

Dragon_Phy_Assignment

Joel Sagman

Load necessary packages

```
library(ape)
library(ggplot2)
library(reshape2)
library(ggtree)
```

```
## ggtree v3.2.1 For help: https://yulab-smu.top/treedata-book/
##
## If you use ggtree in published research, please cite the most appropriate paper(s):
##
## 1. Guangchuang Yu. Using ggtree to visualize data on tree-like structures. Current Protocols in Bioinformatics
. 2020, 69:e96. doi:10.1002/cpbi.96
## 2. Guangchuang Yu, Tommy Tsan-Yuk Lam, Huachen Zhu, Yi Guan. Two methods for mapping and visualizing associate
d data on phylogeny using ggtree. Molecular Biology and Evolution. 2018, 35(12):3041-3043. doi:10.1093/molbev/msy
194
## 3. Guangchuang Yu, David Smith, Huachen Zhu, Yi Guan, Tommy Tsan-Yuk Lam. ggtree: an R package for visualizati
on and annotation of phylogenetic trees with their covariates and other associated data. Methods in Ecology and E
volution. 2017, 8(1):28-36. doi:10.1111/2041-210X.12628
```

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

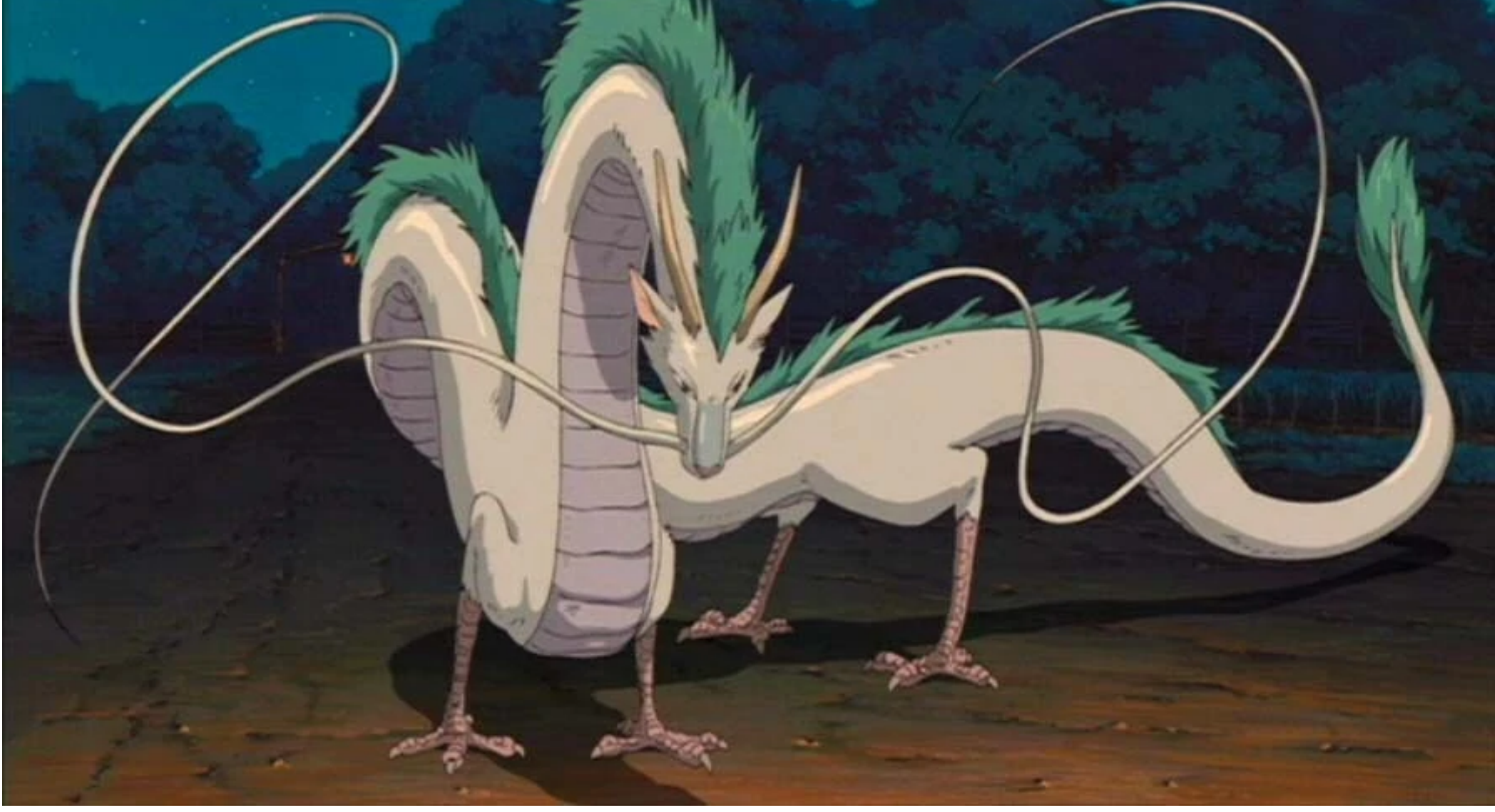
```
## The following object is masked from 'package:ape':
##
##      rotate
```

