Statistique Bayésienne

Projet : Estimation de mouvements boursiers via une approche Bayésienne avec Python

ENSAE 2020-2021

Mastère Spécialisé - Data Science

Nathan BRY - Bertrand VUILLEMOT

Contents

1 Introduction

L'objectif de ce projet est d'utiliser les apports de la statistique bayésienne à des fins prosaïques. En effet, l'idée est ici de calculer la probabilité a posteriori de mouvements de la bourse en ayant connaissance des mouvements passés. L'apport de la statistique bayésienne dans ce projet est d'utiliser de l'information publique et accessible à tous. Nous tenterons de confirmer ou d'infirmer que l'apport d'information, aussi minime soit-elle nous permet de générer des rendements supérieurs à un l'indice boursieur majeur, le S&P 500.
Cet indice, dit *tracker* est l'indice référence du marché boursier américain (couvrant 80% de la capitalisation totale américaine), regroupant la cotation de 500 entreprises faisant partie du *NYSE* et du *NASDAQ*. Le support de notre projet ayant été présenté, en quoi l'approche bayésienne nous permet-elle d'investir de manière plus efficiente ?
Et bien un investisseur sans connaissance particulière en finance, en choisissant d'investir un dimanche soir aura tendance à regarder les évolutions (donc les rendements) de la semaine passée puis celle précédente afin de prendre sa décision. Nous proposons donc à cet investisseur d'analyser lesdits mouvements afin de choisir la stratégie à adopter dès le lundi matin. Pour cela, nous procéderons en plusieurs étapes : dans un premier temps la philosophie du projet sera détaillée, notamment le background financier. Dans un second temps, l'implémentation statistique sera abordée en présentant les outils bayésiens nous permettant de traiter les données des deux dernières semaines afin d'investir le lendemain. Enfin, dans une dernière partie, nous procéderons au backtest de la méthode, c'est-à-dire la calcul du rendement obtenu en faisant confiance aux outils de la seconde partie. Nous jugerons ainsi de l'efficacité de l'approche.

2 Philosophie

Nous allons spécifier l'approche technique puis les hypothèses que nous formulées sur la stratégie d' investissement de l'indice.

2.1 Approche

Nous adoptons une approche technique, c'est-à-dire que nous jugeons des mouvements de l'indice pour le jour suivant selon les mouvements passés. Donc en aucun cas dans ce projet nous jugerons de la valeur fondamentale de cet indice. Cette approche est très plébiscitée, notamment chez les investisseurs peu éduqués à l'évaluation financière. En effet, combien de fois avons-nous entendu "A ce prix, il faut acheter ça ne va pas plus baisser" ou encore "Une semaine que la bourse monte, elle va sûrement baisser demain". Cette approche heuristique, bien que décriée, trouve pourtant écho chez de nombreux boursicoteurs. Elle a prouvé son efficacité car elle revêt une dimension comportementaliste et par son intuitivité.
Afin de procéder à l'analyse des courbes, nous obtenons les données grâce à Yahoo Finance en gardant uniquement l'*Adjusted Close*. Il s'agit de la valeur de l'indice en fin de journée, corrigée des versements de dividendes et des *splits*. Nous gardons donc la substantifique moelle de l'indice "brut" pour nous concentrer sur les méthodes bayésiennes.

2.2 Hypothèses

Pour la stratégie nous formulons trois hypothèses :

1. Premièrement, l'investisseur a le choix entre trois stratégies : acheter, vendre à découvert ou ne rien faire. Il achète lorsque la probabilité a posteriori de croissance de l'indice est importante (i.e. P(Up) supérieure à 0.5), vend si la probabilité de baisse est importante (i.e. P(Down) supérieure à 0.5) et ne rien faire autrement.

2. Deuxièmement, nous prenons une fenêtre mobile de 10 jours de variations (soit 11 journées) afin de calculer la distribution apriori des journées Up, Down et Flat.
   En effet, une fenêtre plus petite réduirait de manière conséquente l'informativité de l'apriori. Par ailleurs, une fenêtre plus importante nous empêcherait de prendre en compte la condition actuelle du marché boursier sur laquelle nous souhaitons inférer.

3. Enfin, nous supposons l'absence de frais lors du passage d'ordres de l'investisseur. Cette hypothèse bien que contraignante se justifie par le développement d'applications sans commission (eToro, Robinhood, ...) à destination des investisseurs dits "retail" (à savoir vous et nous).

