

Transfer functions — Weighted averaging and all that

Gavin Simpson
(gavin.simpson@ucl.ac.uk)

Aarhus University

ACME 2022

Outline

- 1 Motivating Examples
- 2 Introduction
- 3 Weighted Averaging
- 4 Modern Analogue Technique
- 5 Model performance and diagnostics

Example 1: Was acid rain to blame for acid lakes?

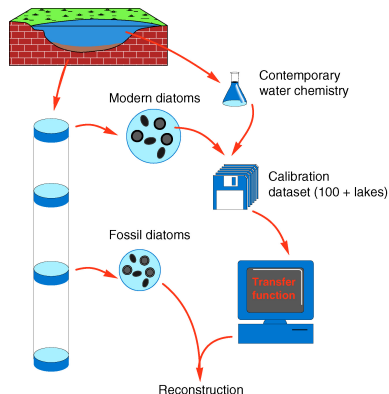
- In the 1970s and early 1980s there was a great deal of concern about acid lakes and rivers in northern Europe
- Driven mainly by losses of Salmon in Scandinavian rivers, this was a major political hot potato
- A vast amount of money was expended to determine the cause of the acidification — was it due to acid emissions from power stations or some other cause?
- Palaeolimnological data provided conclusive proof that acid deposition was the cause
- In Europe, the Surface Waters Acidification Project (SWAP) was a major contributor to the debate
- Diatoms collected from 167 lakes across UK, Norway, Sweden and associated water chemistry
- Can we predict lake-water pH from the diatom species assemblages?
- Apply to diatoms counted from a sediment core from the Round Loch of Glenhead (RLGH) covering most of the Holocene

Example 2: Reconstructing past sea surface temperatures

- Sea surface temperatures are related to global air temperatures
- An important arm of palaeoceanography is involved in reconstructing past climates from various proxies
- These past climates tell use how the world responded to previous climatic shifts and provide targets for climate modellers to try to model
- The data set here is the Imbrie & Kipp data set — the data set that started it all!
- 61 core-top samples from ocean cores, mainly from Atlantic
- 27 species of planktonic foraminifera were identified in the core-top samples
- Summer and Winter sea surface temperatures (SST) and sea water salinity values measured at each of the 61 core locations
- Applied to reconstruct SST and salinity for 110 samples from Core V12-133 from the Caribbean

Palaeoecological transfer functions

- Transfer functions
- Calibration
- Bioindication
- Aim is to predict the environment from observations on species environment
- The reverse of constrained ordination from yesterday
- ter Braak (1995) *Chemometrics and Intelligent Laboratory Systems* **28**: 165–180



Palaeoecological transfer functions

- More formally we have
 - ▶ Matrix of species abundances, Y
 - ▶ Vector of observations of an environmental variable, x
- Assume Y is some function f of the environment plus an error term

$$Y = f(x) + \varepsilon$$

- In the **classical** approach f is estimated via regression of Y on x
- Then invert f , (f^{-1}) to yield estimate of environment x_0 from fossil species assemblage y_0

$$\hat{x}_0 = f(y_0)^{-1}$$

- In all but simplest cases f^{-1} doesn't exist and must be estimated via optimisation

Palaeoecological transfer functions

- To avoid problems of inverting f , the **indirect** approach directly estimates the inverse of f , here g , from the data by regression x on Y

$$x = g(Y) + \varepsilon$$

- We do **not** believe that the species influence their environment!
- This is just a trick to avoid having to estimate f
- The predicted environment for a fossil sample y_0 is

