7 Linear time series analysis

From a purely statistical point of view, one major difference between time series and data sets as discussed in the previous chapters is that temporally consecutive measurements are usually highly *dependent*, thus violating the assumption of identically and *independently* distributed observations on which most of conventional statistical inference relies. Before we dive deeper into this topic, we note that the independency assumption is not only violated in time series but also in a number of other common test situations. Hence, beyond the area of time series, statistical models and methods have been developed to deal with such scenarios. Most importantly, the assumption of independent observations is given up in the class of *mixed models* which combine fixed and random effects, and which are suited for both nested and longitudinal (i.e., time series) data (see, e.g., Khuri et al. 1998, West et al. 2006, for more details). Aarts et al. (2014) discuss these models specifically in the context of neuroscience, where dependent and nested data other than time series frequently occur, e.g., when we have recordings from multiple neurons, nested within animals, nested within treatment groups, thus introducing dependencies. Besides including random effects, mixed models can account for dependency by allowing for much more flexible (parameterized) forms for the involved covariance matrices. For instance, in a regression model like eq. 2.6 we may assume a *full* covariance matrix for the error terms (instead of the scalar form assumed in eq. 2.6) that captures some of the correlations among observations. Taking such a full covariance structure for $\mathbf{\Sigma}$ into account, under the multivariate normal model the ML estimator for parameters $\boldsymbol{\beta}$ becomes (West et al. 2006)

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{\Sigma}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{\Sigma}^{-1}\mathbf{y}\,, \tag{7.1}$$

as compared to the estimate given by eq. 2.5 for the scalar covariance. Note that because of the dependency, in this case the likelihood, eq. 1.14, doesn't factor into the individual observations anymore, but the result (7.1) can still easily be obtained if the observations are jointly multivariate normal. The estimation of the covariance matrices in this class of models is generally less straightforward, however. In general there is no analytical solution, and hence numerical techniques (as described in sect. 1.4) have to be afforded.

From a more general, scientific point of view, time series are highly interesting in their own right as they were supposedly generated by some underlying *dynamical system* that is to be recovered from the data, and which encapsulates the essence of our formal understanding of the underlying process. Often the assumption is that this dynamical (time series) model captures all the dependencies among consecutive data points, such that the residuals from this model are independent again, and hence conventional asymptotic test statistics can more or less directly be revoked. The simplest class of such time series models is *linear*, i.e. consists of (sets of) linear difference or differential equations, as introduced in detail further below. These follow pretty much the same mathematical layout as conventional multiple or multivariate regression models, only that output variables are regressed on time-lagged versions of their own, instead of on a different (independent) set of observations, thus catching the correlations among temporally consecutive measurements.

In many if not most domains of neuroscience, time series models are indeed the most important class of statistical models. Data from functional magnetic resonance imaging (fMRI) recordings, optical imaging, multiple/single-unit recordings, electroencephalography (EEG), or magnetoencephalography (MEG) signals inherently come as time series generated by a dynamical system, the brain, with – depending on the type of signal recorded – stronger or weaker temporal dependencies among consecutive measurements. Also in behavioral data, time series frequently occur, for instance whenever we investigate a learning process that develops across trials, or when we try to assess the impact of cyclic (e.g. hormonal) variations on behavioral performance. Before we get into all that, however, a few basic terms and descriptive statistical tools will be discussed. The introductory material in sects. 7.1 & 7.2 is mainly based on Chatfield (2004), Lütkepohl (2006), and Fan & Yao (2003), as are some bits in sect. 7.4, but the classic text by Box and Jenkins (Box et al., 2008, in the 4th edition) should be mentioned here as well.

7.1 Basic descriptive tools and terms

Auto-correlation

The most common tools for descriptive characterization of (the linear properties of) time series are the auto-correlation function and its "flip-side", the power spectrum (Chatfield 2004; van Drongelen 2007). Given a univariate time series $\{x_t\}$, i.e. variable x sampled at discrete times t (in the case of a time-continuous function we will use the notation $x(t)$ instead), the auto-covariance (acov) function is simply the conventional covariance applied to time-lagged versions of x_t:

$$\text{(7.2)} \quad \text{acov}(x_t, x_{t+\Delta t}) \equiv \gamma(x_t, x_{t+\Delta t}) := E[(x_t - \mu_t)(x_{t+\Delta t} - \mu_{t+\Delta t})],$$

with μ_t and $\mu_{t+\Delta t}$ the means at times t and $t+\Delta t$, respectively. As usual, the auto-*correlation* acorr is obtained by dividing the auto-covariance by the product of standard deviations:

$$\text{(7.3)} \quad \text{acorr}(x_t, x_{t+\Delta t}) \equiv \rho(x_t, x_{t+\Delta t}) := \frac{\text{acov}(x_t, x_{t+\Delta t})}{\sqrt{\text{var}(x_t)\,\text{var}(x_{t+\Delta t})}} = \frac{\gamma(x_t, x_{t+\Delta t})}{\sigma_t \sigma_{t+\Delta t}}.$$

Note that these definitions are based on the idea that we have access to an *ensemble* of time series drawn from the same underlying process, across which we take the expectancies and (co-)variances at specified times t. For obtaining estimates $\hat{\gamma}(x_t, x_{t+\Delta t})$ and $\hat{\rho}(x_t, x_{t+\Delta t})$ from a *single* observed time series $\{x_t\}$, $t = 1..T$ (i.e., of length T), one usually assumes *stationarity* and *ergodicity* (see below). In that case, estimates across samples can be replaced by estimates across time, the mean and variance are the same across all t, i.e. $\mu_t = \mu_{t+\Delta t} = \mu$ and $\sigma_t^2 = \sigma_{t+\Delta t}^2 = \sigma^2$, and the acorr and acov functions depend on time lag Δt only, i.e. $\gamma(x_t, x_{t+\Delta t}) = \gamma(\Delta t)$ and $\rho(x_t, x_{t+\Delta t}) = \rho(\Delta t) = \gamma(\Delta t)/\gamma(0)$. Parameters μ and σ^2 would then be replaced by their respective sample estimates $\bar{x}$ and s_x^2. Strictly, one would also have to acknowledge the fact that any time lag $\Delta t \neq 0$ cuts off Δt values at one end or the other of the empirical time series sample. Hence, one would compute in the denominator the product of standard deviations obtained across the first $1..T\text{-}\Delta t$ and the last $\Delta t{+}1..T$ values (and likewise for the means), but in practice this technicality is usually ignored (and irrelevant for sufficiently long time series).

The acorr-function (7.3) describes the *dependencies* among temporally neighboring values along a time series, and how quickly with time these dependencies die out (i.e., the

acorr drops to zero as Δt increases), and is thus an important tool to characterize some of the temporal structure in a time series. Fig. 7.1 illustrates its application on different types of neural time series, including series of inter-spike-intervals obtained from single-unit recordings (Fig. 7.1, top-row) and fMRI BOLD signal traces (Fig. 7.1, bottom-row). As can be seen, the auto-correlative properties in these different types of data are quite different. In general, the auto-correlation function can already inform us about some important properties of the underlying system, e.g. oscillations (indicated by periodic increases and decreases in the auto-correlation, as in Fig. 7.1, bottom) or 'long-memory' properties (indicated by a very slow decay of the auto-correlation; Jensen 1998). Note that by definition, just as the standard Pearson correlation, the acorr function is bounded within [-1,+1] and is symmetrical, i.e. $\rho(x_t, x_{t+\Delta t}) = \rho(x_{t+\Delta t}, x_t)$, or $\rho(\Delta t) = \rho(-\Delta t)$ in the stationary case. Given i.i.d. random numbers $\{x_t\}$ and some basic conditions, it can be shown that *asymptotically* (see Kendall et al. 1983; Chatfield 2004)

(7.4) $\hat{\rho}(\Delta t) \sim N(-1/T, 1/T)$,

which can be used to establish confidence bounds or check for significance of the auto-correlations.

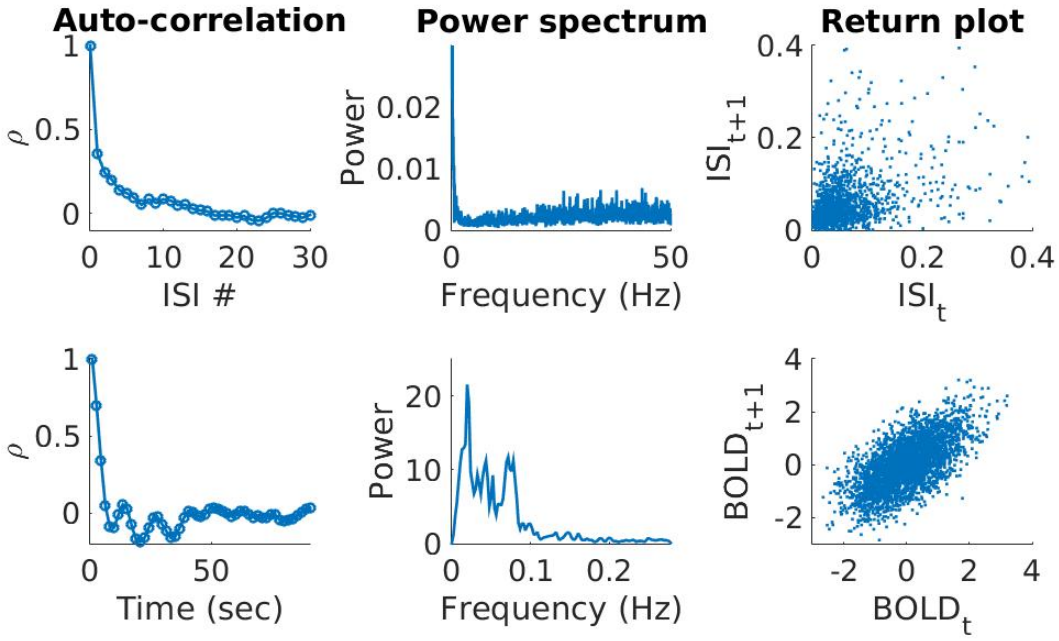

Fig. 7.1. Illustration of sample auto-correlation functions (left), power spectra (center) and return plots (right) on inter-spike-interval (ISI) series (top row; from rat prefrontal cortex) and BOLD signals (bottom row) from human fMRI recordings. For the spike data, the power spectrum was computed on the original binned (at 10 ms) spike trains, not the ISI series. Spike train data recorded by Christopher Lapish, Indiana University Purdue University Indianapolis (see also Lapish et al. 2008, Balaguer-Ballester et al. 2011). Human fMRI recordings obtained by Florian Bähner, Central Institute for Mental Health Mannheim (Bähner et al. 2015). **MATL7_1**.

Power spectrum

There is also – according to the Wiener-Khintchin theorem – a 1:1 relationship between the acorr-function and the so-called *power spectrum* (or spectral density) of a time series, provided it is weak-sense stationary (see below) and satisfies certain conditions, i.e. if you know one you know the other (van Drongelen 2007). Loosely, the power spectrum of a time series describes its decomposition into a weighted sum of harmonic oscillations, i.e. pure sine and cosine functions. More specifically, the *frequency domain representation* of a

periodic function $x(t)$ (i.e., one for which $x(t)=x(t+\Delta t)$ for some fixed Δt and all t) gives its approximation by a series of frequencies (the so-called *Fourier series*) as (van Drongelen 2007)

$$(7.5) \quad x(t) \approx \frac{a_0}{2} + \sum_{k=1}^{\infty} \left[a_k \cos(\omega k t) + b_k \sin(\omega k t) \right] = \sum_{k=-\infty}^{\infty} c_k e^{i \omega k t} ,$$

where $\omega=2\pi f$ is the angular frequency, $f=1/\Delta t$ the oscillation frequency in Hz (Δt = oscillation period), and $i = \sqrt{-1}$ is the complex number i (under certain, practically not too restrictive conditions, *Dirichlet's conditions*, the Fourier series is known to converge to $x(t)$). The power spectrum plots the coefficients $(a_k^2 + b_k^2)/2$ against frequency ω or f, and quantifies the energy contribution of each frequency f to the 'total energy' in the signal. In statistical terms, the first coefficient $a_0/2$ in the expansion (7.5) simply gives the *mean* of $x(t)$ across one oscillation period Δt, and the power $(a_k^2 + b_k^2)/2$ of the k^{th} frequency component is the *amount of variance* in the signal explained by that frequency (Chatfield 2004; van Drongelen 2007). In practice, an estimate of these functions is most commonly obtained by an algorithm called the *Fast Fourier Transform* (FFT). Whole textbooks have been filled with frequency domain analysis, Fourier transforms, and the various potential pitfalls and caveats that come with their estimation from empirical time series (see, e.g., van Drongelen, 2007, for an excellent introduction targeted specifically to neuroscientists). Here we will therefore not dive too much into this extensive topic, but rather stay with the main objective of this book of giving an overview over a variety of different statistical techniques. In anticipation of the material covered in Ch. 8 & 9, it may also be important to note that the Fourier transformation of $x(t)$ only captures its *linear* properties (as fully specified through the acorr function).

In neuroscience, the frequency domain representation of neurophysiological signals like the local field potential (LFP) or the EEG has been of uttermost importance for characterizing oscillatory neural processes in different frequency bands, e.g. the theta (~3-7 Hz) or gamma (~30-80 Hz) band (Buzsaki & Draguhn 2004). Oscillations are assumed to play a pivotal role in neural information processing, e.g. as means for synchronizing the activity and information transfer between distant brain areas (e.g. Engel et al. 2001; Jones & Wilson 2005), or as a carrier signal for phase codes of external events or internal representations (e.g. Hopfield & Brody 2001; Brody & Hopfield 2003; Buzsaki 2011). For instance, stimulus-specific increases in the power within the gamma or theta frequency band have been described both in response to *external* stimuli, e.g. in the bee olfactory system in response to biologically relevant odors (Stopfer et al. 1997), and in conjunction with the *internal* active maintenance of memory items, e.g. during the delay phase of a working memory task (Pesaran et al. 2002; Lee et al. 2005). Neurons in the hippocampus coding for specific places in an environment have been described to align their spiking activity with a specific phase of the hippocampal theta rhythm while the animal moves through the neuron's preferred place field, thus encoding environmental information in the *relative phase* (forming a *phase code*) with respect to an underlying oscillation (see Fig. 9.20; Buzsaki 2011; Harris et al. 2003). Likewise, Lee et al. (2005) have shown that neurons in visual cortex may encode and maintain information about visual patterns in working memory by aligning their spike phase with an underlying theta oscillation during the delay period; this, again, occurred in a stimulus-specific manner with the phase relationship breaking down for items not preferred by the recorded cell. Jones and Wilson (2005) discovered that the hippocampus and prefrontal cortex phase-lock (see sect. 9.2.2) during working memory tasks, especially during the choice epochs where the animal

chooses the response in a two-arm maze based on previous choices or stimuli; thus, oscillations may help to organize the information transfer among areas. These are just a few examples that highlight the importance of the analysis of oscillatory activity in neuroscience, the literature on this topic is extensive (e.g. Buzsaki 2011; Traub & Whittington 2010). Fig. 7.1 (center) illustrates the representation of the spike train and BOLD time series from Fig. 7.1 (left) as power spectra in the frequency domain.

White noise

The simplest form of a time series process $\{x_t\}$ is a pure *random process* with zero mean and fixed variance but no temporal correlations at all, that is we may have $E[x_t]=0$ for all t and

$$(7.6) \quad E[x_t x_{t'}] = \begin{cases} \sigma^2 & for\ t = t' \\ 0 & otherwise \end{cases}.$$

Such processes are called *white noise* processes (Fan & Yao 2003), abbreviated $W(0,\sigma^2)$ here, since in the frequency domain representation discussed above, there would be no distinguished frequency: Their power spectrum is completely flat, no specific 'color' would stick out but there would be a uniform mixture of all possible colors, giving white (but note that $W(0,\sigma^2)$ is not necessarily Gaussian). Thus, in accordance with the Wiener- Khintchin theorem, it is the unique setup of auto-correlation coefficients at different time lags $\Delta t \neq 0$ which give the time series its oscillatory properties – if they are all zero, there are no (linear) oscillations. For most of the statistical inference on time series the assumption is that the residuals from a model form a white noise sequence. In fact, according to the Wold decomposition theorem, each stationary (see below) discrete-time process $x_t = z_t + \eta_t$ can be split into a systematic (purely deterministic) part z_t and an uncorrelated purely stochastic process $\eta_t = \sum_{k=0}^{\infty} b_k \varepsilon_{t-k}$ where $\varepsilon_t \sim W(0,\sigma^2)$ (Chatfield 2004).

Often one would assume *Gaussian* white noise, i.e. $\varepsilon_t \sim N(0,\sigma^2)$, $E[\varepsilon_t \varepsilon_{t'}]=0$ for $t \neq t'$. One could explicitly check for this assumption by comparing the empirical ε_t distribution to a Gaussian using common Kolmogorov-Smirnov or χ^2 based test statistics, and evaluating whether any of the auto-correlations significantly deviates from 0 (or -1/T, see eq. 7.4) for $\Delta t \neq 0$ (recall that moments up to 2nd order completely specify a white noise process in general and the Gaussian in particular). Alternatively, one may evaluate whether the power spectrum conforms to a uniform distribution. Or one could employ more general tests for randomness in the time series by checking for any sort of sequential dependencies (Kendall et al. 1983; Chatfield 2004). For instance, one may discretize (bin) ε_t, chart the transition frequencies among different bins, and compare them to the expected base rates under independence using e.g. χ^2 tables. One could also examine the binned series for unusually long runs of specific bin-values, based on the binomial or multinomial distribution (Kendall et al. 1983; Wackerly et al. 2008). Another possibility is to chart the intervals between successive maxima (or minima) of a real-valued series – the length of an interval I_i between any two successive maxima should be independent of the length I_{i-1} of the previous interval for a pure random process, i.e. $p(I_i \mid I_{i-1})=p(I_i)$. One could get a visual idea of whether this holds by plotting all pairs (I_i, I_{i-1}) (sometimes called a 'first-return plot') and inspecting the graph for systematic trends in the distribution (Fan & Yao 2003; Fig. 7.1, right column, illustrates this for the inter-spike-interval [*ISI*] and BOLD time series). Durstewitz & Gabriel (2007) used this to examine whether single neuron ISI series recorded under different pharmacological conditions exhibit any evidence of deterministic structure, or whether they are indeed largely random as suggested by the common

Poisson assumption of neural spiking statistics (Shadlen & Newsome 1998). More formally, a significant regression coefficient relating I_i to I_{i-1} would shed doubt on the assumption of independence. In general, there are really a number of different informal checks or formal tests one may think of in this context (see Kendall et al. 1983; Chatfield 2004).

Stationarity and ergodicity

A fundamental concept (for model estimation and inference) in time series analysis is that of *stationarity*, which roughly means that properties of the time series do not change across time. In statistical terms, one commonly distinguishes between *weak sense* and *strong stationarity* (Fan & Yao 2003), where the former is defined by the conditions

(7.7) $E[x_t] = \mu = const.,\ \text{acov}(x_t, x_{t+\Delta t}) = \text{acov}(\Delta t)\ \forall t, \Delta t$ (weak stationarity),

i.e. the mean is constant and independent of time, and the acov (acorr)-function is a function of time lag only but does not change with *t* either. The stronger form of stationarity requires that the joint distribution *F* of the $\{x_t\}$ is time-invariant,

(7.8) $F(\{x_t \mid t_0 \le t < t_1\}) = F(\{x_t \mid t_0 + \Delta t \le t < t_1 + \Delta t\})$ for all t_0, t_1, and Δt (strong stationarity),

which implies that all higher-order moments of the $\{x_t\}$-distribution must be independent of *t* as well (equivalent to eq. 7.7 for a purely Gaussian process). It is important to note that these definitions assume that we have access to a large sample of time series $\{x_t\}^{(i)}$ generated by the same underlying process, from which we take the expectancies across all series *i* at time *t*, for instance to evaluate the first moments $E_i[x_t^{(i)}] = \lim_{N\to\infty} \sum_{i=1}^{N} x_t^{(i)} / N$.

Thus, the definition does not exclude *conditional dependence* in the series, i.e. we may have $E[x_t \mid x_{t-1}] \neq E[x_t \mid x'_{t-1}]$ for $x_{t-1} \neq x'_{t-1}$. In fact, this is central for identifying periodic (like harmonic oscillatory) processes as stationary where x_t may indeed systematically change across time. For instance, we may deal with a time series generated by the harmonic oscillatory process with noise (cf. Fan & Yao 2003)

(7.9) $x_t^{(i)} = \sin(2\pi f t + \varphi_i) + \varepsilon_t\ ,\ \varepsilon_t \sim N(0, \sigma^2)$.

Treating φ_i as a random variable across different realizations $\{x_t\}^{(i)}$ of the process, we still have $E[x_t]=const$ for all *t*, although consecutive values x_t in time are conditionally dependent as defined through the sine function (the systematic part; Fan & Yao 2003; Chatfield 2004).

This already hints to some of the problems we may encounter in practice if we would like to establish stationarity empirically. Commonly we may have access only to one realization of the time series process, and hence in practice we often employ a principle called *ergodicity*, which means that estimates across different independent realizations of the same process at fixed *t* could be replaced by estimates across time. Thus, taking the mean, for instance, we assume $E_i[x_t^{(i)}] = E_t[x_i^{(t)}]$, and likewise for ergodicity in the variance we would require $E_i[(x_t^{(i)} - \bar{x}_t^{(i)})^2] = E_t[(x_i^{(t)} - \bar{x}_i^{(t)})^2]$, where the first expectation is meant to be taken across sample series *i* (fixed *t*) and the second across time points *t* (fixed *i*). Given that time series data are commonly not i.i.d. but governed by auto-correlations, it is not at all evident that such properties hold. A sufficient condition for a stationary process to be ergodic in the mean is, however, that the auto-correlations die out to zero as the lag

increases. But auto-correlations still affect the sampling distribution of a time series mean $\bar{x}$ estimated from a finite series of length *T*, with its squared standard error given by (Fan & Yao 2003; Chatfield 2004)

$$(7.10)\quad E[(\bar{x}_T - \mu)^2] = \frac{\sigma^2}{T}[1 + 2\sum_{\Delta t=1}^{T-1}(1 - \frac{\Delta t}{T})\rho(\Delta t)]\,.$$

Thus, unlike the conventional i.i.d. case (def. 1.4), if we would like to obtain an unbiased estimate of the standard error of $\bar{x}$ from a single time series $\{x_t\}$, we would have to acknowledge these auto-correlations. This is a reflection of the more general issue that in time series we are dealing with dependent data, hence violating a crucial assumption of most conventional statistics.

Another problem is that, *empirically*, what we consider as stationary also depends on our observation period *T* – something that may appear non-stationary on short time scales may be stationary on longer scales, e.g. if *T* is brief compared to the period of an underlying oscillation. Finally, there may be other ways of defining stationarity: We may for instance call a time series stationary if the *generating process* has time-invariant parameters, e.g. if we have a process $x_t = f_{\boldsymbol{\theta}}(x_{t-1}) + \varepsilon_t$ where the parameter set $\boldsymbol{\theta}$ is constant. It is not clear whether such a definition is generally consistent with def. (7.7) or (7.8). A dynamical system (see Ch. 9) with constant parameters may generate time series which potentially violate the above *statistical* definition of stationarity, for instance if the dynamical system possesses multiple co-existing attractor states characterized by different distributions among which it may hop due to perturbations (see sect. 9.1, 9.2). Vice versa, a process with time-varying parameters $\boldsymbol{\theta}$ might still be stationary according to defs. (7.7) and (7.8) if the parameters at each point in time are themselves drawn from a stationary distribution.

In the experimental literature, different tests have been proposed to directly check whether statistical moments of the time series stay within certain confidence limits across time: For instance, Quiroga-Lombard et al. (2013) developed a formal test based on def. (7.7) which first standardizes and transforms the observed quantities (in their case, interspike-intervals [ISI]) through the Box-Cox transform (Box & Cox 1964) to bring their distribution into closer agreement with a standard Gaussian, and then checks within sliding windows of *k* consecutive variables whether the local average and standardized sum of squares fall outside predefined confidence bounds of the normal and χ^2-distribution estimated from the full series, respectively (Fig. 7.2). The test ignores auto-correlations in the series (see eq. 7.10), however, which for ISI series in-vivo often decay rapidly (e.g. Quiroga-Lombard et al. 2013). In Durstewitz & Gabriel (2007) Kolmogorov-Smirnov tests were used to check whether distributions across a set of consecutive samples of ISI series significantly deviate from each other. In the context of time series models, non-stationarity may also be recognized from the estimated coefficients of the model as detailed further below (sect. 7.2.1).

One obvious type of non-stationarity is a systematic trend across time (where we caution again that a slow oscillation, for instance, may look like a trend on shorter time scales). This may be indicated by having a lot of power in the lowest frequency bands or, equivalently, having very long-term auto-correlations. There are at least three different ways of removing a systematic trend, oscillations, or other forms of non-stationarity and undesired confounds (see Chatfield 2004; Box et al. 2008):

1) We may fit a parametric or non-parametric model to the data (e.g. a linear regression model, a locally linear regression, or a spline model) and then work from the residuals, i.e. after removing the trend, oscillation, or any other systematic component in the data that may spoil the process of interest.

2) We may remove trends or oscillations in the frequency domain by designing a filter that takes out the slowest frequency bands, or any other prominent frequency band.

3) A third very common technique is differencing the time series as often as required. For instance, a non-stationary time series $\{x_t\}$ may be transformed into a stationary one by considering the series of first-order differences $\{x_{t+1} - x_t\}$. In some cases, higher-order differencing may be required to make the series stationary.

Sometimes transformations of the data to stabilize the variance (e.g. a log-transform) or to move them towards a normal distribution (e.g. Box-Cox transforms) may also help (Chatfield 2004; Yu et al. 2009). Any of these techniques should be used carefully, as they could potentially also lead to spurious phenomena (e.g. induce oscillations) or inflate the noise.

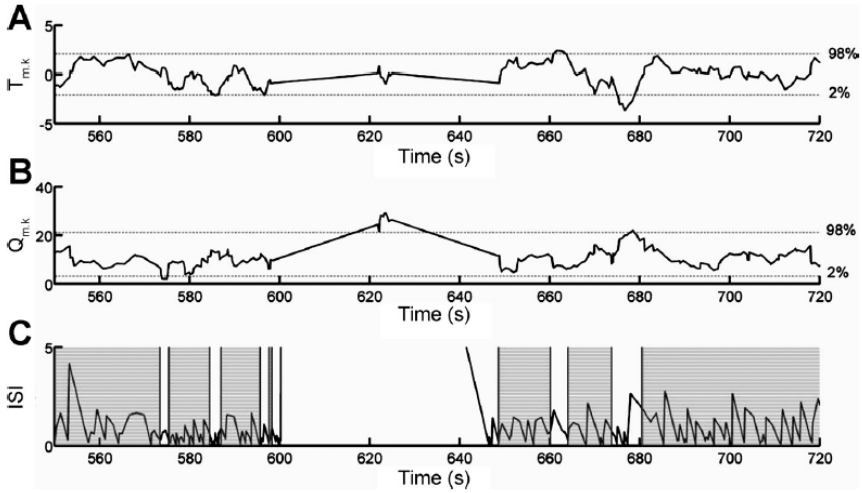

Fig. 7.2. Dissecting spike trains into stationary segments. A) Running estimate of test statistic $T_{m,k}$ comparing the local average to the grand average of the series on sliding windows of 10 Box-Cox-transformed interspike-intervals (ISIs), with [2%,98%] confidence bands. B) Running estimate of χ^2-distributed statistic $Q_{m,k}$ evaluating the variation of the local ISIs around the grand average, with [2%,98%] confidence bands. C) Original ISI series with resulting set of jointly stationary segments in gray shading. Reprinted from Quiroga-Lombard et al. (2013), Copyright (2013) by The American Physiological Society, with permission.

