UNIVERSITY OF CALIFORNIA, BERKELEY

HW 7: EE127

Oscar Ortega

March 22, 2019

1 PSD EQUIVALENCE

- a:

$$\begin{bmatrix} X^T & Y^T \end{bmatrix} M \begin{bmatrix} X \\ Y \end{bmatrix}$$

$$= X^T AX + X^T BY + Y^T B^T X + Y^T CY$$
$$= X^T AX + 2Y^T B^T X + Y^T CY$$
$$\nabla_x = 2AX + 2Y^T B^T = 0$$
$$AX = -BY = 0$$

Because A is PD, this implies it is invertible.

$$X = -A^{-1}BY$$

$$\min_x = (-A^{-1}BY)^T A(-A^{-1}BY) + 2(-A^{-1}BY)^T BY + Y^T CY$$
$$= Y^T B^T A^{-1}BY - 2Y^T B^T A^{-1}BY + Y^T CY$$
$$= Y^T (C - B^T A^{-1}B)Y$$

- b: Proof $\rightarrow$:

$$\text{Let } (C - B^T A^{-1}B) \succeq 0$$
$$Y^T (C - B^T A^{-1}B)Y \geq 0$$

$$\min_x X^T AX + 2Y^T B^T X + Y^T CY \geq 0$$

$$\forall x \in \mathbb{R}^n : \begin{bmatrix} X^T & Y^T \end{bmatrix} M \begin{bmatrix} X \\ Y \end{bmatrix} \geq 0 \rightarrow M \geq 0$$

Proof $\leftarrow$:

$$\text{Let } M \geq 0$$

$$\forall x, y \in \mathbb{R}^n, \mathbb{R}^m : X^T AX + X^T BY + Y^T B^T X + Y^T CY \geq 0$$

$$\min_x X^T AX + X^T BY + Y^T B^T X + Y^T CY \geq 0$$

$$Y^T (C - B^T A^{-1} B) Y \geq 0$$

$$(C - B^T A^{-1} B) \geq 0$$

- c: Consider the following expression:

$$\begin{bmatrix} X^T & Y^T \end{bmatrix} M \begin{bmatrix} X \\ Y \end{bmatrix}$$

$$\min_y \begin{bmatrix} X^T & Y^T \end{bmatrix} M \begin{bmatrix} X \\ Y \end{bmatrix}$$

$$= X^T (A - B^T C^{-1} B) X$$

The rest will hold by symmetry:

2 Convexity of Functions

3 Visualizing Dual Problem

This section is on the iPython Notebook

4 Does Strong Duality Hold

- a: If we consider the following minimization:

$$\min_{x,y\in\mathscr{D}} e^{-x}$$

$$\text{s.t } \frac{x^2}{y} \leq 0$$

$$\mathscr{D} = \{(x,y) | y > 0\}$$

  We can reparametrize this optimization problem as follows:

$$\min_x e^{-x}$$

  Where the objective function is convex as the second derivative is strictly positive.

$$\text{s.t } x^2 \leq 0$$

  This problem is convex as the objective function is convex, and the inequality constraints are also convex.

  The optimal value of this function is 1 with optimizer $x^* = 0$

- b:

$$\mathscr{L}(x,\lambda) = e^{-x} + x^2\lambda$$

$$\frac{\partial \mathscr{L}}{\partial x} = -e^{-x} + 2x\lambda = 0$$

$$g(\lambda) = (2x + x^2)\lambda$$

$$\lambda^* = 0 \rightarrow \max_\lambda g(\lambda) = 0$$

  Here, the duality gap is 1.

- c:
  Here, because there there isn't a strictly feasible point as the only feasible solution is x = 0, we know slater's condition for strong duality does not hold.

5 MAGIC WITH CONSTRAINTS

- a: If we consider the following optimization problem, the critical points are the following set:

$$\mathscr{S} = \{1, -1\}$$

$$f_0(1) = (1)^3 - 3(1)^2 + 4 = 2$$
$$f_0(-1) = -(-1)^3 - 3(-1)^2 + 4 = 2$$

As it is the case the regions where the gradient of the function are not in the feasible set of this problem as well as positive and negative infinity. As we can see, the minimized value of this function is -2.

