Level 6

Structures of Information beyond *Differentiation*

The key concept in cybernetic systems is information, availability of information determining the models that are constructed. Assumption of environment-orientedness means that it is the information coming from the environment that dictates the results in a more or less unique way.

Despite the assumed uniqueness there still are many ways how the world can be seen and how this view can be interpreted. As the neocybernetic models are based on observed correlation structures, by appropriate scaling of the variables one can implement continuous modifications to the information that is visible to the system. This all is familiar from principal component analysis. However, here the goal is to extend beyond the existent intuitions: What happens when the amount of available information increases? How can the *emergence of structures* be understood?

6.1 Towards more and more information

Being based on principal components, neocybernetic model is robust against high dimensionality. To assure maximum information availability, a reasonable strategy is to *include all available data among the measurements* — the modeling machinery can automatically select the relevant pieces of information. When the data dimension becomes high, there are also qualitative and theoretical benefits.

6.1.1 About optimality and linearity

Thinking holistically is a comprehensive challenge. For example, one should no assume that there is some centralized optimization criterion being reached for by the system. But if the data dimension is high enough, a common goal is a useful abstraction: It turns out that *optimality become reducible.*

The most straightforward way to assure the supply of information is to inflate the space of input variables, so that m, the dimension of input data, grows. To

analyze this issue, assume that the cost criterion can be locally decomposed so that its differential change can be expressed as a sum of N weighted parts:

$$\delta J = c_1 \delta J_1 + \cdots + c_N \delta J_N \tag{6.1}$$

Here the sub-criteria are assumed to be locally linearizable, so that

$$\delta J_i = Q_i^T \, \delta u \tag{6.2}$$

for some parameter vector Q_i and variable vector u. If the sub-criteria are independent, for high number of variables there holds for correlations among different vectors i and j

$$\frac{Q_i^T \, Q_j}{m} \quad \rightarrow \quad 0, \qquad \text{as} \qquad m \quad \rightarrow \quad \infty. \tag{6.3}$$

The more there exist variables, the better random vectors become orthogonal. When solving for gradients, one has

$$\frac{\delta J_i}{\delta u} = Q_i \tag{6.4}$$

so that

$$\frac{\delta J}{\delta u} = c_1 Q_1 + \cdots + c_N Q_N. \tag{6.5}$$

Now, assuming that the variables are adapted along the negative gradient of some sub-criterion, so that $\Delta u = -\gamma Q_i$, the global criterion also goes down:

$$\Delta u^T \frac{\delta J}{\delta u} = -\gamma Q_i^T \, (c_1 Q_1 + \cdots + c_N Q_N) \approx -c_i \, Q_i^T Q_i < 0. \tag{6.6}$$

This means that if the sub-criteria are mutually independent, and if the input data dimension is high enough, the task of multi-objective optimization can be decomposed. Local optimizations result in global optimization.

What is more, when the data dimension is high, getting stuck in local minima is less probable. Multiple variables typically mean better continuity in the data space, and perhaps also evolutionary processes can be characterized in terms of "generalized diffusion". How about the cost criterion (6.1) then — is it not unrealistic to assume linear additivity of the sub-criteria? Again, it is high dimensionality that helps to avoid problems. The more there are features (variables) available, the more probable it is that the problem becomes more linear (compare to the idea of *Support Vector Machines,* where a complex classification problem is changed into a simple problem in high dimension).

6.1.2 New sensors and innovations

When trying to affect the modeling results, selection of variables to be included in input data is the most important decision. How to assure high dimensionality and fresh information in the data, where to find the new sources of observations?

New innovations and new sensors are needed by the system — the term "sensor" being used in a relaxed sense here, as the information capture is to be seen in the holistic perspective. It does not matter what is the physical manifestation of the sensors, as long as the acquired information can cumulate in the model structures. Some examples are given here.

- **Spatial distribution** can be utilized, that is, information from spatially distinct locations can be used. This far it has been assumed that a system is isolated — however, in a real ecosystem, neighboring systems are in close connection, and they can be modeled as a whole. Specially, assuming that there are no limitations for seeds to spread within some area (or no limitations for information flow), the spatial structure can be "collapsed", assuming that the spatial distribution delivers relevant material about the ecosystem in general. This can be utilized when constructing the covariance matrices: Plentiful fresh data and variation is available when each subregion within the ecosystem delivers its contribution to the behaviors of the environmental variables.
- **Temporal distribution** can also be utilized, that is, information from temporally distinct time points can be used. Assuming that a species in an ecosystem has some (hard-coded) memory, it is not only the current state of the environment that is seen by the population, but also the time history: If the previous year was bad, the population is lower this year, no matter what are the current circumstances. The longer-living the individuals of a population are, the longer is the "memory", too. When cybernetic models are constructed for such time-series data, it is no more simple PCA that is being carried out; it is dynamic modeling in the framework of (implicit) *subspace identification* [60]. It can be assumed that if the food level variations are low, then — after adaptation — the environment seems to support longer-living species. Is it because of this that predators live longer than prey, the information being filtered more on the higher trophic layers?

It turns out that the more there are variables, and the more there are possible variable combinations — and the more there are ways to select the "interesting" or most relevant features, different selections resulting in different models and different views of the world. This is a special challenge in constructivistic systems, where the space of candidate variables is potentially infinite; in psychology, one speaks of the *Barnum effect,* meaning that when there are enough degrees of freedom, any model can be matched against the data (making numerological studies, for example, often astonishing).

6.1.3 Example: Transformations implemented by nature

Frequency domain was employed in the previous chapter to study information distribution among subsystems. But such considerations are applicable not only at the ecosystem level — it seems that also within a single individual similar analyses are appropriate. Specially in the processing of auditory and visual information clever data preprocessing is needed to extract fresh features from the temporal and spatial data. Again, it is a nice coincidence that there are

powerful mathematical tools available for analysis and understanding of such features.

When auditory, time-domain signals are received, the cilia in the inner ear implement spectral analysis: Depending on the frequency, sound waves can penetrate different distances in the cochlea. As the cilia are connected to the auditory cortex, energy in each frequency range becomes an input signal of its own, the number of inputs thus becoming expanded. What the brain then can see in the preprocessed signals is combinations of formants; this means that the patterns being modeled are *phonemes.*

It seems that similar frequency-domain reconstruction of signals takes place also when visual signals are processed; however, now the information is not distributed temporally but spatially. Simple networks of neurons can implement (two-dimensional) discrete Fourier transform. This kind of coding of the images is beneficial because cross-correlation between two transformed images efficiently reveals the dislocations and structured differences among the images. For example, movements within the field of vision are manifested when successive transformed images are compared; on the other hand, depth cues become available when using image pairs acquired from nonidentical locations (from the two eyes). The succession of parallel / sequential transformed image vectors is interpreted as input data samples; when the correlation structures among data are modeled in the neocybernetic spirit, the resulting sparse components (see later) perhaps reveal natural-like decomposition of visual patterns. This kind of extra information concerning spatial dependencies among visual entities can perhaps explain the properties of three-dimensional vision.

6.2 Blockages of information

When there is plenty of data available, not all need to be used. Here, some examples are given how the results can be controlled by explicit channeling of information, by explicitly determining structures of data flow. In a sense, it is all about implementing non-idealities again — the ranges of seeing information are limited.

