Computational approaches for the analysis of RNA–protein interactions: A primer for biologists

Kat S. Moore and Peter A. C. ’t Hoen

From the Department of Hematopoiesis, Sanquin, and Landsteiner Laboratory AMC/UvA, 1066 CX Amsterdam and the Centre for Molecular and Biomolecular Informatics, Radboud Institute for Molecular Life Sciences, Radboud University Medical Center, 6500 HB Nijmegen, The Netherlands

Edited by Karin Musier-Forsyth

RNA-binding proteins (RBPs) play important roles in the control of gene expression and the coordination of different layers of post-transcriptional regulation. Interactions between certain RBPs and mRNAs are notoriously difficult to predict, as any given protein–RNA interaction may rely not only on RNA sequence, but also on three-dimensional RNA structures, competitive inhibition from other RBPs, and input from cellular signaling pathways. Advanced and high-throughput technologies for the identification of RNA–protein interactions have come to the rescue, but the identification of binding sites and downstream functional effects of RBPs from the resulting data can be challenging. In this review, we discuss statistical inference and machine-learning approaches and tools relevant for the study of RBPs and the analysis of large-scale RNA–protein interaction datasets. This primer is intended for life scientists who are interested in incorporating these tools into their own research. We begin with the demystification of regression models, as used in the analysis of next-generation sequencing data, and progress to a discussion of Hidden Markov Models, which are of particular value in analyzing cross-linking followed by immunoprecipitation data. We then continue with examples of machine learning techniques, such as support vector machines and gradient tree boosting. We close with a brief discussion of current trends in the field, including deep learning architectures.

Large-scale identification of RNA–protein interactions

RNA-binding proteins (RBPs)\(^1\) may affect the translation of bound transcripts, facilitating or preventing the recruitment of ribosomes and translation initiation factors. They may also affect mRNA transcript stability, localization, and (alternative) splicing. As such, a single RBP can have a broad array of cellular roles, the disruption of which can have far-ranging consequences. A single RBP may influence tens of thousands of target transcripts; for instance, a recent study identified over 55,000 putative transcripts in the regulatory network of Elavl1 (HuR) \(^1\). Although in vitro binding assays, such as RNAcompete and RNA Bind-n-Seq, have been found to be useful for identifying both sequence and structural binding motifs in the RNA, the actual binding of the protein to the RNA is very much cell- and environment-dependent \(^2\), \(^3\). Several high-throughput sequencing technologies have been developed in recent years to identify specific RBP–RNA interactions on a large scale, and this has led to a better understanding of the effect of RBPs on gene expression.

The aforementioned sequencing approaches are modified RNA-sequencing (RNA-seq) protocols. In RNA-seq, large and complex pools of RNA are quantitatively amplified, mapped to a reference genome or transcriptome, and analyzed for differential expression. This can be done on total cellular mRNA or applied to specific subsets of RNA, such as ribosome-associated transcripts. RNA-seq was adapted for discovery of protein–RNA interactions. The technologies available can be broadly divided into three main techniques: RNA-immunoprecipitation followed by high-throughput sequencing (RIP-seq); digestion-optimized RIP-seq (DO-RIP-seq); and cross-linking followed by immunoprecipitation (CLIP-seq) (Fig. 1) \(^4\). All of these technologies involve the incubation of cell lysates with an antibody directed toward the RBP. Alternatively, the RBP may be modified with a small biotinylation motif to utilize the strong affinity of the streptavidin–biotin interaction \(^5\), \(^6\). RNA not bound to the RBP is subsequently washed away prior to sequencing.

RIP-seq and DO-RIP-seq involve the isolation of RNA–RBP complexes without a cross-linking step. Both techniques require careful optimization of washing conditions, which must be stringent enough to minimize background without removing specific RBP–transcription interactions. CLIP-seq techniques circumvent the issue by introducing a covalent link between the RBP and the bound transcript. However, RIP-seq has the advantage of preserving native binding conditions, whereas cross-linking may also stabilize adventitious binding. Furthermore, traditional UV cross-linking at 254 nm is subject to
biases: pyrimidines are more photoactive than purines, and reactivity among amino acid residues is variable (4). Low-affinity RBPs are especially subject to false cross-linking events, which makes them poor candidates for CLIP methods (7). Conversely, RBPs with high-transcript affinity are well-suited to cross-linking techniques.

Standard RIP-seq protocols are not able to identify specific binding sites within a transcript. DO-RIP-seq and CLIP-seq technologies apply a partial nuclease digestion to create short RBP-protected fragments that can undergo adapter ligation and size selection with PAGE (8, 9). This allows greater specificity in determining the binding location of the RBP on the transcript. There are many variations on the CLIP-seq protocol; this review will briefly cover a few of the most popular. High-throughput sequencing CLIP (HITS-CLIP) makes use of cross-link–induced mutations to identify the specific residue that marks the protein–transcript interaction (10). HITS-CLIP utilizes both 5’ and 3’ adapters for reverse transcription. Reverse transcriptases (RT) may prematurely terminate due to modifications at the cross-linking site, resulting in the loss of the 5’ adapter and the exclusion of those sequences from the library (4, 11). A similar protocol, individual nucleotide-resolution CLIP (iCLIP), solves this problem via a second round of adapter ligation. The resulting library will preserve all sequences regardless of RT termination, and, because a subset of reads will halt at cross-linking sites, allows the identification of binding sites at single nucleotide locations (4, 12). Photoactivatable ribonucleoside CLIP (PAR-CLIP) relies upon metabolic RNA labeling to incorporate nucleoside analogs that cross-link at 365 nm, resulting in increased yield and bypassing some of the biases inherent in standard UV 254-nm cross-linking (4, 13). There is, however, evidence to suggest that nucleoside analogs may introduce other biases, such as the inhibition of ribosome biogenesis or the onset of a nucleolar stress response in several cell lines (14).
A comprehensive review of the pros and cons of technologies for measuring RBP–transcript interactions has recently been published (4). After the experimental phase, different algorithms and tools are applied to confirm the RBP–target interactions and to identify the RBP-binding motifs (Fig. 2). Commonly used statistical and machine learning algorithms are the focus of the remainder of this review.

