	Gradient	Subgrad	Prox grad	Newton	Conj grad	quasi-Newton
Criterion	smooth f	any f	smooth + simple, $f = g + h$	doubly smooth f	doubly smooth f	doubly smooth f
Constraints	projection onto constraint set	projection onto constraint set	constrained prox operator	equality constraints	unconstrained	unconstrained
Opti parameters	fixed step size $(t \leq 1/L)$ or line search	diminishing step sizes	fixed step size $(t \leq 1/L)$ or line search	pure step size $(t = 1)$ or line search	FR & PR: line search, versions that use the Hessian: fixed step size	DFP & BFGS: line search
Iteration cost	cheap (compute gradient)	cheap (compute subgradient)	moderately cheap (evaluate prox)	moderate to expensive (compute Hessian and solve linear system)	moderately cheap (compute gradients, inner products)	moderately cheap (compute gradients, inner products; no matrix inversion, but storage for estimated inverse Hessian)
Rate	$O(\frac{1}{\epsilon})$ [$O(\frac{1}{\sqrt{\epsilon}})$ with acceleration, $O(\log(\frac{1}{\epsilon}))$ with strong convexity]	$O(\frac{1}{\epsilon^2})$	$O(\frac{1}{\epsilon})$ [$O(\frac{1}{\sqrt{\epsilon}})$ with acceleration, $O(\log(\frac{1}{\epsilon}))$ with strong convexity]	$O(\log\log(\frac{1}{\epsilon}))$ (quadratic rate)	superlinear rate, n-step quadratic rate (n steps are as effective as one Newton step)	superlinear rate, n-step quadratic rate (n steps are as effective as one Newton step)

	Barrier method	Primal-dual IPM	ADMM	Coord desc
Criterion	doubly smooth f	doubly smooth f	block separable, $f(x,z) = g(x) + h(z)$	smooth + component-wise separable
Constraints	doubly smooth h_i (ineq constraints)	doubly smooth h_i (ineq constraints)	equality constraints (always) & ineq constraints (sometimes)	component-wise separable constraints
Opti parameters	inner loop: fixed step size or use line search, outer loop: diverging barrier parameter	line search for step size & diverging barrier parameter	fixed augmented Lagrange parameter (theory), or varied by iteration (practice)	none!
Iteration cost	expensive to very expensive (one iteration solves one smoothed problem, by Newton)	moderate to expensive (one iteration performs one Newton step)	cheap to expensive (one iteration solves two subproblems, makes a dual step)	cheap to expensive (one iteration performs a full cycle of component minimizations)
Rate	$O(\log(\frac{1}{\epsilon}))$ (both in terms of iterations and Newton steps)	$O(\log(\frac{1}{\epsilon}))$	not known in general, but known in special cases; practically tends to behave like a first-order method	not known in general, but known in special cases; practically tends to behave faster than first-order methods