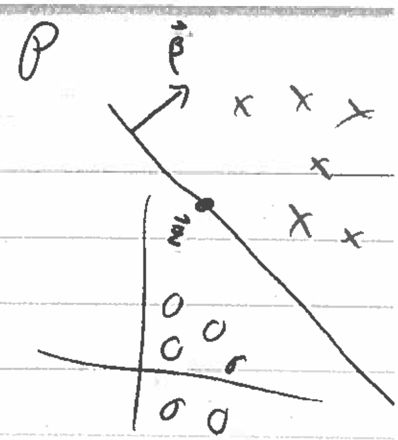
NEW

SVM



"Simple" Case: Totally linearly separable

Seek

Hyperplane that separates w/ maximal margin

1. $\vec{\beta}$ = normal vector
2. $\vec{z}$: point on it

Define $f(\vec{x}) = (\vec{x} - \vec{z}) \cdot \frac{\vec{\beta}}{\|\vec{\beta}\|}$ signed distance from P to $\vec{x}$

decision fcn $\qquad Dst(\vec{x}, P)$

$$f(\vec{x}, P) = \begin{cases} 0 \Rightarrow \text{on } P \\ >0 \Rightarrow \text{"above" } P \\ <0 \Rightarrow \text{"below" } P \end{cases}$$

$s(\vec{x}) = \text{sign}(f(\vec{x}))$ = predicted class of $\vec{x}$

features	true target ± 1	predicted target ± 1	correct +1 / error -1	margin
$\vec{x}_1 \longrightarrow$	y_1	$s(\vec{x}_1)$	$y_1 s(\vec{x}_1)$	$y_1 f(\vec{x}_1)$
$\vec{x}_2 \longrightarrow$	y_2	$s(\vec{x}_2)$	$y_2 s(\vec{x}_2)$	$y_2 f(\vec{x}_2)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\vec{x}_n \longrightarrow$	y_n	$s(\vec{x}_n)$	$y_n s(\vec{x}_n)$	$y_n f(\vec{x}_n)$

2

OPT 1

Original optimization:

variables $\vec{\beta}, \vec{z}, M > 0$ $\quad df = 2p+1$

objective maximize M

constraints $y_i f(\vec{x}_i) \geq M \quad \forall i$

---

refine

Note $\|\vec{\beta}\|$ does not matter b/c $f(\vec{x}) = (\vec{x} - \vec{z}) \cdot \frac{\vec{\beta}}{\|\vec{\beta}\|} = (\vec{x} - \vec{z}) \frac{k\vec{\beta}}{\|k\vec{\beta}\|}$ $\quad \forall k > 0$

Reduce dimensionality by encoding M into $\vec{\beta}$ via

* $\|\vec{\beta}\| = 1/M$ $\quad df \quad 2p+1 \to 2p$

OPT 2 $\quad df = 2p$

$M = \frac{1}{\|\vec{\beta}\|}$

variables $\vec{\beta}, \vec{z}$

objective max $M \to \max \frac{1}{\|\vec{\beta}\|} \to \min \|\vec{\beta}\| \to \boxed{\min \frac{1}{2}\|\vec{\beta}\|^2}$

constraints $y_i f(\vec{x}_i) \geq M \to y_i (\vec{x}_i - \vec{z}) \cdot \frac{\vec{\beta}}{\|\vec{\beta}\|} \geq M$

$\to y_i (\vec{x}_i - \vec{z}) \cdot \vec{\beta} \geq 1$

$\to \boxed{y_i (x_i - \vec{z}) \cdot \vec{\beta} - 1 \geq 0 \quad \forall i}$

More df reduction

Constraints $y_i(\vec{x}\cdot\vec{\beta} - \underbrace{\vec{z}\cdot\vec{\beta}}) - 1 \geq 0$

only role for $\vec{z}$ is its dot w/ $\vec{\beta}$

let $\beta_0 = -\vec{z}\cdot\vec{\beta}$

$$\boxed{\begin{array}{l} \underline{\text{OPT3}} \\ \text{variables} \quad \vec{\beta}, \beta_0 \qquad df: p+1 \\ \underline{\text{objective}} \quad \min \frac{1}{2}\|\vec{\beta}\|^2 \\ \underline{\text{constraints}} \quad y_i(\vec{x}_i\cdot\vec{\beta} + \beta_0) - 1 \geq 0 \quad \forall i \end{array}}$$

