

Rapport Analyse Quantitative en Python

Stratégie de DNN :

Notre stratégie consiste à prédire la direction du mid price en utilisant un deep neural network (DNN).

Création de Features :

En utilisant les attributs de la classe LimitOrderBook `limit_bids` et `limit_offers`, nous avons récupéré les 5 meilleurs bids, les 5 meilleurs offers et les montants respectifs à chaque fois qu'il y a une mise à jour de ces derniers. Ils sont stockés dans la variable `best_offer_best_bid`.

Ces données vont être les features de notre apprentissage.

Construction de la base de Train

Nous avons construit la classe `ML_Strategy` à partir de la classe `Momentum_Strategy`.

Le nombre de données historiques utilisées pour fitter le modèle est défini par l'attribut `__lookback` : int

La première moitié du dataset va servir de la base de train `X_train`

La deuxième moitié du dataset va servir pour construire la variable de tendance `Y_train`.

Nous calculons le mid price future avec cette dernière.

Condition	Y train	Signification
$\text{Mid price} * (1 + \alpha) < \text{Mid price futur}$	1	Mid price va baisser
$\text{Mid price} * (1 - \alpha) > \text{Mid price futur}$	2	Mid price va augmenter
Les autres cas	3	Mid price va stagner

Lorsque la longueur de `best_offer_best_bid` est égale au paramètre `lookback`, nous allons train et fit le modèle.

Lorsque la longueur de `best_offer_best_bid` est supérieur au paramètre `lookback`, nous allons réinitialiser `best_offer_best_bid`

Lorsque la longueur de `best_offer_best_bid` est inférieure ou égale au paramètre `lookback`, nous allons prédire la variable de tendance avec la dernière valeur de `best_offer_best_bid` avec le modèle stocké dans l'attribut `__dnn_model`.

Définition des layers :

En plottant le graphe de l'historique de la couche de perte ('loss'), nous observons que la courbe converge donc le nombre de layers est optimal et il n'y pas de surapprentissage.

Prédiction :

La variable de tendance prédite sera stockée dans l'attribut `__predict_probability` : float

L'historique de `predict_probability` sera stockée dans l'attribut `__all_predict_probability`:list

Trading way :

Après avoir prédit la direction du mid price, la stratégie va donner un trading way en fonction de la direction du mid price, le prix de la dernière cotation et le mid price actuel. Le tableau ci-dessous récapitule les conditions des trading ways.

Conditions	Trading way
$0.98 < \text{Predict_proba} < 1.02$ and le prix de la dernière cotation \leq mid price actuel	Buy
$2.98 < \text{Predict_proba} < 3.02$ and le prix de la dernière cotation $<$ mid price actuel	Buy
$1.98 < \text{Predict_proba} < 2.02$ and le prix de la dernière cotation \leq mid price actuel	Sell
$2.98 < \text{Predict_proba} < 3.02$ and le prix de la dernière cotation $>$ mid price actuel	Sell
Les autres cas	Ne pas changer de position

Cependant, nous remarquons que cette stratégie est très lente pour le High Frequency Trading et que le résultat est peu satisfaisant. Nous avons donc implémenté la stratégie de RSI.

Stratégie d'indicateur RSI :

Nous avons calculé l'indicateur Relative Strength Index (RSI) pour chaque cotation. Il est calculé de la manière suivante :

$RS = \text{Average of } x \text{ days' up closes} / \text{Average of } x \text{ days' down closes}.$

$RSI = 100 - 100 / (1 + RS)$

Si le RSI est supérieur à 70, cela signifie que le titre est suracheté. Les prix ont fortement augmenté et que le prix risque de baisser. Il faut donc vendre.

Si le RSI est inférieur à 30, cela signifie que le titre est survendu. Les prix ont fortement chuté et que le prix pourrait augmenter. Il faut donc acheter.

Résultat sur le dataset du 18 Janvier:

Position 1: profit bps 0.000300, draw down bps 0.000660

Position 2: profit bps 0.000300, draw down bps 0.000380

Total 2 positions opened.

Total profit (loss) in basis points is: 0.00.

Maximal draw down in basis points is: 0.000660.

Calmar ratio: 0.909091.

Number of x: 1800.

Target Profit: 0.000300.

Total transaction price: 12.00.

Total profit (loss): 180.00.

Net profit (loss): 168.00.

Librairies nécessaires : tensorflow numpy os

Ressources utilisées :

Adamantios Ntakaris, Martin Magris, Juho Kanninen, Moncef Gabbouj, Alexandros Iosifidis. Benchmark Dataset for Mid-Price Prediction of Limit Order Book data. Nov 2019

<https://towardsdatascience.com/understanding-different-loss-functions-for-neural-networks-dd1ed0274718>

<https://admiralmarkets.com/fr/formation/articles/indicateurs-forex/indicateur-rsi>