M8L2 Homework Assignment

Joshua Conte

November 26, 2017

1 M8L2 Homework Assignment

R studio was configured with the following parameters before beginning the project:

```
# clears the console in RStudio
cat("\014")
```

```
# clears environment
rm(list = ls())

# Load required packages
library(RCurl)
library(plyr)
library(forecast)
```

1.1 Load the Data

I will be using an output file from my project HGMD_final_CHD2.csv. This data contains the publication year for mutations in the CHD2 gene. I am going to use this data to predict futur publications for mutations in this gene.

```
dfHGMD_final <- read.csv2(
        "HGMD_final_CHD2.csv",
        sep = ",",
        stringsAsFactors = FALSE
      )
```

1.1.1 Formating the data

Inorder to analyze the data I need numeric data. I will make new columns with Consequence, Variantclass, Reported.phenotype, Mutation.type, Overall.consequence, and year converted to numeric.

For Consequence the data will change to: Create column and assign a Consequence with a num Ser112Term = 1, Arg121Term = 2, Pro218Leu = 3, Arg466Term = 4, Leu823Pro = 5, Asp856Gly = 6, Gln906Term = 7, Gly1174Asp = 8, Arg1313Gly = 9, Arg1345Gln = 10, Ser1406Gly = 11, Trp1534Cys = 12, Arg1637Term = 13, Gln1641Term = 14, Trp1657Term = 15, Arg1679Term = 16, c.390C>T = 17, c.1502+1G>A = 18, c.1719+5G>A = 19, c.1810-2A>C = 20, c.1552delC = 21, c.1809+1delG = 22, c.1880_1883delCTTT = 23, c.2895_2898delAGAA = 24, c.3734delA = 25, c.4233_4236delAGAA = 26, c.4256_4274del19 = 27, c.3787dupG = 28, c.4173dupA = 29, c.4949dupG = 30, c.5094dupG = 31

```
# Make Consequence numeric
dfHGMD_final$Consequence.num <- 1:nrow(dfHGMD_final)
```

For Reported Phenotype the data will change to: Autism = 1 Epilepsy = 2 Eyelid myoclonia = 3 Intellectual disability = 4

```
# Make Reported.phenotype numeric
# Assign number to Reported.phenotype
# Create column
dfHGMD_final$Reported.phenotype.num <-
  dfHGMD_final$Reported.phenotype

# Use regex to make data numeric
# Autism = 1
dfHGMD_final$Reported.phenotype.num <-
```

```
  gsub("^Autism.*",
       dfHGMD_final$Reported.phenotype.num,
       replacement = "1")

# Epilepsy = 2
dfHGMD_final$Reported.phenotype.num <-
  gsub("^Epilep.*|^Dravet.*|^Lennox-Gastaut.*",
       dfHGMD_final$Reported.phenotype.num,
       replacement = "2")

# Eyelid myoclonia = 3
dfHGMD_final$Reported.phenotype.num <-
  gsub("^Eyelid.*",
       dfHGMD_final$Reported.phenotype.num,
       replacement = "3")

# Intellectual disability = 4
dfHGMD_final$Reported.phenotype.num <-
  gsub("^Intellectual.*|.*intellectual disability.*",
       dfHGMD_final$Reported.phenotype.num,
       replacement = "4")
```

For Variant Class the data will change to: DM = 1 DM? = 2

```
# Make Variantclass numeric
# Assign number to Variantclass.num
dfHGMD_final$Variantclass.num<-dfHGMD_final$Variantclass
dfHGMD_final$Variantclass.num<-ifelse(dfHGMD_final$Variantclass.num=="DM", 1, 2)
```

For Overall.consequence the data will change to: LoF = 1 nLoF = 2

```
# Make Overall.consequence numeric
# Assign number to Overall.consequence.num
dfHGMD_final$Overall.consequence.num <-
dfHGMD_final$Overall.consequence
dfHGMD_final$Overall.consequence.num <-
ifelse(dfHGMD_final$Overall.consequence.num == "LoF", 1, 2)
```

