Chapter 7
Translational Science under Uncertainty

7.1 Translational Science

Modern engineering begins with a scientific model but in addition to the model there is an objective, such as making a decision based on observations, filtering a signal to reduce noise or accentuate particular frequencies, or intervening in a natural system to force its behavior in a more beneficial direction. The situation changes from modeling behavior to affecting behavior. In medicine, engineering is popularly called *translational science*, which accurately describes modern engineering. A scientific model, whose purpose is to provide a conceptualization of some portion of the physical world, is transformed into a model characterizing human action in the physical world. Scientific knowledge is translated into practical knowledge by expanding a scientific system to include inputs that can be adjusted to affect the behavior of the system and outputs that monitor the effect of the external inputs and feed back information on how to adjust the inputs [Dougherty, 2009a]. For example, in biomedical science models are created with the intention of using them for diagnosis, prognosis, and therapy.

If one is going to transform a physical process, then the conceptualization of that physical transformation takes the form of a mathematical operator on some mathematical system, which itself is a scientific model for the state of Nature absent the transformation. It may be that one cannot obtain a model that can be validated via prediction—that is, a model that has scientific validity—but one may nevertheless find a model that can be used to determine a beneficial operator. The product of pure science is a validated model, whereas the product of translational science is an operator that transforms some aspect of Nature in a quantifiably useful manner. When modeling a cell, the endpoint for pure science is a representation of the dynamical interaction between its macromolecules; for translational science the endpoint might be determination of a drug that will block a signal activating unwanted cellular proliferation. For translation, the scientific model is an intermediate construct used to facilitate control of Nature; its descriptive power is of concern only to the degree that it affects the operator designed from it. For translational science, the epistemological requirements for accepting the model as scientifically valid are replaced by requirements regarding the performance of the operator derived from it. The epistemology of pure science is replaced by the epistemology of practical science [Dougherty, 2016].

The aim of the present chapter is to discuss the basic aspects of translational science in the classical framework of a fully known model and then to examine the situation where the model is uncertain. It is in the presence of uncertainty that the epistemology of translational science confronts operator design in the context of Twenty-first Century complexity. Optimal operator design under uncertainty will be considered in three settings: therapeutic intervention in gene regulatory networks, pattern classification, and signal filtering. Each of these requires some mathematics, but only in the case of signal filtering is some special knowledge required, and we have tried to keep that to a minimum so that the basic ideas are accessible to most readers.

7.2 Anatomy of Translational Science

There are two basic operator problems concerning systems. One is *analysis*: given a system, characterize the properties of the transformed system resulting from the operator in terms of the properties of the original system. Often it is not mathematically feasible to characterize completely the transformed system, or only certain properties of the original system may be known, so that the best one can do is to characterize related properties of the transformed system. This is fine so long as one can characterize those properties of interest to the application. As an example, for a linear operator on a stochastic process, it is usually sufficient to characterize the output covariance function in terms of the input covariance function.

The second basic operator problem is *synthesis*: given a system, design an operator to transform the system in some desirable manner. Synthesis represents the critical act for intervention and forms the basis of modern engineering (translational science). One could grope in the dark, trying one operation after another and observing the result; however, since groping is not grounded in scientific knowledge, we do not consider it to be translational science. In the context of translational science, synthesis begins with the relevant scientific knowledge constituted in a mathematical theory that is used to arrive at an optimal (close to optimal) operator for accomplishing a desired transformation under the constraints imposed by the circumstances. A criterion, called a *cost function* (*objective function*) is defined to judge the goodness of the response—the lower the cost, the better the operator. The objective is to find an optimal way of manipulating the system, which means minimizing the cost function.

Translational-scientific synthesis originated with optimal time series filtering in the classic work of Andrey Kolmogorov [Kolmogorov, 1941] and Norbert Wiener [Wiener, 1949]—although published in 1949, an unpublished version of Wiener's work appeared in 1942. In the Wiener–Kolmogorov theory, the scientific model consists of two random signals, one being the true signal and the other being an observed "noisy" variant of the true signal. The translational aim is to linearly operate on the observed signal so as to transform it to be more like the true signal. Being that a linear operator is formed by a weighted average, the synthesis problem is to find an optimal weighting function for the linear operator and the goodness criterion is the mean-square difference between the true and filtered signals (for a detailed account of the translational nature of the Wiener–Kolmogorov theory, see [Dougherty, 2009b]).

For translational science, synthesis generally involves four steps:

1. Construct the mathematical model.
2. Define a class of operators.
3. Define the optimization problem via a cost function.
4. Solve the optimization problem.

One might prefer a valid scientific model when synthesizing an operator because design would then be based on a system that accurately reflects Nature and thus would portend a better performing operator; however, there is no requirement that the model provides a predictive representation of Nature when application is the goal. With translation, one approaches Nature with the aim of achieving a practical benefit, which is contextual, relative to the cost function and the conditions of application. A translational perspective may be the only viable option when only a targeted objective can reduce the scale of the problem to one that is experimentally, mathematically, and computationally tractable. The predictive capacity of the scientific model is not primary because it is merely a tool and the relevant knowledge applies to the objective, not to the tool. The objective is an optimally performing operator, where performance is measured by the cost function.

In practice, optimality will not be achieved because a physical realization of the mathematical operator must be constructed. Moreover, since there is no assumption of validity regarding the scientific model, one cannot expect that a translationally optimal operator will perform optimally relative to a validated model, although it might. Thus, while the theoretical objective is an optimal mathematical operator, the practical objective is a close-to-optimal physical operator. The actual performance can be evaluated by applying the designed physical operator and estimating the cost function from the data. This is often less burdensome than model validation; nevertheless, there may still be insufficient data for obtaining a good estimate, depending on the complexity of the cost function and the difficulty of testing.

7.2.1 Structural intervention in gene regulatory networks

When every gene in a Boolean network (or PBN) has a positive perturbation probability, then for any state $\mathbf{x}$ the probability that the network is in state $\mathbf{x}$ in the long run (in the limit) is independent of the initial state. This limiting probability is called a *steady-state probability* and the collection of all such probabilities is called the *steady-state distribution*. Not every network possesses a steady-state distribution. For instance, consider a 3-gene deterministic Boolean network with two basins: $100 \rightarrow 010 \rightarrow 001 \rightarrow 000$ and $110 \rightarrow 011 \rightarrow 101 \rightarrow 111$. Then the long-run probability of 000 is 1 if the network is initialized at 100 and is 0 if it is initialized at 110. There is no steady-state distribution.

Assuming the existence of a steady-state distribution, structural intervention in a gene regulatory network involves a one-time change of the regulatory structure to reduce the steady-state probabilities of undesirable (pathological) states [Qian and Dougherty, 2008]. This means minimizing the sum of the steady-state probabilities corresponding to the undesirable states. Following [Yoon et al., 2013], to illustrate structural intervention we consider a mammalian cell cycle Boolean network with perturbation ($p = 0.01$) based on a regulatory model proposed by [Faure et al., 2006]. Intervention is based on the fact that in molecular biology there are techniques for "pathway blockage." We employ a structural intervention that models small interfering RNA (siRNA) interference in regulatory relationships: an intervention blocks the regulation between two genes in the network.

The cell cycle involves a sequence of events resulting in the duplication and division of the cell. It occurs in response to growth factors and under normal conditions it is a tightly controlled process. The model contains 10 genes: CycD, Rb, p27, E2F, CycE, CycA, Cdc20, Cdh1, UbcH10, and CycB, with genes numbered in this order. The cell cycle in mammals is controlled via extra-cellular stimuli. Positive stimuli activate Cyclin D (CycD) in the cell, thereby leading to cell division. CycD inactivates the Rb protein, which is a tumor suppressor. When gene p27 and either CycE or CycA are active, the cell cycle stops, because Rb can be expressed even in the presence of cyclins. States in which the cell cycle continues even in the absence of stimuli are associated with cancerous phenotypes. For this reason, states with down-regulated CycD, Rb, and p27 ($x_1 = x_2 = x_3 = 0$) are undesirable.

