NOVA IMS INFORMATION MANAGEMENT SCHOOL

Model Selection and Assessment

MACHINE LEARNING| DSAA 2526

October 8, 2025

Carina Albuquerque
calbuquerque@novaims.unl.pt

Agenda

01 **Model Selection**

02 **Model Assessment**

01.

Model Selection

The process of choosing the best model from a set of candidate models (usually, the one that generalizes best to unseen data).

It involves comparing different algorithms or the same algorithm with distinct hyperparameters.

1. Model Selection

The No Free Lunch Theorem

> "If an algorithm performs better than random search on some class of problems then it must perform worse than random search on the remaining problems."

David Wolpert and William G. Macready

In: No Free Lunch Theorems for Optimization

https://ti.arc.nasa.gov/m/profile/dhw/papers/78.pdf

- No single machine learning algorithm is universally the best-performing algorithm for all problems!
- You need to understand the trade-offs

1. Model Selection

The concept of Overfitting

Models rely on training data to learn

If we allow too much complexity, the model will “memorize” the training data, instead of extracting useful relationships

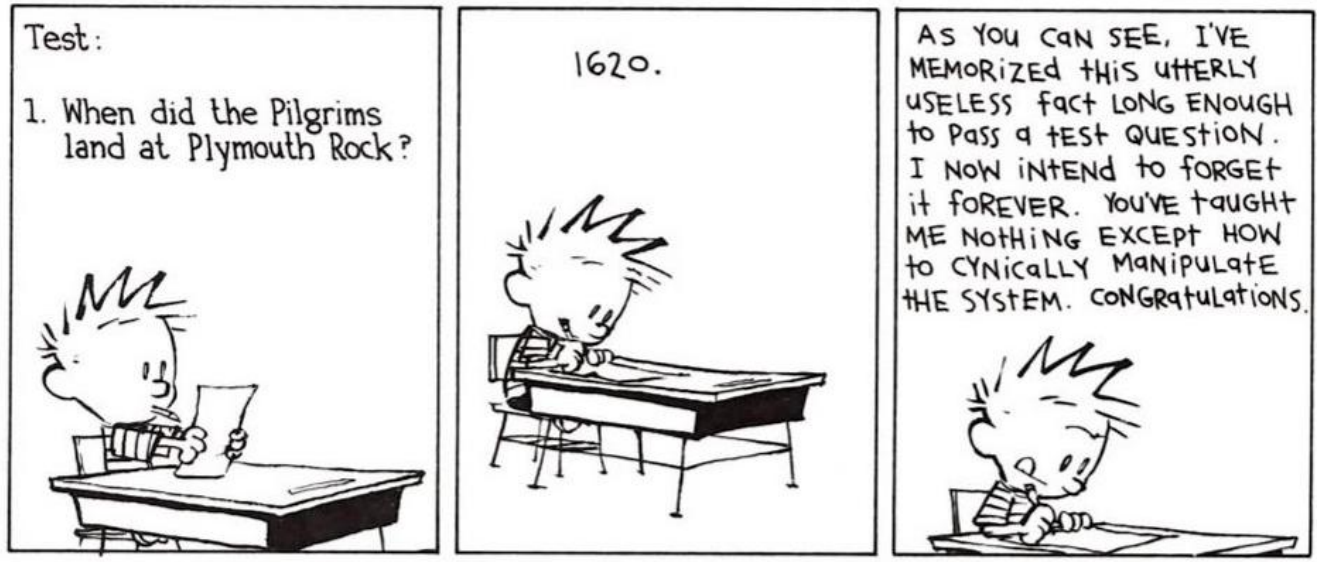
OVERFITTING

1. Model Selection

The concept of Overfitting

Memorizing vs Understanding

- Overfitting is like when someone memorizes things to pass an exam
  - He'll be too biased on the exercises he saw in classes
  - If he gets a slightly different question in the exam, he won't know how to answer



