UPPSALA
UNIVERSITET

UPTEC F 22030
Examensarbete 30 hp
Juni 2022

Signal detection of FX Fixing events

Anton Sjöström

Civilingenjörsprogrammet i Teknisk Fysik

UPPSALA UNIVERSITET

Signal detection of FX Fixing events

Anton Sjöström

Abstract

This master thesis investigates the price dynamics of two currency pairs, GBP/USD and EUR/GBP, during the event called the “London 4 PM Fix”, which is a daily event. The dynamics of this event is understood by first creating a mathematical model to find the theoretical optimal trading strategy given a number of assumptions. Later, unsupervised learning on time series is used to understand price dynamics by firstly clustering on the mathematical model and secondly clustering on real market data. In addition, supervised learning is used to determine how predictable the price evolution after the London 4 PM Fix is. Using these predictions, multiple trading strategies have been constructed and evaluated.

Teknisk-naturvetenskapliga fakulteten

Uppsala universitet, Utgivningsort Uppsala

Handledare: Simon Österberg Ämnesgranskare: Elisabeth Larsson

Examinator: Tomas Nyberg

1 Populärvetenskaplig sammanfattning

Detta examensarbete undersöker marknadsdynamiken för valutaparet GBP/USD samt EUR/GBP i FX spot market, vilket är marknaden för direkta byten av valutor. Projektet avgränsar sig till ett specifikt marknadstillfälle som kallas London 4 PM Fix vilket är ett mått som baseras på ett viktat medelvärde av priset vid just kl. 16 London-tid. Eftersom svensk tid är en timme före sker detta kl. 17 svensk tid.

Detta mått används som ett utgångsläge för att värdera portföljer med utländska värdepapper. Detta ger incitament att placera Fix-ordrar, vilket är en order där man bestämmer att man ska köpa en specifik kvantitet för det pris som blir Fixing-priset den dagen. Detta ger banker, som tex SEB där detta projekt utförs, en stor positions som de vill säkra upp genom att själv handla. Detta ger möjlighet till olika strategier för bankerna, vissa som är på kundens bekostnad och därför förbjudna, och vissa som kan både gynna kund och bank.

För att förstå detta används en matematisk modell där en optimal strategi av ett stokastiskt spel (som ska representera en market makers point of view under Fixen) är beräknat. Vad som händer under detta spel beror främst på två parametrar, något vi kallar temporär- och permanent market impact. Beroende på dessa parametervärden skiftar den optimala strategin från att handla åt samma håll (men olika mycket) varje ronda till att handla i båda riktningarna, ena riktningen innan kl 17 och andra riktningen efter kl 17. Detta ger upphov till två olika prisdynamiker i marknaden, en dynamik med trend och en som reverserar. I denna modell kan man se att hur priset förändras innan kl 17 kan förutse hur priset kommer ändras efter kl. 17. Detta är en signal som kan användas för att ta beslut på hur man själv ska agera i marknaden under denna tidsperiod.

Den optimala strategin enligt modellen tjänar pengar på kundernas bekostnad, vilket gör att den bryter mot praxis som har bestämts globalt inom FX. I detta projekt vill vi förstå hur vi kan använda att andra följer den föreslagna teoretiskt optimala strategin för att vi som bank ska kunna positionera oss bättre på ett rättvist sätt.

För att förstå hur bra denna matematiska modell representerar vad som händer i marknaden används unsupervised learning för att klustra olika marknadsbeteender från interna marknadsdata och man kan se tecken på att de strategier som är optimala enligt modellen verkar användas i praktiken. Detta görs både genom att använda KMeans för tidsserie, samt genom att använda djupinlärning med en autoencoder. Detta öppnar ytterligare upp möjligheten att prediktera hur priset kommer förändras efter Fixen och vi kan ta strategiska beslut för att hedga våra positioner.

Genom att använda supervised learning kan vi se att vi kan prediktera utfallet efter Fixen med runt 60%, detta genom att bara analysera tidsserien fram till Fixen. Med den modellen bygger vi en tradingstrategi och testar på historiska data. Resultatet på tradingen generellt slår en slumpartad strategi men resultaten är inte lika övertygande som prediktionen.

2 Background

2.1 Foreign exchange market

The foreign exchange market is the market to buy and sell foreign currencies and is the largest marketplace in the world [1]. The foreign exchange market is not too different from exchanging currencies before going on a holiday abroad, you buy a foreign currency by selling your domestic currency and the price fluctuates based on supply and demand [1]. Currencies are traded between partners in a so-called OTC (over-the-counter) way. This means that there is no physical exchange and no central exchange. The market is open 24 hours a day, opening in the Sunday evening in NZ and closing Friday night in the US.

Most of the trading is done between banks, funds, and multinational corporations. These trades differ from the exchange done before a holiday. Instead of using the currency for a purchase, some of these trades are done in a speculative manner or for hedging reasons [1]. The three largest markets are the FX swaps market, the spot market, and the forward market [2] but this project only focuses on the spot market where pricing and trades are done in real-time.

2.2 Limit Order Book

In the market, the two most common orders are the *Market Orders* (MO) and the *Limit Orders* (LO). MOs are more aggressive than LOs since they are used to fill a trade of a specific quantity directly at the current best price, while LOs are more passive by stating exactly which price and quantity they want to trade. It then waits until it gets matched [4]. The price of the LO is usually worse than the available best price in the market and often not matched directly. The LOs are contained in a *Limit Order Book* (LOB) which is used to primarily match MOs with LOs and secondarily to match LOs with LOs.

The best bid price (to buy) and the best ask price (to sell) is most often not the same, though it can happen [4]. The price difference is called the *spread* and the mean of the two prices is called the *mid*. The mid is often used as an approximation of the fair value of the asset. The spread, however, can be thought of as the premium you pay if you buy and then immediately (or in near time) sell the asset. An order book is structured in levels of prices, where each level has a quantity and a price that can be traded. The tighter the difference between the best and second-best price (and third etc) are the more *liquid* the asset is thought to be. Also, if the quantity in the order book is larger, then it is thought of as being more liquid.

When an MO is placed, the LOB gives the MO the best price for that given quantity. If the MO requires a greater quantity than the quantity at the best price, the remaining quantity is matched against the second-best price, etc until the whole trade is filled [4]. The deeper into the book you go, the worse the price becomes. Therefore, large orders have a higher risk of getting a worse price.

Since an order book is structured in this way, a price impact can happen if a large volume is bought/sold. The large order dig deep into the order book and remove the best prices. The new best prices are the prices that were the second (or third etc) best price of the order book before the trade occurred [4]. This price impact is however only temporary since it does not change the fair value of the asset, it just depletes the order book for a short time. This type of price impact is called a *temporary market impact*.

Another type of price impact can also happen if there is a consistent imbalance between buying and selling orders. A linear relationship between orders and price change can be seen [4]. This has also been studied in [7]. When there is an imbalance, the price either increase or decrease until it reaches a new fair price. This type of price impact is called *permanent market impact*.

To be clear, temporary and permanent market impact is not physical attributes of the order book, instead, these phenomenons can happen depending on the trading activity and the structure of the order book. In general, small liquidity relates to both types of market impacts [4].

We can further understand these concepts with an example. Let's say you would like to buy 10,000, and in the order book you see that you can get the first 4,000 for EUR/USD 1 (1 EUR per USD) but the other 6,000 you can get for EUR/USD 1.01. We can then see that the average price for all 10,000 is EUR/USD 1.006. So the fact that you wanted to buy 10,000 instead of 4,000 made it cost EUR/USD 0.006 more. Now that the quantity at the best price of EUR/USD 1 has been depleted, all other traders see a new best price of EUR/USD of 1.01. The change in best price for you and the others stemming from a large order is what we define as temporary market impact.

Now, when this has been bought, the market sees that you are willing to buy 10,000 and therefore thinks that it is undervalued. The market, when the order book is completely refilled, increases the price to EUR/USD 1.002 as the best price. This is what we call permanent market impact.

2.3 Market Makers and Liquidity Traders

As mentioned before, the two most common orders are the MOs and the LOs. Which type of trader make these trades? It's possible to separate traders as Market Makers and liquidity traders [4]. In our context, SEB can be thought of as a Market Maker and a Swedish company can be thought of as a liquidity trader. Market Makers are usually the ones placing LOs and the liquidity traders are usually the ones placing the MOs. The Market Makers are acting as liquidity providers and earn the spread by taking on the risk of holding the asset. Liquidity traders pay the spread and try to earn money by predicting the movement of the market. Sometimes market makers can act as liquidity traders, therefore the separation is not perfect.

2.4 Algorithmic Trading

Algorithmic trading, or algorithmic trading, has had a rise in usage these past years. In the year 2000 algorithmic trading only contributed to 1% of all trading, while in 2019 al-

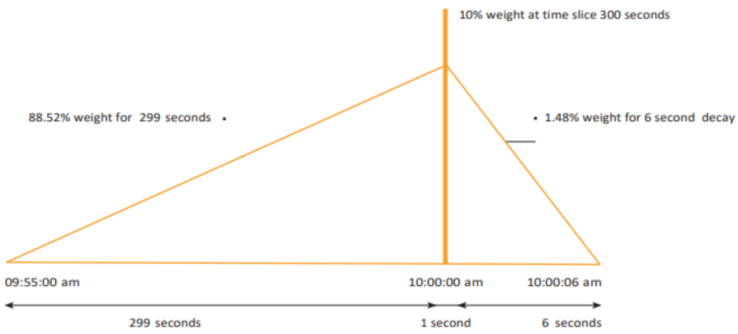

Figure 1: How the Bloomberg Fix is calculated [3].

gorithmic trading contribute to 93% [10] of all trading. Algorithmic trading is the method of using computers, mathematical modeling, machine learning, and artificial intelligence to make trades by following instructions and rules set by the creator. There are several types of algorithmic trading classes, from execution to profit-seeking, to high frequency and they are used for different scenarios.

Algorithmic trading has several additional perks such as getting lower commissions, be more anonymous, and have lower transaction costs to name a few [10]. algorithms can have different types of aggression, ranging from aggressive to passive. An aggressive algorithm could be an algorithm that tries to trade away as much liquidity as possible at a specific price, while a more passive algorithm is the TWAP (time-weighted average price) which splits an order into smaller orders that get traded during a time window to reduce the market impact and cost while having to compensate with being at a greater risk to market change.

Algorithmic trading has reduced the average volume per single transaction. Block trades have been reduced from 50% in the year 2000 to 5%. Large orders are still traded, but they are split into more but smaller batches and traded over some time instead [10]. Algorithmic trading has opened up the field of macro-level strategies where the algorithm uses decision rules to optimize its trading strategy. Using real-time data, such as price and liquidity, it can decide when it is optimal to do a trade. Therefore algorithmic trading is not only used to do trades easier and more cheaply by avoiding market impact, but also to make real-time trading strategies.

2.5 London 4 PM Fix

As discussed earlier, the market never closes on business days, therefore no close-price benchmark prices are available. Instead, the London 4 PM Fix is used as a benchmark to compare portfolios in different currencies [15]. The Fix is a sample average of the best price during a small time window around 4 PM.

Bloomberg's Fix is transparent in the way it is calculated. It is a weighted mean of the price around the Fix where the weights are set according to Figure 1. Here it is described for the 10 AM Fix, but it is the same for the 4 PM Fix.

Since the London 4 PM Fix is considered a major benchmark, the benchmark is used to construct indices. The benchmark is also used to compute returns for a portfolio containing assets in more than one currency. Since this benchmark have this influence, people are incentivised to trade at this particular time and price. By trading at the benchmark price, the holder of the portfolio can justify the evaluation of the portfolio using the Fix [15]. Therefore dealers such as banks have made it possible to place Fix orders. A Fix order is a trade that is determined before the Fix but takes place after the Fix, where the price is determined to be the Fix price. In that way, people can easily trade at the Fix price without actually trading in the market and therefore reduce the tracking error. This reduces the risk for the customer, but the risk for the bank increase. The bank has a large position before the Fix which the bank would like to hedge away. Since the bank takes on this risk, the customer has to pay a premium. This can be thought of as a price the customer pay to make sure the price of their trade is close to the Fix price.

For being a small time window, a lot of trading activity is compressed into this time. This can give rise to special market structures, which Michelberger and Witte have investigated in [19]. Others have also investigated this, e.g. [6].

Both [19] and [6] have seen an increase in volatility and volume traded at the Fix. Local price extremes have an increased probability to be in a small time window around the Fix, and daily extremes also have an increased probability of being in this window [19]. Price correlations between pre-Fix and post-Fix have been seen. An increase/decrease in price before the Fix is correlated with a decrease/increase after the Fix, a type of price-reversal [6].

Qualitative models have also been made and investigated to understand the dynamics around the Fix. Examples of this can be found in articles [12], [14] and [15]. These models are used to find an optimal trading strategy and by using these try to explain why the market dynamics look the way they do during the Fix. In this paper, a similar approach is taken but for both types of market impact. It is used to understand the optimal trading and to further explain the market dynamics we see.

2.6 Problem formulation

The aim of this thesis can be split into three parts. Firstly, the aim is to understand what happens during the Fix and preferably why it happens. Secondly, using this knowledge we want to be able to find a signal in data which describes what happens during the Fix. Thirdly, using this signal we want to be able to use it to predict what will happen during the Fix before it happens.

2.7 The structure of the thesis

In this master thesis, both a qualitative and a quantitative study are performed for the London 4 PM Fix. A mathematical model is analyzed and compared to real market data and model behavior is used to understand the behavior around the Fix. This is done with unsupervised market clustering. An optimal trading strategy is also derived given some assumptions. Supervised learning is also used to predict what happens after the

Fix, in particular, if the price of a currency pair increases or decreases five minutes after the Fix compared to at the Fix. Using these predictions, a trading strategy is deployed to capitalize on the predictiveness of the Fix-dynamics.

This project is important for several reasons: The first reason is economic, both for the bank and the customers. By having an automatic trading strategy that has a stable positive expected gain and lowered risk, the bank earns more money with reduced risk and can offer a better price to the customers.

The second reason why this project is important is the different methods the project uses. This is more important for practical reasons. Machine learning on time series is in general more complex than on data that does not have any sequential order. One way of handling this is to reduce a time series to just one data point with features describing the time series, which enables the use of ordinary machine learning. In this project, however, the time series is kept as a whole and instead special methods are used to handle the sequential order.

Another method explored is labeling for supervised learning by using unsupervised learning. In finance, there is a lot of data and it can be hard to label all this data by hand. Instead of labeling by hand, unsupervised clustering can be used to create labels. The technique of combining unsupervised and supervised learning, if proven to be successful, can be used for similar task in the future.

The last reason is more academic. Several theoretical papers have been made discussing what should happen during the Fix given some assumptions. However, to the author's knowledge, no investigation has been made for a model with both temporary and permanent market impact. Therefore this project takes on this task and see how well theory and reality line up.

