Clasificación

1EST17 - Aprendizaje Estadístico I

Enver G. Tarazona

2022-09-24

¿Que aprenderás?

- Clasificación
- Regresión logística
- Clasificador de Bayes, riesgo de Bayes
- Análisis Discriminante

¿Qué es clasificación?

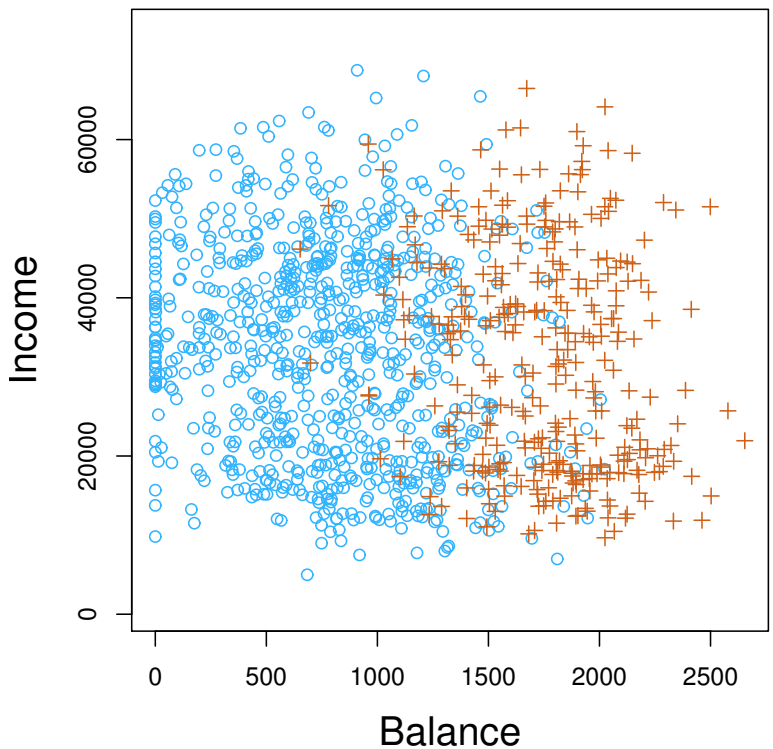
- Hasta ahora nuestras respuestas Y se han asumido como *cuantitativas*, mientras que es posible que las covariables fueran *categóricas*.
- Ahora permitimos que la respuesta sea *categórica*.
- Esto es incluso más común que las respuestas numéricas. Ejemplos:
  - Filtro de spam `email` $\in$ {spam, ham},
  - `Color de Ojos` $\in$ {azul, marrón, verde}.
  - `Condición médica` $\in$ {enfermedad1, enfermedad2, enfermedad3}.

- Supongamos que tenemos un valor de respuesta cualitativa que puede ser miembro de uno de K clases $\mathcal{C} = \{c_1, c_2, \ldots, c_K\}$.
- En clasificación construimos una función $f(X)$ que tome un vector de variables de entrada X y prediga la clase de pertenencia, tal que $Y \in \mathcal{C}$.
- También evaluaríamos la *incertidumbre* en esta clasificación. A veces, el papel de los diferentes predictores puede ser de interés principal.
- Usualmente construimos modelos que **predigan las probabilidades de categorías**, *dadas* ciertas covariables X.

Ejemplo: Datos de tarjetas de crédito

El conjunto de datos `Default` está disponible en la librería ISLR.

Objetivo : predecir si un individuo incumplirá con el pago de su tarjeta de crédito, dados los ingresos anuales y el saldo de la tarjeta de crédito.
Naranja: default=yes, Azul: default=no.



Configuración de la clasificación general

Configuración: Observaciones de entrenamiento $\{(x_1, y_1), ..., (x_n, y_n)\}$ donde la variable respuesta Y es categórica, p.ej. $Y \in \mathcal{C} = \{0, 1, ..., 9\}$ o $Y \in \mathcal{C} = \{perro, gato, ..., caballo\}$.

Objetivo: *Construir* un clasificador $f(X)$ que asigne una etiqueta de clase $\mathcal{C}$ a una observación futura sin etiquetar x y que evalúe la *incertidumbre* en esta clasificación.

Medidas de rendimiento: La más popular es la tasa de error de clasificación errónea (versión de entrenamiento y prueba).

Métodos Estadísticos Tradicionales para Clasificación

Tres métodos de clasificación han sido frecuentemente usados:

- Regresión Logística
- Análisis Discriminante Lineal (LDA)
- Análisis Discriminante Cuadrático (QDA)

¿Regresión lineal para una clasificación binaria?

Supongamos que tenemos una respuesta binaria, por ejemplo si un usuario de tarjeta de crédito incumple con su pago $Y =$ `si` or `no`, dadas las covariables X para predecir Y. Podríamos usar *códigos dummy* para Y como

$$Y = \begin{cases} 0 & \text{si } \texttt{no} \ , \\ 1 & \text{si } \texttt{si} \ . \end{cases}$$

¿Podemos simplemente realizar una regresión lineal de Y sobre X y clasificar como "sí" en caso $\hat{Y} > 0.5$?

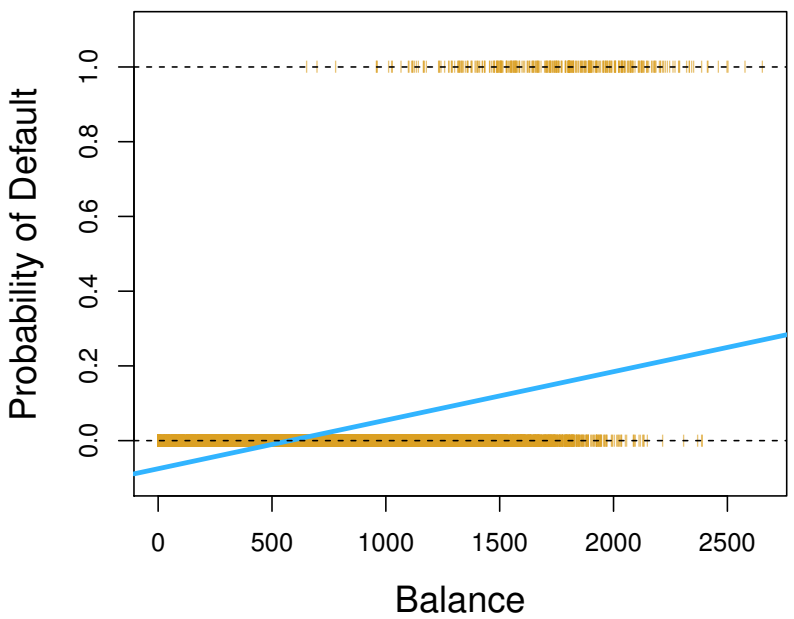
- En este caso de un resultado binario, la regresión lineal hace un buen trabajo como clasificador y es equivalente al análisis discriminante lineal.

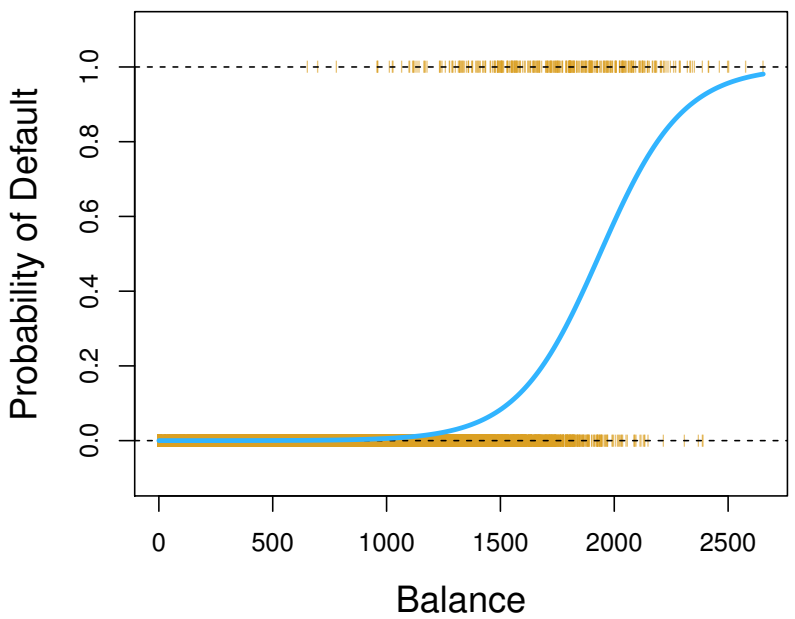
- Sin embargo, la regresión lineal puede producir probabilidades menores que cero o mayores que uno.

$\rightarrow$ Necesitamos usar la **regresión logística**.

Regresión lineal vs. logística

Anticipemos un poco, para ver por qué la regresión lineal no funciona tan bien. Estimamos la probabilidad de que alguien incumpla en su pago, dado el saldo de la tarjeta de crédito como predictor:





¿Regresión lineal para clasificación categórica?

¿Qué ocurre cuando hay más de dos resultados posibles? Por ejemplo, un diagnóstico médico Y, dado que los predictores X se pueden categorizar como

$$Y = \begin{cases} 1 & \text{si } \texttt{ataque fulminante}\,, \\ 2 & \text{si } \texttt{sobredosis de droga}\,, \\ 3 & \text{si } \texttt{ataque epiléptico}\,. \end{cases}$$

Esto sugiere un pedido, pero es artificial.

- La regresión lineal y logística no son apropiadas aquí.
- Necesitamos de la *regresión logística multiclase* y del *análisis discriminante*.

Sin embargo:

- Todavía es posible utilizar la regresión lineal para problemas de clasificación con dos clases. En realidad, ni siquiera es una mala idea y funciona bien en algunas condiciones. Bajo algunos supuestos estándar, esta regresión lineal (con 0 y 1 respuesta) de hecho dará la misma clasificación que el análisis discriminante lineal (LDA).

- Para resultados categóricos con más de dos niveles, se requiere algo de trabajo adicional (Y multivariante debido a la codificación de la variable ficticia).

- Dejamos la regresión lineal por ahora.

Regresión Logística

- En regresión logística consideramos un problema de clasificación con dos clases.
- Asumamos que Y está codificada ($\mathcal{C} = \{1, 0\}$ o {éxito, fracaso}), y nos enfocamos en el éxito ($Y = 1$).
- Podemos asumir que Y_i sigue una **distribución de Bernoulli** con probabilidad de éxito p_i.

$$Y_i = \begin{cases} 1 \text{ con probabilidad } p_i, \\ 0 \text{ con probabilidad } 1 - p_i. \end{cases}$$

- **Objetivo**: Para las covariables $(X_1, \dots, X_p)$, queremos estimar $p_i = \Pr(Y_i = 1 \mid X_1, \dots, X_p)$.

- Necesitamos una manera adecuada de *enlazar* nuestras covariables $X_1, \ldots, X_p$ con esta probabilidad p_i. Objetivo: queremos relacionar al *predictor lineal*

$$\eta_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip}$$

  con p_i. ¿Cómo?
- La idea es usar una llamada *función de enlace* para vincular p_i al predictor lineal.
- En regresión logística, usamos la *función de enlace logística*

$$\log\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip} \ . \qquad (1)$$

- ** P: ** ¿Cuál es la razón de ser de esto?

Regresión logística con una covariable

- La ecuación (1) puede ser reorganizada y expresada para p_i. Veamos esto solo para una covariable:

$$p_i = \frac{e^{\beta_0+\beta_1 x_i}}{1+e^{\beta_0+\beta_1 x_i}}.$$

- **Importante:** Los valores de p_i siempre caerán dentro del intervalo entre 0 y 1, con una curva en forma de S.

Ejemplo: Datos de tarjeta de crédito `Default`

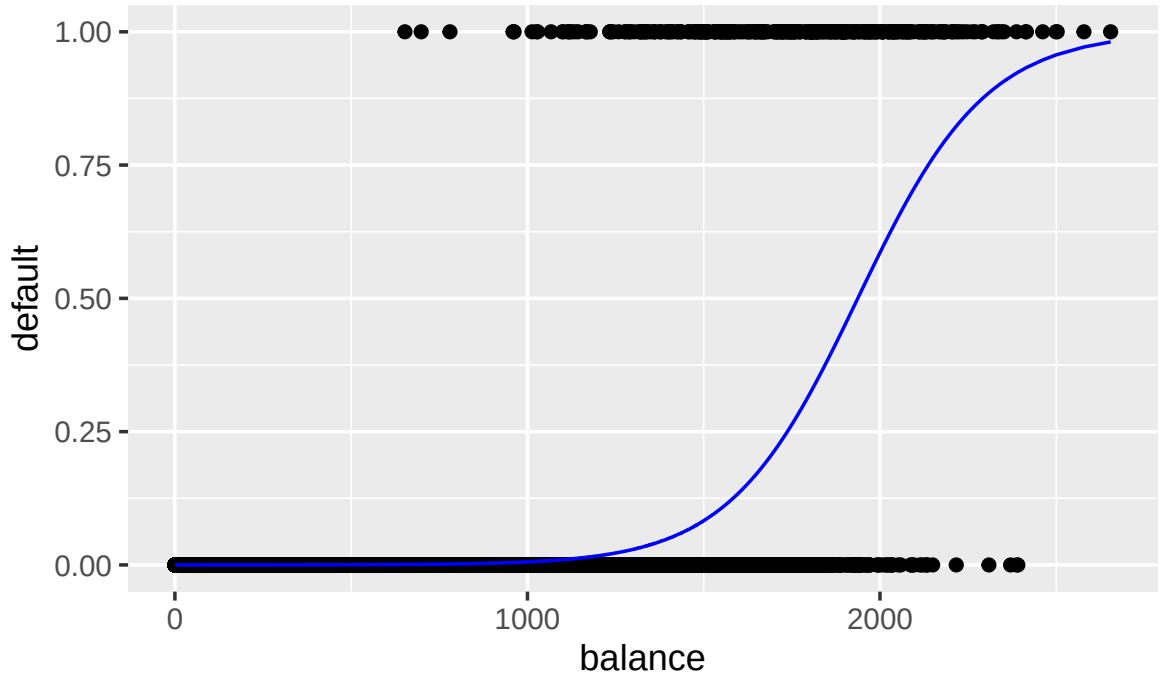
- Los parámetros se estiman a partir del método de máxima verosimilitud
- En R, esto se obtiene con la función `glm()`, donde especificamos `family="binomial"`.

```
library(ISLR)
data(Default)
Default$default <- as.numeric(Default$default) - 1
glm_default = glm(default ~ balance, data = Default, family = "binomial")

summary(glm_default)$coef
```

```
##                   Estimate   Std. Error   z value      Pr(>|z|)
## (Intercept) -10.651330614 0.3611573721 -29.49221 3.623124e-191
## balance       0.005498917 0.0002203702  24.95309 1.976602e-137
```

Trazando la línea ajustada (en azul):



Datos Default: Con $\hat{\beta}_0 =$ -10.65 y $\hat{\beta}_1 = 0.005$.

