Sparse Kernel Machines - RVM

Henrik I. Christensen

Robotics & Intelligent Machines @ GT
Georgia Institute of Technology,
Atlanta, GA 30332-0280
hic@cc.gatech.edu

Outline

Introduction

- We discussed memory based methods earlier
- Sparse methods are directed at memory based systems with minimum (but representative) training samples
- Last time we talked about support vector machines
- A few challenges - ie., multi-class classification
- What we could be more Bayesian in our formulation?

Outline

Regression model

- We are seen continuous / Bayesian regression models before

$$p(t|\mathbf{x}, \mathbf{w}, \beta) = N(t|y(\mathbf{x}), \beta^{-1})$$

- We have the linear model for fusion of data

$$y(\mathbf{x}) = \sum_{i=1}^{N} w_i \phi_i(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x})$$

- A relevance vector formulation would then be:

$$y(\mathbf{x}) = \sum_{i=1}^{N} w_i k(\mathbf{x}, \mathbf{x}_i) + b$$

The collective model

- Consider N observation vectors collected in a data matrix $\mathbf{X}$ where row i is the data vector $\mathbf{x}_i$. The corresponding target vector $\mathbf{t} \in \{t_1, t_2, ..., t_N\}$ the likelihood is then:

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{i=1}^{N} p(t_i|\mathbf{x}_i, \mathbf{w}, \beta^{-1})$$

- If we consider weights to be zero-mean Gaussian we have

$$p(\mathbf{w}|\alpha) = \prod_{i=0}^{N} N(w_i|0, \alpha^{-1})$$

- ie we have different uncertainties/precision for each factor

More shuffling

- Reorganizing using the results from linear regression we get

$$p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \alpha, \beta) = N(\mathbf{w}|\mathbf{m}, \mathbf{\Sigma})$$

where

$$\begin{aligned} \mathbf{m} &= \beta \mathbf{\Sigma} \mathbf{\Phi}^T \mathbf{t} \\ \mathbf{\Sigma} &= \left(\mathbf{A} + \beta \Phi^T \Phi \right)^T \end{aligned}$$

where Φ is the design matrix and $\mathbf{A} = diag(\alpha_i)$. In many cases the design matrix is the same as the GRAM matrix i.e. $\Phi_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$.

Estimation of α and β

- Using maximum likelihood we can derive estimates for α and β. We can integrate out $\mathbf{w}$

$$p(\mathbf{t}|\mathbf{X}, \alpha, \beta) = \int p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta)p(\mathbf{w}|\alpha)d\mathbf{w}$$

- The log likelihood is then

$$\begin{aligned}
\ln p(\mathbf{t}|\mathbf{X}, \alpha, \beta) &= \ln N(\mathbf{t}|0, \mathbf{C}) \\
&= -\frac{1}{2}\left\{N \ln(2\pi) + \ln|\mathbf{C}| + \mathbf{t}^T\mathbf{C}\mathbf{t}\right\}
\end{aligned}$$

- where

$$\mathbf{C} = \beta^{-1}\mathbf{I} + \mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^T$$

Re-estimation of α and β

- We can then re-estimate α and β from

$$\begin{aligned} \alpha_i^{new} &= \frac{\gamma_i}{m_i^2} \\ (\beta^{new})^{-1} &= \frac{||\mathbf{t} - \mathbf{\Phi m}||^2}{N - \sum_i \gamma_i} \end{aligned}$$

- where γ_i are precision estimates defined by

$$\gamma_i = 1 - \alpha_1 \Sigma_{ii}$$

- the precision will go to zero for some of these - ie. very large uncertainty and the corresponding α values will go to zero.
- In the sense of an SVM the training data becomes irrelevant.

Regression for new data

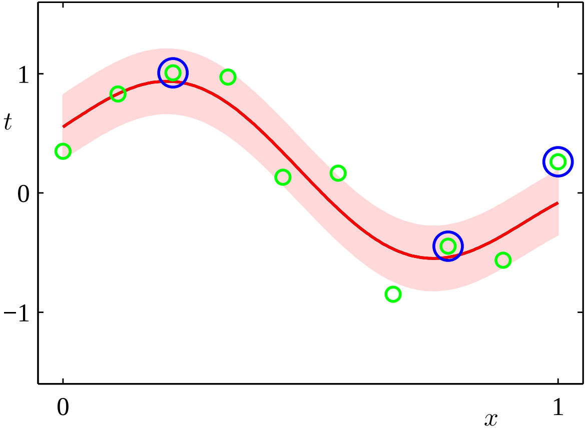
- Once hyper parameters have been estimated regression can be performed

$$p(t|\mathbf{x}, \mathbf{X}, \mathbf{t}, \alpha^*, \beta^*) = N(t|\mathbf{m}^T\phi(\mathbf{x}), \sigma^2(\mathbf{x}))$$

where

$$\sigma^2(\mathbf{x}) = (\beta^*)^{-1} + \phi(\mathbf{x})^T\mathbf{\Sigma}\phi(\mathbf{x})$$

Illustrative example



Status

- Relevance vectors are similar in style to support vectors
- Defined within a Bayesian framework
- Training requires inversion of an $(N+1) \times (N+1)$ matrix which can be (very) costly
- In general the resulting set of vectors is much smaller
- The basis functions should be chosen carefully for the training. Ie. analyze your data to fully understand what is going on.
- The criteria function is no longer a quadratic optimization problem, and convexity is not guaranteed.

