A first order interior point algorithm for solving continuous *Quadratic* Knapsack problems.

María D. González-Lima

Universidad Militar Nueva Granada, Colombia.

Esnil Guevara

Universidad de Carabobo, Venezuela

First Brazilian Workshop on Interior Point Methods. Campinas, April 27-28, 2015

Outline

- Problem and Optimality conditions

- Algorithm.

- Numerical Experimentation: SVM problems. Identification procedure.

- Concluding Remarks

- Comments on SVM (according to time)

- ANNEX: Tables.

Continuous Knapsack Problem

$$\boxed{\begin{array}{rl} \min & \frac{1}{2}x^T Q x - e^T x \\ \text{s.t.} & a^T x = d \\ & 0 \le x \le C \end{array}}$$

with $Q \in \Re^{n \times n}$ positive semidefinite, $a, x, e \in \Re^n$, $d \in \Re$.

Special interest:

Q large and dense.

Why?

Applications: Support Vector Machines (pattern recognition, classification problems, etc)

First order (No second information - "'Matrix Free"') + Interior point methods: INEXACT NEWTON DIRECTION

Many contributions .....

- Iterative (Factorization !)
- Quasi-Newton

Optimality conditions K-K-T:

$$F(x,t,u,w,s) = \begin{pmatrix} Qx - e - sa - t + u \\ a^T x \\ C - w - x \\ XTe \\ UWe \end{pmatrix} = 0$$

$(x,t,u,w) \geq 0$

where X, T, U, W are diagonal matrices with the vectors x, t, u, w in the diagonal, respectively.

Algorithm

Iterative method that generates points $(x, t, u, w) > 0$ such that they approach the optimality conditions.

Search direction $d_k = (\Delta x_k, \Delta t_k, \Delta u_k, \Delta w_k)$:
A perturbed-"'Quasi-Newton"': $Q \approx \lambda I$

$$\tilde{F}' = \begin{pmatrix} \lambda I & -a & -I & I & 0 \\ a^T & 0 & 0 & 0 & 0 \\ -I & 0 & 0 & 0 & -I \\ T & 0 & X & 0 & 0 \\ 0 & 0 & 0 & W & U \end{pmatrix}$$

$\tilde{F^k}' d_k = -F^k + \begin{pmatrix} 0 \\ \mu_k e \\ \mu_k e \end{pmatrix}$ and μ_k going to zero.

LOW COST INTERIOR POINT METHOD

Given $(x^0, t^0, u^0, w^0) > 0, s^0$.
For $k = 0, 1, \ldots$ until convergence

Step 1. Choose $\sigma_k \in (0,1)$, compute $\mu_k = \frac{(x^k)^T t^k + (u^k)^T w^k}{2n}$ and residuals

$$
\begin{aligned}
r_e^k &= Qx^k - e - s^k a - t^k + u^k \\
r_b^k &= a^T x^k - d \\
r_c^k &= C - w^k - x^k \\
r_{tx}^k &= T^k X^k e - \sigma_k \mu_k e \\
r_{uw}^k &= U^k W^k e - \sigma_k \mu_k e
\end{aligned}
$$

Step 2. Compute $\lambda_k > 0$ and find search directions

$$
\begin{aligned}
\Delta s^k &= -\frac{a^t (D^k)^{-1} b_x + r_b}{a^t (D^k)^{-1} a} \\
\Delta x^k &= (D^k)^{-1}(b_x + a\Delta s^k) \\
\Delta w^k &= r_c - \Delta x^k \\
\Delta t^k &= (X^k)^{-1}\left(-r_{tx}^k - T^k \Delta x^k\right) \\
\Delta u^k &= (W^k)^{-1}\left(-r_{uw}^k - U^k \Delta w^k\right)
\end{aligned}
$$

with $b_x = -r_e - (X^k)^{-1} r_{tx} + (W^k)^{-1}\left(r_{uw} + U^k r_c\right)$,
$D^k = \lambda_k I + (X^k)^{-1} T^k + (W^k)^{-1} U^k$.

Step 3. Find $\alpha_k > 0$ such that

$(x^k, s^k, t^k, u^k, w^k) + \alpha_k(\Delta x^k, \Delta s^k, \Delta t^k, \Delta u^k, \Delta w^k) > 0$.