**Statistical aspects of RNA target identification**

In RNA–protein interaction profiling experiments, a typical first question is to identify the transcripts bound by the protein of interest. This is usually done by comparing the immunoprecipitated RNAs with a control sample, typically the total RNA in the cell or RNA isolated from a mock-IP sample. This comparison is necessary because highly abundant nonbound RNAs may still appear as contaminants in the IP sample. This procedure is often referred to as determining the transcripts enriched in the IP fraction. Experiments of this type need to include multiple replicates to avoid false-positive identifications. Simply calculating the average ratio of the abundance of RNAs in the IP fractions over their abundance in the control samples is not a good idea, because of the count nature of the data: count ratios are unstable and may exaggerate the true ratio in the case of low counts. Moreover, in count data the variance scales with the mean abundance, and the more reliable samples (with the high counts), may therefore have a lower influence on the average ratio than the less reliable samples, which is undesirable. Hence, we suggest using a linear regression framework in this type of analyses (see next paragraph). The use of such a regression framework provides the additional advantage that it becomes easy to statistically compare the RNA-bound tran-
scripts across different experimental conditions, because these conditions can be included as an extra variable in the regression framework.

**Linear regression approaches**

Classically, a linear regression is a formula that predicts a dependent variable (in this case, mRNA transcript abundance) as a function of one or more explanatory variables (15). In a RIP-seq experiment, one of the explanatory variables reflects the actual pulldown effect (IP versus control). Other experimental variables may reflect the different experimental conditions (e.g., treatments of the cells). Regression models may be intuitively understood as variations on the familiar algebraic formula for a linear equation, \( y = mx + b \) (where \( m \) is the slope and \( b \) is the intercept). To understand how a linear equation may be derived from a dataset where the effect of explanatory variables is unknown, it is helpful to imagine a line that minimizes the distance between all of the predicted data points and the observed data points. The software finds the formula for the line that best fits the data. We typically run one regression model for each transcript and apply a form of multiple testing to correct for the number of tests performed (16). The quantitative effect of an explanatory variable on the dependent variable is referred to as a coefficient. In case there is twice as much of a certain transcript in the IP fraction than in the controls, the coefficient for the pulldown effect would be two.

The statistical packages that are commonly used for the analysis of RNA-seq data, for example the R packages DESeq2, edgeR, and limma-voom (17–19), implement a form of regression analysis that is also suitable for RIP-seq experiments. Typical experimental design formulas passed to these packages are formulas where the transcript is modeled as a linear combination of factors, such as shown in Equation 1,

\[
\text{transcript}_{ij} \sim \beta_1 \text{pulldown}_{ij} + \beta_2 \text{treatment}_{i} + \text{error} \quad (\text{Eq. 1})
\]

where transcript\(_{ij}\) reflects the abundance of transcript \( i \) in sample \( j \). Pulldown, reflects the pulldown status of sample \( j \) (1 for pulldown or 0 zero for control). Treatment, reflects the treatment status of sample \( j \) (e.g. 1 for treated and 0 for untreated). \( \beta_1 \) and \( \beta_2 \) refer to the corresponding coefficients for the pulldown and treatment and reflect how strong the pulldown and treatment effects are on the transcript abundance. Error (e) refers to irreducible random error.

In a standard linear regression, a number of assumptions are made: data should be (approximately) normally distributed, and the variance should not depend on the condition. These assumptions do not hold for RIP-seq data, because transcript abundance is not normally distributed, and the variance is rarely constant across all conditions. To compensate, DESeq2 and edgeR employ generalized linear models (GLMs), which allow the predicted value to be modeled as a function of any type of exponential probability distribution (20). Specifically, both packages fit a log-linear GLM to a negative binomial distribution. Limma assumes a normal distribution of the data. This distribution is approximated by the voom transformation (part of the limma package), which is essentially a log transform after addition of a small value to avoid taking the logarithm of zero).

