ROYAL HOLLOWAY, UNIVERSITY OF LONDON

Prediction with Expert Advice for Trading and Hedging on the Foreign Exchange Market

by

Najim Al-baghdadi

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

of the
Royal Holloway, University of London
Department of Computer Science

20 April, 2023

Declaration of Authorship

I, Najim Al-baghdadi, declare that this thesis titled, 'Prediction with Expert Advice for Trading and Hedging on the Foreign Exchange Market' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: Najim Al-baghdadi

Date: 20 April 2023

ROYAL HOLLOWAY, UNIVERSITY OF LONDON

Abstract

Faculty Name
Department of Computer Science

Doctor of Philosophy

by Najim Al-baghdadi

In this thesis, we explore the application of prediction with expert advice algorithms for investing in the Foreign Exchange (FX) market.

We introduce a data staging algorithm designed to reconstruct multiple time series databases into a partitioned and regularised database. The Data Aggregation Partition Reduction Algorithm, or DAPRA for short, was designed to solve the practical issue of effective and meaningful visualisation of irregularly sampled time series data.

We apply methods of prediction with expert advice to real-world foreign exchange trading data to find effective investment strategies. We build upon the framework of the long-short game, introduced by Vovk and Watkins (1998), and propose modifications aimed at improving the performance with respect to standard portfolio performance indicators.

We apply the Weak Aggregating Algorithm (WAA) to find optimal risk management strategies for financial Market Makers (MMs), using hedging strategies as experts. We combine their hedging decisions to reduce portfolio risk and maximise profitability. We develop a variation of the WAA using discounting and evaluate the results on commonly traded FX currency pairs.

Acknowledgements

I am pleased to take this opportunity to express my sincere gratitude to all those who have made significant contributions to the completion of this thesis.

Firstly, I would like to extend my heartfelt thanks to my supervisor Yuri Kalnishkan, whose profound expertise and insightful guidance in the field of prediction with expert advice, have been pivotal in shaping this work. His unwavering support and constructive feedback have been invaluable throughout this project, and the privilege to work alongside him has been a true honour.

I would also like to acknowledge the invaluable contribution of my external supervisors, David Lindsay and Sian Lindsay, whose extensive knowledge and mentorship have been indispensable in shaping my understanding of this field.

Furthermore, I would like to express my deep gratitude to my family, Shaab, Jackie, Laith, and Sophia Al-baghdadi, for their unrelenting support and encouragement throughout my academic journey. Their unwavering belief in my abilities has been a driving force behind my success.

Lastly, I would like to thank Royal Holloway, University of London and AlgoLabs for the funding of this research.

Contents

Abbreviations

AA	**A**ggregating **A**lgorithm
AAd	**A**ggregating **A**lgorithm with **d**iscounting
AAse	**A**ggregating **A**lgorithm with **s**pecialist **e**xperts
WAA	**W**eak **A**ggregating **A**lgorithm
WAAd	**W**eak **A**ggregating **A**lgorithm with **d**iscounting
DAPRA	**D**ata **A**ggregation **P**artition **R**eduction **A**lgorithm
MM	**M**arket **M**aker
LS	**L**ong-**S**hort Game
DLS	**D**ownside **L**ong-**S**hort Game
PnL	**P**rofit and **L**oss
FX	**F**oreign E**x**change

Chapter 1

Introduction

In today's world, where decision-making is critical in almost every aspect of our lives, seeking advice from experts has become standard practice. Whether we are placing a bet on a football match or trying to predict the weather, we often rely on the insights of others to guide our decision-making process. However, understanding how we process information to draw meaningful conclusions is still a mystery. The study of prediction with expert advice is a field that seeks to provide a machine-learning framework, to enable models to make decisions based on a set of experts' predictions.

In this thesis, we explore a special case of prediction with expert advice in portfolio management, namely, finding optimal investment and hedging strategies in the Foreign Exchange market. This is a challenging problem for several reasons. Our learners must make decisions in a sequential manner. Where at each time step, a prediction is made and a subsequent outcome is revealed. The goal is that over time, our learners will be able to identify those experts providing insightful predictions and use this to guide their own. This method of learning is advantageous as new data affects a model in real time, as opposed to a batch algorithm that is dependent on training data. This is particularly useful in portfolio management, where changes in market conditions can quickly lead to a model that has been performing well to fail.

The Aggregating Algorithm (AA) developed in [1], is an established prediction with expert advice algorithm that we will use for much of this thesis. The AA first assigns a pool of experts with an initial distribution of weights, then based on an expert's loss at each trial, their weight is adjusted accordingly. The AA provides a theoretical guarantee that for a learner following the AA, their loss will be within some bounds of the best expert, at any given time.

The AA has been used to tackle the problem of investing in financial markets in the past. Firstly in [2] and later developed in [3]. The work in this thesis builds upon these papers to improve the practical application of the AA when investing using real-world market data. This is achieved by looking at the methods used to evaluate experts' loss and developing them to find more optimal investment strategies, concerning financial metrics.

The data used in this thesis is taken from the real world and presents an interesting problem. The nature of prediction with expert advice is that a learner must be able to read a prediction from an expert and then make a prediction in the same space. However, a learner here makes decisions in discrete time and investors in financial markets act in continuous time. We therefore require a method of taking raw trading data and converting it into a regularised format, to be used as an expert prediction. To address this issue we have introduced the Data Aggregation Partition Reduction Algorithm (DAPRA). This is a method of taking multiple time series data sets a producing one regularised data set. We will discuss this further in chapter 4.

We also explore the problem of using prediction with expert advice to find optimal hedging strategies for a broker in the FX market. Namely, apply the Weak Aggregating Algorithm (WAA), and merge a pool of hedging strategies. To improve the application of the WAA, we also introduce discounting to the framework of the algorithm.

The work of this thesis not only refines the Aggregating Algorithm, including the use of experiments on real-world data. That has been shown to significantly improve the ability of the algorithm to distinguish, profitable trading strategies. But, also introduces the DAPRA framework, providing a new method of processing time-series data. Whilst in this thesis, we have used DAPRA primarily for processing data, to facilitate the application of the AA of real word data. We must acknowledge the importance of DAPRA in other areas of data management. This introduction provides the foundational framework, to standardize time series data. We are taking unstructured data, and returning structured and consistent datasets. This breakthrough opens the door to a diverse range of potential applications. Moreover, this research contributes significantly to the study of hedging, particularly in the context of market making. Unlike traditional literature that often focuses on hedging with derivatives, our exploration involves managing a market maker's overall position open with clients, providing a fresh perspective on effective hedging strategies.

1.1 Original Contributions

The key theoretical contributions of the thesis are presented in chapters 5 and 6. Note, chapter 2 presents an overview of the established theory of prediction with expert advice.

1. The introduction of the Data Aggregation Partition Reduction Algorithm (DAPRA), is presented in chapter 4. This is a method used to merge multiple time series datasets, of irregularly sampled data. Into one regularly partitioned dataset.

2. In chapter 5, we study the use of the Aggregating Algorithm for investing in Foreign Exchange markets. This is explored within the framework of the well-established Long-Short game, where we see poor performance in real applications. In this thesis, we introduce modifications to the game that have been shown to improve the performance in empirical trials. These modifications focus on the evaluation of experts' loss, introducing return scaling and downside loss.

3. In chapter 6 we explore the problem of using prediction with expert advice, to find optimal hedging strategies for Foreign Exchange brokers. We introduce the framework to use the WAA, to combine a pool of hedging strategies. In Section 6.3 we show the application of discounting loss to the Weak Aggregating Algorithm, introducing the WAAd.

1.2 Publications

The content of this thesis is based on publications of the following conference proceedings.

- Najim Al-baghdadi, Wojciech Wisniewski, David Lindsay, Sian Lindsay, Yuri Kalnishkan, and Chris Watkins. Structuring Time Series Data to Gain Insight into Agent Behaviour. In *Proceedings of the 3rd International Workshop on Big Data for Financial News and Data. IEEE*, 2020, pages 5480-5490.

- Najim Al-baghdadi, David Lindsay, Yuri Kalnishkan, and Sian Lindsay. Practical investment with the long-short game. *In The 9th Symposium on Conformal and Probabilistic Prediction with Applications: COPA 2020, Proceedings of Machine Learning Research, vol. 128*, 2020, pages 209-228.

- Najim Al-baghdadi, David Lindsay, Yuri Kalnishkan, and Sian Lindsay. Online Portfolio Hedging with the Weak Aggregating Algorithm. *In The 11th Symposium on Conformal and Probabilistic Prediction with Applications: COPA 2022, Proceedings of Machine Learning Research, vol. 179*, 2022, pages 149-168.

- Najim Al-baghdadi, David Lindsay, Yuri Kalnishkan, and Sian Lindsay. Practical investment with the long-short game. *Annals of Mathematics and Artificial Intelligence*, Springer, 2023.

1.3 Thesis Structure

In Chapter 2 we provide an overview of the existing prediction with expert advice algorithms, that we later develop in the thesis. In Chapter 3, we review the Foreign Exchange market and the role brokers play within the industry. In chapter 4, we introduce the ETL algorithm DAPRA and demonstrate its application on multiple datasets. Then in Chapter 5 we show how the Long-Short game performs on real-world FX data, and introduce modifications to improve its performance. Chapter 6 introduces the cylinder hedging model and explores the use of the Weak Aggregating Algorithm to find optimal hedging strategies.

Chapter 2

Prediction with Expert Advice

2.1 Framework

Consider the following prediction scenario. On every step $t = 1, 2, \ldots$, the learner L produces a *prediction* $\gamma_t \in \Gamma$, where Γ is a known prediction space. The nature produces a *loss function* $\lambda_t : \Gamma \to \mathbb{R}$ and the learner suffers loss $\ell_t = \lambda_t(\gamma_t)$. We measure the performance of L by the *cumulative loss* over T steps given by

$$\mathrm{Loss}_T(\mathcal{L}) = \sum_{t=1}^{T} \ell_t \ .$$

We want the cumulative loss to be as low as possible.

Now suppose that there are N *experts* $\mathcal{E}_n$, $n = 1, 2, \ldots, N$, making prediction in the same environment as $\mathcal{L}$ so that their predictions are available to $\mathcal{L}$ before it makes its own. We will treat the experts as black boxes and will not be concerned with their internal mechanics. It is an important requirement that their predictions are available to $\mathcal{L}$ before it makes its own and that they will suffer loss according to he same function λ_t. The interaction with experts may be described by Protocol 1.

Protocol 1 Prediction with Expert Advice Protocol

for $t = 1, 2, \ldots$ **do**
experts $\mathcal{E}_n$ output predictions $\gamma_t^n \in \Gamma$, $n = 1, 2, \ldots, N$
learner $\mathcal{L}$ outputs a prediction $\gamma_t \in \Gamma$
nature produces a function $\lambda_t : \Gamma \to \mathbb{R}$
experts $\mathcal{E}_n$ suffer losses $\ell_t^n = \lambda_t(\gamma_t^n)$, $n = 1, 2, \ldots, N$
learner $\mathcal{L}$ suffers loss $\ell_t = \lambda_t(\gamma_t)$
end for

We want the cumulative loss $\text{Loss}_T(\mathcal{L})$ to be small compared to the minimum of experts' losses $\text{Loss}_T(\mathcal{E}_n) = \sum_{t=1}^{T} \ell_t^n$. Formally one can think of $\mathcal{L}$ as a *merging strategy*

$$\mathcal{L} : \left((\Gamma \times \mathbb{R})^N\right)^* \times \Gamma^N \to \Gamma$$

turning an array of experts' predictions and a history of their prediction and losses into its prediction.

We will aim to impose minimal restrictions on the loss functions λ_t output by the nature; we will not be assuming that the nature can be modeled in a reasonable sense.

Remark 2.1. A simple scenario covered by Protocol 1 is the one where $\Gamma = [0, 1]$ and $\lambda_t(\gamma) = |\gamma - \omega_t|$, where ω_t is generated by the nature. Here a predictor aims to output predictions γ_t approximating outcomes ω_t.

2.2 Aggregating Algorithm

We will introduce the aggregating algorithm following [3].

A *game* a $\mathfrak{G}$ is a triple $\langle \Omega, \Gamma, \lambda \rangle$ consisting of an *outcome space* Ω, a *prediction space* Γ, and a *loss function* $\lambda : \Omega \times \Gamma \to [0, +\infty]$.

The outcomes $\omega_1, \omega_2, \dots$ occur in succession. A prediction strategy $\mathcal{S}$ outputs a prediction γ_t before seeing each outcome ω_t and suffers loss $\lambda(\omega_t, \gamma_t)$ after the outcome ω_t is revealed. The performance of the strategy over T steps is measured by the *cumulative loss* $\text{Loss}_T(\mathcal{S}) = \sum_{t=1}^{T} \lambda(\omega_t, \gamma_t)$. In the investment scenarios, the semantics is slightly different. The value γ_t represents the decision taken on step t and $\lambda(\omega_t, \gamma_t)$ quantifies the consequences of γ_t facing the developments represented by ω_t. An investor is not aiming to make γ_t "close" to the values of ω_t in any sense, but still wants to minimize the cumulative loss. We will retain the prediction terminology though.

The aggregating algorithm (AA) is a way of making a prediction-based on the predictions provided by a pool of *experts* (prediction strategies) Θ, where $\gamma_t(\theta) \in \Gamma$ denotes the prediction of expert $\theta \in \Theta$ at trial t. The AA treats experts as black boxes but have access to their predictions $\gamma_t(\theta)$ before making its own prediction γ_t.

The AA takes the following parameters: a learning rate $\eta > 0$ and an initial distribution on experts $P_0(d\theta)$, which quantifies the initial trust in each expert. We will denote the prediction strategy using AA with parameters η and P_0 by $\text{AA}(\eta, P_0)$.

The AA maintains weights P_t on experts Θ. After each trial t, the experts' weights are updated as follows:

$$P_t(d\theta) = e^{-\eta\lambda(\omega_t,\gamma_t(\theta))}P_{t-1}(d\theta) \ .$$

Therefore the larger the expert's loss the greater the reduction of its weight. To define the AA we first will introduce the aggregating pseudo-algorithm (APA), which at trial t produces a *generalized prediction* (a function $g : \Omega \to (-\infty, +\infty]$) based on the normalized weights as follows:

$$g_t(\omega) = -\frac{1}{\eta}\ln\int_\Theta e^{-\eta\lambda(\omega_t,\gamma_t(\theta))}P^*_{t-1}(d\theta) \ ,$$

where P^*_{t-1} denotes the normalised weights $P^*_{t-1} = P_{t-1}(d\theta)/P_{t-1}(\Theta)$. One can define the cumulative loss of APA as $\mathrm{Loss}(\mathrm{APA}(\eta, P_0)) = \sum_{t=1}^{T} g_t(\omega_t)$. The following lemma can be proven by induction.

Lemma 2.2. *For any learning rate $\eta > 0$, initial distribution P_0, and $T = 1, 2, ...$ we get*

$$\mathrm{Loss}_T(\mathrm{APA}(\eta, P_0)) = -\frac{1}{\eta}\ln\int_\Theta e^{-\eta\,\mathrm{Loss}_T(\theta)}P_0(d\theta).$$

for all $\omega_1, \omega_2, \ldots, \omega_T$.

To obtain the AA from the APA, we need to find a permitted prediction $\Sigma(g_t)$, where the *substitution function* Σ maps a generalised prediction $g : \Omega \to (-\infty, \infty]$ to a prediction $\Sigma(g) \in \Gamma$ while keeping the loss as low as possible. Let $\mathrm{GA}(\eta)$ be the set of all generalised actions that can be produced by the APA with learning rate η:

$$\mathrm{GA}(\eta) = \left\{ g : \Omega \to \mathbb{R} \mid \exists P\ \forall\omega : g(\omega) = -\frac{1}{\eta}\ln\int_\Gamma e^{-\eta\lambda(\omega,\gamma)}P(d\gamma) \right\} \ ,$$

where P ranges over all distributions[1] on Γ. For a generalized action g, let

$$C(g) = \inf_{\gamma\in\Gamma}\sup_{\omega\in\Omega} \lambda(\gamma, \omega)/g(\omega) \ .$$

Under mild continuity assumptions (namely, if Γ is a compact topological space and $\lambda(\omega, \cdot)$ is continuous in the second argument) the value of $C(g)$ is *achieved* on some $\gamma \in \Gamma$ and we can replace g by γ such that $\lambda(\omega, \gamma) \leq C(g)g(\omega)$ for every $\omega \in \Omega$.

[1] We assume that all $\gamma_t(\cdot) : \Theta \to \Gamma$ are Borel mappings of topological spaces and induce measures on Γ.

The mixabthe ility constant C_η is defined as

$$C_\eta = \sup_{g \in \mathrm{GA}(\eta)} C(g) \ .$$

There is a substitution function Σ mapping generalized predictions g to Γ satisfying:

$$\forall g \in \mathrm{GA}(\eta)\, \forall \omega \in \Omega : \lambda(\omega, \Sigma(g)) \leq C_\eta g(\omega) \ . \tag{2.1}$$

Substitution functions satisfying condition (2.1) are the ones allowed to be used in the AA. Condition (2.1) and Lemma 2.2 imply

$$\mathrm{Loss}_T(\mathrm{AA}(\eta, P_0) \leq C_\eta \, \mathrm{Loss}_T(\mathrm{APA}(\eta, P_0)) = -\frac{C_\eta}{\eta} \ln \int_\Theta e^{-\eta \, \mathrm{Loss}_T(\theta)} P_0(d\theta) \ . \tag{2.2}$$

A game is said to be *η-mixable* if $C_\eta = 1$ and *mixable* if it is η-mixable for some $\eta > 0$. For mixable games the learner following the AA can perform almost as well as any expert from a finite pool, as the following lemma shows.

Lemma 2.3. *For a finite pool of experts Θ,*

$$\mathrm{Loss}_T(\mathrm{AA}(\eta, P_0)) \leq C_\eta \, \mathrm{Loss}_T(\theta) + \frac{C_\eta}{\eta} \ln \frac{1}{P_0(\theta)} \tag{2.3}$$

for every expert $\theta \in \Theta$ and time $T = 1, 2, \ldots$ Moreover, if the game is η-mixable, then

$$\mathrm{Loss}_T(\mathrm{AA}(\eta, P_0)) \leq \mathrm{Loss}_T(\theta) + \frac{1}{\eta} \ln \frac{1}{P_0(\theta)} \ . \tag{2.4}$$

Indeed, for a finite pool of experts, the integral in (2.2) turns into a sum of non-negative terms and the sum can be bounded from below by each of its terms.

Bounds (2.3) provided by the AA are optimal in their class for the uniform initial distribution [4]. If an algorithm provides a guarantee of this type, the AA with some η can do the same or better; hence the significance of the AA.

Taking η such that $C_\eta = 1$ minimizes the first term on the right-hand side of (2.3); this is important because this term may be growing with T (and would normally grow as a linear rate). Thus ηs making the game mixable are good choices for practice as long as they exist. Out of such ηs the maximum value should be chosen because it minimizes the second term on the right-hand side of (2.4).

Algorithm 1 is the AA for the case of finitely many experts, which is the main case for this thesis. Here N is the number of experts and Θ is identified with $\{1, 2, \ldots, n\}$. In this context, one can think of Σ as a function $\Gamma^N \times \mathbb{P}_{N-1} \to \Gamma$ (where $\mathbb{P}_{N-1}$ is the

$(N-1)$-simplex) mapping arrays of experts' predictions and distributions on them to predictions.

Algorithm 1 The aggregating algorithm for finitely many experts

---

Parameters: Learning rate $\eta > 0$ and
initial experts' weights $p_0(n)$, $n = 1, 2, \ldots, N$
for $t = 1, 2, \ldots$ **do**
  read experts' predictions $\gamma_t(n) \in \Gamma$, $n = 1, 2, \ldots, N$
  normalise the weights: $p^*_{t-1}(n) = p_{t-1}(n) / \sum_{n=1}^{N} p_{t-1}(n)$
  produce the generalised prediction

$$g_t(\omega) = -\frac{1}{\eta} \ln \sum_{n=1}^{N} p^*_{t-1}(n) e^{-\eta\lambda(\omega, \gamma_t(n))}$$

  calculate and output $\gamma_t = \Sigma(g_t)$
  read $\omega_t \in \Omega$
  update the weights $p_t(n) = p_{t-1}(n) e^{-\eta\lambda(\omega_t, \gamma_t)}$
end for

2.2.1 Covers Game

Cover's game formalizes a basic investment scenario. We include Cover's game in our discussion for completeness.

Cover's game describes investment into a market of M assets. The outcome space Ω describes the behavior of the market with the non-negative price relative vector $\omega = (\omega[0], ..., \omega[M-1]) \in \Omega = [0, \infty)^M$, where $\omega_t[m]$ represents the ratio of the value of asset m at trial t to the value at trial $t-1$. If $S_t[m]$ denotes the price of asset m at time t, then $\omega_t[m] = S_{t+1}[m]/S_t[m]$. An investment in this market is represented by the m-dimensional portfolio vector γ, where $\gamma[m]$ denotes the proportion of the investor's wealth invested in asset m. In Cover's game, we assume that all wealth is invested on every step and no short positions or trading on credit is allowed; in other terms, $\gamma[m] \geq 0$ for $m = 0, 1, \ldots, M-1$, and $\sum_{m=0}^{M-1} \gamma[m] = 1$. The prediction space Γ is the $(M-1)$-simplex $\mathbb{P}_{M-1}$. One can say that the investor partitions the wealth between M assets.

If an investor invests γ and then outcome ω occurs, the investor's wealth changes by a factor of $\langle \omega, \gamma \rangle = \sum_{m=1}^{M-1} \omega[m]\gamma[m]$. To link this with the additive framework of prediction games, we define the loss function by $\lambda(\omega, \gamma) := -\ln\langle \omega, \gamma \rangle$. If the investor starts from wealth of $W_0 = 1$ and follows an investment strategy $\mathcal{S}$, where $\mathcal{S}$ represents any given sequence of predictions. Then the wealth after step T equals

$$W_T = \prod_{t=1}^{T} \langle \omega_t, \gamma_t \rangle = e^{-\operatorname{Loss}_T(\mathcal{S})} \quad .$$

Let us discuss mixability of Cover's game. Note that the game does not quite follow our framework as $\lambda(\omega, \gamma)$ may take negative values. Indeed, the scalar product $\langle \omega, \gamma \rangle$ may well exceed 1 leading to negative loss (in this situation the investor actually earns money). Following [3], consider a modified loss function

$$\tilde{\lambda}(\omega, \gamma) = \lambda(\omega, \gamma) - \inf_{\delta} \lambda(\omega, \delta) \ . \tag{2.5}$$

The new loss function is non-negative by construction. The following lemma describes the properties of the game.

Lemma 2.4 ([3]). *For every $\eta \leq 1$, $C_\eta = 1$. Moreover, for every $\eta \leq 1$ and every $g \in \mathrm{GA}(\eta)$, $C(g) = 1$. The only prediction attaining $C(g) = 1$ is the average*

$$\gamma^* := \int_\Gamma \gamma P(d\gamma) \ , \tag{2.6}$$

where P is a probability distribution in Γ generating g:

$$g(\omega) = -\frac{1}{\eta} \ln \int_\Gamma e^{-\eta\lambda(\omega,\gamma)} P(d\gamma) \ .$$

Remark 2.5. Let $\tilde{\lambda}(\omega, \gamma) = \lambda(\omega, \gamma) + f(\omega)$ for some fucntion of ω. Then the weights $p_t(n)$ in Algorithm 1 calculated for these two functions are proportional and the normalized weights $p_t^*(n)$ are identical. Indeed, all experts suffer loss according to the same ω and $f(\omega)$ drops in normalisation.

Remark 2.6. Now let $C_\eta = 1$ in (2.1) and a substitution rule Σ (considered as a function of predictions $\gamma_1, \ldots, \gamma_N$ and a distribution on them) works w.r.t. the loss function $\tilde{\lambda}$. Then the same rule satisfies (2.1) for the original λ, which may take negative values. Indeed, $f(\omega)$ cancels out on both the sides.

The remarks imply that we can apply the AA for Cover's game with $\eta \in (0, 1]$ using the weighted average as the substitution rule.

Remark 2.7. When $C_\eta = 1$ and (2.4) holds for $\tilde{\lambda}$, the terms $f(\omega_t)$ on both the sides of (2.4) cancel out so we get the bound for cumulative losses in terms of the original loss function λ.

In the case of $\eta = 1$ it is easy to prove that the average of experts' predictions have the desired properties directly. Linearity of the scalar product implies

$$g(\omega) = -\ln \int_\Gamma e^{-\lambda(\omega,\gamma)} P(d\gamma) = -\ln \int_\Gamma \langle \omega, \gamma \rangle P(d\gamma) = \\ -\ln \left\langle \omega, \int_\Gamma \gamma P(d\gamma) \right\rangle \ .$$

For larger values of η, we do not get mixability.

Lemma 2.8 ([3]). *When $\eta > 1$, $C_\eta = \eta$.*

We see that $\eta = 1$ is the optimal choice of the parameter and the substitution rule

$$\gamma_t = \sum_{n=1}^{N} p^*_{t-1}(n)\gamma_t(n) \tag{2.7}$$

should be used with Cover's game in the finite case.

2.2.2 Long-Short Game

The long-short game is a modification of Cover's game aimed at a more general and more realistic trading scenario. A trader is usually allowed to open positions, both long and short, within certain limits based on their deposit and money they had earned previously. The limits are aimed to minimize the chances of bankruptcy so that the intermediary providing access to the market could avoid handling the consequences of the trader's default.

In a *bounded long-short game* with the *prudence parameter* $a > 0$ an investment decision is represented by a vector $\gamma \in \mathbb{R}^M$ such that

$$\|\gamma\|_1 = |\gamma[0]| + \ldots + |\gamma[M-1]| \leq a \ ; \tag{2.8}$$

in other terms, $\Gamma \subseteq \mathbb{R}^M$ is a ball w.r.t. the $\|\cdot\|_1$-norm. The intuitive interpretation of γ is as follows. Suppose that before step t the trader has wealth $W_{t-1} > 0$. Then on step t the trader opens positions of size $W_{t-1}\gamma_t[m]$, $m = 0, 1, 2, \ldots, M-1$ (long or short depending on the sign of $\gamma_t[m]$) in assets $0, 1, \ldots, M-1$. The sum of the sizes of the positions are bounded by $W_t a$.

It is more convenient to represent outcomes by a vector of returns here, so $\omega_t[m] = (S_t[m] - S_{t-1}[m])/S_{t-1}[m] = S_t[m]/S_{t-1}[m] - 1 \geq -1$. Thus on the position in asset m the trader gets the profit of $W_{t-1}\omega_t[m]\gamma_t[m]$ and the overall trader's wealth changes according to

$$W_t = W_{t-1}(1 + \langle \omega_t, \gamma_t \rangle) \ .$$

We let

$$\lambda(\omega, \gamma) = -\ln(1 + \langle \omega, \gamma \rangle) \ .$$

Note that for some values of ω the expression $1 + \langle \omega, \gamma \rangle$ can go below zero; the trader then goes bankrupt and the expression $-\ln(1 + \langle \omega, \gamma \rangle)$ is undefined. In a bounded game, we assume this never happens because all ωs satisfy

$$\|\omega\|_\infty = \max_{m=0,1,\dots,M-1} |\omega_m| \le \frac{1}{a} \ . \tag{2.9}$$

Thus the outcome space in the a-bounded game is the intersection of $[-1,+\infty)^M$ with the $\|\cdot\|_\infty$ ball.

For the analysis of mixability, we need to modify the loss function by (2.5). The following lemma holds for every a-bounded game with $\tilde{\lambda}$.

Lemma 2.9 ([3]). *For any a-bounded game, $a > 0$, and for every $\eta \le 1$, $C_\eta = 1$. Moreover, for every $\eta \le 1$ and every $g \in GA(\eta)$, $C(g) = 1$. The only prediction attaining $C(g) = 1$ is the average (2.6), whereas before P is a probability distribution in Γ generating g. When $\eta > 1$, $C_\eta > 1$.*

As for Cover's game, Remarks 2.5–2.7 imply that for $\eta = 1$ we can apply the AA with the original loss function using the weighted average as the substitution rule.

