AACR GENIE Variant Summary Report

Overview

This document contains analysis summarizing variant, sample, and patient information from the AACR Project GENIE data set. This is among the largest public cancer data sets with data from over 160,000 patients.

Data have been are aggregated from multiple institutions and multiple NGS panels were used to collect mutation data.

Loading data

```
bDir <- "../../data/processed/balderResultsDb"
figDir <- "../../output/actionability_db_curration_20231220"
mydb <- DBI::dbConnect(RSQLite::SQLite(), paste0(bDir,"/actionable-biomarker-db.sqlite"))

genie <- RSQLite::dbGetQuery(mydb, 'SELECT * FROM GeniePatientVarients')
sample.genie <- RSQLite::dbGetQuery(mydb, 'SELECT * FROM GenieClinicalSampleData')
patient.genie <- RSQLite::dbGetQuery(mydb, 'SELECT * FROM GeniePatientData')

genie.full <- genie %>%
  dplyr::left_join(sample.genie,by=c("Tumor_Sample_Barcode"="SAMPLE_ID")) %>%
  dplyr::left_join(patient.genie,by="PATIENT_ID")
dim(genie.full)
```

```
[1] 1712997      81
```

```
### to do: print out column names. What is there in terms of variant info and annotations
```

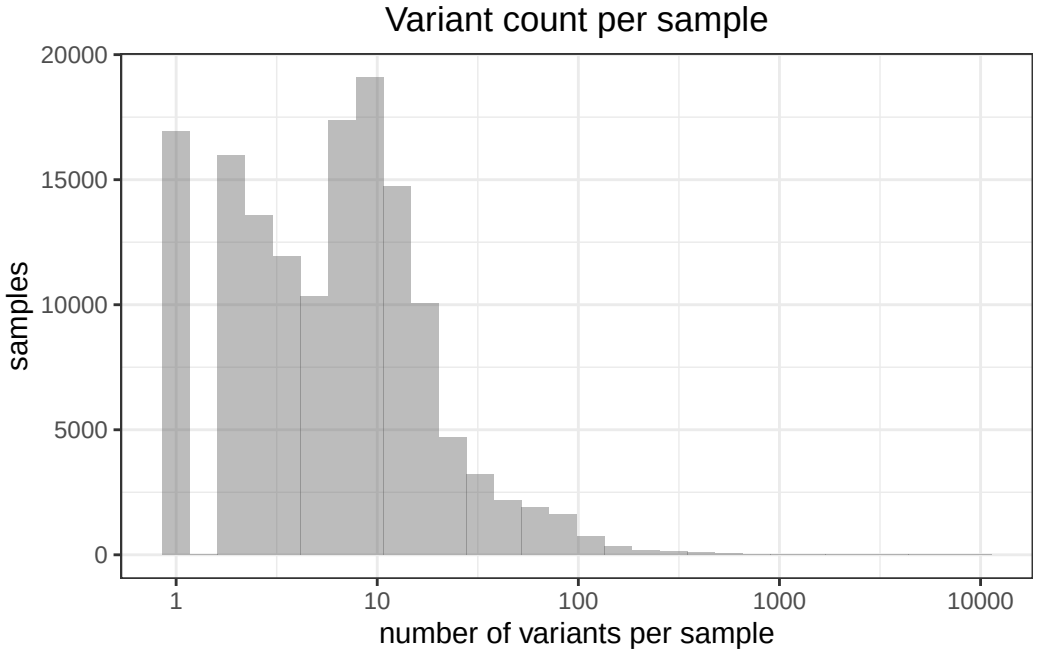
Results summary

what proportion of all patients are in the variant table?

