



T. JACQUIOT

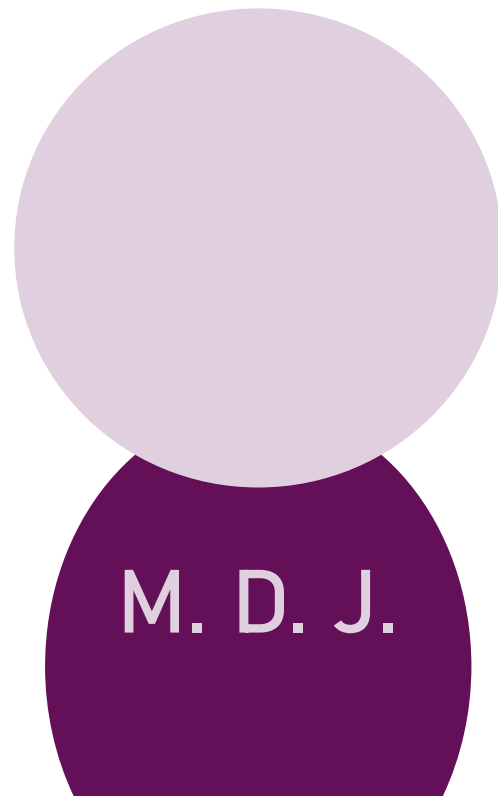
ENCADRÉE PAR J. REYNAUD & B. CRÉMILLEUX

SOUTENANCE DE STAGE

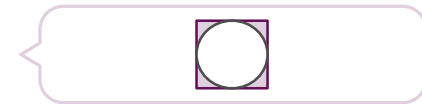
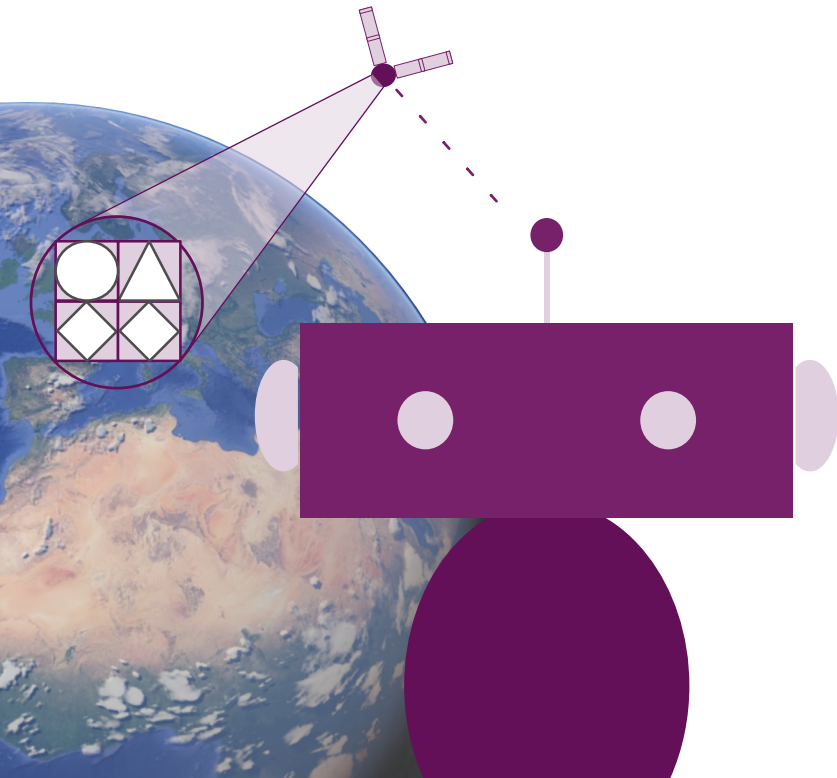
03
Sept.
2021

Bonjour à tous,

Aujourd'hui, je vais vous présenter le fruit de mon stage de 6 mois au sein du projet ANR Herelles.



Dans Herelles, notre expert, c'est Monsieur D. J., un géographe qui travaille dans l'aménagement de territoire.