Estimando los coeficientes de regresión con MV

- Los coeficientes $\beta_0, \beta_1, \ldots$ son estimados con *Máxima Verosimilitud* (MV).

- Dados n pares de observaciones independientes $\{x_i, y_i\}$, la función de verosimilitud del modelo de regresión logística puede ser escrito como:

$$L(\beta) = \prod_{i=1}^{n} L_i(\beta) = \prod_{i=1}^{n} f(y_i; \beta) = \prod_{i=1}^{n} (p_i)^{y_i} (1 - p_i)^{1-y_i},$$

donde $\beta = (\beta_0, \beta_1, \beta_2, \ldots, \beta_p)^T$ se define dentro de p_i

$$p_i = \frac{\exp(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip})}{1 + \exp(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip})} .$$

- Las estimaciones de máxima verosimilitud se obtienen maximizando la verosimilitud.
- Para hacer las matemáticas más fáciles, usualmente trabajamos con la log-verosimilitud (el logaritmo es una transformación monótona, por lo que dará el mismo resultado que maximizar la probabilidad).

$$\begin{aligned}
\log(L(\beta)) = l(\beta) &= \sum_{i=1}^{n} \Big(y_i \log p_i + (1 - y_i) \log(1 - p_i) \Big) \\
&= \sum_{i=1}^{n} \left(y_i \log \left(\frac{p_i}{1 - p_i} \right) + \log(1 - p_i) \right) \\
&= \sum_{i=1}^{n} \Big(y_i (\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}) - \log(1 + e^{\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}}) \Big).
\end{aligned}$$

- Para maximizar la función logarítmica de verosimilitud, encontramos las derivadas parciales $p+1$ y las establecemos en 0.
- Esto nos da un conjunto de $p+1$ ecuaciones no lineales para los βs.
- Este conjunto de ecuaciones no tiene una solución de forma cerrada.
- Por tanto, el sistema de ecuaciones se resuelve numéricamente utilizando el *algoritmo de Newton-Raphson* (o Fisher Scoring).

Interpretación cualitativa de los coeficientes

Veamos nuevamente el resultado de la regresión:

```
summary(glm_default)$coef
```

```
##                   Estimate   Std. Error   z value      Pr(>|z|)
## (Intercept) -10.651330614 0.3611573721 -29.49221 3.623124e-191
## balance       0.005498917 0.0002203702  24.95309 1.976602e-137
```

- La estadística z es igual a $\frac{\hat{\beta}}{SE(\hat{\beta})}$, y está distribuida aproximadamnte como $N(0,1)$.[1]
- El p-valor es $\Pr(|Z| > |z|)$ para una variable aleatoria $Z \sim N(0,1)$
- Verifique el p-valor para `Balance`. ¿Conclusión?

[1] Con este conocimiento podemos construir intervalos de confianza y probar hipótesis sobre los βs, con el objetivo de comprender qué covariables contribuyen a nuestras probabilidades posteriores y clasificación.

Interpretación cuantitativa de los coeficientes

Recordar que de la ecuación (1) tenemos

$$\log\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} \ ,$$

entonces

$$\frac{p_i}{1-p_i} = e^{\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}+...+\beta_p x_{ip}} = e^{\eta_i} \ .$$

La cantidad $p_i/(1-p_i)$ es llamada los *odds (ventajas)*. Odds representan *posibilidades* (p.ej. en apuestas).

P: Responder:

1. Crees que tu equipo de fútbol ganará esta noche con una probabilidad $p = 0.8$. ¿Cuáles son los odds de que gane?
2. Las ventajas del mejor caballo en una carrera son $9:1$. ¿Cuál es la probabilidad de que gane este caballo?

¿Por qué son relevantes los odds?

Reorganicemos nuevamente los *odds* en el modelo de regresión logística:

$$\frac{p_i}{1-p_i} = \frac{\mathrm{P}(Y_i = 1 \mid X = x)}{\mathrm{P}(Y_i = 0 \mid X = x)}$$

$$= \exp(\beta_0) \cdot \exp(\beta_1 x_{i1}) \cdot \ldots \cdot \exp(\beta_p x_{ip}) \ .$$

$\rightarrow$ Tenemos un *modelo multiplicativo* para los odds - que puede ayudarnos a interpretar nuestros βs.

La razón de odds

Para comprender el efecto de un coeficiente de regresión β_j, veamos qué sucede si aumentamos x_{ij} hasta $x_{ij} + 1$, mientras que todas las demás covariables se mantienen fijas.

Usando álgebra simple y la fórmula de la diapositiva anterior, se verá que

$$\frac{\text{odds}(Y_i = 1 \mid X_j = x_{ij} + 1)}{\text{odds}(Y_i = 1 \mid X_j = x_{ij})} = \exp(\beta_j) \ . \tag{2}$$

Interpretación:
Al aumentar la covariable x_{ij} en una unidad, cambiamos las probabilidades de $Y_i = 1$ por un factor $\exp(\beta_j)$.

Es más:
Tomando logaritmos a la ecuación (2), se deduce que β_j se puede interpretar como **logaritmo de razón de odds** .

Ajustemos ahora el modelo de regresión logística para `default`, dado `balance`, `income` y la variable binaria `student` como predictores:

```
glm_default2 = glm(default ~ balance + income + student, data = Default,
    family = "binomial")

summary(glm_default2)$coef
```

```
##                  Estimate   Std. Error    z value      Pr(>|z|)
## (Intercept) -1.086905e+01 4.922555e-01 -22.080088 4.911280e-108
## balance      5.736505e-03 2.318945e-04  24.737563 4.219578e-135
## income       3.033450e-06 8.202615e-06   0.369815  7.115203e-01
## studentYes  -6.467758e-01 2.362525e-01  -2.737646  6.188063e-03
```

Preguntas:

- ¿Qué sucede con los odds de incumplimiento cuando `income` aumenta en 10 000 dólares?
- ¿Qué sucede con los odds de incumplimiento cuando `balance` aumenta en 100 dólares?

Predicciones

Vamos a ponernos al día. ¿Por qué estábamos haciendo todo esto en primer lugar?

Respuesta: Queremos construir un modelo que **prediga las probabilidades de las categorías** de Y, *dadas* ciertas covariables $X_1, \ldots, X_p$.

- Para estimaciones de parámetros dados $\hat{\beta}_0, \hat{\beta}_1, \ldots \hat{\beta}_p$ y una nueva observación x_0, podemos estimar la probabilidad $\hat{p}(x_0)$ de que la nueva observación pertenezca a la clase definida por $Y = 1$

$$\hat{p}(x_0) = \frac{e^{\hat{\eta}_0}}{1 + e^{\hat{\eta}_0}} \ ,$$

con el predictor lineal

$$\hat{\eta}_0 = \hat{\beta}_0 + \hat{\beta}_1 x_{01} + \ldots + \hat{\beta}_p x_{0p} \ .$$

En el caso de las covariables cualitativas, es necesario introducir variables ficticias. Esto se puede hacer como en el caso de la regresión lineal.

Entonces, en el ejemplo de `Default`, podemos predecir la probabilidad de que alguien no cumpla con su pago.

Por ejemplo: "¿Cuál es la probabilidad estimada de que un estudiante incumpla con su pago si tiene un saldo de 2000 y un ingreso de 40000?"

$$\hat{p}(X) = \frac{e^{\beta_0+2000\cdot\beta_1+40000\cdot\beta_2+1\cdot\beta_3}}{1+e^{\beta_0+2000\cdot\beta_1+40000\cdot\beta_2+1\cdot\beta_3}} = 0.5196$$

Usando R:

```
eta <- summary(glm_default2)$coef[, 1] %*% c(1, 2000, 40000, 1)
exp(eta)/(1 + exp(eta))
```

```
##              [,1]
## [1,] 0.5196218
```

(o via la función `predict()` en R.)

El clasificador de Bayes

- Teníamos como objetivo realizar una clasificación, pero terminamos estimando $\hat{p}(X) = \Pr(Y \mid X)$. Idea: la probabilidad puede ser usada para clasificación.
- Asumamos que conocemos o podemos estimar la probabilidad de que una nueva observación x_0 pertenezca a la clase k, para K clases $\mathcal{C} = \{c_1, c_2, \ldots, c_K\}$, con elementos numerados como $1, 2, ..., K$

$$p_k(x_0) = \Pr(Y = k | X = x_0), \quad k = 1, 2, ...K \ .$$

  Esto es la probabilidad de que $Y = k$ dada la observación x_0.
- El *clasificador de Bayes asigna una observación a la clase más probable*, dado los valores de los predictores.
- **Ejemplo** para dos grupos $\{A, B\}$. Una nueva observación x_0 será clasificada como A si $\Pr(Y = A | X = x_0) > 0.5$ y a la clase B en caso contrario.

Propiedades del clasificador de Bayes

- Tiene la *tasa de error de prueba más pequeña.*
- Los límites de clase que utilizan el clasificador de Bayes se denominan *límites de decisión de Bayes.*
- La tasa de error general de Bayes se da como

$$1 - \mathrm{E}(\max_j \Pr(Y = j \mid X))$$

  donde la esperanza es sobre X.

- La tasa de error de Bayes es comparable al *error irreducible* en la configuración de regresión.
- **Advertencia:** por lo general, no conocemos la verdadera distribución condicional $\Pr(Y|X)$ para datos reales

Algo de terminología

Conjunto de entrenamiento : Observaciones independientes $\{(x_1, y_1), ..., (x_n, y_n)\}$ con variable *respuesta cualitativa* $Y \in \{1, 2, ..., K\}$, usadas para construir la regla de clasificación (mediante la estimación de parámetros en densidades de clases o probabilidades posteriores).

Conjunto de prueba: Observaciones independientes del mismo formato que el conjunto de entrenamiento, utilizadas para evaluar la regla de clasificación.

Función de pérdida: A las clasificaciones erróneas se les asigna la pérdida 1 y las clasificaciones correctas pérdida 0 - esta es conocida como *pérdida-0/1*.

Error de entrenamiento

- **Tasa de error de entrenamiento**: La proporción de errores que se cometen si aplicamos nuestro estimador $\hat{f}$ a las observaciones de entrenamiento, p.ej. $\hat{y}_i = \hat{f}(x_i)$

$$\frac{1}{n}\sum_{i=1}^{n} \mathrm{I}(y_i \neq \hat{y}_i) \ ,$$

con función indicadora I, la cuál está definida como:

$$\mathrm{I}(a \neq \hat{a}) = \begin{cases} 1 \text{ si } a \neq \hat{a} \ , \\ 0 \text{ caso contrario.} \end{cases}$$

- La tasa de error de entrenamiento es la fracción de clasificaciones erróneas realizadas en nuestro conjunto de entrenamiento.

- Una tasa de error de entrenamiento muy baja puede implicar un sobreajuste.

Error de prueba

- **Tasa de error de prueba**: La fracción de clasificaciones erróneas cuando nuestro modelo se aplica en un conjunto de prueba

$$\text{Ave}(I(y_0 \neq \hat{y}_0)) \ ,$$

  donde el promedio es sobre todas las observaciones de prueba (x_0, y_0).

- Nuevamente, esto da una mejor indicación del verdadero desempeño del clasificador que el error de entrenamiento (¿recuerdas por qué?).

- Suponemos que un *buen* clasificador es un clasificador que tiene un *error de prueba bajo*.

Regla de decisión de Bayes: dos paradigmas

Dos enfoques para estimar $\Pr(Y = k \mid X = x)$:

Paradigma de diagnóstico

Esto es lo que hicimos hasta ahora: El enfoque ha sido estimar *directamente* la distribución a posteriori para las clases

$$\Pr(Y = k \mid X = x) \ .$$

Ejemplos: Regresión Logística. Clasificación KNN.