6.2.1 Hierarchic models

As an example, study a cybernetic system with the following model matrices (assume "clever agents"):

$$A = \begin{pmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} \cdot & & \\ & \cdot & \\ & & \cdot \\ & & \cdot \end{pmatrix}. \tag{6.7}$$

Dots in the structures mean that those connections are non-zero, whereas empty slots denote missing connection. The above B matrix form can be appropriate in sensor/actuator structures, where each actor has its own measurements. There is no complete information available, and data flow becomes localized. The

nonideal flow of information introduces distortion in the data, and the analyses in chapter 3 become outdated: The degrees of freedom in input data are no more a limiting factor, non-trivial structure emerges even though $n = m$. Closer analysis reveals that the basis vector ϕ_i is dominated by the local measurements u_i.

More interesting results are found if one has triangular interaction matrix, each actor only seeing the actors in front of it, the last actor being capable of seeing all information. The structure becomes strictly hierarchic:

$$A = \begin{pmatrix} \cdot & & \\ \cdot & \cdot & \\ \cdot & \cdot & \cdot \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{pmatrix}. \tag{6.8}$$

This means that the first variable is not affected by the other ones — it lives a life of its own, exploiting all the information that is available in input. Thus, it alone constructs a principal subspace model of the data; because this model is one-dimensional, it basis vector must coincide with the first principal component axis. In this sense, the first variable implements (trivial) principal component analysis rather than principal subspace analysis. Such reasoning can be recursively continued: The second variable is affected only by the first variable representing the first principal component, meaning that its contribution is deflated from the data. This way, looking at the second variable alone, it is the *second* principal component that must be represented by it. And the same analysis can be continued until the variable n, meaning that the hierarchic structure implements explicit principal component analysis. Because of the information blockages, principal components get separated, and structure emerges.

6.2.2 "Clever agent algorithm"

Implementing an algorithm is a compromise between theoretical and practical aspects. Now it seems that the nonideality — triangular blockage of information, as motivated above — enhances convergence, as the variables disturb each other less. It turns out that the Hebbian/anti-Hebbian adaptation becomes a useful PCA algorithm, as it is robust — there are few free parameters — and because the explicit construction of the covariance matrix $\mathrm{E}\{uu^T\}$ is avoided: In the cybernetic cases, m is typically high, and the covariance matrix can be huge.

In `Matlab` syntax one can write the "vanilla" algorithm as showqn in Fig. 6.2 (matrix `U` containing the k sample vectors u^T as rows, and matrix `Xbar` containing the k internal variable vectors $\bar{x}^T$ as rows).

The data structures `Exx` and `Exu` are initialized to small values (matrix `Exx` having to remain positive definite at any time). The parameter λ determines the adaptation rate. After convergence, the basis vectors can be picked out from the matrix $\phi^T = \mathrm{E}\{\bar{x}\bar{x}^T\}^{-1}\mathrm{E}\{\bar{x}u^T\}$.

As an example, a case of coding hand-written digits is represented. As data material, there were over 8000 samples of handwritten digits (see Fig. 6.1) written in a grid of 32×32 intensity values [50]. The 1024-dimensional intensity vectors were used as data u, and the algorithm was iterated until convergence. The results are shown in Fig. 6.3.

Figure 6.1: Examples of handwritten digits

```
while ITERATE

  % Balance of latent variables
  Xbar = U * (inv(Exx)*Exu)';

  % Model adaptation
  Exu = lambda*Exu + (1-lambda)*Xbar'*U/k;
  Exx = lambda*Exx + (1-lambda)*Xbar'*Xbar/k;

  % PCA rather than PSA
  Exx = tril(ones(n,n)).*Exx;

end
```

Figure 6.2: **Algorithm 1:** Hebbian/anti-Hebbian PCA by "intelligent agents"

6.2.3 On-line selection of information

There are information flows and blockages on many levels in an adapting system, and frequency-domain characterizations are possible here, too. The slowest-scale control of information takes place in the adaptation processes: For example, gene pools that restrict information to remain within the species implement en extreme block for spreading of information. The results become visible as peculiar evolutionary developments on the species level.

In the other extreme end, the information blockages can also be very temporary. For example, the routing of information can be dependent of the actual signal properties — meaning that the signal path is nonlinear. As seen from the opposite point of view, it can be said that nonlinearities are information filters.

Linearity means homogeneity and predictability, whereas nonlinearity is the key to emerging differentiation among structures. When dropping the assumption of linearity, the strong guidelines are lost: There is an infinite number of possible nonlinearities available, and there is no general theory to understand the

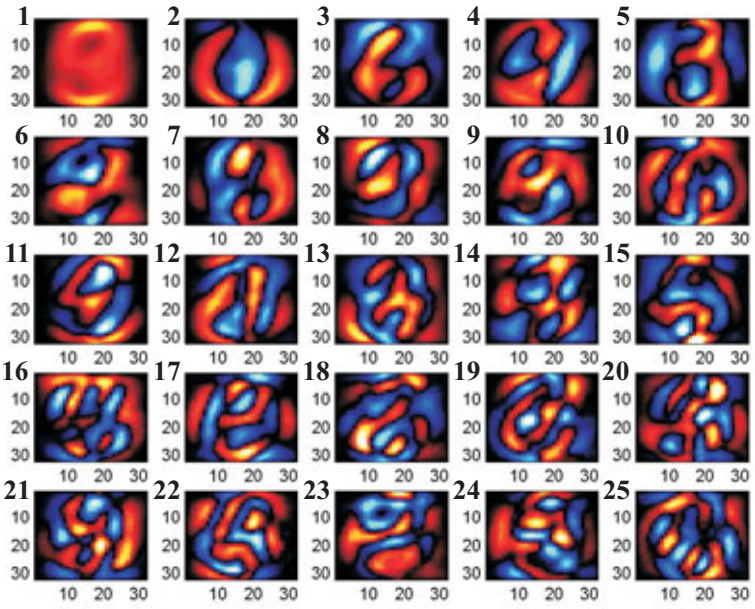

Figure 6.3: The 25 principal components extracted from the handwritten digits. The 1024 dimensional feature vectors have been decoded as planar patterns to reveal their connection to input data properties (dark regions mean that there is no correlation with the feature and the corresponding pixels; light blue regions denote high negative correlation, and light red denote high positive correlation). Because of the hierarchically structured feature extraction, the sparse subspace has been decomposed into the actual PCA basis vectors: First, there is the mean vector, and thereafter the correlation structures are presented in the mathematically motivated decreasing order. The coding is efficient when there is scarcity of latent variables, but the physical relevance of the features is questionable when the basis dimension becomes large

resulting functionalities. What kind of nonlinearity to choose, then? It turns out a good compromise is a function that implements a volatile *switch.*

$$f_{\text{cut,i}}(x) = \begin{cases} x_i, & \text{if } x_i > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{6.9}$$

This *cut function* (see Fig. 6.4) lets positive signals go directly through, but eliminates negative ones. This function is piecewise linear — this offers theoretical benefits as between the transition regions linear model structures are applicable. There exist also strong physical motivations for this selection of nonlinearity: Whatever are the signal carriers — concentrations, frequencies, agent activities — such activities can never become negative. In more complex cases, for example, when modeling gene activation, the cut function is still applicable: Remember that there are excitatory and inhibitory transcription factors

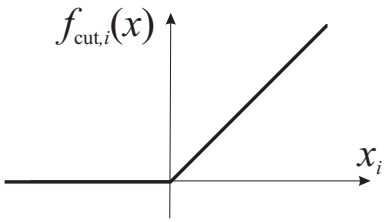

Figure 6.4: Nonlinearity for online-blocking of information

controlling the process; there must be excess of excitation to start the process in the first place, and the more there is excess, the more chromatin packing of DNA opens up to promote gene expression.

