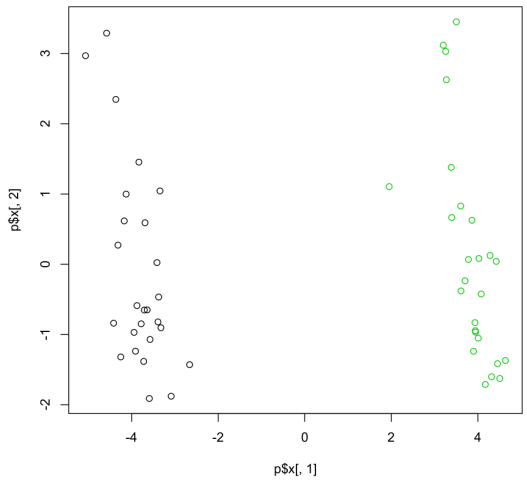
Tarea 7, Reconocimiento de Patrones

Francisco Javier Peralta Ramírez

1. Obtuvimos información de google maps usando la librería *gmapdistance* de tal forma que pudiéramos ver la diferencia de tráfico de norte a sur en la ciudad de México a la hora de salida de trabajo. Para esto se tuvo que cosechar datos durante cuatro horas de 5pm a 9pm haciendo queries cada 10 minutos. Se tomaron diez puntos en el mapa de tal forma de que cinco estuvieran en la zona norte y los otros cinco en la zona sur de la ciudad.

   Cada query regresa con veinticinco datos (la combinación de los cinco puntos norte con cinco sur), estos se pasan a un vector y se agregan a un data frame junto con el tiempo del query y su categoría $y \in \{-1, 1\}$ donde -1 corresponde *norte a sur*. Un dato importante a considerar es el orden de la matriz que se obtiene al hacer un query, para que los tiempos correspondan a los mismos puntos, la matriz al hacer el query *sur a norte* se transpuso.

   Teniendo los datos, aplicamos PCA para ver cómo se comportan al reducir la dimensión de 26 a 2 y no muy sorprendentemente se comportan bien. Esto se debe a que claramente la cantidad de tráfico no es igual en ambas direcciones.



   Como se puede observar, al aplicar PCA los datos son claramente linealmente separables por lo que decidimos usar una SVM con kernel lineal para clasificar. El modelo converge con doce vectores de soporte, el cual es algo alto considerando que sólo tenemos 52 datos.

2. Tomando los datos de un estudio de aire en Noruega, para determinar la calidad del aire y ver si la contaminación no rebasa $50\mu g/m^3$, creamos múltiples modelos regresión logística y redes neuronales para comprar su desempeño.

   Tomamos las funciones dadas por

```
f1 <- "highpm10 ~ cars+temp2m+winddirection+time"
f2 <- "highpm10 ~ cars+temp2m"
f3 <- "highpm10 ~ cars*temp2m"
```

   Obtenemos los resultados:

	Correcto	
	Reg-Log	N-Net
f1	0.752	0.752
f2	0.744	0.752
f3	0.736	0.744

Cuando se corrieron las pruebas múltiples veces los resultados tienen a ser similares, por lo que no podemos decir que un método tiene mejor desempeño que el otro. Cabe notar que la red neuronal puede converger en diferentes puntos, y no siempre tener el mejor desempeño posible.

3. Extendiendo el ejercicio de la tarea 7, ahora usamos un modelo de regresión logística para clasificar los correos como *SPAM* o *no SPAM* y se compararon los resultados con *boostig* y *random forests*.

   En este caso tomamos un modelo completo es decir $f = x_1 + x_2 + \cdots + x_n$ ya que hacer combinaciones de todos los modelos posibles sería muy pesado.

Cuadro 1: Random Forest

		classs	
		0	1
pred	0	695	153
	1	9	294

Cuadro 2: Boosting

		classs	
		0	1
pred	0	679	26
	1	25	421

Cuadro 3: Regresión Logística

		classs	
		0	1
pred	0	667	56
	1	37	391

Podemos ver que Random forest es el que mejor clasifica el correo no spam, pero es el que más spam deja pasar, por otra parte lo contrario pasa con boosting, y como es de esperar regresión logística es un punto intermedio. Podríamos modificar el porcentaje de "aceptación" para dejar pasar más SPAM y rechazar menos correo bueno, haciendo el modelo de regresión logística bastante usable.

4. Usando datos de un estudio sobre la relación entre tomar la medicina AZT, la raza del paciente y mostrar síntomas de SIDA, buscamos un modelo re regresión logística adecuado que relaciona la probabilidad de mostar síntomas de SIDA con taomar AZT y la raza.

		Symptoms	
Race	AZT Use	Yes	No
White	Yes	14	93
	No	32	81
Black	Yes	11	52
	No	12	43

Source: New York Times, Feb. 15, 1991.

Generamos todos los modelos posibles

```
f1 <- "cbind(Y, N) ~ Race + AZT"
f2 <- "cbind(Y, N) ~ Race * AZT"
f3 <- "cbind(Y, N) ~ Race"
f4 <- "cbind(Y, N) ~ AZT"
```

haciendo uso de la función *summary* notamos que *azt* siempre es significativo. También podemos ver que por el “Residual deviance” el modelo que sólo toma en cuenta AZT es el mejor.