UNIVERSITY OF Southampton

University of Southampton Research Repository

Copyright © and Moral Rights for this thesis and, where applicable, any accompanying data are retained by the author and/or other copyright owners. A copy can be downloaded for personal non-commercial research or study, without prior permission or charge. This thesis and the accompanying data cannot be reproduced or quoted extensively from without first obtaining permission in writing from the copyright holder/s. The content of the thesis and accompanying research data (where applicable) must not be changed in any way or sold commercially in any format or medium without the formal permission of the copyright holder/s.

When referring to this thesis and any accompanying data, full bibliographic details must be given, e.g.

Thesis: Author (Year of Submission) "Full thesis title", University of Southampton, name of the University Faculty or School or Department, PhD Thesis, pagination.

Data: Author (Year) Title. URI [dataset]

UNIVERSITY OF SOUTHAMPTON

Faculty of Social Science
School of Business

The Measurement of Expert Judgement Uncertainty in Central Bank Forecasting

by

Yujia Chang

A thesis for the degree of
Doctor of Philosophy

November 2024

University of Southampton

Abstract

Faculty of Social Science
School of Business

Doctor of Philosophy

The Measurement of Expert Judgement Uncertainty in Central Bank Forecasting

by Yujia Chang

This thesis provides a comprehensive analysis of the determinants of performance and behavioural uncertainty in professional forecasters' macroeconomic predictions. It introduces a novel framework that integrates statistical, psychological, and computational techniques to model behavioural uncertainty under rational expectations. Additionally, it makes empirical contributions by reanalysing the UK-SPF datasets, offering fresh insights into the use of survey-based projections for real-world decision-making.

The thesis comprises three main chapters. Chapter 2 introduces a knowledge elicitation framework to assess expert performance based on statistical accuracy and knowledge informativeness, highlighting substantial variations in experts' abilities. Chapter 3 extends this analysis by exploring expert behaviour from a cognitive perspective, classifying forecasters into risk attitude groups (optimists vs. pessimists) and evaluating whether their predictions align with rational behaviour. Chapter 4 presents a hybrid framework combining machine learning (SVR, RF) and deep learning (DNN, LSTM) models to optimise the Bank of England's external professional macroeconomic forecasts. This approach offers an innovative solution to selecting optimal hyperparameters, a key challenge in machine learning, and demonstrates the effectiveness of these methods, even with limited data.

The key contributions of this thesis lie in developing new methods to evaluate expert performance, including scoring forecasting accuracy and informativeness, introducing a cognitive perspective to forecasting behaviour, and advancing the application of machine learning in macroeconomic prediction. These findings enhance our understanding of expert biases, improve predictive accuracy, and offer practical implications for decision-making in economic forecasting.

Contents

List of Figures

List of Tables

Declaration of Authorship

I declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. 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;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. 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;
5. I have acknowledged all main sources of help;
6. 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;
7. Parts of this work have been published as:

Signed:..... Date:.....

Acknowledgements

I am profoundly grateful to those who have walked beside me on this PhD journey.

To my supervisors, Prof. Mario Brito, Prof. Tapas Mishra, and Prof. Di Luo, for their wisdom and patience. To my peers, whose shared laughter and late-night discussions illuminated even the most challenging days. To my family, whose unwavering love has been my constant anchor. And to my friends, whose words uplifted me when I needed it most. And to my beloved pet, Idol, whose quiet companionship brought warmth and comfort through many long hours. This journey was never mine alone, and I thank you all from the depths of my heart.

I would also like to express my heartfelt gratitude to the Bank of England for providing the invaluable data that played a pivotal role in shaping this research. Their resources not only enabled this study but also sparked my curiosity and deepened my passion for exploring the complexities of this field.

My last tribute is to those who know I am not perfect but still love me.

To my grandfather, who, I think, would be very happy to see the accomplishment of this thesis, if he could.

Definitions and Abbreviations

w	Weight vector is used to aggregate expert judgements.
λ	Regularization parameter in statistical models.
α	Learning rate or significance level in hypothesis testing.
β	Coefficient representing relationships between variables in regression models.
γ	Scale parameter in Cauchy-Lorentz distribution.
δ	Change or difference in values, often used in time series analysis.
σ	Standard deviation, representing data dispersion.
θ	Parameter or angle in optimization problems.
μ	Mean or expected value of a distribution.
ρ	Correlation coefficient between variables.
ϕ	Probability density function in statistics.
ξ	Represents random variables or shocks in econometric models.
CM	Classical Model for expert judgement .
EXCALIBUR	Expert judgement aggregation procedure.
UK-SPF	UK Survey of Professional Forecasters.
SVR	Support Vector Regression.
RF	Random Forest.
DNN	Deep Neural Networks.
LSTM	Long Short-Term Memory network.
ME	Mean Error, measures the average error between predictions and actual values.
MAE	Measures the average absolute error between predictions and actual values.
RMSE	Quantifies the differences between predicted and observed values.

Chapter 1

Introduction

1.1 Research Background

1.1.1 Background

In recent decades, there has been a substantial growth in academic attention across various disciplines towards using judgement-based methods for making predictions. Particularly, the attitudes of researchers regarding the role of expert judgement experience a significant reverse turn (Lawrence et al., 2006). Previously, it was commonly believed that judgement was inherently subjective. Certain subjective components, especially those that involve unquantifiable elements, were perceived as obstacles to achieving accuracy. However, subsequently, expert opinion and judgement have been better understood for their significant strengths in forecasting by researchers. Expert judgement has been recognised as one of the most widely used estimation methods. It has now been integrated into the practice of supporting statistical inference and decision-making in various ways. Judgemental knowledge inevitably entails subjectivity; however, through the process of careful, objective, and scientific elicitation, it is feasible to mitigate potential biases (O'Hagan, 2019).

In the field of macroeconomic forecasting, expert judgement—also referred to as "professional forecasters' forecasting"—plays a pivotal role in shaping economic activities, market interactions, and sentiments, whether in an overt or subtle manner. Central banks provide estimates of future economic growth, where positive growth projections can bolster market confidence, while lower-than-expected projections may raise concerns about an economic slowdown or recession, thereby influencing market sentiment. As highlighted by (Huang et al., 2022), when economic agents face substantial or dynamic uncertainty, these signals are first propagated to different layers of iterative economic markets. Subsequently, the resulting emotions and moods influence individual decision-making behaviours underlying economic activities. These emotions are

often interconnected and can shift rapidly, driving market trends and outcomes. Consequently, they play a significant role in shaping market movements, which can lead to fear, panic, or doubt.

Does the central bank's projection report affect market activity and sentiment? The answer is a definitive yes. Central banks, such as the Federal Reserve in the United States, the European Central Bank, and the Bank of England, play a pivotal role in shaping monetary policy and economic conditions. Their projection reports provide crucial insights into future economic trends, including interest rates, inflation, and other key factors. These reports directly influence market expectations, often driving changes in investor behaviour, financial market performance, and overall market sentiment (Gürkaynak et al., 2004). As a result, markets closely monitor these reports, reacting to signals about potential shifts in economic policy or outlook.

Otway and von Winterfeldt (1992) proposed that expert judgement has always played a significant role in the forecasting analysis of regulation and management even though it is often unrecognised. Especially, the hazard activities increasingly rely on formal expert judgement processes to provide wisdom that is unable to be directly supplied from practical science. Brito et al. (2008) emphasised that risk assessment of complex systems is heavily dependent on expert judgement elicitation. This is particularly the case for problems where there is no hard data, and the consequences of potential hazards can be catastrophic. Marti et al. (2021) confirmed expert elicitation plays a prominent role in fields where the data are scarce. As consulting multiple experts is critical in expert elicitation practices, combining various expert opinions is an important topic.

Forecasting plays an important role in economic analysis and affects the decisions of households, firms, and policymakers. This, in turn, has important consequences for macroeconomic dynamics as emphasised for instance by (Lucas, 1973). How agents form expectations and, particularly, whether they are rational and efficiently incorporate all available information into their forecasts is thus a question of fundamental economic importance (Elliott and Timmermann, 2008). Debaere (2008) described that GDP growth, Inflation, and Unemployment are the "big three" indicators that are carefully monitored by consumers, firms, and policymakers worldwide. They are the scorecard of an economy and allow people to understand its overall health. Inflation reflects changes in the overall price level. Inflation is a key indicator for the central banks around the world. Although the academic debate about expectation formation is still open, the role played by inflation expectations and accurate measurements of the public's beliefs is important to both researchers and policymakers. In addition, to monitor the effectiveness of its communication, a central bank needs to regularly assess the consistency of the public's beliefs with policy objectives.

When complex decisions must be made while data is unavailable, structured expert judgement can be used to combine uncertainty distributions resulting from experts'

assessments. Expert opinions are frequently sought when complex decisions must be made in situations where appropriate information cannot be acquired from existing data and models. Experts are asked to quantify their uncertainty over quantities of interest that inform the decision-making process. Furthermore, the experts are unlikely to be in complete agreement with one another. In such situations, expert judgement can be employed to quantify the uncertainty that ensues and to aggregate expert opinion. Structured expert judgement methods are intended to quantify uncertainty, not to remove it from the decision process.

1.1.2 Challenges in Forecasting Accuracy

Forecasting, especially by professional forecasters, presents significant challenges in achieving accuracy due to a variety of factors. One major challenge is the inherent uncertainty and volatility of economic and financial environments, which can render even the most sophisticated models prone to errors (Keane and Runkle, 1990). Professional forecasters must continuously adapt to rapidly changing conditions, such as unexpected geopolitical events, natural disasters, and sudden market shifts, all of which can disrupt previously stable patterns (Elliott et al., 2005). Moreover, the reliance on historical data as a basis for predictions often falls short when unprecedented scenarios arise, leading to potential biases and inaccuracies (Armstrong, 2001). Cognitive biases and heuristic-driven decision-making further complicate the forecasting process, as forecasters may inadvertently project their subjective views onto objective analyses (Tversky and Kahneman, 1974). These challenges underscore the complexity of producing reliable forecasts and highlight the need for continual refinement of forecasting methodologies to enhance their robustness and accuracy (Fildes and Goodwin, 2007).

1.1.3 Definition of Expert Judgement

An expert is an individual with extensive knowledge, skills, and experience gained through both education and practical engagement in a specific field. Informally, an expert is someone who is widely acknowledged as a dependable source of techniques or skills, and their ability to make sound, just, or wise judgements is granted authority and recognition by their peers or the general public within a well-defined domain characterised by established cognitive structures and processes (Ericsson and Staszewski, 2013). In a broader sense, expertise refers to a person's in-depth knowledge or proficiency rooted in research, hands-on experience, or a particular occupation within a specific area of study.

In particular domains, the definition of an expert is commonly accepted through consensus, and formal professional or academic qualifications may not always be a prerequisite for expert recognition. For instance, consider a shepherd with five decades

of practical experience in tending flocks; such an individual is widely regarded as possessing comprehensive expertise in areas such as sheepdog training and sheep care. Another example is found in the realm of computer science, where an expert system can be taught by a human and subsequently regarded as an expert, often surpassing human performance in specific tasks.

The definition of experts and expert judgement used in this study focuses on UK-SFP professional forecasters. Central banks, such as the Bank of England, routinely collect and assess economic forecasts from various sources, including academic institutions, financial organisations, research bodies, and independent economists. These forecasts provide valuable insights into collective expectations for economic conditions, which can significantly influence the central bank's decisions on monetary policy.

1.2 General Research Contributions

The general research contributions of this thesis are threefold:

- Expanding the literature by bridging measurable statistical accuracy with implicate expert knowledge informativeness: This research establishes a novel connection between quantifiable statistical precision and the insights provided by expert judgement. It introduces an innovative approach to understanding and managing uncertainty, offering a more intuitive framework for decision-making in complex, uncertain environments.
- Offering a fresh perspective on the interpretation of attitudinal differences among experts in macroeconomic forecasting: This thesis provides a new lens for analysing how experts' attitudes towards uncertainty and bounded rationality influence their predictions. By recognising the limitations of human cognition in decision-making, it highlights how judgements made under conditions of bounded rationality shape forecasting outcomes. Furthermore, this research bridges cognitive theory and statistical models, creating a clearer pathway for understanding the complex interplay between human judgement, rational limitations, and predictive accuracy in economic forecasting.
- Developing an advanced explainable hybrid machine learning framework for macroeconomic prediction: This research introduces a robust hybrid approach that combines different machine learning techniques to significantly enhance the accuracy and interpretability of macroeconomic forecasts. By integrating Support Vector Regression (SVR), Random Forest (RF), Deep Neural Networks (DNN), and Long Short-Term Memory (LSTM) models, this hybrid framework addresses the limitations of individual models and improves prediction performance. Moreover, the framework emphasizes explainability, providing insights

into how various input features influence the predictions. This level of transparency enables policymakers and analysts to better understand the drivers behind forecast outcomes, fostering trust and facilitating more informed decision-making.

1.3 Research Aims

Specific research aims of each main chapter are displayed as follows:

The research aim of Chapter 2 is to explore the utilization of known historical SPF data in forming a novel expert expectation for macroeconomic predictions. We quantify each individual expert performance uncertainty by assessing their performance using calibration scores and information scores. Subsequently, these scores are employed to assign new weights to each expert, resulting in a combined expert expectation for forecasting macroeconomic indicators.

The research aim of Chapter 3 is to investigate whether professional forecasters exhibit behaviourally rational tendencies when making macroeconomic predictions. Instead of directly assuming rationality, we establish rational boundaries to evaluate whether expert forecasts fall within a rational range. This approach allows us to identify differences in their risk attitudes, categorising these forecasters as either optimistic or pessimistic.

The research aim of Chapter 4 is to develop a hybrid machine learning framework to optimise macroeconomic predictions. Recognising the limitations of traditional macroeconomic forecasting models and the inefficiencies of relying solely on a single machine learning model, this chapter seeks to address these challenges. Our approach involves constructing a robust hybrid framework that integrates multiple machine learning techniques to enhance predictive accuracy. Specifically, we combine Support Vector Regression (SVR), Random Forest (RF), Deep Neural Networks (DNN), and Long Short-Term Memory (LSTM) models. This hybrid methodology is designed to address and minimise prediction errors, offering an innovative solution for macroeconomic forecasting.

1.4 Research Objectives

The research aims of this thesis can be expressed as specific research objectives in each main chapter, followed by corresponding conclusions.

The research objectives of Chapter 2:

- To introduce the application of structured expert judgement into macroeconomic prediction.
- To transform expert predictions from time-series numerical data into probability distributions representing expert beliefs.
- To score and weight expert performance based on their predictions of key macroeconomic indicators.
- To elicit the aggregation of experts' judgements to form a new expert expectation for the macroeconomic outlook.

The research objectives of Chapter 3:

- To visually depict the shape of expert attitudes based on the cumulative probability distribution of expert judgement.
- To measure and combine the classification methods of identifying expert risk-taking attitudes under different methods.
- To understand the methods of establishing rational boundaries for understanding forecasters' behaviour.
- To connect and extend the individual difference of expert prediction behaviour into psychology theory.

The research objectives of Chapter 4:

- To address a gap in the literature by utilising machine learning applications to optimize macroeconomic expert predictions.
- To measure the performance between different algorithms (ML and DL) to understand their respective contributions to the process of improving prediction accuracy.
- To validate machine learning and deep learning work on small data samples.
- To explore the integration of hybrid machine learning frameworks that combine traditional and advanced algorithms, aiming to enhance robustness and adaptability in macroeconomic forecasting.

1.5 Structure of the Thesis

This thesis is structured around three core papers, beginning with an overall introduction and concluding with remarks on key findings, limitations, and future directions.

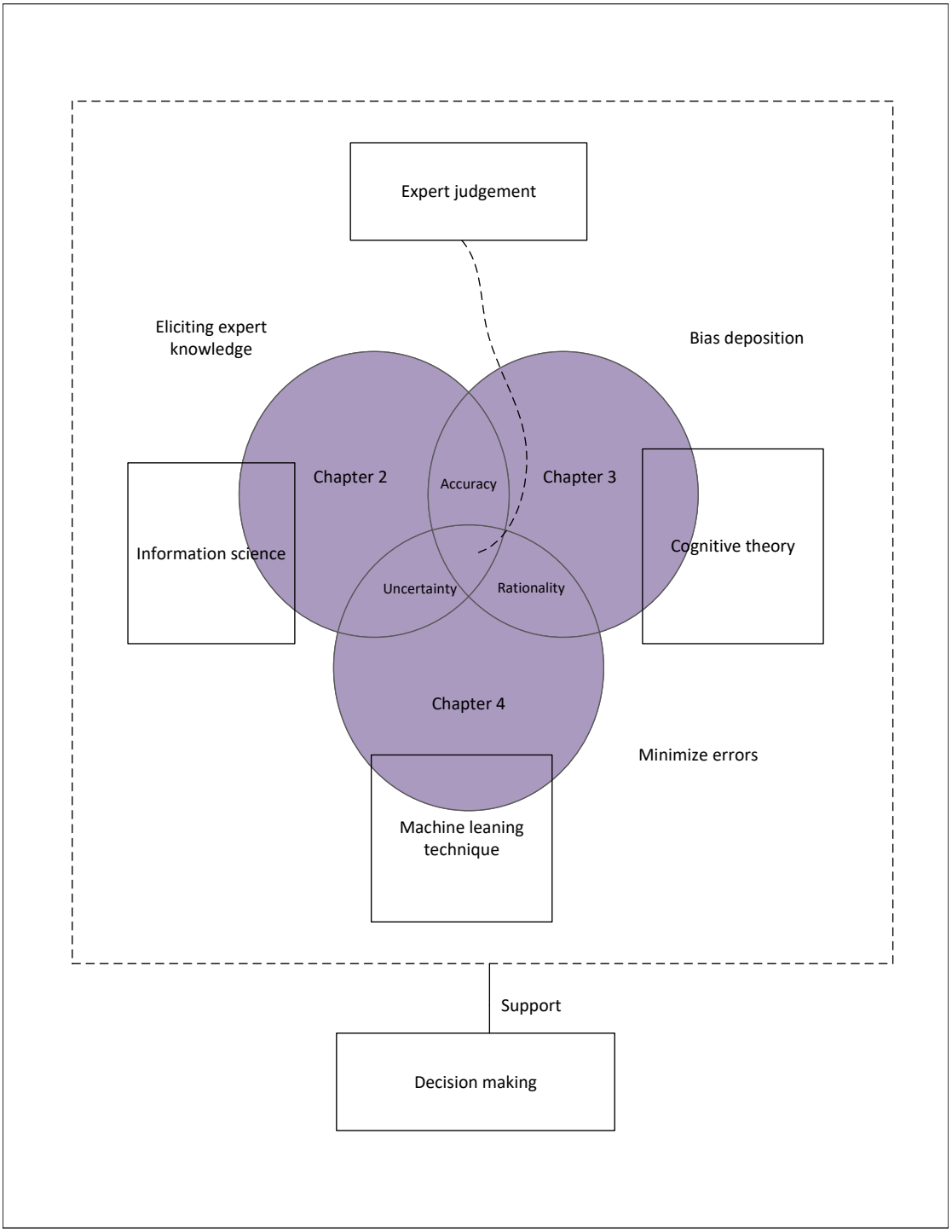

FIGURE 1.1: The basic structure of this thesis

Chapter 2

Scoring and Eliciting Expert Judgemental Knowledge under Uncertainty

2.1 Introduction

Professional forecasters' accuracy and uncertainty identification is an ongoing research topic in macroeconomic variable forecasting (Dror and Charlton, 2006). In macroeconomic decision-making, professional forecasters play a significant role in informing central bank policymakers, businesses, and the general public about the potential future trends of some key economic indicator variables, such as GDP growth, inflation, interest rates, unemployment, and exchange rate (Ball et al., 2005; Manski, 2004; Paciello and Wiederholt, 2014; Angeletos and Lian, 2017). Thus, the performance of professional forecasters in predicting the future state of leading macroeconomic indicators has received great attention. Galí (2011) questioned if central banks' projections are meaningful by the concern of central banks' forecasts conditional on a given path, which are often criticised on the grounds that their assumptions are inconsistent with the existence of a unique equilibrium in many forward-looking models. In addition to this, Fildes and Stekler (2002) argued that traditional methods of measuring accuracy, such as ME, MAE, and RMSE, do not provide sufficiently meaningful information and have been subject to significant criticism. These concerns motivate this study to assess the performance uncertainty of the Bank of England's external professional forecasters from two key dimensions: forecast accuracy and knowledge informativeness.

Morgan et al. (1990) indicated statistical models prove inadequate in supplying the necessary inputs for modelling problems or policy analysis, the best course of action is to seek insights from experts. This is in line with Kynn (2008) emphasised that although statistical modelling has a valid but limited application, it cannot fully replace expert

judgement. Colson and Cooke (2018) pointed to existing data and modelling tools that cannot provide decision-makers with all of the information they need to design and implement effective policies and make optimal management choices. Thus, decision-makers often supplement other forms of information with the judgement of experts. McAndrew et al. (2021) mentioned the key concern is that even though many statistical models can produce accurate forecasts with abundant training data, they may struggle to capture underlying dynamics when data is sparse or rapidly changing. McAndrew et al. (2021) further highlighted statistical models heavily rely on correlations between data to identify patterns, but when the data is inadequate for modelling, the accuracy of predictions may fall short. As a solution, McAndrew et al. (2021) suggested overcoming data limitations in statistical models by adapting judgemental forecasting and eliciting expert knowledge.

Furthermore, Clements and Hendry (2011) found that while the optimal combinations of survey and model-based forecasts always outperform the latter, they do not systematically do so over the former. And survey forecasts provide an effective way of removing expected variations in macroeconomic series. Casey (2020) highlighted subjective forecasts by professionals have often been found to be more accurate than forecasts from econometric models, especially over short horizons. However, going back to earlier works, we find that few studies discuss the formalization of the aggregation expert process; even the professional forecasters in macroeconomic predictions represent a critical role. However, even though professional forecasts are important, the measurement of economic forecast accuracy remains under debate. As Mankiw et al. (2003) revealed the concern of assumption in most theories in macroeconomics is no disagreement among agents. It is assumed that everyone shares the same information and that all are endowed with the same information processing technology. Consequently, everyone ends up with the same expectation. However, they provide evidence to reveal three facts of disagreement by examining inflation expectations and indicating that disagreement is crucial to macroeconomic dynamics.

We encounter similar concerns when examining the projection data of external professional forecasters from the Bank of England. Two primary issues arise: firstly, the survey data is incomplete, with many participants failing to provide their expected values for the following year's outlook. Secondly, preliminary analysis reveals that the prediction data significantly deviates from historical actual values. These issues motivate us to assess the performance of these professional forecasters and investigate the factors contributing to inefficiencies in forecasting the main macroeconomic indicators.

We introduce Cooke's Classical Model (CM), developed by Cooke (1991), to evaluate the performance of professional forecasters by assessing their prediction accuracy and the informativeness of their knowledge. First, the study identifies varying degrees of prediction deviations between the forecasts made by external professional forecasters

and actual historical data for key economic indicators, including GDP growth, CPI inflation (CPI), the Bank of England base rate (BoE), the LFS unemployment rate (UR), and the Sterling Exchange Rate Index (ERI). This finding illustrates the observable differences between these forecasts and actual historical data, prompting further exploration into the underlying causes. Subsequently, two scores—the calibration score and the information score—are calculated using the CM for all experts. These scores are then used to derive a performance-based weight for each expert. Finally, an aggregated forecast is generated by combining the performance-weighted contributions of all experts, resulting in a forecast that demonstrates improved accuracy compared to those of individual forecasters.

This study is organised into five main sections. Section 2 reviews the literature on uncertainty measurement in economics and introduces the application of the Classical Model (CM). Section 3 outlines our methodology, including calibration and information scoring principles. Section 4 provides a detailed description of the data used in this study. Section 5 concludes with a summary of the research findings, a discussion of the study's limitations, and suggestions for future research.

2.2 Literature Review

2.2.1 Forecast Accuracy and Uncertainty

Bloom (2009) mentioned uncertainty is also a ubiquitous concern of policymaker. Abel et al. (2016) claimed that uncertainty is of great significance for understanding the expectation formation process and potentially explaining changes in key economic and financial time series. However, they also confirmed that the measurement of uncertainty remains challenging due to its difficulty in observing the subjective magnitude of an individual. In their study, they examined the matched point and density forecasts of output growth, inflation, and unemployment from the ECB survey of professional forecasters. Jo and Sekkel (2019) defined macroeconomic uncertainty as the conditional time-varying standard deviation of a factor that is common to the forecast errors for various macroeconomic indicators from the Survey of Professional Forecasters (SPF).

Professional macroeconomic forecasting encourages confidence-building in financial market activities and allows for more informed decision-making. With this said, inaccurate prediction, whether the future state of the economy is underestimated or overestimated, will invariably lead to losses stemming from irrational decisions and rising costs Buturac (2021). In addition, Hess and Orbe (2013) found some surprising data and unanticipated information in regular scheduled macroeconomic outlook releases, which will change market participants' perceptions and behaviours in their engagement in economic activities.

In addition to forecast accuracy, forecast uncertainty has attracted much attention in the evaluation process of macroeconomic forecasting. Uncertainty arises in prediction is attributable to forecasters need to express their uncertainty in their forecasts. This uncertainty has received the focus of government policymakers, analysts, and researchers to find out its proxy variables in the process of evaluating the current economic situation and future economic states (Boero et al., 2015). Further, Abel et al. (2016) highlighted that uncertainty is of great significance for understanding the expectation formation process and potentially explaining changes in economic behaviours. The challenge remains to empirically assess the behaviour of uncertainty and the corresponding influences on macroeconomic market activity (Jurado et al., 2015).

Uncertainty affects a variety of decision-makers in the world including governments and a large number of tools have been established to deal with these uncertainties, and there is no need to eliminate uncertainty if it can be quantified by Oppenheimer et al. (2016). In contrast, Huang et al. (2022) approached uncertainty in a macroeconomic setting as a systemic risk that poses challenges for most economic actors regarding measurement. For instance, when participants in the economic market perceive heightened unpredictability, it can influence their decision-making behaviour.

2.2.2 Expert Judgement and Knowledge Elicitation

Expert judgement elicitation (EJE) is described by Dias et al. (2018) as the process of aiding experts in quantitatively expressing their subjective judgements, encompassing both factual and evaluative aspects. The process involves a facilitator extracting assessments from highly knowledgeable experts to identify and measure risks and uncertainties. This approach aims to minimize bias introduction and enhance the reproducibility of results. The initial application of this approach took place during safety assessments carried out by the US Nuclear Regulatory Commission in 1975. Kynn (2008) introduced a formal process for eliciting judgements aimed at addressing and minimizing biases, thereby improving the replicability of outcomes. This process encompasses multiple phases. Various formal expert judgement elicitation (EJE) techniques, such as the Delphi method Turoff and Linstone (2002), the SHELF-R framework Morgan et al. (1990), and the EXCALIBUR approach Cooke and Solomatine (1992), offer different approaches to model and aggregate expert judgements.

Burnham et al. (1998) outlined a process involving the interaction of experts, followed by some basic mathematical manipulation of each expert's judgement to yield a single aggregated probability density function per variable. Typically, these approaches employ straightforward combination techniques, such as assigning equal weight to all participating experts. There are various approaches for aggregating expert judgements; these approaches are categorised as "mathematical" and "behavioural" Clemen and Winkler (1999). Mathematical methods aim to create a single composite assessment for

each variable or item of interest by applying procedures or analytical models that treat each variable independently (Budnitz et al., 1998). Morris (1977) stated that Bayesian aggregation methods necessitate decision-makers to define their prior probability distribution. The a priori decision-maker's belief is updated after acquiring the probabilities from the experts. In addition, the linear weighted opinion pool method is widely adopted for its simplicity as described by (Clemen and Winkler, 1999). The mathematical equation for aggregating expert judgement technique is expressed as follows in Equation 2.1 by (Clemen and Winkler, 1999).

$$p(\theta) = k \cdot \sum_{i=1}^{n} w_i p_i(\theta), \tag{2.1}$$

where n is the number of experts, θ represents expert's probability distribution for unknown θ, p (θ) represents the combined probability distribution, and the weights w_i are non-negative and sum to one. This method aggregates Expert Judgements (EJs) by calculating the weighted average of the assessments offered by the experts. Additionally, to address situations where the decision-maker or facilitator must update probabilities when new relevant information becomes available, Clemen and Winkler (1999) introduced the method of logarithmic opinion pool, as an add. On the other hand, Hanea et al. (2018) noted behavioural aggregation involves convening experts, providing them with comprehensive information, and encouraging open discussions of their viewpoints.

Subsequently, they collectively generate distributions (Phillips and Phillips, 1993; O'Hagan et al., 2006). During these interactions, certain behavioural approaches, such as the expert information approach (Kaplan, 1992), strive to achieve a consensus among experts regarding each variable's ultimate probability density function. Popular methods for this purpose include Group Elicitation (Porthin et al., 2018), the Delphi method by (Rowe and Wright, 2001), and the Nominal Group Technique (McMillan et al., 2016). Behavioural methods are time-intensive and can lead to systematic biases from group polarization, in contrast to mathematical aggregation methods (Isenberg, 1986).

A collective of experts often demonstrates superior performance compared to an individual expert. Nevertheless, there are instances where the most proficient individual within a group can still outshine the entire group, as noted by (Clemen and Winkler, 1999). This observation encourages using methods that solicit evaluations from individual experts without fostering interaction among them during the actual elicitation phase. Subsequently, a straightforward mathematical aggregation process is employed to derive a single assessment for each variable. This approach ensures that assessments from individual experts are obtained impartially and are weighted based on the performance and merit of each expert.

2.2.3 Expert Performance Evaluate Techniques

The Classical Model (CM) has found extensive use in various professional applications aimed at quantifying uncertainties to facilitate informed decision-making (Hanea et al., 2021). These uncertainties often pertain to unmeasured variables that exist on a continuous scale. Simply providing point or "best" estimates proves inadequate when the primary objective is to quantify uncertainty since they fail to convey the potential range within which the actual (unknown) values may plausibly deviate from these point estimates. In the CM approach, expert uncertainties are thus quantified as subjective probability distributions. Experts are asked to provide points that describe the distribution in the form of a fixed and finite number of percentiles (usually three). From these percentiles, a minimally informative non-parametric distribution is constructed. Parametric distributions may be fitted instead, but these will add extra information to the three percentiles provided by the experts when compared to the minimally informative non-parametric distribution. This extra information may or may not be following experts' views.

2.3 Methodology

Our baseline model, originally proposed and designed by (Cooke, 1991), is intended for science-based quantitative uncertainty analysis. It is widely recognised as a performance-based approach for mathematically aggregating judgements from a panel of experts, facilitating reasoning about target questions under conditions of uncertainty (Quigley et al., 2018).

2.3.1 (Expert) Knowledge Elicitation Framework

What is knowledge elicitation? As Shadbolt et al. (2015) described, knowledge elicitation consists of techniques and methods that attempt to elicit knowledge from experts in their domain. They emphasize that the conceptualizations of knowledge elicitation have varied from extracting or mining knowledge from experts' brains in the early stage to developing the process as a modelling exercise oriented. It combines the collaboration of knowledge elicitor (analyst) and domain experts to create a model of expert knowledge. Verdolini et al. (2018) further indicated that expert elicitation is a structured approach for obtaining expert judgements about items of interest to decision-makers.

This study builds upon a well-known elicitation approach, the EXCALIBUR procedure, formulated and documented by (Cooke, 1991). The standard procedure includes six main steps: selecting experts, eliciting knowledge, assessing variables, scoring experts' performance, forming weights, and combining experts' uncertainty distributions

(Quigley et al., 2018). Further details can be found in Table A.1 in Appendix A. However, this study faces challenges in data availability in several aspects. One key issue is the format of the expert prediction: the EXCALIBUR approach requires experts to express their beliefs as probabilities to construct distributions. However, since we are working with time series data from 1998, it is impossible to revise the data format retrospectively. and the experts are seen as anonymous when the Bank of England collects the data from them, so we can not interview the expert panel further to collect their beliefs in probability.

To address this limitation, we extend the approach by incorporating the probability inversion method developed by (Oppenheimer et al., 2016), allowing us to infer expert subjective distributions through the inversion of a function or parameter within a given distribution. This extension partially addresses the challenge of adapting expert point forecasts into interval forecasts. By applying probability inversion, we transform the single-value predictions provided by experts (point forecasts) into probability distributions that capture uncertainty, ultimately forming a probability belief distribution. This transmission allows us to better evaluate the accuracy and informativeness of expert judgements by providing a more nuanced view of their predictions. By capturing a single forecast value and a range of possible outcomes, we can assess how well experts understand the underlying uncertainty and how rich their insights convey probabilistic information. This method is especially valuable in time series analysis, where understanding the range of potential future outcomes is crucial for informed decision-making.

2.3.2 Classical Model Basics

The Classical Model (CM), also known as structured expert judgement (SEJ), introduced by (Cooke, 1991), is a method that uses performance scores to aggregate and validate expert judgements. In this approach, experts' uncertainty is quantified through two types of elicitation questions: target questions and calibration questions. Target questions involve variables of interest that cannot be adequately addressed through other methods, requiring expert judgement. Calibration questions, on the other hand, are related to the experts' domain but contain uncertainties. While the true values of these calibration questions are not known or accessible to the experts, they are known to the analyst, allowing for evaluating the experts' accuracy.

A key feature of the CM is its scoring mechanisms for assessing expert performance. It compares expert calibration and classical statistical hypothesis testing, employing two primary scores: calibration and information scores. The calibration score evaluates how well the experts' probability distributions align with observed empirical data, assessing the accuracy of their predictions. The information score measures the concentration of an expert's uncertainty distribution, reflecting how precisely an expert expresses

their beliefs. These scores are used to assign performance-based weights to experts, facilitating a more reliable aggregation of their judgements.

Moreover, in the CM, experts express their subjective estimates using a quantile-based format. Specifically, they provide estimates for an uncertain quantity by specifying predetermined percentiles of their uncertainty distribution, typically at the 5th, 50th, and 95th percentiles. The 50th percentile represents the median estimate, indicating the point where the expert believes the true value is equally likely to be above or below. The 5th and 95th percentiles define a 90% confidence interval, suggesting that experts believe there is a 90% chance that the true value lies within this range (Quigley et al., 2018).

2.3.3 Scoring Mechanisms

The calibration score is a metric for assessing the expert performance based on the seed questions in terms of statistical accuracy. In statistics science, it indicates the Kullback-Leibler (KL) distance between the expert's belief probability distribution and the reference distribution. According to the definition of (Kullback and Leibler, 1951; Kullback, 1997), the KL distance is denoted as $D_{KL}(P \parallel Q)$, it measures how the expert probability distribution P is different from the reference distribution Q.

Definition 2.1.: Setting each expert to express their belief using four inter-quantile intervals, represented by a probability vector (P_1, P_2, P_3, P_4), which is used to form an expert probability distribution.

Namely,

- P_1 = 0.05 is for realization value, $P_1 \leq 5\%$ value,
- P_2 = 0.45 is for realization value, $5\% \leq P_2 \leq 50\%$ value,
- P_3 = 0.45 is for realization value, $50\% \leq P_3 \leq 95\%$ value,
- P_4 = 0.05 is for realization value, $95\% \leq P_4 \leq 1$ value.

$$p = \{0.05, 0.45, 0.45, 0.05\} \tag{2.2}$$

Remark 2.1.: In this approach, we set the realization value to 0, implying that the expert's belief is consistent with the true value. We propose an inverse reasoning method to deduce the expert's belief probability distribution from their predicted value.

Step 1. Define the realization value as 0, which means that ideally it is expected no errors between the expert prediction values and true value. It denotes the KL divergence measure will be 0. It will be as a baseline for dividing expert's belief intervals in

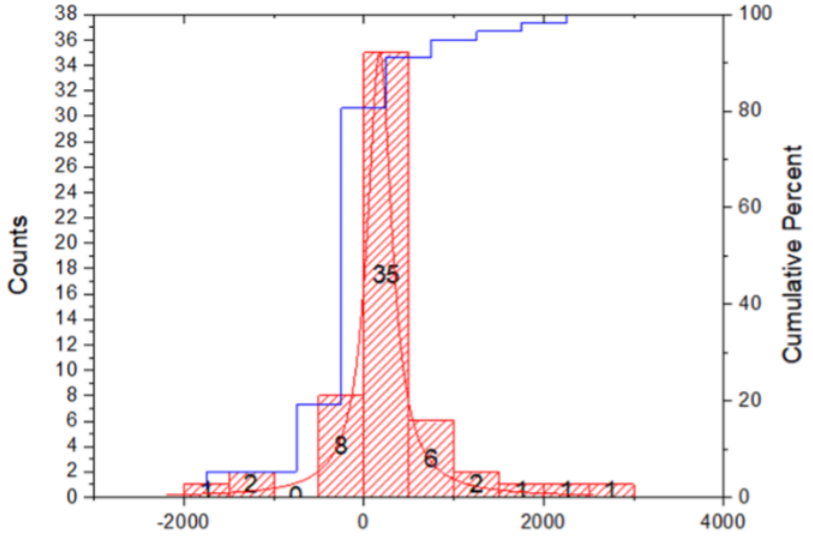

FIGURE 2.1: An example of a cumulative count graph.

Note: This graph illustrates the information on the cumulative count of errors during the fitting process, represented by the red histogram and the blue cumulative line. The numbers inside the bars indicate the counts of errors between the expert predictions and the true values within each range. The X-axis represents the error percentage (difference between predicted and actual values); the left Y-axis displays the count of errors, and the right Y-axis shows the cumulative percentage of errors.

reverse. Here we calculate the actual differences in values between expert predictions in the target's variables (GDP, Inflation, BR, UR, and ERI) and their corresponding true values.

Step 2. Plot a histogram for the difference points of the time series for each viable to create a cumulative count graph and let it fit in a Cauchy - Lorentz distribution.

Assumption 2.1:

A1: We assume that the cumulative distribution function (CDF) follows a Cauchy-Lorentz distribution due to its specific properties.

Property 1: According to (Chyzak and Nielsen, 2019), the Kullback–Leibler (KL) divergence between two Cauchy distributions has a symmetric closed-form expression:

$$KL = D_{KL}(px_{0,1}\gamma_1 : px_{0,2}\gamma_2) = \log\left(\frac{(\gamma_1+\gamma_2)^2+(x_{0,1}-x_{0,2})^2}{4\gamma_1\gamma_2}\right) \tag{2.3}$$

Property 2: As noted by Nielsen and Okamura (2022), any f- divergence between two Cauchy distributions is symmetric and can be expressed as a function of the chi-squared divergence.

Step 3: Perform nonlinear curve fitting using the Cauchy-Lorentz model. The resulting parameter values from the fitted curve will be displayed.

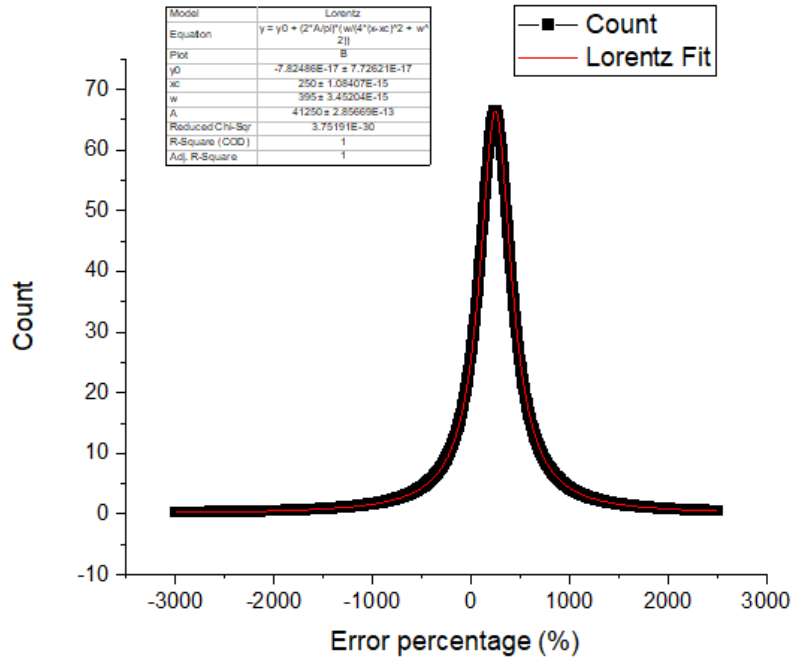

FIGURE 2.2: A cumulative count graph fits in a Cauchy - Lorentz distribution.

Nonlinear Curve Fit (Lorentz)

Notes

Input Data

Parameters

		Value	Standard Error	t-Value	Prob>|t|	Dependency
B	y0	8.31408E-16	9.21733E-17	9.02006	9.45481E-19	0.35402
	xc	175	1.42677E-15	1.22655E17	0	3.62712E-7
	w	337.14286	4.51164E-15	7.47273E16	0	0.59985
	A	28500	3.00177E-13	9.49441E16	0	0.67707
	H	53.81595	4.55831E-16			

Reduced Chi-sqr = 5.48820808958E-30
COD(R^2) = 1
Iterations Performed = 400
Total Iterations in Session = 400
Fit did not converge. Maximum iteration setting of 400 was reached.
Standard Error was scaled with square root of reduced Chi-Sqr.
H are derived parameter(s).

Statistics

	B
Number of Points	1000
Degrees of Freedom	996
Reduced Chi-Sqr	5.48821E-30
Residual Sum of Squares	5.46626E-27
R-Square (COD)	1
Adj. R-Square	1
Fit Status	Failed(-200)

FIGURE 2.3: Parameter values in a nonlinear curve fitting of Cauchy - Lorentz distribution.

```
clear all
clc

X=-100:0.1:450;%x-axis range
y0=-7.82486E-17;%
xc=250;%
w=395;%
A=41250;%
y=y0+(2*A/pi)*(w./(4*(X-xc).^2+w.^2));
%figure(1);hold;
%plot(X,y,".");
y=y';

yl=[];
yl(1)=y(1);
for i=2:1:length(y)
    yl=[yl,yl(i-1)+y(i)];
end

yl=yl/yl(length(yl));
%figure(2);
%plot(X,yl,".");

XX=find(yl>=0.0499 & yl<=0.0501);
ZZ=find(yl>=0.9499 & yl<=0.9501);

Quantiles=[X(XX(1)) xc X(ZZ(1))];
n=length(num2str(X(2)))-find(num2str(X(2))=='.');%number of decimal digits for X
num = floor(xc);
str = num2str(num);
len = length(str);
err = xc-num;
digits(n+len);
need_num = num+vpa(err,n+len);
need_str = num2str(double(need_num));

disp(['The Calculated 5% Value is: ', num2str(X(XX(1))),]);
disp(['The Calculated 50% Value is: ', num2str(need_str),]);
disp(['The Calculated 95% Value is: ', num2str(X(ZZ(1))),]);
```

FIGURE 2.4: Calculation of Quantiles from a Cauchy-Lorentz Distribution.

Note: This calculation demonstrates the process for calculating the 5%, 50%, and 95% quantiles from a Cauchy-Lorentz distribution. The code defines the distribution parameters, computes the Probability Density Function (PDF), and then generates the Cumulative Distribution Function (CDF) through a cumulative sum of the PDF values. After normalising the CDF, the code identifies the quantile values by finding the points where the CDF approximates 5% and 95%. The 50% quantile, or median, is calculated based on the distribution's centre, with precision adjustments.

Step 4. Here is a function generated from the above nonlinear curve fitting. The values of parameters derived in step 3 will be the input value to rebuild a new function for each time. Each expert's prediction for each variable will generate a new function.

Definition 2.2.: We draw on (Aspinall, 2008) to define if the realizations are indeed drawn independently from a distribution with three quantiles (5%, 50%, and 95%), then the quantity:

$$2N \cdot I(s(e_1, \ldots, e_{10}) \mid p) = 2N \cdot \sum_{i=1}^{4} \{s_i \cdot \ln\left(\frac{s_i}{p}\right)\} \tag{2.4}$$

Where $I(s(e) \mid p)$ is the relative information of distribution s with respect to p for each expert $(e_1, \ldots, e_{10})$. Let a discrete distribution have a probability function s, and let a second discrete distribution have a probability function p. Then the relative information of p with respect to s is: $s \cdot \ln\left(\frac{s}{p}\right)$.

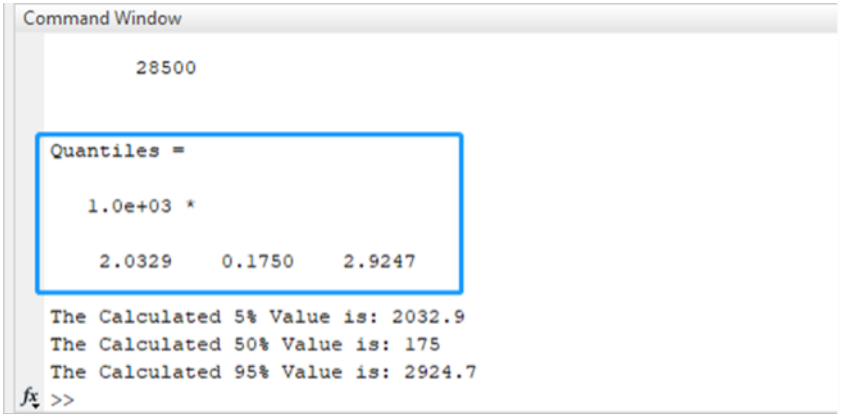

FIGURE 2.5: Output of Expert Probability Distribution Intervals Bounded by Values.

Hypothesis 2.1.: We have a Hypothesis (H_e): "the inter-quantile interval containing the true value for each variable is drawn independently from probability vector p" (Aspinall, 2008). Unlike general statistical hypotheses, this one does not require acceptance or rejection. Instead, it is used to assess the extent to which each expert's probabilities align with the hypothesis, thereby measuring the accuracy of their assessments.

A simple test for this hypothesis uses the information likelihood ratio statistic using the above Equation 2.4, and the probability value for this hypothesis is used to shape the calibration score. The equation can be expressed as follows:

$$CalibrationScore = Prob\{2N \cdot I(s(e) \mid p \geq r \mid H_e\} \tag{2.5}$$

Where, $Prob\{|\}$ denotes the probability that the information likelihood ratio statistic is greater than or equal to r, given the hypothesis is true, where r is the relevant quantity value from the expert's sample distribution. Thus, the calibration score is the probability under hypothesis H_e that a deviation at least as great as r could be observed on N realizations if H_e were true (Aspinall, 2008).

2.3.4 Information Scoring

The information score can be described as the degree how which the expert's distribution is concentrated or spread out.

