Comentarios sobre la práctica de support vector machines

Detección de spam

- Corpus:
  - spam.zip: 500 mensajes
  - easy_ham.zip: 2551 mensajes
  - hard_ham.zip: 250 mensajes

- Objetivo:
  Evaluar distintas configuraciones del sistema de aprendizaje, incluyendo qué mensajes se usan para entrenamiento y cuáles para evaluación

- Representación de los mensajes:

  mensaje y vocabulario → mensaje procesado → vector de 0s y 1s

$x^{(i)}$ es un vector de 1899 (palabras del vocabulario) componentes de 0s y 1s, ¿de dónde sacamos $y^{(i)}$?

Procesamiento de los mensajes

- Lectura de mensaje (el nombre de fichero se puede generar con `format`)

```
email_contents = open( 'spam/0001.txt', 'r' ).read()
```

```
...
Más líneas de cabecera
...
<CENTER>Save up to 70% on Life Insurance.</CENTER></FONT><FONT color=3D#ff=
0000
face=3D"Copperplate Gothic Bold" size=3D5 PTSIZE=3D"10">
<CENTER>Why Spend More Than You Have To?
<CENTER><FONT color=3D#ff0000 face=3D"Copperplate Gothic Bold" size=3D5 PT=
SIZE=3D"10">
<CENTER>Life Quote Savings
...
If you reside in any state which prohibits e-mail solicitations for insuran=
ce, please disregard this email.<BR></
FONT><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR=
><BR><BR><BR></FONT></P></CENTER></CENTER></TR></TBODY></TABLE></CENTER></=
CENTER></CENTER></CENTER></CENTER></BODY></HTML>
```

- Procesamiento del mensaje

```
tokens  = email2TokenList(email_contents);
```

```
['save', 'up', 'to', 'number', 'on', 'life', 'insur', 'whi','spend', 'more', 'than',
'you', 'have', 'to', 'life', 'quot', 'save',
...
'pleas', 'disregard', 'thi', 'email']
```

Procesamiento de corpus

- Corpus:
  - spam.zip: 500 mensajes
  - easy_ham.zip: 2551 mensajes
  - hard_ham.zip: 250 mensajes

- Objetivo:
  Evaluar distintas configuraciones del sistema de aprendizaje, incluyendo qué mensajes se usan para entrenamiento y cuáles para evaluación

```
directorio = "spam"
i = 1
email_contents = codecs.open(
            '{0}/{1:04d}.txt'.format(directorio, i), 'r',
            encoding='utf-8', errors='ignore').read()
```

Construcción de array por filas

```
In [15]: X = np.empty((0, 5))

In [20]: for i in range(3):
    ...:     X = np.vstack((X, np.ones(5)*i))
    ...:

In [21]: X
Out[21]:
array([[0., 0., 0., 0., 0.],
       [1., 1., 1., 1., 1.],
       [2., 2., 2., 2., 2.]])
```