Dragon Phylogeny Assignment

Jayde MacMillan 20068250

3/7/2022

Github - jaydealexandra (https://github.com/jaydealexandra/DragonPhylogeny.git)

Setup

Loads the required libraries.

```
library(ape)
library(reshape2)
library(ggplot2)
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 Bioinfor
matics. 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 ass
ociated data on phylogeny using ggtree. Molecular Biology and Evolution. 2018, 35(12):3041-3043. doi:10.109
3/molbev/msy194
## 3. Guangchuang Yu, David Smith, Huachen Zhu, Yi Guan, Tommy Tsan-Yuk Lam. ggtree: an R package for visua
lization and annotation of phylogenetic trees with their covariates and other associated data. Methods in E
cology and Evolution. 2017, 8(1):28-36. doi:10.1111/2041-210X.12628
```

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

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

```
library(ggimage)
```

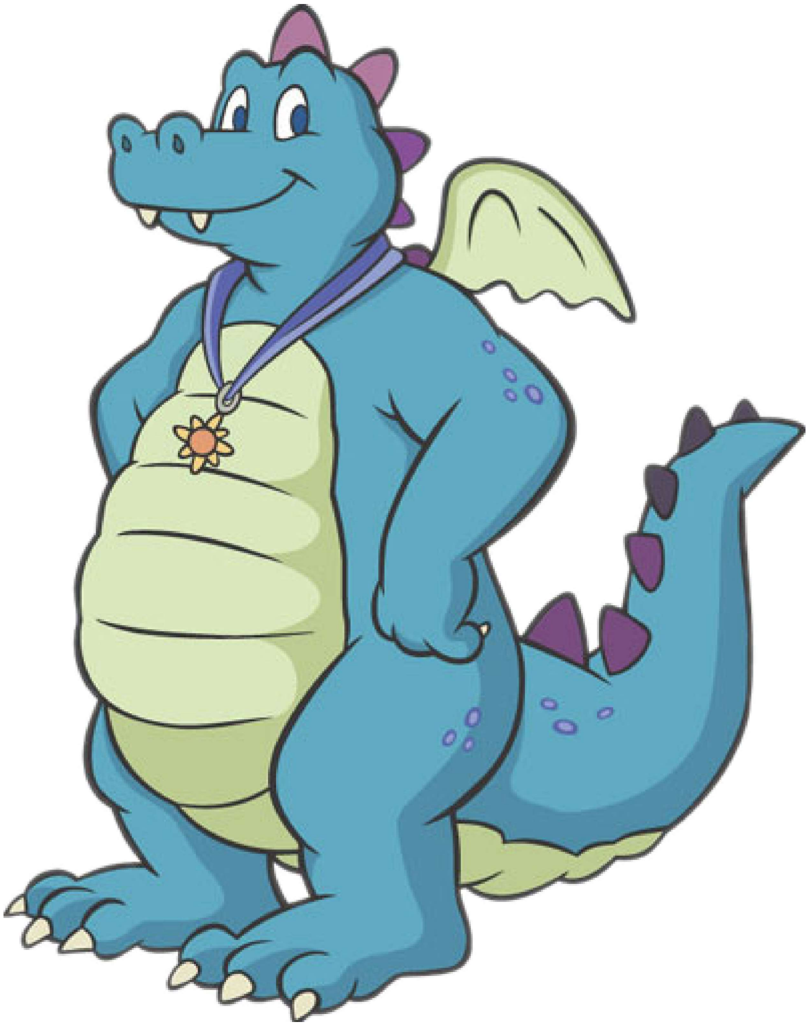
Imports ‘DragonMatrix.nex’ into an object called ‘DragonNexus’.

```
DragonNexus <- read.nexus.data("input/DragonMatrix.nex")
```

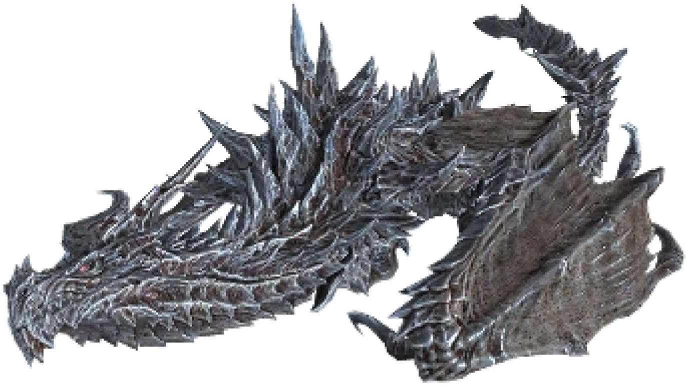
Dragons

The following dragons were added into the ‘DragonMatrix.nex’ dataset.

