5-2019

BaMORC: A Software Package for Accurate and Robust ^{13}C Reference Correction of Protein NMR Spectra

Xi Chen
University of Kentucky, billchenxi@uky.edu

Andrey Smelter
University of Kentucky, andrey.smelter@uky.edu

Hunter N. B. Moseley
University of Kentucky, hunter.moseley@uky.edu

BaMORC: A Software Package for Accurate and Robust ^{13}C Reference Correction of Protein NMR Spectra

Abstract

We describe Bayesian Model Optimized Reference Correction (BaMORC), a software package that performs ^{13}C chemical shifts reference correction for either assigned or unassigned peak lists derived from protein NMR spectra. BaMORC provides an intuitive command line interface that allows non-nuclear magnetic resonance (NMR) experts to detect and correct ^{13}C chemical shift referencing errors of unassigned peak lists at the very beginning of NMR data analysis, further lowering the bar of expertise required for effective protein NMR analysis. Furthermore, BaMORC provides an application programming interface for integration into sophisticated protein NMR data analysis pipelines, both before and after the protein resonance assignment step.

Keywords

protein NMR, chemical shift reference correction, software package

Disciplines

Biochemistry, Biophysics, and Structural Biology | Bioinformatics | Biomedical | Systems and Communications

Notes/Citation Information

Published in *Natural Product Communications*, v. 14, issue 5, p. 1-7.

Natural Product Communications
May 2019: 1–7

DOI: 10.1177/1934578X19849142
journals.sagepub.com/home/npx

BaMORC: A Software Package for Accurate and Robust ^{13}C Reference Correction of Protein NMR Spectra

Xi Chen[1,2], Andrey Smelter[1,3], and Hunter N. B. Moseley[1,3,4,5]

Abstract

We describe Bayesian Model Optimized Reference Correction (BaMORC), a software package that performs ^{13}C chemical shifts reference correction for either assigned or unassigned peak lists derived from protein NMR spectra. BaMORC provides an intuitive command line interface that allows non-nuclear magnetic resonance (NMR) experts to detect and correct ^{13}C chemical shift referencing errors of unassigned peak lists at the very beginning of NMR data analysis, further lowering the bar of expertise required for effective protein NMR analysis. Furthermore, BaMORC provides an application programming interface for integration into sophisticated protein NMR data analysis pipelines, both before and after the protein resonance assignment step.

Keywords

protein NMR, chemical shift reference correction, software package

Received: October 25th, 2018; Accepted: January 7th, 2019.

Chemical shifts derived from protein nuclear magnetic resonance (NMR) spectra have a wide variety of uses including protein structure determination,[1,2] characterizing ligand binding,[3-5] and drug discovery and design.[6,7] However, deriving accurate chemical shift values requires the referencing of NMR spectra to a certain standard, typically an internal standard.[8,9] Due to human errors and a variety of experimental factors,[10,11] errors occur quite frequently in ^{13}C protein NMR data. An estimated 40% of the entries in the biological magnetic resonance bank (BMRB) have referencing issues.[12] The resulting referencing discrepancies are highly problematic since prior methods for reference correction required either assignment and/or structure,[13,14] which are the exact downstream aims that reference correction is trying to target. This leads to a co-dependency between reference correction and NMR structure determination, crippling the progress of many protein NMR analyses.

We therefore developed the Bayesian model optimized reference correction (BaMORC) method[15] that helps non-expert scientists to detect and correct ^{13}C C_α and C_β chemical shifts, at the beginning of the protein NMR analysis process, when chemical shifts are unassigned. Here, we describe the BaMORC method implemented in an easy-to-use software package written in the R programming language. BaMORC uses a Bayesian model to estimate an amino acid frequency from C_α and C_β chemical shift statistics inferred from the re-referenced protein chemical shift Database (RefDB),[12] with or without resonance assignment information. As shown in Figure 1, by optimizing the minimal between the actual amino acid frequency calculated from known protein sequence and an estimation based on the observed chemical shifts, BaMORC returns the reference correction value and re-referenced chemical shifts data. Figure 2 illustrates the required input and expected output generated by the BaMORC R package.

