The Importance of Being Calibrated: A Study on Probability Calibration and Interval Estimation for Binary Classification

Sandro Khizanishvili[1] Sezer Mezgil[2] and Elisa Terzini[3]

Abstract— **The ability of machine learning models to output well-calibrated probabilities—where a predicted probability of p corresponds to an observed event frequency of p—is critical for high-stakes decision-making in fields like credit risk assessment. While powerful classifiers like XGBoost excel at discrimination, their probability estimates are often poorly calibrated, leading to overconfident and unreliable predictions.**

This study presents a comprehensive empirical evaluation of probability calibration methods—Platt Scaling, Isotonic Regression, and Venn-Abers Predictors (IVAP and CVAP)—applied to Logistic Regression and XGBoost models on a real-world credit risk dataset. To systematically assess performance under controlled conditions, we supplemented the real-data analysis with a novel simulation framework generating miscalibrated scores from mixed Beta distributions across 18 scenarios varying dataset size and class imbalance.

Our results confirm that calibration is a distinct necessity separate from accuracy. We found that while Logistic Regression is often naturally better calibrated, XGBoost requires post-processing to achieve reliability. Among the methods, Venn-Abers predictors proved to be highly competitive, offering distribution-free calibration guarantees. Crucially, they provided a unique secondary benefit: the generation of significantly tighter and more efficient prediction intervals (5-8x narrower on average) compared to traditional bootstrap methods, while maintaining statistical validity.

This work underscores that practitioners must actively evaluate and calibrate model probabilities to ensure trustworthiness. We demonstrate that Venn-Abers is a superior choice for applications demanding precise uncertainty quantification, providing a robust framework for trustworthy machine learning in risk-sensitive domains.

I. Introduction

When a model assigns a 20% probability to the positive class, what does this value actually represent?
In a well-calibrated model, such predictions should be statistically reliable: among all instances predicted with a 20% probability, roughly one in five should indeed belong to the positive class.
Calibration therefore reflects the degree to which predicted probabilities correspond to observed frequencies.

However, many state-of-the-art machine learning algorithms,such as Support Vector Machines (SVM), Gradient Boosting Machines (GBM), Random Forests, and Logistic Regression, are primarily designed as scoring classifiers.
They are highly effective at discrimination, meaning they can successfully rank instances by relative risk, but their raw probability outputs are often poorly calibrated.

This project addresses credit risk classification and the calibration of predicted probabilities. Using a real credit risk dataset (default = 1, non-default = 0), we trained Logistic Regression and XGBoost models and obtained raw (uncalibrated) probability outputs. To improve reliability, we applied several calibration methods: Platt Scaling, Isotonic Regression, and Venn–Abers Calibration (both IVAP and CVAP).

In addition to real data experiments, we designed a simulation framework where scores were generated from mixtures of Beta distributions to capture both accurate predictions and systematic miscalibration. For the negative class, 70% of samples came from Beta$(2, 8)$ and 30% from Beta$(7, 3)$, while for the positive class, 70% came from Beta$(8, 2)$ and 30% from Beta$(3, 7)$. We varied dataset size (1,000, 10,000, 50,000) and class imbalance (positive prevalence of 5–50%), yielding 18 simulation scenarios. Performance was evaluated using discrimination (ROC-AUC), calibration (Log-Loss, Brier Score, ECE), and reliability diagrams.
This dual approach, real-world modeling and controlled simulation, allowed us to systematically evaluate the strengths and limitations of different calibration methods in credit risk prediction.

II. Dataset Preprocessing

Before applying any models, the dataset (the publicly available *Credit Risk Dataset* from Kaggle), was carefully examined and preprocessed to ensure data quality, consistency, and suitability for predictive modeling. Preprocessing is a crucial step, as raw data often contains inconsistencies, missing values, outliers, and redundant information that can distort model training and evaluation.

The dataset used in this study (Table I) consists of 32,581 entries and 11 features, including both categorical and numerical variables. A systematic approach was applied to handle categorical encodings, outlier detection and treatment, missing value imputation, and feature selection. This process ensures that the derived training, validation, and test sets are robust, representative, and free from data leakage, providing a reliable foundation for subsequent modeling and performance evaluation.

