Extended appendix

In section 4.2 of the main text we describe a polynomial time smoother for multinomial data confined to the unit simplex. Here we derive the forward variables of this smoother, which we need to perform efficient inference. The necessary backward variables may be derived following the same integration steps but starting from time T and proceeding backward.

Define $s_0 \equiv 0$, and let k_i be the multi-index $[k_{i1}, k_{i2}]$, over which the sum $\sum_{k_i} x_i^{k_{i1}} x_{i+1}^{k_{i2}}$ couples x_i to x_{i+1} (in addition to the implicit coupling by the simplex constraint). Let $\{a_k(t)\} = \{a_k\}$ be a set of coefficients, indexed by multi-index k, that we assume to be constant over time, merely for notational convenience. Pushing all sums as far to the right as possible, and defining $\delta_i \equiv \delta(s_i = s_{i+1} - x_{i+1})$, the joint density in the expanded state-space is

$$
\begin{aligned}
p(Q) = &\sum_{k_1}^{R} a_{k_1} x_1^{\nu_1 + k_{1,1} - 1} \delta_0 \times \\
&\sum_{k_2}^{R} a_{k_2} x_2^{\nu_2 + k_{1,2} + k_{2,1} - 1} \delta_1 \times \cdots \\
&\sum_{k_{T-1}}^{R} a_{k_{T-1}} x_{T-1}^{\nu_{T-1} + k_{T-2,2} + k_{T-1,1} - 1} \delta_{T-2} \times \\
&x_T^{\nu_T + k_{T-1,2} - 1} \delta_{T-1} \delta(s_T = 1)
\end{aligned}
$$

For the purpose of computing the forward variables, the multinomial exponents $\{\nu_i - 1\}$ may be omitted and considered absorbed by the $k_{i,j}$, so as to further simplify the notation.

We compute the normalization constant of the joint distribution:

$$
\begin{aligned}
\mathcal{Z} &= \int_0^1 ds_1 \int_0^{s_1} dx_1 \cdots \int_0^1 ds_T \int_0^{s_T} dx_T \, p(Q) \\
&= \int_0^1 ds_1 \int_0^{s_1} dx_1 \sum_{k_1}^{R} a_{k_1} x_1^{k_{1,1}} \delta_0 \times \\
&\int_0^1 ds_2 \int_0^{s_2} dx_2 \sum_{k_2}^{R} a_{k_2} x_2^{k_{1,2} + k_{2,1}} \delta_1 \times \cdots
\end{aligned}
$$

To compute $\mathcal{Z}$, we integrate from left to right, first integrating dx_1, then ds_1, then dx_2, then ds_2, and so on until ds_T. Even after only the first five integrations, a pattern begins to emerge:

$$
\begin{aligned}
\mathcal{Z} = &\sum_{k_1} a_{k_1} \int_0^1 ds_1 (s_1 - s_0)^{k_{1,1}} \times \\
&\int_0^1 ds_2 \int_0^{s_2} dx_2 \sum_{k_2} a_{k_2} x_2^{k_{1,2} + k_{2,1}} \delta_1 \times \cdots
\end{aligned}
$$

$$
\begin{aligned}
= &\sum_{k_1} a_{k_1} \int_0^1 ds_2 \int_0^{s_2} dx_2 (s_2 - x_2)^{k_{1,1}} \times \\
&\sum_{k_2} a_{k_2} x_2^{k_{1,2} + k_{2,1}} \times \cdots
\end{aligned}
$$

$$
\begin{aligned}
= &\sum_{k_1} a_{k_1} \sum_{k_2} a_{k_2} \mathrm{B}(k_{1,2} + k_{2,1}, k_{1,1}) \times \\
&\int_0^1 ds_2 s_2^{k_{1,1} + k_{1,2} + k_{2,1}} \times \\
&\int_0^1 ds_3 \int_0^{s_3} dx_3 \sum_{k_3} a_{k_3} x_3^{k_{2,2} + k_{3,1}} \delta_2 \times \cdots
\end{aligned}
$$