2.2.3 General Long-Short Game

Although [3] restrict their attention to the bounded long-short game, its practical applications suffer from an important problem. While bound (2.8) on the norm of γ is realistic (and can be linked to the restrictions imposed by the market access provider), bound (2.9) on the norm of ω cannot be guaranteed; the market is not under our control in any way. In this section, we will discuss the general long-short game.

Consider the general long-short game with $\Gamma = \mathbb{R}^m$, $\Omega = [-1,+\infty)^M$ and the loss function given by

$$\begin{aligned}\lambda_{\mathrm{LS}}(\omega,\gamma) &= \begin{cases} -\ln(1+\langle\omega,\gamma\rangle), & \text{if } 1+\langle\omega,\gamma\rangle > 0 \\ +\infty & \text{otherwise}\end{cases} \\ &= -\ln(\max(1+\langle\omega,\gamma\rangle, 0)) \ ,\end{aligned}$$

where $\ln 0 = -\infty$. Our earlier analysis suggests that we should apply the AA using $\eta = 1$ and the weighted average of the expert' predictions as the substitution rule. We formulate the case for finitely many experts (i.e., with a finite pool of experts Θ of size N) as Algorithm 2.

Let us formulate its properties.

Algorithm 2 The aggregating algorithm for the general long-short game

Parameters: initial distribution P_0 over experts, $p_0(n)$, $n = 1, 2, \ldots, N$

for $t = 1, 2, \ldots$ **do**

read experts' predictions $\gamma_t(n) \in \Gamma$, $n = 1, 2, \ldots, N$

normalise the weights: $p^*_{t-1}(n) = p_{t-1}(n) / \sum_{n=1}^N p_{t-1}(n)$

calculate and output $\gamma_t = \sum_{n=1}^N p^*_{t-1}(n)\gamma_t(n)$

read $\omega_t \in \Omega$

update the weights $p_t(n) = p_{t-1}(n)e^{-\lambda_{\mathrm{LS}}(\omega_t, \gamma_t)}$

end for

Lemma 2.10. *Suppose that the outcomes ω_t and experts' predictions $\gamma_t(n)$ are such that $1 + \langle \omega_t, \gamma_t(n) \rangle \geq 0$ for all $t = 1, 2, \ldots, T$, $n = 1, 2, \ldots, N$. Then the predictions $\gamma_t = \sum_{n=1}^N p^*_{t-1}(n)\gamma_t(n)$ satisfy $1 + \langle \omega_t, \gamma_t \rangle \geq 0$ and (2.4) holds with $\eta = 1$:*

$$\mathrm{Loss}_T(\mathrm{AA}(1, P_0)) \leq \mathrm{Loss}_T(n) + \ln \frac{1}{p_0(n)} \tag{2.10}$$

for every expert $n = 1, 2, \ldots, N$.

Note that conditions $1 + \langle \omega_t, \gamma_t(n) \rangle \geq 0$ should hold on to the actual realised sequence rather than for all possible ω.

The proof is along the same lines as the general analysis of the AA described in the previous sections. We will give it here for completeness.

Proof. Let us show by induction that

$$e^{-\mathrm{Loss}_s(AA)} \geq \sum_{n=1}^N p_0(n) e^{-\mathrm{Loss}_s(n)} \tag{2.11}$$

(as a matter of fact, this will hold as equality). Dropping all terms from the sum on the right-hand side except for one and taking the logarithm of this inequality yields (2.10).

Let (2.11) hold for $s = t - 1$. If $1 + \langle \omega_t, \gamma_t(n) \rangle \geq 0$ for all n, then

$$e^{-\lambda_{\mathrm{LS}}(\omega_t, \gamma_t)} = \max(1 + \langle \omega_t, \gamma_t \rangle, 0) \geq \\ \sum_{n=1}^N p^*_{t-1}(n) \max(1 + \langle \omega_t, \gamma_t(n) \rangle, 0) = \sum_{n=1}^N p^*_{t-1}(n) e^{-\lambda_{\mathrm{LS}}(\omega_t, \gamma_t(n))}$$

holds (as an equality) by the linearity of the scalar product.

Multiplying this inequality by (2.11) with $s = t - 1$ and observing that

$$p^*_{t-1}(n) = \frac{p_{t-1}(n)}{\sum_{i=1}^N p_{t-1}(i)} = \frac{p_0(n) e^{-\mathrm{Loss}_{t-1}(n)}}{\sum_{i=1}^N p_0(i) e^{-\mathrm{Loss}_{t-1}(i)}}$$

completes the inductive step. □

Let, however, $1 + \langle \omega_t, \gamma_t(n) \rangle < 0$ for some n with weights $p^*_{t-1}(n) > 0$ (but let there be values n such that $p^*_{t-1}(n) > 0$ and $1 + \langle \omega_t, \gamma_t(n) \rangle > 0$). In this situation the inequality

$$\max\left(1 + \langle \omega_t, \gamma_t \rangle, 0\right) \geq \sum_{n=1}^{N} p^*_{t-1}(n) \max(1 + \langle \omega_t, \gamma_t(n) \rangle, 0) \ ,$$

where $\gamma_t = \sum_{n=1}^{N} p^*_{t-1}(n)\gamma_t(n)$, does not hold. Indeed, consider the case $1 + \langle \omega_t, \gamma_t \rangle \leq 0$. In this case, we get 0 on the left-hand side and a positive expression on the right. In the opposite case $1 + \langle \omega_t, \gamma_t \rangle > 0$, we get

$$1 + \langle \omega_t, \gamma_t \rangle = 1 + \left\langle \omega_t, \sum_{n=1}^{N} p^*_{t-1}(n)\gamma_t(n) \right\rangle = \sum_{n=1}^{N} p^*_{t-1}(n)(1 + \langle \omega_t, \gamma_t(n) \rangle)$$

on the left-hand side and the sum of terms that are the same or greater (by our assumption some are greater) on the right-hand side.

The former case is quite hopeless: the strategy following AA goes bankrupt and suffers infinite loss, while some of our experts still have finite. The latter case is not. Consider

$$c_t = \frac{\sum_{n=1}^{N} p^*_{t-1}(n) \max(1 + \langle \omega_t, \gamma_t(n) \rangle, 0)}{1 + \left\langle \omega_t, \sum_{n=1}^{N} p^*_{t-1}(n)\gamma_t(n) \right\rangle} \ .$$

We have $c_t > 1$, but arguably not by a lot. Although in principle c_t can be arbitrarily high, it is reasonable to expect that even if $1 + \langle \omega_t, \gamma_t(n) \rangle$ is below 0, then not by a lot. Expert n that suffers a bankruptcy is unlikely to have performed very well previously and so its weight $p^*_{t-1}(n)$ should also be small, especially if the pool of experts is large.

We then get

$$\begin{aligned} c_t e^{-\lambda_{\mathrm{LS}}(\omega_t, \gamma_t)} &= c_t \max(1 + \langle \omega_t, \gamma_t \rangle, 0) \geq \\ &\sum_{n=1}^{N} p^*_{t-1}(n) \max(1 + \langle \omega_t, \gamma_t(n) \rangle, 0) = \sum_{n=1}^{N} p^*_{t-1}(n) e^{-\lambda_{\mathrm{LS}}(\omega_t, \gamma_t(n))} \end{aligned}$$

and the term $\ln c_t$ finds its way into the right-hand side of (2.10), which turns into

$$\mathrm{Loss}_T(\mathrm{AA}(1, P_0)) \leq \mathrm{Loss}_T(n) + \ln \frac{1}{p_0(n)} + {\sum}' \ln c_t \ , \tag{2.12}$$

where $\sum'$ is taken over steps when there are bankrupt experts, as long as the strategy following the AA is not bankrupt. Note that an expert going bankrupt worsens the bound for *all* other experts. On the other hand, each expert can contribute to $\sum'$ only once: after it goes bankrupt, its weight is zeroed.

2.3 Specialist Experts

In this section, we will introduce specialist following [5] after [6].

A specialist expert is an expert who may refrain from predicting any given trial. In the event an expert makes a prediction the expert is said to be "awake" and if not we say the expert "sleeps". This is a particularly useful approach to take when it comes to finding optimal investment strategies. It is common for investors to have periods without a position in the market, especially in the case of investors with shorter investment time horizons. How can we interpret this? First, the investor's behavior may be understood as making zero predictions. However, this is not the only possibility. In investment, "doing nothing" often means making a passive investment into an asset perceived as riskless and reliable. This asset is often an index portfolio tapping into the wisdom of the crowd. We thus may think of an expert making no predictions as investing in a kind of index portfolio based on the behavior of fellow experts.

Specialist experts provide a natural way to implement the later approach. We assume that a sleeping expert joins the crowd making the same prediction as the learner and therefore suffering the same loss.

Recall Lemma 2.3 with a bound on the loss the aggregating algorithm guarantees. If an expert is sleeping and makes the same prediction as the learner on some turn, the loss terms in (2.4) would cancel out leaving us with the sums over times when experts are awake to define our learners bound on loss. However, this idea is still hypothetical: to work out the learner's prediction one needs to know the expert's prediction to begin with. We can solve this by recalling how the learner makes a prediction in AA. The prediction γ_t is chosen to satisfy

$$e^{-\eta\lambda(\omega,\gamma_t)} \leq \sum_{n=1}^{N} p_{t-1}(n) e^{-\eta\lambda(\omega,\gamma_t(n))} \quad ,$$

where ω ranges over Ω. Assuming that the loss of a sleeping expert is equal to that of the learner we get

$$e^{-\eta\lambda(\omega,\gamma_t)} \leq \sum_{n\colon\ E_n \text{ is awake}} p_{t-1}(n) e^{-\eta\lambda(\omega,\gamma_t(n))} + \sum_{n\colon\ E_n \text{ is sleeps}} p_{t-1}(n) e^{-\eta\lambda(\omega,\gamma_t)}. \tag{2.13}$$

We can then subtract the second sum from each side getting

$$e^{-\eta\lambda(\omega,\Sigma(g_t))} \leq \frac{1}{z_t} \sum_{n:\ E_n \text{ is awake}} p_{t-1}(n) e^{-\eta\lambda(\omega,\gamma_t(n))},$$

where

$$z_t = \sum_{n:\ E_n \text{ is awake}} p_{t-1}.$$

Algorithm 3 Specialist Experts learning protocol

Parameters: Learning rate $\eta > 0$ and
initial experts' weights $p_0(n)$, $n = 1, 2, \ldots, N$
for $t = 1, 2, \ldots$ **do**
read awake experts' predictions $\gamma_t(n) \in \Gamma$, $n = 1, 2, \ldots, N$
normalise awake experts weights:

$$p^*_{t-1}(n) = p_{t-1}(n) / \sum_{n:\ E_n \text{ is awake}} p_{t-1}(n)$$

produce the generalised prediction:

$$g_t(\omega) = -\frac{1}{\eta} \ln \sum_{n:\ E_n \text{ is awake}} p^*_{t-1}(n) e^{-\eta\lambda(\omega,\gamma_t(n))}$$

calculate and output $\gamma_t = \Sigma(g_t)$
read $\omega_t \in \Omega$
update the awake weights $p_t(n) = p_{t-1}(n) e^{-\eta\lambda(\omega_t,\gamma_t)}$
update the sleeping weights $p_t(n) = p_{t-1}(n) e^{-\eta\lambda(\omega,\Sigma(g_t)}$
end for

2.4 Aggregating Algorithm with Discounting

Discounting loss is a well-established practice in both on-line and reinforcement learning. It may be used to account for inflation over time or to reflect changes in market conditions. In this section, we will explore the use of discounting and how it can be implemented within the aggregating algorithm. We will follow the approach of [7] and [5].

Take coefficients $\alpha_1, \alpha_2, ... \in [0, 1]$ and define the cumulative discounted loss for a learner as

$$\widetilde{\mathrm{Loss}}_T(\mathrm{AA}(\eta, P_0)) = \sum_{t=1}^{T} \lambda(\omega_t, \gamma_t) \left(\prod_{s=t}^{T-1} \alpha_s \right) =$$
$$\alpha_{T-1} \widetilde{\mathrm{Loss}}_{T-1}(\mathrm{AA}(\eta, P_0)) + \lambda(\omega_T, \gamma_T) \ .$$

We define the discounted loss of an expert in the same way. In the case where all α_i are equal, $\alpha_1 = \alpha_2 = ... = \alpha$ and $\lambda(\gamma_t, \omega_t)$ comes into the formula with the discounting coefficient α^{t-1}.

Let us calculate experts' weight as follows

$$p_{t-1}^{\theta} \propto p_0(\theta) e^{-\eta \alpha_{t-1} \widetilde{\mathrm{Loss}}_{t-1}(\theta)}$$

[5] shows by induction that the following bound on the learner's loss holds

$$\widetilde{\mathrm{Loss}}_T(\mathrm{AA}(\eta, P_0)) \leq \widetilde{\mathrm{Loss}}_T(\theta) + \frac{1}{\eta} \ln \frac{1}{P_0(\theta)}. \tag{2.14}$$

Algorithm 4 The aggregating algorithm with discounting

Parameters: Learning rate $\eta > 0$ and
discounting factors $\alpha_1, \alpha_2, ...$ and
initial experts' weights $p_0(n)$, $n = 1, 2, \ldots, N$
for $t = 1, 2, \ldots$ **do**
read experts' predictions $\gamma_t(n) \in \Gamma$, $n = 1, 2, \ldots, N$
normalise the weights:

$$p_{t-1}^*(n) = P_0(n)(p_{t-1}(n))^{\alpha_{t-1}} / \sum_{n=1}^{N} P_0(\theta)(p_{t-1}(n))^{\alpha_{t-1}}$$

produce the generalised prediction

$$g_t(\omega) = -\frac{1}{\eta} \ln \sum_{n=1}^{N} p_{t-1}^*(n) e^{-\eta \lambda(\omega, \gamma_t(n))}$$

calculate and output $\gamma_t = \Sigma(g_t)$
read $\omega_t \in \Omega$
update the weights $p_t(n) = p_{t-1}^{\alpha_{t-1}}(n) e^{-\eta \lambda(\omega_t, \gamma_t)}$
end for

2.5 Weak Aggregating Algorithm

Let Γ be a convex set so that for any $\gamma_1, \gamma_2, \ldots, \gamma_N \in \Gamma$ and probabilities $p_1, p_2, \ldots, p_N$ ($p_n \geq 0$ for $n = 1, 2, \ldots, N$ and $\sum_{n=1}^{N} p_n = 1$) the convex combination $\gamma = \sum_{n=1}^{N} p_n \gamma_n$ is defined and belongs to Γ. Then the learner $\mathcal{L}$ can use Algorithm 5, which we will call the Weak Aggregating Algorithm (WAA).

To obtain performance bounds for WAA, one needs to assume convexity of loss functions λ_t; this ensures the inequality $\ell_t \leq \sum_{n=1}^{N} p_{t-1}^n \ell_t^n$. We will also need losses to be bounded.

Algorithm 5 Weak Aggregating Algorithm

Parameters: Initial distribution $q_1, q_2, \ldots, q_N$, $q_n \geq 0$ for $n = 1, 2, \ldots$ and $\sum_{n=1}^{N} = 1$.
Learning rates $\eta_t > 0$, $t = 1, 2, \ldots$
let $L_0^n = 0$, $n = 1, 2, \ldots, N$
for $t = 1, 2, \ldots$ **do**
calculate weights $w_{t-1}^n = q_n e^{-\eta_t L_{t-1}^n}$, $n = 1, 2, \ldots, N$
normalise the weights $p_{t-1}^n = w_{t-1}^n / \sum_{i=1}^{N} w_{t-1}^i$, $n = 1, 2, \ldots, N$
read experts' predictions $\gamma_t^n \in \Gamma$, $n = 1, 2, \ldots, N$
output $\gamma_t = \sum_{n=1}^{N} p_{t-1}^n \gamma_t^n$
read experts losses ℓ_t^n, $n = 1, 2, \ldots, N$
update $L_t^n = L_{t-1}^n + \ell_t^n$, $n = 1, 2, \ldots, N$
end for

Let $L \in \mathbb{R}$ be such that

$$\max_{n=1,2,\ldots,N} \ell_t^n - \min_{n=1,2,\ldots,N} \ell_t^n \leq L \tag{2.15}$$

for every $t = 1, 2, \ldots$. This is guaranteed if, for example, $\sup_\Gamma \lambda_t(\gamma) - \inf_\Gamma \lambda_t(\gamma) \leq L$ for all $t = 1, 2, \ldots$ Sometimes we assume that L is known in advance.

Theorem 2.11. *Let the learning rates in WAA be $\eta_t = c/\sqrt{t}$ for every $t = 1, 2, \ldots$, where $c > 0$. If all loss functions λ_t are convex and L satisfies (2.15) for $t = 1, 2, \ldots$ then*

$$\mathrm{Loss}_T(\mathcal{L}) \leq \mathrm{Loss}_T(\mathcal{E}_n) + \frac{\sqrt{T}\ln(1/q_n)}{c} + \frac{cL^2\sqrt{T}}{4}$$

for all $T = 1, 2, \ldots$ and all experts $\mathcal{E}_n$, $n = 1, 2, \ldots, N$.

Corollary 2.12. *Under the conditions of Theorem 2.11, if L satisfying (2.15) is known in advance, one can take $c = 2\sqrt{\ln N}/L$ and ensure for equal weights $q_1 = q_2 = \ldots = q_N = 1/N$ the bound*

$$\mathrm{Loss}_T(\mathcal{L}) \leq \mathrm{Loss}_T(\mathcal{E}_n) + L\sqrt{T \ln N}$$

for all $T = 1, 2, \ldots$ and all experts $\mathcal{E}_n$, $n = 1, 2, \ldots, N$.

These results improve on both Corollary 14 by [8] and Theorem 2.3 by [9]. An equivalent result was obtained by [10]. The theorem can be proven along the same lines as the result for discounted loss below.

Chapter 3

Introduction to Financial Markets

In this thesis we will be applying the prediction with expert advice algorithms developed, to solve the problem of investing and hedging in Foreign Exchange (FX) markets. Therefore, before exploring these methods we will first provide an overview of the FX market and the brokerage industry.

3.1 Foreign Exchange Markets

The FX market is an over-the-counter marketplace where currencies are exchanged for one another. Whilst the origin of the market was predominately to minimize the risk of conducting international trade, and still accounts for a large amount of the market. It is now common for individuals to trade in the market for speculative gains.

Currencies are traded as pairs, giving the value of one currency against another. We often refer to this value as the rate and represents the value of a base currency against a counter currency. For example, let us suppose the rate for GBP/USD is 1.2, this tells us for each pound sterling traded we would receive 1.2 United States dollars. Now suppose we bought 100,000 GBP/USD, entering a long position, predicting the rate will rise. In the event the rate did indeed rise to 1.21, this would result in a profit of \$1000 $(100,000 \times 0.01)$. If however, the rate fell to 1.195, this would lead to a loss of \$500 $(100,000 \times 0.005)$. It is important to note that profit and loss (PnL) is always paid in the counter currency, in this case, USD. Therefore, if we wish the compare the PnL of different currency trades we must first account for this difference, we discuss this further in chapter 4. An investor can equally predict that the value of the base currency will fall in value compared to the counter currency. In this case, they would sell the pair, entering into a short position in the hope the rate would fall.

Many factors affect the FX market when trading for speculation. Such as interest rates, economic growth, inflation, and geopolitical events just to name a few. The FX market is highly liquid, meaning that traders can easily enter and exit positions at any time. This liquidity also means that the FX market is highly volatile and can experience sudden price fluctuations. Traders use various technical and fundamental analysis techniques to predict market movements and make trading decisions.

3.1.1 Brokerage Industry

The FX brokerage industry provides access to the FX market for individuals and institutions. FX brokers act as intermediaries between traders and the market, providing a platform for traders to buy and sell currencies. A broker earns revenue through spreads, which is the difference between the bid and ask prices of currency pairs. The bid is the rate at which the broker is willing to sell the currency pair to a trader, and the ask is the rate at which they are willing to purchase the pair.

This allows the broker to act as a liquidity provider for clients, also known as a Market Maker (MM). MMs are central to the operation of financial markets, allowing clients to both buy and sell instruments at any given point in time. If a client wants to trade on the prices published by a market maker, the client must first place an order which is defined as a request to trade a given financial instrument. Market makers will compete for client liquidity by ensuring their spreads are as tight as possible, publishing ask prices that are competitively low and bid prices that are competitively high. If a market maker 'wins' a client trade, it then takes on the risk associated with that trade and assumes a position in the market (either 'long', 'short', or 'flat'). The market maker can choose to hedge its position, placing trades to flatten its position to (hopefully) make money. The market maker can do this by executing trades with tighter price spreads than those published to the clients or by using a specialized hedging model that makes more intelligent decisions about when and how to hedge.

3.2 Hedging

The FX brokerage industry is highly competitive, with many brokers offering similar services and platforms. This competition has led to innovation and the development of more sophisticated broker hedging methods, facilitating brokers to offer more competitive services to their clients. A financial hedge is a strategy that investors or businesses use to reduce the risk of adverse price movements in an asset or liability. A hedge can be used to protect against potential losses, but it can also limit potential gains.

For example, a company that relies on imports from another country may face the risk of currency fluctuations that could increase the cost of those imports. To hedge against this risk, the company could enter into a financial contract, such as a forward contract or an option, to lock in a specific exchange rate. This way, if the exchange rate moves unfavorably, the company is protected against potential losses.

In the context of a FX broker risk can be thought of in the following way. When clients place trades on the broker the summation of said trades pushes the market maker longer or shorter, to define the position the broker takes in the market. Based on the movement in the underlying market, this position will result in either a profit or loss, referred to as the broker's PnL (profit and loss). Resources are finite so keeping the two elements of risk in check is important. This can be done by placing "hedge" trades that mitigate the risk by bringing the net position closer to zero. One can imagine a disastrous scenario whereby the market maker has a very large position in the market but not enough PnL to reduce its position to mitigate further loss. Therefore position risk limits must be put in place. Aside from avoiding position risk limit breaches, knowing when, what, and how much to hedge is the job of the market maker's Hedging Model. It is worth reminding the reader that we can only hedge if we have a long or short position in the market, we cannot hedge if we are flat.

There are effectively 2 default hedging strategies currently employed by all brokerages – 'A-booking' and 'B-booking':

3.2.1 A-Booking

A booking – for every client trade that the market maker executes, the market maker goes back to their Liquidity Provider (LP – who quoted the prices used in the trade) and places a trade of an equal amount and opposite side with this LP. For example:

1. Market maker publishes a EUR/USD Ask price of 1.15 – this is an inflated price offered by LPs (which is 1.14)
2. Client X places a Buy trade with the market maker, buying 1000 EUR/USD at the rate of 1.15. At the same time, the Market maker goes back to LP and places a Sell 1000 EUR/USD trade at a rate of 1.15.
3. The market maker instantly makes a profit (1.15-1.14)/1.15*1000= USD 8.69

The rationale for A-booking focuses on speed since we don't want the LP to re-quote at a less favorable price. A-booking offers a riskless pass-through with the broker taking a small cut in the middle with the inflated price. The profit made on this is almost guaranteed (depending on how fast the LP fills and whether they reject). Brokers typically do this for what they deem to be high-risk clients, where they consistently make money or build up too big a position for them to handle. The market maker assumes no position, all PnL is realized in milliseconds in small positive amounts based on the markup added to the client's prices.

3.2.2 B-Booking

B-booking is the opposite end of the spectrum to A-booking, and this hedging strategy is typically used for lower-risk clients. With this model, a broker does not flatten a client's position imminently but instead allows client positions to accumulate. In the hope, they will profit from the client trade. In reality, both hedging approaches are overly simplistic, no client is consistently 'good' or 'bad' at trading, so a Hedging Model may employ a hybrid approach, for example:

- The hedging model will place hedge trades when it predicts that the market will turn against the broker's position, mitigating any loss in PnL.
- If the position is within the market maker's global risk limits and predicts that its position is favorable i.e. where the market is trending, the Hedging Model will not place any hedges or lift existing hedges.

In Chapter 6 we look at an example of such a model and explore how prediction with expert advice methods can be used to find the optimal hedging strategy.

Chapter 4

Time Series ETL Algorithm DAPRA

4.1 Introduction

In the age of information, big data is the resource that makes the difference between failure and success. Data collection focuses more on the initial task of gathering as much data as possible and ensuring the robustness of data collection and storage systems. Yet the significant task of building usable data structures is by comparison somewhat neglected. The result is that data scientists encounter challenges in producing meaningful visual analyses or implementing machine learning algorithms to predict future data trends. It has become common practice to store data using online analytical processing (OLAP) cubes since these structures allow for fast and effective querying and drill-down analysis. Many commercial products that make use of OLAP cubes are available to purchase for businesses to help them easily evaluate and manage their data. Studies such as [11] have proven the concept to be beneficial for financial analysis. Before data is stored in an OLAP cube, an intermediate process of ETL (Extract, Transform, and Load) is carried out. This process (also known as cleansing and staging) involves extracting the raw data from its source to a data staging area (DSA), where the data is manipulated to suit the requirements of the target data [12]. For example, when dealing with data that is not synchronized or when a set of events has closed but many are still in flux, it is useful to first pre-process the data to take into account this. In the context of time series data, we have identified a need for a generally applicable tool capable of structuring irregularly-sampled data gathered from multiple sources into a format that is practically intuitive to understand, and further avoids loss of information because it is designed "from the ground up".

Interval analysis of time series data has been widely explored in papers such as [13] & [14], but we have not been able to find publications showing how to generate interval data from source data. In this study, we introduce and rationalize a novel DSA process that manages the challenges associated with irregularly sampled time series. We call this the Data Aggregation Partition Reduction Algorithm, or DAPRA for short. We will describe how the DAPRA framework can reformat irregularly sampled data from two distinct domains (financial and travel). We further illustrate how end-users can use DAPRA to gain previously "unseen" insights into agent behavior to supplement current business intelligence and decision-making capabilities.

4.2 Data Aggregation Partition Reduction Algorithm (DAPRA)

DAPRA is specific to analyzing time series data which is derived from some concept of "agents" interacting with their environment. Fig.4.1 illustrates this concept across three distinct time series datasets. The agents are (from left to right) foreign exchange traders, licensed taxis, or network IP addresses. These agents interact with their respective environments of an online trading platform, pick-up/drop-off geographical locations in New York City (NYC), or a computer network. The main requirement for DAPRA is that each agent's action has a start time at which that action was started (i.e. the start time of a network data transmission) and a finish time at which that action was completed (i.e. the time at which a taxi dropped off a customer and so finished the journey).

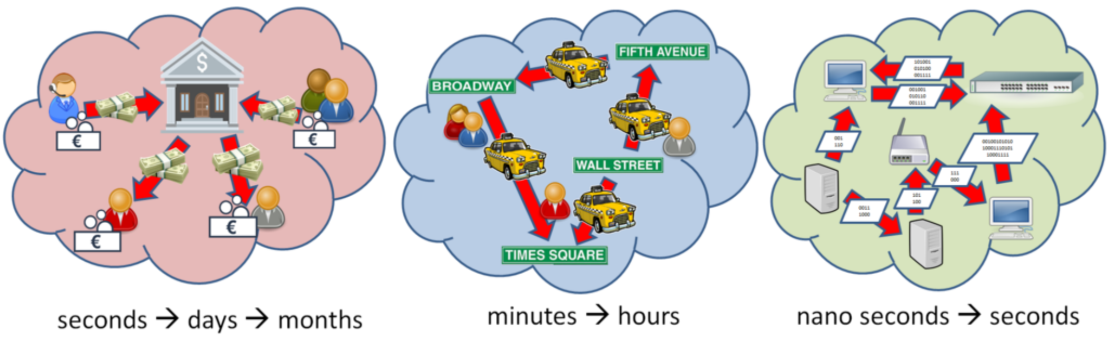

FIG. 4.1: Agents interacting with their environment

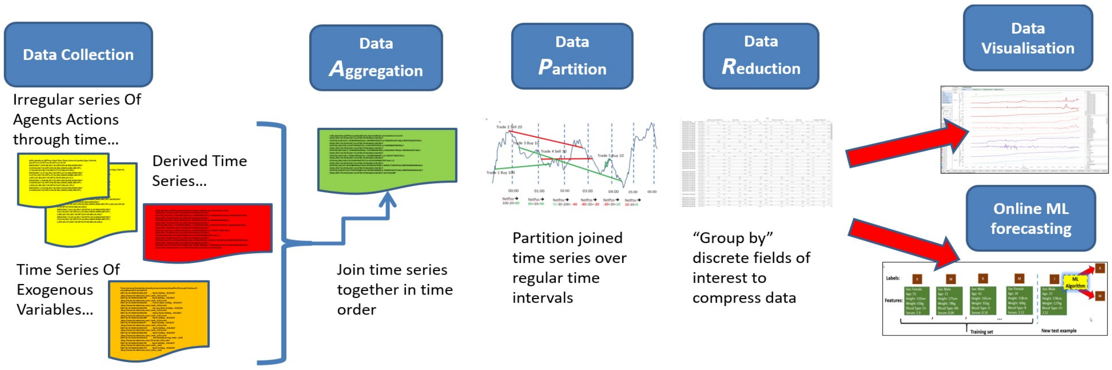

FIG. 4.2: Diagram of the main processes in the DAPRA framework

Fig.4.1 shows that the periods of such actions by agents vary according to the problem domain, ranging from short-lived (i.e. *nanoseconds* for a network data transmission) to very large (e.g. a trader could keep a trade open for *several months* until they speculate on a favorable price movement that will increase the profit on their trade). Time series data can easily contain hundreds if not thousands of agents and all agents are free to carry out as many actions as they are capable of (e.g. an inner-city taxi driver makes more short-duration journeys than taxi drivers operating on the city outskirts).

