GeminiProj

2024-05-08

```
library(ggplot2)
```

```
## Warning: package ’ggplot2’ was built under R version 4.2.3
```

```
library(dplyr)
```

```
## Warning: package ’dplyr’ was built under R version 4.2.3
```

```
##
## 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(e1071)
```

```
## Warning: package ’e1071’ was built under R version 4.2.3
```

```
data <- read.csv("text_emotion_with_gemini.csv")
head(data)
```

```
##     tweet_id sentiment        author
## 1 1956967341     empty    xoshayzers
## 2 1956969456   neutral    feinyheiny
## 3 1956971981     worry andreagauster
## 4 1956974706      hate   MavrickAces
## 5 1956977084 happiness      ktierson
## 6 1956979894   neutral  lookitsholly
##                                                                                          content
## 1 @tiffanylue i know  i was listenin to bad habit earlier and i started freakin at his part =[
## 2                                                                             cant fall asleep
## 3       @raaaaaaek oh too bad! I hope it gets better. I’ve been having sleep issues lately too
## 4         It is so annoying when she starts typing on her computer in the middle of the night!
## 5                      mmm much better day... so far! it’s still quite early. last day of #uds
## 6                          Chocolate milk is so much better through a straw. I lack said straw
##   gemini
## 1     12
## 2     11
## 3     10
## 4      1
## 5      6
## 6      2
```

```
str(data)
```

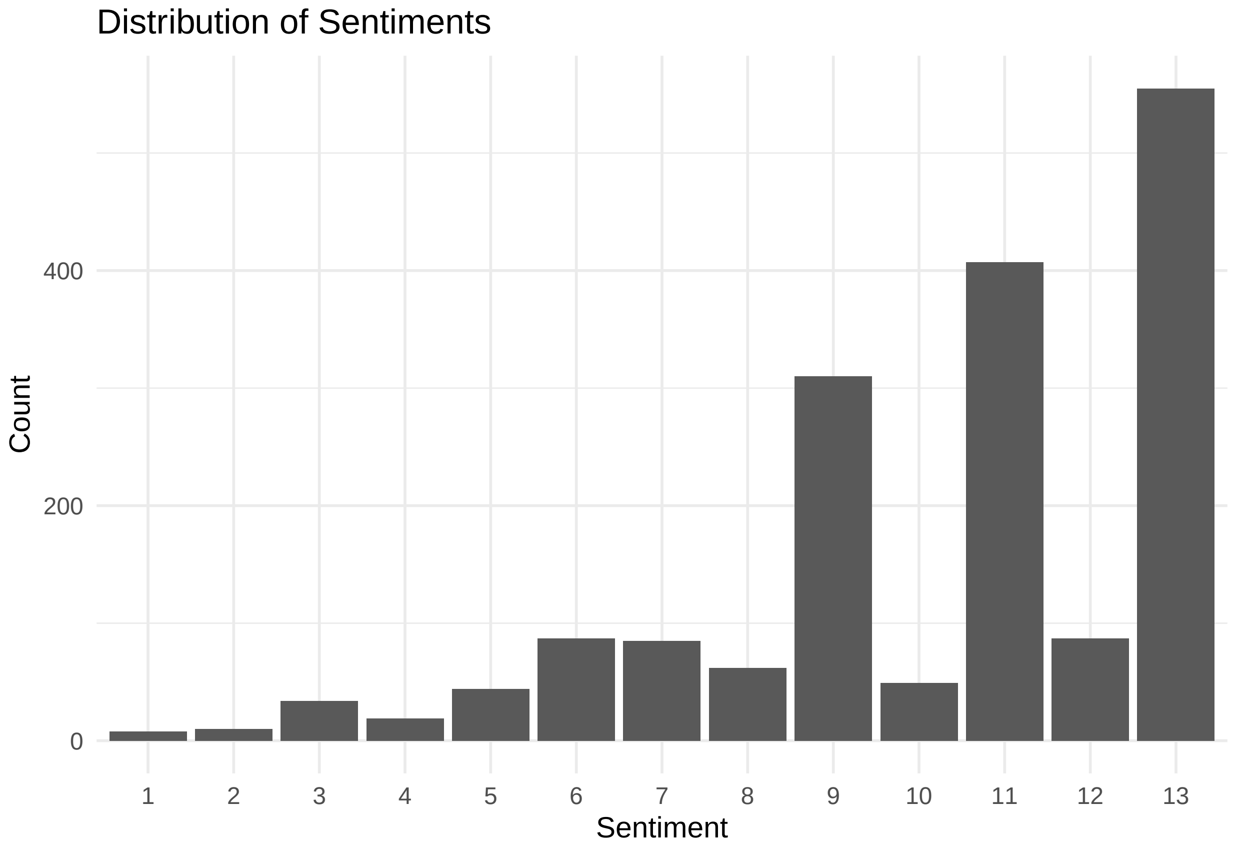
```
## 'data.frame':    1757 obs. of  5 variables:
##  $ tweet_id : int  1956967341 1956969456 1956971981 1956974706 1956977084 1956979894 1956982449 19569
##  $ sentiment: chr  "empty" "neutral" "worry" "hate" ...
##  $ author   : chr  "xoshayzers" "feinyheiny" "andreagauster" "MavrickAces" ...
##  $ content  : chr  "@tiffanylue i know  i was listenin to bad habit earlier and i started freakin at
##  $ gemini   : int  12 11 10 1 6 2 5 9 13 12 ...
```

```
sentiment_mapping <- c('anger' = 1, 'boredom' = 2, 'empty' = 3, 'enthusiasm' = 4,
                       'fun' = 5, 'happiness' = 6, 'hate' = 7, 'love' = 8,
                       'neutral' = 9, 'relief' = 10, 'sadness' = 11, 'surprise' = 12,
                       'worry' = 13)

data$sentiment <- as.integer(factor(data$sentiment, levels = names(sentiment_mapping), labels = sentimen

head(data$sentiment)
```

```
## [1]  3  9 13  7  6  9
```

