STAT 215A Fall 2020 Week 11

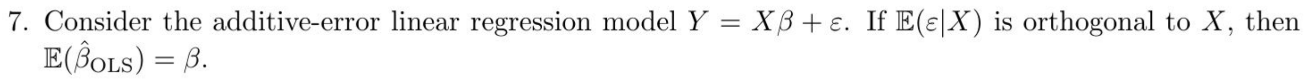
James Duncan, OH: M, Th 2-4pm

Thanks to Tiffany Tang for sharing her slides

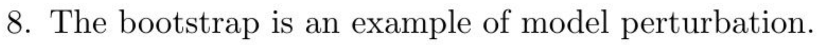
Announcements

- Lab 4 due in less than two weeks on November 19 at 11:59pm
    - Everyone in the group submits the **same** lab4 report / files but each person needs to push the files to their individual private repos
- Good job on the midterm: median 33/38

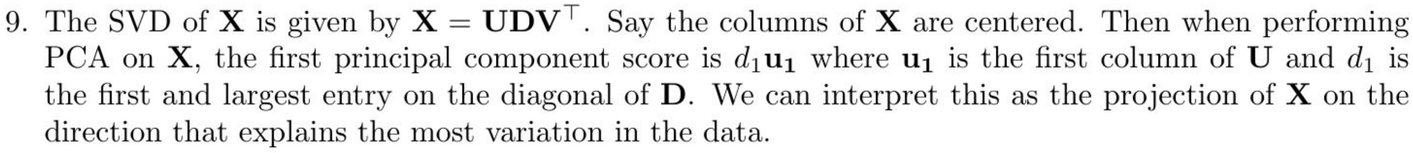
Midterm T/F



7. Consider the additive-error linear regression model $Y = X\beta + \varepsilon$. If $\mathbb{E}(\varepsilon|X)$ is orthogonal to X, then $\mathbb{E}(\hat{\beta}_{\text{OLS}}) = \beta$.

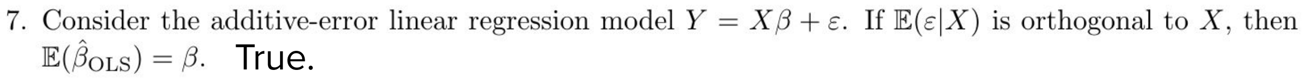


8. The bootstrap is an example of model perturbation.

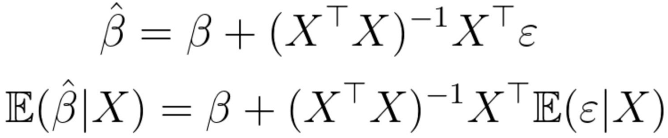


9. The SVD of $\mathbf{X}$ is given by $\mathbf{X} = \mathbf{UDV}^\top$. Say the columns of $\mathbf{X}$ are centered. Then when performing PCA on $\mathbf{X}$, the first principal component score is $d_1\mathbf{u_1}$ where $\mathbf{u_1}$ is the first column of $\mathbf{U}$ and d_1 is the first and largest entry on the diagonal of $\mathbf{D}$. We can interpret this as the projection of $\mathbf{X}$ on the direction that explains the most variation in the data.

Midterm T/F

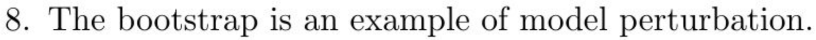


7. Consider the additive-error linear regression model $Y = X\beta + \varepsilon$. If $\mathbb{E}(\varepsilon|X)$ is orthogonal to X, then $\mathbb{E}(\hat{\beta}_{\text{OLS}}) = \beta$. True.

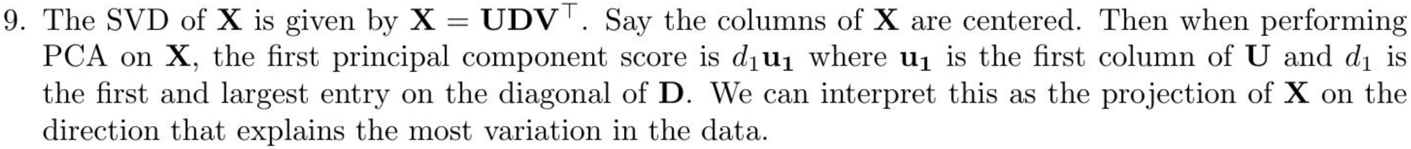


$$\hat{\beta} = \beta + (X^\top X)^{-1} X^\top \varepsilon$$

$$\mathbb{E}(\hat{\beta}|X) = \beta + (X^\top X)^{-1} X^\top \mathbb{E}(\varepsilon|X)$$



8. The bootstrap is an example of model perturbation.



9. The SVD of $\mathbf{X}$ is given by $\mathbf{X} = \mathbf{UDV}^\top$. Say the columns of $\mathbf{X}$ are centered. Then when performing PCA on $\mathbf{X}$, the first principal component score is $d_1\mathbf{u_1}$ where $\mathbf{u_1}$ is the first column of $\mathbf{U}$ and d_1 is the first and largest entry on the diagonal of $\mathbf{D}$. We can interpret this as the projection of $\mathbf{X}$ on the direction that explains the most variation in the data.

