Regression and Classification

PSC 8185: Machine Learning for Social Science

Iris Malone

January 24, 2021

Materials adapted from Sergio Ballacado and Rochelle Terman

Announcements

Problem Set 1 Released

Recap

Where We’ve Been:

Recap

Where We've Been:

- 2 classes of ML: supervised and unsupervised

Recap

Where We've Been:

- 2 classes of ML: supervised and unsupervised
- 2 types of ML uses: inference and prediction

Recap

Where We've Been:

- 2 classes of ML: supervised and unsupervised
- 2 types of ML uses: inference and prediction
- Model selection depends on model performance and bias-variance trade-off

Recap

Where We've Been:

- 2 classes of ML: supervised and unsupervised
- 2 types of ML uses: inference and prediction
- Model selection depends on model performance and bias-variance trade-off
- Model performance aims to minimize loss function $(y_i - \hat{y}_i)$

Recap

Where We've Been:

- 2 classes of ML: supervised and unsupervised
- 2 types of ML uses: inference and prediction
- Model selection depends on model performance and bias-variance trade-off
- Model performance aims to minimize loss function $(y_i - \hat{y}_i)$

New Terminology:

- Supervised Learning
- Prediction Problem
- Loss Function
- Test MSE

Unit 1 Overview: Supervised Learning

Where We're Headed:

- Focus mostly on prediction problems

Unit 1 Overview: Supervised Learning

Where We're Headed:

- Focus mostly on prediction problems
- Overview of different models:
    - Estimation Goals
    - Basic Model Intuition
    - Loss Function (and some technical details)

Unit 1 Overview: Supervised Learning

Where We're Headed:

- Focus mostly on prediction problems
- Overview of different models:
  - Estimation Goals
  - Basic Model Intuition
  - Loss Function (and some technical details)
- Overview of different model selection techniques:
  - Advantages and Disadvantages
  - Assessment/Performance Metrics
  - Potential Applications

Agenda

1. Review: Regression Methods
   - Estimation Goals
   - Model Assessment
2. Classification Methods
   - Estimation Goals
   - Conditional Expectation
3. Classification Models
   - Logistic
   - K-Nearest Neighbors
   - Linear Discriminant Analysis (LDA)

Regression vs Classification

Regression Problems:

Classification Problems:

Regression vs Classification

Regression Problems:

- Predict quantitative or continuous outcome
- 1 Explanatory Variable $\rightarrow$ Bivariate Regression
- 2+ Explanatory Variables $\rightarrow$ Multivariate Regression
- Example: Stock Price, Test Scores, Vote Share

Classification Problems:

Regression vs Classification

Regression Problems:

- Predict quantitative or continuous outcome
- 1 Explanatory Variable $\rightarrow$ Bivariate Regression
- 2+ Explanatory Variables $\rightarrow$ Multivariate Regression
- Example: Stock Price, Test Scores, Vote Share

Classification Problems:

- Predict qualitative, categorical, or discrete outcome
- 2 Discrete Outcome Classes $\rightarrow$ Two-Class or Binary Classification Problem
- 3+ Discrete Outcome Classes $\rightarrow$ K-Class or Multi-Class Problem
- Example: Oscar Winner, Covid Exposure, Ethnic Exclusion

Review: Regression Methods

What is Linear Regression?

Linear regression aims to understand the linear relationship between quantitative response y and explanatory variable(s) x:

- Truth: $y \sim \beta_0 + \beta_1 d + \cdots + \beta_p x_p + \epsilon_i$
- Estimate: $\hat{y} \sim \hat{\beta}_0 + \hat{\beta}_1 X_1 + \ldots \hat{\beta}_p X_p$

What is Linear Regression?

Linear regression aims to understand the linear relationship between quantitative response y and explanatory variable(s) x:

- Truth: $y \sim \beta_0 + \beta_1 d + \cdots + \beta_p x_p + \epsilon_i$
- Estimate: $\hat{y} \sim \hat{\beta}_0 + \hat{\beta}_1 X_1 + \ldots \hat{\beta}_p X_p$
- Interpretation: β_i explains how a one-unit increase in x_i is associated with a change in y (controlling for other factors)

What is Linear Regression?

