

Results Section: Antibiotic Resistance Patterns

```
library(staphopia)
library(ggplot2)
library(reshape2)
library(scales)
library(dplyr)

##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##   filter, lag
## The following objects are masked from 'package:base':
##   intersect, setdiff, setequal, union
library(gridExtra)

##
## Attaching package: 'gridExtra'
## The following object is masked from 'package:dplyr':
##   combine
USE_DEV = TRUE
produce_all_plots = FALSE
```

In this section we will look into resistance patterns *Staphylococcus aureus*.

Aggregating Data For Public Samples

First we'll get all publicly available *S. aureus* samples.

```
ps <- get_public_samples()
```

MRSA and MSSA

We defined MRSA by the presence of the *mecA*. Samples which did not have evidence for *mecA* were classified as MSSA.

Primer based classification

First we'll use the results from the primer based SCCmec classification to identify samples with full matches to the *mecA* primers. It is important to note these results will only identify SCCmec types containing *mecA* (Example: SCCmec Xi has *mecC* and will not be included in these)

Strict (Full Hits Only)

```
sccmec_primer <- get_sccmec_type(ps$sample_id)
table(sccmec_primer$meca)

##
## FALSE TRUE
## 16206 26743

sccmec_counts <- as.data.frame(colSums(sccmec_primer[,2:11]))
colnames(sccmec_counts) <- c('Total')
sccmec_counts <- data.frame(Type=rownames(sccmec_counts), Total=sccmec_counts$Total)
sccmec_counts

##      Type Total
## 1       I    41
## 2      II   5146
## 3     III   3212
## 4     IV  14016
## 5      V   4301
## 6     VI    28
## 7    VII  4301
## 8   VIII     1
## 9    IX    53
## 10 meca 26743
```

Relaxed (Hamming Distance)

```
sccmec_type_hd <- get_sccmec_type(ps$sample_id, hamming = TRUE)
table(sccmec_type_hd$meca)

##
##      0     1     2     3     4     5     7     8     9     10    11    12
## 26743  2     1     2     1     2     1    15   128    86     4     1
## 13    15    16    17    18    19    20    21    29    30    42
## 1     12    18   216   4370  10846   460     4     5    30     1
```

Protein Based Classification

```
sccmec_proteins <- get_sccmec_protein_hits(ps$sample_id)
max_score <- group_by(sccmec_proteins,target) %>%
  summarise(maxscore = max(bitscore))
sccmec_proteins <- merge(sccmec_proteins, max_score, by='target')

sccmec_proteins$BSR <- sccmec_proteins$bitscore / sccmec_proteins$maxscore
table(sccmec_proteins[sccmec_proteins$BSR >0.95,]$target)

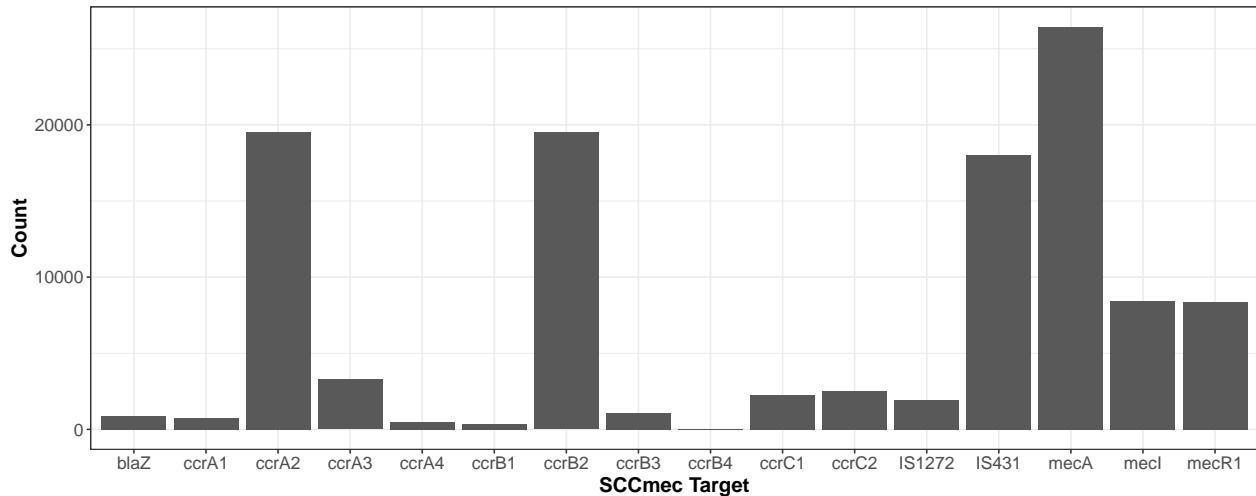
##
##      blaZ  ccrA1  ccrA2  ccrA3  ccrA4  ccrB1  ccrB2  ccrB3  ccrB4  ccrC1
##      867    773  19521   3279    504    357  19500   1084     31   2279
##      ccrC2 IS1272  IS431   mecA   mecI  mecR1
##      2534   1909  18021  26430   8434   8360
```

```

p <- ggplot(data=sccmec_proteins[sccmec_proteins$BSR > 0.95], aes(x=target)) +
  ylab("Count") +
  xlab("SCCmec Target") +
  geom_bar() +
  theme_bw() +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14,face="bold"),
        legend.title = element_blank())

```

p

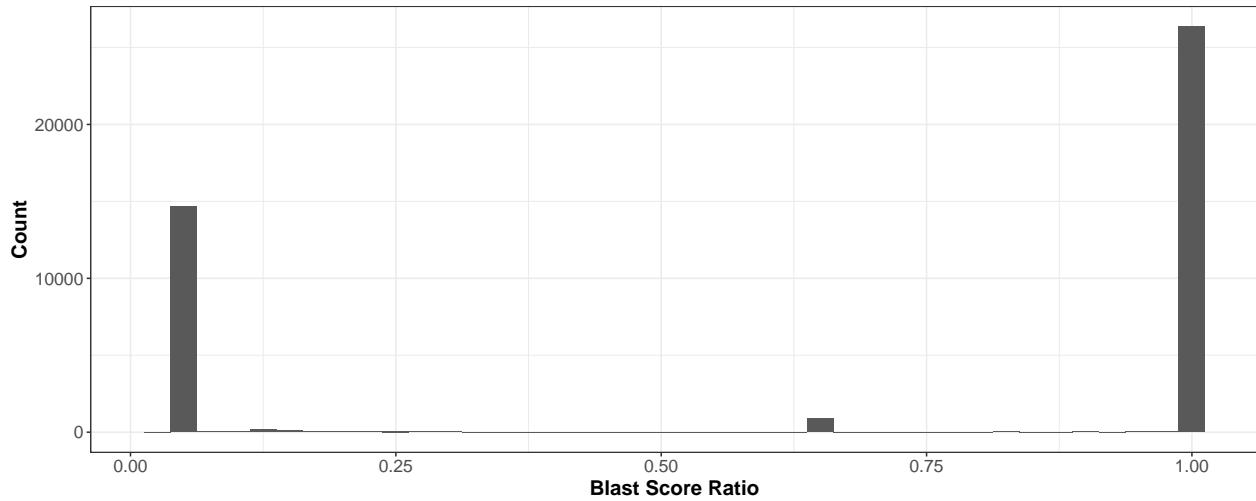


```

a <- sccmec_proteins[sccmec_proteins$target == 'mecA',]
sccmec_proteins_mec <- data.frame(sample_id=a$sample_id, BSR=a$BSR)
sccmec_proteins_mec$mec <- ifelse(sccmec_proteins_mec$BSR >= 0.95, TRUE, FALSE)
p <- ggplot(data=sccmec_proteins_mec, aes(x=BSR)) +
  xlab("Blast Score Ratio") +
  ylab("Count") +
  geom_histogram(binwidth = 0.025) +
  theme_bw() +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14,face="bold"),
        legend.title = element_blank())

```

p



```

a <- hist(sccmec_proteins_mec$BSR, plot=FALSE)
meca_bsr <- data.frame(
  region=sapply(1:length(a$counts), function(x){paste0(a$breaks[x], " - ", a$breaks[x+1])}),
  count=a$counts
)
meca_bsr

##           region count
## 1      0 - 0.05    21
## 2     0.05 - 0.1 14731
## 3     0.1 - 0.15   243
## 4     0.15 - 0.2   133
## 5     0.2 - 0.25   120
## 6     0.25 - 0.3    60
## 7     0.3 - 0.35   45
## 8     0.35 - 0.4    26
## 9     0.4 - 0.45   26
## 10    0.45 - 0.5    17
## 11    0.5 - 0.55   23
## 12    0.55 - 0.6    24
## 13    0.6 - 0.65   23
## 14    0.65 - 0.7  879
## 15    0.7 - 0.75   18
## 16    0.75 - 0.8    24
## 17    0.8 - 0.85   33
## 18    0.85 - 0.9    13
## 19    0.9 - 0.95   51
## 20    0.95 - 1 26430

```

Ariba based classification

We can use the results from Ariba (via MEGARes) to identify samples with predicted resistance to methicillin. We will do this in two ways, first by only looking at results with a match (strict), and the other being those results that include partial assemblies (relaxed). A reminder, the Ariba results only include samples with paired end reads.

Strict

These results are based on the a match to a SCCmec related cluster.

```
ariba <- get_sccmec_ariba(ps$sample_id, resistance_report = TRUE)

table(ariba$mec)

## 
## FALSE TRUE
## 15342 27120
```

Relaxed

These results allow for partial matches to a SCCmec related cluster.

```
ariba_relaxed <- get_sccmec_ariba(ps$sample_id, resistance_report = TRUE, include_all=TRUE)

table(ariba_relaxed$mec)

## 
## FALSE TRUE
## 14621 27841
```

SCCmec Cassette Coverage

```
sccmec_coverage <- get_sccmec_cassette_coverages(ps$sample_id)
```

Group By Most Covered SCCmec Type

```
top_type <- sccmec_coverage %>% group_by(sample_id) %>% slice(which.max(total))
table(top_type[top_type$total > 0.5,]$cassette)

## 
##    I   IIa   IIe   III   IVa   IVb   IVc   IVe   IVg   IVh   IVi   IVj   IVl   V   VI
##  688  5156    26  2801  5540    98   549   829   206   280   178  6586   250 1188   170
##    VII  VIII     X   XI
##    12   452    19   895
length(top_type[top_type$total > 0.5,]$cassette)

## [1] 25923
```

Group By Most Covered *mec* Region

```
top_mec <- sccmec_coverage %>% group_by(sample_id) %>% slice(which.max(meca_total))
table(top_mec[top_mec$total > 0.5,]$cassette)

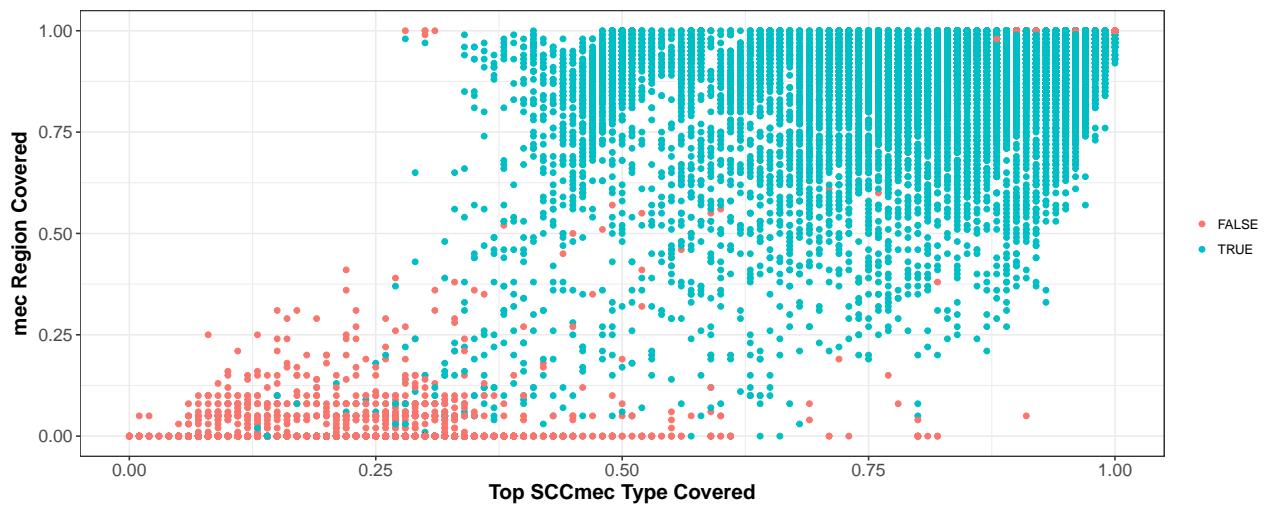
## 
##    I   IIa   IIe   III   IVa   IVb   IVc   IVe   IVg   IVh   IVi   IVj   IVl   IX   V
## 1642   468  2268  1221  2020  1942  1521  1643  1512  1198  1600  1193  1412 1329   844
##    VI   VII  VIII     X   XI
##   772   975 1300   169   894
```