The paper is structured in the following way: First, we look more in-depth at the related papers, then we explain some theory and draw some conclusions about the optimal trading strategy. Later the methods used in the project are discussed. After this, some investigation of unsupervised and supervised learning are done to understand the dynamics around the Fix and lastly, trading strategies are investigated and discussed. Some general conclusions are drawn and recommendations for further works are mentioned.

3 Related Papers

3.1 Quantitative Research

In this section we return to the quantitative findings from the literature around the Fix. Michelberger and Witte [19] used spot rate data from Bloomberg for 12 different currency pairs in the periods 2008–2014 and 2010–2014. The data was minute-by-minute and covered the whole day except the first and last hour of the day. They saw that price volatility extremes would consistently be found around the London 4 PM Fix across all investigated currency pairs. They could see this both in the minute before and after the

Fix, but more consistently before the Fix. For some currency pairs, the volatility spike after the Fix was not significant. Not only this, they could also see an increase in the probability of having a local extreme and daily extreme in spot prices around the Fix. The same could be said about the volumes traded.

Evans in [6] used price data between the years 2004–2013 from Gain Capital. He investigated 21 currency pairs and looked at the market dynamics before the Fix, after the Fix, and the correlations between before and after. He could see that the market behavior before the Fix was dissimilar to other parts of the day. By bootstrapping samples of the size of the price change before the Fix and at other times Evans could see that they did not seem to come from the same distribution. Price changes before the Fix are greater than price changes throughout the day.

Similar results can be seen right after the Fix as well. The price changes are not as big after the Fix as right before the Fix, but still larger than throughout the day, around twice as large. Another interesting insight Evans found is that price movement is negatively correlated before and after the Fix. In other words, an increase/decrease in price before the Fix seems to give rise to a decrease/increase in the price right after. This looks like a price reversal around the Fix. By looking at the average price path for an increase/decrease before the Fix, he could see that the average price path reverted again after the Fix.

3.2 Qualitative Research

Not only quantitative research has been made around the Fix. Mathematical modeling has also been performed to try to understand the market behavior around the Fix. The models are slightly different, but the overall goal is the same: Given the model, what is the optimal trading strategy as a dealer when trying to hedge your Fix position?

Saakvitne in [14] used a model with an exogenous temporary price impact linear to the trade volume to find an optimal trading strategy. The optimal trading strategy (in this theoretical model) included a concept called *banging the close*. This is discussed more in depth later on, but in general, the concept capture the behavior of hedging some of the position early and then trading a large portion of the position at the Fix to later hedge the rest after. This behavior gives a net-positive gain because the temporary market impact at the Fix moves the price favorably for the dealer.

Saakvitne again in [15] used a permanent market impact and showed that another strategy became the optimal trading strategy for this new model. This trading strategy used two concepts called *front running* and *overtrading*. Front running happens when the market maker trades before the customer to move the price in the market maker's interest due to permanent market impact. In this optimal strategy, front running trades combined with the trade at the Fix were greater than the total position such that after the Fix the market maker had to buy back the asset to end up at a net-zero position, this is what we call overtrading. This type of trading gave a rise to upward-downward/downward-upward reversing dynamic, which is similar to what Evans found in [6].

Both the concepts banging the close and front running is regulated by the FX Global Code. Hedging in a way to benefit the market maker on the clients cost goes against the code [5].

Saakvitne in [15] used another model with an endogenous price impact and found different results. Using this price impact function, front running was not as profitable as before. Then again in "Order anticipation and large traders - evidence of FX markets" from [16] he used the same exogenous price impact function to show the profitability of front running before large trades in general and he found evidence in market data that this is what happens.

4 Theory

4.1 Theoretical Fix model

A mathematical model has been developed to capture attributes around the Fix. This model is heavily inspired by the models in [14] and [15] and includes an exogenous market impact that is linear to the traded quantity. The value of the underlying asset, in this case, a currency pair, moves as a Geometric Brownian Motion (GBM) without a drift term. The price impact are separated into two parts, the first is a permanent market impact that change the value of the underlying asset, and the second is a temporary market impact that only affect the price during that trading period. To include both of these market impacts separates this model from the models by Saakvitne. These two different market impacts are linear to the total traded quantity but with different constants, α and β. The true value of the underlying asset can be described as equation (1) where ϵ_t is a normal distributed random variable at time step t, x_t is a the traded quantity at time t and a_t is the fair value of the asset at time t. The variable p_t is the price of the asset at time t. The time index is denoted as t, where each time step does not necessarily represent a trading round. For time step which are not trading rounds, $x_t = 0$. Therefore, if a trading round is happening at time t, the time t-1 does not necessarily refer to a trading round. Note that you pay the price (p_t) when you trade the quantity x_t, which is how Saakvitne did in his models as well, both the one with temporary market impact and the one with permanent market impact.

$$\begin{aligned} \epsilon_t &= N(0,1), \\ a_t &= a_{t-1} - \beta x_t + \sigma^2 a_{t-1}\epsilon_t. \end{aligned} \tag{1}$$

The motion of the price of the underlying asset can be described by equation (2)

$$p_t = a_t - \alpha x_t. \tag{2}$$

There are N trading rounds that are equidistant in time, where one of these trading rounds are denoted as the Fix. There is one market maker which has a large Fix order (x_f) and liquidity traders such that the market maker can do its trades.

The market maker initiates with a specific Fix order that they need to hedge arbitrarily, but with the condition that they end with a zero inventory. This can be described by equation (3)

$$\sum_{t=0}^{N} x_t = x_f. \tag{3}$$

The earnings for each player can be determined when everything has been traded and are calculated according to equation (4), which can be a negative earning if there has been a loss

$$G = \sum_{t=0}^{N} p_t x_t - p_f x_f. \tag{4}$$

In previous works, the models have been used to derive an optimal trading strategy. However, this model is both be used to find the optimal trading strategy (section 4.2) and to be able to simulate different market scenarios to understand how that affects the price during the Fix (section 6.2).

4.2 Optimal trading strategy

There have been two different models investigating the optimal trading strategy during the Fix. The first was [14] which looked at the optimal strategy when having a temporary market impact but no permanent market impact. The second was [15] which looked at the optimal strategy when having a permanent market impact but no temporary market impact.

In the model suggested by this thesis, both permanent market impact and temporary market impact are investigated at the same time. When both types of market impacts are in play, what is the new optimal trading strategy? Let's define a simple game where there is one market maker which holds a Fix position, $x_f = 1$, which is promised to be traded at the Fix price. In the model the Fix price is just the price at the Fix round. Let there be three trading rounds, where the second round is the Fix. After the game is done, the market maker earns an amount corresponding to equation (4).

According to the paper in [14], the optimal strategy is to trade [1/6, 2/3, 1/6] such that the majority of the position is traded at the Fix round. This gives the maximum expected value of this game. The paper in [15] suggests that the optimal strategy is [2/3, 2/3, -1/3], where before and during the Fix the market maker has over-traded and has to buy back at the last round. Remember, these two models have different types of market impact and therefore have different optimal strategies.

In the model suggested in this thesis, both types of market impact are considered, which to the author's knowledge has not be done before. What is the optimal trading strategy in this game according to this model? How does it depend on the relation between the relative size of temporary and permanent market impact?

The expected value of this game is given by equation (5), where β represents the permanent market impact and α represents the temporary market impact. Here we have used the fact that $x_3 = 1 - x_1 - x_2$ according to equation (3)

$$E[G] = \beta(-x_1^2 - x_2^2 - x_1x_2 + 2x_1 + 2x_2 - 1) + \alpha(-2x_1^2 - 2x_2^2 - 2x_1x_2 + 2x_1 + 3x_2 - 1). \quad (5)$$

Taking the first derivative of x_1 and x_2 and solve the roots for those derivatives gives us the optimal strategy. The optimal trading strategy are according to equation (6)

$$\begin{aligned} x_1 &= \frac{2\beta + \alpha}{3(\beta + 2\alpha)}, \\ x_2 &= \frac{2}{3}, \\ x_3 &= \frac{-\beta + \alpha}{3(\beta + 2\alpha)}. \end{aligned} \quad (6)$$

In the limiting cases, we recover both the strategies suggested in [15] (equation (7)) and in [14] (equation (8))

$$\begin{aligned} \lim_{\alpha \to 0} x_1 &= \frac{2}{3}, \\ \lim_{\alpha \to 0} x_2 &= \frac{2}{3}, \\ \lim_{\alpha \to 0} x_3 &= -\frac{1}{3}, \end{aligned} \quad (7)$$

$$\begin{aligned} \lim_{\beta \to 0} x_1 &= \frac{1}{6}, \\ \lim_{\beta \to 0} x_2 &= \frac{2}{3}, \\ \lim_{\beta \to 0} x_3 &= \frac{1}{6}. \end{aligned} \quad (8)$$

The main interesting result of this is that in one limit we over-trade, but in the other limit we do not. Therefore, there must be a relation between the permanent and temporary market impact where it is more favorable to over-trade than not. In Figure 2 and in equation (6), we can see where this change happens. In figure 2 the x-axis represent the relationship β/α and on the y-axis the quantity that should be traded is represented.

In Figure 2 we see that when the permanent market impact becomes larger than the temporary market impact it is worth to over-trade, which can also be seen in equation (6). This implies that on days where the permanent market impact is larger the price has a reversion after the Fix, and on days where the temporary market impact is larger the price continues in its trajectory. Also, when the permanent market impact is larger, more quantity is traded before the Fix to front-run. Therefore, on days where we expect a reversion of price, the price should increase more rapidly before the Fix.

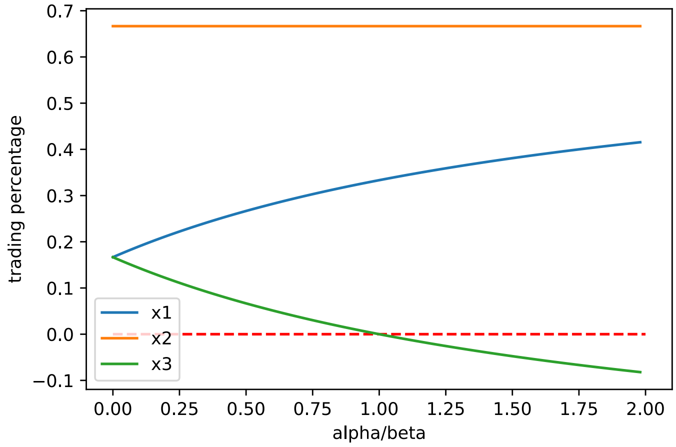

Figure 2: The optimal trading strategy given the relationship between permanent and temporary market impact.

Another question is what is the expected gain from trading in this way? Does some relationship between the two types of market impact lead to a negative expected gain? Inserting the optimal trading strategy into the equation of expected gain, we can see how the expected gain depends on the permanent and temporary market impact only. This is done in equation (9)

$$E[G] = \frac{\alpha^2}{\beta + 2\alpha} + \frac{\beta}{3} - \frac{\alpha}{3}. \tag{9}$$

Equation (9) is always larger than 0 since $\frac{1}{3}(\beta^2 + \alpha^2 + \beta\alpha) > 0$ for all $\beta, \alpha > 0$. This does not mean that we can't lose money on this strategy, it means that on average we should gain money using this strategy. Remember, this is just the expected value, since the underlying asset moves as a GBM.

In Figure 3 we can see the expected gain for different market impacts. We can also see that an increase in permanent market impact increases the expected gain more than an increase in temporary market impact, at least in this domain of $[0.1, 5] \times [0.1, 5]$. For very large market impacts an increase in temporary market impact increases the expected value more. This is because of the non-linear relationship in temporary market impact. It is better to either have no temporary market impact or a lot in this domain, not to be in the middle.

We have now worked this out for three trading rounds, but it can be extended for more trading rounds. For ten trading rounds, where the Fix is at round five, the optimal trading

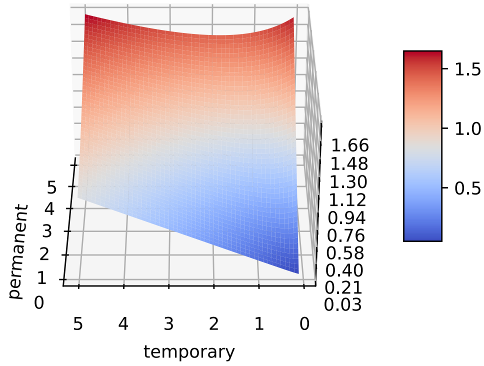

Figure 3: The expected gain given the relationship between permanent and temporary market impact.

strategy according to equation (10) becomes

$$
\begin{aligned}
x_{i<f} &= \frac{6\beta + \alpha}{10(\beta + 2\alpha)}, \\
x_f &= \frac{6\beta + 11\alpha}{10(\beta + 2\alpha)}, \\
x_{i>f} &= \frac{-4\beta + \alpha}{10(\beta + 2\alpha)}.
\end{aligned} \tag{10}
$$

Now the quantity at the Fix round is not constant anymore, but we can see that we still bang the close when we have a high temporary market impact. This can be understood by changing the α to 11α between the pre-Fix rounds and the Fix round. We see also that all the trading rounds before the Fix are equal, and the same can be said about the trading rounds after the Fix. The relationship between temporary and permanent market impact when the strategy goes to overtrading also changes. Now we almost always overtrade unless we have a lot more temporary market impact than permanent market impact.

How much information about the trading after the Fix is contained in the trading before the Fix? If we look at equation (10) we can see that the pre-trading and post-trading are correlated. An increase in β in pre-trading leads to an increase in pre-trade quantity but leads to a decrease in post-trading, or an increase in the negative direction. However, if we focus on α, an increase in pre-trading results in an increase in post-trading as well. In Figures 4 and 5 we can see how they relate to each other for different values of α and β. We can see that in general, when we have a large pre-trade quantity, we have a large negative post-trade quantity. Since we have a permanent market impact, this can be translated to the price trend. If we have a large price trend in either direction before the Fix, we can expect to have a large price trend in the opposite direction after the Fix. So there is quite a lot of information about the price dynamics after the Fix coming from the information about the price dynamics before the Fix. This can be useful since there

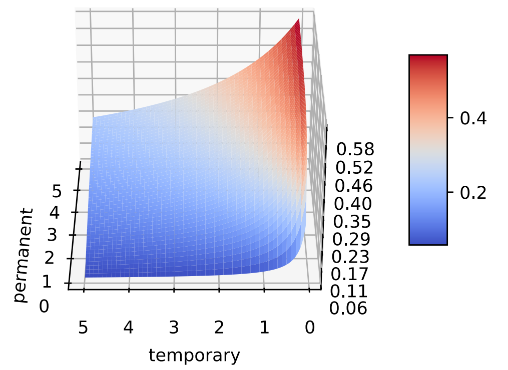

Figure 4: The pre-trade amount for different α and β.

