Why Would a Credit Officer Want a Probability Instead of a Rule?

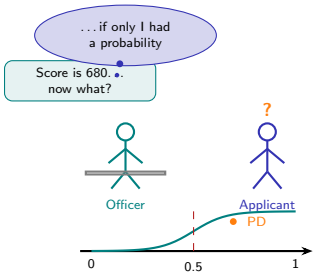
The Dilemma

- A loan applicant scores 680 – right at the boundary
- The rulebook says "approve above 700, deny below 650" but says nothing about 680
- A linear model predicts repayment of 1.3 – but what does that even mean?

What if instead of a number, you got the probability of default?

Insight

Logistic regression replaces arbitrary cutoffs with calibrated probabilities: $P(\text{default})$ maps every applicant to a number between 0 and 1.



Logistic regression outputs $P(Y{=}1|X)$ – a calibrated probability, not an unbounded score

Predicting an Outcome – Did You Think in Probabilities?

Think Before You Compute

Imagine you are reviewing a stack of loan applications. Without any model, you instinctively sort them: this one looks safe, that one is risky, this one could go either way. You are not assigning 0 or 1. You are assigning a feeling of likelihood – "probably fine", "probably not", "fifty-fifty". That is a probability.

- How confident were you in each assessment?
- Did you use features like income, employment, or debt?
- Were some features more important than others?

Pause and reflect:

When you last decided whether to bring an umbrella, you estimated $P(\text{rain})$ from features: clouds, forecast, season. You did not predict rain $= 1.3$.

That is logistic regression.

Reflection Prompt

Write down one real decision you made this week where you mentally estimated a probability. What features did you use?

Human intuition naturally produces probabilities – logistic regression formalizes this into a trainable model

What Makes Logistic Regression Different from Linear Regression and Decision Trees?

Taxonomy of Classifiers

Property	LogReg	LinReg	Dec. Tree
Output	Prob.	Real	Class/Prob.
Boundary	Linear	N/A	Axis-aligned
Loss	Cross-ent.	MSE	Gini/Entropy
Interpret.	High (OR)	High	Medium
Regularize	L1/L2/EN	L1/L2	Pruning
Calibration	Native	Poor	Poor

Logistic regression is the only method here that directly outputs well-calibrated probabilities.

LogReg: Probabilistic, parametric, linear boundary

LinReg: Continuous output, no classification

Tree: Non-parametric, axis-aligned, uncalibrated

Insight

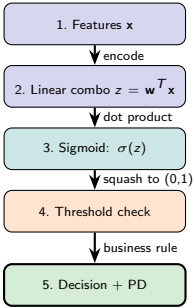
LogReg occupies a unique niche: parametric, interpretable, probabilistic, and naturally calibrated – which is why regulators love it.

Odds ratio interpretation: each coefficient tells you the multiplicative change in odds per unit feature change

How Does One Loan Application Travel from Features to Probability?

One Prediction, Step by Step

- Applicant arrives with features: income=50k, debt-ratio=0.35, employment=4yr
- Compute linear combination: $z = w_0 + w_1 \cdot \text{income} + w_2 \cdot \text{debt} + w_3 \cdot \text{empl.}$
- Apply sigmoid: $P(\text{default}) = \frac{1}{1+e^{-z}}$
- Compare to threshold (e.g., 0.5): if $P > 0.5$, predict default
- Bank uses the raw probability for capital reserves (Basel PD)



Insight

The sigmoid is the bridge: it maps any real number z to a probability in $(0, 1)$. The threshold is a business decision, not a model decision.

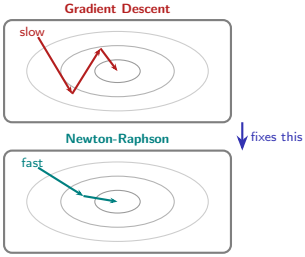
The model outputs P(default); the threshold is chosen by the business based on cost trade-offs

Who Should Fit the Model – Gradient Descent, Newton, or a Library?

Three Optimization Approaches

- **Gradient Descent**: update $\mathbf{w} = \mathbf{w} - \eta \frac{X^T(\mathbf{p}-\mathbf{y})}{n}$, simple but slow
- **Newton-Raphson (IRLS)**: uses Hessian, converges quadratically, standard in statsmodels/R
- **L-BFGS**: quasi-Newton, memory-efficient, default in scikit-learn

All three minimize the same convex cross-entropy loss – global optimum guaranteed.



Insight

Cross-entropy is convex in the weights, so every optimizer converges to the same solution. The choice is about speed, not correctness.

Newton-Raphson converges in 5–10 iterations vs. hundreds for gradient descent – but requires Hessian inversion

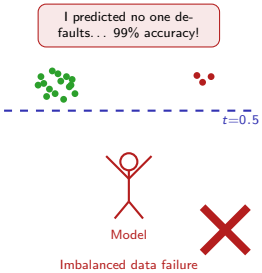
What Could Go Wrong If You Trust the Default Threshold?