Midterm T/F

7. Consider the additive-error linear regression model $Y = X\beta + \varepsilon$. If $\mathbb{E}(\varepsilon|X)$ is orthogonal to X, then $\mathbb{E}(\hat{\beta}_{\text{OLS}}) = \beta$. True.

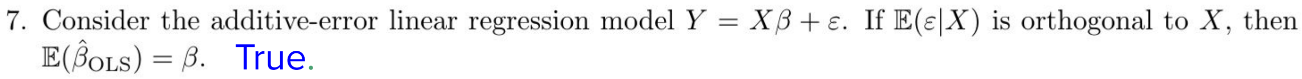
$$\hat{\beta} = \beta + (X^\top X)^{-1} X^\top \varepsilon$$

$$\mathbb{E}(\hat{\beta}|X) = \beta + (X^\top X)^{-1} X^\top \mathbb{E}(\varepsilon|X)$$

8. The bootstrap is an example of $\underbrace{\text{model}}_{\text{“data”} \Rightarrow \text{True}}$ perturbation. False

9. The SVD of $\mathbf{X}$ is given by $\mathbf{X} = \mathbf{UDV}^\top$. Say the columns of $\mathbf{X}$ are centered. Then when performing PCA on $\mathbf{X}$, the first principal component score is $d_1\mathbf{u_1}$ where $\mathbf{u_1}$ is the first column of $\mathbf{U}$ and d_1 is the first and largest entry on the diagonal of $\mathbf{D}$. We can interpret this as the projection of $\mathbf{X}$ on the direction that explains the most variation in the data.

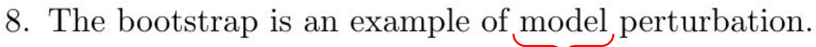
Midterm T/F



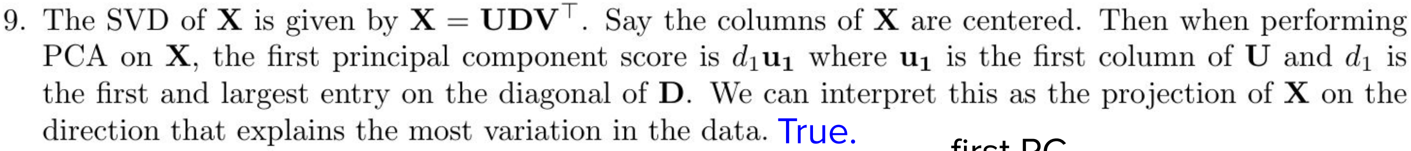
7. Consider the additive-error linear regression model $Y = X\beta + \varepsilon$. If $\mathbb{E}(\varepsilon|X)$ is orthogonal to X, then $\mathbb{E}(\hat{\beta}_{\text{OLS}}) = \beta$. True.

$$\hat{\beta} = \beta + (X^\top X)^{-1}X^\top\varepsilon$$

$$\mathbb{E}(\hat{\beta}|X) = \beta + (X^\top X)^{-1}X^\top\mathbb{E}(\varepsilon|X)$$



8. The bootstrap is an example of $\underbrace{\text{model}}_{\text{“data”} \Rightarrow \text{True}}$ perturbation. False.



9. The SVD of $\mathbf{X}$ is given by $\mathbf{X} = \mathbf{UDV}^\top$. Say the columns of $\mathbf{X}$ are centered. Then when performing PCA on $\mathbf{X}$, the first principal component score is $d_1\mathbf{u_1}$ where $\mathbf{u_1}$ is the first column of $\mathbf{U}$ and d_1 is the first and largest entry on the diagonal of $\mathbf{D}$. We can interpret this as the projection of $\mathbf{X}$ on the direction that explains the most variation in the data. True.

$$\mathbf{XV} = \mathbf{UD} \implies \underbrace{\mathbf{X}\overset{\text{first PC}}{\mathbf{v}_1}}_{\substack{\text{projection of data onto} \\ \text{first PC direction}}} = d_1\mathbf{u}_1$$

