Chapter 9
Analysis of Network Flow Data

Flows are at the heart of the form and function of many networks, and understanding their behavior is often a goal of primary interest. Here we consider problems of statistical estimation and prediction arising in connection with various types of measurements relating to network flows.

9.1 Introduction

Many networks serve as conduits – either literally or figuratively – for *flows*, in the sense that they facilitate the movement of something, such as materials, people, or information. For example, transportation networks (e.g., of highways, railways, and airlines) support flows of commodities and people, communication networks allow for the flow of data, and networks of trade relations among nations reflect the flow of capital. We will generically refer to that of which a flow consists as *traffic*.

Much of the quantitative work in the literature on flows is concerned with various types of problems involving, for example, questions of network design, provisioning, and routing, and their solutions have involved primarily tools in optimization and algorithms. We will not address such problems here; see Phillips and Garcia-Diaz [313] for a general overview of this area. Instead, we will concern ourselves – broadly speaking – with the statistical modeling and analysis of measurements relating to network flows. More specifically, we will look at a handful of problems in which regression-like models can be used to describe the association between one set of flow-related variables and another, and regression-based inference and prediction can be used to infer unknown characteristics of the network flows.

Let $G = (V, E)$ be a network graph. Since flows have direction, from an origin to a destination, formally G will be a digraph. Following the convention in this literature, we will refer to the (directed) edges in this graph as links. Traffic typically passes over multiple links in moving between origin and destination vertices. The manner in which traffic moves throughout the network is captured by a matrix, say $\mathbf{B}$, called the *routing matrix*, defined to have the same number of rows and columns

E.D. Kolaczyk, *Statistical Analysis of Network Data*, Springer Series in Statistics,
DOI 10.1007/978-0-387-88146-1_9, © Springer Science+Business Media, LLC 2009

as links and origin-destination pairs in G, respectively. In the case that each origin-destination pair (i,j) has only a single route, from i to j, $\mathbf{B}$ is a binary matrix, with the entry in the row corresponding to link e and the column corresponding to pair (i,j) being

$$B_{e;ij} = \begin{cases} 1, & \text{if link } e \text{ is traversed in going from } i \text{ to } j \ , \\ 0, & \text{otherwise} \ . \end{cases} \tag{9.1}$$

That is, $\mathbf{B}$ is in this case a matrix describing the incidence of routes with links. If multiple routes are possible, the entries of $\mathbf{B}$ are instead fractions representing, for example, the proportion of traffic from i to j that is expected to use the link e. We will frequently find it convenient here in this chapter to assume, wherever it can aid in simplifying our presentation, that routes between origin-destination pairs are unique.

A quantity of fundamental interest in the study of network flows is the so-called *origin-destination* (OD) matrix, which we will denote by $\mathbf{Z} = [Z_{ij}]$, where Z_{ij} is the total volume of traffic flowing from an origin vertex i to a destination vertex j in a given period of time. The matrix $\mathbf{Z}$ is also sometimes referred to as the *traffic matrix*. A number of other quantities of interest relate directly to $\mathbf{Z}$. For example, the *net out-flow* and *net in-flow* corresponding to vertices i and j can be represented through the sums

$$Z_{i+} = \sum_j Z_{ij} \qquad \text{and} \qquad Z_{+j} = \sum_i Z_{ij} \ , \tag{9.2}$$

respectively. Similarly, defining X_e to be the total flow over a given link $e \in E$, and letting $\mathbf{X} = (X_e)_{e \in E}$, the link totals in $\mathbf{X}$ can be related to $\mathbf{Z}$ through the expression $\mathbf{X} = \mathbf{BZ}$, where $\mathbf{Z}$ now represents our traffic matrix written as a vector.

In addition to these various measures of traffic volume, the notion of a 'cost' c, usually associated either with paths or links, is also important. For example, the concept of *generalized cost* in a socioeconomic sense, is often invoked in transportation analysis, where it may be desired to understand and model the behavior of consumers of transportation resources, particularly the manner in which they make choices in their consumption. Similarly, the concept of *quality of service* (QoS) is important in the literature on computer network traffic analysis, where it is in the interests of both Internet network service providers and their consumers to, for example, monitor basic performance characteristics of the network, detect areas of congestion or anomalous behavior, and validate compliance with contractual service agreements.

Note that our notation above implicitly assumes a static snapshot of the flows in the network, a perspective that is convenient and adopted often in analyzing network flow data. But given the dynamic nature of flows, a time-varying perspective is generally more realistic and, depending on the context, may be more appropriate. Where relevant, we will signify this change by equipping the various quantities with a time index t. For example, a time-dependent traffic matrix will be denoted $\mathbf{Z}^{(t)}$. However, we will largely consider the routing matrix $\mathbf{B}$ to be fixed in the ma-

Section	Measurements	Analysis Goal
9.2	OD flow volumes Z_{ij}	Model observed flow volumes Z_{ij}
9.3	Link volumes X_e	Predict unobserved OD flow volumes Z_{ij}
9.4	OD costs c_{ij}	Predict unobserved OD and link costs

Table 9.1 Summary of topics covered in Chapter 9, broken down by type of measurements taken and the goal of the statistical analysis.

terial that follows. This assumption is common and is generally justified in contexts where changes in routing occur on longer time scales than those associated with the dynamics of the underlying flows.

The statistical problems we consider in this chapter are organized largely according to the types of measurements taken, and the goal of the statistical analysis in which they are to be used, as summarized in Table 9.1. For example, in some contexts, it is possible to observe the entire traffic matrix **Z**, and it is of interest to model these observations, both to obtain an understanding as to how potentially relevant factors (e.g., costs) affect flow volumes and to be able to make predictions of future flow volumes. A class of models commonly used for these purposes are the so-called gravity models, which we discuss in Section 9.2. In many contexts, however, it is difficult or impossible to observe the traffic matrix entries Z_{ij} directly, but comparatively easier to measure the link totals X_e. Recovery of the traffic matrix entries from these totals is then of central interest. We present a number of approaches to the problem of traffic matrix estimation in Section 9.3. Finally, there is the problem of modeling and inference of network cost parameters. In Section 9.4 we consider two related problems of this type, wherein it is assumed possible to obtain direct measurements of a cost c_{ij} for some subset of origin-destination pairs (i, j), and the goal is to infer characteristics of this cost more broadly across the network.

9.2 Gravity Models

Gravity models are a class of models, developed largely in the social sciences, for describing aggregate levels of interaction among the people of different populations. They have traditionally been used most in areas like geography, economics, and sociology, for example, but also have found application in other areas of the sciences, such as hydrology and the analysis of computer network traffic. Although our interest in gravity models in this chapter will be largely for their use in contexts where the relevant traffic flows are over a network of sorts, we note that it is not necessary for the formulation of these models that a network be defined explicitly, and our exposition in this section will reflect this fact accordingly. In our use of gravity models later in the chapter, however, the role of a network will be made explicit.

The term ‘gravity model’ derives from the fact that, in analogy to Newton’s law of universal gravitation, it is assumed that the interaction among two populations varies in direct proportion to their size, and inversely, with some measure of their separation. The concept goes back at least to the work of Carey [74] in the 1850's, but arguably was formulated in the strictest sense of the analogy by Stewart [368] in 1941. This area has been developed substantially in the past 50 years. Here we focus chiefly on providing an overview of a general version of the gravity model and corresponding methods of inference. For a comprehensive introduction to the topic, from the perspective of statistical modeling and inference, see the book by Sen and Smith [345], which itself contains references to numerous other works.

9.2.1 Model Specification

Suppose that $\mathscr{I}$ and $\mathscr{J}$ represent sets of origins and destinations, of cardinality $I = |\mathscr{I}|$ and $J = |\mathscr{J}|$, respectively, and Z_{ij} denotes – as defined in the introduction – a measure of the traffic flowing from $i \in \mathscr{I}$ to $j \in \mathscr{J}$ over a given period of time. The *general gravity model* specifies that the traffic flows Z_{ij} be in the form of counts, with independent Poisson distributions and mean functions of the form

$$\mathbb{E}(Z_{ij}) = h_O(i)\, h_D(j)\, h_S(\mathbf{c}_{ij}) \ , \tag{9.3}$$

where h_O, h_D, and h_S are positive functions, respectively, of the origin i, the destination j, and a vector $\mathbf{c}_{ij}$ of K so-called separation attributes. The K elements of $\mathbf{c}_{ij}$ are chosen to quantify some notion(s) of separation ascribed to the origin-destination pair (i, j), often based on concepts of 'distance' or 'cost' of either a literal or figurative nature. The functions h_O and h_D are sometimes referred to as the *origin* and *destination functions*, respectively, while h_S is commonly called the *separation* or *deterrence function*. Often the separation function is constrained to be non-increasing in the elements of $\mathbf{c}_{ij}$.

An early and now classical example of the gravity model is that of Stewart [368], proposed in connection with his theory of 'demographic gravitation,' which specifies that

$$\mathbb{E}(Z_{ij}) = \gamma\, \pi_{O,i}\, \pi_{D,j}\, d_{ij}^{-2} \ , \tag{9.4}$$

where $\pi_{O,i}$ and $\pi_{D,j}$ are measures of the origin and destination population sizes, respectively, for two geographical regions i and j, and d_{ij} is a measure of distance between the centers of these regions. This formulation is completely analogous to Newton's universal law, right down to the use of a 'demographic gravitational constant' γ. However, unlike Newton's law, neither empirical evidence nor theoretical arguments suggest that this form of the gravity model is strictly accurate in practical contexts.

Many proposals have been made for alternative forms of origin, destination, and separation functions. For example, power functions have been proposed for the origin and destination functions of the form

$$h_O(i) = (\pi_{O,i})^{\alpha} \quad \text{and} \quad h_D(j) = (\pi_{D,j})^{\beta} \ , \tag{9.5}$$

with exponents $\alpha, \beta \geq 0$. If more than one relevant measure of population size (or some other similarly predictive quantity) is to be used, it is common to generalize the formulas for $h_O(i)$ and $h_D(j)$ in (9.5) to products of such power functions. Alternatively, the values $\{h_O(i)\}_{i \in \mathscr{I}}$ and $\{h_D(j)\}_{j \in \mathscr{J}}$ are often simply treated as a collection of $I + J$ unknown parameters.

Power functions also have been proposed for the separation function, but in the form

$$h_S(c_{ij}) = (c_{ij})^{-\theta} \quad , \tag{9.6}$$

for scalar $c_{ij} \in \mathrm{I\!R}$, with $\theta \geq 0$, so as to force the function to be non-increasing in c_{ij}. However, arguably more popular is the use of exponential functions of the general form

$$h_S(\mathbf{c}_{ij}) = \exp\left(\theta^T \mathbf{c}_{ij}\right) \quad , \tag{9.7}$$

for vectors $\theta, \mathbf{c}_{ij} \in \mathrm{I\!R}^K$.

These choices of origin, destination, and separation functions generalize the model of Stewart in a natural manner. They can be motivated using various arguments, based on perspectives ranging from sociophysics to economic utility theory to maximum entropy. See Sen and Smith [345] and Fotheringham and O'Kelly [147], for example, for extensive treatments of such arguments. We note, however, that these choices are also convenient, in that on the logarithmic scale, the mean function in (9.3) becomes linear in the unknown parameters. For example, specifying power origin and destination functions and an exponential separation function, we have

$$\log \mathbb{E}\left(Z_{ij}\right) = \alpha \log \pi_{O,i} + \beta \log \pi_{D,j} + \theta^T \mathbf{c}_{ij} \quad . \tag{9.8}$$

This log-linear form facilitates the use of log-linear and linear regression methods for statistical inference on the model parameters, as we discuss below in Section 9.2.2.

To illustrate the preceding discussion, consider the following example.

Example 9.1 (Austrian Call Data). It is of interest to understand the spatial structure of telecommunication interactions among populations in different geographical regions. Such understanding is relevant to, for example, planning for government (de)regulation of the telecommunication sector and anticipating the influence of telecommunication technologies on regional development. It is natural to think of telecommunication patterns as flows, and gravity models are a standard tool for modeling these flows. The choice of relevant variables for constructing such models depends upon many factors, including the type of telecommunication and the level of spatial aggregation involved. Here we consider data on a network of inter-regional phone traffic (i.e., phone calls between geographical regions).

Fischer and Gopal [145] describe a set of data for phone traffic between 32 telecommunication districts in Austria throughout a period during the year 1991.

These data[1] consist of $32 \times 31 = 992$ flow measurements z_{ij}, $i \neq j = 1, \ldots, 32$, capturing contact intensity over the period studied, and are shown in Figure 9.1. Also shown are data quantifying the gross regional product (GRP) for each region, which may serve as a proxy for economic activity and income, both of which are relevant to business and private phone calls. In addition, data reflecting a notion of road-based distance between regions are shown too.

All data are shown on a logarithmic scale (using base 10, for ease of interpretation). It is evident that there is a reasonably strong relationship between call volume and the origin GRP, destination GRP, and distance. Moreover, this relationship is fairly linear (i.e., the dotted lines, produced by ordinary least-squares, lie close to the solid lines, which are the result of a nonparametric smoother) and is increasing in origin and destination GRP and decreasing in distance. These observations suggest that a model of the form

$$\mathbb{E}(Z_{ij}) = \gamma \, (\pi_{O,i})^{\alpha} \, (\pi_{D,j})^{\beta} \, (c_{ij})^{-\theta} \tag{9.9}$$

might be reasonable, where $\pi_{O,i}$ is the GRP of origin i, $\pi_{D,j}$ is the GRP of destination j, and c_{ij} is the distance from origin i to destination j.

Note, however, that the flow volumes between regions vary dramatically in scale. Most origin-destination pairs exchanged a volume of traffic in the range of thousands to tens of thousands, but some had a volume in the hundreds of thousands, while a few had a volume of only a hundred or less. Such disparity in magnitude is not atypical of flow volume data and tends to make it difficult to achieve uniform accuracy across origin-destination pairs, as we will see later. □

An alternative parameterization under which gravity models frequently are formulated is in terms of the so-called *interaction probabilities*

$$f_{ij} = \mathbb{E}\left(Z_{ij}/Z_{++} \,|\, Z_{++} > 0\right) \quad , \tag{9.10}$$

where $Z_{++} = \sum_{i,j} Z_{ij}$. These ‘probabilities’ represent the expected relative frequency at which interactions are specifically ij-interactions. Under the general gravity model specification in (9.3), they can be expressed as

$$f_{ij} = \frac{h_O(i)\, h_D(j)\, h_S(\mathbf{c}_{ij})}{\sum_{i' \in \mathscr{I}, j' \in \mathscr{J}} h_O(i')\, h_D(j')\, h_S(\mathbf{c}_{i'j'})} \quad . \tag{9.11}$$

In addition to the general gravity model in (9.3), there are various other related forms of gravity models. For example, *destination gravity models* pertain to the counts Z_{ij} to all destinations j from a given origin i. This perspective corresponds to considering the so-called *conditional destination probabilities*

[1] The original data are in units of erlang (i.e., number of phone calls, including faxes, times the average length of the call divided by the duration of the measurement period). For the purpose of illustration, we have converted these data to units of counts, by scaling by the minimum measurement value and rounding, to enable the types of log-linear statistical analyses described below in Section 9.2.2.

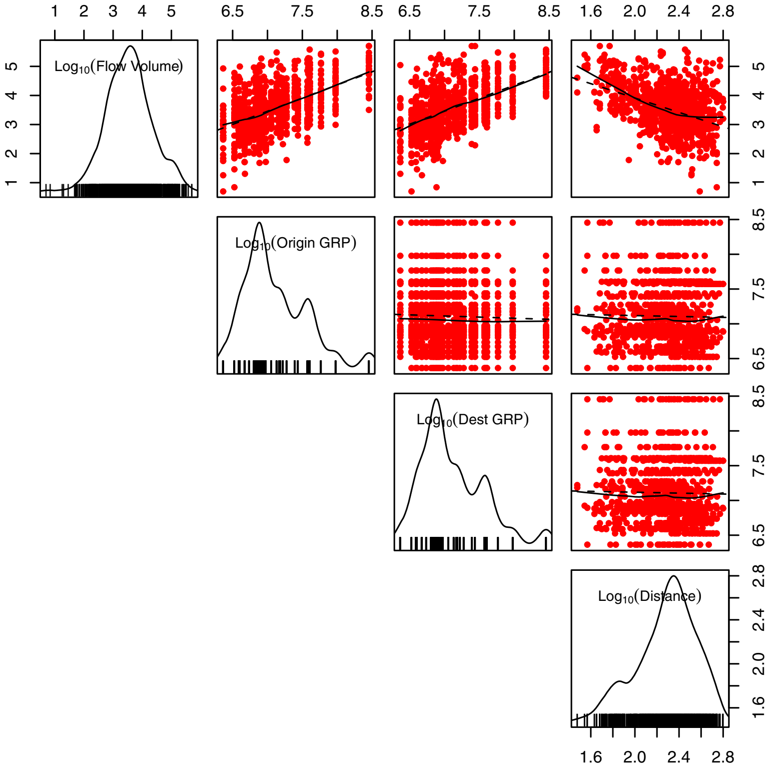

Fig. 9.1 Austrian call data. Scatterplots are shown for call flow volume versus each of origin GRP, destination GRP, and distance, along the top row, and for the latter three variables against each other, in the other rows. All axes are on log-log scales. Superimposed on each scatterplot are two lines, for descriptive purposes, showing fits based on simple linear regression (dotted) and a nonparametric smoother (solid). Density plots for each of the four variables are shown along the diagonal.

$$f_{j|i} = \frac{f_{ij}}{\sum_{j' \in \mathscr{J}} f_{ij'}} \;, \tag{9.12}$$

rather than the unconditional probabilities f_{ij}. In terms of the components in (9.3), these values can be expressed as

$$f_{j|i} = \frac{h_D(j)\, h_S(\mathbf{c}_{ij})}{\sum_{j' \in \mathscr{J}} h_D(j')\, h_S(\mathbf{c}_{ij'})} \;. \tag{9.13}$$

The set of *origin gravity models* can be specified analogously and pertain to *conditional origin probabilities* $f_{i|j}$, which can be expressed as in (9.13), but with $h_D(j)$ replaced by $h_O(i)$ in numerator and denominator.