Recall the form of the Lagrangian:

$$\mathscr{L}(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{n} \lambda_i f_i(x) + \sum_{i=1}^{m} \nu_i h_i(x)$$

Where the $\lambda_i \geq 0, i = 1, ...n$. Applying this piece-wise we arrive at the following function.

$$\inf_x \mathscr{L}(x, \lambda) = g(\lambda) = \min \begin{cases} g_1(\lambda) = \inf_{x \geq 0} x^3 - 3x^2 + 4 - \lambda_1(x+1) + \lambda_2(x-1) \\ g_2(\lambda) = \inf_{x<0} - x^3 - 3x^2 + 4 - \lambda_1(x+1) + \lambda_2(x-1) \end{cases}$$

If we let $x = 2$:

$$\mathscr{L}(x, \lambda) = 2^3 - 3(2)^2 + 4 - \lambda_1(3) + \lambda_2 = -3\lambda_1 + \lambda_2$$

Similarly, if we now consider $x = -2$

$$\mathscr{L}(x, \lambda) = -(-2)^3 - 3(-2)^2 + \lambda_1 + \lambda_2(-3) = \lambda_1 - 3\lambda_2$$

$$g_1(\lambda) = \inf \mathscr{L}_1(x, \lambda)_{x \geq 0} \leq \mathscr{L}(2, \lambda) = -3\lambda_1 + \lambda_2$$

$$g_2(\lambda) = \inf \mathscr{L}_1(x, \lambda)_{x<0} \leq \mathscr{L}(-2, \lambda) = \lambda_1 - 3\lambda_2$$

For the sake of contradiction assume $g(\lambda) > 0$

$$\min(g_1(\lambda), g_2(\lambda)) > 0$$

$$g_1(\lambda) > 0, g_2(\lambda) > 0$$

$$-3\lambda_1 + \lambda_2 > 0, \lambda_1 - 3\lambda_2 > 0$$

$$\lambda_2 > 3\lambda_2, \lambda_1 > 3\lambda_2$$

$$\lambda_1 = \lambda_2 = 0$$

$$-3\lambda_1 + \lambda_2 = 0$$

$$g(\lambda) \leq 0$$

In this proof we also have shown that the maximum value of $g(\lambda) = g(0) = 0$.

This shows that strong duality does not hold as the maximum of the dual does not equal the maximum value of the primal.

- b: If we replace the two constraints with the quadratic constraint we can reformulate the Lagrangian as follows:

$$\inf_x \mathscr{L}(x,\lambda) = g(\lambda) = \min \begin{cases} g_1(\lambda) = \inf_{x\geq 0} x^3 - 3x^2 + 4 + \lambda(x^2-1) + \lambda_2(x-1) \\ g_2(\lambda) = \inf_{x<0} - x^3 - 3x^2 + 4 + \lambda(x^2-1) \end{cases}$$

$$\inf_x \mathscr{L}(x,\lambda) = g(\lambda) = \min \begin{cases} g_1(\lambda) = \inf_{x\geq 0} x^3 + (-3+\lambda)x^2 + 4 - \lambda \\ g_2(\lambda) = \inf_{x<0} - x^3 + (-3+\lambda)x^2 + 4 - \lambda \end{cases}$$

Note that this function is odd: $f(x) = -f(-x)$, therefore, we can reformulate this minimization as follows:

$$g(\lambda) = \inf_{x^2\leq 1} x^3 + (-3+\lambda)x^2 + 4 - \lambda$$

$$\frac{\partial \mathscr{L}}{\partial x} = x(3x + 2(-3+\lambda)) = 0$$

$$\mathscr{X} = \{0, 2 - \frac{2}{3}\lambda\}$$

We know the following points are local minima if the hessian is greater than 0.

$$\frac{\partial^2 \mathscr{L}}{\partial x^2} = 6x + 2(-3+\lambda) > 0$$

With $x = 0$:

$$\frac{\partial^2 \mathscr{L}}{\partial x^2} = 2(-3+\lambda) > 0$$

$$\lambda > 3$$

With $x = 2 - \frac{2}{3}\lambda$:

$$\frac{\partial^2 \mathscr{L}}{\partial x^2} = 6(2 - \frac{2}{3}\lambda) + 2(-3+\lambda) > 0$$

$$\lambda < 3$$

Let $\lambda > 3$:

$$\mathscr{L}(0,\lambda) = 4 - \lambda < 1$$

If we check the boundaries:

$$\mathscr{L}(1,\lambda) = \mathscr{L}(-1,\lambda) = 2$$

Therefore:

$$\inf_{\lambda>3} \mathscr{L}(x,\lambda) = 4 - \lambda$$

Note, this will still hold when the value of $\lambda = 3$ Let $0 \leq \lambda < 3$

$$\mathscr{L}(2 - \frac{2}{3}\lambda, \lambda) = (2 - \frac{2}{3}\lambda)^3 + (-3+\lambda)(2 - \frac{2}{3}\lambda)^2 + 4 - \lambda$$

$$= (3-\lambda)^3\left(\frac{8}{27} + \frac{-3+\lambda}{3-\lambda}\frac{4}{9}\right) + 4 - \lambda$$

$$= (3-\lambda)^3\left(\frac{8}{27} - \frac{12}{27}\right) + 4 - \lambda$$

$$\frac{-4}{27}(3-\lambda)^3 + 4 - \lambda$$

Similarly, this value is less than the value of the boundary points. Therefore:

$$g_1(\lambda) = g_2(\lambda) = \begin{cases} 4-\lambda & \lambda \geq 3 \\ -\frac{4}{27}(3-\lambda)^3 + 4 - \lambda & 0 \leq \lambda < 3 \end{cases}$$

$$\sup g_1(\lambda)_{\lambda\geq 3} = 1$$

$$\frac{\partial g}{\partial \lambda} = -\frac{12}{27}(3-\lambda)^2 - 1 = 0$$

$$(3-\lambda)^2 = 1$$

$$\lambda = \{\frac{3}{2}, \frac{9}{2}\}$$

We only consider $\frac{3}{2}$ because its in the domain.

$$g_1(\frac{3}{2}) = -\frac{4}{27}(3-\frac{3}{2})^3 + 4 - \lambda$$

$$d^* = g(\frac{3}{2}) = 2$$

Here, strong duality is achieved as the optimal of the primal is equal to the optimal of the dual.

6 KKT Conditions

- a:

  If we consider the following minimization:

$$\min_{x\in\mathbb{R}^n} \sum_{i=1}^{n} \frac{1}{2} d_i x_i^2 + r_i x_i$$

$$\text{s.t} a^T x = 1, x_i \in [-1,1], i = 1,...,n$$

  If we let $D = \text{diag}(d1,...,d_n)$ and let $r = (r_1,...,r_n)$. Then the following holds.

$$\sum_{i=1}^{n} \frac{1}{2} x_i^2 d_i = \frac{1}{2} x^T D x$$

  As this is just a quadratic form of a diagonal form.
  Furthermore:

$$\sum_{i=1}^{n} r_i x_i = r^T x$$

  as the following is just the definition of the dot product. Finally, $x_i \in [-1,1] \to x_i^2 \leq 1$. Therefore, the problem can be rewritten to the following form:

$$\min_x \frac{1}{2} x^T D x + r^T x$$

$$\text{s.t} a^T x = 1$$

$$x_i^2 \leq 1$$

- b: Taking derivatives with respect to x:

$$\nabla f_{0x} = Dx + r$$

$$\nabla f_{1x} = a$$

  And for the inequality constraints:

$$\nabla f_i(x)_x = 2x_i, i = 1,...,n$$

  We can verify that strong duality holds through Slater's condition.

  Consider the vector $\tilde{x} = \frac{a}{\|a\|_2^2}$

$$a^T \tilde{x} = a^T \frac{a}{\|a\|_2^2} = \frac{a^T a}{a^T a} = 1$$

$$\|\tilde{x}\|_2^2 = \sum_{i=1}^{n} \frac{{a_i}^2}{\sum_{i=1}^{n} a_i^2}$$

$$x_i^2 < 1, i = 1, ..., n$$

Therefore, Slater's condition holds.

Based on the D matrix given, we can see the hessian of objective is just D itself, which is positive semi-definite, and is therefore convex. The inequality can be seen to be an affine transformation, and is in fact linear. Taking second derivatives of the constraint functions, we can see the function is convex as the hessian, which is equal to 2, is positive-semi-definite. Therefore, the optimization problem is convex.

These conditions together imply strong duality.

- c: Recall the definition of the Lagrangian:

$$\mathcal{L}(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{n} \lambda_i f_i(x) + \sum_{i=1}^{m} \nu_i h_i(x)$$

$$\mathcal{L}(x, \lambda, \mu) = \frac{1}{2} x^T D x + r^T x + \sum_{i=1}^{n} \lambda_i (x_i^2 - 1) + (\mu(a^T x - 1))$$

$$= \frac{1}{2} x^T D x + \sum_{i=1}^{n} \lambda_i (x_i^2 - 1) + (r + \mu a)^T x - \mu$$

$$\frac{1}{2} x^T D x + x^T \Lambda x - \sum_{i=1}^{n} \lambda_i + (r + \mu a)^T x - \mu$$

If we let $\lambda' = \frac{\lambda}{2}$, We can reduce this as follows:

$$\frac{1}{2} x^T D x + \frac{1}{2} x^T \Lambda x - \frac{1}{2} \sum_{i=1}^{n} \lambda_i + (r + \mu a)^T x - \mu$$

$$= \frac{1}{2} x^T (D + \Lambda) x + (r + \mu a)^T x - (\mu + \frac{1}{2} \sum_{i=1}^{n} \lambda_i)$$

The KKT conditions are as follows:

  - $f_i(x) \le 0, i = 1, ..., n$
  - $h_i(x) = 0, i = 1, ..., m$
  - $\lambda_i \ge 0, i = 1, .., n$
  - $\lambda_i f_i(x) = 0, i = 1, ..., n$
  - $\mathcal{L}(x^*, \lambda^*, \nu^*) = 0$