A significance test can be performed to calculate the likelihood that the observed change in transcript abundance is the result of one of the elements in the model; for example, there is more protein bound to a particular transcript in the treated than in the untreated condition. In edgeR, this is the result of a likelihood ratio test (LRT), which compares the coefficient determined by the GLM to an alternative model that excludes that coefficient (18). An LRT for the effect of the treatment may be understood as a test to see whether the formula “transcript ~ pulldown + treatment” better fits the data than the simpler model “transcript ~ pulldown.” DESeq2 provides the additional (default) option of performing a Wald test, which calculates the probability that the reported coefficient could be zero, based on the amount of variation between replicates (17, 21). Imagine that for a certain transcript, the average abundance across all biological replicates in treated cells is 1200, whereas the average abundance in control cells is 600. We are much more certain that the coefficient for treatment is really 2 if the standard error is 5 than we are if the standard error is 500. Because the null hypothesis in DESeq2 is that the log fold change between conditions is zero, the Wald stat as employed by DESeq2 can be more simply understood as the log fold change divided by the standard error. The LRT or Wald statistics are automatically \( \chi^2 \) distributed, which means that significance values can be calculated based on degrees of freedom and corrected for multiple testing via the same methods familiar to biologists from \( t \) tests.

To identify transcripts that are more or less strongly bound between different conditions (say cellular treatments), one should model a statistical interaction term (22) as shown in Equation 2,

\[
\text{transcript}_{ij} \sim \beta_1 \text{pulldown}_{ij} + \beta_2 \text{treatment}_{ij} + \beta_3 \text{pulldown}_{ij} \cdot \text{treatment}_{ij} + \text{error} \quad (\text{Eq. 2})
\]

The pulldown-treatment interaction term reflects whether the pool of protein-bound transcripts is larger or smaller in the treated versus the untreated condition.

The linear regression approaches discussed above are examples of statistical modeling, i.e., the mathematical formalization of relationships between experimental parameters as equations. After identification of a pool of protein-bound transcripts, it is worth examining whether we can make predictions on the characteristics of transcripts that determine whether they are likely to be bound by the RBP (under a given condition). For this, we consider algorithms that are capable of utilizing pattern recognition to identify the fundamental differences between categories of transcripts. These algorithms are generally referred to as machine learning algorithms and are discussed in the subsequent paragraphs.

**Introduction to machine learning**

The term “machine learning” refers to a broad array of computational algorithms that can be used to predict and classify
outcomes. The strength of machine learning algorithms lies in the ability to derive the relevant predictors when the predictors are not known in advance. In this sense, the algorithm "learns" the predictors from the data itself (23). The diversity of the myriad machine learning algorithms and their applications across many disciplines constitutes its own field of expertise. In this section, a brief overview of machine learning algorithms, as pertains to the prediction of RNA–protein interactions, is presented.

Machine learning can be divided into two basic categories: supervised, in which a validated dataset is used as input to train the algorithm, and unsupervised, in which the algorithm searches for patterns without any evaluation of accuracy derived from a validated reference (24). Examples of unsupervised learning methods are dimensionality reduction (principal component analysis, for biological replicates of RNA-seq experiments) and clustering (k-means or hierarchical, as depicted in heat maps) (25, 26). Examples of popular methods for supervised learning are classification algorithms like support vector machines, k nearest neighbors and random decision forests.

Many recent studies have successfully applied a machine learning approach to RNA biology. The proliferation of online bioinformatics databases allows the use of diverse physical and biochemical properties as input for the modeling of RNA–protein interactions (27). Although these databases and the algorithms to create them do not use RIP-seq data, they are useful for the interpretation of RIP-seq experiments (Fig. 2A). Many of the most popular algorithms used to identify binding motifs or characteristics are either based upon the plotting of various data points in multidimensional space or upon so-called decision trees. In the former method, a line may be drawn that maximally separates the data into classes from the training set. This is the fundamental basis of support vector machines (SVMs) (28). Alternatively, k-nearest neighbors attempt to cluster the maximum number (k) of members in a labeled group based on the distance between them (29). This may be Euclidean distance or, as is especially appropriate for nucleic acid sequences, a metric that identifies the number of positions that differ between two strings (Hamming distance) (30). In their simplest form, decision trees use a heuristic series of yes-no answers to determine the likeliest outcome. More complex versions may be applied to both discrete and continuous predicted outcomes (Classification and Regression Trees) (31). Poorly predicting decision trees can be aggregated to form an ensemble algorithm that performs better than its components (32). Sophisticated algorithms designed to reduce model bias and variance, such as gradient tree boosting (GTB) or random decision forests (RDFs), fall under the broader category of ensemble decision tree algorithms (33).

One modern tool, RPI-Pred, uses a combination of protein and RNA higher-order 3D structures together with primary sequence as input for a SVM model to predict noncoding RNA–protein interactions (34). This approach is made possible by the existence of validated training sets from the protein–RNA interaction database (PRIDB) and the nucleic acid database and existing software for RNA and protein structure analysis (35, 36). RPI-Pred has a 94% prediction accuracy when using experimentally determined higher-order structures and an 83% prediction accuracy when trained with in silico predicted structures. Comparably, PredRBR, utilizes a total of 63 sequence and structure site features as input for a gradient tree boosting (GTB) model for prediction of RNA–protein-binding residues (37). This input includes, but is not limited to, chemical properties (electrostatic charge, molecular mass, and potential hydrogen bonds), side chain environmental features, position-specific scoring matrices, evolutionary conservation scores, and secondary structures. PredRBR has an overall accuracy of 84%. Although no direct comparison has been made between the performance of RPI-Pred and PredRBR, it is worth noting that the authors of PredRBR report lower precision metrics, meaning that their algorithm is more likely to include false positives.

