# Mapping FS space

```
library(dplyr)
```

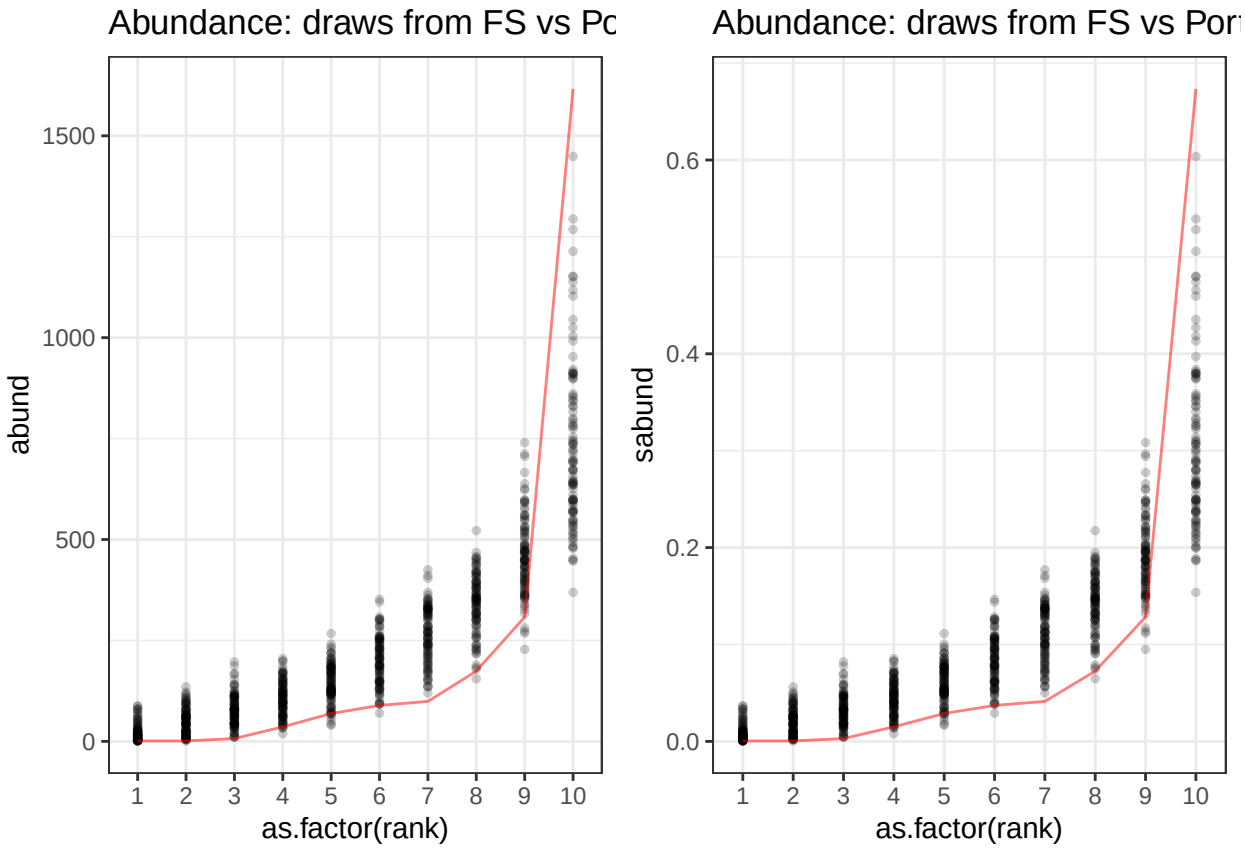
```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

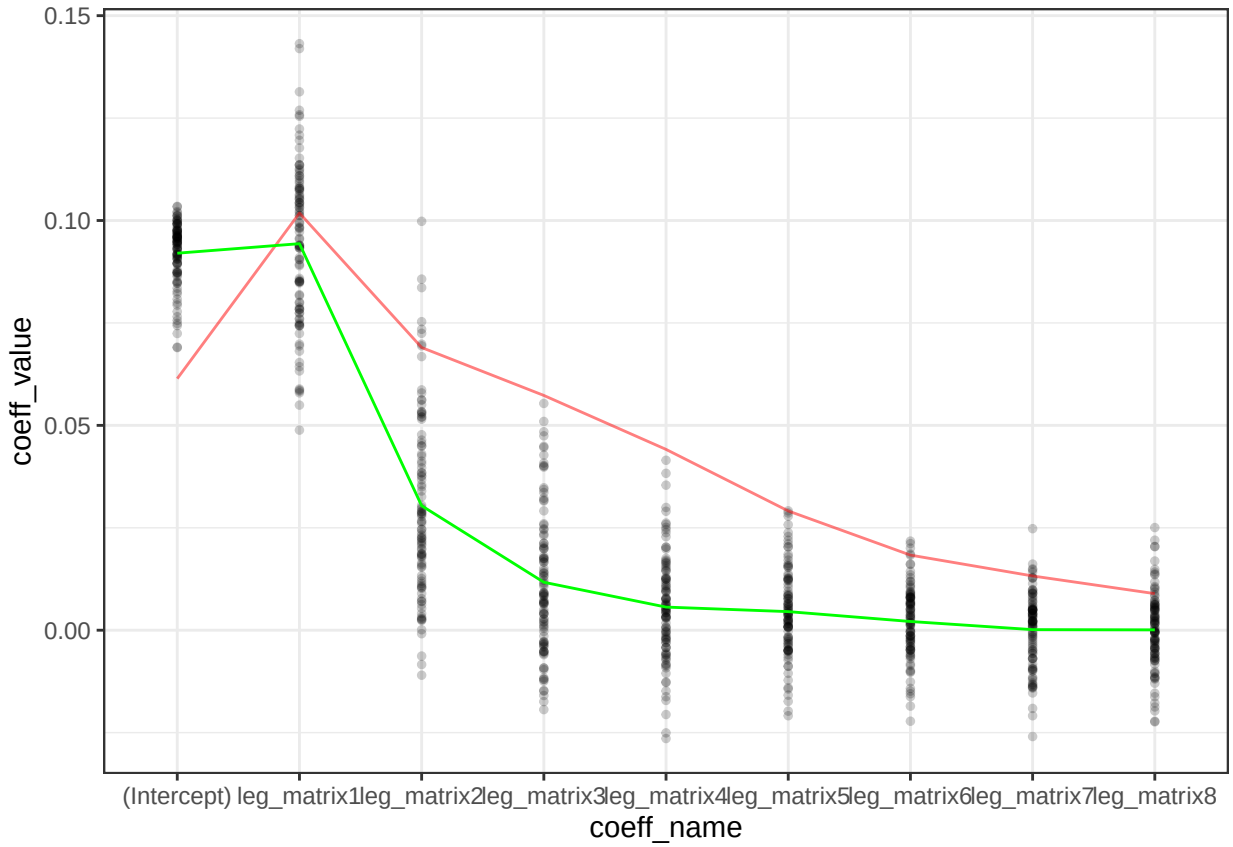
```
library(scads)
library(ggplot2)
small = T
