Aggregating Algorithm

Andrew Barraclough

Submitted for the Degree of Master of Science in

Machine Learning

Department of Computer Science
Royal Holloway University of London
Egham, Surrey TW20 0EX, UK

July 17, 2024

Declaration

This report has been prepared on the basis of my own work. Where other published and unpublished source materials have been used, these have been acknowledged.

Word Count:

Student Name: Andrew Barraclough

Date of Submission: 29 August 2024

Signature:

Abstract

Your abstract goes here.

Contents

Acknowledgements

While the contents of this report are on the basis of my own work, none of this would have been possible without the patience and mentorship of my supervisor Dr. Yuri Kalnishkan to whom I am extremely grateful. It was your advice, clear explanations, and expertise that made this project what it is now and something that I am incredibly proud of. I would also like to express my gratitude to the group of friends who made this academic year possible, namely Cougar Tasker, Einstein Ebereonwu, Hayden Amarr, Mohammadreza Yazdian, Niraj Jain, and Ray Mahbub, without whom I would have struggled to maintain my discipline and motivation.

1 Introduction *(1,000)* (728)

1.1 Project Scope and Objectives (205)

The aim of this project is to implement methods of Prediction with Expert Advice, such as the Aggregating Algorithm, and to evaluate their performance in different scenarios, specifically targeting real-world applications.

As an introduction to the concepts explored in the chapters to come, these methods allow for the pooling of different prediction algorithms (known as 'experts') with the goal of improving prediction accuracy—allowing the final prediction to be nearly as accurate as the best-performing expert.

This project will encompass several key areas, including:

- **Explaining the Theory of Prediction with Expert Advice.** To effectively experiment with different methods of Prediction with Expert Advice, the underlying theory must first be understood by conducting a review of the relevant literature.

- **Implementing the Aggregating Algorithm.** This project will primarily investigate the Aggregating Algorithm introduced by Vovk (see [1], [2]).

- **Handling Specialist Experts.** Introduced by Freund [3], *Specialist Experts* may refrain from making predictions at certain points, meaning that the Aggregating Algorithm has to be modified slightly [4].

- **Applying Prediction with Expert Advice to Real-World Data.** The methods described in this report will be applied to real-world datasets in order to evaluate their practicality outside of theoretical models, including an investigation into the perception of randomness by utilising specialist experts.

1.2 Motivation and Interest in the Subject Area (261)

The motivation for selecting a project in this subject area is rooted in both my personal and professional interests, as well as the discussions I had with my now-supervisor, Dr. Yuri Kalnishkan, before finalising my selection.

During this academic year, the module that most piqued my interest was CS5200 – On-line Machine Learning because I was interested in the techniques that allowed machine learning models to gradually improve over time as more data became available to them without the need to retrain the model on the entire newly-updated dataset; something that had not been covered previously by other modules. Due to the module's small size and frequent absentees, I was able to gain a deeper understanding of the module, in large part due to Dr. Kalnishkan's willingness to explain portions of the syllabus in extreme detail. Alongside the lectures, I felt like I was strongly suited to the contents of the module because it has strong ties to the field of statistics – another area that I thoroughly enjoyed throughout my education.

Regarding my professional aspirations, I am set to begin my career later this year and I am of the firm belief that the work that I have done in this subject area is highly relevant, not only to the job I am to start in September, but also for my career plan due to its relevance across a variety of industries – including finance, energy, and insurance.

Ultimately, the combination of all of these factors led me to pursue a project investigating on-line prediction, and prediction with expert advice.

1.3 Structure of the Dissertation (262)

The dissertation is split into distinct chapters, each dedicated to exploring a specific aspect of the work. The following outline guides the reader through the report by providing a brief overview of the contents of each chapter.

Chapters 2 through 5 contain a literature review organised to explain the concepts that the practical portion of the dissertation aims to explore. Chapter 2 defines the problem of On-line Prediction, outlining the scenarios that it applies to, and the protocols that such problems follow. Additionally, it explores how on-line learning differs from traditional batch learning and defines concepts that will be critical to understanding the following sections. Chapter 3 introduces the problem of Prediction with Expert Advice, explaining its significance and applications in the real world, as well as exploring some algorithms that are used to solve such problems – including their theoretical bounds. Chapter 4 introduces the Aggregating Algorithm that this report is centred around, exploring how it differs from other methods of Prediction with Expert Advice. Chapter 5 focuses on Specialist Experts, defining what they are and how the base Aggregating Algorithm must be modified to accommodate them.