The frequency or regularity of actions by a given agent can vary too, for example, a trader can place hundreds of trades in one day but in the week that follows place none at all. There is no obligation on a trader to trade, it is entirely their choice as to when and how often they wish to do so. Similarly, if we imagine our agents are licensed taxis they cannot be making journeys all the time, as their drivers will need to rest at predictable periods of the day. Conversely such "dead time" may be unpredictable, for example, a taxi may have to go into the garage for unexpected repair work. Some actions can also be completed simultaneously, for example, a trader can place 10 orders at once. In summary, the problem that DAPRA effectively manages is that of irregularity i.e. irregularity between start and finish times of an agent's actions, as well as irregularity in the number and frequency of actions carried out by agents in a time series.

We set out to develop DAPRA to help generate regular time series databases from irregularly sampled data. Below we explain DAPRA's design and implementation[1] and later show how DAPRA data-restructuring enables the user to easily discern agent behaviors, creating regularised data that is simple to visualize using Business Intelligence tools such

[1]Example DAPRA code can be found at [15]

as Tableau [2].

Firstly it is important to identify and collect the different streams of time series data that DAPRA will be applied to (Fig. 4.2). Essentially these are the irregular time series that describe agents' actions through time, in addition to some time series of "exogenous" variables that are deemed useful in helping to explain agent behavior. For example, one of the case studies used in this thesis - that of taxi journeys around NYC - comprises irregular time series data about the pick-up and drop-off time stamps of taxi journeys undertaken by individual taxis. The exogenous data stream in this case describes the weather in NYC at similar points in time to the taxi journey data stream. It is rational to assume that the weather will have some effect on the number and frequency of taxi journeys made. For example, heavy rain is likely to generate greater demand for taxis and so an increased number of journeys should take place. In addition to the streams of irregular and exogenous time series, it is possible to derive an extra data stream by recording the outputs of simple calculations performed on the irregular or exogenous time series fields. For example, the time distance between the last drop-off and the next pick-up of a given taxi. Likewise, (as will be discussed later for this chapter's second case study on client trades), one can derive extra data by summing the value of individual trades to produce a trader's overall profit and loss (PnL) and position in the market (i.e. long, short or flat). After data collection, Fig.4.2 illustrates that DAPRA follows a three-step process:

1. **D**ata **A**ggregation, where data from one or more sources of irregularly sampled time series data are combined into a regular sampled time series. In particular, we are focusing on sequences of variable time-length actions and trying to correlate these actions with other useful time series to explain the agent's behavior.
2. The aggregated stream is **P**artitioned into intuitive epoch types which allow us to re-sample the data into regular time intervals. The size of partitions has an impact on the effectiveness of reduction and we must take care in selecting an appropriate resolution to partition our time series into, this will be discussed further in section 4.2.2.
3. Finally this aggregated, partitioned stream of data is **R**educed where the aggregated data is grouped depending on the requirements of the application.

The following sections describe the two irregularly sampled time series datasets that were used as case studies for the application of DAPRA. These are also summarised in

[2] *Tableau*, Seattle WA, Tableau Software. Available: https://www.tableau.com

Table 4.1.

(a) *Case Study 1: Retail Foreign Exchange Broker Client Trades*

Since the proliferation of mobile computing, a plethora of retail Foreign Exchange (FX) brokerages have led the way in enabling retail investors (or clients) from around the world to speculate on the FX market. The FX broker is the 'middle-man' – it links to the best liquidity providers (or LP's, such as investment banks), and the LP's streams 'trade-able' currency prices to the broker. The FX broker then passes these prices on to its thousands of clients worldwide all of whom can trade from their mobile phones or personal computers at the click of a button, investing as little as $100 or as much as $1,000,000 on each order. To open an account with the broker, retail clients must deposit initial funds and agree on a leverage ratio before they are allowed to trade, typically anywhere between 20-100 to 1. Thus an initial deposit of $1000 will allow the clients to place orders and enter positions up to $100,000. Once an account is live the client is free to speculate trading as many different currency symbols (e.g. GBP, USD, or EUR) as they wish provided they remain within the confines of their (leveraged) funds, in addition to any profit and loss (PnL) amounts. A typical retail broker will provide their clients with trading platform software such as MetaTrader 4 (MT4)[3]. Clients use the trading platform to place trades, monitor positions, track both historic and live movements in prices, and access the latest world economic news which influence and drive volatility in the markets. In the publicly available dataset [16] we apply the DAPRA framework to analyze the trades of 684 clients during January 2017. The client trades are in MT4 format and relate to 30 of the most liquid FX symbols. Table 4.1 outlines the main features of two data streams about our case study 1: *Client Trades* (the irregularly-sampled time series) and *Prices* (the 'exogenous' dataset in this case).

(b) *Case Study 2: New York City Taxi Rides*

Our second case study describes data about taxi journeys in NYC. The data was made available by C. Whong, using freedom of information laws to obtain the data from the NYC Taxi and Limousine Commission (TLC), [17].

The TLC provided data in the form of two separate data streams comprising information about individual taxi rides and taxi fare data throughout 2013 (*NYC*

[3] *MetaTrader 4 Trading Platform*, Metaquotes Software Corp. Available: http://www.metaquotes.net/en/metatrader4

Taxi Rides and *NYC Taxi Fares* - see Table 4.1). We used these data streams as the irregularly-sampled time series data for this case study. For practical reasons, we also chose to extract data from January 2013 only. The exogenous dataset for the second case study refers to the weather conditions in NYC over the same period. This data stream was published by D. Beniaguev on Kaggle [18] and provides an hourly snapshot of the weather conditions featuring the weather classification (such as "Heavy Rain" or "Clear Skies") as well as the temperature and humidity.

4.2.1 Aggregation

As previously discussed, the aggregation step of DAPRA involves merging the different data streams into one large dataset that effectively helps to explain or describe the actions of agents. Table 4.1 shows some key features of each case study's data streams that are aggregated and the derived fields that result. Differences in the number of fields and rows for each case study data stream illustrate clear differences in the irregularity of data sampled over the same time horizons. For example, the regularly sampled Market Prices data has 937,830 rows whereas the irregularly sampled Client Trades data has only 177,132 rows. This difference is what allows us to gain further insight into the trading behavior of a client. Furthermore, Table 4.1 shows how many newly derived fields result from simple computations performed on aggregated data streams. For our first case study, we can use aggregated Client Trades and Market Prices data to generate and track derived fields such as PnL and position at varying levels of granularity.

It is important to note the derived fields PnL and position are normalized to a common currency, in this case, USD. This allows for meaningful comparison between client trades and analysis of the broker's position. When a client enters a trade their position is evaluated in the base currency of the symbol they traded. We can therefore use the exchange rate between the base currency and USD (base multiplier) to calculate the position in USD. Likewise, PnL is quoted in the contra currency we use a similar method utilising the exchange rate between the contra currency and USD (contra multiplier) to calculate the PnL in USD.

4.2.2 Partition

The partitioning step serves to "bucket" the aggregated data into regular time epochs, allowing for effective analysis of each time epoch in detail. For this, we must decide the partition size which will vary depending on the data/problem domain. The partition

TABLE 4.1: Case Study Data Types

Case Study	Type of Data	Number of Fields	Number of Rows	Description of Data	Derived Field
Case Study 1: Trade Data	Client Trades	9	177,132	Detailed profile of each client order.	Profit and Loss (PnL)
	Market Prices	5	937,830	1 minute price data sampled for 30 major currency pairs.	Position
Case Study 2: Taxi Data	NYC Taxi Rides	9	14.7 mio	Detail of each taxi ride, with pickup and dropoff location.	Wait Time
	NYC Taxi Fares	4	14.7 mio	Fare prices for corresponding trip data	Location Latitude Location Longitude
	NYC Weather	1	8,760	Weather type sampled each hour.	Speed

size is ultimately the choice of the end-user, however, if it is too small there will not be any meaningful aggregation of the data. Similarly too large a partition size will not enable sufficient breakdown/analysis of the data.

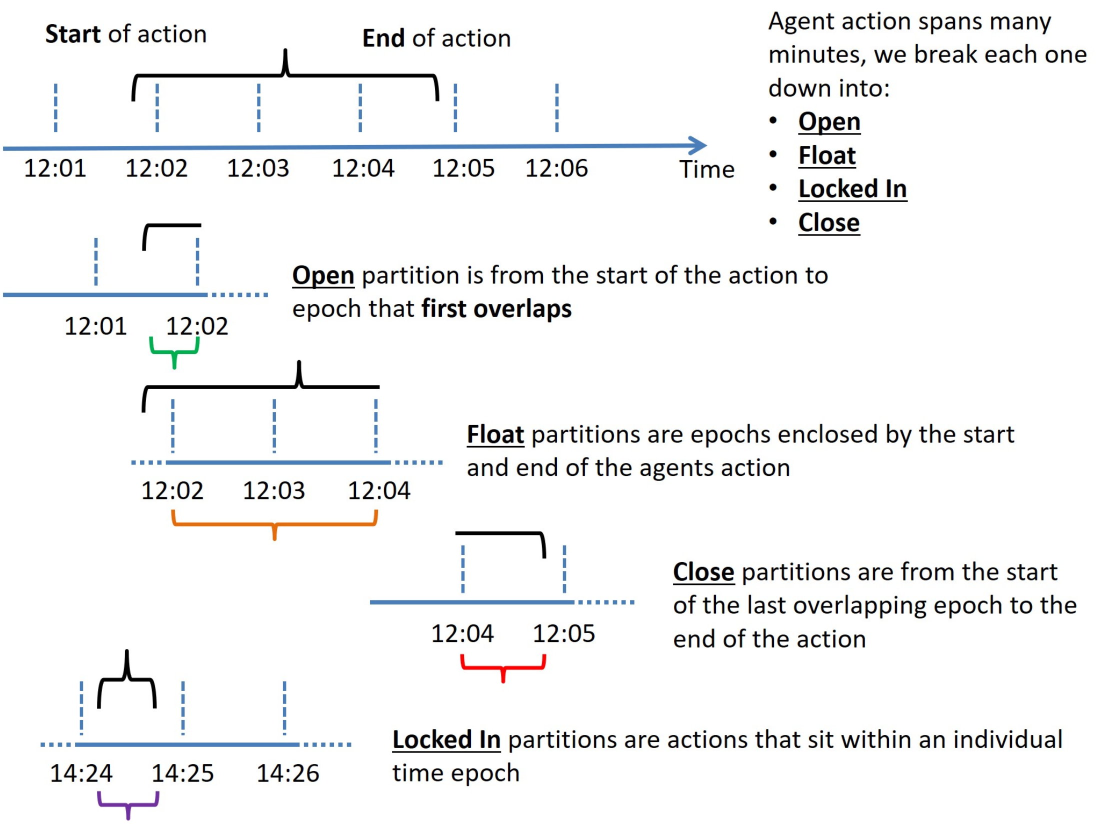

FIG. 4.3: DAPRA Partition Classification

There are four classifications of epoch partition which we can break each agent's actions into, we refer to these as *type parameters*, τ. Actions of a given agent can span many epochs depending on the resolution chosen and will affect the number of classification types in the resulting partitioned data, as illustrated in Fig.4.3.

The four type parameters can be defined as follows:

- **Open**: Actions opened in the current epoch.
- **Closed**: Actions closed in the current epoch.
- **Locked**: Actions opened and closed in the current epoch.
- **Float**: Actions that have been opened in a previous epoch and are still open at the end on the current epoch.

Note, that the definition of these epoch classifications is novel to DAPRA and is not related to any established data management methodologies.

Over an interval from t to $t + \Delta$ where Δ is the size of the resolution. Let OT and CT be the open and close times of an agent's action. We assign the type parameter, τ, using a method similar to Allen's interval algebra [19]. We can represent this logic in pseudo-code as follows:

Algorithm 6 DAPRA Pseudo Code

```
if OT < t then
    if CT ⩾ t + Δ then
        τ = float
    else
        τ = close
    end if
else
    if CT ⩾ t + Δ then
        τ = open
    else
        τ = locked
    end if
end if
```

As Table 4.2 shows, the choice of epoch resolution intuitively influences how each action breaks down into the different type classifications. For example, if the resolution is too small most of the actions will be classified as "floating", and as the resolution increases, an increasing number of actions are classified as "locked in".

We can also see how the choice of resolution depends on the problem domain for example, in the case of trade data a resolution of 2-2.5 days provides an optimal ratio between

TABLE 4.2: Number of epoch types at various resolutions

Case Study	Resolution	Open	Closed	Locked	Float
Case Study 1: Trade Data	5 days	42 k	42 k	135 k	29 k
	2.5 days	59 k	59 k	118 k	108 k
	1 day	70 k	70 k	107 k	330 k
	60 min	93 k	93 k	84 k	9.4 mio
	5 min	104 k	104 k	73 k	113 mio
Case Study 2: Taxi Data	1 day	0.1 mio	0.1 mio	14.3 mio	53
	60 min	2.7 mio	2.7 mio	11.7 mio	8 k
	15 min	9.2 mio	9.2 mio	5.2 mio	1.8 mio
	10 min	11.4 mio	11.4 mio	3.0 mio	5.1 mio
	5 min	13.6 mio	13.6 mio	0.8 mio	19.4 mio

the type parameters. Whereas for the taxi data, an optimal ratio is found at a resolution of around 10 minutes. This intuitively makes sense as taxi rides are shorter on average than trade holding periods. It is also noteworthy that our tolerance for the number of "locked" classifications varies from one problem domain to the next. For example, short trades under 10 minutes are fairly typical, whereas taxi rides under 1 minute are extremely scarce. Therefore, variance in the distribution of the length of agent actions is a key factor when deciding a sensible resolution to partition the data.

4.2.3 Reduction

Once the staging process is complete, the resulting target data is stored in an OLAP cube. This is depicted in Fig.4.4 where the Market Prices and Client Trade data are transformed into an OLAP cube with time, symbols, and clients as its dimensions.

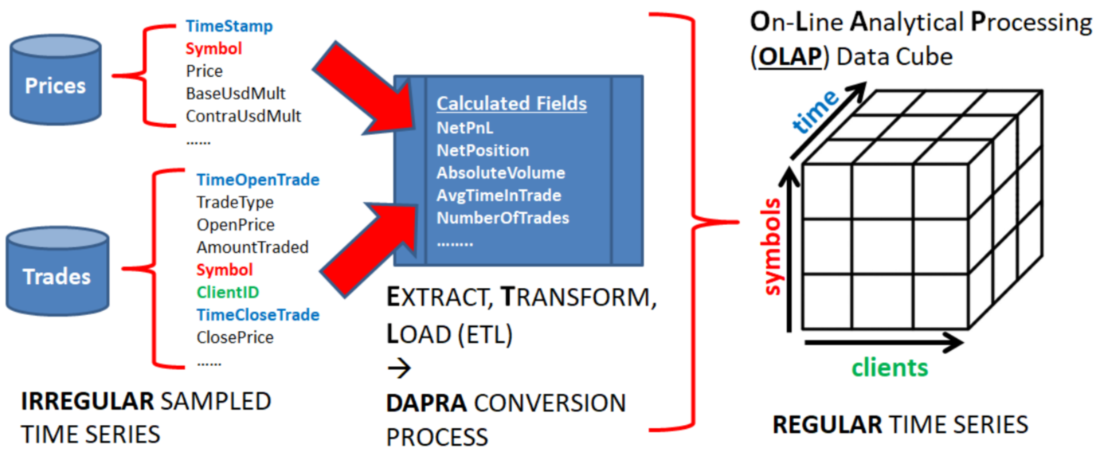

FIG. 4.4: DAPRA storage in an OLAP cube

The hierarchical dimensions of the OLAP cube allow for straightforward operations to be executed with relative ease, allowing for effective reduction of the data [20]. This step involves focusing on a specific discrete field that we use to regroup the data in a meaningful way. For example, for the taxi case study, we may want to break down the data by location of where pickups are taking place. Despite longitude and latitude being real numbers we can bucket them into rough locations rounding to 2 decimal places. We take the partitioned data and then group it by the relevant statistics that we want to compute such as fare amount, distance traveled, or weather type. The choice of which discrete field we want to group by and the resolution determines how many rows DAPRA finally compresses down to.

TABLE 4.3: Number of rows in target data using different groupings

Case Study 1: Trade Data			
Time Resolution	Group by client + symbol (20,520 Unique)	Group by client (684 Unique)	Group by symbol (30 Unique)
5 days	5,0635	4,264	210
2.5 days	99,661	8,799	411
1 day	217,539	20,267	934
60 min	4,780,691	465,308	21,605
5 min	57,095,054	5,571,898	258,778
Case Study 2: Taxi Data			
Time Resolution	Group by Location to 2 decimal places (3k unique)	Group by taxi license plate (32.5k unique)	Group by weather type (14 unique)
1 day	18,760	714,959	185
60 min	148,861	5,968,452	2,639
15 min	455,175	18,303,614	10,312
10 min	647,205	25,350,520	15,433
5 min	1,210,543	44,095,732	30,793

Table 4.3 shows the effectiveness of the data reduction step at various resolutions and with 3 choices of grouping for each case study. For example, grouping by location in the case of taxi journeys enables us analyse locations as zones rather than individual points which offers a more practical overview. In addition (and assuming a 1:1 relationship between taxi driver and taxi licence plate), grouping taxi journeys by license plate allows us to build a profile for each driver offering insights into their behaviour.

4.3 Case Studies

4.3.1 Case Study 1: Foreign Exchange Broker Data

We will now define the fields within each of the datasets for the foreign exchange broker data as described in Table 4.1.

The *Client Trades* data stream has nine fields defined as follows:

- OT: Open time is the time stamp at the open time of the order.
- ID: Is the unique identification number[4] of the client.
- TA: This is the total amount traded in the order, quoted in the base currency.
- SD: This is the side referring to the position the broker takes in the traded, 1 for long and -1 for short.
- SY: Is the symbol traded.
- ON: Is the number assigned to the order.
- OP: Open price is the price of the symbol at the time of the order.
- CT: Close time is the time stamp of the close time of the order.
- CP: Close price is the price of the symbol at the close time of the order.

An example of this from the source data is shown in table 4.4.

Table 4.4: Example of source client trade data

Open Time	Client	Amount	Side	Symbol	Order Id	Open Price	Close Time	Close Price
03/01/2017 01:01:33	B40	2000	-1	USD/CHF	B40_0	1.0223	03/01/2017 13:04:50	1.0308
03/01/2017 13:04:50	B40	3000	1	USD/CHF	B40_1	1.0308	11/01/2017 00:26:22	1.026
17/01/2017 06:43:26	B40	2000	-1	EUR/CHF	B40_27	1.0729	19/01/2017 00:31:22	1.0723

Next we have the *Market Prices* data stream, containing:

- TS: This is the time stamp at the time of the sample.

[4] This is an encrypted identification number.

- SB: Is the symbol being quoted.
- BM: This is the base multiplier, which refers the the exchange rate from the base currency to USD, used to calculate position in USD.
- CM: Is the contra Multiplier, referring to the rate between the contra currency and USD, used the calculate the profit and loss in USD.
- SP: Is the price of the symbol at the time of the observation.

Table 4.5 gives an example of the source market data.

TABLE 4.5: Example of source market price data

Datetime	Symbol	Open Usd Mult	Contra Usd Mult	Price
04/01/2017	USD/CHF	1	0.9745	1.0262
06/01/2017	USD/CHF	1	0.9903	1.0098
08/01/2017	USD/CHF	1	0.983	1.0173
10/01/2017	USD/CHF	1	0.9838	1.0164
12/01/2017	USD/CHF	1	0.9894	1.0107
16/01/2017	EUR/CHF	1.0627	0.9902	1.0734
18/01/2017	EUR/CHF	1.0632	0.993	1.0728
20/01/2017	EUR/CHF	1.07	0.9981	1.072

Now we will define the functions used to calculate the derived fields, PnL, and position.

$$PnL_t = \begin{cases} (SP_t - OP) \times TA \times SD \times CM & \text{if } \tau = open \\ (SP_{t+1} - SP_t) \times TA \times SD \times CM & \text{if } \tau = float \\ (CP - SP_t) \times TA \times SD \times CM & \text{if } \tau = close \\ (CP - OP) \times TA \times SD \times CM & \text{if } \tau = locked \end{cases}$$

$$POS_t = TA \times SD \times BM$$

It is important to note that the derived fields describe the PnL and position from the perspective of the broker. Table 4.6 shows an example of data populating the derived fields following the application of DAPRA to the data in Tables 4.4 and 4.5.

4.3.2 Case Study 2: NYC Taxi Journey Data

In this section, we will define the fields of the second case study as outlined in Table 4.1 .

Table 4.6: Example of target data after the application of DAPRA for case study 1

Order ID	client_ID	symbol	Type	Open	Close	PnL USD	Position USD	Side
B40_0	B40	USD/CHF	LOCKED	02/01/2017	04/01/2017	-16.559	-2000	-1
B40_1	B40	USD/CHF	OPEN	02/01/2017	04/01/2017	-13.532	3000	1
B40_1	B40	USD/CHF	FLOAT	04/01/2017	06/01/2017	-48.678	3000	1
B40_1	B40	USD/CHF	FLOAT	06/01/2017	08/01/2017	23.273	3000	1
B40_1	B40	USD/CHF	FLOAT	08/01/2017	10/01/2017	-9.4088	3000	1
B40_1	B40	USD/CHF	CLOSE	10/01/2017	12/01/2017	35.296	3000	1
B40_27	B40	EUR/CHF	OPEN	16/01/2017	18/01/2017	0.1327	-2139.23001	-1
B40_27	B40	EUR/CHF	CLOSE	18/01/2017	20/01/2017	0.9069	-2131.74999	-1

The *NYC Taxi Ride* data stream has nine fields as described below.

- ID: This is the encrypted license plate for each taxi.
- OT: This is the pick-up time of the trip.
- CT: Is the drop off time of the trip.
- TT: Is the total time of the trip.
- TD: Refers to the total distance of the trip.
- PA: The pick up longitude.
- PL: The pick up latitude.
- DA: The drop off longitude.
- DL: The drop off latitude.

Table 4.7: Example of source NYC taxi ride data

Encrypted license	Pickup datetime	Dropoff datetime	Trip Time in secs	Trip distance	Pickup longitude	Pickup latitude	Dropoff longitude	Dropoff latitude
ED7D81135978A1933D109A6434A7C563	13/01/2013 11:48	13/01/2013 12:10	1320	9.94	-73.98394	40.760399	-73.872818	40.77441
52830E3E5933F38822EF7A73A17C98C2	13/01/2013 12:05	13/01/2013 12:09	240	0.92	-73.983047	40.722694	-73.991402	40.729877
CFF79EB7D13568950B06C4DBC3329A8E	13/01/2013 12:01	13/01/2013 12:16	900	3.7	-73.973038	40.785358	-73.982933	40.745277

The *NYC Taxi Fares* data stream comprises five fields:

- ID: This is the encrypted license plate for each taxi.

- OT: This is the pick-up time of the trip.
- FA: This is the fare amount of each trip.
- TP: Is the tip amount giver to the driver.
- TO: This is the amount the passenger paid as a result of tolls encountered during the journey.

Table 4.8: Example of source NYC taxi fare data

Encrypted license	Pickup datetime	Fare amount	Tip amount	Tolls amount
ED7D81135978A193 3D109A6434A7C563	13/01/2013 11:48	30	0	0
52830E3E5933F3882 2EF7A73A17C98C2	13/01/2013 12:05	5	0	0
CFF79EB7D1356895 0B06C4DBC3329A8E	13/01/2013 12:01	14	1	0

Next, the exogenous *NYC Weather* data:

- TS: The Time-Stamp of the time the sample was taken.
- WL: This refers the weather conditions at the time of the observation as a Weather Label.

Table 4.9: Example of source weather data

Datetime	Weather description
13/01/2013 10:00	mist
13/01/2013 11:00	overcast clouds
13/01/2013 12:00	mist
13/01/2013 13:00	mist

Following DAPRA we can derive four new fields: (1) wait time, (2) interpolated pick-up/drop-off latitude, (3) interpolated pick-up/drop-off longitude, and (4) average speed for each taxi journey. These can be defined as:

$$
\begin{aligned}
WaitTime_t &= OT_{t+1} - CT_t \\
Speed_t &= \frac{TD_t}{CT_t - OT_t} \\
Loclat &= PL + \mathcal{P} \times (DL - PL) \\
Loclong &= PA \times \frac{1 - (Loclat - PL)}{DL - PL} + DA \times \frac{Loclat - PL}{DL - PL}
\end{aligned}
$$

where $\mathcal{P}$ = the fraction of the journey complete.

TABLE 4.10: Example of target data after application of DAPRA for case study 2

Encrypted license	Type	Open	Close	Fare	Duration	Distance	Location longitude	Location latitude	Weather	Wait Time	Avg.Speed
ED7D8113 5978A193 3D109A64 34A7C563	OPEN	13/01/2013 11:40	13/01/2013 11:50	2.727273	120	0.903636	-73.9839	40.7604	overcast clouds	01:00:00	27.10909
ED7D8113 5978A193 3D109A64 34A7C563	FLOAT	13/01/2013 11:50	13/01/2013 12:00	13.63636	600	4.518182	-73.9233	40.76804	overcast clouds	01:00:00	27.10909
ED7D8113 5978A193 3D109A64 34A7C563	CLOSE	13/01/2013 12:00	13/01/2013 12:10	13.63636	600	4.518182	-73.8728	40.77441	overcast clouds	01:00:00	27.10909
52830E3E5 933F38822 EF7A73A1 7C98C2	LOCKED	13/01/2013 12:00	13/01/2013 12:10	5	240	0.92	-73.9872	40.72629	mist	02:10:00	13.8
CFF79EB7 D1356895 0B06C4DB C3329A8E	OPEN	13/01/2013 12:00	13/01/2013 12:10	8.4	540	2.22	-73.973	40.78536	mist	00:30:00	14.8
CFF79EB7 D1356895 0B06C4DB C3329A8E	CLOSE	13/01/2013 12:10	13/01/2013 12:20	5.6	360	1.48	-73.9829	40.74528	mist	00:30:00	14.8

4.4 Discussion of Findings

Here we present and discuss a fraction of interesting findings following the application of DAPRA to the aforementioned case study datasets. We have purposefully focused on findings that have practical and commercial implications for end users in the financial and transport industries and have visually presented these using Tableau.

Our first case study aimed to gain insight into client trading behavior. Using DAPRA we analysed and compared the trades carried out by 684 clients during January 2017. Each client was allowed to buy or sell any of the 30 available unique currency pairs (e.g.