3 Methode & Implémentation

Nous utiliserons donc Python pour implémenter les différentes méthodes Bayésiennes. Intéressons-nous tout d'abord aux données.

3.1 Les données

Comme dit plus haut, nous utiliserons les données des adjusted close du **S&P 500**.
Pour cette première implémentation nous utiliserons une période de 11 jours ouvrés afin d'avoir 10 évolutions d'un jour au suivant (Up, Flat ou Down), puis nous calculons les probabilités de chaque outcome pour le 12ème jour. La période sélectionnée ici sera donc de 11 jours ouvrés (= 11 séances de bourse), du 16 au 31 Décembre 2021. Ces données proviennent de la librairie *yfinance*, développée par Yahoo Finance et disponible dans l'environnement Python.

3.2 La méthode

Les différentes étapes seront détaillées ci-dessous.

Tout d'abord il faut classifier nos 10 évolutions. Nous avons utilisé la méthode suivante :

- Une hausse de + de 0.1% sera considérée comme une hausse (**Up**)
- Une baisse de + de 0.1% sera considérée comme une baisse (**Down**)
- Toute évolution dans ce range [-0.1%,+0.1%] sera considérée comme plate (**Flat**)

Voici les résultats obtenus pour la période utilisée (16/12/2021 - 31/12/2021) : 5 séances haussières, 1 séance neutre et 4 séances baissières (voir *image 1*). Dans une approche fréquentiste les probabilités des outcomes pour le jour suivant seraient alors naturellement de (0.5 - 0.1 - 0.4), comme illustrées sur *l'image 2*.

```
L'indice a monté (en % des journées) : 50.0
L'indice n'a pas bougé (en % des journées): 10.0
L'indice a baissé (en % des journées): 40.0
[5, 1, 4]
```

Figure 1: Moves over the observed period

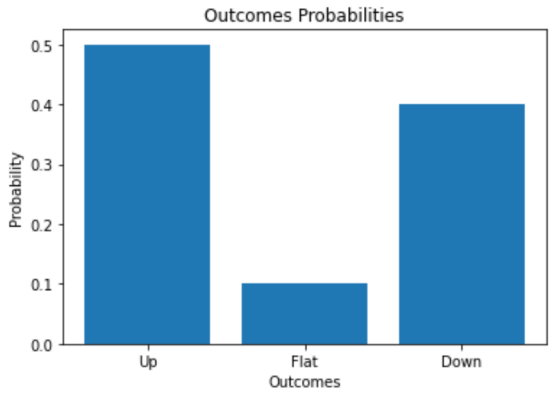

Figure 2: Outcomes Frequentist Probabilities

Le modèle est composé des différentes composantes suivantes (chacune sera développée ultérieurement):

- Le modèle sous-jacent est une distribution multinomiale avec des paramètres p_k
- La loi a priori de p_k est une Loi de Dirichlet
- Le vecteur α est un paramètre de la Loi de Dirichlet

Une loi multinomiale avec un a priori Dirichlet est définie comme une Loi Multinomiale de Dirichlet (`https://en.wikipedia.org/wiki/Dirichlet-multinomial_distribution`).

Modèle :

$$\begin{array}{lcll} \boldsymbol{\alpha} & = & (\alpha_1, \ldots, \alpha_K) & = \text{ concentration hyperparameter} \\ \mathbf{p} \mid \boldsymbol{\alpha} & = & (p_1, \ldots, p_K) & \sim \text{Dir}(K, \boldsymbol{\alpha}) \\ \mathbb{X} \mid \mathbf{p} & = & (x_1, \ldots, x_K) & \sim \text{Mult}(K, \mathbf{p}) \end{array}$$

L'objectif est d'obtenir p_{Up}, p_{Flat}, p_{Down} sachant le vecteur d'observation $c = [c_{Up}, c_{Flat}, c_{Down}]$

Le vecteur α est un hyperparamètre (`https://en.wikipedia.org/wiki/Hyperparameter`), un paramètre d'une loi a priori. Ce vecteur peut à son tour avoir sa propre loi a priori, appelée hyperprior (`https://en.wikipedia.org/wiki/Hyperprior`). Cependant nous n'utiliserons pas d'hyperprior, nous nous contenterons de spécifier les valeurs des hyperparamètres.
Le vecteur hyperparamètre peut être considéré comme un pseudo-count, que nous utilisons pour montrer notre croyance a priori quant à la prévalence de chaque issue (outcome) possible. Si nous voulons un hyperparamètre uniforme reflétant que nous croyons que la probabilité d'avoir une certaine issue est la même pour chaque issue, nous utilisons la même valeur pour chaque alpha, comme par exemple $\alpha = [1, 1, 1]$. Nous pouvons augmenter ou diminuer l'effet des a priori en augmentant ou en diminuant les valeurs de α. Cela peut être utile lorsque nous avons plus ou moins confiance en nos croyances a priori. Nous verrons les effets de l'ajustement des hyperparamètres plus tard dans le rapport et le notebook, mais nous travaillerons principalement avec $\alpha = [1, 1, 1]$.

