final_project

Ashley Malmlov & Katrina Wheeler

2022-05-31

First, we load and clean the data - isolating the specific columns we are interested in, and removing any NA values.

```
raw.dat <- read.csv(file = 'batdata.csv')
colnames(raw.dat)
```

```
##  [1] "Sample"              "SamplingEventID"     "Samplers"
##  [4] "Samplingdate"        "Birthdate"           "Mths.since.birthing"
##  [7] "Cont_island"         "Country"             "Region"
## [10] "Location"            "Latitude"            "Longitude"
## [13] "Bat.wt"              "Sex"                 "Age"
## [16] "Age.3"               "Teeth.Age"           "Teeth.CC"
## [19] "Teeth.Age.Range"     "Teeth.Age.mths"      "Age.mths"
## [22] "Repro.status"        "Mother.ID"           "Offspring.ID"
## [25] "Forearm"             "Band.no."            "LBV.mFAVN"
## [28] "Henipavirus"         "AchPV1"              "AchPV2"
## [31] "GeneticsID"          "Cytb"                "GenBankAccession"
## [34] "T.1"                 "T.2"                 "S.1"
## [37] "S.2"                 "F.1"                 "F.2"
## [40] "W.1"                 "W.2"                 "N.1"
## [43] "N.2"                 "Q.1"                 "Q.2"
## [46] "X.1"                 "X.2"                 "P.1"
## [49] "P.2"                 "K.1"                 "K.2"
## [52] "Ac.1"                "Ac.2"                "Af.1"
## [55] "Af.2"                "Ai.1"                "Ai.2"
## [58] "Ad.1"                "Ad.2"                "Y.1"
## [61] "Y.2"                 "Ag.1"                "Ag.2"
## [64] "Ah.1"                "Ah.2"                "B.1"
## [67] "B.2"                 "M.1"                 "M.2"
```

```
dat <- raw.dat[, c(4,5,8, 14, 15, 16, 28)]

dat.clean <- na.omit(dat)
```

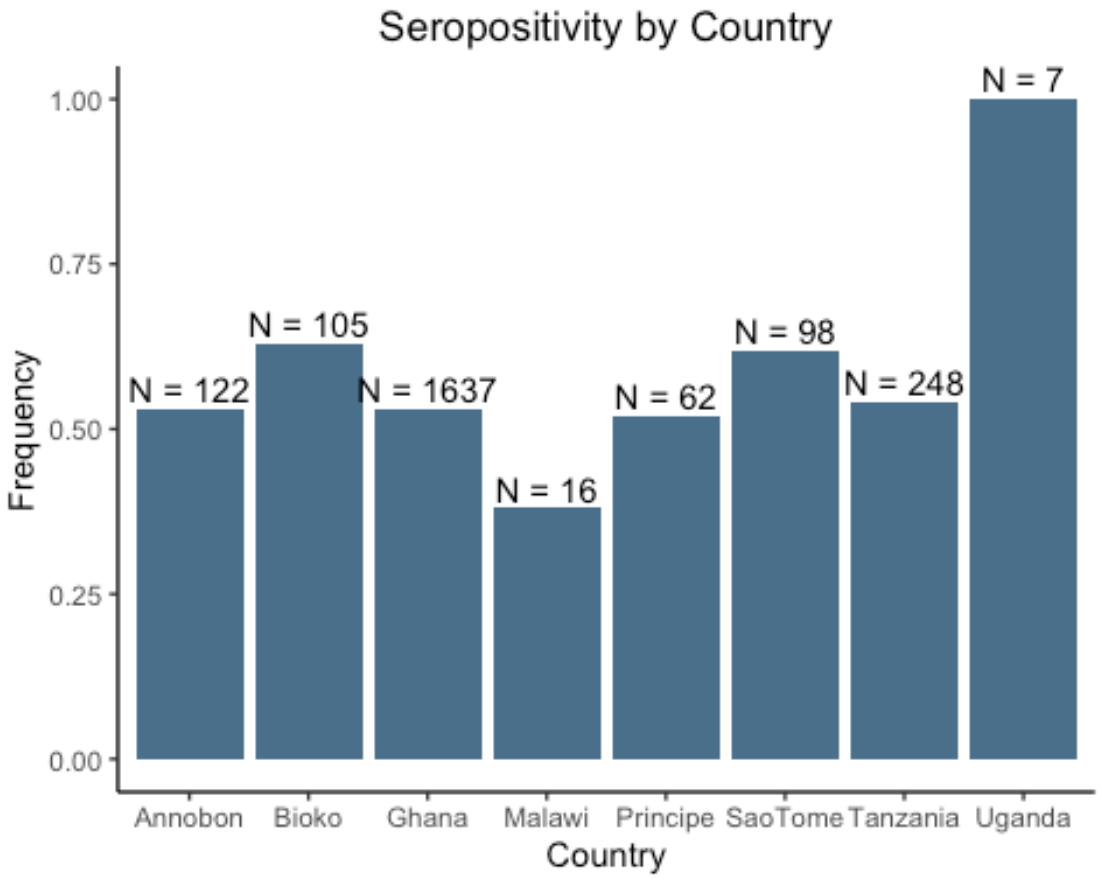
Below, is a bar graph of seropositivity by country

```
country.neg <- dat.clean%>%group_by(Country)%>%summarize(No.Neg=sum(Henipavir
us == 0))

country.pos <- dat.clean%>%group_by(Country)%>%summarize(No.Pos=sum(Henipavir
us != 0))
```

```r
count.country <- cbind(country.neg, No.Pos=country.pos$No.Pos)
count.country <- cbind(count.country, Total=count.country$No.Neg+count.countr
y$No.Pos)
count.country <- cbind(count.country, f.i=round(count.country$No.Pos/count.co
untry$Total, digits=2))

count.country
```

```
##    Country No.Neg No.Pos Total  f.i
## 1  Annobon     57     65   122 0.53
## 2    Bioko     39     66   105 0.63
## 3    Ghana    767    870  1637 0.53
## 4   Malawi     10      6    16 0.38
## 5 Principe     30     32    62 0.52
## 6  SaoTome     37     61    98 0.62
## 7 Tanzania    113    135   248 0.54
## 8   Uganda      0      7     7 1.00
```

