Lesson 2

About Distributions

When modeling large amounts of data, individual samples are of no special relevance. To construct *relevant* models, one has to reach the “big picture” beyond the surface. The relevance is captured in the statistical general properties of the whole data set; these statistical properties are represented by *probability distributions.* After all, it is distributions that are being modeled by the data-oriented methods.

To find appropriate methods for data modeling, it is also necessary to first study the properties of data distributions. Later, however, the statistical considerations can be ignored: Assumptions concerning the data-generating procss make it possible just to concentrate on some emergent distribution characteristics, like variance and covariance. The existence of the underlying distributions is reflected in the modeling methods and resulting model structures. In this chapter, the basic model structures are motivated in statistical terms, and their correspondence with real data is discussed.

2.1 Data mining

When facing something new, it is clever to first look it from a distance, from different points of view. It is the same with data: Before starting any harder labor, it is clever to gain insight. There are efficient and innovative data analysis tools available where the computing power available today is utilized to reveal different ways to see the data.

The diversity of data mining approaches and tools is not surveyed here. Only as an example, in Fig. 2.1 industrial data is visualized applying the *Self-Organizing Map* or *SOM* (see [**?**]; also see Sec. 8.1.1). SOM efficiently utilizes the human pattern recognition capability: The data is typically projected onto a two-dimensional surface, so that the dependencies among data are visually manifested. However, *computer is notoriously bad in such pattern recognition tasks;* SOM is a good front-end for humans, but not for implementing some machine-to-machine (or “algorithm-to-algorithm”) interaction.

Such SOM models can directly be used, and they have been used, also directly

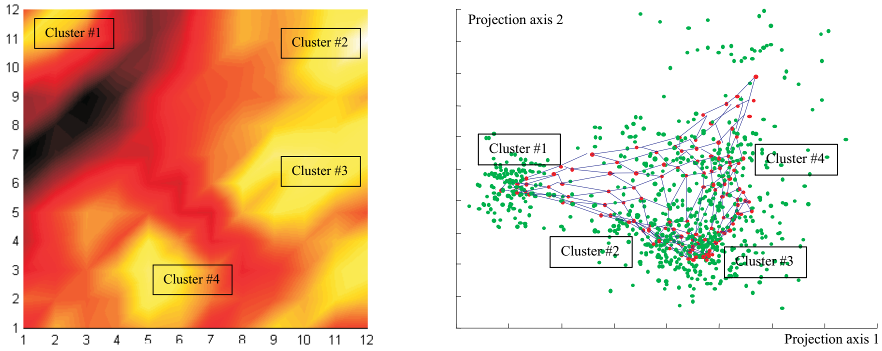

Figure 2.1: How the character of the data can be visualized: The SOM approach. High-dimensional industrial data has been projedcted onto a two-dimensional manifold or "hypersurface", so that the *topology* among the data has been maximally preserved. On the left, the 12 × 12 SOM grid is shown: The regions of many "hits" have been printed with lighter color. On the right, the converged SOM map itself has been projected into the original variable space, showing its curved nature. It seems that there are perhaps four (or more?) separate concentrations of data, or *clusters,* perhaps revealing something about the variability in the operating regimes in the process

for process monitoring purposes, etc., but when implementing prediction or control, SOM should be seen as a pre-analysis tool only. SOM implements extreme compression, mapping data from high-dimensional continuous-valued variable space onto a discrete set of map nodes, so that unavoidably very much of the available information is lost. Better regression can be implemented if the intuition offered by SOM is exploited for adjusting the more traditional modeling methods.

It also needs to be mentioned that when the data is high-dimensional, the wonders of high dimensions can look too fancy for the inhabitants of Flatlands. The higher the dimension, the more there exist alternative explanations for the observations, at least if the evidence is interpreted in an appropriate way ... One should remember the "Barnum effect" and recognize that *You see what you want to see.*

2.2 Normal distribution

The *Gaussian* or *normal distribution* is the most important abstraction for more or less stochastic measurement data. The famous *central limit theorem* states that if a large number of independent random variables are added together, the sum is normally distributed under very general conditions, no matter what is the distribution of the original variables. Usually, when making process measurements, it can be assumed that underlying the actual measurement values there is a large number of minor phenomena that cannot be separately analyzed

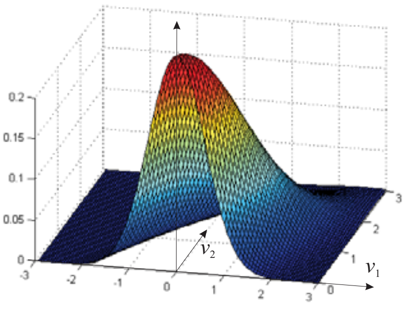

Figure 2.2: Visualizing the density of the two-dimensional Gaussian distribution: In surface form ...

