Introducción a la analítica

Profesores César Augusto Gómez, Mauricio Alejandro Mazo y Juan Carlos Salazar

DEFINICIÓN FORMAL DEL PROBLEMA DE CLASIFICACIÓN

Dado un vector de características X y una respuesta cualitativa Y que toma valores en un conjunto $\mathcal{C}$ (por ejemplo, valores 0/1, color de cabello, cumple/incumple una obligación), la tarea de clasificación consiste en construir o formular una función $f(X)$ que toma como entrada el vector de características X y predice su valor para Y; es decir, $f(X) \in \mathcal{C}$. Frecuentemente, interesa estimar las probabilidades de que X pertenezca a cada categoría en el conjunto de categorías $\mathcal{C}$.

CLASIFICADOR DE BAYES. Es posible demostrar que la tasa de error $Average\left(I\left(y_0 \neq \widehat{y}_0\right)\right)$ se minimiza en promedio por medio de un clasificador muy simple que clasifica cada observación a la clase más probable o factible, dados los valores de sus predictores.

En otras palabras, se asigna simplemente una observación del conjunto de prueba con predictor x_0 a la clase j para la cual

$$Pr\left(Y = j|X = x_0\right)$$

sea la más grande. Esta es una probabilidad condicional. Este clasificador tan simple se conoce como **CLASIFICADOR DE BAYES**. En una situación con solo dos clases (0,1), el clasificador de Bayes corresponde a predecir la clase 1 si

$$Pr\left(Y = j|X = x_0\right) > 0.5$$

y a la clase 0 si

$$Pr\left(Y = j|X = x_0\right) \leq 0.5$$

CLASIFICADORES COMUNES EN AE

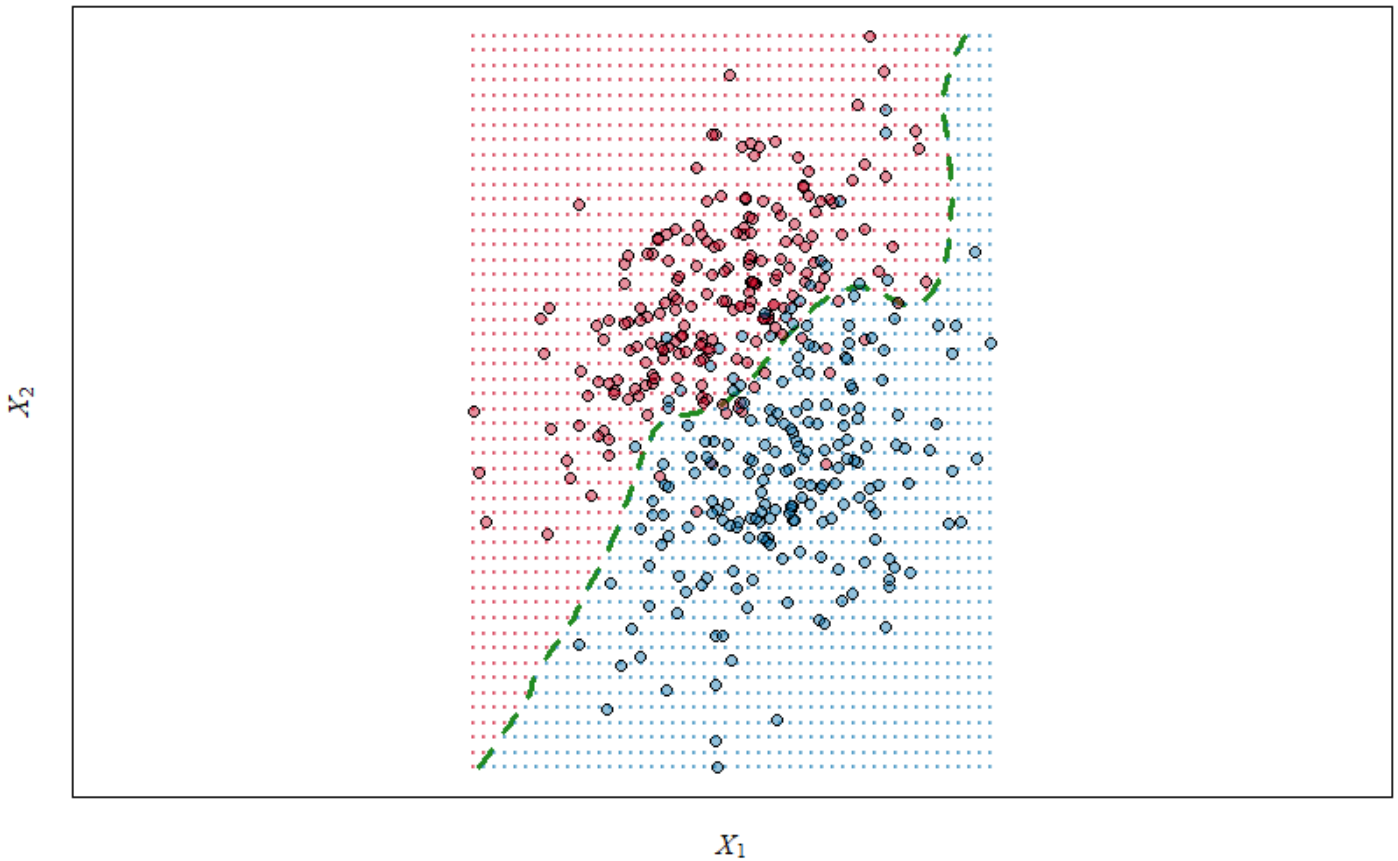


Figura 1: Bayes decision boundary separating two classes

El gráfico anterior ilustra un conjunto de datos simulado en un espacio bi-dimensional definido por las features X_1 y X_2. los puntos rojos y azules corresponden a observaciones de entrenamiento que pertenecen a dos clases distintas. Para cada valor de X_1 y X_2 hay una probabilidad distinta de que la respuesta sea roja o azul. Puesto que este es un conjunto de datos simulado, se sabe cómo los datos fueron generados y se pueden calcular estas probabilidades para cada valor de X_1 y X_2.

La región sombreada roja refleja el conjunto de puntos para los cuales $Pr(Y = Rojo \mid X = (X_1, X_2))$ es mayor que 0.5, mientras que la región sombreada en azul indica el conjunto de puntos para los cuales la probabilidad es menor o igual a 0.5. La línea verde se conoce como Frontera de decisión de Bayes (Bayes´s Boundary, la probabilidad es igual a 0.5). La predicción usando un clasificador de Bayes, está determinada por la Frontera de decisión de Bayes.