Multivariate time series

The concepts introduced above can easily be generalized to *multivariate* time series. In this case, instead of auto-covariance and auto-correlation functions we would be dealing with *cross-covariance* and *cross-correlation* functions (with analogue measures like *coherence* defined in the frequency domain; see van Drongelen 2007). That is, for each time lag Δt, we would have a covariance matrix $\Gamma(\Delta t)=[\gamma_{ij}(\Delta t)]$ among different time series variables indexed by *i* and *j*, with elements $\gamma_{ij}(\Delta t) := E[(x_{it} - \mu_{it})(x_{j,t+\Delta t} - \mu_{j,t+\Delta t})]$. Hence, diagonal entries of $\Gamma(\Delta t)$ would be the usual auto-covariance functions while off-diagonal entries would indicate the temporal coupling among different time series at the specified lags. This may introduce additional issues, however, which one has to be careful about. For instance, strong auto-correlations may inflate estimates of cross-correlations and lead to spurious results for time series which are truly independent (Chatfield 2004).

The analysis of cross-correlations among experimentally recorded single-unit activities is one of the most important neurophysiological applications, and has been fundamental in theories of neural coding and functional dynamics in the nervous system. Peaks in the spike-time cross-correlation function, when plotting $\hat{\gamma}_{ij}(\Delta t)$ as a function of Δt,

have been interpreted as indication of the underlying connectivity, i.e. the sign (excitatory or inhibitory) and potential direction (from time lag Δt) of neural connections (strictly, however, directedness cannot be inferred from $\hat{\gamma}_{ij}(\Delta t)$ alone, see sects. 7.4 & 9.5). Such physiological information may hence be used to reconstruct the underlying network structure (e.g. Aertsen et al. 1989; Fujisawa et al. 2008; Pernice et al. 2011). However, neural cross-correlations are found to be highly dynamic and may change with behavioral task epochs (Vaadia et al. 1995; Funahashi & Inoue 2000; Fujisawa et al. 2008) and stimulus conditions (Gray et al. 1989). Thus, they may only partly reflect anatomical connectivity as proposed in the influential concept of a synfire chain where synchronized spike clusters travel along chains of feedforward-connected neurons (Abeles 1991; Diesmann et al. 1999). Rather, spike-time cross-correlations may indicate more the *functional connectivity* (Aertsen et al. 1989) and have been interpreted as a signature of the transient grouping of neurons into functional (cell) assemblies representing perceptual or internal mental entities (Hebb 1949; Harris et al. 2003; Singer & Gray 1995). For instance, von der Malsburg and Singer (Singer & Gray 1995) suggested that the precisely synchronized alignment of spiking times as reflected by significant zero-lag peaks in the cross-correlation function could serve to 'bind' different features of sensory objects into a common representation, while at the same time segregating it in time from other co-active representations (as in foreground-background separation in a visual scene) through anti-correlations (i.e., peaks at $\Delta t=\pi$ or at least $\Delta t\neq 0$).

The functional interpretation of neural cross-correlations is, however, not without problems, and has been hampered by a number of experimental and statistical issues (Brody 1998, 1999; Grün 2009; Quiroga-Lombard et al. 2013). For one thing, it relies on the validity of the spike sorting process, i.e. the preceding numerical process (still partly performed 'by hand') by which patterns in the recorded extracellular signals are identified as spike waveforms and assigned to different neurons (Lewicki 1998; Einevoll et al. 2012). Obviously, incorrect assignments can give rise to both artifactual correlations (e.g., when the same signal is wrongly attributed to different units) as well as the loss of precise spike time relations. Non-stationarity across presumably identical trials, or within trials, can be another source of error that could induce apparent sharp spike-time correlations where there are none (Brody 1998, 1999). Potential non-stationarities therefore have to be taken care of in the analysis of cross-correlations, e.g. by using sliding windows across which the process can safely be assumed to be (locally) stationary (e.g. Grün et al. 2002b), or by explicitly removing them from the cross-correlation function (e.g. Quiroga-Lombard et al. 2013; Fig. 7.3).

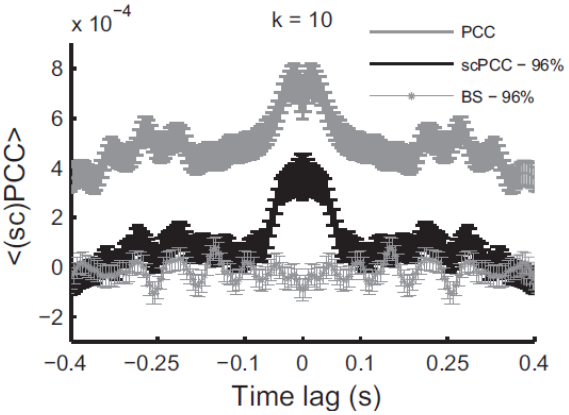

Fig. 7.3. Example of Pearson spike-time cross-correlogram from prefrontal cortical neurons. 'Raw' spike-time cross-correlogram (PCC) in bold gray, stationarity-corrected Pearson cross-correlogram (scPCC) in black, and cross-correlogram from block permutation bootstraps in thin gray. Reprinted from Quiroga-Lombard et al. (2013),

7.2 Linear time series models

In its most general form, a linear time series model assumes that observations x_t depend on a linear combination of past values (the so-called auto-regressive, AR, part) and of present and past noise inputs (the so-called moving-average, MA, part; Fan & Yao 2003; Chatfield 2004; Box et al. 2008):

$$(7.11)\quad x_t = a_0 + \sum_{i=1}^{p} a_i x_{t-i} + \sum_{j=0}^{q} b_j \varepsilon_{t-j}\ ,\ \varepsilon_t \sim W(0,\sigma^2)\ .$$

Parameters p and q determine the order of the model (how much in time 'it looks back'; also written as ARMA(p,q) model), while the sets of coefficients $\{a_i\}$ and $\{b_j\}$ determine the influence past (or present noise) values have on the current state of the system. As one may guess, these coefficients are strictly related to the (partial) auto-correlations of the time series as shown further below. There are several things we might want to do now: Given an empirically observed time series $\{x_t\}$, we may want to evaluate whether a linear model like (7.11) is appropriate at all, whether it gives rise to a stationary or a non-stationary time series, what the proper orders p and q are, what the coefficients $\{a_i\}$ and $\{b_j\}$ are, and we may want to test specific hypotheses on the model, e.g. whether certain coefficients significantly deviate from zero or from each other. Before we come to that, however, it may be useful to expose some basic properties of this class of models (based on Chatfield, 2004, Lütkepohl, 2006, and Fan & Yao, 2003), specifically their relationship to the acorr-function and the relation between AR and MA parts. ARMA models are integral building blocks of linear state space models (sect. 7.5.1) and linear implementations of the Granger causality concept (sect. 7.4) through which they have found widespread applications in neuroscience. They have also frequently been employed as tools to generate null hypothesis distributions (sect. 7.7, Ch. 8), as time series of interest in neuroscience are usually not linear.

There is a basic duality between pure AR and pure MA models: Any AR model of order p can be equivalently expressed as an MA model of infinite order as can easily be seen by recursively substituting previous values of x_t in the equation (Chatfield 2004; Lütkepohl 2006). For instance, let

$$(7.12)\quad x_t = a_0 + a_1 x_{t-1} + \varepsilon_t\ ,\ \varepsilon_t \sim W(0,\sigma^2),$$

an AR(1) model, and assume, for simplicity, that we start the series at $x_1 = a_0 + \varepsilon_1$. Then we could expand this into

$$(7.13)\quad \begin{aligned} x_t &= a_0 + a_1 x_{t-1} + \varepsilon_t = a_0 + a_1(a_0 + a_1 x_{t-2} + \varepsilon_{t-1}) + \varepsilon_t \\ &= a_0 + a_1(a_0 + a_1(a_0 + a_1 x_{t-3} + \varepsilon_{t-2}) + \varepsilon_{t-1}) + \varepsilon_t = \ldots = a_0 \sum_{i=0}^{t-1} a_1^i + \sum_{i=0}^{t-1} a_1^i \varepsilon_{t-i}\ . \end{aligned}$$

Hence we have rewritten (7.12) in terms of an (ultimately, for $t\to\infty$) infinite order MA model. Note that the expectancy of x_t, E[x_t], is given by a *geometric series*, since E[ε_t]=0, which converges only for $|a_1|<1$, namely to $a_0/(1-a_1)$ for $t\to\infty$ (Fig. 7.4; Chatfield 2004; Lütkepohl

2006). More generally, if for an AR model we have $\left|\sum a_i\right| \geq 1$, x_t will systematically drift or grow across time, and the process is *non-stationary* (i.e., will exhibit trend)! In fact, in the example above, for a_1=1 we have what is called a *random walk*: The process will just randomly be driven around by the noise (Fig. 7.4, right) plus a systematic drift imposed by a_0, while for $|a_1|>1$ x_t will exponentially grow (Fig. 7.4, center)!

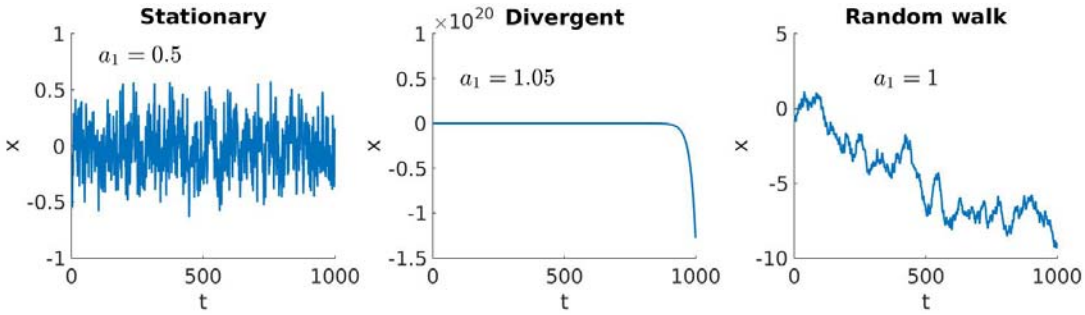

Fig. 7.4. Time series from stationary (left), divergent (center), and random walk (right) AR(1) processes with a_0=0. **MATL7_2**.

Conversely, any pure MA model of order *q* could equivalently be expressed as an infinite order (for $t\rightarrow\infty$) AR process; for instance, expanding an MA(1) process (and starting at $x_1 = \varepsilon_1$) we get

$$(7.14)\quad x_t = \varepsilon_t + b_1\varepsilon_{t-1} = \varepsilon_t + b_1(x_{t-1} - b_1\varepsilon_{t-2}) = \varepsilon_t + b_1(x_{t-1} - b_1(x_{t-2} - b_1\varepsilon_{t-3})) = \ldots = \sum_{i=1}^{t-1} b_1^i x_{t-i} + \varepsilon_t .$$

To simplify notation and derivations, the so-called backward-shift operator *B* defined by (Chatfield 2004; Lütkepohl 2006)

$$(7.15)\quad B^j x_t = x_{t-j}$$

was introduced. This allows to express any ARMA(*p*,*q*) model in the form (Chatfield 2004)

$$(7.16)\quad f(B)x_t = g(B)\varepsilon_t \quad with \quad f(B) = 1 - \sum_{i=1}^{p} a_i B^i \;\; and \;\; g(B) = 1 + \sum_{j=1}^{q} b_j B^j \quad .$$

The relationship between AR or MA models and the acov-function can be seen by multiplying left and right hand sides of (7.12) through by time-lagged versions of x_t and taking expectations (Chatfield 2004). Let us assume a_0=0, in which case we take from (7.13) that $E[x_t] = 0$. For a stationary AR(1) model of the form (7.12) we then get

$$(7.17)\quad E[x_t x_{t-1}] = E[a_1 x_{t-1} x_{t-1}] + E[\varepsilon_t x_{t-1}] = a_1 E[x_{t-1} x_{t-1}] .$$

The term $E[\varepsilon_t x_{t-1}]$ evaluates to 0 since we assumed ε_t to be a white noise process, and since x_{t-1} can be expressed as an infinite sum of previous ε_{t-1} … $\varepsilon_{t-\infty}$ terms (which by definition are uncorrelated with ε_t). Thus we obtain the simple relationship (assuming the process is stationary)

(7.18) $\mathrm{aco}v(1) = a_1 \mathrm{aco}v(0)$.

Repeating the steps above, multiplying through with x_{t-2}, we obtain

(7.19) $E[x_t x_{t-2}] = E[a_1 x_{t-1} x_{t-2}] + E[\varepsilon_t x_{t-2}] \Rightarrow \mathrm{aco}v(2) = a_1 \mathrm{aco}v(1) = a_1^2 \mathrm{aco}v(0)$.

This leads into a set of equations termed *Yule-Walker equations* (Chatfield 2004; Lütkepohl 2006), and we may obtain a simple estimate of a_1 as

(7.20) $a_1 = \mathrm{aco}v(1) / \mathrm{aco}v(0) = \mathrm{aco}v(1) / \sigma^2 = acorr(1)$.

From (7.17-7.19) we also see that for an AR(1) model, auto-correlations simply exponentially decay with time lag Δt as $a_1^{\Delta t}$, while for an higher-order AR(p) model we may have a mixture of several overlaid exponential time courses.

Say we have an AR(p) process for which we regress out the effects of direct temporal neighbors x_{t-1} from x_t by performing the optimal AR(1) prediction. The correlation with the remaining auto-predictors is called the (first-order) *partial auto-correlation* (pacorr) *function* of the time series after removing the influence of x_{t-1} on x_t. Now note that we can do this at most p times, after which we are left with the pure noise process ε_t since x_t depends on earlier observations x_{t-q}, $q > p$, only through the preceding p values whose influence has been removed (Fan & Yao 2003). Thus, since the ε_t themselves are mutually independent, for lags > p the pacorr function must drop to 0. The important conclusion from all this is that, in principle, we could get an estimate of the order p of an AR process by examining the pacorr function of the time series (Fig. 7.5), and estimates of the parameters through the auto-correlations. However, in practice this is not recommended (Fan & Yao 2003; Chatfield 2004), since the Yule-Walker estimates always give rise to a stationary AR process (in fact presume it), although it really might be not (Lütkepohl 2006). For instance, as correlations are always bounded in [-1,1], for an AR(1) model with a_0=0, we would always end up with a stationary process unless the correlation is perfect, since for $|a_1|<1$ series (7.13) would always converge as explained above (only for a perfect correlation, |acorr(1)|=1, we would obtain a random walk or 'flipping' process).

Likewise, we could – in principle – determine the order q and coefficients of an MA process through the auto-correlations. For an MA(q) process,

(7.21) $x_t = \sum_{j=0}^{q} b_j \varepsilon_{t-j}$, $\varepsilon_t \sim W(0, \sigma^2)$.

Since the ε_t at different times are all uncorrelated, by multiplying through with ε_{t-q-1} and taking expectations (Chatfield 2004), we see that the acov function cuts off at lag q (i.e. all longer lag auto-correlations evaluate to 0 for such a process; Fig. 7.5).

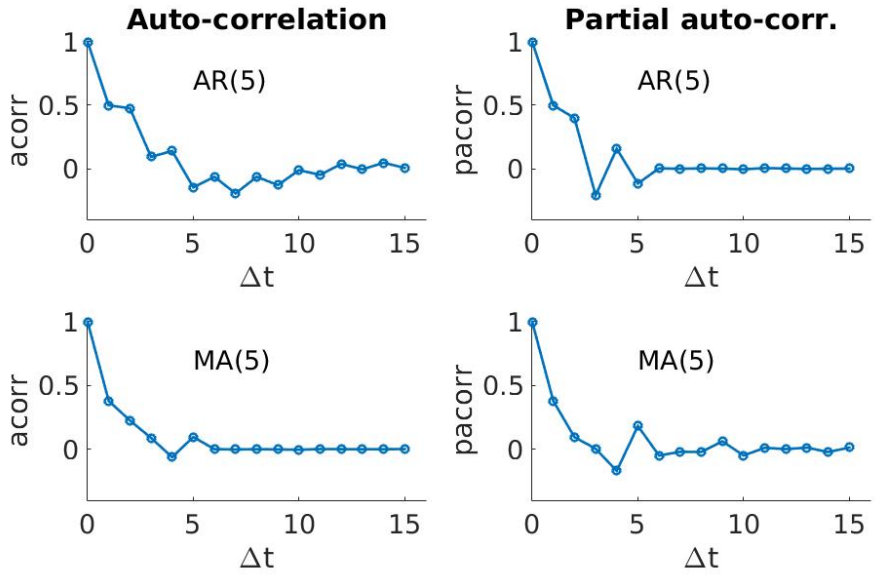

Fig. 7.5. Auto-correlation (left column) and partial auto-correlation (right column) function for an AR(5) process (top row) and a MA(5) process (bottom row). Note that the pacorr-function precisely cuts off after lag 5 for the AR(5) process [**a**=(0.5 0.3 -0.3 0.2 -0.2), b_0=0.2], while the acorr-function cuts off after lag 5 for the MA(5) process [**b**=(0.8 0.3 0.2 0.1 -0.1 0.1)]. **MATL7_3**.

Finally, once parameters of an ARMA process have been determined (see next section), forecasts $x_{t0+\Delta t}$ can be simply obtained from x_{t0} by iterating the estimated model Δt steps ahead into the future (formally, one seeks $E[x_{t0+\Delta t}]$ based on the estimated model, where $E[\varepsilon_t]=0$ for all $t>t_0$).

7.2.1 Estimation of parameters in AR models

We have established above a basic equivalence between AR and MA models, and for the following will therefore focus on pure AR models (for which parameter estimation is more straightforward than for MA models; although in practice a MA model might sometimes be the more parsimonious or convenient description). Thus we assume a model of the form

(7.22) $$x_t = a_0 + \sum_{i=1}^{p} a_i x_{t-i} + \varepsilon_t \;,\; \varepsilon_t \sim W(0,\sigma^2)$$

for the data. Collecting the last T-p observations of an observed time series $\{x_t\}$, t=1…T, in a vector $\mathbf{x}_T=(x_{p+1}\ldots x_T)^T$, arranging for each x_t in $\mathbf{x}_T$ the p preceding values $(x_{t-1}\ldots x_{t-p})$ in a $(T\text{-}p)\times p$ matrix $\mathbf{X}_p$ which we further augment by a leading column of 1s, this can be written as

(7.23) $$\mathbf{x}_T = \mathbf{X}_p\mathbf{a} + \boldsymbol{\varepsilon} \;,\; \boldsymbol{\varepsilon} \sim W(\mathbf{0},\sigma^2\mathbf{I})$$

with (p+1 x 1) coefficient vector $\mathbf{a}=(a_0\ldots a_p)^T$. Note that this has exactly the same form as the multiple regression model (2.6) with p predictors and a constant term. And indeed, the parameter estimation could proceed in the very same way by LSE or ML (Lütkepohl 2006; usually assuming Gaussian white noise for ML), yielding

(7.24) $$\mathbf{a} = (\mathbf{X}_p^T\mathbf{X}_p)^{-1}\mathbf{X}_p^T\mathbf{x}_T \,.$$

See Fig. 7.6 for an example.

Based on the same type of expansion of an AR model as in eq. 7.13, we can furthermore obtain the steady-state mean (in the limit of an infinitely long time series) of this process as

$$\begin{aligned}\lim_{T\to\infty} E[x_T] &= \lim_{T\to\infty} E\left[a_0 + \sum_{i=1}^{p} a_i x_{T-i} + \varepsilon_T\right] \\ &= \lim_{T\to\infty}\left(a_0 \sum_{t=0}^{T}\left[\sum_{i=1}^{p} a_i\right]^t + \sum_{t=0}^{T}\left[\sum_{i=1}^{p} a_i\right]^t E[\varepsilon_{T-t}]\right) = a_0\left(1-\sum_{i=1}^{p} a_i\right)^{-1},\end{aligned} \tag{7.25}$$

since by assumption $E[\varepsilon_t] = 0$ for all *t*, and provided the series converges.

It is also straightforward to generalize all this to the multivariate setting, where the multivariate AR model (also called a *vector auto-regressive*, VAR, model in this context) takes the form of a multivariate linear regression

$$\mathbf{x}_t = \mathbf{a}_0 + \sum_{i=1}^{p} \mathbf{A}_i \mathbf{x}_{t-i} + \boldsymbol{\varepsilon}_t \,,\, \boldsymbol{\varepsilon}_t \sim W(\mathbf{0}, \boldsymbol{\Sigma}) \,. \tag{7.26}$$

where $\mathbf{x}_t$ is a *K*-variate *column* vector (with *K* = number of time series, i.e., we arrange time across columns now, not rows), the $\mathbf{A}_i$ are full (*K* x *K*) coefficient matrices which also specify (linear) *interactions* among variables, and $\boldsymbol{\Sigma}$ is a full covariance matrix. Parameter estimation proceeds along the same lines as for the univariate model (7.22)-(7.25), and in accordance with the multivariate regression model described in sect. 2.2 (i.e., multivariate parameter estimation is given by the concatenation of the multiple regression solutions, and makes a real difference only for statistical testing) (Fig. 7.6). We furthermore note that any AR(*p*) or VAR(*p*) model can be reformulated as a *p*-variate VAR(1) or (*p* x *K*)-variate VAR(1) model, respectively, by concatenating the variables, vectors, or matrices on both sides of eqn. (7.22) or (7.26) the right way (see Lütkepohl 2006). E.g., an AR(2) model (ignoring offset a_0 for convenience) may be rewritten as (Lütkepohl 2006)

$$\mathbf{x}_t = \begin{bmatrix} x_t \\ x_{t-1} \end{bmatrix}, \mathbf{A} = \begin{bmatrix} a_1 & a_2 \\ 1 & 0 \end{bmatrix}, \boldsymbol{\varepsilon}_t = \begin{bmatrix} \varepsilon_t \\ 0 \end{bmatrix}, \mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \boldsymbol{\varepsilon}_t \,. \tag{7.27}$$

Hence, everything we derive below for AR(1) or VAR(1) models directly transfers to AR(*p*) and VAR(*p*) models, respectively.

The *stationarity (stability) condition* for the model as provided by convergence of the geometric series in (7.25) (requiring $\left|\sum_{i=1}^{p} a_i\right| < 1$), in the multivariate setting generalizes to the requirement that all *eigenvalues* of the transition matrix **A** must be smaller than 1 in absolute value (modulus), i.e. we must have (e.g. Lütkepohl 2006)

$$\max |\, eig(\mathbf{A}) \,| < 1 \quad \text{(stationarity condition)}. \tag{7.28}$$

MATL7_4 (Fig. 7.6) implements parameter estimation in multivariate (vector) AR models.

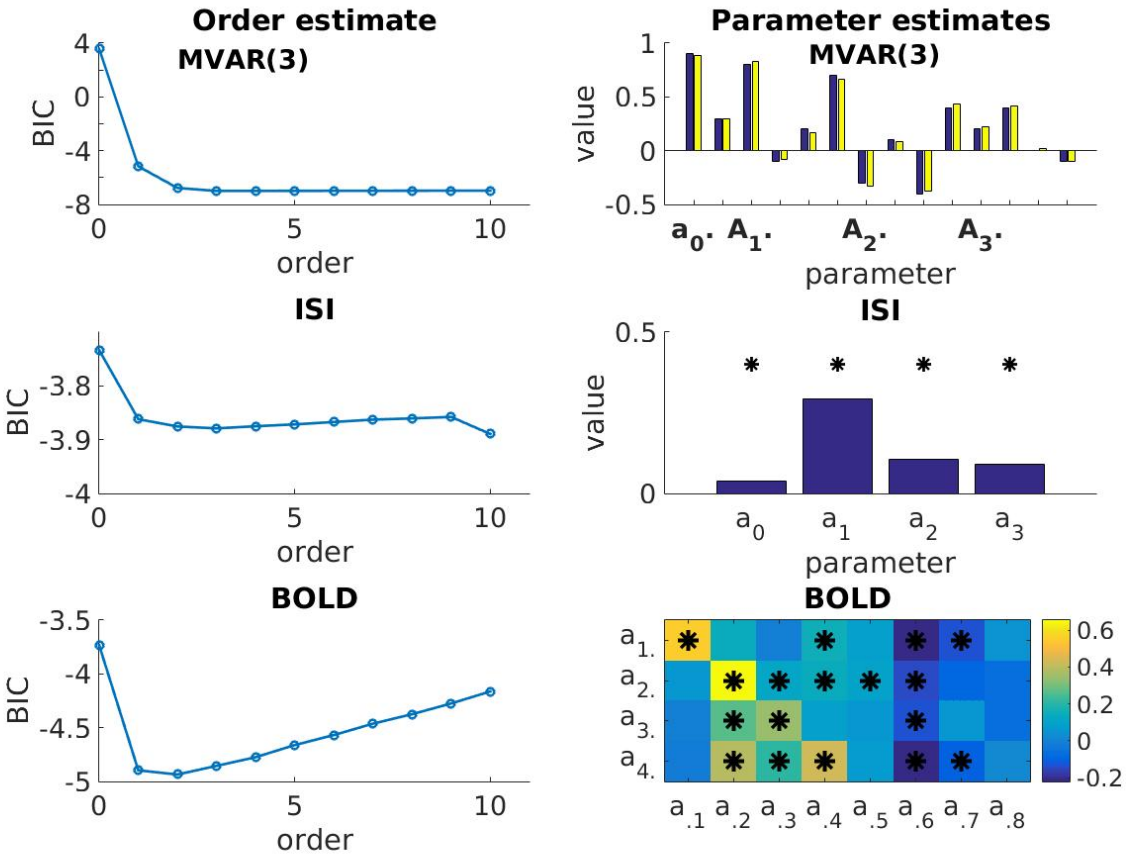

Fig. 7.6. Order (left column) and parameter (right column) estimation in a multivariate AR(3) process (top row), in ISI series (center row), and for four simultaneously recorded BOLD time series (bottom row); same data as in Fig. 7.1 (Lapish et al. 2008, Bähner et al. 2015). For the MVAR(3) process, although hard to see in the graph (see **MATL7_4** file for more details), the BIC indeed indicates a third-order process with a minimum at 3 (and higher orders do not significantly reduce the model error according to the sequential test procedure described in the text). True parameters (blue bars) and estimates (yellow bars) tightly agree. The ISI series is well described by a third-order process (left) with all estimated parameters achieving significance ($p < .05$; right). For the 4-variate BOLD series a 2nd-order MVAR process (according to the BIC) appears appropriate. The matrix on the right shows parameter estimates for the first two auto-regressive matrices (concatenated in the display; $\mathbf{a}_0$ coefficients omitted), with significance ($p<.01$) indicated by stars. Note, however, that assumptions on residuals were not checked for the purpose of these illustrative examples! **MATL7_4**.

Having determined whether our model gives rise to a stationary process (otherwise we may consider procedures for detrending or removing other types of non-stationarity first, see sect. 7.1 above), one may examine the appropriateness of the model by checking the distribution of the residuals, very much like in conventional regression models.

