Testing the Heterozygosity per Island

Grace Saville

01/02/2022

Preamble

```
library(stringr)
library(dplyr)
getwd()
setwd("C:/Users/airhe/OneDrive/Documents/Masters/Project 3/kiore-project")
```

Dataset loading

```
data <- read.csv("./data/RStudio/ratsSNPs_clean.csv")

copy <- data # making a copy of the data
colnames(copy)
copy <- copy[,-c(2:16)] # removing unnecessary columns for this analysis
```

Replaces SNP's with symbols for heterozygous and homozygous

```
dim(copy) # 370 rows 283 columns
str(copy)
unique(unlist(copy[,17:283])) # checking what SNP combinations are present
# "?"   "A:A" "G:G" "A:G" "T:T" "A:T" "C:C" "C:T" "T:G" "C:G" "A:C"

het <- c("A:G", "A:T", "C:T", "T:G", "C:G", "A:C") # vector of heterozygous combinations
hom <- c("G:G", "A:A", "C:C", "T:T") # vector of homozygous combinations

# Replacing specific combos
for (i in 1:nrow(copy)){
  copy[i,][copy[i,] %in% het] <- "E"
  copy[i,][copy[i,] %in% hom] <- "O"
}

rm(i)

str(copy)
copy[copy == "?"] <- NA # replacing ? with NA's
```

Calculating heterozygous and homozygous totals per specimen

```
o.freq <- vector()
e.freq <- vector()
na.freq <- vector()
for (i in 1:370) {
  x <- sum(grepl("O", copy[i,], fixed = TRUE)) # counting row O's
  o.freq <- append(o.freq, x) # adding sum to vector
  x <- sum(grepl("E", copy[i,], fixed = TRUE)) # counting row E's
  e.freq <- append(e.freq, x)
  x <- sum(is.na(copy[i,])) # counting row NA's
  na.freq <- append(na.freq, x)
  rm(x)
}

rm(i)

per.specimen <- data.frame(copy$island, o.freq, e.freq, na.freq) # making a df with the freq totals
perc.o <- round(as.vector((per.specimen$o.freq / (per.specimen$o.freq + per.specimen$e.freq)) * 100), di
per.specimen$perc.o <- perc.o
perc.missing <- round(as.vector((per.specimen$na.freq / (per.specimen$o.freq + per.specimen$e.freq + pe
per.specimen$perc.missing <- perc.missing

rm(o.freq, e.freq, na.freq, perc.o, perc.missing)

str(per.specimen)
```

Calculating heterozygous and homozygous totals per island

```
data <- data[order(data$island, decreasing = FALSE),] # ordering df alphabetically by island
names(data)
data[c(grep("Mainland", data$island.1)),c(1, 8)] # checking which populations fall in the mainland cate
unique(data$island)
shortpopnames <- as.character(
  c(
    "aotea",
    "borneo",
    "cambodia",
    "grtmercury",
    "halmaher",
    "hatutaa",
    "honuea",
    "kaikura",
    "kamaka",
    "kayangel",
    "laos",
    "late",
    "luzon",
    "malenge",
    "mohotani",
    "motukawa",
    "newbrita",
    "newguine",
    "normanby",
    "rakiura",
    "reiono",
    "rimatuu",
    "slipper",
    "southland",
    "sulawesi",
    "tahanea",
    "thailand",
    "wake"
  )
) # writing shortened names as is in the data df so I can use the character strings with grep()

# splitting the data df into df objects by island:
for (i in 1:length(shortpopnames)) {
  y <- as.vector(grep(shortpopnames[i], copy[,1], ignore.case = TRUE, value = FALSE))
  assign(paste(shortpopnames[i]), copy[y,])
}
rm(y, i)

# making a "mainland" df:
mainland <- rbind(cambodia, thailand, laos)
rm(cambodia, thailand, laos)
shortpopnames
shortpopnames <- shortpopnames[-c(3, 11, 27)] # removing the names now in mainland category
shortpopnames <- append(shortpopnames, "mainland")

# double checking the dfs have the correct specimens per island:
length(shortpopnames) #26
for (i in 1:26) {
  print(shortpopnames[i])
  y <- get(shortpopnames[i])
  print(y[,1])
}
rm(y)
```

```
# counting values:
per.island <- data.frame()
for (i in 1:length(shortpopnames)) {
  o <- sum(grepl("O", unlist(get(shortpopnames[i])), fixed = TRUE)) # counting row O's
  e <- sum(grepl("E", unlist(get(shortpopnames[i])), fixed = TRUE)) # counting row E's
  na <- sum(is.na(get(shortpopnames[i]))) # counting row NA's
  vec <- as.vector(c(shortpopnames[i], nrow(get(shortpopnames[i])), o, e, na))
  per.island <- rbind(per.island, vec)
}
rm(o, e, na, vec, i)

colnames(per.island) <- c("island", "specimens.based.on", "o.freq", "e.freq", "na.freq") # renaming col
str(per.island)
per.island$o.freq <- as.integer(per.island$o.freq) # changing the numbers from characters to integers
per.island$e.freq <- as.integer(per.island$e.freq)
per.island$na.freq <- as.integer(per.island$na.freq)

# calculating percentages:
perc.o <- round(as.vector((per.island$o.freq / (per.island$o.freq + per.island$e.freq)) * 100), digits
per.island$perc.o <- perc.o
perc.missing <- round(as.vector((per.island$na.freq / (per.island$o.freq + per.island$e.freq + per.islan
per.island$perc.missing <- perc.missing

rm(perc.o, perc.missing, het, hom)
rm(list = shortpopnames)
```

```
getwd()
write.csv(per.island, "./results/heterozygosity_testing_results_table.csv", row.names = FALSE)
```