Paradigma de muestreo

- Enfoque indirecto: Modele la distribución condicional de predictores $f_k(x) = \Pr(X = x \mid Y = k)$ para cada clase y las probabilidades a priori $\pi_k = \Pr(Y = k)$.
- Luego clasifique en la clase con el producto máximo $\pi_k f_k(x)$. En este paradigma, necesitamos modelar la función de distribución para cada clase.

Dada una X continua y Y categórica, y

- la función de *densidad* de probabilidad $f_k(x) = \Pr(X = x \mid Y = k)$ para X en la clase k.
- la probabilidad a *priori* para la clase k $\pi_k = \Pr(Y = k)$ es la probabilidad a priori.

¿Cómo obtenemos $\Pr(Y = k|X = x_0)$? Esto es, ¿cómo podemos "voltear" la condicional?

Teorema de Bayes

$$p_k(X) = \Pr(Y = k \mid X = x) = \frac{\Pr(X = x \cap Y = k)}{f(x)}$$
$$= \frac{f_k(x)\pi_k}{\sum_{l=1}^{K} f_l(x)\pi_l} \ . \qquad (3)$$

Análisis Discriminante

Definición

- El análisis discriminante se basa en el *paradigma de muestreo*.
- El enfoque es modelar la distribución de X en cada una de las clases por separado, y luego usar el teorema de Bayes para invertir las cosas y obtener $\Pr(Y \mid X)$.
- Cuando usamos distribuciones normales para cada clase, esto conduce a un análisis discriminante lineal o cuadrático (pero pueden usarse otras distribuciones.)

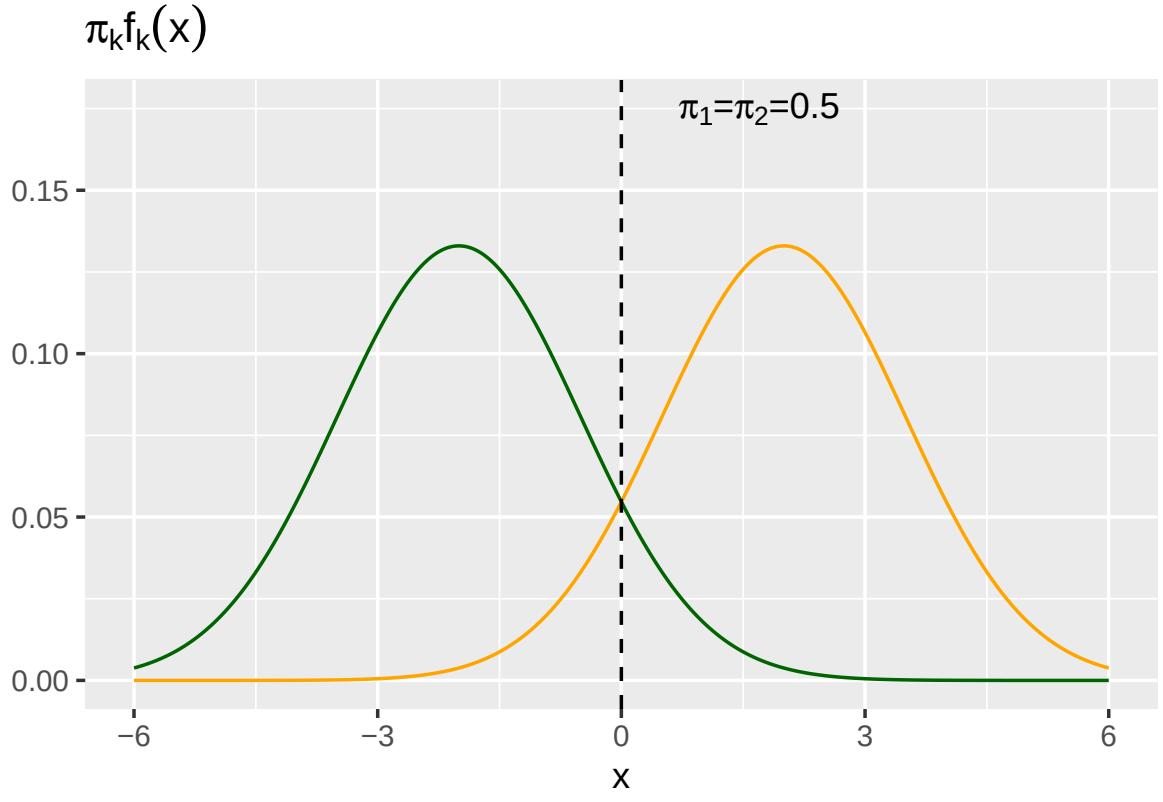
Ejemplo

Supongamos que tenemos observaciones que provienen de dos clases: {verde, naranja}

$$X_{\text{verde}} \sim \mathcal{N}(-2, 1.5^2) \text{ y } X_{\text{naranja}} \sim \mathcal{N}(2, 1.5^2)$$

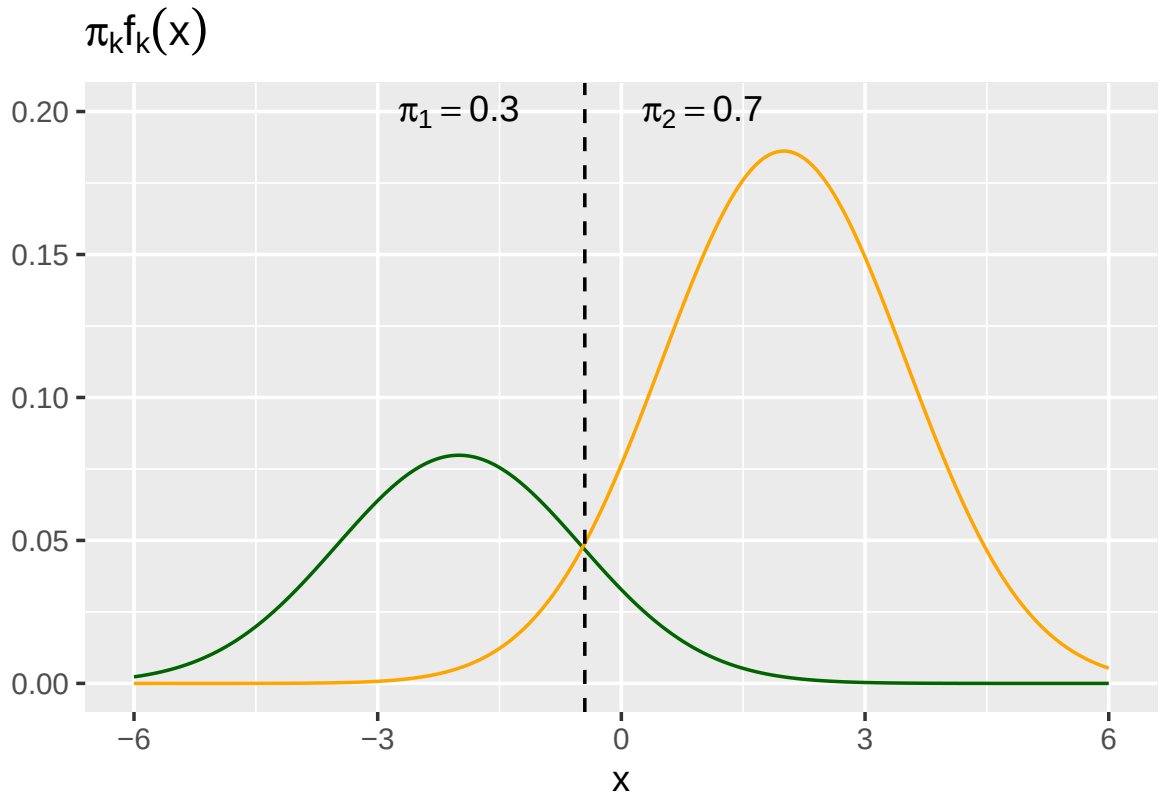
Suponga que las probabilidades son iguales, $\pi_1 = \pi_2 = 0.5$.

Podemos graficar $\pi_k f_k(x)$ para las dos clases:



El límite de la frontera es donde está el punto de intersección de las dos líneas, porque aquí $\pi_1 f_1(x) = \pi_2 f_2(x)$.

Para diferentes prioris $\pi_1 = 0.3$ y $\pi_2 = 0.7$, la frontera de decisión se desplaza hacia la izquierda:



¿Por qué usar el análisis discriminante?

- El análisis discriminante lineal es más estable que la regresión logística cuando
  - las clases están bien separadas. En ese caso, las estimaciones de los parámetros para el modelo de regresión logística son muy inestables.
  - n es pequeño y la distribución de los predictores X es aproximadamente normal en cada una de las clases.

- Además, el análisis discriminante lineal es popular cuando tenemos más de dos clases de respuesta.

Análisis Discriminante Lineal (LDA) cuando $p = 1$

- Las distribuciones condicionales a las clases $f_k(X)$ son asumidas como normales para $k = 1, \ldots, K$, esto es

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma_k}\right)^2}$$

  con parámetros μ_k (media) y σ_k (desviación estándar).
- Con LDA asumimos que todas las clases tienen *la misma desviación estándar* $\sigma_k = \sigma$.
- Adicionalmente, tenemos las probabilidades a priori para las clases $\pi_k = \Pr(Y = k)$, tal que $\sum_{k=1}^{K} \pi_k = 1$.

Podemos insertar la expresión anterior para cada distribución de clases en la fórmula de Bayes para obtener la probabilidad a posteriori $p_k(x) = \Pr(Y = k|X = x)$

$$p_k(x) = \frac{f_k(x)\pi_k}{f(x)} = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma}\right)^2}}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_l}{\sigma}\right)^2}} \ .$$

Nuestra regla es clasificar a la clase para la que $p_k(x)$ es más grande.

Tomando logaritmos y descartando términos que no dependen de k, vemos que esto equivale a asignar x a la clase con la mayor *puntuación discriminante* $\delta_k(x)$:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k).$$

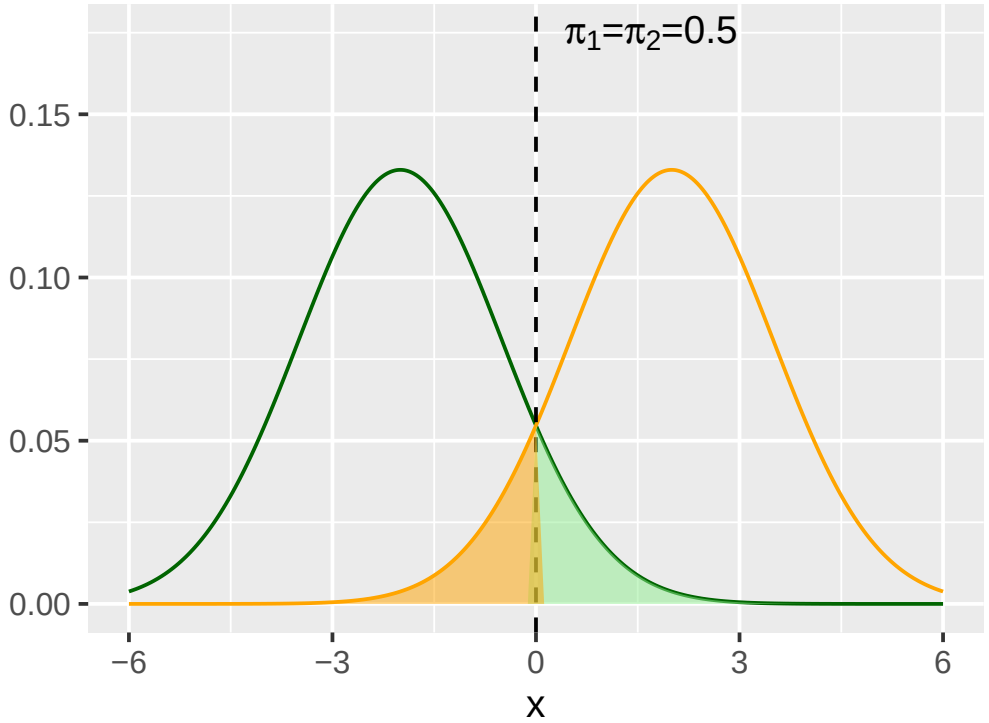
- Estas fronteras de decisión entre clases son *lineales* en x.
- Para $K = 2$ clases y $\pi_1 = \pi_2$, la frontera de decisión se encuentra en:

$$x = \frac{\mu_1 + \mu_2}{2} \ .$$

(Demostrarlo haciendo $\delta_1(x) = \delta_2(x)$ y resolviendo para x.)

Regresando a nuestro ejemplo

$$X_{\text{verde}} \sim \mathcal{N}(-2, 1.5^2) \text{ y} X_{\text{naranja}} \sim \mathcal{N}(2, 1.5^2)$$



- La frontera de decisión de bayes se encuentra en $x = 0$.
- La tasa de error de Bayes: `round(pnorm(0,2,1.5))`=0.09.
- El clasificador de Bayes tiene la tasa de error de prueba más baja.

En el ejemplo anterior, conocíamos las verdaderas distribuciones $p_k(X)$ y las prioris π_k. Pero normalmente no conocemos estos parámetros, solo tenemos los datos de entrenamiento.

Idea: simplemente estimamos los parámetros y los conectamos a la regla.

