Calculating Failure Rates for Handle and Swab Samples

SECTION 0: Required Packages and Files

The following packages must be installed and loaded into the R script prior to execution.

```
library(dplyr)
library(stringr)
library(DescTools)
library(openxlsx)
```

The datasets required to do the failure rate calculation can be downloaded from github (https://github.com/bbi-lab)

1. failure_detection_SCAN_SFS_dataset.csv
2. angry_swab_dataset.csv

SECTION 1: Creating the Swab Sample and Handle Sample Dataframes

Loading the Swab Dataset and the Handle Sample Dataset

Remember to set your working directary to the appropriate path.

Two datasets are loaded and read:

1. The raw dataset of all SCAN and SFS samples (both swab and handle)
   failure_detection_SCAN_SFS_dataset
2. The dataset of the samples where partipants used the handle side for nasal collection
   angry_swab_dataset

```
# load raw SCAN and SFS samples dataset
read_scan_sfs_ds = read.csv("failure_detection_SCAN_SFS_dataset.csv")

# load and read angry (handle) swab dataset
read_angry_swab_ds = read.csv("angry_swab_dataset.csv)
```

Creating Dataframes in Preparation of Data Wrangling/Manipulation

Once the two datasets are loaded and read, convert them into dyplr dataframes to allow conveniency towards data wrangling and manipulation.

```
# dataframe for the raw SCAN/SFS dataset
SCAN_SFS_df = tbl_df(read_scan_sfs_ds)

# dataframe for angry (handle) swab dataset
angry_swab_df = tbl_df(read_angry_swab_ds)

# delete unneeded data
rm(read_angry_swab_ds)
rm(read_scan_sfs_ds)
```

Modifying SCAN/SFS Samples Dataframe to Exclude Handle Samples

```
# only swab samples AKA excludes handle type
swab_failed_detection <- anti_join(SCAN_SFS_df, angry_swab_df, by = "investigator_sample_id")
swab_failed_detection <- select(swab_failed_detection, investigator_sample_id, target_status)

# filter out NA target status and duplicate sample id
swab_failed_detection <- filter(swab_failed_detection, !is.na(target_status) &
                                  !duplicated(investigator_sample_id))

# save results as csv
write.csv(swab_failed_detection, "swab_failed_detection.csv")
```

At the end of this section, three dataframes are created and saved in the Global Environment.

1. **SCAN__SFS__df**
2. **swab__failed__detection**
3. **angry__swab__df**

Please note that the swab_failed_detection includes all SFS/SCAN samples, EXCEPT the samples collected via handle.

With these these two dataframes created in this section, failure rates can now be calculated in the next section.

SECTION 2: Getting Handle Failed Detection Dataframe

Merging SFS/SCAN Dataframe with Handle Sample Dataframe

swab_failed_detection is already in the Global Environment. handle_failed_detection still needs to be created where it only contains samples collected via handle.

```
# merge SCAN_SFS_df with angry_swab_df
handle_failed_detection <-merge(x = SCAN_SFS_df ,y = angry_swab_df,
                                by ="investigator_sample_id", all.y = TRUE)

# select for needed columns
handle_failed_detection <- select(handle_failed_detection, investigator_sample_id, target_status)
```

```r
# filter out NA target status and duplicate sample id
handle_failed_detection <- filter(handle_failed_detection, !is.na(target_status) &
                                    !duplicated(investigator_sample_id))

# save results as csv
write.csv(handle_failed_detection, "handle_failed_detection.csv")

# remove unneeded dataframes
rm(angry_swab_df)
rm(SCAN_SFS_df)
```

At the end of this section, two dataframes are saved in the Global Environment:

1. **swab_failed_detection**
2. **handle_failed_detection**

These two dataframes are the finalized results on samples' Rnase P detection. The next section wil create a better visualization on how many handle versus swab samples are deemed as "failed" samples (no detection of Rnase P).

SECTION 3: Creating Contingency Table

With a large sample size seen in the swab_failed_detection dataframe, a contingency table is useful to get a count on detected versus not detected samples for each catagory (handle or swab).

```r
# add column to each dataframe, indicating type - handle/swab
handle_failed_detection$type <- 'handle'
swab_failed_detection$type <- 'swab'

# create dataframe where handle and swab samples are combined
handle_swab_failed <- rbind(handle_failed_detection, swab_failed_detection)

# create table based on type - handle/swab and target status - detected/not detected
handle_swab_table <- table(handle_swab_failed$type, handle_swab_failed$target_status)

# add sum of handle and swab sample to table
handle_swab_table_w_sum <- addmargins(table(handle_swab_failed$type,
                                      handle_swab_failed$target_status), 2)

# view table on console
handle_swab_table
```

The table with the sum should appear as:

	Detected	Not Detected	Sum
handle	99	3	102
swab	11914	192	12106

SECTION 4: Calculating Significant Difference between Failure Rates

With the contingency table created in the previous section, a Fisher's exact test can be used to determine if the handle and swab's failure rates are significantly different from each other.

```
failure_rate_p_value <- fisher.test(handle_swab_table)
failure_rate_p_value
```

The p-value is **0.22**, thus making the two failure rates insignificant from each other.