MONASH University

Forecast reconciliation: Geometry, optimization and beyond

Anastasios Panagiotelis

2 July 2025

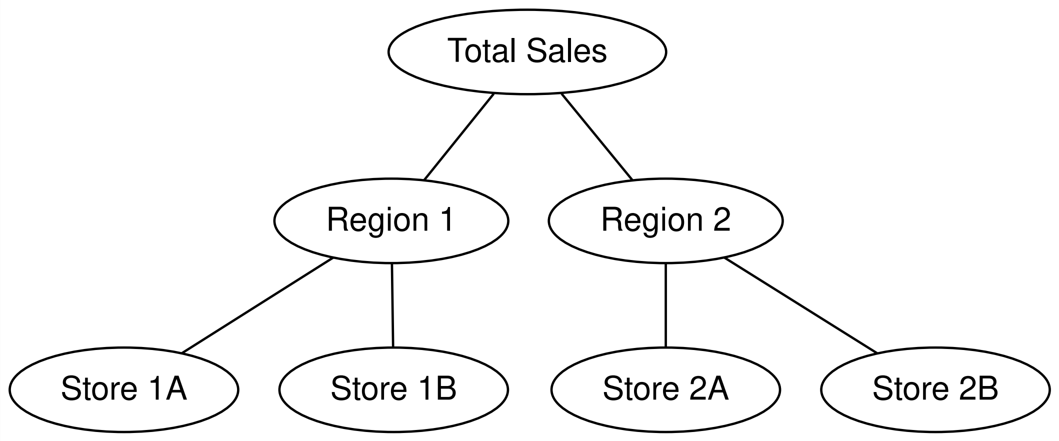
Outline

Outline

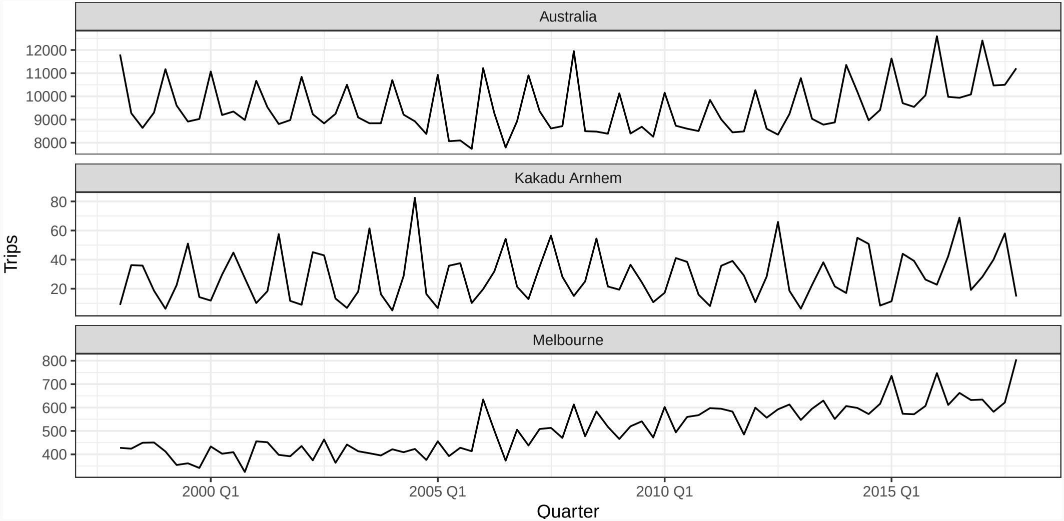
Hierarchical Time Series

- At its most general, **multivariate** data $\mathbf{y} \in \mathbb{R}^n$ bound together by some constraints.
- Typically these constraints are **linear**, although later I will present new work for non-linear constraints.
- Most commonly arise due to an **aggregation** structure, hence the name 'hierarchical'.
- Need not be hierarchical, alternative structures are grouped (or crossed) aggregation, or temporal aggregation.

Hierarchy


Total Sales
Region 1
Region 2
Store 1A
Store 1B
Store 2A
Store 2B

Data



Australia
12000
11000
10000
9000
8000
Kakadu Arnhem
80
60
40
20
Melbourne
800
700
600
500
400
Trips
2000 Q1
2005 Q1
2010 Q1
2015 Q1
Quarter

Context

- Observations will always **cohere** to the constraints.
- Forecasts generally will not.
  - Different forecasts are made by different agents.
  - Hard to construct a method that guarantees coherence.
- Note there is work on **end-to-end** forecasting (e.g. Rangapuram et al. 2021).
- This talk is about **two-stage** processes.

Forecasting reconciliation

- Start with a vector of incoherent forecasts $\hat{\mathbf{y}}$.
- Choose m variables $\mathbf{b}$ (usually **bottom level**).
- Construct an $n \times m$ matrix $\mathbf{S}$ such that $\mathbf{y} = \mathbf{S}\mathbf{b}$.
- 'Regress' $\mathbf{y}$ on $\mathbf{S}$.
- Use prediction as reconciled forecasts $\tilde{y}$, i.e.

$$\tilde{\mathbf{y}} = \mathbf{S}(\mathbf{S}\mathbf{S})^{-1}\mathbf{S}'\mathbf{y}$$

- This is called OLS reconciliation.

