Point Prediction for Streaming Data

Aleena Chanda[1], N. V. Vinodchandran[2], Bertrand Clarke[,1†]

[1]Department of Statistics, U. Nebraska-Lincoln, 340 Hardin Hall North Wing, Lincoln, 68583-0963, NE, USA.
[2]School of Computing, U. Nebraska-Lincoln, Avery Hall, 1144 T St Suite 256, Lincoln, 68508, NE, USA.

Contributing authors: achanda2@huskers.unl.edu; vinod@unl.edu; bclarke3@unl.edu;
[†]Corresponding author; this work was mainly done by the first author under the supervision of the second and third.

Abstract

We present two new techniques for point prediction with streaming data. One uses 'hash' functions and is based on the Count-Min sketch and the other is based on Gaussian process priors with a random bias (GPPRB). These methods are intended for the most general predictive problems where no true model can be assumed to exist for the data stream. In statistical contexts, this is often called the $\mathcal{M}$-open problem class. On the other hand, if a true model exists, our techniques have the usual consistency properties.

We compare our two new methods with three established predictors using cumulative L^1 predictive error. The first of these is based on the Shtarkov solution (often called the normalized maximum likelihood) in the normal experts setting. The other two are Bayesian: one uses basic Gaussian process priors (GPP's), i.e., no additive bias term, and the other is based on Dirichlet process priors (DPP's). For streaming data it is important that predictors be one-pass. For predictors that aren't one-pass we pre-process the data by streaming K-means with a large value of K and use the cluster centers as a finite representative data set.

Preliminary computational work suggests that hash function based methods and (one-pass) GPP methods perform better than Shtarkov predictors and DPP's.

Keywords: Count min sketch, Gaussian processes, Shtarkov solution, streaming K-means, Bayes

1 Prediction with Streaming Data

Consider a string of real numbers, say $y^n = (y_1, y_2, \cdots, y_n, \cdots)$ and suppose our goal is sequential prediction. That is, we want to form a good predictor $\hat{Y} = \hat{Y}_{n+1}(y^n)$ for y_{n+1}. This is often called prediction along a string or streaming data when no assumptions can be made about the distributional properties of the y_i's.

In practice, streaming data means high volume observational data continually and rapidly generated with no meaningful start or end. We have to process the data outcome-by-outcome and can't use 'batch' processing because it takes too long. The result from time n must be available before time $n+1$; there is no time to redo an analysis. The time sensitivity also means we have limited data storage meaning we must discard most of the data. Thus, our analysis must be 'one pass', i.e., we look at the new data point and our accumulated data summary, and then compute our output for the next time step in one running-time bounded procedure.

More formally, these problems are often called $\mathcal{M}$-open as opposed to $\mathcal{M}$-complete (there is a true model but it is not accessible to us) or $\mathcal{M}$-closed (there is a true model and it is accessible to us). There has been an extensive discussion about such problem classes in the statistical literature. The original definitions can be found in [4] but the definitions more commonly used now can be found in [11], with discussion and references. One of the earliest contributions to studying $\mathcal{M}$-open problems, chiefly in the classification context, was [16]; this was prescient because it was published before [4]. Treating $\mathcal{M}$-open problems was a key point in the celebrated book [5], even though these authors did not use that term.

When we say a problem is $\mathcal{M}$-open and there simply is no true model for the data, we are effectively forced into the *prequential* setting of [13]; for a more recent exposition see [32]. When we assess performance here, we use a (prequential) cumulative error based on absolute value i.e., L^1 distance. There seems to be little systematic work on prequential prediction for $\mathcal{M}$-open data even though this is its most important setting.

Here we propose two new forms of predictors $\hat{Y}$ for Y_{n+1}. The first uses y^n to form an estimated empirical distribution function (EEDF) for the empirical distribution (EDF). Our EEDF is based on the Count-Min sketch, see [12], extended to continuous random variables. Count-Min sketch is based on the probabilistic selection of hash functions that we describe in Subsection 2.1. This algorithm is used in streaming data scenarios when you need to efficiently estimate the frequency of elements in a very large or infinite data stream with limited memory. One reason to use this EEDF is that we want to ensure that we can shrink the interval length in the histogram generated from the Count-Min sketch so small that it gives an arbitrarily good approximation of the EDF and hence the DF – if it exists. Since we make the intervals arbitrary small, the number of intervals will be very large and traditional way of keeping all the intervals to estimate the empirical distribution will be inefficient and impractical from a storage point of view. Another reason for our EEDF is that with streaming data we usually impose a storage requirement forcing us to use a 'one-pass' algorithm and our EEDF hash-based predictors (HBP's) satisfy this. The mean and median of our EEDF are the natural HBP's to use. Thus our Count-Min based predictor will outperform the usual EDF predictor when the sample size and number of items in the stream is very large. We note that by construction, estimators from the Count-Min sketch never

underestimate the true frequencies of elements so it favors high frequency elements even though low frequency elements maybe overestimate albeit not by much.

The second predictor we propose is based on Gaussian processes (GP's) that have a random additive bias. It has long been known that the posterior distribution can be regarded as an input-output relation giving a distribution from a specific data set as if it were a deterministic operation, see [9], Sec. 3. This means Bayesian predictors can legitimately be used for streaming data. On the other hand, in $\mathcal{M}$-open problems, we may not be able to identify useful properties of the data generator. So, we want to prevent the posterior distributions from converging and thereby misleading us into believing their limit. Modifying a GP to include a random bias helps ensure that unjustifiable convergence won't occur.

Unfortunately, GPP predictors are not one-pass in general. So, we make them one-pass by pre-processing the data using streaming K-means with a fixed large value of K, $K = 200$, and using the K cluster centers as if they were the data. Streaming K-means is a one-pass procedure that updates the clustering data with each new data point received. Any other streaming convex clustering procedure would give analogous results. To ensure our computational comparisons are fair, we also evaluate our HBP predictors under the same data pre-processing, see Subsec. 5.3.

There are several other existing techniques for prediction in $\mathcal{M}$-open problems. Perhaps the earliest explicitly intended for this case is the Shtarkov solution, see [26]), sometimes called the normalized maximum likelihood. The Shtarkov solution is based on log-loss and requires the analyst to choose a collection of 'experts', essentially parametric families, and tries to match the performance of the best of them. Different Shtarkov solutions result from different choices of experts. Computational and theoretical work on the Shtarkov solution is extensive and often form a very general perspective, see [3] and [34]. Moreover, the log-loss is used to construct Shtarkov solutions via the concept of regret, see [33], even though we use L^1 to measure predictive error here. (This is desirable in the prequential setting.) The specific form of Shtarkov predictor here is the simplest. It is based on normality and is a ratio of Shtarkov solutions. Thus, our Shtarkov predictor *mimics* a conditional density. It is not in fact a conditional density because the Shtarkov solution does not marginalize properly. Nevertheless, the mode of the Shtarkov predictor often performs well.

In addition, in our computational comparisons, we include two other well-known Bayesian predictors, one based on regular GP's i.e., with no bias term, and one based on Dirichlet processes. Being Bayesian, both of these require prior selection and when needed we use an empirical Bayes approach. In general, Bayesian methods assume a stochastic model for the data and are expected to perform best when the model is approximately true but possibly poorly, at least in the parametric case, otherwise. However, we invoke the interpretation of [9] to justify our comparisons. Again, for fairness, we give computational results not only for the regular Shtarkov solution and DPP predictors also for versions of these predictors in which the data is pre-processed using streaming K-means, thereby ensuring both methods have a one-pass version. None of our GPP predictors are one-pass, so we pre-process the data for all of them using streaming K-means. Thus, all the methods we compare here are one-pass and satisfy a storage constraint.

There are two main points to this paper. The first is to propose two more predictors specifically designed for $\mathcal{M}$-open problems, namely HBP's and GPP's with random bias. The second main point is that computational comparisons suggest that HBP and GP based predictors perform better overall than Shtarkov or regular GP predictors. This implication is only observed from computational results shown here for rainfall data and one technology data set; see [6] for other data types. Our finding is tentative because the class of $\mathcal{M}$-problems is very large. On the other hand, we give a heuristic reason for why we observe this, see Sec. 6.

There are many other predictors that we could have included in our comparisons. However, due to space and time constraints, we have not been able to investigate them to the point where we can observe suggestive patterns.

In the next three sections we formally present the predictors we study here and give some of their properties. All are applicable in streaming data settings where no stochastic model can be assumed. In Sec. 2, we define HBP methods and give various properties of them including a sort of consistency, space bound, and 'classical' convergence properties. In Sec. 3, we present our Bayesian predictors. We define standard GPP predictors and extend them to GPPRB predictors, the case that a random additive bias term is included. We also define DPP point predictors. In Sec. 4, we present our Shtarkov based predictors, based on the normalized maximum likelihood in the normal case where explicit expressions can be derived. Then in Sec. 5 we present our computational comparisons. We conclude in Sec. 6 with some general observations.

2 Hash Function Based Predictors

We adapt the Count-Min sketch algorithm so it can be used with real data to estimate an empirical distribution function. The idea is to partition the real data into intervals of equal size over the central domain of Y leaving infinite intervals on each side and then compute the relative frequencies of the intervals as an approximation of a DF (that strictly speaking does not exist). These techniques use 'hash' functions, hence we call them hash-based predictors (HBP's). After defining our predictors, we give bounds on the frequency estimates used to form them, show they satisfy a storage bound, and observe that if a DF exists, they reduce as expected.

2.1 The HBP Method

For the domain $[K] = \{0, 1, 2, \cdots, K, K+1\}$ and the range $[W] = \{0, 1, 2, \cdots, W, W+1\}$, with $K > W$ let $\mathcal{H} \subseteq \{h \mid h : [K] \longrightarrow [W]\}$. The class $\mathcal{H}$ is called a hash family and its elements h are hash functions. The idea is that $h \in \mathcal{H}$ is not one-to-one and so 'hashes' i.e., compresses, $[K]$ to a $[W]$. HBP's use many hash functions and therefore tend to mix the values of K, another sense of the English word 'hash'. Here, we assume $\mathcal{H}$ is equipped with a probability $P_{\mathcal{H}}$ that is '2-universal' meaning that for any $x_1, x_2 \in [K]$ with $x_1 \neq x_2$ we have

$$P_{\mathcal{H}}(H(x_1) = H(x_2)) = \frac{1}{(W+2)},$$

where H is the random variable varying over $h \in \mathcal{H}$. With some loss in generality, we take $P_{\mathcal{H}}$ to be uniform over $\mathcal{H}$.

Next consider the range $[-M, M]$ for some real $M > 0$ and fix $K \in \mathbb{N}$. Now, partition $[-M, M]$ uniformly into K intervals each of length $2M/K$ and denote the k^{th} interval by $I_k = I_{Kk}$, for $k = 1, 2, \cdots, K$. That is,

$$I_k = I_{Kk} = \left(-M + (k-1)\frac{2M}{K}, -M + k\frac{2M}{K}\right]. \tag{2.1}$$

Also, let $I_0 = (-\infty, -M)$ and $I_{K+1} = (M, \infty)$. In practice, if the stream y^n is bounded e.g., $M_1 \leq y_i \leq M_2$, it is convenient to modify these definitions so the intervals only cover $[M_1, M_2]$. Indeed, in our computations, we take $M_1 = 0$, fix an upper bound M_2 and can ignore I_0 and I_{K+1}. To link the y_i's to the I_k's, let

$$a_k = a_{Kk}(n) = \#\{y_i \in I_k \mid i = 1, \cdots, n\}.$$

That is, $a_k(n)$ is the frequency of items in the stream $(y_1, \cdots, y_n)$ that fall in I_k.

Let $h_1, \cdots, h_{d_K}$ be d_K randomly chosen hash functions where the domain is $[K]$ and the range is $[W_K]$. That is, $\forall j = 1, \cdots, d_K$; $h_j : \{1, \cdots, K\} \longrightarrow \{1, \cdots, W_K\}$. Here d_K and W_K are parameters that can be chosen by the user. For effective data compression, or space efficiency, $W_K << K$. We extend our discrete hash functions to $\mathbb{R}$ by setting

$$\tilde{h}_j : \mathbb{R} \longrightarrow \{0, 1, 2, \cdots, W_K, W_K + 1\}$$

where $\tilde{h}_j(s) = h_j(k)$ for $s \in I_k$ and $k = 0, 1, \ldots, K, K+1$. Thus, for $y_i \in I_k$ we have

$$\tilde{h}_j(y_i) = h_j(k).$$

Let I_{kjl} indicate when the j-th hash function makes an error, i.e., h_j assigns the same value to two different elements k, ℓ of its domain I_k. That is, set

$$I_{kjl} = \begin{cases} 1, & \text{if } h_j(k) = h_j(l); k \neq l \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

We tolerate this (small) error for the sake of compression. Next, we extend (2.2) to the interval case by writing

$$I_{k,j,\ell} = \begin{cases} 1, & \text{if } \tilde{h}_j(s_1) = \tilde{h}_j(s_2); s_1 \nsim s_2 \\ 0, & \text{otherwise,} \end{cases} \tag{2.3}$$

where $s_1 \in I_k$, $s_2 \in I_\ell$ and $\nsim$ means that $k \neq \ell$. We link the extent to which h_j is not one-to-one with the occurrence of y_i in the intervals by defining

$$X_{kj}(n) = \sum_{l=0}^{K+1} I_{k,j,\ell} a_l(n) \geq 0. \tag{2.4}$$

Thus, $X_{kj}(n)$ is the number of y_i's in the stream that are not in I_k but still give $h_j(k)$, i.e., $y_i \notin I_k$ but $\tilde{h}_j(y_i) = h_j(k)$, because the h_j's are not one-to-one.

