Mantel Test Script

Grace Saville

01/02/2022

1. Preamble

```
library(ade4)
library(geosphere)
library(ggplot2)
library(car)
library(plyr)
getwd()
setwd("C:/Users/airhe/OneDrive/Documents/Masters/Project 3/kiore-project")
```

2. Loading the data

```
data <- read.csv("./data/RStudio/ratsSNPs_clean.csv")
gen.matrix <- as.matrix(read.delim("./data/SplitsTree/geneticdist_SplitsTree_output.txt", sep = "\t", h
# Make sure the text file is the distances only, remove any extras e.g. column vector at bottom of file
```

3. Creating geographical distance matrix

ONLY NEED TO RUN THE FOLLOWING CODE CHUNK ONCE AND THEN SAVE

Using the mapview funtion in the last code chunk of this script I noticed that Tahanea is plotted in Australia, and found out the longitude number is missing a negative sign, and that Reiono and Honuea have the same coordinates even though they're different islands. Fixing that here:

```
x <- grep("Tahanea", data$island.1, value = FALSE) # finding rows of Tahanea
names(data) # finding column number for "geo_long"
data[x, 12]
data[x, 12] <- -144.97 # replacing the number

# the coordinates given for Reiono and Honuea point to an island in French Polynesia called Moorea-Maia
# Reino should be approx. -17.046, -149.546
# Honuea should be approx -17.009, -149.585
# Checking other islands close by:
# Rimatu'u is -17, -149.57 (over the sea) but should be approx -17.03, -149.558
```

```
x <- grep("Reiono", data$island.1, value = FALSE)
data[x, c(11, 12)]
data[x, 11] <- -17.046 # replcing geo_lat
data[x, 12] <- -149.546 # replacing geo_long

x <- grep("Honuea", data$island.1, value = FALSE)
data[x, c(11, 12)]
data[x, 11] <- -17.009 # replcing geo_lat
data[x, 12] <- -149.585 # replacing geo_long

x <- grep("Rimatuu", data$island.1, value = FALSE)
data[x, c(11, 12)]
data[x, 11] <- -17.03 # replcing geo_lat
data[x, 12] <- -149.558 # replacing geo_long

rm(x)

write.csv(data, "./data/RStudio/ratsSNPs_clean.csv", row.names = FALSE)
```

(only need to do the above code once since it saves the edited df to file)

4. Setting up longitudes and latitudes

Different distance functions:

- distHaversine() assumes earth is a sphere
- distm() makes distance matrix
- distGeo() assumes earth is elliptical (ish), can choose specific model

```
names(data) # need "geo_lat" "geo_long"
longlat <- data[,c(1,11,12)]
head(longlat) # checking correct columns are used

longlat <- as.matrix(longlat[,c(3,2)]) # distGeo function needs a matrix with 2 columns, col 1 longitud

geo.matrix <- distm(longlat, fun = distGeo) # converting to pairwise distance matrix
dim(geo.matrix) # 370 370

geo.dist <- as.dist(geo.matrix, diag = TRUE, upper = TRUE) # converting to dist object
# diag = TRUE #includes diagonal zeros
# upper = TRUE #includes upper triangle
```

5. Creating genetic distance matrix

```
dim(gen.matrix) # 370 370
gen.dist <- as.dist(gen.matrix, diag = TRUE, upper = TRUE) # converting to dist object
```

6. Mantel test

```
r1 <- mantel.rtest(gen.dist, geo.dist, nrepet = 999)
r1
plot(r1$plot$hist, main = "Mantel test", xlim = c(-0.1, 0.1)) # plotting simulated p-values
sum(r1$plot$hist$counts)
```

Results:

> Monte-Carlo test
> Call: mantelnoneuclid(m1 = m1, m2 = m2, nrepet = nrepet)
>
> Observation: 0.4987612
>
> Based on 999 replicates
> Simulated p-value: 0.001
> Alternative hypothesis: greater
>
> Std.Obs: 2.370409e+01
> Expectation: 5.514360e-04
> Variance: 4.417514e-04

- -1 suggests strong negative correlation, e.g. closer islands mean further genetically or further islands means closer genetically
- 0 suggests no correlation, e.g. genetic difference is not correlated to island distance
- 1 suggests strong positive correlation e.g. closer islands mean closer genetically

Therefore the observed correlation of 0.4987612 suggests that there is a positive correlation between genetic distance and geographic distance (and the null hypothesis of no correlation is rejected).

