Online Prediction Under Model Uncertainty Via Dynamic Model Averaging (DMA): Application to a Cold Rolling Mill

Adrian E. Raftery

University of Washington
www.stat.washington.edu/raftery

Joint work with Miroslav Kárný, Josef Andrýsek (ÚTIA, Czech Academy of Sciences, Prague), and Pavel Ettler (COMPUREG, Plzeň, ČR)

Workshop on Bayesian Model Selection
University of Florida
January 11–12, 2008

Outline

Outline

- Rolling mill prediction problem and data

Outline

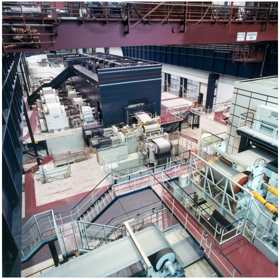
- Rolling mill prediction problem and data
- Dynamic Model Averaging (DMA)

Outline

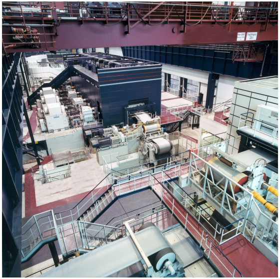
- Rolling mill prediction problem and data
- Dynamic Model Averaging (DMA)
- Results for rolling mill data

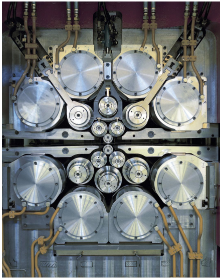
Cold Rolling Mill

Cold Rolling Mill

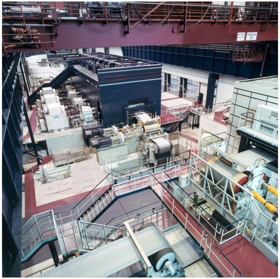


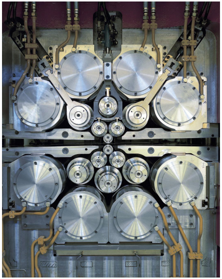
Cold Rolling Mill





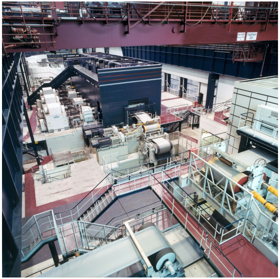
Cold Rolling Mill

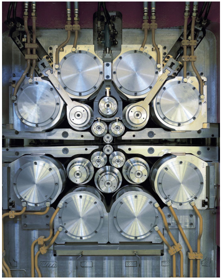




- Part of the process for making special alloys (originally steel):

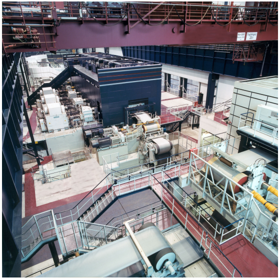
Cold Rolling Mill

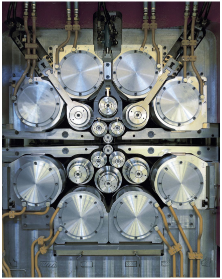




- Part of the process for making special alloys (originally steel):
  - Production of thin strips used, e.g., for computer chips.

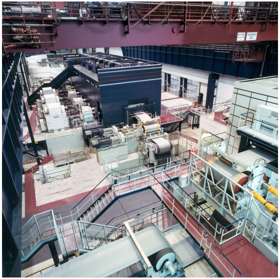
Cold Rolling Mill

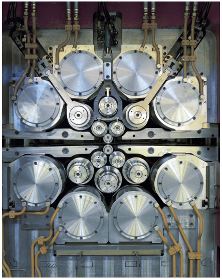




- Part of the process for making special alloys (originally steel):
  - Production of thin strips used, e.g., for computer chips.
  - Typical target thickness about 1000 micros (1mm).

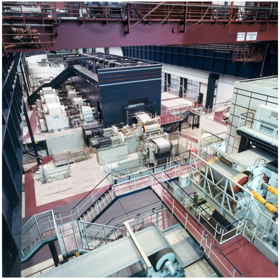
Cold Rolling Mill

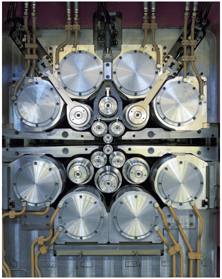




- Part of the process for making special alloys (originally steel):
  - Production of thin strips used, e.g., for computer chips.
  - Typical target thickness about 1000 micros (1mm).
  - Goal is to within 10 microns.

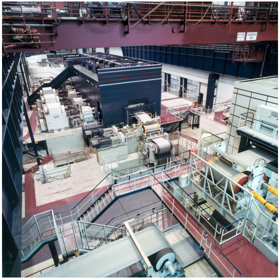
Cold Rolling Mill

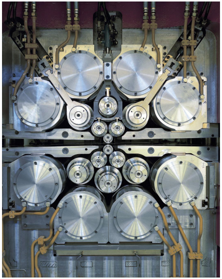




- Part of the process for making special alloys (originally steel):
  - Production of thin strips used, e.g., for computer chips.
  - Typical target thickness about 1000 micros (1mm).
  - Goal is to within 10 microns.
- Cold rolling reduces the thickness and gets it "right" (sometimes after hot rolling):

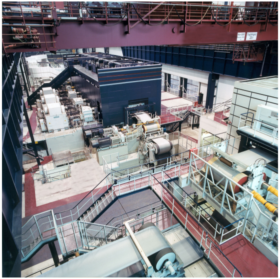
Cold Rolling Mill

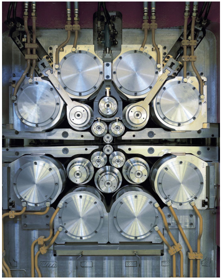




- Part of the process for making special alloys (originally steel):
  - Production of thin strips used, e.g., for computer chips.
  - Typical target thickness about 1000 micros (1mm).
  - Goal is to within 10 microns.
- Cold rolling reduces the thickness and gets it “right” (sometimes after hot rolling):
  - Metal sheet is passed through a gap and subjected to the rolling force.

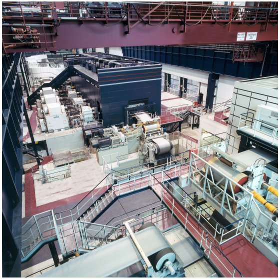
Cold Rolling Mill

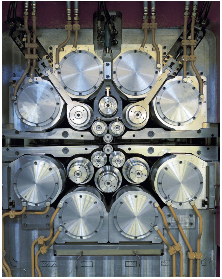




- Part of the process for making special alloys (originally steel):
  - Production of thin strips used, e.g., for computer chips.
  - Typical target thickness about 1000 micros (1mm).
  - Goal is to within 10 microns.
- Cold rolling reduces the thickness and gets it “right” (sometimes after hot rolling):
  - Metal sheet is passed through a gap and subjected to the rolling force.
  - Machine settings adjusted continuously (automatic control)

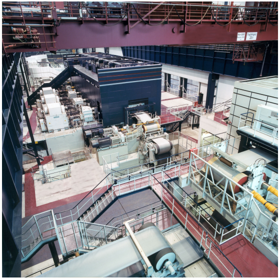
Cold Rolling Mill

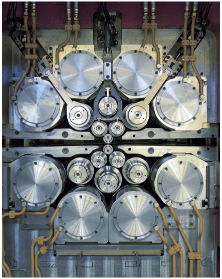




- Part of the process for making special alloys (originally steel):
  - Production of thin strips used, e.g., for computer chips.
  - Typical target thickness about 1000 micros (1mm).
  - Goal is to within 10 microns.
- Cold rolling reduces the thickness and gets it "right" (sometimes after hot rolling):
  - Metal sheet is passed through a gap and subjected to the rolling force.
  - Machine settings adjusted continuously (automatic control)
  - Online prediction can improve this

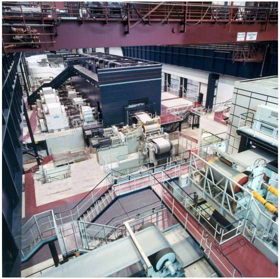
Cold Rolling Mill

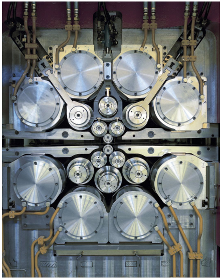




- Part of the process for making special alloys (originally steel):
    - Production of thin strips used, e.g., for computer chips.
    - Typical target thickness about 1000 micros (1mm).
    - Goal is to within 10 microns.
- Cold rolling reduces the thickness and gets it “right” (sometimes after hot rolling):
    - Metal sheet is passed through a gap and subjected to the rolling force.
    - Machine settings adjusted continuously (automatic control)
    - Online prediction can improve this
    - Initial period hardest to control ($\sim$500 samples discarded)

