Mastering Machine Learning

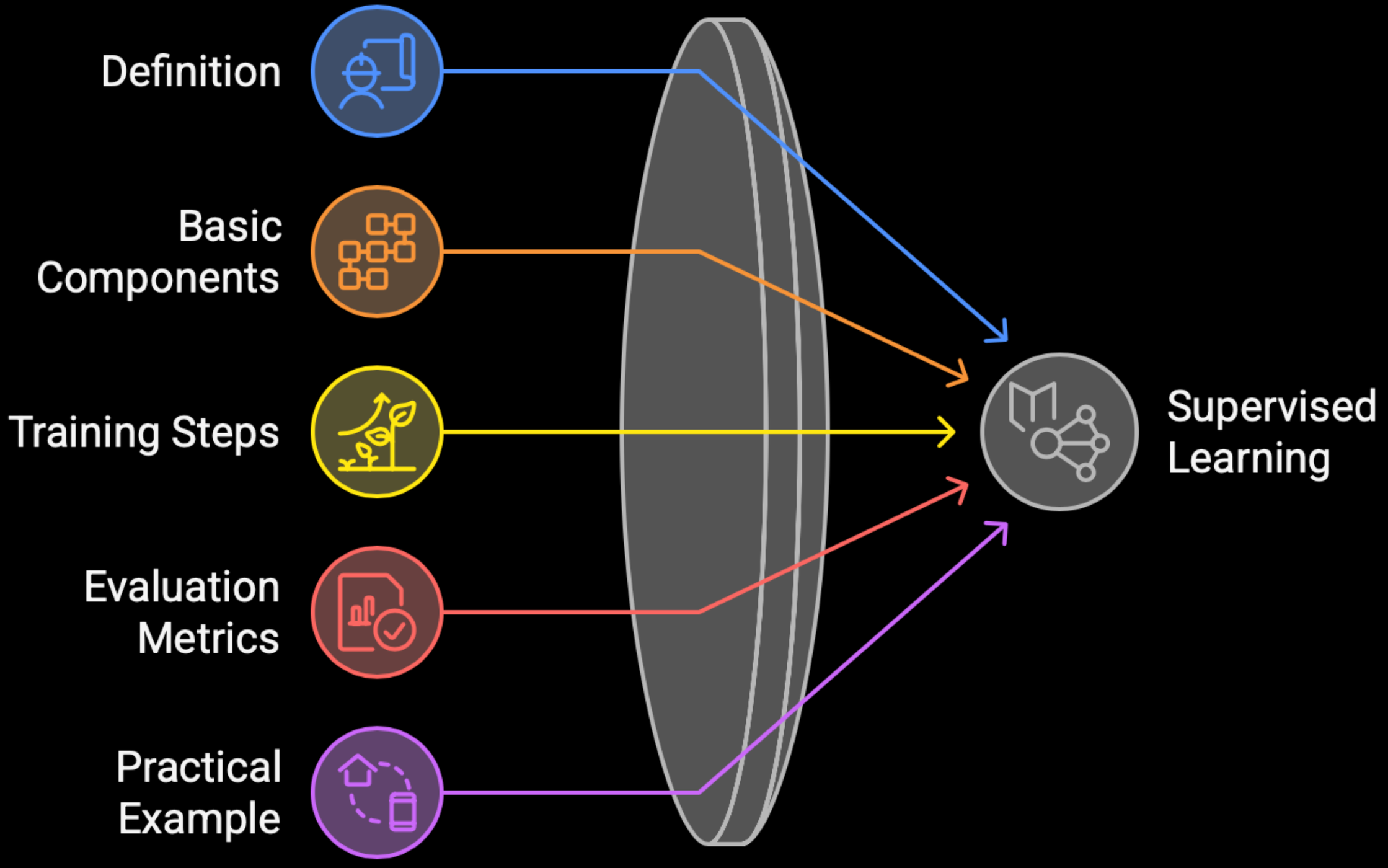
Model Training

A Complete Guide to Building Supervised Models

Edgar Rios Linares

Roadmap

Mastery of supervised learning



Definition

We "teach" a machine from labeled data so that it learns to predict future outcomes.

Definition

Supervised learning is a technique in which a model is trained with a set of labeled data.

Each training example is accompanied by a desired output.

The goal is for the model to learn to map inputs (features) to correct outputs (labels).

