Canis lupus and Canis lupus familiaris Population Structure Comparision

Julia Harvie

12/12/2020

GitHub repository: github.com/JuliaHarvie/CanislupusPopulationInvestigation

Introduction

In a 2016 article published in the journal Nature, the scientific name used for the domestic dog is *Canis lupus familiaris* (Suraci et al. 2016). However, in a 2015 article also published in Nature, the domestic dog is referred to as *Canis familiaris* (Tonoike et al. 2015). This is just one of many examples of the lack of consensus even peer reviewed sources as to the correct naming convention for the taxa whose common name is dog. At first this may appear to be a trivial issues, as either version will still identify the taxa as dog. But besides the obvious recording keeping issues associated with one taxa having two names, there is the problem that in accordance with the International Code of Zoological Nomenclature (ICZ) the two versions of the name carry two very different meaning (International Commission on Zoological Nomenclature, n.d.). According to ICZN the naming convention *C. lupus familiaris* indicates it is a subspecies of *C. lupus* (wolf) . Where as the naming convention identifies the taxa *C. familiaris* is its own species that belongs to the genus *Canis* along side *C. lupus*.

Real life applications that are influenced by this distinction are conservation efforts and DNA biomonitoring. Severe inbreeding depression has been found in some *C. lupus* populations and if *C. lupus framiliaris* is a subspecies of *C. lupus* individuals from *C. lupus familiaris* could be used for breeding plans to decrease the inbreeding depression well still maintaining a genome that is purely *Canis lupus*. The alternative argument being if *C. lupus framiliaris* has progressed far enough down the path of evolution that it is its own species, individuals from that population should not be used for *C. lupus* conservation work. If DNA sequenced based biomnitoring is going to be performed on a *C. lupus population*, it maybe be important to distinguish if a sample testing positive for *C. lupus* came specifically from *C. lupus familiaris*. Knowing the level of allelic differences in their respective genomes will indicate how difficult making this distinction using sequence may be.

This experiment will define and compare the population structures of *C. lupus* and *C. lupus familiaris* using Cytochrome c oxidase I (CO1) sequences mined from the Barcode of Life Database (BOLD) (Hebert and Ratnasingham 2007). These sequences will be filtered according to sequence length and base quality and then aligned. Population statistics will be calculated for these aligned sequences, including Nei's Gst and Jost's Das well as a AMOVA. Finally the structure of the sequence variance will be visualized through a principal components analysis (PCA). The null hypothesis is that there may be some detectable population structure among *Canis lupus familiars* but it is still just a sub population or sub species of *Canis lupus* that still experience genetic drift due to gene flow from the rest of the population. The alternative hypothesis to this is that *Canis lupus familiaris* not only has a distinct population structure but it is significantly variant from the population structure of *Canis lupus* and this should be acknowledged through identifying individuals of said population as *Canis familiaris*. For the remainder of this report the naming convention *Canis lupus familiaris* will be used in accordance with the null hypothesis.

The Data set

All sequence data used for this experiment was obtained from the publicly available data on BOLD on December 12, 2020. This initial BOLD query was for all database records. belonging to the genius *Canis*, for any marker type, and all available metadata for the records was included. This yielded a data set of 1828 records with 79 possible metadata fields, and one DNA sequence. This data set was then reduced to only contain records produced using the CO1 marker that were labeled as belonging to the species *Canis lupus*. The number of fields in this reduced data set was also restricted to only include 6 pieces of metadata and the sequence data. This produced a data set of 1685 records, with 7 associated fields, spanning one species and 5 subspecies.

Code Section 1 –Data Acquisition, Exploration,Filtering, and Quality Control

Confirmation the initial data acquisition and filtering occurred as expected

species_name	n
Canis familiaris	4
Canis lupus	1681

subspecies_name	n
Canis lupus chanco	3
Canis lupus desertorum	1
Canis lupus familiaris	254
Canis lupus laniger	2
Canis lupus lupus	3
NA	1422