Estimadores de Parámetros

- La probabilidad a priori para la clase k es (usualmente) estimada tomando la proporción de observaciones n_k (sobre n) que vienen de la clase k: $\hat{\pi}_k = \frac{n_k}{n}$.
- El valor de la media para la clase k es simplemente la media muestral de todas las observaciones que pertenecen a la clase k:

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i.$$

- La desviación estándar: desviación estándar de la muestra en todas las clases (desviación estándar "ponderada"):

$$\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2 = \sum_{k=1}^{K} \frac{n_k - 1}{n-K} \cdot \hat{\sigma}_k^2.$$

$\hat{\sigma}_k$: desviación estándar estimada de todas las observaciones de la clase k.

¿Cómo probar la bondad del estimador?

1. Utilice el conjunto de entrenamiento para estimar parámetros y límites de clase.
2. Utilice el conjunto de prueba para estimar la tasa de clasificación errónea.

Ejemplo simulado:

```
n = 1000
pi1 = pi2 = 0.5
mu1 = -2
mu2 = 2
sigma = 1.5
set.seed(1)
n1train = rbinom(1, n, pi1)
n2train = n - n1train
n1test = rbinom(1, n, pi1)
n2test = n - n1test
train1 = rnorm(n1train, mu1, sigma)
train2 = rnorm(n2train, mu2, sigma)
test1 = rnorm(n1test, mu1, sigma)
test2 = rnorm(n2test, mu2, sigma)
var2.1 = var(train1)
var2.2 = var(train2)
var.pool = ((n1train - 1) * var2.1 + (n2train - 1) * var2.2)/(n - 2)
```

Entonces se establece

$$\hat{\delta}_1(x) = \hat{\delta}_2(x)$$

y se resuelve para x para obtener una regla de decisión (frontera).

Ejercicio: Verifique que el siguiente código le proporcione las tasas de error de entrenamiento y prueba:

```
rule = 0.5 * (mean(train1) + mean(train2)) + var.pool * (log(n2train/n) -
    log(n1train/n))/(mean(train1) - mean(train2))

trainingError <- (sum(train1 > rule) + sum(train2 < rule))/n
testError <- (sum(test1 > rule) + sum(test2 < rule))/n

c(trainingError, testError)
```

```
## [1] 0.105 0.115
```

Este es un rendimiento bastante bueno, en comparación con la tasa de error mínima de Bayes. Pero tenga en cuenta que el clasificador LDA se basa en el supuesto de normalidad, y que se asume $\sigma_k = \sigma$ para todas las clases. [2].

[2] Ambos de los cuales sabíamos que se cumplían aquí.

La matriz de confusión

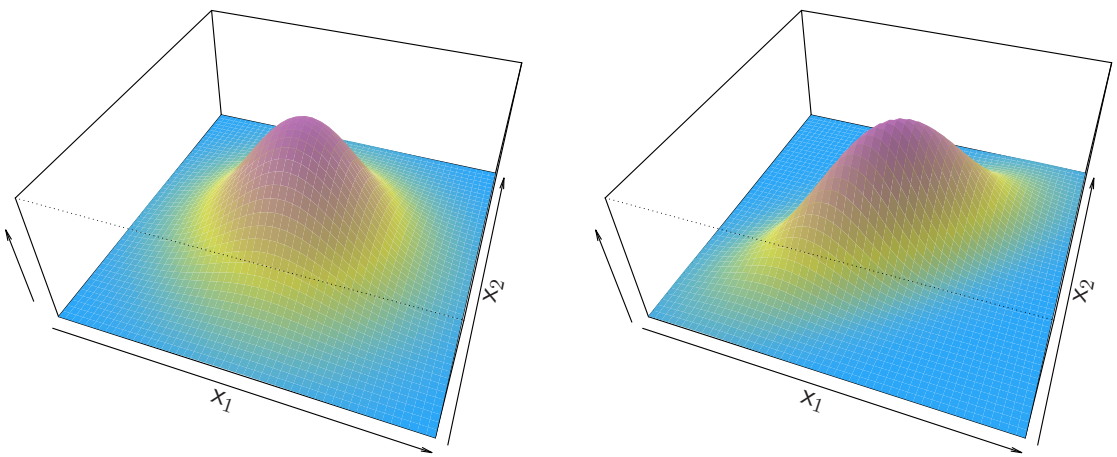
- La matriz de confusión es una tabla que puede mostrar el desempeño de un clasificador, dado que se conocen los valores verdaderos.
- Podemos hacer una matriz de confusión a partir del conjunto de entrenamiento o prueba.
- La suma de la diagonal es el número total de clasificaciones correctas. La suma de todos los elementos fuera de la diagonal es el número total de clasificaciones erróneas.

- La matriz de confusión se puede obtener en `R` usando la función `table`, o directamente usando el paquete `caret`.

LDA Multivariado ($p > 1$)

- LDA se puede generalizar a situaciones en las que se utilizan covariables $p > 1$. Los límites de decisión siguen siendo lineales.
- La función de distribución normal multivariante:

$$f(x) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu))$$



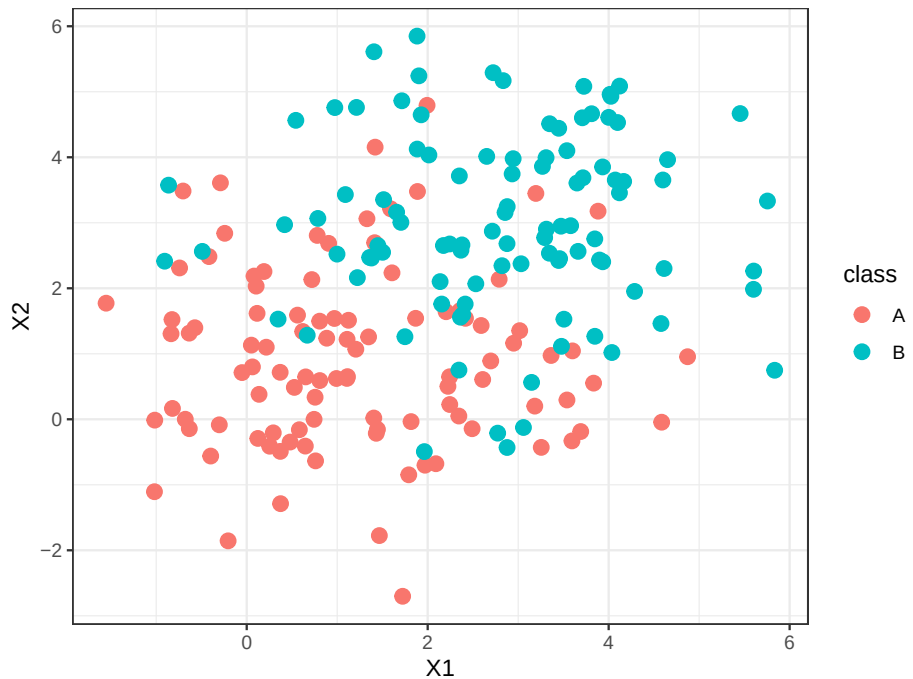
- Al colocar esta densidad en la ecuación (3) se obtiene la siguiente expresión para la función discriminante:

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

- Nota: $\delta_k(x) = c_{k0} + c_{k1} x_1 + ... + c_{kp} x_p$ es una función lineal en $(x_1, ..., x_p)$.

Regresando al ejemplo sintético

- Considere nuevamente nuestra simulación a partir de una distribución normal bivariada con vectores de medias $\mu_A = (1, 1)^T$, $\mu_B = (3, 3)^T$, y matriz de covarianzas $\Sigma_A = \Sigma_B = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$.
- Objetivo: Utilizar LDA para clasificar una nueva observación x_0 a la clase A o B.



- Dado que *aquí se conoce la verdad*, podemos calcular el límite de Bayes y el error de Bayes.
- Dado que tenemos distribuciones de clases normales bivariadas con una matriz de covarianza común, el límite óptimo está dado por LDA, con límite dado en $\delta_A(x) = \delta_B(x)$.

$$x^T\Sigma^{-1}\mu_A - \frac{1}{2}\mu_A^T\Sigma^{-1}\mu_A + \log\pi_A = x^T\Sigma^{-1}\mu_B - \frac{1}{2}\mu_B^T\Sigma^{-1}\mu_B + \log\pi_B$$

$$x^T\Sigma^{-1}(\mu_A - \mu_B) - \frac{1}{2}\mu_A^T\Sigma^{-1}\mu_A + \frac{1}{2}\mu_B^T\Sigma^{-1}\mu_B + \log\pi_A - \log\pi_B = 0$$

Ingresando los valores numéricos se obtiene $-x_1 - x_2 + 4 = 0$, por lo tanto, un límite con forma funcional

$$x_2 = 4 - x_1 \ .$$

Matriz de confusión para el ejemplo sintético

Podemos usar las fronteras de Bayes para encontrar la tasa de error:

```
r.pred <- ifelse(df$X2 < 4 - df$X1, "A", "B")
table(real = df$class, r.pred)
```

```
##     r.pred
## real  A  B
##    A 82 18
##    B 21 79
```

Por supuesto, el límite de Bayes generalmente no se conoce y debemos estimarlo a partir de los datos.

Estimadores para p>1:

- Probabilidad a priori para la clase k (simiar a $p = 1$): $\hat{\pi}_k = \frac{n_k}{n}$.
- El valor de la media para la clase k es simplemente la media muestral para todas las observaciones de la clase k (pero ahora son vectores):

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} \mathbf{X}_i.$$

- La matriz de covarianza para cada clase:

$$\hat{\Sigma}_k = \frac{1}{n_k - 1} \sum_{i:y_i=k} (\mathbf{X}_i - \hat{\mu}_k)(\mathbf{X}_i - \hat{\mu}_k)^T$$

- Versión ponderada:

$$\hat{\Sigma} = \sum_{k=1}^{K} \frac{n_k - 1}{n - K} \cdot \hat{\Sigma}_k.$$

Analizando el ejemplo sintético con `lda()`

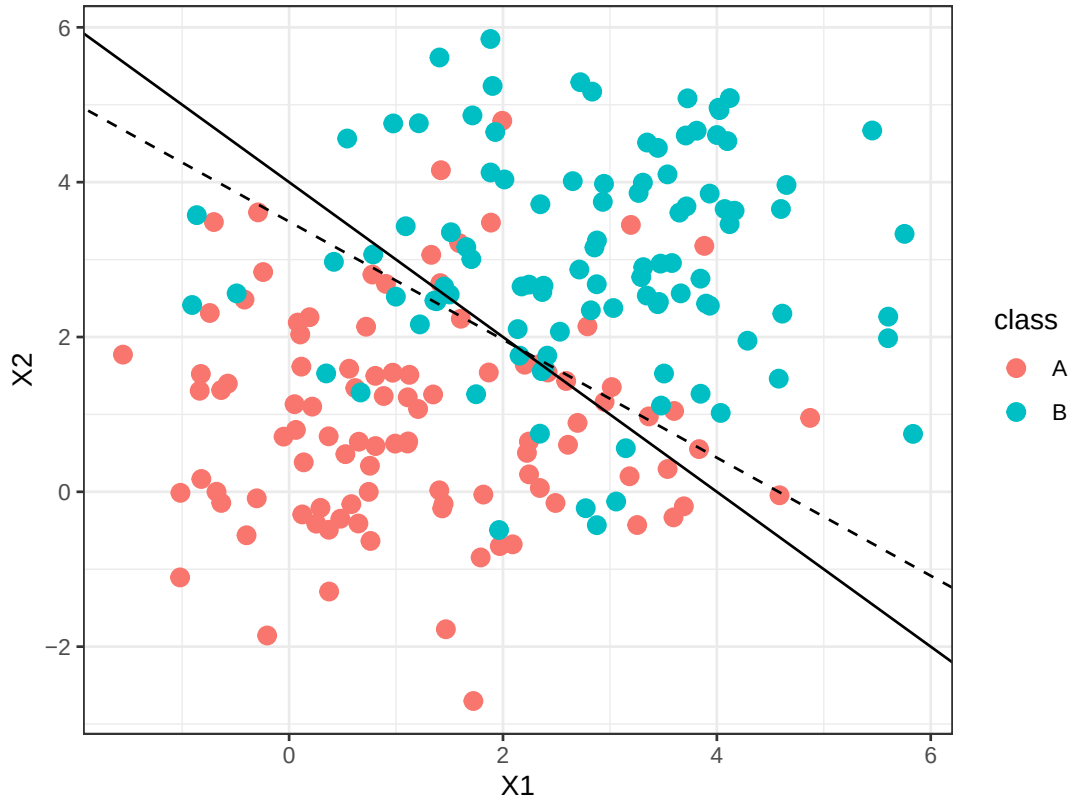
```
r.lda <- lda(class ~ X1 + X2, df)
r.pred <- predict(r.lda, df)$class
table(real = df$class, predicted = r.pred)
```

```
##     predicted
## real  A  B
##    A 87 13
##    B 18 82
```

Nota: El error de entrenamiento es menor que para el límite de Bayes. ¿Por qué?

Comparación de Bayes con el límite estimado

Línea continua: límite de Bayes
Línea discontinua: límite estimado



Probabilidades a posteri

- A veces, la probabilidad de que una observación provenga de una clase k es más interesante que la clasificación real en sí.