Déterminons maintenant la prévalence de chaque issue. Une façon d'obtenir une estimation ponctuelle de la prévalence est d'utiliser la valeur attendue de l'a posteriori de p_k. La valeur attendue d'une Loi Dirichlet-Multinomiale est :

$$\text{E}[p_i \mid \mathbb{X}, \boldsymbol{\alpha}] = \frac{c_i + \alpha_i}{N + \sum_k \alpha_k}$$

Avec $\alpha = [1, 1, 1]$, et le vector des observations $c = [5, 1, 4]$, on obtient les prevalences attendues :

$$p_{Up} = \frac{6}{13} = 46.2\%$$

$$p_{Flat} = \frac{2}{13} = 15.4\%$$

$$p_{Down} = \frac{5}{13} = 38.5\%$$

Comme dit précédemment, augmenter ou diminuer les valeurs des α influe considérablement sur l'impact des a priori. On peut voir sur *l'image 3* que plus on augmente les valeurs α, plus on réduit le poids des données historiques et donc des a priori. Nos probabilités tendent alors vers des probabilités égales (3 outcomes ==¿ $p = 0.33$).

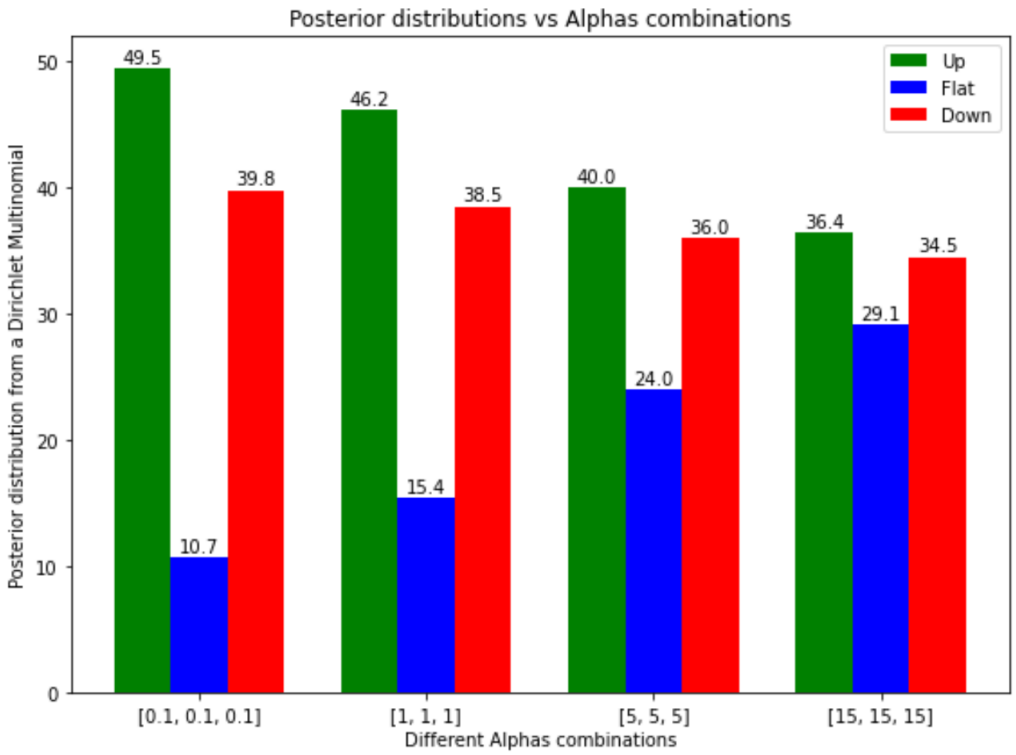

Figure 3: Posterior distributions vs Alphas combinations

Cependant, si les valeurs attendues sont un bon résultat pour estimer un seul essai, nous n'avons pas encore d'intervalle permettant d'exprimer notre incertitude. Pour pouvoir en constuire nous devons passer à la construction et à l'échantillonnage à partir d'un modèle bayésien. Pour rappel, nous utilisons une loi Multinomiale comme modèle, une Loi de Dirichlet comme a priori, et un vecteur hyperparamètre fixé. L'objectif est de trouver les paramètres de la loi multinomiale, p_k qui sont la probabilité de chaque espèce compte tenu des données.