For a comprehensive treatment of additional variations on the general gravity model, including versions that relax the assumption of independence between origin and destination effects implicit in the product form $h_O(i) \times h_D(j)$, see Sen and Smith [345].

9.2.2 Inference for Gravity Models

In light of the specification that the Z_{ij} be independent Poisson random variables with means $\propto_{ij} = \mathbb{E}(Z_{ij})$, statistical inference in the general gravity model is most naturally approached through likelihood-based methods. Our discussion here will focus on the model

$$\log \propto_{ij} = \alpha_i + \beta_j + \theta^T \mathbf{c}_{ij} \ , \tag{9.14}$$

where $\alpha_i = \log h_O(i)$, $\beta_j = \log h_D(j)$, and $\theta, \mathbf{c}_{ij} \in \mathrm{I\!R}^K$. Other cases, such as when the origin and destination functions are parameterized, as in (9.8), may be handled similarly. These models are log-linear models, which in turn are a specific instance of the class of generalized linear models. See, for example, McCullagh and Nelder [272, Ch. 6].

Let $\mathbf{Z} = \mathbf{z}$ be an $(IJ) \times 1$ vector of observations of the flows Z_{ij}, which for convenience are usually ordered by origin i, and by destination j within origin i. The relevant portion of the Poisson log-likelihood for $\propto$ takes the form

$$\ell(\propto) = \sum_{i,j \in \mathscr{I} \times \mathscr{J}} z_{ij} \log \propto_{ij} - \propto_{ij} \ . \tag{9.15}$$

Substituting the gravity model (9.14), taking partial derivatives with respect to the parameters α_i, β_j, and θ_k, setting the resulting equations equal to zero, and simplifying, it can be shown that the maximum likelihood estimates for these parameters must yield estimates $\hat{\propto}_{ij} = \hat{\alpha}_i \hat{\beta}_j \exp(\hat{\theta}^T \mathbf{c}_{ij})$, for $\propto_{ij}$ satisfying the equations

$$\hat{\propto}_{i+} = z_{i+}, \text{ for } i \in \mathscr{I} \quad \text{ and } \quad \hat{\propto}_{+j} = z_{+j}, \text{ for } j \in \mathscr{J} \tag{9.16}$$

$$\sum_{i,j \in \mathscr{I} \times \mathscr{J}} c_{ij;k} \hat{\propto}_{ij} = \sum_{i,j \in \mathscr{I} \times \mathscr{J}} c_{ij;k} z_{ij}, \text{ for } k = 1, \ldots, K \ , \tag{9.17}$$

where $\hat{\propto}_{i+} = \sum_{j \in \mathscr{J}} \propto_{ij}$ and $\hat{\propto}_{+j} = \sum_{i \in \mathscr{I}} \propto_{ij}$, and the z_{i+} and z_{+j} are defined similarly. Here $c_{ij;k}$ denotes the k-th element of $\mathbf{c}_{ij}$. Under mild conditions, these estimates will be well defined, and furthermore, the values $\hat{\theta}_k$ and $\hat{\propto}_{ij}$ will be unique. The values $\hat{\alpha}_i$ and $\hat{\beta}_j$ will be unique only up to a constant, due to the fact that the underlying model is over-parameterized by one degree of freedom.[2] Note that we are assuming here, without loss of generality, that origins i and destinations j for which $z_{i+} = 0$ or $z_{+j} = 0$ are dropped, as they contribute nothing to the analysis.

Various algorithms may be used to calculate the maximum likelihood estimates. Most straightforward is to use standard software for fitting log-linear models, usually available as an option in routines for fitting generalized linear models. These procedures are based on an iteratively re-weighted least-squares algorithm, like that used in logistic regression, which derives from application of the Newton-Raphson algorithm. See, for example, McCullagh and Nelder [272, Ch. 2.5] for details. Standard output includes parameter estimates and approximate standard errors, where the latter are driven by the usual arguments for asymptotic normality of maximum likelihood estimators. We illustrate through an analysis of the Austrian call data of Example 9.1.

Example 9.2 (Analysis of Austrian Call Data). We consider two models for these data. The first is that stated in (9.9), while the second is that stated in (9.14), but with the term $\theta^T \mathbf{c}_{ij}$ replaced by $-\theta \log(\text{Distance})$, where θ now is a non-negative scalar. We will refer to the former model as our standard gravity model and the latter as our general gravity model.

Both standard and general gravity models were fit using the generic iteratively re-weighted least-squares method for generalized linear models, having specified a Poisson model. Based on the standard asymptotic tests, all variables in both models were judged to be significant at the 0.05 level – in fact, the p-value for nearly every variable was beyond machine precision. The fit for the general gravity model had a much higher likelihood than that for the standard gravity model, which is not surprising, given that the former incorporates 64 variables, compared to four variables in the latter. However, the extent of this difference would appear to go well beyond that attributable simply to an increased number of parameters, judging by the fact that the Akaike information criterion (AIC) statistic[3] is reduced by almost 50% percent.

Since prediction of traffic volume is typically of most interest in this setting, we give in Figure 9.2 two pairs of plots characterizing the accuracy of traffic volume estimates under each model. The first pair of plots shows the fitted values $\hat{\propto}_{ij}$ versus

[2] More precisely, we can write (9.14) in the form $\log(\propto) = \mathbf{M}\gamma$, where $\mathbf{M}$ is an $(IJ) \times (I+J+K)$ matrix, and $\gamma = (\alpha_1, \ldots, \alpha_I, \beta_1, \ldots, \beta_J, \theta_1, \ldots, \theta_K)^T$ is an $(I+J+K) \times 1$ vector. The first $I+J$ columns of $\mathbf{M}$ are binary vectors, indicating the appropriate origin and destination for each entry of $\propto$, and are redundant in that both the first I and the next J sum to the unit vector. The last K columns correspond to the K variables defining the $\mathbf{c}_{ij}$. Assuming that the latter are linearly independent of themselves and of the former, the rank of $\mathbf{M}$ will be $(I+J-1)+K$. See Sen and Smith [345, Ch. 5.2].

[3] The AIC statistic for a likelihood-based model, with k-dimensional parameter η, is defined as $AIC = -2\ell(\hat{\eta}) + 2k$, where $\ell(\eta)$ is the log-likelihood evaluated at η, and $\hat{\eta}$ is the maximum likelihood estimate of η. This statistic, as with others of its type, provides an estimate of the generalization error associated with the fitted model, in this case effectively by off-setting the assessment of how well the model fits the data by a measure of its complexity. See, for example, Hastie, Tibshirani, and Friedman [194, Ch. 7.5] for additional details.

observed flow volumes z_{ij}. These plots are presented on a log-log scale, due to the large dynamic range of the values involved. The relationship between the two quantities is found to be fairly linear for both models, and the variation around their linear trend, fairly uniform. However, we note that there is evidence to suggest that the standard model tends to over-estimate in somewhat greater frequency than the general model, particularly for medium- and low-volume flows. The second pair of plots shows the relative errors $(z_{ij} - \hat{\propto}_{ij})/z_{ij}$ versus the flow volumes z_{ij}, again on a log-log scale. We see that for both models the relative error varies widely in magnitude. A large proportion of the flows are estimated with an error on the order of z_{ij} or less (i.e., the logarithm of relative error is less than zero), but a substantial number are estimated with an error on the order of up to ten times z_{ij}, and few others are even worse. In addition, we can see that, roughly speaking, the relative error decreases with volume. Finally, it is clear that for low volumes both models are inclined to over-estimate, while for higher volumes, they are increasingly inclined to under-estimate.

A concise relative comparison of the estimation accuracy of our standard and general gravity models can be obtained by plotting the empirical cumulative distribution functions of their (log) relative prediction errors together, as in Figure 9.3. As the CDF for the general model lies to the left of that for the standard model over most of the range of the relative error, it is clear that the former dominates the latter in accuracy. In fact, the general model produces estimates on the order of z_{ij} or less for 72% of the origin-destination pairs, as compared to 58% under the standard model. For both models, the average magnitude of the relative error for these pairs is just under 0.30. □

Various other algorithms have also been developed for maximum likelihood estimation in the general gravity model. A case of some particular interest is that where the functions $h_S(\mathbf{c}_{ij})$ in (9.3) are replaced simply by values $h_S(i,j)$, and the latter are fixed and known. Only the parameters α_i and β_j then remain to be estimated. The method of *iterative proportional fitting* (IPF), usually credited to Deming and Stephan [113] in the statistics literature, and a standard tool in the statistical analysis of contingency tables, can be used here. Specifically, under mild conditions, estimates of the form $\hat{\propto}_{ij} = \hat{\alpha}_i \hat{\beta}_j h_S(i,j)$ satisfying the constraints in (9.16) can be obtained through iteration of the expressions

$$\propto_{ij}^{(2m-1)} = \frac{\propto_{ij}^{(2m-2)} z_{i+}}{\propto_{i+}^{(2m-2)}} \tag{9.18}$$

$$\propto_{ij}^{(2m)} = \frac{\propto_{ij}^{(2m-1)} z_{+j}}{\propto_{+j}^{(2m-1)}} \ , \tag{9.19}$$

starting with some initial set of values $\propto_{ij}^{(0)} = \hat{\alpha}_i^{(0)} \hat{\beta}_j^{(0)} h_S(i,j)$, with $\hat{\alpha}_i^{(0)}, \hat{\beta}_j^{(0)} > 0$. See Sen and Smith [345, Ch. 5.3], for example, for additional background and justification in the context of gravity models. More general discussions can be found

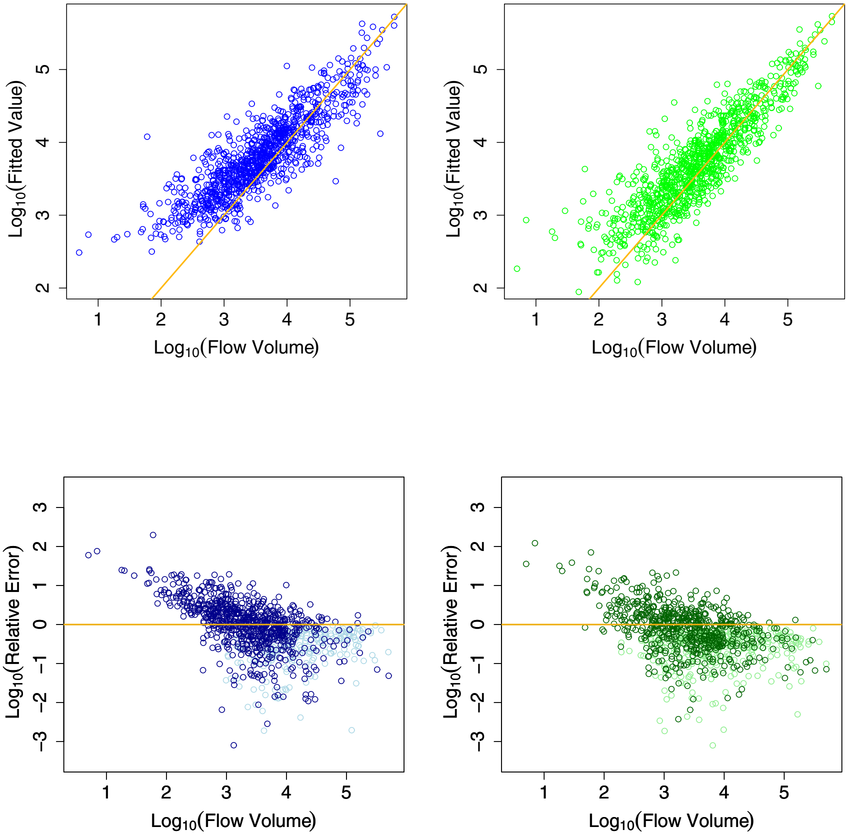

Fig. 9.2 Accuracy of estimates of traffic volume made by the standard (left, in blue) and general (right, in green) gravity models for the Austrian call data. Top: Fitted values versus flow volume. Bottom: Relative error versus flow volume, where light and dark points indicate under- and over-estimation, respectively. All axes are on logarithmic scales, base ten. The lines $y = x$ and $y = 0$ are shown in yellow in the top and bottom sets of plots, respectively, for reference.

in most texts on log-linear models and the analysis of categorical data (e.g., Christensen [83, Ch. 3.3]).

The IPF algorithm can also be used for solving other related problems. For example, in the gravity model in (9.14), given a value for θ, such as an estimate $\hat{\theta}$ from past data, and anticipated values for the marginal flow totals Z_{i+} and Z_{+j}, the prediction of future values Z_{ij} can be approached using this algorithm. Similarly, the algorithm can be used within a larger, hybrid algorithm for obtaining maximum likelihood estimates of the α_i, β_j, and θ_k in (9.14). For instance, we can iterate between estimation of the α_i and β_j, on the one hand, and of θ, on the other, by calling the IPF algorithm for estimating α_i and β_j and a Newton-Raphson algorithm

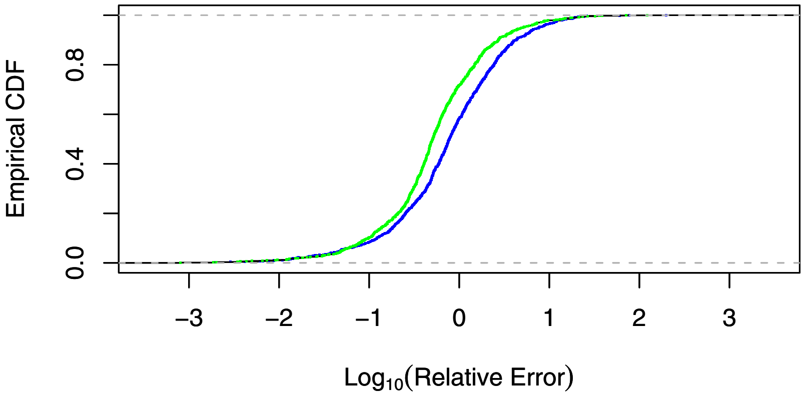

Fig. 9.3 Empirical CDF of the logarithm of the relative prediction error for the standard (blue) and general (green) gravity models in predicting call volume for the Austrian call data.

for estimating θ. Sen and Smith [345, Ch. 5] describe a number of algorithms of this nature. They also present evidence from a small numerical study that suggests such algorithms can be noticeably more computationally efficient than straightforward usage of standard generalized linear model fitting routines, particularly in problems with large I and J.

On a final note, it is also worth pointing out that procedures based on least squares can be used to obtain estimates for parameters in the general gravity model. Indeed, least-squares procedures were the norm for much of the history in this area, until the use of maximum-likelihood methods for generalized-linear models became computationally feasible and routine. Motivated by (9.3), least-squares procedures are generally based on models of the form

$$\log Z_{ij} \approx \alpha_i + \beta_j + \theta^T \mathbf{c}_{ij} + \varepsilon_{ij} \ , \tag{9.20}$$

where ε_{ij} is an error term. However, naive implementation of ordinary least-squares estimation is likely to give unsatisfactory results, particularly in settings where many of the means $\propto_{ij}$ are relatively small. Specifically, since $\mathbb{E}(\log Z_{ij}) \leq \log \propto_{ij}$, a bias can be expected in our estimates of the parameters. In addition, since $\mathbb{V}(\log Z_{ij})$ depends upon $\propto_{ij}$, the usual assumption of constant variance underlying ordinary least squares is violated, and accompanying confidence statements and tests are suspect.

Standard corrective techniques from linear modeling can go some ways towards alleviating these problems. For example, we might replace the variables $\log(Z_{ij})$ on the left-hand side of (9.20) by $\tilde{Z}_{ij} = \log(Z_{ij} + 1/2)$, since it can be shown that $\mathbb{E}(\tilde{Z}_{ij})$ and $\mathbb{V}(\tilde{Z}_{ij})$ are equal to $\log \propto_{ij}$ and $\propto_{ij}^{-1}$, respectively, up to terms of order $O(\propto_{ij}^{-2})$.

Application of a weighted least-squares procedure (e.g., Weisberg [399, Ch. 5.1]), with weights $w_{ij} \propto \propto_{ij}^{1/2}$, would then address the concerns about non-constant variance to a reasonable extent. Since these weights are not available, an iterative procedure can be used, starting for example with the observed $z_{ij}^{1/2}$ as weights and then using $\hat{\propto}_{ij}^{1/2}$ at the stages thereafter. Additionally, if our interest is focused primarily on the parameter θ, the overall dimension of the inference problem can be reduced by working not with the $\tilde{Z}_{ij}$, but rather with their centered versions $\tilde{Z}_{ij} - \tilde{Z}_{i\bullet} - \tilde{Z}_{\bullet j} + \tilde{Z}_{\bullet\bullet}$, where '$\bullet$' denotes averaging over the appropriate index.

Sen and Smith [345, Ch. 6] provide a detailed discussion of these topics. Least-squares techniques like these are still sometimes used for their simplicity. Nevertheless, all else being equal, it would seem preferable to use maximum likelihood techniques.

9.3 Traffic Matrix Estimation

Consider now the task of monitoring the flow volumes Z_{ij} on a network, with network graph G. In many types of networks, it is difficult – if not effectively impossible – to actually measure the Z_{ij}. For example, in monitoring road networks, various types of surveys are sometimes used (e.g., roadside surveys, home or destination surveys, etc.), but such devices can be highly inaccurate at small scales (e.g., due to small samples and sampling bias) and their large-scale usage is generally prohibitive in cost and effort. Furthermore, automated monitoring systems, based on collections of sensors, cameras, and the like, still face non-trivial technological hurdles prior to their successful large-scale deployment. On the other hand, in Internet networks carrying computer traffic, the network infrastructure itself can be equipped with the means to sample, store, and transmit (e.g., to a central location) such measurements. However, such capabilities are only relatively recent, and network service providers face some concern in enabling such measurements, in that they do wish to adversely affect the quality of service experienced by their customers.

