Organized_Code_Ancestral_Reconstruction

Saya Maratova

2025-05-27

STEP 1: Load libraries

```
library(ape)
library(phangorn)
```

STEP 2: Load aligned sequences

```
aligned <- read.phyDat("short_align_new.fas", format = "fasta", type = "AA")
```

STEP 3: Distance matrix and tree building (WAG model)

```
dm <- dist.ml(aligned, model = "WAG")
treeUPGMA <- upgma(dm)
treeNJ <- NJ(dm)
```

STEP 4: Tree visualization

```
par(mfrow=c(1,1))
plot(treeUPGMA, main = "UPGMA Tree", cex = 0.5, label.offset = 0.02)
plot(treeNJ, main = "Phylogenetic Tree NJ", cex = 0.5, label.offset = 0.02)
```

STEP 5: Bootstrap analysis with WAG model

```
bootstrap_function <- function(x) NJ(dist.ml(x, model = "WAG"))
bootstrap_analysis <- bootstrap.phyDat(aligned, FUN = bootstrap_function, bs = 100)

plotBS(bootstrap_function(aligned), bootstrap_analysis, p = 0, cex = 0.5, type = "phylogram")
```

STEP 6: Alignment visualization

```
image(aligned[1:10, 1:50], scheme = "Clustal", show.aa = TRUE)
```

STEP 7: ML tree fitting with WAG model

```
fit <- pml(treeUPGMA, data = aligned)
fit_opt <- optim.pml(fit, model = "WAG", rearrangement = "stochastic")
```

STEP 8: Midpoint root and label tree

```
tree <- fit_opt$tree
tree <- midpoint(tree)
tree <- makeNodeLabel(tree)
fit_opt_2 <- update(fit_opt, tree = tree)
plot(tree, show.node.label = T)
plot(tree, show.node.label = TRUE, type = "fan", cex = 0.5)
```

STEP 9: Ancestral state reconstruction

```
anc <- anc_pml(fit_opt_2)
```

Visualize ancestral states

```
plotAnc(anc, scheme = "Clustal")
plotAnc(anc, scheme = "Clustal", 2)
```

STEP 10: Gaps-as-state model

```
align21 <- gap_as_state(aligned)
align21
fit$call
fit21 <- pml(tree = tree, data = align21, model = "WAG")
anc21 <- anc_pml(fit21)
```

STEP 11: Visual Comparison: Standard vs Gap-Aware Reconstructions

```
par(mar = c(4, 10, 4, 2))
image(as.phyDat(anc),scheme = "Clustal")
image(as.phyDat(anc)[, 180:220], scheme = "Clustal")
image(as.phyDat(anc21)[, 180:220], scheme = "Clustal")
plotAnc(anc21, 201)
plotAnc(anc21, 202)
```

STEP 12: Sequence logo plots

```
plotSeqLogo(anc21, node = "Node1", 1, 25, scheme = "Clustal")
plotSeqLogo(anc21, node = "Node2", 1, 25, scheme = "Clustal")
```

STEP 13: Key Residue Tracing

```
which(as.character(align21)=="C", arr.ind = TRUE)
plotAnc(anc21, 97, scheme = "Clustal")
plotAnc(anc21, 212, scheme = "Clustal")
plotAnc(anc21, 194, scheme = "Clustal")
```