How to solve? Lagrange Multipliers

Calc 3 review

Baby Lagrange

1 objective $f(\vec{x})$

1 equality constraint $h(\vec{x}) = 0$

Critical points

I) $\nabla f(\vec{x})$ or $\nabla h(\vec{x})$ DNE

II) $\nabla f(\vec{x}) = \lambda \nabla h(\vec{x})$

Lagrange Multiplier

Daddy Lagrange

1 objective $f(\vec{x})$

many equality constraints $h_1(\vec{x}) = 0$ ... $h_n(\vec{x}) = 0$

I) $\nabla f(\vec{x})$ or any $\nabla h_i(\vec{x})$ DNE

II) $\nabla f(\vec{x}) = \lambda_1 \nabla h_1(\vec{x}) + \lambda_2 \nabla h_2(\vec{x}) + \lambda_n \nabla h_n(\vec{x})$

$$\nabla \underbrace{\left[f(\vec{x}) - \sum_{i=1}^{n} \lambda_i h_i(\vec{x})\right]}_{L_P} = \vec{0}$$

Lagrange Primal

Karush - Kuhn Tucker

Follow Wikipedia KKT

1 objective $f(\vec{x})$

many constants, equality & inequality $g_i(\vec{x}) \le 0$

$h_i(\vec{x}) = 0$

$$L_p = f(\vec{x}) \pm \sum \mu_i g_i(\vec{x}) \pm \sum \lambda_i h_i(\vec{x})$$

+ for minimize

− for maximize

KKT Solve
ⓐ $\nabla L_p(\vec{x}) = \vec{0}$
ⓑ $g_i(\vec{x}) \le 0 \quad \forall_i$
ⓒ $h_i(\vec{x}) = 0 \quad \forall_i$
ⓓ $\mu_i \ge 0 \quad \forall_i$
ⓔ $\mu_i g_i(\vec{x}) = 0 \quad \forall_i$

Apply KKT to SVM

$$f(\vec{\beta}, \beta_0) = \frac{1}{2}\|\vec{\beta}\|^2 = \frac{1}{2}\sum_{j=1}^{p} \beta_j^2$$

− to make $\ge 0 \rightsquigarrow \le 0$ as KKT desires

$$g_i(\vec{\beta}, \beta_0) = -\left[y_i(\vec{x}_i \cdot \vec{\beta} + \beta_0) - 1\right] \le 0$$

no h_i

$$= -\left[y_i\left(\sum_{j=1}^{p} x_{ij}\beta_j + \beta_0\right) - 1\right] \le 0$$

ⓐ
$$L_p(\vec{\beta}, \beta_0) = \frac{1}{2}\|\vec{\beta}\|^2 - \sum_{i=1}^{n} \mu_i\left[y_i(\vec{x}_i \cdot \vec{\beta} + \beta_0) - 1\right]$$

$$= \frac{1}{2}\sum_{j=1}^{p} \beta_j^2 - \sum_{i=1}^{n} \mu_i\left[y_i\left(\sum_{j=1}^{p} x_{ij}\beta_j + \beta_0\right) - 1\right]$$

$\nabla L_P(\vec{\beta}, \beta_0) = 0$

$k=0$ $\quad 0 = \dfrac{\partial L_P}{\partial \beta_0} = \dfrac{1}{2}\sum_{j=1}^{p} 0 - \sum_{i=1}^{n} \mu_i \left[y_i \left(\sum_{j=1}^{p} 0 + 1 \right) - 0 \right]$

