

Tarea 6, Reconocimiento de Patrones

Francisco Javier Peralta Ramírez

1. Consideramos un problema de clasificación binaria $y \in \{-1, 1\}$. Se quiere construir un clasificador de la forma

$$\hat{y}(x) = \text{sign}(f(x))$$

donde f minimiza la función de costo de $C = E(1 - Yf(X))^2$. Verifica que para la f que minimiza esta función de costo $\hat{y}(x) = \text{sign}(f(x))$ corresponde al Clasificador Bayesiano óptimo.

$$\begin{aligned}\min_f C &= E(1 - Yf(X))^2 \\ &= E_X E_{Y=y|X=x} (1 - Yf(x))^2\end{aligned}$$

Basta con minimizar para cada x

$$\begin{aligned}\min_f E_{Y=y|X=x} (1 - Yf(x))^2 \\ &= (1 - f(x))^2 P(Y = 1|X = x) + (1 + f(x))^2 P(Y = -1|X = x) \\ &= (1 - f(x))^2 P(Y = 1|X = x) + (1 + f(x))^2 (1 - P(Y = 1|X = x)) \\ &= (1 + f(x))^2 + P(Y = 1|X = x)[(1 - f(x))^2 - (1 + f(x))^2] \\ &= 1 + 2f(x) + f(x)^2 + P(Y = 1|X = x)[-4f(x)] \\ &= 1 + 2f(x)[1 - P(Y = 1|X = x)] + f(x)^2\end{aligned}$$

Si nos limitamos a $-1 \leq f(x) \leq 1$ cuando $P(Y = 1|X = x) > 1/2$ el valor $f(x)$ que minimiza será positivo por lo que tomamos $\hat{y}(x) = 1$, cuando $P(Y = 1|X = x) < 1/2$ tomamos $\hat{y}(x) = -1$. Lo cual equivale al Clasificador Bayesiano óptimo

$$\begin{aligned}P(Y = 1|X = x) > P(Y = -1|X = x) &\rightarrow \hat{y}(x) = 1 \\ P(Y = 1|X = x) < P(Y = -1|X = x) &\rightarrow \hat{y}(x) = -1\end{aligned}$$

2. Usamos SVM para clasificar dos tipos (clases) de textos. Leemos los documentos y creamos una matriz de frecuencias.

```
library("tm")
library("SnowballC")
library("e1071")

alldir=DirSource("./texts", encoding = "UTF-8", recursive=TRUE)
news <- Corpus(alldir, readerControl=list(reader=readPlain, language="en"))
dtm <- DocumentTermMatrix(news, control=list(removePunctuation=TRUE,
                                             tolower=TRUE, stopwords=c("english"), stripWhitespace=TRUE,
                                             minWordLength=2))
d <- DocumentTermMatrix(news, list(dictionary=findFreqTerms(dtm, 50)))
```

Generamos las categorías usando los nombres de las carpetas padre de los archivos, *sci.med* y *sci.space*.

```
classvec <- vector()
# código para categorías de
# https://github.com/chenmiao/Big_Data_Analytics_Web_Text
for (filedir in alldir$filelist) {
  classlabel=basename(dirname(filedir))
  classvec=c(classvec, classlabel)
}
classvec <- factor(classvec)
```

Separamos los datos en un conjunto de entrenamiento y de prueba

```
#separar conjunto de pruebas
rand_idxes <- sample(nrow(d))
n_test = floor(nrow(d) * 0.75)

train <- d[head(rand_idxes, n=n_test), ]
test <- d[tail(rand_idxes, n=(nrow(d)-n_test)), ]

train_class <- classvec[head(rand_idxes, n=n_test)]
test_class <- classvec[tail(rand_idxes, n=(nrow(d)-n_test))]
```

Entrenamos el modelo y probamos

```
svm_model <- svm(train, train_class, kernel="linear")

pred <- predict(svm_model, test)
table(pred, test_class)
```

Y obtuvimos los resultados:

		classes	
		sci.med	sci.space
pred	sci.med	244	19
	sci.space	8	224

3. Supongamos que tenemos los datos bidimensionales de clasificación binaria. Para la Figura 1 (a) es suficiente hacer una transformación polinomial de grado 2, i.e. es suficiente usar un kernel polinomial de grado 2

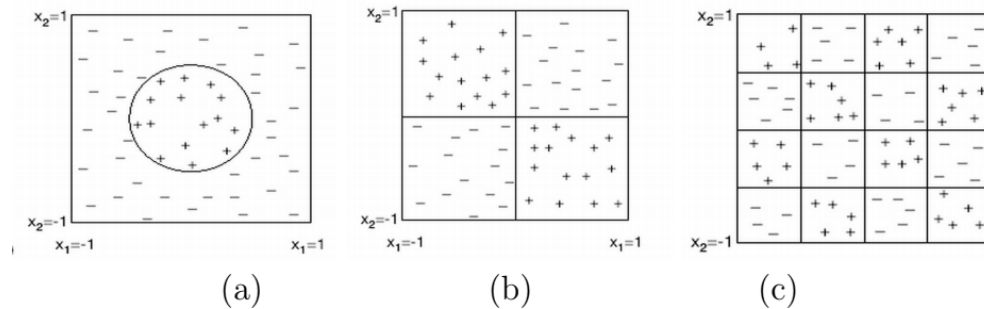
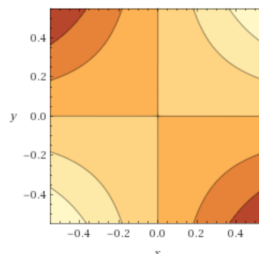


Figura 1

¿De qué grado mínimo debe ser el kernel polinomial para los datos de la Figura 1(b) y de la Figura 1(c)?

Para la 1(b), basta con una función polinomial de grado dos de la forma $4x^2 - y^2 + \alpha xy$ con un α grande, lo cual nos daría las curvas de nivel:



Para la 1(c), basta con una función polinomial de grado 6. es más difícil graficar una curva de nivel que cumpla con esto, pero el razonamiento es:

En la diagonal más grande tenemos cuatro bloques de una categoría. Para que un polinomio que siempre sea positivo (o negativo) pase por ahí con ceros en los puntos de cruce, este necesitaría cinco cambios de dirección, por lo que el polinomio es de grado 6. Si se agregara otra fila y columna de este mismo modo, el polinomio sería de grado 8.

Otra forma de ver el grado del polinomio es contando los cambios de categorías sobre x_1 y x_2 , el grado del polinomio será igual a la suma de los cambios en $x_1 + x_2$