Plot Of Top SCCmec Covered and *mec* Region Covered

mec Predicted By Primers

```
p <- ggplot(data=merge(top_type, sccmec_primer, by='sample_id'),
             aes(total, meca_total, colour = meca)) +
  ylab("mec Region Covered") +
  xlab("Top SCCmec Type Covered") +
  geom_point() +
  scale_fill_manual(values=c("#2ca25f", "#5ab4ac")) +
  theme_bw() +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14, face="bold"),
        legend.title = element_blank())
```

p

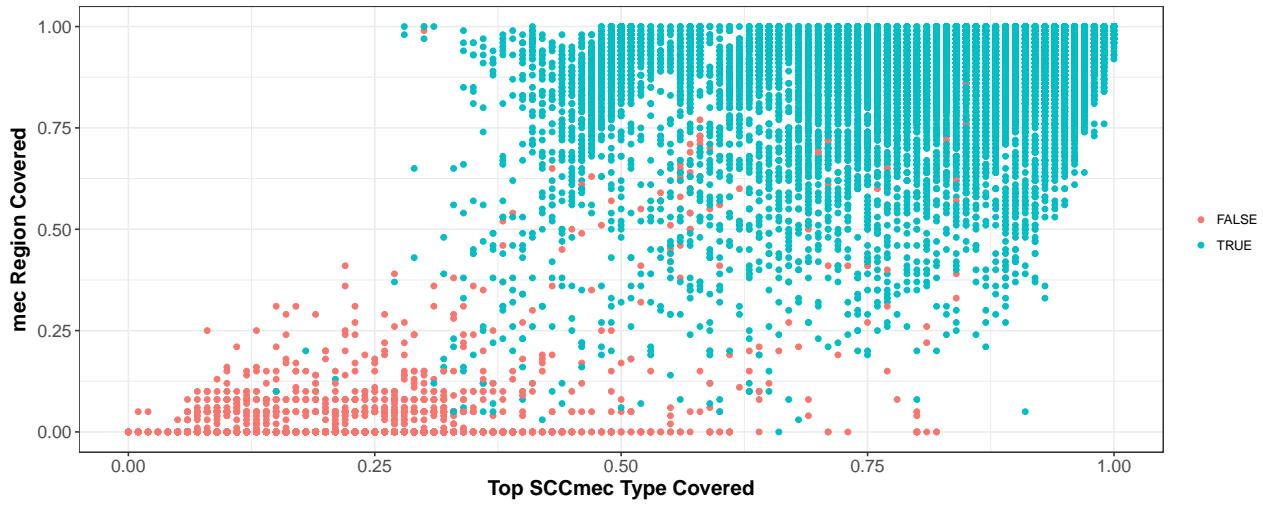


```
# Output plot to PDF and PNG
staphopia::write_plot(p, paste0(getwd(), '/images/figure-x-sccmec-coverage-primer'))
```

mec Predicted By Ariba (Strict)

```
p <- ggplot(data=merge(top_type, ariba, by='sample_id'),
             aes(total, meca_total, colour = meca)) +
  ylab("mec Region Covered") +
  xlab("Top SCCmec Type Covered") +
  geom_point() +
  scale_fill_manual(values=c("#2ca25f", "#5ab4ac")) +
  theme_bw() +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14, face="bold"),
        legend.title = element_blank())
```

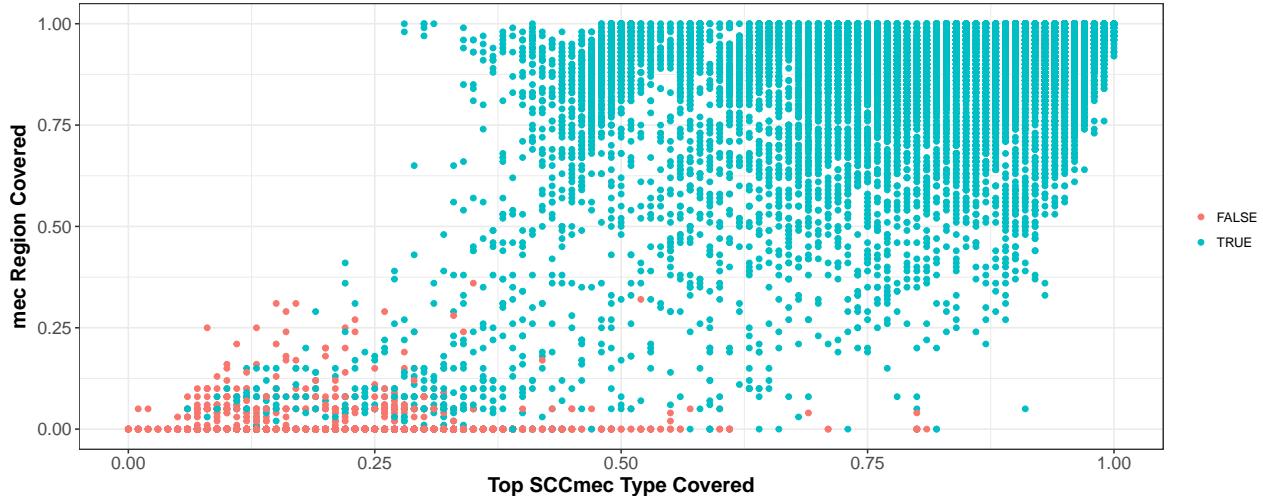
p



```
# Output plot to PDF and PNG
staphopia::write_plot(p, paste0(getwd(), '/images/figure-x-sccmec-coverage-ariba'))
```

mec Predicted By Ariba (Relaxed)

```
p <- ggplot(data=merge(top_type, ariba_relaxed, by='sample_id'),
             aes(total, meca_total, colour = mec)) +
  ylab("mec Region Covered") +
  xlab("Top SCCmec Type Covered") +
  geom_point() +
  scale_fill_manual(values=c("#2ca25f", "#5ab4ac")) +
  theme_bw() +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14, face="bold"),
        legend.title = element_blank())
p
```



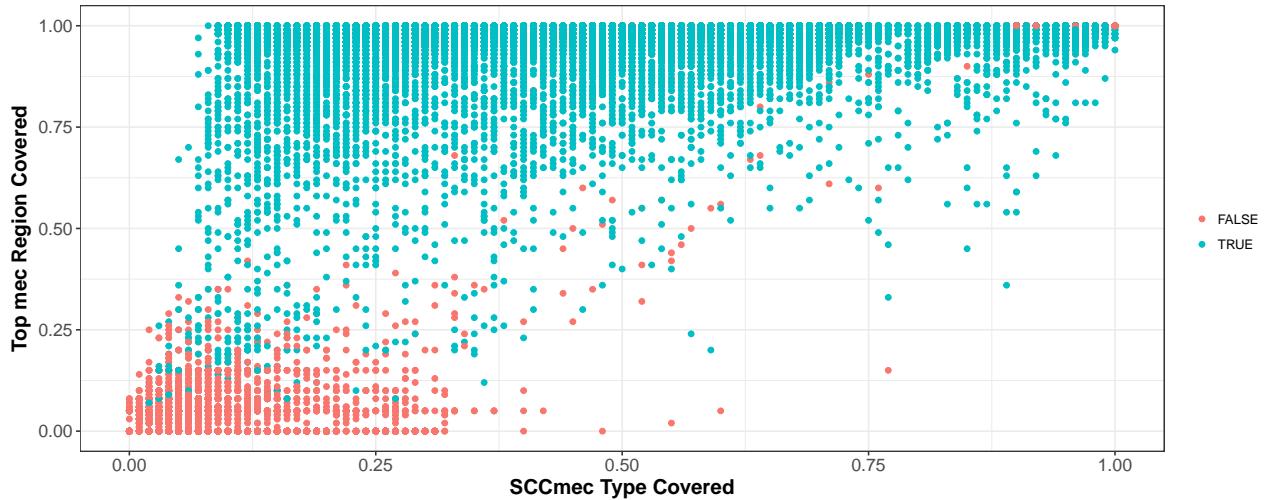
```
# Output plot to PDF and PNG
staphopia::write_plot(p, paste0(getwd(), '/images/figure-x-sccmec-coverage-ariba-relaxed'))
```

Plot Of SCCmec Covered and Top *mec* Region Covered

mec Predicted By Primers

```
p <- ggplot(data=merge(top_mec, sccmec_primer, by='sample_id'),
             aes(total, meca_total, colour = meca)) +
  ylab("Top mec Region Covered") +
  xlab("SCCmec Type Covered") +
  geom_point() +
  scale_fill_manual(values=c("#2ca25f", "#5ab4ac")) +
  theme_bw() +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14, face="bold"),
        legend.title = element_blank())
```

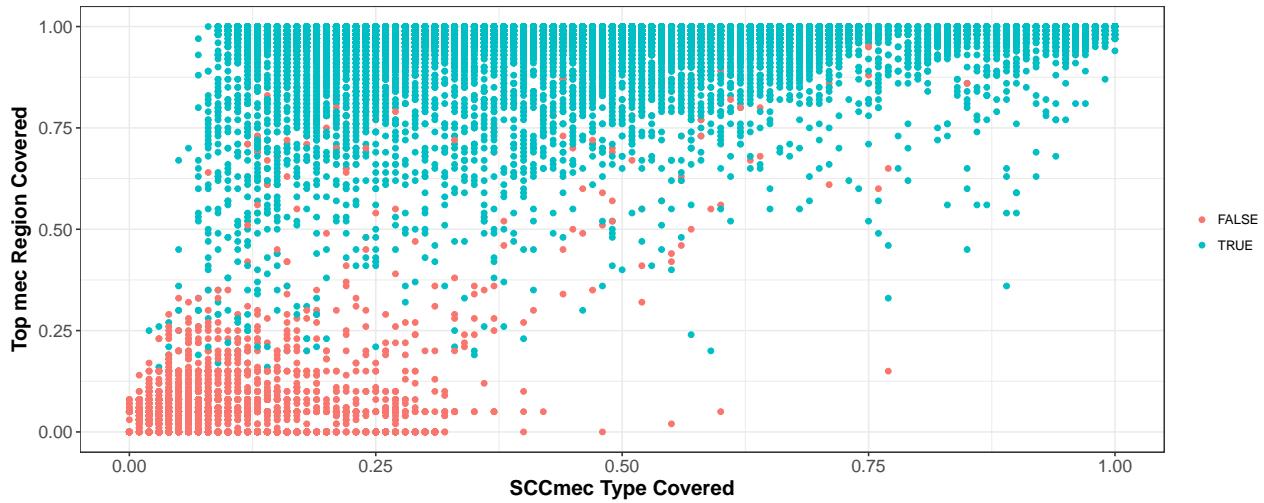
p



mec Predicted By Ariba (Strict)