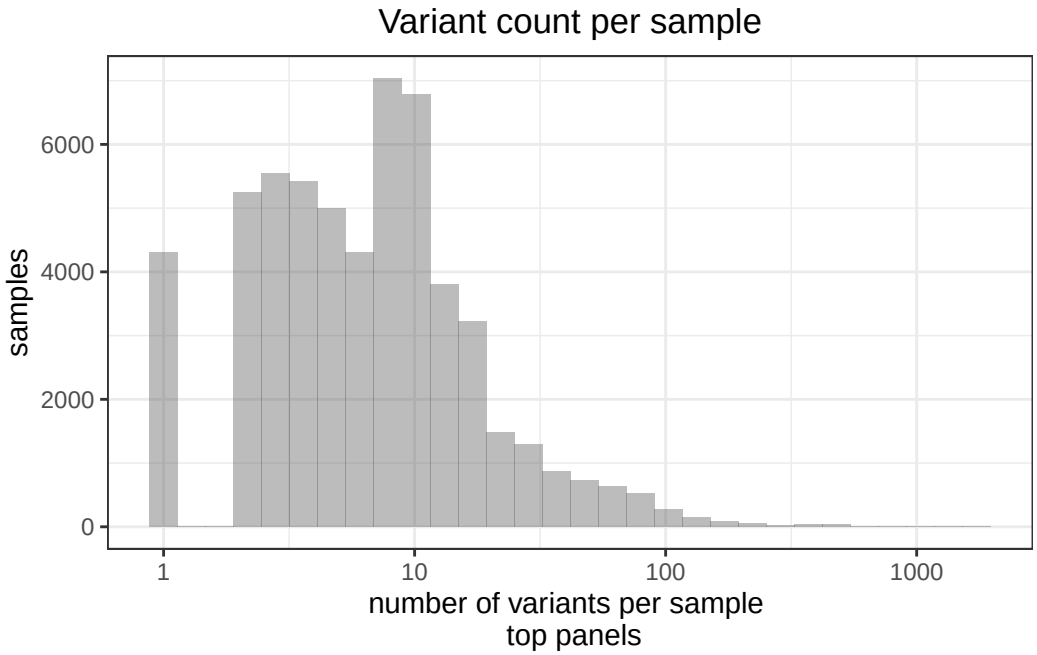
```
FALSE   TRUE
15746 145219
```

Variant counts

Variants per sample for all available data:



Variants per sample for three top panels: “MSK-IMPACT468”, “DFCI-ONCOPANEL-3.1”, “MSK-IMPACT505”.



Panels

Patients per panel type

SEQ_ASSAY_ID	number.patients.per.panel	number.genes.per.panel	number.distinct.AA.changes
MSK-IMPACT468	30713	478	182817
DFCI-ONCOPANEL-3.1	13976	498	116329
MSK-IMPACT505	13413	510	97813
PROV-TSO500HT-V2	9076	535	100986
MSK-IMPACT410	8759	414	52770
DFCI-ONCOPANEL-2	8154	383	50080
DFCI-ONCOPANEL-3	6647	489	61895
MSK-IMPACT-HEME-400	5547	396	17981
JHU-50GP	4984	51	3709
UCSF-IDTV5-TO	4406	574	71184

SEQ_ASSAY_ID	number.patients.per.panel	number.genes.per.panel	number.distinct.AA.changes
DFCI-ONCOPANEL-1	2976	284	17842
MSK-IMPACT341	2463	344	13229
CRUK-TS	2345	174	9676
UCSF-NIMV4-TO	2163	500	34204
DUKE-F1-DX1	1942	319	16347
COLU-CSTP-V1	1709	47	3732
MSK-IMPACT-HEME-468	1635	431	4982
UCSF-NIMV4-TN	1635	500	27257
VICC-01-T7	1590	325	16114
NKI-CHPV2-SOCV2-NGS	1316	41	763
VICC-01-MYELOID	1271	35	1720
MDA-50-V1	1266	47	953
JHU-500STP	1216	49	640
UHN-48-V1	1181	48	1179
DUKE-F1-T7	1047	324	9793
UHN-54-V1	1014	53	1586
YALE-OCP-V3	998	144	2838
MDA-46-V1	944	45	532
UCSF-IDTV5-TN	928	555	15215
VICC-01-SOLIDTUMOR	844	31	605
COLU-CCCP-V1	758	469	9577
GRCC-MOSC3	739	66	1038
SCI-PMP68-V1	723	66	1435
NKI-TSACP-MISEQ-NGS	678	41	560
COLU-CSTP-V2	630	29	1661
PROV-FOCUS-V1	554	34	212
COLU-TSACP-V1	531	52	2619
UHN-555-V1	490	561	16752
VICC-02-XTV4	428	707	7477
CHOP-STNGS	407	226	1777
VICC-01-T5A	396	242	3226
MDA-409-V1	395	348	2038
WAKE-CLINICAL-T7	395	80	214
PROV-TRISEQ-V2	387	310	2750
UHN-OCA-V3	379	45	241