Cold Rolling Mill

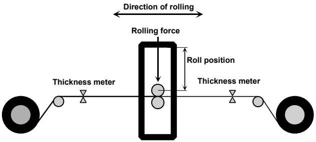




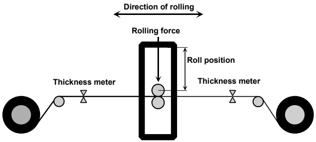
- Part of the process for making special alloys (originally steel):
  - Production of thin strips used, e.g., for computer chips.
  - Typical target thickness about 1000 micros (1mm).
  - Goal is to within 10 microns.
- Cold rolling reduces the thickness and gets it “right” (sometimes after hot rolling):
  - Metal sheet is passed through a gap and subjected to the rolling force.
  - Machine settings adjusted continuously (automatic control)
  - Online prediction can improve this
  - Initial period hardest to control (∼500 samples discarded)
  - Very large errors can harm metal sheet

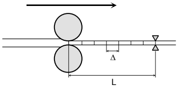
Rolling Mill Prediction Problem

Rolling Mill Prediction Problem

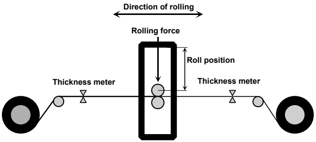


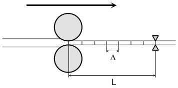
Rolling Mill Prediction Problem





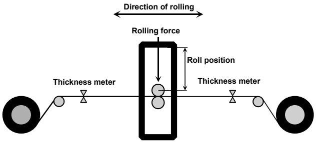
Rolling Mill Prediction Problem

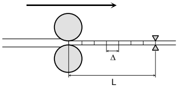




- Goal: Predict output thickness of samples of material. Variables:

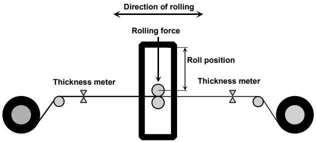
Rolling Mill Prediction Problem

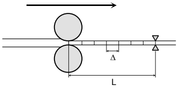




- Goal: Predict output thickness of samples of material. Variables:
  - y_t = deviation of output thickness from target value for sample t

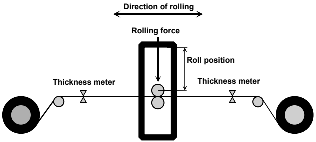
Rolling Mill Prediction Problem

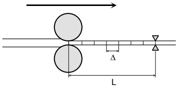




- Goal: Predict output thickness of samples of material. Variables:
  - y_t = deviation of output thickness from target value for sample t
  - u_t = deviation of input thickness from nominal value

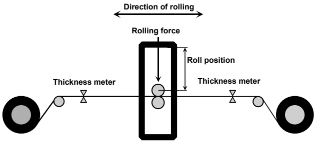
Rolling Mill Prediction Problem

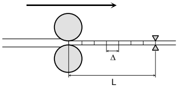




- Goal: Predict output thickness of samples of material. Variables:
  - y_t = deviation of output thickness from target value for sample t
  - u_t = deviation of input thickness from nominal value
  - v_t = size of gap between rolling cylinders (control variable)

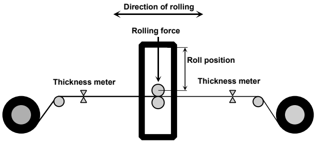
Rolling Mill Prediction Problem

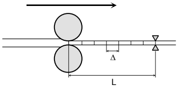




- Goal: Predict output thickness of samples of material. Variables:
  - y_t = deviation of output thickness from target value for sample t
  - u_t = deviation of input thickness from nominal value
  - v_t = size of gap between rolling cylinders (control variable)
  - w_t = ratio of input to output speeds (control variable)

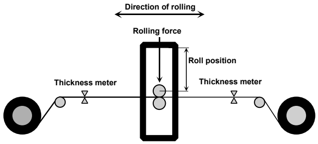
Rolling Mill Prediction Problem

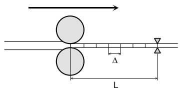




- Goal: Predict output thickness of samples of material. Variables:
  - y_t = deviation of output thickness from target value for sample t
  - u_t = deviation of input thickness from nominal value
  - v_t = size of gap between rolling cylinders (control variable)
  - w_t = ratio of input to output speeds (control variable)
  - z_t = rolling force (control variable)

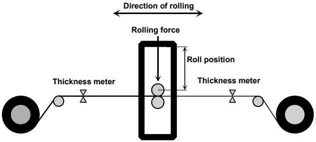
Rolling Mill Prediction Problem

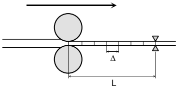




- Goal: Predict output thickness of samples of material. Variables:
  - y_t = deviation of output thickness from target value for sample t
  - u_t = deviation of input thickness from nominal value
  - v_t = size of gap between rolling cylinders (control variable)
  - w_t = ratio of input to output speeds (control variable)
  - z_t = rolling force (control variable)
- Time constraints: Samples processed rapidly ($\sim$ 20 milliseconds each), so calculations must be fast

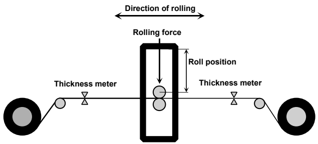
Rolling Mill Prediction Problem

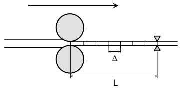




- Goal: Predict output thickness of samples of material. Variables:
    - y_t = deviation of output thickness from target value for sample t
    - u_t = deviation of input thickness from nominal value
    - v_t = size of gap between rolling cylinders (control variable)
    - w_t = ratio of input to output speeds (control variable)
    - z_t = rolling force (control variable)
- Time constraints: Samples processed rapidly ($\sim$ 20 milliseconds each), so calculations must be fast
- Time measurement delay: Data for estimating prediction model available only with a delay of $d = 24$ samples.

Rolling Mill Prediction Problem

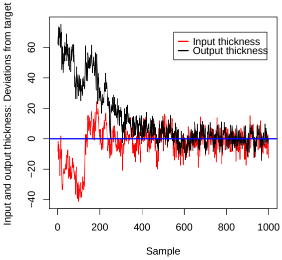




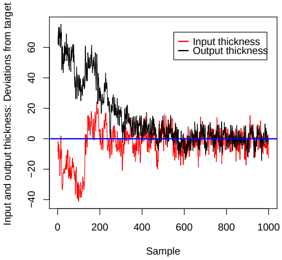
- Goal: Predict output thickness of samples of material. Variables:
  - y_t = deviation of output thickness from target value for sample t
  - u_t = deviation of input thickness from nominal value
  - v_t = size of gap between rolling cylinders (control variable)
  - w_t = ratio of input to output speeds (control variable)
  - z_t = rolling force (control variable)
- Time constraints: Samples processed rapidly ($\sim$ 20 milliseconds each), so calculations must be fast
- Time measurement delay: Data for estimating prediction model available only with a delay of $d = 24$ samples.
- Data: 19,058 samples: each 4cm long, 40ms in machine

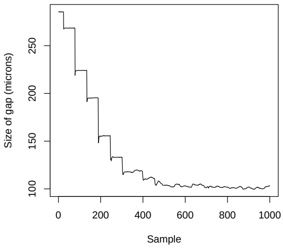
Data

Data

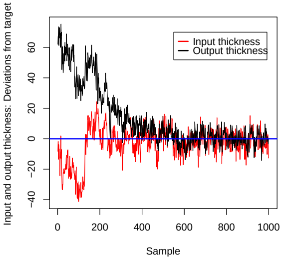

Input thickness
Output thickness
Input and output thickness: Deviations from target
−40
−20
0
20
40
60
0
200
400
600
800
1000
Sample