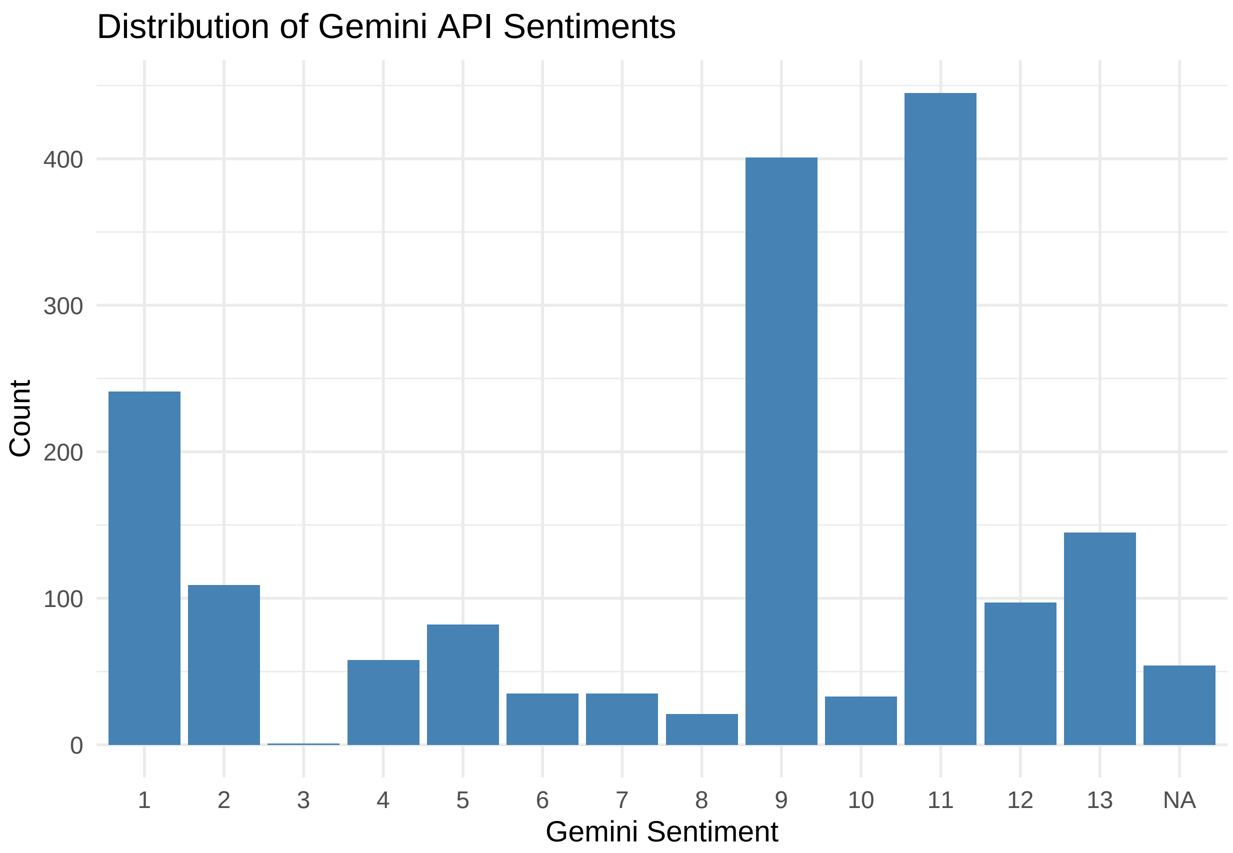
Sentiment Plot

```
sentiment_counts <- table(data$sentiment)

ggplot(data, aes(x=factor(sentiment, levels = names(sentiment_counts)))) +
  geom_bar() +
  labs(x = "Sentiment", y = "Count", title = "Distribution of Sentiments") +
  theme_minimal()
```



