Use of ***color_seq.pl*** and ***color_nums.pl*** scripts

Ryan Koehler 5/26/14
ryan@verdascend.com

Background and overview

The script*s* ***color_seq.pl*** and ***color_nums.pl*** are simple command line filters intended to colorize specific value-associated “words” contained within simple ASCII text. This is useful to highlight specific values for quick value identification and pattern visualization. Both tools take text from a file or standard input then output this with ANSI escape sequences interjected to color specific characters in a terminal display. Both tools also accept various command line options to allow user control over character or token selection and color choice.

This document briefly describes options available for these tools and, probably more importantly, provides example uses. Given that the point of these tools is to colorize text, examples are accompanied by screen captures illustrating outputs. (And this documentation is in pdf format to allow these graphics).

color_seq.pl is for coloring DNA sequences. The default behavior is to color what appears to be sequences with a four-color scheme associated with A C G and T. The script contains logic to guess whether a “word” (or token; continuous block of characters) is sequence-containing or not based on the fraction of ACGT characters; Words deemed to *not* be sequence-containing are not colored. By default, entire lines beginning with “#” or “>” are not colored, as these traditionally are associated with comments or the header lines of *fasta* format files. This behavior can be changed via command line switch.

In addition to simply finding and coloring DNA sequences, ***color_seq.pl*** can be used to highlight attributes often encountered in sequence files. For example, lower case sequence can be ignored (i.e. not colored), or upper and lower case sequence can be differentiated by bold / non-bold coloring. Another use is to highlight non-ACGT characters within (what appears to be) sequence. This is useful to quickly find IUB characters and any non-sequence characters (both colored differently to stand out).

Finally, ***color_seq.pl*** can be used to highlight *window-based* sequence features. With this feature runs of contiguous characters containing, as well as not containing, a specific base are highlighted. This feature supports IUB degenerate codes in addition to the normal ACGT bases. For example, using window-based highlighting on can readily identify runs of GC (IUB code “S”), runs of purines (IUB code “R”), etc. In addition to highlighting windows containing the specified base (or bases corresponding to an IUB code), windows *not* containing runs of the chosen base(s) are also highlighted in a second color.

color_nums.pl is for coloring numbers. The default behavior is to highlight numbers within text using a single color (This is distinct from simply coloring numeral characters). By default, entire lines beginning with “#” are not colored, as these traditionally are associated with comments, but this behavior can be changed via command line switch. It is also possible to only color a subset of number-containing “columns” via command line switch.

Assigning colors based on value allows rapid identification of “high” or “low” values. In addition, coloring a whole table of numbers together can serve as s simple means to visualize patterns in the data.

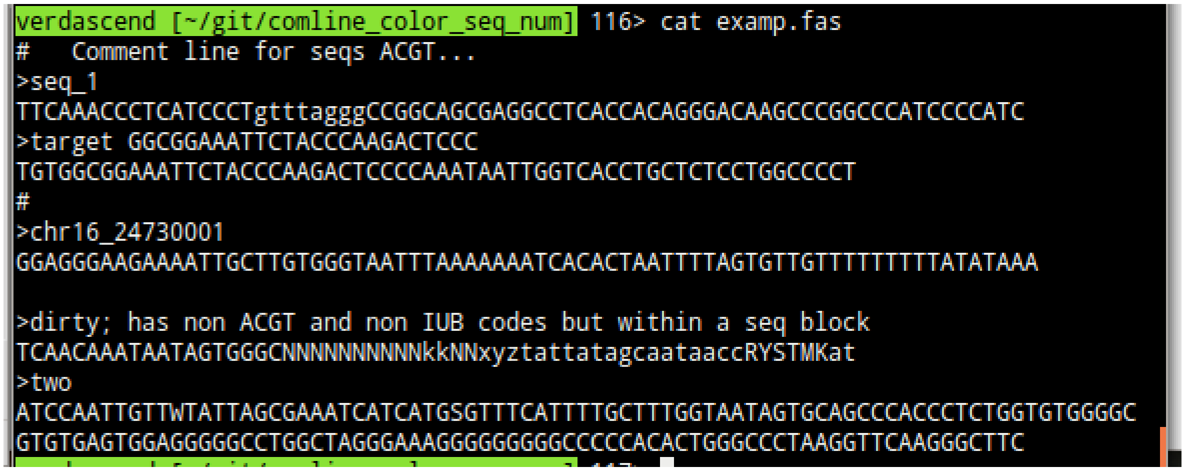
Use of ***color_seq.pl***

Calling ***color_seq.pl*** without any arguments (or with "-help" or an unrecognized argument) yields this splash / help screen:

```
=============================================================================
color_seq.pl V0.51; RTK 3/14/14

Usage: <infile> ['-' for stdin] [...options]
  <infile>   Sequence file
  -cabi      Color scheme 'ABI' style
  -cgc       Color scheme GC-warm / AT-cool style
  -win X     Color by windows of base type X (IUB is OK)
  -ws #      Window size #; Default is 5
  -nacgt     Only color non-ACGT bases; IUB = red; Other = blue
  -bc        Bold case (i.e. upper = bold, lower not)
  -lw        Lowercase white (i.e. upper = color, lower no)
  -all       Color all lines; Default ignores fasta '>' and comment '#'
  -verb      Verbose; print color mapping
=============================================================================
```