SEQ__ASSAY_ID	number.patients.per.panel	number.genes.per.panel	number.distinct.AA.changes
VHIO-300	341	413	4028
GRCC-MOSC4	302	71	510
WAKE-CLINICAL-DX1	296	127	521
WAKE-CA-NGSQ3	281	283	759
UCHI-ONCOHEME55-V1	269	56	737
VICC-02-XTV3	267	694	5223
UHN-TSO500-V1	261	225	836
UHN-555-GYNE-V1	259	563	20390
VHIO-COLORECTAL-V01	241	30	283
YALE-HSM-V1	234	35	288
UHN-555-V2	209	564	7780
UCHI-ONCOSCREEN50-V1	194	42	257
VICC-01-DX1	188	297	1942
CHOP-COMPT	179	188	653
NKI-CHP-V2-PLUS	174	34	154
WAKE-CA-01	172	19	136
VHIO-PANCREAS-V01	167	23	140
VICC-01-D2	159	359	1795
UHN-555-PAN-GI-V1	117	550	6499
CHOP-HEMEP	113	98	379
GRCC-CHP2	99	25	130
DUKE-F1-T5A	84	202	778
NKI-CHPV2-NGS	74	19	96
VHIO-OVARY-V01	74	14	89
VHIO-BREAST-V02	69	22	76
WAKE-CLINICAL-R2D2	69	122	216
VHIO-GENERAL-V01	63	19	64
VHIO-HEAD-NECK-V01	58	14	73
VHIO-LUNG-V01	58	20	81
UHN-555-BREAST-V1	54	514	3006

SEQ__ASSAY_ID	number.patients.per.panel	number.genes.per.panel	number.distinct.AA.changes
VHIO-BREAST-V01	54	22	74
VHIO-GASTRIC-V01	51	18	74
YALE-OCP-V2	44	33	105
VHIO-BRAIN-V01	36	10	48
UHN-555-HEAD-NECK-V1	34	490	2378
VHIO-BILIARY-V01	31	13	43
NKI-PATH-NGS	28	9	36
VHIO-ENDOMETRIUM-V01	27	17	56
GRCC-CP1	26	13	40
UHN-555-LUNG-V1	26	473	2163
VICC-02-XTV2	24	340	671
VHIO-URINARY-BLADDER-V01	21	14	38
VHIO-KIDNEY-V01	16	7	21
VHIO-SKIN-V01	16	10	17
VICC-01-T4B	14	116	261
UHN-555-PROSTATE-V1	13	352	923
WAKE-CLINICAL-T5A	13	69	105
UHN-555-MELANOMA-V1	10	345	806
UHN-555-GLIOMA-V1	9	345	805
UHN-555-RENAL-V1	8	287	574
UHN-50-V2	7	5	5
UHN-555-BLADDER-V1	4	219	363
VHIO-PAROTIDE-V01	4	4	7
VICC-01-T6B	4	30	34
COLU-CCCP-V2	2	46	54
WAKE-CLINICAL-AB2	1	4	4
WAKE-CLINICAL-AB3	1	3	3

SEQ_ASSAY_ID	number.patients.per.panel	number.genes.per.panel	number.distinct.AA.changes
WAKE-CLINICAL-CF3	1	4	4
WAKE-CLINICAL-R2	1	4	4

