

# **Bio3D**: Interactive Tools for Structural Bioinformatics

<http://thegrantlab.org/bio3d/>

# What is Bio3D

A freely distributed and widely used **R package** for structural bioinformatics.

Provides a **large number of integrated utilities** for biomolecular sequence, structure and dynamics analysis.

Provides an unparalleled **interactive environment** for evolutionary and comparative dynamics analysis.\*

# What Bio3D is **NOT**

A performance optimized software library for incorporation into your own C/C++ etc. programs

A molecular graphics program with a slick GUI

Backed by a commercial guarantee or license agreement

# Bio3D Features

Search, annotate and analyze using online sequence and structure databases.

Perform multivariate analysis of large structural and sequence datasets (PCA, MDS, DA, etc.).

High throughput ensemble normal mode analysis (NMA) of heterogenous structures.

Dynamic network analysis from NMA, molecular dynamics and experimental structure ensembles.

...

# Bio3D Features

**As well as foundational functionality for:**

3D visualization, alignment, superposition, atom selection, rigid and dynamic domain analysis, sequence and conformational clustering, distance matrix analysis, sequence and structural conservation analysis, etc...

# Features = **functions()**

> **library(bio3d)**

> **l**bio3d**()**

[1] "aa.index"

"aa.table"

[3] "aa123"

"aa2index"

[5] "aa2mass"

"aa321"

...<cut>...

[307] "write.fasta"

"write.ncdf"

[309] "write.pdb"

"write.pir"

[311] "write.pqr"

"xyz2atom"

[313] "xyz2z.pca"

"z2xyz.pca"

# Features = **functions()**

## > **help(package="bio3d")**

aa.index	AAindex: Amino Acid Index Database
aa.table	Table of Relevant Amino Acids
aa123	Convert 1-letter and 3-letter Amino Acid Codes
aa2index	Convert Sequence to AAindex Values
aa2mass	Amino Acid Residues to Mass Converter
...<cut>...	
write.fasta	Write FASTA Formated Sequences
write.ncdf	Write AMBER Binary netCDF files
write.pdb	Write PDB Format Coordinate File
write.pir	Write PIR Formated Sequences
write.pqr	Write PQR Format Coordinate File

# Features = **functions()**

> **help(read.pdb)**

read.pdb                                    package:bio3d                                    R Documentation

Description:

Read a Protein Data Bank (PDB) coordinate file.

Usage:

```
read.pdb(file, multi = FALSE, rm.insert = FALSE,  
          rm.alt = TRUE, verbose = TRUE)
```

Arguments:

file: the name of the PDB file to be read.

multi: logical, if TRUE multiple ATOM records are read for all models in multi-model files.

...<cut>...

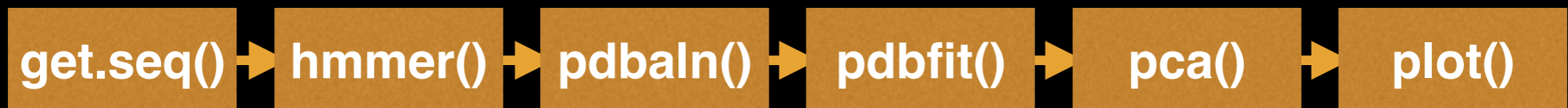


<b>Modularity</b>	Core bio3d functions are modular and work well with others
<b>Interactivity</b>	R/bio3d offers an unparalleled exploratory data analysis environment
<b>Infrastructure</b>	Access to existing tools and cutting-edge statistical and graphical methods
<b>Support</b>	Extensive documentation and tutorials available online for both bio3d and R
<b>R Philosophy</b>	Encourages open standards and reproducibility

<b>Modularity</b>	Core bio3d functions are modular and work well with others
<b>Interactivity</b>	R/bio3d offers an unparalleled exploratory data analysis environment
<b>Infrastructure</b>	Access to existing tools and cutting-edge statistical and graphical methods
<b>Support</b>	Extensive documentation and tutorials available online for both bio3d and R
<b>R Philosophy</b>	Encourages open standards and reproducibility

# Modularity

Bio3D was designed to allow users to interactively build complex workflows by interfacing smaller '**modular**' **functions** together.



An alternative approach is to write a **single complex program** that takes raw data as input, and after hours of data processing, outputs publication figures and a final table of results.

All-in-one custom 'Monster' program

Which would you prefer and why?



