Predicting Drug Misuse with Logistic Model on Binary Indicator Variables based on BIC Score

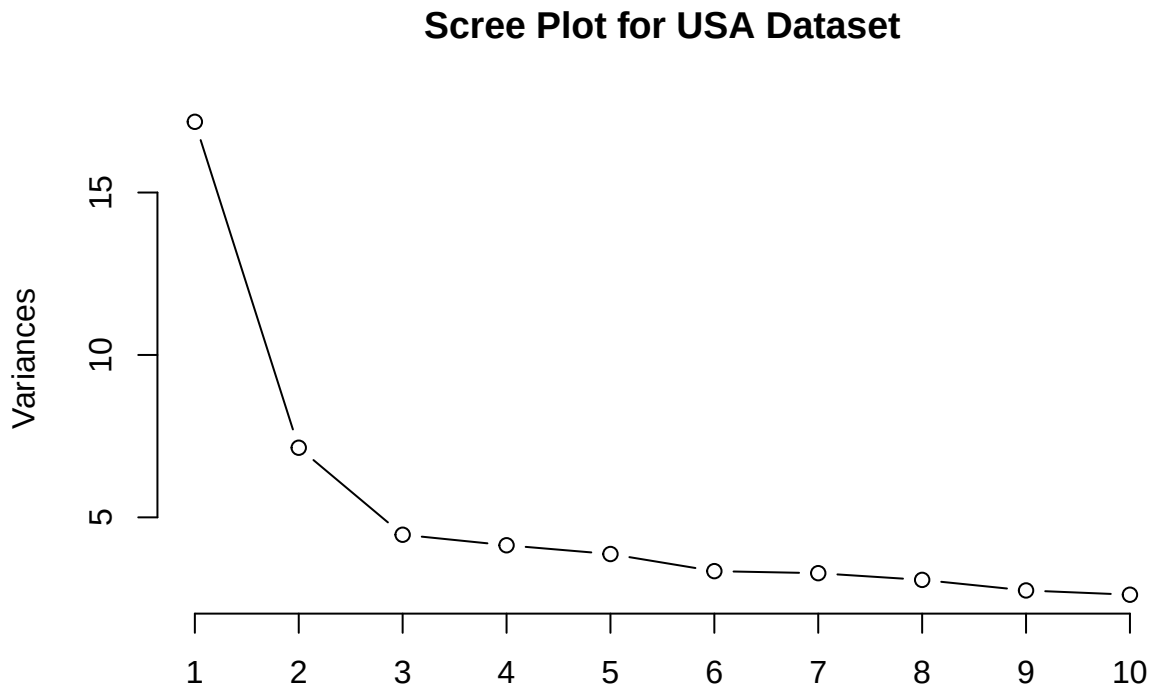
From Rocky Mountain Poison & Drug Safety

