Chocolate Bar Ratings – Analysis and Regression

Made by majafoi

UPDATED 4/8/2021

```
CBR <- read.csv ("flavors_of_cacao.csv")
cs = complete.cases (CBR) #since there is a possibility that this original dataset has some missing data or NAs (not available data), I will use the function complete.cases in order to remove that part of the dataset.
CBR = CBR [cs,]
attach (CBR) #I will only work with a complete dataset, or dataset without missing data and NAs.
dfrm <- as.data.frame (CBR)
colnames (dfrm) <- c ("company","specific_bean_origin","REF","renew_date","cocoa_percentage","company_location","rating","bean_type","broad_bean_origin")
attach (dfrm)
```

Hello, everyone!

This time, I'm going to make an analysis of a dataset called Chocolate Bar Ratings.

We all know that chocolate is one of the favorite food in the whole world, possibly even yours, so you must've know that every chocolate has its own flavor and texture. People like Brady Brelinski, Founding Member of the Manhattan Chocolate Society (Flavors of Cacao) have compiled ratings of some of the variables of cocoa or chocolate.

The database I found on Kaggle website (https://www.kaggle.com/rtatman/chocolate-bar-ratings) (there is an updated version on the original page - http://flavorsofcacao.com/chocolate_database.html), focuses only on plain dark chocolate.

The variables that are being chosen here for analysis are:

1) Company (maker-if known) - name of the company manufacturing the bar
2) specific bean origin or bar name - the specific geo-region of origin for the bar
3) REF - a value linked to when the review was entered in the database. The higher the value - the more recent it is.
4) Review Data - date of publication of the review
5) Cocoa Percent - cocoa percentage (darkness of the chocolate) of the chocolate bar being reviewed - I changed it into relative numbers (60% = 0,60)
6) Company Location - manufacturer base country
7) Rating - expert rating for the bar
8) Bean Type - the variety of bean used, if provided
9) Broad Bean Origin - the broad geo-region of origin for the bean.

Since the names of the variables are a bit long, I have narrowed them down with colnames code above. This way, the new variables names are still recognizable, but easier to work with. Every variable is pretty much straightforward, except maybe the Rating variable.

When talking about rating, the system is following, and it concerns the flavor:

5 = Elite 4 = Premium 3 = Satisfactory 2 = Disappointing 1 = Unpleasant/unpalatable

As in all my coding and PDFs, I am trying to do my best to explain the findings, and the code itself, in a way that the non-data-scientists people understand it. There are going to be talking about normal distribution, regression analysis, EDA, and all other *fancy* words that might seem intimidating, but they aren't if explained neat and simple.

So, let's start with the basic table of content of what we are going to do here.

1) Summary of variables & graphics 1a) FIVENUM 1b) Graphics - histograms, boxplot, scatterplots…
2) Correlation analysis
   2a) Correlation between numeric variables
   2b) Correlation between non-numeric variables (checking the connection)
3) Regression analysis

Most part of this table of content is not very easy to understand, but I'll explain every step of the way. So, let's start!

Summary of variables

In order to make summaries of variables, we have two possibilities, and both are very frequently used in R programming or Statistics. First, we can aggregate the numeric variables and use a FIVENUM method. The second possibility is to make graphs to neatly show how the variables are evolving and which shape they take. In 99% of the cases, one can't go without the other.

FIVENUM method

FIVENUM is just that - the most significant 5 numbers that explain the distribution of a numeric variable. Those five numbers are:

1) minimum - minimal value in the distribution of that variable.
2) first quartile - (1st Qu.) - a value that separates the first 25% of the distribution (onto 25% and the remaining 75%)
3) median - a value that separates a distribution into 50% of the distribution (half)
4) Mean - average value
5) third quartile (3st Qu.) - a value that separates the last 25% of the distribution (onto 75% and the remaining last 25%).
6) maximum - the maximum value in the distribution of that variable

The only significant value that is missing is SD or standard deviation. In statistics, that is a measure or an amount of variation/dispersion of a set of values in one distribution, from the mean (which is sometimes called expected value). It basically means how much is the data dispersed from the average value. We can calculate it by a separate function.

