STATS 315A: Statistical Learning

Erich Trieschman

2022Q1 class notes

1 Supervised learning overview

1.1 Least squares

1.2 Nearest neighbors

1.3 Classes of restricted estimators

1.4 Bias-Variance tradeoff

1.5 Cross validation

Cross validation is used to select the tuning parameters of a particular model, not the variables themselves. For example with subset selection, we use cross validaiton to select s, the subset size, **not** the actual predictors to use in the model
It would be ideal to have both a test set and a cross validation set. Running the CV and tuning parameters can bias the results. A separate test set provides convincing independent assessment

1.5.1 K-fold cross validation

- For each k, fit the model with parameter λ to the other K-1 parts, getting $\hat{\beta}^{-k}(\lambda)$
- Compute error, $RSS_{-k} = \sum_{i \in k}(y_i - x_i\hat{\beta}^{-k}(\lambda))^2$
- Cross validation error, $CV(\lambda) = \frac{1}{K}\sum_{k=1}^{K} RSS_{-k}(\lambda)$

1.6 Bootstrap

Sample N times with replacement from teh training set to form a bootstrap data set Estimate model on bootstrap data, with predictions made from the original training data Repeat process many times and average results Poor estimate of prediction error (why?) Good estimate for standard errors of predictions and confidence intervals for parameters

2 Linear methods for regression

Functions in the real world are rarely linear, but linear approximations are a good heuristic for the biance-variance tradeoff.

2.1 Linear regression and least squares

Assuming X full rank. Geometrically, the point, $\hat{\beta}$ which solves $argmin_x \left\| X\hat{\beta} - y \right\|_2$ is one where $X\hat{\beta} - y$ is orthogonal to the range of X. To solve for this:

$$\text{Want: } (X\hat{\beta} - y) \perp \{z | z = X\hat{\beta}\} \longleftrightarrow (X\hat{\beta} - y) \perp range(A) \longleftrightarrow (X\hat{\beta} - y) \perp x_i, \forall i \in X$$
$$x_i^T(X\hat{\beta} - y) = 0,\ \forall i \in X \longleftrightarrow X^T(X\hat{\beta} - y) = 0 \longleftrightarrow \hat{\beta} = (X^TX)^{-1}X^TY$$

Properties:

- Regression coefficient $\hat{\beta}_i$ estimates the expected change in y per unit change in x_i *holding all other predictors fixed*
- For X_1, X_2, mutually orthogonal matrices or vectors, the joint regression coefficients for $X = (X_1, X_2)$ on y, can be found from separate regressions. (Proof: $X_1^T(y - X\hat{\beta}) = X_1^T(y - X_1\hat{\beta}_1) = 0$)
- The multiple regression coefficient of x_p, the last column of X, is the same as the univariate coefficient in the regression of $y \sim z_p$. Here, $z_p = x_p - X_p^T\alpha$ (the part of x_p orthogonal to X_p, all but column x_p of X). Variance also comes form the univariate regression.

  - $\hat{\beta}_p = (z_p^T z_p)^{-1} z_p^T y = z_p^T y / z_p^T z_p$
  - $Var(\hat{\beta}_p) = \sigma^2 / z_p^T z_p$

Assumptions:

- Errors, $\epsilon_i \sim N(0, \sigma^2)$ assumed to be *independent* of the x_i's
- X considered fixed, not random.
- X is full rank. When not (because multiple variables are perfectly correlated), $X^T X$ is singular and the coefficients, $\hat{\beta}$, are not uniquely defined. In these cases, features can be reduced by filtering or with a regularization.
- Conditional expectation of y is linear in X, $y = E(y \mid X) + \epsilon$. With this assumption, we can show $\hat{\beta} \sim N(\beta, (X^T X)^{-1} \sigma^2)$