The regulatory model, shown in Fig. 7.1, has blunt arrows representing suppressive regulations and normal arrows representing activating regulations. Genes are assumed to be regulated according to the majority vote rule. At each time point, a gene takes the value 1 if the majority of its regulator genes are activating and the value 0 if the majority of the regulator genes are suppressive; otherwise, it remains unchanged. A structural intervention removes an arrow from the regulatory graph because it blocks a regulation between two genes. By the optimization methods of [Qian and Dougherty, 2008] it is determined that the structural intervention that maximally lowers undesirable steady-state probability blocks the regulatory action from gene CycE to p27 and reduces total undesirable steady-state probability from 0.3405 to 0.2670. The steady-state distributions for the original network and the treated network are shown in Fig. 7.2.

The translational character of structural intervention is reflected in how the four aspects of synthesis are manifested:

1. Model the cell cycle by a Boolean network with perturbation.
2. An intervention operator blocks a single regulation between two genes.
3. The cost is the total steady-state probability of the undesirable states.
4. An optimal action is found via the method of [Qian and Dougherty, 2008].

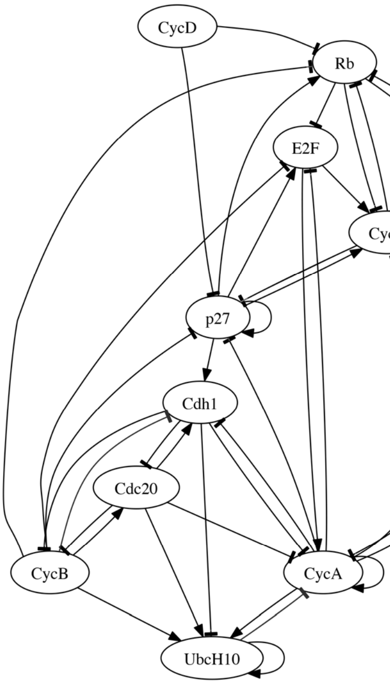
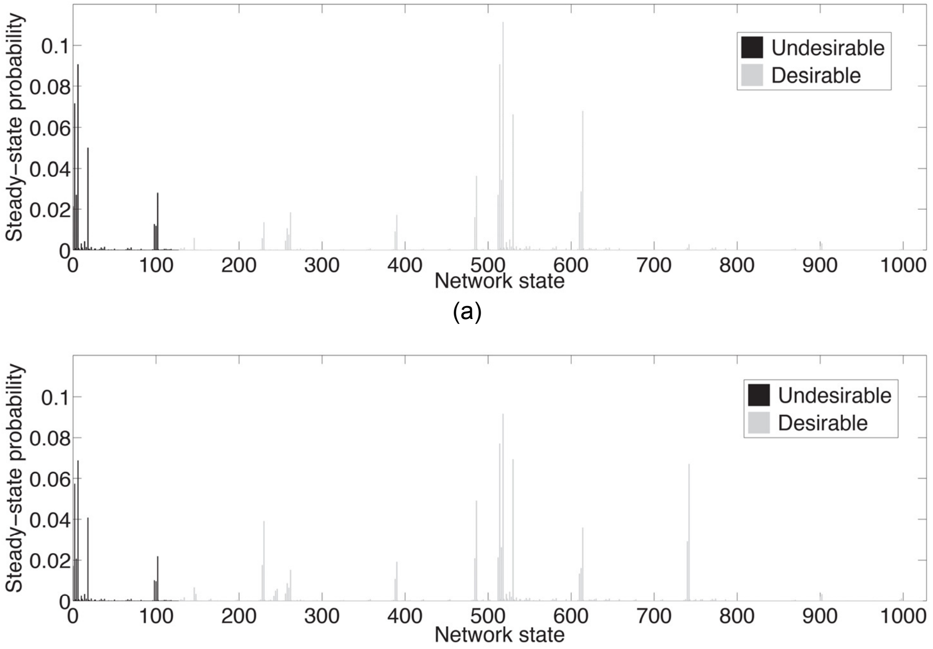


Figure 7.1 Mammalian cell cycle network (adapted from [Yoon et al., 2013]).



(a)

(b)

Figure 7.2 Steady-state distribution for mammalian cell cycle network (states listed numerically): (a) original and (b) after optimal structural intervention. (Part (a) adapted from [Yoon et al., 2013]).

In practice, basing the optimization on a cost function alone may not be satisfactory and constraints on the optimization may need to be imposed. In the case of gene regulation optimization can be phenotypically constrained, meaning that when altering steady-state probabilities one may wish to constrain where the probability is moved [Qian and Dougherty, 2012]. For instance, while lowering steady-state probability for undesirable states, one may wish to keep it from being moved to states known to be associated with carcinogenesis or to states that do not typically occur in healthy cells. In general, the optimization problem should be set up with input from cancer biologists.

7.3 Operator Design in the Presence of Model Uncertainty

To formulate optimization when there is model uncertainty, consider a stochastic model $\mathcal{M}$ with uncertainty class Θ. For example, $\mathcal{M}$ might be a gene regulatory network with some unknown regulations, so that Θ consists of all possible parameter vectors corresponding to the unknown regulations. Let C be a cost function and Ψ be a class of operators on the model whose performances are measured by the cost function. This means that for each operator $\psi \in \Psi$ there is a cost $C_\theta(\psi)$ of applying ψ on model $\theta \in \Theta$. For example, suppose Ψ consists of 5 drugs, meaning that each operator acts by applying a drug. Suppose the goal of the drug treatment is to reduce the expression of a particular gene g associated with metastasis in breast cancer and that the gene regulatory network being used is uncertain, so that there is an uncertainty class Θ of models. The cost function might be the average gene expression for g over some time interval after the drug has had time to take effect. Then $C_\theta(\psi)$ is the average gene expression over the time interval when drug ψ is applied to model θ. Since the full network model is unknown, there being uncertain parameters, one would like to choose a drug whose performance works well over the uncertainty class.

An *intrinsically Bayesian robust* (IBR) operator on $\mathcal{M}$ is an operator $\psi_{\mathrm{IBR}} \in \Psi$ such that the expected (average) value over Θ of the cost $C_\theta(\psi)$ is minimized by ψ_{IBR}, the expected value being with respect to a *prior probability distribution* $\pi(\theta)$ over Θ [Dalton and Dougherty, 2014]. An IBR operator is robust in the sense that on average it performs well over the whole uncertainty class. Since each parameter vector $\theta \in \Theta$ corresponds to a model, a probability distribution on the space of possible models quantifies our belief that some models are more likely to be the actual full model than are others. Such a distribution reflects prior knowledge. If there is no prior knowledge beyond the uncertainty class itself, then the prior distribution is taken to be *uniform*, meaning that all models are assumed to be equally likely.

Denoting the expected value over Θ by E_Θ, an IBR operator minimizes the expected value of the cost:

$$E_\Theta[C_\theta(\psi_{\mathrm{IBR}})] = \min\{E_\Theta[C_\theta(\psi)], \psi \in \Psi\}. \tag{7.1}$$

If the uncertainty class is finite, say, $\Theta = \{\theta_1, \theta_2,\ldots, \theta_m\}$, then the expected cost over the uncertainty class is a weighted average, the costs being weighted by the prior distribution:

$$E_\Theta[C_\theta(\psi)] = C_{\theta_1}(\psi)\pi(\theta_1) + C_{\theta_2}(\psi)\pi(\theta_2) + \ldots + C_{\theta_m}(\psi)\pi(\theta_m). \tag{7.2}$$

If Θ is infinite, then the expected value over Θ is given by the integral of the cost over Θ with respect to the prior distribution:

$$E_\Theta[C_\theta(\psi)] = \int_\Theta C_\theta(\psi)\pi(\theta)d\theta. \tag{7.3}$$

The basic idea is straightforward: find an operator that minimizes the average cost when applied to all models in the uncertainty class. Based on existing knowledge, which is captured in the known parameters and the prior probability distribution over the uncertainty class, an IBR operator provides the best robust performance across the uncertainty class. When one possesses no knowledge concerning the likelihoods of the models in the uncertainty class and the prior distribution is *uniform* over Θ, then $\pi(\theta_1) = \pi(\theta_2) = \ldots = \pi(\theta_m) = 1/m$ in Eq. (7.2).