Example of S

- In the simple hierarchy shown earlier

$$
\begin{pmatrix} y_{\mathrm{Tot}} \\ y_1 \\ y_2 \\ y_{1A} \\ y_{1B} \\ y_{2A} \\ y_{2B} \end{pmatrix}
=
\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
\times
\begin{pmatrix} y_{1A} \\ y_{1B} \\ y_{2A} \\ y_{2B} \end{pmatrix}
$$

$$
\mathbf{y} = \mathbf{S} \times \mathbf{b}
$$

An optimization lens

- Ensure that $\hat{\mathbf{y}}$ and $\tilde{\mathbf{y}}$ are ‘close’.

$$\tilde{\mathbf{y}} = \underset{\mathbf{y}}{argmin} ||\mathbf{y} - \hat{\mathbf{y}}||_2$$

- Under certain assumptions, this yields the same solution as the regression interpretation.
- Also has a game theoretic interpretation (see van Erven and Cugliari 2015)

Generalizations

- Where OLS works, it makes sense to consider GLS

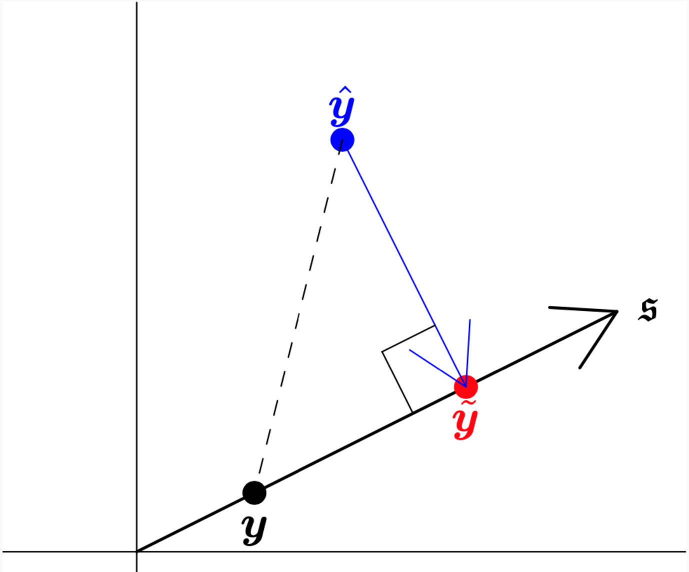
$$\tilde{\mathbf{y}} = \mathbf{S}(\mathbf{S W S})^{-1}\mathbf{S}'\mathbf{W y}$$

- Setting $\mathbf{W}^{-1}$ to the covariance matrix of $\mathbf{y} - \hat{\mathbf{y}}$ optimizes expected squared error loss.
- This is the well-known **MinT method** of Wickramasuriya, Athanasopoulos, and Hyndman (2019)

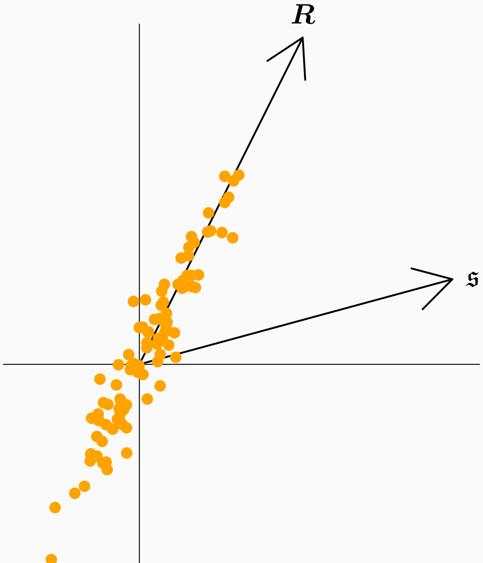
A geometric view

- Another alternative is a geometric view.
- The constraints imply a linear subspace $\mathcal{S} \subset \mathbb{R}^n$.
  - Sometimes $\mathfrak{s}$ notation is used.
- Reconciliation is a **map** from $\mathbb{R}^n \to \mathcal{S}$.
- More intutive understanding of why reconciliation ‘works’.
- Provides a framework to extend reconciliation to other settings.

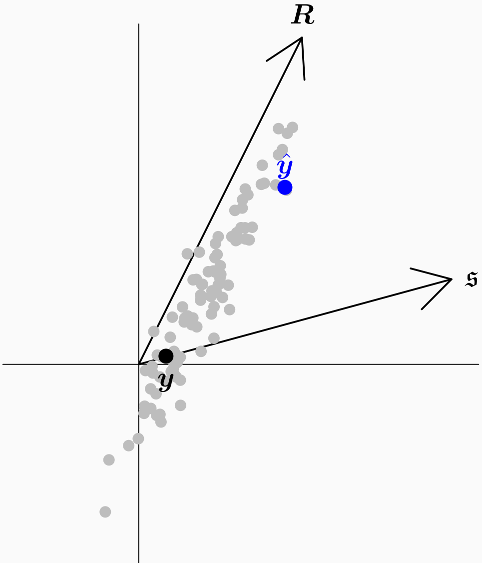
In a picture



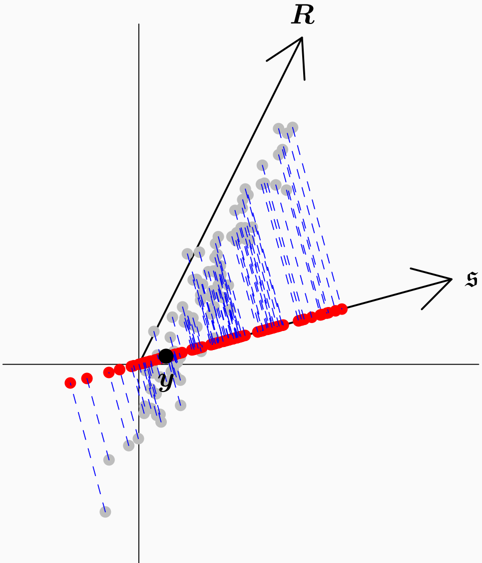
Why MinT?