[1]Sandro Khizanishvili, 2175979, Sapienza University of Rome.
[1]Sezer Mezgil, 2180600, Sapienza University of Rome.
[2]Elisa Terzini, 1982602, Sapienza University of Rome.

TABLE I
DATASET FEATURES (32,581 ENTRIES, 12 VARIABLES)
4 CATEGORICAL VALUES AND 8 NUMERICAL VALUES.
LOAN_STATUS IS THE TARGET VARIABLE

Column	Description	Type
person_age	Person's age	int64
person_income	Annual income	int64
person_home_ownership	Home ownership status	object
person_emp_length	Employment length (years)	float64
loan_intent	Loan purpose	object
loan_grade	Credit rating assigned to loan	object
loan_amnt	Loan amount requested	int64
loan_int_rate	Loan interest rate	float64
loan_status	Loan outcome (default/no default)	int64
loan_percent_income	Loan as % of income	float64
cb_person_default_on_file	Previous default on record	object
cb_person_cred_hist_length	Credit history length (years)	int64

A. Dataset Split

The dataset was randomly split into training (64%), validation (16%), and test (20%) subsets (Fig. 1) using a fixed random seed for reproducibility. Each subset was saved as a separate CSV file.

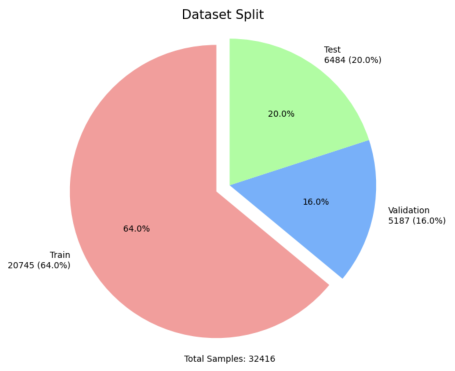


Fig. 1. Dataset split

B. Categorical Feature Encoding

Categorical features were encoded using a target-based label encoding approach, where each category is assigned a numerical value based on the mean of the target variable (loan_status), representing the relative risk associated with that category (Tables II, III, IV and V show the target mean values, category frequencies, and the corresponding label encodings for each categorical variable used in the model.). Categories with similar risk profiles were grouped together, and small groups with few observations were merged to avoid statistical insignificance. This method captures the predictive information in categorical features while keeping the encoding consistent across training, validation, and test sets.

TABLE II
MEAN TARGET, COUNT, AND ENCODING FOR PERSON_HOME_OWNERSHIP.

Category	Mean	Count	Mapping
OWN	0.077	1,616	1
MORTGAGE	0.131	8,587	2
OTHER	0.299	67	3
RENT	0.316	10,475	3

TABLE III
MEAN TARGET, COUNT, AND ENCODING FOR LOAN_INTENT.

Category	Mean	Count	Mapping
VENTURE	0.149	3,612	1
EDUCATION	0.172	4,165	2
PERSONAL	0.208	3,550	3
HOMEIMPROVEMENT	0.260	2,270	4
MEDICAL	0.267	3,831	5
DEBTCONSOLIDATION	0.291	3,317	6

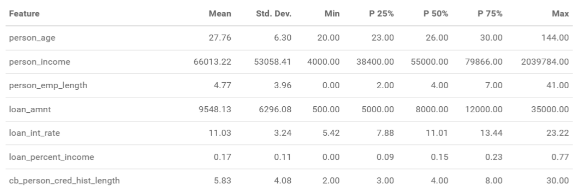
The preprocessed and encoded training, validation, and test datasets were saved to CSV files for reproducibility.

C. Numerical Feature Processing

After reviewing the summary statistics of our numerical features (Figure 2), we identified several variables with highly skewed distributions and potential outliers. For example, (person_age) has a maximum value of 144, which is an impossible age and clearly a data entry error. Similarly, (person_income) has a maximum value that is drastically higher than its 75th percentile, indicating the presence of extreme values.

To mitigate the impact of these outliers, we applied a common treatment technique known as *percentile capping*. For each numerical feature, we capped the values that fall below the 1st percentile or above the 99th percentile. This means any value lower than the 1st percentile is replaced by the value at the 1st percentile, and any value higher than the 99th percentile is replaced by the value at the 99th percentile.