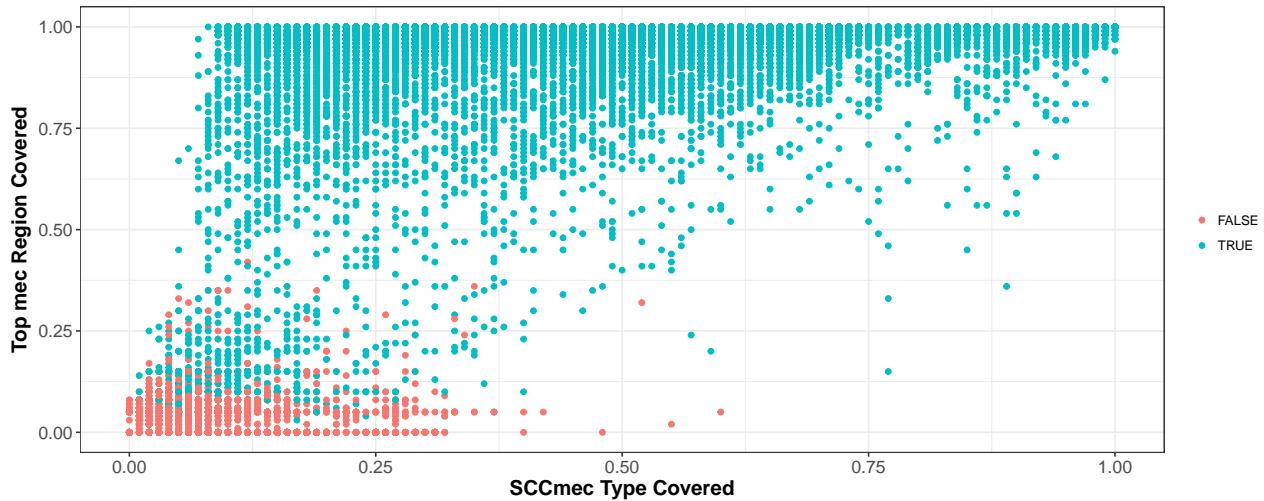
```
p <- ggplot(data=merge(top_mec, ariba, by='sample_id'),
             aes(total, meca_total, colour = mec)) +
  ylab("Top mec Region Covered") +
  xlab("SCCmec Type Covered") +
  geom_point() +
  scale_fill_manual(values=c("#2ca25f", "#5ab4ac")) +
  theme_bw() +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14, face="bold"),
        legend.title = element_blank())
```

p



mec Predicted By Ariba (Relaxed)

```
p <- ggplot(data=merge(top_mec, ariba_relaxed, by='sample_id'),
             aes(total, meca_total, colour = mec)) +
  ylab("Top mec Region Covered") +
  xlab("SCCmec Type Covered") +
  geom_point() +
  scale_fill_manual(values=c("#2ca25f", "#5ab4ac")) +
  theme_bw() +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14, face="bold"),
        legend.title = element_blank())
p
```



Plot of Each SCCmec Type Individually

Function For Plotting

```
plot_by_sccmectype <- function(coverage, column) {
  p <- ggplot(data=coverage,
```

```

        aes(x=total, y=meca_total, colour = mec)) +
    ylab("mec Region Covered") +
    xlab(paste0("SCCmec Type ", column," Covered")) +
    geom_point() +
    scale_fill_manual(values=c("#2ca25f", "#5ab4ac")) +
    theme_bw() +
    theme(axis.text=element_text(size=12),
          axis.title=element_text(size=14,face="bold"),
          legend.title = element_blank())
  return(p)
}

```

Each SCCmec Type Individually (Colored By Ariba (Strict))

```

if (produce_all_plots) {
  for (column in unique(sccmec_coverage$cassette)) {
    print(plot_by_sccmectype(
      merge(sccmec_coverage[sccmec_coverage$cassette == column,],
            ariba,
            by='sample_id'),
      column
    )))
  }
}

```

Each SCCmec Type Individually (Colored By Ariba (Relaxed))

```

if (produce_all_plots) {
  for (column in unique(sccmec_coverage$cassette)) {
    print(plot_by_sccmectype(
      merge(sccmec_coverage[sccmec_coverage$cassette == column,],
            ariba_relaxed,
            by='sample_id'),
      column
    )))
  }
}

```

Compare *mec* Predictions

```

mec <- merge(
  ps,
  data.frame(
    sample_id=sccmec_proteins_mec$sample_id,
    protein_mec=sccmec_proteins_mec$mec
  ),
  by='sample_id',
  all=TRUE
)
mec[is.na(mec$protein_mec),]$protein_mec <- FALSE
mec <- merge(mec, data.frame(sample_id=sccmec_primer$sample_id, primer_mec=sccmec_primer$meca), by='sample_id')
mec <- merge(mec, data.frame(sample_id=ariba$sample_id, ariba_mec=ariba$mec), by='sample_id', all=TRUE)

```

```

mec <- merge(mec, data.frame(sample_id=ariba_relaxed$sample_id, ariba_relaxed_mec=ariba_relaxed$mec), by='sample_id')
mec$agreement <- paste0(
  ifelse(mec$protein_mec, 1, 0),
  ifelse(mec$primer_mec, 1, 0),
  ifelse(is.na(mec$ariba_mec), '-', ifelse(mec$ariba_mec, 1, 0)),
  ifelse(is.na(mec$ariba_relaxed_mec), '-', ifelse(mec$ariba_relaxed_mec, 1, 0)))
)
table(mec$agreement)

## 
##   00--    0000    0001    0011    01--    0100    0101    0111    1011    11--    1101    1111
##   211  14614     496    875      12       7    145    159      10    264      80  26076

```

Notes

- 0: **mec** not predicted
- 1: **mec** predicted
- -: Not tested by Ariba (Single-End reads)

The order of numbering is:

- 1: **mecA** with BSR > 0.95 based on Proteins
- 2: **mecA** with full Primer hit
- 3: **mec** with full match based on Ariba
- 4: **mec** with partial match based on Ariba

Example:

- 00- : Single-End, Protein and Primer are False
- 0000 : All approaches agree that **mec** is not predicted
- 1111 : All approaches agree that **mec** is predicted

```

mec <- merge(mec, data.frame(sample_id=top_type$sample_id, total=top_type$total), by='sample_id')
table(mec[mec$total >= 0.5]$primer_mec)

## 
## FALSE  TRUE
##   965 25497
table(mec[mec$total >= 0.5]$protein_mec)

## 
## FALSE  TRUE
##   1138 25324
table(mec[mec$total >= 0.5]$ariba_mec)

## 
## FALSE  TRUE
##   198 25987
table(mec[mec$total >= 0.5]$ariba_relaxed_mec)

## 
## FALSE  TRUE
##    51 26134

```

mecA Presence By ST

```

meca_groups <- merge(data.frame(sample_id=ps$sample_id, st=ps$st, rank=ps$rank),
                      ariba,
                      by='sample_id')
meca_groups$status <- ifelse(meca_groups$mec == TRUE, 'MRSA', 'MSSA')
meca_by_st <- plyr::count(meca_groups, c('st', 'status'))
head(meca_by_st)

##   st status freq
## 1  0    MRSA  225
## 2  0    MSSA  372
## 3  1    MRSA  244
## 4  1    MSSA  549
## 5  3    MSSA   29
## 6  5    MRSA 3323

```

Get the Top 10 STs

```

top_st <- get_top_sequence_types()
top_st

```

```

##      st count percent overall
## 1    22  7189   16.74  16.74
## 2     8  6184   14.40  31.14
## 3     5  4664   10.86  42.00
## 4   239  3123    7.27  49.27
## 5   398  2326    5.42  54.68
## 6    30  1872    4.36  59.04
## 7    45  1663    3.87  62.91
## 8    15  1172    2.73  65.64
## 9    36   857    2.00  67.64
## 10   105   857    2.00  69.63

```

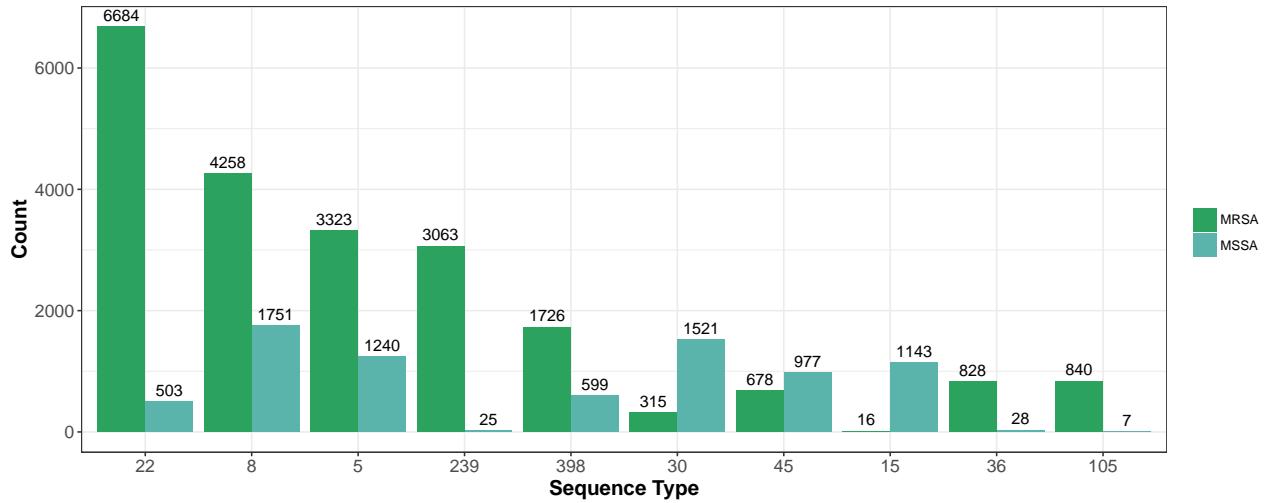
MRSA/MSSA Distribution For Top 10 Sequence Types

Now we are ready to plot out the distribution of MRSA and MSSA predictions by sequence type.

```

top_st_meca <- merge(top_st, meca_by_st, by='st')
p <- ggplot(data=top_st_meca, aes(x=reorder(st, -count), y=freq, fill = status)) +
  xlab("Sequence Type") +
  ylab("Count") +
  geom_bar(stat="identity", position = "dodge") +
  geom_text(aes(label=freq), vjust = -0.5, position = position_dodge(.9)) +
  scale_fill_manual(values=c("#2ca25f", "#5ab4ac")) +
  theme_bw() +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14, face="bold"),
        legend.title = element_blank())
p

```



```
# Output plot to PDF and PNG
staphopia::write_plot(p, paste0(getwd(), '/images/figure-x-mrsa-top-10-sequence-types'))
```

Resistance Patterns Based on Ariba and MegaRes

Resistance Classes With A Match

```
resistance_report <- get_resistance_results(ps$sample_id, resistance_report=TRUE)
df <- as.data.frame(colSums(resistance_report[,2:ncol(resistance_report)]))
colnames(df) <- c('total')
resistance_report_counts <- data.frame(class=rownames(df), total=df$total)
resistance_report_counts[resistance_report_counts$total > 0,]

##          class total
## 1      Aminocoumarins    46
## 2      Aminoglycosides 17968
## 4      Betalactams    37758
## 5      Betalactams (mec) 27120
## 8      Fluoroquinolones    69
## 9      Fosfomycin    24205
## 10     Fusidic Acid    346
## 11     Glycopeptides   5777
## 12     Lipopeptides    44
## 13      MLS    22322
## 14 Multi-Drug Resistance 13653
## 16      Phenicol    852
## 17      Rifampin    46
## 18      Sulfonamides    36
## 19      Tetracyclines  8638
## 21      Trimethoprim   6605
```

Resistance Classes Without A Match

```
data.frame(class=as.character(resistance_report_counts[resistance_report_counts$total == 0,]$class))

##          class
## 1      Bacitracin
## 2 Cationic Antimicrobial Peptides
```

```

## 3                      Elfamycins
## 4 Mycobacterium Tuberculosis-Specific Drug
## 5                      Thiomycine
## 6                      Tunicamycin

cluster_report <- get_resistance_results(ps$sample_id, cluster_report=TRUE)
df <- as.data.frame(colSums(cluster_report[,2:ncol(cluster_report)]))
colnames(df) <- c('total')
cluster_report_counts <- data.frame(cluster=rownames(df), total=df$total)
cluster_report_counts[cluster_report_counts$total > 0,]

