Sep 21, 2020

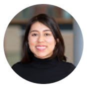
nf-vcf-cataloguer

DOI
dx.doi.org/10.17504/protocols.io.bkmzku76

Israel Aguilar Ordoñez[1]

[1]Instituto Nacional de Medicina Genómica (INMEGEN)

Whole genome variation...



Judith Ballesteros Villascan
Centro de Investigación y de Estudios Avanzados del IPN (Cin...

DOI: dx.doi.org/10.17504/protocols.io.bkmzku76

External link: https://github.com/laguilaror/nf-VCF-cataloguer

Protocol Citation: Israel Aguilar Ordoñez 2020. nf-vcf-cataloguer. **protocols.io**
https://dx.doi.org/10.17504/protocols.io.bkmzku76

Manuscript citation:
Aguilar-Ordoñez I, Pérez-Villatoro F, García-Ortiz H, Barajas-Olmos F, Ballesteros-Villascán J, González-Buenfil R, Fresno C, Garcíarrubio A, Fernández-López JC, Tovar H, Hernández-Lemus E, Orozco L, Soberón X, Morett E (2021) Whole genome variation in 27 Mexican indigenous populations, demographic and biomedical insights. PLoS ONE 16(4): e0249773. doi: 10.1371/journal.pone.0249773

Protocol status: Working
We use this protocol and it's working

Created: September 01, 2020

Last Modified: September 21, 2020

Protocol Integer ID: 41369

Abstract

'nf-vcf-cataloguer' is a tool, implemented in Nextflow, that generates a general table description in TSV format of the description of each category and subgroup of a VCF with the extended annotation made by VEP. Furthermore, it plots each subset of the consequences of variants.

Guidelines

Instalation

Download nf-vcf-cataloguer from Github repository:

```
git clone https://github.com/Iaguilaror/nf-vcf-cataloguer.git
```

Compatible OS*:

- Ubuntu 18.04.03 LTS

* nf-vcf-cataloguer may run in other UNIX based OS and versions, but testing is required.

Software Requirements:

Software	
bcftools	NAME

Software	
htslib	NAME

Software	
filter_vep	NAME

Software

Nextflow	NAME

Software

Plan9	NAME
https://github.com/9fans/plan9port	SOURCE LINK

Software

R	NAME

Materials

Pipeline Inputs

- A compressed VCF file with extension '.vcf.gz', which must have a TABIX index with .tbi extension, located in the same directory as the VCF file.

The header names the eight mandatory columns: CHROM, POS, ID, REF, ALT, QUAL, FILTER, INFO. INFO must contain "AN", which is the target for filtering of this module.
For more information about the VCF format, please go to the next link:**Variant Call Format**

Example line(s):

```
##fileformat=VCFv4.2 #CHROM  POS     ID      REF     ALT     QUAL    FILTER  INFO
chr21   5101724 .       G       A       .       PASS
AC=1;AF_mx=0.00641;AN=152;DP=903;nhomalt_mx=0;ANN=A|intron_variant|MODIFIER|GATD3B|ENS
G00000280071|Transcript|ENST00000624810.3|protein_coding||4/5|ENST00000624810.3:c.357+
19987C>T|||||||||-1|cds_start_NF&cds_end_NF|SNV|HGNC|HGNC:53816||5|||ENSP00000485439||
A0A096LP73|UPI0004F23660|||||||chr21:g.5101724G>A||||||||||||||||||||||||||||2.079|0.0
34663||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
chr21   5102165 rs1373489291    G       T       .       PASS
AC=1;AF_mx=0.00641;AN=140;DP=853;nhomalt_mx=0;ANN=T|intron_variant|MODIFIER|GATD3B|ENS
G00000280071|Transcript|ENST00000624810.3|protein_coding||4/5|ENST00000624810.3:c.357+
19546C>A|||||||rs1373489291||-1|cds_start_NF&cds_end_NF|SNV|HGNC|HGNC:53816||5|||ENSP0
0000485439||A0A096LP73|UPI0004F23660|||||||chr21:g.5102165G>T|||||||||||||||||||||||||
|||5.009|0.275409|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
|||||||||||
```

- A '.txt' selection signals file chich lists rsIDs.

