Spark Machine Learning in Jupyter

Step 4- Declare the model that we want to use

The model here is Naive Bayes. It will output each prediction into a 'prediction' column. Naive Bayes is a probabistic model that learns based on previous decisions. We will take a best guess at the paramater 'smoothing'- SparkML will help us tune it later!

In [16]:
```
nb = NaiveBayes(smoothing=1.0, modelType="multinomial", labelCol="label", predictionCol="prediction")
```

Step 5 - Setup the Pipeline

The pipeline is the guts of the algorithm that strings all the work we've done together.

The stages are run in order and the input DataFrame is transformed as it passes through each stage. First, comes the feature transformations, then the assembler to put them togather into one DF. We pass that into the model.

In machine learning, it is common to run a sequence of algorithms to process and learn from data, so this can get as complex as we want to make it!

In [17]:
```
pipeline = Pipeline(stages=[labelIndexer,occupationIndexer,countryIndexer, genderIndexer, yearOfBirthIndexer, vecAssembler, normalizer, nb, cc
```

Step 6 - Train the model

We will split it into training data which is marked and test data which will be used to test the efficiency of the algorithms.

It is common to split the split up the data randomly into 70% for training and 30% for testing. If we were to use a bigger test set, we might use an 80% / 20% split.

In [18]:
```
train, test = LabeledVettingData.randomSplit([70.0,30.0], seed=1)
train.cache()
test.cache()
print('The number of records in the training data set is {}.'.format(train.count()))
print('The number of rows labeled high is {}.'.format(train.filter(train['VETTING_LEVEL'] == 10).count()))
print('The number of rows labeled medium is {}.'.format(train.filter(train['VETTING_LEVEL'] == 20).count()))
print('The number of rows labeled low is {}.'.format(train.filter(train['VETTING_LEVEL'] == 30).count()))
print('')

print('The number of records in the test data set is {}.'.format(test.count()))
print('The number of rows labeled high is {}.'.format(test.filter(test['VETTING_LEVEL'] == 10).count()))
print('The number of rows labeled medium is {}.'.format(test.filter(test['VETTING_LEVEL'] == 20).count()))
print('The number of rows labeled low is {}.'.format(test.filter(test['VETTING_LEVEL'] == 30).count()))
```

SPARK JOB PROGRESS Show All

Application Development in RStudio

IBM Data Science Experience | Projects | Tools | Data Services | Community | US South | Docs

File Edit Code View Plots Session Build Debug Profile Tools Help | Disk usage: 18%

Go to file/function | Addins | Project: (None)

connection.R | dash-connection.Rmd | app.R | Run App

```
29    'Pending'
30  }
31
32  shinyApp(
33    ########################################################################
34    # UI                                                                   #
35    ########################################################################
36    ui = fluidPage(
37      shinythemes::themeSelector(),
38      tags$head(tags$style('body {background-color: #FFFFEF; }')),
39      theme = shinythemes::shinytheme('yeti'),
40      # Application title
41      titlePanel('Human Trafficking'),
42      sidebarLayout(
43        sidebarPanel(
44          width = 3,
45          plotlyOutput('vettingPie', height = 450),
46          conditionalPanel(
47            condition="(typeof input.tbl_rows_selected !== 'undefined' && input.tbl_rows_selected.length > 0)", hr(),
48            verbatimTextOutput('selectionDetails'),
49            wellPanel(
50              fluidRow(
51                column(
52                  width=6, radioButtons(
53                    'vetting', label='Vetting Level',
54                    choices=c('Pending' = 100, 'HIGH' = 10, 'MEDIUM' = 20, 'LOW' = 30)
55                  )
56                )
57              )
58            ),
59            actionButton('saveVetting', label = 'Save', icon = icon('save', lib = 'glyphicon')),
60            actionButton('entityProfile', label='Entity Profile', icon=icon('id-card-o'))
61          )
62        ),
63        mainPanel(
64          width = 9,
65          DT::dataTableOutput('tbl')
66        )
67      )
68    ),
69
70    ########################################################################
71    # SERVER                                                               #
```

109:11 | server(input, output, session) | R Script

Console

Environment | History | Spark

Import Dataset | List | Global Environment

data	1085 obs. of 25 variables
delays	25043 obs. of 10 variables
predicted	1085 obs. of 27 variables
probs	1154580 obs. of 6 variables
Values	
ch	Class 'RODBC' atomic [1:1] 1
con	-1L
con.text	"DRIVER=BLUDB;Database=BLUDB;Hostname=dashdb-entry-yp-dal09-…
conn	Class 'RODBC' atomic [1:1] 1
conn_path	"BLUDB;DATABASE=BLUDB;HOSTNAME=dashdb-entry-yp-dal09-09.serv…
conn.path	"BLUDB;DATABASE=BLUDB;HOSTNAME=dashdb-entry-yp-dal09-09.serv…

Files | Plots | Packages | Help | Viewer

New Folder | Upload | Delete | Rename | More

Home › Proof-of-Technology › DSX › Lab-3

Name	Size	Modified
..		
app.R	7.8 KB	Mar 27, 2017, 5:14 PM
connection.R	439 B	Mar 27, 2017, 5:20 PM
dashConnectAndInteractInR.nb.html	820.2 KB	Mar 27, 2017, 5:22 PM
dashConnectAndInteractInR.Rmd	4.2 KB	Mar 27, 2017, 5:22 PM