

BASICS OF NEURAL NETWORKS

Session: Model Evaluation



[credits](#)

2025 SCHOOL AT THE IAA-CSIC

EDUARDO SÁNCHEZ KARHUNEN

DEPT. ARTIFICIAL INTELLIGENCE. UNIV. SEVILLE. SPAIN

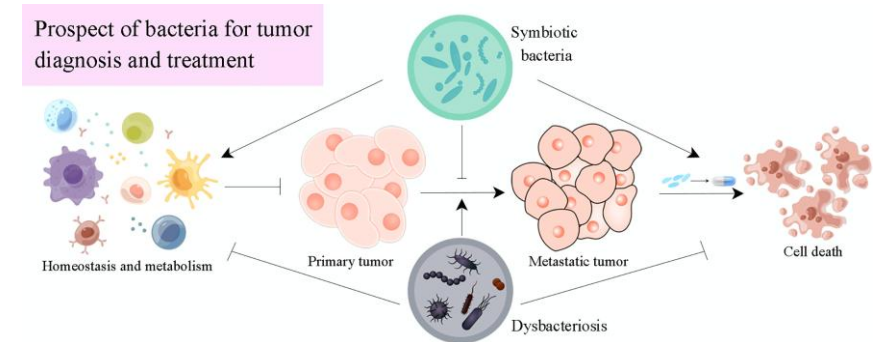
The Accuracy Paradox

► Accuracy can be misleading:

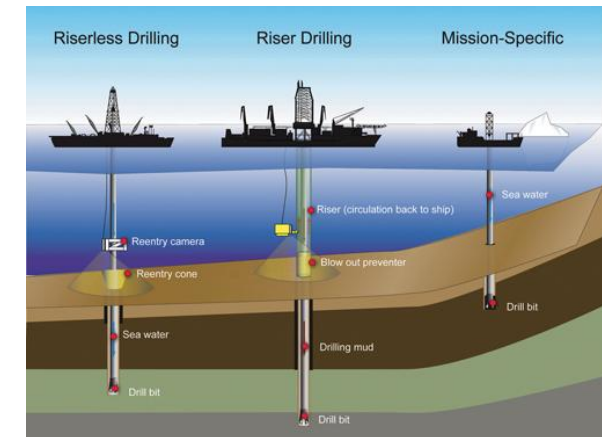
- Human first reaction: use accuracy.
- Maybe reliable: balanced dataset.
- Wrong tool: for imbalanced dataset.

► Rare cases detector:

- Disease affects 1 in 10,000.
- Model behavior:
 - Probably model will predict: All no disease.
 - Accuracy = 99,99% but model is completely useless.
 - Balance between accuracy and usefulness.



[credits](#)

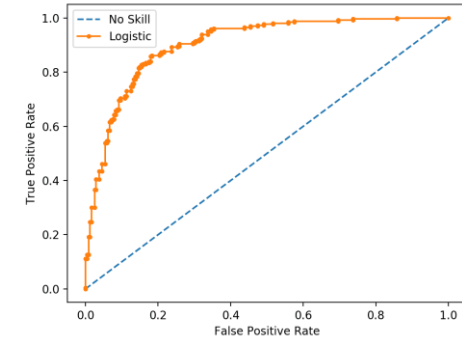


[credits](#)

Handling Imbalanced Datasets

► Changes in how to measure: Evaluation Metrics

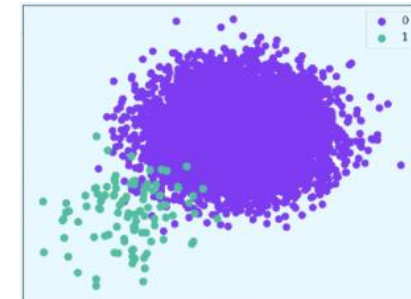
- Focus on the [confusion matrix](#).
- Prioritize [Precision & Recall](#).
- Use the [ROC Curve](#).



[credits](#)

► Changes in Data: Resampling Techniques

- [Augmentation](#) only in the minority class.
- Undersampling in majority class.



