

# Dragon Phylogeny

Alyssa Green

2023-03-01

Link to GitHub (<https://github.com/alyssagreen02/BIOL432-Week7-DragonPhylogeny>)

## Load library and read nexus file

```
library(ape)
DragonNexus=read.nexus.data("DragonMatrix.nex")
head(DragonNexus)
```

```

## $`0.1FishXXX`
## [1] "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0"
## [20] "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0"
## [39] "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0"
## [58] "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "0"
## [77] "0" "0"
##
## $`0.2SnakeXX`
## [1] "1" "1" "1" "1" "0" "0" "0" "0" "1" "1" "1" "0" "0" "1" "1" "1" "0" "1" "1"
## [20] "0" "0" "0" "0" "0" "0" "0" "1" "1" "0" "0" "0" "0" "0" "1" "0" "0" "0"
## [39] "1" "0" "0" "0" "1" "0" "0" "0" "1" "0" "0" "0" "0" "0" "1" "1" "1" "1"
## [58] "1" "1" "0" "0" "0" "0" "0" "0" "0" "0" "1" "0" "0" "0" "0" "0" "0" "0"
## [77] "0" "0"
##
## $`0.3MammalX`
## [1] "1" "0" "0" "0" "0" "0" "0" "0" "0" "0" "1" "1" "1" "0" "0" "0" "0" "0"
## [20] "0" "0" "0" "1" "0" "0" "0" "1" "1" "1" "0" "0" "0" "0" "1" "1" "0" "0"
## [39] "1" "1" "0" "0" "1" "1" "0" "0" "0" "0" "0" "0" "0" "0" "1" "0" "0" "0"
## [58] "0" "0" "0" "1" "1" "0" "0" "0" "0" "0" "1" "0" "0" "0" "0" "1" "0" "0"
## [77] "0" "0"
##
## $`1GermanXXX`
## [1] "0" "1" "0" "0" "1" "1" "1" "0" "0" "1" "1" "1" "1" "0" "0" "0" "0" "1" "1"
## [20] "0" "0" "0" "0" "1" "0" "1" "1" "1" "1" "1" "0" "0" "0" "0" "1" "1" "0" "1"
## [39] "0" "0" "0" "0" "1" "1" "0" "1" "1" "0" "0" "0" "0" "0" "0" "1" "1" "1" "0"
## [58] "0" "0" "0" "1" "0" "0" "0" "1" "0" "0" "1" "0" "0" "0" "0" "0" "0" "1"
## [77] "1" "1"
##
## $`2FrenchXXX`
## [1] "0" "1" "0" "0" "1" "1" "0" "1" "0" "1" "1" "0" "0" "1" "1" "1" "0" "1" "0"
## [20] "0" "0" "0" "0" "0" "0" "1" "1" "1" "1" "1" "0" "0" "0" "0" "1" "1" "0" "0"
## [39] "1" "0" "0" "0" "1" "1" "0" "0" "0" "0" "0" "0" "0" "0" "1" "1" "0" "0"
## [58] "0" "0" "1" "0" "0" "1" "0" "0" "0" "1" "1" "0" "0" "0" "1" "1" "0" "0"
## [77] "1" "0"
##
## $`3FrenchXXX`
## [1] "0" "1" "0" "0" "1" "1" "1" "0" "0" "0" "1" "1" "0" "0" "0" "0" "0" "0"
## [20] "0" "0" "1" "0" "1" "0" "0" "1" "1" "1" "1" "0" "0" "0" "0" "1" "1" "0" "1"
## [39] "1" "1" "0" "1" "1" "1" "0" "1" "0" "0" "0" "0" "0" "0" "1" "1" "1" "0"
## [58] "0" "0" "0" "1" "0" "0" "0" "1" "0" "0" "1" "0" "0" "0" "1" "1" "0" "0"
## [77] "1" "1"