For a uniform background measure, the probability density is constant between the assessed quantiles and is such that the total mass between the quantiles agrees with the probability vector p. In addition, both the uniform and log-uniform background measures require an intrinsic range on which these measures are concentrated. The CM implements the so-called "k% overshoot rule": for each item. First, the smallest interval I = [q5, q95] is determined to contain all the assessed quantiles of all experts

and contains the realization for that item. The interval is extended to a new, wider interval:

$$I^* = [q_L, q_H] \tag{2.6}$$

where,

- $q_L = q_5 - k \cdot (q_{95} - q_5)/100$
- $q_H = q_{95} + k \cdot (q_{95} - q_5)/100$

The value of k is established based on the extent to which the range is expanded, and this choice is made by the analyst. In this study, we have set k to 10 to generate a 10% overshoot. Once the intrinsic range is defined, the information score for expert e concerning assessments for N uncertain quantities can be expressed as:

$$Informationscore(e) = (1/N) \cdot \sum_{i=1}^{N} I(f_{e,i} \mid g_i) \tag{2.7}$$

Where, g_i represents the underlying probability density for variable I across the extended intrinsic range, while $f_{e,i}$ stands for the probability density function provided by expert e for item i. The relative information for all variables is aggregated and then adjusted based on the N quantities under consideration. This adjusted sum is proportionate to the relative information derived from the combined distribution of the experts in comparison to the background distribution. This is under the assumption that the variables are independent.

2.3.5 Pooling Weights

Considering the following scoring weight for expert e:

$$w_\alpha(e) = \ln\left(d_\alpha(\text{calibration score}(e) * \text{calibration score}(e) * \text{information score}(e))\right) \tag{2.8}$$

Where, $\ln d_\alpha(\cdot)$ denotes an indicator function with $\ln d_\alpha(x) = 0$, if $x < \alpha$, and $\ln d_\alpha(x) = 1$. In this case, $\ln d_\alpha(\cdot)$ is based on the expert's calibration score and only allows e to gain a non-zero weight $w_\alpha(e)$ if his score exceeds a threshold level defined by some value, α. Cooke (1991) showed that the expert's score $w_\alpha(e)$ is an asymptotically strictly proper scoring rule for average probabilities. The scoring rule constraint requires the term

$\ln d_\alpha(\text{calibration score}(e))$ to be applied to the expert's score but does not say what the value of α should be. Thus, α can be chosen to maximize the combined score of the resulting decision maker when all the experts' distributions are pooled together.

2.4 Data

2.4.1 Survey of Professional Forecasters data

This section outlines the data sources and data collection procedures employed in this research. The data comprises both primary data from the SPF surveys conducted by the Bank of England and secondary data obtained from the Bank of England and the UK's official national statistical database. The survey data, specifically, was gathered from a group of experts for whom five seed questions were designed. The variables included in the secondary data are detailed in Table 2.1.

This study introduces a novel source of survey data: the Bank of England Survey of Professional Forecasters (SPF). As noted by Cooke et al. (2021), expert-based surveys differ from simpler surveys due to a credentialing process that ensures participants meet professional standards before selection. The Bank of England collects this survey data annually, with updates provided regularly.

As described by Boero et al. (2008) and Boero et al. (2015), the Bank of England engages a panel of external professional forecasters each quarter to gather their expectations for the macroeconomic outlook over the next year. These forecasts focus on key indicators, including real GDP growth, inflation, the Bank Rate, the unemployment rate, and the Sterling Exchange Rate Index. Each panellist is recognised as a formal professional forecaster, providing their personal estimations for these five variables one year ahead.

The external experts represent diverse backgrounds, including city firms, academic institutions, and private consultancies based in London. While participants remain anonymous, they are assigned unique identification numbers, enabling their responses to be tracked over time without compromising confidentiality.

2.4.2 Variables Description

Our raw data comprises two datasets. The first dataset consists of quarterly SPF data for five key UK macroeconomic indicators: GDP growth (GDP), the CPI inflation rate (Inflation), the Bank Rate (BoE), the LFS unemployment rate (UR), and the Sterling Exchange Rate Index (ERI). This dataset spans the period from 2000Q4 to 2022Q4. The second dataset contains the corresponding actual values for these indicators over the same period.

TABLE 2.1: Overview of Variables Description

Variables	Definition	Data Source
Real GDP growth (%)	GDP growth, also called economic growth or simply "growth" – is a key measure of the economy's overall strength.	Bank of England
Inflation Rate (%)	Inflation is a measure of how much the prices of goods (such as food or televisions) and services (such as haircuts or train tickets) have gone up over time.	Office for National Statistics
Base Bank Rate (%)	Bank Rate is the most important interest rate in the UK. In the news, it's sometimes called the 'Bank of England base rate' or 'the interest rate'.	Bank of England
LFS Unemployment Rate (%)	The level and rate of UK unemployment measured by the Labour Force Survey (LFS), using the International Labour Organisation's definition of unemployment	Office for National Statistics
Sterling Exchange Rate Index (%)	The sterling exchange rate index (ERI) measures the overall change in the trade-weighted exchange value of sterling. It is designed to measure changes in the price competitiveness of traded goods and services.	Bank of England

Note: The five variables are referred to as "seed variables" in the Classical Model (CM). Experts are asked to provide their forecasts for these variables, focusing on expectations for each quarter of the next year. The specific questions posed to the experts are as follows: 1. What is next year's real GDP growth for each quarter? 2. What is next year's inflation rate for each quarter? 3. What is next year's base bank rate for each quarter? 4. What is next year's unemployment rate for each quarter? 5. What is each quarter's sterling exchange rate index for next year?

2.4.3 Preliminary Analysis

We perform an ANOVA test as a preliminary analysis to determine whether the differences between expert prediction values and actual values for each variable are statistically significant. This initial test evaluates whether the variations in predictions across different experts or variables arise from systematic differences or are merely the result of random fluctuations. By comparing the variance within groups (expert predictions) to the variance between groups (actual values), the ANOVA test provides a framework for identifying whether further, more detailed analysis is warranted. Accordingly, the null hypothesis (H_0) and alternative hypothesis (H_1) are defined as follows:

H_0: There is no difference between each expert prediction and true value.

H_1: There is an existing difference between each expert prediction and true value.

The results of the ANOVA tests, presented in Tables 2.2 to 2.6, indicate statistically significant differences between the prediction values provided by the ten experts and the

actual values for key macroeconomic indicators, including GDP growth, the inflation rate, the unemployment rate, the bank rate, and the Sterling Exchange Rate Index (ERI). These findings provide an initial statistical assessment, highlighting measurable discrepancies between forecasts and actual outcomes. Moreover, the observed differences exhibit systematic patterns, suggesting the need for further investigation to understand the underlying causes and their implications.

TABLE 2.2: Analysis of variance test for GDP growth

Real - Expert	**Obs.**	**Mean1**	**Mean2**	**Dif.**	**St Err**	**t value**	**p value**
Real - B1	62	.024	1.903	-1.879	.36	-5.2	0
Real - G1	70	.424	2.264	-1.84	.413	-4.45	0
Real - I1	57	.702	1.945	-1.243	.321	-3.85	.001
Real - L1	64	.575	2.833	-2.258	.319	7.1	0
Real - N1	75	.49	2.374	-1.883	.398	-4.75	0
Real - O1	69	.339	2.111	-1.772	.422	-4.2	0
Real - S1	72	.366	2.068	-1.701	.104	-16.25	0
Real - T1	74	.486	2.045	-1.558	.447	-3.5	0.001
Real - X1	70	.668	2.093	-1.424	.271	-5.25	0
Real - B2	73	.688	1.872	-1.185	.29	-4.1	0

Note: Summary of ANOVA results for the comparison between the forecasters' value sand the true values in GDP growth.

TABLE 2.3: Analysis of variance test for Inflation rate

Real - Expert	**Obs.**	**Mean1**	**Mean2**	**Dif.**	**St Err**	**t value**	**p value**
Real - B1	62	1.895	1.957	-.061	.126	-.5	.627
Real - G1	49	2.161	2.265	-.104	.136	-.75	.449
Real - I1	57	1.958	2.085	-.127	.136	-.95	.352
Real - L1	64	2.011	2.057	-.045	.135	-.35	.737
Real - N1	75	1.895	1.814	.08	.111	.75	.471
Real - O1	69	1.95	1.965	-.016	.129	-.1	.902
Real - S1	72	1.952	2.079	-.128	.115	-1.1	.271
Real - T1	75	1.911	2.141	-.23	.118	-1.95	.053
Real - X1	72	1.935	2.136	-.201	.106	-1.9	.064
Real - B2	73	1.998	2.055	-.058	.114	-0.5	.614

Note: Summary of ANOVA results for the comparison between the forecasters' value and the true values in Inflation rate.

2.4.4 Data Augmentation

As indicated by Little et al. (1995), the social and behavioural sciences frequently suffer from missing data. For instance, sample surveys often have some individuals who either refuse to participate or do not supply answers to certain questions, and panel studies often have incomplete data due to attrition. Brick and Kalton (1996) also mentioned missing data problems occur in survey data-based research because an element

TABLE 2.4: Analysis of variance test for Base Bank rate

Real - Expert	Obs.	Mean1	Mean2	Dif.	St Err	t value	p value
Real - B1	62	2.038	2.386	-.348	.116	-3	.004
Real - G1	70	2.196	2.614	-.418	.118	-3.55	.001
Real - I1	55	2.303	2.738	-.435	.14	-3.1	.003
Real - L1	62	2.194	2.558	-.364	.08	-4.55	0
Real - N1	75	1.957	2.271	-.314	.098	-3.2	.002
Real - O1	69	2.21	2.708	-.499	.118	-4.25	0
Real - S1	58	2.671	3.191	-.52	.146	-3.55	.001
Real - T1	75	2.056	2.631	-.575	.13	-4.45	0
Real - X1	72	2.308	2.744	-.436	.12	-3.6	.001
Real - B2	72	2.276	2.585	-.308	.095	-3.25	.002

Note: Summary of ANOVA results for the comparison between the forecasters' value and the true values in Bank rate.

TABLE 2.5: Analysis of variance test for Unemployment rate

Real - Expert	Obs.	Mean1	Mean2	Dif.	St Err	t value	p value
Real - B1	21	4.585	5.229	-.643	.087	-7.35	0
Real - G1	22	4.686	5.509	-.823	.189	-4.35	.001
Real - I1	15	4.854	5.388	-.534	.138	-3.85	.002
Real - L1	15	4.787	5.407	-.62	.147	-4.2	.001
Real - N1	28	4.572	5.000	-.428	.137	-3.1	.005
Real - O1	16	4.694	5.150	-.456	.201	-2.25	.038
Real - S1	20	4.565	5.305	-.74	.114	-6.5	0
Real - T1	23	4.635	5.411	-.776	.164	-4.75	0
Real - X1	19	4.679	5.405	-.726	.12	-6	0
Real - B2	20	4.685	5.280	-.595	.144	-4.15	.001

Note: Summary of ANOVA results for the comparison between the forecasters' value and the true values in Unemployment rate.

TABLE 2.6: Analysis of variance test for ERI

Real - Expert	Obs.	Mean1	Mean2	Dif.	St Err	t value	p value
Real - B1	62	2.038	2.386	-.348	.116	-3	.004
Real - G1	70	2.196	2.614	-.418	.118	-3.55	.001
Real - I1	55	2.303	2.738	-.435	.14	-3.1	.003
Real - L1	62	2.194	2.558	-.364	.08	-4.55	0
Real - N1	75	1.957	2.271	-.314	.098	-3.2	.002
Real - O1	69	2.21	2.708	-.499	.118	-4.25	0
Real - S1	58	2.671	3.191	-.52	.146	-3.55	.001
Real - T1	75	2.056	2.631	-.575	.13	-4.45	0
Real - X1	72	2.308	2.744	-.436	.12	-3.6	.001
Real - B2	72	2.276	2.585	-.308	.095	-3.25	.002

Note: Summary of ANOVA results for the comparison between the forecasters' value and the true values in ERI.

in the target population is not included in the survey's sampling frame (noncoverage), because a sampled element does not participate in the survey (total nonresponse) and because a responding sampled element fails to provide acceptable responses to one or more of the survey items (item nonresponse).

Andridge and Little (2010) proposed that missing data are often a problem in large-scale surveys, arising when a sampled unit does not respond to the entire survey (unit non-response) or a particular question (item non-response). Van Buuren (2018) indicated missing data poses challenges to real-life data analysis. Its occurrence of missing data can cause serious issues, including decreased sample size, biased estimates, and algorithmic problems. Therefore, properly treating missing data is a significant part of statistical data analysis. Considering the possible issue that missing data may cause, we learn the technique of data augmentation from (Little and Rubin, 2002) to deal with.

In this study, we also address a common limitation of survey-data-based research. We review the many ways to perform data imputations to address this concern. Zhang (2016) emphasised that, like complete case analysis, imputations with mean, median, and mode are simple but can introduce bias on mean and deviation. Because they ignore relationships with other variables, regression imputation can preserve the relationship between missing values and other variables. Learned from Little and Rubin (2002) stated that when sample sizes are small, a useful alternative to ML is adding a prior distribution for the parameters and computing the posterior distribution of the parameters of interest.

The posterior distribution for a model with an ignorable missing data mechanism is:

$$p(\theta \mid Y_{\text{obs}}, M) \equiv p(\theta \mid Y_{\text{obs}}) = \text{constant} \times p(\theta) \times f(Y_{\text{obs}} \mid \theta) \tag{2.9}$$

Where $p(\theta)$ is the prior distribution, and $f(Y_{\text{obs}} \mid \theta)$ is the density of the observed data. The likelihood was factored into complete data components.

$$L(\varnothing \mid Y_{\text{obs}}) = \prod_{q=1}^{Q} L_q(\varnothing_q \mid Y_{\text{obs}}) \tag{2.10}$$

Assuming the parameters $\varnothing_1, \ldots, \varnothing_Q$ were also a prior independent, the posterior distribution factored in an analogous way, with $\varnothing_1, \ldots, \varnothing_Q$ posterior independent. A draw, $\varnothing^{(d)} = (\varnothing_1^{(d)}, \ldots, \varnothing_Q^{(d)})$ could be obtained directly from the factored complete-data posterior distribution. Draws of the θ were then obtained as $\varnothing^{(d)} = \theta(\varnothing^{(d)})$, where $\theta(\varnothing)$ is the inverse distribution from $\varnothing$ to θ.

2.5 Main Results and Findings

2.5.1 Calibration Score

This section presents the calibration scores of ten experts across five seed variables. To compute these scores, ten experts were given five seed questions. Each expert provided distributional belief for the following target variables: GDP growth, Inflation, Bank rate, Unemployment, and ERI. The realization values for each expert's predictions were aggregated across these five variables, as illustrated in Figure 2.6. It is evident that for Expert B1, four true realization values fall between the 5th and 95th quantiles, while one realization value is below the 5th quantile. Similar patterns are observed for Experts G1, L1, N1, T1, and X1. In contrast, for Experts I1, O1, and S1, one realization value lies below the 5th quantile, three fall between the 5th and 50th quantiles, and one is between the 50th and 95th quantiles. Expert B2's five true realization values are between the 5th and 50th quantiles. Table 2.7 summarises the overall performance, detailing the proportion of realization values observed in each of the four intervals.

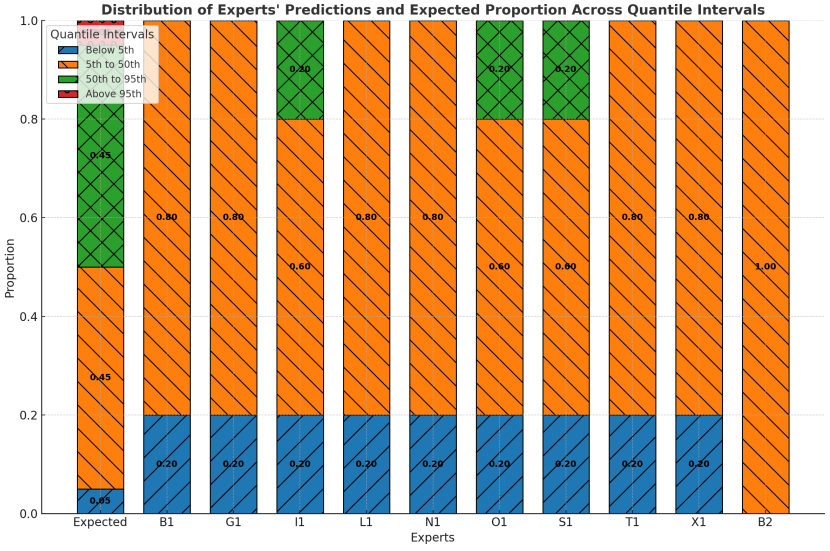

FIGURE 2.6: Distribution of the belief of ten experts on five seed questions.

Table 2.7 summarises the observed proportions of realisations across the specified quantile intervals for each expert. The expected proportions for these intervals are 0.05 for both tails (below the 5th percentile and above the 95th percentile) and 0.45 for the central intervals (5th to 50th percentile and 50th to 95th percentile). The results indicate that none of the experts displayed overconfidence, as the majority of realisations are concentrated within the central quantile intervals. Notably, none of the experts' realisations fell into the distribution's extreme tails.

TABLE 2.7: Comparison of Experts' Realisations Distributions and Expected Proportions Across Quantile Intervals

Observations	Quantile Intervals			
	Below 5th	5th to 50th	50th to 95th	Above 95th
Expected Proportion	0.05	0.45	0.45	0.05
Expert B1	0.2	0.8	0	0
Expert G1	0.2	0.8	0	0
Expert I1	0.2	0.6	0.2	0
Expert L1	0.2	0.8	0	0
Expert N1	0.2	0.8	0	0
Expert O1	0.2	0.6	0.2	0
Expert S1	0.2	0.6	0.2	0
Expert T1	0.2	0.8	0	0
Expert X1	0.2	0.8	0	0
Expert B2	0	1	0	0

In this analysis, the expected standard is set at 0.5 (10% of the total number of seed questions). To further quantify the extremity of the realisations relative to the experts' specified distributions, the Kullback-Leibler (KL) divergence measure is employed. This measure evaluates the divergence between two probability distributions and is used here to compare the distribution specified by each expert with the empirical distribution derived from the raw frequencies. Table 2.7 illustrates an example by comparing the observed and expected frequencies, highlighting the disparities between the two distributions.

TABLE 2.8: KL Divergence for Experts

Experts	**KL Divergence**
B1	0.73
G1	0.73
I1	0.28
L1	0.73
N1	0.73
O1	0.28
S1	0.28
T1	0.73
X1	0.73
B2	0.79

The formula for the divergence measure, denoted by I(s,p) is:

$$I(s,p) = \sum_{i=1}^{n} S_i \cdot \ln\left(\frac{S_i}{P_i}\right) \tag{2.11}$$

Where,

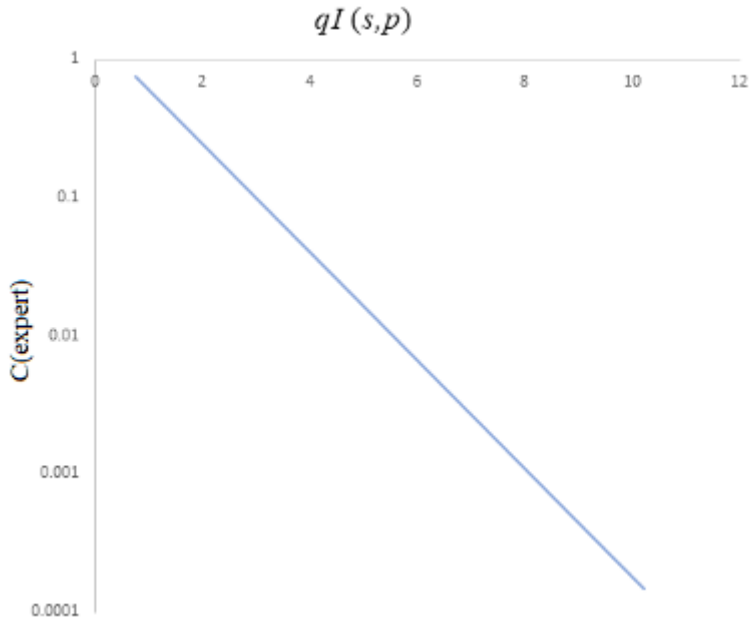

FIGURE 2.7: An example demonstrating the relationship between an expert's Calibration Score, $C(\text{expert})$, and $qI(s,p)$ where q represents the number of seed questions and $I(s,p)$ denotes the KL divergence) illustrates a sharp decline in the score as the divergence moves away from 0, displayed on a logarithmic scale, as shown by (Dias et al., 2018).

- S_i is the observed proportion of realizations in interval i
- P_i is the expected proportion of realizations in interval i
- n is the number of intervals

In the KL divergence measure, if the observed proportions perfectly match the expected proportions, the divergence measure equals 0. As the difference between the observed and expected proportions increases, the divergence value also increases. From Table 2.8, it is evident that no expert perfectly matches the expected proportions. However, Experts I1, O1, and S1 exhibit relatively smaller divergence values compared to the other experts, indicating that their probability distributions are closer to the expected distributions. In the following section, we will examine all experts' calibration and information scores. The calibration score used within Cooke's Classical Model (CM) represents the probability of observing a more extreme divergence statistic than that observed between specified and actual proportions. An ideal expert would achieve a calibration score of 1, while a score of 0 would indicate the worst performance.

Fig. 2.7 provides an example illustrating the relationship between the calibration score of an expert, denoted as $C(\text{expert})$, and $qI(s,p)$, where q represents the count of seed questions and $I(s,p)$ denotes the KL divergence. Notably, the graph is presented on a logarithmic scale. Starting from a maximum score of 1 when the divergence is 0,

there is a rapid decline in the score as the divergence value increases. This logarithmic representation of the calibration score demonstrates an almost linear relationship with $I(s,p)$. Across the range of $qI(s,p) \in [1,11]$, there exists an approximate relationship denoted as $C(\text{expert}) \approx 1.44 \cdot e^{-0.9qI(s,p)}$. It's important to note that this expression is a rough indicator of the relationship between these variables. It suggests that the logarithm of the ratio of any two calibration scores is roughly $0.9q$ times the difference in their KL divergence measures.

2.5.2 Information Score

An expert can create the illusion of achieving exceptional, or even flawless, calibration by employing excessively large quantile intervals. However, this approach ultimately offers limited informational value. The ideal expert demonstrates two key attributes: being well-calibrated and providing valuable information. While various conventional methods exist for gauging the degree of dispersion within a probability distribution—such as measuring the standard deviation or the width of prediction intervals—these methods have limitations, particularly when dealing with variables measured in different units (e.g., transitioning from grams to kilograms may affect certain variables disproportionately). In this context, the Calibration Model (CM) utilises the Kullback-Leibler (KL) divergence measure due to its scale-invariant properties.

To assess the extent of dispersion in the experts' probability distributions, a reference range is established in relation to a background context. During the elicitation process, the expert does not specify exact minimum or maximum values. Consequently, it becomes necessary to determine the lengths of the lower and upper intervals. This is achieved using the intrinsic range, which is based on the range of judgements provided by all experts for a given variable, whether it is a target or a seed.

An intrinsic range is computed for each question, both seed and target. By default, this intrinsic range extends 10% beyond the span of the lowest and highest assessed values. (It should be noted that the extent of this extension is determined by the analyst and impacts only the information score; a larger extension tends to make all information scores more uniform, while a smaller extension accentuates differences.)

To measure the informativeness of an expert's probability distribution, the KL divergence measure is applied relative to a uniform distribution mapped onto the intrinsic range. This uniform distribution represents the least informative scenario across the entire collected range.

2.5.3 Calibration Scores, Information Scores and Weights

From Table 2.9, the performance of the experts can be categorised into two distinct groups based on their calibration and information scores. Experts B1, G1, L1, N1, T1, and X1 demonstrate low calibration scores (approximately 0.0521) and moderate information scores ranging between 0.0398 and 0.0651. These values indicate consistent but modest performance in both calibration and information metrics, reflected in their relatively stable average weights, which are clustered around 0.0034.

In contrast, Experts I1, O1, and S1 exhibit significantly higher calibration scores (approximately 0.3950), suggesting greater accuracy in aligning their predictions with observed data. Their information scores are also notably higher, with I1 achieving the highest information score of 0.0910 among all experts. This indicates that these experts contribute substantially to the overall information content of the forecasts, which is further supported by their higher average weights, particularly for Expert I1 (0.1050), indicating a stronger influence in the aggregation process.

Expert B2, however, stands out for having the lowest average weight (0.0017). This is attributable to both its low calibration score (0.0396) and a modest information score (0.0441), suggesting a limited contribution to the aggregated decision-making process.

Overall, the results highlight a clear distinction between the consistently moderate performance of one group of experts (B1, G1, L1, N1, T1, X1) and the higher variability but greater informational contributions of another group (I1, O1, S1).

TABLE 2.9: Performance metric score of each expert

Experts	Calibration Scores	Information Scores	Average Weights
B1	0.0521	0.0552	0.0029
G1	0.0521	0.0651	0.0034
I1	0.3950	0.0910	0.1050
L1	0.0521	0.0434	0.0023
N1	0.0521	0.0653	0.0034
O1	0.3950	0.0314	0.0124
S1	0.3950	0.0721	0.0285
T1	0.0521	0.0398	0.0021
X1	0.0521	0.0646	0.0034
B2	0.0396	0.0441	0.0017

Notes: The average weights are calculated by averaging each expert's information score across all seed questions, following the method by Dias et al. (2018).

Table 2.10 presents the relative rankings of calibration scores, information scores, and resulting weights, ordered from highest to lowest. Interestingly, experts with exceptionally high performance-based scores, such as Expert I1 and Expert S1 (the top two highest-scoring experts), demonstrate consistency across statistical accuracy and

knowledge informativeness. Their calibration scores, information scores, and resulting weights are all notably high, reflecting strong overall performance.

Conversely, experts such as Expert T1 and Expert B2, whose weight scores are very low, consistently perform poorly in both calibration and information metrics, indicating limited contributions to the aggregation process. However, for experts whose scores fall within the middle range, a noticeable inconsistency emerges between statistical accuracy and informativeness. Notably, the weight scores tend to align more closely with variations in the calibration scores rather than the information scores, suggesting a stronger emphasis on calibration in the weighting process.

TABLE 2.10: Calibration, Information, and Average Weights Scores

Calibration scores	Information scores	Average Weights
Expert I1	Expert I1	Expert I1
Expert S1	Expert S1	Expert S1
Expert O1	Expert N1	Expert O1
Expert N1	Expert G1	Expert G1
Expert G1	Expert X1	Expert X1
Expert X1	Expert L1	Expert N1
Expert B1	Expert B1	Expert B1
Expert L1	Expert B2	Expert L1
Expert T1	Expert T1	Expert T1
Expert B2	Expert O1	Expert B2

2.5.4 Normalised Weights

TABLE 2.11: Normalised Weights for Experts

Experts	**Normalised Weights**
I1	0.6360
S1	0.1726
O1	0.0751
G1	0.0206
N1	0.0206
X1	0.0206
B1	0.0176
L1	0.0139
T1	0.0127
B2	0.0103

Note: The rank is described as the expert's normalised weight in descending order.

Table 2.11 presents the normalised weights of the experts, re-ranked in descending order. Expert I1 is assigned the highest weight (0.6360), signifying the highest level of expertise among the panel of experts. This substantial weight indicates exceptional

performance in terms of both calibration and the provision of valuable information within Cooke's Classical Model (CM).

In contrast, the weights assigned to the other experts are considerably lower, with the second-highest weight (Expert S1) being 0.1726—significantly smaller than that of Expert I1. This notable disparity suggests that the remaining experts are perceived as less well-calibrated and less informative. The findings highlight the dominance of Expert I1 in the weighting process, which underscores their critical contribution to the aggregated results.

2.5.5 Aggregated Expert Prediction

From Figs. 2.8 to 2.12, we compare the expert predictions, aggregated with their assigned weights, to the true value data series. Overall, the aggregated expert predictions demonstrate a notable performance improvement compared to individual forecasts.

For GDP growth, the aggregated expert predictions are generally higher than the true values, indicating a systematic overestimation by experts for the one-year-ahead horizon. This suggests a potential optimism bias in their forecasting process. Additionally, we observe a consistent one-year lag in the expert predictions compared to the actual values, highlighting a tendency for experts to rely on past trends rather than current economic signals. This reliance on historical data may limit the responsiveness of their predictions to recent changes in the economic environment.

Overestimation and lagging trends are particularly pronounced during periods of economic downturns, such as the 2008 financial crisis and the 2019 COVID-19 pandemic. During these periods, experts' predictions not only overestimated the speed of economic recovery but also fail to capture the immediate impacts of the downturns. This behaviour suggests that experts might be anchored to pre-crisis economic conditions, leading to a slower expectation adjustment.

In the comparison of Inflation forecasts, we have identified another noteworthy pattern. During two specific periods, namely the 2008 economic crisis and the 2019 pandemic, experts' inflation expectations were consistently higher than the actual inflation rates. This discrepancy suggests that experts tend to overestimate inflation during periods of significant economic disruption. However, in the later stages of these two periods, the experts' expectations exhibited a clear downward trend, aligning more closely with actual inflation rates. This shift indicates a correction in their forecasts as the economic situation stabilised.

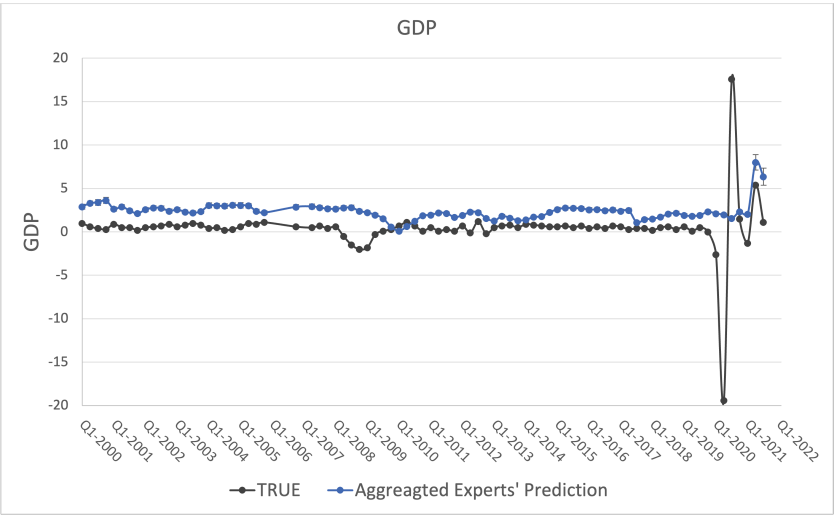

FIGURE 2.8: Comparison of true GDP value with aggregated expert expectation value

This pattern reveals a broader trend in expert behaviour: during periods of economic stability, experts tend to adopt a cautious or even pessimistic stance on inflation, possibly reflecting a conservative approach to forecasting in the absence of significant economic pressures. Conversely, during periods of economic turbulence, experts display a more optimistic outlook, often predicting higher inflation rates than those that eventually materialise. This optimistic bias in uncertain times may arise from expectations of increased economic activity or policy interventions designed to stimulate the economy, which experts anticipate will drive inflation upwards.

At the Bank rate, we find that expert predictions are very close to the actual values. However, we have identified distinctive features. Firstly, in 2008, we noticed a lagging trend in expert predictions. This lag can be attributed to the unprecedented nature of the global financial crisis, which likely caught many experts by surprise. During this period, the rapid economic downturn and significant policy interventions by central banks, including drastic cuts in interest rates, led to delays in adjusting forecasts to reflect the rapidly changing economic environment.

Furthermore, after 2008, the overall stance of experts towards bank rate predictions shifted from being slightly below the actual values to an overall overestimation. This shift suggests a recalibration of expert expectations post-crisis, as central banks maintained lower interest rates for an extended period to support economic recovery. Experts may have overcorrected their initial conservatism, leading to overestimation. Understanding these prediction patterns is crucial for both policymakers and market participants. Recognising the tendency of experts to lag during crises and overestimate rates, post-crisis can inform communication strategies and policy decisions at the Bank of England while also enhancing investment strategies and risk management practices for market participants.

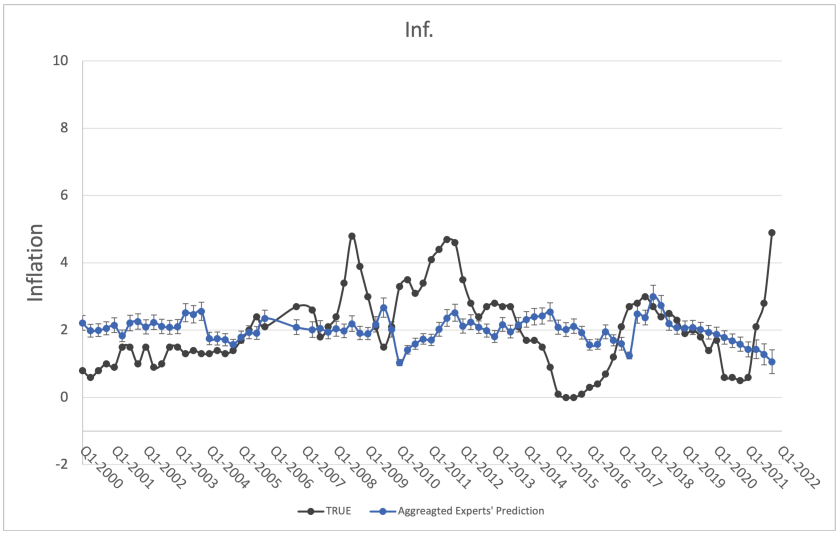

FIGURE 2.9: Comparison of true Inflation rate value with aggregated expert expectation value.

In addition, the expert predictions have exhibited a systematic overestimation regarding the Unemployment rate, with only a brief underestimation observed in 2020-2021. This suggests that the actual employment situation has been consistently better than what the experts predicted, reflecting a long-term conservative stance of the experts towards the job market and the economy. This conservative outlook implies that experts tend to underestimate the resilience and adaptability of the labour market during economic fluctuations. One potential reason for this conservative stance could be the inherent uncertainty and risk aversion that experts exhibit when forecasting economic indicators. Experts might be factoring in worst-case scenarios and potential economic shocks, leading to more cautious predictions.

Furthermore, the overestimation of the Unemployment rate could be influenced by historical contexts and the memory of past economic downturns, such as the 2008 financial crisis. These events may have instilled a sense of caution and a tendency to predict higher unemployment rates as a protective measure against unexpected negative developments. The brief period of underestimation in 2020-2021, during the COVID-19 pandemic, is also notable. This could be attributed to the unprecedented nature of the pandemic and the initial underestimation of the swift policy responses and fiscal measures taken by governments worldwide to mitigate the economic impact. The rapid deployment of stimulus packages, support for businesses, and unemployment benefits likely contributed to a better-than-expected employment situation, which experts failed to fully anticipate.

In predicting the Sterling Exchange Rate Index, experts exhibit a notable tendency for overestimation following the two critical periods mentioned: the 2008 global financial crisis and the 2019 pandemic. This pattern indicates that experts maintain a slightly optimistic outlook on the exchange rate market, especially after significant economic

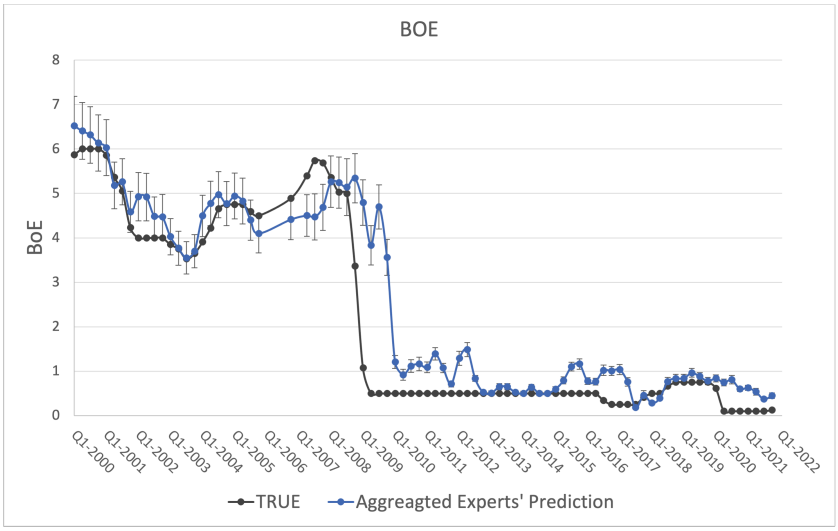

FIGURE 2.10: Comparison of true Bank rate value with aggregated expert expectation value.

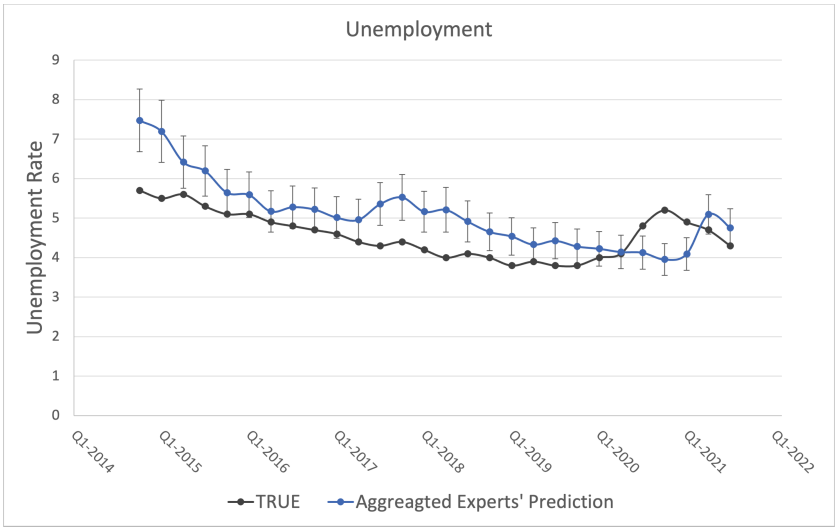

FIGURE 2.11: Comparison of true unemployment rate value with aggregated expert expectation value.

disruptions. The observed optimism likely stems from expecting a strong recovery and stabilization in the exchange rate following economic turmoil. This tendency underscores the need for caution in interpreting expert forecasts, as their predictions might not fully account for ongoing vulnerabilities and uncertainties in the exchange rate market. It also highlights the importance of incorporating a range of scenarios and more robust modelling techniques to mitigate the impact of overly optimistic forecasts on policy and financial decisions.

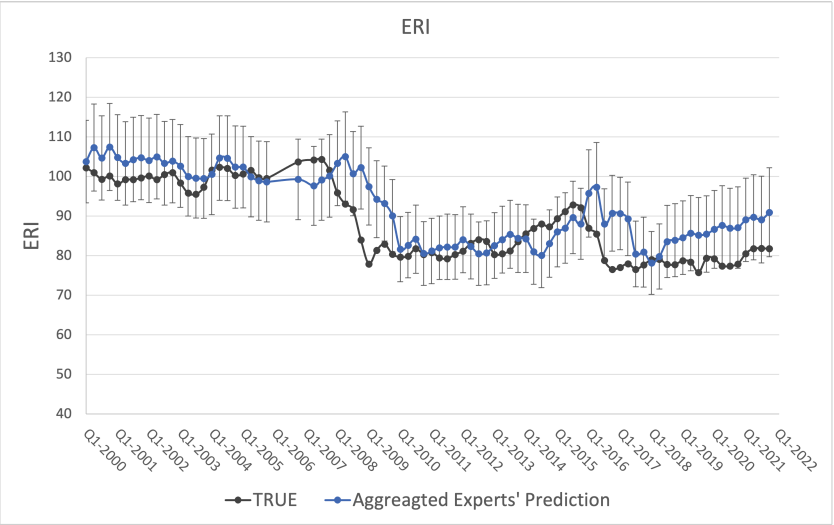

FIGURE 2.12: Comparison of true sterling exchange rate value with aggregated expert expectation value.

2.6 Conclusion

To summarize, this study begins with two fundamental questions: (i) How accurate and reliable are the predictions made by the Bank of England's External Forecasters regarding the target variables, namely GDP growth, CPI inflation rate, Bank rate, LFS unemployment rate, and Sterling ERI ? (ii) What characteristics can be discerned from expert judgement? These questions are crucial for understanding the presence of uncertainty in expert judgement.

Furthermore, we present an analytical framework for scoring the calibration and informativeness provided by each expert. The normalised weights are derived from these experts' calibration and information scores. The results indicate the presence of centralised features that are less well-calibrated and uninformative among these experts. Existing literature has consistently shown empirical evidence that experts are often less efficiently calibrated, and while some may be well-calibrated, they may possess limited knowledge (Boero et al., 2015).

Therefore, we form a new expectation by aggregating the judgements of a panel of experts to address the limitations of each individual expert. This aggregated approach leads to a significant improvement in prediction accuracy. However, we also find that experts' predictions exhibit a systematic bias toward the values of the previous year. This indicates that experts rely heavily on historical data when providing economic expectations for the year ahead, potentially overlooking changes in the current economic market environment.

Additionally, we have identified that experts tend to be overconfident in macroeconomic growth during systemic shocks, such as the global financial crisis and the 2019

pandemic. They maintain an overly optimistic outlook on the speed of economic recovery, potentially introducing another bias into their attitudes. This overconfidence can skew predictions, making them less reliable during periods of economic instability.

In conclusion, while the aggregation of expert judgements improves overall prediction accuracy, the inherent biases in expert predictions, such as over-reliance on historical data and overconfidence during systemic shocks, must be addressed. Future work should focus on developing methods to mitigate these biases and further enhance the reliability of aggregated forecasts. This study contributes to a better understanding the complexities involved in expert judgement and economic forecasting, providing a foundation for more robust predictive models.

Chapter 3

Measuring the Attitude Divergence in Expert Prediction under Bounded Rationality

3.1 Introduction

In the real world, making a straightforward accept or reject decision in the decision-making process can be challenging due to the presence of incomplete and inaccurate information. This complexity is particularly pronounced in risk-related models, where objective and subjective factors both play a role. In such decision-making contexts, it becomes crucial not only to consider the objectively available risk information but also to factor in the subjective judgment and initiative of the decision-makers themselves. Additionally, as early as the 1970s, Tversky and Kahneman (1974) introduced three heuristics that are used in making judgements under uncertainty. They also emphasized the issue of identifying human inadequacies in assessing probabilities. While these heuristics are highly efficient and often effective, it is essential to note that they can also lead to systematic and predictable errors.

However, despite the significance of heuristics and biases in influencing the accuracy and quality of the judgement and decision-making process, it has not received sufficient attention. As indicated by Kynn (2008), biases in the heuristics process have largely gone unnoticed, even though the heuristic method itself has been a definitive groundbreaking research in probability assessment. He also highlights that bias in cognitive models has been nearly completely overlooked by the statistical literature on expert elicitation. Moreover, Montavon et al. (2018) also expressed concern that the study of cognitive biases in judgement and decision-making is not as extensive as it should be. They also point up that the reasons behind these cognitive biases, whether conscious

or subconscious, stem from various factors such as self-interest, social pressures, and organizational context.

Like any other forecasters, professional forecasters inherently follow a 'strategic' behaviour. These forecasters possess 'bounded rationality' and often work with incomplete information. This incompleteness arises partly because they lack full knowledge of the strategies employed by other forecasters and partly due to unexpected policy changes and external shocks. However, a notable distinction is that these forecasters are among the most well-informed participants in the economic market. Their decision-making processes are always influenced by a multitude of factors, both known and measurable, as well as those that remain unknown and immeasurable. Consequently, it is often reasonable to assume that their forecasts are driven by strategic considerations at various points in the forecasting process.

Existing theories that characterise the behaviour of professional forecasters have been developed based on the implicit assumption that the economy is "stable and potentially predictable". Even if there is some degree of randomness, this randomness is not explicitly modelled but is indirectly concealed within the strategies they employ. This raises a profound question: regardless of their knowledge of the economic environment, an agent will behave differently depending on whether they are confronted with ongoing uncertainty or an environment characterised by average recovery uncertainty. From a psychological theory perspective, we already know that rational forecasters may behave differently in an "uncertain" environment than in stable environments. A common example is "herding behaviour". Assuming that uncertainty leads to behavioural differences, it can also be hypothesised that the next best strategy that expert forecasters adopt under uncertainty will differ from those in a relatively uncertain environment.

Forecasting the economy has always been a significant and complex task that relies heavily on subjective professional forecaster's judgement and knowledge in addition to analysing historical data. Most forecasting judgements are made without advance knowledge of their consequences. Thus, it is imperative to capture this expert knowledge quantitatively. Although the initial focus of the expert study is the accuracy of their judgements, here, we called it explicit traits. Based on this aim, but not only, this study is going to deeply explore the tacit characteristic behind expert behaviour. Hess and Orbe (2013) found that macroeconomic survey forecasts are anchoring biased and, therefore, inefficient. However, despite highly significant test coefficients, a bias adjustment does not improve the forecasts' quality. We find that cognitive bias is a statistical artefact because the anchoring test is biased in itself.

Combining these concerns has prompted us to explore potential biases to contribute to the literature in this area. This study will first provide a new perspective for examining the attitudinal differences among experts in macroeconomic forecasting while

leaving other forms of bias for our future research. As denoted by with Tversky and Kahneman (1974), a better understanding of heuristics and their resulting biases can enhance judgement and decision-making under uncertainty. This research is grounded in both psychology and cognitive theory, and our specific aim is to reverse-reasoning expert uncertainty attitudes based on their prediction behaviours in the judgement and decision-making process.

On the definition of differences in expert attitudes, this study relies on the classification by Brito et al. (2008), which suggested that experts can be categorised into two main groups based on their emotional characteristics: optimistic experts and pessimistic experts. This classification approach is based on the width of the "S" shape of the expert's cumulative probability distribution. Additionally, we also learn from (Huang et al., 2022; Engelberg et al., 2009) that to classify professor forecasters into optimists and pessimists by using the measure of central tendency.

This paper is organised into five sections. The first section introduces the background, motivation, and objective of this study. The second section outlines the previous findings on behaviour bias in expert judgement. The third section describes the approach used in modelling the classification of experts' different attitudes and the data. The fourth section contains the data description. The fifth part presents the main results and discussion. It ends with a brief conclusion with future work.

3.2 Literature Review

3.2.1 Judgemental Bias in Cognitive Theory

Following the ground-breaking findings of Tversky and Kahneman (1974) set in the notion of the heuristics and biases, the underlying principle of which is that people's judgements are often made based on heuristics, which are quick, short-cut reasoning processes. Daniel (2017) used the terms "system 1" and "system 2" in the context of intuitive judgement to illustrate the concepts of rapid and deliberate thinking. System 1 functions automatically and swiftly, requiring minimal effort and lacking a sense of voluntary control. On the other hand, System 2 directs attention toward mentally demanding tasks, such as intricate calculations. The activities of System 2 are frequently linked to the subjective feelings of agency, choice, and focused concentration.

Here, we cross-cite the summary of Rezaei (2021) and highlight that decisions are typically made through decision-maker's assessments. These individuals form judgements by comparing various options across multiple criteria. Often, decision-makers lack access to or choose not to use objective data, relying instead on their personal evaluations. Behavioural psychologists have observed that people simplify this complex task by employing specific shortcuts or heuristics (Gilovich et al., 2002). Various heuristics

are employed by individuals (Gigerenzer and Gaissmaier, 2011; Gilovich et al., 2002), and, in many instances, people are unaware of these heuristics' roles in their decision-making processes, and they cannot consciously control them. Nevertheless, individuals can potentially recognize and rectify resulting biases (Tversky and Kahneman, 1974). While, in general, heuristics are immensely useful for decision-making, there are occasions when they lead to significant errors that can incur substantial costs (Arkes, 1991). These errors are known as cognitive biases and result in skewed decision-making.

