# *CLS Dundee Introduction to Jalview – October 2014*

## **Supplementary exercises and Advanced tutorial materials**

All files in this tutorial are included in the zipfile you downloaded from:

www.jalview.org/tutorial/training-materials/2014/Dundee/Oct

## Disorder prediction

1. Import the interleukin7.fa alignment to Jalview.
2. Apply the *Web Service -> Protein Disorder ->* .. *-> Disembl* protein disorder predictor
3. Use *Sequence ID popup -> Structure -> Discover PDB Ids* to retrieve all the PDB structures for the sequences
4. Open and align the structures
5. Compare the disorder predictions to the structure data
6. Apply the IUPred disorder prediction method
7. Use the ***Per sequence*** option in the *Colour -> colour by annotation* dialog to shade the sequences by the long and short disorder predictors
8. Do the two methods agree with the structure ?

## Calculating T-COFFEE Reliability scores

1. Load the tcoffee_scores.fasta alignment into Jalview
2. Calculate a T-COFFEE score file for the alignment using the T-COFFEE COREX score server at http://tcoffee.crg.cat/apps/tcoffee/do:core
3. Locate the ascii score file on the results page – it's called 'sequence alignment in ascii format' (url should end in result.score_ascii)
    a. Save this to your desktop
    b. drag the file onto the alignment in Jalview
4. Toggle the T-COFFEE score colouring
5. Experiment with the Colour By Annotation dialog and export Annotation open in the file menu to see how Jalview handles T-COFFEE scores
6. Try uploading another file to the score server to calculate scores for your own sequences

## Viewing RNA secondary structure

1. Import RF00162 from *Rfam (Full)*
2. Select 'Colour by RNA Helices'
3. Open VARNA with '*Structure->View Structure->RNA Secondary Structure*'
    a. Explore difference between trimmed and untrimmed views
4. Bonus points – add and link a **Jmol structure view**
    a. ***Bacillus_amyloliquef.9*** *corresponds to PDB 3NPB*

## *Groovy Scripting and Jalview*

1. Open the groovy console from the Desktop’s Tools menu.

Load and experiment with the following scripts using the example alignment:

2. Printing the alignment’s title
http://www.jalview.org/examples/groovy/printtitle.groovy

3. Generating CSV from annotation
http://www.jalview.org/examples/groovy/annotationsascsv.groovy

4. Parsing description strings as alignment annotation (using the ‘Extract scores’ function)
Example file:
http://www.compbio.dundee.ac.uk/user/ws-dev1/examples/scanps_out.blc
Script:
http://www.jalview.org/examples/groovy/parseproperties.groovy

6. Manipulating features programmatically
http://www.jalview.org/examples/groovy/removeFeaturesByGroup.groovy

7. Manipulating sequence IDs in alignment
Load some sequences from Uniprot using the sequence fetcher.
Try this script http://www.jalview.org/examples/groovy/stripUniprotPrefixes.groovy

## *Setting up the Jalview Yoxos Eclipse development environment – if you want to add to Jalview!*

1. Download and double click the jalviewDeveloper-www profile (on tutorial page)
2. wait a few minutes then click the 'I accept button' and OK
3. wait a few minutes more for the eclipse download to happen
4. If you want to do commits, then you’ll need to get a username and password over at issues.jalview.org
5. Press ok for default eclipse workspace paths.
6. use your the username and password to log in to Jalview's git repository at source.jalview.org if you have one, otherwise you can hit cancel when it prompts you for username/password (If you hit ok then you'll need to set master password details .. don't worry - these machines will be wiped at the end of the day).
7. Wait a bit more - Jalview's git repository will be cloned automatically
8. Open git repository browser and Switch branches to Release_2_8_Branch to get the latest patched release branch in your desired repository.
9. Choose Import ..-> From Git -> Import Existing project -> From Local Repository -> pick the public or personal repository

10. Fix paths (some of the below may not be necessary)
i. Open builders - there's one invalid builder that you'll need to get rid of
ii. Open build path - VARNAv3.9-dev.jar reports being not found:
 * first remove from project path.
 * then add it again by right clicking after browsing to the 'lib' directory
iii. Fix the plugin.jar user library
* select plugin.jar in the build path and hit 'Edit ...'
* select 'user libraries ...' button.
* create a new user library called plugin.jar that links to the jalview 'plugin.jar' classpath entry
* select 'Add external jar' and locate plugin.jar on your system "Program files (x86)/Java then search for plugin.jar
* hit ok then finish then ok to finish editing the classpath.

11. source should be ready to hack ! test you can launch Jalview by :
i. right click on project and pick Run As -> External Application
ii. once main types are found look for jalview.bin.Jalview
iii. let the program launch.
iv. to double check - make sure you can import the example alignment at http://www.jalview.org/examples/exampleFile_2_7.jar

12. Building the applet is tricker (need to use the makeapplet target to build the jalviewApplet.jar), but it should launch via the appletviewer using as 'launch java applet' runtime profile. Use parameters from one of the examples in the examples/applets.html page.