Options Trading involving Hidden Markov Model

Melis Guillaume

November 2024

"As I never learnt mathematics, so I have had to think."
Joan Robinson (1903–1983)

Contents

1 Introduction **4**

2 Finance Theory **5**
2.1 Option Theory . . . 5
2.1.1 Introduction to options . . . 5
2.1.2 Vanilla Options . . . 5
2.1.3 Exotic Options . . . 6
2.2 Valuation of an Option . . . 8
2.2.1 The Binomial Tree . . . 8
2.2.2 The Trinomial Tree . . . 12
2.2.3 The Black-Scholes Merton Model . . . 13
2.3 Basic Options Strategies . . . 15
2.3.1 Bull Spread . . . 15
2.3.2 Bear Spread . . . 17
2.3.3 Butterfly . . . 18
2.3.4 Straddles . . . 19
2.3.5 Strangles . . . 20

3 Mathematical Study **21**
3.1 Introduction to Markov Chain . . . 21
3.2 The Hidden Markov Model . . . 24
3.3 Computation of the Model . . . 25
3.3.1 Forward Algorithm . . . 25
3.3.2 Viterbi Algorithm . . . 27
3.3.3 Forward-Backward Algorithm . . . 29

4 Combining Options and HMM **32**
4.1 Defining the Strategy . . . 32
4.1.1 Observation Period and time series . . . 32
4.1.2 Extracting a trend with HMM . . . 33
4.1.3 Defining good option strategy regarding the trend . . . 34
4.1.4 More complex models . . . 34

5 Implementation of the Strategy **36**

6 Conclusion **37**

7 Bibliography **38**

8 Sources **39**

1 Introduction

This paper focus on a quantitative trading strategy involving options using Hidden Markov Model as an indicator of the market trend. This paper assume a basic knowledge in finance and derivatives, and a strong knowledge in mathematics as basic concepts will not be explained and assumed to be known. The main focus of this paper is to understand if using Hidden Markov Model will allow us to unveil a trend in the market that could not be seen with classic indicators and trying to take profit of this knowledge.

A quantitative strategy is based on mathematics and purely mathematics, it does not involve the trader feelings and the quantity to trade are computed using algorithm only. Quantitative Investment Strategies also known as QIS bring us back to the 20^{th} century and has been a long journey in wich mathematicians, physicists, statisticians, economists and quants have pursued a common objective : managing volatiliy, also known as the risk mesure in finance, and developping strategies that are maximizing the profit.
The developpment of Quant Finance followed the publication of Louis Bachelier Thesis : "*The Theory of speculation*" (1900), involving for the first time a Brownian motion to approximate asset prices' volatile path. A second breakthrough was the developpment of options pricing theory and the publication of the Fisher Black and Myron Scholes formula that formalized the way of valuing this class of derivatives in 1973. The original Black-Scholes Merton model had limitation, it mainly assume the no-arbitrage argument and the volatiliy as a constant. Despite this strong hypothesis, the Black-Scholes-Merton model is still mainly used in finance for valuing options as the stochasticity of the volatiliy is involving complex models such as the Heston Model (1993), which is way more difficult to calibrate and is not mainly used for the trading of vanilla options. As the market is based on the concensus of using the Black-Scholes formula, if you're using another one you are taking the risk of being arbitrated.
Today, the research is mainly focusing on developping a stochastic volatility model that fit perfectly the market conditions, following Heston, Bruno Dupire proposed a local volatility model to describe the volatiliy of European option while capturing the smile fitting market price with accuracy. The more recent stochastic model is due to Hagan (2002), which proposed the Stochastic Alpha Beta Rho, known as SABR to overcome a mismatch verified between the dynamic of the smile Dupire's model and actual market realisation.

The research on volatiliy is enhancing the fact that possessing the right information is crucial to price and developping trading strategies over assumptions that are verified by the market. As volatiliy is mainly used in portfolio management, I propose here another indicator to make the portfolio allocation, managing risk with trading only european call options which allows limiting the loss and maximizing profit regarding the market conditions.

2 Finance Theory

In this part we will introduce the option theory, history and uses of an option, Vanilla and exotic options. We will also introduce to different models to price an option and then introduce to option strategies that can be used to take profit of market conditions.

2.1 Option Theory

2.1.1 Introduction to options

The first options were used in ancient Greece to speculate on the olive harvest so we can date them from the mid fourth century B-C in a story related by Aristotle. Thales of Miletus had great interest in mathematics and astronomy and he combined both to predict the olive harvest in his region. Once he have made his prediction, he created the first known options contracts. He recognized that there would be a significant demand for olive presses and wanted to basically corner the market. But Thales didn't have sufficient funds to own all the olive presses, so instead he paid the owners a sum of money each in order to secure the rights tu use them at harvest time. When the harvest come, as predicted by Thales it was huge and Thales used his rights on the presses to realize an important profit. The term wasn't used at the time but Thales created the first call option.
We can see many other occurences referring to similar trading strategies to take profit of informations or a trend of the market. We can cite the Tulip Bulb Mania in the 17^th century on the Deutsch market. At this point of history call and put were already used in many different markets primarily for headging purpose. For example, the farmers would buy puts to protect his profits instead the prices on the market would go down, and tulip wholesalers would buy calls to protect against the price of tulip going down.

Definition Call Option: Right to buy at a certain time, for a certain price the underlying asset.

Definition Put Option: Right to sell at a certain time, for a certain price the underlying asset.

The one who issue the option is also referred as the *writer*. To acquire the buyer pays a *premium*. When a call is *exercised*, the holder pays the writer the *strike* price in exchange of the stock and the option cease to exist. When a put is exercised, the holder recerives the strike price from the writer in exchange for the stock and the option cease to exist.
Options are very interesting for investors because they give the *right* and not the obligation to buy or to sell. And as a right, it may not be used. By contrasts, in many other derivatives, the two parties have committed themselves to some action. This flexibility and the hability to be used for hedging purpose can explain the popularity of options trading. We consider two types of instrument in finance, simple products that we refer to as Vanilla and more complexe products that we refer to as Exotic. In the following sections, we will introduce this products and their uses.

2.1.2 Vanilla Options

Vanilla options are simple derivatives that are easy to price and to understand. First we introduce the European options.
European options are the simplest options. They are call or put, have a strike price and a maturity date. Contrary to American, Europan options cannot be exercised prior the maturity date. An American Option is worth at least as much as the European one because of the early exercise feature.

Considering an investor who believes that the underlying will take value. He buys a call option with a strike price K of 120\$ and a time to maturity of T=0.25 (in years). The actual level of the underlying is 100\$. Suppose that he pays the option 5\$. If the stock rises to 150\$ 3 months later, the investor exercise the call and make a profit of 25\$ by actions. In the opposite scenario, if the stock level fall to 80\$, the investor does not exercise the option and loss only 5\$ instead of 40\$.

In this extremely simple example, we have seen that an option can be used to make profit but also provide a security in case that the underlying level is falling. This is one important aspect of the derivative, the investor does not own the underlying. Considering the opposite scenario, we can easily understand how to use a put and how it could be useful to hedge our position aggainst the stock variations.

European options seems good but American looks better because of the early exercise possibility. In the market, most of the options traded are american, the early exercise come with a greater premium. I will explain how we value them in compare to the European in the dedicated part.

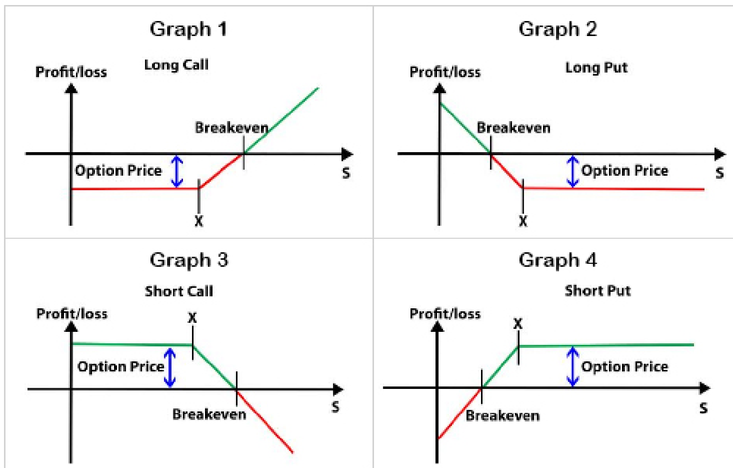


Figure 1: Call and Put Payoffs

Following this charts we can determine the payoff from vanilla Options. At the time to exercise, wheter it's an European or an American the payoff is :

- If a call : max(0, S_T - K)
- If a put : max(0, K - S_T)

More complex options are existing, as the needs of investors could be more specific. They are fine tuned products with specific features and much more complex formula called Exotic options and will be presented in the next section.

2.1.3 Exotic Options

Exotic Options are interesting products that are generally much more profitable than the Vanilla ones.They are products with no standard and well defined properties, sometimes meeting genuine hedging need in the market or sometimes used for tax, accounting, legal or even regulatory reasons by corporate treasurers, fund managers or financial institutions.

I will present some classic Exotic options and there features :

- Perpetual American options is a type of option contract that has no expiration date, allowing the holder to exercise the option at any time indefinitely. This contrasts with standard American options, which have a set expiration date. They are often studied in mathematical finance, especially in problems involving optimal stopping theory.

- Bermudan Option, an american options where the early exercise possibility may be restricted to certain dates.
- Nonstandard american option with a floating strike price.
- Nonstandard american with an initial lock-out period, this feature is regularly encountered on embedded options.
- Gap option, is an european call option that pays $S_T - K_1$ when $S_T > K_2$.
- Forward start option, is a contract that will be effective at some date in the future
- Cliquet option or ratchet option, is a series of call or put with given rules to determine the strike price.
- Compound options, simply options on options. There is four types of compound options : A call on a call, a call on a put, a put on a put and a put on a call. Compound options have two strike prices and two striking dates.
- Chooser option or "as you like option" is an option that after a specific period of time, the holder can choose if the option is a call or a put.
- Barrier Options, are options where the payoff depends on whether the underlying reaches a certain level during a period of time. We can extract two categories, knock-in and knock out. The knock-in feature bring the option alive contrary to the knock-out that whipe out the investor.
- Binary options, which are generally cash or nothing option. They pay a fix amount of money if a certain strike price is reached. In the other case, the investor win nothing.
- Lookback options, are options on which the payoff depend on the maximum or minimum asset price during the life of the option. The payoff for a floating lookback call is the amount that the final asset price exceed the minimum reached during the lifetime of the option.
- Asian option, are options in which the payoff depends on the arithmetic average of the price of the underlying asset during the life of the option.
- Exchange option, is an option to exchange an asset for another.
- Rainbow option, an option involving two or more risky underlying assets.