7.2.2 Statistical inference on model parameters

For asymptotic statistical inference, it is essential that the model assumptions introduced above are all met. Specifically, in the following we assume an V/AR(p) model of the form (7.22) or (7.26), where we now distinguish between sample estimates $\hat{\alpha}_i = a_i$ of the

coefficients and underlying population parameters α_i. Furthermore, the restriction is imposed that the white noise comes from a *Gaussian* process, i.e. $\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$, or $\boldsymbol{\varepsilon}_t \sim N(\mathbf{0}, \boldsymbol{\Sigma})$, respectively, with $E[\boldsymbol{\varepsilon}_t \boldsymbol{\varepsilon}_{t'}^T] = \mathbf{0}$ for $t \neq t'$. By separating systematic (V/AR) and pure noise part in this manner, one can apply most of the asymptotic theory developed in the context of the GLM (see sect. 2.1-2.2) to V/AR models (bootstrap-based testing for time series will be introduced below, sect. 7.7). For instance, *t*-type statistics for individual parameter estimates $\hat{\alpha}_i = a_i$ of the model, testing $H_0 : \alpha_i = 0$, can be defined – analogously to eq. 2.8 – as (Lütkepohl 2006)

(7.29) $$\frac{\hat{\alpha}_i}{\hat{\sigma}\sqrt{v_{ii}}} \sim t_{T-2p-1}$$

for the univariate case (for *K* variables the degrees of freedom become [*T*-*p*]- [*Kp*+1]), with v_{ii} being the *i*-th diagonal element of $(\mathbf{X}_p^T \mathbf{X}_p)^{-1}$ (see (7.24)). (Note that we assumed here that the length of time series available for estimation is *T*-*p*, not *T*.) Hence, this statistic has the same form and follows the same assumptions as in the standard multivariate/multiple linear regression model, and can be derived the same way (just as in eq. 2.7, the assumption $\boldsymbol{\varepsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$ leads to a normal distribution for parameter estimates $\hat{\alpha}_i$ and a corresponding χ^2-distribution for the denominator of eq. 7.29).

More generally, linear hypotheses of the form

(7.30) $$H_0 : \mathbf{LA} = \mathbf{C}$$

can be checked by likelihood ratio or Wald-type statistics (see sect. 2.2; Lütkepohl 2006), where **A** is the matrix of coefficients, **C** is a matrix of constants (usually 0's), and indicator matrix **L** picks out or combines elements from **A** in accordance with the specific hypothesis to be tested.

The likelihood function of an V/AR(*p*) process follows directly from the distributional assumptions on the residuals. First note that according to model definition (7.22), the x_t depend on the past only through the previous *p* values $\{x_{t-1} \ldots x_{t-p}\}$, and are conditionally independent from any earlier values once these are known. Hence, using Bayes' law, the total likelihood factorizes as

(7.31) $$\mathrm{L}(\{\alpha_i\}, \sigma) = f([x_{p+1}, \ldots, x_T] \,|\, \{\alpha_i\}, p, \sigma, \{x_{1:p}\}) = \prod_{t=p+1}^{T} f(x_t \,|\, x_{t-1} \ldots x_{t-p}),$$

where '*f*' is used to indicate the density here (in order to avoid confusion with parameter *p*). (Note that model parameters in eq. 7.24 were only estimated from the last *T*-*p* observations, since for the first *p* observations we don't have a complete set of *p* known predecessors, and hence the likelihood above is also formulated in terms of the last *T*-*p* outputs only. Other choices are possible, but might complicate estimation and inference since we essentially may have to add unobserved random variables to our system.) Since the residuals are independently Gaussian distributed with zero mean and variance σ^2, $\varepsilon_t \sim N(0, \sigma^2)$, it follows from model (7.22) that

(7.32) $$x_t \,|\, x_{t-1} \ldots x_{t-p} \sim N\left(\alpha_0 + \sum_{i=1}^{p} \alpha_i x_{t-i}, \sigma^2\right).$$

Putting this together we thus obtain

$$(7.33)\quad \mathrm{L}(\{\alpha_i\},\sigma) = \prod_{t=p+1}^{T} (2\pi\sigma^2)^{-1/2} e^{-\frac{1}{2}[x_t-(\alpha_0+\sum_{i=1}^{p}\alpha_i x_{t-i})]^2/\sigma^2} = (2\pi)^{-(T-p)/2} \,|\,\sigma^2\mathbf{I}\,|^{-1/2}\; e^{-(1/2)\boldsymbol{\varepsilon}^T\boldsymbol{\varepsilon}\sigma^{-2}} .$$

The last equality holds since $\varepsilon_t = x_t - (\alpha_0 + \sum_{i=1}^{p}\alpha_i x_{t-i})$ according to the model definition, and we collected all residuals into a single vector $\boldsymbol{\varepsilon} = (\varepsilon_{p+1}...\varepsilon_T)^T$ which follows a multivariate Gaussian with covariance matrix $\sigma^2\mathbf{I}$. Hence, we can equivalently express the likelihood in terms of a multivariate distribution on the residuals. The log-likelihood of model (7.22) then becomes

$$(7.34)\quad \log \mathrm{L}(\{\alpha_i\},\sigma) = -\frac{T-p}{2}\log(2\pi) - \frac{T-p}{2}\log(\sigma^2) - \frac{1}{2}\boldsymbol{\varepsilon}^T\boldsymbol{\varepsilon}\sigma^{-2} ,$$

from which we see that likelihood maximization w.r.t. parameters $\{\alpha_i\}$ essentially reduces to minimizing the residual sum of squares, as we had seen already in Example 2 of sect. 1.3.2. In case of a multivariate model with *K* variables, eq. 7.26, the covariance would become a block-diagonal matrix with *K* x *K* blocks of $\boldsymbol{\Sigma}$ and all residuals concatenated into one long vector $\boldsymbol{\varepsilon} = (\varepsilon_{1,p+1},...,\varepsilon_{K,p+1},...,\varepsilon_{1T}...\varepsilon_{KT})^T$ (Lütkepohl 2006).

We can use the likelihood function to define a log-likelihood-ratio test statistic (cf. sect. 1.5.2) for, e.g., determining the proper order of the V/AR model. First note that plugging in for σ^2 in eq. 7.34 the ML estimator $\hat{\sigma}^2 = \sum_{t=p+1}^{T}\hat{\varepsilon}_t^2/(T-p)$, the last term reduces to $-(T-p)/2$. Given, more generally, a *K*-variate VAR process, models of orders *p* vs. *p*+1 differ by a total of K^2 parameters, yielding (from eq. 7.34) the approximately *F*-distributed log-likelihood-ratio-based statistic (Lütkepohl 2006)

$$(7.35)\quad \frac{T-p-1}{K^2}[\log|\boldsymbol{\Sigma}_p| - \log|\boldsymbol{\Sigma}_{p+1}|] \sim F_{K^2,(T-p-1)-K(p+1)-1} .$$

To be precise, this is the log-likelihood ratio statistic determined *only* from the last *T*-*p*-1 time series observations available for estimation in *both* the larger (*p*+1) and the smaller (*p*) model, divided by K^2. Based on this, a series of successive hypothesis tests may be performed, starting from *p*=1, and increasing the order of the process as long as the next higher order still explains a significant amount of variance in the process (i.e., reduces the residual variance significantly according to eq. 7.34). **MATL7_4** implements this incremental test procedure for determining the order of a VAR process (Fig. 7.6).

It is to be emphasized that ML estimation and testing in AR time series models is to be treated with much more caution than in conventional regression models: We already know that we are dealing with (often highly) dependent data, and so it is crucial that all these dependencies have been covered by the systematic part of the model (through parameters **A**). One should for instance plot the residuals from the model as a function of time on which these should not depend in any way, or the auto-correlation function of the residuals (which should be about 0 everywhere except for the 0-lag). More formally, potential deviations of the residual auto-correlations from zero could be checked, for instance, by Portmanteau lack-of-fit tests which yield asymptotically χ^2 distributed statistics under the H_0 (Ljung & Box 1978; Lütkepohl 2006).

7.3 Auto-Regressive Models for Count and Point Processes

The models discussed in sect. 7.2 assumed normally distributed errors for inference. While this may be appropriate for fMRI or EEG data, it is generally not for spike data from single unit recordings or behavioral error counts, for instance (although transformations, like those from the Box-Cox class, sect. 1.5.2, may sometimes help out). This section therefore introduces general*ized* linear time series models which are more proper for describing count or point process series as they typically result from single unit recordings. The distinction between 'linear' and 'nonlinear' models admittedly becomes quite blurry here, since the binary or count-type nature of the data imposes restrictions on how to express the conditional mean of the process (e.g. McCullagh & Nelder 2009; Fahrmeir & Tutz 2010). The models discussed below are included here, in Ch. 7, mainly because the relation between previous observations and some function of the current process mean is still given by a linear equation, in contrast to models for which the transitions in time themselves clearly constitute a nonlinear process, as described in Chs. 8 & 9.

We introduce the topic by assuming that we have observed a *p*-variate time series of spike *counts* $\mathbf{c}_t = (c_{1t}...c_{pt})^T$ from, e.g., in-vivo multiple single unit recordings. Depending on our choice of bin size, the single unit counts c_{it} may either be just binary numbers (say with bin width 5-20 ms for cortical neurons), or could be larger counts as in a (peri-stimulus) time histogram. Since the probability for a spike being generated in a small temporal interval Δt will go to 0 as $\Delta t \to 0$, while at the same time the number of such elementary events within a bin will go to infinity, one can invoke Poisson distributional assumptions for the c_{it} (Koch 1999). Relating the conditional mean of the Poisson process to previous spike counts through a generalized linear equation, we obtain the Poisson AR model (cf. McCullagh & Nelder 2009; Fahrmeir & Tutz 2010)

$$\begin{aligned} & c_{it} \sim Poisson(\mu_{it}) \quad \forall i \\ & \log \boldsymbol{\mu}_t = \mathbf{a}_0 + \sum_{m=1}^{M} \mathbf{A}_m \mathbf{c}_{t-m} \end{aligned} \quad . \tag{7.36}$$

Through the nonlinear logarithmic link function it was assured that the conditional mean (spike rate) $\boldsymbol{\mu}_t$ is non-negative, and it is connected to the spike counts at previous time steps through the linear transition matrices $\mathbf{A}_m$. An offset $\mathbf{a}_0$ is included in the model to allow for a defined base rate in the absence of other inputs. One may interpret the $\mathbf{A}_m$ as time-lag-dependent *functional connectivity matrices* among the set of recorded units which we might want to estimate from the data using model (7.36). In fact, it is a better way to assess functional interactions than the much more common procedure of computing pair-wise cross-correlations, since the joint estimation of all interaction terms in $\mathbf{A}$ may account for some of the 'third-party' effects (i.e., spurious correlations induced in a given pair by common input from other units). As noted above, the Poisson output assumption (as opposed to the Gaussian error terms in standard linear regression models) is also the more appropriate one for spike count data.

The following discussion is simplified by assuming – without loss of generality – that interactions at all time lags *m* have been absorbed into one big $p\times(p\times M)$ matrix $\mathbf{A}$ (we may simply concatenate all the $\mathbf{A}_m$ and stack the $\mathbf{c}_{t-m}$ on top of each other to yield $(p\times M)\times 1$ column vector $\mathbf{c}_{t'} = (c_{1,t'}...c_{pM,t'})^T$, see sect. 7.2.1). We further accommodate offset $\mathbf{a}_0$ as usual by a leading 1 in the concatenated vector $\mathbf{c}_{t'}$. Assuming that all dependencies in time and between units have been resolved through the transition equation $\mathbf{A}\mathbf{c}_{t'}$, the observations c_{it} are all conditionally independent given $\boldsymbol{\mu}_t$, and the log-likelihood of this

model can be expressed as

$$(7.37)\quad \log p(\{\mathbf{c}_t\}\,|\,\mathbf{A}) = \log\left[\prod_{t=M+1}^{T}\prod_{i=1}^{p}\frac{\mu_{it}^{c_{it}}}{c_{it}!}e^{-\mu_{it}}\right] = \sum_{t=M+1}^{T}\sum_{i=1}^{p} c_{it}\log\mu_{it} - \mu_{it} - const.,$$

where for maximization the constant terms $\log(c_{it}!)$ drop out.

Since we are dealing with sums of exponentials on the right hand side (note the μ_{it}'s are exponentials, cf. eq. 7.36), in general this optimization problem can only be solved numerically (using, e.g., gradient descent, sect. 1.4.1). One may exploit, however, for an analytical approximation, the fact that for many choices of bin width Δt, the c_{it} will only be small integer numbers (perhaps in the range of 0…3). Without loss of generality let us focus on a single unit *i* for now. Assume that for that unit all regression weights up to $a_{i,j-1}$ have already been estimated, where j=1…pM indexes the elements of concatenation vector $\mathbf{c}_{t'}$ as defined above (i.e., runs over both variables and previous time steps). Define $z_{it} = a_{i0} + \sum_{k=1}^{j-1} a_{ik}c_{k,t'}$. Then the log-likelihood contribution of the j^{th} term for unit *i* can be expressed as

$$(7.38)\quad l_{ij} = \sum_{t=M+1}^{T} c_{it}\log\mu_{it} - \mu_{it} = \sum_{t=M+1}^{T} c_{it}[z_{it} + a_{ij}c_{j,t'}] - e^{z_{it}+a_{ij}c_{j,t'}}\,.$$

Taking the derivative w.r.t. a_{ij} one obtains

$$(7.39)\quad \frac{dl_{ij}}{da_{ij}} = \sum_{t=M+1}^{T} c_{it}c_{j,t'} - c_{j,t'}e^{z_{it}+a_{ij}c_{j,t'}} = \sum_{t=M+1}^{T} c_{it}c_{j,t'} - \sum_{t=M+1}^{T} c_{j,t'}e^{z_{it}}\left(e^{a_{ij}}\right)^{c_{j,t'}} = \sum_{t=M+1}^{T} c_{it}c_{j,t'} - \sum_{t=M+1}^{T} c_{j,t'}e^{z_{it}}\beta_{ij}^{c_{j,t'}},$$

where by the substitution $\beta_{ij} = e^{a_{ij}}$, function dl_{ij}/da_{ij} becomes an $\max(c_{j,t'})^{th}$-order polynomial in β_{ij} (note that all the c_{it}, $c_{j,t'}$ are known, as is z_{it} by assumption). Thus, if we have at most 2 spikes per bin [i.e., $\max(\{c_{it}\})\leq 2$], (7.39) could easily be solved explicitly for β_{ij}, and we obtain $a_{ij} = \log\beta_{ij}$ through back-substitution. If counts higher than 2 occur but are rare, we may still subsume them under the second-order term. Note that this solution is only approximative since we have not solved the full system of simultaneous equations in the $\{a_{ij}\}$, but instead solved (7.37) stepwise by including one regressor at a time and fixing z_{it} from the previous step. Nevertheless, for spike count data this works very well and hugely reduces the computational burden that comes with numerical optimization (Fig. 7.7; **MATL7_5**).

Another possible way to reduce the problem is to assume that coefficients a_{ij} are of some functional form, e.g. $a_{ij} = \lambda_k \exp(-m_j/\tau)$, with m_j the time step associated with entry *ij* in **A**, and τ some globally determined decay constant. That way one may reduce the full regression matrix **A** to a much smaller set of coefficients λ_k (e.g. one per variable). The assumption that regression weights decay exponentially in time is indeed a very reasonable one for a set of interacting neurons (with postsynaptic potentials usually dropping off exponentially in time). Hence, exploiting our knowledge about the specific dynamical system at hand, we may often be able to considerably simplify the estimation problem.

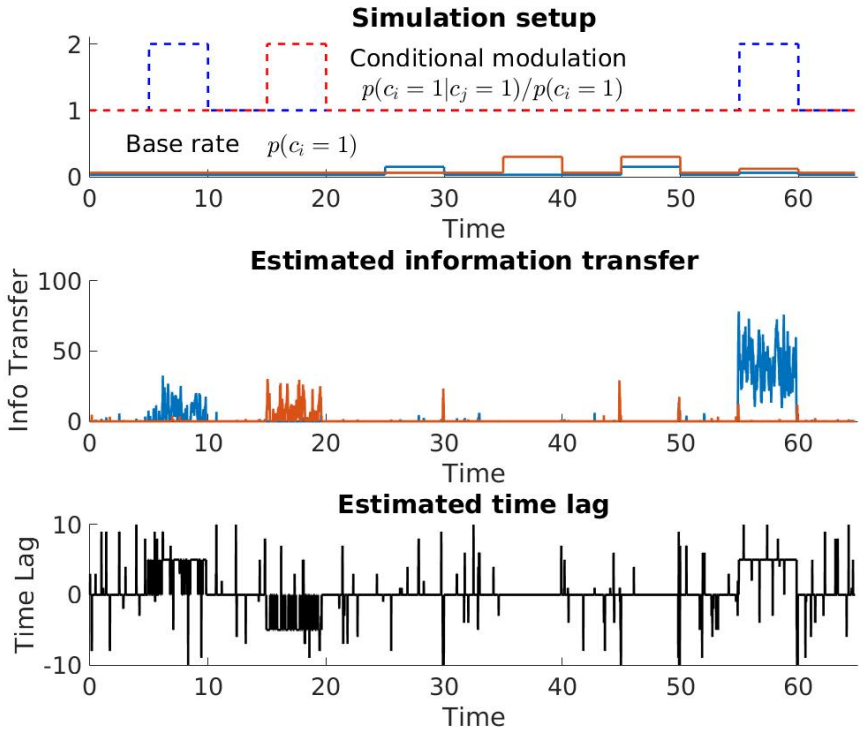

Fig. 7.7. Using Poisson AR model (7.36) with iterative maximization of (7.38) to check for interactions (information transfer) among two sparse count processes (see **MATL7_5** for implementational details and parameters). Top graph shows in blue and red solid the base rates (priors) of the two processes (with episodes of only one or both processes increasing their base rate 2- or 5-fold, respectively), and in blue and red dashed the conditional modulation at lag 5, i.e., the factor by which the spiking probability in one process is increased when having a spike in the other process 5 time steps earlier. Center graph gives the information transfer on a sliding 20-time-bin basis as measured by the increase in BIC based on log-likelihood (7.38) achieved through adding regressors from the respective other process to model (7.36) (see **MATL7_5** for details). Note that episodes of conditional modulation > 1 are correctly picked out, while episodes of mere base rate modulation are largely ignored (except for some edge effects when the sliding window extends across regions of different base rate). Bottom graph gives the estimated time lag on the same sliding window basis, correctly indicating the 5-step lag among the two processes in one or the other direction, respectively. **MATL7_5**.

Rather than discretizing (binning) the spiking process and converting it into a series of counts, one may also work directly with the series of spike time points. Pillow et al. (2011; Paninski et al. 2010) formulated such a point process model directly in terms of spike-based interactions among units. In this type of formalism, the spiking probability is commonly expressed in terms of a conditional intensity function, defined as (Kim et al. 2011)

$$(7.40)\quad \lambda_i[t \mid H_i(t)] \equiv \lambda_t^{(i)} := \lim_{\Delta t \to 0} \frac{pr[N_i(t+\Delta t) - N_i(t) = 1 \mid H_i(t)]}{\Delta t},$$

where $\lambda_i[t \mid H_i(t)]$ is the spiking intensity (or instantaneous spike rate) of unit *i* at time *t* given the full history $H_i(t)$ of all units' spike times in the observed set (or of all units known

to affect unit *i*) at time *t*. $N_i(t+\Delta t)$ is the cumulated spike count of unit *i* one time step Δt ahead, and $N_i(t)$ is the spike count at time *t*. Thus, $pr[N_i(t+\Delta t) - N_i(t) = 1 | H_i(t)]$ is the probability that unit *i* emits one spike within interval Δt as $\Delta t \rightarrow 0$, given network spiking history $H_i(t)$.

Specifically, Pillow et al. (2011) relate the spiking intensity $\lambda_t^{(i)}$ for each neuron *i* to the spiking history of the *N* recorded units through

$$(7.41)\quad \log \lambda_t^{(i)} = \mathbf{k}_i \mathbf{s}_t + \sum_{j=1}^{N} \sum_{\{t_{sp,n}^{(j)} < t\}} h_{ij}(t - t_{sp,n}^{(j)}) + b_i \ ,$$

where $\mathbf{s}_t$ is a stimulus (external) input linearly filtered by $\mathbf{k}_i$, b_i sets a baseline rate for that unit, and the sum runs over all other units *j* in the network, and across the set $\{t_{sp,n}^{(j)} < t\}$ of their spike times preceding *t*. h_{ij} corresponds to a kind of 'postsynaptic potential function' (or kernel in terms of sect. 5.1.2) that quantifies the impact of unit *j*'s n^{th} spike $t_{sp,n}^{(j)}$ on unit *i*'s instantaneous rate through time. Multiplying the instantaneous rate with sufficiently small (to allow for a maximum of one spike) bin width Δt gives the (Poisson) probability of generating a spike in that particular bin, i.e. $p(\text{"}spike\text{"} | H_i) = \Delta t \lambda_t \exp(-\Delta t \lambda_t)$. From this Poisson probability for small enough Δt, the log-likelihood for this spike-based model now moves across all actual spike times (at which we would require the estimated intensity to be maximal; Pillow et al. 2011):

$$(7.42)\quad \log p(\{t_{sp,n}^{(i)}\} | \boldsymbol{\theta}) = \sum_{i=1}^{\#units} \sum_{n=1}^{\#spikes(i)} \log[\lambda^{(i)}(t_{sp,n}^{(i)})] - \sum_{i=1}^{\#units} \int_0^T \lambda^{(i)} dt + const.,$$

where the integral on the right hand side results from taking the limit $\Delta t \rightarrow 0$ (converting the sum across time bins into an integral), and the constant log(Δt) terms could be dropped for maximization w.r.t. $\boldsymbol{\theta} = \{\{\mathbf{k}_i\}, \{h_{ij}\}, \{b_i\}\}$. It should be mentioned that Pillow et al. (2011) assumed that the driving stimulus $\mathbf{s}_t$ is generally *not* observed but has to be inferred from the spiking activity as well. In that case, (7.41) becomes a *latent factor* model (to be treated in section 7.5 below) intended by the authors as a *decoding model* for predicting the unknown stimulus from the neural spiking activity. A similar model was employed by Pillow et al. (2008) to assess the functional connectivity and contributions from neural correlations to stimulus decoding in retinal ganglion cells.

7.4 Granger causality

The concept of causality is a central one in the explanatory framework of the natural sciences, although its role in understanding highly complex, nonlinear dynamical systems like the brain, which consist of billions of interacting feedback loops, is not a trivial one. Often, however, we lack the opportunity to interact with the system of interest in a causal manner, by a well-defined experimental manipulation, especially when working with human subjects where many manipulations affecting the nervous system are obviously out of the question. Or in in-vivo electrophysiological recordings we usually have observations from multiple neurons in parallel with the behavior, but it is difficult to causally interact with them at the same time, at least at the temporal and spatial scale that may be required (although with the advance of optogenetic techniques such things now start to become feasible; Airan et al. 2009). A long-standing dream therefore has been to determine causal interactions from the mere observation of a couple of time series processes by

themselves, e.g. among sets of recorded brain areas or neurons.

Granger's idea was to formalize the concept of causality in terms of predictability: If X causes Y, then the current state of X should predict something about Y's future, but not necessarily vice versa, unless there is a mutual causal interaction. So one key point here is directionality, unlike mere correlation which is always mutual, and another is predictability across time. Granger's conception of causality is in fact quite general and based on conditional probabilities (Granger 1980): Say we are interested in whether **X** 'Granger-causes' **Y**, and $\mathbf{Z}_t$ summarizes all knowledge about the world at time *t* about all factors (including **X**) that potentially influence **Y**, then causality from **X** to **Y** is established if

$$(7.43)\quad pr(\mathbf{Y}_t \in \mathbf{U} \mid \mathbf{Z}_{t-past} \setminus \mathbf{X}_{t-past}) \neq pr(\mathbf{Y}_t \in \mathbf{U} \mid \mathbf{Z}_{t-past}),$$

where **U** is some non-empty set, '**Z**\ **X**' denotes the set **Z** of all possible predictors with all those in set **X** removed, and we used $\mathbf{X}_{t\text{-past}}$ as a shortcut for the set $\{\mathbf{X}_{t-1}\ldots\mathbf{X}_{t-\infty}\}$. In words, if the conditional probability of observing some outcome $\mathbf{Y}_t$ is different when predictors $\mathbf{X}_{t\text{-past}}$ are not taken into account, then it is reasonable to assume that **X** exerts some causal influence on **Y**. So eq. 7.43 gives the intuitive idea of causality a mathematically precise definition.

Note that definition (7.43) encompasses nonlinear situations. In practice, however, a huge amount of data may be required to evaluate (7.43) with acceptable variance, so one may have to retract to linear approximations. In fact, the Granger concept is most commonly implemented in terms of multivariate (vector) AR models (7.26). Specifically, the idea is to test whether the past of **X** significantly contributes to predicting current or future values of **Y** beyond of what could already be predicted by the other variables in $\mathbf{Z}_{t\text{-past}}$, including the past of **Y** itself. Thus, two VAR models are formulated, one including, and the other excluding, past values of $\{\mathbf{x}_t\}$ up to the specified order *p* (Granger 1969; Lütkepohl 2006):

$$(7.44)\quad \begin{aligned} &(i)\quad \mathbf{y}_t = \mathbf{a}_0 + \sum_{i=1}^{p} \mathbf{A}_i \mathbf{y}_{t-i} + \sum_{i=1}^{p} \mathbf{B}_i \mathbf{x}_{t-i} + \boldsymbol{\varepsilon}_t \\ &(ii)\quad \mathbf{y}_t = \mathbf{a}_0 + \sum_{i=1}^{p} \mathbf{A}_i \mathbf{y}_{t-i} + \boldsymbol{\varepsilon}_t\ ,\ \boldsymbol{\varepsilon}_t \sim N(0, \boldsymbol{\Sigma}) \end{aligned}$$

(For notational brevity and clarity, we ignore here other covariates $\mathbf{Z}_{t\text{-past}}$, although of course they could be easily added to the model.) Based on these, similar to the GLM framework, we can ask whether the residual amount of variance in model (*ii*) is significantly larger than in model (*i*), or – formulated differently – whether $\mathbf{x}_{t\text{-past}}$ accounts for a significant amount of variation in $\mathbf{y}_t$ beyond the variation that can already be explained by **y**'s own past, $\mathbf{y}_{t\text{-past}}$, and potentially other predictors. This may be based on common multivariate test statistics as introduced in sect. 2.2, or as derived from the likelihood-ratio principle in sect. 7.2.2 (see eq. 7.35). Following Lütkepohl (2006), here we will employ a Wald-type statistic: As described in sect. 7.3 for the Poisson AR model, let us assume we have combined all predictors of model eq. 7.44(*i*) into a single large $K_2 \times [p \times (K_1+K_2)+1]$ coefficient matrix **A**, with K_2 and K_1 the number of variables in $\mathbf{y}_t$ and $\mathbf{x}_t$, respectively, and the constant vector $\mathbf{a}_0$ accommodated as well. Let's further concatenate all columns of **A** into a single long vector $\tilde{\boldsymbol{\alpha}}$ (spelled out in detail in Lütkepohl, 2006). For testing a hypothesis of the form (7.30) one then defines

$$(7.45)\quad \lambda_W = (\mathbf{L}\tilde{\boldsymbol{\alpha}} - \mathbf{c})^T [\mathbf{L}((\mathbf{V}^T\mathbf{V})^{-1} \otimes \boldsymbol{\Sigma})\mathbf{L}^T]^{-1} (\mathbf{L}\tilde{\boldsymbol{\alpha}} - \mathbf{c}) \sim \chi^2_m\ ,\quad m = rank(\mathbf{L}),$$

where $\mathbf{V}=[\mathbf{1},\mathbf{Y}_{t-1:t-p},\mathbf{X}_{t-1:t-p}]$ is the full matrix of predictors, Σ the residual covariance matrix (from the full model (*i*) in eq. 7.44 above), $\otimes$ denotes the Kronecker product, $\mathbf{c}=\mathbf{0}$ in this context, and **L** a matrix which picks out exactly those elements from $\tilde{\boldsymbol{\alpha}}$ and $(\mathbf{V}^T\mathbf{V})^{-1}\boldsymbol{\Sigma}$ related to the hypothesis (see sect. 2.1-2.2). Hence, in this case **L** will be 0 everywhere except for those coefficients in $\tilde{\boldsymbol{\alpha}}$ that quantify the influence of $\mathbf{X}_{t-1:t-p}$ on $\mathbf{Y}_t$. Only for these the corresponding entries L_{ij} in **L** will be 1, so that according to the general form of the H_0 given in (7.30) all coefficients $\tilde{\boldsymbol{\alpha}}$ related to $\mathbf{X}_{t-1:t-p}$ are tested against 0. The degrees of freedom *m* in (7.45) are given by the rank of **L** (which in turn, in this case, is determined by the number of coefficients in $\tilde{\boldsymbol{\alpha}}$ set to 0). Alternatively to the χ^2-approximation for λ_W in (7.45), one may use the *F*-approximation $\lambda_W / m \sim F_{m,T-p-(K_1+K_2)p-1}$ for testing (Lütkepohl 2006). If the observed value for λ_W turns out significant, we may conclude that **X** 'Granger-causes' **Y**, or – more cautiously – that $\mathbf{X}_{t\text{-past}}$ makes a significant contribution to predicting $\mathbf{Y}_t$ beyond what is known from $\mathbf{Y}_{t\text{-past}}$ already. Fig. 7.8 (**MATL7_6**) illustrates these concepts at work.