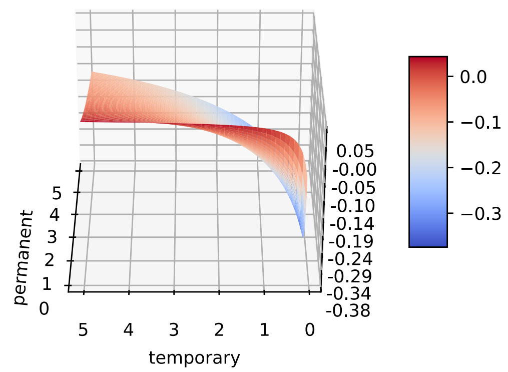

Figure 5: The post-trade amount for different α and β.

is no clear way how to measure temporary and permanent market impact and instead by just looking at the price trend before the Fix we can get an idea about what will happen after the Fix.

4.3 Relevant insight from Banging the close

In "'Banging the Close': Price Manipulation or Optimal Execution?" [14] Saakvitne derives an optimal trading strategy from a model similar to this paper's model, but without permanent market impact. Instead, the model has multiple market makers. In his model, everybody knows what positions the other market makers have initially. Regardless of which direction your position is, to trade optimally you trade equal to everybody else before the Fix since market maker can profit from the other market makers orders. The direction of the trades are in the same direction as the sum of all market maker's Fix positions. Then at the Fix these market makers make sure to end up with a net zero position by trading at the Fix. The expected gain in the same for all market makers and is linear to the market impact in the model.

In the first model of Saakvitne in [14], the market makers were not risk-averse at all. Later in the paper, this was changed and risk-averse agents were considered. With this change, a change in the optimal strategy is made where less is traded before the Fix to reduce the risk, but otherwise the strategy is similar.

What conclusions can be drawn from this model? Let's start with the fact that everyone starts by trading the same volumes and direction before the Fix regardless of Fix position. Therefore adding more market makers does not seem to increase the complexity of the strategy. Since the problems are such alike between this thesis's model and Saakvitne's model, it indicates that adding more market makers does not increase the complexity of this thesis's model as well.

The second conclusion that can be drawn is that the expected profit is linear in the temporary market impact. This means that on days with a large market impact, we would expect to gain more from using this strategy. On days where the market impact is low, we would expect to gain little from using this strategy. We can see the same in thesis model.

The third conclusion that can be drawn is that risk-averse market makers tend to trade less before the Fix since trading far away from the Fix is more risk-full. If we combine this conclusion with the second conclusion, we can conclude that it might only be worth committing to this strategy when the expected profits are higher, in other words, when the temporary market impact is high. At other times, it might be better to trade more safely to minimize risk. This reasoning should also hold in this thesis's model. This means that on days with low market impact we might not see the market makers sticking to this optimal trading strategy.

4.4 Simple trading strategy

Let's say that you as a dealer does not want to participate in this front running and banging strategy, which is against the FX Global Code since it benefits from artificially increasing the price for the customers. Let's say you only want to hedge your risk by trading after the Fix has taken place, what strategy should you use?

From the conclusions in section 4.2, we can create a simple predictor for the dynamic after the Fix. Let's say we observe that the price is starting to increase rapidly five minutes before the Fix and it continues to trend upward a lot until the Fix, then we might expect that an overtrading strategy has been put into play. If that is the case, then we would expect the price to start to decrease after the Fix. Let's say we were positioned such that we would like to buy in this asset, we would then wait until after the Fix to buy since the price is expected to be reduced, due to permanent market impact and overtrading. This increases the chance of earning some money, but it also exposes the trader to a risk.

If the price has not increased too much, then we would expect the "banging" strategy and not overtrading. Therefore the price is more likely to continue to trend and not revert. We would therefore trade at the Fix price and not wait. In this project, it is assumed that the dealer can hedge in a way to get exactly the Fix price. In an earlier project done internally, this has been shown to not be a bad assumption. This can be done using a TWAP. Remember, a TWAP split a large order into smaller orders and trade them over a period of time instead of all at once. If we split these orders in the same way as the Fix's weight schedule, the average price of the TWAP would be close to the Fix price.

The model could be built like this: If the price has moved more than a threshold of x between five minutes before the Fix and at some time point closer to the Fix, then we expect the price to revert after the Fix and we act accordingly. If not, we don't expect the price to revert but instead continue trending and therefore we should reduce our risk by trading using the Fix-weighted TWAP. This is a very simple strategy which I call the Simple Classifier (SC) which will be evaluated later. The only parameter to adjust is the threshold x, which is not evident what it should be. Therefore we have to try a lot of different thresholds and see which performs the best.

More advanced models using machine learning and deep learning is discussed later in the thesis, and they build upon this simple strategy and the other conclusions from the end of section 4.2.

4.5 KMeans

To understand some of the more complex models, and to understand the investigation of the market dynamics that happens later, some theory around the methods need to be discussed. Mainly two types of methods is used, both in the domain of machine learning. The first method is based on understanding a signal based on signals that it resembles, which is the method KMeans do. The other method uses deep learning. Deep learning also tries to understand the signal based on how it resembles other signals, but it does this in a non-linear way and by finding a function to take this decision that minimizes some error. The differences becomes evident in the following sections.

KMeans is an unsupervised machine learning technique used to cluster data points into k clusters. This is used later to investigate and understand some of the dynamics around the Fix by seeing if some clusters of time series emerge during the Fix. For example, could we separate four types of time series clusters during the Fix?

The way KMeans work is that some distance measured is defined, for example, *squared Euclidean distance*, and the clustering algorithm finds clusters of points that have a small distance between each other and greater distance to other clusters. The squared Euclidean distance between the points X and Y in the Euclidean space can be defined by equation (11)

$$d(X, Y) = ||X - Y||^2. \tag{11}$$

The algorithm begins by creating k centroids and placing them randomly in our space. Then each data point is clustered by its closest centroid. Hereby closest we mean the least distance according to the chosen distance measure. When every time series has been clustered, the centroids change positions to the mean of the data points in its cluster. Then we cluster again by assigning each data point to its closest centroid and the process reiterates. Once no change in centroids positions is made the algorithm has converged to a (local) optimum and the algorithm terminates.

4.5.1 Time series KMeans

Time series can be clustered in a similar way as points in space can be. A point in Euclidean space is defined as a one-dimensional vector with the number of elements equalling the number of dimensions in the space. However, this is not true for a time series in the same space. A time series is defined as a list of vectors where each element of the list defines a point in the space. Now, how would we define a distance measure between two-time series? One approach is to use a variation of the earlier defined Euclidean distance. This distance measure is defined in equation (12) where we take the sum of ordinary squared Euclidean distance over time

$$d(\mathbf{X}, \mathbf{Y}) = \sum_{t=0}^{T} d(X_t, Y_t) = \sum_{t=0}^{T} ||\mathbf{X_t} - \mathbf{Y_t}||^2. \tag{12}$$

4.6 K Nearest Neighbors (KNN)

The method K Nearest Neighbors (KNN) is similar to KMeans, but instead of using it as an unsupervised clustering algorithm, it is used as a supervised classifier. KNN trains on a data set by looking at the time series and remembering the class label for each time series. Then when it sees a new time series, it looks at what class label the majority of the k nearest time series and classifies the new time series as that label. This is a very simple but effective way of classifying time series.

One of the more advanced strategies is be based on KNN. This has the benefit that it does not only look at how much the price has moved up between the start (five minutes before the Fix) and the end (the Fix), it looks at the whole price path between start and end in detail. This means that it has access to information that the simple strategy does not. The other benefit is that no threshold has to be defined beforehand.

4.7 Deep learning

Deep learning has been successful in multiple tasks, and time series analysis is one of these areas. It has been used for classification and prediction, which is similar to this project. But it has as well been used for anomaly detection and segmentation [9].

There are several types of deep learning networks, but they all share similar components. A neural network can be defined as having linked layers where the output of one layer is the input of another layer, with a weight to determine the scaling of the output. Between these layers, a non-linear transformation of the output is performed. These non-linear transforms can be different functions, but three common functions are the *Sigmoid* activation, the *hyperbolic tan*, and the *Rectified Linear Unit*. These are defined in equation

(13) in the same order as mentioned

$$\begin{aligned} \sigma(x) &= \frac{1}{e^{-x}+1}, \\ tanh(x) &= \frac{2}{e^{-2x}+1} - 1, \\ ReLU(x) &= max(0, x). \end{aligned} \tag{13}$$

Training a neural network works by changing the weights between the layers. The goal is to find the right set of weights to be able to solve the task at hand. This is done by changing the weights in an iterative manner where the weights are changed to reduce a *loss function* which measures how far off the goal the network currently is. To be able to do this the gradient of the loss function is calculated and *gradient descent* is applied.

The layers in a network can be composed differently, one approach is to combine layers called *Convolutional* layers and *LSTM* layers, which is described later. The combination can be found in [11] and [20].

4.7.1 Convolutional Neural Networks

Convolutional Neural Networks (CNN) are commonly used for image processing to reduce the number of trainable parameters. This is important since training a network with too many parameters becomes slow, sometimes infeasible slow [9]. Therefore, instead of having unique weights between each neuron in two layers, the weights are shared between the layers. When the layers share the weights, it is the same operation as a convolution between the layers, hence the name Convolutional Neural Network. The convolutions become features of the input signal.

In this project, CNN is used to find features of our time series in an automated way. It is used to understand the short-term time components in our signal. Since CNN can be described as a convolution, we can describe its kernel. By changing the length of the kernel the amount of information of the signal that is composed of a feature changes. A larger kernel looks at a broader range of the signal, while a smaller kernel looks at just neighboring time points. Then the kernel slides across the signal such that the convolution is performed and the signal has been compressed into a new time series.

4.7.2 Recurrent nets and Long-short time memory (LSTM)

Recurrent nets are used to understand temporal information in a sequence. It can range from data such as videos, audio, text, or in our case financial data.

Recurrent neural networks (RNN) are constructed a bit differently from ordinary neural networks since they allow cycles in the computational graph [13]. The cycles have a delay which gives the network a sense of memory, in other words, it can use information about earlier time steps to understand the current time step. RNN can have problems when trying to incorporate information from time steps that happened a while ago due to a concept called *vanishing gradients*. Vanishing gradients says that when you multiply gradients that are less than one the multiplication tends to be very small. Therefore, when backpropagation is applied to that weight the weight does not get updated and the network has trouble learning.

To counter vanishing gradients Long-short term memory (LSTM) has been developed instead. LSTM uses memory cells instead which copy information from earlier time steps. In this project, knowledge about what happened earlier might dictate what happens later and thus LSTM has been chosen to be incorporated into the network.

4.7.3 Autoencoders

Autoencoders have been used for different tasks, from de-noising to anomaly detection of time series. In this project, it is used to compress multiple time series into fewer more descriptive time series. These new time series lays in a manifold defined by the weights of the encoder and is called the *latent space*. In the latent space, similar time series are put closer together to compress the information, and in other words, it is used as dimensional reduction which hopefully make the clustering of time series better.

An autoencoder works by having two parts, an encoder, and a decoder. The encoder compresses the information using a transformation and the decoder later decompresses it using an approximate inverse transformation. How these transformations should be done is up to the autoencoder. The user only specifies what the dimensions of the compressed space should be. The way to train an autoencoder is using *reconstruction loss*, in other words, how similar are the input and the output of the autoencoder. When the reconstruction loss of the autoencoder is low, the model has learned how to represent the data in a compressed way. The loss function is often the *mean squared error*.

4.7.4 Clustering layers and KL divergence

KMeans clustering is not the only way to cluster time series. Using deep learning, a neural network can also be used to separate time series. Instead of having a distance measure between time series, *Kullback–Leibler divergence* (KL divergence) can be used to cluster the time series.

In this project, the same method is used as in this paper [17] and the method is summarized below. To start, the probability that a signal z_i belongs to the cluster j with centroid w_j is given by equation (14), which is normalized according to Student's t-distribution kernel and siml is a similarity measurement. The similarity measurement is a correlation-based similarity measurement using the Pearson's correlation (COR) between the signal z_i and the centroid w_j. This is defined in equation (15)

$$q_{ij} = \frac{(1 + siml(z_i, w_j))^{-0.5}}{\sum_{j=1}^{k}(1 + siml(z_i, w_j))^{-0.5}}, \tag{14}$$

$$
\begin{aligned}
siml(z_i, w_j) &= \sqrt{2(1 - \rho(z_i, w_j))} \\
\rho(z_i, w_j) &= \frac{cov(z_i, w_j)}{\sigma_{z_i}\sigma_{w_j}}.
\end{aligned} \tag{15}
$$

To be able to measure the KL divergence, a reference distribution has to be defined. The reference distribution is defined in equation (16). The distribution is chosen in this way to increase the high confidence assignments

$$
\begin{aligned}
p_{ij} &= \frac{q_{ij}^2/f_j}{\sum_{j=1}^{k} q_{ij}^2/f_j} \\
f_j &= \sum_{i=1}^{n} q_{ij}.
\end{aligned} \tag{16}
$$

Using these two distributions, the KL divergence loss can be defined as equation (17) which derivative can be calculated

$$
loss = \sum_{i=1}^{n} \sum_{i=j}^{k} p_{ij} log \frac{p_{ij}}{q_{ij}}. \tag{17}
$$

When the loss function is defined, it is possible to incorporate it into the loss function of the autoencoder. By letting the autoencoder change its latent space to separate the clusters more but still maintaining the reconstruction error, we can create an autoencoder where signals from different clusters are separated more. The combination of the autoencoder and the KL divergence therefore hopefully separates signals which comes from different market dynamics.

One thing worth mentioning between using KL divergence and Euclidean distance is how much "control" you have over what you want the algorithm to cluster by. When using Euclidean distance, you know that the algorithm looks the distance between points in space. Maybe not which points, but at least the sum of points. This type of control does not exist when using a latent space and KL divergence. Any arbitrary coding can be used to separate it, from separating on the distance to separating on how volatile the signal is.

4.7.5 Grad-CAM

One benefit of clustering with deep learning is that the model can more easily be understood. When clustering with KMeans, we know which distance measure is used to calculate the distance between signals, but it is hard to see which part of the signal made it become clustered in its respective cluster. We don't know if it is something that happened early in the signal which made it go into its cluster, or if it is something that

Figure 6: Where the model looked to determine dog, Keras example from [8].

Figure 7: Where the model looked to determine cat, Keras example from[8].

happened at the end of the signal. Using deep learning, we can better understand this by looking at the weights of the network.

Grad-CAM [18] was a method invented to understand which parts in a picture were used to classify the picture. For example, in Figures 6 and 7 we can see where the model looked to classify the image as a dog and where it looked to classify it as a cat.