markercode	n
COI-5P	1685

Confirmation records were renamed as expected

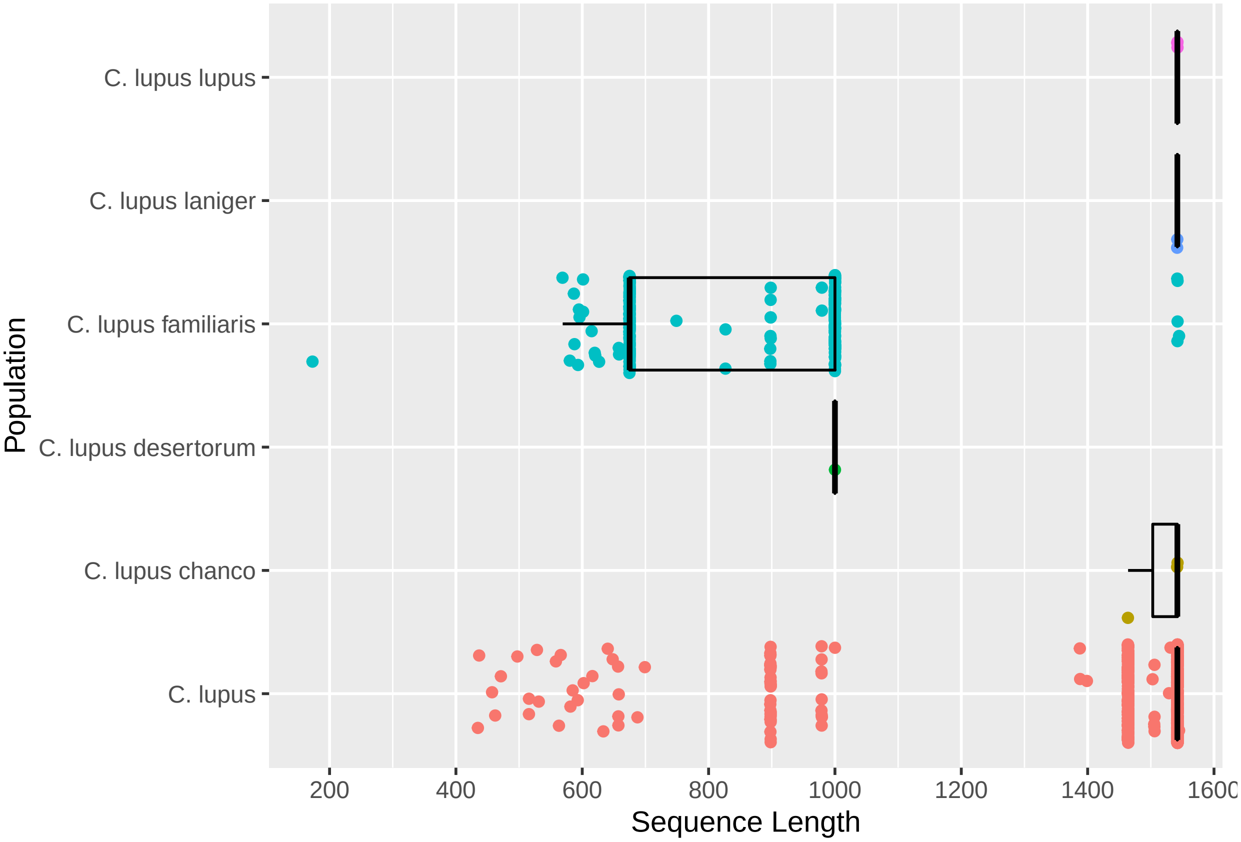
species_name	n
C. lupus	1685

subspecies_name	n
C. lupus	1418
C. lupus chanco	3
C. lupus desertorum	1
C. lupus familiaris	258
C. lupus laniger	2
C. lupus lupus	3