Crucially, to prevent data leakage, these percentile bounds were calculated using only the training data and then applied consistently across both the training and test sets. This protocol is essential for ensuring that our final model evaluation is unbiased and provides a true measure of its performance on unseen data.



Feature	Mean	Std. Dev.	Min	P 25%	P 50%	P 75%	Max
person_age	27.76	6.30	20.00	23.00	26.00	30.00	144.00
person_income	66013.22	53058.41	4000.00	38400.00	55000.00	79866.00	2039784.00
person_emp_length	4.77	3.96	0.00	2.00	4.00	7.00	41.00
loan_amnt	9548.13	6296.08	500.00	5000.00	8000.00	12000.00	35000.00
loan_int_rate	11.03	3.24	5.42	7.88	11.01	13.44	23.22
loan_percent_income	0.17	0.11	0.00	0.09	0.15	0.23	0.77
cb_person_cred_hist_length	5.83	4.08	2.00	3.00	4.00	8.00	30.00

Fig. 2. Numerical Feature Summary

Missing values were imputed based on `loan_grade`, as it was considered the most informative feature for capturing a customer's risk profile.

TABLE IV

MEAN TARGET, COUNT, AND ENCODING FOR `LOAN_GRADE`.

Category	Mean	Count	Mapping
A	0.102	6,784	1
B	0.167	6,710	2
C	0.210	4,080	3
D	0.577	2,345	4
E	0.636	632	4
F	0.709	151	4
G	1.000	43	4

TABLE V

MEAN TARGET, COUNT, AND ENCODING FOR `CB_PERSON_DEFAULT_ON_FILE`.

Category	Mean	Count	Mapping
N	0.187	17,094	1
Y	0.375	3,651	2

D. Feature Selection

The predictive power of each feature was assessed by examining its relationship with the target variable, `loan_status`. For categorical features, the mean default rate was computed per category, while numerical features were discretized into deciles and the mean default rate calculated for each bin. These relationships were visualized in a grid of plots, with consistent y-axis limits to facilitate comparison. Point-biserial correlation coefficients were also calculated for each feature, providing a quantitative measure of association with the target.

A correlation matrix (Fig. 3) was computed for all numerical and encoded categorical features to identify multicollinearity. It was found that `loan_int_rate` and `loan_grade` are highly correlated, which is expected because the assigned grade directly influences the interest rate.

Similarly, `cb_person_cred_hist_length` and `person_age` exhibit high correlation due to redundancy, as older individuals typically have longer credit histories. For modeling purposes, `cb_person_cred_hist_length` and `loan_grade` was retained while `person_age` and `loan_int_rate` were dropped, as the former provides a more direct measure of creditworthiness.

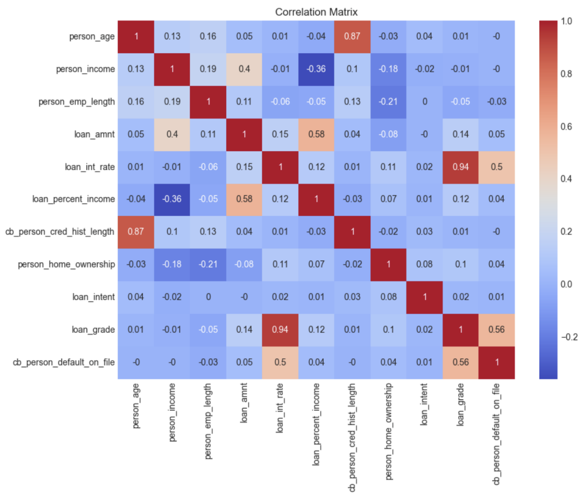


Fig. 3. Correlation Matrix

Besides the correlation analysis, the Population Stability Index (PSI) was calculated to assess the stability of feature distributions between the training and validation datasets. For numerical features, values were divided into bins, and the distribution across bins was compared between datasets. For categorical features, normalized value counts were compared directly.

PSI interpretation guidelines:

- PSI < 0.1: No significant change (stable).
- $0.1 \leq$ PSI < 0.25: Moderate change.
- PSI ≥ 0.25: Significant change (feature instability).