3.3 Le modèle

Nous allons construire un modèle en utilisant la librairie PyMC3 (`https://docs.pymc.io/`) et ensuite utiliser une variante de la chaîne de Markov Monte Carlo pour tirer des échantillons de l'a posteriori. Avec suffisamment d'échantillons, l'estimation convergera vers le véritable a posteriori. En plus des estimations ponctuelles (comme la moyenne des echantillons), le MCMC nous donne également une notion incertitude car nous obtenons des milliers de valeurs possibles à partir de l'a posteriori.

Comme dit précédemment, nous utiliserons $\alpha = [1, 1, 1]$ pour notre modèle (*Image 4*). Les résultats sont visibles sur l'*Image 5*, les plus intéressants étant ceux surlignés, à savoir la valeur moyenne et l'intervalle des valeurs possible pour chacune des 3 issues. On peut voir que la moyenne des échantillons est très proche de la valeur attendue. Cependant, au lieu d'obtenir un seul chiffre, nous obtenons une fourchette d'incertitude comme l'indique le grand écart type et l'intervalle de probabilité à 95%.

```python
#Here we create our Dirichlet, using 1 for the alpha value

alphas = np.array([1,1,1])
c = np.array(move)

# Create model
with pm.Model() as model:
    # Parameters of the Multinomial are from a Dirichlet
    parameters = pm.Dirichlet('parameters', a=alphas, shape=3)
    # Observed data is from a Multinomial distribution
    observed_data = pm.Multinomial(
        'observed_data', n=sum(move), p=parameters, shape=3, observed=c)
    # Sample from the posterior
    trace = pm.sample(draws=1000, chains=2, tune=1000,
                      discard_tuned_samples=True)
```

Figure 4: Model creation

	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_mean	ess_sd	ess_bulk	ess_tail	r_hat
Up	0.459	0.132	0.241	0.722	0.004	0.003	1121.0	1121.0	1122.0	1366.0	1.0
Flat	0.156	0.097	0.007	0.321	0.003	0.002	814.0	814.0	728.0	728.0	1.0
Down	0.385	0.127	0.162	0.623	0.003	0.002	1673.0	1656.0	1657.0	1464.0	1.0

Figure 5: Model results

Si on visualise nos échantillons, on constate en effet qu'il y a une grande incertitude, même si une tendance semble se dégager. En effet, l'issue Up (en bleu) est un peu plus fréquente que l'issue Down (vert), et l'issue Flat (jaune) est plus loin derrière (*Image 6*).