Considering island size and distance from mainland

```
island.km <- read.csv("./data/island_size_data.csv", header = TRUE)
head(island.km)
island.km <- island.km[,c(2, 4)]
island.km <- island.km[order(island.km$ISLAND, decreasing = FALSE),] # sorting alphabetically
```

```
distance.from.ML <- data[,c(1,8,11,12)]
distance.from.ML <- distance.from.ML[!duplicated(distance.from.ML$island.1),] # keeping only 1 coordina
distance.from.ML <- distance.from.ML[order(distance.from.ML$island.1, decreasing = FALSE),] # sorting a
row.names(distance.from.ML) <- seq(nrow(distance.from.ML)) # renaming row numbers to be sequential
distance.from.ML # checking

#shortening some of the long island names to match what is already in the island.km df:
distance.from.ML[1,2] <- "Aotea"
distance.from.ML[20,2] <- "Rakiura"
distance.from.ML[22,2] <- "Rimatuu"

#decided to keep Cambodia as the mainland coordinates, will use this as the base for distance from main
distance.from.ML <- distance.from.ML[,-1] # removing specimen ID column since no longer necessary

library(geosphere)
km.from.ML <- vector()
for (i in 1:nrow(distance.from.ML)) {
  x <- distGeo(as.vector(distance.from.ML[13,c(3,2)]), as.vector(distance.from.ML[i,c(3,2)]))
  km.from.ML <- append(km.from.ML, x)
}

km.from.ML <- km.from.ML / 1000 # converting from metres to kilometres
distance.from.ML$km.from.ML <- km.from.ML # adding the kms to the distance df

rm(i, x, km.from.ML)
```

```
names(island.km)
names(distance.from.ML)
island.km <- merge(island.km, distance.from.ML, by.x = "ISLAND", by.y = "island.1", all = TRUE)

rm(distance.from.ML)

island.km$ISLAND
per.island$island
per.island$island <- c(
  "Aotea",
  "Borneo",
  "Great Mercury Island",
  "Halmahera",
  "Hatutaa",
  "Honuea",
  "Kaikura Island",
  "Kamaka",
  "Kayangel",
  "Late Island",
  "Luzon",
  "Malenge",
  "Mohotani",
  "Motukawanui",
  "New Britain",
  "New Guinea",
  "Normanby Island",
  "Rakiura",
  "Reiono",
  "Rimatuu",
  "Slipper Island",
  "Doubtful Sound",
  "Sulawesi",
  "Tahanea",
  "Wake Island",
  "Mainland"
) # editing the names to match those in the other df so I can merge them

per.island <- merge(per.island, island.km, by.x = "island", by.y = "ISLAND", all = TRUE)
```

```
getwd()
write.csv(per.island, "./results/heterozygosity_testing_results_table.csv", row.names = FALSE)
```

Linear regression and plots

```
per.island <- read.csv("./results/RStudio_Heterozygosity/heterozygosity_testing_results_table.csv", hea
```

Issues to consider before/during the statistical analyses:

- Some islands may be outliers because they are based on too few specimens (e.g. due to removal because of too many missing SNPs)

- Some islands may have specimens that are closely related to each other if they were sampled from the same site therefore may not accurately represent the population
- Distance from the mainland may not be the best measure because it doesn't indicate the difficultly of reaching the island in all cases, from example Normanby isl. is right next to New Guinea which is very large and likely has a diverse population that can easily move to Normanby and back.

