Correlation of somatic SNV mutations with EGFR status and Proteome Expression

2017140086 Jungeun Ji

1. Background

Our data, which consists of lung cancer patients in East Asia, is characterized by predominant EGFR mutations. Known mutation of EGFR gene are L858R mutation and exon 19 deletion.

Among these patients, 337 SNV mutated peptides corresponding to 319 proteins were identified. Some of the protein expressions are upregulated with cancer, while others are downregulated. Variant isoforms of cancer driver genes were identified among the proteins. TP53BP1, RNF213, and KRAS mutations are in the top ranking genes.

This project focuses on the correlation between SNV types in proteins and other factors. First, we will find out their correlation with EGFR mutation type, and how the relations differ among different smoking status. Then, we will look into how the distribution of SNV types differs according to protein expression change(upregulation or downregulation) upon cancer.

2. Exploring Data

2.1. Importing Data

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(readr)
```

```
# importing characteristics and clinical data of TW lung cancer patients
clinical_patient <- read_csv("clinical_patient.csv")
```

```
## Rows: 103 Columns: 9

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (8): ID, Proteome_Batch, Gender, Smoking Status, Histology Type, Stage, ...
## dbl (1): Age

##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
# importing list of translated somatic SNV mutations on proteins
snv_info <- read_csv("snv_info.csv")
```

```
## Rows: 377 Columns: 15

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (7): Acession number_mutation, Protein, Chromosome, Reference_Allele, Al...
## dbl (8): No. of patients, positition, WES_depth, WES_ALF, RNAseq_T_depth, RN...

##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
#importing protein expression data (calculated as log2TN)
protein_exp <- read_csv('ProteomeLog2TN.csv')
```

```
## Rows: 14034 Columns: 92

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): Accession, Gene, Protein
## dbl (89): P002, P006, P007, P009, P010, P011, P012, P013, P015, P016, P017, ...

##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

Let’s find out what column names were used in these datasets.

```
colnames(clinical_patient)
```

```
## [1] "ID"                     "Proteome_Batch"         "Gender"
## [4] "Age"                    "Smoking Status"         "Histology Type"
## [7] "Stage"                  "EGFR_Status"            "Primary Tumor Location"
```

```
colnames(snv_info)
```

```
##  [1] "Acession number_mutation" "Protein"
##  [3] "No. of patients"          "Chromosome"
##  [5] "positition"               "Reference_Allele"
##  [7] "Alteration"               "Patient ID"
##  [9] "Batch"                    "WES_depth"
## [11] "WES_ALF"                  "RNAseq_T_depth"
## [13] "RNAseq_T_AF"              "RNAseq_N_depth"
## [15] "RNAseq_N_AF"
```

```
colnames(protein_exp)
```

```
##  [1] "Accession" "Gene"      "Protein"   "P002"      "P006"      "P007"
##  [7] "P009"      "P010"      "P011"      "P012"      "P013"      "P015"
## [13] "P016"      "P017"      "P018"      "P019"      "P020"      "P021"
## [19] "P022"      "P023"      "P024"      "P025"      "P026"      "P027"
## [25] "P028"      "P029"      "P030"      "P031"      "P032"      "P033"
## [31] "P034"      "P036"      "P037"      "P039"      "P040"      "P042"
## [37] "P043"      "P044"      "P045"      "P048"      "P049"      "P050"
## [43] "P051"      "P052"      "P053"      "P054"      "P055"      "P056"
## [49] "P057"      "P058"      "P059"      "P060"      "P061"      "P062"
## [55] "P063"      "P064"      "P066"      "P067"      "P068"      "P070"
## [61] "P071"      "P072"      "P073"      "P074"      "P075"      "P076"
## [67] "P077"      "P080"      "P081"      "P082"      "P085"      "P086"
## [73] "P088"      "P089"      "P090"      "P091"      "P092"      "P093"
## [79] "P094"      "P095"      "P097"      "P098"      "P099"      "P100"
## [85] "P101"      "P102"      "P103"      "P104"      "P109"      "P110"
## [91] "P111"      "P112"
```

2.2. Modifying Data

2.2.1. Adding new columns to snv_info

1. 'snv_type' Right now, our snv_info data is providing the reference allele and alternative allele at a seperate column.

```
snv_info %>%
  select(Protein, Chromosome, Reference_Allele, Alteration)
```

```
## # A tibble: 377 x 4
##    Protein Chromosome Reference_Allele Alteration
##    <chr>   <chr>      <chr>            <chr>
##  1 ABCF3   chr3       G                A
##  2 ACAA1   chr3       T                C
##  3 ACAN    chr15      A                G
##  4 ACAT2   chr6       A                G
##  5 ACIN1   chr14      A                G
##  6 ACIN1   chr14      T                C
##  7 ACOT2   chr14      A                G
##  8 ACSL1   chr4       C                T
##  9 ACSL5   chr10      G                 A
## 10 ACTN2   chr1       G                 A
## # ... with 367 more rows
```

Let’s make a new column, ‘snv_type’ where we can check the nucleotide change at once.

```
snv_info <- snv_info %>%
mutate(snv_type = paste(as.character(.$Reference_Allele), '>', as.character(.$Alteration)))

snv_info %>% select(snv_type)
```

```
## # A tibble: 377 x 1
##    snv_type
##    <chr>
##  1 G > A
##  2 T > C
##  3 A > G
##  4 A > G
##  5 A > G
##  6 T > C
##  7 A > G
##  8 C > T
##  9 G > A
## 10 G > A
## # ... with 367 more rows
```

