Workshop Mutation 2019

Lee and Martell

February 5, 2019

workshop.mutation.2019

In this workshop you will learn about the fundamentals of mutation identification in cancer. This will include the following.

1. Basic bioinformatic workflow: from sample to identifying potentially targetable mutations. Types of sequencing, e.g. WES and WGS, and also DNA vs RNA Standard pipelines to go from reads to actionable information The size of modern data sets (TCGA) and their current applications, e.g. in healthcare What does a standard genome look like? For example, 4-5 million variants in an average genome compared to the reference human genome. Then this can lead in to talking about germline vs somatic mutations, and how this is crucial for studying cancer

2. Basic vocabulary and concepts.

- Classes of somatic mutations

Pt mutations

- Coding
- Silent
- Missense
- Nonsense
- Noncoding (UTR)
- Intronic
- Intergenic
- Splice site variants?

Small regional mutations

- Insertion
- Deletions
- Duplications

3. Deciphering nomenclature of sequence variations.

What is his section? Is this deciphering single letter vs three letter mutatioion codes etc.?

4. Identifying functionally relevant mutations - passenger vs driver How RNA and DNA sequencing data can be integrated to find functional variants? Variant prediction tools - e.g. CADD scores, and other methods Comparison to known cancer genes or even known cancer causing variants Pathway analysis - how this leads in to the design/search of drugs Structural biology for coding variants

Workshop (this will constitute the bulk of the workshop)

- We will start off with basic data mining. To begin with will learn how to directly download mutation data from R. There are many sources and API's however here will be using cbioportal.

- Although this is not a course in R per se, but you will learn how to manipulate/wrangle a the mutation data.frame.

- Subset type of mutations.

