

---

---

# Rcpi Quick Reference Card

---

---

Nan Xiao, Dongsheng Cao, Qingsong Xu

Package Version: Release 3

2015-04-16



**COMPUTATIONAL BIOLOGY &  
DRUG DESIGN GROUP  
CENTRAL SOUTH UNIV., CHINA**

Table 1: Retrieving protein sequence data from various online databases

Function name	Function description
<code>getProt()</code>	Retrieve protein sequence in FASTA format or PDB format from various online databases
<code>getFASTAFromUniProt()</code>	Retrieve protein sequence in FASTA format from UniProt
<code>getFASTAFromKEGG()</code>	Retrieve protein sequence in FASTA format from KEGG
<code>getPDBFromRCSBPDB()</code>	Retrieve protein sequence in PDB Format from RCSB PDB
<code>getSeqFromUniProt()</code>	Retrieve protein sequence from UniProt
<code>getSeqFromKEGG()</code>	Retrieve protein sequence from KEGG
<code>getSeqFromRCSBPDB()</code>	Retrieve protein sequence from RCSB PDB

Table 2: Retrieving drug molecular data from various online databases

Function name	Function description
<code>getDrug()</code>	Retrieve drug molecules in MOL format and SMILES format from various online databases
<code>getMolFromDrugBank()</code>	Retrieve drug molecules in MOL format from DrugBank
<code>getMolFromPubChem()</code>	Retrieve drug molecules in MOL format from PubChem
<code>getMolFromChEMBL()</code>	Retrieve drug molecules in MOL format from ChEMBL
<code>getMolFromKEGG()</code>	Retrieve drug molecules in MOL format from the KEGG
<code>getMolFromCAS()</code>	Retrieve drug molecules in InChI format from CAS
<code>getSmiFromDrugBank()</code>	Retrieve drug molecules in SMILES format from DrugBank
<code>getSmiFromPubChem()</code>	Retrieve drug molecules in SMILES format from PubChem
<code>getSmiFromChEMBL()</code>	Retrieve drug molecules in SMILES format from ChEMBL
<code>getSmiFromKEGG()</code>	Retrieve drug molecules in SMILES format from KEGG

Table 3: Calculating commonly used protein sequence derived descriptors

Function name	Descriptor name	Descriptor group
<code>extractProtAAC()</code>	Amino acid composition	Amino acid composition
<code>extractProtDC()</code>	Dipeptide composition	
<code>extractProtTC()</code>	Tripeptide composition	
<code>extractProtMoreauBroto()</code>	Normalized Moreau-Broto autocorrelation	Autocorrelation
<code>extractProtMoran()</code>	Moran autocorrelation	
<code>extractProtGeary()</code>	Geary autocorrelation	
<code>extractProtCTDC()</code>	Composition	CTD
<code>extractProtCTDT()</code>	Transition	
<code>extractProtCTDD()</code>	Distribution	
<code>extractProtCTriad()</code>	Conjoint Triad	Conjoint Triad
<code>extractProtSOCN()</code>	Sequence-order-coupling number	Quasi-sequence-order
<code>extractProtQSO()</code>	Quasi-sequence-order descriptors	
<code>extractProtPAAC()</code>	Pseudo-amino acid composition	Pseudo-amino acid composition
<code>extractProtAPAAC()</code>	Amphiphilic pseudo-amino acid composition	
<code>AAindex</code>	AAindex data of 544 physicochemical and biological properties for 20 amino acids	Dataset

Table 4: Generating profile-based protein representations

Function name	Function description
<code>extractProtPSSM()</code>	Compute PSSM (Position-Specific Scoring Matrix) for given protein sequence or peptides
<code>extractProtPSSMFeature()</code>	Profile-based protein representation derived by PSSM
<code>extractProtPSSMAcc()</code>	Profile-based protein representation derived by PSSM and auto cross covariance (ACC)