Most of the Exotic options involve conditional expectation on another asset price to exist. In some case, modifying the Black-Scholes formula introducing the mathematical translation of the special features can lead to a valuation formula. In other cases, and also during the option lifetime, the value of the option is calculated by Monte-Carlo simulation.

2.2 Valuation of an Option

Valuing Vanilla options can be made using multiples methods. In this section we will see how, by starting with an intuitive methods from statistics and slowly complexifying the model to the use of stochastic calculus and the Black-Scholes formula.
An important relation between the value of a call, c and the value of a put, p is the *put-call parity* :

$$c + Ke^{-rT} = p + S_0$$

This relationship is extremely important because it illustrate the no-arbitrage condition. If this equation does not hold, it exist the following opportunities. Let,

$$c_t - p_t > Ke^{-r(T-t)} + S_t$$

A t we buy a stock and a put and we short a call. We then make a profit of :

$$c_t - p_t - S_t$$

If this sum is positive, we invest it at the risk free rate until T or we borrow it at the same rate. Then, at T there is two possibilities :

- $S_T > K$: the call is exercised, we deliver the stock and end the borrowing. The following profit is made : $K + e^{r(T-t)}(C_t - P_t - S_t) > 0$
- $S_T \leq K$: the put is exercised, and we close the deal like above so the positive amount $K + e^{r(T-t)}(C_t - P_t - S_t)$ is made

Let see how to compute the following quantities c and p.

2.2.1 The Binomial Tree

The general approach to construct trees has been introduced in a paper published in 1979 by Cox, Ross and Rubinstein. Using a Binomial Tree to price options is a popular technique and not so complicated one. We construct the tree as a probabilistic path representing the possible level that the underlying could take. The underlying assumption is that the underlying asset follow a random walk.

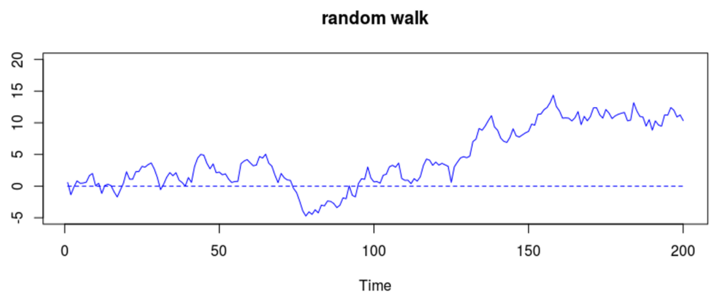
Definition A random walk is a mathematical model that describes a path consisting of a succession of random steps. It is widely used in fields like physics, economics, and finance to model random processes over time.
For a simple random walk in one dimension, the position at step n, denoted as X_n, is given by the sum of independent random variables:

$$X_n = X_0 + \sum_{i=1}^{n} \epsilon_i$$

Where :

- X_0 is the starting position
- ϵ_i are independant and identically distributed variables that represent the step size. It typically take values between -1 and 1 with for a simple symmetric random walk.

Let's introduce the concept by using a very simple case. We consider the underlying level is 25$ and it is known that at the end of the month, it will be 28$ or 22$. We want to value a call option, for a strike price of 26$ who has an expiration date at the end of the month. Assuming that at the end of the month the underlying is taking one of the two values above, we can say that the option value will be 2$ or 0$. In this



example, it is relatively simple to value the option, taking account of the no-arbitrage argument.
Now let's dive deeper into the theory. Consider a portfolio of a long position is Δ shares of the underlying stock and a short position in one call option. We want to compute the value of Δ that makes the portfolio riskless. If the stock is moving up until 28\$ the value of the shares is 28Δ and the value of the option is 2\$, so the total value of the portfolio is $28\Delta - 2$. Following this logical, if the value of the stock moves in the opposite direction, the value of the portfolio is 22Δ. The portfolio is riskless if the value of Δ is chosen so that the final value of the portfolio is the same independent to the path chosen by the underlying. It follows that :

$$28\Delta - 2 = 22\Delta$$
$$\Delta = 0.333$$

The riskless portfolio is therefore constituted of 0.333 shares and 1 option. Regardless of the stock movement, it's value will be the same. In absence of arbitrage opportunities, riskless portfolio must earn the risk-free rate.

Let's generalize this process. In the following we will consider :

- A stock, denoted S_0, as we consider the time of pricing t_0
- An option on the stock, denoted f
- Time to expiration T
- An Up level of the stock, $S_0 u$ where $u > 1$
- A down level of the stock $S_0 d$ where $d < 1$
- The payoff from the option if there is a up movement is denoted f_u
- The payoff from the option if there is a down movement is f_d

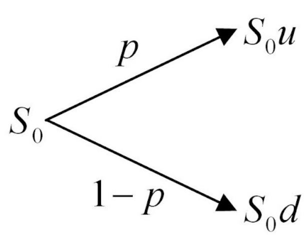


Figure 2: Binomial Tree of pricing

As before, we create a portfolio constituting of a long position in Δ shares and a short position in an option. We calculate the value of the delta that make the portfolio riskless and if there is an up movement in the underlying the value of the portfolio at the end of the life of the option is :

$$S_0u\Delta - f_u$$

If there is the opposite movement in the stock price, then the value of the portfolio become :

$$S_0d\Delta - f_d$$

The two are equal when :

$$S_0u\Delta - f_u = S_0d\Delta - f_d$$
$$\Delta = \frac{f_u - f_d}{S_0u - S_0d}$$

In this case, the portfolio is riskless and it mus earn the risk-free rate considering the no-arbitrage argument. If the risk free rate is r, the PV of the portfolio is :

$$(S_0u\Delta - f_u)e^{-rT}$$

As the cost of setting up the portfolio is

$$S_0\Delta - f$$

It follows :

$$S_0\Delta - f = (S_0u\Delta - f_u)e^{-rT}$$
$$f = S_0\Delta(1 - ue^{-rT} + f_ue^{-rT}$$

Substituting the Δ in the equation we now have :

$$f = S_0(\frac{f_u - f_d}{S_0u - S_0d})(1 - ue^{-rT} + f_ue^{-rT}$$

$$f = \frac{f_u(1 - de^{-rT}) + f_d(ue^{-rT} - 1)}{u - d}$$

$$f = e^{-rT}[pf_u + (1-p)f_d]$$

With

$$p = \frac{e^{rT} - d}{u - d}$$

Considering the *risk-neutral valuation* assumption. The parameter p should be interpreted as the probability of an up movement and $(1-p)$ is the probability of a down movement. We assume $u > e^{rT}$, so that $0 < p < 1$. The equation

$$pf_u + (1-p)f_d$$

is the expected future payoff from the option in a risk-neutral world.

Proof :

$$E[S_T] = pS_0u + (1-p)S_0d$$
$$E[S_T] = pS_0(u - d) + S_0d$$

Substituing p we compute above in the equation, we have :

$$E[S_T] = S_0e^{rT}$$

So the stock price grows on average at the risk-free rate when p is the probability of an up-movement.

We can generalize the concept of a 1 step binomial tree to a tree constituted of n nodes. The pricing of the option can be done by repeatedly applying the principles established above because a n step binomial tree can be reduced to a succession of 1 step binomial trees.
A generalization of the preceeding formula can be done. We now consider the time step to be Δt so the equation of f become :

$$f = e^{-r\Delta t}[pf_u + (1-p)f_d]$$

With

$$p = \frac{e^{r\Delta t} - d}{u - d}$$

A repeated application of the equation of f gives :

$$f_u = e^{-r\Delta t}[pf_{uu} + (1-p)f_d]$$
$$f_d = e^{-r\Delta t}[pf_{ud} + (1-p)f_{dd}]$$

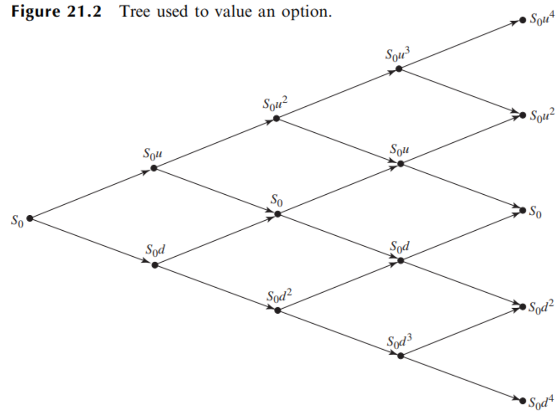
Now if we substitute f_d and f_u in the new equation of f we get :

$$f = e^{-2r\Delta t}[p^2 f_{uu} + 2p(1-p)f_{ud} + (1-p)^2 f_{dd}]$$

Which is consistent with the risk-neutral principle.
Variables p^2, $2p(1-p)$ and $(1-p)^2$ are the probabilities that the upper, middle and lower final nodes will be reached.

However, the pricing is slightly different for an American option because of the early exercise feature. In this case, the procedure is to work back trough the tree to see whether early exercise is optimal or not at each node. The value of the option at the final node is the same as for European option but at earlier nodes, the value of the option is greater than the value of f given by the equation above and the payoff from early exercise.
Working backward to a tree can be see as starting at the end of the tree (time T). The value of the option is known at T because for example, a *put* option is worth *max(*$K - S_T$*,0)* and a *call* is worth *max(*$S_T - K$*, 0)*, with S_T the asset price at time T. We denote K as the strike price.
Because the risk-neutral assumption, the value at each node at time $T - \Delta t$ can be calculated as the expected value at time T discounted at the rate factor r for the given time period Δt. Similarly, for the option value at time $T - 2\Delta t$, we take the expected value of the option at time $T - \Delta t$ discounted at r for the Δt period. Walking back through the tree gives us the value of the option at t_0. For American options, we check at each node if early exercise or holding it for a further Δt period of time is preferable.

Figure 21.2 Tree used to value an option.



2.2.2 The Trinomial Tree

A Trinomial Tree is a complexification of the Binomial Tree model generally used for pricing options in a model considering local volatiliy. In this model, we approximate a *diffusion* in continuous time of S by modelizing the underlying level with considering a Markov Chain $\hat{S}_t, i = 0, ..., N; t_i = t_0 + i\Delta t$ by using the convergence of Markov Chain to diffusion.

Theorem : Let σ and μ continuous and bounded process and let $(X_n^h)_{n\geq 1}^{h\geq 0}$ a Markov Chain family weakly consistent with X. So the interpolated process defined by

$$\hat{X}_t^h = X_{[t/h]}^h$$

converges weakly to a Markovian diffusion with a drift μ and a diffusion coefficient σ.