Chapter 6 contains the practical portion of the dissertation, explaining how the research problem was handled based on the concepts explored in the literature review, the findings from conducting the requirements analysis and design processes, and the results found when comparing an individual's idea of "random" to that of a random number generator.

Finally, Chapter 7 contains a conclusion which discusses the findings of the investigation as well as a self-evaluation of the project.

2 Literature Review

2.1 Introduction

Purpose: Overview of the goals of the literature review.
Scope: Outline of the topics covered and their relevance to the dissertation.

2.2 Perceived Randomness

2.2.1 Introduction to Perceived Randomness

Definition: Explanation of what perceived randomness is.
Importance: Discussion on why perceived randomness is significant in various fields.

Perceived Randomness refers to the human tendency to asses sequences of events or data as random or non-random based on subjective criteria. This perception is often influenced by cognitive biases and heuristics, which can lead to misjudgements. Humans typically look for patterns or irregularities in sequences and may perceive a truly random sequence as non-random if it doesn't match their expectation of what randomness should look like.

Binary sequences are sequences composed of two distinct symbols, 0 and 1. When evaluating the randomness of such sequences, people often expect certain characteristics, such as:

- A roughly equal number of 0s and 1s.
- No long runs of identical symbols.
- A lack of obvious patterns or regularities.

However, truly random binary sequences can occasionally contain runs of identical symbols or other patterns that might appear non-random to an observer. People's perception of randomness in binary sequences is therefore influenced by these expectations and may not always align with actual randomness.

2.2.2 Randomness in Binary Sequences

Human vs. Algorithmic Generation
Human Perception: How humans perceive randomness.
Algorithmic Methods: Comparison of human and algorithmic sequence generation.

When dealing with generated binary sequences, such as those produced by random number generators or algorithms, people apply the same subjective criteria to judge randomness. Even though these sequences are designed to be random, they may sometimes exhibit patterns or anomalies that seem non-random. This perception is shaped by the same cognitive biases that affect judgements of naturally occurring sequences. Evaluating the randomness of generated binary sequences often involves statistical tests and analysis to ensure they meet the mathematical criteria for randomness, which can differ from human perceptions.

Methods for Generating Sequences
Techniques: Different methods for generating binary sequences.
Comparative Analysis: Evaluation of these methods in terms of perceived randomness.

2.3 Prediction with Expert Advice

2.3.1 Introduction to On-line Prediction

Within the areas of Machine Learning and Statistics, there lies the problem of accurately "predicting future events based on past observations" [5] known as *on-line prediction*. This problem refers to methods where a model makes predictions sequentially and updates its parameters in real-time as new data points become available. There is a particular class of algorithm that is designed to tackle this, with one of the most notable being the "Strong" Aggregating Algorithm proposed by Volodymyr Vovk [1] which forms the basis of this study. The adjective "Strong" is emphasised with inverted commas to help distinguish the algorithm from the "Weak" Aggregating Algorithm proposed by Yuri Kalnishkan and Michael Vyugin [6] that will be touched upon but not explored in detail in this dissertation.

Given that the foundations of this dissertation lie firmly in this subject area, this section aims to lay a comprehensive foundation, exploring the key concepts and frameworks that will set the stage for the discussions in Chapter **TODO**.

On-line Prediction, Batch Learning and Timeseries Analysis

Herein the first distinction between on-line prediction and the traditional batch learning framework. With batch learning, a whole training set of labelled examples (x_i, y_i) is given to the learner at once in order to train a model. In contrast, on-line learning involves gradually feeding the learner information over time, requiring the model to continuously adapt to the new data it is given while requiring the learner to take actions on the basis of the information it already possesses instead of waiting for a complete picture. [4] This forced adaptability ensures that the predictions outputted by the algorithm remain accurate based on the information that the model deems as relevant as it gains additional knowledge, making these models particularly valuable in applications that require immediate responses and fluidity such as financial market analysis and weather forecasting.

