Building and deploying a model with the Causata R package

Justin Hemann

Version 4.1-0, May 14, 2013

This vignette shows how functions in the Causata R package are used to build a logistic regression model in R and deploy it to Causata.

A simple example

A very simple example is shown below, illustrating a complete model build process from data extraction to model migration. Note that the code below is not entirely executable since a Causata server is required to extract the data (the GetData command) and upload the model (the UploadModel command).

The first step is to extract data using the Causata SQL interface. The focal point is set at page view events, meaning that the variables are built with respect to customers' page views. The variables returned in the query are:

- mortgage$_{a}application_{a}pproved_{n}30disabooleanvariableindicatingifamortgageapplicationisapprovedinthe30daysafterapa$
- web$_{h}as_{b}rowsed_{o}n_{i}pad_{l}7disabooleanvariableindicatingifacustomerhasvisitedthewebsitewithaniPadinthe7daysbeforethepagevie$

```
library(Causata)

# Load data using the SQL interface.
# A page view event is selected as a focal point.
conn <- Connect(group='fsdemo')
query <- paste(
  'SELECT mortgage_application_approved__n7d, total_assets__AP,'
  'web_has_browsed_on_ipad__l7d',
  'FROM Scenarios S, page_view P',
  'WHERE where S.profile_id = P.profile_id',
  'AND S.focal_point = P.timestamp'
  'LIMIT 20000'
df <- GetData(conn, query)
```

Next a formula is constructed, missing values are removed, and a glmnet model is trained. The model predicts the probability of a mortgage approval in the 30 days following a page view as a function of total assets and whether a customer has used an iPad.

```
formula.str <- 'mortgage_application_approved__n30d ~ total_assets__AP + web_has_browsed_on_ipad__AP'

# Preprocess data, track changes for export.  Remove missing values.
causataData <- CausataData(df, 'mortgage_application_approved__n30d')
causataData <- CleanNaFromContinuous(causataData)
causataData <- CleanNaFromFactor(causataData)

# build a model
modelmatrix <- model.matrix(formula(formula.str), data=causataData$df)
cv.glmnet.obj <- cv.glmnet(modelmatrix,
  causataData$df$mortgage_application_approved__n30d, alpha=0.0, family='binomial')
```

Last, the model is given a variable name and uploaded to Causata.

```
# upload model
model.def <- ModelDefinition(
 cv.glmnet.obj, causataData, formula(formula.str), cv.glmnet.obj$lambda.min)
variable.def <- VariableDefinition(
 name='score-mortgage-application', display.name='Score: Mortgage Application',
 description='Logistic regression model for demo.', author='Justin',
 labels='Scores')
result <- UploadModel(CausataConfig(group='fsdemo'), model.def, variable.def)
```

A longer example

The R code shown below is all executable provided that the Causata R package is installed.

Extract and load data

The first step in a model building process is to extract and load model training data. In Causata the easiest way to extract data is to use the Causata-SQL interface, which requires a connection to a Causata server. This section of the vignette is intended to work without a connection, so an example data set from the Causata package is used instead of the SQL interface. The example data set includes records for individuals who were shown a mobile banner ad.

First packages are loaded, along with the example data. The data resides in a data frame `df.causata`.

```
library(Causata)

## Warning:  package 'Matrix' was built under R version 2.15.3
## Warning:  package 'lattice' was built under R version 2.15.3
## Warning:  replacing previous import 'reset' when loading 'RCurl'

library(pROC)
```

```
# Set a random seed so random numbers are repeated
set.seed(87352)

# Load example data from the Causata package
data(df.causata)
```

Next a dependent variable is created. This example uses binary classification, so the dependent variable is 1 if a user clicks a mobile banner ad, and 0 otherwise.

```
# Set the dependent variable
dvname <- "has.responded.mobile.logoff_next.hour_466"
dv <- rep(0, nrow(df.causata))
dv[df.causata[[dvname]] == "true"] <- 1

# Remove unwanted variables from the data frame
df.causata[[dvname]] <- NULL
```