- d:

$$\nabla_x \mathscr{L}(x,\lambda,\mu) = (D+\Lambda)x + (r+\mu a) = 0$$
$$x^* = -(D+\Lambda)^{-1}(r+\mu a)$$
$$x_i^* = -\frac{r_i+\mu a_i}{d_i+\lambda_i}$$

- e:

  Let $A = (D+\Lambda)$

$$\mathscr{L}(x^*,\mu,\lambda) = \frac{1}{2}(r+\mu a)^T - A^{-1}A - A^{-1}(r+\mu a) - (r+\mu a)^T A^{-1}(r+\mu a) - (\mu + \frac{1}{2}\sum_{i=1}^{n}\lambda_i)$$

$$\mathscr{L}(x^*,\mu,\lambda) = \frac{1}{2}(r+\mu a)^T A^{-1}(r+\mu a) - (r+\mu a)^T A^{-1}(r+\mu a) - (\mu + \frac{1}{2}\sum_{i=1}^{n}\lambda_i)$$

$$\mathscr{L}(x^*,\mu,\lambda) = \frac{-1}{2}(r+\mu a)^T A^{-1}(r+\mu a) - (\mu + \frac{1}{2}\sum_{i=1}^{n}\lambda_i)$$

$$\mathscr{L}(x^*,\mu,\lambda) = g(\mu,\lambda) = -\mu - \frac{1}{2}\sum_{i=1}^{n}\left(\frac{(r_i+\mu a_i)^2}{d_i+\lambda_i} + \lambda_i\right)$$

- f:

$$\mathscr{L}(x^*,\mu,\lambda) = g(\mu,\lambda) = -\mu - \frac{1}{2}\sum_{i=1}^{n}\left(\frac{(r_i+\mu a_i)^2}{d_i+\lambda_i} + \lambda_i\right)$$

$$\frac{\partial g}{\partial \lambda_i} = \frac{1}{2}(r_i+\mu a_i)^2(d_i+\lambda_i)^{-2} - \frac{1}{2} = 0$$
$$= (r_i+\mu a_i)^2 = (d_i+\lambda_i)^2$$
$$(r_i+\mu a_i) = (d_i+\lambda_i)$$

  However we ignore the negative quantity because it is not a feasible point.

$$\lambda_i^* = (r_i+\mu a_i) - d_i$$

- g:

  Recall:

$$x_i^* = -\frac{r_i+\mu a_i}{d_i+\lambda_i}$$

$$x_i^* = -\frac{r_i+\mu a_i}{d_i+(r_i+\mu a_i)-d_i}$$

$$x_i^* = -1$$

- h:

  Recall that for complementary slackness to hold:

  $$\lambda_i f_i(x) = 0, i = 1, \ldots, m$$

  Substituting our values of λ_i and $f_i(x_i^*)$:

  $$\lambda_i f_i(x_i^*) = (r_i + \mu a_i) - d_i f_i(-1)$$

  $$(r_i + \mu a_i) - d_i((-1)^2 - 1))$$

  $$= 0$$

7 Trust Region Subproblem (Part 2)

- a: We can express the Lagrangian as follows:

  $$\mathscr{L}(z,\nu) = \sum_{i=1}^{n} \lambda_i z_i - 2|d_i|\sqrt{z_i} + \nu\left(\sum_{i=1}^{n}(z_i - 1)\right)$$

  Strong duality holds as there exists $z_1, ..., z_n$ whose sum equal to 1, which implies there exists $z_1, ..., z_n$ in the relative interior of the objective as the equality constraints are affine.

  $$\frac{\partial \mathscr{L}}{\partial z_i} = \lambda_i - \frac{|d_i|}{\sqrt{z_i}} + \nu = 0$$

  $$z_i^* = \left(\frac{|d_i|}{\lambda_i + \nu}\right)^2$$

  $$g(\nu) = \sum_{i=1}^{n} \lambda_i \left(\frac{|d_i|}{\lambda_i + \nu}\right)^2 - 2|d_i|\left(\frac{|d_i|}{\lambda_i + \nu}\right) + \nu \sum_{i=1}^{n} \left(\frac{|d_i|}{\lambda_i + \nu}\right)^2 - 1$$

  $$g(\nu) = \sum_{i=1}^{n} \lambda_i \left(\frac{|d_i|}{\lambda_i + \nu}\right)^2 - 2|d_i|\left(\frac{|d_i|}{\lambda_i + \nu}\right) + \nu \sum_{i=1}^{n} \left(\frac{|d_i|}{\lambda_i + \nu}\right)^2 - 1$$

- b:

  Given optimal variables ν^* we could recover optimal variable z^* by plugging in the optimized ν, we could compute $z_i^*, i = 1, ..., n$ we could then apply the inverse transformations on the optimal z values to retrieve the value of x^*. In this case we know that for every $z_i = \pm y_i^2 = \pm x_i^2$