Table 5: Generating scales-based descriptors for proteochemometrics modeling

Function name	Descriptor class	Derived by
<code>extractPCMScales()</code>	Generalized scales-based descriptors derived by principal components analysis (PCA)	Principal components analysis
<code>extractPCMPPropScales()</code>	Generalized scales-based descriptors derived by amino acid properties (AAindex)	
<code>extractPCMDescScales()</code>	Generalized scales-based descriptors derived by 2D and 3D molecular descriptors (Topological, WHIM, VHSE, etc.)	
<code>extractPCMFAScales()</code>	Generalized scales-based descriptors derived by factor analysis	Factor analysis
<code>extractPCMMDScales()</code>	Generalized scales-based descriptors derived by multidimensional scaling (MDS)	Multidimensional scaling
<code>extractPCMBLOSUM()</code>	Generalized BLOSUM and PAM matrix-derived descriptors	Substitution matrix
<code>acc()</code>	Auto cross covariance (ACC) for generating scales-based descriptors of the same length	

Table 6: Pre-calculated molecular descriptor sets of the 20 amino acids in **Rcpi** for generating scales-based descriptors for proteochemometrics modeling. Note that the non-informative descriptors (like the descriptors have only one value across all the 20 amino acids) in these datasets have already been filtered out.

Dataset name	Dataset description	Dimensionality	Calculated by
OptAA3d	Optimized 20 amino acids	–	MOE
AA2DACOR	2D autocorrelations descriptors	92	Dragon
AA3DMoRSE	3D-MoRSE descriptors	160	Dragon
AAACF	Atom-centred fragments descriptors	6	Dragon
AABurden	Burden Eigenvalues descriptors	62	Dragon
AAConn	Connectivity indices descriptors	33	Dragon
AAConst	Constitutional descriptors	23	Dragon
AAEdgeAdj	Edge adjacency indices descriptors	97	Dragon
AAEigIdx	Eigenvalue-based indices descriptors	44	Dragon
AAFGC	Functional group counts descriptors	5	Dragon
AAGeom	Geometrical descriptors	41	Dragon
AAGETAWAY	GETAWAY descriptors	194	Dragon
AAInfo	Information indices descriptors	47	Dragon
AAMolProp	Molecular properties descriptors	12	Dragon
AARandic	Randic molecular profiles descriptors	41	Dragon
AARDF	RDF descriptors	82	Dragon
AATopo	Topological descriptors	78	Dragon
AATopoChg	Topological charge indices descriptors	15	Dragon
AAWalk	Walk and path counts descriptors	40	Dragon
AAWHIM	WHIM descriptors	99	Dragon
AACPSA	CPSA descriptors	41	Accelrys Discovery Studio
AADescA11	All the 2D descriptors calculated by Dragon	1171	Dragon
AAMOE2D	All the 2D descriptors calculated by MOE	148	MOE
AAMOE3D	All the 3D descriptors calculated by MOE	143	MOE
AABLOSUM45	BLOSUM45 matrix for 20 amino acids	20 × 20	Biostrings
AABLOSUM50	BLOSUM50 matrix for 20 amino acids	20 × 20	Biostrings
AABLOSUM62	BLOSUM62 matrix for 20 amino acids	20 × 20	Biostrings
AABLOSUM80	BLOSUM80 matrix for 20 amino acids	20 × 20	Biostrings
AABLOSUM100	BLOSUM100 matrix for 20 amino acids	20 × 20	Biostrings
AAPAM30	PAM30 matrix for 20 amino acids	20 × 20	Biostrings
AAPAM40	PAM40 matrix for 20 amino acids	20 × 20	Biostrings
AAPAM70	PAM70 matrix for 20 amino acids	20 × 20	Biostrings
AAPAM120	PAM120 matrix for 20 amino acids	20 × 20	Biostrings
AAPAM250	PAM250 matrix for 20 amino acids	20 × 20	Biostrings

