16:958:588:02
Data Minning Final Project

Selected project paper:
Yang, Z. et al. (2024)
Bellman Conformal Inference: Calibrating Prediction Intervals for Time Series

Instructor:
Linjun Zhang

By:
Arun Mishra (am3464)

Introduction

Prediction interval is a crucial technique in quantifying uncertainty in time series forecasting. The well-calibrated prediction interval would accurately capture this uncertainty by ensuring that the proportion of intervals containing true outcomes matches a predefined coverage level over time. However, traditional model-based prediction intervals often suffer from poor calibration as they heavily rely on underlying assumptions of the statistical models. These models may struggle with mis-specification due to nonstationarity or changing environments, leading to inaccurate prediction intervals.

Conformal Inference, on the other hand, offers a distribution-free approach to uncertainty quantification, constructing robust prediction intervals based solely on the observed data. While this method is more resilient to violations of distributional assumptions, achieving calibration with conformal inference is not always guaranteed due to the complex nature of temporal dependencies and shifts in distributions. Despite numerous variants proposed under weaker assumptions, many still require precise model estimates, limited distribution shifts, restrictive dependence structure, or multiple independent copies of the time series.

In response to these challenges, Adaptive Conformal Inference (ACI) has emerged as a promising method for generating approximately calibrated prediction intervals without specific assumptions about underlying time series. ACI provides a flexible framework that adapts its nominal miscoverage rate over time to achieve calibration. By formulating ACI as an online gradient descent algorithm, the nominal miscoverage rate is adjusted iteratively based on the observed errors, which allows ACI to continuously refine its prediction intervals and improve calibration performance. Nonetheless, ACI lacks a mechanism to explicitly optimize the average interval lengths and may encounter issues regarding the fraction of infinite-length prediction intervals.

To address these limitations, Bellman Conformal Inference (BCI) has been proposed as an extension of ACI, incorporating multi-step ahead prediction intervals within a stochastic control problem (SCP) framework. Drawing inspiration from Model Predictive Control (MPC) principles, BCI models the dynamics of the forecasting process and optimizes interval lengths, thereby enhancing adaptability and robustness of prediction intervals. By formulating an objective function that trades off between the average length of multi-step prediction intervals and the estimated future coverage by treating the nominal coverage rate as an action at each time point, BCI explicitly considers calibration alongside coverage guarantees, aiming to provide more reliable and dynamically adjustable prediction intervals.

Background Needed to Understand BCI

Conformal Inference

Conformal inference constitutes a machine learning framework utilized for the establishment of true outcome prediction sets. A fundamental assumption within this framework is that of exchangeability, wherein the joint distribution remains invariant under different permutations of data points. This property implies that the ordering or arrangement of the data does not affect the underlying distribution.

Consider a scenario involving a linear regression model applied to a dataset comprising a best-fit line and multiple data points. The objective is to develop a score function that accounts for data uncertainty. A straightforward example of such a score function is the calculation of the distance between the predicted values and the observed y-values. Upon computing scores for all data points, a score set is generated and subsequently ranked. A specified quantile (e.g., 90%) is then used to filter out a subset, constituting the true outcome prediction set.

$$S(X_t, y) = |\hat{\mu}(X_t) - y|.$$

Alternative score function: Conformal Quantile Regression (CQR):

$$S(X_t, y) = \max\{\hat{q}(X_t; \alpha/2) - y, y - \hat{q}(X_t; 1 - \alpha/2)\}.$$

However, complications arise when dealing with non-exchangeable data, such as in time series analysis, where the underlying distribution may shift over time. In such cases, adjustments to the score function and quantile thresholds are necessary for each distinct time lag to accommodate the temporal dynamics inherent in the data. This adaptation ensures the integrity and applicability of the conformal inference methodology within the context of time-dependent dataset.

Adaptive Conformal Inference (ACI)

The ACI method involves an iterative process aimed at refining prediction intervals based on observed errors and coverage rates. This algorithm systematically adjusts interval widths in response to past performance metrics. Specifically, an error indicator is defined (err_t)which takes the value 1 if the points from the last time lag were not encompassed within the interval, and 0 if they were.

When an error occurs (i.e., err_t=1), indicating that points were excluded, it is infered that the interval was too narrow. To address this, interval is widen after considering a higher quantile (corresponding to a smaller α_t). Conversely, if no error is observed (i.e., err_t=0), indicating successful inclusion of points, the interval can be contemplated for narrowing.