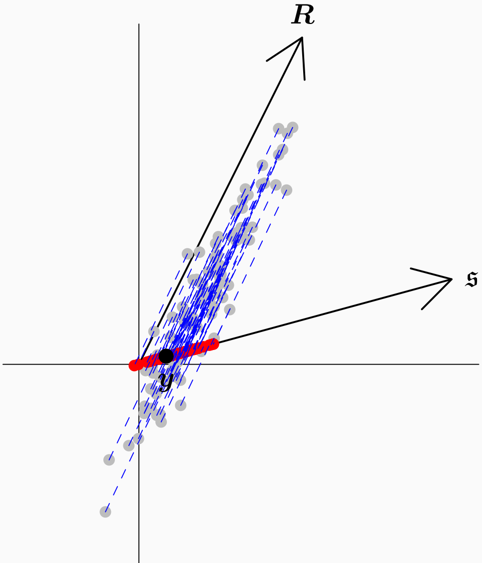
Why MinT?



Why MinT?



Why MinT?

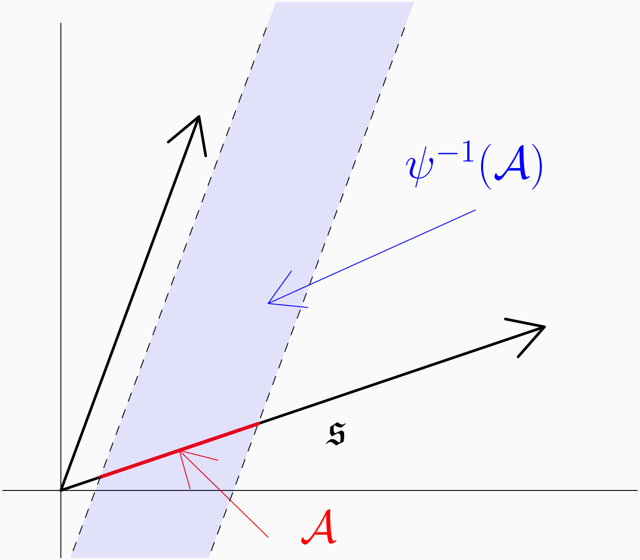


Outline

Problem

- Probabilistic forecasts are important!
- The regression interpretation does not naturally lend itself to be extended to probabilistic forecasting
- Some notions of reconciling draws from probabilistic distributions (Jeon, Panagiotelis, and Petropoulos 2019)
- Formalized as a **pushforward** (Panagiotelis et al. 2023).
- Alternative approaches define the reconciled distribution using *copulas* (Ben Taieb, Taylor, and Hyndman 2021) or by *conditioning* (see Corani et al. 2021).

Probabilistic Reconciliation



Probabilistic Reconciliation

- Let $\hat{\mu}$ be a measure on the usual σ-algebra defined on $\mathbb{R}^n$.
- Let $\mathcal{A}$ be some region entirely within $\mathcal{S}$ and $\psi : \mathbb{R}^n \rightarrow \mathcal{S}$.
- Let $\mathcal{B}$ be the pre-image of $\mathcal{A}, i.e. \forall \mathbf{x} \in \mathcal{B}, \psi(\mathbf{x}) \in \mathcal{A}$.
  - Pre-image denoted $\psi^{-1}(\mathcal{A})$
- The reconciled measure $\tilde{\mu}$ is defined as

$$\tilde{\mu}(\mathcal{A}) = \hat{\mu}(\mathcal{B})$$

- $\tilde{\mu}$ is the **pushforward** of $\hat{\mu}$ by ψ, denoted as $\psi \# \hat{\mu}$

Optimality

- This merely defines a way of getting a reconciled distribution from some incoherent base distribution.
- What is the optimal ψ?
- In the point forecasting world the ψ given by MinT is optimal for squared loss.
- What does optimality even mean for distributional forecasts?

Scoring rules

- A scoring rule $S : \mathcal{P} \times \mathbb{R}^n \to \mathbb{R}$ takes a distributional forecast from a family $\mathcal{P}$ and an observation and assigns a *score* that measures forecast quality.
- In the context of probabilistic forecasting, it has been proven that choosing ψ to be the same projection as MinT is optimal for log score when probabilistic forecasts are Gaussian (Wickramasuriya 2023)
- In other cases we can optimize using a data driven approach.

Score Optimization

- Obtain a sequence of base forecasts $\hat{\mu}_t$ and corresponding realizations $\mathbf{y}_t$.
- Parameterize ψ by some parameters $\boldsymbol{\theta}$ (denoted $\psi_{\boldsymbol{\theta}}$)
- For example, $\psi(\mathbf{y}) = \mathbf{S}(\mathbf{d} + \mathbf{G}\mathbf{y})$, $\boldsymbol{\theta} = \left(\mathbf{d}', vec(\mathbf{G})'\right)'$
- Optimize the following

$$\underset{\boldsymbol{\theta}}{argmin} \sum_t S(\psi_{\boldsymbol{\theta}} \# \hat{\mu}_t, \mathbf{y}_t)$$

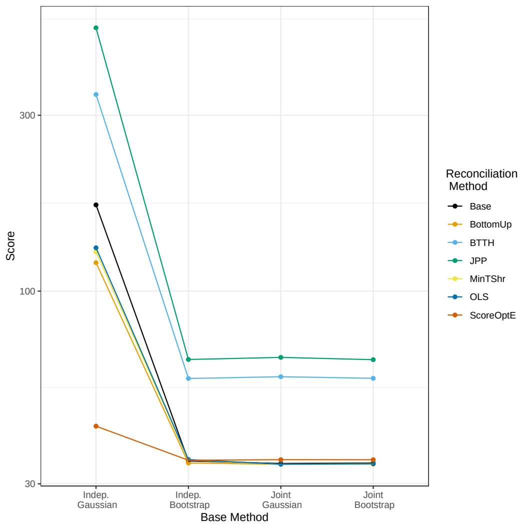
Some practical points

- Scoring rules that have been used include log score, energy score and variogram score.
- Paired forecasts and observations can be obtained using rolling or expanding window schemes.
- Different optimal values can be obtained for different forecast horizons.
- Often draw a sample from $\hat{\mu}$ rather than work with the distribution itself.
- Optimization by first order methods (e.g. SGD).

