The Pviz User Guide

Renan Sauteraud*

August 17, 2012

Contents

1 Introduction

`Pviz` is an R package inspired by and depending on `Gviz`. It introduces new types of track and extends the existing ones in order to deal with amino-acid based data.

This package keeps most of the mechanics of `Gviz`, notably the use of `DisplayParameters` and the same plotting function: `plotTracks`. Therefore, the user is invited to refer to `Gviz` help pages and vignette for more information and examples.

As with any R package, it should first be loaded in the session

```
> library(Pviz)
```

2 Gviz tracks

`Pviz` extends and uses the most common classes of `Gviz` to make them easier to use with amino acid data. We removed the requirement for a genome and a chromosome when creating these tracks. Moreover, they support the functions defined in Pviz.

*rsautera@fhcrc.org

2.1 ATrack

`ATrack` extends Gviz's `AnnotationTrack` and behave the same way. However, it does not require to specify a chromosome and a genome. Please refer to `Gviz` documentation for more details about `AnnotationTrack` and the available `DisplayParameters`.

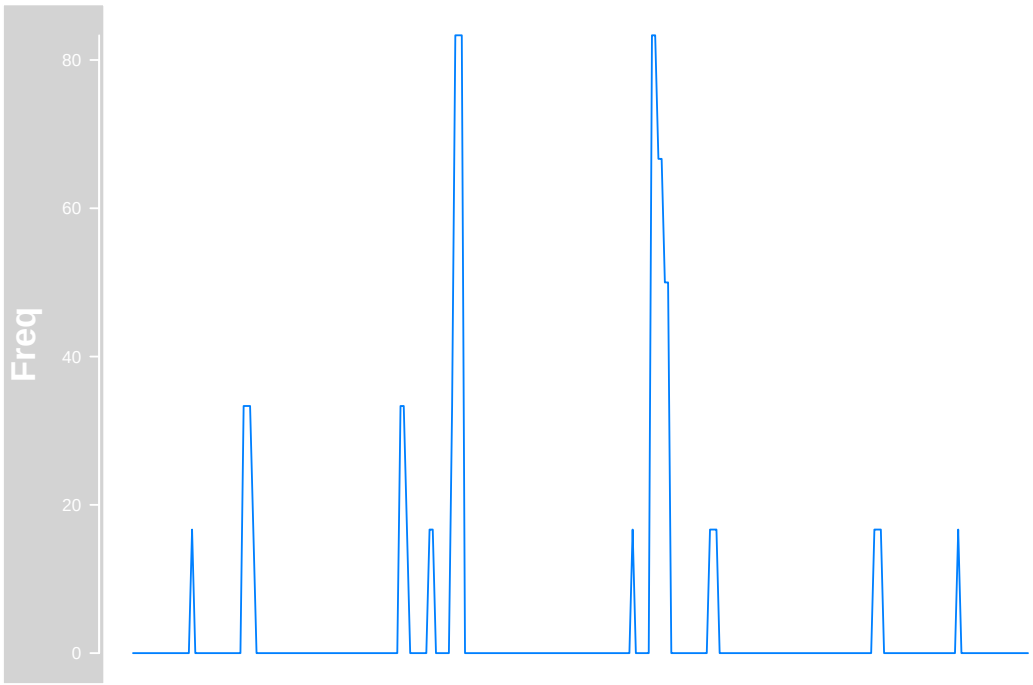
```
> at <- ATrack(start = c(250, 480), end = c(320, 520), name = "Annotations")
> plotTracks(at, from = 1, to = 600)
```


2.2 DTrack

Naturally `DTrack` extends Givz's `DataTrack`. Here again, please refer to `Gviz` documentation for details on how to use `DataTrack`.

Some example data are available in this package. Expression frequency for peptide in hxb2 envelope.

```
> data(PvizVignette)
> dt <- DTrack(data = freqEx, start = posEx, width = 15, name = "Freq")
> plotTracks(dt, from = 1, to = 850, type = "l")
```



3 Pviz new track types

`Pviz` introduce some new track types to deal with amino-acid based data. The new tracks look can be modified using the `DisplayParameters` and will most of the time offer the same options as the ones available for `Gviz` tracks.