Nevertheless, knowledge of the traffic flow volumes Z_{ij} is fundamental to a variety of network-oriented tasks. These include traffic management, network provisioning, and planning for network growth. Fortunately, in many of the same contexts in which measurement of the flow volumes Z_{ij} between origins and destinations is difficult, it is often relatively easy to measure the volumes X_e on network links. For example, in highway road networks, sensors may be positioned at the entrances to on- and off-ramps. Similarly, routers in an Internet network come equipped with the facility to monitor the data on incident links. In cases like these, we are then faced with a problem of predicting the Z_{ij} or, alternatively, estimating their means $\propto_{ij}$, from the observed link counts $\mathbf{X} = (X_e)_{e \in E}$.

This is the *traffic matrix estimation* problem. It has received a great deal of attention in the transportation sciences literature, going back at least to the mid-1970's, with seminal work by Robillard [327], van Zuylen and Willumsen [385], Bell [30], and Cascetta [76]. Also, more recently, it has received substantial attention in the literature on computer network traffic analysis. The problem appears to have been formulated in this latter context, and at the same time introduced to the statistics literature, by Vardi [386], who dubbed it 'network tomography.'

In any case of practical interest, the traffic matrix estimation problem will be highly under-constrained, in the sense that we effectively seek to invert the routing matrix $\mathbf{B}$ in the relation $\mathbf{X} \approx \mathbf{BZ}$, and $\mathbf{B}$ typically has many fewer rows (i.e., network links) than columns (i.e., origin-destination pairs). Various additional sources of information therefore typically are incorporated into the problem, which effectively serve to better constrain the set of possible solutions. Methods proposed in this area can be roughly categorized as *static* or *dynamic*, depending on whether they are aimed at estimating a traffic matrix for a single time period or successively over multiple time periods. We will discuss representative examples of both types.

Note that in what follows we assume that the routing matrix $\mathbf{B}$ is known. In reality, its construction is a separate task in itself, the details of which vary by context. In computer traffic networks this task often can be done, for example, by processing tables of local routing protocols gathered from nodes throughout the network. Some discussion may be found in Crovella and Krishnamurthy [105, Ch. 5.3.2]. In contrast, in the field of transportation studies, the entries of the routing matrix typically are based on models, often fit to data, of human behavior. Ortúzar and Willumsen [306] provide a detailed introduction.

9.3.1 Static Methods

The various static methods proposed for traffic matrix estimation are similar in that they all ultimately involve the optimization of some objective function, usually subject to certain constraints. However, they can differ widely in the construction of and justification for this objective function and, to a lesser extent, the constraints imposed. Here we discuss three classes of methods, deriving from principles of least squares and Gaussian models, from Poisson models, and from principles of entropy minimization.

9.3.1.1 Methods Based on Least-Squares and Gaussian Models

Although traffic typically is discussed in units of 'counts' (e.g., number of vehicles per hour or number of bytes per second), some of the earliest proposed methods were based on least-squares and Gaussian measurement models. Much insight into the traffic matrix estimation problem can be gained from the least-squares perspective, and the Gaussian measurement model that accompanies least squares (whether it be made explicit or not) is often not unreasonable in high-count settings. So we begin our discussion within this context.

A simple but commonly adopted model for the link counts $\mathbf{X}$ is one of the form

$$\mathbf{X} = \mathbf{B}\propto + \varepsilon \;, \tag{9.21}$$

where $\mathbf{X} = (X_e)_{e \in E}$ is an $N_e \times 1$ vector of link counts, $\mathbf{B}$ is an $N_e \times IJ$ routing matrix, as defined in (9.1), $\propto$ is an $IJ \times 1$ vector of expected flow volumes over all origin destination pairs $(i, j) \in \mathscr{I} \times \mathscr{J} \subseteq V^{(2)}$, and ε is an $N_e \times 1$ vector of errors, which we assume for the moment to be independent with mean zero and common variance σ^2.

In principle, this formulation suggests that $\propto$ be estimated through ordinary least squares. However, typically $N_e \ll IJ$, and so the least-squares problem is poorly posed. More precisely, the vector $\propto$ is generally not estimable, in the sense that there is no matrix $\mathbf{M}$ such that $\mathbb{E}(\mathbf{MX}) = \propto$. In fact, this issue can arise in even the smallest of systems, as illustrated by the following example.

Example 9.3 (Traffic Matrix Estimation on a Small Network (Robillard [327])). Figure 9.4 shows a simple network, consisting of five vertices and four links. There are two origin vertices (i.e., a and b), two destination vertices (i.e., c and d), and one intermediate vertex (i.e., v), resulting in the set $\{ac, ad, bc, bd\}$ of origin-destination pairs. In this case, the model (9.21) reduces to

$$\begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \propto_{ac} \\ \propto_{ad} \\ \propto_{bc} \\ \propto_{bd} \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \end{bmatrix} \;. \tag{9.22}$$

Although here $N_e = IJ = 4$, the rank of $\mathbf{B}$ is nevertheless only 3, and therefore $\mathbf{B}^T\mathbf{B}$ is not invertible. As a result, for observations $\mathbf{X} = \mathbf{x}$, the normal equations $(\mathbf{B}^T\mathbf{B})\propto = \mathbf{B}^T\mathbf{x}$ corresponding to the least-squares problem

$$\min_{\propto} \left(\mathbf{x} - \mathbf{B}\propto\right)^T \left(\mathbf{x} - \mathbf{B}\propto\right) \tag{9.23}$$

will generally have an infinite number of possible solutions $\hat{\propto}$. □

Although $\propto$ generally is not estimable in this model, under certain conditions the expected origin and destination volumes $\propto_{i+}$ and $\propto_{+j}$ are in fact estimable, and any solution $\hat{\propto}$ of the least-squares problem will produce the same estimates $\hat{\propto}_{i+}$ and $\hat{\propto}_{+j}$. Robillard [327] proposes using a gravity model for the expected flow volumes $\propto_{ij}$. Specifically, he proposes a model in the form $\log \propto_{ij} = \alpha_i + \beta_j + \log \gamma_{ij}$, for pre-specified gravity constants γ_{ij}, and suggests estimating the α_i and β_j under the constraints that $\propto_{i+} = \hat{\propto}_{i+}$ and $\propto_{+j} = \hat{\propto}_{+j}$, using a nonlinear least-squares algorithm. The resulting solution $\hat{\propto}$ will be unique if the values $\hat{\propto}_{i+}$, $\hat{\propto}_{+j}$, and γ_{ij} are all positive.

Unfortunately, it has been observed that in practice gravity models often fit too poorly to produce good estimates $\hat{\propto}$. However, in some situations we have available some initial – although perhaps rough and inaccurate – set of origin-destination flow volume measurements, say $\mathbf{Z}^{(0)} = \mathbf{z}^{(0)}$. In this case, we might use these measurements, rather than a gravity model, to suitably constrain our estimate $\hat{\propto}$. Cascetta [76] proposes a method for doing so based on generalized least-squares.

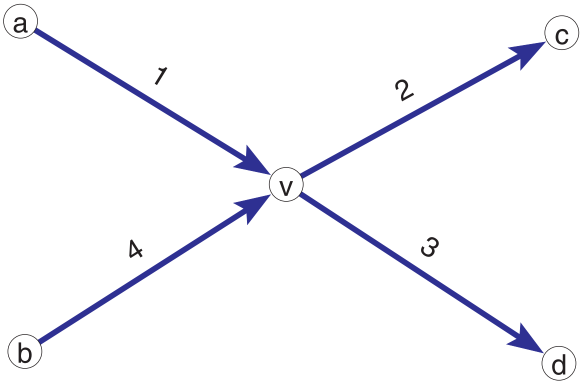

Fig. 9.4 A simple network illustrating the traffic matrix estimation problem.

Specifically, consider the model

$$\begin{bmatrix} \mathbf{Z}^{(0)} \\ \mathbf{X} \end{bmatrix} = \begin{bmatrix} \mathbf{I} \\ \mathbf{B} \end{bmatrix} \propto + \begin{bmatrix} \xi \\ \varepsilon \end{bmatrix} \quad , \tag{9.24}$$

where $\mathbf{I}$ is the $IJ \times IJ$ identity matrix, and ξ and ε are independent error vectors of dimension $IJ \times 1$ and $N_e \times 1$, respectively, with mean zero and covariance matrices Ψ and Σ, respectively. The solution to the corresponding generalized least-squares problem

$$\min_{\propto} \begin{bmatrix} \mathbf{z}^{(0)} - \propto \\ \mathbf{x} - \mathbf{B}\propto \end{bmatrix}^T \begin{bmatrix} \Psi^{-1} & 0 \\ 0 & \Sigma^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{z}^{(0)} - \propto \\ \mathbf{x} - \mathbf{B}\propto \end{bmatrix} \tag{9.25}$$

takes the form

$$\hat{\propto} = \left(\Psi^{-1} + \mathbf{B}^T \Sigma^{-1} \mathbf{B}\right)^{-1} \left(\Psi^{-1} \mathbf{z}^{(0)} + \mathbf{B}^T \Sigma^{-1} \mathbf{x}\right) \quad , \tag{9.26}$$

which is a linear combination of $\mathbf{z}^{(0)}$ and $\mathbf{x}$. Following standard linear model theory, under the model assumptions the estimator (9.26) is unbiased (i.e., $\mathbb{E}(\hat{\propto}) = \propto$) and its covariance $\mathbb{V}(\hat{\propto}) = \left(\Psi^{-1} + \mathbf{B}^T \Sigma^{-1} \mathbf{B}\right)^{-1}$ is a minimum among all unbiased estimators.

The covariance may be used to accompany the estimator $\hat{\propto}$ with some quantification of the uncertainty involved in estimating $\propto$, the exact nature of which is dictated through the error covariances Ψ and Σ. In practice, Σ is often taken to be a diagonal matrix and the form of Ψ will depend on the nature of the sampling underlying the measurements $\mathbf{z}^{(0)}$. Cascetta and Nguyen [77, Sec. 2.2] offer some discussion on this topic, for example. The values of the entries in Σ and Ψ are often set using historical data (i.e., earlier sets of measurements $\mathbf{z}^{(0)}$) or previous estimates $\hat{\propto}$.

Note that it is possible – and, indeed, likely, in the case of low link volume counts – for the estimated expected flow volumes $\hat{\propto}_{ij}$ in (9.26) to be negative. Cascetta [76], and later Bell [31] in more detail, derive an analogous generalized least-squares estimate of $\propto$ under the constraint that $\propto_{ij} \geq 0$, for all pairs i, j. The result of incorporating these constraints is effectively to perturb the estimate $\hat{\propto}$ by adding another term to the expression in (9.26). The contribution of this term is increasingly more pronounced the more active these constraints become.

Although no distributional assumptions are necessary for the generalized least-squares framework above, from a likelihood-based perspective a Gaussian noise model is implicit in its formulation. Maher [267], adopting such a perspective, has proposed a Bayesian approach to the estimation of $\propto$, based on the use of both a Gaussian measurement model and Gaussian prior. To illustrate, consider the measurement model in (9.21), and suppose we specify in addition that

$$\propto = \propto^{(0)} + \eta \quad , \tag{9.27}$$

where $\propto^{(0)}$ is an $IJ \times 1$ prior mean vector for $\propto$ and η is an $IJ \times 1$ random vector, with independent components of mean zero and prior variance τ^2. If it is further assumed that ε and η have independent multivariate Gaussian distributions, then it follows that the posterior distribution of $\propto$, given observations $\mathbf{X} = \mathbf{x}$, is itself multivariate Gaussian with mean

$$\mathbb{E}(\propto | \mathbf{X} = \mathbf{x}) = \propto^{(0)} + \mathbf{B}^T \left(\mathbf{B}\mathbf{B}^T + \lambda I\right)^{-1} \left(\mathbf{x} - \mathbf{B}\propto^{(0)}\right) \tag{9.28}$$

and covariance matrix

$$\mathbb{V}(\propto | \mathbf{X} = \mathbf{x}) = \tau^2 \left[I - \mathbf{B}^T \left(\mathbf{B}\mathbf{B}^T + \lambda I\right)^{-1} \mathbf{B}\right] \quad , \tag{9.29}$$

where $\lambda = \sigma^2/\tau^2$. Maher [267] derives the analogous expressions under an assumption of general covariances for ε and η. The form of the resulting posterior mean parallels that of the generalized least-squares estimator (9.26).

Note that the form of $\hat{\propto}$ in (9.28) is such that it may be interpreted as starting with the prior mean $\propto^{(0)}$ and then 'correcting' it through a term driven by the error in the 'prediction' of $\mathbf{x}$ by $\mathbf{B}\propto^{(0)}$. The covariance matrix in (9.29) can be used to fashion corresponding statements regarding the uncertainty of this estimate (e.g., point-wise posterior credible intervals for each $\propto_{ij}$). The value λ acts as a smoothing parameter. In the case that $\lambda \to 1$, the estimate $\hat{\propto}$ behaves so that $\mathbf{B}\hat{\propto} \to \mathbf{x}$ (i.e., an equality constraint with the observed link volumes is enforced). On the other hand, as $\lambda \to 0$, the estimate $\hat{\propto}$ tends to the prior mean $\propto^{(0)}$.

9.3.1.2 Methods Based on Poisson Models

As was noted above, the flow volumes Z_{ij} typically are discussed in units of counts. While a Gaussian measurement model may be reasonable in high-count settings, it will likely not be appropriate if even just some of the expected flow volumes $\propto_{ij}$ are low. In this latter case, a Poisson model can be more appropriate.

Specifically, consider modeling the distribution of $\mathbf{Z}$, given $\propto$, as a product of independent Poisson random variables i.e.,

$$\mathbb{P}\left(\mathbf{Z}=\mathbf{z}\,|\,\propto\right)=\prod_{ij}\mathbb{P}\left(Z_{ij}=z_{ij}\,|\,\propto_{ij}\right)=\prod_{ij}\frac{\exp\left(-\propto_{ij}\right)\propto_{ij}^{z_{ij}}}{z_{ij}!} \quad . \tag{9.30}$$

Furthermore, suppose that $\mathbf{X}=\mathbf{B}\mathbf{Z}$, indicating that no additional measurement error is incurred in observing the link volumes. The distribution of $\mathbf{X}$ is therefore induced directly by that of $\mathbf{Z}$, and in general the elements of $\mathbf{X}$ will not be independent. Note too that there is, of course, the same (usually substantial) reduction in dimension from $\mathbf{Z}$ to $\mathbf{X}$ as was observed at the beginning of the previous section on least squares methods, so that typically there are many values of $\mathbf{z}$ that might solve the equations $\mathbf{x}=\mathbf{B}\mathbf{z}$, given observed link counts $\mathbf{X}=\mathbf{x}$.

It is perhaps somewhat surprising then that distinct values of $\propto$ will nevertheless yield distinct distributions for $\mathbf{X}$ under fairly general conditions on $\mathbf{B}$. Specifically, Vardi [386] shows that if the columns of $\mathbf{B}$ are all distinct, and each column has at least one non-zero element, then $\mathbb{P}\left(\mathbf{X}=\mathbf{x}\,|\,\propto\right)=\mathbb{P}\left(\mathbf{X}=\mathbf{x}\,|\,\tilde{\propto}\right)$ implies that $\tilde{\propto}$ differs from $\propto$ by at most a simple rescaling.[4] In principle, it would therefore seem that we can proceed in a standard manner with maximum likelihood estimation of the relative magnitudes of the $\propto_{ij}$. However, potential difficulties still remain, as illustrated by the following toy example.

Example 9.4 (Difficulties with Poisson Maximum Likelihood (Vardi [386])). Recall the toy network of Example 9.3, depicted in Figure 9.4, and consider just that subnetwork consisting of the three vertices $\{a,v,c\}$ and the two links 1 and 2. We will consider this network as having a total of three origin-destination pairs $\{av,vc,ac\}$. Its routing matrix is of the form

$$\mathbf{B}=\begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 1\end{bmatrix} \quad . \tag{9.31}$$

Assume model (9.30) and suppose that we observe the link volumes $\mathbf{x}=(x_1,x_2)^T=(1,2)^T$. There are only two possible values for the corresponding flow volumes in this case: $\mathbf{z}=(0,1,1)^T$ and $\mathbf{z}=(1,2,0)^T$. The likelihood of the data can therefore be expressed as

$$\mathscr{L}\left(\propto\right)=\mathbb{P}\left(\mathbf{X}=(1,2)^T\right)$$

[4] That is, under these conditions, the parameter vector $\propto$ is identifiable. See also Singhal and Michailidis [354].

$$= \mathbb{P}\left(\mathbf{Z} = (0,1,1)^T\right) + \mathbb{P}\left(\mathbf{Z} = (1,2,0)^T\right)$$
$$= \left(\propto_{ac}\propto_{vc} + \propto_{av}\propto_{vc}^2/2\right)\exp\left(-\propto_{ac} - \propto_{av} - \propto_{vc}\right) \quad . \tag{9.32}$$

The maximum likelihood estimate $\hat{\propto}$ of $\propto$ will be the solution to the problem

$$\max_{\propto \geq 0} \ \ell\left(\propto\right) \quad , \tag{9.33}$$

where $\ell(\propto) = \log \mathscr{L}(\propto)$ is the log-likelihood function. Straightforward calculations show that the likelihood score equations $\partial \ell(\propto)/\partial \propto = 0$ have a unique solution $\propto = (1,2,0)^T$. But, in fact, it can be shown that the value of $\propto$ that maximizes the likelihood in (9.32) is instead $\propto = (0,1,1)^T$. So the maximum likelihood solution differs from that which we would obtain through standard techniques, based on the likelihood score function. □

The seeming paradox in this example illustrates the difficulty, under the specified Poisson model, of maximum likelihood estimation in the traffic estimation problem. This difficulty arises due to the possibility of solutions to the score equations at the boundary of the parameter space (i.e., where some subset of the elements of $\propto$ take on the value zero). Vardi further shows that the log-likelihood $\ell(\propto)$ is, in general, not even necessarily concave. He argues, however, that if there are multiple independent and identically distributed samples $\mathbf{x}_1, \ldots, \mathbf{x}_n$, the likelihood is asymptotically concave in the limit of large n, assuming all elements of $\propto$ are strictly positive. Approximating this likelihood by that of a multivariate Gaussian, with mean $\mathbf{B}\propto$ and covariance $\mathbf{B}M\mathbf{B}^T$, where $M = \mathsf{diag}(\propto)$, he proposes a method-of-moments estimator, where these quantities are set equal to their empirical counterparts. But some care must be exercised in solving the resulting equations for $\hat{\propto}$, due to the possibility of algebraic inconsistencies. See Vardi [386, Sec. 3–4] for discussion and some proposed solutions.