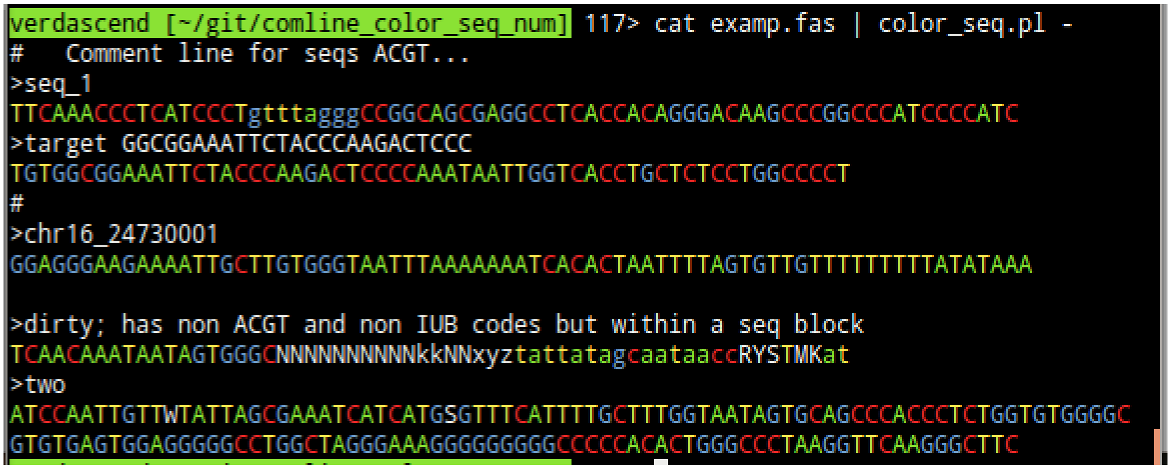
Each command line switch has a brief description of what it does. Given only input and no command switches, default coloring is performed. Figure 1 shows a test-case sequence file displayed via the *cat* command. Figure 2 shows this same content with output from *cat* piped through ***color_seq.pl***, which is given a "naked dash" to indicate input is from standard input (*stdin*). Other color schemes may be specified with the `-cabi` and `-cgc` switches. The result of these is shown in Figure 3, with "ABI" style first and GC-based coloring second.



```
verdascend [~/git/comline_color_seq_num] 116> cat examp.fas
#   Comment line for seqs ACGT...
>seq_1
TTCAAACCCTCATCCCTgtttagggCCGGCAGCGAGGCCTCACCACAGGGACAAGCCCGGCCCATCCCCATC
>target GGCGGAAATTCTACCCAAGACTCCC
TGTGGCGGAAATTCTACCCAAGACTCCCCAAATAATTGGTCACCTGCTCTCCTGGCCCCT
#
>chr16_24730001
GGAGGGAAGAAAATTGCTTGTGGGTAATTTAAAAAAATCACACTAATTTTAGTGTTGTTTTTTTTTATATAAA

>dirty; has non ACGT and non IUB codes but within a seq block
TCAACAAATAATAGTGGGCNNNNNNNNNNkkNNxyztattatagcaataaccRYSTMKat
>two
ATCCAATTGTTWTATTAGCGAAATCATCATGSGTTTCATTTTGCTTTGGTAATAGTGCAGCCCACCCTCTGGTGTGGGGC
GTGTGAGTGGAGGGGGCCTGGCTAGGGAAAGGGGGGGGGCCCCCACACTGGGCCCTAAGGTTCAAGGGCTTC
```

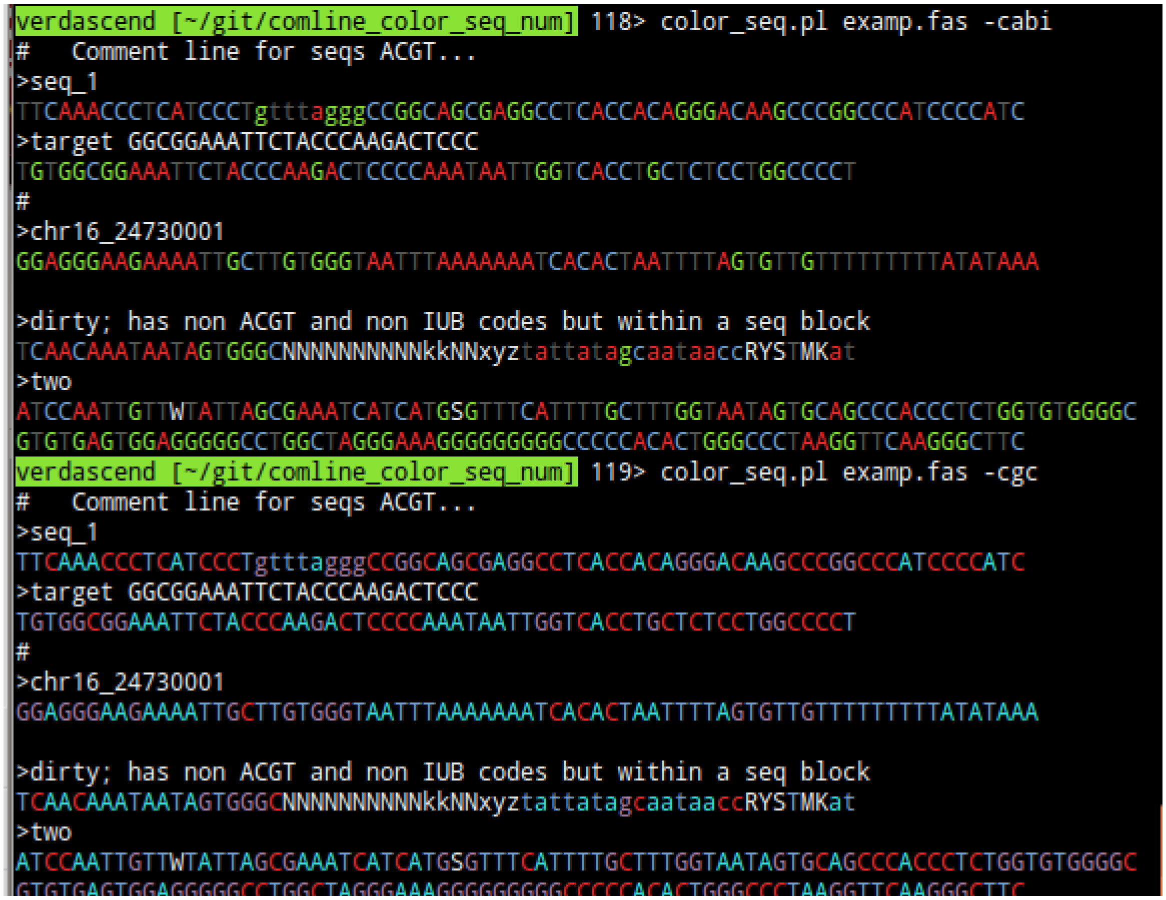
Figure 1. Test-case sequence file.



```
verdascend [~/git/comline_color_seq_num] 117> cat examp.fas | color_seq.pl -
#   Comment line for seqs ACGT...
>seq_1
TTCAAACCCTCATCCCTgtttagggCCGGCAGCGAGGCCTCACCACAGGGACAAGCCCGGCCCATCCCCATC
>target GGCGGAAATTCTACCCAAGACTCCC
TGTGGCGGAAATTCTACCCAAGACTCCCCAAATAATTGGTCACCTGCTCTCCTGGCCCCT
#
>chr16_24730001
GGAGGGAAGAAAATTGCTTGTGGGTAATTTAAAAAAATCACACTAATTTTAGTGTTGTTTTTTTTTTATATAAA

>dirty; has non ACGT and non IUB codes but within a seq block
TCAACAAATAATAGTGGGCNNNNNNNNNNkkNNxyztattatagcaataaccRYSTMKat
>two
ATCCAATTGTTWTATTAGCGAAATCATCATGSGTTTCATTTTGCTTTGGTAATAGTGCAGCCCACCCTCTGGTGTGGGGC
GTGTGAGTGGAGGGGGCCTGGCTAGGGAAAGGGGGGGGGCCCCCACACTGGGCCCTAAGGTTCAAGGGCTTC
```

Figure 2. Default coloring. Test-case sequence 'piped' as input. The command switch '-' is used to indicate input will come from *stdin*.