Similar methods have been used to predict the presence of viral and cellular internal ribosome entry sites (IRESs), a notoriously difficult problem hampered by the poor predictive value of sequence motifs, biochemical features, and structural elements, especially for cellular IRESs (38). IRESPred is a modern tool that utilizes an SVM model with a total of 35 metrics that include probabilistic interactions between small ribosomal subunits and the 5′-UTR (5′-UTR), in addition to standard sequence and structural features (39). This innovative approach takes into consideration the finding that the conformation of the IRES is an essential factor in the ability to compete with cap-dependent translation (40). IRESPred has an accuracy of ~76% in identifying human IRESs in an experimentally validated dataset, which represents a significant improvement over the reported accuracy of 21% by previous in silico predictions (41).

When it comes to modeling of RNA–protein interactions from CLIP experiments, Hidden Markov Models (HMM) are frequently used. An HMM calculates the probability of finding an observable event based upon unobservable or “hidden” states (42). In CLIP experiments, RNA is covalently linked to the RNA-binding protein, which is isolated by immunoprecipitation and partially digested to reveal a specific sequence at the site of RNA–protein interaction (43). The probability of finding a series of nucleotides at the cross-linked site can be used as a training set in an HMM, where the nucleotide sequence is the observed state and the unknown state is “bound” or “not bound.” The individual nucleotides do not contribute independently to the probability of a transcript being bound or not. Rather, the probability of being in the unbound or bound state of adjacent nucleotides are dependent. The dependences are captured in the HMM model and used to assess the likelihood of an RNA–protein interaction. The distribution of nucleotides within the CLIP fragments can subsequently be used to not only predict novel targets but also to identify positional elements that facilitate the RNA–protein interaction (44). It is worth noting that although HMMS have been most often applied to nucleotide sequences, they have also been used to predict structural elements (45), and even combined sequence-structure motifs from CLIP-seq data (46).
Examples of machine learning algorithms for RNA–protein interactions

One example of an approach that combines machine learning with statistical inference to identify properties that predict binding of a particular protein can be seen in a study by Han et al. (44). In their study, the authors utilize a Hidden Markov Model trained on CLIP-seq data to score potential binding sites of polyuridylic tract–binding protein 1 (PTBP1), a regulator of mRNA splicing (44). The scored sites from the Markov model were combined with known PTBP1-regulated exons to develop a multinomial logistic regression model (similar to linear regression, except that the dependent variable is categorical (47)) to identify previously unrecognized PTBP1-regulated exons throughout the mouse transcriptome, as validated by RT-PCR and RNA-seq following PTBP1 depletion. This regression model treated PTBP1-enhanced and PTBP1-inhibited exons as separate classes and could distinguish between them, although the prediction was better for exons in the repressed category. This may be because structural elements, which were not considered in either the HMM scoring or the logistic regression, contribute more heavily to PTBP1 facilitated enhancement of exon expression. Of particular interest is that, aside from the obvious preference for pyrimidines, nucleotide triplets that contained guanines were more likely to increase binding probability, whereas triplets containing adenosine negatively affected PTBP1 binding. Furthermore, exons repressed by PTBP1 were much more likely to have high-affinity sites upstream than exons that display PTBP1-mediated enhancement. Given that the authors deliberately limited the model to a single factor (nucleotide sequence), it is exciting to speculate how the same approach could perform when provided with a more extensive selection of parameters. In ssHMM (46), both sequence and structural elements are considered as HMM parameters. The resulting Markov-chain style graph visualizes not only the probability of finding a given nucleotide at a position relative to the binding site, but also the likelihood of finding a structural emission state (hairpin, multiloop, etc.) in a surprisingly intuitive manner. Although ssHMM does not itself attempt to identify new potential targets outside of the CLIP-seq dataset used for training, there is no reason it could not be combined with a logistic regression model to search for new targets. The developers of PredRBR report that structural features were significantly better predictors of RNA–protein interactions than sequence or site features, indicating the importance of including structural input in the algorithm (37).

Limitations of machine learning approaches: The problem of overfitting

The success of SVM and GTB-based approaches on predicting interactions between transcripts and proteins is largely dependent on the availability of large training datasets from multiple experiments (i.e. number of bound versus unbound samples). When working with smaller datasets (say 100–200 bound transcripts) from a single experiment, the results may provide an overly rosy picture. Limited size and biases in the training data set may lead to “overfitting,” essentially meaning that the algorithm “learned the wrong lessons.” “Overfitting” (sometimes referred to as “overtraining”) is an issue in both predictive statistics and machine learning. The nature of overfitting and various strategies that can be employed to prevent it will be discussed next.

In statistical inference and machine learning, it is commonly observed that models perform well on the training data, but perform poorly on data not previously seen by the algorithm (33). There are two kinds of error that may cause a model to perform poorly. Underfitting occurs when the model lacks the correct parameters to represent the fundamental structure of the data (47). One may consider an attempt to draw a linear classifier through data that is not linearly separated (Fig. 3) or, in a biological example, attempting to predict RNA–protein interactions on the basis of sequence alone, without considering higher-order structures. As structural elements are generally better predictors of RNA–protein interactions than sequence motifs alone, a model that considers only the sequence would likely be subject to high bias (48, 49).