- A reference file to extract certain fields of vcf and transform it to tsv format.

Dataset	
fields_to extract.txt	NAME

Before start

Test

To test nf-vcf-cataloguer's execution using test data, run:

```
./runtest.sh
```

Your console should print the Nextflow log for the run, once every process has been submitted, the following message will appear:

```
======
nf-vcf-cataloguer: Basic pipeline TEST SUCCESSFUL
======
```

nf-vcf-cataloguer results for test data should be in the following file:

```
nf-vcf-cataloguer/test/results/catgorizeVCF-results
```

Usage

To run nf-vcf-cataloguer go to the pipeline directory and execute:

```
nextflow run categorize-vcf.nf --vcf <path to input 1> [--output_dir path to results]
[-resume]
```

For information about options and parameters, run:

```
nextflow run categorize-vcf.nf --help
```

Pre-processing

1 **Custom filter**

Remove the variants that have the AN (total number of alleles in called genotypes) value assigned.

Note

a) Includes sites where the compressed VCF file '.vcf.gz' comply with the AN value.

Dependencies:

Software

bcftools NAME

2 **Separate SNVs and indels**

Keep only certain types of variants.

Note

a) Includes SNPs of a compressed VCF file '.vcf.gz'.
b) Includes indels of a compressed VCF file '.vcf.gz'.

Dependencies:

Software

bcftools NAME

3 **Separate rare, low and common frequencies**

Keep only certain types of variants, set by its allele frequency.

Note

a) Separate variants by its allele frequency category.
b) Separate variants in common frequency.
c) Separate variants in low frequency.
d) Separate variants in rare frequency.

Dependencies:

Software

bcftools NAME

4 **Separate selection signals**

Keep only variants with selection signals.

Note

a) Separate variants on selection, with a reference ID list of selection signals.
b) Sort output file.

Dependencies:

Software

bcftools NAME

5 **Separate low EAS and low EUR variants**

Note

a) Filter variants with more than 5% of allele frequency in the local population.
b) Filter variants with less than 5% of allele frequency in EAS and NFE gnomAD population.

Dependencies:

Software	
filter_vep	NAME

6 **Separate common AMR and low EUR variants**

Separate variants by its allele frequency comparing other populations of the *gnomAD database.*

Note

a) Filter variants with more than 5% of allele frequency in local population.
b) Filter variants with more than 5% of allele frequency in AMR gnomAD population.
c) Filter variants with less than 5% of allele frequency in NFE gnomAD population.

Dependencies:

Software	
filter_vep	NAME

Core-processing

7 **Get CLINVAR and OMIM variants**

Separate variants annotated by the *ClinVar database.*

Note

a) Separate variants annotated by ClinVar.
b) Extract OMIM variants.

Dependencies:

Software	
filter_vep	NAME

8 **Get GenHancer variants**

Separate variants with a GeneHancer ID.

Note

a) Filter variants that match with annotations in the "GeneHancer type and genes" field.

Dependencies:

Software	
filter_vep	NAME

9 **Get GWASCatalog variants**

Separate variants with a GeneHancer ID.

Note

a) Filter variants that match with annotations in "gwascatalog" field.

Dependencies:

Software	
filter_vep	NAME

10 **Get miRNAs variants**

Separate variants with miRNA data.

Note

a) Filter variants that match with annotations in "miRBase" field.

Dependencies:

Software

filter_vep NAME

11 **Get novel and known variants**

Separate known and unknown variants.

Note

a) Filters variants that have a rsID and are reported by dbSNP.
b) Separate unknown variants (without rsID in dbSNP).

Dependencies:

Software

filter_vep NAME

12 **Get coding variants**

Separate variants in coding regions.

Note

a) Separate exonic variants.
b) Filter intronic variants.

Dependencies:

Software	
filter_vep	NAME

13 **Get PGKB variants**

Separate variants found in PGKB database.

