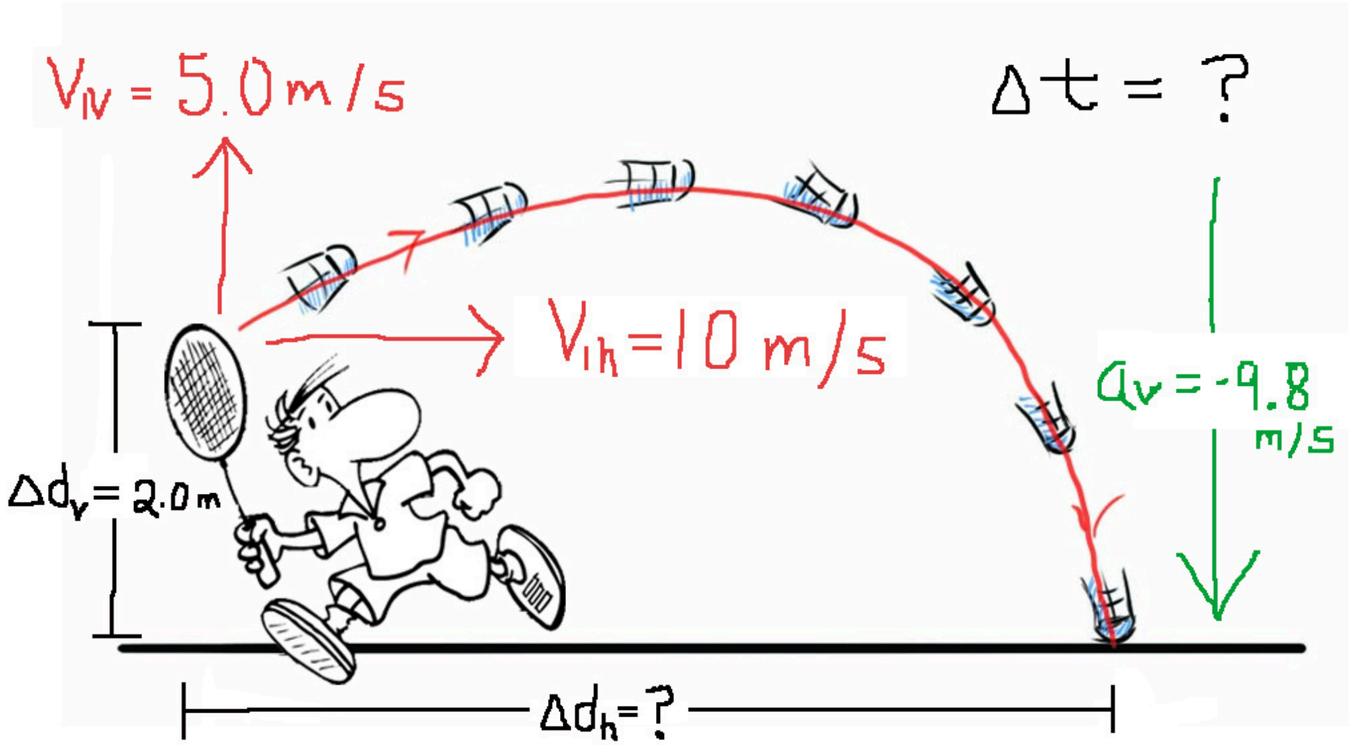
Trajectory Learning and Optimized Tracking in Simulated 2D Space


$V_{iv} = 5.0\ m/s$
$\Delta t = ?$
$V_{ih} = 10\ m/s$
$a_v = -9.8\ m/s$
$\Delta d_v = 2.0\ m$
$\Delta d_h = ?$

Les objets en mouvement sous l'influence de la gravité suivent des trajectoires prévisibles. Pour un projectile (comme un volant de badminton), sa position à tout moment est donnée par :

$$x(t) = v_x \cdot t$$
$$y(t) = v_y \cdot t - \frac{1}{2} g \cdot t^2$$

vx et vy sont les vitesses initiales respectivement suivant les directions x et y

Effet de la traînée et de la résistance de l'air

Les trajectoires dans le monde réel dévient en raison de forces comme la traînée

$$F_D = \frac{1}{2}(C_D \rho A V^2)$$

⚠ Le mouvement d'un volant de badminton est fortement influencé par la traînée, entraînant une descente plus abrupte par rapport aux paraboles idéales.