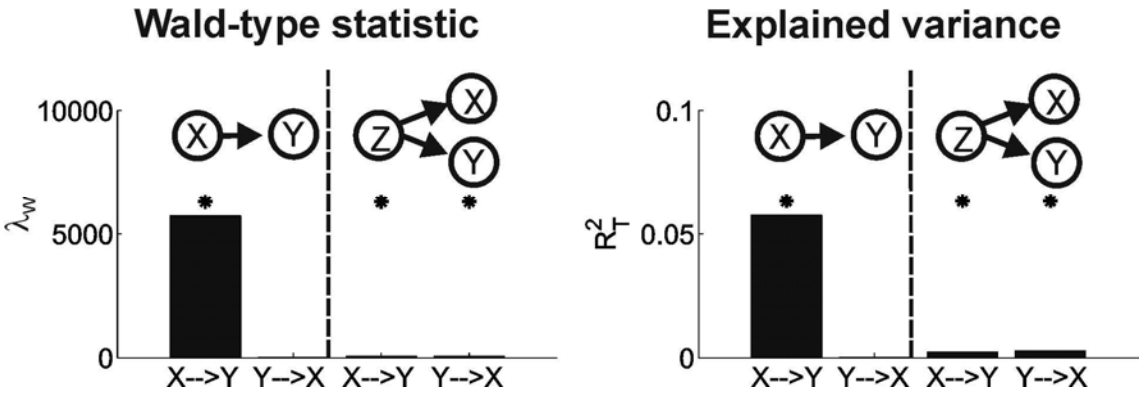

Fig. 7.8. Granger causality among two bivariate sets **X** and **Y** evaluated by the Wald-type statistic (7.45) using AR model (7.44) (left) and by the shared variance (R^2) along the maximum eigen-direction from CCA (right). The first two bars in each graph illustrate the scenario where **X** drives **Y** with no feedbacks from **Y** to **X** (as clearly indicated by the significant λ_W resp. R^2 value in one but not the other direction), while the second two bars illustrate a common driver scenario with no interactions among **X** and **Y** but both driven by a common source **Z**. Although both λ_W and R^2 are strongly reduced in this case, they still achieve significance in both directions. See **MATL7_6** for parameters and implementation.

In practice, of course, we rarely have access to the complete set $\mathbf{Z}_{t\text{-past}}$ (eq. 7.43) potentially related to **Y**, which can cause trouble. For instance, we may not be able to rule out that there is a common underlying cause or driver to **X** and **Y**, not included in **Z**, which may give rise to the spurious impression of a causal relationship between **X** and **Y** (Fig. 7.8, right). In reality this would be caused by the common variance induced by this driver not represented in the model, in particular when there are differential time lags from the not observed driver to **X** and **Y** such that **X** leads **Y** (or vice versa). Hence, care must be taken when interpreting predictability in terms of causality.

Another issue is the prediction time lag we choose, i.e. whether we attempt to predict **Y** one, two, or more steps into the future. Often one may actually only have access to **X** and **Y**, and no other external variables. In that case, considering predictions only one step ahead is sufficient, and larger forecast steps will not add any further information (see Lütkepohl, 2006, for details). If, however, say, *R* sets of potential predictors (excluding **Y** itself) are available, then also *R* forecast steps have to be considered to reveal the full 'causal' structure. Intuitively this is because **X** may cause **Y** only indirectly through other variables in **Z**, that is there might be a causal chain along the *R* sets such that the impact

of variables **X** on **Y** may surface only *R* time steps later.

We can also utilize regularization techniques (as introduced in sect. 2.4) when dealing with large variable sets and/ or comparatively short time series, by regularizing the covariance matrices involved in solving the regression model eq. 7.44 and adjusting the denominator d.f. accordingly. This is described in further detail below.

Granger causality may also quite elegantly be approached from the perspective of canonical correlation analysis (CCA, see sect. 2.3; Sato et al. 2010, Wu et al. 2011). One advantage here is that CCA comes with a genuine dimensionality reduction, if only directions in the canonical space associated with the largest eigenvalues are kept (c.f. sect. 2.3 for details). This may be very useful, for instance, in the analysis of high-dimensional fMRI time series where it is of interest to extract only the most informative directions of 'causal interaction' from the large sets of voxels within each ROI (Sato et al. 2010). However, we would like to assess gains in predictability from **X** to **Y**, beyond what is already known from $\mathbf{Y}_{t\text{-past}}$ and other predictors $\mathbf{Z}_{t\text{-past}}$, not just mere correlation. So the conventional CCA procedure has to be slightly modified along these aims. We start by regressing out **Y**'s own past and potentially other confounding predictors $\mathbf{Z}_{t\text{-past}}$ from $\mathbf{Y}_t$, as well as the *current* value of $\mathbf{X}_t$, to ensure that the result does not just reflect instantaneous (non-causal) correlations between **X** and **Y**, but really a temporally predictive relationship (Sato et al. 2010, Wu et al. 2011). Thus, we form the model

$$(7.46)\quad \hat{\mathbf{y}}_t = \mathbf{a}_0 + \sum_{i=1}^{p} \mathbf{A}_i \mathbf{y}_{t-i} + \mathbf{B}_0 \mathbf{x}_t + \sum_{i=0}^{q} \mathbf{C}_i \mathbf{z}_{t-i} \ ,$$

and continue to work on the residuals $\tilde{\mathbf{y}}_t = \mathbf{y}_t - \hat{\mathbf{y}}_t$, i.e. run CCA between the adjusted sets $\{\tilde{\mathbf{y}}_t\}$ and $\mathbf{X}_{t\text{-past}}$. Based on this, one may then proceed by computing any of the common test statistics for the CCA/ GLM framework as introduced in sect. 2.2-2.3. **MATL7_6** (Fig. 7.8) also implements the CCA-based Granger causality concept and compares it in performance to the one derived from MVAR models (as pointed out in sect. 2.3, ultimately all these approaches are closely related within the linear framework). Sato et al. (2010) and Wu et al. (2011) applied this CCA-based scheme to reveal 'causal brain connectivity maps' from fMRI and EEG data, respectively.

Regularization techniques can easily be incorporated into the CCA-based model by replacing all of the involved covariance matrices through estimators of the form $\tilde{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma} + \lambda \mathbf{I}$, with λ determined by any of those techniques discussed in Ch. 4. This will lead to modified numerator and denominator degrees of freedom in any of the *F*-type test statistics introduced in sect. 2.2, where the contribution of the number of variables *p* and *q* on each side of the CCA model to the degrees of freedom is reduced to effective values given by (2.32).

Granger causality, as defined above, however, is only directly applicable to continuously-valued Gaussian variables like EEG or fMRI measurements. In the case of spike trains, either these need to be pre-processed to give (continuous) spike density estimates (see sect. 5.1.2; but **caution**: This may introduce spurious interactions!!), or Granger causality is better directly defined in terms of interacting point processes. In the latter case, we leave the strictly linear framework and enter the world of general*ized* linear time series models as introduced in sect. 7.3 above. Such a framework was provided by Kim et al. (2011), who express Granger causality in terms of the conditional intensity $\lambda_i[t \mid H_i(t)]$ of a (spiking) point process *i*, defined in eq. 7.40 (sect. 7.3), where history $H_i(t)$ collects the previous spike times of all units which could potentially affect unit *i*. This

conditional intensity is modeled through a generalized linear model (cf. discussion of logistic regression in sect. 3.3), which takes the form

$$(7.47)\quad \log \lambda_i[t \mid H_i(t), \boldsymbol{\gamma}_i] = \gamma_{i0} + \sum_{j=1}^{J}\sum_{p=1}^{P_i} \gamma_{ijp} R_{jp}(t)$$

where γ_{i0} defines a background (spontaneous) spiking rate for unit *i*, and parameters γ_{ijp} quantify the impact of units *j* on unit *i* through their spike counts $R_{jp}(t)$ within the p^{th} time interval, up to P_i time intervals into the past (the influence of a spike will decay over time). In practice, a discrete time representation for the spike count process $R_{jp}(t)$ is used with resolution Δt fine enough to allow only for a maximum of one spike per bin (but large enough to make computations most efficient under this limitation). Given this, we have a Bernoulli probability process (with binary outcomes) and counts N_i will be binomially (or, in the limit, Poisson) distributed. Hence, the data likelihood given parameters $\boldsymbol{\gamma}$ can be written (Kim et al. 2011)

$$(7.48)\quad L_i(\boldsymbol{\gamma}_i) = \prod_{k=1}^{K} (\lambda_i[t_k \mid H_i(t_k), \boldsymbol{\gamma}_i]\Delta t)^{\Delta N_i(t_k)} (1 - \lambda_i[t_k \mid H_i(t_k), \boldsymbol{\gamma}_i]\Delta t)^{1-\Delta N_i(t_k)} ,$$

where the total time has been split into *K* intervals of width Δt, $\Delta N_i(t_k) \in \{0,1\}$ specifies whether a spike has occurred in the k^{th} interval, and hence having it in the exponent picks out the right probability (conditional intensity times bin width) to maximize for that interval (cf. sect. 3.3). One can estimate the parameters as usual by maximum likelihood, differentiating the log-likelihood for mathematical convenience, and setting to 0 (see sect. 3.3).

Similar to eq. 7.44, we can now define a reduced and a full model, with the reduced being equivalent to the full except for removal of spike train *m* from the history $H_i(t)$, for which we like to examine a causal effect on unit *i*. Thus, the reduced model for testing causality $m \rightarrow i$ takes the form (Kim et al. 2011)

$$(7.49)\quad \log \lambda_i^{-m}[t \mid H_i^{-m}(t), \boldsymbol{\gamma}_i^{-m}] = \gamma_{i0}^{-m} + \sum_{j \neq m}^{J}\sum_{p=1}^{P_i} \gamma_{ijp}^{-m} R_{jp}(t) ,$$

where superscript -*m* indicates that unit *m* has been removed from the history and for parameter estimation. To formally test whether unit *m* has a significant impact on the future fate of neuron *i* (in the sense of spike time prediction) we can employ the likelihood-ratio test statistic (sect. 1.5.2; Kim et al. 2011)

$$(7.50)\quad \theta_{im} = -2 \log \frac{L_i(\boldsymbol{\gamma}_i^{-m})}{L_i(\boldsymbol{\gamma}_i)} = -2[\log L_i(\boldsymbol{\gamma}_i^{-m}) - \log L_i(\boldsymbol{\gamma}_i)] \sim \chi^2_{P_i} ,$$

where the degrees of freedom P_i follow from the fact that we have exactly P_i less parameters in the reduced compared to the full model (cf. also sect. 7.2.2). Note that $L_i(\boldsymbol{\gamma}_i^{-m})$ can only be as large as or smaller than $L_i(\boldsymbol{\gamma}_i)$, since the reduced model has less free parameters for fitting the data, and hence we have $\theta_{im} \geq 0$.

7.5 Linear Time Series Models with Latent Variables

The time series models introduced in the preceding sections were entirely formulated in terms of those variables directly observed. However, often what we really might be interested in may be those processes which *gave rise to the observed time series* but could not be directly measured themselves. Such variables are called *latent*, *hidden*, or simply *unobserved*. Factor analysis (sect. 6.4) provides an example of a latent variable model where the observed variables are assumed to arise from a linear mixing of latent factors, like unobserved personality traits underlying performance in various questionnaires and tests (see examples in sect. 6.4), plus noise terms. The situation of unobserved processes of interest is indeed frequently encountered in neuroscience. For instance, we might only have direct experimental access to signals like the local field potential, EEG, or fMRI activity, or – ultimately – the overt behavior, but might really be interested in the underlying, unobserved spiking activity of neurons which generated these observed processes or phenomena. Or, we might only be able to measure the spiking activity of a tiny fraction of all neurons within a brain area, but might need to refer to other, unobserved network processes to account for the observed spiking dynamics.

In the time series domain, one commonly tries to capture such situations by formulating a *measurement* or *observation equation* $p(\mathbf{x}_t \mid \mathbf{z}_t) = f(\mathbf{z}_t)$ which relates the directly observed process $\mathbf{x}_t$ to the underlying latent state $\mathbf{z}_t$, and a *transition process* $p(\mathbf{z}_t \mid \mathbf{z}_{t-1}) = g(\mathbf{z}_{t-1})$ which connects the latent states in time through a (usually) first-order Markov process. This Markov assumption is indeed a crucial ingredient to all these models: The present latent state $\mathbf{z}_t$ depends only on the immediately preceding state, $\mathbf{z}_{t-1}$ (or set of preceding states in higher-order Markov models), and not on the whole history of the process [i.e., $p(z_t \mid z_{t-1}, z_{t-2}, z_{t-3}, \ldots, z_0) = p(z_t \mid z_{t-1})$]. Some authors therefore refer to this class of models commonly as Hidden Markov Models (HMMs), although we will reserve this term more specifically for a class of models with discrete, categorical states $\mathbf{z}_t$ (as discussed in sect. 8.4). Another crucial property of these models is that an observation $\mathbf{x}_t$ depends only on the underlying state $\mathbf{z}_t$ at time t, and any two consecutive observations $\mathbf{x}_t$ and $\mathbf{x}_{t'}$ are *conditionally independent* given the hidden states $\mathbf{z}_t$ and $\mathbf{z}_{t'}$, i.e. (Bishop 2006)

$$(7.51)\quad p(\mathbf{x}_t, \mathbf{x}_{t'} \mid \mathbf{z}_t, \mathbf{z}_{t'}) = p(\mathbf{x}_t \mid \mathbf{z}_t, \mathbf{z}_{t'}) p(\mathbf{x}_{t'} \mid \mathbf{z}_t, \mathbf{z}_{t'}) = p(\mathbf{x}_t \mid \mathbf{z}_t) p(\mathbf{x}_{t'} \mid \mathbf{z}_{t'}) .$$

The last equality holds since $\mathbf{x}_t$ does not depend on $\mathbf{z}_{t'}$ once $\mathbf{z}_t$ is known (same for $\mathbf{x}_{t'}$, i.e. the current state $\mathbf{z}_t$ completely specifies the conditional distribution of $\mathbf{x}_t$). Note that this does *not* imply $p(\mathbf{x}_t, \mathbf{x}_{t'}) = p(\mathbf{x}_t)p(\mathbf{x}_{t'})$. In fact, a nice thing about these models is that they allow for *temporal dependency up to any lag (or order) among observations* $\mathbf{x}_t$, while the hidden states $\mathbf{z}_t$ (usually) depend only on the directly preceding state $\mathbf{z}_{t-1}$ (Bishop 2006). Thus, there is a common structure which all of these models share, including state space models (to be discussed below) and Hidden Markov Models (HMM; to be discussed in sect. 8.4): A hidden (to the observer) underlying Markovian latent process gives rise to ('emits') the observed quantities whose conditional distribution depends only on the current latent state $\mathbf{z}_t$, and thus all observations $\mathbf{x}_t$ are conditionally independent once the $\mathbf{z}_t$ are given (Ghahramani 2001; Bishop 2006; Fahrmeir & Tutz 2010).

One general complication in this class of models is that to obtain the likelihood for the observed data $\mathbf{X}=\{\mathbf{x}_t\}$, $t=1\ldots T$, given the parameters $\boldsymbol{\theta}$, one has to integrate across ('marginalize out') the set of *all possible latent state trajectories* (*paths*) $\mathbf{Z}=\{\mathbf{z}_t\}$:

$$(7.52)\quad \begin{aligned} \log p(\mathbf{X} \mid \boldsymbol{\theta}) &= \log \int_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta}) d\mathbf{Z} = \log \int_{\mathbf{Z}} p(\mathbf{X} \mid \mathbf{Z}, \boldsymbol{\theta}) p(\mathbf{Z} \mid \boldsymbol{\theta}) d\mathbf{Z} \\ &= \log \int_{\mathbf{Z}} p(\mathbf{x}_1 \mid \mathbf{z}_1, \boldsymbol{\theta}) p(\mathbf{z}_1 \mid \boldsymbol{\theta}) \prod_{t=2}^{T} p(\mathbf{x}_t \mid \mathbf{z}_t, \boldsymbol{\theta}) p(\mathbf{z}_t \mid \mathbf{z}_{t-1}, \boldsymbol{\theta}) d\mathbf{Z} \quad , \end{aligned}$$

where the equality in the last row rests on the model's Markov and conditional independence assumptions. This usually very high-dimensional integral with the log going in front (rather than inside, where it could convert Gaussians into quadratic functions) generally prevents an analytical solution. The next section will therefore discuss the most common numerical scheme for solving these equations, based on the expectation-maximization (EM) algorithm (sect. 1.4.2).

In this chapter we will only deal with latent time series models which are *linear* in their transition equations (albeit not necessarily in their outputs), like the ARMA models discussed in the previous sections. Models with *nonlinear dynamics* will be deferred to Ch. 8 & 9. This includes HMMs as defined here, since these assume a set of discrete states $\mathbf{z}_t$ among which the system jumps and in this sense behaves nonlinearly.

7.5.1 Linear State Space Models

State space models like ARMA models are discrete-time linear dynamical systems (cf. sect. 9.1), only that they include latent variables $\mathbf{z}$ which are not directly observed. In fact, state space models contain the class of ARMA models as special cases, and each ARMA model can equivalently be expressed as a state space model (Lütkepohl 2006). They extend ARMA models through the idea that the observed time series were generated by an underlying hidden process (evolving in an unobserved state space), which is then related to the observed time series by another linear process. In their simplest form, they may be written as (Rauch et al. 1965; Bishop 2006; Fahrmeir & Tutz 2010; Durbin & Koopman 2012)

$$
\begin{aligned}
&\mathbf{x}_t = \mathbf{B}\mathbf{z}_t + \boldsymbol{\eta}_t \,,\; \boldsymbol{\eta}_t \sim N(\mathbf{0}, \boldsymbol{\Gamma}) && (\textit{observation or measurement equation}) \\
&\mathbf{z}_t = \mathbf{A}\mathbf{z}_{t-1} + \boldsymbol{\varepsilon}_t \,,\; \boldsymbol{\varepsilon}_t \sim N(\mathbf{0}, \boldsymbol{\Sigma}) && (\textit{transition equation}) \\
&\mathbf{z}_1 \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}) && (\textit{initial condition}) \,,
\end{aligned} \tag{7.53}
$$

where the first (observation) equation gives the observed (p x 1) quantities $\{\mathbf{x}_t\}$ as a linear function of the current (q x 1) state $\mathbf{z}_t$ and measurement noise $\boldsymbol{\eta}_t$, while the second (transition) equation basically takes the form of an V/AR(1) model, with the usual independence assumptions for the two noise processes $\boldsymbol{\varepsilon}_t$ and $\boldsymbol{\eta}_t$. Note that the observed time series $\{\mathbf{x}_t\}$ depends on past states only through the latent variables $\{\mathbf{z}_t\}$, where we may have dim($\mathbf{z}$) < dim($\mathbf{x}$), i.e. the model may imply a dimensionality reduction with the observed multivariate time series generated by potentially much fewer latent variables (see sect. 7.5.2).

The 'measurement noise' $\boldsymbol{\eta}_t$ in (7.53) captures the usual statistical uncertainty in the observations, which represent just a small sample from a much larger population. But why would we include yet another noise process with the latent states, especially since it makes inference so much harder? One reason is that transition processes in real-world systems are often intrinsically stochastic, like activity in the nervous system (Jahr & Stevens 1990; Koch 1999; see also Ch. 9). Hence, if we had no probability assumptions for the transition process itself, we would misattribute noisy fluctuations in the transitions to the deterministic part of the dynamics. Another reason is that our transition model will most likely only represent some approximation to the true dynamics, and $\boldsymbol{\varepsilon}_t$ could account for some of this uncertainty and misspecification in the underlying model.

The basic linear model (7.53) could be extended in various ways, for instance, by including an external ('exogenous') input $\mathbf{w}_t$ both into the measurement and the transition equations. Let us also point out that assuming that the initial state $\mathbf{z}_1$ is distributed with the same covariance $\boldsymbol{\Sigma}$ as $\boldsymbol{\varepsilon}_t$ is a simplification we made here to reduce the number of

parameters and slightly ease the following presentation. In general, the co-variation among the $\mathbf{z}_t$ induced by the transition process adds to the noise covariance (as shown below), and this should also be true for $\mathbf{z}_1$ although its history is not known. It may therefore be more reasonable to afford $\mathbf{z}_1$ its own covariance matrix, different from $\boldsymbol{\Sigma}$.

As noted above, direct maximum likelihood estimation of the model from empirical data is hampered by the fact that one has to integrate out the hidden state path, eq. 7.52. The most common remedy is the EM algorithm (see sect. 1.4.2) which will be derived here mainly in close relation to the superb presentation in Bishop (2006; but see also Rauch et al. 1965; Lütkepohl 2006; Durbin & Koopman 2012). The EM algorithm separates and iterates the latent state path and the parameter estimation steps (McLachlan & Krishnan 1997): Assuming parameters $\boldsymbol{\theta}=\{\mathbf{A},\mathbf{B},\boldsymbol{\Sigma},\boldsymbol{\Gamma},\boldsymbol{\mu}_0\}$ of the model to be known, one seeks the posterior distribution of the unknown latent variables $\{\mathbf{z}_t\}$ from the observed series $\{\mathbf{x}_t\}$. Vice versa, if we had estimates of the latent states $\mathbf{Z}=\{\mathbf{z}_t\}$ in addition to the observed series $\mathbf{X}=\{\mathbf{x}_t\}$, $t=1\ldots T$, model parameters $\{\mathbf{A},\mathbf{B},\boldsymbol{\Sigma},\boldsymbol{\Gamma},\boldsymbol{\mu}_0\}$ could be inferred. Specifically, we start with an initial guess of parameters and determine the posterior distribution $p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta})$ of the latent states $\mathbf{Z}=\{\mathbf{z}_t\}$ for computing the *expected* joint ('complete data') log-likelihood $\mathrm{E}_{\mathbf{Z}}[\log L_{\mathbf{X},\mathbf{Z}}(\boldsymbol{\theta})]$ across hidden states $\mathbf{Z}$ (E-step), which provides a lower bound for $\log L_{\mathbf{X}}(\boldsymbol{\theta})$ (e.g. Roweis & Ghahramani 2001), and then in the M-step optimize model parameters $\boldsymbol{\theta}$ with regards to $\mathrm{E}_{\mathbf{Z}}[\log L_{\mathbf{X},\mathbf{Z}}(\boldsymbol{\theta})]$ fixing $p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta})$ from the E-step (see 1.4.2). Plugging in the multivariate normal assumptions and Markovian probability structure from model (7.53), this expectancy can be spelled out as

(7.54)

$$
\begin{aligned}
E_{\mathbf{Z}}\{\log p(\mathbf{X},\mathbf{Z}\,|\,\boldsymbol{\theta})\} &= E_{\mathbf{Z}}\{\log[p(\mathbf{X}\,|\,\mathbf{Z},\boldsymbol{\theta})p(\mathbf{Z}\,|\,\boldsymbol{\theta})]\} \\
&= E_{\mathbf{Z}}\left\{\log\left[\, p(\mathbf{z}_1\,|\,\boldsymbol{\theta})p(\mathbf{x}_1\,|\,\mathbf{z}_1,\boldsymbol{\theta})\prod_{t=2}^{T} p(\mathbf{z}_t\,|\,\mathbf{z}_{t-1},\boldsymbol{\theta})p(\mathbf{x}_t\,|\,\mathbf{z}_t,\boldsymbol{\theta})\right]\right\} \\
&= E_{\mathbf{Z}}\left\{-\frac{1}{2}\left[T\log|\boldsymbol{\Sigma}| + (\mathbf{z}_1-\boldsymbol{\mu}_0)^T\boldsymbol{\Sigma}^{-1}(\mathbf{z}_1-\boldsymbol{\mu}_0) + \sum_{t=2}^{T}(\mathbf{z}_t-\mathbf{A}\mathbf{z}_{t-1})^T\boldsymbol{\Sigma}^{-1}(\mathbf{z}_t-\mathbf{A}\mathbf{z}_{t-1})\right.\right. \\
&\qquad \left.\left. + T\log|\boldsymbol{\Gamma}| + \sum_{t=1}^{T}(\mathbf{x}_t-\mathbf{B}\mathbf{z}_t)^T\boldsymbol{\Gamma}^{-1}(\mathbf{x}_t-\mathbf{B}\mathbf{z}_t)\right] + const.\right\}
\end{aligned}
$$

To provide a little bit of intuition, note that the first terms (second to last line) in this expected likelihood kind of 'measure' the consistency of states $\mathbf{z}_t$ in time as commanded by the transition equation in (7.53), while the last term assesses the consistency of outputs predicted from the current states $\mathbf{z}_t$ with the actually observed outputs $\mathbf{x}_t$ (weighted with the respective uncertainty along each dimension through the covariance matrix). Obviously both these consistencies should be high for system (7.53) to provide a good model for the data.

Now, a key aspect to note is that for maximization w.r.t. parameters $\mathbf{A}$, $\mathbf{B}$, $\boldsymbol{\Gamma}$, $\boldsymbol{\Sigma}$, and $\boldsymbol{\mu}_0$, "only" expectations of the form $E[\mathbf{z}_t]$, $E[\mathbf{z}_t\mathbf{z}_t^T]$ and $E[\mathbf{z}_t\mathbf{z}_{t-1}^T]$ are required. To see this, we exploit the linearity of expectancy values and rewrite (7.54):

(7.55)

$$E_{\mathbf{Z}}\{\log p(\mathbf{X},\mathbf{Z}\,|\,\boldsymbol{\theta})\} = -\frac{1}{2}\Big\{T\log|\boldsymbol{\Sigma}| + T\log|\boldsymbol{\Gamma}| + E[\mathbf{z}_1^T\boldsymbol{\Sigma}^{-1}\mathbf{z}_1] - \boldsymbol{\mu}_0^T\boldsymbol{\Sigma}^{-1}E[\mathbf{z}_1] - E[\mathbf{z}_1^T]\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0 + \boldsymbol{\mu}_0^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0$$
$$+\sum_{t=2}^{T}\left(E[\mathbf{z}_t^T\boldsymbol{\Sigma}^{-1}\mathbf{z}_t] - E[\mathbf{z}_t^T\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{z}_{t-1}] - E[\mathbf{z}_{t-1}^T\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{z}_t] + E[\mathbf{z}_{t-1}^T\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{z}_{t-1}]\right),$$
$$+\sum_{t=1}^{T}\left(\mathbf{x}_t^T\boldsymbol{\Gamma}^{-1}\mathbf{x}_t - \mathbf{x}_t^T\boldsymbol{\Gamma}^{-1}\mathbf{B}E[\mathbf{z}_t] - E[\mathbf{z}_t^T]\mathbf{B}^T\boldsymbol{\Gamma}^{-1}\mathbf{x}_t + E[\mathbf{z}_t^T\mathbf{B}^T\boldsymbol{\Gamma}^{-1}\mathbf{B}\mathbf{z}_t]\right) + const.\Big\}$$

which can be further shaped into the desired form by using the relationship $\mathbf{x}^T\mathbf{A}\mathbf{y} = tr[\mathbf{A}\mathbf{y}\mathbf{x}^T]$, yielding $E[\mathbf{z}_t^T\boldsymbol{\Sigma}^{-1}\mathbf{z}_t] = tr\left[\boldsymbol{\Sigma}^{-1}E[\mathbf{z}_t\mathbf{z}_t^T]\right]$, $E[\mathbf{z}_t^T\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{z}_{t-1}] = tr\left[\boldsymbol{\Sigma}^{-1}\mathbf{A}E[\mathbf{z}_{t-1}\mathbf{z}_t^T]\right]$, and so forth (for maximization, this is not really necessary [one may take the derivatives first], but it highlights the structure of the complete data log-likelihood).

