

Unconstrained Optimization Algorithms

General Ideas

- Select step size s_k and direction v_k such that $f_0(x_k + s_k v_k)$ is substantially smaller than $f_0(x_k)$.
- v_k is usually selected such that $\nabla f_0(x_k)^\top v_k < 0$
- First order descent

Armijo backtracking step size rule

1. Choose $\alpha, \beta \in (0, 1)$, set $s = 1$
2. If $f_0(x_k + s v_k) \leq f_0(x_k) + s\alpha \nabla f_0(x_k)^\top v_k$, then return $s_k = s$.
3. Set $s = \beta s$ and return to step 2.

Newton-type methods

- Newton's method: $v_k = -\nabla^2 f_0(x_k)^{-1} \nabla f_0(x_k)$
- Interpretation: Minimizing second order approximation: $\min_x f_0(x_k) + \nabla f_0(x_k)^\top x + \frac{1}{2} x^\top \nabla^2 f_0(x_k) x$
- Quasi-Newton and variable-metric: $v_k = -H_k \nabla f_0(x_k)$
- Combined with any step size rule

Constrained Optimization Algorithms

Methods

- **Projected Decent:** Use a direction v_k from unconstrained algorithms and compute $x'_{k+1} = x_k + s_k v_k$ and then project x'_{k+1} onto the feasible set.
- Construct an unconstrained problem that approximates the original problem.

Barrier method

- Consider the problem $\min_x f_0(x)$ subject to $f_i(x) \leq 0$ for $i = 1 \dots m$.
- Define the approximate problem for some parameter $t \geq 0$: $\min_x f_0(x) + \frac{1}{t} \phi(x)$, where the barrier function $\phi(x)$ tends to ∞ as x tends to the boundary of the feasible set from inside.
- We can use the logarithmic barrier function: $\phi(x) = -\sum_{i=1}^m \log(-f_i(x))$
- If f_0, f_i are convex, the approximated problem is convex
- The KKT condition is $t \nabla f_0(x) + \sum_{i=1}^m \frac{1}{-f_i(x)} \nabla f_i(x) = 0$
- The approximation error: $0 \leq f_0(x^*(t)) - p^* \leq \frac{m}{t}$

Machine Learning

Linear Model

- Linear model: $y(x) = w^\top \phi(x) + e(x)$, where $\phi(x)$ is a feature map and $e(x)$ is a random error
- Least square: $\hat{w} \in \arg \min_w \sum_i (w^\top \phi(x_i) - y_i)^2$
- Quantile for $\alpha \in [0, 1)$: $\hat{w} \in \arg \min_w \frac{1}{m(1-\alpha)} \sum_i \max\{0, w^\top \phi(x_i) - y_i\} - \frac{1}{m} \sum_i w^\top \phi(x_i) - y_i$. Still a linear program.

Support Vector Machine

- Classification rule: $\hat{y}(x) = \text{sign}(\phi(x)^\top x)$
- Convex error function(hinge): $\max\{0, 1 - \alpha\}$
- Objective: $\min_w \frac{1}{m} \sum_i \max\{0, 1 - y_i \phi(x_i)^\top w\}$
- We can add L_1 or L_2 norm regularization on w

Logistic Regression

- Variant of SVM with another loss function: $\min_w \frac{1}{m} \sum_i \log[1 + \exp(-y_i \phi(x_i)^\top w)]$
- Interpretation: $P\{Y = y|X = x\} = \frac{1}{1 + \exp(-y \phi(x)^\top w)}$

Fisher discrimination

- Idea: find a direction in \mathbb{R}^n , such that the projection of data points onto the direction results in those points with positive label are from those with negative label
- Define $A_+ \in \mathbb{R}^{n, m_+}$ is the matrix whose column consists of x_i such that $y_i = 1$, and A_- similarly for negive labeled points. The centered(mean subtracted) data matrices are $A_\pm = A_\pm(I_{m_\pm} - \frac{1}{m_\pm} \mathbf{1}\mathbf{1}^\top)$
- The mean-squared variation of the data projected on u around its centroid is: $u^\top M u = u^\top (\frac{1}{m_-} \tilde{A}_- \tilde{A}_-^\top + \frac{1}{m_+} \tilde{A}_+ \tilde{A}_+^\top) u$
- The goal is: $\max_{u \neq 0} \frac{(u^\top c)^2}{u^\top M u} = \min u^\top M u$ subject to $u^\top c = 1$
- The problem is convex, hence KKT condition implies: $u = \frac{1}{c^\top M^{-1} c} M^{-1} c$

SPCA and NNMF

- The original PCA problem: $\max_z z^\top \tilde{X} \tilde{X}^\top z$ s.t. $\|z\|_2 = 1$
- In sparse PCA, we add constraint $\text{card}(z) \leq k$
- The problem leads to the rank 1 approximation problem: $\min_{p, q} \|X - pq^\top\|_F$ s.t. $\text{card}(p) \leq k, \text{card}(q) \leq h$
- One example of rank 1 approximation is: $\min_{p \geq 0, q \geq 0} \|X - pq^\top\|_F$ s.t. $\text{card}(p) \leq k, \text{card}(q) \leq h$. A standard solution is coordinate descent.

