DISTRIBUTION SHIFT

A Study on Their Effects on Statistical Models and Strategies for Mitigation

Andrea Spinelli, Giacomo Amerio,
Giovanni Lucarelli, Tommaso Piscitelli

University of Trieste

Introduction

Aims of project

Models:

- Logistic Regression
- Random Forest
- Gradient Boosting
- XGBoost

Roadmap:

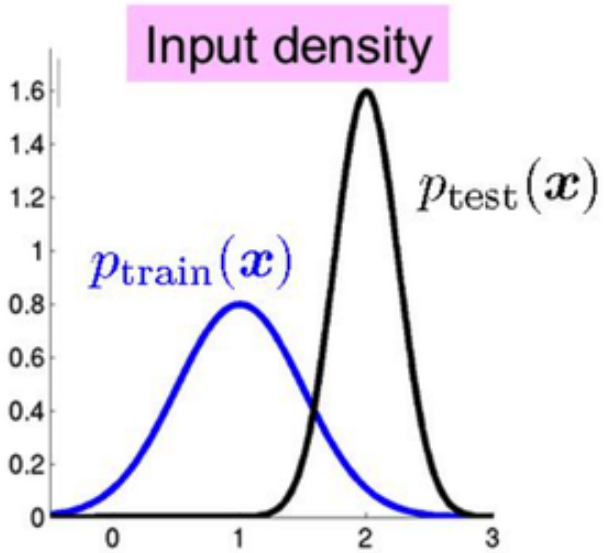
- Creation of synthetic data and data affected by shift
- Evaluation of model performance on the data
- Identification of improvement strategies (R.A.W.)

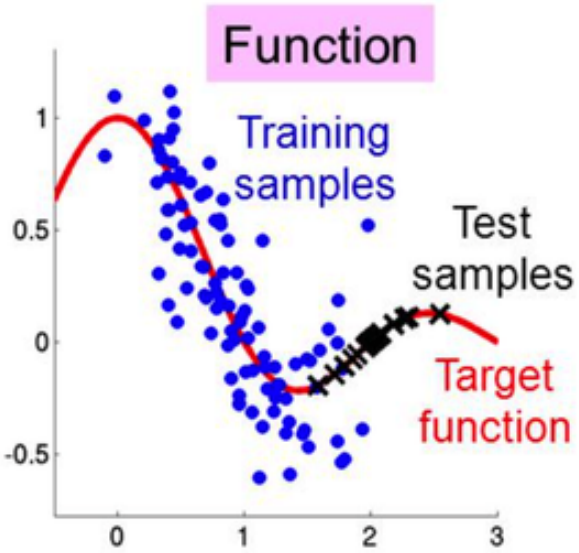
Dataset shift

- **Dataset shift** is a common problem in machine learning.
- It occurs when the distribution of the training data differs from the distribution of the test data.
- The two most common and well-studied causes of Dataset shift are:
  - **Sample selection bias**
  - **Non stationary environments**

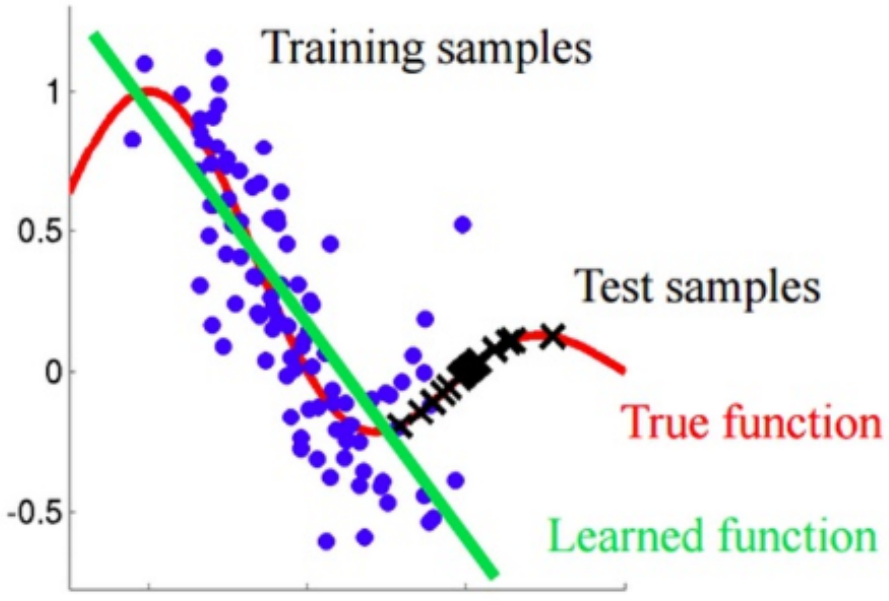
Covariate shift

$$P_{\text{tra}}(Y \mid X) = P_{\text{tst}}(Y \mid X) \quad \text{but} \quad P_{\text{tra}}(X) \neq P_{\text{tst}}(X)$$





Inaccurate Model

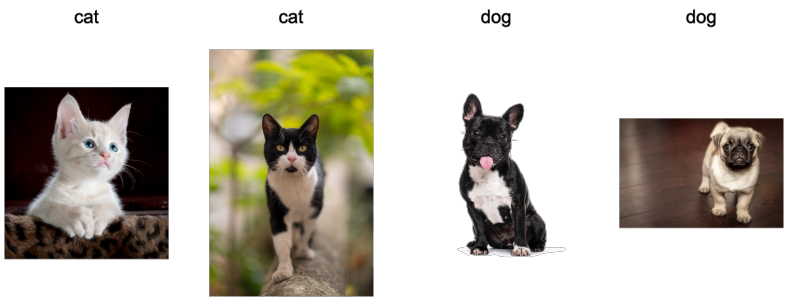


Changes in the features distribution can significantly impact the model's *effectiveness*.