Step 4. Choose $\eta \in (0, 1)$ and update

$$(x^{k+1}, s^{k+1}, t^{k+1}, u^{k+1}, w^{k+1}) =$$

$(x^k, s^k, t^k, u^k, w^k) + \eta\alpha_k(\Delta x^k, \Delta s^k, \Delta t^k, \Delta u^k, \Delta w^k)$.

Observations:

- $(x^k, s^k, t^k, u^k, w^k) > 0$
- Primal feasible if $r_b^0 = 0$.
- **Dual infeasible**.
- d_k satisfies $F^{k\prime} d_k + F^k = \begin{pmatrix} (Q - \lambda_k I)\Delta x^k \\ 0 \\ 0 \\ \mu_k \\ \mu_k \end{pmatrix}$:

  Inexactness !
- If $r_b^k = 0$ then $\Delta x^k = -(D^k)^{-1} P^k b_x$ with $P_k = I - \frac{aa^T (D^k)^{-1}}{a^T (D^k)^{-1} a}$ a projection matrix.

Choice of λ_k (Spectral Gradient: Barzilai and Borwein '88,....)

$\lambda_k = \arg\min_{\lambda \geq \lambda_0} \|(Q - \lambda I)\Delta^{k-1}\|$ (Secant Equation)

$$\lambda_k = \frac{(x^k - x^{k-1})^T Q (x^k - x^{k-1})}{(x^k - x^{k-1})^T (x^k - x^{k-1})}$$

Numerical Experiments:

SVM Problem:

- $Q_{ij} = a_i a_j K(w_i, w_j) \in \Re^{m \times n}$ (n: number of data points (dimension); m: dimension of points (attributes)).

- $a_i = 1$ or -1, e the vector of all ones, $d = 0$.

- Kernels: 1. (Lineal): $K(x, y) = x^T y$, 2. (Polynomial): $K(x, y) = (\frac{1}{m} x^T y + 1)^3$, 3. (Gaussian): $K(x, y) = \exp(-\frac{1}{m}\|x - y\|^2)$.

- C=1,5, 10.

- Tested Problems: ($m > n$) 13: 107 instances, ($m \leq n$) 19: 171 Instances.

Many methods for SVM: SOM (Platt '99), LIBSVM (Fan, Chen, Lin 06'), SVM^{light} (Joachims '1998), GPDT (Serafini, Zanguireti, Zanni 03'), ...

We compared to ASL (Gonzalez-Lima, Hager, Zhang; SIOPT 2011):

The ASL is an affine scaling primal-feasible interior point algorithm that also uses the spectral length to "'approximate"' the Hessian. A nonlinear equation is solved at each iteration.

Competitive to ASL when $m >>> n$ (CPU times are almost the same) .

.......$>$ Identification technique using the advantage of being working with primal and dual points.

Identification of zero variables (El-Bakry, Tapia, Zhang '94; Facchinei, Fischer, Kansow '98 ... Jung, O'Leary, Tits '2008 ...

$$I_i = \frac{t_i}{x_i} + \frac{u_i}{w_i}$$

If I_i is large, $x_i \to 0$ (is not a support vector) or C (bounded support vector).
If I_i is small, $0 < x_i < C$ (support vector)

When iterates are "'close"' to the solution set, compute an estimate q of the number of support vectors:

$$|\{i : I_i^{-1} \geq 100\sqrt{\mu}\}| \leq q \leq |\{i : \frac{x_{i+1}}{x_i} > 0.9\}|$$

The candidate to support vectors are the q smaller components of vector I.

We also consider a predictor-corrector direction (Predictor direction helps to identify the variables)

....> LC-I ALGORITHM.

Number of problems ($m < n$) where each method wins or gets tie.