El clasificador de Bayes produce la menor tasa de error de prueba (Test Error Rate) posible (Ejercicio) y se conoce como Tasa de Error de Bayes. Puesto que el clasificador de Bayes siempre selecciona la clase para la cual $Pr(Y = j \mid X = x_0)$ es la más grande, la tasa de error en $X = x_0$ será $1 - \max_j Pr(Y = j \mid X = x_0)$. En general, **la tasa de error global de Bayes** está dada por:

$$1 - E\left(\max_j Pr(Y = j \mid X)\right)$$

Donde la esperanza promedia la probabilidad sobre todos los posibles valores para X.

CLASIFICADORES COMUNES EN AE

La tasa de error de entrenamiento o de prueba (Test Error Rate) se puede calcular de manera simple usando una tabla 2×2 (Confusion Matrix) de las respuestas observadas y las respuestas predichas. Por ejemplo, si esta tabla se obtiene con el training set (similar para una tabla obtenida con el test set):

Observed		Predicted	
	0	1	Total
0	a	b	a+b
1	c	d	c+d
Total	a+c	b+d	n

Entonces

$$\text{Training Error Rate} = \frac{b + c}{n}$$

CLASIFICADORES COMUNES EN AE

En teoría, siempre se desearía predecir respuestas categóricas usando el clasificador de Bayes. Pero, para datos reales, generalmente no se conoce la distribución condicional de la variable aleatoria $Y \mid X$, y por lo tanto no es fáctible calcular el clasificador de Bayes. Por lo tanto, el clasificador de Bayes es un 'gold estándard' inalcanzable contra el cual comparar otros métodos. Muchas aproximaciones intentan estimar la distribución condicional de la variable aleatoria $Y \mid X$, para luego clasificar una observación dada a la clase con la probabilidad estimada más grande. Se presentan algunos de los más comunes en la práctica.

NAIVE BAYES (BAYES INGENUO). En el AE, un clasificador ingenuo de Bayes (Naive Bayes classifier) es un clasificador probabilístico simple basado en la aplicación del teorema de Bayes con supuestos de independencia fuertes (ingenuos) entre las características o features. El naive Bayes es un modelo de probabilidad condicional. Considere el caso de dos clases 0 o 1 (la misma formulación es válida para más clases)

$$Pr\left(Y = j | X = x\right), \quad j \in \{0,\ 1\}$$

Se lee, *Probabilidad de asignar un objeto a la clase j dado que se observó un vector de features x*.

Dada una nueva observación x, (1) encuentre la probabilidad de que esta pertenezca a cada clase, y (2) seleccione la más probable.

CLASIFICADORES COMUNES EN AE

- Naive Bayes, como su nombre lo indica, se basa en el Teorema de Bayes.
- ¿Por qué ingenuo? Asume que todas las características en la base de datos son igual de importantes y además, asume que todas las características son independientes, lo cual en la práctica podría no verificarse pues unas características podrían ser más importantes que otras.
- Los clasificadores bayesianos utilizan datos de entrenamiento para calcular la probabilidad de un resultado.
- Dicha probabilidad depende de las características (features) de los objetos a clasificar.
- Así, sirve para determinar a qué clase pertenece un objeto, según sus características.

Recuerde el teorema de Bayes:

$$Pr\left(Y = j \mid x\right) = \frac{Pr\left(x \mid Y = j\right) Pr\left(Y = j\right)}{Pr\left(x\right)}$$

Aquí,

- $Pr\left(Y = j \mid x\right)$: Probabilidad Posterior, es la probabilidad de estar en la clase j dado el vector x.
- $Pr\left(x \mid Y = j\right)$: Likelihood, es la probabilidad de observar el vector x dado que se está en la clase j.
- $Pr\left(Y = j\right)$: Probabilidad a priori o previa de ocurrencia de la clase j.
- $Pr\left(x\right)$: Probabilidad a priori o previa de observar el vector x.

El clasificador Naive de Bayes asume que todas las features (componentes del vector x) son independientes y que son igualmente importantes. Con este supuesto, la probabilidad (Likelihood) de observar el vector de features $x = (x_1,\ x_2, \ldots, x_p)$ dado que se está en la clase j es:

$$Pr\,(x \mid Y = j) = Pr\,(x_1 \mid Y = j) \times Pr\,(x_2 \mid Y = j) \times \cdots \times Pr\,(x_p \mid Y = j)$$

Aquí, $Pr\,(x_i \mid Y = j)$: Probabilidad de que la clase j genere el valor observado para la feature i, $i = 1, 2, \ldots, p$.

De esta manera, teniendo en cuenta que $x = (x_1, x_2, \ldots, x_p)$, el método Naive Bayes, se puede implementar usando:

$$
\begin{aligned}
Pr(Y = j \mid x) &= \frac{Pr(x \mid Y = j)\, Pr(Y = j)}{Pr(x)} \\
&\approx \frac{1}{Pr(x)} \times \left(Pr(x_1 \mid Y = j) \times Pr(x_2 \mid Y = j) \times \cdots \times Pr\left(x_p \mid Y = j\right) \times Pr(Y = j) \right) \\
&\propto Pr(x_1 \mid Y = j) \times Pr(x_2 \mid Y = j) \times \cdots \times Pr\left(x_p \mid Y = j\right) \times Pr(Y = j)
\end{aligned}
$$

En este caso $Pr(x)$ es una constante de normalización.

EJEMPLO. Considere los siguientes datos:

Name	Sex
Morgan	M
Reid	F
Morgan	M
Morgan	F
Everardo	M
Francis	M
Jennifer	F

hay tres 'F' y 4 'M'. Suponga que llega una nueva observación pero que no la podemos ver, solo se sabe que el nombre es Morgan. La pregunta es ¿Es Morgan F o M?

EJEMPLO. Considere los siguientes datos: la situación es como sigue:

Name	Sex
Morgan	M
Reid	F
Morgan	M
Morgan	F
Everardo	M
Francis	M
Jennifer	F
Morgan	??

¿Es Morgan F o M?

