AAstretch Project

User Manual

Phylosophy of AAstretch

AAstretch is a collaborative project of the University of Florence, Italy and the CNRS, France, aimed at a systematic evaluation of amino acidic (AA) residue repeats in genomes across evolution. As a bioinformatic project, it arises from the necessity of expanding the classical concept of "poly-residue repeat", usually defined as a consecutive stretch of the same residue, to a more biologically meaningful definition taking into account relatively small insertions and percentage thresholds to define the beginning and the end of a stretch.

The project started with a precise biological aim in mind: to discover the features and deepen the knowledge of the poly-glutamine (Q) repeats typical of triplet expansion diseases such as Huntington's disease, spinocerebellar ataxia and many other. Aggregation of polyQ containing proteins, in facts, lead to the formation of fibrils that impair the cellular machinery and lead to organic or systemic dysfunctions.

To accomplish this task, we conceived a computer program, AAstretch, that it is able to scan a properly formatted set of protein sequences and their corresponding coding sequences and extract from them stretches of pure/impure poly-residue stretches and emit a tabular text output in which a set of features are reported for each stretch. Such features include their positioning on the sequence, their flanking regions, the annotations and GO terms of the containing protein and many other information information for a correct biological interpretation of the presence of that stretch in the sequence.

If coding sequences are available for the proteins, this kind of information can also be investigated using what we call the “synchronization output”, i.e. a cds version of the AAstretch output (see below). These rather large panel of data can be graphically displayed on AAexplore, a GUI-based tool specifically developed to get the best out of AAstretch results.

Even if, in principle, any sequence or sequence set can be scanned with AAstretch, the best can be obtained working on whole genomic sets. We therefore developed an automatic builder (AAprepare) that, linked to EnsEMBL genomic database and taking advantage of the BioMart services, prepares organism specific annotated gene sets for the analysis with AAstretch. You can find that ready-to-use files for a number of organism spanning the different life kingdoms in the Organisms section of the AAstretch website. Since this section is intended to be yearly updated (following EMBL genome releases, possibly) all you need to do is download AAstretch, download your preferred organism and start analyzing. A full genome scanning takes one minute or two on modern computers. We do not have enough fundings to maintain a web-based engine, so we kept AAstretch very simple to run: edit a configuration file to change rather intuitive parameters, then run the analysis trough the interactive menu.

Installation

To install the programs, simply extract the files in the downloaded zip file into a desired working folder. This will leave you with three files: AAstretch.pl (the main scanner), AAsync.pl (for codon syncronization) and AAstretch.conf (the configuration file). All the .pl scripts must be placed in the working directory along with the genomic files (see below). AAexplore.pl should be placed in the AAstretch working directory, but any other location is acceptable. AAprepare.pl can be placed everywhere and will put prepared genomic files into its current directory.

Dependencies and modules

AAstrech programs are entirely written in perl, so perl needs to be installed into the box. On Linux and Mac OS X, perl usually comes preinstalled on the system. On Windows, ActiveState holds a well-curated perl distribution (others are available, but this one is suggested). Only base modules are used in AAstretch, so a minimal perl installation is sufficient for most purposes. AAprepare additionally requires Net::FTP, Archive::Extract and IO::Compress::Gzip modules. AAexplore additionally requires Tk, GD and GD::Graph modules.

Obtaining genomic files

Genomic files are Ensembl derived fasta files containing protein sequence and the corresponding coding sequences, provided with transcript, protein and gene codes, descriptions, gene ontology annotations and omim annotations (in human only). They can be downloaded in zip format from the AAstretch Project page. All the

genomes available in Ensembl have already been processed and prepared, so they are ready-to-use by AAstretch. They only need to be extracted into the working folder and AAstretch has to be configured to use to use them (in the .conf file). In cases of genomes absent from the organism list in the website (e.g. due to delays in update after a new EnsEMBL release) AAprepare.pl can be run to build up an brand new genomic file. Its use is straightforward: launch the program, select the database by typing the appropriate number, then select from the list the organism name (again, by typing the correct number) and wait till the protein and the cds files are created.