This adaptive approach allows for continuous adjustment of prediction intervals, optimizing them based on observed outcomes and thus enhancing the overall accuracy and reliability of the inference process.

$$\text{err}_t := \begin{cases} 1, \text{ if } Y_t \notin \hat{C}_t(\alpha_t), \\ 0, \text{ otherwise,} \end{cases} \quad \text{where } \hat{C}_t(\alpha_t) := \{y : S_t(X_t, y) \leq \hat{Q}_t(1 - \alpha_t)\}.$$

Then, fixing a step size parameter $\gamma > 0$ we consider the simple online update

$$\alpha_{t+1} := \alpha_t + \gamma(\alpha - \text{err}_t). \tag{2}$$

We refer to this algorithm as *adaptive conformal inference*. Here, err_t plays the role of our estimate of the historical miscoverage frequency. A natural alternative to this is the update

$$\alpha_{t+1} = \alpha_t + \gamma \left(\alpha - \sum_{s=1}^{t} w_s \text{err}_s \right), \tag{3}$$

Dynamic Programming

Dynamic programming is a method used in computer science to solve problems by breaking them down into simpler subproblems and solving each of those subproblems just once, storing their solutions. The main idea is to avoid calculating the same thing multiple times, which saves a lot of time and computing resources

Model Predictive Control (MPC)

Model Predictive Control (MPC) is a type of control strategy used primarily in industrial and process engineering to manage complex systems. The basic idea behind MPC is to predict the future behavior of a system over a short time horizon and then use this prediction to choose the best possible action to take right now.

Bellman Conformal Inference (BCI)

Bellman Conformal Inference (BCI) is an innovative framework designed to enhance the reliability of prediction intervals in time series forecasting. This method intelligently integrates the robustness of conformal inference with dynamic programming principles derived from Bellman's equation, making it exceptionally suited for complex forecasting tasks where anticipating the evolution of outcomes over time is crucial

Method

The paper introduces three models for time series forecasting and applies to three distinct datasets to evaluate their performance. The ARMA model is implemented as the first model to forecast daily stock return for companies AMD, Amazon, and Nvidia. For this model, the authors utilize the one-day log return of each company's stock as the target variable and the lagged sequences of log returns on each day t as predictors. Then they employ a decoder-only transformer of embedding size 16 with 8 heads and 2 layers which leads to a low-dimensional self-attention operation suitable for univariate time series. For the output, they use a linear layer to map the embedding to a 2×T dimensional vector, representing the predicted mean and standard deviation of the for the next T days. The model is trained using an optimization problem that minimizes the negative log-likelihood of the observed returns under a normal distribution assumption. The resulting prediction intervals capture the uncertainty in the forecasted mean, with the width of the interval determined by the forecasted standard deviation and the desired coverage level $1-\beta$.

The second model addresses volatility forecasting on Amazon stock data using the GARCH model. The squared volatility of each company's stock is computed as the target variable and the lagged sequences of squared volatilities as predictors. The parameters of the GARCH(1,1) model are estimated using maximum likelihood estimation, and prediction intervals are constructed based on the quantiles of the chi-squared distribution.

And the third model applies sequence-to-sequence neural net to forecast the popularity of the keyword "deep learning" in Google search trends. The lagged sequences of search popularity are used as predictors, and the 5-layer LSTM recurrent neural network is trained using an optimization problem to minimize the negative log-likelihood of observed search popularity under a normal distribution assumption. Prediction intervals are then constructed based on the predicted mean and standard deviation output by the LSTM network.

In addition to model constructions, two methods for generating prediction intervals - ACI and BCI are introduced and compared among each model. For the metrics of evaluation, the paper used local average of measures over a moving window of size 500 and an approach for fair comparison. For each step size γ, the paper performs a grid search on the step size of BCI, and chooses the best one (match sample variance). In the Baseline Conformal Inference (BCI) framework, two foundational assumptions, monotonicity and safeguard, critically shape the optimization of prediction intervals. Monotonicity ensures that increasing the coverage expectation by decreasing the nominal miscoverage rate (β) results in wider prediction intervals, making them more conservative. Conversely, the safeguard assumption guarantees that at the maximum coverage level (100% expectation), the prediction interval includes all possible future observations, thus ensuring complete theoretical coverage of future data points (Y). These assumptions are essential for maintaining the statistical integrity and practical utility of the BCI method.