2.1.1 Standard error and confidence intervals

We often assume $y_i = \hat{\beta} x_i + \epsilon_i$ with $E(\epsilon_i) = 0$ and $Var(\epsilon_i) = \sigma^2$. Then

$$se(\hat{\beta}) = \left[\frac{\sigma^2}{\sum(x_i - \bar{x})^2}\right]^{\frac{1}{2}}, \text{ approximating with } \hat{se}(\hat{\beta}) = \left[\frac{\hat{\sigma}^2}{\sum(x_i - \bar{x})^2}\right]^{\frac{1}{2}} \text{ where } \hat{\sigma}^2 = \frac{\sum(y_-\hat{y_i})^2}{N-2}$$

2.1.2 Expectation of $\hat{\beta}$

$Var(\hat{\beta}) = (X^T X)^{-1} \sigma^2$ $\hat{\sigma}^2 = \frac{1}{N-p-1} \sum (y_i - \hat{y}_i)^2$. The $N-p-1$ denominator makes $\hat{\sigma}$ unbiased ($E(\hat{sigma}^2) = \sigma^2$)

2.2 Subset selection

Subset methods help us tradeoff an increase in bias with lower variance. Here we retain a subset of predictor variables for the final regression. **Approaches:**

- All subsets regression: finds the best subset of size $s \in \{1, \ldots, p\}$ that minimizes the residual sum of squares. Limited use in high dimensions (¿30) because of the computational complexity.
- Forward stepwise selection: beginning with a model of the intercept only, sequentially add to the model the predictor that most reduces the residual sum of squares
- Backward stepwise selection: beginning with the full OLS model, sequentially remove from the model the predictor to most reduce residual sum of squares

Note: The tuning parameter, s of each subset selection approach should be determined through cross validation

2.3 Shrinkage methods

- Shrinkage methods often help us tradeoff an increase in bias with lower variance
- It is important to standardize (mean=0, variance=1) the predictors before running shrinkage methods to make the pentalty meaningful; centering also eliminates the need for an intercept

2.3.1 Ridge regression

Ridge regression is a linear regression with a square penalty on the size of the model parameters:

$$\begin{aligned}
\hat{\beta}^{ridge} &= argmin(y - X\beta)^T (y - X\beta) + \lambda \beta^T \beta \\
\hat{\beta}^{ridge} &= (X^T X + \lambda I)^{-1} X^T y
\end{aligned}$$

This is a biased estimator for y that may reduce MSE. Note when $\lambda = 0$, this is the same as OLS

Ridge regression shrinks the coefficients of the principal components (Xv_j), with relatively more shrinkage on the smaller components. Proof:

$$\begin{aligned}
X\hat{\beta} &= X(X^T X + \lambda I)^{-1} X^T y \\
X\hat{\beta} &= UDV^T (VD^2 V^T + \lambda I)^{-1} VDU^T y \\
&= UD(D^2 + \lambda I)^{-1} DU^T y \\
&= \sum_{j=1}^{p} u_j \frac{d_j^2}{d_j^2 + \lambda} u_j^T y
\end{aligned}$$

2.3.2 The Lasso

The lasso is a shrinkage method for linear regressions like ridge, but uses the 1-norm as a penalty instead of the 2-norm.

$$\hat{\beta}^{lasso} = argmin(y - X\beta)^T(y - X\beta) + \lambda \left\|\beta\right\|_1$$

There is no analytical solution to this objective, but the lasso is a convex problem when stated below (meaning it can be minimized)

$$\begin{aligned} \text{min. } & argmin(y - X\beta)^T(y - X\beta) \\ \text{subject to } & \lambda \left\|\beta\right\|_1 \leq t \end{aligned}$$

We find with the lasso that the parameter vector often inclues zeros for specific parameters. Inuitively this makes sense since the pointed 1-norm ball is likely to be maximized at one of its corners.