Linear regression aims to understand the linear relationship between quantitative response y and explanatory variable(s) x:

- Truth: $y \sim \beta_0 + \beta_1 d + \cdots + \beta_p x_p + \epsilon_i$
- Estimate: $\hat{y} \sim \hat{\beta}_0 + \hat{\beta}_1 X_1 + \ldots \hat{\beta}_p X_p$
- Interpretation: β_i explains how a one-unit increase in x_i is associated with a change in y (controlling for other factors)

Estimation Goals:

1. Infer how well explanatory variables X predict Y
2. Estimate parameters $(\beta_1, \ldots \beta_p)$ to minimize ϵ_i or $y_i - \hat{y}_i$

Estimate Parameters using Loss Function

Regression aims to estimate $\hat{\beta}_0$, $\hat{\beta}_1$, ...$\hat{\beta}_p$

- Estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ chosen to optimize least squares loss function.
- Loss function minimizes residual sum of squares (RSS)
-

$$
\begin{aligned}
\text{RSS} &= \sum_{i=1}^{n} \epsilon_i^2 \\
&= \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \\
&= \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_{i,1} - \ldots \beta_p x_{ip})
\end{aligned}
$$

Regression Assessment Tools

To assess the fit using RSS, we focus on the residuals:

- Mean Squared Error
- Residual Standard Error (RSE)
- F-Test

Mean Squared Error (MSE)

- Main Idea: Determine how well model minimizes ϵ

Mean Squared Error (MSE)

- Main Idea: Determine how well model minimizes ϵ
- Estimation Equation:

$$\mathsf{MSE} = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$$

Mean Squared Error (MSE)

- Main Idea: Determine how well model minimizes ϵ
- Estimation Equation:

$$\text{MSE} = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$$

Residual Standard Error (RSE)

- Estimation Equation:

$$\text{RSE} = \sqrt{\frac{\text{RSS}}{n-p-1}}$$

Residual Standard Error (RSE)

- Estimation Equation:

$$\text{RSE} = \sqrt{\frac{\text{RSS}}{n - p - 1}}$$

- Relation to MSE:
  - $\text{MSE} \approx \frac{1}{n}\text{RSS}$
  - Will produce similar results

F-Test

- Main Idea: Determine whether a variable or set of variables is important

F-Test

- Main Idea: Determine whether a variable or set of variables is important
- Analysis of Variance Test:

$$F = \frac{\text{RSS}_0 - \text{RSS}_1/q}{\text{RSS}_1/(n-p-1)}$$

  - RSS_0 from Null or Base Model
  - RSS_1 from Model 1 (more complex/incl. variable of interest)
  - p is the number of variables in null model
  - q is the number of new variables in model 1
- F-statistic tells us likelihood more complex model better fit to the data

Warning: R-Squared is a Poor Model Assessment Measure

R^2 tells us proportion of variation in Y explained by X

Warning: R-Squared is a Poor Model Assessment Measure

R^2 tells us proportion of variation in Y explained by X

$$R^2 = 1 - \frac{\text{RSS}}{\text{TSS}}$$

- $\text{TSS} = 1 - \sum_{i=1}^{n}(y_i - \bar{y})$

Warning: R-Squared is a Poor Model Assessment Measure

R^2 tells us proportion of variation in Y explained by X

$$R^2 = 1 - \frac{\text{RSS}}{\text{TSS}}$$

- TSS $= 1 - \sum_{i=1}^{n}(y_i - \bar{y})$
- Risks:
    - Adding more terms $\rightarrow$ model always decreases RSS, but not TSS (essentially can't minimize ϵ_i^2 as much when there are more parameters)
    - R^2 rewards more complex models and overfitting
- Advice: MSE $> R^2$

Limits to Linear Regression

Gauss-Markov Assumptions Often Violated:

Limits to Linear Regression

Gauss-Markov Assumptions Often Violated:

1. Variables Interact or Non-Additive

Limits to Linear Regression

Gauss-Markov Assumptions Often Violated:

1. Variables Interact or Non-Additive
2. Non-Linear Relationships

Limits to Linear Regression

Gauss-Markov Assumptions Often Violated:

1. Variables Interact or Non-Additive
2. Non-Linear Relationships
3. Heteroskadsticity (Non-Normal Errors)

Limits to Linear Regression

Gauss-Markov Assumptions Often Violated:

1. Variables Interact or Non-Additive
2. Non-Linear Relationships
3. Heteroskadsticity (Non-Normal Errors)
4. Collinearity

Limits to Linear Regression

Gauss-Markov Assumptions Often Violated:

1. Variables Interact or Non-Additive
2. Non-Linear Relationships
3. Heteroskadsticity (Non-Normal Errors)
4. Collinearity
5. High leverage/outlier observations

Classification Methods

What is a Classification Model?