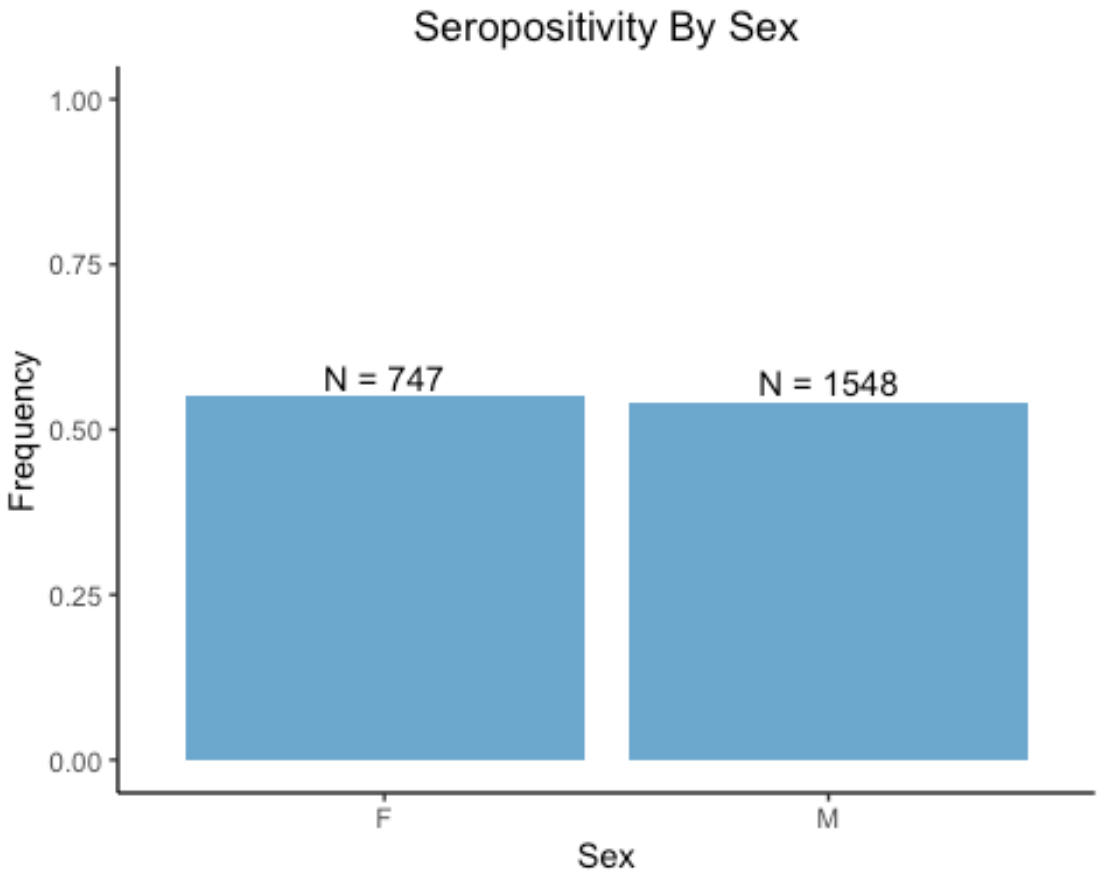
```r
bar.country <- ggplot(count.country, aes(x=Country, y=f.i)) + geom_bar(fill='
skyblue4', stat = 'identity') + ylab('Frequency') + ggtitle('Seropositivity b
y Country') + theme_classic()

bar.country <- bar.country + theme(plot.title = element_text(hjust=0.5))

bar.country <- bar.country + geom_text(aes(label=paste('N =', Total)), vjust
= -0.3)
bar.country
```



The following graph represents seropositivity by sex.

```
sex.neg <- dat.clean%>%group_by(Sex)%>%summarize(No.Neg=sum(Henipavirus == 0)
)

sex.pos <- dat.clean%>%group_by(Sex)%>%summarize(No.Pos=sum(Henipavirus != 0)
)

count.sex <- cbind(sex.neg, No.Pos=sex.pos$No.Pos)
count.sex <- cbind(count.sex, Total=count.sex$No.Neg+count.sex$No.Pos)
count.sex <- cbind(count.sex, f.i=round(count.sex$No.Pos/count.sex$Total, dig
its=2))

count.sex

##   Sex No.Neg No.Pos Total  f.i
## 1   F    337    410   747 0.55
## 2   M    716    832  1548 0.54

bar.sex <- ggplot(count.sex, aes(x=Sex, y=f.i)) + geom_bar(fill='skyblue3', s
tat = 'identity') + ylab('Frequency') + ggtitle('Seropositivity By Sex') + th
eme_classic()
```

```
bar.sex <- bar.sex + theme(plot.title = element_text(hjust=0.5))

bar.sex <- bar.sex + geom_text(aes(label=paste('N =', Total)), vjust = -0.3)

bar.sex <- bar.sex + coord_cartesian(ylim =c(0,1))
bar.sex
```



Our next graph represents seropositivity by age, as characterized by the following.