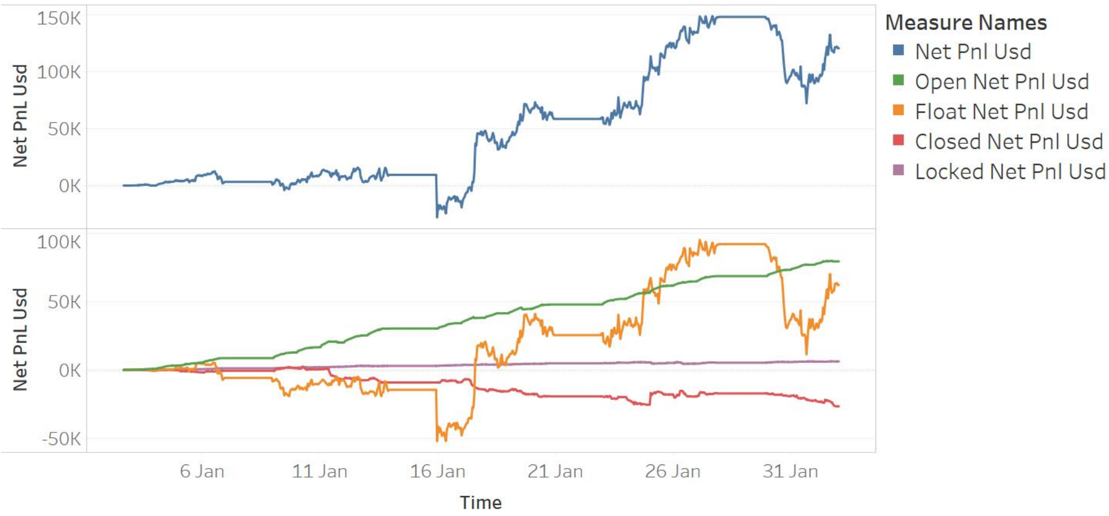

FIG. 4.5: **TOP:** Broker's Net PnL. **BOTTOM:** Broker's Net PnL grouped by the type parameter

EUR/USD) and they could place trades as many times as they wanted, at any time of day provided they stayed within the confines of their leveraged funds. During this time the clients' broker accumulates a position in the currency market which is essentially an aggregation of all of its client trades with the LP's. The broker position could be *long* (meaning clients are placing more sell trades), *short* (clients are placing more buy trades), or *flat* (the amount of sell and buy trades by clients are equal). Application of DAPRA enables the broker to visualize how its market position accumulates over time and then answer specific questions about it, as well as predict its position in the future.

Fig.4.5 shows the broker's Net PnL in USD over January with an interval time of one hour. The application of DAPRA allows the broker to track and analyze their PnL at a high resolution not possible from the source data alone. This strategic information allows the broker to examine the volatility and draw-down of their PnL, which in turn helps to inform hedging strategies. Fig.4.5 also shows the Net PnL of the broker grouped using the type parameter. This provides an interesting insight into client behavior: we notice the closed Pnl (in green) is negative, meaning the broker lost on the segment of trades labeled as closed. This pattern is repeated at various interval lengths, suggesting that in general clients tend to wait for positions to rise in value before closing them irrespective of the overall performance of the trade. We also see that the PnL of floating positions far outweighs that of locked positions, suggesting that client trading strategies have a greater potential to be profitable over a longer holding period.

The storage of the data using an OLAP cube combined with DAPRA staging allows for dynamic insight into individual client trading behavior and characteristics which were

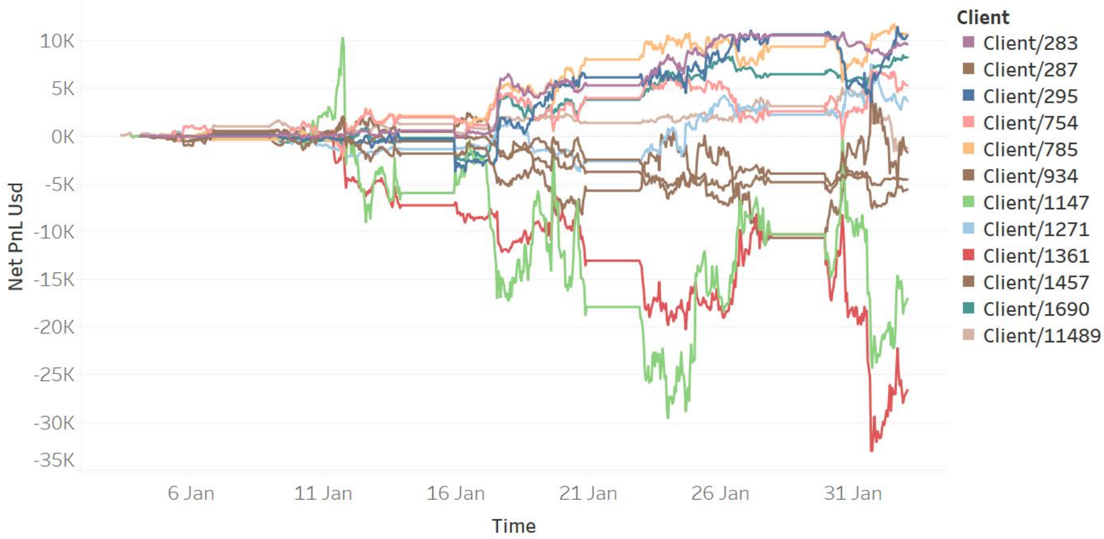

FIG. 4.6: Broker's Net PnL grouped by client

previously unseen. Following DAPRA, Fig.4.6 shows the net PnL of each client (again from the broker side), for clients with a final PnL of at least 6,000 USD. This allows the broker to identify clients that pose the greatest risk and those from whom they can profit. For example, clients 1147 and 1362 contribute significantly larger losses to the broker than average, thus the broker may decide to use a specific hedging strategy for all orders placed by these clients. This drilling-down of the data also offers the broker a comprehensive view of all trades made by their clients, from their open to their close. By pre-processing the data with DAPRA, we can now examine the variation in floating (or unrealized PnL) across the lifespan of each client trade and see at which point the client decided to close the trade and realize their PnL. This behind-the-scenes intuition into client decision-making is an invaluable resource for the broker that helps identify and differentiate successful and unsuccessful trading strategies.

Equally important is the broker's position - as previously discussed DAPRA can be used to calculate the position of all symbols in one common currency (USD). In Fig.4.7 we observe the broker's net position over the sample period, as grouped by symbol. A broker would undoubtedly find this information invaluable to ensure that it limits its exposure to unexpected changes in a given currency market. Such information becomes even more meaningful when it is continually re-evaluated according to the current exchange rate and at an interval length specified by the broker. This is all made possible with DAPRA. For example, the net position grouped by GBP/JPY (in blue in Fig.4.7) shows a steady accumulation over time which would likely compel the broker to flatten its position (to reduce its market exposure) by hedging against it. In another example, the net position grouped by EUR/USD (in grey) shows a dramatic switch from long to short during the

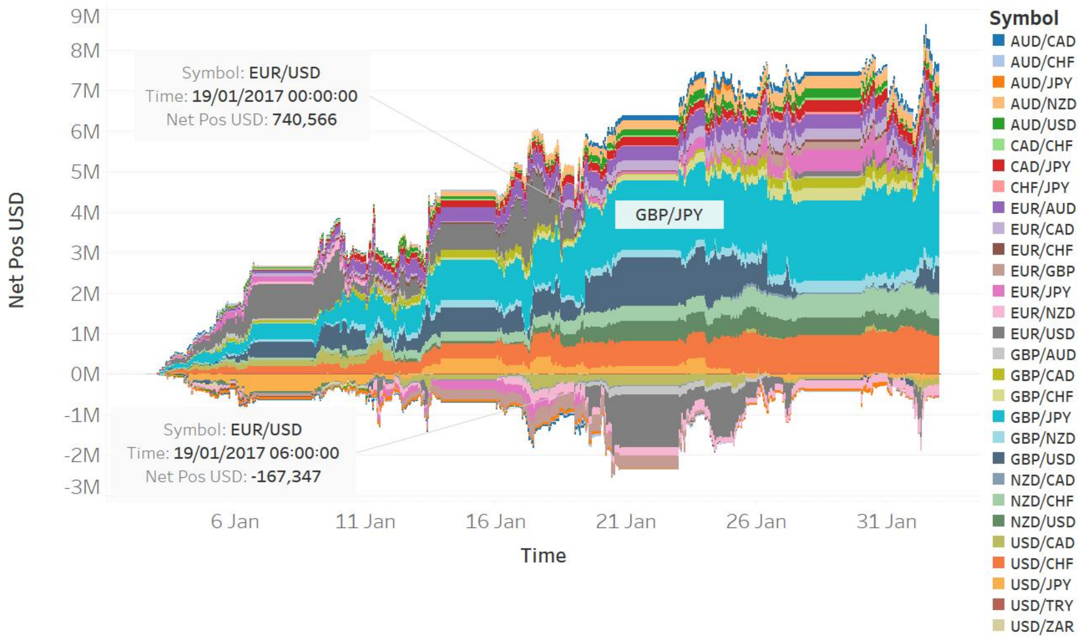

Fig. 4.7: Broker's Net position grouped by symbol

first 6 hours of 19th Jan 2017. This demonstrates that during this time, the clients' combined view of the market moved to predict a rise in the value of EUR/USD. This information would be of great interest to the broker and when combined with individual client PnL would offer considerable insight into client views of the market and what may drive them to open and close their positions.

Fig. 4.8 shows the application of DAPRA to data of the NYC taxi journey case study where 10-minute intervals have been used to partition the data, further grouped by the type parameter. Grouping of the partitioned data reveals some interesting aspects regarding taxi journeys which can be visualised by plotting the interpolated location over a map of NYC. For instance, we can see what appears as a small cluster in the bottom right corner of the open (green) and closed (blue) maps. This cluster maps to JFK airport and as the parameters of this cluster are present in open and closed types we would identify these locations to be airport pick-up/drop-off points (note this finding supports those of a 2018 study in [21]). This is to be expected however the versatility of DAPRA can be seen when we analyze the floating data groupings. This reveals a collection of location points from the island of Manhattan to JFK Airport. This data not only provides information on the taxis traveling to and from the airport but also classifies a section of land between Manhattan and JFK airport where a driver would have little chance of securing a pickup. Finally, the intensity of locked-in trips is highest in Manhattan. Intuitively this shows that taxi rides under 10 minutes are more common in built-up areas than the surrounding suburbs.

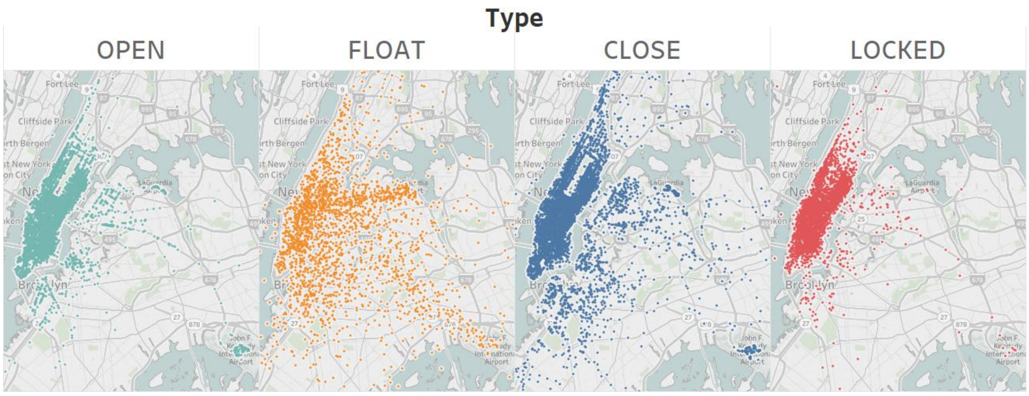

Fig. 4.8: Visualisation of taxi journeys in NYC during January 2013 for each of the four type parameters

Fig. 4.9 shows the results of grouping 10-minute partitioned taxi journey data with weather classifications. The groupings are as follows: average waiting around time, average tip amount, average toll amount, and average speed. Here we gain an intuitive insight into how taxi drivers and passengers react to changing weather conditions. Perhaps the most striking result is the correlation between the weather label and the average wait time from the last drop-off to the next pick-up. This grouping clearly shows taxi drivers experience far longer wait times between pickups during dry weather ("clear sky") than in wet weather ("wet"). This is probably because passengers are less inclined to require a taxi when the weather is good, yet demand for taxis increases when passengers want to avoid getting wet so waiting time between jobs is reduced. An equally intuitive reason for shorter waiting times is that drivers reduce their speed in bad weather, this is reflected in the DAPRA data showing a significantly lower speed of 27kph in wet weather.

Continuing with Fig.4.9, the DAPRA framework was used to analyze data about the tolls paid by passengers with weather condition data. The findings revealed a less intuitive story, suggesting that in foggy and misty conditions, taxi drivers are more likely to choose routes with tolls in place. This is accompanied well by the data on passenger tips which reveal an interesting finding. When the weather is poor (wet or foggy/misty) drivers can expect to receive higher tips, despite passengers being charged extra for tolls in foggy weather. However, in good weather when tips are generally lower, the choice of a driver to impose a toll charge on a passenger has a detrimental effect on their likelihood of receiving a tip. This is demonstrated by the correlations between average toll amount / average tip amounts with the broken clouds and clear skies weather groupings.

Finally, we used DAPRA to examine patterns of taxi journeys and their features (e.g. duration, fare paid) of individual taxi drivers across January 2013. These findings are illustrated in Fig.4.10. Here we grouped the data by taxi ID and used a bubble diagram to clearly visualise an obvious working pattern throughout time for each driver. The size of each bubble represents the distance of each trip and the vertical axis shows the computed fare of the whole trip. This concise representation of each driver's week provides valuable insights, not only can we see when a driver was active but we can also quickly assess the success of each day. For example, we can see driver "0A0B39F7A97A6CFAA62F11C4BDA6BBF8" consistently works a six-day week whereas, driver "0A1A0478120C8A7B0C035A06321D3B91" rarely works more that two days a week. We can also see which days are good days for drivers. For example, Monday 21^{st} of January was a slow day for driver "0A0B39F7A97A6CFAA62F11C4BDA6BBF8", differing from their usual busy days (described by a dense concentration of circles). On the other hand, driver "0A2A8EB88CDB6F1287C1DC86309DC047" had a fare of over $200 on Friday 11^{th} of January, which from looking at the size of the circle is clearly a result of an unusually long trip distance.

4.5 Conclusions

Here we presented DAPRA - Data Aggregation Partition Reduction Algorithm - to manage, examine, and uncover insights into the time-sequenced actions of agents in an irregular time series. DAPRA originated from our need to find a framework capable of regularising data into a format more amenable to visual analysis and machine learning techniques. We used two case studies to demonstrate the practical application of DAPRA.

Our first case study applied DAPRA in an attempt to better understand the factors influencing client trading behaviors and the downstream effects of these on the broker's PnL and position. We demonstrated how using DAPRA enabled us to progress from the painstaking task of wading through millions of irregularly-sampled client trades, to the generation of a detailed and meaningful data stream that when visualized provided commercially-valuable information regarding client/broker positions and PnL over time. DAPRA requires some exogenous data stream and in this study, we referred to a simple one that described currency prices over the same period as the client trades. However, it

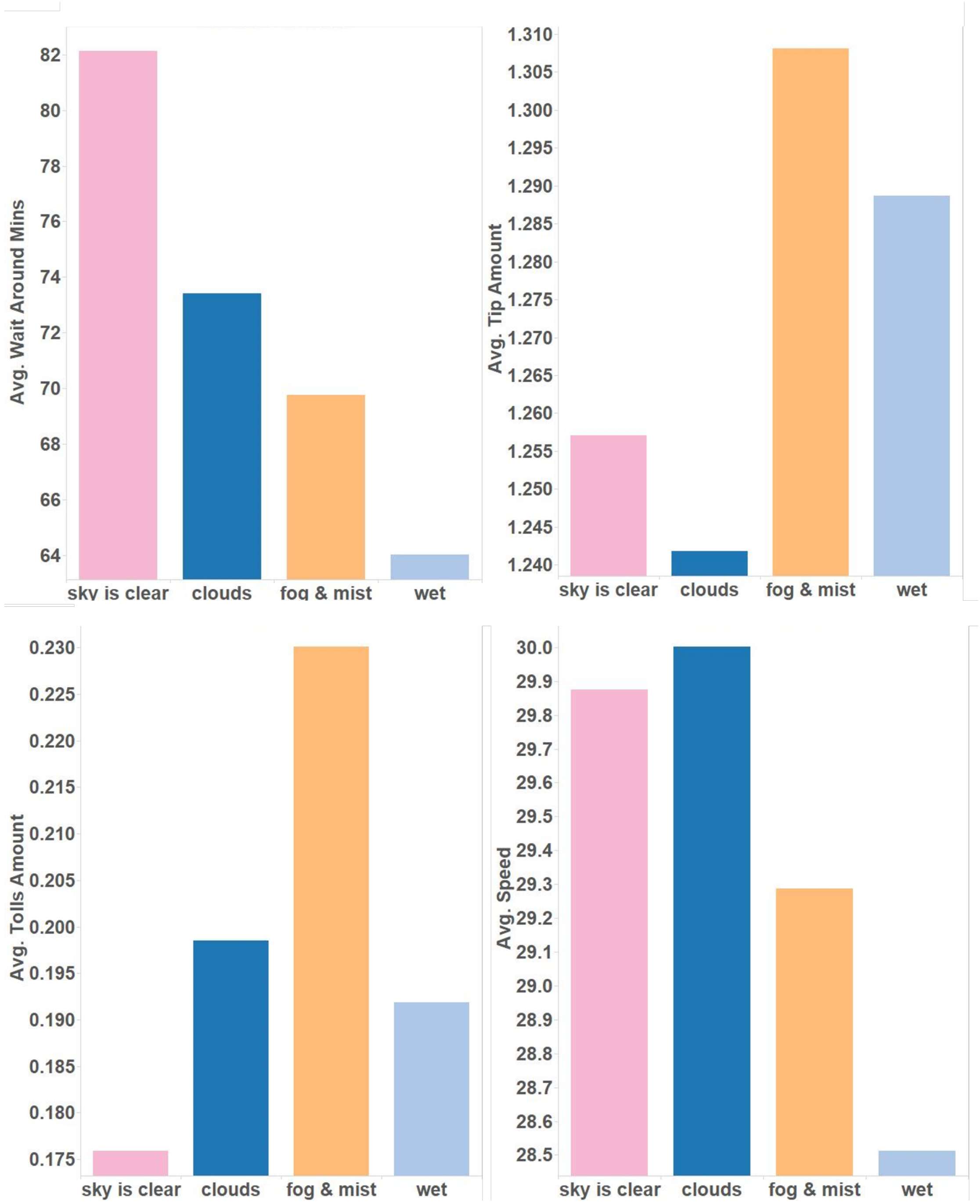

FIG. 4.9: Grouping taxi journey data by weather

is important to note that many more exogenous data streams could have been aggregated with. For example, one major driver of intra-day pricing volatility in the FX market is the economic release calendar [22]. This calendar dataset is known ahead of time and is essentially a schedule outlining when the world's major economies will publish their latest macroeconomic data. Sometimes estimated figures for economic announcements are provided in the calendar too. For example on the first Friday of every month, most traders will await the release of so-called "Non-Farm Payrolls" (average earnings) which

FIG. 4.10: Visualising patterns of taxi journey information for individual taxi drivers

act as a major indicator of the strength of the US economy. The authors have investigated in detail the use of economic release data to provide a deeper understanding of the underlying market conditions through time, adding features such as counting how many high-importance USD and EUR economic releases happened in the previous and next hour. Given such data, we could identify clients that trade in and around specific economic releases and separate these from clients that are trading based on longer-term price movements. This would provide even further insights into client trading behavior.

In our second case study, we looked at NYC taxi journeys during January 2013 and attempted to correlate this activity with a weather dataset. Intuitively, a taxi company will want to know the best approaches to allocate its fleet of taxis to avoid too much downtime, meet demand and thus maximize profitability. Even on an an individual level, a taxi driver may want to leverage such information for example to know where best to drive to given specific weather conditions to maximise their chances of getting follow-on fares in the next hour. It should be noted that we did make a crude assumption about the taxi journey dataset so that it would fit within the DAPRA framework. This assumption was to interpolate the location through time linearly between pickup and drop-off. One improvement to this would be to use a more realistic path through the city using actual road map data using algorithms such as Google's Poly Lines service[5], or better still in the future have full telemetry samples of each taxi ride throughout the journey such as the data used in [23]. It is even conceivable that with smartphone app-based taxi hailing / booking services such as Uber[6] we also could collect and use encoded customer passenger data and link this with taxi ID and journey data. This would open up another rich set of agents that we could cross-correlate behaviors with, perhaps even recording which journeys the taxi drivers had to pick up and which at any given time they chose to carry out.

One area for future work relates to the choice of optimal time resolution during the partition step of DAPRA. This choice has to consider: (1) the nature of the raw data (low/high frequency, origin, and size) and (2) the type of analysis we intend to carry out, be it forecasting, visualization, or guiding some actions. In each of our case studies we decided on arbitrary equal width time intervals, indeed other studies such as [24] and [25] look at methods of using variable width intervals that increase sampling during more active regions in the time series. In other work such as [26], [27], [28], the bin intervals are chosen using an iterative process that results in an interval with the lowest forecast error. Future implementations of DAPRA could look to use such techniques of variable width time intervals to see if better performance can be achieved.

In future work, we aim to take DAPRA-structured datasets and use them for online machine-learning tasks. The beauty of the DAPRA approach is that it lends itself perfectly to training classical online machine learning techniques [29]. Classification or regression labels can be easily constructed using features of future DAPRA data instances and detailed features can be derived from the Open/Float/Locked In/Closed

[5] Google Developers, "Geometry Library", Available: https://developers.google.com/maps/documentation/javascript/geometry

[6] *Uber*, Uber Technologies Inc. Available: https://www.uber.com

formulation of the statistics. For example, in our recent work, we have used DAPRA in portfolio optimization models. This is achieved by grouping orders by client and treating each client as an "expert" that predicts the future state of the market. Furthermore, we are currently researching the use of online machine learning techniques such as the aggregating algorithm [8] - early results show promising potential to outperform the market.

By referring to time series data from two distinct and complex domains we have tried to present DAPRA as a universal framework for all sorts of time series data streams. Time horizons can be of any size provided agent actions with defined start- and stop-times can be identified and agents are free to perform their actions as and when they choose to. Furthermore, agents do not need to relate to human-based behaviors such as trading or taxi driving. Indeed in other work, we have analysed network traffic data, where the agents are IP addresses of different machines on a computer network and the duration of sending data is measured in nanoseconds. In this era of Big Data, there is certainly no shortage of data streams that DAPRA can be applied to both now and in the future. As our world becomes increasingly connected with smart devices in homes and cities, the "sea" of tracked data measurements that results remains ever-expanding. It is an exciting time to find out how using frameworks such as DAPRA, we will use insights into agent behavior to our benefit.

Chapter 5

Practical Investment with the Long-Short Game

5.1 Introduction

Since modern portfolio theory was first introduced by [30], the problem of portfolio selection has become increasingly prominent. We approach this problem using the framework of on-line learning and apply methods of prediction with expert advice, where an investor makes investment decisions based on the observations of a pool of investors' strategies.

A well-known formalization of the investment process is Cover's game (section 2.2.1), where an investor partitions the available money between the assets. [2] construct a universal investment strategy for Cover's game: it performs nearly as well as any constant rebalanced portfolio. This approach is a special case of the more general aggregating algorithm, which is capable of combining a finite or infinite pool of portfolios.

The aggregating algorithm can be applied to a general problem of prediction with expert advice and is an evolution of the weighted majority algorithm introduced by [31]. The aggregating algorithm was developed by [1, 4] to include a more general concept of a loss, function on prediction and outcome spaces. Given a series of predictions from a pool of experts, a learner following the aggregating algorithm assigns to each expert weights quantifying its trust in them and then combines experts' predictions according to the weights.

The framework of Cover's game is popular in mathematical finance, but it is very restrictive and does not capture important aspects of trading such as short positions and investments on a margin. [3] took steps to consider more realistic trading scenarios. A trader does not partition their money between the assets as in Cover's game. Instead, they open positions, long and short, within some limits set by the exchange or the intermediary providing market access. [3] introduce a modification of Cover's game, namely, the long-short game (see Section 2.2.2). It admits long positions exceeding the trader's capital (within specified limits) and short positions. A major feature of this framework is the possibility of bankruptcy. While in Cover's game, the investor may lose all their capital only in a totally unlikely event of all stocks simultaneously plunging to zero, with the long-short game losing all the money is a much more realistic prospect.

In this chapter, we apply the aggregating algorithm to the long-short game in the case of the currency exchange market. The experts are based on the trading activity of 100 clients who used demo trading accounts to trade so-called basket orders of 55 of the most liquid currency pairs during the period from September 2019 to January 2020. We describe a method of deriving predictions from raw trade data using the data staging algorithm DAPRA [32]. We evaluate the performance of the aggregating algorithm at the long-short game and in Section 5.4 we propose modifications aimed at improving the practical performance of the resulting portfolio.

Substantial literature exists on applications of prediction with expert investment advice, but it usually concentrates on Cover's game with no short positions or uses different techniques and approaches. [33] and [34] carry out extensive computational experiments with universal strategies competitive with constantly rebalanced portfolios (no short positions are allowed). [35, 36] consider portfolio selection methods based on weak aggregating algorithms merging finite and infinite pools of experts. In their computational experiments, real stock market data is used but without short positions.

[37] consider universal investment strategies involving short positions and carry out computational experiments. The methods used by [37] to construct universal strategies are based on calibration and defensive forecasting. [38] apply a different class of prediction with expert advice methods, namely, AdaHedge-type algorithms, to stock trading in a different context. The algorithms are used to predict stock values and then predictions are fed to other automated trading algorithms.

The organization of this chapter is as follows. Section 5.2 introduces a novel data set based on the currency market trades of 100 clients over 4 months. We then describe the application of the aggregating algorithm on the long-short game, resulting in a portfolio with unimpressive results and motivates us to propose modifications to the loss function which are detailed in Section 5.4. In Section 5.5 we apply the modified method to the data and discuss the improvements made.

5.2 Data Set

The data we are using is derived from the basket orders of 100 clients using demo trading accounts over 4 months, from September 2019 to January 2020. The data is representative of the behavior of investors trading in the currency exchange market over a relatively calm period. A basket order allows a group of financial market instruments to be traded simultaneously. Different weighting criteria for different instruments can be used to tailor the basket according to the client's needs. Clients can either trade their baskets manually or use automated models. In this data the clients construct their baskets from the 55 most liquid Foreign Exchange (FX) pairs, as shown in Table 5.1.

TABLE 5.1: 55 FX (currency) pairs, the symbol format is a pair of currency codes delimited by a "/", where the currency code is in the ISO 4217 format.

AUD/CAD	EUR/AUD	EUR/SGD	HKD/JPY	USD/DKK	CAD/CHF	EUR/JPY	GBP/JPY	NZD/USD	USD/RUB	AUD/USD
AUD/CHF	EUR/CAD	EUR/USD	MXN/JPY	USD/HKD	CAD/JPY	EUR/MXN	GBP/NZD	SGD/JPY	USD/SEK	EUR/HKD
AUD/JPY	EUR/CHF	GBP/AUD	NZD/CAD	USD/JPY	CAD/SGD	EUR/NZD	GBP/SEK	USD/CAD	USD/SGD	GBP/HKD
AUD/NZD	EUR/DKK	GBP/CAD	NZD/CHF	USD/MXN	CHF/JPY	EUR/PLN	GBP/SGD	USD/CHF	USD/TRY	NZD/SGD
AUD/SGD	EUR/GBP	GBP/CHF	NZD/JPY	USD/NOK	CHF/SGD	EUR/SEK	GBP/USD	USD/CNH	USD/ZAR	USD/PLN

Table 5.2 illustrates raw trade data from the basket orders of four different clients, B1, B2, B3, and B10. We see client B1 has a basket trading 4 different currency pairs (NZD/SGD, GBP/SGD, NZD/CAD, and CHF/JPY) where each block of trades all have the same opening timestamp, for example, 24^{th} Oct 2019 at 07:08 and holds that position for 3 days. Later that same day at 19:45 we see client B1 trades the same basket again, so building on their position. "Position" is the summation of the client's trades and at a given point in time is described as being long, flat, or short. At 19:45, client B1's position goes long in NZD/SGD by 27,000, and short in NZD/CAD by 41,000. Table 5.2 further demonstrates that clients have the freedom to trade any combination of currency pairs and with any notional weightings they desire for their baskets (these can be derived manually or using proprietary algorithms). For example, client B3 trades a basket of 5 different currency pairs, whereas client B2 trades larger notional preferring GBP currency crosses. Client B10 solely trades 3 symbols which are all USD crosses.