```

	cluster	total
## 4	AAC6_PRIME_14;Aminoglycosides	6660
## 5	AAC6_PRIME_15;Aminoglycosides	2
## 9	AAC6_PRIME_31;Aminoglycosides	10
## 23	ADEAI;Multi-drug resistance	1
## 26	ADEC;Multi-drug resistance	10
## 38	ANT3_DPRIME_10;Aminoglycosides	9850
## 42	ANT4_PRIME_1;Aminoglycosides	7302
## 43	ANT6;Aminoglycosides	978
## 44	ANT6_2;Aminoglycosides	79
## 46	ANT6_6;Aminoglycosides	620
## 47	ANT9_1;Aminoglycosides	76
## 48	APH3_-;Aminoglycosides	1
## 51	APH3_PRIME_7;Aminoglycosides	8363
## 66	BLAZ;betalactams	868
## 67	BLAZ_1;betalactams	33652
## 70	BRP;Glycopeptides	5773
## 72	CAT;Phenicol	225
## 74	CAT_-1;Phenicol	392
## 76	CATA_2;Phenicol	124
## 80	CFRA;Phenicol	6
## 81	CLS_1;Lipopeptides	10
## 94	DFRC;Trimethoprim	1851
## 95	DFRD;Trimethoprim	8
## 96	DFRF;Trimethoprim	10
## 97	DFRG;Trimethoprim	4301
## 98	DFRK;Trimethoprim	510
## 107	EMEA;Multi-drug resistance	34
## 111	EMRE_1;Multi-drug resistance	1
## 115	ERM_-1;MLS	317
## 116	ERM_-2;MLS	454
## 118	ERM_7;MLS	1
## 119	ERMA;MLS	10133
## 120	ERMA_1;MLS	22
## 121	ERMA_2;MLS	3
## 122	ERMC;MLS	7944
## 129	FEXA;Phenicol	144
## 137	FOLP_2;Sulfonamides	2
## 139	FOLP_23;Sulfonamides	34
## 143	FOSA;Fosfomycin	17
## 146	FOSB;Fosfomycin	24172
## 147	FOSB_1;Fosfomycin	54
## 148	FUSB;Fusidic acid	346
## 160	GYRA_25;Fluoroquinolones	34

```

## 161      GYRA_28;Fluoroquinolones    2
## 166      GYRA_6;Fluoroquinolones    23
## 174      LIAFSR;Lipopeptides       10
## 175      LIAFSR_1;Lipopeptides     34
## 178      LNUA;MLS                 240
## 179      LNUB;MLS                 190
## 181      LSA;MLS                 34
## 182      LSA_1;MLS                187
## 183      LSA_3;MLS                1
## 206      MEC-;betalactams        26175
## 207      MECA;betalactams        8056
## 208      MECI;betalactams        4460
## 210      MEPB;Multi-drug resistance 10839
## 221      MPHc;MLS                5178
## 225      MSRA_1;MLS               5167
## 226      MSRC;MLS                10
## 234      NORA_1;Fluoroquinolones  23
## 243      OXA_14;betalactams      1
## 256      PARC_12;Fluoroquinolones 2
## 257      PARC_13;Fluoroquinolones 34
## 258      PARC_14;Fluoroquinolones 10
## 266      PARE;Aminocoumarins    10
## 267      PARE_1;Aminocoumarins   34
## 275      PARE_9;Aminocoumarins   2
## 276      PBP1A;betalactams       2
## 277      PBP1B;betalactams       2
## 278      PBP2;betalactams        878
## 281      PBP2X;betalactams       2
## 292      QACAB;Multi-drug resistance 3482
## 308      RPOB_4;Rifampin         2
## 309      RPOB_5;Rifampin         10
## 310      RPOB_6;Rifampin         34
## 314      SAT_1;Aminoglycosides   6260
## 321      SPD;Aminoglycosides    165
## 324      TEM;betalactams        51
## 333      TETK;Tetracyclines     5739
## 335      TETL_1;Tetracyclines    484
## 337      TETM;Tetracyclines     2213
## 338      TETM_1;Tetracyclines    3638
## 339      TETO;Tetracyclines     8
## 343      TETS;Tetracyclines     5
## 344      TETT;Tetracyclines     65
## 358      VANA_5;Glycopeptides    1
## 362      VANHA_4;Glycopeptides   1
## 363      VANRA;Glycopeptides    1
## 374      VANXA_1;Glycopeptides   3
## 379      VANYA;Glycopeptides    3
## 381      VANZA;Glycopeptides    2
## 382      VATA;MLS                2
## 383      VATB;MLS                3
## 384      VGA;MLS                 175
## 385      VGA_1;MLS                77
## 386      VGAB;MLS                3
## 387      VGBA;MLS                2

```

Resistance Clusters Without A Match

```
data.frame(class=as.character(cluster_report_counts[cluster_report_counts$total == 0,]$cluster))

##                                     class
## 1          AAC2_PRIME_3;Aminoglycosides
## 2          AAC3_12;Aminoglycosides
## 3          AAC3_3;Aminoglycosides
## 4          AAC6_PRIME_16;Aminoglycosides
## 5          AAC6_PRIME_29;Aminoglycosides
## 6          AAC6_PRIME_30;Aminoglycosides
## 7          AAC6_PRIME_8;Aminoglycosides
## 8          ABEM;Multi-drug resistance
## 9          ABES;Multi-drug resistance
## 10         ACRB;Multi-drug resistance
## 11         ACRD;Aminoglycosides
## 12         ACRE;Aminoglycosides
## 13         ACRF;Aminoglycosides
## 14         ACRR;Multi-drug resistance
## 15         ACRR_1;Multi-drug resistance
## 16         ACRS;Aminoglycosides
## 17         ACT;betalactams
## 18         ADC;betalactams
## 19         ADEA;Multi-drug resistance
## 20         ADEB;Multi-drug resistance
## 21         ADEB_1;Multi-drug resistance
## 22         ADEF+;Multi-drug resistance
## 23         ADEG;Multi-drug resistance
## 24         ADEH;Multi-drug resistance
## 25         ADEK;Multi-drug resistance
## 26         ADEN;Multi-drug resistance
## 27         ADER;Multi-drug resistance
## 28         ADES;Multi-drug resistance
## 29         AMPH;betalactams
## 30         AMPH_1;betalactams
## 31         ANT2_DPRIME;Aminoglycosides
## 32         ANT3_DPRIME;Aminoglycosides
## 33         ANT3_DPRIME_6;Aminoglycosides
## 34         ANT3_DPRIME_7;Aminoglycosides
## 35         ANT3_DPRIME_8;Aminoglycosides
## 36         ANT6_5;Aminoglycosides
## 37         APH3_DPRIME_1;Aminoglycosides
## 38         APH3_PRIME_11;Aminoglycosides
## 39         APH4_1;Aminoglycosides
## 40         APH6_4;Aminoglycosides
## 41         ARMA;Aminoglycosides
## 42         ARNA_1;Cationic antimicrobial peptides
## 43         ARR_6;Rifampin
## 44         ASMA;Multi-drug resistance
## 45         ASMA_1;Multi-drug resistance
## 46         BACA;Bacitracin
## 47         BAER;Multi-drug resistance
## 48         BAES;Multi-drug resistance
## 49         BLAA;betalactams
## 50         BLAA_1;betalactams
```

```

## 51           BLAA_2;betalactams
## 52           BLAEC;betalactams
## 53           BLT;Multi-drug resistance
## 54           BMR;Multi-drug resistance
## 55           BRP_1;Glycopeptides
## 56           CAT-;Phenicol
## 57           CATA_1;Phenicol
## 58           CATA_3;Phenicol
## 59           CATB;Phenicol
## 60           CEOB;Multi-drug resistance
## 61           cme;Multi-drug resistance
## 62           cme_2;Multi-drug resistance
## 63           cme_3;Multi-drug resistance
## 64           CMER+;Multi-drug resistance
## 65           CML;Phenicol
## 66           CMXAB;Phenicol
## 67           CPXA-;Multi-drug resistance
## 68           CPXA_-1;Multi-drug resistance
## 69           CRP;Multi-drug resistance
## 70           CTX_1;betalactams
## 71           CTX_5;betalactams
## 72           CTX_7;betalactams
## 73           DHFR+;Trimethoprim
## 74           DHFR_1;Trimethoprim
## 75           DHFR_6;Trimethoprim
## 76           DHFR_7;Trimethoprim
## 77   EMBA;Mycobacterium tuberculosis-specific Drug
## 78   EMBB+;Mycobacterium tuberculosis-specific Drug
## 79   EMBC;Mycobacterium tuberculosis-specific Drug
## 80   EMBR;Mycobacterium tuberculosis-specific Drug
## 81           EMRA;Multi-drug resistance
## 82           EMRB;Multi-drug resistance
## 83           EMRD;Multi-drug resistance
## 84           EMRK;Multi-drug resistance
## 85           EMRR;Multi-drug resistance
## 86           EMRY;Multi-drug resistance
## 87           ERM37;MLS
## 88           ERMD;MLS
## 89           ERMT;MLS
## 90           ERMX;MLS
## 91   ETHA;Mycobacterium tuberculosis-specific Drug
## 92           EVGA;Multi-drug resistance
## 93           EVGS;Multi-drug resistance
## 94           FLOR;Phenicol
## 95           FOLP_1;Sulfonamides
## 96           FOLP_12;Sulfonamides
## 97           FOLP_13;Sulfonamides
## 98           FOLP_14;Sulfonamides
## 99           FOLP_16;Sulfonamides
## 100          FOLP_17;Sulfonamides
## 101          FOLP_22;Sulfonamides
## 102          FOLP_3;Sulfonamides
## 103          FOLP_5;Sulfonamides
## 104          FOLP_8;Sulfonamides

```

```

## 105 FOSA_1;Fosfomycin
## 106 FOSA_8;Fosfomycin
## 107 GADX;Multi-drug resistance
## 108 GIDB;Aminoglycosides
## 109 GOLS;Multi-drug resistance
## 110 GYRA_11;Fluoroquinolones
## 111 GYRA_12;Fluoroquinolones
## 112 GYRA_14;Fluoroquinolones
## 113 GYRA_15;Fluoroquinolones
## 114 GYRA_17;Fluoroquinolones
## 115 GYRA_2;Fluoroquinolones
## 116 GYRA_20;Fluoroquinolones
## 117 GYRA_24;Fluoroquinolones
## 118 GYRA_29;Fluoroquinolones
## 119 GYRA_3;Fluoroquinolones
## 120 GYRA_31;Fluoroquinolones
## 121 GYRA_4;Fluoroquinolones
## 122 GYRB_1;Fluoroquinolones
## 123 HNS;Multi-drug resistance
## 124 INHA;Mycobacterium tuberculosis-specific Drug
## 125 INIA;Mycobacterium tuberculosis-specific Drug
## 126 INIC;Mycobacterium tuberculosis-specific Drug
## 127 KASA;Mycobacterium tuberculosis-specific Drug
## 128 KPC;betalactams
## 129 LIAFSR_2;Lipopeptides
## 130 LMRA+;Multi-drug resistance
## 131 LNUC;MLS
## 132 MACA;Multi-drug resistance
## 133 MACB;MLS
## 134 MARA;Multi-drug resistance
## 135 MARR;Multi-drug resistance
## 136 MDFA;Multi-drug resistance
## 137 MDSA;Multi-drug resistance
## 138 MDSB;Multi-drug resistance
## 139 MDSC;Multi-drug resistance
## 140 MDTA;Multi-drug resistance
## 141 MDTB;Multi-drug resistance
## 142 MDTC;Multi-drug resistance
## 143 MDTC_1;Multi-drug resistance
## 144 MDTC_2;Multi-drug resistance
## 145 MDTD;Multi-drug resistance
## 146 MDTE;Multi-drug resistance
## 147 MDTG;Multi-drug resistance
## 148 MDTH;Multi-drug resistance
## 149 MDTK;Multi-drug resistance
## 150 MDTL;Multi-drug resistance
## 151 MDTN;Multi-drug resistance
## 152 MDTO;Multi-drug resistance
## 153 MDTP;Multi-drug resistance
## 154 MEFA;MLS
## 155 MEXE_1;Multi-drug resistance
## 156 MEXF;Multi-drug resistance
## 157 MEXK;Multi-drug resistance
## 158 MEXP;Multi-drug resistance

