Validation of systems biology models
Hasdemir Durmus, D.

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: http://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
Introduction

Systems biology is a highly inter-disciplinary modern approach to the analysis of living systems. It employs an integrative approach where the focuses are the interactions between biological entities and their operation as a system. The typical systems biology approach incorporates modeling as a fundamental tool. It utilizes various modeling approaches and computational tools from a large selection of fields such as statistics, control theory and mathematics.

With the advancements in data acquisition and the increasing recognition of the systems biology concept, a substantial number of models have been introduced that aim to explain various biological systems. Nevertheless, both the structures and the parameters of these models are prone to uncertainty. This arises from the fact that the data used to obtain the models are noisy and the biological knowledge regarding the systems under study is incomplete. This brings the need for careful model validation and selection. However, the importance of validating models is often underestimated in the field. The major reason is that the employment of the modeling concept itself has recently started to become such prevalent and the concept of validation is still one step behind. While we witness the maturation of systems biology models, guidelines that are tailored for their validation has to be introduced in the field. In this respect, resampling strategies are worth attention.

Resampling strategies have been extensively used to assess the validity and stability of models and to infer the precision of estimated model parameters in many fields. These include but are not limited to statistics, chemometrics and machine learning. Cross validation and bootstrapping, two well known resampling approaches have
been applied previously also in systems biology. However, they deserve more attention and can be exploited further. When appropriately adapted for the characteristics of the data and for the models encountered in the field, they can provide the guidelines needed for model validation and selection.

The goals of this thesis are:

- Developing methods for the assessment of the reliability and informative levels of models,
- Presenting guidelines for reliable model validation and selection,
- Finally, incorporating resampling strategies for the purposes above regarding extensively employed systems biology models.

In Sections 1.1 and 1.2 we present background information regarding the types of systems biology models that are the focus of this thesis and the resampling strategies that have been applied in the systems biology concept, respectively. In Section 1.1, we also detail our goals that are specific to each type of model. In Section 1.2, we have a special focus on application examples of bootstrapping and cross validation in systems biology. We exclude the application of bootstrapping for cluster analysis and the application of cross validation for the analysis of ODE based kinetic models, since we discuss them further in the introductory sections of Chapters 5 and 3, respectively.

1.1. Common modeling approaches in systems biology

Models in systems biology can be categorized in two basic classes. The first type is a top-down model which is constructed by the analysis and summarization of large scale data such as transcriptomics, proteomics and metabolomics data. Examples to these are network models inferred from large scale data or statistical models such as cluster analysis. The other type of model is a bottom-up model that is built upon physical and biochemical knowledge regarding the individual parts of the system at a molecular level [25]. Kinetic nonlinear models of biochemical systems are typical examples of this type. Either type of models are important means for investigating the biological system as a whole and hence, are pivotal to systems biology.
1.1. Common modeling approaches in systems biology

1.1.1. ODE based kinetic models

Modeling using differential equations is commonly applied in systems biology for deterministic systems explained by biochemical laws. Primary examples to these systems are signaling and metabolic pathways. In these models, concentration behavior of biochemical species (state variables) is governed by mathematical formulations which follow the laws of biochemical kinetics such as Michaelis-Menten, mass action or Hill kinetics. The independent variable in the model is often time, additionally complemented with location in the cell leading to partial differential equation models when the spatial distribution of the variables is also important. When the independent variable is only the time, ordinary differential equations (ODE) are sufficient to describe a system deterministically.

Model parameters are kinetic constants used in the formulation of the kinetics involved. When these parameters are known, the time dependent behavior of the state variables can be achieved by solving the ODE system and later, can be used for predictive purposes. Theoretically, these parameter values can be determined by in vitro experiments with isolated enzymes. However, not all of them are applicable under the in vivo conditions of the cell. Therefore, in vivo time series measurements of the species that are involved in the model are commonly used to infer the unknown parameters [10]. This is accomplished by minimizing the difference between the measurements and the concentration values obtained by the model. During parameter inference by this approach, a fixed model structure is assumed to be the correct structure. In other words, the components of the model and the governing kinetic laws are assumed to be known. However, this assumption can never be entirely fulfilled. Therefore, modelers have to keep in mind the uncertainty of the model structure such as the possibility of putative feedback loops or allosteric regulation [100, 128].

