`formatPro`, v. 0.5: Format Profiles for `mlRho`

Bernhard Haubold

Max-Planck-Institute for Evolutionary Biology, Plön, Germany

October 29, 2012

1 Introduction

`formatPro` formats Profiles for analysis by `mlRho`. Take a look at the documentation for `mlRho` to see how `formatPro` is used.

2 Getting Started

`formatPro` was written in C on a computer running Linux and should work on any standard UNIX system. However, please contact me at `haubold@evolbio.mpg.de` if you have any problems with the program.

- Unpack the program

  `tar -xvzf formatPro_XXX.tgz`

  where `XXX` indicates the version.

- Change into the newly created directory

  `cd FormatPro_XXX`

  and list its contents

  `ls`

- Generate `formatPro`

  `make`

- List its options

  `./formatPro -h`

3 Listing

The following listing documents the driver program for `formatPro`.

```
1  /***** formatPro.c ******************************
    * Description:
    * Author: Bernhard Haubold, haubold@evolbio.mpg.de
    * Date: Mon Oct 15 20:36:50 2012
    **************************************************/
6  #include <stdio.h>
```

```c
    #include <stdlib.h>
    #include <string.h>
    #include <fcntl.h>
    #include <unistd.h>
11  #include "interface.h"
    #include "eprintf.h"
    #include "profileTree.h"

    void scanFile(FILE *fp, Args *args);

16
    int main(int argc, char *argv[]){
      int i, fd;
      char *version, *fileName;
      Args *args;
21    FILE *fp;
      Profile *profiles;
      Node *root;
      ContigDescr *cd;

26    version = "0.5";
      setprogname2("formatPro");
      args = getArgs(argc, argv);
      fileName = (char *)emalloc(256*sizeof(char));
      if(args->v)
31      printSplash(version);
      if(args->h || args->e)
        printUsage(version);
      fileName = strcpy(fileName,args->n);
      fileName = strcat(fileName,".pos");
36    fp = fopen(fileName,"wb");
      fwrite("pos",sizeof(char),3,fp);
      root = NULL;
      profiles = NULL;
      cd = (ContigDescr *)emalloc(sizeof(ContigDescr));
41    cd->n = 0;
      cd->len = NULL;
      if(args->numInputFiles == 0){
        fd = 0;
        root = writePositions(fd, fp, root, cd, args);
46    }else{
        for(i=0;i<args->numInputFiles;i++){
          fd = open(args->inputFiles[i], O_RDONLY, 0);
          root = writePositions(fd, fp, root, cd, args);
          close(fd);
51      }
      }
      fclose(fp);
      profiles = getProfileArray();
      printf("#Positions_written_to_%s\n", fileName);
56    /* write profiles to file */
      writeProfiles(profiles, args);
      /* write contig lengths to file */
      cd = getContigDescr();
      writeContigs(cd, args);
```

2

```
61     freeProfileArray(profiles);
       freeProfileTree(root);
       free(args);
       free(progname());
       free(fileName);
66     return 0;
    }
```

4   Change Log

- Version 0.1 (October 17, 2012)

    - First running version.

- Version 0.2 (October 18, 2012)

    - Switched from binary format to text format.

- Version 0.3 (October 18, 2012)

    - Mixture between text (profiles and contigs) and binary (positions).
    - Fixed bug in treatment of contig lengths from multiple files.

- Version 0.4 (October 20, 2012)

    - All files binary.

- Version 0.5 (October 29, 2012)

    - Included option for setting the minimum coverage (-c).