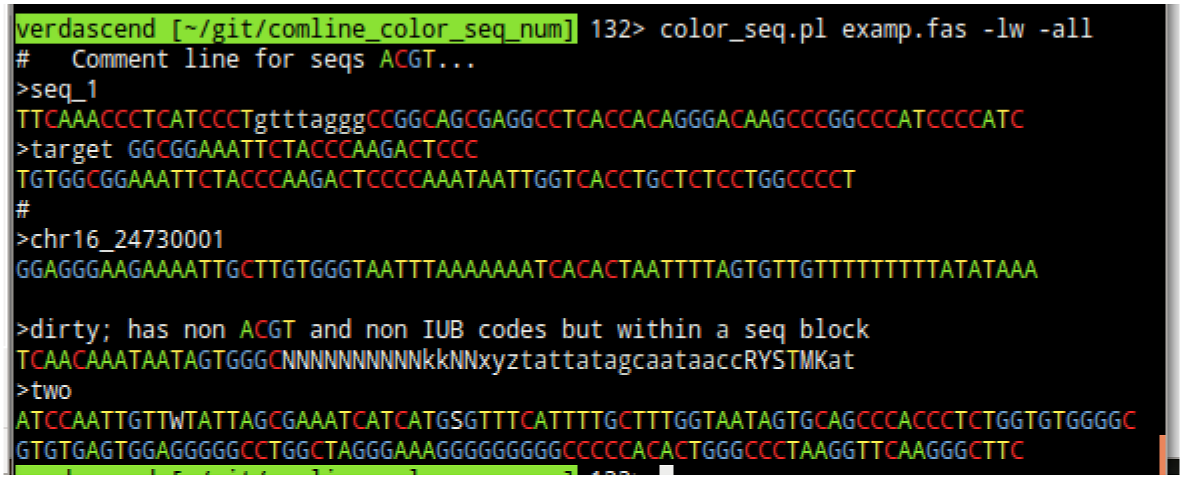
```
verdascend [~/git/comline_color_seq_num] 118> color_seq.pl examp.fas -cabi
#   Comment line for seqs ACGT...
>seq_1
TTCAAACCCTCATCCCTgtttagggCCGGCAGCGAGGCCTCACCACAGGGACAAGCCCGGCCCATCCCCATC
>target GGCGGAAATTCTACCCAAGACTCCC
TGTGGCGGAAATTCTACCCAAGACTCCCCAAATAATTGGTCACCTGCTCTCCTGGCCCCT
#
>chr16_24730001
GGAGGGAAGAAAATTGCTTGTGGGTAATTTAAAAAAATCACACTAATTTTAGTGTTGTTTTTTTTTTATATAAA

>dirty; has non ACGT and non IUB codes but within a seq block
TCAACAAATAATAGTGGGCNNNNNNNNNNkkNNxyztattatagcaataaccRYSTMKat
>two
ATCCAATTGTTWTATTAGCGAAATCATCATGSGTTTCATTTTGCTTTGGTAATAGTGCAGCCCACCCTCTGGTGTGGGGC
GTGTGAGTGGAGGGGGCCTGGCTAGGGAAAGGGGGGGGGCCCCCACACTGGGCCCTAAGGTTCAAGGGCTTC
verdascend [~/git/comline_color_seq_num] 119> color_seq.pl examp.fas -cgc
#   Comment line for seqs ACGT...
>seq_1
TTCAAACCCTCATCCCTgtttagggCCGGCAGCGAGGCCTCACCACAGGGACAAGCCCGGCCCATCCCCATC
>target GGCGGAAATTCTACCCAAGACTCCC
TGTGGCGGAAATTCTACCCAAGACTCCCCAAATAATTGGTCACCTGCTCTCCTGGCCCCT
#
>chr16_24730001
GGAGGGAAGAAAATTGCTTGTGGGTAATTTAAAAAAATCACACTAATTTTAGTGTTGTTTTTTTTTTATATAAA

>dirty; has non ACGT and non IUB codes but within a seq block
TCAACAAATAATAGTGGGCNNNNNNNNNNkkNNxyztattatagcaataaccRYSTMKat
>two
ATCCAATTGTTWTATTAGCGAAATCATCATGSGTTTCATTTTGCTTTGGTAATAGTGCAGCCCACCCTCTGGTGTGGGGC
GTGTGAGTGGAGGGGGCCTGGCTAGGGAAAGGGGGGGGGCCCCCACACTGGGCCCTAAGGTTCAAGGGCTTC
```

Figure 3. ABI and GC-based coloring schemes.

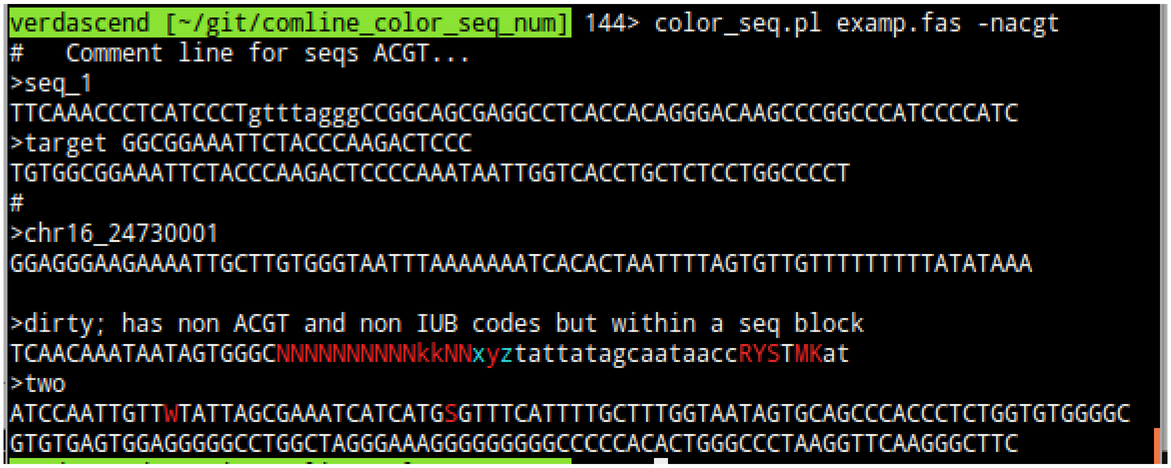
Some options can be combined, such as the `-lw` and `-all` switches, which, respectively, direct the script to color lowercase characters white (i.e. do not color them) and to include all lines (i.e. do not ignore lines starting with “>” or “#”). The result is shown in Figure 4, which uses the default color scheme. Note that the “ACGT” in the comment on the first line is colored, while the lowercase sequence characters on the third line (and further down) are not. Highlighting non-ACGT within sequence-like strings is showin in Figure 5 with the `-nacgt` switch. Normal ACGT bases are not colored, IUB code letters are colored red, and other (non-DNA) characters are colored cyan.



```
verdascend [~/git/comline_color_seq_num] 132> color_seq.pl examp.fas -lw -all
#   Comment line for seqs ACGT...
>seq_1
TTCAAACCCTCATCCCTgtttagggCCGGCAGCGAGGCCTCACCACAGGGACAAGCCCGGCCCATCCCCATC
>target GGCGGAAATTCTACCCAAGACTCCC
TGTGGCGGAAATTCTACCCAAGACTCCCCAAATAATTGGTCACCTGCTCTCCTGGCCCCT
#
>chr16_24730001
GGAGGGAAGAAAATTGCTTGTGGGTAATTTAAAAAAATCACACTAATTTTAGTGTTGTTTTTTTTTATATAAA

>dirty; has non ACGT and non IUB codes but within a seq block
TCAACAAATAATAGTGGGCNNNNNNNNNNkkNNxyztattatagcaataaccRYSTMKat
>two
ATCCAATTGTTWTATTAGCGAAATCATCATGSGTTTCATTTTGCTTTGGTAATAGTGCAGCCCACCCTCTGGTGTGGGGC
GTGTGAGTGGAGGGGGCCTGGCTAGGGAAAGGGGGGGGGCCCCCACACTGGGCCCTAAGGTTCAAGGGCTTC
```

