Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

Vilde Brady

Bayesian ensemble filtering for a model with categorical and continuous variables

Master's thesis in Industrial Mathematics
Supervisor: Håkon Tjelmeland
June 2021

NTNU
Norwegian University of Science and Technology

Vilde Brady

Bayesian ensemble filtering for a model with categorical and continuous variables

Master's thesis in Industrial Mathematics
Supervisor: Håkon Tjelmeland
June 2021

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

NTNU
Norwegian University of
Science and Technology

Preface

This report presents the final results of the Master's thesis related to the course TMA4900. The report is the final work in the Master's degree program, Industrial Mathematics, at the Norwegian University of Science and Technology, NTNU. It is a continuation of the work done in the course TMA4500, where we decide on a specialization topic and investigate theory and methods related to this topic. This thesis applies the Bayesian ensemble filtering method to an assumed model with both categorical and continuous variables. A professor at NTNU has guided the project closely. I want to give a special thanks to professor Håkon Tjelmeland at the Department of Mathematical Sciences for the thorough and constructive guidance this semester.

Abstract

The Bayesian ensemble filter is proposed in Loe and Tjelmeland (2021). It is a generalization of the traditional ensemble Kalman filter (EnKF), a solution to the filtering problem in statistics. The ensemble Kalman filter updates a prior ensemble to a posterior ensemble, where the updated ensemble consists of realizations from the desired filtering distribution. The Bayesian ensemble filter opens up for other parametric families than the Gaussian, which are the assumptions for the popular ensemble Kalman filter. The framework they present is what they call fully Bayesian, in the sense that we treat the model parameters as random variables. The first task of updating a prior ensemble to a posterior ensemble is to derive the distribution from which to simulate the unknown parameters. The next step is to find a class of updating distributions from which to sample the posterior ensemble. An optimality criterion is thereafter stated, restricting the class of updating distribution to the optimal distribution with respect to the criterion.

In Loe and Tjelmeland (2021), they study two assumed models, one with continuous state variables and one with categorical state variables. In this report, we apply the Bayesian ensemble filter on a new assumed model. The new assumed model has state variables consisting of one categorical and one continuous vector. The categorical state vector follows a Markov chain with non-homogeneous transition probabilities. The continuous state vector is assumed Gaussian and contributes to the model as noise. The optimality criterion in our assumed model states that we want to minimize the expected Mahalanobis distance between the prior and posterior ensemble members. Finding the updating distributions then reduces to two optimality problems, one for each state vector. To show the algorithm's performance, we present three simulation examples with different degrees of the variance's influence of the observations. We compare the results from the simulation to the generated true values. The filter captures the true value of the categorical vector well in two of the examples, which is expected. However, the spread of the simulated filtering ensemble members might be too narrow.

Sammendrag

Det Bayesianske ensemble filteret er presentert i Loe og Tjelmeland (2021). Det er en generalisering av det tradisjonelle ensemble Kalmanfilteret (EnKF) og er en løsning på filtreringsproblemet i statistikk. Ensemble Kalmanfilteret oppdaterer et prediksjonsensemble til et oppdateringsensemble, hvor det oppdaterte ensemblet består av realisasjoner fra den ønskede filtreringsfordelingen. Det Bayesianske ensemble-filteret åpner opp for andre antagelser enn Gaussiske, som er antagelsene for ensemble Kalmanfilteret. Det Bayesianske ensemble filter algoritmen er fullt Gaussisk, i den forstand at den betrakter alle parametre som tilfeldige variabler. Det første steget i den Bayesianske ensemble filteret er å finne en måte å simulere de ukjente parameterne på. Det neste steget er så å finne ett sett med fordelinger som oppdateringsensemblet blir simulert fra. Deretter vil et optimeringskriterium bestemme hvilken fordeling som er den best mulige, i henhold til kriteriet.

To modeller blir studert i Loe og Tjelmeland (2021). Den ene modellen betrakter kontinuerlige tilstandsvariable og den andre modellen kategoriske tilstandsvariable. I denne rapporten vil vi studere en ny antatt modell. Den nye modellen har tilstandsvariable som består av en kategorisk og en kontinuerlig vektor. Den kategoriske vektoren kommer fra en Markovkjede, mens den kontinuerlige vektoren er antatt normalfordelt og bidrar som støy i modellen. Vi velger et optimeringskriterium som sier at vi vil gjøre så få endringer som mulig når vi oppdaterer medlemmer fra prediksjonsensemblet til oppdateringsensemblet. For å finne settet med oppdateringsfordelinger må vi så løse to optimeringsproblemer, ett optimeringsproblem for hver tilstandsvektor. Vi utfører tre simulerte eksempler for å vise hvordan algoritmen presterer på eksempler der de sanne verdiene er kjent. Filtreringsalgoritmen gjør det bra på de to eksemplene den er antatt å gjøre bra på. Det kan tyde på at filteret underestimerer variansen, ettersom det ser at spredningen av de simulerte realisasjonene fra oppdateringsfordelingen ikke ser ut til å dekke de sanne verdiene.

Contents

1 Introduction

A state space model consists of evolving latent states, $\{x^t\}_{t=1}^T$ and related observations, $\{y^t\}_{t=1}^T$, where T is the number of time steps in the model. We say that each latent state variable x^t emits one observation y^t and makes a transition to a new latent state variable x^{t+1}, at each time t. The latent states constitute a Markov chain, and the observations depend on the latent state in their respective time. The observation at one time is conditionally independent of the previous latent state variables given its respective state variable at that specific time. The filtering problem in statistics refers to making sequential inference in state space models. More specifically, it takes on finding the distribution of the current latent state variable, given all the observations up until the current time, $p(x^t|y^1, y^2, \ldots, y^t)$. In other words, it helps us learn as much as possible about the latent states together with the associated uncertainties of the estimations made through the observations in the state space model.

The filtering problem arises in many fields, and some notable examples are weather forecasting and reservoir modeling. Numerical models for describing the atmosphere's evolution are widely used in weather forecasts, such as wind speed prediction, see Cassola and Burlando (2012). Atmospheric observations are often of low resolution and come with measurement errors. Solving the filtering problem can, in this case, make it possible to estimate the wind speed in some interpolated fields using the numerical model of the wind speed evolution, while also taking the observations with measurement errors into account. In reservoir modeling, a relevant problem might be to estimate some properties of the petroleum present in a reservoir, such as the amount or the saturation. The property measured might depend on some underlying states in the reservoir, such as porosity, rock type and pressure, corresponding to the state space model's latent variables. The observation could be some measurement of the hidden state of the reservoir, which would have some measurement errors.

The Kalman filter provides an exact solution to the filtering problem, assuming linear transitions and emission equations with additive Gaussian noise. The Kalman filter starts with a prediction step. This step updates the model parameters by applying a linear shift from the states' transition equations. Following the prediction step is the update step. The update step combines the prediction distribution with the distribution of the observations given the current state variable to find the desired filtering distribution. This way, the current filtering distribution is based on the previous state variable and the observation made in the current iteration. With linear Gaussian assumptions to the transitions and emissions, the update step is a linear

update of the mean vector and covariance matrix from the prediction step. Since we operate with linear Gaussian assumptions in all steps of the model, the update step returns a Gaussian filtering distribution. In high dimensional state spaces, the computation and storage of the model parameters often become computationally expensive, hence approximations are needed.

The ensemble Kalman filter (EnKF) presented by Evensen (1994) finds an approximate solution to the filtering problem. In contrast to the Kalman filter, which propagates distributions in all steps, the EnKF instead propagates an ensemble of independent realizations from the desired filtering distributions. One advantage of working with realizations and not with distributions directly is that we can easily reduce the problem's dimension and the updating algorithm's complexity. Therefore, the EnKF is essential in, for example, weather forecasting, where the dimension of the state space tends to be high. The EnKF also consists of a prediction and an update step. The prediction step applies a dynamical model to the ensemble from the filtering distribution at time $t-1$, resulting in a prediction ensemble of independent realizations from the prediction distribution at time t. The prediction step is assumed doable in an exact manner. Like the traditional Kalman filter, the update step involves a linear shift of the prediction ensemble to obtain an ensemble from the filtering distribution. The prediction ensemble might not be realizations from a Gaussian distribution, and the EnKF might retain non-Gaussian properties of the correct underlying distribution.

One problem of the EnKF is that it often underestimates the uncertainties in its estimations. In the EnKF, we first estimate the model parameters. When used in the model, we assume them to be known and do not include the uncertainties in the estimations. The hierarchical EnKF (HEnKF) introduced by Myrseth and Omre (2010) aims at making more reliable uncertainty estimates by treating the model parameters as random variables, with a Gauss-conjugate density function. The HEnKF seems to give more reliable predictions, see Myrseth and Omre (2010).

The EnKF implicitly assumes a linear Gaussian model for the prediction distribution when combined with the linear Gaussian likelihood model. Loe and Tjelmeland (2020) introduce a more general class of updating distributions. They open up for other distribution families than the Gaussian, for instance, one supporting categorical variables. Other assumptions for the model distributions can, for instance, be useful in the reservoir modeling example, where the latent variable might depend on the type of rock in the reservoir. A discrete distribution for the state variables might then be useful. From the assumed prediction distribution and the assumed likelihood, we can use Bayes' rule to find the distribution from which to simulate the final filtering ensemble at each time t. In Loe and Tjelmeland (2021), this Bayesian

framework is taken one step further to what they call a fully Bayesian framework for the filtering problem in statistics. The framework is fully Bayesian in the sense that, in similarity with the HEnKF, it treats its parameters as random variables. In particular, they discuss two different models. In the first discussed model, the prediction and forecast approximations constitute a linear Gaussian model, and in the second model, they constitute a hidden Markov model with categorical states.

This report applies the fully Bayesian framework defined in Loe and Tjelmeland (2021) to another assumed model, including latent states consisting of both categorical and continuous variables. In particular, the state space's latent variables consist of one categorical vector, whose elements takes values of either 0 or 1, and one continuous vector acting as the noise in the model. The categorical vector is assumed to follow a Markov chain, and the continuous vector is assumed Gaussian distributed with mean zero. This report aims at finding a class of updating distributions from which to simulate realizations from the filtering distribution with this assumed model. Our focus lies on the structure of the updating procedure, and we do not take the complexity of the resulting algorithm into account. As the computation of some of the distributions requires a sum over all possible state vectors, we assume the state space dimension to be sufficiently small in order for the algorithm to be computationally feasible. We leave an algorithm for higher dimensional state spaces for future work.

Section 2 provides the relevant background material for state space models and some known solutions to the filtering problem. In Section 3 the fully Bayesian framework from Loe and Tjelmeland (2021) is presented. It is followed by our assumed model, which is presented in Section 4. Here, the class of updating distributions for our assumed model is derived. In Section 5, we present a simulation example of the algorithm in use. Finally, we finish the report in Section 6 with some closing remarks.

2 Preliminaries

This section introduces state space models and the filtering problem, and provides some relevant background material. We present two well-known solutions to the filtering problem, the traditional Kalman filter and the ensemble Kalman filter. We end the preliminaries with a section on linear programming, a valuable tool in the final updating algorithm in this report and a quick presentation of the VAR(1) model for time series.

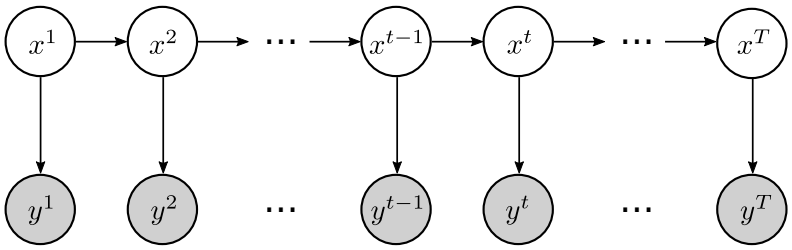

Figure 1: Graphical illustration of a state space model. The shaded variables are observable, whereas the non shaded variables are hidden.

2.1 State space models

A state space model consists of latent random state variables $\{x^t\}_{t=1}^T$, $x^t \in \mathbf{\Omega}_x$, and observations $\{y^t\}_{t=1}^T$, $y^t \in \mathbf{\Omega}_y$, whose value is depending on its corresponding state variable at time t. The dependency structure of a general state space model is shown in Figure 1. The latent states constitute a Markov chain, and the Markov property is that each variable x^t is conditionally independent of $x^{t-2}, x^{t-3}, \ldots, x^1$, given its parent x^{t-1}. We use the notation $x^{1:T} = (x^1, \ldots, x^T)$, to denote all the underlying variables from time 1 to time T. The joint distribution of all the latent state variables then factorizes into a product of the initial state distribution $p(x^1)$ and the transition probabilities $p(x^t|x^{t-1})$ as,

$$p(x^{1:T}) = p(x^1) \prod_{t=2}^{T} p(x^t|x^{t-1}).$$

The observed random variable y^t results from the latent state x^t emitting a random value. The state space model stipulates that the observed random variables are independent given the latent state variables. In other words, we can write that $y^{1:T} = (y^1, y^2, \ldots, y^T)$ are conditionally independent given $x^{1:T}$. The joint distribution of the observations conditioned on latent states then factorizes into the product,

$$p(y^{1:T}|x^{1:T}) = \prod_{t=1}^{T} p(y^t|x^t).$$

For convenience, we define $p(x^{T+1}|x^T) = 1$, so that we can write the joint distribution of all variables in the state space model as,

$$p(x^{1:T}, y^{1:T}) = p(x^1) \prod_{t=1}^{T} p(x^{t+1}|x^t) p(y^t|x^t). \tag{1}$$

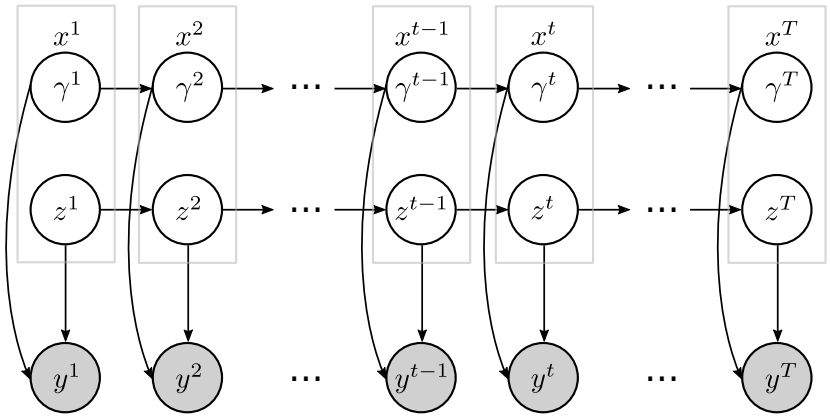

Figure 2: Graphical illustration of a state space model with both categorical variables γ^t, and continuous variables z^t. The shaded variables are observable, whereas the non-shaded variables are hidden. The squares surround the latent variables in one time step.