The following plot shows the number of genes tested in each panel

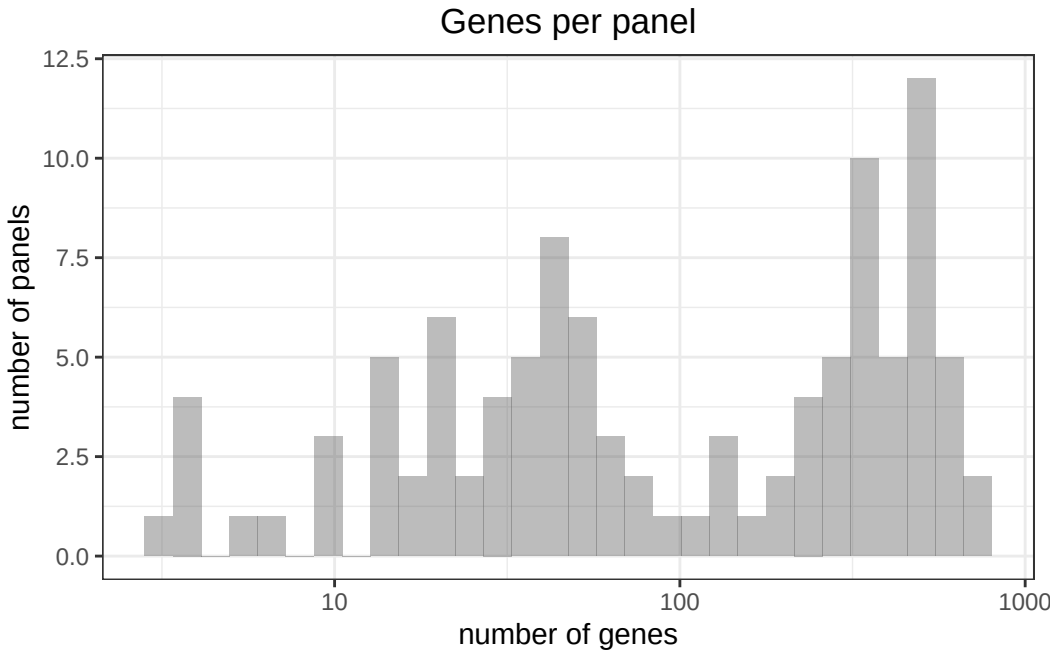


Figure 1: Number of genes tested across all assay panels in GENIE

How many samples (or patients) have been tested with more than one panel?

To-do: Is the coverage across panels? 10X-30X? Tumor and normal coverage?

Variant Effect and Protein annotation

Variant type

Variant_Type	variant_count	Percentage
DEL	141736	8.2741534
DNP	18308	1.0687701

Variant_Type	variant_count	Percentage
INS	61449	3.5872217
ONP	2403	0.1402805
SNP	1487530	86.8378637
TNP	1571	0.0917106

Variant Classificaitons

Variant_Classification	variant_count	Percentage
3'Flank	6314	0.3685938
3'UTR	3626	0.2116758
5'Flank	23870	1.3934642
5'UTR	3491	0.2037949
Frame_Shift_Del	96194	5.6155381
Frame_Shift_Ins	43753	2.5541784
In_Frame_Del	23505	1.3721565
In_Frame_Ins	9187	0.5363115
Intron	51690	3.0175184
Missense_Mutation	1123523	65.5881476
Nonsense_Mutation	118828	6.9368481
Nonstop_Mutation	908	0.0530065
RNA	1846	0.1077643
Silent	115459	6.7401753
Splice_Region	47540	2.7752530
Splice_Site	41399	2.4167585
Translation_Start_Site	1864	0.1088151

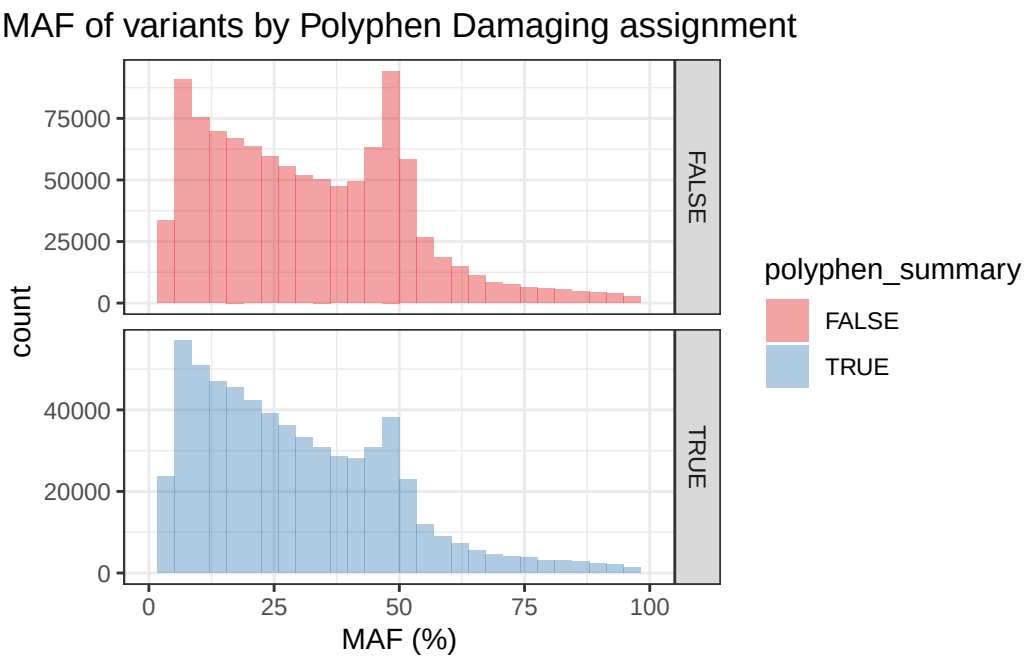
Concordance between Polyphen and SIFT predictions

```
table(genie.full$Polyphen_Prediction,genie.full$SIFT_Prediction)
```

```
                      deleterious deleterious_low_confidence tolerated
                   594406           0                       0         0
benign               3297      107757                   29706    301244
possibly_damaging    1039      113485                   12978     54096
probably_damaging    1304      381393                   17896     39106
unknown                69          27                     202        15
```