- Estas probabilidades de clase se pueden estimar a partir de las distribuciones condicionales de clase y a priori, o de las funciones discriminantes:

$$\begin{aligned}\hat{P}(Y=k|X=x) &= \frac{\hat{\pi}_k \cdot \frac{1}{(2\pi)^{p/2}|\hat{\Sigma}|^{1/2}} \exp(-\frac{1}{2}(x-\hat{\mu}_k)^T \hat{\Sigma}^{-1}(x-\hat{\mu}_k))}{\sum_{l=1}^{K} \hat{\pi}_l \frac{1}{(2\pi)^{p/2}|\hat{\Sigma}|^{1/2}} \exp(-\frac{1}{2}(x-\hat{\mu}_l)^T \hat{\Sigma}^{-1}(x-\hat{\mu}_l))} \\ &= \frac{e^{\hat{\delta}_k(x)}}{\sum_{l=1}^{K} e^{\hat{\delta}_l(x)}}.\end{aligned}$$

Análisis discriminante cuadrático (QDA)

- En LDA nosotros asumimos que $\Sigma_k = \Sigma$ para todas las clases.
- En QDA permitimos diferentes matrices de covarianza Σ_k para cada clase, mientras que los predictores siguen siendo normales multivariantes

$$X \sim N(\mu_k, \Sigma_k) \ .$$

- Las funciones discriminantes ahora vienen dadas por:

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

- Estos límites de decisión son funciones *cuadráticas* de x.

LDA vs QDA

QDA es más flexible que LDA, ya que permite matrices de covarianza específicas de grupo.

P:

- Pero, si las matrices de covarianza en teoría son iguales, ¿no se estimarán iguales?
- ¿No deberíamos preferir siempre QDA a LDA?

R:

Explicación similar a una "compensación de sesgo-varianza":

- Si la suposición de matrices de covarianza iguales es incorrecta, entonces LDA puede sufrir un alto sesgo para los estimadores de parámetros.
- Pero para tamaños de muestra pequeños, las matrices de covarianza pueden estar mal estimadas (alta varianza de los estimadores).

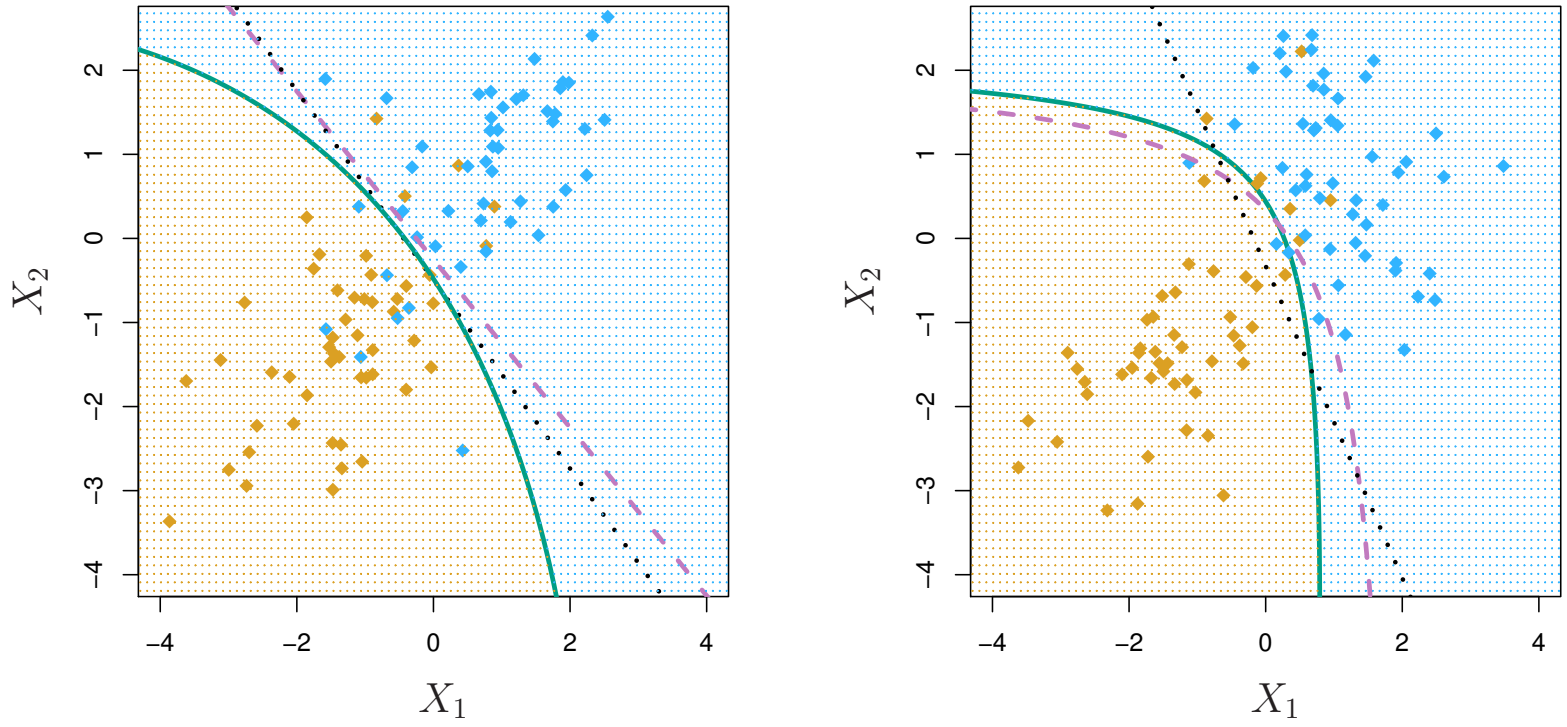
Si el número de covariables es alto:

- entonces QDA requiere estimar $K \cdot p \cdot (p+1)/2$ parámetros,
- mientras que LDA solo requiere $p \cdot (p+1)/2$.

Por lo tanto, LDA es menos flexible que QDA y, por lo tanto, podría tener mucha menos varianza.

LDA vs QDA – Ilustración

Límites de decisión de Bayes (punteado púrpura), LDA (punteado negro) y QDA (verde sólido) para los casos en los que $\Sigma_1 = \Sigma_2$ (izquierda) y $\Sigma_1 \neq \Sigma_2$ (derecha).



Ejemplo: ¿Qué tipo de especie de iris?

El conjunto de datos de flores de `iris` fue introducido por el estadístico y biólogo británico Ronald Fisher en 1936.

- **Tres especies de plantas**: {setosa, virginica, versicolor}.
- **Cuatro atributos**: `Sepal.Length`, `Sepal.Width`, `Petal.Length` y `Petal.Width`.

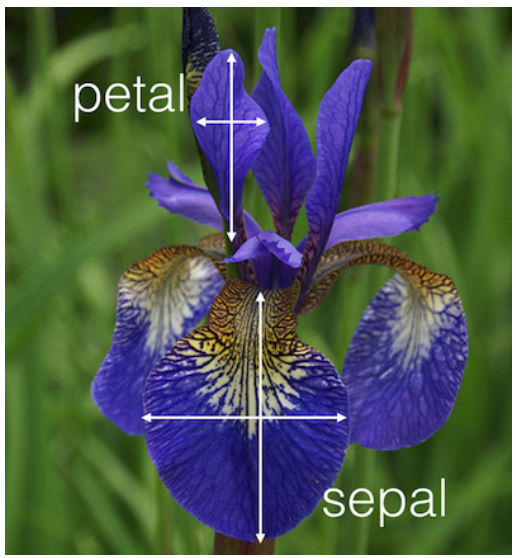


Figure 1: Planta de iris con sépalos y hojas de pétalos.

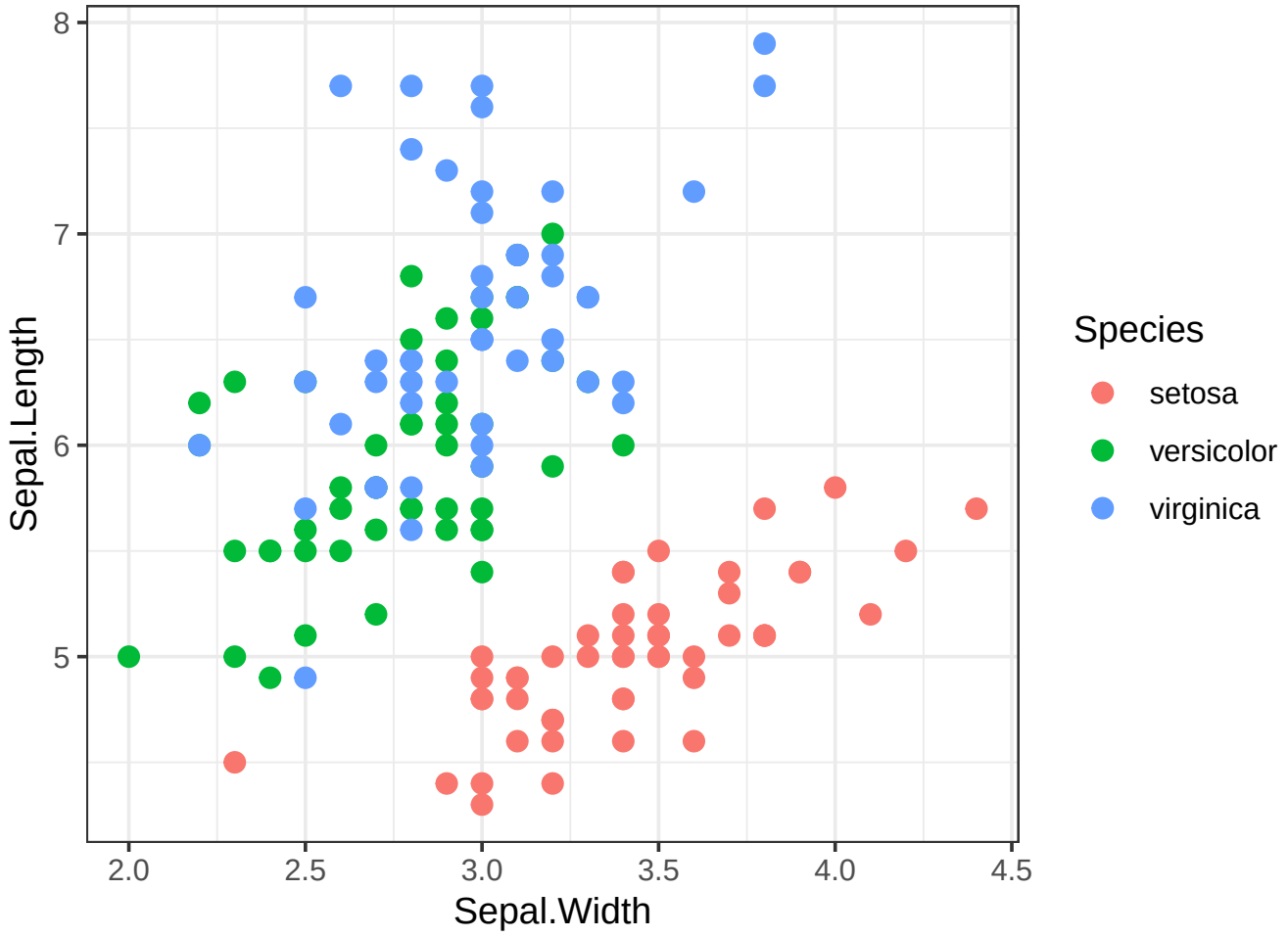
http://blog.kaggle.com/2015/04/22/scikit-learn-video-3-machine-learning-first-steps-with-the-iris-dataset/

Ejemplo: clasificación de plantas de iris

Usaremos `sepal width` y `sépal length` para construir un clasificador. Tenemos 50 observaciones de cada clase.