$$
\begin{aligned}
= &\sum_{k_1} a_{k_1} \sum_{k_2} a_{k_2} \mathrm{B}(k_{1,2} + k_{2,1}, k_{1,1}) \times \\
&\int_0^1 ds_3 \int_0^{s_3} dx_3 (s_3 - x_3)^{k_{1,1} + k_{1,2} + k_{2,1}} \times \\
&\sum_{k_3} a_{k_3} x_3^{k_{2,2} + k_{3,1}} \times \cdots
\end{aligned}
$$

$$
\begin{aligned}
= &\sum_{k_1} a_{k_1} \sum_{k_2} a_{k_2} \mathrm{B}(k_{1,2} + k_{2,1}, k_{1,1}) \times \\
&\sum_{k_3} a_{k_3} \mathrm{B}(k_{2,2} + k_{3,1}, k_{1,1} + k_{1,2} + k_{2,2}) \times \\
&\int ds_3 s_3^{k_{1,1} + k_{1,2} + k_{2,1} + k_{2,2} + k_{3,1}} \times \\
&\int_0^1 ds_4 \int_0^{s_4} dx_4 \sum_{k_4} a_{k_4} x_4^{k_{3,2} + k_{4,1}} \delta_3 \times \cdots
\end{aligned}
$$

The first equality results from integration with respect to x_1, which removes the delta function $\delta_0 = \delta(s_0 = s_1 - x_1) = \delta(x_1 = s_1)$, leaving $(s_1 - s_0)^{k_{1,1}}$ in place of $x_1^{k_{1,1}}$. The second equality results from integration with respect to s_1, which removes the delta function $\delta_1 = \delta(s_1 = s_2 - x_2)$, which replaces $(s_1 - s_0)^{k_{1,1}} = s_1^{k_{1,1}}$ with $(s_2 - x_2)^{k_{1,1}}$. Next we integrate with respect to x_2, which proceeds by the following change of variables:

$$
\begin{aligned}
&\int_0^s dx (s - x)^{\alpha - 1} x^{\beta - 1} \overset{u = x/s}{=} \\
&\int_0^1 du (s(1 - u))^{\alpha - 1} (su)^{\beta - 1} s = \\
&s^{\alpha + \beta - 1} \int_0^1 (1 - u)^{\alpha - 1} u^{\beta - 1} = s^{\alpha + \beta - 1} \mathrm{B}(\alpha, \beta)
\end{aligned}
$$

where $\mathrm{B}(\alpha, \beta)$ is the beta function. The next integration with respect to s_2 removes the delta function $\delta_2 = \delta(s_2 = s_3 - x_3)$, resulting in the power of $(s_3 - x_3)$. The last equality is a result of the same change of variables when integrating with respect to x_3, yielding another beta function, and leaving the last line of the expression identical to the last line of the expression two steps before, except that the "time" indices have all increased by one.

To condense this expression, define $k_{0,2} \equiv k_{T,1} \equiv 0$, and $K_i \equiv \sum_{j=1}^{i} k_{j-1,2} + k_{j,1}$. Lastly, define $b(k_{1:n}) \equiv \mathrm{B}(k_{n-1,2} + k_{n,1}, K_{n-1})$. Continuing with the integration from left to right, we find an expression for the normalization constant as the following nested sum:

$$\mathcal{Z} = \sum_{k_1}^{R} a_{k_1} \left(\sum_{k_2}^{R} a_{k_2} b(k_{1:2}) \left(\sum_{k_3}^{R} a_{k_3} b(k_{1:3}) \times \cdots \right.\right.$$
$$\left(\sum_{k_{T-2}}^{R} a_{k_{T-2}} b(k_{1:T-2}) \left(\sum_{k_{T-1}}^{R} a_{k_{T-1}} b(k_{1:T-1}) \times \right.\right.$$
$$\left.\left.\left.\left. b(k_{1:T}) \right)\right) \cdots \right)\right)$$