While cognitive biases have been extensively explored in fields such as psychology (Gigerenzer, 1991; Hilbert, 2012; West et al., 2008), marketing (Fisher and Statman, 2000; Thompson et al., 2011), healthcare (Phillips-Wren et al., 2019), organizational studies (Das and Teng, 2001; Schwenk, 1984; Tetlock, 2000), business intelligence (Ni et al., 2019), and political science (Arceneaux, 2012; Rouhana et al., 1997), it is surprising that, as also noted by (Montibeller and Von Winterfeldt, 2015), we have come across only a limited number of studies in the realm of multi-attribute decision-making, most of which remain theoretical in nature.

3.2.2 Expert Errors and Bias

Counter to the common belief that expert knowledge and expertise can reflect superior abilities and capacities, recent studies find an ongoing debate on it. The initial focus of psychological investigations into experts centred on assessing the psychometric properties, particularly validity, as discussed by (Bower and Cohen, 2014). For instance, in a study conducted by Trumbo et al. (1962). observed that expert grain assessors often exhibited both invalidity and unreliability. They discovered that nearly one-third of wheat samples were incorrectly graded, and upon re-evaluation, more than one-third of these samples received different grades. Interestingly, Trumbo et al. (1962) noted that while greater experience heightened the judges' confidence, it did not necessarily correlate with the accuracy of their grain assessments. Notably, the experts were unaware of these various shortcomings.

Similar findings have been observed among other experts, including those in the medical field. Previous research on expert decision-makers (SHANTEAU, 1988; Einhorn, 1974; Goldberg, 1959) consistently suggests that, due to cognitive constraints, experts tend to be generally inaccurate, unreliable, susceptible to biases, lack self-awareness, and show limited improvement with increasing experience. Moreover, Baddeley et al. (2004) revealed that when making probabilistic judgements, most individuals tend to commit common errors in their assessments of probabilities, as demonstrated in the work of (Burnham et al., 1998). Nevertheless, experts are not immune to these biases, individually and when considering group-level biases. These biases persist due to cognitive limitations inherent in the human mind's processing capabilities, a notion supported by the works of (Anderson, 2009; Tversky and Kahneman, 1974). Further, Dror

and Charlton (2006) pointed out that expert performance and accuracy are still important issues in almost all specialised domains.

It has greatly discussed how judgemental bias can affect expert judgemental accuracy. Overall, previous studies have painted a rather bleak picture of the decision-making abilities of experts. Dror and Charlton (2006) pointed out that being an expert does not necessarily mean error-free performance and argued that errors are made in almost every professional field. Unfortunately, Daniel (2017) also indicated that professionals' intuitions do not all arise from true expertise.

3.2.3 Overconfidence Bias

There is a growing demand for expert opinions when evaluating expert performance and uncertainties. However, a limitation arises when experts assess single events due to the influence of heuristics, as demonstrated by heuristics (Ursacki and Vertinsky, 1992; Tversky and Kahneman, 1974). These heuristics have been shown to introduce systematic and predictable biases, as highlighted by (Finucane et al., 2000). The primary heuristics are detailed in Table 3.1.

TABLE 3.1: Heuristics/Biases and Related Studies

Heuristics/Biases	Definition	Related Studies
Representativeness heuristic	Judgement is made with an over-reliance on certain characteristics, neglecting others.	(Taffler, 2010; Grether, 1992, 1980)
Availability heuristics	The probability of events are often judged based on how easily they can be imagined or recalled.	(Keller et al., 2006; Folkes, 1988; Tversky and Kahneman, 1973)
Anchoring and Adjustment	Individuals generally struggle to make accurate judgements starting from a given point and then making related adjustments.	(Epley and Gilovich, 2006; Northcraft and Neale, 1987)
Overconfidence Bias	Individuals often overestimate their abilities, knowledge, and the accuracy of their beliefs and predictions.	(Berthet, 2021; Moore and Schatz, 2017; Baker and Nofsinger, 2002)
Confirmation Bias	People tend to seek out and place more weight on information that confirms their pre-existing beliefs, while ignoring contradictory information..	(Peters, 2022; Kelly and Sharot, 2021; Kappes et al., 2020)

Overconfidence remains a prevalent issue in human judgement and decision-making Binnendyk and Pennycook (2023). Researchers from various fields, including psychology, economics, statistics, and engineering, have sought to understand and mitigate overconfidence in expert judgements (Kaustia and Perttula, 2012; Lambert et al., 2012; Lin and Bier, 2008; Angner, 2006). Drawing on concepts from (Clemen and Lichtendahl, 2002; Morris, 1974; Cox, 1958), a model was developed that uses historical data to estimate "inflation factors" for evaluated distributions retrospectively. Moore and Schatz (2017) outline three distinct manifestations of overconfidence.

Overestimation is the belief that one has greater abilities than one actually does, whereas overplacement is an inflated sense of one's abilities compared to others. Overprecision involves excessive confidence in one's knowledge of the truth. As summarised by Montibeller and von Winterfeldt (2018), both laypeople and experts tend to provide estimates for a given parameter that exceed actual performance (overestimation) (Lichtenstein et al., 1977) or give a range of variation that is too narrow (overprecision) (Moore and Healy, 2008). This bias has been demonstrated in various quantitative estimates across fields such as defence, law, finance, and engineering (Lin and Bier, 2008; Moore and Healy, 2008). It is also evident in judgements regarding the completeness of a hypothesis set (Fischhoff et al., 2013; Mehle, 1982).

(Russo et al., 1992; Skala, 2008; Moore and Schatz, 2017) argued that the inclination to feel more self-assured than what is justified by one's knowledge, expertise, or experience can be viewed as a meta-bias that serves as the foundation for various other decision-making biases. In fact, overconfidence can result in flawed reasoning (Russo et al., 1992; Soll and Klayman, 2004; Deaves et al., 2010; Kahneman, 2011; Chen et al., 2015; Ortoleva and Snowberg, 2015) or, in some cases, entirely supplant the process of reasoning itself (Thompson et al., 2011; Ackerman and Thompson, 2017). When applied to broader societal issues, overconfidence may play a significant role in assessing the reliability of information. For instance, overconfidence has been associated with belief in conspiracy theories (Vitriol and Marsh, 2018), adopting anti-scientific viewpoints (Light et al., 2022), and increased vulnerability to misinformation in general (Lyons et al., 2021).

3.2.4 Mood Effects in Judgement

Previous research has extensively explored expert judgement bias and disagreement, particularly in relation to emotions, from various psychological perspectives. Many studies have examined how emotions influence human memory, perception, judgement, and thinking, revealing the powerful impact that feelings can have on cognitive processes.

Sleboda and Lagerkvist (2022) identified a connection between attitudes and behavioural intentions, aligning with earlier studies and theories such as the Theory of Planned Behaviour (Armitage and Conner, 2001; Kahneman et al., 2000; Ajzen, 1991). Kahneman et al. (2000) highlighted that economics and psychology provide differing views on how people value things, suggesting that choices are better understood as expressions of attitudes rather than economic preferences. Phelps et al. (2014) found that moods influence decisions and hinted at the neural changes that may mediate these effects. Many studies have demonstrated that moods also impact risky choices; for instance, sad moods tend to increase preferences for high-risk options, while anxious moods bias preferences towards low-risk options (Raghunathan and Pham, 1999).

Wright and Bower (1992) discovered that an individual's mood can directly influence their judgement regarding the uncertainty of future events. They had subjects report subjective probabilities for personal and nonpersonal events while in happy, neutral, or sad moods. Further, Wright and Bower (1992) pointed out that traditional explanations for disagreements among experts, such as incompetence, venality, and ideology, fall short. Even skilled, honest, and impartial experts can have persistent disagreements because of human judgement's inherent characteristics and limitations. Additionally, Kahneman and Ritov (1994) and Kahneman et al. (2000) proposed that automatic affective valuation, which is the emotional core of an attitude, plays a crucial role in shaping many judgements and behaviors.

Kahneman et al. (2000) proposed that economics and psychology offer differing perspectives on how people value things. They emphasized that people's choices can be better understood as expressions of attitudes rather than as indications of economic preferences. After that, Loewenstein et al. (2001) introduced an alternative theoretical perspective known as the risk-as-feelings hypothesis, which underscores the influence of emotions experienced during decision-making. Moreover, Moyer and Song (2016) noted that affective feelings have either a "positive" or "negative" quality and that affective reactions often serve as initial responses to uncertainty, aiding individuals in navigating a complex, uncertain, and potentially hazardous world efficiently. Phelps et al. (2014) demonstrated that moods influence decision-making and provided insights into the neural changes that may mediate these effects. Numerous studies have shown that moods also impact risky choices; for instance, sad moods can lead to a preference for high-risk options, while anxious moods can bias preferences toward low-risk options (Raghunathan and Pham, 1999). More recent, Sleboda and Lagerkvist (2022) indicated a connection between attitudes and behavioural intentions, aligning with prior studies and theories like the Theory of Planned Behaviour (Armitage and Conner, 2001; Kahneman et al., 2000; Ajzen, 1991).

3.3 Methodology

The purpose of this chapter is to identify the attitude disagreement in judgemental prediction. We investigate the expert individual differences in terms of risk attitudes in forecasting the UK's main macroeconomic indicators. The risk attitude is defined here as the expert confidence level on one-year-ahead UK macroeconomy expectation. Here we define an expert who overall overestimates as an optimist expert and an expert who overall underestimates as a pessimist expert.

3.3.1 Construction of Experts' Judgements in Distributed Shape

In previous work, Brito et al. (2008) classified expert judgement with different attitudes into optimists and pessimists and also captured the expert judgement variability across different environments. They define an optimist expert as someone whose cumulative probability judgement distribution forms a narrow 'S' shape. A pessimistic expert is someone whose cumulative probability judgement distribution displays a broad 'S' shape. The construction of the 'S' shape is based on the cumulative frequency distribution of the expert's judgement. It is used to describe the cumulative probability distribution of the relative frequency at which P(loss) lies in different ranges of probability judgements is plotted to support analysis.

Here, we add a new perspective for depicting the shape of a cumulative probability distribution of expert judgement. Firstly, making an assumption that the expert error function fits its probability density function (PDF) to the Cauchy distribution. This assumption is based on the nature of properties of the Cauchy distribution, which itself contains the location parameter x_0 for specifying the location of the peak of the distribution, and the scale parameter γ which specifies the half-width at half-maximum. By applying this assumption, we can determine the location of the probability density distribution for each expert's point estimate. This approach simplifies the process of assessing the frequency of expert judgements occurring within different ranges of intervals.

According to Feller (1991) and Johnson et al. (1995), the model of the probability density function (PDF) in the Cauchy distribution can be expressed:

$$f(x; x_0; \gamma) = \frac{1}{\pi r}\left[1 + \left(\frac{x - x_0}{\gamma}\right)^2\right] = \frac{1}{\pi}\left[\frac{\gamma}{(x - x_0)^2 + \gamma^2}\right] \tag{3.1}$$

Where, x_0 is the location parameter, specifying the location of the peak of the distribution, and γ is the scale parameter which specifies the half-width at half-maximum,

alternatively 2γ is the full width at half maximum. γ is also equal to half the interquartile range and is known as the probable error. The maximum value or amplitude of the Cauchy PDF is $\frac{1}{\pi\gamma}$, located at $x = x_0$.

A Cumulative distribution function (CDF) is derived from (3.1),

$$F(x; x_0; \gamma) = \frac{1}{\pi} \arctan\left(\frac{x - x_0}{\gamma}\right) + \frac{1}{2} \tag{3.2}$$

and the quantile function (inverse cdf) of the Cauchy distribution is:

$$Q(p; x_0; \gamma) = x_0 + \gamma \tan\left[\pi\left(p - \frac{1}{2}\right)\right] \tag{3.3}$$

It follows that the first and third quartiles are $(x_0 - \gamma, x_0 + \gamma)$, and hence the interquartile range is 2γ. For the standard distribution, the cumulative distribution function simplifies to the arctangent function $\arctan(x)$.

$$F(x; 0; 1) = \frac{1}{\pi} \arctan(x) + \frac{1}{2} \tag{3.4}$$

In this study, we extend the analysis to include the Cumulative Distribution Function (CDF) for each expert and compare the width of the 'S' shape of the CDFs. Experts whose CDFs display an obviously broad 'S' shape will be identified as pessimistic experts, while those with an obviously narrow 'S' shape in their CDFs will be regarded as optimistic experts. Each expert's S - shaped cumulative distribution is shown to be rotationally symmetric around the axis of the standard Cauchy CDF distribution.

3.3.2 Construction of Central Tendency

Another approach that will be employed to identify whether forecasters lean towards pessimism or optimism involves applying the central tendency measurement, as derived from the insights of Huang et al. (2020) and Engelberg et al. (2009) employed a non-parametric approach, with a specific focus on the asymmetry within forecasters' point predictions. That is if the majority of forecasters' point predictions show minimal deviation from the median/mean/mode, but a small fraction deviates noticeably (approximately 10%). These forecasts display an asymmetry that indicates the forecasters tend to underestimate inflation and overestimate economic growth. To arrive at these findings, they focus on the limited number of forecasts that fall beyond the limits set by the median/mean/mode. Then, they analyse whether there is an asymmetry in these forecasts to shape their attitudes is optimism or pessimism. However,

Huang et al. (2020) mentioned that the approach of Engelberg et al. (2009) encountered a notable challenge: forecasts may occasionally deviate from the bounds set by the mean/mode/median due to the inherent randomness of forecasting. Additionally, forecasters might make occasional errors when generating their point forecasts, or they could incorporate asymmetric loss functions when determining their optimal forecasts. These factors can introduce inaccuracies in assessing whether a forecaster slopes toward optimism or pessimism. Expanding on this, Huang et al. (2020) presented an alternative method for evaluating the proportions of forecasters' point predictions that fall within the intervals spanning the 25th and 75th quartiles of the forecast's distribution. This method offers a more in-depth and informative assessment of the underlying optimism or pessimism within these forecasts. Additionally, it strengthens our capacity to derive more insightful insights into the potential loss functions that forecasters may employ. When it comes to the asymmetric linear loss function, it's crucial to specifically review the Bayesian decision theory, particularly in the discussion of optimal Bayesian point estimates under loss functions in Theorem 6.7.1 of (Poirier (1995)). It defines the concept of asymmetric loss functions, which allow for different quartiles of a distribution to be considered optimal forecasts. Forecasters who adopt such loss functions may generate point forecasts that either exceed or fall short of the median.

This loss function is given by Eq. (3.5)

$$C(\hat{\theta}, \theta) = \begin{cases} C_1 & |\hat{\theta} - \theta|, \text{ if } \hat{\theta} \leq \theta \\ C_2 & |\hat{\theta} + \theta|, \text{ if } \hat{\theta} \geq \theta \end{cases} \tag{3.5}$$

According to Theorem 6.7.1 in the study of (Poirier, 1995), the optimal prediction corresponds to the η-th quantile denoted as q_η, where η is calculated as $\eta = \frac{c_1}{c_1+c_2}$. If we set $c_1 = 1$ and $c_2 = 3$, this results in the optimal forecast being $q_{0.25}$. To understand this loss function, it's important to note that when the forecast, denoted as $\hat{\theta}$, exceeds the realized value, θ, it incurs a more significant penalty compared to a forecast that is lower than θ.

3.4 Data

3.4.1 Description of Variables

The data we use is sourced from the Bank of England Survey of Professional Forecasters (BoE-SPF). This dataset includes projections of the value for one year ahead of real GDP growth, inflation rate, base bank rate, unemployment rate, and sterling index in the UK. The survey commenced in 1999Q1, with quarterly publication of results. Point forecasts and probability distributions are provided by a panel of external professional

forecasters in London representing financial and academic institutions. For our analysis in this study, we restrict to data time horizon from 2000:Q1 to 2021:Q4.

Table 3.2 below displays fundamental data descriptions, abbreviations, sampling periods, and corresponding units for the indicators employed in this study. In analysing the point values of each variable in five indicators (GDP, Inflation, BoE, UR, and ERI), we examine the point value predicted by experts and its corresponding actual value. However, a limitation exists: for subjective estimation data by experts, the Bank of England did not provide information for the Sterling exchange rate index (ERI). Consequently, our evaluation is limited to four indicators GDP, Inflation, Bank rate, and Unemployment rate based on the available data for their subjective values.

TABLE 3.2: Indicator Details and Sample Periods

Indicator	Abbreviation	Start	End	Unit
Real GDP growth	GDP	2000Q1	2021Q4	Percentage %
Inflation rate	Inflation	2000Q1	2021Q4	Percentage %
Base bank rate	BoE	2000Q1	2021Q4	Percentage %
Unemployment rate	UR	2014Q4	2021Q4	Percentage %
Sterling exchange rate index	ERI	2000Q1	2021Q4	Percentage %

Note: 1. Sample periods indicate the start and end quarters of the data. 2.The overall period of the Unemployment rate is different from others due to the data access and availability limited. Besides, it has a lack of data for 2006Q1-Q3 and 2007Q1 for the other four indicators.

3.4.2 Point Prediction Data and Subjective Probabilistic Forecasts.

Table 3.3 presents a comprehensive overview of statistical measures for the difference between point-predicted data and the actual value for each of the five indicators. Specifically, it includes the means μ and standard deviations σ related to various variables. These variables comprise the actual announced value (referred to as "Actual"), the predicted value by experts ("Experts"), and the resulting surprise, which is derived from the disparity between the Actual and Expert values.

The collection of subjective probabilistic data is similar with the point prediction collection. The SPF instrument divides the real number line into several intervals and ask respondents to report their subjective probabilities that the variable of interest and take a value in each interval. For example, in GDP growth, The intervals are $[-1\%, 0\%)$, $[0\%, 1\%)$, $[1\%, 2\%)$, $[2\%, 3\%)$, $[3\%, 5\%)$, $[5\%, 7\%)$, $[7\%, 9\%)$, $[9\%, \infty)$, and the sum of each interval expressed is 100%. Table 3.4 provides useful summary information on the subjective prediction data. Table 3.5 lists the total interval quantity for each variable and its corresponding subset interval value.

TABLE 3.3: Actual, Expert Forecast, and Surprise Statistics for Point Data

	Actual			Expert Forecast		Surprise	
	N	μ	σ	μ	σ	μ	σ
GDP	681	0.4179	3.0994	2.1508	1.6736	1.7329	1.4258
Inflation	668	2.0	0.0913	1.8499	0.8112	0.1025	0.1020
Bank rate	221	2.0929	2.1541	3.0933	1.9299	1.0005	0.2242
Unemployment	205	4.7452	0.7767	5.3082	2.4248	0.5631	1.6481
ERI	598	88.1971	9.5592	92.963	40.0396	4.7659	30.4803

Note: In this case, we do not consider predictions from all forecasters. Instead, we select ten experts who provide the most informative values to form a panel. The amount of observation is theoretically the data number we should have. However, due to certain forecasters not providing their beliefs at some time points, we encounter missing data.

TABLE 3.4: Variable, Experts, Observations, Missing Observations

Variable	Experts	Observations	Missing Observations
GDP	10	681	129
Inflation	10	668	131
Unemployment	10	205	87
Bank rate	10	221	124
ALL	-	1775	471

Note: In this case, we do not consider predictions from all forecasters. Instead, we select ten experts who provide the most informative values to form a panel. The amount of observation amount is theoretically the data number we should have. However, due to certain forecasters not providing their beliefs at some time points, we encounter missing data.

TABLE 3.5: Variable, Intervals Number, Intervals Values

Variable	Intervals Number	Intervals Values
GDP	8	$[-1\%, 0\%), [0\%, 1\%), [1\%, 2\%), [2\%, 3\%), [3\%, 5\%), [5\%, 7\%), [7\%, 9\%), [9\%, \infty)$
Inflation	8	$[-\infty, 0\%), [0\%, 1\%), [1\%, 1.5\%), [1.5\%, 2\%), [2\%, 2.5\%), [2.5\%, 3\%), [3\%, 3.5\%), [3.5\%, \infty)$
Unemployment	10	$[-\infty, 4\%), [4\%, 4.5\%), [4.5\%, 5\%), [5\%, 5.5\%), [5.5\%, 6\%), [6\%, 6.5\%), [6.5\%, 7\%), [7\%, 7.5\%), [7.5\%, 8\%), [8\%, \infty)$
Bank rate	10	$[\infty, 0\%), [0\%, 0.5\%), [0.5\%, 1\%), [1\%, 1.5\%), [1.5\%, 2\%), [2\%, 2.5\%), [2.5\%, 3$

3.4.3 Preliminary Analysis

Table 3.6 presents the average values of experts' upper bounds on median, mean, mode point predictions, and quantiles. The upper bound on the median point prediction represents the maximum value that the median of the predictive model can attain. Similarly, the upper bound on the mean point prediction indicates the highest possible value for the mean of a predictive model. The upper bound on the mode point prediction reflects the highest achievable value for the mode of your predictive model.

Additionally, the upper bounds on quantiles, specifically the 75th percentile, signify the maximum values that these quantiles can reach within the predictive model. These upper bounds on the 25th quantile provide insight into the potential range of values

within specific data intervals, contributing to a better understanding of the data distribution.

TABLE 3.6: Average of upper bounds on median/mean/mode point predictions and quantiles.

Variable	Median	Mean	Mode	25th quartile	75th quartile
GDP	53.80%	59.78%	57.79%	21.92%	47.28%
Inflation	40.27%	40.57%	37.13%	21.56%	19.76%
Unemployment	20.98%	12.68%	14.63%	9.76%	10.73%
BoE	16.29%	11.31%	15.84%	13.57%	12.67%
ALL	49.46%	50.00%	48.47%	24.16%	33.97%

3.5 Main Results

The main results of the study are presented in two key parts. Firstly, the parameter values used to construct the cumulative "S" shapes are highlighted, with a focus on identifying the key points in forming the cumulative distribution functions (CDFs) of the experts' predicted values. Table 3.7 summarises the parameter values, including the location of the distribution peak and the scale parameter (measured by the half-width), for ten experts (B1 to X1) across the five target observation variables: GDP, inflation, bank rate, unemployment, and ERI. All values are reported in percentage format for consistency.

Subsequently, Fig. 3.1 illustrates the differences between the cumulative "S" shapes of the ten experts when predicting the five target variables. This visual comparison provides insights into the variability in the predictive distributions of individual experts. Furthermore, Table 3.8 presents the classification of each expert's predictive attitude, indicating tendencies in their approach to forecasting the target variables.

3.5.1 Summary of the Value of the Location of the Peak of the Distribution and the Scale Parameter with the Half-width

The results in Table 3.7 provide key information on the parameters used to fit the experts' cumulative distribution functions (CDFs) within a Cauchy distribution. These parameters include the location of the peak, the scale parameter, and the corresponding half-widths. These values enable the construction of "S"-shaped CDFs for each expert, reflecting their subjective probability distributions when predicting the UK's five key macroeconomic indicators.

TABLE 3.7: Parameter Values for Experts

Variables	Parameters	B1	G1	I1	L1	N1	O1	S1	T1	X1	B2
*GDP	x_0	250	2.17	228.57	314.29	242.86	228.57	283.33	240	239.29	200
	γ	202.92	1.44	147.27	175.45	171.53	204.58	157.5	155	150.83	165.89
*Inflation	x_0	9.13	4.76	11.11	7.14	-4.33	9.52	14.84	10.82	15.48	0
	γ	60.66	42.50	61.85	54.30	63.43	65.19	45.23	66.60	49.23	40.52
*BOE	x_0	1.40	4.71	3.86	7.02	1.67	6.45	0	6.25	4.14	2.59
	γ	25	50	96.07	37.5	41.28	50	57.60	100	25	25
*Unemployment rate	x_0	11.36	12.66	10.64	9.43	11.79	12.75	15.21	16.98	13.16	15.35
	γ	7.23	10.69	4.45	6.58	7.37	8.65	5.66	8.93	5.57	9.37
*ERI	x_0	12.44	-0.0037	4.32	2.54	2.56	2.81	2.50	2.48	0.51	1.37
	γ	9.53	4.26	4.33	2.51	2.71	3.53	3.84	4.75	4.82	3.88

Notes: In the Cauchy distribution, x_0 is the location parameter, specifying the location of the peak of the distribution, and λ is the scale parameter which specifies the half-width at half-maximum. All value is presented in %.

3.5.2 Frequency Distribution of Experts' Judgements

This section examines two types of variability typically found in expert probability judgements. We will then infer the forecaster's preference in prediction based on evidence from these variabilities. Firstly, we explore expert judgement variability across different target variables. Secondly, we delve into the variability in the totality of their assessments between experts. The analysis of judgement variability begins by examining how often experts use different ranges of probability judgements. In our initial analysis, we utilize ten probability intervals. The cumulative distribution of the relative frequency, depicting the difference between the forecaster's prediction and the actual value within each range, is plotted to support the analysis—refer to Fig. 3.1.

Differing from Brito et al. (2008), who constructed the "S" shape of experts' subjective probability distribution with ten intervals of probability, we employ a simpler variation on their original model to facilitate a more intuitive comparison. We standardize all subjective "S" shapes by gathering them at a central point and allowing their variations to rotate around this central point. This approach enables us to intuitively compare each individual expert's confidence level when making predictions for each target variable. Consequently, we can derive a preliminary observation of each individual expert's preference in prediction.

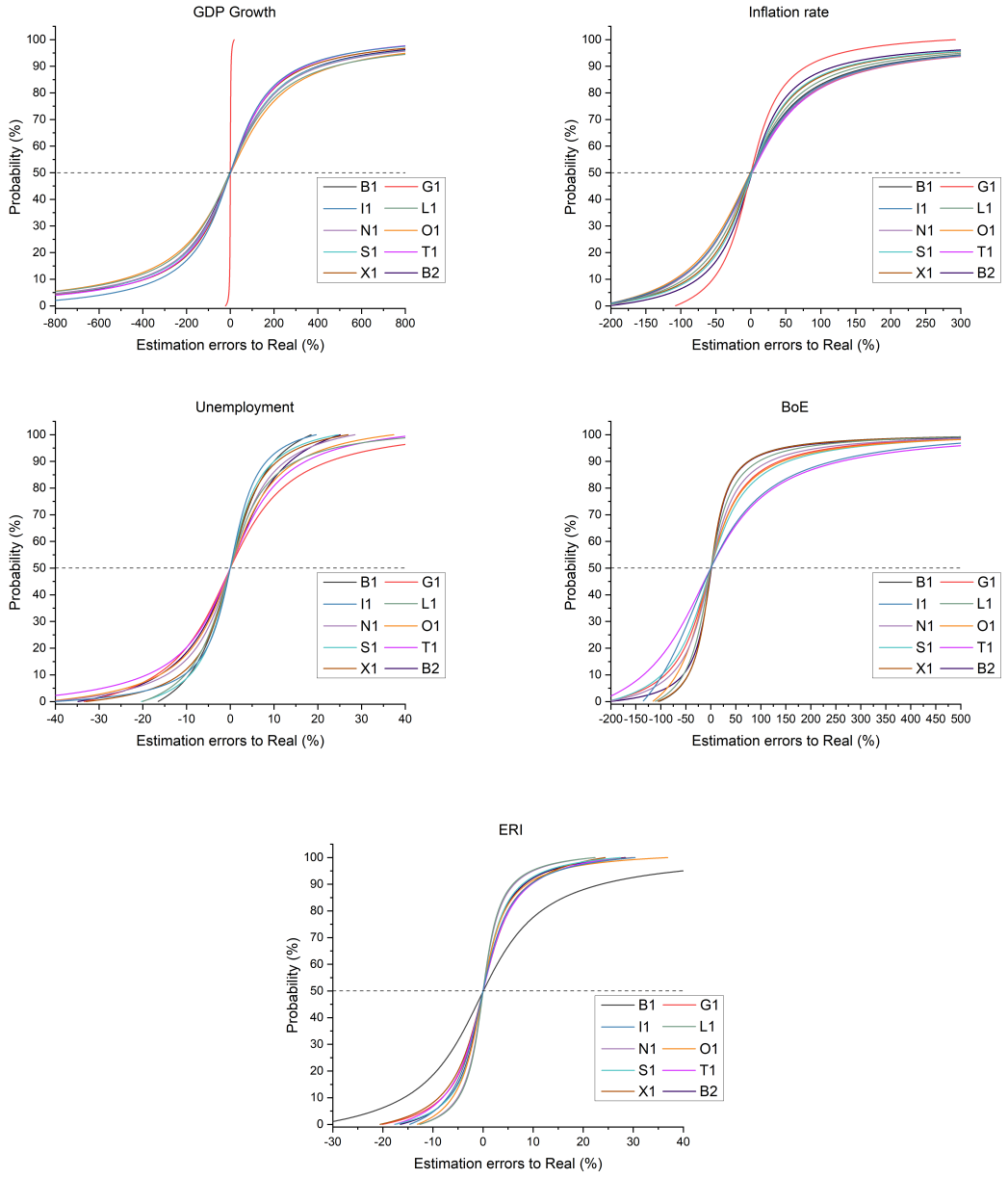

FIGURE 3.1: Cumulative probability distributions. GDP (top left), Inflation (top middle), ERI (top right), BoE (bottom left), and Unemployment (bottom right).

3.5.3 Classification of Expert's Attitude Based on "S" Shape.

Table 3.8 summarises the attitudes of each expert when providing predictions for the five observed variables. These classifications are derived from the results presented in Fig. 3.1. Experts are categorised based on the distance (rotation angle) between the shape of their "S" curve and the axis centre. The width of the "S" shape reflects the confidence level of an expert in predicting that their estimated interval will contain the actual value.

If an expert's "S" shape rotates significantly away from the axis centre, it indicates a wider belief interval, suggesting a pessimistic attitude towards their prediction. Conversely, a narrower rotation closer to the centre represents higher confidence, typically associated with an optimistic attitude. This analysis provides a systematic approach to characterising the subjective confidence and prediction tendencies of each expert across the observed variables.

TABLE 3.8: Variables, Attitude, and Experts

Observation Variables	Attitude	Experts
GDP	Optimist	G1
	Pessimist	B1, L1, B2, I1, N1, O1, S1, T1, X1
Inflation	Optimist	G1, B2
	Pessimist	B1, I1, N1, O1, T1, L1, S1, X1
Bank rate	Optimist	B1, X1, B2
	Pessimist	I1, T1, G1, L1, N1, O1, S1
Unemployment rate	Optimist	I1, L1
	Pessimist	G1, B1, B2, N1, O1, S1, T1, X1
ERI	Optimist	L1, N1
	Pessimist	B1, G1, I1, O1, S1, T1, X1, B2

In the process of quantifying the frequency of classifying each expert's attitudes as either optimistic or pessimistic, a preliminary analysis suggests that experts G1, L1, and B2 adopt a notably optimistic outlook regarding the UK's five key macroeconomic indicators. Meanwhile, experts B1, X1, I1, and N1 display a relatively optimistic stance, albeit less pronounced than the aforementioned experts. By contrast, the remaining experts, O1, S1, and T1, exhibit a distinctly pessimistic attitude. Overall, across all variables, the number of experts with a pessimistic outlook significantly exceeds those with an optimistic perspective.

3.5.4 Bounding Means, Medians, and Modes and Quartiles

The rationality of the forecasts can be evaluated in terms of non-parametric statistical theory. Since the BoE-SPF asks forecasters to provide their subjective probability distribution and point forecast, we can use the distribution to construct bounds on the relevant median/mean/mode forecasts. We follow (Engelberg et al., 2009), who explained how to calculate bounds on the median, mean, and mode. In addition, we follow (Huang et al., 2020) calculated bounds intervals for lower and upper bounds on the 25th and 75th quartiles. In summary, the central tendency of median/mean/mode and the lower bound 25th and upper bound 75th quartiles are used to shape the interval bounds for counting the frequency of subjective probability distribution where they fall into or exceed.

TABLE 3.9: Summary Statistics for Experts on GDP

Experts	Median	Mean	Mode	25th Quartile	75th Quartile
B1	42.00%	56.00%	44.00%	22.00%	60.00%
G1	50.00%	53.33%	55.00%	20.00%	55.00%
I1	44.68%	48.94%	48.94%	21.28%	51.06%
L1	58.97%	58.97%	64.10%	15.38%	61.54%
N1	61.29%	74.19%	70.97%	20.97%	40.32%
O1	64.41%	71.19%	55.93%	20.34%	23.73%
S1	55.56%	63.49%	61.90%	25.40%	55.56%
T1	70.21%	70.21%	74.47%	17.02%	53.19%
X1	48.39%	51.61%	54.84%	17.74%	53.23%
B2	44.44%	49.21%	49.21%	34.92%	28.57%

TABLE 3.10: Summary Statistics for Experts on Inflation

Experts	Median	Mean	Mode	25th Quartile	75th Quartile
B1	35.29%	41.18%	35.29%	25.49%	41.18%
G1	36.84%	39.47%	31.58%	23.68%	10.53%
I1	44.00%	40.82%	26.53%	22.45%	26.53%
L1	56.76%	54.05%	56.76%	24.32%	21.62%
N1	66.67%	60.00%	65.00%	35.00%	11.67%
O1	45.76%	54.24%	35.59%	15.25%	18.64%
S1	57.14%	52.38%	55.56%	34.92%	28.57%
T1	56.00%	52.00%	52.00%	30.00%	36.00%
X1	45.45%	46.97%	42.42%	10.61%	21.21%
B2	51.56%	57.81%	54.69%	43.75%	28.13%

TABLE 3.11: Summary Statistics for Experts on Unemployment

Experts	Median	Mean	Mode	25th Quartile	75th Quartile
B1	35.29%	41.18%	35.29%	25.49%	41.18%
G1	36.84%	39.47%	31.58%	23.68%	10.53%
I1	44.00%	40.82%	26.53%	22.45%	26.53%
L1	56.76%	54.05%	56.76%	24.32%	21.62%
N1	66.67%	60.00%	65.00%	35.00%	11.67%
O1	45.76%	54.24%	35.59%	15.25%	18.64%
S1	57.14%	52.38%	55.56%	34.92%	28.57%
T1	56.00%	52.00%	52.00%	30.00%	36.00%
X1	45.45%	46.97%	42.42%	10.61%	21.21%
B2	51.56%	57.81%	54.69%	43.75%	28.13%

Following the interpretation by (Engelberg et al., 2009), if the point prediction lies within the bound for the median, then we cannot reject the hypothesis that the point prediction is the median. We can reject this hypothesis if the point prediction does not lie within the bound for the median. The same reasoning applies to the mean and

TABLE 3.12: Summary Statistics for Experts on Bank Rate

Experts	Median	Mean	Mode	25th Quartile	75th Quartile
B1	41.18%	29.41%	41.18%	41.18%	35.29%
G1	33.33%	28.57%	33.33%	23.81%	33.33%
I1	28.57%	0.00%	14.29%	0.00%	42.86%
L1	-	-	-	-	-
N1	47.83%	21.74%	47.83%	34.78%	26.09%
O1	-	-	-	-	-
S1	-	-	-	-	-
T1	33.33%	33.33%	33.33%	0.00%	0.00%
X1	26.67%	26.67%	33.33%	40.00%	33.33%
B2	33.33%	16.67%	33.33%	41.67%	8.33%

mode. Thus, we can determine the frequency with which point predictions are inconsistent with the three measures of central tendency. Combining the results in Table 3.9 (the average of expert's point prediction), we found that in GDP growth prediction, the experts G1, L1, and S1 are consistent with the experts' average performance. Experts N1, O1, and T1's consistency is slightly above the average performance. In Inflation rate prediction, the overall consistency of experts is above the average level except for expert B1. In unemployment prediction, the experts are above average overall, except for B2. In Base bank rate prediction, all experts are above average overall. We need to learn more about the evidence to further understand the expert's point prediction's favorableness. More evidence will be interpreted in the following section.

By examining the percentage frequency of point prediction consistency with a subjective probability distribution on the intervals of central tendency, we cannot infer whether these forecasters tend toward holding an optimistic or pessimistic attitude toward the state of the UK macroeconomy when predicting key indicators. However, if we consider in addition to the lower bound and upper bound for analysing the expert's favourable scenarios, we will infer the skewness of each individual expert's and subjective probability distribution. More specifically, if it has more forecasters' predictions align with the interval for the 75th quartile, as opposed to the 25th quartile of their distribution, it suggests that forecasters often make predictions favouring the higher end of expectations. This implies optimism in their forecasts for GDP growth and the Base bank rate but tends to be pessimistic in their predictions for Inflation and Unemployment rates.

Analysis by variables in GDP growth, combined with Table 3.6 and Table 3.9, shows that there is a larger proportion of forecasters' point predictions consistent with the 75th quartile of their subjective distributions, ranging from 28.57% to 60% across all forecasters, compared to those consistent with the 25th quartile which ranges from 22% to 34.92% across all forecasters. This suggests that forecasters are relatively typical

optimists since their GDP growth predictions are more toward the right tail of their subjective distribution than towards the left tail.

For inflation predictions, Table 3.10 indicates that there are slightly fewer forecasters whose point predictions align with the 75th quartile. This percentage ranges from 28.13% to 41.18% across all forecasters. In comparison, the proportion of predictions consistent with the 25th quartile ranges from 25.49% to 43.75% across all forecasters within their subjective distributions. This suggests that, overall, forecasters tend to be pessimists when it comes to reporting inflation predictions. Their predictions lean more slightly towards the left tail of their subjective distribution than towards the right tail.

Regarding unemployment rate predictions, Table 3.11, shows the proportion of point predictions aligning with the 75th quartile ranges from 26.67% to 28.57%, whereas those consistent with the 25th quartile range from 13.33% to 14.29% across all forecasters. This suggests that, in general, forecasters in the UK-SPF tend to be optimistic in their unemployment predictions. The predictions are more towards the right tail of their subjective distribution than towards the left tail.

Finally, for the bank rate predictions, Table 3.12 reveals that the percentage of forecasters' point predictions align with the 75th quartile of their subjective distributions ranging from 8.33% to 35.29% across all forecasters. In contrast, the proportion of predictions consistent with the 25th quartile ranges from 41.18% to 41.67% across all forecasters. This observation implies that forecasters tend to lean toward pessimism in their BoE predictions. The overall predictions are more towards their subjective distribution's left tail than the right tail.

3.5.5 Inconsistency Tend to Present Favorable Scenarios

Now consider the SPF panel members whose point predictions are inconsistent with their subjective medians, means, or modes. Table 3.13 reports the percentage of cases in which point predictions lie above or below the bounds. A clear finding emerges: most inconsistent point predictions reflect a view of the economy that is favourable relative to the central tendencies of the experts' subjective distributions. This suggests that forecasters who skew their point predictions often present optimistic scenarios.

For instance, when forecasting GDP growth, point predictions inconsistent with measures of central tendency are far more likely to exceed the upper bounds than fall below the lower bounds of the means, medians, and modes. This trend indicates a preference for more optimistic forecasts of economic growth, suggesting that forecasters view the economy as performing better than indicated by their subjective distributions. Conversely, for inflation forecasts, point predictions tend to fall below the lower

bounds rather than above the upper bounds, implying a bias towards projecting more favourable (i.e., lower) inflation outcomes.

The unemployment rate predictions follow a similar pattern to GDP growth, with inconsistent point predictions skewed towards the upper bounds, reflecting a more optimistic view of labour market conditions. On the other hand, for the Base Bank Rate forecasts, the inconsistent point predictions tend to fall below the lower bounds, indicating a more cautious or pessimistic stance on monetary policy outcomes.

We cannot definitively determine why forecasters skew their point predictions in this manner. One potential explanation could involve behavioural tendencies, such as optimism bias, where forecasters inherently overestimate positive outcomes for certain variables while underestimating risks for others. Alternatively, strategic considerations may play a role. For example, Capistrán and Timmermann (2009) proposed a model in which forecasters have incentives to under- or overpredict due to asymmetries in the cost functions associated with prediction errors. These strategic biases might reflect forecasters' attempts to align their predictions with perceived expectations of stakeholders or decision-makers.

However, since individual forecasts are anonymised in the public release of the panel data, and forecasters are ostensibly unaware of each other's responses during the survey process, it is unlikely that herding behaviour or career-related concerns are the primary drivers of these findings. Further research is required to explore the motivations and mechanisms behind these observed patterns of inconsistency.

3.5.6 Additional Analysis

This section provides additional analysis aimed at further elucidating the differences in preferences among expert predictions. This is achieved by comparing the average subjective distribution of each individual expert with the actual value for each variable.

Beginning with Fig. 3.2, the red dotted line represents the average true value of each variable, depicted as a straight line since the true value at each quarterly time point is a numerical value. The distributions reflect the average of each expert's subjective probability distribution. Notably, expert predictions for all four variables consistently appear on the right side of the true value, indicating a tendency among experts to overestimate the macroeconomy.

For more granular insights, Fig. 3.3 provides complementary perspectives on experts' forecasting preferences, focusing on the width and position of their subjective distributions. This figure further demonstrates that experts' predicted values systematically fall to the right of the true value distributions, reinforcing the observation of a general overestimation bias.

TABLE 3.13: Summary statistics for variables

Variable	Median		Mean		Mode		25th Quartile		75th Quartile	
	Below	**Above**	**Below**	**Above**	**Below**	**Above**	**Below**	**Above**	**Below**	**Above**
GDP	255		222		233		431		291	
	34.12%	65.88%	39.64%	60.36%	39.91%	60.09%	9.05%	90.95%	84.19%	15.81%
Inflation	269		266		289		393		405	
	31.97%	68.03%	46.62%	53.38%	32.87%	67.13 %	8.40%	91.60%	85.19%	14.81%
Unemployment	75		92		98		96		405	
	22.67%	77.33%	31.52%	68.48%	18.18%	81.82%	8.16%	91.84%	61.46%	38.54%
BoE	62		73		68		70		405	
	43.55%	56.45%	58.90%	41.10%	47.62%	52.38%	26.47%	73.53%	82.86%	17.14%

Differences in the shapes of distributions reveal additional insights. For GDP growth predictions, a steep slope and systematically narrow ”S” shapes suggest that experts exhibit an optimistic outlook towards economic growth. Conversely, inflation predictions display flatter ”S” shapes, and a high expectation value indicates a more pessimistic perspective on the economic environment.

Similarly, for unemployment forecasts, the interpretation aligns with inflation. Higher predicted values imply an economic downturn, reflecting a pessimistic stance among experts. Finally, for bank rate predictions, experts tend to exhibit optimism, evidenced by steep slopes and consistently skewed distributions to the right of the true values.

3.5.6.1 Comparison of Expert Average Subjective Prediction with True Value.

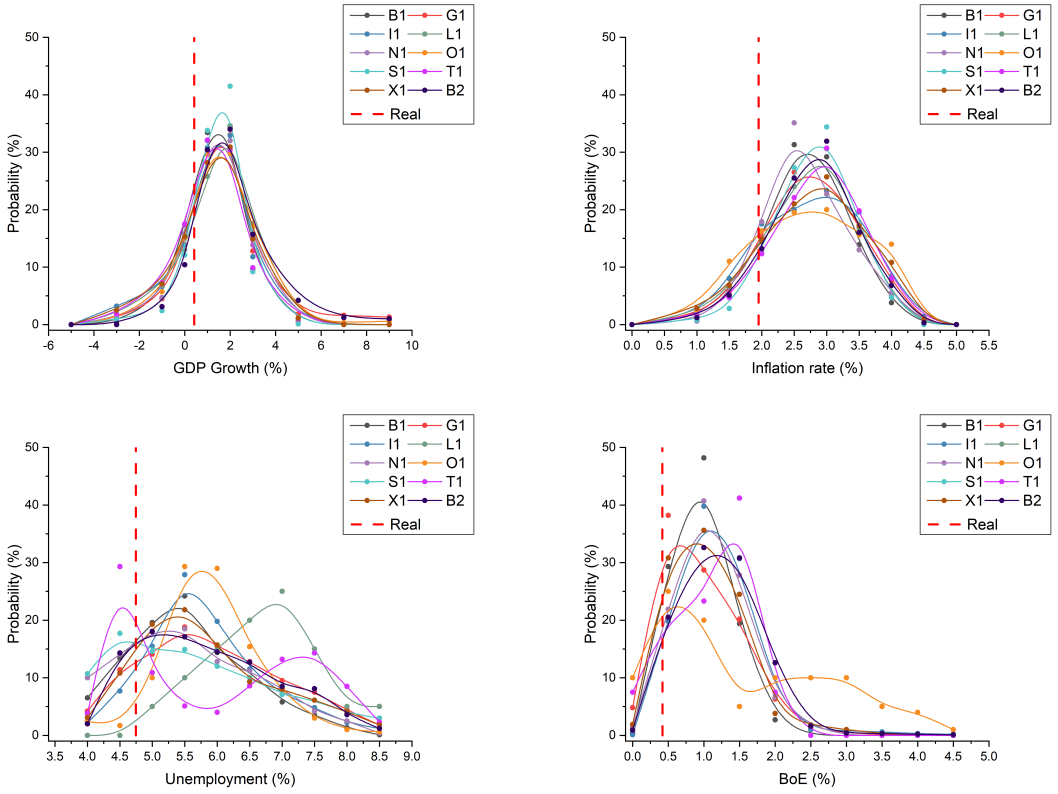

FIGURE 3.2: Comparison of expert average subjective prediction with true value.