The state space model is general, and countless models have this underlying structure. The latent states can be high dimensional and coming from both continuous and categorical state spaces $\mathbf{\Omega}_x$. The same yields for the observation space $\mathbf{\Omega}_y$. To illustrate the state space model's generality, described in this section, we present an example. One can, for instance, have a state space $\mathbf{\Omega}_x = \mathbf{\Omega}_\gamma \times \mathbf{\Omega}_z$ of a categorical space $\mathbf{\Omega}_\gamma$ and a continuous space $\mathbf{\Omega}_z$. The latent states in this state space model, $x^t = (\gamma^t, z^t)$, $t = 1, \ldots, T$, consists of $\gamma_t \in \mathbf{\Omega}_\gamma$ and $z^t \in \mathbf{\Omega}_z$. We assume that $\{\gamma^t\}_{t=1}^T$ follows a Markov chain. The continuous state vector z^t could contribute to the model as noise, and the dependency between the variables satisfies the Markov property, meaning $p(z^t|z^{t-1}, z^{t-2}, \ldots, z^1) = p(z^t|z^{t-1})$ for all $t = 1, \ldots, T$. Further, we assume that $\gamma^t \perp z^t$ for all $t = 1, \ldots, T$ and that emission of the observable variables in this model depend on the entire state vector in their respective time. An illustration of this specific state space model is given in Figure 2. Now define $p(\gamma^{T+1}|\gamma^T) = 1$ and $p(z^{T+1}|z^T) = 1$. The same independency structure as in Figure 1 yields in this model. Now we can factorize the expression in (1) further by using that the $\gamma^t \perp z^t$. The initial probability then comes down to $p(x^1) = p(\gamma^1)p(z^1)$ and the transition probabilities $p(x^{t+1}|x^t) = p(\gamma^{t+1}|\gamma^t)p(z^{t+1}|z^t)$, so that the joint distribution of all the variables in this particular state space model becomes,

$$p(x^{1:T}, y^{1:T}) = p(\gamma^1)p(z^1)\prod_{t=1}^{T} p(\gamma^{t+1}|\gamma^t)p(z^{t+1}|z^t)p(y^t|\gamma^t, z^t), \tag{2}$$

We could, for instance, use this model in reservoir modeling. The state vector could represent the amount or the saturation of petroleum present at one specific location or some other property we want to investigate. We can assume that this property could depend on what type of rock is there, representing the categorical vector. The continuous vector acts as the noise not captured in the emission elsewhere. The observations y^t could then be some measurement of the state vector, bound to have some measurement errors and uncertainties. This model will be our assumed model in this report, described in Section 4. In Section 4, we will find a class of updating distributions from which to simulate realizations from the distribution $p(x^t|y^{1:t})$, i.e. the filtering distribution.

2.2 The filtering problem

We now go back to the general state space model with joint distribution defined in (1). In state space models we observe some variables $\{y^t\}_{t=1}^T \in \mathbf{\Omega}_y$, but are interested in some hidden states $\{x^t\}_{t=1}^T \in \mathbf{\Omega}_x$. The filtering problem in statistics takes on finding the hidden state's distribution at time t, x^t, given all observations made up to time t. We are thereby interested in the filtering distribution $p(x^t|y^{1:t})$. The filtering distribution can be found iteratively, where we in each iteration assume that the filtering distribution, $p(x^{t-1}|y^{1:t-1})$, in the previous time $t-1$ is known. Each iteration consists of two steps, the prediction- and update steps. In the prediction step we find the so-called prediction distribution $p(x^t|y^{1:t-1})$, by marginalizing over x^{t-1} from the joint distribution $p(x^t, x^{t-1}|y^{1:t-1})$ and using that $x^t \perp y^{1:t-1}|x^{t-1}$,

$$\begin{aligned} p(x^t|y^{1:(t-1)}) &= \int_{\mathbf{\Omega_x}} p(x^t, x^{t-1}|y^{1:(t-1)})dx^{t-1} \\ &= \int_{\mathbf{\Omega_x}} p(x^t|x^{t-1})p(x^{t-1}|y^{1:(t-1)})dx^{t-1}. \end{aligned} \tag{3}$$

Following the prediction step comes the update step, where we find the filtering distribution at time t. The update step uses Bayes' rule to compute the filtering distribution. The prediction distribution serves as a prior distribution for the filtering distribution, meaning it is our best guess for a filtering distribution without taking the evidence, being the current observation, into account. With a likelihood $p(y^t|x^t)$, the filtering distribution comes out as the posterior distribution in Bayes' rule,

$$\begin{aligned} p(x^t|y^{1:t}) &= p(x^t|y^{1:(t-1)}, y^t) \\ &= \frac{p(y^t|x^t)p(x^t|y^{1:(t-1)})}{\int_{\mathbf{\Omega_x}} p(y^t|x^t)p(x^t|y^{1:(t-1)})dx^t}. \end{aligned} \tag{4}$$

Both the marginalization in (3) and the calculation of the normalizing constant in (4) require an integral over the entire state space, and the calculations are generally not feasible. However, assuming a linear Gaussian model for the prediction and the likelihood distribution makes the joint, all marginals and all conditionals also Gaussian. The algorithm with these assumptions is called the Kalman filter, and we present it in the next section.

2.3 The Kalman filter

The Kalman filter (Katzfuss et al., 2016) is a framework for finding the filtering distribution, that is $p(x^t|y^{1:t})$, given a linear Gaussian state space model. We start by assuming a Gaussian initial distribution for x^1 with initial mean vector μ_0 and covariance matrix Σ_0, i.e. $x^1 \sim \mathcal{N}(\mu_0, \Sigma_0)$. A linear Gaussian state space model assumes linear Gaussian equations for all transitions between the hidden states and all emissions from the hidden states to the model's observations. We write the transition equation in the state space model at time t as,

$$x^t = Fx^{t-1} + u^t, \tag{5}$$

with $u^t \sim \mathcal{N}(0, Q)$, and the matrix F takes the state one step forward at time. We write the linear emission equation as,

$$y^t = Hx^{t-1} + v^t, \tag{6}$$

where $v^t \sim \mathcal{N}(0, R)$, and the matrix H relates the observation at time t to its respective hidden state. As described in Section 2.2, the Kalman filter solves the filtering problem iteratively with an algorithm of a so-called prediction step and an update step. With linear Gaussian transitions and emissions, the prior, likelihood and posterior distribution in (4) will also return Gaussian. In other words, the prediction and filtering distribution will be Gaussian. For the Gaussian distribution, we only need the mean and covariance matrix to identify the distribution. Thus, to update the distribution in the prediction steps and the updating steps, the Kalman filter only updates the mean and covariance matrix. We denote with $\check{\mu}^t \in \mathbb{R}^n$ and $\check{\Sigma}^t \in \mathbb{R}^{n\times n}$ the mean vector and the covariance matrix of the prediction distribution at time t. The notation $\tilde{\mu}^t \in \mathbb{R}^n$ and $\tilde{\Sigma}^t \in \mathbb{R}^{n\times n}$ denote the mean vector and the covariance matrix for the filtering distribution at time t. The same notation yields for the following section. The Kalman filter first calculates the parameters for the prediction distribution, $x^t|y^{1:t-1} \sim \mathcal{N}(\check{\mu}^t, \check{\Sigma}^t)$. With the filtering distribution at time

$t-1$ as, $x^{t-1}|y^{1:t-1} \sim \mathcal{N}(\mu^{t-1}, \Sigma^{t-1})$, and using the transition in (5) we get,

$$\begin{aligned}\check{\mu}^t &= \mathrm{E}(x^t|y^{1:t-1}) = F\mu^{t-1},\\ \check{\Sigma}^t &= \mathrm{Var}(x^t|y^{1:t-1}) = F\Sigma^{t-1}F^T + Q.\end{aligned}$$

Here the notation A^T denotes the transpose of a matrix A. The prediction distribution is the prior distribution in (4), and the associated likelihood distribution can be found by the emission in (6) to be,

$$y^t|x^t \sim \mathcal{N}(Hx^t, R). \tag{7}$$

The prior and likelihood distributions produces the posterior distribution or the filtering distribution at time t, which can be found by combing Gaussian distributions to be,

$$x^t|y^{1:t} \sim \mathcal{N}(\tilde{\mu}^t, \tilde{\Sigma}^t),$$

where

$$\begin{aligned}\tilde{\mu}^t &= \check{\mu}^t + K^t(y^t - H\check{\mu}^t),\\ \tilde{\Sigma}^t &= (I - K^tH)\check{\Sigma}^t\end{aligned} \tag{8}$$

and $K^t = \check{\Sigma}^t H^T(H\check{\Sigma}^t H^T + R)^{-1}$ is the so-called Kalman gain matrix.

The Kalman filter calculates an analytical filtering distribution given a linear Gaussian state space model. If the underlying state space model is not linear Gaussian, our assumptions are wrong, and another model might be preferred. For high dimensional state spaces $\boldsymbol{\Omega}_x$ or observation spaces $\boldsymbol{\Omega}_y$, the rank of the parameters in the Kalman filter becomes too large to handle computationally. The Kalman filter is then not computationally feasible. The EnKF introduces so-called ensembles, being realizations of either the prediction or filtering distribution. The EnKF does not assume Gaussian distributions in the prediction step, and the resulting filtering ensemble might therefore not come from a Gaussian distribution.

2.4 The ensemble Kalman filter

The ensemble Kalman filter (EnKF), first introduced by Evensen (1994), is an evolution of the regular Kalman filter described in Section 2.3. It solves the filtering problem, where the observations are, assumed to come from a linear Gaussian model as in (6), and with likelihood distribution as in (7). In contrast to the regular Kalman filter, the transitions do not need to be linear Gaussian. It also consists of a prediction and an update step performed at each time in the algorithm. Instead of updating

each step's distribution, it updates a so-called ensemble. The ensemble consists of independent realizations of the distribution in a given step, and we start with an ensemble from the initial distribution $p(x^1)$. Assume an ensemble, $\{x^{t-1(m)}\}_{m=1}^M$, of M realization from the filtering distribution at time $t-1$, $p(x^{t-1}|y^{1:t-1})$, is available. In the prediction step, the EnKF assumes that an ensemble from the prediction distribution can be obtained by simulating from the distribution $p(x^t|x^{t-1})$. More precisely, the EnKF assumes we can obtain an ensemble $\{\check{x}^{t(m)}\}_{m=1}^M$ from the prediction distribution at time t by applying a dynamic model,

$$\check{x}^{t(m)} = f(x^{t-1(m)}) + u^{t(m)},$$

with $u^{t(m)} \sim \mathcal{N}(0, Q)$, to all the ensemble members from the previous time step. This is analogous to the prediction step in the Kalman filter in Section 2.3, where we would have the same model for the prediction distribution in the EnKF as in the Kalman filter choosing $f(x^{t-1(m)}) = Fx^{t-1(m)}$. The prediction step is usually doable in an exact manner. The realizations in the prediction ensemble do not necessarily have to come from a Gaussian distribution, as opposed to the regular Kalman filter. Next is the update step, where we will find a linear equation that updates a prediction realization to a realization from the filtering distribution at time t. To find this equation, we assume that the prediction ensemble, $\{\check{x}^{t(m)}\}_{m=1}^M$ consists of independent realizations from a Gaussian distribution with mean vector $\check{\mu}^t$ and covariance matrix $\check{\Sigma}^t$. Namely, we assume,

$$\check{x}^{t(m)} \sim \mathcal{N}(\check{\mu}^t, \check{\Sigma}^t), \tag{9}$$

for all $m = 1, 2, \ldots, M$. The mean vector and covariance matrix from the assumed prediction distribution are not available. Hence, we estimate the prediction distribution's parameters, $\check{\mu}^t$ and $\check{\Sigma}^t$, from the prediction ensemble at each time t. The likelihood distribution for ensemble member m is the Gaussian distribution in (7) replacing x^t with the respective ensemble member,

$$y^t|\check{x}^{t(m)} \sim \mathcal{N}(H\check{x}^{t(m)}, R).$$

Suppose now that $\check{x}^{t(m)}$ is independent of $v^{t(m)} \sim \mathcal{N}(0, R)$ and consider the linear update

$$\tilde{x}^{t(m)} = \check{x}^{t(m)} + K^t(y^t - H\check{x}^{t(m)} + v^{t(m)}), \tag{10}$$

with K^t as in (8). Using that $(H\check{\Sigma}^t H^t + T)^{-1}$ is symmetric, one can show that $\tilde{x}^{t(m)}$ is Gaussian distributed with mean vector $\tilde{\mu}^t$ and covariance matrix $\tilde{\Sigma}^t$ as,

$$\begin{aligned}
\tilde{\mu}^t &= \check{\mu}^t + K^t(y^t - H\check{\mu}^t) \\
\tilde{\Sigma}^t &= \mathrm{Var}(\check{x}^{t(m)}) + \mathrm{Var}\big(K^t(H\check{x}^{t(m)} + v^{t(m)})\big) \\
&\qquad - 2\mathrm{Cov}\big(K^t(H\check{x}^{t(m)} + v^{t(m)}), \check{x}^{t(m)}\big) \\
&= \check{\Sigma}^t + K^t(H\check{\Sigma}^t H^T + R)(H\check{\Sigma}^t H^T + R)^{-1} H\check{\Sigma}^t - 2K^t H\check{\Sigma}^t \\
&= (I - K^t H)\check{\Sigma}^t.
\end{aligned} \tag{11}$$

This leads to that $\tilde{x}^{t(m)} \sim N(\tilde{\mu}^t, \tilde{\Sigma}^t)$, with $\tilde{\mu}^t$ and $\tilde{\Sigma}^t$ as in (8). The updated ensemble member $\tilde{x}^{t(m)}$ is therefore approximately a realization from the filtering distribution at time iteration t. As the number of ensemble members M is much smaller than the state space's dimension, the estimated covariance matrix has a low rank. For problems with higher dimensional state space or observation space, the EnKF yields a more desired algorithm than the regular Kalman filter because it becomes more computationally efficient. In the construction of the linear update equation, we indirectly assume a linear Gaussian model for the true underlying model. Suppose the true underlying model is not linear Gaussian. In that case, we might preserve some of the non-Gaussian properties of the true distribution when updating with (10) and not simulating from the Gaussian with parameters in (11) directly. In Section 3, we present the Bayesian ensemble filtering method, a generalized version of the EnKF. The Bayesian ensemble filtering method opens up for other assumptions than the linear Gaussian model. It does not assume the model parameters to be known but treat them as random variables as opposed to the two filters presented so far. Before diving into the Bayesian ensemble filtering method, we take a quick detour through linear programming and vector autoregression in the following two sections.

2.5 Linear programming

Linear programming (Nocedal and Wright, 2006) is a special case of mathematical optimization. In mathematical optimization, we want to find the optimal solution, $x^\star$, of a problem given some criteria, with respect to some variable x. Linear programming demands all equations and criteria to be linear functions of the variable x. The standard or canonical form of a linear programming problem is,

$$\begin{aligned}
\min_x \quad & g^T x \\
\text{subject to} \quad & Ax = b, \quad x \geq 0,
\end{aligned}$$

where $g, x \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$. The function we minimize over is called the objective function. In this case the objective function is $g^t x$. Any linear programming problem can be described on this form, with some modifications. We can, for instance, have both upper and lower bounds on the variable x, which gives the linear programming problem,

$$\begin{aligned} \min_{x} \quad & g^T x \\ \text{subject to} \quad & A^T x = b, \quad l \leq x \leq u, \end{aligned} \tag{12}$$

with $l, u \in \mathbb{R}^n$, not written on the standard form. All the values of x that satisfy the constraint in the problem defines the feasible set. Linear programming problems do not necessarily have a unique solution. If the feasible set is empty or the objective function is unbounded below the feasible region, i.e., the region that the constraints define, the problem will not have a solution. There are different methods for solving linear programming problems, and two popular choices are the simplex method and the interior-point method. In Section 4.3 we use linear programming, where the optimal solution will be the filtering distribution of a categorical state vector.

