



ANALYSE DE L'ALGORITHME DE FORCE BRUTE

AVANTAGES

- Explore toutes les combinaisons possibles pour trouver la solution optimale.

INCONVÉNIENTS

- Complexité temporelle exponentielle : $O(2^n)$.
- Consommation mémoire élevée pour stocker toutes les configurations.
- Pas adapté aux problèmes de taille moyenne ou grande.



PSEUDOCODE DE L'ALGORITHME GLOUTON



PSEUDOCODE

- Calcul des ratios « rentabilité » : on calcule pour chaque action le ratio rendement/cout, soit (%) / 100.
- Initialisation :
 - budget_utilise = 0 €
 - benefice_realise = 0 €
 - investissements = []
- Boucle de sélection gloutonne : on parcourt les actions triées par ratio décroissant, et on « achète » si le coût entre encore dans le budget.
- Résultats finaux :
 - Actions retenues : [10, 6, 13, 19, 4, 20, 5, 11, 18, 17, 16, 14]
 - Budget utilisé : 498 €
 - Bénéfice total réalisé : 97,48 €
 - Valeur totale (budget initial + bénéfice) : 597,48 €



LIMITES & CAS CRITIQUES DE L'ALGORITHME GLOUTON



AVANTAGES

- Très rapide et simple à mettre en œuvre.
- Fonctionne bien dès que les rendements sont relativement homogènes.

INCONVÉNIENTS

- Ne trouve pas toujours la meilleure combinaison : on ne peut pas acheter de fractions d'action, donc il reste parfois du budget inutilisé.
- Peut passer à côté d'actions de forte valeur mais coûteuses si leur ratio rendement/prix est faible.
- Ses performances varient selon la répartition des rendements et des coûts : il faut toujours vérifier les résultats sur des cas réels.



COMPARAISON DES ALGORITHMES

ALGORITHME	COMPLEXITÉ TEMPORELLE	CONSOMMATION MÉMOIRE
FORCE BRUTE	$O(2^n)$	EXPONENTIELLE
ALGO GROUTON	$O(n \log n)$	$O(n)$



RAPPORT D'EXPLORATION

Après vérifications, les calculs de Sienna concernant les profits sont corrects. Cependant, la solution proposée n'est pas optimale.

Afin d'améliorer le résultat, j'ai développé un nouvel algorithme, qui s'appuie sur une approche gloutonne. Cet algorithme lit les deux fichiers dataset.csv contenant toutes les actions disponibles à l'achat, sélectionne les actions en respectant le budget alloué et génère une solution plus rentable que celle obtenue initialement par Sienna.

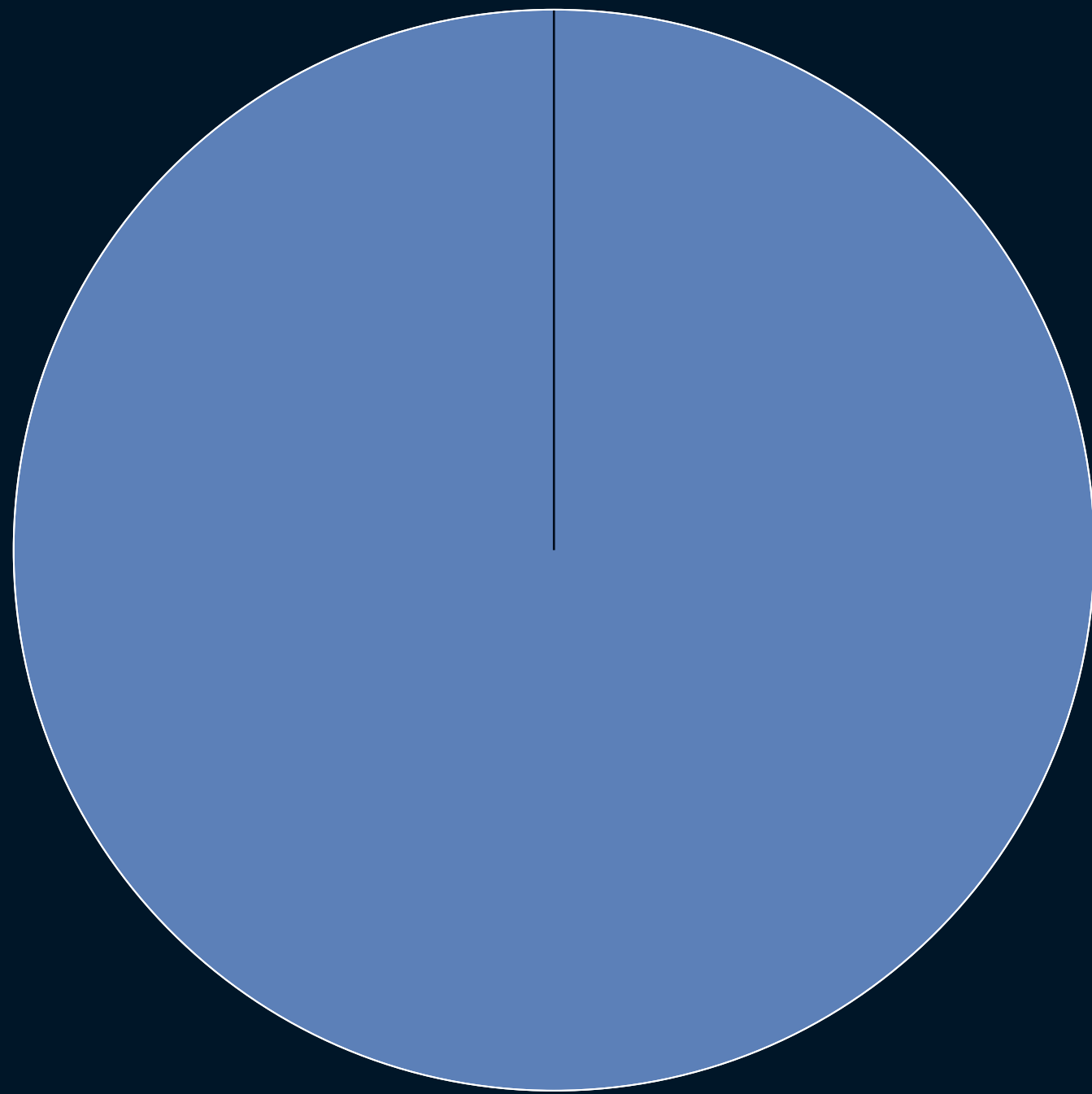
L'algorithme n'offre pas la solution optimale globale, car il utilise une stratégie gloutonne. Cette approche permet d'obtenir rapidement un résultat satisfaisant sans avoir à explorer toutes les combinaisons possibles (ce qui serait extrêmement long et coûteux en temps de calcul).