Alternatively, the Poisson likelihood can be coupled with a prior distribution on $\propto$, and inference on $\propto$ and $\mathbf{Z}$ can be pursued through Bayesian methods. Similar to the Gaussian setting, incorporating an appropriately designed prior can have the effect of regularizing the traffic matrix estimation problem in the Poisson setting, which can lead to more stable solutions. In addition, the use of prior information – whether implicitly or explicitly – is almost obligatory when only a single sample $\mathbf{X} = \mathbf{x}$ is available.

Tebaldi and West [377] explore the Bayesian approach in some detail, where their goal is to conduct inference based on the joint posterior distribution $\mathbb{P}(\mathbf{Z}, \propto \,|\, \mathbf{X})$. In particular, they concentrate on the problem of simulating from this posterior using Markov chain Monte Carlo methods. Note that under the choice of independent priors on the components of $\propto$, $\mathbb{P}(\propto) = \prod_{ij} \mathbb{P}(\propto_{ij})$, and we have

$$\mathbb{P}(\propto \,|\, \mathbf{X}, \mathbf{Z}) = \mathbb{P}(\propto \,|\, \mathbf{Z}) = \prod_{ij} \mathbb{P}(\propto_{ij} \,|\, Z_{ij}) \quad . \tag{9.34}$$

Therefore, given $\mathbf{Z}$, it should be reasonably straightforward to simulate values of $\propto$. So consider the conditional posterior distribution $\mathbb{P}(\mathbf{Z} \,|\, \propto, \mathbf{X})$. Since the system of equations $\mathbf{X} = \mathbf{BZ}$ serves as a set of linear constraints on $\mathbf{Z}$, given $\mathbf{X} = \mathbf{x}$, these constraints must be incorporated explicitly when simulating from this posterior. The following example serves to illustrate this point.

Example 9.5 (Constraints in the Conditional Posterior (Tebaldi and West [377])). Recall the three-vertex network underlying Example 9.4, with routing matrix $\mathbf{B}$ given in (9.31). Since $X_1 = Z_{av} + Z_{ac}$ and $X_2 = Z_{vc} + Z_{ac}$, it is clear that knowledge of $Z_{ac} = z_{ac}$, in addition to $\mathbf{X} = \mathbf{x}$, is sufficient to recover z_{av} and z_{vc}. Furthermore, under the assumption of independent prior distributions on the elements of $\propto$, $\mathbb{P}(\propto_{av}, \propto_{vc}, \propto_{ac}) = \mathbb{P}(\propto_{av})\mathbb{P}(\propto_{vc})\mathbb{P}(\propto_{ac})$, and the marginal posterior distribution of Z_{ac} can be expressed in the form[5]

$$\begin{aligned}\mathbb{P}(Z_{ac} = z_{ac}\,|\,\propto, \mathbf{X} = \mathbf{x}) \;\propto\; & \mathbb{P}(Z_{ac} = z_{ac}\,|\,\propto_{ac}) \times \mathbb{P}(Z_{av}\,|\,\propto_{av})\, I_{x_1 - z_{ac}}(Z_{av}) \\ & \times \mathbb{P}(Z_{vc}\,|\,\propto_{vc})\, I_{x_2 - z_{ac}}(Z_{vc}) \ ,\end{aligned} \tag{9.35}$$

where $I_s(r)$ is the indicator that $r = s$, for fixed s.

As a result of these two facts, it follows that

$$\mathbb{P}(Z_{ac} = z_{ac}\,|\,\propto, \mathbf{X} = \mathbf{x}) \propto \frac{\propto_{ac}^{z_{ac}}}{z_{ac}!}\,\frac{\propto_{av}^{x_1 - z_{ac}}}{(x_1 - z_{ac})!}\,\frac{\propto_{vc}^{x_2 - z_{ac}}}{(x_2 - z_{ac})!} \ , \tag{9.36}$$

for $z_{ac} \in \{0, 1, \ldots, \min(x_1, x_2)\}$. Therefore, we can simulate from the full joint conditional posterior $\mathbb{P}(\mathbf{Z}\,|\,\propto, \mathbf{X} = \mathbf{x})$ in this problem by first drawing $Z_{ac} = z_{ac}$, according to (9.36) and then setting $z_{av} = x_1 - z_{ac}$ and $z_{vc} = x_2 - z_{ac}$. □

More generally, if $\mathbf{B}$ is of full rank N_e, we can re-order its columns so that $\mathbf{B} = [\mathbf{B}_1, \mathbf{B}_2]$, where $\mathbf{B}_1$ is a $N_e \times N_e$ non-singular matrix, and we can similarly write $\mathbf{Z}^T = (\mathbf{Z}_1^T, \mathbf{Z}_2^T)$, where $\mathbf{Z}_1 = \mathbf{B}_1^{-1}(\mathbf{X} - \mathbf{B}_2\mathbf{Z}_2)$. It then follows that the conditional posterior has the form

$$\mathbb{P}(\mathbf{Z} = \mathbf{z}\,|\,\propto, \mathbf{X} = \mathbf{x}) = \mathbb{P}(\mathbf{Z}_1 = \mathbf{z}_1\,|\,\mathbf{Z}_2 = \mathbf{z}_2, \propto, \mathbf{X} = \mathbf{x})\,\mathbb{P}(\mathbf{Z}_2 = \mathbf{z}_2\,|\,\propto, \mathbf{X} = \mathbf{x}) \ , \tag{9.37}$$

where the distribution $\mathbb{P}(\mathbf{Z}_1 = \mathbf{z}_1\,|\,\mathbf{Z}_2 = \mathbf{z}_2, \propto, \mathbf{X} = \mathbf{x})$ is concentrated at $\mathbf{z}_1 = \mathbf{B}_1^{-1}(\mathbf{x} - bB_2\mathbf{z}_2)$ and

$$\mathbb{P}(\mathbf{Z}_2 = \mathbf{z}_2\,|\,\propto, \mathbf{X} = \mathbf{x}) \propto \prod_{ij} \frac{\propto_{ij}^{z_{ij}}}{z_{ij}!} \ . \tag{9.38}$$

Compare this expression to (9.36).

Tebaldi and West [377, Sec. 3] describe how to sample from the distribution in (9.38) using Gibbs sampling,[6] with particular attention given to designing efficient

[5] Technically, this expression should also distinguish between the random variable $\propto$ and a realization of that random variable, analogous to the manner in which we distinguish between $\mathbf{Z}$ and $\mathbf{X}$, on the one hand, and $\mathbf{z}$ and $\mathbf{x}$, on the other. We suppress this detail here and immediately below, however, as it is cumbersome and the role of $\propto$ is secondary.

[6] See the discussion in Section 8.3.2.2 and the references therein.

sampling strategies. Inference for the individual elements in $\mathbf{Z}$ or $\propto$ then follows from the resulting Monte Carlo approximations to the appropriate marginal distributions $\mathbb{P}(Z_{ij}\,|\,\mathbf{X})$ and $\mathbb{P}(\propto_{ij}\,|\,\mathbf{X})$. In applications, these authors choose to use prior distributions $\mathbb{P}(\propto_{ij})$ that either are uniform on some interval $[0,m]$ or that follow a gamma distribution. Although simple and convenient, these choices appear to yield reasonable results in the examples studied.

Additional, earlier work on Bayesian approaches in the Poisson setting is surveyed in Cascetta and Nguyen [77], where some alternative choices of prior distributions are discussed and where it is sketched how inference on $\propto$ may be carried out in some cases through direct numerical optimization of the posterior $\mathbb{P}(\propto\,|\,\mathbf{X})$.

9.3.1.3 Methods Based on Entropy Minimization

Proposed in a somewhat different vein than the methods of the two classes above are a handful of methods based on the minimization of a particular form of entropy. Let $\propto$ be the vector of expected origin-destination flow volumes and $\propto^{(0)}$, a prior 'guess' for $\propto$, and suppose that the two have been normalized so that $\propto_{++} = \propto^{(0)}_{++}$. The relative entropy 'distance' between $\propto$ and $\propto^{(0)}$ is given by

$$D(\propto||\propto^{(0)}) = \sum_{ij} \frac{\propto_{ij}}{\propto_{++}} \log \frac{\propto_{ij}}{\propto^{(0)}_{ij}} \quad . \tag{9.39}$$

This quantity is also known as the Kullback-Liebler divergence and summarizes how similar the two 'distributions' $\{\propto_{ij}/\propto_{++}\}$ and $\{\propto^{(0)}_{ij}/\propto^{(0)}_{++}\}$ are to each other. It is always non-negative and will be equal to zero if and only if $\propto = \propto^{(0)}$. See, for example, Cover and Thomas [101, Ch. 2.3].

Entropy-based methods seek to minimize the expression in (9.39), subject to certain constraints that encourage fidelity to the observed data (i.e., $\mathbf{x} \approx \mathbf{B}\propto$). For example, an early proposal in this area, due to van Zuylen and Willumsen [385], effectively seeks to minimize (9.39) subject to the constraints $X_e = \sum_{ij} B_{e;ij}\propto_{ij}$, for $e \in E$. Formally, through the method of Lagrange multipliers, this problem is equivalent to the problem

$$\min_{\propto,\lambda_1,\ldots,\lambda_{N_e}} D\left(\propto||\propto^{(0)}\right) + \sum_{e=1}^{N_e} \lambda_e \left(\sum_{ij} B_{e;ij}\propto_{ij} - x_e\right) \quad , \tag{9.40}$$

where $\lambda_1,\ldots,\lambda_{N_e}$ are the multipliers introduced to enforce the N_e constraints.

Differentiating the expression in (9.40) with respect to the $\propto_{ij}$ and setting the result to zero, we find that for fixed $\lambda = (\lambda_1,\ldots,\lambda_{N_e})$, the elements of the resulting estimator have the form

$$\hat{\propto}_{ij}(\lambda) = \propto^{(0)}_{ij} \exp\left(-1 - \sum_{e\in E} \lambda_e B_{e;ij}\right) \quad . \tag{9.41}$$

So the solution $\hat{\propto}$ is a multiplicative perturbation of the initial $\propto^{(0)}$. In practice, the values in λ are found using an iterative optimization algorithm starting, say, from $\lambda = \mathbf{1}$. The value $\propto^{(0)}$ is generally specified based on previous estimates or initial measurements $\mathbf{z}^{(0)}$. Note that a non-negative solution $\hat{\propto}$ is guaranteed if $\propto^{(0)} \geq 0$.

More recently, Zhang, Roughan, Lund, and Donoho [414] have instead proposed to optimize the relative entropy (9.39) subject to the constraint $(\mathbf{x}-\mathbf{B}\propto)^T(\mathbf{x}-\mathbf{B}\propto) \leq C$, for some constant $C > 0$, and $\propto \geq 0$. That is, the constraints of van Zuylen and Willumsen, with their strict equality, are relaxed to a single constraint on the overall accuracy of the prediction of $\mathbf{x}$ by $\mathbf{B}\propto$. Formally, this optimization is equivalent to the problem

$$\min_{\propto \geq 0} \, (\mathbf{x}-\mathbf{B}\propto)^T(\mathbf{x}-\mathbf{B}\propto) + \lambda D\left(\propto || \propto^{(0)}\right) \quad . \tag{9.42}$$

The expression in (9.42) is seen to be a penalized least-squares problem, like that in (2.39) of Chapter 2. Methods of this type are common in the literature on ill-posed inverse problems, where they are commonly referred to as *maximum entropy regularization*. Alternatively, interpreted from a Bayesian perspective, the estimator (9.42) is simply a maximum *a posteriori* estimate of $\propto$, under the assumption of the Gaussian measurement model in (9.21) and a prior $f(\propto)$, where $\log f(\propto)$ behaves like $D\left(\propto || \propto^{(0)}\right)$. This particular choice of prior can be viewed as an approximation to a multinomial prior for $\propto$, with probabilities proportional to the values $\propto_{ij}^{(0)}$. See Cascetta and Nguyen [77, Sec. 2.3].

As the relative entropy is convex in its argument $\propto$, the overall optimization is a convex optimization problem, for which existing software may be used, as Zhang, Roughan, Lund, and Donoho [414] discuss. These authors implement a particular version of this estimation strategy, under the additional specification that $\propto_{ij}^{(0)}$ follows a certain gravity model. We illustrate below, in the case study of Section 9.3.3, with a simple gravity model of the form $\propto_{ij}^{(0)} \propto \propto_{i+}^{(0)} \propto_{+j}^{(0)}$, where $\propto_{i+}^{(0)}$ and $\propto_{+j}^{(0)}$ are origin- and destination-specific net out-flow and in-flow volumes, respectively. In practice, there is also the issue of choosing the smoothing parameter λ in (9.42). Cross-validation, for example, may be used for this purpose. Zhang et al., however, report that their results are relatively insensitive to the value of λ and suggest that such careful tuning may be unnecessary.

9.3.2 Dynamic Methods

Suppose now that we are interested in recovering traffic matrices $\mathbf{Z}$, or their means $\propto$, over a sequence of consecutive time periods $t = 1, \ldots, \tau$. Specifically, given vectors $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(\tau)}$ of observed link counts, we wish to infer the corresponding traffic matrices $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(\tau)}$ or their means $\propto^{(1)}, \ldots, \propto^{(\tau)}$, where $\mathbf{x}^{(t)} \approx \mathbf{B}^{(t)}\mathbf{z}^{(t)}$, for possibly time-dependent routing matrices $\mathbf{B}^{(t)}$. Dynamic methods of traffic matrix estimation are designed for this purpose and may be categorized roughly according to whether they produce inferences simultaneously at all time periods or sequentially. Simultaneous and sequential methods also are sometimes referred to as non-recursive and recursive, respectively. An overview of work in this area, from the field of transportation studies, may be found in Sherali and Park [349]. Similarly, Soule et al. [364] is a useful reference from the field of computer network traffic analysis.

The dynamic methods proposed to date seem to be predominantly based on principles of least squares. In addition, the majority of the methods require that the length of a typical trip time, from any given origin to any given destination, be substantially shorter than the length of each time interval t during which measurements were taken. This assumption has the advantage of simplifying the nature of the routing information that must be encoded in the routing matrices $\mathbf{B}^{(t)}$, since it effectively allows us to ignore the possibility that trips beginning in one time period actually end in a different time period. See Sherali and Park [349] for further discussion on this point.

Simultaneous methods of inference typically work from a model of the form $\mathbf{x}^{(t)} = \mathbf{B}^{(t)}\propto^{(t)} + \varepsilon^{(t)}$, in analogy to (9.21), and penalized least-squares criteria are often employed to estimate the values $\propto^{(t)}$. That is, we estimate the vector of means $\left(\propto^{(1)^T}, \ldots, \propto^{(\tau)^T}\right)^T$ as

$$\left(\hat{\propto}^{(1)^T}, \ldots, \hat{\propto}^{(\tau)^T}\right)^T = \arg\min \quad \sum_{t=1}^{\tau} \left(\mathbf{x}^{(t)} - \mathbf{B}^{(t)}\propto^{(t)}\right)^T \left(\mathbf{x}^{(t)} - \mathbf{B}^{(t)}\propto^{(t)}\right) + J\left[\left(\propto^{(1)^T}, \ldots, \propto^{(\tau)^T}\right)\right] \;, \tag{9.43}$$

where the minimization is over all values of the collection of τ vectors $\propto^{(t)}$, possibly with the constraint that they have only non-zero elements. The penalty $J(\cdot)$ is needed because, based on the squared-error loss alone, estimation of the full collection of means $\propto^{(t)}$ reduces to τ separate ordinary least-squares sub-problems, and individually each of these sub-problems will of course be under-determined, as in the static case. A simple but common specification of the penalty is as a sum $J = \sum_{t=1}^{\tau} J^{(t)}$, where the $J^{(t)}$ are, for example, based on independent Gaussian priors (9.27), or entropy-based priors (9.39), say with $\propto^{(0)}$ replaced by time-indexed analogues $\propto^{(t,0)}$. In this case, the optimization in (9.43) then reduces to τ separate static inferences. See Cascetta [75], for example.

Note that from this perspective, any temporal correlations in the measurements $\mathbf{x}^{(t)}$ are effectively ignored. If we impose the additional assumption that the means $\propto^{(t)}$ are constant (i.e., that $\propto^{(t)} \equiv \propto$, for some $\propto$), then the $\mathbf{x}^{(t)}$ may effectively be treated as replicates. Although this assumption is unlikely to be true when the length of the overall period of measurement is long, it can sometimes be a useful approximation over sufficiently short periods. Of course, in the least-squares setting, even with replicates the estimation of $\propto$ is not generally a well-posed problem. However, as was mentioned in Section 9.3.1.2, this estimation problem is well posed under the Poisson model, and for sufficiently large τ the method-of-moments estimator of Vardi [386] can be used. Similarly, Cao, Davis, Wiel, and Yu [73] show that the estimation problem can also be well posed when the $\mathbf{x}^{(t)}$ are modeled according to a multivariate Gaussian distribution, with mean $\propto$ and covariance $\mathbf{BMB}^T$, where $\mathbf{M} = \mathsf{diag}(\propto_{ij}^c)$, for some $c > 0$. Cao et al. propose a maximum likelihood estimator that can be interpreted as a weighted least-squares estimator, where the weights depend on the unknown elements $\propto_{ij}$.

Sequential methods of traffic matrix estimation differ from simultaneous methods in that, not only do they try to account for correlations in the measurements $\mathbf{x}^{(t)}$ over time, they in fact seek to exploit these correlations. Most frameworks of this type can be viewed as variations on or extensions of the concept of Kalman filtering, as described in, for example, Nihan and Davis [303] and Soule, Salamatian, Nucci, and Taft [365]. Our description here parallels that of the latter authors.