```

# Code my dragons and add them to the nexus file

```
P=list
('1','0','0','1','1','1','1','1','0','1','0','0','1','0','0','1','1','0','0','0','0','1','0','0',
'0','0','0','1','1','0','0','1','1','0','0','0','0','0','0','0','1','1','0','?','?','?','?'
,'?','?','0','0','0','1','0','0','0','0','0','0','0','1','1','1','0','0','0','0','0','0','1','0'
,'0','1','1','0','1','0')

G=list
('1','0','0','1','0','1','1','1','0','1','?','?','?','?','0','1','0','0','1','0','0','0','0','0'
,'1','0','0','1','1','1','0','1','0','0','0','1','1','0','1','1','0','0','0','0','1','1','0','0'
,'0','0','1','1','0','0','1','0','?','?','?','?','?','?','?','?','?','?','?','?','?','?','0','1'
,'1','1','1','1','0','0')

R=list
('1','0','1','0','0','1','1','1','1','0','0','0','0','0','1','1','0','0','0','0','0','0','0','1'
,'0','0','1','1','0','1','0','0','0','0','0','0','1','0','0','0','1','1','0','0','0','0','0','0'
,'1','1','0','0','0','0','0','?','?','?','?','0','1','0','0','0','0','0','?','?','?','0','0','0'
,'1','1','0','0','1')

P=as.character(P)
G=as.character(G)
R=as.character(R)
PGR=list(P,G,R)
DragonNexusFinal=c(DragonNexus, PGR)

names(DragonNexusFinal)[78]="75PinkXXXX"
names(DragonNexusFinal)[79]="76GreenXXX"
names(DragonNexusFinal)[80]="77RainbowX"
```

## Export nexus file and save to input folder

```
write.nexus.data(DragonNexusFinal, "./input/DragonNexusFinal.nexus")
```

## Convert nexus file to data frame

```
DragonNexusDF=data.frame(matrix(unlist(DragonNexusFinal), ncol=78, byrow=T))
```

```
## Warning in matrix(unlist(DragonNexusFinal), ncol = 78, byrow = T): data length
## [6239] is not a sub-multiple or multiple of the number of rows [80]
```

```
row.names(DragonNexusDF)=names(DragonNexusFinal)
head(DragonNexusDF)
```

##	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19
## 0.1FishXXX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
## 0.2SnakeXX	1	1	1	1	0	0	0	0	1	1	1	0	0	1	1	1	0	1	1
## 0.3MammalX	1	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
## 1GermanXXX	0	1	0	0	1	1	1	0	0	1	1	1	1	0	0	0	0	1	1
## 2FrenchXXX	0	1	0	0	1	1	0	1	0	1	1	0	0	1	1	1	0	1	0
## 3FrenchXXX	0	1	0	0	1	1	1	0	0	0	1	1	0	0	0	0	0	0	?
##	X20	X21	X22	X23	X24	X25	X26	X27	X28	X29	X30	X31	X32	X33	X34	X35	X36		
## 0.1FishXXX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
## 0.2SnakeXX	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0		
## 0.3MammalX	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	1	1		
## 1GermanXXX	0	0	0	0	1	0	1	1	1	1	1	0	0	0	0	1	1		
## 2FrenchXXX	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	1	1		
## 3FrenchXXX	?	?	1	0	1	0	0	1	1	1	1	0	0	0	0	1	1		
##	X37	X38	X39	X40	X41	X42	X43	X44	X45	X46	X47	X48	X49	X50	X51	X52	X53		
## 0.1FishXXX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
## 0.2SnakeXX	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0		
## 0.3MammalX	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0		
## 1GermanXXX	0	1	0	0	0	0	1	1	0	1	1	0	0	0	0	0	0		
## 2FrenchXXX	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0		
## 3FrenchXXX	0	1	1	1	0	1	1	1	0	1	0	0	0	0	0	0	0		
##	X54	X55	X56	X57	X58	X59	X60	X61	X62	X63	X64	X65	X66	X67	X68	X69	X70		
## 0.1FishXXX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
## 0.2SnakeXX	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0		
## 0.3MammalX	1	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0		
## 1GermanXXX	1	1	1	0	0	0	0	1	0	0	0	1	0	0	1	0	?		
## 2FrenchXXX	1	1	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0		
## 3FrenchXXX	1	1	1	0	0	0	0	1	0	0	0	1	0	0	1	0	0		
##	X71	X72	X73	X74	X75	X76	X77	X78											
## 0.1FishXXX	0	0	0	0	0	0	0	0											
## 0.2SnakeXX	0	0	0	0	0	0	0	0											
## 0.3MammalX	0	0	0	1	0	0	0	0											
## 1GermanXXX	?	?	?	0	0	1	1	1											
## 2FrenchXXX	0	1	1	0	0	1	1	0											
## 3FrenchXXX	0	1	1	0	0	1	1	1											