$$\hat{x}_0 = g(y_0)$$

Assumptions of palaeoecological transfer functions

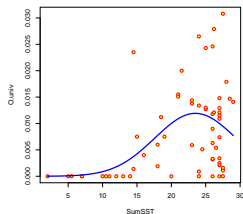
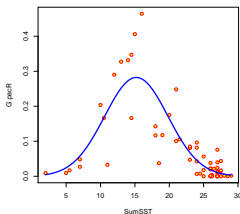
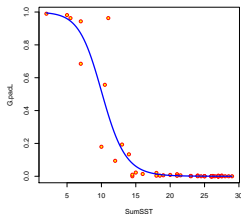
- Taxa in training set are systematically related to the environment in which they live
- Environmental variable to be reconstructed is, or is linearly related to, an ecologically important variable in the ecosystem
- Taxa in the training set are the same as in the fossil data and their ecological responses have not changed significantly over the timespan represented by the fossil assemblages
- Mathematical methods used in regression and calibration adequately model the biological responses to the environment
- Other environmental variables have negligible influence, or their joint distribution with the environmental variable of interest is the same as in the training set
- In model evaluation by cross-validation, the test data are independent of the training data — the **secret assumption** until Telford & Birks (2005)

Different types of transfer functions

- There are a large number of transfer function models
- Many motivated from chemometrics, but modified to deal with non-linear species responses
- Partial least squares (PLS) and WA-PLS
- Mutual Climate Range method
- So-called maximum likelihood method (Multivariate Gaussian logistic regression)
- Two of the most used (except WA-PLS) are
 - ▶ Weighted Averaging (WA)
 - ▶ Modern Analogue Technique (MAT)
 - ▶ These are the two techniques we will investigate today
- Large number of potential techniques from machine learning, bioinformatics, that have yet to be investigated

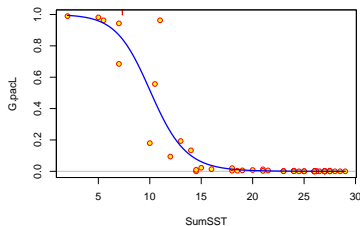
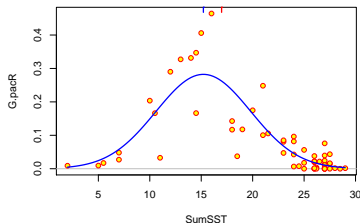
Weighted averaging

- Species don't respond in simple ways to *environmental gradients*
- Maximum likelihood method fitted Gaussian curves to each species and then numerical optimisation used to predict for fossil samples
- Computationally very intensive, especially when doing cross-validation
- Weighted averaging is an approximation to this maximum likelihood approach



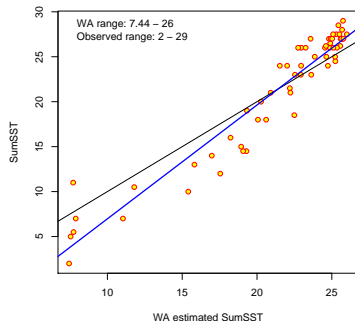
Weighted averaging

- A very simple idea
- In a lake, with a certain pH, a species with their pH optima close to the pH of the lake will tend to be the most abundant species present
- A simple estimate of the a species' pH optimum is an average of all the pH values for lakes in which that species occurs, weighted by their abundance
- An estimate of a lake's pH is the weighted average of the pH optima of all the species present, again weighted by species abundance



Deshrinking

- By taking averages twice, the range of predicted values is smaller than the observed range
- Deshrinking regressions stretch the weighted averages back out to the observed range
- Can do **inverse** or **classical** regressions
 - ▶ inverse: regress gradient values on WA's
 - ▶ classical: regress WA's on gradient values
 - ▶ Vegan also allows to just make variances equal
- Inverse and classical regression remove both bias and error, equalising variances deshrinks without adjusting the bias



WA in analogue

- **analogue** contains R code for fitting WA transfer functions and associated helper functions

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m SumSST <- imbric.env$SumSST  
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

WA — diagnostic plots

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m opar <- par(mfrow = c(1,2))
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m plot(mod)
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
```

WA — predictions I

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
  \let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
  \let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m pred <- predict(mod, imbric.f  
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
  \let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
  \let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m pred
```