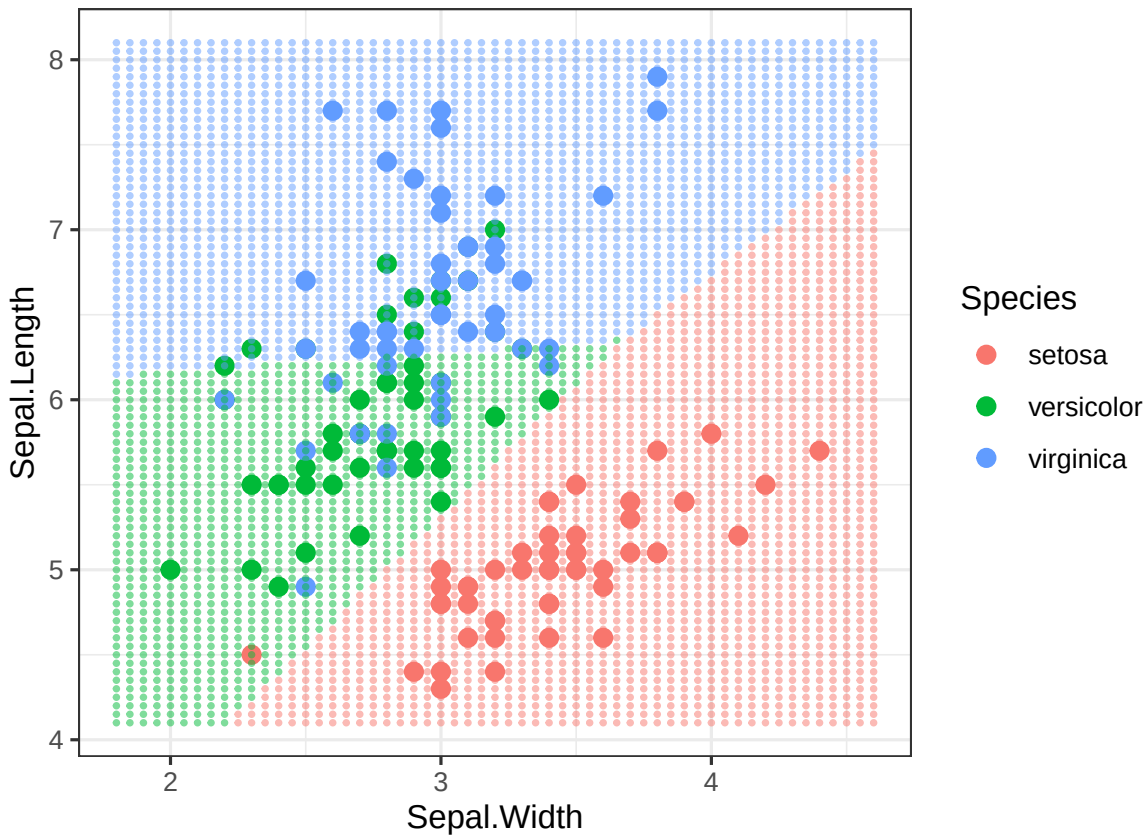
```
attach(iris)
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```


Sepal.Length
8
7
6
5
2.0
2.5
3.0
3.5
4.0
4.5
Sepal.Width
Species
setosa
versicolor
virginica

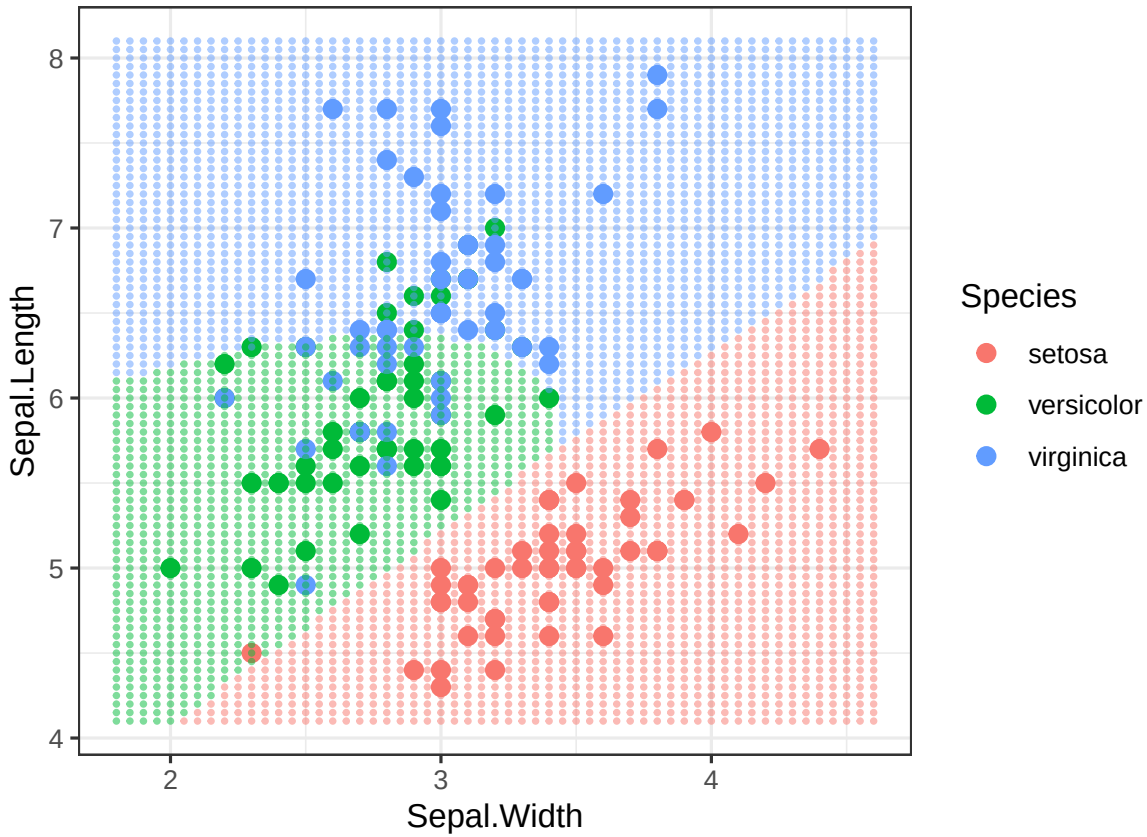
Iris: LDA

```
iris_lda = lda(Species ~ Sepal.Length + Sepal.Width, data = iris, prior = c(1,
    1, 1)/3)
```



Iris: QDA

```
iris_qda = qda(Species ~ Sepal.Length + Sepal.Width, data = iris, prior = c(1,
    1, 1)/3)
```



Iris: Comparación de LDA y QDA

Para comparar el *rendimiento predictivo* de nuestros dos clasificadores, dividimos el conjunto de datos `iris` original al azar en muestras de entrenamiento y prueba de igual tamaño:

```
set.seed(1)
train = sample(1:150, 75)

iris_train = iris[train, ]
iris_test = iris[-train, ]
```

Ejecutar LDA y QDA *en el mismo conjunto de datos de entrenamiento*:

```
iris_lda2 = lda(Species ~ Sepal.Length + Sepal.Width, data = iris_train,
    prior = c(1, 1, 1)/3)

iris_qda2 = qda(Species ~ Sepal.Length + Sepal.Width, data = iris_train,
    prior = c(1, 1, 1)/3)
```

LDA error de entrenamiento: $\frac{14}{75} = 0.19$

```
table(predict(iris_lda2, newdata = iris_train)$class, iris_train$Species)
```

```
##
##              setosa versicolor virginica
##   setosa         27          0         0
##   versicolor      1         15         8
##   virginica       0          5        19
```

LDA error de prueba: $\frac{19}{75} = 0.26$.

```
iris_lda2_predict = predict(iris_lda2, newdata = iris_test)
table(iris_lda2_predict$class, iris$Species[-train])
```

```
##
##              setosa versicolor virginica
##   setosa         22          0         0
##   versicolor      0         22        11
##   virginica       0          8        12
```

QDA error de entrenamiento: $\frac{13}{75} = 0.17$.

```
table(predict(iris_qda2, newdata = iris_train)$class, iris_train$Species)
```

```
##
##              setosa versicolor virginica
##   setosa         28          0         0
##   versicolor      0         16         9
##   virginica       0          4        18
```

QDA error de prueba: $\frac{24}{75} = 0.32$.

```
iris_qda2_predict = predict(iris_qda2, newdata = iris_test)
table(iris_qda2_predict$class, iris$Species[-train])
```

```
##
##              setosa versicolor virginica
##   setosa         22          0         0
##   versicolor      0         18        12
##   virginica       0         12        11
```

Resultado: El clasificador LDA ha dado el error de prueba más pequeño [3] para clasificar las plantas de iris según el ancho y la longitud del sépalo para nuestro conjunto de prueba y debería ser preferible en este caso.

Pero:

1. ¿Otra división de los datos en entrenamiento y conjunto de prueba daría la misma conclusión (que LDA es mejor que QDA para este conjunto de datos)? R: No necesariamente, pero probablemente. → Examinaremos estas preguntas en el siguiente capítulo (validación cruzada).

2. ¿Qué pasa con las otras dos covariables? ¿Agregarlos al modelo (4 covariables) proporcionaría una mejor regla de clasificación? R: Probablemente. Pruébelo si lo desea.

[3] Tenga en cuenta que el error de entrenamiento es de mucho menos interés; podría ser bajo debido a *sobreajuste* solamente.

Diferentes formas de análisis discriminante

- LDA
- QDA
- RDA: Análisis Discriminante Regularizaddo
- Naive Bayes ("Bayes Ingenuo"): Suponga que la densidad de cada clase es el producto de las densidades marginales, es decir, las entradas son condicionalmente independientes en cada clase

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

  Esto generalmente no es cierto, pero simplifica drásticamente la estimación.

- Otras formas proponiendo modelos de densidad específicos para $f_k(x)$, incluyendo enfoques no paramétricos.

Análisis Discriminante Regularizado (RDA)

- Friedman (1989) propuso una alternativa mixta entre LDA y QDA
  - Considera un estimador robusto para las matrices de covarianza.
  - Reducir las variancias separadas de QDA hacia una covarianza común en LDA.
  - Un método más general que tiene como casos particulares: LDA y QDA.
- Consideremos ahora 2 clases C_1 y C_2 y que siguen un distribución normal multivariada

$$x \mid C_1 \sim N_p(\mu_1, \Sigma_1)$$
$$x \mid C_2 \sim N_p(\mu_2, \Sigma_2)$$

donde Σ_1 y Σ_2 son estimadas en forma penalizada.

Las matrices de covarianzas regularizadas son

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

donde $\hat{\Sigma}$ es la matriz de covarianzas muestrales conjunta a lo largo de todas las clases y $\hat{\Sigma}_k$ la matriz de covarianzas para la clase k

- $\alpha \in [0, 1]$, es un parámetro de afinamiento que representa la continuidad de modelos entre un LDA y QDA. ¿Qué sucede si α está cerca de 1? ¿Y si está cerca de 0?
- En la práctica, elegir α con los datos de validación o usando CV.

RDA: Elección del parámetro alpha

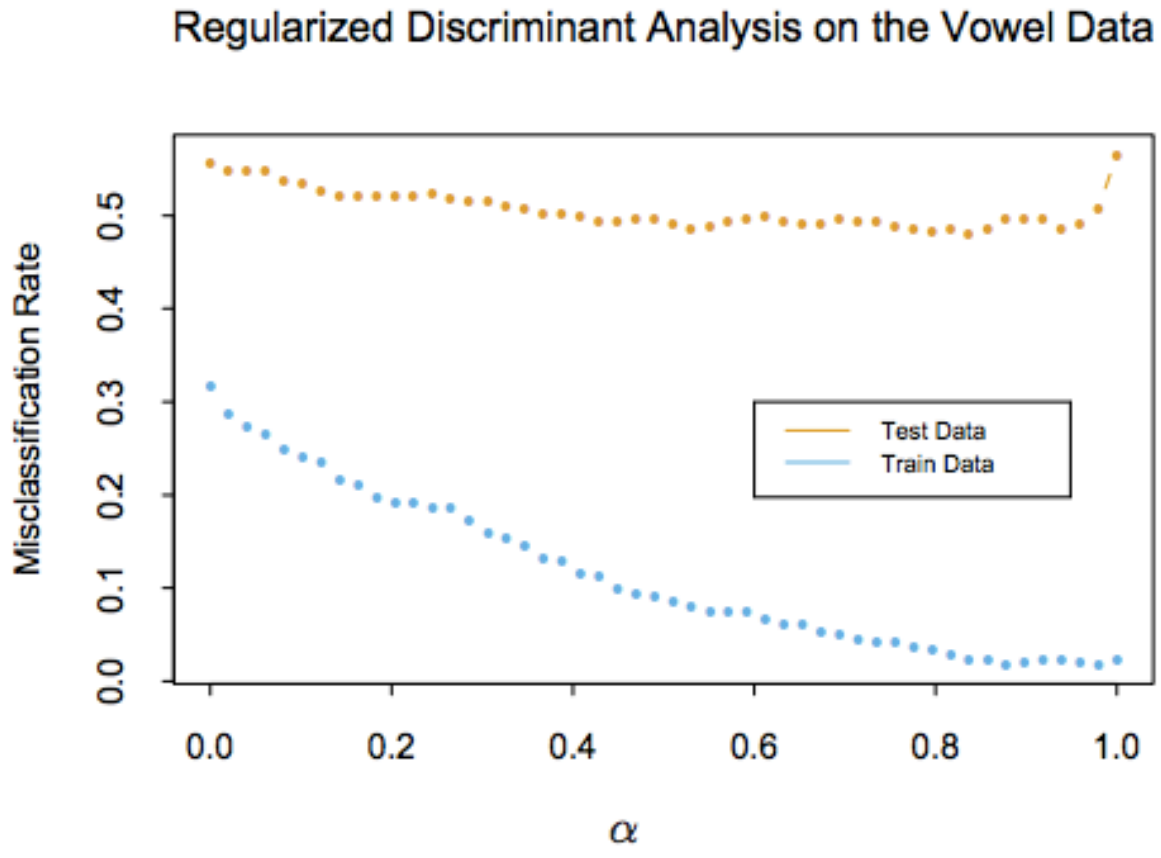


FIGURE 4.7. *Test and training errors for the vowel data, using regularized discriminant analysis with a series of values of* $\alpha \in [0, 1]$. *The optimum for the test data occurs around* $\alpha = 0.9$, *close to quadratic discriminant analysis.*

RDA: Otras Generalizaciones

- Para LDA, podemos transformar la matriz de covarianzas usando el escalar $\gamma \in [0, 1]$

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

- Una familia generalizada para QDA es obtenida con (α, γ)

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

- Notemos que
  - Si $\gamma = 0$ y $\alpha = 1 \Rightarrow$ QDA
  - Si $\gamma = 0$ y $\alpha = 0 \Rightarrow$ LDA
- Estos parámetros son estimados considerando como función objetivo la tasa de mala clasificación.

RDA en R

Esta metodología se encuentra implementada en la función `rda` de la librería `klaR`.