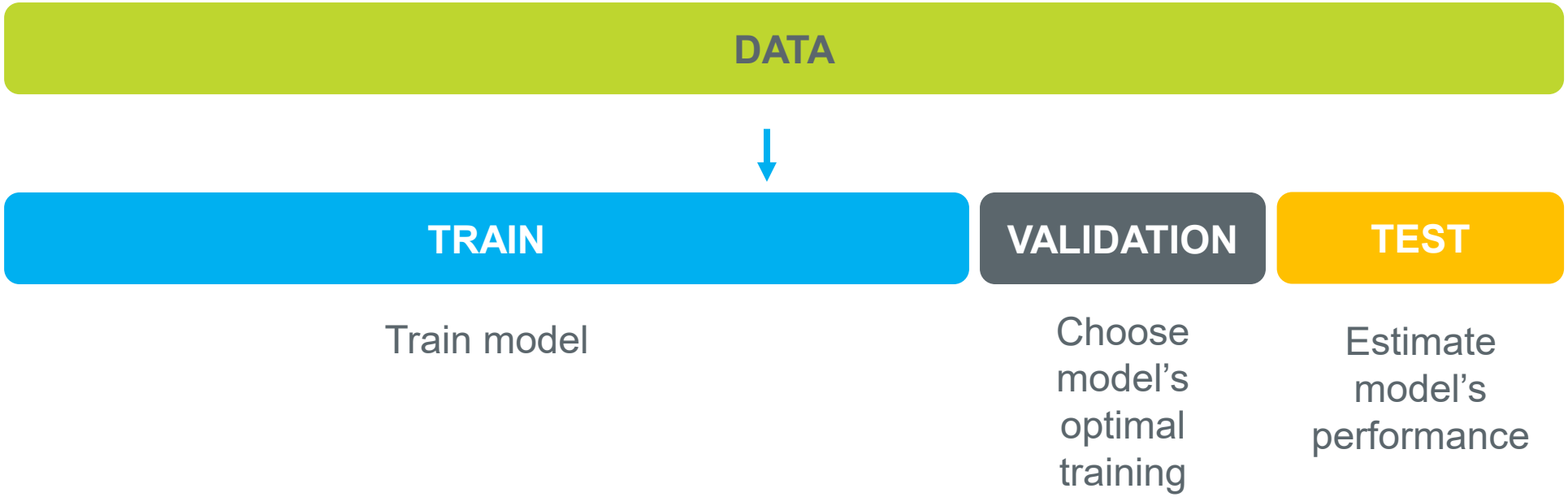
1. Model selection

The concept of overfitting

How to avoid overfitting?

How to prepare for the unknown?

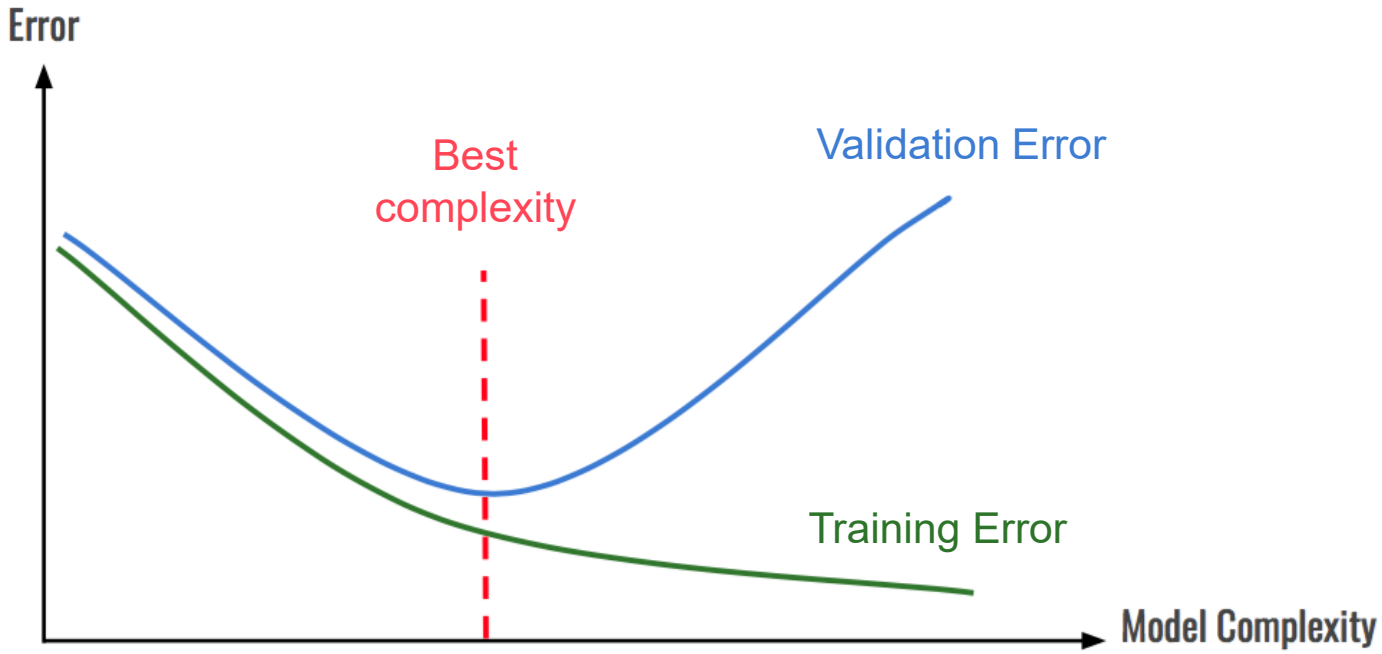
- Keep some data aside!



1. Model selection

The concept of overfitting

How to avoid overfitting?

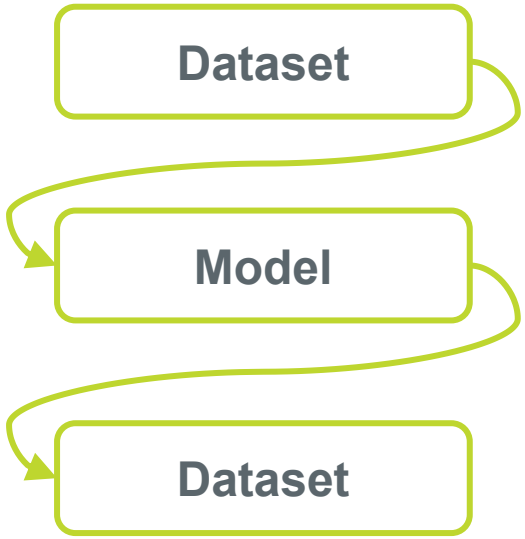


1. Model selection

How to split your data?

TRAINING SET	The bigger the better the classifier.
VALIDATION SET	The bigger the best the estimate of the optimal training.
TEST SET	The bigger the best the estimate of the performance of the classifier on unseen data.

