Complex Support Vector Machines for Regression and Quaternary Classification

P. Bouboulis, S. Theodoridis, C. Mavroforakis, L. Dalla Learning (MKL) 19, 2010 (?)
(I got it from arXiv)
Classification of complex data

• Classifying bird songs:

  ![signal](image1)

  ![spectrogram](image2)

  • But... the spectrogram contains **complex** values!
  • Can I classify a complex-valued image?
Complex support vector machine

• Inputs are complex vectors: $\tilde{x} = x^r + ix^i$
• Output can be:
  • a complex number (regression)
  • a class label (classification)

• To output a label:
  For real valued SVM:
  $$f(x) = \langle w, x \rangle + b \begin{cases} 
  \geq 0 & \text{assign to } \omega_1 \\
  < 0 & \text{assign to } \omega_2 
\end{cases}$$

  For complex SVM:
  $$f(x) = \langle \tilde{w}, \tilde{x} \rangle_{\mathbb{H}} + b$$


donderdag 1 mei 2014
Complex inner product

- You can define inner product in different ways:
- Simple:
  \[ \langle \tilde{w}, \tilde{x} \rangle_{\mathbb{H}} = \langle w^r, x^r \rangle + \langle w^i, x^i \rangle \]
  - Just a concatenation of the real and imaginary spaces.
  - You can use the same SVM as always...

- Complex:
  \[ \langle \tilde{w}, \tilde{x} \rangle_{\mathbb{H}} = \langle w^r, x^r \rangle + \langle w^i, x^i \rangle + i \left( \langle w^i, x^r \rangle - \langle w^r, x^i \rangle \right) \]
  - Now the output is not so clear:
    \[ \text{sign} \left( \langle \tilde{w}, \tilde{x} \rangle_{\mathbb{H}} + b \right) \]
    - sign of a complex number?
Not 2-class, but 4-class classifier!

Now the classifier can distinguish FOUR classes:

\[ \text{sign} \left( \text{Re}(\langle \tilde{w}, \tilde{x} \rangle_{H} + b) \right) \geq 0 \]

\[ \text{sign} \left( \text{Im}(\langle \tilde{w}, \tilde{x} \rangle_{H} + b) \right) \geq 0 \]
Further things in the paper...

• To find the Quadratic program (or, find derivatives of real-valued functions in a complex space), use “Wirtinger’s calculus (it appears you can write it as two independent quadratic programming problems)

• Derive for both regression as classification

• Propose kernel mappings for complex data: $\Phi_C(\tilde{x})$
For your information:

- It looks like:

\[
\text{maximize } \sum_{n=1}^{N} a_n - \sum_{n,m=1}^{N} a_n a_m d_n^r d_m^r \kappa_C^r(z_m, z_n) \\
\text{subject to } \begin{cases} \\
\sum_{n=1}^{N} a_n d_n^r = 0 \\
0 \leq a_n \leq \frac{C}{N} \\
\text{for } n = 1, \ldots, N 
\end{cases}
\]

and

\[
\text{maximize } \sum_{n=1}^{N} b_n - \sum_{n,m=1}^{N} b_n b_m d_n^i d_m^i \kappa_C^r(z_m, z_n) \\
\text{subject to } \begin{cases} \\
\sum_{n=1}^{N} b_n d_n^i = 0 \\
0 \leq b_n \leq \frac{C}{N} \\
\text{for } n = 1, \ldots, N. 
\end{cases}
\]

- Labels

\( d_n = \pm 1 \pm i \)

(four classes)

- QPs for the real and the complex part of the label
Conclusion

• Treating the real and imaginary part of $\tilde{x} = x^r + ix^i$ as independent features is OK!

• In the fully complex treatment, we get a four-class classifier

• ... and we can define new kernels

![Fig. 1. The element $\kappa^C(\cdot, (0, 0)^T)$ of the induced real feature space of the complex Gaussian kernel.](image)