The same can be done on a signal. Grad-CAM backtracks and looks at the gradients of the CNN layer and can display wherein the signal the information that leads to the clustering are. Using this information, a heatmap can be created to see which parts of the signal contributed the most to the decision.

Understanding how the model works is important. If the model takes decisions based on attributes that are financially justified, it is easier to trust the model.

5 Methods

5.1 Data

The data used is internal data from SEB. Exactly what data and how it is used are not discussed for competitive reasons. Instead, high-level descriptions of the data is used to explain what has been used and not. The currency pairs investigated are GBP/USD and EUR/GBP and the years 2017–2020 are used. The data is represented as a tensor with shape (N,T,D) where N is the number of days, T is the length of the signal, and D is the number of features of the signal.

5.2 Features

The price that has been used is the mid from the median bid and offer from many trading venues. The median has been used instead of the mean since using the mean gave rise to negative spreads, which is not realistic. The features that have been used in the project are the price trend, some measurements of price impact, and some measure of trading activity. These are not discussed in too much detail for competitive reasons. However, we have to keep in mind that market impact is somewhat of an abstract concept and is not some evident feature that exists in an LOB by itself. Market impact comes from a trade that moves the market and is only evident after it has happened. Therefore, we have to handcraft features that more or less measures something that could correlate or describe potential market impact from a trade before it happens.

Different types of scaling have been performed for different tasks in the project. When clustering on price paths, a mean-variance scaling has been done on each signal, and not the same scaling on all. To be more precise, the mean of the individual signal is subtracted from the signal and then the signal is divided by the standard deviation of itself. This is done since we are more interested in the general shape of the signal and not the magnitude for some of the unsupervised tasks. An increase in the price is an increase regardless of how much it has increased.

For supervised learning, the same approach can be misleading. When we only look at half of the signal, the magnitude can play an important part. Therefore, for each signal, the mean of the k last signals is subtracted and then the signal is divided by the mean of the k last signals' standard deviations. Then after this, we scale all the new signals by removing the mean of all new signals and dividing by the mean standard deviation of all new signals. We do this to dynamically scale the signals to make them more static without losing the magnitude.

In all Figures, the scaling has been reversed such that the original signals are the ones plotted.

5.3 Look-Ahead Bias

One concept that should be discussed and clarified is the concept of Look-Ahead Bias. Look-Ahead Bias happens when the model "accidentally" gets information from the future, which can help it make better predictions. This sounds obvious, however, it can be quite hard to realize that your model contains Look-Ahead Bias.

One common way to include Look-Ahead Bias is to do a random split of train and test data for a k-fold. If we train in the year 2018 and we test in the year 2017, then some information about the end of the year 2017 can be contained in the information early in 2018. Therefore we have to do a k-fold where we train on the first fold and then test on the second. Then we train on the first and second fold and test on the third. If we don't do this, we risk having Look-Ahead Bias.

Another common way is by scaling the data. In the unsupervised section, we scale the data according to the whole signal of the day. If we would do that for supervised learning, and then cut half of the signal and predict the second half, we would introduce Look-Ahead Bias since some of the information of the second half is now contained in the scaling of the first half.

Similarly, if we scale on the whole data set, but we do it before we do the K-folds, Look-Ahead Bias exists since early years have information about later years in its scaling.

Lastly, when we scale the data on the last k days, it is important to not include the current day, since that introduce Look-Ahead Bias. If we keep the current day, some of the information about the end of that day exists in the scaling.

One might think the information contained is so small that it should not matter, however that is not the case. We can fool our self by artificially improving the performance of a model by just making one of the mistakes stated above.

5.4 Python libraries

Python has been the main language used in the project. The libraries Keras, SKlearn, and TSlearn have been used to perform the machine learning tasks.

5.4.1 Temporal Convolution Neural Network (TCNN)

The network used in this project is called a Temporal Convolution Neural Network and is similar to the network expressed in this paper [17]. This network is a composition of multiple different types of layers where each layer has its benefits. The first layer is a convolutional layer that is used to find high-level features in the signal. The second layer is an LSTM layer. The LSTM layer is used to further understand the temporal aspect of the signal and how earlier parts of the signal intertwine with later parts. We can say that the convolutional layer is used to understand short-term relationships of the signal and the LSTM layer is used to understand long-term relationships. This is the main part of the model, which is called the encoder.

Now depending on the task at hand, different layers are added to the model. When doing unsupervised learning, we want to add a decoder to create a full autoencoder. This is done to be able to encode the time series to a more appropriate latent space. Between the encoder and decoder, a clustering layer is added as well. This layer is later trained to change the latent representation such that clusters separate more.

When we want to do supervised learning, instead of attaching a decoder and a clustering layer, we instead flatten the latent space into a vector and attach a fully connected layer to act as a classifier. Since we only have two classes the output is a scalar that has been passed through a Sigmoid activation such that it is between 0 and 1.

5.5 Training the TCNN

The KNNs are easy to train. However, the TCNN trains in different ways depending on what task it tries to do. When training for clustering, the autoencoder is first trained for a number of epochs. The training is done with reconstruction loss as loss function, which is just the mean squared error between the signals for each time step. When the autoencoder can encode and decode well, the network continues to train but with the KL divergence loss as a loss function. When the network reduces the KL divergence loss, the reconstruction loss typically increases. Therefore we can train on both losses at the same time and not let any of them be to high.

When training for classification, binary cross-entropy is used as a loss function and the whole network is trained at once.

6 Unsupervised Learning

6.1 Post-Fix unsupervised price trend clustering

To start with the clustering, it can be good to understand one of the more basic clusterings in this project. Using KMeans we can try to cluster time series that either trend up or trend down. By looking at the trend after the Fix, we can determine if the price went up or down during a five-minute window after the Fix.

When clustering with two clusters in this way these two clusters emerge, where the mean of the clusters can be seen in Figure 8 where the shaded area is the 95%:th confidence interval of the mean is in this interval. In general, they seem to diverge from each other, but looking at Figure 9 we can see that there are still some overlap between them even at the end. In Figure 9 the shaded area represents one standard deviation.

By only having two labels, we force some price trends to be in either one of them. This might not be ideal for every day since some days the price trend might be flat. A flat price trend is forced to be either labeled as an increase or a decrease.

This was not only for demonstration. These clusters are later used as labels for the supervised learning models. From here on, these clusters will be referred to as "post-Fix clusters".

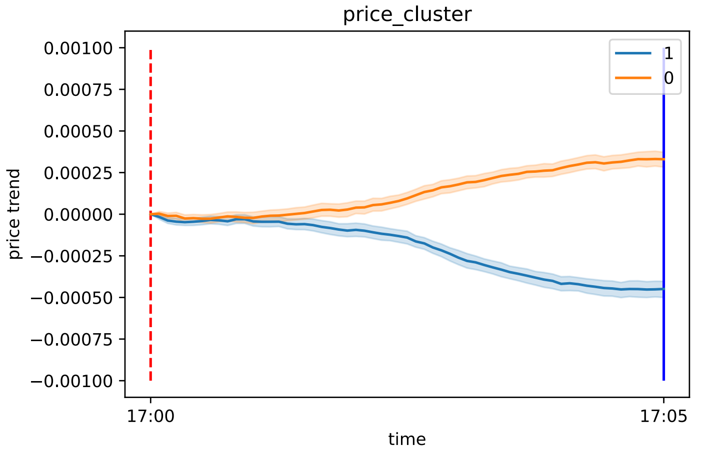

Figure 8: The two clusters emerging directly after the Fix, the shaded area represent the 95%:th confidence interval of the mean.

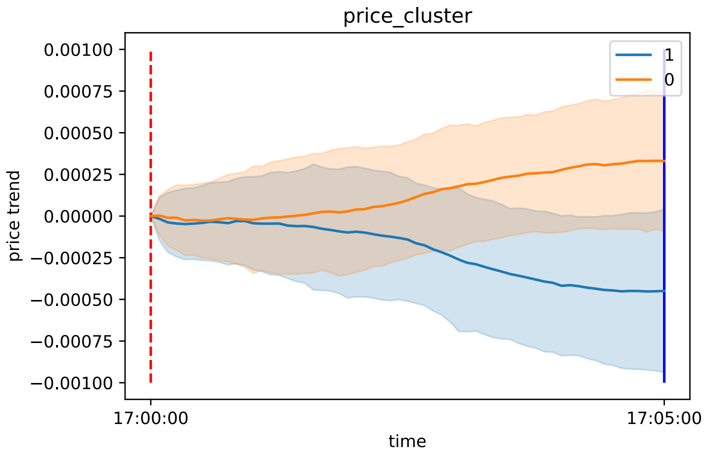

Figure 9: The two clusters emerging directly after the Fix, shaded area represent the first standard deviation.

6.2 Theoretical model clustering

The model described in section 4.1 has been programmed in Python and can be used to simulate how the price path should evolve according to the model. Running this theoretical simulation for different model parameters yield different results. One way to understand these results is to cluster them into clusters and look at the mean price path. Let's start by clustering a completely unaffected Geometric Brownian Motion. Unaffected in this case means that trades have zero temporary and zero permanent market impact and therefore the price is unaffected by the traders. The results are presented in two different kinds of Figures. Figure 10 is a plot of the price trend where the start is at the first trading round and the end is the last trading round. There are ten trading rounds, the fifth round is the Fix and between each round ten time steps are taken. Notice that the plot is not presenting the price, it is the price minus the price at the beginning, and therefore the price trend. Figure 11 is a *kernel density estimate* (KDE) plot showing the distribution of the trend at the denoted fix time.

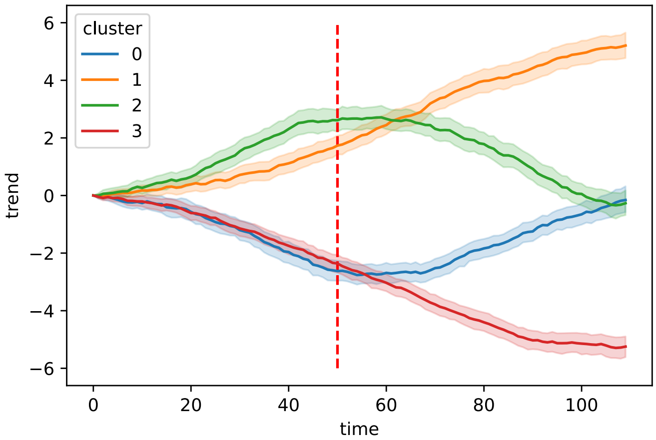

Figure 10: Clustering on a Geometric Brownian Motion, the shaded area represent the 95%:th confidence interval of the mean.

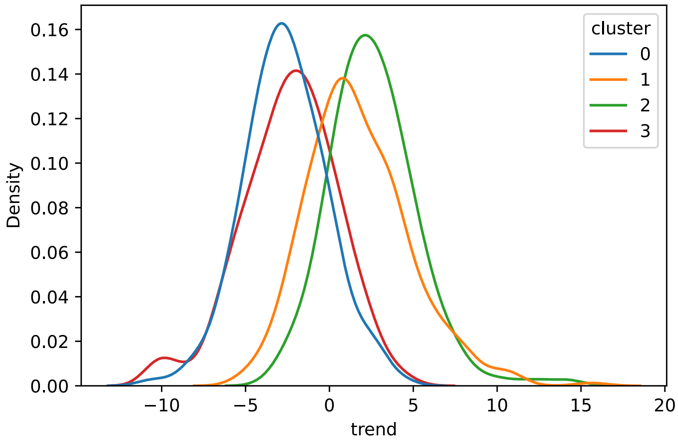

Figure 11: KDE plot of the trend on a Geometric Brownian Motion at the Fix time.

We can see that the clustering separates the clusters at the midpoint which is at time step 55 out of 110, while the red line crosses time step 50 which is the designated Fix time in our model (we trade every tenth time step). The four clusters we get are a trend up/down, and a mean reversion starting with a trend up/down. Therefore, without any influence of trading, we find these four clusters directly by clustering on a Geometric Brownian Motion.

Now we implement market impact into our model defined by equation 1. We let the market either have a temporary market impact (α) of value 0.1 or 1, both as probable. We always have a permanent market impact (β) of 0.1. These impact values are then be perturbed by Gaussian noise with a mean of zero and a standard deviation of 0.005. The volatility (σ) was 0.1 to control the noise in the price evolution and the trading strategy is the optimal trading strategy given the market impacts.

Figure 12 displays the mean of the four new clusters and Figure 13 shows the KDE plot for the trend at the Fix time. We can see that the four clusters differ between the GBM and the market impact scenario. When we overtrade, we trade larger volumes before the Fix to later buy back these volumes after the Fix. This can be seen by the rapid change in price trend initially before the Fix. The same thing can be said about the KDE plot. For the GBM, the reversion and the continuation of price trend have about the same price trend initially up toward the Fix and are not well separated at the Fix, while for the market impact scenario the price trend at the Fix is separated well.

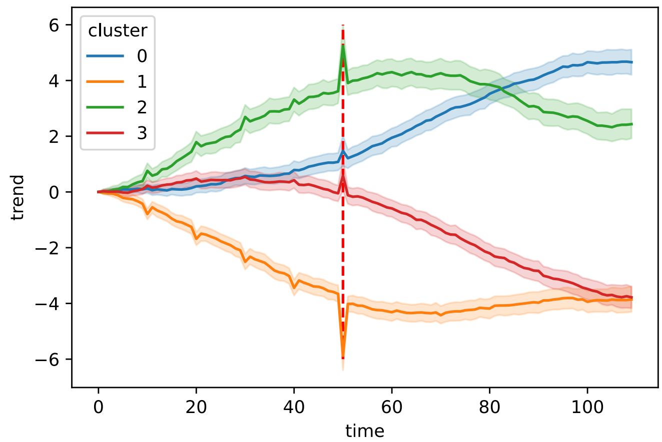

Figure 12: Clustering on the theoretical model's simulation, the shaded area represent the 95%:th confidence interval of the mean.

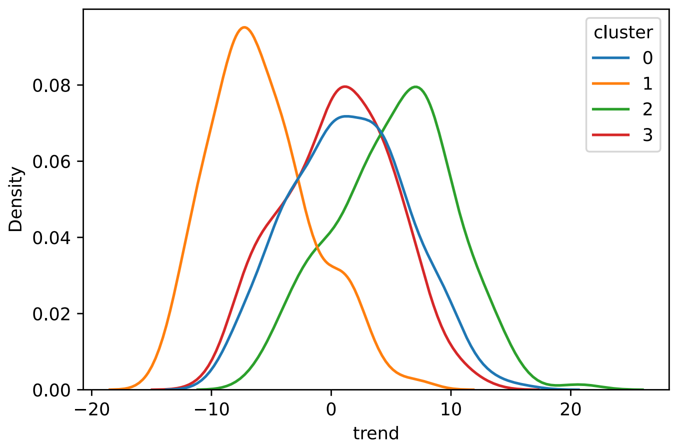

Figure 13: KDE plot of the trend on the banging simulation at the Fix time.