Preprocessing

Now that the data is loaded into R, the next step is to preprocess the data. Variables with no variation (all zero, all missing, etc.) must be removed. The loop shown below removes factors with a single level and numeric columns with a single value.

```
# Loop to remove variables with no variation
for (col in colnames(df.causata)){
  if (length(unique(df.causata[[col]])) == 1){
    # Single valued variable, remove it
    df.causata[[col]] <- NULL
  }
}
```

Next a `CausataData` object is created, which tracks preprocessing steps that will be repeated during model scoring.

```
# Create a CausataData object
causataData <- CausataData(df.causata, dependent.variable=dv)
```

Missing values

Two functions from the Causata R package are used to replace missing values in factors (categorical variables) and in numeric variables. The replacement values are stored in `causataData`.

```
# Replace missing values
causataData <- CleanNaFromContinuous(causataData)
causataData <- CleanNaFromFactor(causataData)
```

Categorical variables

Categorical variables with many levels will generate many dummy variables, which can be problematic during model training. One way to mitigate this problem is to merge smaller levels into an "Other" level. In the example below, the number of levels is capped at 15, and the mapping of original levels to the new levels is stored in `causataData`.

```
# Merge levels in factors with # levels exceeding a threshold
causataData <- MergeLevels(causataData, max.levels=15)
```

Outliers

The final preprocessing step illustrated here is the replacement of outliers. In this example any variable with `online.average.authentications.per.month` in the name will have values above 200 replaced with 200.

```
# Replace outliers in authentication count variables
for (col in grep("^online.average.authentications.per.month",
                 names(causataData$df), value=TRUE)){
  causataData <- ReplaceOutliers(causataData, col, upperLimit=200)
}
```

Other preprocessing steps

Other preprocessing steps may include:

- **Removing collinear variables:** Highly correlated variables can make model interpretation and variable importance calculations more difficult. If multiple variables are correlated then selecting only one for the model may be beneficial.
- **Linearization and discretization:** Numeric variables with a nonlinear relationship with the log-odds of the dependent variable may benefit from a linearizing transformation, such as the Weight of Evidence (WOE) transformation. Causata supports the discretization of continuous variables where continuous values are discretized into bins and mapped to a prescribed value.
- **Interaction terms:** Interaction terms are supported during scoring. They can be added to the model formula in R and will automatically be translated into the Causata model.

Train a glmnet model

Next the `model.matrix` function will be used to convert the data frame into a design matrix where factors are encoded as dummy variables. This step requires

a model formula, which is simply the addition of all independent variables in the data frame.

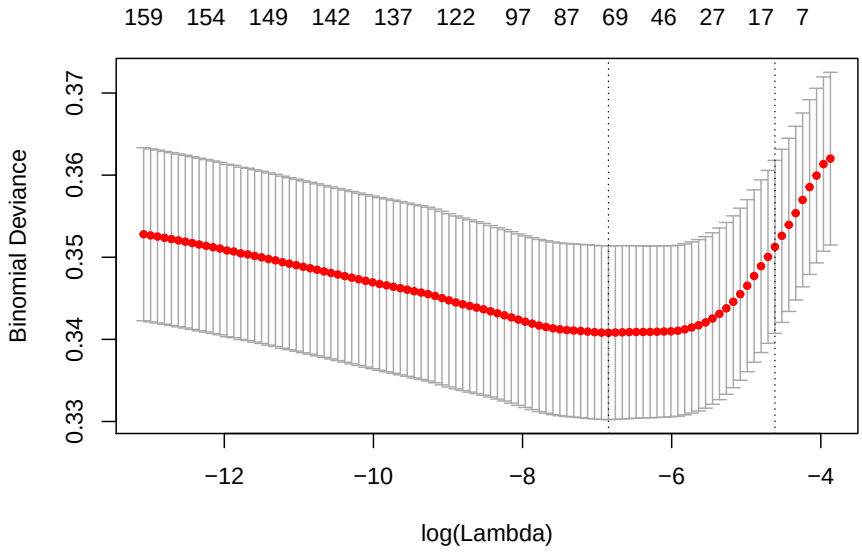
```
# Extract a set of independent variable names
indep.vars <- colnames(causataData$df)
indep.vars <- indep.vars[!(indep.vars == "dependent.variable") ]
# Build a formula string
formula.string <- paste("dependent.variable", "~",
                        paste(indep.vars, collapse=" + "))
formula.object <- formula(formula.string)
# Build a model matrix
x.matrix <- model.matrix(formula.object, data=causataData$df)
```

The design matrix is split into two matrices for model training and testing / validation. Here 75% of the rows are selected at random for the training data, and the remainder are used for testing.

```
# Split into training and testing data
idx.split <- sample(1:nrow(x.matrix), round(0.75 * nrow(x.matrix)))
x.matrix.train <- x.matrix[ idx.split, ]
x.matrix.test  <- x.matrix[-idx.split, ]
y.train <- causataData$df$dependent.variable[ idx.split ]
y.test  <- causataData$df$dependent.variable[-idx.split ]
```

Next the training data is supplied to `cv.glmnet` (from the `glmnet` package), which builds multiple models while varying the regularization parameter Lambda. The `plot` command generates a diagnostic plot of the model error (deviance) with respect to Lambda. The error bars indicate the standard error across the ten-fold cross validation, which appears to be high with respect to the overall change in the mean deviance. The standard error could be reduced by using a larger data set or reducing multicollinearity in the training data.

```
# Build model, select alpha value near 1
# since we expect most coefficients to be zero
cv.glmnet.obj <- cv.glmnet(x.matrix.train, y.train, alpha=0.8,
                           family=c("binomial"))
plot(cv.glmnet.obj)
```



Now the model is applied to the training and test data. Predicted values are calculated for each data set.

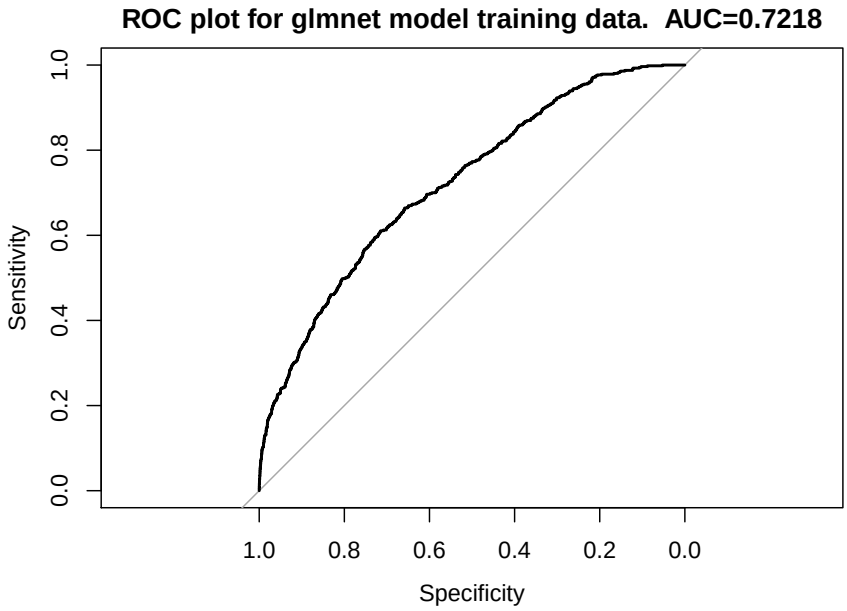
```
# Use the model to predict responses for training / testing data
predicted.train <- predict(cv.glmnet.obj, newx=x.matrix.train,
                           type="response", s="lambda.min")
predicted.test  <- predict(cv.glmnet.obj, newx=x.matrix.test,
                           type="response", s="lambda.min")
```

The next step is to assess the model accuracy using the area under the ROC curve. A value of 0.72 is considered fair, though not strong. Ideally the area under the ROC curve will be very close for the train / test data sets, which indicates a robust model that is not overfit. In this example the areas are within 2%.

```
# Compute area under the ROC curve using the pROC package
roc.train <- roc(y.train, predicted.train)
roc.test  <- roc(y.test,  predicted.test )
cat("Training / testing area under ROC curve: ",
    roc.train$auc, ", ", roc.test$auc, "\n")

## Training / testing area under ROC curve:  0.7218 ,  0.7095

plot(roc.train,
     main=sprintf("ROC plot for glmnet model training data.  AUC=%6.4f",
     roc.train$auc))
```




```
##
## Call:
## roc.default(response = y.train, predictor = predicted.train)
##
## Data: predicted.train in 12118 controls (y.train 0) < 560 cases (y.train 1).
## Area under the curve: 0.722
```

In the final steps the model name and description are set, and the Causata server parameters are provided so that the model can be uploaded. The final command `UploadModel` is commented since it requires a live connection to a Causata server.

```
# Prepare to import the model into Causata
model.def <- ModelDefinition(
  cv.glmnet.obj,
  causataData,
  formula.object,
  cv.glmnet.obj$lambda.min )
variable.def <- VariableDefinition(
  name = "score-test-model",
  display.name = "Score: Test Model",
  description = "A logistic regression model trained with sampledata.",
  author = "Test User" )

# Generate a string of PMML representing the model and
# preprocessing transformations.
# This step is not required to upload a model,
```

```
# it's shown for illustration purposes only
pmml.string <- ToPmml(model.def, variable.def)

# Upload model to Causata.
# The parameters below are for illustration only.
causata.config <- CausataConfig(
  config.server.host = "123.456.789.10",
  config.server.port = 8002,
  config.username = "testuser",
  cinfig.password = "1234abcd")

## Error:  unused argument(s) (cinfig.password = "1234abcd")
```

The UploadModel command requires a connection to a Causata server. The command creates a new variable in Causata that is available immediately for scoring.

```
result <- UploadModel(causata.config, model.def, variable.def)
```

The model coefficients exported to Causata will match the set obtained from the `cv.glmnet` object. This code extracts the nonzero coefficients.

```
# extract nonzero coefficients
coefs.all <- as.matrix(coef(cv.glmnet.obj, s="lambda.min"))
idx <- as.vector(abs(coefs.all) > 0)
coefs.nonzero <- as.matrix(coefs.all[idx])
rownames(coefs.nonzero) <- rownames(coefs.all)[idx]
```