```

```

## 159 MEXT;Multi-drug resistance
## 160 MEXV;Multi-drug resistance
## 161 MFPA;Fluoroquinolones
## 162 MIR+;betalactams
## 163 MPHA;MLS
## 164 MPHB;MLS
## 165 MPHE_1;MLS
## 166 MPRF+;Multi-drug resistance
## 167 MSR;Multi-drug resistance
## 168 MTRA;Multi-drug resistance
## 169 MTRC;Multi-drug resistance
## 170 MTRD;Multi-drug resistance
## 171 MTRE;Multi-drug resistance
## 172 MTRR;Multi-drug resistance
## 173 NDH;Mycobacterium tuberculosis-specific Drug
## 174 NDM;betalactams
## 175 OMP36+;betalactams
## 176 OMPD;betalactams
## 177 OMPF_1;Multi-drug resistance
## 178 OMPF_2;Multi-drug resistance
## 179 OMPF_3;Multi-drug resistance
## 180 OQXA;Multi-drug resistance
## 181 OQXB;Multi-drug resistance
## 182 OXA;betalactams
## 183 OXA_17;betalactams
## 184 OXA_19;betalactams
## 185 OXA_24;betalactams
## 186 OXA_3;betalactams
## 187 OXA_34;betalactams
## 188 OXA_4;betalactams
## 189 OXA_41;betalactams
## 190 OXA_5;betalactams
## 191 OXA_6;betalactams
## 192 OXA_7;betalactams
## 193 OXY_1;betalactams
## 194 PARC_1;Fluoroquinolones
## 195 PARC_2;Fluoroquinolones
## 196 PARC_3;Fluoroquinolones
## 197 PARC_4;Fluoroquinolones
## 198 PARC_5;Fluoroquinolones
## 199 PARC_6;Fluoroquinolones
## 200 PARC_8;Fluoroquinolones
## 201 PARC_9;Fluoroquinolones
## 202 PARE+_1;Aminocoumarins
## 203 PARE_10;Aminocoumarins
## 204 PARE_11;Aminocoumarins
## 205 PARE_12;Aminocoumarins
## 206 PARE_13;Aminocoumarins
## 207 PARE_2;Aminocoumarins
## 208 PARE+_2;Fluoroquinolones
## 209 PBP2_1;betalactams
## 210 PBP2B;betalactams
## 211 PBP4B;betalactams
## 212 PHOP+;Multi-drug resistance

```

```

## 213          PHOQ;Multi-drug resistance
## 214          pmrA;Cationic antimicrobial peptides
## 215          PMRB_1;Cationic antimicrobial peptides
## 216          PMRC;Cationic antimicrobial peptides
## 217          PMRC_1;Cationic antimicrobial peptides
## 218          PMRF;Cationic antimicrobial peptides
## 219  PNCA;Mycobacterium tuberculosis-specific Drug
## 220          POR-;Multi-drug resistance
## 221          QNRB;Fluoroquinolones
## 222          RAMA;Multi-drug resistance
## 223          RAMR;Multi-drug resistance
## 224          RAMR_1;Multi-drug resistance
## 225          ROBA;Multi-drug resistance
## 226          ROSB+;Cationic antimicrobial peptides
## 227          RPOB;Rifampin
## 228          RPOB_1;Rifampin
## 229          RPOB_12;Rifampin
## 230          RPOB_13;Rifampin
## 231          RPOB_16;Rifampin
## 232          RPOB_17;Rifampin
## 233          RPOB_18;Rifampin
## 234          RPOB_19;Rifampin
## 235          RPOB_2;Rifampin
## 236          RPOB_7;Rifampin
## 237          RPOB_8;Rifampin
## 238          RPSL;Aminoglycosides
## 239          SAT_2;Aminoglycosides
## 240          SDIA;Multi-drug resistance
## 241          SHV;betalactams
## 242          SHV+;betalactams
## 243          SOXS;Multi-drug resistance
## 244          SOXS_2;Multi-drug resistance
## 245          SULI+;Sulfonamides
## 246          SULII;Sulfonamides
## 247          TET32;Tetracyclines
## 248          TET33;Tetracyclines
## 249          TETA_1;Tetracyclines
## 250          TETB_1;Tetracyclines
## 251          TETC;Tetracyclines
## 252          TETD;Tetracyclines
## 253          TETD_1;Tetracyclines
## 254          TETJ;Tetracyclines
## 255          TETL;Tetracyclines
## 256          TETL_2;Tetracyclines
## 257          TETR_1;Tetracyclines
## 258          TETR_3;Tetracyclines
## 259          TETR_7;Tetracyclines
## 260          TETW;Tetracyclines
## 261          TLYA;Aminoglycosides
## 262          TMRB;Tunicamycin
## 263          TOLC;Multi-drug resistance
## 264          TSNR;Thiostrepton
## 265          TUFAB_1;Elfamycins
## 266          TUFAB_12;Elfamycins

```

```

## 267 TUFAB_3;Elfamycins
## 268 TUFAB_4;Elfamycins
## 269 TUFAB_5;Elfamycins
## 270 TUFAB_6;Elfamycins
## 271 TUFAB_7;Elfamycins
## 272 TUFAB_8;Elfamycins
## 273 VANB;Glycopeptides
## 274 VANC;Glycopeptides
## 275 VANC_1;Glycopeptides
## 276 VANRB;Glycopeptides
## 277 VANRC_1;Glycopeptides
## 278 VANRC_2;Glycopeptides
## 279 VANSA;Glycopeptides
## 280 VANSB_1;Glycopeptides
## 281 VANSC;Glycopeptides
## 282 VANSC_2;Glycopeptides
## 283 VANTC;Glycopeptides
## 284 VANTC_1;Glycopeptides
## 285 VANWB;Glycopeptides
## 286 VANXA_4;Glycopeptides
## 287 VANXB;Glycopeptides
## 288 VANXYC;Glycopeptides
## 289 VANXYC_1;Glycopeptides
## 290 VANYA_1;Glycopeptides
## 291 YKKC;Multi-drug resistance
## 292 YKKD;Multi-drug resistance

```

Group

```

resistance_groups <- merge(data.frame(sample_id=ps$sample_id, st=ps$st, rank=ps$rank),
                           resistance_report,
                           by='sample_id')

```

By ST

Function For Plotting

```

plot_by_st <- function(group, top_st, column) {
  by_st <- plyr::count(
    group,
    c('st',
      ifelse(
        length(strsplit(column, ' '))[[1]] >= 2,
        paste0("``", column, "``"),
        column
      )
    )
  )
  by_st$status <- ifelse(by_st[,make.names(column)] == TRUE, 'Resistant', 'Susceptible')
  top_st_resistance <- merge(top_st, by_st, by='st')
  p <- ggplot(data=top_st_resistance, aes(x=reorder(st, -count), y=freq, fill = status)) +
    xlab("Sequence Type") +
    ylab(paste0("Count")) +

```

```

    geom_bar(stat="identity", position = "dodge") +
    geom_text(aes(label=freq), vjust = -0.5, position = position_dodge(.9)) +
    scale_fill_manual(values=c("#2ca25f", "#5ab4ac"), name = column) +
    theme_bw() +
    theme(axis.text=element_text(size=12),
          axis.title=element_text(size=14,face="bold"),
          legend.position="top")
  return(p)
}

```

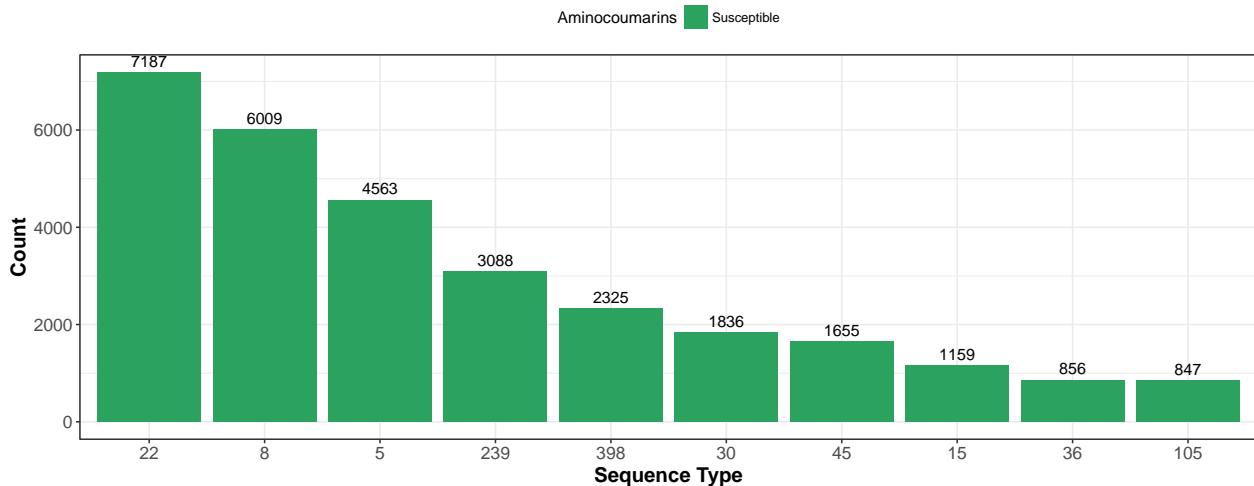
Resistance Classes With A Match By Top 10 Sequence Types

```

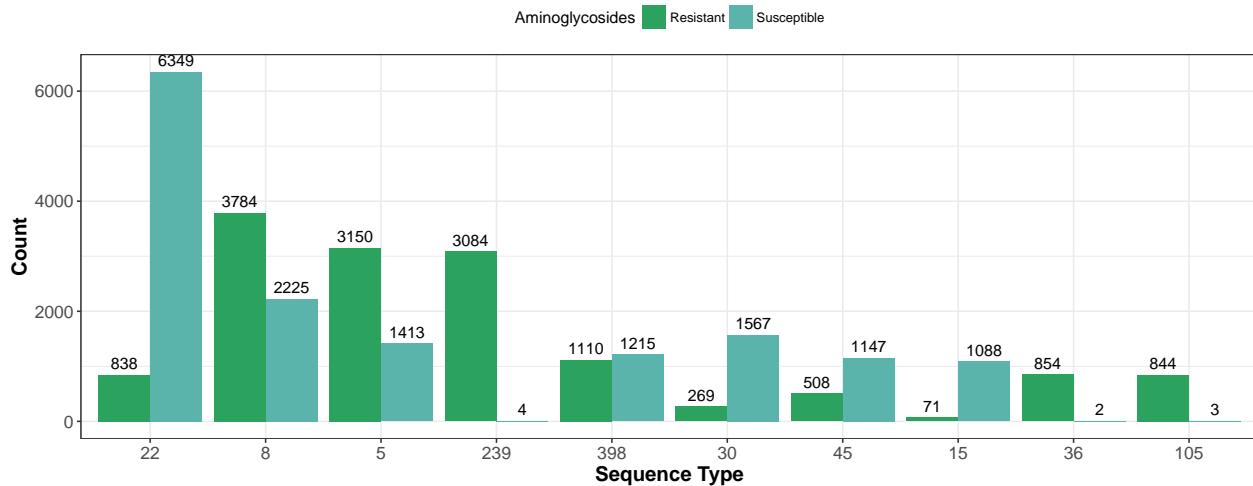
top_st <- get_top_sequence_types()
plots <- list()
i <- 1
for (column in sort(as.character(resistance_report_counts[resistance_report_counts$total > 0,]$class)))
  p <- plot_by_st(resistance_groups, top_st, column)
  print(column)
  print(p)
  if (column == 'Fosfomycin' | column == 'Aminoglycosides' | column == 'MLS') {
    # Output plot to PDF and PNG
    staphopia::write_plot(
      p,
      paste0(getwd(),
        '/images/figure-x-', tolower(column), '-top-10-sequence-types'
      ))
    plots[[i]] <- p
    i <- i + 1
  }
}