Energy Generation Example

- Consider a moderate sized hierarchy (approx 20 variables) of electricity generation from different sources.
- Consider four different base forecasts
  - Assume Gaussianity or bootstrap
  - Assume independence or dependence
- Reconcile using projections (OLS, MinT) and also by optimising Energy score.

Energy Generation Example



Some thoughts

- Interplay between reconciliation and model misspecification.
- Reconciliation via score optimization most effective when base models heavily misspecified.
- For reasonably well specified models
  - Projections are robust
  - Gains over base forecasts are not as big
- Generalization of point forecast reconciliation to probabilistic setting gives forecaster a ‘second chance’.

Outline

Pinball loss

- Many forecasting problems involve optimizing pinball loss.

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

- Here, I(.) equals 1 when the statement in parentheses is true, 0 otherwise.
- Quantiles minimize expected pinball loss $E_Y\left[L_\alpha(y, q)\right]$

In reconciliation

- To target quantiles we optimize.

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

- Subject to the constraints

$$\tilde{q}_{i,t} = \underset{q}{argmin} E_{\tilde{Y}_{i,t}} \left[L_\alpha(\tilde{y}_{i,t}, q) \right]$$

Optimization

- This is an example of **bi-level optimization**.
- It is further complicated by the fact that pinball loss is not smooth.
- It is also complicated by the need to approximate expectations with sample equivalents

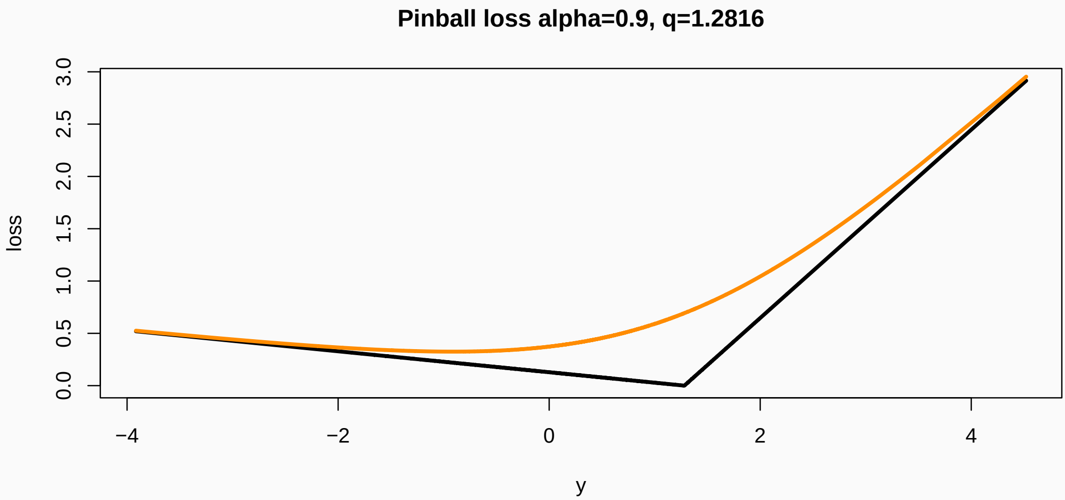
Smooth pinball loss

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

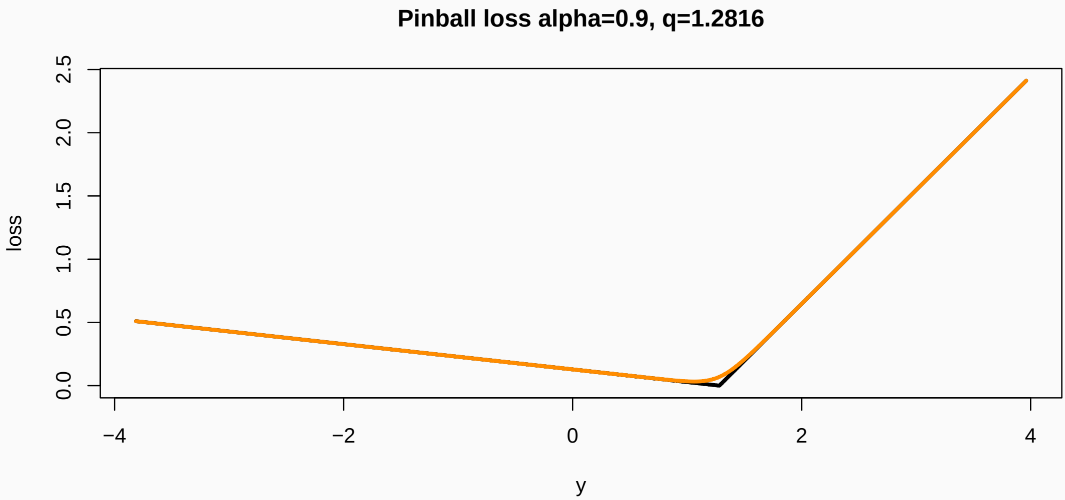
$$L_\alpha^\beta(y, q) = \frac{1}{\beta} \log\left(e^{\beta\alpha(y-q)} + e^{\beta(1-\alpha)(q-y)}\right)$$

Unlike the pinball function it is smooth, meaning we can use first order methods (like Stochastic Gradient Descent).