Components

Labeled dataset

Features

Labels

Objective function

Components

Labeled dataset: Input data with its respective expected outputs.

Features: Relevant attributes that will be used to make predictions (pixels of an image)

Labels: **Target** values that the model must learn to predict (dog or cat)

Objective Function: Metric that the model optimizes during training to improve its predictions.

Training Steps

Data Splitting

training
testing
validation

Model Selection

Regression
Trees
Random Forest
SVM

Parameter Optimization

Cross Validation
Grid Search

Performance Evaluation

Precision
Recall
F1-Score
Matrix Confusion

Training Steps

1. Data Splitting

   Training: Train the model. (70-80%)

   Validation: Tune hyperparameters. (10-15%)

   Testing: Evaluate final performance. (10-15%)

2. Model Selection

   Regression, trees, SVM, networks, etc.

3. Parameter Optimization

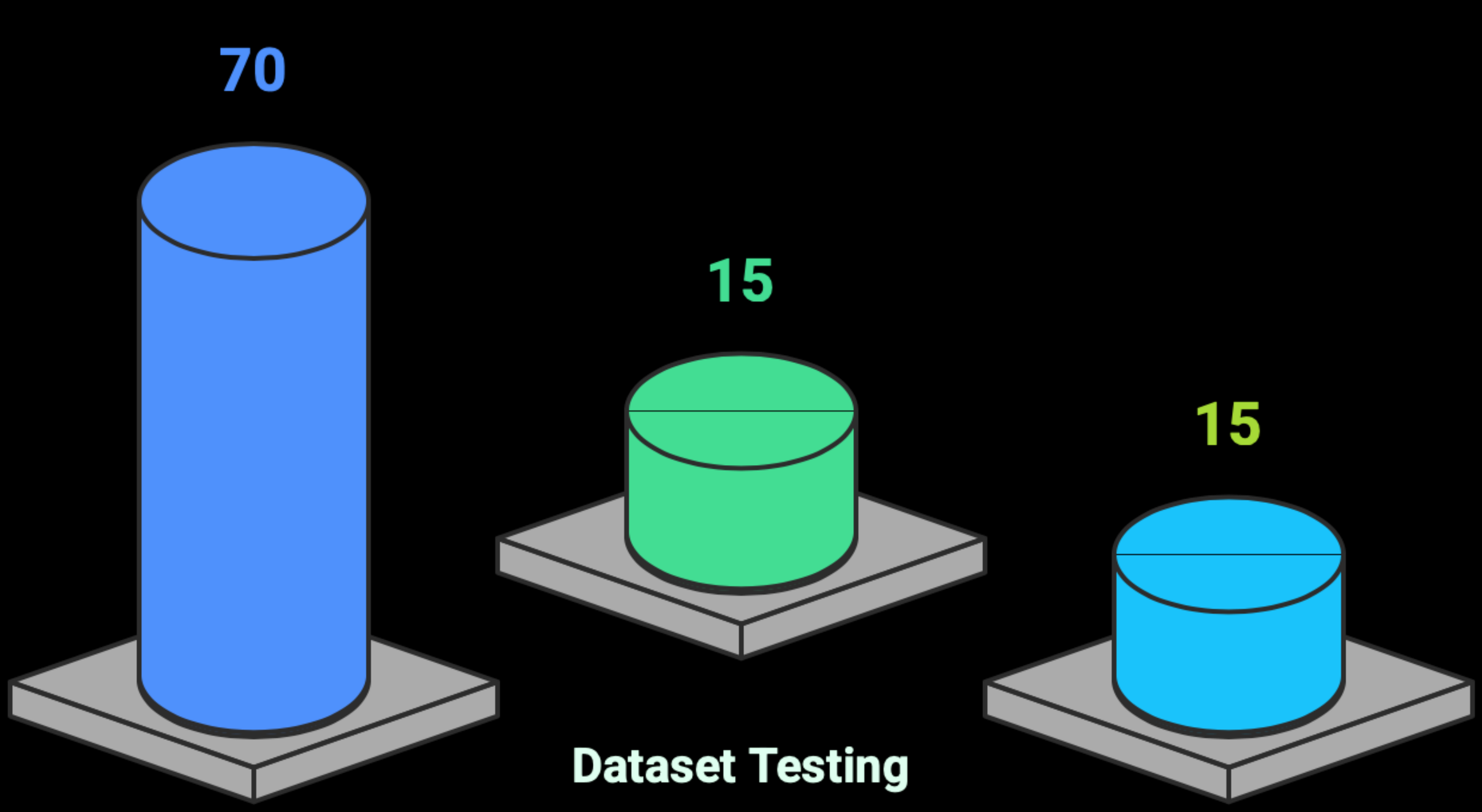
   Cross-validation, Grid Search, etc.