Analysis of sparsity

- There is a different way to estimate the parameters that is more efficient. I.e brute force is not always optimal
- The iterative estimation of α poses a challenge, but does suggest an alternative. Consider a rewrite of the $\mathbf{C}$ matrix

$$
\begin{aligned}
\mathbf{C} &= \beta^{-1}\mathbf{I} + \sum_{j \neq i} \alpha_j^{-1}\phi_j\phi_j^T + \alpha_i^{-1}\phi_i\phi_i^T \\
&= C_{-i} + +\alpha_i^{-1}\phi_i\phi_i^T
\end{aligned}
$$

- I.e. we have made the contribution of the i'th term explicit.
- Standard linear algebra allow us to rewrite

$$
\begin{aligned}
det(\mathbf{c}) = |\mathbf{C}| &= |\mathbf{C}_{-i}||1 - +\alpha_i^{-1}\phi_i^T\mathbf{C}_{-i}^{-1}\phi_i| \\
\mathbf{C}^{-1} &= \mathbf{C}_{-i}^{-1} - \frac{\mathbf{C}_{-i}^{-1}\phi_i\phi_i^T\mathbf{C}_{-i}^{-1}}{\alpha_i + \phi_i^T\mathbf{C}_{-i}^{-1}\phi_i}
\end{aligned}
$$

The seperated log likelihood

- This allow us to rewrite the log likelihood

$$L(\alpha) = L(\alpha_{-i}) + \lambda(\alpha_i)$$

- The contribution of alpha is then

$$\lambda(\alpha_i) = \frac{1}{2}\left[\ln \alpha_i - \ln(\alpha_i + s_i) + \frac{q_i^2}{\alpha_i + s_i}\right]$$

- Here we have the complete dependency on α_i
- We have used

$$\begin{aligned} s_i &= \phi_i^T \mathbf{C}_{-i}^{-1} \phi_i \\ q_i &= \phi_i^T \mathbf{C}_{-i}^{-1} \mathbf{t} \end{aligned}$$

s_i is known as the sparsity and q_i is known as the quality of ϕ_i

Evaluation for stationary conditions

- It can be shown (see Bishop pp. 351-352)
- if $q_i^2 > s_i$ then there is a stable solution

$$\alpha_i = \frac{s_i^2}{q_i^2 - s_i}$$

- otherwise α_i goes to infinity == irrelevant

Status

- There are efficient (non-recursive) ways to evaluate the parameters.
- The relative complexity is still significant.

Outline

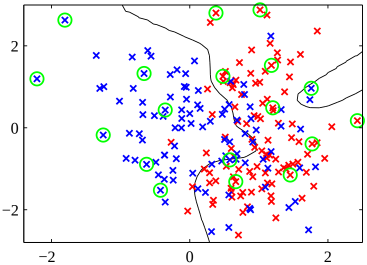
Relevance vectors for classification

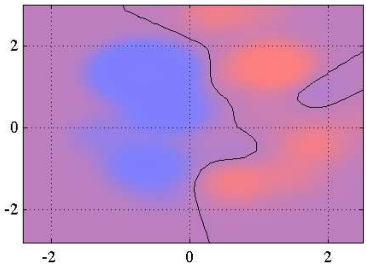
- For classification we can apply the same framework
- Consider the two class problem with binary targets $t \in \{0, 1\}$ then the form is

$$y(\mathbf{x}) = \sigma(\mathbf{w}^t \phi(\mathbf{x}))$$

- where $\sigma(.)$ is the logistic sigmoid function
- Closed form integration is no longer an option
- We can use the Laplace approach to estimate the mode and which in turn allow estimation of weights (α) and in term re-estimate the mode and then new values for α until convergence.
- The process is similar to regression (see book)

Synthetic example





Outline

Summary

- A Bayesian approach to definition of a sparse model
- The model is more comprehensive / but also with more assumptions
- Creates sparser model with 'similar' performance
- Training can be slow - especially for large data-sets
- Execution is faster due to a sparser model
- Selection of basis functions for relevance vectors can pose a challenge.

Projects

- Halfway through the course!
- Covered the basics
- Next Monday & Wednesday - IROS-09
- Next Friday - Update on projects
  - What is your problem?
  - What is your approach?
  - How will you train the system?
  - How will you evaluate performance?