

# Forecast Reconciliation for Quantiles using Bilevel Optimisation

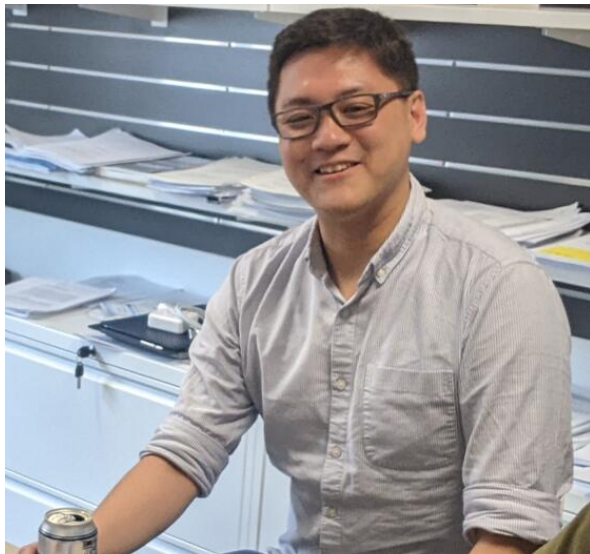
H. Alipour, G. Athanasopoulos, N. Ho-Nguyen and **A. Panagiotelis**

September 8, 2023

## Joint work with. . .



... and



# Probabilistic Forecast Reconciliation

# A refresher

- ▶ Consider a probabilistic forecast  $\hat{F}_{t+h|t}$  for a random  $n$ -vector  $\hat{\mathbf{y}}_{t+h}$  made using all information up to time  $t$ .
- ▶ Draw a sample of size  $J$  from this probabilistic forecasts

$$\hat{\mathbf{y}}_{t+h|t}^{(1)}, \hat{\mathbf{y}}_{t+h|t}^{(2)}, \dots, \hat{\mathbf{y}}_{t+h|t}^{(J)} \sim \hat{F}_{t+h|t}$$

- ▶ Problem: these draws do not respect (aggregation) constraints.

# Reconciliation

- ▶ Find any mapping  $\psi : \mathbb{R}^n \rightarrow \mathfrak{s}$  where  $\mathfrak{s}$  is the linear subspace where constraints hold.
- ▶ Usually (including for this talk), this is a linear mapping  $\mathbf{S}\mathbf{G}$ , where the columns of  $\mathbf{S}$  span the coherent subspace.
- ▶ Letting  $\tilde{\mathbf{y}}_{t+h|t}^j = \psi(\hat{\mathbf{y}}_{t+h|t}^j)$

$$\tilde{\mathbf{y}}_{t+h|t}^{(1)}, \tilde{\mathbf{y}}_{t+h|t}^{(2)}, \dots, \tilde{\mathbf{y}}_{t+h|t}^{(J)} \sim \tilde{F}_{t+h|t}$$

- ▶ Where  $\tilde{F}_{t+h|t}$  is the **reconciled** probabilistic forecast.

## How to choose $\psi/G$

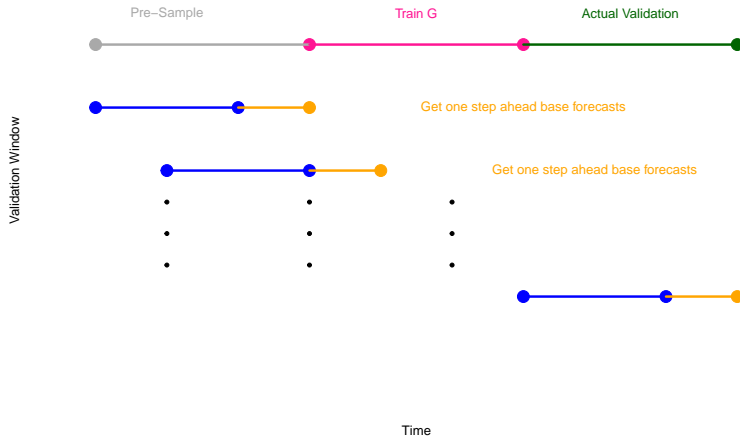
- ▶ There are popular choices from the point forecast reconciliation literature.
- ▶ Apart from the Gaussian case, where MinT is optimal w.r.t the logarithmic score (Wickramasuriya, 2023) little is known theoretically.
- ▶ Instead we can optimise  $\mathbf{G}$  with respect to some loss function.
- ▶ This is what we did in Panagiotelis et al (2023), others have done similar.

