Analytics Modeling HW3

Fall 2024

Question 5.1

Using the crime dataset, test to see whether there are any outliers in the last column (number of crimes per 100k people). Use the `grubbs.test` function in the outliers package of R.

The crime dataset contains data on the effect of punishment regimes on crime rates and includes the columns:

- `M` : percentage of males aged 14-24 in total state population
- `So` : indicator for a southern state
- `Ed` : mean years of schooling of the population aged 25 and up
- `Po1` : per capita expenditure on police protection in 1960
- `Po2` : per capita expenditure on police protection in 1959
- `LF` : labor force rate of civilian urban males aged 14-24
- `M.F` : the number of males per 100 females
- `Pop` : the state population in 1960 in hundreds of thousands
- `U1` : the unemployment rate of urban males aged 14-24
- `U2` : the unemployment rate of urban males aged 35-39
- `Wealth` : the median value of transferable assets of family income
- `Ineq` : income inequality; the percentage of families earning below half the median income
- `Prob` : the probability of imprisonment; the ratio of number of commitments to number of offenses
- `Time` : the average time in months served by offenders in state prisons before their first release
- `Crime` : the crime rate; number of offenses per 100k population in 1960

So this dataset appears to largely focus on male criminal offenders, which may be a hidden source of bias.

The Grubbs Test is a method to identify outliers in univariate data that involves quantifying how far away a datapoint is from other values using the Normal Distribution. The test statistic Z is calculated from the most extreme datapoint and the test statistic corresponds to a p-value that represents the likelihood of seeing that outlier.

Z = |mean-datapoint| / standard deviation

The Null Hypothesis and Alternative Hypothesis for Grubbs Test are as follows:

H_0: there are no outliers in the dataset

H_a: there is an outlier in the dataset

There is quite a range in the `Crime` column, from 342.0 to 1993.0. Additionally, the difference between the mean and the median (`905.1 - 831.0 = 74.1`) is large enough to make me believe that there likely are outliers in this column that are pulling the mean higher. Graphing this data with a boxplot, qqplot, and histogram gives me further reason to believe there are outliers this column.

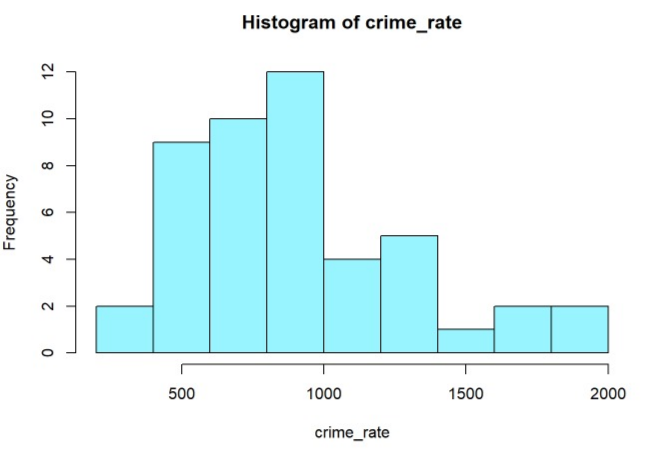
```
# load data
crime <- read.table("uscrime.txt",  stringsAsFactors = FALSE, header = TRUE)
head(crime, 5)
```

```
##      M So   Ed  Po1  Po2    LF   M.F Pop   NW    U1  U2 Wealth Ineq     Prob
## 1 15.1  1  9.1  5.8  5.6 0.510  95.0  33 30.1 0.108 4.1   3940 26.1 0.084602
## 2 14.3  0 11.3 10.3  9.5 0.583 101.2  13 10.2 0.096 3.6   5570 19.4 0.029599
## 3 14.2  1  8.9  4.5  4.4 0.533  96.9  18 21.9 0.094 3.3   3180 25.0 0.083401
## 4 13.6  0 12.1 14.9 14.1 0.577  99.4 157  8.0 0.102 3.9   6730 16.7 0.015801
## 5 14.1  0 12.1 10.9 10.1 0.591  98.5  18  3.0 0.091 2.0   5780 17.4 0.041399
##      Time Crime
## 1 26.2011   791
## 2 25.2999  1635
## 3 24.3006   578
## 4 29.9012  1969
## 5 21.2998  1234
```

