Capstone Project: ADNI Data - Markdown 2

Brian Collica, Ben Searchinger, Ryan Roggenkemper, James Koo

3/30/2021

Model Description

Our model consists of two stages. Stage one tries to predict β-amyloid positivity for a given patient. Research has shown that a β-amyloid positive status is highly correlated with Alzheimer's Disease, but testing for positivity is costly and invasive to the patient. We hope to cut down on these costs by accurately predicting a patient's status using other easily accessible data.

Stage two is a deep learning classifier that is fit to tau PET brain scans in order to classify a patient as having AD-related tau pathology or normal tau levels. The spatial arrangement of the tau protein in the brain has been a topic of recent AD research, and we hope that the deep learning classifier can pick up on these subtle arrangements.

The ultimate goal is to effectively sort patients which are at high risk for AD using the first stage, and then accurately classify their AD status based on the PET brain scans. The idea is that a patient would only receive a PET scan if they are labeled as β-amyloid in the first stage. If successful, we hope this technique can cut down on unnecessary costly procedures which are invasive and potentially harmful to the patient while also maintaining a high level of accuracy in quantifying an individual's AD risk.

We plan to evaluate our success with the following metrics:

- First Stage True Positive, False Negative, Misclassification Rates, and Correlation with AD Diagnosis.
- Second Stage Accuracy, True Positive, False Negative, & Misclassification Rates
- Combined Model Accuracy, True Positive, False Negative, & Misclassification Rates

Ideally, we want a high proportion of the subjects predicted as β-amyloid positive to be diagnosed with AD or MCI. In particular, we want the proportion of AD/MCI subjects among those predicted as β-amyloid positive to be higher than the proportion in the general population. That being said, some other metrics of interest will be the distribution of characteristics among those labeled as β-amyloid negative in the first stage, and if there is any particular information which can be used to fine tune the model.

First Stage Initial Fits

Description

For the first stage model, we used the ADNIMERGE dataset available on the ADNI website. This dataset includes many key variables of interest for each ADNI participant. The dataset was filtered to include only patients who had cerebrospinal-fluid measurements taken at baseline. Each patient's amyloid measurements were measured in a lab at the University of Pennsylvania and included in the UPENNBIOMK_MASTER dataset, also available on the ADNI website. We combined the two datasets to get 1,087 unique baseline observations.

The two initial models considered were (1) a linear regression to predict `beta_csf` - the specific level of β-amyloid for a given patient, and (2) a logistic regression to predict positivity status, `beta_pos`. The variable `beta_pos` was coded as 1 if an individual had `beta_csf` less than 192 and 0 otherwise. The level of 192 is a well-established clinical cutoff point for determining β-amyloid positivity. In the linear regression model, patients were classified as β-amyloid positive if their *predicted* amyloid level was below 192.

The linear and logistic models were both fit using the following variables known to be linked to AD:

- Age
- Sex (1 for Male 0 for Female)
- Immediate Memory Recall Score in the Rey Auditory Verbal Learning Test (RAVLT)
- Presence of APO$\epsilon - 4$ Gene
  - `APOE$_1` $= 1$ if only one allele is present
  - `APOE$_2` $= 1$ if two alleles are present

Both models were also fit again with the addition of two variables containing brain volumetrics estimated from MRI scans:

- Whole Brain Volume
- Hippocampus Volume

Code and Output

The first stage models are fit using R and requires the following packages: `readr`, `dplyr`, `stringr`, and `origami`. The following code reads in the data, filters by baseline visit, and renames the baseline variables of interest.

```r
# Load Packages ####
library(readr)
library(dplyr)
library(stringr)
library(origami)

# Read Data ####
beta_tau <- read_csv("AmyloidData/beta_tau_csf_new_vars.csv")
source("cv_functions.R")

# Filter By Baseline Visit ####
baseline <- beta_tau %>%
  filter(VISCODE == "bl")

bl_na1 <- which(is.na(baseline$beta_pos) | is.na(baseline$RAVLT_immediate_bl) |
                  is.na(baseline$Hippocampus) | is.na(baseline$WholeBrain))
bl_na2 <- which(is.na(baseline$beta_pos) | is.na(baseline$RAVLT_immediate_bl))

# Rename _bl Variables ####
# Rename _bl Variables ####
bl_rm <- c(27, 54, 55)
baseline_1 <- baseline[-(bl_na1), -(bl_rm)]
```

```r
baseline_1 <- rename(
  baseline_1,
  RAVLT_immediate = RAVLT_immediate_bl,
  Hippocampus = Hippocampus_bl,
  WholeBrain = WholeBrain_bl)

baseline_2 <- baseline[-(bl_na2), -(bl_rm)]
baseline_2 <- rename(
  baseline_2,
  RAVLT_immediate = RAVLT_immediate_bl,
  Hippocampus = Hippocampus_bl,
  WholeBrain = WholeBrain_bl)
```

The linear and logistic fits using all the baseline data can be seen below.

```r
# Initial Model Fits ####
# Including MRI data
bl_cont <- lm(beta_csf ~ AGE + MALE + APOE4_1 + APOE4_2 + RAVLT_immediate
              + Hippocampus + WholeBrain,
              data = baseline_1)
bl_logit <- glm(beta_pos ~ AGE + MALE + APOE4_1 + APOE4_2 + RAVLT_immediate
                + Hippocampus + WholeBrain,
                data = baseline_2, family = "binomial")

# Excluding MRI Data
bl_cont2 <- lm(beta_csf ~ AGE + MALE + APOE4_1 + APOE4_2 + RAVLT_immediate,
               data = baseline_1)
bl_logit2 <- glm(beta_pos ~ AGE + MALE + APOE4_1 + APOE4_2 + RAVLT_immediate,
                 data = baseline_2, family = "binomial")
```

5-fold crass-validation was then run for each of the four initial baseline models. The following code performs the cross-validation and also calculates performance metrics of interest. The metrics calculated include:

- Misclassification Rate: Proportion of incorrect predictions
- True Positive Rate: Proportion of β-amyloid positive cases correctly classified
- False Positive Rate: Proportion of β-amyloid negative cases classified as positive
- True Negative Rate: Proportion of β-amyloid negative cases correctly classified
- False Negative Rate: Proportion of β-amyloid positive cases classified as negative
- DX Correlation: Correlation between positive prediction status and being diagnosed as AD at baseline
- Positive AD: Number of subjects predicted as β-amyloid positive who were also diagnosed with AD
- Negative AD: Number of subjects predicted as β-amyloid negative who were also diagnosed with AD
- Percent Positive AD: Percentage of positive predictions who were also diagnosed with AD

Specific functions needed for the cross-validation and metrics calculations are defined in the `cv_functions.R` script.

```r
# 5-Fold Cross Validation ####
# Make Folds
bl_folds1 <- folds_vfold(nrow(baseline_1), V = 5)
bl_folds2 <- folds_vfold(nrow(baseline_2), V = 5)

# Linear Models ####
bl_cv_lm <- origami::cross_validate(
  cv_fun = cv_lm, folds = bl_folds1, data = baseline_1,
  reg_form = "beta_csf ~ AGE + MALE + APOE4_1 + APOE4_2 + RAVLT_immediate + Hippocampus + WholeBrain")

bl_cv_lm2 <- origami::cross_validate(
  cv_fun = cv_lm, folds = bl_folds2, data = baseline_2,
  reg_form = "beta_csf ~ AGE + MALE + APOE4_1 + APOE4_2 + RAVLT_immediate")

# Logistic Models ####
bl_cv_logit <- origami::cross_validate(
  cv_fun = cv_logit, folds = bl_folds1, data = baseline_1,
  reg_form = "beta_pos ~ AGE + MALE + APOE4_1 + APOE4_2 + RAVLT_immediate + Hippocampus + WholeBrain")

bl_cv_logit2 <- origami::cross_validate(
  cv_fun = cv_logit, folds = bl_folds2, data = baseline_2,
  reg_form = "beta_pos ~ AGE + MALE + APOE4_1 + APOE4_2 + RAVLT_immediate")

# Linear Models Stats ####
bl_lm_stats1 <- colMeans(dplyr::bind_rows(bl_cv_lm$c_stats))
bl_lm_stats2 <- colMeans(dplyr::bind_rows(bl_cv_lm2$c_stats))

# Logistic Model Stats ####
bl_log_stats1 <- colMeans(dplyr::bind_rows(bl_cv_logit$c_stats))
bl_log_stats2 <- colMeans(dplyr::bind_rows(bl_cv_logit2$c_stats))

# Combine Data ####
bl_summary <- dplyr::bind_rows(
        bl_lm_stats1, bl_lm_stats2, bl_log_stats1, bl_log_stats2)
row.names(bl_summary) <- c(
        "Linear w/MRI", "Linear", "Logistic w/MRI", "Logistic")
colnames(bl_summary) <- c(
        "Misclass", "T Pos", "F Pos", "T Neg", "F Neg",
        "DX Corr", "Pos AD", "Neg AD", "% Pos AD")
```

Table 1: Cross-Validation Results

	Misclass	T Pos	F Pos	T Neg	F Neg	DX Corr	Pos AD	Neg AD	% Pos AD
Linear w/MRI	0.2303	0.9120	0.2028	0.4084	0.3589	0.2188	36.0	0.4	0.2434
Linear	0.2294	0.9239	0.2060	0.3758	0.3388	0.2166	44.2	0.2	0.2470
Logistic w/MRI	0.2669	0.7069	0.0999	0.8009	0.4846	0.3675	33.8	2.6	0.3303
Logistic	0.2650	0.7093	0.0946	0.8068	0.4821	0.3479	40.0	4.4	0.3338

A small subset of patients also had CSF measurements taken at other visits including 12-month and 24-month follow-ups, but the small sample size led us to omit these particular observations.

Second Stage Initial Fits

Description

Code and Output

Discussion and Follow-up