Figure 4. Changing which characters are considered for coloring. The `-lw` and `-all` switches result in no color for lower case characters, and all lines being “colorable”, respectively.



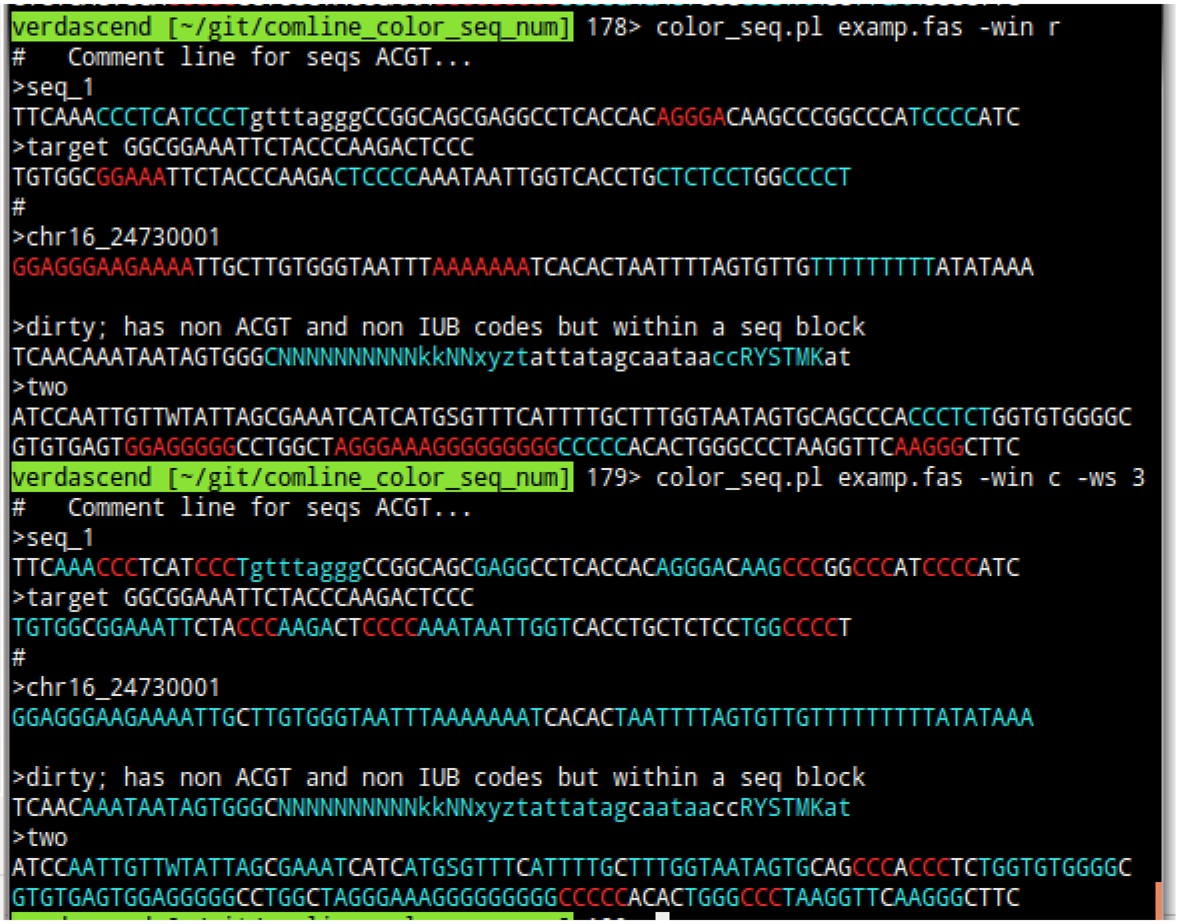
```
verdascend [~/git/comline_color_seq_num] 144> color_seq.pl examp.fas -nacgt
#   Comment line for seqs ACGT...
>seq_1
TTCAAACCCTCATCCCTgtttagggCCGGCAGCGAGGCCTCACCACAGGGACAAGCCCGGCCCATCCCCATC
>target GGCGGAAATTCTACCCAAGACTCCC
TGTGGCGGAAATTCTACCCAAGACTCCCCAAATAATTGGTCACCTGCTCTCCTGGCCCCT
#
>chr16_24730001
GGAGGGAAGAAAATTGCTTGTGGGTAATTTAAAAAAATCACACTAATTTTAGTGTTGTTTTTTTTTATATAAA

>dirty; has non ACGT and non IUB codes but within a seq block
TCAACAAATAATAGTGGGCNNNNNNNNNNkkNNxyztattatagcaataaccRYSTMKat
>two
ATCCAATTGTTWTATTAGCGAAATCATCATGSGTTTCATTTTGCTTTGGTAATAGTGCAGCCCACCCTCTGGTGTGGGGC
GTGTGAGTGGAGGGGGCCTGGCTAGGGAAAGGGGGGGGGCCCCCACACTGGGCCCTAAGGTTCAAGGGCTTC
```

Figure 5. Highlighting non-ACGT characters within sequence-containing strings.

Highlighting *window-based* sequence features is shown in Figure 6. Contiguous runs of specified base characters, and runs *devoid* of these, are colored within sequence-containing strings, while the remaining sequence characters are not colored. Two parameters define windowing behavior: base type and window size. Base type may be ACGT or any IUB code. Window size dictates the minimal length of runs that qualify for coloring. Runs of selected bases (and non-selected bases) that are at least window size long are colored; Selected base type(s) are red, and non-selected base type(s) are cyan.