$$= -\sum_{i=1}^{n} \mu_i y_i$$

$$\boxed{(f) \quad \sum_{i=1}^{n} \mu_i y_i = 0}$$

all terms $j \neq k$ vanish

$k \geq 1$ $\quad 0 = \dfrac{\partial L_P}{\partial \beta_k} = \dfrac{1}{2}(2\beta_k) - \sum_{i=1}^{n} \mu_i \left[y_i \left(x_{ik} + 0 \right) - 0 \right]$

$$= \beta_k - \sum_{i=1}^{n} \mu_i y_i x_{ik}$$

$$\boxed{(g) \quad \beta_k = \sum_{i=1}^{n} \mu_i y_i x_{ik}}$$

<u>Now</u> sub (f) & (g) into L_P & do a bunch of algebra

Get "Wolfe Dual"

$$(a') \quad L_D = \sum_{i=1}^{n} \mu_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \mu_i \mu_k y_i y_k (\vec{x}_i \cdot \vec{x}_k)$$

variables $\vec{\beta}, \beta_0, \mu_i$

<u>OPT 4</u> Maximize L_D subject to (b) $\rightarrow$ (g)

Now Focus on (e) $\mu_i g_i = 0$

drop b/c $=0$

$$\mu_i\left[-y_i\left(\vec{x}_i\cdot\vec{\beta}+\beta_0\right)-1\right]=0$$

<u>2 cases</u>

1) $\mu_i = 0$. For each such i, corresponding term drops out of L_D ☺ Major complexity reduction

2) $\mu_i > 0$ & $y_i(\vec{x}_i\cdot\vec{\beta}+\beta_0)-1=0$.

<u>recall</u>: $y_i = \pm 1$

$\beta_0 = -\vec{z}\cdot\vec{\beta}$

$M = 1/\|\vec{\beta}\|$

$$\text{dist}(\vec{x},P) = (\vec{x}-\vec{z})\cdot\frac{\vec{\beta}}{\|\vec{\beta}\|} = \left[(\vec{x}-\vec{z})\cdot\vec{\beta}\right]M = M(\vec{x}_i\cdot\vec{z}+\beta_0)$$

So $y_i(\vec{x}_i\cdot\vec{\beta}_i+\beta_0)-1=0$

$$\frac{M}{y_i}\cdot y_i\left(\vec{x}_i\cdot\beta_i+\beta_0\right) = 1\cdot\frac{M}{y_i}$$

$$\boxed{\text{Dist}(\vec{x}_i, P) = \pm M}$$

"Only terms of L_D that survive are from vectors that are exactly the minimal margin M from P $\rightarrow$ support vectors"

Significance There are typically far fewer support vectors than $n = \#$rows. Solving constrained optimization in, say $\mathbb{R}^4$, is MUCH faster than in $\mathbb{R}^{3000}$.

OPTS Final
Variables $\vec{\beta}, \beta_0, \mu_i$

For each $\vec{\beta}, \beta_0$, compute $S = \{ i \mid y_i(\vec{x}_i \cdot \vec{\beta} + \beta_0) - 1 = 0 \}$

objective Maximize (a'') $L_D = \sum_{i \in S} \mu_i - \frac{1}{2} \sum_{i,k \in S} \mu_i \mu_k y_i y_k (\vec{x}_i \cdot \vec{x}_k)$

constraints (b) $\to$ (g)

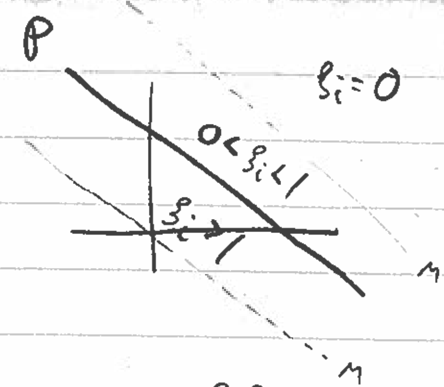
Again (a'') has far fewer terms than (a') or (a)

This is a convex optimization problem. We worked hard to reduce complexity to a point where quadratic programming methods from numerical analysis can efficiently solve. We won't go farther down that rabbit hole.