```
rda(formula, data, prior)

formula:  grupos ~ x1 + x2 + ?
data: conjunto de datos
prior: probabilidades a prior, por default son
proporcionales al tamaño del grupo.
```

Análisis Discriminante Lineal Generalizado

Mayor flexibilidad, fronteras de decisión más complejas

- FDA: Análisis Discriminante flexible
  - Permite fronteras no lineales: Reformula el problema LDA como un problema de regresión lineal y luego utiliza expansiones base para hacer la discriminación. Revisar Hastie et al. (1994)
- PDA: Análisis Discriminante Penalizado
  - Selección de variables: Penaliza los coeficientes para ser suavizados o ser más dispersos facilitando la interpretación.
- MDA: Análisis Discriminante Mixto
  - Permite que cada clase sea una mixtura de dos o màs distribuciones normales con diferentes centroides, pero cada componente (dentro y entre las clases) comparten la misma matriz de covarianzas. Revisar Clemmensen et al. (2011).

Naive Bayes

- Naive Bayes es un método popular cuando p es grande.

- El *Bayes ingenuo original*: distribuciones marginales normales univariadas. Como consecuencia

$$\delta_k(x) \propto \log\left[\pi_k \prod_{j=1}^{p} f_{kj}(x_j)\right] = -\frac{1}{2}\sum_{j=1}^{p} \frac{(x_j - \mu_{kj})^2}{\sigma_{kj}^2} + \log(\pi_k) \ ,$$

  por lo tanto, se supone que Σ_k es diagonal y solo se estiman los elementos de la diagonal.

- Se pueden hacer generalizaciones arbitrarias. Por ejemplo, características mixtas (predictores cualitativos y cuantitativos).

- Este método a menudo produce buenos resultados, aunque el pdf conjunto no es el producto del pdf marginal. Esto podría deberse a que no nos estamos enfocando en la estimación de los pdf de clase, sino en los límites de clase.

Naive Bayes: Aspectos Generales

- Los clasificadores Bayesianos buscan responder a la pregunta: "basados en los valores observados de los predictores, ¿Cuál es la probabilidad de que el resultado sea la clase C_K?' '
- Se encuentra basado en el teorema de Bayes-Price.
- Estudios comparativos de diversos algoritmos de clasificación han encontrado que Naïve Bayes es comparable en performance a los árboles de decisión y las redes neuronales.
- Suelen tener alta precisión y rapidez cuando son aplicados a grandes bases de datos.
- Una asunción importante es que el efecto del valor de un atributo para una clase dada es independiente de los valores del resto de los atributos (Independencia Condicional dentro de clases).
- El supuesto anterior simplifica los cálculos, por eso recibe el nombre de *naïve*.

Construcción de un clasificador bayesiano

- Asumir que se quiere predecir la variable Y que asume K valores distintos y que estos valores son $1, \ldots, K$.
- Asumir que hay m variables predictoras $X = (X_1, \ldots, X_m)$
- Dividir el conjunto de datos en K subconjunto de datos llamados $DS_1, DS_2, \ldots, DS_K$
- Definir DS_k =Registros en los cuales $Y = k$
- Para cada grupo DS_k, usamos estimacion de densidad para estimar el modelo M_k que modela la distribucion de las variables de entrada entre los registros $Y = k$.
- M_K estima la función de probilidad conjunta por clase $\Pr[X|Y = k]$

Cálculo de la Probabilidad a Posteriori

Sea Y la variable de clasificación que puede tomar los valores $k = 1, \ldots, K$ y X la colección de variables predictoras. La probabilidad de pertenecer a una clase k para un conjunto de valores de X (evidencia) estará dada por la Regla de Bayes-Price:

$$
\begin{aligned}
\Pr[Y = k|X] = & \frac{\Pr[Y = k]\Pr[X|Y = k]}{\Pr[X]} \\
& \frac{\Pr[Y = k]\Pr[X_1 = x_1 \ldots X_m = x_m|Y = k]}{\Pr[X_1 = x_1 \ldots X_m = x_m]} \\
& \frac{\Pr[Y = k]\Pr[X_1 = x_1 \ldots X_m = x_m|Y = k]}{\sum_{j=1}^{K}\Pr[X_1 = x_1 \ldots X_m = x_m, Y = k]} \\
& \frac{\Pr[Y = k]\Pr[X_1 = x_1 \ldots X_m = x_m|Y = k]}{\sum_{j=1}^{K}\Pr[Y = j]\Pr[X_1 = x_1 \ldots X_m = x_m|Y = j]}
\end{aligned}
$$

- $\Pr[Y = k|X]$ es la **probabilidad a posteriori**. Por ejemplo supongamos que tenemos las variables predictoras referidas a clientes descritos por los atributos *edad* e *ingreso*. Y de manera específica, un cliente tiene 35 años de edad con un ingreso de \$40000. Supongamos que k es la hipótesis de que el cliente comprará una computadora. Entonces se desea calcular la probabilidad de que el cliente compre una computadora conociendo su edad e ingreso.
- $\Pr[Y = k]$ es la probabilidad a priori del resultado. Esencialmente, basado en lo conocemos acerca del problema, cuánto esperaríamos que sea la probabilidad de pertenecer a una clase. Para el ejemplo anterior, sería la probabilidad de que un cliente compre una computadora independientemente de su edad, ingreso o cualquier otra información.

- $\Pr[X]$ es la probabilidad de los valores de los predictores. En otras palabras, si una nueva muestra será predicha, ¿Qué tan probable es este patrón en comparación con el resto de datos de entrenamiento? Formalmente, esta probabilidad es calculada usando una distribución de probabilidad multivariada. En la práctica, se realizan asunciones para reducir la complejidad de este cálculo. Para nuestro ejemplo, será la probabilidad de que un cliente tenga 35 años y gane \$40000
- $\Pr[X|Y = k]$ Es la probabilidad condicional. Para los datos asociados con la clase k, ¿Cuál es la probabilidad de observar los valores para las variables predictoras? En el ejemplo, cuál es la probabilidad de que un cliente tenga 35 años y gane \$40000 si sabemos que comprará una computadora.

Estimación de un clasificador bayesiano

1. Estimar la distribucion de las predictoras en cada clase. Es decir, estimar $\Pr[X_1 = x_1 \dots X_m = x_m | Y = k]$. Opciones

- Estimador de densidad conjunta (kernel, k-nn)
- Estimador de densidad Naïve

2. Estimar $\Pr[Y = k]$ como la fraccion de registros en la cual $Y = k$
3. Para una nueva predicción:

$$
\begin{aligned}
Y^{predict} = & \arg\max_y \Pr[Y = k | X_1 = x_1 \dots X_m = x_m] \\
& \arg\max_y \Pr[X_1 = x_1 \dots X_m = x_m | Y = k] \Pr[Y = k]
\end{aligned}
$$

Estimador Naive Bayes

El modelo Naïve Bayes simplifica el cálculo asumiendo que los predictores son independientes en cada una de las clases:

$$\Pr\left[X_1 = x_1 \dots X_m = x_m | Y = k\right] = \Pr\left[X_1 = x_1 | Y = k\right] \dots \Pr\left[X_m = x_m | Y = k\right]$$

Luego,

$$Y^{predict} = \arg\max_y \Pr\left[X_1 = x_1 \dots X_m = x_m | Y = k\right] \Pr\left[Y = k\right]$$

Se convierte en:

$$Y^{predict} = \arg\max_y \Pr\left[Y = k\right] \prod_{j=1}^{m} \Pr\left[X_j = x_j | Y = k\right]$$

Si hay muchos atributos de entrada este producto puede producir underflow, asi que es mejor usar logaritmos.

$$Y^{predict} = \underset{y}{\arg\max} \left(\log \Pr\left[Y = k\right] + \sum_{j=1}^{m} \log \Pr\left[X_j = x_j | Y = k\right] \right)$$

Cálculo de Probabilidades Condicionales

- Si X_j es discreta, la estimación sería la frecuencia relativa dentro de cada clase.
- El clasificador Naïve Bayes puede ser aplicado tambien cuando hay predictoras continuas. Alternativas:
  1. Aplicar previamente un metodo de discretizacion tal como: intervalos de igual ancho, intervalos con igual frecuencia, ChiMerge,1R, Discretizacion usando el metodo de la entropia.
  2. Estimación no paramétrica de la densidad del kernel (Hardle et al. 2004).

- 3. Asumiendo una distribucion para cada predictora, por lo general Gausiana, con media y varianza estimada de los datos. La libreria `e1071` de R contiene una función `naiveBayes` que calcula el clasificador naïve Bayes. En este caso:

$$\Pr\left[X_j = x_j | Y = k\right] = \frac{1}{s_j\sqrt{2\pi}} \exp\left[\frac{-\left(x_i - \bar{x}_j\right)^2}{2s_{j^2}}\right]$$