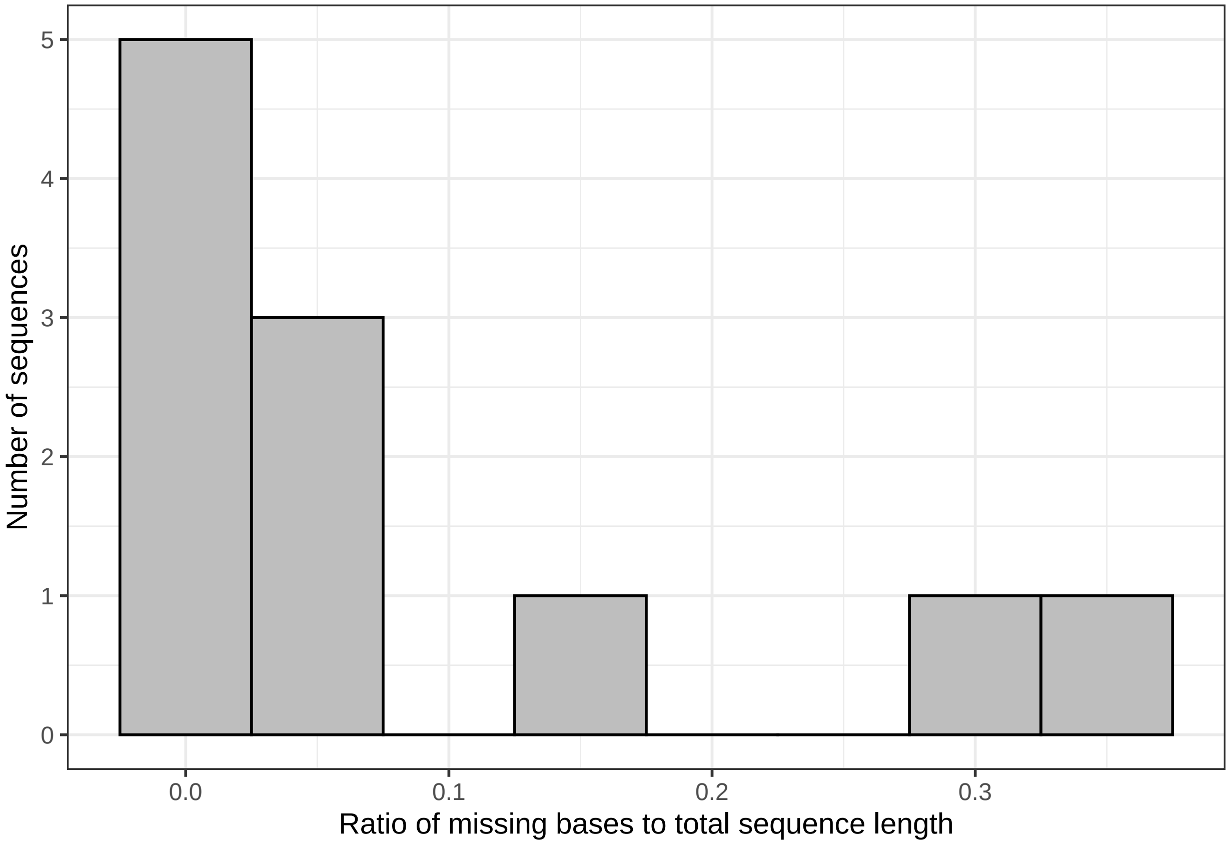
Confirmation no NA sequences remain and sequence data is appearing as expected

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     173    1464    1542    1390    1542    1545
```

Jitter plot of sequence length in the data set. From this a sequence length range of 600 - 1600 was chosen to filter over.



Histogram depicting the number of sequences that have over 1% ambiguous bases. There are some present in this data set, but they are not numerous so all sequences with >1% ambiguous bases will be filtered out of the data set to improve overall sequence quality.



Visualization of the the representation of the distribution of the populations after all quality filtering steps have occurred

subspecies_name	n
C. lupus	1393
C. lupus chanco	3
C. lupus desertorum	1
C. lupus familiaris	248
C. lupus laniger	2
C. lupus lupus	3

Due to the very low entries remaining for the subspecies *C. lupus chanco*, *C. lupus desertorum*, *C. lupus laniger* and *C. lupus lupus* they will all also be removed from the data set as their sample size is too small to perform any meaningful statistical analysis. There is still more than enough *C. lupus familiars* sequences to perform an accurate population structure analysis.

Confirmation the final filter worked as expected

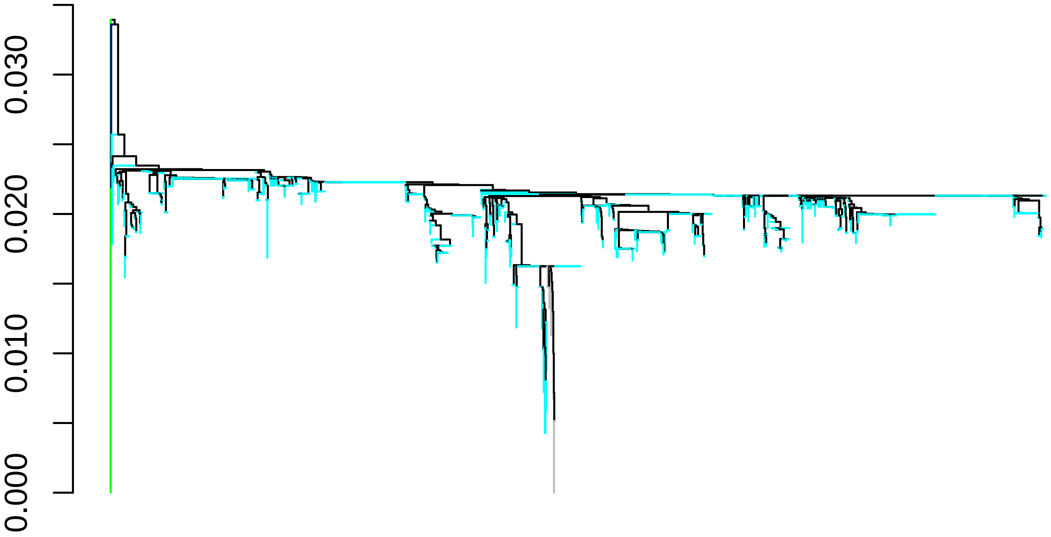
subspecies_name	n
C. lupus	1393
C. lupus familiaris	248

Confirmation sequences are stored correctly for the alighnment function to work.