This is not in sum-product form because, e.g., $b(k_{1:T})$ depends on all the indices of summation. We may put this into sum-product form as follows. However, we can rearrange this summation into the form of a tractable sum-product algorithm as follows. Each sum over k_i is first a sum over $k_{i,1} \in \{0, \cdots, R\}$, and then a sum over $k_{i,2} \in \{0, \cdots, R\}$. The sums are in the order $k_{1,1}$, $k_{1,2}$, $k_{2,1}$, $k_{2,2}$, and so on. Now, notice that every set of values $\{k_i\}_{i=1}^{T} = \{[k_{i,1}, k_{i,2}]\}_{i=1}^{T}$ corresponds uniquely to a set of values $\{[k_{i,1}, K_i]\}_{i=1}^{T}$, with K_i as defined above, and vice versa. Then we may sum over values of the latter quantity in the order K_T, $k_{T-1,1}$, K_{T-1}, $k_{T-2,1}$, and so on, instead of summing over values of k_i, and obtain the same result. That is, we treat the K_i as sum indices instead of explicitly summing over the $k_{i,2}$.

To sum over all possible values of this second set of indices, the values of the indices must be constrained to be compatible. For instance, $k_{1,1} = 1$ is not compatible with $K_2 = 0$, because $K_2 = k_{1,1} + k_{1,2} + k_{2,1} \geq k_{1,1}$. This compatibility requirement manifests in the upper and lower bounds of the sums over these indices in eq. (1). Consider the sum over $k_{1,1}$, given some values for $k_{2,1}$ and K_2 (which come earlier in the multiple sum). *A priori*, $k_{1,1} \in \{0, \cdots, R\}$. However, $k_{1,1} = K_2 - k_{2,1} - k_{1,2} \leq K_2 - k_{2,1}$ and $k_{1,1} = K_2 - k_{2,1} - k_{1,2} \geq K_2 - k_{2,1} - R$, so we have $k_{1,1} \in \{\max\{0, K_2 - k_{2,1} - R\}, \cdots, \min\{R, K_2 - k_{2,1}\}\}$. These are the values of $k_{1,1}$ that are to be summed over, i.e. that are compatible with the values of previously specified indices. More generally,

$$\begin{aligned} k_{i,1} &= K_{i+1} - k_{i+1,1} - K_{i-1} - k_{i-1,2} - k_{i,2} \\ &\geq K_{i+1} - k_{i+1,1} - (2i-1)R \end{aligned}$$

$$\begin{aligned} k_{i,1} &= K_{i+1} - k_{i+1,1} - K_{i-1} - k_{i-1,2} - k_{i,2} \\ &\leq K_{i+1} - k_{i+1,1} \end{aligned}$$

$$0 \leq k_{i,1} \leq R$$

$$\begin{aligned} k_{i,1} \in \{&\max\{0, K_{i+1} - k_{i+1,1} - (2i-1)R\}, \cdots, \\ &\min\{R, K_{i+1} - k_{i+1,1}\}\} \end{aligned}$$

Similarly,

$$\begin{aligned} K_i &= K_{i+1} - k_{i+1,1} - k_{i,2} \\ &\geq K_{i+1} - k_{i+1,1} - R \end{aligned}$$

$$\begin{aligned} K_i &= K_{i+1} - k_{i+1,1} - k_{i,2} \\ &\leq K_{i+1} - k_{i+1,1} \end{aligned}$$

$$K_i \geq k_{i,1}$$

$$\begin{aligned} K_i \in \{&\max\{k_{i,1}, K_{i+1} - k_{i+1,1} - R\}, \cdots, \\ &K_{i+1} - k_{i+1,1}\} \end{aligned}$$

We redefine $b(k_{1:n}) = \mathrm{B}(k_{n-1,2} + k_{n,1}, K_{n-1})$ as the same quantity in terms of the new indices of summation, $b(K_{n-1}, K_n) \equiv \mathrm{B}(K_n - K_{n-1}, K_{n-1})$. Putting this all together we can write the normalization constant in the form of a sum-product algorithm, arranging the sums in the prescribed order and pushing all factors as far to the left as possible:

$$\begin{aligned} \mathcal{Z} = \sum_{K_T=0}^{R(2T-2)} & \left(\sum_{k_{T-1,1}=\max\{0, K_T-(2(T-2)-1)R\}}^{\min\{R, K_T\}} \right. \\ & \sum_{K_{T-1}=\max\{k_{T-1,1}, K_T-R\}}^{K_T} \\ & a_{[k_{T-1,1}, K_T-K_{T-1}]} b(K_{T-1}, K_T) \left(\cdots \left(\right.\right. \\ & \sum_{k_{3,1}=\max\{0, K_4-k_{4,1}-5R\}}^{\min\{R, K_4-k_{4,1}\}} \sum_{K_3=\max\{k_{3,1}, K_4-k_{4,1}-R\}}^{K_4-k_{4,1}} \\ & a_{[k_{3,1}, K_4-K_3-k_{4,1}]} b(K_3, K_4) \left(\right. \\ & \sum_{k_{2,1}=\max\{0, K_3-k_{3,1}-3R\}}^{\min\{R, K_3-k_{3,1}\}} \sum_{K_2=\max\{k_{2,1}, K_3-k_{3,1}-R\}}^{K_3-k_{3,1}} \\ & a_{[k_{2,1}, K_3-K_2-k_{3,1}]} b(K_2, K_3) \left(\right. \\ & \sum_{k_{1,1}=K_1=\max\{0, K_2-k_{2,1}-R\}}^{\min\{R, K_2-k_{2,1}\}} \\ & \left.\left.\left.\left.\left. a_{[k_{1,1}, K_2-K_1-k_{2,1}]} b(K_1, K_2) \right)\right)\right) \cdots \right)\right) \end{aligned} \tag{1}$$

Note that since we are no longer explicitly summing over $k_{i,2}$, it has been replaced by $K_{i+1} - K_i - k_{i+1,1}$

in the second element of the multi-index indexing the coefficients $\{a_k\}$. The forward variables $A^{(i,j)}$, then, can be immediately read off as follows. The first and second superscripts index values of $k_{t,1}$ and K_t, respectively:

$$A_2^{(i,j)} = \sum_{k=\max\{0,j-i-R\}}^{\min\{R,j-i\}} a_{[k,j-k-i]}\mathrm{B}(j-k,k)$$

$$i \in \{0,\cdots,R\}, j \in \{i,\cdots,i+2R\}$$

$$A_t^{(i,j)} = \sum_{k=\max\{0,j-i-(2t-3)R\}}^{\min\{R,j-i\}} \sum_{l=\max\{k,j-i-R\}}^{j-i}$$

$$a_{[k,j-l-i]}\mathrm{B}(j-l,l)A_{t-1}^{(k,l)}$$

$$i \in \{0,\cdots,R\}, j \in \{i,\cdots,i+2(t-1)R\}$$

$$A_T^{(j)} = \sum_{k=\max\{0,j-(2t-4)R\}}^{\min\{R,j\}} \sum_{l=\max\{k,j-R\}}^{j}$$

$$a_{[k,j-l]}\mathrm{B}(j-l,l)A_{T-1}^{(k,l)}$$

$$j \in \{0,\cdots,2(T-1)R\}$$

and $\mathcal{Z} = \sum_{i=0}^{2(T-1)R} A_T^{(i)}$. Similarly we can derive backward variables $C_t^{(i,j)}$, where the first superscript indexes $k_{t-1,2}$ and the second indexes $L_t \equiv \sum_{i=t}^{T} k_{i-1,2} + k_{i,1}$. Marginal quantities can be computed readily. The singleton marginal density is

$$p(x_t) = \frac{1}{\mathcal{Z}} \sum_{i=0}^{R} \sum_{j=i}^{i+2(t-2)R} \sum_{k=0}^{R} \sum_{l=k}^{k+2(T-1-t)R} A_{t-1}^{(i,j)} C_{t+1}^{(k,l)} \mathrm{B}(j,l) \sum_{k_{t-1,2}=0}^{R} \sum_{k_{t,1}=0}^{R} a_{[i,k_{t-1,2}]} a_{[k_{t,1},k]} x_t^{k_{t-1,2}+k_{t,1}} (1-x_t)^{j+l} \quad (2)$$

Therefore this method requires $O(R^2T^2)$ storage for the forward and backward variables, and $O(R^6T^2)$ processing time to compute marginals. For processing time, one factor of T comes from the number of marginals to compute. The other factor comes from the number of forward variables for each time, which increases linearly with T, manifesting in the limits of the sums over j and l in eq. (2).