Before we can apply the AA to this data we must first:

- Normalise client positions into a common currency, in our case we use USD. We do this because clients trade many different currencies, all of whose notional values differ through time. Therefore, to compare the positions and derive a price vector they must be normalized.
- Sample the data at regular time intervals (for this data we chose a resolution of 1 minute) across all clients and currency pairs. This is because whilst clients are at liberty to trade and hold positions for however long they wish, the AA must predict the future behavior of the market at regular time intervals.

TABLE 5.2: Raw trade data taken from basket orders of four different clients (B1, B2, B3, and B10) on the 24^{th} Oct 2019. Each trade has an open and close timestamp and corresponding open and close price. Whether the trade was a buy or sell is denoted by a 1 or -1 sign, respectively.

Open Time	Open Price	Client	Amount	Sign	Symbol	Order Id	Close Time	Close Price	Mins In Trade
2019.10.24T07:08:00.000	0.87236	B1	27,000	1	NZD/SGD	B1_87	2019.10.27T22:12:00.000	0.86643	5,224
2019.10.24T07:08:00.000	1.76210	B1	6,000	-1	GBP/SGD	B1_267	2019.10.30T00:20:00.000	1.75285	8,232
2019.10.24T07:08:00.000	0.83763	B1	41,000	-1	NZD/CAD	B1_447	2019.10.30T00:20:00.000	0.83112	8,232
2019.10.24T07:08:00.000	109.76200	B1	8,000	1	CHF/JPY	B1_634	2019.10.27T22:12:00.000	109.30400	5,224
2019.10.24T13:36:00.000	9.61360	B3	7,000	-1	USD/SEK	B3_64	2019.11.04T10:38:00.000	9.64949	15,662
2019.10.24T13:36:00.000	0.93133	B3	30,000	-1	AUD/SGD	B3_230	2019.11.03T22:12:00.000	0.93782	14,916
2019.10.24T13:36:00.000	2.01415	B3	2,000	-1	GBP/NZD	B3_397	2019.10.27T22:27:00.000	2.01888	4,851
2019.10.24T13:36:00.000	1.74062	B3	9,000	-1	EUR/NZD	B3_573	2019.10.28T03:50:00.000	1.74552	5,174
2019.10.24T13:36:00.000	19.08875	B3	1,000	-1	USD/MXN	B3_746	2019.11.05T07:47:00.000	19.13142	16,931
2019.10.24T14:58:00.000	10.07130	B2	451,000	1	GBP/HKD	B2_25	2019.11.01T09:23:00.000	10.09472	11,185
2019.10.24T14:58:00.000	8.70289	B2	59,000	-1	EUR/HKD	B2_84	2019.10.25T11:04:00.000	8.71086	1,206
2019.10.24T14:58:00.000	1.27429	B2	338,000	-1	GBP/CHF	B2_144	2019.10.25T14:54:00.000	1.27440	1,436
2019.10.24T14:58:00.000	12.38930	B2	19,000	-1	GBP/SEK	B2_202	2019.10.25T11:19:00.000	12.40480	1,221
2019.10.24T15:03:00.000	0.99168	B10	167,000	-1	USD/CHF	B10_22	2019.10.27T22:12:00.000	0.99454	4,749
2019.10.24T15:03:00.000	3.85160	B10	95,000	-1	USD/PLN	B10_91	2019.10.27T22:12:00.000	3.85830	4,749
2019.10.24T15:03:00.000	6.72958	B10	749,000	1	USD/DKK	B10_161	2019.10.28T08:11:00.000	6.73906	5,348
2019.10.24T19:45:00.000	0.86990	B1	27,000	1	NZD/SGD	B1_88	2019.10.27T22:12:00.000	0.86643	4,467
2019.10.24T19:45:00.000	1.75211	B1	5,000	-1	GBP/SGD	B1_269	2019.10.30T00:20:00.000	1.75285	7,475
2019.10.24T19:45:00.000	0.83419	B1	41,000	-1	NZD/CAD	B1_449	2019.10.30T00:20:00.000	0.83112	7,475
2019.10.24T19:45:00.000	109.47500	B1	9,000	1	CHF/JPY	B1_636	2019.10.27T22:12:00.000	109.30400	4,467

[32] introduced the data staging technique DAPRA (Data Aggregation Partition Reduction Algorithm) which, when applied to data streams about client trades and market data, allows one to normalize and sample the data as required for this study. DAPRA follows a three-step process:

1. **D**ata **A**ggregation, where data from one or more sources of irregularly sampled time series data are combined into a regular sampled time series. Here this will be the client trade and market price data.

2. The aggregated stream is **P**artitioned into regular time intervals. It is also at this stage where derived fields such as position are calculated, using features from both data sets.

3. Finally this aggregated, partitioned stream of data is **R**educed where the aggregated data is grouped based on client and symbol in this study.

Fig. 5.1 compares the net positions of the first 10 clients in the data set, following DAPRA transformation over the trial period. We can see here that clients have different trading strategies and activities through time, some clients build a position steadily over days and weeks, such as B3 and B4. Whereas others such as B6 and B9, open and close fixed amounts over short periods. All clients trade different amounts resulting in net positions between long and short 5 million across all 55 symbols shown in table 5.1. The positions in Fig. 5.1 show step changes when trades of basket orders are placed. Returning to our earlier example, we can see the shifts in position related to the trades in

Table 5.2 on the 24^{th} Oct 2019. Hence, we see that client B10's basket order comprises a sell USD/DKK trade which means a position change from flat to short. Importantly we can see great variability in the notional sizes, basket symbol composition, and holding periods of the different clients, producing are varied range of investment strategies.

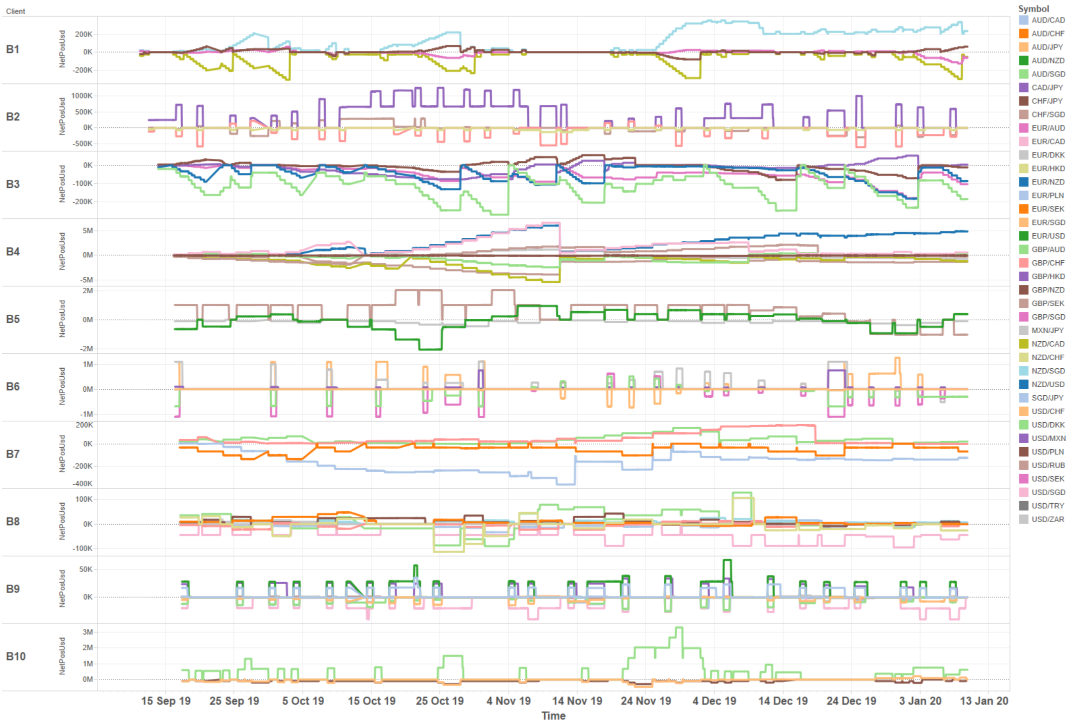

FIG. 5.1: Net positions of first 10 clients in data set from Sept 2019 to Jan 2020.

Until now we have assumed the existence of the portfolio vector γ, describing an expert's investment decisions. However, as we can see from the example trading data presented in Table 5.2, it is not clear how to define a client's prediction with that on the long-short game. We must therefore define a method of calculating an investor's portfolio vector from raw trade data that describes their investment decision over each time interval. The portfolio vector γ describes the sizes of investors' positions in relation to their wealth. This requires knowledge of the investor's wealth however, as is common we do not have access to the total funds available to an investor. Instead, we can assume that at each trial the investor has invested their total funds across each of the available assets. The DAPRA data set provides us with the normalized positions an investor held in each asset at the start of each interval where we will use $\text{Pos}_t^\theta[n]$ to denote the position of investor θ in asset n at time t. Therefore, a natural method of approximation is to define the portfolio vector $\gamma_t^\theta \in \mathbb{R}^M$ as $\gamma_t^\theta[m] = \text{Pos}_t^\theta[m] / \sum_{m=0}^{M-1} |\text{Pos}_t^\theta[m]|$

The data set is available online [39].

5.3 Empirical Results

Here we will compare the portfolio performance of an investor following the investment protocol of the long-short game applied to the 100 clients to give each client a fixed equal weight. We will make the assumption all assets within our market are arbitrarily indivisible and no transaction costs are present. Naturally, as we are using historical market data it is implicit our trading behavior does not affect the market. A practical performance measure of a portfolio is the return on investment (ROI), assuming no transaction cost we can calculate from the wealth of the investor using $\text{ROI} = (W_T - W_0)/W_0 \times 100 = (W_T - 1) \times 100$. This is equal to $e^{-\text{Loss}_T(\text{AA}(\eta, P_0))} \times 100$ where $\text{Loss}_T(\text{AA}(\eta, P_0))$ is the total loss w.r.t the long-short game loss following the AA with learning rate η and initial distribution P_0.

After partitioning our data into 1-minute intervals the result is 123596 trials. In Fig. 5.2 we can see the number of awake experts at each trial. Here an awake expert is defined as one that has an open position in at least one symbol over a given trial, otherwise, they are defined as sleeping. We can see it takes around 60,000 trials for the number of awake experts to stabilize. This is not an uncommon problem to face in the real world in cases such as an emerging market maker or the introduction of a new investment instrument. To keep the comparison between the various implementations of the aggregating algorithm fair, we will be ignoring the initial 60,000 trials from our experiments, however, the full data set has been provided.

The percentage return on investment (ROI) to the portfolio of the long-short game is 0.6449% as we can see in Fig. 5.3 this is good growth over the investment period. However, comparing this to assigning all experts equal weights we see an ROI of 0.6447%, this is a minor change in the ROI. In Fig. 5.4 we plot the excess ROI to the long-short game compared to equal weights, While the difference is small we do see a clear indication that AA has the potential superior predictions than simply following each expert equally.

Whilst the long-short game appears to be an improvement over using equal weights the gains are small with the difference in ROI after 63,596 trials being 0.002%.

The resulting ROI close to 1% is disconcerting compared to the results of the experts. The returns of the experts range from -2.33% to 4.62% with the mean of 0.64%. Clearly, AA fails to align itself with the best experts.

In terms of Lemma 2.3 this may be explained as follows. The total losses of the experts range from -0.05 to 0.020. The extra term in (2.4) with $\eta = 1$ and uniform initial weights of $P_0(\theta) = 1/100$ equals 4.6052. The guarantees of Lemma 2.3 are thus very loose.

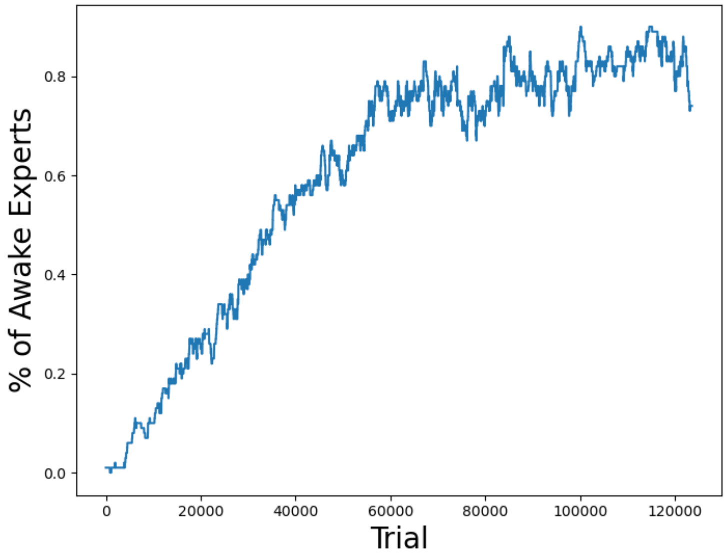

Fig. 5.2: Percentage of Awake Experts Through Time

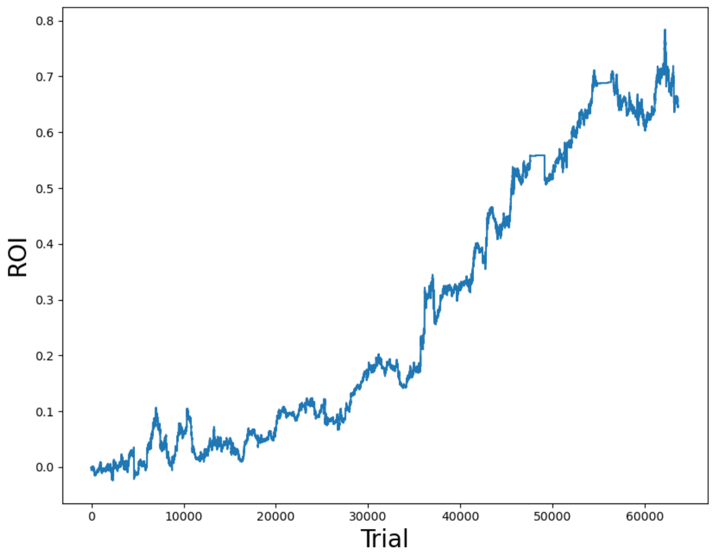

Fig. 5.3: Long Short Game ROI

In Fig. 5.5 we can see the weight the AA assigns to each client in the long-short game, at each trial. As client's weights are updated using their loss at each trial, the weight is a reflection of the ROI. Therefore, showing there are clients in the pool with returns far greater than those achieved by the AA. We see the game does differentiate between the various strategies however, there appears to be insufficient discrimination of weights to allow above-average strategies to influence the overall investment decisions of the AA. In this case, the model's final weights have a mean of 1.01 with a standard deviation of

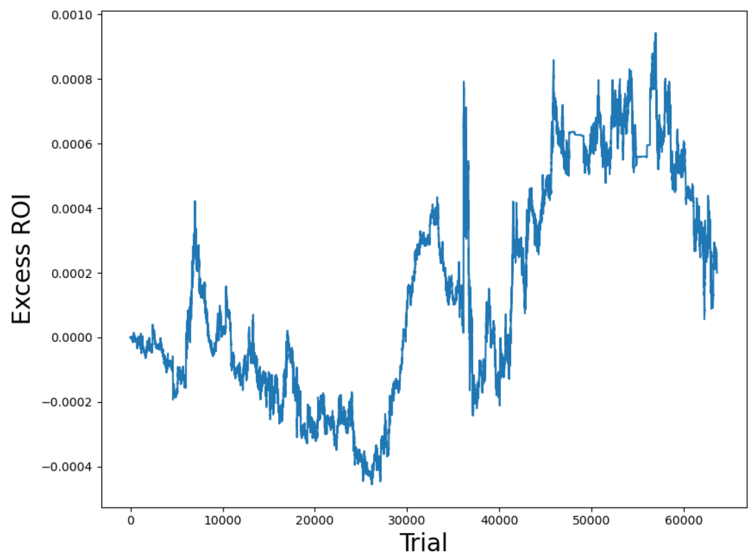

FIG. 5.4: Long Short Game Excess ROI

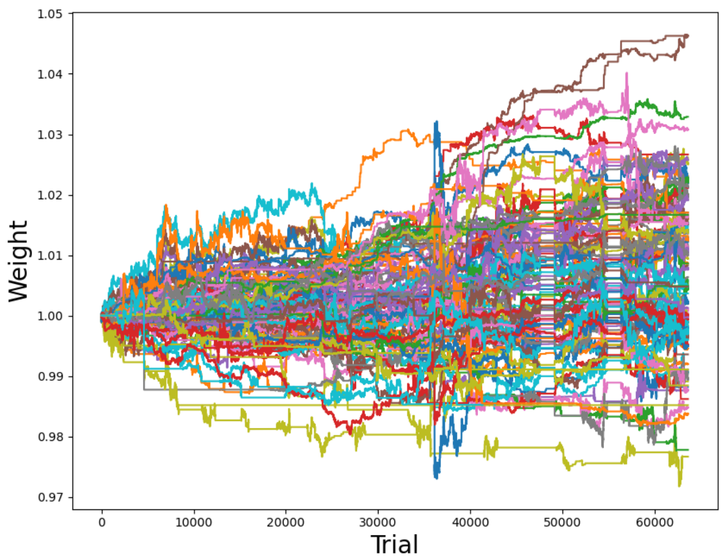

FIG. 5.5: Long-Short Game Client Weights

0.013 and with maximum and minimum weights of 1.04 and 0.98, this may explain the limited performance improvements of the long-short game. Therefore, it seems logical the performance of the long-short game may be increased by modifications to the game that increase the discrimination between the weights of each investment strategy.

5.4 Long-Short Game Modifications

In this section, we introduce several modifications in order to improve the practical performance.

The ideas developed here stem from the following intuition. While Cover's game has a natural scaling (the components of γ sum to 1), the Long-Short game does not. In the example we considered above, AA produces vectors γ_t that lead to a very small but positive profit. One can multiply these vectors by a factor of $A > 1$ and the profit will increase. Where the investor earned the profit of $W_{t-1}\langle\omega_t, \gamma_t\rangle$, they will earn $W_{t-1}\langle\omega_t, A\gamma_t\rangle = AW_{t-1}\langle\omega_t, \gamma_t\rangle$.

The downside of this is risk. Larger positions can potentially lead to bankruptcy. In the spirit of prediction with expert advice, we can analyze the possibility w.r.t. the bankruptcy of experts.

We will call a prediction γ satisfying (2.8) a-bounded.

Definition 5.1. A method of merging experts' predictions is *a-conservative* if for all experts' predictions $\gamma(\theta)$ if $\gamma(\theta)$ is a-bounded for all $\theta \in \theta$, then the prediction γ produced by the method is a-bounded.

Note that in practice being a-bounded is neither necessary nor sufficient for avoiding bankruptcy. An investor may take a calculated risk and get away with it.

Definition 5.2. A method of merging experts' predictions is *conservative* if for all experts' predictions $\gamma(\theta)$ the prediction γ produced by the method is such that for all $\omega \in \Omega$ if the values $-\ln(1 + \langle\omega, \gamma(\theta)\rangle)$ are uniformly bounded from above by a finite number, i.e., $-\ln(1 + \langle\omega, \gamma(\theta)\rangle) \leq C < +\infty$ for all $\theta \in \Theta$, then the value $-\ln(1 + \langle\omega, \gamma\rangle)$ is finite, i.e., $-\ln(1 + \langle\omega, \gamma\rangle) < +\infty$.

Theorem 5.3. *Any merging algorithm outputting an average of experts' predictions w.r.t. some distribution, i.e., $\gamma = \int_\Theta \gamma(\theta)P(d\theta)$, where P is some distribution, is conservative and a-conservative for all $a > 0$.*

Proof. If $-\ln(1 + \langle\omega, \gamma(\theta)\rangle \leq C < +\infty$ for all $\theta \in \Theta$, then $1 + \langle\omega, \gamma(\theta)\rangle \geq 2^{-C} > 0$ and

$$1 + \langle\omega, \gamma\rangle = 1 + \left\langle \omega, \int_\Theta \gamma(\theta)P(d\theta) \right\rangle = \int_\Theta (1 + \langle\omega, \gamma(\theta)\rangle)P(d\theta) \geq \int_\Theta 2^{-C}P(d\theta) > 0$$

if $\|\gamma(\theta)\|_1 \leq a$, then

$$\left\| \int_\Theta \gamma(\theta)P(d\theta) \right\|_1 = \sum_{m=0}^{M-1} \left| \int_\Theta \gamma(\theta)[m]P(d\theta) \right| \leq \sum_{m=0}^{M-1} \int_\Theta |\gamma_t(\theta)[m]|P(d\theta) \leq \int_\Theta aP(d\theta) = a$$

□

We can see that whenever the experts' predictions satisfy a broker's safety requirements, so do the AA predictions, and whenever the experts' predictions do not lead to bankruptcy, neither does the AA.

In this section, we will introduce a modification of the Long-Short game keeping this property w.r.t. the original λ_{LS} and improving on the practical performance of the AA.

5.4.1 Return Scaling

Take a number $\rho > 0$ and consider the loss function

$$\lambda_{\mathrm{LS},\rho} = \begin{cases} -\ln(1+\rho\langle\omega,\gamma\rangle) & \text{if } 1+\rho\langle\omega,\gamma\rangle > 0 \\ +\infty \text{ otherwise} \end{cases} .$$

One can define the a-bounded and general versions of the game with the same Γ and Ω as the original Long-short game.

Theorem 5.4. *For any $\rho > 0$, for any a-bounded game, $a > 0$, and general game with loss $\lambda_{\mathrm{LS},\rho}$ and for every $\eta \le 1$ we have $C_\eta = 1$. Moreover, for every $\eta \le 1$ and every $g \in GA(\eta)$, $C(g) = 1$. The only prediction attaining $C(g) = 1$ is the average (2.6), whereas before P is a probability distribution in Γ generating g. When $\eta > 1$, $C_\eta > 1$.*

The proof is the same as for Lemma 2.9, which is proven by [3].

We will apply the AA in the following fashion. We will use $\lambda_{\mathrm{LS},\rho}$ in the algorithm for calculating weights and working out the predictions γ_t. Then we will evaluate w.r.t. the original λ_{LS}.

Of course, the λ_{LS}-loss of the resulting algorithm will not satisfy Lemma 2.3. However, Lemma 2.3 will hold for the loss $\lambda_{\mathrm{LS},\rho}$. Note that the $\lambda_{\mathrm{LS},\rho}$-loss of a strategy is the same as the loss of the strategy with all predictions multiplied by ρ. This strategy suffers a larger loss and the term $\frac{1}{\eta}\ln\frac{1}{P_0(\theta)}$ will be small in comparison. Thus the algorithm will allow better differentiation of the weights, which *may* result in a better λ_{LS}-loss.

As discussed above, there is a danger that the strategy with predictions multiplied by ρ goes bankrupt. However, this will not propagate to the mixture.

Corollary 5.5. *The aggregating algorithm applied w.r.t. the loss $\lambda_{\mathrm{LS},\rho}$ and is conservative and a-conservative for every $a > 0$.*

Proof. We still average experts' predictions with some weights. While the weights may be different to AA, the argument of Theorem 5.3 stays. □

We may be affected by bankruptcy in the following way. If a ρ-multiple of an original expert goes bankrupt, its weights in the algorithm drop to zero. Its future predictions disappear from the mixture and the losses do not appear in the comparison. Still, we do not go bankrupt as per Corollary 5.5.

5.4.2 Downside Loss

The developments of this section are based on the following intuition.

From the practical perspective, the ability of a strategy not to lose money may be more important than the ability to earn money. Consider a strategy that earns little money, but does so very consistently and never loses much. This strategy can then be scaled up and earn more money.

Thus one often wants to minimise the drawdown of a trading strategy. There are various indicators quantifying it; they are discussed in the next section. One cannot apply AA directly to this problem because the notion of a drawdown is not local in time. Still, one can try and modify the loss function to penalize financial losses strongly.

Consider the downside loss function modifying the scaled Long-Short loss:

$$\lambda_{\mathrm{LS,down},\rho}(\omega,\gamma) = \max(-\ln(1+\rho\langle\omega,\gamma\rangle),0) = -\ln(1+\rho\min(\langle\omega,\gamma\rangle,0))$$

This function penalizes financial losses but does not reward gains.

The following statement can be made about its mixability properties.

Theorem 5.6. *For $\lambda_{\mathrm{LS,down},\rho}$ the average (2.6) attains $C = 1$ for every g, where as before P is a probability distribution in Γ generating g.*

Proof Consider a distribution P on Θ, and predictions $\gamma(\theta)$. One has

$$e^{-\lambda_{\mathrm{LS,down},\rho}(\omega,\gamma)} = 1+\rho\min(\langle\omega,\gamma\rangle,0)$$

and therefore it is sufficient to prove that

$$1+\rho\min(\langle\omega,\int_\Theta\gamma(\theta)P(d\theta)\rangle,0) \geq \int_\Theta(1+\rho\min(\langle\omega,\gamma(\theta)\rangle,0)P(d\theta)\ .$$

This follows from the concavity of $\min(x,0)$ in x and Jensen's inequality. ■

It is important to point out that this loss function is really special. There is $\gamma_0 = 0$ such that $0 = \lambda_{\mathrm{LS,down},\rho}(\omega, \gamma_0) \leq \lambda_{\mathrm{LS,down},\rho}(\omega, \gamma)$ for any ω and any other γ. Technically $C = 0$ and the problem of prediction with expert advice is trivial for this loss function: the learner only needs to predict 0.

Still applying the AA with $\lambda_{\mathrm{LS,down},\rho}$ and the substitution (2.6) in meaningful and the losses will satisfy Lemma 2.3.

5.4.3 Combined Loss

One can consider the combined loss function parameterised by scalings $\rho_1 \geq 0$ and $\rho_2 \geq 0$ and coefficients $u \geq 0$ and $v \geq 0$ (we assume that $\rho_1 + \rho_2 > 0$ and $u + v > 0$):

$$\begin{aligned}
\lambda(\omega, \gamma) &= -\ln(ue^{-\lambda_{\mathrm{LS},\rho}(\omega,\gamma)} + ve^{-\lambda_{\mathrm{LS,down},\rho}(\omega,\gamma)}) \\
&= -\ln((u + v) + u\rho_1\langle\omega, \gamma\rangle + v\rho_2 \min(\langle\omega, \gamma\rangle, 0)) \\
&= -\ln(u + v) - \ln\left(1 + \frac{u\rho_1}{u + v}\langle\omega, \gamma\rangle + \frac{v\rho_2}{u + v}\min(\langle\omega, \gamma\rangle, 0)\right)
\end{aligned}$$

Since this loss function is mixable (as we will see in a moment) the additive term $-\ln(u+v)$ makes no difference and can be ignored. One may think of the combined function as having only two parameters, $u\rho_1/(u+v)$ and $v\rho_2/(u+v)$, but speaking of four parameters may be more convenient.

Theorem 5.7. *For any $\rho_1, \rho_2, u, v \geq 0$ such that that $\rho_1 + \rho_2 > 0$ and $u+v > 0$, for any a-bounded game, $a > 0$, and general game with combined loss $\lambda_{\mathrm{LS},\rho}$ and for $\eta = 1$ we have $C_\eta = 1$. This is attained by the average substitution function (2.6), whereas before P is a probability distribution in Γ generating g.*

Corollary 5.8. *For any $\rho_1, \rho_2, u, v \geq 0$ such that $\rho_1 + \rho_2 > 0$ and $u + v > 0$, the aggregating algorithm with the combined loss function in conservative and a-conservative for every $a > 0$.*

5.5 Experimental Conclusions

Here we evaluate the performance of the proposed modifications to the long-short game and discuss the advantages of the various investment strategies. We will also be comparing the effect of sleeping experts using the same parameters.

5.6 Portfolio Performance Evaluation

To evaluate the performance we will be using well-established portfolio risk measures. As the games we are studying use returns over each trial to update the weight assigned to each expert investor, we will naturally use the ROI of each learner's portfolio as a measure of success. However, it is typical to not only evaluate a portfolio based on return alone but rather on the risk-reward of the portfolio. The Sharpe ratio [40] of a portfolio P is a measure of the amount of return an investor receives per unit of risk defined as:

$$\text{Sharpe}(P) = \frac{R_P - R_f}{\sigma(R_P)} \quad , \tag{5.1}$$

where R_p denotes the return to the portfolio p and R_f the return of the risk-free asset. This allows us to compare the risk of each of the learner's portfolios, using the standard deviation of the returns to the portfolio as a measure of volatility.