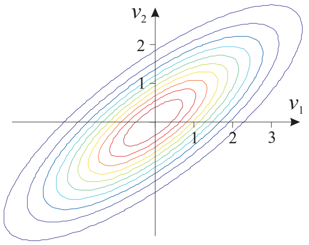

Figure 2.3: ... and as a contour map. Here the covariance matrix was such that $r_{11} = r_{22} = 1$ and $r_{12} = r_{21} = 0.8$

— their net effect, according to the central limit theorem, is that the overall distribution becomes normal.

Similarly as in the one-dimensional case, *multinormality* holds for multivariate data (see Figs. 2.2 and 2.3). Let v stand for the measurement vector of length $\dim\{v\}$. Assuming multinormal distribution, the density function value (corresponding to the probability; note that finite probabilities are found only when the density function is integrated within some region in v space) for a data sample v can be calculated as

$$p(v) = \frac{1}{\sqrt{(2\pi)^{\dim\{v\}} \det\{R\}}} \, \mathrm{e}^{-\frac{1}{2}\cdot(v-\bar{v})^T R^{-1} (v-\bar{v})}. \tag{2.1}$$

Here, $\bar{v}$ stands for the center of the distribution and R is the covariance matrix, $\det\{R\}$ being its determinant. This prototypical distribution can be compactly denoted as $\mathcal{N}\{\bar{v}, R\}$. The distribution formula consists essentially of a (decaying) exponential function, making the "bell-shaped" distribution extend to infinity in all directions; due to the normalization factor, its integral over the whole space equals 1. The statistical properties of multinormal distributions are not elaborated on in this context; it suffices to note that all projections of a normal distribution are also normal, and, generally, linear functions of normally distributed data result in normal distributions (see Figs. 2.2 and 2.3).

It needs to be noted that above it is all variables that are assumed similarly stochastic. Traditionally when doing modeling and identification, there is a distinction between deterministic and stochastic variables; there are inputs and outputs; there is noise and there is information. Now, the framework is homogeneous: All variables have the same stochastic nature to begin with. This means that methodologies for analysing the variables also remain uniform. The information is assumed to be buried in correlations among the variables.

2.2.1 About distribution parameters

Multinormal distribution is uniquely determined by its mean and covariance. If there are measurements $v(1)$ to $v(k)$ taken from the distribution, the unbiased mean, $\bar{v} = \mathrm{E}\{v(\kappa)\}$, can be approximated as the *sample mean*

$$\bar{v} = \frac{1}{k} \cdot \sum_{\kappa=1}^{k} v(\kappa). \tag{2.2}$$

The covariance matrix, $R = \mathrm{E}\{v(\kappa)v^T(\kappa)\}$, can be approximated as *sample covariance*

$$R = \frac{1}{k} \cdot \sum_{\kappa=1}^{k} (v(\kappa) - \bar{v})(v(\kappa) - \bar{v})^T, \tag{2.3}$$

or, if the individual sample vectors v are collected as rows in the $k \times \dim\{v\}$ matrix V,

$$R = \frac{1}{k} \cdot (V - \bar{V})^T (V - \bar{V}). \tag{2.4}$$

The matrix $\bar{V}$ now consists of k copies of $\bar{v}^T$. It is assumed that the covariance matrix has full rank and it is invertible; this means that necessarily there must hold $k \geq \dim\{v\}$ (there are at least as many data vectors as there are separate measurements in the measurement vector) and the measurements V_1 to $V_{\dim\{v\}}$ are *linearly independent.*

Note that the presented estimate for covariance that is based on the estimate of the sample mean is *biased;* one should take into account the reduced degrees of freedom to find the unbiased estimate (that is, the denominator should read $k - \dim\{v\}$). However, it is not always clear what is the theoretically appropriate normalizing factor (for example, if calculating the cross-correlation $X^T Y$, where X is a $k \times n$ matrix and Y is a $k \times m$ matrix). In what follows, it is assumed that the number of measurements is so high that this bias can be neglected (that is, $k \gg \dim\{v\}$). Later, it turns out that it is the covariance matrix that plays a central role when determining the model structure — and it is the *structure* of the covariance matrix that is of relevance, ratios between elements, revealing the interconnections among variables, not its *scaling.*

The covariance matrix is such a central data construcy in subsequent analyses that it deserves a still closer look — it is still *intuition* that plays a central role when constructing good models. Understanding the structure of the covariance matrices, and understanding how this structure is related to data properties, is fundamental knowledge when trying to understand multivariate statistical methods.

As visualized in Fig. 2.3, the covariance structure can be visualized in terms of (hyper)ellipsoids in the data space: The ellipsoids represent the "equi-probability" surfaces in the data space. The projections of Gaussians onto lower dimensions (also having Gaussian distribution) can be visualized es ellipses.

