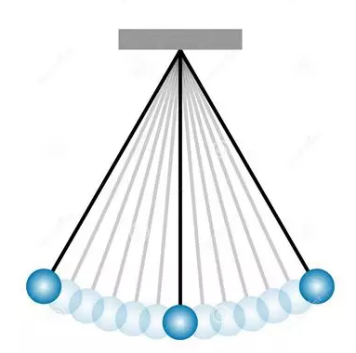
## Intelligence Artificielle et Analyse de données

TP1 : Prédiction de trajectoires (réseaux récurrents)



On s’intéresse à la dynamique d’un pendule :

connaissant la valeur de l’angle aux instants , on cherche à prédire la valeur aux instants .

Pour cela, 3 *datasets* (*train, valid, test*) sont fournis.

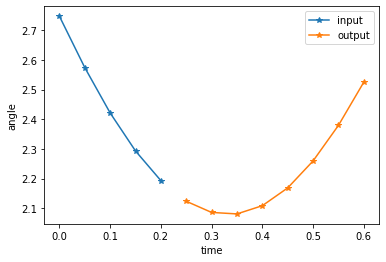
Une donnée d’un *dataset* est constituée de ( où

* et sont des vecteurs de temps (pas nécessaires à l’entraînement, car le temps d’échantillonage est constant),
* un vecteur d’angle aux instants (données d’entrée) et
* un vecteur d’angle aux instants (données de sortie, à prédire).

1. Comprendre les données :

Tracer quelques trajectoires du dataset *train*, sur le même graphique.

Voici un exemple d’une trajectoire.



1. Entraîner un réseau RNN (Elman) :

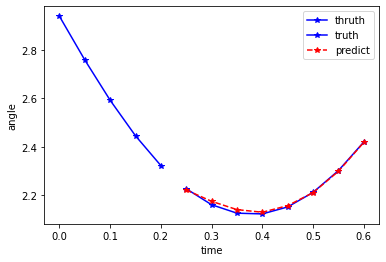
Implémenter la boucle d’entraînement (*epochs*).

Mémoriser les valeurs de coût dans les tableaux *train\_losses* et *valid\_losses*. En fin d’entraînement, tracer leur évolution, et analyser (stabilité, sur-apprentissage, …).

Modifier les hyperparamètres pour améliorer les performances.

1. Exploitation du modèle :

Tracer une courbe prédite (utiliser le dataset *test*) par le réseau RNN et comparer avec la donnée.



1. Prédictions avec un MLP :

Le MLP servira de *baseline* pour mieux évaluer les performances du RNN.

Conseil : Utiliser la fonction flatten (*x\_flat = x.flatten(start\_dim=1)*) pour adapter les données séquentielles en vecteurs (dans la fonction *forward* du MLP).