The numeric variables here in the dataset that need summaries are Cocoa percent and rating, as the REF and ReviewData variables aren't really significant here.

By default, the variable cocoa_percentage class was character, not numeric as it should be. The reason for that was because there was a "%" sign in the variable result. I decided to change the class in Excel, as you can open CSV file in the program, and now the numbers are in relative, and numeric. Now, we can proceed with the summaries.

```
Summary (cocoa_percentage)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.500   0.700   0.700   0.719   0.750   1.000

sd (cocoa_percentage)

## [1] 0.06116981
```

Let's repeat - cocoa percentage is the darkness of the chocolate bar being reviewed. When looking at the summary, we can see that the minimal value is 50%, whereas the maximum is 100%. The median and the mean are close together, around 70%, with the standard deviation of 6%. All these numbers are hard to understand, if you aren't a data scientist. So, let's explain. Minimal and maximum values are pretty straightforward, so I won't explain them. You could see that I was comparing the mean and the median straight away earlier. In statistics, there is something called skewness of the distribution.

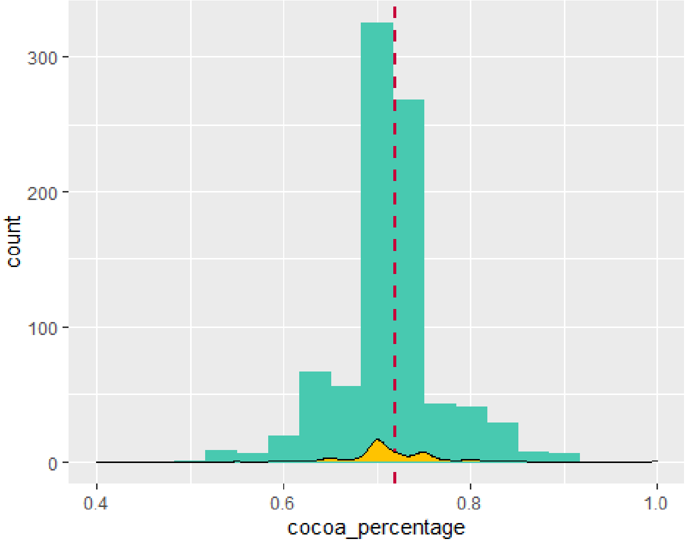
Skewness of the distribution is a measure of asymmetry of the value of a variable around its mean. It can be negative, positive or zero (we also call the latter normal distribution - and it looks bell-like). Also, a normal distribution has the mean and the median pretty much at the same value.

- When a skewness is negative (or left-tailed), then the most of the distribution's values are concentrated at the right side (side with higher numbers) of a histogram (graph of distribution). It is said to be left-tailed, because if most of the values are on the right side, the left side of the distribution is leaving a narrow tail. Median is higher than the mean, because the mean is a value that is influenced by extreme values (also called outliers), this time on the left side (a lot of smaller numbers lower the mean value).

- when a skewness is positive (or right-tailed), it is the opposite. Most of the distribution's values are concentrated at the left side of the distribution (at the lower ends of values). Again, the tail this time is on the right side. Median is lower than the mean, because the mean is influenced by extreme values on the higher end.

Let’s check it out on the histograms I mentioned just now.

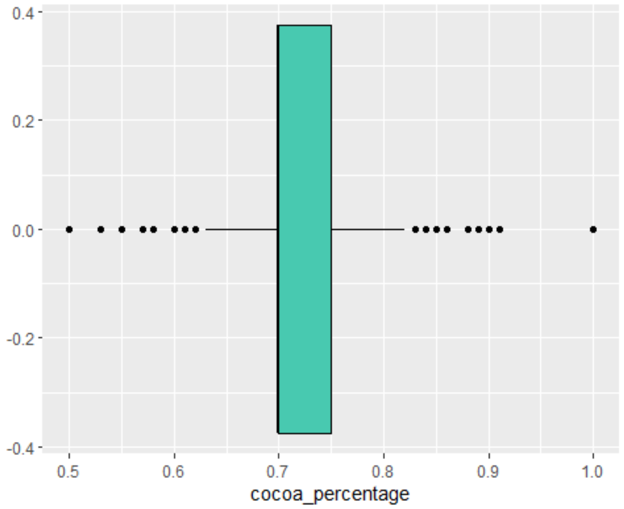
```
binsize = diff (range (dfrm$cocoa_percentage)) / 15
ggplot (dfrm, aes (x = cocoa_percentage)) + geom_histogram (binwidth = binsize, fill = "#48C9B0", colour = "#48C9B0")+
 geom_vline (aes (xintercept = mean (cocoa_percentage)), color = "#C70039", linetype = "dashed", size=1)+
 geom_density (fill = "#FFC300")+
 xlim (0.4,1)
```



