

Cahier des charges client

Fonctionnalités

- Après avoir choisi :
 - les caractéristiques de l'option,
 - la date à partir de laquelle je veux récupérer des données de marché,
 - la date de fin de mon test,
 - si les données sont simulées, semi-historiques ou historiques,
 - la fenêtre d'estimation des paramètres,

je veux obtenir en sortie le payoff de l'option, la valeur du portefeuille de couverture correspondant¹ et la *tracking error* du test : $\frac{V_{Lpf} - payoff}{prix_option}$.

- Après avoir fourni les mêmes entrées que précédemment, je veux pouvoir comparer à chaque date la valeur de mon portefeuille de couverture et le prix théorique de mon option en affichant les courbes de ces deux valeurs sur un même graphe.
- Je veux pouvoir spécifier :
 - la date à partir de laquelle je veux récupérer des données de marché²,
 - la date de fin du test,
 - si les données sont simulées, semi-historiques ou historiques.

Je veux que des valeurs par défaut soient rentrées dans l'interface.

- Je veux pouvoir spécifier la taille de la fenêtre d'estimation directement dans l'interface. Je veux qu'une valeur par défaut soit entrée dans l'interface.
- Je veux pouvoir donner en amont les caractéristiques de plusieurs options et choisir dans l'interface l'option à tester.
- Je veux pouvoir spécifier la fréquence de rebalancement directement dans l'interface. Je veux qu'une valeur par défaut soit entrée dans l'interface.
- Je veux pouvoir choisir dans l'interface si mes paramètres seront estimés lors du test, ou sont codés en dur.
- Je veux pouvoir spécifier les caractéristiques des options à tester dans un fichier qui sera chargé au lancement de l'application.
- Je veux pouvoir ajouter/supprimer des options directement dans l'application.

¹ Aucune précision n'est donnée sur la façon dont les paramètres sont fournis, ils peuvent donc être codés en dur dans l'application.

² Attention : il ne s'agit pas de la date de début du test, puisqu'il faut tenir compte de la fenêtre d'estimation.

Remarques générales

- **Attention** : nous devons être consultés avant toute modification du comportement de l'algorithme (ex : modification des paramètres de calcul sans en notifier l'utilisateur)
- L'ergonomie de l'application sera prise en compte dans la notation.
- L'architecture logicielle doit être réfléchie.
- Toute erreur doit être rattrapée, avec un message explicatif pertinent. **Attention** : un *try-catch* global n'est pas considérée comme une bonne gestion d'erreur