**Benign-Ex User Manual**

**Version 1.0**

**May 2021**

----------------------------------------------------------------------------------------------------------------------------------------------------------------

**INTRODUCTION:**

Benign-Ex is designed to classify regions of the genome as “benign” from the perspective of copy number status. Benign-Ex relies on two assumptions. First, that CNVs commonly classified clinically as benign or likely benign occur in regions of the genome that are benign from a copy number status perspective. Second, that regions of the genome that exhibit a high prevalence of copy number variation in the general population are benign from a copy number perspective.

Benign-Ex can be run in two ways:

1. Based on a single set of parameters provided by the user, Benign-Ex will identify normal copy number variable (“benign”) regions of the human genome from frequency-based and/or classification-based datasets.
2. Based on multiple sets of parameters provided by the user, Benign-Ex will identify the optimal set of parameters for identifying normal copy number variable (“benign”) regions of the human genome from frequency-based and/or classification-based datasets. The optimal set of parameters is determined by comparing the amount of overlap between the Benign-Ex identified “benign” regions with a set or sets of genomic regions known (or highly likely) to be pathogenic from a copy number perspective.

Datasets from the Database of Genomic Variants (DGV; "frequency") and the Clinical Genome Resource (ClinGen; "classification") are included in the distribution for the hg19 and GRCh38 assemblies.

**Table of Contents**

[**REQUIREMENTS:** 2](#_Toc72834343)

[**INSTALLATION:** 2](#_Toc72834344)

[**MINIMAL SETUP INSTRUCTIONS:** 2](#_Toc72834345)

[**COMPLETE SETUP INSTRUCTIONS:** 3](#_Toc72834346)

[File Structure & Location for SETUP Files 3](#_Toc72834347)

[Add and Format Additional Datasets for the Identification of “benign” Regions 3](#_Toc72834348)

[Modify the Set of Parameters Used by Benign-Ex to Call “benign” Regions 5](#_Toc72834349)

[Modify the Set of Known Pathogenic Regions for Evaluating Individual Benign-Ex “benign” Regions 6](#_Toc72834350)

[Set-up the AUTOMATION file 7](#_Toc72834351)

[**RUNNING BENIGNEX:** 8](#_Toc72834352)

[EXAMPLE 1: Heatmap Only Run 8](#_Toc72834353)

[EXAMPLE 2: Full Benign-Ex Run 10](#_Toc72834354)

[**COMMON ERRORS:** 11](#_Toc72834355)

[Use of Python3 instead of Python2 (2.7.X): 11](#_Toc72834356)

[Use of R version which is not installed: 11](#_Toc72834357)

# **REQUIREMENTS:**

Benign-Ex requires Python (2.7.X) and R (>=3.6.0) in a Linux operating system to run. Python3 is currently not supported. A list of the required Python modules and R libraries is listed below. Benign-Ex will attempt to install the required Python modules, but the R libraries must be manually installed. If the required Python modules do not install on their own, they will need to be installed manually as well.

Required Python Modules:

* progress
* networkx; version 1.8.1
* intervaltree

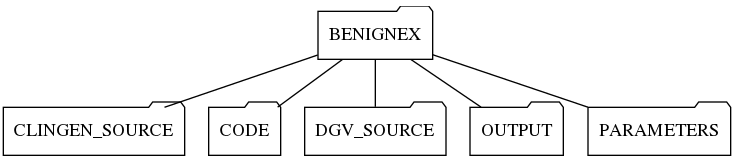
Required R Libraries:

* bestNormalize
* BoutrosLab.plotting.general
* tidyverse

# **INSTALLATION:**

After you have downloaded the BENIGNEX\_X.X.tar.gz file, move zipped file to the directory in which you want to run Benign-Ex. Then, decompress the file.

|  |
| --- |
| tar -xzvf BENINGEX\_1.0.tar.gz --one-top-level=BENIGNEX --strip-components 1 |