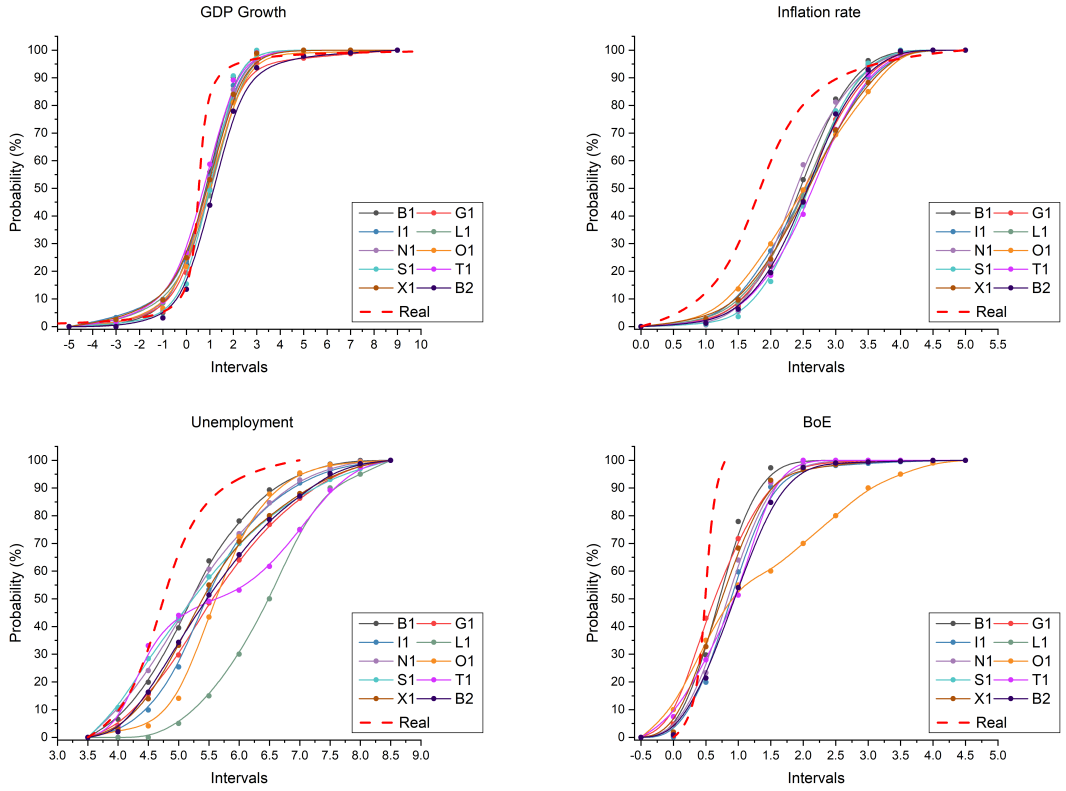

FIGURE 3.3: Comparison of expert average subjective prediction with true value in CDF.

3.6 Conclusion

This research paper makes several significant contributions to understanding whether expert behaviour in macroeconomic forecasting is rational. Firstly, it introduces a novel approach to understanding attitudinal differences among experts by classifying them into optimists and pessimists based on the "S" shape width of their cumulative probability distributions, rooted in their emotional characteristics and prediction behaviours. Secondly, the study is deeply grounded in psychology and cognitive theory, drawing on the foundational work of (Tversky and Kahneman, 1974) on heuristics and biases to enhance judgement and decision-making under uncertainty. Thirdly, it examines the impact of cognitive biases on expert judgement, emphasising how heuristics can lead to skewed decision-making despite their general utility. Additionally, the paper employs a robust approach, integrating Bayesian decision theory to assess the consistency of forecasts with central tendency values and interquartile ranges, thereby evaluating the rationality of these predictions. This is supported by extensive data analysis of economic indicators like GDP growth, unemployment rate, bank rate, and inflation. These contributions collectively advance the literature on expert judgement and decision-making, offering new insights into the rationality, or lack thereof, in economic forecasting.

Moreover, our research expands on some of the recent literature on survey-based measures of macroeconomic uncertainty and forecasters' rationality. It also contributes to understanding the optimism or pessimism of point forecasts made by participants in the BoE-SPF. To do this, we employ the BoE-SPF, which uses 21 years of quarterly survey data covering 84 surveys with ten experts' point predictions in a time series format.

Our results, viewed from different perspectives, indicate that forecasters exhibit less rationality in the process of making macroeconomic predictions. The method proposed by presents the evidence (Brito et al., 2008). First, we broadly categorize experts' attitudes into different moods based on the width of their subjective distribution in an "S" shape. Additionally, we establish bounds for a rational threshold to examine whether forecasters' point predictions are consistent or inconsistent with the central tendencies of their subjective distributions.

The results reveal that most BoE-SPF point predictions are inconsistent with these central tendencies. This inconsistency implies that the forecasters' point predictions do not align well with their subjective probability distributions' central values (mean, median, mode). In other words, the specific numerical forecasts provided by the experts differ significantly from the central tendency measures that summarize their overall expectations. This misalignment suggests potential biases or errors in the experts' forecasting process, highlighting a lack of coherence between their predictions and broader probabilistic assessments. This discrepancy can undermine the rationality of the forecasts and indicates the need for improved methods to ensure consistency in expert judgement.

Furthermore, we observe that deviations between point predictions and the central tendencies of forecasters' subjective distributions tend to be asymmetric. The point predictions consistently present a more favourable view of the economy than suggested by subjective means, medians, modes, and the 25th and 75th quartiles. This skewness in subjective distribution reports indicates predictions that give a more favourable view of the economy. Our findings align with the results of (Batchelor, 2007), which document the presence of systematic bias in real GDP and inflation forecasts.

However, when classifying experts into optimists or pessimists, establishing a strict rule for the spread of the 'S' shape poses a challenge. This study does not reflect an obvious criterion for distinguishing experts into the two groups. We classify them based on their relative performance with their peers. The limitation is that, when predicting some variables, there is a possibility that they are all optimists or pessimists, so the comparison with peers is not efficient. One way to address this limitation could be to introduce a dynamic threshold that adapts to the overall distribution of predictions within the group. Instead of having a fixed threshold, consider using statistical measures like standard deviations or percentiles to determine the optimism or pessimism

of each expert relative to the group. This way, the classification takes into account the broader context of predictions within the peer group.

In conclusion, this study significantly impacts the understanding and practice of economic forecasting. By highlighting the inconsistencies and biases in expert predictions, it provides a foundation for improving forecasting methods, ensuring more accurate and reliable economic predictions. These insights are particularly relevant for the Bank of England as they inform the need for refined approaches in eliciting and aggregating expert judgements, ultimately enhancing the reliability of macroeconomic forecasts used in policy-making.

Chapter 4

Machine Learning in Expert Prediction Optimization

4.1 Introduction

The development of measurement in macroeconomic forecasting has a long history. As early as 1936, the econometrist known as the father of macroeconomics, Keynes, discussed in his study, "The General Theory of Employment, Interest, and Money", that state intervention was necessary to moderate the "boom and bust" cycles of economic activity (Keynes, 1937). Keynes advocates using fiscal and monetary policies to alleviate the adverse effects of economic recessions and depressions. However, this work was criticised by Pigou (1936), who argued that Keynes' method is less appropriate in the way of scientific modelling and develops a far-reaching generalization. More recently, Diebold (1998) noted that many observers interpret the failure of the early models as indicative of a bleak future for macroeconomic forecasting more generally. Diebold (1998) further indicated that following the decline of Keynesian theory, a powerful new dynamic stochastic general equilibrium theory has been developed, and structural macroeconomic forecasting is poised for resurgence.

Likewise, numerous modern researchers also emphasize the importance of macroeconomic forecasting, as Schuh et al. (2001) highlighted that macroeconomic forecasts are a useful tool and can be used extensively in industry and government. However, Heilemann and Stekler (2007) indicated that the accuracy of macroeconomic forecasts in the G7 has not improved over the last fifty years. He further comments on the concern that predicting the future of economic forecasting is more difficult than the forecasting itself because there is no theory we can use, and few clear trends are evident. Moreover, Mullainathan and Spiess (2017) demonstrated traditional economic forecasting models rely on established variables and typically adopt a top-down, theory-driven approach that considers the cause-and-effect relationships between dependent and independent

variables. However, how efficiency of these models depends on the economic insight and judgement of forecasters concerning both data and methodologies employed. If there are any errors or irrationality in the assumptions made by forecasters, it can lead to inaccurate predictions by these models.

On the other hand, as described by Lu et al. (2009), there has been an increasing interest in economic time series forecasting in recent years, driven by the importance of accurate economic index predictions for investment decision-making. However, financial time series are inherently noisy and non-stationary (Deboeck, 1994; Yaser and Atiya, 1996). The noise characteristic refers to incomplete information from past economic market behaviour, making it difficult to fully capture the dependency between future and past states. Information not included in the forecasting model is considered noise, while the non-stationary characteristic implies that the distribution of economic time series changes over time. Therefore, economic time series forecasting is considered one of the most challenging tasks in time series forecasting. As a result, there is a continuous need to explore and refine optimal methods for economic forecasting.

Additionally, Clemen (1989) proposed a persistent issue with statistical-based forecasting models is their heavy reliance on correlations between data to identify patterns. However, the accuracy of predictions may fall short when the data is inadequate for modelling. Thus, we refer to Varian (2014) pointed out that machine algorithms primarily focus on pure prediction, unlike many traditional economic forecasting models. Yoon (2021) evidenced that machine learning models exhibit greater flexibility than their traditional counterparts, as they can generate predictions without needing pre-established assumptions or human judgement. In fact, with advancements in technology and enhanced predictive capabilities, machine learning models have found extensive applications across various domains, ranging from predicting transportation patterns to forecasting housing prices. Notably, machine learning methods often outperform traditional econometric models, as evidenced in the case of forecasting US housing prices by (Plakandaras et al., 2015). Furthermore, machine learning models are effective when applied to datasets with relatively lower frequencies, as demonstrated in studies on inflation prediction conducted by (Inoue and Kilian, 2008; Medeiros et al., 2021).

On the other hand, the fusion of machine learning and econometrics has emerged as a significant research domain within the field of economics. ML has gained importance primarily due to the availability of extensive datasets, particularly in microeconomic contexts, as highlighted by (Belloni et al., 2017; Abadie and Kasy, 2019). While the interest in ML grows, comprehending how ML techniques can effectively contribute to forecasting macroeconomic outcomes remains a remarkable challenge. However, this understanding holds significant value, potentially surpassing the reliance on a singular

algorithm. Applied econometricians often find it more appealing to augment a conventional framework with specific insights from ML rather than replacing it entirely with an ML model (Goulet Coulombe et al., 2022).

As Huang et al. (2020) denoted deep learning (DL) is an advanced technique of machine learning (ML) based on artificial neural network (NN) algorithms. Deep learning has been widely applied in computer vision (Guo et al., 2016), natural language processing (Collobert et al., 2011), and audio-visual recognition (Chai and Li, 2019). The overwhelming success of deep learning as a data processing technique has sparked the interest of the research community. However, a detailed study of the applications of deep learning in the finance and banking field is lacking in the existing literature. This study provides a powerful comprehensive framework to compare how different for improving macroeconomic prediction by using deep learning logarithms and machine learning algorithms. It helps to understand how the DL and ML procedures can address the remaining challenge described (Goulet Coulombe et al., 2022).

In this context, we contribute to the literature by conducting an extensive empirical study, employing a comprehensive framework that integrates both deep learning and machine learning as a hybrid approach to optimize UK macroeconomic forecasts. The algorithm architectures are constructed using well-known deep learning models, including deep neural networks (DNN) and Long Short-Term Memory (LSTM), along with machine learning models such as Support Vector Regression (SVR) and Random Forest (RF). We establish the optimal combination of machine learning algorithms tailored for UK professional forecasters in macroeconomic forecasting. Our results indicate that machine learning models demonstrate a significant predictive ability, and including deep learning models proves effective in achieving optimization goals in macroeconomic forecasting. Additionally, our contribution lies in finding solutions for selecting optimal hyperparameters, addressing a critical problem in machine learning. Furthermore, our findings provide evidence that the application of machine learning remains valid even with a small data sample. Our exploration introduces a new perspective on strategies for optimising macroeconomic forecasting.

This paper is organised into several sections. The first section summarizes the research background and objectives. The second section describes data construction and preprocessing. We will then introduce each individual ML model (SVR and RF) and DL model (DNN and LSTM) applied in this study. The following section presents the details of prediction results and discussion. The final part will conclude with some findings, future work, and limitations.

4.2 Literature Review

Barboza et al. (2017) pointed out that machine learning and deep learning as useful

tools that have been found applied across a wide range of research fields. El Naqa and Murphy (2015) summarised techniques based on machine learning and deep learning have been applied successfully in diverse fields ranging from pattern recognition (Melati et al., 2019), computer vision (Khan et al., 2021), engineering (Panchal et al., 2019), finance (Gogas and Papadimitriou, 2021), arts (Fiebrink, 2019), education (Giannakos et al., 2020) and computational biology (Tarca et al., 2007) and medical applications (Magoulas and Prentza, 1999). However, recent Goulet Coulombe et al. (2022) indicated the gap that only recently did macroeconomic forecasting experience a surge in the number of studies applying (successfully) ML methods, and many tasks remain to be explored. This section reviews the application of machine learning and deep learning in macroeconomic fields.

In traditional econometrics, multiple linear regression and Ordinary Least Squares (OLS) are seen as the two frequently used econometric techniques in the process of analysing economic data (Shobana and Umamaheswari, 2021). The main goal of econometricians is to determine such an estimator that can possess certain desirable statistical properties like consistency, efficiency, and unbiasedness. However, (Shobana and Umamaheswari, 2021) further highlighted the limitation of traditional econometrics is that certain econometric models may end up with a result that gives a relationship that is spurious among two variables.

Goulet Coulombe et al. (2022) described ML has a long history in econometrics. As earlier as Lee et al. (1993) proposed a new test - neural network test for testing neglected nonlinearity in time series models by comparing neural network methods with alternative tests (White dynamic information matrix test, the McLeod-Li test, the Ramsey RESET test, the Brock-Dechert-Scheinkman test, and the Bispectrum test). They found their results suggest that the neural network test can play a valuable role in evaluating model adequacy. Then, Breiman (2001b) proposed the two cultures in the use of statistical modelling to reach conclusions from data. The statistics community has, by and large, accepted the machine learning (ML) revolution that Breiman refers to as the algorithm modelling culture, and many textbooks discuss ML methods alongside more traditional statistical methods (e.g., (Hastie et al., 2009; Hartford et al., 2016).

In the macro forecasting area, Moshiri and Cameron (2000) compared the performance of Back-Propagation Artificial Neural Network (BPN) models with the traditional econometric approaches to forecasting the inflation rate. Their results show the hybrid BPN models can forecast as well as all the traditional econometric methods, and outperform them in some cases. Nakamura (2005) evaluated the usefulness of neural networks for inflation forecasting, their results especially suggest that the early stopping procedure contributes considerably to the predictive success of the NN approach and should be incorporated into future forecasting experiments involving NNs. Choudhary and Haider (2012) assessed the power of diverse Artificial neural network (ANN) models as forecasting tools for monthly inflation rates for 28 Organization for

Economic Co-operation and Development (OECD) countries. They develop arithmetic combinations of several ANN models and find that these may also serve as credible tools for forecasting inflation.

Sermpinis et al. (2014) introduced a hybrid genetic algorithm–support vector regression (GA-SVR) model in economic forecasting and macroeconomic variable selection. The proposed machine learning algorithm is applied to the task of forecasting US inflation and unemployment. Their results imply the proposed GA-SVR algorithm outperforms all benchmark models. Milunovich (2020) compared the accuracy of forecasting Australia's real house price index by applying machine learning methods and traditional time series models. They provide evidence on forecasts generated by deep learning nets rank well across medium and long forecast horizons.

With the application of machine learning in macroeconomic forecasting, Yoon (2021) denoted forecasting macroeconomic data, such as real GDP growth, is not a simple process. To forecast data, considering the causal relationship between the dependent variable and independent variable, traditional economic forecasting models require predetermined relevant variables to make predictions and often take top-down and theory-driven approaches. Goulet Coulombe et al. (2022) referred to as ML also has successfully applied in microeconomic applications attributable to the availability of large data sets. Mullainathan and Spiess (2017) indicated the process of macroeconomic forecasting also requires economic intuition and judgement by forecasters regarding the data and methods used. If there is any flaw in the assumptions made by the forecasters, the models could produce inaccurate predictions. Varian (2014) emphasised conventional statistical and econometric techniques such as regression often work well, but there are issues unique to big datasets that may require different tools. By contrast, machine learning models mostly deal with pure prediction tasks in contrast to many traditional economic forecasting models.

Yoon (2021) highlighted machine learning models are more flexible than traditional economic forecasting models and can produce predictions without predetermined assumptions or judgements. As Jung et al. (2018) proposed a key advantage of ML is that ML views empirical analysis as "algorithms" that estimate and compare many alternative models. This approach contrasts with economics, where (in principle, though rarely in reality) the researcher picks a model based on principles and estimates it once. Instead, ML algorithms build in "tuning" as part of the algorithm. The tuning is essentially model selection, and in an ML algorithm that is data-driven.

4.3 Methodology

We propose a hybrid approach that combines two machine learning models—support Vector Regression (SVR) and Random Forest—with two deep learning models—deep

Neural Network (DNN) and Long Short-Term Memory (LSTM). All four models are supervised learning algorithms that analyze training data to construct predictive functions for new, unseen data.

4.3.1 Hyperparameters Selection and Tuning Strategies

Cupallari (2020) highlighted the necessity of addressing a set of hyperparameter selections before constructing and training models. Kim and Chung (2019) believed that the critical problem in machine learning is determining the hyperparameters, such as the learning rate, mini-batch size, and regularization coefficient. (Bergstra and Bengio, 2012; Young et al., 2015) identified three widely used methods for hyperparameter selection in deep learning: (1) manual search, (2) grid search, and (3) random search. This work applies a framework for optimising the hyperparameters of a deep network by using the grid search method.

• Grid search

Probst et al. (2019) proposed one of the simplest strategies is grid search, in which all possible combinations of given discrete parameter spaces are evaluated. This is typically done by creating a dictionary in which the keys are the hyperparameters, and the values are the different values to test for each hyperparameter. For example, in Fig. 4.1, we aim to identify the best combinations of batch size and learning rate. Each blue point represents a combination of batch size and learning rate. We then score each combination and find the best combination of hyperparameters by using cross-validation tests. The main strength of grid search is that it is guaranteed to find the optimal combination of parameters specified in the grid. However, its weakness is that it can be computationally expensive and time-consuming, especially when the space of the hyperparameter is large or the dataset is very large.

•Cross-validation

Learned from Kamal (2021) by using the function of `train_test_split` to create a training set and testing set. The `TimeSeriesSplit` is introduced to create walk-forward cross-validation sets when tuning the model parameters. The data will be split into training and testing datasets, where 80% of the data will be used for training and 20% of the data will be used for testing. An example of `TimeSeriesSplit` can be seen in Fig. 4.2.

We construct a grid that encompasses all possible combinations of hyperparameters. To identify the optimal hyperparameters, we apply cross-validation specifically tailored to the time series nature of our data. This method ensures that future observations are not used to inform predictions of past values. In particular, we utilise the TimeSeriesSplit technique from the Scikit-learn library, which is specifically designed for

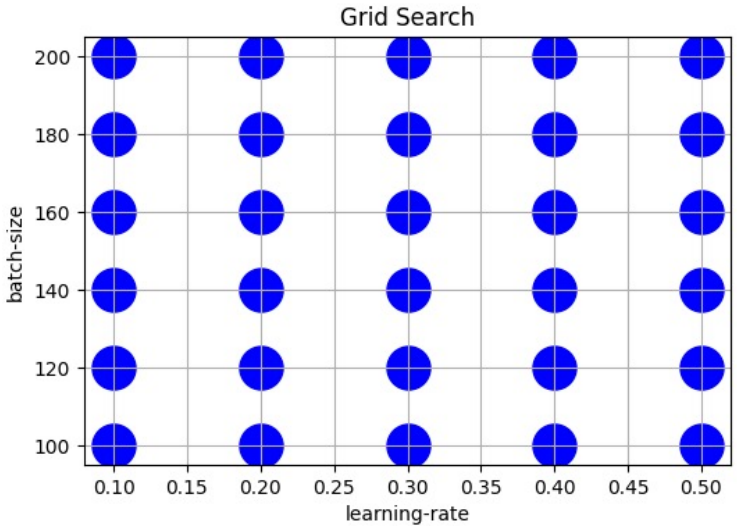

FIGURE 4.1: Time Series Split.
Note: The figure illustrates the grid search process used for hyperparameter tuning. The x-axis represents different values for the learning rate, and the y-axis represents different batch sizes. Each blue dot represents a combination of hyperparameters (learning rate and batch size) that is evaluated during the grid search process. The goal of grid search is to exhaustively search through these combinations to identify the best-performing hyperparameters for the model.

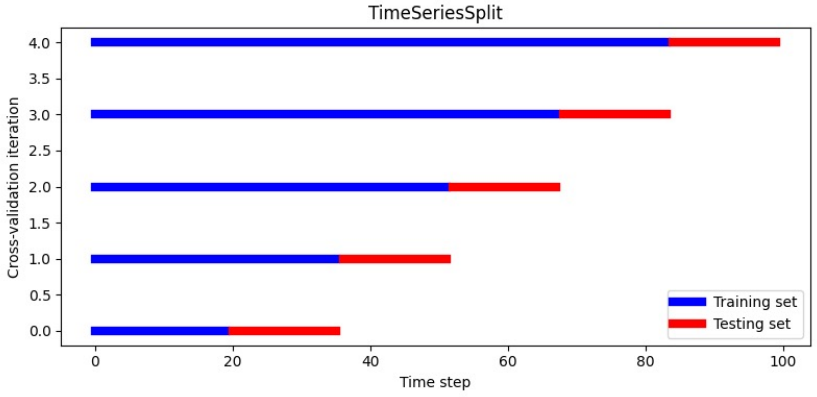

FIGURE 4.2: Roll-forward cross validation splits with TimeSeriesSplit.
Note: The figure illustrates the rolling-forward cross-validation splits using the TimeSeriesSplit method. The x-axis represents the index or time steps of the data points in the time series, progressing sequentially from left to right. The y-axis represents the different cross-validation iterations (CV Iteration), where each iteration uses a progressively larger portion of the time series for training (shown in blue) and a subsequent portion for testing (shown in red). This approach ensures that future data is not used to predict past data, maintaining the temporal order essential for time series analysis.

time-series datasets. Unlike other data types where observations can be randomly assigned to training and testing sets, time-series data is characterised by the sequential order of observations, and disrupting this sequence would compromise the integrity of the analysis.

The TimeSeriesSplit method provides a mechanism to progressively 'roll forward' data splits in accordance with the time order. It ensures that, in each iteration, the test set advances in time, incorporating all preceding data into the training set for the next iteration. Initially, the first part of the sequence is designated as the training set, with a subsequent, smaller segment assigned as the test set. In each subsequent iteration, the test set progresses forward in time by a specified number of steps, and all data up to the new test set, including the previous one, forms the new training set. This process continues until the test set spans the entire time series.

4.3.2 Machine Learning Models

This section introduces the principles underlying the four models employed in this study: Support Vector Regression (SVR), Random Forest (RF), Deep Neural Network (DNN), and Long Short-Term Memory (LSTM). The study evaluates and compares the forecasting accuracy of these machine learning and deep learning models against a linear regression benchmark, providing insights into their respective predictive capabilities.

4.3.2.1 Support Vector Regression (SVR):

Support Vector Regression (SVR) is a machine learning algorithm used for regression problems. It attempts to fit the data and find a hyperplane to minimize the gap between actual values and model predictions. Its characteristics is applicable to both linear and nonlinear regression problems, with its performance often influenced by the choice of kernel functions and hyperparameters. Its role in an ensemble, SVR can provide modelling for regression problems, especially when dealing with complex regression relationships.

To address a nonlinear regression problem, Support Vector Regression (SVR) first maps the inputs nonlinearly into a high-dimensional feature space (F) where they become linearly correlated with the outputs. The SVR formalism then uses the following linear estimation function (Vapnik, 1999):

$$f(X) = (V \cdot \Phi(X)) + b \tag{4.1}$$

Where V represents the weight vector, b is a constant,$\Phi(X)$ describes a mapping function in the feature space, and $V \cdot \Phi(X)$ denotes the dot product in the feature space F.

A number of cost functions such as the Laplacian, Huber's Gaussian, and ε-insensitive can be used in the SVR formulation. Among these, the robust ε-insensitive loss function (L_ε), given below, is the most commonly adopted (Vapnik, 1999).

$$L_\epsilon(f(X), q) = \begin{cases} |f(x) - q| - \epsilon & \text{if } |f(x) - q| \geq \epsilon \\ 0 & \text{otherwise} \end{cases} \tag{4.2}$$

The parameter ε serves as a precision measure, defining the radius of the tube surrounding the regression function $f(x)$ (depicted by the broken lines in Fig. 4.3). Fig. 4.3 provides a schematic illustration of Support Vector Regression employing the ε-insensitive loss function. The area within the tube is referred to as the "ε-insensitive zone," as the loss function assigns a value of zero within this region, thereby ignoring prediction errors with magnitudes smaller than ε.

The weight vector v and constant b in Eq. (4.1) are determined by minimising the following regularised risk function:

$$R(C) = C\frac{1}{n}\sum_{i=1}^{n} L_e(f(x_i), q_i) + \frac{1}{2}|W|^2 \tag{4.3}$$

Where $L_e(f(x), q)$ denotes the ε-insensitive loss function, as specified in Eq. (4.2); $\frac{1}{2}\sum_j w_j^2$ represents the regularisation term, which balances model complexity against approximation accuracy to ensure better generalisation performance; and C is the regularisation constant that controls the trade-off between empirical risk and the regularisation term. Both C and ε are parameters determined by the user.

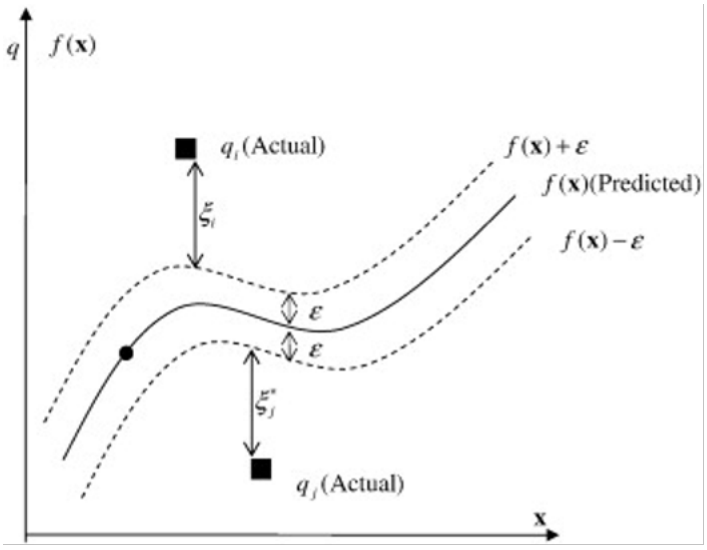

FIGURE 4.3: The SVR using ε-insensitive loss function (Yaser and Atiya, 1996).

4.3.2.2 Random Forest (RF)

The random forest model, as introduced by Breiman (2001a) presented another technique closely resembling boosting models. As noted by Dietterich (2000), the random forest stands out as one of the highly effective ensemble models in the realm of machine learning. Like the gradient boosting model, the random forest model employs regression trees. However, in contrast to the gradient boosting approach, the random forest trains regression trees independently using bootstrapped data, and the predictions from these trees are then averaged to generate the final predictions.

We follow the basic steps for the random forest model as outlined in Yoon (2021).

Step 1. For $m = 1$ to M: Generate a bootstrapped sample set, Z, of size N from the training data. Develop a random forest tree, T_m, for the bootstrapped data by performing the following steps for each terminal node of the tree until the minimum node size, n_{min}, is attained.

- Randomly select x variables from the p variables.
- Choose the optimal variable and split point among the x variables.
- Divide the node into two daughter nodes. The split is determined in a manner that minimizes the Mean Squared Error (MSE), which is calculated as follows:

$$F_0(x) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \gamma)^2 \tag{4.4}$$

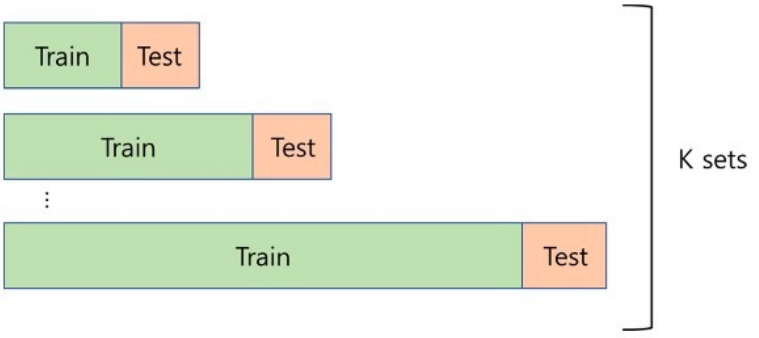

FIGURE 4.4: Cross-validation process (Yoon, 2021).

Where, y_i is an observed value and λ is a predicted value.

In addition to generating unique datasets through bootstrapping for each tree predictor, further randomness is introduced at each node by randomly selecting a subset of variables for node splitting. This stochastic process significantly reduces the interdependence among individual trees and enhances the model's ability to handle potential overfitting issues.

When a tree is allowed to grow without restrictions, it can often lead to overfitting, implying that it fits the training data perfectly but may not generalize well to new, unseen data. In other words, a model composed of nearly perfectly fitting trees might not provide accurate predictions when confronted with new data. To mitigate this problem, a random forest model may opt to prune the trees or limit the number of nodes, even if it comes at the expense of the in-sample fit.

Step 2. Output the ensemble of trees, $\{T_m\}_{m=1}^{M}$:

$$\hat{F}_{\text{rf}}^{M}(x) = \frac{1}{M} \sum_{m=1}^{M} T_m(x) \tag{4.5}$$

The final prediction, denoted as, $\hat{F}_{\text{rf}}^{M}(x)$ is derived by calculating the mean of the outputs generated by all individual decision trees in the forest. By aggregating multiple predictions, this approach effectively reduces the overall variance, thereby improving the robustness and consistency of the model's predictive performance.

4.3.2.3 Deep Neural Network (DNN):

DNN is a powerful model used for complex nonlinear modelling and feature learning. They can adapt to various types of data. Its characteristics is can automatically extract features from data and model them through multiple layers. Their performance is often affected by factors like network structure, activation functions, and regularization.

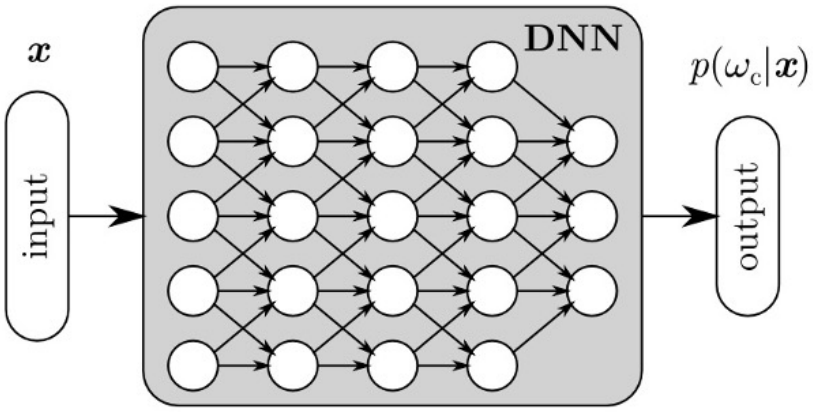

FIGURE 4.5: An example of a neural network is one made up of numerous interconnected neurons, which assigns a probability to the input x being linked to a specific concept ω_c, by (Montavon et al., 2018).

Its role in an ensemble is it can offer highly flexible modelling capabilities, especially excelling in recognising complex patterns and features.

As the description of (Montavon et al., 2018), in a DNN model, the typically abstract concept that needs interpretation is often symbolised by a neuron situated in the top layer. These top-layer neurons are inherently abstract, meaning they cannot be visually examined. Conversely, the input domain of the DNN, such as an image or text, is typically something that can be understood and interpreted. Below, we explain the process of constructing an interpretable prototype within this input domain, which serves as a representative of the abstract concept learned by the model. This prototype construction can be framed within the activation maximization framework.

Activation maximization is an analytical method that seeks an input pattern which elicits the highest possible response from a model for a specific quantity of interest (Simonyan et al., 2013; Erhan et al., 2009; Berkes and Wiskott, 2006). Consider a DNN classifier that maps data points x to a set of classes $(\omega_c)_c$. The output neurons represent the estimated class probabilities $p(\omega_c|x)$. To find a prototype x that is representative of the class ω_c, one can optimize the following:

- Activation maximization (AM)

$$\max_{\mathbf{x}} \log p(\omega_c \mid \mathbf{x}) - \lambda \|\mathbf{x}\|^2 \tag{4.6}$$

To derive more meaningful prototypes, the regularizer can be substituted with a more advanced one, referred to as an "expert" (Nguyen et al., 2017; Mahendran and Vedaldi, 2015). The expert could be, for instance, a model $p(x)$ of the data. This results in a new optimization problem:

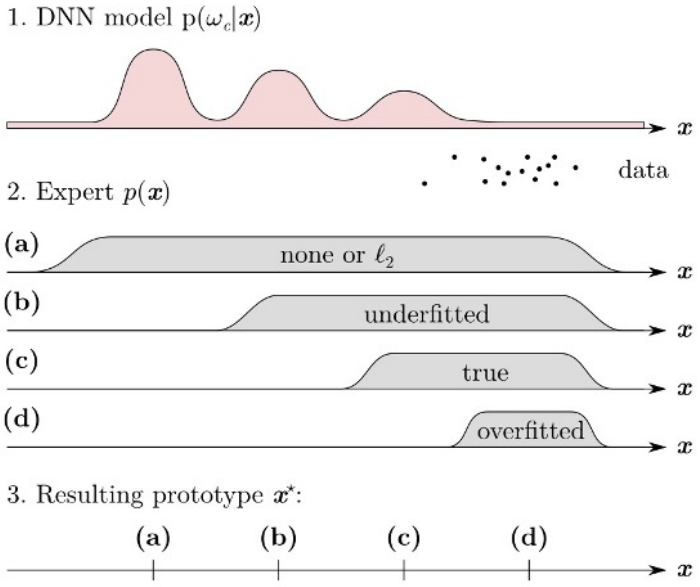

FIGURE 4.6: An illustration of how the choice of expert p(x)affects the prototype x - found by AM. The horizontal axis represents the input domain, (Montavon et al., 2018).

- Improving AM with an expert

$$\max_x \log p(\omega_c|x) + \log p(x) \tag{4.7}$$

A viable choice for the expert is the Gaussian RBM (Nguyen et al., 2017). It is capable of representing complex distributions and has a gradient in the input domain. Its log-probability function can be expressed as:

The terms $f_j(x) = \log(1 + \exp(\omega_j x + b_j))$ are learned from the data and are combined with the original L_2-norm regularization. More complex density models like convolutional RBMs/DBMs (Nguyen et al., 2017) or pixel-RNNs (Van Buuren, 2018) can be utilised for interpreting concepts such as natural image classes. In practice, the choice of the expert $p(x)$ significantly influences the appearance of the resulting prototype. Fig. 4.6 demonstrates the dependence of the prototype on the chosen expert.

$$\log p(x) = \sum_j f_j(x) - \lambda \|x\|^2 + \text{cst} \tag{4.8}$$

When using Activation Maximization (AM) to validate a DNN model, it is crucial to avoid an overfitted expert (d), as it could obscure important failure modes of the DNN. Instead, a slightly under-fitted expert (b), which may simply favor images with natural colors, can be sufficient.

Conversely, when employing AM to understand a concept ω_c that the DNN correctly predicts, the primary concern should be to prevent underfitting. An underfitted expert (b) might reveal optima of $p(\omega_c|x)$ that are far from the actual data, resulting in a prototype x that does not accurately represent ω_c. Therefore, in this scenario, it is essential to learn a density model that closely approximates the true data distribution (c).

4.3.2.4 Long Short-Term Memory (LSTM):

LSTM is a deep learning model used for processing sequence data, particularly adept at capturing long-term dependencies in time series and sequential data. Its characteristics is suitable for handling sequence data like text, time series, and speech. It can capture temporal information and sequence patterns. It's role in an ensemble, LSTM can provide modelling for sequence data, especially when considering temporal relationships in problems.

Data mining techniques are utilised to extract valuable insights from large datasets and present them in easily interpretable visualizations. Decision trees, introduced in the 1960s, stand out as one of the most effective methods among these techniques. They have been widely adopted across various fields. According to Hastie et al. (2009), decision trees are favoured for their user-friendliness, lack of ambiguity, and robustness, even when dealing with missing values. Moreover, discrete and continuous variables can be employed as target or independent variables (Song and Ying, 2015).

One-step-ahead prediction in financial time series necessitates the most recent data and the preceding data. Thanks to the self-feedback mechanism within the hidden layer, the RNN model holds an advantage in addressing long-term dependency challenges. However, Bengio et al. (1994) indicated its practical application has posed difficulties. To tackle the problem of vanishing gradients in RNNs, Hochreiter and Schmidhuber (1997) introduced the LSTM model in their research, which has more recently been enhanced and popularised by Alex Graves (Graves, 2013). The LSTM unit comprises a memory cell that stores information and is updated by three distinct gates: the input gate, the forget gate, and the output gate. The structural layout of an LSTM unit is depicted in Fig. 4.7.

At time t, x_t is the input data of the LSTM cell, h_{t-1} is the output of the LSTM cell at the previous moment, c_t is the value of the memory cell, h_t is the output of the LSTM cell. The calculation process of the LSTM unit can be divided into the following steps, as outlined by (Cao et al., 2019).

(1) First, calculate the value of the candidate memory cell $\tilde{c}_t$, where w_c is the weight matrix, and b_c is the bias.

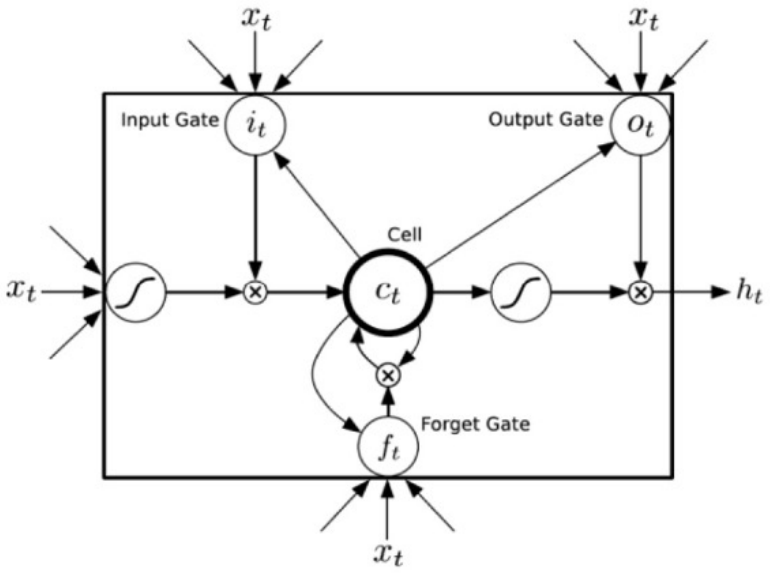

FIGURE 4.7: LSTM unit structure, Cao et al. (2019).

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{4.9}$$

(2) Calculate the value of the input gate i_t, where the input gate controls the update of the current input data to the state value of the memory cell. σ is the sigmoid function, w_i is the weight matrix, and b_i is the bias.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{4.10}$$

(3) Calculate the value of the forget gate f_t, where the forget gate controls the update of the historical data to the state value of the memory cell. W_f is the weight matrix, and b_f is the bias.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{4.11}$$

(4) Calculate the value of the current moment memory cell (c_t), and $c_(t-1)$ is the state value of the last LSTM unit.

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \tag{4.12}$$

Where "*" represents the dot product. The update of the memory cell depends on the state value of the last cell and the candidate cell, and it is controlled by the input gate and forget gate.

(5) Calculate the value of the output gate o_t, where the output gate controls the output of the state value of the memory cell. W_0 is the weight matrix, and b_0 is the bias.

$$o_t = \sigma(W_0 \cdot [h_{t-1}, x_t] + b_0) \tag{4.13}$$

(6) Finally, calculate the output of LSTM unit (h_t).

$$h_t = o_t \cdot \tanh(c_t) \tag{4.14}$$

Overall, their common characteristics are that these models can learn complex data relationships. They can be improved through hyperparameter tuning and feature engineering. All models can be used for regression tasks, but they may excel in different scenarios.

4.3.3 Objective Function

Briefly, the definition of objective function is the function that is ultimately to be predicted. In this paper, we study the objective function, which is composed by formulating the optimization problem to minimize the error between the expert-predicted value and the true value in the training data, along with a regularization term for each evaluation.

4.3.3.1 Loss Function

The loss function measures the performance of each prediction model mentioned in Section 4.2.2. As Wang et al. (2020) described, the loss function plays an important role in constructing machine learning algorithms and improving their performance. It is a crucial component that serves as an index, measuring the performance of our prediction models by quantifying the differences between predicted values and true values. In the following section, we investigate the role of the loss function, particularly in the context of its application to predictions, shedding light on its significance in enhancing the accuracy and effectiveness of machine learning algorithms.

Bickel and Doksum (2015) indicated that the Mean Square Error (MSE) is one of the most common regression loss functions, which refers to the mean value of the squared deviations of the predictions from the true values, that is given by:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{4.15}$$

Where a vector of n predictions is generated from a sample of n data points on all variables, and Y is the vector of observed values of the variable being predicted, with $\hat{Y}$ being the predicted values. Hyndman and Koehler (2006) mentioned that MSE is more sensitive to outliers than MAE.

Mean Absolute Error (MAE) is a measure of errors between paired observations expressing the same phenomenon. It describes the average model-performance error examined. Willmott and Matsuura (2005) indicated that MAE is a more natural measure of average error.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4.16}$$

It is an arithmetic average of the absolute errors $|e_i| = |y_i - \hat{y}_i|$, where y_i is the predicted value and $\hat{y}_i$ is the true value. Wang et al. (2020) indicated absolute loss is more robust than square loss when there are outliers in the training set, and absolute loss should be selected when outliers are detected and have an impact on the learning of the model.

The Huber loss function is defined by (Huber, 1992; Wang et al., 2020) indicated that this loss is a piecewise function of square loss and absolute loss. Huber loss uses the parameter as the boundary to judge whether it is a more singular sample. The samples within this boundary use square loss, and the samples beyond this boundary use absolute loss, to reduce the weight of the loss of outliers in the total loss and avoid the model overfitting.

$$\text{Huber Loss} = \frac{1}{n}\sum_{i=1}^{n}\begin{cases} \frac{1}{2}(y_i - \hat{y}_i)^2, & \text{if } |y_i - \hat{y}_i| \leq \sigma \\ \delta\left(|y_i - \hat{y}_i| - \frac{1}{2}\delta\right), & \text{otherwise} \end{cases} \tag{4.17}$$

As Vovk (2015) mentions the log loss function is the standard loss function used in the literature on probabilistic prediction

$$\text{Log Loss} = -\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right) \tag{4.18}$$

The R^2 score is the coefficient of determination, it is a very important metric that is used to evaluate the performance of a regression-based machine learning model. It is also known as the coefficient of determination. It works by measuring the amount of variance in the predictions explained by the dataset. If the value of the r-squared score is 1, it means that the model is perfect, and if its value is 0, it means that the model will perform badly on an unseen dataset. This also implies that the closer the value of the r-squared score is to 1, the more perfectly the model is trained. It is calculated as:

$$R^2 \text{ score} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{4.19}$$

In statistics, it is used to determine the proportion of variation in the dependent variable that is explained by the independent variables, thereby assessing the explanatory power of the regression model Draper and Smith (1998).

4.3.4 Penalised Estimation

Overfitting occurs when a model performs worse on test data compared to training data. Common reasons include the model being too complex, insufficient training data, or the sample having too many characteristics. One solution to overfitting is the use of Lasso and Ridge regularization techniques (Pereira et al., 2016).

Method 1: L1 (Lasso)

For the linear regression model $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \xi$, if only some independent variables are significant, you can use Lasso regression for variable selection, removing non-significant variables to prevent overfitting and improve the model's predictive accuracy.

For the solution of linear regression models, the ordinary least squares (OLS) involve minimising the sum of squared residuals:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_{i1} - \ldots - \beta_p x_{ip})^2 \tag{4.20}$$

Lasso regression minimizes the least square's objective function with the addition of an L1 penalty:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_{i1} - \ldots - \beta_p x_{ip})^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{4.21}$$

Then, the optimal tuning parameter λ is obtained by minimising the generalised cross-validation:

$$\hat{\lambda} = \arg\min_{\lambda} \left(\frac{1}{n} \frac{\|Y - X(\hat{\beta}_\lambda)\|_2}{\left(1 - \frac{\mathrm{df}(\lambda)}{n}\right)^2} \right) \tag{4.22}$$

Method 2: L2 (Ridge)

L2 (Ridge) for the linear regression model $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \xi$ when there is complete multicollinearity between the variables $X_1, \ldots, X_p$. In such cases, when the regression coefficient β has no solution, one can consider ridge regression. This method involves introducing a certain bias to reduce variance, resulting in a biased estimate of the regression coefficients.

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_{i1} - \ldots - \beta_p x_{ip})^2 \tag{4.23}$$

Ridge regression minimizes the least square's objective function with the addition of an L2 penalty:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_{i1} - \ldots - \beta_p x_{ip})^2 + \lambda\sum_{j=1}^{p}\beta_j^2 \tag{4.24}$$

To obtain the minimum value mentioned above, we can express it as how to express it in mathematics:

$$\hat{\beta}_{\text{ridge}} = (X'X + n\lambda I_p)^{-1}X'Y \tag{4.25}$$

where $\lambda \geq 0$ is the adjustment of parameters. With an increase in λ, to make the objective function reach a minimum, it is evident that reducing the value of b will compress it towards zero. The optimal value for x can be selected through cross-validation.

Typically, obtaining the optimal tuning parameter λ involves minimising the generalised cross-validation:

$$\hat{\lambda} = \arg\min_{\lambda}\frac{1}{n}\frac{\|(I_n - H(\lambda))Y\|_2}{\left[\frac{\text{tr}(I_n - H(\lambda))}{n}\right]^2} \tag{4.26}$$

Where, $H(\lambda) = (X'X + n\lambda I_p)^{-1}X'$, then we solve (4.21), it is equivalent to solving the constrained least squares:

$$\begin{aligned}&\min_{\beta}\frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_{i1} - \ldots - \beta_p x_{ip})^2\\ &\text{s.t.}\quad \sum_{j=1}^{p}\beta_j^2 \leq c\end{aligned} \tag{4.27}$$

With $c \to 0$, the constraint on the regression coefficients becomes stronger.

4.3.5 Evaluation Metrics:

We evaluate the performance of our ensemble model using standard regression metrics, focusing on the Mean Absolute Error (MAE). Chai and Draxler (2014) highlighted that both the root mean square error (RMSE) and MAE are commonly used to assess model accuracy. RMSE is often favoured for its sensitivity to larger errors, which makes it useful when outliers are particularly important. However, Willmott and Matsuura (2005) argued that RMSE may not effectively represent the average performance of a model, as it disproportionately emphasizes larger deviations, potentially leading to misleading conclusions when assessing overall accuracy. In contrast, MAE provides a more balanced measure of average errors by giving equal weight to all differences between predicted and actual values, making it a more suitable metric for evaluating model performance in many scenarios.

4.3.6 Model Generative Process

Tables 4.1 to 4.4 present the generative processes of four models: SVR, RF, DNN, and LSTM. Each table details the initialization of parameters, the training process, and the evaluation method for each model. SVR employs ϵ-insensitive loss and quadratic programming for optimization, while the Random Forest model generates multiple decision trees and aggregates their predictions to reduce overfitting and enhance accuracy. Meanwhile, using error gradients, DNN and LSTM adjust weights through forward and backward passes. These tables comprehensively overview how each model processes various data types and addresses prediction tasks.