Ord from Dragon Tales. This image is from Dragon Tales Wikia (https://dragontales.fandom.com/wiki/Ord)



Alduin from Elder Scrolls. This image is from Skyrim Wiki (https://skyrim.fandom.com/wiki/Alduin)



Spyro from Spyro the Dragon. This image is from Wikipedia (https://en.wikipedia.org/wiki/Spyro_(character))

Weights

Imports ‘Weight.csv’ into a single vector and splits it into substrings.

```
WeightsDat<-read.csv("input/Weights.csv") #imports 'Weights.csv'
Weights<-paste0(WeightsDat$Weight,collapse="") #links vectors together after converting to character
Weights<-strsplit(Weights,split="")[[1]] #splits the character vector into substrings
```

Converts each letter to a value using a custom function.

```
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)
```

Multiplies the weight value by the trait vector for each dragon.

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

Distance Matrix

Uses the `unlist` function to turn the object into a vector, then converting it to a matrix.

```
WtDragonNexusDF<-data.frame(matrix(unlist(WtDragonNexus),ncol=78,byrow=T)) #unlists and creates a dataframe
row.names(WtDragonNexusDF)<-names(WtDragonNexus)  #sets the row names

WtDragonDist<-dist(WtDragonNexusDF,method='euclidean') #computes the distance between the rows of the data
 matrix using the euclidean method
```

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

```
WtDragonDistMat<-as.matrix(WtDragonDist) #creates a matrix using the weighted data
```

Tree Formatting

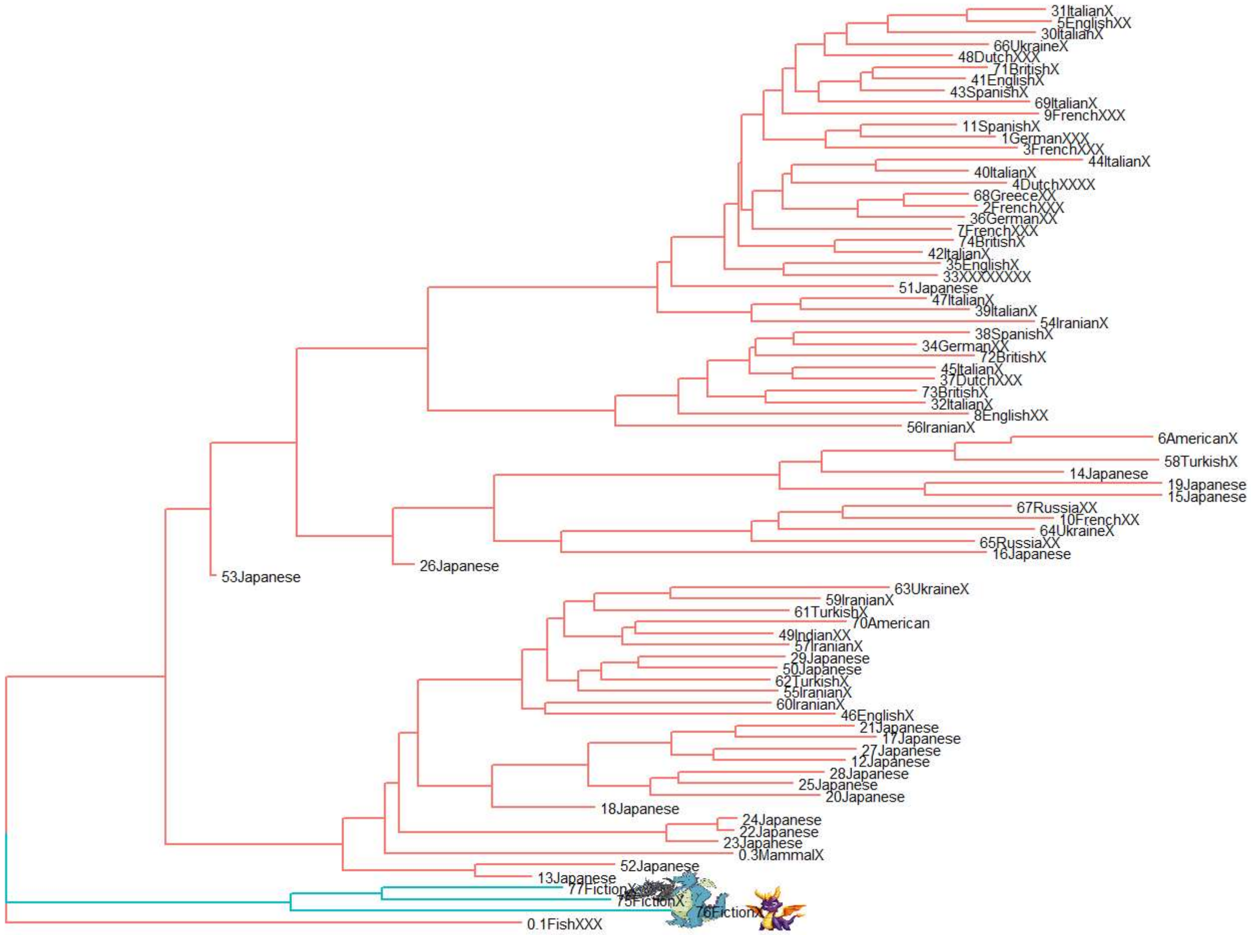
Builds a phylogeny.

```
WtDragonTree<-fastme.bal(WtDragonDist)
```