7.3.1 IBR structural intervention in gene regulatory networks

We return to the mammalian cell cycle network but now consider intrinsically Bayesian robust structural intervention. Uncertainty occurs because there are D pairs of genes for which the existence of a regulatory relationship is known but the type of relationship, activating or suppressing, is unknown. Consequently, the network uncertainty class Θ consists of 2^D possible networks, where each $\theta \in \Theta$ corresponds to a specific assignment of regulation types to the D uncertain edges. The uncertainty class is governed by a uniform prior distribution, meaning that we have no knowledge concerning model likelihood and all uncertain parameters have prior probability $1/2^D$. As previously assumed, a structural intervention blocks the regulatory action between a pair of genes in the network. Once gain, the cost function is the total undesirable steady-state probability. Based on the given mammalian cell cycle network, simulations have been run in [Yoon et al., 2013] that incrementally increase the number of edges with unknown regulation from $D = 1$ to $D = 10$. In each case, 50 uncertain networks are created by randomly selecting uncertain edges while keeping the regulatory information for the remaining edges.

Grouping the models with 1 to 5 uncertain edges, 54.0% of the time the IBR structural intervention is the actual optimal intervention, which blocks the regulation from CycE to p27. As seen in Section 7.2.1, when applied to the full model, this reduces total undesirable steady-state probability to 0.2639. The second most selected IBR intervention blocks the regulation from CycE to Rb. It is chosen 41.6% of the time and reduces total undesirable steady-state probability to 0.2643. Four other interventions are chosen a total of 4.4% of the time.

Since the optimization provides the intervention that works best on average over the uncertainty class, it may choose an intervention that performs poorly on the full network. In this simulation, blocking regulation between CycB and p27 is selected 2.0% of the time and only reduces undesirable steady-state probability to 0.3244. When the simulation is run with 6 to 10 uncertain edges, blocking CycE to p27 or blocking CycE to Rb accounts for 88.8% of the IBR interventions, as opposed to 95.6% of the IBR interventions for 1 to 5 uncertain edges. This change reflects the greater uncertainty.

7.4 Pattern Classification

Pattern classification is used in every applied discipline because it is the mathematical formulation of decision making and every discipline requires decisions. In cancer medicine, classification can be between different kinds of cancer, stages of tumor development, or prognoses. This section considers optimal binary classification when the model is known and when it is uncertain.

7.4.1 Optimal classification for a known feature-label distribution

The basic idea for classification is that *features* are calculated on objects from two different populations, and based on a vector of features a classifier decides which population an object belongs to. For instance, gene expressions are measured for k genes, and based on the measurements it is decided which drug should be administered. A feature vector belongs to one of two classes, labeled 0 and 1. The model is stochastic and consists of feature-label pairs $(\mathbf{X}, Y)$, where $\mathbf{X} = (X_1, X_2, \ldots, X_k)$ and $Y = 0$ or $Y = 1$. A *classifier* ψ is a decision function on the set of feature vectors: $\psi(\mathbf{X}) = 0$ or $\psi(\mathbf{X}) = 1$. It partitions the feature space into two regions, R_0 and R_1.

For classification, the scientific model consists of two distributions, called *class-conditional distributions*: $f(\mathbf{x}|0)$ and $f(\mathbf{x}|1)$ are the probability distributions governing the behavior of feature vectors in class 0 and class 1, respectively. The model also requires the probability c_0 that a randomly selected object comes from class 0, which automatically gives the probability c_1 that it comes from class 1 since $c_1 + c_0 = 1$. Taken together, $f(\mathbf{x}|0)$, $f(\mathbf{x}|1)$, and c_0 provide the *feature-label distribution* $f(\mathbf{x}, y)$ governing the feature-label vectors. For simplicity, we assume that $c_0 = c_1 = ½$, so that the classes are equally likely.

The *error* of any classifier ψ is the probability of erroneous classification, $\varepsilon[\psi] = P(\psi(\mathbf{X}) \neq Y)$, which can be found from the feature-label distribution. Letting Ψ denote the set of all classifiers on the model, an optimal classifier is called a *Bayes classifier* and is denoted by ψ_{Bay}. It has minimum error among all classifiers in Ψ and need not be unique. Given $c_0 = c_1 = ½$, a Bayes classifier is defined by a simple rule: for a given feature vector $\mathbf{x}$,

$$\psi_{\text{Bay}}(\mathbf{x}) = \begin{cases} 1, \text{if } f(\mathbf{x}\,|\,1) \geq f(\mathbf{x}\,|\,0) \\ 0, \text{if } f(\mathbf{x}\,|\,1) < f(\mathbf{x}\,|\,0) \end{cases}. \tag{7.4}$$

This is equivalent to $\psi_{\text{Bay}}(\mathbf{x}) = 1$ if and only if $f(\mathbf{x}, 1) \geq f(\mathbf{x}, 0)$, which intuitively means that ($\mathbf{x}$, 1) is more likely than ($\mathbf{x}$, 0). The error of a Bayes classifier is known as the *Bayes error*. It is denoted by ε_{Bay} and is the minimum among the errors of all classifiers on the feature-label distribution. While there may be many Bayes classifiers for a feature-label distribution, the Bayes error is unique.

Consider a single measurement X of a system that has a normal distribution with mean 0 and standard deviation σ when the system is in the unperturbed state, but when the system is perturbed in a particular way the normal distribution shifts so that its mean becomes $\theta > 0$, while maintaining the same standard deviation. We desire a classifier to predict the state of the system (unperturbed or perturbed) based on the measurement X. Assuming equal likelihood for the two states, Fig. 7.3 shows that a Bayes classifier is defined by

$$\psi_{\text{Bay}}(x) = \begin{cases} 1, \text{if } x \geq \theta/2 \\ 0, \text{if } x < \theta/2 \end{cases}, \tag{7.5}$$

and the error is the area of the shaded region.

For a more visual example, consider the two normal two-dimensional class-conditional distributions in Fig. 7.4. They have different mean vectors in the plane and have the same covariance matrix (which determines the shape of the surfaces). A Bayes classifier is defined by the straight line that separates the plane into regions R_0 and R_1. If $\mathbf{x} \in R_0$, then $\psi_{\text{Bay}}(\mathbf{x}) = 0$; if $\mathbf{x} \in R_1$, then $\psi_{\text{Bay}}(\mathbf{x}) = 1$. If the covariance matrices were not equal, then the class-conditional distributions would not have the same shape and the decision boundary would be quadratic instead of linear.

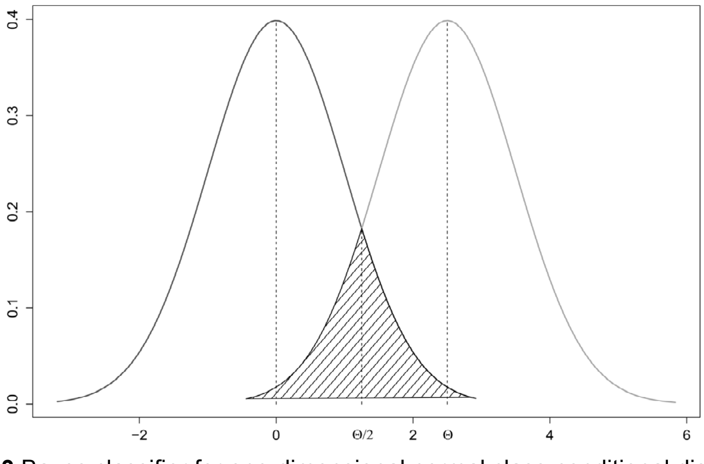


Figure 7.3 Bayes classifier for one-dimensional normal class-conditional distributions.