NOTE*. Only ACGT bases qualify for selection and lines are considered independently. This means that IUB characters within strings do not count for runs, and runs spanning multiple lines are not colored.... (Maybe in a future version???)*



```
verdascend [~/git/comline_color_seq_num] 178> color_seq.pl examp.fas -win r
#   Comment line for seqs ACGT...
>seq_1
TTCAAACCCTCATCCCTgtttagggCCGGCAGCGAGGCCTCACCACAGGGACAAGCCCGGCCCATCCCCATC
>target GGCGGAAATTCTACCCAAGACTCCC
TGTGGCGGAAATTCTACCCAAGACTCCCCAAATAATTGGTCACCTGCTCTCCTGGCCCCT
#
>chr16_24730001
GGAGGGAAGAAAATTGCTTGTGGGTAATTTAAAAAAATCACACTAATTTTAGTGTTGTTTTTTTTTATATAAA

>dirty; has non ACGT and non IUB codes but within a seq block
TCAACAAATAATAGTGGGCNNNNNNNNNNkkNNxyztattatagcaataaccRYSTMKat
>two
ATCCAATTGTTWTATTAGCGAAATCATCATGSGTTTCATTTTGCTTTGGTAATAGTGCAGCCCACCCTCTGGTGTGGGGC
GTGTGAGTGGAGGGGGCCTGGCTAGGGAAAGGGGGGGGGCCCCCACACTGGGCCCTAAGGTTCAAGGGCTTC
verdascend [~/git/comline_color_seq_num] 179> color_seq.pl examp.fas -win c -ws 3
#   Comment line for seqs ACGT...
>seq_1
TTCAAACCCTCATCCCTgtttagggCCGGCAGCGAGGCCTCACCACAGGGACAAGCCCGGCCCATCCCCATC
>target GGCGGAAATTCTACCCAAGACTCCC
TGTGGCGGAAATTCTACCCAAGACTCCCCAAATAATTGGTCACCTGCTCTCCTGGCCCCT
#
>chr16_24730001
GGAGGGAAGAAAATTGCTTGTGGGTAATTTAAAAAAATCACACTAATTTTAGTGTTGTTTTTTTTTATATAAA

>dirty; has non ACGT and non IUB codes but within a seq block
TCAACAAATAATAGTGGGCNNNNNNNNNNkkNNxyztattatagcaataaccRYSTMKat
>two
ATCCAATTGTTWTATTAGCGAAATCATCATGSGTTTCATTTTGCTTTGGTAATAGTGCAGCCCACCCTCTGGTGTGGGGC
GTGTGAGTGGAGGGGGCCTGGCTAGGGAAAGGGGGGGGGCCCCCACACTGGGCCCTAAGGTTCAAGGGCTTC
```

Figure 6. Coloring windows with like-base runs. Red denotes selected base type(s) and cyan denotes *absence* of selected base type. Top example highlights runs of purine bases (A or G) in red, indicated by IUB code “R”, using default window size (= 5). Bottom example highlights runs of C, with window size given as 3.

Use of ***color_num.pl***

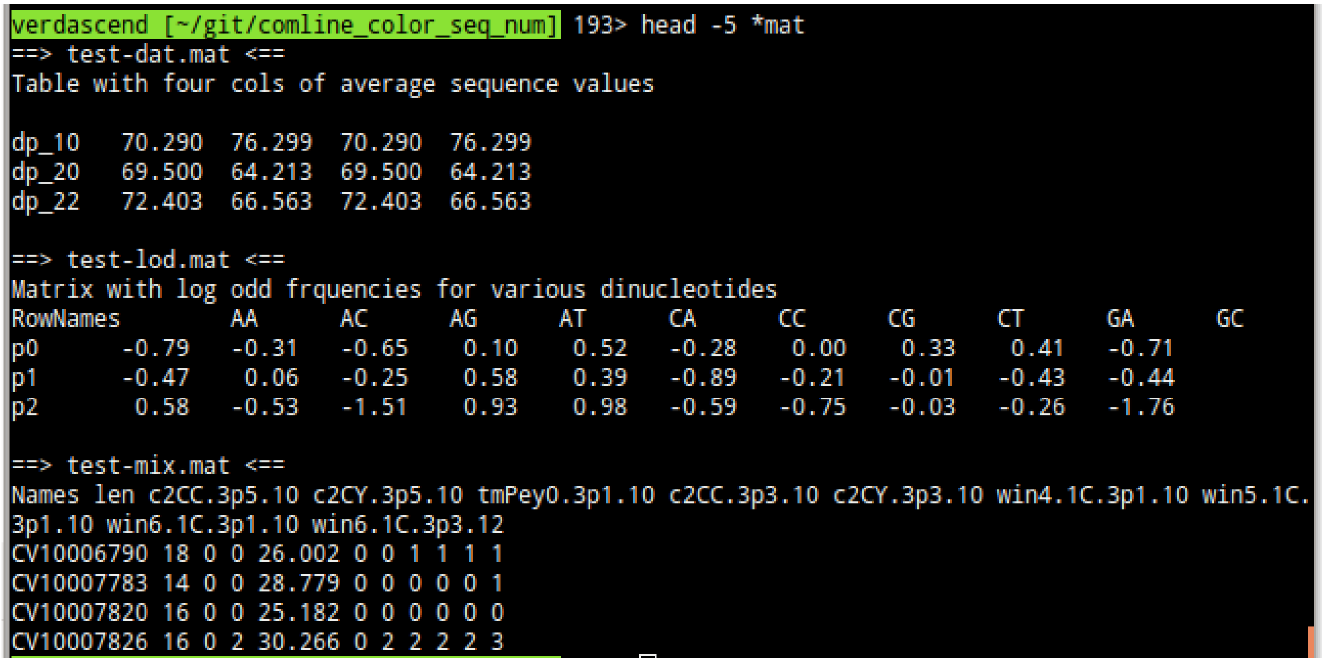
Calling ***color_num.pl*** without any arguments (or with “-help” or an unrecognized argument) yields this splash / help screen:

```
=============================================================================
color_nums.pl V0.21; RTK 4/4/14

Usage: <infile> ['-' for stdin] [...options]
  <infile>   Text file (e.g. matrix of numbers)
  -rwb       Color scheme Red White Blue
  -ryc       Color scheme Red Yellow Cyan
  -rg # #    Range from # to #
  -lt #      Less than #
  -gt #      Greater than #
  -nr        Normal range: Color 0 to 1
  -n2        2-sided normal range: Color -1 to 1
  -ok        Only qualifying numbers colored
  -iz        Ignore zero (i.e. no color)
  -col # #   Columns # to # (Token-based count)
  -not       Invert col qualifications
  -all       Color all lines; Default ignores comment '#'
  -verb      Verbose; print color mapping
=============================================================================
```

Each command line switch has a brief description of what it does. Given only input and no command switches, default coloring is performed. Figure 7 shows the first five lines of three number-containing files displayed via the *head* command*.* Figure 8 shows this same content with output from *head* piped through ***color_num.pl***, which is given a “naked dash” to indicate input is from standard input.