2. ‘top_cancer_driver’ We are also going to add a new column named ‘top_cancer_driver’, indicating the peptides that are isoforms of top rank cancer driver genes, mentioned above.

```
snv_info <- snv_info %>%
  mutate(top_cancer_driver = ifelse(Protein %in% c('TP53BP1', 'KRAS', 'RNF213'), 'Y', 'N'))
```

3. ‘log2TN’, ‘tumor_expression’ Let’s add a column indicating whether the protein was upregulated or downregulated in the cancer patients, compared to normal protein expression.

```
#merging snv_info data and protein_exp data
snv_info <- rename(snv_info, Protein_name = Protein)
protein_exp <- rename(protein_exp, Protein_name = Gene)

merged_test <- merge(snv_info, protein_exp, by = 'Protein_name' )
```

```
#log2T/N value of specific patient with specific protein mutation given in snv_info
merged_test <- merged_test %>%
  mutate(log2TN = as.numeric(merged_test[cbind(1:360, match(.$`Patient ID`, names(merged_test)))]))
```

```
#check protein expression change in tumor
merged_test <-
  merged_test %>% mutate(tumor_expression = ifelse (log2TN > 0,'U','D'))
```

Let’s check our final data, with the information that is required for our plot!

```
head(merged_test %>% select(Protein_name, `Patient ID`, snv_type, top_cancer_driver, log2TN, tumor_expr
```

```
##   Protein_name Patient ID snv_type top_cancer_driver log2TN tumor_expression
## 1        ABCF3       P020    G > A                 N  0.199                U
## 2        ACAA1       P054    T > C                 N -0.062                D
## 3         ACAN       P054    A > G                 N  0.504                U
## 4        ACAT2       P048    A > G                 N  0.125                U
## 5        ACIN1       P049    A > G                 N  0.017                U
## 6        ACIN1       P049    T > C                 N  0.017                U
```

###2.2.2 Merging Data

Now let's merge our two datasets together! We will merge by patient ID.

```
snv_data <- merged_test %>% select(Protein_name, `Patient ID`, snv_type, top_cancer_driver, log2TN, tumo

snv_data <- snv_data %>% rename(ID = `Patient ID`)

merged_ds <- merge(snv_data, clinical_patient, by = 'ID')
head(merged_ds)
```

```
##     ID Protein_name snv_type top_cancer_driver log2TN tumor_expression
## 1 P002        H2AFY    C > G                 N -0.376                D
## 2 P002       NDUFS3    G > A                 N  0.123                U
## 3 P002        TOR3A    G > C                 N  0.185                U
## 4 P007        NAMPT    G > C                 N  3.234                U
## 5 P009        SPTA1    C > G                 N -2.264                D
## 6 P009         MYH7    C > G                 N -0.421                D
##   Proteome_Batch Gender Age Smoking Status Histology Type Stage EGFR_Status
## 1          B01-2   Male  74       Nonsmoke            ADC    IB      others
## 2          B01-2   Male  74       Nonsmoke            ADC    IB      others
## 3          B01-2   Male  74       Nonsmoke            ADC    IB      others
## 4          B02-3   Male  67       Nonsmoke            ADC   IIA          WT
## 5          B03-1 Female  54       Nonsmoke            ADC   IIA       L858R
## 6          B03-1 Female  54       Nonsmoke            ADC   IIA       L858R
##   Primary Tumor Location
## 1                    LUL
## 2                    LUL
## 3                    LUL
## 4                    RLL
## 5                    LLL
## 6                    LLL
```

3. Visualizing Data

3.1. SNV type & EGFR status

3.1.1. Creating Main Plot

What we want to know is the ratio of SNV type to each EGFR status.

To get this information, we will create a new dataset called 'snv_percentage', containing the snv rate of each snv type within each EGFR status, divided by Smoking Status.

```
snv_percentage <- merged_ds %>%
  group_by(`Smoking Status`, EGFR_Status) %>%
  count(snv_type) %>%
  summarize(snv_type = snv_type, snv_rate = n/sum(n))
```

```
## 'summarise()' has grouped output by 'Smoking Status', 'EGFR_Status'. You can override using the '.gro
```

```
snv_percentage
```