Elastic net combines the ridge and lasso penalties through tuning parameter α. It can be effective for sparse models with correlated predictors.

$$\hat{\beta}^{enet} = argmin(y - X\beta)^T(y - X\beta) + (1-\alpha)\left\|\beta\right\|_2 + \alpha\left\|\beta\right\|_1$$

2.4 Methods using derived input directions

Here we choose a set of linear combinations of $x_i \in X$, and run a regression on these combinations

2.4.1 Principal component regression

Linear combinations are selected to maximize variance. These maximal-variance combinations are called the **principal components**. For standardized X, the principal components, z_i, are

$$z_1 = Xv \text{ such that } v \text{ maximizes } Var(Xv) = \frac{1}{N}v^T X^T X v \text{ subject to } \left\|v\right\|_2 = 1$$

$$z_i = Xv_i \text{ where } v_i \text{ is the ith column of the SVD; singular values determine the ordering}$$

The principal component analysis is highly connected to the singular value decomposition of standardized X. Since the SVD can help us construct the eigendecomposition of $X^T X$

$$\frac{1}{N}X^T X = \frac{1}{N}VD^2V^T \iff \frac{1}{N}V^T X^T X V = D^2$$

Principal Component Analysis regression then generates a linear regression using a subset $s \leq p$ of the principal compoents. Since these principal components are orthogonal, the regression is a sum of univariate regressions

2.4.2 Partial least squares

Linear combinations are constructed using both y and X, both standardized.

- Compute univariate regression coefficients, $\hat{\gamma}_l$ of y on eacy x_l
- Construct $z_1 = \sum_l \hat{\gamma}_l x_l$
- Get $\hat{\beta}_1$ from $y \sim z_1$
- Orthogonalize $y, x_1, \ldots, x_p$ with respect to z_1
  - $y^* = y - \hat{\beta}_1 z_1$
  - $x_l^* = x_l - \frac{z_1^T x_l}{z_1^T z_1} z_1$
- Repeat until $s \leq p$ directions have been obtained (we get back OLS if $s = p$)

2.5 Degrees of freedom

Degrees of freedom for linear regressions is the number of free parameters that determines the model.

$$\begin{aligned} &\text{For } \hat{y} = Hy,\ df = tr(H) \\ &\text{Note } \hat{y} = X\hat{\beta} = X(X^T X)^{-1}X^T y \text{ so } H = X(X^T X)^{-1}X^T \text{ in OLS} \\ &\hat{y} = X\hat{\beta} = X(X^T X + \lambda I)^{-1}X^T y \text{ so } H = X(X^T X + \lambda I)^{-1}X^T \text{ in Ridge regression} \end{aligned}$$

The lasso is not a linear regression. Its degrees of freedom are defined as

$$df = \sum_i cov(y_i, \hat{y}_i)/\sigma^2$$

3 Linear methods for classification

For classification, the input space can be divided into regions of constant classification, with decision boundaries

$$\{x \mid \hat{\beta}_k x = \hat{\beta}_l x\}$$

In general, linear methods for classification model *discriminant functions*, $\delta_k(x)$, or *posterior probabilities*, $P(G = k \mid X = x)$, for each class, classifying x to the class with the largest discriminant or probability.
For this theory, we require that these functions have some monotone transformation to a linear function; it turns out that both linear discriminant analysis and linear logistic regression result in linear log-odds (logits).

3.1 Linear regression of indicator matrix

Categorical $y \in \mathbb{R}^{n\times 1}$ is one-hot-encoded into a series of boolean vectors in $Y \in \mathbb{R}^{n\times k}$, and model parameters, $\hat{\beta} \in \mathbb{R}^{p\times k}$ are estimated in the same way

$$\begin{aligned}
\hat{\beta} &= (X^T X)^{-1} X^T Y \\
\hat{Y} &= X\hat{\beta} = X(X^T X)^{-1} X^T Y \\
\hat{G} &= argmax_{k\in\mathcal{G}} x^T \hat{\beta} \text{ for new observation } x
\end{aligned}$$

Notes

- Rigid nature of regression model, classes can be masked by other classes for $K > 2$
- A loose, but general rule is that if $K \geq 3$ classes are lined up in one direction, polynomial terms of degree $K - 1$ might be needed to resolve the classes.