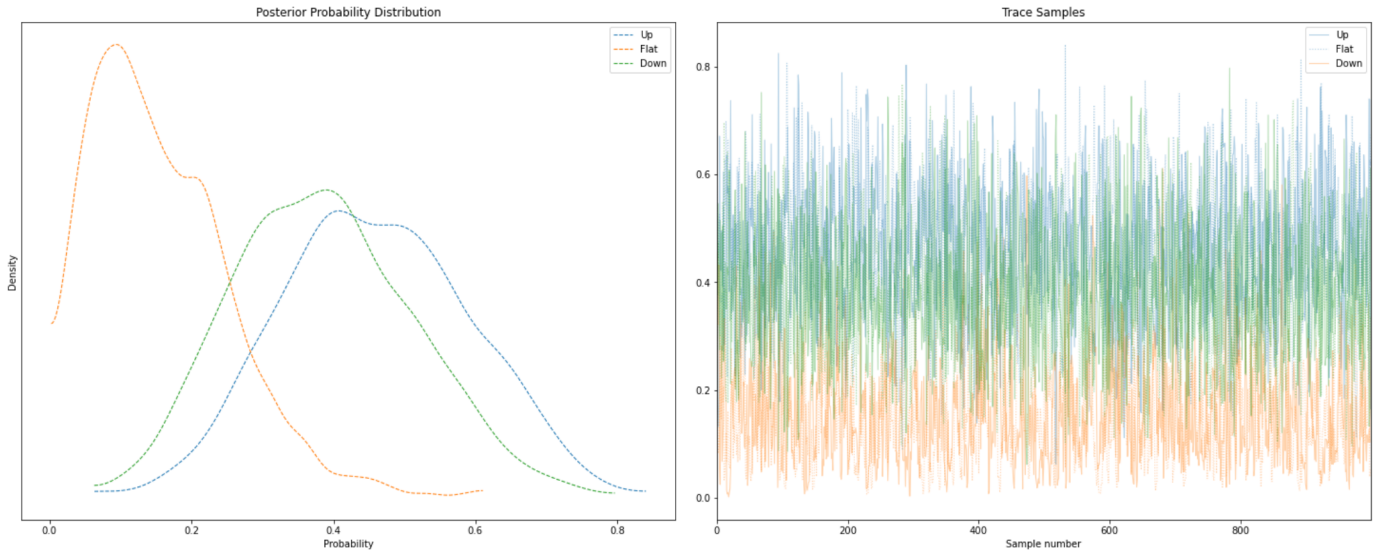

Figure 6: Visualizing samples

3.4 La prochaine observation

Maintenant que notre modèle est construit, nous pouvons enfin estimer les probabilités des 3 issues pour notre 12ème jour. Pour cela, nous utilisons l'a posteriori pour tirer des échantillons. Ici nous choisissons de simuler 10000 nouveaux jours. Nous obtenons alors les valeurs suivantes pour les probabilités de chaque issue :

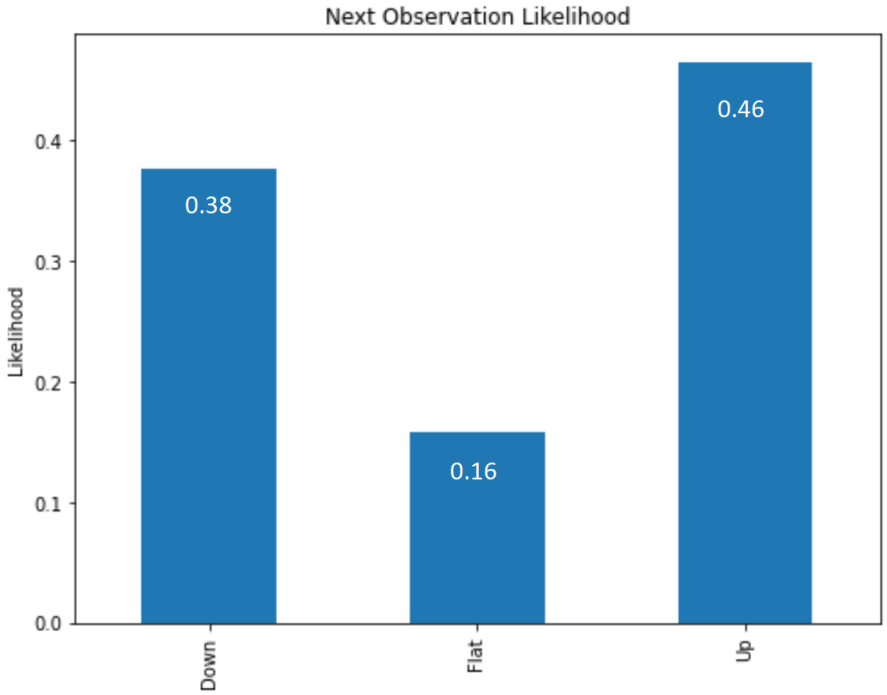

Figure 7: Next Observation Likelihood

4 Backtest et simulation

Maintenant que nous avons pu constuire un modèle afin d'estimer les intervalles pour chaque issue possible lors du 12ème jour, nous allons backtester ce modèle en appliquant aux données passées. Lors ce cette simulation, nous appliquons les exactes mêmes étapes afin de constuire 10 modèles, sur 10 périodes successives de 11 jours, toujours dans le but d'estimer les intervalles de probabilité de chaque issue lors du 12ème jour. Nous pourrons ainsi comparer ces estimations avec la réalité et challenger notre modèle. Nous utiliserons à nouveau les données du S&P 500, cette fois du 29 Décembre au 28 Janvier afin de constuire 10 plages de 12 séances. Nous pouvons voir les résultats de cette simulation sur l'image suivante.