Outline for today

- Classification algorithms
    - Logistic regression
    - Naive Bayes
    - Discriminant analysis
    - KNN classifier

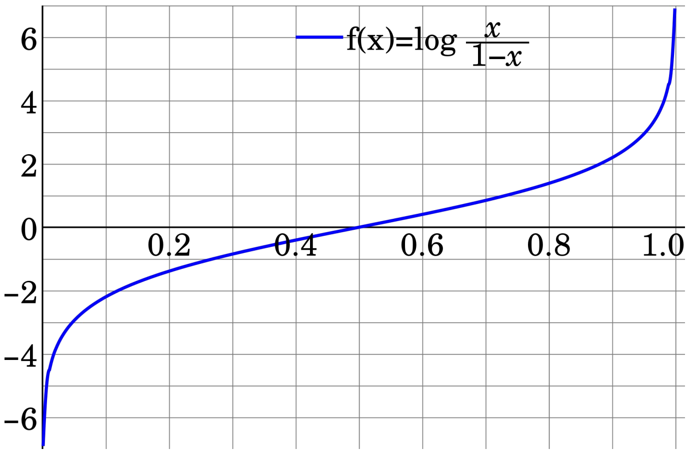
Why classification and not regression?

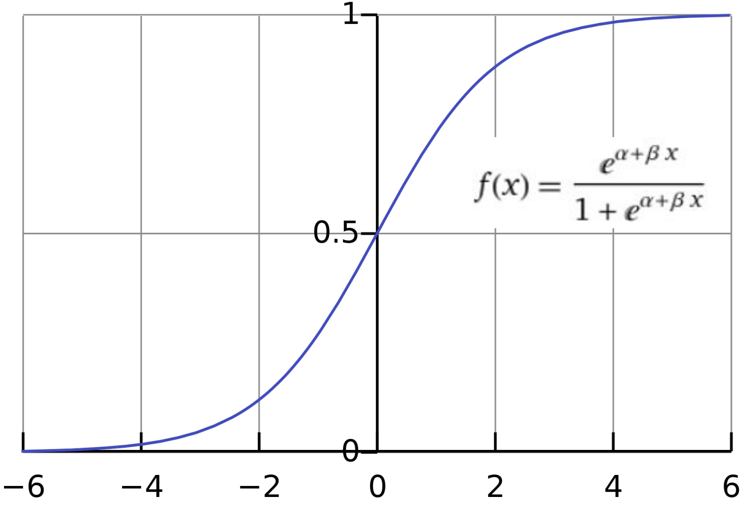
- Suppose we have data X_1, ..., X_n and categorical responses $y_1, \cdots, y_n$, i.e. $y_i \in 1, \ldots, K$.
- Problems with regression:
  - Hard to assign numeric values to categories
  - Usually no ordering of the categories
  - Even if categories are ordered, not necessarily equally spaced

Logistic regression

Assume there are two classes and $y_i|x_i \sim \text{Bernoulli}(\pi_i)$ are independent with

$$\log\left(\frac{\pi_i}{1-\pi_i}\right) = \alpha + \beta x_i \iff \pi_i = \frac{\exp\{\alpha + \beta x_i\}}{1 + \exp\{\alpha + \beta x_i\}}$$





Find MLE via Newton-Raphson / IRLS. **`glmnet`** can fit large logistic regression models efficiently.

Logistic Regression Extensions

- What if more than 2 classes?
  - Multinomial logistic regression
- What if $p > n$ (or p large)?
  - Regularized logistic regression: $\max_{\alpha,\beta} \ell(\alpha, \beta, X) - \lambda q(\beta)$

    Penalty, e.g. L^1, L^2
- What assumptions are you making?
  - Linear relationship between covariates and log-odds.
  - Correlated predictors can inflate variance and bias of coefficients

Modeling via class conditional densities

If we know the class posterior distribution $P(Y = k|\overset{\in \mathbb{R}^p}{X})$, then we could just predict the class k with the highest probability given the observation.

- Say $f_k(x)$ is the conditional density of an observation within the class k
- Call π_k the prior probability of the class k and assume $\sum_{k=1}^{K} \pi_k = 1$

Modeling via class conditional densities

If we know the class posterior distribution $P(Y = k|\overset{\in \mathbb{R}^p}{X})$, then we could just predict the class k with the highest probability given the observation.