Such a simple form of nonlinearity makes it possible to implement "soft" transients between structures. When a variable becomes active, a new dimension in the data space becomes visible. As the nonlinearity is monotonic and (mostly) smooth, optimization in pattern matching can thus take place among structures.

As explained in [92], locally unstable models become possible because of the nonlinearity: Extreme growth in variables is limited by the cut functionality. When combined with a dynamic model, it is possible to implement bistable "flip-flops", where minor differences in initial states or in the environment result in completely differing outcomes. When comparing to natural systems, only the stem cells are assumedly free of such imprinting; in practice, the evolved "epigenetic states" can be very stable after such a development has started (for the coloring of animal fur, see [81]). These peculiarities that are made possible by nonlinear structures are not elaborated on here; the cut nonlinearity will be employed in what follows only to *boost* linearity.

6.2.4 Switches and flip-flops

To see how the nonlinearity can affect the originally linear and well-understood smooth behaviors, an example is needed. Assume that the "cut" function is included in the system model so that one has

$$\frac{d\,\xi}{d\,t}(t) = A\,x(t) + B\,u, \tag{6.10}$$

where the visible activities are limited to positive values:

$$x(t) = f_{\mathrm{cut}}(\xi(t)). \tag{6.11}$$

Applying this model structure, a "comparator system" was simulated with two mutually inhibitory subsystems:

$$\begin{pmatrix} \dot{\xi}_1(t) \\ \dot{\xi}_2(t) \end{pmatrix} = \begin{pmatrix} -\gamma_1 & -1 \\ -1 & -\gamma_2 \end{pmatrix} \cdot \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} \gamma_1 & 0 \\ 0 & \gamma_2 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}. \tag{6.12}$$

The negative non-diagonal elements in A matrix implement negative feedback among the subsystems. In simulations, $\gamma_1 = \gamma_2 = 0.75$; this means that the eigenvalues of the matrix A are $\lambda_{1,2} = \gamma_{1,2} \pm 1$ or $\lambda_1 = 1.75$ and $\lambda_2 = -0.25$,

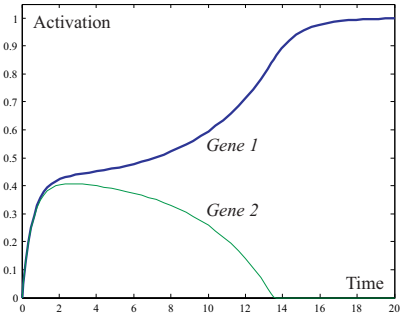

Figure 6.5: Incoming concentration ratio $u_1/u_2 = 1.00/0.99$

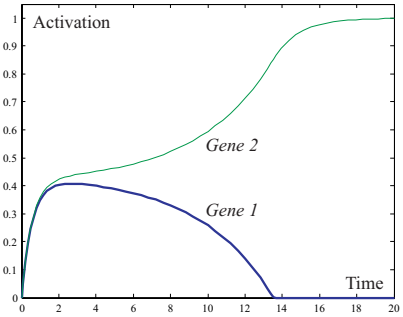

Figure 6.6: Incoming concentration ratio $u_1/u_2 = 0.99/1.00$

and the linear system without the additional nonlinearity would be unstable, x_1 and x_2 escaping to infinity, the one in positive and the other in negative direction. The variables escaping in opposite directions "pump" each other; as the nonlinearity prevents variables from escaping in negative direction, it simultaneously stabilizes the positive variable as well.

The simulation results (starting from zero initial values) are shown in Figs. 6.5 and 6.6. It seems that in this framework inhibition and excitation together define a system where some variables stabilize to non-zero values and other to zeroes ("winner-take-all"), depending on the input value distribution: Using the above model, x_1 wins and x_2 vanishes altogether if $u_1 > u_2$, and vice versa, the inputs being constant. It turns out that, qualitatively, the behavior is rather robust regardless of the exact parameter values.

The presented model structure makes it possible, for example, to define a genetic functional "state". Remember that the gene expression is controlled by specific inhibitory and excitatory transcription factors, these transcription factors forming a complex network, all of them being products of the activity of other genes. Minor changes in input concentrations make the resulting environment within the cell completely different: The "flip-flops" take either of the alternative values depending of the ratio between inputs, and once they have ended in some state, it is difficult to change it. In this sense, associations to properties of stem cells are easily made: A cell that has specialized cannot any more take some other role. Other bonus intuitions are also available: Today, there is the link missing between strictly biophysical considerations and qualitative ones. The purely numeric, quantitative, continuous approaches and the qualitative and discontinuous approaches are incompatible. The claim here is that the presented model makes it possible to study *emergence of structures.*

In the framework that is boosted with nonlinearity, competition among agents can be intensified: Effects of substructures can be wiped away altogether. Such extreme behavior is only possible in nonlinear systems because it is due to the nonlinearities that the system remains stable. Indeed, there emerge alternative minima depending on the initial values. In complex cybernetic systems, mastering such local solutions is of utmost importance, and rather than studying individual nonlinearities, a higher-level view is needed.

6.3 Real world of nonlinearity

The basic neocybernetic model is linear. This was a reasonable starting point as there is no strong theory for nonlinear systems. Linear structures are, after all, always easy, as there exist unique minima in a given environment; for nonlinear systems this does not hold, and typically the results are not identical even if the environment remains the same. However, nonlinearity is the nature of the real world, and when the objective is to model it, the modeling machineries have to accept this fact. So, how to characterize nonlinearities, and, specially, how have the real systems managed to do that?

6.3.1 What is relevant, what is reasonable

This far, the method has been the starting point, and its properties have been examined; however, now concentrate on the applications. Now there is the whole wide world ahead of us, the class of nonlinear functions being infinite and indefinite, and one should be careful not to open the *Pandora's box.*

To have a balanced view of the problem and the possible ways to attack it, one can utilize the above discussions, and *exploit the cybernetic model of an existing memetic system:* Ideas have been competing in bright minds, and an equilibrium can be observed. In the spirit of the Delphi method [25], different arguments have been thoroughly discussed by experts — in the field of *artificial neural networks* the problems of capturing "natural data" have been intensively studied from different points of view (for example, see [36]). This ANN research is a well-established branch where compromises have been found between what seems promising from the point of view of representing natural data and what seems possible and practical from the point of view of available tools, and today, a "model of models" can be compiled: What are the dimensions of the problem, what are the interesting applications and promising methodologies. Within such a memetic "supersystem" the degrees of memetic freedom are manifested, helping to see the "intra-paradigm paradigms", combinations of aspirations and visions, where different points of view are weighted in different ways.

Some advantages can compensate other disadvantages. For example, despite the theoretical deficiencies, there is so much physiological and mathematical support for linear structures that today there exists a large body of literature, and still there is active research in that direction. As a paradigmatic approach, there are different kinds of algorithms to implement principal component analysis, and, further, there are different kinds of extensions to the basic models (see [26]). These studies are motivated by the physiological studies concerning the Hebbian neurons, and they are further boosted by the strong theoretical intuitions and interesting applications: The research is still going strong specially in the field of *independent component analysis* [41].