Basic usage

Once both the scripts and the organism files are in the same folder, open the terminal/prompt and cd into that directory. Launch AAstretch with "perl AAstretch.pl" and an interactive menu appears listing the various possibilities:

1. Run: launches the scanner according to parameters specified in the Aastretch.conf file
2. Restart: used to delete previous analyses and recreate the starting environment
3. Clean: used to delete previous analyses, including the organism files
4. Isolate: moves all the files of the current analysis results into a new timestamped folder
5. Help: offers a brief explanation of these points
6. Quit: shut off the program (same as ctrl-c)

At the first run the program spend time in loading the proteome into memory and, if requested, to create isoforms details. Then the analysis starts and, according to the configuration, in the end emits several files as output. They can be investigated in several ways.

Once a run has finished, the proigrams

The AAstretch.conf file

This is the very heart of the AAstretch procedure: this simple text file configures AAstretch.pl to process the genomic files in the desired way. It is ideally divided into four main sections, that will be described here:

1. General parameters

residue	one letter code, the residue whose stretches must be searched.
flank_start	the position before (N-term) or after (C-term) the stretch at which the flanking regions begins
flank_length	the length of the regions on left and right side of a stretch
scanmode	Can be set to rich, seed or patt (this decides the core engine to be used, see below for details).
isoform_check	On or off, this removes duplication in stretches due to protein isoforms (see below for a working definition of isoform)

2. Parameters for the specific engines

AAstretch contains three search engine (seed, patt and rich) with different performances and scopes. The engine parameters are easily identifiable thanks to the prepended flag (seed_, rich_, patt_). The engine is selected above so there is no need to put comment marks in this sections to hide the configuration of the engines other that that in use.

Seed engine	
seed_seed_min_size	A seed is an homopolymer of a residue inside a protein sequence. This impose the minimal length of the seed

seed_seed_max_size	This impose the maximal length of the seed
seed_stretch_min_aa_perc	The minimal % of the selected residue in the whole stretch
seed_gap_max_size	The maximal length of "gaps" into homopolymers of the desired residue

Rich engine	
rich_win_length	The length of the sliding window over which to calculate the % of the desired residue
rich_gap_tolerance	The length of a gap that can be bypassed even if the % threshold fall below that imposed as lower limit.
rich_stretch_min_size	The minimal length of the whole stretch
rich_stretch_min_aa_perc	The minimal % of the selected residue in the whole stretch

Patt engine	
patt_stretch_min_size	The minimal size of a stretch
patt_stretch_max_size	The maximal size of a stretch
patt_gap_min_size	The minimal size of the gap (used to exclude homopolymers from the analysis)
patt_extrem_len	This controls the length of homopolymers at the C- and N- termini of the stretch
patt_gap_max_number	This controls how many gaps are tolerated

4. Parameters for the workflow control

The workflow decides which programs to run given an instance of the AAstretch launched. The three programs involved are AAstretch, AAsync and AAexplore. A full analysis is performed when all options are set to 1, but in some cases it is useful to exclude some parts (e.g. in explorative sessions, sychronization is not appropriate), that is done giving setting the desired option to zero.

scan	If set to 1, instruct AAstretch to perform the scan according to parameters specified above
sync	If set to 1, after the proteome-based search for stratches, it loads the coding sequences and create an alternative version of the AAstretch output containing the coding sequences of the corresponding stretches/flanks.
explore	After the analysis is completes, automatically launches AAexplore for the investigation of the results obtained from the scan.