4. Performance Evaluation

   Validation and testing performance.

Training Steps

1. Data Splitting



Dataset Training

Used to train the model

Dataset Testing

Used to tune hyperparameters

Dataset Validation

Used to evaluate final performance

Training Steps

1. Data Splitting

Features (X) Label (y)

X1	X2	…	Xn	y
999	787	678	554	CAT
123	332	444	555	DOG
123	333	405	554	DOG
999	787	678	554	CAT
123	333	405	554	DOG
999	787	678	554	CAT
999	787	678	554	CAT
123	787	405	554	DOG
999	787	678	554	CAT
999	787	678	554	CAT
123	333	543	233	DOG

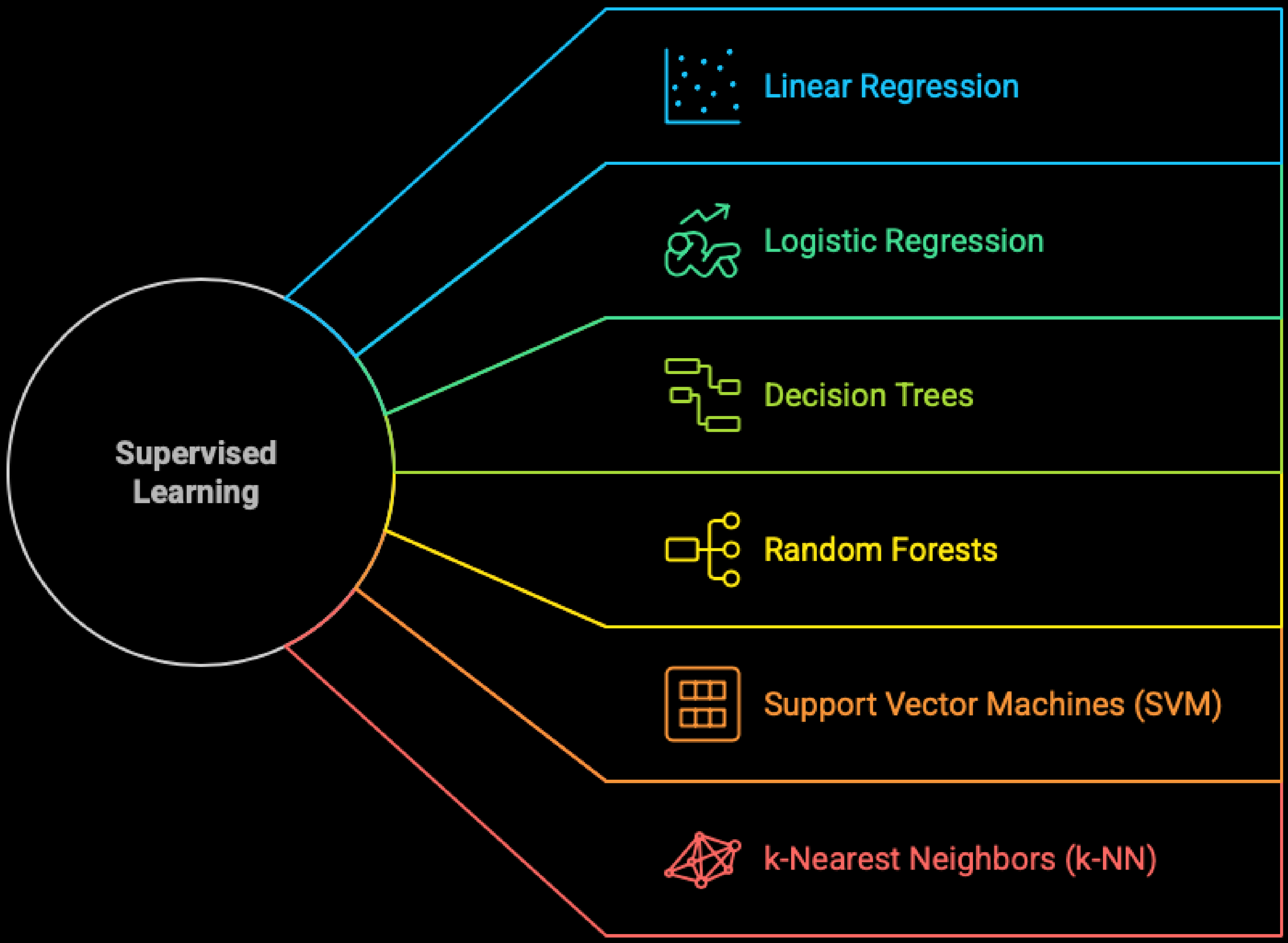
Training 70%

Test 15%

Validation 15%

Training Steps

2. Model Selection



Training Steps

3. Parameter Optimization

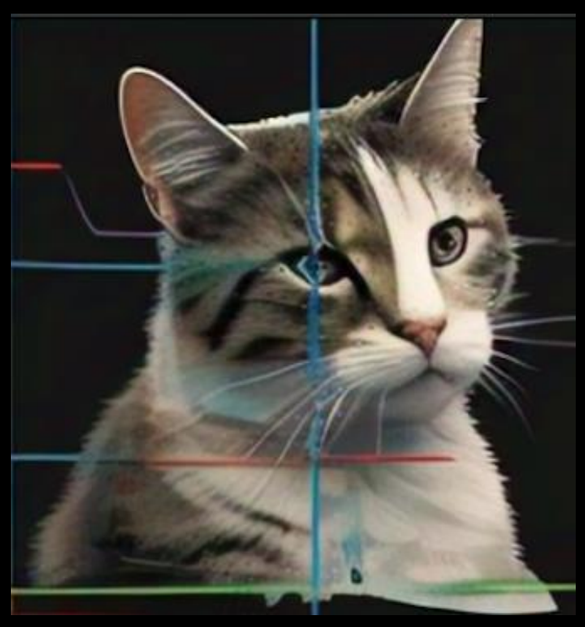
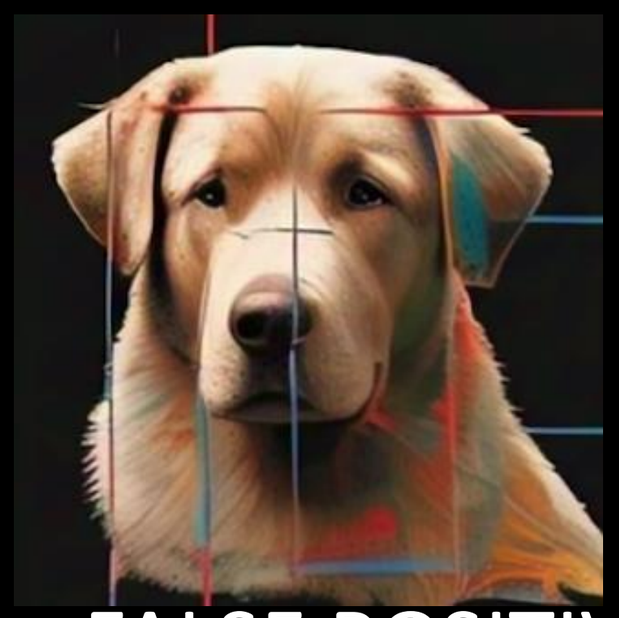
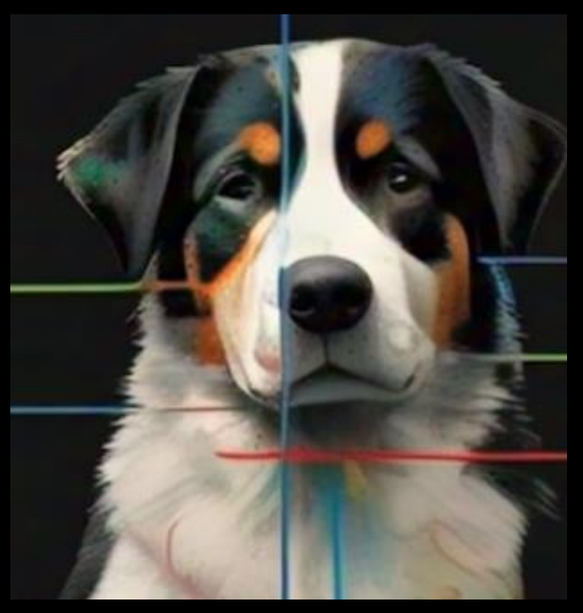
Features (X)				Label (y)	
X1	X2	…	Xn	y	
999	787	678	554	CAT	Training
123	332	444	555	DOG	
123	333	405	554	DOG	
999	787	678	554	CAT	
123	333	405	554	DOG	
999	787	678	554	CAT	
999	787	678	554	CAT	
123	787	405	554	DOG	
999	787	678	554	CAT	Test
999	787	678	554	CAT	
123	333	543	233	DOG	