EXPERTISE

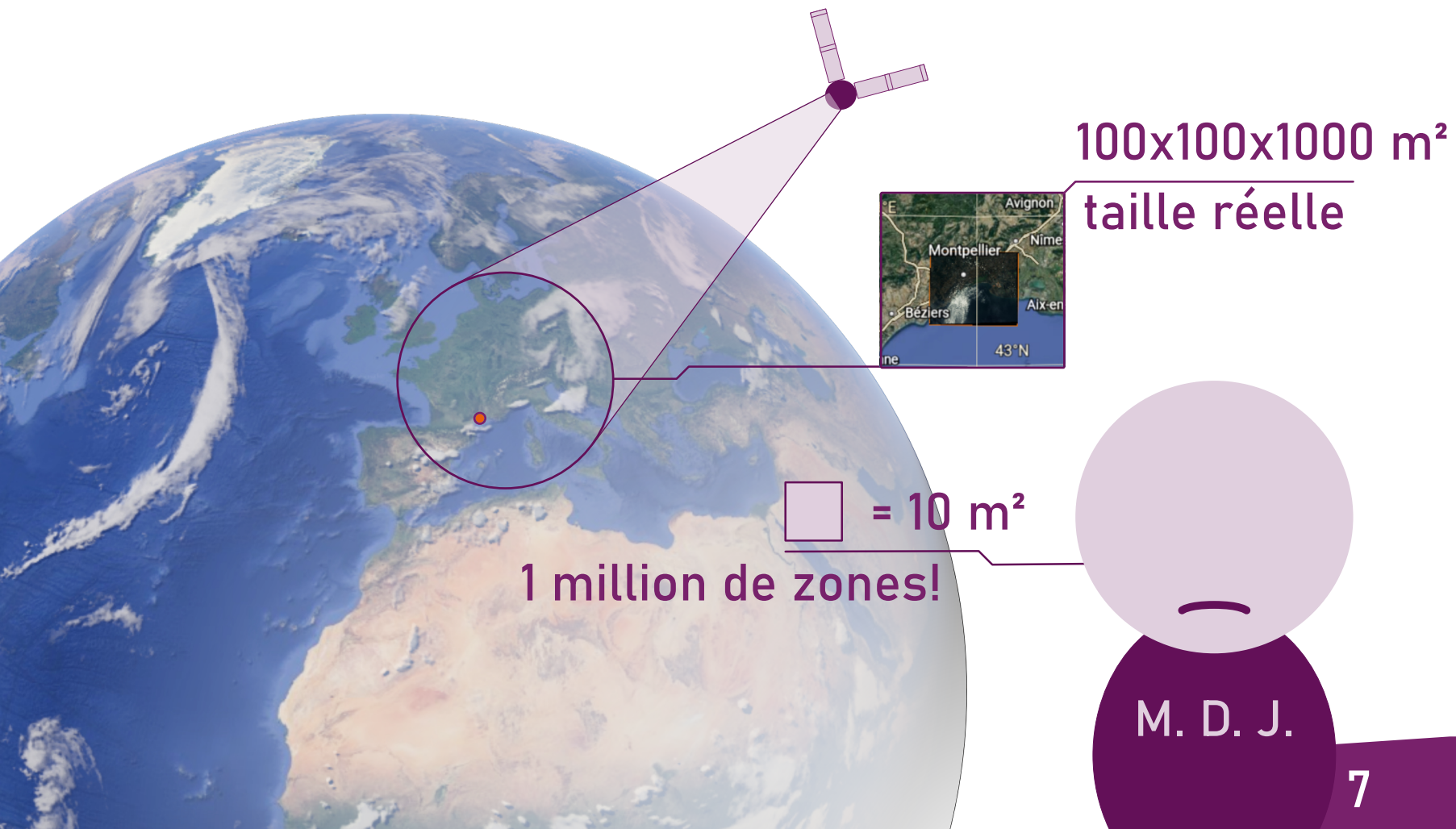


EXPERTISE

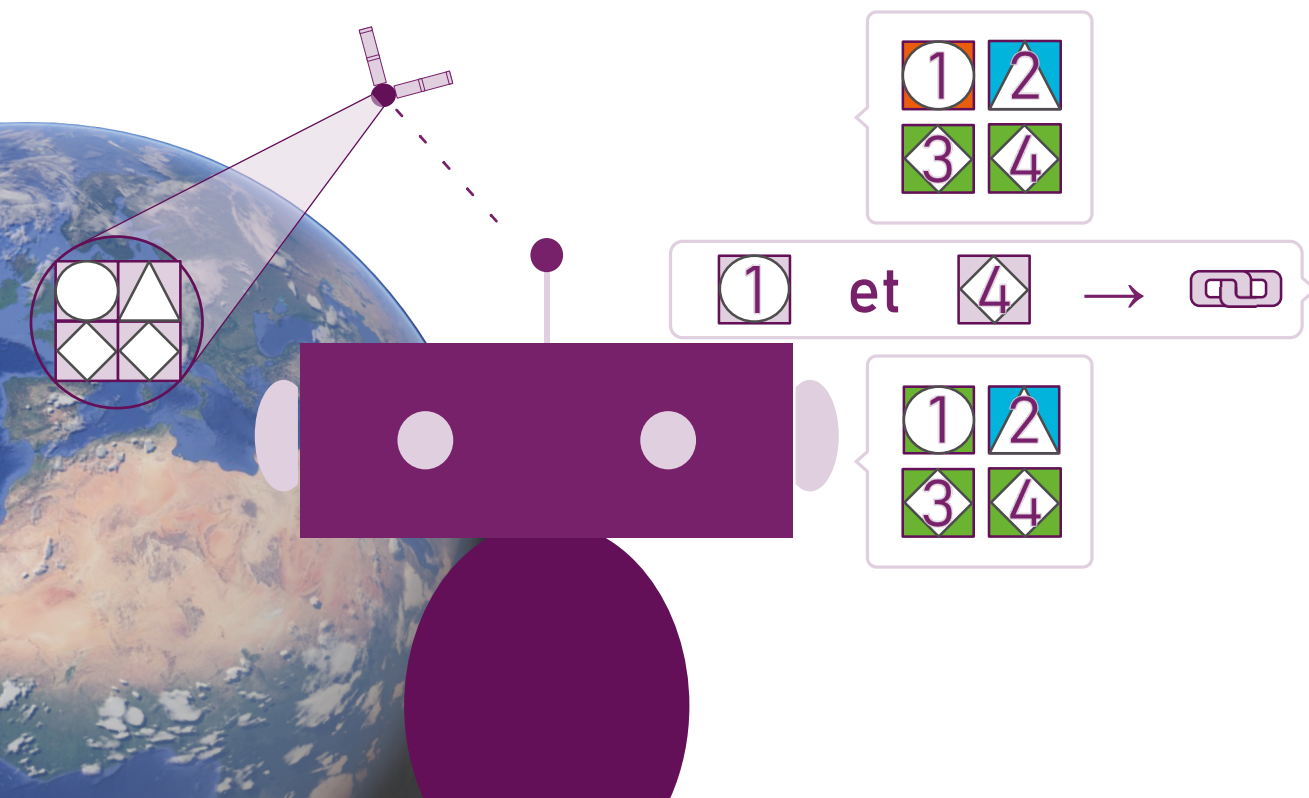


M. D. J.

À partir d'images satellites, son travail est d'analyser des régions, zone par zone, et de classer ces zones: Telle zone est forestière, telle autre est urbaine, etc...