The BaMORC R package provides a command-line interface (CLI) for general use and an application programming interface (API) for users that are familiar with R programming, especially for use within an integrated development

[1] Department of Molecular and Cellular Biochemistry, University of Kentucky, Lexington, KY, USA
[2] Department of Statistics, University of Kentucky, Lexington, KY, USA
[3] Center for Environmental and Systems Biochemistry, Lexington, KY, USA
[4] Markey Cancer Center, University of Kentucky, Lexington, KY, USA
[5] Institute for Biomedical Informatics, University of Kentucky, Lexington, KY, USA

Corresponding Author:
Hunter N. B. Moseley, Department of Molecular and Cellular Biochemistry, University of Kentucky, Lexington, KY 40536-0093, USA.
Email: hunter.moseley@uky.edu

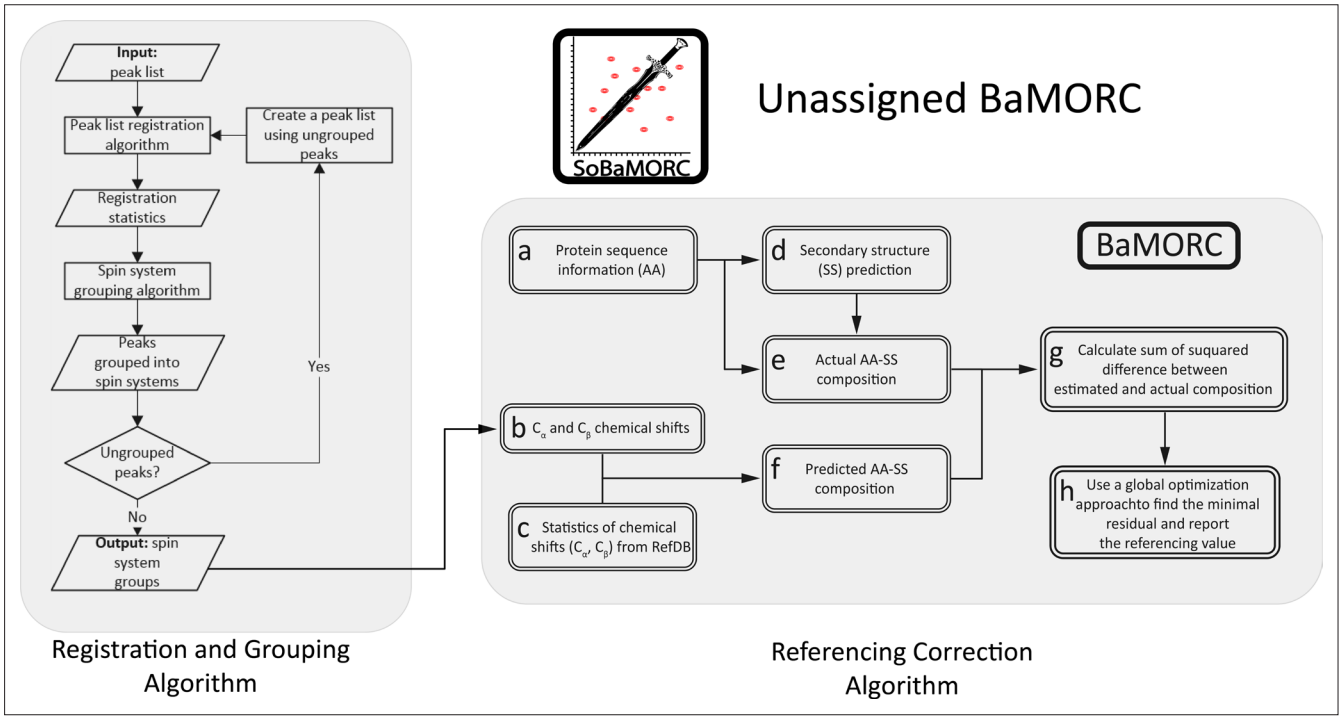

Figure 1. Overview of the (unassigned) BaMORC algorithm.

environment like RStudio.[16] As illustrated in Figure 2, the BaMORC R package can use the protein sequence and chemical shifts in a variety of unassigned and assigned formats including the NMR-STAR format utilized by the BMRB. As illustrated in Figure 2, the general row-based text format may be delimited by a comma or white space, but with the protein sequence on the first line followed by unassigned peaks or assigned C_α and C_β chemical shift pairs on following rows.

