HANDYCNV: AN R PACKAGE FOR STANDARDIZED SUMMARY, ANNOTATION, COMPARISON, AND VISUALIZATION OF CNV AND CNVR

J. Zhou[1,2], L. Liu[1,2], T.J. Lopdell [3], D. Garrick[2*]and Y. Shi[1*]

[1]School of Agriculture, Ningxia University, Yinchuan, China
[2]AL Rae Centre of Genetics and Breeding, Massey University, Hamilton, New Zealand
[3] Research and Development, Livestock Improvement Corporation, Ruakura Road, Hamilton, New Zealand

SUMMARY

There is a need for a pipeline to provide standard, reproducible and timesaving post-analysis of CNV (copy number variants) from SNP (Single nucleotide polymorphisms) chip genotyping. We present a package built with a dozen functions that can convert the coordinates of SNP map files, compare the positions of SNPs between the given maps, summarize the CNVs, call CNVRs (Copy number variation regions), provide gene annotation, compare CNV, CNVR and the annotated gene lists, and visualize CNVs at both individual and population level.

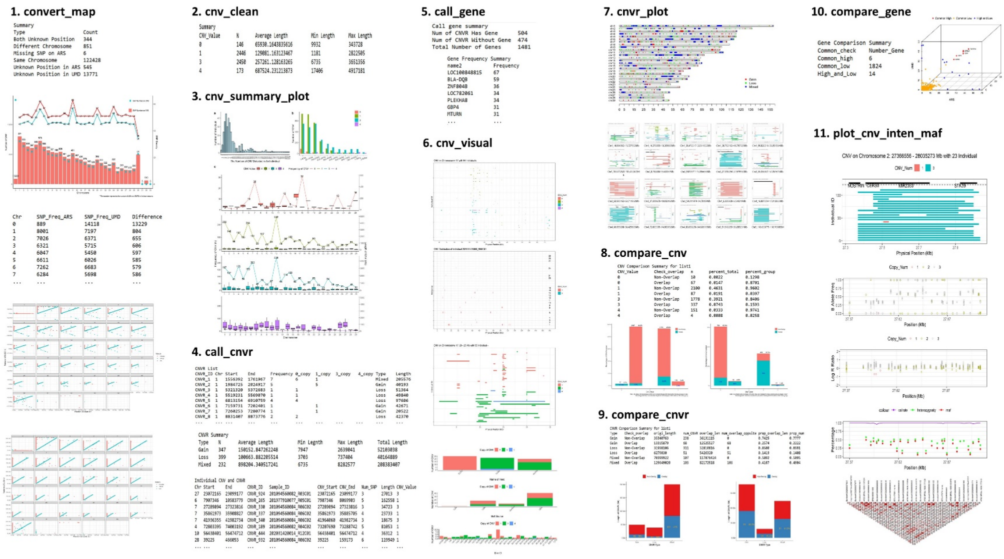
INTRODUCTION

Copy number variants are a type of structural variation of a DNA fragment which comprise the deletion or duplication type depending upon how many copies an individual has compared with the two copies in the diploid reference genome. These structure variants could change the structure or dosage of genes that might further affect the phenotypes. Studies on CNVs have become common in livestock research in recent years. The fluorescent signal intensity of SNPs chip provides the general source to detect CNV, and thanks to the wide application of genome-wide association studies and genomic selection in animal breeding, there is now a lot of SNP data suitable for analysis of structural variants. Software such as PennCNV (Wang *et al*. 2007), CNVPartition (Illumina) and SVS Golden Helix (Bozeman) are designed to detect CNV from SNP data, but each method has its own advantages and shortcomings, so it is recommended to use more than one method to infer CNVs (Winchester *et al*. 2009), therefore comparison of multiple CNV results is a normal task in characterizing structural variation.