1. Model selection



We can estimate and evaluate the performance of the model on the training data

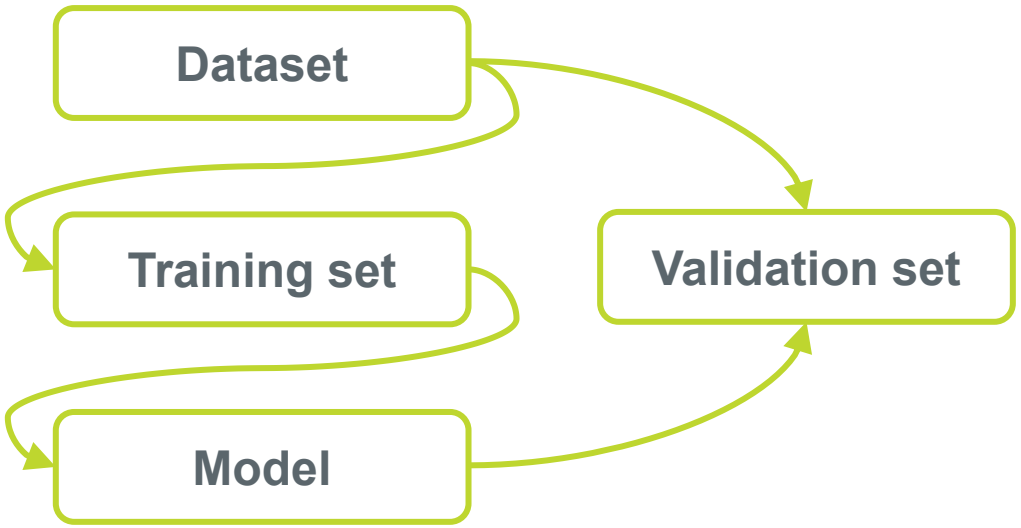
- However, estimates based only on the training data are not good indicators of the performance of the model on unseen data
- New data will probably not be exactly as the same as the training data – these estimates suffer from overfitting

This is not the ideal!

1.1. Model selection

Hold-out Method

One possible solution:



- Use an independent validation set
- Used when:
  - There is plenty of data

RULE OF THUMB

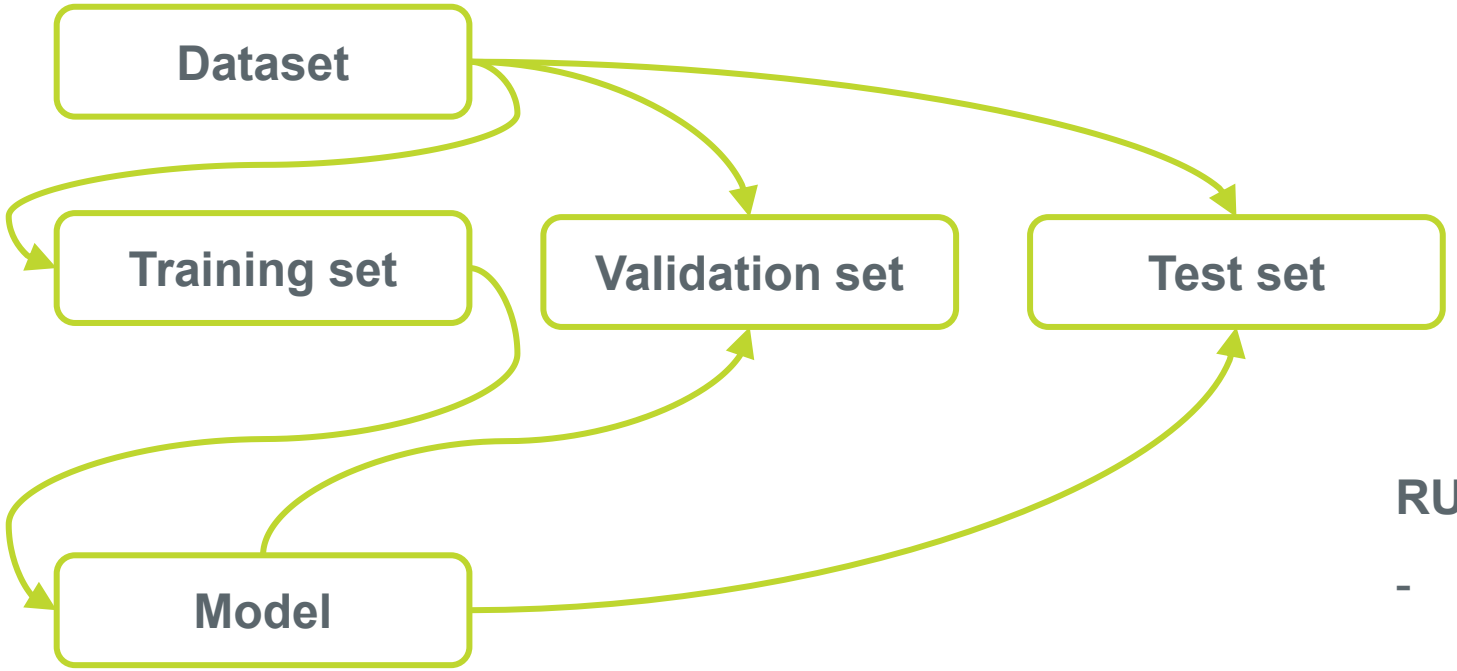
- 70 % for training set
- 30 % for validation set

(Some authors defend 80% train and 20% validation – it depends on the size of the dataset)

1.1. Model selection

Hold-out Method

When there is plenty of data we can split our dataset into training, validation and test set:



RULE OF THUMB

- 70 % for training set
- 15 % for validation set
- 15 % for test set

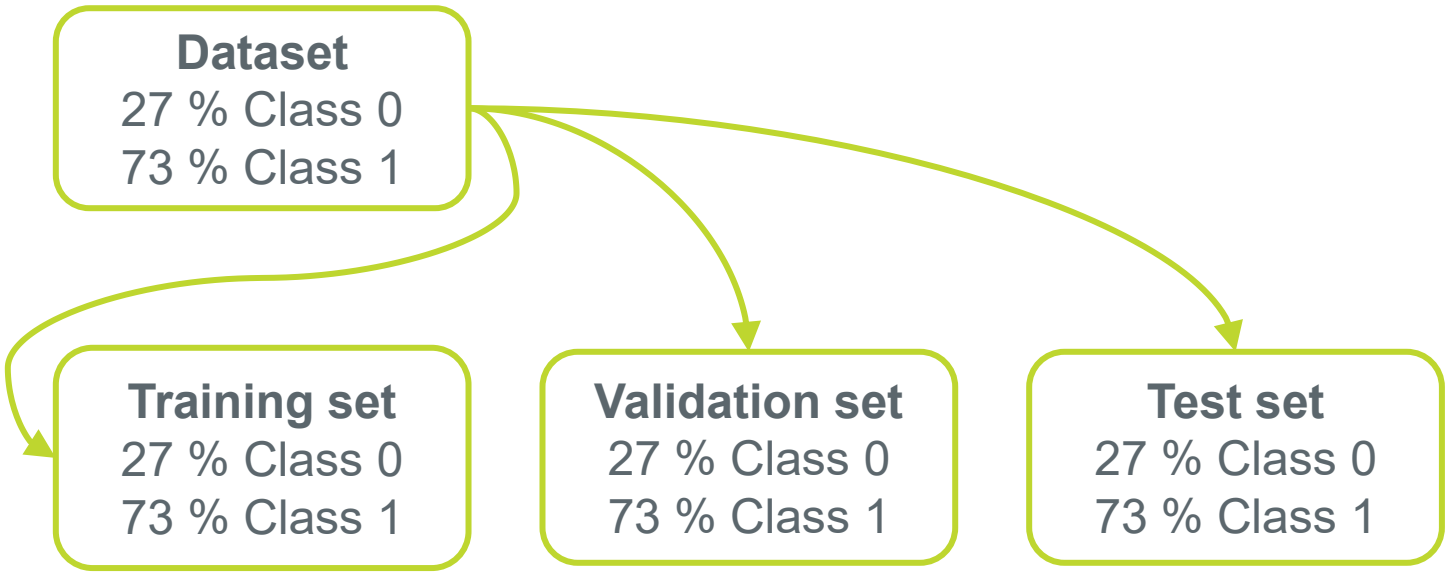
1.1. Model selection

Question: What if your dataset is imbalanced?

Answer: Then, for some runs, certain classes may be not (well) represented. We need to assure that the data partition will not change the original proportions of each class on the new datasets.

Stratified sampling

Sample each class independently such that each dataset has the same ratio of any given class



1.1. Model selection

Hold-out Method

PROBLEMS OF THE HOLD-OUT METHOD

- It only provides a single estimate
  - **SOLUTION** : Use a repeated hold-out method
- It requires a large enough dataset
  - **SOLUTION** : Use K-Fold cross-validation

Repeated hold-out method

1. Run the holdout method many times
2. Each run will have a different train and validation set
3. Each run will lead to slightly different metric values

↓

Average value allows a more robust estimate
Compute standard deviation to estimate variability

Still not optimum
The different tests sets overlap, but we would like all our instances from the data to be tested at least once
Can we prevent overlapping?

1.1. Model selection

Hold-out Method

PROBLEMS OF THE HOLD-OUT METHOD

- It only provides a single estimate
    - **SOLUTION** : Use a repeated hold-out method
- It requires a large enough dataset
    - **SOLUTION** : Use K-Fold cross-validation

1.2. Model selection

K-Fold Cross Validation

Step 1

Split data into k subsets of equal sizes

Step 2

Each turn, use one subset for testing / validation and the remainder for training

Step 3

Average all estimates to get a final value

The common pipeline…

After tuning the model using cross-validation, you evaluate the model's performance on a completely separate test set that was never used during cross-validation

- This ensures that the final performance metric reflects how the model generalizes to unseen data.

1.2. Model selection

K-Fold Cross Validation

SOME TIPS

- Use a **stratified sample** (it reduces the variance)
- **10 folds is by far the most used**
  - Extensive experiments have shown that this is the best choice to get an accurate estimate
  - Smaller datasets can demand a small number of folds
  - Big datasets can make use of more folds (and in that way get a more robust estimate)

1.3. Model selection

Other options

- Use a **repeated stratified cross-validation** to further reduce variance
- **Leave–One–Out Cross-Validation:**
  - CV with as many folds as instances
  - Very expensive computationally
  - Makes best use of the data

THE BEST METHOD DEPENDS ON EACH CASE !

1.4. Model selection

Compare two models

Step 1 Choose your **metric**

Step 2 Choose an **evaluation method**

Step 3 **Run** the evaluation for both models and collect metrics

Step 4 **Compare** metrics

Step 5 **Pick** the model with the best performance

How to be sure one model is always better than the other?

Using different datasets will lead to slightly different results

The train and validation on different splits and the computation of standard deviation will help you

1.4. Model selection

Compare two models

What is the best model?