Images of added dragons



Villentretenmerth (Golden Dragon)

[Source](#)



Haku

[Source](#)



Kaidou

[Source](#)

Import Data

```
DragonNexus<-read.nexus.data("Input/DragonMatrix.nex") # Imports dragon nexus file that has been edited to contain the new dragons
tail(DragonNexus) # Shows that tail end of the file for confirmation purposes
```

```
## $`72BritishX`
## [1] "0" "1" "1" "0" "0" "1" "1" "0" "0" "1" "1" "1" "0" "0"
## [20] "1" "0" "0" "1" "0" "1" "0" "0" "1" "1" "0" "0" "1" "0" "1"
## [39] "1" "1" "0" "1" "1" "0" "1" "1" "0" "1" "0" "0" "1" "1" "0"
## [58] "0" "0" "1" "0" "0" "1" "0" "1" "0" "0" "1" "2" "2" "2" "0" "0" "0"
## [77] "0" "1"
##
## $`73BritishX`
## [1] "0" "1" "1" "0" "0" "1" "1" "0" "1" "0" "0" "1" "1" "0" "0" "0"
## [20] "1" "0" "0" "0" "1" "0" "0" "1" "1" "1" "0" "1" "0" "1" "1" "0" "0"
## [39] "1" "1" "0" "0" "1" "1" "0" "1" "1" "0" "1" "0" "1" "1" "1" "0" "0"
## [58] "0" "0" "0" "0" "1" "0" "1" "0" "0" "1" "2" "2" "2" "2" "0" "0" "1"
## [77] "0" "0"
##
## $`74BritishX`
## [1] "0" "1" "0" "0" "1" "1" "1" "0" "0" "1" "1" "0" "1" "1" "1" "0" "0"
## [20] "1" "0" "0" "0" "1" "0" "0" "1" "1" "1" "1" "0" "1" "1" "1" "1" "0" "0"
## [39] "1" "1" "1" "0" "1" "1" "0" "1" "1" "1" "0" "1" "0" "1" "1" "0" "0"
## [58] "0" "0" "0" "0" "1" "0" "1" "0" "0" "1" "2" "2" "2" "2" "0" "0" "1"
## [77] "1" "1"
##
## $`75HakuXXXX`
## [1] "1" "0" "0" "1" "0" "1" "1" "1" "1" "0" "0" "1" "0" "0" "1" "0" "0" "1"
## [20] "1" "0" "1" "0" "0" "1" "0" "1" "1" "1" "0" "1" "1" "0" "0" "0" "0" "1"
## [39] "1" "1" "1" "0" "0" "0" "0" "0" "0" "0" "0" "1" "1" "1" "1" "0" "0" "0"
## [58] "0" "0" "0" "1" "1" "0" "0" "0" "0" "0" "0" "1" "1" "1" "0" "0" "2"
## [77] "2" "2"
##
## $`76KaidouXX`
## [1] "1" "1" "0" "1" "1" "1" "1" "1" "1" "0" "0" "0" "0" "1" "0" "0" "1"
## [20] "0" "0" "1" "0" "0" "0" "0" "1" "1" "1" "1" "0" "0" "0" "0" "0" "0" "1"
## [39] "1" "0" "1" "0" "0" "0" "0" "0" "0" "0" "0" "1" "1" "1" "0" "2" "1"
## [58] "0" "0" "0" "1" "1" "0" "0" "0" "0" "0" "0" "1" "1" "1" "0" "0" "2"
## [77] "2" "2"
##
## $`77GoldenXX`
## [1] "1" "0" "0" "1" "0" "1" "1" "1" "0" "1" "1" "0" "0" "0" "1" "0" "0"
## [20] "0" "0" "0" "0" "0" "0" "1" "1" "0" "0" "1" "0" "1" "0" "0" "0" "0"
## [39] "1" "0" "1" "0" "0" "0" "2" "2" "2" "2" "2" "1" "0" "0" "1" "0" "0"
## [58] "0" "0" "0" "1" "1" "1" "1" "0" "0" "0" "1" "2" "2" "2" "0" "0" "1" "1"
## [77] "0" "0"
```

