CVX demo

CS 709

CVX

- CVX is an open source MATLAB-based modeling tool.
- The optimization problem has to be a convex optimization problem
- Convex Optimization programming made easy
- Matlab toolbox
  - Allows you to flexibly express convex optimization problems
  - Solves LP, QP, SOCP, SDPs
- CVX is not for large scale problems
- Programs must follow Disciplined Convex Programs (DCP) rules. Otherwise, cannot solve even if convex

CVX

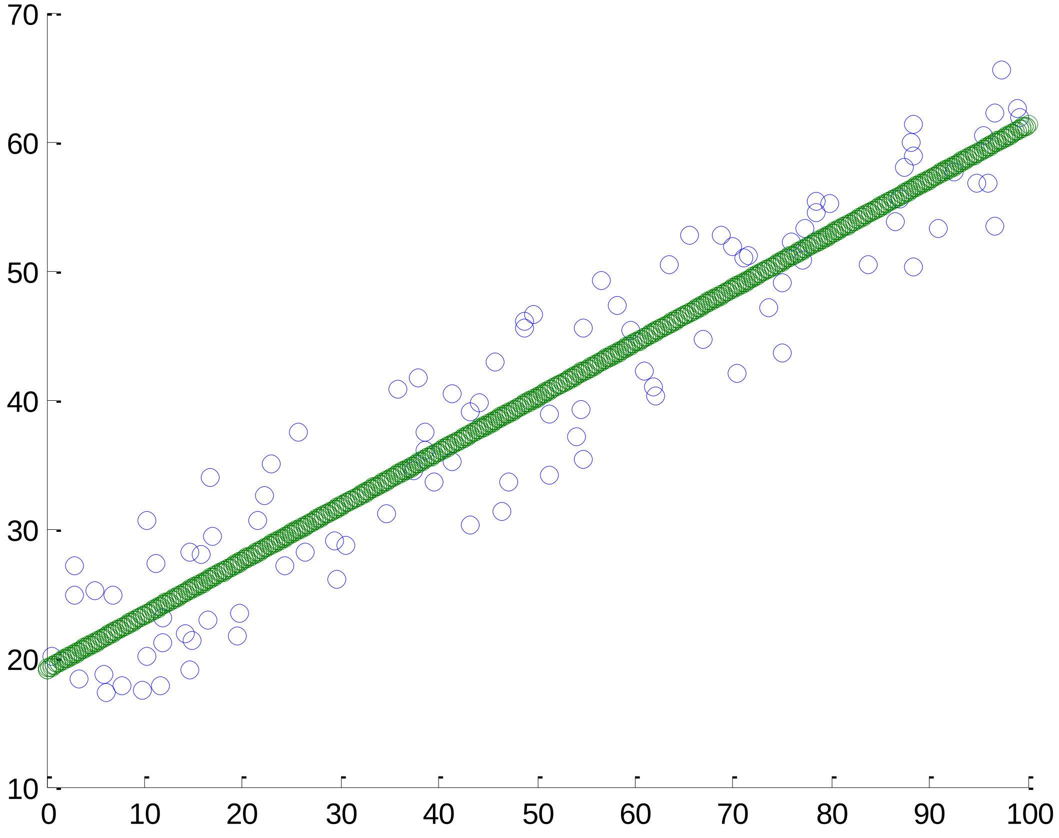
- Core solvers used in CVX:
  - SeDuMi (http://sedumi.ie.lehigh.edu/)
  - SDPT3 (http://www.math.nus.edu.sg/~mattohkc/sdpt3.html)
- Both are open-source interior-point solvers based on MATLAB.
- CVX converts the problem into a format accepted by those solvers and call them to solve the problem.

Example-1

- Least Square
  - Data points $\{(x_i, y_i)\}_{i=1}^{i=N}$
  - Fit a line with least squared error
  - Objective:

$$\min_{m,c} \sum_{i=1}^{n} \left((mx_i + c) - y_i\right)^2$$

Example-1



Least Square: Matlab Style

- Matlab’s quadratic optimization API
  - quadprog(H,f,A,b,Aeq,beq,lb,ub)

$$\min_{X}\left(\frac{1}{2}X^{T}HX+f^{T}X\right)$$

subject to

$$AX \leq b$$
$$AeqX = beq$$
$$lb \leq X \leq ub$$

Least Square: Matlab Style

- Express LS objective in matlab format
- $\sum_{i=1}^{n}\left((mx_i + c) - y_i\right)^2$
- Prepare
  - H
  - f

Least Square: Matlab Style

- Expanding

$$\sum_{i=1}^{n} \left(m^2 x_i^2 \ + c^2 + y_i^2 + 2mcx_i - 2cy_i \right.$$

- $H = \begin{bmatrix} \sum_i 2x_i^2 & \sum_i 2x_i \\ \sum_i 2x_i & 2n \end{bmatrix}$
- $f = \begin{bmatrix} -2\sum_i x_i y_i \\ -2\sum_i y_i \end{bmatrix}$

Least Square: Matlab Style

```
function [p, fval] = solve_matlab_style(x, y)
    H=2*[sum(square(x)), sum(x); sum(x), size(x,1)];
    f=[-2*sum(x.*y); -2*sum(y)];
    A=[];
    b=[];
    [p, fval]=quadprog(H,f,A,b);

    display(p);
    display(fval);
end
```