```
## [1] "character"
## [1] "DNAStringSet"
## attr(,"package")
## [1] "Biostrings"
```

Dendrogram produced from the distance matrix of the aligned sequence in to see if the data set contains any potentially mislabeled or reverse compliment sequences.

```
## ================================================================================
```



```
##
## Time difference of 9.59 secs
```

cluster	n
1	1
2	1
3	1625
4	14

ID	genbank_accession	species_name	subspecies_name
C. lupus_3599865	JF342908	C. lupus	C. lupus

One sequence is clearly visible as an outlier. Its associated genebank accession was used to search it in NCBI's BLAST. Results of 99% similarity to *Canis lupus* were returned so it is most likely not a mislabeled sequence. The distance between it and the nearest cluster is not larger enough to suspect the sequence is a reverse compliment either. Therefor it is assumed outlier is a actual representation of large variation in population and will be included in the analysis.

Main Software Tools Description

The main resource for the analysis section of this experiment was the vignette "Population Differentiation for Sequence Data" by Margarita M. López- Uribe (Author) and Zhian N. Kamvar (edits) (López- Uribe and Kamvar, n.d.). The workflow described in this vignette was chosen as a skeleton to build the analysis around as it begins with importing FASTA files containing aligned sequence data and shows the functions required to produce population genetic test statistics for said data. The strength of picking this vignette is that it starts with data in the same form my data is currently stored in, aligned FASTA files. Using the same starting point greatly increasing the likelihood I will be able to successfully recreate the pipeline with my own data. However a weakness of this vignette is it only shows how to produce the various test statistics, not how to choose the one that will be best suited for a specific data set. When performing a statistical analysis choosing an appropriate test statistic is critical. Just because a data set can be used to calculate a test statistic, does not mean the output will offer any meaningful description of the data set. I will do additional research on the test statistics demonstrate to select the ones appropriate for my data set. In addition I will reference the “An introduction to adegenet 2.0.0” vignette by Thibaut Jombart for direction on how to create figures to visulaize the results the first vignette will help me produce (Jombart 2015).

Code Section 2 –Main Analysis

Table 10: Number of alleles present per loci

	x
19	2
171	2
201	2
276	2
298	2
352	2
396	2
507	2
523	2
589	2
700	2
711	2
714	2
717	3
720	2
744	2
822	2
909	2
922	2
954	2
1053	3
1107	2
1122	2
1170	2
1206	2
1262	2
1281	2
1363	2
1378	2
1392	2
1416	2
1456	2
1506	2
1512	2
1515	2
1534	2

All but one loci is biallelic therefor Nei's Gst will be an appropriate statistic to use for this analysis as opposed to Hedrick's Gst which is suited for multiallelic data sets (Kane n.d.). Another situation in which Nei's Gst does not offer great resolution of population structure is when working with markers with high mutation rates such as microsatellites. Again, not super relevant for this data set as CO1 has a slower mutation rate by comparison, but the Jost's D will also be looked to confirm this assumption is accurate (Kane n.d.).

