CHAPTER ELEVEN

Protein Engineering and Stabilization from Sequence Statistics: Variation and Covariation Analysis

Venuka Durani*, Thomas J. Magliery*,†,1

*Department of Chemistry, The Ohio State University, Columbus, Ohio, USA
†Department of Biochemistry, The Ohio State University, Columbus, Ohio, USA
1Corresponding author: e-mail address: magliery.1@osu.edu

Contents

1. Introduction 238
2. Case Study: BPTI 239
3. Acquiring an MSA 240
   3.1 Protocol for acquiring an MSA from Pfam 241
   3.2 Building an MSA using the MyHits Web site 243
4. Relative Entropies: Quantifying the Degree of Positional Variation 244
   4.1 Protocol for calculating RE values 245
   4.2 Relative entropy in BPTI 245
5. Mutual Information: Quantifying the Degree of Covariation 246
   5.1 Protocol for calculating mutual information values 247
   5.2 Mutual information in BPTI 250
6. Protocol for Predicting Stabilizing Mutations 253
7. Summary 253
References 254

Abstract

The concepts of consensus and correlation in multiple sequence alignments (MSAs) have been used in the past to understand and engineer proteins. However, there are multiple ways of acquiring MSA databases and also numerous mathematical metrics that can be applied to calculate each of the parameters. This chapter describes an overall methodology that we have chosen to employ for acquiring and statistically analyzing MSAs. We have provided a step-by-step protocol for calculating relative entropy and mutual information metrics and describe how they can be used to predict mutations that have a high probability of stabilizing a protein. This protocol allows for flexibility for modification of formulae and parameters without using anything more complicated than Microsoft Excel.
We have also demonstrated various aspects of data analysis by carrying out a sample analysis on the BPTI-Kunitz family of proteins and identified mutations that would be predicted to stabilize this protein based on consensus and correlation values.

1. INTRODUCTION

The information required to fold a protein into its native structure is encoded in its amino acid sequence (Anfinsen, 1973). Studying commonalities and differences between several sequences that fold into the same structure can provide information about the necessary and sufficient parameters of this code. Vast databases of sequences (Consortium, 2011; Finn et al., 2010) classified into structural folds based on sequence similarity (Chothia, 1992; Chothia & Lesk, 1986; Orengo, Jones, & Thornton, 1994) have become available with the advent of the genomic era. These databases have multiple sequence alignments (MSAs) of proteins with nearly the same structure and function and can be exploited to gain information about their folds. Upon comparison of sequences within an MSA, it is often observed that some amino acids occur again and again at a particular position. These might be important for the fitness of the protein in a structural, functional, or dynamic way and hence did not tolerate much mutation over the course of evolution. This concept of consensus has been used to design peptide motifs (Tripp & Barrick, 2003) such as tetratricopeptide repeats (TPRs) (Main, Xiong, Cocco, D’Andrea, & Regan, 2003) and ankyrin repeats (Binz, Stumpp, Forrer, Amstutz, & Pluckthun, 2003; Mosavi, Minor, & Peng, 2002) and even larger globular proteins such as fungal phytases (Lehmann et al., 2000). Recently, we demonstrated the consensus design of a triosephosphate isomerase with native-like activity and very high stability (Sullivan, Durani, & Magliery, 2011). Various methods have been used by researchers to estimate the degree of conservation of positions in an MSA. Some methods take into account the physical properties of amino acids such as hydrophobicity and charge (Mosavi et al., 2002), some others use global propensities (Main et al., 2003), some use statistical free energy-based metrics (Steipe, Schiller, Pluckthun, & Steinbacher, 1994), and yet others use information theory-based metrics (Magliery & Regan, 2005). Previous studies on antibodies, phytases, and thioredoxin suggest that consensus mutations—where a residue is mutated to the amino acid that is the most common at that position in the MSA—stabilize proteins about half the time (Godoy-Ruiz, Perez-Jimenez, Ibarra-Molero, & Sanchez-Ruiz, 2005; Knappik et al., 2000; Steipe et al., 1994).
Consensus design implicitly assumes that all amino acid positions function independently, but in reality, the amino acids in a protein interact with each other and work cooperatively to produce the optimum structure required for its function. These interactions of the amino acids can be studied using correlation analysis of MSA databases. Correlations can be calculated using various algorithms (Fodor & Aldrich, 2004) that employ approaches such as amino acid similarity matrices, $\chi^2$-tests (Kass & Horovitz, 2002), perturbation-based approaches (Dekker, Fodor, Aldrich, & Yellen, 2004; Halabi, Rivoire, Leibler, & Ranganathan, 2009; Lockless & Ranganathan, 1999), and information theory (Ackerman & Gatti, 2011). Studies on WW domains have shown that capturing a structural fold is more successful when both consensus and correlation are used, as opposed to using consensus alone (Russ, Lowery, Mishra, Yaffe, & Ranganathan, 2005; Socolich et al., 2005). Studies on TPR motifs have shown that complex charge networks on the surface of proteins can skew results obtained by consensus, but correlation analysis can capture them, leading to better results in protein engineering (Magliery & Regan, 2004). The concept of correlation has also been used to predict structural contacts (Bartlett & Taylor, 2008; Miller & Eisenberg, 2008; Sullivan et al., 2012), probe allosteric communications (Kass & Horovitz, 2002; Suel, Lockless, Wall, & Ranganathan, 2003), and to identify sectors of interacting residues in proteins (Halabi et al., 2009; Lockless & Ranganathan, 1999). We wished to use consensus and correlation approaches to study proteins, particularly for protein engineering (Magliery, Lavinder, & Sullivan, 2011). We have also shown recently that a combination of consensus and correlation data can be used to predict stabilizing mutations to a protein with higher accuracy than using consensus information alone (Sullivan et al., 2012). However, there are multiple ways of acquiring MSA databases and also numerous mathematical metrics that can be applied to calculate each of the parameters. This chapter describes the overall methodology that we have employed to acquire and statistically analyze MSAs.