"decision function" $f(\vec{x}) = \vec{x} \cdot \vec{\beta} + \beta_0$

That was the simple case

<u>Now</u> the non-simple cases



~~Totally~~ Linearly Separable

Introduce "slack variables" ξ_i where

$$\xi_i = \begin{cases} = 0 & \text{if } \vec{x}_i \text{ is on correct side of } \beta \text{ \& farther than } M \\ \in (0,1) & \text{if } \vec{x}_i \text{ is on correct side of } \beta \text{ but closer than } M \\ \in [1, \infty) & \text{if } \vec{x}_i \text{ is on wrong side of } \beta \end{cases}$$

• Hyperparameter C = cost "total budget of)"

OPT minimize $\frac{1}{2}\|\vec{\beta}\|^2 + C \sum_{i=1}^{n} \xi_i$

constraints $\xi_i \geq 0$ $\forall i$

$y_i(\vec{x}_i \cdot \vec{\beta} + \beta_0) \geq 1 - \xi_i$

Do same KKT thing as before

decision fnc $F(\vec{x}) = \vec{x} \cdot \vec{\beta} + \beta_0$

<u>Notes</u>

• ξ_i are parameters like μ_i which are fit by computer. You don't control.

• C is a regularization hyperparameter. You <u>tune</u> this.

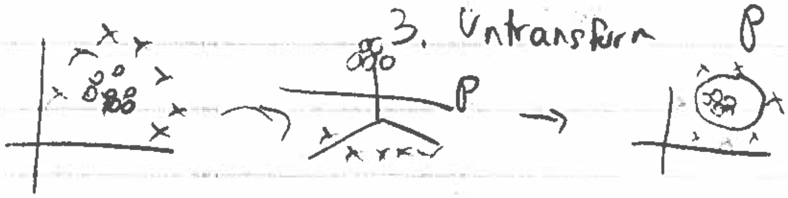
bigger $C \Rightarrow$ penalizes vectors in strip more

$\Rightarrow$ smaller margin but fewer vectors in it

smaller $C \Rightarrow$ bigger margin, more vectors in it

~~Totally~~ ~~Linearly~~ Separable

<u>Idea</u>
1. Transform features (often to higher dimension)
2. Fit linear SVM β
3. Untransform β usually becomes non-linear



<u>Detail</u>

Pick function $\vec{h} : \mathbb{R}^p \to \mathbb{R}^q$ $\quad p \le q$

Apply $\vec{h}$ to each row $\vec{x}_i$

Goal: the 2 classes can be separated with a hyperplane in the higher dim $\mathbb{R}^q$ post-transform

<u>Changes</u>:

- (a′) $L_D = \sum_{i=1}^{n} \mu_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \mu_i \mu_k y_i y_k (\vec{x}_i \cdot \vec{x}_k)$ becomes $\vec{h}(\vec{x}_i) \cdot \vec{h}(\vec{x}_k)$

- (g) $\beta_k = \sum_{i=1}^{n} \mu_i y_i x_{ik}$ $\quad h_k(\vec{x}_i)$

where h_k is the k^{th} component fcn of $\vec{h}$. In other words, entry k of the <u>output</u> vector is given by h_k

- decision fcn

$$f(\vec{x}) = \vec{x} \cdot \vec{\beta} + \beta_0$$

$$\vec{h}(\vec{x})$$

$$f(\vec{x}) = \left[\vec{h}(\vec{x}) \cdot \vec{\beta}\right] + \beta_0$$

$$= \left[\sum_{k=1}^{q} h_k(\vec{x})\, \beta_k\right] + \beta_0 \qquad \text{Def of dot product}$$

$$= \left[\sum_{k=1}^{q} h_k(\vec{x}) \sum_{i=1}^{n} \mu_i y_i h_k(\vec{x}_i)\right] + \beta_0 \qquad \text{Using (9)}$$

$$= \left[\sum_{i=1}^{n} \mu_i y_i \left(\sum_{k=1}^{q} h_k(\vec{x}) h_k(\vec{x}_i)\right)\right] + \beta_0 \qquad \text{reverse order of } \Sigma\Sigma$$

$$f(\vec{x}) = \left[\sum_{i=1}^{n} \mu_i y_i \left(\vec{h}(\vec{x}) \cdot \vec{h}(\vec{x}_i)\right)\right] + \beta_0 \qquad \text{def of dot product}$$

It may not be obvious, but this is <u>great</u> b/c the only place $\vec{h}$ is involved in the decision function is a dot product. $\vec{h}(\vec{x}) \cdot \vec{h}(\vec{x}_i)$.