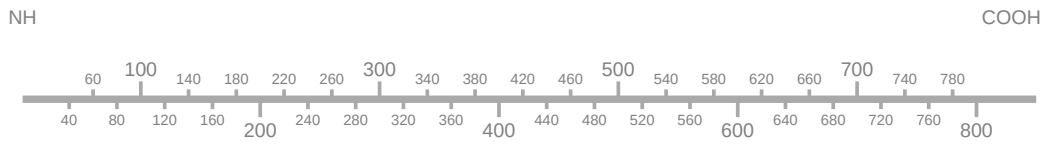
3.1 ProteinAxisTrack

This track acts as a replacement for the `GenomeAxisTrack`. I comes with the same coloration, transparency and other customization options but loses the DNA representation for a simple segment.

```
> pat <- ProteinAxisTrack()
> plotTracks(pat, from = 1, to = 850)
```

Just like in `GenomeAxisTrack`, it is possible to use littleTicks to get a more precise scale. Moreover, because `Pviz`, has been made to deal with peptides and protein, the option addNc can display indicators for N-term and C-term ends on the axis.

```
> pat <- ProteinAxisTrack(addNC = TRUE, littleTicks = TRUE)
> plotTracks(pat, from = 1, to = 850)
```



3.2 SequenceTrack

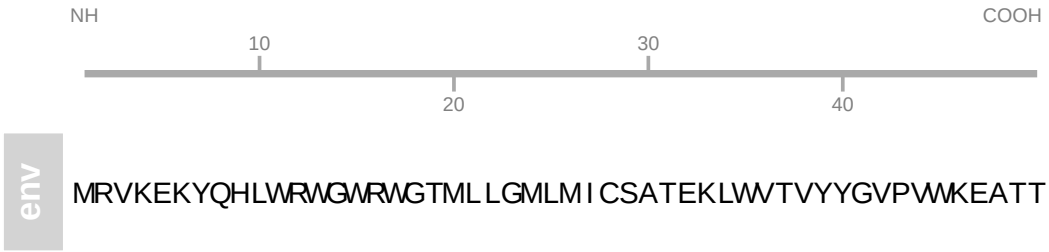
This new track simply displays a selected sequence. It can takes both `AAstring` or regular `string` and thus, can be used for both AA and DNA sequences.

Note that the first amino acid of the sequence should correspond to the first position of any other element you choose to display at the same time.

The previously loaded dataset also contains an `AAstring` object for the sequence of the envelope of hxb2.

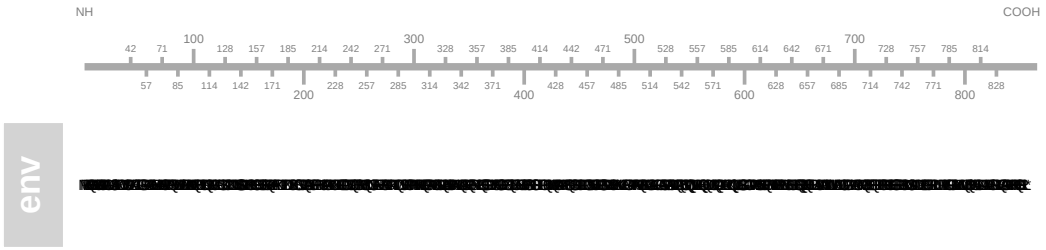
```
> seqEx

  857-letter "AAString" instance
seq: MRVKEKYQHLWRWGWRWGTMLLGMLMICSATEKLWV...VAEGTDRVIEVVQGACRAIRHIPRRIRQGLERILL*
```

```
> st <- SequenceTrack(sequence = seqEx, name = "env")
> plotTracks(trackList = c(pat, st), from = 1, to = 50)
```



Note that if the plotting ranges becomes too wide, the letters will stack and it may lead to an unreadable output. The default parameters usually allows to plot up to 100-150 characters depending on your graphic window. The user can modify the character size via the DisplayParameters system when creating the track with the cex argument. Here is an example of bad plotting.

```
> plotTracks(trackList = c(pat, st), cex = 0.5)
```