WA — predictions II

Weighted Averaging Predictions

```
Call:
predict(object = mod, newdata = imbric.fos)
```

```
Deshrinking      : Inverse
Crossvalidation  : none
Tolerance DW     : No
```

Performance:

RMSEP	R2	Avg.Bias	Max.Bias
2.0188	0.9173	0.0000	-3.8155

Predictions:

0	10	20	30	40	50	60	70	80
26.8321	26.7870	26.5611	26.1722	26.1857	26.1670	25.9064	26.0574	26.2797
90	100	110	120	130	140	150	160	170
25.6723	26.1054	25.6092	25.8379	25.7696	25.7891	26.0105	25.8400	26.1986
180	190	200	210	220	230	240	250	260
26.0054	26.4729	26.4282	26.5318	26.7689	26.7812	26.8077	26.0786	26.4078
270	280	290	300	310	320	330	340	350
26.3981	26.1494	26.4148	26.2799	25.8553	26.0269	25.3974	26.0271	26.2423

WA — predictions III

360	370	380	390	400	410	420	430	440
26.3020	26.7047	26.7140	26.2727	25.4927	26.7538	26.6039	26.6019	26.1936
450	460	470	480	490	500	510	520	530
26.7939	26.7742	26.2152	25.4620	26.7682	26.8107	26.2679	25.7851	25.8562
540	550	560	570	580	590	600	610	620
25.5992	25.0000	25.3488	25.3794	25.3995	26.5347	26.1509	26.1765	26.1447
630	640	650	660	670	680	690	700	710
25.8472	26.3835	26.3507	26.0932	24.5383	25.3052	26.6331	26.3173	26.4848
720	730	740	750	760	770	780	790	800
26.0882	26.1193	26.1579	26.0043	26.3400	26.6920	26.9768	26.9926	26.8074
810	820	830	840	850	860	870	880	890
26.4448	25.4736	25.8549	26.0450	26.2881	25.6021	26.1688	25.8223	24.1910
900	910	920	930	940	950	960	970	980
24.4447	24.9817	25.4642	26.2359	26.4497	26.2772	26.1387	26.1874	25.8485
990	1000	1010	1020	1030	1040	1050	1060	1070
25.7372	25.8538	24.8725	24.1065	24.4843	24.1864	25.6200	25.1869	24.8619
1080	1090							
26.0186	25.6395							

Plotting reconstructions

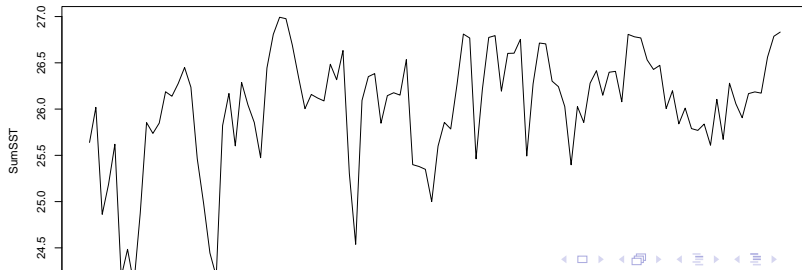
```
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
  \let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
  \let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m reconPlot(pred, use.labels = T
```



Modern Analogue Technique

- WA take a species approach to reconstruction — each species in the fossil sample that is also in the training set contributes to the reconstructed values
- MAT takes a more holistic approach — we predict on basis of similar assemblages
- In MAT, only the most similar assemblages contribute to the fitted values
- MAT is steeped in the tradition of **uniformitarianism** — **the present is the key to the past**
- We take as our prediction of the environment of the past, the (possibly weighted) average of the environment of the k sites with the most similar assemblages
- Several things to define; k , (dis)similarity
- MAT is k nearest neighbours (k -NN) regression/calibration

Measuring association — binary data

	Object j	
	+	-
Object i	+	a b
	-	c d

- Dissimilarity based on the number of species present only in i (b), or j (c), or in present in both (a), or absent in both (d).