Effet de la traînée et de la résistance de l'air

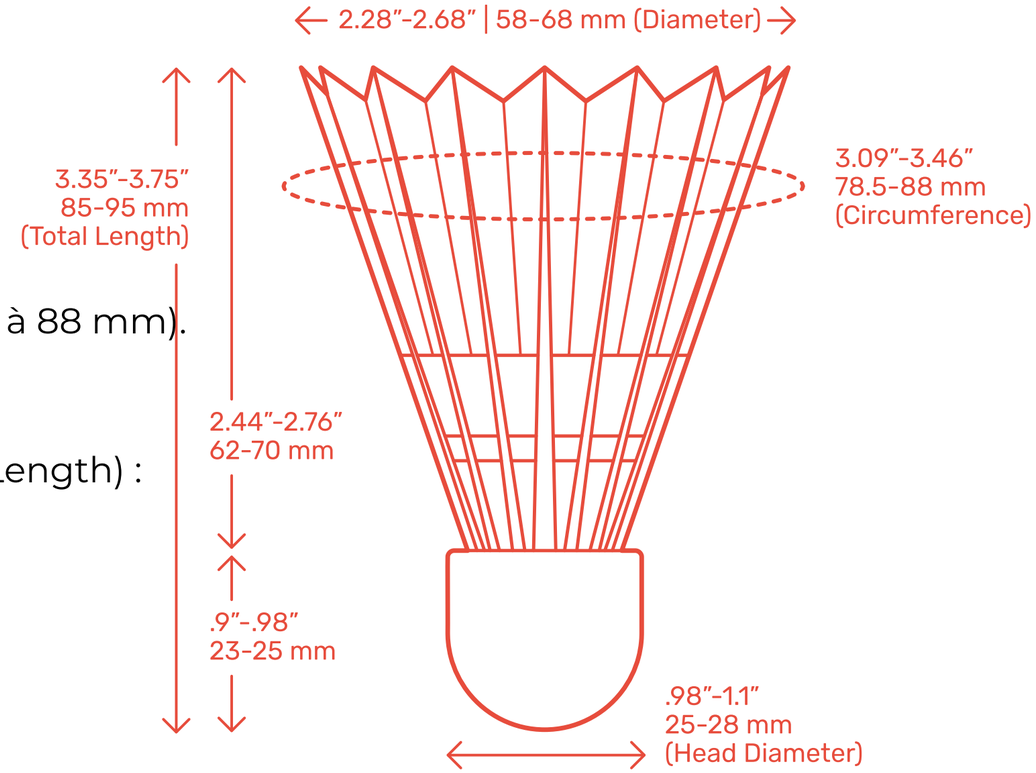
Les trajectoires dans le monde réel dévient en raison de forces comme la traînée

$$F_D = \frac{1}{2}(C_D \rho A V^2)$$

⚠ Le mouvement d'un volant de badminton est fortement influencé par la traînée, entraînant une descente plus abrupte par rapport aux paraboles idéales.

caractéristiques du volant du Badminton

- Longueur totale (Total Length) :
- Entre 3,35 et 3,75 pouces (85 à 95 mm).
- Diamètre du cercle supérieur (Circumference) :
- La circonférence du haut des plumes varie de 3,09 à 3,46 pouces (78,5 à 88 mm).
- Diamètre total du volant (Diameter) :
- De 2,28 à 2,68 pouces (58 à 68 mm).
- Hauteur entre la base des plumes et le sommet des plumes (Feather Length) :
- Entre 2,44 et 2,76 pouces (62 à 70 mm).
- Diamètre de la tête (Head Diameter) :
- Entre 0,98 et 1,1 pouce (25 à 28 mm).
- Hauteur de la base de la tête (Cork Height) :
- Entre 0,9 et 0,98 pouce (23 à 25 mm).



Pourquoi utiliser des réseaux de neurones pour l'apprentissage des trajectoires ?

Modélisation de dynamiques complexes :
Les réseaux de neurones peuvent approximer des fonctions non linéaires, ce qui les rend idéaux pour apprendre des trajectoires physiques influencées par la traînée et d'autres forces.

Gestion des données bruitées :
Les mesures du monde réel contiennent souvent du bruit. Les réseaux de neurones peuvent apprendre à lisser les données et à inférer les informations manquantes.

Flexibilité :
Ils s'adaptent à des conditions variées, telles que différentes vitesses, angles ou coefficients de traînée.

Le théorème d'approximation universelle

Tout réseau de neurones artificiels à une seule couche cachée, avec un nombre suffisant de neurones dans cette couche et une fonction d'activation non linéaire appropriée (comme la sigmoïde ,ReLU,tanh ...), peut approximer arbitrairement bien toute fonction continue définie sur un sous-ensemble compact de Rn .

