

Spatial Statistics

Helene Wagner, University of Toronto

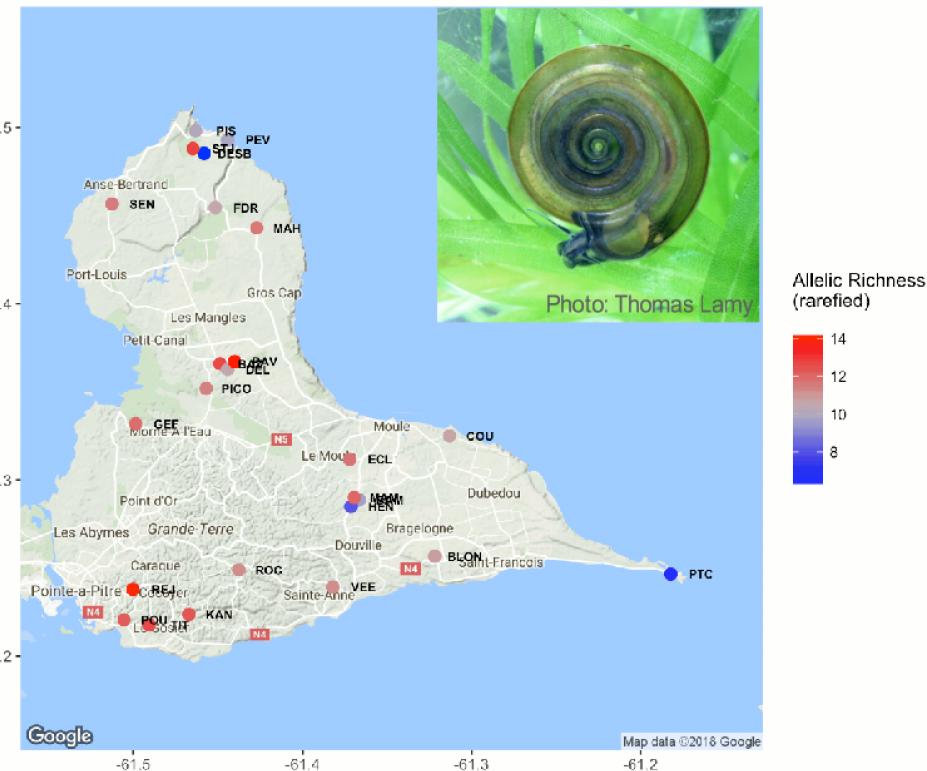
Goal:

- Test for spatial autocorrelation in the snail data set
- Genotypic data (pairwise genetic distances among pops)
 - Site-level data (diversity, site-level Fst, predictors)

Technical Challenges:

- From allele frequencies to genetic distances
- Exploratory analysis: scatterplot density and smooth line
- Three ways to analyze pair-wise distances
- Defining spatial weights in R with package 'spdep'
- How to access S3 and S4 object attribute data

Genetic Diversity of *D. depressissimum*



Genetic Distances

Individual-level

	Locus A
Ind 1	204 : 210
Ind 2	204 : 204
Ind 3	210 : 218
Ind 4	NA : NA

adegenet :: makefreq

	204	210	218	
Ind 1	0.5	0.5	0	1
Ind 2	1	0	0	1
Ind 3	0	0.5	0.5	1
Ind 4	0	0	0	0

missing = c(NA, 0, "mean")

adegenet :: propShared

	Ind 1	Ind 2	Ind 3	Ind 4
Ind 1	1	0.5	0.5	0
Ind 2	0.5	1	0	0
Ind 3	0.5	0	1	0
Ind 4	0	0	0	1

Dps = 1 - propShared

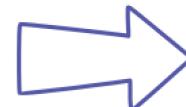
	Ind 1	Ind 2	Ind 3	Ind 4
Ind 1	0	0.5	0.5	1
Ind 2	0.5	0	1	1
Ind 3	0.5	1	0	1
Ind 4	1	1	1	0