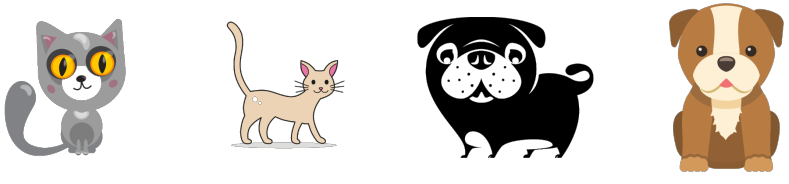
Example

Consider a model designed to distinguish between images of cats and dogs:

Training set:



Test set:



Model will not accurately distinguish between cats and dogs because the feature distribution will differ.

Data Generation

Training Dataset: Features

The dataset consists of $n = 10^4$ observations with 3 features and 1 binary target variable.

Features:

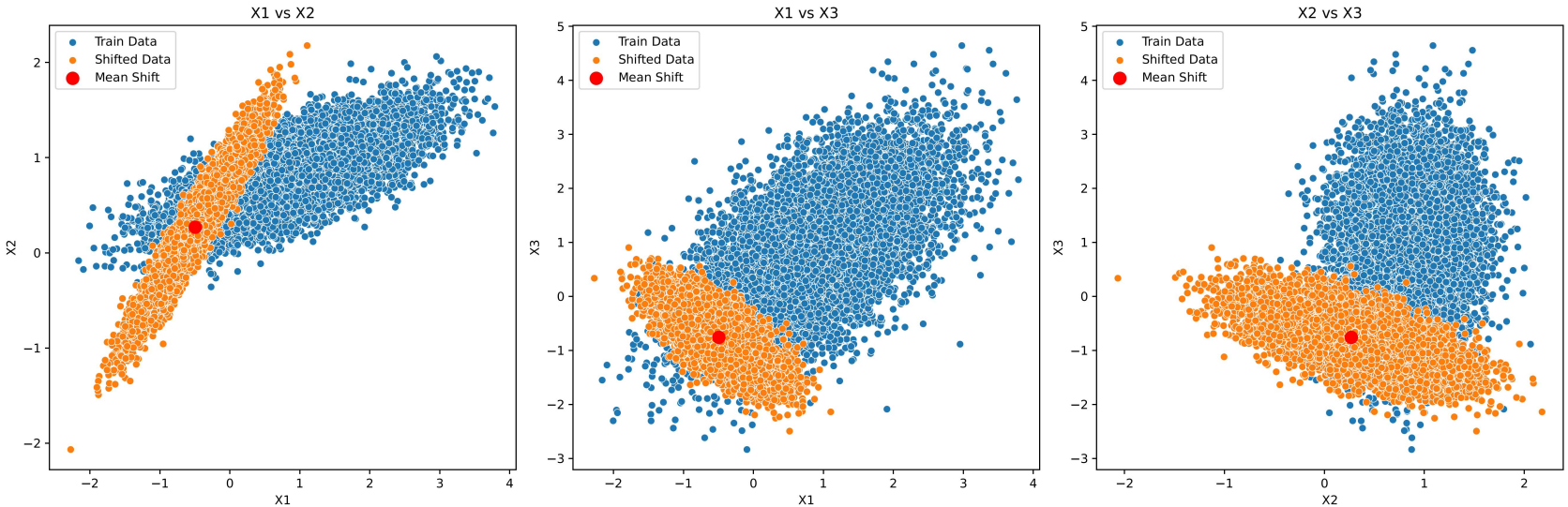
- $X_{\text{train}} = (X_{\text{train}\,1}, X_{\text{train}\,2}, X_{\text{train}\,3}) \sim \mathcal{N}(\boldsymbol{\mu}_{\text{train}}, \boldsymbol{\Sigma}_{\text{train}})$
- $\mu_{\text{train}\,i} \sim \mathcal{U}_{[0,1]}$ for $i = 1, 2, 3$
- $[\boldsymbol{\Sigma}_{\text{train}}]_{i,j} \sim \mathcal{U}_{[-1,1]}$ for $i, j = 1, 2, 3$

Note: The $\boldsymbol{\Sigma}$ randomly generated has been transformed to a symmetric and positive semidefinite matrix by computing $\boldsymbol{\Sigma}\boldsymbol{\Sigma}^T$.

Testing Dataset

Same dataset structure as the train set, but:

- $X_{\text{shift}} = (X_{\text{shift}\,1}, X_{\text{shift}\,2}, X_{\text{shift}\,3}) \sim \mathcal{N}(\boldsymbol{\mu}_{\text{shift}}, \boldsymbol{\Sigma}_{\text{shift}})$
- $\boldsymbol{\mu}_{\text{shift}} = \mathcal{Q}_{0.05}(X_{\text{train}})$
- $[\boldsymbol{\Sigma}_{\text{shift}}]_{i,j} \sim \mathcal{U}_{[-0.5,0.5]}$ for $i, j = 1, 2, 3$