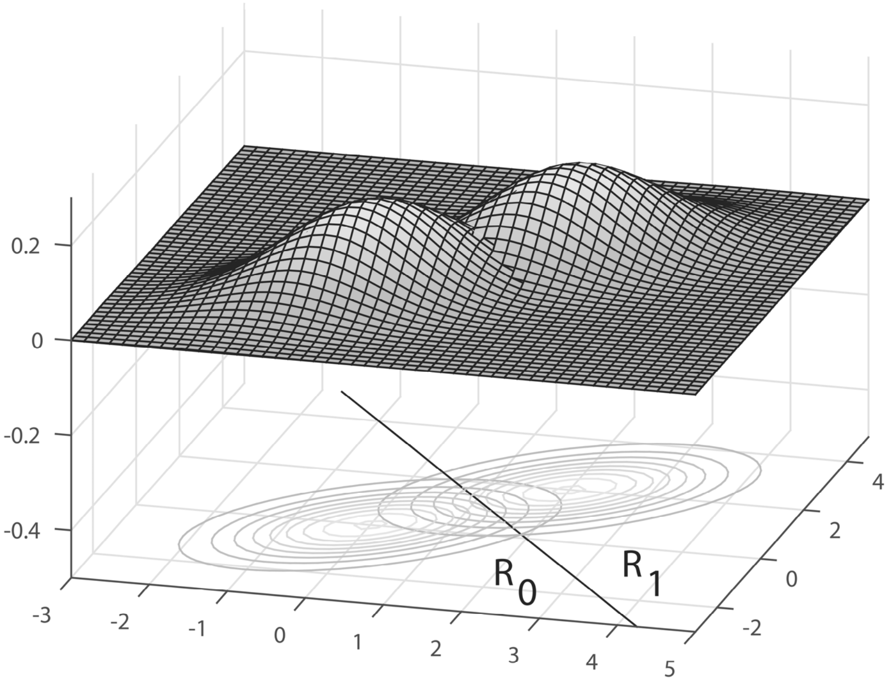


Figure 7.4 Bayes classifier for two-dimensional normal class-conditional distributions.

Considering features and labels as physical measurements, the feature-label distribution represents knowledge of the variables X_1, X_2,…, X_k, Y. The Bayes error is intrinsic to the model and quantifies the separability of the classes relative to the features. We desire features that separate well the class-conditional distributions. Given a feature-label distribution, one can in principle find a Bayes classifier and the Bayes error; however, for important models, only in rare cases have these been analytically derived from the feature-label distribution, but they can be approximated by numerical methods.

Corresponding to the four generic steps for optimal operator synthesis are the following four steps for classification:

1. Construct the feature-label distribution.
2. The operators consist of classifiers on the feature-label distribution.
3. The cost is classifier error.
4. An optimal operator is given by a Bayes classifier.

7.4.2 Intrinsically Bayesian robust classification

Model uncertainty arises when full knowledge of the feature-label distribution is lacking. Knowledge must come from existing scientific knowledge regarding the features and labels or be estimated from data. Since accurate estimation of distributions requires a huge amount of data, the amount increasing rapidly with dimension and distributional complexity, full knowledge of the feature-label distribution is rare. With model uncertainty, there is an uncertainty class Θ of parameter vectors corresponding to feature-label distributions. In this setting, an intrinsically Bayesian robust classifier is defined by minimizing the expected error across the uncertainty class. Letting $\varepsilon_\theta[\psi]$ denote the error of classifier ψ on model θ and recalling Eq. (7.1), an IBR classifier satisfies

$$E_\Theta[\varepsilon_\theta[\psi_{\mathrm{IBR}}]] = \min\{E_\Theta[\varepsilon_\theta[\psi]], \psi \in \Psi\} = \min\left\{\int_\Theta \varepsilon_\theta[\psi]\pi(\theta)d\theta, \psi \in \Psi\right\}, \quad (7.6)$$

where π(θ) is the prior distribution over Θ and where the integral has the same dimensionality as the parameter vectors in Θ.

We return to the classification problem of Fig. 7.3 with the supposition that θ is unknown but is known to lie in the interval [0, *b*]. Then the uncertainty class Θ = [0, *b*] corresponds to an infinite number of feature-label distributions. Absent other knowledge of θ, it is assumed to be uniformly distributed over [0, *b*], meaning that it is described by the prior distribution π(θ) = 1/*b* if θ ∈ [0, *b*] and π(θ) = 0 if θ ∉ [0, *b*]. For each value of θ ∈ [0, *b*], a Bayes classifier is defined by Eq. (7.5) and its error is found as in Fig. 7.3. Then, according to Eq. (7.6), an IBR classifier satisfies

$$E_\Theta[\varepsilon_\theta[\psi_{\mathrm{IBR}}]] = \min\left\{\frac{1}{b}\int_0^b \varepsilon_\theta[\psi]d\theta, \psi \in \Psi\right\}, \quad (7.7)$$

where the integral is one-dimensional.

The minimization of Eq. (7.6) is analogous to the minimization for determining a structural intervention in a gene regulatory network except that, whereas for structural intervention as defined for the mammalian cell cycle network one can compute a finite number of operator costs (undesirable steady-state probabilities) and take the least, for IBR classification there is an infinite number of operators (classifiers) to consider. As expressed in Eq. (7.6), and exemplified in Eq. (7.7), one is left with the problem of finding a minimizing classifier when the collection of classifiers is infinite. A formula is needed that produces an IBR classifier.

This problem is solved in [Dalton and Dougherty, 2013] under very general conditions. The method uses *effective class-conditional distributions* for the uncertainty class. These are defined by the expected values of the individual class-conditional distributions over the uncertainty class. Formally, let $f(\mathbf{x}|0; \theta)$ and $f(\mathbf{x}|1; \theta)$ denote the class-conditional distributions for θ ∈ Θ. Then the effective class-conditional distributions are defined by the expected values (averages) of these over the uncertainty class:

$$f(\mathbf{x}|0;\Theta) = E_\Theta[f(\mathbf{x}|0;\theta)] = \int_\Theta f(\mathbf{x}\,|\,0;\theta)\pi(\theta)\,d\theta\,, \tag{7.8}$$

$$f(\mathbf{x}|1;\Theta) = E_\Theta[f(\mathbf{x}|1;\theta)] = \int_\Theta f(\mathbf{x}\,|\,1;\theta)\pi(\theta)\,d\theta\,. \tag{7.9}$$

Continuing to assume that $c_0 = c_1 = ½$, an IBR classifier is found in exactly the same manner as a Bayes classifier, except that the effective class-conditional distributions are used:

$$\psi_{\mathrm{IBR}}(\mathbf{x}) = \begin{cases} 1, \text{if } f(\mathbf{x}\,|\,1;\Theta) \geq f(\mathbf{x}\,|\,0;\Theta) \\ 0, \text{if } f(\mathbf{x}\,|\,1;\Theta) < f(\mathbf{x}\,|\,0;\Theta) \end{cases}. \tag{7.10}$$

7.5 Posterior Distribution

In addition to a prior distribution coming from existing knowledge, suppose one has a data sample S independently sampled from the full model. Then a *posterior distribution* is defined by $\pi^*(\theta) = \pi(\theta|S)$, which is the prior distribution conditioned on the sample. The posterior distribution is derived using standard statistical techniques, although, depending on the prior distribution, it may not be mathematically feasible to obtain an exact expression for $\pi^*(\theta)$ and numerical methods may be used to approximate it. Once the posterior distribution has been found, the IBR theory can be used with $\pi^*(\theta)$ in place of $\pi(\theta)$, the resulting operator being known as an *optimal Bayesian operator*. As illustrated in Fig. 7.5, under appropriate conditions, as the sample grows, the posterior distribution becomes more tightly centered about the parameter vector for the full model.