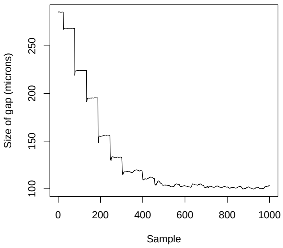
Data

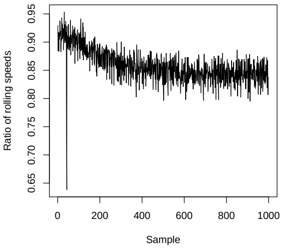

Input and output thickness: Deviations from target
Input thickness
Output thickness
Sample
0
200
400
600
800
1000
−40
−20
0
20
40
60


Size of gap (microns)
Sample
0
200
400
600
800
1000
100
150
200
250

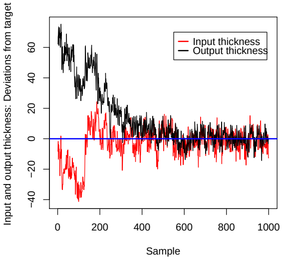
Data

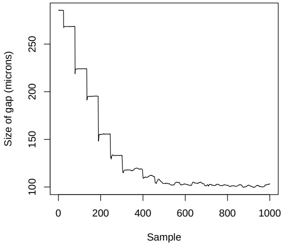

Input thickness
Output thickness
Input and output thickness: Deviations from target
−40
−20
0
20
40
60
0
200
400
600
800
1000
Sample

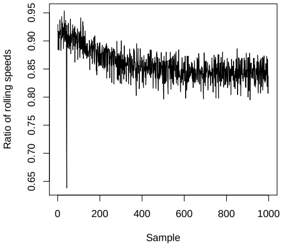

Size of gap (microns)
100
150
200
250
0
200
400
600
800
1000
Sample

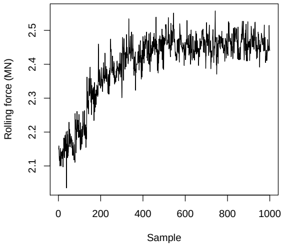

Ratio of rolling speeds
0.65
0.70
0.75
0.80
0.85
0.90
0.95
0
200
400
600
800
1000
Sample

Data


Input thickness
Output thickness
Input and output thickness: Deviations from target
−40
−20
0
20
40
60
0
200
400
600
800
1000
Sample


Size of gap (microns)
100
150
200
250
0
200
400
600
800
1000
Sample


Ratio of rolling speeds
0.65
0.70
0.75
0.80
0.85
0.90
0.95
0
200
400
600
800
1000
Sample


Rolling force (MN)
2.1
2.2
2.3
2.4
2.5
0
200
400
600
800
1000
Sample

Regression Approach: Ettler Models

Regression Approach: Ettler Models

- Which predictors to use?

Regression Approach: Ettler Models

- Which predictors to use?
- Ettler et al (2007) considered 3 models:

$$
\begin{aligned}
M_1 : \quad x_t^{(1)} &= (1, v_t, z_t), \\
M_2 : \quad x_t^{(2)} &= (1, w_t, u_t w_t), \\
M_3 : \quad x_t^{(3)} &= (1, u_t, v_t, w_t).
\end{aligned}
$$

M_1 and M_2 physically motivated; M_3 empirical

Regression Approach: Ettler Models

- Which predictors to use?
- Ettler et al (2007) considered 3 models:

$$\begin{aligned} M_1: \quad x_t^{(1)} &= (1, v_t, z_t), \\ M_2: \quad x_t^{(2)} &= (1, w_t, u_t w_t), \\ M_3: \quad x_t^{(3)} &= (1, u_t, v_t, w_t). \end{aligned}$$

  M_1 and M_2 physically motivated; M_3 empirical
- We propose Dynamic Model Averaging (DMA):

Regression Approach: Ettler Models

- Which predictors to use?
- Ettler et al (2007) considered 3 models:

$$
\begin{aligned}
M_1: \quad x_t^{(1)} &= (1, v_t, z_t), \\
M_2: \quad x_t^{(2)} &= (1, w_t, u_t w_t), \\
M_3: \quad x_t^{(3)} &= (1, u_t, v_t, w_t).
\end{aligned}
$$

  M_1 and M_2 physically motivated; M_3 empirical
- We propose Dynamic Model Averaging (DMA):
  - Dynamic extension of Bayesian model averaging (BMA) for regression (Raftery et al, 1997, JASA)

Regression Approach: Ettler Models

- Which predictors to use?
- Ettler et al (2007) considered 3 models:

$$
\begin{aligned}
M_1: \quad x_t^{(1)} &= (1, v_t, z_t), \\
M_2: \quad x_t^{(2)} &= (1, w_t, u_t w_t), \\
M_3: \quad x_t^{(3)} &= (1, u_t, v_t, w_t).
\end{aligned}
$$

  M_1 and M_2 physically motivated; M_3 empirical
- We propose Dynamic Model Averaging (DMA):
  - Dynamic extension of Bayesian model averaging (BMA) for regression (Raftery et al, 1997, JASA)
  - Parameters of each model are recursively updated

Regression Approach: Ettler Models

- Which predictors to use?
- Ettler et al (2007) considered 3 models:

$$\begin{aligned} M_1 : \quad x_t^{(1)} &= (1, v_t, z_t), \\ M_2 : \quad x_t^{(2)} &= (1, w_t, u_t w_t), \\ M_3 : \quad x_t^{(3)} &= (1, u_t, v_t, w_t). \end{aligned}$$

  M_1 and M_2 physically motivated; M_3 empirical
- We propose Dynamic Model Averaging (DMA):
  - Dynamic extension of Bayesian model averaging (BMA) for regression (Raftery et al, 1997, JASA)
  - Parameters of each model are recursively updated
  - Model indicator changes according to a Markov chain (model state equation), and is recursively updated.

Regression Approach: Ettler Models

- Which predictors to use?
- Ettler et al (2007) considered 3 models:

$$\begin{aligned} M_1: \quad x_t^{(1)} &= (1, v_t, z_t), \\ M_2: \quad x_t^{(2)} &= (1, w_t, u_t w_t), \\ M_3: \quad x_t^{(3)} &= (1, u_t, v_t, w_t). \end{aligned}$$

  M_1 and M_2 physically motivated; M_3 empirical
- We propose Dynamic Model Averaging (DMA):
  - Dynamic extension of Bayesian model averaging (BMA) for regression (Raftery et al, 1997, JASA)
  - Parameters of each model are recursively updated
  - Model indicator changes according to a Markov chain (model state equation), and is recursively updated.
  - Parameter state equation and model state equation both specified by forgetting.

Regression Approach: Ettler Models

- Which predictors to use?
- Ettler et al (2007) considered 3 models:

$$\begin{aligned} M_1 : \quad x_t^{(1)} &= (1, v_t, z_t), \\ M_2 : \quad x_t^{(2)} &= (1, w_t, u_t w_t), \\ M_3 : \quad x_t^{(3)} &= (1, u_t, v_t, w_t). \end{aligned}$$

  M_1 and M_2 physically motivated; M_3 empirical
- We propose Dynamic Model Averaging (DMA):
  - Dynamic extension of Bayesian model averaging (BMA) for regression (Raftery et al, 1997, JASA)
  - Parameters of each model are recursively updated
  - Model indicator changes according to a Markov chain (model state equation), and is recursively updated.
  - Parameter state equation and model state equation both specified by forgetting.
  - Version 1: Use 3 Ettler models

Regression Approach: Ettler Models

- Which predictors to use?
- Ettler et al (2007) considered 3 models:

$$
\begin{aligned}
M_1 : \quad x_t^{(1)} &= (1, v_t, z_t), \\
M_2 : \quad x_t^{(2)} &= (1, w_t, u_t w_t), \\
M_3 : \quad x_t^{(3)} &= (1, u_t, v_t, w_t).
\end{aligned}
$$

  M_1 and M_2 physically motivated; M_3 empirical
- We propose Dynamic Model Averaging (DMA):
  - Dynamic extension of Bayesian model averaging (BMA) for regression (Raftery et al, 1997, JASA)
  - Parameters of each model are recursively updated
  - Model indicator changes according to a Markov chain (model state equation), and is recursively updated.
  - Parameter state equation and model state equation both specified by forgetting.
  - Version 1: Use 3 Ettler models
  - Version 2: Consider all possible combinations of predictor variables that are not physically excluded.

One-Model Case

One-Model Case

- Standard Kalman filtering with forgetting and variance updating.