All features demonstrated PSI values < 0.1, indicating high stability across datasets. Therefore, no additional features were removed at this stage.

The final preprocessed datasets, containing only stable and non-redundant features, were saved for subsequent modeling:

preprocessed_train.csv
preprocessed_valid.csv
preprocessed_test.csv

III. METHODOLOGY

Our methodology is organized into two main stages: (i) Logistic Regression and XGBoost model training and metric evaluation, and (ii) Logistic Regression and XGBoost model calibration

evaluation of predictive performance and stability. This systematic approach ensures both the reliability of the input data and the robustness of the resulting models.

A. Logistic Regression and XGBoost model training and metric evaluation

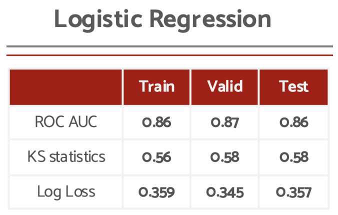
Two base classifiers were employed: Logistic Regression and XGBoost. These models were chosen to contrast a linear, interpretable baseline with a powerful non-linear ensemble learner.

For both models metrics such as:

- **ROC-AUC:** The Area Under the Receiver Operating Characteristic Curve measures a model's ability to discriminate between classes. It represents the probability that a randomly chosen positive instance is ranked higher than a randomly chosen negative one. Values closer to 1 indicate better discrimination.

– **KS statistic:** The Kolmogorov–Smirnov statistic quantifies the maximum separation between the cumulative distributions of predicted scores for positives and negatives. Higher KS values indicate stronger class separation, with values near 0 suggesting poor discrimination.
– **Log-loss:** Also known as cross-entropy loss, it evaluates the accuracy of predicted probabilities. It penalizes confident but incorrect predictions more heavily, meaning lower values correspond to better-calibrated probability estimates.

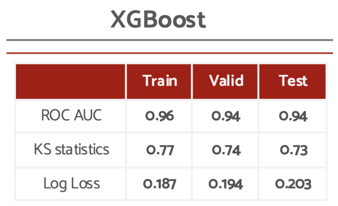
were used on the training, validation and test set.
Figure 5 shows Logistic Regression's metrics, while Figure 5 shows the XGBoost's metrics.



Logistic Regression

	Train	Valid	Test
ROC AUC	0.86	0.87	0.86
KS statistics	0.56	0.58	0.58
Log Loss	0.359	0.345	0.357

Fig. 4. Training, Validation and Test set Metrics for the Logistic Regression Model.



XGBoost

	Train	Valid	Test
ROC AUC	0.96	0.94	0.94
KS statistics	0.77	0.74	0.73
Log Loss	0.187	0.194	0.203

Fig. 5. Training, Validation and Test set Metrics for the XGBoost Model.

Two plots are shown as model performance analysis plots, Figure 6 for the test set of Logistic Regression and Figure 7 for the test set of the XGBoost Model.

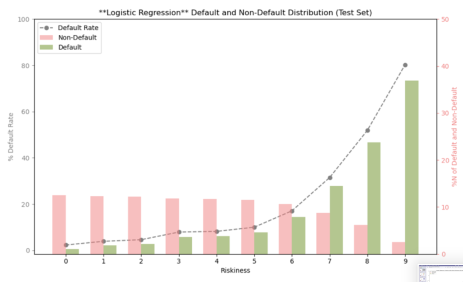


Fig. 6. Logistic Regression Default and Non-Default Distribution (Test set).

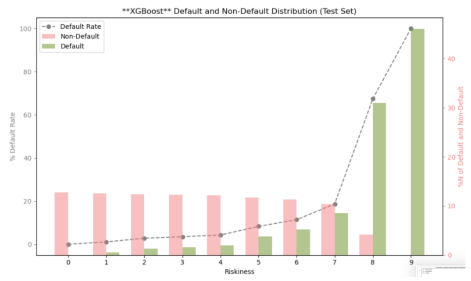


Fig. 7. XGBoost Default and Non-Default Distribution (Test set).