- The **simplest** representation of knowledge
  - Easier to understand
  - Decision Trees vs Neural Networks
- The knowledge representation with **less error**
- The most simple…
  - **Occam's Razor** – " the simplest explanation is usually the right one"

1. Model selection

Wrap-up

- Use **test sets** and the hold-out method for “large” data
- Use the **cross-validation** method for middle-sized data
- Use the **leave-one-out** method for small data
- Take advantage of **stratified sampling**
- **Don’t use test data for parameter tuning** – use separate validation data

02.

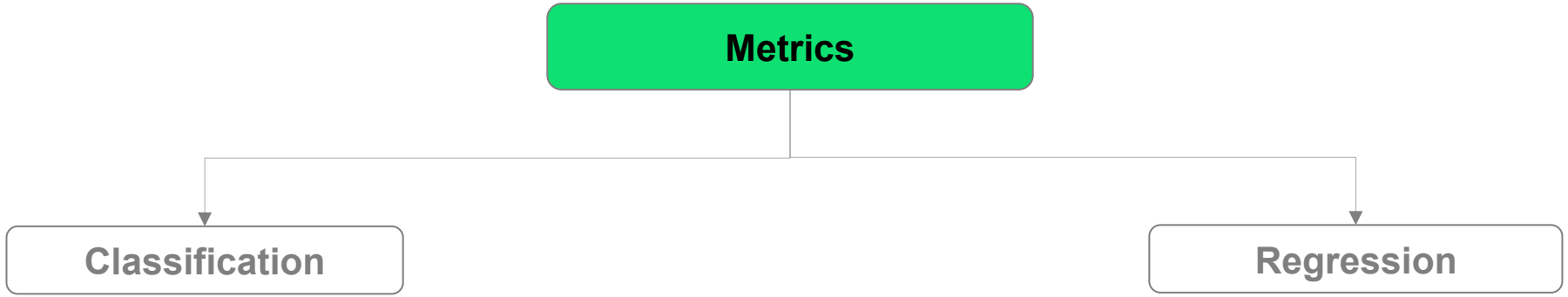
Model Assessment

The process to evaluate the performance of the model (after selection) on unseen data.

2. Model Assessment

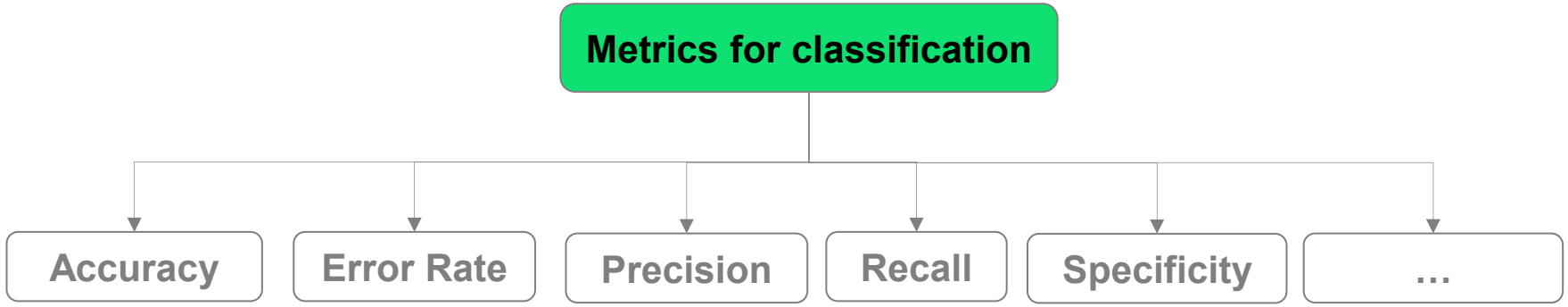
What are evaluation metrics?

An evaluation metric quantifies the performance of a predictive model



2.1. Model Assessment

Metrics for classification



2.1. Model Assessment

Metrics for classification

It all starts with …

		True Class	
		Positive	**Negative**
Predicted Class	**Positive**	TP	FP
	Negative	FN	TN

Type I Error

Type II Error

In classification, predictions are either correct or wrong.

We can encode this in a confusion matrix.

2.1. Model Assessment

Metrics for classification

Predicted Class: Goat
True Class: Goat

TRUE POSITIVE

		True Class	
		Goat	Not Goat
Predicted Class	Goat	1	
	Not Goat		

2.1. Model Assessment

Metrics for classification

Predicted Class: Goat
True Class: Not Goat

FALSE POSITIVE

		True Class	
		Goat	Not Goat
Predicted Class	Goat	1	1
	Not Goat		

2.1. Model Assessment

Metrics for classification

Predicted Class: Not Goat
True Class: Goat

FALSE NEGATIVE

		True Class	
		Goat	Not Goat
Predicted Class	Goat	1	1
	Not Goat	1	

2.1. Model Assessment

Metrics for classification

Predicted Class: Not Goat
True Class: Not Goat

TRUE NEGATIVE

		True Class	
		Goat	Not Goat
Predicted Class	Goat	1	1
	Not Goat	1	1

2.1. Model Assessment

Metrics for classification

		True Class	
		Goat	Not Goat
Predicted Class	Goat	30	4
	Not Goat	6	20

2.1.1. Model Assessment

Metrics for classification - Accuracy

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

		True Class	
		Goat	Not Goat
Predicted Class	Goat	30	4
	Not Goat	6	20

Accuracy

Proportion of events correctly identified (positive or negative) on all events

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

Overall, how often is the classifier correct?