Specific number values can be highlighted. For example, the `-rg` switch can be used to specify a range of values; Numbers within the range are shown with one color, numbers less than the range are shown in another color, and numbers more than the range are shown in still another color. It is also possible to highlight *only* the within-range values using the `-ok` switch. Figure 9 shows this behavior, using a chosen range of 69 to 71. There are also options to differentiate only two classes of numbers; Those with values greater than or less than some chosen threshold and the rest.



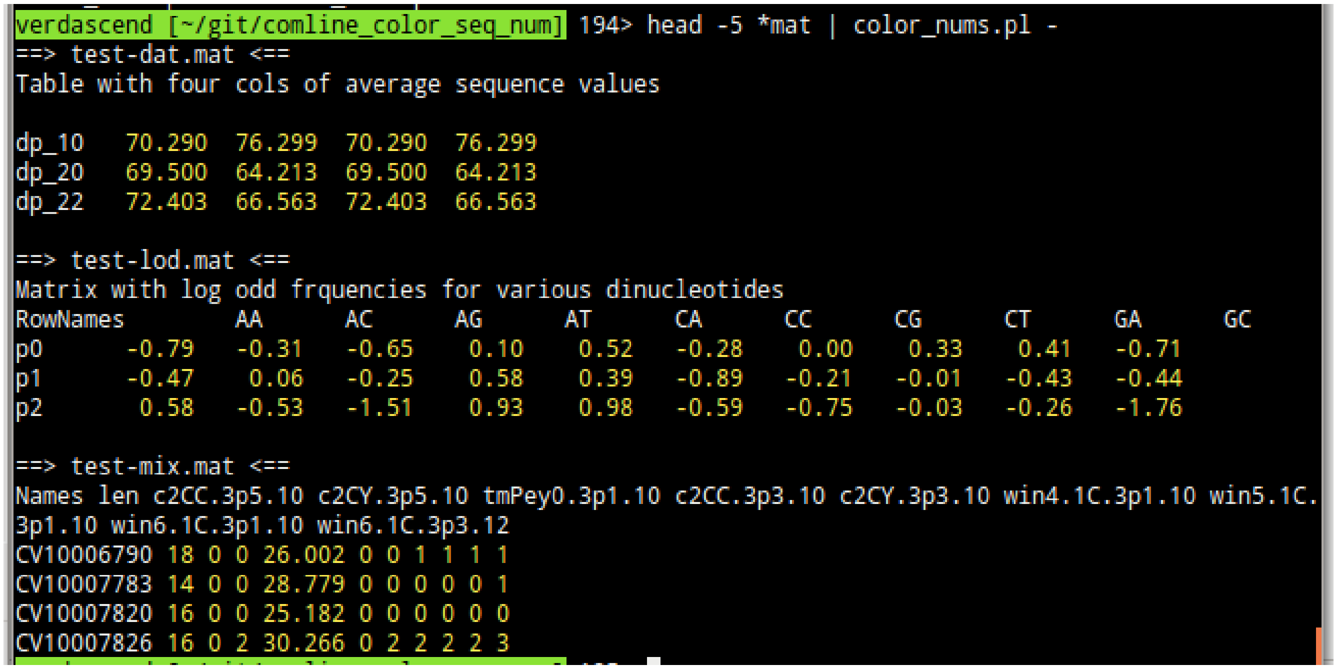
```
verdascend [~/git/comline_color_seq_num] 193> head -5 *mat
==> test-dat.mat <==
Table with four cols of average sequence values

dp_10   70.290  76.299  70.290  76.299
dp_20   69.500  64.213  69.500  64.213
dp_22   72.403  66.563  72.403  66.563

==> test-lod.mat <==
Matrix with log odd frquencies for various dinucleotides
RowNames        AA      AC      AG      AT      CA      CC      CG      CT      GA      GC
p0      -0.79   -0.31   -0.65    0.10    0.52   -0.28    0.00    0.33    0.41   -0.71
p1      -0.47    0.06   -0.25    0.58    0.39   -0.89   -0.21   -0.01   -0.43   -0.44
p2       0.58   -0.53   -1.51    0.93    0.98   -0.59   -0.75   -0.03   -0.26   -1.76

==> test-mix.mat <==
Names len c2CC.3p5.10 c2CY.3p5.10 tmPey0.3p1.10 c2CC.3p3.10 c2CY.3p3.10 win4.1C.3p1.10 win5.1C.
3p1.10 win6.1C.3p1.10 win6.1C.3p3.12
CV10006790 18 0 0 26.002 0 0 1 1 1 1
CV10007783 14 0 0 28.779 0 0 0 0 0 1
CV10007820 16 0 0 25.182 0 0 0 0 0 0
CV10007826 16 0 2 30.266 0 2 2 2 2 3
```

Figure 7. Test-case number-containing files (first five lines for each of three).