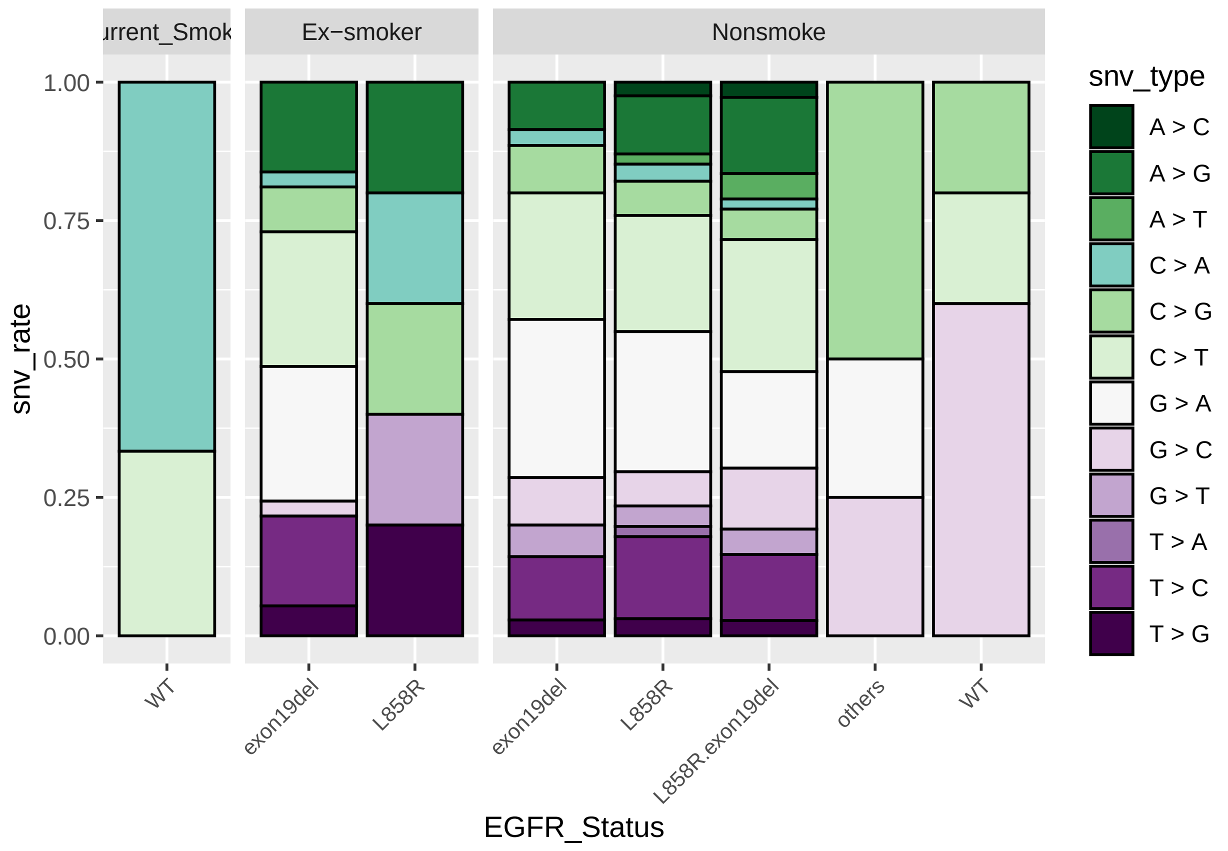
```
## # A tibble: 53 x 4
## # Groups:   Smoking Status, EGFR_Status [8]
##    'Smoking Status' EGFR_Status snv_type snv_rate
##    <chr>            <chr>       <chr>       <dbl>
##  1 Current_Smoker   WT          C > A      0.667
##  2 Current_Smoker   WT          C > T      0.333
##  3 Ex-smoker        exon19del   A > G      0.162
##  4 Ex-smoker        exon19del   C > A      0.0270
##  5 Ex-smoker        exon19del   C > G      0.0811
##  6 Ex-smoker        exon19del   C > T      0.243
##  7 Ex-smoker        exon19del   G > A      0.243
##  8 Ex-smoker        exon19del   G > C      0.0270
##  9 Ex-smoker        exon19del   T > C      0.162
## 10 Ex-smoker        exon19del   T > G      0.0541
## # ... with 43 more rows
```

Now let's visualize our data! We will be using a bar plot.

```
pal = c('#00441b','#1b7837', '#5aae61','#80cdc1',
        '#a6dba0','#d9f0d3' , '#f7f7f7','#e7d4e8',
        '#c2a5cf', '#9970ab' ,'#762a83','#40004b')

snv_bar <- snv_percentage %>%
  ggplot(aes(EGFR_Status, snv_rate, fill = snv_type))+
    geom_bar(stat = 'identity', color = 'black')+
    facet_grid(~ `Smoking Status`, scale = 'free', space = 'free_x')+
  scale_fill_manual(values = pal)+
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1, size = 8))

snv_bar
```



3.1.2. Highlighting top__cancer__driver

We want to highlight our top cancer driver genes, since there is a high chance that their SNV mutation is actually related to EGFR mutation and patients' smoking status.