## 2. CASE STUDY: BPTI

Throughout this chapter, the workings of each method will be illustrated using examples from the bovine pancreatic trypsin inhibitor (BPTI) MSA. BPTI is a Kunitz domain, the active domains of proteins that inhibit the function of proteases and are called Kunitz-type protease inhibitors (Chen et al., 2001; Paesen et al., 2009; Schmidt, Chand, Cascio, Kisiel, & Bajaj, 2005). They are about 50–60 amino acids long with a molecular
weight of about 6 kDa and fold into a disulfide-rich $\alpha/\beta$ structure (Fig. 11.1; Parkin, Rupp, & Hope, 1996). Standalone Kunitz domains have been used as a framework for the development of new pharmaceutical drugs. BPTI is an extensively studied model protein belonging to this family.

3. ACQUIRING AN MSA

This section discusses how MSAs can be acquired or built and curated based on the availability of information and the goal of the project. We will discuss two ways of acquiring MSAs. One is by accessing data already organized as MSAs in Pfam and the other by building MSAs using the MyHits Web site. Pfam (Punta et al., 2012) (http://pfam.janelia.org/) is an online database of MSAs and hidden Markov models (HMMs) (Eddy, 2004) powered by the HMMER (Finn, Clements, & Eddy, 2011) search tool. Each Pfam MSA represents a protein family or domain. Pfam has two categories, and Pfam-A families have better annotation and alignment quality than Pfam-B. For the purposes of this chapter, Pfam-B will be ignored and Pfam will refer to Pfam-A. Each protein family listed in Pfam has a seed alignment and a full alignment (Sonnhammer, Eddy, & Durbin, 1997). The seed alignment is a high quality, manually checked alignment of a few hand-picked sequences that represent the protein family in question. It is used to build an HMM profile which is used for automated searches of more sequences that belong to the protein family. The positions that occur in seed alignments are called canonical positions, and the alignment of these positions is usually of good quality for the full alignment too. Insertions and deletions with respect to the seed alignment are noncanonical
positions, which tend to have low occupancies and poor alignment quality. Pfam has an option to download sequences in a format where canonical positions are represented with one-letter amino acid codes in capital letters (A, C, D, etc.) and gaps as hyphens (-), whereas the noncanonical positions are represented in small letters (a, c, d, etc.) with gaps as periods (.). This format allows for choosing only canonical positions for analysis if so required. To keep Pfam updated with new protein sequences that are discovered, the HMM-based search of available protein databases is carried out and released about once a year. While the full alignments change substantially between releases, the seed alignments remain almost invariant. If the quality of an alignment is not sufficient, it can be curated and realigned with more manual intervention using alignment tools such as ClustalX (Thompson, Gibson, & Higgins, 2002).