Formulation mathématique :

Theorem (Cybenko)

Let σ be any continuous discriminatory function.
Then finite sums of the form

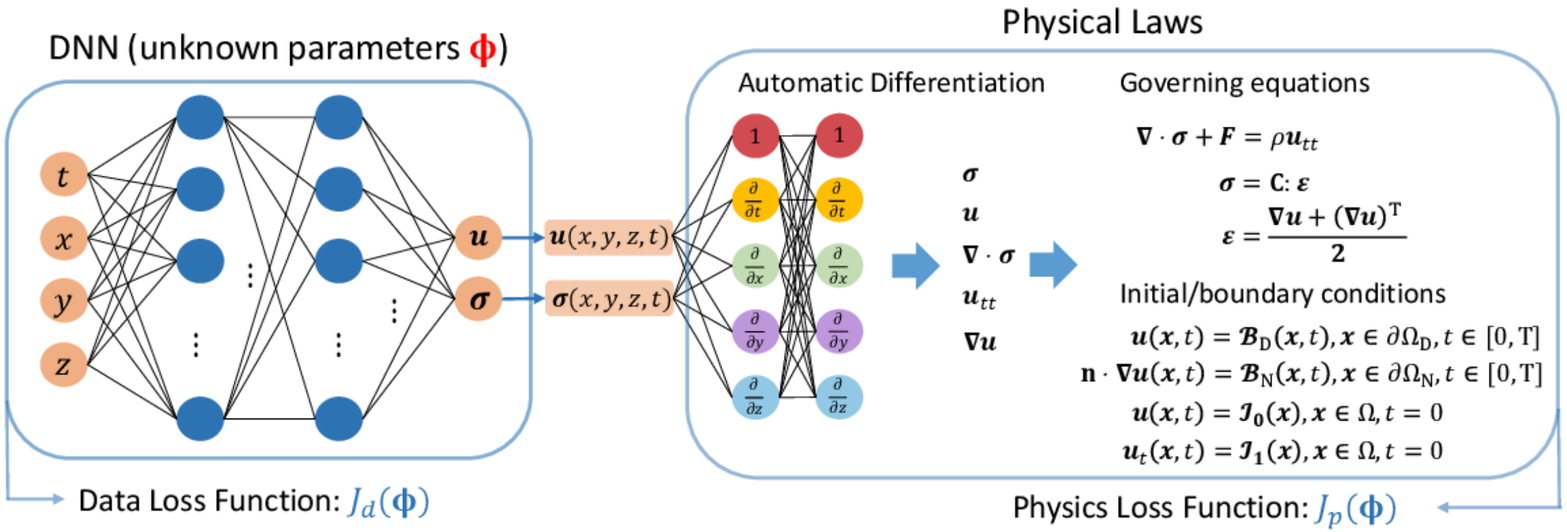
$$G(x) = \sum_{j=1}^{N} \alpha_j \sigma(w_j^T x + b_j), \text{ where } w_j \in \mathbb{R}^n, \ \alpha_j, \ b_j \in \mathbb{R}$$

are dense in $C(I_n)$.

In other words, given any $\varepsilon > 0$ and $f \in C(I_n)$, there is a sum $G(x)$ of the above form such that

$$|G(x) - f(x)| < \varepsilon, \quad \forall x \in I_n$$

PINNS



$$\widehat{\boldsymbol{\phi}} = \underset{\boldsymbol{\phi}}{\operatorname{argmin}} \left\{ J_d(\boldsymbol{\phi}) + J_p(\boldsymbol{\phi}) \right\}$$

Réseaux de Neurones Informés par la Physique (Physics-Informed Neural Networks - PiNNs)

Les PiNNs améliorent le processus d'apprentissage en combinant :

Apprentissage basé sur les données : À partir de données observées ou simulées.

Contraintes basées sur la physique : Issues des équations fondamentales comme les lois de Newton, la force de traînée, etc.

Ces réseaux de neurones sont conçus pour intégrer directement les lois de la physique dans leur processus d'entraînement,
garantissant que les prédictions restent cohérentes avec les principes physiques.

RÉSEAUX DE NEURONES POUR LES TRAJECTOIRES

Les trajectoires sont régies par des équations physiques bien définies (par ex. : gravité, résistance de l'air).

Avantages des PiNNs :

Efficacité des données : Nécessite moins de données, car la physique agit comme un régularisateur.

Robustesse au bruit : Les prédictions restent précises même en présence de données
bruitées ou incomplètes.

Généralisation : Apprend des trajectoires cohérentes avec les lois de la physique, même dans des conditions non observées.

GÉNÉRATION DE DONNÉES SIMULÉES POUR LA TRAJECTOIRE D'UN VOLANT DE BADMINTON

Processus de Simulation en étapes

Étape 1 : Définir les conditions initiales
Angle de lancement (θthetaθ) : 30°, 45°, 60°.
Vitesse initiale (V0) : Dépend du type de coup (par ex. smash, dégagement) et du niveau du joueur .

Étape 2 : Calculer les trajectoires
Utiliser les équations du mouvement pour calculer x(t)x(t)x(t) et y(t)y(t)y(t) à chaque intervalle de temps.

Étape 3 : Ajouter du bruit et des données manquantes
Introduire un bruit **gaussien** pour simuler les imprécisions du monde réel.
Simuler des mesures manquantes pour tester la robustesse.