Classification model aims to understand how different indicators (x) explain patterns in qualitative response y:

- Truth: No fixed f to describe data
- Estimate: $\hat{f}$ is “black box”

What is a Classification Model?

Classification model aims to understand how different indicators (x) explain patterns in qualitative response y:

- Truth: No fixed f to describe data
- Estimate: $\hat{f}$ is "black box"

Estimation Goals:

1. Assess **conditional probability** of observing outcome given indicators ($P(y \mid x)$)
2. Classify probabilities into distinct categories given threshold t ($I(P(y \mid x) > t)$), e.g.

$$P(y \mid x) \geq t = 1$$
$$P(y \mid x) < t = 0$$

Classification Model

Different Algorithms

Different Evaluation Tools

Classification Model

Different Algorithms

- Logit
- K-Nearest Neighbors (KNN)
- Linear Discriminant Analysis (LDA)

Different Evaluation Tools

Classification Model

Different Algorithms

- Logit
- K-Nearest Neighbors (KNN)
- Linear Discriminant Analysis (LDA)

Different Evaluation Tools

- Accuracy (Classification Rate)
- Precision
- Recall
- Area Under the Curve (AUC)

Conditional Expectation $\rightarrow$ Bayes Theorem

Main Idea: Our predicted probability $p(y \mid x)$ depends on available data and the model we use to fit the data.

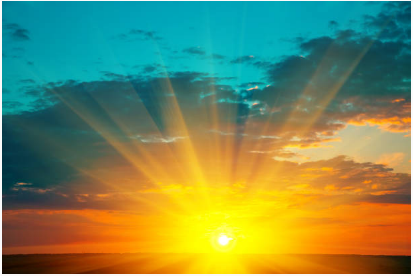
$$\textit{outcome} \propto \textit{model} \times \textit{data}$$

$$p(y \mid x) \propto p(x \mid y)p(x)$$

- $p(x)$: prior probability (data)
- $p(x \mid y)$: likelihood function (model)
- $p(y \mid x)$: posterior probability (outcome)

An Analogy

Question: Will the sun rise tomorrow?
Bayesian: Given prior information that the sun routinely rises p(yesterday), we can create a model $\hat{f}$ =p(yesterday | tomorrow) to make – with relative confidence – a posterior prediction that the sun will rise again p(tomorrow | yesterday).



How Bayes Theorem Affects Classification

We often use Bayes theorem to estimate classification model.

- Outcome variable is conditional probability $p(y \mid x)$
- May manipulate Bayes equation to estimate $p(y \mid x)$ given beliefs about data and model, e.g. LDA

3 Classification Models

So You Have a Classification Problem...

If you have a binary dependent variable, you could use a special type of linear regression $\rightarrow$ Linear Probability Model
$P(y = 1 \mid x) = \beta_0 + \beta_1 X_1 + \ldots \beta_p X_p$

So You Have a Classification Problem...

If you have a binary dependent variable, you could use a special type of linear regression $\rightarrow$ Linear Probability Model
$P(y = 1 \mid x) = \beta_0 + \beta_1 X_1 + \ldots \beta_p X_p$

Problems:

- Allows probabilities outside [0, 1] range
- Difficult to extend to more than 2 classes

So You Have a Classification Problem...

If you have a binary dependent variable, you could use a special type of linear regression $\rightarrow$ Linear Probability Model
$P(y = 1 \mid x) = \beta_0 + \beta_1 X_1 + \ldots \beta_p X_p$

Problems:

- Allows probabilities outside [0, 1] range
- Difficult to extend to more than 2 classes

Solution: Logistic regression?