In a Binomial tree we've seen before, the risk-neutral transition probability p is set by the martingale condition : $pu + (1-p)d = 1 + r\Delta t$.
To take the variation of the volatility, we should change the parameters taking account of the position of the node in the tree. But it's too complicated to implement in the Binomial tree model. To do so, we have to use a Trinomial tree, define as following :

$$\hat{S}_{t_1+1} = \begin{cases} u\hat{S}_{t_i}, & p \\ (1+r\Delta t)\hat{S}_{t_i}, & 1-p-q \\ d\hat{S}_{t_i}, & q \end{cases}$$

For the process convergence, we impose :

$$E[\hat{S}_{t_i+1}|\hat{S}_{t_i}] = (1+r\Delta t)\hat{S}_{t_1} \text{ (risk-neutral tree)}$$

$$Var[\hat{S}_{t_i+1}|\hat{S}_{t_i}] = \hat{S}_{t_i}^2\sigma^2(t_i, \hat{S}_{t_1})\Delta t$$

Lets define the transition probabilities $p(t_i, \hat{S}_{t_i})$ and $q(t_i, \hat{S}_{t_1})$ needed to satisfy the equations above. A quick computation gives us :

$$p(t_i, \hat{S}_{t_i}) = \tfrac{\sigma(t_i,\hat{S}_{t_i})^2\Delta t}{(u-d)(u-1-r\Delta t)} \text{ and } p(t_i, \hat{S}_{t_i}) = \tfrac{\sigma(t_i,\hat{S}_{t_i})^2\Delta t}{(u-d)(1+r\Delta t-d)}$$

To be sure that the tree is recombinant, we must have $ud = (1+r\Delta t)^2$.

Defintion : A recombinant tree is a tree where the number of nodes in every branches is growing linearly regarding the time and not exponentially.

The tree is not allowing arbitrage possibilities only if all transition probabilities are positive. Regarding this condition, we must have $0 < p + q \leq 1$ in every node of the tree. This is always true only if Δt is small enough comparing to the space step.

Now we want to price backward an option price, using a trinomial tree. To price an european call, the following procedure can be used :

- At T all prices are given by $c(t_N, S) = max(S-K, 0)$
- At every time t_i $\forall i < N$, the prices in every nodes of the tree are calculated using the prices known at time t_{i+1}, using the risk-neutral evaluation.

Using this principles we can define the following formula for an European call :

$$c(t_i, S) = \tfrac{1}{1+r\Delta t}(pc(t_{i+1}, uS) + qc(t_{i+1}, dS) + (1-p-q)c(t_{i+1}, (1+r\Delta t)S))$$

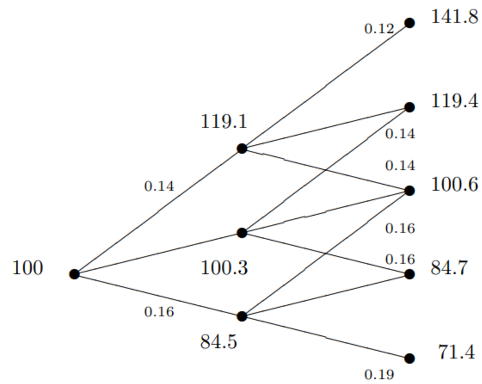


Figure 3: A Trinomial Tree example

Where d is equal to :

$$d = 1 + r\Delta t - \bar{\sigma}\sqrt{\Delta t}$$

Where $\bar{\sigma}$ is a constant that satisfies the condition $\bar{\sigma} > \sigma(t, S)$ in every node of the tree. If $\sigma(t, S)$ is not bounded, $\bar{\sigma}$ can be hard to find. To avoid this problem we can choose $\bar{\sigma}$ big enough and use the following model instead of the initial one :

$$\frac{dS_t}{S_t} = r_t dt + min(\bar{\sigma}, \sigma(t, S_t))dW_t$$

To conclude, the Trinomial tree is a powerful model to use in option pricing, providing flexibility and accuracy by capturing different possible price paths at each time step. Its ability to incorporate more price states (up, down, and no movement) makes it more stable and computationally efficient than the Binomial tree model, especially for complex derivatives and American-style options. Moreover, the Trinomial tree's simplicity in implementation allows it to be an accessible and practical tool for various option pricing scenarios.
However, while the Trinomial tree is highly effective for discrete time intervals, it is important to note that the Black-Scholes Model is another foundational approach, particularly well-suited for continuous-time pricing.

2.2.3 The Black-Scholes Merton Model

The Black Scholes Merton Model is the most used model to price options in finance today. Developped in 1973 by Fischer Black, Myron Scholes and Robert Merton, the model bring them the nobel prize of economy, confirming that their work was a major breakthrough.
The model is based on several assumption that I will list now :

- The returns of the underlyings are considered following a *lognormal* distribution
- The volatiliy is assumed to be constant, we generally take the historical volatiliy or the implied volatility
- The underlying is following the stochastic process $dS = \mu S dt + \sigma S dz$

The Black-Scholes-Merton differential stochastic equation gives :

$$\frac{\partial f}{\partial t} + rS\frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} = rf$$

Where r is the risk-free rate and σ is the volatility of the stock calculated using the *lognormal* returns.
I will not present here the demonstration to get the solution because it's well documented. Using stochastic calculus and Itô's Lemma we can compute two remarkable solutions from this equation :

$$c = S_0 N(d_1) - Ke^{-rT} N(d_2)$$
$$p = Ke^{-rT} N(-d_2) - S_0 N(-d1)$$

These are the equations to compute the value of a call and the value of a put. With the following notation :

- S_0 is the price of the stock at the time of computation
- $N()$ is the cumulative probability distribution for the normal distribution
- K is the strike price
- r is the risk-free rate
- T is the time to maturity in years computed as $\frac{\text{Number of trading days until maturity}}{252}$

We now explicit d_1 and d_2:

$$d1 = \frac{ln(S_0/K) + (r + \sigma^2/2)T}{\sigma\sqrt{T}}$$

$$d2 = \frac{ln(S_0/K) + (r - \sigma^2/2)T}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T}$$

We can interpret $N(d_2)$ as the probability of that a call option will be exercised in a risk-neutral world. $N(d_1)$ can be more complex to understand but, denoting $S_0 N(d_1) e^{rT}$ as the expected stock price at time T when stock prices less than the strike price are counted as zero.
If we rearange the equation of a call as :

$$c = e^{-rT} N(d_2)[S_0 e^{rT} N(d_1)/N(d_2) - K]$$

We can explicit $e^{rT} N(d_1)/N(d_2)$ as the expected percentage increase in stock price considering the risk-neutral assumption is option is exercised.

We've seen tree main way to compute an option price. They seems to differ at first but leads to a relative homogeneity of the results given. While trees suppose the underlying as a random walk, the Black-Scholes equation is using stochastic calculus, introducing a Wiener process in the modelization of the underlying to extend the model in continuous time. From classical options, we can make some useful combination to created relevant trading strategies regarding the market. Let's introduce them.

2.3 Basic Options Strategies

After we've seen how to value an option, now let's take a look at how to use them. It exists many options strategies, the main ones will be presented with their payoff-chart and the market conditions that make them profitable.

2.3.1 Bull Spread

A *Bull Spread* is part of the family of **Vertical Spreads**. They are the most popular vertical spread strategies for investors who have a *bullish view* of the market. The bull spread is a combination of 2 options. The first one is sold while the second is purchased by the investor.
Let's denote K_1 the strike of the first option and K_2 the strike of the second, with $K_1 < K_2$. If we create our bull spread with call options, whe short the option with strike K_2 and buy the option with strike K_1. Here's the payoff chart of a *bull call spread*. When an investor buy this type of spread, he belives that the

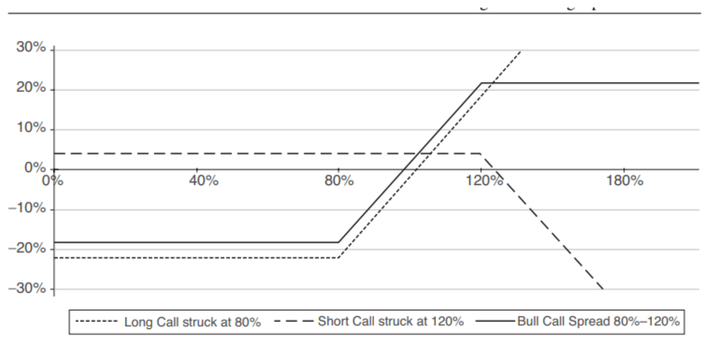


Figure 4: A 80-120 Bull Call Spread Payoff Chart

underlying will increase above the first strike K_1 but will not be able to increase over K_2.
Let's the payoff of this strategy being :

$$BullCallSpreadPayoff = \begin{cases} 0, & if S_T < K_1 \\ S_T - K_1, & if K_1 < S_T < K_2 \\ K_2 - K_1, & if S_T \geq K_2 \end{cases}$$

We can also create the following spread using only calls. To do so, we short the put with strike K_2 and we long the put with strike K_1. Here's the payoff chart :
The payoff is given by :

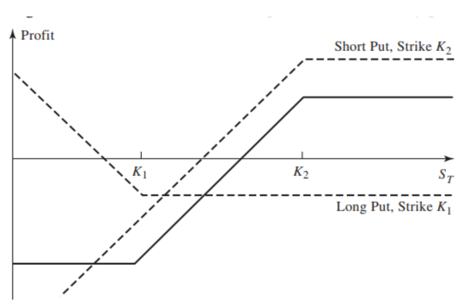


Figure 5: Bull Put Spread Payoff Chart

$$BullPutSpreadPayoff = \begin{cases} K_2 - K_1, & if S_T \leq K_2 \\ S_T - K_1, & if K_1 < S_T < K_2 \\ 0, & if S_T \geq K_1 \end{cases}$$

Tree types of bull spreads can be distinguished :

1 Both options are initially out of the money

2 One options is in the money while the second is out of the money

3 Both options are initially in the money

The most aggressive strategy is the first one because they do not cost a lot of money to set up and have a relative small probability of giving high payoff. As we move to type 2 and 3, the strategy become more conservative.

