Classification

Damek Davis
School of ORIE, Cornell University
ORIE 4740 Lec 5–6 (Feb 8, 10)

Announcements

Nonlinear models

Suppose we fit a linear model

$$Y \approx \beta_0 + \beta_1 X.$$

to a given data set. If the relationship is truly linear, what kind of pattern should we see in the *residual plot*?

A. a nonlinear pattern
B. a linear pattern
C. a lack of a pattern.

Nonlinear models

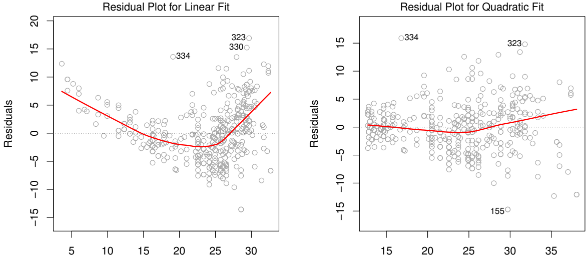
Suppose we fit a linear model

$$Y \approx \beta_0 + \beta_1 X.$$

to a given data set. If the relationship is truly linear, what kind of pattern should we see in the *residual plot*?

A. a nonlinear pattern
B. a linear pattern
C. a lack of a pattern.

Residual plots: $y_i - \hat{y}_i$ versus $\hat{y}_i$



Classification

(Reading: ISLR Sections 4.1–4.3, 4.5–4.6, 2.2.3)

Recall: statistical learning

$$Y = f(X_1, X_2, \ldots, X_p) + \epsilon$$

Classification

(Reading: ISLR Sections 4.1–4.3, 4.5–4.6, 2.2.3)

Recall: statistical learning

$$Y = f(X_1, X_2, \ldots, X_p) + \epsilon$$

The `Default` dataset:

$X_1 =$ balance
$X_2 =$ income
$Y =$ default or not

Classification

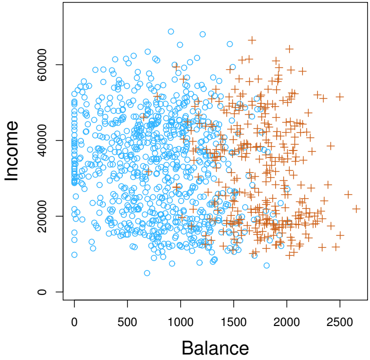
(Reading: ISLR Sections 4.1–4.3, 4.5–4.6, 2.2.3)

Recall: statistical learning

$$Y = f(X_1, X_2, \ldots, X_p) + \epsilon$$

The `Default` dataset:

X_1 = balance
X_2 = income
Y = default or not



- Classification: Given balance and income, predict default or not.

Classification

Why not linear regression with Y converted to dummy variables?

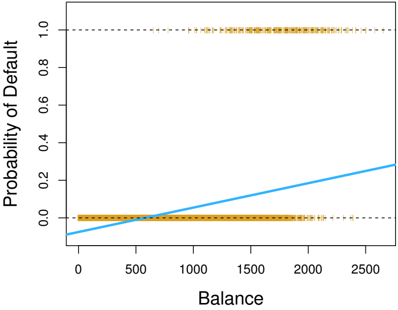
$$\begin{aligned} Y &= f(X_1, \ldots, X_p) + \epsilon \\ &= \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon \end{aligned}$$

Classification

Why not linear regression with Y converted to dummy variables?

$$\begin{aligned} Y &= f(X_1, \ldots, X_p) + \epsilon \\ &= \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon \end{aligned}$$

- Issue 1:



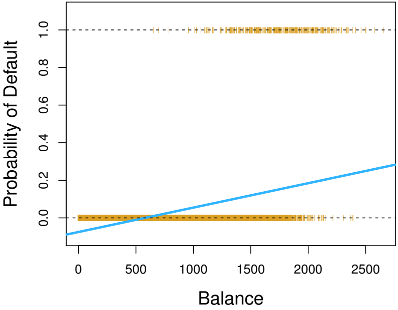
$$\Pr(Y = 1 \mid X) = \beta_0 + \beta_1 X$$

Classification

Why not linear regression with Y converted to dummy variables?

$$\begin{aligned} Y &= f(X_1, \ldots, X_p) + \epsilon \\ &= \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon \end{aligned}$$

- Issue 1:



$$\Pr(Y = 1 \mid X) = \beta_0 + \beta_1 X$$

- Issue 2: Multi-class problems

Suppose we already have the model from the previous slide. How could we find the probability that an individual will not default under this model?

A. Fit another linear regression model after encoding default as 0 and not default as 1

B. Use

$$\Pr(Y = 0 \mid X) = 1 - \Pr(Y = 1 \mid X)$$

C. Both are valid choices

Classification

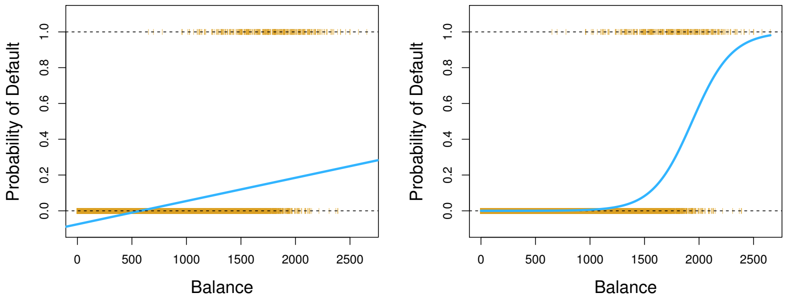
Classification techniques:

- Logistic Regression
- Linear Discriminant Analysis (not covered; ISLR Sec 4.4)
- K-Nearest Neighbor

- Support Vector Machines (later this semester)
- Tree-based Methods (later this semester)

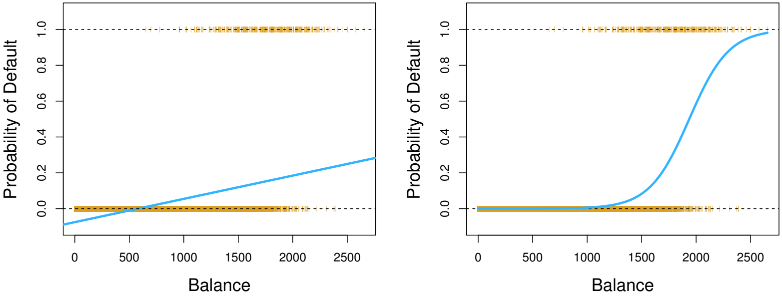
Logistic Regression

(ISLR Sec 4.3)



Logistic Regression

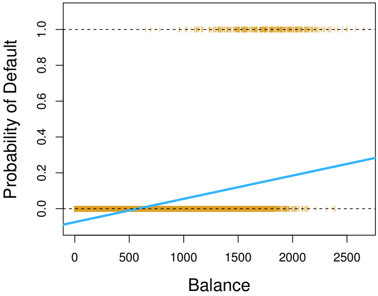
(ISLR Sec 4.3)

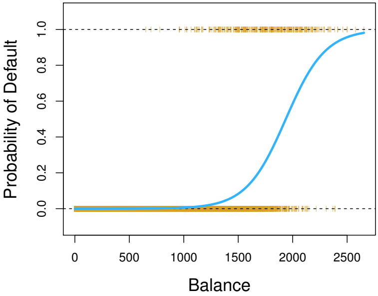


- Counterpart of linear regression

Logistic Regression

(ISLR Sec 4.3)

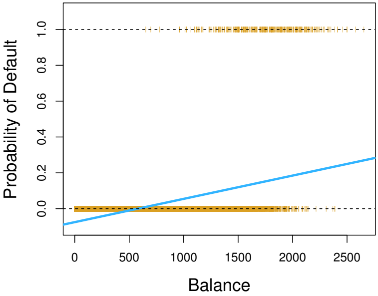


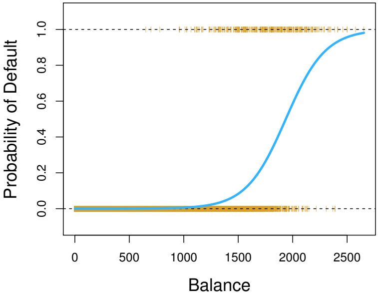


- Counterpart of linear regression
- Simple. Easy to interpret. Not too flexible (avoid overfitting)

Logistic Regression

(ISLR Sec 4.3)





- Counterpart of linear regression
- Simple. Easy to interpret. Not too flexible (avoid overfitting)
- Often good performance

Logistic Regression

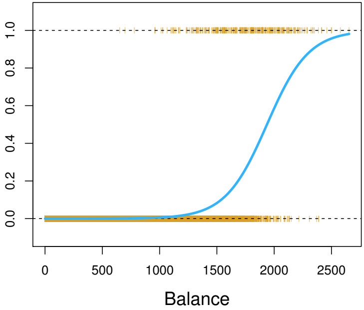
Try to predict $\Pr(Y = 1|\vec{X})$

Logistic Regression

Try to predict $\Pr(Y = 1|\vec{X})$

The Logistic Model:

$$\Pr(Y = 1|\vec{X}) = \text{logistic\_func}(\vec{X}^\top \vec{\beta}) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$



Logistic Regression

Try to predict $\Pr(Y = 1|\vec{X})$

The Logistic Model:

$$\Pr(Y = 1|\vec{X}) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

Equivalently:

$$\log\left(\frac{\Pr(Y = 1|\vec{X})}{1 - \Pr(Y = 1|\vec{X})}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Logistic Regression

Two equivalent views of logistic regression:

- "Probability $=$ logistic function of linear function of predictors"

$$\Pr(Y = 1|\vec{X}) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

- "Log odds $=$ linear combination of predictors"

$$\log\left(\frac{\Pr(Y = 1|\vec{X})}{1 - \Pr(Y = 1|\vec{X})}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Probability vs Odds vs Logodds

prob	odds	logodds
.001	.001001	-6.906755
.01	.010101	-4.59512
.15	.1764706	-1.734601
.2	.25	-1.386294
.25	.3333333	-1.098612
.3	.4285714	-.8472978
.35	.5384616	-.6190392
.4	.6666667	-.4054651
.45	.8181818	-.2006707
.5	1	0
.55	1.222222	.2006707
.6	1.5	.4054651
.65	1.857143	.6190392
.7	2.333333	.8472978
.75	3	1.098612
.8	4	1.386294
.85	5.666667	1.734601
.9	9	2.197225
.999	999	6.906755
.9999	9999	9.21024

Logistic Regression: Estimation

Model:

$$\Pr_{\vec{\beta}}(Y = 1|\vec{X}) = \frac{e^{\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p}}{1 + e^{\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p}}$$

- Make a probability distribution that thinks the observed data is likely.