[credits](#)

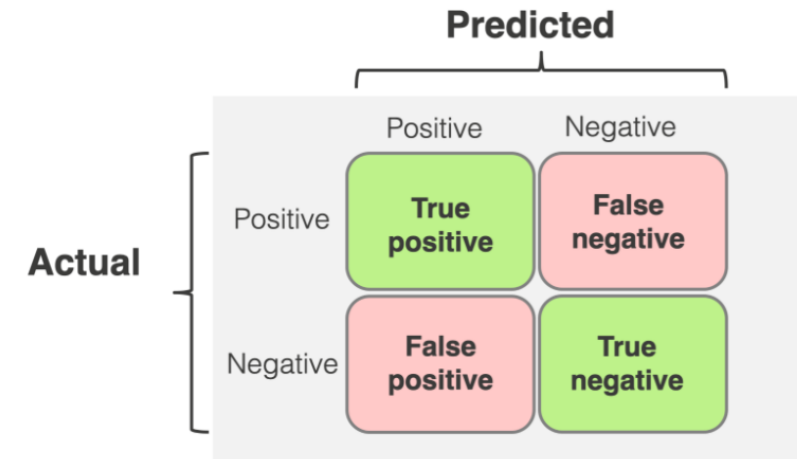
► Changes in model learning: Algorithmic approach

- [Class weights](#).
- [Loss functions](#) adapted for these cases

The Core Diagnostic Tool: Confusion Matrix

► Foundations of model evaluation:

- Moves from a single performance number:
 - To a detailed map.
- Analyze in which classes:
 - Model success and where it fails.
 - It is easier predict a dog as a cat than as a bird.
- True Positives & True Negatives:
 - Diagonal cells represents the correct predictions
- False Negative & False Positive:
 - Off diagonal, show which classes are confused easily.



		Expected			
		1	2	3	4
Predicted	1	52	3	7	2
	2	2	28	2	0
	3	5	2	25	12
	4	1	1	9	40

[credits](#)

The Key Diagnostic Metrics

► The Precision-Recall trade-off:

- Precision:

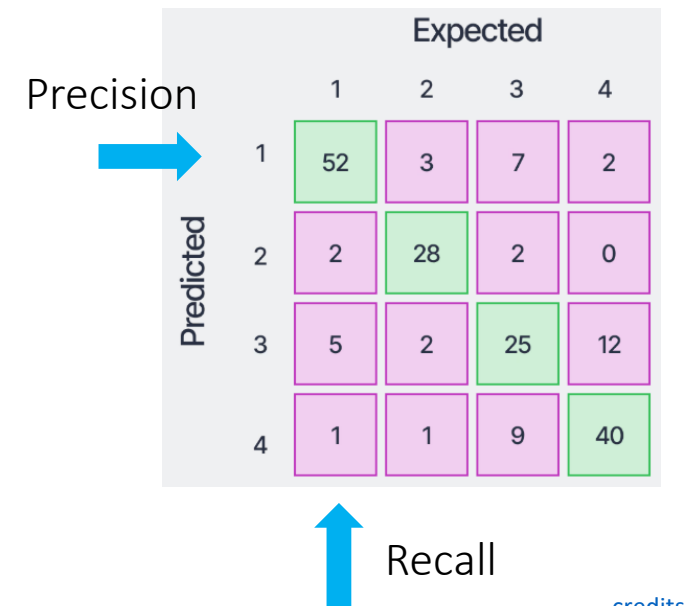
- Measures the **precision of the predictions**.
- When model makes a positive prediction, how often is it correct?

- Recall:

- Measures the **completeness of the predictions**.
- Of all the positive actual instances, how many did the model find?

- F1-score:

- Combines both metrics in a unique measure.
- It is a harmonic combination, high when both high.



A confusion matrix diagram illustrating Precision and Recall. The matrix is a 4x4 grid with 'Expected' values (1, 2, 3, 4) as columns and 'Predicted' values (1, 2, 3, 4) as rows. A blue arrow labeled 'Precision' points to the first column (Expected=1). A blue arrow labeled 'Recall' points to the first row (Predicted=1). The cells are colored: green for correct classifications (52, 28, 25, 40) and pink for misclassifications. The values in the cells are: Row 1: 52, 3, 7, 2; Row 2: 2, 28, 2, 0; Row 3: 5, 2, 25, 12; Row 4: 1, 1, 9, 40.