2.6 Vector autoregression

In this section we present the vector autoregressive model of order 1, i.e. the VAR(1) model, see Brockwell and Davis (2016). The VAR(1) model can be written as,

$$z^t = \Phi z^{t-1} + \omega^t, \tag{13}$$

for $t = 2, \ldots, T$, where $\Phi \in \mathbb{R}^{n \times n}$ and $\omega^t \sim \mathcal{N}(0, \Upsilon)$. A stationary time series is a time series where the probabilistic character of the time series does not change over time. More precisely, if the joint distribution of $(z_1, \ldots, z_m)$ and the joint distribution $(z_{1+h}, \ldots, z_{m+h})$ are the same for all $h \in \mathbb{N}$ and $m > 0$. For the VAR(1) model to be stationary we need that all eigenvalues of the matrix Φ lies inside the unit circle. The VAR(1) model will be used as a model for the continuous latent variables in the simulation examples in Section 5.

3 Bayesian ensemble filtering

The Bayesian ensemble filtering method introduced in Loe and Tjelmeland (2021) finds the filtering distribution in state space models with what they call a fully Bayesian updating framework. The framework is fully Bayesian in the sense that

the model parameters are treated as random variables. The filtering method can be viewed as a more generalized updating procedure than the EnKF, which is a particular case given the correct assumptions. The framework takes some elements from the EnKF, in which it also propagates ensembles in each iteration instead of distributions as in the regular Kalman filter. It also uses a prediction and an update step to solve the filtering problem. The Bayesian filtering method's prediction step does not differ from the previously introduced methods. For that reason, our focus lies on the updating of an ensemble of realizations, $\{\check{x}^{(m)}\}_{m=1}^{M}$, from the prediction distribution to an ensemble of realizations, $\{\tilde{x}^{(m)}\}_{m=1}^{M}$, from the filtering distribution at time t. Since the parameters are random variables, they also need to be updated before updating the prediction ensemble. The parameter simulation is explained further in Section 3.3. The superscript t is omitted from the notation in this and the following section, as we only focus on one specific time t. As before, we write variables x from the prediction distribution at time t as $\check{x}$, and variables x from the filtering distribution as $\tilde{x}$ at time t, to easily distinguish between variables from the prediction distribution and the filtering distribution.

3.1 Assumed Bayesian model

Here we present the assumed Bayesian model in the Bayesian ensemble filtering method, following Loe and Tjelmeland (2021). The model will be used to update a realization $\check{x}^{(m)}$ from the prediction distribution to a realization $\tilde{x}^{(m)}$ from the filtering distribution. Figure 3 illustrates the updating step's Bayesian model. Here, we denote with $\theta \in \mathbf{\Omega}_\theta$, the parameters defining the distributions of the hidden state, x, at the current time, where $\mathbf{\Omega}_\theta$ is the parameter space. We define $u_{(m)} = \{\check{x}^{(1)}, \check{x}^{(2)}, \ldots, \check{x}^{(m-1)}, \check{x}^{(m+1)}, \ldots, \check{x}^{(M)}\}$, in other words all the realizations from the prediction distribution omitting realization m. We assume the true underlying hidden state x and the realizations $\check{x}^{(1)}, \ldots, \check{x}^{(M)}$ to be conditionally independent and identically distributed given θ. Additionally, we assume that $y \perp \{\check{x}^{(m)}\}_{m=1}^{M}, \theta | x$ and that $\tilde{x}^{(m)} \perp u_{(m)} | \theta, y$. The joint distribution of $\theta, x, \check{x}^{(1)}, \ldots, \check{x}^{(M)}$ and y then simplifies to

$$p(\theta, x, \check{x}^{(1)}, \ldots, \check{x}^{(M)}, y) = p(\theta)p(x|\theta)p(y|x) \prod_{m=1}^{M} p(\check{x}^{(m)}|\theta).$$

In this equation, we have introduced the prior distribution $p(\theta)$ for θ. This prior should be chosen as a conjugate prior distribution for the distribution $p(x|\theta)$. In Section 3.3 we will derive an algorithm from which to simulate $\theta \sim p(\theta|u_{(m)}, y)$. Before that, we will in Section 3.2 describe a method for finding the class of updating

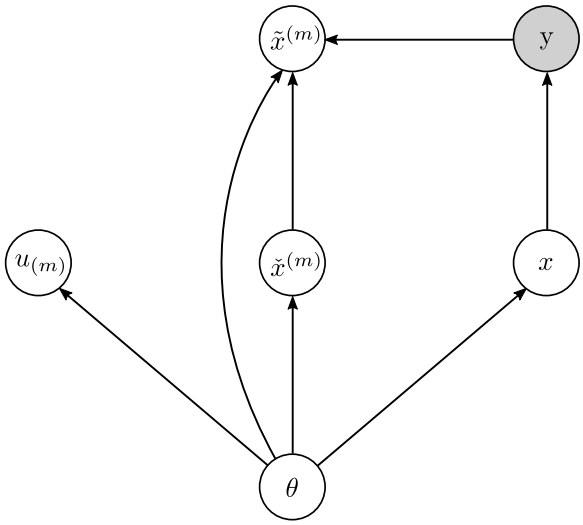

Figure 3: Graphical illustration of the dependency structure in the update of one ensemble member in the Bayesian ensemble filtering method. The superscript t is omitted in all the variables. The shaded nodes are directly observable, whereas the non-shaded nodes are hidden. This figure is inspired by Figure 2 in Loe and Tjelmeland (2021).

distributions, which is used to update an ensemble of prior realizations from the prediction distribution to an ensemble of posterior realizations from the filtering distribution. Here we also state a possible optimality criterion and find the optimal filter with respect to that criterion.

3.2 Simulation from the filtering distribution

After having obtained an ensemble from the prediction distribution, we want to find a class of updating distributions from which to simulate an ensemble from the filtering distribution. According to the assumed model in the previous section, we will update $\check{x}^{(m)}$ by simulating $\tilde{x}^{(m)}$ from some distribution,

$$\tilde{x}^{(m)}|\check{x}^{(m)}, \theta, y \sim p(\tilde{x}^{(m)}|\check{x}^{(m)}, \theta, y),$$

for each $m = 1, \ldots, M$. We want that the simulated $\tilde{x}^{(m)}$ resembles the true hidden value in distribution to the highest degree possible based on the information we have. In other words, we want that the sampling distribution satisfy

$$\tilde{x}^{(m)}|u_{(m)}, y \stackrel{\mathrm{d}}{=} x|u_{(m)}, y. \tag{14}$$

Note that we do not condition on $\check{x}^{(m)}$. This we do not do to preserve the randomness in $\check{x}^{(m)}$ in the updating of $\tilde{x}^{(m)}$. The distribution of the variable on the right hand side of (14) can be written as the marginalization of θ over the joint distribution of $x, \theta | u_{(m)}, y$ like,

$$\begin{aligned} p(x|u_{(m)}, y) &= \int_{\mathbf{\Omega}_\theta} p(x, \theta|u_{(m)}, y) d\theta \\ &= \int_{\mathbf{\Omega}_\theta} p(x|\theta, y) p(\theta|u_{(m)}, y) d\theta, \end{aligned} \tag{15}$$

using that $x \perp u_{(m)}|\theta, y$. The same procedure can be done for the distribution of $\tilde{x}^{(m)}|u_{(m)}, y$ on the left hand side of (14). As $\tilde{x}^{(m)} \perp u_{(m)}|\theta, y$, we obtain that,

$$\begin{aligned} p(\tilde{x}^{(m)}|u_{(m)}, y) &= \int_{\mathbf{\Omega}_\theta} p(\tilde{x}^{(m)}, \theta|u_{(m)}, y) d\theta \\ &= \int_{\mathbf{\Omega}_\theta} p(\tilde{x}^{(m)}|\theta, y) p(\theta|u_{(m)}, y) d\theta. \end{aligned} \tag{16}$$

We exploit that (15) and (16) holds for all x and $\tilde{x}^{(m)}$, to find that to satisfy the constraint in (14), it is sufficient to restrict the updating distribution to satisfy,

$$\tilde{x}^{(m)}|\theta, y \overset{\mathrm{d}}{=} x|\theta, y, \tag{17}$$

for all θ and y. Given a θ, we can update $\check{x}^{(m)}$ by simulating $\tilde{x}^{(m)}$ from a distribution $p(\tilde{x}^{(m)}|\check{x}^{(m)}, \theta, y)$ consistent with (17). To find the parameters θ, we can before the updating of $\check{x}^{(m)}$, simulate $\theta|u_{(m)}, y \sim p(\theta|u_{(m)}, y)$. This process will be described in Section 3.3. By writing the distribution on the left hand side of (17) as the marginalization of $\check{x}^{(m)}$ over the joint distribution of $\tilde{x}^{(m)}, \check{x}^{(m)}|\theta, y$ we have,

$$p_{\tilde{x}^{(m)}|\theta,y}(\tilde{x}^{(m)}|\theta, y) = \int_{\mathbf{\Omega}_\mathbf{x}} p_{x|\theta}(\check{x}^{(m)}|\theta, y) p_{\tilde{x}^{(m)}|\check{x}^{(m)},\theta,y}(\tilde{x}^{(m)}|\check{x}^{(m)}, \theta, y) d\check{x}^{(m)},$$

where we use subscripts on p to make it clear what distributions the various p's refer to. As this must be equal to the distribution of the right hand side of (17), we can rewrite our restriction of the desired sampling distribution to be,

$$p_{x|\theta,y}(\tilde{x}^{(m)}|\theta, y) = \int_{\mathbf{\Omega}_\mathbf{x}} p_{x|\theta}(\check{x}^{(m)}|\theta, y) p_{\tilde{x}^{(m)}|\check{x}^{(m)},\theta,y}(\tilde{x}^{(m)}|\check{x}^{(m)}, \theta, y) d\check{x}^{(m)}. \tag{18}$$

This restriction tells us that there might be infinitely many solutions of updating distributions, $p(\tilde{x}^{(m)}|\check{x}^{(m)}, \theta, y)$, which satisfies (18). Which solution we choose does

not matter if the assumed model is correct. However, if the assumed model is wrong, the choice of solution might have a great impact on the final result. We want to use the information in $\check{x}^{(m)}$ in the updating of the ensemble, and it is therefore essential that the solution depend on this variable. Generally we define an optimal solution $p^\star(\tilde{x}^{(m)}|\check{x}^{(m)}, \theta, y)$ to the class of distributions satisfying (17) as the solution which minimizes the expected value of some function $g(\check{x}^{(m)}, \tilde{x}^{(m)})$

$$p^\star(\tilde{x}^{(m)}|\check{x}^{(m)}, \theta, y) = \underset{q(\cdot)}{\operatorname{argmin}} \, \mathrm{E}\Big[g(\check{x}^{(m)}, \tilde{x}^{(m)})\Big]. \tag{19}$$

One natural choice is to choose $g(\check{x}^{(m)}, \tilde{x}^{(m)})$ to be the Mahalanobis distance between $\check{x}^{(m)}$ and $\tilde{x}^{(m)}$

$$g(\check{x}^{(m)}, \tilde{x}^{(m)}) = (\tilde{x}^{(m)} - \check{x}^{(m)})^T \Sigma^{-1} (\tilde{x}^{(m)} - \check{x}^{(m)}), \tag{20}$$

where $\Sigma \in \mathbb{R}^{n \times n}$ is some positive definite matrix. With this choice of $g(\check{x}^{(m)}, \tilde{x}^{(m)})$, we keep as much of the information from the prior sample, $\check{x}^{(m)}$, as possible by making minimal changes in the update to $\tilde{x}^{(m)}$. In Section 4, we apply the Bayesian filtering method described in this section on a new assumed model, with both discrete and continuous variables. When discussing the update step based on the new assumed model, our focus is on the structure of the updating procedure and not on computational speed. The resulting algorithm is therefore computational feasible for low dimensional state vectors only.

3.3 Parameter simulation

Before the updating of $\check{x}^{(m)}$ described in Section 3.2, we first need to simulate $\theta \sim p(\theta|u_{(m)}, y)$. The parameter simulation is the first step in the update step, and is done for each ensemble member at each time. The prior distribution $p(\theta)$ determines the posterior distribution $p(\theta|u_{(m)}, y)$. To simulate θ it is easier to simulate from the distribution $p(\theta, x|u_{(m)}, y)$ and ignore x. Introducing a Gibbs sampler, we can simulate from $p(\theta, x|u_{(m)}, y)$ by alternating between simulating from from the full conditionals $p(\theta|x, u_{(m)}, y)$ and $p(x|\theta, u_{(m)}, y)$. By Bayes' rule, and exploiting that $\theta \perp y|x$ and that $x \perp u_{(m)}|\theta$, we get that the first full conditional becomes,

$$\begin{aligned} p(\theta|x, u_{(m)}, y) &= p(\theta|x, u_{(m)}) \\ &\propto p(x|\theta, u_{(m)}) p(\theta|u_{(m)}) \\ &\propto p(x|\theta) p(u_{(m)}|\theta) p(\theta). \end{aligned} \tag{21}$$

In Section 3.1 we stated that the distribution $p(\theta)$ should be chosen as a conjugate prior for $p(x|\theta)$. If this is done, the distribution $p(\theta|x, u_{(m)})$ belongs to the same

family of distributions as $p(\theta)$. Finding $p(\theta|x, u_{(m)})$ should therefore be possible. For the other full conditional in the Gibbs sampler we also make use of Bayes' rule, and that $x \perp u_{(m)}|\theta$ and $y \perp \theta|x$ to get that

$$\begin{aligned} p(x|\theta, u_{(m)}, y) &= p(x|\theta, y) \\ &\propto p(y|x, \theta)p(x|\theta) \\ &\propto p(y|x)p(x|\theta). \end{aligned} \tag{22}$$

Performing enough iteration, a θ simulated from the Gibbs sampler should essentially be a sample from the distribution $p(\theta|u_{(m)}, y)$ as desired. After having done the simulation of θ, the class of updating distributions can be found using the previous section's procedure. In the next section, we take a deep dive into the model by introducing a new assumed model, with both continuous and categorical variables, where all calculations needed for the updating procedure are derived. The parameter simulation in this assumed model is described with all necessary calculations in Section 4.2.

4 Our assumed model

Loe and Tjelmeland (2020) applied the Bayesian ensemble filtering framework on two assumed models; one with a continuous state space and one with a categorical state space. In this section, we apply the updating procedure described in Section 3 on a new assumed model, including both continuous and categorical variables. The new assumed model is a state space model identical to the example in Section 2.1, with joint distribution presented in (2). To simplify, we have not assumed that all parameters are unknown. The unknown parameters θ are limited to the Markov chain parameters generating the categorical vectors in the hidden states. The following section will present all assumptions in the new model. This will be followed by the simulation of the parameters θ in Section 4.2. In Section 4.3 we will derive the class of updating distribution in our new assumed model and finally find the optimal sampling distribution from which to simulate realizations from the filtering distribution.