For Mutation.type the data will change to: frameshift = 1 Missense = 2 Nonsense = 3 splice = 4

```
# Make Reported.phenotype numeric
# Assign number to Reported.phenotype
# Create column
dfHGMD_final$Mutation.type.num <-
  dfHGMD_final$Mutation.type

# Use regex to make data numeric
# frameshift = 1
dfHGMD_final$Mutation.type.num <-
  gsub("^frameshift",
       dfHGMD_final$Mutation.type.num,
       replacement = "1")

# Missense = 2
dfHGMD_final$Mutation.type.num <-
  gsub("^Missense",
```

```
       dfHGMD_final$Mutation.type.num,
       replacement = "2")

# Nonsense = 3
dfHGMD_final$Mutation.type.num <-
  gsub("^Nonsense",
       dfHGMD_final$Mutation.type.num,
       replacement = "3")

# Splice = 4
dfHGMD_final$Mutation.type.num <-
  gsub("^splice",
       dfHGMD_final$Mutation.type.num,
       replacement = "4")
```

Change data to numeric and make new dataframe:

```
# Convert to numeric
dfHGMD_final[8] <- sapply(dfHGMD_final[8], as.numeric)
dfHGMD_final[14:18] <- sapply(dfHGMD_final[14:18], as.numeric)


# Columns to keep
keep <-
  c(
    "Consequence.num",
    "Reported.phenotype.num",
    "Variantclass.num",
    "Overall.consequence.num",
    "Mutation.type.num",
    "Year")

# Make new dataframe with keep data
data <- dfHGMD_final[keep]

# Check dataframe
str(data)
```

```
## 'data.frame':    31 obs. of  6 variables:
##  $ Consequence.num        : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ Reported.phenotype.num : num  4 2 3 2 2 1 1 1 2 2 ...
##  $ Variantclass.num       : num  1 1 2 1 1 2 2 2 1 1 ...
##  $ Overall.consequence.num: num  1 1 2 2 2 2 2 2 2 2 ...
##  $ Mutation.type.num      : num  3 3 2 2 2 2 2 2 2 2 ...
##  $ Year                   : num [1:31, 1] 2014 2013 2015 2013 2013 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr "Year"
```

```
summary(data)
```

```
##  Consequence.num Reported.phenotype.num Variantclass.num
##  Min.   : 1.0    Min.   :1.000          Min.   :1.000
##  1st Qu.: 8.5    1st Qu.:1.500          1st Qu.:1.000
##  Median :16.0    Median :2.000          Median :1.000
```

```
##  Mean   :16.0    Mean   :2.097          Mean   :1.484
##  3rd Qu.:23.5    3rd Qu.:2.000          3rd Qu.:2.000
##  Max.   :31.0    Max.   :4.000          Max.   :2.000
##  Overall.consequence.num Mutation.type.num      Year.Year
##  Min.   :1.000           Min.   :1.000     Min.   :2012.0000
##  1st Qu.:1.000           1st Qu.:1.000     1st Qu.:2013.0000
##  Median :1.000           Median :2.000     Median :2015.0000
##  Mean   :1.323           Mean   :2.194     Mean   :2014.5806
##  3rd Qu.:2.000           3rd Qu.:3.000     3rd Qu.:2016.0000
##  Max.   :2.000           Max.   :4.000     Max.   :2017.0000
```