Results before I fixed the missing negative sign in Tahanea and other location issues:

> Monte-Carlo test
> Call: mantelnoneuclid(m1 = m1, m2 = m2, nrepet = nrepet)
>
> Observation: 0.4345602
>
> Based on 999 replicates
> Simulated p-value: 0.001
> Alternative hypothesis: greater
>
> Std.Obs: 2.329675e+01
> Expectation: 7.916798e-04
> Variance: 3.466772e-04

Results from uncleaned dataset:

> Monte-Carlo test
> Call: mantelnoneuclid(m1 = m1, m2 = m2, nrepet = nrepet)

Observation: 0.2807331

Based on 99 replicates
Simulated p-value: 0.01
Alternative hypothesis: greater

Std.Obs: 26.5367570885
Expectation: -0.0002961658
Variance: 0.0001121521

7. Creating pairwise distances dataframe

```
specimens <- read.delim("./data/SplitsTree/geneticdist_SplitsTree_output_taxa_only.txt", header = FALSE)
specimens <- as.vector(specimens[,2])

colnames(gen.matrix) <- specimens
rownames(gen.matrix) <- specimens # naming the rows and columns by the order given in the SplitsTree ou

colnames(geo.matrix) <- data[,1]
rownames(geo.matrix) <- data[,1] # naming the rows and columns here by the order of lat/long, which cam

gen.matrix[lower.tri(gen.matrix)] <- NA # keeping only the upper triangle of each matrix
geo.matrix[lower.tri(geo.matrix)] <- NA

dist.summary <- data.frame(
  col = colnames(gen.matrix)[col(gen.matrix)],
  row = rownames(gen.matrix)[row(gen.matrix)],
  gen.dist = c(gen.matrix)
) # converting the genetic matrix into a df with columns describing which combos result in the distance

x <- data.frame(
  col = colnames(geo.matrix)[col(geo.matrix)],
  row = rownames(geo.matrix)[row(geo.matrix)],
  geo.dist = c(geo.matrix)
) # doing the same with the geographic matrix

dist.summary <- merge(dist.summary, x, by = 1:2, all = TRUE) # merging the 2 dfs
rm(x)
```

The above df "dist.summary" is not ideal since it's not space efficient (~12MB), however it provides and easy way to link the row and column specimens that generated the distances, therefore making points on the graph label-able.

```
dist.summary <- tidyr::unite(dist.summary, specimens.combo, 1:2, sep = ":", remove = TRUE) # combining

dist.summary <- na.omit(dist.summary) # removing NA's that were in the bottom triangle

# creating a column of island combination labels, rather than specimen combination:
islands.combo <- dist.summary$specimens.combo
islands.combo <- gsub("[0-9]+", "", islands.combo) # removing specimen ID numbers
```

```
islands.combo <- gsub("_", "", islands.combo) # removing underscores
unique(islands.combo)
islands.combo <- gsub("Thailand", "Mainland", islands.combo) # replacing 3 countries with mainland
islands.combo <- gsub("Laos", "Mainland", islands.combo)
islands.combo <- gsub("Cambodia", "Mainland", islands.combo)
islands.combo <- gsub("Halmahera", "Halmaher", islands.combo) # replacing some double ups
islands.combo <- gsub("NewBritai", "NewBrita", islands.combo)
islands.combo <- gsub("NewGuinea", "NewGuine", islands.combo)
islands.combo <- gsub("Motukawan", "Motukawa", islands.combo)
dist.summary <- cbind(dist.summary, islands.combo) # adding this new column
rm(islands.combo)

# removing self-self distances, since I'm looking at inter-island not intra-island
intra <- as.character(c(
  "Aotea:Aotea",
  "Borneo:Borneo",
  "GrtMercury:GrtMercury",
  "Halmaher:Halmaher",
  "Hatutaa:Hatutaa",
  "Honuea:Honuea",
  "Kaikura:Kaikura",
  "Kamaka:Kamaka",
  "Kayangel:Kayangel",
  "LateIs:LateIs",
  "Luzon:Luzon",
  "Mainland:Mainland",
  "Malenge:Malenge",
  "Mohotani:Mohotani",
  "Motukawa:Motukawa",
  "NewBrita:NewBrita",
  "NewGuine:NewGuine",
  "Normanby:Normanby",
  "Rakiura:Rakiura",
  "Reiono:Reiono",
  "Rimatuu:Rimatuu",
  "Slipper:Slipper",
  "Southland:Southland",
  "Sulawesi:Sulawesi",
  "Tahanea:Tahanea",
  "WakeIs:WakeIs"
))
dist.summary[which(dist.summary$islands.combo %in% intra),4] <- NA # removing same-same island combos
dist.summary[dist.summary == 0] <- NA # turning zeros to NAs
dist.summary <- na.omit(dist.summary) # removing rows with NAs
rm(intra, specimens)

dist.summary$geo.dist <- dist.summary$geo.dist/1000 # going from metres to km
```

