Milestone 4

Building the pipeline

- Main goal: Build optimal pipeline/meta predictor
- Additionally: Other predictions can be used to improve/overwrite your own predictions
- >some_protein TMP 9 TMP/nTMP RI
  ++++++++---------++++++++++ TMR/nTMR
  789998766789876678999999 RI TMR
  iiiiiiiiiiooooiiiiiiiiiiiiiiiooooiiioiioi In/Out
  111111111568998762144111 RI In/Out
  HHHHHHHHHHHHHHHHHHHH Helix/Loop
  899999998555546667999999 RI Helix/Loop

What do we need?

1. Building a sub-predictor
   - 1.1 Cross-validation predictions for every training data set (incl. Std Error/Std Dev)
   - 1.2 Performance evalutaion of your neighbor's method on independent data sets (incl. Std Error/Std Dev)

2. Building a meta-predictor
   - 2.1 Each group needs to provide prediction results (i.e. predicted classes & reliability indices) from their cross-validation
   - 2.2 Each group also needs to provide predictions for all the other sets (i.e. for the training data of other groups, and all independent sets from 1.2).
     Hint: train a model on your complete data and make a prediction

What do we need? Cont'd

2. Building a meta-predictor

2.3 Put your predictions in a separate folder in:
/mnt/opt/…/predictors/YOUR_PROBLEM_NAME

2.4 Collect predictions of all groups into the aligned view (slide 2)

2.5 Find a good way to combine predictions such that:

- A) your personal predictor improves (profit from other groups): Q2, MCC
- B) the overall combined model becomes optimal: Qtop(Qok)

2.6 Make sure you only optimize on the training data!

3. Evaluating your meta-predictor

Report the performance of your optimized sub-model model and the overall predictor on your independent test set

How to evaluate

- For every problem you can calculate Q2 and MCC on the corresponding independent data
- For the entire model you calculate Q_{top} (Q_{ok})
- => optimize model on training data, then evaluate “your problem” on independent data. Did other group’s predictions improve your performance?
- Q_{top} is only defined on the segment level (= TM residues), do this on tmps_independent

Second independent testing

We will additionally evaluate your pipelines on a second test set. Instances of this set have no class labels => blind data for you!

Since at this point you already have all the predictions of the other groups, use your meta predictor to combine these into your final prediction. The final output file for this second independent set should look like this:

>some_protein TMP (or non-TMP)

HHHHHHHHHiiiiiiHHHHHHHHHHoooooLLLLooooHHHHHHiii

Data Sets Overview

- Training data (/mnt/opt/…/dataset/)
  - tm protein/non-tm protein:

    tmp_sol.arff, tmps.arff, sol.fasta, imp.fasta
  - tm residues/ non-tm residues; l/h:

    tmps.arff, imp.fasta, imp_struct.fasta
  - i/o:

    tmps_i_o.arff, tmps_i_o.fa

Data Sets Overview, Cont'd

- 1st Independent test data (/mnt/opt/…/independent/)
  - tm protein/non-tm protein:

    imp.bioinfo.independent.fasta, sol.bioinfo.independent.fasta,

    imp.bioinfo.independent.arff, sol.bioinfo.independent.arff
  - tm residues/ non-tm residues:

    imp.independent.fasta, imp.independent_struct.fasta and tmps_independent.arff
  - i/o, h/l:

    tmps_i_o_independent.fa, tmps_i_o_independent.arff

- 2nd Independent test data (/mnt/opt/…/independent/)

  tmp_ind_2.fasta, tmp_ind_2.arff