Population-level

adegenet :: makefreq

	204	210	218	
Pop 1	0.5	0.33	0.17	1
Pop 2				1
Pop 3				1

Choose
distance measure



adegenet :: dist.genpop
gstudio :: genetic_distance

	Pop 1	Pop 2	Pop 3
Pop 1	0	d12	d13
Pop 2	d21	0	d23
Pop 3	d31	d32	0

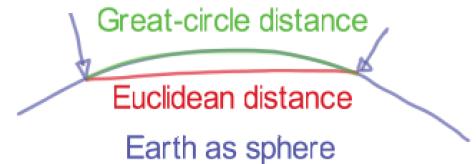
Check how missing values are treated at each step (import, allele frequencies, genetic distance)

'dist' Objects in R

Example: Geographic distance

	XY	
	X-coord	Y-coord
Pop 1		
Pop 2		
Pop 3		

- XY coordinates must be metric (e.g., UTM)
- Lat-Lon coordinates must be converted
- Euclidean distance (or great-circle distance)



`D = dist(XY)`

`as.matrix(D)`

`as.dist(D)`

`as.vector(D)`

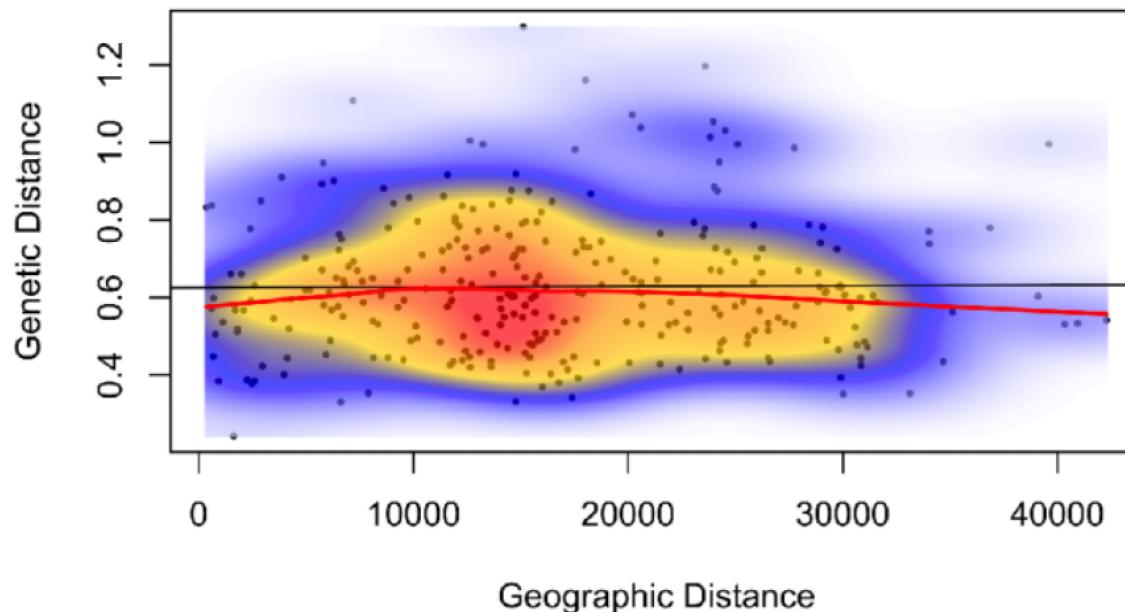
	Pop 1	Pop 2	Pop 3
Pop 1	0	d12	d13
Pop 2	d21	0	d23
Pop 3	d31	d32	0

	Pop 1	Pop 2
Pop 2	d21	
Pop 3	d31	d32

`c(d21, d31, d32)`

Exploratory Analysis

Visualization with kernel density and smooth regression line



- 2D kernel density:
- Color palette:
- Add image to plot:
- Smooth regression line:

```
density <- MASS :: dke2d( Dgeo, Dgen )
myPalette <- colorRampPalette( c( "white", "blue", "gold", "orange", "red" ) )
image( density, myPalette, add = TRUE)
loess.smooth( Dgeo, Dgen )
```

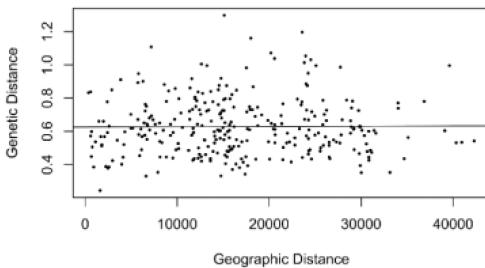
What Hypothesis to Test?

Biological hypothesis of IBD:
Positive spatial autocorrelation

- Direction?
- Effect size?
- Statistical significance?