Para responder la pregunta, se usará el Naive Bayes. La probabilidad condicional de ser mujer (F) dado que se llama Morgan:

$$Pr\left(Y = j \mid x\right) = \frac{Pr\left(x \mid Y = j\right) Pr\left(Y = j\right)}{Pr\left(x\right)}$$

$$\begin{aligned} Pr\left(Y = F|x = Morgan\right) &= \frac{Pr\left(x = Morgan|Y = F\right) Pr\left(Y = F\right)}{Pr\left(x = Morgan\right)} \\ &= \frac{1/3 \times 3/7}{3/7} \\ &= 1/3 \\ &= 0.333 \end{aligned}$$

La probabilidad condicional de ser hombre (M) dado que se llama Morgan:

$$
\begin{aligned}
Pr\left(Y = M | x = Morgan\right) &= \frac{Pr\left(x = Morgan | Y = M\right) Pr\left(Y = M\right)}{Pr\left(x = Morgan\right)} \\
&= \frac{2/4 \times 4/7}{3/7} \\
&= 0.667
\end{aligned}
$$

Cómo $Pr\left(Y = F | x = Morgan\right) < Pr\left(Y = M | x = Morgan\right)$ es más probable que Morgan sea clasificado como un hombre (M).

Usando R, libreria e1071 (library(e1071)):

```
library(e1071)
names <- read.csv("F:/INTRODUCCIÓN A LA ANALÍTICA/IA Virtual 02_2020/names.csv",
                  stringsAsFactors=TRUE,sep=';')
names
```

```
##   ID     NAME SEX
## 1  1   MORGAN   M
## 2  2     REID   F
## 3  3   MORGAN   M
## 4  4   MORGAN   F
## 5  5 EVERALDO   M
## 6  6  FRANCIS   M
## 7  7 JENNIFER   F
```

```
names_classifier <- naiveBayes(SEX~NAME, data=names)
summary(names_classifier)
```

```
##           Length Class  Mode
## apriori   2      table  numeric
## tables    1      -none- list
## levels    2      -none- character
## isnumeric 1      -none- logical
## call      4      -none- call
```

```
test_data=data.frame(NAME=c("MORGAN"))
names_predict <- predict(names_classifier, test_data, type="raw")
names_predict
```

```
##              F         M
## [1,] 0.3333333 0.6666667
```

```
sex_predict <- predict(names_classifier, test_data, type="class")
sex_predict
```

Caso de más de una feature. Considere el siguiente conjunto en el que se tiene información de cumplimiento crediticio (defaulted=Yes, cumple con la obligación):

Home-Owner	Marital-Status	Job-Experience	Defaulted
Yes	Single	3	No
No	Married	4	No
No	Single	5	No
Yes	Married	4	No
No	Divorced	2	Yes
No	Married	4	No
Yes	Divorced	2	No
No	Married	3	Yes
No	Married	3	No
Yes	Single	2	Yes

Dado que Juan tiene las siguientes features: Home-Owner=No, Marital-Status=Married, y Job-Experience=3, prediga si Juan entrará en mora. Hagalo a mano y en R.

Sea $x = \left(Home - Owner =' No', Marital - Status =' Married', Job - Experience =' 3'\right)$. Entonces

$$
\begin{aligned}
Pr\left(Y =' No' \mid x\right) &= \frac{Pr\left(x|Y =' No'\right) \times Pr\left(Y =' No'\right)}{Pr\left(x\right)} \\
&= \frac{1}{Pr\left(x\right)} \left(Pr\left(x_1 =' N0'|Y =' No'\right) \times Pr\left(x_2 =' Married'|Y =' No'\right)\right. \\
&\left. \times Pr\left(x_3 =' 3'|Y =' No'\right) \times Pr\left(Y =' No'\right)\right) \\
&= \frac{\frac{4}{7} \times \frac{4}{7} \times \frac{2}{7} \times \frac{7}{10}}{Pr\left(x\right)} \\
&= \frac{0.065}{Pr\left(x\right)} \\
&\propto 0.065
\end{aligned}
$$

$$
\begin{aligned}
Pr\left(Y =' \ Yes' \mid x\right) &= \frac{Pr\left(x|Y =' \ Yes'\right) \times Pr\left(Y =' \ Yes'\right)}{Pr\left(x\right)} \\
&= \frac{1}{Pr\left(x\right)} \left(Pr\left(x_1 =' \ N0'|Y =' \ Yes'\right) \times Pr\left(x_2 =' \ Married'|Y =' \ Yes'\right)\right. \\
&\left. \times Pr\left(x_3 =' \ 3'|Y =' \ Yes'\right) \times Pr\left(Y =' \ Yes'\right)\right) \\
&= \frac{\frac{2}{3} \times \frac{1}{3} \times \frac{1}{3} \times \frac{3}{10}}{Pr\left(x\right)} \\
&= \frac{0.022}{Pr\left(x\right)} \\
&\propto 0.022
\end{aligned}
$$

Como $0.065 > 0.022$ Juan se clasifica como no moroso (defaulted=‘No’)

Para hallar la constante de normalización $Pr(x)$, note que

$$\frac{0.065}{Pr(x)} + \frac{0.022}{Pr(x)} = 1$$

Entonces $Pr(x) = 0.087$ y así

$$Pr(Y =' No' \mid x) = \frac{0.065}{0.087} = 0.7471$$

y

$$Pr(Y =' Yes' \mid x) = \frac{0.022}{0.087} = 0.2529$$

Son las probabilidades posteriores.

Usando R, libreria e1071 (library(e1071)):

```
library(e1071)
defaulted <- read.csv("F:/INTRODUCCIÓN A LA ANALÍTICA/IA Virtual 02_2020/Defaulted_1.csv",
                      stringsAsFactors=FALSE,sep=';')
train=data.frame(defaulted[1:10,])
test=data.frame(defaulted[11:11,1:4])

defaulted_classifier <- naiveBayes(Defaulted~Home_Owner
                                   +marital_Status+Job_Experience, data=train, laplace=0.128)
defaulted_predict <- predict(defaulted_classifier, test,type="raw")
defaulted_predict
```

```
##              No       Yes
## [1,] 0.7470587 0.2529413
```

Hasta el momento muchas de las características han sido categóricas, pero algunas características pueden ser numéricas, por ejemplo, la hora a la que se envía un email. ¿Qué hacer en esos casos? Discretizar la variable numérica puede ser una alternativa. Usar quintiles u otras divisiones que sean razonables. Por ejemplo, franja del día en la que llega un email.

EJEMPLO. CALCULO DEL TRAINING Y DEL TEST ERROR RATES. CONSIDERE LOS DATOS DE DEFAULT.