In this model, the price path depends both on the temporary market impact (since permanent was locked), and if we initially want to sell or buy our position. Let's see how well these parameters cluster. In Figure 14 we can see that the underlying variables do not get clustered perfectly. Since the price signal is noisy some signals get assigned to the "wrong" cluster. Therefore, even in this controlled simulation, the clustering is not perfect. A perfect clustering would be if cluster 1 and 2 would belong to a low temporary market impact and cluster 0 and 4 would belong to a high temporary market impact.

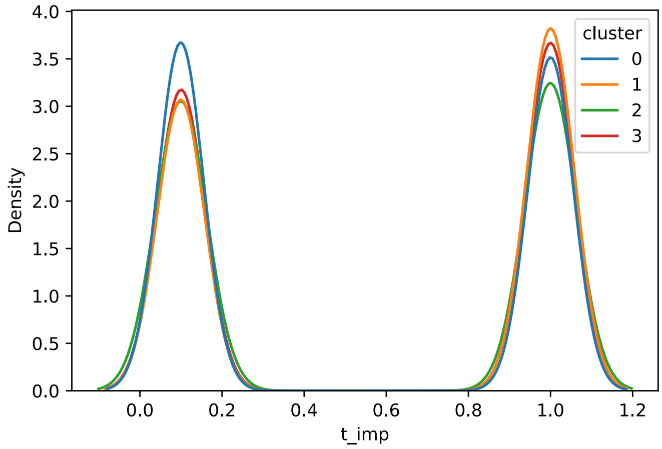

Figure 14: How temporary market impact gets clustered.

Since the variables do not get perfectly clustered, we can instead try to cluster on these variables and see how that clusters the price. In Figure 15 we can see this. We see that when clustering on these variables the price paths get clustered differently.

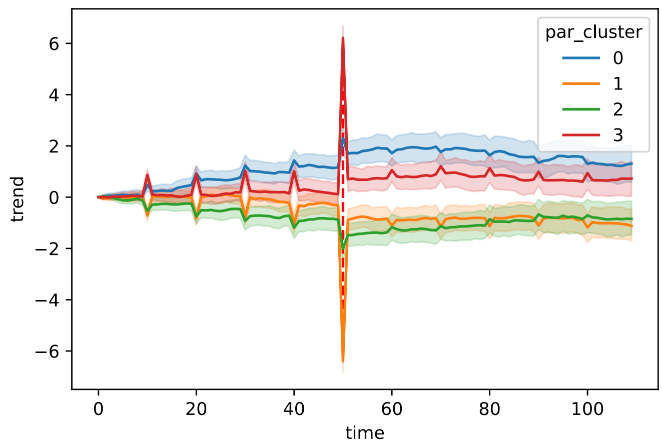

Figure 15: How the price is clustered when clustering on the three core variables, the shaded area represent the 95%:th confidence interval of the mean.

6.3 Real price data clustering

Evans in [6] found that price dynamics had a negative correlation between pre-Fix trends and post-Fix trends. Therefore, on more days than not, but not all days, the price revert after the Fix. However, some days, the price trend should continue through the Fix. We have seen that clustering a GBM gives these four clusters but that overtrading and

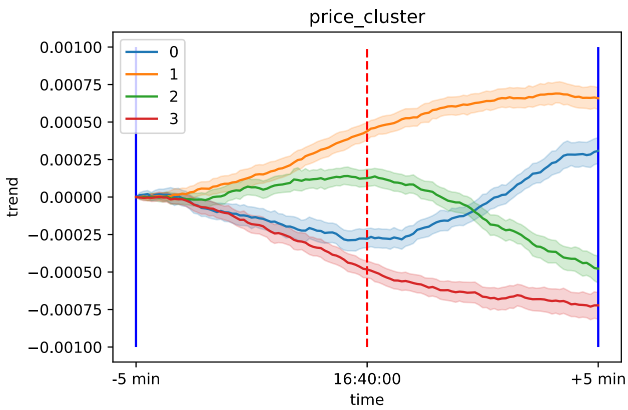

Figure 16: Clustering of market data at 16:40:00, the shaded area represent the 95%:th confidence interval of the mean.

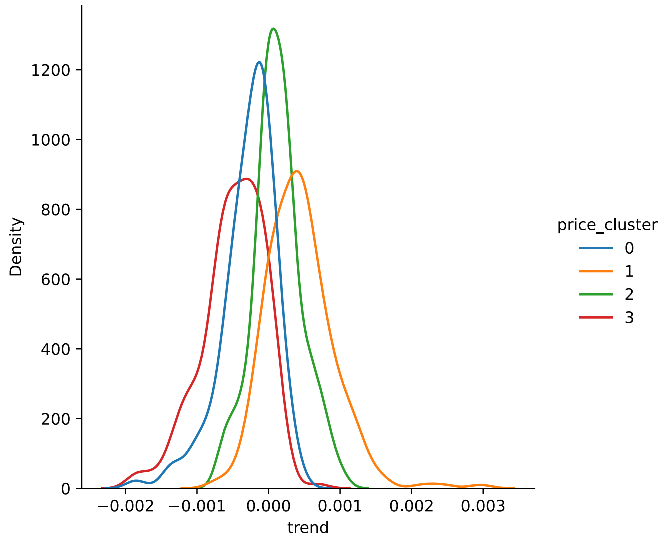

Figure 17: Clustering of market data at 16:40:00.

banging separate these clusters more. Can the same be seen in market data? Do the clusters during the Fix separate more than at other times? And if so, does it resemble the price paths from the simulated market? To not get confused, the plots have 17:00 as the Fix time since the project is done with local time and not London time.

Let's start by clustering on time windows that do not overlap with the Fix. Figure 16 to 24 show the clusters when clustering in a ten-minute window with 16:40, 16:45, 17:15, and 17:20 as midpoints. We see that they all differ from each other. In Figures 16 and 24 we have two clusters of early trends and two clusters of late trends, one in each direction. In Figure 22 we see something similar to to the GBM in Figure 10. Figures 18 and 20 we see something similar to 12, especially in 20 which is clustered over the Fix. This indicates that some strategy is deployed at the Fix. However, we need to keep in mind that we see a slightly less strong signal of equal kind at 16:45.

Another way to look at this is to look at the trend distribution at the mid-time for each of these clusters and see how much they separate. This can be seen in Figures 17 to 21. We can see that there is a lot of overlap between the clusters, which makes it hard to separate the signal from noise during just one day. This will play a greater role in later discussions.

What do these findings mean? In general, the price trend differs between times but during the Fix, the dynamics get more predictable. It seems that the information in the price trend before the Fix carries more information about what happens after the "mid-time" than the price trend does at other times that are not the Fix. However, since the distributions still overlap the signal is covered in a lot of noise.

In Table 1 we can see how many time series in each cluster belong to the post-Fix cluster as well. For example, for the cluster zero, 81.82% of the signals also belong to the post-cluster zero, which is the upward-trending cluster. Put in other words, 81.82% of

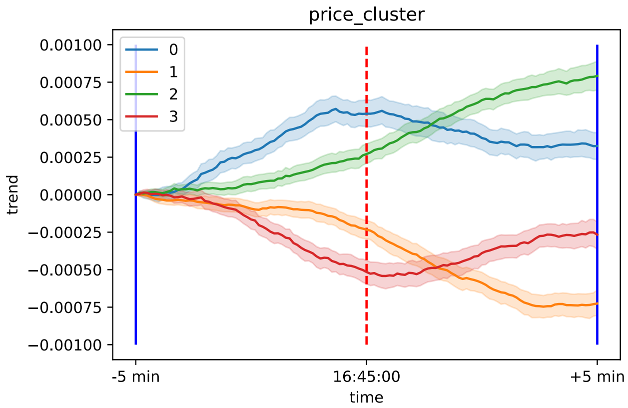

Figure 18: Clustering of market data at 16:45:00, the shaded area represent the 95%:th confidence interval of the mean.

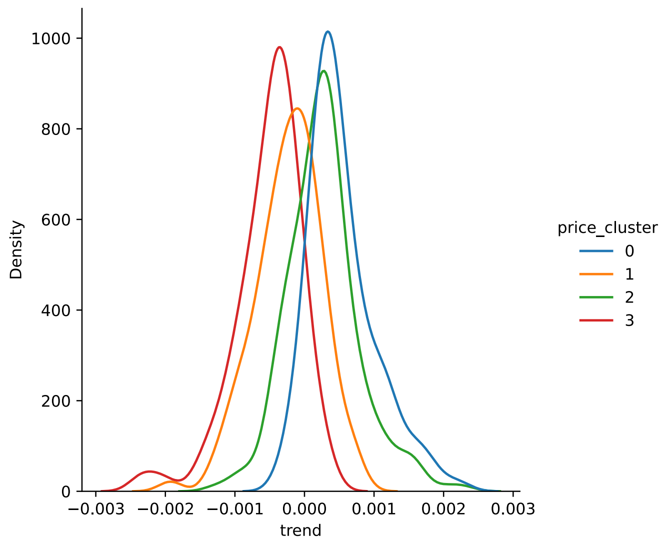

Figure 19: Clustering of market data at 16:45:00.

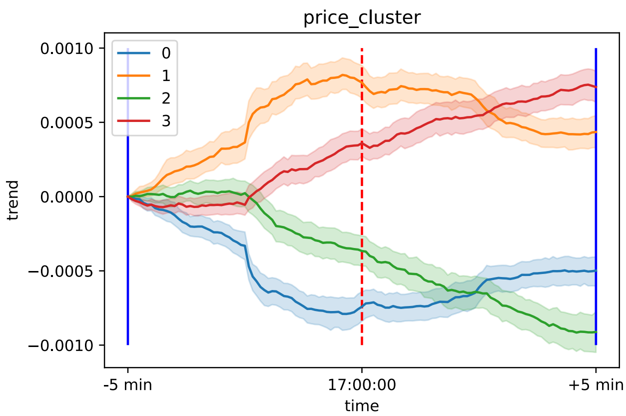

Figure 20: Clustering of market data at 17:00:00, the shaded area represent the 95%:th confidence interval of the mean.

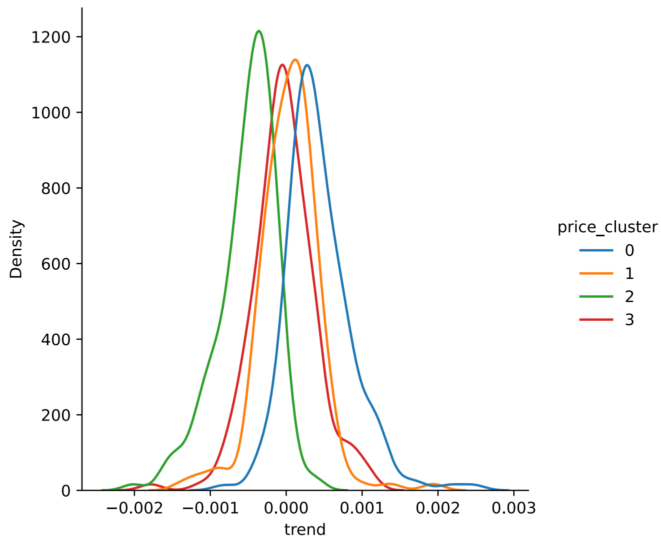

Figure 21: Clustering of market data at 17:00:00.

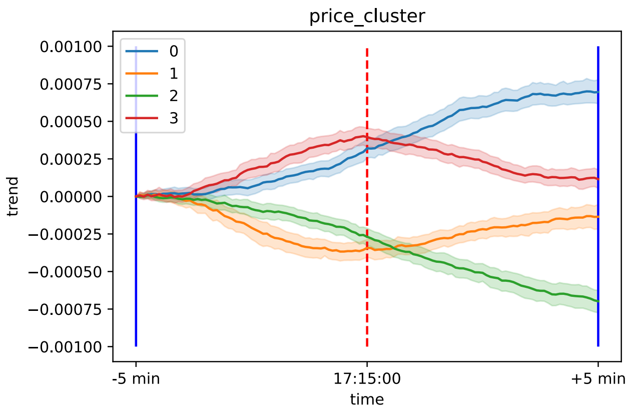

Figure 22: Clustering of market data at 17:15:00, the shaded area represent the 95%:th confidence interval of the mean.

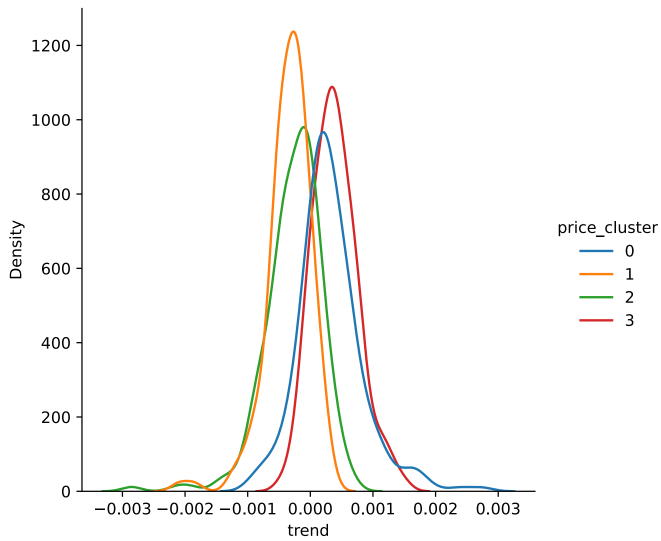

Figure 23: Clustering of market data at 17:15:00.

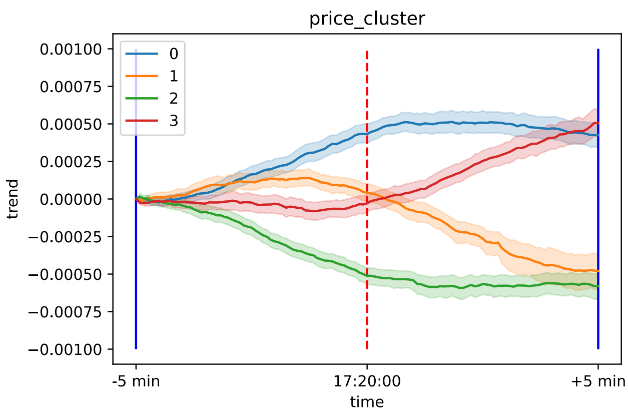

Figure 24: Clustering of market data at 17:20:00, the shaded area represent the 95%:th confidence interval of the mean.