Another family of intuitions have motivated the study of *feedforward perceptron networks:* It has been observed that within this model structure, all smooth nonlinearities can be approximated, at least in principle. In practice, this unlimited expressional power is a problem: To select among the alternative functional structures and to determine the parameters within the selected structure, there is need for very high numbers of data. Often some additional assumptions are

(implicitly) employed for pragmatic reasons — for example, typically one limits the search to smooth mappings. There is also another class of model structures with equally high expressional power: The *radial basis function networks* are based on basis functions, simple prototype functions of localized support (like Gaussians); when such functions are scaled and scattered appropriately in the data space, their combined envelope can again be matched with any smooth function. As compared to the perceptron networks, the basis function networks are better manageable as the representation there is more local and easier to interpret.

The above ANN structures have continuous output, and they can be applied for function approximation; a more special application is *pattern recognition,* where one only needs discrete output. There is a very special network architecture that deserves to be mentioned here because of its close relation to the neocybernetic discussions concerning dynamic models and balances: In *Hopfield networks* the input is given as an initial state to the system, and a dynamic process searches for the minimum of the energy function, revealing the pattern that is nearest to the input. The construction of the network is such that it assures that the attractors of the dynamic process are the stored patterns. However, as compared to the neocybernetic model structure, now there is no input; the end result is unique after the initial state (the incomplete pattern to be completed) is given.

All of the above neural network structures are mathematically rather involved; in the other extreme, there are the intuition-oriented approaches where it is the actual brain structures and functionalities that one tries to reproduce. One of such intuitions concerns brain maps: The mental functions have their own locations in the brain, related functions and patterns assumedly being stored near each other. The *self-organizing maps* try to mimic the formation such (two-dimensional) maps [46]. There are many applications what comes to data visualization: On the SOM map the high-dimensional data distributions are often made better comprehensible. As the high-dimensional real-valued vectors are coded in terms of N integers (map nodes) only, there is extreme data compression, and information loss cannot be avoided. The most interesting issue about the SOM is that in some sense it seems to match our mental structures — perhaps there are lessons to be learned here (see chapter 7).

It seems that all ANN methods attack only one issue at a time. To address different needs, a compromise is needed; and it can be claimed that the neocybernetic model can be extended to combine the ideas of basis functions, dynamic attractors, and intuitive considerations, combining comprehensibility and expressional power in the same framework.

6.3.2 Models over local minima

For a moment, it is beneficial to look at modeling in the probabilistic perspective. When seen in the probabilistic framework, the goal of a model is to capture the data distribution, the model explaining as economically as possible where an individual data sample is located in the data space.

How can the neocybernetic model be characterized in terms of distributions? It is not the degrees of freedom alone (as studied in chapter 2) that would capture the variable distributions; when elasticity is also taken into account, tensions

pulling the system towards balance, samples tend to become clustered around the nominal state. Assuming that the scores have normal distribution (being results of many independent equally distributed random variables being added together), and assuming that the basis axes are mutually independent, one could use the *multivariate normal (Gaussian) distribution spanned by the degrees of freedom* as representing the behaviors of cybernetic variables. However, natural data is *multimodal,* it cannot be represented by a single one-peaked (Gaussian) distribution — but an arbitrary smooth distribution can be approximated as a combination of (Gaussian) sub-distributions. Together the candidates define a basis, so that (if there is enough of them and they are appropriately combined) one can implement a *mixture model.* Strictly distinct clusters are implemented if the representation is *sparse* (see below).

Thus, the radial basis function metaphor would be applicable here; however, the structure also suggests more appropriate interpretations. Because the basis functions are now linear, the vectors ϕ_i determining the basis functions through the dot product operation, so that the matching against the input is calculated as $\phi_i^T u$, the basis functions have infinite support and there is no finite maximum. One does not only have a mixture of basis vectors that determine the distributions, one has "basis subspaces", determining the directional components present in data. The structural components define *feature axes* to be exploited by the higher-level model.

This far the model has been assumed linear. If the representation is *sparsely coded,* so that only a subset of features is employed at a time (see 6.4), the contributions of some features (the least significant ones) being cut to zero, there emerge structural alternatives, not all submodels sharing the same components. The sparse coded structure, where the substructures are linear, becomes *piecewise linear.* When the active components vary, there exists a wealth of candidate structures. Out of the n available features, in principle one can in the sparse coded case construct as many structurally differing distributions as there exist partitionings of the n variables between active and inactive ones. For large n this becomes a huge number. Such wealth of distributions is difficult to visualize: The sub-distributions are not clusters in distinct locations, and, indeed, one should not think using intuitions from low dimensions. What does this kind of a world look like, is elaborated on in chapter 7. In any case, such sharing of features is versatile, and it helps to reach generality and efficiency of coding; from now on such mixture of linear submodels is assumed as the prototypical model when the strictly linear models no more suffice.

When the mixture metaphor is employed as the basis of modeling, some extensions to the adopted model framework are needed; in a complex hierarchic system it is not only the highest level that is assumed to be nonlinear, but the model extension needs to be applied fractally. Before, the models were based on the linear features determined by vectors ϕ_i, and stacking of submodels was straightforward, linear structures being directly summable. Now one needs to extend to *nonlinear features:* When seen from above, the mixture model also defines a "feature" to be exploited by the higher-level model. To facilitate this, to make the extended model compatible with the linear model, the mixture model needs to look the same as the simple one, when seen from outside. The "interface" of the simple model is the latent variable activity, or score of the fea-

ture, in the form $\bar{x}_i = \phi_i^T u$; the submodel only delivers one real number to the outer world, revealing the match of the input data with the submodel. Also the mixture model has to be manifested in the similar manner to the outside world when such a model is further being used as a submodel — how to accomplish this?

The experience with the linear case helps here: The goal of the basic neocybernetic model is optimum match with the environment, and as complete reconstruction capability as possible so that no variation in the input data is lost. The latent variable $\bar{x}_i$ is a measure for how the submodel ϕ_i alone managed in this matching task, or, indeed, how much this submodel was "trusted" in this task, the balance of these latent variables being determined through competition among candidates. When ϕ_i are interpreted as basis functions, the outputs $\bar{x}_i$ represent the matches, or activities of individual, vector $\bar{x}$ revealing the success pattern, determining a coding of the prevailing environment. This view can directly be extended to the nonlinear case. The whole grid model is to be collapsed to one number characterizing the fit with the environment; let this number be called *fitness* of the model[1]. When employing the model, the cybernetic fitness criterion is how well the environment can be modeled, or reconstructed, and this can be expressed in the form $|\hat{u}|^2$, representing the length of the reconstructed input vector when the model is used for its reconstruction. To the outside world, the mixture model thus looks like

$$\bar{x}_i = \phi_i(\bar{u}) = |\hat{u}_i|, \tag{6.13}$$

where $\phi_i(\cdot)$ denotes some scalar-valued function, and $|\hat{u}_i|$ is the contribution of the submodel i in the input reconstruction, when various submodels compete in that task, and when equilibrium has been found. Remember that this "input reconstruction" actually means resource exploitation, making the assumptions about the same goals of subsystems generally justifiable. The value $|\bar{x}|$ becomes zero if the environment cannot be captured at all by the submodels, whereas if there is complete match, the whole variance of the input data is transferred further. It is also variance (average of the reconstruction vector length squared) that is a universal optimality measure in the nonlinear as well as in the linear case. Discussions concerning information, etc., thus remain valid also in the nonlinear case. The presented fitness definition abstracts away the implementation of the submodel, encapsulating it as a black box — indeed, it need not even be based on the presented mixture structure; there are no constraints as long as the model structure has mechanisms of producing the estimate $\hat{u}_i$. This means that the neocybernetic framework offers a general-purpose environment for studying very different kinds of coevolving complex systems.