An ideal MSA for sequence statistics would have many unique sequences of equal length that align well with each other. However, typical MSA databases downloaded from Pfam are imperfect due to sequence repeats and truncated sequences. These imperfections must be accounted for, especially in correlation analysis. For example, an abundance of repeats may bias correlations between the pairs that occur in them, and small changes to consensus sequences from differently assembled databases can have large effects on protein properties (Magliery et al., 2011). To avoid such biases, the MSA used for correlation analysis can be curated by removing exact sequence repeats and short sequence fragments. This gives rise to a database that looks more like an ideal MSA (Fig. 11.2). Our lab tends to favor the use of Microsoft Excel for simple manipulations, sometimes with the addition of Visual Basic macros, and the use of Perl scripts for more complicated manipulations.

### 3.1. Protocol for acquiring an MSA from Pfam

**Step 1.** *Downloading sequences:* The “keyword search” tool on the home page of the Pfam Web site (http://pfam.janelia.org/) is used to search for the protein of interest. If the “description” for an entry matches the protein of interest, the entry for the protein is accessed by clicking on either the “accession” or the “ID.” The MSA is accessed by going to the “sequences” tab. The following formatting options are typically used for the sequences: Alignment: Full; Format: Selex; Order: Tree; Sequence: All upper case; Gaps: Gaps as “-” (dashes); Download/view: Download. The “Generate” option generates a text file with all sequences and their names.
Step 2. *Curating for length of sequences:* Sequences from the text file are imported into a Microsoft Excel file (office 2010) with the names in column A and sequences in column B. Column B is copied into column C. Using the “replace” function, all dashes (-) are removed from column C. The lengths of the sequences in column C are calculated in column D using the \( \text{LEN} \) function (\( D2 = \text{LEN}(C2) \) and so on). Column D is selected and the filter function is used to filter the data for sequences that have a reasonable length. For example, in the BPTI database, the average length of a sequence was 53.5 amino acids and about half of the sequences in the database were 53 amino acids long. We kept all sequences that were 50–60 amino acids long. These filtered data were copied into another worksheet.

Step 3. *Removing sequence repeats:* Column C, that has all the sequences, is selected. Then the “Data” tab is accessed to select the “Advanced” filter option. The option for “Unique records only” is selected. This removes all the sequence repeats. Columns A and B of the filtered data are copied to another worksheet.

Step 4. *Occupancy of positions:* The curated database is first tabulated using the \( \text{MID} \) function that returns a specified number of characters from a specified position in a text string. Column A has the sequence names, column B has all the aligned sequences, and row 1 has position numbers. Since the BPTI alignment containing 880 sequences had 184 aligned positions, numbers 1–184 were written in row 1. The formula \( C2 = \text{MID}($B2,C$1,1) \) was copied over the entire table \( C2:GD881 \). Then the occupancy for each column is calculated by

---

**Figure 11.2** Curation of multiple sequence alignments: Ideally, an MSA would have many unique sequences of equal length (A). However, the MSAs available in Pfam often have sequence fragments and repeated sequences (B). This can cause biases in correlation analysis, and hence, the MSA should be curated by removing sequence fragments and exact repeats so that it resembles an ideal MSA (C).
counting the number of gaps (-) and subtracting it from the total number of sequences \((C882 = 100 \times (880 - \text{COUNTIF(C2:C881, "}")/880))\). Columns with high occupancy are chosen. A quick way to filter sequences based on occupancy is to transpose the table so that each position is in a row and use the “filter” function on the column containing occupancy values. The numerical value used for the occupancy cutoff can be chosen such that the length of the final sequence is close to the length of a wild-type protein, typically eliminating sites with less than about 65% occupancy.

