Drug Consumption From Personality

John Curcio

April 22, 2017

Dataset Summary

This dataset can be downloaded from the UCI Machine Learning Repository. It contains records for 1885 respondents. For each respondent, personality attributes are collected, as well as demographic stuff like age, education, etc. In addition, participants were questioned concerning their use of 18 legal and illegal drugs - everything from caffeine and chocolate to crack and ketamine. Also, one fictitious drug (Semeron) which was introduced to identify over-claimers (lol).

Attribute Information:

1. ID is number of record in original database. Cannot be related to participant. It can be used for reference only.
2. Age (Real) is age of participant and has one of the values: Value Meaning Cases Fraction -0.95197 18-24 643 34.11% -0.07854 25-34 481 25.52% 0.49788 35-44 356 18.89% 1.09449 45-54 294 15.60% 1.82213 55-64 93 4.93% 2.59171 65+ 18 0.95% Descriptive statistics Min Max Mean Std.dev. -0.95197 2.59171 0.03461 0.87813
3. Gender (Real) is gender of participant: Value Meaning Cases Fraction 0.48246 Female 942 49.97% -0.48246 Male 943 50.03%
4. Education (Real) is level of education of participant and has one of the values: Value Meaning Cases Fraction -2.43591 Left school before 16 years 28 1.49% -1.73790 Left school at 16 years 99 5.25% -1.43719 Left school at 17 years 30 1.59% -1.22751 Left school at 18 years 100 5.31% -0.61113 Some college or university, no certificate or degree 506 26.84% -0.05921 Professional certificate/ diploma 270 14.32% 0.45468 University degree 480 25.46% 1.16365 Masters degree 283 15.01% 1.98437 Doctorate degree 89 4.72%
5. Country (Real) is country of current residence of participant and has one of the values: Value Meaning Cases Fraction -0.09765 Australia 54 2.86% 0.24923 Canada 87 4.62% -0.46841 New Zealand 5 0.27% -0.28519 Other 118 6.26% 0.21128 Republic of Ireland 20 1.06% 0.96082 UK 1044 55.38% -0.57009 USA 557 29.55%
6. Ethnicity (Real) is ethnicity of participant and has one of the values: Value Meaning Cases Fraction -0.50212 Asian 26 1.38% -1.10702 Black 33 1.75% 1.90725 Mixed-Black/Asian 3 0.16% 0.12600 Mixed-White/Asian 20 1.06% -0.22166 Mixed-White/Black 20 1.06% 0.11440 Other 63 3.34% -0.31685 White 1720 91.25%
7. Nscore (Real) is NEO-FFI-R Neuroticism.
8. Escore (Real) is NEO-FFI-R Extraversion.
9. Oscore (Real) is NEO-FFI-R Openness to experience.
10. Ascore (Real) is NEO-FFI-R Agreeableness.
11. Cscore (Real) is NEO-FFI-R Conscientiousness.
12. Impulsive (Real) is impulsiveness measured by BIS-11.
13. SS (Real) is sensation seeing measured by ImpSS.

14 - 30: Alcohol, Amphet, Benzos, Caff, Cannabis, Chocolate, Cocaine, Crack, Ecstasy, Heroin, Ketamine, "Legal Highs", LSD, Meth, Mushrooms, Nicotine,

31. Semer is class of fictitious drug Semeron consumption.
32. VSA is class of volatile substance abuse consumption.

Some Quick Cleaning

Inclusion of the fictional drug "Semeron", encoded in the dataset as `Semer`, was pretty clever. I'll remove all respondents who claim to have used Semeron, because they've proven themselves to be totally unreliable.

Also, I don't need `ID`. Doesn't tell us anything about the individual patient, and I can already be assured of uniqueness.

```
df <- df[which(df$Semer == "CL0"),]
df$ID <- NULL
```

Logistic Regression

For now, I'll just use logistic regression to predict use of nicotine at least as recently as last month (`CL4` to `CL6`, inclusive).

I've decided to ignore `Country`, `Ethnicity`, and use of other drugs. While they're *extremely* important and insightful variables, I'm just exploring right now.

```
train.rows <- sample(c(1:nrow(df)), 1000)
train.df <- df[train.rows,]
train.df$Nicotine <- (df$Nicotine[train.rows] == "CL4" |
                      df$Nicotine[train.rows] == "CL5" |
                      df$Nicotine[train.rows] == "CL6")
mod <- glm(Nicotine ~ as.factor(Age) + as.factor(Gender) +
             as.factor(Education) +
             Nscore + Escore + Oscore + Ascore + Cscore +
             Impulsive + SS,
           data = train.df, family=binomial)
```

Now that we've trained our model, `mod`, let's check its accuracy.

```
pred.labels <- (mod$fitted.values >= 0.5) #confusingly, the fitted values
                                          #are probabilities...
trainacc <- sum(train.df$Nicotine == pred.labels) / nrow(train.df)

pred.logodds <- predict.glm(mod, newdata=df[-train.rows,])#...while predict
                                                          #returns logodds
pred.labels <- (pred.logodds >= 0)
true.labels <- (df$Nicotine[-train.rows] == "CL4" |
                df$Nicotine[-train.rows] == "CL5" |
                df$Nicotine[-train.rows] == "CL6")
testacc <- sum(pred.labels == true.labels) / nrow(df[-train.rows,])
trainacc
```

```
## [1] 0.663
```

```
testacc
```

```
## [1] 0.6841505
```

Not bad. Even *more* encouraging is the fact that the test and train accuracy scores are quite similar. Our model doesn't seem to be overfitting to the training data at all - it's generalizing well! Repeating this partitioning of train and test data (called "cross-validation") would give us a better idea of whether this is just a fluke or not.