# **MINIMAL SETUP INSTRUCTIONS:**

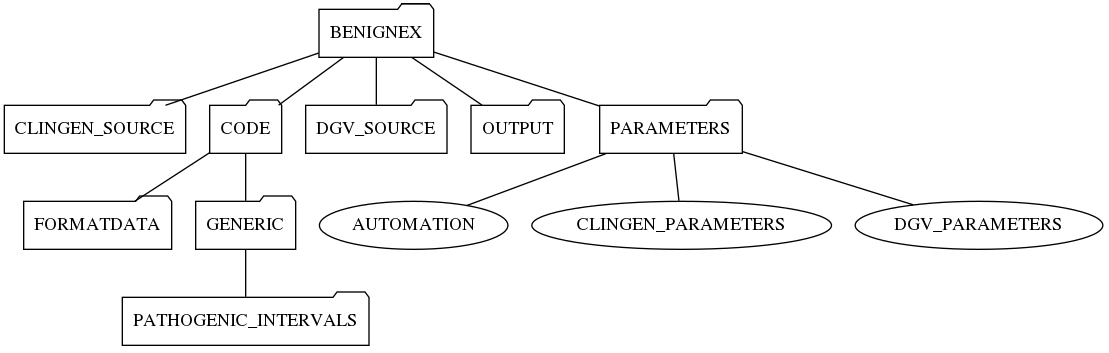
Open the ‘AUTOMATION’ file in the PARAMETERS folder (~/BENIGNEX/PARAMETERS/AUTOMATION) and modify lines (4) and (5) to call to the correct version of Python (2.7.X) and R (>=3.6.0). Examples include: [ python, python2, py2 ] [ R, R-3.6.0, R\_3.6.0 ]

**AUTOMATION File Contents:**

|  |
| --- |
| EXAMPLE  CLINGEN\_HG19\_2021,DGV\_HG19\_2020  BENIGN-EX=N,HEATMAP=Y  <python>  <R>  DEFAULT,Primary\_CNVs\_hg19.BED |

# **COMPLETE SETUP INSTRUCTIONS:**

## File Structure & Location for SETUP Files



## Add and Format Additional Datasets for the Identification of “benign” Regions

The FORMATDGV.py and BREAKINTOCHR.py scripts are used to generate individual chromosome files for use by Benign-Ex. The FORMATDGV.py script is specifically formatted to process the DGV database file downloaded directly, while the BREAKINTOCHR.py script will work for all other frequency-based and classification-based datasets. These scripts are natively configured to remove inversion events, 0-bp and 1-bp events, and non-canonical chromosomes and scaffolds (e.g. chrN, chrUn\_gl000228, chr11\_gl000202\_random) from the database during processing. The resultant individual chromosome files will be output to the specified folder. **NOTE: The hg19 and hg38 datasets for the 2020 DGV release are included within the Benign-Ex distribution packaged as tar.gz files.**

For data downloaded directly from DGV:

1. Add DGV file to the FORMATDATA folder ( ~/PATH/BENIGNEX/CODE/FORMATDATA )
2. Run the FORMATDGV.py script

|  |
| --- |
| python FORMATDGV.py <INPUT FILE PATH> <OUTPUT DIRECTORY PATH> |

**Example:**

|  |
| --- |
| python BENIGNEX/CODE/FORMATDATA/FORMATDGV.py BENIGNEX/CODE/FORMATDATA/ GRCh37\_hg19\_variants\_2020-02-25.txt BENIGNEX/DGV\_SOURCE/HG19\_2020 |

For data downloaded from all other data sources:

1. Format the user-supplied dataset in accordance with the frequency-based or classification-based format below.

**FREQUENCY-BASED**: Datasets should be tab-delimited and all spaces should be converted to underscore "\_". Users are required to maintain the 11-column structure, but columns 5 and 6 can be left blank or modified to record any relevant information about the variant. Benign-Ex does not natively apply filters based on these columns. Columns 7 and 8 can also be left blank if you choose not to filter on methodology and/or year.