Mantel test



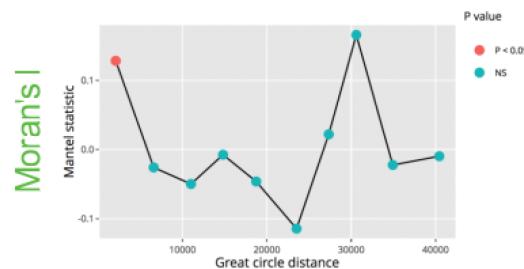
Testing for IBD:
- One-sided test?

Pop Gen (Rousset 1997):

- Linearize: $D = FST / (1 - FST)$
- 2D: Plot D against $\log(Dgeo)$

$$b = \frac{1}{4N\pi\sigma^2}$$

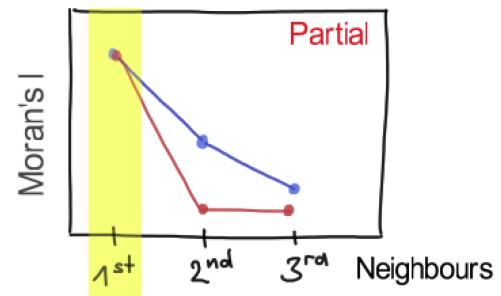
Correlogram



Testing for IBD:
- One-sided test?

Sequential testing:
- Test first lag
- If significant, test second
- Sequential Holm's correction

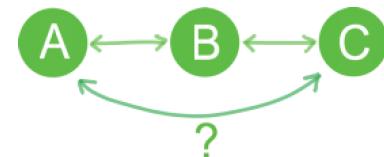
Moran's I



Testing for IBD:
- One-sided test?

Stepping-stone model:

- Test only first neighbours



Three Analytical Paradigms

	All pairs	Distance lags	First neighbors
What?	Mantel test (usually multivariate)	Multivariate: Mantel correlogram Univariate: Moran correlogram	Moran's I (usually univariate)
How?	Global	For each distance lag	Among first neighbors
R packages?	<code>ade4:: mantel.randtest</code> <code>vegan :: mantel</code>	<code>EcoGenetics :: eco.mantelcor</code> <code>EcoGenetics :: eco.correlogram</code>	<code>spdep :: moran.mc</code> <code>spdep :: lm.morantest</code>

The diagram illustrates the three analytical paradigms with a central vertical line and two red curved lines branching out to the left and right. The left branch is labeled 'All pairs' and 'Ecology: multivariate dissimilarity', featuring a scatter plot of genetic vs geographic distance with a regression line. The right branch is labeled 'Distance lags' and 'Geostatistics: random fields', featuring a line graph of the Mantel statistic vs great circle distance with points colored by P-value. The central vertical line is labeled 'First neighbors' and 'Geography: spatial objects', featuring a map with colored regions and a network of connected nodes representing first neighbors.

Connection Networks with 'spdep'

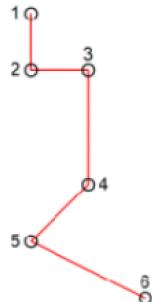
Neighbour definition

```
adegenet :: chooseCN( XY )
```

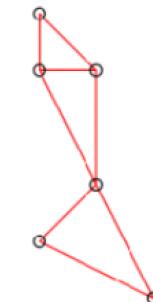
Local mean:

- More neighbours: more precise
- Fewer neighbours: more local

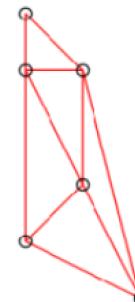
Minimum
spanning tree
'type = 4'



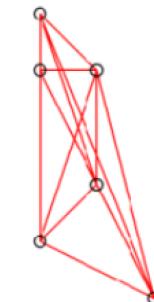
Gabriel
graph
'type = 2'



Delaunay
triangulation
'type = 1'



Inverse
distances
'type = 7'



Spatial weights

```
spdep :: nb2listw( neighbours, style, glist )
```

Binary

'style = "B" '

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	
1	0	1	1	0	0	0	2
2	1	0	1	1	0	0	3
3	1	1	0	1	0	0	3
4	0	1	1	0	1	1	4
5	0	0	0	1	0	1	2
6	0	0	0	1	1	0	2

Row-standardized (default)

