1 Convex Sets

1.1. (a) Sketch the following sets Ω and (b) identify the convex sets:

- $\Omega = \{x | \, a^T x \leq b\}$
- $\Omega = \{x | \, (x - x_c)^T P (x - x_c) \leq 1\}, P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
- $\Omega = \{x | \, ||x||_2 \leq 1\}$

1.2. Show that the solution set of linear equations $\{x | Ax = b\}$ with $x \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ is an affine set.

2 Convex Functions

2.1. Define the function $f(x,y) = \frac{||Ax-b||_2^2}{Py-q}$ with $x, y \in \mathbb{R} \wedge Py - b > 0$. Check the convexity of f over the domain.

3 Convex Problems

3.1. For the following optimization problem

$$\begin{array}{ll} \text{minimize} & ||(2x+3y,-3x)^T||_\infty \\ \text{subject to} & |x-2y| \leq 3 \end{array}$$

(a) Express the problem as a a linear program. (b) Convert the LP so that all variables are in $\mathbb{R}_+$ and there are no other inequality constraints.

4 Duality

4.1. Derive a dual problem for

$$\text{minimize} \quad -\sum_{i=1}^{m} \log(b_i - a_i^T x)$$

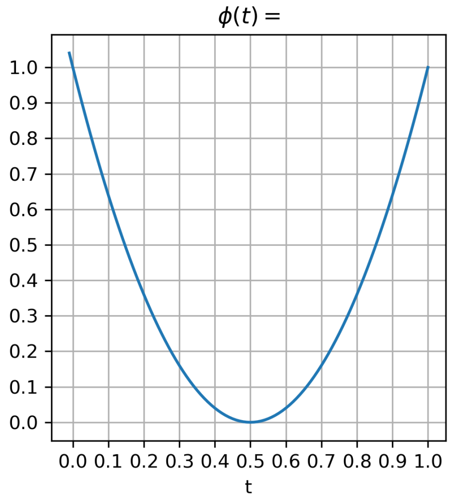
with domain $\{x | a_i^T x < b_i, i = 1, ..., m\}$. First introduce new variables y_i and equality constraints $y_i = b_i - a_i^T x$. Write out the infimum in the final result.

5 Line Search

5.1. Consider the function

$$f(x,y) = 4(x-1/2)^2 + y^2.$$

A line search optimization algorithm starting at $(x^{(0)}, y^{(0)}) = (1,0)$ found a search direction $(\Delta x, \Delta y) = (-1,0)$ for which the function f constrained a line gives the function $\phi(t)$ in the plot.



(1) What is the expression for $\phi(t)$ for the give starting point and direction?

(2) Compute the step that an exact line search would produce for this function.

(3) Sketch the interval that a backtracking line search satisfying the armijo condition with $\alpha = 0.25$ would consider. Explain why, a geometric argument is enough.

(4) Sketch the interval that a backtracking line search satisfying the Wolfe conditions with $\alpha = 0.25$ and $\beta = 0.8$ would consider. Explain why, a geometric argument is enough.

(5) Verify the intervals above by performing the computations.

6 Active Set Method

6.1. Consider the following 2-dimensional minimization problem

$$
\begin{array}{lr}
\text{min} & q(x,y) = (x-1)^2 + (y-2.5)^2 \\
\text{subject to:} & x - 2y + 2 \geq 0, \\
 & -x - 2y + 6 \geq 0, \\
 & -x + 2y + 2 \geq 0, \\
 & x \geq 0, \\
 & y \geq 0.
\end{array}
$$

(a) Sketch the feasible set.

(b) Assume we start from point $(1, 1/2)^T$, what are the active constraints?

(c) Start from the initial state of (2), write down the steps of the active set method until it reaches the optimum point. In each step, show the KKT system, active set and solution.

7 Algorithms

7.1. Newton's Methods

(a) Why does Newton's method require that the Hessian matrix is positive definite?

(b) In the projected Newton's method, what do we do to the hessian matrix so that the algorithm applies for non-convex problems?

8 Programming

8.1. The following is the implementation of Newton's method. Indicate the wrong code and give the correction.

```
// get starting point
int n = _problem->n_unknowns();
Vec x(n);
_problem->initial_x(x);
// allocate gradient storage
Vec g(n);
// allocate hessian storage
SMat H(n, n);
// allocate search direction vector storage
Vec delta_x(n);
int iter(0);

Eigen::SimplicialLDLT<SMat> solver;
do {
    ++iter;

    // solve for search direction
    _problem->eval_gradient(x, g);
    _problem->eval_hessian(x, H);

    // solve for delta_x
    solver.compute(H);
    delta_x = solver.solve(g);

    // Newton decrement
    double lambda2 = g * delta_x;

    // Stop check
    if (lambda2 <= eps_) break;

    // step size
    double t = LineSearch::backtracking_line_search(...);

    // update
    x += t * delta_x;
} while (iter < _max_iters);
```