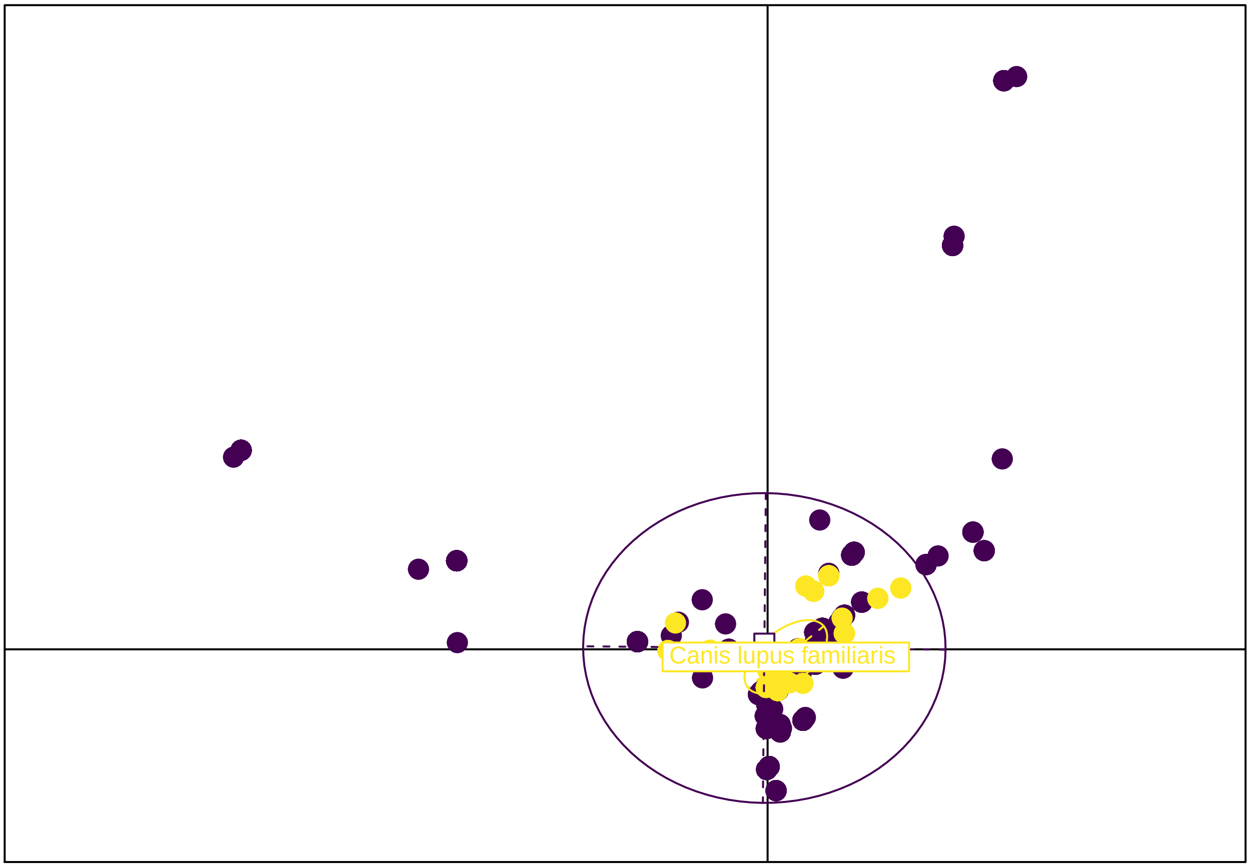
Table 11: Table 1. Summary Population test Staistics.

	x
Hs	0.0861642
Ht	0.0938044
Gst_est	0.0814487
Gprime_st	0.1648315
D_het	0.0167213
D_mean	NA

Table 12: Table 2. AMOVA of C. Lupus.

	SSD	MSD	df
Pop	0.0000744	7.44e-05	1
Error	0.0068688	4.20e-06	1639
Total	0.0069432	4.20e-06	1640

PCA of the CO1 sequence variation within *Canis lupus* (purple) and *Canis lupus familiaris* (yellow).



Discussion

Table one shows the global population has a Nei's Gst (Gst_est) of between 0.05 and 0.15 indicative of moderate allelic differences between the subpopulations. A low Jost's D (D_het) of 0.017 indicates the two populations share a majority of their alleles (Jost et al. 2018). The AMOVA in table two attributes the majority of total sequence variation to within individuals, not among populations, 0.0069 and 0.0007 respectively. This is the expected result if populations do not differ significantly. The results of the AMOVA are supported by the PCA. Only a single cluster was produced, not two as would be expected if the population were significantly different. Multiple points labeled *C. lupus* were placed further away from other *C. lupus* points, than any *C. familiaris* points were. Therefor, there is not enough evidence it reject the null hypothesis that there may be some detectable population structure among *Canis lupus familiars* but it is still just a sub population or sub species of *Canis lupus* that still experience genetic drift do to gene flow from the rest of the population. The conclusion that dogs and wolves should not be considered two difference species may seem surprising to a random individual without a genetic or evolutionary background, as colloquially dogs are referred to as being their own species distinct from wolves. However from a molecular biology perspective it is not too shocking that *Canis lupus familiaris* has not evolved enough from the previously established *Canis lupus* population to be bale to maintain its own population with unique fixed alleles. The earliest estimates of when the *Canis lupus familiaris* population began to be established through domestication only date it at approximately 33,000 years ago (Wayne and Vonholdt 2012). However estimations of how long it takes a population to become fixed as a unique, distinguishable species start around one million years (Uyeda et al. 2011)

There are a few weakness with the experimental design that should be addressed. First off, the sample size of *Canis lupus familiaris* was an order of magnitude less than was used for *Canis lupus* and there was not enough metadata to determine how similar the individuals sampled for *Canis lupus familiaris* were. Including more *Canis lupus familiaris* sequences and ensuring the cover a broad geographic and physiological range would give more support to the results. Some population structure was detected among *Canis lupus familiaris*, but not enough to be considered significantly different. It would have been more