**What is JPred**

JPred is an online server that primarily predicts protein secondary structure of query sequences, alongside solvent accessibility, and coiled-coil regions

A user submits a single amino acid sequence or an alignment of multiple sequences, also called a multiple sequence alignment

The server then returns a 3-state secondary structure prediction at every residue position along with a confidence score

The confidence score indicates the reliability for each residue and is as important as the prediction

There are many applications of JPred, including predicting structure *ab initio*, act as a constraint in fold-recognition algorithms, and in the design of site directed mutants that do not destabilize folding

**How does JPred4 work**

The actual prediction itself relies on Jnet, a collection of independent artificial neural networks that returns structure prediction given sequence.

However, the main bulk of JPred’s computation time lies in generating the input for Jnet

In JPred4, the two inputs for Jnet are a PSI-BLAST position-based scoring matrix (or PSSM) and a hidden markov model profile

*Step 1 – multiple sequence alignment*

This step only applies if a user submits a single sequence

PSI-BLAST searches for related sequences in UniRef90. This step is done iteratively with increasingly stringent criteria.

The set of related sequences are then aligned against the query sequence to generate a multiple sequence alignment

*Step 2 – Generation of profiles*

Using the multiple sequence alignment, an HMM profile is generated from the HMMer program, and a PSSM profile is generated from PSI-BLAST

Together, the two profiles are fed as input into Jnet

*Step 3 – Prediction using Jnet*

Jnet v2.3.1 consists of two independent neural networks, one for the HMM profile and one for the PSSM profile

These neural networks convert a profile into a structure prediction for the query sequence

As structure predictions can sometimes be non-biological, an additional pair of structure-to-structure neural networks come after the sequence-structure network

The mathematical average of the two predictions is then calculated followed by a confidence score

**What data is Jnet trained and tested on**

Jnet v2.3.1 was trained on 1348 sequences via 7-fold cross validation and then tested on a 149 blind-test sequences

These sequences were obtained from the structural classification of proteins *extended*, or SCOPe

SCOPe is a hierarchical database of protein domains, where a domain is defined as a globular protein unit that has been experimentally determined to fold and exist on its own

To avoid redundancy, 1 representative sequence is selected from each SCOPe superfamily, which comprises of proteins with evolutionary conserved structure and function

For the ‘ground truth’ secondary structure, DSSP information is used

DSSP is a database of ‘ground truth’ secondary structures extracted from crystal structures on PDB.

DSSP database contains 8 states, which is reduced to 3 states to train Jnet

**Selection of training and testing data**

The selection started from 1987 representative sequences from each domain superfamily in SCOPe

The following filtering steps were then applied to refine the data set:

Removing sequences whose corresponding structures had low resolution

Filtering by length. Sequences need to be long enough to fold as domains, but too long and PSIBLAST computation will take too much time

Domains made up of more than one polypeptide chain but are removed. There is no reason given but I think it is because multi-chain domains are made of *super*-secondary structures

Sequences were checked for pairwise redundancy using the AMPS algorithm with 100 randomizations. No such sequences removed

110 sequences which had missing secondary structure assignments in the DSSP database were excluded.

**A glance at training and blind-test sets**

**Aims of project**

Rewrite Jnet in python to keep up with the times. Do so without modifications to its training method, and observe its effects on prediction accuracy

Retrain Jnet with modern machine learning techniques with the aim of improving prediction accuracy

Enlarge the size of the dataset with new sequences, or diversify the input sequence by training the network on human population genomic data

**Things to beware of**

Redundancy in training sets. It can lead to artificially high accuracies if sequences in the test set show similarity to one another