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
%matplotlib inline
```

Duality

In this problem we look at the primal and dual problems for simple cases and get insights into duality

Helper functions

In [2]:
```
## DO NOT MODIFY THIS CODE
# Wrapper function to get primal functions
def get_primal(subpart):
    if subpart == 0:
        x = np.linspace(-2, 8, 100)
        return x, get_primal_lp(x), get_p_opt_lp(x), get_primal_constraint_lp
()
    elif subpart == 1:
        x, y = np.linspace(-2, 2, 100), np.linspace(-2, 2, 100)
        X, Y = np.meshgrid(x, y)
        return X, Y, get_primal_e(X, Y), get_primal_constraint_e(x, y), get_p_
opt_e(np.linspace(0, 10, 100))
    elif subpart == 2:
        # TODO @course staff
        x = np.linspace(-5, 5, 100)
        return
    else:
        print("Subpart must be 0, 1, or 2.")

# Wrapper function to get dual functions
def get_dual(subpart):
    if subpart == 0:
        lam = np.linspace(-2, 8, 100)
        dual_constraints = get_dual_constraints_lp()
        return lam, get_dual_lp(lam), dual_constraints
    elif subpart == 1:
        lam = np.linspace(10**-10, 4, 100)
        return lam, get_dual_e(lam)
    elif subpart == 2:
        return
    else:
        print("Subpart must be 0, 1, or 2.")

# Helper function for plotting. Do not change this function.
def visualize(subpart):
    p = get_primal(subpart)
    d = get_dual(subpart)
    if subpart == 0: # 1d primal and dual variables
        (lam, dual, dual_constraints) = d
        x, primal, p_opt, primal_constraint = p
        fig, (ax1, ax2) = plt.subplots(nrows=1, ncols=2, sharey=True, figsize
= [12,6])
        ax1.plot(x, primal, color="b", label=r'$f(x)$')
        ax1.plot(x, p_opt, color="k", linestyle="dotted", label=r'$p^* = 6$')
        ax1.axvspan(-2, primal_constraint, alpha=0.5, facecolor="r")
        ax1.set_xlim([-2, 8])
        ax1.legend(loc="upper left")
        ax1.set_title("Primal")
        ax1.set_xlabel(r'$x$')
        ax1.set_ylabel(r'$f(x)$')


        ax2.plot(lam, dual, color="g", label=r'$g(\lambda)$')
        ax2.plot(x, p_opt, color="k", linestyle="dotted", label=r'$p^* = 6$')
#         ax2.plot(dual_constraints[0], dual_constraints[1], color="y")
        ax2.axvspan(-2, dual_constraints[0], alpha=0.5, facecolor="r")
```

```
        ax2.axvspan(dual_constraints[1], 8, alpha=0.5, facecolor="r")

        ax2.legend(loc="upper left")
        ax2.set_title("Dual")
        ax2.set_xlabel(r'$\lambda$')
        ax2.set_ylabel(r'$g(\lambda)$')

    elif subpart == 1: # 2d primal variables
        (lam, dual) = d
        X, Y, primal, primal_constraint, p_opt = p
        plt3d = plt.figure().gca(projection='3d')
        plt3d.plot_surface(X, Y, primal_constraint, alpha=.2, color = 'red')
        ax = plt.gca()
        ax.contour3D(X, Y, primal, 50, cmap='binary')
        ax = plt.gca()
        ax.set_xlabel(r'$x$')
        ax.set_ylabel(r'$y$')
        ax.set_zlabel(r'$f(x,y)$')
        ax.set_title("Primal")

        fig2, (ax1, ax2, ax3) = plt.subplots(nrows=1, ncols=3, sharey=True, fi
gsize = [12,6])

        x = np.linspace(0, 2, 100)
        ax1.set_title(r'Cross section: $f(x,y)$ s.t. $x + y = 1$')
        ax1.plot(x, np.exp(x) + np.exp(1-x), label=r'$f(x,y)$')
        ax1.plot(x, p_opt, color="k", linestyle="dotted", label=r'$p^*$')
        ax1.set_xlabel(r'$x$')
        ax1.legend(loc="upper right")

        ax2.set_title(r'Cross section: $f(x,y)$ s.t. $y = \frac{1}{2}$')
        ax2.plot(x, np.exp(x) + np.exp(.5), label=r'$f(x,y)$')
        ax2.plot(x, p_opt, color="k", linestyle="dotted", label=r'$p^*$')
        ax2.axvspan(0, .5, alpha=0.5, facecolor="r")
        ax2.set_xlabel(r'$x$')
        ax2.legend(loc="upper right")