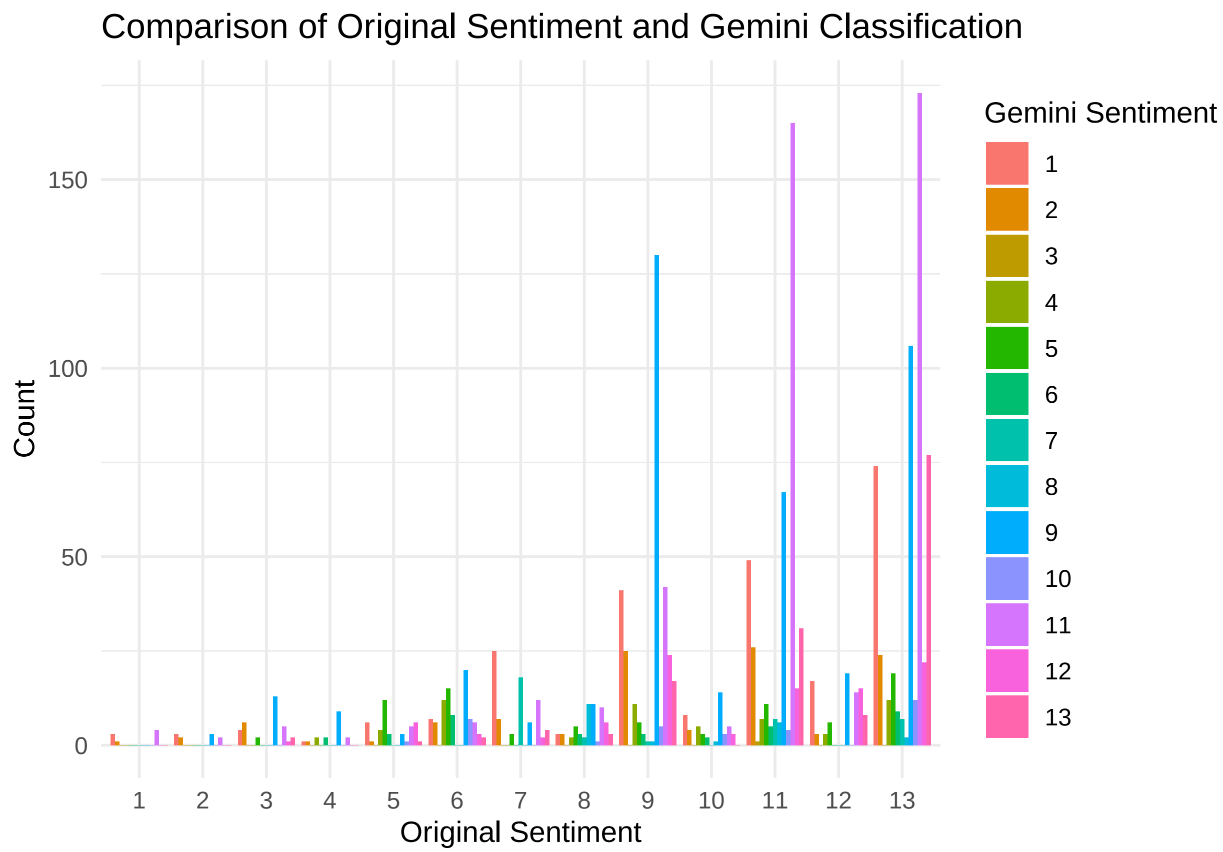
Gemini Plot

```
gemini_counts <- table(data$gemini)

ggplot(data, aes(x=factor(gemini, levels = names(gemini_counts)))) +
  geom_bar(fill = "steelblue") +
  labs(x = "Gemini Sentiment", y = "Count", title = "Distribution of Gemini API Sentiments") +
  theme_minimal()
```