3 Classification Models

1. **Logistic**
2. K-Nearest Neighbors
3. LDA

Logistic Regression

Main Aim:

- Model a binary dependent variable using a logit function
- Assumes parametric relationship such that $P(y \mid x) = f(X\beta)$

Logistic Regression

Main Aim:

- Model a binary dependent variable using a logit function
- Assumes parametric relationship such that $P(y \mid x) = f(X\beta)$

Estimation Goals:

- Assess probability of observing each outcome $P(y \mid x)$:
  - $Pr(y = 1 \mid x)$, e.g. event occurs, outcome present
  - $Pr(y = 0 \mid x)$, e.g. event doesn't occur, outcome not present
- Estimate parameters $(\beta_1, \ldots \beta_p)$ to optimize maximum likelihood function

Logit Function

Given an input x_0, we predict the response using a logit function:

$$\hat{y_0} = \text{argmax}[P(y = 1 \mid x = x_0)]$$

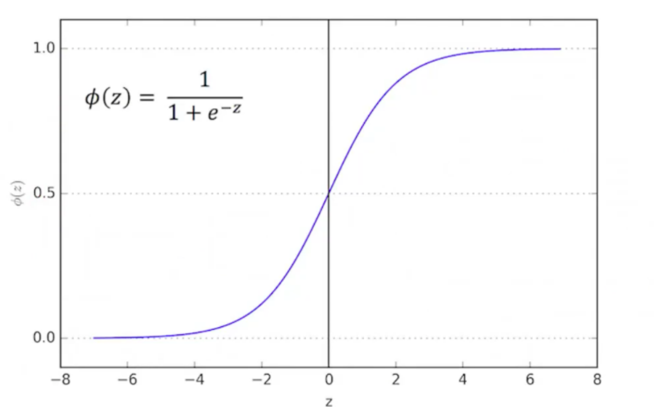
Logit Function

Given an input x_0, we predict the response using a logit function:

$$\hat{y_0} = \text{argmax}[P(y = 1 \mid x = x_0)]$$

Logit Function (Sigmoid Function):

$$P(y = 1 \mid x) = f(X\beta) = \frac{e^{X\beta}}{1 + e^{X\beta}} = \frac{1}{1 + e^{-X\beta}}$$



Fitting a logistic regression

We model the joint probability using logit function:

$$P(y = 1 \mid X) = \frac{e^{X\beta}}{1 + e^{X\beta}} = \frac{e^{\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p}}{1 + e^{\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p}}$$

$$P(y = 0 \mid X) = 1 - P(y = 1 \mid X) = \frac{1}{1 + e^{\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p}}$$

This is the same as using a linear model for the log odds:

$$log[\frac{P(y = 1 \mid X)}{P(Y = 0 \mid X)}] = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$

Loss Function

Logistic functions are usually fit by maximum likelihood estimation

Procedure:

- Find model that maximizes likelihood ($L(x \mid y)$) or probability of observing outcome given data $p(y \mid x)$.

$$L(x \mid y) = \prod_{i=1}^{n} p(y = y_i \mid x_i)$$

  This is often written as the log-likelihood:

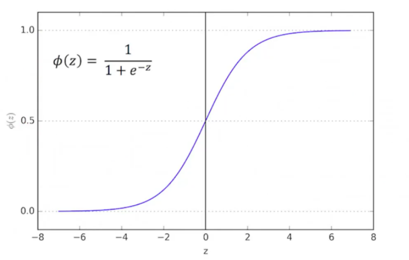
$$lnL(x \mid y) = \sum_{i=1}^{n} p(y = y_i \mid x_i)$$

- Choose estimates $\hat{\beta}_0, \ldots \hat{\beta}_p$ which maximize this likelihood
- Solve with analytical, grid-search, or numerical methods, e.g. Newton's algorithm

Logistic Interpretation

β_i tells us average change in *log odds* with 1-unit increase in x_i

- Amount that p(x) changes due to 1-unit change in X depends on current value of X
- Odds ratio: $exp(\beta_i)$ tells us how the odds change with 1-unit increase in β_i holding all other variables constant



Transform Probabilities into Distinct Categories

To interpret odds back to binary outcome, classify logit probabilities into distinct categories based on given threshold t, e.g. ($I(P(y \mid x) > t)$)

$$P(y \mid x) \geq t = 1$$
$$P(y \mid x) < t = 0$$

Rule of Thumb for 2 Class Problem is $t = 0.5$:

$$P(y = 1 \mid x) \geq 0.5 = 1$$
$$P(y = 0 \mid x) < 0.5 = 0$$

Advantages and Disadvantages to Logit

Advantages:

Disadvantages:

Advantages and Disadvantages to Logit

Advantages:

- Workhorse model for binary DV
- Fits most binary classification problems
- Lots of technical support available

Disadvantages:

Advantages and Disadvantages to Logit

Advantages:

- Workhorse model for binary DV
- Fits most binary classification problems
- Lots of technical support available

Disadvantages:

- Algorithm convergence problems
- Collinearity $\rightarrow$ unstable coefficients ($p > n$)
- Well-separated class $\rightarrow$ unstable coefficients
- Standard error estimates for panel data a mess (heteroskedasticity)

Is My Logit Model Any Good?