No communication among submodels is needed: The model becomes balanced just in the same way as in the linear case. No matter how the individual submodels are implemented, they compete with each other, exploiting the resources; better models exhaust the available variation, leaving less resources for others to exploit. The coordination among submodels is again implemented implicitly through the environment, and there is no need for external supervision and "selection of the fittest" as in traditional clusters-based structures, etc. The universal "fitness criterion" is the modeling capacity: How well a (sub)model

[1]Indeed, there is a close connection to *genetic algorithms* here

can explain (and thus exploit) the environment. Trusting one's own observations, or the available remaining resources, makes it possible to implement local adaptation without compromising the emergence of higher-level structures.

To conclude, the mixture models constitute the basis functions for the next-level models. In the linear case it was the vectors ϕ_i that were thought of as characterizing the submodels, $\phi_i^T u$ giving the matches; in the nonlinear case it is $\phi_i(u)$ that returns the submodel activity.

The operation of the cybernetic model is defined through a dynamic process; similarly, the mixture model should be seen as implementing a *set* of such dynamic processes. Each of the submodels that determines a sub-distribution simultaneously determines an *attractor* in the data space, hosting a local minimum of the cost criterion, where the data matching process converges in favorable conditions. The final location of the fixed point within the basin of attraction is dependent of the input data. There are no "strange attractors" or the like, everything is quite traditional, being based on even (locally) linear processes and local balances. The proposed combination of linear and nonlinear structures seems to usually assure unique fixed points in the framework of many basins of attraction, thus combining simplicity and expressional power. However, being such a powerful framework, not very much can be said in general terms about such mixture models; one approach to examine the possibilities, based on simulation, is studied in chapter 8.

Model consisting of multiple attractors — this seems to be an appropriate way to model complex natural systems, too. Remember that nature is working in a distributed way, there is no central design unit: Finding the absolute optimum in a complex environment is just as difficult for nature as it is for humans. Nature is so varied because different solutions have ended in different local minima of the cost criterion. Perhaps a cybernetic model constructing a multiple model characterization over the alternatives better captures the natural diversity of natural systems? The cybernetic model can be seen as a *compressed model optimized over local candidate solutions.* This is a major difference as compared to traditional modeling where it is the only global optimum that is of interest. Remember that many problems of computability theory are concentrated on the NP-hard problems that are practically undecidable in large systems — but rather good local minima are easily found.

6.3.3 How nature does it

The mixture model seemingly has a complex hierarchic structure of submodels. Does such a "model library" need to be stored in some centralized location and maintained by some master mind? The answer, of course, is *no* — nature routinely runs such mixture models in a distributed manner.

Traditionally when deriving clustered basis function models, the key challenge is to determine the locations and the outlooks of the basis functions. Now it is the competitive learning among agents that carries out the matching against the environment in a distributed manner: The basis functions themselves are composed of still simpler basis functions — the agents themselves. When looking at cybernetic systems, it is important to recognize that it is not only one system that is running at a time: It is typically *populations* where there are

individual more or less identical subsystems. This populations-based structure is quite universal, and it applies fractally to all levels of the systems: Within an ecosystem there is the large number of separate species, and within a species there are the individual animals[2]; in an economy there are the companies, and within the companies there are the humans; in a tissue there are the cells, and in the cells there are the chemical molecules.

Nature implements the whole mixture model in a parallel fashion, running the subsystems side by side, and constantly evaluating the performance of them. Optimization in such a structure is completely distributed. Each individual represents a local optimum having adapted to match its local view of the environment; the number of individuals representing a single solution reflects the relative goodness of the solution candidate, a good solution (or niche) being capable of supplying more resources to share. It is the whole set of functionalities that together characterize the nonlinear system of systems — the final mixture model representing a human, for example, being a coordinated-looking composition of the functions of its subsystems. Regardless of the distributed nature of the structure, the non-coordinated submodels can still share common features if there exist statistically consistent properties visible in the environment (see chapter 10).

There is no need of explicit coordination whatsoever — the mixture model is a simple extension of the linear case that was already shown to self-regulate and self-organize. As interactions with the environment are crystallized in the activity patterns, it is the feedbacks through the environment that assumedly again can accomplish the regulation task. What is more, all agents agree upon the goodness criterion — maximum activity and exploitation of the environment — and after that explicit coordination is no more necessary as the structure assumedly emerges from the competition. Whether or not some structure truly emerges in such a system is a difficult question — yet, the practical experience seems to support this hypothesis.

As a more abstract example, think of a formless social or memetic environment where it is difficult to uniquely quantify the structure or the variables. As studied in chapter 4, the neocybernetic view offers an escape here: It is the subjective individuals or individual minds that anchor the environment in the realm of observables. The population of minds determines the outlook of the constructivistic world, or the model for it — and, indeed, without this model the world itself would not exist!

The seemingly inaccurate and non-optimal mechanisms of representing the properties of individuals — genes in a biological system, and memes in a memetic one — seems to be nature's way to assure that not all submodels can end in the same local optimum. When there is no continuity among representations, separate individuals more probably produce different outcomes, ending in separate local minima of the cost criterion. Differences in genes span new directions in the high-dimensional property space, mutations perhaps augmenting this space, introducing new functionalities. Yet, there is some continuity, as it is the combination of the parent's genes that characterizes the offspring, mak-

[2]The ecosystem consists directly of the individual animals — the level of "species" is motivated for pragmatic reasons, because the spread of genetic information is limited by the species boundaries

ing the mechanisms of property inheritance more continuous, facilitating some level of simple parameter tuning even within a fixed structural framework. The genotype just determines the framework, and it is the dynamic interplay among the system and its environment that determines the outlook of the final phenotype. On the other hand, as the personal catastrophes (deaths of individuals) are non-synchronized, statistical properties of the population do not change abruptly, and the adaptation of the population becomes smoother. The nonlinear environment becomes modeled by local attractors; in a converged model the submodels are rather densely located, exhausting the information available in the environment more or less continuously.

The Darwinian mechanisms that come here to play to exploit the submodels, implementing the adaptation of the population level mixture model, good solutions among submodels being promoted in the mixture. The basic structure of an individual is determined by the genes, and within that framework, the familiar neocybernetic adaptation processes assumedly have tuned the parameters so that it is the best that can be achieved within that framework, so that the structures, and thus the underlying gene combinations, can be compared in an objective way. However, the idea of "survival of the fittest" is not so categorical as it is normally thought to be: Best solutions dominate, yes, but the outperformed ones also can survive, making the view of the reality more complete. Indeed, samples far from the mainstream solutions can carry very much valuable information. There are no outliers among the reproducing individuals, all models are valid: If an individual has survived so long, there must be something special about it; it is the whole adolescence that is there to filter out the actual mistakes. What is more, one needs to remember that the environment is not a predetermined entity, but it consists of other ever-adapting subsystems, and a stubborn individual can change its environment to make a new personal niche exist.

The role of birth and death are very central in Darwinian evolution theory. Now the system is more important than any individual; life is in the system, and in the population of individuals. As long as the system survives, there is no actual death. Another point is that because the genes only offer the pool of alternatives, the properties of an organism being mainly determined by the environmental conditions, one specific gene combination does not have such a crucial role.