Pourquoi utiliser un bruit gaussien dans le contexte de la simulation de trajectoires ?

1. Représentation réaliste des erreurs du monde réel
Le bruit gaussien (ou bruit normal) est utilisé car il reflète les erreurs naturelles et aléatoires souvent rencontrées dans les mesures physiques
Dans le monde réel, les instruments de mesure introduisent des imprécisions qui suivent généralement une distribution normale autour de la valeur vraie.

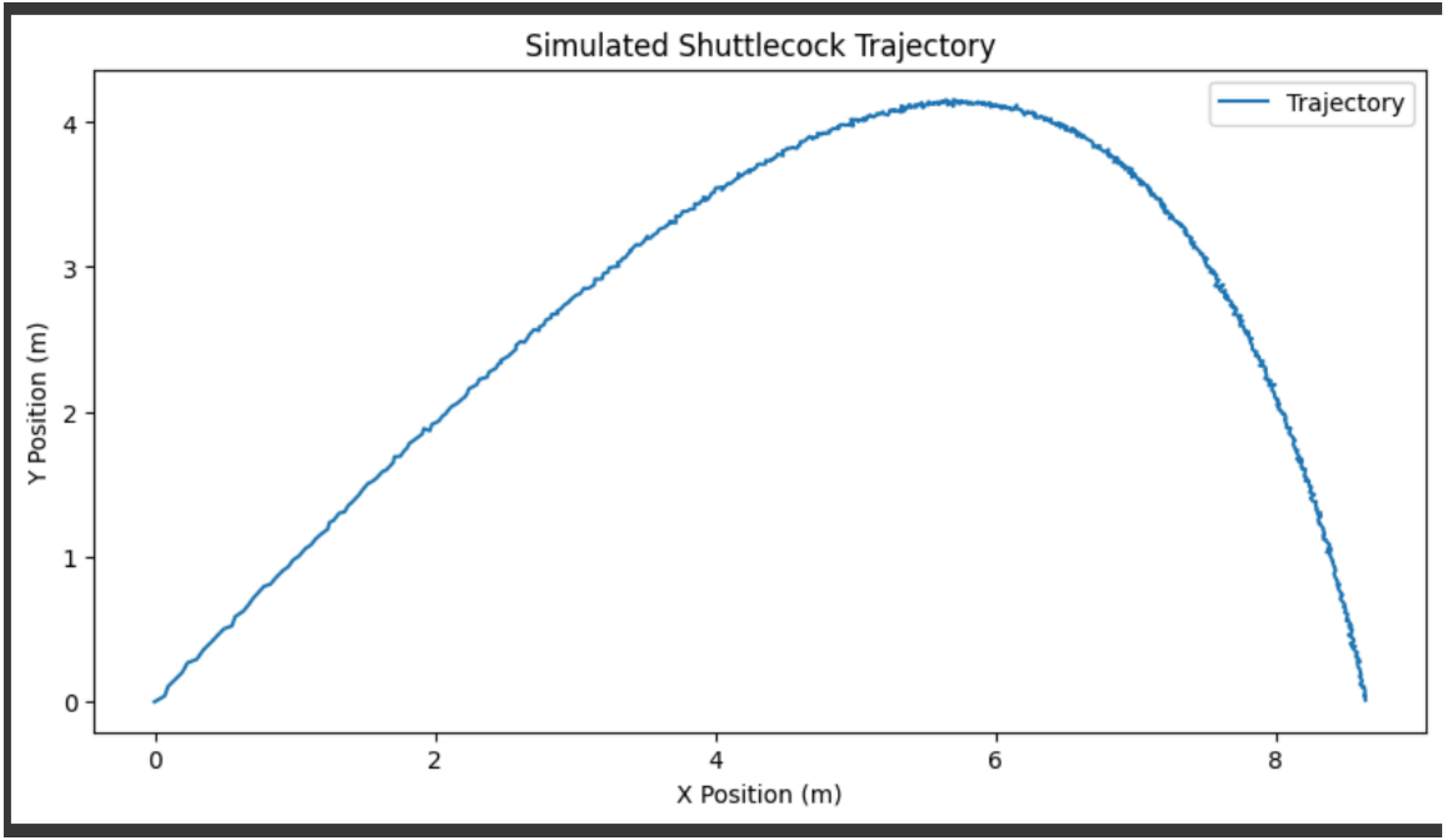
2. Caractéristiques statistiquement pertinentes
Distribution symétrique : Le bruit gaussien est centré autour de la moyenne (généralement 0), ce qui permet de modéliser des erreurs sans biais.
Variance contrôlable : La variance (σ^2) détermine l'étendue des fluctuations, permettant d'ajuster le degré de bruit pour s'adapter aux scénarios réalistes.