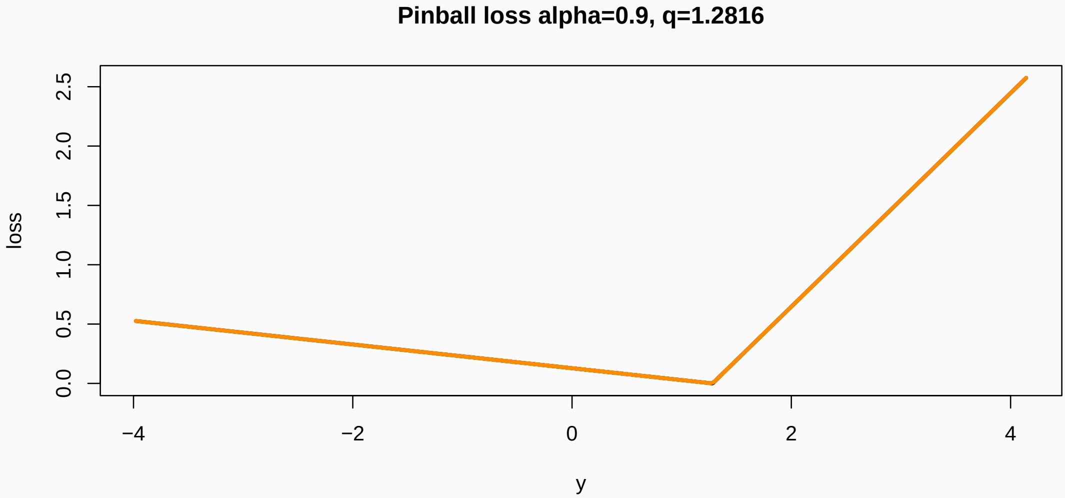
Smoothed pinball loss (β = 1)



Smoothed pinball loss (β = 10)



Smoothed pinball loss ($\beta = 100$)



Optimization problem

The problem we can actually solve is

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

subject to

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

where $\tilde{y}_{i,t}^{(j)} = \mathbf{SG}\hat{y}_{i,t}^{(j)}$ and $\hat{y}_{i,t}^{(j)} \sim \hat{\mu}_t$ for $j = 1, \ldots, J$

What has been proven

Let $f(\mathbf{G})$ be the problem we want to solve, $f^{\beta}(\mathbf{G})$ be the smooth approximation of pinball loss, and $f^{(J)}$ the approximation from using J draws. We prove

$$\sup_{\mathbf{G}\in\mathcal{G}} \left| f^{\beta}(\mathbf{G}) - f(\mathbf{G}) \right| \to 0 \text{ as } \beta \to \infty$$
$$\sup_{\mathbf{G}\in\mathcal{G}} \left| f^{(J)}(\mathbf{G}) - f^{\beta}(\mathbf{G}) \right| \to 0 \text{ as } J \to \infty.$$

This implies the minimizer of the approximate problem converges to the minimizer of the 'true' problem.

Convergence of SGD

- Optimization via SGD, taking care to pass gradient through argmin in lower level.
- Note that the approximation of the expectation in the constraint means that the gradient is a biased estimate
- However the variant of SGD we use will converge if
  - Bounded second moment of the stochastic gradient
  - L-smoothness
- Both are proven to hold for the functions we consider.
- Important to check convergence of SGD.

Empirical study

- Use Australian tourism data.
- Grouped hierarchy of states and purpose of travel.
- Dimension of $\mathbf{S}$ is 40×28.
- Seasonal ARIMA used for base forecasts. Distributional forecasts assume Gaussian errors and skew t errors.
- Train on 10 years (120 observations), evaluation on 7 years (84 observations).

Pinball Loss - Out of Sample (Normal errors)

	Quantile Level			
Method	0.05	0.2	0.8	0.95
Base	32*	85*	101	46
OLS	32*	84*	104	51
WLS	**31***	**82***	112	65
MinT	**31***	**82***	111	65
QOpt	35*	85*	100*	**41***

Bold denotes best performing method, asterisk(*) denotes inclusion in model confidence set (Hansen et. al., 2011).

Outline

The problem

- What if the constraints are non-linear?
- For example ratios are common quntities of interest.
  - Mortality rates are Deaths divided by Exposure.
  - Unemployment rates are number of unemployed divided by labor force.
- Both of these examples are also be subject to aggregation.

Problem formulation

- In general there are C constraints $g_1(\mathbf{y}) = 0, \ldots g_C(\mathbf{y}) = 0$, or more compactly $g(\mathbf{y}) = \mathbf{0}$.
- The level set of points $\mathbf{y} : g(\mathbf{y}) = \mathbf{0}$ defines a coherent *surface* or *manifold* continue to be denoted as $\mathcal{S}$.
- Non-linear reconciliation solves the following problem:

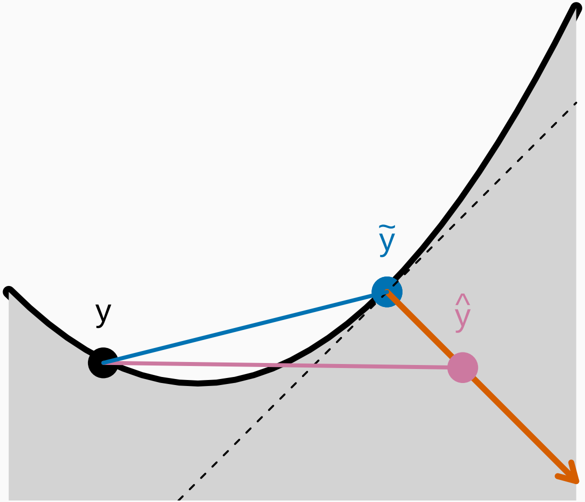
$$\tilde{\mathbf{y}} = \underset{\mathbf{y}}{argmin}(\mathbf{y} - \hat{\mathbf{y}})'\mathbf{W}(\mathbf{y} - \hat{\mathbf{y}})$$

- Subject to $\mathbf{y} \in \mathcal{S}$.