knitr::opts_chunk$set(echo = FALSE)
```

```
## Loading in data version 1.127.0
```



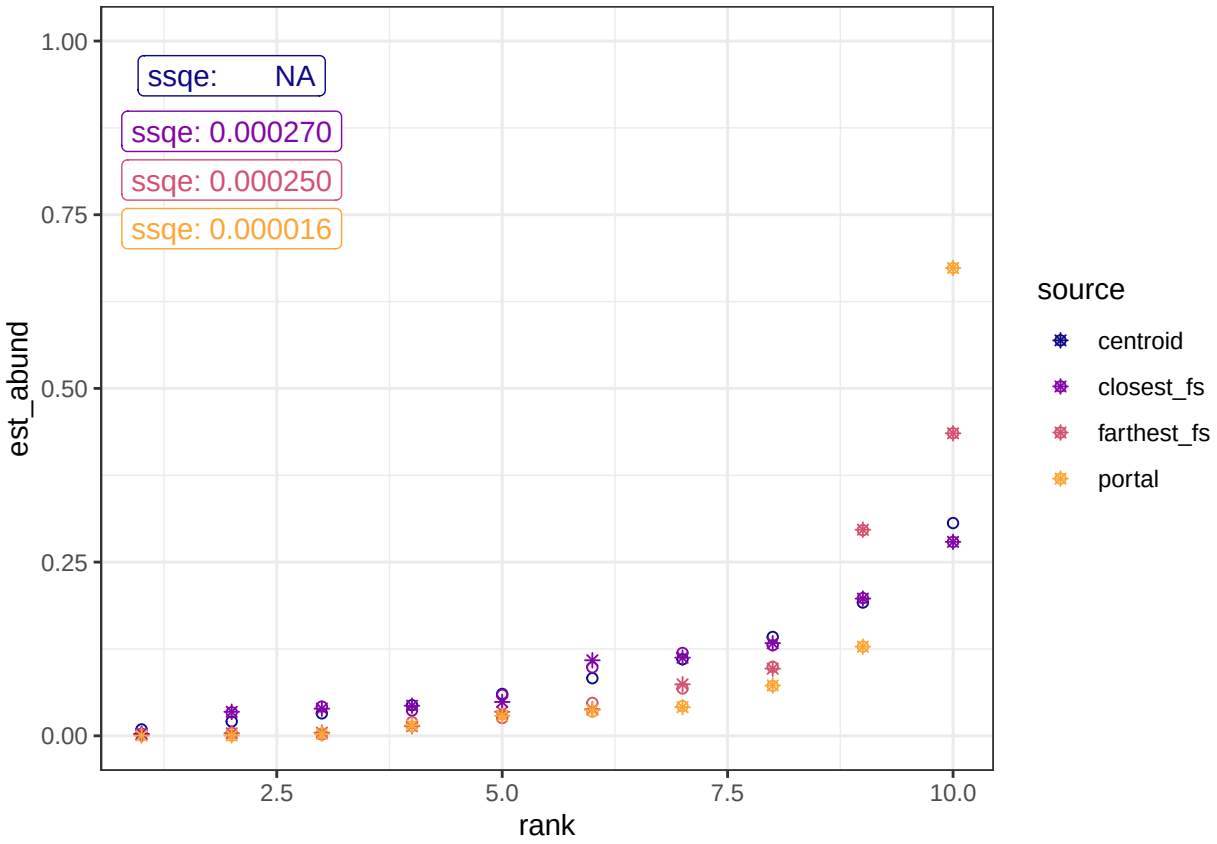

The rescaled vectors (on the right) are what go into Legendre approximation. In this case, these plots should