Jaccard similarity

$$s_{ij} = \frac{a}{a + b + c}$$

Simple matching coefficient

$$s_{ij} = \frac{a + d}{a + b + c + d}$$

Jaccard dissimilarity

$$d_{ij} = \frac{b + c}{a + b + c}$$

Simple matching coefficient

$$d_{ij} = \frac{b + c}{a + b + c + d}$$

Measuring association — quantitative data

Euclidean distance

$$d_{ij} = \sqrt{\sum_{k=1}^m (x_{ik} - x_{jk})^2}$$

Manhattan distance

$$d_{ij} = \sum_{k=1}^m |x_{ik} - x_{jk}|$$

Bray-Curtis

$$d_{ij} = \frac{\sum_{k=1}^m |x_{ik} - x_{jk}|}{\sum_{k=1}^m (x_{ik} + x_{jk})}$$

Measuring association — quantitative data

- Euclidean distance dominated by large values.
- Manhattan distance less affected by large values.
- Bray-Curtis sensitive to extreme values.
- Similarity ratio (Steinhaus-Marczewski \equiv Jaccard) less dominated by extremes.
- Chord distance, used for proportional data; **signal-to-noise** measure.

Similarity ratio

$$d_{ij} = \frac{\sum_{k=1}^m x_{ik} x_{jk}}{\left(\sum_{k=1}^m x_{ik}^2 + \sum_{k=1}^m x_{jk}^2 - \sum_{k=1}^m x_{ik} x_{jk} \right)^2}$$

Chord distance

$$d_{ij} = \sqrt{\sum_{k=1}^m (\sqrt{p_{ik}} - \sqrt{p_{jk}})^2}$$

Gower's coefficient

$$s_{ij} = \frac{\sum_{k=1}^m w_{ijk} s_{ijk}}{\sum_{k=1}^m w_{ijk}}$$

- s_{ijk} is similarity between sites i and j for the k th variable.
- Weights w_{ijk} are typically 0 or 1 depending on whether the comparison is valid for variable k . Can also use variable weighting with w_{ijk} between 0 and 1.
- w_{ijk} is zero if the k th variable is missing for one or both of i or j .
- For binary variables s_{ijk} is the Jaccard coefficient.
- For categorical data s_{ijk} is 1 if i and j have same category, 0 otherwise.
- For quantitative data $s_{ijk} = (1 - |x_{ik} - x_{jk}|) / R_k$

- Once you have chosen a suitable dissimilarity coefficient, MAT begins
- We calculate the dissimilarity between each training set sample and every other
- For each site in turn, we order the training set samples in terms of increasing dissimilarity to the target training set sample
- Calculate the (weighted) average of the environment for the closest site, then the two closest sites, then the three closest sites, ... and so on
- The weights, if used, are the inverse of the dissimilarity $w_{jk} = 1/d_{jk}$
- For each model of size k we calculate some performance statistics
- Choose as our model, the k that achieves the lowest RMSEP across the whole training set
- Very simple!

MAT in analogue I

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
  \let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
  \let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m data(swapdiat, swappH, rlgh)  
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
  \let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
  \let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m dat <- join(swapdiat, rlgh, ve
```

MAT in analogue II

Summary:

	Rows	Cols
Data set 1:	167	277
Data set 2:	101	139
Merged:	268	277

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m swapdiat <- with(dat, swapdiat
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

MAT in analogue III

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
  \let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m rlgh <- with(dat, rlgh / 100)  
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
  \let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
  \let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m swap.mat <- mat(swappH ~ ., da  
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
  \let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

MAT in analogue IV

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begin group \imm  
 \let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m swap.mat
```

Modern Analogue Technique

Call:

```
mat(formula = swappH ~ ., data = swapdiat, method = "SQchord")
```

Percentiles of the dissimilarities for the training set:

	1%	2%	5%	10%	20%
	0.416	0.476	0.574	0.668	0.815

Inferences based on the mean of k-closest analogues:

k	RMSEP	R2	Avg Bias	Max Bias
1	0.4227	0.7139	-0.0254	-0.3973
2	0.3741	0.7702	-0.0493	-0.4689
3	0.3387	0.8088	-0.0379	-0.4034

MAT in analogue V

4	0.3282	0.8200	-0.0335	-0.4438
5	0.3136	0.8356	-0.0287	-0.4124
6	0.3072	0.8444	-0.0386	-0.4152
7	0.3167	0.8364	-0.0481	-0.4179
8	0.3065	0.8474	-0.0433	-0.4130
9	0.3049	0.8495	-0.0436	-0.4111
10	0.3015	0.8548	-0.0473	-0.4083

Inferences based on the weighted mean of k-closest analogues:

k	RMSEP	R2	Avg Bias	Max Bias
1	0.4227	0.7139	-0.0254	-0.3973
2	0.3711	0.7734	-0.0476	-0.4614
3	0.3375	0.8102	-0.0385	-0.4088
4	0.3272	0.8213	-0.0346	-0.4433
5	0.3144	0.8348	-0.0298	-0.4205
6	0.3077	0.8435	-0.0371	-0.4253
7	0.3148	0.8377	-0.0451	-0.4250
8	0.3049	0.8483	-0.0407	-0.4206
9	0.3035	0.8500	-0.0408	-0.4205
10	0.3005	0.8546	-0.0442	-0.4180

MAT in analogue

- The RMSEP here is a leave-one-out RMSEP
- Each prediction for training set sample i is produced on the basis of using all sites other than i
- **analogue** is unique (as far as I know) as it evaluates all k models at once
- This means it is slow at times...
- ...But you only need to do the fitting once to determine the model with lowest RMSEP

MAT diagnostic plots

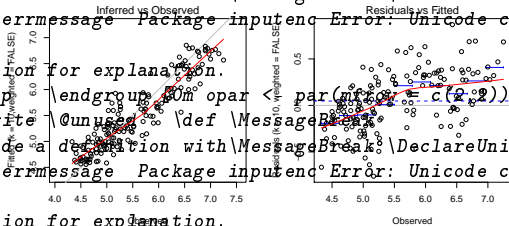
```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode chara
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imme
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode chara
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m opar < par(mf=0, E=c(2,2))
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode chara
```

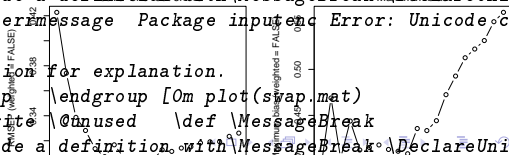


See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imme
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode chara
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m plot(swap.mat)
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
```



MAT predictions I

- To make a prediction for a fossil sample using MAT:
- Calculate dissimilarity between each fossil sample and each training set sample
- Take the k closest training set samples for each fossil sample
- The prediction for a fossil sample is the (weighted) average of these k closest training set samples

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \im
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m rlgh.mat <- predict(swap.mat,
\protect \begingroup \immediate \write \@unused \def \MessageBreak
```


MAT predictions II

```
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m rlgh.mat
```

Modern Analogue Technique predictions

Dissimilarity: SQchord

k-closest analogues: 10, Chosen automatically? FALSE

Weighted mean: FALSE

Bootstrap estimates: FALSE

Model error estimates:

RMSEP	r.squared	avg.bias	max.bias
0.3015	0.8548	-0.0473	-0.4083

Predicted values:

MAT predictions III

000.3	000.8	001.3	001.8	002.3	002.8	003.3	003.8	004.3	004.8	005.3	006.3
4.82	4.79	4.83	4.78	4.79	4.82	4.79	4.84	4.79	4.87	4.84	4.90
007.3	008.3	009.3	010.3	011.8	013.3	014.3	015.3	016.3	017.3	018.3	019.3
4.88	5.01	5.10	5.12	5.13	5.26	5.50	5.43	5.43	5.36	5.32	5.36
020.3	022.3	024.3	025.3	026.3	027.3	028.3	030.5	032.5	036.5	040.5	044.5
5.37	5.40	5.50	5.36	5.37	5.48	5.36	5.36	5.47	5.57	5.36	5.55
048.5	052.5	056.5	060.5	064.5	068.5	072.5	076.5	080.5	084.5	088.5	092.5
5.63	5.75	5.72	5.57	5.42	5.44	5.62	5.75	5.51	5.51	5.35	5.51
096.5	100.5	104.5	108.5	112.5	118.5	120.5	124.5	128.5	130.5	132.5	134.5
5.47	5.43	5.46	5.41	5.48	5.31	5.51	5.60	5.66	5.40	5.25	5.41
136.5	138.5	140.5	142.5	144.5	146.5	148.5	150.5	152.5	154.5	156.5	158.5
5.41	5.45	5.49	5.43	5.01	5.23	5.27	5.27	5.27	5.27	5.27	5.31
160.5	162.5	164.5	166.5	168.5	170.5	172.5	174.5	176.5	178.5	180.5	182.5
5.27	5.27	5.25	5.28	5.28	5.23	5.24	5.27	5.27	5.27	5.20	5.23
184.5	188.5	192.5	196.5	200.5	204.5	208.5	212.5	216.5	220.5	224.5	228.5
5.20	5.23	5.30	5.23	5.20	5.21	5.21	5.22	5.20	5.19	5.57	5.61
244.5	248.5	252.5	254.5	256.5							
5.59	5.59	5.61	5.69	5.62							

MAT reconstructions

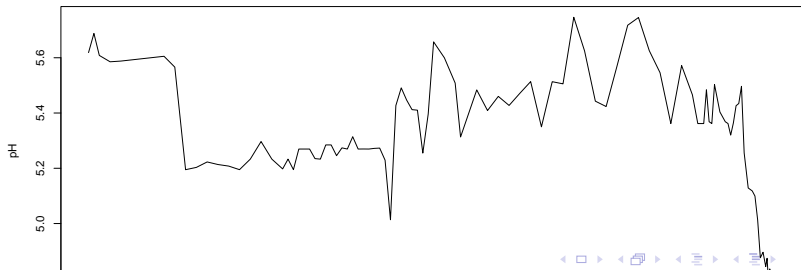
```
\protect \begingroup \immediate \write \@unused \def \MessageBreak  
  \let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm  
  \let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicod  
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

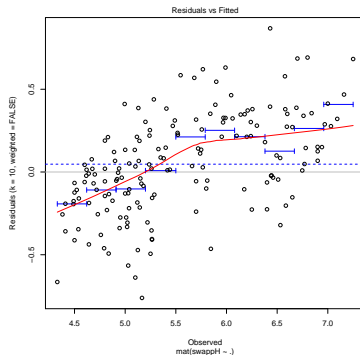
See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m reconPlot(rlgh.mat, use.labels
```



Bias

- Bias is the tendency for the model to over or under predict
- Average bias is the mean of the residuals
- Maximum bias is found by breaking the range of the measured environment into n contiguous chunks ($n = 10$ usually)
- Within each chunk calculate the mean of the residuals for that chunk
- Take the maximum value of these as the maximum bias statistic



Cross-validation

- Without cross-validation, prediction errors, measured by RMSEP, will be biased, often badly so
- This is because we use the same data to both fit **and** test the model
- Ideally we'd have such a large training set that we can split this into a slightly smaller training set and a small test set
- Palaeoecological data is expensive to obtain — in money and person-hours!
- Also these ecosystems are complex, species rich, noisy etc., so we want to use all our data to produce a model
- One solution to this problem is to use cross-validation
- General idea is we perturb our training set in some way, build a new model on the perturbed training set and assess how well it performs
- If we repeat the perturbation several time we get an idea of the error in the model
- Several techniques; n -fold, leave-one-out, bootstrapping (aka **bagging**)