The model performance plots illustrate how well the model separates defaults from non-defaults across different levels of predicted risk. The x-axis represents deciles of predicted probabilities, where observations are grouped into ten bins from low to high risk. The gray dashed line shows the observed default rate within each bin, providing insight into whether higher predicted scores align with higher actual default rates. The bar chart displays the distribution of defaults (green) and non-defaults (red) across the bins, indicating how well the model concentrates defaults in high-risk groups and non-defaults in low-risk groups. An effective model will show a steadily increasing default rate from left to right, with defaults concentrated in the higher bins and non-defaults in the lower bins. This visualization therefore provides a combined view of both calibration (the alignment of predicted probabilities with observed outcomes) and discrimination (the model's ability to distinguish between defaults and non-defaults).
While both models are effective discriminators, they often lack probability calibration. To address this, we applied and compared multiple calibration techniques.

B. Logistic Regression and XGBoost model calibration

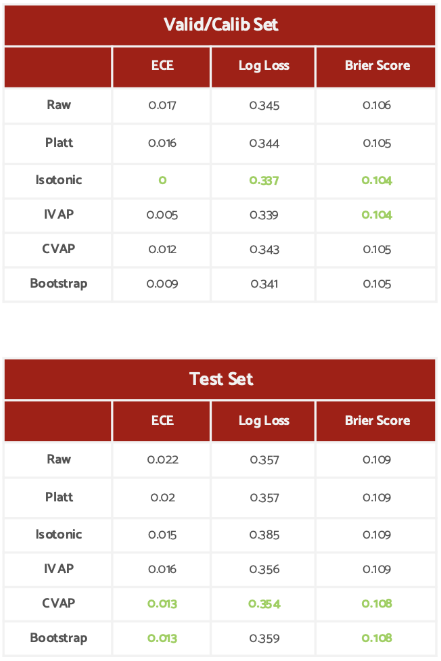
Four types of Calibration methods were used on the Logistic Regression model (Figure 8 and Figure 9) and on the XGBoost model (Figure 10 and 11):

- **Platt Scaling** (sigmoid-based parametric calibration),
- **Isotonic Regression** (non-parametric calibration),
- **Inductive and Cross Venn-Abers Predictors** (theoretically valid multi-probabilistic calibration methods).

Additionally, bootstrap-based interval estimation was employed to provide uncertainty quantification around probability predictions, complementing the Venn-Abers approach.

Model performance was assessed along three dimensions:

- **ECE (Expected Calibration Error):** A calibration metric that measures the average difference between predicted probabilities and observed frequencies across bins of predictions. Lower ECE indicates better alignment between predicted risk and actual outcomes.
- **Log Loss:** as described above.
- **Brier Score:** A proper scoring rule that computes the mean squared error between predicted probabilities and actual outcomes. It captures both calibration and discrimination, with lower values reflecting better probabilistic predictions.



Valid/Calib Set			
	ECE	Log Loss	Brier Score
Raw	0.017	0.345	0.106
Platt	0.016	0.344	0.105
Isotonic	0	0.337	0.104
IVAP	0.005	0.339	0.104
CVAP	0.012	0.343	0.105
Bootstrap	0.009	0.341	0.105

Test Set			
	ECE	Log Loss	Brier Score
Raw	0.022	0.357	0.109
Platt	0.02	0.357	0.109
Isotonic	0.015	0.385	0.109
IVAP	0.016	0.356	0.109
CVAP	0.013	0.354	0.108
Bootstrap	0.013	0.359	0.108

Fig. 8. Calibration Metrics on Logistic Regression Model.