Modular

vs



Custom

# Advantages/Disadvantages

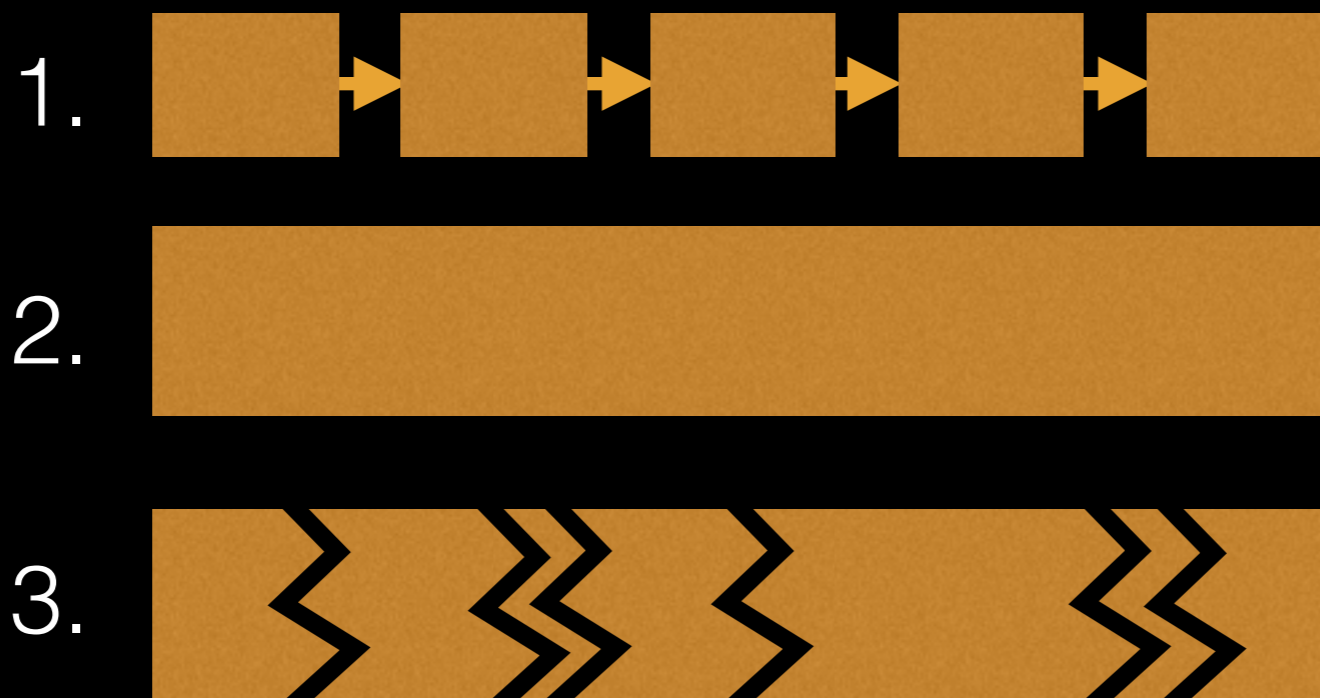
The 'monster approach' is **customized to a particular project** but results in **massive, fragile** and difficult to modify (therefore **inflexible, untransferable, and error prone**) code.

With **modular workflows**, it's easier to:

- **Spot errors** and figure out where they're occurring by inspecting intermediate results.
- **Experiment** with alternative methods by swapping out components.
- **Tackle novel problems** by remixing existing modular tools.

# 'Scripting' approach

Another common approach to bioinformatics data analysis is to write individual scripts in Perl/ Python/ Awk/ C etc. to carry out each subsequent step of an analysis



This can offer many advantages but can be challenging to make robustly modular and interactive.

# Interactivity & exploratory data analysis

Learning R/Bio3D will give you the freedom to explore and experiment with your biomolecular data.

*“Data analysis, like experimentation, must be considered as a highly interactive, iterative process, whose actual steps are selected segments of a stubbornly branching, tree-like pattern of possible actions”*. [**J. W. Tukey**]