We next define an estimate of a_k (frequency of the k^{th} interval) denoted by $\hat{a}_k$, at time n. For the j^{th} hash function h_j, an interval k and time n, define

$$\mathsf{count}_n(j, h_j(k)) = \#\{i \leq n | \ \tilde{h}_j(y_i) = h_j(k)\}.$$

For the j^{th} hash function let $\hat{a}_{jk}(n) = \mathsf{count}_n(j, h_j(k))$. Then the estimate $\hat{a}_k$ of a_k is

$$\hat{a}_k(n) = \min_j \ \hat{a}_{jk}(n) \geq 0. \tag{2.5}$$

To predict y_{n+1}, we use two HBP's. These are essentially weighted means and medians which we define for the sake of being explicit. Given y^n, we set $\hat{Y}_{n+1}$ to be:

- the weighted mean of the midpoints of the intervals $I_k; k = 1, 2, \cdots, K$ defined in (2.1), where the weights are $\hat{a}_k$ as defined in (2.5). Let the mid-point interval I_k be m_k for $k = 1, \ldots, K$. Then,

$$\hat{y}_{n+1} = \hat{y}_{K,n+1} = \sum_{k=1}^{K} m_k \frac{\hat{a}_k(n)}{n}; \tag{2.6}$$

- the weighted median from the intervals I_k for $k = 1, \ldots, K$ with weights $W_k = \frac{\hat{a}_k}{\sum_{k=1}^{K} \hat{a}_k}$ is defined as the average of m_q and m_{q+1}, where m_q satisfies

$$\sum_{k=1}^{q} W_k \leq \frac{1}{2} \text{ and } \sum_{k=q+1}^{K} W_k \geq \frac{1}{2}. \tag{2.7}$$

These definitions ignore I_0 and I_{K+1}; it is assumed that M is chosen large enough so that a_0 and a_{K+1} are small enough in practice that the weighted means and medians are representative of the stream.

2.2 A Few Key Properties

We define an EEDF $\hat{F}_n$ using the normalized counts from applying our continuous extension of the Count-Min sketch to the intervals I_k. In this subsection, we give several important properties of this EEDF. Arguably, the main novelty of our convergence results is that the mode of convergence is defined by $P_{\mathcal{H}}$, a distribution on the random selection H of the hash functions $h \in \mathcal{H}$. This preserves the assumption that y^n does not have a distribution and hence remains $\mathcal{M}$-open. For one result, we invoke a distribution on the data (violating the $\mathcal{M}$-open assumption). We indicate this case clearly and regard the result as 'counterfactual' in that it shows our methods reduce as they should under the usual assumptions.

Our first result is that $\hat{a}_k$ from (2.5) estimates the actual frequency a_k of an interval I_k well, asymptotically, for any fixed $k = 0, \ldots, K+1$. Let $||a||_1 = \sum_{k=0}^{K+1} a_k(n) = n$

be the sum over k of the number of elements y_i up to time n that land in I_k; K and n are suppressed in the notation $||a||_1$ for brevity. Recall that d_K is the number of hash functions randomly chosen at the K-th stage. We have the following.

Theorem 1. *Let $\epsilon, \delta > 0$ and K be fixed. Then, if $W_K \geq \lceil \frac{e}{\epsilon} \rceil$ and $d_K \geq \lceil \log(1/\delta) \rceil$ we have that*

$$P(\forall j = 1, \cdots, d_K; \hat{a}_{jk}(n) \leq a_k(n) + \epsilon ||a||_1) \leq \delta.$$

Proof. The proof in [23] extends readily to our continuous case here; see also [7]. Here log has base e. □

Recall that, by construction, $\hat{a}_k \geq a_k$ so a lower bound is automatic.

Next, we address the storage requirement for the procedure used in Theorem 1. Heuristically, observe that the storage is upper bounded by the number of hash functions $\lceil \log(1/\delta) \rceil$ multiplied by the number of values each hash function can take, namely $\lceil e/\epsilon \rceil$ giving $\mathcal{O}((1/\epsilon) \log(1/\delta))$. In fact, following [23], $\mathcal{O}(1/\epsilon)$ will suffice.

Theorem 2. *Let $\epsilon, \delta > 0$ be given. Then, if the storage available is $\Omega(1/\epsilon)$[1], we still obtain the conclusion of Theorem 1, in particular*

$$P(\forall j : \hat{a}_{jk} \leq a_k + \epsilon ||a||_1) \leq \delta.$$

Proof. The proof in [23] extends readily to our continuous case here; see also [7]. □

Separate from Theorems 1 and 2, we establish the usual statistical convergence properties for our EEDF. The mode of convergence is in the joint probability of the hash functions and the data. Because we are using a distribution on the data, we are automatically not in a $\mathcal{M}$-open setting. Essentially, we are showing that our EEDF $\hat{F}_n$ reduces to the usual EDF $\hat{F}$ and converges to the 'true' DF F asymptotically. To get these statements, we let $K, d_K, n \to \infty$ at appropriate rates. We focus on the convergence of the EEDF to the EDF so F is only used for the mode of convergence.

Theorem 3. *Let $y_i \in \mathbb{R}$. Then, for any given $\epsilon > 0$ and $\delta > 0$ we have*

$$P(|\hat{F}_n(y_i) - F_n(y_i)| > \epsilon) \leq \delta. \tag{2.8}$$

for K, d_K, W_K, and n large enough, where the EEDF is $\hat{F}_n(y_i) = \sum_{k \leq y_i} \frac{\hat{a}_k(n)}{n}$ and the EDF is $F_n(y_i) = \sum_{k \leq y_i} \frac{a_k(n)}{n}$.

Proof. This follows from a routine modification of the proof of Theorem 3 in [7]. □

Remark: Because we are using fixed δ and ϵ in this result, it is sufficient for K, d_K, W_K, and n to be large enough.

Under similar conditions, we can show a Glivenko-Cantelli theorem for the EEDF to converge to the EDF and the DF (if it exists), see [7] and [25] for details. It is essential to remember that the randomness in $\hat{F}_n(y_i)$ does not come from the data points y except when we assume a distribution on y. In fact, the randomness in $\hat{F}_n(y_i)$ only comes from the random selection of hash functions via the $\hat{a}_k$'s. Strictly in the

[1] Ω-notation gives a lower bound in contrast to big-$\mathcal{O}$ notation that gives an upper bound.

'FWIW' category, the main implication from Theorem 3 is that in principle we can obtain asymptotically valid prediction intervals, not just point predictors, from an the EEDF (or EDF), in the $\mathcal{M}$-closed and -complete cases.

A useful property of the EEDF is that it can track the location of the data. For example, if the data is located around zero initially but drifts higher the EEDF, like the EDF, will shift higher. In this sense, the EEDF is adaptive.

3 Bayesian Predictors

In this section we define three Bayesian predictors. The first is the usual Gaussian Process Prior predictor. The second is an extension of this to include a random additive bias. The third is the usual Dirichlet Process Prior predictor, essentially the Bayesian's histogram possibly mimicking the EDF or EEDF. Predictive distributions are well-known for the first and third of these; we review them here for the sake of completeness. We provide full details for the second since it seems to be new. Recall that these must be seen as predictors only; the data being $\mathcal{M}$-open means that modeling e.g., by the convergence of a Bayes model, would be a contradiction.

3.1 No Bias

We assume $Y_i = f_i + \epsilon_i, \quad i = 1, \cdots, n$ where the i^{th} data point y_i is distributed according to Y_i and $f = (f_1, f_2, \cdots, f_n)^T$ is equipped with a Gaussian process prior. That is, $f \sim \mathcal{N}(a, \sigma^2 K_{11})$, where, $a = (a_1, a_2, \cdots, a_n)^T$ is the mean and $K_{11} = \left(\left(k_{ij}\right)\right); i, j = 1, \cdots, n$ is the covariance function in which $k_{ij} = k_{ij}(y_i, y_j)$. First, we assume there is no bias i.e., $a_i = 0$ for all i, so the joint distribution of $Y = (Y_1, Y_2, \cdots, Y_n)^T$ and Y_{n+1} is

$$\begin{pmatrix} Y \\ \vdots \\ Y_{n+1} \end{pmatrix} = \begin{pmatrix} f \\ \vdots \\ f_{n+1} \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_{n+1} \end{pmatrix}$$
$$\sim \mathcal{N}\left[\begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}, \sigma^2 \begin{pmatrix} K_{11} + I & \vdots & K_{12} \\ \cdots & \vdots & \cdots \\ K_{21} & \vdots & K_{22} + 1 \end{pmatrix} \right] \tag{3.1}$$

where $K_{12} = (k_{1,n+1}, k_{2,n+1}, \cdots, k_{n,n})^T$ and $K_{21} = K_{12}^T$, $K_{22} = k_{n+1,n+1}$. More compactly, we write

$$Y^{n+1} \sim \mathcal{N}(0^{n+1}, \sigma^2 (I + K)_{n+1 \times n+1}). \tag{3.2}$$

It is well known that the predictive distribution of Y_{n+1} given Y is

$$Y_{n+1}|Y \sim \mathcal{N}(\mu^*, \Sigma^*)$$

where

$$\mu^* = \sigma^2 K_{12}\{\sigma^2(K_{11}+I)\}^{-1}y = K_{12}\{(K_{11}+I)\}^{-1}y \tag{3.3}$$

and

$$\begin{aligned}\Sigma^* &= \sigma^2(K_{22}+1) - K_{21}\{\sigma^2(K_{11}+I)\}^{-1}\sigma^2 K_{12} \\ &= \sigma^2(K_{22}+1) - K_{21}(K_{11}+I)^{-1}K_{12}\end{aligned} \tag{3.4}$$

Hence, in the zero bias case, our optimal point predictor (under squared error loss for instance) is simply the conditional mean μ^* in (3.3).

To complete the specification it remains to estimate σ^2 for use in (3.4). In the general case, we have $Y \sim \mathcal{N}(a, \sigma^2(I+K)_{n\times n})$. Hence, $(I+K)^{\frac{1}{2}}Y \sim \mathcal{N}(a, \sigma^2 I)$. Letting $Y' = (I+K)^{\frac{1}{2}}Y$ and $S_k = \frac{1}{n-1}\sum_{i=1}^{n}(y_i' - \bar{y}')^k$ we can estimate σ^2 by S_2. Note that σ^2 cancels out in (3.3) and since we are only looking at point prediction in our computations below we do not have to use (3.4).

3.2 Random Additive Bias

Consider a Gaussian process prior in which the bias $a = (a_1, \ldots, a_n)^T$ is random. That is, when we estimate function value – an f_i for $i \leq n$ – the prior adds a small amount of bias effectively enlarging the range of the estimate. For the prediction of f_{n+1} a similar sort of widening happens. To see this, write

$$a \sim \mathcal{N}(\gamma \mathbf{1}_n, \sigma^2\delta^2 I_{n\times n}) \tag{3.5}$$

where the expected bias is $\gamma \in \mathbb{R}$ and $\sigma^2 > 0$ has distribution

$$\sigma^2 \sim \mathcal{IG}(\alpha, \beta). \tag{3.6}$$

Here, α, β, and δ are strictly positive, and, like γ are unknown. Expression (3.5) means that, with some loss of generality, the biases are independent, identical, symmetric, unimodal, and have light tails. Since

$$Y \sim \mathcal{N}(a, \sigma^2(I_{n\times n} + K_{n\times n})), \tag{3.7}$$

its likelihood is

$$\begin{aligned}\mathcal{L}_1(a, \sigma^2|y) &= \mathcal{N}(a, \sigma^2(I_{n\times n} + K_{n\times n}))(y) \\ &= \frac{e^{-\frac{1}{2\sigma^2}(y-a)'(I_{n\times n}+K_{n\times n})^{-1}(y-a)}}{(2\pi)^{\frac{n}{2}}(\sigma^2)^{\frac{n}{2}}|I_{n\times n}+K_{n\times n}|^{\frac{1}{2}}}\end{aligned} \tag{3.8}$$

and the joint prior for (a, σ^2) is

$$w(a, \sigma^2) = \mathcal{N}(\gamma 1, \sigma^2\delta^2 I_{n\times n})\ \mathcal{IG}(\alpha, \beta)$$

$$= \frac{e^{-\frac{1}{2\sigma^2}(a-\gamma 1)'(\delta^2 I_{n\times n})^{-1}(a-\gamma 1)}}{(2\pi)^{\frac{n}{2}}(\sigma^2\delta^2)^{\frac{n}{2}}}\frac{\beta^\alpha}{\Gamma(\alpha)} \times \left(\frac{1}{\sigma^2}\right)^{\alpha+1} e^{-\frac{\beta}{\sigma^2}}. \tag{3.9}$$

Our first result is the identification of the posterior predictive density for Y_{n+1} given y^n. We have the following.