1. The first column contains the variant ID
2. The second column contains the chromosome on which the variant is located, formatted without "chr" such as 1,2,3
3. The third column contains the start coordinate for the variant
4. The fourth column contains the end coordinate for the variant
5. The fifth column contains descriptive information for that variant (e.g. variant type)
6. The sixth column contains descriptive information for that variant (e.g. variant subtype)
7. The seventh column contains the reference for the variant. If you are planning on excluding variants based on publication year, make sure the reference ends with the year "\_####" (e.g. "Reference\_2009")
8. The eight column contains a description of the methodology by which the variant was identified.
9. The ninth column contains the sample size of the study in which the variant was identified.
10. The tenth column contains the number of times the variant was identified as a "gain" within the study
11. The eleventh column contains the number of times the variant was identified as a "loss" within the study

**Example File Preview:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Chrom | Start | End | Type | Subtype | Reference | Method | N | Gain | Loss |
| nsv482937 | 1 | 1 | 2300000 | CNV | loss | Iafrate\_et\_al\_2004 | BAC\_aCGH,FISH | 39 | 0 | 1 |
| dgv1n82 | 1 | 10001 | 22118 | CNV | duplication | Sudmant\_et\_al\_2013 | Oligo\_aCGH,Sequencing | 97 | 10 | 0 |
| nsv7879 | 1 | 10001 | 127330 | CNV | gain+loss | Perry\_et\_al\_2008 | Oligo\_aCGH | 31 | 25 | 1 |
| esv3648995 | 1 | 10151 | 10389 | OTHER | complex | Besenbacher\_et\_al\_2015 | Sequencing | 20 | 0 | 0 |
| nsv958854 | 1 | 10191 | 10281 | CNV | deletion | Dogan\_et\_al\_2014 | Sequencing | 1 | 0 | 1 |
| nsv428112 | 1 | 10377 | 177417 | CNV | gain | Perry\_et\_al\_2008b | BAC\_aCGH,FISH,PCR | 62 | 1 | 0 |
| esv2758911 | 1 | 10377 | 1018704 | CNV | gain+loss | Redon\_et\_al\_2006 | BAC\_aCGH,SNP\_array | 270 | 17 | 169 |
| esv27265 | 1 | 10499 | 177368 | CNV | gain+loss | Conrad\_et\_al\_2009 | Oligo\_aCGH | 40 | 32 | 6 |
| nsv3319479 | 1 | 10501 | 78200 | CNV | duplication | Audano\_et\_al\_2019 | Sequencing | 14 | 4 | 0 |
| nsv3320784 | 1 | 10726 | 10726 | CNV | insertion | Audano\_et\_al\_2019 | Sequencing | 14 | 1 | 0 |

**CLASSIFICATION-BASED:** Datasets should be tab-delimited and all spaces should be converted to underscore "\_".

1. The first column contains the chromosome on which the variant is located
2. The second column contains the start coordinate for the variant
3. The third column contains the end coordinate for the variant
4. The fourth column contains the variant ID
5. The fifth column contains the variant classification (accepted values: "Benign", "Likely Benign", or "Uncertain Significance")
6. The sixth column contains the variant direction (accepted values: "GAIN", "LOSS")

**Example File Preview:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Chrom | Start | End | VariantID | Classification | Direction |
| chr1 | 6942338 | 8130466 | nssv13640694 | Benign | gain |
| chr1 | 7075128 | 7581003 | nssv13639619 | Likely Benign | gain |
| chr1 | 15492847 | 16400774 | nssv13655274 | Uncertain Significance | gain |
| chr1 | 43371442 | 43381512 | nssv1603001 | Likely Benign | gain |
| chr1 | 85066622 | 85319147 | nssv576372 | Uncertain Significance | loss |
| chr1 | 88930084 | 89181354 | nssv582885 | Uncertain Significance | loss |
| chr1 | 104027758 | 104551555 | nssv1415176 | Uncertain Significance | loss |
| chr1 | 141867688 | 142051761 | nssv13651472 | Benign | gain |
| chr1 | 162504648 | 162787928 | nssv13644415 | Benign | loss |
| chr1 | 189430101 | 189562274 | nssv581922 | Benign | loss |

For data downloaded from all other data sources (continued):