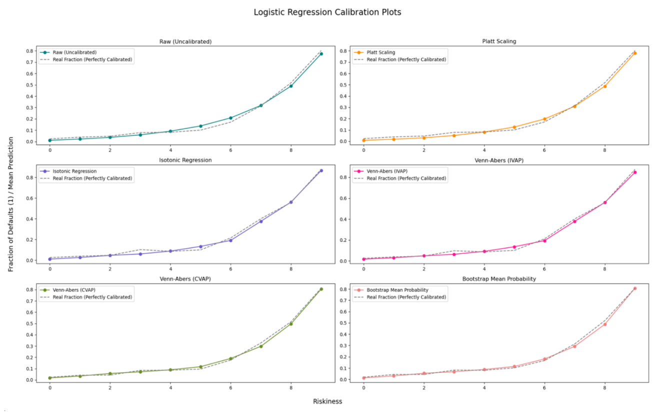
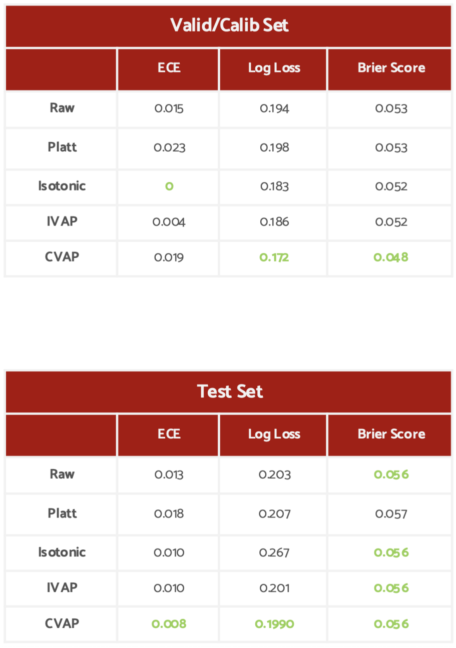


Fig. 9. Calibration Plots on Logistic Regression Model.



Valid/Calib Set			
	ECE	Log Loss	Brier Score
Raw	0.015	0.194	0.053
Platt	0.023	0.198	0.053
Isotonic	0	0.183	0.052
IVAP	0.004	0.186	0.052
CVAP	0.019	0.172	0.048

Test Set			
	ECE	Log Loss	Brier Score
Raw	0.013	0.203	0.056
Platt	0.018	0.207	0.057
Isotonic	0.010	0.267	0.056
IVAP	0.010	0.201	0.056
CVAP	0.008	0.1990	0.056

Fig. 10. Calibration Metrics on XGBoost Model.

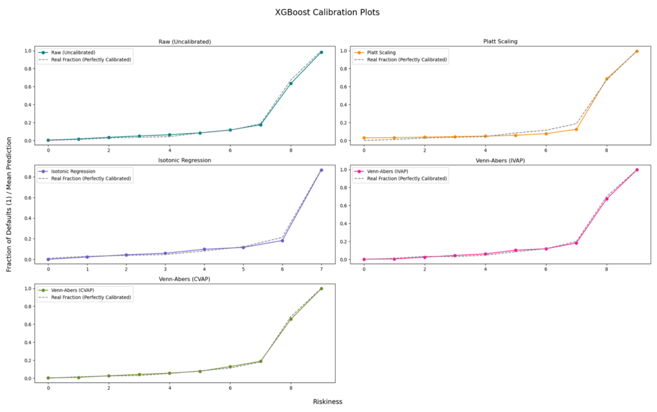


Fig. 11. Calibration Plots on XGBoost Model.

By combining classical classifiers with advanced calibration methods and stability checks, we ensure that the final predictive models are not only accurate but also trustworthy in their probabilistic estimates, which is essential for credit risk assessment.

C. Simulated Data Generation

In our simulations, we generated predicted probabilities for two classes by mixing different Beta distributions.
For the **negative class (Class 0)**:

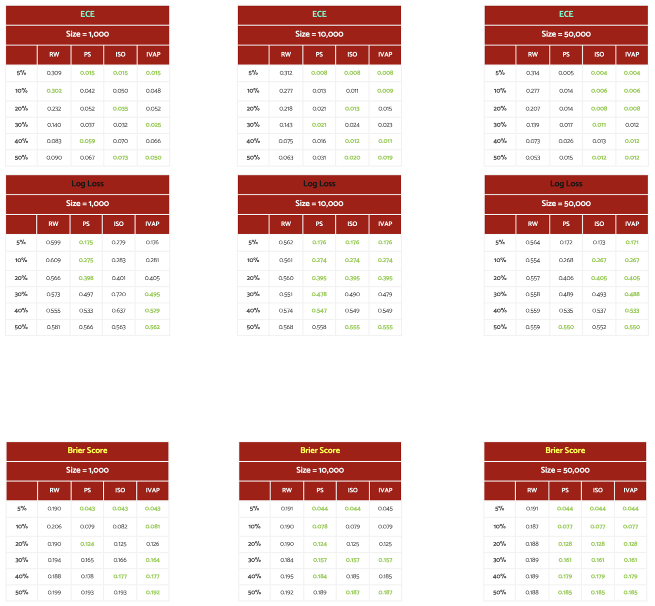
- 70% of the samples were drawn from a Beta($\alpha = 2, \beta = 8$) distribution.
- The remaining 30% were drawn from a Beta($\alpha = 7, \beta = 3$) distribution.