```

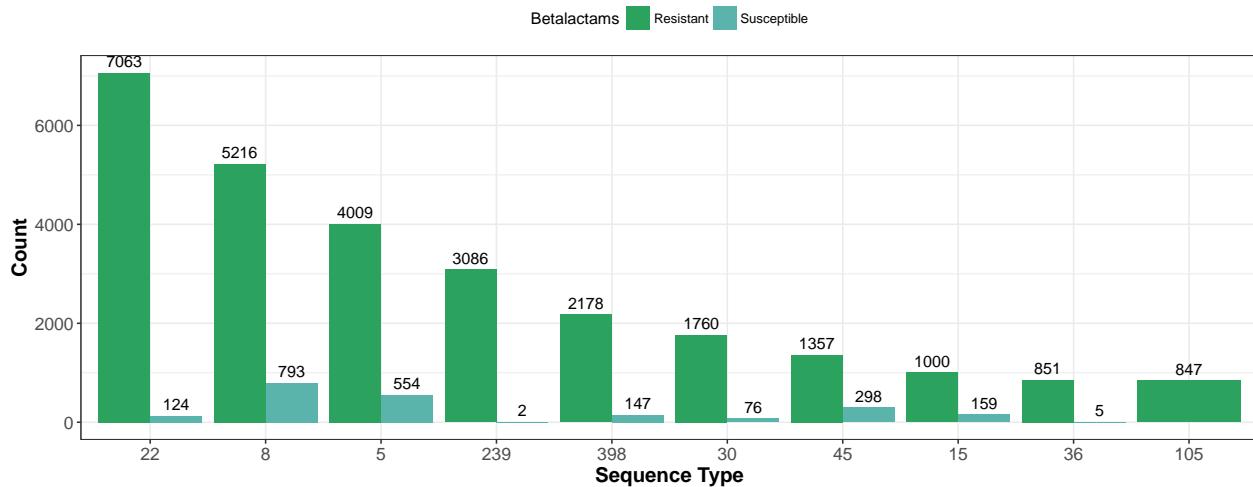
```
## [1] "Aminocoumarins"
```



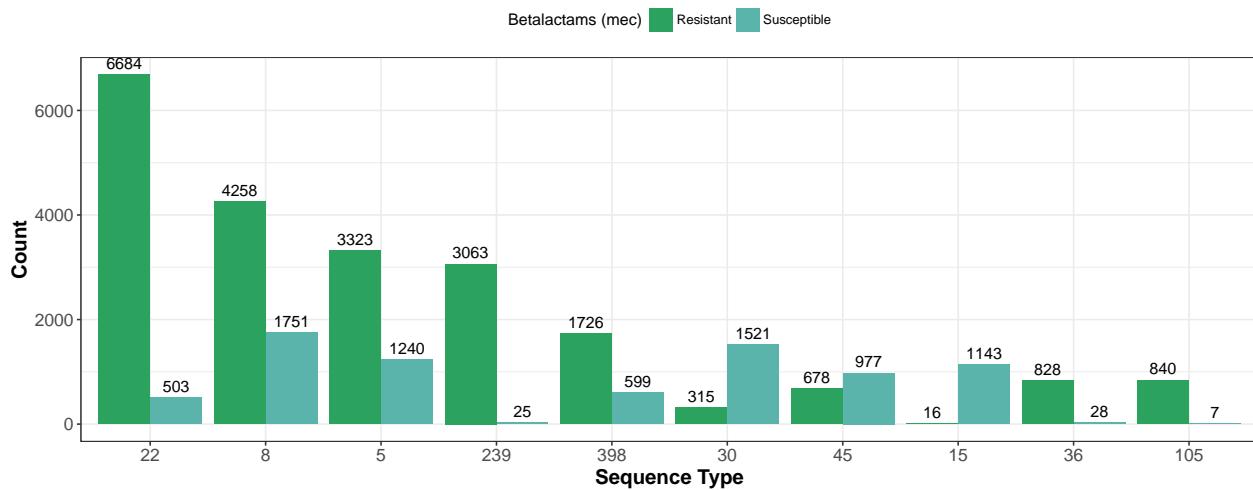
```
## [1] "Aminoglycosides"
```



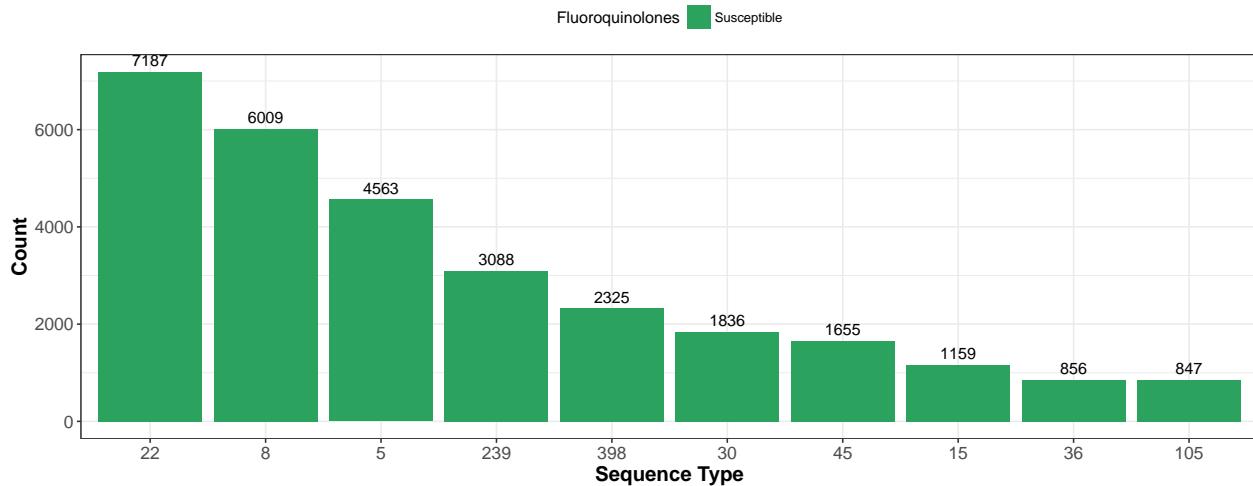
```
## [1] "Betalactams"
```



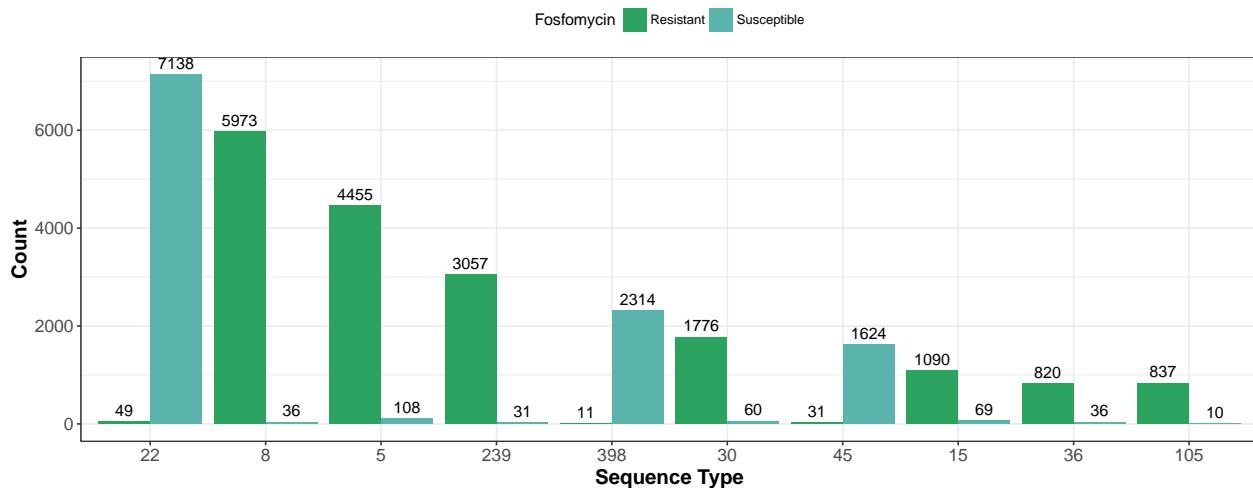
```
## [1] "Betalactams (mec)"
```



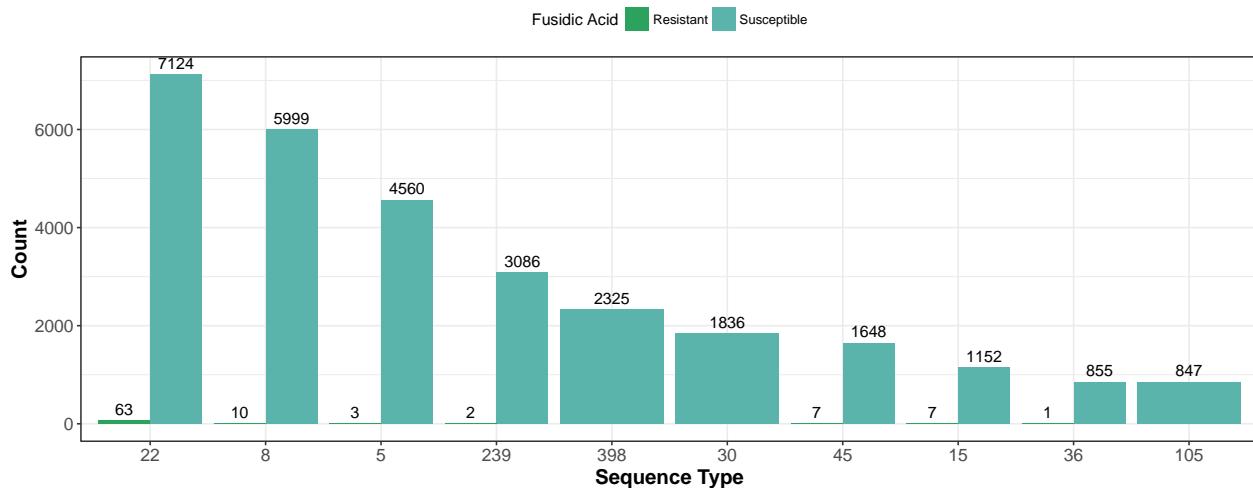
```
## [1] "Fluoroquinolones"
```



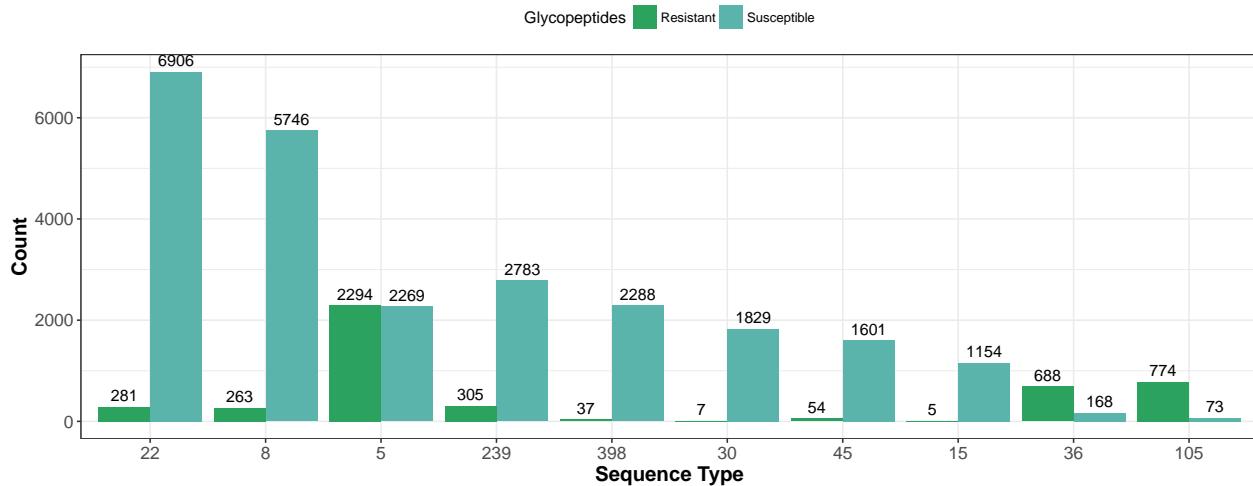
```
## [1] "Fosfomycin"
```



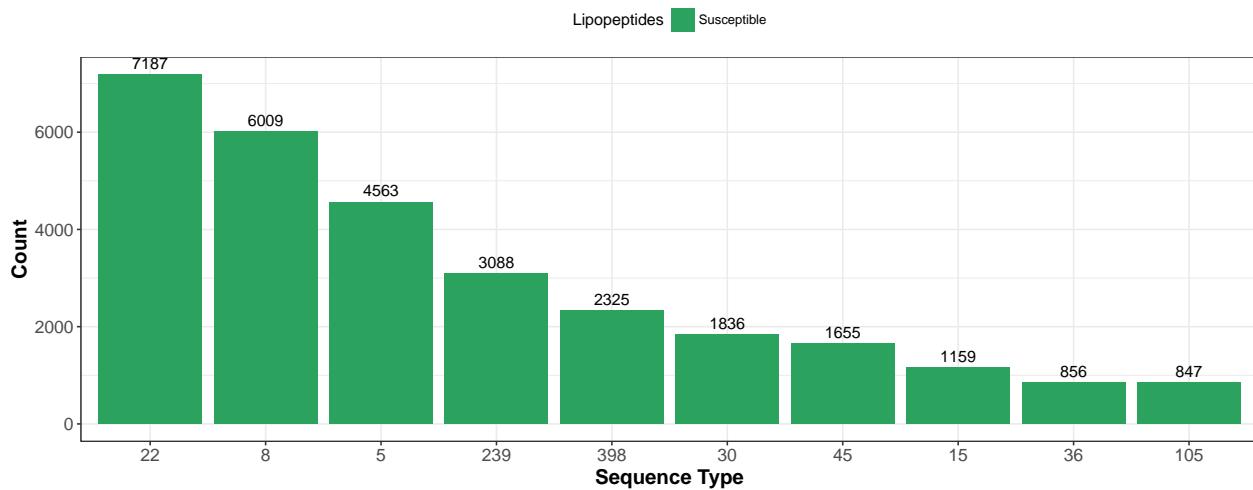