3. Lien avec la loi centrale limite
La prévalence du bruit gaussien dans les données du monde réel peut être expliquée par la loi centrale limite. Selon ce théorème, la somme de nombreuses variables aléatoires indépendantes (même si elles ne suivent pas une distribution normale) tend à suivre une distribution normale lorsque le nombre de ces variables est suffisant. Cela explique pourquoi les erreurs issues de multiples sources indépendantes prennent naturellement la forme d'un bruit gaussien.

4. Robustesse et tests
En ajoutant du bruit gaussien :
On teste la capacité des modèles (par ex. réseaux de neurones) à généraliser et à faire face à des données bruitées.
Cela aide à déterminer si les prédictions restent fiables même avec des imperfections dans les données.

Angle = 40°

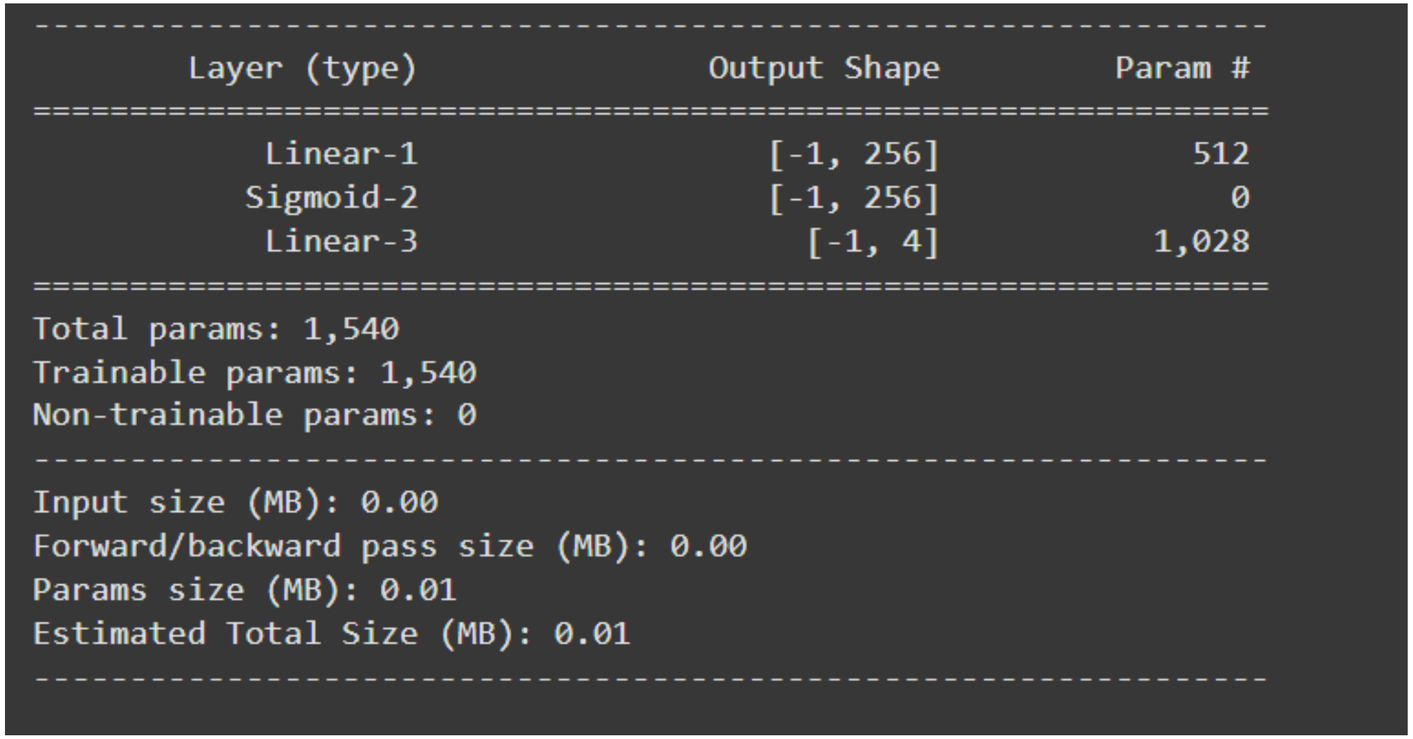
Vitesse initiale = 30 m/s



1 er Essai

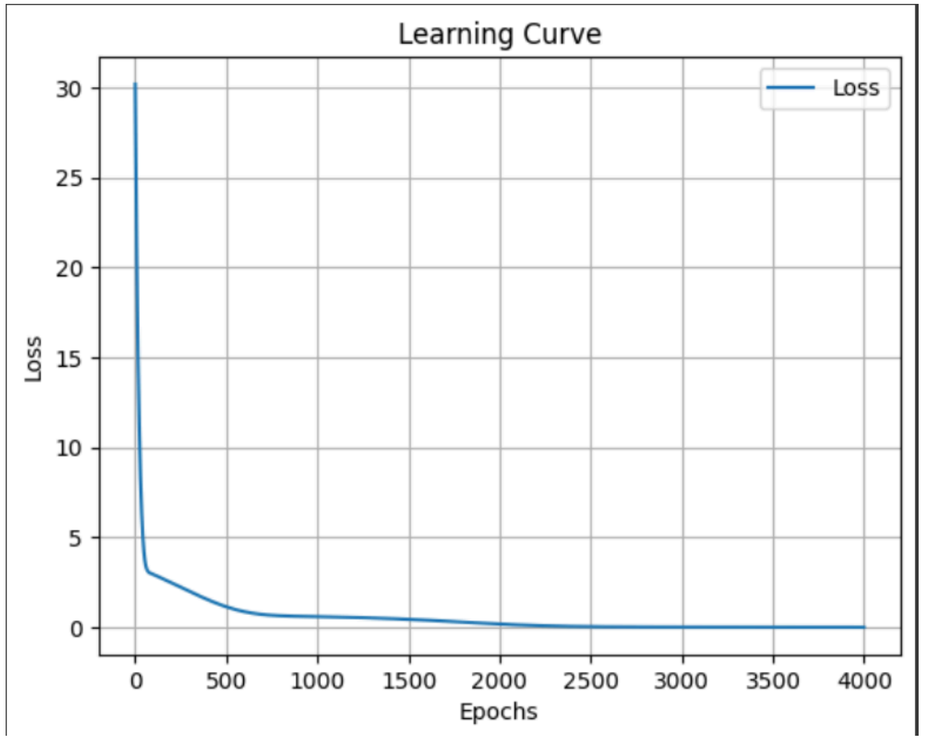
Architecture du modèle

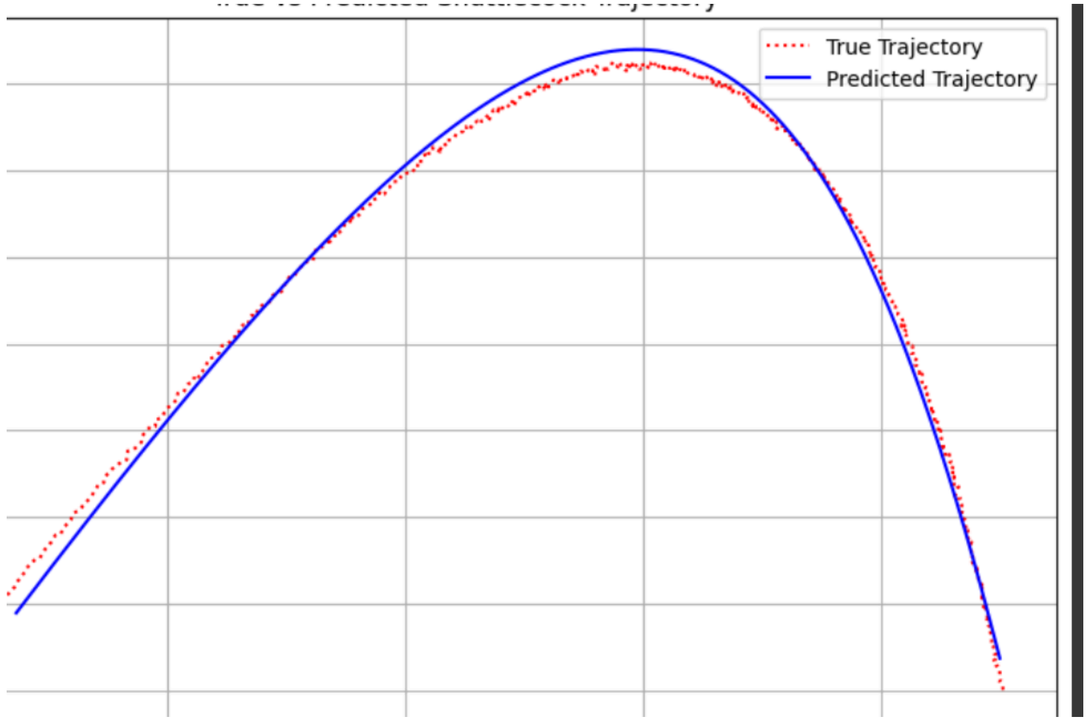
- Type de modèle : Réseau de neurones à une couche cachée
- Entrée : Vecteur unidimensionnel (1,)
- **Couches** :
  a. Couche Linéaire 1 : 1 → 256 neurones (512 paramètres)
  b. Activation Sigmoid : Non-linéarité appliquée à la sortie de la première couche
  c. Couche Linéaire 2 : 256 → 4 neurones (1,028 paramètres)
- **Sorties** : 4 valeurs (x, y, v_x, v_y)
- Total des paramètres : 1,540 (entraînables)



```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Linear-1                [-1, 256]             512
           Sigmoid-2                [-1, 256]               0
            Linear-3                  [-1, 4]           1,028
================================================================
Total params: 1,540
Trainable params: 1,540
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.00
Forward/backward pass size (MB): 0.00
Params size (MB): 0.01
Estimated Total Size (MB): 0.01
----------------------------------------------------------------
```

Résultat





Pas mal mais doit penser maintenant à faire des optimisations

1 ère optimisation

1. Cross-Validation (CV)

But : Évaluer la performance du modèle sur plusieurs sous-ensembles de données.
Avantages:
Réduit le risque de surapprentissage : Chaque modèle est testé sur des données qu'il n'a pas vues.
Estimation plus robuste de la performance du modèle.