Hedging Considerations : It's crucial to understand these strategies because they are used as components for more sophisticated structured products such as capped cliquet or other payoffs involving digits. Bull call and puts involve the same risks. Here are two interesting figure to know : As we can see, the

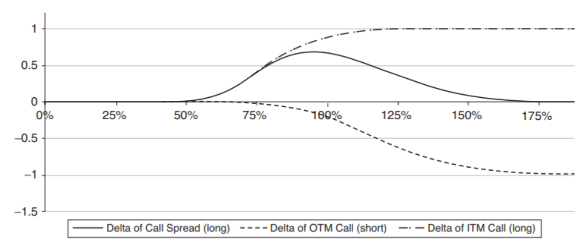


Figure 6: Variation of Delta for a Call Spread

Delta of the call spread remind us the Gamma of vanilla options. Wa can also note that the Delta is always positive, conforting our assumption that the holder is always long the stock price. As we can see on this

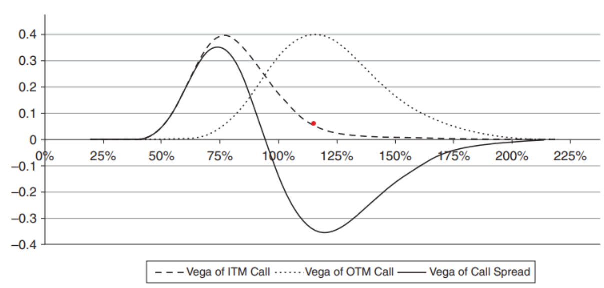


Figure 7: Variation of Vega for a Call Spread

figure, Vega is more difficult to manage as the sign is changing on the basis movement in the underlying price. We can also denote that the Vega is the difference between the two Vegas as one is long the smallest strike and short the largest one. In general case, the Vega changes sign arround the forward, which can be several percent away for the 100% ATM point.

2.3.2 Bear Spread

A *Bear Spread* are also **Vertical spread** strategies that have a similar payoff mechanism of bull spreads but correspond to a *bearish* view of the market. Bear spreads can be created by using put options or call options. In this strategy, the strike of the option purchased is greater than the strike of the option sold. Let's denote as before the strikes K_1, K_2 with $K_1 < K_2$. Here are the payoff charts of the strategy : The payoff of the

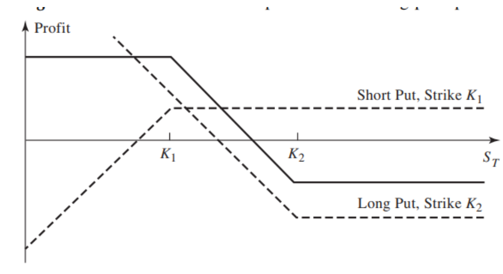


Figure 8: Bear Put Spread Payoff Chart

strategy which consist of buying a put with strike K_2 and selling a put which strike K_1 is given by :

$$BearPutSpreadPayoff = \begin{cases} K_2 - K_1, & if S_T \leq K_1 \\ K_2 - S_T, & if K_1 < S_T < K_2 \\ 0, & if S_T \geq K_1 \end{cases}$$

Considering the same strategy consisting of calls, we now short the call with strike K_1 and buy the call with strike K_2. As the payoff is given by :

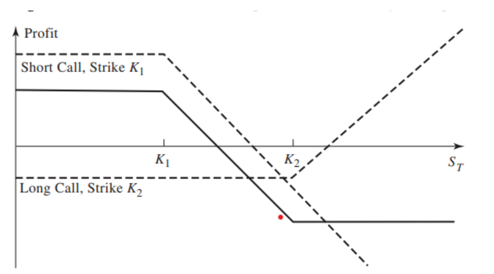


Figure 9: Bear Call Spread Payoff Chart

$$BearCallSpreadPayoff = \begin{cases} 0, & if S_T \leq K_1 \\ K_1 - S_T, & if K_1 < S_T < K_2 \\ K_1 - K_2, & if S_T \geq K_1 \end{cases}$$

A bear spread strategy limits not only the upside risk but also the downside since the maximum payoff that can be received by the holder is equal to $K_2 - K_1$. The profit pattern of a bear spread is given by deducting the price of the bought option from the payoff of the sold option and then offsetting when the options arrive in the money.
It is important to note that the payoff of a bear call spread is always negative whereas the payoff of a bear

put spread is always positive. This is due to the fact that an investor implementing a bear call is in fact selling a financial product and receives the global premium equal to *Call*(K_1, *T*) - *Call*(K_2, *T*). This amount represents the maximum profit he could get and it occurs if the underlying stock price finishes below K_1. Otherwise, the investor holding a bearish call spread can start loosing money, which is why the payoff is negative.

Hedging Considerations : As for bull spreads, we will consider the simple case where we enter a Put Bear Spread. First let's analyse the Delta :

$$BearPutSpread = Put(K_2, T) - Put(K_1, T)$$

So if we take the first derivative with respect to S we get :

$$\Delta_{BearPut} = \Delta_{Put(K_2,T)} - \Delta_{Put(K_1,T)}$$

And since $\Delta_{Put(K,T)} = \Delta_{Call(K,T)-1}$, then :

$$\Delta_{BearPut} = \Delta_{Call(K_2,T)} - \Delta_{Call(K_1,T)} = -\Delta_{BullCallSpread}$$

As we've done with the Delta of the Bull spread before, we can see here that the Delta of a Bear Spread is always negative, which proves that holding a bear spread strategy is always short the stock price.
Similarly, after deriving the bear put with respect to σ we get :

$$Vega_{BearPut} = Vega_{Put(K_2,T)} - Vega_{Put(K_1,T)}$$

And because $Vega_{Put(K,T)} = Vega_{Call(K,T)}$ we get :

$$Vega_{BearPut} = Vega_{Call(K_2,T)} - Vega_{Call(K_1,T)} = -Vega_{BullCallSpread}$$

The value of Vega of Bear put is the opposite value of Vega of Bull call with equivalent strikes. Traders needs to be cautious when managing the Vega of call spreads because they change signe as the undelrying moves.

2.3.3 Butterfly

The *Butterfly Spread* is a spread considered to be a neutral option strategy. It's a combination of a bull spread and a bear spread with the same maturity. It involve position in options with tree different strike prices. It offer a limited profit but also a limited amount of risk. To do so, we long two call options with strikes K_1 and K_3 and short two calls with strike price K_2, where $K_1 < K_2 < K_3$. Generally K_2 is relatively close to the stock price. The pattern of profit is as it follow :

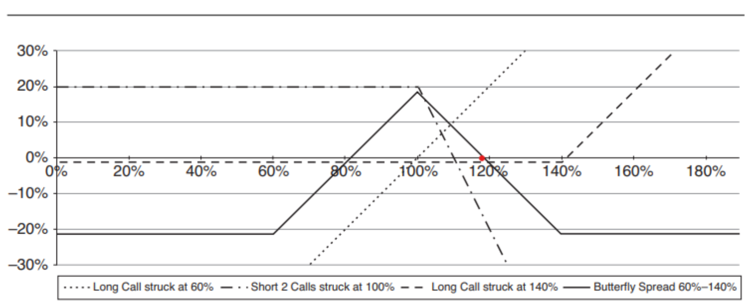


Figure 10: Payoff Chart from a Butterfly spread

The payoff using calls is as follows :

$$
ButterflySpreadPayoff = \begin{cases} 0, & if S_T \leq K_1 \\ S_T - K_1, & if K_1 \leq S_T \leq K_2 \\ K_3 - S_T, & if K_2 \leq S_T \leq K_3 \\ 0, & if S_T \geq K_3 \end{cases}
$$

The investor buying this strategy receives a positive payoff and pays the initial investment equal to :

$$
ButterflyPrenium = c(K-\epsilon) - 2c(K) + c(K+\epsilon)
$$

Where ϵ is the maximum payoff and occurs when $S_T = K$. In our exemple $K = K_2$ which we define as $K_2 = (K_1 + K_3)/2$.
We can also create a butterfly spread using puts, the investor buys two European puts, one with a lower strike and one with a high strike price, and sell two puts with an intermediate one. The use of put otpions results in exactly the same spread as the use of call options. Put-Call parity can be use to show that the initial investment is the same in both cases.

2.3.4 Straddles

The *straddle* is one of the most common combination and consists of a long position in a call and a long position in a put on the same underlying, with the same strike K and maturity T. If the underlying price is close to the strike price, the *straddle* leads to a loss, however if there is a sufficiently large move in either direction, a significant profit will result.
Her's the payoff chart of the strategy :

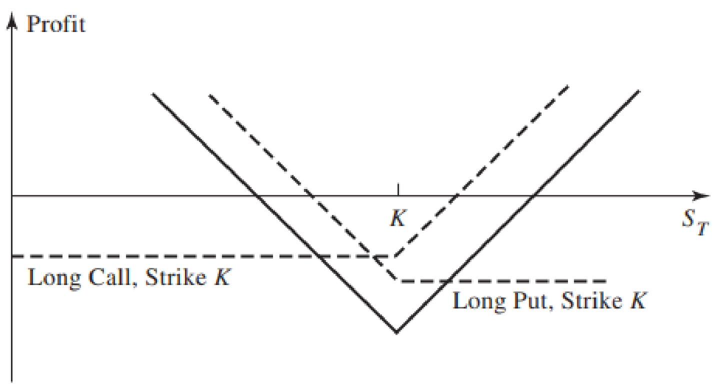


Figure 11: Payoff Chart from a Straddle

A straddle is appropriate when an investor is expecting a large move in stock price but does not know the direction. Holding a straddle is characterized by an unlimited profit potential and a maximum loss limited to the net initial premium paid to establish the position. The premium is equal to :

$$
Straddle_{Premium} = c(K,T) + P(K,T)
$$

The payoff of the strategy is therefore :

$$
StraddlePayoff = \begin{cases} K - S_T, & if S_T \leq K \\ S_T - K, & if S_T > K \end{cases}
$$

The seller of a straddle gets an initial premium to bear the risk linked to a large move in the stock price. Indeed, the buyer expects implied volatility to decrease. For the seller, the potential loss is unlimited and comes from the fact that a call is shorted, which is a dangerous position if not hedged.
Note that a straddle is very sensitive to volatility, Gamma and Vega of a straddle are positive and two times higher than Gamma and Vega of a call. The holder of the strategy is long volatility since this parameter increase the value of his product. At the money, the Delta of a straddle is not exactly zero but close.

2.3.5 Strangles

To conclude on options strategies, I will talk about *strangles*. Sometimes referred to as *bottom vertical combination*, it's a combination of a call and a put with same expiration date but different strikes prices. The strikes are K_1 and K_2, with $K_1 < K_2$. We long a call with strike K_2 and long a put with strike K_1. A strangle is a similar strategy to a straddle, the investor expect a large move in the stock price but is uncertain about the way it will move. The profit pattern depend on how close together are the strikes prices. Farther they're appart, less is the downside risk and more the stock price has to move to realize profit.