In our example…

$$Accuracy = \frac{(30 + 20)}{(30 + 20 + 4 + 6)} = \frac{50}{60} = 0.83$$

Seems good, right?

Well… Not always!

(check slides 39 and 40)

2.1.2. Model Assessment

Metrics for classification – Error Rate

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

		True Class	
		Goat	Not Goat
Predicted Class	Goat	30	4
	Not Goat	6	20

Error Rate / Misclassification rate

Proportion of events incorrectly identified (positive or negative) on all events

$$Error\ Rate = \frac{(FP + FN)}{(TP + TN + FP + FN)}$$

Overall, how often is the classifier wrong?

In our example…

$$Error\ rate = \frac{(4 + 6)}{(30 + 20 + 4 + 6)} = \frac{10}{60} = 0.167$$

The lower the better!

2.1.3. Model Assessment

Metrics for classification – Precision

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Precision

Proportion of correctly positive events from all events identified as positive

$$Precision = \frac{TP}{(TP + FP)}$$

When it predicts as positive, how often is the classifier correct?

In our example…

		True Class	
		Goat	Not Goat
Predicted Class	Goat	30	4
	Not Goat	6	20

$$Precision = \frac{30}{(30 + 4)} = \frac{30}{34} = 0.882$$

- *Email Spam detection: A false positive means that an e-mail that is not spam has been identified as spam. The user might lose important emails if the precision is not high.*

2.1.4. Model Assessment

Metrics for classification – Recall

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Recall / Sensitivity / True Positive Rate (TPR)

Proportion of events identified as positive on all positive events

$$Recall = \frac{TP}{(TP + FN)}$$

How well the positive class was predicted? Or…

When it's actually positive, how often does the classifier predict positive?

		True Class	
		Goat	Not Goat
Predicted Class	Goat	30	4
	Not Goat	6	20

In our example…

$$Recall = \frac{30}{(30 + 6)} = \frac{30}{36} = 0.833$$

- *Sick patient detection: If a sick patient (actual positive) goes through the test and predicted as not sick (false negative), the cost will be extremely high if the sickness is contagious*

2.1.5. Model Assessment

Metrics for classification – Specificity

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

		True Class	
		Goat	Not Goat
Predicted Class	Goat	30	4
	Not Goat	6	20

Specificity / True Negative Rate (TNR)

Proportion of events identified as negative on all negative events

$$Specificity = \frac{TN}{(FP + TN)}$$

How well the negative class was predicted? Or….

When it's actually negative, how often does the classifier predict negative?

In our example…

$$Specificity = \frac{20}{(4 + 20)} = \frac{20}{24} = 0.833$$

2.1.6. Model Assessment

Metrics for classification – The problem of imbalanced datasets

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Example 2

- We have a dataset with 10000 images. Only 10 have goats.
  - What is the accuracy of the model?

$$Accuracy = \frac{(1 + 9990)}{(1 + 9990 + 0 + 9)} = \frac{9991}{10000} = 0.9991$$

		True Class	
		Goat	Not Goat
Predicted Class	Goat	1	0
	Not Goat	9	9990

This seems like an almost perfect model!

But in reality, the model is not able to identify well goats, which is our main goal!

The recall is equal to 0.1!

The problem of imbalanced datasets

2.1.6. Model Assessment

Metrics for classification – The problem of imbalanced datasets

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

		True Class	
		Goat	Not Goat
Predicted Class	Goat	1	0
	Not Goat	9	9990

Metrics for imbalanced datasets

$$Balanced\ Accuracy = \frac{Sensitivity + Specificity}{2} = \frac{\frac{TP}{(TP+FN)} + \frac{TN}{(FP+TN)}}{2} = \frac{\frac{1}{10} + \frac{9990}{9990}}{2} = 0.55$$

$$F1\ Score = 2 \times \frac{precision \times recall}{precision + recall} = 2 \times \frac{1 \times 0.1}{1 + 0.1} = 0.18$$

2.1. Model Assessment

Metrics for classification

- **Precision** is a good measure **when the cost of False Positive is high**
- **Recall** is a good measure **when the cost of False Negative is high**
- **F1 Score** is a good measure if you seek a **balance between precision and recall**
- **Accuracy** is a good measure only if the cost of False Positive and False Negative is equal and the dataset is balanced

2.1. Model Assessment

Metrics for classification – Cutoff

Most predictive algorithms classify via a 2-step process. For each record:

- Compute probability of belonging to class “1”
- Compare to cutoff value, and classify accordingly

- Default cutoff value is 0.5
  - If >= 0.5, then classify as “1”
  - If <0.5, classify as “0”
- We can use different cutoff values (typically, error rate is lowest for cutoff = 0.5)

