**Alphabet**

Represents the motifs (states) in the substitution model. Relates alphabet motifs to ambiguity codes, and performs translation for different genetic codes.

**AlignAnalysis**

For reading and manipulating sequence alignments.

**LikelihoodFunction**

Performs the likelihood calculation either by itself, or by calling `calculatelikelihood`, a C++ module. Can also simulate an alignment, and estimate posterior probabilities of ancestral motifs.

**LikelihoodFunction**

**ParameterController**

Defines the parameterisation of the statistical model, sets parameter starting values and bounds for optimisation. Specifies the mapping of parameters in the optimisation vector to the likelihood calculation.

**Optimisers**

Bound-constrained numerical optimiser. Takes a vector of parameter values, their bounds, and a controller object for optimisation.

**Parallel**

 Defines a parallelisation stack with virtual processors. Communicates among processors using PyPar, a Python MPI interface.

**SubstitutionModel**

Services for defining and implementing Markov models of substitution. Both the preparation of the instantaneous rate matrix and the matrix exponential calculations is performed by a C / C++ module.

**Tree**

For reading and manipulating phylogenetic trees.

**AlignAnalysis**

For reading and manipulating sequence alignments.

**ParametricBootstrap**

Used to assess parameter confidence intervals or likelihood-ratio probabilities.