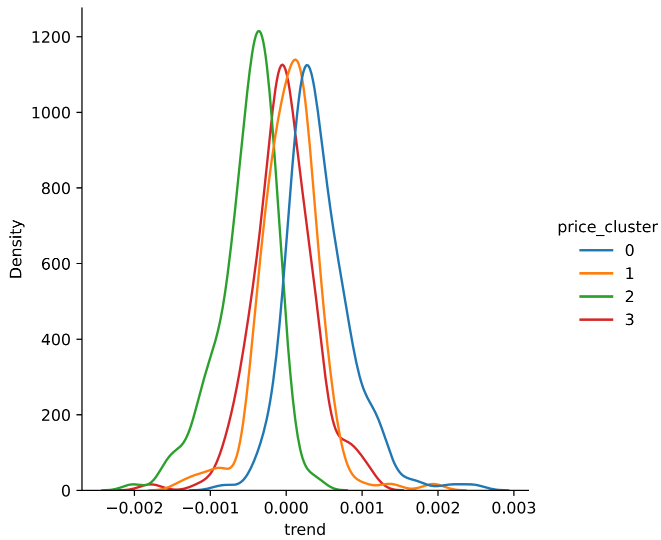

Figure 25: Clustering of market data at 17:20:00.

the signals in cluster zero trend upwards after the fix.

$Cluster$	$Post-cluster$	$Percentage$
0	0	81.82%
0	1	18.18%
1	0	16.48%
1	1	83.52%
2	0	13.37%
2	1	86.63%
3	0	84.00%
3	1	16.00%

Table 1: The percentage of being in the post-cluster for each cluster.

6.4 Grad-CAM on the unsupervised price trend

In Figure 20 we can see the clustering of price trends around the Fix. We can see that the model separates the clusters quite well, but we do not on what grounds it makes its decisions. Where in the signal does it put the emphasis? Instead of clustering using KMeans, let's cluster with our neural network. In Figure 26 we can see that both information before the Fix and after the Fix is used to make the decision. However, from this information alone it is not clear whether the information before the Fix can solely be used.

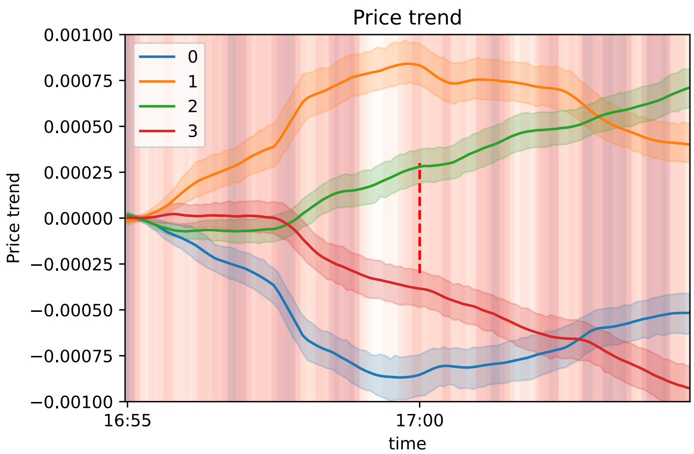

Figure 26: Clustering on market data at 17:00:00 with Grad-CAM as background. Dark red means more importance.

6.5 Unsupervised LOB clustering

One way to understand the dynamics around the Fix is to just look at the price, which we have done in the previous section. Another way is to look at the LOB and how it evolves during the Fix. Since the LOB contains more information than just the mid, can it be possible to find clusters in the LOB that correspond to different price dynamics? This is similar to what was done at the end of section 6.2.

The theoretical model uses information about the market impact and trading to understand what happens during the Fix, so preferably, the features that are used in our LOB should be similar. The features are time series that in some way correspond to or capture one of the attributes of the model. These have been aggregated over multiple market places to capture an overall dynamic of the market instead of detailed dynamics between markets.

6.5.1 Clustering on LOB, multivariate KMeans

These features have been defined using heuristics and are not perfect measures, since there is no "market impact"-features in an LOB, in contrast to the mathematical model. The clustering results can be found in Figures 27 to Figure 31.

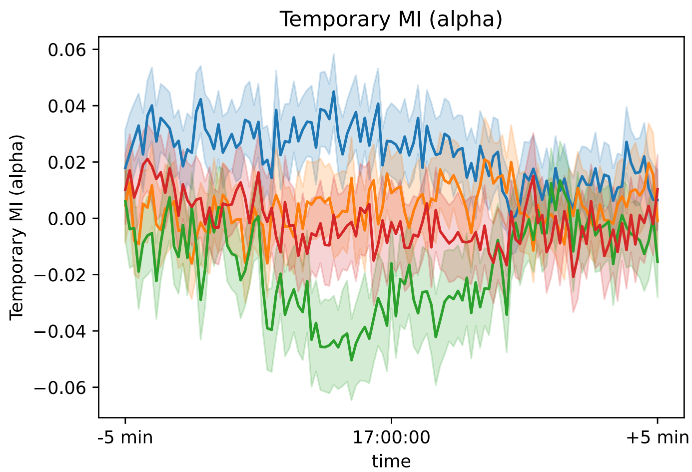

Figure 27: The temporary market impact feature when clustering on market impact and trading, the shaded area represent the 95%:th confidence interval of the mean.

From Figure 31 we can see that the clusters in the LOB could separate the price trends as well, without actually looking at the price. Some of the dynamics seen from price clustering are also visible here, but the negative mean reversion is not present in this clustering. The features are separated well around the Fix.

The question is, does the LOB contain this amount of information outside of the Fix as well? The short answer is no. Doing the same analysis on LOB with a mid-time of 16:45 we get the results in Figure 32. Here the price does not get clustered at all. This implies that 16:45 and 17:00 (the Fix) are different even though Figures 18 and 20 have four similar clusters. The same can be said about the other times as well.

This gives indications that the LOB contains more information about the price movement during the Fix than at other times of the day.

In Table 2 we can see how many are both in the cluster (the cluster during the fix) and in the post-Fix cluster. Preferably, we would like to see each cluster separate from

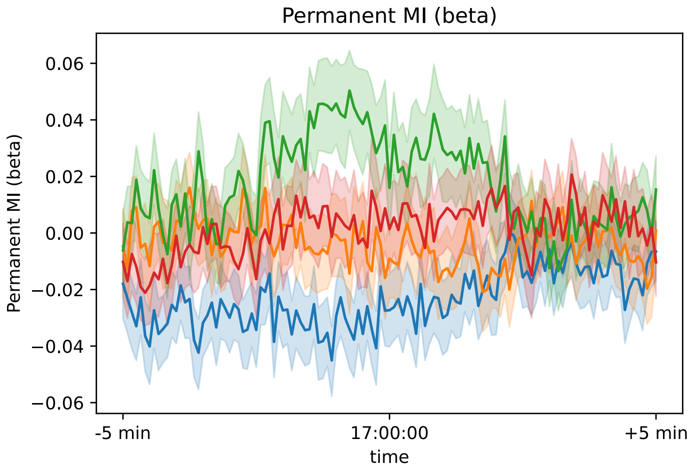

Figure 28: The permanent market impact feature when clustering on market impact and trading, the shaded area represent the 95%:th confidence interval of the mean.

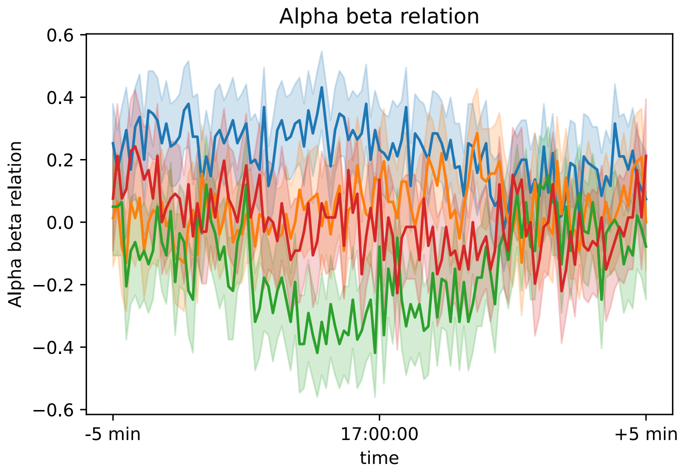

Figure 29: The "alpha-beta" feature when clustering on market impact and trading, the shaded area represent the 95%:th confidence interval of the mean.

the post-Fix clusters as much as possible. Therefore, this is one way to determine how successful the clustering is. We can see that we are not as good as the price clustering (Table 1), which is expected. However, it can still separate the clusters a bit.

6.6 Autoencoders and Latent Space

6.6.1 The autoencoder

Multivariate KMeans Time Series clustering works well in separating clusters of multiple individual time series, however, this is not exactly what we want. Instead of clustering multiple time series, we want to cluster market dynamics. However, we do not have a single "market dynamic" time series. One way to try to create this special time series is by combining non-linear combinations of time series to create one or more high-level market dynamic time series and then cluster these new time series.

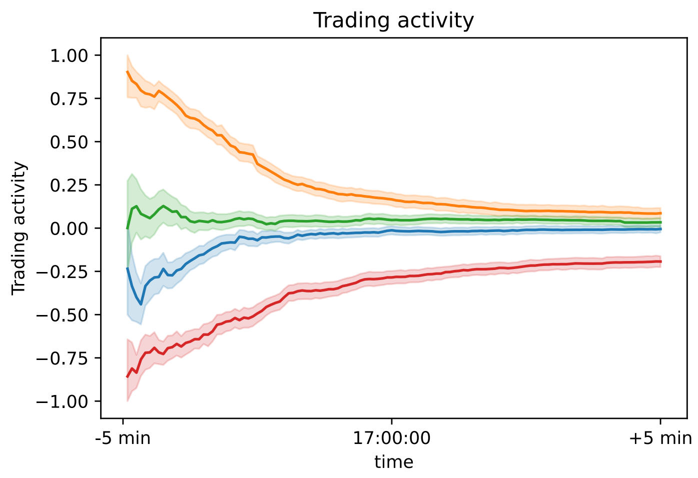

Figure 30: The trading feature when clustering on market impact and trading, the shaded area represent the 95%:th confidence interval of the mean.

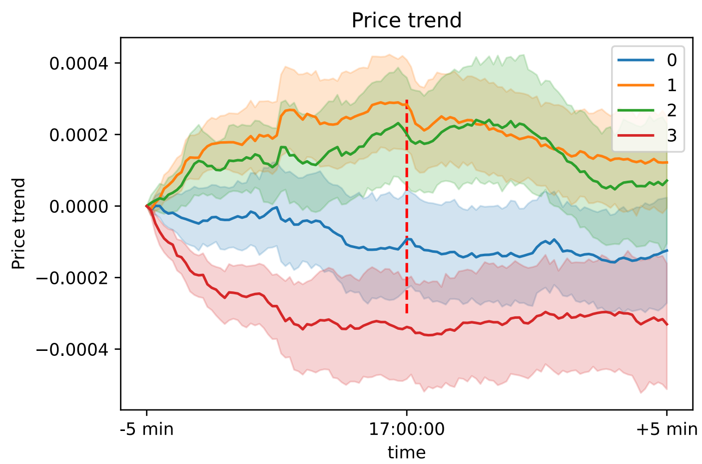

Figure 31: The price trend when clustering on market impact and trading, the shaded area represent the 95%:th confidence interval of the mean.

One way to do this is to use an autoencoder to encode the dynamics from the time series into fewer dimensions. The autoencoder is modeled as a deep neural network with multiple different layers to handle different dynamics of the signals and is structured as the network in section 5.4.1.

In the middle of the autoencoder, the "bottleneck" layer lies, which we call the latent space. This is the layer where our new encoder high-level market dynamic time series lives. These are the signals we later want to cluster on. To be able to create better clusters, it is preferable to have a latent space where clusters are separated as much as possible according to some measurement. This can be done by adding a clustering layer in the latent space which is trained to separate clusters. The measurement is the Kullback–Leibler divergence which measures the difference between two distributions, as mentioned before.

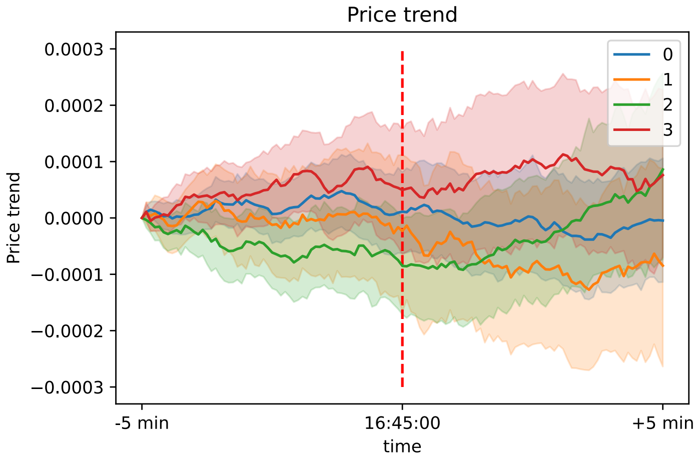

Figure 32: The price trend when clustering on market impact and trading at 16:45, the shaded area represent the 95%:th confidence interval of the mean.

$Cluster$	$Post-cluster$	$Percentage$
0	0	50.00%
0	1	50.00%
1	0	44.81%
1	1	55.20%
2	0	36.17%
2	1	63.83%
3	0	54.55%
3	1	45.45%

Table 2: The percentage of being in the post-cluster for each cluster.

This layer is trained iteratively in a similar way as KMeans is done iteratively. First, the layer is given a cluster created by KMeans to initialize the network, then it trains to change the latent representation such that these clusters separate more. Then it creates new clusters and separate these as much as possible. This is done until the new cluster is the same as the old cluster and no change has occurred, or when a maximum number of iterations has been done.

The network has twenty filters, kernel size of two, strides of one, and pooling of two in the first layer. There are then two LSTM layers where after the last layer the signals have been compressed to just one signal. Then the process is reversed using the same network structure but in the opposite sequence.

Now, after this step, we have a latent space that can capture the market dynamics. Clustering in this space instead and plotting the price curves accordingly, we can see in Figure 33 that clusters emerge in price as well. It seems to be slightly better at finding the dynamics we saw in the price clustering and from Table 3 it seems to perform better at separating the post-Fix clusters in each cluster.

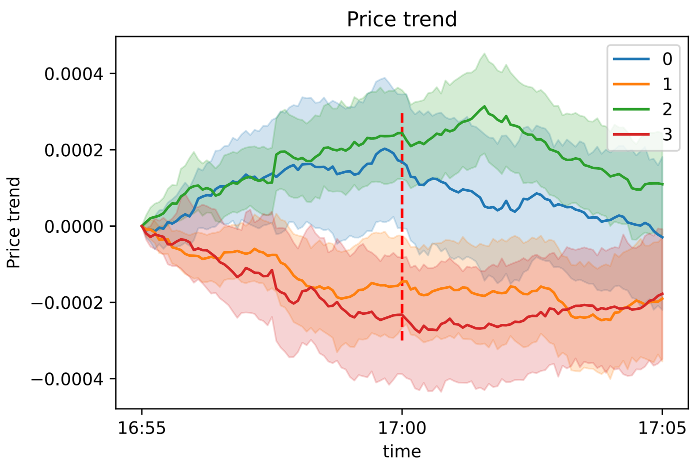

Figure 33: The price trend when clustering in the latent space, the shaded area represent the 95%:th confidence interval of the mean.