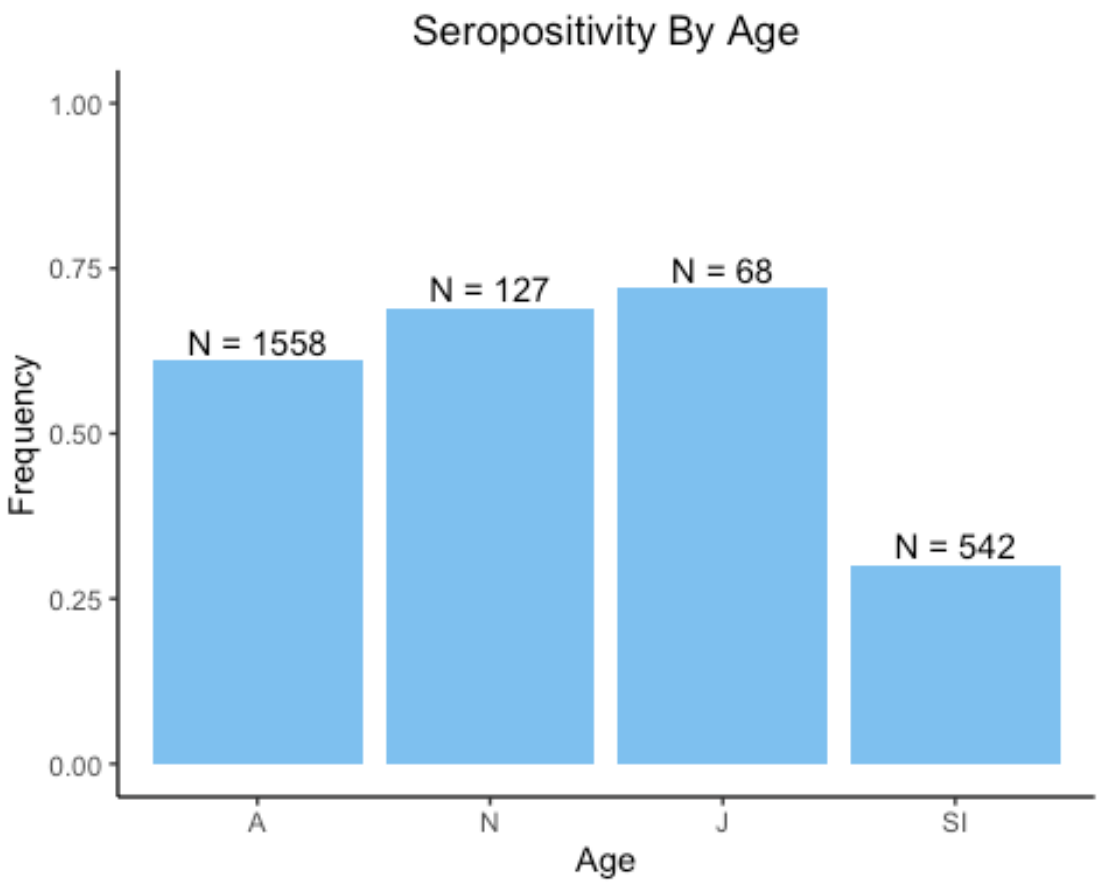
Description of Age classification (assessed by morphological characteristics): N – Neonate; <2mths J – Juvenile; 2 – <6 months SI – Sexually Immature; 6 – <24 months A – Adult; ≥24 months

```
age.neg <- dat.clean%>%group_by(Age)%>%summarize(No.Neg=sum(Henipavirus == 0)
)

age.pos <- dat.clean%>%group_by(Age)%>%summarize(No.Pos=sum(Henipavirus != 0)
)

count.age <- cbind(age.neg, No.Pos=age.pos$No.Pos)
count.age <- cbind(count.age, Total=count.age$No.Neg+count.age$No.Pos)
count.age <- cbind(count.age, f.i=round(count.age$No.Pos/count.age$Total, dig
its=2))
```

```
count.age
```

```
##   Age No.Neg No.Pos Total  f.i
## 1   A    612    946  1558 0.61
## 2   J     19     49    68 0.72
## 3   N     40     87   127 0.69
## 4  SI    382    160   542 0.30
```

```
bar.age <- ggplot(count.age, aes(x=Age, y=f.i)) + geom_bar(fill='skyblue2', s
tat = 'identity') + ylab('Frequency') + ggtitle('Seropositivity By Age') + th
eme_classic()

bar.age <- bar.age + theme(plot.title = element_text(hjust=0.5))

bar.age <- bar.age + geom_text(aes(label=paste('N =', Total)), vjust = -0.3)

bar.age <- bar.age + coord_cartesian(ylim =c(0,1))

bar.age <- bar.age +scale_x_discrete(limits = c("A", "N", "J", "SI"))
bar.age
```



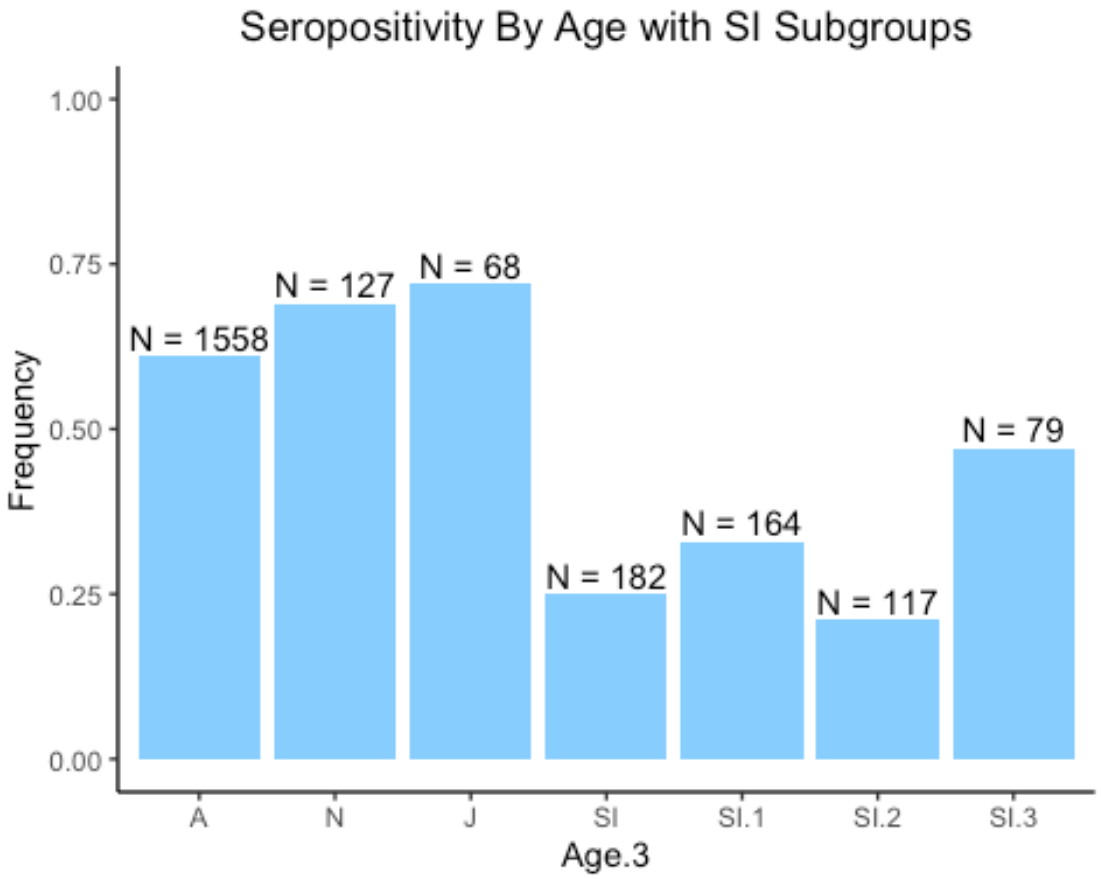
We've broken the age down even farther as designated in the original date, breaking the Sexually Immature category down to 6 month increments.