Clayton Covington, Sameer Rao, Kyle Sorensen

4/9/2021

Part 1: Attempt at Reducing Dimensionality

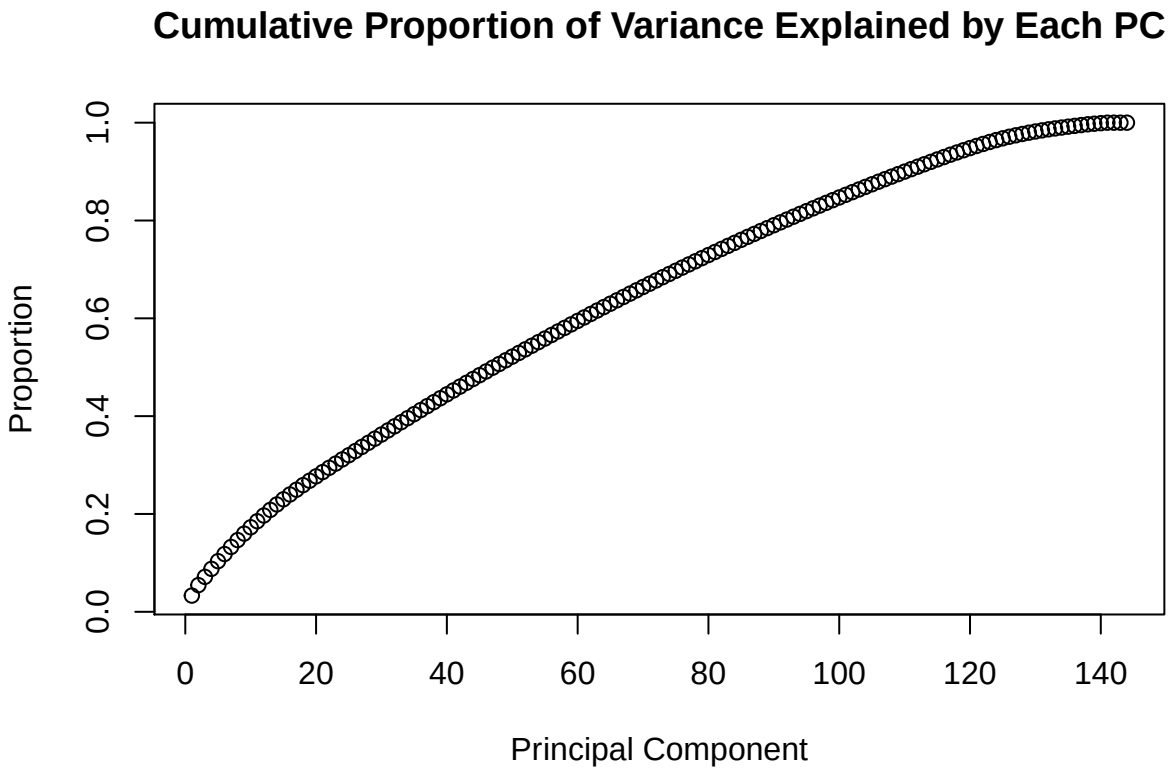
In an effort to reduce the dimensionality of our dataset since it is quite large, we attempted to perform a principal component analysis of the 2018 USA dataset. To do this, we first filtered the dataset for `NA` values and kept numeric data only. This step was primarily exploratory. The output below is a scree plot showing the proportion of variance that can be accounted for with a given number of principal components. This plot does not show any desired "bend," so we move on to other methods.



Next, we observed change in the cumulative proportion of variance explained by each principal component in order to support our findings in the scree plot. This further suggests that a PCA is not appropriate for this dataset.

```
## [1] 0.1728027
```

```
## [1] 125.0141
```



Part 2: Identifying Binary Variables

After some exploratory data analysis with PCA, we started with a function called `all_probs()` which takes in one of the data sets provided by Rocky Mountain Poison & Drug Safety, as well as a minimum `drug_misuse_rating`. The output of this function is a list of all binary indicator variables, which we have decided to use as our primary predictors. In addition, the output gives details on which binary variables lead to increased and decreased risks.

Below, we see a use case of the `all_probs()` function, where the dataset analyzed is the 2018 USA data set and the drug misuse threshold discussed above is given by `DAST_SUM > 1`.

The command `results$increased_risk` reveals all binary variables such that a response coded to 1 in the dataset leads to a proportional increase in risk of drug misuse.

```
## Note: Using an external vector in selections is ambiguous.
## i Use ‘all_of(clear_vector)‘ instead of ‘clear_vector‘ to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
##  [1] "DEM_STDNT"       "DEM_VET"         "DEM_HEALTH"      "FENT_USE"
```

```
##  [5] "BUP_USE"          "METH_USE"         "MORPH_USE"        "OXY_USE"
##  [9] "OXYM_USE"         "TRAM_USE"         "TAP_USE"          "HYD_USE"
## [13] "HYDM_USE"         "SUF_USE"          "COD_USE"          "DIHY_USE"
## [17] "BENZ_USE"         "STIM_USE"         "THC_USE"          "KTM_USE"
## [21] "OTH_RX_DRUG_USE" "HELP_SUB_USE"    "PAIN_ACUTE"       "RXDRUGSAFE_HIGH"
## [25] "RXDRUGSAFE_THER" "DRSHOP_USE"      "DRSHOP_SELL"      "MENT_ANX"
## [29] "MENT_ADHD"        "MENT_AUT"         "MENT_BIP"         "MENT_BPD"
## [33] "MENT_DEP"         "MENT_EAT"         "MENT_OCD"         "MENT_PANIC"
## [37] "MENT_PPD"         "MENT_PTSD"        "MENT_SCH"         "MENT_OTH"
## [41] "OP_USE"           "GABA_USE"         "OP_NMU_EVER"      "GABA_NMU_EVER"
## [45] "BENZ_NMU_EVER"    "STIM_NMU_EVER"    "OP_NMU_YR"        "OP_NMU_NTY"
## [49] "OP_NMU_MNTH"      "OP_NMU_WK"        "BENZ_NMU_YR"      "BENZ_NMU_NTY"
## [53] "BENZ_NMU_MNTH"    "BENZ_NMU_WK"      "STIM_NMU_YR"      "STIM_NMU_NTY"
## [57] "STIM_NMU_MNTH"    "STIM_NMU_WK"      "GABA_NMU_YR"      "GABA_NMU_NTY"
## [61] "GABA_NMU_MNTH"    "GABA_NMU_WK"      "ILL_USE"          "ILL_YR"
## [65] "ILL_MNTH"         "ILL_WK"           "BUP_NMU_NTY"      "COD_NMU_NTY"
## [69] "DIHY_NMU_NTY"     "FENT_NMU_NTY"     "HYD_NMU_NTY"      "HYDM_NMU_NTY"
## [73] "METH_NMU_NTY"     "MORPH_NMU_NTY"    "OXY_NMU_NTY"      "OXYM_NMU_NTY"
## [77] "SUF_NMU_NTY"      "TAP_NMU_NTY"      "TRAM_NMU_NTY"
```