Another distinction that needs to be made is between on-line prediction and timeseries analysis as, while these are both ways of handling sequential data in machine learning and statistics, they are unique. On-line learning is based on processing data points sequentially and updating predictive models in real-time whereas timeseries analysis is based on modelling and forecasting data that is collected over successive time intervals. The prior approach does not impose any strict assumptions about the underlying data-generating process, even going so far as to not assume the existence of such a process [7], while the latter assumes a structured approach where observations are dependent on previous observations. These are typically modelled using stochastic processes such as *autoregressive integrated moving average (ARIMA)* or *state-space* models [8]. The majority of the literature on Online Prediction takes a similar stance that no assumptions can be made about the sequence of outcomes that are observed. Because of this, the analyses are done over the worst-case and may be better in reality [5].

Notation

In on-line prediction, we consider a scenario where the elements of a sequence, known as ***outcomes***, ω_t occur at discrete times $\omega_1, \omega_2, \dots$ which we assume to be drawn from a known ***outcome space*** Ω. In this problem, a learner is tasked with making ***predictions*** γ_t about these *outcomes* one at a time before they occur. Similarly, we assume that the learner's predictions are drawn from a known ***prediction space*** Γ which may or may not be the same as the *outcome space* Ω.

Once the learner has made their *prediction*, the true *outcome* is then revealed and the quality of the learner's prediction is assessed by a ***loss function*** $\lambda(\gamma_t, \omega_t)$. This function measures the discrepancy between the *prediction* and *outcome* or, more generally, quantifies the effect of when the *prediction* γ_t is confronted with the *outcome* ω_t [9] by mapping the input space $\Gamma \times \Omega$ to a subset of the real-number line $\mathbb{R}$, typically $[0, +\infty)$ [10].

Across several time steps T, the learner will suffer multiple losses which can be referred to as their cumulative loss up to time T. Their performance is measured by this cumulative loss, so their natural objective is to suffer as low a cumulative loss as they can.

Protocol 1 On-line Prediction Framework

```
FOR $t = 1, 2, \dots$
    learner $L$ outputs $\gamma_t \in \Gamma$
    nature outputs $\omega_t \in \Omega$
    learner $L$ suffers loss $\lambda(\gamma_t, \omega_t)$
END FOR
```

2.3.1.1 Games and Mixability

The combination of a *prediction space*, *outcome space*, and *loss function* can be referred to with a triple $< \Gamma, \Omega, \lambda >$, known as a ***Game*** G. *TODO: Explain mixability and touch on (Kalnishkan & Vyugin, 2008)* [6]

2.3.2 Prediction with Expert Advice

Framework: Description of the prediction with expert advice framework.
Mechanisms: Detailed explanation of how this framework operates.

Protocol 2 Prediction with Expert Advice Framework

```
FOR $t = 1, 2, \dots$
    experts $E_1, \dots, E_N$ output predictions$\gamma_t^1, \dots, \gamma_t^N \in \Gamma$
    learner $L$ outputs $\gamma_t \in \Gamma$
    nature outputs $\omega_t \in \Omega$
    experts $E_1, \dots, E_N$ suffer losses $\lambda(\gamma_t^1, \omega_t), \dots, \lambda(\gamma_t^N, \omega_t)$
    learner $L$ suffers loss $\lambda(\gamma_t, \omega_t)$
END FOR
```

2.3.3 Aggregating Algorithm (AA)

Algorithm Description: Introduction to the Aggregating Algorithm.
Functionality: How the Aggregating Algorithm works in practice.

Algorithm 1 Aggregating Algorithm (AA)

```
initialise weights $w_0^i = q_i, i = 1, 2, \dots, N$
FOR $t = 1, 2, \dots$
    read the experts' predictions $\gamma_t^i, i = 1, 2, \dots, N$
    normalise the experts' weights $p_{t-1}^i = w_{t-1}^i / \sum_{j=1}^N w_{t-1}^j$
    output $\gamma_t \in \Gamma$ that satisfies the inequality for all $\omega \in \Omega$:
        $\lambda(\gamma_t, \omega) \leq -\frac{C}{\eta} \ln \sum_{i=1}^N p_{t-1}^i e^{-\eta\lambda(\gamma_t^i, \omega)}$
    observe the outcome $\omega_t$
    update the experts' weights $w_t^i = w_{t-1}^i e^{-\eta\lambda(\gamma_t^i, \omega_t)}, i = 1, 2, \dots, N$
END FOR
```

$$\mathrm{Loss}_T(L) \leq C \cdot \mathrm{Loss}_T(\mathcal{E}_i) + \frac{C}{\eta} \ln \frac{1}{q_i} \tag{1}$$

2.3.4 Aggregating Algorithm for Specialist Experts (AASE)

Having introduced the Aggregating Algorithm in its base form, we can now discuss the modification that this paper's experiment will be centred around:

the Aggregating Algorithm for Specialist Experts.