Model Assessment Tools for Classification Problems:

- Test Prediction Error (Test Error Rate)
- Accuracy
- Sensitivity (Recall)
- Specificity
- Precision
- F-Score (F1 Score)

Test Error Rate

One of the most common ways to assess classification model is the test error rate (0-1 loss).

- AKA MSE for Classification Problems
- Assign $\hat{y}$ to 0/1 categories based on $I(P(y \mid x) > t)$
- Compare average test prediction error using test data $(x'_1, y'_1), (x'_2, y'_2), \ldots (x_m, y_m)$

$$\frac{1}{m} \sum_{i=1}^{m} 1(y'_i \neq \hat{y}'_i)$$

Confusion Matrix Tells Us Model Performance

Confusion Matrix describes model performance for test data:

	Actual	
Guess	0	1
0	True Negative (TN)	False Negative (FN)
1	False Positive (FP)	True Positive (TP)

Confusion Matrix Tells Us Model Performance

Confusion Matrix describes model performance for test data:

	Actual	
Guess	0	1
0	True Negative (TN)	False Negative (FN)
1	False Positive (FP)	True Positive (TP)

Accuracy $=$ Error Rate $= \frac{TN+TP}{TN+FN+FP+TP}$

Confusion Matrix Tells Us Model Performance

Confusion Matrix describes model performance for test data:

	Actual	
Guess	0	1
0	True Negative (TN)	False Negative (FN)
1	False Positive (FP)	True Positive (TP)

Accuracy = Error Rate = $\frac{TN+TP}{TN+FN+FP+TP}$

Sensitivity/Recall = $\frac{TP}{TP+FN}$

Confusion Matrix Tells Us Model Performance

Confusion Matrix describes model performance for test data:

	Actual	
Guess	0	1
0	True Negative (TN)	False Negative (FN)
1	False Positive (FP)	True Positive (TP)

Accuracy = Error Rate = $\frac{TN+TP}{TN+FN+FP+TP}$

Sensitivity/Recall = $\frac{TP}{TP+FN}$

Specificity = $\frac{TN}{TN+FP}$

Confusion Matrix Tells Us Model Performance

Confusion Matrix describes model performance for test data:

	Actual	
Guess	0	1
0	True Negative (TN)	False Negative (FN)
1	False Positive (FP)	True Positive (TP)

Accuracy $=$ Error Rate $= \frac{TN+TP}{TN+FN+FP+TP}$

Sensitivity/Recall $= \frac{TP}{TP+FN}$

Specificity $= \frac{TN}{TN+FP}$

Precision $= \frac{TP}{TP+FP}$

Confusion Matrix Tells Us Model Performance

Confusion Matrix describes model performance for test data:

	Actual	
Guess	0	1
0	True Negative (TN)	False Negative (FN)
1	False Positive (FP)	True Positive (TP)

Accuracy $=$ Error Rate $= \frac{TN+TP}{TN+FN+FP+TP}$

Sensitivity/Recall $= \frac{TP}{TP+FN}$

Specificity $= \frac{TN}{TN+FP}$

Precision $= \frac{TP}{TP+FP}$

F-Score $= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$

Comparing Difference Performance Metrics

Compare performance to the No Information Rate (NIR)

No Information Rate (NIR):

Accuracy: Overall Classification Rate

Sensitivity/Recall: True Positive Rate

Specificity: True Negative Rate

Comparing Difference Performance Metrics

Compare performance to the No Information Rate (NIR)

No Information Rate (NIR):

- Predicted accuracy if we always predicted majority class (0)
- Imbalanced data (large majority class) $\rightarrow$ large NIR

Accuracy: Overall Classification Rate

Sensitivity/Recall: True Positive Rate

Specificity: True Negative Rate

Comparing Difference Performance Metrics

Compare performance to the No Information Rate (NIR)