1. Run the FORMATDATA.py script to create the required individual chromosome files.

|  |
| --- |
| BREAKINTOCHR.py <INPUT FILE> <OUTPUT DIRECTORY> <HEADER; Y or N> <DATABASE; DGV or CLINGEN> |

**Example:**

|  |
| --- |
| python BENIGNEX/CODE/FORMATDATA/BREAKINTOCHR.py DGV\_Formatted\_hg19\_2020-02-25.txt BENIGNEX/DGV\_SOURCE/HG19\_2020 Y DGV  python BENIGNEX/CODE/FORMATDATA/BREAKINTOCHR.py ClinGen\_hg19\_2021-02-15.txt BENIGNEX/CLINGEN\_SOURCE/HG19\_2021 Y CLINGEN |

## Modify the Set of Parameters Used by Benign-Ex to Call “benign” Regions

**DGV\_PARAMETERS**: This file within the PARAMETERS folder contains the default user-supplied parameter sets for calling regions of the genome benign using frequency-based data. You may choose to run the predefined default parameters or add your own using the following format:

EVENTS=#,#,(GAIN/LOSS),#,(Y/N),(NA/20XX)

1. The first option takes an integer such as 1,2,3....
2. The second option takes a integer such as 10,20,100…
3. The third option takes 'GAIN' or 'LOSS' value.
4. The fourth option takes a float such as 0.1,0.2... or number value such as 1,2,3..
5. The fifth option takes 'Y' or 'N' value.
6. The sixth option takes a value in year format such as 2009, 2010… or 'NA'

**Example Line of File:**

|  |
| --- |
| EVENTS=3,30,GAIN,0.1,Y,2009 |

Benign-Ex will define a benign region for a chromosome in the human genome if the it meets the following conditions:

1. Number of unique variants covering this region should at least be 3 or more.
2. Sample size for the study in which the variant was found should be at least be 30 or more.
3. Restrict to only look at variants which are GAINs (as opposed to a LOSS)
4. Gain frequency for each variant within the respective study should at least be 0.1 or more.
5. 'Y' restricts the inclusion of variants to only those which come from studies which use one of the following techniques: 'BAC aCGH','SNP array', 'Oligo aCGH', and/or 'Sequencing'.
6. Year filter allows you to restrict studies to those published in or after 2009. If the value 'NA' is provided, Benign-Ex will use all studies regardless of when the studies were published.

**Example File:**

|  |
| --- |
| EVENTS=1,30,GAIN,1,N,NA  EVENTS=1,30,GAIN,2,N,NA  EVENTS=1,30,GAIN,3,N,NA  EVENTS=1,30,GAIN,4,N,NA  EVENTS=1,30,GAIN,5,N,NA  EVENTS=1,30,LOSS,1,N,NA  EVENTS=1,30,LOSS,2,N,NA  EVENTS=1,30,LOSS,3,N,NA  EVENTS=1,30,LOSS,4,N,NA  EVENTS=1,30,LOSS,5,N,NA |

**CLINGEN\_PARAMETERS:** This file within the PARAMETERS folder contains the default user-supplied parameter sets for calling regions of the genome benign based upon data within CLINGEN. You may choose to run predefined default parameters or add your own using the following format:

EVENTS=#,CNVTYPES=(B : B/USLB : B/USLB/VUS):

1. The first option takes an integer value such as 1,2,3....
2. The second option takes any combination of 'B', 'USLB', and 'VUS'. These correspond to classifications of benign (B), likely benign (USLB), or variant of unknown significance (VUS). A forward-slash '/' can be used to combine classifications.

**Example Line of File:**

|  |
| --- |
| EVENTS=3,CNVTYPES=B/USLB |

Benign-Ex will define a benign region for a chromosome in the human genome if the it meets the following conditions:

1. Number of CNVs within CLINGEN covering this benign region should at least be 3 or more.
2. Restricts to only look at variants which are classified as either benign or likely benign.

