Data Science

Open Source Powered

# 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