Theorem 4. *The posterior predictive distribution of the future observation y_{n+1} given the past observations y^n is*

$$m(y_{n+1}|y^n) = St_{2\alpha+n}\left(A_1, \frac{\beta^{**}}{\frac{2\alpha+n}{2}}\right)(y_{n+1}), \tag{3.10}$$

where $St_v(\theta, \Sigma)$ denotes the Student's t distribution with v degrees of freedom with parameters θ and Σ. In (3.10), $\beta^{**} = \beta + A_2$ and $A_1 = \frac{\gamma_2 - y'^n g_1^n}{\gamma_1}$. Expressions for g_1^n, γ_1, γ_2, and A_2 are given in the proof and can be explicitly written as functions of the variance matrix $K_{n+1\times n+1}$, y^n, γ, and δ.

Proof. A complete proof is given in the Appendix, Sec. 7. □

This result identifies A_1 as the optimal point predictor. However, to use it we must find many parameters and hyperparameters. Specifically we require values for γ_1, γ_2, and g_1^n as well as values for α, β, δ and for the bias γ.

First, to find γ_1 and g_1^n, we used (7.22) in the Appendix. For γ_2 we used (7.23), also in the Appendix.

Then, for α, β, δ and γ we started with α and β. Recall (3.6) and define $S_2' = \frac{1}{n-1}\sum_{i=1}^n (y_i' - \bar{y'})^k$. We can show that

$$\sigma^2 = E\left(\frac{S_2'}{1+\delta^2}\right).$$

So, for a given $\delta > 0$ (discussed below), we use

$$\hat{\sigma}^2 = \frac{S_2'}{1+\delta^2}. \tag{3.11}$$

Since we need to estimate two parameters, α and β, we use the second moment of $\hat{\sigma}^2$ as well. We can approximate $Var(\hat{\sigma}^2)$ as

$$\widehat{Var}(\hat{\sigma}^2) = \frac{2\hat{\sigma}^2}{(n-1)^2}\left[\hat{\sigma}^2 + \frac{2n\gamma^2}{1+\delta^2}\right].$$

For an inverse gamma we have

$$\hat{\sigma}^2 \approx E(\sigma^2) = \frac{\beta}{\alpha - 1} \tag{3.12}$$

for the mean of σ^2 and

$$\widehat{Var}(\hat{\sigma}^2) \approx Var(\sigma^2) = \frac{\beta^2}{(\alpha-1)^2(\alpha-2)} \tag{3.13}$$

for the variance of σ^2. Now, we can solve solve for α and β from (3.12) and (3.13) and invoke the method of moments to find

$$\hat{\alpha} \approx \frac{E^2(\hat{\sigma}^2)}{Var(\hat{\sigma}^2)} + 2 \tag{3.14}$$

and

$$\hat{\beta} \approx E(\hat{\sigma}^2)(\hat{\alpha}-1). \tag{3.15}$$

Next, we obtain serviceable estimates of γ and δ^2. Start by forming the likelihood $\mathcal{L}_3(y|\gamma,\delta^2,\sigma^2)$ by integrating a out from the product of (3.5) and (3.8). Then, in principle, this likelihood can be maximized to find $\hat{\gamma}$ and $\hat{\delta}$. To effect this, we state a result that gives the forms of the likelihood we want to maximise. We write it in two different ways so the optimization will be clear. We also use this result to estimate the parameters in the location and scale of the predictive distribution in Theorem 4.

Theorem 5. *The likelihood of y^n given γ, δ^2 and σ^2, marginalizing out a, can be written in two equivalent forms:*
Clause I:

$$\begin{aligned}\mathcal{L}_2(y^n|\gamma,\delta^2,\sigma^2) = h(\gamma)&\frac{|V_{n\times n}|^{\frac{1}{2}}}{(2\pi\sigma^2\delta^2)^{\frac{n}{2}}|(I+K)_{n\times n}|^{\frac{1}{2}}} \\ &\times e^{-\frac{1}{2\sigma^2}\left[{y'}^n\left\{(I+K)^{-1}_{n\times n}+(I+K)^{-1}_{n\times n}V_{n\times n}(I+K)^{-1}_{n\times n}\right\}y^n\right]},\end{aligned} \tag{3.16}$$

where

$$h(\gamma) = e^{-\frac{1}{2\sigma^2}\left[-2\gamma{y'}^n\frac{(I+K)^{-1}_{n\times n}V_{n\times n}}{\delta^2}1+\gamma^2 1'\left(\frac{I}{\delta^2}-\frac{V_{n\times n}}{\delta^4}\right)1^n\right]}. \tag{3.17}$$

and Clause II:

$$\mathcal{L}_2(y^n|\gamma,\delta^2,\sigma^2) = g(\delta^2)\times\frac{1}{(2\pi\sigma^2)^{\frac{n}{2}}|I+K|^{\frac{1}{2}}}e^{-\frac{1}{2\sigma^2}[{y'}^n(I+K)^{-1}_{n\times n}y^n]}, \tag{3.18}$$

where

$$g(\delta^2) = \frac{\left|\left\{(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}\right\}\right|^{\frac{1}{2}}}{(\delta^2)^{\frac{n}{2}}}$$

$$\times e^{\frac{1}{2\sigma^2}\left[y'^n(I+K)^{-1}_{n\times n}\left\{(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}\right\}^{-1}(I+K)^{-1}_{n\times n}y^n\right]}$$
$$\times e^{\frac{1}{2\sigma^2}\left[\frac{2\gamma}{\delta^2}y'^n(I+K)^{-1}_{n\times n}\left\{(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}\right\}^{-1}1^n\right]}$$
$$\times e^{\frac{1}{2\sigma^2}\left[\frac{\gamma^2}{\delta^4}1'^n\left\{(I+K)^{-1}+(\delta^2 I_{n\times n})^{-1}\right\}^{-1}1-\frac{\gamma^2}{\delta^2}1'^n1^n\right]}. \tag{3.19}$$

Remark: Clause I lets us find the maximum likelihood estimator $\hat{\gamma}_{MLE}$ by looking only at $h(\gamma)$ while Clause II lets us find $\hat{\delta}_{MLE}$ by looking only at $g(\delta^2)$.

Proof. A complete proof can be found in [7]. □

To use Theorem 5 to find estimates of γ and δ^2 we start with γ. Taking logarithms on both sides of (3.17), differentiating w.r.t. γ, and equating the derivative to zero gives:

$$\hat{\gamma} = \frac{y'^n(I+K)^{-1}_{n\times n}V_{n\times n}1^n}{1'^n\left(I_{n\times n}-\frac{V_{n\times n}}{\delta^2}\right)1^n}, \tag{3.20}$$

in which it is seen that σ^2 does not appear.

The second derivative is

$$\frac{d^2\ln h(\gamma)}{d\gamma^2} = -\frac{1}{\sigma^2\delta^2}1'^n\left(I_{n\times n}-\frac{V_{n\times n}}{\delta^2}\right)1^n$$

which is typically less than 0 because, as we will see, δ^2 is usually small. Hence, our solution $\hat{\gamma}$ to (3.20) will typically be a local maximum.

Next, we use (3.19) to help find a good estimate of δ. Unfortunately, we cannot simply differentiate $g(\delta^2)$, set the derivative to zero, and solve. The resulting equations are just too complicated to be useful in any obvious way. So, we did a grid search over interval $\mathbf{I} \subset \mathbb{R}^+$ to maximize g. In computational work not shown here, we found that the optimal $\delta \in \mathbf{I}$ was almost always the left hand end point, even as $\mathbf{I}$ moved closer and closer to 0. In the limit of $\delta \to 0$, $\hat{\gamma} \to 0$ as well. We interpret this to mean that the optimal value of the mean and variance of the bias a are zero i.e., there is no bias. Hence, for the method to be nontrivial, we always pragmatically set δ to be small so that in our computations the bias would not overwhelm the data. For instance, we typically set $\delta = .1$ and tested larger values up to $\delta = 1$. When we recomputed with larger values we typically found that the predictive error increased very slowly.

3.3 Dirichlet Process prior prediction

Suppose a discrete prior G is distributed according to a Dirichlet Process and write $G \sim DP(\alpha, G_0)$ where α is the mass parameter and G_0 is the base measure with $\mathbb{E}(F) = G_0$. Then, by construction, we have the following; see [17].If the sample space $\mathbb{R}$ is partitioned into $A_1, A_2, \cdots, A_k$, then the random vector of probabilities $(G(A_1), G(A_2), \cdots, G(A_k))$ follows a Dirichlet distribution, i.e.,

$p(G(A_1), G(A_2), \cdots, G(A_k)) \sim Dirichlet(\alpha(A_1), \alpha(A_2), \cdots, \alpha(A_k))$, where $\alpha(\mathbb{R}) = M$, which we take here to be one.

Now, the posterior distribution of $G(A_1), G(A_2), \cdots, G(A_k)|Y_1, Y_2, \cdots, Y_n$ is also Dirichlet but with parameters $\alpha(A_j)+n_j$ where, $n_j = \sum_{i=1}^{n} I(Y_i \in A_j); j = 1, 2, \ldots, k$. If $Y_j^{'}; j = 1, 2, \cdots, k$ are the distinct observations in $\{Y_i; i = 1, 2, \cdots, n\}$, the posterior predictive distribution of $Y_{n+1}|Y_1, Y_2, \cdots, Y_n$ is

$$Y_{n+1}|Y_1, Y_2, \cdots, Y_n = \begin{cases} \delta_{Y_j'}, \text{with probability } \frac{n_j}{M+n}; j = 1, 2, \cdots, \text{k; and} \\ G_0, \text{with probability } \frac{M}{M+n} \end{cases}.$$

Thus, our Dirichlet Process Prior (DPP) predictor is

$$\hat{Y}_{n+1} = \sum_{j=1}^{k} y_j' \frac{n_j}{M+n} + \frac{M}{M+n} median(G_0). \tag{3.21}$$

4 Shtarkov Solution Based Predictors

We distinguish between the Shtarkov *solution* that solves an optimization problem (giving the normalized maximum likelihood estimator as the minimax regret solution) and the Shtarkov *predictor* that is the ratio of two Shtarkov solutions. The latter can be derived explicitly for the normal case when the variance is known and we use it as one of our predictors.

Here, for the sake of completeness, we give the Shtarkov solution and predictors in general. Then we look at special cases to present the predictor we actually use in our computational comparisons.

4.1 The Shtarkov solution

Consider a game being played between Nature N and a Player P. P has access to experts indexed by $\theta \in \Theta \subset \mathbb{R}^k$. The goal of P is to make the best prediction of the value that N issues. Let us consider the univariate case. Suppose P can call on experts and they provide their best predictive distributions $p(\cdot|\theta)$. After receiving these, P announces the prediction $q(\cdot)$. In practice, P might choose $q(\cdot)$ to match the performance of the best expert θ.

Assume the y_i's are from a univariate data stream $y_1, y_2, \ldots$ issued by N. N can issue y_i's by any rule s/he wants or, here, by no rule, probabilistic or otherwise, since we are regarding the y_i's as $\mathcal{M}$-open. Regardless of how N generates data, after the n^{th} step, $P's$ cumulative regret with respect to expert θ is given by

$$\log \frac{1}{q(y^n)} - \log \frac{1}{p(y^n|\theta)} = \log \frac{p(y^n|\theta)}{q(y^n)}. \tag{4.1}$$

If P wants to minimize the maximum regret, s/he chooses

$$q_{\mathsf{opt}}(y^n) = \arg\min_q \sup_{y^n} \sup_\theta \log \frac{p(y^n|\theta)}{q(y^n)} = \frac{p(y^n|\hat{\theta})}{\int p(y^n|\hat{\theta})dy^n} \tag{4.2}$$

where $\hat{\theta} = \hat{\theta}(y^n)$ where $\hat{\theta}$ is the maximum likelihood estimator, provided the integral exists; see [26] and [24]. The normalized maximum likelihood q_{opt} is called the (frequentist) Shtarkov solution. If a weighting function $w(\theta)$ across experts is given then the Bayesian form of (4.2) is

$$q_{\mathsf{opt,B}}(y^n) = \frac{w(\tilde{\theta}(y^n))p(y^n|\tilde{\theta}(y^n))}{\int w(\tilde{\theta}(y^n))p(y^n|\tilde{\theta}(y^n))dy^n} \tag{4.3}$$

where $\tilde{\theta}$ is the posterior mode.