2.1.7. Model Assessment

Metrics for classification – ROC Curve

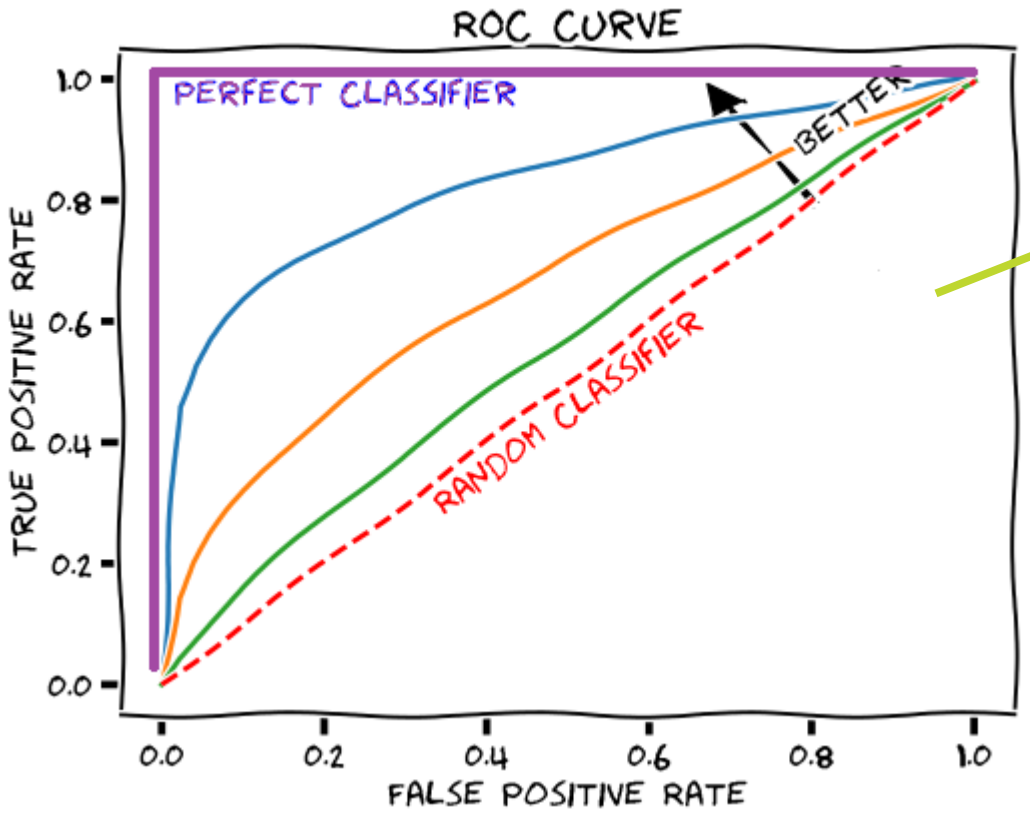


Image Source

Area under the ROC Curve ("AUC") is a useful metric

RULE OF THUMB

AUC values	Test quality
0.9–1.0	Excellent
0.8–0.9	Very good
0.7–0.8	Good
0.6–0.7	Satisfactory
0.5–0.6	Unsatisfactory