	About mean	**About origin**
Unnormalized	Covariance matrix	Inner product matrix
Normalized	Correlation matrix	Cosine matrix

Figure 2.4: Some association matrices

It is instructive to interpret the covariances in terms of concrete ellipses, end here are some guidelines to interpret them. The variances of the individual variables that are collected on the diagonal of the covariance matrix dictate how far the ellipsoid extends in that variable direction; zero variance means that the ellipsoid "collapses" into a (hyper)planar structure. The non-diagonal elements in the covariance matrix reveal how much the ellipsoid is "tilted" as compared to the variable axes. This "tiltedness" connects the variables together, variables becoming dependent, and it is indeed these dependency structures, cross correlations, that make it possible to estimate the values of some variables when some other variables only are known, making regression analysis feasible. However, the properties of such tilted ellipsoids cannot be seen in the original coordinate frame, and more closer analyses have to be postponed to the eigenvalue/eigenvector analysis of the covariance matrix in Chapter 5. It turns out that the extent of the ellipsoid in different directions (as determined by the eigenvectors of the covariance matrix) is revealed by the square roots of the corresponding eigenvalues; the "volume" of the ellipsoid is proportional to the product of the eigenvalues.

2.2.2 Association matrices

The covariance matrix reveals the second-order properties of the data (variances and co-variances) in a compact form, and it turns out that it is these second-order properties that one concentrates on in multivariate modeling. It turns out that *determination of the model structure is based on the analysis of the data covariances.* However, there also exist other ways to capture the second-order properties.

Covariance measures *similarity* between variables, and it makes it possible to define *associations* among them. Generalizing slightly, rather than speaking merely of covariance matrices, we can speak of *association matrices.* The idea is the same: the second-order "nearness" properties between variables should be captured in a compact form so that the assumedly relevant phenomena would become tractable. Fig. 2.4 shows some common selections that are found when the data either *is* centralized or it is not, and when the data either *is* normalized to unit variance or it is not (there will be more about data preprocessing in the next chapter). In all of the above cases, the association matrices are constructed as

$$R = \frac{1}{k} \cdot \sum_{\kappa=1}^{k} x'(\kappa) x'^T(\kappa) = \frac{1}{k} \cdot X'^T X' \tag{2.5}$$

where x' is the correspondingly scaled (and perhaps centered) data sample. Again, if being theoretically orthodox, one would have problems with the nor-

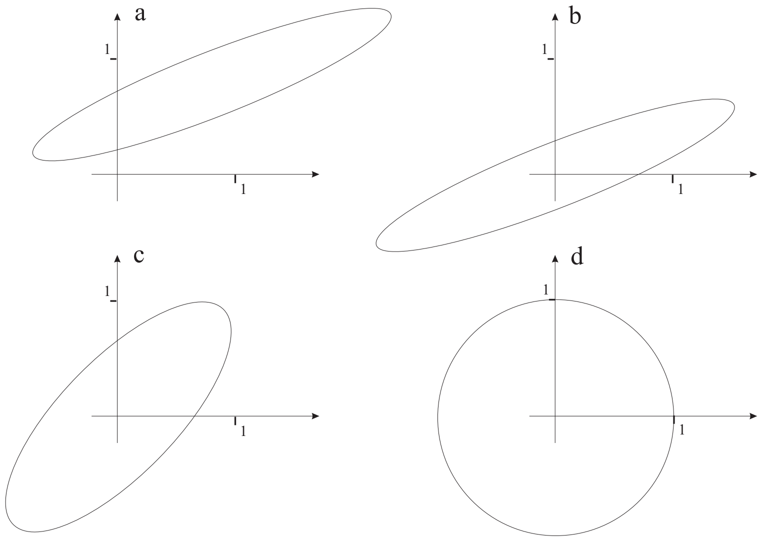

Figure 2.5: Data distributions after different preprocessing operations. First, in **a**, the assumed original data distribution is shown, and in **b** data is *centered.* In **c** data is additionally *normalized* to unit variance, and in **d**, it is *whitened* (in addition to being centered)

malization factor: If the origin is now assumed to be the "center" of data the degrees of freedom are not reduced?

All of the above association matrices are positive semi-definite, that is, $\xi^T R\xi \geq 0$ for any vector ξ:

$$\xi^T R\xi = \frac{1}{k} \cdot \sum_{\kappa=1}^{k} \xi^T x'(\kappa)x'^T(\kappa)\xi = \frac{1}{k} \cdot \sum_{\kappa=1}^{k} \left(\xi^T x'(\kappa)\right)^2 \geq 0. \tag{2.6}$$

This means that all eigenvalues are non-negative. Note that when discussing general association matrices one is violating the basic assumptions concerning covariance matrices on purpose: One is no more analyzing the properties of the original Gaussian distribution but some *virtual* distribution. "Forgetting" the centering, for example, has major effect on the data distribution as seen by the algorithms. In Fig. 2.5, the effects of different preprocessing operations (see Chapter 3) on the data distribution are shown.