**Example File:**

|  |
| --- |
| EVENTS=1,CNVTYPES=B  EVENTS=2,CNVTYPES=B  EVENTS=3,CNVTYPES=B  EVENTS=1,CNVTYPES=B/USLB  EVENTS=2,CNVTYPES=B/USLB  EVENTS=3,CNVTYPES=B/USLB  EVENTS=1,CNVTYPES=B/USLB/VUS  EVENTS=2,CNVTYPES=B/USLB/VUS  EVENTS=3,CNVTYPES=B/USLB/VUS |

## Modify the Set of Known Pathogenic Regions for Evaluating Individual Benign-Ex “benign” Regions

The PATHOGENIC\_INTERVALS folder contains a set of "likely pathogenic regions" consisting of CNV intervals and/or genes associated with haploinsufficiency or human disease. As the goal of Benign-Ex is to call as much of the genome "benign" as possible, while limiting the amount of overlap between Benign-Ex called "benign" regions and pathogenic regions of the genome, these lists are used to assess the performance of Benign-Ex for each parameter set. The ratio between the amount of the genome called "benign" and the amount of overlap between the “benign” called regions and the “likely pathogenic regions” was used to assess Benign-Ex’s performance. This ratio was calculated using a variation of the overlap coefficient.

1. Navigate to the correct assembly within the PATHOGENIC\_INTERVALS folder (~/BENIGNEX/CODE/GENERIC/ PATHOGENIC\_INTERVALS/HGXX)
2. To remove "likely pathogenic region" lists, simply remove the list from the corresponding folder within the PATHOGENIC\_INTERVALS folder.
3. To add "likely pathogenic region" lists, add the list (in BED format) to the corresponding folder within the PATHOGENIC\_INTERVALS folder.

## Set-up the AUTOMATION file

The AUTOMATION file within the PARAMETERS folder contains all the arguments needed to run Benign-Ex.

**AUTOMATION File Contents:**

|  |
| --- |
| <FOLDERNAME>  CLINGEN/DGV\_<ASSEMBLY>\_<YEAR>  BENIGN-EX=Y/N,SR=Y/N,HEATMAP=Y/N  **<python>**  **<R>**  DEFAULT |

1. The first line takes the FOLDER\_NAME.
2. The second line takes the DGV and CLINGEN versions formatted as such :: (DGV/CLINGEN)\_(ASSEMBLY)\_(YEAR)
3. The third line determines what portions of Benign-Ex you want to run. (e.g. generate benign intervals and/or generate new heatmaps).

* If BENIGN-EX=Y, the program will identify benign regions using the parameters within the DGV\_PARAMETERS and/or CLINGEN\_PARAMETERS files
* If BENIGN-EX=N, the program will skip this step \*\*Use only if you have already generated benign intervals from a previous run\*\*
* If SR=Y (and DGV data is being processed), the program will run the suspicious regions code. This is a secondary step currently under testing which is used to account for segmental duplications where duplications or deletions may be favored by the calling software, or the presence of a CNV within the control genome.
* If SR=N, the program will skip the suspicious regions step.
* If HEATMAP=Y, the program will identify the optimal parameter for a given set of benign and pathogenic intervals.
* If HEATMAP=N, the program will NOT identify the optimal parameter for a given set of benign and pathogenic intervals.

1. The fourth line is used to call python. (e.g. 'python' or a specific version such as 'python2' if multiple versions are installed)
2. The fifth line is used to call R. (e.g. 'R' or a specific version such as 'R-3.6.0' if multiple versions are installed)
3. The sixth line is used to determine how many sets of FINAL bed files to create. Each set will contain a final gain and loss BED file.

* If “DEFAULT”, Benign-Ex will create the Final BED files using the optimal parameter across all supplied pathogenic lists.
* If <PATHOGENIC LIST>, Benign-Ex will create the Final BED files with the optimal parameter from the specified pathogenic list.
* Multiple pathogenic lists can be supplied by using commas to delimit different lists.

**Example:**

|  |
| --- |
| RUN1  DGV\_HG19\_2020,CLINGEN\_HG19\_2021  BENIGN-EX=Y,SR=N,HEATMAP=Y  python2  R-3.6.3  DEFAULT,Primary\_CNVs\_hg19.BED |