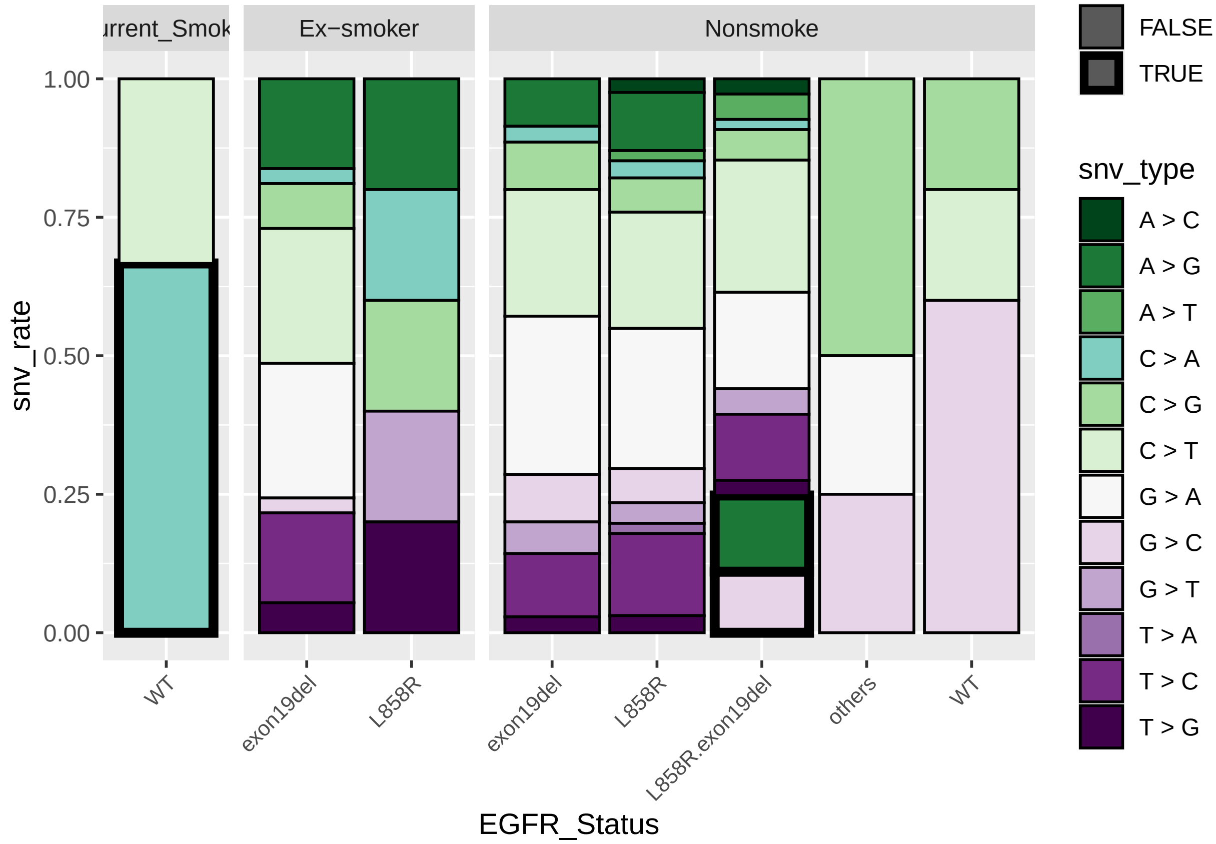
First, let's check which categories they belong to

```
merged_ds %>%
  filter(top_cancer_driver == 'Y') %>%
  select(ID, Protein_name, `Smoking Status`, EGFR_Status, snv_type)
```

```
##     ID Protein_name Smoking Status     EGFR_Status snv_type
## 1 P051      TP53BP1       Nonsmoke L858R.exon19del    G > C
## 2 P051       RNF213       Nonsmoke L858R.exon19del    G > C
## 3 P051       RNF213       Nonsmoke L858R.exon19del    A > G
## 4 P061         KRAS Current_Smoker              WT    C > A
```

We have four peptides that are isoforms of top cancer driver genes, which we will now highlight within our plot.

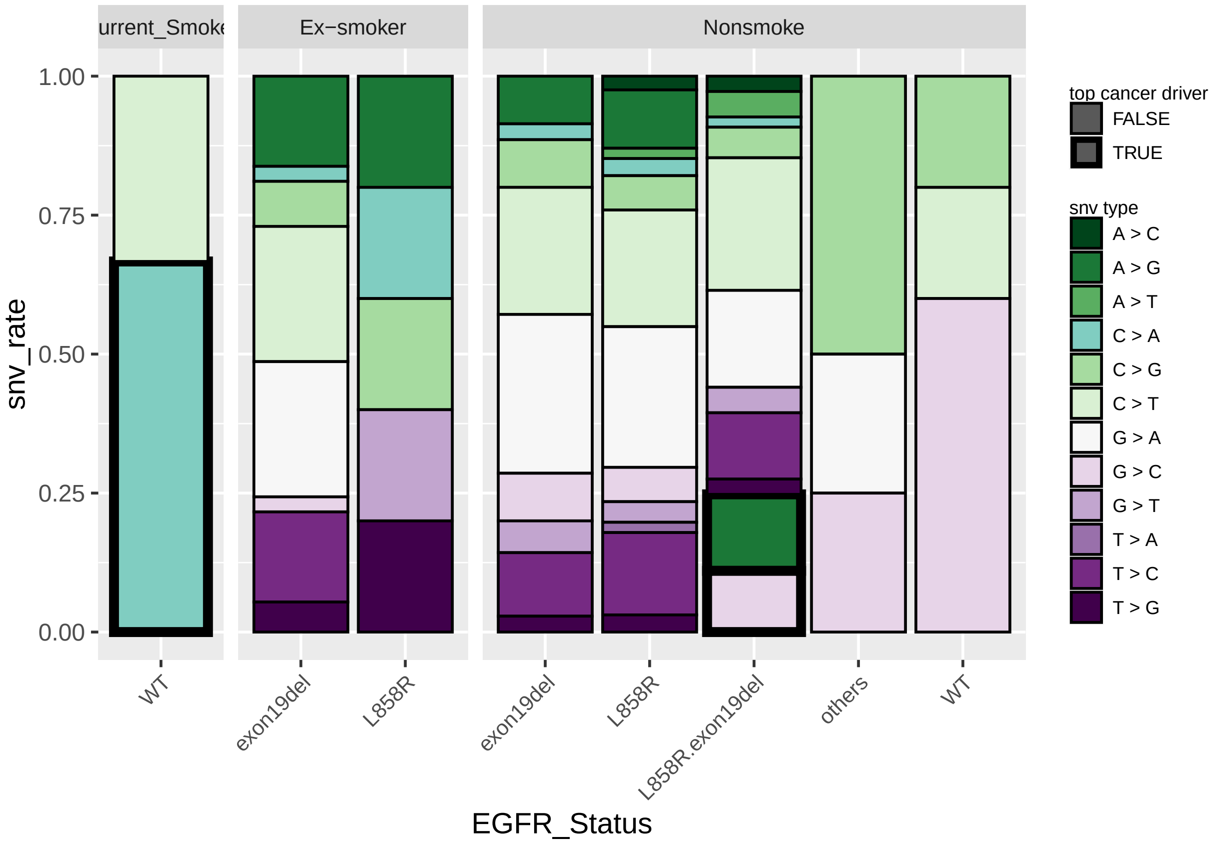
```
snv_percentage %>%
  mutate(tcd = (`Smoking Status` == 'Nonsmoke' &
                        EGFR_Status == 'L858R.exon19del' & snv_type %in% c("G > C", "A > G")) |
                       (`Smoking Status` == 'Current_Smoker' & EGFR_Status == "WT" &
                       snv_type == "C > A"))%>%
   ggplot(aes(EGFR_Status, snv_rate, fill = snv_type))+
   geom_bar(aes(size = tcd, col = tcd),
            stat = 'identity', color = 'black', position = "fill") +
   facet_grid(~ `Smoking Status`, scale = 'free', space = 'free_x') +
   scale_fill_manual(values = pal) +
  scale_size_manual(values = c(0.5, 1.75)) +
   theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1, size = 8))
```