Description of Age.3 classification (assessed by morphological characteristics and timing of sampling relative to the birth pulse. In some bats classified as SI, the timing of sampling in relation to the birthing season permitted further classification of SI individuals into 6-month age groups: SI.1; 6 – <12 months SI.2; 12 – <18 months SI.3; 18 – <24 months For those SI bats that could not be more accurately classified, they remain in the category of SI, ranging in age from 6 months to <24 months.

```
ageSI.sub.neg <- dat.clean%>%group_by(Age.3)%>%summarize(No.Neg=sum(Henipavir
us == 0))

ageSI.sub.pos <- dat.clean%>%group_by(Age.3)%>%summarize(No.Pos=sum(Henipavir
us != 0))

count.ageSI.sub <- cbind(ageSI.sub.neg, No.Pos=ageSI.sub.pos$No.Pos)
count.ageSI.sub <- cbind(count.ageSI.sub, Total=count.ageSI.sub$No.Neg+count.
ageSI.sub$No.Pos)
count.ageSI.sub <- cbind(count.ageSI.sub, f.i=round(count.ageSI.sub$No.Pos/co
unt.ageSI.sub$Total, digits=2))

count.ageSI.sub
```

```
##   Age.3 No.Neg No.Pos Total  f.i
## 1     A    612    946  1558 0.61
## 2     J     19     49    68 0.72
## 3     N     40     87   127 0.69
## 4    SI    137     45   182 0.25
## 5  SI.1    110     54   164 0.33
## 6  SI.2     93     24   117 0.21
## 7  SI.3     42     37    79 0.47
```

```
bar.age3 <- ggplot(count.ageSI.sub, aes(x=Age.3, y=f.i)) + geom_bar(fill='sky
blue1', stat = 'identity') + ylab('Frequency') + ggtitle('Seropositivity By A
ge with SI Subgroups') + theme_classic()

bar.age3 <- bar.age3 + theme(plot.title = element_text(hjust=0.5))

bar.age3 <- bar.age3 + geom_text(aes(label=paste('N =', Total)), vjust = -0.3
)

bar.age3 <- bar.age3 + coord_cartesian(ylim =c(0,1))

bar.age3 <- bar.age3 + scale_x_discrete(limits = c("A", "N", "J", "SI", "SI.1
", "SI.2", "SI.3"))
bar.age3
```