One-Model Case

- Standard Kalman filtering with forgetting and variance updating.
- Observation equation: $y_t = x_t^T \theta_t + \varepsilon_t$, where $\varepsilon_t \overset{\text{iid}}{\sim} N(0, V)$

One-Model Case

- Standard Kalman filtering with forgetting and variance updating.
- Observation equation: $y_t = x_t^T \theta_t + \varepsilon_t$, where $\varepsilon_t \overset{\text{iid}}{\sim} N(0, V)$
- State equation for regression parameters θ_t:

$$\theta_t = \theta_{t-1} + \delta_t, \quad \text{where } \delta_t \overset{\text{ind}}{\sim} N(0, W_t)$$

One-Model Case

- Standard Kalman filtering with forgetting and variance updating.
- Observation equation: $y_t = x_t^T \theta_t + \varepsilon_t$, where $\varepsilon_t \overset{\text{iid}}{\sim} N(0, V)$
- State equation for regression parameters θ_t:

$$\theta_t = \theta_{t-1} + \delta_t, \quad \text{where } \delta_t \overset{\text{ind}}{\sim} N(0, W_t)$$

- Estimation: Start with $\theta_{t-1} | Y^{t-1} \sim N(\hat{\theta}_{t-1}, \Sigma_{t-1})$.

One-Model Case

- Standard Kalman filtering with forgetting and variance updating.
- Observation equation: $y_t = x_t^T \theta_t + \varepsilon_t$, where $\varepsilon_t \overset{\text{iid}}{\sim} N(0, V)$
- State equation for regression parameters θ_t:

$$\theta_t = \theta_{t-1} + \delta_t, \quad \text{where } \delta_t \overset{\text{ind}}{\sim} N(0, W_t)$$

- Estimation: Start with $\theta_{t-1} | Y^{t-1} \sim N(\hat{\theta}_{t-1}, \Sigma_{t-1})$.
- Prediction equation:

$$\theta_t | Y^{t-1} \sim N(\hat{\theta}_{t-1}, R_t), \quad \text{where } R_t = \Sigma_{t-1} + W_t$$

One-Model Case

- Standard Kalman filtering with forgetting and variance updating.
- Observation equation: $y_t = x_t^T \theta_t + \varepsilon_t$, where $\varepsilon_t \overset{\text{iid}}{\sim} N(0, V)$
- State equation for regression parameters θ_t:

$$\theta_t = \theta_{t-1} + \delta_t, \quad \text{where } \delta_t \overset{\text{ind}}{\sim} N(0, W_t)$$

- Estimation: Start with $\theta_{t-1} | Y^{t-1} \sim N(\hat{\theta}_{t-1}, \Sigma_{t-1})$.
- Prediction equation:

$$\theta_t | Y^{t-1} \sim N(\hat{\theta}_{t-1}, R_t), \quad \text{where } R_t = \Sigma_{t-1} + W_t$$

- Forgetting: Instead use $R_t = \lambda^{-1} \Sigma_{t-1}$
  ($\lambda =$ *forgetting factor* $\approx 1-$ a bit)

One-Model Case

- Standard Kalman filtering with forgetting and variance updating.
- Observation equation: $y_t = x_t^T \theta_t + \varepsilon_t$, where $\varepsilon_t \overset{\text{iid}}{\sim} N(0, V)$
- State equation for regression parameters θ_t:

$$\theta_t = \theta_{t-1} + \delta_t, \quad \text{where } \delta_t \overset{\text{ind}}{\sim} N(0, W_t)$$

- Estimation: Start with $\theta_{t-1} | Y^{t-1} \sim N(\hat{\theta}_{t-1}, \Sigma_{t-1})$.
- Prediction equation:

$$\theta_t | Y^{t-1} \sim N(\hat{\theta}_{t-1}, R_t), \quad \text{where } R_t = \Sigma_{t-1} + W_t$$

- Forgetting: Instead use $R_t = \lambda^{-1} \Sigma_{t-1}$
  ($\lambda =$ *forgetting factor* $\approx 1-$ a bit)
- Updating equation: $\theta_t | Y^t \sim N(\hat{\theta}_t, \Sigma_t)$ ($\hat{\theta}_t$, Σ_t from Kalman filter).

One-Model Case

- Standard Kalman filtering with forgetting and variance updating.
- Observation equation: $y_t = x_t^T \theta_t + \varepsilon_t$, where $\varepsilon_t \overset{\text{iid}}{\sim} N(0, V)$
- State equation for regression parameters θ_t:

$$\theta_t = \theta_{t-1} + \delta_t, \quad \text{where } \delta_t \overset{\text{ind}}{\sim} N(0, W_t)$$

- Estimation: Start with $\theta_{t-1} | Y^{t-1} \sim N(\hat{\theta}_{t-1}, \Sigma_{t-1})$.
- Prediction equation:

$$\theta_t | Y^{t-1} \sim N(\hat{\theta}_{t-1}, R_t), \quad \text{where } R_t = \Sigma_{t-1} + W_t$$

- Forgetting: Instead use $R_t = \lambda^{-1} \Sigma_{t-1}$
  ($\lambda =$ *forgetting factor* $\approx 1-$ a bit)
- Updating equation: $\theta_t | Y^t \sim N(\hat{\theta}_t, \Sigma_t)$ ($\hat{\theta}_t$, Σ_t from Kalman filter).
- Prediction of system output: $\hat{y}_{t+d} = x_{t+d}^T \hat{\theta}_{t-1}$

One-Model Case

- Standard Kalman filtering with forgetting and variance updating.
- Observation equation: $y_t = x_t^T \theta_t + \varepsilon_t$, where $\varepsilon_t \overset{\text{iid}}{\sim} N(0, V)$
- State equation for regression parameters θ_t:

$$\theta_t = \theta_{t-1} + \delta_t, \quad \text{where } \delta_t \overset{\text{ind}}{\sim} N(0, W_t)$$

- Estimation: Start with $\theta_{t-1} | Y^{t-1} \sim N(\hat{\theta}_{t-1}, \Sigma_{t-1})$.
- Prediction equation:

$$\theta_t | Y^{t-1} \sim N(\hat{\theta}_{t-1}, R_t), \quad \text{where } R_t = \Sigma_{t-1} + W_t$$

- Forgetting: Instead use $R_t = \lambda^{-1} \Sigma_{t-1}$
  ($\lambda =$ *forgetting factor* $\approx 1-$ a bit)
- Updating equation: $\theta_t | Y^t \sim N(\hat{\theta}_t, \Sigma_t)$ ($\hat{\theta}_t$, Σ_t from Kalman filter).
- Prediction of system output: $\hat{y}_{t+d} = x_{t+d}^T \hat{\theta}_{t-1}$
- Initialization: $\hat{\theta}_0 = 0$, $\Sigma_0 =$ diagonal matrix with large elements.

Recursive Estimation of Innovations Variance, V

Recursive Estimation of Innovations Variance, V

- Based on the one-step predictor of y_t:

$$y_t | Y^{t-1} \sim N(x_t^T \hat{\theta}_{t-1}, V + x_t^T R_t x_t).$$

Recursive Estimation of Innovations Variance, V

- Based on the one-step predictor of y_t:

$$y_t | Y^{t-1} \sim N(x_t^T \hat{\theta}_{t-1}, V + x_t^T R_t x_t).$$

- Thus

$$V_t^* = \frac{1}{t} \sum_{r=1}^{t} \left[(y_t - x_t \hat{\theta}_{t-1})^2 - x_t^T R_t x_t \right]$$

  is a consistent estimator of V.

Recursive Estimation of Innovations Variance, V

- Based on the one-step predictor of y_t:

$$y_t | Y^{t-1} \sim N(x_t^T \hat{\theta}_{t-1}, V + x_t^T R_t x_t).$$

- Thus

$$V_t^* = \frac{1}{t} \sum_{r=1}^{t} \left[(y_t - x_t \hat{\theta}_{t-1})^2 - x_t^T R_t x_t \right]$$

is a consistent estimator of V.

- This leads to the recursive estimator

$$\hat{V}_t = \begin{cases} A_t & \text{if } A_t > 0; \\ \hat{V}_{t-1} & \text{otherwise,} \end{cases}$$

$$\text{where } A_t = \left(\frac{t-1}{t} \right) \hat{V}_{t-1} + \frac{1}{t} (e_t^2 - x_t^T R_t x_t).$$

Recursive Estimation of Innovations Variance, V

- Based on the one-step predictor of y_t:

$$y_t | Y^{t-1} \sim N(x_t^T \hat{\theta}_{t-1}, V + x_t^T R_t x_t).$$

- Thus

$$V_t^* = \frac{1}{t} \sum_{r=1}^{t} \left[(y_t - x_t \hat{\theta}_{t-1})^2 - x_t^T R_t x_t \right]$$

is a consistent estimator of V.