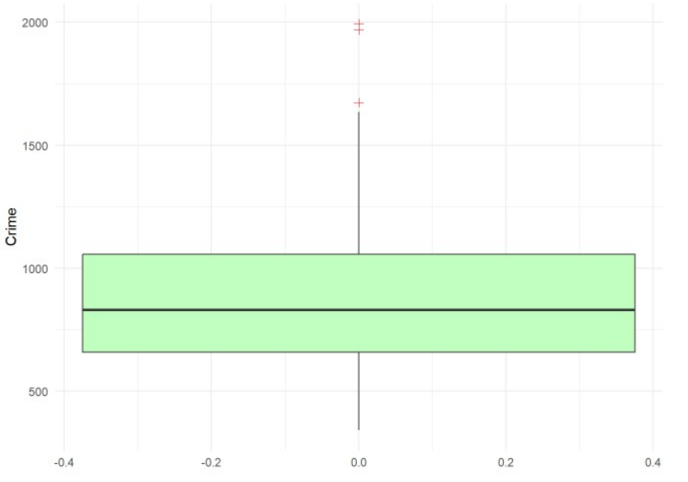
```
crime_rate <- crime$Crime           # grab just the last column

# plotting
summary(crime_rate)                 # get summary statistics on the column
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   342.0   658.5   831.0   905.1  1057.5  1993.0
```

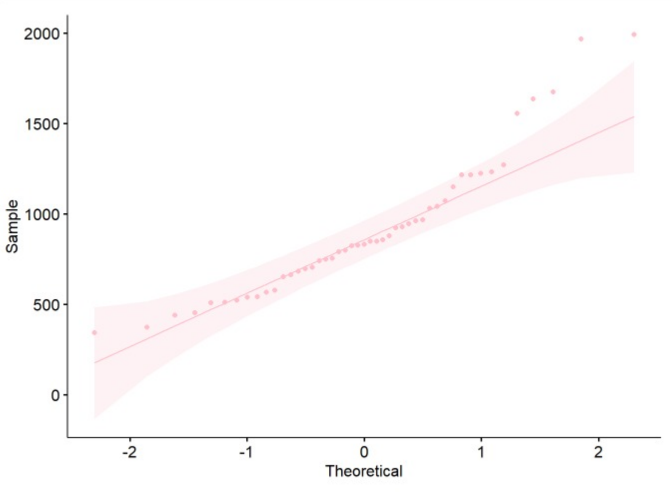
```
hist(crime_rate, col = "cadetblue1")  # plot a histogram of the crime column
```




```
ggplot(crime, aes(y=Crime)) +         # plot a boxplot to observe potential outliers that way
  geom_boxplot(outlier.colour="red"
              , outlier.shape=3
              , outlier.size=2
              , fill = "darkseagreen1"
              ) +
  theme_minimal()
```




```
ggqqplot(crime                      # plot a qqplot
        , x="Crime"
        , color = "pink")
```




```
# run the test, testing for two outliers
gtest <- grubbs.test(x=crime_rate, type=10)
gtest
```

```
##
##  Grubbs test for one outlier
##
## data:  crime_rate
## G = 2.81287, U = 0.82426, p-value = 0.07887
## alternative hypothesis: highest value 1993 is an outlier
```

Since the p-value is greater than 0.05, the Null Hypothesis that the dataset has no outliers can't be rejected. This indicates that the potential outliers seen in the graphs may have been caused by randomness.

Question 6.1

Describe a situation or problem from your job, everyday life, current events, etc., for which a Change Detection model would be appropriate. Applying the CUSUM technique, how would you choose the critical value and the threshold?

I work for a sports & casino gambling company as a data scientist. One problem at work for which a Change Detection model would be appropriate is for Customer Risk Scores. Customers are assigned a value between [0,1] indicating how risky they are to have as a gambling customer based off their past betting behaviors. Players who are sharks or who are cheaters will have a higher risk score than an average player.

As an example, let's say an average player would have a risk score of 0.5, a very successful player may have a risk score of 0.75, and a cheater's score could be all the way up to 1.0.

If we want to identify players that are trending towards higher and higher risk scores, an appropriate threshold could be 0.75 (to start identifying sharks). The threshold could also be raised to 0.80 or higher to strictly identify potentially fraudulent players. Since the scale of the CRS variable is quite small, the critical value `C` would need to be relatively small as well or else the model would have an issue with a lot of false positives. Perhaps an initial critical value of 0.01 could be used and fine tuned for a more effective model.

Question 6.2.1

Using the Atlanta temperature 1996-2015 dataset, use a CUSUM approch to identify when unofficial summer ends (when the temperature starts cooling off) each year.

I decided to try to do all of this in R to get more practice with it. I grabbed the dataset and feature engineered what I needed to do a CUSUM analysis:

- `dates` : the dates of the data
- `mu` : the mean temperature across dates
- `xi` : the observed average temperature across all years for each date
- `iDiff` : the calculation for increasing differences
- `dDiff` : the calculation for decreasing differences
- `increase` : the cumulative sum of increasing differences
- `decrease` : the cumulative sum of decreasing differences
- `iChange` : a boolean marking rows where an increase has been identified
- `dChange` : a boolean marking rows where a decrease has been identified

I wanted to check for changes in both directions, increasing and decreasing, so I calculated both metrics at the same time.