Least Square: CVX Style

```
function [p, fval] = solve_cvx_style(x, y)
    cvx_begin
        variable p(2);
        minimize( sum(square(x*p(1) + p(2) - y)) );
    cvx_end

    fval = cvx_optval;

    display(p);
    display(fval);
end
```

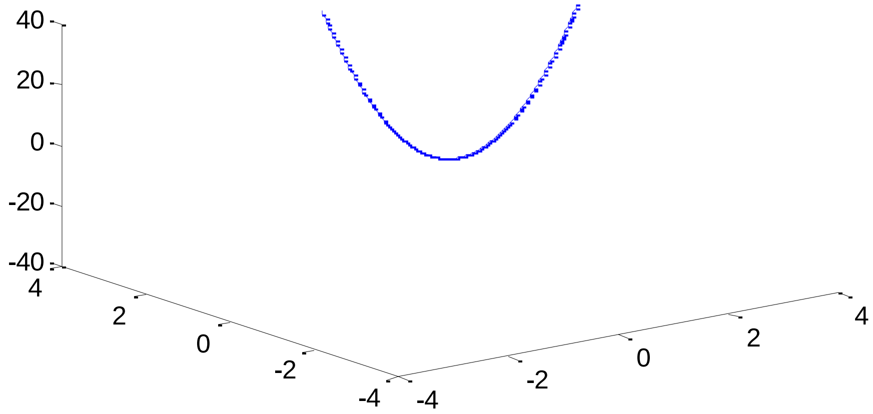
Example with Constraints

```
function ex1
   cvx_begin
      variable x1;
      variable x2;
      minimize( 0.5*x1*x1 + x2*x2  - 2*x1 - 6*x2);
      subject to
         x1 + x2 <= 2;
         x1 + 2*x2 <= 2;
         2*x1 + x2 <= 3;
         0 <= x1, 0 <= x2;
   cvx_end

   display(x1);
   display(x2);
   display(cvx_optval);
end
```

DCP violation

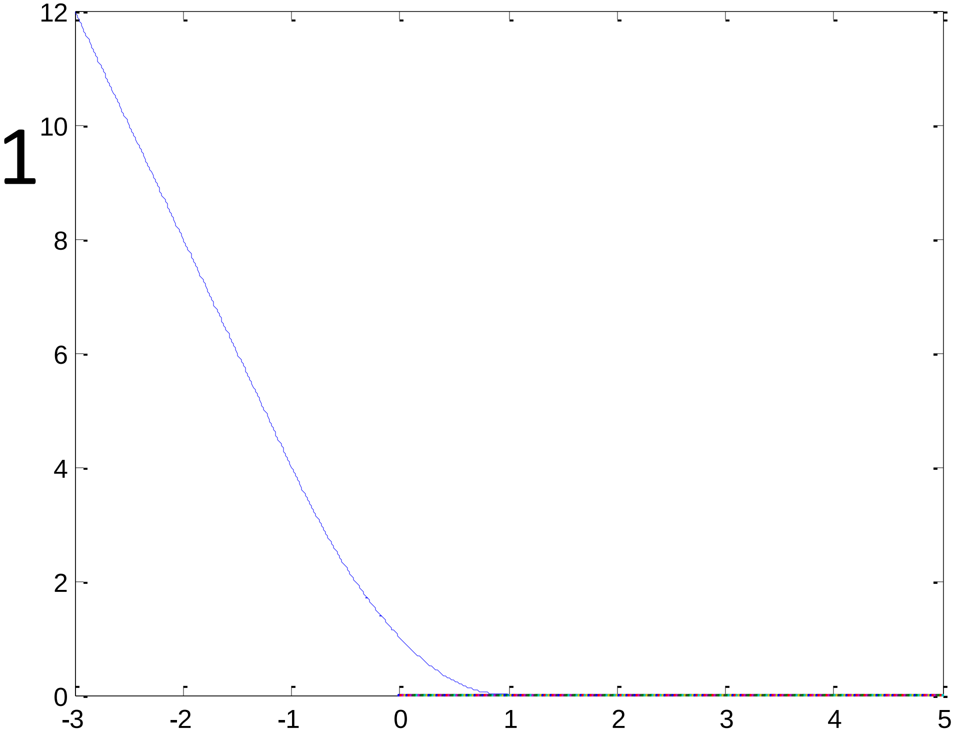
- minimize(0.5*x1*x1 + x2*x2 – x1*x2 - 2*x1 - 6*x2);
  - – Cannot be solved as is
    - Product of two variables is not convex
    - Violates DCP rule



- Rearrange
  minimize(square(0.5*x1-x2) + 0.25*x1*x1 - 2*x1 - 6*x2);

Limitations

- Not for large scale problems
- Cannot solve convex problems if DCP not followed.
- L(t) = 0 for t >= 1
  L(t) = (1-t)^2 for |t| < 1
  L(t) = -4t for t <= -1



References

- [1] http://cvxr.com/cvx/cvx_usrguide.pdf
- [2] http://www.mathworks.in/help/optim/ug/quadprog.html

Try it out…

- **Steps to start matlab and work with CVX**

1. Login to 10.129.1.105 using following credentials: guest/guest709
2. Start matlab: ~/matlab/bin/matlab
3. Change directories to CVX at matlab prompt: cd /home/guest/cvx
4. Run command at matlab prompt: cvx_setup
5. Now you can start running your cvx programs. For details refer to http://cvxr.com/cvx/doc/CVX.pdf