# Data Split

- ▶ We require both  $\hat{\mathbf{y}}_{t+h|t}^{(j)}$  and  $\mathbf{y}$  to train  $\mathbf{G}$ .
- ▶ Data is split into three
  - ▶ Pre-sample (used to get  $\hat{\mathbf{y}}$  but not train  $\mathbf{G}$ )
  - ▶ Training (used to train  $\mathbf{G}$ )
  - ▶ Validation (once  $\mathbf{G}$  is trained, does it give good forecasts?)
- ▶ There are variations, what we do so far described on next slide



# In a picture



# Optimisation

- In general we want to find

$$\underset{\mathbf{G}}{\operatorname{argmin}} \sum_{t \in \mathcal{T}_{\text{train}}} L(\hat{\mathbf{y}}_t, \tilde{F}_{t|t-1})$$

- Here,  $L(., .)$  is a loss function such as a scoring rule.
- Note  $\tilde{F}_{t|t-1}$  depends on  $\mathbf{G}$  through  $\tilde{\mathbf{y}}_{t|t-1}^{(j)}$ .
- Could generalise to h-step ahead forecasts

# Quantiles

# Quantile reconciliation

- ▶ Quantiles do not need to be coherent!
- ▶ If you disagree
  - a) you are wrong, and
  - b) avoid Stephan Kolassa!
- ▶ However that does not mean that we cannot borrow information from different series in a hierarchy to improve quantile forecasts.
- ▶ That is the motivation for this project.

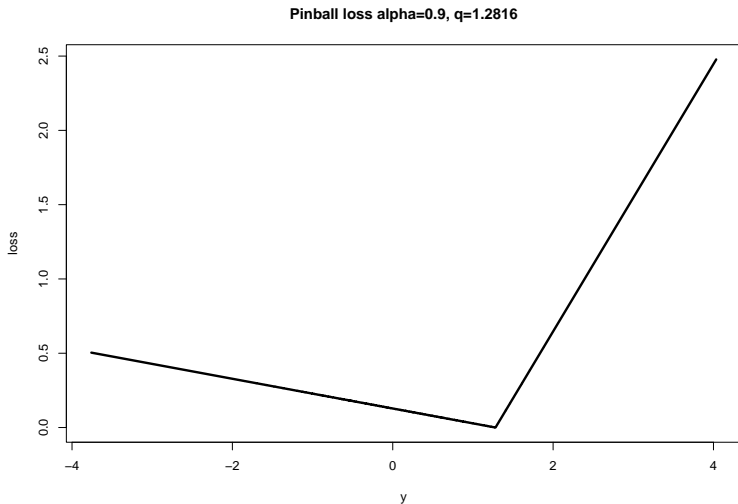
# Pinball loss

For the rest of the talk we will consider the **pinball loss** function

$$L_{\alpha}(y, q) = \alpha(y_i - q)I(y_i \geq q) + (1 - \alpha)(q - y_i)I(y_i < q)$$

- ▶ The level of quantile is  $\alpha$  and  $I(\cdot)$  equals 1 when the statement in parentheses is true, 0 otherwise.
- ▶ This is an *consistent* loss function for the  $\alpha$  quantile.

# In a picture



# Simple setting

- ▶ Sample quantiles can be recovered using the pinball loss function

$$q^* = \underset{q}{\operatorname{argmin}} \sum_i L_\alpha(y_i, q)$$

- ▶ Note that these might not exactly match the quantiles that come out of your favorite statistical package, there are alternative definitions of quantiles.

## In reconciliation

- ▶ Reconciliation is a multivariate problem.
- ▶ If we are targeting quantiles we can optimise.

$$\underset{G}{\operatorname{argmin}} \sum_{i=1}^n \sum_{t \in \mathcal{T}_{\text{train}}} L_{\alpha}(y_{i,t}, \tilde{q}_{i,t|t-1})$$

- ▶ However for each variable we want  $\tilde{q}_{i,t|t-1}$  to be the  $\alpha$ -quantile of  $\tilde{y}_{i,t|t-1}^{(1)}, \dots, \tilde{y}_{i,t|t-1}^{(J)}$  meaning it must satisfy

$$\tilde{q}_{i,t|t-1} = \underset{q}{\operatorname{argmin}} \sum_j L_{\alpha}(\tilde{y}_{i,t|t-1}^{(j)}, q)$$



# Optimisation

- ▶ This is an example of **bi-level optimisation**.
- ▶ It is further complicated by the fact that pinball loss is not smooth.
- ▶ This is when you phone a friend who knows about optimisation. . .