Each input file is referred to as a "task" within a larger "job". The BaMORC R package automatically interfaces with the registration, grouping and referencing algorithms to set up tasks and derive the most optimized correction values for a given input, and returns the corrected chemical shifts in csv format. The package can also accept a BMRB ID such as BMR 4020 as input to retrieve corresponding files from the BMRB web server, automatically parsing the file, correcting the referencing, and returning the same set of output as mentioned before.

We have evaluated BaMORC against 568 ^{13}C protein NMR datasets from the RefDB with 90% or higher completeness with respect to C_α and C_β chemical shift assignments. Outputted reference correction values should match closely to 0 ppm, since each dataset from RefDB has been reference-corrected using protein structure information. With chemical shift assignments, BaMORC provides

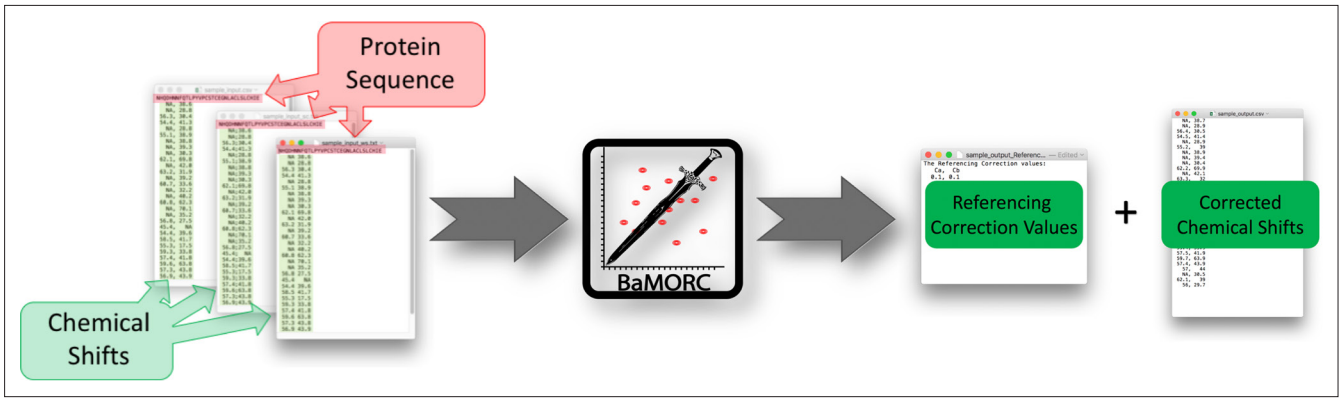

Figure 2. Input utilized and output generated by the BaMORC R package.

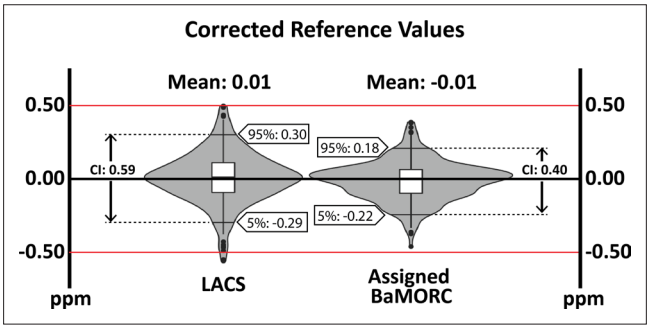

Figure 3. Comparison of assigned BaMORC to the LACS method.