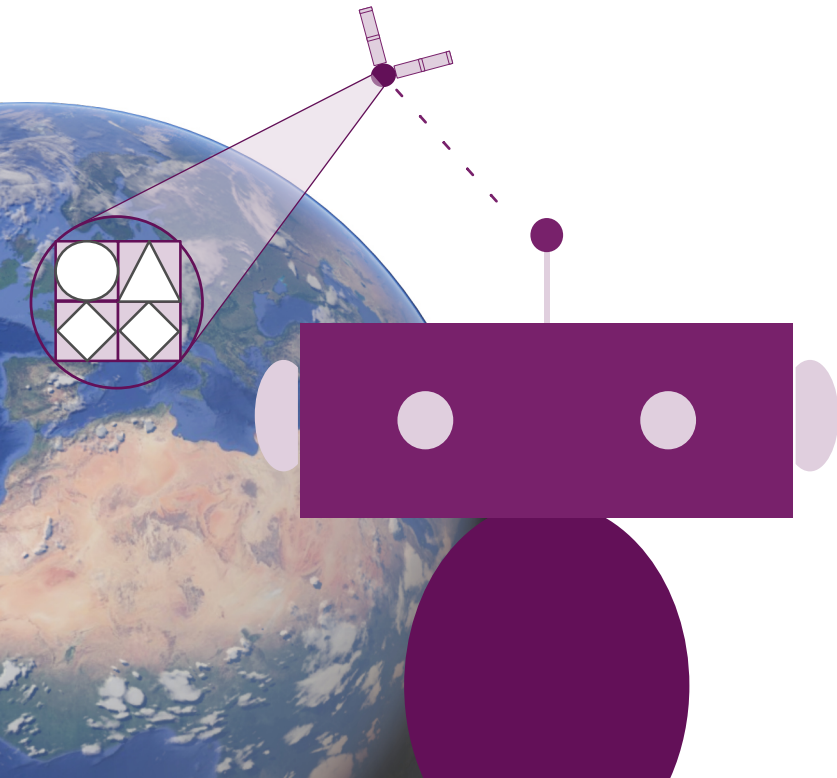


Cependant, parfois la région à traiter est trop grande pour MdJ. Il n'a pas le temps pour traiter toutes les zones.



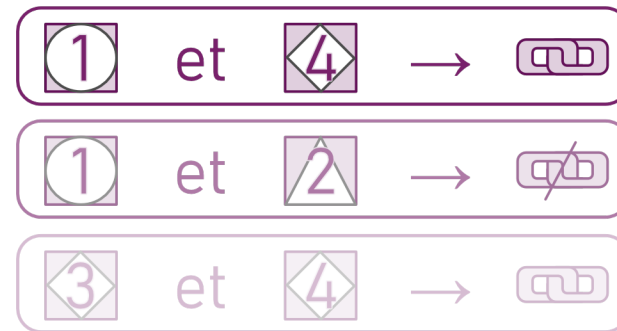
Alors, pour aller plus vite, il utilise des outils qui attribuent des classes aux zones automatiquement.

Au lieu de classer lui-même zone par zone, l'expert peut maintenant contraindre deux zones différentes à être de la même classe.



De son côté, l'outil peut alors chercher toutes les zones concernées par cette contrainte et agir à échelle de la région.

Mais pour extrapoler ces contraintes à d'autres zones, l'outil doit comprendre le lien entre ces deux zones et leur classe commune.



EXPERTISE

Comment apprendre une expertise
en un minimum d'interaction avec l'expert?

Plus précisément, l'outil doit résoudre deux problèmes:

1. "Comment apprendre une expertise à partir de ces retours?"
2. "L'expert ayant un temps limité, comment l'apprendre en minimisant ces interactions?"

1 et 4 → 

1 et 2 → 

3 et 4 → 



EXPERTISE

HERELLES

K et ♥ → ✓

K et ♣ → ✗

1 et ♥ → ✓



RÈGLES SECRÈTES

ELEUSIS

Comment apprendre les règles secrètes
en un minimum d'interactions avec le MDJ?

Pour répondre à ces questions, nous nous sommes penchés sur un problème analogue: le jeu d'Éleusis.

Dans ce jeu, le maître du jeu (MDJ) choisit des règles secrètes selon lesquelles il accepte ou refuse les cartes que les joueurs lui suggèrent.

Le problème est le même: "Comment apprendre des règles secrètes à partir des retours du MDJ?"

Dans ce cadre, pour chaque retour de l'expert:

- les deux zones correspondent à la valeur et le symbole d'une carte de jeu,
- la contrainte à une acceptation ou un refus.