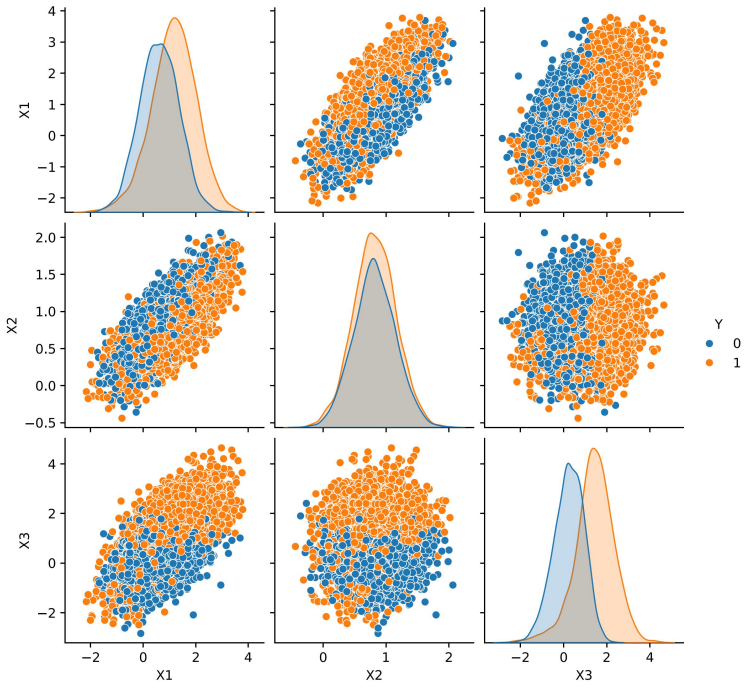
Target Variable

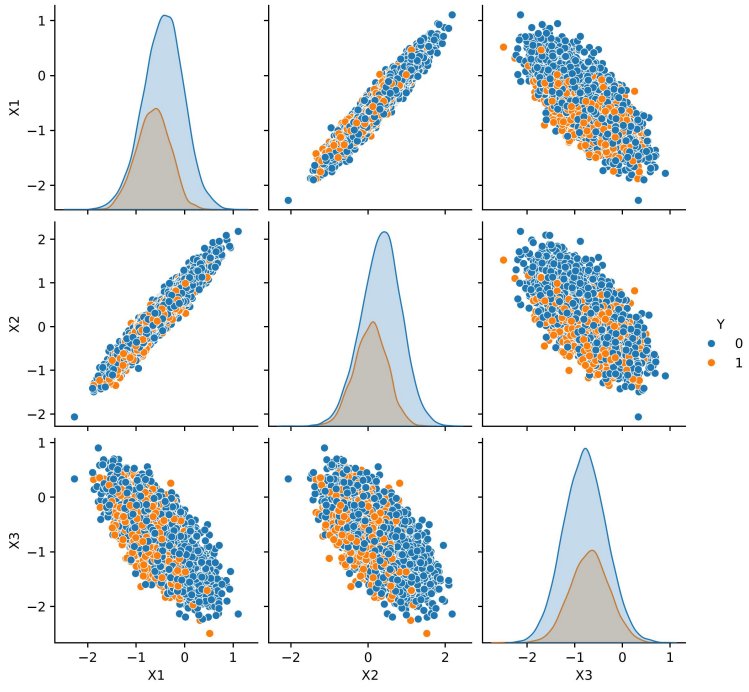
Building the **target variable** $Y \in \{0, 1\}$:

$$z = \beta_0 + \sum_{i=1}^{3} \beta_i x_i + \sum_{i=1}^{3} \beta_{ii} x_i^2 + \sum_{i=1}^{2} \sum_{j=i+1}^{3} \beta_{ij} x_i x_j \,, \quad \beta_\cdot \sim \mathcal{U}_{[-1,1]}$$

$$Y \sim \mathrm{Be}(p) \,, \quad p = \frac{1}{1 + e^{-z}}$$

Label Distributions





Note: IR from 1.19 to 2.36

Testing Mixture

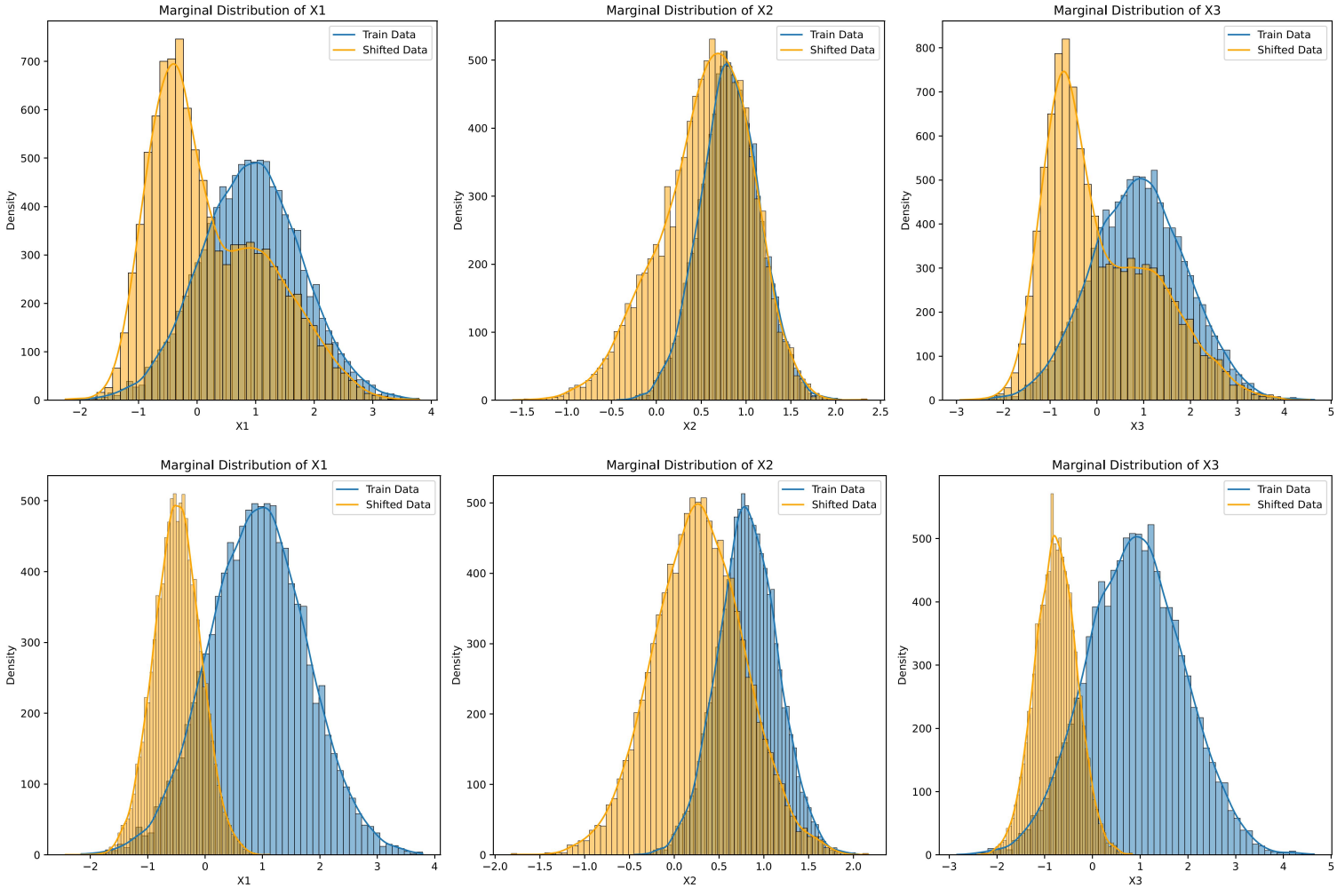
Series of datasets using **statistical mixtures** of the training features distribution and the fully shifted distribution.