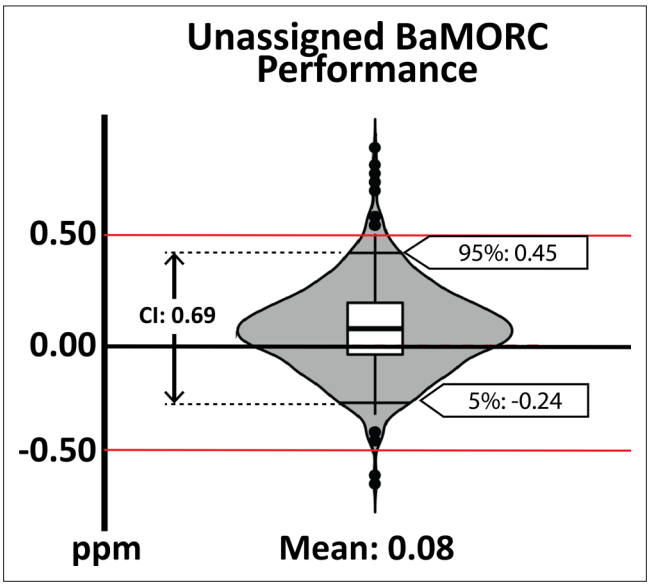

Figure 4. Unassigned BaMORC reference correction accuracy.

reference correction values within ±0.50 ppm for all datasets and within ±0.22 ppm for 90% of the datasets, representing a 90% confidence interval (CI) of 0.40 ppm (Figure 3).[15] This level of performance is superior to the prior state-of-the-art linear analysis of chemical shifts (LACS) method.[14]

However, in the real-world situation, ^{13}C reference correction is most valuable before protein resonance assignments are known. This situation is what the BaMORC package was really designed to address. The unassigned BaMORC method has two major components, grouping and referencing correction. With an input peak list, the grouping algorithm will return a list of C_α and C_β grouped peaks (spin systems) as output, which will be the input for the referencing correction algorithm, as shown in Figure 2. The grouping algorithm is a variance-informed DBSCAN algorithm that employs derived dimension-specific match tolerance values to group peaks into spin systems. A peak list registration step is used to derive the necessary match tolerance values.[17] In addition to the grouped peaks, the referencing correction component uses the JPred4[18] server to generate sequence-based secondary structure predictions and then calculates the reference correction.

Again we used the same 568 ^{13}C protein NMR datasets from the RefDB to evaluate the reference correction component of unassigned BaMORC, but without chemical shift assignments. As shown in Figure 4, the reference correction component of unassigned BaMORC provides reference correction values within ±0.45 ppm for 90% of the datasets, representing a 90% CI of 0.69 ppm.[15] This suggests that the unassigned BaMORC algorithm can achieve the same level of performance when handling unassigned ^{13}C protein NMR peak list data. This level of real-world performance is demonstrated with a set of peak lists derived from solution NMR HN(CO)CACB spectra for 10 different proteins. In this real-world evaluation, unassigned BaMORC provided reference correction values all within ±0.40 ppm.[15]

Experimental

Software

The Python programming language, version 3.6, is used for the grouping algorithm. The R programming language, version 3.4, is used for the BaMORC core component. The library dependencies are listed below:

Table 1. Summary of BaMORC Package Interface (API).

Command	Description	Example
read_file	Import local files	input_data = read_file(file_path = "./sample_input.txt", delim = "ws", assigned = T)
read_nmrstar_file	Import files in NMR-STAR format	input_data = read_nmrstar_file ("BMR4020.str")
read_db_file	Use BMRB ID to import files	input_data = read_db_file(id = "BMR4020")
bamorc	Using sequence, secondary structure and chemical shift data to estimate the reference correction value	bamorc(sequence, secondary_structure, chemical_shifts_input, from=-5, to = 5)
unassigned_bamorc	Using only sequence and chemical shift data to estimate the reference correction value	Unassigned_bamorc(sequence, chemical_shifts_input, from=-5, to = 5)

- Python Library Dependencies: Python (≥3.6), gcc (≥5.1)
- R Library Dependencies: R (≥3.4), data.table, tidyr, DEoptim, httr, docopt, stringr, jsonlite, readr, devtools, RBMRB, BMRBr

Experimental Data Sources

We used data from the RefDB to derive chemical shift statistics within the BaMORC package. For testing and evaluation, we used datasets from the RefDB and experimental peak lists from a variety of sources.

Installation

To use the BaMORC package, users must first install the R 3.4.x (or higher version) and Python 3.6.x (or higher version) interpreters on their machine. For Linux distributions, this is typically accomplished through the distribution's package management system. For other operating systems, installation may require a more manual procedure. R language is a language and environment for statistical computing.[19] The installation guide is located in the website [https://cran.r-project.org/web/packages/BaMORC/index.html] of the comprehensive R Archive Network [https://cran.r-project.org/]. Python language[20] can be install from this website [https://www.python.org/].

