RBMRB

Kumaran Baskaran

July 29, 2016

RBMRB

RBMRB is a library to fetch chemical shift data directly from BMRB into R environment as a data frame in R. This facilitates access to BMRB data for statistical analysis and data visualization. It is using the BMRB-API in the back end to fetch the data from BMRB database.

Installation

Requirements

RBMRB library is developed and tested in R version 3.3. It requires the following R packages

data.table to import data from BMRB web server **httr** to format the imported data into a data frame in R. **rjson** to deal with BMRB-API **ggplot2** to simulate spectra **plotly** for interactive graphics in simulated spectra

Users should ensure the proper installtion of above packages, before proceeding to RMBRM insallation.

Here is the instruction to install those packages. Open your R and use the following command

Once the necessary packages has been installed, the source file can be downloaded from GitHub. Source package can be downloaded from this link

After downloading use the following command to install RBMRB

```
install.packages("~/Downloads/RBMRB_1.0.tar.gz",repos=NULL,type="source")
```

Note: provide the correct path to the downloaded file.

Usage

RBMRB can be used in a similar way like any other library in R.

```
library(RBMRB)
```

To fetch chemical shift table of a specific entry

```
df<-fetch_entry_chemical_shifts(15060)
```

Imported data frame contains following columns

```
colnames(df)
```

```
##  [1] "ID"                          "Assembly_atom_ID"
##  [3] "Entity_assembly_ID"          "Entity_ID"
##  [5] "Comp_index_ID"               "Seq_ID"
##  [7] "Comp_ID"                     "Atom_ID"
##  [9] "Atom_type"                   "Atom_isotope_number"
## [11] "Val"                         "Val_err"
## [13] "Assign_fig_of_merit"         "Ambiguity_code"
## [15] "Occupancy"                   "Resonance_ID"
## [17] "Auth_entity_assembly_ID"     "Auth_asym_ID"
## [19] "Auth_seq_ID"                 "Auth_comp_ID"
## [21] "Auth_atom_ID"                "Details"
## [23] "Entry_ID"                    "Assigned_chem_shift_list_ID"
```

Its also possible to load multiple entries as a single data frame

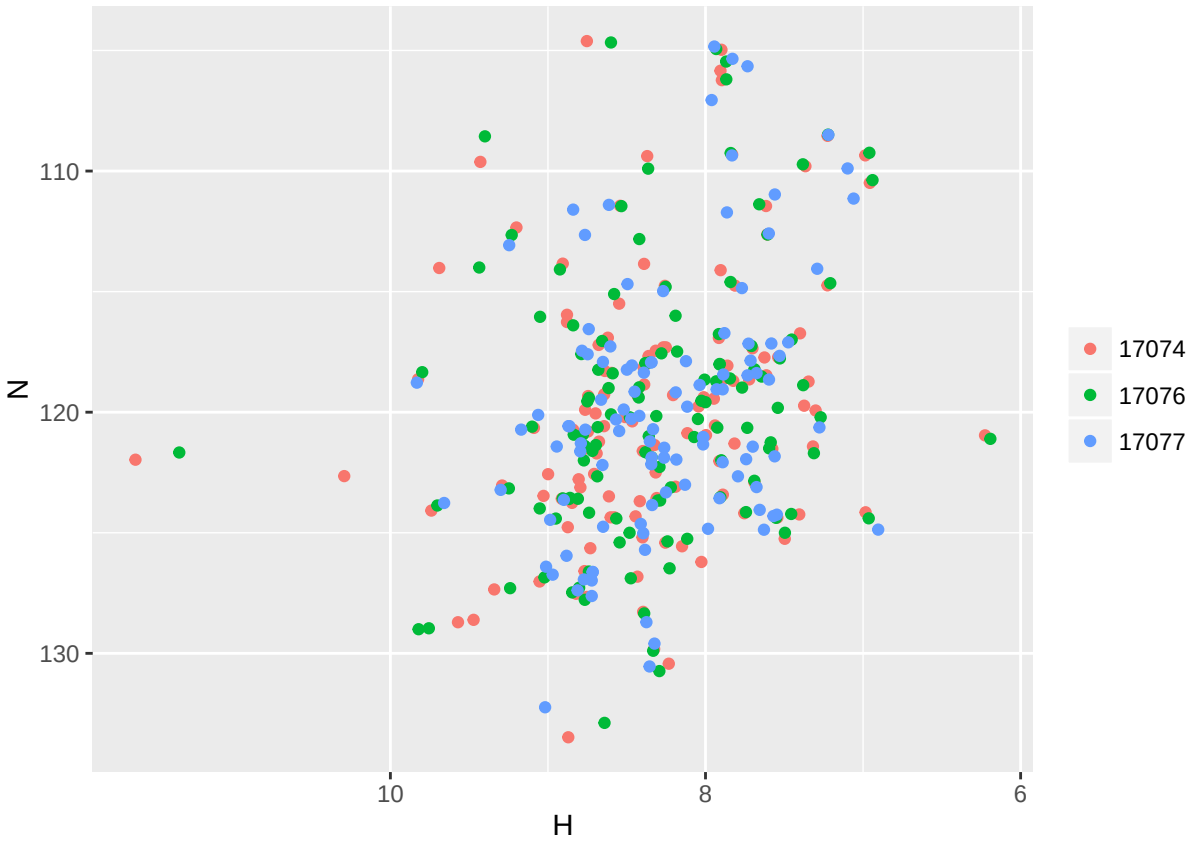
```
df<-fetch_entry_chemical_shifts(c(17074,17076,17077))
```