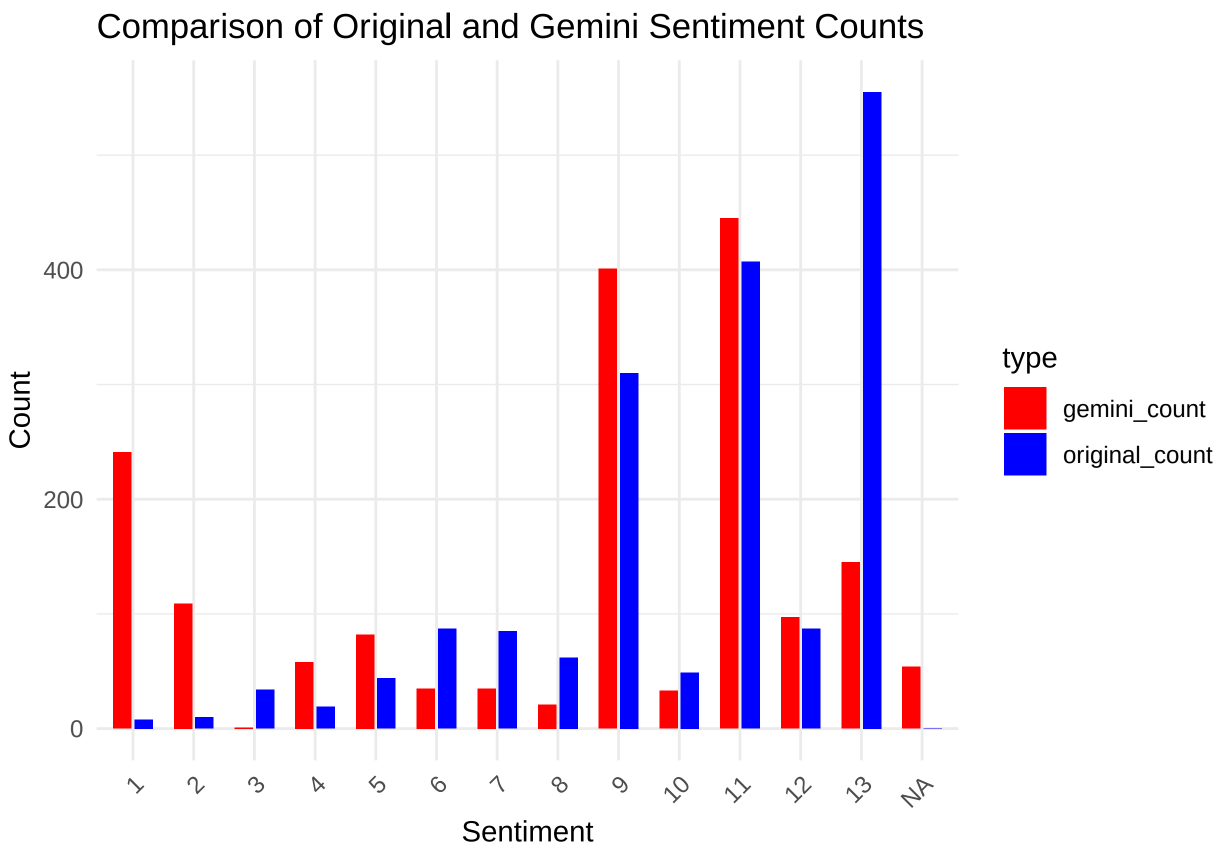
Comparison Plot

```r
cross_tab <- table(data$sentiment, data$gemini)

ggplot(as.data.frame(cross_tab), aes(Var1, Freq, fill = Var2)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Original Sentiment", y = "Count", fill = "Gemini Sentiment",
       title = "Comparison of Original Sentiment and Gemini Classification") +
  theme_minimal()
```