Installing BaMORC From the Command Line (Linux and Mac Only)

- To use BaMORC, the user first needs to install the package from the GitHub or CRAN.
- $ wget -q https://cran.r-project.org/src/contrib/BaMORC_<version > .tar.gz

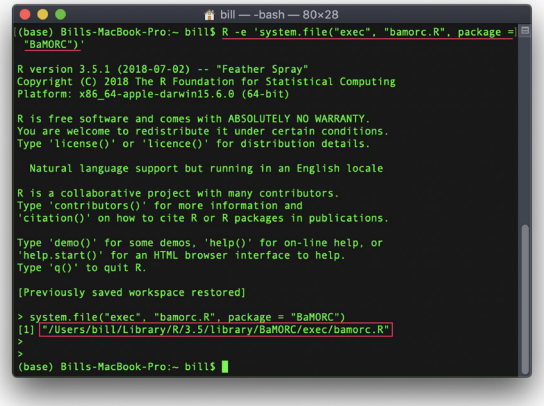

Figure 5. Finding the CLI run-script location.

- $ sudo R CMD INSTALL BaMORC_<version > .tar.gz

Install From Command Line via R Console

- $ R # to start R console
- >install.packages("BaMORC")

Install From R Console

- >install.packages("BaMORC")

Installing Unassigned BaMORC Dependencies

The unassigned BaMORC analysis requires the ssc (spin system creator) package, which includes a variance-informed implementation of the DBSCAN algorithm used for protein NMR spin system clustering. A docker container including the ssc package is required. Therefore, the user needs to install both docker and SSC docker image.

- Install Docker from https://www.docker.com/products/docker-desktop.
- Install SSC docker container after docker is installed by running following code:

>docker pull moseleybioinformaticslab/ssc.

The BaMORC Application Programming Interface

After importing the BaMORC in R either on R Console or in RStudio, the user will first read in NMR chemical shifts data via the read_file function with parameters of file path, file delimiter, and a flag that indicates whether data are either assigned or unassigned. BaMORC currently supports file delimiters of comma, semicolon, and whitespace. For users who want to run an analysis on an existing dataset from the BMRB (NMR-STAR version 2 and 3), they can use either the read_nmrstar_file function with a parameter for a local file path or the read_db_file function with a parameter for the BMRB ID and a flag that indicates whether data are assigned or unassigned. If read_db_file is used, BaMORC will utilize the BMRB web API to fetch the corresponding BMRB entry matching the ID. Table 1 shows common usage patterns for reading input data into the BaMORC referencing correction analysis pipeline. For a full list of available conversion options and more detailed examples and documentation of all the functions, please refer to "The BaMORC Reference" and "Quickstart."

Next, the user will pass the input data as parameters to the bamorc() or unassigned_bamorc()function, which will perform the reference correction analysis. Both functions utilize the output from the read-in functions mentioned above and will perform a secondary structure estimation based on the provided protein sequence if secondary structure

Table 2. BaMORC CLI Commands and Their Parameters.

Command	Parameter	Example
Assigned	Required parameter	
	Input file path or ID	--table=sample_input.csv or --bmrb=bmr4020 or --id=BMR4020
	Optional parameter	
	Estimation range	--range=(-5,5)
	Delimiter	--delim=comma
	Output path	--output=sample_output.csv
	Report file path	--report=sample_report.txt
Unassigned	Required parameter	
	Input file path	--table=sample_input.csv
	Optional parameter	
	Grouped peaklist or not	--grouped=true
	Protein sequence	--seq=sample_sequence.txt
	Search range	--range=(-5,5)
	Output path	--output=sample_output.csv
	Report file path	--report=sample_report.txt
Help	Help menu	--h or -help
Version	Version number	--v or -version

information is not provided. Through a series of optimization calculations (for details refer to paper[15]), bamorc() and unassigned_bamorc() will return the estimated referencing correction value in a plain text file and corrected chemical shifts for both C_α and C_β as a table, as shown in Figure 2. The user can optionally customize the search range. Table 1 contains a basic example of calling each function. For detailed examples and expected outputs of BaMORC API functions, refer to the online documentation: https://moseleybioinformaticslab.github.io/BaMORC/index.htm.