This brings the need for model validation and selection. Proper handling of the data for parameter inference and model validation tasks is essential. In Chapters 2 and 3, we develop guidelines for this purpose by exploiting cross validation.

1.1.2. Network models

Graph theory provides a framework in the systems biology perspective for the analysis of interactions between entities. The framework represents the biological system as a network. Depending on the system studied, the nodes in the network are often proteins such as transcription factors or enzymes, genes and metabolites.

Protein-protein interaction networks are one of the most studied network type which include proteins as nodes and the physical interactions between the proteins...
as edges [144]. Often, the interactions in these networks are nondirectional such as the cases of protein complexes and hence, no directionality has to be imposed on the edges. Protein-protein interaction networks are usually obtained by combining information from different sources such as yeast two hybrid experiments, co-immunoprecipitation assays and computational prediction of interactions by structural modeling. There are also studies that model signaling pathways using directional protein interaction networks and employ graph theoretical analysis [8, 41].

An example network model where directionality has to be imposed is a metabolic network model. In metabolic networks, the edges usually correspond to chemical reactions between the metabolites [119]. In other example networks, both enzymes and metabolites can be the nodes. Then, the edges imply the enzyme’s role in the chemical reactions producing/consuming the metabolites it is connected to [120]. Network modeling is also used to summarize top-down information and build correlation networks. In these, metabolites with similarities in terms of their steady state or time-resolved behavior can be linked [28, 62]. In a similar way, gene networks may also be constructed by linking similarly expressed genes to each other [87]. However, the term transcriptional regulatory networks or transcription networks have traditionally been reserved for another type of network model of genes which we explain later.

Biological networks have certain properties that discriminate them from any random network. Most importantly, they are scale-free, a property that characterizes the degree distribution of the nodes [3]. Degree of a node means the number of nodes it is linked to, in other words, its neighbors. As a result of being scale-free, biological networks typically contain a small number of nodes with very high degree, called the hubs and a large number of nodes with low degree. Many studies focus on identifying the topological properties of biological networks such as the hubs of the network, the mean degree in the network, average distance between the nodes (diameter) and the distribution of the clustering coefficients. In such studies where the focus is on static analysis, the network motifs are also crucial. Network motifs are local patterns in the network that occur significantly common compared to other patterns [96]. For example, motifs in metabolic networks may correspond to abundant reaction types or motifs in a protein interaction network can reveal abundant protein complex structures. Identifying the topological features of a network increases our understanding of the big picture of the biological systems that we investigate. Network based modeling framework is not restricted to static analysis, though. A Boolean network model is an example of network modeling approach that links time points and allows discrete dynamic modeling [163].
1.1. Common modeling approaches in systems biology

Transcriptional regulatory networks

Transcriptional regulatory networks consist of transcription factors, genes and the regulatory interactions between those [12, 96]. Transcription factors are proteins that bind to the promoter regions upstream of the coding regions of the genes. They regulate the expression of genes by either promoting or repressing the transcription of the mRNA. Post-transcriptional modification of gene expression is also possible by micro-RNAs which bind to the transcribed mRNA itself [45]. However, post-transcriptional modification is not a part of these models since the focus here is in the regulation of transcription.

Protein-DNA interactions can be represented by adjacency matrices. Such matrices can be binary in which a 1 is assigned in the respective place if a certain transcription factor regulates the transcription of a certain gene. Adjacency matrices can also consist of numbers which correspond to the strength of the interaction between transcription factors and genes. The strength of the interaction affects the degree of the influence that the transcription factor has on the regulation of the transcription [99].