Comparison Plot 2

```
original_counts <- data %>%
  count(sentiment, name = "original_count")
gemini_counts <- data %>%
  count(gemini, name = "gemini_count")
levels_sentiment <- sort(as.numeric(unique(c(as.character(original_counts$sentiment), as.character(gemi
levels_sentiment <- as.character(levels_sentiment)
levels_sentiment[is.na(levels_sentiment)] <- "NA"

original_counts$sentiment <- factor(original_counts$sentiment, levels = levels_sentiment)
gemini_counts$gemini <- factor(gemini_counts$gemini, levels = levels_sentiment)

combined_counts <- full_join(original_counts, gemini_counts, by = c("sentiment" = "gemini"))
plot_data <- tidyr::pivot_longer(combined_counts, cols = c("original_count", "gemini_count"),
                                 names_to = "type", values_to = "count")
plot_data$count[is.na(plot_data$count)] <- 0

ggplot(plot_data, aes(x = sentiment, y = count, fill = type)) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.7), width = 0.6) +
  scale_fill_manual(values = c("original_count" = "blue", "gemini_count" = "red")) +
  labs(x = "Sentiment", y = "Count", title = "Comparison of Original and Gemini Sentiment Counts") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Word Cloud Prep

TEMPORARILY GREYED, NEED TO TROUBLESHOOT

```
library(wordcloud)
```

```
## Warning: package 'wordcloud' was built under R version 4.2.3
```

```
## Loading required package: RColorBrewer
```

```
#library(wordcloud)
#library(RColorBrewer)

#gemini_to_sentiment <- c('1' = "anger", '2' = "boredom", '3' = "empty", '4' = "enthusiasm",
#                          '5' = "fun", '6' = "happiness", '7' = "hate", '8' = "love",
#                          '9' = "neutral", '10' = "relief", '11' = "sadness", '12' = "surprise",
#                          '13' = "worry", 'NA' = "NA")
#data$sentiment2 <- as.character(gemini_to_sentiment[as.character(data$gemini)])
#data$sentiment2[is.na(data$sentiment2)] <- "Unknown"
#table(data$sentiment2)
library(tm)
```