- This leads to the recursive estimator

$$\hat{V}_t = \begin{cases} A_t & \text{if } A_t > 0; \\ \hat{V}_{t-1} & \text{otherwise,} \end{cases}$$

$$\text{where } A_t = \left(\frac{t-1}{t} \right) \hat{V}_{t-1} + \frac{1}{t} (e_t^2 - x_t^T R_t x_t).$$

- Adaptive version (not implemented):

$$A_t = \lambda \hat{V}_{t-1} + (1 - \lambda)(e_t^2 - x_t^T R_t x_t)$$

Recursive Estimation of Innovations Variance, V

- Based on the one-step predictor of y_t:

$$y_t | Y^{t-1} \sim N(x_t^T \hat{\theta}_{t-1}, V + x_t^T R_t x_t).$$

- Thus

$$V_t^* = \frac{1}{t} \sum_{r=1}^{t} \left[(y_t - x_t \hat{\theta}_{t-1})^2 - x_t^T R_t x_t \right]$$

  is a consistent estimator of V.
- This leads to the recursive estimator

$$\hat{V}_t = \begin{cases} A_t & \text{if } A_t > 0; \\ \hat{V}_{t-1} & \text{otherwise,} \end{cases}$$

$$\text{where } A_t = \left(\frac{t-1}{t} \right) \hat{V}_{t-1} + \frac{1}{t} (e_t^2 - x_t^T R_t x_t).$$

- Adaptive version (not implemented):

$$A_t = \lambda \hat{V}_{t-1} + (1 - \lambda)(e_t^2 - x_t^T R_t x_t)$$

- Previously in literature?

Dynamic Model Averaging (DMA)

Dynamic Model Averaging (DMA)

- Models $M_1, \ldots, M_K$.

Dynamic Model Averaging (DMA)

- Models $M_1, \ldots, M_K$.
  - Model indicator: $L_t = k$ if model M_k is operating for sample t.

Dynamic Model Averaging (DMA)

- Models $M_1, \ldots, M_K$.
  - Model indicator: $L_t = k$ if model M_k is operating for sample t.
- Observation equation:

$$y_t | L_t = k \sim N(x_t^{(k)T} \theta_t^{(k)}, V^{(k)})$$

Dynamic Model Averaging (DMA)

- Models $M_1, \ldots, M_K$.
  - Model indicator: $L_t = k$ if model M_k is operating for sample t.
- Observation equation:

$$y_t | L_t = k \sim N(x_t^{(k)T} \theta_t^{(k)}, V^{(k)})$$

- Parameter state equation:

$$\theta_t^{(k)} | M_k \sim N(\theta_{t-1}^{(k)}, W_t^{(k)})$$

Dynamic Model Averaging (DMA)

- Models $M_1, \ldots, M_K$.
  - Model indicator: $L_t = k$ if model M_k is operating for sample t.
- Observation equation:

$$y_t | L_t = k \sim N(x_t^{(k)T} \theta_t^{(k)}, V^{(k)})$$

- Parameter state equation:

$$\theta_t^{(k)} | M_k \sim N(\theta_{t-1}^{(k)}, W_t^{(k)})$$

- Model state equation: L_t changes slowly according to a Markov chain determined by the transition matrix $Q = (q_{k\ell})$, where

$$q_{k\ell} = P[L_t = \ell | L_{t-1} = k]$$

Estimation and Prediction in DMA

Estimation and Prediction in DMA

- Estimation: Let $\pi_{t-1|t-1,\ell} = P[L_{t-1} = \ell | Y^{t-1}]$. *Model prediction equation*:

$$\pi_{t|t-1,k} \equiv P[L_t = k | Y^{t-1}] = \sum_{\ell=1}^{K} \pi_{t-1|t-1,\ell} q_{k\ell}$$

Estimation and Prediction in DMA

- Estimation: Let $\pi_{t-1|t-1,\ell} = P[L_{t-1} = \ell | Y^{t-1}]$. *Model prediction equation*:

$$\pi_{t|t-1,k} \equiv P[L_t = k | Y^{t-1}] = \sum_{\ell=1}^{K} \pi_{t-1|t-1,\ell} q_{k\ell}$$

- Forgetting: Instead, use

$$\pi_{t|t-1,k} = \frac{\pi^{\alpha}_{t-1|t-1,k}}{\sum_{\ell=1}^{K} \pi^{\alpha}_{t-1|t-1,\ell}},$$

where α is the *model forgetting factor* ($\approx 1-$ a bit)

Estimation and Prediction in DMA

- Estimation: Let $\pi_{t-1|t-1,\ell} = P[L_{t-1} = \ell | Y^{t-1}]$. *Model prediction equation*:

$$\pi_{t|t-1,k} \equiv P[L_t = k | Y^{t-1}] = \sum_{\ell=1}^{K} \pi_{t-1|t-1,\ell} q_{k\ell}$$

- Forgetting: Instead, use

$$\pi_{t|t-1,k} = \frac{\pi^{\alpha}_{t-1|t-1,k}}{\sum_{\ell=1}^{K} \pi^{\alpha}_{t-1|t-1,\ell}},$$

  where α is the *model forgetting factor* ($\approx 1-$ a bit)
- Model updating equation:

$$\pi_{t|t,k} = \omega_{tk} / \sum_{\ell=1}^{K} \omega_{t\ell}, \quad \text{where} \quad \omega_{t\ell} = \pi_{t|t-1,\ell}\, p(y_t | Y^{t-1}, L_t = \ell)$$

Estimation and Prediction in DMA

- Estimation: Let $\pi_{t-1|t-1,\ell} = P[L_{t-1} = \ell | Y^{t-1}]$. *Model prediction equation*:

$$\pi_{t|t-1,k} \equiv P[L_t = k | Y^{t-1}] = \sum_{\ell=1}^{K} \pi_{t-1|t-1,\ell} q_{k\ell}$$

- Forgetting: Instead, use

$$\pi_{t|t-1,k} = \frac{\pi^{\alpha}_{t-1|t-1,k}}{\sum_{\ell=1}^{K} \pi^{\alpha}_{t-1|t-1,\ell}},$$

  where α is the *model forgetting factor* ($\approx 1-$ a bit)
- Model updating equation:

$$\pi_{t|t,k} = \omega_{tk} / \sum_{\ell=1}^{K} \omega_{t\ell}, \quad \text{where} \quad \omega_{t\ell} = \pi_{t|t-1,\ell} \, p(y_t | Y^{t-1}, L_t = \ell)$$

- System output prediction:

$$\hat{y}^{\mathrm{DMA}}_{t+d} = \sum_{k=1}^{K} \pi_{t|t-1,k} \, \hat{y}^{(k)}_{t+d} = \sum_{k=1}^{K} \pi_{t|t-1,k} \, x^{(k)T}_{t+d} \, \hat{\theta}^{(k)}_{t-1}$$

Comments on DMA

Comments on DMA

- Combining Kalman filtering and an unobserved Markov chain is an old idea: the *Conditional Linear Dynamic Model* (Ackerson & Fu 1970, IEEE TAC; Harrison & Stevens 1971; West & Harrison 1989; Chen & Liu 2000, JRSS B)

Comments on DMA

- Combining Kalman filtering and an unobserved Markov chain is an old idea: the *Conditional Linear Dynamic Model* (Ackerson & Fu 1970, IEEE TAC; Harrison & Stevens 1971; West & Harrison 1989; Chen & Liu 2000, JRSS B)
  - Also used in speech recognition and genomics (*Hidden Markov Model*), economics (*Markov switching model*), tracking objects in aerospace engineering (*Interacting Multiple Models algorithm*)

Comments on DMA

- Combining Kalman filtering and an unobserved Markov chain is an old idea: the *Conditional Linear Dynamic Model* (Ackerson & Fu 1970, IEEE TAC; Harrison & Stevens 1971; West & Harrison 1989; Chen & Liu 2000, JRSS B)
  - Also used in speech recognition and genomics (*Hidden Markov Model*), economics (*Markov switching model*), tracking objects in aerospace engineering (*Interacting Multiple Models algorithm*)
- *But* the DMA model is not quite a special case of the CDLM

Comments on DMA

- Combining Kalman filtering and an unobserved Markov chain is an old idea: the *Conditional Linear Dynamic Model* (Ackerson & Fu 1970, IEEE TAC; Harrison & Stevens 1971; West & Harrison 1989; Chen & Liu 2000, JRSS B)
  - Also used in speech recognition and genomics (*Hidden Markov Model*), economics (*Markov switching model*), tracking objects in aerospace engineering (*Interacting Multiple Models algorithm*)
- *But* the DMA model is not quite a special case of the CDLM
  - because the state $\theta_t^{(k)}$ is different for each model.