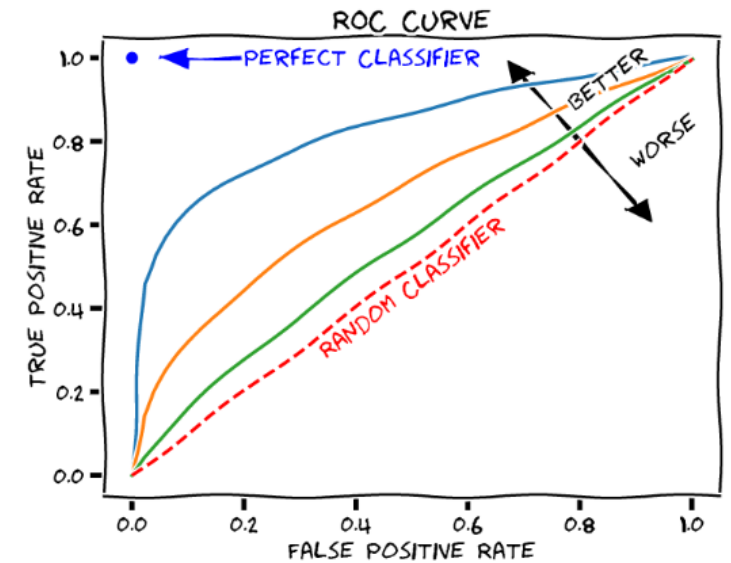
	Expected			
	1	2	3	4
Predicted 1	52	3	7	2
Predicted 2	2	28	2	0
Predicted 3	5	2	25	12
Predicted 4	1	1	9	40

[credits](#)

The ROC Curve

► When threshold moves:

- Goal: how classifier behaves as discrimination threshold varies.
- Measures plotted:
 - Recall (TPR), positives correctly identified.
 - FPR: negatives incorrectly identified as positive.
- Interpretation:
 - Ideal: $TPR = 1$ (all positives captured) and $FPR = 0$.
 - Better performance: the closer the curve to the top-left corner
- Summarize the curve across all thresholds: Area Under The Curve (AUC)
 - $AUC = 1$. Perfect classifier
 - $AUC < 0.5$. Worse than a random guess.

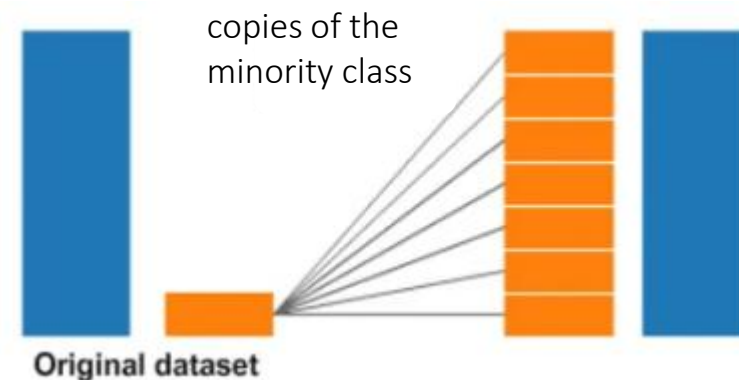


[credits](#)

Tackling Imbalance: Upsampling

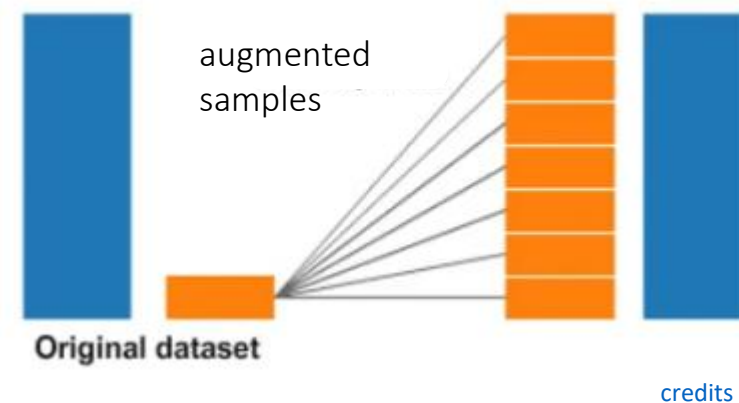
► Classical ML approach:

- **Random oversampling** of the minority class.
 - Exact same photos multiple times in the training.
 - Model sees the exact same images over and over again.
- Leads to memorizing and **contribute to overfitting**.