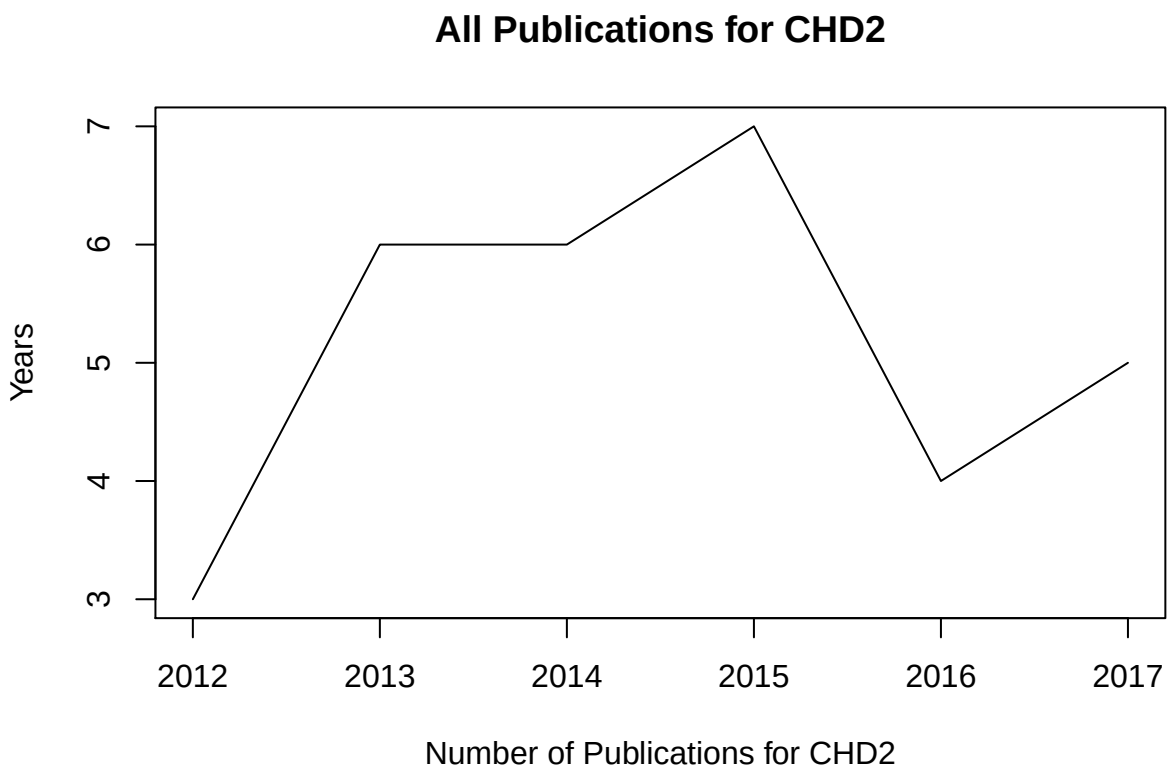
1.2 Time series analysis in R

The ts() function will convert a numeric vector into an R time series object. The format is ts(vector, start=, end=, frequency=) where start and end are the times of the first and last observation and frequency is the number of observations per unit time (1=annual, 4=quartly, 12=monthly, etc.).

```
HGMD <- count(data, c("Year"))
HGMD
```

```
##   Year freq
## 1 2012    3
## 2 2013    6
## 3 2014    6
## 4 2015    7
## 5 2016    4
## 6 2017    5
```

```
# set the freq parameter to 1 to indicate annual readings
# ts() function to create a new time series
HGMD_timeseries <- ts(HGMD$freq, start = c(min(data$Year), 1), end = c(max(data$Year), 1), frequency = 
plot(HGMD_timeseries, xlab='Number of Publications for CHD2' ,ylab='Years', main='All Publications for 
```




```
## -------------   USE ARIMA MODEL  ---------------------
#creating ranges of possible values for the order parameters p, d, and q.
d <- 0 : 1
p <- 0 : 1
q <- 0 : 1
HGMD_models <- expand.grid(d = d, p = p, q = q)
head(HGMD_models, n = 4)
```

```
##   d p q
## 1 0 0 0
## 2 1 0 0
## 3 0 1 0
## 4 1 1 0
```

```
getTSModelAIC <- function(ts_data, p, d, q) {
                                              ts_model <- arima(ts_data, order = c(p, d, q))
                                                return(ts_model$aic)
                                                }


getTSModelAICSafe <- function(ts_data, p, d, q) {
                                                    result = tryCatch({
                                                    getTSModelAIC(ts_data, p, d, q)
                                                     }, error = function(e) {
                                                    Inf
                                                    })
                                                    }
```