4.2 The Shtarkov Predictors

We take as our frequentist Shtarkov point predictor the mode of

$$q_{\mathsf{Sht}}(y_{n+1}) = q_{\mathsf{Sht}}(y_{n+1}; y^n) = \frac{\mathsf{q_{opt}(y^{n+1})}}{q_{\mathsf{opt}}(y^n)}, \tag{4.4}$$

the ratio of two Shtarkov solutions. The analogous ratio denoted $q_{\mathsf{Sht,B}}(y_{n+1})$ using (4.3) gives the Bayes Shtarkov point predictor. Expression (4.4) looks like a conditional density but in fact is just a distribution. Here, we use the mode of the numerator over y_{n+1} given that y^n is fixed. The mode turns out to be a good predictor – better than the mean or median because often q_{Sht} is often highly skewed (see [22]). Heuristically, q_{Sht} can be regarded as an approximation to a conditional density for y_{n+1}, i.e., $q(y_{n+1}|y^n)$ if it were to exist. However, this cannot be formalized because Shtarkov solutions do not marginalize and hence do not form a stochastic process.

On the other hand, the Bayes Shtarkov solutions is arguably close to the (conditional) predictive distribution $m(y_{n+1}|y^n) = m(y^{n+1})/m(y^n)$ where $m(y^n) = \int w(\theta)p(y^n|\theta)\mathrm{d}\theta$; see [10]. For discrete θ this means that the sequential Bayes Shtarkov solution can be regarded as an example of the Multiplicative Weights Updates algorithm; see [1] for a review of these methods.

4.3 The Normal Example

Shtarkov solutions and predictors can be found for many settings; see [6], [3], and [28]. Here, for comparison purposes, we only note and use the (simplest) predictor when the experts follow a normal $N(\mu, \sigma^2)$ distribution in the frequentist cases of i) μ unknown and σ known, and ii) both μ and σ unknown.

Given data $y_1, y_2, \ldots$, write $\bar{y} = \bar{y}_n$ for the sample mean from the first n observations. The frequentist Shtarkov solution for y^n is the normalized version of the maximum likelihood for the normal mean problem with known and unknown variance. In both cases, the normalized likelihood is maximized at the predictor $\hat{y}_{n+1} = \bar{y}_n$,

independently of σ. We use this in our computations in Sec. 5. Shtarkov point predictors, Bayes and frequentist, can be found in many other exponential families cases, but not in general in closed form; see [6]. On the other hand, we conjecture that other parametric examples will have performance similar to the normal case because the predictors they generate are analogous.

5 Computational comparisons

To present our computational results we begin by listing our predictors. Then we describe the settings for our comparisons. Finally, we present our computations and interpret what they imply about the methods.

5.1 Our predictors

We computationally compare the predictors that have been presented in the earlier sections. There were two predictors based on hash functions. These HPB's used the mean and the median of the empirical DF generated by the Count-Min sketch. They were explicitly given by (2.6) and (2.7) in Sec. 2. There were three Bayesian predictors namely GPP's with no bias, GPP's with a random additive bias, and DPP's. They were given in (3.3), Theorem 4, and (3.21). The predictor in (3.3) requires the estimation of parameters as discussed in Subsec. 3.1. The estimation of parameters required to use GPP's with a random additive bias (A_1 from Theorem 4) is discussed in Subsec. 3.2. For the DPP predictor in (3.21) the 'parameter' G_0 has to be chosen and is user dependent. Finally, we used one frequentist Shtarkov point predictor based on normality. It was simply the mean, as derived in Subsec. 4.3.

5.2 Settings for the comparisons

We compare point predictors by their cumulative L^1 error. That is, for each method, we have a sequence of errors $|y_i - \hat{y}_i|$ where $\hat{y}_i$ depends on $y_1, \ldots, y_{i-1}$ (and possibly a burn-in set $\mathcal{D}_b$) and we find the cumulative predictive error

$$CPE = CPE(n) = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|. \tag{5.1}$$

It is seen that

$$CPE(n+1) = \frac{1}{n+1} \left(nCPE(n) + |y_{n+1} - \hat{y}_{n+1}| \right)$$

so it is easy to update the CPE from time step to time step. For each method, we report the final CPE.

Since we are using HBP's, it is natural to exploit the fact they can be computed in one pass. We can do this readily for the Shtarkov predictor and the DPP predictor. However, it is difficult to do this for either of the GPP predictors because the variance matrix increases in dimension. In particular, Gaussian process priors we use here would

have an $n \times n$ variance matrix for prediction at the $(n+1)$th stage, which is infeasible to store for large n.

So, to include GPP's in our comparisons we have to ensure the variance matrices in the GPP predictors do not increase in size excessively. We do this by using what we call a representative subset of fixed size that is updated from time step to time step. Essentially, we use the cluster centers from streaming K-means for a fixed choice of K, here $K = 200$. Under streaming K-means, the cluster centers at time step n update easily to give the cluster centers at time step $n+1$. We then use the cluster centers at time n as the data to form our predictors for time $n+1$. That is, we used streaming K means for fixed K to pre-process the data to avoid storage problems and ensure the GPP predictorss are one pass.

Thus, we have two sets of comparisons of CPE's, one for the four methods that can be implemented in one pass and another for all six methods using streaming representative subsets. In fact, we compare all of them in an effort to understand how the various methods behave.

We use different forms of the three predictors for different data sets. However, the quantities that must be chosen are the same in all cases. For the HPB methods (mean and median), we must choose K, d_K, and W_K. For the Bayesian methods our choices are as follows. For GPP, we chose the variance matrix $K_{n\times n}$ to be of the form of a correlation matrix for an $AR(1)$ time series. That is, for given correlation ρ we used

$$K_{n\times n} = \begin{pmatrix} 1 & \rho & \cdots & \rho^{n-1} \\ \rho & 1 & \cdots & \rho^{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ \rho^{n-1} & \rho^{n-2} & \cdots & 1 \end{pmatrix}$$

and set $\rho = .8$. For the GPP with random bias, we used A_1 as our point predictor and so only had to find values for g_1^n, γ_1, γ_2, δ and σ. We listed our choices at the end of Subsec. 3.2. For DPP's we chose the base measure G_0 to be a Discrete Uniform Distribution on the range $[\min\{y_1, \ldots, y_n\}, \max\{y_1, \ldots, y_n\}]$. For the Shtarkov predictor in the normal case, we got an expression involving data only. The predictor was fully specified once the family of experts and weighting had been fixed.

To finish the general specifications of our predictors, we initialized all our sequences of predictions using 10% of the total data we intended to use. Thus, in the Colombia rainfall data below where we had $n = 5000$, we used a burn-in of the first 500 data points to form the predictions for each of the 501 time step. These then gave us the first terms in our CPE's for the ten methods.

5.3 Results

In this subsection we use the ten methods described in Subsec. 5.1 on four real data sets. The first three are rainfall data sets from three jurisdictions here called Colombia, India, and Bhubhneshwar. Note that because our methods are designed for $\mathcal{M}$-open problems, simulated data will not typically be complex enough. Indeed, in other computational work not shown here using $\mathcal{M}$-complete and $\mathcal{M}$-closed data we found a very

different ordering of techniques by performance: the techniques designed for streaming data performed relatively poorly.

Our first computational example uses the Colombia data set that can be found at https://data.world/hdx/f402d5ef-4a74-4036-8829-f04d6f38c8e9. The dataset contains daily values of total precipitation (mm) in Colombia over a period of four years ending in the year 2013. They were collected from 27 different base stations and the 'time' index cycles through them. We suggest this cycling will be typical of many kinds of streaming data. We use the first 5000 rows of the value column of the dataset. This data set, like India below, is not a pure time series – it's as if there were a mixing distribution over the stations. However, there is a pattern that would allow prediction so this is a fair test of how well a predictor can perform on complex data.

Our second data set – India – is similar and can be found at https://data.world/hdx/687c4f99-6ec6-4b30-ada2-a5a0f9eac629. Like Colombia, this dataset contains values of daily total precipitation (mm) cycling over 76 different base stations. Measurements of total precipitation for a two year period (2010-2011) can be found in the dataset. Again we use the first 5000 rows of the value column of the dataset For the HBP's computations with Colombia and India, $d_K = 10$, $W_K = 50$, and $K = 100$. We set $K = \lceil n/50 \rceil$ in all cases.

Our third data set – Bhubhneshwar – can be found at https://www.kaggle.com/datasets/vanvalkenberg/historicalweatherdataforindiancities. It has daily precipitation data (mm) from 01/01/1990 to 07/20/2022. The column prcp was used for getting the values of CPE. Rows with missing values were deleted leaving 6838 data points. For the computations with Bhubhneshwar, $d_K = 15$, $W_K = 50$, $K = 137$; these were larger than for Colombia and India because we used a larger sample size.

The fourth data set is drawn from the phones accelerometer benchmark data that can be found at [31], which provides a complete description. We extracted the first 10,000 rows of the data set and used the column "y" for our computing. We split the data into four parts, i.e., sets of 2500 each, and computed the results. For the HBP computations with accelerometer, $d_K = 10$, $W_K = 20$, and $K = 50$.

In our tables, we follow the convention that the numbers in **bold** denote the smallest errors and the asterisk (*) represents the second best. Headings indicate whether the error in a column is from a one-pass method or used a representative subset from K-means. We abbreviate the names of our methods as Sht, DPP, GPP(RB) and GPP(no RB) to mean the Shtarkov (Normal), Dirichlet process prior, and GPP with and without random bias, respectively.

Turning to the numerical results, we begin with the CPE's for Colombia given in Table 1. In this case we see exactly the pattern of errors that we expect. Namely, the one-pass HBP median has the lowest error and GPP(RB) has the second lowest error. The other methods performed notably worse. We attribute the good performance of GPP(RB) to the extra spread from the random additive bias and the poor performance of Shtarkov to its extreme simplicity. We do not compute analogs of variances because the data are assumed not to have a distribution.[2]

[2]We *could* compute an SE anyway and it would satisfy the Markov inequality if we used the EDF or EEDF. As an alternative we define 'stability curves'; see [6].

One pass				Representative					
HBP (Mean)	HBP (Median)	Sht	DPP	HBP (Mean)	HBP (Median)	Sht	DPP	GPP (RB)	GPP (no RB)
1006.8	**944.8**	986.8	989.1	972.8	960.1	959.7	985.2	946.9*	1000.5

Table 1: Final CPE's for the ten predictors using the Colombia rainfall data.

Table 2 presents the final CPE's for the India data. It is seen that the best methods have CPE around 1050. These are the representative subset versions of HBP mean, Sht, GPP(RB), and the one-pass version of HBP median. The only minor surprise here is that the representative set Shtarkov is doing so well. However, this does not contradict our basic inference that the top methods for the class of data used here are HBP and GPP methods, with a nod to HBP median and GPP(RB).

One pass				Representative					
HBP (Mean)	HBP (Med)	Sht	DPP	HBP (Mean)	HBP (Median)	Sht	DPP	GPP (RB)	GPP (no RB)
1231	1054*	1227	1237	**1050**	1171	**1050**	1152	**1050**	1065

Table 2: Final CPE's for the ten predictors using the India rainfall data.

Table 3 presents the final CPE's for the Bhubhneshwar data. It is seen that the HBP median predictors, one-pass and representative subset versions, are the best followed by GPP (no RB). Again, the best predictors are from the HBP and GPP classes, although GPP(RB) does relatively poorly.

One pass				Representative					
HBP (Mean)	HBP (Med)	Sht	DPP	HBP (Mean)	HBP (Median)	Sht	DPP	GPP (RB)	GPP (no RB)
9.619	**7.102**	9.633	9.934	10.201	**7.102**	10.476	9.648	8.613	7.460*

Table 3: Final CPE's for the ten predictors using the Bhubhneshwar rainfall data.

To look into the appearance of GPP(no RB) as a top method replacing GPP(RB), we plot the Colombia and Bhubhneshwar data as time series in Fig. 1. Although hard to see at the scale of this plot, the Bhubhneshwar data shows more regularity than the Colombia data which looks more patternless. Since patterns can indicate structure to improve prediction, the prediction problem of Bhubhneshwar may be a little easier so that the random bias term does not help the GPP. The pattern in the time series plot may also ensure that a representative subset from K-means really is representative enough to help prediction substantially. This leads us to propose that the Bhubhneshwar data is slightly less complex than the Colombia data and hence a little bit easier to predict well by using GPP(no RB) which is simpler than GPP(RB). We return to this point in Sec. 6.