look identical, because all the draws from the feasible set have the same number of individuals as the Portal vector. I have at other times compared SADs without the total abundance constraint, where the rescaled plots could look quite different.

```
## Loading required package: polynom
```

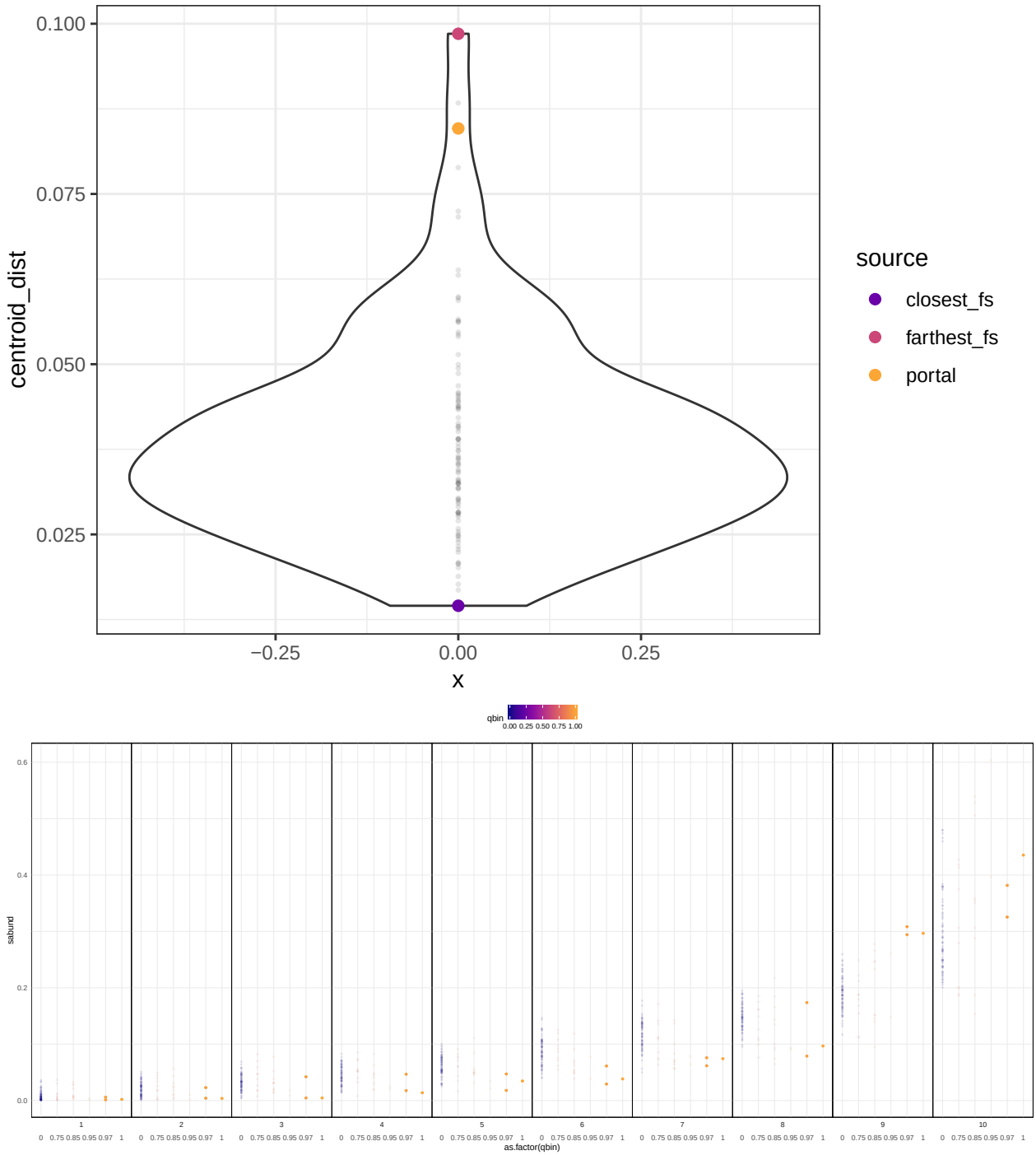


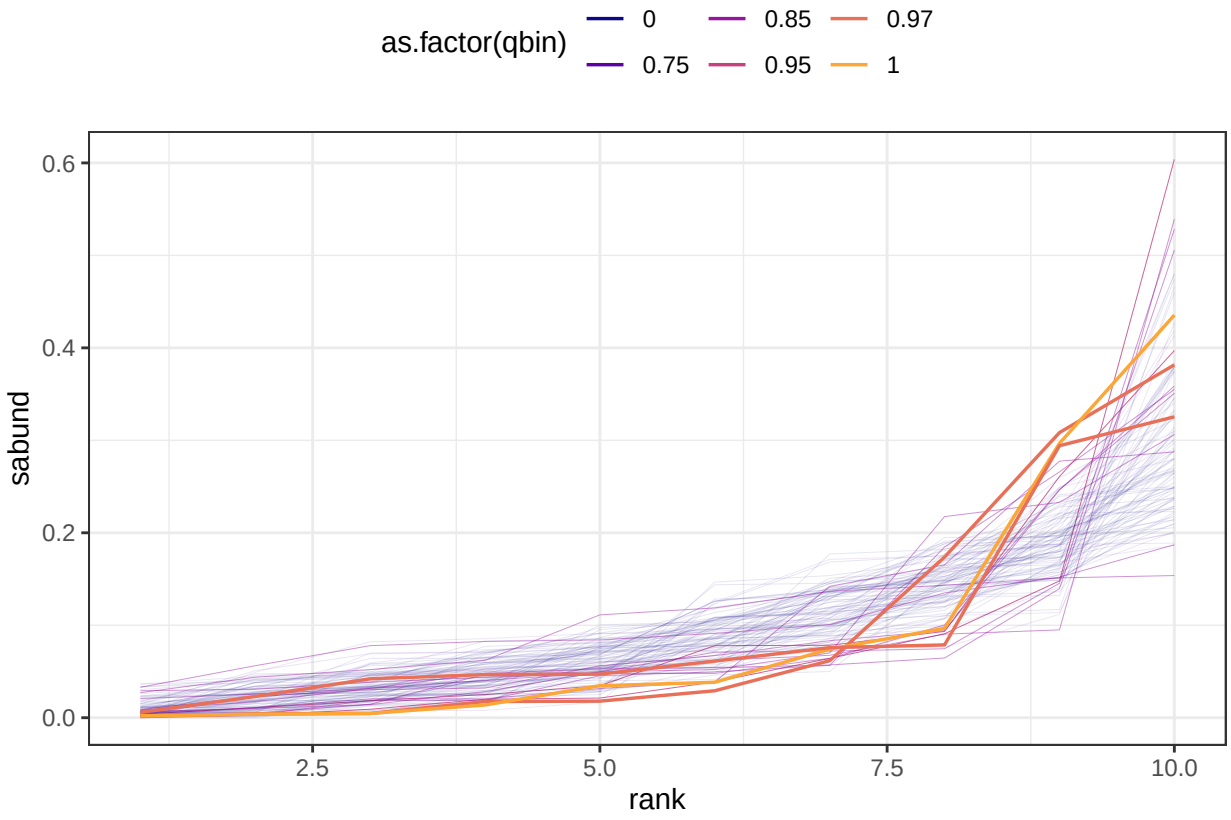

The green line marks the centroid, and the red line marks the Portal estimation values.