ID	Home_Owner	marital_Status	Job_Experience	Defaulted
1	Yes	Single	3	No
2	No	Married	4	No
3	No	Single	5	No
4	Yes	Married	4	No
5	No	Divorced	2	Yes
6	No	Married	4	No
7	Yes	Divorced	2	No
8	No	Married	3	Yes
9	No	Married	3	No
10	Yes	Single	2	Yes
11	No	Married	3	No

EJEMPLO. CALCULO DEL TRAINING Y DEL TEST ERROR RATES. Training Set:

ID	Home_Owner	marital_Status	Job_Experience	Defaulted
4	Yes	Married	4	No
8	No	Married	3	Yes
11	No	Married	3	No
10	Yes	Single	2	Yes
7	Yes	Divorced	2	No
1	Yes	Single	3	No
3	No	Single	5	No
9	No	Married	3	No

EJEMPLO. CALCULO DEL TRAINING Y DEL TEST ERROR RATES. Test Set:

ID	Home_Owner	marital_Status	Job_Experience	Defaulted
2	No	Married	4	No
5	No	Divorced	2	Yes
6	No	Married	4	No

CLASIFICADORES COMUNES EN AE

```
#Calculating the test error rate and the training error rate
library(naivebayes)
```

```
Warning: package 'naivebayes' was built under R version 3.6.3
```

```
naivebayes 0.9.7 loaded
```

```
set.seed(1)
defaulted <- read.csv("F:/INTRODUCCIÓN A LA ANALÍTICA/IA Virtual 02_2020/Defaulted_2.csv",
                      stringsAsFactors=FALSE,sep=';')
df=data.frame(defaulted)
smp_size <- floor(0.75 * nrow(df))
train_ind <- sample(seq_len(nrow(df)), size = smp_size)
train <- df[train_ind, ]
test <- df[-train_ind, ]
y_train=train$Defaulted
y_test=test$Defaulted
defaulted_classifier <- naive_bayes(Defaulted~Home_Owner
                                    +marital_Status+Job_Experience, data=train[,2:5],laplace=0.128)
predict_train<-predict(defaulted_classifier,newdata=train[,2:4],type="class")
predict_test<-predict(defaulted_classifier,newdata=test[,2:4],type="class")
t<-table(predict_train,y_train)
t1<-table(predict_test,y_test)
Train_error<-(t[1,2]+t[2,1])/(sum(t))
Test_error<-(t1[1,2]+t1[2,1])/(sum(t1))
Train_error
```

```
[1] NaN
```

```
Test_error
```

```
[1] NaN
```

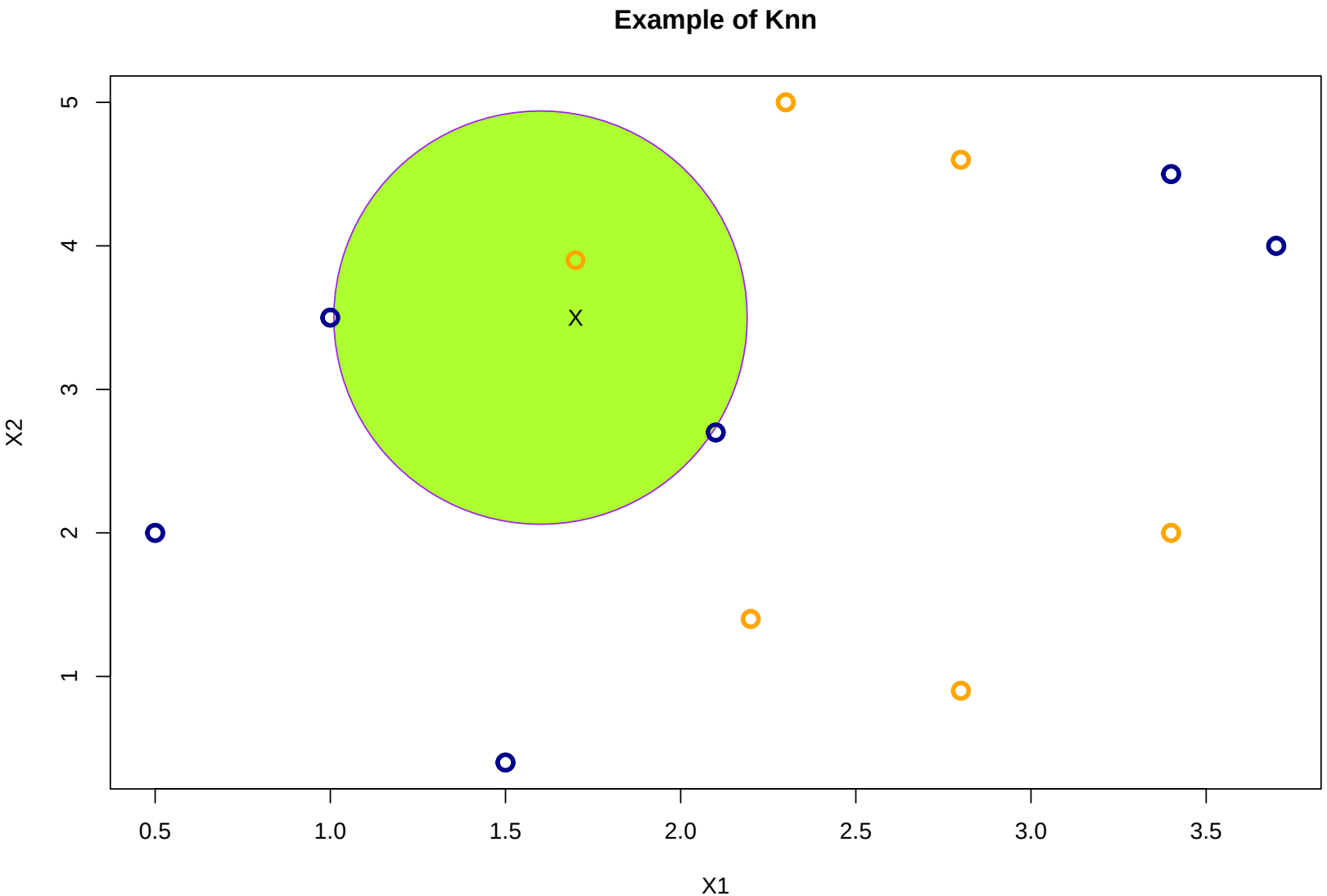
K VECINOS MÁS CERCANOS (K-Nearest Neighbors, KNN o Knn). Recuerde que en teoría, siempre desearíamos predecir respuestas categóricas usando el Clasificador de Bayes. Pero, para datos reales, generalmente no se conoce la distribución condicional de la variable aleatoria $Y \mid X$, y por lo tanto no es fáctible calcular el clasificador de Bayes. Por lo tanto, el Clasificador de Bayes es un gold estándard inalcanzable contra el cual comparar otros métodos. Muchas aproximaciones intentan estimar la distribución condicional de la variable aleatoria $Y \mid X$, para luego clasificar una observación dada a la clase con la probabilidad estimada más grande.

