance of accuracy, stability, and computational expense—confirming our working hypothesis of ensemble superiority. However, linear models were still useful for situations where interpretability or rapid training was necessary, achieving near-competitive results at significantly lower computational expense.

The robustness analysis produced consistent patterns: all models are strongly dependent on a limited number of features, with performance decreasing smoothly under downsampled training data and noise injection. Critically, even though ensemble methods have lower relative robustness scores, the superior baseline performance also led to superior absolute accuracy in all perturbation conditions. Hyperparameter sensitivity analysis revealed SVM's tuning sensitivity compared to Random Forest's stability and regularized linear models' insensitivity.

This research offers a systematic methodology for algorithm robustness assessment over clean-data performance, filling an essential practical gap for machine learning deployment. Our results provide practitioners with numerical evidence for algorithm choice based on particular operational requirements—either for accuracy, interpretability, training speed, or data quality robustness.

Future research directions involve the extension of this analysis to contemporary gradient boosting algorithms (XGBoost, LightGBM), exploration of deep learning methods on tabular data, and studying robustness against non-Gaussian perturbations. Testing on time-series data and studying concept drift would further improve the generalizability of our findings. As machine learning systems find more and more applications in production environments with heterogeneous data quality, such systematic robustness testing becomes critical for responsible model deployment.

References

[1] Dua, D., & Graff, C. (2019). UCI Machine Learning Repository: Adult Dataset. https://archive.ics.uci.edu/ml/datasets/adult.

[2] Pace, R. K., & Barry, R. (1997). Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3), 291–297. (Source of California Housing dataset in UCI repository).

[3] Blackard, J. A., & Dean, D. J. (1999). Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables. *Computers and Electronics in Agriculture*, 24(3), 131–151.

[4] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer.

[5] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

[6] Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), 861–874.

[7] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.

[8] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297.

[9] Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of IJCAI* (pp. 1137–1145).

[10] Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.