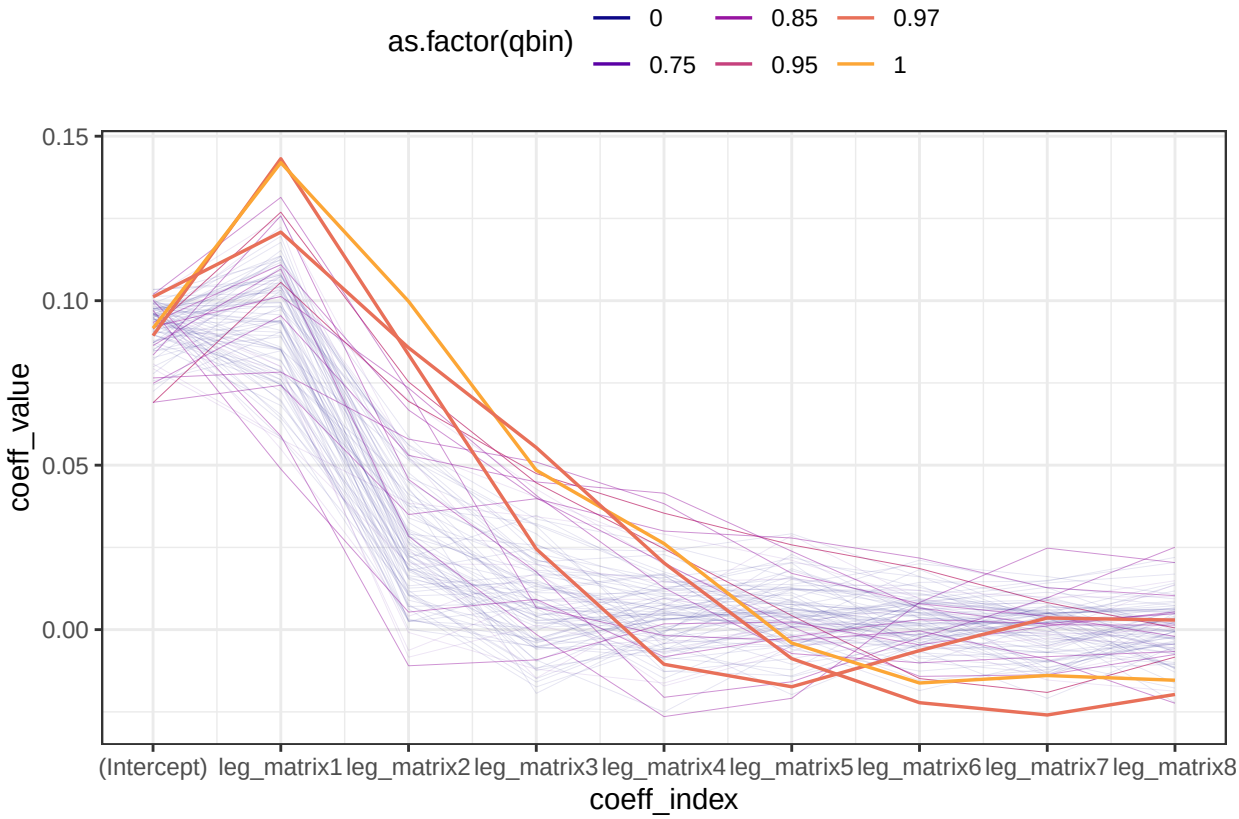
```
## Warning: Removed 1 rows containing missing values (geom_point).
```




Here we are reconstructing the scaled SAD from the approximated coefficients. The hollow points are estimates from approximation with 8 polynomials, and the stars mark the true values. Using 8 polynomials for a 10-species vector gives us extremely low SSQE. There is no real value for the centroid, but `closest_fs` is the element of the feasible set with the lowest euclidean distance between its coefficients and the centroid coefficients.


centroid_dist
0.100
0.075
0.050
0.025
−0.25
0.00
0.25
x
source
closest_fs
farthest_fs
portal
qbin
0.00 0.25 0.50 0.75 1.00
sabund
0.6
0.4
0.2
0.0
1
2
3
4
5
6
7
8
9
10
0 0.75 0.85 0.95 0.97 1
as.factor(qbin)


as.factor(qbin)
0
0.75
0.85
0.95
0.97
1
sabund
0.0
0.2
0.4
0.6
2.5
5.0
7.5
10.0
rank


as.factor(qbin)
0
0.75
0.85
0.95
0.97
1
coeff_value
0.00
0.05
0.10
0.15
(Intercept)
leg_matrix1
leg_matrix2
leg_matrix3
leg_matrix4
leg_matrix5
leg_matrix6
leg_matrix7
leg_matrix8
coeff_index