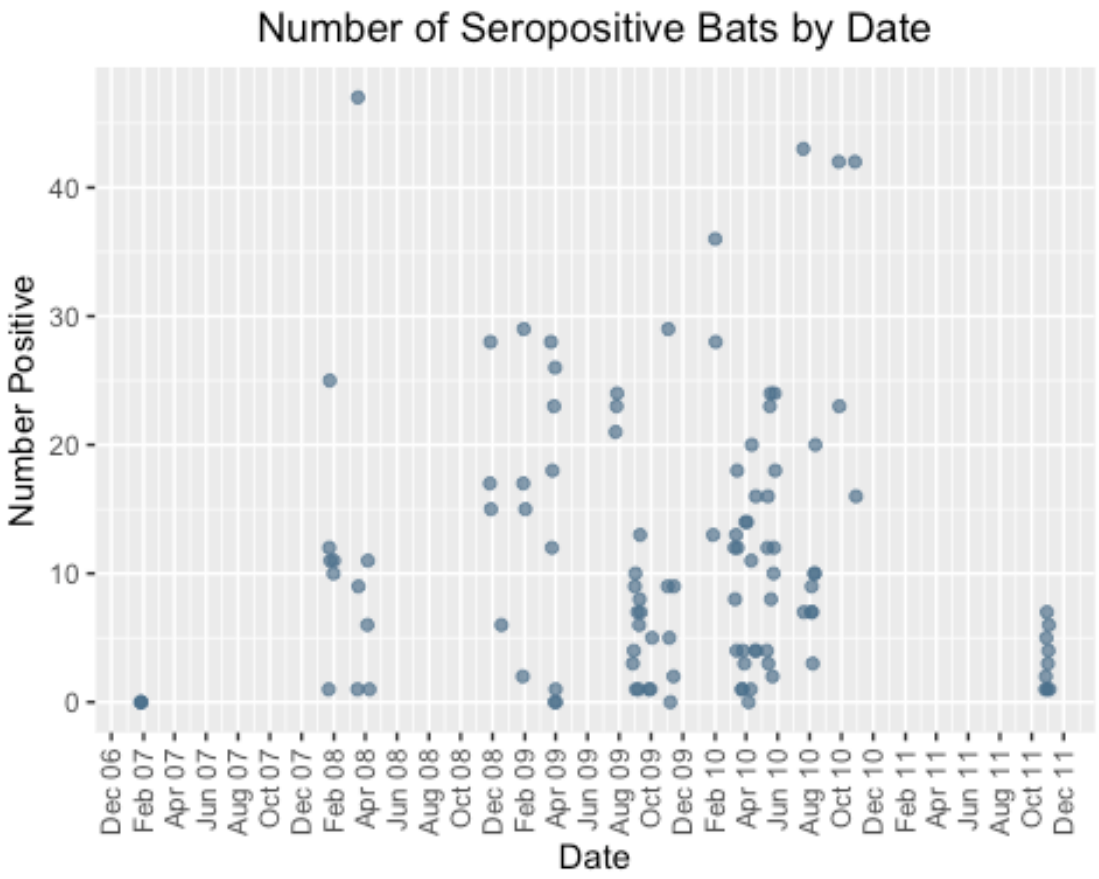


Our next visualization presents the number of seropositive bats by sampling date.

```
samp.date.neg <- dat.clean%>%group_by(Samplingdate)%>%summarize(No.Neg=sum(He
nipavirus == 0))

samp.date.pos <- dat.clean%>%group_by(Samplingdate)%>%summarize(No.Pos=sum(He
nipavirus != 0))

count.samp.date <- cbind(samp.date.neg, No.Pos=samp.date.pos$No.Pos)
count.samp.date <- cbind(count.samp.date, Total=count.samp.date$No.Neg+count.
samp.date$No.Pos)
count.samp.date <- cbind(count.samp.date, Prob.Pos=round(count.samp.date$No.P
os/count.samp.date$Total, digits=2))

count.samp.date$Samplingdate <- as.Date(count.samp.date$Samplingdate)

head(count.samp.date)

##   Samplingdate No.Neg No.Pos Total Prob.Pos
## 1   2007-01-26     24      0    24     0.00
## 2   2007-01-27     24      0    24     0.00
## 3   2007-01-28     15      0    15     0.00
## 4   2008-01-22      1      1     2     0.50
```

```
## 5   2008-01-23      4     12     16     0.75
## 6   2008-01-24     13     25     38     0.66
```

```
ggplot(count.samp.date, aes(Samplingdate, No.Pos)) +
          geom_point(color="skyblue4", alpha = 0.7) + ggtitle("Number of Ser
opositive Bats by Date") +
          xlab("Date") + ylab("Number Positive") + (scale_x_date(breaks=date
_breaks("2 months"),
      labels=date_format("%b %y"))) + theme(plot.title = element_text(hjust=0
.5)) + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Below, we've converted the strings we are interested in into factors. We then use the factors to test a logistic regression on the Age variable, and plot the predicted probability of seropositivity based on age.

```
date_conversion <- dat.clean%>%mutate(Samplingdate=as.Date(Samplingdate, form
at="%Y-%m-%d"))
```

```
data_new1 <- date_conversion
data_new1$year <- strftime(data_new1$Samplingdate, "%Y")
data_new1$month <- strftime(data_new1$Samplingdate, "%m")
head(data_new1)
```

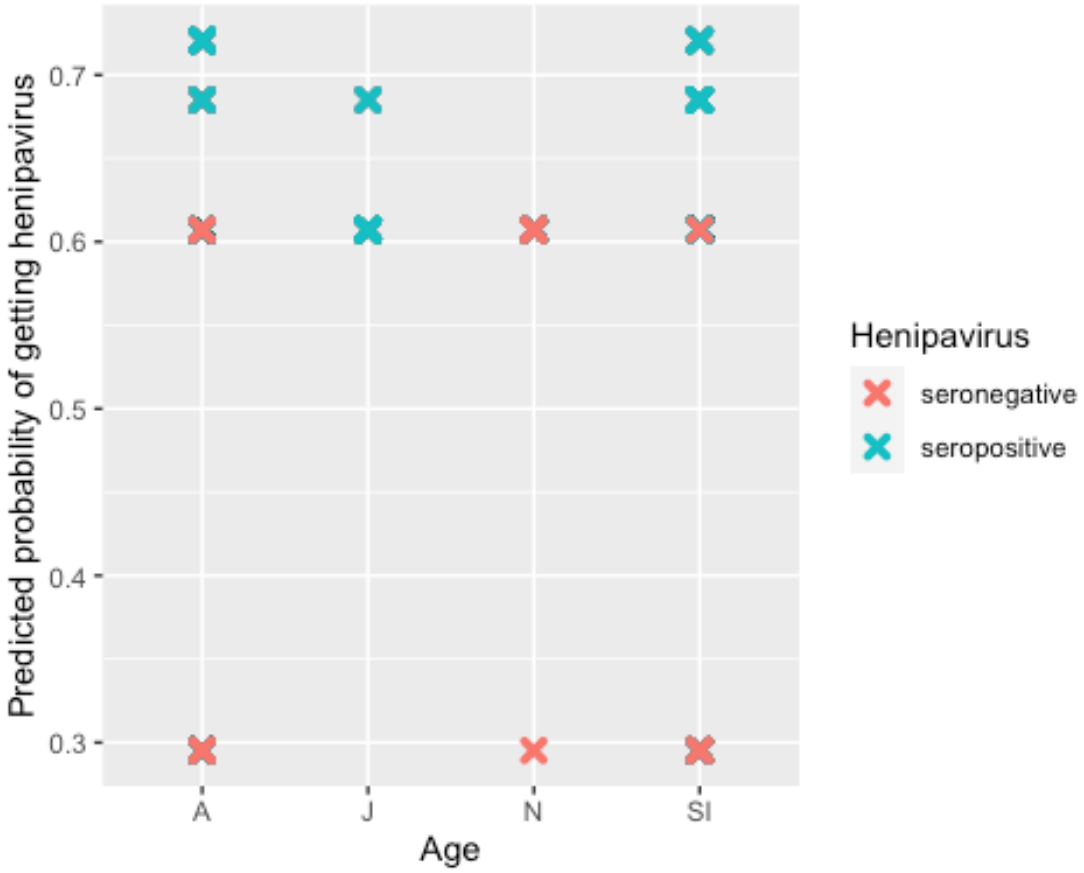
```
##    Samplingdate Birthdate Country Sex Age Age.3 Henipavirus year month
## 7    2007-01-26     1-Mar   Ghana   F   A     A           0 2007    01
## 8    2007-01-26     1-Mar   Ghana   F   A     A           0 2007    01
## 9    2007-01-26     1-Mar   Ghana   F   A     A           0 2007    01
## 10   2007-01-26     1-Mar   Ghana   M   A     A           0 2007    01
## 11   2007-01-26     1-Mar   Ghana   M   A     A           0 2007    01
## 12   2007-01-26     1-Mar   Ghana   F   A     A           0 2007    01
```