ILCLASS

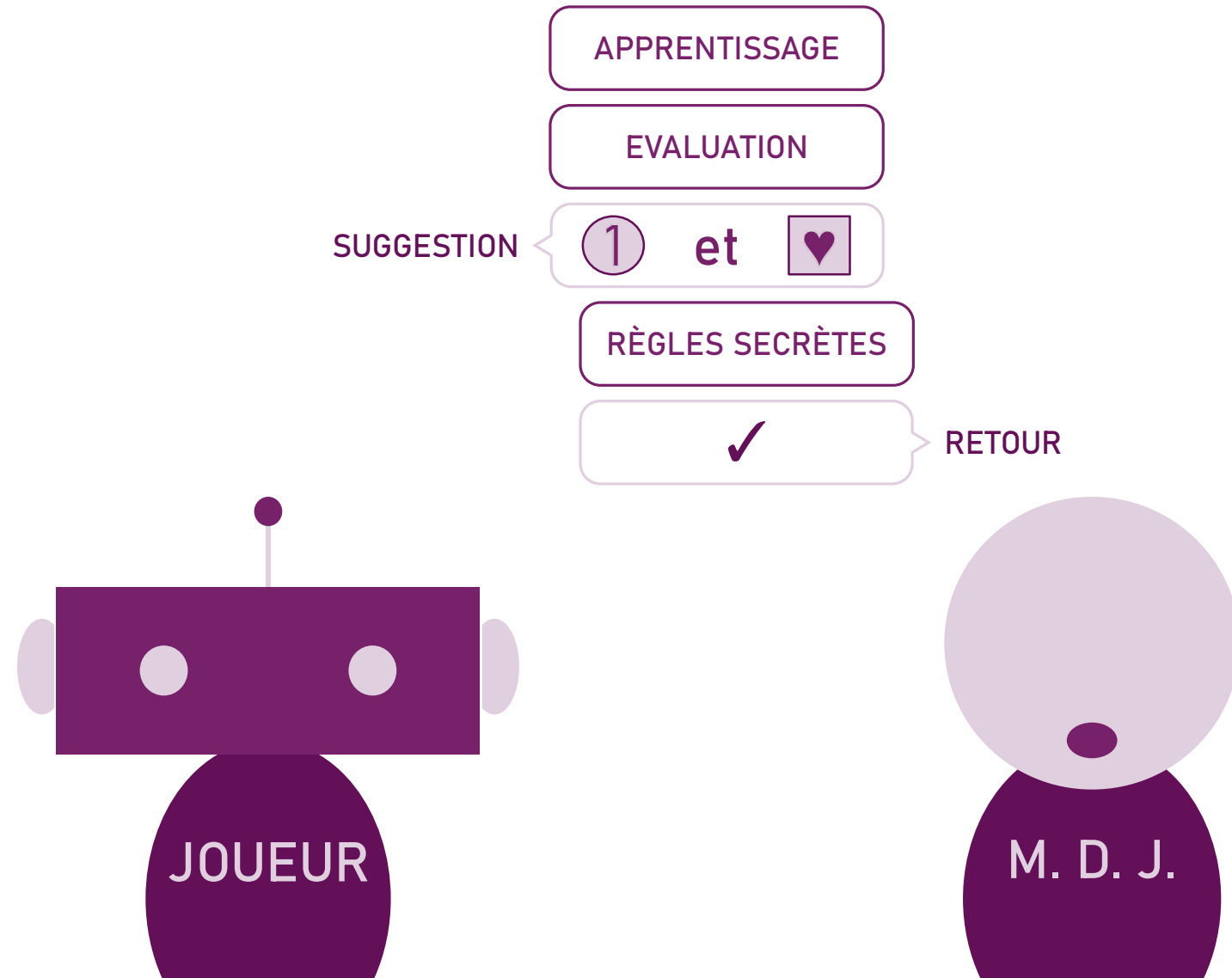
Apprentissage
Évaluation
Suggestion
Interaction

VALIDATION

Méthodologie
Priorité optimale
Performances

Maintenant que le contexte est posé, je vais vous présenter dans les grandes lignes:

- l'algorithme ILCLASS, qui prends le rôle du joueur dans une partie d'Eleusis.
- la validation de cet algorithme et ses résultats.



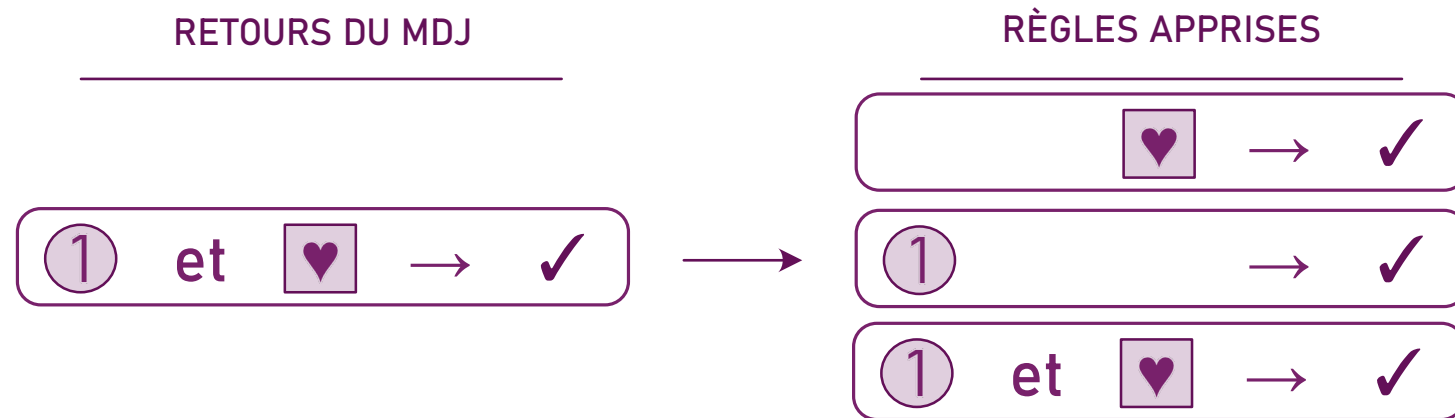
Lors d'une partie d'Éleusis, le joueur apprend des règles sous-jacentes aux retours du MDJ: initialement, il ne sait rien.

Il évalue alors les cartes qu'il n'a pas encore présentées: pour chaque carte, il déduit le retour du MDJ.

