

Dec 27, 05 22:45

GP3xCLI.awk

Page 1/5

```
#####
#
#          CCCCCC  SSSSSS  IIIIII  RRRRRR  OOOO          L          IIIII  #
#          C          S          I          R  R  O  O          L          I          #
#          C          SSSSSS  I          RRRRRR  O  O  =====  L          I          #
#          C          S          I          R  RR  O  O          L          I          #
#          CCCCCC  SSSSSS  IIIIII  R  RR  OOOO          LLLLLL  IIIII  #
#
#####
echo " #####"
echo " # # "
echo " # GP3xCLI # "
echo " # GenePix Processing Program by CSIRO Livestock Industries # "
echo " # Bioinformatics Group # "
echo " # # "
echo " # Enquiries: Tony.Reverter-Gomez@csiro.au # "
echo " # Copyright (c) 2003-2005 CSIRO Livestock Industries # "
echo " # # "
echo " #####"

echo
filename=`ls -l $1 | awk '{print $NF}'`
echo "GPR Input:" $filename | \
    awk '{printf "%3s%7s%5s%-30s\n", $1, $2, " ", $3}'

date | awk '{printf "%9s%4s%5s%4s%3s%9s%4s%5s\n", \
    "Processed", "on:", $1, $2, $3, $4, $5, $6}'
echo

#####
#
# Column Location for bits of interest (Starts at row 31)
#
# 1. Block      2. Row      3. Colum      5. Gene ID
# 9. R fg Med  10. R fg Mean  12. R bg Med  14. R bg SD
# 18. G fg Med  19. G fg Mean  21. G bg Med  23. G bg SD
# 43. Q Flag
#
# NB: Some Gene ID contain a "." to accommodate GeneSpring limitations
# Needs to build "tempo0" file once and everything resolves from it
#
#####

awk 'NR>30 {print $5}' $1 | sed "s/./g" | awk '{print $1}' > id.nodot
awk 'NR>30 {print $1, $2, $3, \
    $9, $10, $12, $14, \
    $18, $19, $21, $23, \
    $43}' $1 | \
paste - id.nodot | \
awk '{print $1, $2, $3, $13, \
    $4, $5, $6, $7, $8, $9, $10, $11, $12}' > tempo0
rm id.nodot

echo
echo " ===== IMAGE QUALITY ====="
echo
T=`awk '{print $0}' tempo0 | wc | awk '{print $1}'`
echo "Total No. of Spots ----->" $T

echo
echo " QUALITY FLAG  SPOTS"
echo " -----"
awk '{print $13}' tempo0 | sort | \
    awk 'BEGIN{pcg="";n=1};
    {if($1==pcg) n++;
    else{ if(pcg!="")printf"%14s%9d\n", pcg, n;
    n=1;
    pcg=$1};
    }END{ printf"%14s%9d\n", pcg, n}'
```

Dec 27, 05 22:45

GP3xCLI.awk

Page 2/5

```

echo
N=`awk '$7>$5 {print $0}' tempo0 | wc | awk '{print $1}'`
echo "Red dye with Background >= Foreground ---->" $N | \
awk '{printf "%3s%6s%5s%11s%3s%11s%5s%6d\n", \
    $1, $2, $3, $4, $5, $6, $7, $8}'

N=`awk '$11>$9 {print $0}' tempo0 | wc | awk '{print $1}'`
echo "Green dye with Background >= Foreground ---->" $N | \
awk '{printf "%5s%4s%5s%11s%3s%11s%5s%6d\n", \
    $1, $2, $3, $4, $5, $6, $7, $8}'

echo
echo " Median to Mean Correlation Analysis:"
echo

# NB: Set zeroes to twos so their log exist

awk '$5=="0" {$5="2"}; $6=="0" {$6="2"}; \
    $9=="0" {$9="2"}; $10=="0" {$10="2"}; \
    {print $5, $6, $9, $10, \
        log($5)/log(2), log($6)/log(2), \
        log($9)/log(2), log($10)/log(2)}' tempo0 > rg

awk '{print ($1>$2?$2/$1:$1/$2)}' rg > rr
awk '{print ($3>$4?$4/$3:$3/$4)}' rg > gr
awk '{print ($5>$6?$6/$5:$5/$6)}' rg > rl
awk '{print ($7>$8?$8/$7:$7/$8)}' rg > gl

echo "          DATA LEFT"
echo "          RED      GREEN"
echo " Corr    Raw Log2  Raw Log2"
echo " _____"

for minr in 0 0.2 0.4 0.6 0.8 0.85 0.9
do
    T1=`awk -v corr=$minr '$1>corr {print $0}' rr | wc | awk '{print $1}'`
    T2=`awk -v corr=$minr '$1>corr {print $0}' rl | wc | awk '{print $1}'`
    T3=`awk -v corr=$minr '$1>corr {print $0}' gr | wc | awk '{print $1}'`
    T4=`awk -v corr=$minr '$1>corr {print $0}' gl | wc | awk '{print $1}'`
    echo ">" $minr $T1 $T2 $T3 $T4 | \
    awk '{printf "%2s%5.2f%9d%7d%9d%7d\n", $1, $2, $3, $4, $5, $6}'
done

rm rg rr rl gr gl

echo
echo " ===== VALID SPOTS* ====="
echo
V=`awk '$13==0 && $5>$7 && $9>$11 {print $0}' tempo0 | wc | \
    awk '{print $1}'`
echo "Total No. of Valid Spots ----->" $V | \
awk '{printf "%5s%4s%4s%6s%6s%19s%6d\n", \
    $1, $2, $3, $4, $5, $6, $7, $8}'

N=`echo $V $T | awk '{printf "%7.1f\n", $1/$2*100}'`
echo "Percentage of Valid Spots ----->" $N | \
awk '{printf "%10s%3s%6s%6s%19s%6.1f\n", \
    $1, $2, $3, $4, $5, $6, $7}'

echo

awk '$13==0 && $5>$7 && $9>$11 {print $4}' tempo0 | sort | \
    awk 'BEGIN{pcg="";n=1}; \
        {if($1==pcg) n++; \
        else{ if(pcg!="")print pcg, n; \
            n=1; \
            pcg=$1}; \
        }END{ print pcg, n}' > gcnt