For the ***E-step***, an efficient way to compute the terms $E[\mathbf{z}_t]$, $E[\mathbf{z}_t\mathbf{z}_t^T]$ and $E[\mathbf{z}_t\mathbf{z}_{t-1}^T]$ are the Kalman "filter-smoother" recursions (Kalman 1960; Rauch et al. 1965). They start from the following temporal dissection of the posterior p(**Z**|**X**,θ) (see Bishop 2006):

$$p(\mathbf{z}_t\,|\,\{\mathbf{x}_t\},\boldsymbol{\theta}) = \frac{p(\mathbf{z}_t,\mathbf{x}_1,...,\mathbf{x}_t,\mathbf{x}_{t+1},...,\mathbf{x}_T\,|\,\boldsymbol{\theta})}{p(\mathbf{x}_1,...,\mathbf{x}_T\,|\,\boldsymbol{\theta})} = \frac{p(\mathbf{z}_t,\mathbf{x}_1,...,\mathbf{x}_t\,|\,\boldsymbol{\theta})\,p(\mathbf{x}_{t+1},...,\mathbf{x}_T\,|\,\mathbf{z}_t,\mathbf{x}_1,...,\mathbf{x}_t,\boldsymbol{\theta})}{p(\mathbf{x}_1,...,\mathbf{x}_T\,|\,\boldsymbol{\theta})}$$
$$= \frac{p(\mathbf{z}_t,\mathbf{x}_1,...,\mathbf{x}_t\,|\,\boldsymbol{\theta})}{p(\mathbf{x}_1,...,\mathbf{x}_t\,|\,\boldsymbol{\theta})} \times \frac{p(\mathbf{x}_{t+1},...,\mathbf{x}_T\,|\,\mathbf{z}_t,\boldsymbol{\theta})}{p(\mathbf{x}_{t+1},...,\mathbf{x}_T\,|\,\mathbf{x}_1,...,\mathbf{x}_t,\boldsymbol{\theta})} \qquad (7.56)$$

where we have used Bayes' rule and the fact that all $\mathbf{x}_t$, $t > \tau$, are conditionally independent from all $\mathbf{x}_t$, $t \le \tau$, given $\mathbf{z}_\tau$. In a *forward pass*, called the *Kalman filter*, the first product term in the rightmost expression in (7.56) is recursively determined from

$$\frac{p(\mathbf{z}_t,\mathbf{x}_1,...,\mathbf{x}_t\,|\,\boldsymbol{\theta})}{p(\mathbf{x}_1,...,\mathbf{x}_t\,|\,\boldsymbol{\theta})} = p_{\boldsymbol{\theta}}(\mathbf{z}_t\,|\,\mathbf{x}_1,...,\mathbf{x}_t) = \frac{p_{\boldsymbol{\theta}}(\mathbf{z}_t,\mathbf{x}_t,\mathbf{x}_1,...,\mathbf{x}_{t-1})}{p_{\boldsymbol{\theta}}(\mathbf{x}_t\,|\,\mathbf{x}_1,...,\mathbf{x}_{t-1})\,p_{\boldsymbol{\theta}}(\mathbf{x}_1,...,\mathbf{x}_{t-1})}$$
$$= \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_t\,|\,\mathbf{z}_t,\mathbf{x}_1,...,\mathbf{x}_{t-1})\,p_{\boldsymbol{\theta}}(\mathbf{z}_t\,|\,\mathbf{x}_1,...,\mathbf{x}_{t-1})\,p_{\boldsymbol{\theta}}(\mathbf{x}_1,...,\mathbf{x}_{t-1})}{p_{\boldsymbol{\theta}}(\mathbf{x}_t\,|\,\mathbf{x}_1,...,\mathbf{x}_{t-1})\,p_{\boldsymbol{\theta}}(\mathbf{x}_1,...,\mathbf{x}_{t-1})}$$
$$= \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_t\,|\,\mathbf{z}_t)\,p_{\boldsymbol{\theta}}(\mathbf{z}_t\,|\,\mathbf{x}_1,...,\mathbf{x}_{t-1})}{p_{\boldsymbol{\theta}}(\mathbf{x}_t\,|\,\mathbf{x}_1,...,\mathbf{x}_{t-1})} \qquad (7.57)$$
$$= \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_t\,|\,\mathbf{z}_t)\int_{\mathbf{z}_{t-1}} p_{\boldsymbol{\theta}}(\mathbf{z}_t\,|\,\mathbf{z}_{t-1})\,p_{\boldsymbol{\theta}}(\mathbf{z}_{t-1}\,|\,\mathbf{x}_1,...,\mathbf{x}_{t-1})\,d\mathbf{z}_{t-1}}{p_{\boldsymbol{\theta}}(\mathbf{x}_t\,|\,\mathbf{x}_1,...,\mathbf{x}_{t-1})},$$

where again Bayes' rule and the conditional independence property were employed at various stages. Thus, once we have $p_{\boldsymbol{\theta}}(\mathbf{z}_{t-1}\,|\,\mathbf{x}_1,...,\mathbf{x}_{t-1})$, $p_{\boldsymbol{\theta}}(\mathbf{z}_t\,|\,\mathbf{x}_1,...,\mathbf{x}_t)$ can be recursively derived (as indicated by the yellow highlighting) until we hit the end of the chain t=T.

It is to be emphasized that this temporal dissection, eq. 7.56, and recursion relationships, eq. 7.57, is general in the sense that it relies only on the Markov and conditional independence properties, but not for instance on the linearity of model (7.53). This will become important later on in section 9.3 where the same temporal decomposition and recursions are used in the context of nonlinear models.

There is one issue, however: We have to perform an integration across previous states $\mathbf{z}_{t-1}$. This involves in principle straightforward, but somewhat tedious matrix manipulations. Nevertheless, we will carry out the key steps here as they will give some

insight into how to solve such problems more generally. First, note that by model assumptions (7.53), $p_{\boldsymbol{\theta}}(\mathbf{z}_t \mid \mathbf{z}_{t-1}) = N(\mathbf{A}\mathbf{z}_{t-1}, \boldsymbol{\Sigma})$. Since the probability distributions involved in the observation and transition equations are both linear Gaussian, $p_{\boldsymbol{\theta}}(\mathbf{z}_{t-1} \mid \mathbf{x}_1,...,\mathbf{x}_{t-1})$ will also be Gaussian, say with mean $\boldsymbol{\mu}_{t-1}$ and covariance matrix $\mathbf{V}_{t-1}$. Hence, for the integral we have

(7.58)

$$\int_{\mathbf{z}_{t-1}} p_{\boldsymbol{\theta}}(\mathbf{z}_t \mid \mathbf{z}_{t-1}) p_{\boldsymbol{\theta}}(\mathbf{z}_{t-1} \mid \mathbf{x}_1,..., \mathbf{x}_{t-1}) d\mathbf{z}_{t-1} =$$

$$\int_{\mathbf{z}_{t-1}} (2\pi)^{-q/2} \mid \boldsymbol{\Sigma} \mid^{-1/2} e^{-\frac{1}{2}(\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})^T \boldsymbol{\Sigma}^{-1} (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})} (2\pi)^{-q/2} \mid \mathbf{V}_{t-1} \mid^{-1/2} e^{-\frac{1}{2}(\mathbf{z}_{t-1} - \boldsymbol{\mu}_{t-1})^T \mathbf{V}_{t-1}^{-1} (\mathbf{z}_{t-1} - \boldsymbol{\mu}_{t-1})} d\mathbf{z}_{t-1}$$

We will focus now on further manipulating the exponent after multiplying the two exponentials, which is

(7.59)

$$\begin{aligned}
&-\frac{1}{2}\left[(\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})^T \boldsymbol{\Sigma}^{-1} (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1}) + (\mathbf{z}_{t-1} - \boldsymbol{\mu}_{t-1})^T \mathbf{V}_{t-1}^{-1} (\mathbf{z}_{t-1} - \boldsymbol{\mu}_{t-1})\right] \\
&= -\frac{1}{2}\left[\mathbf{z}_{t-1}^T (\mathbf{A}^T \boldsymbol{\Sigma}^{-1} \mathbf{A} + \mathbf{V}_{t-1}^{-1}) \mathbf{z}_{t-1} - (\mathbf{z}_t^T \boldsymbol{\Sigma}^{-1} \mathbf{A} + \boldsymbol{\mu}_{t-1}^T \mathbf{V}_{t-1}^{-1}) \mathbf{z}_{t-1} - \mathbf{z}_{t-1}^T (\mathbf{A}^T \boldsymbol{\Sigma}^{-1} \mathbf{z}_t + \mathbf{V}_{t-1}^{-1} \boldsymbol{\mu}_{t-1}) + \mathbf{z}_t^T \boldsymbol{\Sigma}^{-1} \mathbf{z}_t + \boldsymbol{\mu}_{t-1}^T \mathbf{V}_{t-1}^{-1} \boldsymbol{\mu}_{t-1}\right] \\
&= -\frac{1}{2}\left[\mathbf{z}_{t-1}^T \mathbf{H}^{-1} \mathbf{z}_{t-1} - \mathbf{m}^T \mathbf{z}_{t-1} - \mathbf{z}_{t-1}^T \mathbf{m} + \mathbf{m}^T \mathbf{H}^T \mathbf{H}^{-1} \mathbf{H} \mathbf{m} - \mathbf{m}^T \mathbf{H}^T \mathbf{H}^{-1} \mathbf{H} \mathbf{m} + \mathbf{z}_t^T \boldsymbol{\Sigma}^{-1} \mathbf{z}_t + \boldsymbol{\mu}_{t-1}^T \mathbf{V}_{t-1}^{-1} \boldsymbol{\mu}_{t-1}\right] \quad ,
\end{aligned}$$

where we have defined $\mathbf{H}^{-1} = \mathbf{A}^T \boldsymbol{\Sigma}^{-1} \mathbf{A} + \mathbf{V}_{t-1}^{-1}$ and $\mathbf{m} = \mathbf{A}^T \boldsymbol{\Sigma}^{-1} \mathbf{z}_t + \mathbf{V}_{t-1}^{-1} \boldsymbol{\mu}_{t-1}$. The goal here is to integrate out $\mathbf{z}_{t-1}$, and for that purpose we have added and subtracted off again stuff in the last row, so that after reinserting everything into eq. 7.58 we arrive at

$$\begin{aligned}
&\int_{\mathbf{z}_{t-1}} (2\pi)^{-q/2} \mid \boldsymbol{\Sigma} \mid^{-1/2} e^{-\frac{1}{2}(\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})^T \boldsymbol{\Sigma}^{-1} (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})} (2\pi)^{-q/2} \mid \mathbf{V}_{t-1} \mid^{-1/2} e^{-\frac{1}{2}(\mathbf{z}_{t-1} - \boldsymbol{\mu}_{t-1})^T \mathbf{V}_{t-1}^{-1} (\mathbf{z}_{t-1} - \boldsymbol{\mu}_{t-1})} d\mathbf{z}_{t-1} \\
&= (2\pi)^{-q/2} \mid \boldsymbol{\Sigma}\mathbf{V}_{t-1} \mid^{-1/2} \mid \mathbf{H} \mid^{1/2} e^{-\frac{1}{2}\left[\mathbf{z}_t^T \boldsymbol{\Sigma}^{-1} \mathbf{z}_t + \boldsymbol{\mu}_{t-1}^T \mathbf{V}_{t-1}^{-1} \boldsymbol{\mu}_{t-1} - \mathbf{m}^T \mathbf{H} \mathbf{m}\right]} \int_{\mathbf{z}_{t-1}} (2\pi)^{-q/2} \mid \mathbf{H} \mid^{-1/2} e^{-\frac{1}{2}(\mathbf{z}_{t-1} - \mathbf{H}\mathbf{m})^T \mathbf{H}^{-1} (\mathbf{z}_{t-1} - \mathbf{H}\mathbf{m})} d\mathbf{z}_{t-1} \\
&= (2\pi)^{-q/2} \mid \boldsymbol{\Sigma}\mathbf{V}_{t-1} (\mathbf{A}^T \boldsymbol{\Sigma}^{-1} \mathbf{A} + \mathbf{V}_{t-1}^{-1}) \mid^{-1/2} e^{-\frac{1}{2}\left[\mathbf{z}_t^T \boldsymbol{\Sigma}^{-1} \mathbf{z}_t + \boldsymbol{\mu}_{t-1}^T \mathbf{V}_{t-1}^{-1} \boldsymbol{\mu}_{t-1} - \mathbf{m}^T \mathbf{H} \mathbf{m}\right]}
\end{aligned} \tag{7.60}$$

Thus, we got rid of the integral across $\mathbf{z}_{t-1}$ which evaluates to 1. (Note that $\mathbf{H}$ is a covariance matrix, hence $\mathbf{H}=\mathbf{H}^T$.) It remains to clean up the mess in the left-over expression and show it is a proper Gaussian indeed. We do so by focusing on the exponent again and using a matrix identity known as Woodbury's identity:

$$
\begin{aligned}
&-\frac{1}{2}\left[\mathbf{z}_t^T\boldsymbol{\Sigma}^{-1}\mathbf{z}_t+\boldsymbol{\mu}_{t-1}^T\mathbf{V}_{t-1}^{-1}\boldsymbol{\mu}_{t-1}-\mathbf{m}^T\mathbf{H}\mathbf{m}\right]\\
&=-\frac{1}{2}\left[\mathbf{z}_t^T\boldsymbol{\Sigma}^{-1}\mathbf{z}_t+\boldsymbol{\mu}_{t-1}^T\mathbf{V}_{t-1}^{-1}\boldsymbol{\mu}_{t-1}-(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{z}_t+\mathbf{V}_{t-1}^{-1}\boldsymbol{\mu}_{t-1})^T(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}+\mathbf{V}_{t-1}^{-1})^{-1}(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{z}_t+\mathbf{V}_{t-1}^{-1}\boldsymbol{\mu}_{t-1})\right]\\
&=-\frac{1}{2}\left[\mathbf{z}_t^T\left\{\boldsymbol{\Sigma}^{-1}-\boldsymbol{\Sigma}^{-1}\mathbf{A}(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}+\mathbf{V}_{t-1}^{-1})^{-1}\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\right\}\mathbf{z}_t+\boldsymbol{\mu}_{t-1}^T\left\{\mathbf{V}_{t-1}^{-1}-\mathbf{V}_{t-1}^{-1}(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}+\mathbf{V}_{t-1}^{-1})^{-1}\mathbf{V}_{t-1}^{-1}\right\}\boldsymbol{\mu}_{t-1}\right.\\
&\qquad\left.-\boldsymbol{\mu}_{t-1}^T\left\{\mathbf{V}_{t-1}^{-1}(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}+\mathbf{V}_{t-1}^{-1})^{-1}\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\right\}\mathbf{z}_t-\mathbf{z}_t^T\left\{\boldsymbol{\Sigma}^{-1}\mathbf{A}(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}+\mathbf{V}_{t-1}^{-1})^{-1}\mathbf{V}_{t-1}^{-1}\right\}\boldsymbol{\mu}_{t-1}\right]\\
&=-\frac{1}{2}\left[\mathbf{z}_t^T(\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma})^{-1}\mathbf{z}_t+\boldsymbol{\mu}_{t-1}^T\mathbf{A}^T(\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma})^{-1}\mathbf{A}\boldsymbol{\mu}_{t-1}\right.\\
&\qquad\left.-\boldsymbol{\mu}_{t-1}^T\mathbf{A}^T(\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma})^{-1}\mathbf{z}_t-\mathbf{z}_t^T(\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma})^{-1}\mathbf{A}\boldsymbol{\mu}_{t-1}\right]\\
&=-\frac{1}{2}\left[(\mathbf{z}_t-\mathbf{A}\boldsymbol{\mu}_{t-1})^T\mathbf{L}_{t-1}^{-1}(\mathbf{z}_t-\mathbf{A}\boldsymbol{\mu}_{t-1})\right]\qquad\text{with }\mathbf{L}_{t-1}=\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma}
\end{aligned}
\tag{7.61}
$$

Woodbury's identity used in here took the two forms (cf. Petersen & Pedersen, 2012)

$$
\begin{aligned}
&\boldsymbol{\Sigma}^{-1}-\boldsymbol{\Sigma}^{-1}\mathbf{A}(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}+\mathbf{V}_{t-1}^{-1})^{-1}\mathbf{A}^T\boldsymbol{\Sigma}^{-1}=(\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma})^{-1}\\
&\mathbf{V}_{t-1}^{-1}-\mathbf{V}_{t-1}^{-1}(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}+\mathbf{V}_{t-1}^{-1})^{-1}\mathbf{V}_{t-1}^{-1}=\mathbf{V}_{t-1}^{-1}-\mathbf{V}_{t-1}^{-1}\left[\mathbf{V}_{t-1}-\mathbf{V}_{t-1}\mathbf{A}^T(\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma})^{-1}\mathbf{A}\mathbf{V}_{t-1}\right]\mathbf{V}_{t-1}^{-1}\\
&\qquad\qquad=\mathbf{A}^T(\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma})^{-1}\mathbf{A}
\end{aligned}
\tag{7.62}
$$

Putting everything together we arrive at a normal distribution with mean $\mathbf{A}\boldsymbol{\mu}_{t-1}$ and covariance matrix $\mathbf{L}_{t-1}=\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma}$ for the integral in (7.58-7.60). We are not quite done yet: The resulting expression $N(\mathbf{A}\boldsymbol{\mu}_{t-1},\mathbf{L}_{t-1})$ for the integral still has to be combined with the emission probability $p_{\boldsymbol{\theta}}(\mathbf{x}_t\,|\,\mathbf{z}_t)=N(\mathbf{B}\mathbf{z}_t,\boldsymbol{\Gamma})$ from the numerator, and the term in the denominator. The way one goes about this is by trying to shape the numerator into a single Gaussian while using the denominator to ensure proper normalization. Hence, this is again an exercise in combining several Gaussians into a single one, involving tedious but not really complicated matrix manipulations along similar lines as sketched in (7.58)-(7.62), making frequent use of the Woodbury identity.

Note that in this Gaussian context, recursive update equations for $p_{\boldsymbol{\theta}}(\mathbf{z}_t\,|\,\mathbf{x}_1,...,\mathbf{x}_t)=N(\boldsymbol{\mu}_t,\mathbf{V}_t)$ boil down to update equations for the mean $\boldsymbol{\mu}_t$ and covariance matrix $\mathbf{V}_t$. Thus, after carrying out the matrix manipulations outlined above, one finally arrives at the following updating equations for $\boldsymbol{\mu}_t$ and $\mathbf{V}_t$ (Lütkepohl 2006; Bishop 2006; Fahrmeir & Tutz 2010; Durbin & Koopman 2012):

$$
\begin{aligned}
&\boldsymbol{\mu}_t=\mathbf{A}\boldsymbol{\mu}_{t-1}+\mathbf{K}_t(\mathbf{x}_t-\mathbf{B}\mathbf{A}\boldsymbol{\mu}_{t-1})\\
&\mathbf{V}_t=(\mathbf{I}-\mathbf{K}_t\mathbf{B})\mathbf{L}_{t-1}=\left[(\mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T+\boldsymbol{\Sigma})^{-1}+\mathbf{B}^T\boldsymbol{\Gamma}^{-1}\mathbf{B}\right]^{-1},\\
&\mathbf{K}_t=\mathbf{L}_{t-1}\mathbf{B}^T(\mathbf{B}\mathbf{L}_{t-1}\mathbf{B}^T+\boldsymbol{\Gamma})^{-1}
\end{aligned}
\tag{7.63}
$$

where $\mathbf{K}_t$ is called the Kalman gain matrix. The intuitive interpretation of these equations is the following: The updated mean $\boldsymbol{\mu}_t$ at time *t* is obtained from the previous one by iterating it with transition matrix **A** one step forward in time (see transition part in eq. 7.53). This value is then corrected by a term proportional to the difference between the actually observed $\mathbf{x}_t$ at time *t*, and the value *predicted* from the forwarded mean $\mathbf{A}\boldsymbol{\mu}_{t-1}$ through observation matrix **B** (see observation part in eq. 7.53). Likewise, one obtains the updated covariance matrix $\mathbf{V}_t$ by forwarding $\mathbf{V}_{t-1}$ through transition matrix **A** in time, adding the

variation from the noise input to the latent state, and further adjusting by the imprecision in the observation process. Running these updates from t =1 to t =T completes the forward ('filtering') pass.

In the *backward*, or '*Kalman smoother*', recursions, the updates from the forward pass are now combined with the second multiplicative term in eq. 7.56 (rightmost expression) to give the full posterior for the latent states $\{\mathbf{z}_t\}$ using the entire observation history $\{\mathbf{x}_t\}$, t=1..T. To ease notation in the derivations, we define (Bishop 2006)

$$\begin{aligned}
&\alpha_t := p_{\boldsymbol{\theta}}(\mathbf{z}_t \mid \mathbf{x}_1 ... \mathbf{x}_t) = N(\boldsymbol{\mu}_t, \mathbf{V}_t) && (\textit{density from forward pass at time } t) \\
&\beta_t := \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1},...,\mathbf{x}_T \mid \mathbf{z}_t)}{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1},...,\mathbf{x}_T \mid \mathbf{x}_1,...,\mathbf{x}_t)} && (\textit{factor in backward pass at time } t) \\
&\gamma_t := p_{\boldsymbol{\theta}}(\mathbf{z}_t \mid \mathbf{x}_1 ... \mathbf{x}_T) = \alpha_t \beta_t \equiv N(\tilde{\boldsymbol{\mu}}_t, \tilde{\mathbf{V}}_t) && (\textit{full state posterior at time } t)
\end{aligned} \tag{7.64}$$

With these definitions, one can write the full state posterior at time t as

$$\begin{aligned}
\gamma_t = \alpha_t \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1},...,\mathbf{x}_T \mid \mathbf{z}_t)}{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1},...,\mathbf{x}_T \mid \mathbf{x}_1,...,\mathbf{x}_t)} &= \alpha_t \frac{\int_{\mathbf{z}_{t+1}} p_{\boldsymbol{\theta}}(\mathbf{x}_{t+2},...,\mathbf{x}_T \mid \mathbf{z}_{t+1}) p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{z}_{t+1}) p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t) d\mathbf{z}_{t+1}}{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+2},...,\mathbf{x}_T \mid \mathbf{x}_1,...,\mathbf{x}_{t+1}) p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{x}_1,...,\mathbf{x}_t)} \\
&= \alpha_t \frac{\int_{\mathbf{z}_{t+1}} \beta_{t+1} p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{z}_{t+1}) p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t) d\mathbf{z}_{t+1}}{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{x}_1,...,\mathbf{x}_t)} \\
&= \alpha_t \frac{\int_{\mathbf{z}_{t+1}} \alpha_{t+1}^{-1} \gamma_{t+1} p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{z}_{t+1}) p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t) d\mathbf{z}_{t+1}}{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{x}_1,...,\mathbf{x}_t)}
\end{aligned} \tag{7.65}$$

Now note that all the densities involved in the bottom row expression have already been computed at the time γ_t is to be evaluated: γ_{t+1} has been determined in the previous backward pass step. The denominator in eq. 7.65 (bottom) is just the normalization constant from eq. 7.57 (r.h.s.), so we know this term as well, from the forward pass. Likewise, α_t and α_{t+1} are the known filtering pass densities, and the remaining terms in the numerator of (7.65) are just the model's observation and transition densities, respectively. However, for a full derivation it is convenient to rewrite this a bit further:

$$\begin{aligned}
\gamma_t &= \alpha_t \frac{\int_{\mathbf{z}_{t+1}} \alpha_{t+1}^{-1} \gamma_{t+1} p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{z}_{t+1}) p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t) d\mathbf{z}_{t+1}}{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{x}_1,...,\mathbf{x}_t)} \\
&= \alpha_t \int_{\mathbf{z}_{t+1}} \frac{\gamma_{t+1} p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{z}_{t+1}) p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t)}{p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{x}_1,...,\mathbf{x}_t, \mathbf{x}_{t+1}) p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{x}_1,...,\mathbf{x}_t)} d\mathbf{z}_{t+1} \\
&= \alpha_t \int_{\mathbf{z}_{t+1}} \frac{\gamma_{t+1} p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{z}_{t+1}) p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t)}{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1} \mid \mathbf{z}_{t+1}, \mathbf{x}_1,...,\mathbf{x}_t) p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{x}_1,...,\mathbf{x}_t)} d\mathbf{z}_{t+1} = \alpha_t \int_{\mathbf{z}_{t+1}} \frac{\gamma_{t+1} p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t)}{p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{x}_1,...,\mathbf{x}_t)} d\mathbf{z}_{t+1}
\end{aligned} \tag{7.66}$$

with $p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{x}_1,...,\mathbf{x}_t) = N(\mathbf{A}\boldsymbol{\mu}_t, \mathbf{L}_t)$ (the 'one-step forward density'), and using the conditional independence property. Thus, we have defined backward recursions in terms of the full state posterior γ_t. Using the state estimates from the forward run which for t = T coincide with those from the backward loop (as there is 'no future' to T), i.e. initializing with $\gamma_T = p_{\boldsymbol{\theta}}(\mathbf{z}_T \mid \mathbf{x}_1 ... \mathbf{x}_T) = \alpha_T$, we work our way backward along the chain until we arrive at the

root *t*=1.