En résumé, la solution proposée est meilleure que celle de Sienna, mais pas nécessairement la meilleure solution. Elle représente un compromis entre rapidité d'exécution et augmentation du profit obtenu.

Ainsi, bien que la méthode développée ne garantisse pas une optimalité absolue, elle permet de proposer une solution de qualité supérieure à celle initialement retenue, tout en respectant des contraintes de temps de calcul réalistes. Cette approche démontre l'intérêt pratique de méthodes heuristiques dans des contextes nécessitant un compromis entre performance et rapidité d'exécution.

DATASET - 1

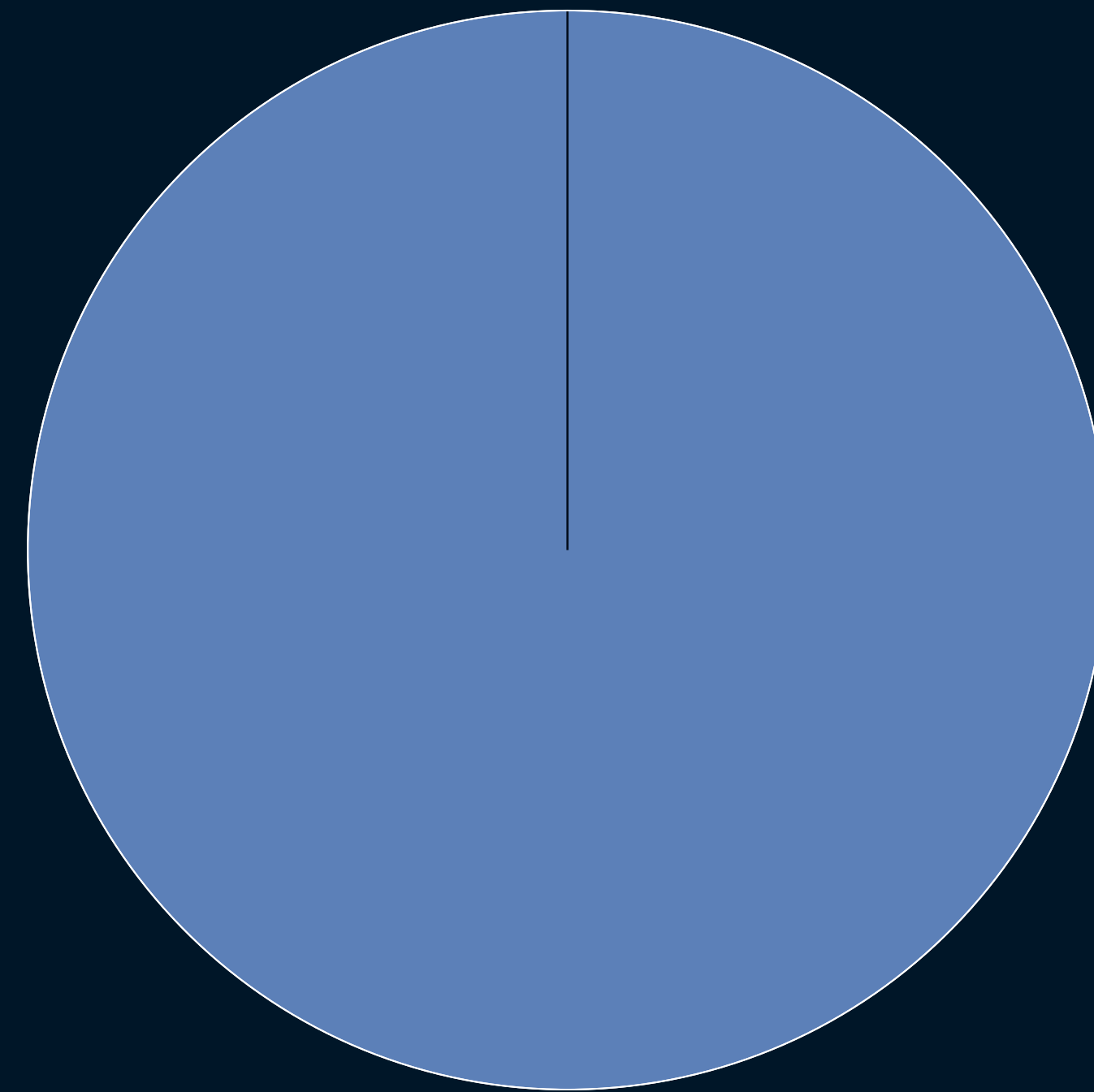
BUDGET UTILISÉ : 499.94 €



PROFIT : 198.47 €

DATASET - 2

BUDGET UTILISÉ : 499.98 €



PROFIT : 197.77 €