3.2. Building an MSA using the MyHits Web site

Sometimes, a project may require an MSA with different specifications than are applied in making the Pfam MSAs. As an example, we generated the BPTI49 MSA of standalone structures (as opposed to Kunitz domains in larger proteins).

Step 1. Seed alignment: First, a seed alignment is required which can be constructed by compiling known sequences of the protein of interest and aligning them using a program such as ClustalX. Since the Kunitz_BPTI family has a Pfam MSA, its seed alignment was downloaded. The 151 sequences in this alignment were crosschecked for domain organization in the UniProt database and any sequence shown as a part of a multidomain construct was discarded. At the end of this process, only 30 sequences remained. These sequences were aligned using ClustalX and saved in the FASTA format. This MSA was used as the seed alignment.

Step 2. Search based on HMM profile: The seed alignment is used to create an HMM using HMMER3 on the MyHits (Pagni et al., 2007) Web site (http://myhits.isb-sib.ch/). The HMMER3 option under the “search” section of “Tools” is selected. The seed MSA from step 1 is copied into the space provided. For constructing the BPTI database, the Swiss-Prot option was selected under the “seq_source” section and this “search” provided about 200 hits.

Step 3. Screening of search results: The search results from step 2 can be curated based on parameters such as sequence identity and length of sequences. For the BPTI database, we screened for sequences that were annotated as standalone structures. In order to do this, the “entry” link was used to access the UniProt data for each of the hits from step 2. Sequences that were annotated as standalone sequences were selected. “Send checked matches” option was set
to “MSA Hub” and the sequences were copied. The final alignment for the BPTI sequences selected using this process had 49 sequences.

4. RELATIVE ENTROPIES: QUANTIFYING THE DEGREE OF POSITIONAL VARIATION

We estimate the conservation of positions in the MSAs using the relative entropy (RE, sometimes symbolized as $D$) metric. This is an information theoretic metric that gives the “distance” of a distribution from a reference distribution. It is logarithmically related to the multinomial probability of observing a particular distribution if you expect the reference distribution (Magliery & Regan, 2005). If a position is highly conserved, its amino acid distribution would be very different from that expected at random, making its RE value very high relative to a neutral reference state (Fig. 11.3). The reference distribution chosen in this case was based on codon usages in the yeast proteome. This is a good estimate of the amino acid frequency expected in an average eukaryotic protein, accounting for factors such as codon usage and side chain chemistry. This reference state is conveniently invariant over time. RE ($D$) is calculated as

$$D = \sum_i p(x_i) \log \frac{p(x_i)}{q(x_i)}$$

where $p(x_i)$ is the probability of observing residue $i$ at position $x$ in the MSA and $q(x_i)$ is the probability of observing residue $i$ in the yeast proteome. We used fraction of sequences in each category as approximation of probability.

Figure 11.3 Amino acid distributions corresponding to various relative entropy values: Each color in the pie charts represents the frequency of occurrence of an amino acid residue. (A) Amino acid distribution expected at random in an average eukaryotic protein. (B) Amino acid distribution at a position with low a RE value of 0.2. (C) Amino acid distribution at a position with a high RE value of 2.6. (D) Amino acid distribution at a position with a very high RE value of 4.3.
4.1. Protocol for calculating RE values

The formulae for calculating consensus values need to be set up so that disallowed functions like division by zero or taking log of a negative number do not crop up in any calculation. Keeping this in mind, the following protocol was established.

**Step 1.** *MSA format:* The tabulated form of the curated MSA is used (Section 3.1, step 4), and the first column of the table (column C) is assigned the name pos1. This worksheet is named “seq.”