By contrast, overfitting occurs when the model includes an excess of parameters, and therefore incorrectly interprets a degree of random noise as signal (47). A classification line that carefully winds its way around every data point, as seen in Fig. 3, will succeed at “memorizing” the training data, but it has not inferred a true trend that will apply to new datasets in the future. The incorporation of too much noise in the dataset means that the model is subject to high variance. In other words, an underfit model with high bias and low variance would consistently make incorrect predictions, whereas an overfit model with low bias and high variance would make correct predictions very inconsistently. The challenge of designing a model that minimizes both the bias and the variance is sometimes referred to as the bias-variance tradeoff.

Overfitting is more common than underfitting in both statistical inference and machine learning and is generally harder to address (21, 50). There are, however, a number of methods that can be employed to minimize overfitting. As mentioned previously, classification algorithms are typically built by dividing known classified data into subsets. The algorithm first attempts to recognize patterns on the training subset, after which the fidelity of its predictions is tested on a validation subset. The process of iteratively generating partitioned sets of training and validation data to reduce sampling bias is referred to as cross-validation (47). In the most common type of cross-validation, the data are partitioned into subsets of equal size, with each subset used as validation data no more than once (51). This is
superior to iterative random sampling because it ensures that all data points are equally represented among the training and validation datasets. The authors of RPI-Pred, PredRBR, and IRESpred all utilize 10-fold cross-validation to assess the performance of their respective models (34, 37, 39). Even better is to train and cross-validate the model on a subset and to use the rest of the data for the final evaluation of its performance, a process often referred to as “external cross-validation.”

Another approach to reducing overfitting in linear or logistic regression models with a large number of features is Lasso (least absolute shrinkage and selection operator) (52). Lasso serves two functions: it both reduces the magnitude of the coefficient as model complexity increases (a process called “regularization”), and it discards explanatory variables when they have a poor predictive value. Models with a large number of variables are prone to a number of problems. First, coefficient size tends to scale with model complexity. Large coefficients are not as useful because too much of the predicted outcome depends upon changes in that variable, causing wild swings in the predicted outcome based on minor changes in the input. As a result, penalizing large coefficients reduces variance due to overfitting. Lasso does this by adding a penalty equivalent to the absolute value of the magnitude of coefficients. Second, models with many variables are likely to have variables that are mathematically related to one another, a problem known as multicollinearity. If you were to model both %GC content and %AT content as part of a linear regression, %AT content would always increase as %GC content decreased, and vice versa. This confounds the outcome, as the explanatory variables are now responding to each other as well as impacting the outcome. In many cases, the relationship between coefficients is not so obvious. Although it does not address multicollinearity directly, Lasso does have the advantage of performing variable selection, which it does by imposing a maximum limit on the absolute value of coefficients (53). This essentially forces coefficients with a poor predictive power to be set to zero, dropping them from the model.

Perhaps the most important way to avoid overfitting is to apply “common sense” limitations, i.e. to include only those explanatory variables that can be reasonably expected to have a strong predictive effect. In general, the models that perform best will be those that adhere to Occam’s razor: they contain the fewest number of assumptions necessary to explain the results. As Albert Einstein once said: “Everything should be made as simple as possible, but not simpler”. According to this fundamental scientific principle, the most intelligent and complex algorithm will never replace proper hypothesis formulation in obtaining a comprehensive understanding of the underlying biological theory. Barring the advent of true artificial intelligence, the human biologist will always be necessary to guide the machine.

**Outlook: Deep learning approaches for RNA–protein interactions**

In recent years, advanced algorithms based upon artificial neural networks (ANNs) are becoming increasingly popular in studying RNA–protein interactions at a transcriptome-wide level (54). As the name suggests, ANNs were inspired by the biological function of neurons as they operate in image processing (55). In the context of ANNs, a “neuron” takes multiple data inputs (analogous to neurotransmitters within a synapse) and applies a weight to each signal to provide information for the so-called activation function (56). Depending on the application, the output may be either binary, as intuitively associated with an action potential, or continuous, as seen in research questions more traditionally associated with linear regression. A neural network can be constructed by grouping many such computational neurons in layers, so the output from one neuron may be used as the input in the next layer. The layers between the input unit and output unit are referred to as “hidden,” as the values within are not observed within the input data.

Deep neural networks (DNNs) can be understood as an ANN consisting of several nonlinear layers, i.e. the activation function is nonlinear, and the neural network may contain loops or cycles between layers (55, 57). The deep learning architecture seeks to cyclically optimize the weight parameters in each layer. In a given cycle, the input is processed into the output layer, at which point a loss-of-function algorithm will compute the difference between the calculated outcomes and the labeled data (57). This information can be used in a process known as back propagation, in which a gradient to recalculate the weights is produced (58). DNNs typically include a regularization element to reduce overfitting (55, 57). For instance, the popularly used “dropout” method will randomly remove hidden neurons from the network (59). This creates a mesh of possible subnetworks that are forced to independently evolve, increasing the generalizability of the DNN (55, 57, 59).