The use of the term ‘specialist’ was first introduced by the work of Avrim Blum [11] for the Winnow and Weighted-Majority algorithms, and can be thought of as a natural extension to traditional experts insofar as it enables these ‘specialists’ to abstain from making a prediction “when the current expert does not fall into their ‘specialty”’. While the criteria for an expert to abstain from making a prediction is sufficient in our context, it can also be extended to allow for other scenarios like those suggested in [12], namely if “a prediction algorithm [sees] that its internal confidence is low and [decides] to skip a turn in order to re-train” or if a prediction algorithm breaks down, as would be the case if a regression algorithm “[has] its matrix very close to singular.”

In order to accommodate these specialist experts, the Prediction with Expert Advice Framework given in (1) has to be modified as follows:

Protocol 3 Modified Prediction with Expert Advice Framework

```
FOR $t = 1, 2, \dots$
    nature chooses a subset of experts $\mathcal{E}_i \in \mathcal{E}$ that are awake
    awake experts $\mathcal{E}_1, \dots, \mathcal{E}_N$ output predictions $\gamma_t^1, \dots, \gamma_t^N \in \Gamma$
    learner $L$ outputs $\gamma_t \in \Gamma$
    nature outputs $\omega_t \in \Omega$
    awake experts $\mathcal{E}_1, \dots, \mathcal{E}_N \in \mathcal{E}_i$ suffer losses $\lambda(\gamma_t^1, \omega_t), \dots, \lambda(\gamma_t^N, \omega_t)$
    learner $L$ and sleeping experts $\mathcal{E}_j \notin \mathcal{E}_i$ suffers loss $\lambda(\gamma_t, \omega_t)$
END FOR
```

As referenced above, another colloquial way of referring to ‘specialist experts’ is ‘sleeping experts’; Freund postulated that “a specialist is awake when it makes a prediction and that it is asleep otherwise”, going so far as to refer to the traditional on-line prediction framework as “the insomniac framework since it is a special case in which all specialists are awake all the time.” [3] This colloquialism is useful when adapting the bounds of the base Aggregating Algorithm because a natural interpretation of what happens when an expert is sleeping is that it simply “joins the crowd” [12], meaning that it mimics the learner’s prediction on the time steps that it is asleep because the learner’s prediction is formed based on the weighted majority of experts’ predictions. Given this definition, it can be seen that on some time steps t, the learner’s prediction and the expert $\mathcal{E}_i$’s predictions are the same; $\gamma_t = \gamma_t^i$. Recall that, in the mixable case, the Aggregating Algorithm guarantees that the following inequality is satisfied:

$$\sum_{t=1}^{T} \lambda(\gamma_t, \omega_t) \leq \sum_{t=1}^{T} \lambda(\gamma_t^i, \omega_t) + \frac{1}{\eta} \ln \frac{1}{q_i}$$

Typically, the Aggregating Algorithm's performance is measured in terms of the learner's cumulative loss compared to the best expert's cumulative loss but given that, on certain time steps t, $gamma_t = \gamma_t^i$, it is clear that the corresponding terms in both sums cancel out and what is left are the sums over the time steps where the learner's and the expert's predictions are different, i.e. where expert $\mathcal{E}_i$ is awake. What follows from this is that, instead of wanting the learner's loss to be nearly as good as the best expert's loss over a period of time T, we judge the Aggregating Algorithm for Specialist Experts' performance based on the learner's loss compared to the best expert's $\mathcal{E}_i$ loss over the steps in which it was awake. A learner following the algorithm achieves a cumulative loss that satisfies the following inequality:

$$\sum_{\substack{t=1,2,\dots,T: \\ \mathcal{E}_i \text{ is awake} \\ \text{on step } t}}^{T} \lambda(\gamma_t, \omega_t) \leq C \cdot \sum_{\substack{t=1,2,\dots,T: \\ \mathcal{E}_i \text{ is awake} \\ \text{on step } t}}^{T} \lambda(\gamma_t^i, \omega_t) + \frac{C}{\eta} \ln \frac{1}{q_i} \tag{2}$$

As is the case for the traditional Aggregating Algorithm, we make no assumptions about the outcome-generating mechanism (including the existence of such a mechanism) and this bound holds for *any* adversarial strategy, meaning that the adversary cannot inflict a large loss on the learner without inflicting a large loss on the specialists and ensuring that the performance will be good whenever there is a good mixture of specialists.