a. Distance

```
LM <- lm(perc.o ~ km.from.ML, data = per.island) # model
summary(LM) # model results
par(mfrow = c(2, 2)) # changing the number of plots visible at once
plot(LM) # diagnostic plots
```

Results:

Residuals:

Min	1Q	Median	3Q	Max
-14.4165	-1.6867	0.6933	2.6053	9.1871

Coefficients:

	Estimate	Std. Error	t value	p-value
(Intercept)	8.249e+01	2.156e+00	38.262	< 2e-16 ***
km.from.ML	8.927e-04	2.470e-04	3.614	0.00139 **

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

Residual standard error: 5.149 on 24 degrees of freedom
Multiple R-squared: 0.3525
Adjusted R-squared: 0.3255
F-statistic: 13.06 on 1 and 24 DF, p-value: 0.001387

- Higher t-value is generally more significant, related to p-value.
- Multiple R-squared indicates ~35% of the variance is explained by the model
- Significant p-value(s), can reject H0 where x and y are not correlated.

Notes on diagnostic plots

- Residuals vs. Fitted plot shows a line not quite horizontal, but close enough to flat consider the relationship linear. Questionable points: 10 (Kayangel), 18 (New Guinea), 19 (Normanby Island).
- Normal QQ plot shows residuals generally in line, indicating normality. Questionable points are again: 10 (Kayangel), 18 (New Guinea), 19 (Normanby Island).
- Scale-Location plot is used to indicate constant residual variance with a line that does not trend up or down overall. Here it may be flat or trend down however it is not entirely clear due to the questionable points (10 (Kayangel), 18 (New Guinea), 19 (Normanby Island)) pulling a section upwards.

- Residuals vs Leverage (Cook's distance) plot assesses outliers. None of the points are over 1 (Cook's distance line which would make them statistical outliers) nor are any over 0.5 (which would make them questionable). Point 13 (Mainland) has high leverage, however this makes sense because the distance to the Mainland from the Mainland is 0, which is an unusually low number in this model.

```
ggplot(data = per.island, aes(x = km.from.ML, y = perc.o)) +
  geom_point(colour = "red", size = 3) +
  geom_smooth(method = 'lm', se = TRUE, level = 0.95) +
  ggtitle("Effect of Island distance from Mainland on Homozygosity") +
  xlab("Distance from Mainland (km)") + ylab("Homozygosity (%)") +
  theme_light()
```

```
## making column in df without labels for the points which are within the confidence interval
x <- predict(LM, interval = "confidence")
per.island$c <- ifelse(per.island$perc.o < (x[,"upr"]) & per.island$perc.o > x[,"lwr"], "", as.characte
per.island[3,12] <- "D.S." # making these two names shorter so they don't overlap in the plot
per.island[23,12] <- "S.I."

ggplot(per.island, aes(x = km.from.ML, y = perc.o)) +
  geom_point(size = 3, colour = "red") +
  geom_smooth(method = 'lm') +
  geom_text(aes(label = c), hjust = 0.5, vjust = -0.5) +
  ggtitle("Effect of Island distance from Mainland on Homozygosity") +
  xlab("Distance from Mainland (km)") + ylab("Homozygosity (%)") +
  theme_light()
```

b. Area

```
LM2 <- lm(perc.o ~ log10(area_km2), data = per.island)
summary(LM2)
# trying log10 of the area since there's a cluster of small islands. Bigger islands "squeezed" more tha
# log10 relationship might make sense because the effect of islands size decreases as the islands get b
par(mfrow = c(2, 2)) # changing the number of plots visible at once
plot(LM2) # diagnostic plots
par(mfrow = c(1, 1))
MASS::boxcox(LM2) # checking the recommended transformation (on the model w/o a transformation)

plot(perc.o ~ log10(area_km2), data = per.island)
abline(coef = coef(LM2), col = 4, lwd = 2)
```

Results:

Call: lm(formula = perc.o ~ log10(area_km2), data = per.island)

Residuals:

Min	1Q	Median	3Q	Max
-11.4512	-1.0882	0.5662	2.0754	8.7053

Coefficients:

	Estimate	Std. Error	t value	p-value
(Intercept)	93.1164	1.2963	71.834	< 2e-16 ***
log10(area_km2)	-1.5162	0.4345	-3.489	0.00198 **

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

Residual standard error: 4.649 on 23 degrees of freedom (1 observation deleted due to missingness)
Multiple R-squared: 0.3461
Adjusted R-squared: 0.3177
F-statistic: 12.17 on 1 and 23 DF, p-value: 0.001979