**Step 2.** *Counting amino acid frequencies:* On another worksheet named “RE,” a table is made to record the number of times various amino acids occur at various positions. Position numbers are recorded in column A and one-letter amino acid codes in row 1. Then the formula \( B2 = \text{COUNTIF(OFFSET(pos1,0,$A2-1),B$1)} \) is entered followed by Ctrl+Shift+Return. (This is how array formulae are entered in Excel.) This formula counts the number of occurrences of amino acids and is copied into the entire table. In order to calculate frequency of occurrence, each value is divided by the sum of the row it is contained in and the values are recorded in another table.

**Step 3.** *Calculating relative entropies:* The formula used to calculate relative entropies is \( AB4 = \text{IF}(B4=0,0,B4*\ln(B4/B$2)) \) copied over table \( AB4:AU56 \) (each row of the table is summed for RE corresponding to each position) where \( B4:U56 \) is the table of amino acid frequencies in the MSA, and \( B2:U2 \) has the amino acid frequencies in the yeast proteome (A, 0.056; C, 0.013; D, 0.058; E, 0.065; F, 0.045; G, 0.051; H, 0.021; I, 0.065; K, 0.073; L, 0.095; M, 0.021; N, 0.061; P, 0.044; Q, 0.039; R, 0.044; S, 0.089; T, 0.059; V, 0.057; W, 0.010; Y, 0.034).

4.2. Relative entropy in BPTI

The frequency of occurrence of all the amino acids, although similar, is not the same in the reference state (Fig. 11.3A). The RE values depend on both the frequency of an amino acid and its identity. While an average amino acid occurs in nature about 5% of the time, the Cys residue has only 1.3% frequency of occurrence in the reference set while the most common residue Leu occurs 9.5% of the time. When a rarer residue such as Cys dominates a position, then the RE becomes higher than when a more common amino acid such as Leu dominates the position. In the BPTI database, due to three
conserved disulfide bonds, six Cys residues were the most conserved. The extremely high RE values for these positions are in part because Cys residues are relatively less common in nature. The amino acid distributions for positions with varying RE scores are shown in Fig. 11.3. While the amino acid distribution at a position with a small RE value of 0.2 (Fig. 11.3B) looks very similar to that of the reference state, a position dominated by three residues Phe, Tyr, and Trp has an RE value of 2.6 (Fig. 11.3C) and a position dominated by Cys residue has an RE value of 4.3 (Fig. 11.3D).

Comparison of RE values for the two BPTI databases shows that while they agree with each other very well (Fig. 11.4A), the smaller database has slightly higher RE values. The average RE value for the smaller BPTI49 database is 1.7, while the average RE value for the significantly larger Pfam database is 1.3. The good agreement between data from both databases ($R^2 = 0.95$, Fig. 11.4A) is in part because of the dominant Cys residues that have very high RE values in both databases. At lower RE values, the databases are not as concordant.

5. MUTUAL INFORMATION: QUANTIFYING THE DEGREE OF COVARIATION

Positional correlations within the MSA can be calculated using the mutual information (MI) statistic from information theory (Applebaum, 1996; Shannon & Weaver, 1949). MI is the RE between the actual joint probability distribution for two sites, and the hypothetical independent joint
probability distribution determined from the product of the marginal distributions for each site. For instance, if Ala occurs at position \( x \) 50% of the time and Leu occurs at position \( y \) 50% of the time, then if both positions are independent of each other, the Ala–Leu pair would be observed in 25% of the sequences. If this pair is observed much more (or much less) than 25% of the time, then the pair of positions is correlated (or anticorrelated).