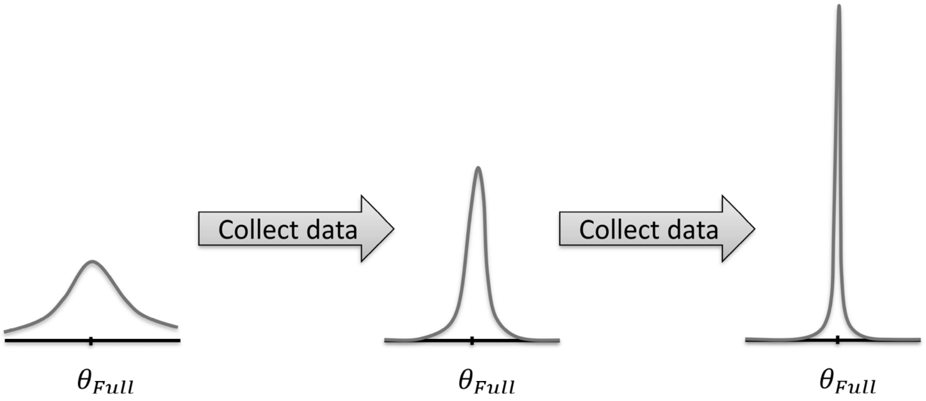


Figure 7.5 Tightening of the posterior distribution with increasing data.

7.5.1 Optimal Bayesian classification

For classification, the sample data consist of feature-label pairs and these are used to find the posterior distribution [Dalton and Dougherty, 2011]. Effective class-conditional distributions are defined by Eqs. (7.8) and (7.9) with $\pi^*(\theta)$ in place of $\pi(\theta)$, and an *optimal Bayesian classifier* (OBC) is defined by Eq. (7.10) [Dalton and Dougherty, 2013]. An OBC has minimum expected error relative to the posterior distribution $\pi^*(\theta)$, which contains all of our knowledge, prior knowledge as interpreted via the prior distribution and experimental data.

Owing to the growing concentration of the posterior distribution around the full model, as illustrated in Fig. 7.5, as the sample size grows ever larger, the OBC typically converges to a Bayes classifier for the full model (Fig. 7.6). While this is an attractive property and is common for optimal Bayesian operators defined via posterior distributions, its practical significance is limited because the basic problem is lack of data.

Figure 7.7 illustrates OBC behavior. There is an uncertainty class of feature-label distributions, each possessing normal class-conditional distributions with equal covariance matrices. The dotted lines are level curves for the normal class-conditional distributions corresponding to the average means and covariance matrices relative to a given posterior distribution. The dashed straight line is the decision boundary for the Bayes classifier corresponding to average mean and covariance parameters. The solid line is the boundary for the OBC. Note that every feature-label distribution in the uncertainty class and the average feature-label distribution have linear (straight line) Bayes classifiers; however, the OBC has a more complex decision boundary. This results from the fact that all class-conditional distributions in the uncertainty class are normal but the effective class-conditional distributions are not normal.

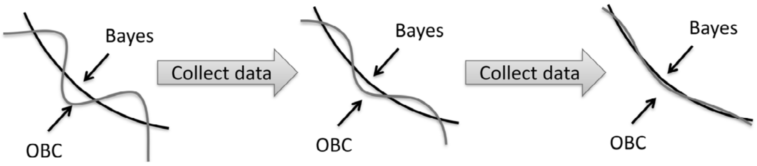


Figure 7.6 Convergence of the OBC to the Bayes classifier.

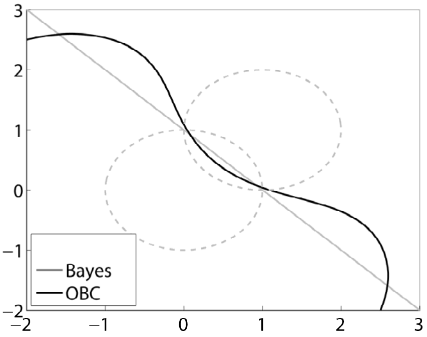


Figure 7.7 Comparison of OBC and Bayes classifier for an average model.

7.5.2 Distribution-free classification

In pattern recognition it is common to assume no prior knowledge concerning the feature-label distribution, so that classifier design is *distribution-free*. Hence, the subject has grown around the notion of a *classification rule*, which is some procedure applied to the sample data to construct a classifier, such as a support vector machine, neural network, or a host of other procedures. The particularities of classification rules are not of interest here. For our purposes, one need only recognize that a classification rule uses sample data to construct a classifier, and individual performance depends on the unknown feature-label distribution and sample size. The rules are heuristic, in the sense that their formulation is based on some guiding principle rather than optimization.

Once a classifier is designed, the salient issue is its error. The problem is that, given a sample, we cannot find the error of the resulting classifier because the feature-label distribution is not known. The problem then is to find an estimate of the error. In general, this can be approached in two ways.

If there is an abundance of sample data, then the data can be split into two disjoint sets: a *training set* used to design the classifier and *test set* used to estimate the error of the classifier. Once the classifier is designed, the test error is the proportion of errors it makes on the test set. This error is called the *hold-out error* owing to the fact that the test set has been held out from the training procedure. How good is the hold-out estimate? The obvious answer would be to quantify goodness by $|\hat{\varepsilon} - \varepsilon|$, where ε and $\hat{\varepsilon}$ are the true and estimated errors, respectively; however, this is impossible because the true error is unknown. Instead, we consider how well the estimation procedure works on average. This performance measure is given by the root-mean-square (RMS) error, which is the square root of the expected value of $|\hat{\varepsilon} - \varepsilon|^2$, namely, RMS = $E[|\hat{\varepsilon} - \varepsilon|^2]^{1/2}$, which is the square root of the mean-square error (MSE) and where the expectation (average) is taken with respect to the sampling procedure.

As it stands, the RMS cannot be found because it requires knowledge of the feature-label distribution. Nevertheless, it is known that, irrespective of the feature-label distribution, RMS $\leq 1/(2m^{1/2})$, where m is the size of the test set [Devroye, et al., 1996]. This is a good result since it is distribution-free and for a test sample of the modest size m = 100, RMS $\leq$ 0.05. It does not depend on dimension (number of features) or classifier complexity. Even though the accuracy of the specific estimate is not known, there is a precise bound on estimation performance. While the RMS bound for classification error estimation is encouraging, one should keep in mind that a classifier is a very simple model, just a binary function.

Because classifier error quantifies the predictive capacity of a classifier, error-estimation accuracy is the salient epistemological issue for classification. Hence, the bound on the hold-out estimate is a fundamental epistemological measure.

Hold-out error estimation requires a sufficiently large sample so that there are enough data to design the classifier (a problem we will not consider) and enough independent data for error estimation. Based on the RMS bound, 100 test points provides reasonably good error estimation. If sample data are limited, say, to a sample size of 100, then hold-out error estimation cannot be employed and the error must be estimated using the training data. Numerous methods have been proposed for training-data-based error estimation, each possessing different properties [Braga-Neto and Dougherty, 2015]. The simplest method is known as *resubstitution*, where the error estimate is the proportion of errors made by the designed classifier on the training data. Resubstitution is usually optimistically biased (often strongly) and therefore rarely used. Error-estimation methods in which the training data are re-sampled for design and testing within the training data include cross-validation and bootstrap. These tend to perform poorly on small samples owing to large variance and lack of regression with the true error. There are very few known distribution-free RMS bounds for training-data error estimation. For the few cases in which distribution-free RMS bounds are known, they are very weak and a large sample is required to obtain an acceptable bound, which renders the bound useless because training-data error estimation methods are being used precisely because the sample is too small to split into training and test data. In sum, the salient epistemological issue for small-sample classification is that quantifiable distribution-free error estimation is virtually impossible.