Comments on DMA

- Combining Kalman filtering and an unobserved Markov chain is an old idea: the *Conditional Linear Dynamic Model* (Ackerson & Fu 1970, IEEE TAC; Harrison & Stevens 1971; West & Harrison 1989; Chen & Liu 2000, JRSS B)
  - Also used in speech recognition and genomics (*Hidden Markov Model*), economics (*Markov switching model*), tracking objects in aerospace engineering (*Interacting Multiple Models algorithm*)
- *But* the DMA model is not quite a special case of the CDLM
  - because the state $\theta_t^{(k)}$ is different for each model.
- Updating each model at each step is only an approximation to the exact posterior distribution (which has the usual exponential explosion in the number of terms)

Comments on DMA

- Combining Kalman filtering and an unobserved Markov chain is an old idea: the *Conditional Linear Dynamic Model* (Ackerson & Fu 1970, IEEE TAC; Harrison & Stevens 1971; West & Harrison 1989; Chen & Liu 2000, JRSS B)
    - Also used in speech recognition and genomics (*Hidden Markov Model*), economics (*Markov switching model*), tracking objects in aerospace engineering (*Interacting Multiple Models algorithm*)
- *But* the DMA model is not quite a special case of the CDLM
    - because the state $\theta_t^{(k)}$ is different for each model.
- Updating each model at each step is only an approximation to the exact posterior distribution (which has the usual exponential explosion in the number of terms)
    - Reasonable because the predictive distribution of y_{t+d} depends only on the *conditional* distribution of $\theta_t^{(k)}$ given that $L_t = k$.

Comments on DMA

- Combining Kalman filtering and an unobserved Markov chain is an old idea: the *Conditional Linear Dynamic Model* (Ackerson & Fu 1970, IEEE TAC; Harrison & Stevens 1971; West & Harrison 1989; Chen & Liu 2000, JRSS B)
  - Also used in speech recognition and genomics (*Hidden Markov Model*), economics (*Markov switching model*), tracking objects in aerospace engineering (*Interacting Multiple Models algorithm*)
- *But* the DMA model is not quite a special case of the CDLM
  - because the state $\theta_t^{(k)}$ is different for each model.
- Updating each model at each step is only an approximation to the exact posterior distribution (which has the usual exponential explosion in the number of terms)
  - Reasonable because the predictive distribution of y_{t+d} depends only on the *conditional* distribution of $\theta_t^{(k)}$ given that $L_t = k$.
  - Also because it leads to a forgetting version of Bayes factors and Bayesian model averaging, generalizing Dawid (1984, JRSS A).

DMA and BMA

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.
- BMA predictive distribution:
  $p(y_{T+d}|Y^T) = \sum_{k=1}^{K} p(y_{T+d}|Y^T, M_k)p(M_k|Y^T)$

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.
- BMA predictive distribution: $p(y_{T+d}|Y^T) = \sum_{k=1}^{K} p(y_{T+d}|Y^T, M_k)p(M_k|Y^T)$
- Posterior model probabilities: $p(M_k|Y^T) \propto p(Y^T|M_k)p(M_k)$

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.
- BMA predictive distribution:
  $p(y_{T+d}|Y^T) = \sum_{k=1}^{K} p(y_{T+d}|Y^T, M_k)p(M_k|Y^T)$
- Posterior model probabilities: $p(M_k|Y^T) \propto p(Y^T|M_k)p(M_k)$
- Integrated likelihood:
  $p(Y^T|M_k) = \int p(Y^T|\theta^{(k)}, M_k)p(\theta^{(k)}|M_k)d\theta^{(k)}$

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.
- BMA predictive distribution: $p(y_{T+d}|Y^T) = \sum_{k=1}^{K} p(y_{T+d}|Y^T, M_k)p(M_k|Y^T)$
- Posterior model probabilities: $p(M_k|Y^T) \propto p(Y^T|M_k)p(M_k)$
- Integrated likelihood: $p(Y^T|M_k) = \int p(Y^T|\theta^{(k)}, M_k)p(\theta^{(k)}|M_k)d\theta^{(k)}$
- Prequential version (Dawid 1984): $p(Y^T|M_k) = \prod_{t=1}^{T} p(y_t|Y^{t-1}, M_k)$

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.
- BMA predictive distribution: $p(y_{T+d}|Y^T) = \sum_{k=1}^{K} p(y_{T+d}|Y^T, M_k)p(M_k|Y^T)$
- Posterior model probabilities: $p(M_k|Y^T) \propto p(Y^T|M_k)p(M_k)$
- Integrated likelihood: $p(Y^T|M_k) = \int p(Y^T|\theta^{(k)}, M_k)p(\theta^{(k)}|M_k)d\theta^{(k)}$
- Prequential version (Dawid 1984): $p(Y^T|M_k) = \prod_{t=1}^{T} p(y_t|Y^{t-1}, M_k)$
- Bayes factor for M_k against M_ℓ: $B_{k\ell} = p(Y^T|M_k)/p(Y^T|M_\ell)$

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.
- BMA predictive distribution:
  $p(y_{T+d}|Y^T) = \sum_{k=1}^{K} p(y_{T+d}|Y^T, M_k)p(M_k|Y^T)$
- Posterior model probabilities: $p(M_k|Y^T) \propto p(Y^T|M_k)p(M_k)$
- Integrated likelihood:
  $p(Y^T|M_k) = \int p(Y^T|\theta^{(k)}, M_k)p(\theta^{(k)}|M_k)d\theta^{(k)}$
- Prequential version (Dawid 1984):
  $p(Y^T|M_k) = \prod_{t=1}^{T} p(y_t|Y^{t-1}, M_k)$
- Bayes factor for M_k against M_ℓ: $B_{k\ell} = p(Y^T|M_k)/p(Y^T|M_\ell)$
- It can also be written as $\log B_{k\ell} = \sum_{t=1}^{T} \log B_{k\ell,t}$, where $B_{k\ell,t} = p(y_t|Y^{t-1}, M_k)/p(y_t|Y^{t-1}, M_\ell)$ is the *sample-specific Bayes factor* for sample t.

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.
- BMA predictive distribution: $p(y_{T+d}|Y^T) = \sum_{k=1}^K p(y_{T+d}|Y^T, M_k)p(M_k|Y^T)$
- Posterior model probabilities: $p(M_k|Y^T) \propto p(Y^T|M_k)p(M_k)$
- Integrated likelihood: $p(Y^T|M_k) = \int p(Y^T|\theta^{(k)}, M_k)p(\theta^{(k)}|M_k)d\theta^{(k)}$
- Prequential version (Dawid 1984): $p(Y^T|M_k) = \prod_{t=1}^T p(y_t|Y^{t-1}, M_k)$
- Bayes factor for M_k against M_ℓ: $B_{k\ell} = p(Y^T|M_k)/p(Y^T|M_\ell)$
- It can also be written as $\log B_{k\ell} = \sum_{t=1}^T \log B_{k\ell,t}$, where $B_{k\ell,t} = p(y_t|Y^{t-1}, M_k)/p(y_t|Y^{t-1}, M_\ell)$ is the *sample-specific Bayes factor* for sample t.
- In DMA: $\log\left(\frac{\pi_{T|T,k}}{\pi_{T|T,\ell}}\right) = \sum_{t=1}^T \alpha^{T-t} \log B_{k\ell,t}$.