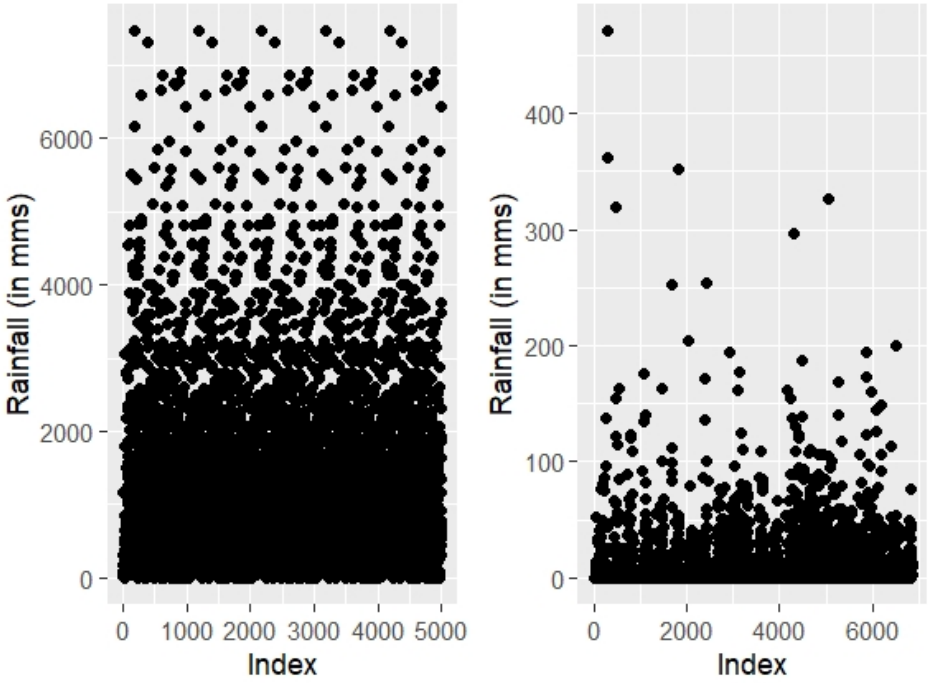

Fig. 1: Left: Plot of the Colombia data as a time series. Right: Plot of the Bhubhneshwar data as a time series.

Table 4 presents the final CPE's for the four disjoint subsets of size 2500 from the first 10000 data points in the accelerometer data. Overall, the results are in accord with our expectations that HBP and GPP methods will routinely perform best. The only extra comment is that in rows three and four other methods perform well, too. Indeed, better than expected. We explain this by noting that histograms of the four quarters of the data look well-behaved suggesting that this data set may also be at the less complex end of the class of $\mathcal{M}$-open data sets.

One pass				Representative					
HBP (Mean)	HBP (Mdn)	Sht	DPP	HBP (Mean)	HBP (Medn)	Sht	DPP	GPP (RB)	GPP (no RB)
0.326*	0.340	0.336	0.336	0.340	0.347	0.339	0.335	**0.305**	0.346
0.045	**0.034**	0.035*	0.042	0.166	0.067	0.074	0.040	0.074	0.353
0.069	**0.026**	**0.026**	0.027*	0.054	**0.026**	**0.026**	**0.026**	**0.026**	0.395
0.084	**0.026**	**0.026**	0.027*	0.043	**0.026**	**0.026**	0.027*	0.027*	0.383

Table 4: The four rows give the final CPE's for the four quarters of the 10000 data points extracted from the accelerometer data, in order. Here, Mdn abbreviates Median.

To explore these results a bit further, note that GPP (no RB) is essentially the worst performer in all four cases, perhaps because it is least flexible. Dropping it makes the range of errors for the four rows relatively similar. It is easily seen that the range for row 2 is the highest and the error for HBP (median, representative) is atypically large. This example does not contradict our inference that the best classes of predictors for $\mathcal{M}$-open data sets are GPP's and HBP's, but it does indicate the picture is not yet as clear cut as we would like. We return to this point in Sec. 6.

6 Discussion

The main contribution of this paper is to present and evaluate four predictive techniques for complex data, specifically $\mathcal{M}$-open data. We presented two new techniques – hash function based predictors (HBP's) using the Count-Min sketch and a Gaussian process prior with random additive bias predictor (GPP RB). We gave some of the key properties of these two predictors. In addition, we identified the mean as the Shtarkov predictor with normal experts and used a Dirichlet process prior to form a Bayesian non-parametric predictor, analogous to a frequentist histogram predictor.

The GPP methods were not one-pass and hence required modification to be used in the streaming setting. We did this by pre-processing the data by streaming K-means. We set $K = 200$ and for each time step found the cluster centers. We used these as our past data points, updating them as new data was received. We used the same pre-processing with our other other predictors (HBP's, Shtarkov, and DPP's) to ensure fair comparisons of L^1 predictive error. Thus, all the methods we compared satisfied a storage constraint and were one-pass, whether this was built into their construction as with HBP's or whether they required us to use a 'representative' subset from K-means.

Our computational results show that in all data sets we used here, and in other work (see [6]), at least one of the HBP or GPP methods is best. That is, for predictive purposes, Shtarkov and DPP predictors can be omitted. On the one hand, this is a weak statement because we are unable to specify which HBP or GPP method to use. We think that HBP median (one pass) or GPP (RB) are most often the best, and we have some evidence this is true, but we do not yet have enough evidence to claim this is true in substantial generality. The class of $\mathcal{M}$-open predictive problems is so large that our conclusions are necessarily tentative.

On the other hand, finding some regularity of performance over such a large scale class of problems may be all that can be expected at this time. We refer the reader to the literature on 'No Free Lunch' Theorems, the earliest statement of which is likely [29]. For a more recent review, see [27]. The intuition behind No Free Lunch Theorems suggests that the broad class of $\mathcal{M}$-open data sets can be partitioned into subclasses on which it will be possible to identify best predictors more effectively.

We can also imagine a complexity matching principle that relates the complexity of a data source to the complexity of a predictor i.e., for optimal prediction the complexity of the predictor class should 'match' the complexity of the data. Moreover, it is possible to use a stability evaluation of predictors, see [15] and [6]. In some cases, the predictive error 'flatlines' as a function of the size of perturbations. We interpret this to mean that when the complexity of the predictor is too small relative to the complexity of

the data, it can 'bail out' from the predictive problem i.e., become insensitive to the data. For the $\mathcal{M}$-open data sets we have used, this often happens with Shtarkov and DPP predictors, suggesting they are often simply not complex enough to predict well.

References

[1] S. Arora, E. Hazan, and S. Kale (2012) The Multiplicative Weights Update Method: A Meta-Algorithm and Applications. *Theory of Computing*, **8**, 121-164.

[2] R. Barber, E. Candès, A. Ramdas, and R. Tibshirani (2023) Conformal prediction beyond exchangeability. *Ann. Statist.*, **51**, 816-845.

[3] A. Barron, T. Roos, and Kazuho Watanabe (2014). Bayesian Properties of Normalized Maximum Likelihood and its Fast Computation. *Proc. IEEE International Symposium on Information Theory*. Honolulu, HI, 1667-1671.

[4] J. Bernardo and A. F. M. Smith (2000) *Bayesian Theory*, John Wiley and Sons, Chichester.

[5] N. Cesa-Bianchi and G. Lugosi *Prediction, Learning, and Games*. Cambridge University Press, Cambridge.

[6] A. Chanda and B. Clarke, (2024) Online prediction for Streaming Observational data. Submitted to *Stat. Surveys*.

[7] A. Chanda, N. V. Vinodchandran, and B. Clarke (2024) Point Prediction for Streaming Data https://arxiv.org/abs/2408.01318v1

[8] K. L. Chung (1974) *A First Course in Probability Theory* 2nd Ed. Academic Press, San Diego.

[9] C.-F. Chen (1985) On asymptotic normality of limiting density functions with Bayesian implications. *J. R. Stat. Soc.* Ser. B, **47**, 540-546.

[10] B. Clarke (2007) Information optimality and Bayesian modelling. *J. Econometrics*, **138**, 405-429.

[11] B. Clarke and Y. Yao (2025) A Cheat Sheet for Bayesian Prediction. *Stat. Sci.*, **40**, 3–24.

[12] G. Cormode and S. Muthukrishnan (2005) An Improved Data Stream Summary: The CountpMin Sketch and Its Applications *J. Algorithms*, **55**, 58-85.

[13] A. P. Dawid (1984) Present Position and Potential Developments: Some Personal Views: Statistical Theory: The Prequential Approach *J. Roy. Stat. Soc. Ser. A*, **147**,278-292.

[14] A. P. Dawid, M. Musio, and L. Ventura (2016) Minimum Scoring Rules Inference. *Scan. J. Stat.*, **43**, 123-138.

[15] D. Dustin, B. Clarke, and J. clarke (2024) Predictive stability criteria for penalty selection in linear models. *Comp. Stat.*, **39**, 1241–1280.

[16] D. Haussler and A. R. Barron (1993). How well do Bayes methods work for on-line prediction of + or -1 values? *Computational Learning and Cognition: Proc. Third NEC Research Symposium*, SIAM, Philadelphia, pp.74-101.

[17] S. Ghoshal (2010). The Dirichlet process, related priors and posterior asymptotics. In: *Bayesian nonparametrics* Hjort, Holmes, Muller, and Walker Eds. 28–35.

[18] S. Ghoshal and A. van der Vaart (2017) *Fundamentals Bayesian Nonparametric Inference*, Cambridge University Press, Cambridge.

[19] T. Gneiting (2011) Making and Evaluating Point Forecasts. *J. Amer. Stat. Assoc.*, **108**, 746-762.

[20] P. Kontkanen and P. Myllymaki (2007). A linear-time algorithm for computing the multinomial stochastic complexity. *Inform. Process. Lett.*, **103**, 227–233.

[21] T. Le and B. Clarke (2016) Using the Bayesian Shtarkov solution for predictions. *Comp. Stat. and Data Analysis*, **104**, 183–196

[22] T. Le and B. Clarke (2017) A Bayes Interpretation of Stacking for M-Complete and M-Open Settings. *Bayesian Anal.*, **12**, 807-829.

[23] S. Muthukrishnan, S. (2009) *Data stream algorithms.* The Barbados Workshop on Computational Complexity.

[24] J. Rissanen (1996) Fisher Information and Stochastic Complexity. *Trans. Inform. Theory*, **41**, 40-47.

[25] A. Shaikh (2009) https://home.uchicago.edu/~amshaikh/classes/topics_winter09.html and https://home.uchicago.edu/~amshaikh/webfiles/glivenko-cantelli.pdf. Last accessed 4 February 2025.

[26] Y. Shtarkov (1988). Universal sequential coding of single messages. Translation from *Problem of Information Transmission*, 3-17. San Mateo, Calif.: Morgan Kaufmann.

[27] A. Stavros, N. Stamatios-Aggelos, N. Alexandropoulos, P. Pardalos, and M. Vrahatis. (2019) No Free Lunch Theorem: A Review. In: *Approximation and optimization*, I. Demetriou and P. Pardalos (Eds.) 57-82.

[28] A. Suzuki, and K. Yamanishi (2018). Exact Calculation of Normalized Maximum Likelihood Code Length Using Fourier Analysis. See: https://arxiv.org/pdf/1801.03705v1

[29] D. Wolpert and W. Macready (1997). No Free Lunch Theorems for Optimization. *IEEE Trans. Evol. Comp.* **1**, 67–82.

[30] G. Shafer and V. Vovk (2008) A Tutorial on Conformal Prediction *J. Machine Learning Res.*, **9**, 371-421.

[31] H. Blunck, S. Bhattacharya, T. Prentow, M. Kjrgaard, and A. Dey (2015). Heterogeneity Activity Recognition. UCI Machine Learning Repository. https://doi.org/10.24432/C5689X.

[32] V. Vovk and A. Shen (2001) Prequential Randomness and Probability. *Theoretical Computer Science*, **411**, 632-646.

[33] Q. Xie and A. Barron (2000). Asymptotic minimax regret for data compression, gambling, and prediction. *IEEE Trans. Inform. Theory*, **46**, 431–445.

[34] X. Yang and A.R. Barron (2017) Minimax compression and large alphabet approximation through Poissonization and tilting. *IEEE Trans. Inform. Theory*, **63**, 2866-2884.