For CLIP-type experiments, modern tools include DeepNet-RBP and iDeepS (54, 60). DeepNet-RBP successfully integrates the base RNA sequence alongside both secondary and tertiary transcript structures as input for the DNN. It performs especially well when predicting polypyrimidine tract–binding sites for IRES segments. This is an exciting prospect for researchers interested in IRES trans-acting factors such as Csde1/Unr, many of which contain IRESs. Indeed, DeepNet-RBP performed better in predicting interactions between polypyrimidine tracts and the Csde1 IRES than alternative methods that lack tertiary structure integration. By contrast, iDeepS combines multiple deep learning approaches to predict both binding sequence and structural motifs associated with RBP-binding sites from RNA transcripts (60). The iDeepS approach is a hybrid of convolutional neural networks (CNNs), a DNN variant inspired by the animal visual cortex, and long short-term memory (LSTM) networks, a type of DNN developed to model time series of unknown intervals. The addition of the LSTM layer yields better performance than variants that rely exclusively on CNNs (60). However, the authors of iDeepS state that it is currently limited to predicting binding targets for the RBPs present within the training data, and it does not incorporate tertiary structural information. Nevertheless, the overall approach shows promise.

In summary, statistical inference and machine learning are valuable tools in the analysis of transcriptome-wide RIP data. Based on the characteristics of the RNA-binding protein and the exact biological questions at hand, a specific experimental
design and an appropriate set of analysis tools need to be selected. An example of such an experiment and the considerations for analysis are given in Fig. 2B. At the same time, all analysis methods are subject to biases, overfitting, and false discoveries. General conclusions can usually be drawn from these analyses, but specific RNA–protein interactions identified by such methodologies require experimental validation.