3.1.3. Modifying our plot

First, we want both of our legends to be shown clearly.

```
snv_percentage %>%
 mutate(tcd = (`Smoking Status` == 'Nonsmoke' &
            EGFR_Status == 'L858R.exon19del' & snv_type %in% c("G > C", "A > G")) |
           (`Smoking Status` == 'Current_Smoker' & EGFR_Status == "WT" &
           snv_type == "C > A"))%>%
  ggplot(aes(EGFR_Status, snv_rate, fill = snv_type))+
  geom_bar(aes(size = tcd, col = tcd),
      stat = 'identity', color = 'black', position = "fill") +
  facet_grid(~ `Smoking Status`, scale = 'free', space = 'free_x') +
 theme(strip.text.x = element_text(size = 7.5))+
  scale_fill_manual(values = pal) +
 scale_size_manual(values = c(0.5, 1.75)) +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1, size = 8))+
 labs(size = 'top cancer driver', fill = 'snv type')+
 theme(legend.key.size = unit(0.45, 'cm'),
   legend.spacing.y = unit(0, "cm"),
       legend.title = element_text(size = 7),
       legend.text = element_text(size = 7))
```



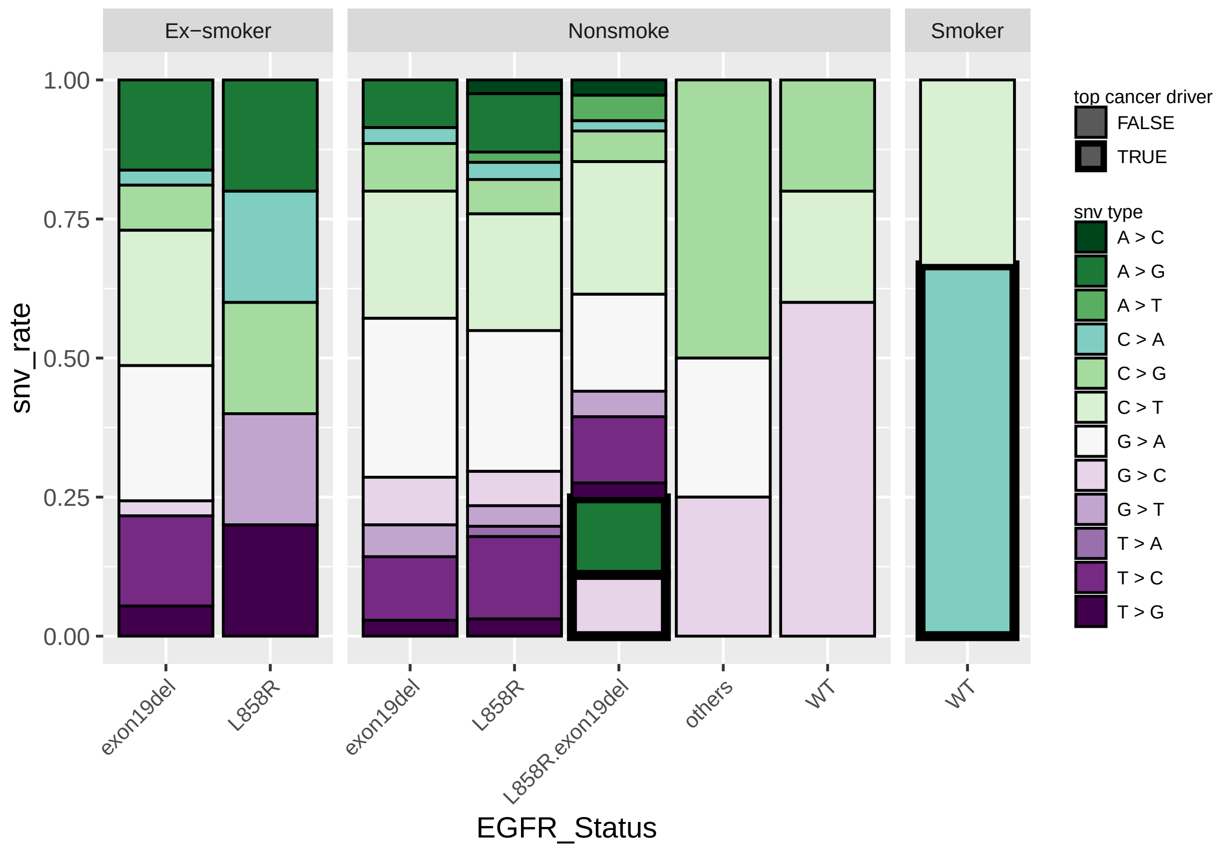
Finally, we can't see the full 'current_smoker' label, so we will change our 'Current_Smoker' status to 'Smoker'.

```
snv_percentage$`Smoking Status` <-
  gsub('Current_Smoker', 'Smoker',
       snv_percentage$`Smoking Status`)
```