- Say $f_k(x)$ is the conditional density of an observation within the class k
- Call π_k the prior probability of the class k and assume $\sum_{k=1}^{K} \pi_k = 1$

Then, using Bayes rule we have $P(Y = k|X) = \dfrac{f_k(x)\pi_k}{\sum_l f_l(x)\pi_l}$

Naive Bayes

Assumes that given the class label, the features are independent!

$$f_k(x) = \prod_{j=1}^{p} f_{jk}(x_j)$$

- E.g., model the covariates via independent Gaussians: $X|Y = k \sim N(\mu_k, \sigma^2 \mathrm{I})$
- This makes estimation much simpler, and can actually work well in practice in spite of this strong assumption.

Linear discriminant analysis (LDA)

LDA is based upon modeling the class conditional density $f_k(x)$ via a Gaussian with **equal variance** within each class (but not necessarily independent).

$$X|Y = k \sim N(\mu_k, \Sigma_w)$$

within class covariance matrix, common across classes

Linear discriminant analysis (LDA)

LDA is based upon modeling the class conditional density $f_k(x)$ via a Gaussian with **equal variance** within each class (but not necessarily independent).

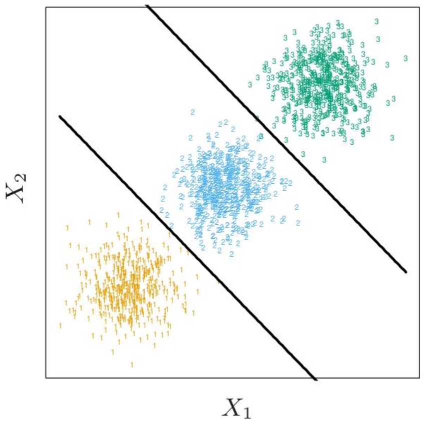
$$X|Y = k \sim N(\mu_k, \Sigma_w)$$

within class covariance matrix, common across classes

- **Exercise**: show that, for this model, we have

$$\log \frac{P(Y=k|X)}{P(Y=l|X)} = \log \frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k - \mu_l)^\top \Sigma^{-1}(\mu_k + \mu_l) + x^\top \Sigma^{-1}(\mu_k - \mu_l)$$

linear in x!



Linear discriminant analysis (LDA)

LDA is based upon modeling the class conditional density $f_k(x)$ via a Gaussian with **equal variance** within each class (but not necessarily independent).

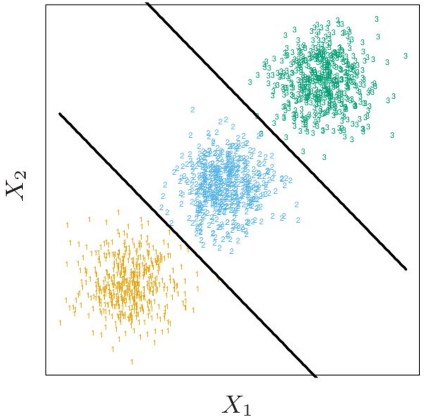
$$X|Y = k \sim N(\mu_k, \Sigma_w)$$

within class covariance matrix, common across classes

- **Exercise**: show that, for this model, we have

$$\log \frac{P(Y=k|X)}{P(Y=k|X)} = \log \frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k - \mu_l)^\top \Sigma^{-1}(\mu_k + \mu_l) + x^\top \Sigma^{-1}(\mu_k - \mu_l)$$

linear in x!



- We can fit the parameters via MLE:

$$\hat{\pi}_k = \frac{1}{n}\sum_{i=1}^{n} 1\{Y_i = k\} \qquad \hat{\mu}_k = \frac{1}{n_k}\sum_{i=1}^{n} 1\{Y_i = k\}X_i \qquad \hat{\mathbf{\Sigma}}_w = \frac{1}{n-K}\sum_{k=1}^{K}\sum_{i:Y_i=k}(X_i - \hat{\mu}_k)(X_i - \hat{\mu}_k)^\top$$

LDA as decomposition of variance

Can think about LDA as a decomposition of variance:

$$\underset{\text{Total variation}}{\hat{\mathbf{\Sigma}}_t} = \underset{\text{Between-class variation}}{\hat{\mathbf{\Sigma}}_b} + \underset{\text{Within-class variation}}{\hat{\mathbf{\Sigma}}_w}$$

- LDA finds a a linear projection of the data that maximizes the between-class variation while controlling for the within class variation

$$\max_{v_k} \; v_k^\top \hat{\mathbf{\Sigma}}_b v_k \qquad \text{subject to } v_k^\top \hat{\mathbf{\Sigma}}_w v_k = 1, \quad v_k^\top \hat{\mathbf{\Sigma}}_w v_j = 0 \;\; (\forall\, j < k)$$

- Collect into a matrix $V = [v_1, \ldots, v_K]$ and look at discriminant components XV
    - Low-dim projection of data that best separates the classes!