1. Benign-Ex will output all files in the “RUN1” folder under the OUTPUT folder
2. Benign-Ex will generate “benign” intervals for the hg19 assembly for both DGV and CLINGEN from the HG19\_2020 and HG19\_2021 source files ( found in DGV\_SOURCE and CLINGEN\_SOURCE )
3. Benign-Ex will both generate the “benign” intervals and determine the optimal parameter setting for the given set of parameters supplied by DGV\_PARAMETERS and CLINGEN\_PARAMETERS. Suspicious region processing will *not* be performed.
4. Benign-Ex will call a compatible version of python (2.7.X) via the “python2” command
5. Benign-Ex will call a compatible version of R (>=3.6.0) via the “R-3.6.3” command
6. Benign-Ex will create two sets of Final BED files using the optimal parameter across all lists (DEFAULT), and the

optimal parameter from the Primary\_CNVs\_hg19.BED list

# **RUNNING BENIGNEX:**

Enter the BENIGNEX directory, and use the following command to run Benign-Ex.

|  |
| --- |
| cd ~/PATH/BENIGNEX  python CODE/INTERFACE\_BENIGNEX.py PARAMETERS/AUTOMATION |

## EXAMPLE 1: Heatmap Only Run

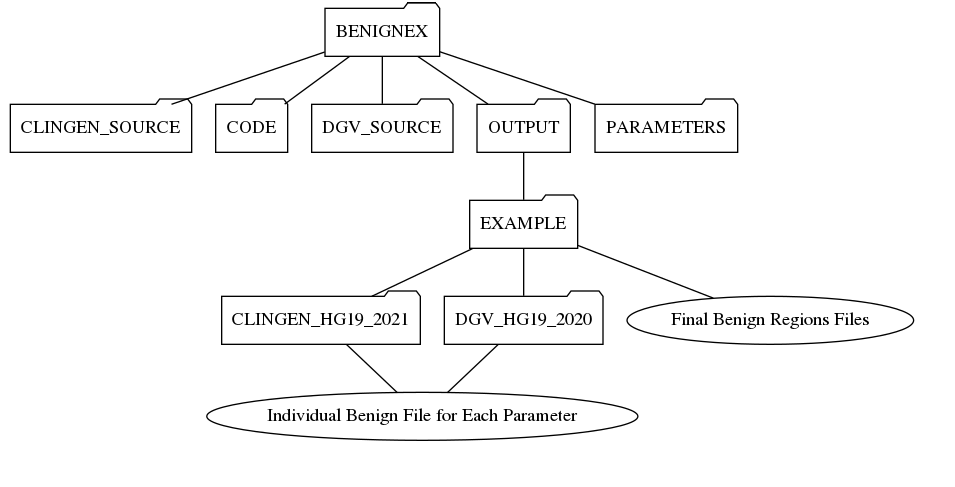
AUTOMATION File Contents:

|  |
| --- |
| EXAMPLE  CLINGEN\_HG19\_2021,DGV\_HG19\_2020  BENIGN-EX=N,SR=N,HEATMAP=Y  python  R  DEFAULT |

If all the required Python modules and R libraries are installed Benign-Ex will run according to the AUTOMATION file and you will get the following output to the console. Note that this run only contains chromosomes 1, 17, and 22 to decrease the run time of the program. The overall runtime is dependent on the total number of parameters submitted.

|  |
| --- |
| RUNNING A DEPENDENCIES CHECK ...  YOUR PYTHON VERSION INSTALLED IS: [ Python Version ]  YOUR R VERSION INSTALLED IS: [ R Version ]  ALL THE DEPENDENCIES ARE UP TO DATE...  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  PROCESSING CLINGEN DATABASE...  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  COMPUTING OVERLAP COEFFICIENT FOR...CLINGEN  THIS MIGHT TAKE TIME, PLEASE BE PATIENT...  Processing |################################| 100%  IDENTIFYING THE OPTIMAL PARAMETER SET FOR ...CLINGEN  Processing |################################| 100%  YOUR OUTPUT IS SUBMITTED HERE: ~/PATH/BENIGNEX /OUTPUT/EXAMPLE/CLINGEN\_HG19\_2021  Processing |################################| 100%  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  PROCESSING DGV DATABASE...  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  COMPUTING OVERLAP COEFFICIENT FOR...DGV  THIS MIGHT TAKE TIME, PLEASE BE PATIENT...  Processing |################################| 100%  IDENTIFYING THE OPTIMAL PARAMETER SET FOR ...DGV  Processing |################################| 100%  YOUR OUTPUT IS SUBMITTED HERE: ~/PATH/BENIGNEX/OUTPUT/EXAMPLE/DGV\_HG19\_2020  Processing |################################| 100%  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  GENERATING FINAL BENIGN-EX BENIGN REGIONS...  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  YOUR OUTPUT IS SUBMITTED HERE: ~/PATH/BENIGNEX /OUTPUT/EXAMPLE |