TABLE 4.1: The generative process of our SVR model

1. Initialize model parameters:
   - Weight vector $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.
   - Bias term $b \sim \mathcal{N}(0, 1)$.
2. For each training data point $i = 1, \ldots, N$:
   (a) Map the input features $\mathbf{x}_i$ to a higher-dimensional feature space using a mapping function $\phi(\mathbf{x}_i)$.
   (b) Compute the linear function output $f(\mathbf{x}_i) = \mathbf{w}^\top \phi(\mathbf{x}_i) + b$.
   (c) Calculate the ϵ-insensitive loss $L_\epsilon(y_i, f(\mathbf{x}_i))$:
      – If $|y_i - f(\mathbf{x}_i)| \leq \epsilon$, the loss is 0.
      – Otherwise, the loss is $|y_i - f(\mathbf{x}_i)| - \epsilon$.
3. Minimize the regularised risk $R(C)$ to update model parameters:
   (a) Minimize $\frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{i=1}^{N} L_\epsilon(y_i, f(\mathbf{x}_i))$.
   (b) Use optimization techniques such as quadratic programming to solve for optimal $\mathbf{w}$ and b.
4. Solve the dual problem to find optimal Lagrange multipliers α_i and α_i^*:
   (a) Minimize the dual objective function with constraints: $0 \leq \alpha_i, \alpha_i^* \leq C$ and $\sum_{i=1}^{N} (\alpha_i - \alpha_i^*) = 0$.
   (b) Use the kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ to compute inner products in the high-dimensional space.
5. Make predictions for a new input $\mathbf{x}^*$:
   (a) Calculate $\hat{y}^* = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) K(\mathbf{x}_i, \mathbf{x}^*) + b$.
   (b) Output $\hat{y}^*$ as the predicted value for the new input.

TABLE 4.2: The generative process of our RF model

1. Initialize parameters:

- Number of trees T in the forest.
- Number of features m to consider at each split.
- Maximum depth of each tree $d_{\max}$.

2. For each tree $t = 1, \ldots, T$:

(a) Draw a bootstrap sample $\mathcal{D}_t$ of size n from the training data.

(b) Grow a decision tree $h_t(\mathbf{x})$ from the bootstrap sample $\mathcal{D}_t$:

- For each node, randomly select m features from the d features.
- Choose the best split from the m features based on a certain criterion (e.g., Gini impurity or entropy).
- Repeat until the maximum depth $d_{\max}$ is reached or the node cannot be split further.

3. Aggregate the predictions of all trees:

(a) For regression: Compute the final prediction $\hat{y}$ by averaging the predictions of all trees:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^{T} h_t(\mathbf{x}).$$

(b) For classification: Compute the final prediction $\hat{y}$ by majority voting among the predictions of all trees.

4. Evaluate the Random Forest model:

(a) Calculate performance metrics such as Mean Squared Error (MSE) for regression or accuracy for classification.

(b) Use Out-of-Bag (OOB) samples to estimate the generalization error of the model.

TABLE 4.3: The generative process of our DNN model

1. Initialize the network architecture:

- Number of layers L and number of neurons in each layer n_l for $l = 1, \ldots, L$.
- Activation functions σ_l for each layer l (e.g., ReLU, Sigmoid, Tanh).
- Learning rate η for gradient descent optimization.

2. Initialize weights and biases:

(a) Initialize weights $\mathbf{W}_l \sim \mathcal{N}(0, \frac{1}{\sqrt{n_l}})$ and biases $\mathbf{b}_l = 0$ for each layer l.

3. For each training epoch:

(a) For each training example $(\mathbf{x}_i, y_i)$:

- Forward pass: Compute the output of each layer using the activation function:

$$\mathbf{a}_l = \sigma_l(\mathbf{W}_l \mathbf{a}_{l-1} + \mathbf{b}_l), \quad l = 1, \ldots, L.$$

- Compute the loss (e.g., Mean Squared Error for regression or Cross-Entropy Loss for classification).

(b) Backward pass: Calculate the gradients of the loss with respect to the weights and biases using backpropagation.

(c) Update the weights and biases using gradient descent:

$$\mathbf{W}_l := \mathbf{W}_l - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{W}_l}, \quad \mathbf{b}_l := \mathbf{b}_l - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{b}_l}.$$

4. Evaluate the DNN model:

(a) Calculate performance metrics such as Mean Squared Error (MSE) for regression or accuracy for classification on validation data.

(b) Adjust hyperparameters (e.g., learning rate, number of neurons) based on performance.

TABLE 4.4: The generative process of our LSTM model

1. Initialize the LSTM architecture:

- Number of LSTM layers L and number of units in each layer n_l for $l = 1, \ldots, L$.
- Learning rate η and sequence length T.

2. Initialize weights and biases for each gate:

(a) Initialize weights $\mathbf{W}_f, \mathbf{W}_i, \mathbf{W}_o, \mathbf{W}_c$ and biases $\mathbf{b}_f, \mathbf{b}_i, \mathbf{b}_o, \mathbf{b}_c$ for the forget, input, output gates, and cell state, respectively.

3. For each training epoch:

(a) For each training sequence $(\mathbf{x}_{1:T}, \mathbf{y}_{1:T})$:

- Forward pass for each time step $t = 1, \ldots, T$:
  - Forget gate: $f_t = \sigma(\mathbf{W}_f[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f)$.
  - Input gate: $i_t = \sigma(\mathbf{W}_i[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i)$.
  - Output gate: $o_t = \sigma(\mathbf{W}_o[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_o)$.
  - Cell state: $\tilde{c}_t = \tanh(\mathbf{W}_c[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_c)$.
  - Update cell state: $c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$.
  - Compute hidden state: $\mathbf{h}_t = o_t \odot \tanh(c_t)$.
- Compute the loss based on the output $\mathbf{h}_t$ and the true output $\mathbf{y}_t$.

(b) Backward pass: Calculate the gradients using Backpropagation Through Time (BPTT).

(c) Update the weights and biases using gradient descent.

4. Evaluate the LSTM model:

(a) Calculate performance metrics such as Mean Squared Error (MSE) for regression or accuracy for classification on validation data.

(b) Adjust hyperparameters (e.g., learning rate, number of units) based on performance.

4.4 Dataset Construction

4.4.1 Sample Variables

We focus on predicting five key macroeconomic indicators of the UK economy: GDP growth (GDP), Inflation rate (Inflation), Base Bank rate (BoE), Unemployment rate (UR), and Sterling exchange rate Index (ERI). We collect data from the Bank of England's survey of professional forecasts, which includes predictions from 56 experts for each variable, providing quarterly values for one-year ahead expectations on the five main UK macroeconomic indicators.

Additionally, we gather historical data for these five indicators to evaluate and compare with professional forecasts. Table 4.5 provides basic statistics for the raw data input. The Bank of England collects survey data from a total of 56 forecasters. We apply a rule to filter out experts who provide responses relatively less frequently, with a response rate below 50%. Subsequently, we perform data imputation to address missing values in the data series for these selected experts.

TABLE 4.5: Descriptive statistics for data sample.

Variable	Total Experts	Counted Experts (%)		Sample period	Observations	
		No.	%		Original	with imputation
GDP	56	25	44.64	2000Q1:2022Q4	1615	2200
Inflation	56	25	44.64	2000Q1:2022Q4	1611	2200
Unemployment	56	20	35.71	2014Q4:2022Q4	400	640
BoE	56	24	42.86	2000Q1:2022Q4	1534	2112
ERI	56	15	26.79	2000Q1:2022Q4	922	1320
Total	280	109	39.29	-	6084	8472

Note: We exclusively consider data from experts whose response rate remained above 50% for the entire period from 2000: Q4 to 2022: Q4 for each variable. The original observation number denotes the count of included experts. We also present the amount of data after processing the data imputation.

4.4.2 Data Preparation

Kotsiantis et al. (2006) mentioned that the issue of incomplete data is an unavoidable problem when dealing with most real-world data sources. Emmanuel et al. (2021) presented that missing values are usually attributed to human error, machine error, respondent refusal to answer certain questions, dropout in studies, and merging unrelated data. Kotsiantis et al. (2006) also demonstrated that the data pre-processing can often have a significant impact on the generalization performance of a supervised ML algorithm. They summarise data pre-processing including data cleaning, normalization, transformation, feature extraction and selection, etc.

In this study, the survey data of professional forecasts has revealed a sparsity feature, with more than half of the forecasters not providing frequent responses. The presence of missing data can adversely impact the performance and accuracy of machine learning models. As indicated by Rasmussen and Bro (2012) the concept of using sparsity actively for achieving simpler models has received huge attention within fields such as statistical learning, data mining, and signal processing (Lu et al., 2009; Donoho, 2006; Tibshirani, 1996; Chen and Donoho, 1994).

To address the limitation of data sparsity and ensure that the dataset is more robust and suitable for training machine learning models to achieve the best performance For the missing data, we are employing the technique of Bayes and Multiple Imputation as discussed in Little and Rubin (2002) to process a data imputation. It utilizes a useful alternative approach to multiple imputation is to add a prior distribution for the parameters and compute the posterior distribution of the parameters of interest. In summary, data imputation is an essential pre-processing step to handle missing values and create a well-structured dataset for training machine learning models.

4.5 Results and Discussion

Figs 4.8 - 4.11 compare the true and predictive values by DNN, LSTM, RF, and SVR models for each target indicator. Each figure (b) represents additional analysis eliminating two crisis periods: the global financial crisis: 2007Q3-2009Q3 and the COVID-19 crisis: 2020Q1-2022Q1. (The start time for COVID-19 differs for each country, depending on the date of the first case for each country (Rizwan et al., 2020).

4.5.1 Prediction Accuracy

In deep learning models, as shown in Fig 4.8, DNN displays notable efficiency in predicting GDP during periods of normal economic conditions. However, it falls short of capturing the output growth direction during two crisis periods. Upon excluding these crises, the model captures the general direction of the GDP trend, albeit with reduced efficiency and without a perfect fit to the true values.

Turning to LSTM, as shown in Fig 4.9, this algorithm excels in accurately predicting UR and ERI, successfully capturing the direction of the inflation trend, and providing partial predictions for BOE. Remarkably, it anticipates the onset of the COVID crisis in GDP predictions. Furthermore, in the subsample excluding the two crises (b), LSTM showcases its ability to capture the directional trends associated with global crises.

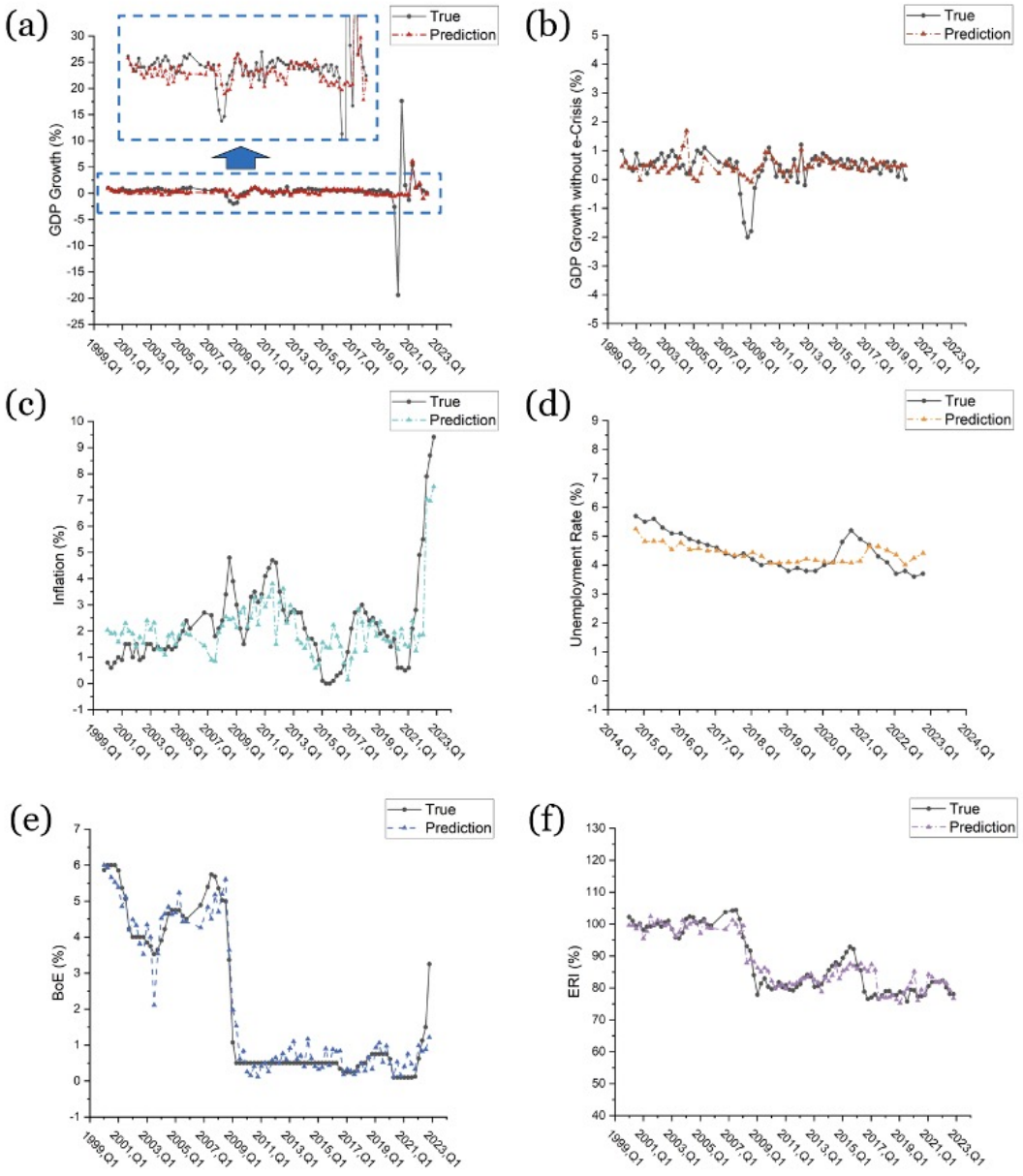

FIGURE 4.8: The comparison of DNN model predictive value and true value in each target indicator.

In machine learning models, as depicted in Fig 4.10, Random Forest (RF) stands out with a robust overall predictive ability across various indicators, including unemployment, bank rate, and ERI. Notably, RF exhibits outstanding performance in GDP forecasting, especially in data series without crises. Furthermore, in the realm of inflation forecasting, RF accurately predicts the direction of the inflation trend.

In addition, as illustrated in Fig 4.11, Support Vector Regression (SVR) also showcases an overall strong predictive ability in each target indicator, albeit slightly trailing behind RF. In GDP forecasting, SVR demonstrates a consistent predictive ability, although

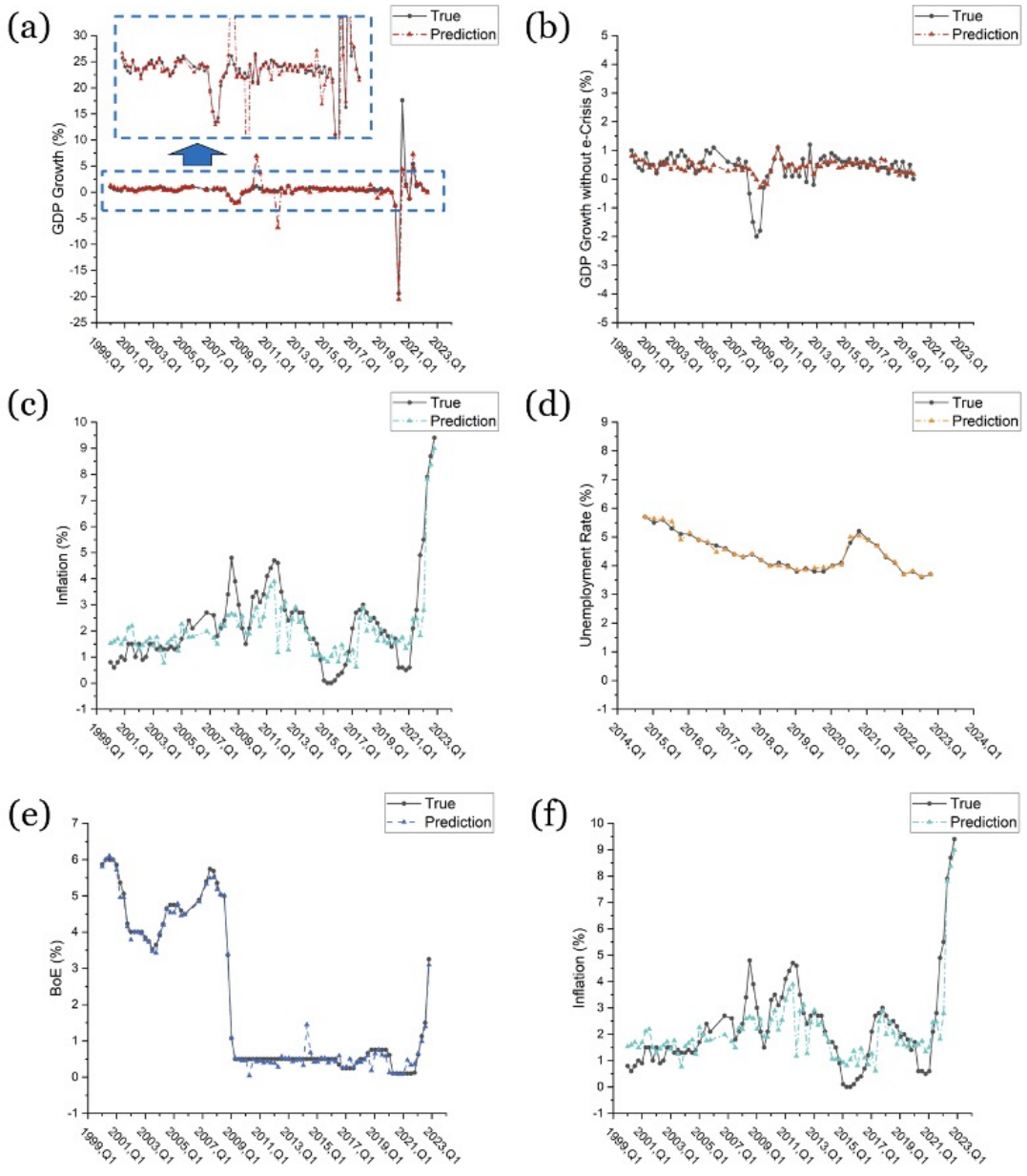

FIGURE 4.9: The comparison of LSTM model predictive value and true value in each target indicator.

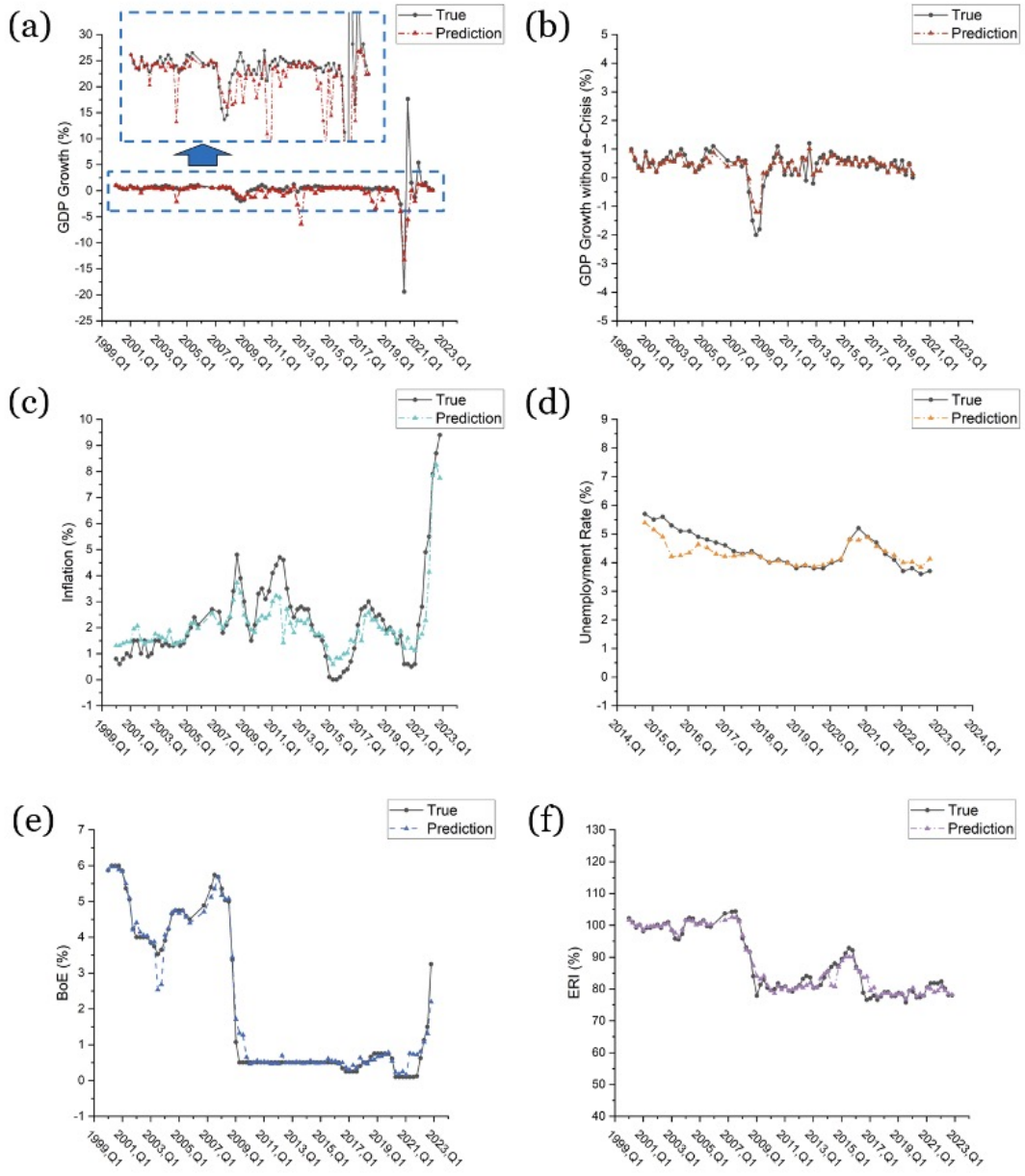

FIGURE 4.10: The comparison of RF model predictive value and true value in each target indicator.

it falls short in capturing the impact of the Covid crisis. However, SVR successfully captures a downturn trend after a data revision that removes the crisis-related data.

Overall, we found that machine learning models have stronger predictive capabilities than deep learning models, but deep learning models can be used as a complement and combined with machine learning to build optimal predictive estimators.

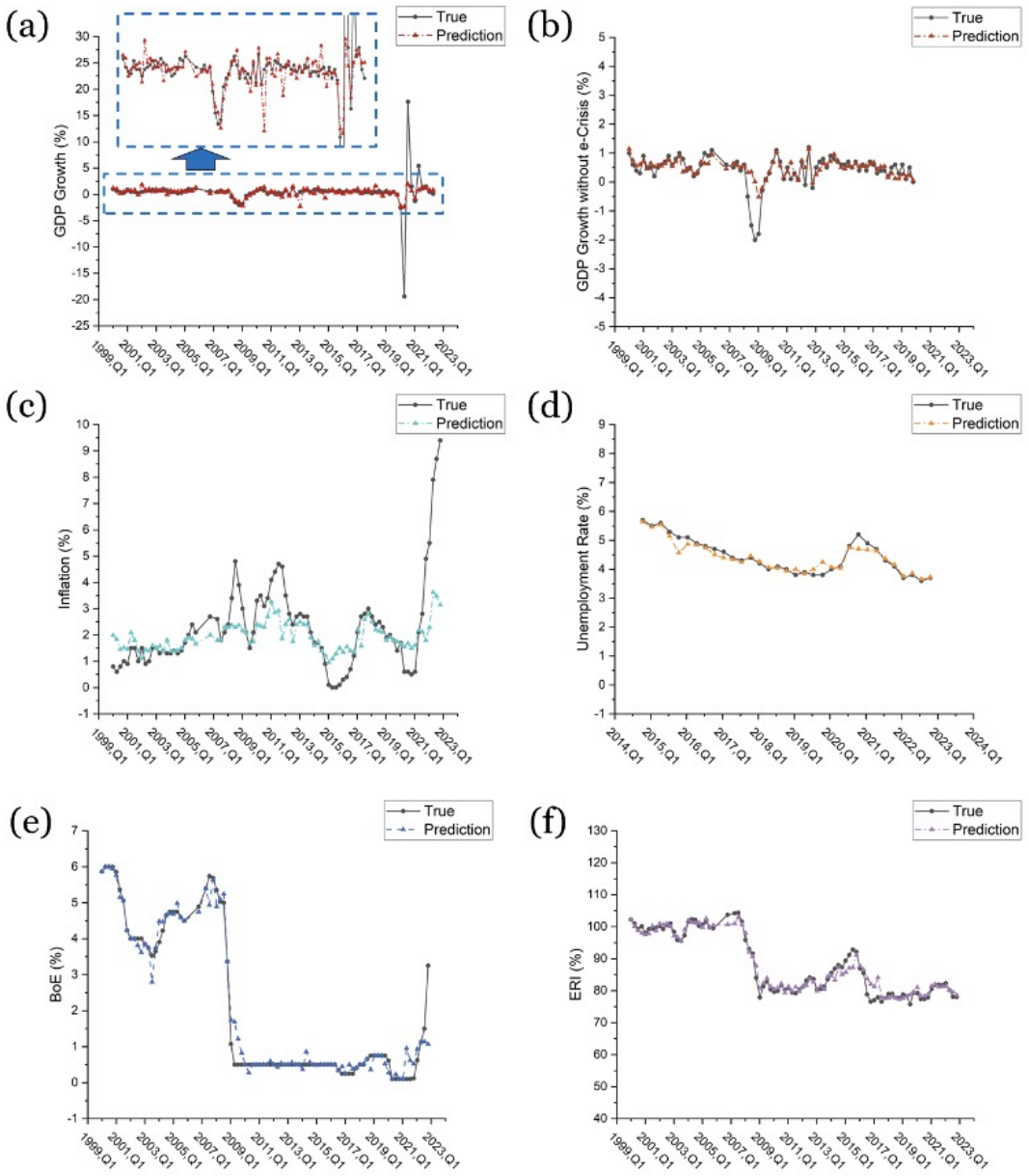

FIGURE 4.11: The comparison of SVR model predictive value and true value in each target indicator.

4.5.2 Performance Evaluation

This section will first present the details of hyperparameter selection and tuning (see details in Table 4.6 and Table 4.7). Then, It Will be followed by the figures comparing data training and testing results (see details in Fig 4.13 and Fig 4.14).

4.5.3 Hyperparameter Tuning

Table 4.6 presents the hyperparameters used by the machine learning models, which are selected by the cross-validation process. As Table 4.6 shows, since the training data receive new data for each new year, the hyperparameters change accordingly to adjust to the new data set.

TABLE 4.6: Hyperparameter Ranges for Different Models

Model	**Hyperparameters**	**Grid Search Range**
DNN	Learning Rate (lr)	0.00001, 0.0001, 0.001, 0.01, 0.1
	Lr Decay Rate	0.55, 0.65, 0.75, 0.85, 0.95
	Dropout	0.25, 0.35, 0.45, 0.55, 0.65
	Regularizer (L2)	0.000001, 0.00001, 0.0001, 0.001
	Loss Function	MSE, Huber
LSTM	Learning Rate (lr)	0.00001, 0.0001, 0.001, 0.01, 0.1
	Lr Decay Rate	0.55, 0.65, 0.75, 0.85, 0.95, 1.00
	Dropout	0, 0.4, 0.5, 0.6
	Regularizer (L2)	0.00001, 0.0001, 0.001, 0.01
	Metrics	MAE, Accuracy
	Algorithm Optimizer	Nadam, SGD
	Batch Size	16, 32, 64
	Hidden Unit	1, 2
Random Forest	No. of trees	100, 150, 200, 250, 300, 400, 500
	Max. depth of the tree	None, 11, 13, 15, 17, 19, 21, 23, 30, 40, 50, 60, 70, 80
SVR	C (Penalty Factor)	0.1, 0.5, 1, 2, 3, 4, 5, 10, 15, 20, 25, 50, 100, 200, 500, 1000
	Epsilon (epsilon-tube)	0.0001, 0.0005, 0.001, 0.005, 0.01, 0.1

Note: The optimised parameters of each model.

TABLE 4.7: Hyperparameters Test for Different Variables

Model	Hyperparameters	ERI	GDP	GDP/E	UR	BoE	Inflation
DNN	Learning Rate (lr)	0.001	0.001	0.001	0.001	0.01	0.001
	Lr Decay Rate	0.75	0.85	0.35	0.55	0.85	0.75
	Dropout	0.25	0.35	0.35	0.55	0.35	0.35
	Regularizer (L2)	0.001	0.0001	0.0001	0.00001	0.001	0.001
	Loss Function	Huber	MSE	MSE	Huber	MSE	MSE
LSTM	Learning Rate (lr)	0.1	0.001	0.00001	0.001	0.01	0.00001
	Lr Decay Rate	1	1	1	1	1	1
	Dropout	0	0	0	0.4	0	0
	Regularizer (L2)	0.001	0.001	0.01	0.00001	0.001	0.01
	Metrics	MAE	Accuracy	Accuracy	Accuracy	Accuracy	MAE
	Algorithm Optimizer	SGD	Nadam	Nadam	Nadam	Nadam	Nadam
	Batch Size	32	16	16	64	16	16
	Hidden Unit	1	1	1	1	1	2
Random Forest	No. of trees	200	150	150	100	100	150
	Max. depth of the tree	23	11	15	15	11	60
SVR	C (Penalty Factor)	4	15	5	10	2	0.5
	Epsilon (epsilon-tube)	0.1	0.1	0.1	0.1	0.0001	0.0005

TABLE 4.8: Statistics of model performance for each indicator

Indicator	Model	MSE	RMSE	MAE	Running time (s)	Scoring
GDP	DNN	1.73	1.31	0.45	0.449	0.436
	LSTM	4.28	2.07	1.03	2.486	1.067
	Random Forest	3.54	1.88	0.87	0.020	1.268
	SVR	1.58	1.26	0.59	0.001	1.070
GDP/E	DNN	0.41	0.64	0.50	0.315	1.028
	LSTM	0.42	0.64	0.59	2.127	1.038
	Random Forest	0.41	0.64	0.55	0.018	1.268
	SVR	0.40	0.63	0.56	0.001	1.149
Inflation	DNN	0.52	0.72	0.53	0.326	0.960
	LSTM	0.51	0.71	0.49	1.981	1.028
	Random Forest	0.34	0.58	0.37	0.020	2.031
	SVR	0.32	0.57	0.34	0.001	1.081
UR	DNN	0.66	0.81	0.73	0.396	0.454
	LSTM	0.07	0.26	0.21	1.005	0.817
	Random Forest	1.24	1.11	1.02	0.016	4.685
	SVR	0.17	0.42	0.33	0.001	0.982
BoE	DNN	0.03	0.17	0.14	0.350	0.403
	LSTM	0.03	0.17	0.14	2.710	0.491
	Random Forest	0.03	0.18	0.12	0.028	6.095
	SVR	0.02	0.15	0.11	0.008	0.619
ERI	DNN	0.16	0.40	0.27	0.367	0.337
	LSTM	0.05	0.23	0.17	1.744	0.754
	Random Forest	0.13	0.36	0.27	0.018	3.181
	SVR	0.08	0.28	0.20	0.001	0.726

When comparing the predictive performance of various models, particularly the distinction between Machine Learning (ML) and Deep Learning (DL), we conduct a comprehensive evaluation using four loss functions (MSE, RMSE, MAPE, and MAE), as outlined in Table 4.8. The results exhibit variability depending on the indicators and predictive models.

As details are shown in Table 4.8 and Table 4.9, the results reveal an apparent difference in the accuracy obtained by the various machine/deep learning methods across the variables and parameter settings. However, when interpreting the models' performance by variables, we found that the prediction of the bank rate demonstrates significant accuracy, and forecasting GDP growth after deleting the crisis period data also

shows a significant improvement. Similarly, machine algorithms experience success in predicting inflation, unemployment, and the exchange index.

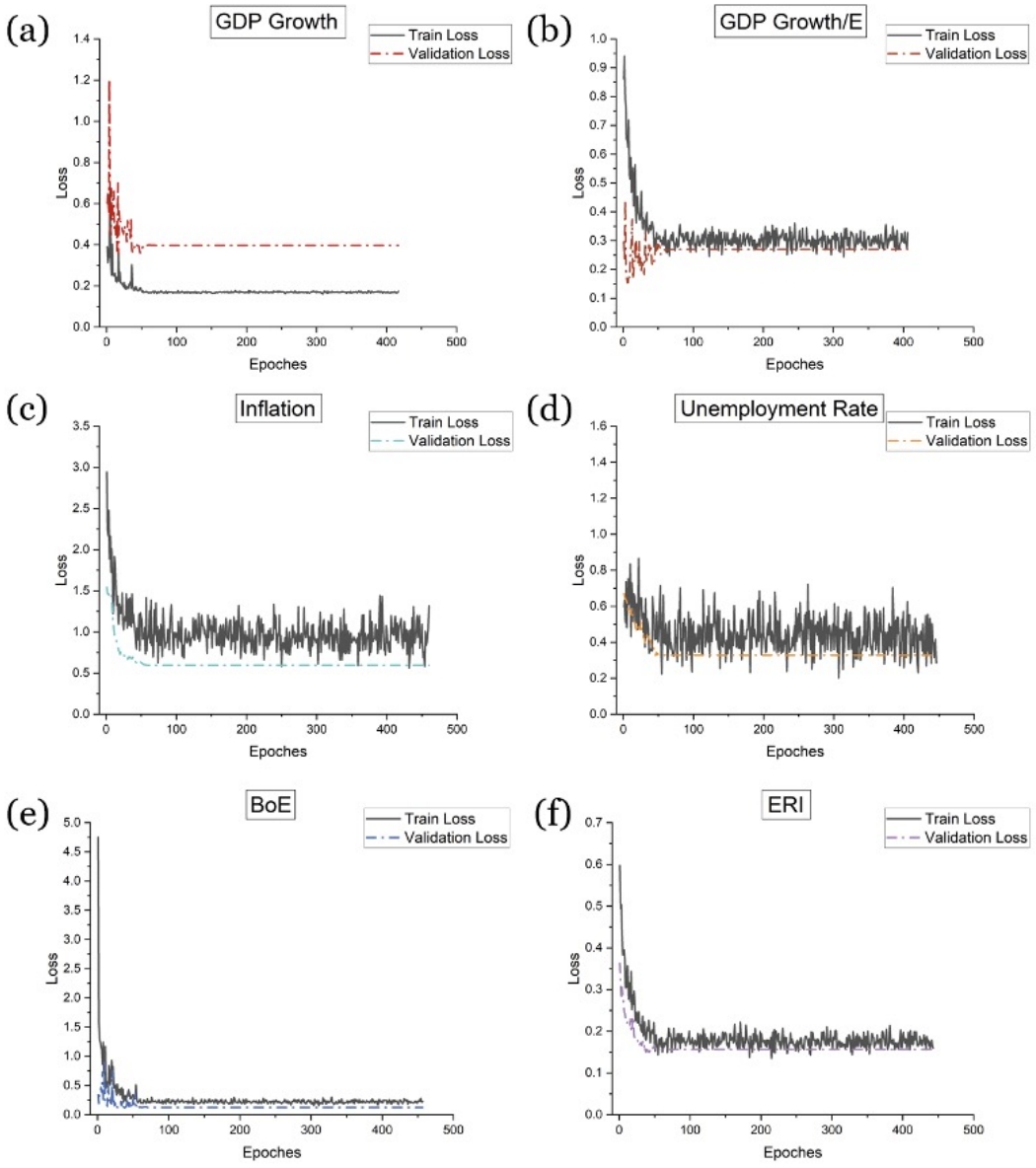

FIGURE 4.12: The comparison of train loss and test loss in DNN model.

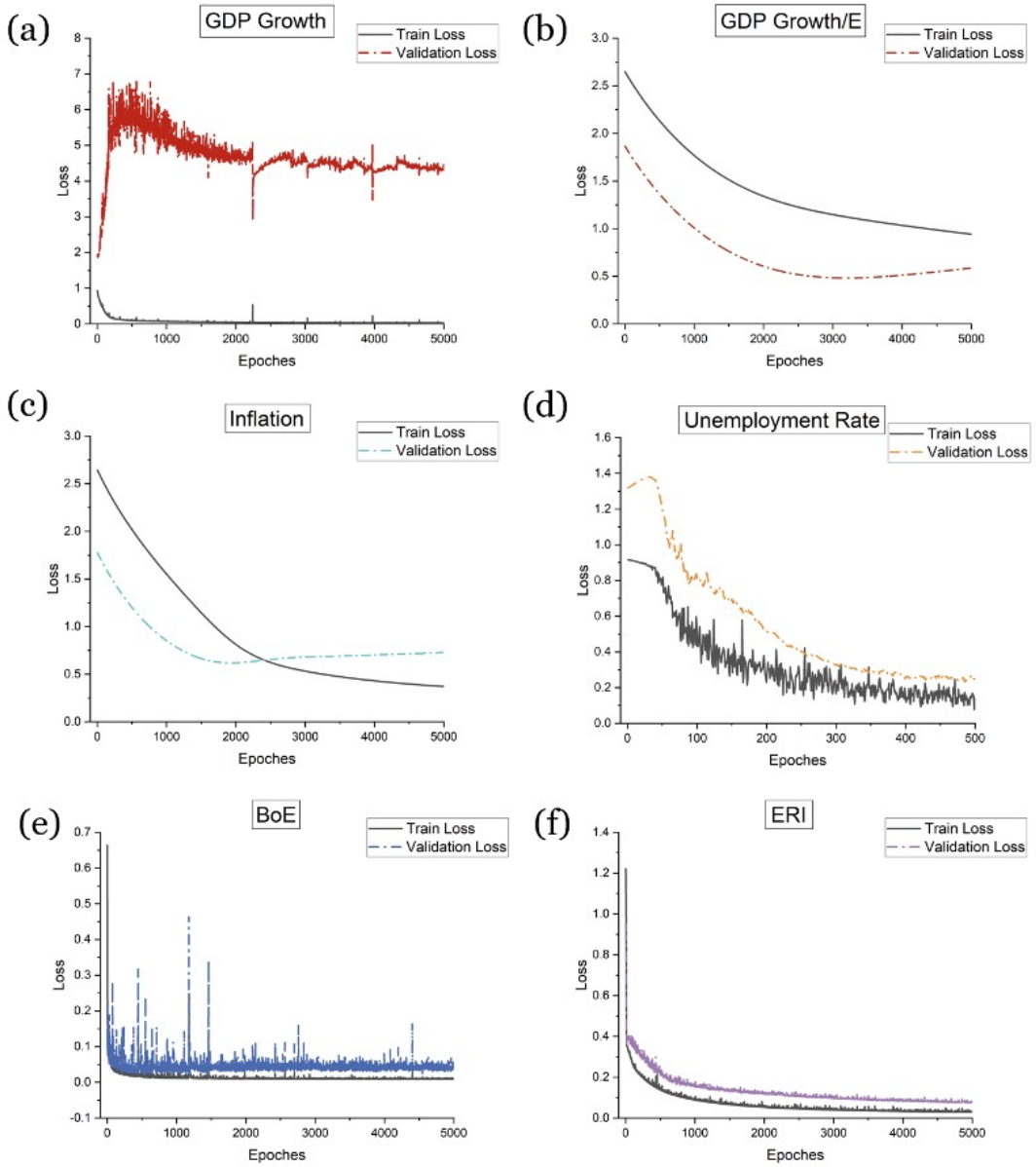

FIGURE 4.13: The comparison of train loss and test loss in LSTM model.

TABLE 4.9: Performance metrics summary

	MSE	RMSE	MAE	Running Time (s)	Scoring
Min	0.02	0.15	0.11	0.00	0.34
Max	4.28	2.07	1.03	2.71	6.10
Avg	0.71	0.67	0.44	0.60	1.37
St.dev.	1.10	0.52	0.27	0.89	1.39

Note: Accuracy of model performance obtained from all the indicators.

Additionally, Bai et al. (2022) pointed out that the time cost of models serves as a valuable evaluation index when assessing algorithms' performance. They indicate its significance in practical applications and highlight the need to precisely control the time spent on model inference to meet equipment requirements. In this study, we incorporate the time cost of each model as an additional evaluation index. A shorter time cost implies that data patterns and features are easier to identify, reflecting an efficiently applicable model. In comparison, a longer time suggests that the feature is concealed in more layers or the model is less efficient. This holds important analytical value for machines seeking to enhance predictive ability. Our results show that the time cost of machine learning models (RF and SVR) is significantly less than that of deep learning models (DNN and LSTM). Furthermore, the time cost of DNN is noticeably less than LSTM's. If we were to rank time cost from smallest to largest, the order would be: SVR – RF – DNN – LSTM.

4.5.4 Additional Analysis with Visualization

To enhance the interpretability of our findings, we've crafted violin plots (Figs 4.14 - 4.19) that illustrate the comparative data distributions across various variables under different machine models. We've also incorporated the aggregated predictions detailed in Chapter 2. Our results underscore a notable improvement in experts' prediction performance by integrating machine algorithms.

For GDP, the data distribution within the LSTM model remarkably aligns with real-world data. Examining the reconstructed data series of GDP without crises reveals a consistent and stable predictive capacity across all four models. Particularly intriguing is the observation that most machine models adeptly capture outliers in the real data in the case of inflation.

Shifting the focus to unemployment, both the LSTM and SVR models align with the data distribution observed in the real world. Noteworthy patterns emerge in bank rate and ERI domains, where the data distribution tendencies of machine models notably mirror those observed in real-world data.

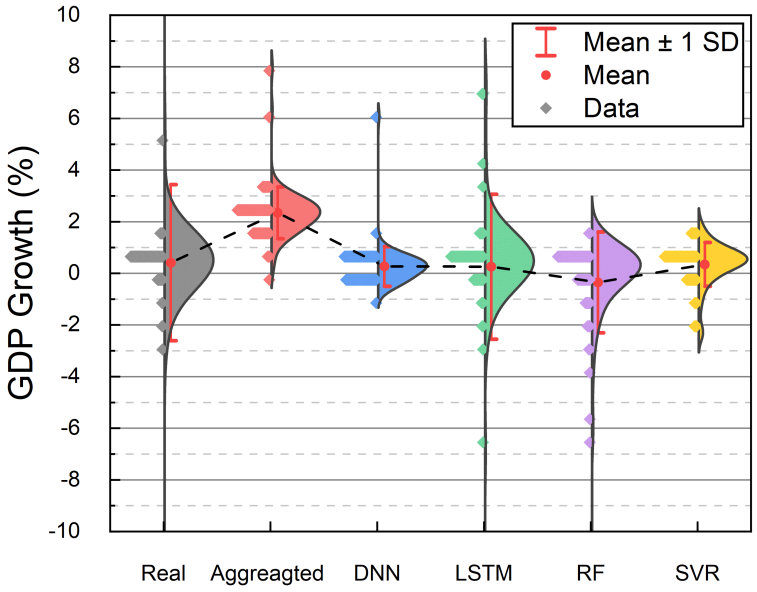

FIGURE 4.14: Comparison of Data Distributions in GDP.

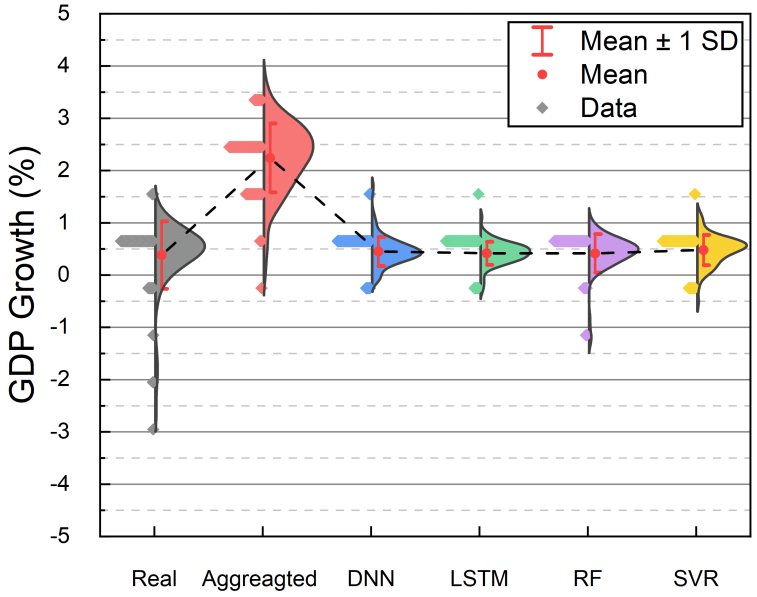

FIGURE 4.15: Comparison of Data Distributions in GDP without crisis.

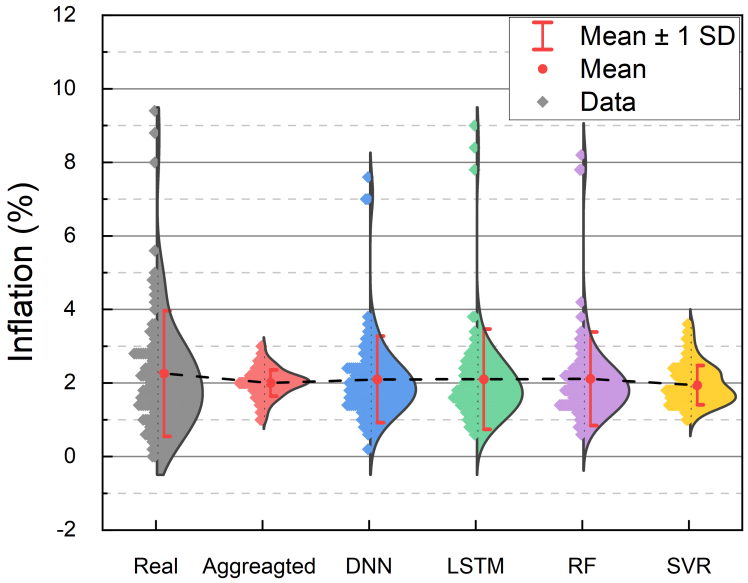

FIGURE 4.16: Comparison of Data Distributions in Inflation.