```
data_new1$Sex <- as.factor(data_new1$Sex)
data_new1$Country <- as.factor(data_new1$Country)
data_new1[data_new1$Henipavirus == 0,]$Henipavirus <- "seronegative"
data_new1[data_new1$Henipavirus == 1,]$Henipavirus <- "seropositive"
data_new1$Henipavirus <- as.factor(data_new1$Henipavirus)
data_new1$Age <- as.factor(data_new1$Age)
data_new1$Age.3 <- as.factor(data_new1$Age.3)

logisticAGE <- glm(Henipavirus ~ Age, data = data_new1, family = "binomial")
summary(logisticAGE)
```

```
##
## Call:
## glm(formula = Henipavirus ~ Age, family = "binomial", data = data_new1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5969  -1.3671   0.8698   0.9989   1.5621
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.43551    0.05188   8.395   <2e-16 ***
## AgeJ         0.51187    0.27519   1.860   0.0629 .
## AgeN         0.34152    0.19795   1.725   0.0845 .
## AgeSI       -1.30576    0.10751 -12.145   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3166.0  on 2294  degrees of freedom
## Residual deviance: 2984.2  on 2291  degrees of freedom
## AIC: 2992.2
##
## Number of Fisher Scoring iterations: 4
```

```
predicted.data <- data.frame(
  probability.of.henipavirus = logisticAGE$fitted.values,
  Henipavirus = data_new1$Henipavirus
)

predicted.data <- predicted.data[
```

```
  order(predicted.data$probability.of.henipavirus, decreasing= FALSE),]

predicted.data$rank <- 1:nrow(predicted.data)

library(ggplot2)
library(cowplot)

##
## Attaching package: 'cowplot'

## The following object is masked from 'package:ggpubr':
##
##     get_legend

## The following object is masked from 'package:ggthemes':
##
##     theme_map

## The following object is masked from 'package:lubridate':
##
##     stamp

ggplot(data=predicted.data, aes(x=data_new1$Age, y=probability.of.henipavirus
)) +
  geom_point(aes(color=Henipavirus), alpha=1, shape=4, stroke=2) +
  xlab("Age") +
  ylab("Predicted probability of getting henipavirus")
```

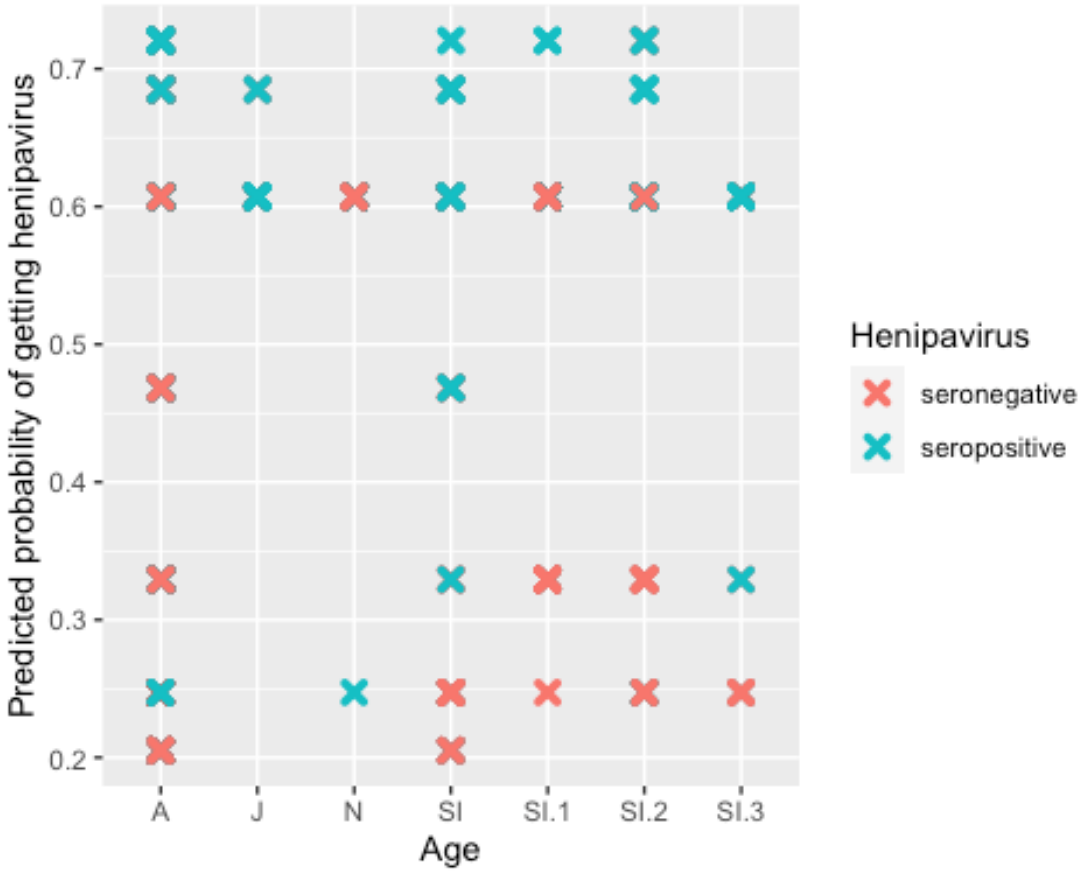


We tested the logistic regression with the Age.3 category, and provided the same predictive graph.