Logistic Regression: Estimation

Model:

$$\Pr_{\vec{\beta}}(Y = 1|\vec{X}) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

- Make a probability distribution that thinks the observed data is likely.
- Maximum Likelihood Estimator (MLE):

Find $\beta_0, \ldots, \beta_p$ that maximize the "likelihood":

$$\Pr_{\vec{\beta}}(\text{Seeing the response values in the training data})$$

Logistic Regression: Estimation

Model:

$$\Pr_{\vec{\beta}}(Y = 1|\vec{X}) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

- Make a probability distribution that thinks the observed data is likely.
- Maximum Likelihood Estimator (MLE):

Find $\beta_0, \ldots, \beta_p$ that maximize the "likelihood":

$$\Pr_{\vec{\beta}}(\text{Seeing the response values in the training data})$$
$$= \prod_{i=1}^{n} \Pr_{\vec{\beta}}(Y = y_i|\vec{X} = \vec{x}_i)$$

Logistic Regression: Estimation

Model:

$$\Pr_{\vec{\beta}}(Y = 1|\vec{X}) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

- Make a probability distribution that thinks the observed data is likely.
- Maximum Likelihood Estimator (MLE):

Find $\beta_0, \ldots, \beta_p$ that maximize the "likelihood":

$$\begin{aligned}
&\Pr_{\vec{\beta}}(\text{Seeing the response values in the training data}) \\
&= \prod_{i=1}^{n} \Pr_{\vec{\beta}}(Y = y_i|\vec{X} = \vec{x}_i) \\
&= \prod_{i:y_i=1} \frac{e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_p x_{ip}}}{1 + e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_p x_{ip}}} \prod_{i:y_i=0} \frac{1}{1 + e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_p x_{ip}}}
\end{aligned}$$

Logistic Regression: Estimation

Model:

$$\Pr_{\vec{\beta}}(Y = 1|\vec{X}) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

- Make a probability distribution that thinks the observed data is likely.
- Maximum Likelihood Estimator (MLE):

Find $\beta_0, \ldots, \beta_p$ that maximize the "likelihood":

$$\begin{aligned}
&\Pr_{\vec{\beta}}(\text{Seeing the response values in the training data}) \\
&= \prod_{i=1}^{n} \Pr_{\vec{\beta}}(Y = y_i|\vec{X} = \vec{x}_i) \\
&= \prod_{i:y_i=1} \frac{e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_p x_{ip}}}{1 + e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_p x_{ip}}} \prod_{i:y_i=0} \frac{1}{1 + e^{\beta_0+\beta_1 x_{i1}+\cdots+\beta_p x_{ip}}}
\end{aligned}$$

- R command: `glm`

Logistic Regression: Prediction

Find MLE solution: $\hat{\beta}_0, \ldots, \hat{\beta}_p$

Probabilistic prediction:

- Given a new data point $\vec{X} = (X_1, X_2, \ldots, X_p)$, predict

$$\hat{\Pr}(Y = 1|\vec{X}) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p}}$$

- R command: `predict`

Logistic Regression: Prediction

Find MLE solution: $\hat{\beta}_0, \ldots, \hat{\beta}_p$

Probabilistic prediction:

- Given a new data point $\vec{X} = (X_1, X_2, \ldots, X_p)$, predict

$$\hat{\Pr}(Y = 1|\vec{X}) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p}}$$

- R command: `predict`

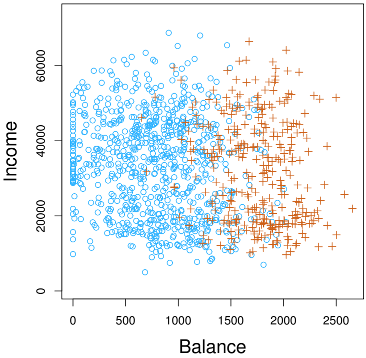
Binary prediction:

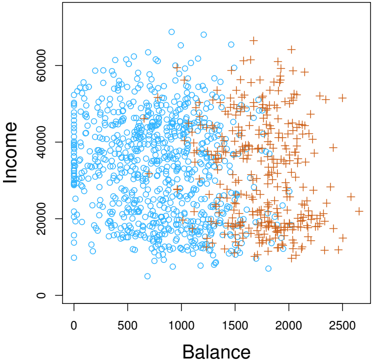
- Choose some threshold. Predict

$$\hat{Y} = 1 \quad \text{if and only if} \quad \hat{\Pr}(Y = 1|\vec{X}) \geq \text{threshold}$$

Which of the following is the classification boundary produced by logistic regression (with two predictors `Income` and `Balance`)

- **A** Left.
- **B** Right.
- **C** Both are possible.
- **D** Neither is possible.





Logistic Regression: Evaluating Model Accuracy

- `Default` dataset:

X_1 = balance, X_2 = income, X_3 = student or not (categorical)
Y = default or not

- Estimation:

```
> logistic.fit = glm(default ~ balance+income+student, data=Default, family=binomial)
> summary(logistic.fit)

Call:
glm(formula = default ~ balance + income + student, family = binomial,
    data = Default)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4691  -0.1418  -0.0557  -0.0203   3.7383

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.087e+01  4.923e-01 -22.080  < 2e-16 ***
balance      5.737e-03  2.319e-04  24.738  < 2e-16 ***
income       3.033e-06  8.203e-06   0.370  0.71152
studentYes  -6.468e-01  2.363e-01  -2.738  0.00619 **
```

Logistic Regression: Evaluating Model Accuracy

```
> logistic.fit = glm(default ~ balance+income+student, data=Default, family=binomial)
> summary(logistic.fit)

Call:
glm(formula = default ~ balance + income + student, family = binomial,
    data = Default)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4691  -0.1418  -0.0557  -0.0203   3.7383

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.087e+01  4.923e-01 -22.080  < 2e-16 ***
balance      5.737e-03  2.319e-04  24.738  < 2e-16 ***
income       3.033e-06  8.203e-06   0.370  0.71152
studentYes  -6.468e-01  2.363e-01  -2.738  0.00619 **
```

Which variable would you recommend dropping from the model?