$$X_\alpha \sim \alpha \cdot \mathcal{N}(\boldsymbol{\mu}_{\text{shift}}, \boldsymbol{\Sigma}_{\text{shift}}) + (1-\alpha) \cdot \mathcal{N}(\boldsymbol{\mu}_{\text{train}}, \boldsymbol{\Sigma}_{\text{train}})$$

$$\alpha \in \{0.0, 0.1, \ldots, 1.0\}$$

Y_α generated as before

Note: $X_{0.0}$ and X_{train} come from the same distribution, but the former are used as fresh new data.



Top: $\alpha = 0.5$. Bottom: $\alpha = 1.0$.

Performance Degradation

Performance Degradation

We evaluated four distinct statistic models:

- **Random Forest**: An ensemble method combining multiple decision trees, where final prediction is determined by majority voting
- **Gradient Boosting**: Sequential ensemble approach that iteratively improves predictions by combining weak learners (in our case decision trees)
- **XGBoost**: Advanced implementation of gradient boosting with enhanced regularization and computational efficiency
- **Logistic Regression**: [baseline] Classic linear classifier used as a reference model

Fine Tuning and Performance Metric

Performance Metric

We used the **Area Under the Receiver Operating Characteristic Curve (ROC-AUC)** as the performance metric for our models.

Fine Tuning

We performed a **hyperparameter tuning** using the **Grid Search** method with 5-fold cross-validation.

Random Forest

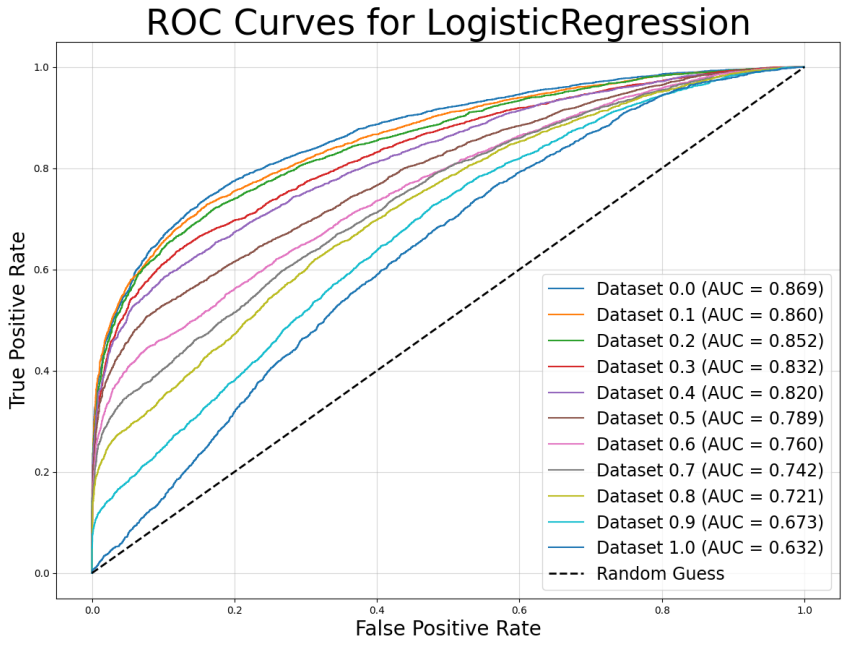
Params	Value
n_estimators	125
max_depth	5
min_samples_split	5
min_samples_leaf	1
bootstrap	True