Algorithm 2 Aggregating Algorithm for Specialist Experts (AASE)

```
initialise weights $w_0^i = q_i, i = 1, 2, \dots, N$
FOR $t = 1, 2, \dots$
    read the awake experts' predictions $\gamma_t^i, i = 1, 2, \dots, N$
    normalise the awake experts' weights
        $p_{t-1}^i = w_{t-1}^i / \sum_{j:\mathcal{E}_j \text{ is awake}} w_{t-1}^j$
    output $\gamma_t \in \Gamma$ that satisfies the inequality for all $\omega \in \Omega$:
        $\lambda(\gamma_t, \omega) \leq -\frac{C}{\eta} \ln \sum_{i:E_i \text{ is awake}} p_{t-1}^i e^{-\eta\lambda(\gamma_t^i, \omega)}$
    observe the outcome $\omega_t$
    update the awake experts' weights $w_t^i = w_{t-1}^i e^{-\eta\lambda(\gamma_t^i, \omega_t)}$
    update the sleeping experts' weights $w_t^i = w_{t-1}^i e^{-\eta\lambda(\gamma_t, \omega_t)/C(\eta)}$
END FOR
```

2.4 Conclusions

Summary: Recap of key points covered in the literature review.
Implications: Implications of the reviewed literature for the current study.

3 Experiment Design and Methodology

Research Design: Overview of the experimental framework.
Methodology: Detailed description of the methods used for data collection and analysis.
Variables: Identification of key variables and how they are measured.
Procedures: Step-by-step outline of the experimental process.

4 Analysis of Perceived Randomness

Data Presentation: Presentation of collected data in an organised manner.
Analytical Techniques: Methods used to analyse the data.
Results: Detailed presentation of findings.
Discussion: Interpretation of results in the context of perceived randomness.
Comparison with Literature: How the findings align or differ from existing research.

5 Conclusions *(1,500)*

Summary of Findings: Recap of the main findings of the study.
Contributions: Discussion on the contributions of the study to the field.
Limitations: Identification of any limitations encountered during the research.
Future Work: Suggestions for future research based on the findings and limitations of this study.

References

[1] V. Vovk, "Aggregating strategies," in *Colt Proceedings 1990*, pp. 371–383, San Francisco: Morgan Kaufmann, 1990.

[2] V. Vovk, "A game of prediction with expert advice," *Journal of Computer and System Sciences*, vol. 56, no. 2, pp. 153–173, 1998.

[3] Y.Freund, R. Schapire, Y. Singer, and M. Warmuth, "Using and combining predictors that specialize," *Conference Proceedings of the Annual ACM Symposium on Theory of Computing*, 01 1997.

[4] Y. Kalnishkan, D. Adamskiy, A. Chernov, and T. Scarfe, "Specialist experts for prediction with side information," in *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pp. 1470–1477, 2015.

[5] N. Cesa-Bianchi, Y. Freund, D. Haussler, D. P. Helmbold, R. E. Schapire, and M. K. Warmuth, "How to use expert advice," *J. ACM*, vol. 44, p. 427–485, may 1997.

[6] Y. Kalnishkan and M. Vyugin, "The weak aggregating algorithm and weak mixability," *Journal of Computer and System Sciences*, vol. 74, no. 8, pp. 1228–1244, 2008. Learning Theory 2005.

[7] V. Vovk, "Competitive on-line statistics," *International Statistical Review*, vol. 69, 2001.

[8] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control.* John Wiley & Sons, 2015.

[9] D. Adamskiy, A. Bellotti, R. Dzhamtyrova, and Y. Kalnishkan, "Aggregating algorithm for prediction of packs," *Machine Learning*, vol. 108, pp. 1231–1260, 2019.

[10] Y. Kalnishkan, "The aggregating algorithm and laissez-faire investment," Tech. Rep. CLRC-TR-09-02, Royal Holloway, University of London, 2009.

[11] A. Blum, "Empirical support for winnow and weighted-majority algorithms: Results on a calendar scheduling domain," *Machine Learning*, vol. 26, pp. 5–23, 1997.

[12] Y. Kalnishkan, "Prediction with expert advice for a finite number of experts: A practical introduction," *Pattern Recognition*, vol. 126, p. 108557, 2022.