Data Mining Assignment 2 Due July 8th

Matthew Dunne

July 1, 2018

Choosing Variables

Many of the variables in the German Credit data are binary (0, 1) signalling membership in a certain group, e.g. someone employed in a skilled profession or whether they have a telephone number. Within the data frame this represents every variable after NumberPeopleMaintenance. There is no mean for such variables, at least not in any meaningful sense and so these variables will be ignored.

Of the remaining variables, some are self-evidently appropriate to include in the k-means analysis (Duration, Amount, and Age). There is a question of whether to include others (ResidenceDuration, InstallmentRatePercentage, and NumberExistingCredits, NumberPeopleMaintenance) because they might be in effect signifiers of category, rather than a measure of a number. We might also choose not to include a variable because it is unlikely to influence of our clustering beyond making the result more difficult to interpret.

```
##  InstallmentRatePercentage ResidenceDuration NumberExistingCredits
##  Min.   :1.000             Min.   :1.000     Min.   :1.000
##  1st Qu.:2.000             1st Qu.:2.000     1st Qu.:1.000
##  Median :3.000             Median :3.000     Median :1.000
##  Mean   :2.973             Mean   :2.845     Mean   :1.407
##  3rd Qu.:4.000             3rd Qu.:4.000     3rd Qu.:2.000
##  Max.   :4.000             Max.   :4.000     Max.   :4.000
##  NumberPeopleMaintenance
##  Min.   :1.000
##  1st Qu.:1.000
##  Median :1.000
##  Mean   :1.155
##  3rd Qu.:1.000
##  Max.   :2.000
```

We shall not include any of these four "borderline" variables for the following reasons:

*Installment Rate Percentage (in percentage of disposable income) - There are only four values for this variable (1, 2, 3, 4). In effect, these are categorical rather than numerical (i.e. mathematical operations on the values would have little real meaning), and so we will not include it.

*Residence Duration - As for Installment Rate Percentage, the values are also essentially categorical. Also, it is unclear how this would readily have predictive effect on a person's grouping, and so we will not include it.

*Number of Existing Credits - This is a somewhat more difficult case because one would think that the number of existing credits should affect a person's grouping. But the data is still essentially categorical (1, 2, 3, 4). Also, only about 3% of the people are in the 3 or 4 category, and more than half are at 1. Its inclusion will not add much information, and so we will not include this variable.

*Number People Maintenance - This represents the number of people for which the borrower is liable to provide maintenance, but there are only two values (1 or 2) and so the variable is essentially binary. We will not include it.

As a result we will included only the following variables in our models:
* Duration
* Amount
* Age

Splitting the Data for Validation and Scaling for Interpretability

To scale the data we will normalize all variables (scale them from 0 to 1). For the k-means test we will have to validate it. So we will generate a training set and a test set. We put 70% of the data into the training set and 30% into the test set. To make coding easier we will first separate into training and testing data, then scale each.

```
library(caret)
set.seed(1234)
intrain<-createDataPartition(y=newdata$Duration,p=0.7,list=FALSE)
training<-newdata[intrain,]
testing<-newdata[-intrain,]
nrow(training)
```

```
## [1] 701
```

```
nrow(testing)
```

```
## [1] 299
```

```
scaledtrain<-as.data.frame(apply(X=training, MARGIN = 2, FUN = function(x){(x-min(x))/(max(x)-min(x))}))
scaledtest<-as.data.frame(apply(X=testing, MARGIN = 2, FUN = function(x){(x-min(x))/(max(x)-min(x))}))
head(cbind(training, scaledtrain))
```

```
##    Duration Amount Age  Duration     Amount        Age
## 2        48   5951  22 0.6470588 0.31270663 0.05357143
## 5        24   4870  53 0.2941176 0.25314084 0.60714286
## 6        36   9055  35 0.4705882 0.48374477 0.28571429
## 8        36   6948  35 0.4705882 0.36764382 0.28571429
## 11       12   1295  25 0.1176471 0.05614944 0.10714286
## 12       48   4308  24 0.6470588 0.22217324 0.08928571
```

Above is a small section of the unscaled training data alongside the scaled training data.

Choosing a K-Means Model

We will run the k-means algorithm on clusters ranging from size 2 to 10, with 100 random starts on the Training Data. Then we will extract the Variance Accounted For (between sum of squares / total sum of squares) for each.