No Information Rate (NIR):

- Predicted accuracy if we always predicted majority class (0)
- Imbalanced data (large majority class) $\rightarrow$ large NIR

Accuracy: Overall Classification Rate

- Good for general model performance
- Bad model: accuracy $<$ NIR

Sensitivity/Recall: True Positive Rate

Specificity: True Negative Rate

Comparing Difference Performance Metrics

Compare performance to the No Information Rate (NIR)

No Information Rate (NIR):

- Predicted accuracy if we always predicted majority class (0)
- Imbalanced data (large majority class) $\rightarrow$ large NIR

Accuracy: Overall Classification Rate

- Good for general model performance
- Bad model: accuracy $<$ NIR

Sensitivity/Recall: True Positive Rate

- Strive for high sensitivity
- Will be **low** if there is class imbalance

Specificity: True Negative Rate

Comparing Difference Performance Metrics

Compare performance to the No Information Rate (NIR)

No Information Rate (NIR):

- Predicted accuracy if we always predicted majority class (0)
- Imbalanced data (large majority class) $\rightarrow$ large NIR

Accuracy: Overall Classification Rate

- Good for general model performance
- Bad model: accuracy $<$ NIR

Sensitivity/Recall: True Positive Rate

- Strive for high sensitivity
- Will be **low** if there is class imbalance

Specificity: True Negative Rate

- Strive for high specificity
- Will be **high** if there is class imbalance

3 Classification Models

1. Logistic
2. **K-Nearest Neighbors**
3. LDA

K-Nearest Neighbors

Main Idea:

- “birds of a feather flock together”

K-Nearest Neighbors

Main Idea:

- “birds of a feather flock together” (similar observations exist in close proximity to each other)

K-Nearest Neighbors

Main Idea:

- "birds of a feather flock together" (similar observations exist in close proximity to each other)
- Non-parametric approach to understand relation between x and y

K-Nearest Neighbors

Main Idea:

- "birds of a feather flock together" (similar observations exist in close proximity to each other)
- Non-parametric approach to understand relation between x and y

Estimation Goal:

Assign observation to most likely class j given neighboring values (N_0):

$$p(y = j \mid X = x_0) = \frac{1}{K} \sum_{i \in N_0} I(y_i = j)$$

KNN Loss Function

Procedure:

- Given integer K and test observation x_0, we identify K points in training data that are closest to x_0
- Label these neighboring points N_0

KNN Loss Function

Procedure:

- Given integer K and test observation x_0, we identify K points in training data that are closest to x_0
- Label these neighboring points N_0
- Estimate conditional probability $p(y = j \mid X = x_0)$ as fraction of N_0 points equal to j (**Bayes Classifier**):

$$p(y = j \mid X = x_0) = \frac{1}{K} \sum_{i \in N_0} I(y_i = j)$$

KNN Loss Function

Procedure:

- Given integer K and test observation x_0, we identify K points in training data that are closest to x_0
- Label these neighboring points N_0
- Estimate conditional probability $p(y = j \mid X = x_0)$ as fraction of N_0 points equal to j (**Bayes Classifier**):

$$p(y = j \mid X = x_0) = \frac{1}{K} \sum_{i \in N_0} I(y_i = j)$$

- Apply Bayes rule and classify test observation x_0 to class with largest probability $p(x_0 \mid y = j)$

Example: KNN assigns color based on nearest observations

Want to assign input data (x) a color (orange or purple) based on $K = 3$ nearest neighbors. Predict color of the majority of neighbors.