These data frames can be reformatted using the function **convert__cs__to__n15hsqc** to simulate HSQC spectrum from the data frame.

```
hsqc<-convert_cs_to_n15hsqc(df)
```

Now user can use his own plotting function to visualize HSQC spectrum for example

```
library(ggplot2)
ggplot(hsqc)+
  geom_point(aes(H,N,color=as.character(Entry_ID)))+
  scale_x_reverse()+scale_y_reverse()+theme(legend.title=element_blank())
```



The entire chemical shift for a given atom can be downloaded usign the following command

```
df_cb2<-fetch_atom_chemical_shifts('CB2')
```

Once it is downloaded, then you have full chemical shift list of a given atom from all entries from the macromolecules databse.

```
colnames(df_cb2)
```

```
##  [1] "Entry_ID"                  "Entity_ID"
##  [3] "Comp_index_ID"             "Comp_ID"
##  [5] "Atom_ID"                   "Atom_type"
##  [7] "Val"                       "Val_err"
##  [9] "Ambiguity_code"            "Assigned_chem_shift_list_ID"
```

Similarly chemical shift list from metabolites database can also be fetched usign the same command

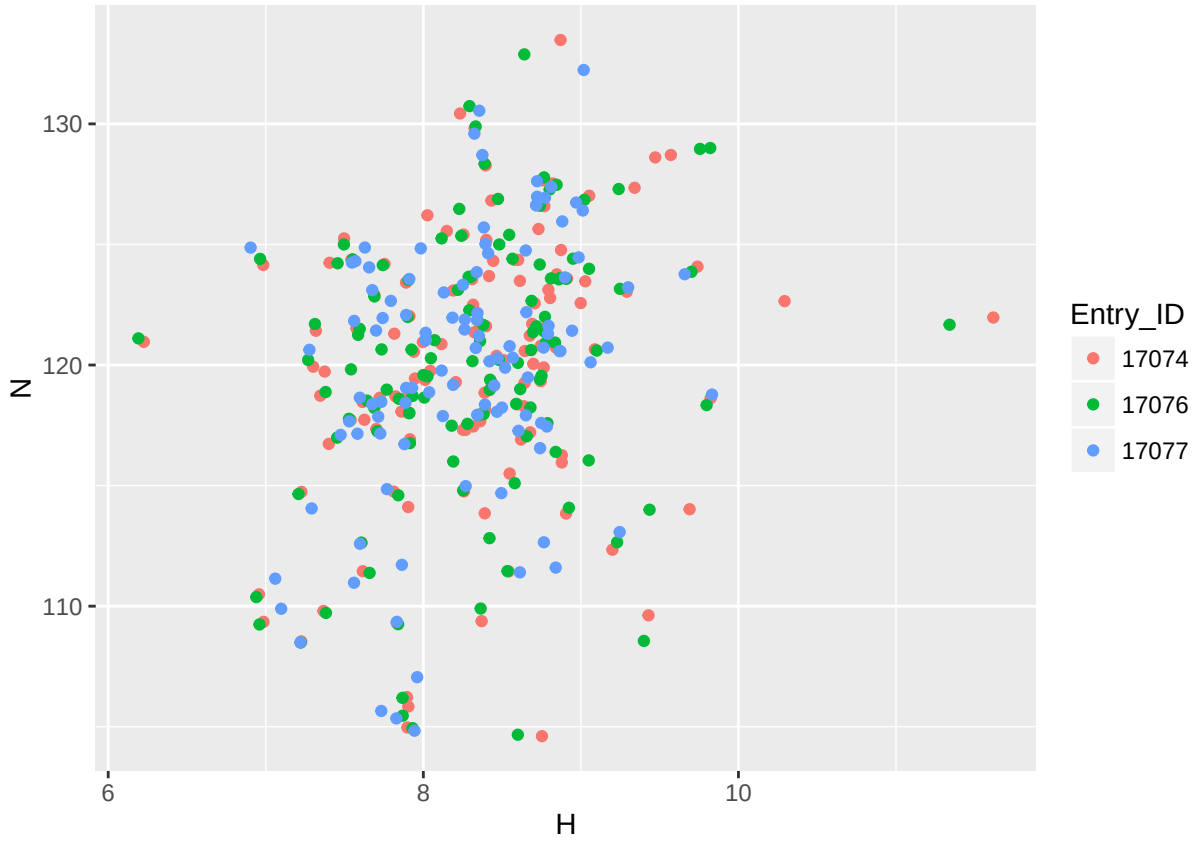
```
df_c1<-fetch_atom_chemical_shifts('C1',db="metabolomics")
```