For the **positive class (Class 1)**:

- 70% of the samples came from a Beta($\alpha = 8, \beta = 2$) distribution.
- while 30% came from a Beta($\alpha = 3, \beta = 7$) distribution.

We varied the **dataset size** (1,000, 10,000, and 50,000) and the **class imbalance** (positive class prevalence of 5%, 10%, 20%, 30%, 40%, and 50%).
This produced a comprehensive grid of 18 simulation scenarios (3 dataset sizes $\times$ 6 prevalence levels), enabling us to systematically evaluate how Platt Scaling, Isotonic Regression, and Venn-Abers calibration perform across different levels of data availability and class imbalance, while accounting for realistic patterns of miscalibration.

Results on the simulated data can be observed in Figure 12



ECE				
Size = 1,000				
	RW	PS	ISO	IVAP
5%	0.309	0.015	0.015	0.015
10%	0.302	0.042	0.050	0.048
20%	0.232	0.052	0.035	0.052
30%	0.140	0.037	0.032	0.025
40%	0.083	0.059	0.070	0.066
50%	0.090	0.067	0.073	0.050

ECE				
Size = 10,000				
	RW	PS	ISO	IVAP
5%	0.312	0.008	0.008	0.008
10%	0.277	0.013	0.011	0.009
20%	0.218	0.021	0.013	0.015
30%	0.143	0.021	0.024	0.023
40%	0.075	0.016	0.012	0.011
50%	0.063	0.031	0.020	0.019

ECE				
Size = 50,000				
	RW	PS	ISO	IVAP
5%	0.314	0.005	0.004	0.004
10%	0.277	0.014	0.006	0.006
20%	0.207	0.014	0.008	0.008
30%	0.139	0.017	0.011	0.012
40%	0.073	0.026	0.013	0.012
50%	0.053	0.015	0.012	0.012

Log Loss				
Size = 1,000				
	RW	PS	ISO	IVAP
5%	0.599	0.175	0.279	0.176
10%	0.609	0.275	0.283	0.281
20%	0.566	0.398	0.401	0.405
30%	0.573	0.497	0.720	0.495
40%	0.555	0.533	0.637	0.529
50%	0.581	0.566	0.563	0.562

Log Loss				
Size = 10,000				
	RW	PS	ISO	IVAP
5%	0.562	0.176	0.176	0.176
10%	0.561	0.274	0.274	0.274
20%	0.560	0.395	0.395	0.395
30%	0.551	0.478	0.490	0.479
40%	0.574	0.547	0.549	0.549
50%	0.568	0.558	0.555	0.555

Log Loss				
Size = 50,000				
	RW	PS	ISO	IVAP
5%	0.564	0.172	0.173	0.171
10%	0.554	0.268	0.267	0.267
20%	0.557	0.406	0.405	0.405
30%	0.558	0.489	0.493	0.488
40%	0.559	0.535	0.537	0.533
50%	0.559	0.550	0.552	0.550

Brier Score				
Size = 1,000				
	RW	PS	ISO	IVAP
5%	0.190	0.043	0.043	0.043
10%	0.206	0.079	0.082	0.081
20%	0.190	0.124	0.125	0.126
30%	0.194	0.165	0.166	0.164
40%	0.188	0.178	0.177	0.177
50%	0.199	0.193	0.193	0.192

Brier Score				
Size = 10,000				
	RW	PS	ISO	IVAP
5%	0.191	0.044	0.044	0.045
10%	0.190	0.078	0.079	0.079
20%	0.190	0.124	0.125	0.125
30%	0.184	0.157	0.157	0.157
40%	0.195	0.184	0.185	0.185
50%	0.192	0.189	0.187	0.187