Comparing to the Darwinian theory, again there is the fit criterion that plays a central role. However, now it is not about the search for the absolutely best fit — the population-level system searches for a *set of good fits* to implement a good mixture model, to better capture all aspects of the nonlinear environment. Indeed, the essence of modeling of the environment is not to find the actual winner, but to *find the definition of what fitness is* and map the whole "fitness landscape". And the primary reason for diversity is not to be prepared for the unknown future — the reason is simply to exploit the prevailing environment as efficiently as possible, now and here, with no future prospects. The traditional Darwinian thinking suffers from a intellectual discrepancy: Whereas the evolution mechanisms and fitnesses are defined on the level of individuals, the results are visible and meaningful only on the emergent level of the whole population. Whereas the lower and higher levels are traditionally incompatible, now both

levels are combined in the same model framework, the individuals being submodels that together constitute the systemic model of the species — and the individual species further being submodels that together constitute the systemic model of the ecosystem. Thus, one can proceed from the analysis of individuals to analysis of populations, and from the analysis of species to the analysis of ecosystems; and if one can extend from the analysis of the existing taxonomies to the spectrum of possible ones, from characterizing details to seeing larger patterns, perhaps biology (and ecology) someday become real sciences.

There also exist less concrete populations where the same cybernetic ideas still apply. In a scientific world, for example, being capable of seeing similarities among individual paradigms and combining them in a larger model is similarly a central goal; rather than going deeper into the paradigmatic system, one tries to find more general systems connecting paradigms. In some environments the submodels need not be co-existing and parallel: The "populations" can be, for example, sequential, as it is often the case when speaking of human cultures. However, memetic systems leave signs of themselves, scriptures and artifacts, and as long as these signs can still be deciphered, faiths of various cultures can be reconstructed, and these cultures can be understood as consistent systems. Indeed, being based on such submodels, the highest-level memetic system can become alive — being manifested in a truly cultivated person. The human capacity blooms when one can put things in a perspective, constructing a balanced model of all aspects and dimensions of human culture: The human endeavor is to truly know what it is to be a human, and to understand how the human is connected to the world around him/her. It is not about memorizing details; it is about having a compact model where the individual facts have been combined into more general dependency structures.

6.4 More about sparse coding

For a moment, return to the linear case — it turns out that closer analysis gives insight to understand the general case, too, and the linear submodels efficiently support the emergence of the localization in the nonlinear global model.

It has been observed before that a cybernetic system implements principal component analysis, the submodels representing the (local) observations in terms of (global) variation structures. This is a simple result, as PCA is a mathematically rather trivial operation. Is there nothing else to be said about cybernetic data processing? Indeed, the PCA view is not the whole truth, it only determines the framework for data compression. Within the compressed data space, it is the selection of the latent basis that plays a major role when interpreting the results.

6.4.1 "Black noise"

In chapter 3, connections among $\bar{x}$ and $\bar{u}$, and among $\bar{x}$ and Δu were studied. When studying the theoretical mapping between $\bar{x}$ and the original undisturbed input u, it turns out that the eigenvalues of $\mathrm{E}\{\bar{x}\bar{x}^T\}$ can be expressed in terms of the n most significant eigenvalues λ_j of the original data covariance matrix

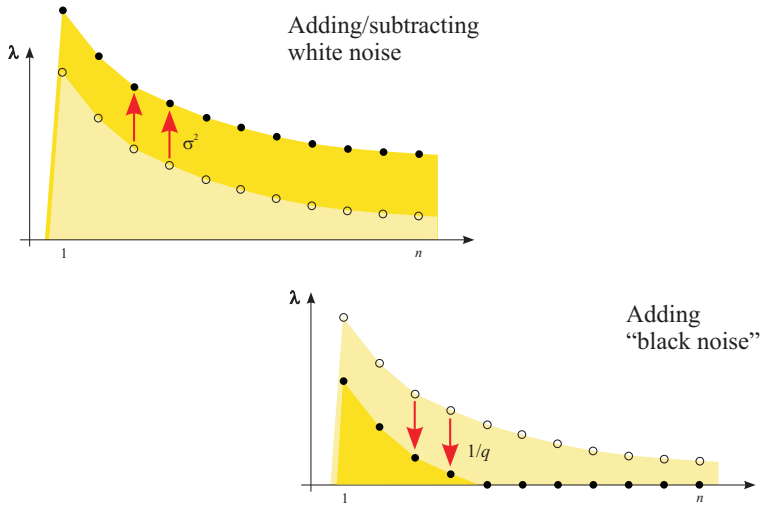

Figure 6.7: Consequences of adding "black noise" are opposite to white noise: The variation decreases in all directions — if possible

$\mathrm{E}\{uu^T\}$, as observed in chapter 3. Specially, if the coupling coefficients q_i and b_i are different for different neurons, the i'th eigenvalue (or latent variable variance) becomes

$$\frac{\sqrt{q_i \lambda_j} - 1}{b_i}, \tag{6.14}$$

indices i and j being ordered randomly. This reveals that there must hold $q_i \lambda_j > 1$ for that input variance direction to remain manifested in the system activity — if this does not hold, variable $\bar{x}_i$ fades away. On the other hand, for the modes fulfilling the constraint, interesting modification of the variance structure takes place; this can best be explained by studying a special case. Assume that one has selected $q_i = \lambda_j$ and $b_i = 1$ for all pairs of i and j. Then the corresponding variances become

$$\lambda_j - 1. \tag{6.15}$$

In each direction in the data space, the effect of the system is to bring the variance level down by a constant factor if it is possible (see Fig. 6.7). Analogically, because white noise increases variation equally in all directions, one could in this opposite case speak of "black noise".

What are the effects of this addition of black noise in the signals? First, it is the principal subspace of u that is spanned by the vectors ϕ_i. But assuming that this subspace is n dimensional, there exist many ways how the basis vectors can be selected, and some of the selections can be physically better motivated. For example, in *factor analysis* the PCA basis vectors are *rotated* to make them aligned with the underlying features, and the same idea takes place in *independent component analysis*. In factor analysis, it can be assumed that the underlying features can be characterized in mathematical terms applying the idea of *sparseness:* When a data vector is decomposed, some of the latent

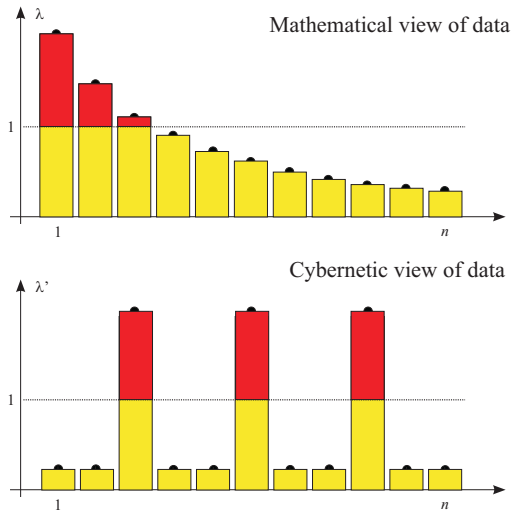

Figure 6.8: How black noise results in sparsity pursuit: Area above the threshold is maximized

variables have high scores while the others have low scores, increasing the differences among latent variable variances. This goal can be iteratively implemented in terms of criteria like *varimax* or *quartimax,* etc. In its extreme form, sparsity means that there are only a few of the candidates employed at a time, and the goal of modeling, rather than being minimization of the number of overall model size, it is the minimization of *simultaneously active constructs.* This means that the total dimension of the latent basis n can even become higher than the dimension m of the input data, the basis being *overcomplete.*

As shown in Figure 6.8, the *Hebbian feedback learning offers an efficient approach to achieving sparsity-oriented basis representation of the PCA subspace.* Whereas the overall captured variation (shown both in yellow and red color in the figure) is not changed by orthogonal rotations, the variation over the bias level (shown in red) *can* be changed. As the nominal PCA approach typically distributes variation more or less evenly along each latent variable, it is most of the variation that remains below the threshold level; now, as it is the area above the threshold level that is maximized, non-trivial basis representations are reached. When doing sparse coding, one can have $n > m$.

