Universidad Nacional de Rosario

Introducción al Aprendizaje Automatizado

Trabajo Práctico Final

Villagra Martín

21 de julio de 2017

Introducción

Este trabajo analiza el desempeño del clasificador de Support Vector Machines.

Ejercicio a

Se utilizó el lenguaje Python con los bindings disponibles de libsvm. La implementación se encuentra en el archivo *ssvm.py*. Se escalaron los atributos al rango $[0, 1]$ ya que es una práctica recomendada. Para implementar los 10-folds, se uso el script *split_dataset.py* para dividir el dataset en las partes requeridas. Se escogió utilizar una función de base radial (RBF) como kernel no lineal debido a su popularidad.

Elección de parámetros

Para elegir el parámetro adecuado (C para el kernel lineal y C y γ para RBF se probó con valores con crecimiento exponencial $(2^{-5}, 2^{-4}, 2^{-3}, ..., 2^{15})$ eligiendo el de menor error. Luego se probaron valores cercanos a este para ajustar con más precisión el valor. Notar que para RBF es necesario confeccionar una tabla con los errores, en las columnas varia el C y en las filas varia el γ. Este procedimiento si bien es simple nos da la certeza de que estamos explorando todos el espectro de valores posibles.

Resultados

Para el kernel lineal se utilizo $C = 10$, mientras que para RBF se fijo $C = 2^{-5} y \gamma = 2^5$. Los errores se muestran en la Tabla 1.

Método	Media	Desv. Est.
SVM - kernel lineal	19.80 %	4.57
SVM - kernel RBF	20.00 %	4.00
Naive Bayes	20.80 %	7.07
Decision Trees	23.60 %	5.64

Tabla 1: Comparación de los Métodos según el error en test entre los 10-folds.

Sorprendentemente vemos que el kernel lineal tuvo un mejor desempeño que el RBF, pero esta diferencia parece despreciable. A su vez Naive Bayes tiene una media similar pero su desviación estándar es mucho mayor, indicando que el método suele ser más sensible a la elección del conjunto de entrenamiento. Notar que el más confiable es el RBF por tener menos desviación estándar.

Ejercicio b

Para realizar el método se utilizó una simple hoja de cálculo disponible en: `https://docs.google.com/spreadsheets/d/1Ah1F7PqSIchT1ryZB9Gq8pgFIZXv_t5gRXrwD9C60e4/edit?usp=sharing`. Una vista previa de la misma se muestra en la Tabla 2.

	SVM LINEAR	SVM RBF	NB	DT		RBF-LINEAR	NB-SVM		Samples	10
	22	22	24	28		0	2		t_{95%, 9}	2.26
	16	18	16	24		2	-2			
	22	14	26	28		-8	12			
	12	20	6	18		8	-14			
	20	20	22	24		0	2			
	20	18	20	16		-2	2			
	18	20	30	20		2	10			
	24	24	24	26		0	0			
	16	16	14	18		0	-2			
	28	28	26	34		0	-2			
Media	19.8	20	20.8	23.6		0.2	0.8			
desv. est.	4.57	4.00	7.07	5.64		3.94	7.13			
				Estimación desv.		1.25	2.25			
				Incerteza (t-test)		2.81	5.10			

Tabla 2: Hoja de cálculo con los resultados del t-test.

Se puede observar que entre el método SVM con kernel lineal y Decision Trees la diferencia media es $0{,}8\,\% \pm 5{,}10$, el signo indica que efectivamente SVM es superior, pero no por mucho y peor aún la incerteza es demasiado grande como para poder tener en cuenta este resultado. Comparando SVM bajo el kernel RBF y el lineal tenemos un resultado de $0{,}2\,\% \pm 2{,}81$, su proximidad al cero y su incerteza alta impiden una vez más establecer una superioridad entre uno y otro. Estos valores son comprensibles dado que para algunos folds particulares el "peor" método resulta ser mejor que el "mejor" (valores negativos en la Tabla 2).

Como conclusión no podemos asegurar que ningún método sea mejor que otro para este dataset.