```

Dec 27, 05 22:45

GP3xCLI.awk

Page 3/5

```

G=`wc gcnt | awk '{print $1}'`
echo "Total No. of Genes ----->" $G | \
awk '{printf "%5s%4s%3s%6s%26s%6d\n", $1, $2, $3, $4, $5, $6}'

R=`awk '{print $2}' gcnt | awk '{sum += $1}; END{print int(sum/NR+0.5)}'`
NG=`awk -v nr=$R '$2==nr {print $0}' gcnt | wc | awk '{print $1}'`
echo "Mean No. Repetitions ----->" $R " for" $NG " Genes" | \
awk '{printf "%4s%5s%12s%7s%4d%5s%6d%7s\n", \
    $1, $2, $3, $4, $5, $6, $7, $8}'

MIN=`awk '{print $2}' gcnt | sort -n | head -1 | awk '{print $1}'`
NG=`awk -v min=$MIN '$2==min {print $0}' gcnt | wc | awk '{print $1}'`
echo "Min. No. Repetitions ----->" $MIN " for" $NG " Genes" | \
awk '{printf "%4s%5s%12s%7s%4d%5s%6d%7s\n", \
    $1, $2, $3, $4, $5, $6, $7, $8}'

MAX=`awk '{print $2}' gcnt | sort -n | tail -1 | awk '{print $1}'`
NG=`awk -v max=$MAX '$2==max {print $0}' gcnt | wc | awk '{print $1}'`
echo "Max. No. Repetitions ----->" $MAX " for" $NG " Genes" | \
awk '{printf "%4s%5s%12s%7s%4d%5s%6d%7s\n", \
    $1, $2, $3, $4, $5, $6, $7, $8}'

rm gcnt

#####
# Build and process 'rgma' #
#####

awk '$13==0 && $5>$7 && $9>$11 {print $4, $5-$7, $9-$11}' tempo0 | \
awk '{print $2, $3, log($2/$3)/log(2), 0.5*log($2*$3)/log(2)}' > rgma

echo
echo
echo "          Log(R/G) vs 0.5*Log(R*G)"
echo "          _____"

awk '{print $3, $4}' rgma | \
awk '{ v1[NR]=$1; v2[NR]=$2}; \
    END{ min1=min2=99999; max1=max2=-99999; \
        for(i=1;i<=NR;i++){ \
            if( v1[i] < min1 ) min1 = v1[i]; \
            if( v2[i] < min2 ) min2 = v2[i]; \
            if( v1[i] > max1 ) max1 = v1[i]; \
            if( v2[i] > max2 ) max2 = v2[i]; \
            s1 += v1[i]; ss1 += v1[i]*v1[i]; \
            s2 += v2[i]; ss2 += v2[i]*v2[i]; \
            ss12 += v1[i]*v2[i] }; \
        mean1 = s1/NR; \
        mean2 = s2/NR; \
        std1 = sqrt(( ss1 - (s1*s1)/NR ) / (NR-1)); \
        std2 = sqrt(( ss2 - (s2*s2)/NR ) / (NR-1)); \
        num = ( ss12 - (s1*s2)/NR ) / (NR-1); \
        den = std1 * std2; \
        corr = num / den; \
        printf "%10s%11d%17d\n", "N", NR, NR; \
        printf "%10s%11.3f%17.3f\n", "Mean", mean1, mean2; \
        printf "%10s%11.3f%17.3f\n", "Std", std1, std2; \
        printf "%10s%11.3f%17.3f\n", "Min", min1, min2; \
        printf "%10s%11.3f%17.3f\n", "Max", max1, max2; \
        printf "%18s%10.3f\n", "Correlation", corr; \
    }'

echo
echo "          Log(R/G) across Intensity Values "
echo "          Intensity Spots %<0 %>0"
echo "          _____"