Above, you can see what we data scientists call - histogram, or a graph representing a distribution of a viewed variable (in this case cocoa percentage). The red dotted line is representing mean value, whereas the median lies just where the tallest “tower” starts (so, a little bit left of the mean). Although the difference between the median and mean is small, we can see on the graph that difference isn’t small. Again, since the median is slightly lower than the mean, we can talk about a positive skewness, which isn’t visible here. That is because both tails look the same.

That is why we can also make a boxplot. It is essentially a graphic notion of the FIVENUM calculated above, but it can show outliers. Outliers are extreme values in the distribution, which can distort the mean value.

```
ggplot(dfrm, aes(x = cocoa_percentage)) + geom_boxplot(fill = "#48C9B0", colour = "black", outlier.colour = "black")
```



From the boxplot above, we can see outliers on both side of the boxplot (black dots). Since the minimum value is around 0.5, the skewness here is a positive one, because of this value 100% (1.0), which is much further from the mean than other values.

Also, you can see the third quartile around 0.75 (the end of the box), whereas the values of the first quartile, mean and the median is at the other side of the box, all crammed together, because their values are so close (check the FIVENUM again).

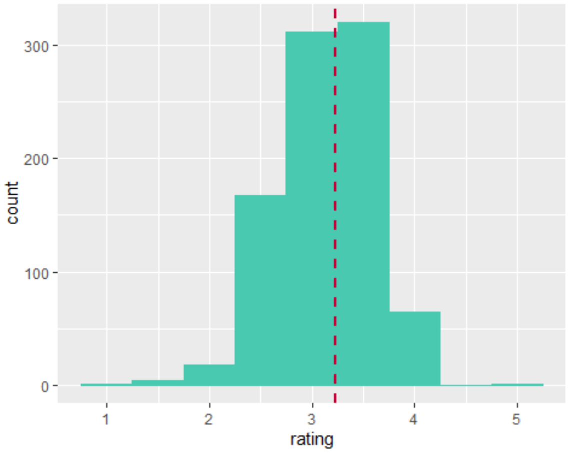
The next numeric value worth taking a look at is rating.

```
summary(rating)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   3.000   3.250   3.228   3.500   5.000

sd(rating)

## [1] 0.4662293
```

Rating goes from 1 to 5, and the representing descriptive values are above, at the beginning of this PDF. Again, the mean and the median values are close together, with the third and the first quartile being also together (but not enough to call this distribution symmetrical). When I say the latter, here is what I mean. A normal and symmetrical distribution has the most crammed values around the mean, so the middle of the distribution is the highest. The extreme values around every side of the mean should be minimal - look like tails. If a distribution is symmetrical around its mean, then the 25% and 75% of the distribution should also be symmetrical (same value). It has something to do this - when you standardize your values in the given distribution (standard deviation is then 1, and mean is 0; that procedure is used when you have various measuring system like kg vs lbs, then standardization helps to bring the values to the same measurement system - the one of standard deviation), then left of the mean are negative values (values smaller than mean), and on the right side of the mean you have positive values (values higher than the mean). Hence, the symmetrical values of the quartiles.