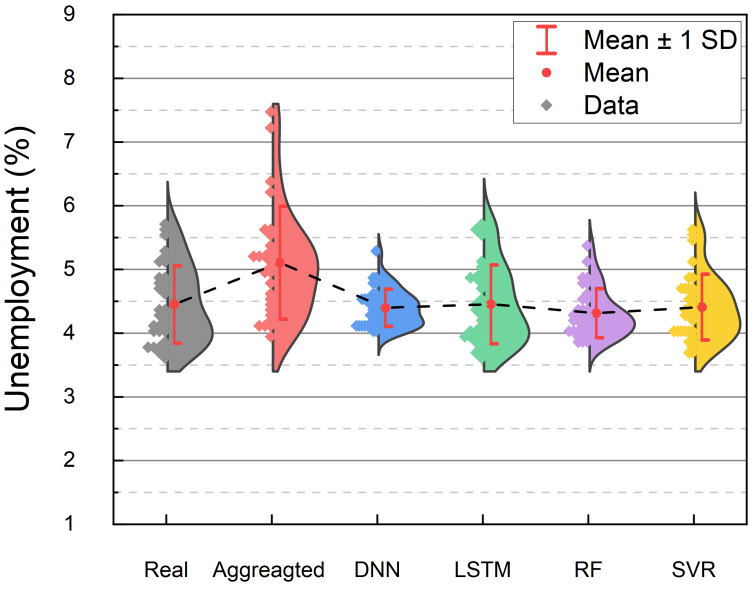

FIGURE 4.17: Comparison of Data Distributions in Unemployment.

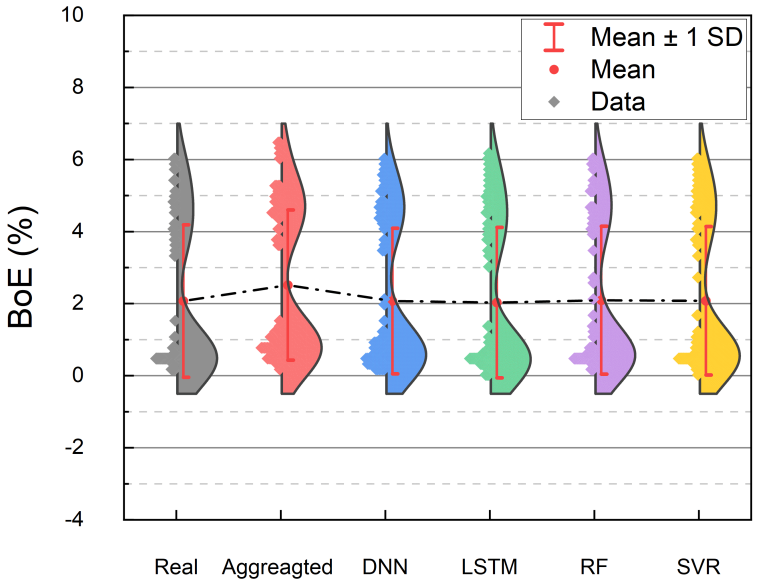

FIGURE 4.18: Comparison of Data Distributions in Bank rate.

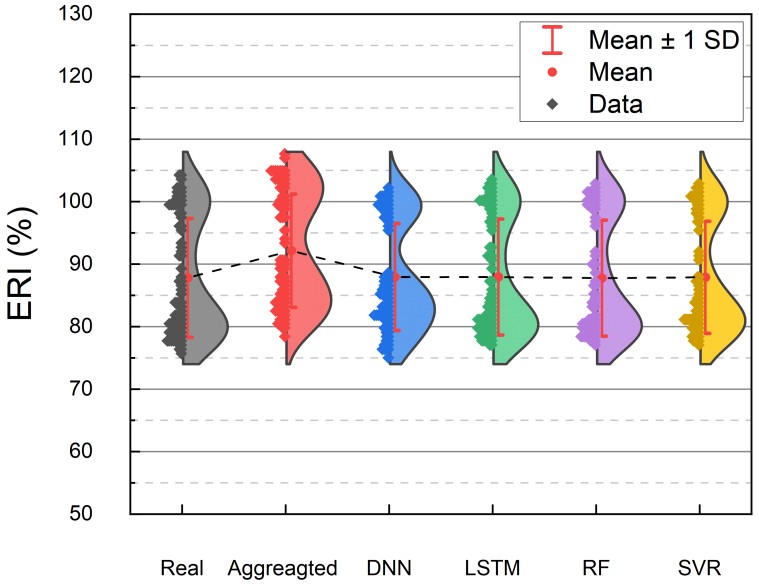

FIGURE 4.19: Comparison of Data Distributions in ERI.

4.6 Conclusion

Agreed with Feuerriegel and Gordon (2019) stated that predicting macroeconomic variables is difficult for many reasons and requires the time-intensive collection of economic data, which, as a result, is often out of date. Professional macroeconomic forecasts, such as those from central banks, usually stem from quantitative predictions and economic experts' judgement (Matsypura et al., 2018). However, numerous studies have analysed professional forecasts for their predictive performance and ability to identify potential biases (Blanc and Setzer, 2015; Mostard et al., 2011). In particular, Jansen et al. (2016) revealed that subjective estimates by experts are less efficient predictors of gross domestic product as compared with statistical models.

Our study introduces a new perspective on strategies for optimising macroeconomic forecasting. We build a comprehensive framework by applying both machine learning and deep learning as a hybrid approach to optimize UK macroeconomic forecasts. The algorithm architectures are constructed by utilising the most well-known deep learning models: deep neural networks (DNN) and Long Short-Term Memory (LSTM), and machine learning models: Support Vector Regression (SVR) and Random Forest (RF). We build the optimal combination of machine learning algorithms for UK professional forecasters in macroeconomic forecasting. We suggest that machine learning exhibits a significant predictive ability, while deep learning can also be effective as an addition to achieving an optimization goal in macroeconomic forecasting. Moreover, our results also highlight in the use of the UK professional forecaster survey data in making such subjective estimation be predictive. Since there is no model or methodology that produces the best result for every type of data set, this study contributes to the literature.

However, as demonstrated in this study and in numerous prior research endeavours, machine learning models consistently exhibit robust predictive capabilities. In conclusion, based on the validated outcomes, this study also advocates and encourages further exploration and utilization of machine learning models for economic variable forecasting and addressing economic inquiries.

In our future work, as Ribeiro et al. (2016) indicated interpretability is a paramount quality that machine learning as it helps to understand why machine learning models behave the way they do and empowers both system designers and end-users in many ways: in model selection, feature engineering. Lipton (2018) defined and formulated interpretability can be divided (but not limited) into two main categories: model transparency and post-hoc interpretability. Turbé et al. (2023) denoted that the machine learning model, e.g., the neural network interpretability for time-series data, was only recently explored. Thus, in our future work, the particular care on model interpretability is also highlighted as an addition to the literature gap.

Chapter 5

Conclusion and Future works

5.1 Conclusion

In chapter 2, we establish a framework for knowledge elicitation to assess expert performance in two critical dimensions: statistical accuracy (calibration score) and the informativeness of their knowledge (information score). To accomplish this, we introduce Cooke's classical model. Within this chapter, we initially observe substantial variations in the predictive abilities of individual experts. Subsequently, we assign new weights to each expert based on their individual performance. With these adjusted weights, we form a new set of expert predictions by combining their original forecasts. Additionally, we validate these new predictions and note a significant overall improvement.

Furthermore, we detect a collective overestimation tendency among the experts, prompting us to delve into whether these experts are following rational strategies, a question we explore in subsequent work. Additionally, we uncover a systematic bias in expert predictions associated with the values of the preceding one-year-ahead releases. This discovery aligns with the findings of (Campbell and Sharpe, 2009), who suggested that such a bias is consistent with the anchoring and adjustment heuristic proposed by (Tversky and Kahneman, 1974).

In Chapter 3, we extend our investigation to gain insights into expert prediction performance from a cognitive perspective. Our goal is set to assess whether professional forecasters make predictions based on rational behaviour. To achieve this, we develop a classification method inspired by the work of (Brito et al., 2008) to categorize expert preferences into different risk attitude groups, such as optimists or pessimists, based on the shape of their cumulative probability. Moreover, we gain inspiration from Engelberg et al. (2009), who investigate the outlook of forecasters, discerning whether they exhibit optimism or pessimism by considering the fraction of a limited subset that falls beyond the central tendency measures. In addition, Huang et al. (2022) expanded

upon this approach by appraising the proportions of forecasters' specific predictions that align with the intervals encompassing the 25th and 75th quartiles of the forecast's distribution.

In Chapter 4, we initially highlight the limitations of traditional macroeconomic prediction models and the suboptimal predictive ability associated with relying solely on a single machine learning model for forecasting tasks. To address the persistent issue in macroeconomic forecasting, we establish a comprehensive machine learning framework by integrating two machine learning algorithms—Support Vector Regression (SVR) and Random Forest (RF) — and two deep learning algorithms—Deep Neural Networks (DNN) and Long Short-Term Memory (LSTM). This approach aims to enhance predictive power and improve the accuracy of macroeconomic predictions. Our results suggest that machine learning exhibits a significant predictive ability, while deep learning can also be effective as an addition to achieve an optimization goal in macroeconomic forecasting. Our contribution lies in finding the solution to selecting optimal hyperparameters, which is a critical problem in machine learning. Our results also provide evidence on the application of machine learning is valid even with a small data sample.

5.2 Research Limitations

While this paper presents insightful results and a unique perspective on improving the use of forecasts by professional forecasters, it also acknowledges some of the following limitations and shortcomings.

The challenge is about the issue of constrained data availability. we notice the potential limitation about data vintage and data revision from the literature. First, we learned the concern of data vintage from Stark and Croushore (2002), they discuss the vintage of the data makes difference for forecast accuracy. The choice of the horizon (long and short term), or the number of forecast observations used to evaluate models are critical. Furthermore, Croushore and Stark (2003) verified some key results in the macroeconomic literature are affected by the choice of vintage. In some cases, the results are significantly upended. Second, Croushore (2006) described reasons why forecasts will be affected by data revisions in three aspects: 1, revisions change the data input into the forecasting model. 2, revisions change the estimated coefficients; and 3, revisions lead to a change in the model itself (such as the number of lags).

5.3 Future Research Directions

Future work direction of Chapter 2:

Learned from Timmermann (2006) the amalgamation of forecasts presents diversification benefits, rendering the consolidation of individual predictions more appealing than relying solely on forecasts from a single model. Therefore, in our upcoming research, we intend to delve into the significance of combining forecasts, especially in scenarios involving asymmetric loss functions. Our investigation will also involve a thorough examination of integrating point, interval, and probability forecasts. Additionally, we aim to explore how a consistent panel of experts behaves when making predictions over extended timeframes, comparing their performance across various periods.

Future work direction of Chapter 3:

Based on insights from Moore and Schatz (2017), we now recognize that overconfidence is not a singular, uniform concept. Instead, it can be categorised into three distinct forms:

1. Overestimation involves overestimating one's own abilities, leading to the belief that one is better than one's actual reality. 2. Overplacement, characterised by an exaggerated belief in one's superiority over others. 3. Overprovision, entailing an excessive faith in one's knowledge and the conviction of possessing absolute truth. These three dimensions of overconfidence manifest in various circumstances, originate from different sources, and result in diverse outcomes. It is crucial not to treat them as identical or assume they share the same psychological foundations. In our upcoming research, we intend to explore expert behaviour by taking into account these distinct dimensions of overconfidence.

Future work direction of Chapter 4

In this chapter, we delve into the application of optimising expert predictions. The future trajectory of our work aligns with ribeiro2016model, emphasising interpretability as a crucial attribute in machine learning. Interpretability aids in understanding the behaviour of machine learning models, empowering system designers and end-users in tasks such as model selection and feature engineering. Lipton (2018) categorised interpretability into two main types: model transparency and post-hoc interpretability. Notably, Turbé et al. (2023) pointed out that the interpretability of machine learning models, particularly neural networks for time-series data, has only recently gained attention. Therefore, in our forthcoming research, we will place particular emphasis on ensuring model interpretability, addressing a gap in the existing literature.

Appendix A

Supplement to Chapter 2

A.1 MATLAB code:

```
1.      25% Quantile


clear all
clc

interval=[];%prediction % range
Real=[];%real data

Predict=[];%prediction data
A=[];%Raw data

% prediction null data filling
for m=1:1:length(Predict)
    if Predict(m,1)==0
        Predict(m,1)=-100;
    else
    end
end

% definition of 25 quartile calculation
x=size(A);
q25=zeros(x(1,1),5);
B=zeros(x(1,1),3);%number of observations;
NULL_p=find(Predict(:)==-100);%null data position;
Predict_p=find(Predict(:)>-100);%valid prediction position;
Predict_No=length(Predict_p);%number of predictions;

% 25 quartile calculation
for n=1:1:x(1,1)
    a=0;
    mode_p=[];
    mode_p=find(A(n,:)==max(A(n,:)));%mode location;
    if mode_p(1,1)==1
        z=1;%first column is mode location;
```

```
    elseif mode_p(1,1)==13
        z=-1;%last column is mode location;
    else
        z=0;
    end
for i=1:1:x(1,2)
    b=A(n,i);
    a=a+b;
 if i==1 & a==25
    q25(n,1)=0;
    q25(n,2)=interval(1,i);
    B(n,1)=1;
    if Predict(n,1)<q25(n,2) & Predict(n,1)>-100
        q25(n,4)=1;
    elseif Predict(n,1)>=q25(n,2) & Predict(n,1)>-100
        q25(n,5)=1;
    else
    end
 elseif i==1 & a>25;
    q25(n,1)=0;
    q25(n,2)=interval(1,i)-(0.5-(a-25)/a);
    B(n,1)=1;
    if Predict(n,1)<q25(n,2) & Predict(n,1)>-100
        q25(n,4)=1;
    elseif Predict(n,1)>=q25(n,2) & Predict(n,1)>-100
        q25(n,5)=1;
    else
    end
 elseif i>=2 & i<=x(1,2) & a>=0 & (a-25)>=0 & (a-b-25)<0;
     if a>=25 & i<=x(1,2)-1;
         q25(n,1)=(25-(a-b))/b-0.5+interval(1,i-1);
         q25(n,2)=(25-(a-b))/b-0.5+interval(1,i);
         B(n,1)=1;
         if Predict(n,1)<q25(n,2) & Predict(n,1)>=q25(n,1)
             q25(n,4)=1;
         elseif Predict(n,1)<q25(n,1) & Predict(n,1)>-100
             q25(n,3)=1;
         elseif Predict(n,1)>=q25(n,2) & Predict(n,1)>-100
             q25(n,5)=1;
         else
         end
     else a>=25 & i==x(1,2);
         q25(n,1)=(25-(a-b))/b-0.5+interval(1,i-1);
         q25(n,2)=0;
         B(n,1)=1;
         if Predict(n,1)<q25(n,1) & Predict(n,1)>-100
             q25(n,3)=1;
         elseif Predict(n,1)>=q25(n,1) & Predict(n,1)>-100
             q25(n,4)=1;
         else
         end
     end
 end
end
end

% valid prediction number
for j=1:1:length(NULL_p)
    B(NULL_p(j,:),2)=-100;
```

```
end

for k=1:1:x(1,1)
    if B(k,1)==1 & B(k,2)>-100
        B(k,3)=1;
    else
    end
end
Sub_Ob_No=sum(B(:,1));
Valid_No=sum(B(:,3));


2.      50% Quantile
clear all
clc

interval=[];%prediction % range
Real=[];%real data

Predict=[];%prediction data
A=[];%Raw data

% prediction null data filling
for m=1:1:length(Predict)
    if Predict(m,1)==0
        Predict(m,1)=-100;
    else
    end
end

%definition of Median, Mean and Mode calculation
x=size(A);
Median=zeros(x(1,1),5);
Mean=zeros(x(1,1),5);
Mode=zeros(x(1,1),5);
B=zeros(x(1,1),3);%number of observations;
NULL_p=find(Predict(:)==-100);%null data position;
Predict_p=find(Predict(:)>-100);%valid prediction position;
Predict_No=length(Predict_p);%number of predictions;

% Median, Mean and Mode calculation
for n=1:1:x(1,1)
    a=0;
    mode_p=[];
    mode_p=find(A(n,:)==max(A(n,:)));%mode location;
    if mode_p(1,1)==1
        z=1;%first column is mode location;
    elseif mode_p(1,1)==13
        z=-1;%last column is mode location;
    else
        z=0;
    end
for i=1:1:x(1,2)
    b=A(n,i);
    a=a+b;
 if i==1 & a==50
    Median(n,1)=0;
    Median(n,2)=interval(1,i);
    Mean(n,1)=0;
```

```
    Mean(n,2)=interval(1,i);
    Mode(n,1)=0;
    Mode(n,2)=interval(1,i);
    B(n,1)=1;
    if Predict(n,1)<interval(1,i) & Predict(n,1)>-100
        Median(n,4)=1;
        Mean(n,4)=1;
        Mode(n,4)=1;
    elseif Predict(n,1)>=interval(1,i) & Predict(n,1)>-100
        Median(n,5)=1;
        Mean(n,5)=1;
        Mode(n,5)=1;
    else
    end
elseif i==1 & a>50;
    Median(n,1)=0;
    Median(n,2)=interval(1,i);
    Mean(n,1)=0;
    Mean(n,2)=interval(1,i)-(0.5-(a-50)/a);
    Mode(n,1)=0;
    Mode(n,2)=interval(1,i);
    B(n,1)=1;
    if Predict(n,1)<Median(n,2) & Predict(n,1)>-100
        Median(n,4)=1;
    elseif Predict(n,1)>=Median(n,2) & Predict(n,1)>-100
        Median(n,5)=1;
    else
    end
    if Predict(n,1)<Mean(n,2) & Predict(n,1)>-100
        Mean(n,4)=1;
    elseif Predict(n,1)>=Mean(n,2) & Predict(n,1)>-100
        Mean(n,5)=1;
    else
    end
    if Predict(n,1)<Mode(n,2) & Predict(n,1)>-100
        Mode(n,4)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100
        Mode(n,5)=1;
    else
    end
elseif i>=2 & i<=x(1,2) & a>=0 & (a-50)>=0 & (a-b-50)<0;
     if a>=50 & i<=x(1,2)-1;
         Median(n,1)=interval(1,i-1);
         Median(n,2)=interval(1,i);
         Mean(n,1)=(50-(a-b))/b-0.5+interval(1,i-1);
         Mean(n,2)=(50-(a-b))/b-0.5+interval(1,i);
         if z==-1;%last column is mode location;
             Mode(n,1)=interval(1,mode_p(1,1)-1);
             Mode(n,2)=0;
         elseif z==1;%first column is mode location;
             Mode(n,1)=0;
             Mode(n,2)=interval(1,mode_p(1,1));
         else z==0;
             Mode(n,1)=interval(1,mode_p(1,1)-1);
             Mode(n,2)=interval(1,mode_p(1,1));
         end
         B(n,1)=1;
         if Predict(n,1)<Median(n,2) & Predict(n,1)>=Median(n,1)
             Median(n,4)=1;
```

```
    elseif Predict(n,1)<Median(n,1) & Predict(n,1)>-100
        Median(n,3)=1;
    elseif Predict(n,1)>=Median(n,2) & Predict(n,1)>-100
        Median(n,5)=1;
    else
    end
    if Predict(n,1)<Mean(n,2) & Predict(n,1)>=Mean(n,1)
        Mean(n,4)=1;
    elseif Predict(n,1)<Mean(n,1) & Predict(n,1)>-100
        Mean(n,3)=1;
    elseif Predict(n,1)>=Mean(n,2) & Predict(n,1)>-100
        Mean(n,5)=1;
    else
    end
    if Predict(n,1)<Mode(n,2) & Predict(n,1)>=Mode(n,1) & z==0
        Mode(n,4)=1;
    elseif Predict(n,1)<Mode(n,1) & Predict(n,1)>-100 & z==0
        Mode(n,3)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100 & z==0
        Mode(n,5)=1;
    elseif Predict(n,1)<Mode(n,2) & Predict(n,1)>-100 & z==1
        Mode(n,4)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100 & z==1
        Mode(n,5)=1;
    elseif Predict(n,1)>=Mode(n,1) & Predict(n,1)>-100 & z==-1
        Mode(n,4)=1;
    elseif Predict(n,1)<Mode(n,1) & Predict(n,1)>-100 & z==-1
        Mode(n,5)=1;
    else
    end
else a>=50 & i==x(1,2);
    Median(n,1)=interval(1,i-1);
    Median(n,2)=0;
    Mean(n,1)=(50-(a-b))/b-0.5+interval(1,i-1);
    Mean(n,2)=0;
    Mode(n,1)=interval(1,mode_p(1,1)-1);
    Mode(n,2)=interval(1,mode_p(1,1));
    B(n,1)=1;
    if Predict(n,1)<Median(n,1) & Predict(n,1)>-100
        Median(n,3)=1;
    elseif Predict(n,1)>=Median(n,1) & Predict(n,1)>-100
        Median(n,4)=1;
    else
    end
    if Predict(n,1)<Mean(n,1) & Predict(n,1)>-100
        Mean(n,3)=1;
    elseif Predict(n,1)>=Mean(n,1) & Predict(n,1)>-100
        Mean(n,4)=1;
    else
    end
    if Predict(n,1)<Mode(n,2) & Predict(n,1)>=Mode(n,1) & z==0
        Mode(n,4)=1;
    elseif Predict(n,1)<Mode(n,1) & Predict(n,1)>-100 & z==0
        Mode(n,3)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100 & z==0
        Mode(n,5)=1;
    elseif Predict(n,1)<Mode(n,2) & Predict(n,1)>-100 & z==1
        Mode(n,4)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100 & z==1
```

```
            Mode(n,5)=1;
        elseif Predict(n,1)>=Mode(n,1) & Predict(n,1)>-100 & z==-1
            Mode(n,4)=1;
        elseif Predict(n,1)<Mode(n,1) & Predict(n,1)>-100 & z==-1
            Mode(n,5)=1;
        else
        end
    end
 end
end
end

% valid prediction number
for j=1:1:length(NULL_p)
    B(NULL_p(j,:),2)=-100;
end

for k=1:1:x(1,1)
    if B(k,1)==1 & B(k,2)>-100
        B(k,3)=1;
    else
    end
end
Sub_Ob_No=sum(B(:,1));
Valid_No=sum(B(:,3));
```

3. 75% quantile

```
clear all
clc

interval=[];%prediction % range
Real=[];%real data

Predict=[];%prediction data
A=[];%Raw data

% prediction null data filling
for m=1:1:length(Predict)
    if Predict(m,1)==0
        Predict(m,1)=-100;
    else
    end
end

% definition of 75 quartile calculation
x=size(A);
q75=zeros(x(1,1),5);
B=zeros(x(1,1),3);%number of observations;
NULL_p=find(Predict(:)==-100);%null data position;
Predict_p=find(Predict(:)>-100);%valid prediction position;
Predict_No=length(Predict_p);%number of predictions;

% 75 quartile calculation
```

```
for n=1:1:x(1,1)
    a=0;
    mode_p=[];
    mode_p=find(A(n,:)==max(A(n,:)));%mode location;
    if mode_p(1,1)==1
        z=1;%first column is mode location;
    elseif mode_p(1,1)==13
        z=-1;%last column is mode location;
    else
        z=0;
    end
for i=1:1:x(1,2)
    b=A(n,i);
    a=a+b;
 if i==1 & a==75
    q75(n,1)=0;
    q75(n,2)=interval(1,i);
    B(n,1)=1;
    if Predict(n,1)<q75(n,2) & Predict(n,1)>-100
        q75(n,4)=1;
    elseif Predict(n,1)>=q75(n,2) & Predict(n,1)>-100
        q75(n,5)=1;
    else
    end
 elseif i==1 & a>75;
    q75(n,1)=0;
    q75(n,2)=interval(1,i)-(0.5-(a-75)/a);
    B(n,1)=1;
    if Predict(n,1)<q75(n,2) & Predict(n,1)>-100
        q75(n,4)=1;
    elseif Predict(n,1)>=q75(n,2) & Predict(n,1)>-100
        q75(n,5)=1;
    else
    end
 elseif i>=2 & i<=x(1,2) & a>=0 & (a-75)>=0 & (a-b-75)<0;
     if a>=75 & i<=x(1,2)-1;
         q75(n,1)=(75-(a-b))/b-0.5+interval(1,i-1);
         q75(n,2)=(75-(a-b))/b-0.5+interval(1,i);
         B(n,1)=1;
         if Predict(n,1)<q75(n,2) & Predict(n,1)>=q75(n,1)
             q75(n,4)=1;
         elseif Predict(n,1)<q75(n,1) & Predict(n,1)>-100
             q75(n,3)=1;
         elseif Predict(n,1)>=q75(n,2) & Predict(n,1)>-100
             q75(n,5)=1;
         else
         end
     else a>=75 & i==x(1,2);
         q75(n,1)=(75-(a-b))/b-0.5+interval(1,i-1);
         q75(n,2)=0;
         B(n,1)=1;
         if Predict(n,1)<q75(n,1) & Predict(n,1)>-100
             q75(n,3)=1;
         elseif Predict(n,1)>=q75(n,1) & Predict(n,1)>-100
             q75(n,4)=1;
         else
         end
     end
 end
```

```
end
end

% valid prediction number
for j=1:1:length(NULL_p)
    B(NULL_p(j,:),2)=-100;
end

for k=1:1:x(1,1)
    if B(k,1)==1 & B(k,2)>-100
        B(k,3)=1;
    else
    end
end
Sub_Ob_No=sum(B(:,1));
Valid_No=sum(B(:,3));
```

A.2 The Procedure of EXCALIBUR in Expert Judgement Elicitation

TABLE A.1: The structured procedure of EXCALIBUR in Expert Judgement Elicitation.

1 To select a group of experts in the economic domain.
2 To ensure that these experts are elicited individually regarding their uncertainty over the results of possible measurements or observations within their domain of expertise.
3 Experts also assess variables within their field, and the true values of these variables are known or known after the fact.
4 Experts are treated as statistical hypotheses and are scored regarding statistical likelihood and informativeness.
5 Scores are combined to form weights. With these weights, statistical accuracy strongly dominates informativeness one cannot compensate for poor statistical performance with very high information.
6 Likelihood and informative scores are used to derive performance-based weighted combinations of the experts' uncertainty distributions.

A.3 Comparison of expert's predication value VS True value.

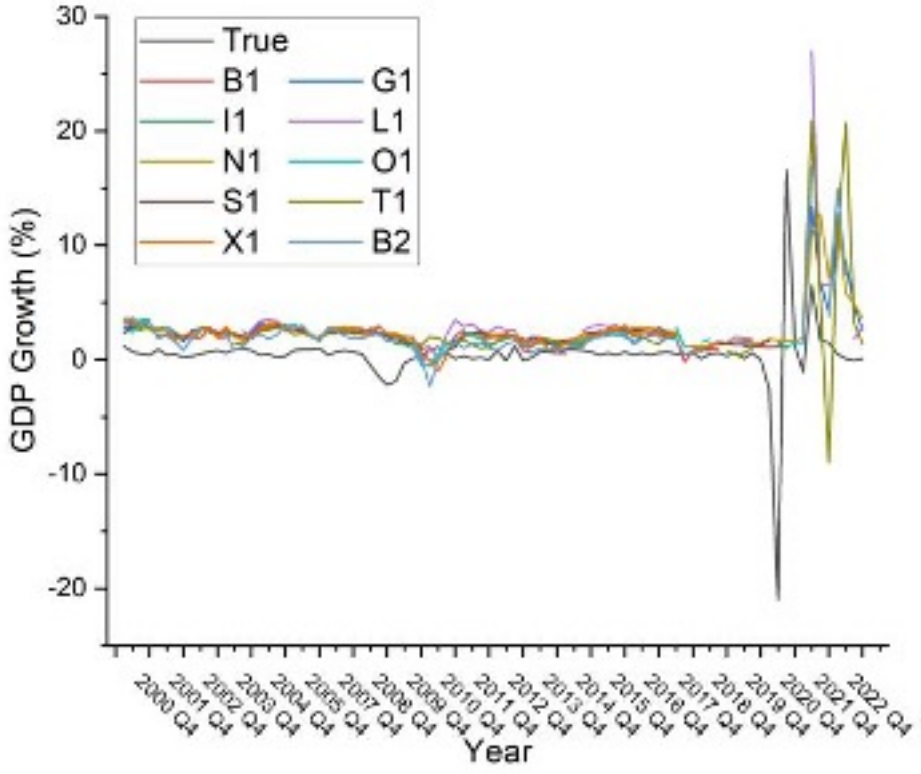

FIGURE A.1: The comparison of ten experts' prediction on the GDP and the real GDP value.

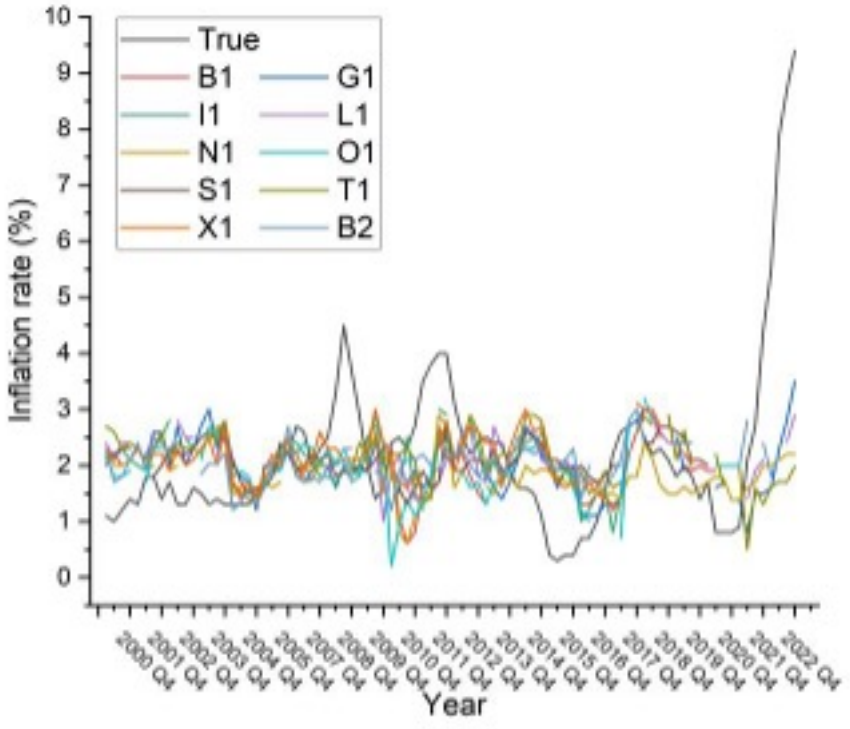

FIGURE A.2: The comparison of ten experts' prediction on the Inflation rate and real Inflation rate.

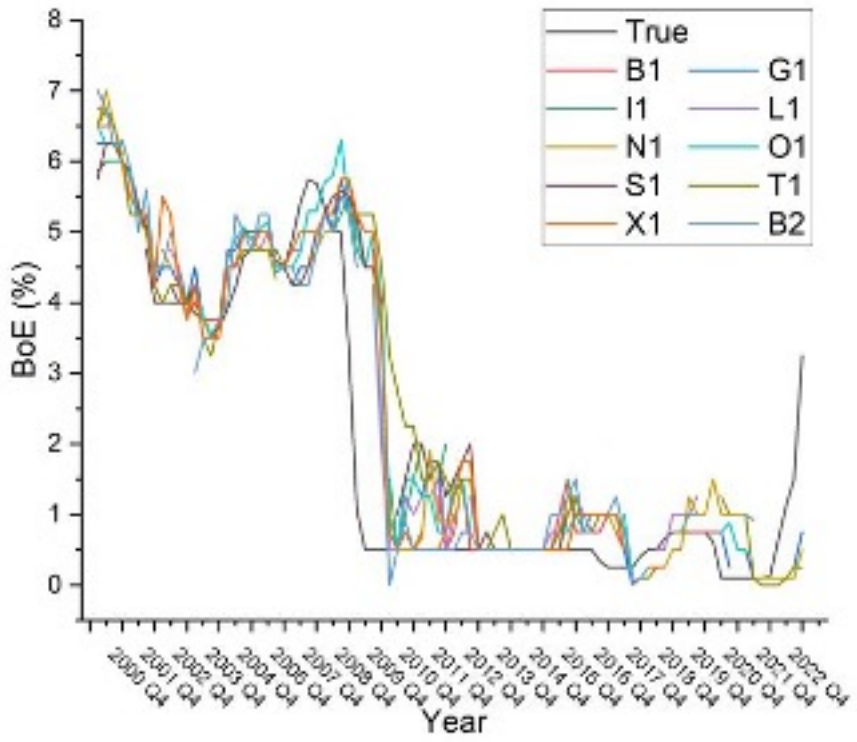

FIGURE A.3: The comparison of ten experts' prediction on the BoE and the real BoE value.

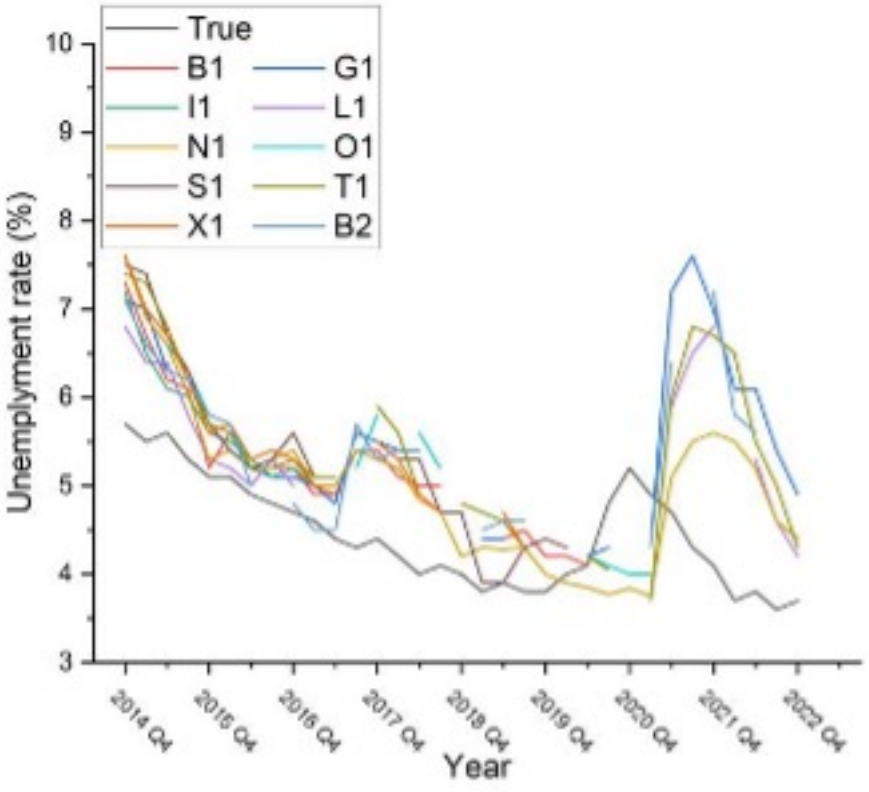

FIGURE A.4: The comparison of ten experts' prediction on the Unemployment rate and the real unemployment rate.

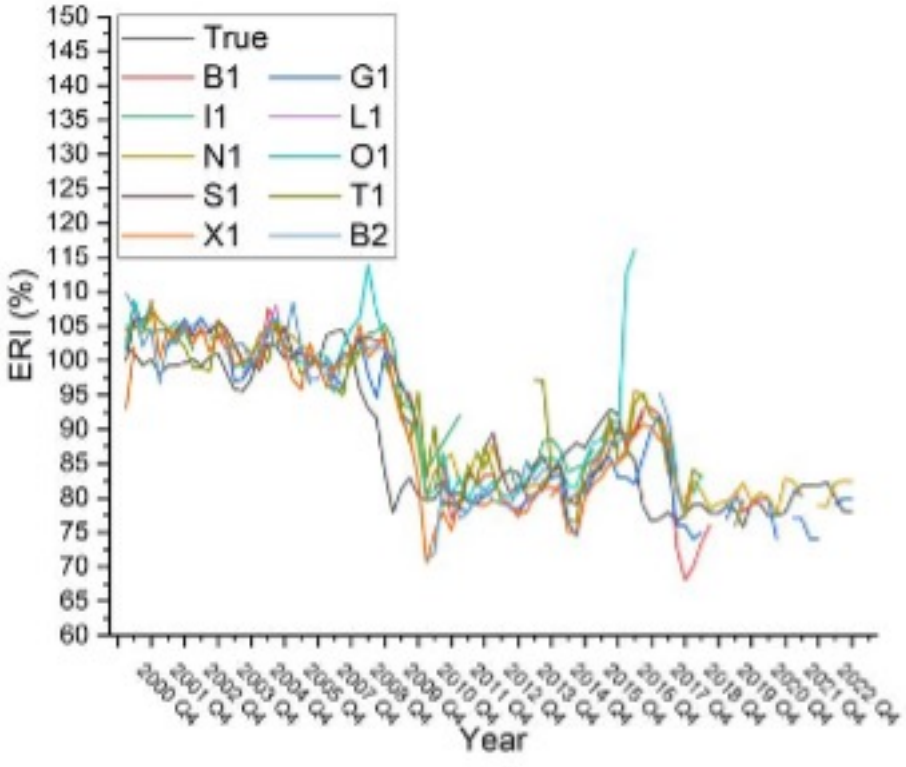

FIGURE A.5: The comparison of ten experts' prediction on the ERI and the real ERI value.

Appendix B

Supplement to Chapter 3

B.1 Matlab code

```
1.      25%Quantile
clear all
clc

interval=[];%prediction % range
Real=[];%real data

Predict=[];%prediction data
A=[];%Raw data

% prediction null data filling
for m=1:1:length(Predict)
    if Predict(m,1)==0
        Predict(m,1)=-100;
    else
    end
end

% definition of 25 quartile calculation
x=size(A);
q25=zeros(x(1,1),5);
B=zeros(x(1,1),3);%number of observations;
NULL_p=find(Predict(:)==-100);%null data position;
Predict_p=find(Predict(:)>-100);%valid prediction position;
Predict_No=length(Predict_p);%number of predictions;

% 25 quartile calculation
for n=1:1:x(1,1)
    a=0;
    mode_p=[];
    mode_p=find(A(n,:)==max(A(n,:)));%mode location;
    if mode_p(1,1)==1
        z=1;%first column is mode location;
    elseif mode_p(1,1)==13
        z=-1;%last column is mode location;
    else
```

```
        z=0;
    end
for i=1:1:x(1,2)
    b=A(n,i);
    a=a+b;
 if i==1 & a==25
    q25(n,1)=0;
    q25(n,2)=interval(1,i);
    B(n,1)=1;
    if Predict(n,1)<q25(n,2) & Predict(n,1)>-100
        q25(n,4)=1;
    elseif Predict(n,1)>=q25(n,2) & Predict(n,1)>-100
        q25(n,5)=1;
    else
    end
 elseif i==1 & a>25;
    q25(n,1)=0;
    q25(n,2)=interval(1,i)-(0.5-(a-25)/a);
    B(n,1)=1;
    if Predict(n,1)<q25(n,2) & Predict(n,1)>-100
        q25(n,4)=1;
    elseif Predict(n,1)>=q25(n,2) & Predict(n,1)>-100
        q25(n,5)=1;
    else
    end
 elseif i>=2 & i<=x(1,2) & a>=0 & (a-25)>=0 & (a-b-25)<0;
     if a>=25 & i<=x(1,2)-1;
         q25(n,1)=(25-(a-b))/b-0.5+interval(1,i-1);
         q25(n,2)=(25-(a-b))/b-0.5+interval(1,i);
         B(n,1)=1;
         if Predict(n,1)<q25(n,2) & Predict(n,1)>=q25(n,1)
             q25(n,4)=1;
         elseif Predict(n,1)<q25(n,1) & Predict(n,1)>-100
             q25(n,3)=1;
         elseif Predict(n,1)>=q25(n,2) & Predict(n,1)>-100
             q25(n,5)=1;
         else
         end
     else a>=25 & i==x(1,2);
         q25(n,1)=(25-(a-b))/b-0.5+interval(1,i-1);
         q25(n,2)=0;
         B(n,1)=1;
         if Predict(n,1)<q25(n,1) & Predict(n,1)>-100
             q25(n,3)=1;
         elseif Predict(n,1)>=q25(n,1) & Predict(n,1)>-100
             q25(n,4)=1;
         else
         end
     end
 end
end
end

% valid prediction number
for j=1:1:length(NULL_p)
    B(NULL_p(j,:),2)=-100;
end

for k=1:1:x(1,1)
```

```
    if B(k,1)==1 & B(k,2)>-100
        B(k,3)=1;
    else
    end
end
Sub_Ob_No=sum(B(:,1));
Valid_No=sum(B(:,3));


2.      50% Quantile
clear all
clc

interval=[];%prediction % range
Real=[];%real data

Predict=[];%prediction data
A=[];%Raw data

% prediction null data filling
for m=1:1:length(Predict)
    if Predict(m,1)==0
        Predict(m,1)=-100;
    else
    end
end

%definition of Median, Mean and Mode calculation
x=size(A);
Median=zeros(x(1,1),5);
Mean=zeros(x(1,1),5);
Mode=zeros(x(1,1),5);
B=zeros(x(1,1),3);%number of observations;
NULL_p=find(Predict(:)==-100);%null data position;
Predict_p=find(Predict(:)>-100);%valid prediction position;
Predict_No=length(Predict_p);%number of predictions;

% Median, Mean and Mode calculation
for n=1:1:x(1,1)
    a=0;
    mode_p=[];
    mode_p=find(A(n,:)==max(A(n,:)));%mode location;
    if mode_p(1,1)==1
        z=1;%first column is mode location;
    elseif mode_p(1,1)==13
        z=-1;%last column is mode location;
    else
        z=0;
    end
for i=1:1:x(1,2)
    b=A(n,i);
    a=a+b;
 if i==1 & a==50
    Median(n,1)=0;
    Median(n,2)=interval(1,i);
    Mean(n,1)=0;
    Mean(n,2)=interval(1,i);
    Mode(n,1)=0;
    Mode(n,2)=interval(1,i);
```

```
    B(n,1)=1;
    if Predict(n,1)<interval(1,i) & Predict(n,1)>-100
        Median(n,4)=1;
        Mean(n,4)=1;
        Mode(n,4)=1;
    elseif Predict(n,1)>=interval(1,i) & Predict(n,1)>-100
        Median(n,5)=1;
        Mean(n,5)=1;
        Mode(n,5)=1;
    else
    end
elseif i==1 & a>50;
    Median(n,1)=0;
    Median(n,2)=interval(1,i);
    Mean(n,1)=0;
    Mean(n,2)=interval(1,i)-(0.5-(a-50)/a);
    Mode(n,1)=0;
    Mode(n,2)=interval(1,i);
    B(n,1)=1;
    if Predict(n,1)<Median(n,2) & Predict(n,1)>-100
        Median(n,4)=1;
    elseif Predict(n,1)>=Median(n,2) & Predict(n,1)>-100
        Median(n,5)=1;
    else
    end
    if Predict(n,1)<Mean(n,2) & Predict(n,1)>-100
        Mean(n,4)=1;
    elseif Predict(n,1)>=Mean(n,2) & Predict(n,1)>-100
        Mean(n,5)=1;
    else
    end
    if Predict(n,1)<Mode(n,2) & Predict(n,1)>-100
        Mode(n,4)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100
        Mode(n,5)=1;
    else
    end
elseif i>=2 & i<=x(1,2) & a>=0 & (a-50)>=0 & (a-b-50)<0;
     if a>=50 & i<=x(1,2)-1;
         Median(n,1)=interval(1,i-1);
         Median(n,2)=interval(1,i);
         Mean(n,1)=(50-(a-b))/b-0.5+interval(1,i-1);
         Mean(n,2)=(50-(a-b))/b-0.5+interval(1,i);
         if z==-1;%last column is mode location;
             Mode(n,1)=interval(1,mode_p(1,1)-1);
             Mode(n,2)=0;
         elseif z==1;%first column is mode location;
             Mode(n,1)=0;
             Mode(n,2)=interval(1,mode_p(1,1));
         else z==0;
             Mode(n,1)=interval(1,mode_p(1,1)-1);
             Mode(n,2)=interval(1,mode_p(1,1));
         end
         B(n,1)=1;
         if Predict(n,1)<Median(n,2) & Predict(n,1)>=Median(n,1)
             Median(n,4)=1;
         elseif Predict(n,1)<Median(n,1) & Predict(n,1)>-100
             Median(n,3)=1;
         elseif Predict(n,1)>=Median(n,2) & Predict(n,1)>-100
```

```
        Median(n,5)=1;
    else
    end
    if Predict(n,1)<Mean(n,2) & Predict(n,1)>=Mean(n,1)
        Mean(n,4)=1;
    elseif Predict(n,1)<Mean(n,1) & Predict(n,1)>-100
        Mean(n,3)=1;
    elseif Predict(n,1)>=Mean(n,2) & Predict(n,1)>-100
        Mean(n,5)=1;
    else
    end
    if Predict(n,1)<Mode(n,2) & Predict(n,1)>=Mode(n,1) & z==0
        Mode(n,4)=1;
    elseif Predict(n,1)<Mode(n,1) & Predict(n,1)>-100 & z==0
        Mode(n,3)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100 & z==0
        Mode(n,5)=1;
    elseif Predict(n,1)<Mode(n,2) & Predict(n,1)>-100 & z==1
        Mode(n,4)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100 & z==1
        Mode(n,5)=1;
    elseif Predict(n,1)>=Mode(n,1) & Predict(n,1)>-100 & z==-1
        Mode(n,4)=1;
    elseif Predict(n,1)<Mode(n,1) & Predict(n,1)>-100 & z==-1
        Mode(n,5)=1;
    else
    end
else a>=50 & i==x(1,2);
    Median(n,1)=interval(1,i-1);
    Median(n,2)=0;
    Mean(n,1)=(50-(a-b))/b-0.5+interval(1,i-1);
    Mean(n,2)=0;
    Mode(n,1)=interval(1,mode_p(1,1)-1);
    Mode(n,2)=interval(1,mode_p(1,1));
    B(n,1)=1;
    if Predict(n,1)<Median(n,1) & Predict(n,1)>-100
        Median(n,3)=1;
    elseif Predict(n,1)>=Median(n,1) & Predict(n,1)>-100
        Median(n,4)=1;
    else
    end
    if Predict(n,1)<Mean(n,1) & Predict(n,1)>-100
        Mean(n,3)=1;
    elseif Predict(n,1)>=Mean(n,1) & Predict(n,1)>-100
        Mean(n,4)=1;
    else
    end
    if Predict(n,1)<Mode(n,2) & Predict(n,1)>=Mode(n,1) & z==0
        Mode(n,4)=1;
    elseif Predict(n,1)<Mode(n,1) & Predict(n,1)>-100 & z==0
        Mode(n,3)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100 & z==0
        Mode(n,5)=1;
    elseif Predict(n,1)<Mode(n,2) & Predict(n,1)>-100 & z==1
        Mode(n,4)=1;
    elseif Predict(n,1)>=Mode(n,2) & Predict(n,1)>-100 & z==1
        Mode(n,5)=1;
    elseif Predict(n,1)>=Mode(n,1) & Predict(n,1)>-100 & z==-1
        Mode(n,4)=1;
```

```
        elseif Predict(n,1)<Mode(n,1) & Predict(n,1)>-100 & z==-1
            Mode(n,5)=1;
        else
        end
    end
 end
end
end

% valid prediction number
for j=1:1:length(NULL_p)
    B(NULL_p(j,:),2)=-100;
end

for k=1:1:x(1,1)
    if B(k,1)==1 & B(k,2)>-100
        B(k,3)=1;
    else
    end
end
Sub_Ob_No=sum(B(:,1));
Valid_No=sum(B(:,3));
```

```
3.      75% quantile

clear all
clc

interval=[];%prediction % range
Real=[];%real data

Predict=[];%prediction data
A=[];%Raw data

% prediction null data filling
for m=1:1:length(Predict)
    if Predict(m,1)==0
        Predict(m,1)=-100;
    else
    end
end

% definition of 75 quartile calculation
x=size(A);
q75=zeros(x(1,1),5);
B=zeros(x(1,1),3);%number of observations;
NULL_p=find(Predict(:)==-100);%null data position;
Predict_p=find(Predict(:)>-100);%valid prediction position;
Predict_No=length(Predict_p);%number of predictions;

% 75 quartile calculation
for n=1:1:x(1,1)
    a=0;
    mode_p=[];
```

```
    mode_p=find(A(n,:)==max(A(n,:)));%mode location;
    if mode_p(1,1)==1
        z=1;%first column is mode location;
    elseif mode_p(1,1)==13
        z=-1;%last column is mode location;
    else
        z=0;
    end
for i=1:1:x(1,2)
    b=A(n,i);
    a=a+b;
 if i==1 & a==75
    q75(n,1)=0;
    q75(n,2)=interval(1,i);
    B(n,1)=1;
    if Predict(n,1)<q75(n,2) & Predict(n,1)>-100
        q75(n,4)=1;
    elseif Predict(n,1)>=q75(n,2) & Predict(n,1)>-100
        q75(n,5)=1;
    else
    end
 elseif i==1 & a>75;
    q75(n,1)=0;
    q75(n,2)=interval(1,i)-(0.5-(a-75)/a);
    B(n,1)=1;
    if Predict(n,1)<q75(n,2) & Predict(n,1)>-100
        q75(n,4)=1;
    elseif Predict(n,1)>=q75(n,2) & Predict(n,1)>-100
        q75(n,5)=1;
    else
    end
 elseif i>=2 & i<=x(1,2) & a>=0 & (a-75)>=0 & (a-b-75)<0;
     if a>=75 & i<=x(1,2)-1;
         q75(n,1)=(75-(a-b))/b-0.5+interval(1,i-1);
         q75(n,2)=(75-(a-b))/b-0.5+interval(1,i);
         B(n,1)=1;
         if Predict(n,1)<q75(n,2) & Predict(n,1)>=q75(n,1)
             q75(n,4)=1;
         elseif Predict(n,1)<q75(n,1) & Predict(n,1)>-100
             q75(n,3)=1;
         elseif Predict(n,1)>=q75(n,2) & Predict(n,1)>-100
             q75(n,5)=1;
         else
         end
     else a>=75 & i==x(1,2);
         q75(n,1)=(75-(a-b))/b-0.5+interval(1,i-1);
         q75(n,2)=0;
         B(n,1)=1;
         if Predict(n,1)<q75(n,1) & Predict(n,1)>-100
             q75(n,3)=1;
         elseif Predict(n,1)>=q75(n,1) & Predict(n,1)>-100
             q75(n,4)=1;
         else
         end
     end
 end
end
end
```

```
% valid prediction number
for j=1:1:length(NULL_p)
    B(NULL_p(j,:),2)=-100;
end

for k=1:1:x(1,1)
    if B(k,1)==1 & B(k,2)>-100
        B(k,3)=1;
    else
    end
end
Sub_Ob_No=sum(B(:,1));
Valid_No=sum(B(:,3));
```

B.2 Table Percent of Experts using N intervals or less.