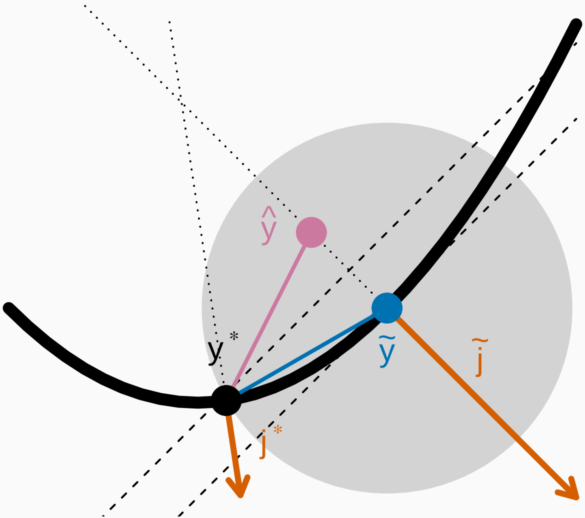
Towards Theory

- Note focus is still on point forecasts.
- First consider case of convex constraints.
  - Reconciliation guaranteed to improve base forecast, but only in hypograph.
- For more general constraints
  - Find closest point on the coherent manifold equidistant from the base and reconciled forecast.
  - This defines a ball in which reconciliation always outperforms base forecasts.

Convex Function: Hypograph



Any function



Radius of the Ball

- The radius of the ball on the previous slide is given by

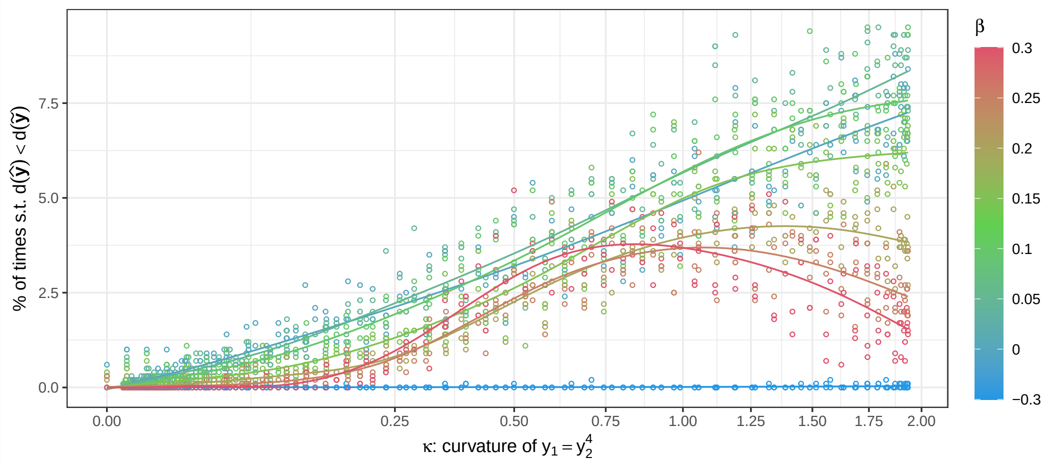
$$r = \sqrt{\boldsymbol{\kappa}'\mathbf{J}^{*\prime}\mathbf{J}^{*}\boldsymbol{\kappa} + \mu\boldsymbol{\kappa}'\mathbf{J}^{*\prime}\tilde{\mathbf{J}}\boldsymbol{\lambda} + \frac{\mu^2}{4}\boldsymbol{\lambda}\tilde{\mathbf{J}}'\tilde{\mathbf{J}}\boldsymbol{\lambda}}$$

- $\mathbf{J}^*$ and $\tilde{\mathbf{J}}$ are gradients of the constraint evaluated at $\mathbf{y}^*$ and $\tilde{\mathbf{y}}$ respectively.
- λ and $\boldsymbol{\kappa}$ are Lagrange multipliers associated with certain optimization problems.

How is this useful?

- This theory tells us that non-linear forecast reconciliation is more likely to succeed when
  - Base forecast is far from coherent manifold.
  - The constraint function has lower curvature.
  - When reconciled forecast is in a high probability region of the true DGP
  - When some constraints are convex and the bast forecast is more likely to lie in the hypographs of these constraints.

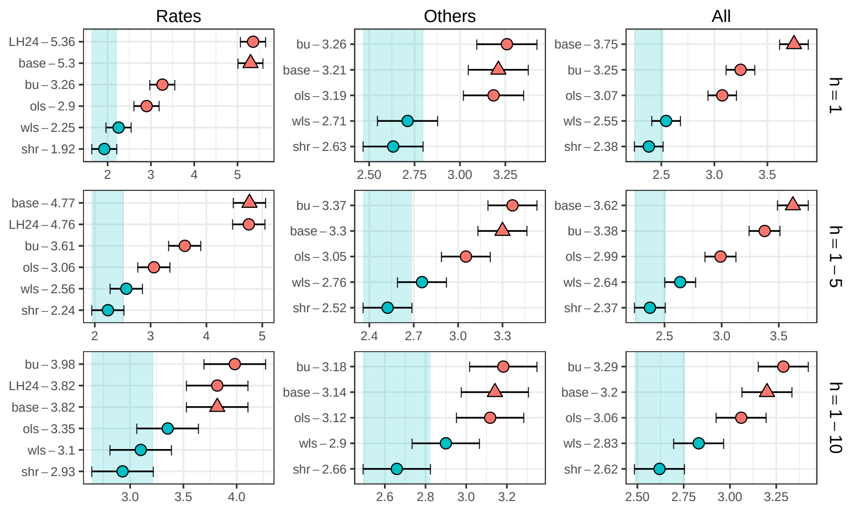
Simulation results



Mortality Data

- Annual (1969-2019) data on
  - Exposure (E)
  - Deaths (D)
  - Mortality rates (M)
- For US as a whole and 9 census regions.
- E and D respect aggregation constraints, M do not but $M = D/E$ for each region.