Below is the optimization problem for BCI which is solved using both the assumptions as boundary conditions.

$$\min_{\alpha_{t|t},\ldots,\alpha_{t+T-1|t}} \mathbb{E}_{\substack{(\beta_{t|t},\ldots,\beta_{t+T-1|t}) \\ \sim F_{t|t}\otimes\ldots F_{t+T-1|t}}} \left[\overbrace{\sum_{s=t}^{t+T-1} L_{s|t}(\alpha_{s|t})}^{\text{Efficiency: interval length}} + \lambda_t \underbrace{\max\left(\frac{1}{T}\sum_{s=t}^{t+T-1} \text{err}_{s|t} - \bar{\alpha}, 0\right)}_{\text{Validity: miscoverage rate}} \right],$$

Algorithm

Implementing BCI involves several clear steps that help ensure your predictive models are adaptive and accurate:

1. Initialization:

- **Set Hyperparameters**: Determine key settings: the target miscoverage level (α), how far ahead you want to predict (T for receding horizon), the maximum allowable weight (λ_{max}) and relative step size $c \in (0, 1)$.
- **Prepare Initial Data**: Use estimated functions that show the likelihood distribution of future prediction errors (marginal cumulative distribution functions, or CDFs) and set up the rules for how long your prediction intervals should be (length functions).

2. Update Security Parameter:

- **Adjust λ_t**: Update the control parameter using the formula $\lambda_t = \lambda_{t-1} - \gamma(\alpha - err_{t-1})$ Here, γ is the rate at which you want to learn or adjust, and err_{t-1} is an indicator that checks if your last prediction missed the mark (0 if missed the mark and 1 if not missed).The term $(\alpha - err_{t-1})$ in the formula calculates the difference between the target miscoverage level (α) and the actual occurrence of miscoverage (err$t-1$err$t-1$). This difference indicates whether the previous prediction intervals were too conservative or too narrow:
- If $err_{t-1} = 0$ (the interval included the true outcome), and α is a positive value (indicating some level of miscoverage is acceptable), then λ_t is decreased, making the future prediction intervals potentially narrower.
- If $err_{t-1} = 1$ (the interval failed to include the true outcome), suggesting that the intervals are too narrow, then λ_t is increased to widen future intervals. [**Note:** Detailed Answer to the question asked during presentation]

3. Set Up the Optimization Problem:

- **Define the Problem**: Use interval length rules ($L_{s|t}$) and prediction error distributions (F_t) to frame a problem that seeks to minimize the overall prediction error and the length of prediction intervals.

4. Apply Dynamic Programming:

- **Run the Algorithm**: Use a dynamic programming technique to find the optimal set of actions that minimize future prediction errors. This will give a series of best actions from now (t) up to $t+T$.

5. Output Prediction Intervals:

- **Determine Interval Settings**: Generate the prediction interval for the current time based on whether the updated λ_t is under the maximum threshold (λ_{max}). If λ_t goes over this limit, set your prediction interval adjustment (α_t) to zero, indicating no need for further changes.

Reproduce Experiment

To reproduce the experiment, we used the Microsoft Stock price dataset obtained using yahoo finance library in python and fitted over a ARMA-GARCH(1,1) model. Miscoverage rates (β) were calculated using z-score and standard error. The configuration file (in '.yaml' format) was updated to include three new configuration sets: ACI model, BCI model, and Fixed model for the Microsoft Dataset.

Results

ACI was used as the baseline, and BCI's step sizes were calibrated to correspond to each ACI version over the dataset produced as a result of predictions by ARMA-GARCH(1,1). The initial analysis

showed high variance in the results (Figure 1(a)). Upon investigating, it was found that the 95% confidence intervals used for generating GARCH(1,1) results, which were to be included in our ACI vs. BCI analysis, showed extremely low or zero miscoverage rates (β). The model predictions for stock price percentage changes ranged broadly from as low as 0.05% to as high as 40% within the accepted interval, rendering such wide intervals practically useless.

.