	tolerated_low_confidence
	0
benign	47420
possibly_damaging	4178
probably_damaging	3299
unknown	80

MAF by Polyphen “probably damaging” or “possibly damaging”



MAF by SIFT “deleterious” or “deleterious low confidence”

To do: Look at SIFT/polyphen scores for TSG genes only but leave out ONC

- read in Vogelstein list of genes

To do: read recent GENIE manuscripts. Have they looked at prevalence of common cancer biomarkers already? can we reproduce their findings?

To do: apply cancer type ontology scheme

To do: perform actionability matching (with and without cancer type matching)

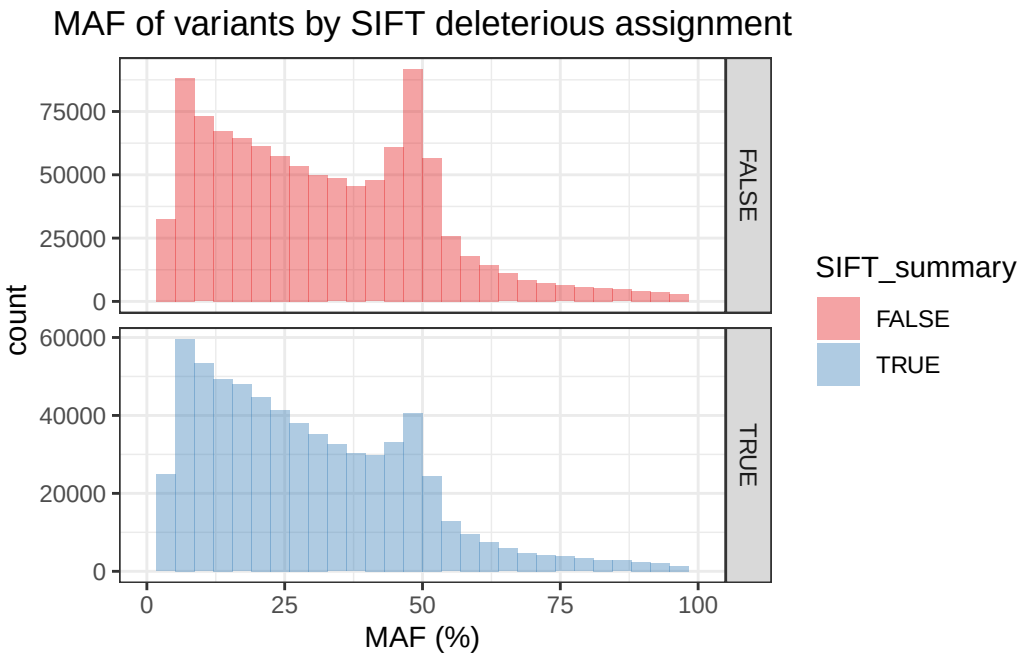


Figure 2: Variant MAF values for variants with and without SIFT deleterious assignments

Variant signatures by assay type

Perform demintionality reduction on base pair transition signatures

Background: Different assays can show different profiles of background variants based. If background signals vary substantially, this could represent an important confounding factor in interpreting mutaitonal signatures

Goal: The goal of this work is to identify different clusters of assay signature types by performing clustering on base pair transition signatures.