LDA for binary classification

In the binary case we only have one linear equation to work with:

$$\log \frac{P(Y=1|X)}{P(Y=0|X)} = \log \frac{\pi_1}{\pi_0} - \frac{1}{2}(\mu_1 - \mu_0)^\top \Sigma^{-1}(\mu_1 + \mu_0) + x^\top \Sigma^{-1}(\mu_1 - \mu_0)$$

- In this case, it can be shown that the estimated covariate vector $\hat{\Sigma}^{-1}(\hat{\mu}_1 - \hat{\mu}_0)$ is **parallel to the OLS (regression) solution** (the intercept may be different).
    - So was the Gaussian assumption really necessary?

LDA for binary classification

In the binary case we only have one linear equation to work with:

$$\log \frac{P(Y=1|X)}{P(Y=0|X)} = \log \frac{\pi_1}{\pi_0} - \frac{1}{2}(\mu_1 - \mu_0)^\top \Sigma^{-1}(\mu_1 + \mu_0) + x^\top \Sigma^{-1}(\mu_1 - \mu_0)$$

- In this case, it can be shown that the estimated covariate vector $\hat{\Sigma}^{-1}(\hat{\mu}_1 - \hat{\mu}_0)$ is **parallel to the OLS (regression) solution** (the intercept may be different).
  - So was the Gaussian assumption really necessary?
- Our decision boundary lies on the hyperplane where $P(Y=1|X) = P(Y=0|X)$
  - This hyperplane *does* rely on the Gaussian assumption.
  - As an alternative, we could instead **choose a cut point to minimize training error**.

LDA vs. Logistic Regression (LR)

The two methods seem to be very similar, but get to their results by very different methods, with important implications.

- Assumptions:
    - **LR makes fewer assumptions** and is therefore more general.
    - The additional assumptions imposed by **LDA leads to lower variance** of estimates (especially when true data is Gaussian).
- Robustness
    - Assumptions make **LDA more sensitive to outliers**
    - **LR downweights outliers** far from the decision boundary, making it more robust
- In practice, results are very similar, but LR may be a safer bet

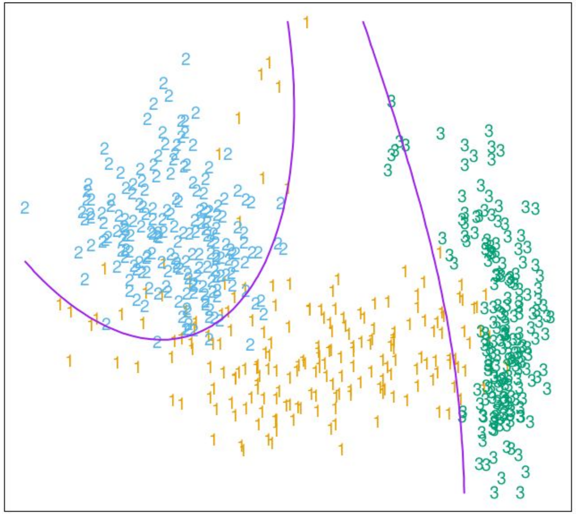
Quadratic discriminant analysis (QDA)

- When classes cannot be separated by a hyperplane, one option is to use LDA with quadratic features.

- Another is to relax the equal variance-covariance constraint, which results in QDA:

$$X|Y = k \sim N(\mu_k, \Sigma_k)$$

- Now we have to estimate separate covariance matrices for each class which can result in many more parameters.
- Another variant: Regularized Discriminant Analysis
  - Shrink the separate covariance matrices toward a common one



Summary so far

	Logistic	Naïve Bayes	LDA	QDA
Pros	• Can do inference (with all the caveats)	• Can choose any likelihood model	• Convenient visualizations • Linearly separable	• Quadratic decision boundaries
Cons	• Problems when p>n (a solution: regularized logistic regression) • Model misspecification?	• Assumes that features are independent (a very strong assumption) • Model misspecification?	• Problems when p>n (a solution: RDA) • Model misspecification? Non-normal or non-linear decision boundaries?	• Problems when p>n (a solution: RDA) • Requires larger n to estimate more parameters adequately (compared to LDA) • Model misspecification? Non-normal or non-linear decision boundaries?

K Nearest Neighbors

Dipping our toes into the realm of non-parametric classification.

For each test sample:

- Find the K “closest” neighbors
  - How to define closeness? Need a distance metric
- Take “majority vote” of neighbor classes as the class of the new observation

Advantages:

- Flexible
- Data-adaptive
- Simple, easy to implement

Main disadvantages:

- Curse of dimensionality

In R: `class::knn()`

Next time

- SVM
- Random forest
- Ensembles
- Evaluation