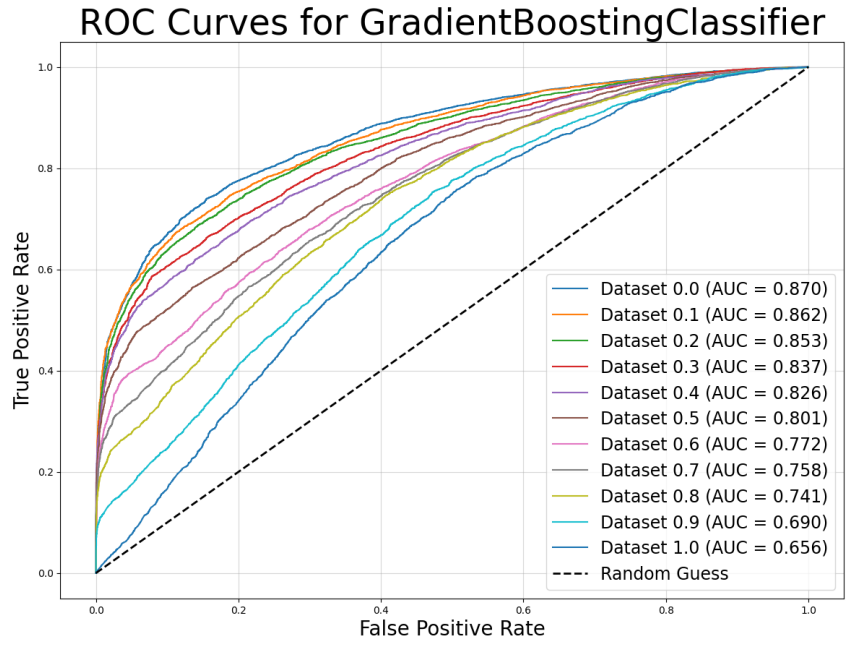
Gradient Boosting

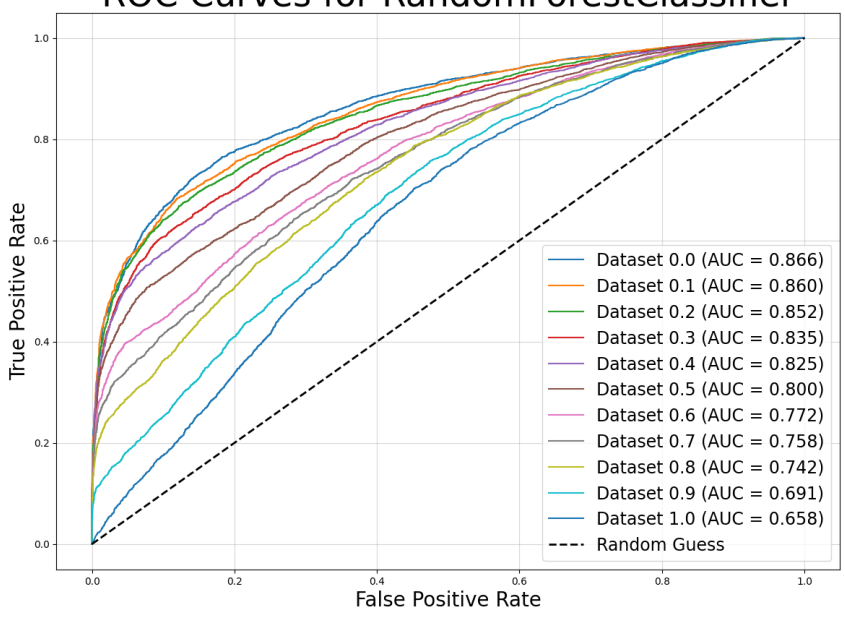
Params	Values
n_estimators	500
max_depth	3
learning_rate	0.005
subsample	0.4

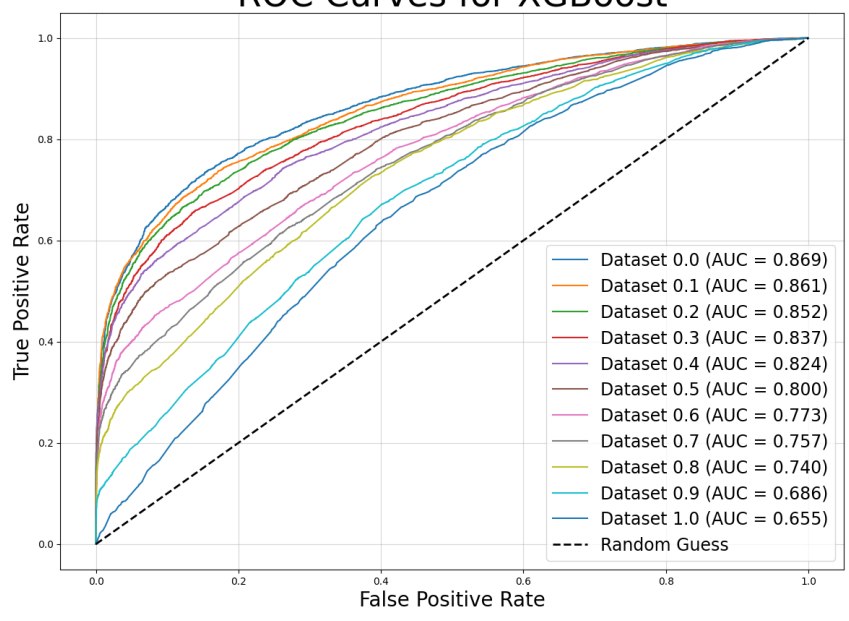
XGBoost

Params	Values
n_estimators	100
max_depth	6
learning_rate	0.1
subsample	0.7
gamma	5