K VECINOS MÁS CERCANOS (K-Nearest Neighbors, KNN o Knn). El Knn es un método ampliamente usado en la práctica como un clasificador. Dado un entero positivo K y una observación de prueba x_0, el clasificador Knn primero identifica los K puntos en el conjunto de entrenamiento que están más cerca de x_0, y los representa por el conjunto $\mathcal{N}_0$. Luego, Knn estima la probabilidad condicional para la clase j, $Pr(Y = j \mid X = x_0)$, como **La fracción de puntos en $\mathcal{N}_0$ cuyas respuestas son iguales a j**:

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

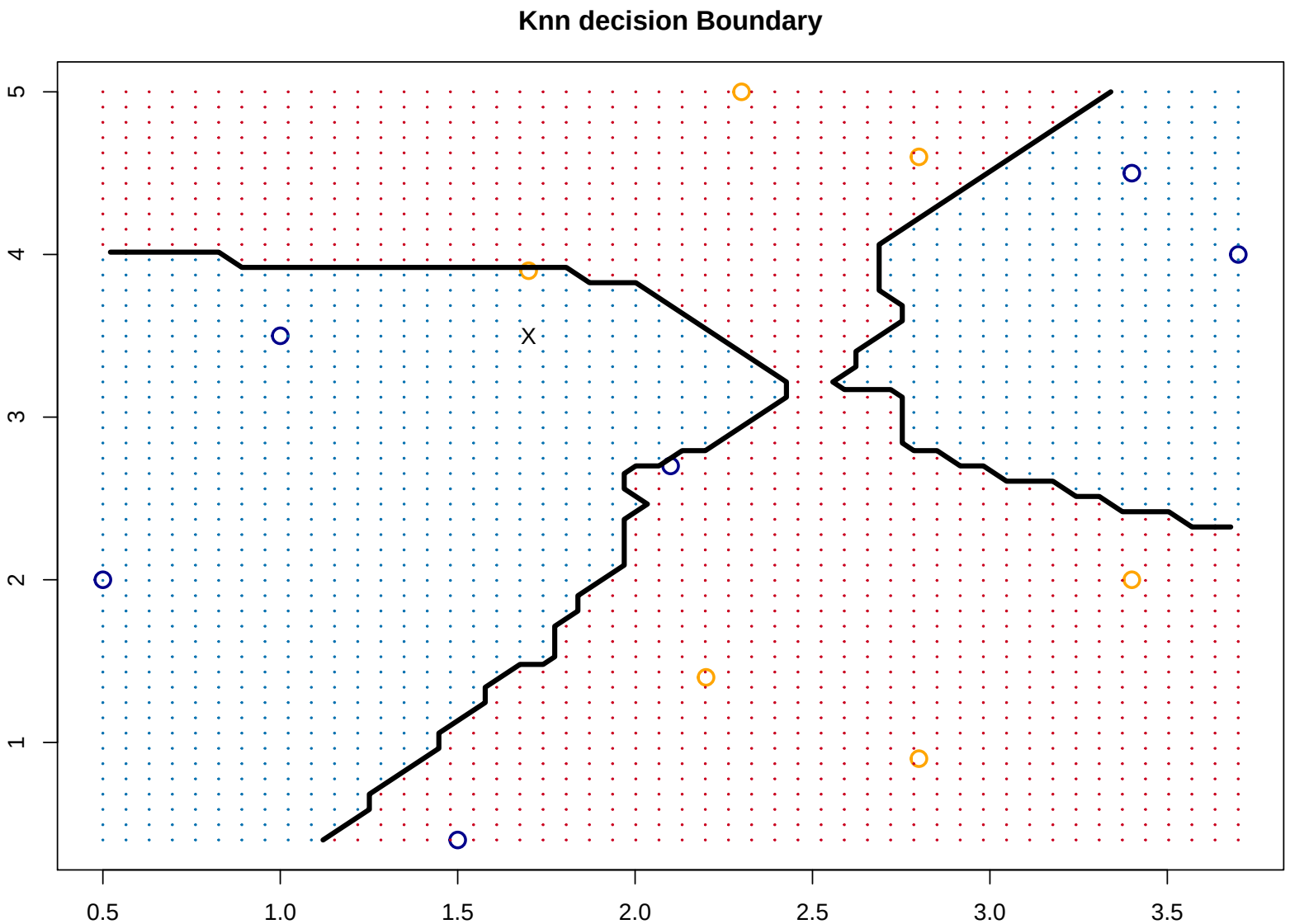
Finalmente, Knn aplica la regla de Bayes y clasifica la observación de prueba (test observation) x_0 a la clase con la probabilidad más grande.

El siguiente gráfico, ilustra el Knn.



El gráfico anterior muestra un conjunto de entrenamiento pequeño que consiste de 6 puntos naranjados y 6 puntos azules. El objetivo es hacer una predicción para el punto respresentado por "X". Suponga que se selecciona $K = 3$. Entonces, Knn identifica primero las 3 observaciones más cercanas al punto "X" (ver círculo verde, ese es el conjunto $\mathcal{N}_0$). $\mathcal{N}_0$ consiste de 2 puntos azules y un punto naranja, que resultan en estimaciones de $2/3$ para la clase azul y $1/3$ para la clase naranja. Entonces Knn dirá que el punto "X" pertenece a la clase azul. El siguiente gráfico muestra Knn aplicado, con $K = 3$, a todos los posibles valores de X_1 y X_2. En la región sombreada azul se ubicaran los puntos clasificados en esa clase y en la región sombreada naranja se ubicarán los clasificados en esa clase.