There are also other possibilities for constructing matrices that are related to similarity matrices — for example, the *distance matrix,* where the element R_{ij} is the distance (Euclidean or other) between vectors X_i and X_j, can be used for structuring the relationship between variables (note that the diagonal contains zeros, making this matrix to be *not* positive definite); also see Sec. 7.3.3. The *Kernel matrices* are yet another of representing (nonlinear) relationships between variables (see Appendix 2). When similarity is measured in some feature space, so that one applies similarity matrices of the form $\mathrm{E}\{f(x)f(x)^T\}$ for analysis, where the features are determined through the nonlinear vector-valued function f, one sometimes speaks of *nonlinear component analysis* or *kernel PCA* (compare to Chapter 5). Indeed, determination of the function f, or feature extraction, is discussed in the next chapter.

Depending on the situation, it can be motivated to study the connections among *samples* rather than among *variables,* that is, rather than finding the structure for X^TX, one can search for the structure of the matrix XX^T. Note that the non-zero eigenvalues are the same in both cases.

The data can also be scaled samplewise; If there holds $y(\kappa) = F^T \cdot x(\kappa)$, then, for some scalar function $g(x(\kappa), y(\kappa), \kappa)$, there must also hold

$$g\, y(\kappa) = F^T\, g\, x(\kappa), \tag{2.7}$$

and these scaled variables can be just as well be used for determining F. Even though the expression above looks like an identity, the statistical properties of the data may be changed remarkably when the samples are individually scaled (see Sec. 7.3.2): This kind of "samplewise" scaling can also be justified if one knows that different samples have different levels of reliability — or if the noise variance level varies along the sampling; this is sometimes called *heterosedasticity.* Specially, assume that $g(\kappa)$ is a function of time index κ alone, and study the properties of the correlation matrix:

$$R = \frac{1}{\sum_\kappa} \cdot \sum_{\kappa=1}^{k} g(\kappa)\, v(\kappa) v^T(\kappa). \tag{2.8}$$

Here, the normalizing factor compensates for the scaling effect of the sequence of the weighting factors. Further, assuming that one wants to apply *exponential forgetting,* so that the "memory" gradually fades away, one can select $g(\kappa) = \lambda^{k-\kappa}$, where $0 \ll \lambda < 1$ is the forgetting factor, one can write the recursive adaptation rule for the covariance estimate in the familiar-looking form (mathematical interpretations ranging from weighted-average to convex-combination):

$$R(k) = \lambda\, R(k-1) + (1-\lambda)\, v(k) v^T(k). \tag{2.9}$$

2.2.3 χ^2 distribution

Multivariate normal distribution (2.1) gives a probability of any point to belong to a Gaussiann distribution. However, in a high-dimensional space the probability of *any location* becomes very low — one would like to have a scalar measure for easily studying whether an observation is characteristic to a distribution or not, regardless of the data dimension. It turns out that the χ^2 *distribution* is a practical tool for this purpose.

If there are n independent, normally distributed normalized variables v_i, where $1 \leq i \leq n$, *the sum of the squares, or* v^Tv, *has* χ^2 *distribution with degrees of freedom* n. This sounds like a rare special case, but this is not so. Study the case where Gaussian variable vectors ν are normalized so that

$$v(\kappa) = R_\nu^{-1/2} \nu(\kappa), \tag{2.10}$$

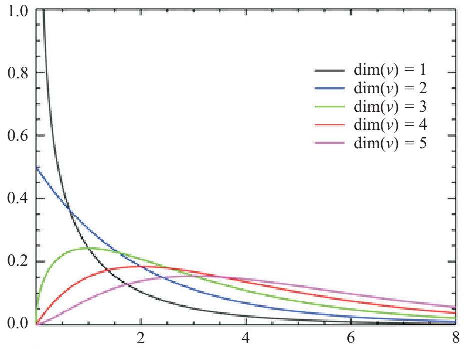

Figure 2.6: Note that in higher dimensions the maximum of χ^2 distribution is not in zero!

where

$$R_\nu = \frac{1}{k} \sum_{\kappa=1}^{k} \nu(\kappa)\nu^T(\kappa). \tag{2.11}$$

Then the new variables v are Gaussian and there holds

$$\begin{aligned} R_v &= \tfrac{1}{k} \sum_{\kappa=1}^{k} v(\kappa)v^T(\kappa) = R_\nu^{-1/2} \tfrac{1}{k} \sum_{\kappa=1}^{k} \nu(\kappa)\nu^T(\kappa)\; R_\nu^{-T/2} \\ &= R_\nu^{-1/2} R_\nu R_\nu^{-T/2} = I. \end{aligned} \tag{2.12}$$