- **A.** Balance
- **B.** Income
- **C.** Student
- **D.** None

Logistic Regression: Evaluating Model Accuracy

```
> logistic.fit = glm(default ~ balance+income+student, data=Default, family=binomial)
> summary(logistic.fit)

Call:
glm(formula = default ~ balance + income + student, family = binomial,
    data = Default)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4691  -0.1418  -0.0557  -0.0203   3.7383

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.087e+01  4.923e-01 -22.080  < 2e-16 ***
balance      5.737e-03  2.319e-04  24.738  < 2e-16 ***
income       3.033e-06  8.203e-06   0.370  0.71152
studentYes  -6.468e-01  2.363e-01  -2.738  0.00619 **
```

Under this model, are students more or less likely to default on their credit cards?

A. More likely

B. Less likely

Logistic Regression: Evaluating Model Accuracy

```
> logistic.fit = glm(default ~ balance+income+student, data=Default, family=binomial)
> summary(logistic.fit)

Call:
glm(formula = default ~ balance + income + student, family = binomial,
    data = Default)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4691  -0.1418  -0.0557  -0.0203   3.7383

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.087e+01  4.923e-01 -22.080  < 2e-16 ***
balance      5.737e-03  2.319e-04  24.738  < 2e-16 ***
income       3.033e-06  8.203e-06   0.370  0.71152
studentYes  -6.468e-01  2.363e-01  -2.738  0.00619 **
```

From this table, can you conclude that income is likely to be unrelated to default status?

- **A.** Yes
- **B.** No
- **C.** It is more subtle than that

Recap: Logistic Regression

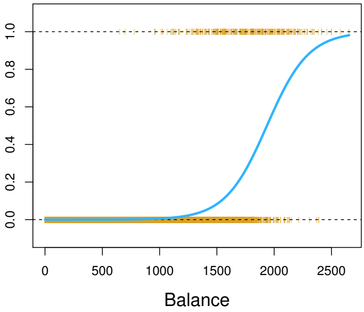
Try to predict $\Pr(Y = 1|\vec{X})$

Recap: Logistic Regression

Try to predict $\Pr(Y = 1|\vec{X})$

The Logistic Model:

$$\Pr(Y = 1|\vec{X}) = \text{logistic\_func}(\vec{X}^\top\vec{\beta}) \ = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$



Fit by MLE (Maximize probability of seeing observed data).

Logistic Regression: Evaluating Model Accuracy

- `Default` dataset:

X_1 = balance, X_2 = income, X_3 = student or not (categorical)
Y = default or not

- Estimation:

```
> logistic.fit = glm(default ~ balance+income+student, data=Default, family=binomial)
> summary(logistic.fit)

Call:
glm(formula = default ~ balance + income + student, family = binomial,
    data = Default)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4691  -0.1418  -0.0557  -0.0203   3.7383

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.087e+01  4.923e-01 -22.080  < 2e-16 ***
balance      5.737e-03  2.319e-04  24.738  < 2e-16 ***
income       3.033e-06  8.203e-06   0.370  0.71152
studentYes  -6.468e-01  2.363e-01  -2.738  0.00619 **
```

Logistic Regression: Prediction

Find MLE solution: $\hat{\beta}_0, \ldots, \hat{\beta}_p$

Probabilistic prediction:

- Given a new data point $\vec{X} = (X_1, X_2, \ldots, X_p)$, predict

$$\hat{\Pr}(Y = 1|\vec{X}) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p}}$$

- R command: `predict`

Logistic Regression: Prediction

Find MLE solution: $\hat{\beta}_0, \ldots, \hat{\beta}_p$

Probabilistic prediction:

- Given a new data point $\vec{X} = (X_1, X_2, \ldots, X_p)$, predict

$$\hat{\Pr}(Y = 1|\vec{X}) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p}}$$

- R command: `predict`

Binary prediction:

- Choose some threshold. Predict

$$\hat{Y} = 1 \quad \text{if and only if} \quad \hat{\Pr}(Y = 1|\vec{X}) \geq \text{threshold}$$

Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

	True Class		
		Negative	Positive
Predicted class	Negative	TN	FN
	Positive	FP	TP

Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

Suppose you are sure you will pass ORIE 4740, but then you get your grades back and see that you failed. This is an example of a...

A. True Negative

B. False Positive

C. False Negative

D. True Positive

Assume (pass = positive, fail = negative).

Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred    No   Yes
          No   9627   228
          Yes    40   105
>
```

Suppose you are sure you will fail ORIE 4740 and then you get your grades back and see that you failed. This is an example of a...

A. True Negative

B. False Positive

C. False Negative

D. True Positive

Assume (pass = positive, fail = negative).

Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred    No  Yes
          No   9627  228
          Yes    40  105
>
```

Suppose you are sure you will pass ORIE 4740 and then you get your grades back and see that you passed. This is an example of a...

A. True Negative

B. False Positive

C. False Negative

D. True Positive

Assume (pass = positive, fail = negative).

Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

Suppose you are sure you will fail ORIE 4740, but then you get your grades back and see that you passed. This is an example of a...

A. True Negative

B. False Positive

C. False Negative

D. True Positive

Assume (pass = positive, fail = negative).

Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

How many people in the data set were incorrectly assigned to the *no default* category?

A. 9627

B. 228

C. 40

D. 105

Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

How many people in the data set were incorrectly assigned to the *default* category?

A. 9627

B. 228

C. 40

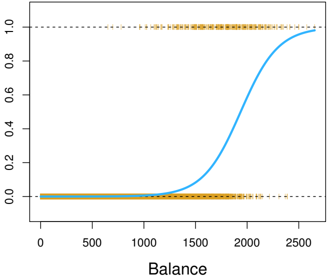
D. 105

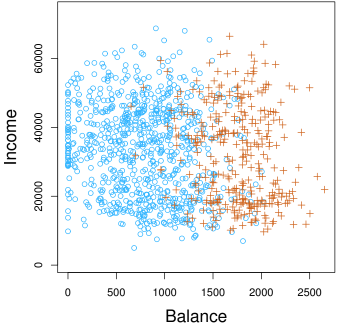
Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```





Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

- Prediction (threshold = 0.2):

```
> logistic.pred[ logistic.probs>0.2] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9390  130
          Yes  277  203
>
```

Logistic Regression: Evaluating Model Accuracy

- Prediction (threshold = 0.5):

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

- Prediction (threshold = 0.2):

```
> logistic.pred[ logistic.probs>0.2] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9390  130
          Yes  277  203
> |
```

CC companies do not want to extend credit to consumers who will ultimately default. They are less concerned with denying credit to those who will not default. If you were a credit card company, which threshold would you prefer?

A. .5

B. .2

Classification Terminology

- Overall error rate $= \frac{\#\text{errors}}{\#\text{data points}} = \frac{\text{FP + FN}}{\text{TN + FP + TP + FN}}$
- FPR $=$ Type-I error rate $= 1 - \text{TNR} = 1 - \text{Specificity} = \frac{\text{FP}}{\text{TN + FP}}$
- FNR $=$ Type-II error rate $= 1 - \text{TPR} = 1 - \text{Sensitivity} = 1 - \text{Recall} = \frac{\text{FN}}{\text{TP+FN}}$

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

Classification Terminology

- Overall error rate $= \frac{\#\text{errors}}{\#\text{data points}} = \frac{\text{FP + FN}}{\text{TN + FP + TP + FN}}$
- FPR $=$ Type-I error rate $= 1 - \text{TNR} = 1 - \text{Specificity} = \frac{\text{FP}}{\text{TN + FP}}$
- FNR $=$ Type-II error rate $= 1 - \text{TPR} = 1 - \text{Sensitivity} = 1 - \text{Recall} = \frac{\text{FN}}{\text{TP+FN}}$

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

What is the overall error rate in this example?

A. $\frac{228+40}{\#\text{data points}}$

B. $\frac{9627+105}{\#\text{data points}}$

Terminology

- Overall error rate $= \frac{\#\text{errors}}{\#\text{data points}} = \frac{\text{FP + FN}}{\text{TN + FP + TP + FN}}$
- FPR $=$ Type-I error rate $= 1 - \text{TNR} = 1 - \text{Specificity} = \frac{\text{FP}}{\text{TN + FP}}$
- FNR $=$ Type-II error rate $= 1 - \text{TPR} = 1 - \text{Sensitivity} = 1 - \text{Recall} = \frac{\text{FN}}{\text{TP+FN}}$

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

What is the FPR in this example?

A. $\frac{9627}{9627+40}$
B. $\frac{40}{9627+40}$
C. $\frac{228}{228+105}$
D. $\frac{105}{228+105}$

Terminology

- Overall error rate $= \frac{\#\text{errors}}{\#\text{data points}} = \frac{\text{FP + FN}}{\text{TN + FP + TP + FN}}$
- FPR $=$ Type-I error rate $= 1 - \text{TNR} = 1 - \text{Specificity} = \frac{\text{FP}}{\text{TN + FP}}$
- FNR $=$ Type-II error rate $= 1 - \text{TPR} = 1 - \text{Sensitivity} = 1 - \text{Recall} = \frac{\text{FN}}{\text{TP+FN}}$

```
> logistic.probs = predict(logistic.fit, type = "response")
>
> logistic.pred = rep("No", 10000)
> logistic.pred[ logistic.probs>0.5] = "Yes"
>
> table(logistic.pred, Default$default)

logistic.pred   No  Yes
          No  9627  228
          Yes   40  105
>
```

What is the FNR in this example?

A. $\frac{9627}{9627+40}$

B. $\frac{40}{9627+40}$

C. $\frac{228}{228+105}$

D. $\frac{105}{228+105}$

Terminology

CC companies do not want to extend credit to consumers who will ultimately default. They are less concerned with denying credit to those who will not default. What do they care more about?