CLASIFICADORES COMUNES EN AE



Knn está diseñado para trabajar con features que son continuas. Más adelante se discute una alternativa para trabajar con features que son categóricas. **En todo caso, si se cuenta con variables continuas, antes de implementar Knn cada variable numérica del conjunto de features se debe normalizar**, usando alguna de estas dos alternativas (asuma que la variable que se quiere normalizar es x):

- $$z = \frac{x - \text{mín}(x)}{\text{máx}(x) - \text{mín}(x)}$$
- $$z = \frac{x - \bar{x}}{\widehat{\sigma}_x}$$

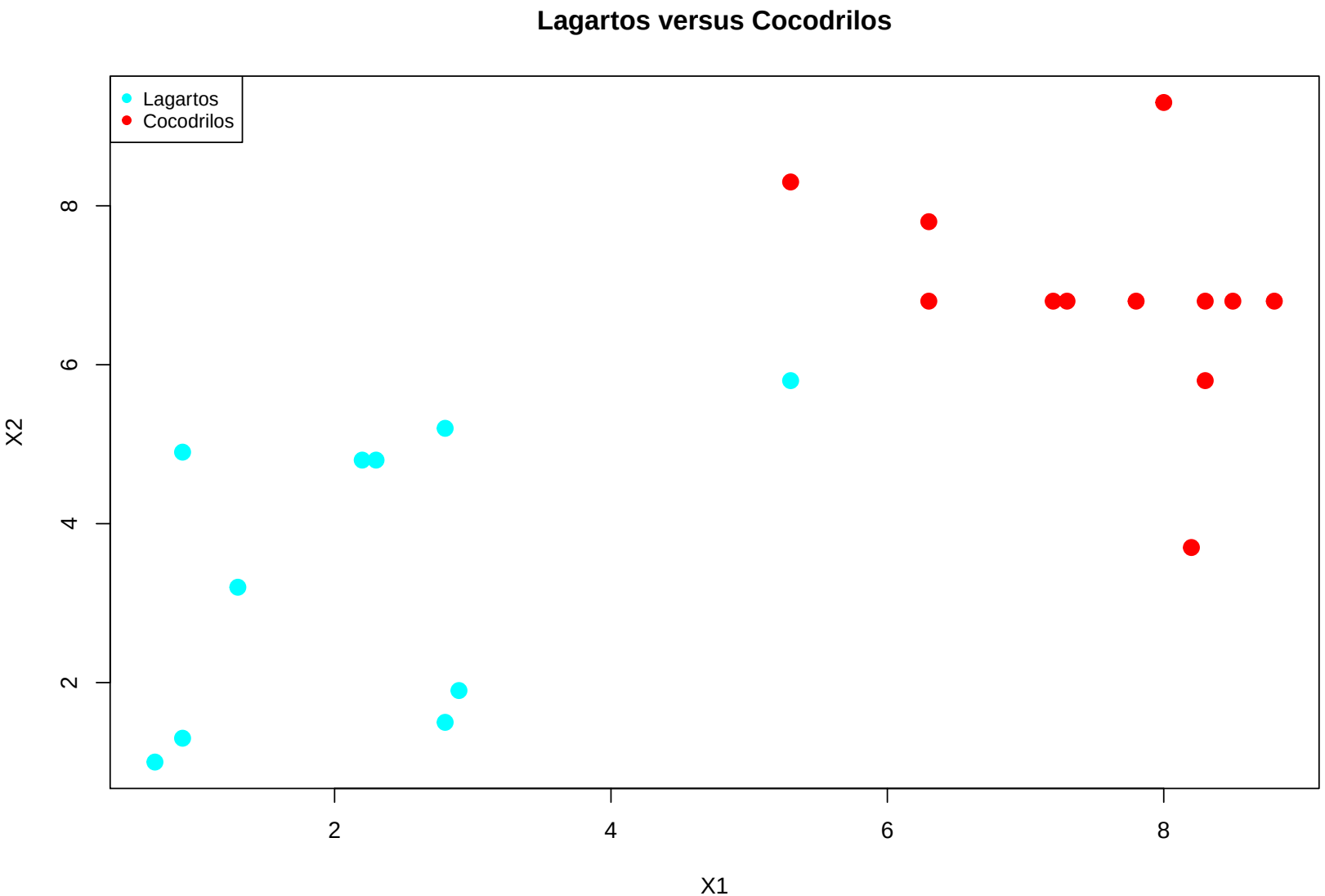
SE DEBE NORMALIZAR EL CONJUNTO DE ENTRENAMIENTO Y EL DE TEST CON EL MISMO CRITERIO AMBOS.

EJEMPLO: LAGARTOS VERSUS COCODRILOS. Considere la siguiente base de datos de medidas tomadas a lagartos (L) y a cocodrilos (C). Las medidas fueron Tamaño de la boca (Mouth_Size) y Longitud del cuerpo (Body_Length). Se muestran los 10 primeros de 22:

ID	Mouth_Size	Body_Length	Clase
1	0.7	1.0	L
2	2.9	1.9	L
3	1.3	3.2	L
4	0.9	4.9	L
5	2.8	5.2	L
6	5.3	8.3	C
7	6.3	7.8	C
8	8.0	9.3	C
9	8.3	5.8	C
10	8.3	6.8	C

Se tienen mediciones de cinco ejemplares: ID=11 Mouth_Size=9 y Body_Length=7.9, ID=12 Mouth_Size=1.2 y Body_Length=3.7, ID=13 Mouth_Size=10 y Body_Length=8, ID=14 Mouth_Size=1 y Body_Length=2, ID=15 Mouth_Size=1.1 y Body_Length=2.3. Usando Knn con K=12, prediga a qué grupo pertenecerá cada nuevo ejemplar.

Plot of the data



Knn con $K = 12$

```
## [1] "1=C" "2=L"
```

```
##      Predicted      prob      Cprob
## [1,]         1 0.9166667 0.08333333
## [2,]         2 0.8333333 0.16666667
## [3,]         1 1.0000000 0.00000000
## [4,]         2 0.8333333 0.16666667
## [5,]         2 0.8333333 0.16666667
```

Es decir, sujetos 11 y 13 fueron clasificados como C y sujetos 12, 14 y 15 como L.

Ahora se calculará la tasa de error de entrenamiento y de prueba, considerando el conjunto completo de datos (incluyendo estas últimas 5 predicciones).

```
library(MASS)
library(class)
library(naivebayes)
cocodrile<-read.csv("F:/INTRODUCCIÓN A LA ANALÍTICA/IA Virtual 02_2020/cocodrilos_1.csv",
                    header=T,dec=',',sep=';')
df=data.frame(cocodrile)
normalize <- function(x) {
  norm <- ((x - min(x))/(max(x) - min(x)))
  return (norm)
}
smp_size <- floor(0.8 * nrow(df))
train_ind <- sample(seq_len(nrow(df)), size = smp_size)
train <- normalize(df[train_ind,2:3 ])
test <- normalize(df[-train_ind,2:3 ])
y_train=df[train_ind,4]
y_test=df[-train_ind,4]
fit.knn_train<-knn(train=train, test=train,cl=y_train, k=12, prob=TRUE)
fit.knn_Test<-knn(train=train, test=test,cl=y_train, k=12, prob=TRUE)
Predicted_train<-factor(fit.knn_train)
Predicted_test<-factor(fit.knn_Test)
t<-table(Predicted_train,y_train)
t1<-table(Predicted_test,y_test)
Train_error<-(t[1,2]+t[2,1])/(sum(t))
Test_error<-(t1[1,2]+t1[2,1])/(sum(t1))
c(Train_error,Test_error)
```

```
## [1] 0.125 0.000
```

A pesar del hecho de que Knn es un método simple, puede producir clasificaciones muy cercanas a las que se obtienen con el clasificador óptimo de Bayes. La siguiente figura, muestra un conjunto de datos en dos variables X_1 y X_2. En este caso se usó $K = 8$. Note que aunque Knn no conoce la verdadera distribución, la frontera de decisión del Knn está muy cercana a la frontera de decisión del clasificador de Bayes.