The BaMORC Command Line Interface

The BaMORC CLI is an extension of the BaMORC package, aimed at the broader NMR community that is not familiar with R programming language. To use BaMORC CLI, the user needs to find the CLI run-script first by opening a terminal and typing the command highlighted in Figure 5.

>R e 'system.file("exec," "bamorc.R," package = "BaMORC")'

The user can then execute the appropriate command listed in Table 2 to run an analysis. Similar to the package, the BaMORC CLI has three major modules: assigned and unassigned reference correction for assigned and unassigned protein NMR data and a miscellaneous collection of other useful tasks. Table 2 lists the components of the CLI and their associated parameters.

To help the user transition between the API and CLI, Table 3 illustrates common BaMORC CLI usage examples with corresponding BaMORC API examples. The CLI is utilized within a command line terminal on Linux and Mac computers. For windows user, refer to our online documentation for more details.

We have developed online documentations, available at: https://moseleybioinformaticslab.github.io/BaMORC/index.html.

Reporting Summary

Further information on the algorithms mentioned above and their development is available.[15]

Code Availability

Source code is available at https://github.com/MoseleyBioinformaticsLab/BaMORC. [The package has been submitted to CRAN and should be available from CRAN soon. We will add a sentence about its availability from CRAN and update installation instructions when the evaluation process is finished]. The code is published under a modified open source BSD-3 license. Academic researchers are free to use it without restriction, except for proper citation. This repository includes code for the BaMORC referencing correction pipeline. For the registration and grouping algorithm, refer to https://github.com/MoseleyBioinformaticsLab/ssc.[21] For further information and assistance visit our laboratory website: http://bioinformatics.cesb.uky.edu.

Data Availability

Datasets are available at: https://doi.org/10.6084/m9.figshare.5270755.v1

Table 3. BaMORC CLI Usage and Corresponding API Commands.

CLI	API
Assigned BaMORC: For user's own protein NMR spectra result	
$ bamorc.R assigned --table=./sample_input.csv --ppm_range=(-5,5) --output=./sample_output.csv --delimiter=comma --report=./sample_report.txt	>user_input = read_file(file_path="./sample_input.csv", delim="comma", assigned = f) >result = bamorc(sequence = user_input[[1]], chemical_shifts_input = user_input[[2]], from = -5, to = 5)
Assigned BaMORC: For data in NMR-STAR format	
bamorc.R assigned --bmrb=BMR4020.str --ppm_range=(-5,5) --output=./sample_output.csv --delimiter=comma --report=./sample_report.txt	>bmrb_format_data = read_nmrstar_file("BMR4020.str") >result = bamorc(sequence = bmrb_format_data[[1]], chemical_shifts_input = bmrb_format_data [[2]], from = -5, to = 5)
Assigned BaMORC: For data already existing in BMRB database	
bamorc.R assigned --id=BMR4020 --ppm_range=(-5,5) --output=./smple_output.csv --delimiter=comma --report=./sample_report.txt	>existing_data = read_db_file(id="BMR4020") >result = bamorc(sequence = existing_data[[1]], chemical_shifts_input = existing_data [[2]], from=-5, to = 5)
Unassigned BaMORC: For user's own protein NMR spectra result	
bamorc.R unassigned table=./sample_input.csv --ppm_range=(-5,5) --output=./sample_output.csv --delimiter=comma --report=./sample_report.txt	>user_input = read_file(file_path="./sample_input.csv", delim="comma") >result = unassigned_bamorc(sequence = user_input[[1]], from = -5, to = 5)
BaMORC CLI: other commands (CLI only)	
bamorc.R valid_ids	To show all the valid BMRB file IDS
bamorc.R -h	To show help menu
bamorc.R -v	To show BaMORC version

Acknowledgments

The authors acknowledge support from the National Science Foundation grant and National Institutes of Health grants.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) declared the following potential conflicts of interest with respect to the research, authorship, and/or publication of this article: Support for this research was provided by the National Science Foundation grant NSF 1419282 (Hunter N.B. Moseley) and National Institutes of Health grants NIH UL1TR001998-01 (Philip Kern) and NIH P30CA177558 (Mark Evers).