        ax3.plot(lam, dual, label=r'$g(\lambda)$')
        ax3.plot(lam, p_opt, color="k", linestyle="dotted", label=r'$p^*$')
        ax3.set_title("Dual")
        ax3.set_xlabel(r'$\lambda$')
        ax3.legend(loc="upper right")

    elif subpart == 2:
        # TODO
        return
    else:
        print("Subpart must be 0, 1, or 2.")
    plt.legend()
```

Problem (1): A Linear program

Problem formulation

Consider the problem of minimizing a linear objective subject to a linear constraint. Suppose a is a positive scalar.

$$p^* = \min_{x \geq 0} ax$$
$$\text{s.t. } x \geq b.$$

We consider the case where $a = 2$ and $b = 3$. In this case clearly the optimal primal solution is $x^* = 3$ and $p^* = 6$.

Constructing the primal

In [3]:
```
## DO NOT MODIFY THIS CODE
a = 2
b = 3

##Get primal for plottting
def get_primal_lp(x):
    """ x: array of scalars """
    return np.multiply(x, a)

def get_p_opt_lp(x):
    """
    Used for plotting the optimal value p*.
    Input:
    Output:
    """
    return [a*b for _ in x]

def get_primal_constraint_lp():
    """
    Used for plotting feasible x region
    """
    return b
```

Part a) Formulate the dual problem for the LP which involves maximizing $g(\lambda_1, \lambda_2)$ where λ_1 is the dual variable corresponding to the constraint $x \geq 2$ and λ_2 is the dual variable corresponding to the constraint $x \geq 0$. Solve it to obtain d^*.

$$\mathcal{L}(x, \lambda_1, \lambda_2) = ax + \lambda_1(3 - x) + \lambda_2(-x)$$
$$\frac{\partial \mathcal{L}}{\partial x} = a - \lambda_1 - \lambda_2 = 0$$
$$\min_x \mathcal{L}(x, \lambda_1, \lambda_2) = g(\lambda_1, \lambda_2) = (a - \lambda_1 - \lambda_2)x + 3\lambda_1$$
$$\max_{\lambda_2} \lambda_2 = 0$$
$$(a - \lambda_1)x + 3\lambda_1$$
$$\text{argmax}_{\lambda_1}(a - \lambda_1)x + 3\lambda_1 = a$$
$$d^* = 3a$$

Next we will consider the Lagrangian only with respect to the constrain $x \geq 3$ and form the dual problem.

Part b) Formulate the dual problem for the LP which involves maximizing $g(\lambda)$ where λ is the dual variable corresponding to the constraint $x \geq 3$. Compare the dual problem to that of the previous part.

$$\mathcal{L}(x, \lambda) = ax + \lambda(3 - x)$$
$$(a - \lambda)x + 3\lambda$$

So we can reformulate the dual as follows:

$$\min_x \mathcal{L}(x, \lambda) = g(\lambda) = (a - \lambda)x + 3\lambda : \lambda \leq a$$
$$\text{argmax}_{\lambda}(a - \lambda)x + 3\lambda = a$$
$$d^* = 3a$$

TODO Fill this cell with the dual problem

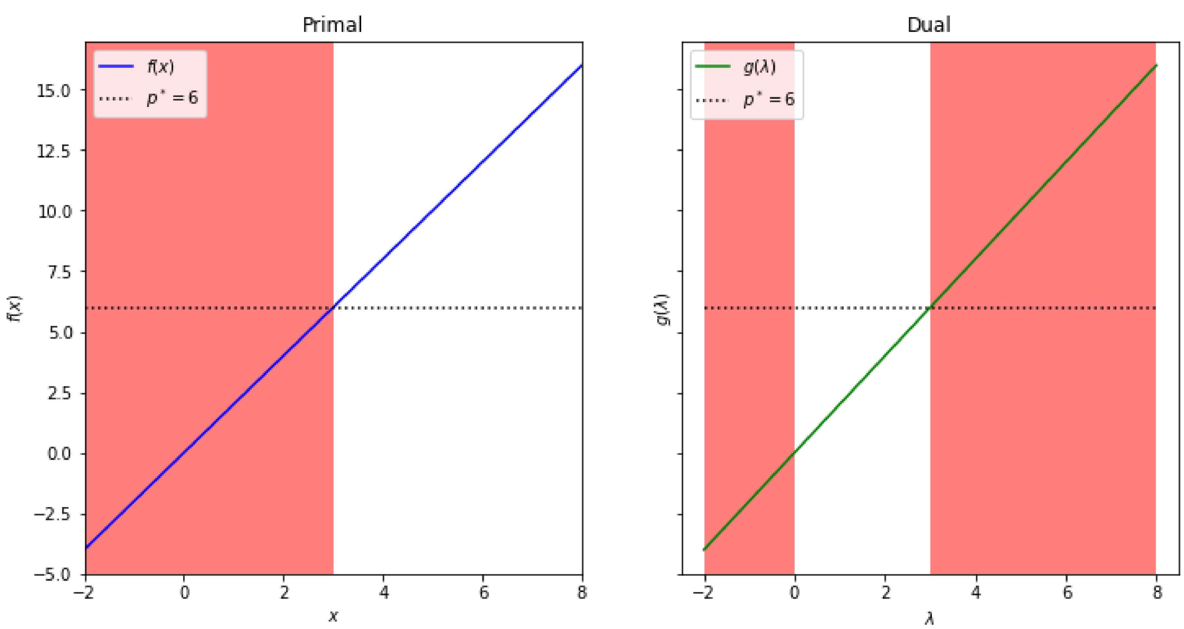
Part c) Fill in the get_dual_lp function and get_dual_constraints function based on the dual problem from part b

In [4]:
```
### Constructing the dual