Donde $\bar{x}_j$ y s_j son la media y la varianza de los valores de la variable X_j en la clase k.

Predicción

Para predecir la clase a la cual pertenece X, $\Pr[Y = k] \Pr[X|Y = k]$ es evaluado para cada clase k. El clasificador predecirá que los valores de X pertenecen a la clase i si y solo si:

$$\Pr[Y = i] \Pr[X|Y = i] > \Pr[Y = j] \Pr[X|Y = j]$$

$$\text{para } 1 \leq j \leq m,\ j \neq i$$

Ejemplo: Calcular la matriz de confusión

X1	X2	X3	Y
0	0	1	0
0	1	0	0
1	1	0	0
0	0	1	1
1	1	1	1
0	0	1	1
1	1	0	1

El clasificador de K-vecinos más cercanos (KNN)

- **Advertencia** para el clasificador de Bayes: normalmente no conocemos la verdadera distribución condicional de $\Pr(Y|X)$ para datos reales.

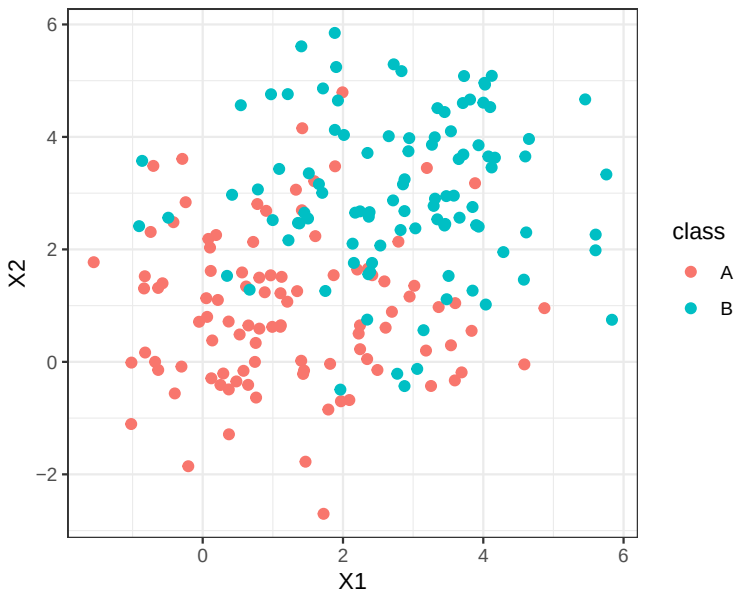
- Hemos discutido cómo estimarlo con regresión logística (para dos categorías).

- Alternativa: El clasificador de K-vecinos más cercanos (KNN) estima esta distribución condicional *no paramétricamente* y elige la categoría más probable (clasificador de Bayes).[4]

[4] Atención!! K se refiere al número de vecinos usados para la clasificación, y *no* al número de clases!! Esto último se supone conocido.

Un ejemplo sintético

- Simular 2×100 observaciones de una distribución normal bivariada con vectores de medias $\mu_A = (1,1)^T$, $\mu_B = (3,3)^T$, y matriz de covarianzas $\Sigma_A = \Sigma_B = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$.



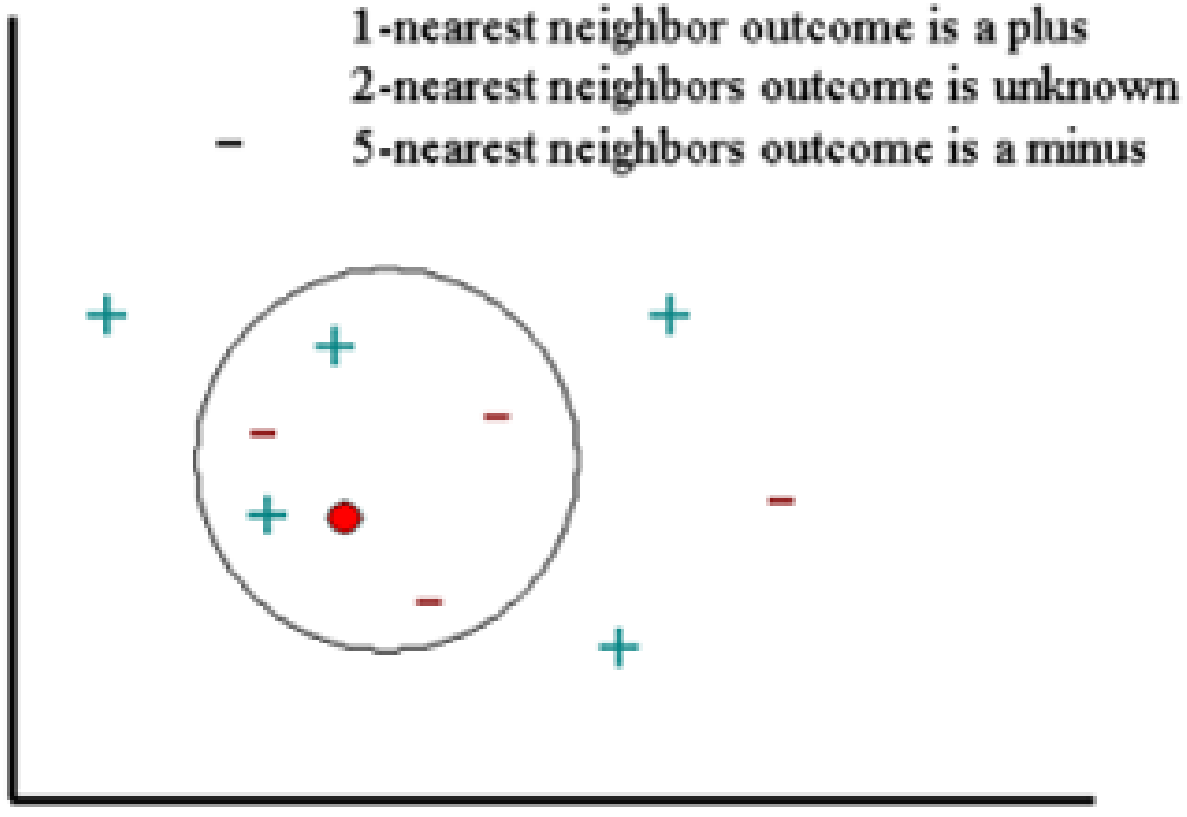
- Objetivo: encontrar una regla para clasificar una nueva observación en A o B, dados solo los puntos de datos (no el conocimiento sobre los parámetros verdaderos).

El clasificador de K-vecinos más cercanos (KNN) trabaja de la siguiente manera:

- Hay dos parámetros que debemos elegir: la distancia o métrica y el valor de K.
- Dada una nueva observación x_0 busca los K puntos en nuestros datos de entrenamiento que estén más cerca de ella (distancia).
  - Euclideana: $d(x, y) = (x - y)'(x - y)$
  - Manhattan: $d(x, y) = |x - y|$
- Estos puntos forman la vecindad de x_0, $\mathcal{N}_0$.
- La clasificación es realizada por *voto mayoritario*: x_0 se clasifica en la clase más frecuente entre sus vecinos

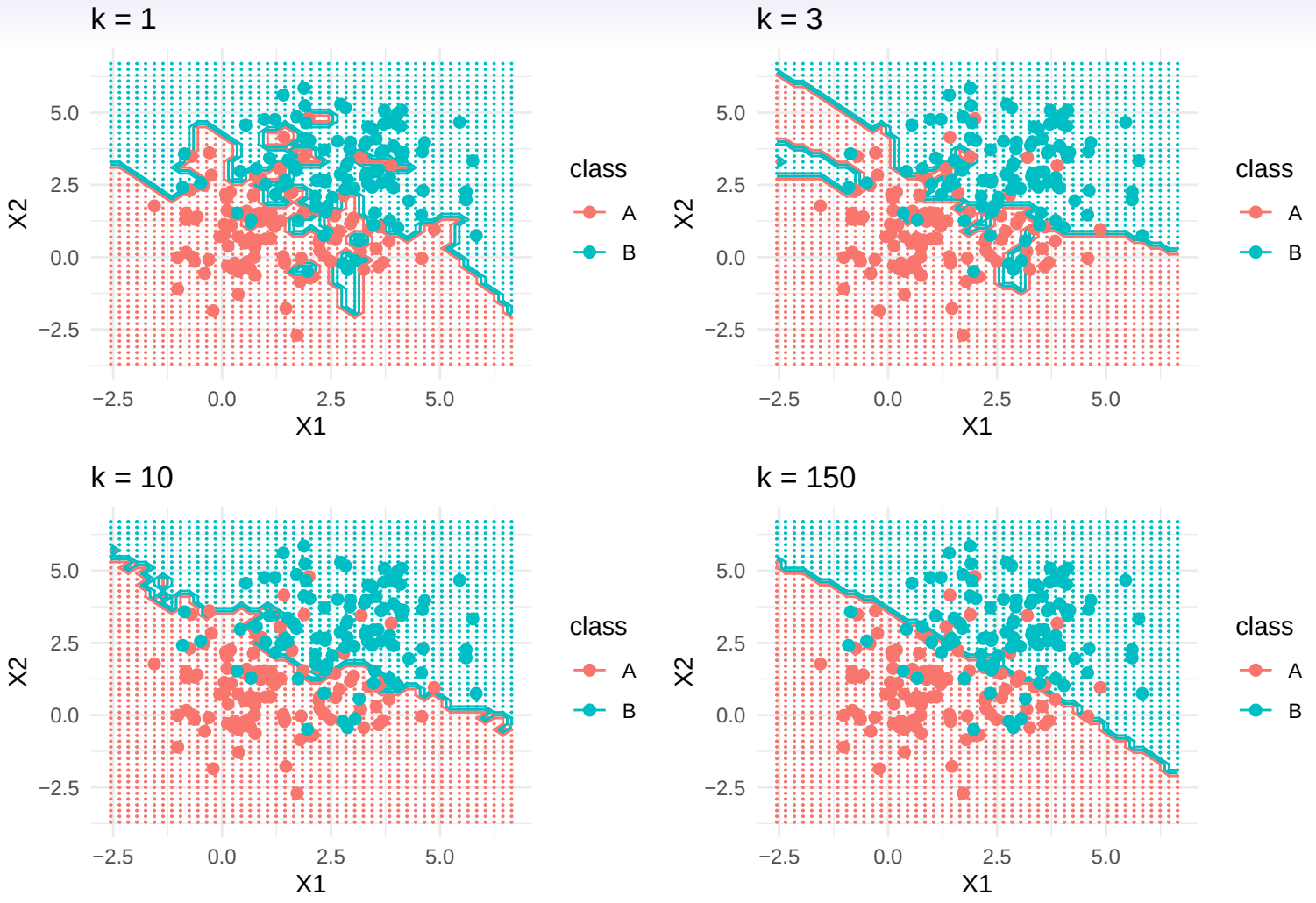
$$\Pr(Y = j|X = x_0) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} I(y_i = j) \ .$$

- En caso de empate se clasifica al azar. En Regresión la predicción es la media de la vecindad.


1-nearest neighbor outcome is a plus
2-nearest neighbors outcome is unknown
5-nearest neighbors outcome is a minus

En nuestro ejemplo:

- Supongamos que tenemos una nueva observación $x_0 = (x_{01}, x_{02})^T$ que queremos clasificar como perteneciente a la clase A o B.
- Ilustramos esto ajustando el clasificador de K-vecinos más cercanos a nuestros datos simulados con $K = 1, 3, 10$ y 150 (siguiente lámina).



- Los pequeños puntos de colores muestran las clases predichas para una cuadrícula espaciada uniformemente.
- Las líneas muestran los límites de decisión.

¿Cómo elegir K?

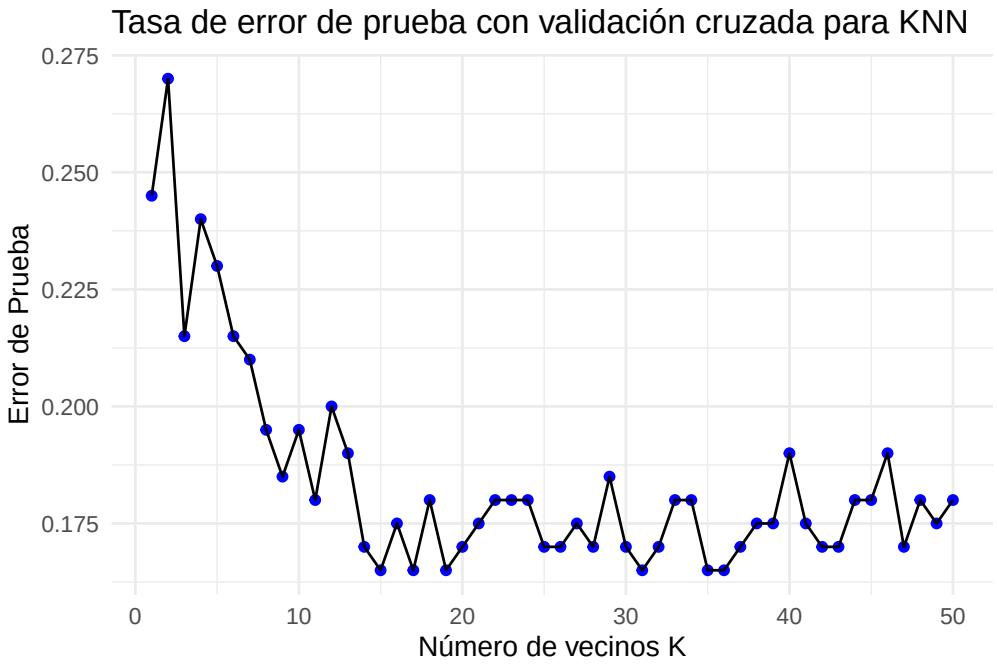
- $K = 1$: la clasificación se hace a la misma clase que el vecino más cercano.
- K grande: el límite de decisión tiende hacia una línea recta (que es el límite de Bayes en esta configuración).

Discusión:

- Dependiendo de la elección de K, ¿cuándo es grande el sesgo, cuándo es grande la varianza?
- ¿Cómo encontrar el valor óptimo de K?

El balance sesgo-varianza en un problema de clasificación

Encontrar el *valor óptimo* de K: Probar el poder predictivo para diferentes K, por ejemplo, mediante validación cruzada (siguiente capítulo).



La maldición de la dimensionalidad

El clasificador vecino más cercano puede ser bastante bueno si el número de predictores p es pequeño y el número de observaciones n es grande. Necesitamos suficientes vecinos cercanos para hacer una buena clasificación.

- La efectividad del clasificador KNN cae rápidamente cuando la dimensión del espacio del preditor es alta.
- ¿Por qué? Porque los vecinos más cercanos suelen estar lejos en grandes dimensiones y el método ya no es local. Esto se conoce como la *maldición de la dimensionalidad*.

Comentarios finales

- Las probabilidades cero afectan al clasificador Naïve Bayes. Una función de probabilidad a priori de Dirichlet es usada para resolver el problema.
- El proceso de discretizacion tambien parecer afectar el rendimiento del clasificador.
- Naïve Bayes es bastante barato. No tiene problemas para trabajar con 10,000 atributos.
- Naïve Bayes es un caso particular de Redes bayesianas

Resumen de Métodos de Clasificación

- Regresión Logística
- Análisis discriminante lineal
- Análisis discriminante cuadrático
- Análisis discriminante regularizado
- Naive Bayes
- KNN

Recordar:

- Regresión logística y KNN *estima directamente* $\Pr(Y = k \mid X = x)$ (paradigma diagnóstico).
- LDA, QDA, RDA y naive Bayes *estima indirectamente* $\Pr(Y = k \mid X = x) \propto f_k(x) \cdot \pi_k$ (paradigma de muestreo).

¿Qué método de clasificación es el mejor?

Ventajas del análisis discriminante

- El análisis discriminante es más estable que la regresión logística cuando
  - las clases están bien separadas. En ese caso, las estimaciones de los parámetros para el modelo de regresión logística son muy inestables.
  - si el número de observaciones n es pequeño y la distribución de los predictores X es aproximadamente (multivariante) normal.

- Además, el análisis discriminante lineal es popular cuando tenemos más de dos clases de respuesta.

Linealidad

Suponga un problema de clasificación binaria con una covariable.

- Recuerde que la regresión logística se puede escribir:

$$\log\left(\frac{p(x)}{1-p(x)}\right) = \beta_0 + \beta_1 x \ .$$

- Para un problema de dos clases, se puede demostrar que para LDA

$$\log\left(\frac{p_1(x)}{1-p_1(x)}\right) = \log\left(\frac{p_1(x)}{p_2(x)}\right) = c_0 + c_1 x_1 \ ,$$

  por lo tanto, la misma forma lineal. La diferencia está en cómo se estiman los parámetros.

LDA vs regresión logística

- La regresión logística utiliza la probabilidad condicional basada en $\Pr(Y \mid X)$.
- LDA utiliza la verosimilitud total basada en $\Pr(X, Y)$.
- A pesar de estas diferencias, en la práctica los resultados suelen ser muy similares [5], pero
  - DA es "más adecuado" en el entorno de clases múltiples.
  - si las distribuciones condicionales de clase son normales multivariadas, entonces se prefiere LDA (o QDA).
  - La regresión logística no hace suposiciones sobre las covariables y, por lo tanto, es preferible en muchas aplicaciones prácticas.
  - en medicina para problemas de dos clases, a menudo se prefiere la regresión logística (para la interpretación) y (siempre) junto con ROC y AUC (para la comparación de modelos).

y KNN?

- KNN se utiliza cuando los límites de la clase no son lineales.

[5] la regresión logística también puede ajustarse a límites cuadráticos como

Entonces: ¿Qué método de clasificación es el mejor?

La respuesta es: **¡depende!**

- La regresión logística es muy popular para la clasificación, especialmente cuando $K = 2$.
- DA es útil cuando n es pequeño o las clases están bien separados, y los supuestos de normalidad son razonables. También cuando $K > 2$.
- Naive Bayes es útil cuando p es muy grande
- KNN es completamente diferente, ya que no hace suposiciones sobre el límite de decisión ni la distribución de las variables (no paramétricas). Se espera que funcione mejor que LDA y la regresión logística cuando el límite es muy no lineal. Advertencia: No es posible la interpretación del efecto de las covariables.

Lea la Sección 4.5 de nuestro libro de texto.