Spark Lightning-fast cluster computing

Titanic - Modeling Survivorship

```
# Using dataset from the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/)
titanicDF <- read.csv('http://math.ucdenver.edu/RTutorial/titanic.txt',sep='\t')

# Creating new title feature from name field
titanicDF$Title <- ifelse(grepl('Mr ',titanicDF$Name),'Mr',ifelse(grepl('Mrs ',titanicDF$Name),'Mrs',ifelse(grepl('Miss',titanicDF$Name),'Miss','Nothing')))
titanicDF$Title <- as.factor(titanicDF$Title)

# Impute age to remove NAs
titanicDF$Age[is.na(titanicDF$Age)] <- median(titanicDF$Age, na.rm=T)

# Reorder data set so target is last column
titanicDF <- titanicDF[c('PClass', 'Age', 'Sex', 'Title', 'Survived')]

# dummy text fields
charcolumns <- names(titanicDF[sapply(titanicDF, is.factor)])
for (colname in charcolumns) {
        print(paste(colname,length(unique(titanicDF[,colname]))))
        for (newcol in unique(titanicDF[,colname])) {
                if (!is.na(newcol))
                        titanicDF[,paste0(colname,"_",newcol)] <- ifelse(titanicDF[,colname]==newcol,1,0)
        }
        titanicDF <- titanicDF[,setdiff(names(titanicDF),colname)]
}

# Split data into training and testing
set.seed(1234)
splitIndex <- base::sample(nrow(titanicDF), floor(0.75*nrow(titanicDF)))
trainDF <- titanicDF[ splitIndex,]
testDF <- titanicDF[-splitIndex,]

# Convert local data frame/RDD/etc to a SparkR DataFrame
train_titanic_sp <- createDataFrame(sqlContext, trainDF)
test_titanic_sp <- createDataFrame(sqlContext, testDF)
dim(train_titanic_sp)

# Fit a linear model over the dataset.
model <- glm(Survived~., data=train_titanic_sp, family='binomial')
predictions <- predict(model, newData = test_titanic_sp )
names(predictions)

predictions_details <- select(predictions, predictions$label, predictions$prediction)
registerTempTable(predictions_details, "predictions_details")

# Let's calculate the accuracy manually:
TP <- sql(sqlContext, "SELECT count(label) FROM predictions_details WHERE label = 1 AND prediction = 1")
TP <- collect(TP)[[1]]
TN <- sql(sqlContext, "SELECT count(label) FROM predictions_details WHERE label =
0 AND prediction = 0")
TN <- collect(TN)[[1]]
FP <- sql(sqlContext, "SELECT count(label) FROM predictions_details WHERE label =
0 AND prediction = 1")
FP <- collect(FP)[[1]]
FN <- sql(sqlContext, "SELECT count(label) FROM predictions_details WHERE label =
1 AND prediction = 0")
FN <- collect(FN)[[1]]
accuracy = (TP + TN)/(TP + TN + FP + FN)
print(paste0(round(accuracy * 100,2), '%'))
```

Kill your clusters!!

Just for fun, instead of killing our clusters from the AWS console window, here is the command to do it directly from the terminal:

```
    ./spark-ec2 -k udemy -i udemy.pem -r us-west-2  destroy my-spark-cluster
```