4. Parameters for the additional options

Additional options are so-called because they do not affect the operative procedures of the programs, but are far to be less important. In fact from here one can set input files and configure protein filters. Filters allows e.g. restrict the analysis on a smaller portion of the genome given some previous results.

ignore	This is used to filter out entries whose description matches the words included here. Some basic knowledge of pattern matching is needed to take full advantage of this option (e.g. "ignore=hypothetical" will trash all proteins

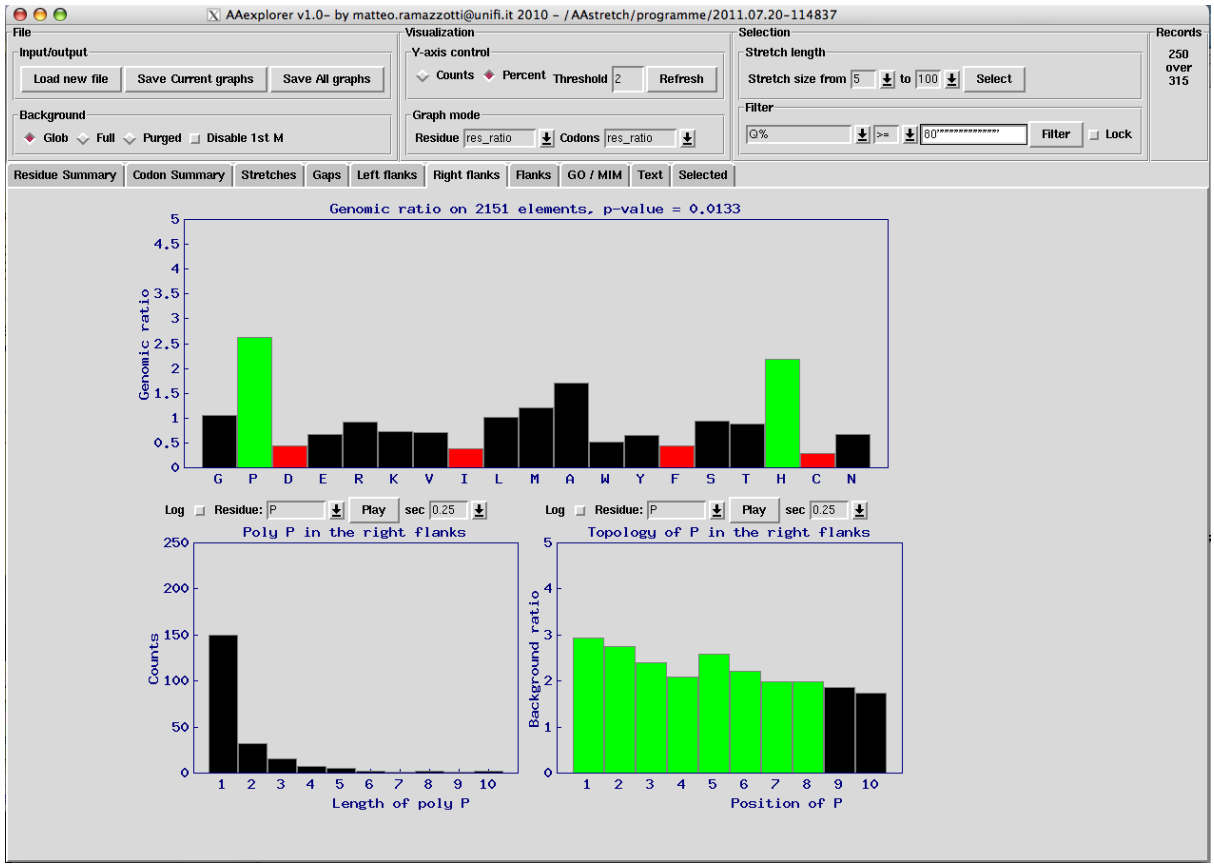
	annotated as hypothetical, while ignore=”hypothetical|predicted” will also additionally exclude predicted proteins, being the “|” sign the boolean OR.
only	Somewhat the contrary of the option above, since only proteins matching the keywords will be provcessed by AAstretch
data_source	This points to the genomic file that AAstretch will used as an input.