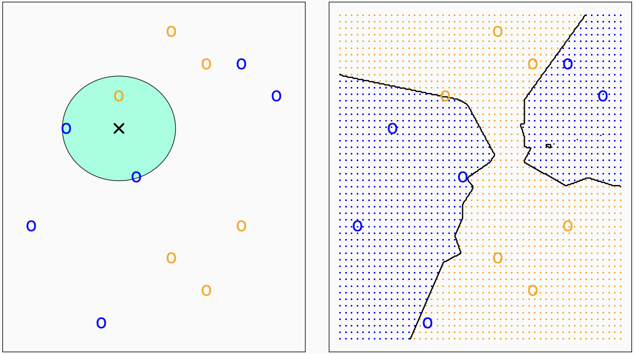
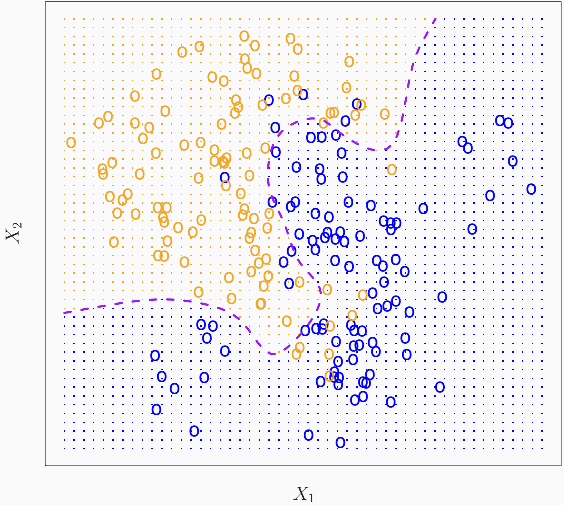


Figure 1: Figure 2.14

KNN has a decision boundary

Bayes Decision Boundary (dashed line) travels through points where probability of belonging to either class is 50%.



Higher K → Smoother Decision Boundary

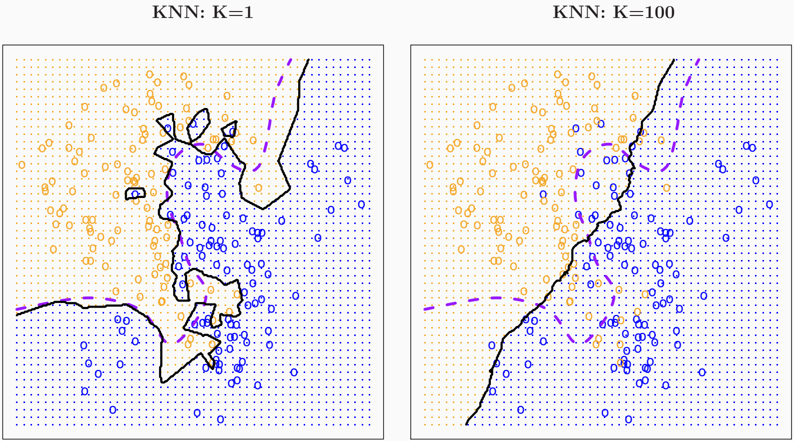


Figure 3: Figure 2.16

KNN Model Performance

Assess KNN Model Based on Test Error Rate (Accuracy):

$$\frac{1}{m}\sum_{i=1}^{m} 1(y_i' \neq \hat{y}_i')$$

Advantages and Disadvantages to KNN

Advantages:

Disadvantages:

Advantages and Disadvantages to KNN

Advantages:

- Very few assumptions about true f
- Flexibility
- Better than regression and logit if true f non-parametric

Disadvantages:

Advantages and Disadvantages to KNN

Advantages:

- Very few assumptions about true f
- Flexibility
- Better than regression and logit if true f non-parametric

Disadvantages:

- Hard to determine optimal K
  - Small K overly flexible and high variance
  - Large K too inflexible (linear) and high bias
- Performs poorly as p increases (curse of dimensionality)

3 Classification Models

1. Logistic
2. K-Nearest Neighbors
3. **LDA**

Linear Discriminant Analysis (LDA)

Main Idea:

- Indirect approach to estimate $P(y = j \mid X = x)$
- Estimate posterior probability observation belongs in j class given prior probability (data) and likelihood function (model)

Linear Discriminant Analysis (LDA)

Main Idea:

- Indirect approach to estimate $P(y = j \mid X = x)$
- Estimate posterior probability observation belongs in j class given prior probability (data) and likelihood function (model)

Estimation Goals:

- Predict $P(y = j \mid X = x)$ for multi-class problem ($K \geq 2$)
- Estimate parameters ($\beta_1, \ldots \beta_p$) to optimize Bayes classifier

Return of the Bayes

- In logit and KNN, we estimate $P(y = j \mid x)$ directly.
- In LDA, we estimate $P(y = j \mid x)$ indirectly. Specifically, we estimate:
    - $\hat{p}(x \mid y)$: given the data, what is the distribution of classes?
    - $\hat{p}(y)$: how likely is each class to occur?
- Use this information to re-arrange Bayes rule and estimate $p(y = j \mid x)$

$$\text{posterior} \sim \text{likelihood} \times \text{data}$$

$$\hat{p}(y = j \mid X = x) = \frac{\hat{p}(X = x \mid y = j)\hat{p}(y = j)}{\sum_j \hat{p}(X = x \mid y = j)\hat{p}(y = j)}$$

