### Diversity Metrics
- **Alpha diversity** (α)
  - 20 species in A
  - 15 species in B
- **Beta diversity** (β)
  - 8 species in both A and B
- **Gamma diversity** (γ)
  - 100 species in the region as a whole

### Diversity Indices
- **Species richness** (S)
- **Simpson’s Index** (H)
- **Simpson’s Index** (D)
  - Where n = individuals of species
  - S = number of species
  - N = total number of individuals
  - P = relative abundance of each species
  - Use raw abundance data

### Diversity function (vegan package)
- **diversity** (community_matrix, index, MARGIN)
  - Default margin is 1, samples in rows, species in columns
  - Index can be "shannon", "simpson", or "invsimpson"
  - Returns a vector of diversity measures

- **diversity** (Brier_Ck, index = "shannon", MARGIN = 1)

- **Shannon’s index** by hand:
  - Brier_p <- decostand(Brier_Ck, MARGIN = 1, method = "total")
  - Brier_p_inv <- Brier_p * log(Brier_p)
  - apply(Brier_p_inv, 1, sum, na.rm = T)^(-1)

### Typical alpha diversity metrics are summaries by site
- **Species richness**
- **Species diversity**
Measures of evenness

\[ J = \frac{H}{H_{\text{max}}} = 1 - \sum_{i=1}^S p_i \ln p_i \]

- Measure of the evenness of species abundances within the community.
- Maximum = \( H_{\text{max}} \) if all species abundances the same
- Equitability is a percentage of the max

Species richness

\[ D = 1 - \sum_{i=1}^S p_i^2 \]

- Simpson’s index by hand:
  \[ \text{Brier}_{\text{p}} < \text{decostand(Brier_Ck, MARGIN=1, method="total"}) \]
  \[ \text{Brier}_{\text{p}^2} < \text{1-apply(Brier}_{\text{p}^2,1,sum, na.rm=T)} \]

Function \( \text{specnumber(community_matrix)} \)

- \( \text{specnumber(Brier_Ck)} \)

**R functions**

- Function **diversityresult**
  - Calculates diversity indices (richness, abundance, Shannon, Simpson) and evenness indices.
  - **diversityresult(matrix, index, method)**
    - Index – type of diversity index to use
    - Method – calculate for all samples pooled, mean of all samples, individual samples, or jackknife
    - Factor – factor dividing samples into groups for comparison
  - Default is to pool all samples together
  - Will also do bootstrap and jackknife estimates of diversity
  - Will also work with factors that divide samples into groups

- Shannon diversity by site
  - **diversityresult(Brier_Ck, index="Shannon", method="each site")**

- Shannon diversity pooling sites
  - **diversityresult(Brier_Ck, index="Shannon", method="pooled")**

- Mean diversity across sites
  - **diversityresult(Brier_Ck, index="Shannon", method="mean")**
  - Same as taking the mean from our earlier vector
    - **mean(Brier_Shannon)**

- Gamma diversity (chao) estimator
  - **diversityresult(Brier_Ck, index="chao", method="pooled")**
Comparing Diversity with Factors

- Factors may be used to describe sample differences
- Shannons for sites with factor “year” level “a”
- Function `diversitycomp` will get diversity for all levels of a factor
  - Note – specify factor as `factor1`
  - Specify `factor1` and `factor2`

Species Accumulation and Rarefaction

- Curves display the rate of new species addition with additional sampling.
- Alpha, beta and gamma diversity
- Diverse ecosystems (high gamma and high beta diversity) will display rapid increase to higher plateau. Less diverse: slow increase to lower plateau.

**Function specaccum**

- `specaccum(matrix, method)`

**Methods:**

- Accumulate species by:
  - Adding samples in the order collected (“collector”)
  - Adding samples in random order (“random”)
  - Calculating expected (mean) number for each sample size (“exact”)
  - Follow methods of Coleman 1982 (“coleman”)
  - Permuting individuals instead of samples (“rarefaction”)

- “collector” does not provide error estimates.
- “exact” provides a mean and standard deviation for observed data
- “random” provides mean and standard deviation for permuted data
Randomizations, bootstraps, jacknifes, Monte Carlo, MCMC

- Function `sample` can be used to randomly sample any data
  - `sample(thing to sample, number of samples, replace)`
- **Bootstrapping** – estimating some parameter by randomly sampling with replacement
  - See also functions in `permute` package

- In our sample community dataset:
  - `mean(Brier_Ck_matrix) = 14.261`
  - `sample(Brier_Ck_matrix, 100, replace=TRUE)` # pick 100 values
  - `mean(sample(Brier_Ck_matrix, 10000, replace=TRUE))` # 10000
  - `sample(Brier_Ck_matrix, 3000, replace=FALSE)` # sample all without replacement

---

Species Accumulation and Rarefaction

- **Rarefaction** – most commonly used technique for estimating diversity
- Pool individuals within a dataset and resample individuals and not individual cells within the matrix.

- Use the same `specaccum` function (method="rarefaction").

- Rarefaction requires integer (count) data, untransformed. Other methods will work with log or proportional transformed data.

- Number of individuals sampled is set as the mean number of individuals per sample in the dataset.

- **Species Accumulation and Rarefaction**
  - Rarefaction curve for Brier Creek community data using `specaccum`, method="rarefaction" which set the number of individuals per sample to 428 (mean number per sample).

- The `specaccum` function uses the function `rrarefy` to do the resampling. You can use `rrarefy` directly if you want more control.

- Function `rrarefy(community, sample)`
  - Provide the community matrix and how many individuals to sample and it will return a permuted sample.

- To duplicate what `specaccum` did:
  - Pool all individuals into one vector
    - `pooled=apply(Brier_Ck,2,sum)`
  - Rarefy all samples by randomly selecting 426 individuals
    - `rrarefy(pooled, sample=428)`
  - Repeat...
One common use for rarefaction is to control for sampling effort (or efficiency) in estimating diversity.

In the sample community, the number of individuals per sample ranges from 2 to 2554.

Diversity and the number of individuals is clearly correlated, what is sample diversity if we control for this?

\[
\text{rarefied}_{100} = \text{rarefy}(\text{Brier}_\text{Ch}, 100, \text{se}=T)
\]

\[
t(\text{rarefied}_{100})
\]

- \text{WJM7} 7.307387 7.14E-01
- \text{WJM8} 7.809496 4.17E-01
- \text{WJM9} 5.000000 0.00E+00
- \text{WJM10} 7.303897 7.14E-01
- \text{WJM11} 9.727267 4.93E-01
- \text{WJM12} 9.764172 1.25E+00
- \text{WJM73} 7.665996 5.25E-01
- \text{WJM74} 8.584473 5.52E-01
- \text{WJM75} 10.32027 7.08E-01
- \text{WJM76} 12.07872 8.08E-01
- \text{WJM77} 7.809496 4.17E-01
- \text{WJM78} 5.000000 0.00E+00
- \text{WJM79} 7.303897 7.14E-01
- \text{WJM80} 9.727267 4.93E-01
- \text{WJM81} 9.764172 1.25E+00
- \text{WJM82} 7.665996 5.25E-01
- \text{WJM83} 8.584473 5.52E-01
- \text{WJM84} 10.32027 7.08E-01
- \text{WJM85} 12.07872 8.08E-01
...

How many individuals should you sample?
- The number should be less than the number in your least abundant sample

Note that sampling this number results in no error for the sample with the fewest individuals (all individuals are sampled in each permutation)

This is where you would consider eliminating samples with fewer individuals.