Importing Weight data and converting them into a single vector

```
WeightsDat<-read.csv("Data/Weights.csv") # Imports Weights so be applied to dragon data
Weights<-paste0(WeightsDat$Weight,collapse="") # Isolates just the weights from the data
Weights<-strsplit(Weights,split="")[[1]] # Splits up the the weights into individual characters
```

Converting each Weights into numerical values

```
WeightsNum<-rep(NA,length(Weights)) # create empty vector for weights numbers
for(i in 1:length(WeightsNum)){ # A loop that converts that weights from letters and number to just number that r
represents their weight
  if(Weights[i] %in% LETTERS){
    WeightsNum[i]<-which(LETTERS==Weights[i])+9
  } else {
    WeightsNum[i]<-Weights[i]
  }
}
WeightsNum<-as.numeric(WeightsNum) # Converts to numerical values for weight
```

Creates a new data frame that includes the weights

```
WtDragonNexus<-DragonNexus # Make a new weighted data frame object
for (i in 1:length(DragonNexus)){ # a for loop that applies the weights to the new weighted data frame
  RepWeight<-DragonNexus[[i]]==1
  WtDragonNexus[[i]][RepWeight]<-WeightsNum[RepWeight]
  RepWeight<-NA
}
```

Calculating the distance matrix with weighted data

```
WtDragonNexusDF<-data.frame(matrix(unlist(WtDragonNexus),ncol= length(DragonNexus[[1]]),byrow=T)) # converts
row.names(WtDragonNexusDF)<-names(WtDragonNexus) #Adds the same row names as the previous data set
WtDragonDist<-dist(WtDragonNexusDF,method='euclidean') # Euclidean algorithm used to calculate distance with weig
hted values attached
```