The VAF

```
clusters<-2:10
between_SS<-c()
tot_SS<-c()
set.seed(1234)
for (num in clusters){
  cluster.object<-(kmeans(scaledtrain, centers = num, nstart = 100))
  between_SS<-append(between_SS, cluster.object$betweenss)
  tot_SS<-append(tot_SS, cluster.object$totss)
}

VAF<-as.data.frame(cbind(clusters, between_SS, tot_SS))
```

```
VAF['VAF']<-(between_SS/tot_SS)
VAF
```

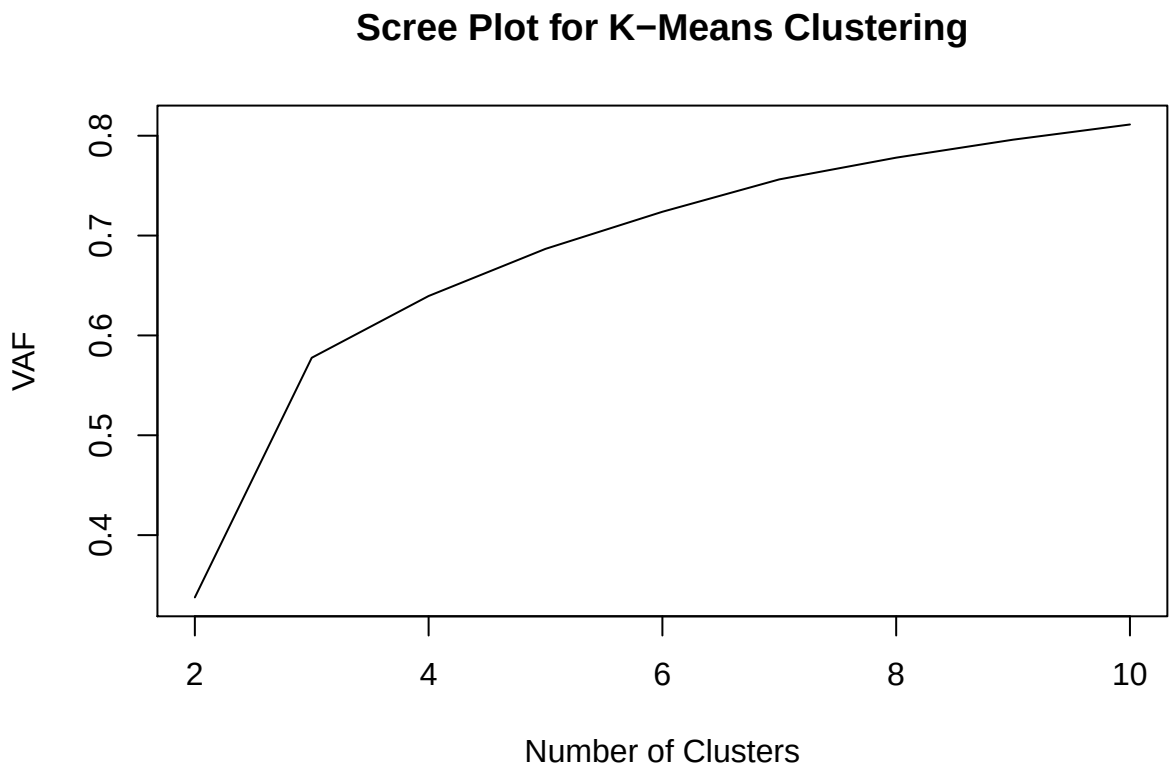
```
##   clusters between_SS   tot_SS       VAF
## 1        2   22.49097 66.61049 0.3376491
## 2        3   38.47452 66.61049 0.5776046
## 3        4   42.59545 66.61049 0.6394707
## 4        5   45.73989 66.61049 0.6866770
## 5        6   48.21518 66.61049 0.7238377
## 6        7   50.37375 66.61049 0.7562436
## 7        8   51.82338 66.61049 0.7780063
## 8        9   53.02339 66.61049 0.7960216
## 9       10   54.03641 66.61049 0.8112297
```

Here we see a large improvement in VAF from two to three clusters and diminishing improvements as we add more clusters.

The Scree Plot

A scree plot shows the fraction of total variance explained as we add additional clusters.

```
plot(2:10, VAF$VAF, type = "l", xlab = "Number of Clusters", ylab = "VAF", main = "Scree Plot for K-Mea
```



The most pronounced "elbow" is at 3 clusters, but at 3 clusters only about 60% of the variance is accounted for. Adding a fourth or fifth cluster explains more variance and that increase is significant. Adding a sixth cluster does not account for enough variance to make its inclusion worthwhile. So we will decide between 4 and 5 clusters.

To decide between 4 or 5 clusters we will look at four factors: VAF, size of clusters, interpretability, and performance on the test data.

VAF of Four and Five Cluster Models

As discussed above, the VAF of the four cluster model is 64% while the VAF of the five cluster model is 69%. This represents a 5% difference, but we must ask if this difference is worth the tradeoffs.

The Size of the Samples

Another factor in deciding between using four or five clusters is the proportion of observations within each cluster. Evenly distributed clusters are ideal, but at the very least we want a cluster to have more than a few observations. Otherwise it is not very useful.

