

GB



[ology@github](mailto:ology@github)

Search this website .

# Bach Chorale Diversity

2017-08-07 BY GENE

What is the relative statistical diversity of Bach chorale harmonisations?

Ever since I stumbled upon this [Bach Choral Harmony Data Set](#) I've been wanting ways to analyze it!

So I set out to do just that today with the [Shannon Diversity Index](#) with this handy perl code:

```
#!/usr/bin/env perl
use strict;
use warnings;

use Text::CSV;
use Statistics::Diversity::Shannon;
```

First off (above), is the standard perl preamble followed by loading of the modules we will use. Next, the data file is read in and parsed into an index of the interesting bits:

```
my $file = shift || die "Usage: perl $0 ~/Documents/data/jsbach_chorals_har

my $index = read_bach($file);
```

This routine, `read_bach()`, takes a raw comma separated value file of note, chord and bass structure, line by line and finds how many times a note group is seen, for each tune.

```
sub read_bach {
    my ($file) = @_ ;

    my $csv = Text::CSV->new ( { binary => 1 } )
        or die 'Cannot use CSV: ' . Text::CSV->error_diag ();

    open my $fh, "<:encoding(utf8)", $file
        or die "Can't read $file: $!";

    my %index;

    while ( my $row = $csv->getline($fh) ) {
        ( my $id = $row->[0] ) =~ s/\s*//g;

        # Columns 2-13 hold the note information
        my $notes = '';
        for my $note ( 2 .. 13 ) {
            $notes .= $row->[$note] eq 'YES' ? 1 : 0;
        }

        $index{$id}{$notes}++;
    }

    $csv->eof or $csv->error_diag();
    close $fh;

    return \%index;
}
```

After reading and parsing the Bach data, all that remains is to compute the Shannon diversity index for each tune.

```
for my $id ( keys %$index ) {
    my @data = values %{ $index->{$id} };

    my $d = Statistics::Diversity::Shannon->new( data => \@data );

    $index->{$id}{shannon} = [ $d->index, $d->evenness ];
}
```

```
}
```

This above code considers the values of the index ids – that is the number of times a note group was seen in each tune. This is handed to the stats module to compute the diversity.

And what are the earth shattering conclusions??

Well, the complete set of 60 tunes is more diverse than any individual tune – that makes perfect sense. Total index = 4.53 and evenness = 0.78.

The individual tunes index summary statistics are these:

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
2.358	2.862	3.012	3.011	3.187	3.504

Ok. My inner geek is satisfied for the moment...

---

... Of course, the Inner-geek is never satisfied without visualization. After searching a bit, I found the Whittaker [Rank abundance curve](#). Naturally I had to make one with this Bach data.

As before, we invoke perl and read-in the Bach raw data file – as a parsed index of how many times each note group has been seen:

```
#!/usr/bin/env perl
use strict;
use warnings;

use Text::CSV;
use List::Util qw(sum0);

my $file = shift || die "Usage: perl $0 ~/Documents/data/jsbach_chorals_har

my $index = read_bach($file);
```

Next up, instead of calculating the statistical diversity, we set the proportional frequencies of the data:

```
# Set the proportional frequencies
my $total = sum0 values %$index;

for my $species ( keys %$index ) {
    $index->{$species} /= $total;
}
}
```

And output those totals in a reverse sorted list for R to consume:

```
for my $value ( sort { $b <=> $a } values %$index ) {
    print "$value\n";
}
}
```

That is, run this in the shell (but with your own jsbach data file of course):

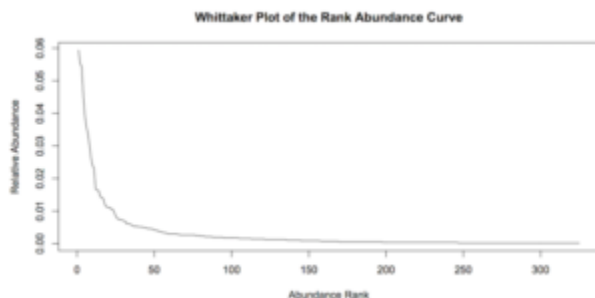
```
$ perl whittaker ~/Documents/data/jsbach_chorals_harmony/jsbach_chorals_har
```