Now let's produce our final plot!

```
plot_1 <- snv_percentage %>%
  mutate(tcd = (`Smoking Status` == 'Nonsmoke' &
                        EGFR_Status == 'L858R.exon19del' & snv_type %in% c("G > C", "A > G")) |
                       (`Smoking Status` == 'Smoker' & EGFR_Status == "WT" &
                       snv_type == "C > A"))%>%
   ggplot(aes(EGFR_Status, snv_rate, fill = snv_type))+
   geom_bar(aes(size = tcd, col = tcd),
            stat = 'identity', color = 'black', position = "fill") +
   facet_grid(~ `Smoking Status`, scale = 'free', space = 'free_x') +
  theme(strip.text.x = element_text(size = 7.5))+
   scale_fill_manual(values = pal) +
  scale_size_manual(values = c(0.5, 1.75)) +
   theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1, size = 8))+
  labs(size = 'top cancer driver', fill = 'snv type')+
  theme(legend.key.size = unit(0.45, 'cm'),
    legend.spacing.y = unit(0, "cm"),
        legend.title = element_text(size = 7),
        legend.text = element_text(size = 7))
```

```
plot_1
```



3.2. SNV type & Protein Expression Level

3.2.1. Creating Main Plot

Let’s create another snv_percentage data, this time based on tumor_expression.

```
snv_percentage <- merged_ds %>%
  group_by(EGFR_Status, tumor_expression) %>%
  count(snv_type) %>%
  summarize(snv_type = snv_type, snv_rate = n/sum(n))
```

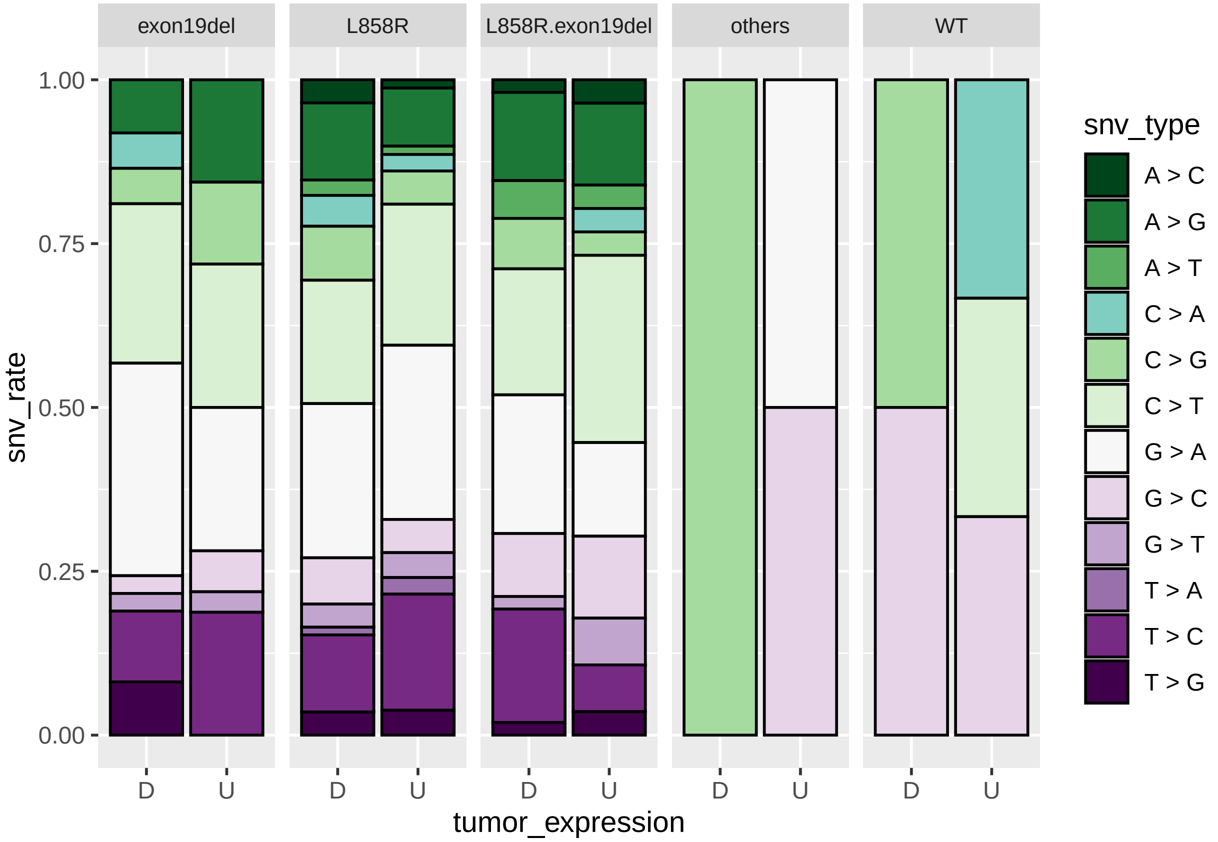
```
## ‘summarise()‘ has grouped output by ’EGFR_Status’, ’tumor_expression’. You can override using the ‘.g
```

```
snv_percentage
```

```
## # A tibble: 76 x 4
## # Groups:   EGFR_Status, tumor_expression [13]
##    EGFR_Status tumor_expression snv_type snv_rate
##    <chr>       <chr>            <chr>       <dbl>
##  1 exon19del   D                A > G      0.0811
##  2 exon19del   D                C > A      0.0541
##  3 exon19del   D                C > G      0.0541
##  4 exon19del   D                C > T      0.243
##  5 exon19del   D                G > A      0.324
##  6 exon19del   D                G > C      0.0270
##  7 exon19del   D                G > T      0.0270
##  8 exon19del   D                T > C      0.108
##  9 exon19del   D                T > G      0.0811
## 10 exon19del   U                A > G      0.156
## # ... with 66 more rows
```

Now let's visualize our data!