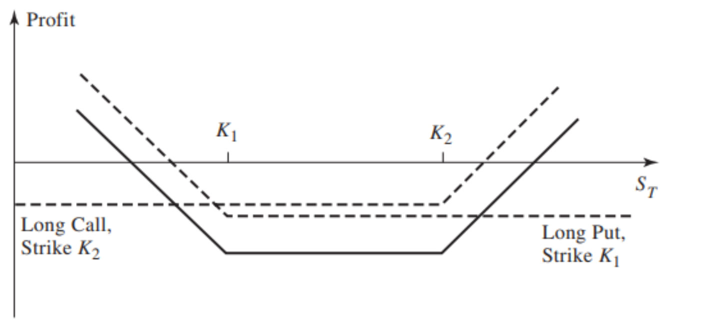


Figure 12: Payoff Chart from a Strangle

Payoff from the strangle are :

$$StranglePayoff = \begin{cases} K_1 - S_T, & if S_T \leq K_1 \\ 0, & if K_1 \leq S_T \leq K_2 \\ S_T - K_2, & if S_T > K_2 \end{cases}$$

Strangles enable investors to trade in volatility. A strangle is less sensitive to volatility than a straddle, it result that the Vega of out-of-the-money options is lower than the Vega of in-the-money options. As the Vega of a strangle is the sum of the Vega of options from the strategy, it's lower than the Vega of a straddle.

Conclusion : Combining options could result in many different stratgies. Some tend more to hedge the investor from risk while other are very aggressive and bet stronger on market moves. However, trading options, enable an investor to maximize his profit while managing is loss.

3 Mathematical Study

In this part I will present a mathematical study Markov Chains and their properties. As it is the indicator that I will use in the strategy, it's important to understand how it is working and how we can detect signals from a model that is an augmented model of the classic one. Here is a introduction to mathematics behind Markov Chain and a derivatives of the model, the Hidden Markov Model.

3.1 Introduction to Markov Chain

A Markov Chain is a model that tells us something about the probabilities of sequences of random variable states, each of which can take values from a set. That set can be words, tags, or symbols representing anything. The most common example is the weather. It is named after the Russian mathematician Andrey Markov, who introduced the concept in the early 20th century.

Let's take a particle who moves on a denumerable set E. At time n, the particle is in position $X_n = i$, it will be at time $n+1$ in position $X_{n+1} = j$ chosen independantly of the past trajectory X_{n-1}, X_{n-2} with probability p_{ij}.
Recall that sequence $(X_n)_{n\geq 0}$ of a random variables with value in the set E a *discrete-time stochastic process* with state space E.

Definition : If for all integers $n \geq 0$ and all states $i_0, i_1, i_2, ..., i_{n-1}, i, j$

$$P(X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, ..., X_0 = i_0) = P(X_{n+1} = j | X_n = i)$$

This stochastic process is called a *Markov Chain*.

This means that a Markov Chain is a no-memory process. The transition probability to go from i to j is conditioned only by the current state and not by the path taken. It's the **Markov Property**. We define an *Homogeneous* Markov Chain as a Markov chain where the transition for moving from one state to another remain the same regarding n.

A Markov Chain is also described by its *transition marix* that we define mathematically as :
The matrix $P = (p_{ij})_{i,j\in E}$ where

$$p_{ij} = P(X_{n+1} = j | X_n = i)$$

The entries are probabilities and represent transition from any state i to another state in E. Because it's probabilities it follows that,for all states i, j :

$$p_{ij} \geq 0$$
$$\text{and, } \textstyle\sum_{k\in E} p_{ik} = 1$$

Example : We consider the prediction of the temperature. As Hot is sate 1, Cold is stated 2 and warm is state three. Let's define the transition matrix as

$$P_a = \begin{pmatrix} 0.6 & 0.1 & 0.3 \\ 0.1 & 0.8 & 0.1 \\ 0.3 & 0.1 & 0.6 \end{pmatrix}$$

Assuming we know the starting distribution as $\pi = [.1, .7, .2]$. The initial distribution is determined from the Bayes sequential rule and in view of the homogeneous Markov property.
We can take another example where an LLM model is trying to predict the following word in your sentence. We can set its transition matrix as follow :

$$P_b = \begin{pmatrix} 0.1 & 0.5 & 0.4 \\ 0.5 & 0.3 & 0.2 \\ 0.6 & 0.2 & 0.2 \end{pmatrix}$$

These matrix can be represented as a graph.
Definition : A graph is a mathematical structure comprising a set of nodes N connected by egdes E. It usually represented as a diagram consisting of points representing the nodes connected by lines representing the edges.
Here the graph representation of the matrixs above :

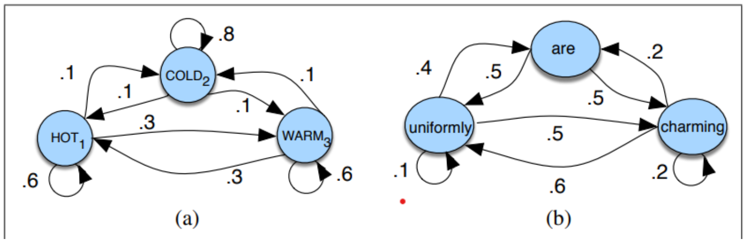


Figure 13: Markov Chain Graph representation, knowing $\pi = [.1, .7, .2]$

We say here that the graph is oriented, as the edges are represented by arrows to determine the way the transition probability occur in the Markov chain.

Let's define communication and periods that are important features of Markov Chains. These are topological in the sense that they concern only the naked transition graph.

Communication

Definition : We define an accessible state j if there exist $M \geq 0$ such that $p_{ij}(M) > 0$. These two states are said to communicate if i is accessible from j and j is accessible from i. We denote this relation as $i \leftrightarrow j$.

Definition : We can define a state as closed if $p_{ii} = 1$. More generally a set C of states such that for all $i \in C, \sum_{j \in C} p_{ij} = 1$ is called close.

Definition : If there exist only one communication class, then the chain, the transition matrix and its transition graph are said to be irreducible.

Period
Considering the random walk on $\mathbb{Z}$, since $0 < p < 1$ it is irreducible. Observing that $E = S_0 + S_1$ where they are the set or even and odd relative integers respectively and have the following property. If you start from $i \in C_0$, then in one step you can go only to a state $j \in C_1$. The chain passes alternately from cyclic class to the other, in this sense, the chain is said to have periodic behavior.

Definition : For any irreducible Markov Chain, one can find a unique partition and E into d classes $C_0, C_1, ..., C_{d-1}$ such that for all $k, i \in C_k$:

$$\sum_{j \in C_{k+1}} p_{ij} = 1$$

where by convention $C_d = C_0$ and d is maximal such $C_0^{"}, C_1^{"}, ..., C_{d-1}^{"}$ where $d^{'}$ is not existing. The number $d \geq 1$ is called the period of the chain. The classes $C_0, C_1, ..., C_{d-1}$ are called the cyclic classes. The chain moves from one class to the other at each transition and this cyclically.

Formally, this definition is bases on the **greatest common divisor (GCD)** of a set of positive integers.

Definition : The period d_i of state $i \in E$ is, by definition

$$d_i = GCD(n \geq 1; p_{ii}(n) > 0)$$

with the convention $d_i = +\infty$ if there is no $n \geq 1$ with $p_{ii}(n) > 0$. If $d_i = 1$, the state i is called aperiodic. For two sates i and j, if they communicate, they have the same period.

In conclusion, Markov chains provide a robust mathematical framework for modeling stochastic processes where future states depend solely on the present state, independent of the past. Their versatility and simplicity make them valuable for a wide range of applications, from statistical mechanics to finance and biology. An important aspect of Markov chains is their reducibility, which determines whether the chain can be broken into smaller, independent subchains. Additionally, their periodicity plays a critical role in understanding the behavior of the system over time, influencing how often states recur. In the following, we will discover an augmented model of classical Markov Chain that we will use in the strategy to unveil patterns in the underlying stock.

3.2 The Hidden Markov Model

A Markov Chain is useful when we want to compute the probability of a sequence of observable elements. However, in many cases, the events we are interested in are not observable directly, we say they are *hidden*. Going back to our text example, generally we do not observe part-of-speech tags in a text, we see words and infer the tags from our observation. In this example, we call our tags *hidden* because we don't obervse them directly. *Hidden Markov Model* allows us to pay attention to both, observed events and hidden ones. We think of this events as causal factor of our probabilistic model. Let's give a formal definition :

Definition : A Hidden Markov Model is a tool for representing probability distribution over sequence of observation. In this model, an observation X_t at time t is produced by a stochastic process but the state Z_t cannot be directly observed. This Hidden process is asumed to satisfy the Markov property. An Hidden Markov Model (reffered as HMM) is defined by the followings :

- A set of N states $Z_0, Z_1, Z_2, ..., Z_N$
- A transition probability matrix denoted $\mathbf{P}$ with each p_{ij} representing the transition probability of moving from state Z_i to state Z_j.
- A sequence of *observation likelihoods*, $B = b_i(X_t)$ also called *emission probabilities*, each expressing the probability of an observation X_t, being generated from a state i_i
- $\pi = [\pi_1, \pi_2, ..., \pi_N]$ the initial probability distribution over states π_i, the probability that the Markov Chain will start in state i

This is in fact called the first-order HMM, here's a graph representation of the model : Even we talk about

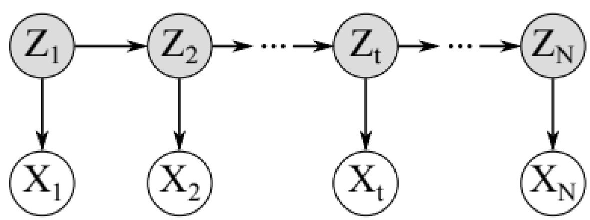


Figure 14: A Bayesian Network representing a first Order HMM

"time" to indicate that observations occur at a discrete "time steps", time could also refer to locations within a sequence.
The joint distribution of a sequence of states and observations for the first order HMM can be :

$$P(Z_{1:N}, X_{1:N}) = P(Z_1)P(X_1|Z_1)\prod_{t=2}^{N} P(Z_t|Z_{t-1})P(X_t|Z_t)$$

Where $Z_{1:N} = Z_1, Z_2, ..., Z_N$, idem for $X_{1:N}$.
At the first order, the probability of an output observation X_i depends only on the state that produced the observation Z_i and not on any other states. This is called the *Output independance*, where :

$$P(X_i|Z_1, Z_2, ..., Z_T, X_1, X_2, ..., X_T) = P(X_i|Z_i)$$

Let's introduce an example to understand how it's working. A task invented by Jason Eisner gives a good one. Imagine you're a climatologist in 2799 studying the history of global warming. You cannot find records of the weather in Baltimore, Maryland (US) for the summer 2020 but you find a diary which list how many ice creams a seller sold every day during this period. The goal is to use these observations to estimate the temperature every day. Assuming we categorize the days at cold of hot, the Eisner task is the following :
Given a sequence of observations X, find the hidden sequence Z of weather states (Hot or Cold) which

caused the vendor to sell ice creams.