$Cluster$	$Post-cluster$	$Percentage$
0	0	44.54%
0	1	55.46%
1	0	45.57%
1	1	54.43%
2	0	38.34%
2	1	61.66%
3	0	60.00%
3	1	40.14%

Table 3: The percentage of being in the post-cluster for each cluster.

6.6.2 Cluster on price and find clusters in the LOB

We can try to reverse the process, can we cluster the price and see if the LOB gets clustered?

When clustering on price, features from the LOB cluster as well. In Figure 34 and Figure 37 we can see some weak clusters emerge, the variables are not perfectly clustered. To cluster well is not expected since the same trouble could be found in the theoretical model as well.

7 Supervised learning

7.1 Supervised learning

Classification is a sub-field of supervised learning where data is used to classify each sample with a label. In our case, we want to use market information before 17:00 to classify if we think the price increase/decrease after 17:00. The labels we acquired in section 6.1

We look at the case where we want to buy and the classes we want to predict are "trade

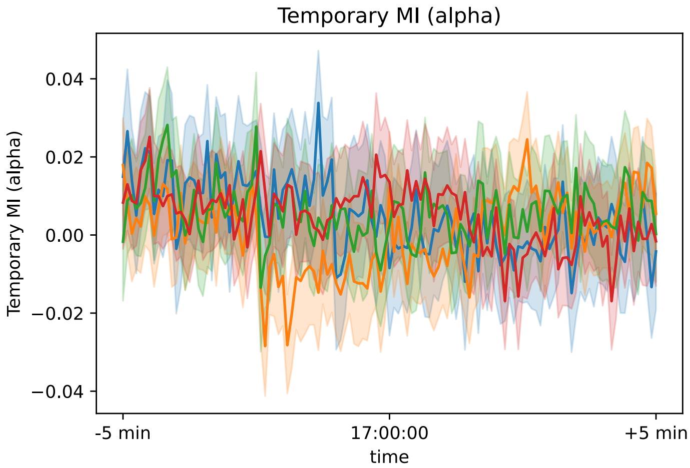

Figure 34: The temporary market impact feature when clustering on the price trend, the shaded area represent the 95%:th confidence interval of the mean.

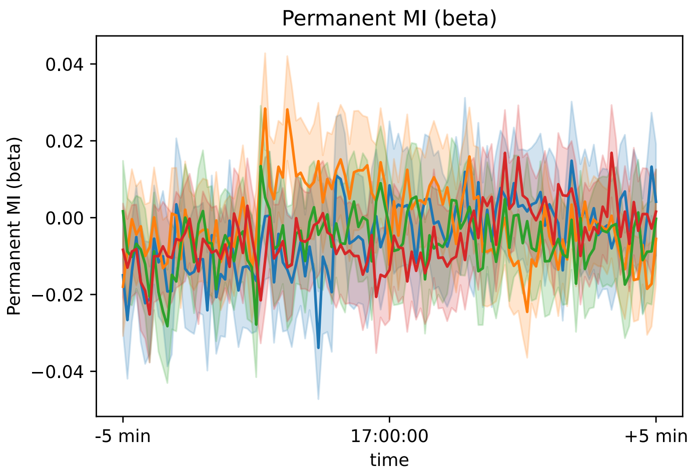

Figure 35: The permanent market impact feature when clustering on the price trend, the shaded area represent the 95%:th confidence interval of the mean.

now" and "wait". If we expect the price to increase, we want to trade now, and if we expect the price to decrease, we want to buy later. Since waiting to trade increases the risk, we can argue that it is worse to be wrong when waiting than to be wrong when trading directly. Or put in other words, we want to be more sure when we wait than when we trade directly. This is because we as a trader want to be risk-averse.

Because of this imbalance in classification importance we do not only want to look at accuracy, we also want to look at recall and precision. Precision might be more important than recall, in this case, however, we also want to wait as many times as is relevant to wait. We denote the waiting class as the positive class such that recall and precision make sense.

Both the KNN models and the deep neural networks (TCNN) seemed to be able to identify different dynamics of the Fix when seeing the whole signal, both before and after

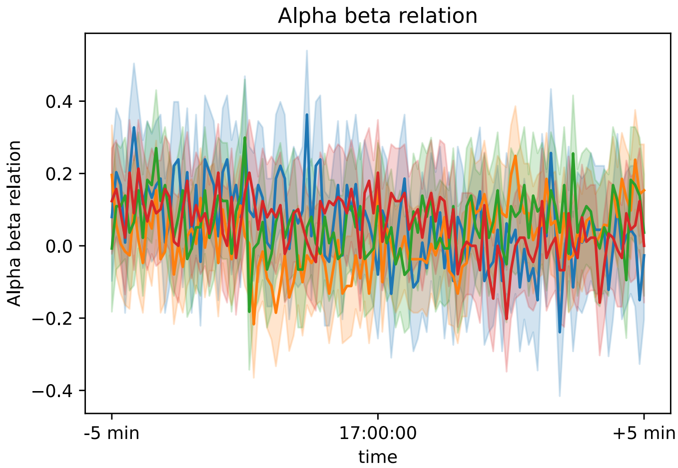

Figure 36: The "alpha-beta" feature when clustering on the price trend, the shaded area represent the 95%:th confidence interval of the mean.

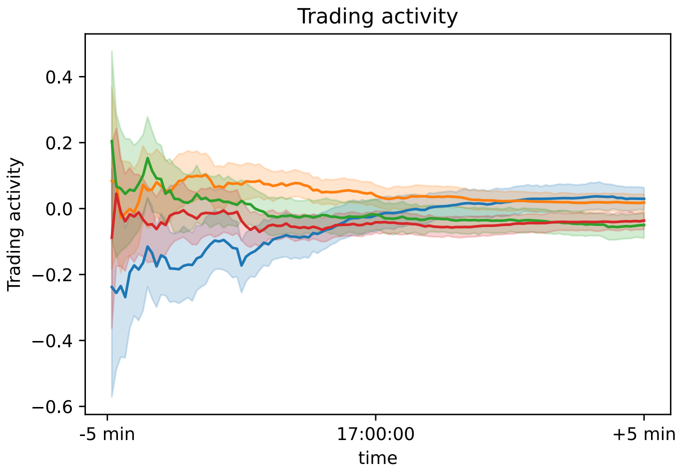

Figure 37: The trading feature when clustering on the price trend, the shaded area represent the 95%:th confidence interval of the mean.

the Fix. Could they also understand the dynamics by only looking before the Fix? Four models are be tested, a univariate KNN and a univariate TCNN which only looks at the price trend. Then a multivariate KNN and a multivariate TCNN are tested which look at the price, market impact, and trading activity. At the end of section 4.2, we saw that the price had information about the market impact. Let's see if that is enough or if the other features help.

Additional models are also investigated. A simple model that always predicts one label is used for comparison, as well as a model that predicts labels at random. The third model to compare to is a model which looks at the price trend at 17:00 (local time Stockholm, 16:00 London) and if the trend (in absolute value) is above a threshold, a mean reversion is predicted. This is the Simple Classifier that was mentioned in section 4.4.

7.2 Classification results

Figures 38 to Figure 40 shows the result of our different models. Remember, the different models (that are not using machine learning) are the Random Classifier (RC), the one-label classifier (DC) and the Simple Classifier (SC). Accuracy, precision, and recall are shown. The best performing model for accuracy is the multivariate KNN model and the univariate KNN model. The simple classifier and the multivariate TCNN perform equally well and then the univariate TCNN performs slightly worse. However, they all beat the random classifier and the one-label classifier. For precision, the multivariate TCNN seems to perform the best but for recall, the multivariate KNN wins (excluding the DC).

For Accuracy, it does seem like the price might be enough to make a good prediction. Adding the other variables only gave a slight increase for the TCNN. However, for precision and recall that seemed to help out more, especially in the recall. Therefore it seems like the multivariate models might be better when we want to reduce the risk of waiting. But in general, all the learning methods had an accuracy around 60% which is better than chance.

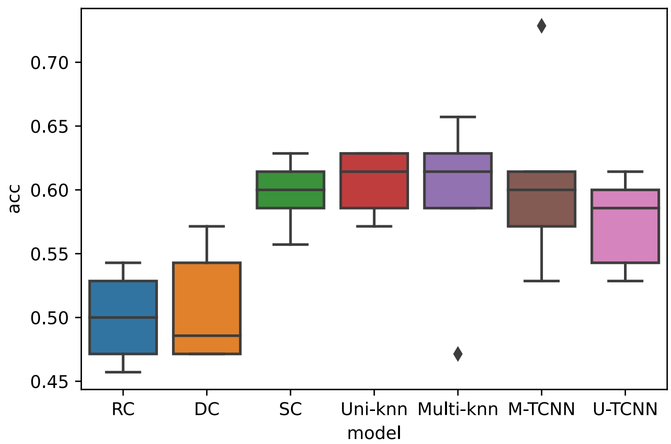

Figure 38: The different models result in accuracy.

7.3 Supervised Grad-CAM

In Figures 38 to 40, we can see that the classifiers seem to be able to predict what happens after the Fix quite well. However, can we understand what it bases its decisions on? If we assume that the KNN and TCNN use the same information to classify, we can use Grad-CAM on the TCNN to understand these classifiers. According to Figure 41, we can see that the importance of the information increases as we approach the Fix. What happens right before the Fix seems to carry more information than what happens five minutes before the Fix.

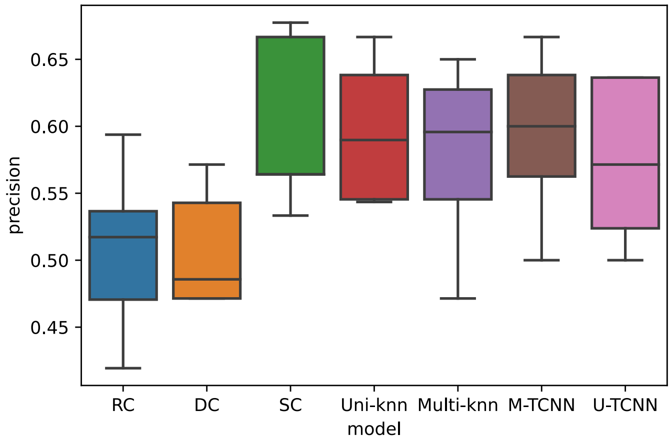

Figure 39: The different models result in precision.

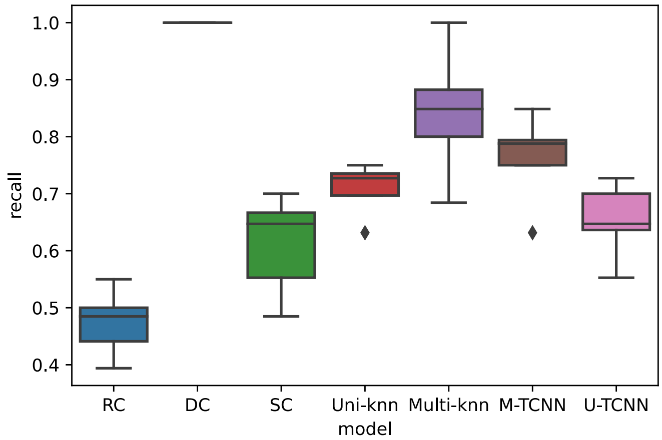

Figure 40: The different models result in recall.

8 Trading strategy

A classification task is one way to determine the effectiveness of the model, however, it does not cover the financial aspect completely. Therefore, the *Profits and Loss* (PnL) and *Sharpe ratio* of the models are measured by back-testing them for the year 2020. So the models are trained in the two years before and then be evaluated in 2020. We assume we always need to buy and if we predict an increase in price after the Fix we will buy directly at the price at 17:00. If we predict a decrease in price we will trade at the price at 17:05. Here we assume that we get exactly that price. The PnL is calculated as the difference in price between the Fix and the time we decided to trade at. The unit we want to trade is always 1 unit.

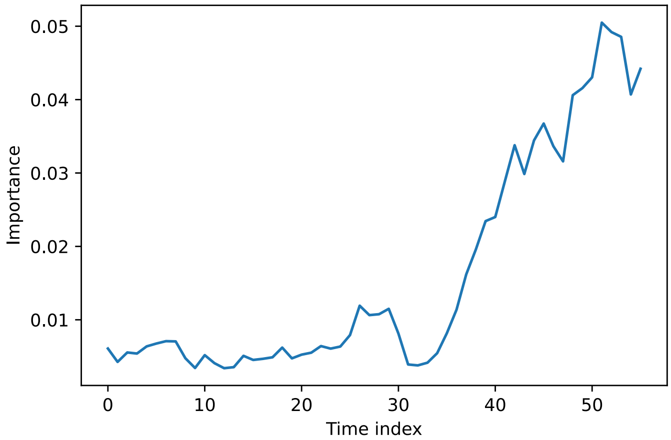

Figure 41: How important the TCNN thinks the information in time carries, using a 5-point mean for smoothing. Time index 60 is on the Fix.

8.1 Trading results GBP/USD

The classification metrics can give us a priori guess on how well it trades, but it does not explain everything. Some days it is more important to be correct than others, so to determine which model is the best we have to back-test the strategies. This is done in the year 2020.

In Figure 42 and Figure 43 we can see how well the models did during the year. All models, except the random algo, performed similarly at the end of the year. While the models performed equally well they were more stable than the "always-wait" strategy. This can also be seen by looking at the Sharpe ratio in Figure 44. The Sharpe ratio is the ratio between expected profits and the standard deviation of profits, where a high Sharpe ratio is preferable. In Figure 45 we can see the fraction of times the algorithm decided to wait. We can see that the algorithms decided to wait around an equal amount of times.

Another common algorithm is the TWAP. The TWAP split its whole trade into smaller buckets of trades and spread them out over a time window. In this case, it is used to try to pay the average price over a time window. The earnings from placing a TWAP with a two-minute time window over the Fix is presented in Figure 46. We can see that the earnings are a lot smaller than the other algorithms, however, the Sharpe ratio is in the same ballpark as the others at 0.125. The Sharpe ratio is quite high since the variance of earnings is low, which compensates for the smaller expected earnings.