This means that the familiar expression has χ^2 distribution:

$$v^T(\kappa)v(\kappa) = \nu^T(\kappa)R_\nu^{-T/2}R_\nu^{-1/2}\nu(\kappa) = \nu^T(\kappa)R_\nu^{-1}\nu(\kappa). \tag{2.13}$$

This $\nu^T R_\nu^{-1} \nu$ is a quantity that is routinely computed in multivariate analysis, and it makes it possible to reduce the high-dimensional distribution into one dimension. The χ^2 distribution can be found, for example, in `Matlab`; to use the functions there, one needs to determine the degrees of freedom, or the number n (see Fig. 2.6). In the Regression Toolbox, there is the function `regrP` that is tailored for course usage.

2.3 Motivation of modeling approaches

After this chapter, the distributions are abstracted away — one only concentrates on a single distribution parameter, (co)variance, forgetting about the other distribution properties. Calculation of variance and covariance can be carried out for any set of data, regardless of the actual distribution, and, similarly, the regression models that are presented later being based on covariance properties can be constructed. However, the Gaussianity assumption is implicitly buried in the model structures: As it turns out, the adopted modeling principles are not only pragmatically motivated, they are *optimal* for the Gaussian distribution.

2.3.1 Why linear models?

Let us study some special properties of the Gaussian distribution. The "most probable" regions in the data space are determined by the formula (2.1). For simplicity, assume that data is zero-mean, $\bar{v} = 0$. Because the exponent function is a monotonously increasing function, the maximum of probability is reached when the following expression reaches minimum:

$$J = v^T R^{-1} v \tag{2.14}$$

To proceed, one has to distinguish between the roles of individual variables in v. As explained in the next chapter, it is reasonable to separate the *input variables* and *output variables* from each other. If it is assumed that some of the variables in v, collected in the vector x, are known, and some, collected in y, are unknown, so that

$$v = \left(\begin{array}{c} x \\ \hline y \end{array} \right), \tag{2.15}$$

expression (2.14) can be divided in parts:

$$J = \left(\begin{array}{c} x \\ \hline y \end{array} \right)^T \left(\begin{array}{c|c} (R^{-1})_{xx} & (R^{-1})_{xy} \\ \hline (R^{-1})_{yx} & (R^{-1})_{yy} \end{array} \right) \cdot \left(\begin{array}{c} x \\ \hline y \end{array} \right), \tag{2.16}$$

or, written explicitly,

$$J = x^T (R^{-1})_{xx} x + x^T (R^{-1})_{xy} y + y^T (R^{-1})_{yx} x + y^T (R^{-1})_{yy} y. \tag{2.17}$$

Here the matrices $(R^{-1})_{xx}$, etc., are formally used to denote the blocks of the inverse covariance matrix; how they should actually be constructed is not of interest here. Minimization with respect to y means solving

$$\begin{aligned} \frac{dJ}{dy} &= \frac{d}{dy} \left(x^T (R^{-1})_{xx} x + x^T (R^{-1})_{xy} y + y^T (R^{-1})_{yx} x + y^T (R^{-1})_{yy} y \right) \\ &= \mathbf{0}, \end{aligned}$$

giving a unique solution:

$$((R^{-1})_{xy})^T x + (R^{-1})_{yx} x + ((R^{-1})_{yy})^T y + (R^{-1})_{yy} y = \mathbf{0}, \tag{2.18}$$

or

$$y = \left(((R^{-1})_{yy})^T + (R^{-1})_{yy} \right)^{-1} \left(((R^{-1})_{xy})^T + (R^{-1})_{yx} \right) x. \tag{2.19}$$

This can be expressed in a very simple form

$$y = Mx. \tag{2.20}$$

It is not of interest here to study any closer the matrices that constitute the solution, or what is the structure of the matrix M; these issues will be concentrated on later in detail. What is crucial is the basic outlook of the maximum likelihood (ML) solution for the regression problem: *The unknown variables are linear functions of the known ones.* Within a Gaussian distribution, linear estimates are optimal — this is a very useful result, justifying the simple model structures that will be applied later.

To be exact, assuming that the distribution is not zero-mean, the general maximum likelihood relationship between variables becomes *affine:*

$$y = Mx + c, \tag{2.21}$$

where c is a constant vector. However, models will be assumed strictly linear later — the techniques to avoid problems that are faced because of this assumption will be discussed in the next chapter.