Although what remains comes down to combining Gaussians again, involving similar steps as in (7.58)-(7.62) above, for clarity we will spell this out here once more, making more explicit now of how to combine the numerator and denominator Gaussians. Writing out the integral in (7.66), we have

(7.67)

$$\int\limits_{\mathbf{z}_{t+1}} \frac{\gamma_{t+1} p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t)}{p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{x}_1,\ldots,\mathbf{x}_t)} d\mathbf{z}_{t+1} = \int\limits_{\mathbf{z}_{t+1}} (2\pi)^{-q/2} \mid \tilde{\mathbf{V}}_{t+1} \mid^{-1/2} e^{-\frac{1}{2}(\mathbf{z}_{t+1}-\tilde{\boldsymbol{\mu}}_{t+1})^T \tilde{\mathbf{V}}_{t+1}^{-1}(\mathbf{z}_{t+1}-\tilde{\boldsymbol{\mu}}_{t+1})} (2\pi)^{-q/2} \mid \boldsymbol{\Sigma} \mid^{-1/2} e^{-\frac{1}{2}(\mathbf{z}_{t+1}-\mathbf{A}\mathbf{z}_t)^T \boldsymbol{\Sigma}^{-1}(\mathbf{z}_{t+1}-\mathbf{A}\mathbf{z}_t)}$$

$$\times \left[(2\pi)^{-q/2} \mid \mathbf{L}_t \mid^{-1/2} e^{-\frac{1}{2}(\mathbf{z}_{t+1}-\mathbf{A}\boldsymbol{\mu}_t)^T \mathbf{L}_t^{-1}(\mathbf{z}_{t+1}-\mathbf{A}\boldsymbol{\mu}_t)} \right]^{-1} d\mathbf{z}_{t+1}$$

$$= \int\limits_{\mathbf{z}_{t+1}} (2\pi)^{-q/2} \mid \tilde{\mathbf{V}}_{t+1}\boldsymbol{\Sigma}\mathbf{L}_t^{-1} \mid^{-1/2} e^{-\frac{1}{2}(\mathbf{z}_{t+1}-\tilde{\boldsymbol{\mu}}_{t+1})^T \tilde{\mathbf{V}}_{t+1}^{-1}(\mathbf{z}_{t+1}-\tilde{\boldsymbol{\mu}}_{t+1}) - \frac{1}{2}(\mathbf{z}_{t+1}-\mathbf{A}\mathbf{z}_t)^T \boldsymbol{\Sigma}^{-1}(\mathbf{z}_{t+1}-\mathbf{A}\mathbf{z}_t) + \frac{1}{2}(\mathbf{z}_{t+1}-\mathbf{A}\boldsymbol{\mu}_t)^T \mathbf{L}_t^{-1}(\mathbf{z}_{t+1}-\mathbf{A}\boldsymbol{\mu}_t)} d\mathbf{z}_{t+1}$$

For the exponent we get (reusing symbols **m** and **H**):

(7.68)

$$-\frac{1}{2}\Big[\mathbf{z}_{t+1}^T(\tilde{\mathbf{V}}_{t+1}^{-1}+\boldsymbol{\Sigma}^{-1}-\mathbf{L}_t^{-1})\mathbf{z}_{t+1} - \mathbf{z}_{t+1}^T(\tilde{\mathbf{V}}_{t+1}^{-1}\tilde{\boldsymbol{\mu}}_{t+1}+\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{z}_t-\mathbf{L}_t^{-1}\mathbf{A}\boldsymbol{\mu}_t) - (\tilde{\boldsymbol{\mu}}_{t+1}^T\tilde{\mathbf{V}}_{t+1}^{-1}+\mathbf{z}_t^T\mathbf{A}^T\boldsymbol{\Sigma}^{-1}-\boldsymbol{\mu}_t^T\mathbf{A}^T\mathbf{L}_t^{-1})\mathbf{z}_{t+1}$$
$$+\tilde{\boldsymbol{\mu}}_{t+1}^T\tilde{\mathbf{V}}_{t+1}^{-1}\tilde{\boldsymbol{\mu}}_{t+1}+\mathbf{z}_t^T\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{z}_t-\boldsymbol{\mu}_t^T\mathbf{A}^T\mathbf{L}_t^{-1}\mathbf{A}\boldsymbol{\mu}_t\Big]$$
$$=-\frac{1}{2}\Big[\mathbf{z}_{t+1}^T\mathbf{H}^{-1}\mathbf{z}_{t+1}-\mathbf{z}_{t+1}^T\mathbf{m}-\mathbf{m}^T\mathbf{z}_{t+1}+\mathbf{m}^T\mathbf{H}^T\mathbf{H}^{-1}\mathbf{H}\mathbf{m}-\mathbf{m}^T\mathbf{H}^T\mathbf{H}^{-1}\mathbf{H}\mathbf{m}+\tilde{\boldsymbol{\mu}}_{t+1}^T\tilde{\mathbf{V}}_{t+1}^{-1}\tilde{\boldsymbol{\mu}}_{t+1}+\mathbf{z}_t^T\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{z}_t-\boldsymbol{\mu}_t^T\mathbf{A}^T\mathbf{L}_t^{-1}\mathbf{A}\boldsymbol{\mu}_t\Big]$$

As before (see eq. 7.59-7.60), the first part of this expression, involving $\mathbf{z}_{t+1}$, integrates to 1. Combining the remainder with the forward density from (7.66), this leaves us with

(7.69) $$\gamma_t = (2\pi)^{-q/2} \mid \mathbf{V}_t\tilde{\mathbf{V}}_{t+1}\boldsymbol{\Sigma}\mathbf{L}_t^{-1}\mathbf{H}^{-1} \mid^{-1/2} e^{-\frac{1}{2}\left[(\mathbf{z}_t-\boldsymbol{\mu}_t)^T\mathbf{V}_t^{-1}(\mathbf{z}_t-\boldsymbol{\mu}_t)-\mathbf{m}^T\mathbf{H}^T\mathbf{m}+\tilde{\boldsymbol{\mu}}_{t+1}^T\tilde{\mathbf{V}}_{t+1}^{-1}\tilde{\boldsymbol{\mu}}_{t+1}+\mathbf{z}_t^T\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{z}_t-\boldsymbol{\mu}_t^T\mathbf{A}^T\mathbf{L}_t^{-1}\mathbf{A}\boldsymbol{\mu}_t\right]}.$$

Grouping all terms linear and quadratic in $\mathbf{z}_t$ in the exponent, we can infer the mean and covariance matrix of the Gaussian:

$$\begin{aligned}
\tilde{\mathbf{V}}_t &= (\mathbf{V}_t^{-1}+\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}-\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{H}^T\boldsymbol{\Sigma}^{-1}\mathbf{A})^{-1} = \left[\mathbf{V}_t^{-1}+\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}-\mathbf{A}^T\boldsymbol{\Sigma}^{-1}(\tilde{\mathbf{V}}_{t+1}^{-1}+\boldsymbol{\Sigma}^{-1}-\mathbf{L}_t^{-1})^{-1}\boldsymbol{\Sigma}^{-1}\mathbf{A}\right]^{-1} \\
&= \left[\mathbf{V}_t^{-1}+\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}-\mathbf{A}^T\left(\boldsymbol{\Sigma}(\tilde{\mathbf{V}}_{t+1}^{-1}-\mathbf{L}_t^{-1})\boldsymbol{\Sigma}^T+\boldsymbol{\Sigma}\right)^{-1}\mathbf{A}\right]^{-1} \\
&= \left[\mathbf{V}_t^{-1}+\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}-\mathbf{A}^T\left(\boldsymbol{\Sigma}^{-1}-\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}\left\{\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}+(\tilde{\mathbf{V}}_{t+1}^{-1}-\mathbf{L}_t^{-1})^{-1}\right\}^{-1}\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}^{-1}\right)\mathbf{A}\right]^{-1} \\
&= \left[\mathbf{V}_t^{-1}+\mathbf{A}^T\left\{\boldsymbol{\Sigma}+(\tilde{\mathbf{V}}_{t+1}^{-1}-\mathbf{L}_t^{-1})^{-1}\right\}^{-1}\mathbf{A}\right]^{-1} \\
&= \mathbf{V}_t-\mathbf{V}_t\mathbf{A}^T\left[\mathbf{A}\mathbf{V}_t\mathbf{A}^T+\left\{\boldsymbol{\Sigma}+(\tilde{\mathbf{V}}_{t+1}^{-1}-\mathbf{L}_t^{-1})^{-1}\right\}\right]^{-1}\mathbf{A}\mathbf{V}_t^T \qquad (7.70) \\
&= \mathbf{V}_t-\mathbf{V}_t\mathbf{A}^T\left[\mathbf{L}_t+(\tilde{\mathbf{V}}_{t+1}^{-1}-\mathbf{L}_t^{-1})^{-1}\right]^{-1}\mathbf{A}\mathbf{V}_t^T \\
&= \mathbf{V}_t-\mathbf{V}_t\mathbf{A}^T\left[\mathbf{L}_t-\tilde{\mathbf{V}}_{t+1}(\tilde{\mathbf{V}}_{t+1}-\mathbf{L}_t)^{-1}\mathbf{L}_t\right]^{-1}\mathbf{A}\mathbf{V}_t^T \\
&= \mathbf{V}_t-\mathbf{V}_t\mathbf{A}^T\mathbf{L}_t^{-1}\left[\mathbf{I}-\tilde{\mathbf{V}}_{t+1}(\tilde{\mathbf{V}}_{t+1}-\mathbf{L}_t)^{-1}\right]^{-1}\mathbf{A}\mathbf{V}_t^T \\
&= \mathbf{V}_t+\mathbf{V}_t\mathbf{A}^T\mathbf{L}_t^{-1}(\tilde{\mathbf{V}}_{t+1}-\mathbf{L}_t)\mathbf{L}_t^{-1}\mathbf{A}\mathbf{V}_t^T = Var_{\boldsymbol{\theta}}[\mathbf{z}_t \mid \{\mathbf{x}_{1:T}\}]
\end{aligned}$$

$$\tilde{\mathbf{V}}_t^{-1}\tilde{\boldsymbol{\mu}}_t = \mathbf{V}_t^{-1}\boldsymbol{\mu}_t + \mathbf{A}^T\boldsymbol{\Sigma}^{-1}(\tilde{\mathbf{V}}_{t+1}^{-1} + \boldsymbol{\Sigma}^{-1} - \mathbf{L}_t^{-1})^{-1}(\tilde{\mathbf{V}}_{t+1}^{-1}\tilde{\boldsymbol{\mu}}_{t+1} - \mathbf{L}_t^{-1}\mathbf{A}\boldsymbol{\mu}_t)$$
$$\Rightarrow \tilde{\boldsymbol{\mu}}_t = (\mathbf{V}_t + \mathbf{V}_t\mathbf{A}^T\mathbf{L}_t^{-1}(\tilde{\mathbf{V}}_{t+1} - \mathbf{L}_t)\mathbf{L}_t^{-1}\mathbf{A}\mathbf{V}_t)\left[\mathbf{V}_t^{-1}\boldsymbol{\mu}_t + \mathbf{A}^T\boldsymbol{\Sigma}^{-1}(\tilde{\mathbf{V}}_{t+1}^{-1} + \boldsymbol{\Sigma}^{-1} - \mathbf{L}_t^{-1})^{-1}(\tilde{\mathbf{V}}_{t+1}^{-1}\tilde{\boldsymbol{\mu}}_{t+1} - \mathbf{L}_t^{-1}\mathbf{A}\boldsymbol{\mu}_t)\right]$$
$$= \boldsymbol{\mu}_t + \mathbf{V}_t\mathbf{A}^T\mathbf{L}_t^{-1}(\tilde{\boldsymbol{\mu}}_{t+1} - \mathbf{A}\boldsymbol{\mu}_t) = E_{\boldsymbol{\theta}}[\mathbf{z}_t \,|\, \{\mathbf{x}_{1:T}\}]$$

Recall that covariance matrices are symmetric – the transpose in the derivations above was sometimes included only for clarity. From the state covariance matrix $\tilde{\mathbf{V}}_t = Var_{\boldsymbol{\theta}}[\mathbf{z}_t \,|\, \{\mathbf{x}_{1:T}\}]$, finally, we obtain $E[\mathbf{z}_t\mathbf{z}_t^T]$ by adding $E[\mathbf{z}_t]E[\mathbf{z}_t]^T$.

Looking back at eq. 7.55, note that we require $E[\mathbf{z}_t\mathbf{z}_{t-1}^T]$ as well for derivation of the full expected log-likelihood. Luckily, these expectancies can be obtained from terms we have already computed. The joint conditional probability of $\mathbf{z}_t$ and $\mathbf{z}_{t-1}$ is given by (Bishop 2006)

$$\begin{aligned} p_{\boldsymbol{\theta}}(\mathbf{z}_t, \mathbf{z}_{t-1} \,|\, \{\mathbf{x}_{1:T}\}) &= \frac{p_{\boldsymbol{\theta}}(\{\mathbf{x}_{1:T}\} \,|\, \mathbf{z}_t, \mathbf{z}_{t-1}) p_{\boldsymbol{\theta}}(\mathbf{z}_t \,|\, \mathbf{z}_{t-1}) p_{\boldsymbol{\theta}}(\mathbf{z}_{t-1})}{p_{\boldsymbol{\theta}}(\{\mathbf{x}_{1:T}\})} \\ &= \frac{p_{\boldsymbol{\theta}}(\{\mathbf{x}_{1:t-1}\} \,|\, \mathbf{z}_{t-1}) p_{\boldsymbol{\theta}}(\mathbf{z}_{t-1}) p_{\boldsymbol{\theta}}(\{\mathbf{x}_{t:T}\} \,|\, \mathbf{z}_t) p_{\boldsymbol{\theta}}(\mathbf{z}_t \,|\, \mathbf{z}_{t-1})}{p_{\boldsymbol{\theta}}(\mathbf{x}_1,\ldots,\mathbf{x}_{t-1}) p_{\boldsymbol{\theta}}(\mathbf{x}_t \,|\, \mathbf{x}_1,\ldots,\mathbf{x}_{t-1}) p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1},\ldots,\mathbf{x}_T \,|\, \mathbf{x}_1,\ldots,\mathbf{x}_t)} \\ &= \frac{p_{\boldsymbol{\theta}}(\{\mathbf{x}_{1:t-1}\}, \mathbf{z}_{t-1})}{p_{\boldsymbol{\theta}}(\mathbf{x}_1,\ldots,\mathbf{x}_{t-1})} \times \frac{p_{\boldsymbol{\theta}}(\mathbf{z}_t \,|\, \mathbf{z}_{t-1}) p_{\boldsymbol{\theta}}(\mathbf{x}_t \,|\, \mathbf{z}_t)}{p_{\boldsymbol{\theta}}(\mathbf{x}_t \,|\, \mathbf{x}_1,\ldots,\mathbf{x}_{t-1})} \times \frac{p_{\boldsymbol{\theta}}(\{\mathbf{x}_{t+1:T}\} \,|\, \mathbf{z}_t)}{p_{\boldsymbol{\theta}}(\mathbf{x}_{t+1},\ldots,\mathbf{x}_T \,|\, \mathbf{x}_1,\ldots,\mathbf{x}_t)} \\ &= \alpha_{t-1} \times \frac{p_{\boldsymbol{\theta}}(\mathbf{z}_t \,|\, \mathbf{z}_{t-1}) p_{\boldsymbol{\theta}}(\mathbf{x}_t \,|\, \mathbf{z}_t)}{p_{\boldsymbol{\theta}}(\mathbf{x}_t \,|\, \mathbf{x}_1,\ldots,\mathbf{x}_{t-1})} \times (\alpha_t^{-1}\gamma_t) \end{aligned} \tag{7.71}$$

where various conditional independencies implied by model (7.53) were exploited. The first multiplicative term in the last row we have computed in the forward pass (7.57)-(7.63), the last one in the backward pass (7.65)-(7.70). For the term in the middle, the numerator is given by the model's transition and observation equations, while the denominator was obtained as the normalizing constant in the forward pass. Knitting together all the corresponding Gaussians will give us the covariance matrix as

$$Cov_{\boldsymbol{\theta}}[\mathbf{z}_t, \mathbf{z}_{t-1} \,|\, \{\mathbf{x}_{1:T}\}] = \tilde{\mathbf{V}}_t\mathbf{L}_{t-1}^{-1}\mathbf{A}\mathbf{V}_{t-1}, \tag{7.72}$$

hence $E[\mathbf{z}_t\mathbf{z}_{t-1}^T]$ by adding $E[\mathbf{z}_t]E[\mathbf{z}_{t-1}^T]$, and we are done with the E-step. The linear Kalman filter-smoother recursions are implemented in **MATL7_7**. As a note of caution, in practice, estimation through the Kalman recursions may suffer from instability issues with numerical errors piling up, leading for instance to covariance matrices which are no longer positive-semidefinite (Lütkepohl 2006). Model parameters are generally also not uniquely identifiable unless further restrictions are imposed (e.g. Roweis & Ghahramani 2001; Mader et al. 2014; Auger-Méthé et al. 2016).

A different way to approach the problem of state estimation, given fixed and known parameters $\boldsymbol{\theta}=\{\mathbf{A},\mathbf{B},\boldsymbol{\Sigma},\boldsymbol{\Gamma},\boldsymbol{\mu}_0\}$, is *direct maximization* of the log-posterior $\log p(\mathbf{Z} \,|\, \mathbf{X},\boldsymbol{\theta})$ w.r.t. $\mathbf{Z}$ (Fahrmeir & Tutz 2010; Paninski et al. 2010). Note that given the linear-Gaussian structure of model (7.53), both $p(\mathbf{Z},\mathbf{X} \,|\, \boldsymbol{\theta})$ and $p(\mathbf{Z} \,|\, \mathbf{X},\boldsymbol{\theta})$ are multivariate Gaussian. For maximizing the likelihood (7.54), we require $E[\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}]$, but for a Gaussian mean and mode are identical, so that in principle the problem boils down to maximizing a log-Gaussian, an undertaking which is only bedeviled by the necessity to invert very high-dimensional matrices. Following the presentation in Paninski et al. (2010), one can write

(7.73)

$$\begin{aligned}
\mathrm{E}[\mathbf{Z}\,|\,\mathbf{X},\boldsymbol{\theta}] &= \arg\max_{\mathbf{Z}} p(\mathbf{Z}\,|\,\mathbf{X},\boldsymbol{\theta}) = \arg\max_{\mathbf{Z}}\big[\log p(\mathbf{Z},\mathbf{X}\,|\,\boldsymbol{\theta}) - \log p(\mathbf{X}\,|\,\boldsymbol{\theta})\big] = \arg\max_{\mathbf{Z}}\big[\log p(\mathbf{Z},\mathbf{X}\,|\,\boldsymbol{\theta})\big] \\
&= \arg\max_{\mathbf{Z}}\left(\log p(\mathbf{z}_1\,|\,\boldsymbol{\theta}) + \sum_{t=2}^{T} \log p(\mathbf{z}_t\,|\,\mathbf{z}_{t-1},\boldsymbol{\theta}) + \sum_{t=1}^{T} \log p(\mathbf{x}_t\,|\,\mathbf{z}_t,\boldsymbol{\theta}) \right) \\
&= \arg\max_{\mathbf{Z}}\Bigg(-\frac{1}{2}(\mathbf{z}_1-\boldsymbol{\mu}_0)^T \boldsymbol{\Sigma}^{-1}(\mathbf{z}_1-\boldsymbol{\mu}_0) \\
&\qquad -\frac{1}{2}\sum_{t=2}^{T}(\mathbf{z}_t-\mathbf{A}\mathbf{z}_{t-1})^T \boldsymbol{\Sigma}^{-1}(\mathbf{z}_t-\mathbf{A}\mathbf{z}_{t-1}) - \frac{1}{2}\sum_{t=1}^{T}(\mathbf{x}_t-\mathbf{B}\mathbf{z}_t)^T \boldsymbol{\Gamma}^{-1}(\mathbf{x}_t-\mathbf{B}\mathbf{z}_t) \Bigg)
\end{aligned}$$

The equalities in the first row hold since p(**X**|θ) is a constant in the maximization w.r.t. **Z** and thus will not change the result, while the strictly monotonic log-transform will not do so either. All other equalities follow from the (Markov) probability structure of model (7.53), as already used in the derivation of the Kalman filter-smoother recursions above, and from the fact that for optimization w.r.t. **Z** we can drop all constant terms and the determinants of the covariance matrices which do not contain the latent states. This becomes, in essence, then a simple weighted LSE-type problem which can be solved by a simple matrix inversion.

More specifically, concatenating all state variables $\mathbf{z}_t$ into one long column vector $\mathbf{z}=(\mathbf{z}_1,\ldots,\mathbf{z}_T)$, collecting all parameters related to the linear and quadratic forms of **z** into a vector **d** and matrix **H**, respectively, and setting the derivatives with respect to the elements of **z** to 0, one obtains in general form

(7.74)
$$\frac{\partial}{\partial \mathbf{z}}\left[-\frac{1}{2}\mathbf{z}^T\mathbf{H}\mathbf{z} + \frac{1}{2}(\mathbf{d}^T\mathbf{z} + \mathbf{z}^T\mathbf{d}) \right] = -\frac{1}{2}(\mathbf{H}+\mathbf{H}^T)\mathbf{z} + \mathbf{d} = \mathbf{0} \;\Rightarrow\; \hat{\mathbf{z}} = \mathbf{H}^{-1}\mathbf{d}$$
$$\text{with } \mathbf{d} = [(\mathbf{B}^T\boldsymbol{\Gamma}^{-1}\mathbf{x}_1 + \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_0)\ldots\mathbf{B}^T\boldsymbol{\Gamma}^{-1}\mathbf{x}_t\ldots\mathbf{B}^T\boldsymbol{\Gamma}^{-1}\mathbf{x}_T]^T$$

Matrix **H** has a block-band-diagonal structure with elements

$$\mathbf{H} = \begin{pmatrix} \mathbf{S} & \mathbf{K} & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{K}^T & \mathbf{S} & \mathbf{K} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{K}^T & \mathbf{S} & \cdots & \cdots \\ \mathbf{0} & \mathbf{0} & \vdots & \ddots & \mathbf{K} \\ \vdots & \vdots & \vdots & \mathbf{K}^T & \mathbf{S}-\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A} \end{pmatrix}$$

(7.75) *with*
$$\mathbf{S} = \boldsymbol{\Sigma}^{-1} + \mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A} + \mathbf{B}^T\boldsymbol{\Gamma}^{-1}\mathbf{B}$$
$$\mathbf{K} = -\mathbf{A}^T\boldsymbol{\Sigma}^{-1}$$

Thus, in principle, one could solve the problem of obtaining E[**Z**|**X**,θ] given **X** and θ in one go (Paninski et al. 2010). The Kalman-recursions are basically an efficient way (linear in *T*) to solve for the states without having to deal with the inversion of potentially very high-dimensional matrices (Fahrmeir & Tutz 2010). However, as pointed out by Paninski et al. (2010), efficient algorithms for solving the linear equations 7.74, exploiting the band-diagonal structure of **H**, exist as well. Finally, note that **H** is the inverse covariance matrix of the multivariate Gaussian $p(\mathbf{Z}\,|\,\mathbf{X},\boldsymbol{\theta})$. Hence, expectancies $E[\mathbf{z}_t\mathbf{z}_t^T]$ and $E[\mathbf{z}_t\mathbf{z}_{t-1}^T]$ required for the M-step can be retrieved from the on- and off-diagonal blocks of $\mathbf{H}^{-1}$ by

adding terms $E[\mathbf{z}_t]E[\mathbf{z}_t]^T$ and $E[\mathbf{z}_t]E[\mathbf{z}_{t-1}]^T$, respectively.

In the ***M-step***, once we have derived $E[\mathbf{z}_t]$, $E[\mathbf{z}_t\mathbf{z}_t^T]$ and $E[\mathbf{z}_t\mathbf{z}_{t-1}^T]$ one way or the other, we can compute parameter estimates by maximizing the expected log-likelihood (7.55) w.r.t. $\boldsymbol{\theta}=\{\mathbf{A},\mathbf{B},\boldsymbol{\Sigma},\boldsymbol{\Gamma},\boldsymbol{\mu}_0\}$, which comes down to a set of straightforward LSE problems. For instance, maximizing w.r.t. **A**, all terms not containing **A** drop out from the derivative of (7.55) and one gets

(7.76)

$$\begin{aligned}
\frac{\partial E_{\mathbf{Z}}\{\log p(\mathbf{X},\mathbf{Z}\,|\,\boldsymbol{\theta})\}}{\partial \mathbf{A}} &= \frac{1}{2}\sum_{t=2}^{T}\left(\frac{\partial tr\left[\boldsymbol{\Sigma}^{-1}\mathbf{A}E[\mathbf{z}_{t-1}\mathbf{z}_t^T]\right]}{\partial \mathbf{A}}+\frac{\partial tr\left[\mathbf{A}^T\boldsymbol{\Sigma}^{-1}E[\mathbf{z}_t\mathbf{z}_{t-1}^T]\right]}{\partial \mathbf{A}}\right)-\frac{1}{2}\sum_{t=2}^{T}\frac{\partial tr\left(\mathbf{A}^T\boldsymbol{\Sigma}^{-1}\mathbf{A}E[\mathbf{z}_{t-1}\mathbf{z}_{t-1}^T]\right)}{\partial \mathbf{A}} \\
&= \frac{1}{2}\sum_{t=2}^{T}\left(\boldsymbol{\Sigma}^{-1}E[\mathbf{z}_{t-1}\mathbf{z}_t^T]^T+\boldsymbol{\Sigma}^{-1}E[\mathbf{z}_t\mathbf{z}_{t-1}^T]\right)-\frac{1}{2}\sum_{t=2}^{T}\left(\boldsymbol{\Sigma}^{-1}\mathbf{A}E[\mathbf{z}_{t-1}\mathbf{z}_{t-1}^T]+\boldsymbol{\Sigma}^{-1}\mathbf{A}E[\mathbf{z}_{t-1}\mathbf{z}_{t-1}^T]^T\right) \\
&= \sum_{t=2}^{T}\boldsymbol{\Sigma}^{-1}E[\mathbf{z}_t\mathbf{z}_{t-1}^T]-\sum_{t=2}^{T}\boldsymbol{\Sigma}^{-1}\mathbf{A}E[\mathbf{z}_{t-1}\mathbf{z}_{t-1}^T]=0 \\
\Rightarrow\quad \mathbf{A} &= \left(\sum_{t=2}^{T}E[\mathbf{z}_t\mathbf{z}_{t-1}^T]\right)\left(\sum_{t=2}^{T}E[\mathbf{z}_{t-1}\mathbf{z}_{t-1}^T]\right)^{-1}\quad ,
\end{aligned}$$

where the last step follows from pre-multiplying by matrix $\boldsymbol{\Sigma}$ and rearranging terms. Note that, not surprisingly, the solution is similar in form to that obtained for linear (auto-)regression models (cf. eq. 2.5 & 7.24), given here in terms of expectancies summed across time. The derivation of the other parameter estimates is no more complicated and can be gleaned from code **MATL7_7** (see also Bishop, 2006, for full details).

Finally, it should be mentioned that regularization techniques (cf. Ch. 2 & 3) could also be used to constrain parameters in the state space framework. For instance, Buesing et al. (2012) used such techniques to enforce stability (stationarity) of the latent dynamical process which is not guaranteed per se.

7.5.2 Gaussian Process Factor Analysis

Gaussian Process Factor Analysis (GPFA), developed by Yu et al. (2009; see also Lam et al. 2011), combines conventional factor analysis (sect. 6.4) with the assumption that a smooth Gaussian process connects observations consecutive in time. The concept was introduced to extract lower-dimensional smooth neural trajectories (where each process is allowed to evolve on its own typical time scale) from potentially much higher-dimensional neural recordings by exploiting correlations among neurons. GPFA provides a somewhat more general framework than the linear state-space models introduced in sect. 7.5.1 above, which they contain as a special case. GPFA consists of linear observation equations that relate a set of hidden factors $\{\mathbf{z}_t\}$ to the observed neural measurements $\{\mathbf{x}_t\}$ in a way identical to factor analysis:

(7.77) $\mathbf{x}_t = \boldsymbol{\mu} + \boldsymbol{\Gamma}\mathbf{z}_t + \boldsymbol{\eta}_t$, $\boldsymbol{\eta}_t \sim N(\mathbf{0},\boldsymbol{\Psi})$,

where $\boldsymbol{\Psi}$ is taken to be diagonal (all correlations among the outputs are introduced by mixing the latent states $\mathbf{z}_t$). With respect to the state transitions, in the GPFA framework the covariance structure across time is explicitly set up for each hidden factor series $\mathbf{z}_k = (z_{k1}\ldots z_{kT})$ by

(7.78) $\mathbf{z}_k \sim N(\mathbf{0}, \mathbf{\Sigma}_k)$, $\Sigma_{k,ij} = \sigma_k^2 e^{-(t_i - t_j)^2/(2\tau_k^2)} + \lambda_k^2 \, \mathrm{I}\{t_i = t_j\}$,

with $\Sigma_{k,ij}$ the $(i,j)^{th}$ element of $\mathbf{\Sigma}_k$, τ_k the exponential decay time of signal co-variance σ_k^2, I the indicator function, and λ_k^2 the noise variance. This explicit definition of the covariance across time is what makes this framework more flexible and general than a conventional linear state space model (where the form of the covariance is determined by the linear, first-order Markovian structure of the transition model).