A. FPR

B. FNR

Terminology

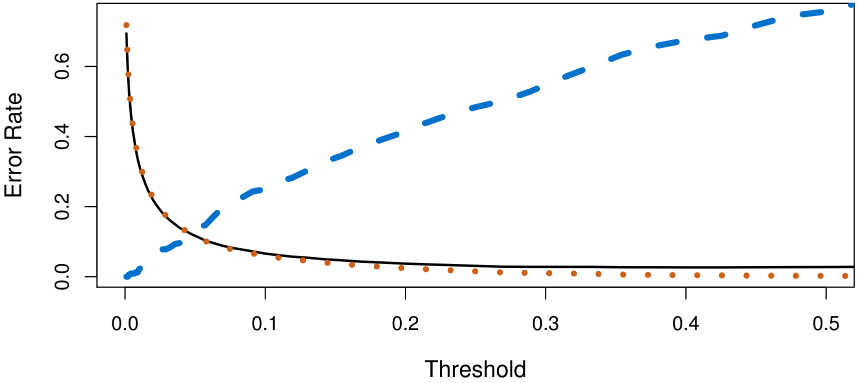
CC companies do not want to extend credit to consumers who will ultimately default. They are less concerned with denying credit to those who will not default. What do they care more about?

A. FPR

B. FNR

- They do not want to wrongly predict negative (no default).

Classification Error Rates: Tradeoffs

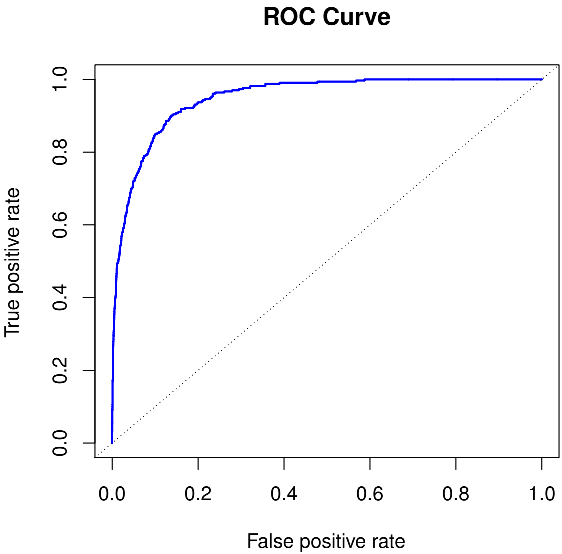


Black: overall error rate.
Blue dashed: false negative rate.
Orange: false positive rate.

Classification Terminology

- ROC curve: TPR vs. FPR (i.e., Sensitivity vs. 1-Specificity)
- Area under the Curve (AUC)
- More in ISLR Table 4.6 and 4.7

Classification Error Rates: Tradeoffs



Dealing with More than Two Classes (ISLR 9.4)

- Extending any two-class methods: One-Versus-One, One-Versus-All

Dealing with More than Two Classes (ISLR 9.4)

- Extending any two-class methods: One-Versus-One, One-Versus-All

- Extending logistic regression: multinomial logistic regression

Dealing with More than Two Classes (ISLR 9.4)

- Extending any two-class methods: One-Versus-One, One-Versus-All
- Extending logistic regression: multinomial logistic regression
- KNN, LDA and tree-based methods naturally handle multiple classes

Dealing with More than Two Classes (ISLR 9.4)

Extending any two-class methods to the K-class case

Dealing with More than Two Classes (ISLR 9.4)

Extending any two-class methods to the K-class case

One-Versus-One:

- For each pair of classes k and k', fit a LogReg for them
- There are $\binom{K}{2} = K(K-1)/2$ pairs and LogReg models
- For a test obs., classify it using each model $\Rightarrow$ $\binom{K}{2}$ predictions
- Final prediction = the most frequent prediction

Dealing with More than Two Classes (ISLR 9.4)

Extending any two-class methods to the K-class case

One-Versus-One:

- For each pair of classes k and k', fit a LogReg for them
- There are $\binom{K}{2} = K(K-1)/2$ pairs and LogReg models
- For a test obs., classify it using each model $\Rightarrow$ $\binom{K}{2}$ predictions
- Final prediction = the most frequent prediction

One-Versus-All:

- Each time pick a class k; combine all other $K-1$ classes as one; fit a LogReg $f_k(\cdot)$ for them
- There are K LogReg models $f_1, \ldots, f_K$
- For a test obs. $\vec{X}$, classify it using each model
  $\Rightarrow$ K scores $f_1(\vec{X}), \ldots, f_K(\vec{X})$
- Final prediction = the class k with the highest score $f_k(\vec{X})$

Dealing with More than Two Classes (ISLR 9.4)

Suppose you have K classes and that the number of training examples in each class is the same. Assuming you use logistic regression for all of your classifiers, which is more computationally intensive at prediction time?

A. one-versus-one

B. one-versus-all

Multinomial Logistic Regression (Optional)

- Also known as *multi-logit* model
- Suppose that there are K classes: $1, 2, 3, \ldots, K$

Multinomial Logistic Regression (Optional)

- Also known as *multi-logit* model
- Suppose that there are K classes: $1, 2, 3, \ldots, K$
- Multinomial logistic model:

$$P(Y = 1 \mid \vec{X}) = \frac{e^{\vec{X}^\top \vec{\beta}_1}}{1 + \sum_{k=1}^{K-1} e^{\vec{X}^\top \vec{\beta}_k}}$$

$$\vdots$$

$$P(Y = K - 1 \mid \vec{X}) = \frac{e^{\vec{X}^\top \vec{\beta}_{k-1}}}{1 + \sum_{k=1}^{K-1} e^{\vec{X}^\top \vec{\beta}_k}}$$

Multinomial Logistic Regression (Optional)