2.3.2 Why sum-of-error-squared criteria?

Continuing from the above linear model structure, assume that there exists such a matrix M that maps x onto y, so that (2.20) is assumed to apply, and one's task is to determine this mapping matrix. Typically (if $k > n$) exact matching cannot be reached, so that for each sample κ there remains a residual error

$$e(\kappa) = y(\kappa) - Mx(\kappa). \tag{2.22}$$

If the data is Gaussian, also this error has Gaussian distribution. Further, assume that the errors in the sequence $e(\kappa)$ are independent of each other, and have identical Gaussian distribution with mean $\bar{e} = 0$ and covariance R_e. The best choice for the matrix M maximizes the probability that the observed sequence of samples has been obtained — that is, the probabilities of observing the sequence $e(\kappa)$ should be maximized. Because the individual errors were assumed independent, the overall probability is the product of individual probabilities, so that the *likelihood function* now becomes

$$\begin{aligned} L &= \prod_\kappa p(e(\kappa)) \\ &= \frac{1}{\sqrt{2\pi \det\{R_e\}}}\, \mathrm{e}^{-\frac{1}{2}\sum_\kappa (y(\kappa)-Mx(\kappa))^T R_e^{-1}(y(\kappa)-Mx(\kappa))}. \end{aligned} \tag{2.23}$$

Because the logarithm function is monotonously increasing, the maximum of the above criterion equals the minimum of the following:

$$J = -\log L = c + \sum_{\kappa=1}^{k} \left(y(\kappa) - Mx(\kappa)\right)^T R_e^{-1} \left(y(\kappa) - Mx(\kappa)\right). \tag{2.24}$$

for scalar y, this reduces essentially to a sum of squared errors, J is proportional to $\sum_\kappa e^2(\kappa)$. This all means that the criterion that makes it possible to find solutions in a mathematically closed form, is again *optimal* for Gaussian data.

There are also many other reasons for selecting the error-squared criterion: From the theoretical point of view, it is nice that the minimum of the quadratic criterion is unique, so that no closer analyses of candidate solutions is needed;

from the practical point of view, it is nice that this criterion has rather natural interpretations in terms of signal powers, error squares are related to noise variances, capturing the essence of the noise distributions, etc. However, in some cases the emphasis on the error squares is clearly a disadvantage: This is the case specially if there exist large spurious variations in the data (perhaps caused by undetected outliers, etc.) — such samples are emphasized excessively in the model construction because of the error-squared criterion.

Often errors in different variables are more critical than in others; however, the error-squared criterion assumes that all errors are equally significant. It is the user's task to assure that this equality assumption is justified; this can be carried out by appropriate scaling of the variables during the preprocessing. If the variables are scaled up, also the errors in those variables are emphasized accordingly.

2.4 Tackling with real-world data

Gaussianity assumption is well-motivated, due to the Central Limit Theorem. However, despite the above optimistim, the things are not so simple in practice.

The real measurement data seldom is purely Gaussian. There are various reasons for this: First, normally distributed data that goes through a nonlinear element is no more Gaussian; second, the measurement samples may be generated by different underlying processes, constituting no single distribution at all. All these phenomena can be explained as different kinds of nonlinearities in the system. If the Gaussianity assumption has to be abandoned, what kind of model structure to adopt instead?

The selection of the model structure is always a compromise between two things: The model should fit the data well, but, at the same time, the model should suit the user's needs, being easily applicable and analyzable. The first of the objectives — matching the data — generally means that complex models should be used, but, on the other hand, the second objective favors overall simplicity. There are no final truths available here, but it turns out that a nice conceptual compromise between real data properties and theoretical preferences is given by the *Gaussian mixture model* of data.

2.4.1 Gaussian mixture models

Non-Gaussianity of a distribution is a symptom of nonlinearity somewhere along the data generation processes. As it was observed in the previous chapter, nonlinearity in high dimensions is a problem defying analyses. But assuming that the nonlinearities can be locally linearized, the function can approximately be substituted with a set of linear functions — and the complex distribution can approximately be substituted with a set of appropriately located Gaussians. Such a collection of Gaussian subdistributions is called *Gaussian mixture model.* Assuming smoothness of functions, nearby samples are related; but the farther apart in the data space the samples are, the less they are assumed to be related, or assumed to contribute to the same model: Thus, there are different (linear) submodels for different clusters. In Fig. 2.1, there exist, say, 4 or 5 data clusters; It is also assumed here that these subdistributions can be approximately characterized by Gaussians.

It seems that the typical nonlinearities can be attacked using a two-level strategy: First, find the set of appropriate clusters Γ, whatever phenomenon has given rise to such clustering, and, after that, apply linear methods for modeling within each of the clusters $c \in \Gamma$ separately.