In the Kalman filtering approach, the time-varying relationship among the means $\propto^{(t)}$ and the link counts $\mathbf{X}^{(t)}$ is modeled through a set of equations

$$\propto^{(t+1)} = \Xi^{(t)} \propto^{(t)} + \eta^{(t)} \tag{9.44}$$

$$\mathbf{X}^{(t)} = \mathbf{B}^{(t)} \propto^{(t)} + \varepsilon^{(t)} \quad , \tag{9.45}$$

where the $\eta^{(t)}$ and the $\varepsilon^{(t)}$ are assumed to be zero-mean random vectors, with covariances $\Psi^{(t)}$ and $\Sigma^{(t)}$, respectively, and uncorrelated both with each other and among themselves across times t. The stochastic process defined in (9.45) is called the *observation process*, and that in (9.44), the *hidden process* or the *state process*. The matrices $\Xi^{(t)}$ govern the nature of the evolution of the hidden process.

The so-called 'Kalman filter' is a sequential, recursive algorithm for determining, at each time $t+1$, an optimal estimate of the state $\propto^{(t+1)}$, based on the observations $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(t+1)}$, where this estimate is optimal in the sense that it is unbiased and has minimum variance among all unbiased estimators.[7] Specifically, the Kalman filter algorithm alternates between making a prediction $\hat{\propto}^{t+1|t}$ of $\propto^{(t+1)}$, based on the first t observations, $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(t)}$, and then updating (or 'correcting') that prediction to a value $\hat{\propto}^{t+1|t+1}$, to adjust for the discrepancy between the observation $\mathbf{x}^{(t+1)}$ and its predicted value $\mathbf{B}^{(t+1)} \hat{\propto}^{t+1|t}$. This second value $\hat{\propto}^{t+1|t+1}$ is what we use to define our estimate $\hat{\propto}^{(t+1)}$.

In more detail, let $\mathbf{P}^{t|t} = \mathbb{E}\left(\hat{\propto}^{t|t} - \propto^{(t)}\right)\left(\hat{\propto}^{t|t} - \propto^{(t)}\right)^T$ be the covariance matrix associated with our estimation error at time t, and let $\hat{\propto}^{0|0} = \mathbb{E}(\propto^{(0)})$ and $\mathbf{P}^{0|0}$ be initial values. In the 'prediction step' we compute the prediction

$$\hat{\propto}^{t+1|t} = \Xi^{(t)} \hat{\propto}^{t|t} \tag{9.46}$$

and its variance

$$\mathbf{P}^{t+1|t} = \Xi^{(t)} \mathbf{P}^{t|t} \Xi^{(t)^T} + \Psi^{(t)} \quad . \tag{9.47}$$

In the 'estimation step' we then update this prediction, resulting in the estimate

[7] That is, Kalman filtering produces a minimum variance unbiased estimator (MVUE), under the model specified in (9.44) and (9.45).

$$\hat{\propto}^{t+1\,|\,t+1} = \hat{\propto}^{t+1\,|\,t} + \mathbf{K}^{(t+1)}\left(\mathbf{x}^{(t+1)} - \mathbf{B}^{(t+1)}\hat{\propto}^{t+1\,|\,t}\right) \quad , \tag{9.48}$$

where

$$\mathbf{K}^{(t+1)} = \mathbf{P}^{t+1\,|\,t}\mathbf{B}^{(t+1)^T}\left[\mathbf{B}^{(t)}\mathbf{P}^{t+1\,|\,t}\mathbf{B}^{(t+1)^T} + \Sigma^{(t+1)}\right]^{-1} \quad . \tag{9.49}$$

At this stage, the covariance matrix of our estimation error also is updated, as

$$\mathbf{P}^{t+1\,|\,t+1} = \left[\mathbf{I} - \mathbf{K}^{(t+1)}\mathbf{B}^{(t+1)}\right]\mathbf{P}^{t+1\,|\,t}\left[\mathbf{I} - \mathbf{K}^{(t+1)}\mathbf{B}^{(t+1)}\right]^T + \mathbf{K}^{(t+1)}\Sigma^{(t+1)}\mathbf{K}^{(t+1)^T} \quad . \tag{9.50}$$

A derivation of these equations may be found in most graduate texts on statistical signal processing. For a thorough introduction to the topic of Kalman filtering see, for example, Chui and Chen [86]. Note the similarity between equation (9.48) here, in the dynamic setting, and equation (9.28) previously, in the static setting. In both cases, an initial estimate of $\propto$ is updated, based on the quality of predictions generated by that estimate for an observed $\mathbf{X} = \mathbf{x}$. In the static case, the prediction often comes from previous studies or a low-level sampling of the underlying origin-destination flows $\mathbf{Z}$, whereas in the dynamic cases, the prediction is generated from the measurements that are preceding in time.

There are certain practical issues that must be addressed in implementing the strategy described above. Most importantly, the matrices $\Xi^{(t)}$, $\Psi^{(t)}$, and $\Sigma^{(t)}$ must be determined. Typically these matrices are assumed to be constants Ξ, Ψ, and Σ over some (perhaps short) length of time, and their values, estimated from measurements. The manner of estimation varies to some extent with both the particulars of the model and the type of data available. In general, however, given, say, observations $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(\tau)}$, some version of the expectation-maximization (EM) algorithm[8] is used, following the method introduced in Shumway and Stoffer [352]. A concise description of the relevant details of the algorithm, as well as additional related references, may be found in the short survey by Ghahramani and Hinton [171]. When a set of auxiliary measurements of the flow volumes $\mathbf{Z}$ can be obtained, as is sometimes possible, the task of estimation can be simplified, and use of the EM algorithm avoided, as we illustrate below in the case study of Section 9.3.3. Regardless of the manner in which estimates are generated, the algorithm periodically must be re-calibrated with new starting values and new estimates Ξ, Ψ, and Σ. In order to decide when to recalibrate, the error process $\mathbf{x}^{(t+1)} - \mathbf{B}^{(t+1)}\hat{\propto}^{(t+1)}$ can be monitored to see whether, for example, any component of it is found to exceed some pre-specified number of standard deviations during too many consecutive time intervals. The appropriate standard deviations are the square roots of the diagonal entries of the matrices $\mathbf{B}^{(t)}\mathbf{P}^{t+1\,|\,t}\mathbf{B}^{(t+1)^T} + \Sigma^{(t+1)}$.

[8] The EM algorithm is an iterative algorithm for maximizing a log-likelihood. It is designed to be used in contexts where it is difficult to work with the original likelihood, but where a more tractable likelihood can be obtained by ‘enlarging’ the data space, through appropriately defined latent variables. The algorithm iterates between an expectation (E) step and a maximization (M) step, and under appropriate conditions will converge to the maximum of the original likelihood. See, for example, Hastie, Tibshirani, and Friedman [194, Ch. 8.5] for an overview and additional references.

Other methods proposed for dynamic traffic matrix estimation in the spirit of Kalman filtering include the recursive least-squares estimators of Nihan and Davis [303] and the time-varying method of Cao, Davis, Wiel, and Yu [73].

9.3.3 Case Study: Internet Traffic Matrix Estimation

The network supporting Internet traffic can be thought of as a network of networks, a collection of so-called *autonomous systems* (ASes), each constituting a set of routers and communication links under the control of a single administrative entity. The latter, which are frequently Internet service providers (e.g., Sprint, AT&T, etc.), monitor their networks routinely for a whole host of purposes. For example, network operators need to be able to identify when a failure has occurred in the network (e.g., a link 'goes down'), the extent of that failure, and if possible, the reason for it. Similarly, they need to be able to track the traffic loads across the network, so that routing may be adjusted as needed, to minimize congestion, in an effort to optimize the quality of service provided. In addition, information on traffic loads is necessary at the engineering and management levels for purposes of capacity planning. Finally, in light of the ever-present threat from malicious traffic, such as is generated by denial of service attacks and worms, traffic monitoring has come to be a critical component of security measures put in place by providers.

The availability of traffic matrices $\mathbf{Z}^{(t)}$ is fundamental to being able to conduct these sorts of monitoring operations. Historically, it has been relatively easy to obtain measurements on the net out-flows $Z_{i+}^{(t)}$ and net in-flows $Z_{+i}^{(t)}$ associated with each vertex i in an Internet network. Traffic is sampled at the routers corresponding to each vertex and then, at regular intervals, relevant information on the sampled traffic volume is sent to a central location. The recovery of traffic matrices from these measurements can be approached as a traffic matrix estimation problem. More recently, it is now possible to obtain direct measurements of traffic flow volumes in Internet networks. However, this option is frequently not exercised by service providers, for various reasons, including concerns about the volume of data generated and its potential to adversely impact the quality of service experienced by users of the network. Rather, these measurements more commonly are used to calibrate and improve the design of methods for Internet traffic matrix estimation.

In order to illustrate the application of the traffic matrix estimation methods presented in this chapter, in the context of the Internet, we present an analysis of data constructed from measurements of traffic flows on a 'backbone' network, which is analogous in its role to that of a nation-wide system of highways. Specifically, starting with actual origin-destination traffic volume measurements, and a routing matrix corresponding to the period of measurement, we manufacture a set of link count pseudo-measurements. We then apply an entropy-based method and the method of Kalman filtering to these pseudo-measurements, as representative static and dynamic methods, respectively, and compare their performance in recovering the original traffic volume measurements. Our analysis is in the spirit of that in Soule et al. [364].

Our data are based on measurements of origin-destination flows on the Abilene network, taken continuously over a seven-day period, starting December 22, 2003. The relevant portion of the Abilene network, as it was at that time, may be conceptualized as a graph consisting of 11 vertices and 30 links (i.e., essentially a simplified version of that depicted in Figure 1.1). Each vertex corresponds to a ‘point-of-presence’ (PoP), encompassing a major metropolitan region, where traffic may enter or exit the network. The links represent systems of optical transportation technologies and routing devices, running from one PoP to another. There are two such links for every one of the 14 corresponding undirected edges shown in Figure 1.1. In addition, there are two other links associated with a connection between Kansas City and Sunnyvale that has since been discontinued (and hence is not shown on the more recent map in Figure 1.1). Data were obtained using the automated sampling protocols implemented on the Abilene routers, which resulted in the capture of roughly 1% of all packets.[9] The size of a packet is measured in bytes, and the volume of traffic between each origin-destination PoP pair was therefore measured in units of total bytes counted. The traffic flow counts were aggregated to five-minute intervals, a standard device used to avoid issues of time synchronization across the network. See Lakhina et al. [245] for additional details of this nature.

We therefore have flow measurements $\{\mathbf{z}^{(t)}\}_{t=1}^{\tau}$ for a total of $\tau = 12 \times 24 \times 7 = 2{,}016$ consecutive time periods, across $11 \times 10 = 110$ origin-destination pairs in the network. Obtaining a 30×110 routing matrix for Abilene during this period, we create pseudo-measurements in the form of link counts $\mathbf{x}^{(t)} = \mathbf{B}\mathbf{z}^{(t)}$, for $t = 1,\ldots,2{,}016$. Our goal will be to recover the flow volumes $\mathbf{z}^{(t)}$ from the link counts $\mathbf{x}^{(t)}$. To illustrate the nature of the challenge we face, in Figure 9.5 are shown the byte counts for the week (in units of kilobytes per second) over the link running from Denver to Sunnyvale. Also shown are the flow volume counts for the two flows that are routed over this link, carrying traffic from Denver to Sunnyvale and from Denver to Los Angeles. We clearly can see certain characteristics of the two individual traffic flows in the composite link data, but the aggregation of the individual flows in constructing the link data is sufficient to make the recovery of these flows impossible in the absence of any additional information.

In solving our traffic estimation problem, our choice of entropy-based method is a version of the ‘tomogravity’ approach proposed by Zhang, Roughan, Lund, and Donoho [414]. This method, already described briefly in Section 9.3.1.3, is so named in reference to the facts that it utilizes a gravity model prior and that traffic matrix estimation in this area is referred to as a ‘network tomography’ problem, following Vardi [386]. For each $\mathbf{x} = \mathbf{x}^{(t)}$, our estimates of the values in $\mathbf{z} = \mathbf{z}^{(t)}$ are obtained by solving the optimization

[9] Packets are the basic building blocks of Internet traffic, each consisting of a small ‘header,’ containing information like source and destination IP addresses, and a ‘payload,’ containing the data being transmitted, such as a portion of an email or a music download.

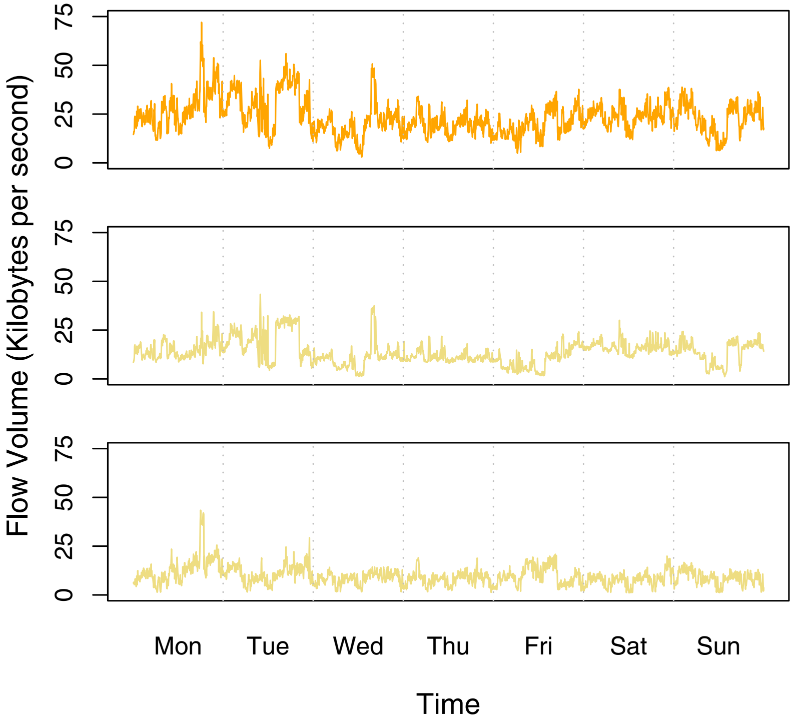

Fig. 9.5 Link counts (top, in orange) for the Denver to Sunnyvale link in the Abilene network, as compared to the origin-destination flow counts for the traffic passing over this link (in gold) from Denver to Sunnyvale (middle) and from Denver to Los Angeles (bottom).

$$\min_{\mathbf{z}\geq 0} \left(\mathbf{x}-\mathbf{B}\mathbf{z}\right)^T \left(\mathbf{x}-\mathbf{B}\mathbf{z}\right) + \lambda \sum_{i,j} \frac{z_{ij}}{z_{++}^{(0)}} \log\left(\frac{z_{ij}}{z_{ij}^{(0)}}\right) \quad , \tag{9.51}$$

where

$$z_{ij}^{(0)} = z_{i+}^{(0)} \times z_{+j}^{(0)} \tag{9.52}$$

is the product of the net out-flow and in-flow at vertices i and j, respectively, and $z_{++}^{(0)}$ is the total traffic in the network. The values $z_{i+}^{(0)}, z_{+j}^{(0)}$, and $z_{++}^{(0)}$ define what Zhang, Roughan, Lund, and Donoho [414] call a 'simple gravity model' prior,[10] and can be

[10] Zhang et al. actually propose and implement a more sophisticated gravity model prior, designed to better capture the effect on traffic volume of certain asymmetrical routing relationships. See also Erramilli, Crovella, and Taft [137].

obtained through appropriate summation of the elements in the vector $\mathbf{x}$ of observed link counts. The optimization in (9.51) was performed using the *maxent* package in **MATLAB**, produced by Per Christian Hansen.[11] Following the recommendation of Zhang, Roughan, Lund, and Donoho [414], a value of $\lambda = (0.01)^2$ was used, although like those authors, our results were found to be rather robust to this choice.

Our implementation of the Kalman filtering method largely parallels that of Soule et al. [365]. Here we specify a model of the form

$$\mathbf{Z}^{(t+1)} = \Xi \mathbf{Z}^{(t)} + \eta^{(t)} \tag{9.53}$$

$$\mathbf{X}^{(t)} = \mathbf{B}\mathbf{Z}^{(t)} \quad , \tag{9.54}$$

where the $\eta^{(t)}$ are assumed to be zero-mean random vectors, with common covariance Ψ. Note that the lack of an error term in the observation equation (9.54) is equivalent to using an error covariance matrix $\Sigma = 0$, in the notation of Section 9.3.2, for calculating our Kalman filter equations. The error term is omitted here due to the fact that no measurement error was injected during our construction of the pseudo-measurements $\mathbf{x}^{(t)} = \mathbf{B}\mathbf{z}^{(t)}$. The unknown parameters in our model are therefore just the two time-independent matrices Ξ and Ψ. These were estimated by reserving the first day of flow volume measurements, $\mathbf{z}^{(t)}$, $t = 1, \ldots, 288$, and fitting independent models of the form

$$Z_{ij}^{(t+1)} = \Xi_{ii} Z_{ij}^{(t)} + \eta_{ij}^{(t)} \quad , \tag{9.55}$$

separately for each pair (i, j), using standard techniques.[12] Such usage of preliminary measurements mimics the idea of a service provider periodically exercising its capability to measure flow volumes directly, but choosing to do so only sparingly. Predictions of the flow volumes for the remaining six days were obtained through application of the appropriate Kalman filter equations to the link volumes $\mathbf{x}^{(t)}$, for $t = 289, \ldots, 2,016$. Starting values were set using the mean and covariance of the first day flow volume measurements.

Note that our usage of these auxiliary measurements, to calibrate our model and initialize the prediction procedure, arguably gives the Kalman filtering method an unfair advantage over the tomogravity method. Soule et al. [364] have noted that tomogravity methods can suffer from a sizable estimation bias. Therefore, we also implemented a simple bias-corrected version of the tomogravity method described above, in which the original estimates were shifted uniformly in time by the amount of the average bias observed over the first day.