'style = "W" '

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	
1	1	0.00	0.50	0.50	0.00	0.00	1
2	2	0.33	0.00	0.33	0.33	0.00	1
3	3	0.33	0.33	0.00	0.33	0.00	1
4	4	0.00	0.25	0.25	0.00	0.25	1
5	5	0.00	0.00	0.00	0.50	0.00	1
6	6	0.00	0.00	0.00	0.50	0.50	1

Distance-weighted

use 'glist'

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	
1	1	0.00	0.41	0.59	0.00	0.00	0.00
2	2	0.24	0.00	0.24	0.53	0.00	0.00
3	3	0.32	0.23	0.00	0.45	0.00	0.00
4	4	0.00	0.28	0.25	0.00	0.18	0.28
5	5	0.00	0.00	0.00	0.39	0.00	0.61
6	6	0.00	0.00	0.00	0.50	0.50	0.00

Object Types in 'spdep'

List of length n:
- list of neighbours

Class 'nb'

```
[[1]]  
[1] 2 3  
  
[[2]]  
[1] 1 3 4  
  
[[3]]  
[1] 1 2 4  
  
[[4]]  
[1] 2 3 5 6  
  
[[5]]  
[1] 4 6  
  
[[6]]  
[1] 4 5
```

Class 'listw'

```
$weights  
$weights[[1]]  
[1] 0.5 0.5  
  
$weights[[2]]  
[1] 0.3333333 0.3333333 0.3333333  
  
$weights[[3]]  
[1] 0.3333333 0.3333333 0.3333333  
  
$weights[[4]]  
[1] 0.25 0.25 0.25 0.25  
  
$weights[[5]]  
[1] 0.5 0.5  
  
$weights[[6]]  
[1] 0.5 0.5
```

List of length 3:
- \$method
- \$neighbours
- \$weights

Extracting Results

For a list of distance matrices (DgenList)

```
ResList <- lapply( DgenList, function( x ) eco.cormantel( x, XY ))
```

```
lapply( ResList, function( x ) ecoslot.OUT( x ))
```

```
[[1]]
```

	d.mean	obs	exp	p.val	cardinal
d= 0 - 4228.418	2090.772	0.0687	0.0006	0.165	28
d= 4228.418 - 8456.836	6592.930	0.0017	-0.0027	0.470	34
d= 8456.836 - 12685.253	11013.614	-0.0688	-0.0036	0.940	43
d= 12685.253 - 16913.671	14810.461	-0.0511	-0.0016	1.000	67
d= 16913.671 - 21142.089	18741.818	0.0032	-0.0045	1.000	32
d= 21142.089 - 25370.507	23525.813	-0.0849	-0.0025	1.000	38
d= 25370.507 - 29598.925	27343.091	0.0498	-0.0075	1.000	28
d= 29598.925 - 33827.342	30605.163	0.1744	-0.0017	0.080	20
d= 33827.342 - 38055.76	34933.340	-0.0534	0.0094	1.000	5
d= 38055.76 - 42284.178	40442.266	0.0075	0.0136	1.000	5

```
sapply( ResList, function( x ) ecoslot.OUT( x )[ [ 1 ] ][ 1, c( 2, 4 ) ] )
```

	pairwiseFst	propShared	Nei	Edwards	Reynolds	Rogers	Provesti	Joost	Hedrick
obs	0.0488	0.0687	0.1284	0.0743	0.0598	0.0661	0.0687	0.1154	0.1105
p.val	0.2250	0.1350	0.0100	0.1500	0.1950	0.1600	0.1250	0.0400	0.0450

For one genetic distance matrix

```
Res <- eco.cormantel( Dgen, XY )
```

```
slotNames( Res )
```

```
Res@OUT  
Result <- ecoslot.OUT( Res )
```

```
> class(Result[1])  
[1] "list"  
> class(Result[[1]])  
[1] "matrix"
```

```
> Result[[1]][1,2]  
[1] 0.0687  
> Result[[1]][1,4]  
[1] 0.165
```

Class: 'eco.correlog' (S4)

S3: attributes()
S4: slotNames()

Class: list of length 1

Mantel statistic for first lag

P-value for first lag

```
ecoslot.OUT( Res )[ [ 1 ] ][ 1, c( 2, 4 ) ]
```

Extract 'obs' and 'p.val'