Mortality Data



Outline

Setup

- Suppose we are interested in multivariate forecasting but do not have linear (or non-linear) constraints.
- Is there anything interesting about forecast reconciliation.
- Surprisingly... Yes!
- New work on Forecast Linear Augmented Projects (FLAP)

What is FLAP?

- Suppose the target is to foreast $\mathbf{y}_t \in \mathbb{R}^m$.
- We construct new synthetic series $\mathbf{c}_t \in \mathbb{R}^p$ where $\mathbf{c}_t = \mathbf{\Phi}\mathbf{y}_t$.
  - The choice of $\mathbf{\Phi}$ is arbitrary.
- The augmented vector $(\mathbf{c}_t', \mathbf{y}_t')'$ coheres to known linear constraints.
- The strategy is to carry out forecast reconciliation on the augmented vector.

What is FLAP?

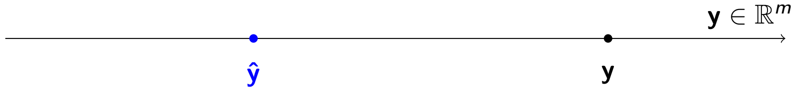
- Forecast all components of $(\mathbf{c}'_t, \mathbf{y}'_t)'$
- Typically $\hat{\mathbf{c}} \neq \mathbf{\Phi}\hat{\mathbf{y}}_\mathbf{t}$.
- Use MinT to obtain $(\tilde{\mathbf{c}}'_t, \tilde{\mathbf{y}}'_t)'$ such that $\tilde{\mathbf{c}} = \mathbf{\Phi}\tilde{\mathbf{y}}_\mathbf{t}$.
- Total forecast error variance of $\tilde{\mathbf{y}}_t$ provably lower than $\hat{\mathbf{y}}_t$.
- Total forecast error variance is non-increasing as more components added.

No free lunch

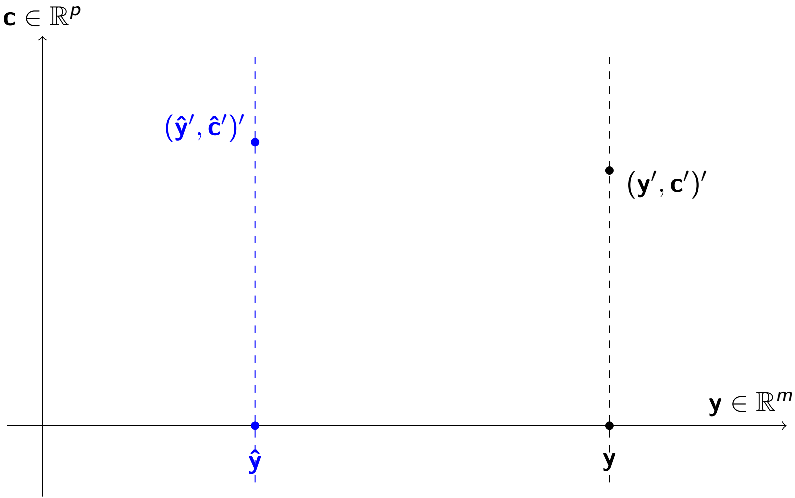
- Originally FLAP stood for ‘Free Lunch’ augmented projection.
- All proofs assume error covariance matrix used in MinT in **known**. In practice it is estimated.
- The quality covariance matrix estimates deteriorate with higher dimension.
- However for finite dimension, the benefit of FLAP outweighs errors in estimating covariance matrix.