def get_dual_lp(lam):
    """
    The objective function of the dual problem. Takes in lam and evaluate g(la
m).
    Input: lam: array of scalars
    Output: g(lam)
    """
    # x, get_primal_lp(x), get_p_opt_lp(x), get_primal_constraint_lp()
    # TODO: Replace the return value with the dual function, g(lambda
    x, primal, opt, constraint = get_primal(0)
    return (a * lam)

def get_dual_constraints_lp():
    """
    Get the bounds of the constraints that limit the feasible set of the dual
 variable.
    Output: Return two floats, lam_low, lam_high so that the dual variable is
    constrainedas lam_low <= lam <= lam_high
    """
    lam_low = 0
    lam_high = 3

    return lam_low, lam_high
```

Part d) Based on the visualization of the primal and dual problem answer the following:

Find the dual optimal solution λ^*, where red region denotes region of infeasibility. Does strong duality hold?

In [5]:
```
visualize(0)
```



TODO Fill answer to part d)

Strong Duality holds, because it is the case $p^* = d^*$.

Problem (2): Minimizing exponentials in $\mathbb{R}^2$

Problem formulation

Consider the following problem, where $z = [x, y]$ is the two-dimensional primal variable.

$$p^* = \min_{x,y} e^x + e^y$$
$$\text{s.t. } x + y \geq 1.$$

Part e) Solve the primal problem to get p^* and primal optimal solutions x^*, y^*.

Hint:

First show that the objective function is componentwise increasing in x and y. Then show that if (x, y) satisifies strict inequality with respect to constraint then we can decrease either x or y while not violating constraint and simultaneously decrease objective value. Thus at optimality we must have $x + y = 1$.

TODO Fill in solution here

$$\text{Let } h(x, y) = e^x + e^y$$
$$\frac{\partial^2 h}{\partial y^2} = e^y$$
$$\frac{\partial^2 h}{\partial x^2} = e^x$$

Both of these have positive second derivatives componentwise, which means these functions are both componentwise increasing. Assume $(x + y) > 1$

$$\rightarrow \exists \epsilon > 0$$
$$\text{s.t}(x - \epsilon, y) \text{ or } (x, y - \epsilon) \text{ is feasible}$$

Because we know this objective function is component wise increasing, we know either one of these values will have a value less than the original point. This implies that at optimality, $x + y = 1$

Constructing the primal

In [6]:
```
def get_primal_e(x, y):
    """ x, y: arrays of scalars """
    return np.add(np.exp(x), np.exp(y))

def get_primal_constraint_e(x, y):
    return np.reshape(x + y - 1, [-1,1])