Kernel	ASL	LC-I	TIE
Polynomial	40	3	14
Gaussian	8	25	24
Linear	45	1	11

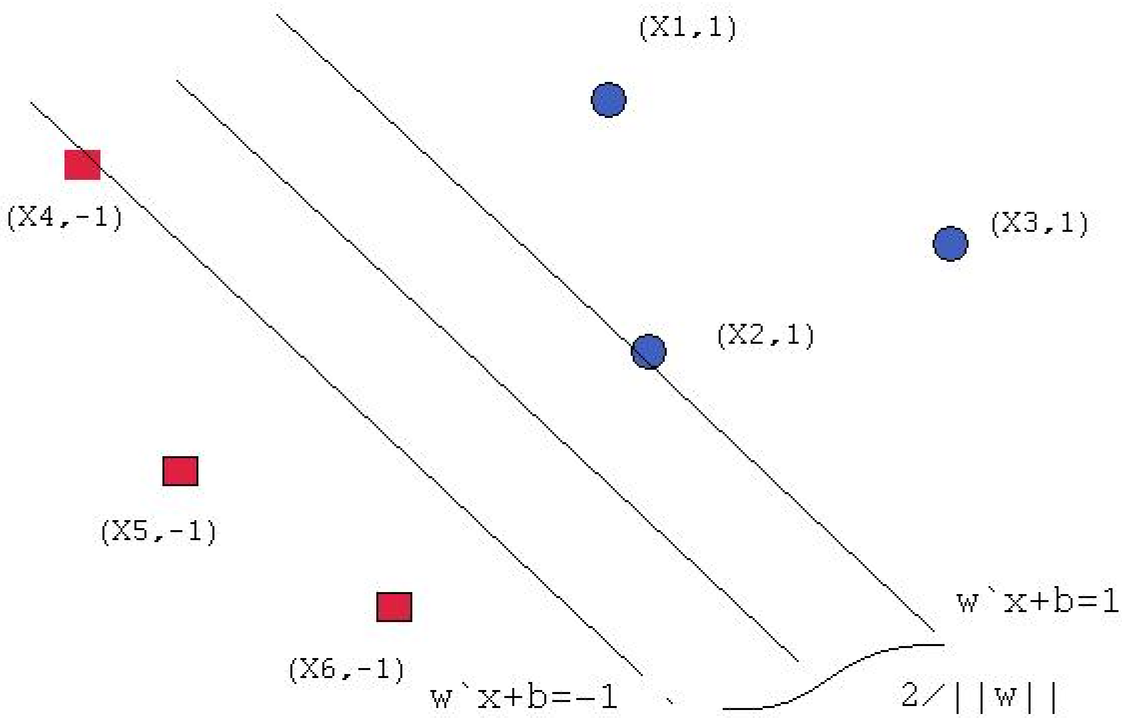
Concluding Remarks

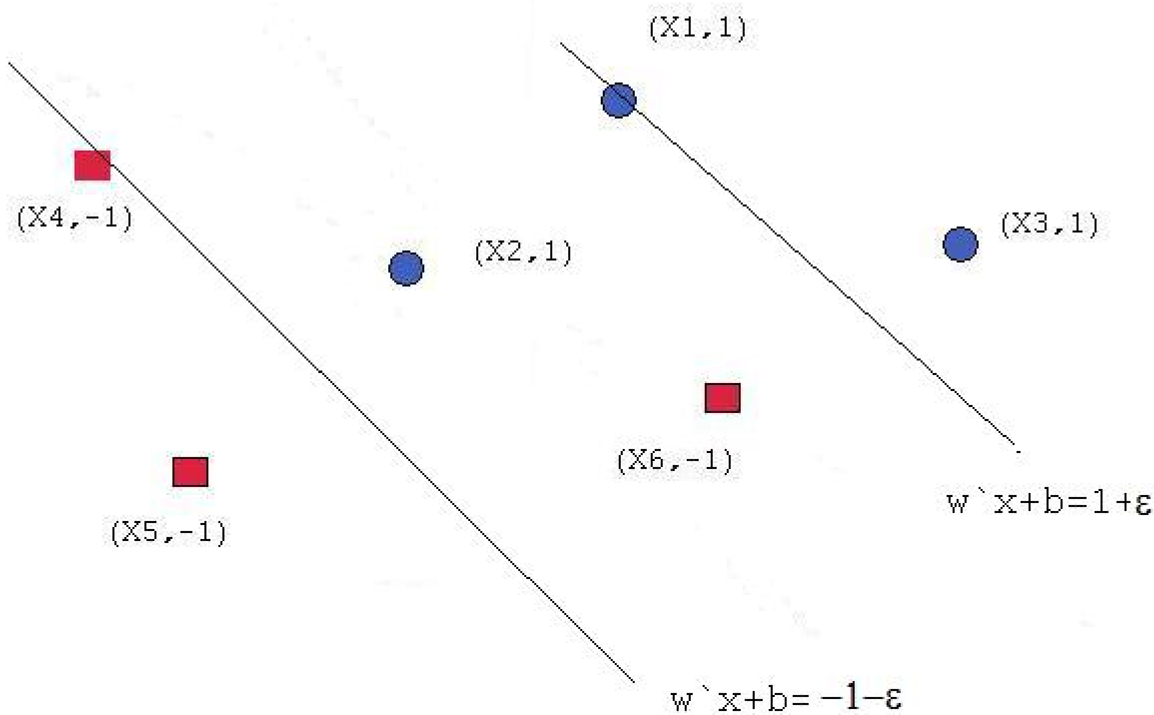
- A first order interior primal-feasible for the continuous quadratic convex Knapsack problem was presented.