However, when constructing regression models, one is not interested in the clusters, but one would like to have (continuous) mappings between variables. Gaussianity (or, indeed, any compact distribution model) as the model for subdistributions gives a consistent way of getting back from discretized (clustered) coding of data to smooth and continuous (nonlinear) input/output functions. Assume that $p_c(\kappa)$ is the probability of sample number κ to belong in the subdistribution κ, as revealed by (2.1), with mean $\bar{v}_c$ and covariance R_c determined using the samples belonging to that distribution, and assume that $\hat{y}_c$ is the output estimate determined by the cluster c. Then, the maximum likelihood estimate that combines the clusterwise sub-estimates in a probabilistically reasonable way, weighting the individual estimates by the appropriate probability, is given by

$$\hat{y}(\kappa) = \sum_{c \in \Gamma} \frac{p_c(\kappa)}{\sum_{c' \in \Gamma} p_{c'}} \hat{y}_c(\kappa) \tag{2.25}$$

The normalization factor in the denominator is needed to assure that the total probability of the sample to belong to some of the clusters is 1.

It is clear that if data only is available, determination of the cluster structure is a difficult task ... In Appendix A, some (more or less heuristic) approaches to determining the cluster structure are presented.

2.4.2 Example: Types of "Natural Data"

The class of nonlinear functions in real processes is hopelessly large, and capturing all alternative behavioral patterns within a single model structure is not possible. However, it turns out that just a few special types of nonlinearities usually exist in measurement data, and these classes of nonlinearity can nicely be captured by the Gaussian mixture model (see 23]). Let us study little closer those nonlinearities that we would assume to detect in a typical system to be modeled. The first type of structural nonlinearities is reflected as *separate clusters* (see Fig. 2.7). During different periods, different conditions in the process apply (sometimes a pump is on, sometimes it is off; sometimes ore is coming from one mine, sometimes from another mine, etc.), and the qualitatively differing process conditions are typically seen in the data in a specific way, the samples being clustered around the cluster centers. Within the operating regimes, however, no structural changes take place, meaning that within the clusters the Gaussianity assumption holds. This means that linear analysis can be carried out for each cluster separately.

The second typical source of distribution non-Gaussianity are the *continuous nonlinearities* (see Fig. 2.8). It is common in practice that this kind of behavior is approximated using piecewise linearized models around the operating points;

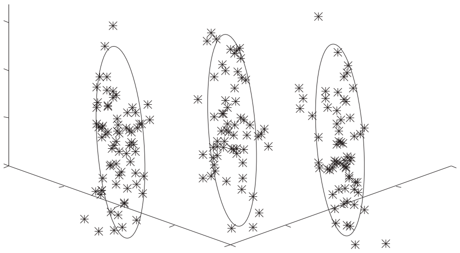

Figure 2.7: Clusters of Type I: Different operating regimes

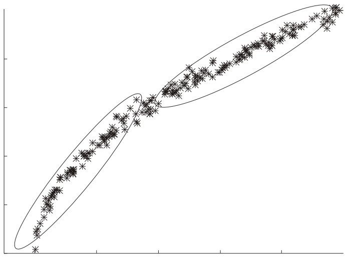

Figure 2.8: Clusters of Type II: Continuous nonlinearities

this means that separating data in clusters and modeling each operating point separately, useful models are again reached. It is a nice coincidence that this piecewise linearity approach is also well compatible with current engineering practices: Smooth nonlinearities are typically linearized around the operating points in control engineering models.

2.4.3 Outliers

A rather special reason giving rise to separate degenerate data clusters is the existence of *outliers* in the data. Outliers are more or less “lonely” samples, being typically not caused by real process originated phenomena but by spurious measurement errors, device or communication failures, etc. Often outliers are located alone far from other samples. However, the normal distribution extends to infinity — there exist no straightforward criteria for distinguishing between valid and outlier samples, and it is more or less visual inspection by a domain-area expert that is needed.

Because it is the error squared criterion that is typically used in modeling, samples far from the more typical ones have a considerable effect on the subsequent modeling. There are two opposite risks:

1. Including outliers among the modeling data may totally ruin the model, the far-away sample dominating in the final model.
2. On the other hand, too cautious selection of samples, neglecting too many samples, also affects the final model: It is those samples that are far from others that carry the most of the fresh information — of course, *assuming* that these samples carry information rather than disinformation.

As all clusters seemingly existing in the data should be checked separately to assess their overall validity, this is specially true in the case of outliers. Detecting outliers is knowledge-intensive, and special expertise on the domain-area, measurement devices, etc., is needed. Often a missing measurement variable is replaced by the measurement machinery by zero (or some other predetermined value), and such outliers can easily be detected, but this is not always the case.

Typically, if there is no scarcity of data, sample vectors with missing values can be simply ignored and eliminated from the data set. If only some of the measurements are missing, all other measurements within a sample being valid, however, it may be reasonable to utilize that sample anyway: Then, the missing values have to be somehow fixed before the sample is used (see "missing values" in Appendix B).