In Figure 9.6 we compare the performance of the tomogravity methods and the Kalman filtering method. Because the performance varies both as a function of time (i.e., temporally) and as a function of origin-destination flow (i.e., 'spatially') we have summarized the error associated with the methods in two ways. In order to compare temporal behavior, we have plotted the absolute error averaged over all

[11] Available at *http://www2.imm.dtu.dk/ pch/Regutools/.*

[12] More precisely, imposing appropriate conditions on the Ξ_{ii}, each origin-destination flow was modeled and fit as a first-order auto-regressive, i.e., AR(1), time series. See, for example, Brockwell and Davis [66] or similar texts.

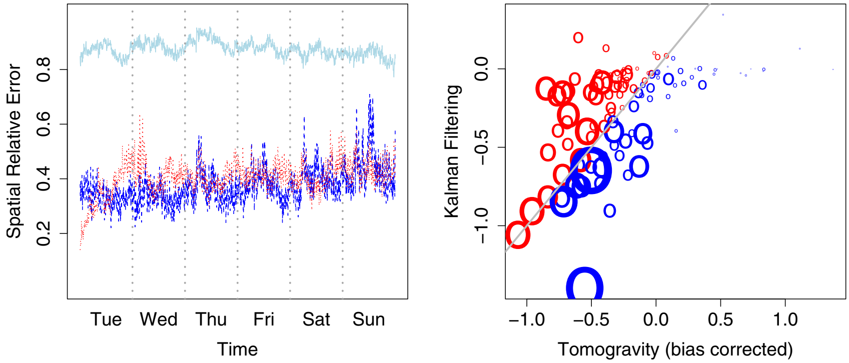

Fig. 9.6 Comparison of relative error (i.e., average absolute difference of predicted and actual origin-destination flow volumes, divided by average flow volume) for Abilene traffic flows. Left: Error aggregated over origin-destination pairs, as a function of time, for each of tomogravity (light blue), bias-corrected tomogravity (blue) and Kalman filtering (red). Right: Error aggregated over time, for each origin-destination pair, on a log-log scale, with area of symbol proportional to the mean volume of that flow, and colored according to which method had the larger relative error (tomogravity, blue; Kalman filtering, red).

origin-destination pairs, relative to the average flow volume, as a function of time. Similarly, in order to compare the spatial behavior, we have plotted absolute error averaged over time, again relative to the average flow volume, as a function of origin-destination. The temporal comparison shows that the tomogravity method, in its original form, consistently has a substantially greater relative error throughout the week. However, we can see too that with our simple bias correction, the two methods have comparable levels of error. Also evident from this plot is the fact that the Kalman filter performs its best at the start of the week, after which its performance degrades as time goes on. This behavior is not surprising, given the evolutionary nature of the algorithm, and could be ameliorated through the use of additional flow volume measurements to recalibrate the system (e.g., see Soule et al. [365]). In comparing the methods spatially, we see evidence that in general (but not always) the Kalman filtering method appears to do better than the (bias-corrected) tomogravity method at predicting the high- and low-volume traffic flows, but that the converse is true for moderate-volume flows.

Some representative examples are shown in Figure 9.7. Specifically, we show the predictions from the bias-corrected tomogravity and Kalman filtering methods for the traffic flows with the 2nd, 11th, 32nd, and 71st largest volumes, along with the flows themselves. The third of these corresponds to the Denver to Sunnyvale flow seen in the bottom of Figure 9.5. Predictions are plotted for the six days Tuesday through Sunday, omitting Monday. The performance is varied, with Kalman filtering outperforming tomogravity in the first and third cases (i.e., having relative errors of 0.0401 and 0.1247 versus 0.2806 and 0.4383, respectively), and vice versa, in the second and fourth cases (i.e., 0.7482 and 0.8464 versus 0.1413 and 0.5927, respectively). Perhaps more interestingly, however, we see that while the Kalman filtering method captures the underlying dynamics of the flow to a reasonable extent in all four cases, this is not so for the tomogravity method, which completely misses the dynamics in the first and third cases. In fact, this phenomenon was observed in a large number of the cases, across the 110 traffic flows, where tomogravity was outperformed by Kalman filtering.

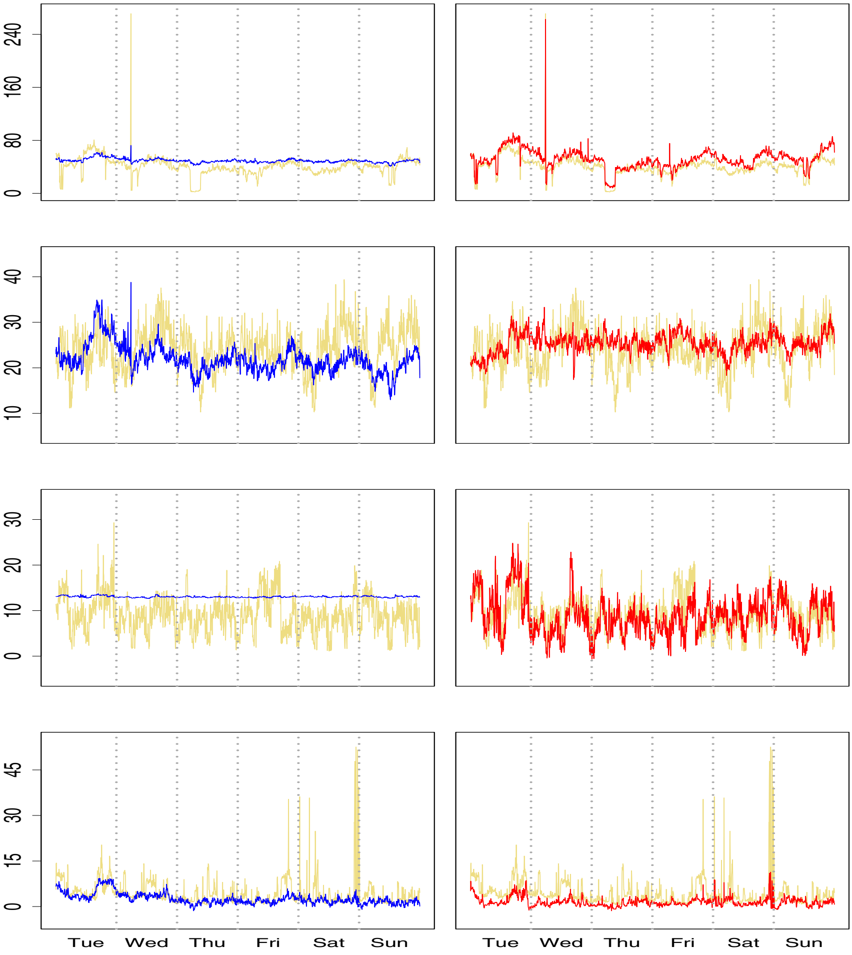

Fig. 9.7 Traffic flow volume predictions from bias-corrected tomogravity (left, in blue) and Kalman filtering (right, in red) methods, for four flows with volumes ranging from high (top) to low (bottom). Actual flow volumes are shown in yellow.

This distinction may be of greater or lesser importance depending on the application to which the estimated traffic matrices are to be put. For example, if the estimates are to be used as input to a system designed to detect anomalous traffic volumes (e.g., due to coordinated attacks on personal machines), it is important to correctly track the shape of the flow, since variations from typical conditions can be telling in this regard. On the other hand, for certain traffic engineering tasks (e.g., making routing changes in the networks to balance network load) or long-term planning for network capacity, it may be sufficient just to track the overall mean behavior successfully.

9.4 Estimation of Network Flow Costs

As was mentioned in Section 9.1, apart from flow volumes, the other major quantity of interest in transportation networks is – broadly speaking – cost. Depending on the context and the application, the term ‘cost’ might refer to a certain concrete quantity, such as the time, distance, or expense pertaining to a trip between locations in the network, or to a more abstract quantity, such as comfort. It might also be defined as a composite of a number of such individual quantities (i.e., as a so-called generalized cost).

Costs are usually discussed at two levels of network granularity: on paths and on links. A standard simplification relates the two through a linear model. For example, for simplicity, suppose that the paths routing between each origin-destination pair (i,j) are unique, so that the routing matrix $\mathbf{B}$ reduces to a binary matrix. Then we would write

$$\mathbf{c} = \mathbf{B}^T\mathbf{x} \ , \tag{9.56}$$

where $\mathbf{c}$ is an $IJ \times 1$ vector of path costs c_{ij} for travel from origins $i \in \mathscr{I}$ to destinations $j \in \mathscr{J}$, and $\mathbf{x}$ here is a $N_e \times 1$ vector of costs $\{x_e\}_{e \in E}$ for the links on our network graph $G = (V,E)$. The expression in (9.56) states simply that the cost associated with each path is the sum of the costs of the links traversed. Such linearity assumptions are often well justified in practice. For instance, the cost in gasoline used to drive on a road network between two geographical locations can be thought of in terms of the gasoline used on each of the intermediary links. Similarly, the time required for a commodity to be moved along a path from one end of a network to another – sometimes called the *delay* – can be broken down according the delays on the relevant links.

In this section, we will focus on a particular class of problems involving the inference of costs, where we assume that it is possible to directly measure path costs c_{ij} for some subset of origin-destination pairs (i,j) in the network. Such measurements are often called *end-to-end measurements.* We will discuss two related problems, with the goal in the first being to infer characteristics of the link costs $\mathbf{x}$, and in the second, characteristics of the full set of path costs $\mathbf{c}$. The former is an instance of what is often called *active network tomography*,[13] while the second has been referred to as *network kriging.*[14] In both problems, we will see that it is possible to exploit redundancies among the links traversed by the various paths to estimate the characteristics of interest. Such problems have been an area of major focus in the field of computer network traffic analysis, although they are in principle directly applicable to other types of transportation networks where the appropriate measurements can be made.

In fact, in the transportation sciences it is traditionally a somewhat different cost estimation problem that is encountered, within the context of the canonical *traffic assignment problem.* There, the goal is to infer (i.e., 'assign') the volume of traffic for all (non-unique) paths in the network, between all origin-destination pairs, using a traffic matrix $\mathbf{z}$ as input. But a set of link costs is also included in the formulation of the problem, and typically is part of the inferred output. These costs are related to $\mathbf{z}$ using particular functional forms deriving, for example, from consideration of the economic theory of supply and demand. A detailed discussion of this problem is beyond the scope of the book; we instead refer the reader to Ortúzar and Willumsen [306, Ch. 10] and Bell and Iida [32, Chs. 5–6].

9.4.1 Link Costs from End-to-end Measurements

Suppose that our interest is in characteristics of the link costs $\mathbf{x}$. In the transportation sciences, as we have just observed, link costs play a fundamental role in the traffic assignment problem. In computer network traffic analysis, they are basic indicators of the local level of performance of the network. Unfortunately, it is generally difficult to obtain measurements of link costs. On the other hand, it can sometimes be comparatively easier to measure the corresponding end-to-end costs on paths.

For example, consider the case where the cost of interest to us is delay (i.e., the time it takes to traverse from one point to another in the network). People generally do not keep track of the time it takes them, say, to move between points along the highway during their morning commute, but can likely report their total commute time on any given day. Alternatively, Internet service providers typically will not even know the topology of networks beyond their own (i.e., recall the case study in Section 7.4.5), much less be able to measure delays on links in those networks, and

[13] The adjective 'active' is used here in reference to the fact that measurements of the path costs c_{ij} are generated by actively injecting traffic into the network, while the term 'tomography' is meant to evoke the image of tomographic medical imaging (e.g., of the human brain) in seeking to infer 'internal' network characteristics from these measurements. The problem of computer network topology identification, described in the case study of Section 7.4.5, is another example of active network tomography. In contrast, the label *passive network tomography* is sometimes applied to the traffic matrix estimation problem of Section 9.3, since the measured link counts are of traffic already flowing in the network.

[14] The term 'kriging' is used in the geosciences in connection with methods of spatial interpolation that predict unknown values of a process from values observed at a subset of locations. See Cressie [102], for example, for an overview and additional references.

yet it is possible to obtain measurements of delays on paths from locations in their own network to others.

Formally, we can represent this situation by saying that we observe some subset $\mathbf{c}_s = (c_{i_1,j_1}, \ldots, c_{i_s,j_s})^T$ of s of the IJ costs in $\mathbf{c}$. Denoting the corresponding $N_e \times s$ sub-matrix of $\mathbf{B}$ by $\mathbf{B}_s$, in light of (9.56) we have $\mathbf{c}_s = \mathbf{B}_s^T\mathbf{x}$. Our goal is to infer some characteristic(s) of $\mathbf{x}$ from the measurements in $\mathbf{c}_s$.

Work in this area can be roughly organized according to four aspects: (i) the type of cost and characteristics of that cost that are of interest, (ii) the mechanism by which end-to-end cost measurements are taken, (iii) the topology of the sub-network defined by the s origin-destination pairs measured, and the links traversed by the paths between them, and (iv) the statistical inference algorithm used. Of course, the specific choice of each these four aspects is often highly inter-related. Useful surveys include Coates, Hero, Nowak, and Yu [96] and Castro et al. [78].

9.4.1.1 End-to-end Measurements

We illustrate the nature of end-to-end measurements in the context where delay is the cost of interest.

Example 9.6 (Multicast Measurements of End-to-end Delay). We have seen in Section 7.4 how it is possible in computer networks to send probes (i.e., small packets of information) from a given origin to various destinations. The ‘multicast’ probes described in Example 7.4 are one such example. There the measurements we discussed were binary measurements, indicating whether or not a probe from an origin arrived at each particular destination. However, it is also possible to measure the delay involved with the travel of packets from the origin to the destinations (with the delay taken to be infinite in the case where the packet does not arrive).

Recall that a multicast probe sends a copy of the same packet from an origin simultaneously to a set of destinations, and that this process induces a sub-network of our network graph G that takes the form of a tree $T = (V_T, E_T)$, with root vertex r and N_l leaf vertices $R \subset V_T$. This tree corresponds to the logical topology underlying the flow of traffic on G from r to the vertices in R. Figure 9.8 shows an illustration, with a tree consisting of one root, $N_l = 6$ leaves, and three internal vertices.

A multicast delay experiment consists of sending n such probes into the network, and recording the delays $\{c_{rj}^{(i)}\}_{i=1}^n$ for each pair (r, j), where $j = 1, \ldots, N_l$. The characteristic most commonly considered in this setting is the distribution of the link delays X_e, for each $e \in E_T$. It is standard to assume (after appropriate binning) a discrete form for this distribution, where the link delays are restricted to $b+2$ intervals indexed by the set $\{0, q, \ldots, bq, \infty\}$. Letting

$$\alpha_e(k) = \mathbb{P}\left(X_e = kq\right) \quad , \tag{9.57}$$

our goal can be stated as that of inferring the N_e sets $\{\alpha_e(k)\}_{k=0}^b$ from the delay measurements.

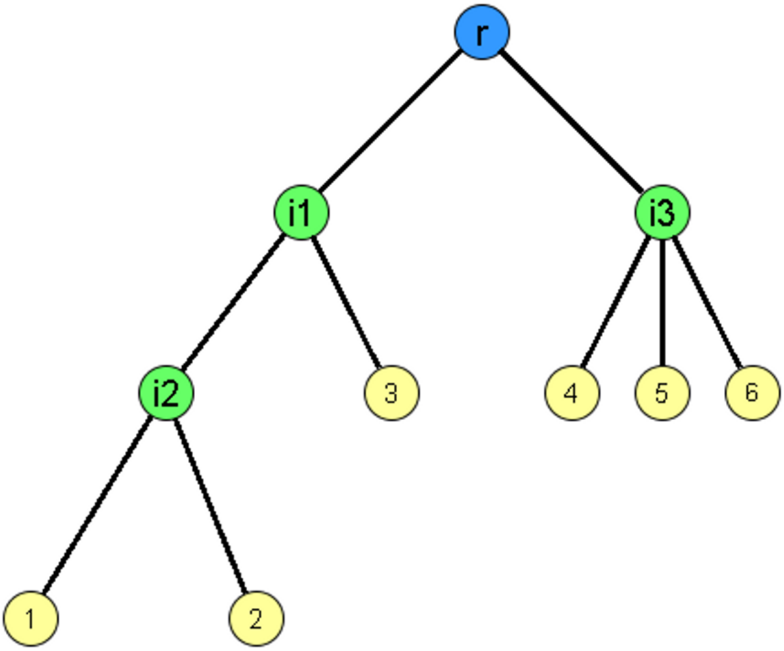

Fig. 9.8 Schematic illustration of a multicast tree for the link delay estimation problem.

In order to ensure identifiability of this inference problem, it is assumed that $\alpha_e(0) > 0$ for all $e \in E_T$, which means that for any given packet there is a positive probability on each link that it encounters ‘zero’ delay. Note that for this assumption to hold, it is generally necessary when defining the discrete delay distribution to compensate for some minimal level of delay common to traffic on all links. □

The standard reference for the multicast delay problem is Lo Presti, Duffield, Horowitz, and Towsley [265]. Citing the possibly prohibitive overhead associated with multicast in some settings, the use of other types of probes has also been proposed, with the primary examples being *unicast* probes (e.g., Tsang, Coates, and Nowak [381]) and *bicast* probes (e.g., Liang and Yu [260]). The bicast scheme basically involves sending multicast probes to individual pairs of destinations. The unicast scheme, on the other hand, involves sending pairs of closely spaced packets from the origin to two separate destinations (i.e., similar to the ‘sandwich’ probes described in the case study of Section 7.4.5). In both cases, each such pair of destinations ends up having a simple binary tree associated with it, consisting of the root r, two leaf vertices, and a single internal vertex. Lawrence, Michailidis, and Nair [250] have introduced a general framework that allows flexibility in combining various of these probing techniques, called *flexicast*. Sufficient conditions for identifiability are discussed. There has also been some work on the use of probes from multiple origins. See, for example, Rabbat, Coates, and Nowak [320].