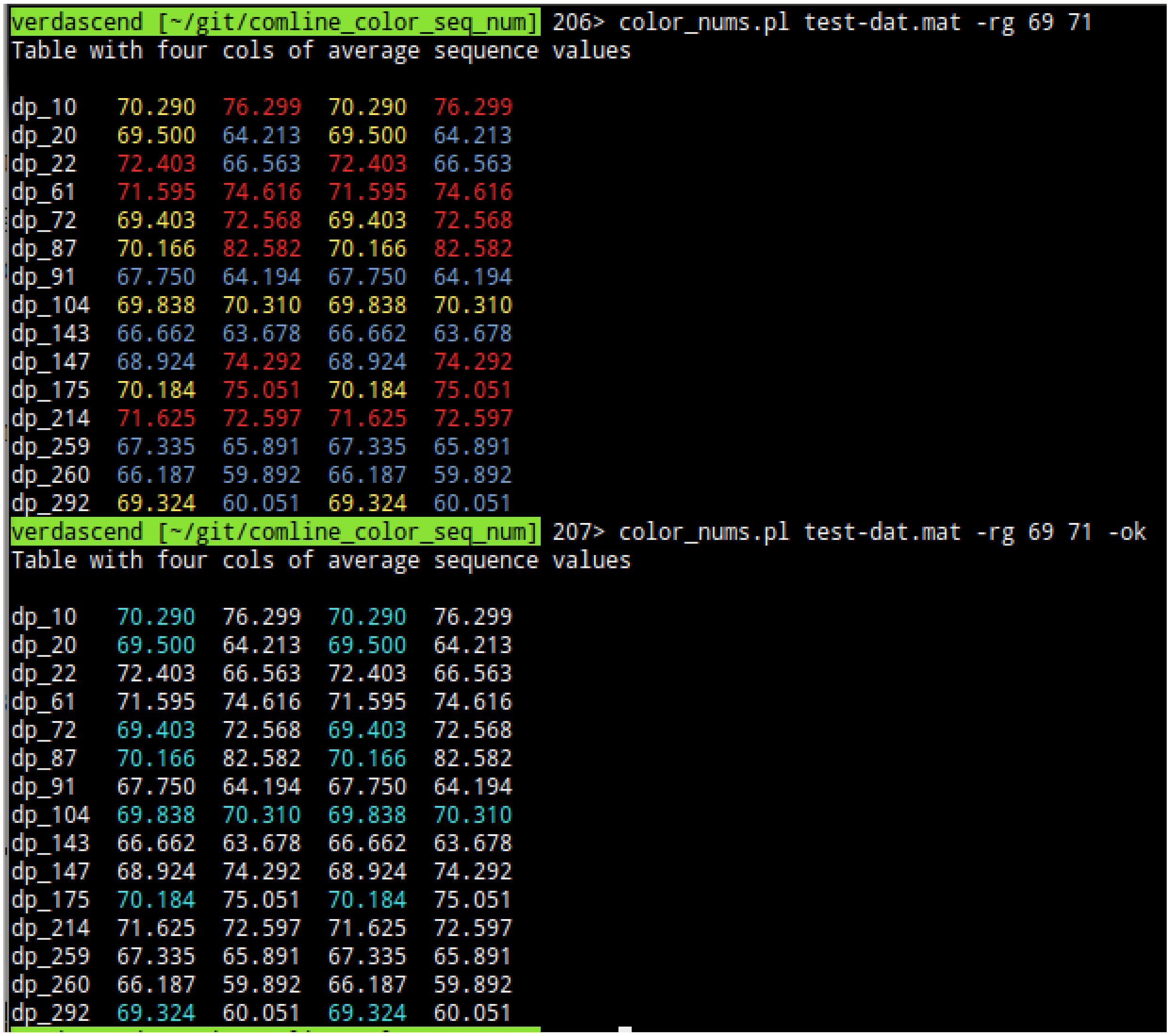
```
verdascend [~/git/comline_color_seq_num] 194> head -5 *mat | color_nums.pl -
==> test-dat.mat <==
Table with four cols of average sequence values

dp_10   70.290  76.299  70.290  76.299
dp_20   69.500  64.213  69.500  64.213
dp_22   72.403  66.563  72.403  66.563

==> test-lod.mat <==
Matrix with log odd frquencies for various dinucleotides
RowNames        AA      AC      AG      AT      CA      CC      CG      CT      GA      GC
p0      -0.79   -0.31   -0.65    0.10    0.52   -0.28    0.00    0.33    0.41   -0.71
p1      -0.47    0.06   -0.25    0.58    0.39   -0.89   -0.21   -0.01   -0.43   -0.44
p2       0.58   -0.53   -1.51    0.93    0.98   -0.59   -0.75   -0.03   -0.26   -1.76

==> test-mix.mat <==
Names len c2CC.3p5.10 c2CY.3p5.10 tmPey0.3p1.10 c2CC.3p3.10 c2CY.3p3.10 win4.1C.3p1.10 win5.1C.
3p1.10 win6.1C.3p1.10 win6.1C.3p3.12
CV10006790 18 0 0 26.002 0 0 1 1 1 1
CV10007783 14 0 0 28.779 0 0 0 0 0 1
CV10007820 16 0 0 25.182 0 0 0 0 0 0
CV10007826 16 0 2 30.266 0 2 2 2 2 3
```

Figure 8. Test-case number-containing files with default coloring.



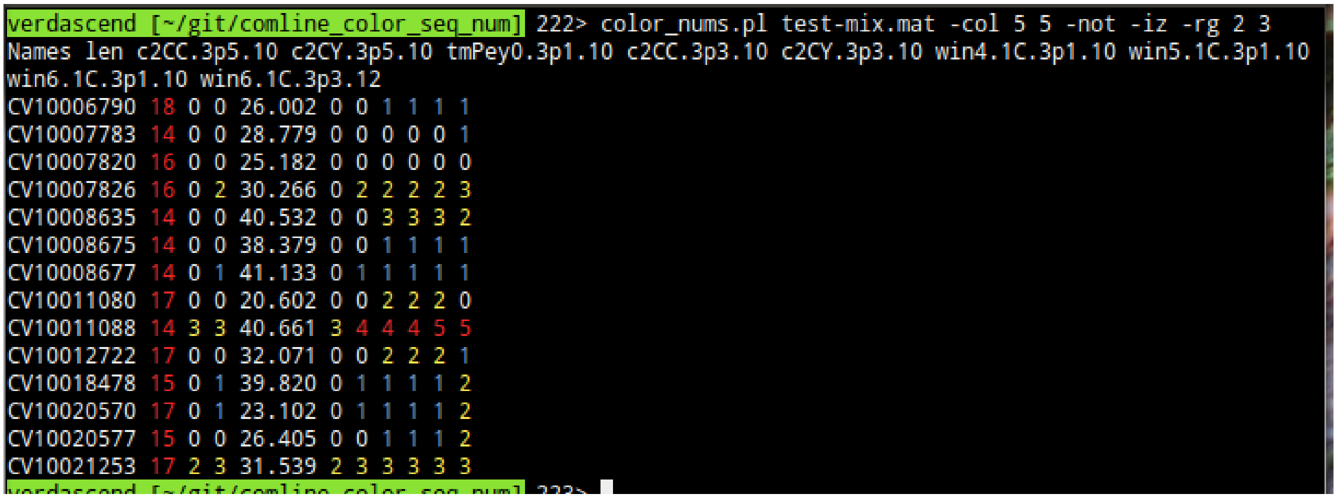
```
verdascend [~/git/comline_color_seq_num] 206> color_nums.pl test-dat.mat -rg 69 71
Table with four cols of average sequence values

dp_10   70.290  76.299  70.290  76.299
dp_20   69.500  64.213  69.500  64.213
dp_22   72.403  66.563  72.403  66.563
dp_61   71.595  74.616  71.595  74.616
dp_72   69.403  72.568  69.403  72.568
dp_87   70.166  82.582  70.166  82.582
dp_91   67.750  64.194  67.750  64.194
dp_104  69.838  70.310  69.838  70.310
dp_143  66.662  63.678  66.662  63.678
dp_147  68.924  74.292  68.924  74.292
dp_175  70.184  75.051  70.184  75.051
dp_214  71.625  72.597  71.625  72.597
dp_259  67.335  65.891  67.335  65.891
dp_260  66.187  59.892  66.187  59.892
dp_292  69.324  60.051  69.324  60.051
verdascend [~/git/comline_color_seq_num] 207> color_nums.pl test-dat.mat -rg 69 71 -ok
Table with four cols of average sequence values

dp_10   70.290  76.299  70.290  76.299
dp_20   69.500  64.213  69.500  64.213
dp_22   72.403  66.563  72.403  66.563
dp_61   71.595  74.616  71.595  74.616
dp_72   69.403  72.568  69.403  72.568
dp_87   70.166  82.582  70.166  82.582
dp_91   67.750  64.194  67.750  64.194
dp_104  69.838  70.310  69.838  70.310
dp_143  66.662  63.678  66.662  63.678
dp_147  68.924  74.292  68.924  74.292
dp_175  70.184  75.051  70.184  75.051
dp_214  71.625  72.597  71.625  72.597
dp_259  67.335  65.891  67.335  65.891
dp_260  66.187  59.892  66.187  59.892
dp_292  69.324  60.051  69.324  60.051
```

Figure 9. Specifying a range of values to highlight (e.g. 69 to 71).

