



# Euro Millions prediction

API applied to AI



# Euro Millions

- Jeu de loterie européen
- Tous les tirages sont indépendants
- Un tirage génère 5 numéros de 1 à 50 et deux numéros étoiles de 1 à 12.
- Un gagnant est défini lorsqu'il obtient 5 bons numéros et 2 bons numéros étoiles.



# Jeu de données

	A	B	C	D	E	F	G	H	I	J
	Date	N1	N2	N3	N4	N5	E1	E2	Winner	Gain
	24/12/2004	3	27	29	4	37	6	5	1	10000000
	17/12/2004	49	22	19	15	46	9	2	1	26215627
	26/11/2004	24	1	34	4	36	8	6	1	43756502
	22/10/2004	9	25	1	40	23	3	9	1	25059130
	10/01/2004	48	20	45	21	12	6	5	1	29785269
	08/06/2004	5	35	44	15	24	6	5	1	24746858
	18/06/2004	28	2	40	23	43	6	2	1	12286876
	06/04/2004	41	42	34	13	9	7	3	1	12488171
0	21/05/2004	37	39	29	15	49	4	9	1	10000000
1	14/05/2004	37	3	1	39	21	2	6	1	33816137



# But : Créer et servir un modèle prédictif

- Créer un modèle prédictif, qui prédit si une suite de 5 numéros et 2 numéros étoiles est gagnant.
- Entraîner ce modèle en utilisant les données d'entrée.



# But : Créer et servir un modèle prédictif

- Créer une application FastAPI permettant d'interagir avec ce modèle prédictif
  - POST /api/predict permet de réaliser une prédiction en fonction d'une proposition de tirage en entrée
    - La prédiction devra être probabiliste (Proba gain : X%, Proba perte : 1-X%).
  - GET /api/predict permet de générer une combinaison de numéros ayant une probabilité de gain élevée. (Définir élevée selon votre compréhension/étude de la donnée/modèle)
    - La prédiction devra être une suite de numéro, prédit gagnant, du modèle.



# But : Créer et servir un modèle prédictif

- Créer une application FastAPI permettant d'interagir avec ce modèle prédictif
  - GET /api/model permet d'obtenir les informations techniques du modèle
    - Métriques de performance
    - Nom de l'algorithme
    - Paramètres d'entraînement
  - PUT /api/model permet d'enrichir le modèle d'une donnée supplémentaire
    - Une donnée supplémentaire doit avoir le même format que le reste des données.
  - POST /api/model/retrain permet de réentraîner le modèle
    - Il doit prendre en compte les données rajoutées a posteriori



# But : Créer et servir un modèle prédictif

## Quelques informations supplémentaires

- Le modèle ne doit pas être réentraîné à chaque démarrage de l'application et doit donc être sauvegardé (sérialisation de l'objet modèle)
- Le modèle doit être sauvegardé à chaque réentraînement (inutile de garder des sauvegardes)
- Les hyperparamètres du modèle doivent être choisis intelligemment
- Une étude sur la donnée doit être faite au préalable



# Informations de rendu

- Fait en duo (si possible IA + ICC pour décupler le panel de compétences)
- Doit disposer d'un Readme qui détaille les choix techniques et l'installation de l'applcatif
- Le code doit être documenté en utilisant la Google Docstring
- Le code doit être aisément maintenable et lisible
- La nomenclature et la nature de l'architecture du code doit se suffire à elle-même
- Ce TP constituera la notre finale pour les IA et un bonus au projet final pour les ICC
- Chaque jour de retard entraîne une pénalité de 4 points
- Le rendu sera fait via une invitation à un projet **Github** et sera en public. Tâchez de le faire bien c'est votre vitrine pour vos premiers pas dans le monde professionnel =)





# Jalon maximal

Doit être rendu avant le **dimanche 19 décembre 2021**  
à 23h59