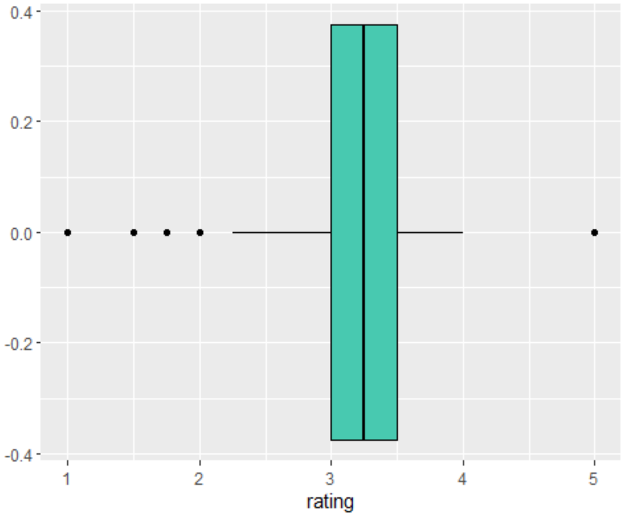
Let's check via histogram and boxplot what kind of distribution this is, and if there are any significant outliers.

```
binsize = diff (range(dfrm$rating))/8
ggplot (dfrm, aes (x = rating)) + geom_histogram (binwidth = binsize, fill = "#48C9B0", colour = "#48C9B0")+
  geom_vline (aes (xintercept = mean (rating)), color = "#C70039", linetype = "dashed", size=1)
```



As seen in the table and histogram above, this distribution has a negative skew or a left tail. Let's draw a boxplot to see if there are any significant outliers.

```
ggplot (dfrm, aes (x = rating)) + geom_boxplot (fill = "#48C9B0", colour = "black", outlier.colour = "black")
```



In the boxplot above, we can see 5 outliers, most of them being on the left side (hence the left tailed distribution). Only one chocolate bar has a rating of 5.0., and same goes for the rating of 1.0.

What should we do with the outliers? When only one or two significant outliers persist in a distribution, the common thing is to evaluate real life situation and see if that situation is also being present there too. We aren't per-se connoisseurs in the chocolate world, but of course there are some chocolate in the world which have top-of-the-line quality, which is also being seen here as well. If we delete those outliers, I am thinking that won't change the summary output much, because 50% of the values are already around grade 3.0., and the boxplot shows that the right whisker doesn't go too far because of the extreme value (5.0). So, for now, let's leave it as it is, and proceed with our analysis.

We still have variables like specific and broad bean origin, company location and the bean type to review. Those are qualitative variables, and we can't do FIVENUM on them. For now, I'll skip non-numeric variables, and proceed to correlation analysis between numeric variables.

Correlation analysis

Correlation is a measure that represents the relationship that two variables have, but not showing cause and effect of one another. For example, the correlation between height and weight is pretty positively high, meaning that the relationship between those two variables is positive (when a child grows, his weight also grows - both variables go in the same direction, but that doesn't mean that the height causes the weight to grow too, the weight itself is being connected to how much a person eats, exercises, etc.).

Real correlation can be calculated for numeric values/variables, and we have two of them here (rating and cocoa percentage). When I say "real", i mean you can back it up with number (percentage of correlation). You can still check correlation between non-numeric or qualitative variables, but then you can only look up the p-value and compare it to your posed p-value, and then get an answer whether two variables have a relationship or not. It is just not backed up by a specific number. Sadly, the qualitative variables need to be in a "factor" class, and our qualitative variables aren't.

Let's start with "real" correlation between numeric values, and immediately put up a scatter plot to see their relationship in graphic notion.

First, I am going to test if variables come from a normally distributed population, or the population itself first isn't normally distributed. For that, I am using Shapiro Wilk test. My posed p value is 0.05, which means that I am 95% certain that the population is normally distributed. P value is commonly used in testing hypotheses, and this value is attached to the null hypothesis. In this case, my hypothesis is that this variable comes from a normally distributed population. If the p value from the test comes back lower than 0.05, I can reject my null hypothesis and say that the variable doesn't come from a normally distributed population. P value greater than 0.05 provides no such evidence.