- Also known as *multi-logit* model
- Suppose that there are K classes: $1, 2, 3, \ldots, K$
- Multinomial logistic model:

$$P(Y = 1 \mid \vec{X}) = \frac{e^{\vec{X}^\top \vec{\beta}_1}}{1 + \sum_{k=1}^{K-1} e^{\vec{X}^\top \vec{\beta}_k}}$$

$$\vdots$$

$$P(Y = K - 1 \mid \vec{X}) = \frac{e^{\vec{X}^\top \vec{\beta}_{k-1}}}{1 + \sum_{k=1}^{K-1} e^{\vec{X}^\top \vec{\beta}_k}}$$

and (for the "reference class" K)

$$P(Y = K \mid \vec{X}) = \frac{1}{1 + \sum_{k=1}^{K-1} e^{\vec{X}^\top \vec{\beta}_k}}$$

Multinomial Logistic Regression (Optional)

- Also known as *multi-logit* model
- Suppose that there are K classes: $1, 2, 3, \ldots, K$
- Multinomial logistic model:

$$P(Y = 1 \mid \vec{X}) = \frac{e^{\vec{X}^\top \vec{\beta}_1}}{1 + \sum_{k=1}^{K-1} e^{\vec{X}^\top \vec{\beta}_k}}$$

$$\vdots$$

$$P(Y = K - 1 \mid \vec{X}) = \frac{e^{\vec{X}^\top \vec{\beta}_{k-1}}}{1 + \sum_{k=1}^{K-1} e^{\vec{X}^\top \vec{\beta}_k}}$$

and (for the "reference class" K)

$$P(Y = K \mid \vec{X}) = \frac{1}{1 + \sum_{k=1}^{K-1} e^{\vec{X}^\top \vec{\beta}_k}}$$

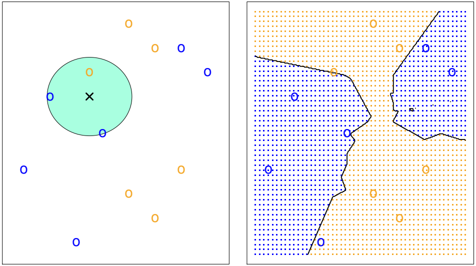
- Can estimate $\vec{\beta}_1, \vec{\beta}_2, \ldots, \vec{\beta}_{K-1}$ by MLE
- R command: `multinom`

K-Nearest Neighbors

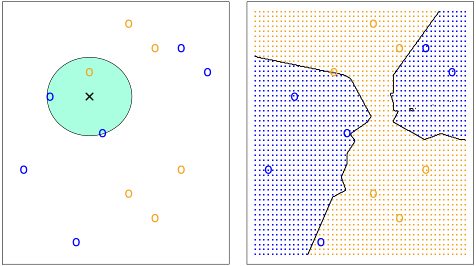
(ISLR Section 2.2.3 second part)

R command: `knn`

K-Nearest Neighbors



K-Nearest Neighbors

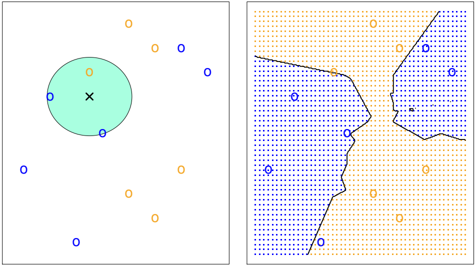


$$\Pr\left(Y = j|\vec{X} = \vec{x}\right) = \text{fraction of the } K \text{ nearest neighbors of } \vec{x} \text{ that are in class } j$$

$$= \frac{1}{K}\sum_{i\in\mathcal{N}} I(y_i = j)$$

$$\text{where } \mathcal{N} = \{\text{ the } K \text{ nearest neighbors of } \vec{x}\ \}$$

K-Nearest Neighbors



$$\Pr\left(Y = j | \vec{X} = \vec{x}\right) = \text{fraction of the } K \text{ nearest neighbors of } \vec{x} \text{ that are in class } j$$

$$= \frac{1}{K} \sum_{i \in \mathcal{N}} I(y_i = j)$$

where $\mathcal{N} = \{$ the K nearest neighbors of $\vec{x}$ $\}$

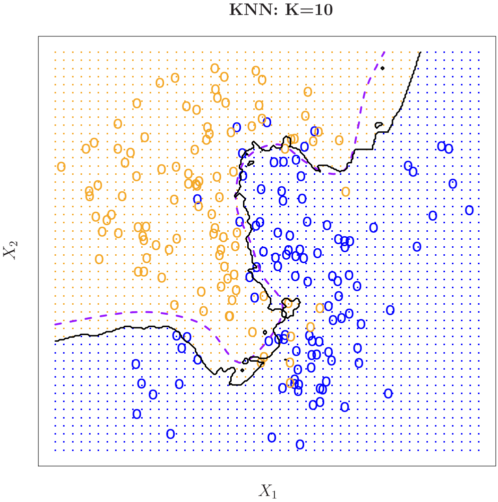
In the left figure, what is $P(Y = \text{orange} \mid X = x)$?