informative if the level of population structure show by *Canis lupus familiaris* could have been compared to the levels shown but other *Canis lupus* subspecies. This type of comparison could not be performed due to lack of samples to perform a meaningful statistically analysis on the other subspecies. In addition to increased sampling efforts, another future direction this investigation could go would be different marker selection. The marker chosen was a mitochondrial marker with a moderate mutation rate. A genomic marker, especially one identified as being linked to domestication traits, may shave experienced more intense allele fixation and there for indicate larger differences in population structure. If the goal of a study was to perform *Canis lupus* DNA biomonitoring as discussed in the introduction, a high mutation region like a microsatellite may still offer sufficient resolution.

#Acknowledgments

All code was written solely by me with help from the lecture materials. All outside help was found online and referenced accordingly.

References

Hebert, Paul D N, and Sujeevan Ratnasingham. 2007. "The Barcode of Life Data System BOLD." *Molecular Ecology Notes*, no. 7: 355–64. https://doi.org/10.1111/j.1471-8286.2006.01678.x.

International Commission on Zoological Nomenclature. n.d. "International Commission on Zoological Nomenclature." https://www.iczn.org/.

Jombart, Thibaut. 2015. "An introduction to adegenet 2.0.0." *R Package*. http://adegenet.r-forge.r-project.org/files/tutorial-basics.pdf.

Jost, Lou, Frederick Archer, Sarah Flanagan, Oscar Gaggiotti, Sean Hoban, and Emily Latch. 2018. "Differentiation measures for conservation genetics." *Evolutionary Applications* 11 (7): 1139–48. https://doi.org/10.1111/eva.12590.

Kane, Nolan. n.d. "Should I use FST, G'ST or D? |." Accessed December 13, 2020. https://www.molecularecologist.com/2011/03/should-i-use-fst-gst-or-d-2/.

López- Uribe, Margarita M., and Zhian N. Kamvar. n.d. "Population Differentiation for Sequence Data." https://popgen.nescent.org/PopDiffSequenceData.html%7B/#%7Dcontributors.

Suraci, Justin P., Michael Clinchy, Lawrence M. Dill, Devin Roberts, and Liana Y. Zanette. 2016. "Fear of large carnivores causes a trophic cascade." *Nature Communications* 7. https://doi.org/10.1038/ncomms10698.

Tonoike, Akiko, Miho Nagasawa, Kazutaka Mogi, James A. Serpell, Hisashi Ohtsuki, and Takefumi Kikusui. 2015. "Comparison of owner-reported behavioral characteristics among genetically clustered breeds of dog (Canis familiaris)." *Scientific Reports* 5 (May): 1–11. https://doi.org/10.1038/srep17710.

Uyeda, Josef C., Thomas F. Hansen, Stevan J. Arnold, and Jason Pienaar. 2011. "The million-year wait for macroevolutionary bursts." *Proceedings of the National Academy of Sciences of the United States of America* 108 (38): 15908–13. https://doi.org/10.1073/pnas.1014503108.

Wayne, Robert K., and Bridgett M. Vonholdt. 2012. "Evolutionary genomics of dog domestication." Springer. https://doi.org/10.1007/s00335-011-9386-7.

Appendix

Below is all of the R scripted used to generate the above report. All lines have been commented out so that the code does not run twice when knitting the pdf together.