Syncronization with codons

To syncronize in AAstretch means to localize which part of a coding sequence codes for a given trait of a polipeptidic sequence. Since the main AAstretch program extracts polyAA repeats and their flanking sequences, and since AAprepare generates, for each gene transcript in an organism, both the coding sequence and the corresponding protein sequence, the AAsync program takes the output of the AAstretch program and convert polypeptidic sequences into coding sequences. This step is fundamental to investigate the role of codons in amino acid repeats. The AAexplore program (see below) can in fact analyze the frequencies of both residues and codons, provided that the two input files are synchronized.

AAexplore

this program has been written for directly investigating the results of AAstretch, that basically emits a tab separated text file with stretch specific results, one per line. A screenshot of AAexplore is shown here below:



The window is organized in two blocks, the control block on the top and the graph block on the bottom. The control block is divided into 4 sub-blocks:

File, with 2 regions:

Input/Output: from there one can load input files and save graphs. Note that saving graphs is also possible by shift-clicking in the graphs themselves.

Background: from there one can change the default background model used when frequencies are calculated (see the statistics section below).
Visualization, with 2 regions:
Y-axis control: from there one can change the Y axis of the main graph in the graph block and the threshold for evidencing and coloring the bars in the graphs.
Graph mode: the two mode of visualization (residue based or codon based) can be switched according to the investigation of interest.
Selection, with 2 regions
Stretch length: from here one can limit the size of the stretches to be investigated
Filter: from here one have access to the different elements of the AAstretch output, including annotations and detailed descriptions of each stretch, in order to selectively exclude form the analysis those stratches with unwanted properties (e.g. the AA%, the position on the protein, or the fact that it is contained in the nucleous etc.)
Records: this is a simple remainder that counts the total number of stretches available and those remained after filtering.

The graphic block instead present several tabs, each containing different output modes and portions of the stretch as well.
The Residue Summary tab describes with three graphs some general feature of the whole output e.g. there is
A bar plot with a dedicated binner for investigating the distribution of the stretch lengths
A scatter plot for evaluating if there is a dependence of the location of the stretch with the stretch length
a scatter plot for evaluating if there is a relationship between stretch length and the % of the residue of interest in the stretch.

The Codon Summary tab is basically the same as above, except that all considerations are based on codons instead of AA residues (to view this section, a codon analysis must have been performed with the program AAsync).

The “Stretches” tab reports a main upper bar plot with genomic ratios of all residues against the background (that can be adjusted) (or simple counts, see the description of the upper block) and a lower bar plot describing, for each residue, the length of the pure stretches in the main stretch. This is useful for investigating if the main stretch is interrupted by homopolymers.

The “Gaps” tab report something that is similar to the stretch, except for the fact that the residue specified into AAstretch is excluded form the counts (both in the background and in the stretched) in order to have and unbiased ratio and fully appreciate the over/under representations of the various residues.

The ”Left Flanks” tab report the same graphs seen for the stretches but in this case the counts are based on the region that the user selected as flanking the stretch at the N-terminal. The length of this region can be changed in AAstretch, not in AAexplore. An additional graph is present reporting the bias (genomic ratio) of selectable residues at the different position of the flanks (note that thay all share the same length), important for evaluating the “topology” of residues in the flank.

The “Right Flanks” is identical to the left one, except for the fact that in this case the C-treminal is taken into account. Please not that the numeration indicate the positionining form where the stretch left thesequence, so a distance of 2 means that is 2 residues far from where the stretch ended (right flank) or started (left flank).

The “Flanks” tab is a simple combinations of the left and the right flank: counts are summed up and expressed as a single entity called flanks.

The “GO/MIM” tab show a bar plot of the distributions of the different GO terms or MIM description in the stretches. It is basically useful to evaluate if there is some kind functional or topological enrichment in the results.

The “Selected” tab lists all the entries form the main AAstretch output that are currently under investigation, since some or many stretches may have been removed form the analysis due to filtering or uninteresting lengths. In the right part of this input there is a window that fills and refreshes automatically when one double-click on a graph, showing the data used to produce that graph. Basically this is useful for reproducing the AAexplore images in external softwares, since, as the name say, AAexplore is not intended to produce high quality images but rather to rapidly explore the output of the AAstretch program.