When doing CNV analysis we are curious about all the information related to any CNV region, not only at the individual level but also at the population level. For instance, we want to know how many individuals have a CNV in a common region? What kind of type of CNVs are there? Are these individuals from the same farm or are they progenies of the same sire? Are there any genes in the CNV region? What are the gene frequencies? How about the signal intensities, call rate, minor allele frequency and linkage disequilibrium conditions of these CNVs? Besides, the common post-analysis of CNV studies includes provision of summary CNVs, generation of CNVR, comparison of CNVs from different software, finding consensus CNVR by comparing results to the gold standard CNV database, gene annotation in the CNV region and CNV-based regression analysis. To accomplish all these tasks various tools are typically required. Therefore, we integrated these functions into a package to make post-CNV analysis easy and reproducible. The use of some developed R package like the Tidyverse family (Wickham *et al*. 2019) made our function development much easier. We believe this package will be convenient to others who are doing similar work. The source code can be found in the Github repository (https://github.com/JH-Zhou/HandyCNV).

MATERIALS AND METHODS

The pipeline and results of this package are shown in Figure 1. The Demo Data are the CNV results from the GeneSeek GGP Bovine 150k BeadChip detected by PennCNV (Wang *et al*. 2007) and CNVPartition (Illumina). All the input files and demo code can be found in Github (https://github.com/JH-Zhou/HandyCNV). Run through all functions in HandyCNV need prepare CNV Results, SNP maps, Reference Gene List, Pedigree, Plink files (Bim, Bed and Fam) and SNP Signal Intensity in total. All the input files have a fixed format, and the file requirements depends on which functions the users are using. Here we only introduce the input data format and some noticeable methods we used in some functions, therefore, we will not cover the data structures or interpret how to use the results in this article, more details can be found by browsing our Demo Data.



```
Summary
Type                        Count
Both Unknown Position       344
Different Chromosome        851
Missing SNP on ARS          6
Same Chromosome             122428
Unknown Position in ARS 545
Unknown Position in UMD 13771
```

```
Chr     SNP_Freq_ARS    SNP_Freq_UMD    Difference
0       889             14118           13229
1       8001            7197            804
2       7026            6371            655
3       6321            5715            606
4       6047            5450            597
5       6611            6026            585
6       7262            6683            579
7       6284            5698            586
...     ...             ...             ...
```

```
Summary
CNV_Value   N     Average Length      Min Length   Max Length
0           146   65930.1643835616    9932         343728
1           2446  129081.163123467    1181         2822505
3           2450  257261.128163265    6735         3651356
4           173   687524.231213873    17406        4917181
```

```
CNVR List
CNVR_ID Chr Start   End     Frequency 0_copy 1_copy 3_copy  4_copy Type  Length
CNVR_1  1   1556392 1761967 7         6      1                     Mixed 205576
CNVR_2  1   1984725 2024917 5                5                     Gain  40193
CNVR_3  1   5321320 5372683 1         1                            Loss  51364
CNVR_4  1   5519231 5569070 1         1                            Loss  49840
CNVR_5  1   6813154 6910759 4         4                            Loss  97606
CNVR_6  1   7159731 7202401 1                1                     Gain  42671
CNVR_7  1   7260253 7280774 1                1                     Gain  20522
CNVR_8  1   8831407 8873776 2         2                            Loss  42370
...     ... ...     ...     ...       ...    ...    ...     ...    ...   ...
```

```
CNVR Summary
Type   N    Average Length     Min Legnth   Max Length   Total Length
Gain   347  150152.847262248   7947         2639041      52103038
Loss   399  100663.882205514   3703         737404       40164889
Mixed  232  898204.340517241   6735         8282577      208383407
```

```
Individual CNV and CNVR
Chr Start    End      CNVR_ID  Sample_ID            CNV_Start CNV_End  Num_SNP Length CNV_Value
27  23072165 23099177 CNVR_924 201094560082_R03C01  23072165  23099177 3       27013  3
6   7907346  10583779 CNVR_265 201977910077_R05C01  7907346   8069903  5       162558 1
7   27289094 27323816 CNVR_330 201094560084_R06C02  27289094  27323816 3       34723  3
7   35861973 35908827 CNVR_337 201094560084_R06C02  35861973  35885705 4       23733  1
7   41936355 41982734 CNVR_340 201094560084_R06C02  41964060  41982734 3       18675  3
4   72003395 74003192 CNVR_189 201094560082_R04C02  73207690  73288742 5       81053  1
10  56438401 56474712 CNVR_444 202031420014_R12C01  56438401  56474712 4       36312  1
28  39225    466053   CNVR_932 201094560084_R06C02  39225     159173   6       119949 1
... ...      ...      ...      ...                  ...       ...      ...     ...    ...
```

```
Call gene summary
Num of CNVR Has Gene       504
Num of CNVR Without Gene   474
Total Number of Genes      1481

      Gene Frequency Summary
      name2           Frequency
      LOC100848815    67
      BLA-DQB         59
      ZNF804B         36
      LOC782061       34
      PLEKHA8         34
      GBP4            31
      MTURN           31
      ...             ...
```

```
CNV Comparison Summary for list1
CNV_Value     Check_overlap   n      percent_total   percent_group
0             Non-Overlap     10     0.0022          0.1298
0             Overlap         67     0.0147          0.8701
1             Non-Overlap     2100   0.4631          0.9602
1             Overlap         87     0.0191          0.0397
3             Non-Overlap     1778   0.3921          0.8406
3             Overlap         337    0.0743          0.1593
4             Non-Overlap     151    0.0333          0.9741
4             Overlap         4      0.0008          0.0258
```

```
CNVR Comparison Summary for list1
Type   check_overlap  origi_length   num_CNVR overlap_len num_overlap_opposite prop_overlap_len prop_num
Gain   Non-Overlap    35340763       238      36131115    0                    0.7425           0.7777
Gain   Overlap        13315879       68       12525527    68                   0.2574           0.2222
Loss   Non-Overlap    31958186       311      32810816    0                    0.8588           0.8591
Loss   Overlap        6275030        51       5426320     51                   0.1419           0.1408
Mixed  Non-Overlap    70399922       107      117876424   0                    0.5892           0.5095
Mixed  Overlap        129640820      103      82172518    106                  0.4107           0.4904
```

```
Gene Comparison Summary
Common_check    Number_Gene
Common_high     6
Common_low      1824
High_and_Low    14
```