3.2 Linear discriminant analysis (LDA)

Category of backward selection models, meaning we use y to generate boundaries of X. Suppose $f(x \mid k)$ is the density of X conditional on class $G = k$, and π_k is the prior probability for class $G = k$. Bayes rule says

$$P(G = k \mid X = x) = \frac{f(x \mid k)\pi_k}{\sum_{l=1}^{K} f(x \mid l)\pi_l}$$

Linear discriminant analysis and its relatives are based on the assumption that $f(x \mid k)$ is multivariate Gaussian

$$f(x \mid k) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} \exp(-\frac{1}{2}(x - \mu_k)^T \Sigma_k^{-1}(x - \mu_k))$$

3.2.1 Linear discriminant analysis

LDA specifically arises out of the special case when we assume all classes have a common variance, i.e., $\Sigma_k = \Sigma \forall k$. This results in sufficient cancelations when comparing classes, leading to an equation linear in x

$$\begin{aligned}
\log \frac{P(G = k \mid X = x)}{P(G = k \mid X = x)} &= \log \frac{f(x \mid k)}{f(x \mid l)} + \log \frac{\pi_k}{\pi_l} \\
&= \log \frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k + \mu_l)^T \Sigma^{-1}(\mu_k - \mu_l) + x^T \Sigma^{-1}(\mu_k - \mu_l)
\end{aligned}$$

Notes

- This implies the decision boundary between classes is linear in x
- These decision boundaries are *not* perpendicular bisectors to the line segments joining centroids (this would be the case if $\Sigma = \sigma^2 \mathbf{I}$) and the priors, π_i were equal)
- In practice we don't know the parameters of the Gaussian distributions and we estimate with
  - $\hat{\pi}_k = n_k/N$
  - $\hat{\mu}_k = \sum_{g_i=k} x_i/n_k$
  - $\hat{\Sigma}_k = \sum_{k=1}^{K} \sum_{g_i=k}(x_i - \hat{\mu}_k)^T(x_i - \hat{\mu}_k)/(N - K)$
- In the case of a binary y, LDA and linear regression have a direct correspondence

3.2.2 Quadratic discriminant analysis (QDA)

If we tighten our assumption about Σ_k so that the covariance matrices are not even across groups, we get a **Quadratic discriminant analysis** that is quadratic in x:

$$\delta_k(x) = \frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + log\pi_k$$

Notes

- Decision boundaries between classes are described by a quadratic equation: $\{x \mid \delta_k(x) = \delta_l(x)\}$
- We can produce quadratic boundaries with LDA as well by enlarging the parameter space to include quadratics ($(x_1, x_2) \implies (x_1, x_2, x_1x_2, x_1^2, x_2^2)$). In general, QDA is the preferred approach

3.2.3 Regularized discriminant analysis (RDA)

Suposing we want to compromise between LDA and QDA, **regularized discriminant analysis** allows us to tune in between our assumptions of common across-class covariance (LDA) and unique within-class covariance (QDA) with a tuning parameter, α. Regularized covariance matrices have the form

$$\hat{\Sigma}_k(\alpha) = \alpha\hat{\Sigma}_k + (1-\alpha)\hat{\Sigma}$$

We can also consider a regularization approach to tune a caommon across-class covariance to an assumption about independence with tuning parameter, γ

$$\hat{\Sigma}(\gamma) = \gamma\hat{\Sigma} + (1-\gamma)\hat{\sigma}^2\mathbf{I}$$

3.2.4 Additional LDA notes

We can reduce the computational complexity of LDA by taking advantage of the eigendecomposition of the covariance matrix: $\hat{\Sigma} = UDU^T$, noticing

$$(x-\hat{\mu}_k)^T\Sigma^{-1}(x-\hat{\mu}_k) = [U^T(x-\hat{\mu}_k)]^TD^{-1}[U^T(x-\hat{\mu}_k)]$$
$$\log|\hat{\Sigma}| = \sum_l \log d_{ll}$$

With these relations, X can be classified through two steps

- Sphere data: $X^* = D^{-1/2}U^TX$ where $\hat{\Sigma} = UDU^T$ (note now that the common covariance matrix is $\mathbf{I}$)
- Classify the closest class centroid in the transformed space, modulo the effect of class probabilities, π_k

3.3 Logistic Regression

4 Basis expansions and regularizations

4.1 Piecewise polynomials and regression splines

4.2 Smoothing splines

4.3 Multidimentional splines

4.4 Regularization and reproducing kernel Hilbert spaces