Note

a) Filter variants that match with annotations in "PGKB" field.

Dependencies:

Software	
filter_vep	NAME

14 **Get UTR variants**

Separate variants found in 5' or 3' UTR regions.

Note

a) Filter variants that are in 5' UTR.
b) Filter variants that are in 3' UTR.

Dependencies:

Software

filter_vep NAME

Pos-processing

15 **VCF to TSV**

Convert vcf files to tsv format.

Note

a) Search ANN header and separates it by tabs.
b) Separate columns by tabs.
c) Add a "." to blank spaces.

Dependencies:

Software

bcftools NAME

Final Output:

Expected result

A '.tsv.gz' file with columns of the VEP annotations, by each vcf converted.

```
CHROM   POS     ID      REF     ALT     AC      AN      DP
AF_mx   nhomalt_mx      Allele  Consequence     IMPACT  SYMBOL
Gene    Feature_type    Feature BIOTYPE EXON    INTRON  HGVSc
HGVSp   cDNA_position   CDS_position    Protein_position
Amino_acids     Codons  Existing_variation      DISTANCE
STRAND  FLAGS   VARIANT_CLASS   SYMBOL_SOURCE   HGNC_ID CANONICAL
TSL     APPRIS  CCDS    ENSP    SWISSPROT       TREMBL  UNIPARC
SOURCE  GENE_PHENO      SIFT    PolyPhen        DOMAINS
HGVS_OFFSET     HGVSg   AF      AFR_AF  AMR_AF  EAS_AF  EUR_AF
SAS_AF  AA_AF   EA_AF   gnomAD_AF       gnomAD_AFR_AF
gnomAD_AMR_AF   gnomAD_ASJ_AF   gnomAD_EAS_AF   gnomAD_FIN_AF
gnomAD_NFE_AF   gnomAD_OTH_AF   gnomAD_SAS_AF   MAX_AF
MAX_AF_POPS     CLIN_SIG        SOMATIC PHENO   PUBMED  MOTIF_NAME
MOTIF_POS       HIGH_INF_POS    MOTIF_SCORE_CHANGE      CADD_PHRED
CADD_RAW        GeneHancer_type_and_Genes       gnomADg gnomADg_AC
gnomADg_AN      gnomADg_AF      gnomADg_DP      gnomADg_AC_nfe_seu
gnomADg_AN_nfe_seu      gnomADg_AF_nfe_seu
gnomADg_nhomalt_nfe_seu gnomADg_AC_raw  gnomADg_AN_raw
gnomADg_AF_raw  gnomADg_nhomalt_raw     gnomADg_AC_afr
gnomADg_AN_afr  gnomADg_AF_afr  gnomADg_nhomalt_afr
gnomADg_AC_nfe_onf      gnomADg_AN_nfe_onf      gnomADg_AF_nfe_onf
gnomADg_nhomalt_nfe_onf gnomADg_AC_amr  gnomADg_AN_amr
gnomADg_AF_amr  gnomADg_nhomalt_amr     gnomADg_AC_eas
gnomADg_AN_eas  gnomADg_AF_eas  gnomADg_nhomalt_eas
gnomADg_nhomalt gnomADg_AC_nfe_nwe      gnomADg_AN_nfe_nwe
gnomADg_AF_nfe_nwe      gnomADg_nhomalt_nfe_nwe gnomADg_AC_nfe_est
gnomADg_AN_nfe_est      gnomADg_AF_nfe_est
gnomADg_nhomalt_nfe_est gnomADg_AC_nfe  gnomADg_AN_nfe
gnomADg_AF_nfe  gnomADg_nhomalt_nfe     gnomADg_AC_fin
gnomADg_AN_fin  gnomADg_AF_fin  gnomADg_nhomalt_fin
gnomADg_AC_asj  gnomADg_AN_asj  gnomADg_AF_asj
gnomADg_nhomalt_asj     gnomADg_AC_oth  gnomADg_AN_oth
gnomADg_AF_oth  gnomADg_nhomalt_oth     gnomADg_popmax
gnomADg_AC_popmax       gnomADg_AN_popmax       gnomADg_AF_popmax
gnomADg_nhomalt_popmax  gnomADg_cov     gwascatalog
gwascatalog_GWAScat_DISEASE_or_TRAIT
gwascatalog_GWAScat_INITIAL_SAMPLE_SIZE
gwascatalog_GWAScat_REPLICATION_SAMPLE_SIZE
gwascatalog_GWAScat_STRONGEST_SNP_and_RISK_ALLELE
gwascatalog_GWAScat_PVALUE
gwascatalog_GWAScat_STUDY_ACCESSION     clinvar clinvar_CLNDN
clinvar_CLNSIG  clinvar_CLNDISDB        miRBase pharmgkb_drug
pharmgkb_drug_PGKB_Annotation_ID        pharmgkb_drug_PGKB_Gene