K1

K2

Training Steps

4. Performance Evaluation

PREDICTION	ACTUAL: CAT	ACTUAL: NO CAT
CAT	 TRUE POSITIVE	 FALSE POSITIVE ERROR
NO CAT	 FALSE NEGATIVE ERROR	 TRUE NEGATIVE

Evaluation Metrics

F1-Score

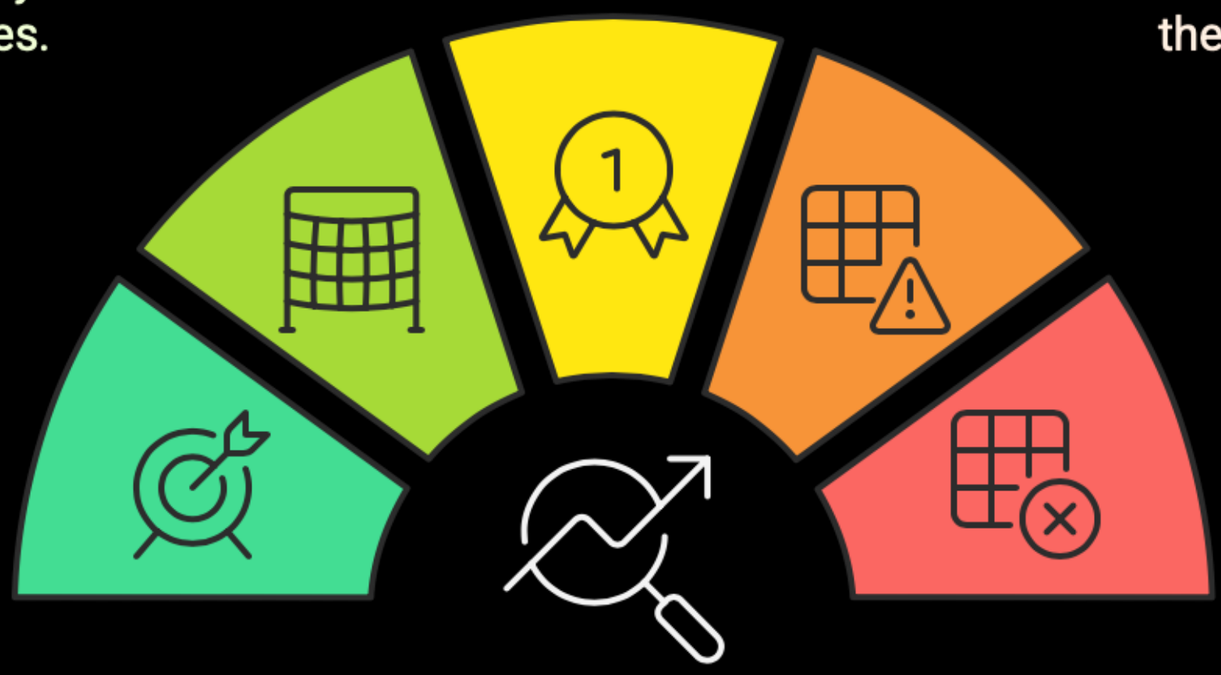
Balances precision and recall in a single score.

Recall

Evaluates the model's ability to identify all positive cases.

MSE

Calculates the average of the squared errors of the predictions.



Precision

Measures the accuracy of the model's positive predictions.

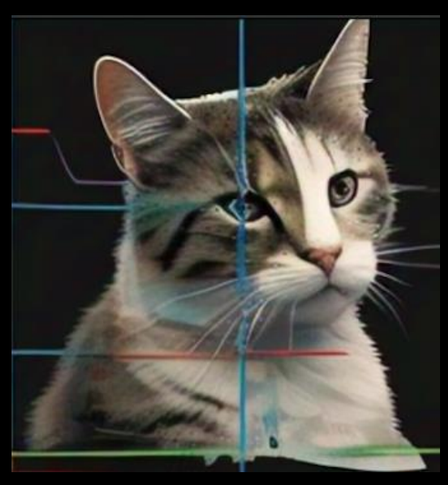
Confusion Matrix

Provides a detailed view of correct and incorrect predictions.