The other type of end-to-end measurements on which substantial work has been done – again, in the area of computer network traffic analysis – is measurements on packet loss. Let p_{ij} be the probability a packet is lost along the path from i to j, and p_e, the probability that it is lost along edge e. Then, since a packet is lost along a path if and only if it is lost along one of the constituent links, under the assumption that packet loss is independent across links we can write $1 - p_{ij} = \prod_{e \in E_{ij}} (1 - p_e)$, where E_{ij} is the set of links defining the relevant path. Taking logarithms on both sides and combining the resulting equations over all origin-destination pairs (i, j), we obtain a linear system relating the log-rates $\log(1 - p_{ij})$ on paths to the log-rates $\log(1 - p_e)$ on links, analogous to that in (9.56). The structure of the problem of estimating the link loss rates p_e from packet loss measurements is closely related to the topology inference problem described in Section 7.4.1. See Caceres, Duffield, Horowitz, and Towsley [71].

9.4.1.2 Estimation of Link Cost Characteristics

Given path cost measurements $\mathbf{c}_s$ on a subset of s paths, our hopes for accurately estimating some characteristic(s) of the link costs $\mathbf{x}$ lie in the fact that in practice most links frequently are traversed by multiple paths. As a result of this sharing of links by paths, the path cost measurements $\mathbf{c}_s$ are correlated in an advantageous manner. In particular, the costs for two paths sharing many links should be more similar than for two paths sharing fewer links, and furthermore, the difference in their costs should be reflective of the difference in costs on the links *not* shared.

For example, in the tree shown in Figure 9.8, the links from the root r to the internal vertices i_1 and i_3 are each traversed by paths to three of the six leaves, and the link joining the internal vertices i_1 and i_2 is traversed by paths to two of the leaves. Consider the paths from r to the leaf vertices 1 and 2. Because they differ only in their last link, sharing as they do the links $\{r, i_1\}$ and $\{i_1, i_2\}$, the difference in their path costs c_{r1} and c_{r2} reflects the difference in delays on the links $\{i_2, 1\}$ and $\{i_2, 2\}$.

Methods of inference in this area exploit these observations in various manners. Consider again, for instance, the problem of multicast-based estimation of delay, as introduced in Example 9.6.

Example 9.7 (Multicast Estimation of Link Delay Distribution). Lo Presti, Duffield, Horowitz, and Towsley [265] propose a framework for estimating the delay distributions $\{\alpha_e(k)\}_{k=0}^{b}$, using multicast measurements, that is qualitatively quite similar to that proposed by Duffield, Horowitz, Lo Presti, and Towsley [128] for tomographic network topology inference, as described in Example 7.5.

For each vertex $j \in V_T \setminus R$, let

$$\gamma_j(k) = \mathbb{P}\left(\min_{j' \in R(j)} C_{rj'} \, \leq \, kq \right) \tag{9.58}$$

be the probability that the shortest delay on paths from r to the leaves in $R(j)$ is at most kq, where $R(j)$ is the set of all leaf vertices in R that are descendants of j. Given our n observations $\{\mathbf{c}_s^{(i)}\}_{i=1}^n$, a natural estimator for the value $\gamma_j(k)$ in (9.58) is

$$\hat{\gamma}_j(k) = \frac{1}{n}\sum_{i=1}^{n} I_{\{\min_{j' \in R(j)} c_{rj'}^{(i)} \leq kq\}} \ . \tag{9.59}$$

The probabilities in (9.58) are analogous to the probabilities in (7.41), and as a result, the former probabilities – like the latter – can be related to the unknown parameters we seek to estimate in terms of a certain function that is invertible under appropriate conditions. Substituting the estimates (9.59) for (9.58), in a manner similar to our substitution of the estimates (7.44) for (7.41), and inverting, yields a collection of estimated delay distributions $\{\hat{\alpha}_e(k)\}_{k=0}^{b}$, for each link $e \in E_T$. The process of inversion, in this case, involves two steps, with a system of equations much like those in (7.43) being solved first, and then the second roots of a set of polynomial equations being found. There is a polynomial for each vertex $j \in V_T \setminus R$, and its order is equal to the out-degree of j. The expressions for these various equations are somewhat involved and therefore omitted here. Details may be found in Lo Presti, Duffield, Horowitz, and Towsley [265]. □

The algorithm of Lo Presti et al. has the advantage of being fast and possessing good asymptotic properties. But a potential concern is the fact that in practice the polynomial second roots can sometimes be negative. An alternative is to use maximum likelihood methods but, unfortunately, the likelihood equations cannot be maximized in closed form. The expectation-maximization (EM) algorithm (i.e., see Footnote 8) can be implemented in a straightforward manner to maximize the likelihood, but tends to be extremely slow. Specifically, the algorithm scales like $O(b^{N_e})$, so that both increasing the number of bins b in our discretization of the delay distribution or increasing the network size N_e quickly becomes problematic. Generally we have some control over b, but little control over N_e. Some proposals have been put forth for speeding up the computations, yielding approximate maximum likelihood estimates, including the pseudo-likelihood method of Liang and Yu [260] and the ‘grafting’ method of Lawrence, Michailidis, and Nair [250].

Methods of estimation have also been proposed for the case where the cost of interest is the link loss rate. See, for example, Caceres, Duffield, Horowitz, and Towsley [71].

9.4.2 Path Costs from End-to-end Measurements

Suppose now that our main focus is on characteristics of some cost at the level of paths, rather than at the level of links. Although we are assuming an ability to measure this cost on paths, it is unusual in practice to be able to do so (or to want to do so) on *all* paths, since the number of origin-destination pairs IJ generally can be quite large (e.g., as large as $O(N_v^2)$ if all vertices can serve as both origins and destinations). Fortunately, it is possible to recover substantial cost information on unmeasured paths from that on measured paths, due to redundancies among paths, if those that are measured are chosen in an appropriate fashion.

9.4.2.1 Interpolation of Path Costs

Consider the expression $\mathbf{c} = \mathbf{B}^T\mathbf{x}$ in (9.56). If we knew $\mathbf{x}$, and of course $\mathbf{B}$, we could recover any and all of the path costs c_{ij} trivially. But as we just saw in Section 9.4.1, it is rare that we will know $\mathbf{x}$. However, since our real interest is in $\mathbf{c}$, from a purely linear algebraic perspective it is not necessary to know $\mathbf{x}$ itself, but rather only that portion of $\mathbf{x}$ that lies in the row space of $\mathbf{B}^T$ (i.e., the column space of $\mathbf{B}$). That is, recall that every such vector $\mathbf{x}$ can be decomposed as

$$\mathbf{x} = \mathbf{x}_{\mathsf{row}(\mathbf{B}^T)} + \mathbf{x}_{\mathsf{null}(\mathbf{B}^T)} \quad , \tag{9.60}$$

where the first vector in (9.60) lies in $\mathsf{row}(\mathbf{B}^T)$, the space of all vectors in $\mathbb{R}^{N_e}$ formed from linear combinations of the rows of $\mathbf{B}^T$, and the second vector, in $\mathsf{null}(\mathbf{B}^T)$, its orthogonal complement. Then, by definition,

$$\mathbf{c} = \mathbf{B}^T\mathbf{x} = \mathbf{B}^T\mathbf{x}_{\mathsf{row}(\mathbf{B}^T)} + \mathbf{B}^T\mathbf{x}_{\mathsf{null}(\mathbf{B}^T)} = \mathbf{B}^T\mathbf{x}_{\mathsf{row}(\mathbf{B}^T)} \quad . \tag{9.61}$$

Exploiting this fact, Chen, Bindel, Song, and Katz [80] describe a method for recovering all of the costs c_{ij} in $\mathbf{c}$ from measurements on just a subset of paths. More precisely, let $\mathbf{c}_s$ denote the cost measurements on a subset of s chosen paths, and let $\mathbf{c}_r$ be the unknown costs on the remaining $r = IJ - s$ paths, ordered such that $\mathbf{c} = (\mathbf{c}_s^T, \mathbf{c}_r^T)^T$. The method of Chen et al. allows us to recover $\mathbf{c}_r$ from $\mathbf{c}_s$, by first recovering from $\mathbf{c}_s$ that portion of $\mathbf{x}$ that lies in $\mathsf{row}(\mathbf{B}^T)$. As such, it is essentially a method of interpolation, and builds to some extent on earlier work of Shavitt, Sun, Wool, and Yenner [348] and of Nguyen and Thiran [300].

In more detail, let $\mathbf{B}_s$ be an $N_e \times s$ sub-matrix of $\mathbf{B}$, with $s = \mathsf{rank}(\mathbf{B})$, constructed so that each of its columns is independent. The selection of these s columns corresponds to a choice of s paths between s origin-destination pairs. The values in $\mathbf{c}_s$ are the cost measurements on these paths. Define $\mathbf{c}_r$ as above, and let $\mathbf{B}_r$ be that $N_e \times r$ sub-matrix of $\mathbf{B}$ such that $\mathbf{B} = [\mathbf{B}_s, \mathbf{B}_r]$. Consider the system of equations

$$\mathbf{c}_s = \mathbf{B}_s^T\mathbf{x}_{\mathsf{row}(\mathbf{B}^T)} \quad . \tag{9.62}$$

Generally, in practice $\mathsf{rank}(\mathbf{B}) \le N_e$, and so this system may be under-determined, in which case it will not have a unique solution. As is common in such situations, we choose from among all possible solutions that with the smallest possible length, which is unique and is obtained through the use of the Moore-Penrose pseudo-inverse. That is, we choose the solution $(\mathbf{B}_s^T)^{-}\mathbf{c}_s$ to (9.62), where $(\mathbf{B}_s^T)^{-} = \mathbf{B}_s(\mathbf{B}_s^T\mathbf{B}_s)^{-1}$. Combining this solution with (9.61), we recover the unknown path costs, as

$$\mathbf{c}_r = \mathbf{B}_r^T\mathbf{B}_s(\mathbf{B}_s^T\mathbf{B}_s)^{-1}\mathbf{c}_s \quad . \tag{9.63}$$

This approach to obtaining $\mathbf{c}$ can offer substantial savings in measurement overhead, compared to explicit measurement of cost on all paths. For instance, when every origin also acts as a destination, in which case $I = J$ and $IJ = I^2$, Chen, Bindel, Song, and Katz [80] conjecture that $\mathsf{rank}(\mathbf{B}) = O(I \log I)$ and present a combination of numerical and theoretical arguments in support of this suggestion. This quantity represents almost an order of magnitude reduction over the nominal number of measurements and is essentially driven by the amount of redundancy in the links traversed by the various paths, according to the routing matrix $\mathbf{B}$. The particular choice of s paths with independent columns in $\mathbf{B}$ that define $\mathbf{B}_s$ can be made using algorithms for computing various so-called 'rank-revealing decompositions,' such as Gaussian elimination with complete pivoting, QR decomposition with column pivoting, and the singular value decomposition. See Golub and van Loan [181]. The overall algorithm of Chen, Bindel, Song, and Katz [80] uses the QR decomposition for both path selection and calculation of the Moore-Penrose pseudo-inverse.

9.4.2.2 Prediction of Path Cost Characteristics

Although the interpolation method just described is appealing when the necessary number of measurements are obtainable, note that if even one less measurement is taken than is necessary, equation (9.63) does not hold and the subset of unknown path costs $\mathbf{c}_r$ cannot be recovered exactly. However, the basic recovery problem may be recast profitably as one of statistical prediction, and standard linear predictive methods may be utilized to obtain results that can still be surprisingly accurate, even when based on $s \ll \mathsf{rank}(\mathbf{B})$ measurements. This approach was proposed by Chua, Kolaczyk, and Crovella [85], and its potential for success rests on the observation that in practice the *effective* rank of $\mathbf{B}$ can be smaller than its nominal rank – often noticeably so.

The singular value decomposition (SVD) is a standard tool for obtaining insight into the effective rank of a matrix. Given a routing matrix $\mathbf{B}$, recall that using the SVD, we can re-write $\mathbf{B}^T$ in the form $\mathbf{B}^T = \mathbf{U}\Lambda^{1/2}\mathbf{V}^T$, where $\mathbf{V}$ is an $N_e \times N_e$ orthogonal matrix, containing the eigenvectors of $\mathbf{B}\mathbf{B}^T$, Λ is an $N_e \times N_e$ diagonal matrix, containing the corresponding eigenvalues, and $\mathbf{U}$ is an $IJ \times N_e$ matrix for which $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. Alternatively, we can write

$$\mathbf{B}^T = \sum_{k=1}^{N_e} \lambda_k^{1/2} \mathbf{u}_k \mathbf{v}_k^T \quad , \tag{9.64}$$

where λ_k is the k-th largest diagonal entry of Λ, and $\mathbf{u}_k$ and $\mathbf{v}_k$ are the k-th columns of $\mathbf{U}$ and $\mathbf{V}$, respectively. See Golub and van Loan [181, Ch. 2.5], for example.

The number of non-zero entries on the diagonal of Λ will be equal to $\mathsf{rank}(\mathbf{B})$. So in fact equation (9.64) shows that $\mathbf{B}^T$ may be expressed as the sum of $\mathsf{rank}(\mathbf{B}) \leq N_e$ matrices $\mathbf{u}_k\mathbf{v}_k^T$, each of which itself is of rank one. However, in the event that some of the smaller values λ_k are 'close' to zero, we can truncate this sum even further and obtain an approximation to $\mathbf{B}^T$ that is still correspondingly ‘close’ to the original.[15] It is in this sense that $\mathbf{B}$ is then said to be of effectively lower rank than $\mathsf{rank}(\mathbf{B})$. We illustrate these ideas through the following example.

Example 9.8 (Reduced Dimensionality in Abilene). Figure 9.9 shows the eigenvalues for the routing matrix corresponding to the Abilene network that was described in the case study of Section 9.3.3. Recall that there are 110 paths traversing just 30 directed links. The large gap between the second and third eigenvalues, and the resulting ‘knee’ in the spectrum, are indicative of there being substantially more linear dependence among the rows of $\mathbf{B}^T$ (i.e., the columns of $\mathbf{B}$) than suggested by its nominal rank of 30. The overall decay in the spectrum of eigenvalues suggests that measurements on roughly five to ten paths, and perhaps as few as two paths, may be sufficient to recover useful information about path costs $\mathbf{c}$ in this network.

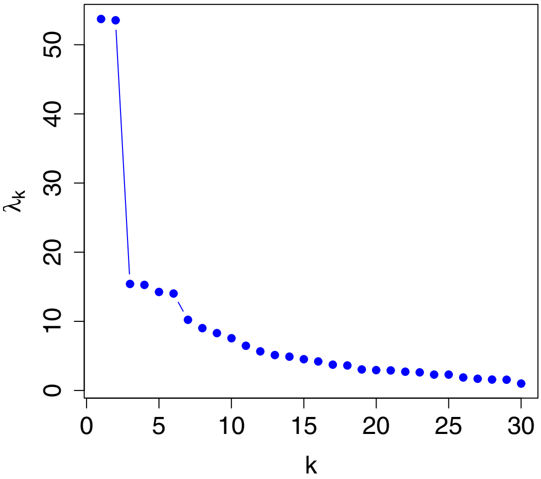

Fig. 9.9 Spectrum of eigenvalues for an Abilene routing matrix.

Chua, Kolaczyk, and Crovella [85] demonstrate an even more pronounced reduced dimensionality in the routing matrices for six additional networks, corresponding to commercial Internet service providers of various sizes. They also provide a general analytic result associating the eigenvalue decay for a routing matrix $\mathbf{B}$ with the decay of the distribution of edge betweenness values, thus providing a

[15] More precisely, let the truncation of (9.64) to the first K terms be denoted $\mathbf{B}_K^T$. Then the sum-of-squares of the entries (i.e., the squared Frobenius norm) of the difference $\mathbf{B}^T - \mathbf{B}_K^T$ is equal to $\lambda_{K+1} + \cdots + \lambda_{\mathsf{rank}(\mathbf{B})}$.

connection between the ideas in this section and the concept of centrality discussed in Section 4.2.2.

Additional insight into the network path properties on the Abilene network may be obtained through consideration of the eigenvectors in $\mathbf{V}$. For our purposes, these eigenvectors may be thought of as representing linearly independent 'meta-paths' in 'link space' (i.e., shared patterns of links common to the paths routed in this network), according to $\mathbf{B}$. These meta-paths are given an ordering through the magnitude of their corresponding eigenvalues, which essentially indicate for each meta-path the relative proportion of routed paths in the network that 'match.' Figure 9.10 shows visual representations of the first four eigenvectors of our Abilene routing matrix. These may be compared to the map of Abilene in Figure 1.1. The first eigenvector can be seen to concentrate along an east-west meta-path across the northern part of the network, while subsequent eigenvectors bring in successive refinements to this picture, with the second and third further emphasizing the connections to and within the meta-path indicated by the first, and the fourth introducing a second east-west meta-path along the southern part of the network. □

Now consider the problem of predicting path costs in $\mathbf{c}$. Actually, we will consider a slightly more general problem – that of predicting an arbitrary linear summary of the path costs. Specifically, suppose we wish to predict a summary of the form $\mathbf{a}^T\mathbf{c}$, where $\mathbf{a}$ is some known $IJ \times 1$ vector. Natural choices for $\mathbf{a}$ are $a_{ij} \equiv 1/IJ$, for which $\mathbf{a}^T\mathbf{c}$ is the network-wide average path cost, and $a_{ij} = 1$ and $a_{i'j'} = 0$ when $(i', j') \neq (i, j)$, for which $\mathbf{a}^T\mathbf{c}$ is just the cost c_{ij} of the origin-destination pair (i, j). Furthermore, assume that $\mathbf{x}$ is the realization of a random vector $\mathbf{X}$, with mean $\propto$ and covariance Σ. From (9.56) it then follows that $\mathbf{c}$ corresponds to a random vector $\mathbf{C}$ that has mean and covariance $\mathbf{B}^T\propto$ and $\mathbf{B}^T\Sigma\mathbf{B}$, respectively. Finally, given a set of measured path costs $\mathbf{c}_s$, corresponding to $s \leq \mathsf{rank}(\mathbf{B})$ independent paths in $\mathbf{B}_s$, and a predictor $p(\mathbf{c}_s)$ of $\mathbf{a}^T\mathbf{c}$, we will judge the quality of that predictor by the mean-squared prediction error $\mathbb{E}\left[\left(\mathbf{a}^T\mathbf{C} - p(\mathbf{C}_s)\right)^2\right]$.