Physical interactions between the DNA and the binding proteins are detected by a series of different techniques. Most importantly, chromatin immunoprecipitation experiments coupled with microarrays (ChIP-Chip) or coupled with high throughput sequencing (ChIP-seq) allow the genome wide detection of DNA binding. Other techniques involve yeast one hybrid systems or X-ray crystallography. Sequence based computational prediction is also possible. Prior information such as co-clustering of genes can be incorporated in such an approach. As an example, sequence based analysis in the upstream regions of co-clustered genes was shown to reveal putative binding sites in the yeast genome [145].

High throughput approaches employed in the discovery of transcriptional regulatory interactions lead to substantial amount of uncertainty in the proposed network structures due to experimental artifacts. Therefore, selection between different networks and refinement of the topological structures both computationally and experimentally are of high importance. In Chapter 4, we deal with these challenges by exploiting the detection of the inconsistencies between the network topology and the associated expression data of the genes in the network.

1.1.3. Cluster analysis

Cluster analysis is a data analysis method used for grouping similar objects together in an unsupervised manner. Similarity is measured by employing a distance metric. An appropriate distance metric has to be chosen depending upon the aim of the cluster analysis and the structure of the data. Distance metrics are either geometr-
rically defined such as the Euclidean distance, corrected for correlation between the
variables as the Mahalanobis distance or based on correlation such as the Pearson
correlation coefficient.

A well known clustering algorithm encountered in the analysis of biological data
is hierarchical clustering [44, 84]. Hierarchical clustering depend on the pairwise
similarity between the object across different variables. Hierarchical clustering al-
 lows the formation of different number of clusters at different hierarchical levels:
in other words, at different levels of distances between the objects. At the highest
level of the hierarchy, there is a single cluster that contains all the objects. Vi-

sual inspection of the hierarchy between the clusters is possible with a dendrogram
presentation, making it popular for biological applications.

Many other clustering algorithms fall under the partitioning algorithms class
where the distance of an object to the cluster centroid is important rather than the
pairwise similarity of hierarchical clustering. A major example is k-means clustering
in which k clusters are formed through the minimization of an objective function.
The objective function is often the sum of squares of the distances of the objects to
the centroids of the clusters. Further assumptions about the clusters can be made
regarding the distribution of the objects in the clusters, leading to model based-
clustering. Gaussian mixture models clustering is one of the most fundamental
model-based clustering algorithms in which data is assumed to consist of Gaussian
clusters defined by a mean vector and a covariance matrix.

Fuzzy clustering is another type of algorithm applied in systems biology appli-
cations. In this framework, clusters are not necessarily distinct. Therefore, objects
can be assigned to multiple clusters with varying levels of membership [51]. Network
based clustering approaches have also been popularly applied for the detection of
densely connected nodes in network models [11].

Clustering is extensively used in the analysis of genome wide gene expression
data. Genes are grouped together based on the similarities of their expression lev-
els across different experimental conditions, patients or time points. Co-clustering
genes give hints on the common regulatory rules acting on them, shedding light on
their functional characteristics. A cluster analysis is often followed by a functional
enrichment analysis which identifies the most dominant functional categories in each
cluster.

Usually, the cluster analysis is assumed to be useful if the results from the fol-
lowing enrichment analysis lead to biologically meaningful explanations. However,
the unbiased assessment of the validity of a cluster analysis without depending upon
biological information is essential. In Chapter 5, we deal with the validation of
k-means clustering analysis through the assessment of its stability.
1.2. Resampling strategies

1.2.1. Bootstrapping

Bootstrapping which was originally proposed by Efron [42] is employed to infer the precision of statistical estimates such as confidence intervals, variance and bias. It provides answers in cases where analytically derived formulas do not exist due to complex estimation procedures. It is also helpful in situations where the assumptions regarding the underlying distribution of the data needed for traditional statistical inference can not be fulfilled. The assumptions behind bootstrapping are rather relaxed [43, 76, 115], making its application suitable for such a situation. There is substantial literature regarding bootstrapping that constructs guidelines for applied statisticians on its implementation [42, 43, 134, 165].

