AVE Allelic Variation Explorer

installation

Below you can find installation instructions with all necessary libraries.

ubuntu server 12.04 LTS

1. install few first prerequisites

   ```
   sudo aptitude install build-essential python-dev curl unzip
   ```

2. install BEDTools

   run following commands in shell

   ```
   curl -O http://bedtools.googlecode.com/files/BEDTools.v2.17.0.tar.gz
   tar xvzf BEDTools.v2.17.0.tar.gz
   cd bedtools-2.17.0
   make
   cp bin/* /usr/local/bin/
   ```

3. install MongoDB

   follow instructions for ubuntu at mongodb website

   after installation mongod process should be running and database should be located at /var/lib/mongodb

4. install virtualenv

   create directory for virtualenvs

   ```
   mkdir ~/venvs
   ```

   download and unpack python-virtualenv

   ```
   wget https://pypi.python.org/packages/source/v/virtualenv/virtualenv-1.8.4.tar.gz
   tar xvzf virtualenv-1.8.4.tar.gz
   cd virtualenv-1.8.4
   ```

   create virtual environment for ave and activate it

   ```
   python virtualenv.py --no-site-packages ~/venvs/ave_env
   source ~/venvs/ave_env/bin/activate
   ```

5. install node.js

   follow instructions at node.js website

setting up AVE

These instructions are independent of the operating system. It is important to work in virtualenv ('source ~/venvs/ave_env/bin/activate', as explained above).

1. Download the application.
2. Unpack the code and checkout the latest version branch

   ```
   tar xvzf ave.tar.gz
   cd ave
   ```

3. install node packages

   ```
   npm install
   ```

4. install python libraries

   from within ave directory run (make sure that ave virtualenv is activated):

   ```
   pip install -r requirements.txt
   ```

5. Setup the db

   To setup the db with your own data, all Arabidopsis example data you can use provided script. You will need:

   - reference sequence in fasta format

     make sure that name of the chromosome (or some other meaningful identifier) is provided as fasta identifier (the string just after ">"). Like in the example for Chromosome 1 sequence:

     ```
     >Chr1 CHROMOSOME dumped from ADB: Jun/20/09 14:53
     CCCTAAACCCTAAACCCTAAACCCTAAACCTCTGAATCCTTAATCCCTA
     ```

   - gene annotations in gff3 format
   - SNP annotations in gff3 format
   - chromInfo.txt file containing information about chromosome names and sizes, for example for Arabidopis:

     ```
     Chr1 30427671
     Chr2 19698289
     Chr3 23459830
     Chr4 18585056
     Chr5 26975502
     ChrC 154478
     ChrM 366924
     ```

identifiers in first column must match identifiers in fasta and gff files

- to simplify, configuration json file can be used, it should be valid json file (json validator), it should look like following:

```
 {
"genome": "TAIR10",
"ref": [
"/path/to/data/annots/TAIR10_chr1.fas",
"/path/to/data/annots/TAIR10_chr2.fas",
"/path/to/data/annots/TAIR10_chr3.fas",
"/path/to/data/annots/TAIR10_chr4.fas",
"/path/to/data/annots/TAIR10_chr5.fas",
"/path/to/data/annots/TAIR10_chrC.fas",
"/path/to/data/annots/TAIR10_chrM.fas"
],
"annot": [
"/path/to/data/annots/TAIR10_GFF3_genes.gff",
"/path/to/data/annots/snps/CDS_snps.gff",
"/path/to/data/annots/snps/three_prime_UTR_snps.gff",
"/path/to/data/annots/snps/five_prime_UTR_snps.gff"
],
"chromInfo": "/path/to/data/annots/chromInfo.txt"
}
```

Please validate gff files before importing them. This can be done at genome tools webiste

```
SNPs should be annotated like in this example
columns 1-7:

    Chr1 1001Genomes SNP_adal_3 138 138 3 . .

column 8 (key value pairs):

    Change=T:C;Strain=adal_3;Project=GMINordborg2010;ID=9323.138


First column should correspond to seq id from fasta file provided as reference.

In last column:

‘Change‘ follows ‘reference:variant‘ order

‘Strain‘ is the name of the strain/accession/ecotype in which this SNP have been called
```

‘Project‘ is the sequencing project

‘ID‘ is any unique identifier for this SNP

You can annotate the SNPs in gff file with SNPs location. To do it run

```
python ./ave_tools.py group_snps_by_loc --annot gene_annotation.gff \
--snps snp_file1.gff --snp_file2.gff
```

or

```
python ./ave_tools.py group_snps_by_loc --annot gene_annotation.gff \
--snps *.gff
```

The script generates new gff files, one for each snp location, with annotated location in last column:

```
Project=GMINordborg2010;Strain=ale_stenar_44_4;variant_location=CDS;
ID=992.6992;Change=T:C
```

To import data into the database run:

```
python ./ave_tools.py import --genome TAIR10 --ref \
reference.fas --annot gene_annotations.gff snps_annotations.gff
```

after `--genome` provide a name of the genome which was used to map the reads and call variants against

after `--ref` provide a list of fasta files with reference sequence

after `--annot` provide a list of files with gene/trait/snp annotations

or use confgiuration file:

```
python ./ave_tools.py import --config conf.json
```

starting up AVE

run:

```
node app.js
```

Access app from within web browser (preferably latest chrome). Ip address and port is provided in app.js output.

important ifo

Example SNP annotations have been obtained from 1001 Genomes Project. Please read the Data Usage Policy at the project website.