4.1 Specifying the assumed distributions

In our assumed model, we consider a state space model with hidden state variable $x = (\gamma, z) \in \{0, 1\}^n \times \mathbb{R}^n$. Here $\gamma \in \{0, 1\}^n = \mathbf{\Omega}_\gamma$ is a categorical vector only taking values of 0 or 1, and $z \in \mathbb{R}^n = \mathbf{\Omega}_\mathbf{z}$ is a continuous vector. For each x we have

a corresponding variable $y \in \mathbb{R}^m$, representing the observation of the latent state variable. We assume that $\gamma \perp z$ and that $z \perp \theta$. Hence, the joint distribution of the state variables given the parameter θ factorizes into the product,

$$p(x|\theta) = p(\gamma, z|\theta) \tag{23}$$
$$= p(\gamma|\theta)p(z) \tag{24}$$

We assume that the distribution of the variables γ constitute a non-homogeneous Markov chain with initial and transition probabilities,

$$\theta = (\theta_1, \{\theta_{r0}\}_{r=2}^n, \{\theta_{r1}\}_{r=2}^n),$$

where $\theta_1, \theta_{r0}, \theta_{r1} \in [0, 1]$ are so that,

$$\begin{aligned}\theta_1 &= p(\gamma_1 = 1),\\ \theta_{r0} &= p(\gamma_r = 1|\gamma_{r-1} = 0),\\ \theta_{r1} &= p(\gamma_r = 1|\gamma_{r-1} = 1),\end{aligned}$$

for $r = 2, \ldots, n$. The non-homogeneity of the Markov chain means that we open up for different transition probabilities in different parts of the Markov chain. The parameters in θ are considered unknown, and treated as random variables. The joint distribution of all the components in γ given θ then amounts to,

$$p(\gamma|\theta) = p(\gamma_1|\theta) \prod_{r=2}^{n} p(\gamma_r|\gamma_{r-1}, \theta). \tag{25}$$

From our definition of θ as the initial and transition probabilities, we have that the probability of the first component in the Markov chain given θ becomes,

$$p(\gamma_1|\theta) = \theta_1^{\gamma_1}(1 - \theta_1)^{1-\gamma_1} = \text{Bern}(\gamma_1; \theta_1). \tag{26}$$

Namely, $\gamma_1|\theta$ is Bernoulli distributed with parameter θ_1. The transition probabilities in the Markov chain will be,

$$p(\gamma_r|\gamma_{r-1}, \theta) = \theta_{r0}^{\gamma_r(1-\gamma_{r-1})}(1 - \theta_{r0})^{(1-\gamma_r)(1-\gamma_{r-1})}\theta_{r1}^{\gamma_r\gamma_{r-1}}(1 - \theta_{r1})^{(1-\gamma_r)\gamma_{r-1}}. \tag{27}$$

As explained in Section 3.1, we adopt a conjugate prior for the unknown parameters θ. Specifically we assume them to be independent and Beta-distributed,

$$\begin{aligned}\theta_1 &\sim \text{Beta}(p_1, q_1)\\ \theta_{r0} &\sim \text{Beta}(p_{r0}, q_{r0})\\ \theta_{r1} &\sim \text{Beta}(p_{r1}, q_{r1}),\end{aligned}$$

for all $r = 2, \ldots, n$, where $p_1, q_1, p_{r0}, q_{r0}, p_{r1}$ and q_{r1} are all known hyperparameters. The joint prior distribution of θ then amounts to,

$$\begin{aligned} p(\theta) &= p(\theta_1) \prod_{r=2}^{n} p(\theta_{r0}) p(\theta_{r1}) \\ &\propto \theta_1^{p_1 - 1}(1-\theta_1)^{q_1 - 1} \prod_{r=2}^{n} \theta_{r0}^{p_{r0}-1}(1-\theta_{r0})^{q_{r0}-1} \theta_{r1}^{p_{r1}-1}(1-\theta_{r1})^{q_{r1}-1}. \end{aligned} \tag{28}$$

The distribution $p(z)$ is chosen to be a multivariate Gaussian distribution with mean 0 and covariance matrix $\sigma^2 I \in \mathbb{R}^{n \times n}$, with $\sigma^2 \in \mathbb{R}$ and I being the identity matrix of size n. Namely,

$$z \sim \mathcal{N}(0, \sigma^2 I). \tag{29}$$

Define so the known parameter $\mu = [\mu_0, \mu_1]^T \in \mathbb{R}^2$, with $\mu_0, \mu_1 \in \mathbb{R}$. The vector $\mu_\gamma \in \{\mu_0, \mu_1\}^n$ is then defined to be,

$$\mu_\gamma = \begin{bmatrix} \mu_{\gamma_1} & \mu_{\gamma_2} & \ldots & \mu_{\gamma_n} \end{bmatrix}^T.$$

Namely, the ith component in the vector μ_γ is either μ_0 or μ_1, depending on whether the ith component of γ is 0 or 1, respectively. The distribution $p(y|x)$ is chosen to be a Gaussian distribution with mean $H(\mu_\gamma + z)$, $H \in \mathbb{R}^{m \times n}$, and covariance matrix $R \in \mathbb{R}^{m \times m}$,

$$y|x \sim \mathcal{N}(H(\mu_\gamma + z), R). \tag{30}$$

With this, the variable z contributes as noise to the mean of y. Figure 4 illustrates the dependency structure in our assumed model graphically, when we want to update an ensemble member $\check{x}^{(m)}$ to an ensemble member $\tilde{x}^{(m)}$ from the filtering distribution. The following section presents the algorithm for updating the unknown parameters θ. The parameter update is the first procedure in the update step. Section 4.3 presents the remaining derivations in the update step, which results in the class of updating distributions from which to simulate $\tilde{\gamma}^{(m)}$ and $\tilde{z}^{(m)}$.

4.2 Parameter simulation in our assumed model

The parameter simulation is the first part of the the updating of $\check{x}^{(m)}$ to $\tilde{x}^{(m)}$. In Section 3.3 we showed that we could simulate $\theta \sim p(\theta, x|u_{(m)}, y)$ and then forget x. We explained that we could simulate from the joint $p(\theta, x|u_{(m)}, y)$ by introducing a Gibbs sampler, where we alternate between simulating from the conditionals $p(\theta|x, u_{(m)}, y)$ and $p(x|\theta, u_{(m)}, y)$. In this section, we will derive the full conditionals

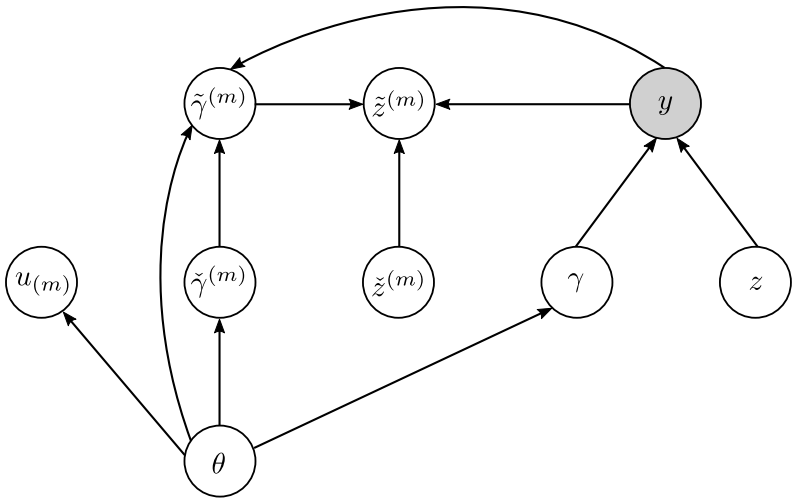

Figure 4: Graph structure of the assumed model when updating an ensemble member $x = [k, z]$ from the prior distribution to $\tilde{x} = [\tilde{k}, \tilde{z}]$ from the posterior distribution. The shaded nodes are observable, whereas the non-shaded are deemed unknown, and need to be simulated.

in the Gibbs sampler in our assumed model with assumptions specified in Section 4.1.

First we will look at how to simulate from the full conditional $p(\theta|x, u_{(m)}, y)$. As explained in Section 3.1, this distribution will belong to the same family of distribution as the prior distribution of θ, $p(\theta)$, as a conjugate prior was chosen for θ. Expanding (21) by using the results in (24) and that the Markov property holds for the categorical state variables, the posterior distribution factorizes into the product

$$\begin{aligned}
p(\theta|x, u_{(m)}, y) &\propto p(\theta)p(x|\theta)p(u_{(m)}|\theta) \\
&= p(\theta)p(z, \gamma|\theta) \prod_{i \neq m} p(\check{z}^{(i)}, \check{\gamma}^{(i)}|\theta) \\
&= p(\theta)p(\gamma|\theta)p(z) \prod_{i \neq m} p(\check{\gamma}^{(i)}|\theta)p(\check{z}^{(i)}) \\
&\propto p(\theta)p(\gamma|\theta) \prod_{i \neq m} p(\check{\gamma}^{(i)}|\theta) \\
&= p(\theta)p(\gamma_1|\theta) \prod_{r=2}^{n} p(\gamma_r|\gamma_{r-1}, \theta) \prod_{i \neq m} \left(p(\check{\gamma}_1^{(i)}|\theta) \prod_{r=2}^{n} p(\check{\gamma}_r^{(i)}|\check{\gamma}_{r-1}^{(i)}, \theta) \right)
\end{aligned} \tag{31}$$

The distribution $p(\theta)$ is written out in (28), and the distribution $p(\gamma|\theta)$ is defined in

(25) with the different factors defined in (26) and (27). Remember that all ensemble members in $\{\gamma^{(m)}\}_{m=1}^{M}$ are assumed to come from the same distribution as γ, given θ. Inserting the expressions found for the distributions $p(\theta)$, $p(\gamma_1|\theta)$ and $p(\gamma_r|\gamma_{r-1},\theta)$ into the last factorization in (31) and pulling together terms of powers with the same base results in that the final expression of the posterior can be factorized into the product,

$$p(\theta|x,u_{(m)}) = p(\theta_1|x,u_{(m)})\prod_{r=2}^{n} p(\theta_{r0}|x,u_{(m)})p(\theta_{r1}|x,u_{(m)}). \tag{32}$$

Here we can recognize the distributions in the factorized product on the right hand side to be the posterior initial distribution,

$$\theta_1|x,u_{(m)} \sim \text{Beta}(p_1^\star, q_1^\star), \tag{33}$$

with parameters

$$\begin{aligned} p_1^\star &= p_1 + \gamma_1 + \sum_{i\neq m}\gamma_1^{(i)}, \\ q_1^\star &= q_1 - \gamma_1 + \sum_{i\neq m}(1-\gamma_1^{(i)}) + 1, \end{aligned}$$

and the posterior transition distributions,

$$\theta_{r0}|x,u_{(m)} \sim \text{Beta}(p_{r0}^\star, q_{r0}^\star), \tag{34}$$

with parameters,

$$\begin{aligned} p_{r0}^\star &= p_{r0} + \gamma_r(1-\gamma_{r-1}) + \sum_{i\neq m}\gamma_r^{(i)}(1-\gamma_{r-1}^{(i)}), \\ q_{r0}^\star &= q_{r0} + (1-\gamma_r)(1-\gamma_{r-1}) + \sum_{i\neq m}(1-\gamma_r^{(i)})(1-\gamma_{r-1}^{(i)}) \end{aligned}$$

and

$$\theta_{r1}|x,u_{(m)} \sim \text{Beta}(p_{r1}^\star, q_{r1}^\star), \tag{35}$$

with parameters,

$$\begin{aligned} p_{r1}^\star &= p_{r1} + \gamma_r\gamma_{r-1} + \sum_{i\neq m}\gamma_r^{(i)}\gamma_{r-1}^{(i)}, \\ q_{r1}^\star &= q_{r1} + (1-\gamma_r)\gamma_{r-1} + \sum_{i\neq m}(1-\gamma_r^{(i)})\gamma_{r-1}^{(i)}. \end{aligned}$$

Having found the full conditional $p(\theta|x, u_{(m)}, y)$, we only need the full conditional $p(x|\theta, u_{(m)}, y)$ to get all the necessary distributions we need to simulate $\theta \sim p(\theta|u_{(m)}, y)$ by the Gibbs sampler. To find this full conditional we start by expanding (22). Using that $z \perp \theta$ and that $\gamma \perp z$, we get,

$$\begin{aligned} p(x|\theta, u_{(m)}, y) &= p(x|\theta, y) \\ &\propto p(y|x)p(x|\theta) \\ &= p(y|\gamma, z)p(\gamma, z|\theta) \\ &= p(y|\gamma, z)p(z)p(\gamma|\theta). \end{aligned} \tag{36}$$

To simulate γ and z from this distribution, we can first simulate $\gamma \sim p(\gamma|\theta, y)$ by marginalizing over z. Given γ, we can thereafter simulate $z \sim p(z|\gamma, \theta, y)$. We start by finding the distribution of $p(z|\gamma, \theta, y)$, which is proportional to the factorized product,

$$\begin{aligned} p(z|\gamma, \theta, y) &= p(z|\gamma, y) \\ &\propto p(z, \gamma, y) \\ &= p(y|\gamma, z)p(\gamma, z) \\ &= p(y|\gamma, z)p(\gamma)p(z) \\ &\propto p(y|\gamma, z)p(z). \end{aligned}$$

By combining the exponents in these Gaussian distributions and introducing the parameters Q and ϕ as,

$$\begin{aligned} Q &= (\frac{1}{\sigma^2}I + H^T R^{-1} H)^{-1} = \sigma^2(I - KH), \\ \phi &= QH^T R^{-1}(y - H\mu_\gamma), \end{aligned} \tag{37}$$

we can show that,

$$p(z|\gamma, \theta, y) \propto \exp\left\{\frac{1}{2}(z - \phi)^T Q^{-1}(z - \phi)\right\}.$$

That is, $p(z|\gamma, \theta, y)$ is Gaussian distributed,

$$p(z|\gamma, \theta, y) \sim \mathcal{N}(\phi, Q), \tag{38}$$

with mean vector ϕ and covariance matrix Q as defined in (37). Next, we look at the marginalized distribution, $p(\gamma|\theta, y)$. By using the result in (36), combining the

exponents in the Gaussian distributions $p(z)$ and $p(y|\gamma, z)$ and integrating out z, $p(\gamma|\theta, y)$ comes down to,

$$\begin{aligned}
p(\gamma|\theta, y) &= \int_z p(x|\theta, y)dz \\
&\propto p(\gamma|\theta) \int_z p(y|\gamma, z)p(z)dz \\
&\propto p(\gamma|\theta) \int_z \exp\Big\{ -\frac{1}{2}\Big[\big(y-H(\mu_\gamma + z)\big)^T \\
&\qquad\qquad \cdot R^{-1}\big(y - H(\mu_\gamma + z)\big) + \frac{1}{\sigma^2} z^T z\Big]\Big\}dz \\
&= p(\gamma|\theta) \exp\Big\{ -\frac{1}{2}\Big[(y - H\mu_\gamma)^T R^{-1}(y - H\mu_\gamma) - \phi^T Q^{-1}\phi\Big]\Big\} \\
&\qquad \cdot \int_z \exp\Big\{ -\frac{1}{2}(z-\phi)^T Q^{-1}(z-\phi)\Big\}dz \\
&\propto p(\gamma|\theta) \exp\Big\{ -\frac{1}{2}\Big[(y - H\mu_\gamma)^T R^{-1}(y - H\mu_\gamma) - \phi^T Q^{-1}\phi\Big]\Big\} \\
&= p(\gamma|\theta)\mathcal{N}(y|H\mu_\gamma, R) \exp\Big\{\frac{1}{2}\phi^T Q^{-1}\phi\Big\}
\end{aligned} \tag{39}$$

We have now calculated all needed distributions in the Gibbs sampler used to simulate $\theta \sim p(\theta|u_{(m)}, y)$ in the beginning of the update step. In the first alternation, we simulate $\theta \sim p(\theta|x, u_{(m)})$ defined in (32), where the factors $p(\theta_1|x, u_{(m)})$, $p(\theta_{r0}|x, u_{(m)})$ and $p(\theta_{r1}|x, u_{(m)})$ are defined in (33), (34) and (35), respectively. In the second alternation, we start by simulating a $\gamma \sim p(\gamma|\theta, y)$ defined in (39). Given this γ, we can then simulate $z \sim p(z|\gamma, \theta, y)$ defined in (38). Performing enough iterations in the Gibbs sampler gives us the parameter used to update the respective ensemble members. The method for updating one realization from the prediction ensemble in our assumed model is described in the following section.