Alternatively, one may define the hidden state dynamic through an AR(1) model as in the conventional state space setup by

(7.79) $\mathbf{z}_t = \mathbf{a}_0 + \mathbf{A}\mathbf{z}_{t-1} + \boldsymbol{\varepsilon}_t$, $\boldsymbol{\varepsilon}_t \sim N(\mathbf{0}, \mathbf{\Sigma})$

where $\mathbf{A}$ and $\mathbf{\Sigma}$ would have to be diagonal matrices to preserve the idea that factors $\mathbf{z}_k$ are uncorrelated (see sect 6.4). Estimation in this framework proceeds by the EM algorithm using Bayesian inference (see Yu et al. 2009 for details).

7.5.3 Latent variable models for count and point processes

In neuroscience, a number of authors have formulated general*ized* state space models for non-Gaussian observation processes, like spike trains or behavioral error counts (e.g. Smith & Brown 2003; Smith et al. 2004, 2007; Paninski et al. 2010, 2012; Pillow et al. 2011; Buesing et al. 2012; Latimer et al. 2015; Macke et al. 2015). Here we will cover those models which are still *linear in their transition equations* (thus are still limited in the repertoire of dynamical phenomena they can produce, see Ch. 9), while a discussion of fully nonlinear models will be deferred to Ch. 9. Even in these cases, where the transitions are linear but the output is non-Gaussian, parameter estimation commonly relies on approximate or numerical sampling methods, as the likelihood functions become intractable.

We start with a seminal contribution by Smith and Brown (2003) who related a first-order linear hidden state process $\{\mathbf{z}_t\}$, t=0…T, to the observed spike counts $c_t^{(i)}$, t=0…T, for each unit i by assuming Poisson outputs,

(7.80)
$$\begin{aligned} c_t^{(i)} \mid z_t &\sim Poisson\left[\lambda_t^{(i)}(z_t)\Delta t\right] \\ z_t &= \alpha z_{t-1} + \beta S_t + \varepsilon_t \;\; , \varepsilon_t \sim N(0, \sigma^2) \;\; , \end{aligned}$$

with the conditional intensity (instantaneous rate) function λ_t given by

(7.81) $\lambda_t^{(i)}(z_t) = \exp(\log[\eta_{0i}] + \eta_{1i} z_t)$.

S_t models a (known) external input to the system, and $\theta=\{\alpha,\beta,\sigma,\{\eta_{0i}\},\{\eta_{1i}\}\}$ are parameters, where the $\{\eta_{0i}\}$ reflect the constant background rates of the units.

The likelihood function for this model has the same general form as (7.52), and hence, again, one has to integrate across the whole unobserved (hidden) state path $\{\mathbf{z}_t\}$ for obtaining the likelihood $p(\{\mathbf{c}_t\}|\boldsymbol{\theta})$. To address this, like in conventional linear state space models, estimation and inference rely on the EM algorithm. Thus, as explained in sect. 7.5.1, the problem is broken down and solved iteratively by inferring the relevant moments of the conditional density $p(\{z_t\} \mid \{\mathbf{c}_t\}, \boldsymbol{\theta})$ given observed spike count data $\{\mathbf{c}_t\}$ and

parameters $\boldsymbol{\theta}$ (E-step), and obtaining estimates of parameters $\boldsymbol{\theta}$ in the M-step through maximization of the expected complete data log-likelihood, $E_{\mathbf{Z}}[\log p(\{\mathbf{c}_t, z_t\}|\boldsymbol{\theta})]$. With the Gaussian assumptions for the AR hidden process combined with the Poisson output assumption, the complete data log-likelihood of this model follows as (Smith & Brown 2003):

$$\log p(\{\mathbf{c}_t, z_t\}|\boldsymbol{\theta}) = \log p(\{\mathbf{c}_t\}|\{z_t\}, \boldsymbol{\theta}) + \log p(\{z_t\}|\boldsymbol{\theta}) \tag{7.82}$$

with

$$\log p(\{c_t^{(i)}\}|\{z_t\}, \boldsymbol{\theta}) = \sum_{t=0}^{T} \log\left(\frac{(\lambda_t^{(i)}\Delta t)^{c_t^{(i)}}}{c_t^{(i)}!} e^{-\lambda_t^{(i)}\Delta t} \right) = \sum_{t=0}^{T} \left(c_t^{(i)} \log(\lambda_t^{(i)}\Delta t) - \log c_t^{(i)}! - \lambda_t^{(i)}\Delta t \right)$$

and

$$\log p(\{z_t\}|\boldsymbol{\theta}) = -\frac{1}{2}\log\left[\frac{2\pi\sigma^2}{1-\alpha^2}\right] - \frac{T}{2}\log(2\pi\sigma^2) - \frac{1}{2}\left(\frac{1-\alpha^2}{\sigma^2} z_0^2 + \sum_{t=1}^{T} \frac{(z_t - \alpha z_{t-1} - \beta S_t)^2}{\sigma^2} \right).$$

We stress, once again, that the counts $c_t^{(i)}$ are conditionally independent both from each other and in time given the hidden process $\{z_t\}$. Thus, the log-probability $\log p(\{c_t^{(i)}\}|\{z_t\}, \boldsymbol{\theta})$ for a single unit *i* can be expressed as a sum, and likewise the total probability $p(\{\mathbf{c}_t\}|\{z_t\}, \boldsymbol{\theta})$ factorizes into a product of the individual terms. We also went with Smith & Brown (2003) in assuming that the first hidden state, z_0, follows a Gaussian with zero mean and variance $\sigma^2/(1-\alpha^2)$ (cf. sect. 7.2). Smith and Brown (2003) furthermore assumed the bin width Δt to be small enough for the counts $c_t^{(i)}$ to take on only the values 0 or 1 (i.e., a Bernoulli process). In that case one has $\log c_t^{(i)}! = 0$ in the sum of eq. 7.82, 2nd row, although from the perspective of maximization these terms drop out as constants anyway. Like before (cf. eq. 7.33), the likelihood $\log p(\{z_t\}|\boldsymbol{\theta})$ (eq. 7.82, last row) can be expressed in terms of the error process $\{\varepsilon_t\}$, which by independence qua assumption factorizes into a product of Gaussians or, equivalently, a multivariate Gaussian with the sum of all the individual terms in the exponent, where the first term z_0 requires special treatment.

Based on eq. 7.82, the total expected log-likelihood is

$$\begin{aligned} E_{\{z_t\}}\left[\log p(\{\mathbf{c}_t, z_t\}|\boldsymbol{\theta})\right] = {} & E\left[\sum_{t=0}^{T}\sum_{i=1}^{N} c_t^{(i)} (\log\eta_{0i} + \eta_{1i} z_t + \log\Delta t) - e^{\log[\eta_{0i}] + \eta_{1i} z_t}\Delta t \,\middle|\, \boldsymbol{\theta} \right] + \\ & E\left[-\frac{1}{2}\sum_{t=1}^{T} \frac{(z_t - \alpha z_{t-1} - \beta S_t)^2}{\sigma^2} - \frac{T}{2}\log 2\pi\sigma^2 \,\middle|\, \boldsymbol{\theta} \right] + \\ & E\left[\frac{1}{2}\log\left(\frac{1-\alpha^2}{\sigma^2}\right) - \frac{z_0^2(1-\alpha^2)}{2\sigma^2} \,\middle|\, \boldsymbol{\theta} \right] \quad , \end{aligned} \tag{7.83}$$

with the first term coming from the Poisson likelihood $p(\{\mathbf{c}_t\}|\{z_t\}, \boldsymbol{\theta})$, and the second and third from the Gaussian $\log p(\{z_t\}|\boldsymbol{\theta})$. As in linear state space models (sect. 7.5.1), a crucial insight is that for computing the expected log-likelihood only the expectancies $E[z_t|\boldsymbol{\theta}]$, $E[z_t^2|\boldsymbol{\theta}]$, and $E[z_t z_{t-1}|\boldsymbol{\theta}]$ are needed here. In this case, due to the Poisson assumption which causes z_t to occur in the exponent within the first expectancy in (7.83), this may be a bit harder to see. It can be derived, however, from the so-called moment-

generating function of the Gaussian (e.g. Wackerly et al. 2008) which holds

$$(7.84)\quad E\left[e^{\eta_{1i} z_t}\right] = \exp\left(\eta_{1i} E[z_t] + \frac{\eta_{1i}^2 Var(z_t)}{2}\right) = \exp\left(\eta_{1i} E[z_t] + \frac{\eta_{1i}^2 (E[z_t^2] - E[z_t]^2)}{2}\right).$$

Note that z_t is indeed a Gaussian random variable according to model definition (7.80). As in the standard linear-Gaussian state space setting, the idea is to compute these expectancies via the Kalman-filter-smoother recursions, using the general factorization given by (7.56) and (7.65). There is, unfortunately, a nasty complication, however, brought in by the Poisson observation assumption.

Before getting into this, let us first reformulate model (7.80)-(7.81) more generally in terms of a *q*-variate latent process $\{\mathbf{z}_t\}$:

$$(7.85)\quad \begin{aligned} & c_t^{(i)} \,|\, \mathbf{z}_t \sim Poisson\left[\lambda_t^{(i)} \Delta t\right] \;\text{ with }\; \lambda_t^{(i)} = \exp(\log[\eta_{0i}] + \boldsymbol{\eta}_{1i} \mathbf{z}_t)\,, \\ & \mathbf{z}_t = \mathbf{A}\mathbf{z}_{t-1} + \mathbf{B}\mathbf{s}_t + \boldsymbol{\varepsilon}_t \;\;, \boldsymbol{\varepsilon}_t \sim N(\mathbf{0}, \boldsymbol{\Sigma})\,, \\ & \mathbf{z}_1 \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma})\,, \end{aligned}$$

where $\boldsymbol{\eta}_{1i}$ is a (1 x *q*) row vector now, specific for each observed unit *i*=1…*p*, and we have modified the assumptions for the initial state to follow model (7.53). For clarity, let us also restate factorization (7.57) here using the notation from model (7.85):

$$(7.86)\quad \begin{aligned} p(\mathbf{z}_t \,|\, \{\mathbf{c}_{\tau \le t}\}, \boldsymbol{\theta}) &= \frac{p(\mathbf{z}_t, \mathbf{c}_t, \{\mathbf{c}_{\tau \le t-1}\} \,|\, \boldsymbol{\theta})}{p(\mathbf{c}_t \,|\, \{\mathbf{c}_{\tau \le t-1}\}, \boldsymbol{\theta}) p(\{\mathbf{c}_{\tau \le t-1}\} \,|\, \boldsymbol{\theta})} \\ &= \frac{p(\mathbf{c}_t \,|\, \mathbf{z}_t, \{\mathbf{c}_{\tau \le t-1}\}, \boldsymbol{\theta}) p(\mathbf{z}_t \,|\, \{\mathbf{c}_{\tau \le t-1}\}, \boldsymbol{\theta})}{p(\mathbf{c}_t \,|\, \{\mathbf{c}_{\tau \le t-1}\}, \boldsymbol{\theta})} \\ &= \frac{p(\mathbf{c}_t \,|\, \mathbf{z}_t, \boldsymbol{\theta}) \int p(\mathbf{z}_t \,|\, \mathbf{z}_{t-1}, \boldsymbol{\theta}) p(\mathbf{z}_{t-1} \,|\, \{\mathbf{c}_{\tau \le t-1}\}, \boldsymbol{\theta}) d\mathbf{z}_{t-1}}{p(\mathbf{c}_t \,|\, \{\mathbf{c}_{\tau \le t-1}\}, \boldsymbol{\theta})} \end{aligned}$$

Although, as before, this yields a recursive prescription for computing $p(\mathbf{z}_t \,|\, \{\mathbf{c}_{\tau \le t}\}, \boldsymbol{\theta})$ from $p(\mathbf{z}_{t-1} \,|\, \{\mathbf{c}_{\tau \le t-1}\}, \boldsymbol{\theta})$ (as indicated in yellow), this density is not Gaussian since $p(\mathbf{c}_t \,|\, \mathbf{z}_t, \boldsymbol{\theta})$ is Poisson, and the recursions in time crumble down. Smith & Brown (2003) therefore decided to try a Gaussian approximation for the left hand side of (7.86), $p(\mathbf{z}_t \,|\, \{\mathbf{c}_{\tau \le t}\}, \boldsymbol{\theta}) \approx N(\boldsymbol{\mu}_t, \mathbf{V}_t) =: \alpha_t$. In that case, the integral in (7.86), now involving two Gaussians (one for the transition, and one by assumption), resolves in exactly the same way as derived in (7.58) to (7.62) in sect 7.5.1. Recall that the result was $N(\mathbf{A}\boldsymbol{\mu}_{t-1} + \mathbf{B}\mathbf{s}_t, \mathbf{L}_{t-1})$ with covariance $\mathbf{L}_{t-1} = \mathbf{A}\mathbf{V}_{t-1}\mathbf{A}^T + \boldsymbol{\Sigma}$ (the only difference here is that the stimulus $\mathbf{B}\mathbf{s}_t$ contributes to the mean; since it's assumed to be fixed, however, it does not affect the variance: it will not show up in the terms quadratic in $\mathbf{z}_t$ in eqn. 7.59-7.62).

How do we obtain the covariance matrix $\mathbf{V}_t$ and mean $\boldsymbol{\mu}_t$ of this approximate Gaussian? Since the mean and mode coincide for the Gaussian, a reasonable estimate for $\boldsymbol{\mu}_t$ is obtained by *maximizing* (7.86) or the logarithm of this expression (which won't change the position of the mode as it's monotonic). Moreover, note that the second derivative of a log of a Gaussian (a quadratic function in $\mathbf{z}_t$) is its negative inverse covariance matrix (as the reader may verify her-/himself). Thus, a reasonable procedure is to instantiate the mean of $p(\mathbf{z}_t \,|\, \{\mathbf{c}_{\tau \le t}\}, \boldsymbol{\theta}) \approx N(\boldsymbol{\mu}_t, \mathbf{V}_t)$ with the maximizer of the logarithm of

(7.86), and the covariance with the negative inverse Hessian matrix of second derivatives. In fact, since the denominator of (7.86) will drop out as a constant for this maximization w.r.t. $\mathbf{z}_t$, one can focus on maximizing the numerator alone. The function to be maximized thus becomes

$$
\begin{aligned}
Q(\mathbf{z}_t) &:= \log p(\mathbf{c}_t \mid \mathbf{z}_t, \boldsymbol{\theta}) + \log N(\mathbf{A}\boldsymbol{\mu}_{t-1} + \mathbf{B}\mathbf{s}_t, \mathbf{L}_{t-1}) + const. \\
&= \sum_{i=1}^{N} c_t^{(i)} (\log \eta_{0i} + \boldsymbol{\eta}_{1i}\mathbf{z}_t) - \sum_{i=1}^{N} \eta_{0i} e^{\boldsymbol{\eta}_{1i}\mathbf{z}_t} \Delta t \\
&\qquad - \frac{1}{2}\log|\mathbf{L}_{t-1}| - \frac{1}{2}(\mathbf{z}_t - \mathbf{A}\boldsymbol{\mu}_{t-1} - \mathbf{B}\mathbf{s}_t)^T \mathbf{L}_{t-1}^{-1}(\mathbf{z}_t - \mathbf{A}\boldsymbol{\mu}_{t-1} - \mathbf{B}\mathbf{s}_t) + const. \\
&= \mathbf{c}_t^T \log \boldsymbol{\eta}_0 + \mathbf{c}_t^T \boldsymbol{\Gamma}\mathbf{z}_t - \boldsymbol{\eta}_0^T e^{\mathbf{H}\mathbf{z}_t} \Delta t - \frac{1}{2}(\mathbf{z}_t - \mathbf{A}\boldsymbol{\mu}_{t-1} - \mathbf{B}\mathbf{s}_t)^T \mathbf{L}_{t-1}^{-1}(\mathbf{z}_t - \mathbf{A}\boldsymbol{\mu}_{t-1} - \mathbf{B}\mathbf{s}_t) + const.
\end{aligned}
\tag{7.87}
$$

Taking first and second derivatives of this expression one arrives at

$$
\begin{aligned}
\frac{\partial Q(\mathbf{z}_t)}{\partial \mathbf{z}_t} &= \mathbf{H}^T \mathbf{c}_t - \mathbf{H}^T (\boldsymbol{\eta}_0 \circ e^{\mathbf{H}\mathbf{z}_t}) \Delta t - \mathbf{L}_{t-1}^{-1}(\mathbf{z}_t - \mathbf{A}\boldsymbol{\mu}_{t-1} - \mathbf{B}\mathbf{s}_t) \\
\frac{\partial^2 Q(\mathbf{z}_t)}{\partial \mathbf{z}_t^2} &= -\mathbf{H}^T (\boldsymbol{\eta}_0 \circ e^{\mathbf{H}\mathbf{z}_t} \circ \mathbf{I}) \mathbf{H} \Delta t - \mathbf{L}_{t-1}^{-1} \quad ,
\end{aligned}
\tag{7.88}
$$

where '$\circ$' denotes the element-wise product. Note that the first derivatives $\partial Q(\mathbf{z}_t)/\partial \mathbf{z}_t$ contain both sums of exponentials and terms linear in $\mathbf{z}_t$, and therefore elude an analytical solution. Rather, at each recursion step we have to move through a couple of Newton-Raphson iterations (or some other numerical procedure, see sect. 1.4) using derivatives (7.88) to obtain the estimate for $\boldsymbol{\mu}_t$. We can then evaluate $\partial^2 Q(\mathbf{z}_t)/\partial \mathbf{z}_t^2$ at $\mathbf{z}_t^{\max} = \boldsymbol{\mu}_t$ (or just use the matrix from the last Newton-Raphson iteration, tolerating a slightly larger error), and arrive at $p(\mathbf{z}_t \mid \{\mathbf{c}_{\tau \le t}\}, \boldsymbol{\theta}) \approx N(\boldsymbol{\mu}_t, \mathbf{V}_t)$.

Looking back at eq. 7.66, note that the Kalman smoother steps only rely on densities $N(\mathbf{A}\boldsymbol{\mu}_{t-1} + \mathbf{B}\mathbf{s}_t, \mathbf{L}_{t-1})$ and $\alpha_t := N(\boldsymbol{\mu}_t, \mathbf{V}_t)$ already computed in the forward recursions, on the Gaussian transition density $p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t)$, and on the full posterior $\gamma_t := p_{\boldsymbol{\theta}}(\mathbf{z}_t \mid \mathbf{c}_1 ... \mathbf{c}_T) \approx N(\tilde{\boldsymbol{\mu}}_t, \tilde{\mathbf{V}}_t)$ computed in the previous step and initialized with $\gamma_T = \alpha_T$. Hence, for the backward smoother recursions, everything remains within the Gaussian setting and proceeds exactly as derived in eqn. 7.66-7.70, sect. 7.5.1. The same is true for estimation of covariance expectancies $E[z_t z_{t-1}]$ which are given by (7.72).

For the M-step, the transition equation parameters α (or **A**), β (**B**), σ (Σ), and $\mathbf{z}_0$ ($\boldsymbol{\mu}_0$) occurring in the Gaussian terms of likelihood eq. 7.83 (or its multivariate generalization) can be solved for analytically as in the standard linear state space model (sect. 7.5.1), since the Poisson terms drop out for this maximization. This yields a set of equations linear in these parameters, since the log-likelihood contains them either in quadratic form or in the log-term (recall that $\partial \log \sigma^2 / \partial \sigma = 2/\sigma$). The only exception is parameter α in the original formulation by Smith & Brown, which gives a cubic equation since it occurs also in the $\log(1-\alpha^2)$ -term in the last row of eq. 7.83; Smith & Brown, however, simply dropped this last term for maximization w.r.t. α since it will have only a minor contribution anyway (while in our multivariate model we did not include this assumption to begin with). Things are not quite that easy with the parameters governing the Poisson observation equations,

especially with the $\{\eta_{1i}\}$ (or $\{\boldsymbol{\eta}_{1i}\}$), since they give rise to sums of linear and exponential forms. But we can use Newton-Raphson iterations again for this maximization (described in more detail in sect. 9.3 where nonlinear, non-Gaussian models are discussed).

Let us briefly discuss an alternative route to state and parameter estimation in these non-Gaussian models (also suitable more generally for models with nonlinear transition equations, sect. 9.3), based on the Laplace-approximation (Koyama et al. 2010; Paninski et al. 2010; Macke et al. 2015). The Laplace approximation is a general method for solving integrals of the form $\int e^{f(x)}dx$ based on a Taylor series expansion of f(x) around the global maximum x_0. It works well if there is a unique global maximum, with f(x) decaying quite sharply as x moves away from x_0. Around the maximum, $f(x) \approx f(x_0) + [(x-x_0)^2/2]f''(x_0)$ since $f'(x_0) = 0$. Defining $f(\mathbf{z}) := \log[p(\mathbf{x}\,|\,\mathbf{z},\boldsymbol{\theta})p(\mathbf{z}\,|\,\boldsymbol{\theta})]$, where $\mathbf{z} = (\mathbf{z}_1,...,\mathbf{z}_T)$ concatenates the whole hidden state path into one vector, one could thus approximate log-likelihood (7.52) by

(7.89)

$$
\begin{aligned}
p(\mathbf{x}\,|\,\boldsymbol{\theta}) &= \int_{\mathbf{z}} e^{f(\mathbf{z})}d\mathbf{z} \approx e^{f(\mathbf{z}^{\max})}\int_{\mathbf{z}} e^{-\frac{1}{2}(\mathbf{z}-\mathbf{z}^{\max})^T(-\mathbf{H}_{\max})(\mathbf{z}-\mathbf{z}^{\max})}d\mathbf{z} \\
&= p(\mathbf{x}\,|\,\mathbf{z}^{\max},\boldsymbol{\theta})p(\mathbf{z}^{\max}\,|\,\boldsymbol{\theta})(2\pi)^{q/2}\,|-\mathbf{H}_{\max}|^{-1/2}\int_{\mathbf{z}}(2\pi)^{-q/2}\,|-\mathbf{H}_{\max}|^{1/2}\,e^{-\frac{1}{2}(\mathbf{z}-\mathbf{z}^{\max})^T(-\mathbf{H}_{\max})(\mathbf{z}-\mathbf{z}^{\max})}d\mathbf{z} \\
&= p(\mathbf{x}\,|\,\mathbf{z}^{\max},\boldsymbol{\theta})p(\mathbf{z}^{\max}\,|\,\boldsymbol{\theta})(2\pi)^{q/2}\,|-\mathbf{H}_{\max}|^{-1/2} \\
\Rightarrow \quad \log p(\mathbf{x}\,|\,\boldsymbol{\theta}) &\approx \log p(\mathbf{x}\,|\,\mathbf{z}^{\max},\boldsymbol{\theta}) + \log p(\mathbf{z}^{\max}\,|\,\boldsymbol{\theta}) - \frac{1}{2}\log|-\mathbf{H}_{\max}| + const.
\end{aligned}
$$

where $\mathbf{H}_{\max} := \left(\frac{\partial^2 f(\mathbf{z}^{\max})}{(\partial\mathbf{z}^{\max})^2}\right)$ is the Hessian matrix of second derivatives at the maximum a posteriori path $\mathbf{z}^{\max}$. Thus, the trick is that by means of the Taylor expansion around $\mathbf{z}^{\max}$ we obtain a Gaussian integral with the negative inverse Hessian $\mathbf{H}_{\max}$ as covariance matrix which evaluates to 1. We may now iteratively maximize $f(\mathbf{z})$ w.r.t. $\mathbf{z}$, and then expression (7.89) w.r.t $\boldsymbol{\theta}$ given $\mathbf{z}^{\max}$ using, e.g., Newton-Raphson steps in both maximizations, or we could in fact attempt to solve for $\{\mathbf{z}, \boldsymbol{\theta}\}$ jointly based on (7.89). Paninski et al. (2010) and Pillow et al. (2011) discuss several examples where they used this approach, e.g. for inferring the synaptic inputs underlying observed membrane potential dynamics, or for stimulus decoding, and for which (7.89) is log-concave yielding a unique maximum.

State space models with non-Gaussian observations found a number of different applications in neuroscience. A model with Poisson output (observation) equation and linear (AR) Gaussian hidden state dynamic similar to (7.85) was used, for instance, by Yu et al. (2007) for decoding movement trajectories and goals from multiple single-unit spiking activity. In Latimer et al. (2015) a formalism like (7.85) was used to infer the state (and parameters) of a drift-diffusion-type model of decision making (Ratcliff 1978; see sect. 7.6 below) from multiple single-unit recordings performed in the macaque lateral intraparietal area. Smith et al. (2004, 2007) developed state space models to capture the dynamics of behavioral learning processes defined by a (time) series of correct and incorrect responses. These models consist of an unobserved learning state x_t which simply follows

an AR Gaussian random walk, connected to a Bernoulli observation process via a logistic function (cf. eq. 3.16) modeling the response probability. Shimazaki et al. (2012) used a state space framework to account for non-stationarity in the parameters governing a multivariate Bernoulli spike process through a linear transition process for the parameters.

7.6 Computational and neuro-cognitive time series models

Most of the time series models described so far (sect. 7.2-7.5) are general purpose models that relate consecutive measurements through time in arbitrary time series (although some of the models discussed were already formulated with neural systems in mind). The variables and parameters in these models do not per se have any meaning but find their specific interpretation in the scientific context at hand. However, the statistical machinery for parameter estimation and testing developed in this and previous chapters could in principle also be applied to time series models constructed from variables which per se represent specific theoretical quantities, for instance in models of cognitive or neural processes. Or, formulated more from the statistical perspective, incorporating, from the outset, domain-specific knowledge into statistical model inference may boost our ability to detect scientifically meaningful patterns in the data. In this section we would like to highlight a recent trend in theoretical and cognitive neuroscience, especially in the areas of reinforcement learning and decision making, where explanatory, computational models are combined with probability assumptions (Balleine & O'Doherty 2010; Dayan & Daw 2008; Daw et al. 2005; Badre et al. 2012; Brunton et al. 2013). This is a powerful approach to look deeper into the computational mechanisms that generated the data at hand, and directly probe assumptions that would provide a theoretical explanation. Placing computational models firmly into a statistical framework this way, it becomes possible to directly test different hypotheses about the computational mechanisms that presumably underlie the observed data. It will also give us formal criteria, like estimators of prediction error, according to which one can judge the appropriateness of the developed model for explaining the data.

As a prominent example, we will focus on computational reinforcement learning theory (Sutton & Barto 1998), a branch of machine learning that originated from findings in behaviorist psychology. It is centered around the idea that organisms strive to maximize their present and future rewards. To do so, for each situation s, and each action a that can be performed in that situation, i.e. for each situation-action pair (s, a), they learn a value (function) $V(s,a)$ which is iteratively updated with repeated experience on (s, a). In the simplest case, this update simply follows the amount of reward r_t the animal received at time t upon executing a in s (punishment may be conceived as negative reward in this framework):

$$V_{t+1}(s,a) = V_t(s,a) + r_{t+1}\,. \tag{7.90}$$

(Note that this update-rule, while linear, is a perfect integration and thus non-stationary.) More sophisticated animals (agents) would, however, usually also take future rewards into account that may follow from choosing a in situation s. Empirically, future rewards are usually discounted by some factor $\gamma^{\Delta t}$ as a function of time (Domjan 2003). Hence, ideally, the total value of (s, a) should reflect the expected sum of present and future discounted rewards when choosing the optimal path of actions (i.e., in each subsequent situation $s_{t+\Delta t}$ that action $a_{t+\Delta t}$ that maximizes reward prediction; Bellman 1957; Sutton & Barto 1998):

$$V(s_t,a_t) \overset{!}{=} E\left[\sum_{i=0}^{\infty}\gamma^i r_{t+i}\middle| s_t,a_t\right]. \tag{7.91}$$

This makes sense from an evolutionary and economical perspective, as the future is uncertain (the world is not stationary), and as this uncertainty usually grows the more distant into the future an event is (γ may be seen as a way to implement this decay in reward probability). Or more proximal rewards may seem more valuable partly because the lifetime of an animal is limited. Either way, this idea can be incorporated into the value-update-rule by noting that reward predictions across subsequent situations and actions should be consistent (Bellman 1957; Barto 1995), i.e.