The mutual information \( \text{MI}_{xy} \) between sites \( x \) and \( y \) is calculated as,

\[
\text{MI}_{xy} = \sum_i \sum_j p(x_i, y_j) \log \frac{p(x_i, y_j)}{p(x_i)p(y_j)}
\]

where \( p(x_i) \) is the probability of observing residue \( i \) at position \( x \) in the MSA, \( p(y_j) \) is the probability of observing residue \( j \) at position \( y \), and \( p(x_i, y_j) \) is the joint probability of observing residue \( i \) at position \( x \) and residue \( j \) at position \( y \). For each pair of positions, this value is summed over all the 400 possible pairs of amino acid residues. MI is symmetric (the extent of information position \( x \) contains about position \( y \) is the same as the extent of information position \( y \) contains about position \( x \)), and there is no meaning to the MI of a position with itself. Consequently, we need to calculate only \( (a \times a) - a \) \( a \) values. For large MSAs, this can save a considerable amount of computational time. MI does not distinguish between correlations and anticorrelations, and it does not give details of correlations at the amino acid level, but it is easily calculated and visualized.

It is not simple to calculate a statistical significance for MI values. Instead, we calculate a threshold “noise level” from MI values for a scrambled dataset with the same amino acid distribution in each position. This takes into account any biases that may arise due to database size and correlation patterns. The maximum MI value over all pairs obtained from this scrambled data is set as the noise level (Fig. 11.5). For better statistical significance, this value can be calculated for multiple randomized datasets.

5.1. Protocol for calculating mutual information values

**Step 1.** *MSA and frequency table:* The MSA and amino acid frequency table can be copied from the RE calculation workbook (worksheets “seq” and “RE” from Section 4.1).

**Step 2.** *Reference distribution:* The reference distribution is calculated using the formula \( B28 = \$C3*VLOOKUP(B$27, \$B$3:$D$22, 3, FALSE) \) copied over all the cells in table B28:U47. Range C3:C22 has the amino acid frequencies for position 1, and range D3:D22 contains amino acid
frequencies for position 2. One-letter codes for amino acids are listed in B3:B22, B27:U27, and A28:A47.

**Step 3.** *Observed distribution:* The observed distribution is calculated using the formula \( B52 = \text{SUM(IF(OFFSET(pos1,0,$C$1-4)=A52,1,0)*IF(OFFSET (pos1,0,$D$1-4)=B51,1,0))/V$51} \) followed by Ctrl+Shift+Return and copied over all the cells of table B52:U71. Cell C1 contains the position number of one position, and D1 contains position number of the other position in question. Cell V51 contains the frequency of cooccurrence of amino acids in the two positions and is calculated using the formula \( V51 = \text{SUM((IF(OFFSET(pos1,0,$C$1-4)<>-",1,0)*IF(OFFSET (pos1,0,$D$1-4)<>-",1,0)))} \) followed by Ctrl+Shift+Return.

**Step 4.** *MI calculation:* MI is calculated using the formula \( B75 = IF (B52=0,0,B52*LN(B52/(B28))) \) copied over all the cells in table B75:U94 and then summed over the whole table. In this worksheet, range B52:U71 is the table of observed frequencies of cooccurrence (from step 3) and range B28:U47 is the table of expected frequencies of cooccurrence assuming mutual independence (from step 2).

**Step 5.** *Repeating the calculation for all pairs of positions:* While calculating correlation values, the same set of calculations need to be repeated for each pair of positions. Once the excel spreadsheet is set up to calculate one iteration of the calculation, the following macro is used to iteratively calculate these values and tabulate them. Since MI is symmetric, calculating values on one side of the diagonal is sufficient.
Sub MI()
,
‘This script calculates MI values for each pair and tabulates them
‘There are 53 positions in this alignment.
‘B96 is the cell that contains MI value calculated in each iteration
‘Row 100 and column A have position numbers
‘The MI values are tabulated in B101:BB153
,
Application.ScreenUpdating = False
For ColumnCounter = 2 To 54
    For RowCounter = 100 + ColumnCounter To 153
        Worksheets("MI").Activate
        Range(“C1”) = Cells(RowCounter, 1)
        Range(“D1”) = Cells(100, ColumnCounter)
        Calculate
        Cells(RowCounter, ColumnCounter) = Range("B96")
    Next RowCounter
Next ColumnCounter
Application.ScreenUpdating = True
ActiveWorkbook.Save
End Sub

Step 6. Converting a table into a list: When correlation values are calculated, the output format is in a matrix or table form. In order to sort the values, it is more convenient to format them into a list. This Excel macro converts a $53 \times 53$ MI table into a list. This macro takes the values from only half of the matrix (below the diagonal). The matrix is located in a worksheet titled “matrix” and starts from cell A1. The first row and first column contain position numbers. Another blank worksheet called “list” needs to be created before the macro is run. The RowCounter and ColumnCounter values can be edited if the table in question is of a different size.