4.3 Class of updating distributions in our assumed model

Having found a way to simulate θ in Section 4.2, we can proceed with the update step described in Section 3.2. We focus on the updating of one specific ensemble member $\check{x}^{(m)} = (\check{\gamma}^{(m)}, \check{z}^{(m)})$, from the prediction distribution, to the ensemble member $\tilde{x}^{(m)} = (\tilde{\gamma}^{(m)}, \tilde{z}^{(m)})$, from the filtering distribution. For the given θ, we find a way to simulate $\tilde{\gamma}^{(m)}, \tilde{z}^{(m)} \sim p(\tilde{\gamma}^{(m)}, \tilde{z}^{(m)}|\check{\gamma}^{(m)}, \check{z}^{(m)}, \theta, y)$ consistent with (17). The restriction in (17)

ca, for our particular case, be written as,

$$\gamma, z|\theta, y \overset{\mathrm{d}}{=} \tilde{\gamma}^{(m)}, \tilde{z}^{(m)}|\theta, y. \tag{40}$$

As we did in Section 3.2, we now denote with $p_X(x^\star)$, the distribution of a variable X evaluated at $x^\star$. Writing out the expression for the joint probability on both sides of the equation, the restriction can be formulated as,

$$p_{\Gamma|\Theta,Y}(\gamma|\theta, y)p_{Z|\Gamma,\Theta,Y}(z|\gamma, \theta, y) = p_{\tilde{\Gamma}^{(m)}|\Theta,Y}(\gamma|\theta, y)p_{\tilde{Z}^{(m)}|\tilde{\Gamma}^{(m)},\Theta,Y}(z|\gamma, \theta, y).$$

From this formulation it is clear that the restriction in (40) is equivalent to the restrictions,

$$p_{\Gamma|\Theta,Y}(\gamma|\theta, y) = p_{\tilde{\Gamma}^{(m)}|\Theta,Y}(\gamma|\theta, y) \tag{41}$$

and

$$p_{Z|\Gamma,\Theta,Y}(z|\gamma, \theta, y) = p_{\tilde{Z}^{(m)}|\tilde{\Gamma}^{(m)},\Theta,Y}(z|\gamma, \theta, y), \tag{42}$$

for all γ, θ and y. Finding a way to simulate $\tilde{\gamma}^{(m)}$ and $\tilde{z}$ consistent with (40), then comes down to finding a way to simulate $\tilde{\gamma}^{(m)}$ from a distribution $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$ consistent with (41). Given this $\tilde{\gamma}^{(m)}$ we find a way to simulate $\tilde{z}^{(m)}$ from a distribution $p(\tilde{z}^{(m)}|\tilde{\gamma}^{(m)}, \check{z}^{(m)}, \theta, y)$ consistent with (42).

We will first look at how to find the updating distribution $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$, from which to simulate $\tilde{\gamma}^{(m)}$. We use that the right hand side of (41) can be written as a sum,

$$p_{\Gamma|\Theta,Y}(\tilde{\gamma}^{(m)}|\theta, y) = \sum_{\check{\gamma}^{(m)}} p_{\tilde{\Gamma}^{(m)}|\check{\Gamma}^{(m)},\Theta,Y}(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)p_{\check{\Gamma}^{(m)}|\Theta}(\check{\gamma}^{(m)}|\theta). \tag{43}$$

The distribution on the left hand side of this equation is calculated in (39), and the distribution $p_{\check{\Gamma}^{(m)}|\Theta}(\check{\gamma}^{(m)}|\theta)$ on the right hand side of the equation is found in (25) with the different factors defined in (26) and (27). The optimal distribution $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$ with respect to an optimality criterion is found in Section 4.4.

We then proceed with finding the updating distribution $p(\tilde{z}^{(m)}|\tilde{\gamma}^{(m)}, \check{z}^{(m)}, \theta, y)$, from which we simulate $\tilde{z}^{(m)}$. We start by noting that the distribution on the left hand side of (42) is the Gaussian defined in (38). We so decide to limit the class of updating distribution to,

$$p(\tilde{z}^{(m)}|\tilde{\gamma}^{(m)}, \check{z}^{(m)}, \theta, y) \sim \mathcal{N}(\tilde{z}^{(m)}|A\check{z}^{(m)} + By + C\mu_{\tilde{\gamma}^{(m)}} + d, S), \tag{44}$$

where $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times n}, C \in \mathbb{R}^{n\times n}, d \in \mathbb{R}^n$ and $S \in \mathbb{R}^{n\times n}$. What is left is to determine A, B, C, d and S making the change from $\check{z}^{(m)}$ to $\tilde{z}^{(m)}$ as small as possible, to

satisfying the optimality criterion, but making sure that the restrictions for the updating distribution are fulfilled. According to (44) we can simulate $\tilde{z}^{(m)}|\tilde{\gamma}^{(m)}, \check{z}^{(m)}, \theta, y$ by drawing an $\epsilon \sim \mathcal{N}(0, S)$ and applying the linear shift of $\check{z}^{(m)}$,

$$\tilde{z}^{(m)} = A\check{z}^{(m)} + By + C\mu_{\tilde{\gamma}^{(m)}} + d + \epsilon. \tag{45}$$

Remember that $\check{z}^{(m)}$ and z are assumed identically distributed, with assumed distribution written in (29). For two Gaussian distributions to be equal, we only nee the mean vectors and covariance matrices to be equal. Hence, the restriction in (42) now comes down to that the mean vectors and covariance matrices of the distribution of $z|\gamma, \theta, y$ and $\tilde{z}^{(m)}|\tilde{\gamma}^{(m)}, \theta, y$ yield the same, when evaluated at the same values. Taking the mean and covariance of both sides of (45) over the distribution $p(\check{z}^{(m)}, \epsilon|\tilde{\gamma}^{(m)}, \theta, y)$ yields,

$$\begin{aligned} \mathrm{E}(\tilde{z}^{(m)}) &= \mathrm{E}(A\check{z}^{(m)} + By + C\mu_{\tilde{\gamma}^{(m)}} + d + \epsilon) \\ &= By + C\mu_{\tilde{\gamma}^{(m)}} + d \end{aligned} \tag{46}$$

and

$$\begin{aligned} \mathrm{Cov}(\tilde{z}^{(m)}) &= \mathrm{Cov}(A\check{z}^{(m)} + By + C\mu_{\tilde{\gamma}^{(m)}} + d + \epsilon) \\ &= \sigma^2 AA^T + S, \end{aligned} \tag{47}$$

respectively. The restriction in (42) is now fulfilled if the result in (46) is equal to ϕ and (47) is equal to Q, with ϕ and Q defined in (37). Namely that,

$$\phi = \sigma^2(I - KH)H^T R^{-1}(y - H\mu_{\tilde{\gamma}^{(m)}}) = By + C\mu_{\tilde{\gamma}^{(m)}} + d \tag{48}$$

and

$$Q = \sigma^2(I - KH) = \sigma^2 AA^T + S, \tag{49}$$

We solve (48) with respect to $By + d$ and obtain,

$$By + d = \sigma^2(I - KH)H^T R^{-1}(y - H\mu_{\tilde{\gamma}^{(m)}}) - C\mu_{\tilde{\gamma}^{(m)}}$$

Inserting this into (45) eliminates B, C and d and results in,

$$\begin{aligned} \tilde{z} &= A\check{z} + C\mu_{\tilde{\gamma}^{(m)}} + \sigma^2(I - KH)H^T R^{-1}(y - H\mu_{\tilde{\gamma}^{(m)}}) - C\mu_{\tilde{\gamma}^{(m)}} + \epsilon \\ &= A\check{z} + \sigma^2(I - KH)H^T R^{-1}(y - H\mu_{\tilde{\gamma}^{(m)}}) + \epsilon. \end{aligned} \tag{50}$$

We then solve (49) with respect to S,

$$S = \sigma^2(I - AA^T - KH). \tag{51}$$

What is now left for obtaining the equation from which we will simulate $\tilde{z}^{(m)}$ from $\check{z}^{(m)}$, is to specify A consistent with the restriction in (42) and making sure that S is positive semi-definite. As shown in Section 3.2, there might be an infinite number of distributions that support these requirements. In Section 4.4, we find the optimal updating distribution for the simulation of both $\tilde{\gamma}^{(m)}$ and $\tilde{z}^{(m)}$, with respect to the optimality criterion in (19) with the Mahalanobis distance defined in (20).

4.4 The optimal solution

The class of updating distributions for simulating $\tilde{\gamma}^{(m)}$ and $\tilde{z}^{(m)}$ from the filtering distributions was found in the previous section. It was found that the distribution $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$ must satisfy (43), and that $\tilde{z}^{(m)}$ can be simulated from the distribution $p(\tilde{z}^{(m)}|\tilde{\gamma}^{(m)}, \check{z}^{(m)}, \theta, y)$, by the linear equation in (50). In this section we will state an optimality criterion, which in turn, determines the distribution $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$ and the matrix A in (50). We use the optimality criterion (19) with the Mahalanobis distance defined in (20) as the objective function. As the state variables contribute to the mean of the likelihood in (30) as the sum $\mu_\gamma + z$, we want to minimize the expected distance between $\mu_{\tilde{\gamma}^{(m)}} + \check{z}^{(m)}$ and $\mu_{\tilde{\gamma}^{(m)}} + \tilde{z}^{(m)}$. We use that Σ^{-1} can be factorized as $\Sigma^{-1} = V^T V$, $V \in \mathbb{R}^{n \times n}$. Hence, we write the objective function for our particular assumed model as,

$$
\begin{aligned}
&\mathrm{E}\left[g(\mu_{\tilde{\gamma}^{(m)}} + \check{z}^{(m)}, \mu_{\tilde{\gamma}^{(m)}} + \tilde{z}^{(m)})\right] \\
&= \mathrm{E}\left[\left((\mu_{\tilde{\gamma}^{(m)}} + \tilde{z}^{(m)}) - (\mu_{\check{\gamma}^{(m)}} + \check{z}^{(m)})\right)^T \Sigma^{-1} \left((\mu_{\tilde{\gamma}^{(m)}} + \tilde{z}^{(m)}) - (\mu_{\check{\gamma}^{(m)}} + \check{z}^{(m)})\right)\right] \\
&= \mathrm{E}\left[\left(V\left((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\right)\right)^T \right. \\
&\qquad\qquad \left. \cdot \left(V\left((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\right)\right)\right].
\end{aligned}
\tag{52}
$$

The expectation is taken over the distribution $p(\tilde{z}^{(m)}, \check{z}^{(m)}, \tilde{\gamma}^{(m)}, \check{\gamma}^{(m)}, \epsilon|\theta, y)$. We want to find A, S and $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$ such that the function in (52) returns its minimum, making sure that the restrictions in (41) and (42) holds, and that S is positive semi-definite. In other words, to find the optimal updating distribution with respect to

our optimality criterion we have to solve the optimization problem,

$$\begin{aligned}
\underset{A,p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)}{\operatorname{argmin}} \quad & \mathrm{E}\left[\left(V\big((\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}})+(\tilde{z}^{(m)}-\check{z}^{(m)})\big)\right)^T \right. \\
& \qquad \left. \cdot\left(V\big((\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}})+(\tilde{z}^{(m)}-\check{z}^{(m)})\big)\right)\right] \\
\text{s.t.} \quad & S=\sigma^2(I-AA^T-KH)\succeq 0, \\
& p_{\Gamma|\Theta,Y}(\tilde{\gamma}^{(m)}|\theta,y)=\sum_{\check{\gamma}^{(m)}} p_{\tilde{\Gamma}^{(m)}|\check{\Gamma}^{(m)},\Theta,Y}(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)p_{\check{\Gamma}^{(m)}|\Theta}(\check{\gamma}^{(m)}|\theta), \\
& 1=\sum_{\check{\gamma}^{(m)}\in\mathbf{\Omega}_\gamma} p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y), \quad \forall \quad \tilde{\gamma}^{(m)}\in\mathbf{\Omega}_\gamma, \\
& 0\le p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)\le 1, \quad \forall \quad \check{\gamma}^{(m)},\tilde{\gamma}^{(m)}\in\mathbf{\Omega}_\gamma.
\end{aligned} \tag{53}$$

In Appendix A, we show that the terms involving A and $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)$ are separated. For this reason, we can solve the minimization problem by solving it with respect to A and $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)$ separately. Hence, the minimization problem in (53) can be solved by solving,

$$\begin{aligned}
\underset{A}{\operatorname{argmin}} \quad & \mathrm{E}\left[\left(V\big((\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}})+(\tilde{z}^{(m)}-\check{z}^{(m)})\big)\right)^T \right. \\
& \qquad \left. \cdot\left(V\big((\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}})+(\tilde{z}^{(m)}-\check{z}^{(m)})\big)\right)\right] \\
\text{s.t.} \quad & S=\sigma^2(I-AA^T-KH)\succeq 0,
\end{aligned} \tag{54}$$

resulting in A, defining the the optimal updating distribution for $\tilde{z}^{(m)}$ and

$$\begin{aligned}
\underset{p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)}{\operatorname{argmin}} \quad & \mathrm{E}\left[\left(V\big((\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}})+(\tilde{z}^{(m)}-\check{z}^{(m)})\big)\right)^T \right. \\
& \qquad \left. \cdot\left(V\big((\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}})+(\tilde{z}^{(m)}-\check{z}^{(m)})\big)\right)\right] \\
\text{s.t.} \quad & p_{\Gamma|\Theta,Y}(\tilde{\gamma}^{(m)}|\theta,y)=\sum_{\check{\gamma}^{(m)}\in\mathbf{\Omega}_\gamma} p_{\tilde{\Gamma}^{(m)}|\check{\Gamma}^{(m)},\Theta,Y}(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)p_{\check{\Gamma}^{(m)}|\Theta}(\check{\gamma}^{(m)}|\theta), \\
& 1=\sum_{\check{\gamma}^{(m)}\in\mathbf{\Omega}_\gamma} p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y), \quad \forall \quad \tilde{\gamma}^{(m)}\in\mathbf{\Omega}_\gamma \\
& 0\le p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)\le 1 \quad \forall \quad \check{\gamma}^{(m)},\tilde{\gamma}^{(m)}\in\mathbf{\Omega}_\gamma.
\end{aligned} \tag{55}$$

resulting in the optimal updating distribution for $\tilde{\gamma}^{(m)}$.