In 1980, Rabiner used a tutorial of Jack Ferguson to introduce the idea that HMM should be characterized by *three fundamentals probems* :

- **Likelihood** : Given a HMM denoted $\lambda = (A, B)$ and an observation sequence X, determine the likelihood $P(0|\lambda)$.
- **Decoding** : Given an observation sequence X and an HMM λ as before, discover the best hidden state sequence Z.
- **Learning** : Given an observation sequence X and the set of states in the HMM, learn the HMM parameters A and B.

In the next section I will introduce algorithms used to solve this problems.

3.3 Computation of the Model

3.3.1 Forward Algorithm

The Forward Algorithm is used to compute the solution to the first problem we evoked above. For example, in the ice-cream selling problem, what is the probability of the sequence (3,1,3) where the two hidden states are H and C and the observations X = (1,2,3) correspond to the number of ice-creams sold the vendor, given a day : For a Markov Chain, where the surface observations are the same as the hidden events, we

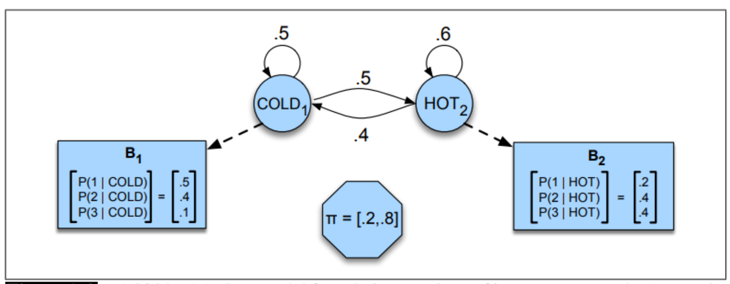


Figure 15: A graph representation of the HMM for relating number of ice creams sold to the weather

can compute the probability of the sequence (3,1,3) just by following the states with the same label and mulitpliying the probabilities along the arcs. But for an HMM, the things are not simple as this because we don't know what the hidden states are. I'll take a simply situation to start.

Suppose that we know the weather and we want to predict the how much ice creams will be sold. For a given sequence, we can easily compute the output likelihood of (3,1,3) with HMM. First, for HMM each hidden state produces only one observation and the sequence of hidden states and observations have the sale length. Given this mapping and the Markov assumption, for a observation sequence $X = X_1, X_2, X_3, ..., X_N$ and an hidden sequence $Z = Z_1, Z_2, Z_3, ..., Z_N$, the Likelihood of the observation sequence is :

$$P(X|Z) = \prod_{i=1}^{T} P(X_i|Z_i)$$

In our exemple, for one possible hidden sequence as *hot, cold, hod* the equation above gives us :

$$P(3, 1, 3|hot, cold, hot) = P(3|hot) \times P(1|cold) \times P(3|hot)$$

But because there are hidden states, we don't actually know what the hidden sequence was. Instead, we need to compute the probability of the sequence of events by summing over all possible hidden sequences weighted by their probability. To do so, we start by computing the joint probability of being in a particular hidden sequence Z and generating a particular observable sequence of events X. In general this given by the following equation :

$$P(X, Z) = P(X|Z) \times P(Z) = \prod_{i=1}^{T} P(X_i|Z_i) \times \prod_{i=1}^{T} P(Z_i|Z_{i-1})$$

In our particular case, it gives us

$$P((3, 1, 3), (hot, cold, hot)) = P(hot|start) \times P(cold|hot) \times P(hot|cold) \times P(3|hot) \times P(1|cold) \times P(3|hot)$$

To compute the total probability of the observations from the joint probability with a particular hidden state sequence, we just need to sum over all the possible hidden state sequences :

$$P(X) = \sum_{Z} P(X, Z) = \sum_{Z} P(X|Z)P(Z)$$

For an HMM of N hidden states and observation sequence of T observations, there are N^T possible hidden sequences. We can see that we can easily have a very large number so we cannot compute the total observation likelihood using this method.

Instead, we will use an efficient algorithm, with complexity $O(N^2T)$, called the **Forward Algorithm**. This algorithm is a kind of *dynamic programming* algorithm that use a table to store intermediate values as it builds up the probability of observation sequence. It then computes the observation probability by summing over the probabilities of all possible hidden state path that could generate the observation sequence. It does it efficiently by implicitely folding each of these path into a *forward trellis*.

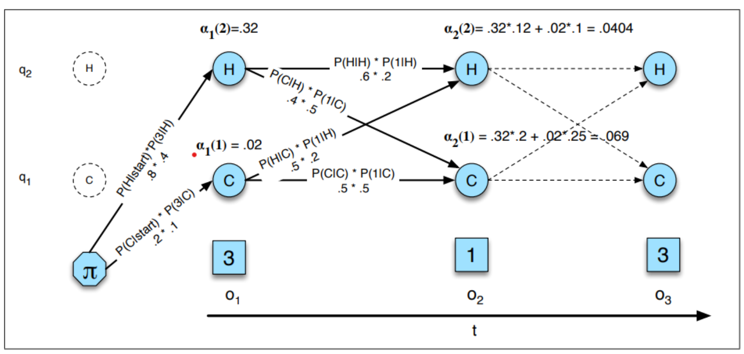


Figure 16: Forward trellis for computing the total observation likelihood in our example

Each cell $\alpha_t(j)$ is the probability of being in this state after seing the first t observations given the automaton λ. Each $\alpha_t(j)$ value is computed by summing all the probabilities of every path that could lead to it. Mathematically it is :

$$\alpha_t(j) = P(X_1, X_2, ..., X_t, Z_t = j|\lambda)$$

Where $q_t = j$ means the "t^{th} state in the sequence of states is state j".
We compute $\alpha_t(j)$ as follow :

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} bj(X_t)$$

Where the tree variables are :

- $\alpha_{t-1}(i)$ the previous *forward path probability* from the previous time step
- a_{ij} the transition probability from previous state Z_i to Z_j
- $b_j(X_t)$ the state observation likelihood of the observation symbol X_t given the current state j

function FORWARD(*observations* of len T, *state-graph* of len N) **returns** *forward-prob*

create a probability matrix *forward[N,T]*
for each state s **from** 1 **to** N **do** ; initialization step
$forward[s,1] \leftarrow \pi_s * b_s(o_1)$
for each time step t **from** 2 **to** T **do** ; recursion step
for each state s **from** 1 **to** N **do**

$$forward[s,t] \leftarrow \sum_{s'=1}^{N} forward[s',t-1] * a_{s',s} * b_s(o_t)$$

$$forwardprob \leftarrow \sum_{s=1}^{N} forward[s,T] \quad \text{; termination step}$$

return *forwardprob*

Figure 17: The Forward Algorithm, $forward(s,t)$ represents $\alpha_t(s)$

Given the pseudocode,

With :

1 Initialisation

$$\alpha_1(j) = \pi_j b_j(X_1) \text{ with } 1 \leq j \leq N$$

2 Recursion

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} bj(X_t) \text{ with } 1 \leq j \leq N, 1 \leq t \leq T$$

3 Termination

$$P(X|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

3.3.2 Viterbi Algorithm

In order to solve the second problem, we will use the *Viterbi Algorithm.* This algorithm computes the shortest path through the trellis diagram by using a kind of *dynamic programming* that we can find in the forward algorithm presented above. Viterbi is strongly looking like the minimum distance algorithm that are frequently used when working on a graph to find the optima sequence of choice to reach a certain state or lower bound the time needed to reach a given state.
It also has a forward and backward pass. In the forward pass, instead of the sum-product algorithm, we use the max-product algorithm.

Definition : The Max Product Algorithm is a dynamic programming approach used to solve the problem of finding the maximum product subarray within a given array of integers. The algorithm is designed to handle both positive and negative integers efficiently, considering that the presence of negative numbers can reverse the product's sign, potentially leading to a higher overall product when two negative numbers are multiplied.

The backward pass recovers the most probable path through the trellis diagram using a traceback procedure, propagating the most likely state at time t back in time to recursively find the most likely *sequence* between times $1:t$. We express it as :

$$\delta_t(j) \triangleq \underset{Z_1,\ldots,Z_t}{max} P(Z_{1:t-1}, Z_t = j | X_{1:t})$$

It can be expressed as a combination of the transition from the previous state i at time $t-1$ and the most probable path leading to i :

$$\delta_t(j) \triangleq \underset{1\leq i\leq K}{max}\,\delta_{t-1}(i)p_{ij}b_{X_t}(j)$$

where $\pi_{X_t}(j)$ is the emission probability of observation X_t given state j. We need to keep track on the **most likely previous state** i,

$$a_t(j) = \underset{i}{argmax}\delta_{t-1}(i)p_{ij}b_{X_t}(j)$$

We define the initial probability as

$$\delta_1(j) = \pi_j b_{X_1}(j)$$

The most probable final state is computed as

$$Z_N^* = \underset{i}{argmax}\delta_N(i)$$

To conclude, the most probable sequence can be computed using trace-back as

$$Z_t^* = a_{t+1}Z_{t+1}^*$$

To avoid overflow, we can work in the log domain since logmax = maxlog, this is not possible with the Forward-Backward algorithm. We will see why in the next section.
Here the pseudocode of the algorithm :

Algorithm 3 Viterbi algorithm

Input: $X_{1:N}$, K, $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{\pi}$
Initialize: $\delta_1 = \boldsymbol{\pi} \odot B_{X_1}$, $a_1 = \mathbf{0}$;
for $t = 2 : N$ **do**
  for $j = 1 : K$ **do**
    $[a_t(j),\ \delta_t(j)] = \max_i(\log\delta_{t-1}(:) + \log\mathbf{A}_{ij} + \log B_{X_t}(j))$;
$Z_N^* = \arg\max(\delta_N)$;
for $t = N-1 : 1$ **do**
  $Z_t^* = a_{t+1}Z_{t+1}^*$;
Return $Z_{1:N}^*$

Figure 18: The Viterbi Algorithm pseudocode

To conclude the Viterbi algorithm is a dynamic programming algorithm that efficiently solves the problem of finding the most likely sequence of hidden states (the "Viterbi path") in a Hidden Markov Model (HMM), given an observed sequence of events. It systematically computes the probabilities of all possible paths to ensure that the best path is identified without evaluating every possible combination, which reduces the computational complexity significantly.

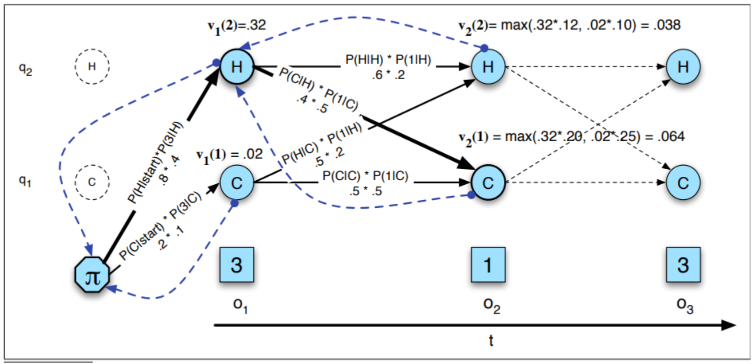


Figure 19: Viterbi Backtrace trellis

3.3.3 Forward-Backward Algorithm

To conclude on algorithms, I will present the *Forward Backward* algorithm that is useful to solve the third problem evoked above : learning the parameters that are A and B matrices. The input of the algorithm would be an unlabeled sequence of observations X and a vocabulary of potential hidden states Z.

