inspectPro, v. 0.2: DESCRIPTION

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

November 14, 2012

1 Introduction

2 Getting Started

inspectPro 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 inspectPro_XXX.tgz
  ```

  where XXX indicates the version.

- Change into the newly created directory

  ```
  cd InspectPro_XXX
  ```

  and list its contents

  ```
  ls
  ```

- Generate inspectPro

  ```
  make
  ```

- List its options

  ```
  ./inspectPro -h
  ```

3 Listing

The following listing documents the driver program for inspectPro.

```
/***** inspectPro.c ******************************
 * Description:
 * Author: Bernhard Haubold, haubold@evolbio.mpg.de
 * Date: Fri Oct 19 09:41:32 2012
 *************************************************/
#include <stdio.h>
#include <stdlib.h>
#include <assert.h>
#include <string.h>
#include "interface.h"
```

```
#include "profileTree.h"
#include "eprintf.h"

void writeProfiles(Profile *pro, int n){
  int i;

  printf("#ID\tCount\tA\tC\tG\tT\n");
  for(i=0;i<n;i++)
    printf("%d\t%d\t%d\t%d\t%d\t%d\n",i,pro[i].n,
           pro[i].profile[0],
           pro[i].profile[1],
           pro[i].profile[2],
           pro[i].profile[3]);
}

void writeLengths(int *len, int n){
  int i;

  printf("#ID\tLength\n");
  for(i=0;i<n;i++)
    printf("%d\t%d\n",i,len[i]);
}

void writePositions(FILE *fp){
  int numRead;
  Position *pos;

  printf("#Pos\tPro\n");
  pos = (Position *)emalloc(sizeof(Position));
  numRead = fread(pos,sizeof(Position),1,fp);
  assert(numRead == 1);
  while(!feof(fp)){
    printf("%d\t%d\n",pos->pos,pos->pro);
    numRead = fread(pos,sizeof(Position),1,fp);
  }
  free(pos);
}

void scanFile(FILE *fp, Args *args){
  char *tag;
  int numRead, *lengths;
  Profile *profiles;
  int n;

  tag = (char *)emalloc(3*sizeof(char));
  numRead = fread(tag,sizeof(char),3,fp);
  assert(numRead == 3);
  while(!feof(fp)){
    if(strcmp(tag,"sum")==0){
      numRead = fread(&n,sizeof(int),1,fp);
      assert(numRead == 1);
      profiles = (Profile *)emalloc(n*sizeof(Profile));
      numRead = fread(profiles,sizeof(Profile),n,fp);
      assert(numRead == n);
```

```
      writeProfiles(profiles, n);
      free(profiles);
    }else if(strcmp(tag,"con")==0){
      numRead = fread(&n,sizeof(int),1,fp);
      assert(numRead == 1);
      lengths = (int *)emalloc(n*sizeof(int));
      numRead = fread(lengths,sizeof(int),n,fp);
      assert(numRead == n);
      writeLengths(lengths, n);
      free(lengths);
    }else if(strcmp(tag,"pos")==0){
      writePositions(fp);
    }else
      assert(0);
    numRead = fread(tag,sizeof(char),3,fp);
  }
  free(tag);
}

int main(int argc, char *argv[]){
  int i;
  char *version;
  Args *args;
  FILE *fp;

  version = "0.2";
  setprogname2("inspectPro");
  args = getArgs(argc, argv);
  if(args->v)
    printSplash(version);
  if(args->h || args->e)
    printUsage(version);
  if(args->numInputFiles == 0){
    fp = stdin;
    scanFile(fp, args);
  }else{
    for(i=0;i<args->numInputFiles;i++){
      fp = efopen(args->inputFiles[i],"rb");
      scanFile(fp, args);
      fclose(fp);
    }
  }
  free(args);
  free(progname());
  return 0;
}
```

4 Change Log

- Version 0.1 (October 22, 2012)
  - First version that runs.
- Version 0.2 (October 24, 2012)

  - When printing positions the last position was printed twice; fixed.
  - Filled in interface.

- Version 0.3 (November 14, 2012)
  - Implement inspection of `*.lik` file; works only with `mlRho` v. $\geq$ 1.25.