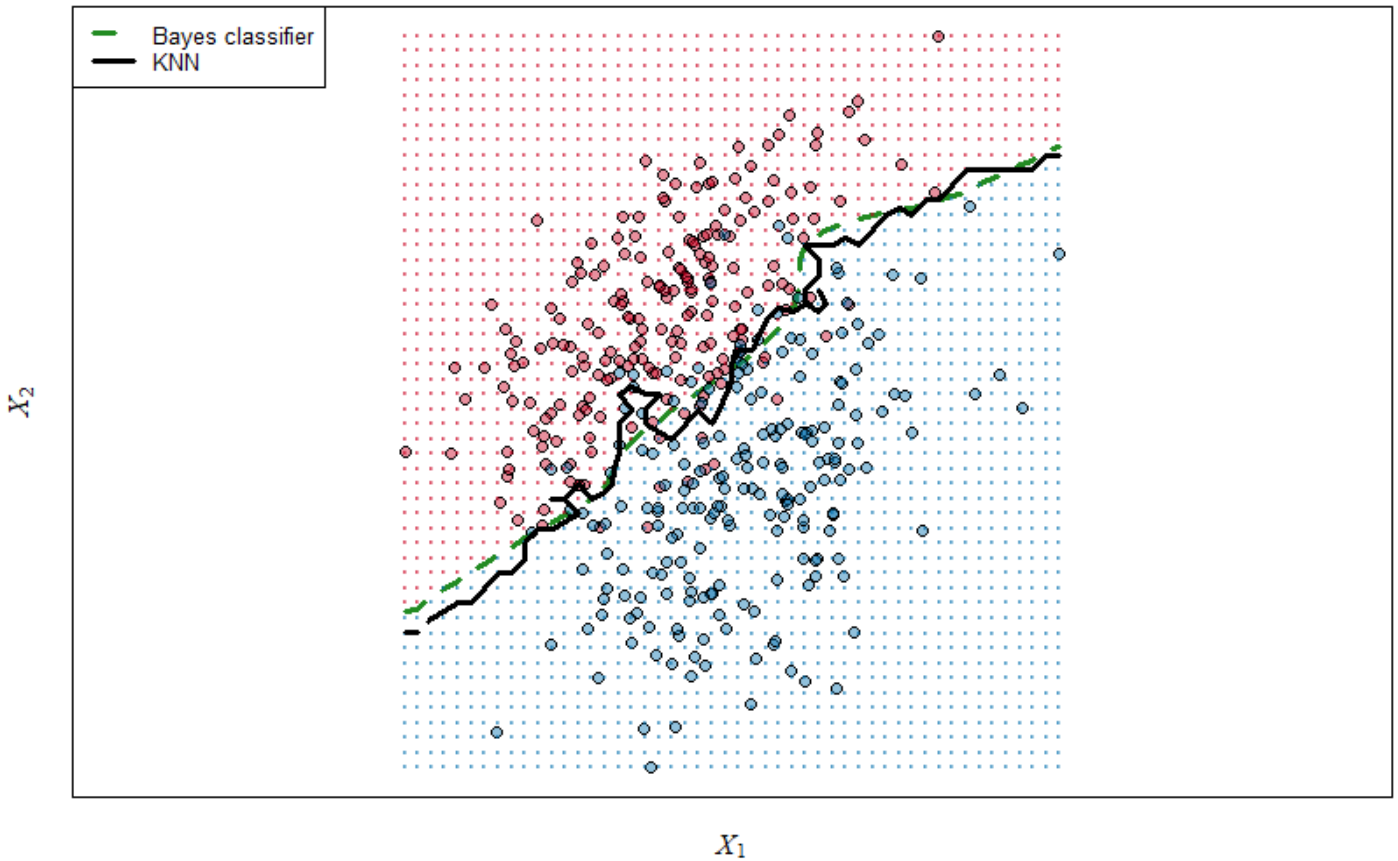


Figura 2: Knn and Bayes decision boundaries separating two classes

La elección de K tiene un efecto drástico en el clasificador Knn que se obtiene. Considere un conjunto de datos simulados en dos variables X_1 y X_2. Se ajustará Knn con valores de K pequeño y grande.