We first focus on solving (54). We begin with subtracting $\check{z}^{(m)}$ form both sides of (50),

$$\begin{aligned}\tilde{z}^{(m)} - \check{z}^{(m)} &= (A-I)\check{z}^{(m)} + \sigma^2(I-KH)H^TR^{-1}(y - H\mu_{\tilde{\gamma}^{(m)}}) + \epsilon \\ &= (A-I)\check{z}^{(m)} + M(y - H\mu_{\tilde{\gamma}^{(m)}}) + \epsilon,\end{aligned} \tag{56}$$

with $M = \sigma^2(I-KH)H^TR^{-1}$. Our objective function in (54) can then be written as,

$$\begin{aligned}&\mathrm{E}\left[g(\mu_{\tilde{\gamma}^{(m)}} + \check{z}^{(m)}, \mu_{\tilde{\gamma}^{(m)}} + \tilde{z}^{(m)})\right] \\ &= \mathrm{E}\left[\left(V\big((A-I)\check{z}^{(m)} + (I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\tilde{\gamma}^{(m)}} + My + \epsilon\big)\right)^T\right. \\ &\qquad\qquad \left.\cdot\left(V\big((A-I)\check{z}^{(m)} + (I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\tilde{\gamma}^{(m)}} + My + \epsilon\big)\right)\right],\end{aligned} \tag{57}$$

In Appendix A, we show that we can write the objective function in (57) as,

$$\begin{aligned}&\mathrm{E}\left[\left(V\big((A-I)\check{z}^{(m)} + (I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\tilde{\gamma}^{(m)}} + My + \epsilon\big)\right)^T\right. \\ &\qquad\qquad \left.\cdot\left(V\big((A-I)\check{z}^{(m)} + (I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\tilde{\gamma}^{(m)}} + My + \epsilon\big)\right)\right]. \\ &= \sigma^2 V \operatorname{tr}\big[2I - KH\big]V^T - 2\sigma^2 V \operatorname{tr}\big[A\big]V^T \\ &\quad + V \operatorname{Cov}\big((I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\tilde{\gamma}^{(m)}} + My\big)V^T\big] \\ &\quad + \mathrm{E}\big[(I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\tilde{\gamma}^{(m)}} + My\big]^T \mathrm{E}\big[(I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\tilde{\gamma}^{(m)}} + My\big].\end{aligned} \tag{58}$$

We see that the second term is the only term depending on A . Hence, solving the minimization problem in (54) is equivalent to solving the maximization problem,

$$\begin{aligned}\underset{A}{\text{argmax}} \quad & \operatorname{tr}\left\{VAV^T\right\} \\ \text{s.t.} \quad & S = \sigma^2(I - KH - AA^T) \succeq 0.\end{aligned} \tag{59}$$

To solve this, we adopt the procedure from Loe and Tjelmeland(2020), Section 4.3. We Begin with the singular value decomposition of the covariance matrix $\sigma^2(I-KH)$,

$$\sigma^2(I-KH) = U\Lambda U^T, \tag{60}$$

where $U \in \mathbb{R}^{n\times n}$ is an orthogonal matrix, and $\Lambda \in \mathbb{R}^{n\times n}$ is a diagonal matrix. Define so,

$$\tilde{S} = \Lambda^{-\frac{1}{2}} U^T S U \Lambda^{\frac{1}{2}},$$

and

$$\tilde{A} = \sigma A^T U \Lambda^{-\frac{1}{2}}.$$

The expression for S in (51) is then equivalent to the expression,

$$\tilde{S} = I - \tilde{A}^T \tilde{A}.$$

Writing the objective function in (59) in terms of $\tilde{A}$ yields,

$$\begin{aligned}\text{tr}\left\{VAV^T\right\} &= \text{tr}\left\{VU\Lambda^{\frac{1}{2}}\tilde{A}^T V^T\right\} \\ &= \text{tr}\left\{\tilde{A}\Lambda^{\frac{1}{2}}U^T V^T V\right\} \\ &= \text{tr}\left\{\tilde{A}Z\right\},\end{aligned}$$

where $Z = \Lambda^{\frac{1}{2}} U^T V^T V$. We now use Theorem 1 in Loe and Tjelmeland (2021). Since $\sigma^2(I - KH)$ is positive definite, Λ is invertible. Thereby, $\Lambda^{\frac{1}{2}}$ is also invertible. We require $(V^T V)^{-1}$ to be positive definite, and this implies invertibility of $V^T V$. U is orthogonal and therefore invertible. Z is thereby a product of invertible matrices and is itself inveritible with full rank. With singular value decomposition $Z = PGF^T$, Theorem 1 states that the maximum value of $\text{tr}\left\{\tilde{A}Z\right\}$ under the constraint that $\tilde{S}$ is positive semi-definite occurs only for $\tilde{A} = FP^T$. We then get that,

$$\tilde{S} = I - \tilde{A}^T\tilde{A} = I - PF^T FP^T = I - I = 0.$$

This implies that all elements of S are also 0. The corresponding optimal value for A is then,

$$A = \frac{1}{\sigma} U \Lambda^{\frac{1}{2}} \tilde{A}^T = \frac{1}{\sigma} U \Lambda^{\frac{1}{2}} P F^T.$$

We have now found the matrix A, identifying the linear equation in (50), which samples $\tilde{z}^{(m)}$ from the filtering distribution. We will now proceed with showing how to solve the optimization problem in (55), resulting in the optimal distribution, $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$, from which to sample $\tilde{\gamma}^{(m)}$. We show that the optimization problem can be written as a linear programming problem, i.e. the objective function and the constraints are linear in the term $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$. First we define,

$$\begin{aligned}G_{\check{\gamma}^{(m)},\tilde{\gamma}^{(m)}} = \big[\mu_{\tilde{\gamma}^{(m)}}^T \big((I - 2H^T M^T)V^T V + H^T M^T V^T V M H\big)\mu_{\tilde{\gamma}^{(m)}} \\ - 2(My + \mu_{\tilde{\gamma}^{(m)}})^T V^T V (I - MH)\mu_{\tilde{\gamma}^{(m)}}\big] p(\check{\gamma}^{(m)}|\theta, y).\end{aligned} \tag{61}$$

In Appendix B, we show that,

$$\begin{aligned}
&\mathrm{E}\left[\left(V\big((\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}})+(\tilde{z}^{(m)}-\check{z}^{(m)})\big)\right)^T\right. \\
&\qquad\qquad\left.\cdot\left(V\big((\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}})+(\tilde{z}^{(m)}-\check{z}^{(m)})\big)\right)\right] \qquad (62)\\
&\quad=\sum_{\check{\gamma}^{(m)}\in\mathbf{\Omega}_\gamma}\sum_{\tilde{\gamma}^{(m)}\in\mathbf{\Omega}_\gamma^{(m)}} G_{\tilde{\gamma}^{(m)},\check{\gamma}^{(m)}}p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)+C,
\end{aligned}$$

where C is a constant not depending on $\tilde{\gamma}^{(m)}$. The last expression is linear in terms of the distribution $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)$. The optimization problem in (55) is therefore a linear programming problem, which can be solved numerically. More thoroughly, we show in Appendix C, that the optimization problem can be written on the same form as the general problem in (12). The optimal sampling distribution for $\tilde{\gamma}^{(m)}$ with respect to our optimality criterion can then be solved by a linear programming tool.

5 Simulation examples

In this section, we present three simulation examples with our assumed model specified in Section 4. In the examples, we generate data with different degrees of overlap of the distribution of the observations, simulated from different hidden state values. First, we describe how we generate the true hidden states for all three examples and simulate corresponding observations in Section 5.1. After that, we specify values for all parameters in our assumed model in Section 5.2. The same parameter assumptions yield for all three examples.

5.1 Experimental setup

In this section we describe how we generate the true hidden states with corresponding observations in all three examples. The generated true hidden states are compared to the update ensembles from the algorithm in Section 5.3. At all times t, we choose the state variable $x^t=[\gamma^t,z^t]$ to consist of the two state vectors $\gamma^t=(\gamma_1^t,\ldots,\gamma_n^t)$ and $z^t=(z_1^t,\ldots,z_n^t)$, both having dimension $n=6$, so that $\mathbf{\Omega}_x=\{0,1\}^6\times\mathbb{R}^6$. We define the state space model for time 1 to time $T=20$. We choose the true categorical states,

$$\gamma^t=\begin{bmatrix}1 & 1 & 1 & 0 & 0 & 0\end{bmatrix}^T \quad \forall \quad t=1,\ldots,20. \qquad (63)$$

In Figure 5 we illustrate the categorical state vectors. One pixel represents the value

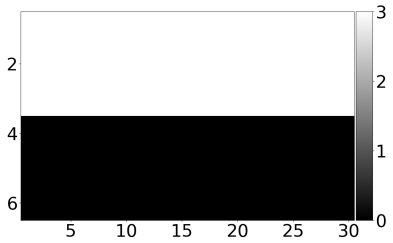

Figure 5: The true value of γ_n^t for all components n at all times t. The x-axis represents time t, and the y-axis represents component n. The component $\gamma_n^t = 1$ where the image pixels are white and $\gamma_n^t = 0$ where the image pixels are black.

of γ_n^t. The pixel is black if $\gamma_n^t = 0$, and white if $\gamma_n^t = 1$. We choose the continuous noise variables z^t to come from a VAR(1) model as in (13). We choose,

$$\Phi = \begin{bmatrix} 0.6 & 0.2 & 0 & 0 & 0 & 0 \\ 0.2 & 0.6 & 0.2 & 0 & 0 & 0 \\ 0 & 0.2 & 0.6 & 0.2 & 0 & 0 \\ 0 & 0 & 0.2 & 0.6 & 0.2 & 0 \\ 0 & 0 & 0 & 0.2 & 0.6 & 0.2 \\ 0 & 0 & 0 & 0 & 0.2 & 0.6 \end{bmatrix},$$

i.e. the matrix with 0.6 on the main diagonal and 0.2 on the super- and subdiagonal. When we choose Υ, the true covariance of ω^t in the VAR(1) model, we want to make sure that the covariance of z^t do not change over time. Namely, we want that $\text{Cov}[z^t] = \text{Cov}[z^{t-1}] = W$ for all $t = 2, \ldots, T$. To choose an appropriate W, we make use of that we can construct a positive semi-definite matrix W by,

$$W = w_0^2 W_1 W_1^T,$$

where $w_0 \in \mathbb{R}^n$ and $W_1 \in \mathbb{R}^{n \times n}$ are any arbitrary scalar and matrix, respectively. We set,

$$W_1 = \begin{bmatrix} 3 & 2 & 0 & 0 & 0 & 0 \\ 1 & 3 & 1 & 0 & 0 & 0 \\ 0 & 1 & 3 & 1 & 0 & 0 \\ 0 & 0 & 1 & 3 & 1 & 0 \\ 0 & 0 & 0 & 1 & 3 & 1 \\ 0 & 0 & 0 & 0 & 2 & 3 \end{bmatrix},$$

and set $w_0 = 0.4$. We then get the value of Υ by taking the covariance on both sides of (13), and solving the resulting equation with respect to Υ,

$$\Upsilon = W - \Phi W \Phi^T.$$

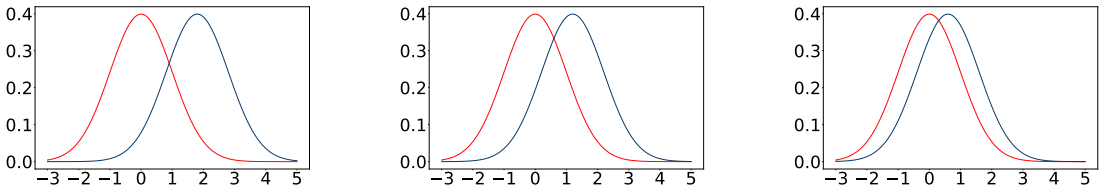

Figure 6: The marginal distribution of the observations of $y_n^t|\gamma_n^t = 0, z_n^t = 0$ are presented as blue curves, and the marginal distribution of the observations of $y_n^t|\gamma_n^t = 1, z_n^t = 0$ are presented as blue curves, for all three simulation examples. The distribution in Example 1 is shown on the left, Example 2 in the middle and Example 3 on the right. The distribution of $y_n^t|x_n^t$ when $z^t = 0$ does not vary over time as we assume the categorical variables to be constant. The matrix $H = 0.6I$ in Example 1, $H = 0.4I$ in Example 2 and $H = 0.2I$ in Example 3.

Note that Υ is not guaranteed to be positive semi-definite with this setup, but it is in this particular case.

Having generated the true state vectors $\{x^t\}_{t=1}^T$, we now generate for for each $t = 1, \ldots, 30$, an observation vector $y^t = (y_1^t, \ldots, y_6^t)$, according to the distribution in (30). We set the parameter

$$\mu = \begin{bmatrix} 0 & 3 \end{bmatrix}^T,$$

i.e. $\mu_0 = 0$ and $\mu_1 = 3$, in all three examples and choose $R = I$. Note that a vector μ with these values will make the categorical vector dominate in the likelihood as the values in the resulting matrix W will be in the range between μ_0 and μ_1. The value of w_0 was chosen with this in mind, so that the elements of W is in the appropriate range. We set the matrix $H = h_0 I$, where h_0 is a scalar we vary in all three examples. We set $h_0 = 0.6$ in Example 1, $h_0 = 0.4$ in Example 2 and $h_0 = 0.2$ in Example 3. With lower value of h_0, and no changes in the other parameters, the covariance R is more dominate in the likelihood $y^t|x^t$, and the value of the categorical variables have less influence. In other words, we vary the difference between the marginal distribution of $y_n^t|x_n^t$ in the three examples, when $\gamma_n^t = 0$ and $\gamma_n^t = 1$. This is illustrated in Figure 6, where we have plotted the distribution of $y_n^t|\gamma_n^t = 0, z_n^t = 0$ in red, and $y_n^t|\gamma_n^t = 1, z_n^t = 0$ in blue, for all three examples. The same effect could be achieved by varying R. We assume the model to perform best in Example 1, where the distribution of $y_n^t|\gamma_n^t = 1$ and $y_n^t|\gamma_n^t = 0$ do not overlap as much as in Example 2 and 3. The overlap in Example 3 seems crucial, and the model is not assumed to perform as well here as in the two other examples. The values of h_0 are chosen with this in mind. The true value of $\mu_{\gamma^t} + z^t$ and the simulated observations y^t for $t = 1, \ldots, 30$ are plotted for all three examples in Figure 7. As is anticipated from

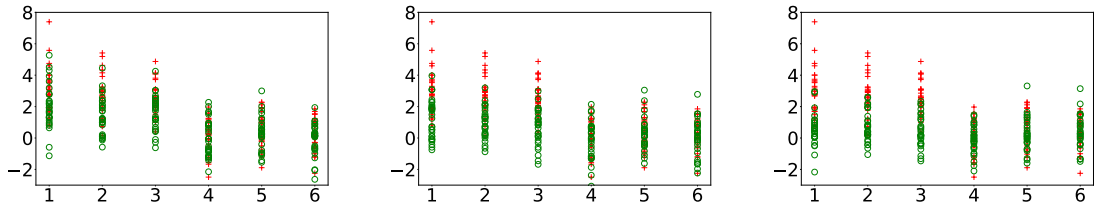

Figure 7: The generated true values $\mu_{\gamma^t} + z^t$ as red plus signs and the simulated observations in green circles, for all times $t = 1, \ldots, 20$. The x-axis represents element n of the vector where $n = 1, \ldots, 6$. Example 1 is shown in the leftmost plot, Example 2 in the plot in the middle and Example 3 in the rightmost plot. Note that the true state vectors are equal in all three plots.