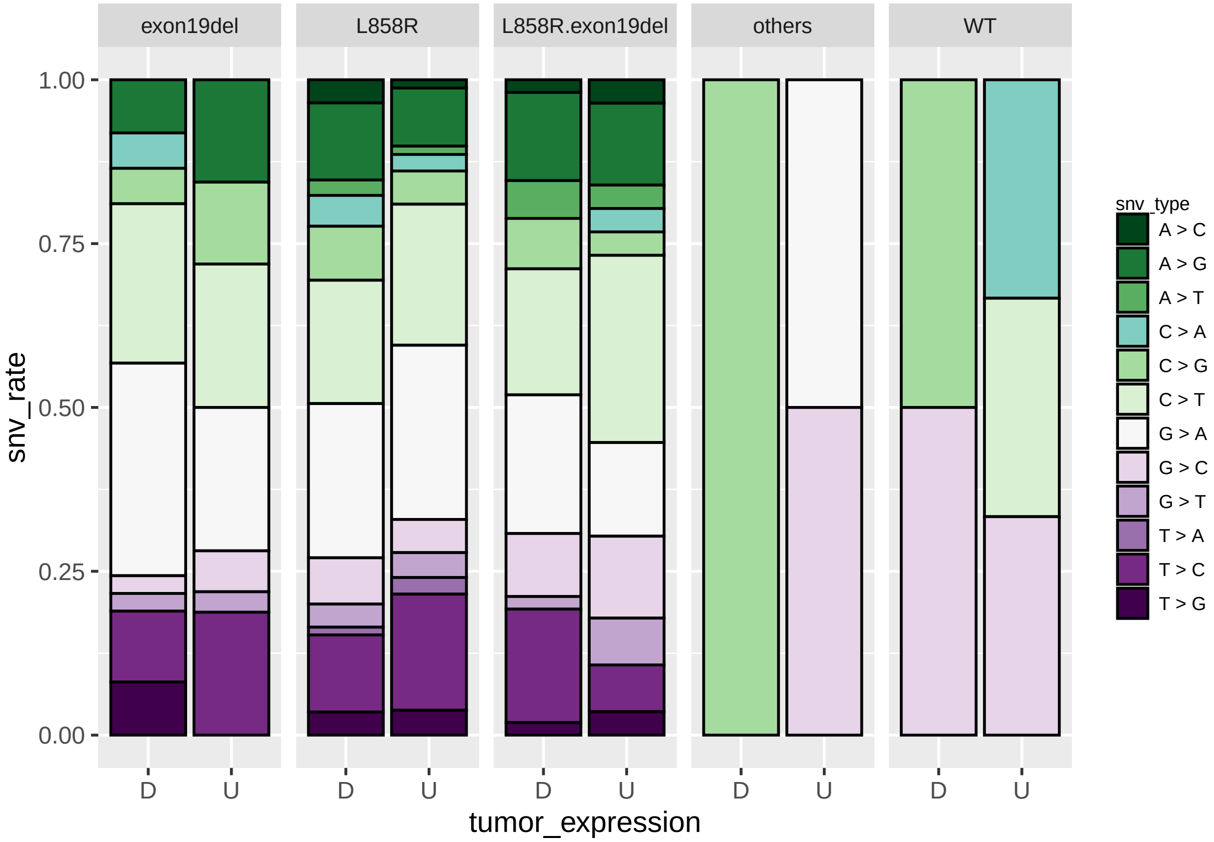
```
snv_percentage %>%
  filter(!is.na(tumor_expression)) %>%
  ggplot(aes(tumor_expression, snv_rate, fill = snv_type))+
  geom_bar(stat = 'identity', color = 'black')+
   facet_grid(~ EGFR_Status, scale = 'free', space = 'free_x') +
  theme(strip.text.x = element_text(size = 7.5))+
   scale_fill_manual(values = pal) +
  scale_size_manual(values = c(0.5, 1.75))
```



3.2.2. Modifying Data

```
#making legend visible

snv_percentage %>%
  filter(!is.na(tumor_expression)) %>%
  ggplot(aes(tumor_expression, snv_rate, fill = snv_type))+
  geom_bar(stat = 'identity', color = 'black')+
   facet_grid(~ EGFR_Status, scale = 'free', space = 'free_x') +
  theme(strip.text.x = element_text(size = 7.5))+
   scale_fill_manual(values = pal) +
  scale_size_manual(values = c(0.5, 1.75))+
theme(legend.key.size = unit(0.45, 'cm'),
    legend.spacing.y = unit(0, "cm"),
        legend.title = element_text(size = 7),
        legend.text = element_text(size = 7))
```