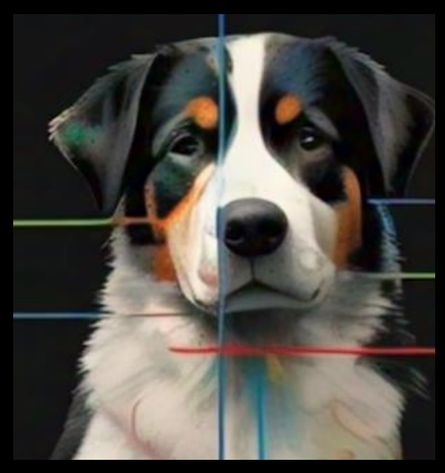
Evaluation Metrics

Precision (accuracy) Proportion of correct predictions out of the total number of predictions.

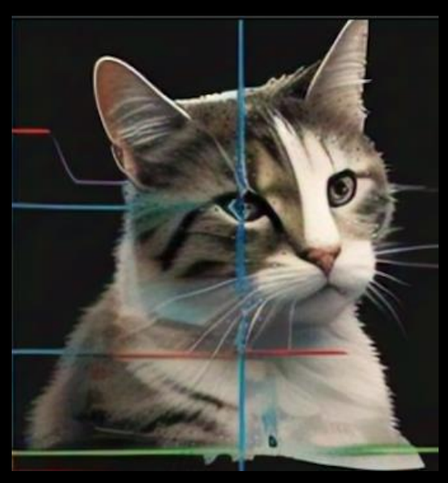
Formula: (True positives + True negatives) / Total.