Figure 6, the values of y_n^t for $n = 1, 2$ and 3 and for $n = 4, 5$ and 6, are separated most in Example 1. The separation is less in Example 2 and least in Example 3.

5.2 Specifications in our assumed model

Once we have generated the reference data, we specify our assumptions for the simulation procedure. We assume that,

$$\theta_1, \theta_{r0}, \theta_{r_1} \sim \text{Beta}(1, 1),$$

for all $r = 2, \ldots, n$, i.e. all θs are uniformly distributed. We use 100 iterations in the Gibbs sampler for the parameter distribution. We choose to simulate $M = 30$ ensemble members at each time t. For the Mahalanobis distance $g(\check{x}^m, \tilde{x}^{(m)})$, We use the Euclidean distance, i.e. $\Sigma = I$. For the prediction step, we assume we can obtain the prediction ensemble member $\check{x}^{t(m)}$ by,

$$\check{x}^{t(m)} = \tilde{x}^{t-1(m)},$$

for $t = 2, \ldots, 20$. In other words, we assume that $\check{\gamma}^{t(m)} = \tilde{\gamma}^{t-1(m)}$ and $\check{z}^{t(m)} = \tilde{z}^{t-1(m)}$. We obtain the initial prediction ensemble member $\gamma^{1(m)}$ by simulating from a first order Markov chain with initial and transition probabilities all set to 0.5. We obtain the initial prediction ensemble $z^{1(m)}$ by simulating from a Gaussian distribution with mean 0 and covariance I.

5.3 Simulation results

In this section we present the result of the simulations using the Bayesian ensemble filtering algorithm derived in Section 4, with data generated as explained in Section

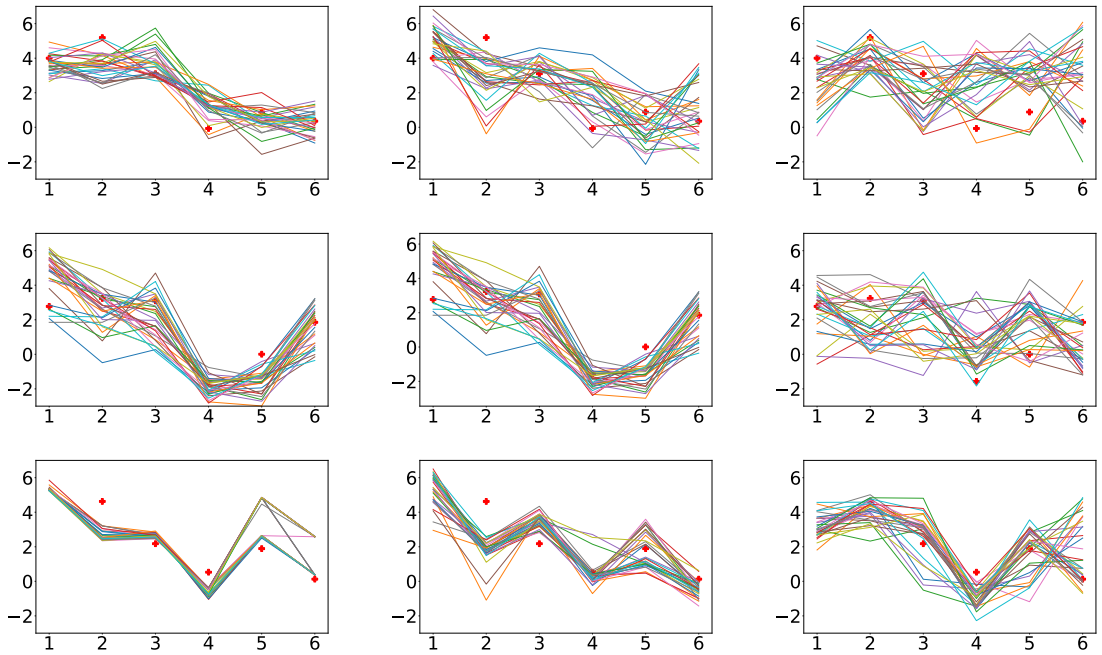

Figure 8: Updating ensemble at $t = 3$, $t = 15$ and $t = 30$ in the first, second and third row respectively, for the three examples column-wise. Example 1 is shown in the column on the left, Example 2 in the column in the middle and Example 3 in the column on the right. The solid lines show the value of the elements of the vectors $\mu_{\tilde{\gamma}^{t(m)}} + \tilde{z}^{t(m)}$, for $m = 1, \ldots, 30$. The true state vector expression, $\mu_{\gamma^t} + z^t$, are shown with red pluses.

5.1 and specifications for the simulation presented in Section 5.2. We evaluate the results based on the generated true data. In Figure 8 , we show the simulated values of $\mu_{\tilde{\gamma}^{t(m)}} + \tilde{z}^{t(m)}$ for all $m = 1, \ldots, 30$, together with the true value $\mu_{\gamma^t} + z^t$, at times $t = 3$, $t = 15$ and $t = 30$. In the upper plots, the true values seem to lie inside the spread of the ensemble members. However, the spread declines over time, and at $t = 30$, the spread of the ensemble members do not cover the true values. It seems that the variance of the filtering distribution is underestimated. As discussed in Section 3, the underestimation of the variance in the EnKF is to some extent accounted for in our assumed model, as we consider the parameters as random variables. However, we have only assumed that the transition probabilities in the categorical variables' Markov chain to be unknown. One improvement of the model could be to simulate all parameters used in the model.

The marginal distribution of the categorical variables are estimated in Figure 9. Compared to the image in Figure 5 , we see that Example 1 identifies the true value

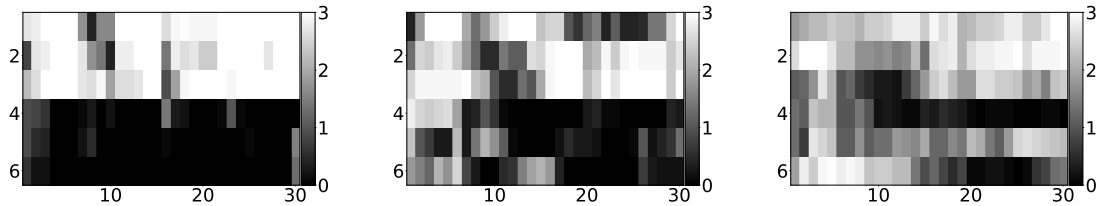

Figure 9: The estimated marginal distributions of the categorical variables for all three examples. Example 1 is shown in the image on the left, Example 2 in the image in the middle and Example 3 in the image on the right. The average value of the simulated categorical variables $\{\tilde{\gamma}_n^{t(m)}\}_{m=1}^M$ are plotted against time, in gray-scale. The y-axis is the elements, n. A white pixel at time t, element n, means that all ensemble members $\tilde{\gamma}^{t(m)}$ have the value one 1, in element n. Note that a perfect simulation would yield the same image as in Figure 5.

of the latent categorical variables almost immediately, with only some deviations. The deviations could be the result of an outlier observation in that specific time the deviation occurs. In Example 2, we see that the model have some difficulty to identify the true categorical values in the beginning, but manages to identify them to some degree at later times. In Example 3, we see that the model struggles to identify the true values and performs quite bad. This result was anticipated in Section 5.1.

6 Closing remarks

In this report, we have applied the Bayesian ensemble filtering framework, proposed in Loe and Tjelmeland (2021), on a new assumed model, consisting of both continuous and categorical variables. For a fixed time t, we derive the class of updating distribution from which to update a prior ensemble to a posterior ensemble of independent realizations from the filtering distribution. We propose an optimality criterion, limiting the class of distributions to the optimal filter with respect to the criterion.

The latent state variables in our state space model consist of two vectors, one categorical and one continuous vector. The elements of the categorical vector take on either the value 0 or the value 1, and we assume that it comes from a first order Markov chain. The continuous vector contributes to the model as noise, and we assume that it is Gaussian distributed with mean 0. We assume a linear-Gaussian model for the observations, with mean depending on both state vectors. The transition probabilities in the Markov chain for the categorical variables are treated as random variables, and we have derived an algorithm from which to simulate the

parameters. The updating framework for this particular model is derived from an optimality criterion that restricts the solution to making minimal changes to the ensemble members

We present three simulation examples, showing the performance of the derived updating algorithm. In the examples, we vary the values on the diagonal of the matrix H, i.e., we vary the contribution of the hidden states to the mean of the likelihood. In example 1, the likelihood distributions when the elements of the categorical vector takes on values 1 and 0 are more separated than in example 2. The model manages to identify the true value of the categorical vectors quite well when it is expected to, that is in Example 1 and 2. In example 3, these distributions have a critical overlap, and the model does not perform well here, as expected. At later times, we see that the spread of the ensemble members do not capture the true values of $\mu_{\gamma^t} + z^t$. which means that the method does seem to underestimate the variance in the final filtering distribution.

We study our assumed model, treating the transition probabilities in the Markov chain as random variables. One interesting task for the future could be to treat also the remaining parameters as random variables. Treating more parameters as random variables could increase the spread of the ensemble members in the simulation and thereby increase the estimated variance. We do not prioritize computational efficiency in the computed framework. Computing all possible transition probabilities from $\check{\gamma}^{(m)}$ to $\tilde{\gamma}^{(m)}$ is the most demanding part of the algorithm. Hence, we choose a sufficient low dimension for the state vectors in our simulation examples. It would be interesting to explore ideas that would lower the complexity of the algorithm. One idea that would reduce the algorithm's complexity would be to restrict the allowed dependencies between $\check{\gamma}^{t(m)}$ and $\tilde{\gamma}^{t(m)}$, which would lower the number of possible transition probabilities, as done in the first order Markov chain assumed model in Loe and Tjelmeland (2021).

Bibliography

Brockwell, P. J. and Davis, R. A. (2016). *Introduction to Time Series and Forecasting*. Springer, Cham.

Cassola, F. and Burlando, M. (2012). "Wind speed and wind energy forecast through Kalman filtering of numerical weather prediction model output." *Applied Energy*, 99, 154–166.

Evensen, G. (1994). "Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics." *Geophysical Research*, 99, C5, 10143–10162.

Katzfuss, M., Stroud, J., and Wikle, C. (2016). "Understanding the ensemble Kalman filter." *The American Statistician*, 70, 350–357.

Loe, M. and Tjelmeland, H. (2020). "Ensemble updating of binary state vectors by maximizing the expected number of unchanged components." *Scandinavian Journal of Statistics*, To Appear.

Loe, M. K. and Tjelmeland, H. (2021). "A generalised and fully Bayesian framework for ensemble updating." arXiv:0712.0157v1.

Myrseth, I. and Omre, H. (2010). "Hierarchical ensemble Kalman filter." *Society of Petroleum Engineers Journal*, 15, 569–580.

Nocedal, J. and Wright, S. (2006). *Numerical Optimization. Second edition*. Springer.

A Expansion of the minimization function

Here we will show that the function in (52) can be separated in terms depending on A and terms depending on $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$. We start by seeing that we can write,

$$\begin{aligned}
&\mathrm{E}\left[V\big((A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\big)\right] \\
&\qquad = V\,\mathrm{E}\left[(A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\right] \\
&\qquad = V\,\mathrm{E}\left[(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My\right],
\end{aligned} \tag{64}$$

where the last equality holds as then mean of $\check{z}^{(m)}$ and ϵ are both 0. Moreover, we have that,

$$\begin{aligned}
&\mathrm{Cov}\left[V\big((A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\big)\right] \\
&\qquad = V\,\mathrm{Cov}\left[(A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\right]V^T \\
&\qquad = V\big[\sigma^2(A-I)(A-I)^T+S\big]V^T \\
&\qquad\qquad + V\,\mathrm{Var}\left[(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My\right]V^T.
\end{aligned}$$

Replacing S with the expression in (51) yields,

$$\begin{aligned}
&\mathrm{Cov}\left[V\big((A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\big)\right] \\
&\qquad = V\big[\sigma^2(A-I)(A-I)^T+\sigma^2(I-AA^T-KH)\big]V^T \\
&\qquad\qquad + V\,\mathrm{Cov}\left[(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My\right]V^T. \\
&\qquad = V\big[\sigma^2(2I-A-A^T-KH)\big]V^T \\
&\qquad\qquad + V\,\mathrm{Cov}\left[(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My\right]V^T.
\end{aligned} \tag{65}$$

For any stochastic vector w, we have the identity $\mathrm{E}[w^Tw] = \mathrm{tr}[\mathrm{Cov}(w)] + \mathrm{E}[w]^T\,\mathrm{E}[w]$. Using this identity with $w = V[(\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})]$, we can write the minimization function in (52) as,

$$\begin{aligned}
&\mathrm{E}\left[\left(V\big((A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\big)\right)^T\right. \\
&\qquad\qquad \left.\cdot\left(V\big((A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\big)\right)\right]. \\
&= \mathrm{tr}\left[\mathrm{Cov}\left(V\big((A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\big)\right)\right] \\
&\qquad + \mathrm{E}\left[V\big((A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\big)\right]^T \\
&\qquad\qquad \cdot \mathrm{E}\left[V\big((A-I)\check{z}^{(m)}+(I-MH)\mu_{\tilde{\gamma}^{(m)}}-\mu_{\check{\gamma}^{(m)}}+My+\epsilon\big)\right]
\end{aligned} \tag{66}$$

Inserting the expressions in (64) and (65) and thereafter using the trace is a linear mapping and that $\mathrm{tr}[X^T] = \mathrm{tr}[X]$ holds for any matrix X, we can further expand (66),

$$
\begin{aligned}
&\mathrm{E}\left[\left(V\big((A-I)\check{z}^{(m)} + (I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}} + My + \epsilon\big)\right)^T \right. \\
&\qquad\qquad \left. \cdot \left(V\big((A-I)\check{z}^{(m)} + (I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}} + My + \epsilon\big)\right)\right]. \\
&= \mathrm{tr}\big[\sigma^2 V(2I - A - A^T - KH)\big)V^T \\
&\qquad\qquad + V\,\mathrm{Cov}\left((I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}} + My\big)V^T\right] \\
&\quad + \mathrm{E}\left[(I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}} + My\right]^T \mathrm{E}\left[(I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}} + My\right] \\
&= \sigma^2 V\,\mathrm{tr}\left[2I - KH\right]V^T - 2\sigma^2 V\,\mathrm{tr}\left[A\right]V^T \\
&\quad + V\,\mathrm{Cov}\left((I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}} + My\big)V^T\right] \\
&\quad + \mathrm{E}\left[(I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}} + My\right]^T \mathrm{E}\left[(I-MH)\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}} + My\right]
\end{aligned}
$$

We see that the second term in the last equality is the only term depending on A, and that it is independent of $\tilde{\gamma}^{(m)}$ and $\check{\gamma}^{(M)}$. This proves that the optimality function in (52), can be written in terms of A and $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$ separately.