Finance

Mean-Variance (Markowitz) Models

- $x \in \mathbb{R}^n$ is the wealth allocation in n groups, r is the random vector of unit wealth return for each group. $\hat{r} = E[r], \Sigma = \text{Cov}(r)$
- Hence, $\hat{r}^\top x$ is the expected return and $x^\top \Sigma x$ is the variance of the portfolio.
- The problems are usually maximize return with variance constraints and minimize variance, with return constraints.
- Budget constraint: $x^\top \mathbf{1} \leq b$, no short-selling: $x \geq 0$
- Sector bounds: $\sum_{i \in S} x_i \leq \alpha x^\top \mathbf{1}$ for some $\alpha \in (0, 1)$ and sector S .
- Diversification: sum of k largest x_i must not exceed $\eta x^\top \mathbf{1}$ for $\eta \in (0, 1)$. This is equivalent to $kt + s^\top \mathbf{1} \leq \eta x^\top \mathbf{1}$, $s \geq 0, x - t\mathbf{1} \leq s$ where $t \in \mathbb{R}$ and $s \in \mathbb{R}^n$ are additional variables.
- The Sharpe Ratio of a portfolio is $(\hat{r}^\top x - r_f)/\sqrt{x^\top \Sigma x}$, where r_f is the return of a risk-free asset.
- If $\hat{r}^\top x > r_f$ and $\Sigma \succ 0$, then the portfolio that maximizes SR corresponds to a point on the efficient frontier. Max SR can be formulated as SOCP.

Value at Risk and Conditional Value at Risk

- The value-of-risk of a portfolio $r^\top x$ at probability level $\alpha \in (0, 1)$ is defined as $\text{VaR}_\alpha(r^\top x) = -F^{-1}(\alpha)$, where F is the CDF of $r^\top x$.
- If r is Gaussian with mean \hat{r} and covariance Σ , we have $\text{VaR}_\alpha(r^\top x) \leq \gamma$ is equivalent as SOC constraint $\Phi^{-1}(1 - \alpha) \|\Sigma^{\frac{1}{2}}\|_2 \leq \gamma + \hat{r}^\top x$

Epi-Spline Curve Fitting

Epi-splines

- Epi-spline: Left continous, piece-wise polynomial
- On the k^{th} segment, $q^k(x) = \sum_{i=0}^p a_i^k x^i$. Hence if we split the domain into N segments with degree p polynomial, our total number of parameters would be $N(p + 1)$
- Continuity constraint: Let $m_1 \dots m_{N-1}$ be the mesh points, we have $q^k(m_k) = q^{k+1}(m_k)$ for $k = 1 \dots N - 1$.
- Continuous differentiability: Continuty constraint plus $(q^k(m_k))' = (q^{k+1}(m_k))'$ for $k = 1 \dots N - 1$.
- Fixed values: $q^k(x) = s(k)$
- Monotonicity (non-increasing): $(q^k(x))' \leq 0$ for $x \in (m_{k-1}, m_k)$, and $q^k(m_k) \geq q^{k+1}(m_{k+1})$. The second requirement is satisfied with continuity.
- Convexity: Continuty constraint and $(q^k(x))'' \geq 0$, $(q^k(m_k))' \leq (q^{k+1}(m_{k+1}))'$ for $k = 1 \dots N - 1$.
- All constraints above are linear.

Optimization in Control

Continuous LTI Systems

- A continuous LTI system is described by a system ODEs: $\dot{x}(t) = Ax(t) + Bu(t)$, $y(t) = Cx(t)$
- The solution is: $x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau$, $t \geq t_0$

Discrete-time LTI System and Discretization

- We discretize the time interval $t = k\Delta$ for $k = \dots -1, 0, 1, \dots$ and DTLTI system can be described as $x(k+1) = Ax(k) + Bu(k)$, $y(k) = Cx(k)$
- The solution is: $x(k) = A^{k-k_0}x(k_0) + \sum_{i=k_0}^{k-1} A^{k-i-1}Bu(i)$ for $k \geq k_0$.
- Assuming constant control input during one time interval, i.e. $u(t) = u(k\Delta)$, $\forall t \in [k\Delta, (k+1)\Delta)$, we can discretize a CTLTI system by letting $A_\Delta = e^{A\Delta}$, $B_\Delta = \int_0^\Delta e^{A\tau}d\tau$. The we get the system $x(k+1) = A_\Delta x(k) + B_\Delta u(k)$

Optimization-based Control Synthesis