Plugging in LDA Values

- We model $\hat{p}(x \mid y) = \hat{f}_j(x)$ as a multivariate normal distribution
- We model $\hat{p}(y = j)$ as fraction of training sample observations in class j
- Assign class based on largest posterior probability $\hat{p}(y = j \mid X = x)$

$$\hat{p}(y = j \mid X = x) = \frac{\hat{p}(X = x \mid y = j)\hat{p}(y = j)}{\sum_j \hat{p}(X = x \mid y = j)\hat{p}(y = j)}$$

LDA has linear decision boundaries

Similar to KNN decision boundaries, but less flexible.

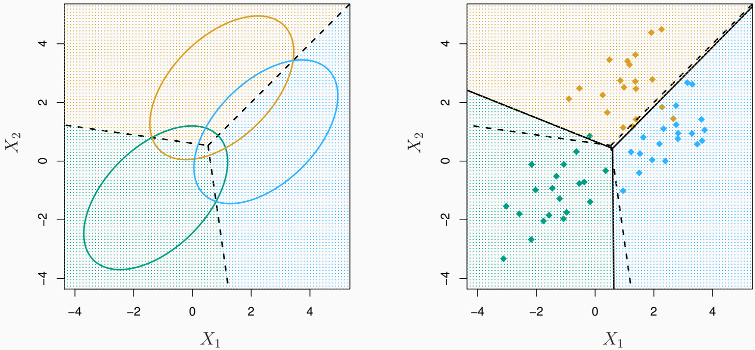


Figure 4: Figure 4.6

LDA Model Assessment

Rule of Thumb: Determine threshold for given class j and create a confusion matrix to examine metrics for j ...

- Accuracy (but careful)
- Sensitivity/Recall
- Specificity
- Precision
- F-Score (F1 Score)

Advantages and Disadvantages to LDA

Advantages:

Disadvantages:

Advantages and Disadvantages to LDA

Advantages:

- Similar estimates as logit regression
- Performs better than logit if n is small
- Performs better than logit if classes well-separated
- Performs better than logit and KNN if true f linear
- Good for $3+$ response classes

Disadvantages:

Advantages and Disadvantages to LDA

Advantages:

- Similar estimates as logit regression
- Performs better than logit if n is small
- Performs better than logit if classes well-separated
- Performs better than logit and KNN if true f linear
- Good for $3+$ response classes

Disadvantages:

- Greedy algorithm $\rightarrow$ minimize global not local error
- Poor performance if Gauss-Markov assumptions violated

Conclusion

- Regression $\rightarrow$ quantitative responses; classification $\rightarrow$ qualitative responses

Conclusion

- Regression $\rightarrow$ quantitative responses; classification $\rightarrow$ qualitative responses
- Loss functions vary by method:
  - Regression estimates $\hat{y}$ minimizes RSS
  - Logistic estimates $\hat{p}(y \mid x)$ by maximum likelihood
  - KNN and LDA estimate $\hat{p}(y \mid x)$ by Bayes classifier

Conclusion

- Regression $\rightarrow$ quantitative responses; classification $\rightarrow$ qualitative responses
- Loss functions vary by method:
  - Regression estimates $\hat{y}$ minimizes RSS
  - Logistic estimates $\hat{p}(y \mid x)$ by maximum likelihood
  - KNN and LDA estimate $\hat{p}(y \mid x)$ by Bayes classifier
- Lots of different performance metrics: accuracy, sensitivity, specificity, etc.

Conclusion

- Regression $\rightarrow$ quantitative responses; classification $\rightarrow$ qualitative responses
- Loss functions vary by method:
    - Regression estimates $\hat{y}$ minimizes RSS
    - Logistic estimates $\hat{p}(y \mid x)$ by maximum likelihood
    - KNN and LDA estimate $\hat{p}(y \mid x)$ by Bayes classifier
- Lots of different performance metrics: accuracy, sensitivity, specificity, etc.
- Best classification model depends on beliefs about true f and number of: classes (j), parameters (p), and observations (n)