```

Dec 27, 05 22:45

GP3xCLI.awk

Page 4/5

```

LOW=0
UPP=4
MAX=16
while [ $UPP -le $MAX ]
do
    T=`awk -v min=$LOW -v max=$UPP \
        '$4>=min && $4<max {print $4}' rgma | wc -l`
    P=`awk -v min=$LOW -v max=$UPP \
        '$4>=min && $4<max && $3>0 {print $4}' rgma | wc -l`
    N=`awk -v min=$LOW -v max=$UPP \
        '$4>=min && $4<max && $3<=0 {print $4}' rgma | wc -l`

    P=`echo $P $T | awk '$2>0 {$3=$1/$2*100}; $2==0 {$3=0}; {print $3}`
    N=`echo $N $T | awk '$2>0 {$3=$1/$2*100}; $2==0 {$3=0}; {print $3}`
    echo "(" $LOW "," $UPP ")" "$T $N $P | \
    awk '{printf "%7s%2d%2s%3d%-2s%9d%8.1f%7.1fn", \
        $1, $2, $3, $4, $5, $6, $7, $8}'

    LOW=`expr $LOW + 4`
    UPP=`expr $UPP + 4`
done
echo " _____"

echo
echo "*NB: Valid Spot defined as spots with Background < Foreground for"
echo " both Red and Green channels and with a Quality Flag of 0."

#####
# Compute densities of variables in rgma file
#####

awk '{print log($1)/log(2)}' rgma | sort -n | \
    awk '{ data[NR] = $1 };
        END { min = data[1]; max = data[NR]; range = max - min;
              n_int = ( int(NR/10) > 1000 ? 1000 : int(NR/10) )
              size = range / n_int;
              mn_int = min + size/2;
              for(i=1; i<=NR; i++){
                  aux = int((data[i] - min)/size) + 1;
                  q[aux]++;
              };
              for(i=1; i<=n_int; i++){
                  if( q[i] < 1 ) q[i] = 1;
                  print mn_int, q[i];
                  mn_int += size;
              };
        }' > logr.d

awk '{print log($2)/log(2)}' rgma | sort -n | \
    awk '{ data[NR] = $1 };
        END { min = data[1]; max = data[NR]; range = max - min;
              n_int = ( int(NR/10) > 1000 ? 1000 : int(NR/10) )
              size = range / n_int;
              mn_int = min + size/2;
              for(i=1; i<=NR; i++){
                  aux = int((data[i] - min)/size) + 1;
                  q[aux]++;
              };
              for(i=1; i<=n_int; i++){
                  if( q[i] < 1 ) q[i] = 1;
                  print mn_int, q[i];
                  mn_int += size;
              };
        }' > logg.d

awk '{printf "%10.5fn", $3}' rgma | sort -n | \
    awk '{ data[NR] = $1 };
        END { min = data[1]; max = data[NR]; range = max - min;

```

Dec 27, 05 22:45

GP3xCLI.awk

Page 5/5

```

        n_int = ( int(NR/10) > 1000 ? 1000 : int(NR/10) )
        size = range / n_int;
        mn_int = min + size/2;
        for(i=1; i<=NR; i++){
            aux = int((data[i] - min)/size) + 1;
            q[aux]++;
        };
        for(i=1; i<=n_int; i++){
            if( q[i] < 1 ) q[i] = 1;
            print mn_int, q[i];
            mn_int += size;
        };
    }' > m.d

awk '{print $4}' rgma | sort -n | \
awk '{ data[NR] = $1 };
    END { min = data[1]; max = data[NR]; range = max - min;
        n_int = ( int(NR/10) > 1000 ? 1000 : int(NR/10) )
        size = range / n_int;
        mn_int = min + size/2;
        for(i=1; i<=NR; i++){
            aux = int((data[i] - min)/size) + 1;
            q[aux]++;
        };
        for(i=1; i<=n_int; i++){
            if( q[i] < 1 ) q[i] = 1;
            print mn_int, q[i];
            mn_int += size;
        };
    }' > a.d

awk '{print ($5>$7?log($5-$7)/log(2):0), \
      ($9>$11?log($9-$11)/log(2):0)}' tempo0 >logRlogG

#rm tempo0

```