If one has distributional knowledge in the form of an uncertainty class and a prior distribution, then a posterior distribution can be derived using the sample data, in which case the error of a designed classifier can be estimated as the expected error over the posterior distribution. The resulting estimate is known as the *Bayesian error estimate* (BEE) [Dalton and Dougherty, 2011]. This can be done because the true error of the classifier can be evaluated for each model in the uncertainty class, after which these errors are averaged with respect to the posterior distribution. It can be proven that the resulting error estimate is optimal relative to the expected (average) RMS over the uncertainty class. It may not be best for all models in the uncertainty class, but it is best on average, which means it is best relative to all of our knowledge, prior distribution plus sample data.

In sum, given a prior distribution on the uncertainty class and sample data, the OBC is the optimal classifier and the BEE is the optimal error estimate. Absent prior knowledge, small-sample classification is essentially pointless owing to the impossibility of obtaining an error estimate whose accuracy can be quantified.

7.6 Translational Science under Model Uncertainty

When the uncertainty class is finite, an intrinsically Bayesian robust operator can be found by computing a finite number of costs, as in Eq. (7.2); however, for infinite uncertainty classes, some other approach must be found. In the case of classification, for each model in the uncertainty class the individual class-conditional distributions are considered as characteristics of the full model that define an optimal operator (Bayes classifier) for that model. The methodology of [Dalton and Dougherty, 2013] is to construct *effective characteristics* and then prove that an IBR operator, which in this case is an IBR classifier, can be constructed in the same way as an individual optimal operator (Bayes classifier) by replacing the individual model characteristics with effective characteristics.

Thus, with model uncertainty we have the following IBR synthesis protocol, which will be illustrated in subsequent subsections:

1. Construct the mathematical model.
2. Define a class of operators.
3. Define the basic optimization problem via a cost function.
4. Solve the basic optimization problem via characteristics of the model.
5. Identify the uncertainty class.
6. Construct a prior distribution.
7. State the IBR optimization problem.
8. Construct the appropriate effective characteristics.
9. Prove that the IBR optimization problem is solved by replacing the model characteristics by the effective characteristics.

7.6.1 Wiener filter

Wiener filtering involves two random signal processes, an unobserved true signal and an observed signal, with the aim being to apply a linear filter on the observed signal to estimate the true signal. For details, see Section 4.7.2 in [Dougherty, 1999]. Here, for those with some background in filtering, we provide highlights without supporting theory to illustrate how the various steps for translational synthesis apply. The true signal and observation processes, $Y(t)$ and $X(t)$, respectively, are jointly wide-sense stationary (WSS) and possess zero means. The autocorrelation function for the observation process is denoted by $r_X(\tau)$ and the cross-correlation function between the signal and observation processes is denoted by $r_{YX}(\tau)$.

A linear filter with weighting function $\hat{g}$ takes the form

$$\hat{Y}(s) = \int_T \hat{g}(s-t)X(t)\,dt\,, \tag{7.11}$$

where the integral is over an observation window T. The objective is to obtain an estimate of the true signal that minimizes the mean-square error (MSE) at a given point s, which is defined as

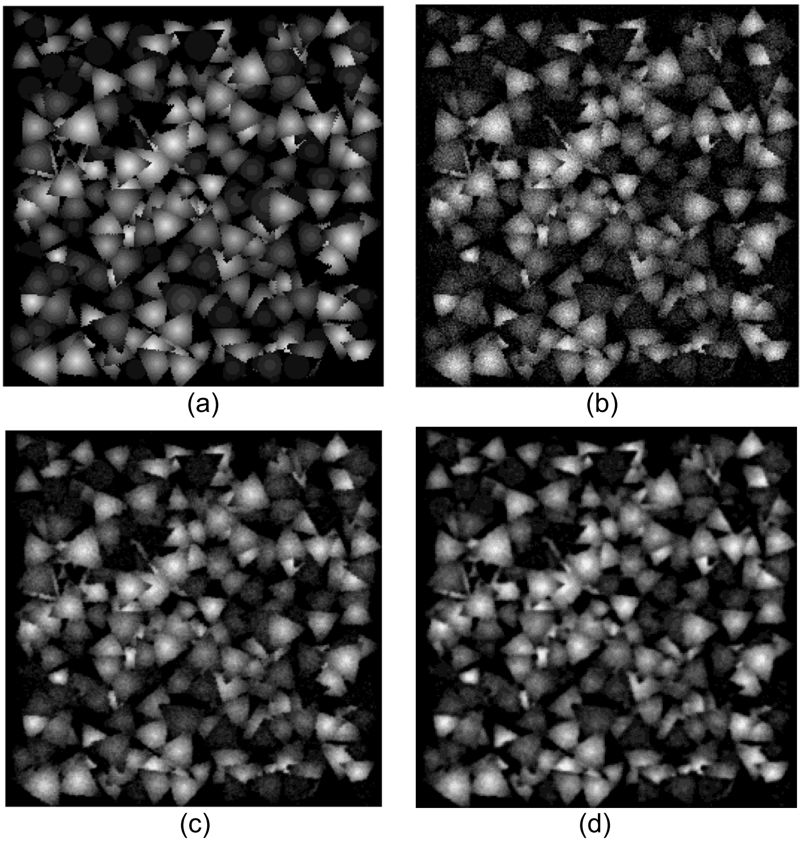
$$\mathrm{MSE}\langle \hat{Y}(s)\rangle = E[|\hat{Y}(s)-Y(s)|^2]\,. \tag{7.12}$$

For any WSS random process, the *power spectral density* of the process is the Fourier transform of the autocorrelation function. For the observation process, it is given by $S_X(\omega) = \mathcal{F}[r_X](\omega)$, where $\mathcal{F}$ denotes the Fourier transform. The *cross power spectral density* is $S_{YX}(\omega) = \mathcal{F}[r_{YX}](\omega)$, the Fourier transform of the cross-correlation function between the signal and observation processes. $S_X(\omega)$ and $S_{YX}(\omega)$ are characteristics of the model, and under rather general conditions it is well-known that the Fourier transform of the optimal weighting function is

$$\hat{G}(\omega) = \frac{S_{YX}(\omega)}{S_X(\omega)}. \tag{7.13}$$

The optimal weighting function, which defines the *Wiener filter*, is obtained by taking the inverse Fourier transform. This is the major classical result of signal filter theory. It applies to images by performing all operations in two dimensions.

Figure 7.8 illustrates Wiener filtering with a digital image process consisting of random grains. Parts (a), (b), and (c) of the figure show an image generated by the process, that image degraded by both blurring and random point noise, and the noisy image filtered by the Wiener filter for the random process, respectively. The filtering problem is made more difficult when there is both blurring and point noise because for blurring alone the image can be "sharpened" and for point noise alone it can be "smoothed." Mixed blurring and point noise is tricky because sharpening makes point noise worse and smoothing makes blurring worse. Without a mathematical approach to the problem it would be virtually impossible to find a close-to-optimal weighting function.



(a) (b)

(c) (d)

Figure 7.8 Wiener filtering of blurred and noisy image: (a) original image, (b) degraded image, (c) optimally filtered image, (d) IBR filtered image (adapted from [Dalton and Dougherty, 2014]).