Assumptions: This works assumes that a large proportion of listed signatures are assay artifacts and that these background mutations drive clustering patterns. Other contributors could also influence the signature results include the diversity of cancer types tested and the segments of the genome tested by each panel.

Procedure

1. Filter to the GENIE variant set to SNV events only
2. Filter out any variants coming from assays that contribute fewer than 1,000 variants across all patients

3. Create a table that has a profile of variant counts for every combination of ref–>alt for each assay
4. Calculate what percentage of variants for a given assay type fit into any given combination of [ref base, tumor alt 2 base, strand].
5. Pivot these data into a matrix where rows are the different base pair combinations and columns are the different assay types. Each entry contains the percentage of variants that match a given base pair transition (e.g. A–>T) for that assay type.
6. Normalize the matrix
7. Perfrom tsNE clustering
8. Visualize clusters

The following table shows the results of step (3):

SEQ_ASSAY_ID	Reference_Allele	Tumor_Seq_Allele2	Strand	nVariants	assayVariantsTotal	percVariants
CHOP-STNGS	A	C	+	51	2064	2.470930
CHOP-STNGS	A	G	+	161	2064	7.800388
CHOP-STNGS	A	T	+	126	2064	6.104651
CHOP-STNGS	C	A	+	116	2064	5.620155
CHOP-STNGS	C	G	+	117	2064	5.668605
CHOP-STNGS	C	T	+	508	2064	24.612403
CHOP-STNGS	G	A	+	480	2064	23.255814
CHOP-STNGS	G	C	+	123	2064	5.959302
CHOP-STNGS	G	T	+	103	2064	4.990310
CHOP-STNGS	T	A	+	60	2064	2.906977

```
#Convert data to matrix form
assaySigTbl$ID <- paste0(assaySigTbl$Reference_Allele,"-",assaySigTbl$Tumor_Seq_Allele2,"-
sigMtrx <- assaySigTbl[,c("ID","SEQ_ASSAY_ID","percVariants")] %>%
  tidyr::pivot_wider(names_from = SEQ_ASSAY_ID,values_from=percVariants)
sigMtrx2 <- sigMtrx[,!colnames(sigMtrx) %in% c("ID")]
```

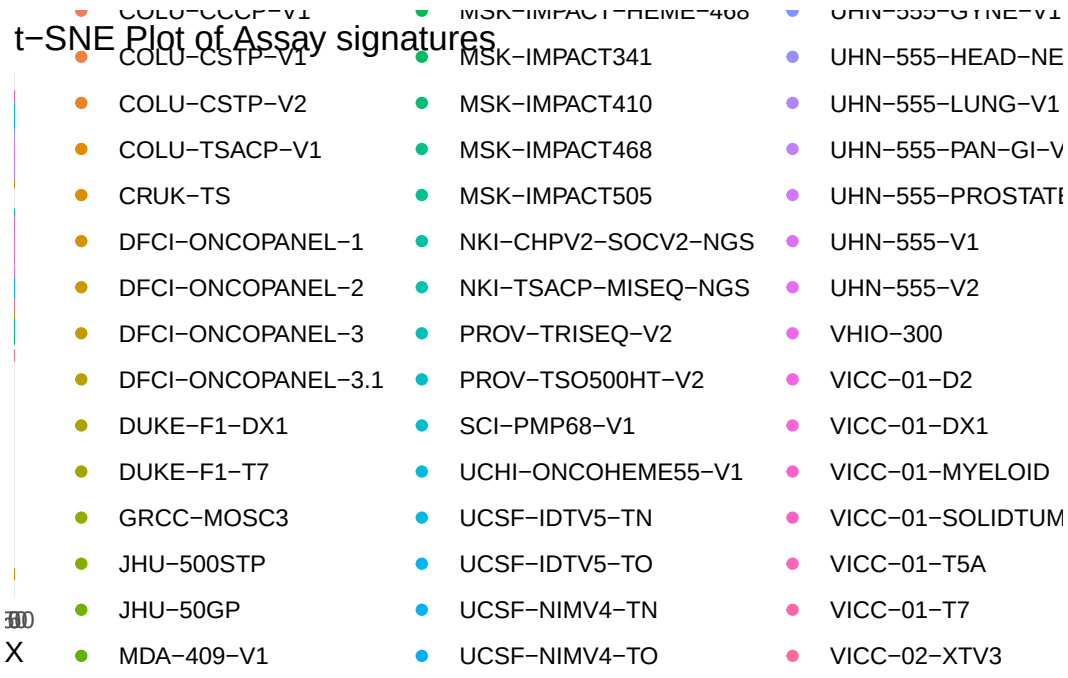
Perform dimentionality reduction on mutational signature vector space

```r
sigMtrxScale <- t(scale(sigMtrx2))

# Run t-SNE
set.seed(144) # For reproducibility
tsne_results <- Rtsne(sigMtrxScale, perplexity = 5, theta = 0.5, max_iter = 1000)

# Create a data frame for plotting
tsne_data <- data.frame(X = tsne_results$Y[,1], Y = tsne_results$Y[,2], assay = colnames(s

# Plot using ggplot2
ggplot(tsne_data, aes(x = X, y = Y, color = assay)) +
  geom_point() +
  theme_minimal() +
  ggtitle("t-SNE Plot of Assay signatures")
```




```r
### Why do the ref/alt calls not always match the bases listed in HGVSc?
head(genie.full[,c("Reference_Allele","Tumor_Seq_Allele1","Tumor_Seq_Allele2","HGVSc")])
```

```
  Reference_Allele Tumor_Seq_Allele1 Tumor_Seq_Allele2
1                C                                   A
2                A                                   T
```