After observing how the data was behaving with this model, I set my critical value to five and threshold to 30. The goal was to not generate many false positives, as I would consider that a greater problem for the model, but also not be so insensitive that the model would be slow to identify changes.

For when the temperature changes each year, with the threshold and critical value I used, the model identified mid-October as the turning point from Summer to Fall in Atlanta (October 13th).

```
# load data
temps <- read.table("temps.txt",  stringsAsFactors = FALSE, header = TRUE)
# remove weird X on column names
names(temps) <- gsub(x=names(temps), pattern = "X", replacement = "")

C <- 5                                  # Critical Value
T <- 30                                 # Threshold

dates <- temps[,1]                      # get list of dates
mu <- mean(as.matrix(temps[,-1]))       # get the mean of all the temperatures
xi <- rowMeans(as.matrix(temps[,-1]))   # get the average temperature for each day
iDiff <- xi-mu-C                        # check for increasing difference
dDiff <- mu-xi-C                        # check for decreasing difference


cusum <- data.frame(dates               # create table to store the cusum data
                    , xi
                    , mu
                    , iDiff
                    , dDiff
                    )

# calculate CUSUM metric, but set to zero if the metric is less than zero
cusum <- cusum %>% mutate(increase = accumulate(iDiff,  ~ ifelse(.x + .y < 0, 0, .x + .y)))
cusum <- cusum %>% mutate(decrease = accumulate(dDiff,  ~ ifelse(.x + .y < 0, 0, .x + .y)))
#cusum$decrease <- temp_dec$decrease   # add decreasing metric to cusum table

# if the metric >= T, mark TRUE
cusum$iChange <- ifelse(cusum$increase>=T, TRUE, FALSE)
cusum$dChange <- ifelse(cusum$decrease>=T, TRUE, FALSE)

# get all rows after the first increase change has been identified
increase_identified <- cusum[which(cusum$iChange == TRUE),]
decrease_identified <- cusum[which(cusum$dChange == TRUE),]

head(cusum, 5)
```

```
##   dates    xi       mu       iDiff      dDiff  increase  decrease iChange
## 1 1-Jul 88.85 83.33902  0.51097561 -10.510976 0.5109756 -10.51098   FALSE
## 2 2-Jul 88.35 83.33902  0.01097561 -10.010976 0.5219512   0.00000   FALSE
## 3 3-Jul 88.40 83.33902  0.06097561 -10.060976 0.5829268   0.00000   FALSE
## 4 4-Jul 88.35 83.33902  0.01097561 -10.010976 0.5939024   0.00000   FALSE
## 5 5-Jul 88.25 83.33902 -0.08902439  -9.910976 0.5048780   0.00000   FALSE
##   dChange
## 1   FALSE
## 2   FALSE
## 3   FALSE
## 4   FALSE
## 5   FALSE
```

```
tail(cusum, 5)
```

```
##      dates    xi       mu     iDiff    dDiff increase decrease iChange dChange
## 119 27-Oct 68.90 83.33902 -19.43902 9.439024        0 125.7817   FALSE    TRUE
## 120 28-Oct 68.60 83.33902 -19.73902 9.739024        0 135.5207   FALSE    TRUE
## 121 29-Oct 69.35 83.33902 -18.98902 8.989024        0 144.5098   FALSE    TRUE
## 122 30-Oct 71.05 83.33902 -17.28902 7.289024        0 151.7988   FALSE    TRUE
## 123 31-Oct 70.50 83.33902 -17.83902 7.839024        0 159.6378   FALSE    TRUE
```

```
# dates identified with increased changes
increase_identified$dates[increase_identified=TRUE]
```

```
## [1] "20-Aug" "21-Aug" "22-Aug" "23-Aug" "24-Aug" "25-Aug" "26-Aug"
```

```
# dates identified with decreased changes
decrease_identified$dates[decrease_identified=TRUE]
```

```
##  [1] "13-Oct" "14-Oct" "15-Oct" "16-Oct" "17-Oct" "18-Oct" "19-Oct" "20-Oct"
##  [9] "21-Oct" "22-Oct" "23-Oct" "24-Oct" "25-Oct" "26-Oct" "27-Oct" "28-Oct"
## [17] "29-Oct" "30-Oct" "31-Oct"
```

Question 6.2.2

Using the temperature dataset, use a CUSUM approch to make a judgment of whether Atlanta's summer climate has gotten warmer in that time (and if so, when).

If we look at the data over the years (via the violin plots), there does appear to be a trend of temperatures increasing in some sense. The temperature fluctuations appear to be more erratic as time goes on, with sharper jumps up and down in temperature over time.