8.2 Trading results EUR/GBP

The same back-testing can be done for another currency pair. In Figure 47 and Figure 48 we can see how well the models did during the year. Here only two of the models could go through the year without losses. In Figure 49 we can see that the non-losing algorithms decided to wait a lot less than the others. In Figure 50 we can see that the TWAP would lose money. The simple classifier had its threshold changed to the "best"

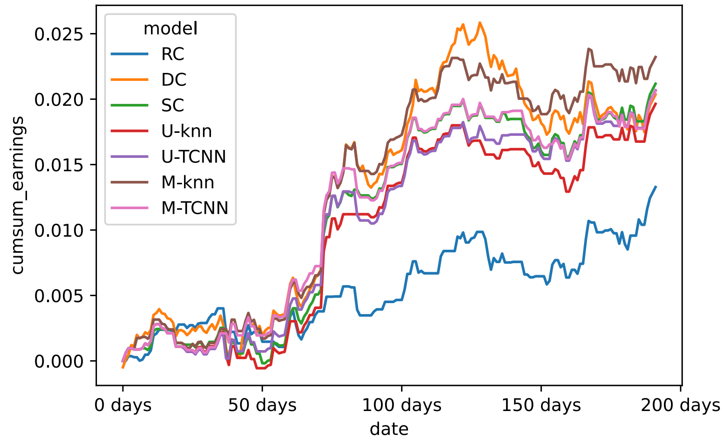

Figure 42: The different models' results on PnL when back-tested.

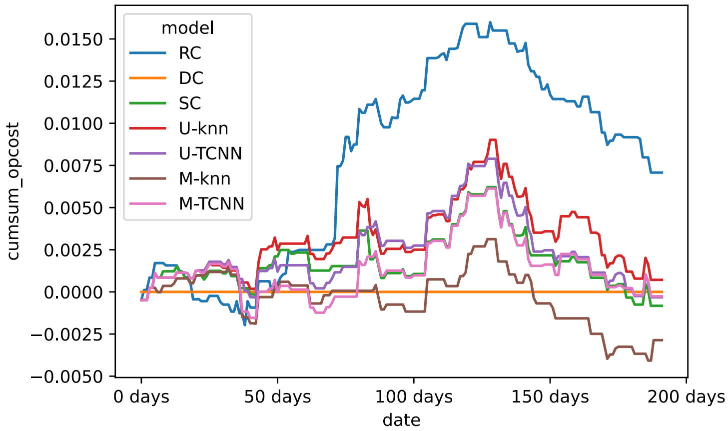

Figure 43: The different models' results on opportunity cost when back-tested.

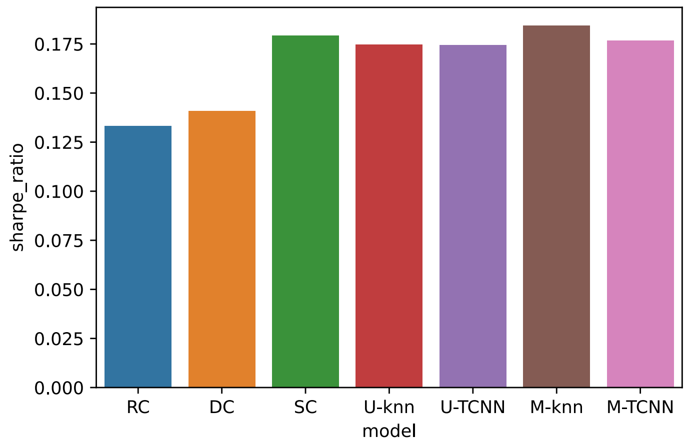

Figure 44: The different models' results on Sharpe when back-tested.

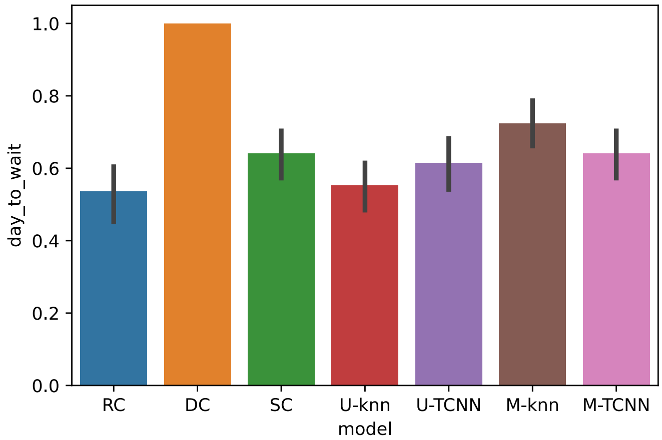

Figure 45: The percentage of days the algorithms decided to wait.

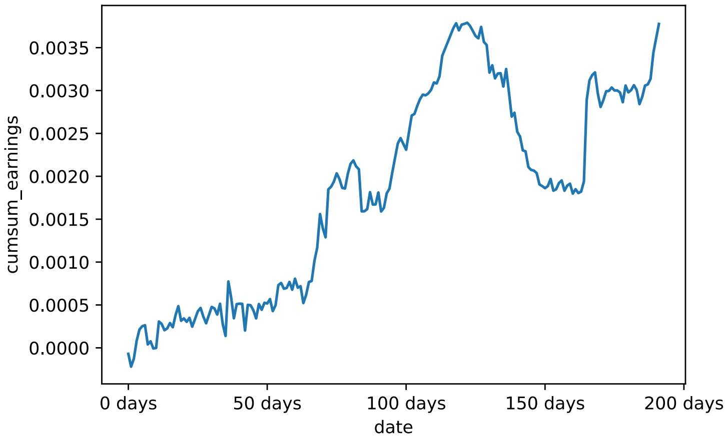

Figure 46: The TWAP algo's result on PnL when back-tested.

for this currency pair.

9 Conclusions

To conclude this work several conclusions can be drawn. The first insight is that the simple theoretical model seems to capture the overall price dynamics around the Fix. By deriving the optimal trading strategy and changing the market impact, the clustering of the simulated price trends and the data around the Fix became similar.

Another insight is that clustering time series is hard in general when noise is present. Even in the controlled theoretical setting, the price paths do not get clustered perfectly. Another insight is that the price path seems to contain most of the predictive information around the Fix for accuracy. Including more variables could however increase the recall and precision, which might be of great importance in a real trading model.

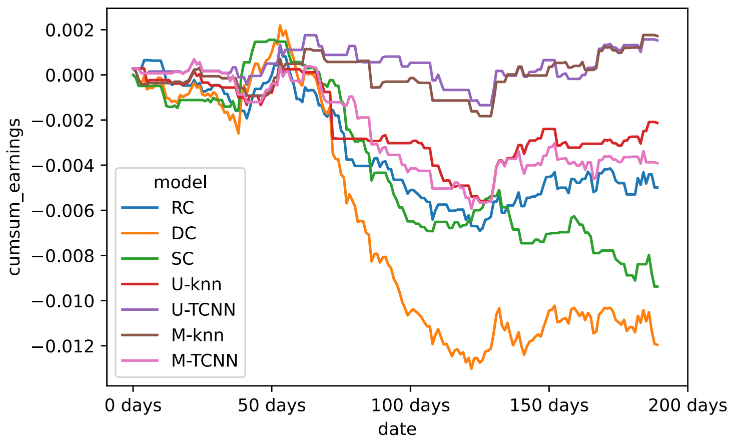

Figure 47: The different models' results on PnL when back-tested.

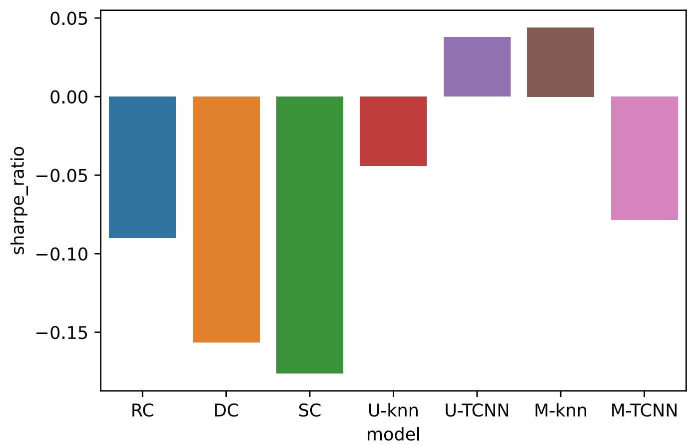

Figure 48: The different models' results on Sharpe when back-tested.

Only at the Fix could we use other variables to cluster the price path. This could not be seen at other times, which indicated that the other variables do contain information about the price. The predictiveness of the direction after the Fix was overall quite good. To be able to classify with an accuracy close to 60% percent indicates that the Fix is predictable and that it is worth investing in a trading strategy similar to this. However, the trading results were not as impressive as the prediction results, indicating that this is something to look further upon.

10 Further work

To be able to push the predictive power it might be better to understand what happens between marketplaces and not just the aggregation of marketplaces. Another thing to consider is to look at measuring market impact in a better way. In this report, one way of measuring the market gave these results, but there is no obvious way to measure market impact and thus this way of measuring might not be appropriate. Another thing to look

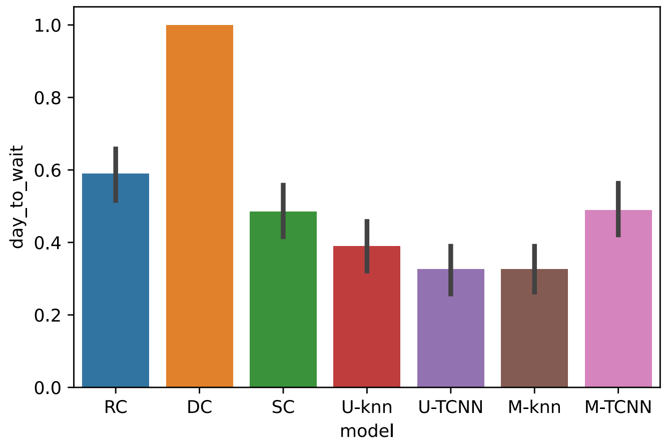

Figure 49: The percentage of days the algorithms decided to wait.

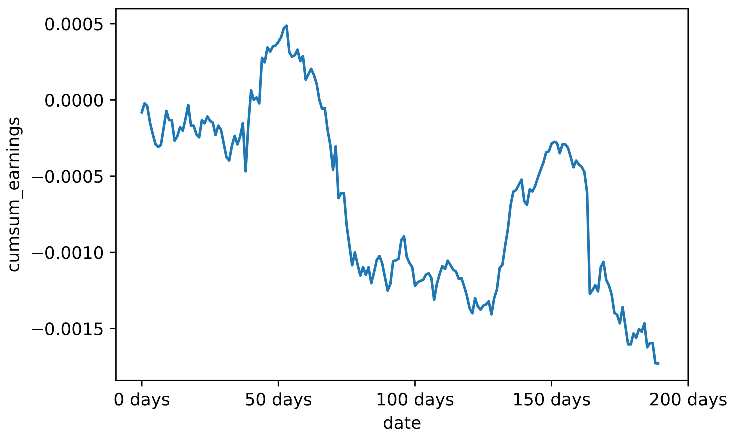

Figure 50: The TWAP algo's result on PnL when back-tested.

into is the result in Figure 41. It seems like the most important information is contained right before the Fix. Maybe it can be beneficial to cut down the time to just two or one minutes before the Fix.

Another thing to look at is why the trading strategy performed, in the author's opinion, worse than the prediction results. Structuring the trading algorithm in a different way might be better at using the prediction power of the models more.

It would also be interesting to extend the mathematical model to consider risk-averse market makers as well. It seems to be harder to solve in general and numerical methods might have to be used, but the result can be insightful. Another extension to the model could be to include more market makers or to extend the model to continues trading.

References

[1] John Schmidt Anna-Louise Jackson. *A Basic Guide To Forex Trading*. 2021. URL: https://www.forbes.com/advisor/investing/what-is-forex-trading/ (visited on 05/05/2022).

[2] BIS. *Triennial Central Bank Survey of Foreign Exchange and Over-the-counter (OTC) Derivatives Markets in 2019*. 2019. URL: https://www.bis.org/statistics/rpfx19.htm (visited on 05/24/2022).

[3] Bloomberg. *BFIX methodology, FX benchmark solutions*. 2020. URL: https://data.bloomberglp.com/notices/sites/3/2016/04/bfix_methodology.pdf (visited on 05/16/2022).

[4] Álvaro Cartea, Sebastian Jaimungal, and José Penalva. *Algorithmic and high-frequency trading*. Cambridge University Press, 2015.

[5] Global Foreign Exchange Committee. *FX GLOBAL CODE*. 2021. URL: https://www.globalfxc.org/docs/fx_global.pdf (visited on 05/27/2022).

[6] Martin DD Evans. "Forex trading and the WMR fix". In: *Journal of Banking & Finance* 87 (2018), pp. 233–247.

[7] Martin DD Evans and Richard K Lyons. "Order flow and exchange rate dynamics". In: *Journal of political economy* 110.1 (2002), pp. 170–180.

[8] fchollet. *Grad-CAM class activation visualization*. 2021. URL: https://keras.io/examples/vision/grad_cam/ (visited on 05/16/2022).

[9] John Cristian Borges Gamboa. "Deep learning for time-series analysis". In: *arXiv preprint arXiv:1701.01887* (2017).

[10] Robert L Kissell. *Algorithmic trading methods: Applications using advanced statistics, optimization, and machine learning techniques*. Academic Press, 2020.

[11] Wenjie Lu et al. "A CNN-LSTM-based model to forecast stock prices". In: *Complexity* 2020 (2020).

[12] Carol Osler, Alasdair Turnbull, et al. "Dealer trading at the fix". In: (2016).

[13] Stuart Russell and Peter Norvig. "Artificial intelligence: a modern approach". In: (2002).

[14] Jo Saakvitne. "'Banging the Close': Price Manipulation or Optimal Execution?" In: *Available at SSRN 2753080* (2016).

[15] Jo Saakvitne. "Trading Strategies and Information Flow Around Price Benchmarks". In: *Available at SSRN 3375564* (2018).

[16] Jo Albertsen Saakvitne. "Essays on Market Microstructure". In: (2019).

[17] Naveen Sai Madiraju et al. "Deep Temporal Clustering: Fully Unsupervised Learning of Time-Domain Features". In: *arXiv e-prints* (2018), arXiv–1802.

[18] Ramprasaath R Selvaraju et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization". In: (2017), pp. 618–626.

[19] Patrick Steffen Michelberger and Jan Hendrik Witte. "Foreign Exchange Market Microstructure and the WM/Reuters 4pm Fix". In: *arXiv e-prints* (2015), arXiv–1501.

[20] Zihao Zhang, Stefan Zohren, and Stephen Roberts. "Deeplob: Deep convolutional neural networks for limit order books". In: *IEEE Transactions on Signal Processing* 67.11 (2019), pp. 3001–3012.