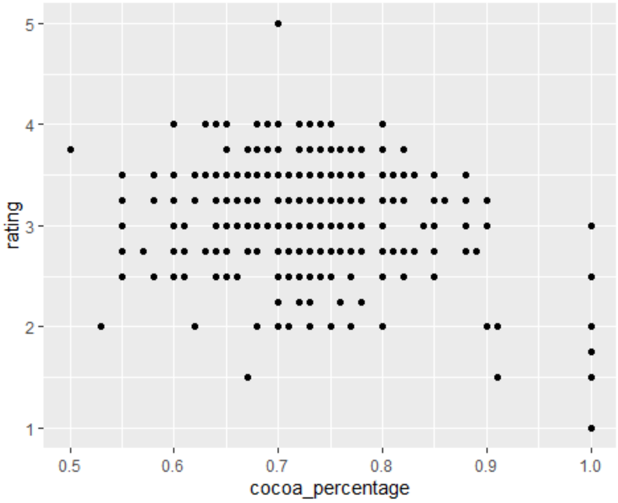
I'll just test the numeric variables, as it doesn't make sense to me personally to test these qualitative variables for normality.

```
shapiro.test (rating)

##
##  Shapiro-Wilk normality test
##
## data:  rating
## W = 0.95195, p-value < 2.2e-16

shapiro.test (cocoa_percentage)

##
##  Shapiro-Wilk normality test
##
## data:  cocoa_percentage
## W = 0.8876, p-value < 2.2e-16
```

Both normality tests conducted on the numeric variables showed p-value lower than 0.05, which means that the population is likely not normally distributed, which is to be expected, because almost nothing in the world is normally distributed in areas of production/supply/demand etc.

Next, we can check the real correlation between the numeric values, and immediately create a scatter plot which puts together both numeric values on the graph and shows their movement.

```
cor (rating,cocoa_percentage)

## [1] -0.1966369

cor.test (rating, cocoa_percentage, method = "pearson")

##
##  Pearson's product-moment correlation
##
## data:  rating and cocoa_percentage
## t = -5.9629, df = 884, p-value = 3.58e-09
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2591433 -0.1324902
## sample estimates:
##        cor
## -0.1966369

ggplot (dfrm, aes (x = cocoa_percentage, y = rating)) + geom_point()
```



Sadly, from the correlation test we conducted, we can see that the correlation is only -0.19, which is too low to be considered significant (it should be around minimum 0.4). Nevertheless, the p-value is lower than 0.05, so by that measure, the correlation can be seen as significant.

We can see on a scatter plot that the relationship isn't showing any way of movement, so hence the low correlation, and the scatter plot is backing that up.

Still, it would be neat to make a regression analysis as well, as we can show the relationship between numerical variables in a form of a linear equation.

Regression analysis

We have two numeric variables in this dataset - cocoa percentage and rating, so we can use regression analysis to see a form of a linear equation between them.

```
reg1 = lm (rating~cocoa_percentage)
print (summary(reg1))

##
## Call:
## lm(formula = rating ~ cocoa_percentage)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.80650 -0.25613  0.02385  0.30382  1.74387
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        4.3052     0.1814  23.739  < 2e-16 ***
## cocoa_percentage  -1.4987     0.2513  -5.963 3.58e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.4574 on 884 degrees of freedom
## Multiple R-squared:  0.03867,    Adjusted R-squared:  0.03758
## F-statistic: 35.56 on 1 and 884 DF,  p-value: 3.58e-09
```

R adjusted (coefficient of determination) means a fraction of the variance (variability/dispersion) of y-variable that is explained by the regression model (variables that are in the regression model itself), and the remaining variance/variability isn't explained by the variables in the model and that must be due to some other factors.

The R squared here is very low, only 0.03, which means that only 3% of the variance of the rating is explained by the cocoa percentage. That means that some other factors influence and are taken into account when the rating is given, and those variables aren't in this model. The p-value is lower than 0.05, which makes the model significant, and the FIVENUM of the residuals (difference between real data in the dataset and the data predicted by this model) is distributed well.

If we wish to use this regression model, the equation would be => rating = 4.3052 - 1,4987*cocoa_percentage

But, since the R squared and the correlation are very low, the model wouldn't give precise output.