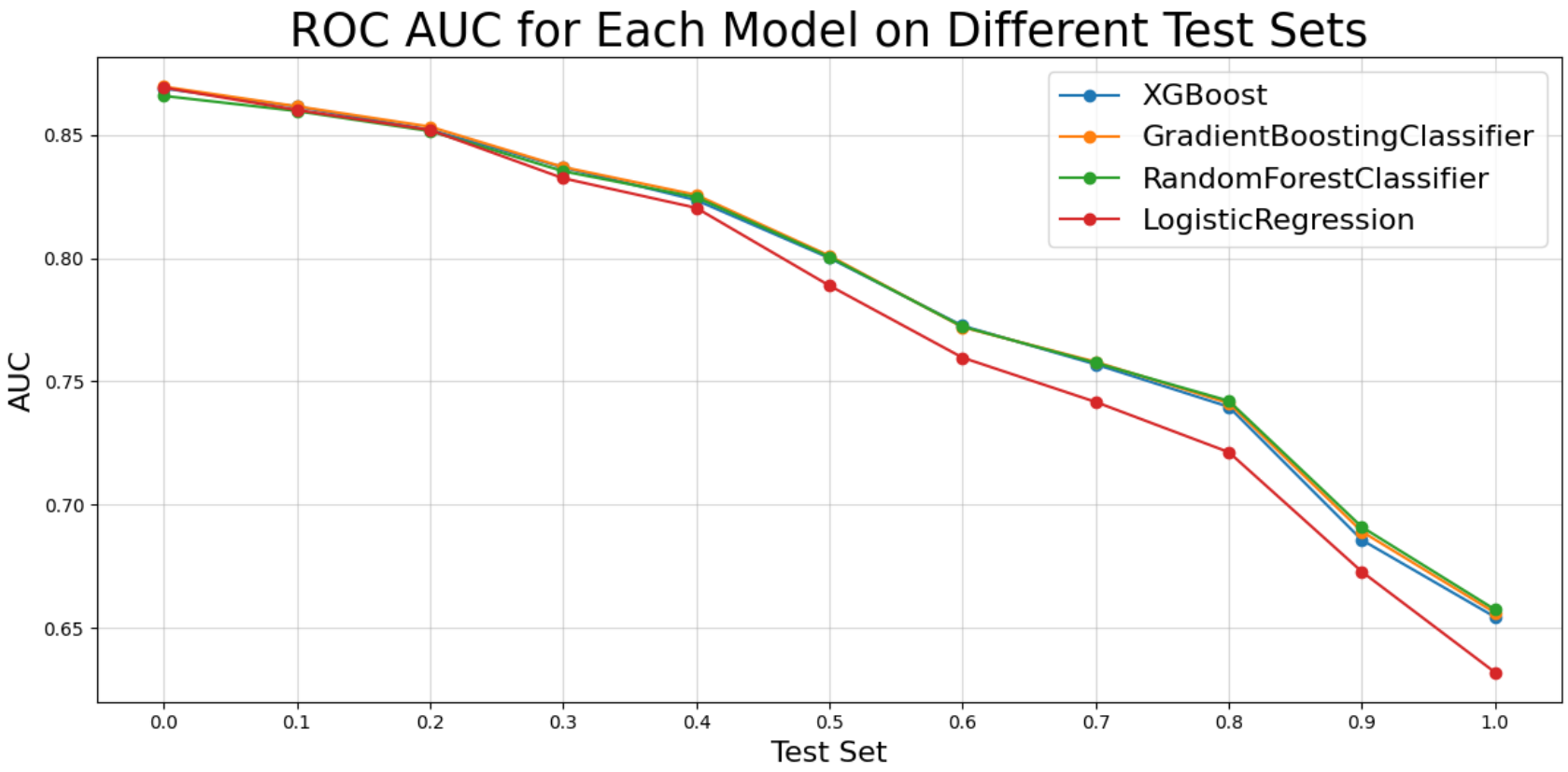

ROC Curves for LogisticRegression
True Positive Rate
False Positive Rate
Dataset 0.0 (AUC = 0.869)
Dataset 0.1 (AUC = 0.860)
Dataset 0.2 (AUC = 0.852)
Dataset 0.3 (AUC = 0.832)
Dataset 0.4 (AUC = 0.820)
Dataset 0.5 (AUC = 0.789)
Dataset 0.6 (AUC = 0.760)
Dataset 0.7 (AUC = 0.742)
Dataset 0.8 (AUC = 0.721)
Dataset 0.9 (AUC = 0.673)
Dataset 1.0 (AUC = 0.632)
Random Guess


ROC Curves for GradientBoostingClassifier
True Positive Rate
False Positive Rate
Dataset 0.0 (AUC = 0.870)
Dataset 0.1 (AUC = 0.862)
Dataset 0.2 (AUC = 0.853)
Dataset 0.3 (AUC = 0.837)
Dataset 0.4 (AUC = 0.826)
Dataset 0.5 (AUC = 0.801)
Dataset 0.6 (AUC = 0.772)
Dataset 0.7 (AUC = 0.758)
Dataset 0.8 (AUC = 0.741)
Dataset 0.9 (AUC = 0.690)
Dataset 1.0 (AUC = 0.656)
Random Guess


ROC Curves for RandomForestClassifier
True Positive Rate
False Positive Rate
Dataset 0.0 (AUC = 0.866)
Dataset 0.1 (AUC = 0.860)
Dataset 0.2 (AUC = 0.852)
Dataset 0.3 (AUC = 0.835)
Dataset 0.4 (AUC = 0.825)
Dataset 0.5 (AUC = 0.800)
Dataset 0.6 (AUC = 0.772)
Dataset 0.7 (AUC = 0.758)
Dataset 0.8 (AUC = 0.742)
Dataset 0.9 (AUC = 0.691)
Dataset 1.0 (AUC = 0.658)
Random Guess


ROC Curves for XGBoost
True Positive Rate
False Positive Rate
Dataset 0.0 (AUC = 0.869)
Dataset 0.1 (AUC = 0.861)
Dataset 0.2 (AUC = 0.852)
Dataset 0.3 (AUC = 0.837)
Dataset 0.4 (AUC = 0.824)
Dataset 0.5 (AUC = 0.800)
Dataset 0.6 (AUC = 0.773)
Dataset 0.7 (AUC = 0.757)
Dataset 0.8 (AUC = 0.740)
Dataset 0.9 (AUC = 0.686)
Dataset 1.0 (AUC = 0.655)
Random Guess

Performance Comparison



The figure illustrates how model performance (AUC) decreases as α increases, reflecting greater covariate shift.