```
## Warning: package 'tm' was built under R version 4.2.3
## Loading required package: NLP
```

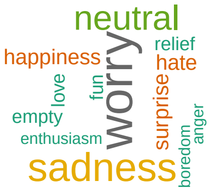
```
##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
ndat = read.csv('./text_emotion_proc_gemini.csv')
```

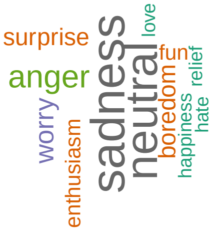
Word Cloud Sentiment

```
corpus = iconv(ndat$sentiment)
corpus = Corpus(VectorSource(corpus))


tdm <- TermDocumentMatrix(corpus)
tdm <- as.matrix(tdm)
w <- sort(rowSums(tdm), decreasing = TRUE)
wordcloud(words = names(w),
          freq = w,
          max.words = 150,
          random.order = F,
          min.freq = 5,
          colors = brewer.pal(8, 'Dark2'),
          scale = c(1.5, 0.5),
          rot.per = 0.7)
```



Word Cloud Gemini

```
corpus = iconv(ndat$gemini)
corpus = Corpus(VectorSource(corpus))


tdm <- TermDocumentMatrix(corpus)
tdm <- as.matrix(tdm)
w <- sort(rowSums(tdm), decreasing = TRUE)
wordcloud(words = names(w),
          freq = w,
          max.words = 150,
          random.order = F,
          min.freq = 5,
          colors = brewer.pal(8, 'Dark2'),
          scale = c(1.5, 0.5),
          rot.per = 0.7)
```



Regression Analysis

```
library(nnet)
data = na.omit(data)
multinom_model <- multinom(sentiment ~ gemini, data = data)
```

```
## # weights:  39 (24 variable)
## initial  value 4368.108756
## iter  10 value 3417.743867
## iter  20 value 3277.493335
## iter  30 value 3266.969667
## final  value 3266.966824
## converged
```

```
summary(multinom_model)
```

```
## Call:
## multinom(formula = sentiment ~ gemini, data = data)
##
## Coefficients:
##    (Intercept)      gemini
## 2    0.3902922 -0.02870317
## 3    1.0393401  0.05683764
## 4    0.2672892  0.07177347
## 5    1.4758613  0.02850373
## 6    2.2064932  0.02638925
## 7    2.3875383 -0.02085133
## 8    1.1501913  0.12016249
## 9    2.9783728  0.09530275
## 10   1.6159412  0.02751329
## 11   2.8535660  0.14152182
## 12   1.6380287  0.10285551
## 13   3.1027309  0.14858091
##
## Std. Errors:
##    (Intercept)     gemini
## 2    0.8086769 0.11115356
## 3    0.7059341 0.09235481
## 4    0.7917572 0.10115080
## 5    0.6794996 0.09010934
## 6    0.6477923 0.08631025
## 7    0.6455279 0.08675251
## 8    0.6859313 0.08892524
## 9    0.6290823 0.08374763
## 10   0.6717291 0.08919306
## 11   0.6292550 0.08363898
## 12   0.6619909 0.08688038
## 13   0.6263834 0.08337872
##
## Residual Deviance: 6533.934
## AIC: 6581.934
```

Linear Regression Model

```
lm_model <- lm(sentiment ~ gemini, data = data)
summary(lm_model)
```

```
##
## Call:
## lm(formula = sentiment ~ gemini, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.6216 -1.3953  0.3784  2.3784  3.5097
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.37718    0.14524  64.565  < 2e-16 ***
## gemini       0.11313    0.01625   6.963 4.75e-12 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 2.685 on 1701 degrees of freedom
## Multiple R-squared:  0.02771,    Adjusted R-squared:  0.02714
## F-statistic: 48.48 on 1 and 1701 DF,  p-value: 4.749e-12
```

Confusion Matrix Actual vs Gemini