Simple Feature Engineering

I'm sure we can get better accuracy through some feature engineering. The origninal dataset comes with some very bizarre encodings of values (for example, if I recall correctly, `Age` is encoded as 0.48246 for men and -0.4826 for women). They might have been useful choices for the researchers who published this data, but my approach might benefit by encoding them differently.

I'll do this by re-encoding the use of each drug to real values. I'm not sure exactly how the possible values should scale, but I'm certain that drug use is very much an *ordinal* variable.

```
useAsNumeric <- function(use){
  result <- c(1:length(use))*0
  labels <- c("CL0", "CL1", "CL2", "CL3", "CL4", "CL5", "CL6")
  for(i in 1:length(use)){
    for(j in 1:length(labels)){
      if(use[i] == labels[j]){
        result[i] = j
      }
    }
  }
  return(result)
}

drugnames <- c("Alcohol",
  "Amphet",
  "Amyl",
  "Benzos",
  "Caff",
  "Cannabis",
  "Choc",
  "Coke",
  "Crack",
  "Ecstasy",
  "Heroin",
  "Ketamine",
  "Legalh",
  "LSD",
  "Meth",
  "Mushrooms",
#  "Nicotine",  #skipping this for now...Re-encoding Nicotine in this way
                #would change this to a *regression* problem.
  "Semer",
  "VSA")

df.recoded <- df
```

```
df.recoded$Nicotine <- (df$Nicotine == "CL4" |
                        df$Nicotine == "CL5" |
                        df$Nicotine == "CL6")
for(i in c(1:length(drugnames))){
  df.recoded[,drugnames[i]] <- useAsNumeric(df[,drugnames[i]])
}
```

Logistic Regression Again

Now that the data has been re-encoded, we're ready to fit our second model, `mod2`. It's just logistic regression again, except we've introduced new predictor variables.

I'm still ignoring `Country` and `Ethnicity`. Reason is they're not ordinal variables, they're purely categorical. Including them would increase the degrees of freedom (complexity) in our model, but each's distribution of values is very skewed (e.g. well over 90% white). We wouldn't be able to be very confident in their respective weights.

```
train.rows <- sample(c(1:nrow(df.recoded)), 1000)
train.df <- df.recoded[train.rows,]
mod2 <- glm(Nicotine ~ . - Country - Ethnicity, #"." is a shortcut for "all other predictors"
            data = train.df, family=binomial)

#get train accuracy
pred.labels <- (mod2$fitted.values >= 0.5)
trainacc <- sum(train.df$Nicotine == pred.labels) / nrow(train.df)
#get test accuracy
pred.logodds <- predict.glm(mod, newdata=df.recoded[-train.rows,])
pred.labels <- (pred.logodds >= 0)
true.labels <- df.recoded$Nicotine[-train.rows]
testacc <- sum(pred.labels == true.labels) / nrow(df.recoded[-train.rows,])
trainacc
```

```
## [1] 0.726
```

```
testacc
```

```
## [1] 0.6784493
```

Okay, test accuracy hasn't improved significantly, and I'm seeing a much wider split between train and test accuracy. Including other drugs as predictors in our model has only led to overfitting.

Obviously, looking at respondents' use of other drugs *is* going to be useful in predicting nicotine use. The issue here lies with logistic regression itself.

The logistic model assumes that there's no collinearity between predictors, when really there probably is some strong relationship between predictors. For example, older people are more likely to have higher education, and people who do mushrooms are more likely to do LSD than they are to do meth (I assume).

Random Forest

A better method might be random forest. Random Forest is an ensemble method where a forest consists of a large number of decision trees, which are generated through a random process. The trees in a random forest tend to be much "deeper" than regular decision trees, but their large number and diversity corrects for the overfitting that would otherwise be expected.

There's a nice implementation in the `randomForest` package in R. We'll use it to train our third model, `mod3`.

```
library(randomForest)
mod3 <- randomForest(as.factor(Nicotine) ~ . - Country - Ethnicity, data = train.df,
                     # importance=T,
                     # mtry=2,
                     ntree=1000)
```

Okay, believe me: there's a *lot* more complexity to random forests than a single function call in R. But this is useful for some quick and dirty analysis.

```
#get train accuracy
pred.labels <- predict(mod3, newdata = df.recoded[train.rows,])
trainacc <- sum(train.df$Nicotine == pred.labels) / nrow(train.df)
#get test accuracy
pred.labels <- predict(mod3, newdata = df.recoded[-train.rows,])
true.labels <- df.recoded$Nicotine[-train.rows]
testacc <- sum(true.labels == pred.labels) / nrow(train.df)

trainacc
```

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

```
testacc
```

```
## [1] 0.631
```

Wow, wasn't expecting that to do so poorly! Pretty underwhelming.

Honestly, I *still* think random forest will do better than plain old logistic regression. This just suggests that some feature engineering is needed.

To Be Continued!