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

Preping data for building tree

```
WtDragonTree<-fastme.bal(WtDragonDist) #
WtDragonTree$tip.label <- gsub("([1-9\\.\\.])([*X]*X*", "\\1\\2", WtDragonTree$tip.label) # Removes just the follow
ing X(s) from the tip label
```

```
WtDTclade<-groupClade(WtDragonTree,node=c(148,134,124, 111, 85, 122)) # Specifies clade groups
ggtree(WtDTclade,layout="fan",aes(colour=group)) + # Makes plot a plot where each clade is labeled and coloured
  geom_tiplab(size=2,aes(angle=angle)) +
  geom_cladelabel(node=148,label="Serpentidae",hjust=0.5,offset=15,offset.text=4,fontsize=3,angle=-47) +
  geom_cladelabel(node=134,label="Wyvernidae",hjust=0.5, offset=18, offset.text=4,fontsize=3,angle=10) +
  geom_cladelabel(node=87,label="Orientalia",hjust=0.5, offset=19, offset.text=4,fontsize=3,angle=40) +
  geom_cladelabel(node=111,label="Dracopterionidae",hjust=0.5, offset=18, offset.text=4,fontsize=3,angle=-55) +
  geom_cladelabel(node=124,label="Dracoverisidae",hjust=0.5, offset=17, offset.text=4,fontsize=3,angle=64) +
  geom_cladelabel(node=122,label="Dracohumandae",hjust=0.5, offset=16, offset.text=4,fontsize=3,angle=-85) +
  theme(legend.position="none")
```

```
## Scale for 'y' is already present. Adding another scale for 'y', which will
## replace the existing scale.
```

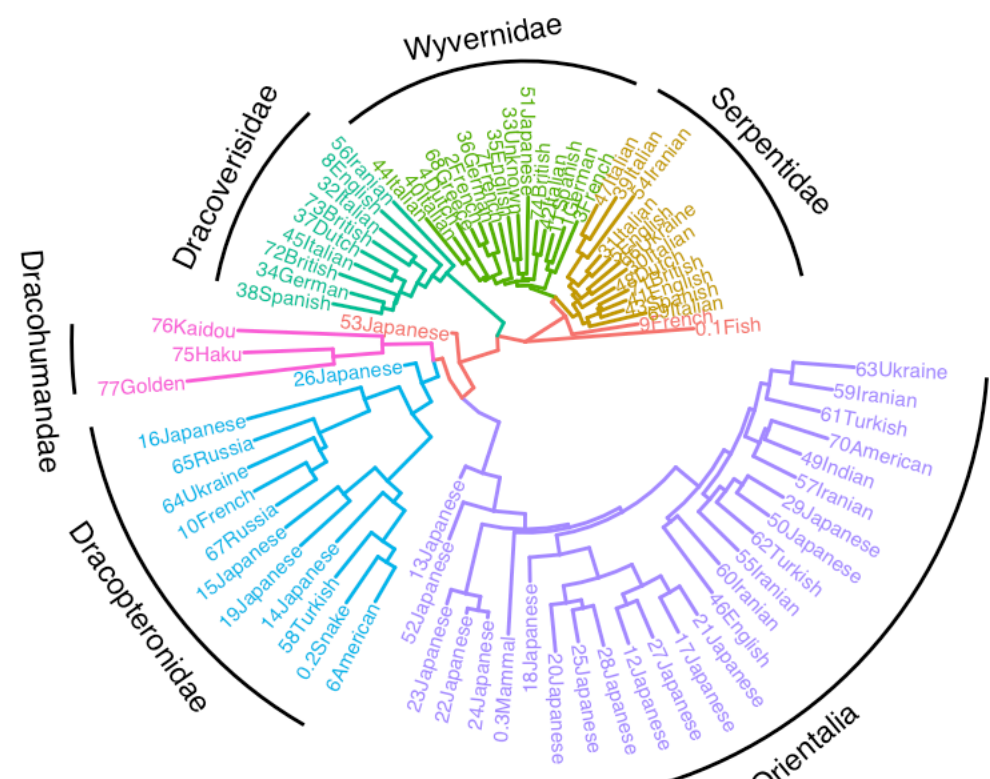


Figure 1. Dragon Phylogeny

Phylogenetic tree of dragons with dragons saparred into clades. Dragons separated into clades based on calculated evolutionary distance. Newly discovered Dracohumandae clade contains dragons with magical shiftshaping abilities.