```
## [1] "Fusidic Acid"
```



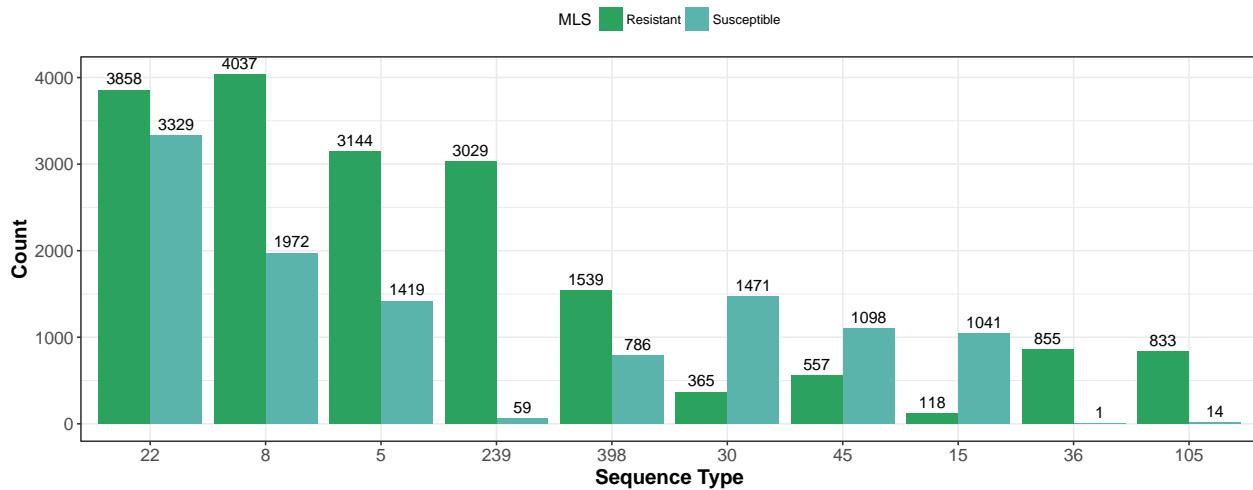
```
## [1] "Glycopeptides"
```



```
## [1] "Lipopeptides"
```

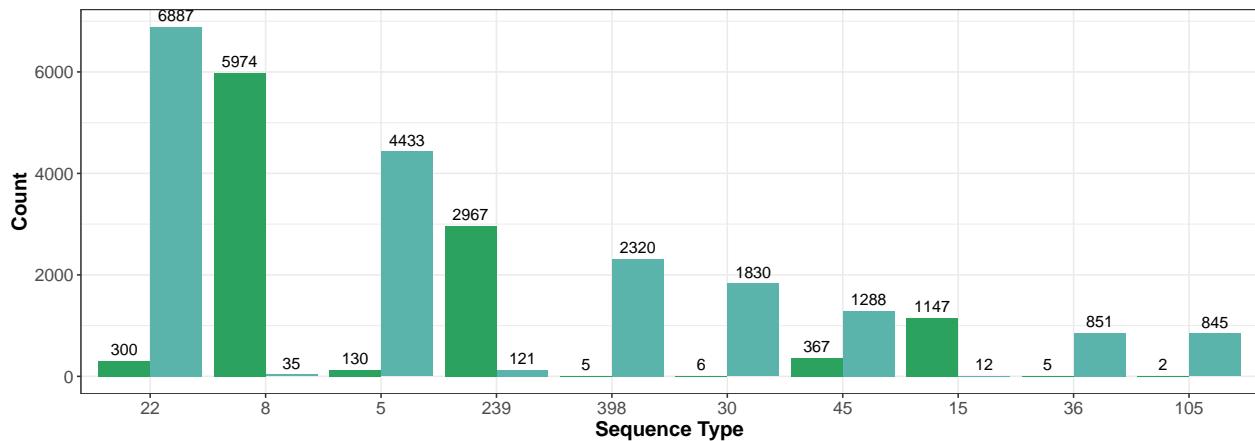


```
## [1] "MLS"
```



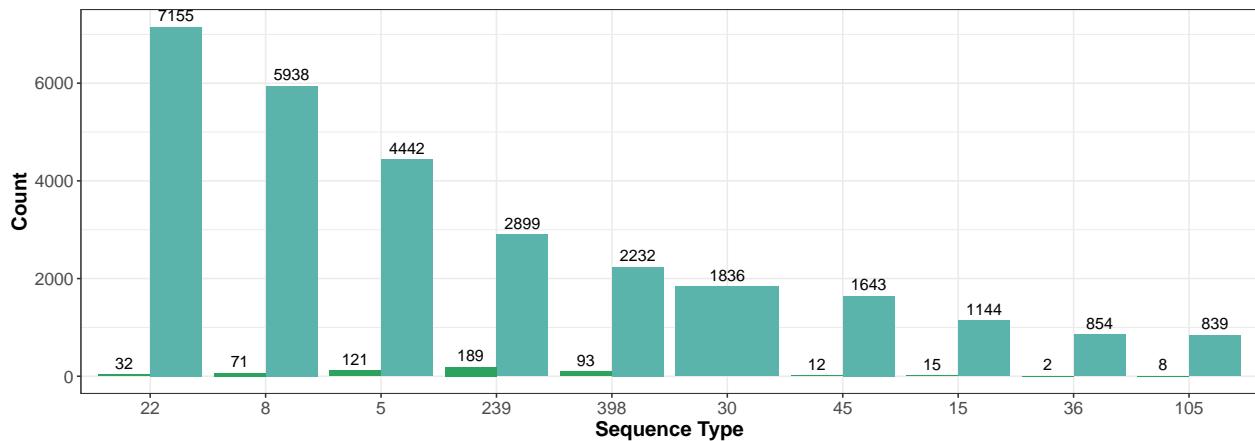
```
## [1] "Multi-Drug Resistance"
```

Multi–Drug Resistance Resistant Susceptible



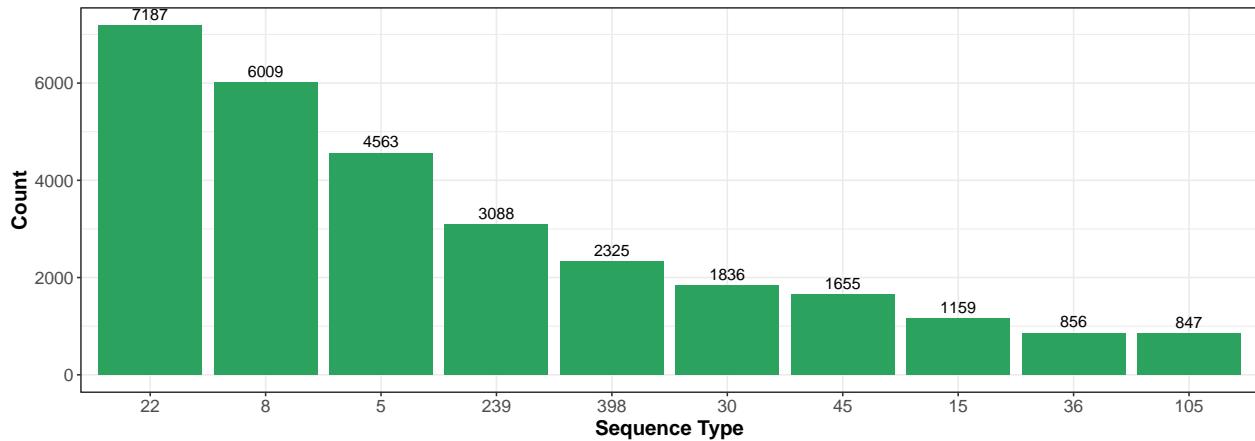
[1] "Phenicol"

Phenicol Resistant Susceptible

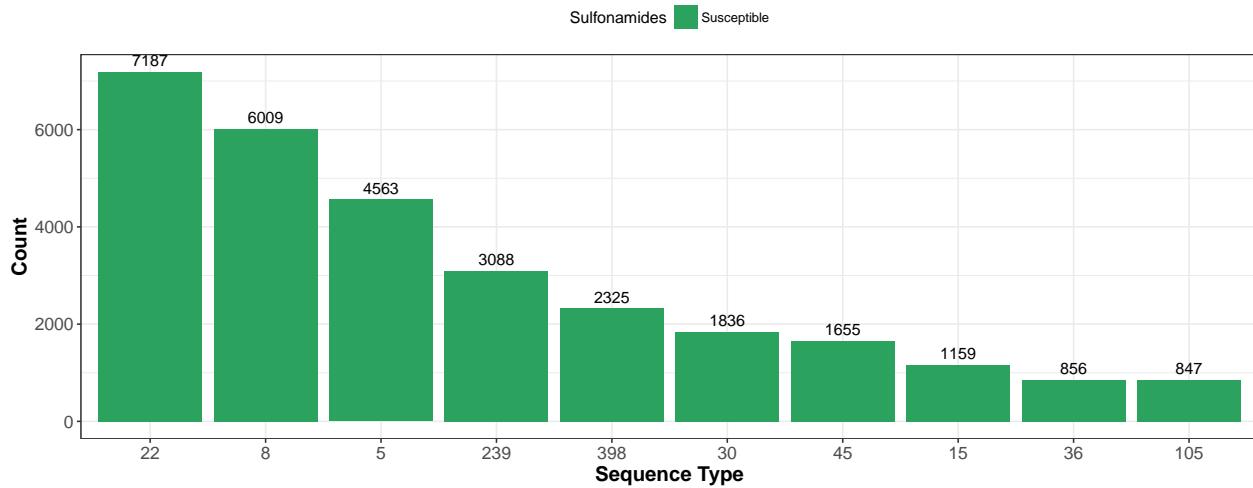


[1] "Rifampin"