2.5 Excursion: Networks and *power law*

Some of the "hottest" areas of research — like *chaos* and *complexity theory* — seem to be very far from the age-old statistical approaches. Specially, linearity seems to be completely out of the question: Interesting behaviors emerge only in nonlinear environments. However, looking the applications in more detail, it seems that there are connections.

It has been observed that there exist peculiar similarities among very different kinds of complex systems. For example, it has been claimed [**?**] that distributions in self-organized complex networks follow the *power law,* that is, there generally holds

$$y = cx^f \tag{2.26}$$

for scalars y, x, and constant f. Here, x stands for the free variable, and y is some emergent phenomenon related to the probability distribution of x; for example, if x is the "ranking of an Internet page", and y represents "number of visits per time instant", the dependency between these variables follows power law: There are some very popular pages, whereas there are huge numbers of seldom visited pages. As compared to Gaussian distribution, the power law distribution has "long tails"; the distribution does not decay so fast[1].

In the multivariate spirit, one can extend the single-variable formula (2.26) by including more variables; if there is only one variable x_i changing at a time, the new formula corresponds to a set of n simultaneous power laws:

$$y = x_1^{f_1} \cdot \cdots \cdot x_n^{f_n}. \tag{2.28}$$

Now, if one takes logarithm on both sides of the formula, one has

$$\log y = f_1 \log x_1 + \cdots + f_n \log x_n, \tag{2.29}$$

[1] It is interesting to note that the power law distribution is closely related to another modern concept, namely *fractal dimension.* Assuming that the variable x represents some kind of "yardstick", determining the scale factor, and y represents the level of *self-similarity,* so that when one zooms the original pattern by the factor of $1/x$, there exist y copies of the original pattern (and this zooming process can be repeated infinitely), the fractal dimension of that pattern can be defined as

$$\dim = \frac{\log y}{\log x}. \tag{2.27}$$

When the pattern is simple, this definition coincides with the traditional ideas concerning dimension, but for complex patterns, non-integer dimensions can exist. Now, it is easy to see that, after taking logarithms, the parameter f in (2.26) closely corresponds to the fractal dimension for the networked system or

$$y' = f_1\, x_1' + \cdots + f_n\, x_n' + c, \tag{2.30}$$

where $x_i' = \log x_i$, etc. It turns out that the multiplicative dependency has become globally linear — by only preprocessing the variables appropriately. There are also other approaches towards reaching a linear (local) model structure: Differentiate (2.29) around the nominal values $\bar{x}_i$, so that there holds

$$\left(\frac{\Delta y}{\bar{y}}\right) = f_1 \left(\frac{\Delta x_1}{\bar{x}_1}\right) \cdot \quad \cdots \quad \cdot f_n \left(\frac{\Delta x_n}{\bar{x}_n}\right). \tag{2.31}$$

Now the variables $\Delta x_i / \bar{x}_i$ are the relative deviations from the nominal state. This kind of variables are assumedly more robust that the log-variables. It is evident that very much can be done by appropriately conditioning the data; these issues are studied closer in the next chapter.

As a final note here, study the outlook of the *multivariate fractal distribution.* variable y' in (2.30) is a sum of assumedly large number of assumedly independent stochastic variables $f_1 x_1'$. Because nothing more accurately about these variables is known, it can be assumed (again according to the Central Limit Theorem) that $y' = \log y$ has normal distribution:

$$p(\log y) = \frac{1}{\sqrt{2\pi}\sigma}\ \exp\left(-\left(\log y - \mu\right)^2 / 2\sigma^2\right). \tag{2.32}$$

Taking logarithms,

$$\log(p(\log y)) = c - \left(\log y - \mu\right)^2 / 2\sigma^2. \tag{2.33}$$

This means that the multivariate fractal distribution is *parabolic* rather than linear on the log/log axis, the three parameters being c, μ, and σ^2. Indeed, this is in conflict with “traditional modern” network intuition!

Computer exercises

1. Try the `dataClust` command in the `Regression Toolbox`. Define one-dimensional data of two Gaussian clusters, both containing 1000 samples and centers being 10 units apart, with the command

   ```
   [X] = dataClust(1,2,1000,10);
   hist(X,50);
   ```

   Modify data, summing variables that have this same distribution:

   ```
   X = X + X(randperm(length(X)));
   hist(X,50);
   ```

   Repeat the above steps sufficiently many times. What happens with the data clusters? Why natural data still typically is clustered — what is the difference in the data production processes?

2. Search for examples of observed distributions in complex networks that have been published in Internet. Applying some search engine, use keywords like

   ```
   power law distribution
   fractal dimension
   ```

   Study the distributions; observe how the claimed linear dependencies on the log/log scales (as resulting from the single-variable fractal dependency) can often indeed better be matched against a parabola (as resulting from the multivariate fractal dependency assumption).