7 Appendix : Proof of Theorem 4

We use p generically to indicate probability densities. We use w when we want to emphasize that a density is a prior or posterior and m to emphasize that a density is a mixture of densities for its indicated arguments. Now, the posterior density for $a^n, \sigma^2 | y^n$ is given by:

$$p(a^n, \sigma^2 | y^n) = \frac{\mathcal{L}_1(a, \sigma^2 | y^n) \times w(a^n, \sigma^2)}{m(y^n)} = \frac{p(y^n, a^n, \sigma^2)}{m(y^n)}. \tag{7.1}$$

. We know that

$$Y^n \sim \mathcal{N}(a^n, \sigma^2 (I_{n\times n} + K_{n\times n})) \tag{7.2}$$

and

$$\begin{aligned} w(a^n, \sigma^2) &= \mathcal{N}(\gamma 1^n, \sigma^2 \delta^2 I_{n\times n})\ \mathcal{IG}(\alpha, \beta) \\ &= \frac{e^{-\frac{1}{2\sigma^2}(a^n - \gamma 1^n)'(\delta^2 I_{n\times n})^{-1}(a^n - \gamma 1^n)}}{(2\pi)^{\frac{n}{2}} (\sigma^2 \delta^2)^{\frac{n}{2}}} \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2}\right)^{\alpha+1} e^{-\frac{\beta}{\sigma^2}}. \end{aligned} \tag{7.3}$$

From (7.2) and (7.3) the numerator in (7.1) is given by

$$p(a^n, \sigma^2 | y^n) = \frac{e^{-\frac{1}{2\sigma^2}(a^n - \gamma 1^n)'(\delta^2 I_{n\times n})^{-1}(a^n - \gamma 1^n)}}{(2\pi)^{\frac{n}{2}} (\sigma^2 \delta^2)^{\frac{n}{2}}}$$

$$
\begin{aligned}
&\times \frac{\beta^{\alpha}}{\Gamma(\alpha)}\left(\frac{1}{\sigma^2}\right)^{2\alpha+2+n} e^{-\frac{\beta}{\sigma^2}} \frac{e^{-\frac{1}{2\sigma^2}(y^n-a^n)'(I+K)^{-1}_{n\times n}(y^n-a^n)}}{(2\pi)^{\frac{n}{2}}(\sigma^2)^{\frac{n}{2}}|(I+K)_{n\times n}|^{\frac{1}{2}}} \\
&= \frac{\beta^{\alpha}}{(2\pi)^{\frac{n}{2}+\frac{n}{2}}|(I+K)_{n\times n}|^{\frac{1}{2}}(\delta^2)^{\frac{n}{2}}\Gamma(\alpha)} \\
&\times e^{-\frac{1}{\sigma^2}[\beta+\frac{1}{2}(y^n-a^n)'(I+K)^{-1}_{n\times n}(y^n-a^n)+\frac{1}{2}(a^n-\gamma 1^n)'(\delta^2 I_{n\times n})^{-1}(a^n-\gamma 1^n)]}
\end{aligned} \tag{7.4}
$$

We simplify the terms in the exponent in (7.4) as follows. It is

$$
\begin{aligned}
&\beta+\frac{1}{2}(y^n-a^n)'(I+K)^{-1}_{n\times n}(y^n-a^n)+\frac{1}{2}(a^n-\gamma 1^n)'(\delta^2 I_{n\times n})^{-1}(a^n-\gamma 1^n) \\
=\ &\beta+\frac{1}{2}y'^n(I+K)^{-1}_{n\times n}y^n-\frac{1}{2}a'^n(I+K)^{-1}_{n\times n}y^n-\frac{1}{2}y'^n(I+K)^{-1}_{n\times n}a^n \\
&+\frac{1}{2}a'^n(I+K)^{-1}_{n\times n}a^n+\frac{1}{2}a'^n(\delta^2 I_{n\times n})^{-1}a^n-\frac{1}{2}\gamma 1'^n(\delta^2 I_{n\times n})^{-1}a^n \\
&-\frac{1}{2}\gamma a'^n(\delta^2)^{-1}1^n+\frac{1}{2}\gamma^2 1'^n(\delta^2 I_{n\times n})^{-1}1^n \\
=\ &\beta+\frac{1}{2}a'^n[(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}]a^n-a'^n[(I+K)^{-1}_{n\times n}y^n+\gamma(\delta^2 I_{n\times n})^{-1}1^n] \\
&+\frac{1}{2}y'^n(I+K)^{-1}_{n\times n}y^n+\frac{1}{2}\gamma^2 1'^n(\delta^2 I_{n\times n})^{-1}1^n \\
=\ &\beta+\frac{1}{2}a'^n[\{(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}\}^{-1}]^{-1}a^n \\
&-a'^n[\{(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}\}^{-1}]^{-1}[(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}]^{-1} \\
&[(I+K)^{-1}_{n\times n}y^n+\gamma(\delta^2 I_{n\times n})^{-1}1^n]+\frac{1}{2}y'^n(I+K)^{-1}_{n\times n}y^n+\frac{1}{2}\gamma^2 1'^n(\delta^2 I_{n\times n})^{-1}1^n.
\end{aligned}
$$

So, we have that $w(a^n,\sigma^2|y^n)$ equals

$$
\begin{aligned}
&\frac{\beta^{\alpha}}{(2\pi)^{\frac{n}{2}+\frac{n}{2}}|(I+K)_{n\times n}|^{\frac{1}{2}}(\delta^2)^{\frac{n}{2}}\Gamma(\alpha)}\left(\frac{1}{\sigma^2}\right)^{\alpha+1+\frac{n}{2}+\frac{n}{2}}\cdot \\
&\times e^{-\frac{1}{\sigma^2}[\beta+\frac{1}{2}a'^n[\{(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}\}^{-1}]^{-1}a^n]} \\
&\times e^{-\frac{1}{\sigma^2}[-a'^n[\{(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}\}^{-1}]^{-1}[(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}]^{-1}]} \\
&\times e^{-\frac{1}{\sigma^2}[(I+K)^{-1}_{n\times n}y^n+\gamma(\delta^2 I_{n\times n})^{-1}1^n]} \\
&\times e^{-\frac{1}{\sigma^2}\{\frac{1}{2}y'^n(I+K)^{-1}_{n\times n}y^n+\frac{1}{2}\gamma^2 1'^n(\delta^2 I_{n\times n})^{-1}1^n]\}} \\
=\ &\frac{\beta^{\alpha}}{(2\pi)^{\frac{n}{2}+\frac{n}{2}}|(I+K)_{n\times n}|^{\frac{1}{2}}(\delta^2)^{\frac{n}{2}}\Gamma(\alpha)} \\
&\times\left(\frac{1}{\sigma^2}\right)^{\alpha+1+\frac{n}{2}+\frac{n}{2}} e^{-\frac{1}{\sigma^2}\{\beta+\frac{1}{2}y'^n(I+K)^{-1}_{n\times n}y^n+\frac{1}{2}\gamma^2 1'^n(\delta^2 I_{n\times n})^{-1}1^n]\}} \\
&\times e^{-\frac{1}{\sigma^2}[\frac{1}{2}a'^n[\{(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}\}^{-1}]^{-1}a^n} \\
&\times e^{-\frac{1}{\sigma^2}[-a'^n[\{(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}\}^{-1}]^{-1}} \\
&\times e^{-\frac{1}{\sigma^2}[(I+K)^{-1}_{n\times n}+(\delta^2 I_{n\times n})^{-1}]^{-1}][(I+K)^{-1}_{n\times n}y+\gamma(\delta^2 I_{n\times n})^{-1}1^n]}.
\end{aligned} \tag{7.5}
$$

So, if we set

$$
\begin{aligned}
V_{n\times n} &= [(I+K)^{-1}_{n\times n} + (\delta^2 I_{n\times n})^{-1}]^{-1} && (7.6)\\
\mu &= [(I+K)^{-1}_{n\times n} + (\delta^2 I_{n\times n})^{-1}]^{-1}[(I+K)^{-1}_{n\times n} y^n + \gamma(\delta^2 I_{n\times n})^{-1} 1^n] \\
&= V_{n\times n}[(I+K)^{-1}_{n\times n} y^n + \gamma(\delta^2 I_{n\times n})^{-1} 1^n] && (7.7)\\
\beta^* &= \beta + \frac{1}{2} y'^n (I+K)^{-1}_{n\times n} y^n + \frac{1}{2}\gamma^2 1^n \delta^2 I_{n\times n}]^{-1} 1^n - \frac{1}{2}\mu'^n V^{-1}_{n\times n}\mu^n && (7.8)\\
\alpha^* &= n + \alpha, && (7.9)
\end{aligned}
$$

the expression in (7.5) becomes

$$
\begin{aligned}
w(a^n, \sigma^2|y^n) &= \frac{\beta^\alpha}{(2\pi)^n |(I+K)_{n\times n}|^{\frac{1}{2}} (\delta^2)^{\frac{n}{2}} \Gamma(\alpha)} \\
&\quad \times \left(\frac{1}{\sigma^2}\right)^{\alpha^*+1} e^{-\frac{1}{\sigma^2}(\beta^* + \frac{1}{2}\mu'^n V^{-1}_{n\times n}\mu^n + \frac{1}{2}a'^n V^{-1}_{n\times n} a^n - a'^n V^{-1}_{n\times n}\mu_{n\times n})}. \\
&= \frac{\beta^\alpha}{(2\pi)^n |I+K|^{\frac{1}{2}} (\delta^2)^{\frac{n}{2}} \Gamma(\alpha)} \\
&\quad \times \left(\frac{1}{\sigma^2}\right)^{\alpha^*+1} e^{-\frac{1}{\sigma^2}[\frac{1}{2}(a^n-\mu^n)' V^{-1}_{n\times n}(a^n-\mu^n)]} e^{-\frac{1}{\sigma^2}\beta^*}.
\end{aligned}
$$

The denominator in (7.1) is $m(y^n)$ equal to

$$
\begin{aligned}
&\int_{\mathbf{R}^+}\int_{\mathbf{R}^n} \mathcal{L}_1(a^n, \sigma^2|y^n) \times w(a^n, \sigma^2) \mathrm{da^n} \mathrm{d}\sigma^2. \\
&= \int_{\mathbf{R}^+}\left[\int_{\mathbf{R}^n} \frac{1}{(2\pi)^{\frac{n}{2}}(\sigma^2)^{\frac{n}{2}}|(I+K)_{n\times n}|^{\frac{1}{2}}} \times \frac{1}{(2\pi)^{\frac{n}{2}}(\sigma^2\delta^2)^{\frac{n}{2}}}\right. \\
&\quad \left.\times e^{-\frac{1}{2\sigma^2}(y^n-a^n)'(I+K)^{-1}_{n\times n}(y^n-a^n)} e^{-\frac{1}{2\sigma^2}(a^n-\gamma 1^n)'(\delta^2 I_{n\times n})^{-1}(a^n-\gamma 1^n)} \mathrm{da^n}\right] \\
&\quad \times \frac{\beta^\alpha}{\Gamma(\alpha)}\left(\frac{1}{\sigma^2}\right)^{\alpha+1} e^{-\frac{\beta}{\sigma^2}} d\sigma^2 \\
&= \int_{\mathbf{R}^+}\left[\frac{1}{(2\pi)^n(\sigma^2)^n|(I+K)_{n\times n}|^{\frac{1}{2}}(\delta^2)^{\frac{n}{2}}}\right. \\
&\quad \times \int_{\mathbf{R}^n}\Big\{e^{-\frac{1}{2\sigma^2}[y'^n(I+K)^{-1}_{n\times n}y - a'^n(I+K)^{-1}_{n\times n}y^n - y'^n(I+K)^{-1}_{n\times n}a^n + a'^n(I+K)^{-1}_{n\times n}a^n]} \\
&\quad \left.\times e^{-\frac{1}{2\sigma^2}[a'^n(\delta^2 I)^{-1}a^n - \gamma 1'^n(\delta^2 I_{n\times n})^{-1}1^n - \gamma a'^n(\delta^2 I_{n\times n})^{-1}1^n + \gamma^2 1'^n(\delta^2 I_{n\times n})^{-1}1^n]}\Big\}\mathrm{da^n} \\
&\quad \left.\times \frac{\beta^\alpha}{\Gamma(\alpha)}\left(\frac{1}{\sigma^2}\right)^{\alpha+1} e^{-\frac{\beta}{\sigma^2}}\right] d\sigma^2
\end{aligned}
$$

$$= \int_{\mathbf{R}^+} \Bigg(\frac{e^{-\frac{1}{2\sigma^2}[y'^n (I+K)^{-1}_{n\times n} y^n + \gamma^2 1'^n (\delta^2 I_{n\times n})^{-1} 1^n]}}{(2\pi)^n (\sigma^2)^n |(I+K)_{n\times n}|^{\frac{1}{2}} (\delta^2)^{\frac{n}{2}}} \times$$
$$\Bigg[\int_{\mathbf{R}^n} e^{-\frac{1}{2\sigma^2}\left[a'^n \left\{ (I+K)^{-1}_{n\times n} + (\delta^2 I_{n\times n})^{-1} \right\} a^n\right]}$$
$$\times e^{-\frac{1}{2\sigma^2}\left[-2a'^n \left\{ (I+K)^{-1}_{n\times n} + (\delta^2 I_{n\times n})^{-1} \right\} \left\{ (I+K)^{-1}_{n\times n} + (\delta^2 I_{n\times n})^{-1} \right\}^{-1}\right]}$$
$$\times e^{-\frac{1}{2\sigma^2}\left\{ (I+K)^{-1}_{n\times n} y^n + \gamma (\delta^2 I_{n\times n})^{-1} 1^n \right\}} \Bigg] \mathrm{da^n} \Bigg] \times \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2}\right)^{\alpha+1} \mathrm{e}^{-\frac{\beta}{\sigma^2}} \Bigg) \mathrm{d}\sigma^2. \quad (7.10)$$

Rewriting (7.10) in terms of equations (7.6) to (7.9) gives

$$= \int_{\mathbf{R}^+} \Bigg[\frac{e^{-\frac{1}{2\sigma^2}[y'^n (I+K)^{-1}_{n\times n} y^n + \gamma^2 1'^n (\delta^2 I_{n\times n})^{-1} 1^n]}}{(2\pi)^n (\sigma^2)^n |(I+K)_{n\times n}|^{\frac{1}{2}} (\delta^2)^{\frac{n}{2}}}$$
$$\times \left(\int_{\mathbf{R}^n} e^{-\frac{1}{2\sigma^2}\left[a' V^{-1}_{n\times n} a^n - 2a'^n V^{-1}_{n\times n} \mu^n\right]} \mathrm{da^n} \right) \times \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2}\right)^{\alpha+1} \mathrm{e}^{-\frac{\beta}{\sigma^2}} \Bigg] \mathrm{d}\sigma^2. \quad (7.11)$$