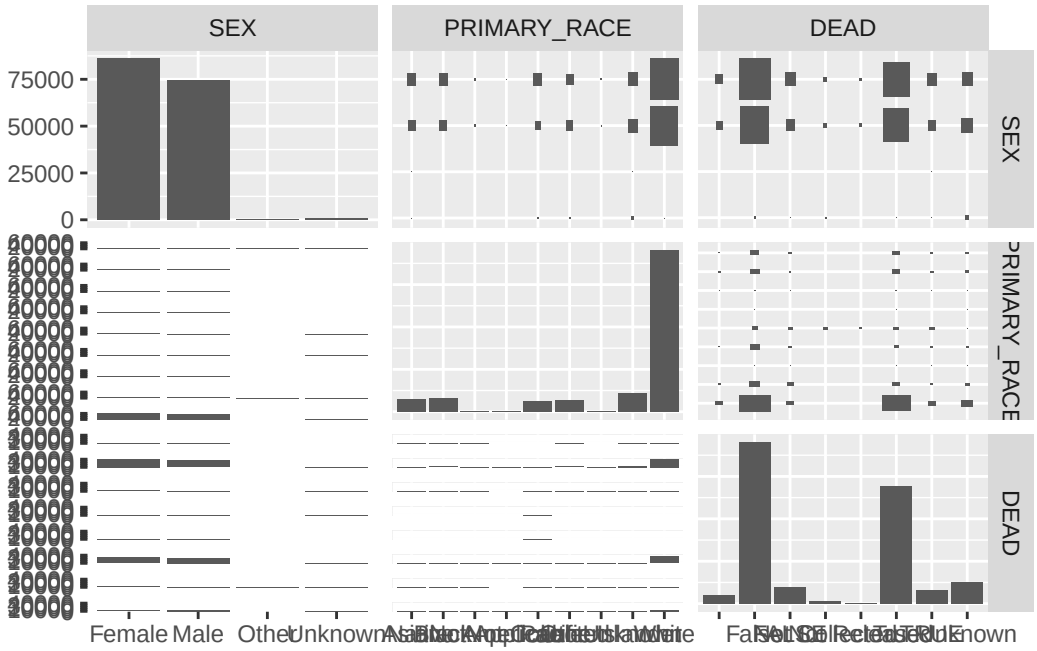
```
3                C                                   T
4                C                                   T
5                T                                   C
6                A                                   G
                        HGVSc
1   ENST00000256078.4:c.34G>T
2 ENST00000288602.6:c.1799T>A
3 ENST00000275493.2:c.2369C>T
4  ENST00000269305.4:c.818G>A
5  ENST00000369535.4:c.182A>G
6 ENST00000263967.3:c.3140A>G
```