Quand une carte se démarque, il la suggère au MDJ: ici, le joueur ne sais rien de l'"As de Coeur"

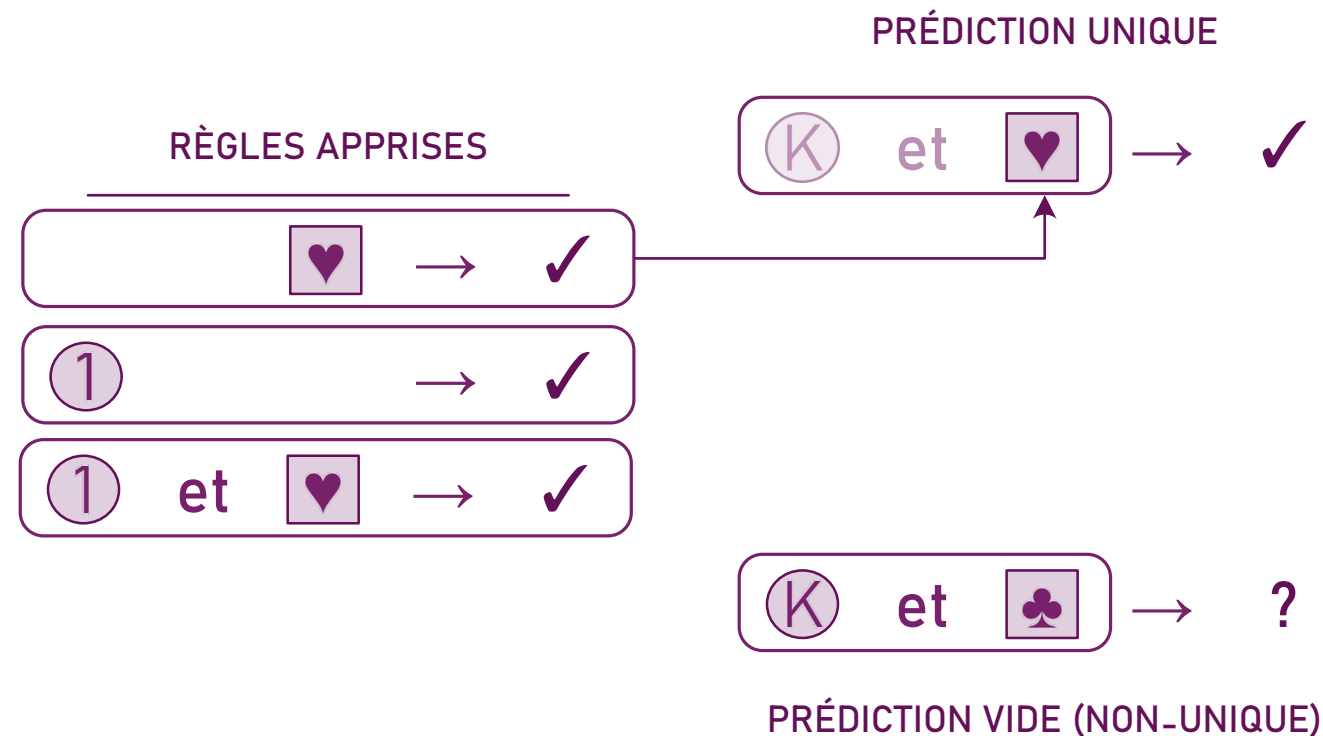
Le MDJ évalue alors cette carte avec ses règles secrètes et répond avec le retour approprié: ici, il valide.

Et on recommence...



Au fil d'une partie, la phase d'apprentissage apprend des règles à partir des retours du MDJ.

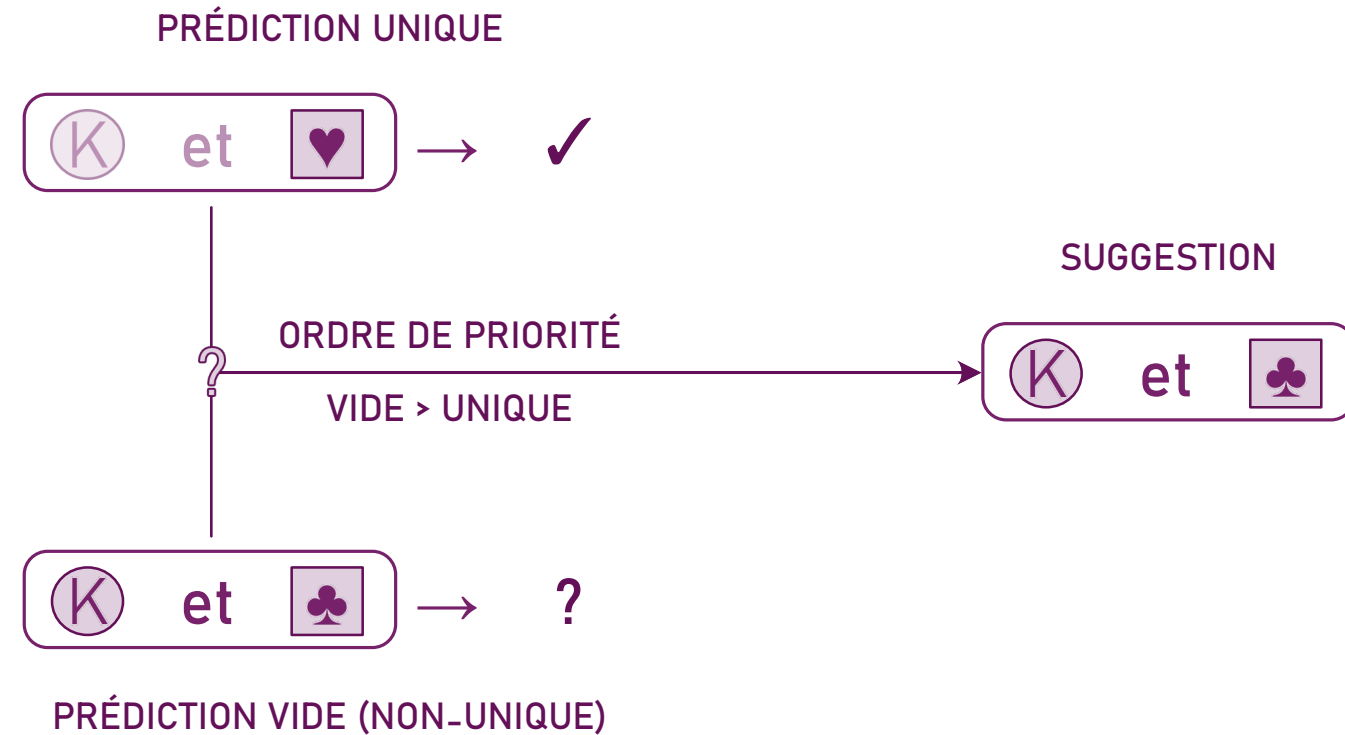
Ici, on sais que l'"As de Coeur" est validé: alors on apprend que jusqu'ici, par exemple, toutes les cartes Coeur sont validées.