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.
- BMA predictive distribution:
  $p(y_{T+d}|Y^T) = \sum_{k=1}^{K} p(y_{T+d}|Y^T, M_k)p(M_k|Y^T)$
- Posterior model probabilities: $p(M_k|Y^T) \propto p(Y^T|M_k)p(M_k)$
- Integrated likelihood:
  $p(Y^T|M_k) = \int p(Y^T|\theta^{(k)}, M_k)p(\theta^{(k)}|M_k)d\theta^{(k)}$
- Prequential version (Dawid 1984):
  $p(Y^T|M_k) = \prod_{t=1}^{T} p(y_t|Y^{t-1}, M_k)$
- Bayes factor for M_k against M_ℓ: $B_{k\ell} = p(Y^T|M_k)/p(Y^T|M_\ell)$
- It can also be written as $\log B_{k\ell} = \sum_{t=1}^{T} \log B_{k\ell,t}$, where $B_{k\ell,t} = p(y_t|Y^{t-1}, M_k)/p(y_t|Y^{t-1}, M_\ell)$ is the *sample-specific Bayes factor* for sample t.
- In DMA: $\log\left(\frac{\pi_{T|T,k}}{\pi_{T|T,\ell}}\right) = \sum_{t=1}^{T} \alpha^{T-t} \log B_{k\ell,t}$.
- Thus in DMA, the log posterior model odds at time T is an *exponentially age-discounted sum of sample-specific log Bayes factors*

DMA and BMA

- In static BMA, the correct model M_k and its parameter vector $\theta^{(k)}$ are fixed but unknown.
- BMA predictive distribution:
  $p(y_{T+d}|Y^T) = \sum_{k=1}^{K} p(y_{T+d}|Y^T, M_k)p(M_k|Y^T)$
- Posterior model probabilities: $p(M_k|Y^T) \propto p(Y^T|M_k)p(M_k)$
- Integrated likelihood:
  $p(Y^T|M_k) = \int p(Y^T|\theta^{(k)}, M_k)p(\theta^{(k)}|M_k)d\theta^{(k)}$
- Prequential version (Dawid 1984):
  $p(Y^T|M_k) = \prod_{t=1}^{T} p(y_t|Y^{t-1}, M_k)$
- Bayes factor for M_k against M_ℓ: $B_{k\ell} = p(Y^T|M_k)/p(Y^T|M_\ell)$
- It can also be written as $\log B_{k\ell} = \sum_{t=1}^{T} \log B_{k\ell,t}$, where $B_{k\ell,t} = p(y_t|Y^{t-1}, M_k)/p(y_t|Y^{t-1}, M_\ell)$ is the *sample-specific Bayes factor* for sample t.
- In DMA: $\log\left(\frac{\pi_{T|T,k}}{\pi_{T|T,\ell}}\right) = \sum_{t=1}^{T} \alpha^{T-t} \log B_{k\ell,t}$.
- Thus in DMA, the log posterior model odds at time T is an *exponentially age-discounted sum of sample-specific log Bayes factors*
- When $\alpha = \lambda = 1$ there is no forgetting and we recover static BMA, *in a recursive implementation*

Rolling Mill: Posterior Probabilities of 3 Ettler Models

Rolling Mill: Posterior Probabilities of 3 Ettler Models

#	Variables				
	u_t	v_t	w_t	z_t	$(u_t w_t)$
1	-	✓	-	✓	-
2	-	-	✓	-	✓
3	✓	✓	✓	-	-

Rolling Mill: Posterior Probabilities of 3 Ettler Models

#	Variables				
	u_t	v_t	w_t	z_t	$(u_t w_t)$
1	-	✓	-	✓	-
2	-	-	✓	-	✓
3	✓	✓	✓	-	-

(a) Post model probs: Samples 26–200

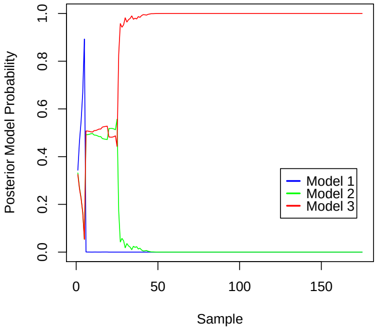
(b) All samples, 26–19058

Rolling Mill: Posterior Probabilities of 3 Ettler Models

#	Variables				
	u_t	v_t	w_t	z_t	$(u_t w_t)$
1	-	✓	-	✓	-
2	-	-	✓	-	✓
3	✓	✓	✓	-	-

(a) Post model probs: Samples 26–200

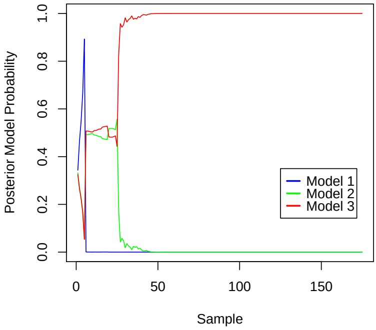
(b) All samples, 26–19058



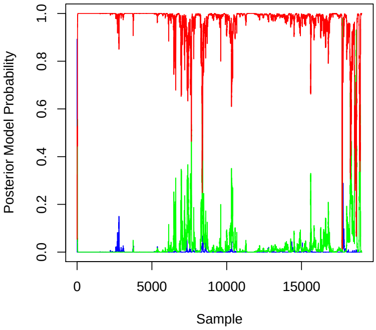
Rolling Mill: Posterior Probabilities of 3 Ettler Models

#	Variables				
	u_t	v_t	w_t	z_t	$(u_t w_t)$
1	-	✓	-	✓	-
2	-	-	✓	-	✓
3	✓	✓	✓	-	-

(a) Post model probs: Samples 26–200



(b) All samples, 26–19058



Results for 3 Ettler Models

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175			

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86			

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118			

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118	26.8	41.4	914

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118	26.8	41.4	914
Model 3	77.5	27.3	46			

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118	26.8	41.4	914
Model 3	77.5	27.3	46	20.7	31.1	523

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118	26.8	41.4	914
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45			

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118	26.8	41.4	914
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118	26.8	41.4	914
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520

- M_3 was much better than M_1 or M_2: Found fast by DMA

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118	26.8	41.4	914
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520

- M_3 was much better than M_1 or M_2: Found fast by DMA
- DMA with 3 models was slighly better than M_3 in the initial unstable period, and the same in the later stable period

Results for 3 Ettler Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 1	243.6	38.3	86	26.2	31.1	989
Model 2	345.5	41.7	118	26.8	41.4	914
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520

- M_3 was much better than M_1 or M_2: Found fast by DMA
- DMA with 3 models was slighly better than M_3 in the initial unstable period, and the same in the later stable period
- No price paid for model uncertainty even when one model best by far

Posterior Model Probabilities for All 17 Models

Posterior Model Probabilities for All 17 Models

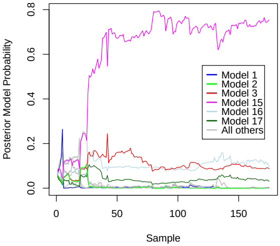
#	Variables				
	u_t	v_t	w_t	z_t	$(u_t w_t)$
1	-	✓	-	✓	-
2	-	-	✓	-	✓
3	✓	✓	✓	-	-

Posterior Model Probabilities for All 17 Models

#	Variables				
	u_t	v_t	w_t	z_t	$(u_t w_t)$
1	-	✓	-	✓	-
2	-	-	✓	-	✓
3	✓	✓	✓	-	-
4	-	-	-	-	-
5	-	-	-	✓	-
6	-	-	✓	-	-
7	-	-	✓	✓	-
8	-	✓	-	-	-
9	-	✓	✓	-	-
10	-	✓	✓	✓	-
11	✓	-	-	-	-
12	✓	-	-	✓	-
13	✓	-	✓	-	-
14	✓	-	✓	✓	-
15	✓	✓	-	-	-
16	✓	✓	-	✓	-
17	✓	✓	✓	✓	-

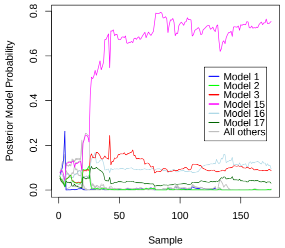
Posterior Model Probabilities for All 17 Models

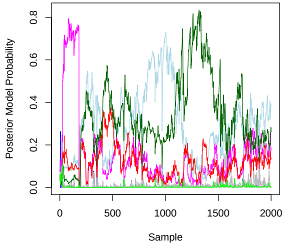
#	Variables				
	u_t	v_t	w_t	z_t	$(u_t w_t)$
1	-	✓	-	✓	-
2	-	-	✓	-	✓
3	✓	✓	✓	-	-
4	-	-	-	-	-
5	-	-	-	✓	-
6	-	-	✓	-	-
7	-	-	✓	✓	-
8	-	✓	-	-	-
9	-	✓	✓	-	-
10	-	✓	✓	✓	-
11	✓	-	-	-	-
12	✓	-	-	✓	-
13	✓	-	✓	-	-
14	✓	-	✓	✓	-
15	✓	✓	-	-	-
16	✓	✓	-	✓	-
17	✓	✓	✓	✓	-