► CNNs Approach:

- **Aggressive data augmentation** to the minority class.
- This is the preferred technique for upsampling the minority class.
- Images are not duplicated, learn more robust features.
- **Not contributing to overfitting**.



Tackling Imbalance: Class Weights

▶ Source of problem:

- Loss function is “dominated” by the majority class.

▶ Mitigate:

- Increases the importance of the minority class, forcing the model to pay attention.
- Making errors in the minority class more painful for the model.
- Applying different weights for each class in the loss function.

▶ Advantages:

- No overhead in the training loop
- It does not contribute to the overfitting, does not duplicate images, learn more robust features

Tackling Imbalance: Loss Functions

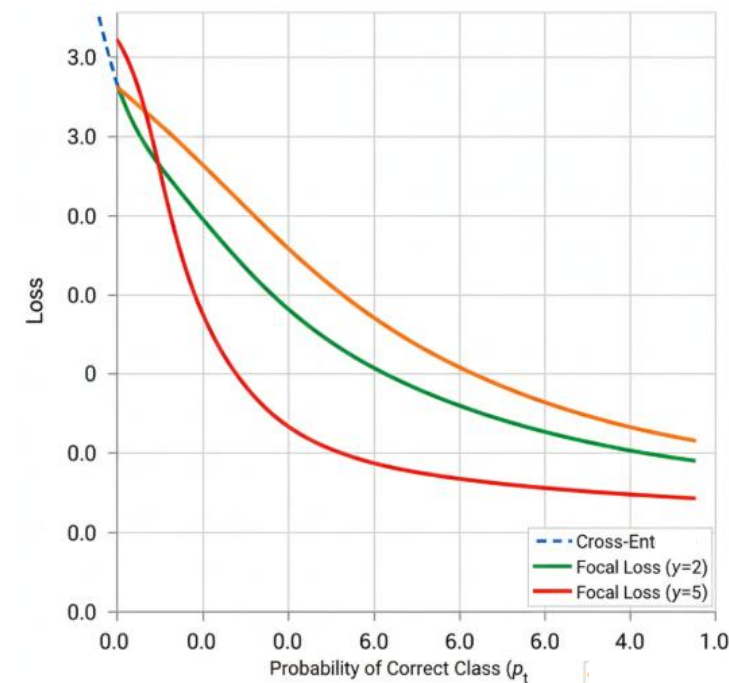
► Focal Loss Function:

- Loss functions specifically designed for class imbalance.
- Focal loss function: modification of cross-entropy.

► Idea:

- Dynamically reduces the influence of easy examples.
- Parameter γ controls how aggressively model down-weight easy examples
 - In an easy example, p_t is near 1: $(1-p_t)^\gamma$ is near zero.
 - In a hard example, p_t is near 0: $(1-p_t)^\gamma$ is near one.

$$\text{Focal Loss} = -(1 - p_t)^\gamma \log(p_t)$$



Individual Error Analysis

► Individual mistakes:

- Find your **worst failures** (based on loss)
- Visualize and **search for patterns**:
 - Is the model always failing on blurry images?
 - Images with unusual lightning?
 - Images where the object is partially obscured?
- Error analysis gives you actionable feedback.
 - **More diverse training** data is needed
 - Or a **better preprocessing** .
 - Or a **different model architecture**.



[credits](#)

► Model drift:

- Degradation of a model predictive **performance over time**.
- Why: Real **world is not static** (e.g. environmental conditions).
- Result:
 - A model trained on historical data becomes less accurate **as data drifts from patterns** it was trained on.
- Impact:
 - **Poor predictions**, unreliable insights and loss of trust in the NN.
- Solution: **continuously monitoring and regular retraining**.