Le joueur évalue alors, selon ces règles, les cartes qu'il n'a pas encore présentées: ici, le "Roi de Coeur" et le "Roi de Trèfle".

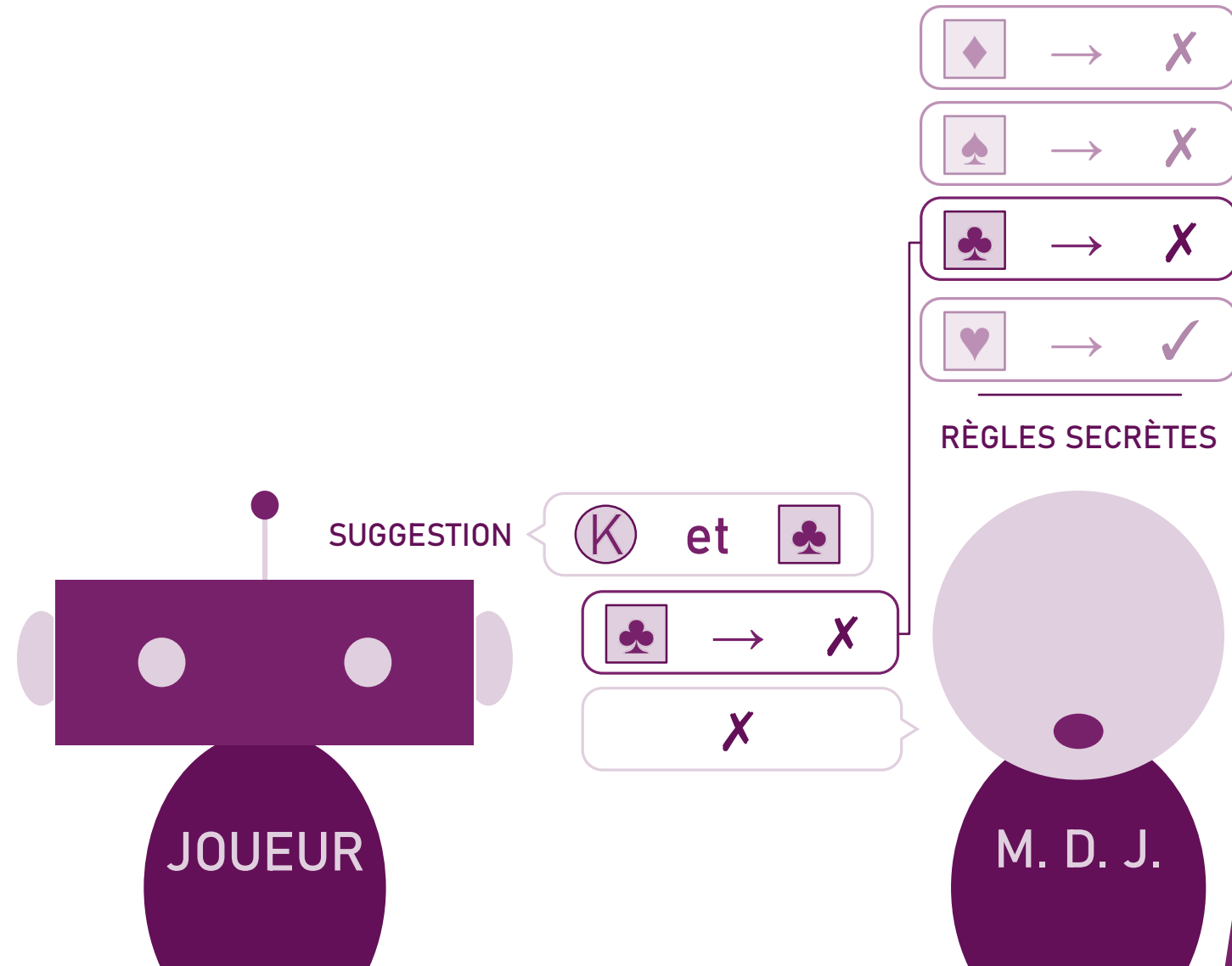
D'après nos règles, toutes les cartes "Coeur" sont validées, donc le MDJ devrait aussi valider le "Roi de Coeur": c'est une prédiction unique.

En revanche, nous ne pouvons rien prédire pour le "Roi de Trèfle": c'est une prédiction vide (ou Non Unique).