```
library(caret) #lol it brokey bc forgor to factor
```

```
## Warning: package 'caret' was built under R version 4.2.3
```

```
## Loading required package: lattice
```

```
confusionMatrix(as.factor(data$sentiment), as.factor(data$gemini), positive = NULL, dnn = c("Prediction
```

```
## Confusion Matrix and Statistics
##
##           Gemini
## Prediction   1   2   3   4   5   6   7   8   9  10  11  12  13
##         1    3   1   0   0   0   0   0   0   0   0   4   0   0
##         2    3   2   0   0   0   0   0   0   3   0   2   0   0
##         3    4   6   0   0   2   0   0   0  13   0   5   1   2
##         4    1   1   0   2   0   2   0   0   9   0   2   0   0
##         5    6   1   0   4  12   3   0   0   3   1   5   6   1
##         6    7   6   0  12  15   8   0   0  20   7   6   3   2
##         7   25   7   0   0   3   0  18   0   6   0  12   2   4
##         8    3   3   0   2   5   3   2  11  11   1  10   6   3
##         9   41  25   0  11   6   3   1   1 130   5  42  24  17
##         10   8   4   0   5   3   2   0   1  14   3   5   3   0
##         11  49  26   1   7  11   5   7   6  67   4 165  15  31
##         12  17   3   0   3   6   0   0   0  19   0  14  15   8
##         13  74  24   0  12  19   9   7   2 106  12 173  22  77
##
## Overall Statistics
##
##                Accuracy : 0.2619
##                  95% CI : (0.2411, 0.2835)
##     No Information Rate : 0.2613
##     P-Value [Acc > NIR] : 0.4873
##
##                   Kappa : 0.1437
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: 1 Class: 2  Class: 3 Class: 4 Class: 5 Class: 6
## Sensitivity          0.012448 0.018349 0.0000000 0.034483 0.146341 0.228571
## Specificity          0.996580 0.994981 0.9806110 0.990881 0.981493 0.953237
## Pos Pred Value       0.375000 0.200000 0.0000000 0.117647 0.285714 0.093023
## Neg Pred Value       0.859587 0.936799 0.9994012 0.966785 0.957857 0.983302
## Prevalence           0.141515 0.064005 0.0005872 0.034058 0.048150 0.020552
## Detection Rate       0.001762 0.001174 0.0000000 0.001174 0.007046 0.004698
## Detection Prevalence 0.004698 0.005872 0.0193776 0.009982 0.024662 0.050499
## Balanced Accuracy    0.504514 0.506665 0.4903055 0.512682 0.563917 0.590904
##                      Class: 7 Class: 8 Class: 9 Class: 10 Class: 11 Class: 12
## Sensitivity           0.51429 0.523810  0.32419  0.090909   0.37079  0.154639
## Specificity           0.96463 0.970868  0.86482  0.973054   0.81797  0.956413
## Pos Pred Value        0.23377 0.183333  0.42484  0.062500   0.41878  0.176471
## Neg Pred Value        0.98954 0.993914  0.80601  0.981873   0.78610  0.949320
## Prevalence            0.02055 0.012331  0.23547  0.019378   0.26130  0.056958
## Detection Rate        0.01057 0.006459  0.07634  0.001762   0.09689  0.008808
## Detection Prevalence  0.04521 0.035232  0.17968  0.028186   0.23136  0.049912
## Balanced Accuracy     0.73946 0.747339  0.59451  0.531981   0.59438  0.555526
##                      Class: 13
## Sensitivity            0.53103
## Specificity            0.70475
## Pos Pred Value         0.14339
## Neg Pred Value         0.94168
## Prevalence             0.08514
## Detection Rate         0.04521
## Detection Prevalence   0.31533
## Balanced Accuracy      0.61789
```