Processus : Divise les données en plusieurs parties (k-fold), s'entraîne sur k-1 parties et valide sur la partie restante.

2. Grid Search avec Optuna

But : Optimiser les hyperparamètres pour améliorer la performance du modèle.
Optuna : Un outil de recherche d'hyperparamètres automatisé.

Avantages:

Exploration efficace de l’espace des hyperparamètres.
Algorithme de recherche adaptatif : Utilise des stratégies comme l’optimisation bayésienne pour explorer les paramètres.
Gain de temps : Réduit la recherche exhaustive du Grid Search classique.

2 ème optimisation

Régularisation L1 et L2

1. Qu'est-ce que la régularisation ?
La régularisation est une technique utilisée pour éviter le surapprentissage (overfitting) en ajoutant une pénalité sur la complexité du modèle.

2. L1 (Régularisation Lasso) :

Réduit les poids non pertinents en les forçant à devenir zéro.

3. L2 (Régularisation Ridge) :

Réduit la magnitude des poids mais ne les annule pas complètement.

4. Pourquoi utiliser les deux ?

Combinaison L1 et L2 (ElasticNet) : Bénéficie des avantages des deux régularisations, en permettant une sélection de variables tout en maintenant la stabilité des poids.

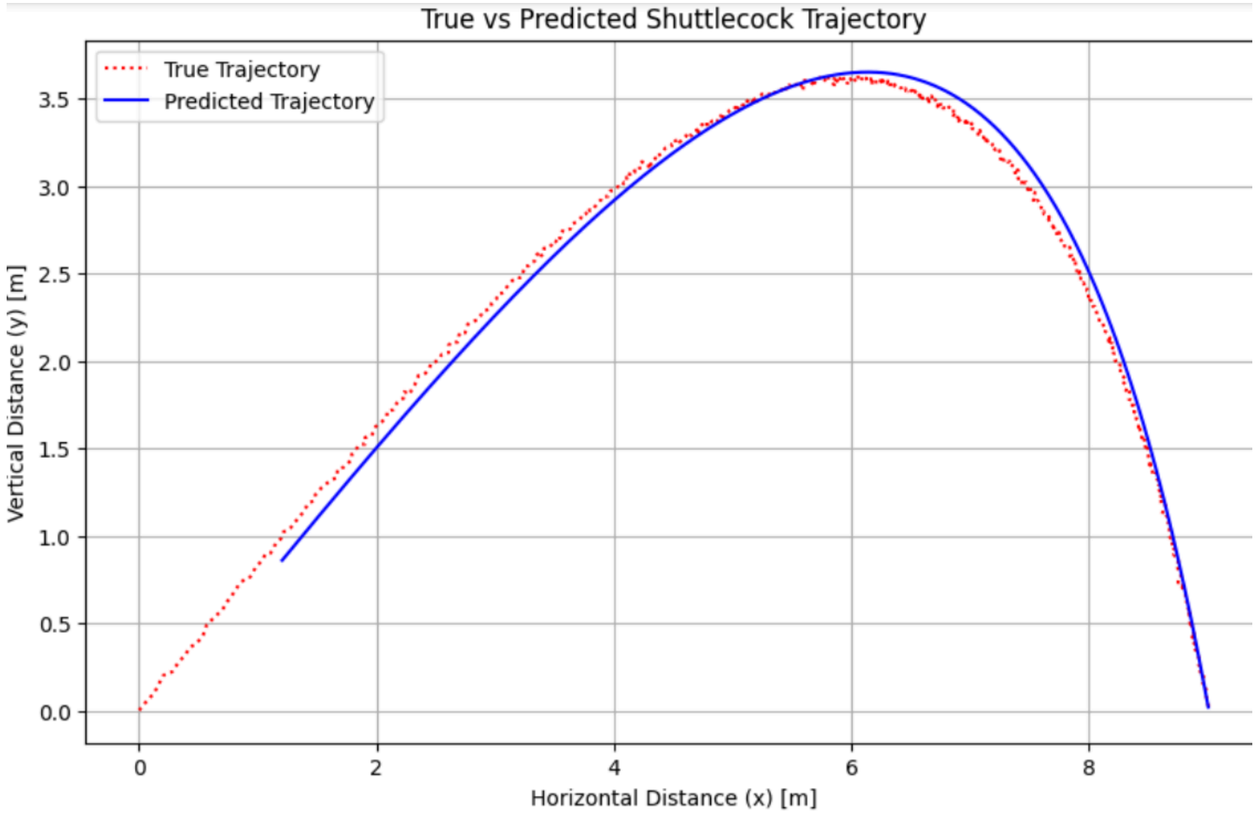
Résultat sur 1000 epochs

```
Best hyperparameters:  {'num_neurons': 512, 'learning_rate': 5.919237038363878e-05, 'activation_fn': 'Tanh'}
Best trial's loss:  0.8925165653228759
```