There are no closed-form expressions for implementing sparse coding for given data — there are only iterative algorithms. It seems that the algorithm proposed by the Hebbian feedback learning offers a compact and efficient alternative (see Fig. 6.9; compare to the algorithm in 6.2).

In the algorithm, the fixed states are first solved; because of the assumed linearity, infinite iteration changes into a matrix inverse. Actually, the linearity assumption here does not exactly hold: To make the sparse components differentiate, the cut nonlinearity is applied for $\bar{x}$, and, in principle, the matrix inversion does not give the fixed point (however, the system tends towards linearity; see below). The determination of `Xbar` is an extension of that in Algorithm 1, making the matrix inverse better invertible:

$$\bar{x} = \mathrm{E}\{I + \bar{x}\bar{x}^T\}^{-1} Q \mathrm{E}\{\bar{x}u^T\}\, u. \tag{6.16}$$

```
while ITERATE

  % Balance of latent variables
  Xbar = U * (inv(eye(n)+Exx)*Q*Exu)';

  % Enhance model convergence by nonlinearity
  Xbar = Xbar.*(Xbar>0);

  % Balance of the environmental signals
  Ubar = U - Xbar*Exu;

  % Model adaptation
  Exu = lambda*Exu + (1-lambda)*Xbar'*Ubar/k;
  Exx = lambda*Exx + (1-lambda)*Xbar'*Xbar/k;

  % Maintaining system activity
  Q = Q * diag(exp(P*(Vref-diag(Exx))));

end
```

Figure 6.9: **Algorithm 2**: Feedback Hebbian SCA by "selfish agents"

This is solved observing the loop structure, and exploiting (3.36). One can add the triangularization of the covariance matrix `Exx` here, too, to separate the components. The matrix Q is diagonal; "proportional control" with `P` as the control parameter is applied for (logarithms of) variable variations to keep the variation level of the variables in reference (`Vref` is the vector of reference values). Because the elements at the diagonal of Q are distinct, the components become distinguished, as discussed in chapter 3, and rather than implementing sparse subspace analysis, the algorithm implements sparse component analysis. Finally, after convergence the mapping of the model can be expressed as $\phi^T = Q\mathrm{E}\{\bar{x}\bar{u}^T\}$.

The neocybernetic algorithms can also be characterized in terms of mathematically compact formulas and theoretically powerful concepts. The sparse components represent (linear) submodels that together characterize a complex domain, perfectly matching the nonlinear case in 6.3.2. Summarizing, one can say conclude:

> It is the "clever agents" applying Hebbian/anti-Hebbian learning that implement theoretically correct principal component analysis that can be explicitly employed for theoretically optimal least-squares regression; the "selfish agents" applying feedback Hebbian learning implement sparse component analysis and simultaneously implicitly carry out robust regularized least-squares regression to control the environment.

This far all has been linear, the sparsity pursuit being implemented only through basis rotations. When the cut nonlinearity is included in the algorithm, cutting the minor (negative) variations explicitly to zero, only then the algorithm becomes strictly nonlinear. It turns out that the convergence properties of the algorithm can be enhanced considerably then. Because of the optimized rotations, one already has minimized the cross-cluster effects, and for "typical" data located in such clusters, there probably are no crossing-overs between linear submodels. Structure changes are located in deserted regions in space, and rather than being piecewise linear, the model is "practically linear". In the converged system, the role of nonlinearity is rather transparent. But there is more.

The nonlinearity that is introduced in the structure does *not* make the system essentially more complicated. When studying closer the data processing (again see Fig. 3.3), it is interesting to note that the nonlinearity that is now applied is *outside* the inner loop, just filtering the incoming information. The basic functionality of the system is still determined by the closed loop as shown in the figure, converging so that the best possible linear matching between the realized $\bar{x}$ and Δu is implemented, however these signals are externally deformed. This means that despite the nonlinearity, the model tends back towards linearity and statistical optimality.

6.4.2 Towards cognitive functionalities

Modeling of the environment is common to all cybernetic systems. The properties of the environment — like nonlinearities — are best quantifiable when the system resides in infosphere, the signals being better commeasurable, and the existing data structures are intuitively comprehensible.

Example: Modeling of biped walking

When studying the geometric structure of limbs, it is evident that the dynamic model for them is highly nonlinear. Still, to keep a two-legged body stable, very precise control is needed. Whether such control structures can be based on linear submodels that are tuned applying measurement data, was studied in [34]. The available data consisted of state vectors characterizing the orientation and velocities of a simplified two-legged structure and its relation to the surrounding world. The nonlinearities in the adopted model structure were distributed in substructures; it was assumed that the nonlinearities are smooth, and "nearby" data samples share the same locally linear model — that is, the observation data was first clustered, and data within each cluster was used to construct a local linear model. Because of the high assumedly redundant dimensionality of the data, the linear models were based on PCA compression of the observation data, and the motion controls to achieve the walking gait were thereafter reconstructed applying principal component regression based on that model.

It turned out that the clustered model could reproduce the motion controls in a satisfactory manner, and the simulated motion remained in control. However, the model was not quite satisfactory: From the cognitive point of view, the model structure was not very plausible. There was the predetermined structure with separate levels of inter-cluster and intra-cluster operation — coarse

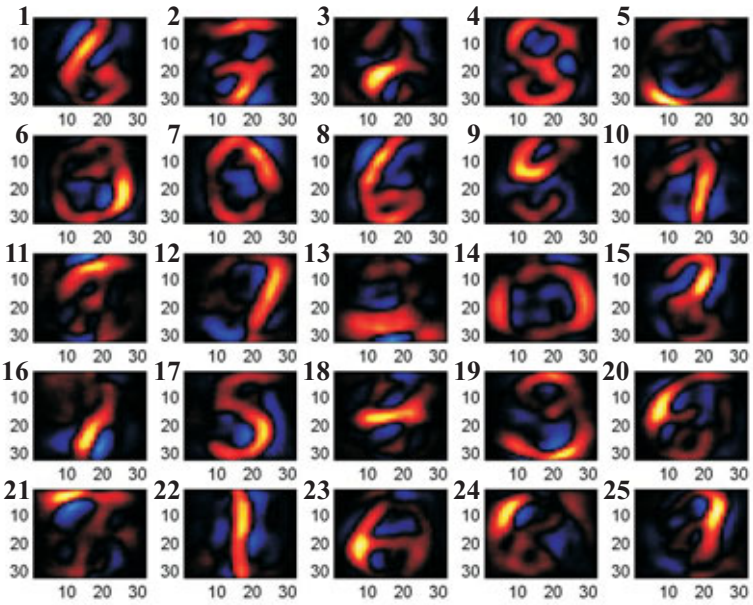

Figure 6.10: The 25 sparse components extracted from the handwritten digits (random ordering). It seems that different kinds of "strokes" become manifested (see below)