Colour by OTU.

```
base <- gsub("[0-9\\.]+([^X]+)X*","\\1", WtDragonTree$tip.label) # Remove leading numbers

baseGroups<-split(WtDragonTree$tip.label, base=="Fiction") # Splits 'WtDragonTree$tip.label' by Fiction

WtDTcol<-groupOTU(WtDragonTree, baseGroups) # Groups OTUs

imagesDF <- data.frame(node = c(78,79,80), #specifies nodes
                       images = c("images/A7_MACMILLAN_JAYDE_Ord.png", #specifies images
                                  "images/A7_MACMILLAN_JAYDE_Alduin.png",
                                  "images/A7_MACMILLAN_JAYDE_Spyro.png"))



ggtree(WtDTcol, branch.length='note') %<+% imagesDF +
  geom_tree(aes(colour=group)) +
  geom_tiplab(aes(image=images), geom = "image", offset=2, size = 0.05) +
  geom_tiplab(size=2) +
  theme(legend.position = "none") +
  xlim(0,50)
```

```
## Warning: Removed 77 rows containing missing values (geom_image).
```


31ItalianX
5EnglishXX
30ItalianX
66UkraineX
48DutchXXX
71BritishX
41EnglishX
43SpanishX
69ItalianX
9FrenchXXX
11SpanishX
1GermanXXX
3FrenchXXX
44ItalianX
40ItalianX
4DutchXXXX
68GreeceXX
2FrenchXXX
36GermanXX
7FrenchXXX
74BritishX
42ItalianX
35EnglishX
33XXXXXXXX
51Japanese
47ItalianX
39ItalianX
54IranianX
38SpanishX
34GermanXX
72BritishX
45ItalianX
37DutchXXX
73BritishX
32ItalianX
8EnglishXX
56IranianX
6AmericanX
58TurkishX
14Japanese
19Japanese
15Japanese
67RussiaXX
10FrenchXX
64UkraineX
65RussiaXX
16Japanese
53Japanese
26Japanese
63UkraineX
59IranianX
61TurkishX
70American
49IndianXX
57IranianX
29Japanese
50Japanese
62TurkishX
55IranianX
60IranianX
46EnglishX
21Japanese
17Japanese
27Japanese
12Japanese
28Japanese
25Japanese
20Japanese
18Japanese
24Japanese
22Japanese
23Japanese
0.3MammalX
52Japanese
13Japanese
77FictionX
75FictionX
76FictionX
0.1FishXXX