	2021-01-14	2021-01-15	2021-01-19	2021-01-20	2021-01-21	2021-01-22	2021-01-25	2021-01-26	2021-01-27	2021-01-28
Up	0.6309	0.5291	0.4642	0.5358	0.5416	0.4651	0.3874	0.3894	0.3791	0.3784
Flat	0.1525	0.1535	0.1579	0.155	0.1527	0.2291	0.2243	0.2319	0.2296	0.1639
Down	0.2166	0.3174	0.3779	0.3092	0.3057	0.3058	0.3883	0.3787	0.3913	0.4577
Actual Outcome	Down	Down	Up	Up	Flat	Down	Up	Down	Down	Up

Figure 8: Issues likelihoods vs actual outcomes

Lorsque P(Up) est supérieur à 0.5, l'investisseur décide d'acheter l'indice (on parle de position longue) car il y a une probabilité assez élevée que l'indice monte selon les informations apportées. Au contraire, si P(Down) est supérieure à 0.5, alors l'investisseur reçoit un signal fort de baisse prévue le lendemain donc il va se porter "short" en vendant l'indice.
Ainsi, lorsque l'investisseur long (resp. short) l'indice, on multiplie son portefeuille par (1+variation) (resp. (1-variation)), variation étant la variation réelle de l'indice le jour concerné. On initialise le portefeuille à une valeur de 100 afin d'avoir une représentation intuitive du rendement de la stratégie. On capitalise, c'est-à-dire que l'on réinvestit l'entièreté du portefeuille sur la stratégie optimale du jour suivant (sauf si aucun outcome n'a de probabilité supérieure à 0.5 auquel cas il n'investit pas le jour suivant).

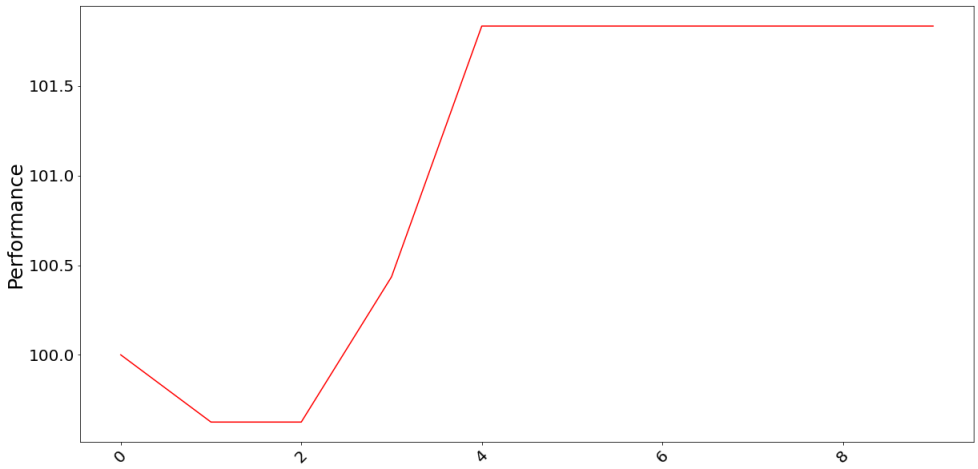

Figure 9: Performance of the strategy (100 at the 14^{th} of January)

5 Conclusion

L'interprétation du graphique ainsi que du tableau (*Image 8*) nous montrent plusieurs faits intéressants. Tout d'abord, les deux premiers jours, alors même que la probabilité de l'événement Up est importante (ce qui pousse l'investisseur à long l'indice), le marché est baissier. En revanche, les meilleures prédictions en milieu de période permettent d'améliorer la performance de la stratégie. Cependant, à partir du 22 janvier la probabilité ni de Up, ni de Down n'est suffisante pour investir puisqu'elles sont toutes deux sous le seuil de 0.5.
Toutefois, la stratégie permet d'obtenir une rentabilité de 1.83% sur une période de 10 jours en ne prenant le risque d'investir que sur 50% des séances possibles. De plus, cette période a été caractérisée par des mouvements extraordianaires d'achats massifs sur des *meme stocks* de la part des clients retail (GameStop, AMC,...). Ces mouvement ont eu un impact majeur sur la volatilité ainsi que les variations de marché. Nous pouvons donc conclure de la robustesse de notre approche face à une période unique.
Enfin, dans le but de diminuer l'aspect technique de notre approche et d'améliorer sa robustesse, il serait intéressant d'y ajouter une composante "avis d'expert" dans l'information a priori. Cela nous permettrait de développer une approche à la fois technique et fondamentaliste.