B.3 Table Evidence of favourable point predictions.

B.4 Table Evidence of favourable point predictions per expert.

TABLE B.1: Experts Intervals in GDP

Experts	1	2	3	4	5	6	7	8	9	10
	$[-5\%, -3\%)$	$[-3\%, -1\%)$	$[-1\%, 0\%)$	$[0\%, 1\%)$	$[1\%, 2\%)$	$[2\%, 3\%)$	$[3\%, 5\%)$	$[5\%, 7\%)$	$[7\%, 9\%)$	$[9\%, \infty)$
B1	0.0%	0.3%	5.0%	22.4%	55.8%	90.2%	99.9%	100.0%	100.0%	100.0%
G1	0.0%	1.0%	5.5%	19.4%	49.5%	82.5%	95.3%	97.1%	98.7%	100.0%
I1	0.0%	3.2%	9.8%	23.5%	54.3%	87.2%	99.0%	100.0%	100.0%	100.0%
L1	0.0%	1.7%	8.5%	21.5%	47.3%	81.9%	99.4%	100.0%	100.0%	100.0%
N1	0.0%	1.2%	5.9%	21.7%	53.7%	85.7%	99.5%	100.0%	100.0%	100.0%
O1	0.0%	0.9%	6.6%	21.6%	51.2%	81.0%	98.1%	99.1%	99.4%	100.0%
S1	0.0%	0.9%	3.3%	15.4%	49.2%	90.7%	99.9%	100.0%	100.0%	100.0%
T1	0.0%	1.8%	9.1%	26.6%	58.7%	89.1%	99.0%	100.0%	100.0%	100.0%
X1	0.0%	2.5%	9.6%	24.9%	53.1%	84.0%	98.9%	100.0%	100.0%	100.0%
B2	0.0%	0.0%	3.1%	13.5%	43.9%	77.9%	93.6%	97.8%	99.0%	100.0%

TABLE B.2: Experts Interval in Inflation

Experts	1	2	3	4	5	6	7	8	9	10
	$(-\infty, 0\%)$	$[0\%, 1\%)$	$[1\%, 1.5\%)$	$[1.5\%, 2\%)$	$[2\%, 2.5\%)$	$[2.5\%, 3\%)$	$[3\%, 3.5\%)$	$[3.5\%, \infty)$	-	
B1	0.0%	1.3%	7.4%	21.8%	53.1%	82.3%	96.2%	100.0%	100.0%	100.0%
G1	0.0%	1.6%	7.3%	22.7%	49.2%	74.9%	92.1%	99.7%	100.0%	100.0%
I1	0.0%	1.8%	9.8%	27.3%	47.3%	70.6%	90.2%	98.6%	100.0%	100.0%
L1	0.0%	2.0%	8.5%	23.3%	47.3%	77.9%	94.7%	99.8%	100.0%	100.0%
N1	0.0%	0.6%	5.5%	23.4%	58.5%	81.2%	94.2%	99.7%	100.0%	100.0%
O1	0.0%	2.6%	13.6%	29.9%	49.4%	69.4%	85.0%	99.0%	100.0%	100.0%
S1	0.0%	0.8%	3.6%	16.3%	43.6%	78.0%	95.3%	100.0%	100.0%	100.0%
T1	0.0%	1.5%	6.2%	18.5%	40.6%	71.3%	91.1%	99.1%	100.0%	100.0%
X1	0.0%	2.8%	9.6%	24.4%	45.4%	71.1%	88.3%	99.1%	100.0%	100.0%
B2	0.0%	1.2%	6.3%	19.5%	45.0%	76.9%	92.9%	99.7%	100.0%	100.0%

TABLE B.3: Experts Interval in Unemployment

Experts	1	2	3	4	5	6	7	8	9	10
	$(-\infty, 4\%)$	$[4\%, 4.5\%)$	$[4.5\%, 5\%)$	$[5\%, 5.5\%)$	$[5.5\%, 6\%)$	$[6\%, 6.5\%)$	$[6.5\%, 7\%)$	$[7\%, 7.5\%)$	$[7.5\%, 8\%)$	$[8\%, \infty)$
B1	6.5%	19.9%	39.5%	63.7%	78.1%	89.3%	95.1%	98.6%	99.9%	100.0%
G1	4.2%	15.6%	29.7%	48.5%	64.0%	76.8%	86.3%	93.8%	98.2%	100.0%
I1	2.2%	9.9%	25.4%	53.3%	73.1%	84.6%	91.8%	96.6%	98.8%	100.0%
L1	0.0%	0.0%	5.0%	15.0%	30.0%	50.0%	75.0%	90.0%	95.0%	100.0%
N1	10.0%	24.1%	42.2%	60.7%	73.5%	84.7%	92.9%	97.1%	99.6%	100.0%
O1	2.4%	4.1%	14.1%	43.4%	72.4%	87.8%	95.4%	98.4%	99.4%	100.0%
S1	10.7%	28.4%	43.0%	57.9%	69.9%	79.9%	87.0%	93.1%	97.0%	100.0%
T1	3.8%	33.1%	44.0%	49.1%	53.1%	61.7%	74.9%	89.2%	97.7%	100.0%
X1	3.1%	13.9%	33.2%	55.0%	70.7%	80.0%	88.0%	94.1%	98.2%	100.0%
B2	2.0%	16.3%	34.3%	51.4%	65.9%	78.6%	87.1%	95.2%	98.8%	100.0%

TABLE B.4: Experts Interval in Base Bank Rate

Experts	1	2	3	4	5	6	7	8	9	10
	$(-\infty, 0\%)$	$[0\%, 0.5\%)$	$[0.5\%, 1\%)$	$[1\%, 1.5\%)$	$[1.5\%, 2\%)$	$[2\%, 2.5\%)$	$[2.5\%, 3\%)$	$[3\%, 3.5\%)$	$[3.5\%, 4\%)$	$[4\%, \infty)$
B1	0.4%	29.7%	77.9%	97.3%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
G1	4.8%	43.0%	71.7%	91.9%	98.2%	99.4%	99.9%	100.0%	100.0%	100.0%
I1	0.1%	19.9%	59.7%	90.4%	97.1%	98.1%	98.9%	99.5%	99.8%	100.0%
L1	-	-	-	-	-	-	-	-	-	-
N1	1.4%	23.3%	64.0%	91.8%	98.4%	100.0%	100.0%	100.0%	100.0%	100.0%
O1	10.0%	35.0%	55.0%	60.0%	70.0%	80.0%	90.0%	95.0%	99.0%	100.0%
S1	-	-	-	-	-	-	-	-	-	-
T1	7.5%	28.0%	51.3%	92.5%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
X1	1.9%	32.7%	68.3%	92.8%	96.6%	98.4%	99.4%	99.7%	99.9%	100.0%
B2	0.9%	21.4%	54.0%	84.8%	97.4%	99.0%	99.4%	99.7%	99.9%	100.0%

TABLE B.5: Evidence of favourable point predictions.

2*Variable	Median		Mean		Mode		25th quartile		75th quartile	
	Below	Above	Below	Above	Below	Above	Below	Above	Below	Above
GDP growth (N)	255	222	233	431	291					
GDP growth (Percent%)	34.12	65.88	39.64	60.36	39.91	60.09	9.05	90.95	84.19	15.81
Inflation (N)	269	266	289	393	405					
Inflation (Percent%)	31.97	68.03	46.62	53.38	32.87	67.13	8.40	91.60	85.19	14.81
Unemployment (N)	75	92	88	98	96					
Unemployment (Percent%)	22.67	77.33	31.52	68.48	18.18	81.82	8.16	91.84	61.46	38.54
BoE (N)	62	73	63	68	70					
BoE (Percent%)	43.55	56.45	58.90	41.10	47.62	52.38	26.47	73.53	17.14	82.86

TABLE B.6: Experts Statistics for GDP Growth

2*Experts	Median			Mean			Mode		
	N	Below	Above	N	Below	Above	N	Below	Above
B1	29	27.59%	72.41%	22	36.36%	63.64%	28	28.57%	71.43%
G1	30	33.33%	66.67%	28	39.29%	60.71%	27	33.33%	66.67%
I1	26	34.62%	65.38%	24	29.17%	70.83%	24	41.67%	58.33%
L1	16	18.75%	81.25%	16	25.00%	75.00%	14	28.57%	71.43%
N1	24	33.33%	66.67%	16	43.75%	56.25%	18	44.44%	55.56%
O1	21	38.10%	61.90%	17	41.18%	58.82%	26	46.15%	53.85%
S1	28	28.57%	71.43%	23	34.78%	65.22%	24	33.33%	66.67%
T1	14	21.43%	78.57%	14	28.57%	71.43%	12	33.33%	66.67%
X1	32	18.75%	81.25%	30	20.00%	80.00%	28	32.14%	67.86%
B2	35	68.57%	31.43%	32	81.25%	18.75%	32	65.63%	34.38%

TABLE B.7: Experts Statistics for Inflation

2*Experts	Below Median			Above Median			Above Mean		
	N	Below	Above	N	Below	Above	N	Below	Above
B1	33	36.36%	63.64%	30	46.67%	53.33%	33	36.36%	63.64%
G1	24	29.17%	70.83%	23	34.78%	65.22%	26	34.62%	65.38%
I1	28	21.43%	78.57%	29	37.93%	62.07%	36	19.44%	80.56%
L1	16	25.00%	75.00%	17	52.94%	47.06%	16	31.25%	68.75%
N1	21	40.00%	60.00%	24	62.50%	37.50%	21	28.57%	71.43%
O1	32	37.50%	62.50%	27	55.56%	44.44%	38	34.21%	65.79%
S1	27	37.04%	62.96%	30	40.00%	60.00%	28	39.29%	60.71%
T1	22	27.27%	72.73%	24	37.50%	62.50%	24	41.67%	58.33%
X1	36	25.00%	75.00%	35	42.86%	57.14%	38	31.58%	68.42%
B2	31	38.71%	61.29%	27	59.26%	40.74%	29	34.48%	65.52%

TABLE B.8: Experts Statistics for Unemployment

2*Experts	Below Median			Above Median			Above Mean		
	N	Below	Above	N	Below	Above	N	Below	Above
B1	9	11.11%	88.89%	12	16.67%	83.33%	11	9.09%	90.91%
G1	16	43.75%	56.25%	20	55.00%	45.00%	17	41.18%	58.82%
I1	7	0.00%	100.00%	9	11.11%	88.89%	9	11.11%	88.89%
L1	-	-	-	-	-	-	-	-	-
N1	77	14.29%	85.71%	13	30.77%	69.23%	14	0.00%	100.00%
O1	6	50.00%	50.00%	5	40.00%	60.00%	6	50.00%	50.00%
S1	10	10.00%	90.00%	12	25.00%	75.00%	10	10.00%	90.00%
T1	1	0.00%	100.00%	1	0.00%	100.00%	1	0.00%	100.00%
X1	7	14.29%	85.71%	9	22.22%	77.78%	7	0.00%	100.00%
B2	12	25.00%	75.00%	11	36.36%	63.64%	13	23.08%	76.92%

TABLE B.9: Experts Statistics for Base Bank Rate

2*Experts	Below Median			Above Median			Above Mean		
	N	Below	Above	N	Below	Above	N	Below	Above
B1	10	40.00%	60.00%	12	58.33%	41.67%	10	40.00%	60.00%
G1	14	42.86%	57.14%	15	53.33%	46.67%	14	42.86%	57.14%
I1	5	40.00%	60.00%	7	42.86%	57.14%	6	50.00%	50.00%
L1	-	-	-	-	-	-	-	-	-
N1	12	58.33%	41.67%	18	61.11%	38.89%	12	58.33%	41.67%
O1	-	-	-	-	-	-	-	-	-
S1	-	-	-	-	-	-	-	-	-
T1	2	50.00%	50.00%	2	50.00%	50.00%	2	50.00%	50.00%
X1	11	27.27%	72.73%	11	63.64%	36.36%	10	40.00%	60.00%
B2	8	50.00%	50.00%	8	75.00%	25.00%	9	55.56%	44.44%

TABLE B.10: Experts Statistics for GDP Growth

2*Experts	Below 25th Quartile			Above 75th Quartile		
	N	Below	Above	N	Below	Above
B1	39	12.82%	87.18%	20	65.00%	35.00%
G1	48	6.25%	93.75%	27	81.48%	18.52%
I1	37	5.41%	94.59%	23	78.26%	21.74%
L1	33	6.06%	93.94%	15	66.67%	33.33%
N1	49	6.12%	93.88%	37	86.49%	13.51%
O1	47	8.51%	91.49%	45	91.11%	8.89%
S1	47	6.38%	93.62%	28	75.00%	25.00%
T1	39	0.00%	100.00%	22	100.00%	0.00%
X1	51	1.96%	98.04%	29	82.76%	17.24%
B2	41	39.02%	60.98%	45	93.33%	6.67%

TABLE B.11: Experts Statistics for inflation

2*Experts	Below 25th Quartile			Above 75th Quartile		
	N	Below	Above	N	Below	Above
B1	38	21.05%	78.95%	30	76.67%	23.33%
G1	29	10.34%	89.66%	34	61.76%	38.24%
I1	38	7.89%	92.11%	36	80.56%	19.44%
L1	28	0.00%	100.00%	29	93.10%	6.90%
N1	39	15.38%	84.62%	53	90.57%	9.43%
O1	50	8.00%	92.00%	48	85.42%	14.58%
S1	41	4.88%	95.12%	45	91.11%	8.89%
T1	35	0.00%	100.00%	32	93.75%	6.25%
X1	59	10.17%	89.83%	52	78.85%	21.15%
B2	36	2.78%	97.22%	46	95.65%	4.35%

TABLE B.12: Experts Statistics for Unemployment

2*Experts	Below 25th Quartile			Above 75th Quartile		
	N	Below	Above	N	Below	Above
B1	13	0.00%	100.00%	11	45.45%	54.55%
G1	16	25.00%	75.00%	21	61.90%	38.10%
I1	9	0.00%	100.00%	7	28.57%	71.43%
L1	-	-	-	-	-	-
N1	19	0.00%	100.00%	19	84.21%	15.79%
O1	4	25.00%	75.00%	7	57.14%	42.86%
S1	13	0.00%	100.00%	9	55.56%	44.44%
T1	1	0.00%	100.00%	1	100.00%	0.00%
X1	11	0.00%	100.00%	11	63.64%	36.36%
B2	12	25.00%	75.00%	10	60.00%	40.00%

TABLE B.13: Experts Statistics for Base bank rate

2*Experts	Below 25th Quartile			Above 75th Quartile		
	N	Below	Above	N	Below	Above
B1	10	10.00%	90.00%	11	81.82%	18.18%
G1	16	31.25%	68.75%	14	78.57%	21.43%
I1	7	28.57%	71.43%	4	75.00%	25.00%
L1	-	-	-	-	-	-
N1	15	33.33%	66.67%	17	88.24%	11.76%
O1	-	-	-	-	-	-
S1	-	-	-	-	-	-
T1	3	33.33%	66.67%	3	66.67%	33.33%
X1	9	11.11%	88.89%	10	80.00%	20.00%
B2	8	37.50%	62.50%	11	90.91%	9.09%

Appendix C

Supplement to Chapter 4

C.1 Python Implementation Code

C.1.1 DNN model

```
import numpy as np
import pandas as pd
from tensorflow import keras
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import Dense
from tensorflow.keras import regularizers
from tensorflow.keras.losses import MeanSquaredError, MeanAbsoluteError, Huber, LogCosh
from tensorflow.keras.callbacks import EarlyStopping, LearningRateScheduler
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, mean_absolute_error
import matplotlib.pyplot as plt
from sklearn.model_selection import TimeSeriesSplit, GridSearchCV
from keras.wrappers.scikit_learn import KerasRegressor
import time
from keras.regularizers import l2 as l2_regularizer
from sklearn.metrics import make_scorer
from sklearn.preprocessing import MinMaxScaler, StandardScaler
import os

import_path = r"C:\Users\Yujia\Desktop\ML v3\fulfilled data\\"
output_path = r"C:\Users\Yujia\Desktop\ML v3\DNN huber results\\"

def huber_loss_scorer(y_true, y_pred):
    error = y_true - y_pred
    threshold = 1  # Huber
    is_small_error = abs(error) <= threshold
    squared_loss = 0.5 * error ** 2
    linear_loss = threshold * (abs(error) - 0.5 * threshold)
    return np.where(is_small_error, squared_loss, linear_loss).mean()

neg_huber_loss_scorer = make_scorer(huber_loss_scorer, greater_is_better=False)

performance = pd.DataFrame()
```

```
# Define the Huber loss
huber_loss = Huber(delta=1.0)
def import_data(dataset_filename): # import data from dataset to DataFrame format
    try:
        # Construct the full file path
        file_path = r"C:\Users\Yujia\Desktop\ML v2\fulfilled data\\" + dataset_filename
        # Read the CSV file into a pandas DataFrame
        raw_data = pd.read_csv(file_path)
        print("File '{}' is successfully imported from file.".format(dataset_filename))
        return raw_data  #DataFrame format
    except FileNotFoundError as e:
        print("File '{}' not found.".format(dataset_filename))
    except Exception as e:
        print("An error occurred: {}".format(e))
def dataframe_to_numpy(dataset_filename):
     raw_data = import_data(dataset_filename)
     print(" '{}' is successfully imported from dataframe.".format(filename))
     start_time = time.time()
     # process the data to matrix
     list_dataTrue = [y for y in raw_data['TRUE']]
     y = np.array(list_dataTrue).ravel()
     columns_to_delete = ['TRUE']
     X = raw_data.drop(columns=columns_to_delete).values
     print(" '{}' is successfully change to numpy.".format(filename))
     return X,y
def MAPE(y_true, y_pred): # mean_absolute_percentage_error
    epsilon = 1e-7
    y_true = np.array(y_true)
    y_pred = np.array(y_pred)
    mape = np.mean(np.abs((y_true - y_pred) / (y_true + epsilon))) * 100
    return mape
def MAE(y_true, y_pred): # mean_absolute_error_loss
    return mean_absolute_error(y_true, y_pred)
def train_DNN(X, y, test_size, random_state, lr, dr, dropout_rate, l2, num_features, hidden_units, acti
              epochs, batch_size, filename, filename1="loss.csv"):

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size, random_state=random_

    model = keras.Sequential()

    for units in hidden_units:
        model.add(Dense(units, activation=activation, kernel_regularizer=regularizers.l2(l2),
                        input_shape=(num_features,)))
        model.add(Dropout(dropout_rate))

    model.add(Dense(units=1, kernel_regularizer=regularizers.l2(l2)))

    opt = keras.optimizers.get(optimizer)
    opt.learning_rate = lr
    model.compile(opt, loss=huber_loss, metrics=['mean_absolute_error'])

    def lr_schedule(epoch, lr):
        if epoch < 50:
            return lr  # Keep initial learning rate for the first 50 epochs
        else:
            return lr * dr  # Reduce learning rate by 5% after 50 epochs

    lr_scheduler = LearningRateScheduler(lr_schedule)
    early_stopping = EarlyStopping(monitor='val_loss', patience=40, restore_best_weights=True)
```

```
    history = model.fit(X_train, y_train, epochs=epochs, batch_size=batch_size, validation_data=
                        callbacks=[lr_scheduler, early_stopping], verbose=3)

    plt.plot(history.history['loss'], label='Train Loss')
    plt.plot(history.history['val_loss'], label='Test Loss')
    plt.xlabel('Epoch')
    plt.ylabel('Loss')
    plt.legend()
    plt.title("Training and Test Loss Over Iterations '{}'".format(filename1))
    file_path_plot1 = output_path + filename + "_plot1.png"
    plt.savefig(file_path_plot1, dpi=300)
    # plt.show()

    loss_history_df = pd.DataFrame({'Train Loss': history.history['loss'], 'Test Loss': history.
    loss_file_path = output_path + "loss" + filename
    loss_history_df.to_csv(loss_file_path, index=False)

    loss = model.evaluate(X_test, y_test)
    print("                  '{}'".format(loss))

    y_pred = model.predict(X_test)
    y_pred = y_pred.reshape(-1)
    mse = mean_squared_error(y_test, y_pred)
    mape = MAPE(y_test, y_pred)
    mae = MAE(y_test, y_pred)

    return model, history.history['loss'], history.history['val_loss'], mse, mape, mae
def graph_with_whole_sample(trained_model, new_data):
    predictions = trained_model.predict(new_data)
    return predictions
def find_hyperparameter(hidden_units=(64, 32), activation='elu',
optimizer='Nadam', kernel_reg=0.0001, dropout_rate=0.5,
                        lr=0.001, dr=0.95):
    num_features = X.shape[1]
    model = keras.Sequential()
    for units in hidden_units:
        model.add(Dense(units, activation=activation,
kernel_regularizer=l2_regularizer(kernel_reg),
                        input_shape=(num_features,)))
        model.add(Dropout(dropout_rate))
    model.add(Dense(1, kernel_regularizer=l2_regularizer(kernel_reg)))

    opt = keras.optimizers.Nadam(learning_rate=lr)
    model.compile(opt, loss='huber_loss', metrics=['mean_absolute_error'])
    return model
class CustomKerasRegressor(KerasRegressor):
    def fit(self, x, y, **kwargs):
        early_stopping = EarlyStopping(monitor='val_loss', patience=20, restore_best_weights=Tru
        lr_scheduler = LearningRateScheduler(create_lr_schedule(kwargs.pop('dr', 0.95)))

        callbacks = kwargs.pop('callbacks', [])
        callbacks.append(early_stopping)
        callbacks.append(lr_scheduler)

        super(CustomKerasRegressor, self).fit(x, y, validation_split=0.2, callbacks=callbacks, *
def create_lr_schedule(dr_value):
    def lr_schedule(epoch, lr):
        if epoch < 50:
```

```
            return lr
        else:
            return lr * dr_value
    return lr_schedule
def best_parameter_MSE(X, y):
    n_splits = 10
    tscv = TimeSeriesSplit(n_splits=n_splits)

    regressor = CustomKerasRegressor(build_fn=find_hyperparameter, verbose=3, epochs=5000, batch_size=3
    param_grid = {
        'lr': [0.00001, 0.0001, 0.001],
        'dr': [0.93, 0.94, 0.95, 0.96],
        'dropout_rate': [0.2, 0.3, 0.4, 0.5],
        'kernel_reg': [0.0000001, 0.000001, 0.00001, 0.0001]  #
    }

    grid = GridSearchCV(estimator=regressor, param_grid=param_grid, cv = tscv, verbose=3,
                        n_jobs=-1,scoring =neg_huber_loss_scorer)


    grid_search = grid.fit(X, y)  # Removed the callbacks here; will be added during model training

    # Now use the best 'dr' for the LearningRateScheduler
    best_dr = grid_search.best_params_['dr']
    lr_scheduler = LearningRateScheduler(create_lr_schedule(best_dr))
    early_stopping = EarlyStopping(monitor='val_loss', patience=20, restore_best_weights=True)
    grid_search.best_estimator_.model.fit(X, y, validation_split=0.2, epochs=5000,
    batch_size=32, callbacks=[lr_scheduler, early_stopping], verbose=3)

    print("Best parameters found: ", grid_search.best_params_)
    print("Best cross-validation score: {:.2f}".format(np.sqrt(-grid_search.best_score_)))
    lr = grid_search.best_params_['lr']
    scoring_method = 'Mean Squared Error'
    score = np.sqrt(-grid_search.best_score_)

    dropout = grid_search.best_params_['dropout_rate']
    kernel_reg = grid_search.best_params_['kernel_reg']
    return lr, best_dr, dropout, kernel_reg, scoring_method, score


if __name__ == "__main__":

    dataset_filename = ["ERI_raw.csv", "GDP_raw.csv", "GDP_without_crisis_raw.csv", "UR_raw.csv"]
    for filename in dataset_filename:
        start_time = time.time()

        X, y = dataframe_to_numpy(filename)

        scaler  = StandardScaler()
        X = scaler.fit_transform(X)
        y = scaler.fit_transform(y.reshape(-1, 1))
        y = y.ravel()

        test_size = 0.2
        random_state = 66  # help you repeat our estimates and receive the same results
        num_features = X.shape[1]
        hidden_units = (64, 32)
        epochs = 5000
        activation = 'elu'
```

```
optimizer = 'Nadam'
batch_size = 64
# lr, dr, dropout_rate, l2, scoring_method, score = best_parameter_MSE(X, y)
lr = 0.001
dr = 0.95
dropout_rate = 0.5
l2 = 0.0001


"""
we can change the scoring methods above by using different functions.
"""
# let's train it
model, train_loss_history, test_loss_history, mse, mape, mae = train_DNN(
    X = X, y = y, test_size = test_size,random_state = random_state,
    lr=lr,dr=dr, dropout_rate = dropout_rate, l2=l2, num_features=num_features,hidden_un
    activation = activation,optimizer = optimizer,epochs = epochs,
    batch_size = batch_size, filename = filename)

rmse = np.sqrt(mse)
end_time = time.time()
elapsed_time = end_time - start_time

print("mse (mean squre error) {:.2f}:".format(rmse))
print("rmse (root of mse): {:.2f}".format(rmse))
print("running  t i m e {:.2f}s".format(elapsed_time))

new_data = X
# receiving predictions by using whole sample period's data
new_predictions = graph_with_whole_sample(model, new_data)
new_predictions = new_predictions.reshape(-1)
print("This is data predicted by our model:", new_predictions)

file_path = output_path + filename
# Save the np to the CSV file
np.savetxt(file_path, new_predictions, delimiter=',', fmt='%.20f')
print("Data saved to '{}'".format(file_path))
# record performance
cleaned_output_path = output_path.rstrip('\\')
file_path_performance = os.path.join(cleaned_output_path, "performance.csv")

# new_row = {"MSE": mse, "RMSE": rmse, "MAPE": mape, "MAE": mae, "Filename": filename,
#            "running time": elapsed_time, "lr": lr,
#            "dr": dr, "dropout_rate": dropout_rate, "l2": l2,
#            "scoring_method": scoring_method, "score": score}
#
# performance = performance.append(new_row, ignore_index=True)
# performance.to_csv(file_path_performance, index=False)
# print(performance)

plt.figure(figsize=(8, 4))
plt.plot(y, label='True value', marker='o')
plt.plot(new_predictions, label='Predicted value', linestyle='--')
plt.xlabel('Time')
plt.ylabel('Y')
plt.legend()
file_path_plot2 = output_path + filename +"_plot2.png"
plt.savefig(file_path_plot2, dpi=300)
```

```
        # plt.show()
```

C.1.2 LSTM model

```
import numpy as np
import pandas as pd
from tensorflow import keras
from tensorflow.keras.layers import Dropout, LSTM, Dense
from tensorflow.keras import regularizers
from tensorflow.keras.losses import MeanSquaredError, MeanAbsoluteError, Huber, LogCosh
from tensorflow.keras.callbacks import EarlyStopping, LearningRateScheduler
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, mean_absolute_error
import matplotlib.pyplot as plt
from sklearn.model_selection import TimeSeriesSplit, GridSearchCV
from tensorflow.keras.wrappers.scikit_learn import KerasRegressor
import time
from tensorflow.keras.regularizers import l2 as l2_regularizer
from sklearn.preprocessing import MinMaxScaler, StandardScaler
import os

performance = pd.DataFrame()
# Define the Huber loss
huber_loss = Huber(delta=1.0)

#enter data path and output path here

import_path = r"C:\Users\Yujia\Desktop\ML v3\fulfilled data\\"
output_path = r"C:\Users\Yujia\Desktop\ML v3\LSTM results\\"

def import_data(dataset_filename): # import data from dataset to DataFrame format
    try:
        # Construct the full file path
        file_path = import_path + dataset_filename
        # Read the CSV file into a pandas DataFrame
        raw_data = pd.read_csv(file_path)
        print("File '{}' is successfully imported from file.".format(dataset_filename))
        return raw_data  #DataFrame format
    except FileNotFoundError as e:
        print("File '{}' not found.".format(dataset_filename))
    except Exception as e:
        print("An error occurred: {}".format(e))
def dataframe_to_numpy(dataset_filename):
     raw_data = import_data(filename)
     print(" '{}' is successfully imported from dataframe.".format(filename))
     start_time = time.time()
     # process the data to matrix
     list_dataTrue = [y for y in raw_data['TRUE']]
     y = np.array(list_dataTrue).ravel()
     columns_to_delete = ['TRUE']
     X = raw_data.drop(columns=columns_to_delete).values
     print(" '{}' is successfully change to numpy.".format(filename))
     return X,y
def MAPE(y_true, y_pred): # mean_absolute_percentage_error
    epsilon = 1e-7
```

```
    y_true = np.array(y_true)
    y_pred = np.array(y_pred)
    mape = np.mean(np.abs((y_true - y_pred) / (y_true + epsilon))) * 100
    return mape
def MAE(y_true, y_pred): # mean_absolute_error_loss
    return mean_absolute_error(y_true, y_pred)
def train_LSTM(X, y, test_size, random_state, lr, dr, dropout_rate, l2, num_features, hidden_uni
               epochs, batch_size, filename, filename1="loss.csv"):

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size, random_state=

    # Reshape the data to be suitable for LSTM
    X_train = X_train.reshape(X_train.shape[0], -1, num_features)
    X_test = X_test.reshape(X_test.shape[0], -1, num_features)

    model = keras.Sequential()

    for units in hidden_units:
        model.add(LSTM(units, activation=activation, kernel_regularizer=regularizers.l2(l2), ret
        model.add(Dropout(dropout_rate))

    model.add(LSTM(units=hidden_units[-1], kernel_regularizer=regularizers.l2(l2)))  # The last
    model.add(Dense(units=1, kernel_regularizer=regularizers.l2(l2)))

    opt = keras.optimizers.get(optimizer)
    opt.learning_rate = lr
    model.compile(opt, loss='mean_squared_error', metrics=['mean_absolute_error'])

    def lr_schedule(epoch, lr):
        if epoch < 50:
            return lr  # Keep initial learning rate for the first 50 epochs
        else:
            return lr * dr  # Reduce learning rate by 5% after 50 epochs

    lr_scheduler = LearningRateScheduler(lr_schedule)
    early_stopping = EarlyStopping(monitor='val_loss', patience=5000,restore_best_weights=True)

    history = model.fit(X_train, y_train, epochs=epochs, batch_size=batch_size, validation_data=
    callbacks=[lr_scheduler, early_stopping], verbose=3)

    plt.plot(history.history['loss'], label='Train Loss')
    plt.plot(history.history['val_loss'], label='Test Loss')
    plt.xlabel('Epoch')
    plt.ylabel('Loss')
    plt.legend()
    plt.title("Training and Test Loss Over Iterations '{}'".format(filename1))
    file_path_plot1 = output_path + filename + "_plot1.png"
    plt.savefig(file_path_plot1, dpi=300)

    loss_history_df = pd.DataFrame({'Train Loss': history.history['loss'], 'Test Loss': history.
    loss_file_path = output_path + filename
    # loss_history_df.to_csv(loss_file_path, index=False)

    loss = model.evaluate(X_test, y_test)
    print("loss in  t e s t  '{}'".format(loss))

    y_pred = model.predict(X_test)
    y_pred = y_pred.reshape(-1)
    mse = mean_squared_error(y_test, y_pred)
```

```
    mape = MAPE(y_test, y_pred)
    mae = MAE(y_test, y_pred)

    return model, history.history['loss'], history.history['val_loss'], mse, mape, mae
def graph_with_whole_sample(trained_model, new_data, num_features):
    new_data = new_data.reshape(new_data.shape[0], -1, num_features)  #
    predictions = trained_model.predict(new_data)
    return predictions
def find_hyperparameter(hidden_units=[64], activation='elu',
optimizer='Nadam', kernel_reg=0.0001, dropout_rate=0.5,
                        lr=0.001, dr=0.95, batch_size = 16):
    num_features = X.shape[1]
    model = keras.Sequential()
    for units in hidden_units:
        model.add(LSTM(units, batch_size = batch_size, activation=activation,
                       kernel_regularizer=l2_regularizer(kernel_reg),
                       return_sequences=True))
        model.add(Dropout(dropout_rate))
    model.add(LSTM(units=hidden_units[-1], kernel_regularizer=l2_regularizer(kernel_reg)))  # The last
    model.add(Dense(1, kernel_regularizer=l2_regularizer(kernel_reg)))

    opt = keras.optimizers.Nadam(learning_rate=lr)
    model.compile(opt, loss='mean_squared_error', metrics=['mean_absolute_error'])
    return model
class CustomKerasRegressor(KerasRegressor):
    def fit(self, x, y, **kwargs):
        early_stopping = EarlyStopping(monitor='val_loss', patience=200, restore_best_weights=True)
        lr_scheduler = LearningRateScheduler(create_lr_schedule(kwargs.pop('dr', 0.95)))

        # callbacks
        callbacks = kwargs.pop('callbacks', [])
        callbacks.append(early_stopping)
        callbacks.append(lr_scheduler)

        super(CustomKerasRegressor, self).fit(x, y, validation_split=0.2, callbacks=callbacks, **kwargs
def create_lr_schedule(dr_value):
    def lr_schedule(epoch, lr):
        if epoch < 50:
            return lr
        else:
            return lr * dr_value
    return lr_schedule
def best_parameter_MSE(X, y):
    n_splits = 10
    tscv = TimeSeriesSplit(n_splits=n_splits)

    regressor = CustomKerasRegressor(build_fn=find_hyperparameter, verbose=3, epochs=5000, batch_size=3
    param_grid = {
        'batch_size':[16, 32, 64],
        'lr': [ 0.1, 0.01, 0.001, 0.0001],
        'dr': [ 1 ],
        'dropout_rate': [0],
        'kernel_reg': [0.001]  # L2
    }

    grid = GridSearchCV(estimator=regressor, param_grid=param_grid, cv=tscv, verbose=3,
                        n_jobs=-1, scoring="neg_mean_squared_error")
    X = X.reshape((X.shape[0], 1, X.shape[1]))  # Reshape the data
    grid_search = grid.fit(X, y)  # Removed the callbacks here; will be added during model training
```

```
    # Now use the best 'dr' for the LearningRateScheduler
    best_dr = grid_search.best_params_['dr']
    lr_scheduler = LearningRateScheduler(create_lr_schedule(best_dr))
    early_stopping = EarlyStopping(monitor='val_loss', patience=20, restore_best_weights=True)
    grid_search.best_estimator_.model.fit(X, y, validation_split=0.2, epochs=5000,
  batch_size=32, callbacks=[lr_scheduler, early_stopping], verbose=3)

    print("Best parameters found: ", grid_search.best_params_)
    print("Best cross-validation score: {:.2f}".format(np.sqrt(-grid_search.best_score_)))
    lr = grid_search.best_params_['lr']
    scoring_method = 'Mean Squared Error'
    score = np.sqrt(-grid_search.best_score_)
    batch_size = grid_search.best_params_['batch_size']
    dropout = grid_search.best_params_['dropout_rate']
    kernel_reg = grid_search.best_params_['kernel_reg']
    return batch_size, lr, best_dr, dropout, kernel_reg, scoring_method, score
if __name__ == "__main__":
    # input data
    dataset_filename = [ "GDP_raw.csv", "GDP_without_crisis_raw.csv", "Inflation_raw.csv"]
    for filename in dataset_filename:
        start_time = time.time()
        X, y = dataframe_to_numpy(filename)

        scaler  = StandardScaler()
        X = scaler.fit_transform(X)
        y = scaler.fit_transform(y.reshape(-1, 1))
        y = y.ravel()

        test_size = 0.2
        random_state = 66  # help you repeat our estimates and receive the same results
        num_features = X.shape[1]
        hidden_units = [64]
        epochs = 5000
        activation = 'relu'
        optimizer = 'Nadam'
        # batch_size = 16
        batch_size, lr, dr, dropout_rate, l2, scoring_method, score = best_parameter_MSE(X, y)


        """
        we can change the scoring methods above by using different functions
        """
        # let's train it
        model, train_loss_history, test_loss_history, mse, mape, mae = train_LSTM(
            X=X, y=y, test_size=test_size, random_state=random_state,
            lr=lr, dr=dr, dropout_rate=dropout_rate, l2=l2, num_features=num_features, hidden_un
            activation=activation, optimizer=optimizer, epochs=epochs,
            batch_size=batch_size, filename=filename)

        rmse = np.sqrt(mse)
        end_time = time.time()
        elapsed_time = end_time - start_time

        print("mse (mean squre error) {:.2f}:".format(rmse))
        print("rmse (root of mse): {:.2f}".format(rmse))
        print("running  t i m e {:.2f}s".format(elapsed_time))

        new_data = X
```

```
        # receiving predictions by using whole sample period's data
        new_predictions = graph_with_whole_sample(model, new_data, num_features)
        new_predictions = new_predictions.reshape(-1)
        print("This is data predicted by our model:", new_predictions)

        file_path = output_path + filename
        # Save the np to the CSV file
        np.savetxt(file_path, new_predictions, delimiter=',', fmt='%.20f')
        print("Data saved to '{}'".format(file_path))
        # record performance

        cleaned_output_path = output_path.rstrip('\\')
        file_path_performance = os.path.join(cleaned_output_path, "performance.csv")

        new_row = {"MSE": mse, "RMSE": rmse, "MAPE": mape, "MAE": mae, "Filename": filename,
                   "running time": elapsed_time, "lr": lr,
                   "dr": dr, "dropout_rate": dropout_rate, "l2": l2,
                   "scoring_method": scoring_method, "score": score}

        performance = performance.append(new_row, ignore_index=True)
        performance.to_csv(file_path_performance, index=False)
        print(performance)

        plt.figure(figsize=(8, 4))
        plt.plot(y, label='True value', marker='o')
        plt.plot(new_predictions, label='Predicted value', linestyle='--')
        plt.xlabel('Time')
        plt.ylabel('Y')
        plt.legend()
        plt.title("True vs. Predicted:'{}'".format(filename))
        file_path_plot2 = output_path + filename +"_plot2.png"
        plt.savefig(file_path_plot2, dpi=300)

        # plt.show()
```

C.1.3 Support Vector Regression (SVR) model

```
import numpy as np
from sklearn.svm import SVR
from sklearn.metrics import mean_squared_error, r2_score,mean_absolute_error
from sklearn.model_selection import train_test_split
import pandas as pd
import time
import matplotlib.pyplot as plt
from sklearn.model_selection import TimeSeriesSplit, GridSearchCV
from sklearn.preprocessing import MinMaxScaler, StandardScaler
import os

import_path = r"C:\Users\Yujia\Desktop\ML v3\fulfilled data\\"
output_path = r"C:\Users\Yujia\Desktop\ML v3\predicted results SVR\\"

def import_data(dataset_filename): # import data from dataset to DataFrame format
    try:
        # Construct the full file path
```

```
        file_path = import_path + dataset_filename
        # Read the CSV file into a pandas DataFrame
        raw_data = pd.read_csv(file_path)
        print("File '{}' is successfully imported from file.".format(dataset_filename))
        return raw_data  #DataFrame format
    except FileNotFoundError as e:
        print("File '{}' not found.".format(dataset_filename))
    except Exception as e:
        print("An error occurred: {}".format(e))
def dataframe_to_numpy(dataset_filename):
     raw_data = import_data(dataset_filename)
     print(" '{}' is successfully imported from dataframe.".format(dataset_filename))
     start_time = time.time()
     # process the data to matrix
     list_dataTrue = [y for y in raw_data['TRUE']]
     y = np.array(list_dataTrue).ravel()
     columns_to_delete = ['TRUE']
     X = raw_data.drop(columns=columns_to_delete).values
     print(" '{}' is successfully change to numpy.".format(dataset_filename))
     return X,y
def MAPE(y_true, y_pred): # mean_absolute_percentage_error
    epsilon = 1e-7
    y_true = np.array(y_true)
    y_pred = np.array(y_pred)
    mape = np.mean(np.abs((y_true - y_pred) / (y_true + epsilon))) * 100
    return mape
def MAE(y_true, y_pred): # mean_absolute_error_loss
    return mean_absolute_error(y_true, y_pred)
"""
train SVR model

parameter
X_train (numpy.ndarray): Training set feature data
y_train (numpy.ndarray): Training set label
kernel (str): SVR kernel function, Default by 'rbf'
C (float): penalty parameter, Default by 1.0
epsilon (float): epsilon-tube parameter, Default by 0.1

back
svr (SVR): trained SVR model
"""
"""
Evaluate SVR model performance.

parameter
svr (SVR): trained SVR model
X_test (numpy.ndarray): Training set feature data
y_test (numpy.ndarray): Training set label
back
mse (float): mean square error
r2 (float): coefficient of determination
"""
def svr(X, y, test_size, random_state, kernel, C, epsilon):
    # split train and test
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size, random_state=
    svr = SVR(kernel=kernel, C=C, epsilon=epsilon)
    svr.fit(X_train, y_train)
    y_pred = svr.predict(X_test)
    mse = mean_squared_error(y_test, y_pred)
```

```
    r2 = r2_score(y_test, y_pred)
    mape = MAPE(y_test, y_pred)
    mae = MAE(y_test, y_pred)

    return svr, mse, r2, mape, mae

def best_parameter_MSE(X, y): #Mean Squared Error (MSE): 'neg_mean_squared_error'
    # initialize TimeSeriesSplit
    n_splits = 10 # 5
    tscv = TimeSeriesSplit(n_splits=n_splits)
    # defining parameters
    param_grid = {
        'C': [0.1, 0.5, 1, 2, 3, 4, 5, 10, 15, 20, 25, 50, 100, 200, 500, 1000],
        'epsilon': [0.0001, 0.0005, 0.001, 0.005, 0.01, 0.1],
        'kernel': ['rbf']
    }

    """
    7 methods for scoring: neg_mean_absolute_error, neg_mean_squared_error ,r2, neg_median_absolute_err
    neg_mean_poisson_deviance, neg_mean_gamma_deviance, neg_mean_tweedie_deviance
    """

    svr = SVR()

    grid_search = GridSearchCV(estimator=svr, param_grid=param_grid,
                               cv=tscv, scoring='neg_mean_squared_error',
                               verbose=3, n_jobs=-1)

    grid_search.fit(X, y)
    # output best parameters
    print("Best parameters found: ", grid_search.best_params_)
    print("Best cross-validation score: {:.2f}".format(np.sqrt(-grid_search.best_score_)))
    C = grid_search.best_params_['C']
    epsilon = grid_search.best_params_['epsilon']
    kernel = grid_search.best_params_['kernel']
    scoring_method = 'Mean Squared Error'
    score = np.sqrt(-grid_search.best_score_)

    return C, epsilon, kernel, scoring_method, score

def graph_with_whole_sample(svr, new_data):
    predictions = svr.predict(new_data)
    return predictions
performance = pd.DataFrame()

if __name__ == "__main__":
    # import data
    dataset_filename = ["BOE_raw.csv", "ERI_raw.csv", "GDP_raw.csv", "GDP_without_crisis_raw.csv", "Inf
    for filename in dataset_filename:
        start_time = time.time()
        X, y = dataframe_to_numpy(filename)
        scaler = StandardScaler()
        X = scaler.fit_transform(X)
        y = scaler.fit_transform(y.reshape(-1, 1))
        y = y.ravel()

        C, epsilon, kernel, scoring_method, score = best_parameter_MSE(X, y)

        """
```

```
we can change the scoring methods above by using different functions
"""
test_size = 0.2
random_state = 66

X, y = dataframe_to_numpy(filename)
scaler = StandardScaler()
X = scaler.fit_transform(X)
y = scaler.fit_transform(y.reshape(-1, 1))
y = y.ravel()

# train the model
trained_svr, mse, r2, mape, mae = svr(X, y, test_size, random_state, kernel, C, epsilon)
#  (RMSE)
rmse = np.sqrt(mse)
end_time = time.time()
elapsed_time = end_time - start_time

print("mse (mean squre error):", mse)
print("rmse (root of mse): {:.2f}".format(rmse))
print("r2: {:.2f}".format(r2))
print("running  t i m e {:.2f}s".format(elapsed_time))

new_data = X
# receiving predictions by using whole sample period's data
new_predictions = graph_with_whole_sample(trained_svr, new_data)
new_predictions = scaler.inverse_transform(new_predictions.reshape(-1, 1)).ravel()

print("This is data predicted by our model:", new_predictions)

file_path = output_path + filename
# Save the np to the CSV file
np.savetxt(file_path, new_predictions, delimiter=',', fmt='%.20f')
print("Data saved to '{}'".format(file_path))
# record performance
cleaned_output_path = output_path.rstrip('\\')
file_path_performance = os.path.join(cleaned_output_path, "performance.csv")

new_row = {"MSE": mse, "RMSE": rmse, "MAPE": mape, "MAE": mae, "Filename": filename,
           "running time": elapsed_time, "C": C,
           "kernel": kernel, "epsilon":epsilon, "scoring_method": scoring_method, "score

performance = performance.append(new_row, ignore_index=True)
performance.to_csv(file_path_performance, index=False)
print(performance)

X, y = dataframe_to_numpy(filename)

# plot the figure
plt.figure(figsize=(8, 4))
plt.plot(y, label='True value', marker='o')
plt.plot(new_predictions, label='Predicted value', linestyle='--')
plt.xlabel('Time')
plt.ylabel('Y')
plt.legend()
plt.title("True vs. Predicted:'{}'".format(filename))
file_path_plot = output_path + filename + "_plot2.png"
plt.savefig(file_path_plot, dpi=300)
```