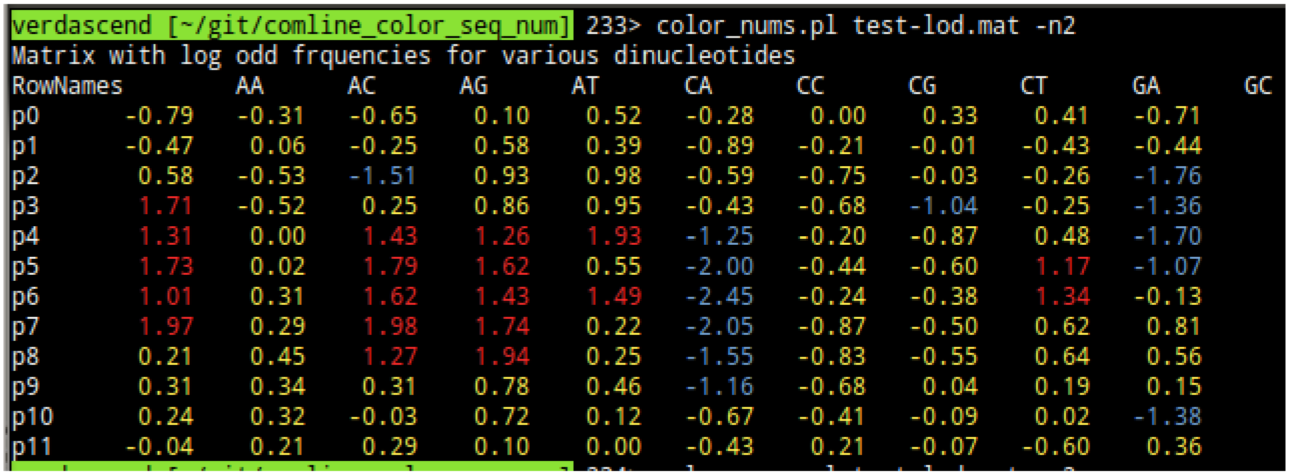
To better highlight specific numbers, subsets of (potentially) number-containing tokens may be ignored. Only values found in a range of columns can be colored, thus allowing some number-containing columns to be easily ignored. Column selection is done with the `-col` switch, which is associated with two number to specify start and end columns. Use of the `-not` switch inverts column selection criteria, so that numbers within selected columns are *not* colored but numbers in other columns may be colored. The `-iz` switch allows zero values to be ignored. Figure 10 shows the result of combining all three above mentioned options, in this case selecting all columns *except* column 5, specifying a coloring range from 2 to 3, and ignoring zeros.



```
verdascend [~/git/comline_color_seq_num] 222> color_nums.pl test-mix.mat -col 5 5 -not -iz -rg 2 3
Names len c2CC.3p5.10 c2CY.3p5.10 tmPey0.3p1.10 c2CC.3p3.10 c2CY.3p3.10 win4.1C.3p1.10 win5.1C.3p1.10
win6.1C.3p1.10 win6.1C.3p3.12
CV10006790 18 0 0 26.002 0 0 1 1 1 1
CV10007783 14 0 0 28.779 0 0 0 0 0 1
CV10007820 16 0 0 25.182 0 0 0 0 0 0
CV10007826 16 0 2 30.266 0 2 2 2 2 3
CV10008635 14 0 0 40.532 0 0 3 3 3 2
CV10008675 14 0 0 38.379 0 0 1 1 1 1
CV10008677 14 0 1 41.133 0 1 1 1 1 1
CV10011080 17 0 0 20.602 0 0 2 2 2 0
CV10011088 14 3 3 40.661 3 4 4 4 5 5
CV10012722 17 0 0 32.071 0 0 2 2 2 1
CV10018478 15 0 1 39.820 0 1 1 1 1 2
CV10020570 17 0 1 23.102 0 1 1 1 1 2
CV10020577 15 0 0 26.405 0 0 1 1 1 2
CV10021253 17 2 3 31.539 2 3 3 3 3 3
verdascend [~/git/comline_color_seq_num] 223>
```

Figure 10. Selecting specific columns, ignoring zero values and coloring by value range. All columns except column 5 are selected via the combination of switches `-col 5 5 -not` and zeros are ignored (i.e. not colored) because of the `-iz` switch. Finally, a color range of 2 to 3 is specified for differential value-based color assignment (i.e. in range values are yellow; under range values are blue; over range values are red).

Options for predefined ranges include the `-nr` switch for 0 to 1, and the `-n2` switch for -1 to 1. Figure 11 shows an example.



```
verdascend [~/git/comline_color_seq_num] 233> color_nums.pl test-lod.mat -n2
Matrix with log odd frquencies for various dinucleotides
RowNames        AA      AC      AG      AT      CA      CC      CG      CT      GA      GC
p0      -0.79   -0.31   -0.65    0.10    0.52   -0.28    0.00    0.33    0.41   -0.71
p1      -0.47    0.06   -0.25    0.58    0.39   -0.89   -0.21   -0.01   -0.43   -0.44
p2       0.58   -0.53   -1.51    0.93    0.98   -0.59   -0.75   -0.03   -0.26   -1.76
p3       1.71   -0.52    0.25    0.86    0.95   -0.43   -0.68   -1.04   -0.25   -1.36
p4       1.31    0.00    1.43    1.26    1.93   -1.25   -0.20   -0.87    0.48   -1.70
p5       1.73    0.02    1.79    1.62    0.55   -2.00   -0.44   -0.60    1.17   -1.07
p6       1.01    0.31    1.62    1.43    1.49   -2.45   -0.24   -0.38    1.34   -0.13
p7       1.97    0.29    1.98    1.74    0.22   -2.05   -0.87   -0.50    0.62    0.81
p8       0.21    0.45    1.27    1.94    0.25   -1.55   -0.83   -0.55    0.64    0.56
p9       0.31    0.34    0.31    0.78    0.46   -1.16   -0.68    0.04    0.19    0.15
p10      0.24    0.32   -0.03    0.72    0.12   -0.67   -0.41   -0.09    0.02   -1.38
p11     -0.04    0.21    0.29    0.10    0.00   -0.43    0.21   -0.07   -0.60    0.36
```

Figure 11. Predefined range coloring, from -1 to 1.