Geometry of FLAP

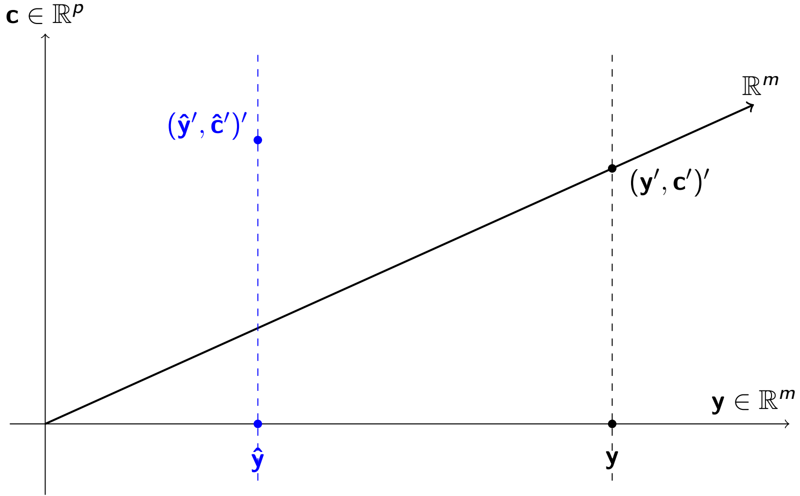
Geometry of FLAP



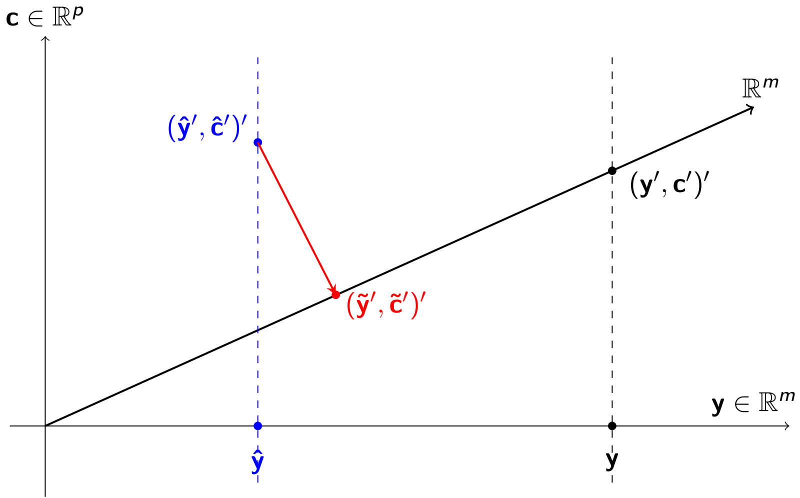
Geometry of FLAP



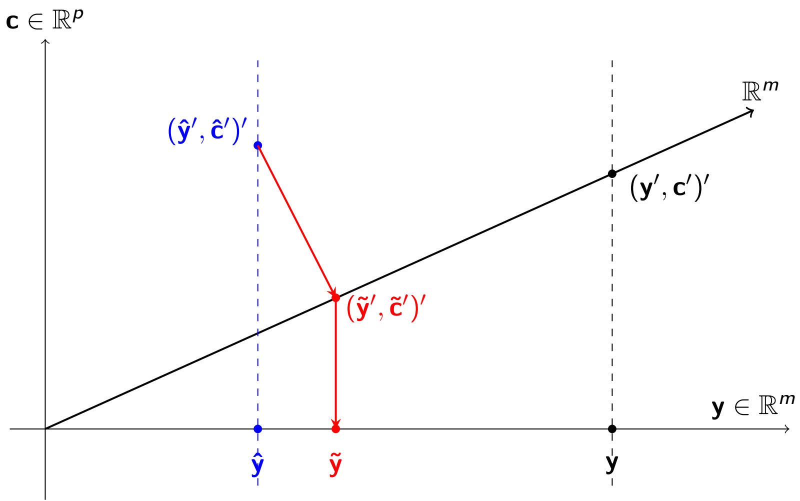
Geometry of FLAP



Geometry of FLAP



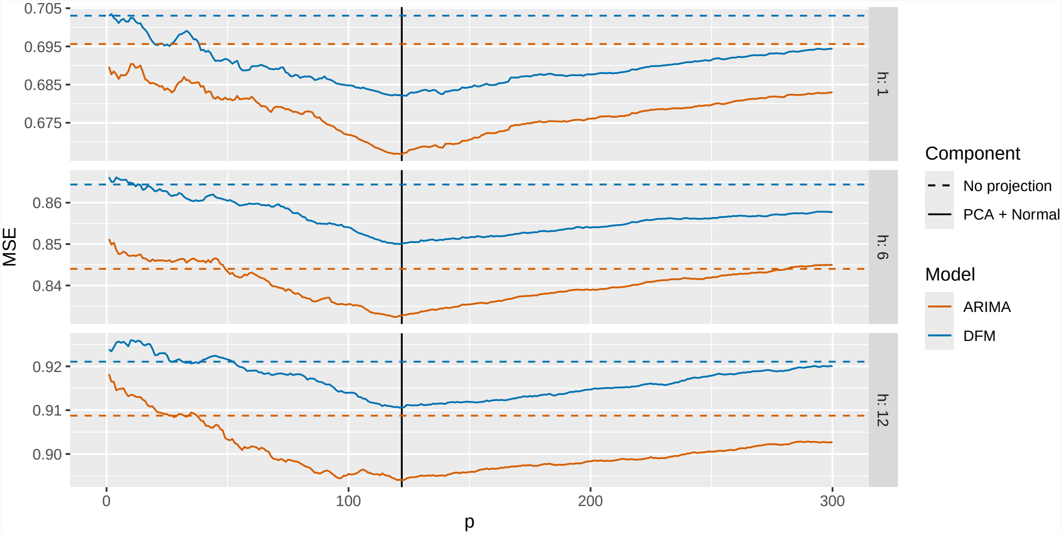
Geometry of FLAP



FRED-MD

- Monthly data of macroeconomic variables (McCracken and Ng, 2016).
- Data from Jan 1959 – Sep 2023. 777 observations on 122 series.
- Same cleaning process as per McCracken and Ng (2016).
- All series scaled to have mean 0 and variance 1.
- Expanding time series cross-validation with initial size of 25 years and forecast horizon 12 months.

FRED-MD



0.705
0.695
0.685
0.675
0.86
0.85
0.84
0.92
0.91
0.90
MSE
0
100
200
300
p
h: 1
h: 6
h: 12
Component
No projection
PCA + Normal
Model
ARIMA
DFM

Working Paper and R Package

YF Yang, G Athanasopoulos, RJ Hyndman, and A Panagiotelis (2024). "Forecast Linear Augmented Projection (FLAP): A free lunch to reduce forecast error variance".

Department of Econometrics and Business Statistics, Monash University, Working Paper Series 13/24.

You can install the stable version from CRAN

```r
## CRAN.R-project.org/package=flap
install.packages("flap")
```

or the development version from Github

```r
## github.com/FinYang/flap
# install.packages("remotes")
remotes::install_github("FinYang/flap")
```

Outline

Final thoughts

- Forecast reconciliation is a practical and interesting problem with many open questions.
  - Jump on the bandwagon!
- Sometimes understanding the same problem in a different way opens new doors in research.
- Theory, methodology and application all matter. The connections and feedback loops between them are important.
  - Do not neglect any of these!

Final Thankyou









Link to these slides