# Optimisation

# Mixed integer programming

- ▶ Pinball loss problems can be recast in terms of Mixed Integer Programming (MIP).
- ▶ The full description has 17 equations...
- ▶ Some intuition on the next slide
- ▶ Can be solved with purpose built solvers for these types of problem (e.g. Gurobi)

# Pinball loss as MIP

$$\begin{aligned}
 \tilde{y}_j - \tilde{q} &\leq M v_j^-, & j \in [J] \\
 \tilde{q} - \tilde{y}_j &\leq M v_j^+, & j \in [J] \\
 0 \leq w_j &\leq 1, & j \in [J] \\
 w_j &\leq 1 - v_j^-, & j \in [J] \\
 1 - w_j &\leq 1 - v_j^+, & j \in [J] \\
 \frac{1}{J} \sum_{j \in [J]} w_j &= 1 - \alpha \\
 v_j^-, v_j^+ &\in \{0, 1\}, & j \in [J].
 \end{aligned}$$

## Unfortunately. . .

- ▶ Even with modern solvers, this is too slow.
- ▶ Even for small problems (3-variable hierarchies,  $T = 50$ ,  $J = 100$ ) it takes a few hours to run.
- ▶ It is not feasible for bigger hierarchies.
- ▶ So we looked for an alternative.

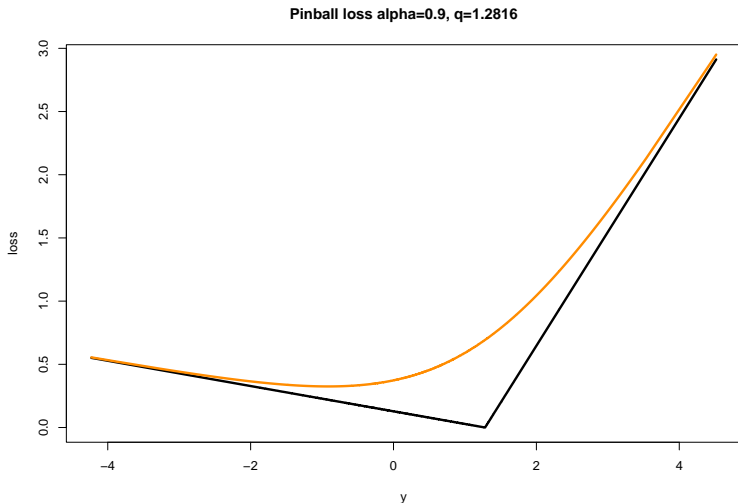
# Smooth pinball loss

The following function approximates the pinball loss and converges to pinball loss as  $\beta \rightarrow \infty$

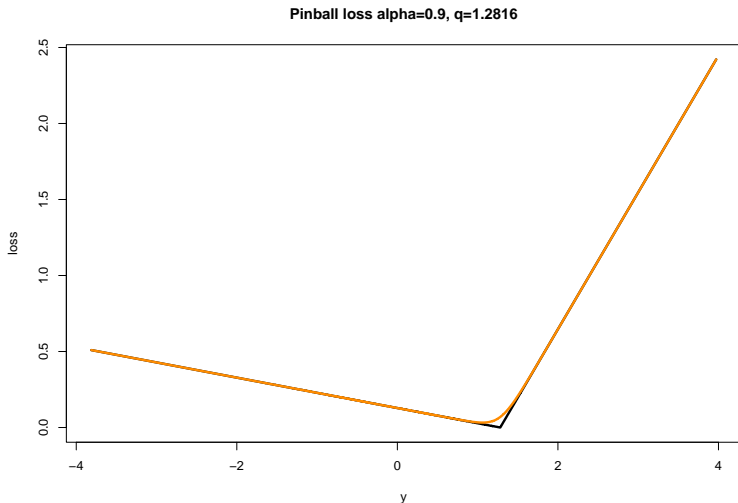
$$L_{\alpha}^{\beta}(y, q) = \frac{1}{\beta} \log \left( e^{\beta \alpha (y - q)} + e^{\beta (1 - \alpha) (q - y)} \right)$$

Unlike the pinball function it is smooth.