"Reproducing Kernel Hilbert Spaces" §5.8 Hastie, Tibshirani, Friedman

The Kernel Trick

Summary: There are special choices for $\vec{h}$ where $\exists\, K$ such that

- $\vec{h}(\vec{x}) \cdot \vec{h}(\vec{y}) = K(\vec{x}, \vec{y})$
- K is symmetric & positive definite
- K is computationally efficient

"kernel"

If we stick to one of these special choices of $\vec{h}$, we only need to actually use K. Makes SVM efficient.

In fact, we never need to think about $\vec{h}$ again. Just express using K directly.

Common Choices

- polynomial $K(\vec{x},\vec{y}) = (1+\vec{x}\cdot\vec{y})^d$
- rbf radial basis $K(\vec{x},\vec{y}) = e^{-\gamma\|\vec{x}-\vec{y}\|^2}$
- neural network $K(\vec{x},\vec{y}) = \tanh(K_1\, \vec{x}\cdot\vec{y} + K_2)$

Sklearn says "sigmoid"

where d, γ, K_1 & K_2 are hyper params.

SVM hyper params

1. C = cost
2. which kernel
3. hyperparams for that kernel.

Decision fcn

$$f(\vec{x}) = \sum_{i=1}^{n} \mu_i y_i K(\vec{x},\vec{x}_i) + \beta_0$$

WIKIPEDIA

Karush–Kuhn–Tucker conditions

In mathematical optimization, the **Karush–Kuhn–Tucker (KKT) conditions**, also known as the **Kuhn–Tucker conditions**, are first derivative tests (sometimes called first-order) necessary conditions for a solution in nonlinear programming to be optimal, provided that some regularity conditions are satisfied.

Allowing inequality constraints, the KKT approach to nonlinear programming generalizes the method of Lagrange multipliers, which allows only equality constraints. Similar to the Lagrange approach, the constrained maximization (minimization) problem is rewritten as a Lagrange function whose optimal point is a saddle point, i.e. a global maximum (minimum) over the domain of the choice variables and a global minimum (maximum) over the multipliers, which is why the Karush–Kuhn–Tucker theorem is sometimes referred to as the saddle-point theorem.[1]

The KKT conditions were originally named after Harold W. Kuhn and Albert W. Tucker, who first published the conditions in 1951.[2] Later scholars discovered that the necessary conditions for this problem had been stated by William Karush in his master's thesis in 1939.[3][4]

Contents

Nonlinear optimization problem

Consider the following nonlinear minimization or maximization problem:

Optimize $f(\mathbf{x})$

subject to

$$g_i(\mathbf{x}) \leq 0,$$
$$h_i(\mathbf{x}) = 0.$$

where $\mathbf{x} \in \mathbf{X}$ is the optimization variable chosen from a convex subset of $\mathbb{R}^n$, f is the objective or utility function, g_i $(i = 1, \ldots, m)$ are the inequality constraint functions and h_i $(i = 1, \ldots, \ell)$ are the equality constraint functions. The numbers of inequalities and equalities are denoted by m and ℓ respectively. Corresponding to the constraint optimization problem one can form the Lagrangian function

$$L(\mathbf{x}, \mu) = f(\mathbf{x}) + \mu^\top \mathbf{g}(\mathbf{x}) + \lambda^\top \mathbf{h}(\mathbf{x})$$

where $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), \ldots, g_m(\mathbf{x}))^\top$, $\mathbf{h}(\mathbf{x}) = (h_1(\mathbf{x}), \ldots, h_\ell(\mathbf{x}))^\top$. The **Karush–Kuhn–Tucker theorem** then states the following.