Notes on diagnostic plots

- Residuals vs. Fitted plot shows a line not quite horizontal, but close enough to flat consider the relationship linear. Points 3, 18, 19 are more extreme and give the residuals a slight cone shape, indicating potentially non-constant variance!
- Normal QQ plot shows residuals generally in line, indicating normality. Questionable points are once again: 3, 18, 19
- Scale-Location plot is used to indicate constant residual variance with a line that does not trend up or down overall. Line trends slightly downwards! Questionable points: 3, 18, 19
- Residuals vs Leverage (Cook's distance) plot assesses outliers. Point 18 (New Guinea) is over the 0.5 Cook's line and high leverage and therefore is noteworthy. This could be explained by New Guinea being the largest island area in the dataset/model.
- The Box-Cox plot indicates a transformation to perform and here lambda lines up almost perfectly with 0 == log transformation, which supports the other results.

```
x <- as.data.frame(predict(LM2, interval = "confidence"))
# Next 3 lines are re-adding row 13 NA's (not calculated in the above line) otherwise the names added t
x[nrow(x) + 1,] <- NA
row.names(x)[26.1] <- 13
x <- x[order(as.numeric(rownames(x))),]

per.island$c <- ifelse(per.island$perc.o < x[,"upr"] & per.island$perc.o > x[,"lwr"], "", as.character(

ggplot(data = per.island, aes(x = log10(area_km2), y = perc.o)) +
  geom_point(colour = "red", size = 3) +
  geom_smooth(method = 'lm', se = TRUE, level = 0.95) +
  geom_text(aes(label = c), hjust = 0.5, vjust = -0.8) +
  ggtitle("Effect of Island Area on Homozygosity") +
  xlab("log Area of Island (log10 km^2)") + ylab("Homozygosity (%)") +
  theme_light()
```

c. Both Distance and Area

```
par(mfrow = c(1, 1))
testLM <- lm(log(area_km2) ~ km.from.ML, data = per.island)
plot(log(area_km2) ~ km.from.ML, data = per.island)
```

```r
abline(coef = coef(testLM), col = 4, lwd = 2)
summary(testLM)
```

The above plot shows that generally the area of the islands decreases with distance (e.g. Borneo and New Guinea are large, Reiono and Tahanea are small) which I believe is a combination of sampling issues and how the islands are naturally on the Pacific. The correlation between these two variables is an issue with only simple linear models because I cannot test the affect of each variable compared with each other because their effect is similar (collinearity in this case).

```r
LM3 <- lm(perc.o ~ km.from.ML + log10(area_km2), data = per.island)
summary(LM3)
par(mfrow = c(2, 2))
plot(LM3)

mctest::mctest(LM3) # tests for collinearity (detected but 3/6 tests)
```

Call: lm(formula = perc.o ~ km.from.ML + log10(area_km2), data = per.island)

Residuals:

Min	1Q	Median	3Q	Max
-11.9825	-1.3484	0.6022	1.9063	7.5237

Coefficients:

	Estimate	Std. Error	t value	p-value
(Intercept)	89.9723695	3.6496351	24.652	<2e-16 ***
km.from.ML	0.0002988	0.0003241	0.922	0.3665
log10(area_km2)	-1.1566839	0.5848702	-1.978	0.0606 .

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

Residual standard error: 4.664 on 22 degrees of freedom (1 observation deleted due to missingness)
Multiple R-squared: 0.3704
Adjusted R-squared: 0.3132
F-statistic: 6.473 on 2 and 22 DF, p-value: 0.006157

- 25 observations - 3 parameters = 22 degrees of freedom
- The R^2 value indicates that only 37% of the variance is explained (31% when considering the extra parameter added (adjusted R^2), however this is not particularly different from the R^2adj of the other 2 models)
- The f-test is significant, therefore the the model explains a significant amount of variance
- The diagnostics for this model do not appear different from the diagnostics of the above 2 models

```r
ggplot(data = per.island, aes(x = km.from.ML, y = perc.o)) +
  geom_point(mapping = aes(colour = log10(area_km2)), size = 3) +
  geom_smooth(method = 'lm', se = TRUE, level = 0.95) +
  scale_color_gradient(low = "red", high = "blue") +
  ggtitle("Effect of Island distance from Mainland on Homozygosity, including island area") +
  xlab("Distance from Mainland (km)") + ylab("Homozygosity (%)") +
  theme_light()

# fix legend title and labels
```