Then open up R and:

```
df <- read.table('whittaker.txt')

plot(df$V1, type='l', xlab='Abundance Rank', ylab='Relative Abundance', mai
```

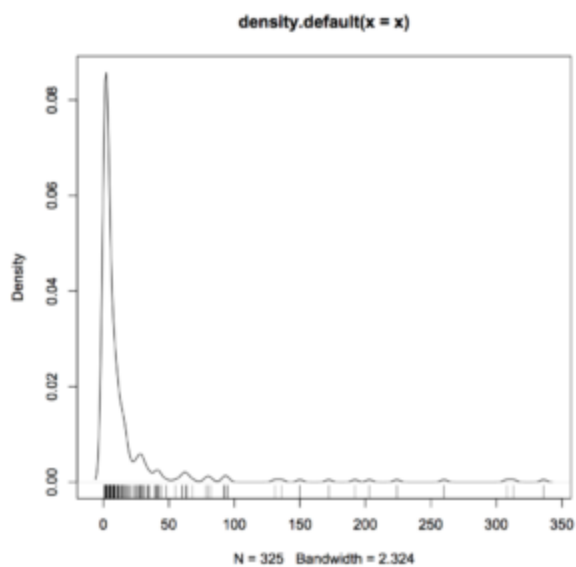
And voilà:



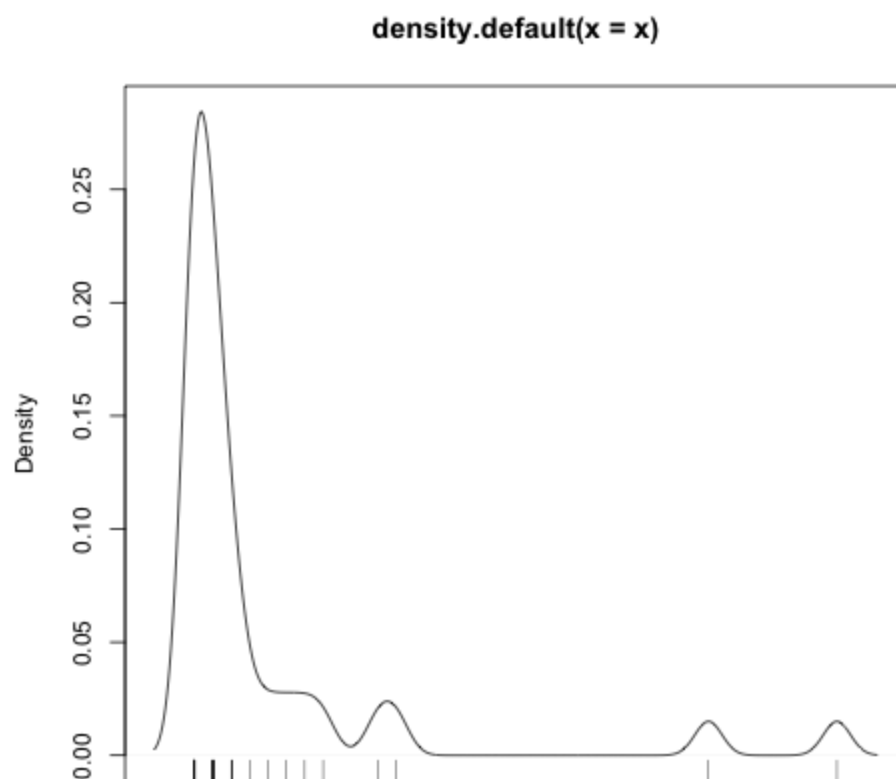
Again, Earth shattering!

We can see that Bach definitely uses a lot of note groups (i.e. chords and arpeggios) – more than 300! But the vast majority of these are in the top 50.

Here is the density plot of all Bach chord groups:



And here is an animated version of the densities of all the 60 chorales:



FILED UNDER: DATA, MUSIC, SOFTWARE  
TAGGED WITH: BACH, HARMONIZATION, PERL, R, WHITTAKER PLOT  
N = 36 Bandwidth = 0.738

Epistemologist-at-large

[^ Top](#)