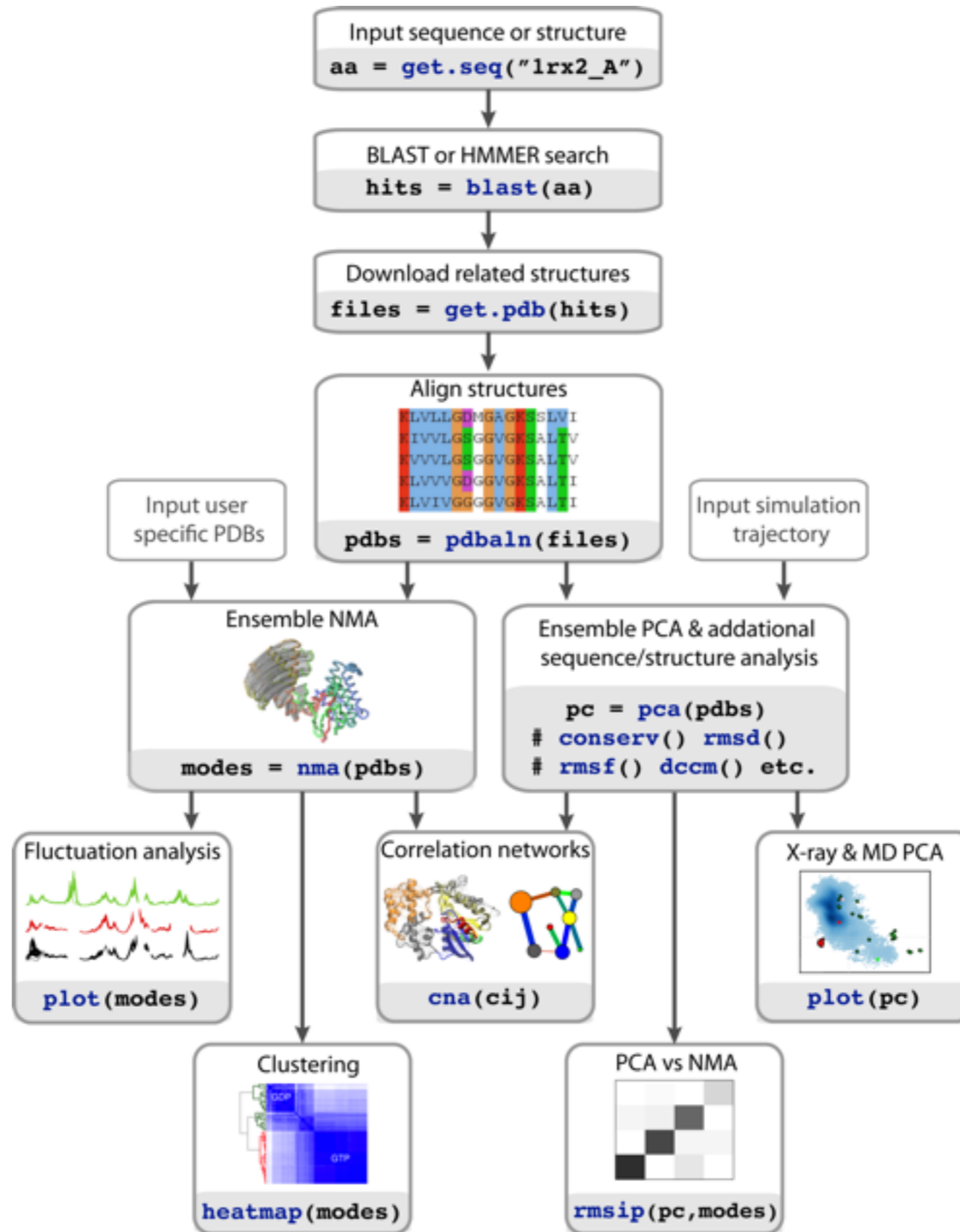
# Interactivity & exploratory data analysis

Learning R/Bio3D will give you the freedom to explore and experiment with your biomolecular data.

*“Data analysis, like experimentation, must be considered as a highly interactive, iterative process, whose actual steps are selected segments of a stubbornly branching, tree-like pattern of possible actions”*. [J. W. Tukey]