Table 7: Molecular descriptors

Function name	Descriptor name
<code>extractDrugAIO()</code>	All the molecular descriptors in the <b>Rcpi</b> package
<code>extractDrugALOGP()</code>	Atom additive logP and molar refractivity values descriptor
<code>extractDrugAminoAcidCount()</code>	Number of amino acids
<code>extractDrugApol()</code>	Sum of the atomic polarizabilities
<code>extractDrugAromaticAtomsCount()</code>	Number of aromatic atoms
<code>extractDrugAromaticBondsCount()</code>	Number of aromatic bonds
<code>extractDrugAtomCount()</code>	Number of atom descriptor
<code>extractDrugAutocorrelationCharge()</code>	Moreau-Broto autocorrelation descriptors using partial charges
<code>extractDrugAutocorrelationMass()</code>	Moreau-Broto autocorrelation descriptors using atomic weight
<code>extractDrugAutocorrelationPolarizability()</code>	Moreau-Broto autocorrelation descriptors using polarizability
<code>extractDrugBCUT()</code>	BCUT, the eigenvalue based descriptor
<code>extractDrugBondCount()</code>	Number of bonds of a certain bond order
<code>extractDrugBPol()</code>	Sum of the absolute value of the difference between atomic polarizabilities of all bonded atoms in the molecule
<code>extractDrugCarbonTypes()</code>	Topological descriptor characterizing the carbon connectivity in terms of hybridization
<code>extractDrugChiChain()</code>	Kier & Hall Chi chain indices of orders 3, 4, 5, 6 and 7
<code>extractDrugChiCluster()</code>	Kier & Hall Chi cluster indices of orders 3, 4, 5 and 6
<code>extractDrugChiPath()</code>	Kier & Hall Chi path indices of orders 0 to 7
<code>extractDrugChiPathCluster()</code>	Kier & Hall Chi path cluster indices of orders 4, 5 and 6
<code>extractDrugCPSA()</code>	Descriptors combining surface area and partial charge information
<code>extractDrugDesc0B()</code>	Molecular descriptors provided by OpenBabel
<code>extractDrugECI()</code>	Eccentric connectivity index descriptor
<code>extractDrugFMF()</code>	FMF descriptor
<code>extractDrugFragmentComplexity()</code>	Complexity of a system
<code>extractDrugGravitationalIndex()</code>	Mass distribution of the molecule
<code>extractDrugHBondAcceptorCount()</code>	Number of hydrogen bond acceptors
<code>extractDrugHBondDonorCount()</code>	Number of hydrogen bond donors
<code>extractDrugHybridizationRatio()</code>	Molecular complexity in terms of carbon hybridization states
<code>extractDrugIPMolecularLearning()</code>	Ionization potential
<code>extractDrugKappaShapeIndices()</code>	Kier & Hall Kappa molecular shape indices
<code>extractDrugKierHallSmarts()</code>	Number of occurrences of the E-State fragments
<code>extractDrugLargestChain()</code>	Number of atoms in the largest chain
<code>extractDrugLargestPiSystem()</code>	Number of atoms in the largest Pi chain
<code>extractDrugLengthOverBreadth()</code>	Ratio of length to breadth descriptor
<code>extractDrugLongestAliphaticChain()</code>	Number of atoms in the longest aliphatic chain
<code>extractDrugMannholdLogP()</code>	LogP based on the number of carbons and hetero atoms
<code>extractDrugMDE()</code>	Molecular Distance Edge (MDE) descriptors for C, N and O
<code>extractDrugMomentOfInertia()</code>	Principal moments of inertia and ratios of the principal moments
<code>extractDrugPetitjeanNumber()</code>	Petitjean number of a molecule
<code>extractDrugPetitjeanShapeIndex()</code>	Petitjean shape indices
<code>extractDrugRotatableBondsCount()</code>	Number of non-rotatable bonds on a molecule
<code>extractDrugRuleOfFive()</code>	Number failures of the Lipinski's Rule Of Five
<code>extractDrugTPSA()</code>	Topological Polar Surface Area (TPSA)
<code>extractDrugVABC()</code>	Volume of a molecule
<code>extractDrugVAdjMa()</code>	Vertex adjacency information of a molecule
<code>extractDrugWeight()</code>	Total weight of atoms
<code>extractDrugWeightedPath()</code>	Weighted path (Molecular ID)
<code>extractDrugWHIM()</code>	Holistic descriptors described by Todeschini et al.
<code>extractDrugWienerNumbers()</code>	Wiener path number and wiener polarity number
<code>extractDrugXLogP()</code>	Prediction of logP based on the atom-type method called XLogP
<code>extractDrugZagrebIndex()</code>	Sum of the squared atom degrees of all heavy atoms