Three Ways Logistic Regression Fails Silently

- **Class imbalance**: 99% non-default means predicting "no default" always gets 99% accuracy
- **Separation**: if one feature perfectly predicts the outcome, coefficients explode to infinity
- **Non-linear boundaries**: logistic regression draws a straight line – curved patterns get misclassified

Insight

The default 0.5 threshold is almost never optimal in finance. Fraud detection might use 0.1; credit scoring might use 0.3.



Always evaluate with AUC, precision-recall, and Gini – never with accuracy alone on imbalanced data

Why Does Every Credit Risk Team Start with Logistic Regression?

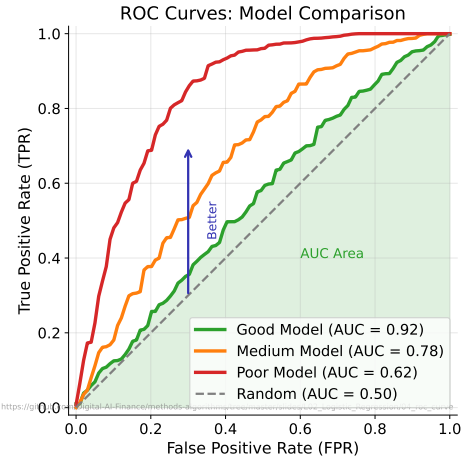
Logistic Regression as the Regulatory Baseline

- Interpretable: every coefficient has a direct odds-ratio interpretation
- Auditable: regulators can inspect and validate each feature's contribution
- Calibrated: output probabilities match observed default rates
- Stable: small data changes produce small coefficient changes

The ROC curve shows how well the model discriminates defaulters from non-defaulters at every threshold.

Insight

Logistic regression is not popular in credit scoring because it is the best predictor – it is popular because it is the most interpretable predictor.



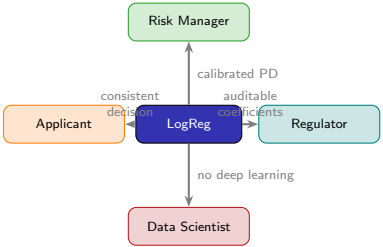
Gini $= 2 \cdot$ **AUC** $- 1$**: industry standard. Gini** > 0.40 **acceptable; Gini** > 0.60 **good for credit scoring**

Who Wins and Who Loses When a Bank Switches to Logistic Regression?

Stakeholder Analysis

- **Winners**: Risk managers (calibrated PD), regulators (interpretable model), applicants (consistent decisions), capital planning (accurate reserves)
- **Losers**: Data scientists wanting complex models, branch managers (discretion reduced), anyone with non-linear intuition

LogReg shifts power from subjective judgment to auditable probability.



Insight

In regulated finance, interpretability is not optional – it is a legal requirement under Basel II/III and GDPR Article 22.

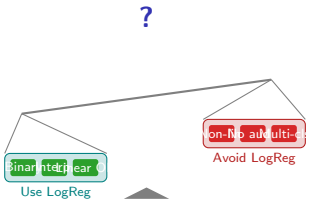
Basel IRB requires banks to demonstrate PD model validity annually – logistic regression passes this test

3 Questions That Reveal Whether Logistic Regression Is the Right Model

The Decision Framework

1. **Is the outcome binary (or ordinal)?** – If continuous, use linear regression; if multi-class, consider softmax
2. **Is interpretability required?** – If regulators demand coefficient explanations, LogReg wins
3. **Is the decision boundary approximately linear?** – If strongly non-linear, consider tree ensembles or feature engineering

If all three answers are yes, logistic regression is the right first model.



Insight

Logistic regression should be your first model for any binary classification problem in a regulated environment.

Even when you suspect non-linearity, start with LogReg as a baseline – you need something to beat

Can You Evaluate This Real Credit Scoring Problem?

The Scenario

A consumer bank wants to predict credit card default. Features: monthly income, credit utilization ratio, number of late payments (past 12 months), years with current employer, total debt. Dataset has 10,000 accounts with 3% default rate.

- Apply the 3-question framework from the previous slide
- Decide: Is logistic regression appropriate here?
- If yes: what threshold would you choose (hint: 3% default rate)?
- What single metric would you report to the regulator?

Deliverable

Fill in the table. Be prepared to defend your verdict to a skeptical Basel auditor.

Question	Your Answer
Binary outcome?	
Interpretable needed?	
Linear boundary OK?	
Verdict	
Recommended threshold	
Key metric for regulator	

Hint: with 3% default rate, accuracy is meaningless. Think about Gini, AUC, or precision-recall.