In our exemple we would start with a sequence of observations $X = (1, 3, 2, ...)$ and the set of hidden states H and C. The algorithm will let us train both the transition probabilities A and the emission probabilities B of the HMM. The F-B algorithm is a special case of the Expected-Maximization (EM) algorithm. It's an iterative algorithm computing an initial estimate for the probabilities, then using this estimate to compute a better estimate and so on. Iteratively, it improves the probability that it learns.

Let's take an example : Considering the much simpler case of training a fully visible Markov model where we knwo both the temperature and the ice cream count for every day. Imagine we see the following set of input observations and magically knew the aligned hidden state sequence :

$$\underset{hot hot cold}{3\ 3\ 2} \quad \underset{cold cold cold}{1\ 1\ 2} \quad \underset{cold hot cold}{1\ 2\ 3}$$

It would easily allow us to compute the HMM using the maximum likelihood estimation from the training data. Computing π, as the initial disitribution of the model, from the count of the hidden states we get :

$$\pi_h = 1/3 \quad \pi_c = 2/3$$

We can directly compute the A matrix from the transitions while ignoring the final hidden states. It gives us :

$$A = \begin{pmatrix} p(hot|hot) = 2/3 & p(cold|hot) = 1/3 \\ p(cold|cold) = 2/3 & p(hot|cold) = 1/3 \end{pmatrix}$$

We can also compute the B matrix

$$A = \begin{pmatrix} p(1|hot) = 0/4 = 0 & p(1|cold) = 3/5 = .6 \\ p(2|cold) = 1/4 = .25 & p(2|cold) = 2/5 = .4 \\ p(3|hot) = 3/4 = .75 & p(3|cold) = 0 \end{pmatrix}$$

In our example it's easy because we assume knowing the path taken for a given input, but for real HMM we can't compute theses from direct observation. We also don't know the count of being in any of the hidden states. To solve this problem, we use the *Baum-Welch* algorithm by iteratively estimating the counts. To do so we start with an estimate of theh transition and observation probabilities and then use them to derive better probabilities. We do it by computing the *forward probability* for an observation and dividing that probability mass among all the different path that contributed.

Let's introduce the *backward* probability, we denote as β, which is the probability of seing the observation from time $t+1$ to the end. Given we are in state i at time t with the automaton λ:

$$\beta_t(i) = P(X_{t+1}, X_{t+2}, ..., X_T | Z_t = i, \lambda)$$

In a similar manner to forward algorithm we compute the backward algorihm as :

- Initialization

$$\beta_T(i) = 1, \;\; 1 \leq i \leq N$$

- Recursion

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(X_{t+1}) \beta_{t+1}(j), \;\; 1 \leq i \leq N, 1 \leq t \leq T$$

- Termination

$$P(X|\lambda) = \sum_{j=1}^{N} \pi_j b_j(X_1) \beta_1(j)$$

Here's a graph representation of the algorithm :

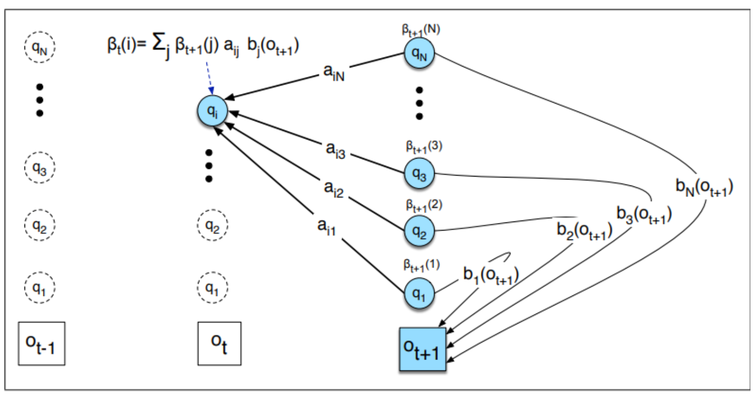


Figure 20: Illustration of the backward induction step

The computation of $\beta_t(i)$ by summing all the successive values $\beta_{t+1}(j)$ weighted by their transition probabilities a_{ij} and their observation probabilities $b_j(X_{t+1})$.

To see how the forward and backward probabilities can help to compute the transition probability a_{ij} and observation probability $b_i(X_t)$ from an observation sequence even if the path is hidden.
To estimate $\hat{a_{ij}}$ we use a variant of the maximum likelihood where

$$\hat{a}_{ij} = \frac{E[\text{number of transitions from state i to j}]}{E[\text{number of transitions from state i}]}$$

To compute the numerator assume that we had some estimate of the probability that given transition i to j given a time t in the observation sequence. If we knew this probability for each particular t we could sum them over the time to estimate the total count for this transition. Formally we define the probability ζ_t as the probability of being in state i at time t and state j at time $t+1$, given the observation sequence and the mode:

$$\zeta_t(i,j) = P(Z_t = i, Z_{t+1} = j | X, \lambda)$$

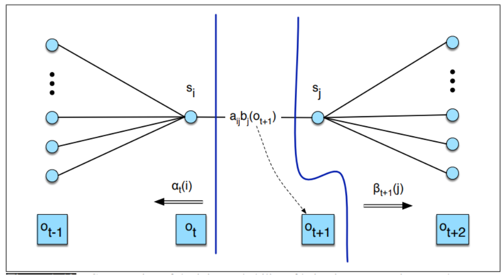


Figure 21: Computation of the joint probability of being in state i at time t and j at $t+1$

To compute it we compute another probability, with a different conditioning on X that we denote $\bar{\zeta}_t$ where :

$$\bar{\zeta}_t(i,j) = P(Z_t = i, Z_{t+1} = j, O|\lambda)$$

The figure above show all the probabilities needed to compute $\bar{\zeta}_t$ where

$$\bar{\zeta}_t(i,j) = \alpha_t(i) a_{ij} b_j(X_{t+1}\beta_{t+1}(j))$$

To compute ζ_t from $\bar{\zeta}_t$ we follow the law of probability and divide by $P(X|\lambda)$, since

$$P(X|Y,Z) = \frac{P(X,Y|Z)}{P(Y|Z)}$$

The probability of the observation given the model is simply the forward probability or the backward probability of the whole utterance :

$$P(X|\lambda) = \sum_{j=1}^{N} \alpha_t(j)\beta_t(j)$$

Now we have all the components to introduce the final equation for ζ_t where :

$$\zeta_t(i,j) = \frac{\alpha_t(i) a_{ij} b_j(X_{t+1})\beta_{t+1}(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}$$

To get $\hat{a}_{ij}$ we only need to sum over all transitions out of state i. So here's the final formula of $\hat{a_{ij}}$:

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \zeta_t(i,j)}{\sum_{t=1}^{T-1} \sum_{k=1}^{N} \zeta_t(i,k)}$$

As we have the formula for $\hat{a}_{ij}$ we need to do the same for $\hat{b}_j(v_k)$ where v_k is a given symbol observed. Where :

$$\hat{b}_j(v_k) = \frac{E[\text{number of times in state j and observing symbol } v_k]}{E[\text{number of times in state j}]}$$

To do so we compute the probability of being in state j at time t which we call $\gamma_t(j)$. As before we will compute it by including the observation sequence in the probability :

$$\gamma_t(j) = \frac{P(Z_t = j, X|\lambda)}{P(X|\lambda)}$$

The numerator of this equation is the product of the forward probability and the backward probability:

$$\gamma_t(j) = \frac{\alpha_t(j)\beta_t(j)}{P(X|\lambda)}$$

Now to compute $\hat{b}_j(v_k)$ we sum $\gamma_t(j)$ at the numerator for all time steps t which the observation X_t is the symbol v_k and at the denominator we sum $\gamma_t(j)$ over all time steps t.

$$\hat{b}_j(v_k) = \frac{\sum_{t=1 s.t. X_t = v_k}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}$$

Now we can re-estimate A and B assuming we already have a previous estimate. These re-estimate are the core of the forward-backward algorithm. Here's the pseudo-code of it :
In practice this algorithm can do completely unsupervised learning, in reality the initial conditions are very important. That's why we try to gives to the algorithm extra information.

function FORWARD-BACKWARD(*observations* of len T, *output vocabulary V*, *hidden state set Q*) **returns** *HMM=(A,B)*

initialize A and B
iterate until convergence
E-step

$$\gamma_t(j) = \frac{\alpha_t(j)\beta_t(j)}{\alpha_T(q_F)} \quad \forall\, t \text{ and } j$$

$$\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}{\alpha_T(q_F)} \quad \forall\, t,\ i, \text{ and } j$$

M-step

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\sum_{k=1}^{N}\xi_t(i,k)}$$

$$\hat{b}_j(v_k) = \frac{\sum_{t=1 s.t. O_t=v_k}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)}$$

return A, B

Figure 22: Illustration of the Forward-Backward algorithm

4 Combining Options and HMM

In this section, I will introduce my strategy and how, using the Hidden Markov Model we could create a profitable strategy by unveiling trends from the market and combining options to make some profit.

4.1 Defining the Strategy

The strategy follows a simple procedure. We make observations of the market on a period of time. Extracting a time series we compute the hidden trends underlying the observations with the Hidden Markov model. Based on the most observed trend we choose an option strategy adapted to the market observation and hold it in our portfolio for a month. At the end of the month, if the market conditions are profitable we exercise the options and take profit, in the other case we take the loss. Here's the strategy formally and mathematically described.

4.1.1 Observation Period and time series

The strategy is based on realizing observations of the returns on an underlying during a rolling 60 days period. Let's denote S_t the level of the underlying at time t. We define the return of the underlying as :

$$return = ln(\frac{S_t}{S_{t-1}})$$

Where ln is the natural logarithm. We compute the return using the logarithm as we assume the underlying is following a log-normal distribution.

Definition : A random variable S is said to have a lognormal distribution if $ln(S)$ has a normal distribution.

The rolling basis is defined as follow : given a time t_0 we observe the data on the window : $[t_0 - 60days; t_0]$. This window is rolling as the expiration date of our options is a month after the purchase date, t_0 is taken as the day of option expiry and the same period is observed twice to detect if there is another trend found by the Hidden Markov Model as we got a new observation window of datas of 30 days. As the market is closed on saturday and sunday, in reality on a daily basis we only get between 20 and 22 days of data by 30 days period.