(7.92)

$$V(s_{t+1},a_{t+1})-V(s_t,a_t)\overset{!}{=}E\left[\sum_{i=1}^{\infty}\gamma^{i-1}r_{t+i}\right]-E\left[\sum_{i=0}^{\infty}\gamma^{i}r_{t+i}\right]=E\left[\sum_{i=1}^{\infty}\gamma^{i-1}r_{t+i}\right]-\left(\gamma E\left[\sum_{i=1}^{\infty}\gamma^{i-1}r_{t+i}\right]+E[r_t]\right)$$

$$\Rightarrow\quad V(s_t,a_t)\overset{!}{=}\gamma V(s_{t+1},a_{t+1})+E[r_t]$$

,

where we have assumed that the agent chooses the *optimal* action in each situation (according to the expectancies given by eq. 7.91), and that – given this – transitions among situations are deterministic (note that for clarity the explicit dependence of the expectancy values on s_t, a_t, was dropped from the notation). The difference between the bottom left and right hand sides in eq. 7.92

(7.93) $\partial_t := \gamma V(s_{t+1},a_{t+1})+r_t-V(s_t,a_t)$

with actually observed reward r_t is called the *temporal difference error* (TDE; or reward prediction error), and can be used to update the values in each step according to

(7.94) $V_{t+1}(s,a)=V_t(s,a)+\alpha\partial_t$,

where α is a learning rate (Barto 1995; Sutton & Barto 1998). One may interpret the updating according to future expected rewards as a kind of 'reward diffusion' from future situations to the present one, a process which in behaviorist language is related to the idea of higher-order (secondary, tertiary, etc.) reinforcers (Domjan 2003).

Some neurophysiological findings relate the TDE (7.93) to the activity of neurons in the midbrain ventral tegmentum (VTA) and substantia nigra (SN; Schultz et al. 1997), which has steered a wave of excitement in the animal and artificial learning literature. During classical and operant conditioning tasks, VTA/SN neurons initially respond (via firing rate increase) only to the unconditioned stimulus (US). During the course of learning, these responses to the now expected US vanish and shift to the predicting conditioned stimulus (CS). When a predicted US is omitted, VTA/SN respond at the expected time of occurrence with a temporary decrease in their firing rate (Schultz et al. 1997). Thus, occurrence and sign of firing rate changes in VTA/SN neurons appear to comply with eq. 7.93.

Note that in the basic form of the model above the value *V* is a function of the present state-action pair (*s*, *a*) only, i.e. the next time-step and all future rewards are fully predicted from the present state-action pair (*s*, *a*). In this sense the model is first-order Markovian like the state-space models introduced in sect. 7.5.1 above. However, one may assume that information about the past is incorporated into the present state representation *s*, for instance in the form of a short- or long-term memory trace. Thus, *s* may encompass internal in addition to external information. The model can also easily be

extended to continuously-valued state and action representations, when *s* for instance represents variables like movement velocity or direction, or *a* the amount of force applied with the hand. In this case, *V* may not be defined in terms of a lookup-table $(s, a) \rightarrow V(s,a)$ as in the discrete state space case, but as a continuous (regression) function of *s* and *a*.

Leaving these details aside, the question now is how, specifically, values $V(s,a)$ are translated into choices of a particular path of actions. A common choice is a 'Boltzmann-type' decision function which chooses an action *a* in situation *s* with a probability corresponding to that action's value (Sutton & Barto 1998):

$$\text{(7.95)} \quad \Pr(a_t = k \mid s_t = m) = \frac{e^{\beta V(k,m)}}{\sum_l e^{\beta V(l,m)}}$$

where parameter β is an 'inverse temperature' which regulates the 'flatness' of the decision landscape: For $\beta \rightarrow \infty$, the agent will deterministically always choose the highest-valued action ('exploit'), while for $\beta \rightarrow 0$, it will choose each action with equal likelihood regardless of value ('explore'). Thus β determines the 'exploitation-exploration-tradeoff'.

There is an burgeoning literature in computer science and artificial intelligence on how to use such learning algorithms and derivatives thereof to train artificial agents (robots) to perform tasks like playing backgammon or checkers, balancing, planning and heuristic search, and so on (see Sutton & Barto 1998; Mnih et al. 2015). Fig. 7.9 (**MATL7_8**) gives an example of an RFL model trained to find reward locations in a virtual maze to which it is repeatedly exposed (i.e., undergoes multiple identical trials).

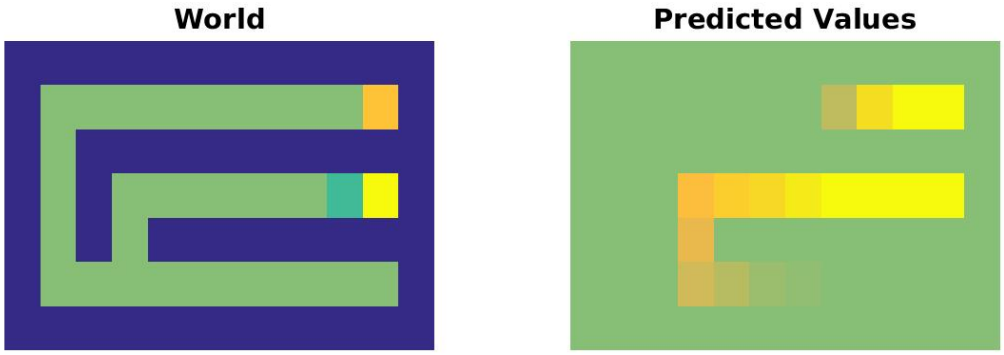

Fig. 7.9. Illustration of reinforcement learning on a virtual maze. Left graph shows the maze with reward locations and magnitudes (reddish and yellow squares), and current position of the agent (dark green), as implemented in **MATL7_8**. Right matrix displays estimated discounted future reward values after several runs.

We will now return to the issue of parameter estimation and the question of how to fit such models to experimental data to gain some insight into underlying mechanisms. Given a time series $\{(s_t,a_t)\}$ of experimentally observed situations and actions performed by an animal, a likelihood function for the system parameters $\boldsymbol{\theta}=\{\beta,\alpha,\gamma\}$ can be constructed as

$$\text{(7.96)} \quad \begin{aligned} p(\{(s_t,a_t)\} \mid \alpha,\beta,\gamma) &= p(s_0,a_0 \mid \boldsymbol{\theta}) \prod_{t=1}^{T} p(s_t,a_t \mid \{s_\tau,a_\tau \mid \tau < t\},\boldsymbol{\theta}) \\ &= p(a_0 \mid s_0,\boldsymbol{\theta}) p(s_0 \mid \boldsymbol{\theta}) \prod_{t=1}^{T} p(a_t \mid s_t,\{s_\tau,a_\tau \mid \tau < t\},\boldsymbol{\theta}) p(s_t \mid \{s_\tau,a_\tau \mid \tau < t\},\boldsymbol{\theta}) \end{aligned}$$

One may interpret the TDE-model as specified above as a general*ized* state space model with *deterministic* linear transition equation (7.93-7.94) and nonlinear measurement equation (7.95) which links the outputs *a* to the underlying state values *V* by means of the multinomial distribution in the discrete case (the linearity in the transition is why we have included this model class here, in Ch. 7, rather than with Ch. 8 or 9). The fact that the update equations for the hidden state *V* are deterministic in the basic model (and the sequence of 'innovations' r_t is observed as well) simplifies estimation considerably compared to a full state space model, as one does not have to integrate across sets of unobserved state paths. Although this simplifies estimation, scientifically it may be more appropriate to account for randomness in the value updating as well. As noted in sect. 7.5.1, inference for the deterministic transitions may otherwise become derailed by noise fluctuations in the true generating process. A full state space framework may also offer some protection against errors introduced by invalid model assumptions. As discussed in sect. 7.5.3 and 9.3.1, approximate EM schemes are available for this case with non-Gaussian observations.

For simplicity, we will furthermore focus on the scenario where transitions among states *s* and reward feedbacks *r* are deterministic (qua experimental design) given the actions *a* (i.e., $p(s_t \mid \{s_\tau, a_\tau \mid \tau < t\}) = 1$ for one specific situation *m*, and 0 for the others), such that everything is fully determined by the course of actions taken and all terms depending solely on *s* could be dropped from (7.96). In this case, since all knowledge about the past course of actions is embodied within the current values *V*(*s*,*a*), the log-likelihood simplifies to

$$\begin{aligned} \log p(\{a_t\} \mid \alpha, \beta, \gamma) &= \sum_{t=0}^{T} \log p(a_t \mid s_t, \{a_\tau, s_\tau \mid \tau < t\}, \boldsymbol{\theta}) = \sum_{t=0}^{T} \log \frac{e^{\beta V_t(a_t = i \mid s_t, \alpha, \gamma)}}{\sum_j e^{\beta V_t(a_t = j \mid s_t, \alpha, \gamma)}} \\ &= \sum_{t=0}^{T} \left[\beta V_t(a_t = i \mid s_t, \alpha, \gamma) - \log \sum_j e^{\beta V_t(a_t = j \mid s_t, \alpha, \gamma)} \right] , \end{aligned} \tag{7.97}$$

where *i* denotes the actually chosen action on each trial. Since the transition process for V_t (and the environment) is deterministic, the V_t are completely specified by the actual path of actions $\{a_t\}$ taken by the animal and the actual reward feedbacks received. One then proceeds as usual by maximizing (7.97) w.r.t. β, α, γ.

Formal statistical tests on hypotheses like, e.g., H_0: γ=0 (only present, no future rewards are considered), may be conducted using likelihood ratio statistics (see eqn. 1.35, 7.35, or 7.50) as long as models are strictly nested. Otherwise, criteria like AIC, BIC, or CVE-based procedures may be used for model comparison and selection, or to assess the prediction quality of a given model in the first place (cf. Ch. 4). If there is uncertainty in the environment or noise in the value updating, a more elaborated state space approach may have to be considered (cf. sect. 7.5 and 9.3.1).

RFL models are used increasingly in the analysis of behavioral, human neuroimaging, or animal in-vivo electrophysiological data (Frank et al. 2004, 2009; Daw et al. 2006; O'Doherty et al. 2007; Schonberg et al. 2010). Often the model is first estimated by maximum likelihood or Bayesian inference from the observed behavioral data as illustrated above, and model parameters (like learning rate α or discount factor γ) or variables (like values *V*(*s*,*a*)) are subsequently used as predictors in (usually linear) regression models to account for variation in the simultaneously recorded neural activity. For instance, Khamassi et al. (2014) used this approach for the analysis of in-vivo electrophysiological recordings by first probing which of a variety of RFL model variants

could best account for the observed behavioral data in a sequential search task (using criteria like the BIC, sect. 4.1). The best-performing model was then used to analyze single neuron recordings from the lateral prefrontal cortex and dorsal anterior cingulate cortex to figure out the neural processes underlying the behavioral performance, in particular the distinct roles of these two brain areas and mechanisms which trade exploration initially in trials for exploitation later on (i.e., the regulation of β in eq. 7.95). Similarly, Sul et al. (2010) used RFL models to examine how processes like prediction errors or value updating were neuronally represented in in-vivo electrophysiological recordings from the rodent orbitofrontal and medial prefrontal cortex.

Although RFL models are probably the ones which have been employed most frequently with this type of approach, they are of course not the only ones which could be used. In principle, one could try to estimate any formal model of an animal's behavior from observed data to which it applies, although in the more complicated, nonlinear cases one may have to resort to approximate numerical techniques or sampling schemes to evaluate likelihood functions or posteriors (cf. sect. 9.3). A close relative of RFL models are belief learning models which originated in economics to account for subjects' learning behavior in game-theoretical settings (Camerer & Ho 1999). These models are similar to RFL models in the sense that they contain update mechanisms upon experienced outcomes (returns), and in fact encompass RFL models as a special case for specific settings of the parameters, at least in the formulation by Camerer & Ho (1999). Like for RFL models, parameters of these models can be obtained by maximum likelihood principles from observed experimental data (Camerer & Ho 1999). Crucially, however, whereas in conventional RFL models only values for the selected action and current situation are updated, in belief learning the beliefs about what the subject would have earned had it chosen a different action given the opponent's response are updated as well. Belief learning models estimated from behavioral data have been used for instance in the analysis of how genetic differences (exploiting natural polymorphisms) affect learning in strategic settings (Set et al. 2014). Another example is given by Brunton et al. (2013) who developed noisy accumulator models (based on the drift-diffusion models introduced by Ratcliff 1978; Ratcliff & McKoon 2008) for the process of evidence integration and decision making in rats and humans. The model has the form of a linear state space model with Gaussian noise and sensory inputs, consisting of a 'decision variable' which triggers the subject's choice once a threshold is crossed, and a variable that models sensory adaptation. Parameters were estimated from the series of subjects' choices by maximum likelihood. Brunton et al. (2013) used their model to differentiate the role of various internal and external noise sources in the decision making process.

As a final remark, although quite successful, most of these behavioral models are *linear* in their transitions (which is why they were included here in Ch. 7), but as will become clear in Ch. 9, linear models are quite limited in the repertoire of dynamical phenomena they can produce (e.g., they cannot produce stable oscillations on their own). A number of interesting behavioral time series may therefore require a shift to nonlinear models.

7.7 Bootstrapping time series

Time series data may potentially bear a highly complicated dependency structure and unusual distributional properties, depending on the precise nature of the generating dynamical system (see Ch. 9) and the level of noise. Think for instance of the membrane potential distribution produced by a spiking neuron. Physiological distributions may exhibit quite sharp cutoffs, unusual tails, or multimodality, induced by biophysical processes and constraints. Examples are the absolute refractory period limiting the maximum spike rate, or reversal potentials confining the voltage distribution. In the preceding sections we have

discussed several test statistics for linear and generalized models based on conventional parametric distributions. These were based on the idea that we can neatly separate a systematic time-varying part from a purely (usually Gaussian) white noise process with no temporal dependencies. This may, in principle, still be possible even in more complex, nonlinear cases, if, for instance, in the example above we had a good process model of the spiking behavior. However, often this may not at all be trivial, and in any case it implies that with time series one has to be much more cautious in applying conventional parametric assumptions than with i.i.d. data. Also recall that parametric time series tests usually require properties like stationarity and ergodicity which are not that easy to establish in practice. Often we have observed only one time series, not a sample of several series produced under identical conditions, and sometimes these series are even quite short (which is always problematic for asymptotic statistics), like for instance time series from fMRI. Thus, parametric significance levels obtained in the time series context may sometimes only be taken as a guidance rather than for strict hypothesis testing (cf. Chatfield 2004).

One particular complication for the methods presented in this chapter is that in the real world, and in the brain in particular, the processes underlying the observed time series will rarely be linear (see next chapter). It is to be stressed, however, that linear models often may still provide a good approximation, especially if the noise is large and the linear part dominates the dynamics, or could still provide useful information about salient features of the series. It may also be possible to remove strongly nonlinear features (e.g., cutting out spikes), or linear models may just serve as null hypothesis reference. That is, there may be situations where we still may want to go ahead with linear models although the residuals may violate Gaussian white noise assumptions to some degree.

For checking significance, bootstrap and permutation methods as introduced in Ch. 1 offer an alternative that may circumvent some of the problems of parametric tests. In any case it is recommended to back up parametric inferences drawn from time series by bootstrapping methods. As laid out in sect. 1.5.3, there are both parametric as well as nonparametric forms of the bootstrap, and we will cover the former first (see Davison & Hinkley, 1997, for a more extensive introduction).

The parametric bootstrap or permutation test is usually based on the residuals from a fitted model. Say we are dealing with an AR(p) model

$$(7.98)\quad x_t = a_0 + \sum_{i=1}^{p} a_i x_{t-i} + \varepsilon_t \,,\ \varepsilon_t \sim N(0,\sigma^2)\,,$$

for which we have obtained parameter estimates $\{a_i\}$ as outlined in sect. 7.2.1. Then we may estimate the variance σ^2 from the residuals as (Davison & Hinkley 1997)

$$(7.99)\quad \hat{\sigma}^2 = \frac{1}{T-2p-1}\sum_{t=p+1}^{T}\hat{\varepsilon}_t^2$$

where T-p is the length of time series available, and create parametric bootstrap samples $\{x_t^*\}$ by randomly drawing numbers $\varepsilon_t^* \sim N(0,\hat{\sigma}^2)$ and iterate process (7.98) forward in time based on these (Efron & Tibshirani 1993; Lütkepohl 2006). That is, we start with initial estimates for the first p observations $\{x_1^* ... x_p^*\}$, and then iteratively update x_t according to eq. 7.98. AR(p) model eq. 7.98 would then be re-fitted on each of these bootstrap samples to obtain, for instance, SE estimates for the parameters $\{a_i\}$. Or a reduced model may be fitted using the estimated residuals for a formal hypothesis test.

A crucial point that makes this procedure different from parametric bootstrapping in linear regression is that we have to iterate the process (7.98) through time: We cannot just randomly resample $\varepsilon_t^* \sim N(0,\hat{\sigma}^2)$ and add to the extracted systematic part $\hat{x}_t$, because this would destroy the temporal consistency of the model, i.e. the requirement $\varepsilon_t = x_t - \left(a_0 + \sum_{i=1}^{p} a_i x_{t-i}\right)$ in model eq. 7.98 (the bootstrap process would not follow anymore the estimated AR(*p*) model that we deem to underlie the observed data).

We may want to drop the Gaussian assumption altogether, since otherwise there seems only little advantage over parametric tests as offered in sect. 7.2.2 (although note that in the parametric bootstrap setting we *make* the residuals conform to a Gaussian, while with the asymptotic tests we assume they are). As we assumed μ_ε=0, we would start by centering the residuals, $\varepsilon' = \varepsilon - \text{avg}(\varepsilon)$ (Efron & Tibshirani 1993; Davison & Hinkley 1997). Retaining the white noise idea, i.e. the independence of residuals ε_t, E[$\varepsilon_{t'}\varepsilon_t$]=0 for $t' \neq t$, one may either just create *B* random permutations of $\mathrm{E} = \{\varepsilon'_0,...,\varepsilon'_T\}$, or draw from E *T* values with replacement as in the classical bootstrap. Based on these, process eq. 7.98 would then be iterated in time as described above.

If we would like to stick with a linear model for simplicity or convenience, but from inspecting the residuals already suspect that a linear model does not capture all the dependencies in the series, i.e. that also the assumption E[$\varepsilon_{t'}\varepsilon_t$]=0 is violated to some degree, we could switch to other bootstrap/ permutation strategies specific for time series (Efron & Tibshirani 1993; Davison & Hinkley 1997). One of these is the block permutation or block bootstrap. In a block permutation or block bootstrap, we divide the whole time series of length *T* into *K* blocks of size *M*, i.e. $T \leq K \times M$, and instead of permuting or bootstrapping individual ε_t, we permute or draw from whole blocks of *M* temporally consecutive ε_t values. That is, we randomly rearrange our *K* non-overlapping sets $\{\varepsilon'_t,...,\varepsilon'_{t+M-1}\}$ into a new time series (Fig. 7.10), or – with bootstrapping – draw *K* sets from these with replacement and concatenate them. The idea here is of course that these bootstraps retain the temporal dependency structure of the original series, and it follows that the block length *M* should be chosen large enough so that any inter-dependencies (or auto-correlations) have largely died out after *M* steps. In fact, we could select a proper *M* by inspecting the auto-correlation and/or auto-mutual-information function of the ε_t series, and could cut off when this falls below a certain threshold (e.g., when there is no significant deviation from zero anymore). On the other hand, however, the number of blocks *K* has to be large enough to allow for a sufficiently large number of distinct permutation or bootstrap samples. In the permutation case, there are a total of *K*! possibilities to arrange the blocks, while with bootstrapping we have K^K. This number should not be too small, perhaps >1000 if feasible, as a rough guideline, to avoid too large of a variance in the estimated *p*-value.

To move on to the completely non-parametric case, with block permutations/bootstraps of course we are also no longer bound to any model assumptions: We can simply dissect the original series {x_t} (rather than the residuals) into *K* blocks and shuffle them around, or draw from them with replacement. Finally we point out that with block permutations/bootstraps there are a number of issues and details that have been discussed in the literature, e.g. that the continuity of the original time series may be broken at the *K*-1 interim block edges, for which there are several strategies (see e.g. Davison & Hinkley 1997).

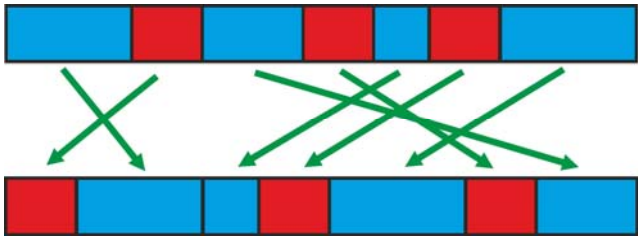

Fig. 7.10. With block permutation bootstraps whole blocks of consecutive values from the original time series (top) are randomly shuffled (bottom). If a hypothesis about properties of a time series in relation to experimenter-defined class labels is to be tested, e.g. as with neural recordings in a behavioral task with different stages ('blue' and 'red' task phases in the graph), one simple strategy is to shuffle blocks of consecutive class labels while leaving the original neural time series completely intact. Reprinted from Durstewitz & Balaguer-Ballester (2010) with permission.

Another popular bootstrapping idea for time series is *phase-randomization* which is based on the basic equivalence of the power spectrum and auto-correlation function of a time series (see sect. 7.1). The idea is to compute the Fourier transform (power spectrum) of the time series, scramble the phases associated with each frequency component, and transfer back to the time domain (Davison & Hinkley 1997; Schreiber & Schmitz 2000; Kantz & Schreiber 2004). By the Wiener-Khintchin theorem, this would also preserve the original auto-correlation function. Thus, importantly, these bootstrap data would be consistent with *any* stationary ARMA model that might have generated the data (up to the limitations imposed by the finite length and sampling rate of the observed process; Kantz & Schreiber 2004), as a stationary ARMA process is completely specified by the auto-correlation function through the Yule-Walker-equations (and a Gaussian noise process is completely specified by moments up to second order as well; see sect. 7.2). In other words, we do not even have to know or estimate the underlying ARMA model; phase randomization will preserve the linear-Gaussian model whatever it is (Kantz & Schreiber 2004). However, and importantly, phase randomization unlike block permutation, retains only the first and second moments of a time series (i.e., means and auto-covariances), which fully specify any stationary Gaussian process, but will destroy any nonlinear dependency (higher moments) in the time series. As shown later in sect. 8.1, this property of phase-randomized bootstraps has been used to check for nonlinear structure in time series.

Block permutations to control for auto-correlations (or, in fact, any higher-order dependencies) have been used widely in various situations with in-vivo electrophysiological recordings where the H_0 distribution of the test statistic was difficult to determine (e.g. Lapish et al. 2008; Balaguer-Ballester et al. 2011; see also Grün 2009). One specific, commonly employed form of block permutation bootstraps in neuroscience is the shuffling of whole (identical) trials to probe whether temporal relations among neurons bear information beyond the independent single unit activities: For each recorded unit i, one has a set of time series $\{x_{it}\}^{(k)}$, one for each distinct trial k, for which the assignments $\{x_{it}\} \rightarrow k$ are randomized. Importantly, this is done for each unit independently, resulting in bootstrap data sets $\{\mathbf{x}_t\}^{(k^*)}$ which presumably preserve the trial-specific auto-correlative structure and potential rate variations for each unit, but destroy the interrelations among them.

In doing so one (implicitly) assumes that the observed set of trials is *stationary*, i.e. that the different trials are identical in terms of single unit behavior (cf. def. 7.7 & 7.8). If this is not the case, e.g. if there are rate variations across trials common to all neurons, these are destroyed as well in the bootstrap data, and the inferences drawn from them

may no longer be valid. To avoid this, "*semi-parametric*" bootstraps may be constructed where for each neuron first a kernel density estimate of the spike density (instantaneous rate) is performed, e.g. using the methods from sect. 5.1.2, and then spike trains are redrawn at random from the estimated density (Fig. 7.11). Such bootstraps would preserve the rate variations and co-variations across trials and neurons, but destroy any finer temporal structure and relationships if present. A similar technique has also been called 'spike dithering' (Fujisawa et al. 2008; Louis et al. 2010).

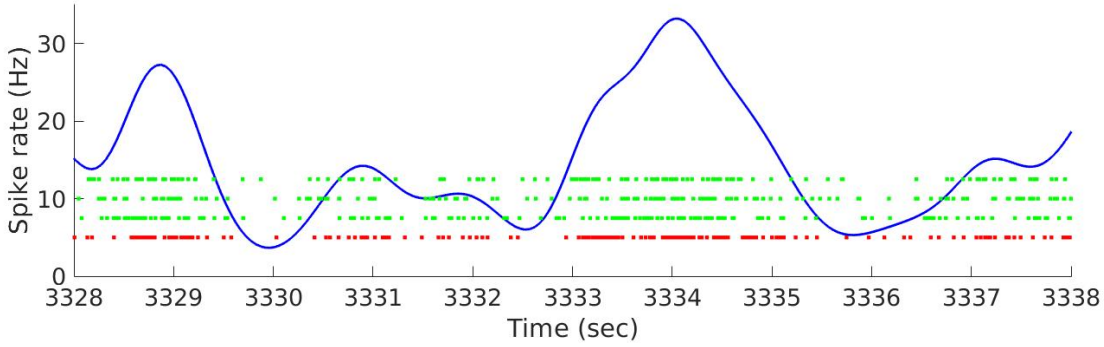

Fig. 7.11. Semi-parametric bootstraps (green dots) from a point process (red dots) preserving the original rate fluctuations as estimated through KDE (blue curve). **MATL7_9**.

Trial permutation bootstraps may also be employed to probe for coding of stimulus information in the neural activity, that is when trials are not identical but can be grouped according to the experimentally enforced stimulus conditions. In fact, in this case one may simply shuffle the class labels, that is the assignments $\{\mathbf{x}_t\} \rightarrow C$ (Fig. 7.10), if the interest is only in whether the recorded ensemble responses contain significant information about the stimulus (**MATL7_10**). One may shuffle single unit assignments within *identical* trials from the same class C in addition, if one would like to probe whether the temporal relations among units contribute to stimulus decoding, or whether the set of independent single unit activities is sufficient. More generally, even if one is not dealing with a set of discrete trials but rather continuous recordings during an extended task with different stimulus and behavioral events, one could shuffle consecutive blocks of class labels, i.e. blocks of consecutive time bins belonging to the same event class, to examine whether neural activity discriminates among the different task events (Fig. 7.10; Lapish et al. 2008). That way one would leave the temporal structure within the neural recordings completely untouched, but rather just scramble its relation to different task phases.

Phase-randomized bootstraps have also found various applications in neuroscience (Durstewitz & Gabriel 2007; Durstewitz et al. 2010), for instance for testing whether neural time series significantly deviate from linear dynamic model assumptions, e.g. whether they harbor predictability that only complies with a nonlinear process (see sect. 8.1).

Proper bootstrapping of time series is a highly important issue in neuroscience, as a famous discussion in Mokeichev et al. (2007) highlights. With reference to previous studies, these authors searched for recurring motifs in in-vivo recorded membrane potential traces, that is segments of V_m traces that appear to repeat with high similarity. They discovered some stunning examples of such repeats, sometimes even minutes apart. This appears to confirm that there are underlying microcircuits with strong intra-circuit connectivity that generate highly reproducible membrane potential trajectories once triggered (i.e., repeating membrane potential segments are taken as signatures of a fixed sequence of synaptic interactions and thus cell spikings). However, surprisingly, this apparently rich and stunning structure in the recorded membrane potential traces was completely reproduced in various forms of time series bootstraps, based on block permutations or model-generated voltage traces. Thus, what appeared like a sensational

discovery at first glance (building blocks of a neural language), may really be due to unspecific auto-correlative and deterministic structure generated by the membrane potential dynamics.