Rows which have either or both gen and geo dist at zero (although there should never be a gen.dist at zero) will sit on the axis and potentially pull on the regression line. Geo distances at 0 will always be specimens on the same island, which I think it best to remove since they don't contribute to whether there is a correlation between islands distance (since they're actually the same island).

```
getwd()
write.csv(dist.summary, "./data/RStudio/gen_geo_distance_matrices_df.csv", row.names = FALSE)
```

8. Linear Modelling

```
dist.summary <- read.csv("./data/RStudio/gen_geo_distance_matrices_df.csv")
```

8a. Test Model

```
testLM <- lm(gen.dist ~ geo.dist, data = dist.summary) # model
summary(testLM) # model results

plot(gen.dist ~ geo.dist, data = dist.summary)
abline(coef = coef(testLM), col = 4, lwd = 2)
```

```
par(mfrow = c(2, 2)) # changes the number of plots visible at once
plot(testLM) # # diagnostic plots, not normal, distribution may be skewed towards the left side ("Right

par(mfrow = c(1, 1))

qqPlot(testLM$residuals, line = "quartiles") # non-normal dist
shapiro.test(testLM$residuals) # doesn't work, sample size must be between 3 and 5000
ks.test(testLM$residuals, 'pnorm') # D = 0.43238, p-value < 2.2e-16, non-normal
hist(testLM$residuals, breaks = 50) # clear positive skew

ncvTest(testLM) # homoscedasticity test: Chisquare = 950.7684, Df = 1, p = < 2.22e-16, H0 of constant v

influenceIndexPlot(testLM) # outliers, hard to judge since there seem to be many, or none which are an
outlierTest(testLM) # lists 10 different points which could be an issue (all over 3 rstudent and signif

boxCox(testLM) # recommended log transformation
```

I decided to do a brief check of observed correlation to compare with the Mantel test results, using the above dataframe;

```
cor.test(dist.summary$gen.dist, dist.summary$geo.dist, method = "pearson") # should be taken with a gra
# t = 134.54, df = 65067, p-value < 2.2e-16
# 95 percent confidence interval: 0.4604831 0.4725059
# sample estimates: cor 0.466516
```

8b. Adjusted Model

I attempted to transform the data (e.g. sqrt, log10) to achieve normal residual distribution however it was not successful so I will build a GLM instead.

```r
GLM <- glm(gen.dist ~ geo.dist, data = dist.summary, family = Gamma(link = "identity")) # Gamma distrub
# I tried the link functions "log" and "inverse" first but the resulting coefficients were very peculia
summary(GLM) # model results

par(mfrow = c(1, 1))
plot(gen.dist ~ geo.dist, data = dist.summary)
abline(coef = coef(GLM), col = 4, lwd = 2)
```

```r
par(mfrow = c(2, 2)) # changing the number of plots visible at once
plot(GLM) # diagnostic plots

residualPlots(GLM)

qqPlot(GLM$residuals, line = "quartiles") #
ks.test(GLM$residuals, 'pnorm') #
hist(GLM$residuals, breaks = 50)
```

