# Assignment 7 - Caroline Tang 20115082

### Project Info

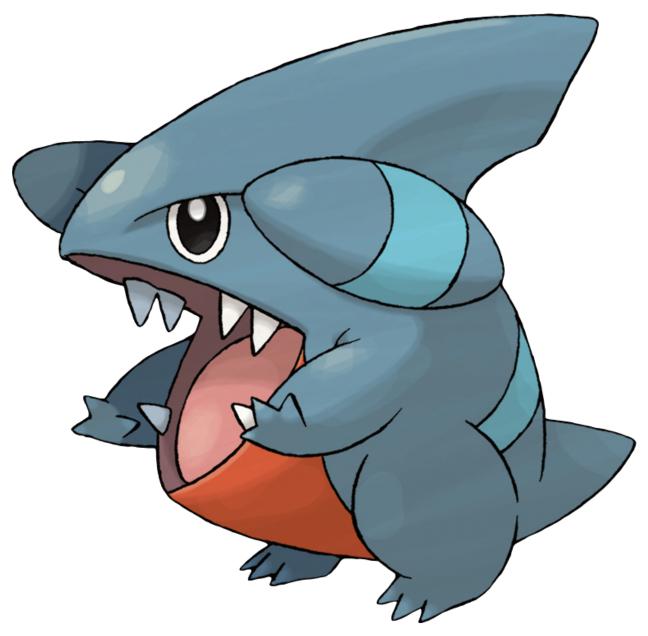
Github username: carolinetang77 Github repository Date: March 9, 2022

#### **Dragon Phylogeny**

#### New dragons

I added Gible, Goomy, and Appletun from the Pokemon series to the phylogeny. All photos and information were obtained from Bulbapedia.

## Gible



Original Gible Art (Source)

## Goomy



Original Goomy Art (Source)

### Appletun



Original Art (Source)

#### Creating the phylogeny

Load in all the 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 Bioi:
## 2. Guangchuang Yu, Tommy Tsan-Yuk Lam, Huachen Zhu, Yi Guan. Two methods for mapping and visualizing
## 3. Guangchuang Yu, David Smith, Huachen Zhu, Yi Guan, Tommy Tsan-Yuk Lam. ggtree: an R package for v

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

library(ggimage)
```

Load in dragon data and trait weights

```
dragonNexus <- read.nexus.data("A7_Tang_Caroline.nex")
weightDat <- read.csv("https://colauttilab.github.io/Data/Weights.csv")</pre>
```

Convert dragon data to weighted traits

```
#separate weights and convert them to a numeric scale (1-35 where A = 10 and Z = 35)
weights <- paste(weightDat$Weight, collapse = "")</pre>
weights <- unlist(strsplit(weights, split = ""))</pre>
numericWeight <- vector(length = length(weights))</pre>
for (i in 1:length(weights)) {
  if (weights[i] %in% LETTERS) {
    numericWeight[i] <- which(LETTERS == weights[i]) + 9</pre>
  } else {
    numericWeight[i] <- as.numeric(weights[i])</pre>
  }
}
#multiply each trait by the weight
wtDragonNexus <- dragonNexus
for (i in 1:length(dragonNexus)) {
  Repweight <- dragonNexus[[i]]==1</pre>
  wtDragonNexus[[i]] <- numericWeight * Repweight</pre>
}
#convert to dataframe
wtDragonDF <- data.frame(matrix(unlist(wtDragonNexus), ncol = length(wtDragonNexus[[1]]), byrow = T))</pre>
row.names(wtDragonDF) <- names(wtDragonNexus)</pre>
```

Calculate Euclidean distances and convert to a matrix

```
wtDragonDist <- dist(wtDragonDF, method = "euclidean")
wtDragonDM <- as.matrix(wtDragonDist)</pre>
```

Create the tree using neighbour joining methods

Create phylogenetic tree, added dragons are highlighted in red

```
ggtree(wtDragoncol, layout = "rectangular") %<+% dragonLabel +
  geom_tree(aes(colour = group)) +
  geom_tiplab(aes(label = lab), size = 1.5) +
  geom_tiplab(aes(image = image), geom = "image", offset = 2, size = 0.1) +
  xlim(0, 70) +
  scale_colour_manual(values = c("black", "red")) +
  theme(legend.position = "none")</pre>
```