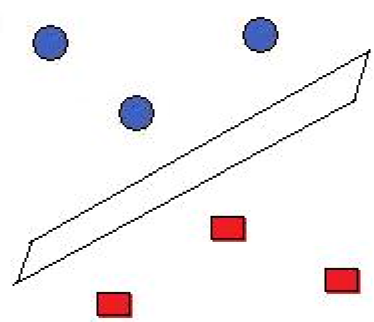
- The method seems to be competitive to ASL (Affine Scaling) for solving SVM real life problems specially when there are more attributes than dimensions or the Kernel is Gaussian.

- Theory and more numerical experiments is part of future research.

SVM ?




(X1,1)
(X4,-1)
(X2,1)
(X3,1)
(X6,-1)
(X5,-1)
w`x+b=1+ε
w`x+b= −1−ε



Example: "Mammographic Mass Data Set"

The prediction obtained from a mammography related to breast cancer is not good and about %70 of them are not necessary because the tumor are benign. It would be of benefit to develop an automatized system that helps the doctor to take the decision of pursuing biopsy or not. This can be done by SVM.

Goal: Using the data from a group of women with tumors, the idea is to classify in benign or not.

n= number of patients.

m= attributes = age, shape, density, ...

Problem			Time (seconds)	
a1a	**kernel**	**C**	**ASL**	**LC-I**
data(n)/attributes(m) 1605 /123	Linear	1	0.2	0.57
		5	0.8	2.56
		10	2.4	3.58
	Polynomial	1	0.3	-
		5	0.3	0.5
		10	0.3	0.5
	Gaussian	1	0.5	0.27
		5	0.6	0.43
		10	0.7	0.46
a2a	**kernel**	**C**	**ASL**	**LC-I**
2265/123	Linear	1	0.4	1.11
		5	1.9	8.11
		10	4.05	13.32
	Polynomial	1	0.6	-
		5	0.6	0.67
		10	0.7	0.75
	Gaussian	1	1.1	0.48
		5	1.3	0.83
		10	1.4	0.91
a3a	**kernel**	**C**	**ASL**	**LC-I**
3185/123	Linear	1	0.8	3.54
		5	3.1	16.54
		10	5.7	36.21
	Polynomial	1	1.2	-
		5	1.3	0.97
		10	1.5	1.39
	Gaussian	1	2.3	1.06
		5	2.6	1.62
		10	2.9	3.86
a4a	**kernel**	**C**	**ASL**	**LC-I**
4781/123	Linear	1	1.5	7.99
		5	7.5	58.09
		10	12.56	73.81
	Polynomial	1	2.8	-
		5	2.9	2.25
		10	3.6	3.26
	Gaussian	1	5.3	2.22
		5	6.2	4.46
		10	7.9	8.80

australian_scale	**kernel**	**C**	**ASL**	**LC-I**
	Linear	1	0.4	0.07
		5	1.7	1.03
		10	3.1	0.85
552/14	Polynomial	1	0.04	0.06
		5	0.1	0.06
		10	0.1	0.07
	Gaussian	1	0.08	0.07
		5	0.1	0.10
		10	0.1	0.22
breast-cancer_scale	**kernel**	**C**	**ASL**	**LC-I**
	Linear	1	0.02	0.05
		5	0.08	0.08
		10	0.1	0.08
547/10	Polynomial	1	0.02	0.04
		5	0.04	0.05
		10	0.1	0.04
	Gaussian	1	0.04	0.04
		5	0.06	0.06
		10	0.07	0.08
diabetes_scale	**kernel**	**C**	**ASL**	**LC-I**
	Linear	1	0.04	0.08
		5	0.2	0.19
data/dimension		10	0.2	0.29
615/8	Polynomial	1	0.02	0.05
		5	0.03	0.06
		10	0.03	0.07
	Gaussian	1	0.06	0.06
		5	0.08	0.1
		10	0.1	0.18
fourclass_scale	**kernel**	**C**	**ASL**	**LC-I**
	Linear	1	0.03	0.07
		5	0.1	0.21
		10	0.02	0.31
690/2	Polynomial	1	0.04	0.15
		5	0.08	0.39
		10	0.1	0.33
	Gaussian	1	0.1	0.07
		5	0.1	0.18
		10	0.2	0.44