2.1.8. Model Assessment

Metrics for classification – Precision-Recall Curve

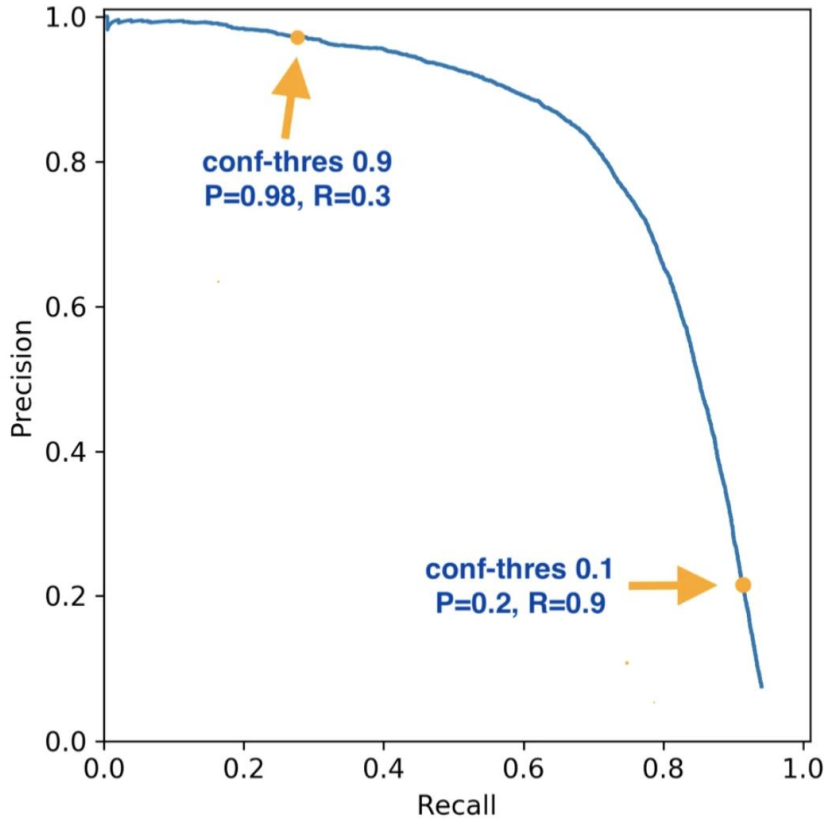


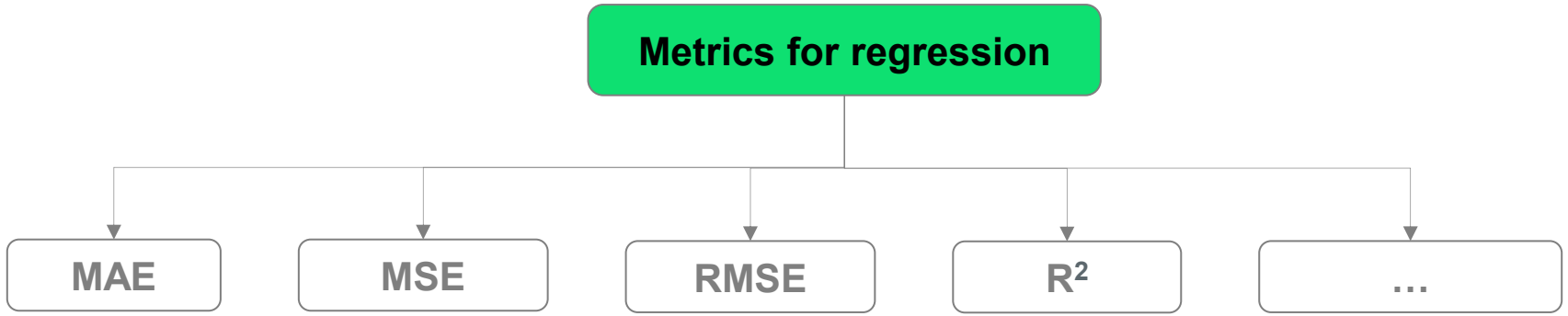
Image Source

Find the best F1 Score:

1. Generate the Precision-Recall Curve
2. At each threshold, calculate the F1 score using the corresponding precision and recall values.
3. Find the threshold that corresponds to the maximum F1 score.
4. Use the threshold for predictions

2.2. Model Assessment

Metrics for regression



2.2. Model Assessment

Metrics for regression

What about when label is continuous?

We can no longer compute the confusion matrix…

Measure how far the prediction ($\hat{y}$) is from the true value (y)

2.2. Model Assessment

Metrics for regression

Model 1

y	$\hat{y}$	$\lvert y - \hat{y} \rvert$
3	4	1
6	5	1
4	6	2
8	7	1
12	11	1
5	5	0

Model 2

y	$\hat{y}$	$\lvert y - \hat{y} \rvert$
3	4	1
6	5	1
4	6	2
8	7	1
12	11	1
5	14	9

2.2.1. Model Assessment

Metrics for regression – Mean Absolute Error (MAE)

Model 1

y	$\hat{y}$	$\lvert y - \hat{y} \rvert$
3	4	1
6	5	1
4	6	2
8	7	1
12	11	1
5	5	0

Model 2

y	$\hat{y}$	$\lvert y - \hat{y} \rvert$
3	4	1
6	5	1
4	6	2
8	7	1
12	11	1
5	14	9

$$MAE\ (y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

Model 1

$$MAE = \frac{1 + 1 + 2 + 1 + 1 + 0}{6} = 1$$

Model 2

$$MAE = \frac{1 + 1 + 2 + 1 + 1 + 9}{6} = 2.5$$

This measures the absolute average distance between the real data and the predicted data, but it fails to punish large errors in prediction.

2.2.2. Model Assessment

Metrics for regression – Mean Squared Error (MSE)

Model 1

y	$\hat{y}$	$\lvert y - \hat{y} \rvert$
3	4	1
6	5	1
4	6	2
8	7	1
12	11	1
5	5	0

Model 2

y	$\hat{y}$	$\lvert y - \hat{y} \rvert$
3	4	1
6	5	1
4	6	2
8	7	1
12	11	1
5	14	9

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

Model 1

$$MSE = \frac{1 + 1 + 4 + 1 + 1 + 0}{6} = 1.33$$

Model 2

$$MSE = \frac{1 + 1 + 4 + 1 + 1 + 81}{6} = 14.83$$

This measures the squared average distance between the real data and the predicted data. Here, larger errors are well noted (better than MAE).

2.2.3. Model Assessment

Metrics for regression – R-Squared / Coefficient of Determination

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

Model 1

			$\bar{y}$ = 6.33
y	$\hat{y}$	$(y - \hat{y})^2$	$(y - \bar{y})^2$
3	4	1	11.09
6	5	1	0.11
4	6	4	5.43
8	7	1	2.79
12	11	1	32.15
5	5	0	1.77
SUM		8	53.34

Model 1

$$R^2 = 1 - \frac{8}{53.34} = 0.85$$

Interpretation: 85 % of the variance in the dependent variable is explained by the model

2.2.4. Model Assessment

Metrics for regression – Adjusted R-Squared

When comparing models….

- As the number of independent variables increase, the value of R-Square will never decrease
  - Reason: Any independent variable has tendency of slightly correlation with the dependent variable
- To compare models with different number of independent variables, we can use the adjusted R-Square:

$$Adjusted\ R^2\ (y, \hat{y}) = 1\ -\ \frac{(1\ -\ R^2)(N\ -1)}{N\ -p\ -1}$$

R Squared

Sample Size (number of observations)

Number of predictors (independent variables)

2.2. Model Assessment

Metrics for regression – Wrap Up

- **MSE / RMSE** has the benefit of penalizing large errors more so can be more appropriate in some cases, for example, if being off by 10 is more than twice as bad as being off by 5.
- If being off by 10 is just twice as bad as being off by 5, then use **MAE**
- Use **adjusted R-Squared** when comparing the performance of models where the number of predictors is different

NOVA IMS INFORMATION MANAGEMENT SCHOOL

Obrigada!

Thank you!

Acreditações e Certificações da NOVA IMS

eduniversal · QS STARS · ABET Computing Accreditation Commission · iSchools · EMOS · UNIGIS · USGIF · Erasmus Mundus Programme · IQNET

Cofinanciado por

PORTUGAL 2030 · 2030 Lisboa · Co-funded by the European Union · fct Fundação para a Ciência e a Tecnologia · PRR · REPÚBLICA PORTUGUESA · Financiado pela União Europeia NextGenerationEU