Bootstrapping can be summarized as follows. Data actually consist of observed random samples from an unknown probability distribution. Parameters of a model (e.g. kinetic parameters in an ODE model) are estimated using the data. The variability of a parameter estimate around the true value of that parameter is mimicked by the variability of the parameter estimates inferred from the bootstrap samples around the parameter estimate inferred from the original data (observed value of the parameter). This is the basic idea behind using bootstrapping to calculate the precision of statistical estimates. In other words, bootstrapping provides us a means of repeating the random sampling process with replacement. How do we then construct the so-called bootstrap samples? In nonparametric bootstrapping, bootstrap samples are drawn from an empirical distribution of the samples which represents the true unknown probability distribution of the data. In parametric bootstrapping, a certain underlying distribution is assumed to be true and bootstrap samples are drawn from this parametric estimate of the population.

The most critical step in bootstrapping is obtaining the appropriate empirical distribution or the appropriate parametric estimate for an unknown probability distribution. This can be stated as obtaining the appropriate bootstrap samples. Bootstrap sampling needs more tedious work and is more error-prone when complicated and dependent data structures are involved [43, 64]. An example is time series data which is widely encountered in biological modeling. In the rest of Section 1.2.1, we summarize the application areas of bootstrapping in systems biology with a focus on the techniques used for obtaining bootstrap samples of time series data.
Bootstrapping for obtaining parameter confidence intervals in ODE based models

Data taken at different time points depend on each other. Therefore, bootstrapping has to be adapted accordingly. One approach is to bootstrap from the residuals in a parametric way [76] since unlike the data, the residuals are not essentially dependent. In this approach, the residuals are bootstrapped from a parametric distribution with zero mean and constant standard deviation. The standard deviation of the residuals at a data point is equal to the standard deviation inferred from the replicates of data at that point. Then, the bootstrapped residuals are added to the means of the replicate data to obtain a bootstrap data point. This procedure gives a set of new data points for the biochemical species over time which constitutes one single bootstrap sample. In this way, a sufficient number of bootstrap samples which would be enough for convergence are created. Convergence is detected by plotting the mean of parameters. If the means of parameters which are obtained by different bootstrap samples are not affected anymore by increasing the number of samples, then the samples are assumed to have converged. 1000 to 3000 samples were reported to be sufficient in a number of studies, but it depends on the complexity of the model [76, 82, 130]. Later, models are fitted to the bootstrap datasets to obtain the bootstrap estimates. The confidence intervals (CI) of parameters are then calculated from the percentiles of the bootstrap estimates. Another alternative for the last step is the use of the bias-corrected accelerated (BCa) intervals since they exhibit better compatibility with exact intervals [33]. The details are outlined in Figure 1.1. In [76], the approach was proved to be valid by showing narrower confidence intervals resulting from a traditional D-optimal experimental design that aims at better estimation of the parameters.

An essential step in the approach outlined above is to identify the type of the error in the data. For example, in [33], an error model in which error is dependent on the log transformed data is used. This is apparently dependent on the knowledge of the experimental procedure by which the data was obtained. In general, an error model where the error is dependent on the mean of the data would be appropriate for biochemical assays.

The bootstrapping approach is superior to the traditional Fisher information matrix (FIM) based approach from certain aspects. Calculation of the FIM requires linearization of the system which can result in inaccurate findings for a nonlinear ODE model. Theoretical confidence intervals obtained by the FIM method are usually unrealistically smaller and essentially symmetric [76, 129]. However, bootstrapping gives more realistic intervals.