Notice the basic four steps of translational synthesis in the present context:

1. The model consists of two jointly WSS random processes.
2. The operator class consists of linear filters over an observation window.
3. Optimization: minimize the MSE as defined in Eq. (7.12).
4. The optimization problem is solved by the Fourier transform of the weighting function in terms of the power spectra $S_X(\omega)$ and $S_{YX}(\omega)$.

7.6.2 IBR Wiener filter

Model uncertainty arises in Wiener filtering when either the autocorrelation or cross-correlation function is unknown. For $\theta \in \Theta$, the signal and observation processes are $Y_\theta(s)$ and $X_\theta(t)$, respectively, and the autocorrelation and cross-correlation functions are $r_{\theta,X}(\tau)$ and $r_{\theta,YX}(\tau)$, respectively. The *effective power spectra* are the Fourier transforms of the expected autocorrelation function, $S_{\Theta,X}(\omega) = \mathcal{F}[E_\Theta[r_{\theta,X}]](\omega)$, and the expected cross-correlation function, $S_{\Theta,YX}(\omega) = \mathcal{F}[E_\Theta[r_{\theta,YX}]](\omega)$. While it is easy to write these down abstractly, the difficulty of evaluating them depends on how the observation process is modeled because they can involve complicated integrals.

With model uncertainty, the optimal linear filter has to minimize the expected mean-square error over the uncertainty class, $E_\Theta[\mathrm{MSE}\langle \hat{Y}_\theta(s) \rangle]$. Under rather general conditions the Fourier transform of the weighting function for the IBR Wiener filter is given by

$$\hat{G}_\Theta(\omega) = \frac{S_{\Theta,YX}(\omega)}{S_{\Theta,X}(\omega)} \tag{7.14}$$

[Dalton and Dougherty, 2014]. The form of the filter is the same as when the model is known, except that the characteristics $S_X(\omega)$ and $S_{YX}(\omega)$ are replaced by the effective characteristics $S_{\Theta,X}(\omega)$and $S_{\Theta,YX}(\omega)$.

For the Wiener filter, the second part of the IBR synthesis protocol takes the following form:

5. The uncertainty class is defined in terms of the uncertain parameters in the autocorrelation and cross-correlation functions.
6. A prior distribution is constructed for these parameters.
7. IBR optimization: minimize the expected MSE.
8. The effective characteristics are the effective power spectra.
9. Prove that the IBR optimization problem is solved by replacing the model characteristics by the effective characteristics.

The fundamental part of the protocol is the last step: find conditions under which the solution to the IBR optimization is solved by replacing the characteristics in the ordinary solution with effective characteristics—and prove it.

A suboptimal Bayesian approach to filtering under model uncertainty was first taken in the case of nonlinear filtering of digital binary images [Grigoryan and Dougherty, 1999] and then for linear filtering of random signals as considered here [Grigoryan and Dougherty, 2001]. These solutions were suboptimal because they restricted filter selection to a filter that is optimal for at least one model in the uncertainty class. An intrinsically Bayesian robust linear filter, where there is no such constraint, was solved more recently. Interestingly, full optimization via the IBR paradigm is mathematically less complex than the suboptimal solution—once conditions are found so that ordinary characteristics can be replaced by effective characteristics. As is often the case in mathematics, framing a problem "correctly" makes the solution transparent.

7.6.3 A more general synthesis protocol

There is a long history of robust Wiener filtering under model uncertainty. The problem was first treated in the form of *minimax* optimization, where the aim was to find a filter having best worst-case performance: if possible, find a linear filter that has the minimum maximum MSE over all models in the uncertainty class [Kuznetsov, 1976; Kassam and Lim, 1977; Poor, 1980]. Minimax robustness is conservative. The drawback is that it can be overly conservative, especially if the uncertainty class is large. From a probabilistic perspective, a minimax robust filter can be overly influenced by outlier models because it does not take into account a prior (or posterior) distribution on the uncertainty class. To place minimax robust filtering into a translational synthesis framework, step 6 of the IBR synthesis protocol is omitted and the optimization of step 7 becomes minimization of the maximum MSE instead of the expected MSE.

Considering translational synthesis from a general perspective, a cost function is introduced based on minimization of the original full-model cost function relative to the uncertainty. In this view, IBR optimization has cost function $C_\Theta(\psi) = E_\Theta[C_\theta(\psi)]$ and minimax robust optimization has cost function $C_\Theta(\psi) = \max_\Theta\{C_\theta(\psi)\}$. From a completely general perspective, steps 6 through 9 of the IBR synthesis protocol reduce to

6′. Choose a cost function on the uncertainty class.
7′. Optimization: minimize the cost function over the uncertainty class.
8′. Find conditions under which the optimization problem can be solved.

The IBR synthesis protocol is a special case of this general synthesis protocol. As stated, the IBR protocol assumes that IBR optimization will take the form of effective characteristics. While this has been the case thus far, it may turn out that for some synthesis problems an IBR operator will not be defined in terms of effective characteristics. Then IBR synthesis will fall into the more general paradigm with cost function $C_\Theta(\psi) = E_\Theta[C_\theta(\psi)]$.

7.7 Objective Cost of Uncertainty

The IBR principle is to find an operator (classifier, filter, structural intervention, etc.) that, based on a cost function, is optimal over an uncertainty class relative to a prior (or posterior) distribution reflecting the state of our knowledge regarding the underlying physical processes. While an IBR operator is optimal over the uncertainty class Θ, it is likely to be suboptimal relative to the full model. This loss of performance is the cost of uncertainty.

To quantify this cost, for $\theta \in \Theta$, let C_θ be the cost function applied on model θ and let ψ_θ be an optimal operator for θ. Then $C_\theta(\psi_\theta) \leq C_\theta(\psi)$ for any operator ψ. Let ψ_{IBR} be an IBR operator for Θ. Owing to the optimality of the IBR operator over the uncertainty class, $E_\Theta[C_\theta(\psi_{\text{IBR}})] \leq E_\Theta[C_\theta(\psi)]$ for any operator ψ. An IBR operator is optimal over Θ; however, there is a cost to this choice relative to applying the optimal operator for θ on θ because $C_\theta(\psi_\theta) \leq C_\theta(\psi_{\text{IBR}})$ for all $\theta \in \Theta$.

For any $\theta \in \Theta$, the *objective cost of uncertainty* (OCU) relative to θ is the cost differential between an IBR operator and an optimal operator for θ applied on θ:

$$\text{OCU}(\theta) = C_\theta(\psi_{\text{IBR}}) - C_\theta(\psi_\theta). \tag{7.15}$$

The cost of uncertainty relative to the full model is $\text{OCU}(\theta_{\text{full}})$, where θ_{full} is the value of θ for the full model; however, since the full model is unknown, this quantity cannot be calculated. Thus, as the basic quantification of uncertainty we use the *mean objective cost of uncertainty* (MOCU):

$$\text{MOCU}(\Theta) = E_\Theta[\text{OCU}(\theta)] = E_\Theta[C_\theta(\psi_{\text{IBR}}) - C_\theta(\psi_\theta)] \tag{7.16}$$

[Yoon, et al., 2013]. If there is no uncertainty, then the uncertainty class contains only one model and $\text{MOCU}(\Theta) = 0$; however, the converse is not true.

From a scientific perspective, one might prefer to use the entropy of the prior (or posterior) distribution because it measures uncertainty with respect to the model; however, entropy does not focus on the translational objective. There may be large entropy but with most (or all) of the uncertainty irrelevant to the objective. For instance, in controlling a network there may be much uncertainty in the overall network but a high degree of certainty regarding the mechanisms involved in the control. In this case, the entropy might be large but the MOCU be small, which is what matters from a translational perspective. Because the MOCU is intrinsic to the translational system, given our knowledge and objective (cost function), it quantifies the uncertainty in our knowledge with respect to our objective and therefore is an epistemological parameter.