References

1. Nishioka, S., Neelamrajju, Y., and Janga, S. C. (2016) Dissecting the expression relationships between RNA-binding proteins and their cognate targets in eukaryotic post-transcriptional regulatory networks. Sci. Rep. 6, 25711 CrossRef Medline
2. Ray, D., Kazan, H., Chan, E. T., Peta Castillo, L. P., Chaudhry, S., Talukder, S., Blencowe, B. J., Morris, Q., and Hughes, T. R. (2009) Rapid and systematic analysis of the RNA recognition specificities of RNA-binding proteins. Nat. Biotechnol. 27, 667–670 CrossRef Medline
3. Lambert, N., Robertson, A., Jangi, M., McGeary, S., Sharp, P. A., and Baker, C. B. (2014) RNA Bind-n-Seq: quantitative assessment of the sequence and structural binding specificity of RNA binding proteins. Mol. Cell 54, 875–890 CrossRef Medline
4. Wheeler, E. C., Van Nostrand, E. L., and Keene, J. D. (2018) Advances and challenges in the detection of transcriptome-wide protein–RNA interactions. Wiley Interdiscip. Rev. RNA. 2018, 9, CrossRef Medline
5. de Boer, E., Rodriguez, P., Bonte, E., Krijgsveld, J., Katsantoni, E., Heck, A., Grosveld, F., and Strouboulis, J. (2003) Efficient biotinylation and single-step purification of tagged transcription factors in mammalian cells and transgenic mice. Proc. Natl. Acad. Sci. U.S.A. 100, 7480–7485 CrossRef Medline
6. He, A., and Pu, W. T. (2010) Genome-wide location analysis by pull down of in vivo biotinylated transcription factors. Curr. Protoc. Mol. Biol. Chapter 21, Unit 21.20 CrossRef Medline
7. Friedersdorf, M. B., and Keene, J. D. (2014) Advancing the functional utility of PAR-CLIP by quantifying background binding to mRNAs and IncRNAs. Genome Biol. 15, R2 CrossRef Medline
8. Nicholson, C. Or., Friedersdorf, M. B., and Keene, J. D. (2017) Quantifying RNA binding sites transcriptome-wide using DO-RIP-seq. RNA 23, 32–46 CrossRef Medline
9. Ule, J., Jensen, K. B., Ruggiu, M., Mele, A., Ule, A., and Darnell, R. B. (2003) CLIP identifies Nova-regulated RNA networks in the brain. Science 302, 1212–1215 CrossRef Medline
10. Moore, M. J., Zhang, C., Gantman, E. C., Mele, A., Darnell, J. C., and Darnell, R. B. (2014) Mapping argonaute and conventional RNA-binding protein interactions with RNA at single-nucleotide resolution using HTS-CLIP and CIMS analysis. Nat. Protoc. 9, 263–293 CrossRef Medline
11. Sugimoto, Y., König, J., Hussain, S., Zupan, B., Curk, T., Frye, M., and Ule, J. (2012) Analysis of CLIP and iCLIP methods for nucleotide-resolution studies of protein–RNA interactions. Genome Biol. 13, R67 CrossRef Medline
12. König, J., Zarnack, K., Rot, G., Curk, T., Kaylic, M., Zupan, B., Turner, D. J., Luscombe, N. M., and Ule, J. (2010) iCLIP reveals the function of hnRNP particles in splicing at individual nucleotide resolution. Nat. Struct. Mol. Biol. 17, 909–915 CrossRef Medline
13. Hafner, M., Landthaler, M., Burger, L., Khorkshid, M., Haussler, J., Berninger, P., Rothbacher, A., Ascano, M., Jr., Jungkamp, A.-C., Munschauer, M., Ulrich, A., Wardle, G. S., Dewell, S., Zavolan, M., and Tuschl, T. (2010) Transcriptome-wide identification of RNA-binding protein and microRNA target sites by PAR-CLIP. Cell 141, 129–141 CrossRef Medline
14. Burger, K., Mühl, B., Kellner, M., Rohrmoser, M., Gruber-Eber, A., Windhager, L., Friedel, C. C., Dölen, L., and Eick, D. (2013) 4-Thiouridine inhibits rRNA synthesis and causes a nuclear stress response. RNA Biol. 10, 1623–1630 CrossRef Medline
15. Freedman, D. (2009) Statistical Models: Theory and Practice, Cambridge University Press, Cambridge, UK
16. Noble, W. S. (2009) How does multiple testing correction work? Nat. Biotechnol. 27, 1135–1137 CrossRef Medline
17. Love, M. I., Huber, W., and Anders, S. (2014) Moderated estimation of fold change and dispersion for RNA-Seq data with DESeq2. Genome Biol. 15, 550 CrossRef Medline
18. McCarthy, D. J., Chen, Y., and Smyth, G. K. (2012) Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. Nucleic Acids Res. 40, 4288–4297 CrossRef Medline
19. Ritchie, M. E., Phipson, B., Wu, D., Hu, Y., Law, C. W., Shi, W., and Smyth, G. K. (2015) Limma powers differential expression analyses for RNA-sequencing and microarray studies. Nucleic Acids Res. 43, e47 CrossRef Medline
20. Nelder, J. A., and Wedderburn, R. W. M. (1972) Generalized linear models. J. R. Stat. Soc. Ser. A. 135, 370 CrossRef
21. Harrell, F. E. (2001) Regression Modeling Strategies: With Applications to Linear Models, Logistic Regression, and Survival Analysis, pp. 94 and 184, Springer, New York
22. Johnson, C. P., Kim, I. K., Esmaeili, B., Amin-Mansour, A., Treacy, D. J., Carter, S. L., Hodis, E., Wagle, N., Seepo, S., Yu, X., Lane, A. M., Gragoudas, E. S., Vazquez, F., Nickerson, E., Cibulskis, K., et al. (2017) Systematic genomic and translational efficiency studies of uveal melanoma. PLoS ONE 12, e0178189 CrossRef Medline
23. Mitchell, T. M. (1997) In Machine Learning (Munson, E., ed) McGraw-Hill Book Co., New York
24. Russell, S. J. (2010) In Artificial Intelligence: A Modern Approach (Hirsch, M., editor in chief) pp. 694–695, Prentice Hall, Upper Saddle River, NJ
25. Neyman, J. (1967) In Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability (Neyman, J., ed) pp. 281–297, University of California Press, Berkeley, CA
26. Abdi, H., and Williams, L. J. (2010) Principal component analysis. Wiley Interdiscip. Rev. Comput. Stat. 2, 433–459 CrossRef
27. Si, I., Cui, J., Cheng, J., and Wu, R. (2015) Computational prediction of RNA-binding proteins and binding sites. Int. J. Mol. Sci. 16, 26303–26317 CrossRef Medline
28. Cortes, C., and Vapnik, V. (1995) Support-vector networks. Machine Learning 20, 273–297 CrossRef
29. Altman, N. S. (1992) An introduction to kernel and nearest-neighbor non-parametric regression. Am. Stat. 46, 175–185 CrossRef
30. Robinson, D. J. S. (2003) An Introduction to Abstract Algebra, Walter de Gruyter, Berlin (online) https://books.google.nl/books/about/An_Introduction_to_Abstract_Algebra.html?id=F92KUcWXWoC&redir_esc=y (Accessed March 9, 2018)
31. Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1993) Classification and Regression Trees, pp. 18 and 216, Chapman & Hall, New York
32. Rokach, L. (2010) Ensemble-based classifiers. Artif. Intell. Rev. 33, 1–39 CrossRef
33. Hastie, T., Tibshirani, R., and Friedman, J. H. (2009) The Elements of Statistical Learning: Data Mining, Inference, and Prediction, p. 605, Springer-Verlag, New York
34. Suresh, V., Liu, L., Adjeroh, D., and Zhou, X. (2015) RPI-Pred: predicting ncRNA–protein interaction using sequence and structural information. Nucleic Acids Res. 43, 1370–1379 CrossRef Medline
35. Lewis, B. A., Walian, R. R., Terrinbini, M., Ferguson, J., Zheng, C., Honavar, V., and Dobbs, D. (2011) PRIDB: a protein–RNA interface database. Nucleic Acids Res. 39, D277–D282 CrossRef Medline
36. Combiatore Narayan, B., Westbrook, J., Ghosh, S., Petrov, A. L., Sweeney, B., Zirbel, C. L., Leonis, N. B., and Berman, H. M. (2014) The Nucleic Acid Database: new features and capabilities. Nucleic Acids Res. 42, D114–D122 CrossRef Medline
37. Tang, Y., Liu, D., Wang, Z., Wen, T., and Deng, L. (2017) A boosting approach for prediction of protein–RNA binding residues. BMC Bioinformatics 18, 465 CrossRef Medline
38. Baird, S. D., Turcotte, M., Korneluk, R. G., and Holcik, M. (2006) Searching for IRES. RNA 12, 1755–1785 CrossRef Medline
39. Kolekar, P., Pataskar, A., Kulkarni-Kale, U., Pal, J., and Kulkarni, A. (2016) IRESpred: web server for prediction of cellular and viral internal ribosome entry site (IRES). Sci. Rep. 6, 27436 CrossRef Medline
40. Thompson, S. R. (2012) Tricks an IRES uses to ensnare ribosomes. Trends Microbiol. 20, 558–566 CrossRef Medline
41. Hong, J.-J., Wu, T.-Y., Chang, T.-Y., and Chen, C.-Y. (2013) Viral IRES prediction system—a web server for prediction of the IRES secondary structure in silico. PLoS ONE 8, e79288 CrossRef Medline