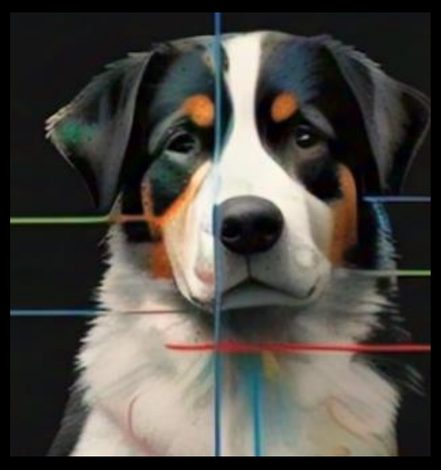
TRUE POSITIVE



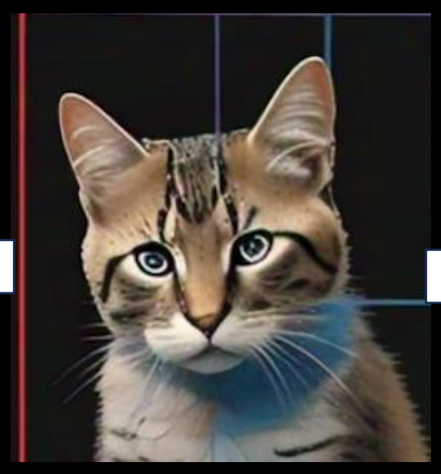
TRUE NEGATIVE



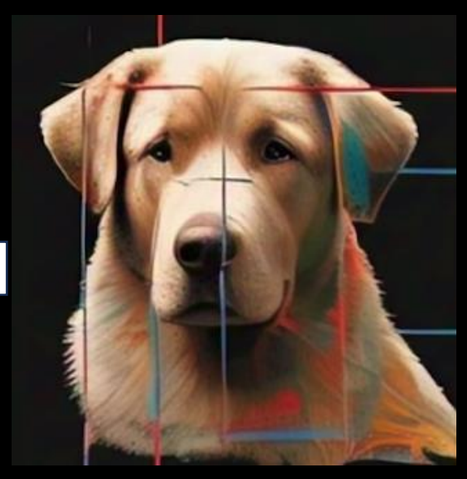
TRUE POSITIVE



TRUE NEGATIVE



FALSE NEGATIVE

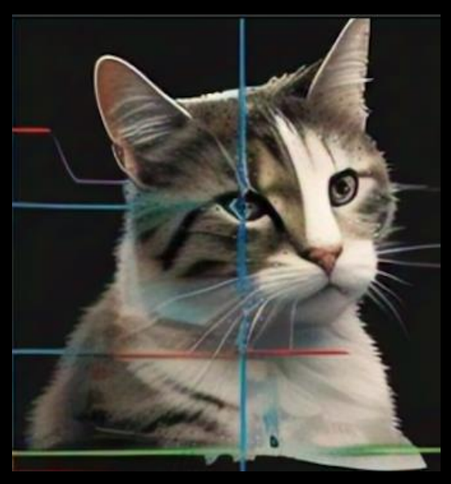


FALSE POSITIVE

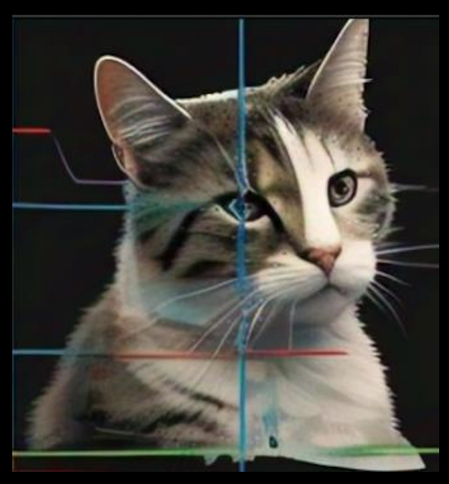
Evaluation Metrics

Recall (Sensitivity) Proportion of true positives out of the total number of real positives.

Formula: True positives / (True positives + False negatives).



TRUE POSITIVE



TRUE POSITIVE

FALSE NEGATIVE

Evaluation Metrics

F1-Score, Average of precision and recall. Useful when there is imbalance in classes.

Formula: 2 * (Precision * Recall) / (Precision + Recall).

$$F1\text{-}score = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$$

Evaluation Metrics

MSE (Mean Square Error): For regression problems, measures the difference between actual and predicted values

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Where :

y_i are the actual values.

$\hat{y}_i$ are the predictions of the model.

n is the total number of observations.

Practical Example

Predicting Solvent Companies

Step 1: Data Collection

A data set is obtained with characteristics such as income, expenses, credits and solvency.

Step 2: Preprocessing

Data cleaning and treatment of null values.

Normalization of numerical variables.

Step 3: Model Training

Data division into training, validation and testing.

LDA model - relationship of characteristics and solvency.

Step 4: Evaluation

Precision, Recall, F1-Score and Confusion Matrix are calculated.

Hyperparameters are adjusted and retraining is carried out.

Practical Example

STEP 1

	income	debt	credit_history	solvent
0	57450.712295	26483.242833	392.115420	0
1	47926.035482	33365.916483	357.512312	0
2	59715.328072	10210.026983	650.036637	1
3	72845.447846	23940.784657	688.561650	1
4	46487.699379	15445.502016	317.372380	1

STEP 2

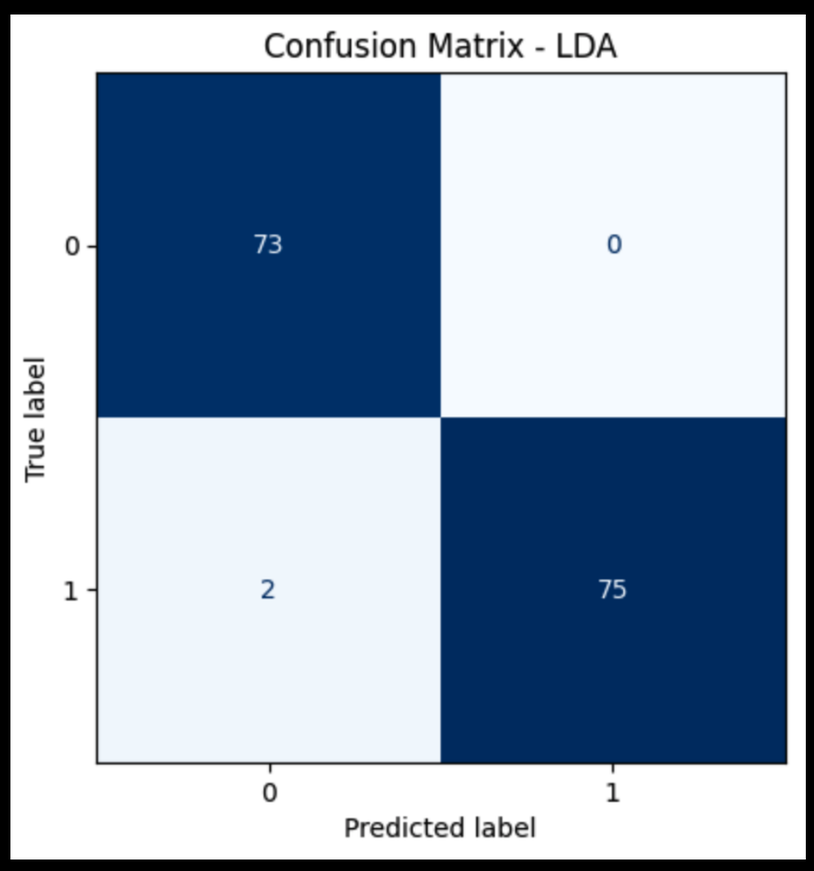
	income	debt	credit_history
0	-0.245825	-0.531158	1.500187
1	-0.042386	0.079393	-0.803370
2	-0.741307	-0.848214	-0.136724
3	-1.040191	0.480125	0.156964
4	-0.720783	0.168569	-1.356440

