**DOCUMENTATION**

Database of Genomic Variants (**hg38)**

* Modified the original file to remove *observedlosses, cohortdescription, genes, samples* since there are no values in these columns. Named the new file **variants\_modified.txt.**
* Removed *varianttype, variantsubtype, reference, method* columns using **cut -d$'\t' -f5-7,9 variations\_modified.txt > check\_for\_unique\_values.txt** to check for unique values for each column
* Removed *chr, start, end* columns using **cut -d$'\t' -f2-4 variations\_modified.txt > variations\_start\_stop.txt** to run the non-structural variation code
* Used **cut -d$'\t' -f1 unique.txt | sort | uniq >> unique\_results\_for\_variantsubtype\_etc. txt** for column one (f1) to find the unique values (did so for the rest). Manually picked out the unique from the **methods** since it was so little, creating a function would be longer than simply choosing the unique ones.
* To find all the unique variant subtypes where the variant type is other, use the following lines:

import pandas as pd

df = pd.read\_csv('check\_for\_unique\_values\_variantsubtypes\_etc.txt', sep='\t')

df\_o = df[df['varianttype'] == 'OTHER']

unq = df\_o = pd.unique(df\_o['variantsubtype'])

* Ran **generate\_nvr\_and\_vr.ipynb** to create **variable\_regions.txt** and **non\_variable\_regions.txt**
* Got data from decipher database (hg 19) and converted it to hf38 via NCBI remapping tool. Re-ran above for to generate new VR and NVR regions named **variable\_regions\_updated.txt and non\_variable\_regions\_updated.txt**

UCSC Genenome Browser (**hg38, refSeq)**

* Removed all columns except *chr, start, end* using **cut -d ' ' -f1-3 genes.txt > genes\_start\_stop.txt (N.B.** removed *‘chr’* from the first column to make it easier to use preexisting functions, via Sublime Text option to replace e.g. chr1 -> 1)
* Ran remove\_duplicate\_gene\_start\_stop.ipynb on **genes\_start\_stop.txt** to concatenate refSeq IDs that have the same start and stop. The new file is **genes\_start\_stop\_no\_dups.txt**

Combining Regions of no structural variation and genes

* Ran **combining\_regions\_and\_genes.ipynb** to produce **non\_variable\_regions\_and\_genes.txt** and **variable\_regions\_and\_genes.txt**

**Fig. 1 - Shows the 9 different possibilities for overlapping and their code for each.**



**Analyzing Genes (See:** [gene\_analysis.ipynb](http://localhost:8888/notebooks/gene_analysis.ipynb)**)**

* Added all genes to IPA using **awk '{print $4}' genes\_start\_stop.txt > genes\_start\_stop\_one\_column.txt** on **genes\_start\_stop.txt** to make **genes\_start\_stop\_one\_column.txt**

GC Content (**hg38)**

* Got GC content from NCIB for each chromosome in a fasta file.
* Removed header from all NCIB file using sed '1d' d.txt > tmp.txt
* Ran GC\_content\_Analysis.ipynb

Regulators (**hg38)**

* Calculated the size of NVR (626685689) and VR (2461584143) in basepairs:

df = pd.read\_csv('../no\_variation\_results.txt', sep='\t', names=['chr', 'start', 'stop'], header=None)

total = df[‘stop’] - df[‘start’] + 1

print sum(total)

* **For all regulators:**
  + Used Sublime to remove *chr* from the chromosome to just have the number or letter (i.e chr1 -> 1 or chrX -> X) but did not change the header line
  + Ran **analyze regulators.ipynb**  to generate the results