The `all_probs()` function identified increased risks for 79 of the variables in the 2018 USA dataset. These increased risks are largely for the questions asking about drug use, but a few are not. In particular, if a respondent was a student, a veteran, or worked in the health profession, they have a higher risk of drug misuse.

The command `results$decreased_risk` reveals all binary variables such that a response coded to 1 in the dataset leads to a proportional increase in risk of drug misuse.

```
## [1] "DEM_GENDER"      "RXDRUGSAFE_PAIN" "MENT_NONE"
```

The `all_probs()` function identified decreased risks for 3 of the variables in the 2018 USA data. This includes gender, suggesting that, overall, women have a lower risk for drug misuse than men.

The command `results$classify` reveals all binary variables such that a response coded to 1 in the dataset leads to an increase of 50 percent or greater in achieving our threshold for risk of drug misuse. In the case with a threshold of 1, there are 27 of this class of variable.

```
##  [1] "HELP_SUB_USE"  "DRSHOP_USE"    "DRSHOP_SELL"   "MENT_BPD"
##  [5] "BENZ_NMU_EVER" "STIM_NMU_EVER" "OP_NMU_WK"     "BENZ_NMU_YR"
##  [9] "BENZ_NMU_NTY"  "BENZ_NMU_MNTH" "BENZ_NMU_WK"   "STIM_NMU_YR"
## [13] "STIM_NMU_NTY"  "STIM_NMU_MNTH" "STIM_NMU_WK"   "BUP_NMU_NTY"
## [17] "DIHY_NMU_NTY"  "FENT_NMU_NTY"  "HYD_NMU_NTY"   "HYDM_NMU_NTY"
## [21] "METH_NMU_NTY"  "MORPH_NMU_NTY" "OXY_NMU_NTY"   "OXYM_NMU_NTY"
## [25] "SUF_NMU_NTY"   "TAP_NMU_NTY"   "TRAM_NMU_NTY"
```

The command `results$Risk_Data` outputs a dataframe that contains the increase in proportion discussed above, as well as the total when that variable is under consideration.

```
##     Variable     Total     Increase
## 2 DEM_GENDER 0.1111555 -0.052166907
## 3  DEM_STDNT 0.2251366  0.096752781
## 4    DEM_VET 0.1437070  0.007243295
## 5 DEM_HEALTH 0.1659808  0.030214219
## 6   FENT_USE 0.3555186  0.232238194
## 7    BUP_USE 0.4619651  0.338987226
```