## Input weights data

```
weightsDat=read.csv("Weights.csv")
```

## create a single vector of weights

```
Weights=paste0(weightsDat$Weight,collapse = "")
Weights=strsplit(Weights, split = "")[[1]]
```

## Convert each letter to a value

```
WeightsNum=rep(NA,length(Weights))
for (i in 1:length(WeightsNum)) {
  if(Weights[i] %in% LETTERS){
    WeightsNum[i]=which(LETTERS==Weights[i])+9
  } else {
    WeightsNum[i]=Weights[i]
  }
}

WeightsNum=as.numeric(WeightsNum)
```

```
length(WeightsNum)
```

```
## [1] 78
```

## Multiply weight value by the trait vector for each dragon

```
WtDragonNexus=DragonNexusFinal
for (i in 1:length(DragonNexusFinal)) {
  RepWeight=DragonNexusFinal[[i]]==1
  WtDragonNexus[[i]][RepWeight]=WeightsNum[RepWeight]
  RepWeight=NA
}
```

```
## Warning in WtDragonNexus[[i]][RepWeight] <- WeightsNum[RepWeight]: number of
## items to replace is not a multiple of replacement length
```

## Calculate the distance matrix

```
WtDragonNexusDF=data.frame(matrix(unlist(WtDragonNexus), ncol=78,byrow=T))
```

```
## Warning in matrix(unlist(WtDragonNexus), ncol = 78, byrow = T): data length
## [6239] is not a sub-multiple or multiple of the number of rows [80]
```

```
row.names(WtDragonNexusDF)=names(WtDragonNexus)
WtDragonDist=dist(WtDragonNexusDF, method = "euclidean")
```

```
## Warning in dist(WtDragonNexusDF, method = "euclidean"): NAs introduced by  
## coercion
```

## Images of Dragons



Citation: Classic Pink Dragon. (2019). Dragon Shield. Retrieved March 1, 2023, from [https://twitter.com/dragonshield\\_/status/1186643372796268547](https://twitter.com/dragonshield_/status/1186643372796268547) ([https://twitter.com/dragonshield\\_/status/1186643372796268547](https://twitter.com/dragonshield_/status/1186643372796268547)).



Citation: Sandara. (2015). Green Dragon v2. Deviant Art. Retrieved March 1, 2023, from <https://www.deviantart.com/sandara/art/Green-Dragon-v2-579637253> (<https://www.deviantart.com/sandara/art/Green-Dragon-v2-579637253>).





Citation: Christou, A. (n.d.). Rainbow Dragon. A Christou Art. Retrieved March 1, 2023, from <https://www.achristouart.com/rainbow-dragon> (<https://www.achristouart.com/rainbow-dragon>).

## Plot Tree

```
library(ggtree)
```



```
## ggtree v3.6.2 For help: https://yulab-smu.top/treedata-book/
##
## If you use the ggtree package suite in published research, please cite
## the appropriate paper(s):
##
## Guangchuang Yu, David Smith, Huachen Zhu, Yi Guan, Tommy Tsan-Yuk Lam.
## ggtree: an R package for visualization and annotation of phylogenetic
## trees with their covariates and other associated data. Methods in
## Ecology and Evolution. 2017, 8(1):28-36. doi:10.1111/2041-210X.12628
##
## LG Wang, TTY Lam, S Xu, Z Dai, L Zhou, T Feng, P Guo, CW Dunn, BR
## Jones, T Bradley, H Zhu, Y Guan, Y Jiang, G Yu. treeio: an R package
## for phylogenetic tree input and output with richly annotated and
## associated data. Molecular Biology and Evolution. 2020, 37(2):599-603.
## doi: 10.1093/molbev/msz240
##
## Guangchuang Yu, Tommy Tsan-Yuk Lam, Huachen Zhu, Yi Guan. Two methods
## for mapping and visualizing associated data on phylogeny using ggtree.
## Molecular Biology and Evolution. 2018, 35(12):3041-3043.
## doi:10.1093/molbev/msy194
```

```
##
## Attaching package: 'ggtree'
```

```
## The following object is masked from 'package:ape':
##
##      rotate
```

```
WtDragonTree=fastme.bal(WtDragonDist)
ggtree(WtDragonTree, layout="circular")
```

```
## ! The tree contained negative edge length. If you want to ignore the edge, you
## can set `options(ignore.negative.edge=TRUE)`, then re-run ggtree.
```