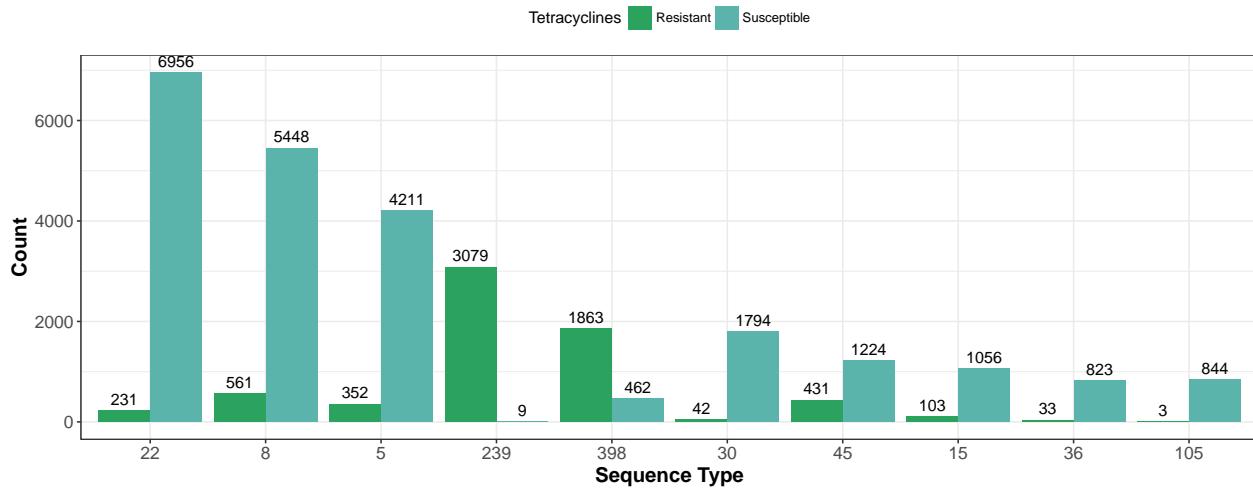
Rifampin Susceptible



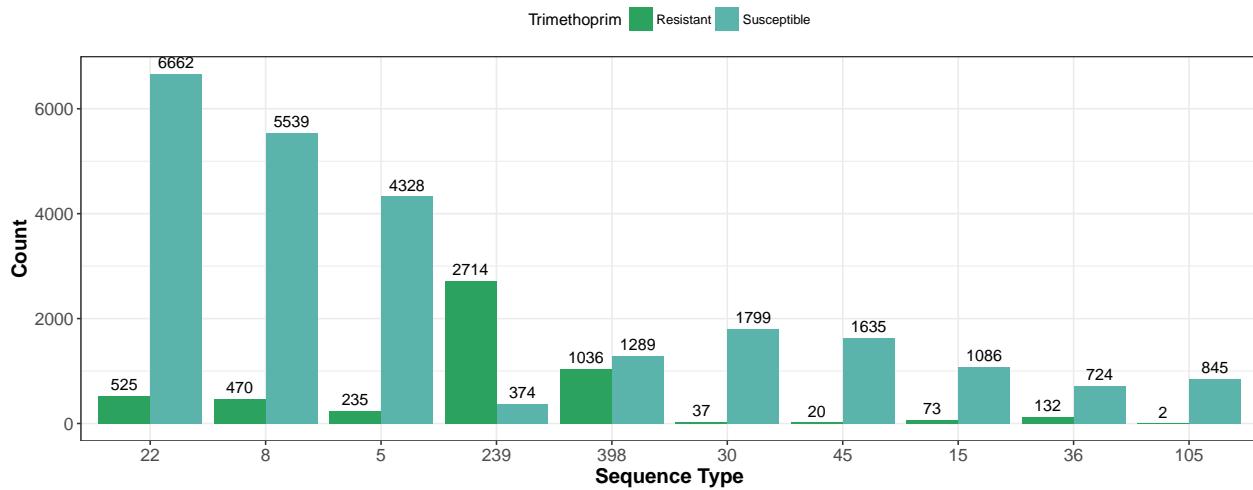
[1] "Sulfonamides"



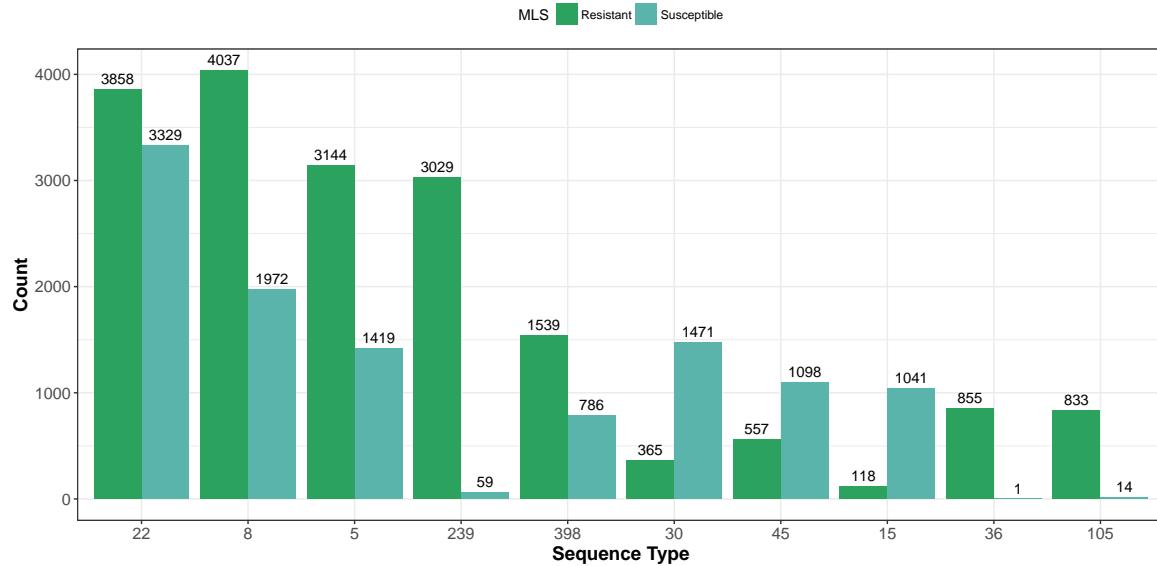
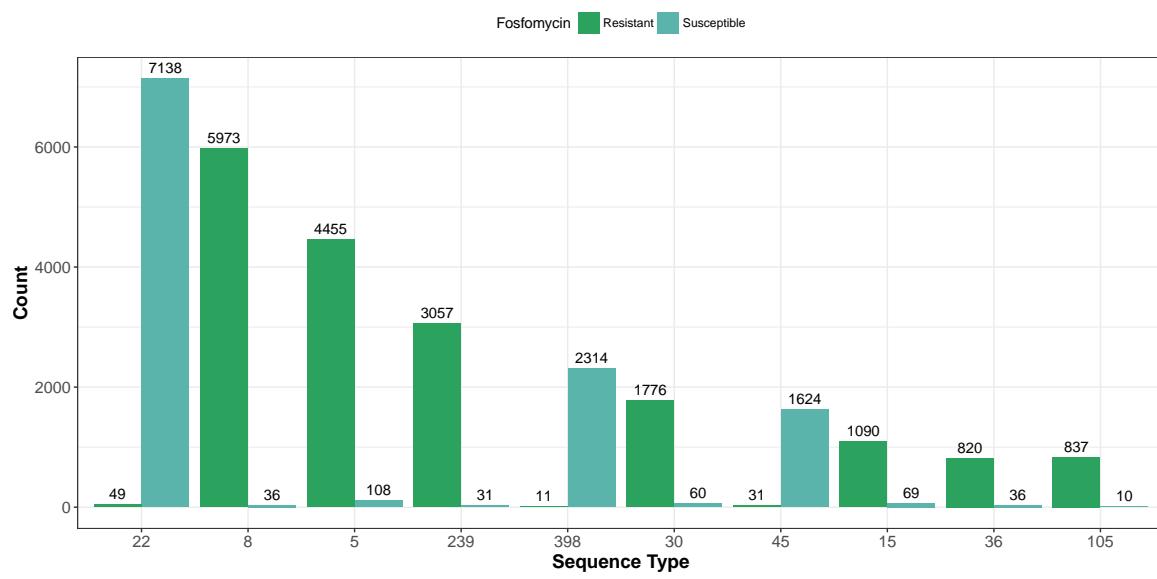
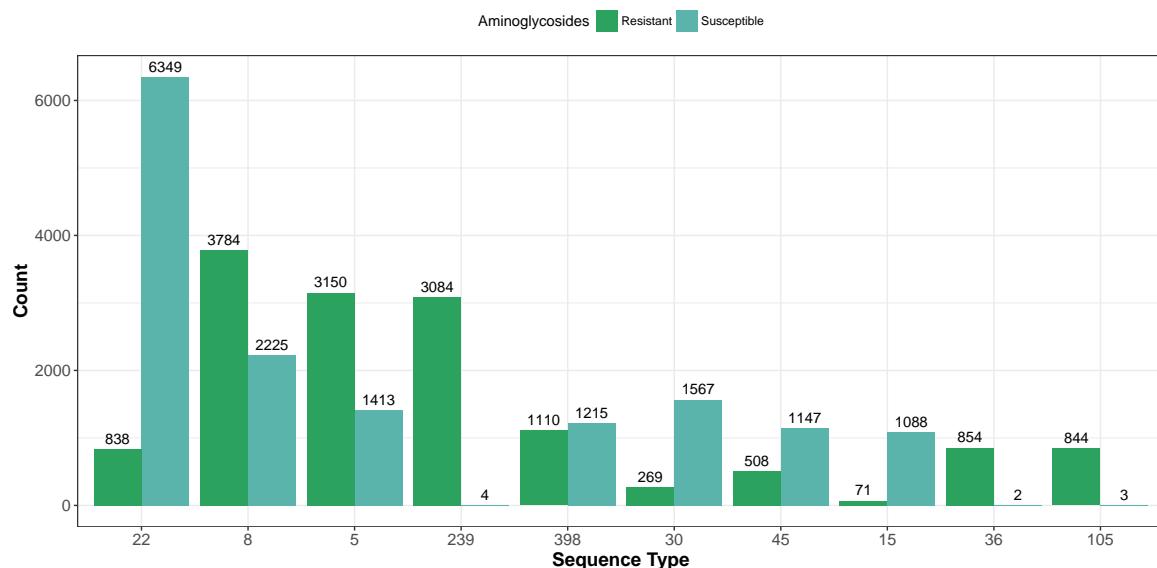
```
## [1] "Tetracyclines"
```



```
## [1] "Trimethoprim"
```



```
grid.arrange(grobs=plots)
```



```

# Output plot to PDF and PNG
pdf(paste0(
  getwd(),
  '/images/figure-x-resistance-facet-top-10-sequence-types.pdf'
), height=15, width=12)
grid.arrange(grobs=plots)
dev_null <- dev.off()

png(paste0(
  getwd(),
  '/images/figure-x-resistance-facet-top-10-sequence-types.png'
), height=1200, width=1200)
grid.arrange(grobs=plots)
dev_null <- dev.off()

```

Session Info

```

sessionInfo()

## R version 3.4.3 (2017-11-30)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 16.04.2 LTS
##
## Matrix products: default
## BLAS: /usr/lib/libblas/libblas.so.3.6.0
## LAPACK: /usr/lib/lapack/liblapack.so.3.6.0
##
## locale:
## [1] LC_CTYPE=en_US.UTF-8          LC_NUMERIC=C
## [3] LC_TIME=en_US.UTF-8          LC_COLLATE=en_US.UTF-8
## [5] LC_MONETARY=en_US.UTF-8       LC_MESSAGES=en_US.UTF-8
## [7] LC_PAPER=en_US.UTF-8          LC_NAME=C
## [9] LC_ADDRESS=C                  LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8    LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats      graphics   grDevices utils      datasets  methods   base
##
## other attached packages:
## [1] bindr_0.2.3     gridExtra_2.3    dplyr_0.7.4      scales_0.5.0
## [5] reshape2_1.4.3   ggplot2_2.2.1    strophe_0.1.9
## [9] 
## loaded via a namespace (and not attached):
## [1] Rcpp_0.12.15      bindr_0.1.1      knitr_1.20
## [4] magrittr_1.5       munsell_0.4.3    colorspace_1.3-2
## [7] R6_2.2.2          rlang_0.1.6      httr_1.3.1
## [10] stringr_1.2.0     plyr_1.8.4       tools_3.4.3
## [13] grid_3.4.3        data.table_1.10.4-3 gtable_0.2.0
## [16] htmltools_0.3.6    assertthat_0.2.0   yaml_2.1.18
## [19] lazyeval_0.2.1     rprojroot_1.3-2    digest_0.6.15
## [22] tibble_1.4.2       curl_3.1         glue_1.2.0
## [25] evaluate_0.10.1    rmarkdown_1.9     labeling_0.3

```

```
## [28] stringi_1.1.6      compiler_3.4.3      pillar_1.1.0
## [31] backports_1.1.2     jsonlite_1.5       pkgconfig_2.0.1
```