8c. GLM Results

Call: glm(formula = gen.dist ~ geo.dist, family = Gamma(link = "identity"), data = z)

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.7423	-0.5323	-0.2198	0.1474	3.0278

Coefficients:

	Estimate	Std. Error	t value	p-value
(Intercept)	9.434e-02	6.950e-04	135.7	<2e-16 ***
geo.dist	2.013e-05	1.621e-07	124.2	<2e-16 ***

Signif. codes: 0 ‘’ ***0.001*** ‘’ *0.01* ‘’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Gamma family taken to be 0.4959804)

Null deviance: 30821 on 65068 degrees of freedom
Residual deviance: 22856 on 65067 degrees of freedom
AIC: -117995

Number of Fisher Scoring iterations: 5

```r
with(summary(GLM), 1 - deviance/null.deviance) # R^2 value of 0.2584382. This is not high therefore the
```

9. Looking at the genetically closest island pairs

```
dist.summary <- dist.summary[order(dist.summary$geo.dist, decreasing = FALSE),] # sorting by geo dist
dist.summary <- dist.summary[order(dist.summary$gen.dist, decreasing = FALSE),] # sorting by gen dist

head(dist.summary[!duplicated(dist.summary$islands.combo),], n = 10)
```

10. Taking a closer look at islands with high distance/close genetics

```
fv <- as.vector(GLM$fitted.values)
dist.summary <- cbind(dist.summary, fv)

under.fv <- data.frame()
for (i in 1:nrow(dist.summary)) { # for every row of the island data (island to island)
  if (dist.summary[i,2] < dist.summary[i,5]) { # if the genetic distance is less than the fitted value
    under.fv <- rbind(under.fv, dist.summary[i,]) # row bind
  }
}
rm(i)

plot(gen.dist ~ geo.dist, data = under.fv) # checking, should be all values up to the regression line

all.combos <- count(dist.summary$islands.combo)
under.fv.combos <- count(under.fv$islands.combo)
combo.totals <- merge(all.combos, under.fv.combos, by = "x", all = TRUE)
colnames(combo.totals) <- c("combo", "all", "below.line")
rm(all.combos, under.fv.combos)
combo.totals$perc.under <- round(((combo.totals$below.line / combo.totals$all) * 100), 3)
combo.totals[is.na(combo.totals)] <- 0

# perc.under is the percentage of the specimens with genetic distances (between the two stated islands)

combo.totals$combo[combo.totals$perc.under == 100]
```

11. Plots

```
ggplot(data = dist.summary, aes(x = geo.dist, y = gen.dist)) +
  geom_point(shape = 1, colour = "grey") +
  geom_smooth(method = "lm") +
  ggtitle("Correlation between Geographic and Genetic Distance between Islands") +
  xlab("Geographic Distance between two islands (km)") + ylab("Genetic Distance between two islands") +
  theme_light()

# log transformed:
ggplot(data = dist.summary, aes(x = (geo.dist)^2, y = gen.dist)) +
  geom_point(shape = 1, colour = "grey") +
  geom_smooth(method = "lm") +
```

```
  ggtitle("Geographic versus Genetic Distance between Islands") +
  xlab("Geographic Distance between 2 islands (km)") + ylab("Genetic Distance between 2 islands") +
  theme_light()
```

11. Experimenting with plotting the coordinates on a map (in progress)

```
names(data)
longlat <- data[,c(1,8,11,12)]
longlat <- longlat[!duplicated(longlat$island.1),] # keeping only 1 coordinate for each island
longlat <- longlat[,-1]
longlat

library(sf)
x <- st_as_sf(longlat, coords = c("geo_long", "geo_lat"),  crs = 4326) # 4326 is WGS84 Coordinate Refer

library(mapview)
mapview(x, grid = FALSE, map.types = "OPNVKarte", legend = FALSE, native.crs = FALSE)
```

For the code above I would need to find an alternative to mapview because there appears to be a bug where the legend colours don't match up to the map point colours