We complete the square in the inner integral (with respect to a^n) by multiplying and dividing (7.11) by $e^{-\frac{1}{2\sigma^2}\mu'^n V^{-1}_{n\times n}\mu^n}$. This gives

$$m(y^n) = \int_{\mathbf{R}^+} \Bigg[\frac{e^{-\frac{1}{2\sigma^2}[y'^n (I+K)^{-1}_{n\times n} y^n + \gamma^2 1'^n (\delta^2 I_{n\times n})^{-1} 1^n]}}{(2\pi)^n (\sigma^2)^n |(I+K)_{n\times n}|^{\frac{1}{2}} (\delta^2)^{\frac{n}{2}}} e^{\frac{1}{2\sigma^2}\mu'^n V^{-1}_{n\times n}\mu^n}$$
$$\times \left(\int_{\mathbf{R}^n} e^{-\frac{1}{2\sigma^2}(a^n-\mu^n)' V^{-1}_{n\times n}(a^n-\mu^n)} \mathrm{da^n} \right) \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2}\right)^{\alpha+1} \mathrm{e}^{-\frac{\beta}{\sigma^2}} \Bigg] \mathrm{d}\sigma^2. \quad (7.12)$$

The integral with respect to a^n in (7.12) becomes 1 if we divide and multiply (7.12) by $(2\pi)^{\frac{n}{2}} (\sigma^2)^{\frac{n}{2}} |V_{n\times n}|^{\frac{1}{2}}$, i.e.,

$$m(y^n) = \int_{\mathbf{R}^+} \Bigg[\frac{e^{-\frac{1}{2\sigma^2}[y'^n (I+K)^{-1}_{n\times n} y^n + \gamma^2 1'^n (\delta^2 I_{n\times n})^{-1} 1^n]}}{(2\pi)^n (\sigma^2)^n |(I+K)_{n\times n}|^{\frac{1}{2}} (\delta^2)^{\frac{n}{2}}} (2\pi)^{\frac{n}{2}} (\sigma^2)^{\frac{n}{2}} |V_{n\times n}|^{\frac{1}{2}}$$
$$e^{\frac{1}{2\sigma^2}\mu'^n V^{-1}_{n\times n}\mu^n} \left(\frac{1}{(2\pi)^{\frac{n}{2}} (\sigma^2)^{\frac{n}{2}} |V_{n\times n}|^{\frac{1}{2}}} \int_{\mathbf{R}} e^{-\frac{1}{2\sigma^2}(a^n-\mu^n)' V^{-1}_{n\times n}(a^n-\mu^n)} \mathrm{da^n} \right)$$
$$\times \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2}\right)^{\alpha+1} e^{-\frac{\beta}{\sigma^2}} \Bigg] \mathrm{d}\sigma^2$$

$$= \int_{\mathbf{R}^+} \left[\frac{(2\pi)^{\frac{n}{2}}(\sigma^2)^{\frac{n}{2}}|V_{n\times n}|^{\frac{1}{2}}}{(2\pi)^n(\sigma^2)^n|(I+K)_{n\times n}|^{\frac{1}{2}}(\delta^2)^{\frac{n}{2}}} e^{\frac{1}{2\sigma^2}\mu'^n V^{-1}_{n\times n}\mu^n} \right.$$
$$\left. \times e^{-\frac{1}{2\sigma^2}[y'^n(I+K)^{-1}_{n\times n}y^n + \gamma^2 1'^n(\delta^2 I_{n\times n})^{-1}1^n]} \frac{\beta^\alpha}{\Gamma(\alpha)} \Big(\frac{1}{\sigma^2}\Big)^{\alpha+1} e^{-\frac{\beta}{\sigma^2}} \right] \mathrm{d}\sigma^2. \quad (7.13)$$

Rearranging (7.13) gives that $m(y^n)$ equals

$$\frac{|V_{n\times n}|^{\frac{1}{2}}}{(2\pi\delta^2)^{\frac{n}{2}}|(I+K)_{n\times n}|^{\frac{1}{2}}} \frac{\beta^\alpha}{\Gamma(\alpha)}$$
$$\times \int_{\mathbf{R}^+} \Big(\frac{1}{\sigma^2}\Big)^{\alpha+\frac{n}{2}+1} e^{-\frac{1}{\sigma^2}\left[\beta+\frac{1}{2}\left\{y'^n(I+K)^{-1}_{n\times n}y^n+\gamma^2 1'^n(\delta^2 I_{n\times n})^{-1}1^n-\mu' V^{-1}_{n\times n}\mu^n\right\}\right]} \mathrm{d}\sigma^2. \quad (7.14)$$

Recall from (7.8) and (7.9) that:

$$\beta^* = \beta + \frac{1}{2}y'^n(I+K)^{-1}_{n\times n}y^n + \frac{1}{2}\gamma^2 1^n\delta^2 I_{n\times n}]^{-1}1^n - \frac{1}{2}\mu'^n V^{-1}_{n\times n}\mu^n$$
$$\alpha^* = n+\alpha.$$

Using them in (7.14) gives

$$m(y^n) = \frac{|V_{n\times n}|^{\frac{1}{2}}}{(2\pi\delta^2)^{\frac{n}{2}}|(I+K)_{n\times n}|^{\frac{1}{2}}} \frac{\beta^\alpha}{\Gamma(\alpha)} \int_{\mathbf{R}^+} \Big(\frac{1}{\sigma^2}\Big)^{\alpha^*-\frac{n}{2}+1} e^{-\frac{1}{\sigma^2}\beta^*} \mathrm{d}\sigma^2. \quad (7.15)$$

The integrand in (7.15) will be the pdf of an Inverse Gamma distribution and the integral will be 1, if we multiply and divide (7.15) by $\frac{\beta^{*\alpha^*-\frac{n}{2}}}{\Gamma(\alpha^*-\frac{n}{2})}$. So we have

$$m(y^n) = \frac{|V_{n\times n}|^{\frac{1}{2}}}{(2\pi\delta^2)^{\frac{n}{2}}|I+K|^{\frac{1}{2}}} \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha^*-\frac{n}{2})}{\beta^{*\alpha^*-\frac{n}{2}}} \int_{\mathbf{R}^+} \frac{\beta^{*\alpha^*-\frac{n}{2}}}{\Gamma(\alpha^*-\frac{n}{2})} \Big(\frac{1}{\sigma^2}\Big)^{\alpha^*-\frac{n}{2}+1} e^{-\frac{1}{\sigma^2}\beta^*} \mathrm{d}\sigma^2.$$
$$= \frac{|V_{n\times n}|^{\frac{1}{2}}}{(2\pi\delta^2)^{\frac{n}{2}}|I+K|^{\frac{1}{2}}} \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha^*-\frac{n}{2})}{\beta^{*\alpha^*-\frac{n}{2}}}. \quad (7.16)$$

Using (7.9) in (7.16) for $n+1$ and n gives

$$\frac{m(y^{n+1})}{m(y^n)} = \frac{\frac{|V_{n+1\times n+1}|^{\frac{1}{2}}}{(2\pi\delta^2)^{\frac{n+1}{2}}|(I+K)_{n+1\times n+1}|^{\frac{1}{2}}} \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha+\frac{n+1}{2})}{\beta^{*\alpha+\frac{n+1}{2}}_{n+1}}}{\frac{|V_{n\times n}|^{\frac{1}{2}}}{(2\pi\delta^2)^{\frac{n}{2}}|(I+K)_{n\times n}|^{\frac{1}{2}}} \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha+\frac{n}{2})}{\beta^{*\alpha+\frac{n}{2}}_n}}$$
$$= \frac{|V_{n+1\times n+1}|^{\frac{1}{2}}}{|V_{n\times n}|^{\frac{1}{2}}} \frac{|(I+K)_{n\times n}|^{\frac{1}{2}}}{|(I+K)_{n+1\times n+1}|^{\frac{1}{2}}}$$

$$\times \frac{(2\pi\delta^2)^{\frac{n}{2}}}{(2\pi\delta^2)^{\frac{n+1}{2}}} \frac{\frac{\Gamma(\alpha+\frac{n+1}{2})}{\Gamma(\alpha)}}{\frac{\Gamma(\alpha+\frac{n}{2})}{\Gamma(\alpha)}} \frac{\beta^\alpha}{\beta^\alpha} \frac{(\beta^*_{n+1})^{-(\alpha+\frac{n+1}{2})}}{(\beta^*_n)^{-(\alpha+\frac{n}{2})}}$$
$$= c \frac{(\beta^*_{n+1})^{-(\alpha+\frac{n+1}{2})}}{(\beta^*_n)^{-(\alpha+\frac{n}{2})}}, \tag{7.17}$$

where

$$c = \frac{|V_{n+1\times n+1}|^{\frac{1}{2}}}{|V_{n\times n}|^{\frac{1}{2}}} \frac{|(I+K)_{n\times n}|^{\frac{1}{2}}}{|(I+K)_{n+1\times n+1}|^{\frac{1}{2}}} \frac{(2\pi\delta^2)^{\frac{n}{2}}}{(2\pi\delta^2)^{\frac{n+1}{2}}} \frac{\frac{\Gamma(\alpha+\frac{n+1}{2})}{\Gamma(\alpha)}}{\frac{\Gamma(\alpha+\frac{n}{2})}{\Gamma(\alpha)}} \frac{\beta^\alpha}{\beta^\alpha}. \tag{7.18}$$

From (7.7) and (7.8), we have

$$\begin{aligned}
\mu^n &= V_{n\times n}[(I+K)^{-1}_{n\times n} y_n + \gamma(\delta^2 I)^{-1} 1_n] \\
\beta^*_n &= \beta + \frac{1}{2} y'_n (I+K)^{-1}_{n\times n} y + \frac{1}{2}\gamma^2 1_n [\delta^2 I]^{-1} 1_n - \frac{1}{2}\mu^{'n} V^{-1}_{n\times n} \mu^n.
\end{aligned}$$

Thus, $\mu^{'n} V^{-1}_{n\times n} \mu^n$ equals

$$\begin{aligned}
&[V_{n\times n}\{(I+K)^{-1}_{n\times n} y^n + \gamma(\delta^2 I)^{-1} 1^n\}]' V^{-1}_{n\times n} [V_{n\times n}\{(I+K)^{-1}_{n\times n} y^n + \gamma(\delta^2 I)^{-1} 1^n\}] \\
&= \left[y'^n (I+K)^{-1}_{n\times n} + 1'^n \frac{\gamma}{\delta^2}\right] V'_{n\times n} V^{-1}_{n\times n} V_{n\times n} \left[(I+K)^{-1}_{n\times n} y^n + \frac{\gamma}{\delta^2} 1^n\right].
\end{aligned} \tag{7.19}$$

Since V is symmetric, i.e., $V' = V$, we have

$$\begin{aligned}
\mu^{'n} V^{-1}_{n\times n} \mu^n &= y'^n (I+K)^{-1}_{n\times n} V_{n\times n} (I+K)^{-1}_{n\times n} y^n + 2\frac{\gamma}{\delta^2} y'^n (I+K)^{-1}_{n\times n} V_{n\times n} 1^n \\
&\quad + \frac{\gamma^2}{\delta^4} 1'^n V_{n\times n} 1'^n.
\end{aligned} \tag{7.20}$$

Using (7.20) in (7.19), we get

$$\begin{aligned}
\beta^*_n &= \beta + \frac{1}{2} y'_n (I+K)^{-1}_{n\times n} y + \frac{1}{2}\gamma^2 1_n [\delta^2 I]^{-1} 1_n \\
&\quad - \frac{1}{2}\left[y'^n (I+K)^{-1}_{n\times n} V_{n\times n} (I+K)^{-1}_{n\times n} y^n\right] \\
&\quad - \frac{1}{2}\left[+ 2\frac{\gamma}{\delta^2} y'^n (I+K)^{-1}_{n\times n} V_{n\times n} 1^n + \frac{\gamma^2}{\delta^4} 1'^n V_{n\times n} 1'^n\right] \\
&= \beta + \frac{1}{2} y'^n [(I+K)^{-1}_{n\times n} - (I+K)^{-1}_{n\times n} V_{n\times n} (I+K)^{-1}_{n\times n}] y^n \\
&\quad - \frac{\gamma}{\delta^2} y'^n (I+K)^{-1}_{n\times n} V_{n\times n} 1^n + \frac{n}{2}\frac{\gamma^2}{\delta^2} - \frac{1}{2}\frac{\gamma^2}{\delta^4} 1'^n V_{n\times n} 1^n.
\end{aligned}$$