Data Visualization

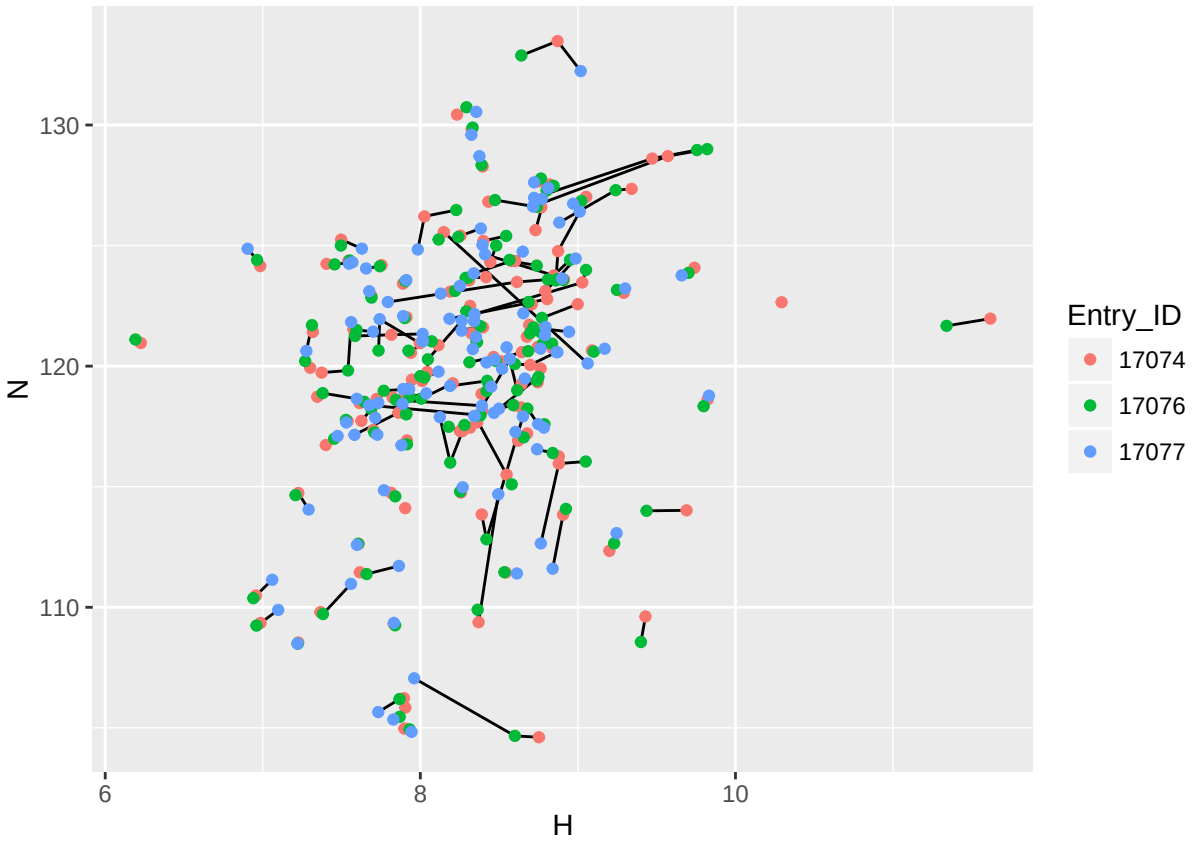
RBMRB library has a function to simulate 1H-15N HSQC spectrum directly from BMRB database.

HSQC Spectra

```
spec1<-simulate_n15hsqc(c(17074,17076,17077))
spec1
```




```
spec2<-simulate_n15hsqc(c(17074,17076,17077),type='line')
spec2
```



An interactive plot can also be created if the package **plotly** has been installed properly.

```
spec3<-simulate_n15hsqc(c(17074,17076,17077),type='line',interactive = T)
spec3
```

Tool tip will show the dtails about the peak when you mouse over and can also be zoomed using mouse.