Problem			Time (seconds)	
german.numer_scale	**kernel**	**C**	**ASL**	**LC-I**
800/24	Linear	1	0.2	0.29
		5	0.6	1.73
		10	0.7	3.05
	Polynomial	1	0.07	0.08
		5	0.1	0.15
		10	0.2	0.27
	Gaussian	1	0.1	0.1
		5	0.3	0.19
		10	0.3	0.39
heart_scale	**kernel**	**C**	**ASL**	**LC-I**
216/13	Linear	1	0.01	0.02
		5	0.04	0.05
		10	0.07	0.11
	Polynomial	1	0.004	0.009
		5	0.008	0.02
		10	0.1	0.02
	Gaussian	1	0.008	
		5	0.02	0.01
		10	0.03	0.02
ionosphere_scale	**kernel**	**C**	**ASL**	**LC-I**
281/34	Linear	1	0.03	0.05
		5	0.06	0.10
		10	0.08	0.11
	Polynomial	1	0.01	0.09
		5	0.02	0.02
		10	0.1	0.03
	Gaussian	1	0.02	0.02
		5	0.03	0.05
		10	0.04	0.05
liver-disorders_scale	**kernel**	**C**	**ASL**	**LC-I**
276/6	Linear	1	0.01	0.02
		5	0.02	0.04
		10	0.02	0.08
	Polynomial	1	0.008	0.01
		5	0.004	0.02
		10	0.01	0.03
	Gaussian	1	0.02	0.02
		5	0.008	0.03
		10	0.03	0.04

Problem			Time (seconds)	
splice_scale	**kernel**	**C**	**ASL**	**LC-I**
	Linear	1	0.5	0.48
		5	3	0.91
data/dimension		10	4.4	1.37
1000/60	Polynomial	1	0.1	0.06
		5	0.2	0.18
		10	0.3	0.34
	Gaussian	1	0.3	0.18
		5	0.4	0.42
		10	0.47	0.46
sonar_scale	**kernel**	**C**	**ASL**	**LC-I**
	Linear	1	0.01	0.03
		5	0.02	0.05
		10	0.05	0.08
167/60	Polynomial	1	0.004	0.007
		5	0.008	0.009
		10	0.008	0.01
	Gaussian	1	0.008	
		5	0.01	0.02
		10	0.01	0.02
svmguide1.scale	**kernel**	**C**	**ASL**	**LC-I**
	Linear	1	0.2	0.92
		5	0.3	1.46
		10	0.5	1.57
3089/4	Polynomial	1	0.4	0.60
		5	0.6	0.74
		10	0.8	1.06
	Gaussian	1	1.3	0.86
		5	1.5	1.01
		10	1.8	1.44
svmguide3	**kernel**	**C**	**ASL**	**LC-I**
	Linear	1	0.1	0.29
		5	0.5	1.07
		10	0.9	7.55
1243/21	Polynomial	1	0.1	0.29
		5	0.2	0.60
		10	0.3	0.91
	Gaussian	1	0.3	0.19
		5	0.4	1.01

w1a	**kernel**	**C**	**ASL**	**LC-I**
2477/300	Linear	1	0.2	1.02
		5	0.5	2.07
		10	0.5	2.33
	Polynomial	1	0.7	1.64
		5	0.8	10.50
		10	0.9	0.15
	Gaussian	1	1.5	4.98
		5	1.7	0.80
		10	1.5	0.55
w2a	**kernel**	**C**	**ASL**	**LC-I**
3470/300	Linear	1	0.3	2.13
		5	0.6	3.42
		10	1.3	5.49
	Polynomial	1	1.3	2.81
		5	1.7	8.2
		10	2	0.28
	Gaussian	1	2.8	3.53
		5	2.8	0.70
		10	3.1	1.18
w3a	**kernel**	**C**	**ASL**	**LC-I**
data/dimension 4912/300	Linear	1	0.6	5.61
		5	1.5	8.50
		10	4	11.84
	Polynomial	1	2.6	7.72
		5	4.2	49.41
		10	4.7	0.15
	Gaussian	1	5.6	1.05
		5	5.9	1.95
		10	6.2	3.06