- Aggregate by attributes such as types of mutations.
- Query (eg. for specific variants)
- Tabulate mutations (eg. frequency tables) .
- How to identify what could be potentially be pathogenic and cross reference it with existing data.
- How to take existing mutation data and predict possible actionable targets for either druggability, diagnostic or prognosis.
- You will also learn a few ways to plot the data.
- Basic plotting of your mutation table.
- How to generate figures to look for total burden across different chromosome/regions.
- Potentially - how to analyse mutational signatures (if there is time)

```
library ( cgdsr)
```

```
## Warning: package 'cgdsr' was built under R version 3.4.4
```

```
## Please send questions to cbioportal@googlegroups.com
```

```
library ( ggplot2)
# Create CGDS object
mycgds = CGDS("http://www.cbioportal.org/public-portal/")


# Get list of cancer studies at server
studies = getCancerStudies(mycgds)
# lets take a look at it.
View ( head ( studies, 10) )
# what columns are in here, this gives you an idea what the dataframe consist of.
names ( studies )
```

```
## [1] "cancer_study_id" "name"            "description"
```

```
# lets find how many unique sets are in here
dim ( studies )
```

```
## [1] 240   3
```

```
# so it looks like 240 studies

# Find the dataset you want.
# According to the syllabus we are looking for TCGA study on breast cancer from Cell, 2015
# so lets search for it using the grepl function
breast = studies[ grepl("Breast", studies$name, ignore.case = T ), ]
View ( breast )

### BONUS EXERCISE: Level 1:  TRY TO FIND SOME OTHER CANCER GROUP, eg Lung Cancer

# from there we find what we are looking for and its called, brca_tcga_pub2015

brca.study = "brca_tcga_pub2015"

# lets see if you sample is in here.
mycaselist = getCaseLists(mycgds,brca.study)
```

```
# first lets study what is in the case list.
View ( head ( mycaselist, 10) )
# here we see that brca_tcga_pub2015_3way_complete is probably the best to use because it includes
# only All Complete Tumors vs something like All tumor samples with methylation data
case.list.id = "brca_tcga_pub2015_3way_complete"
### BONUS EXERCISE: Level 2: count the total categories available for this set.

# ok now using All Complete Tumors lets see if your sample is present.

mysample = mycaselist[ mycaselist$case_list_id == case.list.id, ]$case_ids
mysample[ grepl("TCGA-OL-A66K-01", mysample)]
```

```
## [1] "TCGA-LQ-A4E4-01 TCGA-A2-A3KC-01 TCGA-A2-A3KD-01 TCGA-A7-A0D9-01 TCGA-A7-A0DA-01 TCGA-A7-A0CD-01
```

```
mysample[ grepl("TCGA-FAKE-A66K-01", mysample)]
```

```
## character(0)
```

```
### BONUS EXERCISE: Level 1: search for you sample and see if its there.


### Level 3: BONUS EXERCISE: loop through each one and find the the samples that All tumor samples with


# Now lets see what information is available for your study
mygeneticprofile = getGeneticProfiles(mycgds,brca.study)
View ( mygeneticprofile )
# so looking at that we now know it contains several interesting modalities.  Lets try the mutation
mutation = mygeneticprofile[11, 1]

# ok now we can get the actual mutation data. However first lets download a set of genes that are known
# grab this from cosmic
cosmic = read.csv("https://www.dropbox.com/s/naheek0wicegf77/cancer.list.csv?dl=1")
# we dont need the annotations just the gene however lets take a look at this.
View ( head (cosmic ))

cancer.gene = as.character ( unique ( cosmic$gene) )
length( cancer.gene)
```

```
## [1] 1805
```

```
# as you can see there are 719 genes.
# lets see if BRCA is in here to make sure!
cosmic[grepl("^BRCA|^ATM$|^BARD1$|^CDH1$|^CHEK2$|^NBN$|^NF1$|^PALB2$|^PTEN$", cosmic$gene), ]
```

```
##       gene custom cosmic clinvar.all.cancer FM.combined ucsf500v2 civic
## 108    ATM             X                  X           X         X     X
## 129  BARD1                                X           X         X
## 159  BRCA1             X                  X           X         X     X
## 160  BRCA2             X                  X           X         X     X
## 248   CDH1             X                  X           X         X     X
## 279  CHEK2             X                  X           X         X     X
## 1032   NBN             X                  X                     X     X
## 1052   NF1             X                  X           X         X     X
## 1128 PALB2             X                  X           X         X     X
## 1265  PTEN             X                  X           X         X     X
```

```
##      broad.target msk.impact umich.mioncoseq
## 108             X          X               X
## 129                        X
## 159             X          X
## 160             X          X               X
## 248             X          X               X
## 279                        X
## 1032                       X
## 1052            X          X               X
## 1128                       X               X
## 1265            X          X               X
```

```
# BONUS why did we add ^ in front and $ for only some genes and not others?

### BONUS can you check if your favorite gene is in here?
# lets collect this through a loop so not to overwheml the system

total = ceiling ( length( cancer.gene)/100  )  *100
mutations = data.frame (   stringsAsFactors = F )
e = 1
for(i in seq(from=200, to=total, by=200)){

    if ( i > length(cancer.gene)){
        i = length(cancer.gene)
    }

    print ( paste ( e, i ))
    temp = getMutationData(mycgds , brca.study, mutation, cancer.gene[e:i])
    e = i
    mutations = rbind ( mutations, temp)

    Sys.sleep (2) # lets give the system a break

}
```

```
## [1] "1 200"
## [1] "200 400"
## [1] "400 600"
## [1] "600 800"
## [1] "800 1000"
## [1] "1000 1200"
## [1] "1200 1400"
## [1] "1400 1600"
## [1] "1600 1800"
```

```
dim ( mutations )
```

```
## [1] 9095   22
```

```
colnames ( mutations )
```

```
##  [1] "entrez_gene_id"             "gene_symbol"
##  [3] "case_id"                    "sequencing_center"
##  [5] "mutation_status"            "mutation_type"
##  [7] "validation_status"          "amino_acid_change"
##  [9] "functional_impact_score"    "xvar_link"
```

```
## [11] "xvar_link_pdb"               "xvar_link_msa"
## [13] "chr"                         "start_position"
## [15] "end_position"                "reference_allele"
## [17] "variant_allele"              "reference_read_count_tumor"
## [19] "variant_read_count_tumor"    "reference_read_count_normal"
## [21] "variant_read_count_normal"   "genetic_profile_id"
```

```
# lets take a few minutes here to go over the different fields.

# lets check if ALL your samples are availble

samples = c("TCGA-C8-A3M7-01","TCGA-GM-A2D9-01","TCGA-BH-A1FL-01", "TCGA-E2-A14Z-01","TCGA-BH-A1FC-01",

final =  mutations [ mutations$case_id %in% samples, ]


unique ( final$case_id)
```