```
#Henipavirus vs Age3
logisticAGE3 <- glm(Henipavirus ~ Age.3, data = dat.clean, family = "binomial
")
summary(logisticAGE3)

##
## Call:
## glm(formula = Henipavirus ~ Age.3, family = "binomial", data = dat.clean)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5969  -1.3671   0.8698   0.9989   1.7800
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.43551    0.05188   8.395  < 2e-16 ***
## Age.3J       0.51187    0.27519   1.860   0.0629 .
## Age.3N       0.34152    0.19795   1.725   0.0845 .
## Age.3SI     -1.54883    0.17948  -8.630  < 2e-16 ***
## Age.3SI.1   -1.14701    0.17407  -6.589 4.42e-11 ***
```

```
## Age.3SI.2   -1.79006    0.23476  -7.625 2.44e-14 ***
## Age.3SI.3   -0.56226    0.23136  -2.430   0.0151 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3166.0  on 2294  degrees of freedom
## Residual deviance: 2965.9  on 2288  degrees of freedom
## AIC: 2979.9
##
## Number of Fisher Scoring iterations: 4
```

```
predicted.data <- data.frame(
  probability.of.henipavirus = logisticAGE3$fitted.values,
  Henipavirus = data_new1$Henipavirus
)

predicted.data <- predicted.data[
  order(predicted.data$probability.of.henipavirus, decreasing= FALSE),]

predicted.data$rank <- 1:nrow(predicted.data)

library(ggplot2)
library(cowplot)

ggplot(data=predicted.data, aes(x=data_new1$Age.3, y=probability.of.henipavir
us)) +
  geom_point(aes(color=Henipavirus), alpha=1, shape=4, stroke=2) +
  xlab("Age") +
  ylab("Predicted probability of getting henipavirus")
```



We used cross validation to test the accuracy of our model.

```
fitControl1 <- trainControl(method = "cv", number = 5, savePredictions = T)

mod_fitcv <- train(Henipavirus ~ Age.3, data = data_new1, method = "glm", fam
ily = "binomial", trControl = fitControl1)

summary(mod_fitcv)

##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5969  -1.3671   0.8698   0.9989   1.7800
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.43551    0.05188   8.395  < 2e-16 ***
## Age.3J       0.51187    0.27519   1.860   0.0629 .
## Age.3N       0.34152    0.19795   1.725   0.0845 .
## Age.3SI     -1.54883    0.17948  -8.630  < 2e-16 ***
```

```
## Age.3SI.1   -1.14701    0.17407  -6.589 4.42e-11 ***
## Age.3SI.2   -1.79006    0.23476  -7.625 2.44e-14 ***
## Age.3SI.3   -0.56226    0.23136  -2.430   0.0151 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3166.0  on 2294  degrees of freedom
## Residual deviance: 2965.9  on 2288  degrees of freedom
## AIC: 2979.9
##
## Number of Fisher Scoring iterations: 4
```

```
caret::confusionMatrix(table((mod_fitcv$pred)$pred,(mod_fitcv$pred)$obs))
```

```
## Confusion Matrix and Statistics
##
##
##                seronegative seropositive
##   seronegative          382          160
##   seropositive          671         1082
##
##                Accuracy : 0.6379
##                  95% CI : (0.6179, 0.6576)
##     No Information Rate : 0.5412
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.2429
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.3628
##             Specificity : 0.8712
##          Pos Pred Value : 0.7048
##          Neg Pred Value : 0.6172
##              Prevalence : 0.4588
##          Detection Rate : 0.1664
##    Detection Prevalence : 0.2362
##       Balanced Accuracy : 0.6170
##
##        'Positive' Class : seronegative
##
```

CV with 10 repeated models

```
crossValSettings <- trainControl(method = "repeatedcv", number = 10, savePred
ictions = TRUE)

crossVal <- train(Henipavirus ~ Age.3, data = data_new1, family ="binomial",
```

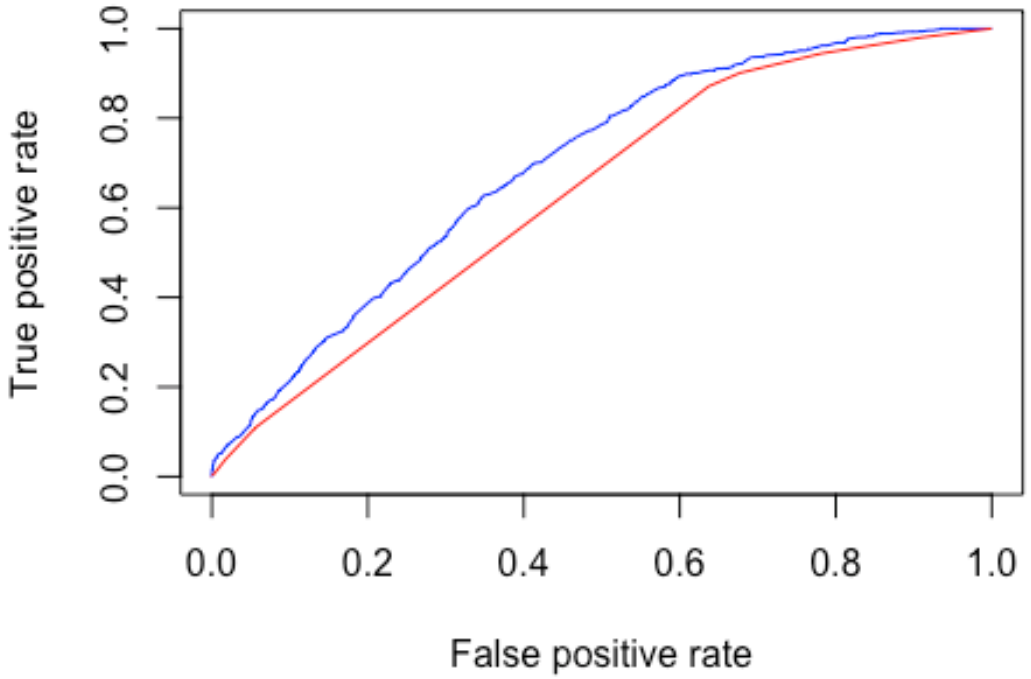
```
method = "glm", trControl = crossValSettings)
crossVal