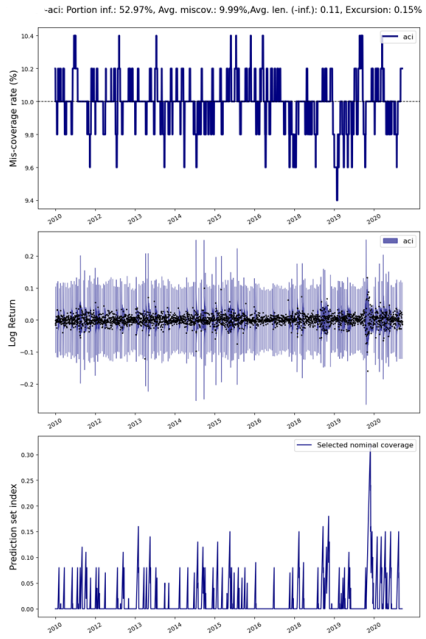

Fig 1(a):Results with 95% CI

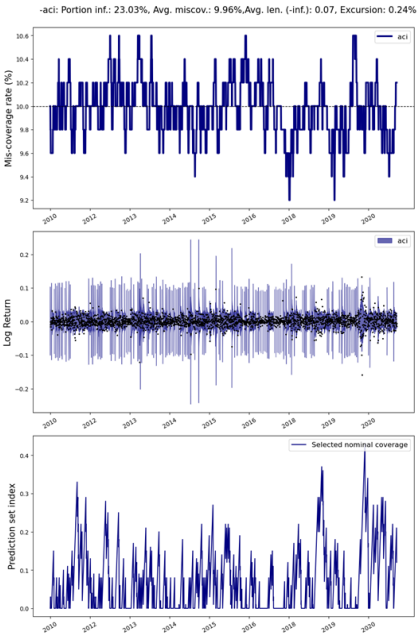

Fig 2(a): Results with 80% CI

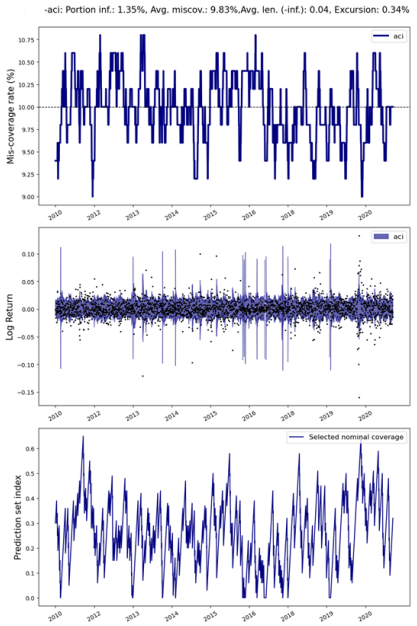

Fig3(a): Results with 50% CI

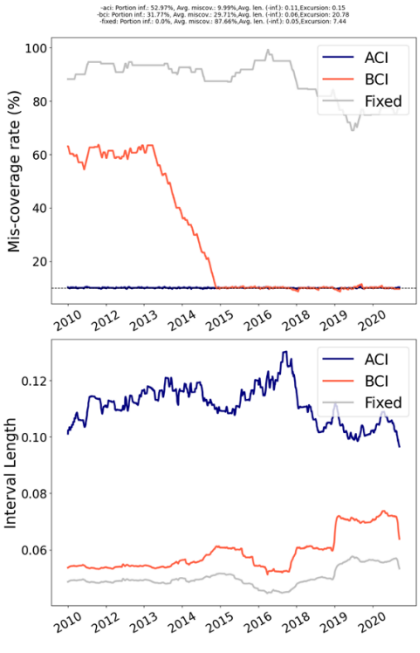

Fig 1(b):Results with 95% CI

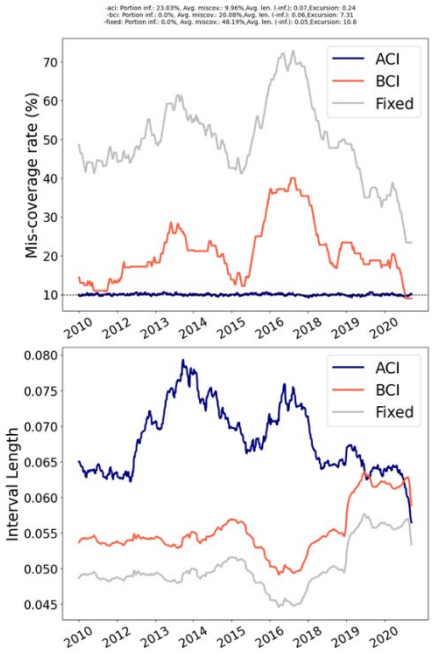

Fig 2(b): Results with 95% CI

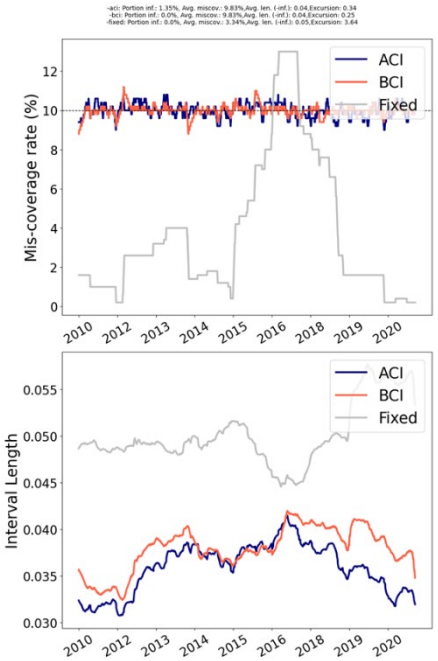

Fig3(b): Results with 95% CI

Note:

- **Portion inf.:** This refers to the portion of intervals that are infinite, the method could not determine finite bounds for the prediction interval.
- **Avg. len. (-inf.):** Average length of the intervals, excluding those that are infinite. This value indicate the average width of the prediction intervals. The lower this number, the more precise the interval prediction, provided it meets the miscoverage criteria.
- **Excursion:** Indicates the maximum deviation from the target miscoverage rate.
- **Log Returns:** The y-axis represents log-returns of a financial asset, which measure the percentage change in value, adjusted garithmically easier handling of data across a broad range of values.
- **Prediction Set Index:** This represents the nominal coverage level selected by a predictive model at each time point.

To address this, the confidence interval was first reduced to 80% (Figures 2(a) and 2(b)), and subsequently to 50% which significantly tightened the interval lengths from the GARCH(1,1) model at the 50% confidence level (Figures 3(a) and 3(b)), the results aligned closely with those documented in the paper. Further tuning might be necessary. Volatility/variance was considerably lower when predictions were made with a 50% confidence interval. At this level, ACI showed a small fraction (1.35%) of intervals with infinite length and a miscoverage rate of 9.83%, close to the target rate of 10%.

Our findings are consistent with those reported in the paper, demonstrating that both ACI and BCI effectively control coverage levels, with BCI generally providing shorter prediction intervals. However, ACI sometimes struggles with infinite-length prediction intervals, occurring when it fails to establish valid prediction bounds, often due to erratic data behavior or numerical issues. When setting tight coverage control with $\gamma_1 = 0.1$, ACI produces a moderate amount of infinite-length intervals. Conversely, with a looser control using $\gamma_2 = 0.08$, the intervals become smoother due to smaller increments in α_t's. In contrast, BCI avoids infinite intervals entirely and usually results in shorter, more informative intervals. Under looser control settings, neither ACI nor BCI produce uninformative intervals

Advantages of Bellman Conformal Inference (BCI)

- Robustness to Distribution Shifts: BCI offers a significant improvement in handling non-stationary data and distribution shifts over time. This adaptability makes it highly suitable for real-world applications where underlying data distributions are not static.
- Multi-Step Prediction Capability: Unlike traditional conformal inference methods that typically focus on immediate or one-step ahead predictions, BCI effectively integrates multi-step ahead prediction intervals. This approach allows for more comprehensive planning and decision-making based on longer-term forecasts.
- Optimization of Interval Lengths: BCI optimizes the length of prediction intervals by balancing the trade-off between interval coverage and length. This leads to more efficient use of intervals, potentially reducing the unnecessary breadth of prediction ranges and focusing on precision.
- Dynamic Adaptation:By employing principles from Model Predictive Control (MPC), BCI dynamically adjusts prediction strategies based on new data, enhancing both the adaptability and accuracy of interval forecasts.

Disadvantages of Bellman Conformal Inference (BCI)

- Computational Complexity: The integration of dynamic programming and the requirement to solve stochastic control problems inherently increase the computational demands of BCI.
- Learning Curve and Implementation Difficulty: The sophisticated nature of BCI, involving advanced concepts from statistical inference and control theory, can pose significant barriers to understanding and implementation, especially for practitioners without a strong background in these areas.
- Potential Overfitting: If not properly calibrated, the dynamic nature of BCI could lead to overfitting, especially in scenarios with high variability or noise in the data. This is particularly challenging when the data does not sufficiently support the complex model structure. This problem was faced while reproducing the results on different dataset.

Conclusion

In summary, BCI extends the capabilities of ACI by incorporating multi-step ahead intervals and enhances prediction intervals by optimizing interval lengths using Model Predictive Control. When nominal intervals are poorly calibrated, BCI outperforms ACI in reducing average interval lengths while maintaining coverage control; otherwise, BCI has comparable performance to ACI.

Literature Review

- Isaac Gibbs and Emmanuel Cand`es. Adaptive conformal inference under distribution shift. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, Advances in Neural Information Processing Systems, 2021. URL https://openreview.net/forum?id=6vaActvpcp3.
- Chen Xu and Yao Xie. Conformal prediction interval for dynamic time-series. In International Conference on Machine Learning, pages 11559–11569. PMLR, 2021.
- Sophia Sun and Rose Yu. Copula conformal prediction for multi-step time series forecasting, 2023.
- Bellman Conformal Inference:Calibrating Prediction Intervals For Time Series Zitong Yang ⋄ Emmanuel Cand`es Lihua Lei