The proportion of observations within each cluster are as follows -

For four clusters:

```
## [1] 0.2068474 0.2724679 0.1398003 0.3808845
```

For five clusters:

```
## [1] 0.24821683 0.22681883 0.09843081 0.28815977 0.13837375
```

The five cluster model could be considered more evenly distributed.

Interpretability of Centroids

The centers of the four cluster model are:

```
##   Duration   Amount      Age
## 1 16.62759 2837.959 51.37241
## 2 25.79058 3336.654 29.54450
## 3 42.77551 8332.541 34.77551
## 4 12.00749 1700.427 29.66667
```

The centers of the five cluster model are:

```
##   Duration   Amount      Age
## 1 26.59770 3353.937 29.20690
## 2 13.57862 2127.597 41.15094
## 3 19.01449 3283.449 58.11594
## 4 12.45545 1748.050 26.81683
## 5 42.47423 8431.660 34.73196
```

To this author, adding a fifth cluster muddies the water with regards to interpretability. With four clusters we can draw meaningful distinctions. Cluster 3 is long duration, high amount loans (we might suppose amount and duration are correlated). Then with short duration, low amount loans there is a distinction between older (Cluster 1) and younger borrowers (Cluster 4). Cluster 2 is younger people who take out loans of medium duration and amount.

The five cluster model is not so readily interpretable. Distinctions can be made, but Cluster 2 in particular looks superfluous when considered in conjunction with Cluster 4, i.e. it is splitting a group that has similar amount and duration based on age, but with no apparent significance.

Performance on Test Data

We will check the four cluster and five cluster model on our test data by plugging the centers derived from those models into our test data. We will compare VAF, size, and centers. Results consistent with training data should confirm we have made a good selection.

```
test.cluster.4<-kmeans(scaledtest, centers = cluster.object.4$centers)
test.cluster.5<-kmeans(scaledtest, centers = cluster.object.5$centers)
VAF.4.cluster<-test.cluster.4$betweenss/test.cluster.4$totss
VAF.5.cluster<-test.cluster.5$betweenss/test.cluster.5$totss
VAF.4.cluster
```

```
## [1] 0.6604965
```

```
VAF.5.cluster
```

```
## [1] 0.7040198
```

We see that the **VAF** for the four cluster model is 66% and for the five cluster model is 71%. These are very close if not slightly better than what we got on the training data, and the spread between the two is the same.

In looking at the **size of the clusters** we see that the clusters are not quite in the same proportion as on the training data, but the sizes are not inconsistent and **there are no de minimis clusters**:

```
## [1] 0.2307692 0.3143813 0.1036789 0.3511706
```

```
## [1] 0.2775920 0.2408027 0.1304348 0.2474916 0.1036789
```

And looking at the centers of the model, i.e. how interpretable they are, for the four cluster model:

```
##   Duration   Amount      Age
## 1 15.81159 2053.174 54.04348
## 2 26.74468 3529.840 32.53191
## 3 42.48387 9556.226 38.58065
## 4 11.86667 1734.600 29.92381
```

and for the five cluster model:

```
##   Duration   Amount      Age
## 1 27.39759 3728.084 32.54217
## 2 12.62500 1828.889 41.59722
## 3 17.48718 2146.282 59.53846
## 4 13.32432 1767.446 25.82432
## 5 42.48387 9556.226 38.58065
```

The centers for both the four and five cluster model as derived from the test data are relatively similar to that derived from the training data. I would say the **four cluster model is more interpretable on the test data as well**.

Choice of Number of Clusters

I choose the model with four clusters for the following reasons:

*While the five cluster model has a higher VAF, the drop off with the four cluster model is only 4-5% (across training and test data).

*While the observations are more evenly distributed in the five clusters model, the difference is not huge, nor is the four cluster model so concentrated that certain clusters have very few observations.

*The four cluster model is significantly more interpretable than the five cluster model.

*Both the four and five cluster models hold up well on the test data, where we see similar VAF and interpretability as in the training data.

In summation, I chose the four cluster model because the improvement in interpretability is well worth the loss of VAF and the increased concentration of observations among the clusters.

KO-means Comparison

First we load the ko-means function.

Then we run it on the scaled training data, using the number of clusters we chose from our k-means model (four clusters).

```
komeans.4 <- komeans(scaledtrain, nclust = 4, lnorm = 2, nloops = 100, tolerance = .001, seed = 3)
```

VAF

We can extract the VAF from this.

```
komeans.4$VAF
```

```
## [1] 0.7713429
```

Interpretability of Centroids

There are two ways I interpreted the centers of the clusters generated by the ko-means algorithm.