A. 2/3

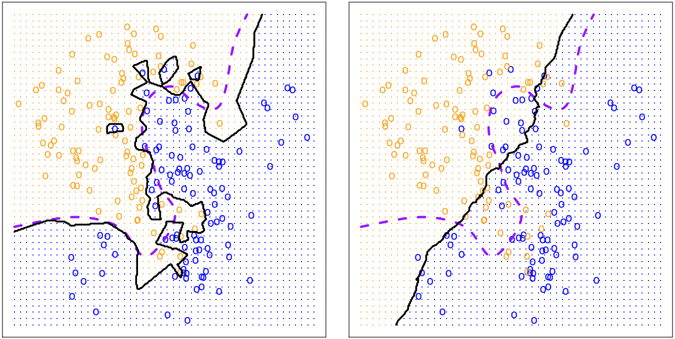
B. 1/3

C. 1/4

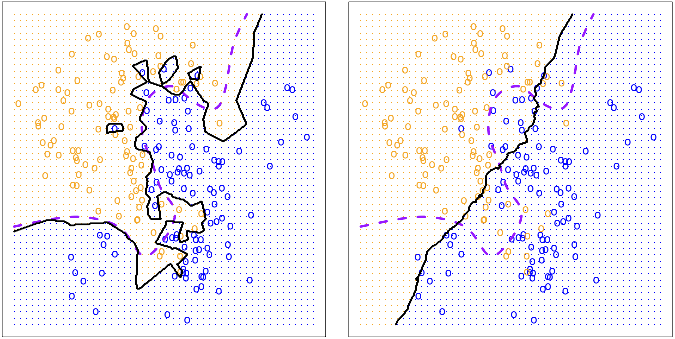
K-Nearest Neighbors



K-Nearest Neighbors



K-Nearest Neighbors



Which is true?

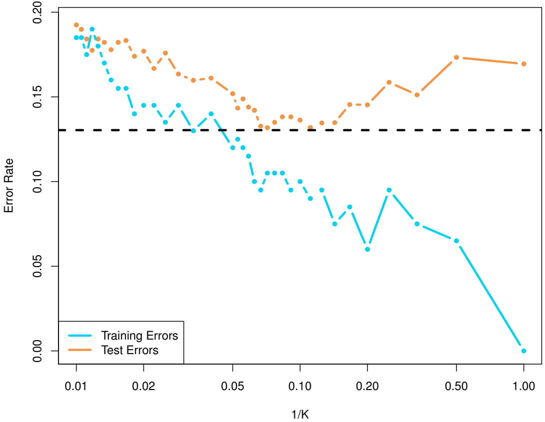
A. Left = ($K = 1$), Right = ($K = 100$)

B. Left = ($K = 100$), Right = ($K = 1$)

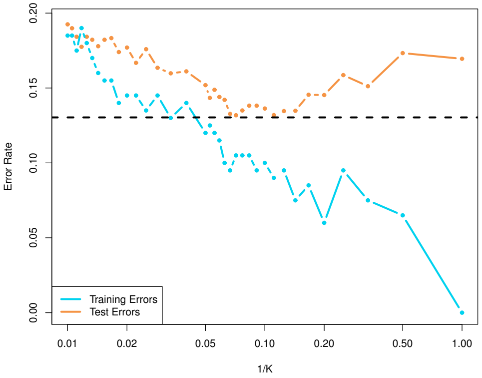
K-Nearest Neighbors

Effect of K:

- Small K: Very flexible. May overfit.
- Large K: Inflexible. Smooth boundary.



K-Nearest Neighbors



Choosing *K*:

- Option 1: Plug in your lucky number.
- Option 2: Cross-validation (next lecture)
- Hard in general. No universal solution.

Choosing Between Logistic Regression and KNN

Suppose we take a dataset and divide it into equally sized training and test sets. We then try out two different classification procedures, which achieve the following results.

1. **Logistic Regression** Achieved 20% training error, and 30% testing error.
2. **KNN with $K = 1$:** Achieved 18% average error rate, averaged over both training and testing set.

Based on these results, which method should we prefer?

A. Logistic Regression

B. KNN

Choosing Between Logistic Regression and KNN

Suppose we take a dataset and divide it into equally sized training and test sets. We then try out two different classification procedures, which achieve the following results.

1. **Logistic Regression** Achieved 20% training error, and 30% testing error.
2. **KNN with $K = 1$:** Achieved 18% average error rate, averaged over both training and testing set.

Based on these results, which method should we prefer?

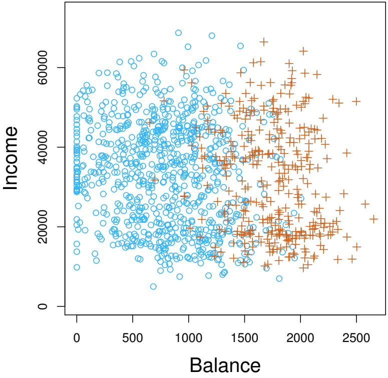
A. Logistic Regression

B. KNN

ISLR 4.7 (Exercise 8)

Classification Techniques: Summary

$$Y = f(X_1, X_2, \ldots, X_p) + \epsilon$$



- Logistic regression:
Simple. Inflexible. Linear boundary.

- KNN:
Simple. Flexibility determined by K. Arbitrary boundary.

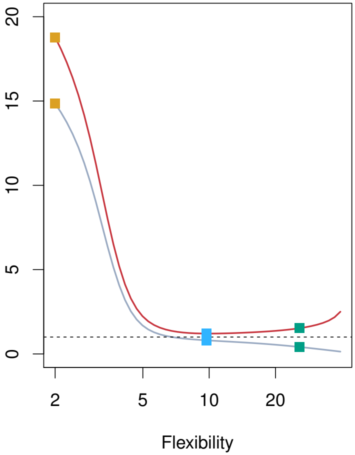
Later:

- SVM, tree-based methods:
Sophisticated. Flexibility chosen by users.

Classification Techniques: Summary

Recurring theme in statistics/data mining/machine learning:

Choose the right amount of flexibility for f.



Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R" (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani
Slides based on Yudong Chen's slides.