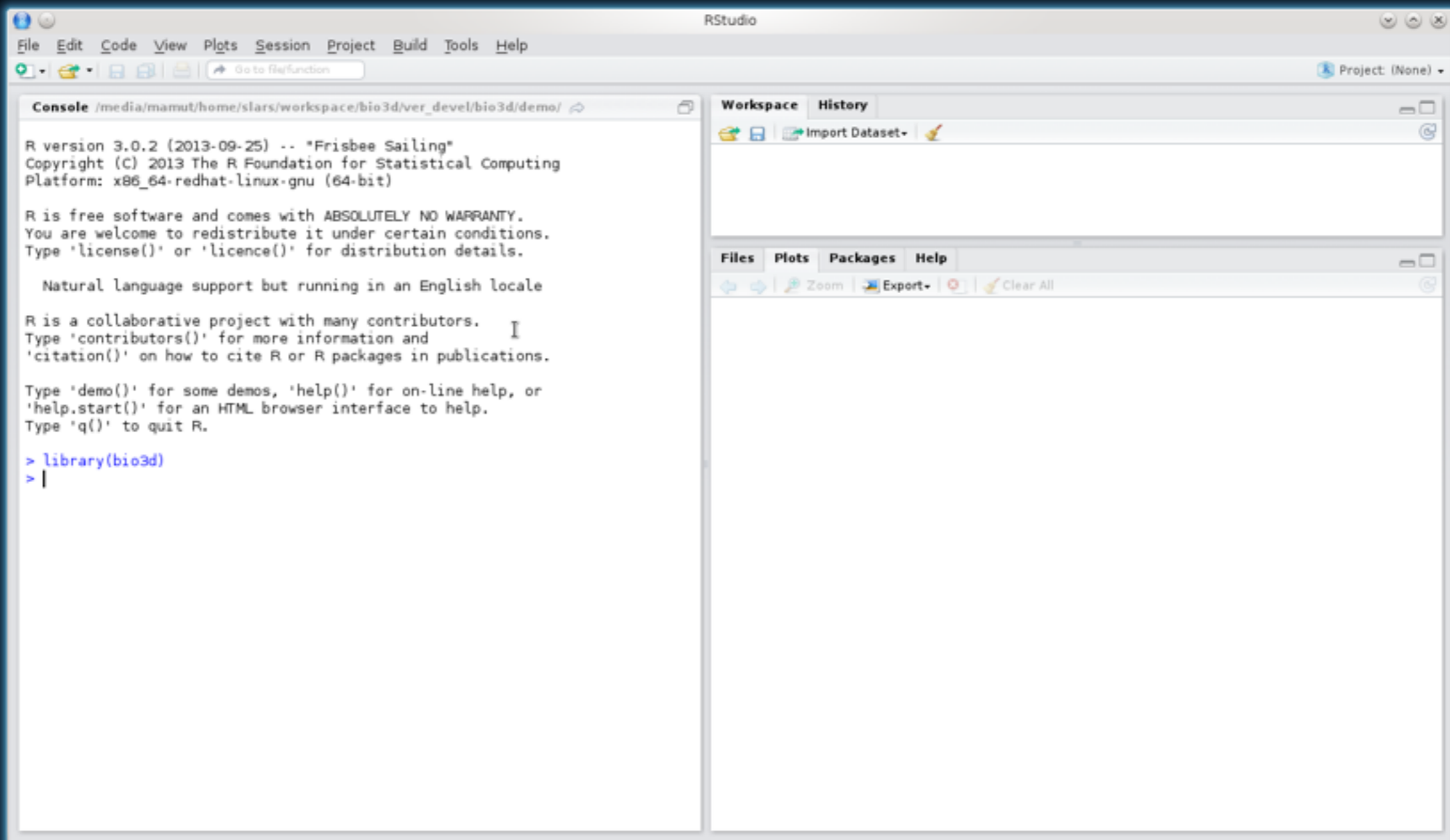
Structural bioinformatics data is intrinsically **high dimensional** and frequently ‘messy’ requiring **exploratory data analysis** to find patterns - both those that indicate interesting biological signals or suggest potential problems.





# Lets get started...

Do it Yourself!



```
RStudio  
File Edit Code View Plots Session Project Build Tools Help  
Go to file/function Project: (None)  
Console /media/mamut/home/slars/workspace/bio3d/ver_devel/bio3d/demo/  
R version 3.0.2 (2013-09-25) -- "Frisbee Sailing"  
Copyright (C) 2013 The R Foundation for Statistical Computing  
Platform: x86_64-redhat-linux-gnu (64-bit)  
  
R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.  
  
Natural language support but running in an English locale  
  
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.  
  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> library(bio3d)  
> |
```

Do it Yourself!

# Demo 1

<https://github.com/bioboot/demo2-github>

# Furthering your R/Bio3D knowledge...

Bio3D's: **Tutorials and vignettes** <http://thegrantlab.org/bio3d/>

Hadley Wickham's: **Advanced R** <http://adv-r.had.co.nz>

Joseph Adler's: **R in a Nutshell** <http://tinyurl.com/rbionutshell>

StackOverflow, Coursera, R-Bloggers