These observations are very important because, the HMM needs the most possible data to extract a trend on the observations. We could imagine a simpler model with a simple observation window on 30 days of data but in some cases, the model would not converge and gives a wrong return.

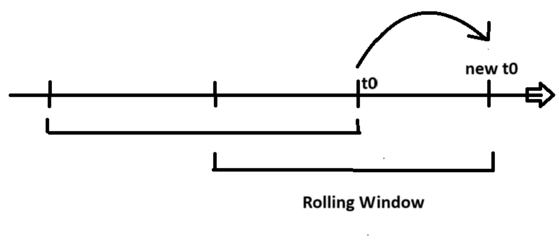


Figure 23: Illustration of the rolling window principle

We compose a time series of data composed of the log return and the variation of the underlying calculated as $variation_{intraday} = S_{high} - S_{low}$.

4.1.2 Extracting a trend with HMM

As we've got our data frame composed of the returns by day and the variation of the underlying intraday we can train our HMM model to unveil a hidden trends. The model gives us back all the patterns found that we can summarize on a graph.
We define 3 state that the model needs to find :

0 A **Bullish** market

1 A **Flat** market

2 A **Bearish** market

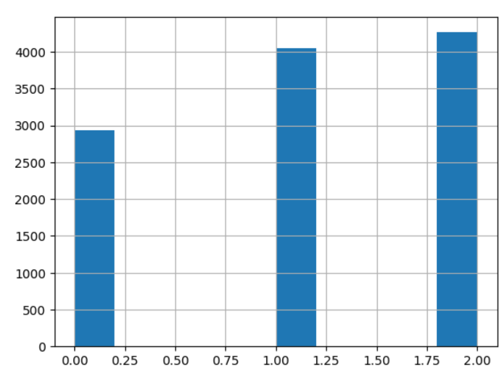


Figure 24: Bar chart of the hidden states unveil on the underlying

Here's the vizualisation of the returns given by the HMM model when trained on the full market data possible on J.P Morgan :
Formally we set the market trend as :

$$MarketTrend = MostObservedState(HMM_{Returns})$$

Where $MostObservedState$ denote the most found hidden state in the time series given to the model.

Figure 25: Market data view of the hidden states on JPM

4.1.3 Defining good option strategy regarding the trend

As we found the most probable trend of the market, we choose the option strategy adapted to the return of the algorithm. Assume that all the strategies are composed of call options valued using the Black-Scholes formula presented before :

- If the model gives a bullish market signal, we buy a bull spread with strikes $K_1 = 110\% \times S_t$ and $K_2 = 120\% \times S_t$ T = 22/252 (about a month)
- If the model gives a flat market signal, we buy a butterfly spread with strikes $K_1 = 105\% \times S_t$, $K_2 = 110\% \times S_t$ and $K_3 = 120\% \times S_t$ and T = 22/252 (about a month)
- If the model gives a bearish market signal, we buy a bear spread with strikes $K_1 = S_t$ and $K_2 = 120\% \times S_t$

If the option strategy enter the optimal zone, we exercise the options, if not we do not exercise, take the loss and after getting the new trend of the market we buy the new optimal strategy.
This strategy gives us the possibility to extract the best from the market, by combining options, we can limit the risk taken and the amount of money needed to enter the market while potentially realizing big gains. But, for professional traders hedging considerations should be taken in account as shorting calls could lead to exponential losses.

4.1.4 More complex models

This model is assumed to be very simple, but open the door to more complex ones who can be more profitable but also with a highest cost to enter the strategy. Let's imagine some variations of the strategy presented above :

First we could imagine making observations of the market on a more frequent basis but using the same rolling window of 60 days and actualizing our portfolio on every 15 days to be sure to always hold the most adapted strategy, regarding the trend given by our HMM, in our portfolio. This strategy could lead to potentially more profit but would cost us a lot more considering our needs to hedge all the options in our portfolio.
We can also imagine a variation of this one buy using American options. It would lead us to potentially more profit as the early exercise feature could lead us to benefit from advantageous market conditions encountered before the expiration date. Using American Options could be seen as better but add complexity to the model as we need to compute the optimal condition to exercise and make observations of the market during the lifetime of the option to ensure an optimal early exercise.

To add more complexity we can imagine longing only American options with time to maturity of 3 months. Using a rolling window of 60 days, we buy the adapted strategy. During each holding period, we could reorganize our options using the different strikes to always getting a winning strategy by taking profit of the early exercise feature.
Let's take an example to make it clearer :
Suppose that at t_0 the HMM unveil a bullish trend of the market we buy a bull spread, as the underlying is $S_{t_0} = 100$ we have two options with strikes $K_1 = 110$ and $K_2 = 120$ in our portfolio. On a second observation day at time t_1, 15 days later, we get a flat signal so as the underlying is $S_{t_1} = 110$ we buy the corresponding butterfly with strikes $K_3 = 115.5$, $K_4 = 121$ and $K_5 = 132$. Later at time t_3, we observe that finally, the market has move down due to a special event, in reality I can rearange my options in my portfolio to construct a bear spread with my short call with strike $K_4 = 121$ and my long call with strike $K_5 = 132$. In this case, imagine my underlying levels falls at $S_{t_3} = 90$, I exercise my bear spread and make a profit of the premium paid for the shorted call with strike K_4.

Variations of this strategy are potentially infinite, while using all the options strategies possible with the right market conditions could lead to a very profitable strategy. We could also assume that a professional investor is buying multiple options at the same time making his portfolio bigger and multiplying his potential gains. We could also imagine implementing a derived strategy taking profit of Exotic options like Gap Options, Barrier Options or Lookback options.

5 Implementation of the Strategy

Please find the code corresponding to this paper at this address :
`https://github.com/Guillaume-melis/options-hmm`

As the ATOS SE cumulative returns over 5 years are -99,11%

Figure 26: S&P 500 Index returns over 5 years

Using the HMM Options strategy, you would have beaten the market by far, investing only a thousand dollar would have bring you back a realized gain of 212.13 euros, equivalent to a cumulative return of 21%, as shown in the Backtest :

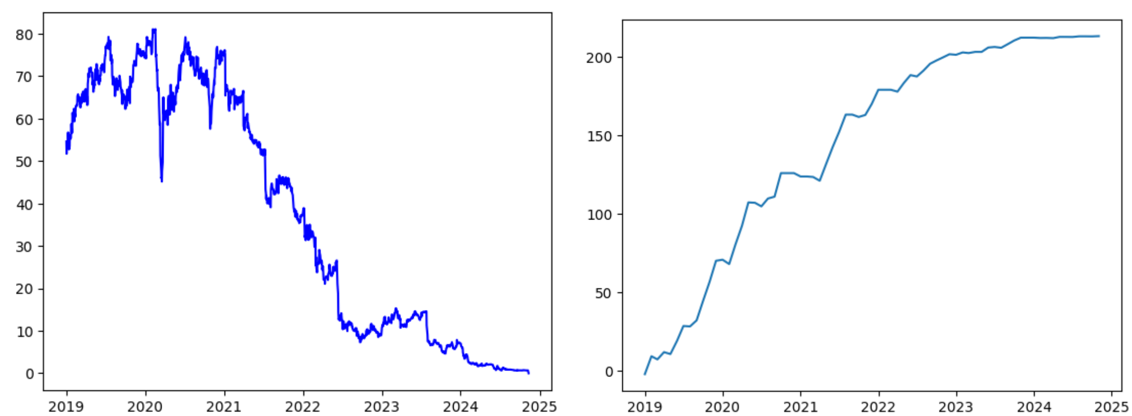


Figure 27: ATOS stock graph comparing to the realized gains of the strategy

We see that the strategy is able to take advantage from poor market conditions, performing well while the underlying stock is crashing.

6 Conclusion

To conclude, I tried to show how combining finance and machine learning could lead to a financially profitable strategy. As we've seen, the math behind the Hidden Markov Model (HMM) is complex, and an observation of the past does not guarantee future trends, especially in the volatile world of financial markets. However, despite these challenges, the Hidden Markov Model remains a powerful tool in financial trading when used correctly.

The HMM excels in identifying latent states of a market, such as bullish or bearish conditions, which are not directly observable. These states drive the underlying market behavior, but traders can only infer them from observable data like price movements, volume, or other technical indicators. By capturing these hidden states and predicting transitions between them, HMMs can help traders to anticipate potential market shifts and adjust their strategies accordingly.

In practical terms, applying HMM to trading involves the following steps:

- **Data Collection and Preprocessing:** The model requires historical market data, including asset prices, returns, or other relevant financial variables. This data needs to be cleaned and prepared for analysis, ensuring it reflects the true underlying market conditions without noise.

- **Model Training:** Once data is prepared, the HMM is trained to fit the historical price movements and identify the hidden market regimes. The model learns how market states (e.g., trending or ranging) evolve over time based on observed data.

- **State Prediction and Trading Strategy:** After training, the model predicts future states of the market. Based on these predicted states, traders can devise strategies such as when to enter or exit trades, the appropriate level of risk to take on, or when to hedge positions. For instance, in a predicted bullish state, a trader may take on more risk, while in a bearish state, they may focus on risk management or exiting positions.

While the HMM cannot eliminate market uncertainty or guarantee profits, it offers a data-driven framework for analyzing market regimes and managing risk. As markets become more data-driven and automated, machine learning models like the HMM can provide traders with a competitive edge, helping them to navigate the inherent unpredictability of financial markets.

Ultimately, the success of such models depends on proper implementation, ongoing tuning, and the recognition that no mathematical model can predict the future with absolute certainty. The value of HMMs lies in their ability to extract meaningful patterns from noisy data, guiding traders in making more informed decisions based on probabilistic outcomes.

7 Bibliography

- Yuh-Daugh Lyuu, Financial Engineering and Computation
- Daniel Jurafsky & James H. Martin, Hidden Markov Model
- J.C. Hull, Options, Futures and other derivatives
- D. Lamberton, B. Lapeyre, Introduction au calcul stochastique appliqué à la finance
- Peter Tankov, Surface de Volatilité
- M. Bouzoubaa, Exotic options and Hybrids, *A Guide to Structuring, Pricing and Trading*
- Anuranjan Misra, Graph Theory and Probability

8 Sources

- Mauro Cesa, A brief history of quantitative finance : https://probability-risk.springeropen.com/articles/10.1186/s41546-017-0018-3
- The history of options trading : https://www.optionstrading.org/history/
- Basic Markov Chains https://www.di.ens.fr/~busic/cours/M2amis/markov.pdf
- Introduction to Hidden Markokv Model https://scholar.harvard.edu/files/adegirmenci/files/hmm_adegirmenci_2014.pdf