The model coefficients are printed below.

```
print(coefs.nonzero)
```

```
##                                                                                  [,1]
## (Intercept)                                                                -2.932e+00
## has.decision_last.7.days_266true                                           -1.628e-01
## has.decision.logout.prompt_all.past_349true                                -2.083e-02
## has.decision.logout.prompt_last.30.days_349true                            -6.025e-01
## has.responded.cd.logoff_all.past_469true                                   -1.159e-01
## has.responded.credit.cards.logoff_last.30.days_464true                      7.776e-01
## has.responded.credit.cards.logoff_last.7.days_464true                       8.011e-01
## has.responded.mobile.logoff_all.past_466true                                1.153e+00
## has.responded.mobile.logoff_last.30.days_466true                            1.076e+00
## has.responded.mobile.logoff_last.7.days_466true                            -6.603e-01
## has.responded.mortgages.logoff_last.7.days_467true                          4.667e-01
## ib.at.least.60.days.since.registration_focal.point_271true                 -2.037e-01
## ib.has.enrollment.event_all.past_145true                                   -5.102e-01
## ib.has.forgotten.password_last.30.days_417true                              6.728e-01
## mobile.has.any.mobile.app.activity_all.past_324true                        -1.779e-01
## mobile.has.any.mobile.app.activity_last.7.days_324true                     -2.038e-01
## mobile.has.used.sms_all.past_43true                                         1.274e-01
## mobile.has.used.sms_last.30.days_43true                                     2.951e-01
## online.average.authentications.per.month_all.past_406                      -2.404e-02
## online.has.online.activity_last.7.days_49true                               1.028e+00
## online.has.used.mobile.device_all.past_418true                              3.549e-01
## online.has.used.mobile.device_last.7.days_418true                           7.014e-01
## online.most.frequent.browser.type_all.past_432Internet Explorer             2.425e-02
## online.most.frequent.browser.type_all.past_432Safari                       -2.291e-01
## online.most.frequent.browser.type_last.7.days_432Firefox                   -6.788e-02
## online.most.frequent.browser.type_last.7.days_432Safari                     1.392e-01
## online.most.frequent.page.category_all.past_433Internet Banking             9.525e-02
## online.most.frequent.page.category_all.past_433Locator                     -4.001e-01
## online.most.frequent.page.category_last.30.days_433Internet Banking         3.989e-01
## online.most.frequent.page.category_last.30.days_433Locator                 -3.017e-02
## online.most.frequent.page.category_last.7.days_433Internet Banking          1.714e-02
```

```
## online.most.frequent.page.category_last.7.days_433Rates                                      -1.107e-01
## online.most.frequent.page.category_last.7.days_433VS Logout                                  -1.737e-01
## online.most.frequent.page.category_last.7.days_433BLANK                                      -9.350e-01
## online.most.frequent.product.category.viewed_all.past_434Credit and Lending                  -1.450e-01
## online.most.frequent.product.category.viewed_all.past_434Deposit Account Rates               -4.606e-01
## online.most.frequent.product.category.viewed_all.past_434Home Equity Loans and Lines         -1.654e-01
## online.most.frequent.product.category.viewed_all.past_434Mortgage Loan Rates                  4.141e-01
## online.most.frequent.product.category.viewed_all.past_434Products and Services               8.430e-02
## online.most.frequent.product.category.viewed_all.past_434Protection Center                   8.702e-01
## online.most.frequent.product.category.viewed_all.past_434Rates                               4.136e-01
## online.most.frequent.product.category.viewed_all.past_434Savings Center                     -2.481e+00
## online.most.frequent.product.category.viewed_all.past_434Student Center                     -9.946e-01
## online.most.frequent.product.category.viewed_last.30.days_434Checking Center                -3.630e-01
## online.most.frequent.product.category.viewed_last.30.days_434Deposit Account Rates          -4.330e-01
## online.most.frequent.product.category.viewed_last.30.days_434Home Equity Loans and Lines -5.663e-01
## online.most.frequent.product.category.viewed_last.30.days_434Mortgage Loan Rates             1.823e-01
## online.most.frequent.product.category.viewed_last.30.days_434Products and Services          3.027e-01
## online.most.frequent.product.category.viewed_last.30.days_434Protection Center             -6.552e-01
## online.most.frequent.product.category.viewed_last.30.days_434Rates                           1.379e-01
## online.most.frequent.product.category.viewed_last.30.days_434BLANK                          -5.681e-03
## online.most.frequent.product.category.viewed_last.7.days_434Checking Center                  7.997e-01
## online.most.frequent.product.category.viewed_last.7.days_434Consumer Loan Rates             -5.907e-01
## online.most.frequent.product.category.viewed_last.7.days_434Credit and Lending              -6.994e-01
## online.most.frequent.product.category.viewed_last.7.days_434Products and Services            1.896e-01
## online.most.frequent.product.category.viewed_last.7.days_434Protection Center               -1.301e+00
## online.most.frequent.product.category.viewed_last.7.days_434Rates                           -2.885e-02
## online.most.frequent.product.category.viewed_last.7.days_434Savings Center                   4.353e+00
## online.product.category.view.count_last.7.days_443                                           -3.774e-02
## online.product.category.view.count.checking_all.past_445                                     8.692e-02
## online.product.category.view.count.checking_last.7.days_445                                  6.129e-02
## online.product.category.view.count.credit.lending_last.30.days_446                           -1.664e-01
## online.product.category.view.count.credit.lending_last.7.days_446                            -1.638e-01
## online.product.category.view.count.mobile.banking_all.past_447                               -6.514e-03
## online.product.category.view.count.mobile.banking_last.7.days_447                            2.297e-01
## online.product.category.view.count.savings.cds_last.7.days_448                               -2.444e-01
## online.time.since.last.authentication_focal.point_408                                        -6.923e-11
## os.os.disclosure.accepted_all.past_388true                                                   -1.546e-01
## os.os.disclosure.accepted_last.30.days_388true                                               1.299e+00
## os.os.disclosure.accepted_last.7.days_388true                                                -4.842e-01
## time.since.last.authentication_focal.point_472                                               -3.584e-11
## time.since.last.session_focal.point_404                                                      -2.996e-09
```