Sub matrix_list()
,
‘This script converts a 53x53 table in “matrix” worksheet
‘into a list in “list” worksheet
‘In the “matrix” worksheet, Row 1 and column A have data labels
,
Application.ScreenUpdating = False
Dim ColumnCounter As Integer
Dim RowCounter As Integer
Dim MyCounter As Long
MyCounter = 2
For RowCounter = 2 To 54
    For ColumnCounter = (RowCounter + 1) To 54
        Worksheets("list").Cells(MyCounter, 1).Value = Worksheets("matrix").Cells(RowCounter, 1).Value
        Worksheets("list").Cells(MyCounter, 2).Value = Worksheets("matrix").Cells(1, ColumnCounter).Value
        Worksheets("list").Cells(MyCounter, 3).Value = Worksheets("matrix").Cells(RowCounter, ColumnCounter).Value
        MyCounter = MyCounter + 1
    Next ColumnCounter
Next RowCounter
Application.ScreenUpdating = True
ActiveWorkbook.Save
End Sub

Step 7. **Calculating noise level:** In order to calculate noise level for a correlation calculation, a randomized MSA is created where each column is scrambled. This keeps the consensus information the same while scrambling the correlations. In order to randomize the MSA, the \texttt{RAND}, \texttt{RANK}, and \texttt{INDEX} functions of Microsoft Excel are used. For the BPTI database, the worksheet containing the original MSA (table C2:BC881) was named “seq” and a table of the same size was created in another worksheet named “rand” where each cell of the table was $=\texttt{RAND()}$ and hence contained a random number between 0 and 1. In a third worksheet named “scramble,” another table of equal size was created where the cell C2 was $=\texttt{INDEX(seq!C2:C881,RANK(rand!C2,rand!C2:C881))}$. This formula was copied over the whole table C2:BC881. Every time the worksheet was refreshed/recalculated (F9 on manual calculation mode), a new column-randomized MSA was generated. MI calculation was carried out for the randomized dataset as per the procedure described in steps 1–5, and the maximum MI value obtained from the column-randomized MSA was accepted as the noise level.

5.2. **Mutual information in BPTI**

Correlation values were calculated for each pair of positions using both the BPTI49 database and the curated Pfam database. The correlation data show a
much stronger dependence on size of the database than the consensus data. As can be seen from the plot comparing correlation values between these datasets, the smaller dataset has significantly higher values for correlations (Fig. 11.4B). The disparity between ranking of pairs of positions in terms of their correlation values is also quite high between the databases ($R^2 = 0.64$, Fig. 11.4B). This shows that the database size has a greater effect on correlation values than on consensus values. For the larger Pfam database, the maximum MI value was 0.49 and that for the randomized dataset was 0.22 leaving 550 correlations above noise. On the other hand, for the smaller BPTI49 database, the maximum MI value was 1.4 and that for the randomized dataset was 1.3 leaving only nine correlations above noise. This shows another aspect of the finite-size effect and reflects the difficulty of achieving statistical significance for smaller MSAs. Previous studies have shown that to avoid “finite-size effects” it is preferable to use MSAs of at least a few hundred sequences for correlation analysis (Gloor, Martin, Wahl, & Dunn, 2005; Weil, Hoffgaard, & Hamacher, 2009). Based on these estimates, the BPTI49 database is too small for reliable correlation calculations. Henceforth, all the correlation calculations discussed were carried out on the curated Pfam database of BPTI.