So, β^*_{n+1} equals

$$\beta + \frac{1}{2}y'^{n+1}[(I+K)^{-1}_{n+1\times n+1} - (I+K)^{-1}_{n+1\times n+1}V_{n+1\times n+1}(I+K)^{-1}_{n+1\times n+1}]y^{n+1}$$
$$-\frac{\gamma}{\delta^2}y'^{n+1}(I+K)^{-1}_{n+1\times n+1}V_{n+1\times n+1}1^{n+1} + \frac{n+1}{2}\frac{\gamma^2}{\delta^2} - \frac{1}{2}\frac{\gamma^2}{\delta^4}1'^{n+1}V_{n+1\times n+1}1^{n+1}. \tag{7.21}$$

Define

$$\Gamma_{1,n+1\times n+1} = (I+K)^{-1}_{n+1\times n+1} - (I+K)^{-1}_{n+1\times n+1}V_{n+1\times n+1}(I+K)^{-1}_{n+1\times n+1} \tag{7.22}$$
$$\Gamma_2^{n+1} = \frac{\gamma}{\delta^2}y'^{n+1}(I+K)^{-1}_{n+1\times n+1}V_{n+1\times n+1}1^{n+1} \tag{7.23}$$
$$\text{and } \Delta = \frac{n+1}{2}\frac{\gamma^2}{\delta^2} - \frac{1}{2}\frac{\gamma^2}{\delta^4}1'^{n+1}V_{n+1\times n+1}1^n. \tag{7.24}$$

Using (7.22), (7.23), and (7.24) in (7.21), we get

$$\beta^*_{n+1} = \beta + \frac{1}{2}y'^{n+1}\Gamma_{1,n+1\times n+1}y^{n+1} - y'^{n+1}\Gamma_2^{n+1} + \Delta. \tag{7.25}$$

Now, we partition y^{n+1}, $\Gamma_{1,n+1\times n+1}$, and Γ_2^{n+1}. Write

$$y'^{n+1}\Gamma_{1,n+1\times n+1}y^{n+1} = \begin{pmatrix} y^n & y_{n+1}\end{pmatrix}\begin{pmatrix} \Gamma_{1,n\times n} & \vdots & g_1^n \\ \dots & \vdots & \dots \\ g_1'^n & \vdots & \gamma_1\end{pmatrix}\begin{pmatrix} y^n \\ y_{n+1}\end{pmatrix}$$
$$= y'^n\Gamma_{1,n\times n}y^n + 2y'^n g_1^n y_{n+1} + y_{n+1}^2\gamma_1 \tag{7.26}$$

and

$$y'^{n+1}\Gamma_2^{n+1} = \begin{pmatrix} y^n & y_{n+1}\end{pmatrix}\begin{pmatrix}\Gamma_2^n \\ \gamma_2\end{pmatrix} = y'^n\Gamma_2^n + y_{n+1}\gamma_2. \tag{7.27}$$

Using (7.26) and (7.27) in (7.25), we get

$$\beta^*_{n+1} = \beta + y'^n\Gamma_{1,n\times n}y^n + 2y'^n g_1^n y_{n+1} + y_{n+1}^2\gamma_1 + y'^n\Gamma_2^n + y_{n+1}\gamma_2 + \Delta.$$
$$= \beta + \frac{1}{2}y'^n\Gamma_{1,n\times n}y^n - y'^n\Gamma_2^n + \Delta + \frac{1}{2}\gamma_1 y_{n+1}^2 - y_{n+1}(\gamma_2 - y'^n g_1^n). \tag{7.28}$$

We complete the square again. The terms in (7.28) containing y_{n+1} become

$$\frac{1}{2}\gamma_1 y_{n+1}^2 - y_{n+1}(\gamma_2 - y'^n g_1^n)$$

$$= \frac{1}{2}\gamma_1\left[y_{n+1}^2 - 2y_{n+1}\frac{\gamma_2 - y'^n g_1^n}{\gamma_1} + \left(\frac{\gamma_2 - y'^n g_1^n}{\gamma_1}\right)^2\right] - \frac{1}{2}\frac{(\gamma_2 - y'^n g_1^n)^2}{\gamma_1}$$
$$= \frac{\gamma_1}{2}\left[y_{n+1} - \frac{\gamma_2 - y'^n g_1^n}{\gamma_1}\right]^2 - \frac{1}{2\gamma_1}(\gamma_2 - y'^n g_1^n)^2. \tag{7.29}$$

For brevity, let

$$\begin{aligned} A_1 &= \frac{\gamma_2 - y'^n g_1^n}{\gamma_1} \\ A_2 &= \frac{1}{2}y'^n \Gamma_{1,n\times n} y^n - y'^n \Gamma_2^n + \Delta - \frac{1}{2\gamma_1}(\gamma_2 - y'^n g_1^n)^2. \end{aligned} \tag{7.30}$$

Using (7.30) in (7.28), we have

$$\beta_{n+1}^* = \beta + \frac{\gamma_1}{2}(y_{n+1} - A_1)^2 + A_2.$$

Now, since $m(y^n)$ is the marginal density of y^n and, $m(y^{n+1})$ is the marginal density of y^{n+1},

$$\int_{\mathbb{R}} \frac{m(y^{n+1})}{m(y^n)} \mathrm{dy_{n+1}} = 1. \tag{7.31}$$

From (7.17) we have that

$$\int_{\mathbb{R}} c \times \frac{{\beta_{n+1}^*}^{-\left(\alpha+\frac{n+1}{2}\right)}}{{\beta_n^*}^{-\left(\alpha+\frac{n}{2}\right)}} \mathrm{dy_{n+1}} = 1. \tag{7.32}$$

So solving for c gives

$$c = \frac{{\beta_n^*}^{-\left(\alpha+\frac{n}{2}\right)}}{\int_{\mathbb{R}} {\beta_{n+1}^*}^{-\left(\alpha+\frac{n+1}{2}\right)} \mathrm{dy_{n+1}}}.$$

Using (7.32) in (7.17), we have

$$\begin{aligned} \frac{m(y^{n+1})}{m(y^n)} &= \frac{{\beta_n^*}^{-\left(\alpha+\frac{n}{2}\right)}}{\int_{\mathbb{R}} {\beta_{n+1}^*}^{-\left(\alpha+\frac{n+1}{2}\right)} \mathrm{dy_{n+1}}} \times \frac{(\beta_{n+1}^*)^{-(\alpha+\frac{n+1}{2})}}{(\beta_n^*)^{-(\alpha+\frac{n}{2})}} \\ &= \frac{{\beta_{n+1}^*}^{-\left(\alpha+\frac{n+1}{2}\right)}}{\int_{\mathbb{R}} {\beta_{n+1}^*}^{-\left(\alpha+\frac{n+1}{2}\right)} \mathrm{dy_{n+1}}}. \end{aligned} \tag{7.33}$$

Now,

$$\beta_{n+1}^{*}{}^{-(\alpha+\frac{n+1}{2})} = \left[\beta + \frac{\gamma_1}{2}(y_{n+1} - A_1)^2 + A_2\right]^{-(\alpha+\frac{n+1}{2})}. \tag{7.34}$$

Rename, $\beta^{**} = \beta + A_2$. Then, (7.34) becomes

$$\begin{aligned}\beta_{n+1}^{*}{}^{-(\alpha+\frac{n+1}{2})} &= \left[\beta^{**} + \frac{\gamma_1}{2}(y_{n+1} - A_1)^2\right]^{-(\alpha+\frac{n+1}{2})} \\ &= \beta^{**-\left(\alpha+\frac{n}{2}\right)}\beta^{**-\frac{1}{2}}\left[1 + \frac{\gamma_1}{2\beta^{**}}(y_{n+1} - A_1)^2\right]^{-\left(\alpha+\frac{n+1}{2}\right)}. \end{aligned} \tag{7.35}$$

By definition, the t-density is given by

$$St_v(\tau, \Sigma)(g) = \frac{\Gamma(\frac{v+d}{2})}{\Gamma(\frac{v}{2})\pi^{\frac{d}{2}}|v\Sigma|^{\frac{1}{2}}}\left(1 + \frac{(g-\tau)'\Sigma^{-1}(g-\tau)}{v}\right)^{-\frac{v+d}{2}}. \tag{7.36}$$

So if we let

$$v = 2\alpha, d = 1, \Sigma = \frac{\beta^{**}}{\frac{2\alpha+n}{2}}\frac{1}{\gamma_1}, g = y_{n+1}, \text{and } \tau = A_1 \tag{7.37}$$

and use (7.37) in (7.36), we get

$$\begin{aligned} & St_{2\alpha+n}\left(A_1, \frac{\beta^{**}}{\frac{2\alpha+n}{2}}\right)(y_{n+1}) \\ = & \frac{\Gamma(\frac{2\alpha+n+1}{2})}{\Gamma(\frac{2\alpha+n}{2})\pi^{\frac{1}{2}}\left|(2\alpha+n)\frac{\beta^{**}}{\frac{2\alpha+n}{2}}\frac{1}{\gamma_1}\right|^{\frac{1}{2}}} \\ & \times\left(1 + \frac{(y_{n+1} - A_1)'\left(\frac{\beta^{**}}{\frac{2\alpha+n}{2}}\frac{1}{\gamma_1}\right)^{-1}(y_{n+1} - A_1)}{2\alpha+n}\right)^{-\frac{2\alpha+n+1}{2}} \\ = & \frac{\Gamma(\frac{2\alpha+n+1}{2})}{\Gamma(\frac{2\alpha+n}{2})}\gamma_1^{\frac{1}{2}}\frac{1}{(2\pi)^{\frac{1}{2}}}\beta^{**-\frac{1}{2}}\left[1 + \frac{\gamma_1}{2\beta^{**}}(y_{n+1} - A_1)^2\right]^{-\frac{2\alpha+n+1}{2}}. \end{aligned}$$

Hence,

$$\begin{aligned} & \beta^{**-\frac{1}{2}}\left[1 + \frac{\gamma_1}{2\beta^{**}}(y_{n+1} - A_1)^2\right]^{-\left(\alpha+\frac{n+1}{2}\right)} \\ & = \frac{\Gamma(\frac{2\alpha+n+1}{2})}{\Gamma(\frac{2\alpha+n}{2})}\frac{(2\pi)^{\frac{1}{2}}}{\gamma_1^{\frac{1}{2}}} \times St_{2\alpha+n}\left(A_1, \frac{\beta^{**}}{\frac{2\alpha+n}{2}}\right)(y_{n+1}). \end{aligned} \tag{7.38}$$

Using (7.38) in (7.35), and (7.35) in (7.33), we have

$$\frac{m(y^{n+1})}{m(y^n)} = \frac{{\beta^{**}}^{-(\alpha+\frac{n}{2})}}{\int_{\mathbb{R}} {\beta^*_{n+1}}^{-(\alpha+\frac{n+1}{2})} \mathrm{dy_{n+1}}} \frac{\Gamma(\frac{2\alpha+n}{2})}{\Gamma(\frac{2\alpha+n+1}{2})} \frac{(2\pi)^{\frac{1}{2}}}{(\gamma_1)^{\frac{1}{2}}} \times St_{2\alpha+n}\left(A_1, \frac{\beta^{**}}{\frac{2\alpha+n}{2}}\right)(y_{n+1}). \tag{7.39}$$

Since $\frac{m(y^{n+1})}{m(y^n)} = m(y_{n+1}|y^n)$ is a density, $\int_{\mathbb{R}} \frac{m(y^{n+1})}{m(y^n)} \mathrm{dy_{n+1}} = 1$. Integrating the right hand side of(7.39) w.r.t y_{n+1} gives that

$$\frac{{\beta^{**}}^{-(\alpha+\frac{n}{2})}}{\int_{\mathbb{R}} {\beta^*_{n+1}}^{-(\alpha+\frac{n+1}{2})} \mathrm{dy_{n+1}}} \frac{\Gamma(\frac{2\alpha+n}{2})}{\Gamma(\frac{2\alpha+n+1}{2})} \frac{(2\pi)^{\frac{1}{2}}}{(\gamma_1)^{\frac{1}{2}}} \int_{\mathbb{R}} St_{2\alpha+n}\left(A_1, \frac{\beta^{**}}{\frac{2\alpha+n}{2}}\right)(y_{n+1}) \mathrm{dy_{n+1}} \tag{7.40}$$

equals 1, since y_{n+1} is only in the argument of the t distribution. Thus,

$$\frac{{\beta^{**}}^{-(\alpha+\frac{n}{2})}}{\int_{\mathbb{R}} {\beta^*_{n+1}}^{-(\alpha+\frac{n+1}{2})} \mathrm{dy_{n+1}}} \frac{\Gamma(\frac{2\alpha+n}{2})}{\Gamma(\frac{2\alpha+n+1}{2})} \frac{(2\pi)^{\frac{1}{2}}}{(\gamma_1)^{\frac{1}{2}}} = 1. \tag{7.41}$$

Finally using (7.41) in (7.39), we get

$$m(y_{n+1}|y^n) = St_{2\alpha+n}\left(A_1, \frac{\beta^{**}}{\frac{2\alpha+n}{2}}\right)(y_{n+1}). \quad \square$$