B Derivation of the linear programming problem for gamma

We will here expand the objective function in (55) and show that it can be written in linear terms of $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$. We start by expanding the objective function,

$$
\begin{aligned}
&\mathrm{E}\left[\left(V\big((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\big)\right)^T \right. \\
&\qquad\qquad \left. \cdot \left(V\big((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\big)\right)\right] \\
&= \mathrm{E}\left[(\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}})^T V^T V(\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}})\right. \\
&\qquad\qquad + 2(\tilde{z}^{(m)} - \check{z}^{(m)})^T V^T V(\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) \\
&\qquad\qquad \left. + (\tilde{z}^{(m)} - \check{z}^{(m)})^T V^T V(\tilde{z}^{(m)} - \check{z}^{(m)})\right].
\end{aligned}
$$

We then insert the expression for $\tilde{z} - \check{z}$ in (56),

$$
\begin{aligned}
&\mathrm{E}\left[\left(V\big((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\big)\right)^T \right. \\
&\qquad\qquad \left. \cdot \left(V\big((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\big)\right)\right] \\
&= \mathrm{E}\big[(\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}})^T V^T V (\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) \\
&\qquad + 2\big((A-I)\check{z}^{(m)} + M(y - H\mu_{\tilde{\gamma}^{(m)}})\big)^T V^T V (\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) \\
&\qquad + \big((A-I)\check{z}^{(m)} + M(y - H\mu_{\tilde{\gamma}^{(m)}})\big)^T V^T \\
&\qquad\qquad \cdot V\big((A-I)\check{z}^{(m)} + M(y - H\mu_{\tilde{\gamma}^{(m)}})\big)\big].
\end{aligned}
$$

Only considering the terms depending on $\tilde{\gamma}^{(m)}$ reduces the expression to,

$$
\begin{aligned}
&\mathrm{E}\left[\left(V\big((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\big)\right)^T \right. \\
&\qquad\qquad \left. \cdot \left(V\big((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\big)\right)\right] \\
&= \mathrm{E}\big[\mu_{\tilde{\gamma}^{(m)}}^T V^T V \mu_{\tilde{\gamma}^{(m)}} - 2\mu_{\check{\gamma}^{(m)}}^T V^T V \mu_{\tilde{\gamma}^{(m)}} \\
&\qquad + 2\big(\check{z}^{(m)^T}(A-I)^T V^T V + y^T M^T V^T V + \mu_{\check{\gamma}^{(m)}}^T V^T V M H\big)\mu_{\tilde{\gamma}^{(m)}} \\
&\qquad - 2\mu_{\tilde{\gamma}^{(m)}}^T H^T M^T V^T V \mu_{\tilde{\gamma}^{(m)}} + \mu_{\tilde{\gamma}^{(m)}}^T H^T M^T V^T V M H \mu_{\tilde{\gamma}^{(m)}} \\
&\qquad + 2\big(\check{z}^{(m)^T}(A-I)^T + y^T M^T\big) V^T V M H \mu_{\tilde{\gamma}^{(m)}}\big] + C \\
&= \mathrm{E}\big[\mu_{\tilde{\gamma}^{(m)}}^T \big(V^T V - 2H^T M^T V^T V + H^T M^T V^T V M H\big)\mu_{\tilde{\gamma}^{(m)}} \\
&\quad + 2\big(\check{z}^{(m)^T}(A-I)^T V^T V - y^T M^T V^T V + \mu_{\check{\gamma}^{(m)}}^T V^T V M H \\
&\qquad - \mu_{\check{\gamma}^{(m)}}^T V^T V - \check{z}^{(m)^T}(A-I)^T V^T V M H - y^T M^T V^T V M H\big)\mu_{\tilde{\gamma}^{(m)}}\big] + C \\
&= \mathrm{E}\big[\mu_{\tilde{\gamma}^{(m)}}^T \big((I - 2H^T M^T) V^T V + H^T M^T V^T V M H\big)\mu_{\tilde{\gamma}^{(m)}} \\
&\qquad + 2\big(\big((A-I)\check{z}^{(m)} - My - \mu_{\check{\gamma}^{(m)}}\big)^T V^T V \\
&\qquad\qquad - \big((A-I)\check{z}^{(m)} - My - \mu_{\check{\gamma}^{(m)}}\big)^T V^T V H\big)\mu_{\tilde{\gamma}^{(m)}}\big] + C \\
&= \mathrm{E}\big[\mu_{\tilde{\gamma}^{(m)}}^T \big((I - 2H^T M^T) V^T V + H^T M^T V^T V M H\big)\mu_{\tilde{\gamma}^{(m)}} \\
&\qquad + 2\big((A-I)\check{z}^{(m)} - My - \mu_{\check{\gamma}^{(m)}}\big)^T V^T V (I - MH)\mu_{\tilde{\gamma}^{(m)}}\big] + C,
\end{aligned}
$$

with C being a constant, not depending on $\tilde{\gamma}^{(m)}$. Now, using that $\check{z} \perp \tilde{\gamma}$ and that $\mathrm{E}(\check{z}) = 0$ yields,

$$
\begin{aligned}
&\mathrm{E}\left[\left(V\big((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\big)\right)^T \right. \\
&\qquad\qquad \left. \cdot \left(V\big((\mu_{\tilde{\gamma}^{(m)}} - \mu_{\check{\gamma}^{(m)}}) + (\tilde{z}^{(m)} - \check{z}^{(m)})\big)\right)\right] \\
&= \mathrm{E}\left[\mu_{\tilde{\gamma}^{(m)}}^T\big((I - 2H^TM^T)V^TV + H^TM^TV^TVMH\big)\mu_{\tilde{\gamma}^{(m)}}\right. \\
&\qquad \left. + 2\big((A-I)\check{z}^{(m)} - My - \mu_{\check{\gamma}^{(m)}}\big)^TV^TV(I-MH)\mu_{\tilde{\gamma}^{(m)}}\right] + C \\
&= \mathrm{E}\left[\mu_{\tilde{\gamma}^{(m)}}^T\big((I - 2H^TM^T)V^TV + H^TM^TV^TVMH\big)\mu_{\tilde{\gamma}^{(m)}}\right. \\
&\qquad \left. - 2(My + \mu_{\check{\gamma}^{(m)}})^TV^TV(I-MH)\mu_{\tilde{\gamma}^{(m)}}\right] + C \\
&= \sum_{\check{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma} \sum_{\tilde{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma} \left[\mu_{\tilde{\gamma}^{(m)}}^T\big((I - 2H^TM^T)V^TV + H^TM^TV^TVMH\big)\mu_{\tilde{\gamma}^{(m)}}\right. \\
&\qquad\qquad \left. - 2(My + \mu_{\check{\gamma}^{(m)}})^TV^TV(I-MH)\mu_{\tilde{\gamma}^{(m)}}\right] \\
&\qquad\qquad \cdot p(\check{\gamma}^{(m)}|\theta, y)p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y) + C \\
&= \sum_{\check{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma} \sum_{\tilde{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma} G_{\check{\gamma}^{(m)},\tilde{\gamma}^{(m)}} p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y) + C,
\end{aligned}
$$

with $G_{\check{\gamma}^{(m)},\tilde{\gamma}^{(m)}}$ as in (61). The last expression is linear in terms of the distribution $p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y)$. This expression is then used in the optimization problem which results in the updating distribution for $\tilde{\gamma}^{(m)}$.

C Matrix representation

Here we will show that the optimization problem in (55) can be written on the same form as the general problem in (12). By replacing the objective function in (43) with the expression in (62), we can rewrite the optimization problem to,

$$
\begin{aligned}
&\underset{p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)},\theta,y)}{\operatorname{argmin}} \sum_{\check{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma} \sum_{\tilde{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma} G_{\check{\gamma}^{(m)},\tilde{\gamma}^{(m)}} p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y) \\
&\text{s.t.} \quad p_{\Gamma|\Theta,Y}(\tilde{\gamma}^{(m)}|\theta, y) = \sum_{\check{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma} p_{\tilde{\Gamma}^{(m)}|\check{\Gamma}^{(m)},\Theta,Y}(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y) p_{\check{\Gamma}^{(m)}|\Theta}(\check{\gamma}^{(m)}|\theta), \\
&\qquad\qquad 1 = \sum_{\check{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma} p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y), \quad \forall \quad \tilde{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma \\
&\qquad\qquad 0 \leq p(\tilde{\gamma}^{(m)}|\check{\gamma}^{(m)}, \theta, y) \leq 1 \quad \forall \quad \check{\gamma}^{(m)}, \tilde{\gamma}^{(m)} \in \mathbf{\Omega}_\gamma.
\end{aligned}
\tag{67}
$$

We order the categorical space $\mathbf{\Omega}_\gamma = \{\gamma^1, \gamma^2, \ldots, \gamma^{2^n}\}$ and define the matrix $X \in \mathbb{R}^{2^n \times 2^n}$ as,

$$X = \begin{bmatrix} p(\tilde{\gamma}^{1(m)}|\check{\gamma}^{1(m)}, \theta, y) & p(\tilde{\gamma}^{2(m)}|\check{\gamma}^{1(m)}, \theta, y) & \ldots & p(\tilde{\gamma}^{2^n(m)}|\check{\gamma}^{1(m)}, \theta, y) \\ p(\tilde{\gamma}^{1(m)}|\check{\gamma}^{2(m)}, \theta, y) & p(\tilde{\gamma}^{2(m)}|\check{\gamma}^{2(m)}, \theta, y) & \ldots & p(\tilde{\gamma}^{2^n(m)}|\check{\gamma}^{2(m)}, \theta, y) \\ \vdots & \vdots & \ddots & \vdots \\ p(\tilde{\gamma}^{1(m)}|\check{\gamma}^{2^n(m)}, \theta, y) & p(\tilde{\gamma}^{2(m)}|\check{\gamma}^{2^n(m)}, \theta, y) & \ldots & p(\tilde{\gamma}^{2^n(m)}|\check{\gamma}^{2^n(m)}, \theta, y) \end{bmatrix}. \quad (68)$$

Namely, the element $X_{i,j}$ in row i, column j, of the matrix X, defines the probability from which we sample $\tilde{\gamma}^{(m)} = \gamma^j$ from the filtering distribution, given the prediction sample $\check{\gamma}^{(m)} = \gamma^i$. We then define the matrix $G \in \mathbb{R}^{2^n \times 2^n}$ as,

$$G = \begin{bmatrix} G_{\check{\gamma}^{1(m)}, \tilde{\gamma}^{1(m)}} & G_{\check{\gamma}^{1(m)}, \tilde{\gamma}^{2(m)}} & \ldots & G_{\check{\gamma}^{1(m)}, \tilde{\gamma}^{2^n(m)}} \\ G_{\check{\gamma}^{2(m)}, \tilde{\gamma}^{1(m)}} & G_{\check{\gamma}^{2(m)}, \tilde{\gamma}^{2(m)}} & \ldots & G_{\check{\gamma}^{2(m)}, \tilde{\gamma}^{2^n(m)}} \\ \vdots & \vdots & \ddots & \vdots \\ G_{\check{\gamma}^{2^n(m)}, \tilde{\gamma}^{1(m)}} & G_{\check{\gamma}^{2^n(m)}, \tilde{\gamma}^{2(m)}} & \ldots & G_{\check{\gamma}^{2^n(m)}, \tilde{\gamma}^{2^n(m)}} \end{bmatrix}, \quad (69)$$

i.e. a matrix with where element $G_{i,j} = G_{\check{\gamma}^{i(m)}, \tilde{\gamma}^{j(m)}}$ defined in (61). We so define the vectors $x \in \mathbb{R}^{2^{2n}}$ and $g \in \mathbb{R}^{2^{2n}}$ as,

$$x = \begin{bmatrix} p(\tilde{\gamma}^{1(m)}|\check{\gamma}^{1(m)}, \theta, y) \\ \vdots \\ p(\tilde{\gamma}^{2^n(m)}|\check{\gamma}^{1(m)}, \theta, y) \\ p(\tilde{\gamma}^{1(m)}|\check{\gamma}^{2(m)}, \theta, y) \\ \vdots \\ p(\tilde{\gamma}^{2^n(m)}|\check{\gamma}^{2(m)}, \theta, y) \\ \vdots \\ p(\tilde{\gamma}^{1(m)}|\check{\gamma}^{2^n(m)}, \theta, y) \\ \vdots \\ p(\tilde{\gamma}^{2^n(m)}|\check{\gamma}^{2^n(m)}, \theta, y) \end{bmatrix}, g = \begin{bmatrix} W_{\check{\gamma}^{1(m)}, \tilde{\gamma}^{1(m)}} \\ \vdots \\ W_{\check{\gamma}^{1(m)}, \tilde{\gamma}^{2^n(m)}} \\ W_{\check{\gamma}^{2(m)}, \tilde{\gamma}^{1(m)}} \\ \vdots \\ W_{\check{\gamma}^{2(m)}, \tilde{\gamma}^{2^n(m)}} \\ \vdots \\ W_{\check{\gamma}^{2^n(m)}, \tilde{\gamma}^{1(m)}} \\ \vdots \\ W_{\check{\gamma}^{2^n(m)}, \tilde{\gamma}^{2^n(m)}} \end{bmatrix}, \quad (70)$$

being the flattened vectors of the matrices X in (68) and W in (69) in row-major order, respectively. We then define the diagonal matrix $D \in \mathbb{R}^{2^n \times 2^n}$ as,

$$D = \begin{bmatrix} p(\check{\gamma}^{1(m)}|\theta) & 0 & \ldots & 0 \\ 0 & p(\check{\gamma}^{2(m)}|\theta) & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & p(\check{\gamma}^{2^n(m)}|\theta) \end{bmatrix}, \quad (71)$$

with diagonal entry $P_{i,i} = p(\check{\gamma}^{i(m)}|\theta)$ for $i = 1, \ldots, 2^n$. Thereafter, we define the block matrix $A \in \mathbb{R}^{2^{n+1} \times 2^{2n}}$,

$$A = \left[\begin{array}{cccc} \mathbb{1}_{2^n}^T & & & \\ & \mathbb{1}_{2^n}^T & & \\ & & \ddots & \\ \multicolumn{3}{c}{\Large 0} & \mathbb{1}_{2^n}^T \\ \hline D & D & D & D \end{array}\right], \tag{72}$$

with $\mathbb{1}_{2^n} = (1, 1, \ldots, 1) \in \mathbb{R}^{2^n}$. In the first 2^n rows of the matrix A, we have in row i, the value 1 from column $i \cdot 2^n$ to column $(i+1) \cdot 2^n$, all other entries are 0. The last 2^n rows consist of a horizontal concatenation of 2^n diagonal matrices, D in (71). We so define the vector $b \in \mathbb{R}^{2^{n+1}}$,

$$b = \left[\begin{array}{c} 1 \\ 1 \\ \vdots \\ 1 \\ \hline p_{\Gamma|\Theta,Y}(\tilde{\gamma}^{1(m)}|\theta, y) \\ p_{\Gamma|\Theta,Y}(\tilde{\gamma}^{2(m)}|\theta, y) \\ \vdots \\ p_{\Gamma|\Theta,Y}(\tilde{\gamma}^{2^n(m)}|\theta, y) \end{array}\right]. \tag{73}$$

i.e. $b_i = 1$ for $i = 1, 2, \ldots, 2^n$ and $b_j = p_{\Gamma|\Theta,Y}(\tilde{\gamma}^{j(m)}|\theta, y)$ for $j = 2^n{+}1, 2^n{+}2, \ldots, 2^{n+1}$. We can then write the optimization problem in (67) on the same form as the general linear programming problem in (12), with x and g as in (70), A as in (72), b as in (73), l being the all-zero vector in $\mathbb{R}^{2^{2n}}$, u being the all-one vector in $\mathbb{R}^{2^{2n}}$.

NTNU

Norwegian University of
Science and Technology