The best predictor under this criterion is known to be the conditional expectation

$$\mathbb{E}(\mathbf{a}^T\mathbf{C} \mid \mathbf{C}_s = \mathbf{c}_s) = \mathbf{a}_s^T\mathbf{c}_s + \mathbb{E}(\mathbf{a}_r^T\mathbf{C}_r \mid \mathbf{C}_s = \mathbf{c}_s) \ , \tag{9.65}$$

where $\mathbf{a} = (\mathbf{a}_s^T, \mathbf{a}_r^T)^T$ is partitioned in the same manner as $\mathbf{c}$. If we restrict our attention to predictors $p(\mathbf{c}_s)$ that are linear in $\mathbf{c}_s$, then this expression yields the best linear predictor (BLP), which can be shown to take the form

$$\widehat{(\mathbf{a}^T\mathbf{c})}_{BLP} = \mathbf{a}_s^T\mathbf{c}_s + \mathbf{a}_r^T\mathbf{B}_r^T\propto + \mathbf{a}_r^T\eta_*\left(\mathbf{c}_s - \mathbf{B}_s^T\propto\right) \ , \tag{9.66}$$

with η_* defined to be any solution to the equation $\eta_*\Omega_{ss} = \Omega_{rs}$, where $\Omega_{ss} = \mathbf{B}_s^T\Sigma\mathbf{B}_s$ and $\Omega_{rs} = \mathbf{B}_s^T\Sigma\mathbf{B}_r$. If no value for $\propto$ is available, such as might be obtained from historical data or a limited set of network-wide path cost measurements, it can be estimated from the measured path costs $\mathbf{c}_s$ as

$$\hat{\propto} = \left[\mathbf{B}_s\Omega_{ss}^{-1}\mathbf{B}_s^T\right]^{-}\mathbf{B}_s\Omega_{ss}^{-1}\mathbf{c}_s \ , \tag{9.67}$$

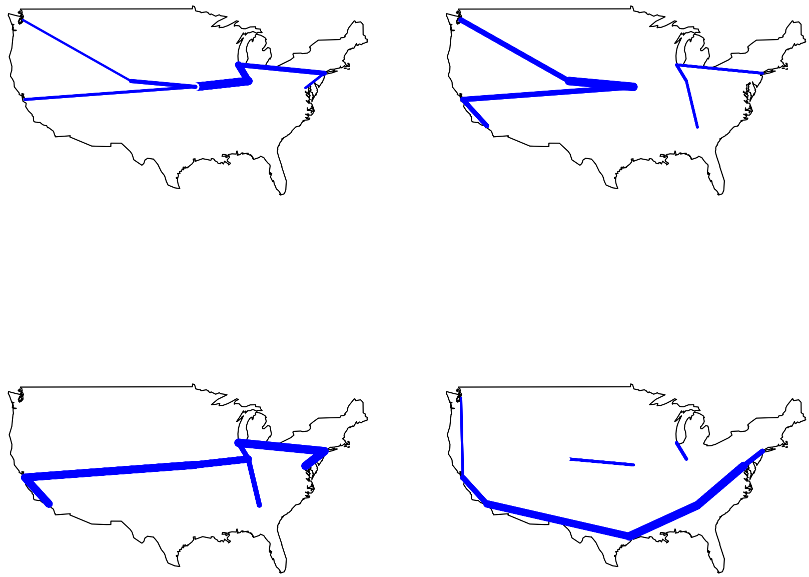

Fig. 9.10 Visual representation of the first four eigenvectors (top left and right, followed by bottom left and right, respectively) of an Abilene routing matrix. Each link is drawn with a thickness in proportion to the magnitude of its corresponding eigenvector component.

assuming Ω_{ss} is invertible, where $\mathbf{M}^{-}$ again denotes the Moore-Penrose pseudo-inverse of a matrix $\mathbf{M}$. Substitution of (9.67) for $\propto$ in (9.66) can be shown to yield a predictor of the form

$$\widehat{(\mathbf{a}^T\mathbf{c})} = \mathbf{a}_s^T\mathbf{c}_s + \mathbf{a}_r^T\,\Omega_{rs}\Omega_{ss}^{-1}\mathbf{c}_s \ . \tag{9.68}$$

The predictor (9.68) and its derivation parallel closely what may be found in the geostatistics literature on spatial prediction (i.e., on kriging). See, for example, Christensen [84, pp. 225-227] for details. Accordingly, prediction using (9.68) has been referred to as ‘network kriging.’ If $\Sigma = \mathbf{I}$ and $s = \mathrm{rank}(\mathbf{B})$, then this predictor effectively performs the interpolation of Chen, Bindel, Song, and Katz [80] in (9.63). In that case, of course, we recover the exact value of $\mathbf{a}^T\mathbf{c}$.

For the task of selecting just which s paths to measure costs upon, Chua, Kolaczyk, and Crovella [85] propose an algorithm based on the SVD of $\mathbf{B}$ that is analogous to the QR-based algorithm of Chen, Bindel, Song, and Katz [80] mentioned previously. They argue that such algorithms can be viewed as trying to select a subset of s paths that minimizes the mean-squared prediction error corresponding to the best linear predictor in (9.66), as a function of all possible subsets of s independent paths. They also characterize the theoretical performance that can be expected of network kriging. We illustrate here with the following numerical example, taken from their work, involving the prediction of path delay times.

Example 9.9 (Predicting Average Path Delay on Abilene). Chua, Kolaczyk, and Crovella [85] describe a process whereby they constructed a dataset of link delays $\mathbf{x}^{(t)}$ in the Abilene network, for every 10 minutes over a period of three days in 2003. From these they created a set of path-cost pseudo-measurements $\mathbf{c}^{(t)} = \mathbf{B}^T\mathbf{x}^{(t)}$, using the Abilene routing matrix $\mathbf{B}$ of Example 9.8. The predictor (9.68) was then applied to a subset $\mathbf{c}_s^{(t)}$ of these path costs, separately for each time t, for various choices of $s \leq \mathsf{rank}(\mathbf{B})$, where the paths measured were chosen according to the SVD-based algorithm mentioned just above. The covariance matrix Σ was taken to be diagonal, and was estimated as described in Chua, Kolaczyk, and Crovella [85].

Figure 9.11 shows the average path delay on Abilene predicted by network kriging, based on measurements from $s = 3,5,7$, or 9 paths, as a function of time period $t = 1,\ldots,432$, and the actual average path delay, which can be obtained from measurements on $s = \mathsf{rank}(B) = 30$ appropriately chosen paths. Two points are evident from examination of this figure. First, there clearly is a bias to the predictions, which becomes more pronounced as s decreases. Second, the predictions nevertheless capture much of the dynamics quite well, for all choices of s. The bias is to be expected, and is a result of the fact that $\mathbf{B}$ is approximated by a matrix of rank $s < 30$. Practically speaking, information is lacking on links in the network that are traversed by none of the s measured paths.

However, it turns out that we can correct for this bias quite easily in these data if we allow ourselves to measure the delay on a set of 30 independent paths just once at the start of the three days studied. This use of auxiliary measurements is similar to that employed in the case study of Section 9.3.3 and represents a negligible addition to the overall network measurement load. Adjusting the predictions by the bias we observe in the first 10-minute time period produces predictions with an average of less than 1% absolute relative error, even with as little as $s = 3$ measured paths.

The accuracy of these predictions can in turn serve as the foundation for performing various high-level tasks. Chua, Kolaczyk, and Crovella [85] describe the use of the predicted average path delay in monitoring for anomalous levels of delay and in comparing the performance of two sub-networks. □

Some extensions of the basic network kriging framework described here have been proposed. Qazi and Moors [317] have demonstrated that the use of principles of ridge regression (i.e., such as underlie equation (2.38) in Chapter 2) can help produce predictions with lower variance than those of (9.68). Coates, Pointurier, and Rabbat [98] introduce the use of Lasso regression and an alternative basis representation, within the context of a spatio-temporal approach. The incorporation of temporal correlations into the predictions is observed by these authors to noticeably

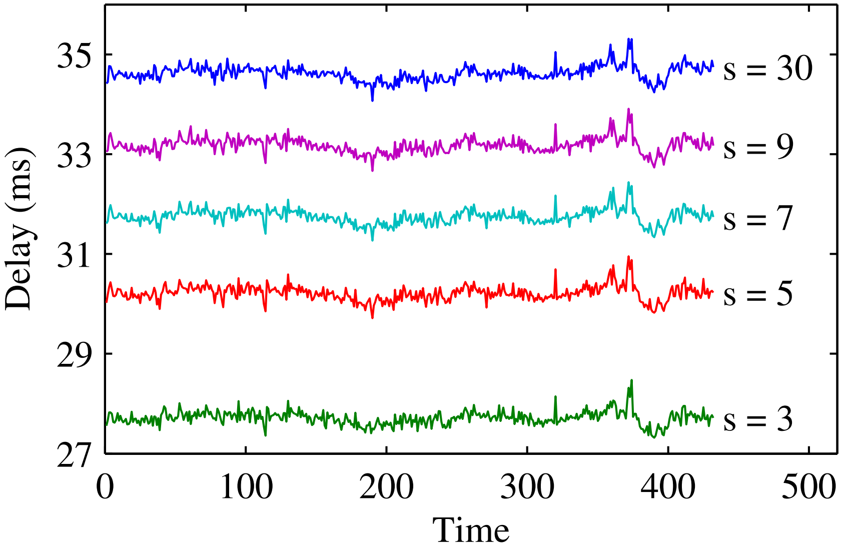

Fig. 9.11 Network kriging predictions of Abilene average delay over a period of three days.

decrease the bias, as might be expected. Finally, Sommers, Duffield, Barford, and Ron [363] describe a method for producing upper and lower bounds on quantiles of the distribution of costs on each path in the network. See also the work by Nguyen and Thiran [300, 301], who treat problems similar to those addressed in this section, but which involve binary measurements and arithmetic in the context of monitoring the ‘aliveness’ of paths in Internet networks.

9.5 Additional Related Topics and Reading

Besides those considered here, there are a number of other statistical topics relating to network flow data that have received nontrivial – and in some cases, substantial – attention in the literature. We briefly summarize a few of these now. As in the main body of this chapter, the examples we provide are dominated by contributions from the extremely active area of computer network traffic analysis, to the somewhat unfair exclusion of other areas in transportation more broadly.

The Poisson-like, and more generally, Markovian nature of the classical models for network flow data, of which the gravity models we discussed are perhaps the quintessential example, can in some contexts be quite insufficient. For example, while historically such models were considered reasonable for voice-data on standard telephone networks, like the Austrian call data we saw in Section 9.2, with the emergence of high-speed computer traffic networks, a new paradigm emerged. In particular, the celebrated paper of Leland, Taqqu, Willinger, and Wilson [254] established that traffic of high temporal resolution on so-called local-area networks (LANs) was *statistically self-similar* (and therefore decidedly non-Markovian), in the sense that the temporal correlation structure of the underlying process was essentially indistinguishable from that of aggregated versions of itself. See the edited volume compiled by Park and Willinger [308], which covers a variety of developments building on this seminal work, as it relates to topics ranging from estimation and simulation, to queuing and congestion, to traffic control.

As the collection of network flow data becomes more and more automated, issues of sampling need to be examined. In computer networks, for example, routers are designed with the facility to sample from the massive flows of data streaming through them. A number of sampling designs are employed in this setting, including systematic sampling, random sampling, and stratified sampling. See, for example, Crovella and Krishnamurthy [105, Ch. 6.3.3] for discussion of such designs and the effect they can have on the types of statistical analyses to which such data are subjected. Similarly, the use of embedded devices (e.g., loop-detectors), cameras, and other related sensing devices is becoming increasingly common in the monitoring of highway traffic. Bickel et al. [38] offer a review of some of the statistical challenges that are emerging as a result, including issues of data quality and missing data.

Lastly, with respect to the measurement and inference of costs in networks, it should be noted that certainly not all costs of interest are amenable to linear formulations like that in (9.56). See, for instance, Shriram and Kaur [351] for a survey and evaluation of techniques for measuring the amount of end-to-end bandwidth available on computer network paths. Conversely, the exploitation of ‘spatial’ correlations among paths is not restricted to just problems involving cost. Lakhina et al. [245], for example, show that an analogous sense of reduced dimensionality exists for traffic flow volumes in backbone-level Internet networks, and use principal component analysis (PCA) to characterize typical temporal patterns across flows. This work in turn has served as the foundation for a series of papers by Lakhina and colleagues on the detection of computer network traffic anomalies using PCA-based methods, including Lakhina, Crovella, and Diot [244]. See also Zhang, Ge, Greenberg, and Roughan [415], whose work aims to address the problems of anomaly detection and network tomography within a single unified framework – which they dub *network anomography* – and which includes the PCA-based framework of Lakhina and colleagues as a special case.

Exercises

9.1. For the general gravity model, under the parameterization in (9.14), the log-likelihood (9.15), as a function of the parameter vectors $\alpha = (\alpha_1, \ldots, \alpha_I)^T$, $\beta = (\beta_1, \ldots, \beta_J)^T$, and $\theta = (\theta_1, \ldots, \theta_K)^T$, takes the form

$$\ell(\alpha, \beta, \theta) = \sum_{i,j \in \mathscr{I} \times \mathscr{J}} z_{ij} \left(\alpha_i + \beta_j + \theta^T \mathbf{c}_{ij} \right) - \exp\left(\alpha_i + \beta_j + \theta^T \mathbf{c}_{ij} \right) \ .$$

Taking partial derivatives with respect to the elements α_i, β_j, and θ_k, setting the resulting equations equal to zero, and simplifying, show that the maximum likelihood estimates of α, β, and θ must satisfy the expressions in (9.16) and (9.17).

9.2. Consider the least-squares formulation of the static traffic matrix estimation problem, as introduced in Section 9.3.1.1. Recall that the goal in this problem is to recover the $IJ \times 1$ vector $\propto$ of expected flow volumes from the $N_e \times 1$ vector of link counts $\mathbf{X}$, where

$$\mathbf{X} = \mathbf{B} \propto + \varepsilon \ ,$$

with $\mathbf{B}$ an $N_e \times IJ$ routing matrix and ε an $N_e \times 1$ vector of errors. Let η be an arbitrary $IJ \times 1$ vector. The value $\eta^T \propto$ is said to be *estimable* if there exists a matrix $\mathbf{M}$ such that $\mathbb{E}(\mathbf{MX}) = \eta^T \propto$.

a. Show that $\eta^T \propto$ is estimable if and only if η is a solution to the equation

$$[\mathbf{I} - \mathbf{B}^T (\mathbf{B}\mathbf{B}^T)^- \mathbf{B}]\eta = 0 \ ,$$

where $(\mathbf{B}\mathbf{B}^T)^-$ denotes the Moore-Penrose pseudo-inverse of $\mathbf{B}\mathbf{B}^T$.

b. For the simple network in Example 9.3, show that the expected origin volumes $\propto_{a+}$ and $\propto_{b+}$ and the expected destination volumes $\propto_{+c}$ and $\propto_{+d}$ are all estimable.

c. If $\eta^T \propto$ is estimable, then given observed link volumes $\mathbf{X} = \mathbf{x}$, the best linear unbiased estimator[16] of $\eta^T \propto$ is $\eta^T \hat{\propto}$, where $\hat{\propto} = (\mathbf{B}^T\mathbf{B})^- \mathbf{B}^T \mathbf{x}$.
Show that, for the network in Example 9.3, the matrix $(\mathbf{B}^T\mathbf{B})^- \mathbf{B}^T$ can be expressed in the form

$$\frac{1}{4} \begin{bmatrix} 2 & 2 & -2 & -2 \\ 1 & -1 & 3 & 1 \\ -1 & 1 & 1 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix} \ .$$

Then, given observations $\mathbf{x} = (30, 27, 15, 10)^T$, show that

$$(\hat{\propto}_{a+}, \hat{\propto}_{b+}, \hat{\propto}_{+c}, \hat{\propto}_{+d}) = (30.5, 10.5, 26.5, 14.5) \ .$$

[16] That is, an unbiased estimator that, among all other unbiased estimators, has minimal variance.

9.3. Consider the Poisson version of the static traffic matrix estimation problem, restricted to the subnetwork of the network in Figure 9.4 consisting of just the three vertices $\{a, v, c\}$ and the two links 1 and 2.

a. Verify the result of Example 9.4, showing that upon observing the link counts $\mathbf{x} = (1, 2)^T$, the maximum likelihood estimate of the expected flow volumes $\propto$ is not given by that value for which the score equations $\partial \ell(\propto)/\partial \propto = 0$.

b. Verify the results of Example 9.5, concluding that for this subnetwork we can simulate from the full joint conditional posterior $\mathbb{P}(\mathbf{Z} \mid \propto, \mathbf{X} = \mathbf{x})$ by first drawing $Z_{ac} = z_{ac}$ and then setting $z_{av} = x_1 - z_{ac}$ and $z_{vc} = x_2 - z_{ac}$.

9.4. Consider the problem of predicting path cost characteristics $\mathbf{a}^T \mathbf{c}$, based on measurements $\mathbf{c}_s$ of a subset of path costs, as described in Section 9.4.2.2.

a. Verify that substitution of (9.67) into (9.66) yields the predictor (9.68).

b. Implement this predictor, and apply it to the Abilene delay data, in the manner of Example 9.9. Explore the accuracy with which you can predict the average delay, as you vary the choice of paths measured. Use the eigenvectors of the routing matrix to help guide your choice of measurements.

c. The prediction of average path delay uses the weight vector **a** with entries $a_{ij} = 1/IJ$. In order to predict the cost for an individual origin-destination pair (i, j), we can let $a_{ij} = 1$ and $a_{i'j'} = 0$ for all $(i', j') \neq (i, j)$. Using the Abilene delay data, explore the accuracy with which you can predict the cost of individual unmeasured paths, as you vary the choice of paths measured.