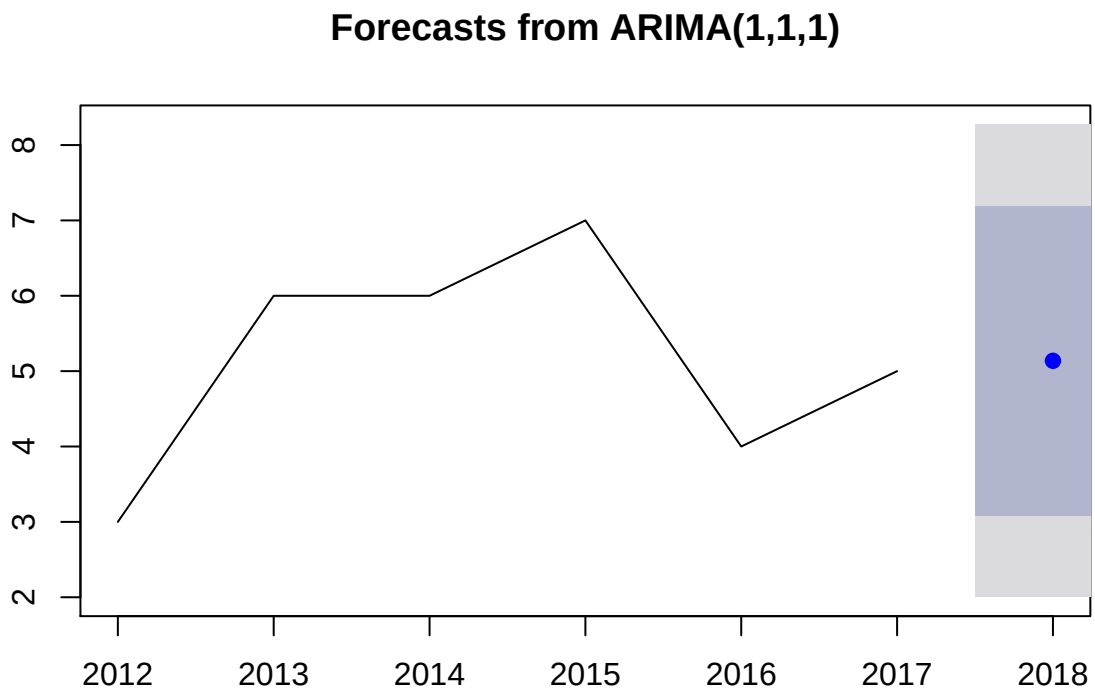
```
 # PICK THE BEST MODEL THAT HAS THE SMALLEST AIC
HGMD_models$aic <- mapply(function(x, y, z)
                            getTSModelAICSafe(HGMD_timeseries, x, y, z), HGMD_models$p,
                            HGMD_models$d, HGMD_models$q)

subset(HGMD_models,aic == min(aic))
```

```
##   d p q      aic
## 2 1 0 0 23.12086
```

```
# ARIMA model for best p,d,q order model
HGMD_model <- arima(HGMD_timeseries, order = c(1, 1, 1))
summary(HGMD_model)
```

```
##
## Call:
## arima(x = HGMD_timeseries, order = c(1, 1, 1))
##
## Coefficients:
##          ar1      ma1
##       0.0533  -1.0000
## s.e.  0.6552   0.7046
##
## sigma^2 estimated as 2.202:  log likelihood = -9.92,  aic = 25.84
##
## Training set error measures:
##                      ME     RMSE     MAE      MPE     MAPE      MASE
## Training set 0.5935149 1.354689 1.09265 7.219212 19.54428 0.6829063
##                      ACF1
## Training set -0.02676874
```

```
#------------------       Prediction    --------------------------------
plot(forecast(HGMD_model, 1))
```



I used a frequency of 1, to look at the data annually. I did this because I only have the publication year not the month and year. The data looks good, I played the the parameters so it would not produce NANs and produce a low AIC. This model predicts that there will be another 5 publications for CHD2 next year, based off of past performance.