STEP 3

```
X_lda
```

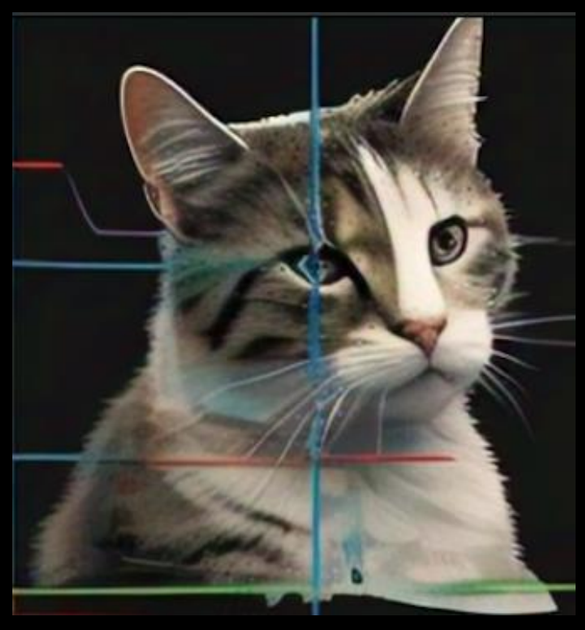
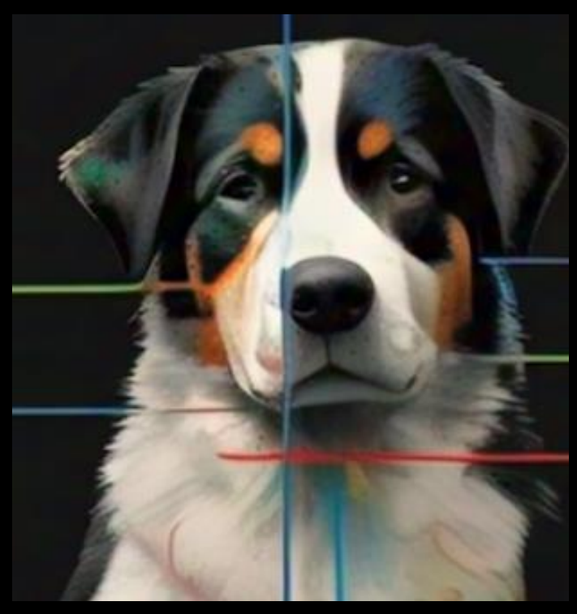
```
▼ LinearDiscriminantAnalysis
LinearDiscriminantAnalysis()
```

STEP 4



OVERFITTING

Overfitting occurs when a model learns the details and noise of the training set too well, but does not generalize well to new data. It memorizes the data.

PREDICTION	ACTUAL: CAT	ACTUAL: NO CAT
CAT	 TRUE POSITIVE	0
NO CAT	0	 TRUE NEGATIVE

Educator in AI

Artificial Intelligence

Data Engineering

Machine Learning

Data Science

📍Linkedin —> https://www.linkedin.com/in/erlinares/

👋 Follow us on 𝕏: https://x.com/erlinares

💻 GitHub: https://github.com/erlinares/365_AI_Journey/

💬 Discord: https://discord.gg/5fFM2zh8

Edgar Rios Linares