## Get tip labels by country

```
Country=gsub("[0-9\\.]+(^[^X]+)x*", "\\1", WtDragonTree$tip.label)
```

## Isolate tip labels for my dragons

```
CountryGroups=split(WtDragonTree$tip.label, Country)
Groups=list(CountryGroups$PinkXXXX, CountryGroups$GreenXXX, CountryGroups$RainbowX)
names(Groups)[1]="75PinkXXXX"
names(Groups)[2]="76GreenXXX"
names(Groups)[3]="77RainbowX"
```

## Make plotting data

```
WtDTcol=groupOTU(WtDragonTree, Groups)
```

```
ggtree(WtDTcol, layout="circular", aes(colour=group))+geom_tiplab(size=1, aes(angle=angle))
```

```
## ! The tree contained negative edge length. If you want to ignore the edge, you
## can set `options(ignore.negative.edge=TRUE)`, then re-run ggtree.
```

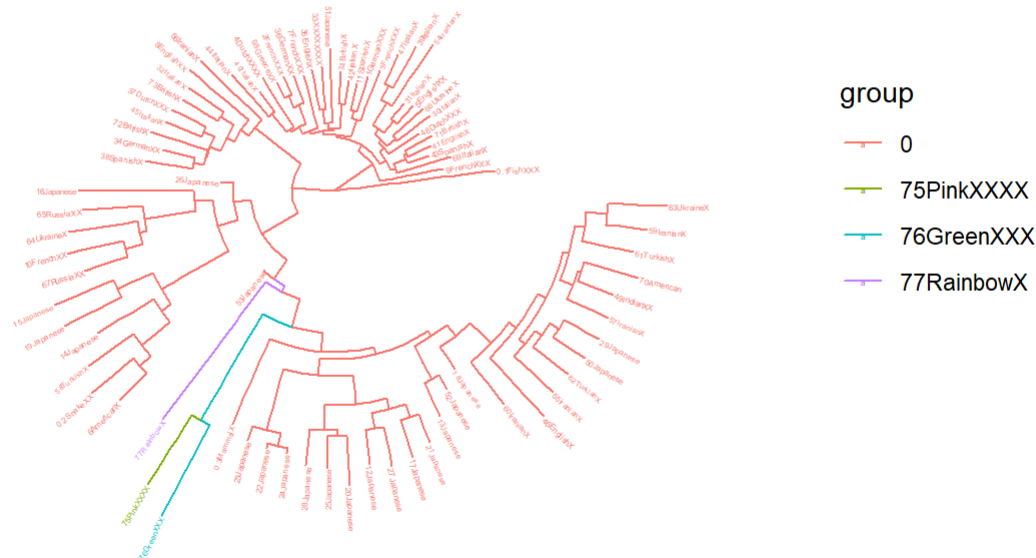


Figure 1: Dragon phylogeny. The phylogeny represents the relationship between dragons based on their phenotypic characteristics. The phylogeny is coloured coded to highlight the three added dragons, with blue for dragon 76GreenXXX, green for dragon 75PinkXXXX and purple for dragon 77RainbowX. The colour orange, labelled 0, denotes all other dragons.

## Backstory and Data Interpretation

The pink, green, and rainbow dragons are highly elusive and rarely witnessed. We collected data on these dragons' phenotypic characteristics using documentation from rare civilian sightings. From this data, we created a phylogeny, mapping the pink, green, and rainbow dragons relative to those from the known dragon database. The information we received from this phylogeny was very intriguing. The pink, green, and rainbow dragons are members of the same clade, with the rainbow dragon deviating first from the ancestral population. The pink and green dragons deviated from the ancestral population after the rainbow dragon, simultaneously as one species, later diverging into separate species. The pink, green, and rainbow dragons are most similar to those known to originate from Japan and therefore are closely related to the Japanese dragons. The phylogeny also shows that the pink, green, and rainbow dragons are more closely related to the mammal outgroup than the fish and snake outgroups. This study is significant as it provides insight into our planet's little-known species of dragons.