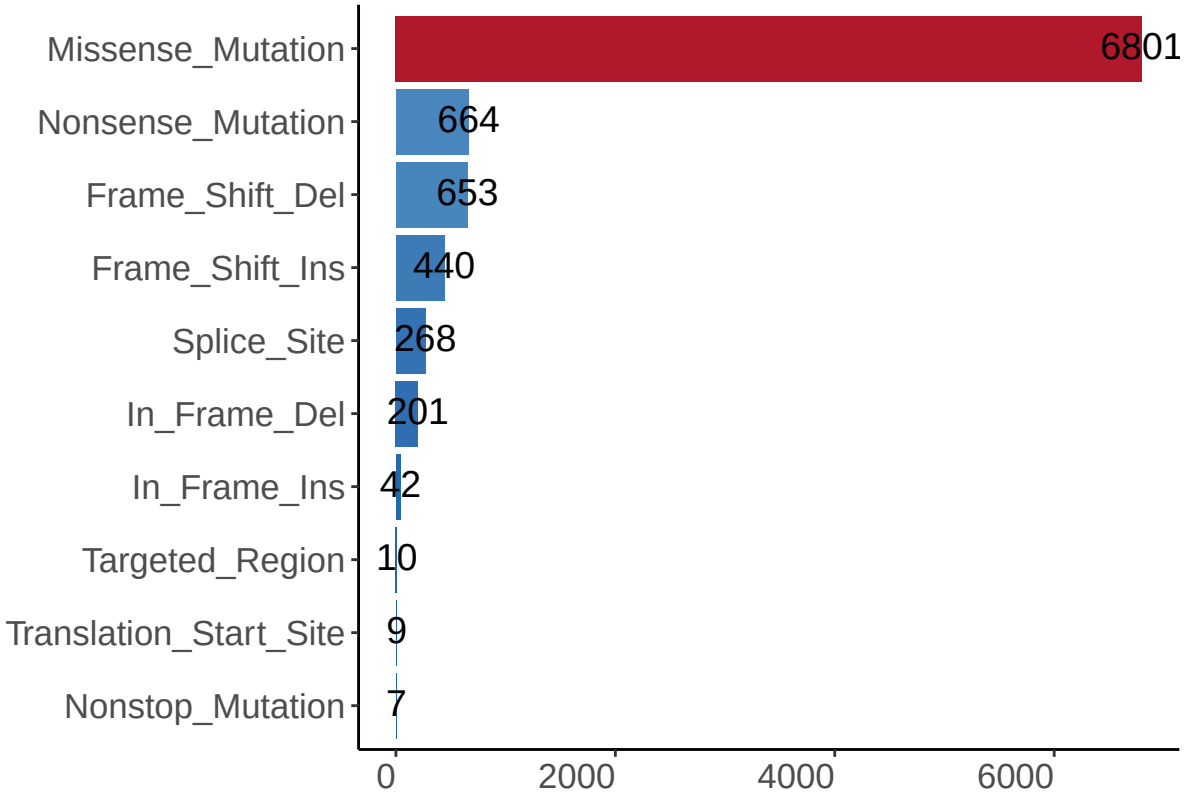
```
##  [1] "TCGA-C8-A3M7-01" "TCGA-E2-A1LK-01" "TCGA-EW-A1P8-01"
##  [4] "TCGA-BH-A1FC-01" "TCGA-GM-A2D9-01" "TCGA-E2-A14Z-01"
##  [7] "TCGA-BH-A1FL-01" "TCGA-E2-A1LE-01" "TCGA-AC-A2FE-01"
## [10] "TCGA-AC-A2QJ-01"
```

```
# lets study your brca!

# lets see what are the mutations types.

unique ( mutations$mutation_type)
```