```
# ```{r setup, include=FALSE} knitr::opts_chunk$set(echo = TRUE, cache = TRUE,
# tidy= TRUE) library(Biostrings) library(tidyverse) library(muscle)
# library(DECIPHER) library(ape) library(viridis) library(cluster)
```

```
# library(seqinr) library(adegenet) library(pegas) library(apex) library(mmod)
# library(poppr) ```

# ```{r Read in data, include=FALSE} #The first time the script is run the two
# uncommented commands must be used. Afterwards to save time the results can just
# be read into R but uncommenting the third command # Initial query pull from
# BOLD Canis <-
# read_tsv('http://www.boldsystems.org/index.php/API_Public/combined?taxon=Canis&format=tsv')
# #Write to disk for future use write_tsv(Canis, 'BoldCanis_data.tsv') #Reading
# back in saved query #Canis <- read_tsv('BoldCanis_data.tsv') ```

# ```{r Reduce data, include=FALSE} #Curate BOLD data to only the species and
# fields required for this analysis #Need to make sure any C. lupus familiaris
# records uploaded under Cani familiaris are also included in this reduced data
# set Canis_rd <- Canis %>% select(species_name, subspecies_name, country,
# markercode, nucleotides, recordID, genbank_accession) %>% filter(species_name
# == 'Canis lupus' | species_name == 'Canis familiaris') %>% filter(markercode ==
# 'COI-5P') ```

# ```{r Check1, echo=FALSE} #Confirm pre-filtering worked as intended by
# visualizing all the fields that remained.  # The function count is masked by
# another library so must specify dplyr::count knitr::kable(
# dplyr::count(Canis_rd, species_name) ) knitr::kable( dplyr::count(Canis_rd,
# subspecies_name) ) knitr::kable( dplyr::count(Canis_rd, markercode) ) ```

# ``` {r Unify naming convention, include=FALSE} #This data set has records under
# both naming schemes. Keeping with the null hypothesis any Canis familiaris will
# be relabeled Canis lupus familiars and assigned as a subspecies.  #In addition
# to remove the NA from the subspecies_name field records that do not have an
# associated subspecies are labeled as just Canis lupus to indicate this for (n
# in 1:nrow(Canis_rd)){ if (Canis_rd[n,'species_name'] == 'Canis familiaris') {
# Canis_rd[n,'species_name'] <- 'Canis lupus' Canis_rd[n,'subspecies_name'] <-
# 'Canis lupus familiaris' } else if (is.na(Canis_rd[n,'subspecies_name'])){
# Canis_rd[n,'subspecies_name'] <- 'Canis lupus' } } #Using the naming convention
# C. instead of Canis cleans up the labels for use in figures down the pipeline
# Canis_rd <- Canis_rd %>% mutate(species_name = str_replace(species_name,
# 'Canis', 'C.')) %>% mutate(subspecies_name = str_replace(subspecies_name,
# 'Canis', 'C.')) ```

# ```{r Check2, echo=FALSE} #Check knitr::kable( dplyr::count(Canis_rd,
# species_name) ) knitr::kable( dplyr::count(Canis_rd, subspecies_name) ) ```

# ```{r Quality Control, include=FALSE} #Remove Ns (ambiguous bases) from
# beginning and end of the sequences, remove any records who do not have a
# sequence associated with them and assign every entry in the remaining data set
# a unique label for identification.  Canis_filtered <- Canis_rd %>%
# mutate(nucleotides2 = str_remove_all(nucleotides, '^N+|N+$|-')) %>%
# filter(!is.na(nucleotides2)) %>% mutate(ID = paste(subspecies_name, recordID,
# sep='_')) ```

# ``` {r, Check3, echo=FALSE} #Check summary(nchar(Canis_filtered$nucleotides2))
# ```
```

```
# ```{r Check Sequence Length, echo=FALSE} #Jitter plot showing the distribution
# of sequence length according to subspecies. DOne on the subspecies level to
# make sure filters are selected that filter fairly over all subspecies
# ggplot(Canis_filtered,aes(x=nchar(nucleotides2), y = subspecies_name, colour =
# subspecies_name), xmin = 0, xmax = max(nchar(nucleotides2))+100) +
# scale_x_continuous(breaks =seq(0, max(nchar(Canis_filtered$nucleotides2))+100 ,
# by = 200)) + geom_jitter(show.legend = F) + geom_boxplot(outlier.shape = NA,
# colour='black', show.legend = F, fill = NA) + xlab('Sequence Length') +
# ylab('Population') #Based off above plot a range of 600-1600 was selected for
# their sequence length Canis_filtered <- Canis_filtered %>%
# filter(nchar(nucleotides2) >= 600) %>% filter(nchar(nucleotides2) <= 1600) ```

# ``` {r Check Sequence Quality, echo=FALSE} #Create column representing percent
# of sequence made up of ambiguous bases Canis_filtered <- Canis_filtered %>%
# mutate(Undefined = str_count(nucleotides2, 'N')/nchar(nucleotides2)) LowQuality
# <- filter(Canis_filtered, Undefined > 0.01) ggplot(LowQuality,
# aes(x=Undefined)) + geom_histogram(binwidth = 0.05, fill='gray', color='black')
# + labs(x='Ratio of missing bases to total sequence length', y = 'Number of
# sequences') + theme_bw() #Conclude this is a good value to filter at
# Canis_filtered <- Canis_filtered %>% filter(Undefined < 0.01) ```