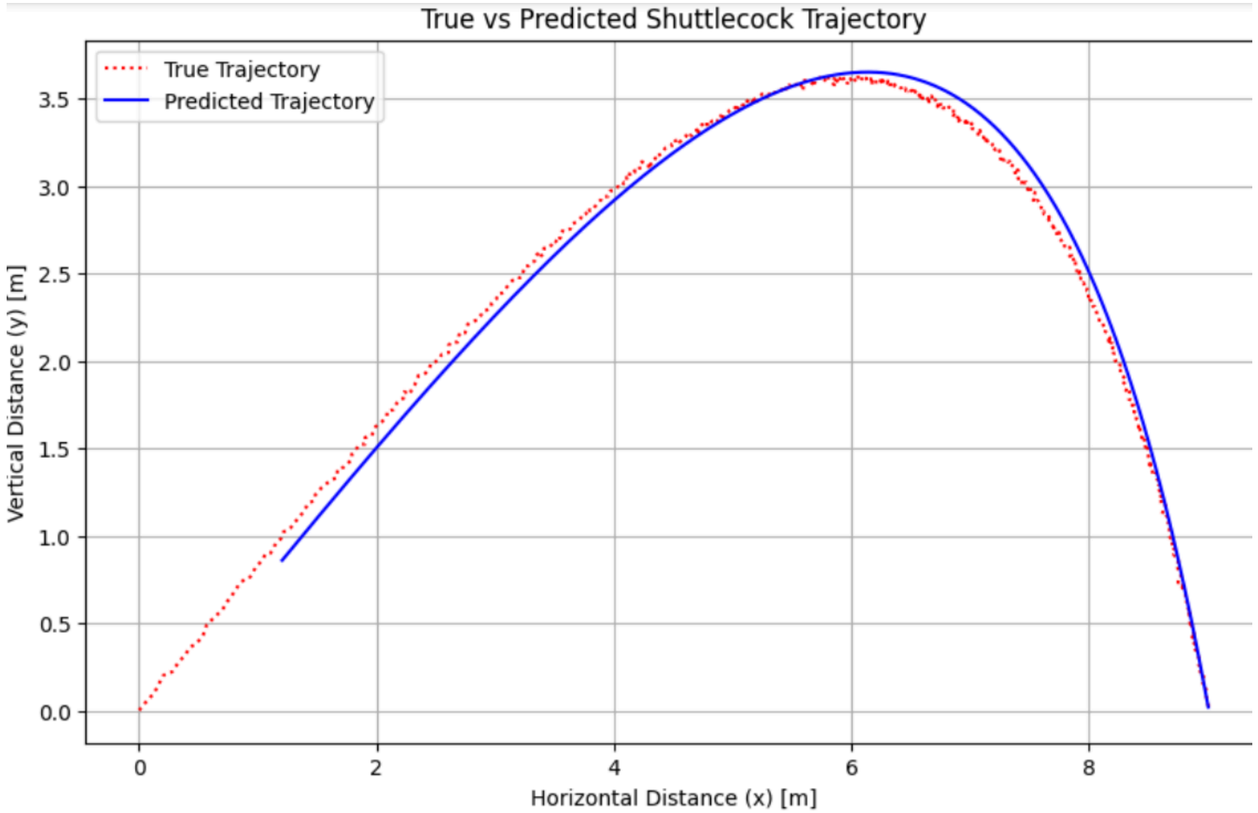
Donc, l'architecture optimale à utiliser est celle contenant 512 neurones, un learning rate de 5.919e-05 et une fonction d'activation tangente hyperbolique.

on va donc entrainer cette architecture sur toute la data sur 4000 epochs

Résultat final



Résultat


True vs Predicted Shuttlecock Trajectory
True Trajectory
Predicted Trajectory
Vertical Distance (y) [m]
Horizontal Distance (x) [m]
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
0
2
4
6
8

Évaluation du modèle

Metric	Value
Mean Absolute Error (MAE)	0.7366
Root Mean Squared Error (RMSE)	0.901
Accuracy (%)	1
False Positives (FP)	0
False Negatives (FN)	64
Inference Latency (seconds)	0.0014

Merci pour votre attention