## Generalized Linear Model
##
## 2295 samples
##    1 predictor
##    2 classes: 'seronegative', 'seropositive'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 1 times)
## Summary of sample sizes: 2066, 2065, 2066, 2066, 2065, 2065, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.6379343  0.2429679

pred <- predict(crossVal, newdata = data_new1)
confusionMatrix(data = pred, data_new1$Henipavirus)

## Confusion Matrix and Statistics
##
##               Reference
## Prediction     seronegative seropositive
##   seronegative          382          160
##   seropositive          671         1082
##
##                Accuracy : 0.6379
##                  95% CI : (0.6179, 0.6576)
##     No Information Rate : 0.5412
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.2429
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.3628
##             Specificity : 0.8712
##          Pos Pred Value : 0.7048
##          Neg Pred Value : 0.6172
##              Prevalence : 0.4588
##          Detection Rate : 0.1664
##    Detection Prevalence : 0.2362
##       Balanced Accuracy : 0.6170
##
##        'Positive' Class : seronegative
##
```

Confusion matrix: Prediction table: reference is what our data was, prediction is what the model got. This shows us true negatives (382), true positives (1082), false negatives (160), and false positives (671)

Accuracy shows us how accurate the model is: 64% Below is ROC curve, which reflects the True Positive rate and False Positive rate

```
logisticALL <- glm(Henipavirus ~ ., data = data_new1, family = "binomial")
summary(logisticALL)

##
## Call:
## glm(formula = Henipavirus ~ ., family = "binomial", data = data_new1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7812  -1.2509   0.8003   0.9581   1.9992
##
## Coefficients: (9 not defined because of singularities)
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -2.878e+10  9.598e+12  -0.003   0.9976
## Samplingdate     9.940e-03  8.920e-03   1.114   0.2651
## Birthdate1-Mar   2.878e+10  9.598e+12   0.003   0.9976
## Birthdate1-Nov   2.878e+10  9.598e+12   0.003   0.9976
## Birthdate10-Sep  2.878e+10  9.598e+12   0.003   0.9976
## Birthdate11-Dec  2.878e+10  9.598e+12   0.003   0.9976
## Birthdate16-Apr  2.878e+10  9.598e+12   0.003   0.9976
## Birthdate18-Nov  2.878e+10  9.598e+12   0.003   0.9976
## CountryBioko            NA         NA      NA       NA
## CountryGhana     1.416e-01  3.424e-01   0.414   0.6791
## CountryMalawi           NA         NA      NA       NA
## CountryPrincipe         NA         NA      NA       NA
## CountrySaoTome          NA         NA      NA       NA
## CountryTanzania         NA         NA      NA       NA
## CountryUganda    2.878e+10  9.598e+12   0.003   0.9976
## SexM            -2.375e-01  1.013e-01  -2.344   0.0191 *
## AgeJ             3.715e-01  2.996e-01   1.240   0.2151
## AgeN             4.274e-01  3.080e-01   1.388   0.1652
## AgeSI           -4.901e-01  2.547e-01  -1.925   0.0543 .
## Age.3J                  NA         NA      NA       NA
## Age.3N                  NA         NA      NA       NA
## Age.3SI         -1.303e+00  3.217e-01  -4.050 5.12e-05 ***
## Age.3SI.1       -6.463e-01  2.907e-01  -2.223   0.0262 *
## Age.3SI.2       -1.507e+00  3.569e-01  -4.223 2.41e-05 ***
## Age.3SI.3               NA         NA      NA       NA
## year2008         1.638e+01  1.615e+03   0.010   0.9919
## year2009         1.268e+01  1.615e+03   0.008   0.9937
## year2010         8.998e+00  1.615e+03   0.006   0.9956
## year2011         5.852e+00  1.615e+03   0.004   0.9971
## month02         -2.437e-01  2.592e-01  -0.940   0.3472
```

```
## month03         -4.979e-01  5.268e-01  -0.945   0.3446
## month04         -4.216e-01  6.889e-01  -0.612   0.5406
## month05         -1.328e+00  1.097e+00  -1.210   0.2262
## month07         -1.861e+00  1.637e+00  -1.137   0.2554
## month08         -1.994e+00  2.108e+00  -0.946   0.3441
## month09         -2.115e+00  2.252e+00  -0.939   0.3476
## month10         -2.556e+00  2.530e+00  -1.011   0.3122
## month11         -3.254e+00  2.614e+00  -1.245   0.2132
## month12                 NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3166.0  on 2294  degrees of freedom
## Residual deviance: 2816.4  on 2265  degrees of freedom
## AIC: 2876.4
##
## Number of Fisher Scoring iterations: 21

#ROC curve of model for full set

probAll <- predict(logisticALL, data_new1, type = "response")

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

predictAll <- prediction(probAll, data_new1$Henipavirus)
perfAll <- performance(predictAll, measure = "tpr", x.measure = "fpr")

#ROC curve of model for Age.3

probAGE <- predict(logisticAGE3, data_new1, type = "response")
predictAge <- prediction(probAGE, data_new1$Henipavirus)
perfAge <- performance(predictAge, measure = "tpr", x.measure = "fpr")

plot(perfAll, col = "blue")
plot(perfAge, add = TRUE, col = "red")
```


True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate