BIOST 544: Homework 3

Department of Biostatistics @ University of Washington

2024-11-11

High-Dimensional Predictive Models

Problem

The existence of a large number of necrotic cells in a tumor can be indicative of a successfully mounted immune defense. We might be interested in understanding biomolecular pathways regulated/dysregulated in a tumor that make it more/less susceptible to the immune system. By identifying genes with expression (in the tumor microenvironment) related to quantity of necrotic tumor tissue, we might hope to a) build a better picture of the biology of immune regulation/dysregulation in cancer and/or b) find potential targets for therapy. In this homework we would like investigate the relationship between gene-expression values in the tumor and the existence and extent of necrotic tissue.

To evaluate this we will again work with the `NOAH` data. The data can be found on the course website in the following files:

- **clinical_data.csv** contains the clinical/phenotypic information.
- **expression_data_probeID.csv** contains the expression information (by probe set).
- **annotation.csv** contains the gene name identifiers corresponding to each probe set.

To get more information on the probesets, feel free to look into the affymetrix human 133 plus 2.0 array annotation (on the affymetrix site). In particular, the variable `necrotic_cells.pct` (the percentage of necrotic tissue in a tumor found by pathology) may be useful.

One approach for assessing the relationship between gene-expression values in the tumor and the existence and extent of necrotic tissue is to use a predictive model to identify genes that are associated with necrotic tissue, while avoiding over-fitting due to the large number of genes. To this end, first build a predictive model for `necrotic_cells.pct` as a function of gene expression values. Next, consider a categorical version of the same outcome, where `necrotic_cells.pct` either 0 or greater than 0. Compare the results of the two predictive models and discuss pros/cons of the two modeling approaches.

Methods

Measure of association To capture potential non-linear associations between necrosis and gene expression, we could introduce polynomial terms. However, adding just a quadratic term for each predictor would double our number of predictors to 1.093×10^5, which exceeds our computational limits. Therefore, we limit our analysis to linear associations with the original set of 5.468×10^4.

Regression For numeric predictors, multiple linear regression is well-suited to estimate the expected percentage of necrotic cells in a tumor, while multiple logistic regression may appropriately model the odds of necrotic cells being present.

Lasso regularization Given the large number of predictors, we will surely overfit our data. We will conduct lasso regularization to penalize models with many covariates, while still minimizing model error. For the penalization hyper-parameter, we will evaluate 100 candidates.

Cross-validation With so many predictors, we will surely find some meaningful predictor by chance. We will follow 5-fold cross-validation to create multiple models and identify predictors that are repeatedly observed as meaningful.

Results

We fit linear regression to model the expected percentage of necrotic cells in a tumor, given numeric measures of 5.468×10^4 gene expressions. To avoid overfitting, we used lasso regularization, testing 100 penalization parameters, and applied 5-fold cross-validation to prevent selection bias. The model identified zero gene expressions as having a meaningful linear association with necrosis. Figure 1 shows the search for a penalization term that yields the best performance, as measured by mean-squared error (MSE); table 1 shows this model's estimated coefficients.

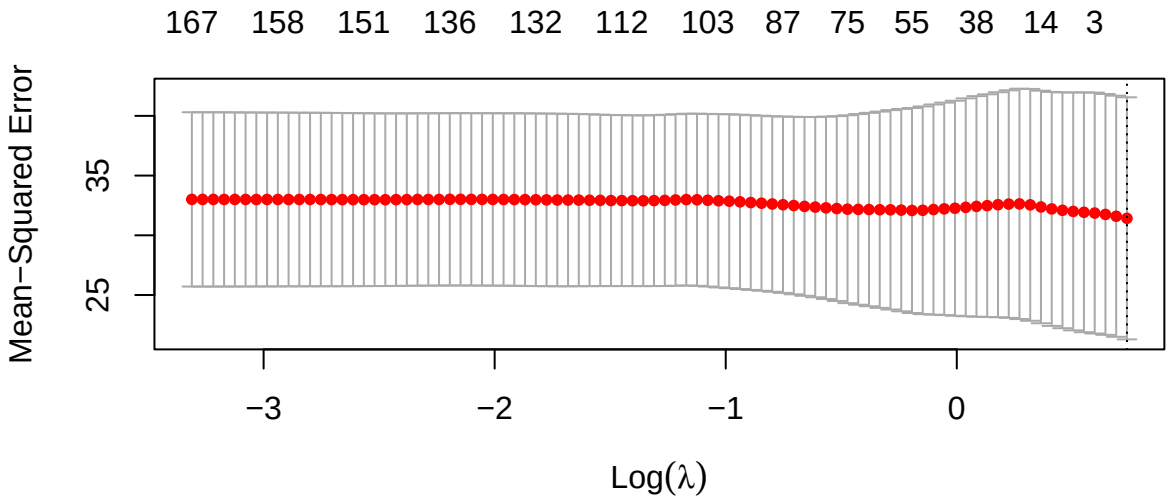

Figure 1: 10-fold Lasso regularization for a continuous model

Table 1: Non-zero regression coefficient(s) from cross-validated, regularized linear model.

	Estimate
(Intercept)	3.03

We fit logistic regression to model the log odds of a tumor having necrotic cells, given numeric measures of 5.468×10^4 gene expressions. To avoid overfitting, we used lasso regularization, testing 100 penalization parameters, and applied 5-fold cross-validation to prevent selection bias. The model identified two gene expressions: `CALM1 & CALM2 & CALM3.7`, with a weak negative, linear association; and `SNX29P2.1`, with a weak positive, linear association with necrosis. Confidence intervals are not interpreted due to the bias introduced by regularization and cross-validation, in exchange for reduced variance. Figure 2 shows the search for a penalization term that yields the lowest MSE; table 2 shows this model's estimated coefficients.

As a note, the model without predictors (estimating necrosis based on observed proportions) had a similar MSE to our 2-predictor model.

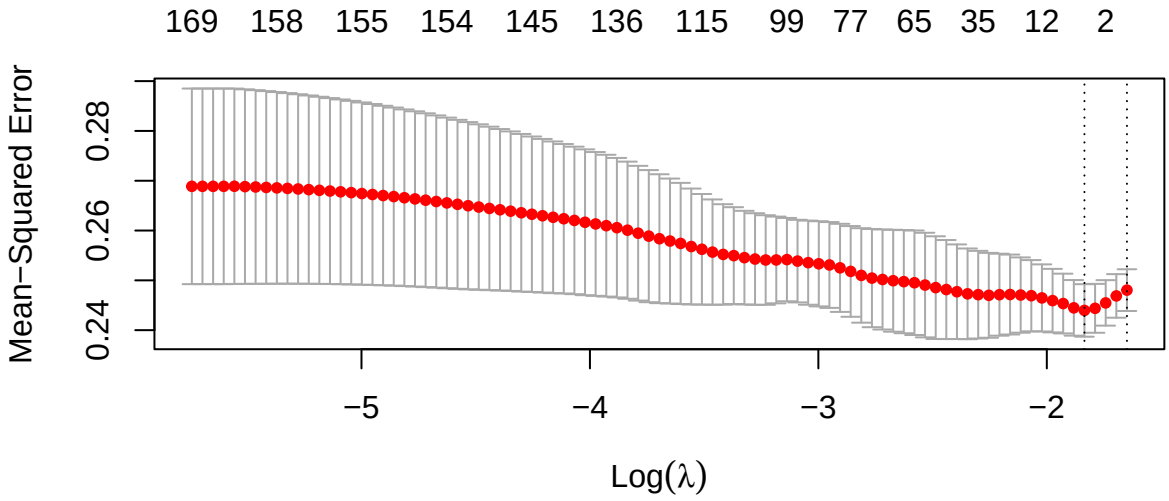

Figure 2: 10-fold Lasso regularization for a categorical model

Table 2: Non-zero regression coefficient(s) from cross-validated, regularized logistic model.

	Estimate	exp(Estimate)
(Intercept)	0.490	1.632
CALM1 & CALM2 & CALM3.7	-0.034	0.966
SNX29P2.1	0.040	1.041

Model Comparison

The continuous model achieved its lowest MSE by discarding all predictors at moderate regularization, while the categorical model required low regularization for optimal MSE. Increasing penalization made the categorical model's MSE estimate more precise, but lessened the precision for the continuous model.

The continuous model has the following advantages:

(a) Higher resolution: Preserving the full range of variability in necrotic tissue can provide a more nuanced understanding of how gene expression is associated with the extent of tumor necrosis. Given more information than a binary outcome, this model has more power than its counterpart.

(b) No arbitrary threshold: A cutoff at zero may not be the best choice to determine considerable or clinically significant extent of necrosis.

The categorical model has the following advantages:

(a) Lower overfitting risk: Distinguishing between presence and absence may make this model less susceptible to overfitting minute changes in percentage of necrosis.

(b) Easier interpretation: It is easier to identify genes that significantly impact the likelihood of necrosis than genes that influence its extent.

End of report. Code appendix begins on the next page.

Code Appendix

```
# clear environment
rm(list = ls())

# setup options
knitr::opts_chunk$set(echo = FALSE, message = FALSE)
options(knitr.kable.NA = '-', digits = 3)
labs <- setdiff(knitr::all_labels(), c("setup", "llm_appendix", "allcode"))
## load relevant packages
library(data.table)   # extension of 'data.frame'
library(dplyr)        # data manipulation
library(glmnet)       # lasso regularization
library(knitr)        # pretty tables

## load data
# clinical/phenotypic information
noah_clinical <- data.table::fread("../data/clinical_data.csv")[,-1]
# expression information (by probe set)
noah_expression <- fread("../data/expression_data_probeID.csv")[,-1] %>%
  data.frame
# gene name identifiers corresponding to each probe set
noah_annotation <- fread("../data/annotation.csv")

## handle missing gene names
# copy probe set ID to empty gene names
noah_annotation$gene.names <- with(noah_annotation,
                                   ifelse(gene.names == "",
                                          probset.ids, gene.names))
# check for any missing gene names
sum(noah_annotation$gene.names == "")

# inspect dimensions of each data frame
dim(noah_clinical)    # 152 tumors; 2 identifiers, 27 features
dim(noah_expression)  # 154 tumors; 2 identifiers, 54675 genes
dim(noah_annotation)  # 54675 genes; 1 identifier, 1 feature
# drop 26 clinical features unrelated to our analysis
noah_clinical <- noah_clinical %>% select(centerid, patid, necrotic_cells.pct)

## merge data
noah <- dplyr::inner_join(noah_clinical, noah_expression,
                          by = c("centerid", "patid")) %>% data.frame()
dim(noah)  # 152 tumors; 2 identifiers, 1 outcome, 54675 predictors
# drop identifiers
noah <- noah %>% select(-centerid, -patid)

## rename columns (from probe set IDs to gene names)
# join current column names to their replacement
new_colnames <- inner_join(data.frame(probset.ids = names(noah)),
                           noah_annotation,
                           by=c("probset.ids"))
# replace column names for the 54675
new_colnames$gene.names <- gsub(" /// ", " & ", new_colnames$gene.names)
```

```r
names(noah)[-1] <- new_colnames$gene.names
# replace column name for the 1 outcome
names(noah)[1] <- "necrotic"

npredictor <- ncol(noah) - 1  # (this will be useful later)

## inspect clean data
head(noah)

## OPTIONAL: create new polynomial predictors
# maxdegree <- 2
# if (maxdegree > 1) {
#   noah_poly <- noah[1]
#   # create new polynomial predictors
#   for (degree in 2:maxdegree) {
#     poly <- mutate(noah[-1],
#                    across(everything(), ~.^degree,
#                           .names = paste("{.col}", degree, sep = "_")))
#     noah_poly <- cbind(noah_poly, poly)
#     }
#   }

## set hyper-parameters for LASSO and CV
nlambda <- 100
nfold <- 5

## model percentage of necrotic tissue in a tumor from gene expression values
set.seed(1)
lasso_cv_fit <- glmnet::cv.glmnet(y = noah[,1],
                                  x = as.matrix(noah[,-1]),
                                  alpha = 1,  # lasso regularization
                                  nfolds = nfold, nlambda = nlambda)

## evaluate model
# plot MSE across candidate lasso values
plot(lasso_cv_fit)
# get nonzero coefficient estimate(s)
coefs <- coef(lasso_cv_fit, s = lasso_cv_fit$lambda.min)
nonzero_coefs <- data.frame(Estimate = coefs[as.list(coefs) != 0,])
if(nrow(nonzero_coefs) == 1) { row.names(nonzero_coefs) <- "(Intercept)"}
nonzero_coefs %>% knitr::kable(caption = "Non-zero regression coefficient(s)
                               from cross-validated, regularized linear model.")

## model presence of necrotic tissue in a tumor from gene expression values
set.seed(1)
lasso_cv_fit2 <- cv.glmnet(y = ifelse(noah$necrotic > 0, 1, 0),
                           x = as.matrix(noah[,-1]),
                           alpha = 1,  # lasso regularization
                           nfolds = nfold, nlambda = nlambda)

## evaluate model
# plot MSE across candidate lasso values
plot(lasso_cv_fit2)
```

```r
# get nonzero coefficient estimates
coefs2 <- coef(lasso_cv_fit2, s=lasso_cv_fit2$lambda.min)
nonzero_coefs2 <- data.frame(Estimate = coefs2[as.list(coefs2) != 0,]) %>%
  mutate('exp(Estimate)' = exp(Estimate))
if(nrow(nonzero_coefs2) == 1) { row.names(nonzero_coefs2) <- "(Intercept)"}
nonzero_coefs2 %>% kable(caption = "Non-zero regression coefficient(s)
                         from cross-validated, regularized logistic model.")

## compare regression models of necrotic tissue as a continuous vs binary variable
lasso_cv_fit$index
lasso_cv_fit2$index

summary(coefs)
summary(coefs2)

noah %>%
  select(necrotic) %>%
  mutate(binaryn = ifelse(necrotic > 0, 1, 0)) %>%
  colMeans()
```

End of document.