As we discuss in Section 5.4.2, one may be specifically interested in reducing the financial losses. The Sortino ratio [41] is a measure of return per unit of downside risk defined as

$$\text{Sortino}(p) = \frac{R_P - R_f}{\sigma(R_d)} \quad , \tag{5.2}$$

where R_d denotes the downside returns to the portfolio P being the returns recorded less than some target return. In the following, we will assume the return to the risk-free asset of 0% and a target return of 0%, for performance comparison.

We will also be looking at the *Drawdown* of investors' portfolios, which is a metric commonly used to measure the volatility of the ROI and refers to how much the ROI retraces from the highest ROI achieved, defined as:

$$\text{Drawdown}(T) = \min_{t \in [0,T]} (0, \text{ROI(T)} - \text{ROI}(t))$$

we ideally want to keep the drawdown as small as possible, a useful summary is to track the so-called 'maximum drawdown' which computes the maximum amount of ROI given away over time:

$$\text{Max Drawdown}(T) = \min_{t \in [0,T]} \text{Drawdown}(t)$$

5.7 Empirical Results

In Fig. 5.6 we can see the ROI of the learner's portfolio using various combined loss coefficients and setting $\rho_1 = \rho_2$ increasing from a value of 0-1000 (sampling every 100 intervals). The first thing to note is that for all games there is an improvement in the ROI up until some point where we begin to see a drop off in the results.

The learner following the investment strategy using a loss function of $\rho = 900, u = 1$ and $v = 0$ has an ROI of 3.83%, which is a 5.9 fold increase compared to the ROI of 0.65% resulting from the portfolio with a loss function with parameters of $\rho = 1, u = 1$ and $v = 0$. However, whilst a combined loss of u =1 and v = 0 produces the highest ROI and Sharpe ratio, its Sortino ratio is much lower when compared to games applying downside loss. This suggested that whilst games without downside loss can provide high ROI per unit of volatility we can expect to see much higher drawdowns in these portfolios when compared to those using downside loss. We can see an example of this represented in Fig. 5.14 comparing the portfolios of two investment strategies in this case with the implementation of specialist experts. We observe that while the combined loss coefficients of $u = 1$ and $v = 0$ produce a final ROI higher than the strategy using coefficients $u = 2$ and $v = 1$, the drawdowns are far greater seen in Fig. 5.15.

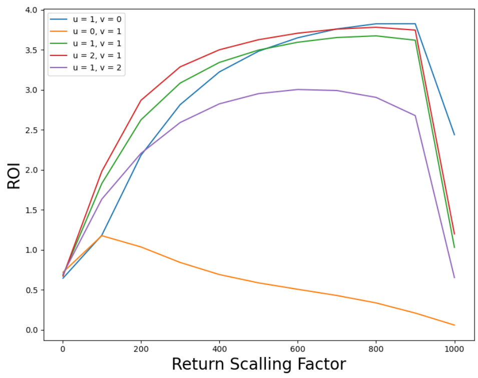

Fig. 5.6: ROI for increasing ρ

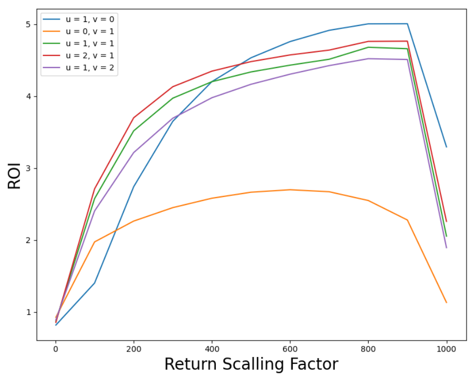

Fig. 5.7: ROI with Sleeping Experts

We see that the performance of the algorithms at first improves with the growth of ρ and then starts falling. According to Section 5.2 and 5.4 the improvement in performance is caused by better differentiation in the experts' weights and an increase in the significance of 2.4. This is offset by the growth of the number of bankrupt experts. In Fig. 5.8 we plot the number of clients with bankrupt trading strategies as the return scaling constant applied to the expert's loss increases. We see the number of bankruptcies steadily grow as $\rho \to \infty$ as we would expect as larger losses force client weights to zero. This may account for the sharp drop in portfolio performance as $\rho_1 = \rho_2$ reaches 1000. This is a

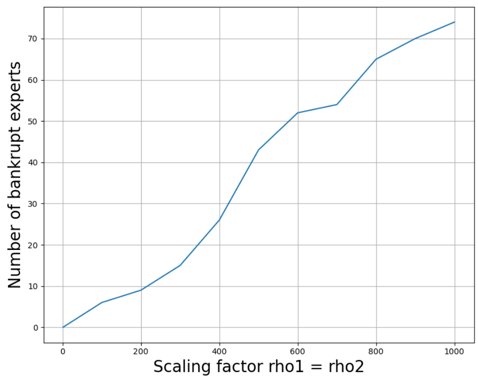

FIG. 5.8: Bankrupt clients as ρ increases

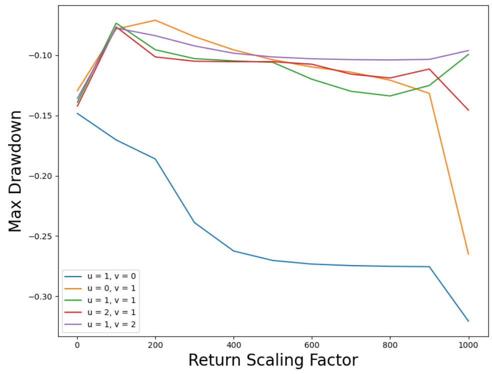

FIG. 5.9: Max Drawdown as ρ increases

clear representation of why we must increase the return scaling constant with caution as too large a value will guarantee a portfolio less than optimal performance.

In Fig. 5.9 we can see the maximum drawdown for games without the use of specialist experts or discounting. Firstly, we see clearly that in the case of $u = 1$ and $v = 0$ as we increase the return scaling constant we increase the maximum drawdown of the portfolio. This must be looked at alongside the Sharpe ratio that is increasing. This suggests that while we increase the return per unit of risk of the learner's portfolio we also increase the volatility of the portfolio. However, we can see for games where $v > 1$ the correlation does not hold. This implies that for investors looking to reduce drawdowns, they can achieve this using combined loss coefficients where $u \leq v$,

Comparing Fig. 5.6 with Fig. 5.7 we can see that while the pattern of the results largely remains the same, there is a significant increase in the ROI of the portfolios following the sleeping experts' decision-making method. Taking the parameters that result in the highest ROI for non-sleeping experts of $\rho = 900, u = 1$ and $v = 0$ and an ROI of 3.83%, the application of sleeping experts increases this to an ROI of 5% a 1.31 fold increase. We also see a similar increase in the Sharpe ratios of learners' portfolios comparing the same two portfolios we see an increase of 20% from 0.52% to 0.62%. However, whilst for the same game we saw an increase in the sortino ratio of 0.08%, overall sortino ratios decreased. The optimal sortino ratio without the use of sleeping experts was achieved using parameters of $\rho = 700, u = 2$ and $v = 1$ of 7.48. With the use of sleeping experts, the highest sortino ratio was given using $\rho = 300, u = 2$ and $v = 1$ of 6.09, a decrease of 18%.

Here we presented the results of discounting taking at $\alpha = 0.995$. Examining the ROI of the learner applying discounting whilst we see an increase correlated to the return scaling constant, the returns are lower and the growth is not as significant as other

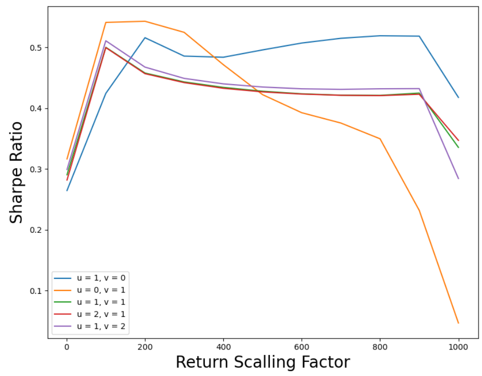

FIG. 5.10: Sharpe Ratio for increasing ρ

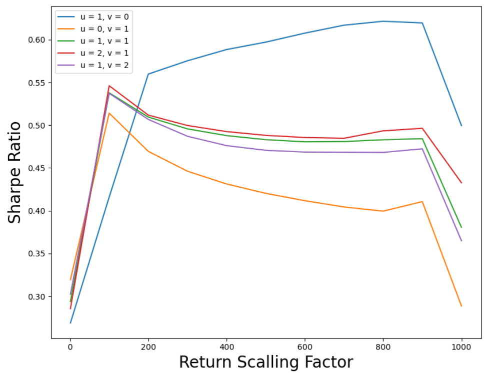

FIG. 5.11: Sharpe Ratio with Sleeping Experts

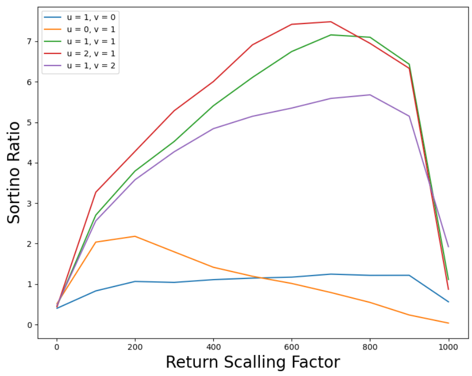

FIG. 5.12: Sortino Ratio for increasing ρ

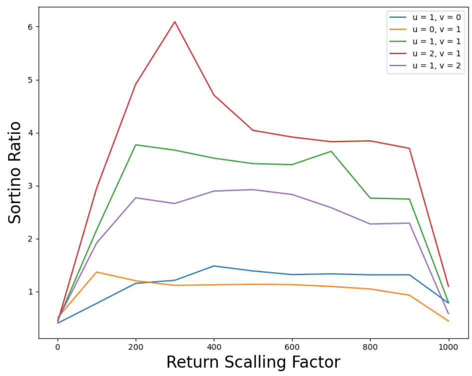

FIG. 5.13: Sortino Ratio with Sleeping Experts

implementations of the AA. However, we do see a notable change in the pattern of ROI taking the combined loss coefficients $u = 0$ and $v = 1$, with stronger performance than without the use of discounting. Moreover, we see a notable improvement in the Sortino ratio that one may assume to be a result of an increase in the effectiveness of downside loss. In Fig 5.19 we can see the ROI with the parameters $u = 1$, $v = 1$, $\rho = 1000$ and $\alpha = 0.995$, this is a noteworthy example as we can see whilst the ROI is less than previous games the learner is still actively investing in the market with minimal drawdowns. This suggested that if an investor prioritizes reducing drawdowns in their portfolio at the expense of long-term ROI the use of discounting may be an optimal strategy for a learner.

In Table 5.3 we can see the parameters and results of some of the AA implementations with optimal results. We see that for less risk-averse investors looking for high returns, the use of specialist experts can achieve this. Where in the case of the highest ROI of any AA it is in fact higher than the ROI of the best expert. However, for more

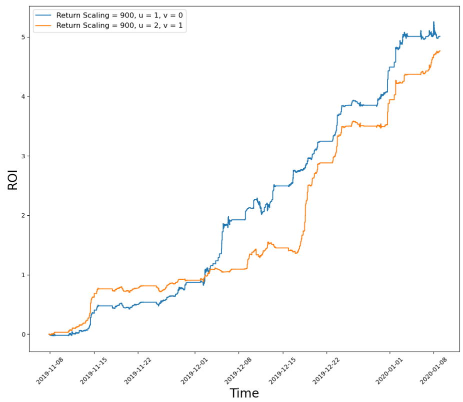

FIG. 5.14: ROI of Learner Portfolio

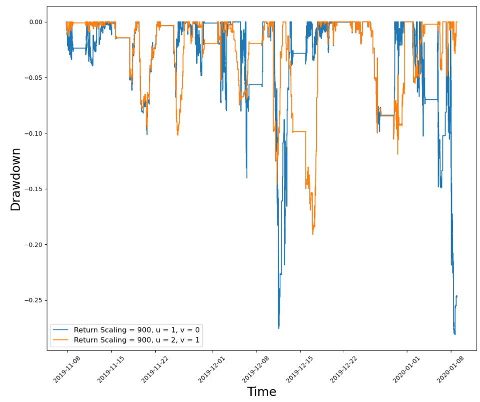

FIG. 5.15: Drawdown of Learner Portfolio

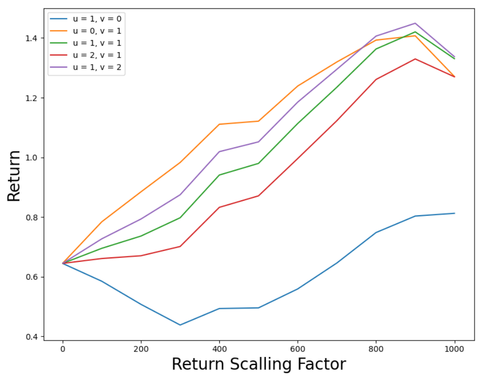

FIG. 5.16: ROI with Discounting

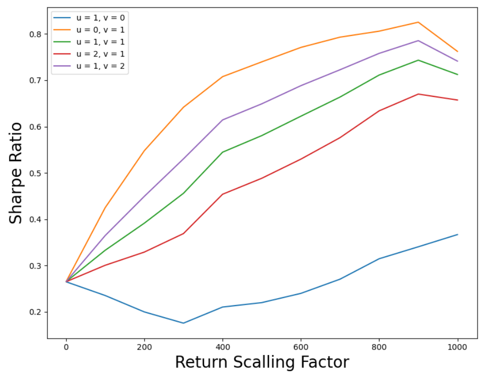

FIG. 5.17: Sharpe Ratio with Discounting

TABLE 5.3: Parameters and results of optimal solutions

	u	v	Return Scalling Factor	Specialist Experts	Alpha	ROI	Sharpe	Sortino	Max Drawdown
Equal Weights	-	-	-	-	-	0.64%	0.27	0.40	-0.15
Worst Expert	-	-	-	-	-	-2.34%	-0.23	-0.15	-3.04
Best Expert	-	-	-	-	-	4.52%	0.42	4.03	-0.22
Highest ROI	1	0	900	YES	1	**5%**	0.62	1.31	-0.28
Highest Sharpe and Best Max Drawdown	0	1	900	NO	0.995	1.4%	**0.83**	4.01	**-0.02**
Highest Sortino	1	1	1000	NO	0.995	1.33%	0.71	**11.28**	-0.035

risk-averse investors the use of downside loss and discounting can certainly reduce the overall volatility of the portfolio. We must also consider the use of specialist experts in combination with downside loss as this has shown to be a good compromise for investors.

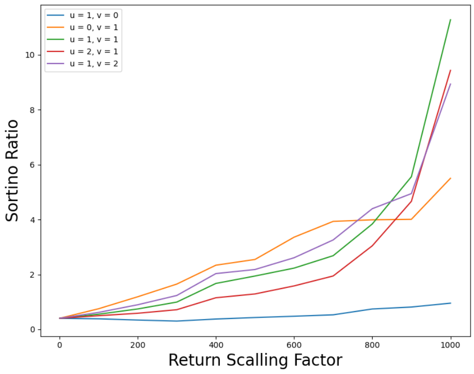

FIG. 5.18: Sortino Ratio with Discounting

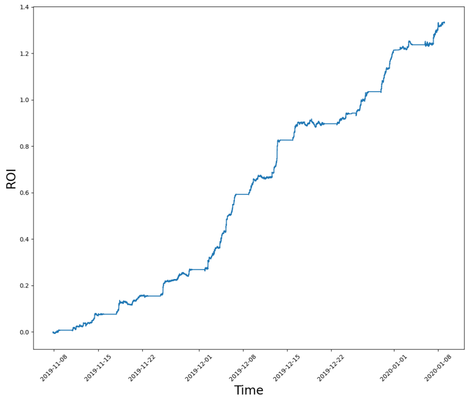

FIG. 5.19: ROI of Learner with Discounting

5.8 Conclusion

We have shown the practical limitations of the long-short game and have introduced modifications with clear performance benefits in our experimental results.

This study has presented a novel time series data set that describes client trades in the Foreign Exchange market ([39]) and has used this data to introduce a method of deriving expert predictions from client positions. Return scaling of the long-short game has been introduced, aimed to address the practical issue of insufficient discrimination of expert weights, and has been shown to provide significant performance improvements. We have also presented the downside long-short game with the motivation of reducing the downside risk of the investment strategy of the AA, which has proven to be effective using the Sortino ratio as a measure of downside risk. We have used combined loss functions to produce optimal performance of AA portfolios both maximising returns and reducing risk. Finally, we have shown the use of specialist experts can improve the overall performance of a learner's portfolio. Whereas, discounting can significantly reduce the volatility of an investment portfolio when used with downside loss at the expense of ROI.

Table 5.3 provides the summary of the trials of the study. Whilst we have outlined the empirical results, we should acknowledge the significance of the results and the impact they have on both the aggregating algorithm. Notably, we see that whilst the theoretical bound of the AA is met in empirical testing. We see this in the case of investment, where the number of experts is not insignificant. The bound provided by the AA is not sufficient to provide a meaningful limit on loss, in practice. The modifications introduced in this paper, have been shown to address this, and as table 5.3 shows. Increasing the ROI of the AA, which performed similarly to equal weights. To outperform the best experts'

ROI, with the new modifications introduced. However, we should not overlook the fact that the theoretical bound was of little use in practice. In further work, we would seek to explore this topic further, to find a more meaningful limit, for the long-short game.

In this chapter, we have shown that by adjusting the relationship between return scaling and downside loss, an investor can improve both their overall profitability as well as reduce their drawdown. These contributions made to the Long-Short game give investors the ability to vary their decision-making process of merging trading strategies, based on their risk/reward appetite.

Chapter 6

Online Hedging with the Weak Aggregating Algorithm

6.1 Introduction

Financial Market Makers (MMs) face a challenging online learning problem when it comes to managing a portfolio of risk. Here we define risk as it relates to market exposure - effectively represented by an MM's overall *position*. In simple terms, the position is the aggregation of all the *buy* and *sell* trades carried out by the MM's clients at a given point in time and evolves due to underlying asset price fluctuations and changes in client trading activity. Profit and Loss (also known as *PnL*) is a function of position and asset price movement. The position is described as being *long*, when the summation of client sell trades, exceeds those of client buy trades. A *short* position indicates a higher summation of client buy trades over sell trades. Roughly speaking, when a position is long and the price goes up, the PnL will increase, and vice versa when a position is short and the price goes down, the PnL will decrease. *Drawdown* is a metric commonly used to measure the volatility of the PnL and refers to how much the PnL retraces from the highest PnL achieved, defined as:

$$\text{Drawdown}(T) = \min_{t\in[0,T]}(0, \text{PnL(T)} - \text{PnL}(t))$$

MMs would ideally like to keep this drawdown time series as small as possible, a useful summary is to track the so-called 'maximum drawdown' which computes the maximum amount of PnL given away over time:

$$\text{Max Drawdown}(T) = \min_{t\in[0,T]} \text{Drawdown}(t)$$

An effective risk management strategy ensures that the position is kept within some predefined bounds and is as flat as possible. If a position is allowed to build up with no limit, becoming either too long or too short, then any unfavorable asset price movement will result in the MM incurring greater losses than if its position had been flatter or neutral. The position is actively maintained (or *hedged*) when the MM places buy or sell trades (*hedges*) when its position is respectively long or short. Strategies that indicate how much is hedged and when hedge trades are to be placed are known as *hedging strategies*. It follows that a hedging strategy that causes a reduction in position will also impact PnL, yet the nature of this impact will vary based on when and how much is being hedged. Hedging too much will more likely reduce drawdown but also reduce any profit. Hedge too little and risk market exposure and being at the mercy of disadvantageous price movements leading to large drawdowns. We discuss a commonly used hedging strategy known as the Cylinder Hedging Model in Section 6.2.

In this chapter, we focus on finding optimal hedging strategies by making use of on-line prediction with expert advice, namely the Weak Aggregating Algorithm or WAA [8, 9].

The problem of finding an optimal hedging strategy follows naturally from that of portfolio selection, which has itself seen extensive application of both the WAA and the Aggregating Algorithm (AA). The problem was first introduced by [2] and later developed by [3] to consider more realistic trading scenarios. In [42] the game was further developed to improve the application of the AA to find profitable trading strategies based on the past observations of a pool of investment strategies. The WAA has also been applied to the problem of finding a universal portfolio by [35] and further developed by [43] and [44].

As well as presenting a hedging framework for the WAA, we also introduce a method for applying discounted loss to the WAA. This allows for a learner to more effectively adapt to changes in market conditions in periods of high volatility. To test the efficacy of the approaches presented here we conduct experimental trials on real-world market and MM client data on three major currency pairs, EUR/USD, GBP/USD, and EUR/GBP.

6.2 Cylinder Hedging Model

Here we will describe the Cylinder Hedging Model, an algorithm that provides a hedging strategy based on the position of assets within a portfolio. The most fundamental cylinder model has two main parameters: (1) a pair of long and short limits (typically specified in US dollars) and (2) a desired hedge fraction specifying how much to hedge if one of these limits is breached. The limits define a so-called “cylinder” of risk, aiming to prevent the underlying position from growing too large, i.e., if the long limit is breached, the MM would place hedge sell trades to reduce the overall net position according to the hedge fraction (and vice versa short limit breaches). It should be noted, that in other areas of finance derivatives are a common choice of method of hedging. In the case of a market maker, a cylinder hedging model has notable benefits. Including the avoidance of risk exposure, taken from unforeseen movements in the derivative markets. And, the ability to manage position risk, with sacrificing excess PnL.

In our application of the cylinder model, the position is recorded at set intervals and a hedge fraction is placed for the duration of the interval. This is a natural hedging model for an MM where the portfolio being hedged is dictated by the flow of client trades. In Figures 6.1 and 6.2 we explain this by using a ‘toy’ example of a cylinder model with symmetrical long and short limits of 50 USD and -50 USD respectively (see red dotted lines), and a hedge fraction of 50% over 15 trial epochs. Figure 6.1 shows that at Trial 0, the client position (in blue) is at 100 USD. This breaches the long cylinder limit which in turn triggers the cylinder model to create an offsetting 50 USD hedge position as indicated by the orange line. This results in an overall net position (green line) of 50 USD. Throughout each trial this basic algorithm is repeated, resulting in the MM’s overall net position staying within the predefined ± 50 USD cylinder limits. Figure 6.2 shows the PnL values that result from trading this basic cylinder model over the 15 trials for the client (in blue), hedge (in orange), and overall net (in green). The raw client PnL ranges from -20 to $+55$ USD, yet after hedging, the volatility in the PnL is reduced between -10 and 45 USD.

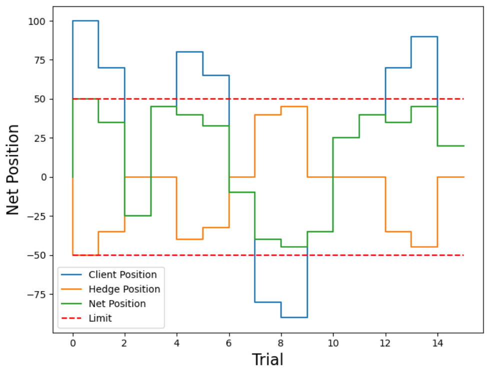

FIG. 6.1: Cylinder Model Position

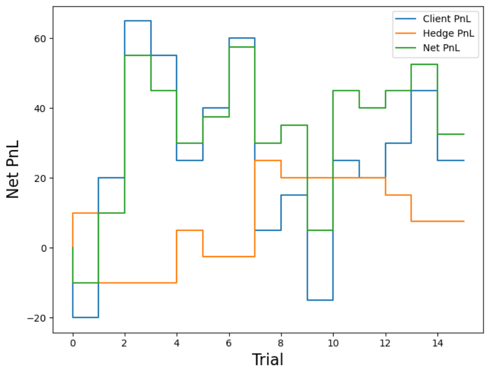

FIG. 6.2: Cylinder Model PnL

One natural extension often used to increase the profitability of the cylinder model is to skew the limits based on the view of the market direction. The motivation for this is illustrated in Table 6.1. If the price of the underlying asset is going up, the MM may want to skew the long cylinder limit upwards to prevent the model from triggering any hedges, thus permitting the MM to "ride the trend" in its existing long position and generate even further profit. By making the cylinder limits dynamic and asymmetric, MMs can hedge more selectively based on their net position and the overall market price movements.

Table 6.1: MM Optimal Hedge Decision

Market Condition	**MM Position Long (Client Position Short)**	**MM Position Short (Client Position Long)**
Price Increase	MM PnL Increase Skew long cylinder upwards → less likely to hedge If MM position breaches skewed long limit → place sell hedge	MM PnL Decrease If MM position breaches long cylinder limit → place sell hedge
Price Decrease	MM PnL Decrease If MM position breaches long cylinder limit → place buy hedge	MM PnL Increase Skew short cylinder downwards → less likely to hedge If MM position breaches skewed short limit → place buy hedge

In this study, we use dynamic cylinder models as our experts. To appropriately skew each model's cylinder limits, we will make use of the moving averages of the underlying asset prices as our market directional indicators. Moving averages are computed over various time windows and are commonly used in technical analysis. Thus when the current asset price is higher than its moving average counterpart, we forecast the market to be rising and hence the long cylinder limit can be positively skewed. Likewise, if the current asset price is lower than its moving average counterpart, then the market is likely to trend downward and the short cylinder limit can be skewed negatively. This dynamic adjustment of the cylinder limits can allow the model to both profit from and hedge against market corrections. Algorithm 7 provides us with the Pseudocode for the cylinder hedging model.

Figures 6.3 and 6.4 illustrate how the moving average is used to compute such dynamic skewed cylinder limits. Figures 6.5 and 6.6 show two ways of assessing the performance of the respective cylinder hedging models, tracking PnL and drawdowns. In these examples cylinder limits have been computed over the first 10,000 hourly epochs of a real-life EUR/USD trading dataset (described later in Section 4.1). Figure 6.3 shows the raw underlying price of the asset (EUR/USD) via the blue line. The moving average of the asset price computed over a window of 140 hours is shown by the red line. This line is smoother and lags behind as it is being computed. For each time epoch, we compute the market directional indicator, which is when the price line is above or below the slower-moving average line. This indicator skews the cylinder limits up or down respectively. The results of using the indicator in Figure 6.3 to skew the cylinder limits are shown in Figure 6.4, where a long limit of 10 million USD and a short limit of −20 million USD are both being skewed up and down by 70% and 80% respectively throughout. Figures 6.5 and 6.6 highlight the benefits of the cylinder hedging model in using these skewed limits. The overall client PnL (blue line) whilst making a profit does suffer from large drawdowns as shown in Figure 6.6, notably around trials 3000 and 8000. By observing the effect of hedging (orange line) it is clear that the cylinder hedging model has significantly reduced these losses, increasing the final net PnL (green line) by 51% from 538, 024 USD to 813, 484 USD.