Correlation calculations generate a lot of data and analyzing it can be challenging. A very visual and intuitive way of representing this information is via a heat map (Fig. 11.6). For most of our applications, we make heat maps by using conditional formatting options in Microsoft Excel to color-code the table of MI values. The MI values are binned into intervals based on the noise value, mean value, and standard deviation. In the heat map shown, all values below the noise level of 0.22 are colored blue, and values up the mean and one, two, or three, or more $\sigma$ above the mean are green, yellow, orange, and red, respectively.

To evaluate how the extent of conservation of positions affects their correlation distribution, the correlation distribution corresponding to positions with below average conservation scores can be compared to that of positions with higher than average conservation scores (Fig. 11.7). Positions with low conservation have a greater number of correlations than positions with high conservation. This is a consequence of the meaning of “correlation” as defined by MI. If a site $x$ is all Ala, then all $x,y$ pairs are Ala and something, and no additional information about $x$ can be given by the identity of $y$.

Although MI is a very useful metric for getting an overall picture of the level of correlation in the database, if we want to look at the details of the
correlations at the amino acid level then other methods including perturbation-based approaches such as statistical coupling analysis (Lockless & Ranganathan, 1999) or correlation coefficients (Wang et al., 2009) are useful. However, while these methods furnish more information at the amino acid level, they are computationally more demanding than MI and require more sequences for statistical significance.

Figure 11.6 A heat map representing the MI data of BPTI: The position numbers as per the MSA are specified on the left and top of the heat map. The color scale is shown on the right side.

Figure 11.7 Effect of residue conservation on MI distributions: (A) MI distribution for all positions in curated BPTI_Kunitz Pfam alignment. (B) MI distribution for positions with below average conservation. (C) MI distribution for positions with above average conservation. (D) Color scale for binning the data.
6. PROTOCOL FOR PREDICTING STABILIZING MUTATIONS

Making “consensus mutations” (i.e., mutating an amino acid to that observed most frequently at that position in an MSA of related proteins) stabilizes a protein about half the time. Recently, we demonstrated that avoiding positions with strong correlations improves the success of predicting stabilizing consensus mutations (Sullivan et al., 2012). This protocol for predicting stabilizing mutations has three steps.

**Step 1.** *Conservation filter:* All positions with above average RE values are chosen. Positions with low RE values do not contain much information at the level of consensus and are avoided. For the BPTI database, the average RE value was 1.3 and there were 18 positions that had RE values above that.

**Step 2.** *Correlation filter:* The positions are ranked based on the maximum MI value for any pair they are involved in and positions with correlations in the top 1% of all \((n^2 - n)/2\) values are discarded. Positions that have high correlation values may need compensatory mutations which are difficult to identify and hence they are avoided. Also, positions that are nearly invariant may have “hidden correlations” (i.e., may be physically coupled even though that cannot be mathematically distinguished from being independently required) and are also avoided. In the BPTI database, the positions with MI values above 0.36 and those with RE values above 3 were discarded thus leaving 11 positions.

**Step 3.** *Identifying mutation sites:* Several positions identified by applying the above filters may already be consensus residues in the protein of interest, so in the final step, we compare the consensus residue and wild-type residue at these sites; the sites where these residues are different are identified as mutation targets. If we wanted to make consensus mutations to the bovine version of the BPTI-Kunitz domain (PDB file 1BPI), then 7 out of the 11 sites would already have the consensus residue, leaving 4 sites as candidates for mutation. The mutations to BPTI suggested based on this protocol are Y21W, A40G, K41N, and F22Y.

7. SUMMARY

This chapter provides an overview of the methods that we use for consensus and correlation analyses and how they can be calculated and
interpreted. Most of the calculations are carried out in Microsoft Excel using various formulae and macros (the details provided are for Microsoft Office 2010). The general trends for such calculations have been illustrated by using the Kunitz-BPTI domain for a sample analysis. We also demonstrated the protocol for predicting stabilizing mutations in a protein using the metrics we calculated.

REFERENCES