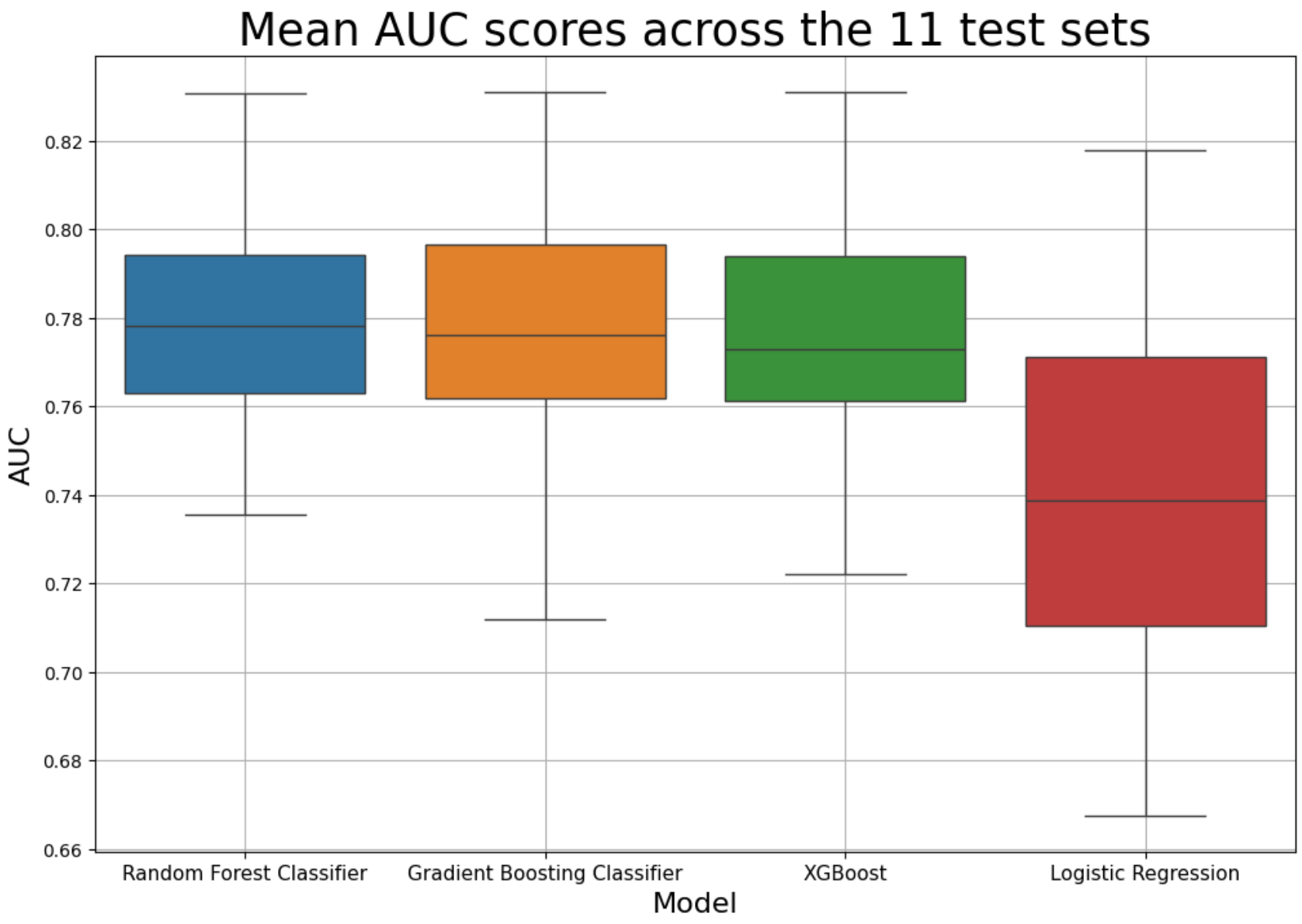
Ensemble models demonstrate higher robustness compared to the Logistic Regression baseline.

Statistical Performance Comparison

To give statistical support to our study we repeated this experiment $N = 50$ times. Keeping always the same training set (in order to not train the models several times), for each repetition we:

1. defined a new shifted ditribution $\mathcal{N}(\boldsymbol{\mu}_{\text{shift}}, \boldsymbol{\Sigma}_{\text{shift}})$
2. created 11 **testing** datasets $\mathcal{D}_\alpha$ with $\alpha \in \{0.0, 0.1, \ldots, 1.0\}$, where α represents the mixing probability as before
3. computed the ROC-AUC score for each model on each testing dataset

Statistical Performance Comparison



Statistical Performance Comparison

Performance Enhancement

A Novel Pre-Processing Method Against Covariate Shift

1. **No Prior Shift Knowledge Needed**
   - Simplifies implementation by eliminating the need for shift estimation.
   - Adaptable to various datasets without additional shift information.
2. **Built-in Regularization**
   - Prevents overfitting by introducing controlled noise.
   - Enhances model generalization on unseen data.

The R.A.W. Method

- Random
- Augmentation
- Walk

Input: $Data_{\text{train}}$, $Size$, N, ε.