Benign-Ex Output Structure:



Benign-Ex will output all of the intermediate files and individual “benign” region files for each parameter setting in the DGV\_PARAMETERS and CLINGEN\_PARAMETERS file under the corresponding database+assembly folder in the EXAMPLE folder (~/BENIGNEX/OUTPUT/EXAMPLE/DGV\_HG19\_2020 and ~/BENIGEX/OUTPUT/EXAMPLE/CLINGEN\_HG19\_2021). The final GAIN and LOSS bed files for the optimal parameter set can be found in the EXAMPLE folder. This will consist of the merged “benign” intervals from the optimal CLINGEN and DGV benign interval sets.

## EXAMPLE 2: Full Benign-Ex Run

**AUTOMATION File Contents:**

|  |
| --- |
| FULLRUN  CLINGEN\_HG19\_2021,DGV\_HG19\_2020  BENIGN-EX=Y,SR=N,HEATMAP=Y  python  R  DEFAULT |

If all the required Python modules and R libraries are installed Benign-Ex will run according to the AUTOMATION file and you will get the following output to the console:

|  |
| --- |
| RUNNING A DEPENDENCIES CHECK ...  YOUR PYTHON VERSION INSTALLED IS: [ Python Version ]  YOUR R VERSION INSTALLED IS: [ R Version ]  ALL THE DEPENDENCIES ARE UP TO DATE...  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  PROCESSING CLINGEN DATABASE...  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Processing |################################| 100% CHR1 GAIN  Processing |################################| 100% CHR1 LOSS  Processing |################################| 100% CHR2 GAIN  Processing |################################| 100% CHR2 LOSS  ...  Processing |################################| 100% CHRX GAIN  Processing |################################| 100% CHRX LOSS  Processing |################################| 100% CHRY GAIN  Processing |################################| 100% CHRY LOSS  COMPUTING OVERLAP COEFFICIENT FOR...CLINGEN  THIS MIGHT TAKE TIME, PLEASE BE PATIENT...  Processing |################################| 100%  IDENTIFYING THE OPTIMAL PARAMETER SET FOR ...CLINGEN  Processing |################################| 100%  YOUR OUTPUT IS SUBMITTED HERE: ~/PATH/BENIGNEX /OUTPUT/FULLRUN/CLINGEN\_HG19\_2021  Processing |################################| 100%  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  PROCESSING DGV DATABASE...  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Processing |################################| 100% CHR1  Processing |################################| 100% CHR2  ...  Processing |################################| 100% CHRX  Processing |################################| 100% CHRY  COMPUTING OVERLAP COEFFICIENT FOR...DGV  THIS MIGHT TAKE TIME, PLEASE BE PATIENT...  Processing |################################| 100%  IDENTIFYING THE OPTIMAL PARAMETER SET FOR ...DGV  Processing |################################| 100%  YOUR OUTPUT IS SUBMITTED HERE: ~/PATH/BENIGNEX/OUTPUT/FULLRUN/DGV\_HG19\_2020  Processing |################################| 100%  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  GENERATING FINAL BENIGN-EX BENIGN REGIONS...  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  YOUR OUTPUT IS SUBMITTED HERE: ~/PATH/BENIGNEX/OUTPUT/FULLRUN |

# **COMMON ERRORS:**

## Use of Python3 instead of Python2 (2.7.X):

|  |
| --- |
| File "CODE/INTERFACE\_BENIGNEX3.py", line 6  print "\nERROR: AUTOMATION FILE NOT PROVIDED. PLEASE REVIEW README FILE."  ^  SyntaxError: Missing parentheses in call to 'print'. Did you mean print("\nERROR: AUTOMATION FILE NOT PROVIDED. PLEASE REVIEW README FILE.")? |

## Use of R version which is not installed:

|  |
| --- |
| sh: 1: R-3.6.0: not found |