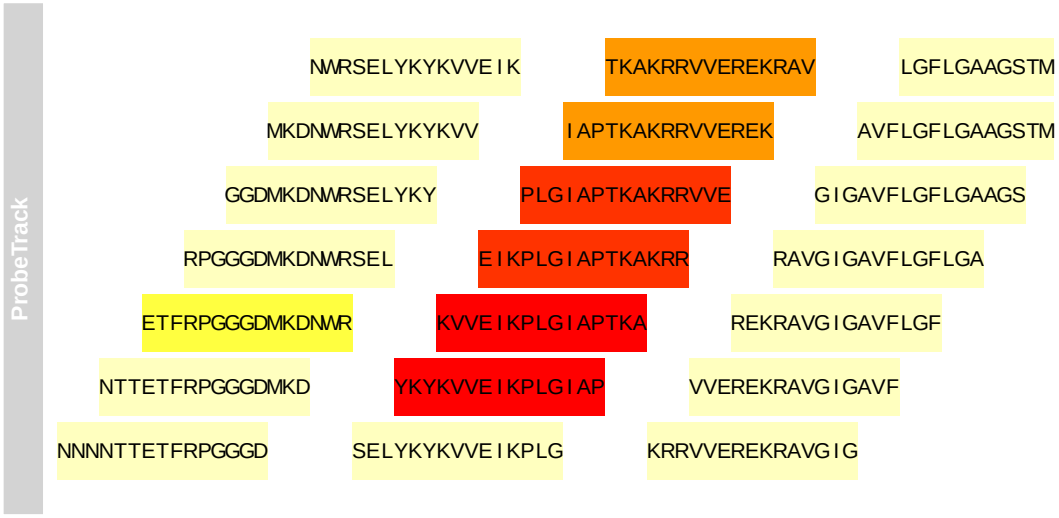


Althought the character expansion has been set to less than 1. The ranges are still too wide for a correct display.

3.3 ProbeTrack

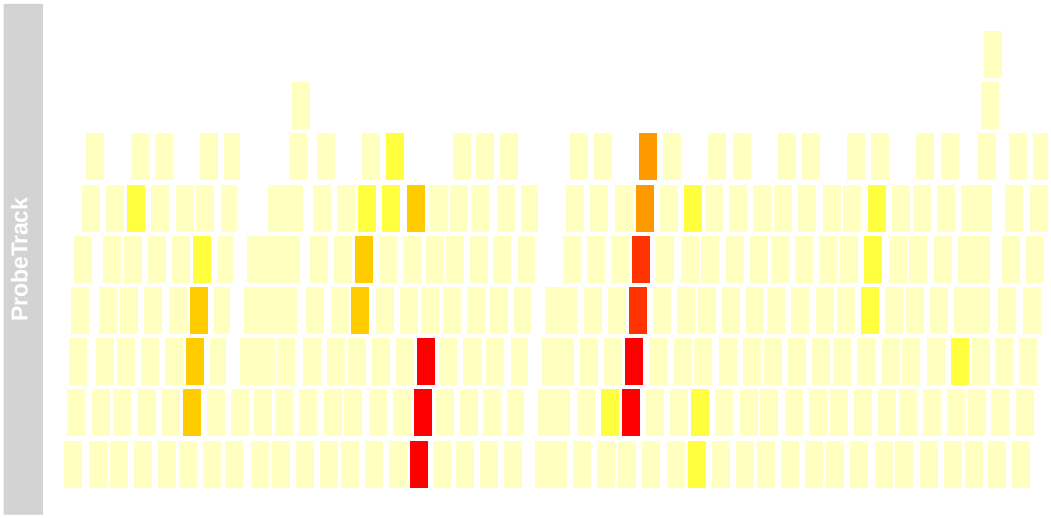
This track is designed to display peptide microarray data. It draws each peptide relative to its position in the sequence and enclose them in rectangles colored depending on their intensity. To create this track, the sequence of the peptides, their intensity or frequency and their starting position have to be passed as arguments.

```
> pt <- ProbeTrack(pepEx, freqEx, posEx)
> plotTracks(pt, from = 460, to = 530)
```



Unlike in `SequenceTrack`, the size of the characters in each peptide sequence depends on the plotting ranges (the user can still choose to change the size manually) and if the ranges become too wide, the characters will appear as dots or completely disappear instead of stacking on top of each other. While it loses the sequence information, it might be relevant to locate regions where peptides have high intensity/frequency.

```
> plotTracks(pt)
```



For a more explicit display, a legend has been implemented for this track and can be called during track creation or in the plotting function. The legend displays the scale of intensities.

```
> plotTracks(pt, from = 460, to = 530, legend = TRUE)
```

4 Example of plot

Naturally, the interest of `Pviz`, just like its parent `Gviz` is the display of multiple tracks at once. Here is an example of what `Pviz` can render, using the tracks previously created.

```
> pt <- ProbeTrack(pepEx, freqEx, posEx, cex = 0.8)
> plotTracks(trackList = c(pat, st, at, pt, dt), from = 460, to = 530,
+     type = "l", legend = TRUE)
```