$Data_{\%} \leftarrow$random subset of N% of Data_{train}

For x_i in $\text{Data}_{\%}$

$$x_i' \leftarrow \begin{cases} X_i + \varepsilon & \text{with probability 0.5} \\ X_i - \varepsilon & \text{with probability 0.5} \end{cases}$$

$y_i' \leftarrow y_i$

End For

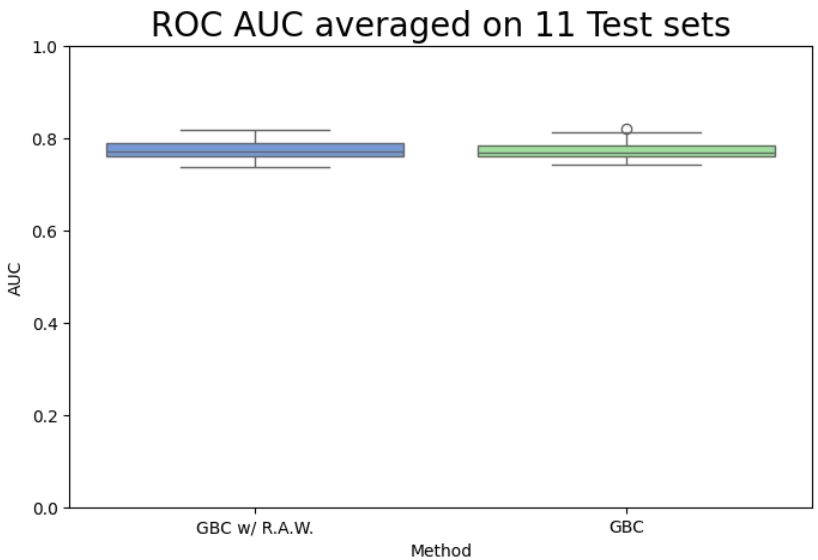
$Data_{\text{aug}} \leftarrow \text{Data}_{train} \cup Data_{\%}$

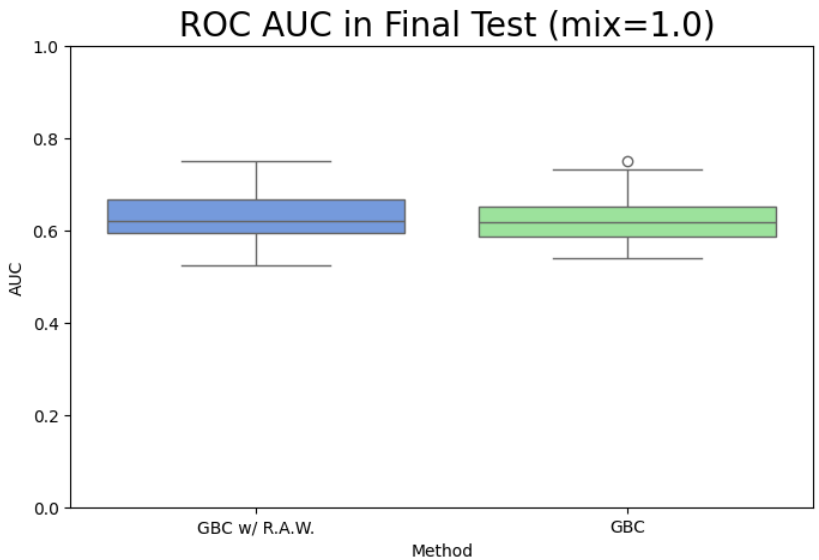
$Data_{\text{final}} \leftarrow \text{Downsample}(\text{Data}_{\text{aug}}, Size)$

Return $Data_{\text{final}}$

Classify With Gradient Boosting Using R.A.W.

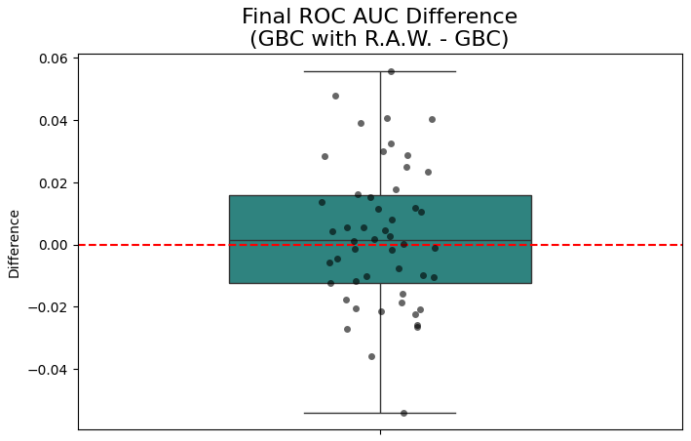
1. Apply the R.A.W. pre-processing method to the training data to address covariate shift.
2. Train a Gradient Boosting Classifier on the augmented dataset.
3. Evaluate the model's performance on shifted test sets.





A Statistical Analysis Of The Results

- H_0:
  $\Delta_{AUC} = AUC_{\text{R.A.W.}} - AUC_{\text{base}} = 0$
- H_1:
  $\Delta_{AUC} = AUC_{\text{R.A.W.}} - AUC_{\text{base}} \neq 0$
- **Test:** Student's t-Test on 50 independent Δ_{AUC}.



	Δ_{AUC}	t-stat	p-value	95% CI
$\Delta_{\overline{\text{AUC}}}$*	0.0003	0.224	0.82	[-0.0023, 0.0029]
$\Delta_{\text{AUC}_{\text{last}}}$**	0.0027	0.841	0.404	[-0.0038, 0.0093]

* Mean AUC score difference across all 11 shifted test sets.

** AUC score difference on the most shifted test set.

Conclusion

How can we address the initial questions posed?

- Covariate shift was explored via synthetic data and repeated simulations.
- Performance degradation highlights the need for ”fine-tuned” robust models.
- Every tuned model performed equally across each shift, with the exception of the Logistic Regression.
- **Future work**: What happens if we use Real World datasets? How to further improve the R.A.W. algorithm?

Thank You!