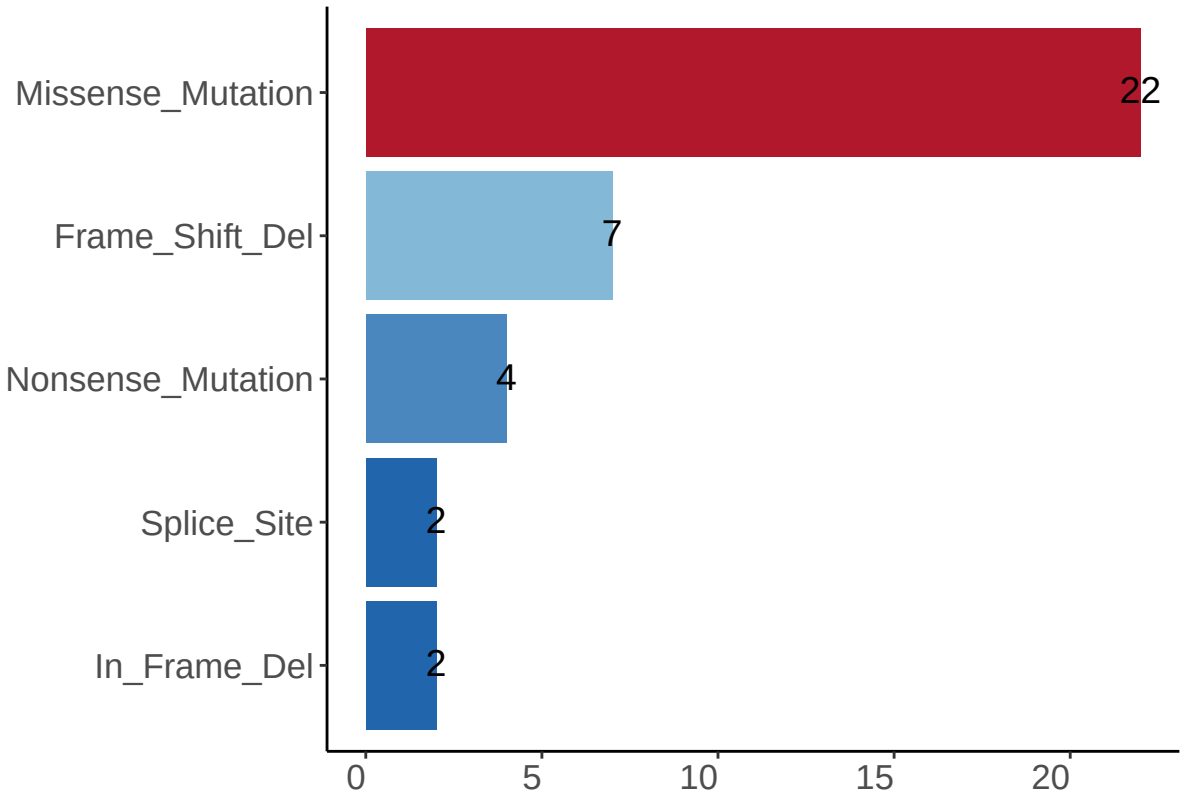
```
##  [1] "Missense_Mutation"      "Nonsense_Mutation"
##  [3] "Splice_Site"            "Frame_Shift_Ins"
##  [5] "Frame_Shift_Del"        "In_Frame_Del"
##  [7] "In_Frame_Ins"           "Translation_Start_Site"
##  [9] "Targeted_Region"        "Nonstop_Mutation"
```

```
# lets talk about each one.

# lets tabulate the types of mutations mostly seen with BRCA

mutation.type = data.frame ( table ( mutations$mutation_type))

# BONUS compare this with another disease cohort.



mutation.type = mutation.type[ order ( mutation.type$Freq), ]
mutation.type$Var1 = factor ( mutation.type$Var1, levels = mutation.type$Var1)

ggplot(mutation.type, aes(Var1,Freq), label=Freq ) +
    geom_bar(aes(fill = Freq), stat="identity", position = "dodge") +
    coord_flip() +
    scale_fill_distiller(palette = "RdBu") + xlab("") + ylab("") +
    theme(strip.text.y = element_text(angle = 0), legend.position="none") +
    geom_text(aes(label=Freq), position=position_dodge(width=0.9), vjust=.4, hjust = .5, size=5) +
    theme(panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
```

```
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        text = element_text(size=16),  # size of label
        axis.text.x = element_text(angle=0, hjust=1) )
```




```
# lets figure out what type of mutations are BRCA
### how would you do this?
#### .... 5 mins

mutation.brca = data.frame ( table ( mutations[ grepl("BRCA", mutations$gene_symbol), ] $mutation_type))

mutation.brca = mutation.brca[ order ( mutation.brca$Freq), ]
mutation.brca$Var1 = factor ( mutation.brca$Var1, levels = mutation.brca$Var1)

ggplot(mutation.brca, aes(Var1,Freq), label=Freq ) +
    geom_bar(aes(fill = Freq), stat="identity", position = "dodge") +
    coord_flip() +
    scale_fill_distiller(palette = "RdBu") + xlab("") + ylab("") +
    theme(strip.text.y = element_text(angle = 0), legend.position="none") +
    geom_text(aes(label=Freq), position=position_dodge(width=0.9), vjust=.4, hjust = .5, size=5) +
    theme(panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
          panel.background = element_blank(),
          axis.line = element_line(colour = "black"),
          text = element_text(size=16),  # size of label
          axis.text.x = element_text(angle=0, hjust=1) )
```




```
# Bonus can you figure out which samples have the BRCA samples and which don't?

# now lets see what are the main genes that are present in missense mutations.

genes.missense = mutations[mutations$mutation_type == "Missense_Mutation", ]

genes.missense.genes = data.frame ( table ( genes.missense$gene_symbol))
genes.missense.genes = genes.missense.genes[ order ( -genes.missense.genes$Freq), ]

View ( head ( genes.missense.genes, 15))
```