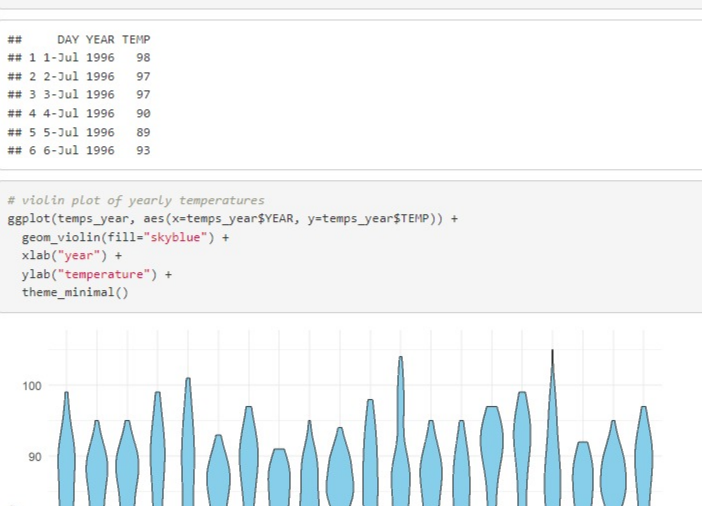
This time instead of taking the average temperature by month, I'm looking at the data over each year to get an idea of how average temperatures in the summer have changed over time. With a smaller dataset to work with and with each datapoint representing more months of time, I have to lower the threshold and critical value so they do not completely swamp the datapoints.

This model detects that there has been an increased change in Atlanta's summer temperatures, with the first change identified in 2010.

```
# make dataset long
temps_year <- melt(temps
                  , id.vars = c("DAY")
                  , variable.name= "YEAR"
                  , value.name="TEMP")
head(temps_year)
```

```
##     DAY YEAR TEMP
## 1 1-Jul 1996   98
## 2 2-Jul 1996   97
## 3 3-Jul 1996   97
## 4 4-Jul 1996   90
## 5 5-Jul 1996   89
## 6 6-Jul 1996   93
```

```
# violin plot of yearly temperatures
ggplot(temps_year, aes(x=temps_year$YEAR, y=temps_year$TEMP)) +
  geom_violin(fill="skyblue") +
  xlab("year") +
  ylab("temperature") +
  theme_minimal()
```




```
C2 <- .5          # Critical Value
T2 <- 3           # Threshold

# get yearly average temps
cusum2 <- setNames(aggregate(temps_year$TEMP, list(temps_year$YEAR), FUN=mean),c('year', 'xi'))

cusum2$mu <- mean(as.matrix(cusum2[,2]))     # overall mean
cusum2$iDiff <- cusum2$xi-cusum2$mu-C2       # check for increasing difference

# calculate CUSUM metric, but set to zero if the metric is less than zero
cusum2 <- cusum2 %>% mutate(increase = accumulate(iDiff,  ~ ifelse(.x + .y < 0, 0, .x + .y)))

# if increasing CUSUM metric >= T, mark TRUE
cusum2$iChange <- ifelse(cusum2$increase>=T2, TRUE, FALSE)
cusum2
```

```
##    year       xi       mu      iDiff   increase iChange
## 1  1996 83.71545 83.33902 -0.1235772 -0.1235772   FALSE
## 2  1997 81.67480 83.33902 -2.1642276  0.0000000   FALSE
## 3  1998 84.26016 83.33902  0.4211382  0.4211382   FALSE
## 4  1999 83.35772 83.33902 -0.4813008  0.0000000   FALSE
## 5  2000 84.03252 83.33902  0.1934959  0.1934959   FALSE
## 6  2001 81.55285 83.33902 -2.2861789  0.0000000   FALSE
## 7  2002 83.58537 83.33902 -0.2536585  0.0000000   FALSE
## 8  2003 81.47967 83.33902 -2.3593496  0.0000000   FALSE
## 9  2004 81.76423 83.33902 -2.0747967  0.0000000   FALSE
## 10 2005 83.35772 83.33902 -0.4813008  0.0000000   FALSE
## 11 2006 83.04878 83.33902 -0.7902439  0.0000000   FALSE
## 12 2007 85.39837 83.33902  1.5593496  1.5593496   FALSE
## 13 2008 82.51220 83.33902 -1.3268293  0.2325203   FALSE
## 14 2009 80.99187 83.33902 -2.8471545  0.0000000   FALSE
## 15 2010 87.21138 83.33902  3.3723577  3.3723577    TRUE
## 16 2011 85.27642 83.33902  1.4373984  4.8097561    TRUE
## 17 2012 84.65041 83.33902  0.8113821  5.6211382    TRUE
## 18 2013 81.66667 83.33902 -2.1723577  3.4487805    TRUE
## 19 2014 83.94309 83.33902  0.1040650  3.5528455    TRUE
## 20 2015 83.30081 83.33902 -0.5382114  3.0146341    TRUE
```