Another way of bootstrapping time series data is based on modeling the depen-
1.2. Resampling strategies

resampling strategies

1. Draw L bootstrap samples:

\[ Y_{ij} = \bar{Y}_{ij} + \epsilon \times Y_{ij} \text{ where } \epsilon \sim \mathcal{N}(0, \hat{\sigma}_{rel,i}^2) \]

where \( L \) = number of bootstrap samples needed for convergence.

2. Estimate parameter vectors from bootstrap samples: \( \hat{\theta}_1, ..., \hat{\theta}_L \)

3. Determine confidence intervals for each parameter by the Percentile Method:

\( (\hat{\theta}_{lo}, \hat{\theta}_{up}) = (\hat{\theta}_{l(\alpha/2)}, \hat{\theta}_{l(1-\alpha/2)}) \)

where \( \hat{\theta}_{l(\alpha/2)} \) is the 100.(\( \alpha/2 \))th percentile of \( L \) bootstrap replications.

Figure 1.1: Calculation of confidence intervals by bootstrapping. The approach that is visually outlined here can be considered as a basic application guideline of bootstrapping tailored for nonlinear ODE models and time series data.

decision in the data. One way to deal with it is using the Gaussian process regression method [82]. This nonlinear regression method from the class of Bayesian techniques does not assume a fixed form of the regression function. It models the underlying time dependent function as a collection of random variables with a mean and a covariance function. The priors for the covariance function is especially important in using such a method since it presents the knowledge on how the data at different
time points co-vary with each other. Once the underlying function is estimated, bootstrap samples can be drawn from this distribution and confidence intervals can be calculated by repeated parameter inference on the bootstrap samples.

Bootstrapping for model comparison in ODE based models
The following major step after parameter estimation is to test whether the model explains the data well enough. $\chi^2$ tests can be used to assess the quality of the data-model fit [29]. The hypothesis test will help to decide if the residuals from a model are small enough for the model to be accepted as adequate. This is the first step in which a model can be rejected and the modeler is forced to come up with an alternative model structure. Here, the test statistic that is expected to come from a $\chi^2$ distribution is the likelihood. Likelihood of a model is the probability of the model to be the true data generating process given the observed data. Model residuals may usually be used instead of likelihood for practical purposes. This replacement is valid also for likelihood ratio tests which are used to compare two models.

Likelihood ratio tests are used to assess the superiority of a given model against an alternative model. The test statistic is the observed difference of the residuals from the two models and follows a $\chi^2$ distribution if the two models are nested and linear. However, the models that are being compared might not be nested. More fundamentally, dynamic ODE based models are typically nonlinear. There are also additional issues such as limited data availability, lack of full parameter identifiability and positivity constraints on the parameters that violate the Gaussian distribution of the parameters. These issues are encountered commonly in this model type [115]. Therefore, there is no clue on the nature of the distribution from which the observed test statistic comes. Sometimes, the distribution is even a mixture of multiple distributions. Therefore, it has to be determined empirically. This can be achieved by bootstrapping.

There is a consensus guideline for the application of bootstrapping in constructing the empirical distribution [65, 148, 160, 169]. However obtaining the empirical distribution comes at a price. Unlike non-bootstrapped likelihood ratio test, the bootstrapped version can not provide a full model selection process because both models can be selected or both can be rejected following the test [29]. Consequently, the model comparison task turns into two parallel model rejection tasks. However the test is still advantageous. It can have more power compared to a single $\chi^2$ test for the rejection of a single model if the alternative model has certain characteristics such as being not too flexible and not too rigid [74]. The alternative model is called a 'help model' and it favors the testing of the model that is primarily in question.
Simulations using a two dimensional $\chi^2$ test in [74] demonstrates this issue.