C.1.4 Radom Forest model

```
import numpy as np
import pandas as pd
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error,mean_absolute_error
import matplotlib.pyplot as plt
from sklearn.model_selection import TimeSeriesSplit, GridSearchCV
import time
from sklearn.preprocessing import MinMaxScaler, StandardScaler
"""
    key words in RF model
    Parameters:
    - X: characteristics
    - y: target
    - test_size: train vs test = 0.8 vs 0.2
    - random_state: random seeds : 66
    - n_estimators: number of trees
    - max_depth: max depth of the tree

    Returns:
    - trained_model: well-trained model
    - mse: loss function etc
"""
import os

import_path = r"C:\Users\Yujia\Desktop\ML v3\fulfilled data\\"
output_path = r"C:\Users\Yujia\Desktop\ML v3\predicted results RF"

# defining parameters


"""
    7 methods can be used to find best hyperparameters with different scoring methods
"""
def import data(dataset filename): # import data from dataset to DataFrame format
    try:
        # Construct the full file path
        file path = import path + dataset filename
        # Read the CSV file into a pandas DataFrame
        raw data = pd.read_csv(file_path)
        print("File '{}' is successfully imported from file.".format(dataset_filename))
        return raw_data  #DataFrame format
    except FileNotFoundError as e:
        print("File '{}' not found.".format(dataset_filename))
    except Exception as e:
        print("An error occurred: {}".format(e))
def dataframe_to_numpy(dataset_filename):
     raw_data = import_data(dataset_filename)
     print(" '{}' is successfully imported from dataframe.".format(dataset_filename))
     start_time = time.time()
     # process the data to matrix
     list_dataTrue = [y for y in raw_data['TRUE']]
     y = np.array(list_dataTrue).ravel()
     columns_to_delete = ['TRUE']
     X = raw_data.drop(columns=columns_to_delete).values
     print(" '{}' is successfully change to numpy.".format(dataset_filename))
```

```
     return X,y
def MAPE(y_true, y_pred): # mean_absolute_percentage_error
    epsilon = 1e-7
    y_true = np.array(y_true)
    y_pred = np.array(y_pred)
    mape = np.mean(np.abs((y_true - y_pred) / (y_true + epsilon))) * 100
    return mape
def MAE(y_true, y_pred): # mean_absolute_error_loss
    return mean_absolute_error(y_true, y_pred)
def RF(X, y, test_size, random_state, n_estimators, max_depth):# let's train the RF model
    # split train and test
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size,random_state=r
    # create our rf model with best hyperparameters
    random_forest = RandomForestRegressor(n_estimators=n_estimators, random_state=random_state,m
    # train the model
    random_forest.fit(X_train, y_train)
    # predict using test set
    y_pred = random_forest.predict(X_test)
    # measure the performance of rf model using trained model
    mse = mean_squared_error(y_test, y_pred)# mean squre error
    mape = MAPE(y_test, y_pred)# mean_absolute_percentage_error
    mae = MAE(y_test, y_pred) # mean_absolute_error_loss
    return random_forest, mse,  mape, mae

def best_parameter_MSE(X, y): #Mean Squared Error (MSE): 'neg_mean_squared_error'
    # initialize TimeSeriesSplit
    n_splits = 10
    tscv = TimeSeriesSplit(n_splits=n_splits)

    # defining parameters
    param_grid = {
        'n_estimators': [100, 150, 200, 250, 300, 400, 500],
        'max_depth': [None, 11, 13, 15, 17, 19, 21, 23, 30, 40, 50, 60, 70, 80]
    }

    """
    7 methods for scoring: neg_mean_absolute_error, neg_mean_squared_error ,r2, neg_median_absol
    neg_mean_poisson_deviance, neg_mean_gamma_deviance, neg_mean_tweedie_deviance
    """

    rf = RandomForestRegressor() #let's say using rf
    grid_search = GridSearchCV(estimator=rf, param_grid=param_grid,
                               cv=tscv, scoring='r2',
                               verbose=3, n_jobs=-1)
    grid_search.fit(X, y)
    # output best parameters
    print("Best parameters found: ", grid_search.best_params_)
    print("Best cross-validation score: {:.2f}".format(np.sqrt(-grid_search.best_score_)))
    max_depth = grid_search.best_params_['max_depth']
    n_estimators = grid_search.best_params_['n_estimators']
    scoring_method = 'Mean Squared Error'
    score = np.sqrt(-grid_search.best_score_)
    return max_depth,n_estimators,scoring_method,score


# use for graph the difference between real obs and predicted obs
def graph_with_whole_sample(trained_model, new_data):
    predictions = trained_model.predict(new_data)
    return predictions
```

```
#let's create a new empty performance list
performance = pd.DataFrame()

if __name__ == "__main__":
    # import data
    dataset_filenames = ["BOE_raw.csv", "ERI_raw.csv", "GDP_raw.csv", "GDP_without_crisis_raw.csv", "In
    for filename in dataset_filenames:
        start_time = time.time()
        X, y = dataframe_to_numpy(filename)

        scaler  = StandardScaler()
        X = scaler.fit_transform(X)
        y = scaler.fit_transform(y.reshape(-1, 1))
        y = y.ravel()

        max_depth,n_estimators,scoring_method,score = best_parameter_MSE(X,y)
        """
        we can change the scoring methods above by using different functions
        """

        test_size = 0.2
        random_state = 66 # help you repeat our estimates and receive the same results
        # let's train it
        X, y = dataframe_to_numpy(filename)
        scaler = StandardScaler()
        X = scaler.fit_transform(X)
        y = scaler.fit_transform(y.reshape(-1, 1))
        y = y.ravel()

        trained_model, mse, mape, mae = RF(X, y,test_size,random_state,n_estimators ,max_depth)
        # calculate (RMSE)
        rmse = np.sqrt(mse)
        print("mse (mean squre error):", mse)
        print("rmse (root of mse): {:.2f}".format(rmse))

        end_time = time.time()
        elapsed_time = end_time - start_time
        print("running  t i m e {:.2f}s".format(elapsed_time))

        # using new data to predict (here we use our whole sample)
        new_data = X
        # receiving predictions by using whole sample period's data
        new_predictions = graph_with_whole_sample(trained_model, new_data)
        print("This is data predicted by our model:", new_predictions)

        file_path = output_path + filename
        # Save the np to the CSV file
        np.savetxt(file_path, new_predictions, delimiter=',', fmt='%.20f')
        print("Data saved to '{}'".format(file_path))
        #record performance

        cleaned_output_path = output_path.rstrip('\\')
        file_path_performance = os.path.join(cleaned_output_path, "performance.csv")

        new_row = {"MSE": mse, "RMSE": rmse, "MAPE": mape, "MAE": mae, "Filename": filename,
                   "running time": elapsed_time, "n_estimators": n_estimators,
                   "max_depth": max_depth, "scoring_method":scoring_method,"score":score}
```

```
performance = performance.append(new_row, ignore_index=True)
performance.to_csv(file_path_performance, index=False)
print(performance)
#plot the figure
plt.figure(figsize=(8, 4))
plt.plot(y, label='True value', marker='o')
plt.plot(new_predictions, label='Predicted value', linestyle='--')
plt.xlabel('Time')
plt.ylabel('Y')
plt.legend()
plt.title("True vs. Predicted:'{}'".format(filename))
file_path_plot = output_path + filename + "_plot2.png"
plt.savefig(file_path_plot, dpi=300)
```

References

Alberto Abadie and Maximilian Kasy. Choosing among regularized estimators in empirical economics: The risk of machine learning. *Review of Economics and Statistics*, 101(5):743–762, 2019.

Joshua Abel, Robert Rich, Joseph Song, and Joseph Tracy. The measurement and behavior of uncertainty: evidence from the ecb survey of professional forecasters. *Journal of Applied Econometrics*, 31(3):533–550, 2016.

Rakefet Ackerman and Valerie A Thompson. Meta-reasoning: Monitoring and control of thinking and reasoning. *Trends in cognitive sciences*, 21(8):607–617, 2017.

Icek Ajzen. The theory of planned behavior. *Organizational behavior and human decision processes*, 50(2):179–211, 1991.

John R Anderson. *How can the human mind occur in the physical universe?* Oxford University Press, 2009.

Rebecca R Andridge and Roderick JA Little. A review of hot deck imputation for survey non-response. *International statistical review*, 78(1):40–64, 2010.

George-Marios Angeletos and Chen Lian. Dampening general equilibrium: From micro to macro. Technical report, National Bureau of Economic Research, 2017.

Erik Angner. Economists as experts: Overconfidence in theory and practice. *Journal of Economic Methodology*, 13(1):1–24, 2006.

Kevin Arceneaux. Cognitive biases and the strength of political arguments. *American Journal of Political Science*, 56(2):271–285, 2012.

Hal R Arkes. Costs and benefits of judgment errors: Implications for debiasing. *Psychological bulletin*, 110(3):486, 1991.

Christopher J Armitage and Mark Conner. Efficacy of the theory of planned behaviour: A meta-analytic review. *British journal of social psychology*, 40(4):471–499, 2001.

Jon Scott Armstrong. *Principles of forecasting: a handbook for researchers and practitioners*, volume 30. Springer, 2001.

Willy Aspinall. Expert judgment elicitation using the classical model and excalibur. *Seventh Session of the Statistics and Risk Assessment Section's International Expert Advisory Group on Risk Modeling: Iterative Risk Assessment Processes for Policy Development Under Conditions of Uncertainty I Emerging Infectious Diseases: Round IV*, pages 1–22, 2008.

Michelle C Baddeley, Andrew Curtis, and Rachel Wood. An introduction to prior information derived from probabilistic judgements: elicitation of knowledge, cognitive bias and herding. *Geological Society, London, Special Publications*, 239(1):15–27, 2004.

Yu Bai, Yan Wang, Dayuan Qiang, Xin Yuan, Jiehui Wu, Weilong Chen, Sai Zhang, Yanru Zhang, and George Chen. Identification of nanocomposites agglomerates in scanning electron microscopy images based on semantic segmentation. *IET Nanodielectrics*, 5(2):93–103, 2022.

H Kent Baker and John R Nofsinger. Psychological biases of investors. *Financial services review*, 11(2):97–116, 2002.

Laurence Ball, N Gregory Mankiw, and Ricardo Reis. Monetary policy for inattentive economies. *Journal of monetary economics*, 52(4):703–725, 2005.

Flavio Barboza, Herbert Kimura, and Edward Altman. Machine learning models and bankruptcy prediction. *Expert Systems with Applications*, 83:405–417, 2017.

Roy Batchelor. Bias in macroeconomic forecasts. *International Journal of Forecasting*, 23 (2):189–203, 2007.

Alexandre Belloni, Victor Chernozhukov, Ivan Fernandez-Val, and Christian Hansen. Program evaluation and causal inference with high-dimensional data. *Econometrica*, 85(1):233–298, 2017.

Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 2012.

Pietro Berkes and Laurenz Wiskott. On the analysis and interpretation of inhomogeneous quadratic forms as receptive fields. *Neural computation*, 18(8):1868–1895, 2006.

Vincent Berthet. The measurement of individual differences in cognitive biases: A review and improvement. *Frontiers in psychology*, 12:630177, 2021.

Peter J Bickel and Kjell A Doksum. *Mathematical statistics: basic ideas and selected topics, volumes I-II package*. CRC Press, 2015.

Jabin Binnendyk and Gordon Pennycook. Individual differences in overconfidence: A new measurement approach. *Available at SSRN 4563382*, 2023.

Sebastian M Blanc and Thomas Setzer. Analytical debiasing of corporate cash flow forecasts. *European Journal of Operational Research*, 243(3):1004–1015, 2015.

Nicholas Bloom. The impact of uncertainty shocks. *econometrica*, 77(3):623–685, 2009.

Gianna Boero, Jeremy Smith, and Kenneth F Wallis. Uncertainty and disagreement in economic prediction: the bank of england survey of external forecasters. *The Economic Journal*, 118(530):1107–1127, 2008.

Gianna Boero, Jeremy Smith, and Kenneth F Wallis. The measurement and characteristics of professional forecasters' uncertainty. *Journal of Applied Econometrics*, 30(7): 1029–1046, 2015.

Gordon H Bower and Paul R Cohen. Emotional influences in memory and thinking: Data and theory. *Affect and cognition*, 13:291–331, 2014.

Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001a.

Leo Breiman. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, 16(3):199–231, 2001b.

J Michael Brick and Graham Kalton. Handling missing data in survey research. *Statistical methods in medical research*, 5(3):215–238, 1996.

Mario P Brito, Gwyn Griffiths, and Art Trembranis. Eliciting expert judgment on the probability of loss of an auv operating in four environments. 2008.

Robert J Budnitz, George Apostolakis, David M Boore, Lloyd S Cluff, Kevin J Coppersmith, C Allin Cornell, and Peter A Morris. Use of technical expert panels: applications to probabilistic seismic hazard analysis. *Risk Analysis*, 18(4):463–469, 1998.

Kenneth P Burnham, David R Anderson, Kenneth P Burnham, and David R Anderson. *Practical use of the information-theoretic approach*. Springer, 1998.

Goran Buturac. Measurement of economic forecast accuracy: A systematic overview of the empirical literature. *Journal of Risk and Financial Management*, 15(1):1, 2021.

Sean D Campbell and Steven A Sharpe. Anchoring bias in consensus forecasts and its effect on market prices. *Journal of Financial and Quantitative Analysis*, 44(2):369–390, 2009.

Jian Cao, Zhi Li, and Jian Li. Financial time series forecasting model based on ceemdan and lstm. *Physica A: Statistical mechanics and its applications*, 519:127–139, 2019.

Carlos Capistrán and Allan Timmermann. Disagreement and biases in inflation expectations. *Journal of Money, Credit and Banking*, 41(2-3):365–396, 2009.

Eddie Casey. Do macroeconomic forecasters use macroeconomics to forecast? *International Journal of Forecasting*, 36(4):1439–1453, 2020.

Junyi Chai and Anming Li. Deep learning in natural language processing: A state-of-the-art survey. In *2019 International Conference on Machine Learning and Cybernetics (ICMLC)*, pages 1–6. IEEE, 2019.

Tianfeng Chai and Roland R Draxler. Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature. *Geoscientific model development*, 7(3):1247–1250, 2014.

Guoli Chen, Craig Crossland, and Shuqing Luo. Making the same mistake all over again: Ceo overconfidence and corporate resistance to corrective feedback. *Strategic Management Journal*, 36(10):1513–1535, 2015.

Shaobing Chen and David Donoho. Basis pursuit. In *Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 41–44. IEEE, 1994.

M Ali Choudhary and Adnan Haider. Neural network models for inflation forecasting: an appraisal. *Applied Economics*, 44(20):2631–2635, 2012.

Frédéric Chyzak and Frank Nielsen. A closed-form formula for the kullback-leibler divergence between cauchy distributions. *arXiv preprint arXiv:1905.10965*, 2019.

Robert T Clemen. Combining forecasts: A review and annotated bibliography. *International journal of forecasting*, 5(4):559–583, 1989.

Robert T Clemen and Kenneth C Lichtendahl. Debiasing expert overconfidence: A bayesian calibration model. *PSAM6, San Juan, Puerto Rico*, 2002.

Robert T Clemen and Robert L Winkler. Combining probability distributions from experts in risk analysis. *Risk analysis*, 19:187–203, 1999.

Michael P Clements and David F Hendry. *The Oxford handbook of economic forecasting*. OUP USA, 2011.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(ARTICLE):2493–2537, 2011.

Abigail R Colson and Roger M Cooke. Expert elicitation: using the classical model to validate experts' judgments. *Review of Environmental Economics and Policy*, 2018.

Roger M Cooke. *Experts in uncertainty: opinion and subjective probability in science*. Oxford University Press, USA, 1991.

Roger M Cooke and D Solomatine. Excalibr integrated system for processing expert judgements version 3.0. *Delft University of Technology and SoLogic Delft, Delft*, 1992.

Roger M Cooke, Deniz Marti, and Thomas Mazzuchi. Expert forecasting with and without uncertainty quantification and weighting: What do the data say? *International Journal of Forecasting*, 37(1):378–387, 2021.

David R Cox. The regression analysis of binary sequences. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 20(2):215–232, 1958.

Dean Croushore. Forecasting with real-time macroeconomic data. *Handbook of economic forecasting*, 1:961–982, 2006.

Dean Croushore and Tom Stark. A real-time data set for macroeconomists: Does the data vintage matter? *Review of Economics and Statistics*, 85(3):605–617, 2003.

Andi Cupallari. *Applications of Machine Learning and Deep Learning in Macroeconomic and Financial Forecasting*. PhD thesis, City University of New York, 2020.

Kahneman Daniel. *Thinking, fast and slow*. 2017.

Tushar Kanti Das and Bing-Sheng Teng. Trust, control, and risk in strategic alliances: An integrated framework. *Organization studies*, 22(2):251–283, 2001.

Richard Deaves, Erik Lüders, and Michael Schröder. The dynamics of overconfidence: Evidence from stock market forecasters. *Journal of Economic Behavior & Organization*, 75(3):402–412, 2010.

Peter Debaere. The big three: Inflation, gdp, and unemployment. 2008.

Guido J Deboeck. *Trading on the edge: neural, genetic, and fuzzy systems for chaotic financial markets*, volume 39. John Wiley & Sons, 1994.

Luis C Dias, Alec Morton, and John Quigley. Elicitation. *Springer International Publishing. MR3700912. doi: https://doi. org/10.1007/978-3-319-65052-4*, 1(2):3, 2018.

Francis X Diebold. The past, present, and future of macroeconomic forecasting. *Journal of Economic Perspectives*, 12(2):175–192, 1998.

Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4): 1289–1306, 2006.

Norman R Draper and Harry Smith. *Applied regression analysis*, volume 326. John Wiley & Sons, 1998.

Itiel E Dror and David Charlton. Why experts make errors. *Journal of Forensic Identification*, 56(4):600, 2006.

Hillel J Einhorn. Expert judgment: Some necessary conditions and an example. *Journal of applied psychology*, 59(5):562, 1974.

Issam El Naqa and Martin J Murphy. *What is machine learning?* Springer, 2015.

Graham Elliott and Allan Timmermann. Economic forecasting. *Journal of Economic Literature*, 46(1):3–56, 2008.

Graham Elliott, Allan Timmermann, and Ivana Komunjer. Estimation and testing of forecast rationality under flexible loss. *The Review of Economic Studies*, 72(4):1107–1125, 2005.

Tlamelo Emmanuel, Thabiso Maupong, Dimane Mpoeleng, Thabo Semong, Banyatsang Mphago, and Oteng Tabona. A survey on missing data in machine learning. *Journal of Big Data*, 8(1):1–37, 2021.

Joseph Engelberg, Charles F Manski, and Jared Williams. Comparing the point predictions and subjective probability distributions of professional forecasters. *Journal of Business & Economic Statistics*, 27(1):30–41, 2009.

Nicholas Epley and Thomas Gilovich. The anchoring-and-adjustment heuristic: Why the adjustments are insufficient. *Psychological science*, 17(4):311–318, 2006.

Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

K Anders Ericsson and James J Staszewski. Skilled memory and expertise: Mechanisms of exceptional performance. In *Complex information processing*, pages 255–288. Psychology Press, 2013.

William Feller. *An introduction to probability theory and its applications, Volume 2*, volume 81. John Wiley & Sons, 1991.

Stefan Feuerriegel and Julius Gordon. News-based forecasts of macroeconomic indicators: A semantic path model for interpretable predictions. *European Journal of Operational Research*, 272(1):162–175, 2019.

Rebecca Fiebrink. Machine learning education for artists, musicians, and other creative practitioners. *ACM Transactions on Computing Education (TOCE)*, 19(4):1–32, 2019.

Robert Fildes and Paul Goodwin. Against your better judgment? how organizations can improve their use of management judgment in forecasting. *Interfaces*, 37(6):570–576, 2007.

Robert Fildes and Herman Stekler. The state of macroeconomic forecasting. *Journal of macroeconomics*, 24(4):435–468, 2002.

Melissa L Finucane, Ali Alhakami, Paul Slovic, and Stephen M Johnson. The affect heuristic in judgments of risks and benefits. *Journal of behavioral decision making*, 13 (1):1–17, 2000.

Baruch Fischhoff, Paul Slovic, and Sarah Lichtenstein. Fault trees: Sensitivity of estimated failure probabilities to problem representation. In *Judgment and Decision Making*, pages 124–145. Routledge, 2013.

Kenneth L Fisher and Meir Statman. Cognitive biases in market forecasts. *Journal of Portfolio Management*, 27(1):72–81, 2000.

Valerie S Folkes. The availability heuristic and perceived risk. *Journal of Consumer research*, 15(1):13–23, 1988.

Jordi Galí. Are central banks' projections meaningful? *Journal of Monetary Economics*, 58(6-8):537–550, 2011.

Michail Giannakos, Iro Voulgari, Sofia Papavlasopoulou, Zacharoula Papamitsiou, and Georgios Yannakakis. Games for artificial intelligence and machine learning education: Review and perspectives. *Non-formal and informal science learning in the ICT era*, pages 117–133, 2020.

Gerd Gigerenzer. How to make cognitive illusions disappear: Beyond "heuristics and biases". *European review of social psychology*, 2(1):83–115, 1991.

Gerd Gigerenzer and Wolfgang Gaissmaier. Heuristic decision making. *Annual review of psychology*, 62:451–482, 2011.

Thomas Gilovich, Dale Griffin, and Daniel Kahneman. *Heuristics and biases: The psychology of intuitive judgment*. Cambridge university press, 2002.

Periklis Gogas and Theophilos Papadimitriou. Machine learning in economics and finance. *Computational Economics*, 57:1–4, 2021.

Lewis R Goldberg. The effectiveness of clinicians' judgments: The diagnosis of organic brain damage from the bender-gestalt test. *Journal of Consulting Psychology*, 23(1):25, 1959.

Philippe Goulet Coulombe, Maxime Leroux, Dalibor Stevanovic, and Stéphane Surprenant. How is machine learning useful for macroeconomic forecasting? *Journal of Applied Econometrics*, 37(5):920–964, 2022.

Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

David M Grether. Bayes rule as a descriptive model: The representativeness heuristic. *The Quarterly journal of economics*, 95(3):537–557, 1980.

David M Grether. Testing bayes rule and the representativeness heuristic: Some experimental evidence. *Journal of Economic Behavior & Organization*, 17(1):31–57, 1992.

Yanming Guo, Yu Liu, Ard Oerlemans, Songyang Lao, Song Wu, and Michael S Lew. Deep learning for visual understanding: A review. *Neurocomputing*, 187:27–48, 2016.

Refet S Gürkaynak, Brian P Sack, and Eric T Swanson. Do actions speak louder than words? the response of asset prices to monetary policy actions and statements. *The Response of Asset Prices to Monetary Policy Actions and Statements (November 2004)*, 2004.

Anca M Hanea, Mark Burgman, and Victoria Hemming. Idea for uncertainty quantification. *Elicitation: The science and art of structuring judgement*, pages 95–117, 2018.

Anca M Hanea, Gabriela F Nane, Tim Bedford, and Simon French. *Expert judgement in risk and decision analysis*. Springer, 2021.

Jason Hartford, Greg Lewis, Kevin Leyton-Brown, and Matt Taddy. Counterfactual prediction with deep instrumental variables networks. *arXiv preprint arXiv:1612.09596*, 2016.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Ullrich Heilemann and Herman Stekler. Introduction to "the future of macroeconomic forecasting", 2007.

Dieter Hess and Sebastian Orbe. Irrationality or efficiency of macroeconomic survey forecasts? implications from the anchoring bias test. *Review of Finance*, 17(6):2097–2131, 2013.

Martin Hilbert. Toward a synthesis of cognitive biases: how noisy information processing can bias human decision making. *Psychological bulletin*, 138(2):211, 2012.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Jian Huang, Junyi Chai, and Stella Cho. Deep learning in finance and banking: A literature review and classification. *Frontiers of Business Research in China*, 14(1):1–24, 2020.

Rong Huang, Keith Pilbeam, and William Pouliot. Are macroeconomic forecasters optimists or pessimists? a reassessment of survey based forecasts. *Journal of Economic Behavior & Organization*, 197:706–724, 2022.

Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pages 492–518. Springer, 1992.

Rob J Hyndman and Anne B Koehler. Another look at measures of forecast accuracy. *International journal of forecasting*, 22(4):679–688, 2006.

Atsushi Inoue and Lutz Kilian. How useful is bagging in forecasting economic time series? a case study of us consumer price inflation. *Journal of the American Statistical Association*, 103(482):511–522, 2008.

Daniel J Isenberg. Group polarization: A critical review and meta-analysis. *Journal of personality and social psychology*, 50(6):1141, 1986.

W Jos Jansen, Xiaowen Jin, and Jasper M de Winter. Forecasting and nowcasting real gdp: Comparing statistical models and subjective forecasts. *International Journal of Forecasting*, 32(2):411–436, 2016.

Soojin Jo and Rodrigo Sekkel. Macroeconomic uncertainty through the lens of professional forecasters. *Journal of Business & Economic Statistics*, 37(3):436–446, 2019.

Norman L Johnson, Samuel Kotz, and Narayanaswamy Balakrishnan. *Continuous univariate distributions, volume 2*, volume 289. John wiley & sons, 1995.

Jin-Kyu Jung, Manasa Patnam, and Anna Ter-Martirosyan. *An algorithmic crystal ball: Forecasts-based on machine learning*. International Monetary Fund, 2018.

Kyle Jurado, Sydney C Ludvigson, and Serena Ng. Measuring uncertainty. *American Economic Review*, 105(3):1177–1216, 2015.

Daniel Kahneman. *Thinking, fast and slow*. macmillan, 2011.

Daniel Kahneman and Ilana Ritov. Determinants of stated willingness to pay for public goods: A study in the headline method. *Journal of Risk and Uncertainty*, 9:5–37, 1994.

Daniel Kahneman, Ilana Ritov, David Schkade, Steven J Sherman, and Hal R Varian. Economic preferences or attitude expressions?: an analysis of dollar responses to public issues. *Elicitation of preferences*, pages 203–242, 2000.

Sheridan Kamal. An analysis of machine learning techniques for economic recession prediction. 2021.

Stan Kaplan. 'expert information'versus 'expert opinions'. another approach to the problem of eliciting/combining/using expert knowledge in pra. *Reliability Engineering & System Safety*, 35(1):61–72, 1992.

Andreas Kappes, Ann H Harvey, Terry Lohrenz, P Read Montague, and Tali Sharot. Confirmation bias in the utilization of others' opinion strength. *Nature neuroscience*, 23(1):130–137, 2020.

Markku Kaustia and Milla Perttula. Overconfidence and debiasing in the financial industry. *Review of Behavioural Finance*, 4(1):46–62, 2012.

Michael P Keane and David E Runkle. Testing the rationality of price forecasts: New evidence from panel data. *The American Economic Review*, pages 714–735, 1990.

Carmen Keller, Michael Siegrist, and Heinz Gutscher. The role of the affect and availability heuristics in risk communication. *Risk analysis*, 26(3):631–639, 2006.

Christopher A Kelly and Tali Sharot. Individual differences in information-seeking. *Nature communications*, 12(1):7062, 2021.

John Maynard Keynes. The general theory of employment. *The quarterly journal of economics*, 51(2):209–223, 1937.

Abdullah Ayub Khan, Asif Ali Laghari, and Shafique Ahmed Awan. Machine learning in computer vision: a review. *EAI Endorsed Transactions on Scalable Information Systems*, 8(32):e4–e4, 2021.

Yonghoon Kim and Mokdong Chung. An approach to hyperparameter optimization for the objective function in machine learning. *Electronics*, 8(11):1267, 2019.

Sotiris B Kotsiantis, Dimitris Kanellopoulos, and Panagiotis E Pintelas. Data preprocessing for supervised leaning. *International journal of computer science*, 1(2):111–117, 2006.

Solomon Kullback. *Information theory and statistics*. Courier Corporation, 1997.

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Mary Kynn. The 'heuristics and biases' bias in expert elicitation. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 171(1):239–264, 2008.

Jérôme Lambert, Véronique Bessière, and Gilles N'Goala. Does expertise influence the impact of overconfidence on judgment, valuation and investment decision? *Journal of Economic Psychology*, 33(6):1115–1128, 2012.

Michael Lawrence, Paul Goodwin, Marcus O'Connor, and Dilek Önkal. Judgmental forecasting: A review of progress over the last 25 years. *International Journal of forecasting*, 22(3):493–518, 2006.

Tae-Hwy Lee, Halbert White, and Clive WJ Granger. Testing for neglected nonlinearity in time series models: A comparison of neural network methods and alternative tests. *Journal of econometrics*, 56(3):269–290, 1993.

Sarah Lichtenstein, Baruch Fischhoff, and Lawrence D Phillips. Calibration of probabilities: The state of the art. In *Decision Making and Change in Human Affairs: Proceedings of the Fifth Research Conference on Subjective Probability, Utility, and Decision Making, Darmstadt, 1–4 September, 1975*, pages 275–324. Springer, 1977.

Nicholas Light, Philip M Fernbach, Nathaniel Rabb, Mugur V Geana, and Steven A Sloman. Knowledge overconfidence is associated with anti-consensus views on controversial scientific issues. *Science Advances*, 8(29):eabo0038, 2022.

Shi-Woei Lin and Vicki M Bier. A study of expert overconfidence. *Reliability Engineering & System Safety*, 93(5):711–721, 2008.

Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

Roderick JA Little and Donald B Rubin. Bayes and multiple imputation. *Statistical analysis with missing data*, pages 200–220, 2002.

Roderick JA Little, Nathaniel Schenker, Gerhard Arminger, Clifford C Clogg, and Michael E Sobel. Handbook of statistical modeling for the social and behavioral sciences. *G. Arminger, CC Clogg, & ME Sobel (Eds.)*, pages 39–75, 1995.

George F Loewenstein, Elke U Weber, Christopher K Hsee, and Ned Welch. Risk as feelings. *Psychological bulletin*, 127(2):267, 2001.

Chi-Jie Lu, Tian-Shyug Lee, and Chih-Chou Chiu. Financial time series forecasting using independent component analysis and support vector regression. *Decision support systems*, 47(2):115–125, 2009.

Robert E Lucas. Some international evidence on output-inflation tradeoffs. *The American economic review*, pages 326–334, 1973.

Benjamin A Lyons, Jacob M Montgomery, Andrew M Guess, Brendan Nyhan, and Jason Reifler. Overconfidence in news judgments is associated with false news susceptibility. *Proceedings of the National Academy of Sciences*, 118(23):e2019527118, 2021.

George D Magoulas and Andriana Prentza. Machine learning in medical applications. In *Advanced course on artificial intelligence*, pages 300–307. Springer, 1999.

Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015.

N Gregory Mankiw, Ricardo Reis, and Justin Wolfers. Disagreement about inflation expectations. *NBER macroeconomics annual*, 18:209–248, 2003.

Charles F Manski. Measuring expectations. *Econometrica*, 72(5):1329–1376, 2004.

Deniz Marti, Thomas A Mazzuchi, and Roger M Cooke. Are performance weights beneficial? investigating the random expert hypothesis. *Expert Judgement in Risk and Decision Analysis*, pages 53–82, 2021.

Dmytro Matsypura, Ryan Thompson, and Andrey L Vasnev. Optimal selection of expert forecasts with integer programming. *Omega*, 78:165–175, 2018.

Thomas McAndrew, Nutcha Wattanachit, Graham C Gibson, and Nicholas G Reich. Aggregating predictions from experts: A review of statistical methods, experiments, and applications. *Wiley Interdisciplinary Reviews: Computational Statistics*, 13(2):e1514, 2021.

Sara S McMillan, Michelle King, and Mary P Tully. How to use the nominal group and delphi techniques. *International journal of clinical pharmacy*, 38:655–662, 2016.

Marcelo C Medeiros, Gabriel FR Vasconcelos, Álvaro Veiga, and Eduardo Zilberman. Forecasting inflation in a data-rich environment: the benefits of machine learning methods. *Journal of Business & Economic Statistics*, 39(1):98–119, 2021.

Thomas Mehle. Hypothesis generation in an automobile malfunction inference task. *Acta Psychologica*, 52(1-2):87–106, 1982.

Daniele Melati, Yuri Grinberg, Mohsen Kamandar Dezfouli, Siegfried Janz, Pavel Cheben, Jens H Schmid, Alejandro Sánchez-Postigo, and Dan-Xia Xu. Mapping the global design space of nanophotonic components using machine learning pattern recognition. *Nature communications*, 10(1):4775, 2019.

George Milunovich. Forecasting australia's real house price index: A comparison of time series and machine learning methods. *Journal of Forecasting*, 39(7):1098–1118, 2020.

Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital signal processing*, 73:1–15, 2018.

Gilberto Montibeller and Detlof Von Winterfeldt. Cognitive and motivational biases in decision and risk analysis. *Risk analysis*, 35(7):1230–1251, 2015.

Gilberto Montibeller and Detlof von Winterfeldt. Individual and group biases in value and uncertainty judgments. *Elicitation: The science and art of structuring judgement*, pages 377–392, 2018.

Don A Moore and Paul J Healy. The trouble with overconfidence. *Psychological review*, 115(2):502, 2008.

Don A Moore and Derek Schatz. The three faces of overconfidence. *Social and Personality Psychology Compass*, 11(8):e12331, 2017.

Millett Granger Morgan, Max Henrion, and Mitchell Small. *Uncertainty: a guide to dealing with uncertainty in quantitative risk and policy analysis*. Cambridge university press, 1990.

Peter A Morris. Decision analysis expert use. *Management Science*, 20(9):1233–1241, 1974.

Peter A Morris. Combining expert judgments: A bayesian approach. *Management Science*, 23(7):679–693, 1977.

Saeed Moshiri and Norman Cameron. Neural network versus econometric models in forecasting inflation. *Journal of forecasting*, 19(3):201–217, 2000.

Julien Mostard, Ruud Teunter, and Rene De Koster. Forecasting demand for single-period products: A case study in the apparel industry. *European Journal of Operational Research*, 211(1):139–147, 2011.

Rachael M Moyer and Geoboo Song. Understanding local policy elites' perceptions on the benefits and risks associated with high-voltage power line installations in the state of arkansas. *Risk Analysis*, 36(10):1983–1999, 2016.

Sendhil Mullainathan and Jann Spiess. Machine learning: an applied econometric approach. *Journal of Economic Perspectives*, 31(2):87–106, 2017.

Emi Nakamura. Inflation forecasting using a neural network. *Economics Letters*, 86(3): 373–378, 2005.

Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4467–4477, 2017.

Feng Ni, David Arnott, and Shijia Gao. The anchoring effect in business intelligence supported decision-making. *Journal of Decision Systems*, 28(2):67–81, 2019.

Frank Nielsen and Kazuki Okamura. On f-divergences between cauchy distributions. *IEEE Transactions on Information Theory*, 69(5):3150–3171, 2022.

Gregory B Northcraft and Margaret A Neale. Experts, amateurs, and real estate: An anchoring-and-adjustment perspective on property pricing decisions. *Organizational behavior and human decision processes*, 39(1):84–97, 1987.

Anthony O'Hagan, Caitlin E Buck, Alireza Daneshkhah, J Richard Eiser, Paul H Garthwaite, David J Jenkinson, Jeremy E Oakley, and Tim Rakow. Uncertain judgements: eliciting experts' probabilities. 2006.

Michael Oppenheimer, Christopher M Little, and Roger M Cooke. Expert judgement and uncertainty quantification for climate change. *Nature climate change*, 6(5):445–451, 2016.

Pietro Ortoleva and Erik Snowberg. Overconfidence in political behavior. *American Economic Review*, 105(2):504–535, 2015.

Harry Otway and Detlof von Winterfeldt. Expert judgment in risk analysis and management: process, context, and pitfalls. *Risk analysis*, 12(1):83–93, 1992.

Anthony O'Hagan. Expert knowledge elicitation: subjective but scientific. *The American Statistician*, 73(sup1):69–81, 2019.

Luigi Paciello and Mirko Wiederholt. Exogenous information, endogenous information, and optimal monetary policy. *Review of Economic Studies*, 81(1):356–388, 2014.

Jitesh H Panchal, Mark Fuge, Ying Liu, Samy Missoum, and Conrad Tucker. Machine learning for engineering design. *Journal of Mechanical Design*, 141(11):110301, 2019.

Jose Manuel Pereira, Mario Basto, and Amelia Ferreira Da Silva. The logistic lasso and ridge regression in predicting corporate failure. *Procedia Economics and Finance*, 39: 634–641, 2016.

Uwe Peters. What is the function of confirmation bias? *Erkenntnis*, 87(3):1351–1376, 2022.

Elizabeth A Phelps, Karolina M Lempert, and Peter Sokol-Hessner. Emotion and decision making: multiple modulatory neural circuits. *Annual review of neuroscience*, 37: 263–287, 2014.

Lawrence D Phillips and Maryann C Phillips. Faciliated work groups: theory and practice. *Journal of the Operational Research Society*, 44(6):533–549, 1993.

Gloria Phillips-Wren, Daniel J Power, and Manuel Mora. Cognitive bias, decision styles, and risk attitudes in decision making and dss, 2019.

Arthur C Pigou. Mr. jm keynes' general theory of employment, interest and money. *Economica*, 3(10):115–132, 1936.

Vasilios Plakandaras, Rangan Gupta, Periklis Gogas, and Theophilos Papadimitriou. Forecasting the us real house price index. *Economic Modelling*, 45:259–267, 2015.

Dale J Poirier. *Intermediate statistics and econometrics: a comparative approach*. Mit Press, 1995.

Markus Porthin, Tony Rosqvist, and Susanna Kunttu. Risk assessment using group elicitation: Case study on start-up of a new logistics system. *Elicitation: The Science and Art of Structuring Judgement*, pages 511–527, 2018.

Philipp Probst, Marvin N Wright, and Anne-Laure Boulesteix. Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery*, 9(3):e1301, 2019.

John Quigley, Abigail Colson, Willy Aspinall, and Roger M Cooke. Elicitation in the classical model. *Elicitation: The science and art of structuring judgement*, pages 15–36, 2018.

Rajagopal Raghunathan and Michel Tuan Pham. All negative moods are not equal: Motivational influences of anxiety and sadness on decision making. *Organizational behavior and human decision processes*, 79(1):56–77, 1999.

Morten Arendt Rasmussen and Rasmus Bro. A tutorial on the lasso approach to sparse modeling. *Chemometrics and Intelligent Laboratory Systems*, 119:21–31, 2012.

Jafar Rezaei. Anchoring bias in eliciting attribute weights and values in multi-attribute decision-making. *Journal of Decision Systems*, 30(1):72–96, 2021.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*, 2016.

Muhammad Suhail Rizwan, Ghufran Ahmad, and Dawood Ashraf. Systemic risk: The impact of covid-19. *Finance Research Letters*, 36:101682, 2020.

Nadim N Rouhana, Anne O'Dwyer, and Sharon K Morrison Vaso. Cognitive biases and political party affiliation in intergroup conflict. *Journal of Applied Social Psychology*, 27 (1):37–57, 1997.

Gene Rowe and George Wright. Differences in expert and lay judgments of risk: myth or reality? *Risk analysis*, 21(2):341–356, 2001.

J Edward Russo, Paul JH Schoemaker, et al. Managing overconfidence. *Sloan management review*, 33(2):7–17, 1992.

Scott Schuh et al. An evaluation of recent macroeconomic forecast errors. *New England Economic Review*, pages 35–36, 2001.

Charles R Schwenk. Cognitive simplification processes in strategic decision-making. *Strategic management journal*, 5(2):111–128, 1984.

Georgios Sermpinis, Charalampos Stasinakis, Konstantinos Theofilatos, and Andreas Karathanasopoulos. Inflation and unemployment forecasting with genetic support vector regression. *Journal of Forecasting*, 33(6):471–487, 2014.

Nigel R Shadbolt, Paul R Smart, J Wilson, and S Sharples. Knowledge elicitation. *Evaluation of human work*, pages 163–200, 2015.

J SHANTEAU. Psychological strategies of expert decision makers. In *BULLETIN OF THE PSYCHONOMIC SOCIETY*, volume 26, pages 523–523. PSYCHONOMIC SOC INC 1710 FORTVIEW RD, AUSTIN, TX 78704, 1988.

G Shobana and K Umamaheswari. Forecasting by machine learning techniques and econometrics: A review. In *2021 6th international conference on inventive computation technologies (ICICT)*, pages 1010–1016. IEEE, 2021.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Dorota Skala. Overconfidence in psychology and finance-an interdisciplinary literature review. *Bank I kredyt*, (4):33–50, 2008.

Patrycja Sleboda and Carl-Johan Lagerkvist. Tailored communication changes consumers' attitudes and product preferences for genetically modified food. *Food Quality and Preference*, 96:104419, 2022.

Jack B Soll and Joshua Klayman. Overconfidence in interval estimates. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 30(2):299, 2004.

Yan-Yan Song and LU Ying. Decision tree methods: applications for classification and prediction. *Shanghai archives of psychiatry*, 27(2):130, 2015.

Tom Stark and Dean Croushore. Forecasting with a real-time data set for macroeconomists. *Journal of Macroeconomics*, 24(4):507–531, 2002.

Richard J Taffler. The representativeness heuristic. *Behavioral finance: Investors, corporations, and markets*, pages 259–276, 2010.

Adi L Tarca, Vincent J Carey, Xue-wen Chen, Roberto Romero, and Sorin Drăghici. Machine learning and its applications to biology. *PLoS computational biology*, 3(6): e116, 2007.

Philip E Tetlock. Cognitive biases and organizational correctives: Do both disease and cure depend on the politics of the beholder? *Administrative Science Quarterly*, 45(2): 293–326, 2000.

Valerie A Thompson, Jamie A Prowse Turner, and Gordon Pennycook. Intuition, reason, and metacognition. *Cognitive psychology*, 63(3):107–140, 2011.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.

Allan Timmermann. Forecast combinations. *Handbook of economic forecasting*, 1:135–196, 2006.

D Trumbo, CALVIN Adams, M Milner, and LOWELL Schipper. Reliability and accuracy in the inspection of hard red winter wheat. *Cereal Science Today*, 7:62–71, 1962.

Hugues Turbé, Mina Bjelogrlic, Christian Lovis, and Gianmarco Mengaldo. Evaluation of post-hoc interpretability methods in time-series classification. *Nature Machine Intelligence*, 5(3):250–260, 2023.

Murray Turoff and Harold A Linstone. The delphi method-techniques and applications. 2002.

Amos Tversky and Daniel Kahneman. Availability: A heuristic for judging frequency and probability. *Cognitive psychology*, 5(2):207–232, 1973.

Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases: Biases in judgments reveal some heuristics of thinking under uncertainty. *science*, 185(4157):1124–1131, 1974.

Terry Ursacki and Ilan Vertinsky. Choice of entry timing and scale by foreign banks in japan and korea. *Journal of Banking & Finance*, 16(2):405–421, 1992.

Stef Van Buuren. *Flexible imputation of missing data*. CRC press, 2018.

Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1999.

Hal R Varian. Big data: New tricks for econometrics. *Journal of economic perspectives*, 28 (2):3–28, 2014.

Elena Verdolini, Laura Díaz Anadón, Erin Baker, Valentina Bosetti, and Lara Aleluia Reis. Future prospects for energy technologies: insights from expert elicitations. *Review of Environmental Economics and Policy*, 2018.

Joseph A Vitriol and Jessecae K Marsh. The illusion of explanatory depth and endorsement of conspiracy beliefs. *European Journal of Social Psychology*, 48(7):955–969, 2018.

Vladimir Vovk. The fundamental nature of the log loss function. *Fields of logic and computation II: Essays dedicated To Yuri Gurevich on the Occasion of His 75th Birthday*, pages 307–318, 2015.

Qi Wang, Yue Ma, Kun Zhao, and Yingjie Tian. A comprehensive survey of loss functions in machine learning. *Annals of Data Science*, pages 1–26, 2020.

Richard F West, Maggie E Toplak, and Keith E Stanovich. Heuristics and biases as measures of critical thinking: Associations with cognitive ability and thinking dispositions. *Journal of educational psychology*, 100(4):930, 2008.

Cort J Willmott and Kenji Matsuura. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate research*, 30(1):79–82, 2005.

William F Wright and Gordon H Bower. Mood effects on subjective probability assessment. *Organizational behavior and human decision processes*, 52(2):276–291, 1992.

SAM Yaser and AF Atiya. Introduction to financial forecasting, applied intelligence. 1996.

Jaehyun Yoon. Forecasting of real gdp growth using machine learning models: Gradient boosting and random forest approach. *Computational Economics*, 57(1):247–265, 2021.

Steven R Young, Derek C Rose, Thomas P Karnowski, Seung-Hwan Lim, and Robert M Patton. Optimizing deep learning hyper-parameters through an evolutionary algorithm. In *Proceedings of the workshop on machine learning in high-performance computing environments*, pages 1–5, 2015.

Zhongheng Zhang. Missing data imputation: focusing on single imputation. *Annals of translational medicine*, 4(1), 2016.