# ```{r Check4, echo=FALSE} knitr::kable( dplyr::count(Canis_filtered,
# subspecies_name) ) ```

# ```{r Remove low entry subspecies, include=FALSE} Canis_filtered <-
# Canis_filtered %>% filter(subspecies_name == 'C. lupus' | subspecies_name ==
# 'C. lupus familiaris') ```

# ``` {r Check5, echo=FALSE} knitr::kable( dplyr::count(Canis_filtered,
# subspecies_name) ) ```

# ```{r Alighnment prep, echo=FALSE} #Make sure the data is stored as the
# required class Canis_filtered <- as.data.frame(Canis_filtered) #Confirm
# print(class(Canis_filtered$nucleotides2)) #Transform for alignment function
# Canis_filtered$nucleotides2 <- DNAStringSet(Canis_filtered$nucleotides2)
# #Confirm print(class(Canis_filtered$nucleotides2)) #Assign unique labels to
# every sequence names(Canis_filtered$nucleotides2) <- Canis_filtered$ID ```

# ``` {r Alighnment, include=FALSE} Due to the size of the data set will set a
# max time of 1 hour just in case Canis_alignment <-
# DNAStringSet(muscle::muscle(Canis_filtered$nucleotides2, maxhours = 1),
# use.names = TRUE) #Export the alignment as a FASTA for use further down the
# pipeline writeXStringSet(Canis_alignment, file = 'CanisAlignment.fasta') #If
# alignment has been created, comment out above lines and just read it in using
# below #Read <- read.FASTA('CanisAlignment.fasta') ```

# ```{r Quality Check of Alighnment, echo=FALSE} #Canis_Bin <-
# as.DNAbin(Canis_alignment) Canis_Bin <- Read #Create a distance matrix based
# off of aligned sequences. default evolutionary model of K80 will be used. Due
# to the variation in sequence length pairwise deletion will be used.
# Canis_distanceMatrix <- dist.dna(Canis_Bin, model = 'k80', as.matrix = TRUE,
# pairwise.deletion = TRUE) #Create a dendrogram Canis_clusters <-
# IdClusters(Canis_distanceMatrix, method = 'NJ', cutoff = 0.02, showPlot = TRUE,
```

```
# type = 'both') #Identify which cluster the outlier belongs to and then extract
# it from the data frame for investigation #Identify which cluster it was
# assigned to knitr::kable( dplyr::count(Canis_clusters[[1]], cluster) ) #Filter
# by cluster checkID <- filter(Canis_clusters[[1]], cluster == 1)
# Canis_filtered$nucleotides2 <- as.character(Canis_filtered$nucleotides2) BLAST
# <- filter(Canis_filtered, ID == row.names(checkID)) #Acquire genbank accession
# knitr::kable( BLAST[,c('ID', 'genbank_accession', 'species_name',
# 'subspecies_name')] ) ```

# ```{r Build geneid object, include=FALSE} #Even though multiple FASTAs are not
# needed to be read in, using this function gives the object the class of
# multiFASTA which is required for another function down the pipeline.
# Canis_Multi <- read.multiFASTA('CanisAlignment.fasta') Canis_genind <-
# multidna2genind(Canis_Multi) #Need to assign each sequence to a population. For
# this experiment the populations of interest can be defiend according to
# subspecies designation.  strata(Canis_genind) <- data.frame('Pop' =
# Canis_filtered$subspecies_name) setPop(Canis_genind) <- ~Pop ```

# ```{r Allele count, echo=FALSE} #Check how many alleles are present at each
# loci knitr::kable( Canis_genind@loc.n.all, caption = 'Number of alleles present
# per loci' ) ```

# ```{r Test stats, echo=FALSE} knitr::kable( diff_stats(Canis_genind)$global,
# caption = 'Table 1. Summary Population test Staistics.'  ) ```

# ```{r AMOVA, echo=FALSE} Canis_DistPair <- dist.multidna(Canis_Multi, pool = T)
# Canis_AMOVA <- pegas::amova(Canis_DistPair ~ Pop, data = strata(Canis_genind),
# nperm = 100) knitr::kable( Canis_AMOVA$tab, caption = 'Table 2. AMOVA of C.
# Lupus.'  ) ```

# ```{r Create PCA, echo=FALSE} Canis_scaled <- scaleGen(Canis_genind,
# NA.method='mean') Canis_PCA <-
# dudi.pca(Canis_scaled,cent=FALSE,scale=FALSE,scannf=FALSE,nf=3)
# s.class(Canis_PCA$li, Canis_genind@pop,col=viridis(2), axesell=T, cstar=0,
# cpoint=2, grid=FALSE, clabel= 0.75, label = c(' ', 'Canis lupus familiaris '))
# ```
```