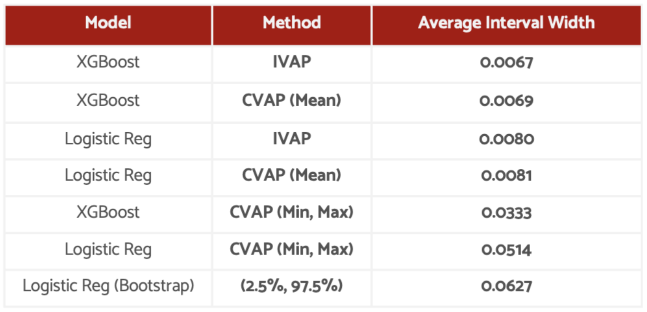
Brier Score				
Size = 50,000				
	RW	PS	ISO	IVAP
5%	0.191	0.044	0.044	0.044
10%	0.187	0.077	0.077	0.077
20%	0.188	0.128	0.128	0.128
30%	0.189	0.161	0.161	0.161
40%	0.189	0.179	0.179	0.179
50%	0.188	0.185	0.185	0.185

Fig. 12. Simulated dataset results on test set.

D. Probability Intervals: Bootstrap vs. Venn-Abers

E. Probability Intervals

To quantify the uncertainty associated with predicted probabilities, we evaluated different interval estimation methods for both Logistic Regression and XGBoost. The results, summarized in Table 13, show that Inductive Venn-Abers Predictors (IVAP) produced the narrowest intervals, with average widths of 0.0067 for XGBoost and 0.0080 for Logistic Regression. Cross Venn-Abers Predictors (CVAP) using mean aggregation yielded slightly wider intervals (0.0069 and 0.0081, respectively), but still provided highly precise uncertainty estimates. In contrast, CVAP intervals based on minimum–maximum aggregation were considerably wider (0.0333 for XGBoost and 0.0514 for Logistic Regression), reflecting a more conservative characterization of uncertainty. Finally, bootstrap-based intervals (2.5%–97.5%) for Logistic Regression were the widest overall (0.0627), highlighting the trade-off between coverage guarantees and interval tightness. These results demonstrate that Venn-Abers predictors provide sharper and more informative probability intervals compared to traditional resampling methods, making them well-suited for applications requiring calibrated confidence estimates in credit risk prediction.



Model	Method	Average Interval Width
XGBoost	IVAP	0.0067
XGBoost	CVAP (Mean)	0.0069
Logistic Reg	IVAP	0.0080
Logistic Reg	CVAP (Mean)	0.0081
XGBoost	CVAP (Min, Max)	0.0333
Logistic Reg	CVAP (Min, Max)	0.0514
Logistic Reg (Bootstrap)	(2.5%, 97.5%)	0.0627

Fig. 13. Probability intervals.

IV. Conclusions and Key Takeaways

Calibration is a Crucial, Separate Property from Accuracy: a model can be highly accurate yet poorly calibrated, making its probability scores misleading and untrustworthy for real-world risk assessment. Explicitly measuring and improving calibration is essential for any application relying on probabilistic predictions.
The Best Calibration Method is Context-Dependent: Platt Scaling: A good, fast default for simple models like Logistic Regression.
Isotonic Regression: Powerful but can overfit on small datasets. Excellent for larger, well-behaved datasets.
Venn-Abers Predictors: Provide robust, distribution-free calibration guarantees and are highly competitive, especially on complex models like XGBoost. They offer a unique advantage: inherently valid probability intervals.
For Precise Uncertainty Quantification, Venn-Abers is Superior: Our results demonstrate that Venn-Abers predictors (IVAP and CVAP mean) generate prediction intervals that are significantly tighter than traditional bootstrap methods, while maintaining validity. This makes them ideal for applications requiring precise uncertainty estimates.

References

[1] Vladimir Vovk and Ivan Petej, Venn–Abers Predictors - https://arxiv.org/pdf/1211.0025/
[2] Vladimir Vovk, Ivan Petej, and Valentina Fedorova, Large-scale probabilistic prediction with and without validity guarantees - https://arxiv.org/pdf/1511.00213
[3] Venn-Abers implementation - https://github.com/ip200/venn-abers