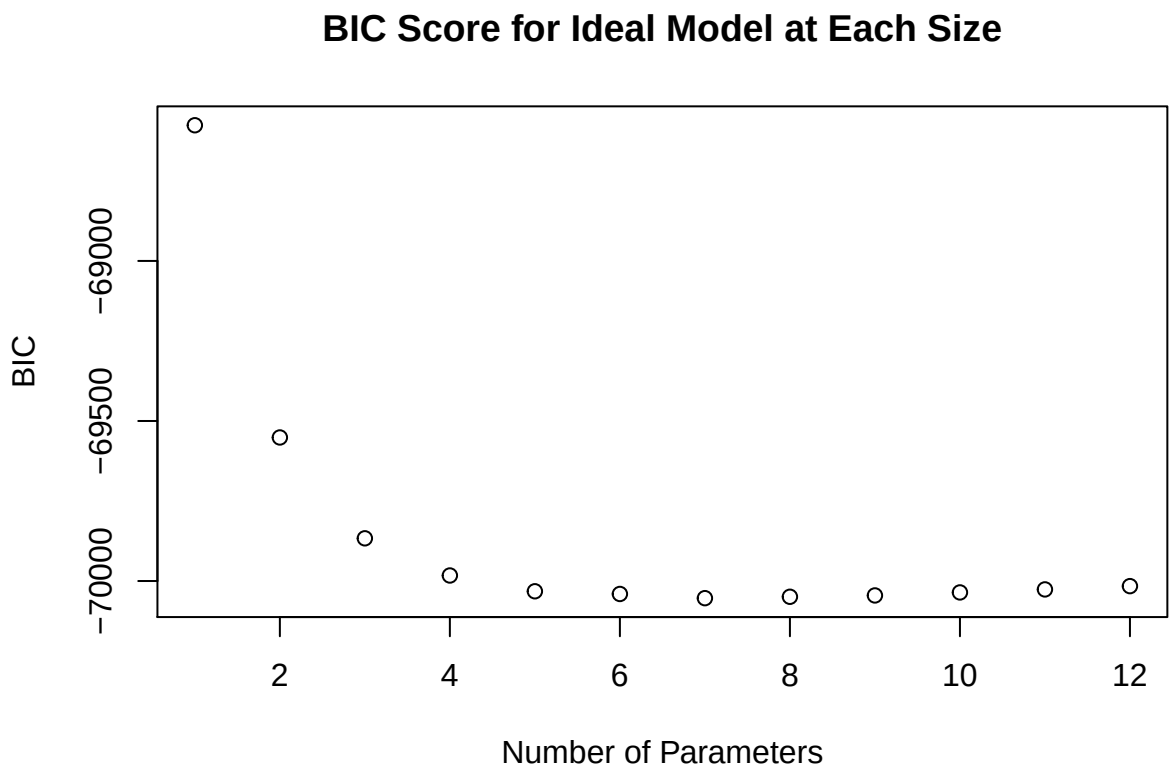
Part 3: Logistic Regression on the Binary Variables

After acquiring data on the binary variables, we made an attempt to predict drug misuse based on some subset of these variables. For this we used a logistic regression because we are trying to predict a binary variable. We could have used linear discriminant analysis, but LDA requires more assumptions about the underlying predictor variables, assumptions that we suspected were not met.