References

1. Sattler M, Schleucher J, Griesinger C. Heteronuclear multidimensional NMR experiments for the structure determination of proteins in solution employing pulsed field gradients. *Prog Nucl Magn Reson Spectrosc*. 1999;34(2):93-158.
2. Shen Y, Lange O, Delaglio F, et al. Consistent blind protein structure generation from NMR chemical shift data. *Proc Natl Acad Sci U S A*. 2008;105(12):4685-4690.
3. Williamson MP. Using chemical shift perturbation to characterise ligand binding. *Prog Nucl Magn Reson Spectrosc*. 2013;73:1-16.
4. Jayalakshmi V, Rama Krishna N, Krishna NR. CORCEMA refinement of the bound ligand conformation within the protein binding pocket in reversibly forming weak complexes using STD-NMR intensities. *J Magn Reson*. 2004;168(1):36-45.
5. Moseley HN, Curto EV, Krishna NR. Complete relaxation and conformational exchange matrix (CORCEMA) analysis of NOESY spectra of interacting systems; two-dimensional transferred NOESY. *J Magn Reson B*. 1995;108(3):243-261.
6. Anderson AC. The process of structure-based drug design. *Chem Biol*. 2003;10(9):787-797.
7. Shuker SB, Hajduk PJ, Meadows RP, Fesik SW. Discovering high-affinity ligands for proteins: SAR by NMR. *Science*. 1996;274(5292):1531-1534.
8. Markley JL, Bax A, Arata Y, et al. Recommendations for the presentation of NMR structures of proteins and nucleic acids. IUPAC-IUBMB-IUPAB Inter-Union task group on the standardization of data bases of protein and nucleic acid structures determined by NMR spectroscopy. *J Biomol NMR*. 1998;12(1):1-23.
9. Wishart DS, Bigam CG, Yao J, et al. ^{1}H, ^{13}C and ^{15}N chemical shift referencing in biomolecular NMR. *J Biomol NMR*. 1995;6(2):135-140.
10. Nowick JS, Khakshoor O, Hashemzadeh M, Brower JO. DSA: a new internal standard for NMR studies in aqueous solution. *Org Lett*. 2003;5(19):3511-3513.
11. Ulrich EL, Akutsu H, Doreleijers JF, et al. BioMagResBank. *Nucleic Acids Res*. 2008;36(Database issue):D402-D408.

12. Zhang H, Neal S, Wishart DS. RefDB: a database of uniformly referenced protein chemical shifts. *J Biomol NMR*. 2003;25(3):173-195.
13. Han B, Liu Y, Ginzinger SW, Wishart DS. SHIFTX2: significantly improved protein chemical shift prediction. *J Biomol NMR*. 2011;50(1):43-57.
14. Wang L, Eghbalnia HR, Bahrami A, Markley JL. Linear analysis of carbon-13 chemical shift differences and its application to the detection and correction of errors in referencing and spin system identifications. *J Biomol NMR*. 2005;32(1):13-22.
15. Chen X, Smelter A, Moseley HNB. Automatic ^{13}C chemical shift reference correction for unassigned protein NMR spectra. *J Biomol NMR*. 2018;72(1-2):11-28.
16. RStudio RT. *RStudio: Integrated Development for R*. Boston, MA: RStudio, Inc; 2015.
17. Smelter A, Astra M, Moseley HNB. A fast and efficient python library for Interfacing with the biological magnetic resonance data bank. *BMC Bioinformatics*. 2017;18(1):175-186.
18. Drozdetskiy A, Cole C, Procter J, Barton GJ. JPred4: a protein secondary structure prediction server. *Nucleic Acids Res*. 2015;43(W1):W389-W394.
19. Team RC2013. R: A Language and Environment for Statistical Computing. https://scholar.google.com/scholar?hl=en&as_sdt=0%2C18&q=R%3A+A+language+and+environment+for+statistical+computing+2018&btnG=
20. Van Rossum G, Drake FL. *The Python Language Reference Manual*. Bristol, UK: Network Theory Ltd; 2011.
21. Smelter A, Rouchka EC, Moseley HNB. Detecting and accounting for multiple sources of positional variance in peak list registration analysis and spin system grouping. *J Biomol NMR*. 2017;68(4):281-296.