42. Yoon, B.-J. (2009) Hidden Markov Models and their applications in biological sequence analysis. Curr. Genomics 10, 402–415 CrossRef Medline

43. Milek, M., Wyler, E., and Landthaler, M. (2012) Transcriptome-wide analysis of protein–RNA interactions using high-throughput sequencing. Semin. Cell Dev. Biol. 23, 206–212 CrossRef Medline

44. Han, A., Stoilov, P., Linares, A. J., Zhou, Y., Fu, X.-D., and Black, D. L. (2014) De novo prediction of PTBP1 binding and splicing targets reveals unexpected features of its RNA recognition and function. PLoS Comput. Biol. 10, e1003442 CrossRef Medline

45. Neuwald, A. F., and Poleksic, A. (2000) PSI-BLAST searches using hidden Markov models of structural repeats: prediction of an unusual sliding DNA clamp and of β-propellers in UV-damaged DNA-binding protein. Nucleic Acids Res. 28, 3570–3580 CrossRef Medline

46. Heller, D., Krestel, R., Ohler, U., Vingron, M., and Marsico, A. (2017) ssHMM: extracting intuitive sequence-structure motifs from high-throughput RNA-binding protein data. Nucleic Acids Res. 45, 11004–11018 CrossRef Medline

47. Everitt, B., and Skrondal, A. (2010) The Cambridge Dictionary of Statistics, Cambridge, UK [online] http://www.cambridge.org/cl/academic/subjects/statistics-probability/statistics-and-probability-general-interest/cambridge-dictionary-statistics-4th-edition?format=HB&isbn=9780521766999#ZWh7cdA7DjqxSvF.97 (Accessed March 14, 2018)

48. Towfic, F., Caragea, C., Gemperline, D. C., Dobbs, D., and Honavar, V. (2010) Struct-NB: predicting protein–RNA binding sites using structural features. Int. J. Data Min. Bioinform. 4, 21–43 CrossRef Medline

49. Ren, H., and Shen, Y. (2015) RNA-binding residues prediction using structural features. BMC Bioinformatics 16, 249 CrossRef Medline

50. Hawkins, D. M. (2004) The problem of overfitting. J. Chem. Inf. Comput. Sci. 44, 1–12 CrossRef Medline

51. McLachlan, G. J., Do, K.-A., and Ambroise, C. (2004) in Analyzing Microarray Gene Expression Data (Balding, D. J., Cressie, N. A. C., Fisher, N. I., Johnstone, I. M., Kadane, J. B., Molenberghs, G., Ryan, L. M., Scott, D. W., Smith, A. F. M., and Teugels, J. L., eds, and Burnett, V., Hunter, J. S., and Kundell, D. G., eds emeriti) pp. 213–214, Wiley Series in Probability and Statistics, John Wiley & Sons, Inc., Hoboken, NJ

52. Tibshirani, R. (1996) Regression shrinkage and selection via the Lasso. J. R. Stat. Soc. Ser. B. 58, 267–288

53. Tibshirani, R. (1997) The lasso method for variable selection in the cox model. Stat. Med. 16, 385–395 CrossRef Medline

54. Zhang, S., Zhou, J., Hu, H., Gong, H., Chen, L., Cheng, C., and Zeng, J. (2016) A deep learning framework for modeling structural features of RNA-binding protein targets. Nucleic Acids Res. 44, e32 CrossRef Medline

55. Cao, C., Liu, F., Tan, H., Song, D., Shu, W., Li, W., Zhou, Y., Bo, X., and Xie, Z. (2018) Deep Learning and Its Applications in Biomedicine. Genomics Proteomics Bioinformatics 16, 17–32 CrossRef Medline

56. Russell, S. J. (2010) in Artificial Intelligence: A Modern Approach (Hirsch, M., editor in chief) pp. 727–728, Prentice Hall, Upper Saddle River, NJ

57. Min, S., Lee, B., and Yoon, S. (2017) Deep learning in bioinformatics. Brief Bioinform. 18, 851–869 CrossRef Medline

58. Hecht-Nielsen, R. (1989) International 1989 Joint Conference on Neural Networks, Washington, D. C., Vol. 1, pp. 593–605 CrossRef

59. Baldi, P., and Sadowski, P. J. (2013) Neural Information Processing Systems 2013, Dec. 5–10, 2013, Lake Tahoe, NV, NIPS, La Jolla, CA

60. Pan, X., Rijnbeek, P., Yan, J., and Shen, H.-B. (2018) Prediction of RNA-protein sequence and structure binding preferences using deep convolutional and recurrent neural networks. BMC Genomics 19, 0511 CrossRef Medline

61. Moore, K. S., Yagci, N., van Alphen, F., Paolini, N. A., Horos, R., Held, N. M., Houtkooper, R. H., van den Akker, E., Meijer, A. B., ‘t Hoen, P. A. C., and von Lindern, M. (2018) Cidea binds transcripts involved in protein homeostasis and controls their expression in an erythroid cell line. Sci. Rep. 8, 2628 CrossRef Medline

62. Raschka, S. (2015) Python Machine Learning, 1st Ed., Packt Publishing, Birmingham, UK