To achieve this, we use the `regsubsets()` function in the `leaps` package, which tells use the "best" model for each dataset based on some performance metric such AIC or BIC. In our situation, we use BIC, since we would like to "reward" a simpler model. We start by looking at the binary variables in order of greatest influence on risk of drug misuse. In this case, our ideal number of parameters is given by the one with minimum BIC at drug misuse threshold greater than 1, indicating any drug misuse. Based on this graph, we determined that the optimal number of parameters is 7.

```
## Subset selection object
## Call: regsubsets.formula(drug_misuse ~ ., data = regsub_comp, y = regsub_comp$drug_misuse,
##     nvmax = 12)
## 12 Variables  (and intercept)
##               Forced in Forced out
## DRSHOP_SELL       FALSE      FALSE
## STIM_NMU_WK       FALSE      FALSE
## DRSHOP_USE        FALSE      FALSE
## HELP_SUB_USE      FALSE      FALSE
## STIM_NMU_MNTH     FALSE      FALSE
## METH_NMU_NTY      FALSE      FALSE
## STIM_NMU_NTY      FALSE      FALSE
## BENZ_NMU_WK       FALSE      FALSE
## MORPH_NMU_NTY     FALSE      FALSE
## DIHY_NMU_NTY      FALSE      FALSE
## TAP_NMU_NTY       FALSE      FALSE
## STIM_NMU_YR       FALSE      FALSE
## 1 subsets of each size up to 12
## Selection Algorithm: exhaustive
##           DRSHOP_SELL STIM_NMU_WK DRSHOP_USE HELP_SUB_USE STIM_NMU_MNTH
## 1  ( 1 )  " "         " "         " "         "*"          " "
## 2  ( 1 )  " "         " "         "*"         "*"          " "
## 3  ( 1 )  " "         " "         "*"         "*"          " "
## 4  ( 1 )  "*"         " "         "*"         "*"          " "
## 5  ( 1 )  "*"         " "         "*"         "*"          " "
## 6  ( 1 )  "*"         " "         "*"         "*"          " "
## 7  ( 1 )  "*"         "*"         "*"         "*"          " "
## 8  ( 1 )  "*"         "*"         "*"         "*"          " "
## 9  ( 1 )  "*"         "*"         "*"         "*"          " "
## 10  ( 1 ) "*"         "*"         "*"         "*"          " "
## 11  ( 1 ) "*"         "*"         "*"         "*"          " "
## 12  ( 1 ) "*"         "*"         "*"         "*"          "*"
##           METH_NMU_NTY STIM_NMU_NTY BENZ_NMU_WK MORPH_NMU_NTY DIHY_NMU_NTY
## 1  ( 1 )  " "           " "          " "         " "           " "
## 2  ( 1 )  " "           " "          " "         " "           " "
## 3  ( 1 )  " "           " "          " "         " "           " "
## 4  ( 1 )  " "           " "          " "         " "           " "
## 5  ( 1 )  " "           " "          "*"         " "           " "
## 6  ( 1 )  " "           " "          "*"         "*"           " "
## 7  ( 1 )  " "           " "          "*"         "*"           " "
```

```
## 8  ( 1 )  " "          " "          "*"         "*"          " "
## 9  ( 1 )  "*"          " "          "*"         "*"          " "
## 10  ( 1 ) "*"          "*"          "*"         "*"          " "
## 11  ( 1 ) "*"          "*"          "*"         "*"          "*"
## 12  ( 1 ) "*"          "*"          "*"         "*"          "*"
##           TAP_NMU_NTY STIM_NMU_YR
## 1  ( 1 )  " "         " "
## 2  ( 1 )  " "         " "
## 3  ( 1 )  " "         "*"
## 4  ( 1 )  " "         "*"
## 5  ( 1 )  " "         "*"
## 6  ( 1 )  " "         "*"
## 7  ( 1 )  " "         "*"
## 8  ( 1 )  "*"         "*"
## 9  ( 1 )  "*"         "*"
## 10  ( 1 ) "*"         "*"
## 11  ( 1 ) "*"         "*"
## 12  ( 1 ) "*"         "*"
```



From the `regsubsets()` function, we note that the best 7 binary parameters to include are given by "DRSHOP_SELL", "DRSHOP_USE", "HELP_SUB_USE", "BENZ_NMU_WK", "STIM_NMU_WK", "STIM_NMU_YR" and "MORPH_NMU_NTY" with a BIC score of -70053.50. Here, we create logistic model based on these binary variables. We also display a summary of the logistic model and note that all parameters are highly significant and that the AIC score (15049) is quite low along with the BIC score given above, indicating a properly fitted logistic model. (WARNING: this logistic regression is on the entire data set for identifying the predictors properly.)

```
## Note: Using an external vector in selections is ambiguous.
## i Use ‘all_of(ideal_vector)‘ instead of ‘ideal_vector‘ to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.


##
## Call:
## glm(formula = drug_misuse ~ ., data = ideal_data)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.47879  -0.09809  -0.09809  -0.09809   0.90191
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.098092   0.001857  52.819  < 2e-16 ***
## DRSHOP_SELL    0.183572   0.017052  10.765  < 2e-16 ***
## DRSHOP_USE     0.259735   0.014287  18.180  < 2e-16 ***
## HELP_SUB_USE   0.470329   0.008527  55.159  < 2e-16 ***
## BENZ_NMU_WK    0.163804   0.022090   7.415 1.25e-13 ***
## STIM_NMU_WK   -0.181175   0.037469  -4.835 1.34e-06 ***
## STIM_NMU_YR    0.371089   0.025833  14.365  < 2e-16 ***
## MORPH_NMU_NTY  0.113344   0.023411   4.841 1.30e-06 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for gaussian family taken to be 0.09664504)
##
##     Null deviance: 3552.9  on 30006  degrees of freedom
## Residual deviance: 2899.3  on 29999  degrees of freedom
## AIC: 15049
##
## Number of Fisher Scoring iterations: 2
```