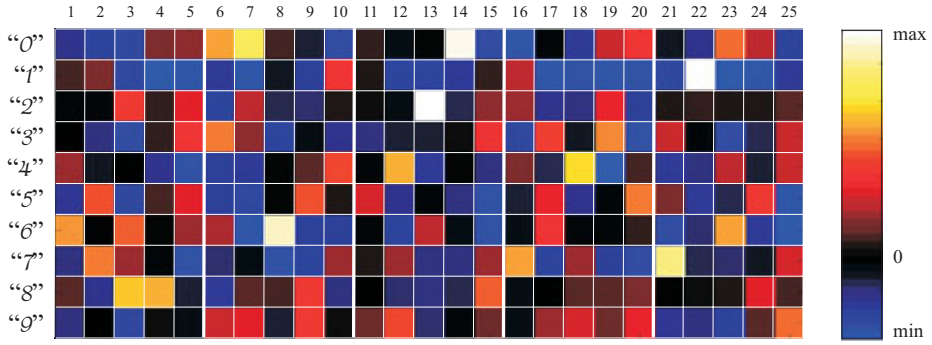

Figure 6.11: How the different digits (left) are represented by the 25 sparse-coded features (above). For example, it seems that feature #14 is only active when the input pattern is "0", feature #22 when it is "1", etc. Feature #13 correlates strongly with "2", as the other patterns seldom occupy the bottom rows. Some patterns have various alternative forms (like "3" is represented either by the feature #6 or feature #19). For most of the input patterns there are no unique matches — they must be composed of parts (for example, "4" seems to be a sum of features #12 and #18). Whether or not the features are disjunctive or conjunctive is determined by the optimization machinery as the data is processed

matching against the clusters, and after that the fine tuning against the cluster-specific submodels. It seems that some higher-level control is necessary here during the model usage as well as during model adaptation. However, it turns out that this is not the case.

As discussed in chapter 7, a grid of linear Hebbian neurons implements the neocybernetic model, modeling the nonlinear environment. Hebbian feedback learning implements the PCA compression of the data, constructing a sparsity-oriented model. Sparse coding results in differentiation of the substructures, or emergence of localized "clusters". Within this framework explicit control of clustering or selection among submodels can be avoided because of the competition among substructures, the best matching submodel automatically receiving most of the activation. There is contribution also by the lesser submodels — this means that there is smooth transfer between submodels in the data space.

What comes to the cluster-based representation of nonlinearities, there is also no need for additional functionalities in the neocybernetic framework. Still, there are challenges: How to implement the input–output structure so that the regression onto the control signals can be implemented in a plausible way? And how to implement optimization towards smoother and faster movements beyond the available prior behaviors?

Structures in infosphere

To illustrate the structure based on sparse codes in more abstract terms, again study the case of hand-written digits (see Sec. 6.2.2). Each of the latent variables $\bar{x}_i$ was kept active by appropriately controlling the coupling factors q_i. Figure 6.10 shows the results when applying the presented algorithm (see also discussion in Fig. 6.3), and Fig. 6.11 presents how the converged features were oriented towards the input patterns. Note that the goal of this coding is not to distinguish but to find similarities — that is why the received feature model is probably not good for classification tasks.

The behaviors in this experiment differed very much from those when applying principal component coding: During the convergence process, in the beginning, something like clustering emerged, each data region being represented by a separate variable; as adaptation proceeded, the features started becoming more orthogonal, and patterns were decomposed further. What is interesting is that this kind of "stroke coding" has been observed also in the visual V1 cortex region in the primate brain (see [29] and [59]): It seems that the visual view is decomposed into simpler, statistically relevant substructures.

What if more complex data is modeled applying the same kind of sparse coding schemes? This was studied using *textual documents*. There were some hundred short descriptions of scientific reports on different aspects of *data mining*. Very simple representation of the texts was selected: It was assumed that the documents can be characterized by the set of words that is found in their descriptions. Data dimension was huge as there was one entry for each of the words in the data vectors. The document were represented by their "fingerprints", or data vectors containing their word counts. After some data preprocessing (see [92]), sparse coding was applied, and the resulting sparse structures representing the correlation structures among the words are shown in Fig. 6.12. It seems that

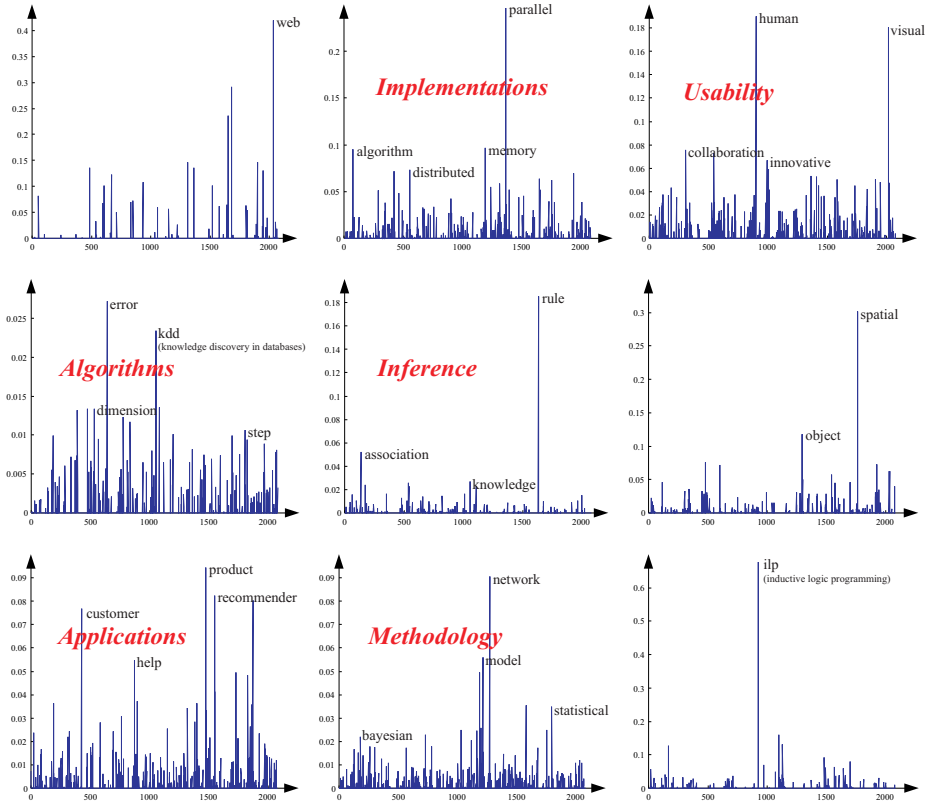

Figure 6.12: Results when textual material (documents on "data mining") are modeled applying sparse coding techniques: It seems that the emerging data structures capturing the correlation structures among the words are *generalized keywords* characterizing the different dimensions in the documents. The nine keywords are projected against the original words that are listed on the horizontal axis in alphabetical order; long bars denote high relevance. The keywords are named afterwards after studying the semantics of the words characterizing them.

the extracted data structures can be used to bring structure even to this semantically complex domain: Different documents can be represented as weighted combinations of the contextual "strokes".

Even though one should be careful about too strong conclusions, these experiments still motivate excursions to truly challenging domains of cybernetics, namely, to the world of cognitive systems — this is done in the next chapter.