Review of patient data

```
## sumarize patient data if numeric
numeric_summary <- patient.genie %>%
  dplyr::select_if(is.numeric) %>%
  #summarise_all(funs(mean, median, sd, min, max))
  #dplyr::summarise_all(tibble::lst(mean,median,sd,min,max))
  dplyr::summarise_all(list(mean = mean,
                            median = median))

categorical_summary <- patient.genie %>%
  select_if(is.factor) %>%
  dplyr::summarise_all(list(table = table,
                            n = length))
```

```
ggpairs(patient.genie[,c("SEX","PRIMARY_RACE","DEAD")])
```




```
patient_py = reticulate::r_to_py(patient.genie)
```

Print data from python

```
r.patient_py.head()
```

```
          PATIENT_ID     SEX PRIMARY_RACE  ... YEAR_CONTACT   DEAD      YEAR_DEATH
0  GENIE-VICC-101416  Female        White  ...          2014  False  Not Applicable
1  GENIE-VICC-102225  Female        White  ...          2015   True            2017
2  GENIE-VICC-102424  Female        White  ...          2016   True            2016
3  GENIE-VICC-102966    Male        White  ...          2015   True            2015
4  GENIE-VICC-103244  Female      Unknown  ...          2014   True            2014

[5 rows x 10 columns]
```

```
import pandas as pd
import matplotlib.pyplot as plt
#import seaborn as sns
```

```python
# Load the dataset
data = r.patient_py

# Display basic information about the dataset
data_info = data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 160965 entries, 0 to 160964
Data columns (total 10 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   PATIENT_ID    160965 non-null  object
 1   SEX           160965 non-null  object
 2   PRIMARY_RACE  160965 non-null  object
 3   ETHNICITY     160965 non-null  object
 4   CENTER        160965 non-null  object
 5   INT_CONTACT   160965 non-null  object
 6   INT_DOD       160965 non-null  object
 7   YEAR_CONTACT  160965 non-null  object
 8   DEAD          160965 non-null  object
 9   YEAR_DEATH    160965 non-null  object
dtypes: object(10)
memory usage: 12.3+ MB
```

```python
# Summarize categorical data
categorical_summary = data.describe(include=['object'])

# Displaying the first few rows of the dataset for a quick overview
first_rows = data.head()

data_info, categorical_summary, first_rows
```

```
(None,                PATIENT_ID     SEX  ...    DEAD      YEAR_DEATH
count              160965  160965  ...  160965          160965
unique             160965       4  ...       8              54
top     GENIE-VICC-101416  Female  ...   False  Not Applicable
freq                    1   86078  ...   76021           83681

[4 rows x 10 columns],           PATIENT_ID     SEX PRIMARY_RACE  ... YEAR_CONTACT   DEAD
0  GENIE-VICC-101416  Female        White  ...         2014  False  Not Applicable
```

```
1  GENIE-VICC-102225  Female        White  ...         2015   True            2017
2  GENIE-VICC-102424  Female        White  ...         2016   True            2016
3  GENIE-VICC-102966    Male        White  ...         2015   True            2015
4  GENIE-VICC-103244  Female      Unknown  ...         2014   True            2014

[5 rows x 10 columns])
```

Review of survival data

```
table(as.numeric(patient.genie$YEAR_DEATH),exclude=NULL)
```

```
Warning in table(as.numeric(patient.genie$YEAR_DEATH), exclude = NULL): NAs
introduced by coercion
```

1900	1950	1977	1980	1981	1982	1983	1984	1985	1986	1987
1	1	1	1	4	3	2	3	6	6	7
1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998
9	8	14	16	33	36	51	57	79	77	64
1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009
59	67	62	72	62	64	68	62	78	73	50
2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
44	36	72	352	1669	3781	5453	6835	7901	8310	9092
2021	2022	2023	<NA>							
8075	5758	493	101898							

Additional modeling ideas

- Does having an actionable mutation correlate with better or worse survival outcomes?
- Supervised
  - Tissue of origin prediction based on various biomarkers
  - Identifying mutational signatures that correlate with outcomes and/or cancer types
  - Mutational signatures that correlate with a given assay type
- Unsupervised modeling
  - Clustering of mutation patterns for each assay by tsne or PCA