Theorem. If $(\mathbf{x}^*, \mu^*)$ is a saddle point of $L(\mathbf{x}, \mu)$ in $\mathbf{x} \in \mathbf{X}$, $\mu \geq \mathbf{0}$, then $\mathbf{x}^*$ is an optimal vector for the above optimization problem. Suppose that $f(\mathbf{x})$ and $g_i(\mathbf{x})$, $i = 1, \ldots, m$, are concave in $\mathbf{x}$ and that there exists $\mathbf{x}_0 \in \mathbf{X}$ such that $\mathbf{g}(\mathbf{x}_0) > 0$. Then with an optimal vector $\mathbf{x}^*$ for the above optimization problem there is associated a non-negative vector μ^* such that $L(\mathbf{x}^*, \mu^*)$ is a saddle point of $L(\mathbf{x}, \mu)$.

Since the idea of this approach is to find a supporting hyperplane on the feasible set $\mathbf{\Gamma} = \{\mathbf{x} \in \mathbf{X} : g_i(\mathbf{x}) \geq 0, i = 1, \ldots, m\}$, the proof of the Karush–Kuhn–Tucker theorem makes use of the hyperplane separation theorem.[5]

The system of equations and inequalities corresponding to the KKT conditions is usually not solved directly, except in the few special cases where a closed-form solution can be derived analytically. In general, many optimization algorithms can be interpreted as methods for numerically solving the KKT system of equations and inequalities.[6]

Necessary conditions

Suppose that the objective function $f : \mathbb{R}^n \to \mathbb{R}$ and the constraint functions $g_i : \mathbb{R}^n \to \mathbb{R}$ and $h_j : \mathbb{R}^n \to \mathbb{R}$ are continuously differentiable at a point x^*. If x^* is a local optimum and the optimization problem satisfies some regularity conditions (see below), then there exist constants μ_i $(i = 1, \ldots, m)$ and λ_j $(j = 1, \ldots, \ell)$, called KKT multipliers, such that the following four groups of conditions hold:

Stationarity

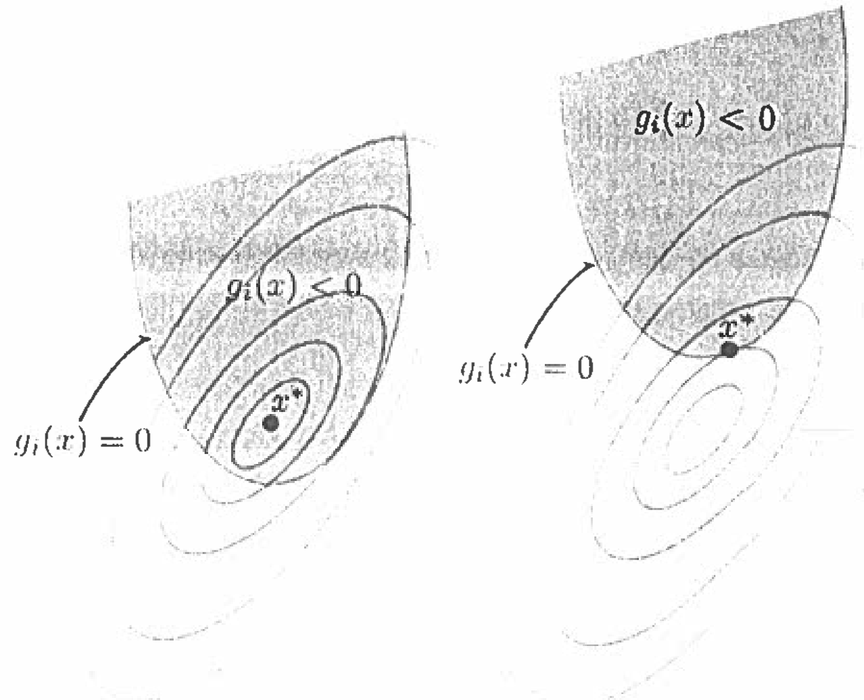
For maximizing $f(x)$: $\nabla f(x^*) - \sum_{i=1}^{m} \mu_i \nabla g_i(x^*) - \sum_{j=1}^{\ell} \lambda_j \nabla h_j(x^*) = 0,$