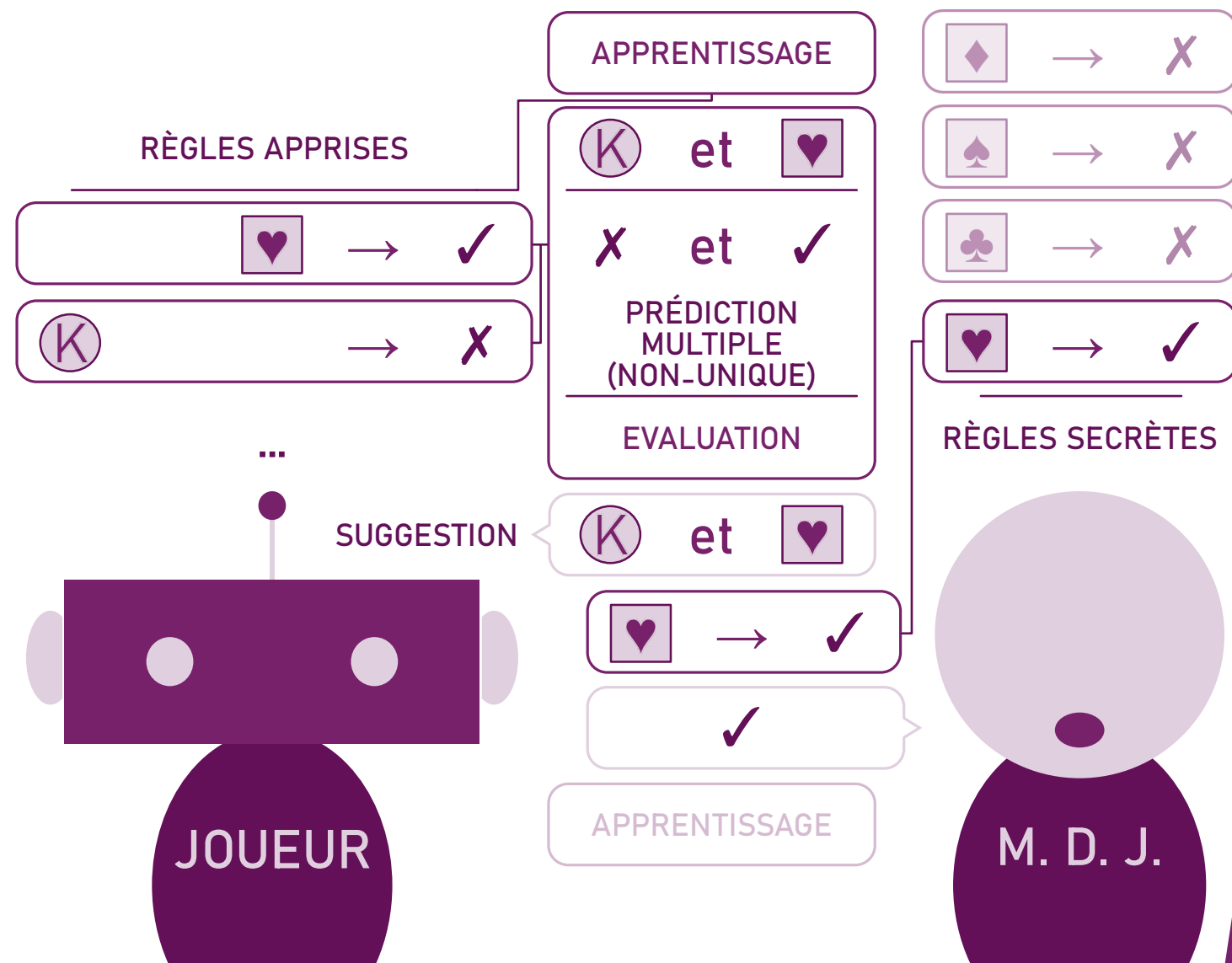
Au moment de choisir une suggestion parmi ces deux cartes inconnues, le joueur compare ce qu'il déduit pour chacune d'entre elles.

Ici, nous donnons la priorité aux cartes à prédiction vide avant celles à prédiction unique: nous suggérons donc le "Roi de Trèfle" au MDJ.



Dans notre exemple, le MDJ ne valide que les cartes "Coeur" et refuse les autres.

Le MDJ évalue alors cette suggestion "Roi de Trèfle" avec ses règles secrètes: c'est du "Trèfle", il refuse.



À partir de ce nouveau retour nous apprenons de nouvelles règles: par exemple, tous les “Roi” sont refusés.

Lors de l'évaluation, la carte inconnue “Roi de Coeur” se voit prédire un refus car c'est un “Roi”, mais aussi une validation car c'est du “Coeur”: C'est une prédiction multiple, non-unique.

Nous suggérons ce “Roi de Coeur”, le MDJ voit que c'est du “Coeur”: il valide.

Et ainsi de suite, nous apprenons de ce nouveau retour, suggérons une nouvelle carte, et caetera... Jusqu'à épuisement du jeu de cartes ou du MDJ.

Tout ce procédé: c'est notre algorithme ILCLASS.

CONVERGENCE

$$\text{RÈGLES APPRISES} \quad \left| \frac{\quad}{\# \text{ SUGGESTIONS}} \right| \quad \text{RÈGLES SECRÈTES}$$

PPNU (Pourcentage de Prédictions Non-Uniques)

$$\frac{\# \text{ PRÉDICTIONS NON-UNIQUES}}{\# \text{ PRÉDICTIONS}}$$

ACCURACY

$$\frac{\# \text{ PRÉDICTIONS CORRECTES}}{\# \text{ PRÉDICTIONS}}$$

Pour valider le bon fonctionnement de notre algorithme, nous nous sommes posé 3 questions:

- Combien de suggestions sont nécessaires avant que les règles apprises convergent vers les règles secrètes?
- Quelle quantité de cartes notre algorithme est-il incapable de classifier (Prédiction Non-Unique)?
- Quelle proportion de celles classifiées (Prédiction Unique) est correctement classifiée?