CLASIFICADORES COMUNES EN AE

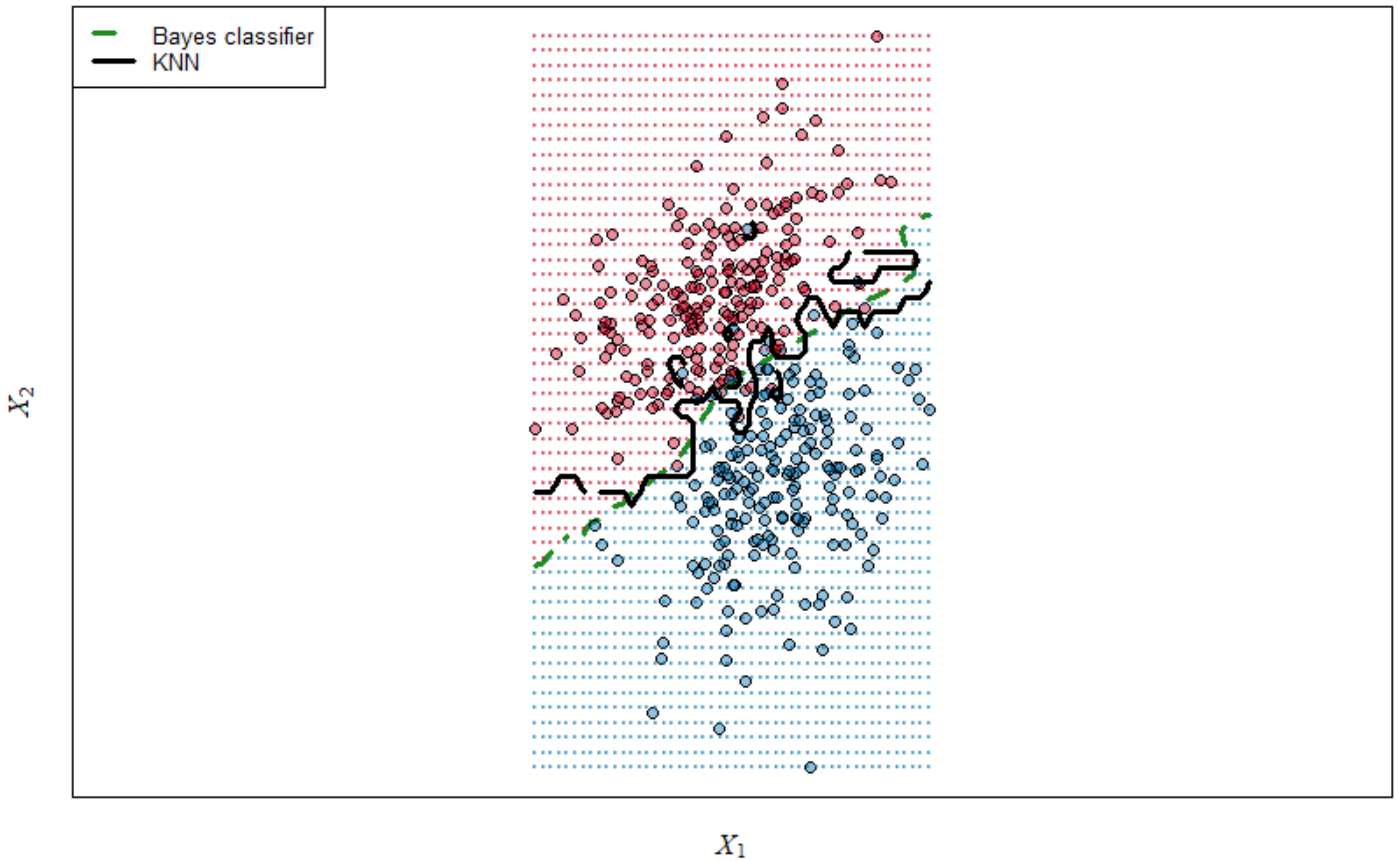


Figura 3: Knn and Bayes decision boundaries separating two classes

CLASIFICADORES COMUNES EN AE

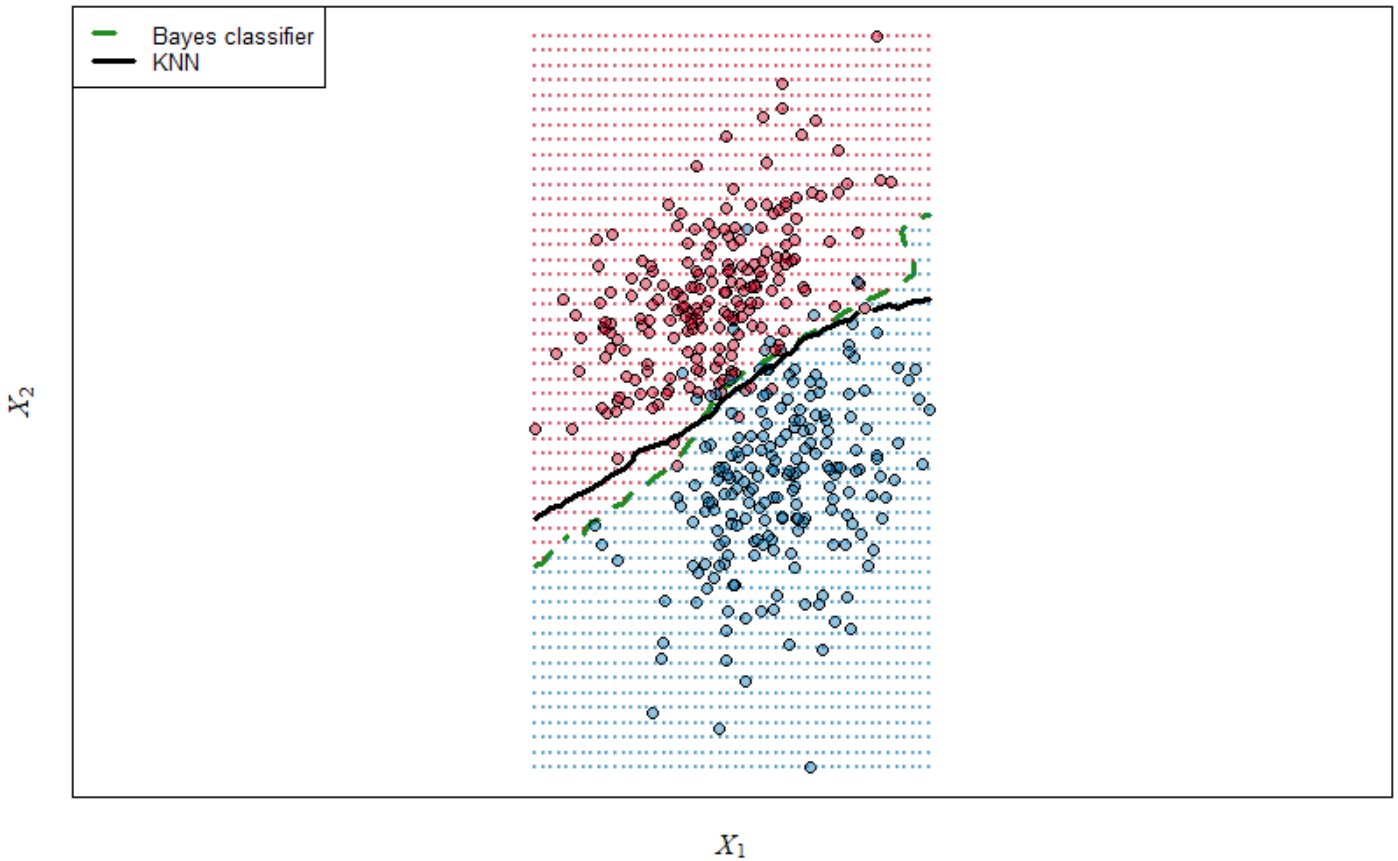


Figura 4: Knn and Bayes decision boundaries separating two classes

Note la diferencia cuando se usa, con unos datos simulados, $K = 1$ y $K = 50$. A menor valor de K la frontera es mucho más curveada e incluso encuentra patrones que la frontera de Bayes no encuentra; en otras palabras, para valores de K pequeños Knn es muy flexible. A medida que K se incrementa, el método es menos flexible y produce una frontera cercana a la lineal. **Conclusión:** Knn es un clasificador de baja varianza y sesgo grande.