Table 8: Molecular fingerprints

Function name	Fingerprint type
<code>extractDrugStandard()</code>	Standard molecular fingerprints (in compact format)
<code>extractDrugStandardComplete()</code>	Standard molecular fingerprints (in complete format)
<code>extractDrugExtended()</code>	Extended molecular fingerprints (in compact format)
<code>extractDrugExtendedComplete()</code>	Extended molecular fingerprints (in complete format)
<code>extractDrugGraph()</code>	Graph molecular fingerprints (in compact format)
<code>extractDrugGraphComplete()</code>	Graph molecular fingerprints (in complete format)
<code>extractDrugHybridization()</code>	Hybridization molecular fingerprints (in compact format)
<code>extractDrugHybridizationComplete()</code>	Hybridization molecular fingerprints (in complete format)
<code>extractDrugMACCS()</code>	MACCS molecular fingerprints (in compact format)
<code>extractDrugMACCSComplete()</code>	MACCS molecular fingerprints (in complete format)
<code>extractDrugEstate()</code>	E-State molecular fingerprints (in compact format)
<code>extractDrugEstateComplete()</code>	E-State molecular fingerprints (in complete format)
<code>extractDrugPubChem()</code>	PubChem molecular fingerprints (in compact format)
<code>extractDrugPubChemComplete()</code>	PubChem molecular fingerprints (in complete format)
<code>extractDrugKR()</code>	KR (Klekota and Roth) molecular fingerprints (in compact format)
<code>extractDrugKRComplete()</code>	KR (Klekota and Roth) molecular fingerprints (in complete format)
<code>extractDrugShortestPath()</code>	Shortest Path molecular fingerprints (in compact format)
<code>extractDrugShortestPathComplete()</code>	Shortest Path molecular fingerprints (in complete format)
<code>extractDrugOBFP2()</code>	FP2 molecular fingerprints
<code>extractDrugOBFP3()</code>	FP3 molecular fingerprints
<code>extractDrugOBFP4()</code>	FP4 molecular fingerprints
<code>extractDrugOBMACCS()</code>	MACCS molecular fingerprints

Table 9: Protein-protein and compound-protein interaction descriptors

Function name	Function description
<code>getPPI()</code>	Generating protein-protein interaction descriptors
<code>getCPI()</code>	Generating compound-protein interaction descriptors

Table 10: Similarity and similarity searching

Function name	Function description
<code>calcDrugFPSim()</code>	Calculate drug molecule similarity derived by molecular fingerprints
<code>calcDrugMCSSim()</code>	Calculate drug molecule similarity derived by maximum common substructure search
<code>searchDrug()</code>	Parallelized drug molecule similarity search by molecular fingerprints similarity or maximum common substructure search
<code>calcTwoProtSeqSim()</code>	Similarity calculation based on sequence alignment for a pair of protein sequences
<code>calcParProtSeqSim()</code>	Parallelized protein sequence similarity calculation based on sequence alignment
<code>calcTwoProtGOSim()</code>	Similarity calculation based on Gene Ontology (GO) similarity between two proteins
<code>calcParProtGOSim()</code>	Protein similarity calculation based on Gene Ontology (GO) similarity

Table 11: Protein sequence data manipulation

Function name	Function description
<code>readFASTA()</code>	Read protein sequences in FASTA format
<code>readPDB()</code>	Read protein sequences in PDB format
<code>segProt()</code>	Protein sequence segmentation
<code>checkProt()</code>	Check if the protein sequence's amino acid types are the 20 default types

Table 12: Molecular data manipulation

Function name	Function description
<code>readMolFromSDF()</code>	Read molecules from SDF files and return parsed Java molecular object
<code>readMolFromSmi()</code>	Read molecules from SMILES files and return parsed Java molecular object or plain text list
<code>convMolFormat()</code>	Chemical file formats conversion

**Affiliation:**

Nan Xiao  
School of Mathematics and Statistics  
Central South University  
Changsha, Hunan, P. R. China  
E-mail: [road2stat@gmail.com](mailto:road2stat@gmail.com)  
URL: <http://r2s.name>

Dongsheng Cao  
School of Pharmaceutical Sciences  
Central South University  
Changsha, Hunan, P. R. China  
E-mail: [oriental-cds@163.com](mailto:oriental-cds@163.com)  
URL: <http://cbdd.csu.edu.cn>

Qingsong Xu  
School of Mathematics and Statistics  
Central South University  
Changsha, Hunan, P. R. China  
E-mail: [dasongxu@gmail.com](mailto:dasongxu@gmail.com)