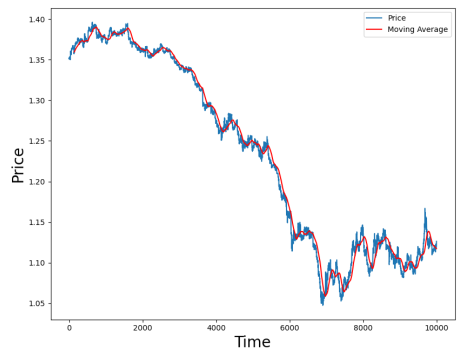

FIG. 6.3: Price of Underlying Asset

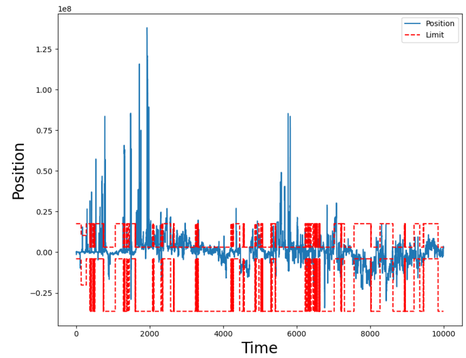

FIG. 6.4: Client Position with Skewed Limits

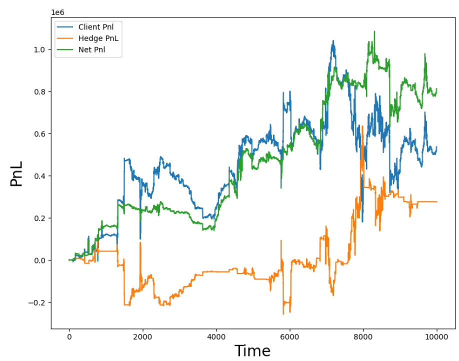

FIG. 6.5: Client, Hedge and Net PnL

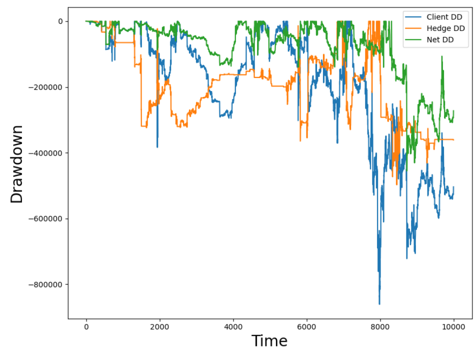

FIG. 6.6: Client, Hedge and Net Drawdown

We can achieve different risk-reward profiles by varying our main parameters to the cylinder model, namely the limits and the hedge fraction, but also by varying the window size of the moving average signal to best capture the main shifts in market price. The task of making accurate long-term price movement forecasts is extremely difficult [45] and whilst moving averages are lagging indicators, our results will later show that using these to dynamically adjust cylinder limits can give good results over different time epochs in this study's historical dataset.

We have made use of moving average market indicators to skew the cylinder limits in this study, yet there are significant challenges presented by the data flows involved. Both the market and underlying client flow evolve through time. Market trends can persist or change rapidly, such that too large a moving average window size could result in signals not skewing limits in time. Conversely, setting the window size too small could feasibly result in erratic oscillations failing to capitalize on any trend. In addition, the nature of client order flow is complex: new clients join and leave, and each client has different capitals to risk, differing trading time horizons, and risk appetites, all of which can result in client positions growing and shrinking accordingly. If the cylinder limits are set too large / too small this could result in hedges being triggered too little / too much. We can derive rough estimates of what appropriate cylinder limits and moving averages to use based on historic client and price data, but this offers a limited guarantee for future success. In this chapter we aim to use the Weak Aggregating Algorithm to combine the hedge predictions from a pool of dynamic skewed cylinder model experts, each with differing parameters (i.e. limits, hedge fractions, and moving average windows), to give an optimal hedging model that attempts to maximize PnL and minimize drawdowns.

Algorithm 7 Cylinder Hedging Model

Parameters: long/short Limit, Hedge fraction and Skew: $L_l, L_s, H_l, H_s, S_l, S_s$
Directional indicators Id_t, $t = 1, 2, \ldots$
for $t = 1, 2, \ldots$ **do**
  if $\text{Position}_t^C > L_l + (L_l \times S_l \times \mathrm{Id}_t)$ **then**
    $\text{Hedge Fraction}_t \leftarrow H_l$
  end if
  if $\text{Position}_t^C < L_s + (L_s \times S_s \times \mathrm{Id}_t)$ **then**
    $\text{Hedge Fraction}_t \leftarrow H_s$
  else
    $\text{Hedge Fraction}_t \leftarrow 0$
  end if
end for

6.3 Weak Aggregating Algorithm With Discounted Loss

In this section, we will introduce WAA for discounted loss. Discounting losses with time in the context of prediction with expert advice was first considered by [7]; see also a concise overview of discounting applied to the Aggregating Algorithm by Kalnishkan [46, Section 9].

The following argument for discounting can be given. The learner may want to align himself with the experts that have performed well lately, rather than over the whole course of history. Distant past may be irrelevant, especially in the context of economics and finance. Discounting offers a convenient framework for discarding the past.

There is also a purely numerical reason. The cumulative losses L_t^n calculated by the WAA may grow quite large with time and, correspondingly, the weights w_t^n very close to zero. This problem can be ameliorated by shifting the losses, but to some extent it is unavoidable if there are experts performing very differently.

Suppose that we are given coefficients $\alpha_1, \alpha_2, \ldots \in (0, 1]$. Let the cumulative discounted loss for a learner $\mathcal{L}$ be given by

$$\widetilde{\mathrm{Loss}}_T(\mathcal{L}) = \sum_{t=1}^{T} \lambda(\gamma_t) \left(\prod_{s=t}^{T-1} \alpha_s \right) = \alpha_{T-1} \widetilde{\mathrm{Loss}}_{T-1}(\mathcal{L}) + \lambda(\gamma_T) \ ;$$

the discounted loss $\widetilde{\mathrm{Loss}}_T(\mathcal{E}_n)$ of an expert $\mathcal{E}_n$ is defined in a similar way.

Algorithm 8 is identical to the Weak Aggregating Algorithm except that it uses discounted losses for L_t^n. We will refer to it as Weak Aggregating Algorithm with discounting (WAAd).

Algorithm 8 Weak Aggregating Algorithm with Discounting

Parameters: Initial distribution $q_1, q_2, \ldots, q_N$, $q_n \geq 0$ for $n = 1, 2, \ldots$ and $\sum_{n=1}^{N} = 1$.
Discounting factors $\alpha_1, \alpha_2, \ldots \in (0, 1]$.
Learning rates $\eta_t > 0$, $t = 1, 2, \ldots$
let $L_0^n = 0$, $n = 1, 2, \ldots, N$
for $t = 1, 2, \ldots$ **do**
  calculate weights $w_{t-1}^n = q_n e^{-\eta_t \alpha_{t-1} L_{t-1}^n}$, $n = 1, 2, \ldots, N$
  normalise the weights $p_{t-1}^n = w_{t-1}^n / \sum_{i=1}^{N} w_{t-1}^i$, $n = 1, 2, \ldots, N$
  read experts' predictions $\gamma_t^n \in \Gamma$, $n = 1, 2, \ldots, N$
  output $\gamma_t = \sum_{n=1}^{N} p_{t-1}^n \gamma_t^n$
  read experts losses ℓ_t^n, $n = 1, 2, \ldots, N$
  update $L_t^n = \alpha_{t-1} L_{t-1}^n + \ell_t^n$, $n = 1, 2, \ldots, N$
end for

Theorem 6.1. *Let the learning rates in WAAd be positive and non-decreasing, $\eta_{t-1} \geq \eta_t > 0$, $t = 1, 2, \ldots$ If all loss functions λ_t are convex and L satisfies (2.15) for $t = 1, 2, \ldots$ then*

$$\mathrm{Loss}_T(\mathcal{L}) \leq \mathrm{Loss}_T(\mathcal{E}_n) + \frac{\ln(1/q_n)}{\eta_T} + \frac{L^2}{8} \sum_{t=1}^{T} \eta_T \prod_{s=t}^{T} \alpha_s \qquad (6.1)$$

for all $T = 1, 2, \ldots$ and all experts $\mathcal{E}_n$, $n = 1, 2, \ldots, N$.

Proof [of Theorem 6.1]

Without loss of generality one can assume that $\ell_t^n \in [0, L]$ for all $n = 1, 2, \ldots, N$ and $t = 1, 2, \ldots$ Indeed, let us change λ_s to $\lambda_s + C_s$ for some constants $C_s \in \mathbb{R}$, $s = 1, 2, \ldots$ The discounted cumulative losses L_t^n then shift by some values independent of n. In the expressions for p_{t-1}^n, the shifts cancel out and the value of p_{t-1}^n will be unaffected. Therefore the predictions γ_t will not change and ℓ_t will change by C_t in line with ℓ_t^n. In inequality (6.1), which we need to prove, the values C_s cancel out.

Lemma A.1 by [9] implies that

$$\ell_t \leq \sum_{n=1}^N p_n^{t-1} \ell_t^n \leq -\frac{1}{\eta_t} \ln \sum_{n=1}^N p_n^{t-1} e^{-\eta_t \ell_t^n} + \frac{L^2}{8} \eta_t \ , \tag{6.2}$$

$t = 1, 2, \ldots, T$. Multiplying the inequality for time t by $\prod_{s=t}^{T-1} \alpha_s$ and adding them together yields

$$L_t \leq -\sum_{t=1}^T \frac{1}{\eta_t} \prod_{s=t}^{T-1} \alpha_s \ln \sum_{n=1}^N p_{t-1}^n e^{-\eta_t \ell_t^n} + \frac{L^2}{8} \sum_{t=1}^T \eta_t \prod_{s=t}^{T-1} \alpha_s \ . \tag{6.3}$$

Let us analyse the logarithm in this inequality. Substituting the expression for p_{t-1}^n yields

$$\begin{aligned}
-\frac{1}{\eta_t} \ln \sum_{n=1}^N p_{t-1}^n e^{-\eta_t \ell_t^n} &= -\frac{1}{\eta_t} \ln \sum_{n=1}^N \frac{q_n e^{-\eta_t \alpha_{t-1} L_{t-1}^n}}{\sum_{m=1}^N q_m e^{-\eta_t \alpha_{t-1} L_{t-1}^m}} e^{-\eta_t \ell_t^n} \\
&= -\frac{1}{\eta_t} \ln \sum_{n=1}^N q_n e^{-\eta_t (\alpha_{t-1} L_{t-1}^n + \ell_t^n)} + \frac{1}{\eta_t} \ln \sum_{n=1}^N q_n e^{-\eta_t \alpha_{t-1} L_{t-1}^n} \ .
\end{aligned}$$

In the first term, we get $\alpha_{t-1} L_{t-1}^n + \ell_t^n = L_t^n$. The second term can be upper bounded using Jensen's inequality

$$\begin{aligned}
\frac{1}{\eta_t} \ln \sum_{n=1}^N q_n e^{-\eta_t \alpha_{t-1} L_{t-1}^n} &= \frac{1}{\eta_t} \ln \sum_{n=1}^N q_n e^{-\eta_{t-1} (\eta_t \alpha_{t-1}/\eta_{t-1}) L_{t-1}^n} \\
&\leq \frac{1}{\eta_t} \ln \left(\sum_{n=1}^N q_n e^{-\eta_{t-1} L_{t-1}^n} \right)^{\eta_t \alpha_{t-1}/\eta_{t-1}} \\
&= \frac{\alpha_{t-1}}{\eta_{t-1}} \sum_{n=1}^N q_n e^{-\eta_{t-1} L_{t-1}^n}
\end{aligned}$$

as long as $\eta_t / \eta_{t-1} \leq 1$.

When we add over t in (6.3), the sum telescopes and we get (6.1). ■

Corollary 6.2. *Under the conditions of Theorem 2.11, if L satisfying (2.15) is known in advance and all discounting factors are equal and less than 1, $0 < \alpha_1 = \alpha_2 = \ldots = \alpha < 1$, one can take*

$$\eta_t = \eta = \frac{2\sqrt{2(1-\alpha)\ln N}}{L}$$

and ensure for equal weights $q_1 = q_2 = \ldots = q_N = 1/N$ the bound

$$\mathrm{Loss}_T(\mathcal{L}) \leq \mathrm{Loss}_T(\mathcal{E}_n) + L\sqrt{\frac{\ln N}{2(1-\alpha)}} \tag{6.4}$$

for all $T = 1, 2, \ldots$ and all experts $\mathcal{E}_n$, $n = 1, 2, \ldots, N$.

Remark 6.3. The discounted loss would not normally grow with time as regular loss usually does. If $0 \leq \lambda_t(\gamma) \leq L$ and the discounting factors are constant, then

$$\widetilde{\mathrm{Loss}}_T(\mathcal{L}) \leq \frac{L}{1-\alpha} \quad .$$

Our bound will then be meaningful only if the extra term in (6.4) is much less than this trivial bound, i.e.,

$$L\sqrt{\frac{\ln N}{2(1-\alpha)}} \ll \frac{L}{1-\alpha} \quad .$$

This happens if α is close to 1 and the difference $1-\alpha$ is small compared to $2/\ln N$.

6.4 Weak Aggregating Algorithm for Hedging

In this section, we discuss how hedging can be considered within the framework of prediction with expert advice. The learner aims to find the optimal hedge fraction for an MM hedging the risk associated with client positions. The pool of experts here is a set of cylinder models with different input parameters producing different hedge fractions.

Here we consider the case of the client position in a single asset, and a hedging decision is represented by $\gamma \in [-1, 0]$, where $\gamma_t = -1$ implies hedging out the entire client position and $\gamma_t = 0$ corresponds to a decision not to hedge over trial t at all.

It is natural to define loss in terms of the MM's PnL resulting from facilitating client orders. Note, as PnL represents the MM's gain, we need to take its inverse when defining the loss. We can therefore take the loss at time t to be $\lambda(\gamma_t) = -\mathrm{PnL}_t\gamma_t$. The cumulative loss over T trials is the negation of PnL over these trials, i.e., the amount MM has lost.

Since $\mathrm{PnL} \in \mathbb{R}$ is linear in the hedge fraction, the loss functions λ_t are convex. To establish a bound on loss, we need to see whether we can define limits on the value of the MM's client PnL. As discussed, client PnL is a function of the client's position and the price of the underlying asset. Let us first consider the case when the MM's client position is long. Here PnL is bounded by the value of the net position because in the worst possible scenario, the value of the underlying can fall to zero and the loss in PnL is equal to the value of the net position. In the case where the MM takes a short position, there is no simple bound on the possible loss. However, based on the interval of predictions and the historical market volatility we can make assumptions on the maximum rise in the value of the underlying asset and therefore the bound on loss.

Following [42], we introduce downside and combined losses. We aim to increase the penalty for losses and reduce the reward for gains. In the context of optimization of investment portfolios, this has been shown to reduce the drawdown of a portfolio; combined loss has proven particularly effective. Here we will take a similar approach considering the loss function with the coefficients $u \geq 0$ and $v \geq 0$:

$$\lambda(\gamma) = -\left(\frac{u}{u+v}\mathrm{PnL}\gamma + \frac{v}{u+v}\min(\mathrm{PnL}\gamma, 0)\right)$$

This has the aim of allowing the learner to focus on finding hedging strategies that minimize drawdown to provide smoother returns. Since $-\min(-x, 0)$ is convex in x, the loss function remains convex.

6.5 Experiments

To empirically evaluate the effectiveness of the WAAd at finding optimal parameters for the cylinder hedging model, we have conducted experiments with a real-world Foreign Exchange (FX) dataset drawn from the data flows of trading three currency pairs from February 2014 to June 2017.

6.5.1 Data Set

The data set we will be using is real-world currency exchange data, based on the trading behavior of individuals opening positions with a FX MM. The dataset focuses on the net position of the following three currency pairs - EUR/USD, GBP/USD, and EUR/GBP, over 41 months (Feb 2014 - June 2017) represented in hourly epochs. Figures 6.7 through 6.15 show the price of each of the currency pairs over this period in addition to the client position and resulting PnL. Note that client position refers to the MM's position resulting from client orders, and similarly, client PnL is the MM's PnL resulting from the net client position. To feed expert predictions to the WAAd, the data was partitioned into regularised time intervals using a technique known as DAPRA, as outlined in [32]. A copy of the data and WAAd implementation can be found at [47].

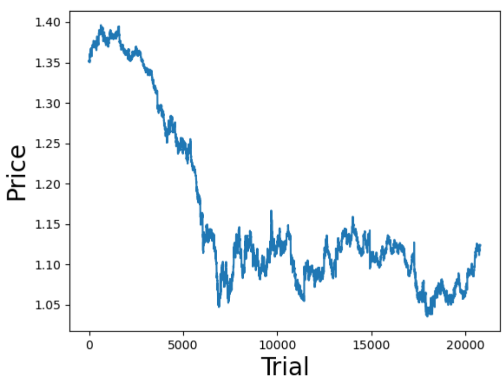

Fig. 6.7: EUR/USD Price

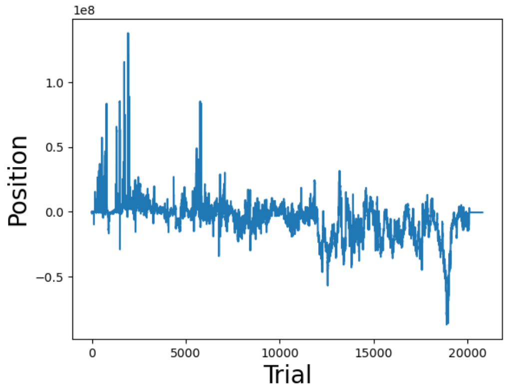

Fig. 6.8: EUR/USD Position

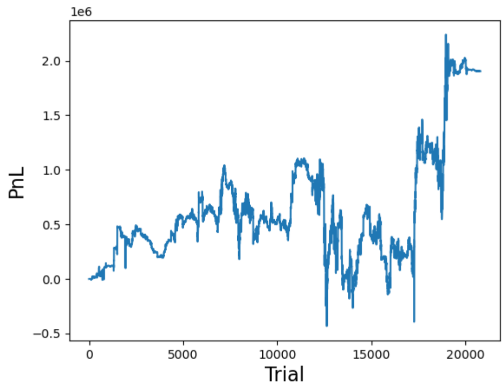

Fig. 6.9: EUR/USD PnL

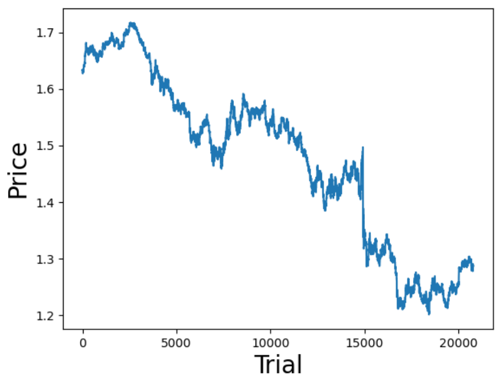

Fig. 6.10: GBP/USD Price

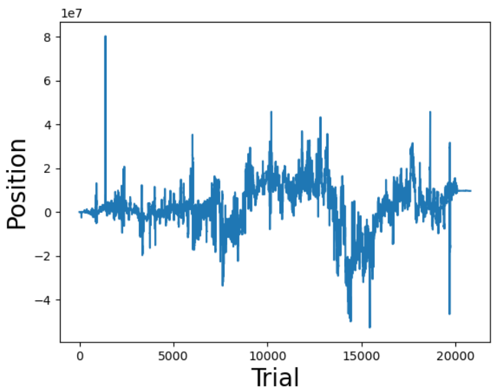

Fig. 6.11: GBP/USD Position

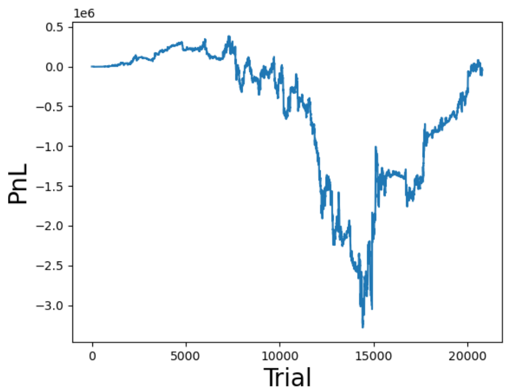

Fig. 6.12: GBP/USD PnL

For each currency pair, 100 unique cylinder model parameter combinations were chosen, resulting in the Net PnLs shown in Figures 6.16 to 6.18. These are the expert predictions used by the WAAd to generate a learner prediction. It is important to note that for

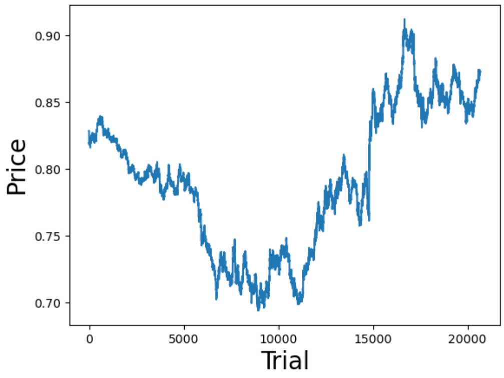

FIG. 6.13: EUR/GBP Price

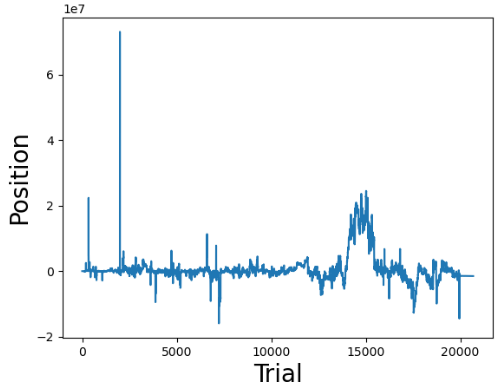

FIG. 6.14: EUR/GBP Position

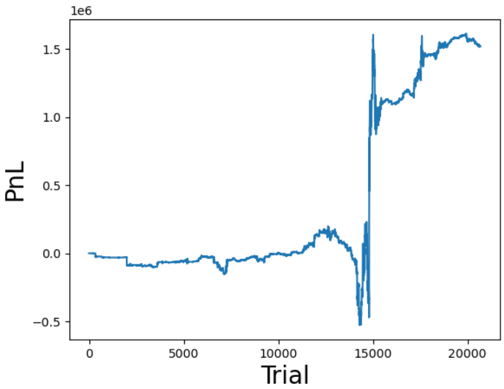

FIG. 6.15: EUR/GBP PnL

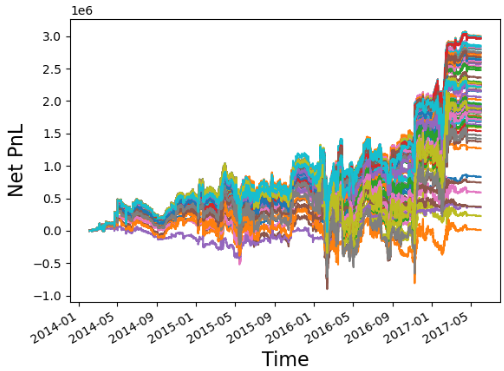

FIG. 6.16: EUR/USD Net PnL

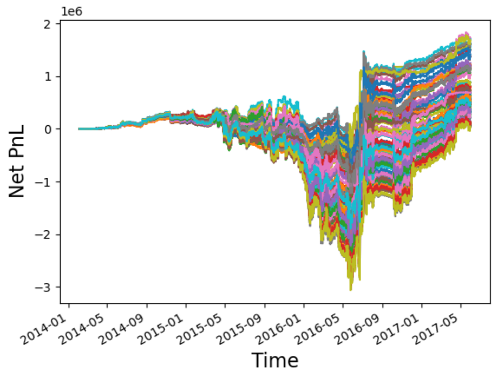

FIG. 6.17: GBP/USD Net PnL

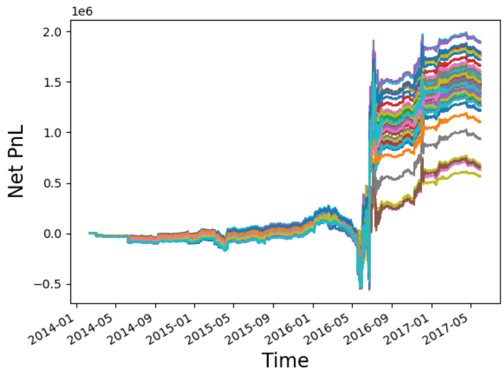

FIG. 6.18: EUR/GBP Net PnL

each currency pair, there are experts that both improve and worsen the MM's overall PnL and drawdown.

6.5.2 Numerical Analysis

We will now analyze the performance of the WAAd on each of the currency pairs. To do this we use the Calmar Ratio, which is a well-defined risk metric used to evaluate the performance of a portfolio, first introduced in [48] and defined as follows:

$$\text{Calmar}(T) = \frac{\text{Return}[0, T]}{|\text{Max Drawdown}(T)|} \tag{6.5}$$

For each currency pair, a scatter plot of Net PnL against maximum drawdown is used to illustrate how the Calmar ratio of the un-hedged and expert portfolios compares to that of the WAAds. Tables 6.2 to 6.4 provide the maximum drawdown, total PnL, and the mean and standard deviation of each epoch's PnL values. Each table shows the results of three useful benchmarks to evaluate the performance of each WAAd result: (1) un-hedged (just client results), (2) the best and (3) the worst of the experts used in the study. We wish to maximize the Calmar, maximum drawdown, PnL, and mean PnL measures, whilst minimizing the standard deviation of the PnL.

Figure 6.19 shows a plot of maximum drawdown against the final PnL of un-hedged, expert, and learner portfolios, all for EUR/USD. We have used different shapes to represent each of the combined loss parameters for the WAAd, and different colors to represent the various discount factors. Models with the highest Calmar ratio and therefore optimal, will by definition be located in the top right of the plot. We can see the worst hedging strategy produced by the WAAd is with a combined loss of $u = 1$ and $v = 0$ and no discounting. Following this strategy results in a Calmar ratio of 0.58 and a decrease in PnL of 55% only reducing max drawdown by 6% when compared to the un-hedged portfolio.

If we refer to Table 6.2 we can see the WAAd learner with the highest Calmar ratio of 3.63, taking combined loss coefficients of $u = 0$ and $v = 1$ applying a discount factor of 10%. This results in a 9.7% increase in PnL and a 62.6% decrease in drawdown when compared to that of the un-hedged portfolio.

As we observed with EUR/USD, all trials of the WAAd hedging GBP/USD reduce the MM's drawdown and in this case also increase PnL significantly. The model with the highest Calmar ratio of 2.86, is that with combined loss coefficients of $u = 0$ and $v = 1$ and a discount factor of 2.5%. When comparing this to the MM's un-hedged portfolio, we observe that the PnL is increased from a loss of $42,908$ USD to a profit of $1,123,464$ USD. This is possible due to the significant drawdowns in the client PnL over the initial 1500 trials and is reflected in the reduction of the maximum drawdown from $3,663,188$ USD to $392,370$ USD.

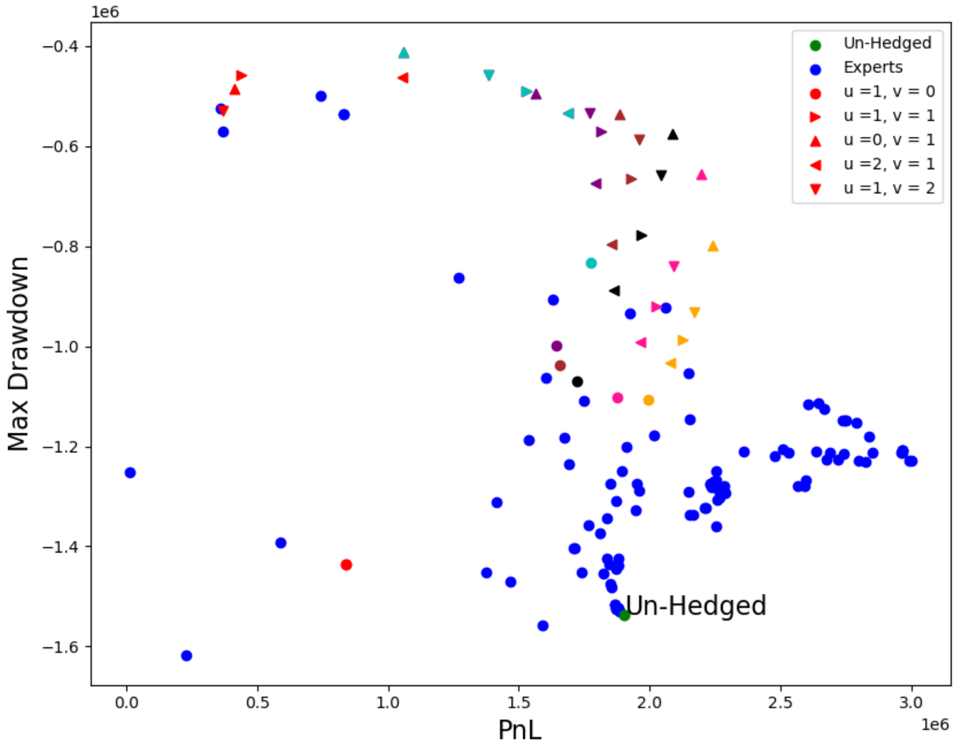

FIG. 6.19: EUR/USD Expert and WAA PnL against Max Drawdown. **Discount Key** Red: 0%, Cyan: 2.5%, Purple: 5%, Black: 7.5%, Pink: 10%, Orange: 20%

However, it is important to note that without applying discounting to experts' loss and taking combined loss coefficients of $u = 1$ and $v = 1$ or $u = 2$ and $v = 1$, we can find solutions with higher PnL at the expense of increased maximum drawdown.