def get_p_opt_e(x):
    """ Used for plotting the optimal value p*. """
    return [2 * np.exp(.5) for _ in x]
```

Part f) Does Slater's condition hold? Is the problem convex? Does strong duality hold?

Slater's condition does hold because, we know there exists a strictly feasible point in the domain of the optimization problem. From part e, the second order conditions of this problem show this problem is strictly convex. This is sufficient to say that strong duality holds.

Part g) Formulate the dual problem for the minimizing exponentials problem, which involves maximizing $g(\lambda)$ where λ is the dual variable corresponding to the constraint $x + y \geq 1$.

TODO Fill this cell with the dual problem

$$\mathcal{L}(x, y, \lambda) = (e^x + e^y) + \lambda(1 - x - y)$$
$$\frac{\partial \mathcal{L}}{\partial y} = e^y - \lambda = 0$$
$$y^* = \ln(\lambda)$$
$$\frac{\partial \mathcal{L}}{\partial x} = e^x - \lambda = 0$$
$$x^* = \ln(\lambda)$$
$$\min_{x,y} \mathcal{L}(\lambda, x, y) = g(\lambda) = \lambda(3 - 2\ln(\lambda)$$

Part h) Fill in the get_dual_lp function function based on the description given

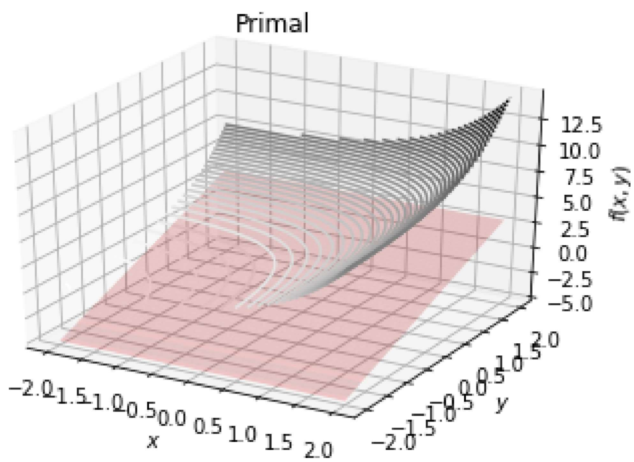
In [7]:
```
def get_dual_e(x):
    """ x: array of scalars """
    # TODO: Replace the return value with the dual function, g(lambda)
    return x * (3 - (2 *np.log(x)))
def get_dual_constraints_e():
    return (0,5)
```

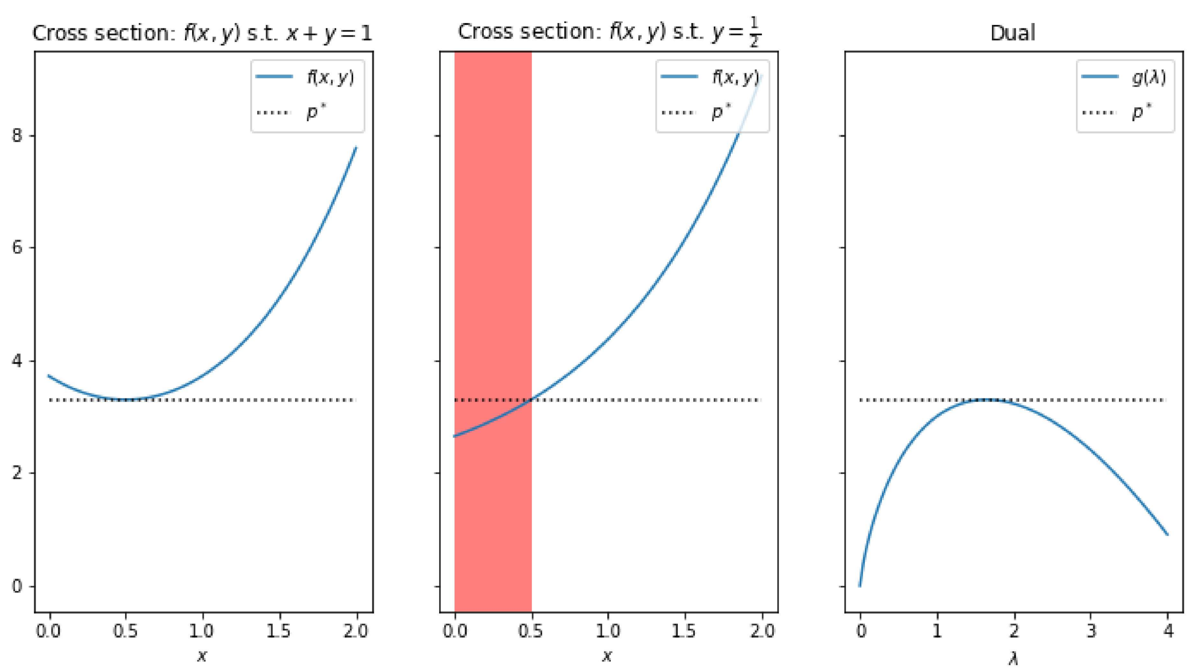
Part i) Find the dual optimal solution by solving the dual problem, λ^*.

TODO Fill in solution here

$$
\begin{gathered}
g(\lambda) = \lambda(3 - 2\ln(\lambda)) \\
\frac{\partial g}{\partial \lambda} = 1 - 2\ln(\lambda) = 0 \\
\frac{1}{2} = \ln(\lambda) \\
e^{\frac{1}{2}} = \lambda^* \\
g(\lambda^*) = 2e^{\frac{1}{2}}
\end{gathered}
$$

In [8]:
```
visualize(1)
```





First note that while the primal problem was over $x \in \mathbb{R}^2$ the dual problem is over scalar variable $\lambda \in \mathbb{R}$. Further the dual objective function $g(\lambda)$ is concave and can be maximized by simply setting derivative of $g(\lambda)$ with respect to λ to 0. Thus it is much easier to solve the dual problem as compared to the primal problem. By appealing to Slater's condition and observing that the problem was convex we know that strong duality holds so once we solve the dual for d^* we know p^*.