Part 4: Diagnostics on Logistic Regression

Finally, we take a subset of 22505 observations (75 percent) from the original 2018 USA data and train our logistic model and test it against the rest of the data (7502 observations, 25 percent). Below, the output shows an estimate of the accuracy of the logistic model when tested against the testing data set. Based on this estimate, our logistic model correctly classifies an estimated 88.75% of all survey takers.

```
##
## Call:
## glm(formula = drug_misuse ~ ., data = training_data)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.27590  -0.09932  -0.09932  -0.09932   0.90068
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.099320   0.002154  46.100  < 2e-16 ***
```

```
## DRSHOP_SELL    0.178923   0.019562   9.146  < 2e-16 ***
## DRSHOP_USE     0.249465   0.016404  15.208  < 2e-16 ***
## HELP_SUB_USE   0.472855   0.009913  47.700  < 2e-16 ***
## BENZ_NMU_WK    0.170990   0.024978   6.846 7.81e-12 ***
## STIM_NMU_WK   -0.189131   0.041698  -4.536 5.77e-06 ***
## STIM_NMU_YR    0.388273   0.029110  13.338  < 2e-16 ***
## MORPH_NMU_NTY  0.104348   0.026346   3.961 7.50e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.09750377)
##
##     Null deviance: 2694.0  on 22504  degrees of freedom
## Residual deviance: 2193.5  on 22497  degrees of freedom
## AIC: 11488
##
## Number of Fisher Scoring iterations: 2

## [1] 0.8823817

## [1] 0.8874967
```

We can better understand our logistic model by looking at its confusion matrix. Based on this matrix, we are correctly identifying 97.78% of non-misusers. We are incorrectly classifying 2.21% of these non-misusers. Further, we are correctly classifying 29.22% of misusers, and incorrectly classifying 70.78%. Thus, if someone is classified as a '1' by our model, there is a 66.74% chance that they are actually a misuser.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 6369  700
##          1  144  289
##
##                Accuracy : 0.8875
##                  95% CI : (0.8801, 0.8946)
##     No Information Rate : 0.8682
##     P-Value [Acc > NIR] : 2.376e-07
##
##                   Kappa : 0.3547
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9779
##             Specificity : 0.2922
##          Pos Pred Value : 0.9010
##          Neg Pred Value : 0.6674
##              Prevalence : 0.8682
##          Detection Rate : 0.8490
##    Detection Prevalence : 0.9423
##       Balanced Accuracy : 0.6351
##
##        'Positive' Class : 0
##
```

Part 5: Conclusions

From our logistic model, we can see that the best binary predictors of drug misuse according to the provided data are (1) whether someone has attempted to get a prescription for a medication that they did not need in order to sell it and (2) in order to misuse the drug, (3) whether someone has sought professional help for substance abuse, (4) whether someone has gotten a prescription for benzodiazepine product in the last 7 days for non-medical use or (5) a prescription stimulant in the last 7 days for non-medical use or (6) a prescription stimulant in the last year days for non-medical use or (7) a prescription morphine in the last 90 days for non-medical use. Of these predictors, the ones with strongest impact on risk of drug misuse are (3), (2), and (1). Surprisingly, the indication for a stimulus prescrition for non-medical use in the last 7 days is actually negative! Perhaps this variable helps distingush between frequent users verson occasional users, or some other explanation?

According to the above conclusions, the best way to predict drug misuse through a questionnaire would be with the following questions:

1. Have you attempted to get a prescription for a medication that you did not need in order to sell it?
2. Have you attempted to get a prescription for a medication that you did not need in order to misuse it?
3. Have you ever sought professional help for substance abuse?
4. Have you gotten a prescription for a benzodiazepine in the last 7 days for non-medical use?
5. Have you gotten a prescription stimulant in the last 7 days for non-medical use?
6. Have you gotten a prescription stimulant in the last year days for non-medical use?
7. Have you gotten a prescription morphine in the last 90 days for non-medical use?

While the first three questions are the strongest indicators of drug misuse based on our model, the last four questions indicate that the drugs most associated with misuse are morphine products, stimulants and benzodiazepines.