EUR/GBP is an interesting experiment - as is shown in Figure 6.15 the majority of the PnL is a result of a surge in the underlying currency pair value. Figure 6.21 shows that the un-hedged portfolio is relatively high, yet has suffered from significant maximum drawdown. In this case, combined loss coefficients of $u = 1$ and $v = 1$ provide the optimal Calmar ratio, with a decrease in maximum drawdown value of 42% and an increase in PnL of 6% compared to the un-hedged portfolio.

It may not appear immediately obvious as to why the application of discounted loss on EUR/USD is more effective at improving the Calmar ratio than its use on GBP/USD and EUR/GBP. However, some intuition can be gleaned if one compares the results of the WAAd to the data outlined in section 6.5.1. When examining the accumulation of the MM's PnL in EUR/USD there is no clear period of profit or loss. One possible explanation for this may be found by referring to the price of the currency pair over the experiment. Excluding the initial quarter of the trial shows there is no dominant trend

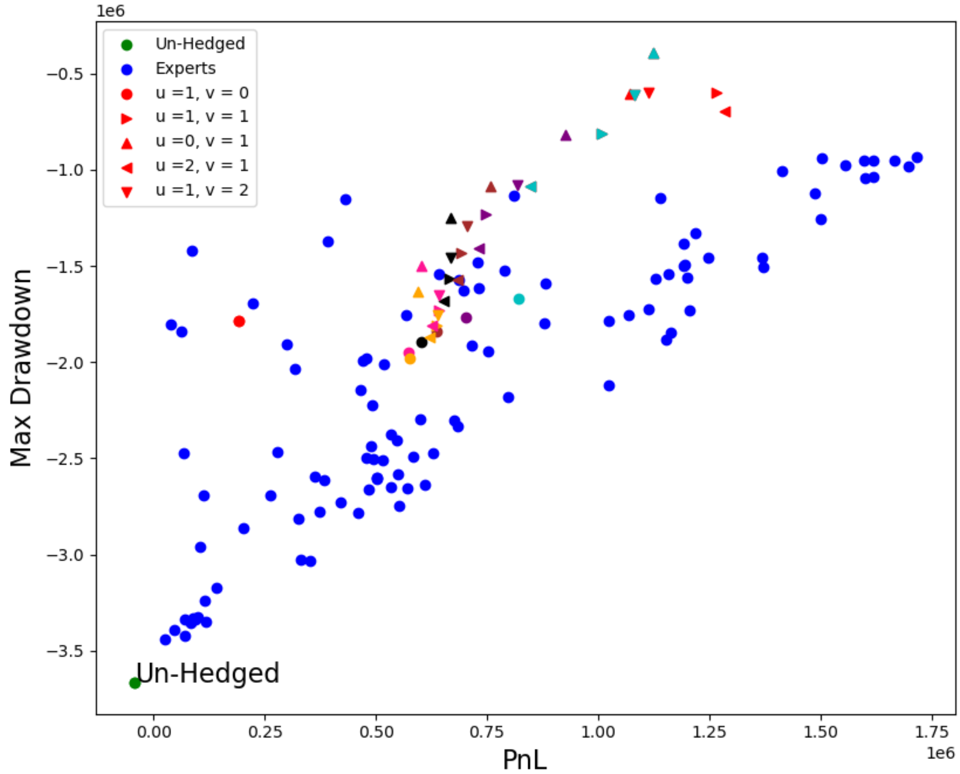

FIG. 6.20: GBP/USD Expert and WAA PnL against Max Drawdown. **Discount Key** Red:0%, Cyan: 2.5%, Purple: 5%, Black: 7.5%:, Pink: 10%, Orange: 20%

in the price. The result of this is that there is no clear optimal hedging strategy for any sustained period throughout the experiment. If we compare this to the client PnL of the GBP/USD currency pair, we see there is a clear downward trend over the initial three quarters, followed by an upward trend for the remainder of the experiment with the MM netting a small loss. Naturally, if there is a clear trend in the direction of PnL we can expect that a pool of models will consistently outperform the majority over this period. Therefore, the benefit of using discounted loss is diminished in such scenarios.

6.6 Conclusions

Building on previous work, we have shown that the Weak Aggregating Algorithm (WAA) can be used to combine the predictions from a pool of cylinder hedging models to improve key performance metrics - namely the overall profit (PnL) - whilst simultaneously not compromising on the smoothness of returns by minimizing drawdowns. In this study, we have further introduced a method for applying discounted loss to the WAA

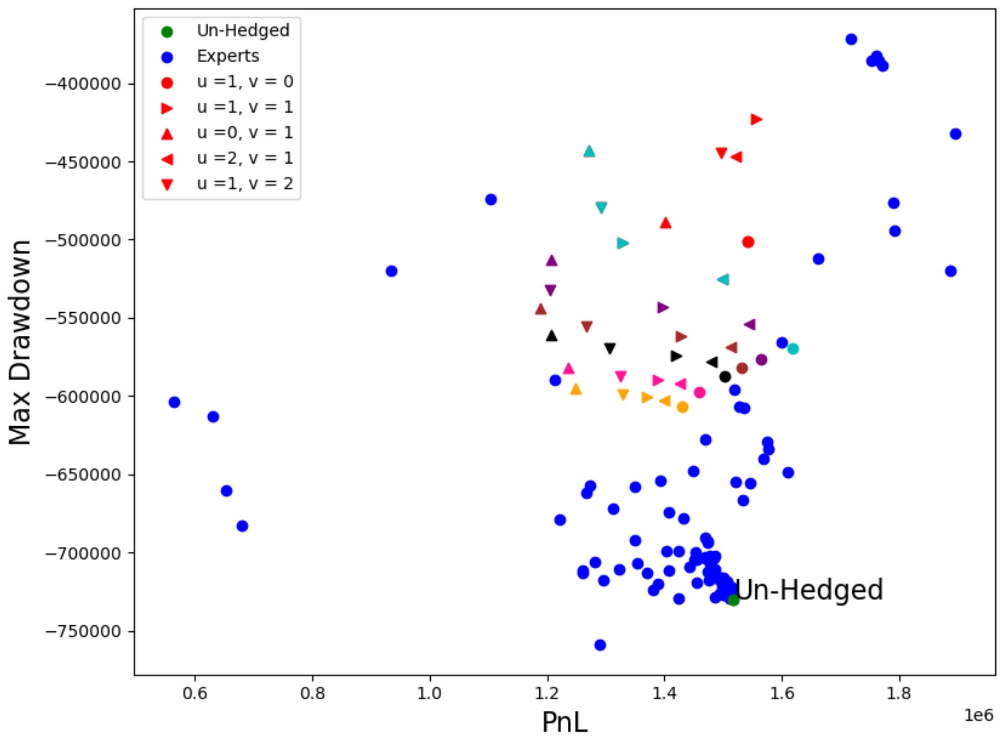

FIG. 6.21: EUR/GBP Expert and WAA PnL against Max Drawdown.
Discount Key Red:0%, Cyan: 2.5%, Purple: 5%, Black: 7.5%, Pink: 10%, Orange: 20%

(WAAd). Using a real-world Foreign Exchange (FX) trading dataset we demonstrate the empirical efficacy of our approach for three major currency pairs, varying the parameters of the WAAd namely the use of combined loss and discount factor. Whilst our analysis was carried out for individual FX currency pairs, a possible extension to this work would allow combinations of pairs into a *single currency risk*, allowing one to focus on individual EUR, USD, and GBP positions that build up across the triad of currency pairs (i.e. EUR/USD, GBP/USD, EUR/GBP). Due to the nature of these triangular relationships, there is an enforced correlation between the price movements. These correlations strengthen and weaken in time and are typically fed in to compute a Value At Risk (VaR) measure, introduced in 1994 by J. P. Morgan [49].

We have observed that discount loss can be varied in the WAAd to achieve an improved Calmar ratio and its effectiveness is a product of market conditions and client behavior. It is worth remembering that the underlying data is a complex combination of two major factors: (1) the trading activity of the MM clients, and (2) the price movements of the symbols being traded. The client net position is the aggregate risk across a *dynamic* portfolio of trader activity: new traders join, some traders leave over time, and each client presents with differing risk appetites, time horizons, and capital that they can trade with. Market conditions like volatility of the underlying prices and correlation between different symbols also vary profoundly through time due to various macroeconomic factors. In future work, we hope to investigate the use of dynamic discounting based on the observations in fundamental changes of trading and market activity to capitalize when "regime change" is detected. By varying the discount factor, the model will be able to react to macroeconomic events, leading to a more practical method of hedging.

Table 6.2: EUR/USD WAA Table of results

Hedging Models u	v	Discount %	Calmar	Max Drawdown (10ˆ6)	PnL (10ˆ6)	Mean PnL	PnL Standard Deviation
Un-hedged			1.22	-1.50	1.90	92	15,988
Best Model			3.63	-0.58	2.10	100	8,448
Worst Model			0.58	-1.40	0.84	40	10,731
1	0	0	0.58	-1.40	0.84	40	10,731
1	0	2.5	2.13	-0.83	1.80	85	10,741
1	0	5	1.64	-1.00	1.60	79	10,856
1	0	7.5	1.6	-1.00	1.70	80	10,897
1	0	10	1.61	-1.10	1.70	83	10,918
1	0	15	1.7	-1.10	1.90	90	10,958
1	0	20	1.8	-1.10	2.00	96	10,997
1	1	0	0.96	-0.46	0.44	21	4,065
1	1	2.5	3.12	-0.49	1.50	74	7,337
1	1	5	3.17	-0.57	1.80	87	8,658
1	1	7.5	2.9	-0.67	1.90	93	9,319
1	1	10	2.53	-0.78	2.00	95	9,698
1	1	15	2.2	-0.92	2.00	97	10,130
1	1	20	2.16	-0.99	2.10	102	10,381
0	1	0	0.85	-0.49	0.41	20	4,029
0	1	2.5	2.57	-0.41	1.10	51	5,646
0	1	5	3.15	-0.50	1.60	75	6,831
0	1	7.5	3.51	-0.54	1.90	91	7,783
0	1	10	3.63	-0.58	2.10	100	8,448
0	1	15	3.36	-0.65	2.20	106	9,248
0	1	20	2.81	-0.80	2.20	108	9,710
2	1	0	2.27	-0.46	1.10	51	4,701
2	1	2.5	3.16	-0.53	1.70	81	8,396
2	1	5	2.65	-0.68	1.80	86	9,404
2	1	7.5	2.33	-0.80	1.90	89	9,856
2	1	10	2.1	-0.89	1.90	90	10,113
2	1	15	1.98	-0.99	2.00	94	10,411
2	1	20	2.01	-1.00	2.10	100	10,591
1	2	0	0.7	-0.53	0.37	18	4,032
1	2	2.5	3.02	-0.46	1.40	66	6,558
1	2	5	3.32	-0.53	1.80	85	7,965
1	2	7.5	3.34	-0.59	2.00	94	8,785
1	2	10	3.1	-0.66	2.00	98	9,279
1	2	15	2.49	-0.84	2.10	100	9,841
1	2	20	2.33	-0.93	2.20	104	10,164

Table 6.3: GBP/USD WAA Table of results

Hedging Models			**Calmar**	**Max Drawdown (10ˆ6)**	**PnL (10ˆ6)**	**Mean PnL**	**PnL Standard Deviation**
u	**v**	**Discount %**					
Un-hedged			-0.01	-3.70	-0.04	-2	12,366
Best Model			2.86	-0.39	1.10	54	4,354
Worst Model			0.11	-1.80	0.19	9	6,922
1	0	0	0.11	-1.80	0.19	9	6,922
1	0	2.5	0.49	-1.70	0.82	39	7,945
1	0	5	0.4	-1.80	0.70	34	8,025
1	0	7.5	0.35	-1.80	0.64	31	8,042
1	0	10	0.32	-1.90	0.60	29	8,050
1	0	15	0.29	-2.00	0.57	28	8,058
1	0	20	0.29	-2.00	0.58	28	8,064
1	1	0	2.12	-0.60	1.30	61	5,531
1	1	2.5	1.24	-0.81	1.00	48	5,749
1	1	5	0.61	-1.20	0.75	36	6,543
1	1	7.5	0.48	-1.40	0.69	33	6,963
1	1	10	0.42	-1.60	0.67	32	7,204
1	1	15	0.37	-1.70	0.64	31	7,467
1	1	20	0.35	-1.80	0.64	31	7,610
0	1	0	1.78	-0.60	1.10	51	5,846
0	1	2.5	2.86	-0.39	1.10	54	4,354
0	1	5	1.13	-0.82	0.93	44	5,075
0	1	7.5	0.7	-1.10	0.76	36	5,670
0	1	10	0.54	-1.30	0.67	32	6,102
0	1	15	0.4	-1.50	0.60	29	6,650
0	1	20	0.36	-1.60	0.59	29	6,981
2	1	0	1.84	-0.70	1.30	62	5,618
2	1	2.5	0.78	-1.10	0.85	41	6,474
2	1	5	0.52	-1.40	0.73	35	7,099
2	1	7.5	0.43	-1.60	0.68	33	7,373
2	1	10	0.39	-1.70	0.65	31	7,522
2	1	15	0.35	-1.80	0.62	30	7,686
2	1	20	0.33	-1.90	0.62	30	7,776
1	2	0	1.85	-0.60	1.10	53	5,611
1	2	2.5	1.77	-0.61	1.10	52	5,118
1	2	5	0.76	-1.10	0.82	39	5,983
1	2	7.5	0.55	-1.30	0.70	34	6,513
1	2	10	0.46	-1.50	0.67	32	6,842
1	2	15	0.39	-1.70	0.64	31	7,215
1	2	20	0.36	-1.80	0.64	31	7,420

Table 6.4: EUR/GBP WAA Table of results

Hedging Models			**Calmar**	**Max Drawdown (10ˆ5)**	**PnL (10ˆ6)**	**Mean PnL**	**PnL Standard Deviation**
u	**v**	**Discount %**					
Un-hedged			2.11	-7.30	1.50	76	9,194
Best Model			3.68	-4.20	1.60	75	7,273
Worst Model			2.1	-6.00	1.20	60	7,017
1	0	0	3.08	-5.00	1.50	75	7,434
1	0	2.5	2.84	-5.70	1.60	78	7,923
1	0	5	2.71	-5.80	1.60	76	7,985
1	0	7.5	2.63	-5.80	1.50	74	8,009
1	0	10	2.56	-5.90	1.50	73	8,023
1	0	15	2.44	-6.00	1.50	71	8,039
1	0	20	2.36	-6.10	1.40	69	8,049
1	1	0	3.68	-4.20	1.60	75	7,273
1	1	2.5	2.65	-5.00	1.30	64	6,417
1	1	5	2.57	-5.40	1.40	68	6,967
1	1	7.5	2.54	-5.60	1.40	69	7,343
1	1	10	2.48	-5.70	1.40	69	7,520
1	1	15	2.35	-5.90	1.40	67	7,689
1	1	20	2.28	-6.00	1.40	66	7,776
0	1	0	2.87	-4.90	1.40	68	6,882
0	1	2.5	2.87	-4.40	1.30	61	5,700
0	1	5	2.35	-5.10	1.20	58	5,789
0	1	7.5	2.18	-5.40	1.20	57	6,080
0	1	10	2.15	-5.60	1.20	58	6,323
0	1	15	2.12	-5.80	1.20	60	6,719
0	1	20	2.1	-6.00	1.20	60	7,017
2	1	0	3.41	-4.50	1.50	74	7,354
2	1	2.5	2.86	-5.30	1.50	73	7,116
2	1	5	2.79	-5.50	1.50	75	7,585
2	1	7.5	2.66	-5.70	1.50	73	7,719
2	1	10	2.56	-5.80	1.50	72	7,787
2	1	15	2.41	-5.90	1.40	69	7,861
2	1	20	2.32	-6.00	1.40	68	7,903
1	2	0	3.37	-4.40	1.50	72	7,133
1	2	2.5	2.7	-4.80	1.30	63	6,126
1	2	5	2.26	-5.30	1.20	58	6,255
1	2	7.5	2.28	-5.60	1.30	61	6,696
1	2	10	2.3	-5.70	1.30	63	7,024
1	2	15	2.26	-5.90	1.30	64	7,388
1	2	20	2.22	-6.00	1.30	64	7,568

Chapter 7

Conclusion

In this thesis, we have considered the use of prediction with expert advice algorithms for investment and hedging in the Foreign Exchange market. In Chapter 2 we provided a review of the existing framework for prediction with expert advice. Including the Aggregating Algorithm and the Weak Aggregating Algorithm. This included the introduction of the covers game and the Long-Short game, later developed in this thesis. In Chapter 3 we gave an overview of the FX market, specifically the brokerage industry, how a profit is made trading currency pairs, and the problem of hedging client positions.

In chapter 4 we introduced DAPRA, an ETL algorithm that merges multiple irregularly sampled time series datasets, into one. We discussed the rationale for DAPRA, walked through its design, and introduced the theoretical foundation of any DAPRA application. We reported empirical evidence that demonstrates the practical relevance of DAPRA by its application with large and complex time series datasets from two distinct domains, financial and travel. This included showing how it can be used to tackle the issue of deriving a broker's PnL and position from client trading data, merged with market data, vital to the application of client data in the following chapters.

A key area of further work is the expansion of DAPRA into other areas of the financial markets. For example, this study has focused on using DAPRA to process data from a foreign exchange broker. However, there is a wealth of data that would adhere to the framework of the algorithm. Such as combining trading data with markets data, to calculate financial models in real-time using derived fields. Moreover, the study of the optimal interval size is also an area of key concern and one that deserves further research.

In chapter 5 we saw how the performance of the Long-Short game, is poor concerning financial performance metrics. To improve the performance of the game, in finding optimal investment decisions, we introduce modifications to the game's loss function. We introduce return scaling and downside loss and show how combined loss can be used to balance to risk-return ratio depending on an investor's risk preference. In section 5.7 we see the results of the modifications, with the application of the Aggregating Algorithm on real-world FX data. Showing a significant increase in performance compared to the general Long-Short game.

Then in Chapter 6, we explore the application of prediction with expert advice to find an FX broker's optimal hedging strategy. We first introduce the cylinder hedging model, which allows an FX broker to manage client risk. We show how the Weak Aggregating Algorithm can be used to find optimal hedging strategies and apply discounting loss to the WAA. In section 6.5, we apply the WAA and WAAd to combine a pool of hedging strategies, showing a reduction in portfolio drawdowns, whilst retaining PnL.

Bibliography

[1] Vladimir Vovk. Aggregating strategies. In M. Fulk and John Case, editors, *Proceedings of the Third Annual Workshop on Computational Learning Theory*, pages 371–383. Morgan Kaufmann, 1990.

[2] Thomas M Cover and Erik Ordentlich. Universal portfolios with side information. *IEEE Transactions on Information Theory*, 42(2):348–363, 1996.

[3] Volodya Vovk and Chris Watkins. Universal portfolio selection. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 12–23. ACM, 1998.

[4] Vladimir Vovk. A game of prediction with expert advice. *Journal of Computer and System Sciences*, 56(2):153–173, 1998.

[5] Yuri Kalnishkan. Prediction with expert advice for a finite number of experts: A practical introduction. *Pattern Recognition*, 126, June 2022. ISSN 0031-3203. doi: 10.1016/j.patcog.2022.108557.

[6] A. Chernov and V. Vovk. Prediction with expert evaluators' advice. In *Algorithmic Learning Theory, ALT 2009, Proceedings*, volume 5809 of *LNCS*, pages 8–22. Springer, 2009.

[7] A. Chernov and F. Zhdanov. Prediction with expert advice under discounted loss. In *Proccedings of ALT 2010*, volume LNAI 6331, pages 255–269. Springer, 2010. See also arXiv:1005.1918 [cs.LG].

[8] Yuri Kalnishkan and Michael V Vyugin. The weak aggregating algorithm and weak mixability. *Journal of Computer and System Sciences*, 74(8):1228–1244, 2008.

[9] N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games.* Cambridge University Press, 2006.

[10] A. Chernov. On Theorem 2.3 in "Prediction, learning, and games" by Cesa-Bianchi and Lugosi. *CoRR*, abs/1011.5668, 2010.

[11] Eitel JM Lauría and Carlos A Greco. Olap for financial analysis and planning-a proof of concept. In *ICSOFT (1)*, pages 347–355, 2010.

[12] Nils H Rasmussen, Paul S Goldy, and Per O Solli. Financial business intelligence: trends, technology, software selection, and implementation. 2002.

[13] Javier Arroyo and Carlos Maté. Introducing interval time series: Accuracy measures. *COMPSTAT 2006, proceedings in computational statistics*, pages 1139–1146, 2006.

[14] Paulo MM Rodrigues, Nazarii Salish, et al. Modeling and forecasting interval time series with threshold models: An application to s&p500 index returns. *Banco de Portugal, Economics and Research Department, Working paper series*, (w201128), 2011.

[15] textitDAPRA, github repository. 11 2019. URL `https://github.com/Wisniewskiw/DAPRA`.

[16] D. Lindsay. *FXClientTrades*, london, uk, kaggle. 8 2019. URL `https://www.kaggle.com/davidlindsay1979/toptradingclientdata/kernels`.

[17] C. Whong. "foiling nyc's taxi trip data". 2014. URL `https://chriswhong.com/open-data/`.

[18] D. Beniaguev. *Historical Hourly Weather Data 2012-2017*, kaggle. 12 2017. URL `https://www.kaggle.com/selfishgene/historical-hourly-weather-data/version/2`.

[19] James F Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

[20] Leticia I Gómez, Silvia A Gómez, and Alejandro A Vaisman. A generic data model and query language for spatiotemporal olap cube analysis. In *Proceedings of the 15th International Conference on Extending Database Technology*, pages 300–311, 2012.

[21] T Schneider. Analyzing 1.1 billion nyc taxi and uber trips, with a vengeance. retrieved july, 2017, 2015.

[22] Forex Factory. *Forex Calendar*, fair economy inc. 2019. URL `ttps://www.forexfactory.com/calendar.php`.

[23] Luca Leonardi and Salvatore Orlando. Alessandra raffaetà, alessandro roncato, claudio silvestri, gennady andrienko & natalia andrienko.

[24] Guy P Nason, Ben Powell, Duncan Elliott, and Paul A Smith. Should we sample a time series more frequently?: decision support via multirate spectrum estimation. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, pages 353–407, 2017.

[25] G. Xuedong and C. Hailan. The method of time granularity determination on time seriesbased on structural similarity measure algorithm. *Proceedings of the International Symposium on the Analytic Hierarchy Process: the 14th ISAHP Conference*, 2016.

[26] Gao Xuedong and Chen Hailan. The method of time granularity determination on time seriesbased on structural similarity measure algorithm. In *Proceedings of the International Symposium on the Analytic Hierarchy Process: the 14th ISAHP Conference*, 2016.

[27] George Athanasopoulos, Rob J Hyndman, Nikolaos Kourentzes, and Fotios Petropoulos. Forecasting with temporal hierarchies. *European Journal of Operational Research*, 262(1):60–74, 2017.

[28] Xian Wu, Baoxu Shi, Yuxiao Dong, Chao Huang, Louis Faust, and Nitesh V Chawla. Restful: Resolution-aware forecasting of behavioral time series data. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1073–1082, 2018.

[29] L eon Bottou. Online learning and stochastic approximations. *On-linelearning in neural networks*, 17(9):142, 1998.

[30] Harry Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952.

[31] Nick Littlestone, Manfred K Warmuth, et al. *The weighted majority algorithm.* University of California, Santa Cruz, Computer Research Laboratory, 1989.

[32] Najim Al-baghdadi, Wojciech Wisniewski, David Lindsay, Sian Lindsay, Yuri Kalnishkan, and Chris Watkins. Structuring time series data to gain insight into agent behaviour. In *Proceedings of the 3rd International Workshop on Big Data for Financial News and Data*. IEEE, 10 2019.

[33] David P. Helmbold, Robert E. Schapire, Yoram Singer, and Manfred K. Warmuth. On-line portfolio selection using multiplicative updates. *Mathematical Finance*, 8 (4):325–347, 1998.

[34] László Györfi, Frederic Udina, and Harro Walk. Experiments on universal portfolio selection usingdata from real markets. 2008. URL `https://api.semanticscholar.org/CorpusID:3133273`.

[35] Yong Zhang and Xingyu Yang. Online portfolio selection strategy based on combining experts' advice. *Computational Economics*, 50(1):141–159, 2017.

[36] Xingyu Yang, Hong Lin, Yong Zhang, et al. Boosting exponential gradient strategy for online portfolio selection: An aggregating experts' advice method. *Computational Economics*, 55(1):231–251, 2020.

[37] V V'yugin and V Trunov. Universal algorithmic trading. *Journal of Investment Strategies*, 2(1):13, 2012.

[38] VV V'yugin, IA Stel'makh, and VG Trunov. Adaptive algorithm of tracking the best experts trajectory. *Journal of Communications Technology and Electronics*, 62 (12):1434–1447, 2017.

[39] Najim Al-baghdadi and David Lindsay. Aggregating algorithm long-short game dataset. 5 2020. URL `https://www.kaggle.com/najimal/aggregating-algorithm-longshort-game-dataset`.

[40] William F Sharpe. Mutual fund performance. *The Journal of business*, 39(1): 119–138, 1966.

[41] Frank A Sortino and Lee N Price. Performance measurement in a downside risk framework. *the Journal of Investing*, 3(3):59–64, 1994.

[42] Najim Al-Baghdadi, David Lindsay, Yuri Kalnishkan, and Sian Lindsay. Practical investment with the long-short game. In Alexander Gammerman, Vladimir Vovk, Zhiyuan Luo, Evgueni Smirnov, and Giovanni Cherubin, editors, *Proceedings of the Ninth Symposium on Conformal and Probabilistic Prediction and Applications*, volume 128 of *Proceedings of Machine Learning Research*, pages 209–228. PMLR, 09–11 Sep 2020. URL `https://proceedings.mlr.press/v128/al-baghdadi20a.html`.

[43] Jin'an He and Xingyu Yang. Universal portfolio selection strategy by aggregating online expert advice. *Optimization and Engineering*, pages 1–25, 2020.

[44] Xingyu Yang, Jin'an He, Jiayi Xian, Hong Lin, and Yong Zhang. Aggregating expert advice strategy for online portfolio selection with side information. *Soft Computing*, 24(3):2067–2081, 2020.

[45] Eugene F Fama. Efficient capital markets: A review of theory and empirical work. *The journal of Finance*, 25(2):383–417, 1970.

[46] Y. Kalnishkan. Prediction with expert advice for a finite number of experts: A practical introduction. *Pattern Recognition*, page 108557, 2022.

[47] Najim Al-baghdadi. Waa cylinder hedging model git repository. 5 2022. URL `https://github.com/Najim-Al/PhD_AA`.

[48] Terry W Young. Calmar ratio: A smoother tool. *Futures*, 20(1):40, 1991.

[49] T. Guldimann. *Riskmetrics TM*, 1995.

[50] Yuri Kalnishkan. *The Aggregating Algorithm and Predictive Complexity.* PhD thesis, Department of Computer Science, Royal Holloway, University of London, Egham, 2002.

[51] Giles Jewitt. *FX derivatives trader school.* John Wiley & Sons, 2015.

[52] Alexey Chernov and Fedor Zhdanov. Prediction with expert advice under discounted loss. In Marcus Hutter, Frank Stephan, Vladimir Vovk, and Thomas Zeugmann, editors, *Algorithmic Learning Theory*, pages 255–269, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-16108-7.

[53] Dmitry Adamskiy, Anthony Bellotti, Raisa Dzhamtyrova, and Yuri Kalnishkan. Aggregating algorithm for prediction of packs. *Machine Learning*, 108(8-9):1231–1260, 2019.

[54] Thomas M. Cover. Universal portfolios. *Mathematical Finance*, 1(1):1–29, 1991. doi: https://doi.org/10.1111/j.1467-9965.1991.tb00002.x. URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9965.1991.tb00002.x`.