GRAD

STAT 149 Spring 2016 Final Project, Process Workbook #3

Submitted by: Kendrick Lo (Harvard ID: 70984997), Amy Lee (Harvard ID: 60984077)

This is a continuation of exploration of models from Process Workbooks 1 and 2, and focuses primarily on validation techniques.

Cross-validation

We implement cross-validation techniques to obtain a measure out-of-sample error. So far, we have been using Kaggle scores as feedback, but this is dangerous since we are making decisions on the test set, and thus effectively training on the test set.

Cross-validation can help to prevent overfitting and allow us to better compare different models. Failure to do this may leave us without an objective measure of how these models might do on unseen data.

Ideally, we would have done this with all the models that we were considering (by setting up the validation sets from the beginning). For the purpose of this assignment, we only performed this for select models below, and reported the validation error.

```
train = read.csv("~/Desktop/stat149/StatKaggle/train_add3.csv", header=TRUE)
test = read.csv("~/Desktop/stat149/StatKaggle/test_add3.csv", header=TRUE)
Id = read.csv("~/Desktop/stat149/StatKaggle/id.csv", header=TRUE)
```

We will perform 5-fold cross validation. As an exercise, we are setting up this manually.

```
# set up folds

breaks = c(8687, 17374, 26061, 34748)
n = 43436

valid1 = train[1:breaks[1],]
train1 = train[(breaks[1]+1):n,]
actual1 = as.integer(valid1$lapsed=="Y")
valid1$lapsed <- NULL

valid2 = train[(breaks[1]+1):breaks[2],]
train2 = rbind(train[1:breaks[1],], train[(breaks[2]+1):n,])
actual2 = as.integer(valid2$lapsed=="Y")
valid2$lapsed <- NULL

valid3 = train[(breaks[2]+1):breaks[3],]
train3 = rbind(train[1:breaks[2],], train[(breaks[3]+1):n,])
actual3 = as.integer(valid3$lapsed=="Y")
valid3$lapsed <- NULL

valid4 = train[(breaks[3]+1):breaks[4],]
train4 = rbind(train[1:breaks[3],], train[(breaks[4]+1):n,])
actual4 = as.integer(valid4$lapsed=="Y")
```

```
valid4$lapsed <- NULL

valid5 = train[(breaks[4]+1):n,]
train5 = train[1:breaks[4],]
actual5 = as.integer(valid5$lapsed=="Y")
valid5$lapsed <- NULL
```

```
# from Kaggle site
MultiLogLoss <- function(act, pred)
    {
      eps = 1e-15;
      nr <- nrow(pred)
      pred = matrix(sapply( pred, function(x) max(eps,x)), nrow = nr)
      pred = matrix(sapply( pred, function(x) min(1-eps,x)), nrow = nr)
      ll = sum(act*log(pred) + (1-act)*log(1-pred))
      ll = ll * -1/(nrow(act))
      return(ll);
    }
```

Logistic Regression

```
model26.glm <- glm(lapsed ~ age * (age_12_under + age_13_15 + age_16_18
                                   + age_19_24 + age_25_64 + age_65_plus)
                   + sex + region + nregions + memtype + mem_mag1 + mem_mag2
                   + hasemail + r1 + r2 + r3 + r.quick + extra + intl
                   + r.intl + allgames5yr * (games_0 + games_1_5 + games_6_10
                                             + games_11_20 + games_21_34
                                             + games_35_49 + games_50_plus)
                   + fastevents + medevents + slowevents + nfloor + age.na
                   + r1.na + r2.na + r3.na + r.quick.na + r.intl.na
                   + memmonths * (mon_less30 + mon_31 + mon_32 + mon_33
                                  + mon_34 + mon_35 + mon_36 + mon_37_60
                                  + mon_61_84 + mon_85_120 + mon_121_263
                                  + mon_264_plus)
                   + allgames1yr * (games_0 + games_1_5 + games_6_10
                                    + games_11_20 + games_21_34 + games_35_49
                                    + games_50_plus) + age:memtype
                   + memtype:r1 + sex:r1 + memtype:hasemail + age:sex
                   + memtype:hasemail:r1 + sex:hasemail:r1
                   + age:sex:memtype, family = "binomial", data = train1)

model26.newpred = predict(model26.glm, newdata=valid1, type="response")
preds = cbind(model26.newpred, 1-model26.newpred)
actuals = cbind(actual1, 1-actual1)
mll1 = MultiLogLoss(actuals, preds)

model26.glm <- glm(lapsed ~ age * (age_12_under + age_13_15 + age_16_18
                                   + age_19_24 + age_25_64 + age_65_plus)
                   + sex + region + nregions + memtype + mem_mag1 + mem_mag2
                   + hasemail + r1 + r2 + r3 + r.quick + extra + intl
                   + r.intl + allgames5yr * (games_0 + games_1_5 + games_6_10
```

```
                                        + games_11_20 + games_21_34
                                        + games_35_49 + games_50_plus)
                    + fastevents + medevents + slowevents + nfloor + age.na
                    + r1.na + r2.na + r3.na + r.quick.na + r.intl.na
                    + memmonths * (mon_less30 + mon_31 + mon_32 + mon_33
                                   + mon_34 + mon_35 + mon_36 + mon_37_60
                                   + mon_61_84 + mon_85_120 + mon_121_263
                                   + mon_264_plus)
                    + allgames1yr * (games_0 + games_1_5 + games_6_10
                                     + games_11_20 + games_21_34 + games_35_49
                                     + games_50_plus) + age:memtype
                    + memtype:r1 + sex:r1 + memtype:hasemail + age:sex
                    + memtype:hasemail:r1 + sex:hasemail:r1
                    + age:sex:memtype, family = "binomial", data = train2)

model26.newpred = predict(model26.glm, newdata=valid2, type="response")
preds = cbind(model26.newpred, 1-model26.newpred)
actuals = cbind(actual2, 1-actual2)
mll2 = MultiLogLoss(actuals, preds)

model26.glm <- glm(lapsed ~ age * (age_12_under + age_13_15 + age_16_18
                                   + age_19_24 + age_25_64 + age_65_plus)
                    + sex + region + nregions + memtype + mem_mag1 + mem_mag2
                    + hasemail + r1 + r2 + r3 + r.quick + extra + intl
                    + r.intl + allgames5yr * (games_0 + games_1_5 + games_6_10
                                        + games_11_20 + games_21_34
                                        + games_35_49 + games_50_plus)
                    + fastevents + medevents + slowevents + nfloor + age.na
                    + r1.na + r2.na + r3.na + r.quick.na + r.intl.na
                    + memmonths * (mon_less30 + mon_31 + mon_32 + mon_33
                                   + mon_34 + mon_35 + mon_36 + mon_37_60
                                   + mon_61_84 + mon_85_120 + mon_121_263
                                   + mon_264_plus)
                    + allgames1yr * (games_0 + games_1_5 + games_6_10
                                     + games_11_20 + games_21_34 + games_35_49
                                     + games_50_plus) + age:memtype
                    + memtype:r1 + sex:r1 + memtype:hasemail + age:sex
                    + memtype:hasemail:r1 + sex:hasemail:r1
                    + age:sex:memtype, family = "binomial", data = train3)

model26.newpred = predict(model26.glm, newdata=valid3, type="response")
preds = cbind(model26.newpred, 1-model26.newpred)
actuals = cbind(actual3, 1-actual3)
mll3 = MultiLogLoss(actuals, preds)

model26.glm <- glm(lapsed ~ age * (age_12_under + age_13_15 + age_16_18
                                   + age_19_24 + age_25_64 + age_65_plus)
                    + sex + region + nregions + memtype + mem_mag1 + mem_mag2
                    + hasemail + r1 + r2 + r3 + r.quick + extra + intl
                    + r.intl + allgames5yr * (games_0 + games_1_5 + games_6_10
                                        + games_11_20 + games_21_34
                                        + games_35_49 + games_50_plus)
                    + fastevents + medevents + slowevents + nfloor + age.na
```

```
                    + r1.na + r2.na + r3.na + r.quick.na + r.intl.na
                    + memmonths * (mon_less30 + mon_31 + mon_32 + mon_33
                                   + mon_34 + mon_35 + mon_36 + mon_37_60
                                   + mon_61_84 + mon_85_120 + mon_121_263
                                   + mon_264_plus)
                    + allgames1yr * (games_0 + games_1_5 + games_6_10
                                     + games_11_20 + games_21_34 + games_35_49
                                     + games_50_plus) + age:memtype
                    + memtype:r1 + sex:r1 + memtype:hasemail + age:sex
                    + memtype:hasemail:r1 + sex:hasemail:r1
                    + age:sex:memtype, family = "binomial", data = train4)

model26.newpred = predict(model26.glm, newdata=valid4, type="response")
preds = cbind(model26.newpred, 1-model26.newpred)
actuals = cbind(actual4, 1-actual4)
mll4 = MultiLogLoss(actuals, preds)

model26.glm <- glm(lapsed ~ age * (age_12_under + age_13_15 + age_16_18
                                   + age_19_24 + age_25_64 + age_65_plus)
                    + sex + region + nregions + memtype + mem_mag1 + mem_mag2
                    + hasemail + r1 + r2 + r3 + r.quick + extra + intl
                    + r.intl + allgames5yr * (games_0 + games_1_5 + games_6_10
                                              + games_11_20 + games_21_34
                                              + games_35_49 + games_50_plus)
                    + fastevents + medevents + slowevents + nfloor + age.na
                    + r1.na + r2.na + r3.na + r.quick.na + r.intl.na
                    + memmonths * (mon_less30 + mon_31 + mon_32 + mon_33
                                   + mon_34 + mon_35 + mon_36 + mon_37_60
                                   + mon_61_84 + mon_85_120 + mon_121_263
                                   + mon_264_plus)
                    + allgames1yr * (games_0 + games_1_5 + games_6_10
                                     + games_11_20 + games_21_34 + games_35_49
                                     + games_50_plus) + age:memtype
                    + memtype:r1 + sex:r1 + memtype:hasemail + age:sex
                    + memtype:hasemail:r1 + sex:hasemail:r1
                    + age:sex:memtype, family = "binomial", data = train5)

model26.newpred = predict(model26.glm, newdata=valid5, type="response")
preds = cbind(model26.newpred, 1-model26.newpred)
actuals = cbind(actual5, 1-actual5)
mll5 = MultiLogLoss(actuals, preds)

(mll1 + mll2 + mll3 + mll4 + mll5)/(5*2)  # log loss sums both cols
```

Validation error: 0.540

Generalized Additive Model (code hidden)

Validation error: 0.539

Random Forests (code hidden)

Model 15b: Validation error: 0.563 Model 15c: Validation error: 0.558 Model 15d: Validation error: 0.558 Model 15e: Validation error: 0.551

Gradient Boosting Models (code hidden)

400 trees: 0.648 / interaction.depth=3: 0.630 600 trees: 0.639 800 trees / interaction.depth=3: 0.605 1000 trees / interaction.depth=3: 0.592

I'm not sure I trust the results of the GBM model.

Neural Network (code hidden)

NN #3: 0.537 (really long time)

These values are generally consistent with what we have been getting as public Kaggle test scores, which alleviates some of our concerns about depending too heavily on the test score results.