Figure 1. Pipeline and results of HandyCNV for the post-analysis of CNV

The first function is convert_map which is used to convert map files from the original to an objective map file provided by the user. There are differences between genome assemblies, for example, in which some SNP might locate on a different chromosome or on the same chromosome but in a different order between different assemblies. Most Bovine SNP chips have been using the UMD3.1 (Shamimuzzaman *et al.* 2019) as the default reference genome assembly, but with the release of new reference genome ARS-UCD1.2 with high continuity, accuracy, and completeness (Rosen *et al*. 2020),it may be of interest to convert the coordinates to the latest assembly to help further research. Four columns are required with no header in the input map files whose columns are Chromosome, SNP ID, Morgan Position (UMD) or Physical Position (ARS) and Physical Position (unit: bp) (Table 1, Table 2).

Table 1. Original map format (UMD 3.1)

14	ARS-BFGL-BAC-10172	0	6371334
14	ARS-BFGL-BAC-1020	0	7928189
14	ARS-BFGL-BAC-10245	28.23	31819743
14	ARS-BFGL-BAC-10345	0	6133529

Table 2. Objective map format (ARS)

14	ARS-BFGL-BAC-10172	5.34266	5342658
14	ARS-BFGL-BAC-1020	6.88966	6889656
14	ARS-BFGL-BAC-10245	30.1241	30124134
14	ARS-BFGL-BAC-10345	5.10573	5105727

The cnv_clean function is designed to convert the CNV results to a standard format, the output clean CNV file is used as input data in many of the other functions. It supports PennCNV and CNVPartition default output results, the length of CNVs are calculated as one plus the end position minus the start position. The CNV results from other software can be prepared as the template format to use in the remaining functions (Table 3). The function cnv_summary_plot will generate several plots to show the number, length group, type, and frequency details of CNVs on individuals and on chromosomes.

Table 3 Template Format of Clean CNV

Sample_ID	Chr	Start	End	CNV_Value	Length
201094560060_R02C01	11	106224443	106359588	4	135146
201094560060_R02C01	12	58073538	58417437	1	343900
201094560060_R02C01	19	27576066	27643677	4	67612
201094560060_R02C02	1	88638760	88904687	3	265928

The call_cnvr function will merge the CNVs which have at least one bp overlapping length to a CNVR. The results are the non-redundant CNVRs,but this method could cause misleading information while reporting the genes and comparing the overlapping length on CNVR. This is because it may appear all CNVs in a CNVR are the same length but in reality there are often lots of short disparate CNVs. To solve this problem, combine call_gene and cnv_visual function will plot all genes located on CNVs of every individual in a CNVR. The call_gene function needs the user to provide the reference genes which can be downloaded from the UCSC website (http://hgdownload.soe.ucsc.edu/downloads.html).