-Method 1

The first was by selecting the largest four (k) clusters of the 16 (2^k) clusters generated by the algorithm.

```
# find how many observations are in each cluster
k.vs.ko <- table(cluster.object.4$cluster,komeans.4$Group); apply(k.vs.ko,2,sum)
```

```
##   0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15
## 113 197  69  92  37  81  15  14   8  20  19  13   3  10   7   3
```

```
#would choose 0, 1, 3, and 5
```

Then we subset on the clusters we chose (0, 1, 3, 5), and after de-normalizing, find the means of each variable for observations within that cluster.

From that we get the following clusters and centroids.

```
#renormalize data
komeans.4$data[1]<-komeans.4$data[1]*(max(training$Duration)-min(training$Duration))+min(training$Durati
komeans.4$data[2]<-komeans.4$data[2]*(max(training$Amount)-min(training$Amount))+min(training$Amount)
komeans.4$data[3]<-komeans.4$data[3]*(max(training$Age)-min(training$Age))+min(training$Age)
#create a data frame of observations and ko groups
ko.groups<-cbind(komeans.4$data, komeans.4$Group)
#subset on relevant groups and aggregate
ko.top.groups<-ko.groups[which (ko.groups$`komeans.4$Group`==0 | ko.groups$`komeans.4$Group`==1 | ko.gr
aggregated<-aggregate(ko.top.groups[c(1:3)], by=list(ko.top.groups[,4]), FUN=mean)
rownames(aggregated)<-aggregated$Group.1; aggregated[1]<-NULL
aggregated
```

```
##   Duration   Amount      Age
## 0 20.56637 2756.708 33.59292
## 1 11.02030 1463.711 28.56853
## 3 23.03261 2673.109 26.50000
## 5 11.14815 1452.481 49.20988
```

These are not very interpretable. Clusters 0 and 3 should probably be combined given their similarity across all variables. They correspond roughly to Cluster 2 in our original k-means algorithm and represent about the same percentage of the observations. Given that we are dividing data into 16 clusters instead of 4 and then taking only 4 of those clusters (leaving out about 30% of the data) it should not be surprising that we have different centers.

-Method 2

The other way I tried was to extract the centroids output from the komeans function, de-standardize and then de-normalize from the training data.

```
##    Duration   Amount      Age
## 1  9.456047 1252.969 29.13390
## 2 37.818094 4993.281 30.88881
## 3 22.358952 3508.577 55.01892
## 4 29.175582 9475.945 36.04388
```

The latter method yielded more interpretable centroids, though it is not clear to me whether the underlying operation is valid.

Summary of Results and Interpretation

In summation, we chose a model based on three variables: duration, age, and amount. When choosing the number of clusters for our k-means model the we narrowed our selection to either four or five clusters because that is the point at which it made sense to ask whether the increase in VAF was worth the tradeoff, mostly in terms of interpretability. The tradeoff was not worth it, and so I decided to use the four cluster k-means model.

When we compared a four cluster k-overlapping means model to the four cluster k-means model we saw an improved VAF and a somewhat compromised level of interpretability.

Question 10: Recruiting People

Using the segments we have decided upon, we are given the task of recruiting 30 people per segment over the telephone for purposes of focus groups and other Attitudinal and Usage studies. Assuming we are tasked with recruiting new people (not those represented in the data we used in our algorithms) I would say the following:

- *What approach will you take to recruit people over the telephone?*

First, I would decide on what variables about the telephone recruiting process we can change (time of call, content of message.)

Second, I would examine if we might use some of the variables not included in our analysis to ascertain characteristics about our segments/clusters. They were categorical and therefore not appropriate to a k-means clustering algorithm, but they still represent information on the characteristics of the people that may influence how we do the calls for that segment, e.g. if a cluster is mostly older, employed people it could influence the time of day we call.

Third, I would call (or robo-call) the people and if ignored leave a voicemail message. I do not answer calls from numbers I do not recognize and would guess neither do others.

-*Assume consumers who are recruited will be reimbursed, which of the consumers will you try to recruit?*

I would not target consumers based on likelihood to respond to reimbursement. This could introduce a selection bias into the data because it selects for a certain "financial personality" type, i.e. those who will participate in a survey for a (presumably small) financial reward. Given that we are trying to group people in order to predict financial behavior, we should not bias the sample by selecting for a trait that would influence the composition of the sample.

-*How will you identify if a new recruit belongs to a particular segment?*

Given that they are new recruits, we do not have information about duration and amount of loans, because they have never had one. We would presumably know age however. So I would ask about a hypothetical loan, e.g. if you were to get a loan would it be for (choose the closest): $1500, $3000, or $8000. If the person is older they go in Cluster 1, unless they choose a large hypothetical loan. If a person is younger, they will be assigned a cluster based on their hypothetical loan amount.