Posterior Model Probabilities for All 17 Models

#	Variables				
	u_t	v_t	w_t	z_t	$(u_t w_t)$
1	-	✓	-	✓	-
2	-	-	✓	-	✓
3	✓	✓	✓	-	-
4	-	-	-	-	-
5	-	-	-	✓	-
6	-	-	✓	-	-
7	-	-	✓	✓	-
8	-	✓	-	-	-
9	-	✓	✓	-	-
10	-	✓	✓	✓	-
11	✓	-	-	-	-
12	✓	-	-	✓	-
13	✓	-	✓	-	-
14	✓	-	✓	✓	-
15	✓	✓	-	-	-
16	✓	✓	-	✓	-
17	✓	✓	✓	✓	-





Results for All 17 Models

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42			

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

- Only 4 models (M_3, M_{15}, M_{16}, M_{17}) had weight past sample 30

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

- Only 4 models (M_3, M_{15}, M_{16}, M_{17}) had weight past sample 30
    - In the unstable period, the simpler M_{15} had high weight

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

- Only 4 models (M_3, M_{15}, M_{16}, M_{17}) had weight past sample 30
  - In the unstable period, the simpler M_{15} had high weight
  - In the stable period, the more complex M_{16} and M_{17} had more weight

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

- Only 4 models (M_3, M_{15}, M_{16}, M_{17}) had weight past sample 30
  - In the unstable period, the simpler M_{15} had high weight
  - In the stable period, the more complex M_{16} and M_{17} had more weight
  - DMA adapts to more complex models as more data become available.

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

- Only 4 models (M_3, M_{15}, M_{16}, M_{17}) had weight past sample 30
  - In the unstable period, the simpler M_{15} had high weight
  - In the stable period, the more complex M_{16} and M_{17} had more weight
  - DMA adapts to more complex models as more data become available.
  - DMA gave a parsimonious solution even with a larger model space.

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

- Only 4 models (M_3, M_{15}, M_{16}, M_{17}) had weight past sample 30
    - In the unstable period, the simpler M_{15} had high weight
    - In the stable period, the more complex M_{16} and M_{17} had more weight
    - DMA adapts to more complex models as more data become available.
    - DMA gave a parsimonious solution even with a larger model space.
- DMA with all 17 models was clearly better in the unstable period (11% gain in MSE over M_3) and slightly better in the stable period.

Results for All 17 Models

Method	Samples 26–200			Samples 201–19058		
	MSE	MaxAE	#AE>10	MSE	MaxAE	#AE>10
Observed	2179.8	68.8	175	30.6	43.1	1183
Model 3	77.5	27.3	46	20.7	31.1	523
DMA – 3 models	76.1	26.3	45	20.7	31.1	520
DMA – 17 models	68.9	22.0	42	20.6	31.1	519

- Only 4 models (M_3, M_{15}, M_{16}, M_{17}) had weight past sample 30
  - In the unstable period, the simpler M_{15} had high weight
  - In the stable period, the more complex M_{16} and M_{17} had more weight
  - DMA adapts to more complex models as more data become available.
  - DMA gave a parsimonious solution even with a larger model space.
- DMA with all 17 models was clearly better in the unstable period (11% gain in MSE over M_3) and slightly better in the stable period.
- Only $\sim$50 samples discarded compared with 500 $\Longrightarrow$ waste reduced by $\sim$90%

Timing

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.
- Several orders of magnitude faster than a windowing algorithm.

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.
- Several orders of magnitude faster than a windowing algorithm.
- Gains by a factor of at least 40 possible with

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.
- Several orders of magnitude faster than a windowing algorithm.
- Gains by a factor of at least 40 possible with
  - better software (compiled)

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.
- Several orders of magnitude faster than a windowing algorithm.
- Gains by a factor of at least 40 possible with
  - better software (compiled)
  - better hardware

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.
- Several orders of magnitude faster than a windowing algorithm.
- Gains by a factor of at least 40 possible with
    - better software (compiled)
    - better hardware
    - $\Longrightarrow$ 0.05 ms/model/sample.

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.
- Several orders of magnitude faster than a windowing algorithm.
- Gains by a factor of at least 40 possible with
  - better software (compiled)
  - better hardware
  - $\Longrightarrow$ 0.05 ms/model/sample.
- If computation has to be done in 20 ms/sample, this would allow processing of 400 models

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.
- Several orders of magnitude faster than a windowing algorithm.
- Gains by a factor of at least 40 possible with
  - better software (compiled)
  - better hardware
  - $\Longrightarrow$ 0.05 ms/model/sample.
- If computation has to be done in 20 ms/sample, this would allow processing of 400 models
  - $\Longrightarrow$ DMA feasible in real time for cold rolling mill.

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.
- Several orders of magnitude faster than a windowing algorithm.
- Gains by a factor of at least 40 possible with
    - better software (compiled)
    - better hardware
    - $\Longrightarrow$ 0.05 ms/model/sample.
- If computation has to be done in 20 ms/sample, this would allow processing of 400 models
    - $\Longrightarrow$ DMA feasible in real time for cold rolling mill.
- Recursive implementation may be useful for static BMA too

Timing

- About 2 milliseconds/model/sample on my 2005 Mac laptop in the interpreted statistical language R.
- Several orders of magnitude faster than a windowing algorithm.
- Gains by a factor of at least 40 possible with
  - better software (compiled)
  - better hardware
  - $\Longrightarrow$ 0.05 ms/model/sample.
- If computation has to be done in 20 ms/sample, this would allow processing of 400 models
  - $\Longrightarrow$ DMA feasible in real time for cold rolling mill.
- Recursive implementation may be useful for static BMA too
- Computational requirements preclude computationally intensive methods (e.g. MCMC, online estimation of forgetting factors, ...)

Summary

Summary

- Motivating problem:

Summary

- Motivating problem:
  - online prediction of a cold rolling mill under model uncertainty

Summary

- Motivating problem:
  - online prediction of a cold rolling mill under model uncertainty
  - severe computational constraints in real-time setting

Summary

- Motivating problem:
  - online prediction of a cold rolling mill under model uncertainty
  - severe computational constraints in real-time setting
- Dynamic model averaging (DMA):

Summary

- Motivating problem:
  - online prediction of a cold rolling mill under model uncertainty
  - severe computational constraints in real-time setting
- Dynamic model averaging (DMA):
  - Model indicator and model parameters evolve in time (hidden Markov model)

Summary

- Motivating problem:
  - online prediction of a cold rolling mill under model uncertainty
  - severe computational constraints in real-time setting
- Dynamic model averaging (DMA):
  - Model indicator and model parameters evolve in time (hidden Markov model)
  - Recursive implementation is computationally efficient

Summary

- Motivating problem:
  - online prediction of a cold rolling mill under model uncertainty
  - severe computational constraints in real-time setting
- Dynamic model averaging (DMA):
  - Model indicator and model parameters evolve in time (hidden Markov model)
  - Recursive implementation is computationally efficient
  - Equivalent to an exponentially age-discounted version of static BMA

Summary

- Motivating problem:
  - online prediction of a cold rolling mill under model uncertainty
  - severe computational constraints in real-time setting
- Dynamic model averaging (DMA):
  - Model indicator and model parameters evolve in time (hidden Markov model)
  - Recursive implementation is computationally efficient
  - Equivalent to an exponentially age-discounted version of static BMA
- Gave better results than a single model for online prediction of a cold rolling mill, and could reduce waste by $\sim$90%

Summary

- Motivating problem:
  - online prediction of a cold rolling mill under model uncertainty
  - severe computational constraints in real-time setting
- Dynamic model averaging (DMA):
  - Model indicator and model parameters evolve in time (hidden Markov model)
  - Recursive implementation is computationally efficient
  - Equivalent to an exponentially age-discounted version of static BMA
- Gave better results than a single model for online prediction of a cold rolling mill, and could reduce waste by $\sim$90%
- Using all possible combinations of variables gave better results than a smaller set of physically motivated models