For minimizing $f(x)$: $\nabla f(x^*) + \sum_{i=1}^{m} \mu_i \nabla g_i(x^*) + \sum_{j=1}^{\ell} \lambda_j \nabla h_j(x^*) = 0,$

Primal feasibility

$$g_i(x^*) \leq 0, \text{ for } i = 1, \ldots, m$$
$$h_j(x^*) = 0, \text{ for } j = 1, \ldots, \ell$$

Dual feasibility



Inequality constraint diagram for optimization problems

$$\mu_i \geq 0, \text{ for } i = 1, \ldots, m$$

Complementary slackness

$$\mu_i g_i(x^*) = 0, \text{ for } \; i = 1, \ldots, m.$$

In the particular case $m = 0$, i.e., when there are no inequality constraints, the KKT conditions turn into the Lagrange conditions, and the KKT multipliers are called Lagrange multipliers.

If some of the functions are non-differentiable, subdifferential versions of Karush–Kuhn–Tucker (KKT) conditions are available.[7]

Regularity conditions (or constraint qualifications)

In order for a minimum point x^* to satisfy the above KKT conditions, the problem should satisfy some regularity conditions; some common examples are tabulated here:

Constraint	Acronym	Statement
Linearity constraint qualification	LCQ	If g_i and h_j are affine functions, then no other condition is needed.
Linear independence constraint qualification	LICQ	The gradients of the active inequality constraints and the gradients of the equality constraints are linearly independent at x^*.
Mangasarian-Fromovitz constraint qualification	MFCQ	The gradients of the equality constraints are linearly independent at x^* and there exists a vector $d \in \mathbb{R}^n$ such that $\nabla g_i(x^*)^\top d < 0$ for all active inequality constraints and $\nabla h_j(x^*)^\top d = 0$ for all equality constraints.[8]
Constant rank constraint qualification	CRCQ	For each subset of the gradients of the active inequality constraints and the gradients of the equality constraints the rank at a vicinity of x^* is constant.
Constant positive linear dependence constraint qualification	CPLD	For each subset of gradients of active inequality constraints and gradients of equality constraints, if the subset of vectors is linearly dependent at x^* with non-negative scalars associated with the inequality constraints, then it remains linearly dependent in a neighborhood of x^*.
Quasi-normality constraint qualification	QNCQ	If the gradients of the active inequality constraints and the gradients of the equality constraints are linearly dependent at x^* with associated multipliers λ_j for equalities and $\mu_i \geq 0$ for inequalities, then there is no sequence $x_k \to x^*$ such that $\lambda_j \neq 0 \Rightarrow \lambda_j h_j(x_k) > 0$ and $\mu_i \neq 0 \Rightarrow \mu_i g_i(x_k) > 0$.
Slater's condition	SC	For a convex problem (i.e., assuming minimization, f, g_i are convex and h_j is affine), there exists a point x such that $h(x) = 0$ and $g_i(x) < 0$.

It can be shown that

$$\text{LICQ} \Rightarrow \text{MFCQ} \Rightarrow \text{CPLD} \Rightarrow \text{QNCQ}$$

and

$$\text{LICQ} \Rightarrow \text{CRCQ} \Rightarrow \text{CPLD} \Rightarrow \text{QNCQ}$$

(and the converses are not true), although MFCQ is not equivalent to CRCQ.[9] In practice weaker constraint qualifications are preferred since they provide stronger optimality conditions.

Sufficient conditions

In some cases, the necessary conditions are also sufficient for optimality. In general, the necessary conditions are not sufficient for optimality and additional information is required, such as the Second Order Sufficient Conditions (SOSC). For smooth functions, SOSC involve the second derivatives, which explains its name.

The necessary conditions are sufficient for optimality if the objective function f of a maximization problem is a concave function, the inequality constraints g_j are continuously differentiable convex functions and the equality constraints h_i are affine functions.

It was shown by Martin in 1985 that the broader class of functions in which KKT conditions guarantees global