Cross-validation in **analogue**

- In **analogue**, several methods are available
- For MAT models, LOO is built into the procedure so only bootstrapping is available
- For WA models, both LOO and bootstrapping currently available
- n -fold CV will be available in a future version

LOO Cross-validation in **analogue**

- LOO CV is very simple
- In turn, leave out each sample from the training set
- Build a model on the remaining samples
- Predict for the left out sample
- Calculate the RMSEP of these predictions

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m loo.pred <- predict(mod, imbri
Leave one out sample 10
Leave one out sample 20
Leave one out sample 30
Leave one out sample 40
Leave one out sample 50
Leave one out sample 60
```

Bootstrap Cross-validation in **analogue**

- Bootstrapping used in machine learning to improve predictions
- Use bootstrapping to get more realistic RMSEP and bias statistics
- We draw a bootstrap sample (sampling with replacement) of the same size as our training set
- Build a model on the bootstrap samples
- Predict for the out-of-bag (OOB) samples
- Bootstrap prediction for each model sample is the mean of the OOB prediction for each sample
- Calculate the residuals and then the RMSEP

$$\text{RMSEP}_{\text{boot}} = \sqrt{s_1^2 + s_2^2}$$

- s_1^2 is the standard deviation of the OOB residuals
- s_2^2 is the mean of the OOB residuals
- We can also calculate the more usual RMSEP $\sqrt{\sum_{i=1}^n (y_i - \hat{y}_i)^2 / n}$

Bootstrap Cross-validation in **analogue**

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m set.seed(1234)
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m swap.boot <- bootstrap(swap.ma
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicode
```

Minimum dissimilarity to a training set sample

- A measure of reliability for the reconstructed values can be determined from the distance between each fossil sample and the training set samples
- For a reconstructed value to be viewed as more reliable, it should have at least one close modern analogue in the training set
- Close modern analogues are defined as those modern training set samples that are as similar to a fossil sample as a low percentile of the observed distribution dissimilarities in the training set, say the 5th percentile

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m rlgh.mdc <- minDC(rlgh.mat)
```

Minimum dissimilarity to a training set sample

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m plot(rlgh.mdc, use.labels = TR
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m quantile(as.dist(swap.mat$Dij)
1% 2.5% 5% 10%
0.4164113 0.4972167 0.5738378 0.6676391
```

Sample-specific error estimates

- We can use the bootstrap approach to generate sample specific errors for each fossil sample

$$\text{RMSEP} = \sqrt{s_{1_{\text{fossil}}}^2 + s_{2_{\text{model}}}^2}$$

- $s_{1_{\text{fossil}}}^2$ is the standard deviation of the bootstrap estimates for the fossil samples
- $s_{2_{\text{model}}}^2$ is the average bias, the mean of the bootstrap OOB residuals from the model

Sample-specific error estimates

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the inputenc package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m swap.boot
Bootstrap results for palaeoecological models
```

Model type: MAT

Weighted mean: FALSE

Number of bootstrap cycles: 200

Leave-one-out and bootstrap-derived error estimates:

	k	RMSEP	S1	S2	r.squared	avg.bias	max.bias
L00	10	0.301	-	-	0.855	-0.0473	-0.408
Bootstrap	11	0.327	0.12	0.305	0.924	-0.0501	-0.435

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with\MessageBreak \DeclareUnicod
```

Sample-specific error estimates

```
\protect \begingroup \immediate \write \@unused \def \MessageBreak
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0;34mr$>\protect \begingroup \imm
\let \protect \edef You may provide a definition with \MessageBreak \DeclareUnicode
(inputenc) \def \errmessage Package inputenc Error: Unicode char
```

See the `inputenc` package documentation for explanation.

```
Type H <return> for immediate help \endgroup [0m reconPlot(rlgh.boot, use.label
```