Estimate parameter vectors $\hat{\theta}_{n=1}$ and $\hat{\theta}_{n=2}$ for the two model structures. $\mathbf{R}_{n=1}$ and $\mathbf{R}_{n=2}$ are then the residual matrices from the model fit. $\mathbf{R}_{ij,n=1}$ and $\mathbf{R}_{ij,n=2}$ denote the residuals at each data point. $\mathbf{r}_{n=1}$ and $\mathbf{r}_{n=2}$ are sum of squares of the residuals. Observed test statistic $T_{obs} = r_{n=1} - r_{n=2}$

Generate two distinct sets of bootstrap samples using both models. $\mathbf{Y}_{i,j,p=1} = \hat{\mathbf{Y}}_{ij,n=1} + \epsilon * \hat{\mathbf{Y}}_{ij,n=2}$ where $\hat{\mathbf{Y}}_{ij,p}$ is the value at the corresponding data point generated assuming model n is true.

Infer parameters for L bootstrap samples. Calculate the residual differences: $T_{p=1} = r_{p=1,n=1} - r_{p=1,n=2}$ Plot the empirical distribution of $T_{p=1}$ values.

According to the consensus approach, two distinct sets of bootstrap samples are generated by assuming each model is the true model in turn. Fitting both models to both sets of samples and calculating the residual differences of two models for each
bootstrap sample gives us the two empirical distributions of residual differences. The final decision can then be given by comparing the observed residual difference which was inferred from the original data to the two distributions. In Figure 1.2, we describe this approach which has been reported to perform better than alternative bootstrapping strategies [57, 74].

Bootstrapping for detecting the reliability of network models

Bootstrapping has been applied to assess the reliability of network models, particularly networks of genes [69, 81, 82]. An example in this respect [82] deals with two types of correlation networks. The first type is a gene relevance network which include edges based on the correlation between genes’ expression profiles. The second type is a graphical Gaussian model which include edges based on partial correlation. Both networks are inferred from time series data. Nevertheless, the inferred topology is not completely robust to uncertainty in the data that results from experimental noise. Gaussian process regression bootstrapping on time series data is employed to quantify the uncertainty in the topology by calculating the confidence intervals for the inferred edges, as previously explained also for the ODE based models. Another example [69] employs bootstrapping for robust estimation of regulatory interactions in a Bayesian network which can be considered as a probabilistic gene network.

1.2.2. Cross validation

Cross validation has been used in systems biology studies in which large scale data are analyzed using supervised methods. Compared to bootstrapping, its application area has been traditionally narrower in the field but it has been more commonly applied. We see primary applications in gene expression data analysis.

Cross validation for classification

Building classifiers using gene expression data is extensively applied in especially cancer research. In this research area, genome wide mRNA levels are selectively used as features for classifiers which are trained to classify sample tissues in terms of prognosis and future possibility of metastasis. Testing the validity of these classifiers on independent data leads to reliable biomarkers for diseases. This can be accomplished by using cross validation. Various cross validation techniques have been employed such as k-fold, leave-one-out, hold-out and Monte-Carlo cross validation.

In leave-one-out cross validation (LOOCV), each sample is left out of the training step once and the discrimination performance of the classifier is tested on the left
out sample. LOOCV has been reported to contribute to the selection of biomarkers and classification of tissues, following a support vector machines classification to select lymphoma subsets [121], following a random forests classification [35] or to predict metastases in breast cancer [156].

Guidelines for the appropriate application of cross validation have also been elaborately discussed. The authors of [139] present adjustments to the technique to correct for the problems arising from small sample size. Furthermore, the bias-variance behavior of different cross validation methods have been documented with applications on various types of supervised classification [113].

In addition to gene expression, large scale proteomics and metabolomics studies have benefited from cross validation based approaches. Double cross validation utilizes an additional cross validation loop within the outer cross validation loop and thus provides the complete separation of the test set samples from those used to train the meta-parameters such as the number of components needed in a principal component analysis model [141]. It has been applied in proteomics and metabolomics studies following supervised classification by using principal component discriminant analysis [141] and partial least squares discriminant analysis [167], respectively.