pharmgkb_drug_PGKB_Chemical     pharmgkb_drug_PGKB_PMID
```

```
pharmgkb_drug_PGKB_Phenotype_Category
pharmgkb_drug_PGKB_Sentence chr21       33241945        rs2229207
T       C       27      152     2003    0.179   2       C
missense_variant        MODERATE        IFNAR2  ENSG00000159110
Transcript      ENST00000342136.8       protein_coding  2/9     .
ENST00000342136.8:c.23T>C       ENSP00000343957.4:p.Phe8Ser
349/2899        23/1548 8/515   F/S     tTc/tCc rs2229207&CM066573
.       1       .       SNV     HGNC    HGNC:5433       YES     1
P4      CCDS13621.1     ENSP00000343957 P48551  .
UPI000012D69B   .       1       tolerated       benign
hmmpanther:PTHR20859&hmmpanther:PTHR20859:SF53&Transmembrane_helic
es:TMhelix      .       chr21:g.33241945T>C     0.1186  0.0809
0.147   0.1706  0.0736  0.1421  0.07558 0.07791 0.1033  0.07742
0.154   0.07741 0.1757  0.08546 0.08082 0.09088 0.1247  0.1757
gnomAD_EAS      risk_factor     .       1&1
16757563&19434718&23009887&28497593&18588853&27186094   .       .
.       .       2.171   0.043682        .       rs2229207
3026    31374   0.0964493       657463  14      106     0.132075
0       3039    31416   0.0967341       168     736     8706
0.0845394       27      198     2136    0.0926966       10
122     848     0.143868        13      317     1552    0.204253
29      164     637     8592    0.0741387       31      607
4584    0.132417        35      1456    15418   0.0944351       76
277     3474    0.0797352       12      23      290     0.0793103
1       95      1086    0.087477        6       eas     317
1552    0.204253        29      32.63   .       .       .       .
.       .       .       chr21:33241945-33241945 _susceptibility_to
risk_factor     OMIM:610424     .       .       .       .       .
.       .       .
```

16

Count variants

Count variants by using "summary_cleaner.R" tool.

Note

Summary_cleaner.R is a tool for counting variants from different types and subgroups.

Dependencies:

- summary_cleaner.R

Final Output:

Expected result

A '.tsv' the description of the counted variants by each subgroup of variants.

Example line(s):

```
row_nam common_freq     commonAMR_lowEUR        low_freq
lowEAS_lowEUR   No_filter       rare_freq       selection_signals
miRNA   0       0       0       0       0       0       0 PGKB  0
0       0       0       0       0       0 GWAScatalog   3       0
0       0       3       0       0 OMIM  6       2       1       0
9       2       0 coding_region 7       0       1       1       19
11      0 clinvar       16      2       2       0       21      3
0 utr   90      5       26      7       161     45      0
dbSNPnovel      35      2       114     3       977     828     0
GeneHancer      599     44      187     47      1033    247     0
dbSNPknown      8995    746     2706    567     14586   2885    0
general (all indels)    9030    748     2820    570     15563
3713    0
```

17 **QC VEP consequence plot**

Plot consequences of each category of variants.

Dependencies:

- plotter.R