Knowledge can be increased by generating data to produce a new posterior distribution. If there is a collection of possible experiments that can supply information relating to the unknown parameters, which experiment should be performed first? (Here we ignore time and cost but these can be factored in if desired.) Since the MOCU quantifies the average lack of optimality owing to uncertainty, a reasonable course of action is to choose an experiment from the space of possible experiments that yields the minimum expected MOCU given the experiment [Dehghannasiri, et al., 2015]. This requires, for each possible experiment, computing the MOCU for every possible outcome of the experiment, averaging these MOCU values, and then taking the minimum of these averages over all possible experiments. The result is *optimal experimental design* relative to the objective uncertainty. This can be done in an iterative fashion, at each stage choosing an optimal experiment, running the experiment, updating to a new posterior distribution, re-computing the MOCUs, determining an optimal experiment, and so on.

Figure 7.9 illustrates the benefit of optimal experimental design in the context of IBR structural intervention (Section 7.3.1). Five parameters in a mammalian cell cycle network are randomly selected to be unknown; two sequences of five experiments are simulated, one in which the experiments are randomly chosen and another in which they are chosen via an optimized iteration; at each step of each sequence the total undesirable steady-state probability is computed for the IBR structural intervention; this procedure is repeated a number of times; and the average undesirable probabilities are computed and plotted on the vertical axis. The advantage of optimal experimental design is clear: on average, the objective knowledge gained from the first two optimally chosen experiments is equivalent to that gained via four randomly chosen experiments.

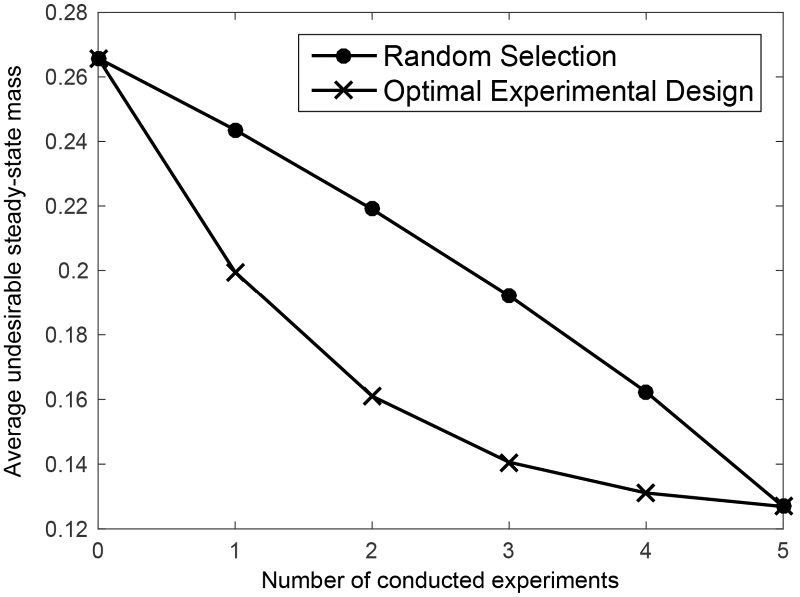


Figure 7.9 Random versus optimal experimental design (adapted from [Dehghannasiri et al., 2015]).

7.8 Small-Data Epistemology

The current crisis in scientific epistemology results from a severe lack of data in relation to the complexity of the systems that people wish to model. Although "Big Data" is the buzzword, the profound problem for science and engineering is small data. There is insufficient data for validation and insufficient data for estimating model parameters. In the present chapter we have taken the view that insufficient data for estimation can be framed in terms of uncertainty classes with prior knowledge represented via a prior distribution over the uncertainty class and translational operator design optimized relative to the posterior distribution.

What kind of knowledge is this prior knowledge? Consider a numerical (not a vector) model parameter θ and suppose, based on an accepted scientific theory, it is deduced that $a \leq \theta \leq b$. For instance in a multi-dimensional normal model, θ might represent the correlation between two features and in the physical system from which the features are constructed it may be that $0 \leq \theta \leq 0.5$. Absent more knowledge concerning θ, we have taken the view that a uniform distribution over the interval $[a, b]$ is a suitable form of prior knowledge. θ is what it is, and that we do not know. True, we know that it is between a and b, but it is not uniformly distributed over $[a, b]$. Saying that θ possesses a uniform prior distribution over $[a, b]$ is not a statement pertaining to the actual value of θ.

Essentially, a prior distribution is a pragmatic construct based on belief as to where a parameter is located. It is clearly advantageous to have scientific knowledge that constrains the parameter and thereby constrains its prior distribution. Since a prior is a construct, not validated scientific knowledge, the more it is constrained by scientific knowledge and the more experience one has with the physical system, the more confident one can be that the prior distribution is concentrated around the full model. If one has confidence, then a tight prior is preferable because tighter priors require less data for good performance; however, there is risk because a prior distribution whose mass is concentrated away from the true parameter will perform worse than one that is uniform. These issues have long been discussed in the Bayesian literature.

In 1946, Harold Jeffreys proposed a uniform prior, referred to as *Jeffrey's prior* [Jeffreys, 1946]. Objective-based methods were subsequently proposed, a few early ones being [Kashyap, 1971], [Bernardo, 1979], and [Rissanen, 1983]. The principle of maximum entropy can be seen as providing a method of constructing least-informative priors [Jaynes, 1957, 1968]. These methods are general and do not target any domain-specific type of prior information. More targeted approaches can be constructed that integrate scientific knowledge specific to the problem at hand. For instance, in relation to the pathway knowledge we have utilized in the p53 and mammalian cell cycle networks, one can construct a prior distribution quantifying and integrating prior knowledge in the form of signaling pathways [Esfahani and Dougherty, 2014]. In 1968, E. T. Jaynes remarked, "Bayesian methods, for all their advantages, will not be entirely satisfactory until we face the problem of finding the prior probability squarely." [Jaynes, 1968] The problem remains.

Yet we must remember that for translational science the prior distribution is a construct to facilitate operator design. Given a cost function, an IBR operator is optimal on average relative to the prior (or posterior) distribution but our real interest is an operator that is optimal relative to θ_{full}, the value of θ for the full model. Only rarely will an IBR operator be optimal for θ_{full}. Can an IBR operator somehow be "validated" on the model corresponding to θ_{full}? Strictly speaking, the question makes no sense if it means to show that an IBR operator is optimal for θ_{full}; indeed, we expect it not to be optimal for θ_{full}, which we do not know. An IBR operator has no direct connection to the full model. It is only related via the prior (or posterior) distribution.

Intrinsically Bayesian robust operators cannot cure the small-data epistemological problem for the complex systems that modern engineering wishes to study and control. What they can do is place operator design under uncertainty in a rigorous optimization framework grounded in an infrastructure utilizing prior knowledge and data, while providing uncertainty quantification relative to a translational objective at the level of the underlying processes. The deep problem is that there appears to be no way to objectively transform existing knowledge into a prior distribution. Although there are ways to construct a mathematically rigorous transformation, these ultimately involve subjective considerations.

Within the bounds set by existing scientific knowledge, the formalization of uncertainty, which is the prior distribution, must be constructed via subjectively imposed criteria. This is similar to the basic epistemology of prediction, since in the latter, even though the model and experimental protocol are inter-subjective, the decision whether to accept or reject a theory depends on subjective criteria; nevertheless, with model uncertainty the situation is more unsettling because it is not even clear that the notion of predictive validation can be tied to observations. If, however, we take the perspective that when application is primary and doing nothing is, in fact, a decision, then at least if one follows a formal translational science optimization protocol the overall procedure will be inter-subjective even though there may be disagreement regarding the criteria imposed for construction of the prior distribution. Subsequent to that construction, the prior distribution and cost function jointly form a hypothesis from which an optimal operator can be deduced.

All men are mortal.
Socrates is a man.
Therefore, Socrates is mortal.

But are all men mortal?