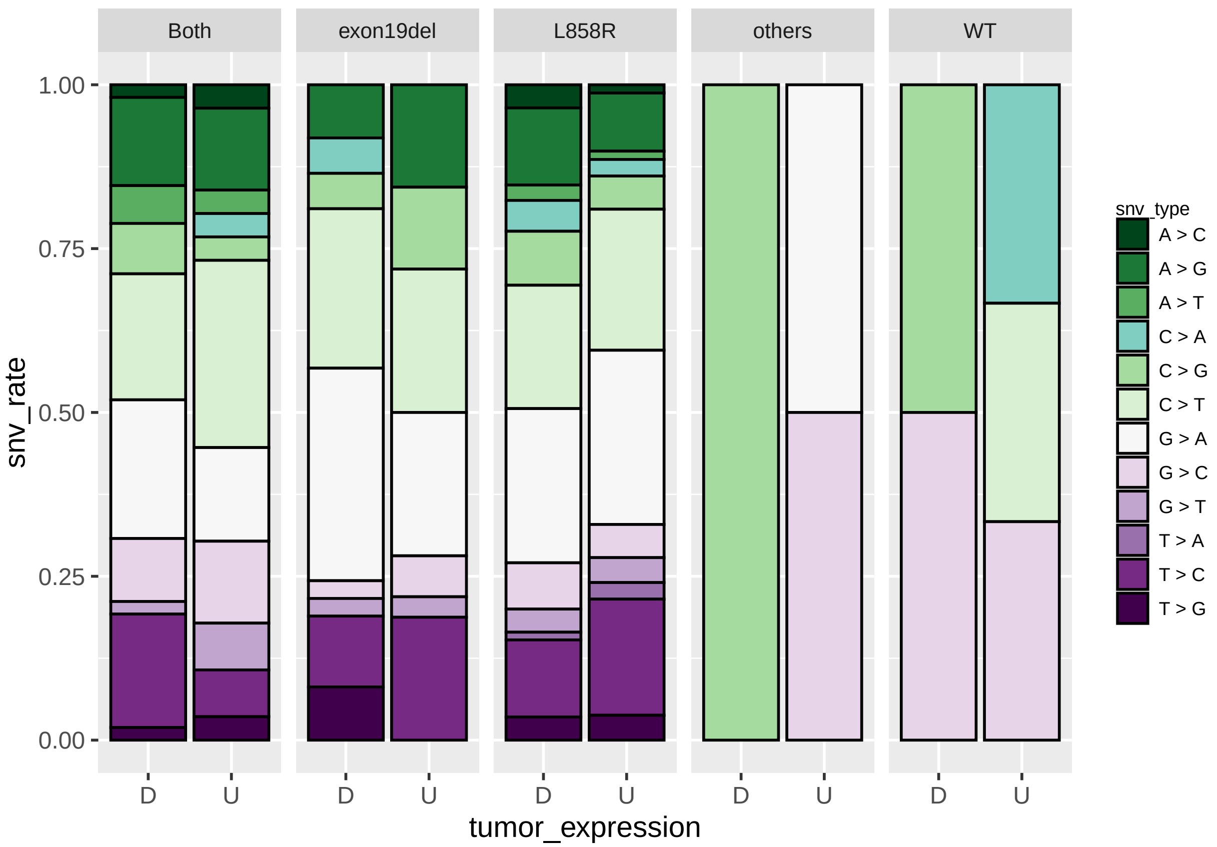
The EGFR status 'L858R.exon19del' is too long, so we will change it to 'both'.

```
snv_percentage$EGFR_Status <-
  gsub('L858R.exon19del', 'Both',
       snv_percentage$EGFR_Status)
```

```
plot_2 <- snv_percentage %>%
  filter(!is.na(tumor_expression)) %>%
  ggplot(aes(tumor_expression, snv_rate, fill = snv_type))+
  geom_bar(stat = 'identity', color = 'black')+
   facet_grid(~ EGFR_Status, scale = 'free', space = 'free_x') +
  theme(strip.text.x = element_text(size = 7.5))+
   scale_fill_manual(values = pal) +
  scale_size_manual(values = c(0.5, 1.75))+
theme(legend.key.size = unit(0.45, 'cm'),
    legend.spacing.y = unit(0, "cm"),
        legend.title = element_text(size = 7),
        legend.text = element_text(size = 7))
```

Now we have our second plot!

```
plot_2
```



4. Discussion

The research indicated that 'C > T' was the most common in this cohort, which is quite consistent with both our Plot_1 and Plot_2. It is especially common among 'non' and 'ex' smokers with exon 19 deletion and L858R mutation at EGFR. Another prevalent SNV type among 'non' and 'ex' smokers with known EGFR mutation type is 'G > A'. Interestingly, there aren't any 'C > T' or 'G > A' found in 'Ex-smokers' with L858R mutation, so this can be a matter of further research.

Among 'Smokers', 'C > A', which is known to be smoking - related, accounts for the majority.

All of the mutations from our 'top cancer driver genes' come from only 2 patients, so it might be difficult to derive meaningful results just from this plot. However, it can be used to support studies finding pathogenic functions of SNV types in lung cancer, especially in correlation with EGFR mutation.

For Plot_2, there isn't any significant difference in SNV type distribution between upregulated and downregulated proteins. However, this may be because of the lack of sample number. In the 'others' EGFR_status, the SNV types are completely different between downregulated and upregulated proteins, but this is because only one or two samples represent this data. It will not be appropriate to generalize this as a data that represents SNV type of proteins in lung cancer cells.

```
merged_ds %>% filter(EGFR_Status == 'others') %>% select(ID, Protein_name, snv_type, top_cancer_driver,
```

```
##     ID Protein_name snv_type top_cancer_driver log2TN tumor_expression
## 1 P002        H2AFY    C > G                 N -0.376                D
## 2 P002       NDUFS3    G > A                 N  0.123                U
## 3 P002        TOR3A    G > C                 N  0.185                U
## 4 P028          FRY    C > G                 N -0.426                D
```