# Smoothed pinball loss ( $\beta = 1$ )

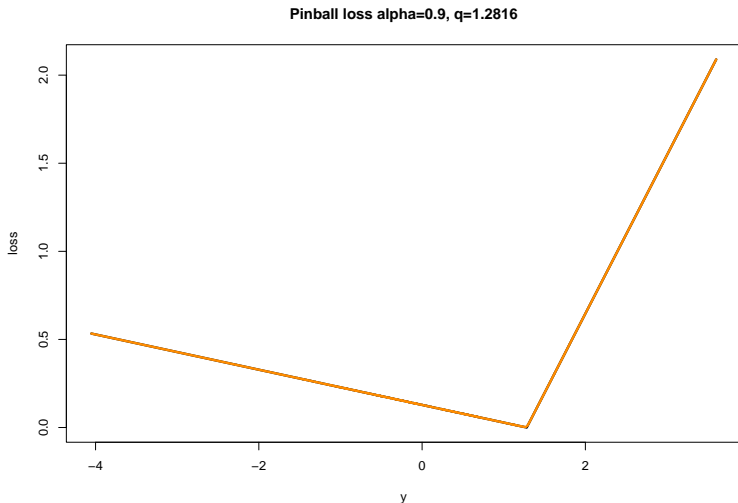


# Smoothed pinball loss ( $\beta = 10$ )





# Smoothed pinball loss ( $\beta = 100$ )



# Optimisation problem

Recall the problem is

$$\underset{G}{\operatorname{argmin}} \sum_{i=1}^n \sum_{t \in \mathcal{T}_{\text{train}}} L_{\alpha}^{\beta}(y_{i,t}, \tilde{q}_{i,t|t-1})$$

where

$$\tilde{q}_{i,t|t-1} = \underset{q}{\operatorname{argmin}} \sum_j L_{\alpha}^{\beta}(\tilde{y}_{i,t|t-1}^{(j)}, q)$$

and

$$\tilde{y}_{i,t|t-1}^{(j)} = \mathbf{SG} \hat{y}_{i,t|t-1}^{(j)}$$

# A trick

To use gradient descent we need

$$\frac{\partial L_{\alpha}^{\beta}(y_{i,t}, \tilde{q}_{i,t|t-1})}{\partial \mathbf{g}_{l,k}}$$

This can be found using the chain rule, however the gradient must *pass through* the argmin in the lower level. A lemma by Gould et. al. (2016) can be used for this.

# Empirical study

- ▶ Use Australian tourism data.
- ▶ Grouped hierarchy of states and purpose of travel.
- ▶ Dimension of **S** is  $40 \times 28$ .
- ▶ Seasonal ARIMA used for base forecasts.

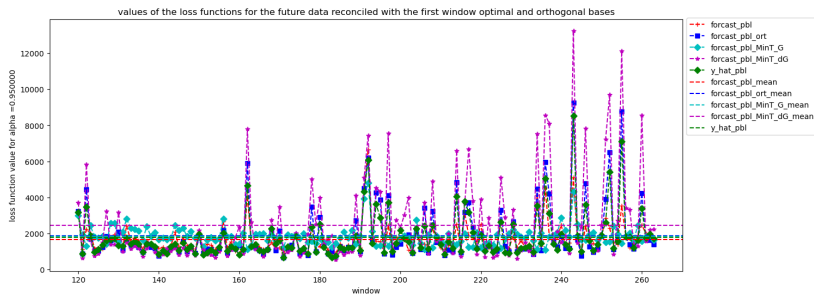
## Results - In Sample

Method	Pinball Loss
Base	1768
OLS	1876
WLS	2440
MinT	1845
QOpt	1673

## Results - Out of Sample

Method	Pinball Loss
Base	92278
OLS	96693
WLS	109926
MinT	112954
QOpt	78776

# Results



## Work to do

- ▶ Determine whether these differences are significant
  - ▶ Use VaR backtesting literature
- ▶ Effect of the quantile
  - ▶ Conjecture is that results will differ for more extreme quantiles
- ▶ Consider regularisation
  - ▶ Shrink **SG** to an orthogonal projection or MinT



# Thanks!

Questions. . .