The compare_cnv and compare_cnvr functions with the similar strategies, when the results have the same version coordinates they will compare directly, but when the coordinates are from different versions, it will convert the position for each file at first then make comparison between the coordinates of the latest version. The overlapped region between two interval results may be slightly different, when reporting and plotting the number and length of overlapping regions correspond to each input files, respectively.

RESULTS AND DISCUSSION

When do you need to convert the coordinates of SNP or CNV? The first scenario is when a new reference genome is released. Take the Bovine reference genome as example, the lasted version (ARS-UCD1.2) has higher coverage and accuracy of its genome assembly than the previous commonly used UMD3.1, so it may help to improve the accuracy of SNP-based CNVs detection by using the latest reference genome. The second scenario is to make comparison between results from different reference genomes. There are lots of studies that have reported CNVs using previous reference genomes, and we may want to compare their results to our assembly.

Why do we need to visualize CNVR? CNV is of interest relative to the comparison of individuals but the CNVR are mostly of interest at the population level. The common method to generate CNVR is to merge all overlapping CNVs from every individual into a common region, then make gene annotation and comparison on CNVRs on the population level.

The main shortcoming of SNP-based CNV detection is that it cannot report the exact start or end position because of the limited marker density, so when we merge these CNV intervals to a common CNVR the actual situation is that not all the CNVs with the same break points as the CNVR, therefore, not all the genes within a CNVR has the same frequency with the CNVs (Figure 1. 11). Sometimes we might find an interested candidate gene within a high frequency CNVR but if only a few individuals have CNV of that gene, the better way to avoid this mistake is to report the gene frequency by counting how many CNVs with this gene in a CNVR, but this not enough, because of some genes may have CNV in just a partial fragment rather than the entire gene, in this circumstance plotting all CNVs and annotated genes in a CNVR by the start and end position can make it much clearer to understand what is happening. We are often curious about all the information in a CNVR in a population, such as the relationship between SNPs in that region, so visualizing a CNVR by plotting all related information in one figure is a good solution.

What are the limitations of this study? First, we have only used it in bovine studies, so some functions may need to be revised to be used in other species. Second, the linkage disequilibrium (LD) plots are based on the Gaston package which was drawing the base plot only, for some CNVRs with fewer number of SNPs the plot size was not well controlled while merging it to other plots, and this could lead some CNVR plots to be unsuitable without further modifications. Third, plots of CNVs on the population level are suitable for small populations but could be too busy for large populations. Fourth, the functions for regression analysis between CNVs or CNVRs and phenotype are still being developed.

CONCLUSIONS

Here we present an R package called HandyCNV in the initial version which includes several functions for tasks such as converting SNP maps, generating CNVR, genome annotation, comparing and visualizing of CNV and CNVR and reporting summary results on each step. This tool provides a standard, reproducible and timesaving post-analysis of copy number variants.

ACKNOWLEDGEMENTS

This project was funded by the China Scholarship Council. This package depends on several developed R packages, such as Tidyverse family and data.table, we appreciate all related contributors to the open source R language.

REFERENCES

Wang K., Li M., Hadley D., et al. Penn C.N.V. (2007) *Genome Res*. **17**(11): 1665.

Illumina. GenomeStudio. https://www.illumina.com/techniques/microarrays/array-data-analysis-experimental-design/genomestudio.html

Bozeman, MT: Golden Helix I. SNP & Variation Suite ™ (Version 8.x). http://www.goldenhelix.com

Winchester L., Yau C., and Ragoussis J. (2009) *Briefings Funct Genomics Proteomics*. **8**(5): 353.

Wickham H, Averick M, Bryan J, et al. (2019) *J Open Source Softw.*

Shamimuzzaman M., Le Tourneau J.J., Unni D.R., et al. (2019) *Nucleic Acids Res.*

Rosen BD, Bickhart DM, Schnabel RD, et al. (2020) *Gigascience.*