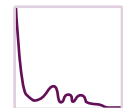
| ORDRE DE PRIORITÉ | |
|--------------------------|-----|
| MULTIPLE > VIDE > UNIQUE | MVU |
| VIDE > MULTIPLE > UNIQUE | VMU |
| VIDE > UNIQUE > MULTIPLE | VUM |
| MULTIPLE > UNIQUE > VIDE | MUV |
| UNIQUE > VIDE > MULTIPLE | UVM |
| UNIQUE > MULTIPLE > VIDE | UMV |
| ALÉATOIRE | RND |

En comparant les résultats de validation des ordres de priorité possibles, nous avons déterminé un optimum:

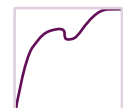
L'ordre de priorité MVU (Multiple > Vide > Unique).

MVU

CONVERGENCE \approx # MIN SUGGESTIONS NECESSAIRES



PPNU ACTIVEMENT MINIMISÉ



ACCURACY ACTIVEMENT MAXIMISÉ

TOUJOURS MEILLEUR QUE L'ALÉATOIRE

Et sur l'intégralité de nos tests, MVU...

- converge quasiment en le minimum de suggestions nécessaires,
- minimise activement la proportion de prédictions inclassifiables,
- maximise activement la proportion de classifications correctes,
- est meilleur que l'aléatoire.

1 et 4 → 

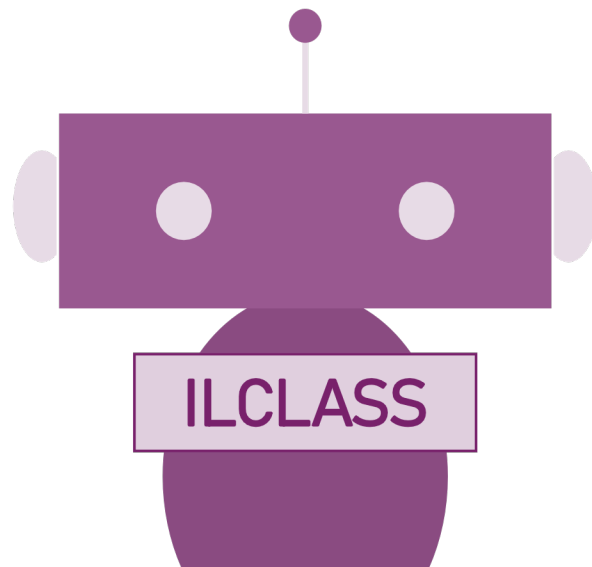
1 et 2 → 

3 et 4 → 



EXPERTISE

HERELLES
À VENIR



ILCLASS

K et ♥ → ✓

K et ♣ → ✗

1 et ♥ → ✓



RÈGLES SECRÈTES

ÉLEUSIS
RÉSOLU

En bref:

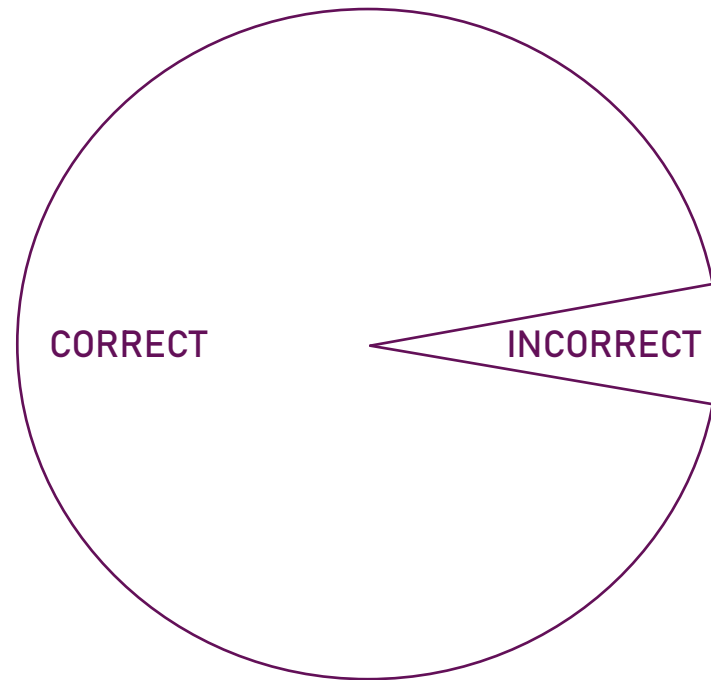
- Notre méthode résouds le problème d'Éleusis.
- La prochaine étape sera de tester ses performances dans le cadre d'Hérelles, avant de l'y intégrer.

Je m'y attellerai durant la thèse succédant à ce stage, qui démarrera en Octobre.

MERCI POUR VOTRE ATTENTION

Transcript

PREDICTIONS UNIQUES



ACCURACY

$$\begin{aligned} & \frac{\# \text{ PRÉDICTIONS CORRECTES}}{\# \text{ PRÉDICTIONS}} \\ & \quad \downarrow \approx \\ & \frac{\# \text{ PRÉDICTIONS UNIQUES}}{\# \text{ PRÉDICTIONS}} \\ & \quad \downarrow = \\ & 1 - \frac{\# \text{ PRÉDICTIONS NON-UNIQUES}}{\# \text{ PRÉDICTIONS}} \\ & \quad \downarrow = \\ & 1 - \text{PPNU} \end{aligned}$$

Dans nos tests, les prédictions uniques incorrectes sont une minorité négligeable: l'accuracy y est approximativement équivalente au pourcentage de prédiction uniques soit "1 - PPNU"